

# SIEMENS

## SIMATIC

### S7

### S7-1200 Programlanabilir kontrolör

#### Sistem Kılavuzu

#### Önsöz

Ürüne genel bir bakış	1
STEP 7 programlama yazılımı	2
Kurulum	3
PLC konseptleri	4
Cihaz konfigürasyonu	5
Programlama konseptleri	6
Temel komutlar	7
Genişletilmiş komutlar	8
Teknoloji komutları	9
Haberleşme	10
Web sunucusu	11
Haberleşme işlemcisi ve Modbus TCB	12
Teleservis haberleşme (SMTP email)	13
Online ve diyognastik araçları	14
SM 1278 4xIO-Link Master	15
Teknik Özellikler	A
Güç ihtiyacı hesaplaması	B
Sipariş numaraları	C
Bir V3.0 CPU'yu V4.0 CPU ile değiştirme	D

## Yasal bilgiler

### Uyarı bilgi sistemi

Bu kılavuz, kişisel güvenliğinizi sağlamak ve mülkiyete gelebilecek zararları önlemek için uymak zorunda olduğunuz hususları içermektedir. Kişisel güvenliğinize yönelik bu uyarılar, kılavuzda güvenlik uyarı sembolleriyle vurgulanmaktadır; yalnızca mülk zararlarına ilişkin uyarılarda herhangi bir güvenlik uyarı sembolü bulunmamaktadır. Aşağıdaki uyarılar, tehlike düzeyine göre sınıflandırılmıştır

<b>⚠ TEHLİKE</b>
uygun önlemler alınmadığı takdirde ölüm veya ağır kişisel yaralanmaların meydana geleceğini gösterir.
<b>⚠ UYARI</b>
uygun önlemler alınmadığı takdirde ölüm veya ağır kişisel yaralanma olabileceğini gösterir.
<b>⚠ DİKKAT</b>
uygun önlemler alınmadığı takdirde küçük kişisel yaralanma meydana gelebileceğini gösterir.
<b>BİLGİ</b>
uygun önlemler alınmadığı takdirde mülke zarar gelebileceğini gösterir.

Birden fazla tehlike derecesi bulunursa en yüksek tehlike derecesini temsil eden uyarı bilgisi kullanılacaktır. Emniyet uyarı sembolü ile kişiler için bir yaralanma uyarı bildirimini, mal hasarına ilişkin bir uyarıyı da içerebilir.

### Uzman Personel

Bu dokümantasyonda açıklanan ürün/sistem, yalnızca belirli görevde ilgili dokümantasyona uygun uzman personel tarafından özellikle güvenlik talimatlarına ve uyarı notlarına göre çalıştırılabilir. Uzman personel, eğitim ve deneyimine bağlı olarak bu ürünler/sistemler ile çalışırken riskleri tanımlama ve potansiyel tehlikelerden sakınma yeterliliğine sahip kişilerdir.

### Siemens ürünlerinin uygun kullanımı

Aşağıdaki hususlara dikkat ediniz:

<b>⚠ DİKKAT</b>
Siemens ürünleri yalnızca bu katalogta ve ilgili teknik dokümanlarda açıklanan uygulamalar için kullanılabilir. Başka üreticilerin ürünleri ve bileşenleri kullanılırsa, bunlar Siemens tarafından tavsiye edilmeli veya onaylanmalıdır. Ürünlerin emniyetli ve sorunsuz çalışması için uygun nakliye, depolama, kurulum, montaj, devreye alma, çalıştırma ve bakım gereklidir. İzin verilen ortam koşullarına uyulmalıdır. İlgili dokümanlardaki bilgiler dikkate alınmalıdır.

### Ticari markalar

© işareti ile tanımlanan bütün isimler Siemens AG'nin kayıtlı ticari markalarıdır. Bu yayındaki diğer ticari markalar, üçüncü taraflar tarafından kendi amaçları için kullanıldığında sahibinin haklarının ihlal edebileceği ticari markalar olabilir.

### Sorumluluğun reddi

Biz bu dokümanı, içerik olarak açıklanan donanım ve yazılım ile uyumlu olduğundan emin olmak amacıyla yeniden inceledik. Tutarsızlıklar tamamen engellenemediği için tam uyumu garanti edemeyiz. Ancak bu yayındaki bilgiler düzenli olarak gözden geçirilmekte ve gerekli düzeltmeler sonraki baskılarda yer almaktadırlar.

# Önsöz

## Kılavuzun amacı

S7-1200 serisi, çeşitli otomasyon uygulamalarını kontrol edebilen programlanabilir lojik kontrolörlerin (PLC) bir devamıdır. Kompakt tasarım, düşük maliyet ve güçlü bir komut setine sahip olan S7-1200 çok çeşitli uygulamaların kontrolü için mükemmel bir çözüm sunar. S7-1200 modelleri ve Windows-tabanlı STEP 7 programlama aracı (Sayfa 35) size otomasyon sorunlarınızı çözmeye ihtiyaç duyduğunuz esnekliği sağlar.

Bu kılavuz, S7-1200 PLC'lerin kurulumu ve programlanması hakkındaki bilgileri sağlamaktadır ve programlanabilir lojik kontrolörler hakkında genel bilgiye sahip mühendisler, programcılar, kurulumcular ve elektrikçiler için tasarlanmıştır.

## Gerekli temel bilgiler

Bu kılavuzu anlamak için otomasyon ve programlanabilir lojik kontrolörler hakkında genel bir bilgi birikimine sahip olmak gereklidir.

## Bu kılavuzun kapsamı

Bu kılavuz aşağıdaki ürünleri açıklar:

- STEP 7 V13 Basic and Professional (Sayfa 355)
- S7-1200 CPU firmware release V4.0

Bu kılavuzda açıklanan S7-1200 ürünlerinin tam bir listesi için teknik özelliklere (Sayfa 829) bakınız.

## Sertifikasyon, CE etiketi, C-Tick ve diğer standartlar

Daha fazla bilgi için teknik özelliklere (Sayfa 829) bakınız.

## Servis ve destek

Siemens, bu dokümana ilave olarak İnternet'te ve müşteri destek web sayfasında (<http://www.siemens.com/automation/>) teknik uzmanlık hizmeti sunar.

Eğitim için veya S7 ürünlerinin siparişi için teknik sorularınız cevaplanmasında yardımcı olmak üzere Siemens distribütörleri veya satış ofisleri ile temas kurunuz. Sizin satış temsilcileriniz teknik olarak eğitildikleri ve sizin operasyonlarınız, işlemleriniz, endüstri ve kullandığınız her bir Siemens ürünü hakkında en ayrıntılı bilgi birikimine sahip oldukları için onlar sizin karşılaşılabileceğiniz herhangi bir probleme en hızlı ve en etkin cevapları sağlayabileceklerdir.

## Dokümantasyon ve bilgiler

S7-1200 ve STEP 7 istediğiniz teknik bilgileri bulmanız için çeşitli dokümantasyonlar ile diğer kaynakları sağlar.

- S7-1200 sistem kılavuzu, komple S7-1200 ürün ailesi için çalışma, programlama ve özellikler hakkında özel bilgiler sağlar. Sistem kılavuzuna ilave olarak S7 Easy Book, S7-1200 ailesinin yetenekleri üzerine daha genel bir tanıtımı sunar.

Hem sistem kılavuzu hem de S7 Easy Book elektronik (PDF) kılavuzlar olarak mevcuttur. Elektronik kılavuzlar müşteri destek web sayfasından indirilebilirler ve ayrıca her S7-1200 CPU ile birlikte gönderilen dokümantasyon diskinde de bulunabilirler.

- Çevrim içi (online) STEP 7 bilgi sistemi, kavramsal bilgilere ve özel komutlara hemen erişimi sağlar. Özel komutlar, programlama paketinin operasyonları ile fonksiyonelliğini ve SIMATIC CPU'ların temel operasyonlarını tanımlar.

- My Documentation Manager, SIMATIC dokümantasyon setinin elektronik (PDF) sürümlerine erişimi sağlar. Bu set; sistem kılavuzu, Easy Book ve STEP 7 bilgi sistemini içerir. My Documentation Manager ile kendinize ait özel kılavuz oluşturmak için çeşitli dokümanlardan konuları sürükleyip bırakabilirsiniz.

Müşteri destek giriş portalı (<http://support.automation.siemens.com>), mySupport altındaki My Documentation Manager için bir bağlantı sağlar.

- Müşteri destek web sayfası ayrıca S7-1200 ve STEP 7 için podcasts, sık sorulan sorular (FAQs) ve diğer kullanışlı dokümanlar sağlar. Podcast'lar, STEP 7 tarafından sağlanan etkileşimler, uygunluk ve yeterliliği göstermek için özel özellikler ve senaryolar üzerine odaklanan eğitim amaçlı kısa video sunumlarını oynatır. Podcast'ların koleksiyonuna erişmek için aşağıdaki web sayfalarını ziyaret ediniz.
  - STEP 7 Basic web sayfası (<http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/step7-basic/Pages/Default.aspx>)
  - STEP 7 Professional web sayfası (<http://www.automation.siemens.com/mcms/simatic-controller-software/en/step7/step7-professional/Pages/Default.aspx>)
- Ayrıca Servis & Destek teknik formunda ürün tartışmalarını izleyebilir ve katılabilirsiniz (<https://www.automation.siemens.com/WW/forum/guests/Conferences.aspx?Language=en&siteid=csius&treeLang=en&groupid=4000002&extranet=standard&viewreg=WW&nodeid=34612486>). Bu forumlar size çeşitli ürün uzmanları ile etkileşime girmenizi sağlar.
  - S7-1200 için forum (<https://www.automation.siemens.com/WW/forum/guests/Conference.aspx?SortField=LastPostDate&SortOrder=Descending&ForumID=258&Language=en&onlyInternet=False>)
  - STEP 7 Basic için forum (<https://www.automation.siemens.com/WW/forum/guests/Conference.aspx?SortField=LastPostDate&SortOrder=Descending&ForumID=265&Language=en&onlyInternet=False>)

## Güvenlik bilgileri

Siemens, endüstriyel güvenlik fonksiyonları olan ürünler ve çözümler sağlar. Böylece tesislerin, çözümlerin, makinelerin ve/veya şebekelerin güvenli çalışmasını destekler. Bunlar bütünselci endüstriyel güvenlik kavramlarında önemli bileşenlerdir. Bunu akılda bulundurarak, Siemens'e ait ürünler ve çözümler sürekli gelişime uğramaktadır. Siemens size ürün güncellemelerini düzenli olarak kontrol etmenizi tavsiye eder.

Siemens ürünlerinin ve çözümlerinin güvenli olarak çalışması için uygun önleyici aksiyonları (örneğin hücre güvenlik kavramı) almak ve her bir bileşeni bütüne entegre etmek modern endüstriyel güvenlik kavramında gereklidir. Kullanımda olabilen üçüncü taraflara ait ürünler de dikkate alınmalıdır. Endüstriyel güvenlik hakkında İnternet'te daha fazla bilgi bulabilirsiniz (<http://www.siemens.com/industrialsecurity>).

Ürünler ortaya çıktığında ürün güncellemeleri hakkında bilgilendirilmek amacıyla ürün özel haber bültenine üye olabilirsiniz. İnternet'te daha fazla bilgi bulabilirsiniz. (<http://support.automation.siemens.com>).



# İçindekiler çizelgesi

Önsöz.....	3
<b>1 Ürüne genel bir bakış .....</b>	<b>23</b>
1.1 S7-1200 PLC tanıtımı .....	23
1.2 CPU genişleme kapasitesi .....	26
1.3 S7-1200 modülleri .....	29
1.4 Yeni özellikler .....	30
1.5 Temel HMI paneller.....	32
<b>2 STEP 7 programlama yazılımı.....</b>	<b>35</b>
2.1 Sistem gerekleri .....	36
2.2 İş daha kolay yapmak için farklı görüşler .....	37
2.3 Kolay kullanım araçları.....	38
2.3.1 Kullanıcı programınıza komutlar ekleme .....	38
2.3.2 "Favorites" araç çubuğundan komutlara erişim .....	38
2.3.3 Bir basit komut ile bir karmaşık eşitlik oluşturma .....	39
2.3.4 Bir LAD veya FBD komutuna girişler veya çıkışlar ekleme .....	41
2.3.5 Genişletilebilir komutlar .....	42
2.3.6 Bir komut için bir versiyon seçmek.....	42
2.3.7 STEP 7'nin görünüm ve konfigürasyonunu değiştirme .....	43
2.3.8 Editörler arasında sürükleme ve bırakma .....	43
2.3.9 CPU'nun çalışma modunu değiştirme.....	44
2.3.10 Bir DB için çağırılma tipinin değiştirilmesi .....	45
2.3.11 Cihazları bir ağdan geçici olarak ayırma.....	46
2.3.12 Cihazların konfigürasyondan sanal olarak çıkarılması .....	47
<b>3 Kurulum .....</b>	<b>49</b>
3.1 S7-1200 cihazlarının kurulumu için kılavuzlar .....	49
3.2 Güç ihtiyacı hesabı .....	51
3.3 Montaj ve sökme işlemleri.....	53
3.3.1 S7-1200 cihazları için montaj boyutları .....	53
3.3.2 CPU'nun montajı ve sökülmesi .....	56
3.3.3 Bir SB, CB veya BB'nin takılması ve çıkarılması .....	58
3.3.4 Bir SM'in takılması ve sökülmesi .....	60
3.3.5 CM ve CP'nin takılması ve sökülmesi.....	62
3.3.6 S7-1200 terminal blok konnektörün çıkarılması ve yeniden takılması.....	63
3.3.7 Genişletme kablosunun takılması ve çıkarılması.....	64
3.3.8 TS (TeleService) adaptör .....	66
3.3.8.1 TeleService Adapter'in bağlanması .....	66
3.3.8.2 SIM Kartının takılması.....	67
3.3.8.3 TS adaptör ünitesinin bir DIN rayı üzerine montajı .....	69
3.3.8.4 TS adaptörünün bir panel üzerine montajı.....	69

3.4	Kablaj kılavuzları .....	70
<b>4</b>	<b>PLC konseptleri .....</b>	<b>77</b>
4.1	Kullanıcı programının yürütülmesi .....	77
4.1.1	CPU'nun işletim modları.....	81
4.1.2	RUN modunda tarama döngüsünün işlenmesi .....	85
4.1.3	Organizasyon blokları (OB'ler) .....	85
4.1.3.1	Program döngüsü OB .....	86
4.1.3.2	Başlangıç OB (Startup OB).....	86
4.1.3.3	Zaman gecikmeli interrupt (kesme) OB .....	87
4.1.3.4	Döngüsel interrupt OB.....	87
4.1.3.5	Donanım interrupt OB .....	87
4.1.3.6	Zaman hatası interruptı OB.....	88
4.1.3.7	Diyagnostik (tanılama) hata interruptı OB.....	89
4.1.3.8	Modüller için çek veya tak OB.....	91
4.1.3.9	Şasi veya istasyon arızası OB .....	92
4.1.3.10	Gün içindeki saat OB .....	92
4.1.3.11	Durum OB .....	93
4.1.3.12	Güncelleme OB.....	93
4.1.3.13	Profil OB .....	94
4.1.3.14	Olay yürütme öncelikleri ve kuyruklama .....	94
4.1.4	Döngü süresinin konfigürasyonu ve izlenmesi.....	97
4.1.5	CPU belleği .....	99
4.1.5.1	Sistem ve darbe jeneratör belleği .....	101
4.1.6	Diyagnostik (tanılama) ara belleği.....	103
4.1.7	Günün saati.....	104
4.1.8	Bir RUN – STOP geçişindeki çıkışların konfigürasyonu .....	104
4.2	Veri depolama, bellek alanları, I/O ve adresleme .....	105
4.2.1	S7-1200'ün verilerine erişim.....	105
4.3	Analog değerlerin işlenmesi .....	110
4.4	Veri tipleri .....	110
4.4.1	Bool, Byte, Word ve DWord veri tipleri.....	111
4.4.2	Tamsayı (integer) sayı tipleri.....	112
4.4.3	Kayan noktalı gerçek veri tipleri .....	112
4.4.4	Saat ve tarih veri tipleri.....	113
4.4.5	Karakter ve String veri tipleri .....	114
4.4.6	Dizi Veri tipleri .....	116
4.4.7	Veri yapısı veri tipi.....	117
4.4.8	PLC veri tipi .....	117
4.4.9	Pointer (işaretçi) veri tipleri.....	118
4.4.9.1	"Pointer" pointer veri tipi.....	118
4.4.9.2	"Any" pointer veri tipi .....	119
4.4.9.3	"Variant" pointer veri tipi.....	120
4.4.10	Etiketli bir veri tipinin bir dilimine ("slice") erişim .....	121
4.4.11	Bir AT bindirmeli (overlay) bir etikete erişim .....	122
4.5	Bellek kartının kullanımı .....	124
4.5.1	CPU'ya bir bellek kartı takmak.....	125
4.5.2	Projeyi hafıza kartına kopyalamadan önce CPU'nun başlangıç parametresini konfigüre etmek.....	127



4.5.3	Transfer kartı .....	127
4.5.4	Program kartı .....	130
4.5.5	Donanım yazılımı güncelleme.....	133
4.6	Kayıp bir şifrenin kurtarılması .....	136
<b>5</b>	<b>Cihaz Konfigürasyonu .....</b>	<b>137</b>
5.1	Bir CPU eklemek.....	138
5.2	Belirtilmemiş bir CPU için konfigürasyon tespiti.....	140
5.3	Konfigürasyona modüller eklemek.....	141
5.4	Bir cihazın değiştirilmesi .....	142
5.5	CPU'nun çalışmasını konfigüre etmek.....	142
5.5.1	Genel bakış .....	142
5.5.2	Dijital giriş filtrelerinin konfigürasyonu.....	144
5.5.3	Pals yakalama .....	146
5.6	Modüllerin parametrelerini konfigüre etmek.....	147
5.7	Haberleşme için CPU'yu konfigüre etmek .....	149
5.7.1	Bir ağ bağlantısı oluşturmak .....	149
5.7.2	Lokal/Partner bağlantı yolunun konfigürasyonu.....	150
5.7.3	PROFINET bağlantısı için parametreler .....	153
5.7.4	İnternet Protokol (IP) adreslerinin atanması .....	155
5.7.4.1	Programlama ve ağ cihazları için IP adreslerinin atanması.....	155
5.7.4.2	Programlama cihazınızın IP adresini kontrol etme .....	157
5.7.4.3	CPU'ya online olarak IP Adresi Atama .....	158
5.7.4.4	Projenizdeki bir CPU için bir IP adresinin konfigürasyonu.....	159
5.7.5	PROFINET ağının test edilmesi.....	162
5.7.6	CPU'daki Ethernet (MAC) adresini konumlandırmak.....	163
5.7.7	Ağ Zaman Protokolü senkronizasyonunun konfigüre edilmesi.....	165
5.7.8	PROFINET cihazı başlangıç süresi, isimlendirme ve adres ataması .....	166
<b>6</b>	<b>Programlama konseptleri .....</b>	<b>169</b>
6.1	PLC sisteminin tasarımı için kılavuzlar .....	169
6.2	Kullanıcı programınızın yapılandırılması .....	170
6.3	Programınızı yapılandırmak için blokları kullanmak .....	172
6.3.1	Organizasyon bloğu (OB) .....	172
6.3.2	Fonksiyon (FC).....	174
6.3.3	Fonksiyon bloğu (FB).....	175
6.3.4	Veri bloğu (DB).....	176
6.3.5	Tekrar kullanılabilir kod blokları oluşturmak.....	178
6.4	Veri tutarlılığının (uyumluluğunun) anlaşılması.....	178
6.5	Programlama dili .....	179
6.5.1	Merdiven mantığı (LAD).....	180
6.5.2	Fonksiyon bloğu şeması (FBD).....	181
6.5.3	SCL .....	181
6.5.4	LAD, FBD ve SCL için EN ve ENO .....	188
6.6	Koruma.....	190
6.6.1	CPU için erişim koruması.....	190

6.6.2	Know-how koruması.....	193
6.6.3	Kopya koruması .....	195
6.7	Programınızın elemanlarını yüklemek .....	196
6.8	CPU'dan yükleme .....	197
6.8.1	Projenin elemanlarının kopyalanması .....	197
6.8.2	Karşılaştırma fonksiyonunu kullanmak .....	198
6.9	Programda hata ayıklama ve test işlemleri .....	198
6.9.1	CPU'daki verilerin izlenmesi ve değiştirilmesi .....	198
6.9.2	İzleme tabloları ve zorlama tabloları .....	199
6.9.3	Kullanımı göstermek için çapraz başvuru .....	199
6.9.4	Çağırma hiyerarşisini incelemek için çağrı yapısı .....	201
<b>7</b>	<b>Temel komutlar .....</b>	<b>203</b>
7.1	Bit lojik çalışmaları .....	203
7.1.1	Bit lojik komutları .....	203
7.1.2	Set ve reset komutları .....	206
7.1.3	Pozitif Geçiş (Yükselen Kenar) ve Negatif Geçiş (Düşen Kenar) Komutları.....	209
7.2	Zamanlayıcı işlemleri .....	212
7.3	Sayıcı İşlemleri.....	220
7.4	Karşılaştırmacı işlemleri .....	226
7.4.1	Karşılaştırma değer komutları .....	226
7.4.2	IN_Range (Aralık içi) ve OUT_Range (aralık dışı) komutlar .....	227
7.4.3	OK (Tamam) ve NOT OK (Tamam değil) komutları.....	228
7.5	Matematik fonksiyonları .....	229
7.5.1	CALCULATE (Hesapla) komutu .....	229
7.5.2	Toplama, çıkarma, çarpma ve bölme komutları.....	230
7.5.3	MOD (bölmenin kalanına dön) komutu .....	231
7.5.4	NEG (ikinin tamamlayıcısını oluştur) komutu.....	232
7.5.5	INC (Arttır) ve DEC (Azalt) komutları .....	232
7.5.6	ABS (Mutlak değer oluştur) komutu .....	233
7.5.7	MIN (Minimum) ve MAX (Maksimum) komutları .....	234
7.5.8	LIMIT (Limit değeri ver) komutu .....	235
7.5.9	Üslü, logaritma ve trigonometri komutları .....	236
7.6	Taşı komutları .....	238
7.6.1	MOVE (değeri taşı), MOVE_BLK (blok taşı), ve UMOVE_BLK (bloğu kesintisiz taşı) komutları.....	238
7.6.2	FieldRead (alanı oku) ve FieldWrite (alana yaz) komutları.....	240
7.6.3	FILL_BLK (bloğu doldur) ve UFILL_BLK (bloğu kesintisiz doldur) talimatları .....	242
7.6.4	SWAP (bitleri deęiş tokuş et) komutu .....	243
7.7	Dönüştürme işlemleri .....	244
7.7.1	CONV (değeri dönüştür) komutu .....	244
7.7.2	SCL için dönüştürme komutları .....	245
7.7.3	ROUND (sayısal değeri yuvarla) and TRUNC (sayısal değeri kes) komutları.....	249
7.7.4	CEIL ve FLOOR komutları (Generate next higher and lower integer from floating-point number) instructions.....	250
7.7.5	SCALE_X (Ölçek) and NORM_X (Normalize) komutları.....	251
7.8	Program kontrol işlemleri .....	254

7.8.1	JMP (RLO = 1 ise atlat), JMPN (RLO = 0 ise atlat) ve Label (Jump etiketi) komutları .....	254
7.8.2	JMP_LIST (atlama listesini tanımla) komutu .....	255
7.8.3	SWITCH (atlama dağıtıcısı) komutu .....	256
7.8.4	RET (Geri dönüş) talimatı .....	258
7.8.5	ENDIS_PW (CPU şifrelerini etkinleştirme/ Enable/geçersiz kılma) talimatı .....	259
7.8.6	RE_TRIGR (yeniden başlatma (Restart) çevrimi izleme zamanı) talimatı .....	262
7.8.7	STP (Çıkış programı) talimatı .....	263
7.8.8	GET_ERROR ve GET_ERROR_ID (yerli olarak hata ve hata ID al) komutları .....	263
7.8.9	SCL program kontrol komutları .....	267
7.8.9.1	SCL program kontrol komutlarına genel bakış .....	267
7.8.9.2	IF-THEN komutu .....	268
7.8.9.3	CASE komutu .....	269
7.8.9.4	FOR komutu .....	270
7.8.9.5	WHILE-DO komutu .....	271
7.8.9.6	REPEAT-UNTIL komutu .....	272
7.8.9.7	CONTINUE komutu .....	273
7.8.9.8	EXIT komutu .....	273
7.8.9.9	GOTO komutu .....	274
7.8.9.10	RETURN komutu .....	274
7.9	Word lojik işlemleri .....	275
7.9.1	AND, OR ve XOR lojik işlem komutları .....	275
7.9.2	INV (birin tamamlayıcısını oluştur) komutu .....	276
7.9.3	DECO (Kodla) ENCO (Kod çöz) komutları .....	276
7.9.4	SEL (Seç), MUX (Çokla), and DEMUX (Çoklama çöz) komutları .....	278
7.10	Kaydır ve Döndür .....	281
7.10.1	SHR (Sağa kaydır) andve SHL (Sola kaydır) komutları .....	281
7.10.2	ROR (Sağa döndür) ve ROL (Sola döndür) komutları .....	282
<b>8</b>	<b>Genişletilmiş komutlar .....</b>	<b>283</b>
8.1	Tarih, gün zamanı ve saat fonksiyonları .....	283
8.1.1	Tarih ve gün zaman komutları .....	283
8.1.2	Saat fonksiyonları .....	286
8.1.3	TimeTransformationRule veri yapısı .....	289
8.1.4	SET_TIMEZONE (Zaman dilimini ayarla) komutu .....	290
8.1.5	RTM (Çalışma zamanı ölçer) komutu .....	291
8.2	String ve karakter .....	292
8.2.1	String (karakter dizisi) veriye genel bakış .....	292
8.2.2	S_MOVE (Karakter string'i kaydır) komutu .....	293
8.2.3	String dönüşüm komutları .....	293
8.2.3.1	S_CONV, STRG_VAL ve VAL_STRG (Karakter string'e ve sayıya veya bunlardan dönüşürme) komutları .....	293
8.2.3.2	Strg_TO_Chars ve Chars_TO_Strg (karakter string'e ve CHAR dizisine veya bunlardan dönüşüm) komutları .....	302
8.2.3.3	ATH ve HTA (ASCII string ve onaltılık sayıya veya bunlardan dönüşüm) komutu .....	304
8.2.4	String işlem komutları .....	306
8.2.4.1	MAX_LEN (Bir karakter string'in maksimum uzunluğu) komutu .....	306
8.2.4.2	LEN (Bir karakter string'in uzunluğunu belirleme) komutu .....	307
8.2.4.3	CONCAT ( karakter string'lerini birleştirme) komutu .....	307
8.2.4.4	LEFT, RIGHT ve MID (bir karakter string'deki substringleri okuma) komutları .....	308
8.2.4.5	DELETE (bir karakter string'deki karakterleri silme) komutu .....	310
8.2.4.6	INSERT (karakterleri bir karakter string'ine yerleştirme) komutu .....	311

8.2.4.7	REPLACE (bir karakter string'indeki karakterleri değiştirme) komutu .....	312
8.2.4.8	FIND (karakter string içindeki karakterleri bulma) komutu .....	313
8.3	Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-i) .....	314
8.3.1	Dağıtılmış I/O Komutları .....	314
8.3.2	RDREC ve WRREC (Veri kaydını okuma/yazma) komutları .....	315
8.3.3	RALRM (Kesme alma) komutu .....	318
8.3.4	RDREC, WRREC ve RALRM için STATUS parametresi .....	322
8.3.5	DPRD_DAT ve DPWR_DAT (DP slave'leri için tutarlı veri okuma/yazma) komutları .....	326
8.3.6	DPNRM_DG (Bir DP slave'den gelen tanılama (diyagnostik) verilerini okuma) komutu .....	328
8.4	Kesmeler (Interrupts) .....	331
8.4.1	ATTACH ve DETACH (OB ve kesme olayını bağlama/ayırma) komutları .....	331
8.4.2	Döngüsel kesmeler .....	334
8.4.2.1	SET_CINT (Döngüsel kesme parametrelerini ayarlama) komutu .....	334
8.4.2.2	QRY_CINT (Döngüsel kesme parametrelerini sorgulama) komutu .....	336
8.4.3	Günün saati kesmeleri .....	337
8.4.3.1	SET_TINTL (günün saati kesmesi ayarlama) .....	337
8.4.3.2	CAN_TINT (günün saati kesmesini iptal etme) .....	339
8.4.3.3	ACT_TINT (günün saati kesmesini etkinleştirme) .....	340
8.4.3.4	QRY_TINT (günün saati kesme durumunu sorgulama) .....	340
8.4.4	Time delay (Zaman gecikme) kesmeleri .....	342
8.4.5	DIS_AIRT ve EN_AIRT (Yüksek öncelikli kesmelerin yürütülmesini geciktirme/ etkinleştirme ve asenkron hata olayları) komutları .....	344
8.5	Tanılama (Diyagnostik) (PROFINET veya PROFIBUS) .....	345
8.5.1	Tanılama (diyagnostik) komutları .....	345
8.5.2	Dağıtılmış I/O için tanılama (diyagnostik) olayları .....	345
8.5.3	LED (LED durumu okuma) komutu .....	346
8.5.4	DeviceStates komutu .....	347
8.5.4.1	DeviceStates örnek yapılandırması .....	349
8.5.5	ModuleStates komutu .....	353
8.5.5.1	ModuleStates örnek konfigürasyonları .....	354
8.5.6	GET_DIAG (Tanılama bilgisini oku) komutu .....	358
8.6	Darbe (pals) .....	364
8.6.1	CTRL_PWM (Darbe genişlik modülasyonu) komutu .....	364
8.6.2	Darbe çıkışları işlemi .....	365
8.6.3	PWM için bir darbe kanalı yapılandırması .....	367
8.7	Tarifler ve veri günlükleri .....	369
8.7.1	Tarifler .....	369
8.7.1.1	Tarife genel bakış .....	369
8.7.1.2	DB tarifi örneği .....	370
8.7.1.3	Tarif verisini aktaran program komutları .....	374
8.7.1.4	Örnek tarif programı .....	378
8.7.2	Veri günlükleri .....	380
8.7.2.1	Veri log kayıt yapısı .....	381
8.7.2.2	Veri günlüklerini (log'larını) kontrol eden program komutları .....	382
8.7.2.3	Veri günlükleriyle çalışma .....	392
8.7.2.4	Veri günlük dosyalarının boyutunu sınırlama .....	393
8.7.2.5	Veri günlüğü örnek program .....	396
8.8	Veri blok kontrolü .....	401

8.8.1	READ_DBL ve WRIT_DBL (Yükleme hafızasındaki bir veri bloğunu okuma/yazma) komutları .....	401
8.9	Adres işleme .....	404
8.9.1	LOG2GEO (slotu donanım kimliğinden belirleme) komutu.....	404
8.9.2	RD_ADDR (Donanım belirtecindeki IO adreslerini belirleme) komutu.....	405
8.10	"Genişletilmiş" komutlar için ortak hata kodları .....	406
<b>9</b>	<b>Teknoloji komutları .....</b>	<b>409</b>
9.1	Yüksek hızlı sayıcı .....	409
9.1.1	Yüksek hızlı sayıcının çalışması .....	411
9.1.2	HSC yapılandırması .....	417
9.2	PID kontrol .....	418
9.2.1	PID komutu ve teknoloji nesnesinin yerleştirilmesi .....	420
9.2.2	PID_Compact komutu .....	422
9.2.3	PID_Compact komutu ErrorBit parametreleri .....	426
9.2.4	PID_3Step komutu .....	428
9.2.5	PID_3Step komutu ErrorBit parametreleri .....	435
9.2.6	PID kontrolörün yapılandırılması.....	437
9.2.7	PID kontrolörün devreye alınması.....	439
9.3	Hareket kontrolü .....	441
9.3.1	Safhalama .....	446
9.3.2	Darbe jeneratörünün yapılandırılması .....	448
9.3.3	Eksenin yapılandırılması .....	449
9.3.4	TO_CommandTable_PTO yapılandırması .....	452
9.3.5	Hareket kontrol komutları .....	455
9.3.5.1	MC komutuna genel bakış .....	455
9.3.5.2	MC_Power (Ekseni serbest bırakma/engelleme) komutu.....	456
9.3.5.3	MC_Reset (Hatayı onayla) komutu .....	459
9.3.5.4	MC_Home (Home ekseni) komutu .....	460
9.3.5.5	MC_Halt komutu.....	462
9.3.5.6	MC_MoveAbsolute (Ekseni mutlak konumla) komutu .....	464
9.3.5.7	MC_MoveRelative (Ekseni bağıl olarak konumla) komutu .....	466
9.3.5.8	MC_MoveVelocity (Ekseni önceden belirlenen hızda hareket ettir) komutu .....	468
9.3.5.9	MC_MoveJog (Jog modunda ekseni hareket ettir) komutu .....	471
9.3.5.10	MC_CommandTable (Eksen komutlarını hareket dizisi olarak çalıştırma) komutu .....	473
9.3.5.11	MC_ChangeDynamic (Eksenin dinamik ayarlarını değiştir) komutu.....	476
9.3.5.12	MC_WriteParam (teknolojisi nesnesi parametrelerini yaz) komutu .....	478
9.3.5.13	MC_ReadParam (teknoloji nesnesi parametrelerini oku) komutu .....	480
9.3.6	S7-1200 için hareket kontrolünün çalışması .....	481
9.3.6.1	Hareket kontrolü için kullanılan CPU çıkışları .....	481
9.3.6.2	Hareket kontrolü için donanım ve yazılım sınır anahtarları .....	483
9.3.6.3	Homing .....	486
9.3.6.4	Sarsıntı sınırı .....	491
9.3.7	Devreye alma .....	492
9.3.8	Aktif komutların izlenmesi .....	497
9.3.8.1	MC komutların "Done" çıkış parametresiyle izlenmesi .....	497
9.3.8.2	MC_Velocity komutunun izlenmesi .....	501
9.3.8.3	MC_MoveJog komutunun izlenmesi .....	505
<b>10</b>	<b>Haberleşme .....</b>	<b>509</b>

10.1	Desteklenen asenkron haberleşme bağlantılarının sayısı .....	511
10.2	PROFINET .....	512
10.2.1	Yerel/Partner bağlantı .....	512
10.2.2	Acık kullanıcı haberleşmesi .....	514
10.2.2.1	Açık kullanıcı haberleşme komutları için bağlantı ID'leri .....	514
10.2.2.2	Protokoller .....	517
10.2.2.3	Ad hoc modu .....	518
10.2.2.4	TCP ve ISO on TCP .....	519
10.2.2.5	TSEND_C ve TRCV_C (Ethernet üzerinden veri gönderme ve alma) komutları .....	520
10.2.2.6	TCON, TDISCON, TSEND ve TRCV (TCP haberleşme) komutları .....	526
10.2.2.7	UDP .....	535
10.2.2.8	TUSEND ve TURCV .....	535
10.2.2.9	T_CONFIG .....	540
10.2.2.10	Komutlar için ortak parametreler .....	548
10.2.3	Programlama cihazıyla haberleşme .....	550
10.2.3.1	Donanım haberleşme bağlantısının kurulması .....	550
10.2.3.2	Cihazların yapılandırılması .....	551
10.2.3.3	İnternet Protokol (IP) adreslerinin atanması .....	551
10.2.3.4	PROFINET ağınızın test edilmesi .....	551
10.2.4	HMI-PLC haberleşmesi .....	552
10.2.4.1	İki cihaz arasındaki mantıksal ağ bağlantılarının yapılandırılması .....	553
10.2.5	PLC-PLC haberleşmesi .....	553
10.2.5.1	İki cihaz arasındaki mantıksal ağ bağlantılarının yapılandırılması .....	554
10.2.5.2	İki cihaz arasındaki Yerel / Partner bağlantı yolunu yapılandırma .....	555
10.2.5.3	İletim (gönderme) ve alma parametrelerini yapılandırma .....	555
10.2.6	Bir CPU ve PROFINET IO cihazını yapılandırma .....	558
10.2.6.1	Bir PROFINET IO cihazı ekleme .....	558
10.2.6.2	CPU ve PROFINET IO cihazı arasındaki mantıksal ağ bağlantılarını yapılandırma .....	558
10.2.6.3	CPU'lar ve cihaz isimlerini atama .....	559
10.2.6.4	İnternet Protokol (IP) adresleri atama .....	559
10.2.6.5	IO çevrim süresinin yapılandırılması .....	560
10.2.7	Bir CPU ve PROFINET I-Cihazın yapılandırılması .....	561
10.2.7.1	I-cihaz işlevselliği .....	561
10.2.7.2	I-cihazın özellikleri ve avantajları .....	562
10.2.7.3	Bir I-cihazın karakteristiği .....	562
10.2.7.4	Daha yüksek ve daha düşük seviyeli IO sistemi arasında veri değişimi .....	565
10.2.7.5	I-cihaz yapılandırması .....	567
10.2.8	Tanılama (diyagnostik) .....	568
10.2.9	Dağıtılmış I/O komutları .....	568
10.2.10	Tanılama (diyagnostik) komutları .....	569
10.2.11	Dağıtılmış I/O için tanılama olayları .....	569
10.3	PROFIBUS .....	569
10.3.1	PROFIBUS CM'lerin haberleşme hizmetleri .....	571
10.3.2	PROFIBUS CM kullanıcı kılavuzlarına referans .....	572
10.3.3	DP master ve slave cihaz yapılandırma .....	572
10.3.3.1	CM 1243-5 (DP master) modülü ve bir DP slave ekleme .....	572
10.3.3.2	İki PROFIBUS cihazı arasındaki mantıksal ağ bağlantılarının yapılandırılması .....	573
10.3.3.3	CM 1243-5 modülü ve DP slave'e PROFIBUS adreslerini atama .....	574
10.3.4	Dağıtılmış I/O komutları .....	575
10.3.5	Tanılama (diyagnostik) komutları .....	575
10.3.6	Dağıtılmış I/O için tanılama olayları .....	576

10.4	AS-i .....	576
10.4.1	Bir AS-i master ve slave cihaz yapılandırma .....	577
10.4.1.1	AS-i master CM 1243-2 ve AS-i slave ekleme.....	577
10.4.1.2	İki AS-i cihazı arasında bağlantı için lojik devrenin konfigürasyonu .....	578
10.4.1.3	AS-i master CM1243-2 özelliklerinin yapılandırılması .....	578
10.4.1.4	Bir AS-i adresinin bir AS-i slave'e atanması .....	579
10.4.2	Kullanıcı programı ve AS-i slave'ler arasında veri alışverişi .....	581
10.4.2.1	STEP 7 temel yapılandırma .....	581
10.4.2.2	STEP 7 ile slave'leri yapılandırma .....	583
10.4.3	Dağıtılmış I / O komutları .....	585
10.4.4	AS-i çevrimiçi araçlarla çalışma .....	585
10.5	S7 haberleşme .....	587
10.5.1	GET ve PUT (uzak CPU' dan oku ve yaz) komutları .....	587
10.5.2	S7 bağlantısı oluşturma .....	591
10.5.3	İki cihaz arasındaki Yerel / Partner bağlantı yolunu yapılandırma.....	592
10.5.4	GET / PUT bağlantı parametre atama .....	592
10.5.4.1	Bağlantı parametreleri.....	593
10.5.4.2	CPU-CPU S7 bağlantı yapılandırma.....	595
<b>11</b>	<b>Web sunucu.....</b>	<b>601</b>
11.1	Web sunucusunu etkinleştirme .....	603
11.2	Web sunucusu kullanıcılarını yapılandırma .....	604
11.3	PC'den Web sayfalarına erişim.....	606
11.4	Bir mobil cihazdan Web sayfalarına erişme.....	607
11.5	Standart Web sayfaları .....	609
11.5.1	Standart Web sayfalarının yerleşim düzeni .....	609
11.5.2	Oturum açma ve kullanıcı ayrıcalıkları.....	610
11.5.3	Giriş.....	613
11.5.4	Başlangıç .....	614
11.5.5	Tanıtım (Kimlik) .....	614
11.5.6	Tanımlama arabelleği .....	616
11.5.7	Modül bilgisi .....	616
11.5.8	İletişim .....	620
11.5.9	Değişken Durumu .....	620
11.5.10	Dosya tarayıcısı .....	622
11.6	Kullanıcı tanımlı Web sayfaları .....	625
11.6.1	HTML sayfası oluşturma .....	626
11.6.2	S7-1200 Web sunucusu tarafından desteklenen AWP komutları.....	627
11.6.2.1	Değişkenleri okuma .....	629
11.6.2.2	Değişkenleri Yazma .....	629
11.6.2.3	Özel değişkenler okuma .....	631
11.6.2.4	Özel değişkenler yazma.....	632
11.6.2.5	Değişken bir referans için bir diğer ad (takma ad) kullanma .....	634
11.6.2.6	Enum tipleri tanımlama .....	634
11.6.2.7	Bir enum tipli CPU değişkenlerine başvurma .....	635
11.6.2.8	Fragmanları oluşturma.....	637
11.6.2.9	Fragmanları içeri aktarma (import).....	638
11.6.2.10	Tanımları birleştirme .....	638
11.6.2.11	Özel karakterler içeren etiket adlarını işleme.....	639

11.6.3	Kullanıcı-tanımlı Web sayfalarının kullanımını yapılandırma .....	641
11.6.4	Kullanıcı-tanımlı web sayfaları için WWW komutu programlama .....	642
11.6.5	CPU' ya program bloklarını yükleme .....	643
11.6.6	Kullanıcı-tanımlı Web sayfalarına erişim.....	644
11.6.7	Kullanıcı-tanımlı Web sayfalarına özgü kısıtlamalar .....	644
11.6.8	Kullanıcı tanımlı bir web sayfası örneği .....	645
11.6.8.1	Bir rüzgar türbininin izlenmesi ve kontrol edilmesi için Web sayfası .....	645
11.6.8.2	Kontrolör verisi okuma ve görüntüleme .....	647
11.6.8.3	Bir enum tipini kullanma .....	648
11.6.8.4	Kontrolöre kullanıcı girişini yazma .....	649
11.6.8.5	Özel bir değişken yazma .....	650
11.6.8.6	Referans: Uzak rüzgar türbini izleme Web sayfasının HTML listesi .....	650
11.6.8.7	Örnek Web sayfasının STEP 7'de yapılandırması .....	654
11.6.9	Kullanıcı-tanımlı Web sayfalarını çoklu dilde ayarlama .....	656
11.6.9.1	Klasör yapısını oluşturma.....	656
11.6.9.2	Dil geçişini programlama .....	657
11.6.9.3	Çoklu dil sayfa yapısını kullanmak için STEP 7 yapılandırma .....	659
11.6.10	Gelişmiş kullanıcı-tanımlı Web sayfası kontrolü .....	659
11.7	Kısıtlamalar .....	663
11.7.1	İnternet seçenekleri JavaScript'i devre dışı bıraktığında özellik kısıtlamaları.....	664
11.7.2	İnternet seçenekleri çerezlere izin vermediğinde özellik kısıtlamaları .....	665
11.7.3	Siemens güvenlik sertifikasını içeri aktarma .....	665
11.7.4	Microsoft Excel'in ABD / İngiltere olmayan sürümleri için CSV formatında veri günlüklerini aktarma .....	667
<b>12</b>	<b>Haberleşme işlemcisi ve Modbus TCP .....</b>	<b>669</b>
12.1	Seri haberleşme ara yüzleri kullanarak.....	669
12.2	RS485 ağ konektörü önerilimleme ve sonlandırma .....	670
12.3	Noktadan noktaya (PtP) haberleşme .....	671
12.3.1	Haberleşme portları yapılandırma.....	672
12.3.1.1	Akış kontrolü yönetme.....	674
12.3.2	İletim (gönderme) ve alma parametrelerini yapılandırma .....	675
12.3.2.1	İletim (gönderme) parametrelerini yapılandırma .....	675
12.3.2.2	Alma parametrelerini yapılandırma .....	676
12.3.3	Noktadan noktaya komutlar .....	684
12.3.3.1	Noktadan noktaya komutlar için ortak parametreler .....	684
12.3.3.2	PORT_CFG (haberleşme parametrelerini dinamik olarak yapılandırınız) komutu .....	686
12.3.3.3	SEND_CFG (Seri iletim parametrelerini dinamik olarak yapılandırınız) komutu .....	688
12.3.3.4	RCV_CFG (Seri alma parametrelerini dinamik olarak yapılandırınız) komutu .....	689
12.3.3.5	SEND_PTP (Gönderilen ara bellek verisini ilet) komutu .....	694
12.3.3.6	RCV_PTP (Alma mesajlarını etkinleştirme) komutu .....	696
12.3.3.7	RCV_RST (Alıcı arabelleği sil) komutu .....	698
12.3.3.8	SGN_GET (RS-232 sinyallerini sorgulama) komutu.....	699
12.3.3.9	SGN_SET (RS-232 sinyallerini ayarlama) komutu .....	700
12.3.4	PtP haberleşmelerini Programlama .....	702
12.3.4.1	Yoklama mimarisi.....	702
12.3.5	Örnek: Noktadan Noktaya haberleşme.....	704
12.3.5.1	Haberleşme modülünün yapılandırılması .....	705
12.3.5.2	RS422 ve RS485 Çalışma modları .....	707
12.3.5.3	STEP 7 programı programlama .....	710
12.3.5.4	Terminal emulatrörünü yapılandırma .....	711



12.3.5.5	Örnek programı çalıştırma .....	712
12.4	Evrensel seri ara yüz (USS) haberleşmesi .....	712
12.4.1	USS protokolünü kullanma gereksinimleri .....	713
12.4.2	USS_PORT (USS devre üzerinden haberleşmeyi düzenleme) komutu .....	715
12.4.3	USS_DRV (Sürücü ile verileri değiştirme) komutu.....	716
12.4.4	USS_RPM (Sürücüden gelen parametreleri okuma) komutu .....	719
12.4.5	USS_WPM (Sürücüdeki parametreleri değiştirme) komutu.....	720
12.4.6	USS durum kodları.....	722
12.4.7	Genel sürücü kurulum bilgisi.....	724
12.5	Modbus haberleşme .....	727
12.5.1	Modbus RTU ve TCP haberleşme Modbus TCP komutları V13'e Genel Bakış.....	727
12.5.2	Modbus TCP .....	730
12.5.2.1	MB_CLIENT (PROFINET aracılığıyla Modbus TCP istemci olarak haberleşme) komutu.....	730
12.5.2.2	MB_SERVER (Modbus TCP sunucu olarak PROFINET aracılığıyla haberleşme) komutu .....	736
12.5.2.3	MB_SERVER örnek: Çoklu TCP bağlantıları.....	742
12.5.2.4	MB_CLIENT örnek 1: Ortak TCP bağlantısı ile çoklu istek.....	742
12.5.2.5	MB_CLIENT örnek 2: farklı TCP bağlantıları ile çoklu istek .....	743
12.5.2.6	MB_CLIENT örnek 3: Çıkış görüntü yazma isteği .....	744
12.5.2.7	MB_CLIENT örnek 4: Çoklu isteklerin koordinasyonu .....	745
12.5.3	Modbus RTU.....	746
12.5.3.1	MB_COMM_LOAD (Modbus RTU için PtP modülündeki portu yapılandırma) komutu .....	747
12.5.3.2	MB_MASTER (Modbus master olarak PtP portu üzerinden haberleşme) komutu.....	749
12.5.3.3	MB_SLAVE (Modbus slave olarak PtP portu üzerinden haberleşme) komutu .....	755
12.5.3.4	Modbus RTU master örnek program .....	762
12.5.3.5	Modbus RTU slave örnek program .....	764
12.6	CP 1242-7 ile telekontrol ve teleservis .....	765
12.6.1	GSM şebekesine bağlantı .....	765
12.6.2	CP 1242-7 uygulamaları .....	767
12.6.3	CP'nin diğer özellikleri.....	768
12.6.4	Yardımcı donanımlar.....	769
12.6.5	Telekontrol için yapılandırma örnekleri .....	770
<b>13</b>	<b>Teleservis haberleşme (SMTP email).....</b>	<b>775</b>
13.1	TM_Mail (email gönderme) komutu .....	775
<b>14</b>	<b>Çevrimiçi ve tanılama araçları .....</b>	<b>783</b>
14.1	Durum LED'leri.....	783
14.2	Çevrimiçi olma ve bir CPU bağlantısı .....	786
14.3	PROFINET IO cihazınıza çevrimiçi bir isim atama .....	787
14.4	IP adresini ve günün saatini ayarlama .....	789
14.5	Fabrika ayarlarına sıfırlama .....	789
14.6	Firmware güncellemesi .....	791
14.7	Çevrimiçi CPU için CPU operatör paneli .....	792
14.8	Çevrim süresi ve bellek kullanımını izleme .....	792
14.9	CPU'daki tanılama olaylarını görüntüleme .....	793
14.10	Çevrimdışı ve çevrimiçi CPU'ların karşılaştırılması .....	794

14.11	CPU değerlerini izleme ve değiştirme .....	795
14.11.1	CPU'daki değerleri izlemek için çevrimiçi olma.....	796
14.11.2	Program editöründe durumu gösterme .....	797
14.11.3	Başlangıç değerlerini sıfırlamak için bir DB çevrimiçi değerlerini yakalama .....	797
14.11.4	CPU değerlerini izlemek ve değiştirmek için bir izleme çizelgesi kullanma .....	798
14.11.4.1	PLC etiketlerini izlerken veya değiştirirken bir tetikleyici kullanma .....	799
14.11.4.2	STOP modunda çıkışları etkinleştirme .....	800
14.11.5	CPU'da zorlama değerleri .....	801
14.11.5.1	Cebir çizelgesini kullanarak.....	801
14.11.5.2	Cebir fonksiyonunun çalışması .....	802
14.12	RUN modunda yükleme .....	803
14.12.1	"RUN modunda yükle" için Önkoşullar .....	804
14.12.2	RUN modunda programınız değiştirme .....	805
14.12.3	Seçilen blokları yükleme .....	806
14.12.4	Başka bir bloktaki bir derleme hatası ile seçilen tek bir bloğu yükleme .....	807
14.12.5	RUN modunda mevcut blokları değiştirme ve yükleme .....	808
14.12.6	Yükleme işleminin başarısız olması halinde sistem tepkisi .....	811
14.12.7	RUN modunda yüklerken dikkat edilecek hususlar .....	811
14.13	Tetikleme koşullarında CPU verilerinin izlenmesi ve kaydedilmesi .....	813
<b>15</b>	<b>SM 1278 4xIO-Link Master.....</b>	<b>815</b>
15.1	SM 1278 4xIO-Link Master genel bakış.....	815
15.1.1	Genel bakış .....	815
15.1.2	IO-Link ve STEP 7 programınız .....	815
15.1.3	Özellikler.....	816
15.1.4	Fonksiyonlar.....	817
15.1.5	SM 4xIO-Link sinyal modülünün değiştirilmesi .....	817
15.1.6	Modülü fabrika ayarlarına reset .....	818
15.2	Bağlantı .....	818
15.2.1	Pin ataması .....	818
15.2.2	Blok diyagramı .....	820
15.3	Parametreler/adres alanı.....	821
15.3.1	Yapılandırma .....	821
15.3.2	Parametreler .....	821
15.3.3	Adres alanı .....	822
15.3.4	Parametre veri kaydı .....	822
15.4	Kesme, hata ve sistem alarmları.....	824
15.4.1	Durum ve hata göstergesi .....	824
15.4.2	Tanımlama alarmları .....	826
<b>A</b>	<b>Teknik Özellikler .....</b>	<b>829</b>
A.1	Genel teknik özellikler .....	829
A.2	CPU 1211C .....	838
A.2.4.1	CPU yerleşik analog girişleri için basamak tepkisi.....	843
A.2.4.2	CPU yerleşik analog portları için örnekleme süresi .....	844
A.2.4.3	Gerim (CPU) için yerleşik analog girişlerinin ölçme aralıkları .....	844
A.3	CPU 1212C .....	848
A.3.4.1	CPU yerleşik analog girişlerinin basamak tepkisi .....	853
A.3.4.2	CPU'nun yerleşik analog portları için örnekleme süresi .....	854

A.3.4.3	Gerilim (CPU'lar) için analog giriş gösterilişi.....	854
A.4	CPU 1214C .....	858
A.4.4.1	CPU yerleşik analog girişleri için basamak tepkisi.....	863
A.4.4.2	CPU yerleşik analog girişleri için örnekleme süresi .....	864
A.4.4.3	Gerilim (CPU'lar) için analog girişlerin ölçme aralıkları.....	864
A.5	CPU 1215C .....	868
A.5.4.1	CPU yerleşik analog girişlerinin basamak tepkisi .....	874
A.5.4.2	CPU yerleşik analog portlarının örnekleme süresi.....	874
A.5.4.3	Gerilime göre analog girişlerin ölçme aralıkları (CPU'lar).....	874
A.5.4.4	Analog çıkış özellikleri.....	875
A.6	CPU 1217C .....	880
A.6.4.1	Analog giriş özellikleri .....	888
A.6.4.2	CPU yerleşik analog girişlerinin basamak tepkisi .....	889
A.6.4.3	CPU'nun gömülü analog girişleri için örnekleme süresi.....	889
A.6.4.4	Analog girişlerin gerilim ölçme aralıkları (CPU'lar) .....	890
A.6.4.5	Analog çıkış özellikleri.....	890
A.7	Dijital sinyal modülleri (SM'ler).....	893
A.8	Analog sinyal modülleri (SM'ler) .....	908
A.9	Termokupl ve RTD sinyal modülleri (SM'ler) .....	920
A.9.1.1	Bir Termokuplun temel çalışması.....	923
A.9.1.2	SM 1231 Termokupl için seçim Çizelgesi .....	924
A.9.2.1	SM 1231 RTD için seçim Çizelgeleri.....	929
A.10	Teknoloji modülleri .....	932
A.10.1.1	SM 1278 4xIO-Link Master sinyal modül özellikleri .....	932
A.10.1.2	SM 1278 4xIO-Link Master SM bağlantı diyagramları .....	935
A.11	Dijital sinyal kartları (SB'ler) .....	936
A.12	Analog sinyal kartları (SB'ler).....	947
A.12.3.1	Analog girişlerin basamak tepkileri .....	951
A.12.3.2	Analog girişler için örnekleme süresi ve güncelleme süresi .....	951
A.12.3.3	Akım ve gerilim (SB ve SM) için analog girişlerin ölçme aralığı .....	951
A.12.3.4	Akım ve gerilim (SB ve SM) için analog çıkışların ölçme aralığı .....	952
A.12.4.1	SB 1231 1 analog Termokupl giriş özellikleri.....	954
A.12.4.2	Bir Termokupl için temel çalışma .....	955
A.12.5.1	SB 1231 1 analog RTD giriş özellikleri .....	958
A.12.5.2	SB 1231 RTD için seçim çizelgesi .....	961
A.13	BB 1297 Batarya paneli .....	963
A.14	Haberleşme ara yüzleri .....	965
A.14.1.1	CM 1242-5 PROFIBUS DP Slave.....	965
A.14.1.2	CM 1243-5 PROFIBUS DP Master.....	966
A.14.2.1	CP 1242-7 GPRS.....	968
A.14.3.1	AS-i master CM 1243-2 için teknik veri.....	971
A.14.3.2	AS-i master CM 1243-2 için elektrik bağlantıları.....	972
A.14.4.1	CB 1241 RS485 özellikler .....	974
A.14.4.2	CM 1241 RS232 özellikler .....	976
A.14.4.3	CM 1241 RS422/485 özellikler .....	977
A.15	TeleService (TS Adapter ve TS Adapter modular) .....	979

*İçindekiler çizelgesi**1.1 S7-1200 PLC tanıtımı*

A.16	SIMATIC bellek kartları .....	979
A.17	Giriş simülatörleri .....	979
A.18	S7-1200 Potansiyometre modülü.....	981
A.19	I/O genişleme kablosu.....	982
A.20	Eşlik eden ürünler .....	983
<b>B</b>	<b>Güç ihtiyacının hesaplanması .....</b>	<b>985</b>
<b>C</b>	<b>Sipariş numaraları .....</b>	<b>989</b>
C.1	CPU modelleri .....	989
C.2	Sinyal modülleri(SM'ler), sinyal kartları (SB'ler) ve batarya panelleri(BB) .....	989
C.3	Haberleşme .....	991
C.4	Diğer modüller .....	992
C.5	Bellek kartları .....	992
C.6	Temel HMI cihazları .....	992
C.7	Yedek parçalar ve diğer donanım .....	993
C.8	Programlama yazılımı .....	994
C.9	Dokümantasyon .....	994
<b>D</b>	<b>Bir V3.0 CPU'yu bir V4.0 CPU ile değiştirme .....</b>	<b>995</b>
	<b>Dizin .....</b>	<b>999</b>





# Ürüne genel bir bakış

# 1

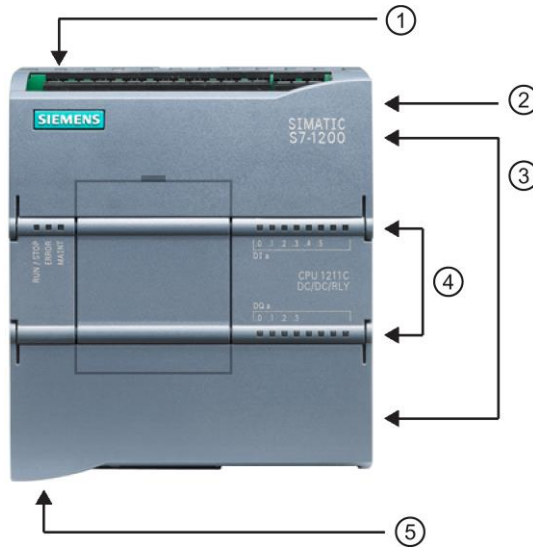
## 1.1 S7-1200 PLC tanıtımı

S7-1200 kontrolör, otomasyon ihtiyaçlarınızın desteklenmesinde çok sayıdaki farklı cihazlarınızı kontrol etmek için esneklik ve güç sağlar. Kompakt tasarım, esnek konfigürasyon ve güçlü komut dizisinin bir araya getirilmesi çok çeşitli uygulamaların kontrollü için S7-1200'ü mükemmel bir çözüm haline getirir.

Güçlü bir kontrolör oluşturmak için CPU; bir mikroişlemci, bir bütünleşik güç kaynağı, giriş ve çıkış devreleri, yerleşik PROFINET, yüksek hızlı hareket kontrol I/O ve kart üzerinde analog girişleri kompakt bir kutuda toplar.

Programınızı yükledikten sonra, uygulamanızdaki cihazları izlemek ve kontrol etmek için CPU gerekli mantığa sahiptir. CPU, girişleri ve çıkışlardaki değişiklikleri sizin kullanıcı programınızın mantığına göre izler. Kullanıcı programı; Boolean mantığı, sayma, zamanlama, karmaşık matematik işlemleri ve diğer akıllı cihazlar ile haberleşmeyi içerir.

CPU, bir PROFINET ağı üzerinde haberleşme için bir PROFINET portu sağlar. PROFIBUS, GPRS, RS485 veya RS232 ağları üzerinde haberleşme yapabilmek için ilave modüller mevcuttur.



- ① Güç konektörü
- ② Bellek kartı yuvası, üst kapak altında
- ③ Sökülebilir kullanıcı bağlantı konektörü, kapakların arkasında
- ④ Kart üstündeki I/O için durum/statü LED'leri
- ⑤ PROFINET konektörü, CPU'nun altında

Çeşitli güvenlik özellikleri hem CPU'ya hem de kontrol programına erişimi korumaya yardımcı olurlar:

- Her CPU parola koruması sağlar (Sayfa 190), böylece sizin CPU fonksiyonlarına erişimi konfigüre etmenize yol açar.
- Kodu, belirli bir blok içinde saklamak için "koruma yapmayı" (Sayfa 193) kullanabilirsiniz.
- Programınızı belirli bir bellek kartında veya CPU'da tutmak için kopyalama koruma (Sayfa 195) kullanabilirsiniz.

## Ürüne genel bir bakış

### 1.1 S7-1200 PLC tanıtımı

Çizelge 1- 1 CPU modellerinin karşılaştırılması

Özellik		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Fiziksel boyut (mm)		90 x 100 x 75	90 x 100 x 75	110 x 100 x 75	130 x 100 x 75	150 x 100 x 75
Kullanıcı belleği	İş	30 Kbytes	50 Kbytes	75 Kbytes	100 Kbytes	125 Kbytes
	Yük	1 Mbyte	1 Mbyte	4 Mbytes	4 Mbytes	4 Mbytes
	Tutan	10 Kbytes	10 Kbytes	10 Kbytes	10 Kbytes	10 Kbytes
Karta yerleşik I/O	Dijital	6 giriş/4	8 giriş/6	14 giriş/10	14 giriş/10	14 giriş/10
		çıkış	çıkış	çıkış	çıkış	çıkış
	Analog	2 giriş	2 giriş	2 giriş	2 giriş / 2	2 giriş / 2
				çıkış	çıkış	
Proses imaj boyutu	Girişler (I)	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes
	Çıkışlar (Q)	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes
Bit bellek (M)		4096 bytes	4096 bytes	8192 bytes	8192 bytes	8192 bytes
Sinyal modülü (SM) genişlemesi		Yok	2	8	8	8
Sinyal kartı (SB), Batarya kartı (BB), veya haberleşme kartı (CB)		1	1	1	1	1
Haberleşme modülü (CM) (sol tarafa genişleme)		3	3	3	3	3
Yüksek- hızlı sayıcı	Toplam	Yerleşik veya SB girişlerinde kullanmak için 6'ya kadar yapılandırılmış				
	1 MHz	--	--	--	--	lb.2 - lb.5
	100/180 kHz	la.0 - la.5	la.0 - la.5	la.0 - la.5	la.0 - la.5	la.0 - la.5
	30/120 kHz	--	la.6 - la.7	la.6 - lb.5	la.6 - lb.5	la.6 - lb.1
Pals çıkışları <sup>2</sup>	Toplam	Yerleşik veya SB girişlerinde kullanmak için 4'e kadar yapılandırılmış				
	1 MHz	--	--	--	--	Qa.0 - Qa.3
	100 kHz	Qa.0 - Qa.3	Qa.0 - Qa.3	Qa.0 - Qa.3	Qa.0 - Qa.3	Qa.4 - Qb.1
	20 kHz	--	Qa.4 - Qa.5	Qa.4 - Qb.1	Qa.4 - Qb.1	--
Bellek kartı		SIMATIC bellek kartı (opsiyonel)				
Gerçek zaman saat tutma süresi		20 gün, typ./12 gün min. 40 C derecede (bakımsız Süper Kapasitör)				
PROFINET Ethernet haberleşme portu		1	1	1	2	2
Gerçek math yürütme hızı		2.3 µs/komut				
Boolean yürütme hızı		0.08 µs/ komut				

<sup>1</sup> HSC operasyonun çeyrek modu için konfigüre edildiği durumda daha düşük hız uygulanabilir.

<sup>2</sup> Röle çıkışları olan CPU modellerinde pals çıkışlarını kullanmak için bir dijital sinyal (SB) kurulumu yapmanız gerekir

Farklı CPU modelleri, değişen uygulamalarınız için etkin çözümler oluşturmaya yardımcı olan özellik ve yetenek çeşitliliği sağlar. Özel bir CPU hakkında ayrıntılı bilgi için teknik özelliklere bakınız (Sayfa 829).

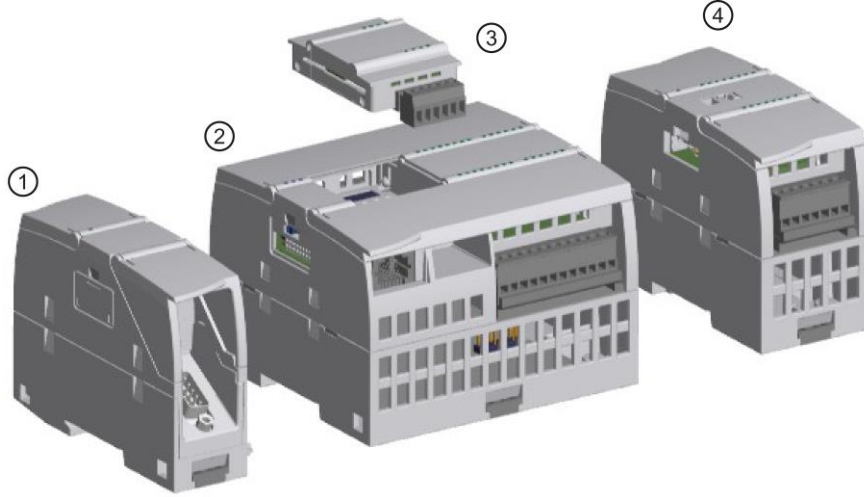


Çizelge 1- 2 S7-1200 tarafından desteklenen bloklar, zamanlayıcılar ve sayıcılar

Element	Açıklama	
Bloklar	Tip	OB, FB, FC, DB
	Boyut	30 Kbytes (CPU 1211C) 50 Kbytes (CPU 1212C) 64 Kbytes (CPU 1214C, CPU 1215C ve CPU 1217C)
	Miktar	1024 bloka kadar toplam(OB'ler + FB'ler + FC'ler + DB'ler)
	Yuvalama derinliği	16 program çevriminden veya OB; 6 herhangi bir interrupt olayı OB
	İzleme	2 kod bloğunun durumu aynı anda izlenebilir
	OB'ler	Program çevrimi
Başlama		Çoklu
Zaman-gecikme interrupt'ları		4 (1 olay başına)
Döngüsel interrupt'lar		4 (1 olay başına)
Donanım interrupt'ları		50 (1 olay başına)
Zaman hata interrupt'ları		1
Diagnostik hata interrupt'ları		1
Modülleri çekme veya takma		1
Şasi ve istasyon arızası		1
Gün içindeki saat		Çoklu
Durum		1
Güncelleme		1
Profil		1
Zamanlayıcılar (Timers)	Tip	IEC
	Miktar	Sadece bellek boyutu tarafından sınırlanır
	Bellek	DB içinde yapı, 16 byte zamanlayıcı başına
Sayıcılar (Counters)	Tip	IEC
	Miktar	Sadece hafıza büyüklüğü ile sınırlıdır
	Bellek	DB içinde yapı, yazma tipine bağlı boyut
		<ul style="list-style-type: none"> <li>• SInt, USInt: 3 byte</li> <li>• Int, UInt: 6 byte</li> <li>• DInt, UDInt: 12 byte</li> </ul>

## 1.2 CPU genişleme kapasitesi

S7-1200 ailesi, CPU'nun yeteneklerini genişletmek için ilave I/O veya diğer haberleşme protokolleri ile çeşitli modüller ve takılabilir kartlar sağlar. Özel bir modül hakkında ayrıntılı bilgi için teknik özelliklerini inceleyiniz (Sayfa 829).



- ① Haberleşme modülü (CM) veya haberleşme işlemcisi (CP)
- ② CPU
- ③ Sinyal kartı (SB), haberleşme kartı (CB) veya Batarya kartı (BB)
- ④ Sinyal modülü (SM)

Çizelge 1- 3 Dijital sinyal modülleri ve sinyal kartları

Tip	Sadece Giriş	Sadece Çıkış	Giriş/Çıkış Kombinasyonu
③ dijital SB	<ul style="list-style-type: none"> <li>• 4 x 24 VDC Giriş, 200 kHz</li> <li>• 4 x 5 VDC Giriş, 200 kHz</li> </ul>	<ul style="list-style-type: none"> <li>• 4 x 24 VDC Çıkış, 200 kHz</li> <li>• 4 x 5 VDC Çıkış, 200 kHz</li> </ul>	<ul style="list-style-type: none"> <li>• 2 x 24 VDC Giriş / 2 x 24 VDC Çıkış</li> <li>• 2 x 24 VDC Giriş / 2 x 24 VDC Çıkış, 200 kHz</li> <li>• 2 x 5 VDC Giriş / 2 x 5 VDC Çıkış, 200 kHz</li> </ul>
④ dijital SM	<ul style="list-style-type: none"> <li>• 8 x 24 VDC Giriş</li> </ul>	<ul style="list-style-type: none"> <li>• 8 x 24 VDC Çıkış</li> <li>• 8 x Röle çıkışı</li> <li>• 8 x Röle çıkışı (konum değiştirme)</li> </ul>	<ul style="list-style-type: none"> <li>• 8 x 24 VDC Giriş / 8 x 24 VDC Çıkış</li> <li>• 8 x 24 VDC Giriş / 8 x Röle Çıkışı</li> <li>• 8 x 120/230 VAC Giriş / 8 x Röle Çıkışı</li> </ul>
	<ul style="list-style-type: none"> <li>• 16 x 24 VDC Giriş</li> </ul>	<ul style="list-style-type: none"> <li>• 16 x 24 VDC Çıkış</li> <li>• 16 x Röle çıkışı</li> </ul>	<ul style="list-style-type: none"> <li>• 16 x 24 VDC Giriş / 16 x 24 VDC Çıkış</li> <li>• 16 x 24 VDC Giriş / 16 x Röle Çıkışı</li> </ul>

Çizelge 1- 4 Analog sinyal modülleri ve sinyal kartları

Tip	Sadece Giriş	Sadece Çıkış	Giriş/Çıkış Kombinasyonu
③ analog SB	<ul style="list-style-type: none"> <li>1 x 12 bit Analog Giriş</li> <li>1 x 16 bit RTD</li> <li>1 x 16 bit Termokupl</li> </ul>	<ul style="list-style-type: none"> <li>1 x Analog Çıkış</li> </ul>	-
④ analog SM	<ul style="list-style-type: none"> <li>4 x Analog Giriş</li> <li>4 x Analog Giriş x 16 bit</li> <li>8 x Analog Giriş</li> <li>Termokupl:               <ul style="list-style-type: none"> <li>4 x 16 bit TC</li> <li>8 x 16 bit TC</li> </ul> </li> <li>RTD:               <ul style="list-style-type: none"> <li>4 x 16 bit RTD</li> <li>8 x 16 bit RTD</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>2 x Analog Çıkış</li> <li>4 x Analog Çıkış</li> </ul>	<ul style="list-style-type: none"> <li>4 x Analog Giriş / 2 x Analog Çıkış</li> </ul>

Çizelge 1- 5 Haberleşme ara yüzleri

Modül	Tip	Açıklama
① Haberleşme modülü (CM)	RS232	Full-duplex
	RS422/485	Full-duplex (RS422) Half-duplex (RS485)
	PROFIBUS Master	DPV1
	PROFIBUS Slave	DPV1
	AS-i Master (CM 1243-2)	AS-Arayız
① Haberleşme işlemcisi (CP)	Modem bağlantısı	GPRS
③ Haberleşme kartı (CB)	RS485	Half-duplex
TeleServis <sup>1</sup>	TS Adapter IE Basic	CPU'ya bağlantı
	TS Adapter GSM	GSM/GPRS
	TS Adapter Modem	Modem
	TS Adapter ISDN	ISDN
	TS Adapter RS232	RS232

1 TS Adapter IE Basic, CPU'nun PROFINET portuna çeşitli haberleşme ara yüzlerini bir Ethernet kablosu kullanarak bağlamanıza izin verir. TS Adapter IE Basic üzerine üçe kadar TS adaptör modülü takabilirsiniz.

Çizelge 1- 6 Teknoloji modülleri

Modül	Tip	Açıklama
④ IO Link	SM 1278 4xIO-Link Master	4 IO link slaves destekler

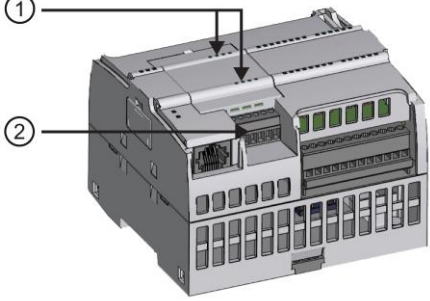
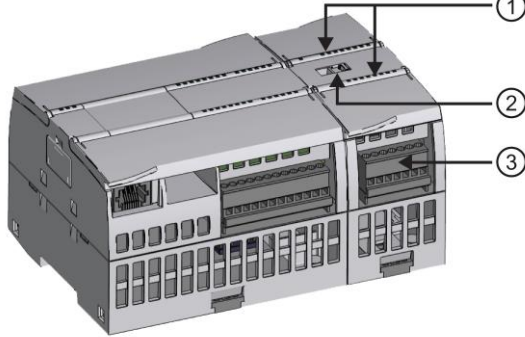
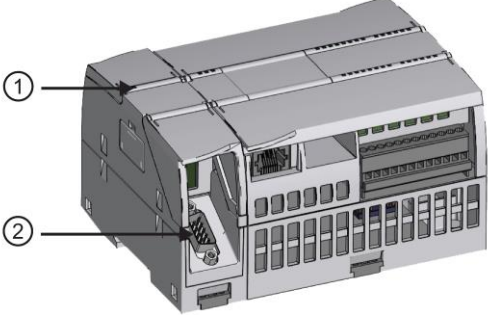
*Ürüne genel bir bakış**1.2 CPU genişleme kapasitesi*

## Çizelge 1- 7 Diğer kartlar

Modül	Açıklama
③ Batarya kartı	CPU önündeki genişleme kartı ara yüzüne takılır. Gerçek zaman saat için uzun süreli besleme sağlar.

## 1.3 S7-1200 modülleri

Çizelge 1- 8 S7-1200 genişleme modülleri

Modül tipi	Açıklama		
<p>CPU bir genişleme kartı takılmasını destekler:</p> <ul style="list-style-type: none"> <li>Bir sinyal kartı (SB) sizin CPU'nuz için ilave I/O sağlar</li> <li>SB bağlantıları CPU'nun ön tarafındadır.</li> <li>Bir haberleşme kartı (CB) sizin CPU'nuzla diğer bir haberleşme kartı ilave etmenize izin verir.</li> <li>Bir batarya kartı (BB) gerçek zaman saatin uzun süre beslenmesini sağlar.</li> </ul>		<p>①</p>	SB üzerindeki durum LED'leri
		<p>②</p>	Sökülebilir kullanıcı kablo konektörü
<p>Sinyal modülleri (SM'ler), CPU'ya ilave fonksiyonları yürütme desteği sağlar. SM'ler CPU'nun sağ tarafına bağlanır.</p> <ul style="list-style-type: none"> <li>Dijital I/O</li> <li>Analog I/O</li> <li>RTD ve Termokupl</li> <li>SM 1278 IO-Link Master</li> </ul>		<p>①</p>	Durum LED'leri
		<p>②</p>	Veri yolu konektörü kayar tırnağı
		<p>③</p>	Sökülebilir kullanıcı kablo konektörü
<p>Haberleşme modülleri (CM'ler) ve haberleşme işlemcileri (CP'ler), CPU'ya haberleşme seçenekleri ilave ederler, PROFIBUS veya RS232/RS485 için bağlanabilme (PtP, Modbus veya USS için), veya AS-i master. Bir CP diğer tip haberleşmeler için yetenek sağlar, bir CPU'nun bir GPRS şebekesi üzerinden bağlantısı gibi.</p> <ul style="list-style-type: none"> <li>CPU, 3 CM'ne veya CP'ne kadar destekler.</li> <li>Her bir CM veya CP, CPU'nun sol tarafına bağlanır (veya diğer bir CM veya CP'nin sol tarafına)</li> </ul>		<p>①</p>	Durum LED'leri
		<p>②</p>	Haberleşme konektörü

## 1.4 Yeni özellikler

Aşağıdaki özellikler bu sürümde yenidir:

- S7-1200, yeni organizasyon bloklarını (OB'ler) (Sayfa 85) öncelik seviyeleri ve kesicilerdeki farklılıklar ile destekler.
- Web sunucu (Sayfa 601) şimdi standart web sayfalarını ve bir mobil cihazdan ve bir PC'den kullanıcı-tanımlı Web sayfalarını görüntülemeleri desteklemektedir. Standart web sayfaları İngilizce, Almanca, Fransızca, İspanyolca, İtalyanca ve bu sürümle beraber basitleştirilmiş Çince dillerinde hazırlanmıştır.
- "Download in Run" (Sayfa 803) özelliği şimdi RUN modunda indirebileceğiniz en fazla 20 adet bloğu destekler. Mevcut veri bloklarında ve fonksiyon bloklarında etiketler ekleyebilir ve değiştirebilir ve RUN modunda değiştirilmiş veri bloklarını yükleyebilirsiniz.
- STEP 7'nin çevrim içi ve diyagnostik araçları sizin CPU'nuz, sinyal modelleri, haberleşme modülleri ve ekli sinyaller veya haberleşme kartları için donanım yazılımı (Sayfa 791)güncellemesini gerçekleştirmek için vasıtalar sağlar
- STEP 7 sizin V4.0 S7-1200 CPU'lar ile kullanabileceğiniz bir izleme ve lojik analizör fonksiyonu (Sayfa 813) içerir. Tanımladığınız trigger şartını CPU karşıladığında, bu özellik ile izlemek ve kaydetmek istediğiniz belirli veriyi yapılandırabilirsiniz.
- CPU kayıt edilen veriyi saklar. STEP 7 kayıt edilen veriyi yeniden inceleme ve analiz için araçlar sağlar.
- Yeni programlama komutları:
  - **Set tag on signal edge:** R\_TRIG (Sayfa 209), F\_TRIG (Sayfa 209)
  - **Write local time:** WR\_LOC\_T (Sayfa 286)
  - **String maximum length:** MAX\_LEN (Sayfa 306)
  - **Time of day interrupts:** SET\_TINTL (Sayfa 337), CAN\_TINT (Sayfa 339), ACT\_TINT (Sayfa 339), QRY\_TINT (Sayfa 340)
  - **Process recipes:** RecipeExport (Sayfa 374), RecipeImport (Sayfa 376)
  - **Address handling:** LOG2GEO (Sayfa 404), RD\_ADDR (Sayfa 406)
  - **Motion control:** MC\_WriteParam (Sayfa 478), MC\_ReadParam (Sayfa 406)
  - **Enable / disable password:** ENDIS\_PW (Sayfa 259)
- HSC (Sayfa 411) yenilikleri, her hangi bir yerleşik veya SB dijital çıkışına atanacak herhangi bir HSC talimat girişi veya çıkışına izin verir.
- PTO/PWM (Sayfa 481) yenilikleri SB dijital çıkışına atanacak herhangi bir her hangi bir PTO/PWM talimat girişi veya çıkışına izin verir.
- Geliştirilmiş kütüphane (Sayfa 177) özellikleri, sürüm belirlemeyi içerir.

## S7-1200 için yeni modüller

Yeni modüller S7-1200 CPU'nun gücünü genişletir ve sizin otomasyon ihtiyaçlarınızı karşılamak için esneklik sağlar:


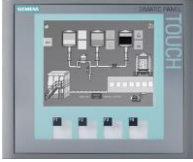

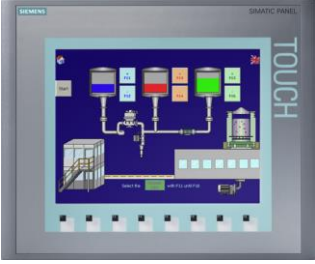
- Yüksek hızlı diferansiyel noktalı yeni CPU 1217C DC/DC/DC (Sayfa 879)
- Yeni ve geliştirilmiş S7-1200 sinyal modülleri. Yeni sinyal modülleri (6ES7 2xx-xxx32-0XB0) mevcut olan sinyal modüllerinin (6ES7 2xx-xxx30-0XB0) yerini almaktadır. Yeni modüller aşağıdakileri karşılar:
  - Analog giriş ve çıkış modüllerine 4-20 mA aralığı eklendi
  - Eklenen 4-20 mA analog giriş ve çıkış modülleri kullanıldığında kablo kopması algılama
  - Alan kablolama konektörü role çıkışlı modüllere eklendiğinde hataları önlemek için konektör anahtarlama
  - Yedek parça uyumluluğu: Mevcut modüllerin yerine herhangi bir değişiklik yapmaksızın yeniden düzenlenmiş modülü kullanabilirsiniz.
- S7-1200 CPU'lar için kullanılmak üzere yeni yedek parçalar (Sayfa 993) mevcut
- Yeni CPU 1217C Input Simulator (Sayfa 978) (6ES7 274-1XK30-0XA0)
- Yeni SM 1278 4xIO-Link Master (Sayfa 815) (6ES7 278-4BD32-0XB0) hem bir sinyal modülü hem de bir haberleşme modülü olarak fonksiyon görür, 4 IO-Link slaves'e (3-telli bağlantı) kadar veya 4 standart aktuatöre veya standart enkodere kadar bağlantıya izin verir.
- Yeni S7-1200 Potansiyometre modülü (Sayfa 980) (6ES7 274-1XA30-0XA0)
- S7-1200 için yeni CM CANopen (Sayfa 983) takılabilir modül, CANopen cihazlarının S7-1200 PLC'ye bağlantılarını sağlar. Hem master hem de slave olarak yapılandırılabilir.

## V3.0 CPU'nuzu bir V4.0 CPU ile değiştirme

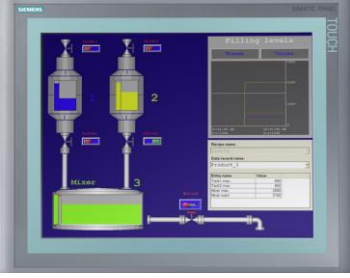
S7-1200 V3.0 CPU'nuzu bir S7-1200 V4.0 CPU ile değiştiriyorsunuz, sürümlerdeki doküman edilen farklılıklara (Sayfa 995) dikkat ediniz.

## 1.5 Temel HMI paneller

SIMATIC HMI Basic Panels temel operatör kontrolü ve izleme görevleri için dokunmatik ekran sağlar. Bütün paneller IP65 koruma derecesine ve CE, UL, cULus ve NEMA 4x sertifikasyonuna sahiptirler.

Temel HMI Paneli	Açıklama	Teknik veri
 <p>KP 300 Basic PN</p>	<p>3.6" dokunma duyarlı klavye, 10 serbestçe konfigüre edilebilir dokunmatik anahtarlı</p> <ul style="list-style-type: none"> <li>• Mono (STN, siyah/beyaz)</li> <li>• 87 mm x 31 mm (3.6")</li> <li>• Arka ışığı rengi programlamalı (beyaz, yeşil, sarı veya kırmızı)</li> <li>• Çözünürlük: 240 x 80</li> </ul>	<ul style="list-style-type: none"> <li>• 250 etiket</li> <li>• 50 proses ekranları</li> <li>• 200 alarm</li> <li>• 25 eğri</li> <li>• 40 KB tarif belleği</li> <li>• 5 tarif, 20 veri kaydı, 20 giriş</li> </ul>
 <p>KTP 400 Basic PN</p>	<p>4" dokunmatik ekran, 4 temas anahtarlı</p> <ul style="list-style-type: none"> <li>• Mono (STN, gri ölçek)</li> <li>• 76.79 mm x 57.59 mm (3.8") Dikey veya yatay</li> <li>• Çözünürlük: 320 x 240</li> </ul>	<ul style="list-style-type: none"> <li>• 250 etiket</li> <li>• 50 proses ekranları</li> <li>• 200 alarm</li> <li>• 25 eğri</li> <li>• 40 KB tarif belleği</li> <li>• 5 tarif, 20 veri kaydı, 20 giriş</li> </ul>
 <p>KTP 600 Basic PN</p>	<p>6" dokunmatik ekran, 6 temas anahtarlı</p> <ul style="list-style-type: none"> <li>• Renk (TFT, 256 renk) veya Mono (STN, gri ölçekler)</li> <li>• 115.2 mm x 86.4 mm (5.7") Dikey veya yatay</li> <li>• Çözünürlük: 320 x 240</li> </ul>	<ul style="list-style-type: none"> <li>• 500 etiket</li> <li>• 50 proses ekranları</li> <li>• 200 alarm</li> <li>• 25 eğri</li> <li>• 40 KB tarif belleği</li> <li>• 5 tarif, 20 veri kaydı, 20 giriş</li> </ul>
 <p>KTP 1000 Basic PN</p>	<p>10" dokunmatik ekran, 8 temas anahtarlı</p> <ul style="list-style-type: none"> <li>• Renk (TFT, 256 renk)</li> <li>• 211.2 mm x 158.4 mm (10.4")</li> <li>• Çözünürlük: 640 x 480</li> </ul>	<ul style="list-style-type: none"> <li>• 500 etiket</li> <li>• 50 proses ekranları</li> <li>• 200 alarm</li> <li>• 25 eğri</li> <li>• 40 KB tarif belleği (entegre flash)</li> <li>• 5 tarif, 20 veri kaydı, 20 giriş</li> </ul>



Temel HMI Paneli	Açıklama	Teknik veri
	<p>15" dokunmatik ekran</p> <ul style="list-style-type: none"><li>• Renk (TFT, 256 renk)</li><li>• 304.1 mm x 228.1 mm (15.1")</li><li>• Çözünürlük: 1024 x 768</li></ul>	<ul style="list-style-type: none"><li>• 500 etiket</li><li>• 50 proses ekranları</li><li>• 200 alarm</li><li>• 25 eğri</li><li>• 40 KB tarif belleği (entegre flash)</li><li>• 5 tarif, 20 veri kaydı, 20 giriş</li></ul>

TP 1500 Basic PN

### Ayrıca bakınız

Müşteri destek (<http://www.siemens.com/automation/>)

*Ürüne genel bir bakış*

*1.5 Temel HMI paneller*

---

## STEP 7 programlama yazılımı

# 2

STEP 7, uygulamalarınızı kontrol etmek için gerekli lojiği geliştirmek, düzenlemek ve izlemek amacıyla kullanışlı bir ortam sağlar. STEP 7, projenizdeki kontrolörler ve HMI cihazları gibi bütün cihazların yönetimi ve yapılandırılması için araçları da içerir. İhtiyaç duyacağınız bilgiyi bulmaya yardımcı olmak için STEP 7 kapsamlı bir çevrimiçi yardım sistemi sağlar.

STEP 7, uygulamalarınız için kontrol programı geliştirmede uygunluk ve etkinlik amacıyla standart programlama dilleri sağlar.

- LAD (merdiven lojik) (Sayfa 180) bir grafiksel programlama dilidir ve gösterilişi devre diyagramları temeline göreder.
- FBD (Fonksiyon Blok Diyagram) (Sayfa 181) bir programlama dilidir ve Boolean cebirinde kullanılan grafiksel lojik sembolleri temeline göreder.
- SCL (yapısal kontrol dili) (Sayfa 181) metin temellidir ve yüksek-seviye programlama dilidir.

Bir kod bloğu oluşturduğunuzda, bu blok tarafından kullanılacak programlama dilini seçersiniz. Kullanıcı programınız herhangi bir veya bütün programlama dillerinde oluşturulan kod bloklarını kullanır.

---

### Not

STEP 7, TIA Portal'ının programlama ve yapılandırma yazılımı bileşenidir.

TIA Portal, STEP 7'ye ilaveten, ayrıca runtime süreç görselleştirme tasarımı ve yürütmesi için WinCC'yi de içerir ve STEP 7'de olduğu gibi WinCC için de online yardımı bünyesinde bulundurur.

---

## 2.1 Sistem gerekleri

STEP 7'yi kurmak için Administrator yetkisiyle giriş yapmalısınız.

Çizelge 2- 1 Sistem gerekleri

Donanım/yazılım	Gerekler
İşlemci tipi	Pentium M, 1.6 GHz veya benzeri
RAM	1 GB
Hazır sabit disk alanı	2 GB, sistem sürücüsü C:\ üzerinde
İşletim sistemleri	<ul style="list-style-type: none"><li>Windows 7 Home Premium veya yükseği (STEP 7 Basic sadece, STEP 7 Professional için desteklenmez)</li><li>Windows 7 veya yükseği (Professional, Enterprise, Ultimate)</li></ul>
Grafik kartı	32 MB RAM 24- bit renk derinliği
Ekran çözünürlüğü	1024 x 768
Ağ	20 Mbit/s Ethernet veya daha hızlısı
Optik sürücü	DVD-ROM

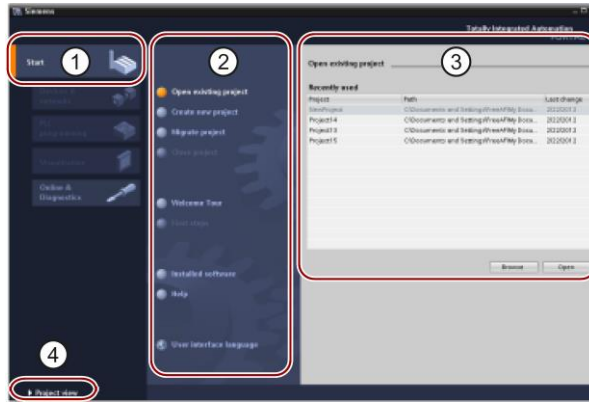
## 2.2 İşi daha kolay yapmak için farklı görüşler

STEP 7; kontrolör lojiğini geliştirmek, HMI görsellemesini yapılandırmak ve ağ haberleşmesini kurmak için kullanıcı-dostu bir ortam sağlar.

Verimliliğinizi artırmaya yardımcı olmak için STEP 7, projenin iki farklı görünümünü sağlar: Araçların işlevselliği temelinde organize edilmiş görev odaklı portallar dizini(Portal görünümü) veya proje içindeki elementlerin proje odaklı bir görünümü (Proje görünümü).

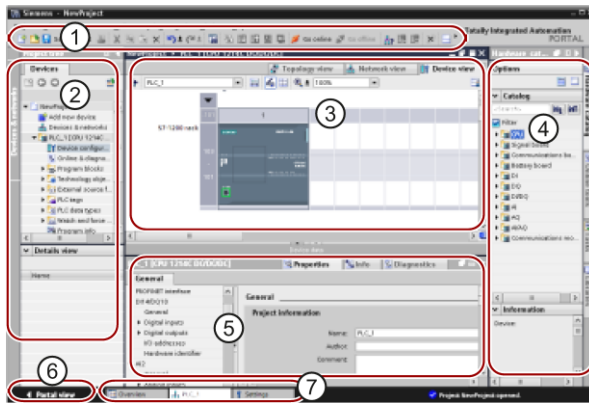
Sizin için en etkin çalışmaya yardımcı olan görünümü seçiniz.

Bir tıklama ile Portal görünümü ile Proje görünümü arasında geçiş yapabilirsiniz



Portal görünüm menüsü

- ① Farklı görevler için portallar
- ② Seçilecek portal için görevler
- ③ Seçilecek aksiyon için seçme paneli
- ④ Proje görünüm menüsünde değişiklikler



Proje görünüm menüsü

- ① Menüler ve araç çubukları
- ② Proje navigatörü
- ③ İş alanı
- ④ Görev kartları
- ⑤ Kontrolör penceresi
- ⑥ Proje görünüm menüsünde değişiklikler
- ⑦ Editör çubuğu

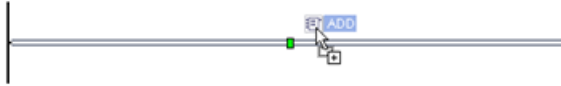
Bu bileşenlerin tümünün tek bir yerde olmasıyla, projenizin her tarafına kolay erişime sahipsiniz. Örneğin denetim penceresi, çalışma alanında seçmiş olduğunuz nesne için özellikleri ve bilgileri gösterir. Farklı nesnelere seçtiğinizde, denetim penceresi yapılandırabileceğiniz özellikleri görüntüler. Denetim penceresi, diyagnostik (tanılama) bilgilerini ve diğer mesajları görmemiz için içinde sekmeler bulundurmaktadır.

Açık olan editörlerin hepsini göstererek editör çubuğu daha hızlı ve etkin çalışmanıza yardımcı olur. Açık editörler arasında geçiş yapmak için basitçe farklı editöre tıklayınız. Dikey veya yatay olarak düzenlenmiş iki editörün beraber görünmesini de düzenleyebilirsiniz. Bu özellik, editörler arasında sürükleyip bırak işlemlerine izin verir.

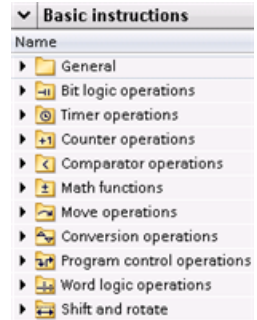
## 2.3 Kolay kullanım araçları

### 2.3.1 Kullanıcı programınıza komutlar ekleme

STEP 7, programınız için komutlar içeren görev kartları sağlar. Komutlar, fonksiyonlarına göre gruplanır.

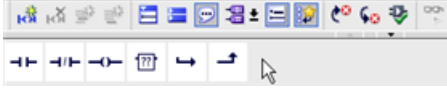


Programınızı oluşturmak için komutları görev kartından bir ağ üzerine sürükleyin.



### 2.3.2 "Favorites" araç çubuğundan komutlara erişim

STEP 7, sık kullandığınız komutlara hızlı erişiminiz için bir "Favorites" araç çubuğu sağlar. Ağınıza yerleştirmek istediğiniz komut için basitçe ilgili ikonu tıklayınız.



(Komut ağacındaki "Favorites" için ikonu çift-tıklayınız.)

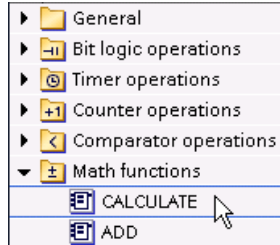


Yeni komutlar ekleyerek "Favorites" işlemini kolayca özelleştirebilirsiniz. Basitçe komutu "Favorites" sürükleyin ve bırakabilirsiniz. Bu komut artık bir tıklama uzaklığındadır!

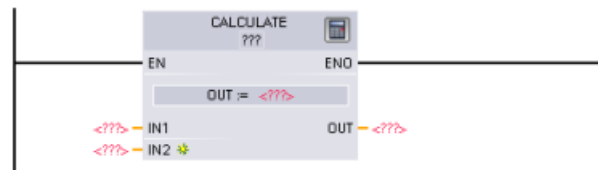


### 2.3.3 Bir basit komut ile bir karmaşık eşitlik oluşturma

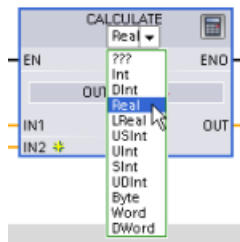
Hesapla komutu, tanımladığınız eşitliğe göre sonucu üretmek için çoklu giriş parametresi üzerinde çalışan bir matematik fonksiyonu oluşturmanıza izin verir.



Basic komut ağacında, Matematik fonksiyonları klasörünü genişletiniz. Komutu programınıza yerleştirmek için Calculate komutuna çift-tıklayınız.



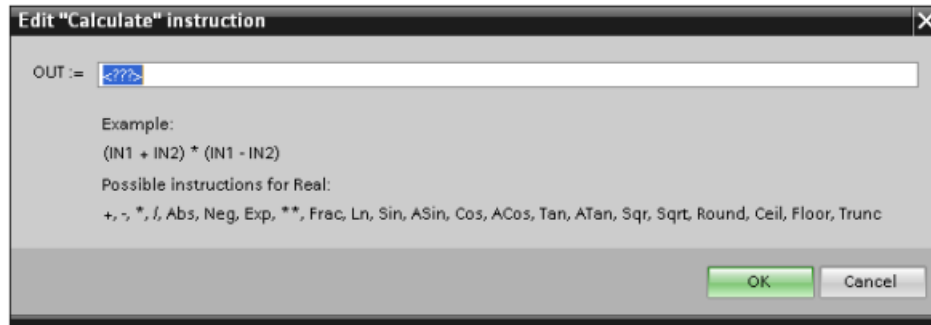
Konfigürasyonu yapılmamış Calculate komutu iki giriş parametresi ve bir çıkış parametresi bulundurulur.



Tıkla "???" ve giriş ve çıkış parametreleri için veri tiplerini seçiniz. (Giriş ve çıkış parametrelerinin hepsi aynı veri tipinde olmalıdır.)  
Bu örnek için, "Real" veri tipini seçiniz.



Eşitliği girmek için "Edit equation" ikonunu tıklayınız.



Bu örnek için, bir ham analog değerini ölçeklemek için izleyen eşitliği giriniz. ("In" ve "Out" gösterimleri Calculate komutunun parametrelerine karşılıktır.)

$$\text{Out value} = ((\text{Out high} - \text{Out low}) / (\text{In high} - \text{In low})) * (\text{In value} - \text{In low}) + \text{Out low}$$

$$\text{Out} = ((\text{in4} - \text{in5}) / (\text{in2} - \text{in3})) * (\text{in1} - \text{in3}) + \text{in5}$$

Burada:            Out value        (Out)        Ölçekli çıkış değeri  
                      In value        (in1)        Analog giriş değeri

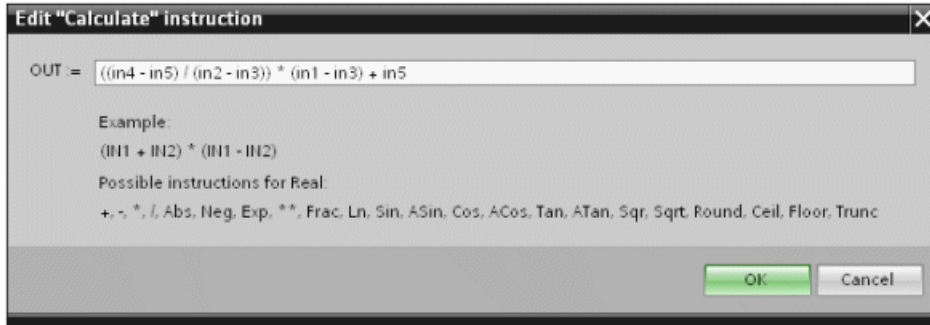
## STEP 7 programlama yazılımı

### 2.3 Kolay kullanım araçları

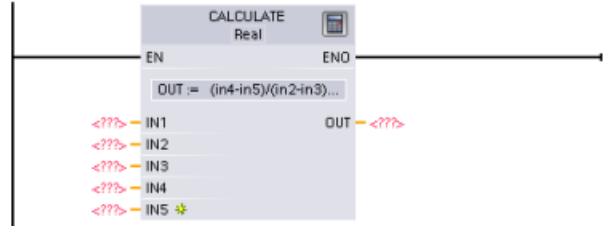
In <sub>high</sub>	(in2)	Ölçekli giriş değeri için üst sınır
In <sub>low</sub>	(in3)	Ölçekli giriş değeri için alt sınır
Out <sub>high</sub>	(in4)	Ölçekli çıkış değeri için üst sınır
Out <sub>low</sub>	(in5)	Ölçekli çıkış değeri için alt sınır

"Edit Calculate" kutusunda, eşitliği parametrelerin isimleriyle giriniz:

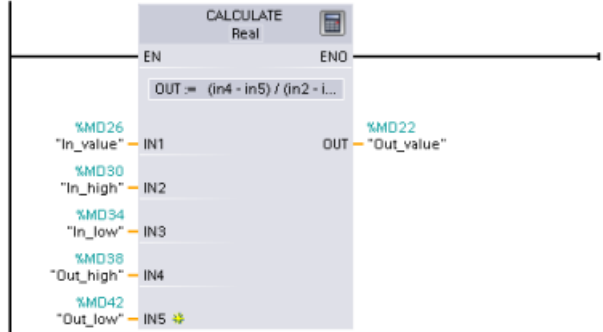
$$OUT = ((in4 - in5) / (in2 - in3)) * (in1 - in3) + in5$$



"OK" butonunu tıkladığınızda, hesapla komutu, komut için gerekli girişleri oluşturur.



Parametrelere karşılık gelen değerler için etiket isimlerini giriniz.





## 2.3.4 Bir LAD veya FBD komutuna girişler veya çıkışlar ekleme

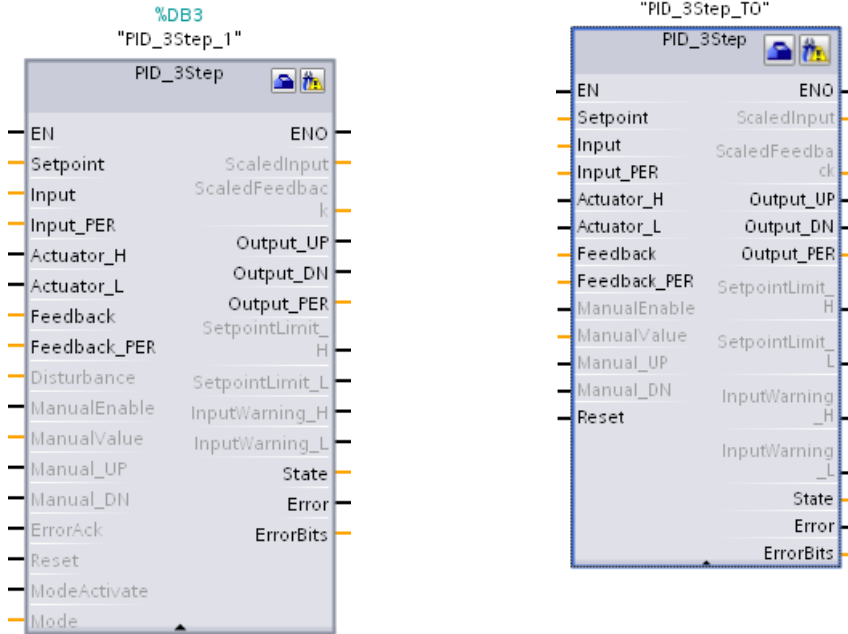


Komutların bazıları ilave girişler veya çıkışlar oluşturmanız için size izin verir.

- Bir giriş veya çıkış eklemek için "Create" ikonuna tıklayınız veya mevcut bulunan IN veya OUT parametrelerinin birisi için bir giriş stub'ı üzerinde sağ-tıklayınız ve "Insert input" komutunu seçiniz.
- Bir giriş veya çıkışı kaldırmak için mevcut olan bir IN veya OUT parametrelerine (orijinal iki girişten daha fazla olduğu durumda) ait stub üzerinde sağ-tıklayınız ve "Delete" komutunu seçiniz.

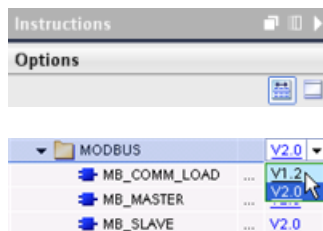
### 2.3.5 Genişletilebilir komutlar

Yalnızca anahtar girişleri ve çıkışlarını görüntüleyen bazı daha karmaşık komutlar genişletilebilir. Girişleri ve çıkışları görüntülemek için komutun altındaki oka tıklayınız.



### 2.3.6 Bir komut için bir versiyon seçmek

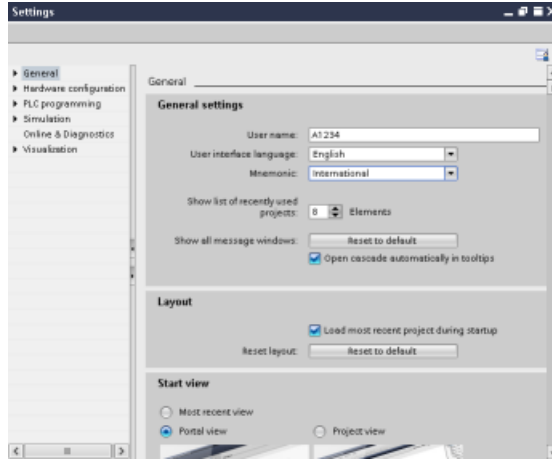
Belirli komut setlerinde (Modbus, PID ve hareket gibi) yapılan geliştirme ve yayımlama çevrimleri bu komutlar birden çok yayımlanmış sürüm oluşturmuştur. STEP 7, daha eski projelerle uyumluluğun ve geçişin sağlanmasına yardımcı olmak için kullanıcı programınıza hangi komut sürümünün ekleneceğini seçmenize izin verir.



Komut ağacının başlıkları ve sütunlarını etkinleştirmek için komut ağacı görev kartı üzerindeki simgeye tıklayınız.

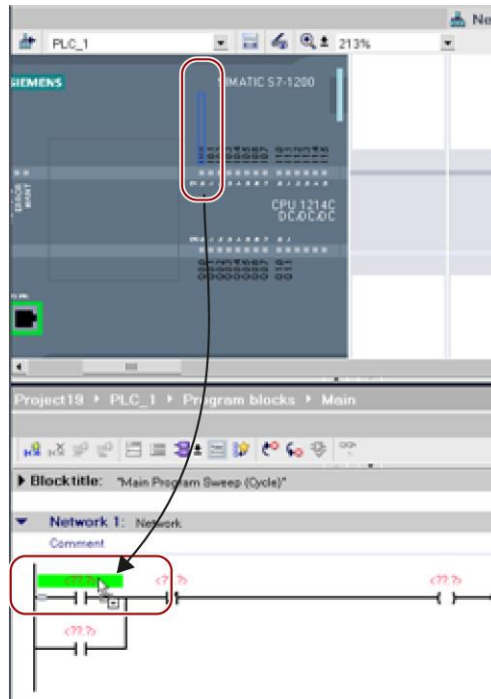
Komutun versiyonunu değiştirmek için açılır listeden uygun versiyonu seçiniz.

### 2.3.7 STEP 7'nin görünüm ve konfigürasyonunu değiştirme



Ara yüzün görünümü, dil veya çalışmalarınızı kayıt edileceği dizin gibi çeşitli ayarları seçebilirsiniz. Bu ayarları değiştirmek için "Option" menüsünden "Settings" komutunu seçiniz.

### 2.3.8 Editörler arasında sürükleme ve bırakma



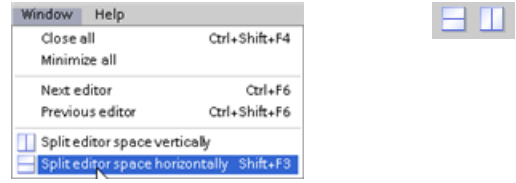
STEP 7, görevleri hızlı ve kolay bir şekilde gerçekleştirmenize yardımcı olmak için bir editörden diğerine elemanları sürükleyip bırakmanıza izin verir.

Örneğin CPU'dan bir girişi, kullanıcı programınızdaki bir komutun adresine sürükleyebilirsiniz.

CPU'nun girişlerini veya çıkışlarını seçebilmek için en az %200 yakınlaştırma yapmalısınız.

Etiket isimlerinin sadece PLC etiket çizelgesinde değil aynı zamanda CPU üzerinde de görüntülediğini fark edeceksiniz.

İki editörü aynı anda görüntülemek için "Split editor" menü komutlarını veya araç çubuğundaki butonları kullanınız.



Açık olan editörler arasında geçiş yapmak için editör çubuğundaki ikonları tıklayınız.



### 2.3.9 CPU'nun çalışma modunu değiştirme

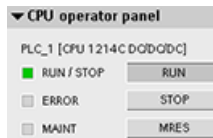
Çalışma modunun (STOP veya RUN) değiştirilmesi için CPU fiziksel bir anahtara sahip değildir.

CPU'nun çalışma modunu değiştirmek için "Start CPU" ve "Stop CPU" araç çubuğu butonlarını kullanınız.



Cihaz konfigürasyon editöründe CPU yapılandırıldığında, başlangıç davranışını CPU'nun özelliklerinden yapılandırabilirsiniz.

"Online and diagnostics" portalı, CPU'nun çalışma modunu değiştirmek için bir operatör paneli de sağlar. "Online tools" görev kartı, çevrimiçi CPU'nun çalışma modunu gösteren bir operatör panelini görüntüler. Operatör paneli, çevrimiçi CPU'nun çalışma modunu değiştirmenize de izin verir.

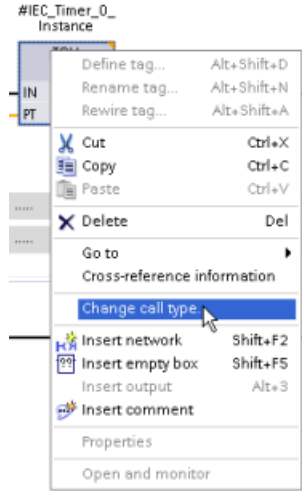


Çalışma modunu (STOP veya RUN) değiştirmek için operatör paneli üzerindeki butonu kullanınız. Operatör paneli ayrıca belleğin reseti için bir MRES butonu sağlar.

RUN/STOP göstergesinin rengi, CPU'nun geçerli çalışma modunu gösterir. Sarı STOP modunu, yeşil RUN modunu gösterir.

Güç altında (enerjili iken) varsayılan çalışma modunu konfigüre etmek için S7 -1200 Sistem Kılavuzunda (Sayfa 81) CPU'nun Çalışma Modlarına bakınız.

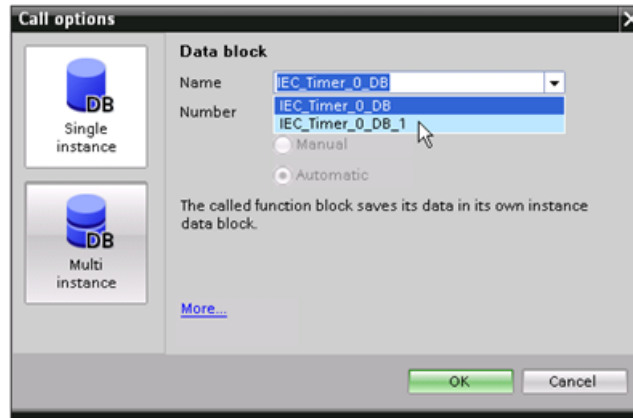
### 2.3.10 Bir DB için çağırılma tipinin değiştirilmesi



STEP 7, bir komut veya bir FB içindeki bir FB için bir DB'nin ilişkilendirmesini kolayca oluşturmanıza ve değiştirmenize izin verir.

- Farklı DB'ler arasındaki ilişkilendirmeleri değiştirebilirsiniz.
- Tek-durum DB ile çoklu-durum DB arasındaki ilişkilendirmeyi değiştirebilirsiniz
- Bir DB oluşturabilirsiniz (Tek-durum DB eksik veya mevcut değilse).

"Change Call type" komutuna komutu veya program editöründeki FB'yi sağ tıklayarak veya "Options" menüsünden "Block call" komutunu seçerek erişebilirsiniz

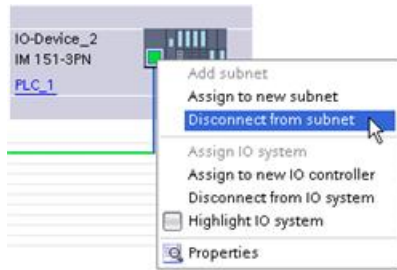
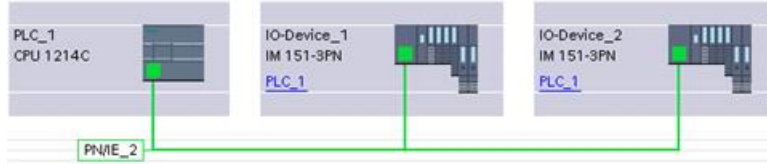


"Call options" diyalogu, tek-durum veya çoklu-durum DB seçimine izin verir.

Ayrıca mevcut DB'lerin açılır listesinden belirli DB'leri seçebilirsiniz

### 2.3.11 Cihazları bir ağdan geçici olarak ayırma

Bireysel ağ cihazlarını alt ağdan ayırabilirsiniz. Cihazın konfigürasyonu projeden çıkarılmadığı için cihaza bağlantıyı kolaylıkla yeniden yapabilirsiniz.



Ağ cihazının ara yüz portuna sağ tıklayınız ve içerik menüsünden "Disconnect from subnet" komutunu seçiniz.

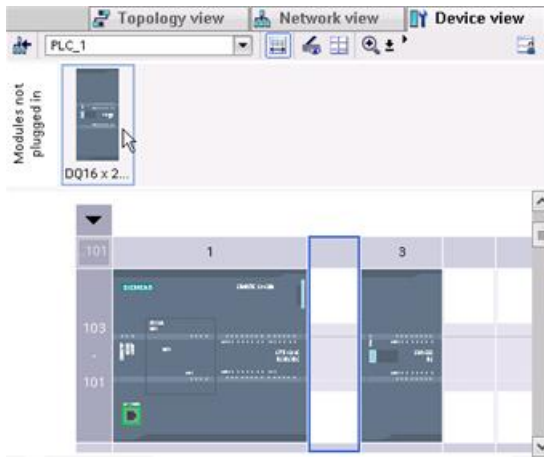
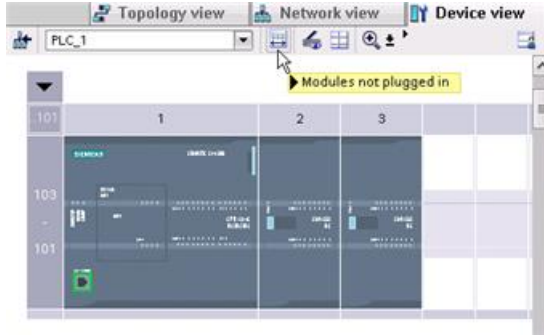
STEP 7 ağ bağlantılarının yeniden konfigüre der, ancak bağlantısı ayrılmış cihazları projeden kaldırmaz. Ağ bağlantısı silinirken ara yüz adresleri değiştirilmez.



Yeni ağ bağlantılarını yüklediğinizde CPU, STOP moduna ayarlanmalıdır.

Cihazı yeniden bağlamak için cihazın portuna basit olarak yeni bir ağ bağlantısı oluşturmanız yeterlidir.

## 2.3.12 Cihazların konfigürasyondan sanal olarak çıkarılması



STEP 7, "unplugged/takılı olmayan" modüller için bir depolama alanı sağlar. Bir modülün konfigürasyonunu kaydetmek için onu raftan çekebilirsiniz.

Bu bağlantısı kesilmiş modüller projenizle birlikte saklanır, böylece size gelecekte parametrelerini yeniden yapılandırmanıza gerek kalmaksızın modülü tekrar yerine takma olanağı sağlar.

Bu özelliğin bir kullanımı geçici bakım içindir. Yeni bir değiştirme modülü bekliyor olabileceğiniz bir senaryo düşününüz. Bu senaryoda kısa-süreli değiştirme amaçlı farklı bir modülü geçici olarak kullanmayı planlıyorsunuz. Konfigürasyonu yapılmış modülü raftan "Unplugged modules" (takılı olmayan modüller) yanına sürükleyebilir ve sonra geçici modülü takabilirsiniz.

*STEP 7 programlama yazılımı*

*2.3 Kolay kullanım araçları*

---




## Kurulum

# 3

### 3.1 S7-1200 cihazlarının kurulumu için kılavuzlar

S7-1200 ekipmanı, kurulumu kolay olacak şekilde tasarlanmıştır. Bir S7-1200'ü bir panel veya standart bir ray üzerine monte edebilir ve S7-1200'ü dikey veya yatay olarak yerleştirebilirsiniz. S7-1200'ün küçük boyutu, alanı etkin kullanımınızı sağlar.

 <b>UYARI</b>
<b>S7-1200 PLC'ler için kurulum kuralları</b>
SIMATIC S7-1200 PLC'ler, Açık Tip Kontrolörlerdir. S7-1200'ün bir muhafaza içine, kabine veya elektrik kontrol odasına kurulması gerekir. Muhafazaya, kabine veya elektrik odasına giriş, yetkili personel ile sınırlandırılmalıdır.
Bu kurulum kurallarına uyulmaması ölüm, ciddi kişisel yaralanma ve/veya mülke zarara neden olabilir.
S7-1200 PLC'leri kurarken her zaman bu kuralları takip ediniz.

#### **S7-1200 cihazlarını ısıdan, yüksek gerilimden ve elektrik gürültüsünden uzak tutunuz**

Sisteminizdeki cihazların yerlerini belirlerken genel bir kural olarak yüksek gerilim ve yüksek elektrik gürültüsü üreten cihazları daima S7-1200 gibi düşük gerilimli ve lojik tipi cihazlardan ayrı tutunuz.

S7-1200'ün panel içerisindeki yerini belirlerken panelinizdeki ısı üreten cihazları dikkate alarak elektronik-tipi cihazları panelin daha soğuk kısımlarına yerleştiriniz. Yüksek sıcaklığa sahip bir ortama maruz kalma azaltıldığında her hangi bir elektronik cihazın ömrü uzayacaktır.

Ayrıca panel içindeki cihazlar için kablo yollarına da dikkat ediniz. Düşük gerilimli sinyal iletkenlerini ve haberleşme kablolarını tepsiye yerleştirirken yüksek enerjili AC güç kabloları ve hızlı anahtarlanan DC kabloları ile aynı tepsiye yerleştirmekten kaçınınız.

Soğutma ve kablaj için yeterli boşluğu sağlayınız.

## Kurulum

### 3.1 S7-1200 cihazlarının kurulumu için kılavuzlar

S7-1200 cihazların soğutması, doğal ısı yayımına göre tasarlanmıştır. Uygun bir soğutma için cihazların altında ve üstünde en az 25 mm boşluk bırakmalısınız. Ayrıca, modüllerin önünde ve mahfazanın iç kısmında en az 25 mm derinlik bırakılmalıdır.

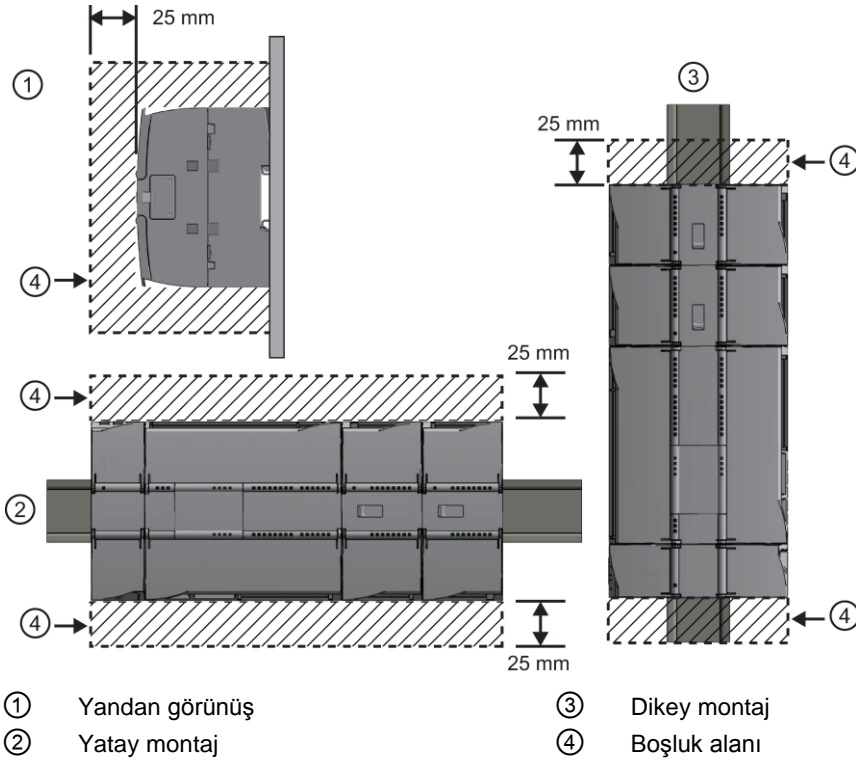
#### DİKKAT

**Dikey montajdan dolayı izin verilen maksimum ortam sıcaklığı 10 C derece azaltılır.**

Bir S7-1200 sistemini aşağıdaki şekilde gösterildiği gibi yerleştiriniz.

S7-1200 sistemi montajının doğru yapıldığından emin olunuz.

S7-1200 sisteminizin yerleşim planını yaparken kablağ ve haberleşme kablo bağlantıları için yeterli boşluğu bırakınız.



## 3.2 Güç ihtiyacı hesabı

CPU'nuzun dahili bir güç kaynağı vardır. Bu kaynak; CPU, sinyal modülleri, sinyal kartı, haberleşme modülleri ve diğer 24 VDC ile çalışan cihazlarınızın güç talebini sağlar.

CPU'nuz tarafından sağlanan 5 VDC lojik güç ihtiyacı ve sinyal modülleri, sinyal kartları ve haberleşme modülleri için gerekli olan 5 VDC gücü hakkında bilgi için teknik özelliklere (Sayfa 829) bakınız. CPU'nuz tarafından konfigürasyonunuza ne kadar güç (akım) sağlanabileceğini belirlemek için "Güç ihtiyacı hesabı" na (Sayfa 985) bakınız.

CPU'nun üzerinde; giriş noktaları, sinyal modülleri üzerindeki röle bobinleri veya diğer ihtiyaçlar için 24 VDC besleme yapabilecek 24 VDC sensör güç kaynağı bulunmaktadır. 24 VDC güç ihtiyacınız sensör güç kaynağının gücünü aşarsa, sisteminize harici bir 24 VDC güç kaynağı eklemeniz gerekir. Özel CPU'nuzda ait 24 VDC sensör güç kaynağının kapasitesi için teknik özelliklere (Sayfa 829) bakınız.

### Not

CM 1243-5 (PROFIBUS master modül), CPU üzerindeki 24 VDC sensör güç kaynağından güç çeker.

Harici bir 24 VDC güç kaynağına ihtiyacınız olursa, harici kaynak ile CPU'nun sensör güç kaynağının paralel bağlanmadığından emin olunuz. İyileştirilmiş bir elektrik gürültüsü koruması sağlamak için farklı güç kaynaklarının ortak uçlarının (M) birbirine bağlı olması tavsiye edilir.

### UYARI

Harici bir 24 VDC güç kaynağının 24 VDC sensör güç kaynağı ile paralel bağlanması durumunda, her bir kaynak kendi çıkış gerilim seviyesini gerçekleştirmeye çalışacağından dolayı iki kaynağın gerilim seviyeleri arasında uyumsuzluk oluşabilir.

Bu uyumsuzluk sonucunda iki güç kaynağından birisi veya her ikisinde ömür kısalması veya hemen arızalanma olabilir. Bu durum PLC'nin öngörülemez davranışına yol açabilir. Öngörülemez davranış ölüme, ciddi kişisel yaralanmalara ve/veya maddi zararlara yol açabilir.

DC sensör güç kaynağı ve herhangi bir harici güç kaynağı farklı noktalara güç sağlamalıdır.

S7-1200 sistemindeki 24 VDC besleme girişlerinden bazıları birden fazla M terminalini bağlayan ortak bir lojik devresiyle birbirlerine bağlantılıdır. Örneğin aşağıdaki devreler veri sayfalarında "izole edilmemiş" olarak gösterildiklerinde birbirleriyle bağlantılıdır: CPU'nun 24 VDC güç kaynağı, bir SM rölesi bobini için güç girişi veya izole edilmemiş bir analog giriş için güç kaynağı. Bütün izole edilmemiş M terminalleri aynı harici referans potansiyeline bağlanmalıdır.

 **UYARI**

**İzole edilmemiş M terminallerinin farklı referans potansiyellerine bağlanması istenmeyen akımların akmasına yol açabilir ve bunun sonucunda PLC'de ve bağlı herhangi bir ekipmanda hasar veya tahmin edilemeyecek işlemler meydana gelebilir.**

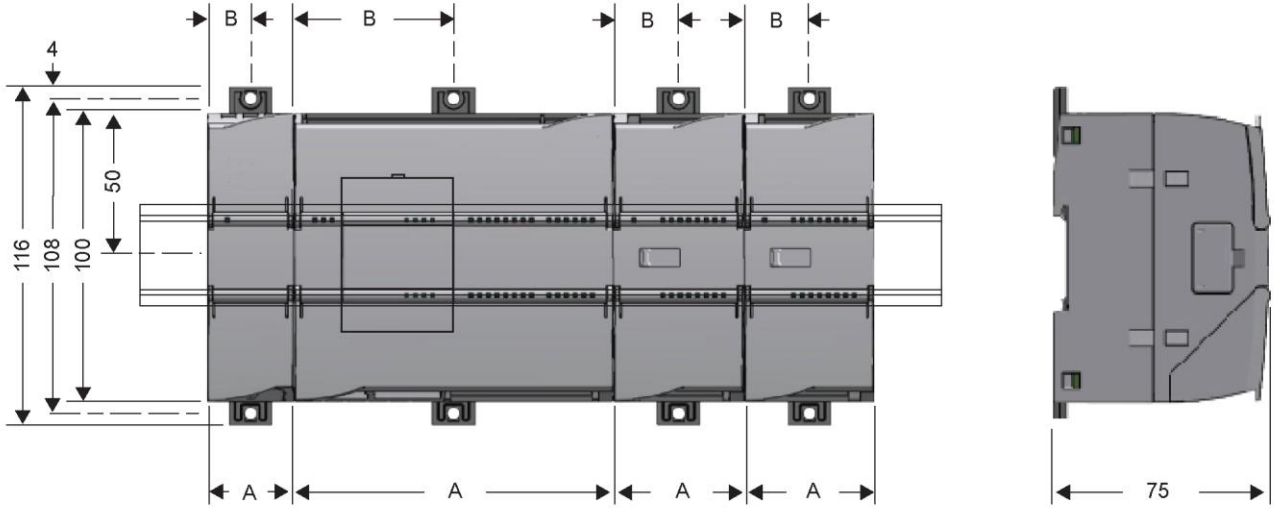
Bu esasların yerine getirilmesindeki başarısızlık ölüm veya ciddi kişisel yaralanmalar ve/veya mal zararları gibi sonuçlar doğurabilecek hasarlar veya beklenmeyen işlemlere neden olabilir.

Bir S7-1200 sistemindeki bütün izole edilmemiş M terminallerinin aynı referans potansiyeline bağlı olduklarından daima emin olunuz.

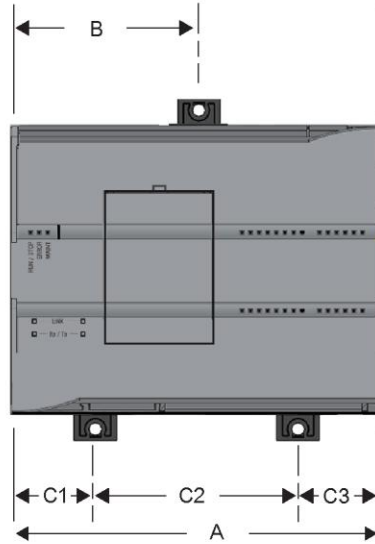
### 3.3 Montaj ve sökme işlemleri

#### 3.3.1 S7-1200 cihazları için montaj boyutları

CPU 1211C, CPU 1212C, CPU 1214C  
(measurements in mm)



CPU 1215C, CPU 1217C



## Kurulum

### 3.3 Montaj ve sökme işlemleri

Çizelge 3- 1 Montaj boyutları (mm)

S7-1200 Cihazları		Genişlik A (mm)	Genişlik B (mm)	Genişlik C (mm)
CPU	CPU 1211C ve CPU 1212C	90	45	--
	CPU 1214C	110	55	--
	CPU 1215C	130	65 (top)	Alt: C1: 32.5 C2: 65 C3: 32.5
	CPU 1217C	150	75	Alt: C1: 37.5 C2: 75 C3: 37.5
Sinyal modülleri	Dijital 8 ve 16 nokta Analog 2, 4, ve 8 nokta Termokupl 4 ve 8 nokta RTD 4 nokta SM 1278 IO Link-Master	45	22.5	--
	Dijital DQ 8 x Röle (kademe değiştirme)	70	35	--
	Analog 16 nokta RTD 8 nokta	70	35	--
Haberleşme ara yüzleri	CM 1241 RS232, ve CM 1241 RS422/485 CM 1243-5 PROFIBUS master and CM 1242-5 PROFIBUS slave CM 1242-2 AS-i Master CP 1242-7 GPRS	30	15	--
	TS (Teleservice) Adapter IE Basic <sup>1</sup>			
	TS Adaptör TS Modül	30 30	15 15	-- --

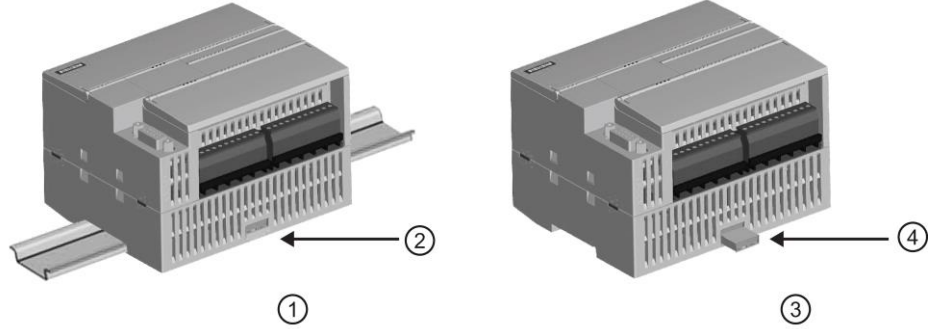
<sup>1</sup> TS (Teleservice) Adapter IE Basic'i montaj etmeden önce ilk olarak TS Adapter ve bir TS modülünü bağlamalısınız. Toplam genişlik (genişlik A) 60 mm'dir.

Her bir CPU, SM, CM ve CP bir DIN rayına veya bir panele montaja uygundur. Cihazı raya güvenle tutturmak için cihaz üzerindeki DIN rayı klipslerini kullanınız. Ayrıca bu klipsler, üniteyi doğrudan bir panele monte etmek için vida montaj yerlerini göstermek üzere açılan yuvaya otururlar. Cihaz üzerindeki DIN klipsleri için deliğin iç boyutu 4.3 mm'dir.

Serbest hava dolaşımı sağlamak amacıyla ünitenin üstünde ve altında 25 mm ısı bölge sağlanmalıdır.

### S7-1200 cihazlarının montajı ve sökülmesi

CPU montajı, standart bir DIN rayı veya bir panel üzerine kolayca yapılabilir. Cihazı DIN rayının üzerine tutturmak için DIN rayı klipsleri sağlanmaktadır. Ayrıca bu klipsler, panel montaj ünitesi için bir vida montaj yerini temin etmek üzere açılan yuvaya otururlar.



- ① DIN ray montajı  
② DIN ray klipsi, kilitli konumda  
③ Panel montajı  
④ Panel montajı için klips açılmış konumda

Herhangi bir elektrikli cihazı monte etmeden veya sökmeden önce, bu ekipmanı besleyen elektriğin kesildiğinden emin olunuz. Ayrıca, ilgili diğer ekipmanı besleyen elektriğin de kesildiğinden emin olunuz.

#### UYARI

**Elektrik beslemesi varken S7-1200'ün veya ilgili ekipmanın monte edilmesi veya sökülmesi, elektrik şokuna veya ekipmanın beklenmeyen işleyişine neden olabilir.**

Montaj veya sökme işlemleri sırasında S7-1200'ü ve ilgili ekipmanı besleyen bütün elektrik gücünün kesilmemesi durumunda, elektrik şoku veya beklenmeyen ekipman işleyişi yüzünden ölüm, ciddi kişisel yaralanmalar ve/veya mal zararı meydana gelebilir.

S7 CPU'lar veya ilgili ekipmanın montajına veya sökülmesine teşebbüs etmeden önce daima uygun emniyet önlemlerine uyunuz ve S7-1200'ü besleyen elektrik gücünün kesildiğinden emin olunuz.

Her ne zaman bir S7-1200 cihazını değiştirdiyseniz veya monte ederseniz daima doğru modülü veya eşdeğeri cihazı kullandığınızdan emin olunuz.

#### UYARI

**Bir S7-1200 modülünün yanlış montajı, S7-1200 içerisindeki programın tahmin edilemeyen fonksiyon görmesine yol açabilir.**

Bir S7-1200 cihazının aynı model, yön veya sırası ile değiştirilmemesi, beklenmedik ekipman işleyişine ve dolayısıyla ölüm, ciddi kişisel yaralanmalar ve/veya mal hasarına neden olabilir.

Bir S7-1200 cihazını mutlaka aynı model ile değiştiriniz ve cihazın yönünün ve konumunun doğru olduğundan emin olunuz.

**⚠ UYARI**

**Ekipmanı, alevlenebilir veya tutuşabilir bir ortamda sökmeyiniz.**

Alevlenebilir veya tutuşabilir atmosferin bulunduğu yerde ekipmanın sökülmesi yangına veya patlamaya neden olabilir ve bunun sonucunda ölüm, ciddi yaralanma ve/veya mala zarar meydana gelebilir.

Yanıcı ve tutuşucu bir ortamın bulunması durumunda daima uygun güvenlik önlemlerine uyunuz.

**Not**

**Elektrostatik deşarj cihaz veya CPU üzerindeki bağlantı uçlarına zarar verebilir.**

Her ne zaman bir cihaz üzerinde çalışacaksanız, topraklanmış iletken pad ile temas yapınız ve/veya topraklanmış bir bileklik takınız.

**3.3.2 CPU'nun montajı ve sökülmesi**

CPU'yu bir panel veya bir DIN ray üzerine monta edebilirsiniz.

**Not**

Herhangi bir haberleşme modülünü CPU'ya takınız ve bir ünite olarak monte ediniz. CPU monte edildikten sonra sinyal modüllerini tek tek takınız.

Üniteleri DIN rayı veya bir panel üzerine monte ederken aşağıdakilere dikkat ediniz:

- DIN ray montajını yapmak için üst DIN ray klipsinin yerine (içe) kilitletiğinden ve alt DIN ray klipsinin CPU ve eklenmiş CM'ler için çıkık konumda olduğundan emin olunuz.
- DIN rayı üzerine cihazların montajını yaptıktan sonra, cihazları DIN rayı üzerine kilitlemek için alt DIN ray klipslerini kilitlemiş konuma doğru hareket ettiriniz.
- Panel montajı için DIN rayı klipslerinin genişletilmiş konuma itildiğinden emin olunuz.



CPU'yu bir panel üzerine monte etmek için şu adımları takip ediniz:

1. Çizelgede mm biriminde (Sayfa 53) gösterilen boyutları kullanarak montaj deliklerinin (M4) yerlerini belirleyiniz, deliniz ve tıkaçı ile kapatınız.
2. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinin kesildiğinden emin olunuz
3. Montaj klipslerini modülden uzatınız. CPU'nun üstündeki ve altındaki DIN rayı klipslerinin uzatılmış konumda olduğundan emin olunuz.
4. Yay ve düz rondela ile birlikte bir sac vidası M4 kullanarak modülü panele bağlayınız. Düz başlı bir vida kullanmayınız.



**Not**

Vidanın tipi üzerine monte edilecek malzeme tarafından belirlenir. Sac vidası düz duruma gelinceye kadar uygun torku uygulamalısınız. Vida montajında aşırı döndürme kuvveti uygulamaktan kaçınınız. Düz başlı bir vida kullanmayınız.

**Not**

CPU'nuz yüksek titreşim potansiyeli olan bir ortamda bulunuyorsa veya CPU dikey olarak monte edilmişse, DIN ray darbe emicilerin (stops) kullanılması yardımcı olabilecektir. S7-1200 için panel montajının kullanılması daha yüksek bir koruma seviyesi sağlayacaktır. Modüllerin bağlantılı kalmasından emin olmak için bir DIN rayı üzerinde bir sonlandırıcı destek (8WA1 808 veya 8WA1 805) kullanınız. Sisteminiz yüksek titreşimli bir ortamda ise CPU panel montajı yüksek seviyede gürültü koruması sağlayacaktır.


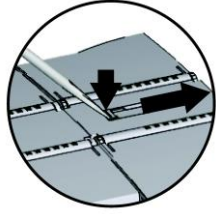

Çizelge 3- 2 CPU'nun bir DIN rayı üzerine montajı

Görev	İşlem
	<ol style="list-style-type: none"><li>1. DIN rayını takınız. Rayı montaj paneline 75 mm aralıklarla bağlayınız.</li><li>2. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</li><li>3. CPU'yu DIN rayının üstüne kancasını geçiriniz.</li><li>4. Ray üzerine geçmesine sağlamak için CPU'nun altındaki DIN rayı klipsini çekiniz.</li><li>5. CPU'yu ray üzerindeki konumuna doğru aşağıya döndürünüz.</li><li>6. CPU'yu raya kilitlemek için klipsleri içeri doğru itiniz.</li></ol>
	

## Kurulum

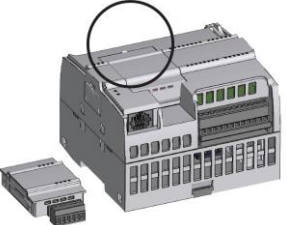
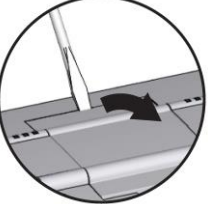
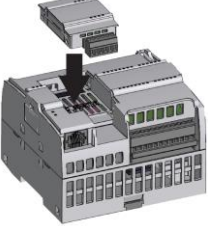
### 3.3 Montaj ve sökme işlemleri

Çizelge 3-3 CPU'yu bir DIN rayından çıkarmak


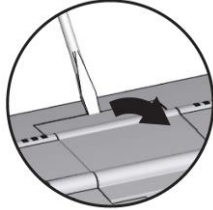
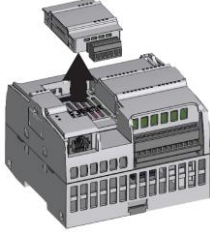
Görev	İşlem
	
	<p>7. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>8. I/O konektörlerini, kabloları ve CPU'dan kabloları çıkarınız (Sayfa 63).</p> <p>9. CPU'yu çıkarınız ve takılı tüm haberleşme modüllerini bir ünite olarak çıkarınız. Bütün sinyal modülleri takılı kalmalıdır.</p> <p>10. Bir SM, CPU'ya bağlı ise bus konektörünü geri çekiniz:</p> <ul style="list-style-type: none"> <li>- Sinyal modülünün üstündeki sekmenin yanına bir tornavida yerleştiriniz.</li> <li>- Konektörü CPU'dan ayırmak için aşağı doğru bastırınız.</li> <li>- Sekmeyi sağa doğru tamamen kaydırınız.</li> </ul> <p>11. CPU'yu çıkarınız:</p> <ul style="list-style-type: none"> <li>- CPU'yu raydan ayırmak için DIN ray klipsini çekiniz.</li> <li>- CPU'yu yukarı ve dışa doğru döndürünüz ve sistemden çıkarınız.</li> </ul>

### 3.3.3 Bir SB, CB veya BB'nin takılması ve çıkarılması

Çizelge 3-4 Bir SB, CB veya BB 1297'nin takılması

Görev	İşlem
	<p>12. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>13. Üst ve alt terminal blok kapaklarını CPU'dan sökünüz.</p> <p>14. Kapağın arkasındaki CPU'nun üstündeki yuvaya bir tornavida yerleştiriniz.</p> <p>15. Kapağı nazikçe kaldırınız ve CPU'dan çıkarınız.</p> <p>16. Modülü, CPU'nun üstündeki montaj yerine düzgün bir şekilde aşağı doğru yerleştiriniz.</p> <p>17. Modülü, yerine geçinceye kadar sıkıca konumuna doğru bastırınız.</p> <p>18. Terminal blok kapaklarını yerine takınız.</p>
	

Çizelge 3- 5 Bir SB, CB veya BB 1297'nin sökülmesi

Görev		İşlem
		19. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz. 20. Üst ve alt terminal blok kapaklarını CPU'dan sökünüz. 21. Kapağın arkasındaki CPU'nun üstündeki yuvaya bir tornavida yerleştiriniz. 22. Kapağı nazikçe kaldırınız ve CPU'dan çıkarınız. 23. Modülü, CPU'nun üstündeki montaj yerinden düzgün bir şekilde yukarı doğru çıkarınız. 24. Modülü, CPU üzerine takınız. 25. Terminal blok kapaklarını yerine takınız.
		

### BB 1297 batarya kartının montajı veya değiştirilmesi

BB 1297, CR1025 tipi batarya gerektirir. BB 1297 ile birlikte batarya sağlanmamaktadır ve batarya satın alınmalıdır. Bataryanın takılması veya değiştirilmesi için şu adımlar izlenir:


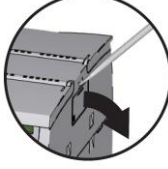
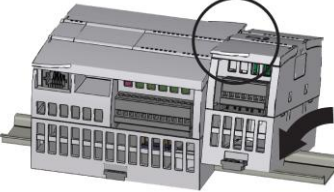

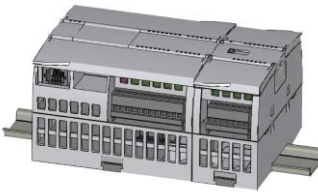
1. BB 1297'de, yeni bataryanın pozitif tarafı yukarıda olacak ve negatif tarafı baskı devre kartı yanına gelecek şekilde yerleştiriniz.
2. BB 1297, CPU'ya takılmak için hazırdır. BB 1297'yi takmak için yukarıdaki takma talimatlarını izleyiniz.

BB 1297'de bataryayı değiştirmek için:

3. CPU'dan BB 1297'yi sökmek için yukarıdaki sökme talimatlarını izleyiniz.
4. Eski bataryaları küçük bir tornavida kullanarak dikkatlice çıkarınız. Klip altından bataryayı dışarı itiniz.
5. Yeni bir CR1025 değiştirme bataryasını bataryanın pozitif tarafı yukarıya ve negatif tarafı baskı devre kartı yanına gelecek şekilde takınız.
6. BB 1297 batarya kartını yukarıdaki takma talimatlarını izleyerek yeniden takınız.

### 3.3.4 Bir SM'in takılması ve sökülmesi

Çizelge 3- 6 Bir SM'nin takılması

Görev	İşlem
	 <p>CPU'yu taktıktan sonra SM'nizi takınız.</p> <p>26. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>27. CPU'nun sağ tarafından kapağı Konektör için sökünüz:</p> <ul style="list-style-type: none"> <li>- Tornavidayı kapak üzerindeki oluğa sokunuz.</li> <li>- Kapağı üst kısımdan nazikçe kaldırın ve kapağı sökünüz.</li> </ul> <p>28. Yeniden kullanım için kapağı muhafaza ediniz.</p>
	 <p>SM'yi CPU'ya bağlayınız:</p> <p>29. SM'yi CPU yanına yerleştiriniz.</p> <p>30. SM'yi DIN rayın tepe üstüne geçiriniz.</p> <p>31. SM'nin rayın üzerine geçmesini sağlamak için DIN ray klipsinin altından çekiniz.</p> <p>32. SM'yi, CPU'nun yanındaki yerine aşağı doğru döndürünüz ve SM'yi raya kenetlemek için alt klipsi itiniz.</p>
	<p>Veri yolu konektörünün uzatılması, SM için hem mekanik hem de elektriksel bağlantıları yapar.</p> <p>33. Bir tornavidayı SM'nin tepesindeki sekmenin yanına yerleştiriniz.</p> <p>34. Veriyolu konektörünü CPU'ya doğru uzatmak için sekmeyi sola doğru tamamen kaydırınız. Bir sinyal modülünü bir sinyal modülüne takmak için aynı işlemi izleyiniz.</p>

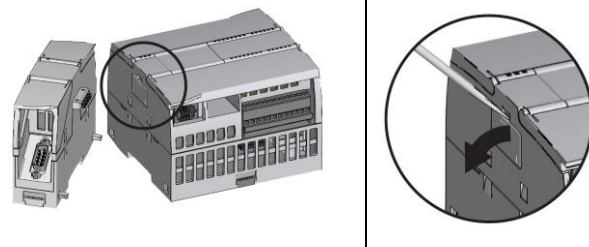
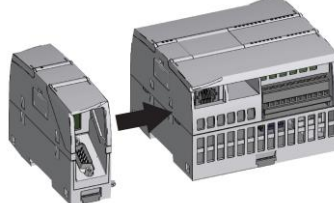
Çizelge 3- 7 Bir SM'nin sökülmesi

Görev	İşlem
	<p>Yerindeki CPU'yu veya diğer bir SM'yi sökmeksizin her hangi bir SM'yi sökebilirsiniz.</p> <p>35. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>36. I/O konektörlerini ve kabloları SM'den çıkarınız (Sayfa 63).</p> <p>37. Bus konektörünü geri çekiniz.</p> <ul style="list-style-type: none"><li>- Tornavidayı SM'nin tepesi üstündeki sekmenin yanına yerleştiriniz.</li><li>- Konektörü CPU'dan ayırmak için aşağıya basınız.</li><li>- Sekmeyi sağa doğru tamamen kaydırınız.</li></ul> <p>Sağa kaydırılacak başka SM var ise bu işlemi o SM için tekrarlayınız.</p>
	
	<p>SM'i sökünüz:</p> <p>38. SM'yi raydan kurtarmak için alt DIN ray klipsini çekiniz.</p> <p>39. SM'yi yukarı doğru çeviriniz ve raydan çıkarınız. SM'yi sistemden çıkarınız.</p> <p>40. Gerekirse, kirlenmeyi önlemek için CPU üzerindeki bus konektörünü örtünüz. Bir sinyal modülünü bir sinyal modülünden çıkarmak için aynı işlemi izleyiniz.</p>

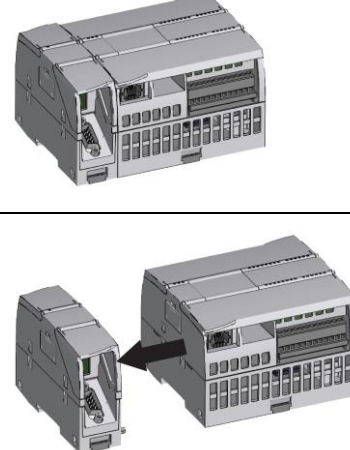
### 3.3.5 CM ve CP'nin takılması ve sökülmesi

Herhangi bir haberleşme modülünü CPU'ya takınız ve bir ünite olarak montajını yapınız, CPU'nun montajı ve sökülmesi bölümünde (Sayfa 56) gösterildiği gibi.

Çizelge 3- 8 CM veya CP'nin takılması

Görev	İşlem
	<p>41. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>42. CPU'nun sol tarafından bus kapağını sökünüz:</p> <ul style="list-style-type: none"> <li>- Tornavidayı bus kapağı üzerindeki oluğa sokunuz.</li> <li>- Kapağı üst kısmından nazikçe kaldırınız.</li> </ul> <p>43. Bus kapağını sökünüz. Yeniden kullanım için kapağı muhafaza ediniz.</p>
	<p>44. CM veya CP'yi CPU'ya bağlayınız:</p> <ul style="list-style-type: none"> <li>- Veri yolu konektörünü ve CM'nin noktalarını CPU'nun delikleriyle aynı hizaya getiriniz.</li> <li>- Postlar yerine geçene kadar üniteleri birlikte sıkıca bastırınız.</li> </ul> <p>45. CPU ve CP'yi bir DIN rayı veya panel üzerine takınız.</p>

Çizelge 3- 9 Bir CM veya CP'nin sökülmesi

Görev	İşlem
	<p>CPU ve CM'yi DIN rayından veya panelden bir ünite olarak çıkarınız.</p> <p>46. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>47. I/O konektörlerini ve CPU ve CM'den bütün kabloları ve kabloları çıkarınız.</p> <p>48. DIN rayı montajı için CPU ve CM'ler üzerindeki en alt DIN ray klipslerini uzatılmış konuma taşıyınız.</p> <p>49. CPU ve CM'leri DIN rayından veya panelden çıkarınız.</p> <p>50. CPU ve CM'leri sıkıca kavrayınız ve çekip ayırınız.</p>

**NOT**

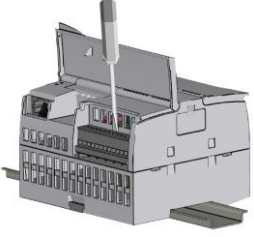
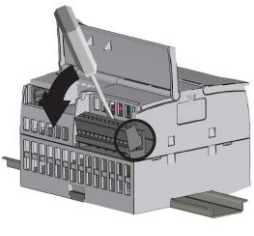
**Bir alet kullanmaksızın modüllerin ayrılması.**

Modülleri ayırmak için bir alet kullanmayınız. Çünkü bu işlem ünitelere zarar verebilir.

### 3.3.6 S7-1200 terminal blok konektörün çıkarılması ve yeniden takılması

CPU, SB ve SM modülleri, kabloların bağlanmasını kolaylaştırmak için çıkarılabilir konektörler sağlarlar.

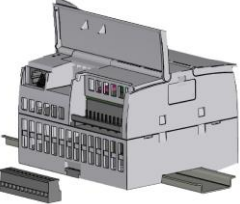
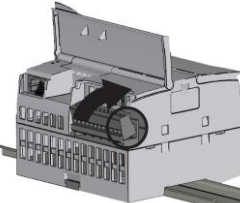
Çizelge 3- 10 Konektörün çıkarılması

Görev	İşlem
	<p>CPU'nun elektrik beslemesinin ayrılması ve konektörün üzerindeki kapağın açılmasıyla terminal bloğu konektörünün çıkarılması için sistemi hazırlayınız.</p> <ol style="list-style-type: none"><li>51. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</li><li>52. Konektörün üst kısmını kontrol ediniz ve yuvayı tornavidanın ucu için konumlandırınız.</li><li>53. Yuva içine bir tornavida sokunuz.</li><li>54. Konektörün üst kısmını CPU'dan nazikçe kaldırınız. Konektör çıt sesiyle birlikte ayrılacaktır.</li><li>55. Konektörü sıkıca tutunuz ve CPU'dan çıkarınız.</li></ol>
	

## Kurulum

### 3.3 Montaj ve sökme işlemleri

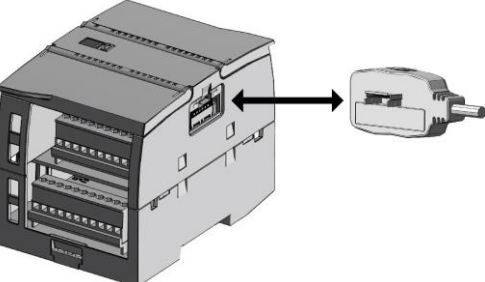
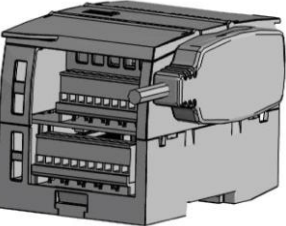
Çizelge 3- 11 Konektörün takılması

Görev	İşlem
	<p>CPU'nun elektrik beslemesinin ayırarak ve konektörün üzerindeki kapağı açarak terminal bloğunun takılması için elemanları hazırlayınız.</p> <p>56. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>57. Konektörü ünitenin üzerindeki pinlerle hizalayınız.</p> <p>58. Konektörün kablo kenarını, Konektör tabanı kenarının içi ile hizalayınız.</p> <p>59. Konektörü, yerine geçinceye kadar sıkıca aşağı doğru bastırınız ve döndürünüz.</p>
	<p>Konektörün uygun bir şekilde hizalandığından ve tamamen yerine geçtiğinden emin olmak için dikkatlice kontrol ediniz.</p>

### 3.3.7 Genişletme kablosunun takılması ve çıkarılması

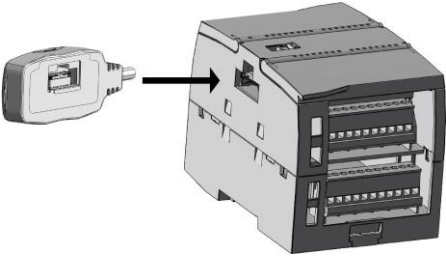
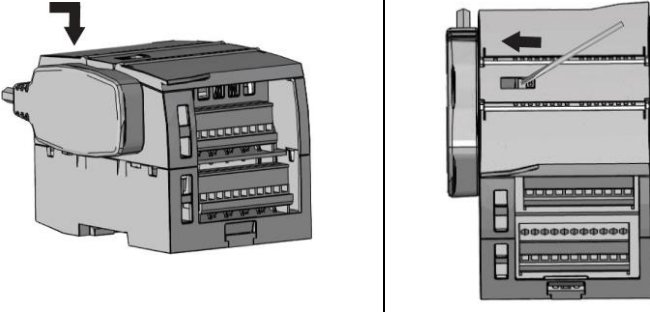
S7-1200 genişletme kablosu, S7-1200 sisteminizin yerleşim konfigürasyonunu yaparken size ilave esneklik sağlar. Her bir CPU sistemi için sadece bir genişletme kablosuna izin verilir. Genişletme kablosunu CPU ile ilk SM arasına veya herhangi iki SM arasına takabilirsiniz.

Çizelge 3- 12 Genişletme kablosu erkek konektörünün takılması ve çıkartılması

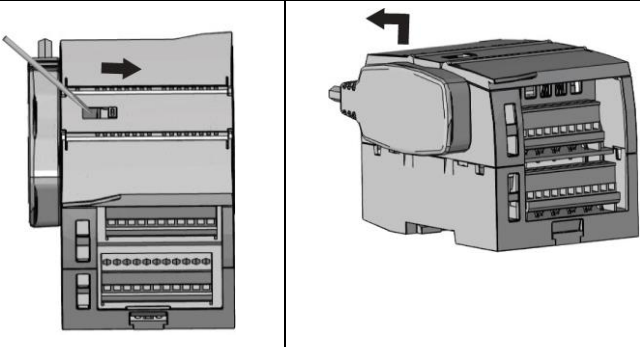
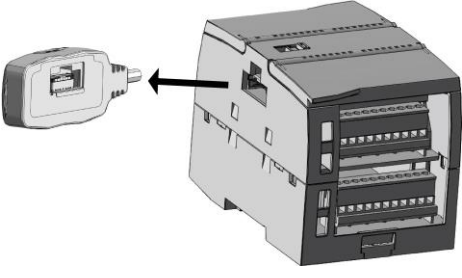
Görev	İşlem
	<p>Erkek konektörü takmak için:</p> <p>60. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>61. Konektörü, CPU veya sinyal modülünün sağ tarafındaki bus konektörüne doğru itiniz.</p> <p>Erkek konektörü çıkartmak için:</p> <p>62. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>63. Erkek konektörü sinyal modülünden veya CPU'dan ayırmak için çekiniz.</p>
	



Çizelge 3- 13 Genişletme kablosu dışı konektörünün takılması

Görev	İşlem
	<p>64. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>65. Dışı konektörü sinyal modülünün sol tarafındaki bus konektörüne yerleştiriniz.</p> <p>66. Dışı konektörün kanca uzantısını bus konektörünün yuvasına doğru kaydırın ve kancayı geçirmek için hafifçe aşağı doğru bastırınız.</p> <p>67. Konektörü yerine kilitleyiniz:</p> <ul style="list-style-type: none"> <li>- Sinyal modülünün üstündeki sekmenin yanına bir tornavida yerleştiriniz.</li> <li>- Sekmeyi sola doğru tamamen kaydırınız.</li> </ul> <p>Konektörü geçirmek için konektör sekmesini sola doğru sonuna kadar kaydırmalısınız. Konektör sekmesi yerine kiletlenmiş olmalıdır.</p>
	

Çizelge 3- 14 Genişletme kablosu dışı konektörünün çıkarılması

Görev	İşlem
	<p>68. CPU ve bütün S7-1200 ekipmanının elektrik beslemesinden ayrıldığından emin olunuz.</p> <p>69. Konektör kilidini açınız:</p> <ul style="list-style-type: none"> <li>- Sinyal modülünün üstündeki sekmenin yanına bir tornavida yerleştiriniz.</li> <li>- Sekmeyi hafifçe aşağı doğru bastırınız ve sağa doğru tamamen kaydırınız.</li> </ul> <p>70. Konektörü hafifçe kaldırınız ve kanca çıkıntısını çıkarınız.</p> <p>71. Dışı konektörü çıkarınız.</p>
	

**Not****Genişleme kablosunun titreşimli bir çevreye takılması**

Genişleme kablosu hareketli modüllere takılırsa veya sıkıca sabitlenmez ise kablonun konektöre erkek geçmesi zamanla gevşeyebilir. İlave gerilme boşalması sağlamak için DIN rayı üzerine erkek kablo ucunu sabitlemek üzere bir kablo bağlayıcısı kullanınız.

Takma esnasında kabloyu çekerken aşırı kuvvet kullanmaktan kaçınınız. Takma işlemi tamamlandığında kablo-modül bağlantısının doğru konumda olduğundan emin olunuz.

---

**3.3.8 TS (TeleService) adaptör****3.3.8.1 TeleService Adapter'in bağlanması**

TS (TeleService) Adapter IE Basic bağlantısını yapmadan önce, ilk olarak TS Adapter ile bir TS modülünü bağlamalısınız.

Mevcut TS modülleri:

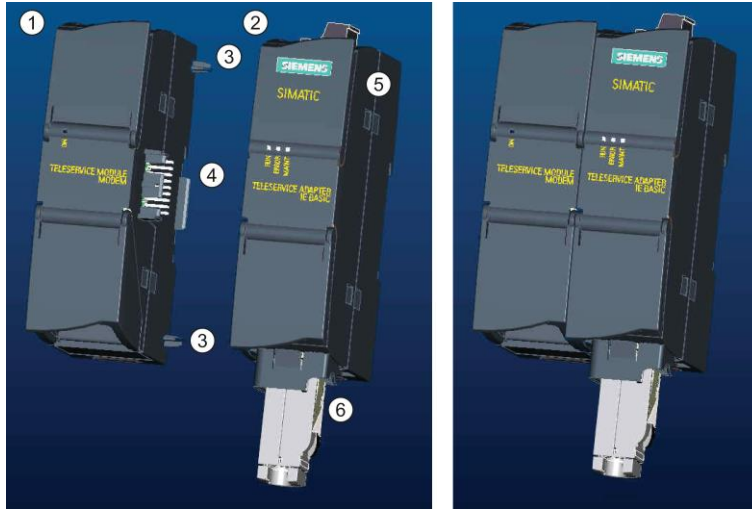
- TS modül RS232
- TS modül Modem
- TS modül GSM
- TS modül ISDN

**Not**

**TS modülünün fiş konektörünün kontaklarına ④ dokunursanız, TS Adaptör modülü zarar görebilir.**

Elektrostatik deşarjdan dolayı TS modüle bir zarar gelmesinden kaçınmak için ESD kılavuzunu takip ediniz. Bir TS modülü ile TS Adaptörünü bağlamadan önce her ikisinin de boş durumda olduğundan emin olunuz.

---



- |   |            |   |                             |
|---|------------|---|-----------------------------|
| ① | TS modül   | ④ | TS modülünden fiş konektörü |
| ② | TS Adapter | ⑤ | Açılmaz                     |
| ③ | Elemanlar  | ⑥ | Ethernet port               |

**Not**

TS modülü ve TS adaptör temel ünitesini bağlamadan önce, kontak pinlerinin ④ bükülü olmadığından emin olunuz.

Bağlantı yaparken erkek konektör ile kılavuz elemanlarının düzgün bir şekilde konumlandırıldığından emin olunuz.

TS Adaptörüne sadece bir TS modülü bağlanabilir. Bir TS Adaptörü bağlantısını, S7-1200 CPU gibi farklı bir cihaza bağlamaya zorlamayınız. Konektörün mekanik yapısını değiştirmeyiniz ve kılavuz elemanlarını çıkarmayınız veya bunlara zarar vermeyiniz.

**3.3.8.2****SIM Kartının takılması**

SIM kartını TS Modülü GSM'nin alt yanındaki kart yuvasına yerleştiriniz.

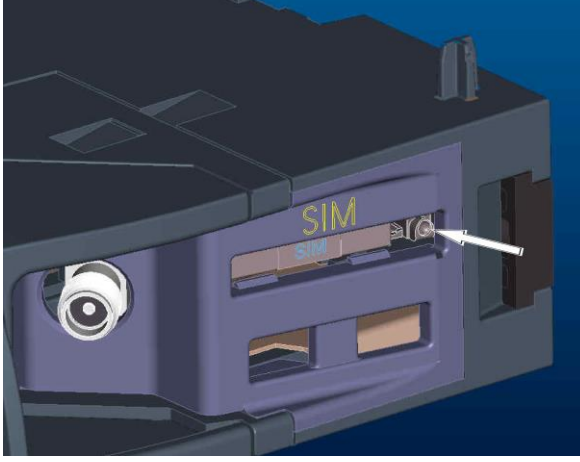
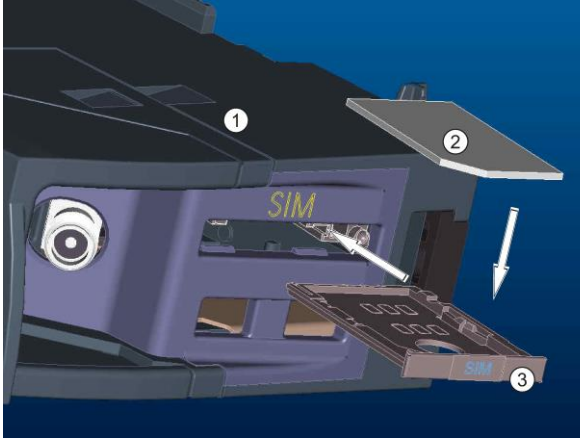
**Not**

TS Modülü GSM'nin enerjisi kesildikten sonra SIM kartı ancak takılabilir veya çıkarılabilir.

## Kurulum

### 3.3 Montaj ve sökme işlemleri

Çizelge 3- 15 SIM kartının takılması

İşlem	Görev						
SIM kart yuvasının çıkartma düğmesine basmak için sivri bir nesne kullanınız (ok yönünde) ve SIM kart yuvasını çıkarınız.							
SIM kartı SIM kart yuvasına, gösterildiği gibi yerleştiriniz ve SIM kart yuvasını yerine doğru geriye itiniz.	 <table border="1" data-bbox="1106 882 1442 1348"><tbody><tr><td>①</td><td>TS Modül GSM</td></tr><tr><td>②</td><td>SIM kart</td></tr><tr><td>③</td><td>SIM kart yuvası</td></tr></tbody></table>	①	TS Modül GSM	②	SIM kart	③	SIM kart yuvası
①	TS Modül GSM						
②	SIM kart						
③	SIM kart yuvası						

#### Not

SIM kartı yuvasının kart yuvası içinde doğru olarak yönlendirildiğinden emin olunuz. Aksi durumda, SIM kart modülle bağlantı yapamaz ve çıkartma düğmesi kart yuvasını yerinden çıkartamayabilir.

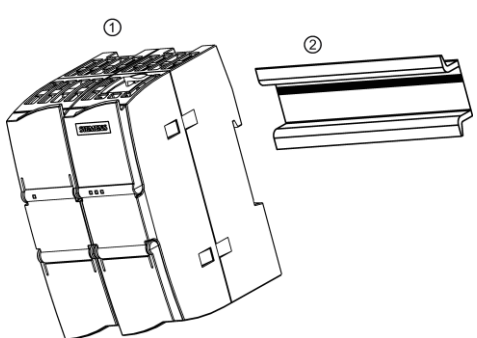
### 3.3.8.3 TS adaptör ünitesinin bir DIN rayı üzerine montajı

Ön koşullar: TS Adaptörünü ve bir TS modülünü birlikte bağlamış olmalısınız ve DIN rayı kurulmuş olmalıdır.

#### Not

TS ünitesini dikey olarak veya yüksek titreşimli bir ortamda monte ederseniz, TS modülü TS Adaptöründen bağlantısı kesilebilir. Modüllerin birbiriyle bağlı kalmasından emin olmak için DIN rayı üzerinde bir 8WA1 808 uç desteği kullanınız.

Çizelge 3- 16 TS Adaptörünün montajı ve sökülmesi

Görev	İşlem
	<p>Montaj:</p> <p>72. TS modülü ile birleşik TS Adaptörünü ①, DIN rayı ② üzerine geçiriniz.</p> <p>73. Üniteyi geçinceye kadar geriye doğru döndürünüz</p> <p>74. Her bir modülü raya tutturmak için her bir modül üzerindeki DIN rayı klipsini itiniz.</p>
	<p>Sökme:</p> <p>75. Analog kablosu ile Ethernet kablosunu TS adaptörünün alt yanından çıkarınız.</p> <p>76. TS Adaptöründen elektrik beslemesini çıkarınız.</p> <p>77. Her iki modül üzerindeki ray klipslerini ayırmak için bir tornavida kullanınız.</p> <p>78. Üniteyi DIN rayından çıkarmak için üniteyi yukarı doğru döndürünüz.</p>

#### ⚠ UYARI

#### TS Adaptörünün montajı ve sökülmesi için güvenlik kuralları

Ünitenin elektrik beslemesini sökmeden önce, analog kablosunu ve Ethernet kablosunu çıkararak TS Adaptörünün topraklama bağlantısını kesiniz. Bu önlemlerin yerine getirilmemesi yüzünden beklenmeyen ekipman çalışmaları sonucunda ölüm, ciddi kişisel yaralanmalar ve/veya mal zararları meydana gelebilir.

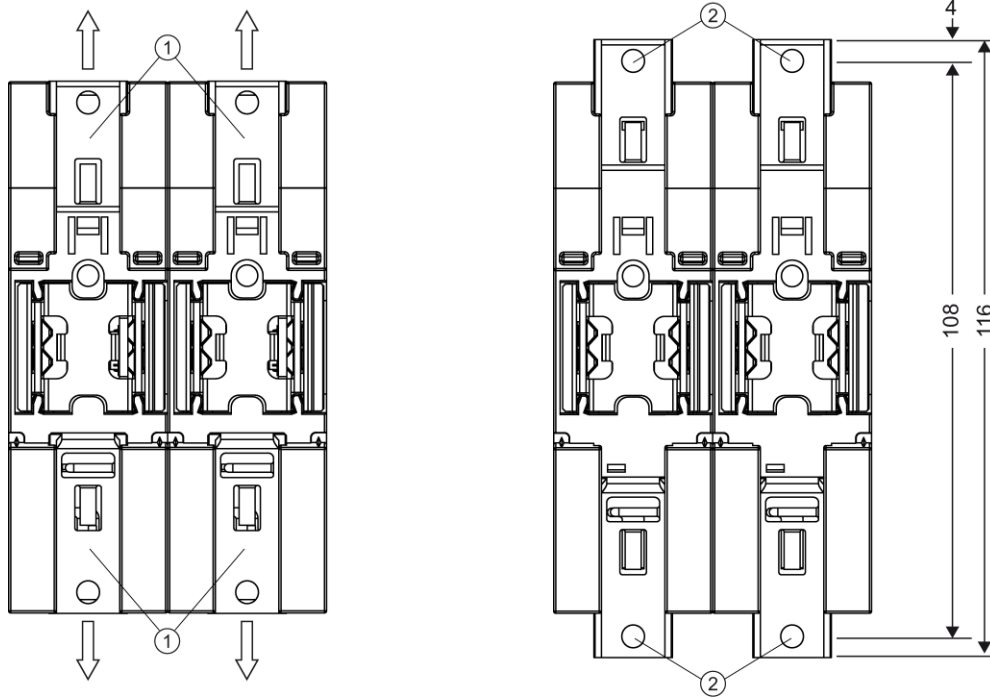
TS Adaptörü monte ederken veya sökerken daima bu kurallara uyunuz.

### 3.3.8.4 TS adaptörünün bir panel üzerine montajı

Ön koşullar: TS Adaptörünü ve bir TS modülünü birlikte bağlamış olmalısınız

- Tutturma kaydırıcısını ok yönünde TS Adaptörü ve TS modülünün arka tarafına doğru yerine geçinceye kadar hareket ettiriniz.
- TS Adaptörünü ve TS modülünü, belirlenmiş montaj duvarında ② ile işaretlenmiş konuma vidalayınız.

Aşağıdaki çizim, TS Adaptörünü her iki konumdaki tutturma kaydırıcısı ① ile birlikte arkadan göstermektedir:



- ① Tutturma kaydırıcısı  
② Duvara montaj için delme delikleri

### 3.4 Kablo kılavuzları

Bütün elektrikli ekipmanların düzgün kablo ve topraklanması, sisteminizin optimum çalışmasını ve uygulamanız ve S7-1200 için ilave elektrik gürültüsü koruması sağlamayı temin etmeye yardımcı olmada önemlidir. S7-1200 kablo diyagramları için teknik özelliklere bakınız (Sayfa 829).

#### Ön koşullar

Herhangi bir elektrikli cihazın kablolarını ve topraklamasını bağlamadan önce, bu ekipmana gelen elektrik beslemesinin kesilmiş olduğundan emin olunuz. Ayrıca, ilgili her hangi bir ekipmana gelen elektrik beslemesinin de kesilmiş olmasından emin olunuz.

S7-1200 ve ilgili ekipmanların kablajını yaparken yürürlükteki tüm elektrik kurallarına uyduğunuzdan emin olunuz. Bütün ekipmanları yürürlükteki tüm ulusal ve yerel standartlara göre monte ediniz ve çalıştırınız. Özel durumunuz için hangi kuralların ve standartların uygulanabilir olduğunu belirlemek için yerel yetkililerle temasa geçiniz.

 **UYARI**

**S7-1200'ün veya ilgili ekipmanların elektrik beslemesi varken takılması veya kablajı, elektrik şokuna veya ekipmanda beklenmeyen işleyişlere neden olabilir.**

Montaj veya sökme işlemleri sırasında S7-1200'e ve ilgili ekipmanlara gelen gücün tamamen kesilmemesi durumunda, elektrik şokuna veya beklenmeyen ekipman işleyişine bağlı olarak ölüm, ciddi kişisel yaralanmalar ve/veya zararlar meydana gelebilir.

Daima ilgili emniyet önlemlerine uyunuz ve S7-1200 veya ilgili ekipmanı montaja veya sökmeye başlamadan önce S7-1200'ün elektrik beslemesinin kesildiğinden emin olunuz.

S7-1200 sisteminizin topraklaması ve kablajını tasarlarken daima emniyet konusunu dikkate alınız. S7-1200 gibi elektronik kontrol cihazları arızalanabilir ve kontrol edilmekte veya izlenmekte olan ekipmanın beklenmedik işleyişine neden olabilir. Bu sebepten, olası kişisel yaralanmalara veya ekipman hasarını karşı korunmak için S7-1200'den bağımsız olan güvenlik önlemlerini almalısınız.

 **UYARI**

**Kontrol cihazları güvenli olmayan bir ortamda arızalanırlar ve kontrollü ekipmanın beklenmedik işleyişine neden olabilirler.**

Böyle beklenmedik işleyişler sonucunda ölüm, ciddi kişisel yaralanmalar ve/veya mal zararı meydana gelebilir.

Bir acil durum durdurma fonksiyonunu, elektromekanik etkisizleştirmeyi veya S7-1200'den bağımsız olan diğer yedek emniyet tedbirlerini kullanınız.

## İzolasyon için kılavuzlar

AC devrelere karşı S7-1200 AC güç kaynağı sınırları ve I/O sınırları, AC hat gerilimleri ve düşük gerilim devreleri arasında emniyetli ayırma sağlamak için tasarlanmış ve onaylanmıştır. Bu sınırlar, çeşitli standartlara uygun olarak, iki kat veya takviyeli izolasyonu veya temel ilave destekli izolasyonu içerir. Bu sınırları geçen optik bağlayıcılar, kapasitörler, transformatörler ve röleler gibi bileşenler emniyetli ayırma sağlayıcı olarak onaylanmıştır. Bu kuralları karşılayan izolasyon sınırları, S7-1200 ürün veri sayfalarında 1500 VAC veya daha fazla izolasyona sahip olarak tanımlanmıştır. Bu gösterilme rutin bir (2Ue + 1000 VAC) fabrika deneyi veya onaylanmış yöntemlere göre eşdeğeri temelindedir. S7-1200 emniyetli ayırma sınırları, 4242 VDC için tip deneyine tabi tutulmuştur.

Bir S7-1200'ün sensör besleme çıkışı, haberleşme devreleri ve AC güç beslemesi olan dahili lojik devreleri EN 61131-2'ye göre SELV (emniyetli ekstra düşük gerilim) olarak adlandırılırlar.

S7-1200 düşük gerilim devrelerinin emniyet özelliğini muhafaza etmek için haberleşme portlarına dışarıdan bağlantıların, analog devrelerin ve tüm 24 V nominal güç kaynağı ve I/O devrelerine yapılan harici bağlantıların beslemeleri çeşitli standartlara göre SELV, PELV, Sınıf 2, Sınırlı Gerilim veya Sınırlı güç kurallarını karşılayan onaylı kaynaklarından yapılmalıdır.

**⚠ UYARI**

**Bir AC hattından düşük gerilim beslemesi için izolesiz veya tek izolasyonlu kaynakların kullanımı, güvenli dokunma olabileceği beklenen haberleşme devreleri ve düşük gerilim sensör kablağı gibi devreler üzerinde tehlikeli gerilimlerin oluşmasına neden olabilir.**

Böyle beklenmeyen yüksek gerilimler, ölüme yol açabilecek ciddi kişisel yaralanmalara ve/veya mal zararı meydana getirebilecek elektrik şoklarına neden olabilir.

Sadece temas korumalı, sınırlı gerilim devrelerinin kaynakları olarak onaylanmış yüksek gerilimden düşük gerilime güç dönüştürücüleri kullanınız.

**S7-1200'ün topraklanması için kılavuzlar**

Uygulamanızın topraklanması için en iyi yol, S7-1200'nüzün ve ilgili ekipmanların bütün ortak ve topraklama bağlantılarının tek bir noktaya topraklanmasından emin olmaktır. Bu tek nokta, sisteminize ait toprak topraklamasına doğrudan bağlanmalıdır.

Tüm topraklama kabloları mümkün olduğunca kısa olmalı ve kalın bir tel kesitine sahip olmalıdır; 2 mm<sup>2</sup> (14 AWG) gibi.

Topraklama yeri belirlenirken güvenlik-topraklama kurallarını ve koruma kesme cihazlarının uygun çalışmasını dikkate alınız.

**S7-1200'ün kablağı için kılavuzlar**

S7-1200'nizin kablağı tasarımı yaparken, enerjisi S7-1200 güç kaynağından, tüm giriş devrelerinden ve tüm çıkış devrelerinden aynı anda kesecek tek bir kesici anahtar olmasına dikkat ediniz. Güç kaynağı kablosunda hata akımlarını sınırlamak için bir sigorta veya devre kesici gibi aşırı akım koruması sağlayınız. Ayrıca her bir çıkış devresine sigorta veya diğer akım sınırlayıcı yerleştirerek ilave koruma sağlamayı da düşünebilirsiniz.

Yıldırım gerilim darbesine maruz kalabilecek her hangi bir kablağı için uygun darbe bastırma cihazları monte ediniz.

Düşük gerilim sinyal kablağı ve iletişim kabloları ile AC kablolar ve yüksek enerjili, hızlı tetiklemeli DC kabloları aynı kablağı tavaasına yerleştirmekten kaçınınız. Kabloları daima nötr veya ortak kablağı ile canlı veya sinyal taşıyıcı kablağı çifti olarak yönlendiriniz.

Mümkün olan en kısa kabloyu kullanınız ve kablonun gerekli akımı uygun bir şekilde taşıyabilecek kesite sahip olmasını temin ediniz. CPU ve SM klemensleri, 2 mm<sup>2</sup> – 0,3 mm<sup>2</sup> (14 AWG - 22 AWG) arası kablo kesitleri için uygundur. SB konektörü, 1,3 mm<sup>2</sup> – 0,3 mm<sup>2</sup> (16 AWG - 22 AWG) arası kablo kesitleri için uygundur. Elektrik gürültüsüne karşı en iyi koruma için ekranlı kablolar kullanınız. Tipik olarak, kablo ekranını topraklamak S7-1200'de en iyi sonuçları verir.



Harici bir güç kaynağı tarafından beslenen giriş devreleri kablajı yapılırken bu devreye bir aşırı akım koruma cihazı koyunuz. Bu devreler S7-200'den 24 VDC sensör kaynağı tarafından beslenen devreler için harici koruma gerekli değildir, çünkü sensör kaynağı zaten akım sınırlamalıdır.

Tüm S7-1200 modülleri kullanıcı kablajı için sökülebilir konektörlere sahiptirler. Gevşek bağlantıları önlemek için konektörün yerine oturduğundan ve kablonun konektöre tam olarak sabitlendiğinden emin olunuz. Konektöre zarar gelmesinden kaçınmak için vidaları aşırı sıkılamaya dikkat ediniz. CPU ve SM konektörü vidası için maksimum sıkma kuvveti 0,56 N-m (5 inç pound), SB konektörü vidası için maksimum sıkma kuvveti 0,33 N-m (3 inç pound) olarak verilmiştir.

Kurulumunuzda istenmeyen akım akışlarını önlemeye yardımcı olmak amacıyla S7-1200 belirli noktalarda izolasyon sınırları sağlar. Sisteminiz için kablaj tasarımı yaparken, bu izolasyon sınırlarını dikkate almalısınız. Sağlanan izolasyon miktarı ve izolasyon sınırlarının konumlarını için teknik özelliklere bakınız (Sayfa 829). Güvenlik sınırı olarak 1500 VAC'den az olan izolasyon sınırlarına güvenmeyiniz.

### Lamba yükleri için kılavuzlar

Lamba yükleri, yüksek anahtar-kapama darbe akımı yüzünden röle kontaklarına zarar veriyor. Bu darbe akım, bir Tungsten lamba için nominal olarak kararlı durum akımının 10 – 15 katı olacaktır. Uygulamanın ömrü boyunca çok fazla açma kapama yapılacak lamba yükleri için araya konulan değiştirilebilir bir röle ya da bir darbe akım sınırlayıcısı tavsiye edilir.

### Endüktif yükler için kılavuzlar

Kumanda gerilimi kapatıldığında endüktif yüklerde oluşacak gerilim yükselmesini sınırlamak için bastırma devresi kullanınız. Bastırma devreleri, bir endüktif yükten geçen akım kesildiğinde oluşan yüksek gerilim değişimi tarafından sebep olunan erken arızalardan çıkışlarınızı korur.

Ayrıca, bastırma devreleri endüktif yüklerin anahtarlandığında üreteceği elektriksel gürültüyü de sınırlar. Zayıf olarak bastırılan endüktif yüklerden kaynaklanan yüksek frekans gürültüleri PLC'nin çalışmasını kesintiye uğratabilir. Harici bir bastırma devresini elektriksel olarak yük uçlarına bağlamak ve fiziksel olarak yükün yakınına yerleştirmek elektriksel gürültünün azaltılması için en etkili yoldur.

S7-1200 DC çıkışları dahili bastırma devrelerini içermektedir ve çoğu uygulamalardaki endüktif yükler için yeterlidir. S7-1200 röle çıkış kontakları hem DC hem de AC yükleri anahtarlama için kullanılabilir olduğundan dahili koruma sağlanmamıştır.

İyi bir bastırma çözümü, imalatçılar tarafından yük cihazları ile entegre edilmiş bastırma devreleri veya opsiyonel aksesuarları sağlanan kontaktörler veya diğer endüktif yükler kullanmaktır. Ancak, bazı imalatçıların sağladıkları bastırma devreleri sizin uygulamanız için uygun olmayabilir. Optimum gürültü azaltma ve kontak ömrü için ilave bir bastırma devresi gerekebilir.

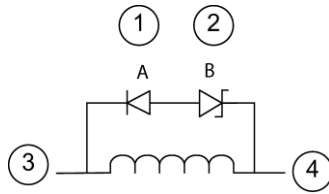
AC yükler için bir metal oksit varistör (MOV) veya diğer gerilim kenetleme cihazı bir paralel RC ile birlikte kullanılabilir, ancak kendi başına kullanıldığındaki gibi etkin değildir. Paralel RC devresi olmayan MOV bastırıcı, kenetleme gerilimi için genellikle dikkate değer yüksek frekanslı gürültü üretir.

İyi kontrol edilen bir enerji kesilme geçici durumu, 10kHz'den daha fazla bir sönümlü doğal frekansa sahip olacaktır, ancak 1kHz'den düşük olanı tercih edilir. AC hattı için pik gerilim toprağa göre +/- 1200 V dahilinde olmalıdır. DC yükler için negatif pik gerilim PLC dahili bastırma kullanılarak 24 VDC kaynak geriliminden yaklaşık 40V düşük olacaktır. Dahili bastırma yüklemem için harici bastırma, geçici durumu 36 V besleme dahilinde sınırlamalıdır.

#### Not

Bir baskılama devresinin etkinliği uygulamaya bağlıdır ve sizin özel kullanımınız için doğrulanmış olmalıdır. Bütün bileşenlerin doğru özellikte seçildiğinden emin olunuz ve enerji kesilmesi anındaki geçici durumu gözlemlemek için bir osiloskop kullanınız.

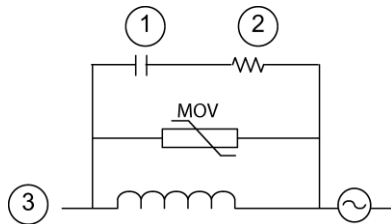
### DC endüktif yükleri anahtarlayan DC veya röle çıkışları için tipik bastırma devresi



- ① 1N4001 diyot veya eşdeğeri
- ② 8.2 V Zener (DC çıkışlar),  
36 V Zener (Röle çıkışları)
- ③ Çıkış noktası
- ④ M, 24 V referans

Çoğu uygulamalarda, bir DC endüktif yük uçlarına bir diyotun (A) eklenmesi uygundur, ancak uygulamanız daha hızlı kesim süresi gerektirirse bir zener diyotunun (B) eklenmesi tavsiye edilir. Çıkış devrenizdeki akım değerine göre uygun özellikte zener diyodu seçtiğinizden emin olunuz.

### AC endüktif yükleri anahtarlayan röle çıkışları için tipik bastırma devresi



- ① C değeri için Çizelgeye bakınız
- ② R değeri için Çizelgeye bakınız
- ③ Çıkış noktası

Metal oksit varistörün (MOV) çalışma geriliminin, nominal hat geriliminden en az %20 fazla olmasını temin ediniz.

Pals-oranlı, endüktif olmayan dirençleri ve pals uygulamaları için tavsiye edilen kapasitörleri (tipik olarak metal film) seçiniz. Elemanların ortalama güç, pik güç ve pik gerilim koşullarını karşıladığını doğrulayınız.

Kendi bastırma devrenizi tasarladığınız durumunda, AC yüklerin bir aralığı için direnç ve kondansatörlerin değerleri aşağıdaki Çizelgede tavsiye edilmektedir. Bu değerler ideal eleman parametreleri ile hesaplamaya görelerdir. Çizelgedeki  $I_{rms}$  (etkin) tamamen ON olduğunda yükün kararlı-durum akımını temsil eder.

Çizelge 3- 17 AC bastırma devresi rezistör ve kapasitör değerleri

İndüktif yük			Bastırma değerleri		
I rms	230 VAC	120 VAC	Rezistör		Kapasitor
Amper	VA	VA	$\Omega$	W (güç değeri)	nF
0.02	4.6	2.4	15000	0.1	15
0.05	11.5	6	5600	0.25	470
0.1	23	12	2700	0.5	100
0.2	46	24	1500	1	150
0.05	115	60	560	2.5	470
1	230	120	270	5	1000
2	460	240	150	10	1500

**Çizelge değerleri tarafından yeterli görülmüş koşullar**

- Maksimum kesme (turn-off) geçiş adımı < 500 V
- Rezistör pik gerilimi < 500 V
- Kapasitör pik gerilimi < 1250 V
- Bastırma akımı < 8% yük akımının (50 Hz)
- bastırma akımı < 11% yük akımının (60 Hz)
- Kapasitör  $dV/dt < 2 V/\mu s$
- Kapasitör pals kaybı :  $\int (dv/dt)^2 dt < 10000 V^2/\mu s$
- Rezonans frekansı < 300 Hz
- Rezistör gücü, 2 Hz max anahtarlama frekansı için
- Güç faktörü 0.3, tipik endüktif yükler için varsayılan

**Diferansiyel giriş ve çıkışlar kılavuzlar**

Diferansiyel girişler ve çıkışların davranışı standart giriş ve çıkışlarından farklıdır. Her bir diferansiyel giriş ve çıkışı için iki pin vardır. Bir diferansiyel giriş veya çıkışın on veya off olup olmadığını belirlemek bu iki pin arasındaki gerilim farkını ölçmenizi gerektirir.

CPU 1217C için ayrıntılı özelliklere Ek A'da (Sayfa 879).bakınız.

*Kurulum*

*3.4 Kablaj kılavuzları*

---

## PLC konseptleri

### 4.1 Kullanıcı programının yürütülmesi

CPU, kullanıcı programınız için etkili bir yapı oluşturmanıza izin veren aşağıdaki kod bloklarını destekler:

- Organizasyon blokları (OB'ler) program yapısını tanımlar. Bazı OB'lerin önceden tanımlanmış davranışlara ve başlatma olaylarına sahiptirler, ancak siz de özel başlatma olayları ile OB'ler oluşturabilirsiniz.
- Fonksiyonlar (FC'ler) ve fonksiyon blokları (FB'ler), özel görevlere veya parametrelerin kombinasyonlarına karşılık gelen program kodunu içerir. Her bir FC veya FB, çağrı yapan blok ile veri paylaşımı için bir giriş ve çıkış parametreleri seti sağlar. Bir FB ayrıca o FB çağırma durumu için veri değerlerini muhafaza etmek için ilişkilendirilmiş bir veri bloğu (bir instance DB olarak adlandırılır) kullanır. Bir FB'yi birçok kere çağırabilirsiniz. Aynı FB'nin farklı durum DB'leri ile çağırılması diğer herhangi bir durum DB'sinin veri değerlerini etkilemez.
- Veri blokları (DB'leri) program blokları tarafından kullanılabilen veriyi saklar.
- Kullanıcı programın yürütülmesi, Run moduna girildiğinde bir kez yürütülen bir veya daha fazla opsiyonel başlangıç organizasyon bloklarının (OB'lerin) yürütülmesi ile başlar ve bir veya daha fazla program döngü OB'sinin döngüsel yürütmesiyle devam eder.
- Bir OB ayrıca bir standart olay veya bir hata olayı olabilen bir interrupt (kesme) olayı ile de ilişkilendirilebilir. Her ne zaman karşılık gelen standart veya hata olayı meydana gelse bu OB'ler yürütür.

Bir fonksiyon (FC) veya bir fonksiyon bloğu (FB), bir OB'den veya başka bir FC'den veya FB'den çağrılabilen bir program kodu olup izleyen iç içe çağırma derinliklerine sahiptir:

- Program döngüsünden veya başlangıç OB'sinden 16 adet
- Herhangi bir interrupt olay OB'sinden 6 adet


FC'ler herhangi bir özel veri bloğu (DB) ile ilişkilendirilmezler. FB'ler doğrudan bir DB'ye bağlanmıştır ve parametrelerin geçişi ve geçici değerlerin ve sonuçların saklanması için DB'yi kullanır.

Kullanıcı programı, veriler ve konfigürasyonun boyutu, CPU içinde mevcut yükleme belleği ve çalışma belleği tarafından sınırlanır. Her bir bağımsız OB, FC, FB ve DB bloğu sayısı için belirli bir limit yoktur. Ancak toplam blok sayısı 2014 ile sınırlıdır.

Her bir döngü; çıkışların yazılmasını, girişlerin okunmasını, kullanıcı programı komutlarının yürütülmesini ve arka plan işlemlerinin gerçekleştirilmesini içerir. Döngü, bir tarama döngüsü veya tarama olarak adlandırılır.

Sizin S7-1200 otomasyon çözümünüz, S7-1200 CPU ve ilave modüllere sahip bir merkezi şasi (central rack) içerir. "merkezi şasi" terimi S7-1200 CPU ve ilgili modüllerin ray veya panel montajını ifade eder. Modüller (SM, SB, BB, CB, CM veya CP) sadece enerjili olmaları durumunda algılanır ve giriş yapılır.

- Enerjili olan merkezi şasideki bir modülün takılması veya çıkarılması desteklenmez. CPU enerjili iken merkezi şasiden herhangi bir modülü asla takmayınız veya çıkarmayınız.

 <b>UYARI</b>
<b>Modüllerin takılması veya çıkarılması için güvenlik kuralları</b>
CPU enerjili iken merkezi şaside bir modülün (SM, SB, CB, CM veya CP) takılması veya çıkarılması tahmin edilemez davranışlara neden olabilir ve bunun sonucunda ekipmana zarar gelebilir ve/veya personel yaralanabilir.
Merkezi şasiden bir modülü takmadan veya çıkarmadan önce daima CPU ve merkezi şasi elektrik beslemesini ayırınız ve uygun güvenlik tedbirlerini alınız.

- CPU enerjili iken bir SIMATIC bellek kartını takabilir veya çıkarılabilirsiniz. Ancak CPU RUN modunda iken bir hafıza kartının takılması ya da çıkarılması CPU'nun STOP moduna geçmesine neden olur.

<b>DİKKAT</b>
<b>CPU RUN modunda iken bir hafıza kartının takılması veya çıkarılmasıyla ilgili riskler</b>
CPU RUN modunda iken bir hafıza kartının takılması veya çıkarılması, CPU'nun STOP moduna geçmesine neden olur. Bu durum ekipmana veya kontrol edilmekte olan prosese zarara yol açabilir.
Her ne zaman bir hafıza kartı taktığınızda veya çıkardığınızda, CPU derhal STOP moduna geçer. Bir hafıza kartını takmadan veya çıkarmadan önce daima CPU'nun bir makineyi veya bir prosesi aktif olarak kontrol etmekte olmadığından emin olunuz. Uygulamanız veya prosesiniz için daima bir acil durdurma devresi kurunuz.

CPU RUN modunda iken dağıtılmış bir I/O şasisine (AS-i, PROFINET veya PROFIBUS) bir modül takar veya çıkarırsanız, CPU diyagnostik ara belleğinde bir giriş üretir, bulunursa OB modülünü itme veya çekme işlemini yürütür ve varsayılan RUN modunda kalır.

## Proses görüntü güncellemesi ve proses görüntü bölüntüleri

CPU, lojik dijital ve analog I/O noktalarını proses görüntüsü adı verilen bir dahili hafıza alanını kullanarak tarama döngüsü ile eş zamanlı olarak günceller. Proses görüntüsü, fiziksel girişlerin ve çıkışların (CPU üzerindeki fiziksel I/O noktaları, sinyal kartı ve sinyal modülleri) bir bellek kopyasını içerir.

Her bir tarama döngüsünde veya özel bir olay interrupt'ı meydana geldiğinde güncellenecek I/O noktalarının konfigürasyonunu yapabilirsiniz. Proses görüntüsü güncellemelerinden hariç tutulacak bir I/O noktasının konfigürasyonunu da yapabilirsiniz. Örneğin, bir donanım interrupt olayı meydana geldiğinde sizin prosesinizin belirli veri değerlerine ihtiyacı olabilir. Bir donanım interrupt'ı OB'ye atadığınız bir bölüntü (partition) ile ilişkili olacak bu I/O noktaları için proses görüntüsünün konfigürasyonunu yaparak sizin prosesiniz sürekli bir güncelleme gerektirmediğinde her tarama döngüsünde CPU'nun veri değerlerini güncellemesinden kaçınabilirsiniz.

Proses görüntüsü döngüsünde güncellenen I/O için CPU her bir tarama döngüsü esnasında aşağıdaki görevleri yerine getirir:

- CPU, proses görüntüsü çıkış alanından çıkışları fiziksel çıkışlara yazar.
- CPU, fiziksel girişleri kullanıcı programının yürütülmesinden hemen önce okur ve giriş değerlerini proses görüntüsü giriş alanında depolar. Böylece bu değerler kullanıcı komutlarının yürütülmesi süreci boyunca sürekli (consistent) kalır.
- CPU, kullanıcı komutlarının mantığını yürütür ve gerçek fiziksel çıkışlara yazmak yerine proses imaj alanındaki çıkış değerlerini günceller

Bu işlem, verilen bir döngü için kullanıcı komutlarının yürütülmesi süresince tutarlı (consistent) bir lojik sağlar ve proses görüntüsü çıkış alanında birkaç kez durum değiştirebilecek fiziksel çıkış noktalarının kırışmalarını (flickering) önler.

S7-1200, I/O noktalarının her tarama döngüsünde otomatik olarak veya olayların tetiklenmesine göre güncellenip güncellenmeyeceğini kontrol etmek için 5 adet proses görüntüsü bölüntüsü sağlar. İlk proses görüntü bölüntüsü, PIP0, her tarama döngüsünde otomatik olarak güncellenecek olan I/O için belirlenmiştir ve varsayılan atamadır.

Geri kalan 4 bölüntüyü PIP1, PIP2, PIP3 ve PIP4 çeşitli interrupt olaylarına I/O proses görüntü güncellemelerini atamak için kullanabilirsiniz.

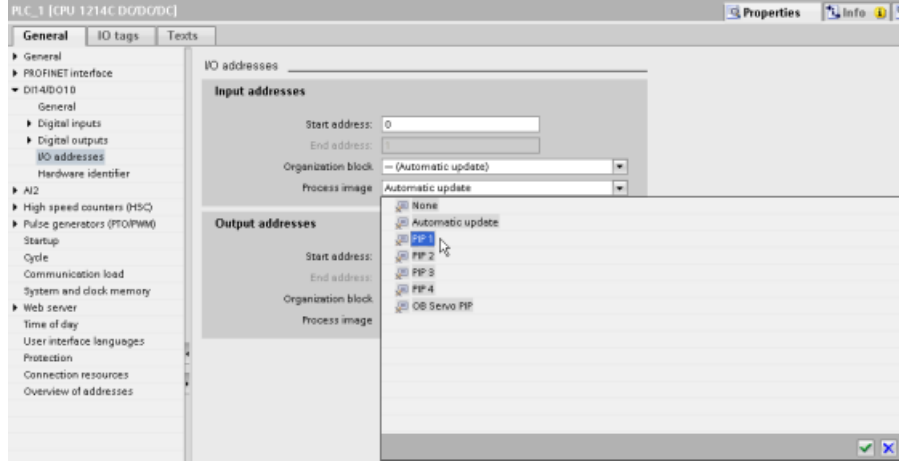
Proses görüntü bölüntülerine I/O atamasını Device Configuration'da yapar ve interrupt OB'lerini (Sayfa 172) oluştururken veya OB özelliklerini (Sayfa 172) düzenlerken proses görüntü bölüntülerini interrupt olaylarına atarsınız.

Varsayılan, cihaz görünümüne bir modül eklediğiniz zaman, STEP 7 kendi I/O proses görüntü güncellemesini "Automatic update"e ayarlar. "Automatic update"te göre konfigüre edilen I/O için CPU her tarama döngüsü boyunca otomatik olarak modül ve proses görüntü alanı arasında veri değişimi işlemini yapar.

Bir proses görüntü bölüntüsüne dijital veya analog noktalar atamak için veya proses görüntü güncellemelerinden I/O noktalarını hariç tutmak için aşağıdaki adımları takip ediniz:

1. Device configuration'da uygun cihaz için Properties sekmesini inceleyiniz.
2. İstenen I/O noktalarını yerleştirmek için gerektiği gibi "General" altında seçimleri genişletiniz.
3. "I/O addresses" i seçiniz.

4. Opsiyonel olarak "Organization block" açılır listesinden bir spesifik OB seçiniz.
5. "Process image" açılır listesinden "Automatic update" i "PIP1", "PIP2", "PIP3", "PIP4" veya "None" a değiştiriniz. "None" seçimi, anında (immediate) komutlar kullanarak bu I/O 'tan sadece okuyabilir ve yazabilirsiniz anlamına gelmektedir. Noktaları proses görüntü otomatik güncellemeye geri eklemek için bu seçimi "Automatic update" e değiştiriniz.



Bir komut yürütüldüğünde, fiziksel giriş değerlerini anında okuyabilir ve fiziksel çıkış değerlerini anında yazabilirsiniz. Bir anında okuma, fiziksel girişin o andaki durumuna erişir ve proses görüntü alanını noktanın proses görüntüsünde saklanmak için konfigüre edilip edilmediğine dikkat etmeksizin güncellemez. Fiziksel girişe bir anında yazma hem proses çıkış alanını(nokta, proses alanında saklanacak şekilde konfigüre edilirse) hem de fiziksel çıkış noktasını günceller. Proses görüntü kullanmak yerine siz programın I/O verisine doğrudan fiziksel noktadan erişmesini isterseniz I/O 'a " :P" takısını iliştiniriz.

## Not

### Proses görüntü bölüntülerinin kullanımı

Siz I/O' u PIP1 - PIP4 proses görüntü bölüntülerinden birine atarsanız ve bir OB'yi o bölüntüye atamazsanız, CPU söz konusu I/O' u proses görüntüsüne veya görüntüsünden asla güncellemez. I/O' un bir PIP' e atanması (denk düşen bir OB atanmasına sahip değildir) proses görüntüsünü bir "None" a atamak ile aynıdır. Bir anında okuma komutu ile fiziksel I/O' tan doğrudan I/O' u okuyabilirsiniz veya bir anında yazma komutu ile fiziksel I/O' a doğrudan yazabilirsiniz. CPU, proses görüntüsünü güncellemez.

CPU, dağıtımlı I/O' u hem PROFINET hem de PROFIBUS ağları için destekler (Sayfa 509).



### 4.1.1 CPU'nun işletim modları

CPU'nun üç çalışma modu vardır: STOP modu, STARTUP modu ve RUN modu. CPU'nun ön kısmındaki durum LED'leri geçerli işletim modunu gösterir.

- STOP modunda, CPU program yürütmez. Bir projeyi yükleyebilirsiniz.
- STARTUP modunda, başlangıç OB'leri (varsa) bir kez yürütülür. CPU, başlangıç modu boyunca Interrupt olaylarını işlemez.
- RUN modunda, program döngüsü OB'leri defalarca yürütme yapar. Interrupt olayları meydana gelebilir ve karşılık gelen interrupt olayı OB'leri RUN modu içinde herhangi bir noktada yürütme yapabilir (Sayfa 803).

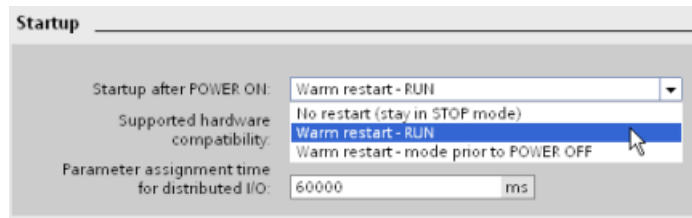
RUN moduna girmek için CPU sıcak (warm) bir yeniden başlatmayı destekler. Sıcak başlatma bir hafıza reset işlemi içermez. CPU, bütün kalıcı olmayan sistem ve kullanıcı verilerini sıcak yeniden başlatmada başlangıç durumuna getirir ve bütün kalıcı kullanıcı verilerinin değerlerini tutar.

Bir bellek resetlemesi, tüm çalışma belleğini temizler, kalıcı ve kalıcı olmayan bellek alanlarını temizler ve yükleme belleğini çalışma belleğine kopyalar, konfigüre edilen "Reaction to CPU STOP"a çıkışları set eder. Bir bellek resetlemesi, diyagnostik ara belleğini veya IP adreslerinin kalıcı olarak kaydedilmiş değerlerini temizler.

Siz CPU'nun "startup after POWER ON" ayarlamasını konfigüre edebilirsiniz. CPU için bu konfigürasyon ögesi "Startup" altında "Device configuration" altında görülür. Enerjilenmesinden sonra CPU başlama diyagnostik kontrolleri ve sistem başlangıç ayarlarının bir dizisini yerine getirir.

Sistem başlangıç ayarları boyunca CPU, bütün kalıcı olmayan bit (M) belleğini siler ve bütün kalıcı olmayan DB içeriklerini yük belleğinden başlangıç değerlerine resetler. CPU, kalıcı bit (M) belleğini ve kalıcı DB içeriğini tutar ve sonra uygun işletim moduna girer. Belirli algılanan hatalar CPU'nun RUN moduna girmesine engel olurlar. CPU, aşağıdaki konfigürasyon seçeneklerini destekler:

- Restart yok- (STOP modda kal) (No restart)
- Sıcak restart – RUN (Warm restart – RUN)
- Sıcak restart - mod (POWER OFF öncesi) (Warm restart-mod)



**DİKKAT****Tamir edilebilir arızalar CPU'nun STOP moduna girmesine neden olabilir.**

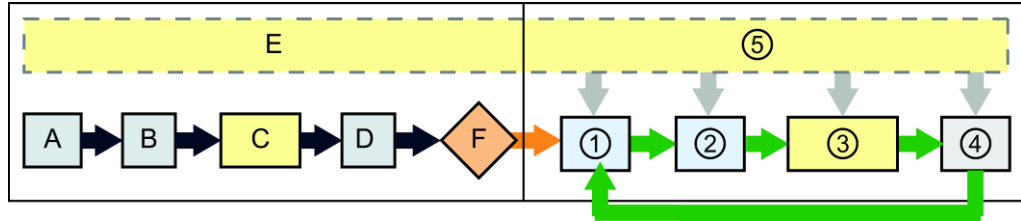
Tamir edilebilir arızalar yüzünden CPU, STOP moduna girebilir. Arızalar, değiştirilebilir sinyal modülündeki bir arıza veya güç beslemesinde bozulma veya değişken güç beslemesi olayı gibi geçici arızalar şeklinde olabilir. Böyle durumlar mal zararına yol açabilir.

CPU'yu "Warm restart - mode prior to POWER OFF"a konfigüre etmiş iseniz, CPU güç kesilmesi veya arıza öncesi olan işletme moduna gider. CPU güç kesilmesi veya arıza anında STOP modunda ise, CPU güç geldiğinde STOP moduna gider ve RUN moduna gitmesi için bir komut gelinceye kadar STOP modunda kalır. CPU güç kesilmesi veya arıza anında RUN modunda ise, CPU bir sonraki güç gelmesinde RUN moduna geçişi engelleyen hatalar algılanmadıkça RUN moduna gider

CPU'ları bir STEP 7'den bağımsız olarak işletilebilmesi amacıyla "Warm restart - RUN" bağlantısına konfigüre ediniz. Böylece CPU bir sonraki güç gelmesi döngüsünde RUN moduna dönebilir.

Programlama yazılımının çevrim içi (çevrimiçi) araçlarından "STOP" veya "RUN" komutlarını (Sayfa 792) kullanarak o anki çalışma modunu değiştirebilirsiniz. CPU'yu STOP moduna değiştirmek için programınıza bir STP komutu da (Sayfa 263) ekleyebilirsiniz. Bu işlem sizin program lojijinize bağlı olarak programınızın yürütmesini durdurmanıza izin verir.

- STOP modunda, CPU herhangi bir haberleşme talebini (uygun olarak) işler ve öz-diyagnostik işlemini gerçekleştirir. CPU, kullanıcı programını yürütmez ve proses görüntünün otomatik güncellemeleri olmaz.
- STARTUP ve RUN modlarında aşağıda şekilde gösterilen görevleri yerine getirir.


**STARTUP**

- A I (image) hafıza alanını temizler
- B Q çıkış (image) hafıza alanı sıfır ile, son değer, veya değiştirme değeri ile, konfigüre edildiği ve sıfır PB, PN ve AS-i çıkışlar gibi başlangıç değerlerine ayarlar
- C Kalıcı olmayan M belleği ve veri bloklarını ilk değerlerine döndürür ve konfigüre edilen döngü interruptını ve olayların gün içindeki saatini etkinleştirir  
Açılış OB'lerini yürütür
- D Fiziksel girişlerin durumunu I belleğe kopyalar
- E Herhangi bir interrupt olayını RUN moduna girdikten sonra işlenmek üzere sıraya koyar
- F Q belleğin fiziksel çıkışlara yazılmasını etkinleştirir

**RUN**

- ① Q belleği fiziksel çıkışlara yazar
- ② Fiziksel girişlerin durumunu I belleğe kopyalar
- ③ Program çevrimleri OB'leri yürütür
- ④ Öz-test diyagnostiklerini yerine getirir
- ⑤ Tarama döngüsünün her hangi bir bölümünde interruptlar ve haberleşmeyi işler

**STARTUP processing (Başlangıç işleme)**

İşletim modu her ne zaman STOP'tan RUN'a değişse, CPU proses görüntü girişlerini temizler ve proses görüntü çıkışlarını başlangıç durumuna getirir ve başlangıç OB'lerini işler. Başlangıç OB'lerindeki komutlarla proses-görüntü girişlerine her hangi bir okuma erişimi geçerli fiziksel giriş değeri yerine sıfır okur. Bu nedenle, başlangıç modu boyunca bir fiziksel girişin o anki durumunu okumak için bir anında okuma gerçekleştirmelisiniz. Sonra başlangıç OB'leri ve ilişkili FC'ler ve FB'ler yürütülür. Birden fazla başlangıç OB'si mevcut ise, her birisi OB numarasına göre (en düşük OB numarası önce yürütülerek) yürütülür.

Her bir başlangıç OB, kalıcı veri ve günün zaman saatinin doğruluğunu belirlemenize yardımcı olan başlangıç bilgisini içerir. Bu başlangıç değerlerini test etmek ve uygun aksiyonu yapmak için komutları başlangıç OB'leri içerisinde programlayabilirsiniz. Aşağıdaki başlangıç yerleri başlangıç OB'leri tarafından desteklenir:

Çizelge 4- 1 Başlangıç OB'si tarafından desteklenen başlangıç yerleri

Giriş	Veri tipi	Açıklama
LostRetentive	Bool	Kalıcı veri saklama alanları kaybedilmiş ise bu bit doğrudur (true)
LostRTC	Bool	Günün zaman saati (Real time Clock) kaybedilmiş ise bu bit doğrudur (true)

Başlangıç işlemleri boyunca CPU aşağıdaki görevleri de yerine getirir:

- Interruptlar sıraya konulur, ancak başlangıç evresi boyunca işlenmezler.
- Başlangıç evresi boyunca döngü süresi izlemesi yapılmaz.
- Konfigürasyon HSC'ye (yüksek-hızlı sayıcı) değişir, PWM (darbe genişlik modülasyonu) ve PtP (noktadan noktaya haberleşme) modülleri başlangıçta yapılabilir.
- HSC, PWM ve noktadan-noktaya haberleşme modüllerinin gerçek çalışması sadece RUN modunda oluşur.

Başlangıç OB'lerinin yürütülmesi bittikten sonra CPU, RUN moduna geçer ve sürekli bir tarama döngüsü içinde kontrol görevlerini gerçekleştirir.

#### 4.1.2 RUN modunda tarama döngüsünün işlenmesi

Her tarama döngüsü için CPU çıkışları yazar, girişleri okur, kullanıcı programını yürütür, haberleşme modüllerini günceller ve kullanıcı interrupt olaylarıyla haberleşme taleplerine yanıt verir. Haberleşme talepleri, tarama aracılığıyla periyodik olarak işlenir.

Bu aksiyonlar (kullanıcı interrupt olayları hariç), düzenli olarak ve sıralı düzende servis edilir. Etkinleştirilen kullanıcı interrupt olayları, meydana gelme sırası önceliğine göre servis edilir. Interrupt olayları için CPU girişleri okur, OB'yi yürütür ve sonra çıkışlara yazar. CPU bu işlemleri, uygulanabilirse, ilişkili proses bölüntüsünü (PIP) kullanarak yapar.

Sistem bu tarama döngüsünün maksimum döngü zamanı adı verilen bir süre içerisinde tamamlanmasını garanti eder; aksi durumda bir zaman hatası olayı üretilir.

- Her bir tarama döngüsü, dijital ve analog çıkışların proses görüntüsünden alınmasıyla ve sonra bunların otomatik I/O güncellemesi için konfigüre edilmiş (varsayılan konfigürasyon) CPU, SB ve SM modüllerinin fiziksel çıkışlarına yazılmasıyla başlar. Bir fiziksel çıkışa bir komut ile erişim sağlandığında, hem çıkış proses görüntüsü hem de fiziksel çıkışın kendisi güncellenir.
- Tarama döngüsü, dijital ve analog girişlerin geçerli değerlerinin otomatik I/O güncellemesi (varsayılan konfigürasyon) için konfigüre edilmiş CPU, SB ve SM'lerden okunması ve sonra bu değerlerin proses görüntüsüne yazılmasıyla devam eder.
- Bir fiziksel girişe bir komut ile erişim sağlandığında, fiziksel girişin değerine komut tarafından erişim sağlanmış olur, ancak giriş proses görüntüsü güncellenmez.
- Girişlerin okunmasından sonra, kullanıcı programı ilk komuttan son komuta kadar yürütülür. Bu, bütün program döngüsü OB'lerine ilaveten bütün kendileriyle ilişkili FC'leri ve FB'leri içerir.
- Program döngüsü OB'leri, OB numara sırasına göre en düşük OB numarası önce yürütülecek şekilde yürütülür.

Haberleşme işlemesi tarama aracılığıyla periyodik olarak gerçekleşir ve muhtemelen kullanıcı programı yürütmesini kesintiye uğratır.

Öz-diyagnostik kontrolleri, sistemin periyodik kontrolleri ile I/O modülünün durum kontrollerini kapsar. Interruptlar, tarama döngüsünün herhangi bir bölümü boyunca meydana gelebilir ve olaya dayalıdır. Bir olay meydana geldiğinde, CPU tarama döngüsünü keser ve bu olayı işlemek için yapılandırılmış OB'yi çağırır. OB olayı işlemeyi bitirdikten sonra, CPU kullanıcı programının yürütülmesini interrupt'ın olduğu yerden sürdürür.

#### 4.1.3 Organizasyon blokları (OB'ler)

OB'ler, kullanıcı programının yürütülmesini kontrol ederler. CPU'daki bazı olaylar, bir organizasyon bloğunun yürütülmesini tetikler. OB'ler birbirlerini çağıramazlar veya bir FC'den veya FB'den çağrılmazlar.

Bir diyagnostik interrupt'ı veya bir zaman aralığı interrupt'ı gibi sadece bir olay bir OB'nin yürütülmesini başlatabilir. CPU, OB'leri öncelik sınıflarına göre işler. Burada yüksek öncelikli OB'ler düşük öncelikli OB'lerden önce yürütülür. En düşük öncelik sınıfı 1 (ana program döngüsü için) iken en yüksek öncelik sınıfı 24 olmaktadır.

### 4.1.3.1 Program döngüsü OB

CPU, RUN modunda iken program döngüsü OB'leri yürütmeyi döngüsel olarak gerçekleştirirler. Programın ana bloğu bir program döngüsü OB'sidir. Burası, programınızı kontrol eden komutları yerleştirdiğiniz ve ilave kullanıcı bloklarını çağırdığınız yerdir. CPU'nun numara sırasına göre yürüttüğü çoklu program döngüsü OB'lerine sahip olabilirsiniz. Esas (OB1) varsayılandır.

### Program döngü olayları

Program döngü olayı her program döngüsü (veya tarama) boyunca bir defa olur. Program döngüsü boyunca CPU çıkışları yazar, girişleri okur ve program döngü OB'lerini yürütür. Program döngü olayı istenir ve daima etkindir. Program döngü OB'lerine sahip olmayabilirsiniz veya program döngü olayı için seçilmiş çoklu OB'lere sahip olabilirsiniz. Program döngü olayı meydana geldikten sonra CPU en düşük numaralı program döngüsü OB'yi (genellikle "Main" OB 1) yürütür. CPU diğer program döngü OB'lerini program döngüsü içinde sırayla (sayısal düzen içinde) yürütür. Program yürütülmesi aşağıdaki zamanlarda oluşan program döngü olayı gibi döngüseldir:

- Son başlangıç OB yürütmeyi bitirdiği zaman
- Son program döngüsü OB yürütmeyi bitirdiği zaman

Çizelge 4- 2 Bir program döngüsü OB için başlama bilgisi

Giriş	Veri tipi	Açıklama
Initial_Call	Bool	OB'nin başlangıç çağırması için doğru (True)
Remanence	Bool	Kalıcı veri hazır ise doğru (True)

### 4.1.3.2 Başlangıç OB (Startup OB)

Başlangıç OB'ler, CPU'nun işletim modu STOP'tan RUN'a geçtiğinde bir kez yürütür ve RUN modunda ve komut verilen STOP-RUN geçişlerinde güç verilmesini de içerir. Tamamlandıktan sonra, ana "Program cycle" yürütmeye başlar.

### Başlangıç olayları (Startup events)

Başlangıç olayı, bir STOP'tan RUN'a geçişte bir kez olur ve CPU'nun başlangıç OB'lerinin yürütmesine yol açar. Başlangıç olayı için çoklu OB configure edebilirsiniz. Başlangıç OB'leri, numara sırasına göre komut gönderir.

Çizelge 4- 3 Bir başlangıç OB'si için başlama bilgisi

Giriş	Veri tipi	Açıklama
LostRetentive	Bool	Kalıcı veri kaybedilirse doğru (True)
LostRTC	Bool	Tarih ve saat kaybedilirse doğru (True)

#### 4.1.3.3 Zaman gecikmeli interrupt (kesme) OB

Konfigüre ettiğiniz bir zaman gecikmesinden sonra zaman gecikmeli interrupt OB'leri komut gönderir.

#### Zaman gecikmeli interrupt olayları

Zaman gecikmeli interrupt olaylarını belirli bir gecikme zamanı dolduktan sonra oluşması için konfigüre edebilirsiniz. Gecikme zamanını atamasını SRT\_DINT komutu ile yaparsınız. Zaman gecikmesi olayları, denk düşen zaman gecikmeli interrupt OB'sini yürütmek için program döngüsünü kesintiye uğratır. Bir zaman gecikmesi olayına sadece bir zaman gecikmeli interrupt OB'si ekleyebilirsiniz. CPU, dört zaman gecikmesi olayını destekler.

Çizelge 4- 4 Bir zaman gecikmeli interrupt OB için başlama bilgisi

Giriş	Veri tipi	Açıklama
Sign	Word	Tanımlayıcı SRT_DINT tetikleme çağrısına geçti

#### 4.1.3.4 Döngüsel interrupt OB

Döngüsel interrupt OB'ler belirli bir aralıkta yürütme yaparlar. Toplam dört döngüsel interrupt olayını kadar konfigüre edebilirsiniz. Burada bir OB her döngüsel interrupt olayına karşılıktır.

#### Döngüsel interrupt olayları

Döngüsel interrupt olayları, bir konfigüre edilmiş döngü süresinde bir interrupt'ın yürütülmesini konfigüre etmek için size izin verir. Döngüsel interrupt OB oluşturduğunuzda, başlama döngü süresini konfigüre edersiniz. Bir döngüsel olay program döngüsünü keser ve karşılık gelen döngüsel interrupt OB'sini yürütür. Döngüsel interrupt olayının program döngü olayından daha yüksek öncelik sınıfında olduğuna dikkat ediniz.

Bir döngüsel olaya sadece bir döngüsel interrupt OB iliştirebilirsiniz.

Her bir döngüsel olaya bir faz kaydırma atayabilirsiniz. Böylece döngüsel interrupt'ların aynı tarama zamanı içerisindeki yürütülmesi bir diğerinden faz ofset miktarı kadar ofset olabilir. Varsayılan faz ofseti 0'dır. Başlamadaki faz kaydırmasını değiştirmek veya bir döngüsel olay için başlamadaki tarama zamanını değiştirmek için proje ağacındaki döngüsel interrupt OB'sinin üzerine sağ tıklayınız, sonra içerik menüsünden "Properties"i seçiniz, sonra "Cyclic interrupt" (döngüsel interrupt) üzerine tıklayınız ve sonra yeni başlangıç değerlerini giriniz. Ayrıca, döngüsel interrupt sorgulama (QRY\_CINT) ve Döngüsel interrupt'ı ayarla (SET\_CINT) komutlarını kullanarak programınızdan tarama zamanını ve faz kaydırmayı sorgulayabilir ve değiştirebilirsiniz. SET\_CINT komutu tarafından ayarlanmış tarama zamanı ve faz kaydırma değerleri, bir güç döngüsü veya STOP moduna geçiş boyunca kalıcı olmaz; tarama zamanı ve faz kaydırma değerleri bir güç döngüsü veya STOP moduna geçiş sonrasında ilk değerlerine döner. CPU, toplamda dört döngüsel interrupt olayını destekler.

#### 4.1.3.5 Donanım interrupt OB

İlgili donanım olayları oluştuğunda donanım interrupt OB'leri yürütme yapar. Bir donanım interrupt OB bir donanın olayından bir sinyale tepki olarak normal program yürütmeyi keser.

### Donanım interrupt olayları

Bir giriş noktasında yükselen veya düşen bir kenar veya (bir HSC, Yüksek Hızlı Sayıcı) olay gibi donanımdaki değişiklikler donanım interrupt olaylarını tetikler. Her bir donanım interrupt olayı için S7-1200 bir interrupt OB destekler. Donanım olaylarını cihaz konfigürasyonunda etkinleştirirsiniz ve cihaz konfigürasyonunda veya kullanıcı programında bir ATTACH komutu ile bir olay için bir OB atayınız. CPU, çok sayıda donanım interrupt olayını destekler. CPU modeli ve giriş noktalarının sayısı hazır olan tam olayları belirler.

Donanım interrupt olayları üzerindeki sınırlamalar aşağıdaki gibidir:

#### Kenarlar:

- Yükselen kenar olayları: maksimum 16
- Düşen kenar olayları: maksimum 16

#### HSC olayları:

- CV=PV: maksimum 6
- Yön değiştirmesi: maksimum 6
- Harici reset: maksimum 6

#### 4.1.3.6 Zaman hatası interrupt'ı OB

Konfigüre edilmiş ise, tarama döngüsü maksimum döngü süresini aştığı veya bir zaman hatası olayı olduğu durumda zaman hatası interrupt'ı OB (OB 80) yürütür. Tetiklenirse, OB normal döngüsel program yürütmesini veya diğer bir olay OB kesmeyi yürütür.

Bu iki olaydan birisinin oluşması olayı açıklayan bir diyagnostik arabellek girişini üretir. Diyagnostik arabellek girişi zaman hatası interrupt'ı OB'nin bulunmasından bağımsız olarak üretilir.

### Zaman hatası interrupt olayları

Birçok farklı zaman hatası şartlarının herhangi birinin oluşması bir zaman hatası olayıyla sonuçlanır:

- Tarama döngüsü maksimum döngü süresini aşar
- Program çevrimi belirlenen maksimum tarama döngüsü süresinde tamamlanmazsa, maksimum döngü süresi aşıldı ("maximum cycle time exceeded") durumu meydana gelir. Maksimum döngü süresi şartına, CPU özelliklerinde maksimum döngü süresini nasıl konfigüre edileceğine ve döngü zamanlayıcısının nasıl resetleneceğine ilişkin bilgi için "Monitoring the cycle time in the S7-1200 System Manual" (Sayfa 97) bölümüne bakınız.
- CPU ilk interrupt OB'sinin yürütmesini bitirmeden önce ikinci bir zaman interrupt'ı (döngüsel veya zaman gecikmeli) başlamasından dolayı CPU talep edilen OB'yi başlatamaz.
- Kuyruk aşırı akışı oluştu

Interrupt'lar CPU'nun onları işlemesinden daha hızlı oluşursa, kuyruk aşırı akışı oluştu ("queue overflow occurred") koşulu meydana gelir. CPU, her bir olay tipi için farklı bir kuyruk kullanarak olayların bekleme sayısını sınırlar. Karşılık gelen kuyruk dolu olduğunda bir olay meydana gelirse, CPU bir zaman hatası olayı üretir.



Bütün zaman hatası olayları, zaman hatası interrupt'ı OB'sinin (var ise) yürütülmesini tetikler. Zaman hatası interrupt'ı OB'si yok ise, CPU'nun cihaz konfigürasyonu zaman hatası için CPU'nun tepkisini belirler:

- CPU'nun ilk yürütmeyi tamamlamasından önce ikinci bir döngüsel interrupt'ın başlatılması gibi zaman hataları için varsayılan konfigürasyon CPU'nun RUN modunda kalmasına göreler.
- Maksimum süreleri geçme için varsayılan konfigürasyon CPU'nun STOP moduna geçmesine göreler.

Kullanıcı programı, döngü süre izlemesini tekrar başlatmak için RE\_TRIGR komutunu (Sayfa 262) yürüterek program döngü yürütme süresini konfigüre edilen maksimum döngü süresinin on katına kadar artırabilir. Ancak, döngü zamanlayıcısını reset etmeksizin aynı program döngüsü içinde "maksimum döngü süresi aşıldı" ("maximum cycle time exceeded") koşulları ilk defa oluşursa, zaman hatası interrupt'ı OB'nin bulunmasına bakılmaksızın CPU sonra STOP'a geçer. "S7-1200 Sistem Kılavuzunda döngü süresini izleme" (Sayfa 97) bölümüne bakınız.

Zaman hatası interrupt'ı OB'si başlama bilgisini içerir. Bu bilgi zaman hatasını hangi olayın ve OB'nin ürettiğini belirlemenize yardımcı olur. Bu başlama değerlerini test etmek ve uygun aksiyonu yapmak için OB içindeki komutları programlayabilirsiniz.

Çizelge 4-5 Zaman hatası OB (OB 80) için başlama bilgisi

Giriş	Veri tipi	Açıklama
fault_id	BYTE	16#01 - maksimum döngü süresi aşıldı 16#02 – talep edilen OB başlatılamaz 16#07 ve 16#09 – kuyruk aşırı akımı oluştu
csg_OBnr	OB_ANY	Hata oluştuğunda yürütülmekte olan OB numarası
csg_prio	UINT	OB'nin önceliği bir hataya sebep oluyor

Projenizde bir zaman hatası interrupt'ı OB eklemek isterseniz, ağaçtaki "Program blocks" altında bulunan "Yeni blok ekle" ("Add new block") üzerine çift tıklayıp, daha sonra da sırasıyla "Organizasyon bloğu" ("Organization block"), ve "Zaman hatası interrupt'ı" ("Time error interrupt") seçimlerini yaparak bir zaman hatası interrupt'ı ekleyebilirsiniz.

Yeni V4.0 CPU için öncelik 22'dir. Bir V3.0 CPU'yu bir V4.0 CPU (Sayfa 995) ile değiştirirseniz öncelik 26 olur, öncelik V3.0 için etkin durumdaydı. Her iki durumda da öncelik alanı düzenlenebilir ve siz önceliği 22-24 aralığında bir değere ayarlayabilirsiniz.

#### 4.1.3.7 Diyagnostik (tanılama) hata interrupt'ı OB

CPU bir diyagnostik hata algıladığında veya bir diyagnostik- yetenekli modül bir hatayı tanırca, diyagnostik hata interrupt'ı OB yürütür ve modül için diyagnostik hata interrupt'ını siz etkinleştirdiniz. Diyagnostik hata interrupt'ı OB, normal döngüsel program yürütmesini keser. Bu tip hata alınması üzerine CPU'nuzun STOP moduna girmesini isterseniz, CPU'yu STOP moduna koymak için diyagnostik hata interrupt OB'si için bir STP komutu dahil edebilirsiniz.

Programınıza bir diyagnostik hata interrupt OB'si dahil etmezseniz, CPU hatayı ihmal eder ve RUN modunda kalır.

**Diyagnostik (tanılama) hata olayları**

Analog (lokal), PROFINET, PROFIBUS ve bazı dijital (lokal) cihazlar diyagnostik hataları algılama ve raporlama yeteneğine sahiptirler. Birçok farklı diyagnostik hata koşullarının her hangi birisinin oluşması ve kaldırılması bir diyagnostik hata olayı ile sonuçlanır. Aşağıdaki diyagnostik hatalar desteklenir:

- Kullanıcı gücü yok
- Yüksek sınır aşıldı
- Düşük sınır aşıldı
- Kablo kopması
- Kısa devre

Diyagnostik hata olayları, var ise, diyagnostik hata interrupt OB'sinin (OB 82) yürütülmesini tetikler. O yok ise, sonra CPU hatayı ihmal eder.

Projenize bir diyagnostik hata interrupt'ı OB'si dahil etmek için ağaçtaki "Program blokları" ("Program blocks") altında "Ekle yeni blok"a ("Add new block") çift tıklayarak bir diyagnostik hata interrupt'ı eklemelisiniz, sonra "Organizasyon bloğu"nu ve sonra "diyagnostik hata interrupt'ı" ("Diagnostic error interrupt") seçmelisiniz.

**Not****Çok kanallı analog cihazlar (I/O, RTD ve Termokupl) için diyagonal hatalar**

Diyagnostik hata interrupt'ı OB'si, bir seferde sadece bir kanalın diyagnostik hatasını rapor edebilir.

Çok-kanallı bir cihazın iki kanallı bir hataya sahip ise, aşağıdaki koşullar altında ikinci hata sonra sadece diyagnostik hata interrupt'ı OB'sini tetikler. Koşullar: ilk kanal hatası temizlenir, ilk hata tetiklemesinin tamamlandığı diyagnostik hata interrupt'ı OB'sinin yürütülmesi ve ikinci hata hala mevcuttur.

Diyagnostik hata interrupt'ı OB'si başlama bilgisini içerir. Bu bilgi, bir olayın bir hatanın oluşması veya kaldırılması yüzünden olup olmadığını ve hatayı raporlayan cihaz ve kanalı tanımlamanızda size yardımcı olur. Bu başlangıç değerlerini denemek ve uygun aksiyonu almak için diyagnostik hata interrupt'ı OB'si içine komutlar programlayabilirsiniz.

**Not****Diyagnostik hata OB başlama bilgisi, diyagnostik olayı beklemesi yok ise bir bütün olarak alt modüle başvurur.**

V3.0'de, devam eden diyagnostik hata olayı için başlama bilgisi daima olayın kaynağını gösterirdi. V4.0'de, bekleme diyagnostiği olmadan devam eden olay alt modülü terk ederse, olayın kaynağı belirli bir kanal olsa bile başlama bilgisi bir bütün olarak (16#8000) alt modüle başvurur.

Örneğin, bir kablo kopması kanal iki üzerinde bir diyagnostik hata olayını tetiklerse, arız sonra düzeltilir ve diyagnostik hata olayı temizlenir, başlama bilgisi kanal 2'ye başvurmaz, ancak alt modüle başvurur(16#8000).

Çizelge 4- 6 Diyagnostik hata interrupt'ı OB'si için başlangıç bilgisi

Giriş	Veri tipi	Açıklama
IState	WORD	Cihazın durumu: <ul style="list-style-type: none"> <li>• Bit 0 = 1 konfigürasyon doğruysa, ve = 0 konfigürasyon artık doğru değilse.</li> <li>• Bit 4 = 1 bir hata var ise (bir kablo kopması gibi). (Bit 4 = 0 hata yok ise)</li> <li>• Bit 5 = 1 konfigürasyon doğru değilse, ve = 0 konfigürasyon tekrar doğruysa.</li> <li>• Bit 6 = 1 Bir I/O erişim hatası oluşmuş ise. Erişim hatası olan I/O'un donanım tanımlayıcısı için LADDR'ye bakınız. (Bit 6 = 0 hata yok ise)</li> </ul>
LADDR	HW_ANY	Cihazın donanım tanımlayıcısı veya hatayı rapor etmiş fonksiyonel ünite <sup>1</sup>
Channel	UINT	Kanal numarası
MultiError	BOOL	TRUE, birden fazla hata var ise

<sup>1</sup> LADDR girişi, cihazın donanım tanımlayıcısını veya hatayı veren fonksiyonel üniteyi içerir. Donanım tanımlayıcısı, bileşenler cihaza veya ağ görünümüne eklendiğinde otomatik olarak atanır ve PLC etiketlerinin Sabitler sekmesinde görünür. PLC etiketlerinin Sabitler sekmesindeki bu girişler değiştirilemez.

#### 4.1.3.8 Modüller için çek veya tak OB

“Modüller için çek ve tak” (“Pull or plug of modules”) OB'si, konfigüre edilmiş ve devre dışı bırakılmış dağıtımlı I/O modülü veya alt modül (PROFIBUS, PROFINET, AS-i) bir modülü eklemeye veya çıkarmaya ilişkin bir olayı ürettiğinde yürütme gerçekleştirir.

#### Modüller için çek veya tak olayı

- Aşağıdaki koşullar modüller için çek veya tak olayını üretir:
- Birisi konfigüre edilmiş bir modülü çıkarır veya takar.
- Konfigüre edilmiş bir modül genişleme şasisinde fiziksel olarak yoktur.
- Eksik bir modül bir genişleme şasisindedir, ancak konfigüre edilmiş bir modüle karşılık değildir.
- Konfigüre edilmiş bir modül için uyumlu bir modül bir genişleme şasisindedir, ancak konfigürasyon yer değişikliklerine izin vermez.

Bu OB'yi programlamamış iseniz, CPU bu koşullardan herhangi biri oluştuğunda STOP moduna geçer.

Çizelge 4-7 Modüllerin çek veya tak OB'si için başlatma bilgisi

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım tanımlayıcı
Event_Class	Byte	16#38: modül takılı 16#29: modül çıkık
Fault_ID	Byte	Hata tanımlayıcı

#### 4.1.3.9 Şasi veya istasyon arızası OB

Şasi veya istasyon arızası OB'si ("Rack or station failure"), bir dağıtımlı şasi veya istasyonun haberleşme arızası veya kaybı meydana geldiğinde yürütmeyi yapar.

#### Şasi veya istasyon arızası olayı

CPU aşağıdakilerden birini algıladığında bir şasi veya istasyon arızası olayı üretir:

- Bir DP master sistemi veya bir PROFINET IO sisteminin arızası (bir gelen veya bir giden olayı durumunda).
- Bir DP slave veya bir I/O cihazının arızası (bir gelen veya bir giden olayı durumunda).
- Bir PROFINET I-cihazı bazı alt modüllerinin arızası

Bu OB'yi programlamamış iseniz, CPU bu koşullardan herhangi biri oluştuğunda STOP moduna geçer.

Çizelge 4-8 Şasi veya istasyon arızaları OB için başlama bilgisi

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım tanımlayıcı
Event_Class	Byte	16#32: Bir DP slave veya bir IO cihazının etkinleştirilmesi 16#33: Bir DP slave veya bir IO cihazının etkisizleştirilmesi 16#38: giden olay 16#39: gelen olay
Fault_ID	Byte	Hata tanımlayıcı

#### 4.1.3.10 Gün içindeki saat OB

Gün içindeki saat OB'leri konfigüre edilen saat zamanı koşullarına dayanarak yürütülür. CPU, iki gün içindeki saat OB'sini destekler.

### Gün içindeki saat olayları

Gün içindeki saat interrupt olayını belirli bir tarih veya zaman veya aşağıdaki döngülerin biriyle döngüsel olarak konfigüre edebilirsiniz:

- Dakikalık: Interrupt her dakikada oluşur.
- Saatlik: Interrupt her saatte oluşur
- Günlük: Interrupt her günde belirli bir zamanda oluşur (saat ve dakika)
- Haftalık: Interrupt her hafta haftanın belirli bir gününde belirli bir zamanda oluşur (örneğin her Salı öğleden sonra saat 4.30'da)
- Aylık: Interrupt her ayda ayın belirli bir gününde belirli bir zamanda oluşur (örneğin gün sayısı 1-28 dahil arasında olmalıdır)
- Her ay sonunda: Interrupt her ayın son gününde belirli bir zamanda oluşur.
- Yıllık: Interrupt her yıl belirli bir tarihte oluşur (ay ve gün). 29 Şubat tarihini belirleyemezsiniz.

Çizelge 4- 9 Gün içindeki saat olayı OB'si için başlama bilgisi

Giriş	Veri tipi	Açıklama
CaughtUp	Bool	OB çağırma yakalanır, çünkü zaman ileri alınmıştır
SecondTimes	Bool	OB çağırma başlatılır, çünkü zaman geri alınmıştır

### 4.1.3.11 Durum OB

Bir DPV1 veya PNIO slave bir durum interrupt'ını tetiklerse, durum OB'leri yürütme yapar. Bir DPV1 veya PNIO slave'in bir bileşeni (modül veya şasi) işletim modunu değiştirirse (örneğin RUN'dan STOP'a), bu bir durum olabilir.

### Durum olayları

Bir durum interrupt'ını tetikleyen olaylar hakkında ayrıntılı bilgi için imalatçının DPV1 veya PINO slave dokümanına bakınız.

Çizelge 4- 10 Durum OB için başlama bilgisi

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım tanımlayıcı
Slot	UInt	Yuva numarası
Specifier	Word	Alarm tanımlayıcı

### 4.1.3.12 Güncelleme OB

Bir DPV1 veya PNIO slave bir güncelleme interrupt'ını tetiklerse, güncelleme OB'leri yürütme yapar.

### Güncelleme olayları

Bir güncelleme interrupt'ını tetikleyen olaylar hakkında ayrıntılı bilgi için imalatçının DPV1 veya PINO slave dokümanına bakınız.

Çizelge 4- 11 Güncelleme OB için başlama bilgisi

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım tanımlayıcı
Slot	UInt	Yuva numarası
Specifier	Word	Alarm tanımlayıcı

### 4.1.3.13 Profil OB

Bir DPV1 veya PNIO slave bir profile özel interrupt'ı tetiklerse, Profil OB'ler yürütme yapar.

### Profile olayları

Bir profil interrupt'ını tetikleyen olaylar üzerine ayrıntılı bilgi için DPV1 veya PINO imalatçı dokümanlarına bakınız.

Çizelge 4- 12 Profile OB için başlama bilgisi

Giriş	Veri tipi	Açıklama
LADDR	HW_IO	Donanım tanımlayıcı
Slot	UInt	Yuva numarası
Specifier	Word	Alarm tanımlayıcı

### 4.1.3.14 Olay yürütme öncelikleri ve kuyruklama

CPU işlemi, olaylar tarafından kontrol edilir. Bir olay, yürütülecek bir interrupt OB'sini tetikler. Bir olay için interrupt OB'sini bloğun oluşturulması sırasında, cihaz yapılandırması sırasında veya bir ATTACH veya DETACH komutu ile belirleyebilirsiniz. Bazı olaylar, program döngüsü veya döngüsel olaylar gibi düzenli esaslar üzerine olur. Başlangıç olayları ve zaman gecikme olayları gibi diğer bazı olaylar sadece tek bir kere olur. Bir girişe noktası üzerindeki bir kenar olayı veya yüksek hızlı bir sayıcı olayı gibi bazı olaylar, donanım bir olayı tetiklendiğinde olur. Ayrıca sadece bir hata oluştuğunda, diyagnostik hatası ve zaman hatası olayı gibi olaylar meydana gelir. Olay interrupt OB'leri için işlem sırasını belirlemek amacıyla olay öncelikleri ve kuyruklar kullanılır.

CPU olayları öncelik sırasına göre işler. Burada 1 en düşük öncelik, 26 en yüksek öncelik. S7-1200 CPU V4.0'den önce, her tip OB sabit öncelik sınıfına aitti (1-26). V4.0 ile konfigüre ettiğiniz her OB'ye bir öncelik sınıfı atayabilirsiniz. OB özelliklerinin özneliklerinde öncelik numarasını konfigüre edebilirsiniz.

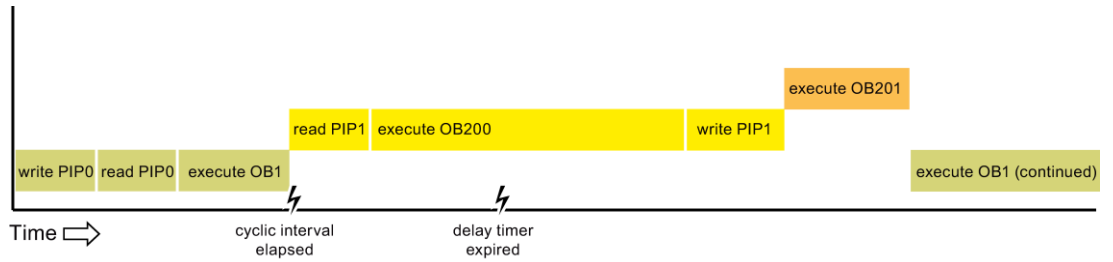
## Kesilebilir veya kesilemez yürütme modları

OB'ler (Sayfa 85) kendilerini tetikleyen olayları öncelik sırasına göre yürütür. S7-1200 CPU'nun V4.0 versiyonu ile, OB yürütmesini kesilebilir veya kesilemez olacak şekilde konfigüre edebilirsiniz. Program döngüsü OB'lerin daima kesilebilir olduğuna dikkate ediniz, ancak diğer bütün OB'leri kesilebilir veya kesilemez olacak şekilde konfigüre edebilirsiniz.

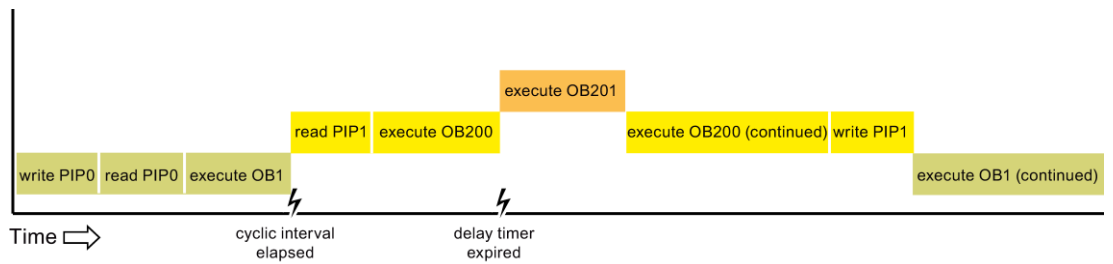
Kesilebilir moda ayarladıysanız, sonra OB kendi yürütmesini tamamlamadan önce bir OB yürütme yapıyor ve yüksek öncelikli bir olay meydana gelirse, yürütmedeki OB yüksek öncelikli olay OB'sini işletmek için kesilir. Yüksek öncelikli olay yürütülür ve tamamlanmasında, kesilen OB devam eder. Bir kesilebilir OB yürütülüyor iken çoklu olaylar oluşursa, CPU çoklu olayları öncelik sırasına göre işler.

Kesilebilir moda ayarlamadıysanız bir OB kendisinin çalışması boyunca tetikleyen diğer olaylara bakılmaksızın tetiklendiğinde tamamlama için çalışır. İnterrupt olayların bir döngüsel OB ve bir zaman gecikme OB'sini tetiklediği izleyen iki durumu göz önünde bulundurunuz.

İki durumda da zaman gecikme OB'si (OB201) proses görüntü bölüntü atamasına sahip değil (Sayfa 77) ve öncelik 4'te yürütülür. Döngüsel OB'si (OB200), PIP1 proses görüntü bölüntü atamasına sahip ve öncelik 2'de yürütülür. Aşağıdaki resimlemeler kesilebilir ve kesilemez yürütme modları arasındaki farkı gösterir:



Şekil 4-1 Durum 1: Kesilemeyen OB yürütmesi



Şekil 4-2 Durum 2: Kesilebilir OB yürütmesi

## Not

OB yürütme modunu kesilemez olacak şekilde konfigüre ederseniz, bir zaman hata OB'si program döngü OB'lerinden başka OB'leri kesemez. S7-1200 CPU, V4.0'dan önce bir zaman hata OB'si herhangi bir yürütme OB'sini kesebilirdi. V4.0'da, program döngü OB'si olmayan yürütme OB'lerini kesilemek için bir zaman hata OB'si isterseniz, OB yürütmesini kesilebilir olacak şekilde konfigüre etmelisiniz.

### Olay yürütme önceliklerinin ve kuyruklamanın anlaşılması

Tek bir kaynaktan gelen bekleyen (kuyruktaki) olayların sayısı her olay tipi için farklı bir kuyruk kullanılarak sınırlanır. Verilen olay tipi için bekleyen olay sınırına ulaşıldığında bir sonraki olay kaybedilir. Kuyruk taşmaları üzerine daha fazla bilgi için zaman hatası interrupt OB'leri (Sayfa 88) hakkındaki konulara bakınız.

Her bir CPU olayı ilişkilendirilmiş bir önceliğe sahiptir. Genellikle CPU, olayları öncelik sırasına göre (en yüksek öncelik ilk olmak üzere) servis eder. CPU, aynı önceliğe sahip olayları "ilk gelen ilk servis edilir" esasına göre servis eder.

Çizelge 4- 13 OB olayları

Olay	İzin verilen miktar	Varsayılan OB önceliği
Program döngüsü	1 program döngü olayı Çoklu OB'lere izin verilir	1 <sup>4</sup>
Başlangıç	1 başlangıç olayı <sup>1</sup> Çoklu OB'lere izin verilir	1 <sup>4</sup>
Zaman gecikmesi	4 defaya kadar olaylar 1 OB olay başına	3
Döngüsel interrupt	4 defaya kadar olaylar 1 OB olay başına	8
Donanım interrupt'ı	50 donanım interrupt olayına kadar <sup>2</sup> 1 OB olay başına, ancak çoklu olaylar için aynı OB'yi kullanabilirsiniz	18 18
Zaman hatası	1 olay (sadece konfigüre edilmiş ise) <sup>3</sup>	22 veya 26 <sup>4</sup>
Diagnostik hata	1 olay (sadece konfigüre edilmiş ise)	5
Çek veya tak modülleri	1 olay	6
Şasi ve istasyon arızası	1 olay	6
Gün içindeki saat	2 defaya kadar olaylar	2
Durum	1 olay	4
Güncelleme	1 olay	4
Profil	1 olay	4

<sup>1</sup> Başlangıç olayı ve program döngüsü olayı asla aynı anda meydana gelmez, çünkü başlangıç olayı, program döngüsü olayı başlatılmadan önce, tamamlamak üzere çalışmaktadır.

<sup>2</sup> DETACH ve ATTACH komutlarını kullanırsanız 50 donanım interrupt olayından daha fazlasına sahip olabilirsiniz.

<sup>3</sup> Tarama döngüsü maksimum tarama döngüsü süresini aşmışsa, CPU'yu RUN modunda kalmak üzere yapılandırabilirsiniz veya döngü süresini sıfırlamak için RE\_TRIGR komutunu kullanabilirsiniz. Ancak bir tarama döngüsü maksimum tarama süresini aştığında CPU ikinci kez STOP moduna geçer.

<sup>4</sup> Yeni V4.0 CPU için öncelik numarası 22'dir. V3.0 CPU'yu V4.0 CPU ile değiştirirseniz, öncelik 26 olur, V3.0 için etkin olan öncelik 26 idi. Her iki durumda da, öncelik alanı düzenlenebilir. Önceliği, 22-26 aralığında bir değere ayarlayabilirsiniz.

Daha fazla ayrıntı için "V3.0 CPU'yu V4.0 CPU ile değiştirme (Sayfa 995)" konusuna bakınız.



İlave olarak, CPU, OB'lerle ilişkisi olmayan diğer olayları tanır. Aşağıdaki Çizelge bu olayları ve karşılık gelen CPU aksiyonlarını açıklar:

Çizelge 4- 14 İlave olaylar

Olay	Açıklama	CPU aksiyonu
I/O erişim hatası	Direkt I/O okuma/yazma hatası	CPU ilk meydana gelişi diyagnostik ara belleğinde tutar ve RUN modunda kalır.
Max döngü süresi hatası	CPU, konfigüre edilene döngü süresini iki kere aşmış	CPU hatayı diyagnostik ara belleğinde tutar ve STOP moduna geçer.
Çevre birimi erişim hatası	I/O hatası, proses görüntü güncellemesi sırasında	CPU ilk meydana gelişi diyagnostik ara belleğinde tutar ve RUN modunda kalır.
Programlama hatası	Program yürütme hatası	Hataya sahip blok hata işleme üretir, hata yapısını günceller; Üretmezse, CPU hatayı diyagnostik ara belleğinde tutar ve RUN modunda kalır.

### İnterrupt gecikmesi

İnterrupt olay gecikmesi (CPU'nun bir olayın oluştuğundan haberdar edilmesinden CPU'nun o olayı servis eden OB'deki ilk komutu yürütmeye başlatmasına kadar geçen süre), yaklaşık olarak 175 mikro saniyedir. Bu süre, bir program döngüsü OB'sinin interrupt olayı süresinde sadece olay servis rutini aktivitesi olması durumu için geçerlidir.

#### 4.1.4 Döngü süresinin konfigürasyonu ve izlenmesi

Döngü süresi, RUN modunun döngüsel safhasını yürütmek için CPU işletim sisteminin gerek duyduğu süredir. CPU, döngü süresi izlemesinin iki yöntemini sağlar:

- Maksimum tarama döngü süresi
- Minimum tarama döngü süresi

Tarama döngü izlemesi, başlangıç olayı tamamlandıktan sonra başlar. Bu özelliğin konfigürasyonu CPU için "Cihaz konfigürasyonu" altındaki "Döngü Süresi" altındadır.

CPU daima döngü süresini izler ve maksimum tarama süresi aşılsa, tepki verir. Konfigüre edilen maksimum tarama döngü süresi aşılsa, bir hata üretilir ve iki yolun birisiyle işlenir:

- Kullanıcı programı bir zaman hata interrupt OB'si içermez ise, zamanlayıcı hata olayı bir diyagnostik ara belleği girişi üretir, ancak CPU RUN modunda kalır. (CPU bir zaman hatası algılar, ancak varsayılan konfigürasyon RUN modunda kalacak şekilde ise, CPU'nun konfigürasyonunu STOP moduna geçecek şekilde yapabilirsiniz.)
- Kullanıcı programı bir zaman hata interrupt OB'si içerir ise, CPU onu yürütür.

RE\_TRIGR komutu (Sayfa 262) (tekrar-tetikleyici döngü süresi izleme) döngü süresini ölçen zamanlayıcıyı resetlemenize izin verir. Geçerli program döngü yürütmesi için geçen süre konfigüre edilen maksimum tarama döngü süresinden 10 kez daha az ise, RE\_TRIGR komutu döngü süre izlemesini tekrar tetikler ve ENO = TRUE ile döner. Değilse, RE\_TRIGR komutu döngü süre izlemesini tekrar tetiklemez ve ENO = FALSE durumuna döner.

#### 4.1 Kullanıcı programının yürütülmesi

Tipik olarak, tarama döngüsü yürütülebildiği kadar hızlı yürütür ve sonraki tarama döngüsü geçerli olan birisi tamamlanır tamamlanmaz başlar. Kullanıcı programı ve haberleşme görevlerine bağlı olarak tarama döngüsünün zaman periyodu taramaya değişir. Bu değişimi engellemek için CPU opsiyonel bir minimum tarama döngüsü süresini sağlar. Bu opsiyonel özelliğini etkinleştirir ve ms biriminde bir minimum tarama döngü süresi sağlayabilirseniz, program döngüsü tekrarlanmadan önce minimum tarama döngü süresi geçinceye kadar program döngüsü yürütüldükten sonra CPU geciktirir.

CPU'nun normal tarama döngüsünü belirlenen minimum döngü süresinden daha kısa sürede tamamlaması durumunda, CPU tarama döngü gerçekleştirme çalışma süresi diyagnostikleri ve/veya haberleşme taleplerini işleme için ilave süre harcar.

CPU'nun normal tarama döngüsünü belirlenen minimum döngü süresinde tamamlamaması durumunda, CPU taramayı normal olarak tamamlar (haberleşme işlemi de içerir) ve minimum tarama süresinin aşılmasının sonucu olarak herhangi bir sistem tepkisi oluşturmaz. İzleyen Çizelge döngü süresi izleme fonksiyonları için aralıkları ve varsayılanları tanımlar.

Çizelge 4- 15 Döngü süresi için aralık

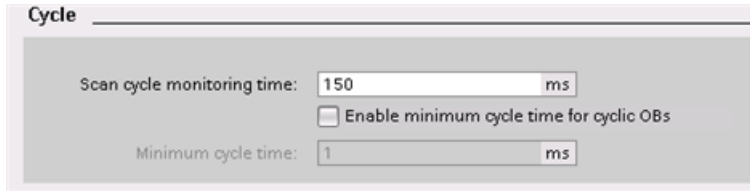
Döngü süresi	Aralık (ms)	Varsayılan
Max döngü süresi <sup>1</sup>	1 - 6000	150 ms
Minimum tarama döngü süresi <sup>2</sup>	1 - maksimum tarama döngü süresi	Etkisiz, disabled

- 1 Maksimum tarama döngü süresi daima etkindir. 1 ms-6000 ms arasında bir döngü süresi konfigüre ediniz. 150 ms varsayılandır.
- 2 Minimum tarama döngü süresi opsiyoneldir ve varsayılan etkin değildir (disabled). Gerekirse, döngü süresini 1 ms ile maksimum tarama döngü süresi arasında konfigüre edebilirsiniz.

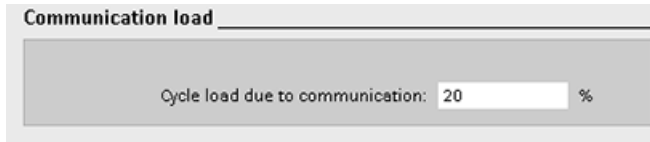
#### Döngü süresi ve haberleşme yükü konfigürasyonu

Cihaz konfigürasyonunda yer alan CPU özelliklerini aşağıdaki parametreleri konfigüre etmek için kullanabilirsiniz:

- Döngü: Bir maksimum tarama döngü izleme süresini siz girebilirsiniz. Bir minimum tarama döngü süresini de siz etkinleştirebilirsiniz.



- Haberleşme yükü: Haberleşme görevleri için özel atanan sürenin yüzdesini konfigüre edebilirsiniz.



Döngü süresi hakkında daha fazla bilgi için "Döndü süresinin izlenmesi"ne bakınız (Sayfa 97).

## 4.1.5 CPU belleği

### Bellek yönetimi

CPU; kullanıcı programı, veriler ve konfigürasyonun saklanması için aşağıda verilen bellek alanlarını sağlamaktadır:

- **Yük belleği;** kullanıcı programı, veriler ve konfigürasyon için kalıcı bir bellektir. Bir projeyi CPU'ya yüklediğinizde, CPU programı önce yük belleği alanında saklar. Bu alan, bir bellek kartında (eğer varsa) veya CPU'da yer alır. Enerji kesilmelerinde, CPU bu kalıcı bellek alanını muhafaza eder. Hafıza kartı, CPU'da yerleşik olandan daha büyük bir depolama alanını destekler.
- **Çalışma belleği,** kullanıcı programının yürütülürken kullanıcı projesinin bazı elemanları için kalıcı olmayan bellektir. CPU, projenin bazı elemanlarını yükleme belleğinden çalışma belleğine kopyalar. Bu geçici alan enerji kesildiğinde silinir ve enerji geri geldiğinde CPU tarafından kurtarılır.
- **Kalıcı bellek,** çalışma belleği değerlerinin sınırlı bir miktarı için kalıcı bir depodur. CPU, kalıcı bellek alanını seçilen kullanıcı belleği alanlarının değerlerini enerji kesilmesi durumunda saklamak için kullanır. Enerji kesildiğinde ve yeniden geldiğinde CPU bu kalıcı değerleri tekrar yükler.

Derlenen bir program için bellek kullanımını göstermek üzere STEP 7'de "Program Blokları" içinde bloka sağ tıklayınız ve menü içeriğinden "kaynakları" seçiniz. Derleme özellikleri yükleme belleğini ve derlenen blok için çalışma belleğini gösterir.

Geçerli CPU için bellek kullanımını çevrimiçi görüntülemek için STEP 7'de "Online ve Diyagnostik" üzerine çift tıklayınız, diyagnostiği genişletiniz ve Belleği seçiniz.

## Kalıcı bellek

Elektrik kesintisi sonrası veri kaybı, bazı verilerin kalıcı olarak işaretlenmesiyle engellenebilir. CPU, aşağıda verilen verileri kalıcı olarak konfigüre etmenize izin verir:

- Bit hafızası (M): PLC etiket çizelgesindeki veya atama listesindeki bit hafızası için belleğin kesin genişliğini tanımlayabilirsiniz. Kalıcı bit hafızası her zaman MBO'da başlar ve belirli bir bayt sayısı boyunca art arda çalışır. Bu değeri PLC etiket çizelgesinden veya atama listesinden "Retain" araç çubuğu simgesine tıklamak suretiyle belirleyebilirsiniz. Kalıcı alanı seçmek için MBO'dan başlanacak şekilde M bayt sayısını giriniz.
- Bir fonksiyon bloğunun (FB) etiketleri: Bir FB "Optimized" kutucuğu işaretlenerek oluşturulmuşsa, bu FB için ara yüz editörü bir "Retain" sütunu içerir. Bu sütunda, her bir etiket için bağımsız olarak "Retain" veya "Non-Retain" veya her bir etiket için ayrı ayrı "Set in IDB" seçimini yapabilirsiniz. Bu FB program editörüne yerleştirilirken oluşturulan bir instance DB'si de bu retain sütununu içerir, ancak sadece görüntüler. Optimize edilen FB içinde etiket için Retain seçiminde "Set in IDB" (instance veri bloğu içinde ayarlama)'yı seçmişseniz, instance DB ara yüzü içinden bir etiketin kalıcı durumunu değiştirebilirsiniz.

Bir FB, "Standard - compatible with S7-300/400" seçilerek oluşturulmuşsa, o zaman bu FB için ara yüz editörü bir "Retain" sütunu içermez. Bu FB program editörüne eklendiğinde oluşturulan bir instance DB'si bir "Retain" sütunun gösterir. Bu sütunu editörün düzenlemesi mümkündür. Bu durumda herhangi bir etiket için "Retain" seçeneğinin seçilmesi, tüm etiketlerin seçilmesiyle sonuçlanır. Buna benzer bir şekilde herhangi bir etiket için bu seçeneğin geri alınması, tüm etiketlerin seçiminin geri alınmasıyla sonuçlanır. "Standard - compatible with S7-300/400" olacak şekilde konfigüre edilmiş bir FB için, kalıcı durumu instance DB editöründen değiştirebilirsiniz, ancak bu durumda tüm etiketler aynı kalıcı duruma ayarlanmış olur.

FB'yi oluşturduktan sonra, "Standard - compatible with S7-300/400" için seçeneğini değiştiremezsiniz. Bu seçeneği sadece FB'yi oluşturduğunuzda seçebilirsiniz. Mevcut bir FB'nin "Optimized" veya "Standard - compatible with S7-300/400" için yapılandırılmış olup olmadığını belirlemek için Proje ağacından FB'ye sağ tıklayınız, "Properties" ve sonra "Attributes" seçiniz. "Optimized block access" seçildiğinde kontrol kutusu bir bloğun optimize edilip edilmediğini gösterir.

- Bir global veri bloğunun etiketleri: Bir global DB'nin kalıcı durum atamasına bağlı davranışı, bir FB'nin davranışına benzerdir. Blok erişim ayarlamasına göre kalıcı durumu müstakil etiketler veya bir global veri bloğunun tüm etiketleri için tanımlayabilirsiniz.
  - DB oluştururken "Optimized" komutunu seçtiyseniz, her bir münferit etiket için kalıcı durum ayarlayabilirsiniz.
  - DB oluştururken "Standard - compatible with S7-300/400" seçtiyseniz, bütün etiketlerin kalıcı olmasına veya hiç birinin kalıcı olmasına bakılmaksızın DB'nin bütün etiketlerine kalıcı durum ayarı uygulanır.

CPU, toplam 10240 bayt kalıcı veriyi destekler. PLC etiket Çizelgesinden veya atama listesinden ne kadar kaldığını görmek için "Retain" araç çubuğu simgesine tıklayınız. M belleği için kalıcı aralığının belirlendiği yerde ikinci satır M ve DB bileşimi için toplam kalan belleği gösterir. Bu değer doğru olabilmesi için kalıcı etiketli bütün veri bloklarını derlemeniz gerektiğine dikkat ediniz.

#### Not

Bir programın yüklenmesi kalıcı hafızada var olan değerlerde her hangi bir değişikliği silmez veya yapmaz. Bir yükleme öncesi kalıcı bir belleği silmek isterseniz, programı yüklemeye önce CPU'nuzu fabrika ayarlarına reset ediniz.

#### 4.1.5.1 Sistem ve darbe jeneratör belleği

"Sistem belleği" ve "darbe jeneratörü belleği" ("clock memory") için baytları etkinleştirmek üzere CPU özelliklerini kullanabilirsiniz. Program mantığınız bu fonksiyonların münferit bitlerini kendi etiket isimleriyle kullanabilir.

- M hafızasında sistem belleği için bir bayt atayabilirsiniz. Sistem belleğinin baytı, kullanıcı programınız tarafından aşağıdaki etiket isimleriyle kullanılacak dört biti sağlar:
  - İlk döngü: ("FirstScan" için etiket ismi) bit'i başlangıç OB'si tamamlandıktan sonra ilk taramanın süresi için 1'e set edilir (İlk taramanın yürütülmesinden sonra, "first scan" biti 0'a set edilir).
  - Diyagnostik durumu değiştirildi (Tag name: "DiagStatusUpdate"), CPU bir diyagnostik olayını sıralı kayıt yaptıktan sonra, bir tarama için 1'e set edilir. CPU, "diyagnostik grafiği değişti" bitini program döngüsü OB'lerinin ilk yürütmesinin sonuna kadar ayarlamadığından, kullanıcı programınız eğer başlangıç OB'lerinin yürütülmesi ya da program döngüsü OB'lerinin ilk yürütülmesi sırasında bir diyagnostik değişikliği meydana gelmişse bunu tespit edemeyebilir.
  - Daima 1 (yüksek): (Etiket ismi "AlwaysTRUE") biti daima 1'e set edilir.
  - Daima 0 (düşük): (Etiket ismi "AlwaysFALSE") biti daima 0'a set edilir.
- Darbe jeneratörü belleği için M belleğine bir bayt atayabilirsiniz. Darbe jeneratörü belleği olarak konfigüre edilen baytın her bir biti, bir kare dalga darbesi üretir. Darbe jeneratörü belleğinin baytı, 0,5 Hz (yavaş) – 10 Hz (hızlı) arasında olmak üzere 8 farklı frekans sağlar. Bu bitleri kontrol bitleri olarak kullanabilirsiniz, özellikle kullanıcı programındaki aksiyonları döngüsel esasta tetiklemek için kenar komutları birleştirildiği durumda.

CPU, bu baytları STOP modundan STARTUP moduna geçişte başlangıç durumuna getirir. Darbe jeneratörü belleğinin bitleri, STARTUP ve RUN modları üzerinden senkronize olarak CPU darbe jeneratörüne değişir.

**⚠ DİKKAT**

**Sistem belleği veya darbe jeneratörü belleği bitlerinin üzerine yazma riskleri**

Sistem belleği ya da darbe jeneratörü belleği bitlerinin üzerine yazılması durumunda bu fonksiyonlardaki veriler bozulabilir ve kullanıcı programınızın hatalı bir şekilde çalışmasına neden olabilir. Bu da ekipmana zarar verebilir, personelin yaralanmasına neden olabilir.

Hem darbe jeneratörü belleğinin hem de sistem belleğinin M hafızası içinde rezerve edilmemiş olmasından, komutlar veya haberleşmeler bu konumlara yazma işlem yapabilir ve verileri bozabilirler.

Bu fonksiyonların düzgün bir şekilde çalışmasını sağlamak için bu konumlara veri yazmaktan kaçınınız ve her zaman işlemlerinizi veya makinanız için bir acil durum durdurma devresi sağlayınız.

Sistem belleği, özel bir olayı etkinleştiren (değer = 1) bitler ile bir baytı konfigüre eder

**System memory bits**

Enable the use of system memory byte

Address of system memory byte (MBx):

First cycle:

Diagnostics status changed:

Always 1 (high):

Always 0 (low):

Çizelge 4- 16 Sistem belleği

7	6	5	4	3	2	1	0
Rezerv edilmiş Değer 0			Daima off Değer 0	Daima on Değer 1	Diyagnostik durum gösterici		İlk tarama göstericisi
					<ul style="list-style-type: none"> <li>1: Değişme</li> <li>0: Değişme yok</li> </ul>		<ul style="list-style-type: none"> <li>1: Başlangıçtan sonra ilk tarama</li> <li>0: İlk tarama değil</li> </ul>

Darbe jeneratörü belleği, bağımsız bitleri sabit aralıklarla açıp kapatarak bir döngü oluşturan bir baytı yapılandırır. Her bir darbe jeneratörü biti, karşılık gelen M hafızası bitinde bir kare dalga darbesi üretir. Bu bitler kontrol bitleri olarak kullanılabilir, özellikle kullanıcı kodundaki hareketleri döngüsel bir esasa göre tetiklemek için kenar komutları ile birleştirildiğinde.

**Clock memory bits**

Enable the use of clock memory byte

Address of clock memory byte (MBx):

10 Hz clock:

5 Hz clock:

2.5 Hz clock:

2 Hz clock:

1.25 Hz clock:

1 Hz clock:

0.625 Hz clock:

0.5 Hz clock:

Çizelge 4- 17 Darbe jeneratörü belleği

Bit numarası	7	6	5	4	3	2	1	0
Etiket ismi								
Periyot (s)	2.0	1.6	1.0	0.8	0.5	0.4	0.2	0.1
Frekans (Hz)	0.5	0.625	1	1.25	2	2.5	5	10

Darbe jeneratörü belleği CPU döngüsüyle asenkronize olarak çalıştığından, darbe jeneratörü belleğinin durumu uzun bir döngü süresince birkaç kez değişebilir.

#### 4.1.6 Diyagnostik (tanılama) ara belleği

CPU, her bir diyagnostik olayı için bir giriş içeren bir diyagnostik ara belleğini destekler. Her bir giriş, olayın meydana geldiği tarihi ve zamanı, bir olay kategorisini ve bir olay açıklamasını içerir. Girişler kronolojik sırada, en yakın zamanlı olanı en üstte olacak şekilde görüntülenir. Bu kütükte, en son 50 olay tutulabilir. Kütük dolduğunda en son gelen olay kütükteki en eski olayın yerini alır. Enerji kesildiğinde, olaylar kaydedilir.

Diyagnostik ara belleğinde şu olay tipleri kaydedilir:

- Her sistem diyagnostik olayı; mesela CPU hataları ve modül hataları
- CPU'nun her durum değişimi (her enerji açılışı, her STOP'a geçiş, her RUN'a geçiş)

Diyagnostik ara belleğine (Sayfa 793) erişmek için çevrimiçi durumda olmalısınız. Online & Diyagnostik" görünümünden "Diagnostics > Diagnostics buffer" altına kütüğü konumlandırınız.

#### 4.1.7 Günün saati

CPU, bir günün saati desteğini sağlar. Bir süper kapasitör CPU enerjili değilken saatin çalışabilmesi için gerekli enerjiyi sağlar. Bu süper kapasitör, CPU'da enerji beslemesi varken şarj olur. CPU en az 24 saat boyunca enerjili kaldığında, süper kapasitör saatin tipik olarak 20 gün boyunca çalışması için yeterli şarja sahip olur.

STEP 7, günün saatini sistem saatine ayarlar. Bu bir kutudan veya izleyen fabrika resetinden bir varsayılan değere sahiptir. Günün saatini kullanabilmek için onu set etmeniz gerekir. Diyagnostik ara bellek girişleri, veri kütük dosyaları ve veri kütük girişlerinde olduğu gibi zaman bilgisi sistem saatine dayanır. Günün saatini online CPU'nun "Online & diagnostics" görünümündeki "Set günün saatini ayarla" fonksiyonundan (Sayfa 789) ayarlayınız. STEP 7 sonra sistem saatini sizin set ettiğiniz saate artı veya eksi olarak Windows İşletim Sisteminin UTC (Coordinated Universal Time) ofseti dikkate alarak hesaplar. Windows İşletim Sisteminin ayarlaması zaman bölgesi için ve yaz saatini yerel konunuza karşılık ise günün saatinin geçerli yerel saate ayarlanması UTC'nin sistem saatini üretir.

STEP 7 sistem saatini okumak ve yazmak için (RD\_SYS\_T ve WR\_SYS\_T) (Sayfa 286) ve local saati okumak için (RD\_LOC\_T) ve saat bölgesini set etmek için (SET\_TIMEZONE) komutlarını içerir. Yerel saat, CPU genel özelliklerindeki (Sayfa 242) "Günün saati" konfigürasyonunda set ettiğiniz saat bölgesi ve yaz saati ofsetini kullanarak RD\_LOC\_T komutu hesaplar. Bu ayarlamalar sizin lokal saat için saat bölgenizi ayarlamanızı ve yaz saati için başlama ve bitiş tarihleri ve saatlerini belirlemenizi mümkün kılar. Bu ayarlamaları konfigüre etmek için SET\_TIMEZONE komutunu da kullanabilirsiniz.

#### 4.1.8 Bir RUN – STOP geçişindeki çıkışların konfigürasyonu

CPU STOP modunda iken dijital ve analog çıkışların davranışını yapılandırabilirsiniz. Bir CPU'nun, SB'nin veya SM'nin herhangi bir çıkışı için çıkışları dondurabilir ya da çıkışlara geçici bir değer atayabilirsiniz:

- Belirli bir çıkış değerinin yerine geçmek (varsayılan): CPU, SB veya SM cihazının her bir çıkışı (kanalı) için geçici bir değer atayabilirsiniz.

Dijital çıkış kanalları için varsayılan geçici değer OFF (kapalı) şeklindedir. Analog çıkış kanalları için geçici değer ise 0'dır.

- Çıkışları son durumda kalmaları için dondurmak: Çıkışlar, RUN'dan STOP'a geçişte o anki değerlerinde kalırlar. Enerjilendikten sonra, çıkışlara varsayılan yedek değer atanır.

Çıkışların davranışını, Cihaz Konfigürasyonundan yapılandırabilirsiniz. Münferit cihazları seçiniz ve her bir cihaz için çıkışları yapılandırmak üzere "Özellikler" sekmesini kullanınız.

CPU, RUN'dan STOP'a geçtiğinde, CPU proses görüntüsünü sabit tutar ve hem dijital hem de analog çıkışlar için konfigürasyonda tanımlanan uygun değerleri yazar.



## 4.2 Veri depolama, bellek alanları, I/O ve adresleme

### 4.2.1 S7-1200'ün verilerine erişim

STEP 7, sembolik programlamayı kolaylaştırır. Verilerin adresleri için bellek adresleriyle ilgili PCL etiketleri ve I/O noktaları ya da bir kod bloğu içinde kullanılan lokal değişkenler olarak sembolik isimler veya etiketler ("tag") oluşturabilirsiniz. Bu etiketleri kullanıcı programında kullanmak için sadece komut parametrelerine etiket ismini giriniz.

CPU'nun bellek alanlarını nasıl yapılandığı ve adreslediğini daha iyi anlamak için aşağıda PLC etiketleri tarafından başvuru "mutlak" adresleme açıklanmıştır. CPU, kullanıcı programının yürütülmesi sırasında veri depolamak için birkaç seçenek sunar:

- Global bellek: CPU, çeşitli özelleştirilmiş bellek alanları sunar: girişler (I), çıkışlar (Q) ve bit belleği (M). Bu belleğe tüm kod blokları herhangi bir kısıtlama olmaksızın erişebilir.
- PLC etiket Çizelgesi: Belirli bellek yerleri için STEP 7 PLC etiket Çizelgesinde sembolik isimleri girebilirsiniz. Bu etiketler STEP 7 için globaldir ve uygulamanız için anlamlı olan isimler ile programlamaya izin verir.
- Veri bloğu (DB): Kod blokları için veri depolamada kullanıcı programınıza DB'ler ekleyebilirsiniz. Depolanan veriler, ilgili kod bloğunun yürütmesi sonlanana kadar kalır. Bir "global" DB, tüm kod blokları tarafından kullanılabilen verileri saklar, bir instance DB'si ise belirli bir FB için verileri depolar ve FB için parametreler tarafından yapılandırılır.
- Geçici bellek: Bir kod bloğunun her çağrılışında, CPU'nun işletim sistemi, bloğun yürütülmesi sırasında kullanılacak geçici ya da lokal hafızayı (L) tahsis eder. Kod bloğunun yürütülmesi bittiğinde, CPU lokal hafızayı diğer kod bloklarının yürütülmesi için tekrar tahsis eder.

Her bellek konumunun benzersiz bir adresi vardır. Kullanıcı programınız bu adresleri bellek konumundaki bilgilere erişmek için kullanır. Giriş (I) veya çıkış (Q) bellek alanlarına IO.3 veya Q1.7 gibi başvurular, proses görüntüsüne erişir. Fiziksel girişe veya çıkışa anında erişim, ":P" referansı ile ekleyiniz (IO.3:P, Q1.7:P veya "Stop:P" gibi)

Çizelge 4- 18 Bellek alanları

Bellek alanı	Açıklama	Kuvvet	Kalıcı
I Proses görüntü girişi	Tarama döngüsünün başlangıcındaki fiziksel girişlerden kopyalanır	Hayır	Hayır
I_:P <sup>1</sup> (Fiziksel giriş)	CPU, SB ve SM üzerindeki fiziksel giriş noktalarından anında okuma	Evet	Hayır
Q Proses görüntü çıkışı	Tarama döngüsünün başlangıcındaki fiziksel çıkışlardan kopyalanır	Evet	Hayır
Q_:P <sup>1</sup> (Fiziksel çıkış)	CPU, SB ve SM üzerindeki fiziksel giriş noktalarına anında yazma	Evet	Hayır
M Bit belleği	Kontrol ve veri belleği	Hayır	Evet (opsiyonel)



**I (proses imaj girişi):** CPU, çevresel (fiziksel) giriş noktalarını, her bir tarama döngüsünün döngüsel OB yürütmesinin hemen öncesinde örnekler ve bu değerleri giriş proses görüntüsüne yazar. Giriş proses görüntüsüne bit, bayt, word veya double word olarak erişebilirsiniz. Hem okuma hem de yazma erişimine izin verilir, ama tipik olarak proses görüntü girişleri sadece okunur.

Çizelge 4- 19 I hafızası için mutlak adresleme

Bit	I[bayt adresi].[bit adresi]	I0.1
Byte, Word veya Double Word	I[boyut][başlama bayt adresi]	IB4, IW5 veya ID12

Adrese bir “:P” ekleyerek CPU’nun, SB’nin veya SM’nin dijital ve analog girişlerini hemen okuyabilirsiniz. I yerine I\_:P kullanarak erişim sağlandığında, veriler giriş proses görüntüsünden gelmek yerine direkt olarak erişilen noktalardan gelir. Bu I\_:P erişimine “anında okuma” erişimi denir. Bunun sebebi verilerin giriş görüntüsünün güncellendiği son zamanda alınan bir kopyadan alınması yerine direkt olarak kaynaktan alınmasıdır.

Fiziksel giriş noktaları değerlerini direkt olarak bu noktalara bağlı olan saha cihazlarından aldığından, bu noktalara yazma yasaklanmıştır. Yani, I erişimleri hem okunur hem yazılabilirken, I\_:P erişimleri salt okunurdur.

I\_:P erişimleri aynı zamanda en yakın bayta yuvarlanmış şekilde tek bir CPU, SB veya SM tarafından desteklenen girişlerin boyutuyla da sınırlıdır. Örneğin eğer bir 2 DI / 2 DQ SB’nin girişleri 14.0’da başlamak üzere yapılandırılmışsa, o zaman giriş noktalarına I4.0:P ve I4.1:P ya da IB4:P olarak erişilebilir. I4.2:P’ye I4.7:P üzerinden erişimler reddedilmez, ancak bu noktalar kullanılmadığından bir anlam da ifade etmez. IW4:P’ye ve ID4:P’ye erişimler, bunlar SB ile ilişkilendirilmiş bayt ofsetini aştıklarından yasaklanır.

I\_:P kullanan erişimler, giriş proses görüntüsünde saklanan karşılık gelen değeri etkilemez.

Çizelge 4- 20 I hafızası için mutlak adresleme (anında)

Bit	I[bayt adresi].[bit adresi]:P	I0.1:P
Byte, Word veya Double word	I[boyut][başlama bayt adresi]:P	IB4:P, IW5:P veya ID12:P

**Q (proses görüntü çıkışı):** CPU, çıkış proses görüntüsünde saklanan değerleri fiziksel çıkış noktalarına kopyalar. Çıkış proses görüntüsüne bit, bayt, word veya double word olarak erişebilirsiniz. Proses görüntü çıkışları için hem okuma hem de yazma erişimine izin verilir.

Çizelge 4- 21 Q hafızası için mutlak adresleme

Bit	Q[bayt adresi].[bit adresi]	Q1.1
Byte, Word veya Double word	Q[boyut][başlama bayt adresi]	QB5, QW10, QD40

Adrese bir “:P” ekleyerek CPU’nun, SB’nin veya SM’nin dijital ve analog çıkışlarına anında yazabilirsiniz. Q yerine Q\_:P kullanılan bir erişimde veriler direkt olarak erişilen noktaya ve buna ek olarak çıkış proses görüntüsüne gider (iki yere de yazılır). Bu Q\_:P erişimine bazen “anında yazma” denir. Çünkü veriler hedef noktaya direkt olarak gönderilir; hedef noktası çıkış proses görüntüsünden gelecek bir sonraki güncelleme için beklemek zorunda değildir.

Fiziksel çıkış noktaları bu noktalara bağlı olan saha cihazlarını direkt olarak kontrol ettiklerinden, bu noktalardan okuma yasaktır. Yani, Q erişimleri hem okunur hem yazılabilirken, Q\_:P erişimleri salt yazılırdır.

Q\_:P erişimleri aynı zamanda en yakın bayta yuvarlanmış şekilde tek bir CPU, SB veya SM tarafından desteklenen çıkışların boyutuyla da sınırlıdır. Örneğin eğer bir 2 DI / 2 DQ SB'nin çıkışları Q4.0'da başlamak üzere yapılandırılmışsa, o zaman çıkış noktalarına Q4.0:P ve Q4.1:P veya QB4:P olarak erişilebilir. Q4.2:P'ye Q4.7:P üzerinden erişimler reddedilmez, ancak bu noktalar kullanılmadığından bir anlam da ifade etmez. QW4:P'ye ve QD4:P'ye erişimler, bunlar SB ile ilişkilendirilmiş bayt ofsetini aştıklarından yasaklanır.

Q\_:P kullanan erişimler hem fiziksel hem de çıkış proses görüntüsünde saklanan karşılık gelen değeri etkiler.

Çizelge 4- 22 Q hafızası için mutlak adresleme (anında)

Bit	Q[bayt adresi].[bit adresi]:P	Q1.1:P
Byte, Word veya Double word	Q[boyut][başlama bayt adresi]:P	QB5:P, QW10:P veya QD40:P

**M (bit belleği alanı):** Bit belleği alanını (M hafızası), operandın veya diğer kontrol bilgisinin anında durumlarını saklamak üzere hem kontrol röleleri hem de veriler için kullanınız. Bit belleği alanına bit, bayt, word veya double word olarak erişebilirsiniz. M hafızası için hem okuma hem de yazma erişimine izin verilir.

Çizelge 4- 23 M hafızası için mutlak adresleme

Bit	M[bayt adresi].[bit adresi]:	M26.7
Byte, Word veya Double Word	M[boyut][başlama bayt adresi]	MB20, MW30, MD50

**Temp (Geçici bellek):** CPU, temp belleğini gerekli olması durumuna göre tahsis eder. CPU, kod bloğu için temp belleğini, kod bloğu başlatıldığında (bir OB için) ya da çağırıldığında (bir FC ya da FB için) tahsis eder.

Temp belleği M hafızasına benzer ancak aralarında tek bir büyük fark bulunur: M hafızasının "global" bir kapsamı vardır, temp belleği ise "lokal" bir kapsama sahiptir:<

- M belleği: M hafızasındaki verilere herhangi bir OB, FC ya da FB erişilebilir. Yani veriler kullanıcı programının tüm elemanları için global olarak kullanılabilir.
- Temp belleği: Temp belleğindeki verilere erişim, temp belleği konumunda oluşturulmuş veya tanımlanmış OB, FC veya FB ile kısıtlanmıştır. Temp hafızası konumları lokal kalır ve kod bloğu başka bir kod bloğunu çağırıldığında bile farklı kod blokları tarafından paylaşılmaz. Bir OB bir FC'yi çağırıldığında, FC onu çağırın OB'nin temp belleğine erişemez.

CPU, üç OB öncelik gurubunun her biri için temp (lokal) hafıza sağlar:

- 16 kilobayt, program döngüsü ve başlatma için, ilgili FB'ler ve FC'ler dahil
- 6 kilobayt, her bir ilave interrupt olayı yürütmesi için, FB'ler ve FC'lerle ilgili olanlar dahil

Temp belleğine sadece sembolik adresleme ile erişebilirsiniz.

**DB (veri bloğu):** DB belleği, bir işlemin ara durumunu ya da FB'ler için diğer kontrol bilgileri parametreleri ve zamanlayıcılar ve sayıcılar gibi çoğu komut için gerekli veri yapıları dahil olmak üzere çeşitli veri türlerinin saklanması için kullanabilirsiniz.

Salt okunur data blokları için sadece okuma erişimine izin verilir.

Çizelge 4- 24 DB belleği için mutlak adresleme

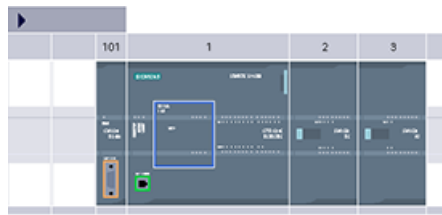
Bit	DB[veri blok numarası].DBX[bayt adresi].[bit adresi]	DB1.DBX2.3
Byte, Word veya Double Word	DB[veri blok numarası].DB [boyut][başlangıç bayt adresi]	DB1.DBB4, DB10.DBW2, DB20.DBD8

### Not

LAD veya FBD içinde bir mutlak adresi belirlediğinizde, STEP 7 mutlak adrestekini göstermek için bir "%" karakteri ile bu adresin önünde yer alır. Programlama yaparken bir mutlak adresi "%" karakterini kullanarak veya kullanmadan girebilirsiniz (örnek, %I0.0 veya I.0). Kullanılmazsa, STEP 7 bu "%" karakterini sağlayacaktır.

SCL'de, bir adresin mutlak adres olduğunu göstermek için adresten önce "%" karakterini girmek zorundasınız. "%" karakteri olmadığında, derleme sırasında STEP 7 tanımlanmamış bir etiket hatası üretir.

### CPU ve I/O modüllerindeki I/O'ların konfigürasyonu



Module	Slot	I address	Q addr...	Type	Order
	103				
	102				
RS485_1	101			OM 1241 (RS485)	6ES7
PLC_1	1			CPU 1214C DOD/C	6ES7
DI14/DO10	1.1	0...1	0...1	DI14/DO10	
AI2	1.2	64...67		AI2	
AO1 x 12bit	1.3		80...81	AO1 signal board	6ES7
HSC_1	1.16	1000....		High speed count	
HSC_2	1.17			High speed count	
HSC_3	1.18			High speed count	
HSC_4	1.19			High speed count	
HSC_5	1.20			High speed count	
HSC_6	1.21			High speed count	
Pulse_1	1.32			Pulse generator (P	
Pulse_2	1.33			Pulse generator (P	
PROFINET L X1				PROFINET interface	
DIB x 24VDC	2	8		SM 1221 DIB x 24	6ES7

Konfigürasyon ekranınıza bir CPU ve I/O modülleri eklediğinizde, I ve Q adresleri otomatik olarak atanır. Konfigürasyon ekranından adres alanını seçip yeni numaralar girerek varsayılan adreslemeyi değiştirebilirsiniz.

- Dijital girişler ve çıkışlar, modül tüm noktaları kullansın ya da kullanmasın 8 noktalık (1 bayt) gruplar halinde atanır.
- Analog girişler ve çıkışlar, 2 noktalık gruplar halinde atanır (4 bayt).

Şekilde, iki SM ve bir SB'li bir CPU 1214C örneği gösterilmiştir. Bu örnekte, DI8 modülünün adresini 8 yerine 2 olarak değiştirebilirsiniz. Araç, hatalı boyuta sahip ya da diğer adreslerle çakışan adres aralıklarını değiştirerek size yardımcı olacaktır.

## 4.3 Analog değerlerin işlenmesi

Analog sinyal modülleri, giriş sinyalleri ya da bir gerilim aralığı veya bir akım aralığını temsil eden beklenen çıkış değerleri sunar. Bu aralıklar  $\pm 10V$ ,  $\pm 5V$ ,  $\pm 2,5V$  veya  $0 - 20mA$  şeklindedir. Modüllerden gelen değerler tamsayı değerleridir.  $0 - 27648$  arası akım için anma aralığını temsil ederken  $-27648 - 27648$  ise gerilim içindir. Aralığın dışındaki her şey, bir taşmayı veya alt taşmayı temsil eder. Ayrıntılar için analog giriş sunumuna ait olan Çizelgelere (Sayfa 916) ve analog çıkış sunumuna ait olan Çizelgelere (Sayfa 917) bakabilirsiniz. Kontrol programınızda, muhtemelen bu değerleri mühendislik birimlerinde, mesela bir hacmi, sıcaklığı, ağırlığı veya başka bir nicel değeri sunmak için kullanmanız gerekebilir. Bunu bir analog giriş için yapmak amacıyla, önce analog değeri  $0,0 - 0,1$  arasında bir gerçek değere (kayan nokta) normalize etmelisiniz. Sonra bunu temsil ettiği mühendislik birimlerinin minimum ve maksimum değerlerine ölçeklemeniz gerekir. Bir analog çıkış değerine dönüştürmeniz gereken mühendislik birimlerindeki değerler için, önce değeri  $0,0$  ile  $1,0$  arasındaki bir değere mühendislik birimleri olarak normalize edersiniz, sonra da bunu analog modülün aralığına bağlı olarak  $0 - 27648$  ya da  $-27648 - 27648$  arasına ölçeklersiniz. STEP 7, bu amaç için NORM\_X ve SCALE\_X komutlarını sunar (Sayfa 251). Analog değerleri ölçeklemek (Sayfa 39) için CALCULATE komutunu da (Sayfa 229) kullanabilirsiniz.

## 4.4 Veri tipleri

Veri tipleri, bir veri elemanının boyutu ile verilerin nasıl yorumlanacağını belirlemek için kullanılır. Her komut parametresi en az bir veri tipini destekler. Bazı parametreler ise birden fazla veri tipini desteklemektedir. Bir parametre için hangi veri tiplerinin desteklendiğini görmek için imleci bir komutun parametre alanının üzerinde tutunuz.

Bir formel parametre, o komut tarafından kullanılacak verilerin konumunu işaretleyen bir komutun üzerindeki tanımlayıcıdır (örnek: bir ADD komutunun IN1 girişi). Bir gerçek parametre, komut tarafından kullanılacak verileri içeren bellek konumu ("% karakter on takısı ile) ya da sabitidir (örnek: %MD400 "Widgetlerin\_Sayısı"). Gerçek parametrenin sizin tarafınızdan belirlenen veri tipi, formel parametrenin komut tarafından belirlenen veri tiplerinin biriyle uyumlu olmalıdır.

Gerçek bir parametre belirlerken, bir etiket (sembol) ya da bir mutlak (direkt) bellek adresi belirlemeniz gerekir. Etiketler, bir sembolik isim (etiket ismi) bir veri tipi, bir bellek alanı, bellek ofseti ve yorumu ile ilişkilendirir ve bunlar PLC etiketleri editöründe ya da bir blok (OB, FC, FB ve DB) için ara yüz editöründe oluşturulabilirler. Eğer ilişkilendirilmiş bir etikete sahip olmayan bir mutlak adres girerseniz, desteklenen veri tipiyle uyumlu olan uygun bir boyut kullanmalısınız, varsayılan bir etiket giriş üzerine oluşturulacaktır. String, Struct, Array ve DTL haricindeki tüm veri tipleri, PLC etiketleri editöründe ve blok ara yüzü editörlerinde mevcuttur. String, Struct, Array ve DTL sadece blok ara yüzü editörlerinde mevcuttur. Giriş parametrelerinin çoğu için sabit bir değer de girebilirsiniz.

- Bit ve Bit dizileri (Sayfa 111): Bool (Bole veya bit değeri), Bayt (8 bit bayt değeri), Word (16 bit değeri), DWord (32 bit double word değeri)
- Tamsayı (Sayfa 112)
  - USInt (işaretsiz 8-bit tamsayı), SInt (işaretili 8-bit tamsayı),
  - UInt (işaretsiz 16-bit integer), Int (işaretili 16-bit tamsayı)
  - UDInt (işaretsiz 32-bit integer), DInt (işaretili 32-bit tamsayı)

- Kayan nokta Gerçek (Sayfa 112): Real (32-bit Real veya kayan nokta değeri), LReal (64-bit Real veya kayan nokta değeri)
- Saat ve Tarih (Sayfa 113): Saat (32-bit IEC zaman değeri), Tarih (16-bit tarih değeri), TOD (32-bit günün saati değeri), DTL (12-byte tarih ve saat yapısı)
- Karakter ve String (Sayfa 114): Char (8-bit tek karakter), String (değişken –uzunluklu string, 254 karaktere kadar)
- Dizi (Sayfa 116)
- Veri yapısı (Sayfa 117): Struct
- PLC Veri tipi (Sayfa 117)
- İşaretçiler (Pointers) (Sayfa 118): Pointer, Any, Variant

Veri tipleri olarak mevcut olmasa da Çizelgede verilen BCD nümerik formatı, dönüştürme komutları tarafından desteklenir.

Çizelge 4- 25 BCD formatın boyut ve aralığı

Format	boyut (bits)	Nümerik aralık	Sabit giriş örnekleri
BCD16	16	-999 ila 999	123, -123
BCD32	32	-9999999 ila 9999999	1234567, -1234567

#### 4.4.1 Bool, Byte, Word ve DWord veri tipleri

Çizelge 4- 26 Bit ve bit dizisi veri tipleri

Veri tipi	Bit boyutu	Sayı tipi	Sayı aralığı	Sabit örnekleri	Adres örnekleri
Bool	1	Boolean	FALSE veya TRUE	TRUE, 1,	I1.0 Q0.1 M50.7 DB1.DBX2.3 Tag_name
		Binary	0 or 1	0, 2#0	
		Octal	8#0 veya 8#1	8#1	
		Hexadecimal	16#0 veya 16#1	16#1	
Byte	8	Binary	2#0 - 2#11111111	2#00001111	IB2 MB10 DB1.DBB4 Tag_name
		İşaretsiz integer	0 - 255	15	
		Octal	8#0 - 8#377	8#17	
		Hexadecimal	B#16#0 - B#16#FF	B#16#F, 16#F	
Word	16	Binary	2#0 - 2#1111111111111111	2#1111000011110000	MW10 DB1.DBW2 Tag_name
		İşaretsiz integer	0 - 65535	61680	
		Octal	8#0 - 8#177777	8#170360	
		Hexadecimal	W#16#0 - W#16#FFFF, 16#0 - 16#FFFF	W#16#F0F0, 16#F0F0	
DWord	32	Binary	2#0 - 2#11111111111111111111111111111111	2#1111000011111111100 001111	MD10 DB1.DBD8 Tag_name
		İşaretsiz integer	0 - 4294967295	15793935	

Veri tipi	Bit boyutu	Sayı tipi	Sayı aralığı	Sabit örnekleri	Adres örnekleri
		Octal	8#0 ila 8#3777777777	8#74177417	
		Hexadecimal	DW#16#0000_0000 ila DW#16#FFFF_FFFF, 16#0000_0000 ila 16#FFFF_FFFF	DW#16#F0FF0F, 16#F0FF0F	

#### 4.4.2 Tamsayı (integer) sayı tipleri

Çizelge 4- 27 Tamsayı veri tipleri (U = unsigned )işaretsiz, S = short (kısa), D= double)

Veri tipi	Bit boyutu	Sayı aralığı	Sabit örnekleri	Adres örnekleri
USInt	8	0 ila 255	78, 2#01001110	MB0, DB1.DBB4, Tag_name
SInt	8	-128 - 127	+50, 16#50	
UInt	16	0 - 65,535	65295, 0	MW2, DB1.DBW2, Tag_name
Int	16	-32,768 - 32,767	30000, +30000	
UDInt	32	0 - 4,294,967,295	4042322160	MD6, DB1.DBD8, Tag_name
DInt	32	-2,147,483,648 - 2,147,483,647	-2131754992	

#### 4.4.3 Kayan noktalı gerçek veri tipleri

Gerçek (ya da kayan nokta) numaralar, ANSI/IEEE 754-1985 standardında tanımlandığı üzere 32 bit tek hassasiyetli numaraları (Real) ya da 64 bit çift hassasiyetli numaralar (LReal) olarak temsil edilir. Tek hassasiyetli kayan nokta numaraları, 6 anlamlı basamağa kadar, çift hassasiyetli kayan nokta numaraları ise 15 anlamlı basamağa kadar kesindir. Bir kayan nokta sabiti girerken hassasiyeti korumak için maksimum 6 (Real) ya da 15 (LReal) anlamlı basamak belirleyebilirsiniz.

Çizelge 4- 28 Kayan noktalı gerçek veri tipleri (L = Uzun)

Veri tipi	Bit boyutu	Sayı aralığı	Sabit örnekleri	Adres örnekleri
Real	32	-3.402823e+38 - ila -1.175 495e-38, ±0, +1.175 495e-38 ila +3.402823e+38	123.456, -3.4, 1.0e-5	MD100, DB1.DBD8, Tag_name
LReal	64	-1.7976931348623158e+308 ila -2.2250738585072014e-308, ±0, +2.2250738585072014e-308 ila +1.7976931348623158e+308	12345.123456789e40, 1.2E+40	DB_name.var_name Rules: <ul style="list-style-type: none"> <li>• direkt adresleme desteği yok</li> <li>• Bir OB, FB veya FC blok ara yüz Çizelgesinde atanabilir.</li> </ul>



Çok büyük ve çok küçük sayıları içeren bir dizi uzun değeri kapsayan hesaplamalar, kesin olmayan sonuçlar verebilir. Numaraların arasındaki fark,  $x > 6$  (real) ya da 15 (LReal) iken, 10 üzeri  $x$  kadar ise bu durum meydana gelebilir. Örneğin (Real):  $100\ 000\ 000 + 1 = 100\ 000\ 000$ .

#### 4.4.4 Saat ve tarih veri tipleri

Çizelge 4- 29 Saat ve tarih veri tipleri

Veri tipi	Boyut	Aralık	Sabit giriş örnekleri
Saat	32 bit	T#-24d_20h_31m_23s_648ms ila T#24d_20h_31m_23s_647ms -2,147,483,648 ms ila +2,147,483,647 ms olarak saklanır.	T#5m_30s T#1d_2h_15m_30s_45ms TIME#10d20h30m20s630ms 500h10000ms 10d20h30m20s630ms
Tarih	16 bit	D#1990-1-1 ila D#2168-12-31	D#2009-12-31 DATE#2009-12-31 2009-12-31
Günün saati	32 bit	TOD#0:0:0.0 ila TOD#23:59:59.999	TOD#10:20:30.400 TIME_OF_DAY#10:20:30.400 23:10:1
DTL (tarih ve saat uzun)	12 bayt	Min.: DTL#1970-01-01-00:00:00.0 Max.: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20:30:20.250

#### Saat

TIME verileri milisaniye olarak yorumlanan bir işaretli çift tamsayı olarak saklanır. Editör formatı; gün (d), saat (h) dakika (m), saniye (s) ve milisaniye (ms) bilgilerini kullanabilir.

Tüm zaman birimlerini belirtmeye gerek yoktur. Örneğin T#5h10s ve 500h geçerlidir.

Tüm belirtilen ünite değerlerinin birleşik değeri, Time veri tipi için milisaniye olarak üst veya alt limitleri aşamaz (-2,147,483,648 ms ila +2,147,483,647 ms).

#### Tarih

DATE verileri, belirlenen tarihi elde etmek için taban tarih 01/01/1990'a eklenen gün sayısı olarak yorumlanan işaretsiz bir tamsayı değeri olarak saklanır. Editör formatı bir yılı, ayı ve günü belirtmelidir

#### TOD

TOD (TIME\_OF\_DAY) verileri belirlenen günün saati için gece yarısından itibaren geçen milisaniye miktarı olarak yorumlanan işaretsiz bir tamsayı olarak saklanır (Gece yarısı = 0 ms). Saat (24 saat/gün), dakika ve saniye belirtilmelidir. Kesirli saniye tanımlaması opsiyoneldir.

## DTL

DTL (Date and Time Long) veri tipi, tarih ve zaman üzerine bilgileri kaydeden 12 baytlık bir veri yapısı kullanır. DTL verilerini bir bloğun Temp belleğinde ya da bir DB içinde tanımlayabilirsiniz. Tüm bileşenler için bir değer, DB editörünün “Başlama değeri” sütununa girilmelidir.

Çizelge 4- 30 DTL için boyut ve aralık

Uzunluk (bytes)	Format	Değer aralığı	Değer giriş örneği
12	Saat ve takvim Yıl-Ay-Gün:Saat:Dakika: Saniye.Nanosaniye	Min.: DTL#1970-01-01-00:00:00.0 Max.: DTL#2554-12-31-23:59:59.999 999 999	DTL#2008-12-16-20:30:20.250

DTL'nin her bir bileşeni, farklı bir veri tipi ve değerler aralığı içerir. Belirtilen bir değer in veri tipi, karşılık gelen bileşenlerin veri tipiyle uyumlu olmalıdır.

Çizelge 4- 31 DTL yapısının elemanları

Bayt	Bileşen	Veri tipi	Değer aralığı
0	Yıl	UINT	1970 - 2554
1			
2	Ay	USINT	1 - 12
3	Gün	USINT	1 - 31
4	Haftanın günü <sup>1</sup>	USINT	1(Pazar) - 7(Cumartesi) <sup>1</sup>
5	Saat	USINT	0 - 23
6	Dakika	USINT	0 - 59
7	Saniye	USINT	0 - 59
8	Nanosaniye	UDINT	0 - 999 999 999
9			
10			
11			

<sup>1</sup> Haftanın günü, değer girişinde dikkate alınmaz.

### 4.4.5 Karakter ve String veri tipleri

Çizelge 4- 32 karakter ve String veri tipleri

Veri tipi	Boyut	Aralık	Sabit giriş örnekleri
Char	8 bit	ASCII karakter kodları: 16#00 - 16#FF	'A', 't', '@'
String	n+ 2 bayt	n = (0 - 254 karakter baytları)	'ABC'

## Char

Char verileri, bellekte bir bayt kaplar ve ASCII formatında kodlanmış tek bir karakter saklar. Editör syntax ASCII karakterinden önce ve sonra tek bir tırnak karakter kullanır. Görünür karakterler ve kontrol karakterleri kullanılabilir. Geçerli kontrol karakterlerinin bir Çizelgesi, String veri tipinin açıklanmasında gösterilmiştir.

## String (Katar)

CPU, tek bayt karakterlerinin bir dizisini depolamak için String veri tipini destekler. String veri tipi bir toplam karakter sayımını (string'deki karakterlerin sayısı) ve geçerli karakter sayımını içerir. String tipi, maksimum karakter sayımı (1 bayt), geçerli karakter sayımı (1 bayt) için 256 bayt sağlar ve her karakter 1 baytta saklanmak üzere 254 karaktere kadardır.

Doğru string'leri (sabitler) tek tırnaklar kullanarak IN tipinin komut parametreleri için kullanabilirsiniz. Örneğin 'ABC' üç karakterli bir string'dir ve S\_CONV komutunun IN parametresi için giriş olarak kullanılabilir. Ayrıca OB, FC, FB ve DB için blok ara yüz editörlerinde "String" veri tipini seçerek String değişkenleri de oluşturabilirsiniz. PLC etiketleri editöründe bir String oluşturamazsınız.

"String" kelimesinden sonra köşeli parantezler içinde maksimum string boyutunu bayt olarak belirleyebilirsiniz ("String" veri tipi açılır menü listesinden seçildikten sonra). Örneğin "MyString[10]" , MyString için maksimum 10 bayt boyut belirleyecektir. Eğer içinde maksimum boyutla birlikte köşeli parantezler kullanmazsanız, bu değer 254 olduğu varsayılır.

Aşağıdaki örnekte, maksimum karakter sayımı 10 ve geçerli karakter sayımı 3 olan bir String tanımlanmıştır. Bu, String'in geçerli olarak 3 adet bir baytlık karakter içerdiği ancak 10 adet bir baytlık karakterler içerecek şekilde genişletilebileceği anlamına gelir.

Çizelge 4- 33 Bir String veri tipi örneği

Toplam Karakter Sayımı	Geçerli Karakter Sayımı	Karakter 1	Karakter 2	Karakter 3	...	Karakter 10
10	3	'C' (16#43)	'A' (16#41)	'T' (16#54)	...	-
Bayt 0	Bayt 1	Bayt 2	Bayt 3	Bayt 4	...	Bayt 11

ASCII kontrol karakterleri, Char ve String verilerinde kullanılabilir. Aşağıdaki Çizelgede, kontrol karakteri syntax örnekleri gösterilmektedir.

Çizelge 4- 34 Geçerli ASCII kontrol karakterleri

kontrol karakterleri	ASCII Hex değeri	kontrol fonksiyonu	Örnekler
\$L veya \$l	0A	Satır atlama	'\$LText', '\$0AText'
\$N veya \$n	0A ve 0D	Satır sonu Yeni satır, string'deki iki karakteri gösterir.	'\$NText', '\$0A\$0DText'
\$P veya \$p	0C	Form besleme	'\$PText', '\$0CText'
\$R veya \$r	0D	Satırbaşı (CR)	'\$RText', '\$0DText'
\$T veya \$t	09	Sekme	'\$TText', '\$09Text'
\$\$	24	Dolar işareti	'100\$\$', '100\$24'
\$'	27	Tek tırnak	'\$'Text\$', '\$27Text\$27'

#### 4.4.6 Dizi Veri tipleri

##### Diziler

Aynı veri türünden çoklu elemanı içeren bir dizi oluşturabilirsiniz. Diziler OB, FC, FB ve DB için blok ara yüzü editörlerinde oluşturulabilir. PLC etiketleri editöründe bir dizi oluşturamazsınız.

Blok ara yüz editöründen bir dizi oluşturmak için diziyi isimlendirin ve veri tipini "Array [lo .. hi] of type" olarak seçtikten sonra "lo", "hi" ve "type" düzenlemelerini şöyle yapınız:

- lo – dizinin için başlangıç (en düşük) indeksi
- hi – dizinin için bitiş (en yüksek) indeksi
- tip - BOOL, SINT, UDINT gibi veri tiplerinden birisi

Çizelge 4- 35 ARRAY (dizi) veri tipi kuralları

Veri tipi	Dizi sözdizimi		
ARRAY	İsim [index1_min..index1_max, index2_min..index2_max] of <veri tipi>		
	<ul style="list-style-type: none"> <li>• Tüm dizi elemanları aynı veri tipinden olmalıdır.</li> <li>• İndeks negatif olabilir, ancak alt limit üst limitten az ya da buna eşit olmalıdır.</li> <li>• Diziler birden altıya kadar boyut sahibi olabilir.</li> <li>• Çok- boyutlu indeks min..max bildirimleri, virgül karakteriyle ayrılır.</li> <li>• Yuvalanmış diziler veya dizilerin dizilerine izin verilmez.</li> <li>• Bir dizinin bellek boyutu = (bir elemanın boyutu * dizideki elemanların toplam sayısı)</li> </ul>		
	Dizi indeksi	Geçerli indeks veri tipleri	Dizi indeks kuralları
Sabit veya değişken	USInt, SInt, UInt, Int, UInt, DInt	<ul style="list-style-type: none"> <li>• Değer limitleri:: -32768 ila +32767</li> <li>• Geçerli: Karışık sabitler ve değişkenler</li> <li>• Geçerli: Sabit ifadeler</li> <li>• Geçerli değil: Değişken ifadeler</li> </ul>	

<b>Örnek: Dizi bildirimleri</b>	ARRAY[1..20] of REAL ARRAY[-5..5] of INT ARRAY[1..2, 3..4] of CHAR	Bir boyut, 20 eleman Bir boyut, 11 eleman İki boyut, 4 eleman
<b>Örnek: Dizi adresleri</b>	ARRAY1[0] ARRAY2[1,2] ARRAY3[i,j]	ARRAY1 eleman 0 ARRAY2 eleman [1,2] i =3 ve j=4 ise, sonra ARRAY3 eleman [3, 4] adreslenir.

#### 4.4.7 Veri yapısı veri tipi

Diğer veri tiplerinden oluşan verilerin bir yapısını tanımlamak için "Struct" veri tipini kullanabilirsiniz. Struct veri tipi, bir ilgili proses veri grubunu tek bir veri ünitesi olarak işlemek için kullanılabilir. Bir struct veri tipi isimlendirilir ve iç veri yapısı veri bloğu editöründe ya da bir blok ara yüz editöründe tanımlanır.

Diziler ve yapılar aynı zamanda daha büyük bir yapıda gruplanabilir. Bir yapı, sekiz seviye derinliğe kadar iç içe oluşturulabilir. Örneğin diziler içeren yapılardan bir yapı oluşturabilirsiniz.

#### 4.4.8 PLC veri tipi

PLC veri tipi editörü, programınızda birden çok defa kullanabileceğiniz veri yapıları tanımlamanıza izin verir. Bir PLC veri tipini, proje ağacından "PLC veri tipleri" ("PLC data types") kolunu açarak ve "Yeni veri tipi ekle" ("Add new data type") ögesine çift tıklayarak oluşturabilirsiniz. Yeni oluşturulan PLC veri tipi ögesinde, varsayılan ismi değiştirmek için ögenin üzerine aralıklı olarak iki kez tıklayınız, PLC veri tipi editörünü açmak için çift tıklayınız.

Veri bloğu editöründe kullanılanlarla aynı düzenleme yöntemlerini kullanarak özel bir PLC veri tipi yapısı oluşturabilirsiniz. İsteddiğiniz veri yapısını oluşturmak için gerekli olan veri tipleri için yeni satırlar ekleyiniz.

Yeni bir PLC veri tipi oluşturulduğunda, bu yeni PLC tipinin ismi, DB editöründeki ve kod bloğu ara yüz editöründeki veri tipi seçimi açılır menü listesinde görünecektir.

PLC veri tiplerinin potansiyel kullanımları:

- PLC veri tipleri, bir kod bloğu ara yüzünde veya veri bloklarında direkt olarak bir veri tipi olarak kullanılabilir
- PLC veri tipleri, aynı veri yapısını kullanan çeşitli global veri bloklarının oluşturulması için bir şablon olarak kullanılabilir.

Örneğin bir PLC veri tipi, renklerin karışımı için bir reçete olabilir. Bu PLC veri tipini çoklu veri bloklarına atayabilirsiniz. Her bir veri bloğu, belirli bir rengi oluşturmak için ayarlanan değişkenlere sahip olabilir.

#### 4.4.9 Pointer (işaretçi) veri tipleri

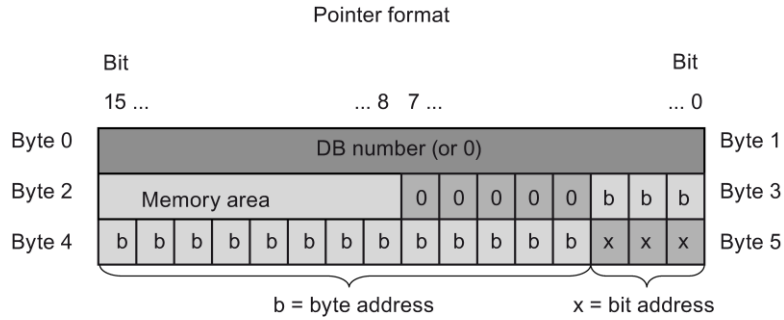
Pointer veri tipleri (Pointer, Any ve Variant), FB ve FC kod blokları için olan blok ara yüzü çizelgelerinde kullanılabilir. Blok ara yüzü veri tipi açılır menü listelerinden bir pointer veri tipini seçebilirsiniz.

Variant veri tipi de komut parametreleri için kullanılabilir.

##### 4.4.9.1 "Pointer" pointer veri tipi

Pointer veri tipi, özel bir değişkene işaret eder. Bellekte 6 bayt (48 bit) işgal eder ve şu bilgileri içerebilir:

- DB numarası veya eğer veriler bir DB'de saklanmıyorsa, 0
- CPU'daki saklama alanı
- Değişken adresi



Komuta bağlı olarak şu üç tip pointer tipini tanımlayabilirsiniz:

- Alan iç pointer'i: bir değişkenin adresindeki verileri içerir
- Alan kesişim pointer'i: bellek alanındaki verileri ve bir değişkenin adresini içerir
- DB pointer'i: bir veri blok numarasını ve bir değişkenin adresini içerir

Çizelge 4- 36 Pointer tipleri:

Tip	Format	Örnek giriş
Alan iç pointer	P#Byte.Bit	P#20.0
Alan kesişim pointer	P#Memory_area.Byte.Bit	P#M20.0
DB-pointer	P#Data_block.Data_element	P#DB10.DBX20.0

Pointer tipinin bir parametresini, ön ek (P #) olmadan girebilirsiniz. Yaptığınız giriş otomatik olarak pointer formatına dönüştürülecektir.

Çizelge 4- 37 Pointer verilerinde bellek alanı kodlaması:

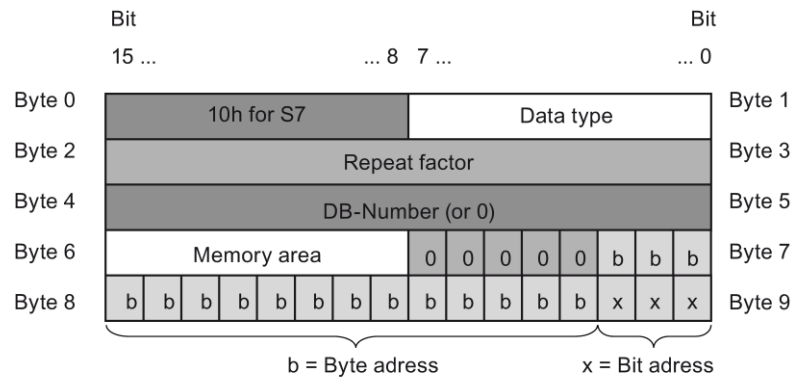
Hexadecimal kod	Veri tipi	Açıklama
b#16#81	I	Giriş bellek alanı
b#16#82	Q	Çıkış bellek alanı
b#16#83	M	Marker bellek alanı
b#16#84	DBX	Veri bloğu
b#16#85	DIX	Instance veri bloğu
b#16#86	L	Lokal veri
b#16#87	V	Önceki lokal veriler

#### 4.4.9.2 "Any" pointer veri tipi

ANY (Herhangi) pointer veri tipi, bir veri alanının başlangıcına işaret eder ve onun uzunluğunu belirtir. ANY pointer'i bellekte 10 bayt kullanır ve şu bilgileri içerebilir:

- Veri Tipi: Veri elemanlarının veri tipi
- Tekrar faktörü: Veri elemanlarının sayısı
- DB Numarası: Veri elemanlarının saklandığı data bloğu
- Depolama alanı: Veri elemanlarının saklandığı CPU bellek alanı
- Başlangıç adresi: Verilerin "Bayt.Bit" başlama adresi

Aşağıdaki görüntü, ANY pointer'inin yapısı gösterilmiştir.



Bir pointer, ANY yapılarını tespit edemez, sadece lokal değişkenlere atanabilir.

Çizelge 4- 38 ANY pointer formatı ve örnekler:

Format	Giriş örneği	Açıklama
P#Data_block.Memory_area Data_address Tip Numarası	P#DB 11.DBX 20.0 INT 10	DBB 20.0'dan başlayarak global DB 11'de 10 word
P#Memory_area Data_address Tip Numarası	P#M 20.0 BYTE 10 P#I 1.0 BOOL 1	MB 20.0'dan başlayarak 10 bayt Giriş 1.0

Çizelge 4- 39 ANY pointer'indeki veri tipi kodlaması

Hexadecimal kod	Veri tipi	Açıklama
b#16#00	Null	Boş pointer
b#16#01	Bool	Bitler
b#16#02	Byte	Baytlar, 8 Bit
b#16#03	Char	8-bit karakter
b#16#04	Word	16-bit-word
b#16#05	Int	16-bit-tamsayı
b#16#37	SInt	8-bit- tamsayı
b#16#35	UInt	16-bit işaretli tamsayı
b#16#34	USInt	8-bit işaretli tamsayı
b#16#06	DWord	32-bit double word
b#16#07	DInt	32-bit double integer
b#16#36	UDInt	32-bit- işaretli double tamsayı
b#16#08	Real	32-Bit kayan nokta
b#16#0B	Time	Zaman
b#16#13	String	Karakter string (katarı)

Çizelge 4- 40 ANY pointer'indeki bellek alanı kodlaması:

Hexadecimal kod	Veri tipi	Açıklama
b#16#81	I	Giriş bellek alanı
b#16#82	Q	Çıkış bellek alanı
b#16#83	M	Marker bellek alanı
b#16#84	DBX	Veri bloğu
b#16#85	DIX	Instance veri bloğu
b#16#86	L	Local veri
b#16#87	V	Önceki local veri

#### 4.4.9.3 "Variant" pointer veri tipi

Varyant veri tipi, farklı veri tiplerinin veya parametrelerin değişkenlerine işaret edebilir. Varyant pointer'i, yapılar ve bağımsız yapısal bileşenlere işaret edebilir. Varyant pointer'i bellekte herhangi bir alan işgal etmez.

Çizelge 4- 41 Varyant pointer'in özellikleri

uzunluk (Bayt)	Gösterilişi	Format	Örnek giriş
0	Sembolik	Operand	MyTag
		DB_name.Struct_name.element_name	MyDB.Struct1.pressure1
	Mutlak	Operand	%MW10
		DB_number.Operand Type Length	P#DB10.DBX10.0 INT 12



#### 4.4.10 Etiketli bir veri tipinin bir dilimine ("slice") erişim

PLC etiketleri ve veri blok etiketleri boyutlarına göre bir bit, bayt veya word seviyesinde erişilebilir.

Böyle bir veri dilimine erişebilmek için sözdizimi aşağıdaki gibidir:

- "<PLC tag name>".xn (bit access)
- "<PLC tag name>".bn (byte access)
- "<PLC tag name>".wn (word access)
- "<Data block name>".<tag name>.xn (bit access)
- "<Data block name>".<tag name>.bn (byte access)
- "<Data block name>".<tag name>.wn (word access)

Bir double word-boyutlu etikete 0-31 bitleri, 0-3 baytları veya 0-1 word'u ile erişilebilir. Bir word-boyutlu etikete 0-15 bitleri, 0-1 baytları veya 0 word'u ile erişilebilir. Bir bayt-boyutlu etikete 0-7 bitleri, 0 baytı ile erişilebilir. Bit, bayt ve word dilimleri bitlerin, baytların veya word'ların beklenen operantlar olduğu her yerde kullanılabilir.

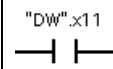
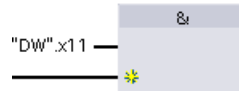
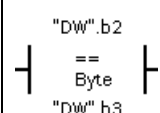
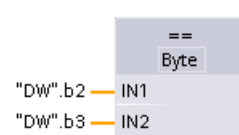

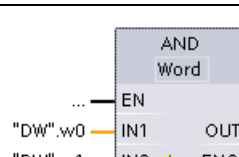
																BYTE															
																WORD															
DWORD																															
x31	x30	x29	x28	x27	x26	x25	x24	x23	x22	x21	x20	x19	x18	x17	x16	x15	x14	x13	x12	x11	x10	x9	x8	x7	x6	x5	x4	x3	x2	x1	x0
b3								b2								b1								b0							
w1																w0															

#### Not

Dilimlemeyle erişilebilecek geçerli veri tipleri; Byte, Char, Conn\_Any, Date, DInt, DWord, Event\_Any, Event\_Att, Hw\_Any, Hw\_Device, HW\_Interface, Hw\_Io, Hw\_Pwm, Hw\_SubModule, Int, OB\_Any, OB\_Att, OB\_Cyclic, OB\_Delay, OB\_WHINT, OB\_PCYLE, OB\_STARTUP, OB\_TIMEERROR, OB\_Tod, Port, Rtm, SInt, Time, Time\_Of\_Day, UDIInt, UInt, USInt ve Word. Real tipli PLC etiketlerine dilimleme ile erişilebilir, ancak Real tipli veri bloğuna erişilemez.

## Örnekler

PLC etiket Çizelgesinde, "DW" beyan edilen tip DWORD etiketidir. Örnekler show bit, bayt, word dilim erişimlerini göstermektedir:

	LAD	FBD	SCL
Bit erişim			<pre>IF "DW".x11 THEN ... END_IF;</pre>
Bayt erişim			<pre>IF "DW".b2 = "DW".b3 THEN ... END_IF;</pre>
Word erişim			<pre>out:= "DW".w0 AND "DW".w1;</pre>

### 4.4.11 Bir AT bindirmeli (overlay) bir etikete erişim

AT etiket bindirmesi, farklı bir tipin bindirilen bildirimine sahip standart erişim bloğunun önceden bildirilmiş bir etiketine erişmenize izin verir. Örneğin; bir baytın, Word'un veya Bool dizini olan DWord veri tipinin bir etiketinin münferit bitlerini adresleyebilirsiniz.

## Bildirim

Bir parametreye bindirim yapmak için, bindirme yapılacak parametreden sonra ilave bir parametreyi direkt olarak bildiriniz ve "AT" veri tipini seçiniz. Editör bindirme oluşturur ve sonra bindirme için kullanmayı istediğiniz veri tipi, yapısı veya dizini seçiniz.

## Örnek

Bu örnek, standart-erişim FB'sinin giriş parametrelerini gösterir. Bayt etiketli B1 bir Booleans diziniyle bindirilir:

■	B1	Byte
▼	AT	AT "B1" Array [0..7] of Bool
■	AT[0]	Bool
■	AT[1]	Bool
■	AT[2]	Bool
■	AT[3]	Bool
■	AT[4]	Bool
■	AT[5]	Bool
■	AT[6]	Bool
■	AT[7]	Bool



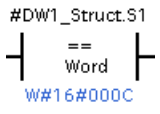
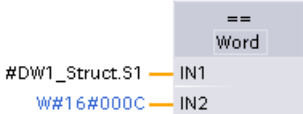
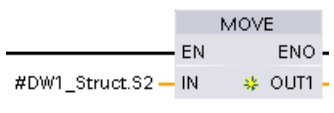

Çizelge 4- 42 Bir Boolean diziniyle bir baytın bindirimi

7	6	5	4	3	2	1	0
AT[0]	AT[1]	AT[2]	AT[3]	AT[4]	AT[5]	AT[6]	AT[7]

Diğer bir örnek bir Struct ile bindirilmiş bir DWord etiketidir:

DW1	DWord
DW1_Struct	Struct
S1	Word
S2	Byte
S3	Byte

Bindirim tipleri program mantığında doğrudan adreslenebilir:

LAD	FBD	SCL
		<pre>IF #AT[1] THEN ... END_IF;</pre>
		<pre>IF (#DW1_Struct.S1 = W#16#000C) THEN ... END_IF;</pre>
		<pre>out1 := #DW1_Struct.S2;</pre>

## Kurallar

- Etiketlerin bindirilmesi sadece standart erişimli FB ve FC bloklarında mümkündür.
- Bütün blok tipleri ve bütün bildirim bölümleri için parametreleri yükleyebilirsiniz (overlay).
- Bindirimli bir parametreyi diğer blok parametreleri gibi kullanabilirsiniz.
- VARIANT tipli parametreleri yükleyemezsiniz.
- Bindirme parametresinin boyutu bindirilen parametrenin boyutundan küçük veya eşit olmalıdır.
- Bindirme değişkeni bindirme yaptığı değişkenden hemen sonra bildirilmeli ve "AT" anahtar sözcük ile tanımlanmalıdır.

## 4.5 Bellek kartının kullanımı

### Not

CPU sadece önceden formatlanmış SIMATIC bellek kartını destekler (Sayfa 978).

Formatlanan bellek kartına herhangi bir program kopyalamadan önce, daha önce kaydedilmiş programları bellek kartından siliniz.

Bellek kartını ya bir transfer kartı ya da bir program kartı olarak kullanınız. Bellek kartına kopyaladığınız her hangi bir program bütün kod bloklarını ve veri bloklarını, teknoloji nesnelere ve cihaz konfigürasyonunu içerir. Kopyalanan bir program zorlama değerlerini içermez.

- STEP 7'yi kullanmaksızın CPU'nun dahili yükleme belleğine bir program kopyalamak için bir transfer kartı (Sayfa 127) kullanınız. Transfer kartını taktıktan sonra, CPU önce kullanıcı programını ve herhangi zorlama değerlerini dahili yükleme belleğinden siler ve programı transfer kartından dahili yükleme belleğine kopyalar. Transfer işlemi tamamlandığında, transfer kartını çıkarmalısınız.

Şifre kaybedildiği veya unutulduğunda (Sayfa 136), şifre-korumalı CPU'ya erişmek için boş bir transfer kartını kullanabilirsiniz. Boş transfer kartının takılması, CPU dahili belleğindeki şifre-korumalı programı siler. Sonra yeni programı CPU'ya yükleyebilirsiniz.

- Bir program kartını (Sayfa 130) CPU için harici bir yükleme kartı olarak kullanınız. Program kartının CPU'ya takılması CPU dahili yükleme belleğinin hepsini siler (kullanıcı programı ve zorlama değerler). CPU sonra dahili yükleme belleğindeki (program kartı) programı yürütür. Bir program kartına sahip olan CPU'ya yükleme sadece harici yükleme belleğini (program kartı) günceller.

CPU'nun dahili yükleme belleği silinmiş olduğu için program kartını taktığınızda program kartı CPU içinde kalmalıdır. Program kartını çıkarırsanız, CPU STOP moduna geçer. Hata LED'i program kartının çıkarılmış olduğunu göstermek için yanar söner.

Bellek kartına kopyalanan program; kod blokları, veri blokları, teknoloji nesnelere ile cihaz konfigürasyonunu içerir. Bellek kartı hiçbir zorlama değeri bulundurmaz. Zorlama değerleri programın bir parçası değildir, ancak CPU'nun dahili yükleme belleği veya harici yükleme belleği (bir program kartı) olup olmadığına bakılmaksızın yükleme belleğinde saklanabilir. Bir program kartı CPU'ya takılırsa, STEP 7 sonra sadece zorlama değerlerini program kartındaki harici yükleme kartına uygular.

Donanım yazılımı güncellemelerini yüklerken bir bellek kartı da kullanırsınız (Sayfa 133).

#### 4.5.1 CPU'ya bir bellek kartı takmak

**UYARI**

**Bellek kartını koruyunuz ve elektrostatik boşalmadan muhafaza ediniz.**

Elektrostatik boşalma, bellek kartına veya CPU üzerindeki yuvaya zarar verebilir.

Bellek kartıyla çalışırken topraklayıcı bir altlık (pad) ile temas sağlayınız ve/veya topraklı bir bileklik takınız. Bellek kartını iletken bir muhafaza içinde saklayınız.



Bellek kartının yazmaya karşı korumalı olmadığından emin olunuz. Koruma anahtarını "Lock" konumundan alınız.

**! UYARI**

**Bir bellek kartını takmadan önce CPU'nun bir prosesi yürütmediğini doğrulayınız.**

Çalışan bir CPU'ya bir bellek kartı takarsanız (bir program kartı, transfer kartı veya cihaz yazılım güncelleme kartı olması fark etmez), CPU derhal STOP moduna geçer. Bu da ekipmana ya da proses kontrolüne zarar verebilir.

Bir bellek kartı takmadan veya çıkarmadan önce her zaman CPU'nun bir makineyi veya bir prosesi aktif olarak kontrol etmekte olmadığından emin olunuz. Uygulamanız veya prosesiniz için her zaman bir acil durum durdurma devresi tesis ediniz.

**Not**

**S7-1200 V4.0 CPU'lara V3.0 program transfer kartını takmayınız.**

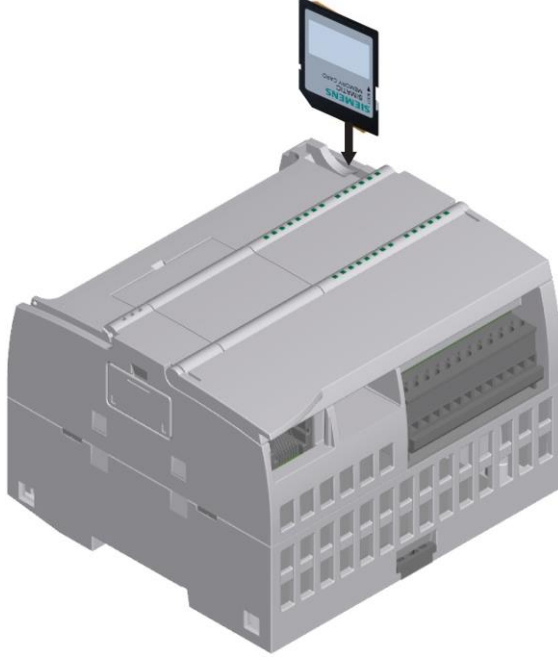
Versiyon 3.0 program transfer kartları S7-1200 V4.0 CPU'ların versiyonları ile uyumlu değildir. V3.0 program içeren bir bellek kartının takılması, bir CPU hatasına neden olur.

Geçersiz bir program transfer kartını (Sayfa 127) takarsanız, kartı çıkarmalı, STOP'tan RUN'a geçiş ve bir bellek reseti (MRES) veya güç döngüsü yapmalısınız. CPU'yu hata durumundan kurtardıktan sonra geçerli bir V4.0 CPU programı yükleyebilirsiniz.

V3.0 programını V4.0 programına transfer etmek için Donanım Konfigürasyonunda (Hardware Configuration) Cihaz Değiştir (Change Device) için TIA Portal'ı kullanmalısınız.

**Not**

STOP modunda iken CPU'ya bir bellek kartı takarsanız, diyagnostik ara belleğinde bellek kartı değerlendirmesinin başladığını belirten bir mesaj görüntülenir. CPU bellek kartını, CPU'nun RUN moduna bir sonraki geçişinde (CPU belleğini bir MRES ile resetleyiniz) ya da CPU'nun enerjisini kapatıp açtığınızda değerlendirecektir.

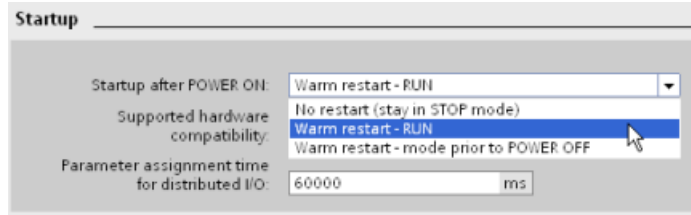


Bir bellek kartını takmak için CPU klemens kapağını açınız ve bellek kartını yatağına takınız. Buradaki itmeli bir konektör takma ve çıkarma işlemlerini kolaylaştırır.

Bellek kartı düzgün bir kurulum için uygun hale getirilmiştir.

## 4.5.2 Projeyi hafıza kartına kopyalamadan önce CPU'nun başlangıç parametresini konfigüre etmek

Bir programı bir transfer kartına veya bir program kartına kopyaladığınızda, program CPU için başlangıç parametresini içerir. Programı bellek kartına kopyalamadan önce her zaman enerji kapatılıp açılmasını izleyen durumda CPU için işletim modunu konfigüre ettiğinizden emin olunuz. CPU'nun STOP modu, RUN modu veya bir önceki moddan (enerji döngüsünden önceki) hangisinde başlatacağınızı seçiniz.



## 4.5.3 Transfer kartı

### UYARI

**Bellek kartını koruyunuz ve elektrostatik boşalmadan muhafaza ediniz.**

Elektrostatik boşalma, bellek kartına veya CPU üzerindeki yuvaya zarar verebilir.

Bellek kartıyla çalışırken topraklayıcı bir altlık (pad) ile temas sağlayınız ve/veya topraklı bir bileklik takınız. Bellek kartını iletken bir muhafaza içinde saklayınız.

**Bir transfer kartı oluşturmak**

Transfer kartına bir program kopyalamadan önce CPU'nun başlangıç parametresini konfigüre etmeyi daima hatırlayınız (Sayfa 127). Bir transfer kartı oluşturmak için şu adımları takip ediniz:

1. Bilgisayarınıza bağlı bir SD kart okuyucusuna/yazıcısına yazma koruması olmayan bir SIMATIC bellek kartı takınız. (Kart yazma korumalı ise koruma anahtarını "Lock" pozisyonundan çekiniz.)

Bir kullanıcı programı veya başka bir yazılım güncellemesi içeren bir SIMATIC bellek kartı kullanıyorsanız, kartı yeniden kullanmadan önce program dosyalarını silmelisiniz. Windows Gezgini kullanarak bellek kartının içeriğini görüntüleyebilir ve "S7\_OB.S7S" dosyası ve mevcut "Data Logs" klasörleri ve klasör dizinlerini de ("SIMATIC.S7S" veya "FWUPDATE.S7S" gibi) silebilirsiniz

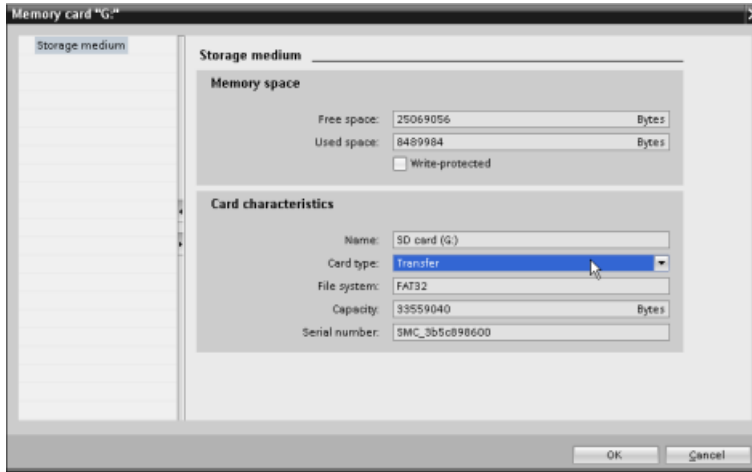
**UYARI**

**Bellek kartından "\_\_LOG\_\_" ve "crdinfo.bin" saklı dosyalarını silmeyiniz.**

"\_\_LOG\_\_" ve "crdinfo.bin" dosyaları bellek kartı için gereklidir. Bu dosyaları silerseniz bellek kartını CPU ile kullanamazsınız.

2. Proje ağacında (Proje görünümü), "SIMATIC Kart Okuyucu" klasörünü genişleterek kart okuyucunuzu seçiniz.
3. Kart okuyucunun içindeki bellek kartına karşılık gelen sürücü harfinin üzerine sağ tıklayarak ve içerik menüsünden "Özellikleri seçerek "Bellek kartı" diyalogunu açınız.
4. "Bellek kartı" diyalogundaki "Kart tipi" açılır menü listesinden "Transfer"i seçiniz.

Bu noktada STEP 7 boş transfer kartını oluşturur. Kayıp bir CPU şifresini kurtarmağa benzer olarak boş bir transfer kartı oluşturuyorsanız (Sayfa 136), transfer kartını kart okuyucudan çıkarınız.




5. Proje ağacından CPU cihazını (PLC\_1 [CPU 1214 DC/DC/DC] gibi) seçip CPU cihazını bellek kartına sürükleyerek programı ekleyiniz. (Başka bir yöntem de CPU cihazını kopyalamak ve hafıza kartına yapıştırmaktır). CPU cihazını bellek kartına kopyalamak, "Yükleme ön izleme" diyalogunu açar.



6. CPU cihazını bellek kartına kopyalamak için “Yükleme ön izleme” diyalogundan, “Yükle” düğmesine tıklayınız.
7. Diyalog, CPU cihazının (program) hatasız bir şekilde yüklendiğini belirten bir mesaj görüntülediğinde, “Bitir” düğmesine tıklayınız.

### Bir transfer kartının kullanımı

 <b>UYARI</b>
<p><b>Bir bellek kartını takmadan önce CPU'nun bir prosesi yürütmediğini doğrulayınız.</b></p> <p>Bir hafıza kartının takılması CPU'nun STOP moduna geçmesine yol açar. Bu da bir çevrimiçi prosesin ya da makinenin çalışmasını etkileyebilir. Bir proseste veya makinede yaşanabilecek beklenmeyen işleyişler, kişisel yaralanmalara veya ölümlere ve/veya mal hasarlarına yol açabilir.</p> <p>Bir bellek kartını takmadan önce CPU'nun STOP modunda ve prosesinizin güvenli bir durumda olduğundan daima emin olunuz.</p>

### Not

#### **S7-1200 V4.0 CPU'lara V3.0 program transfer kartlarını takmayınız.**

Versiyon 3.0 program transfer kartları S7-1200 V4.0 CPU'ların versiyonları ile uyumlu değildir. V3.0 program içeren bir bellek kartının takılması, bir CPU hatasına neden olur.

Geçersiz bir program transfer kartını takarsanız, kartı çıkarmalı, STOP'tan RUN'a geçiş ve bir bellek reseti (MRES) veya güç döngüsü yapmalısınız. CPU'yu hata durumundan kurtardıktan sonra geçerli bir V4.0 CPU programı yükleyebilirsiniz.

Bir CPU'ya bir program transfer etmek için şu adımları takip ediniz:

1. Transfer kartını CPU'ya takın (Sayfa 125). CPU, RUN modunda ise STOP moduna geçecektir. Bellek kartının değerlendirmeye ihtiyaç duyduğunu göstermek için bakım (MAINT) LED'i yanıp söner.
2. Bellek kartını değerlendirmek için CPU' nun enerjisini kesin ve tekrar uygulayınız. CPU'nun yeniden yüklenmesi için alternatif yöntemler STOP'tan RUN moduna geçiş ya da STEP 7'den bir bellek resetlemesi (MRES) yapmaktır.
3. Yeniden yüklemeyen ve bellek kartının değerlendirilmesinden sonra CPU, programı CPU'nun dahili yükleme belleğine kopyalar. RUN/STOP LED'i sırayla yeşil ve sarı olarak yanarak programın kopyalanmakta olduğunu gösterir. RUN/STOP LED'i tamamen sarı olduğunda ve MAINT LED'i yanıp sönmeye başladığında kopyalama işlemi tamamlanmıştır. Bundan sonra bellek kartını çıkarabilirsiniz.
4. Dahili yükleme belleğine transfer edilen yeni programı değerlendirmek için CPU'yu yeniden yükleyiniz (gücü yeniden uygulayarak veya diğer yeniden yükleme yöntemlerini kullanarak).

CPU sonra proje için yapılandırdığınız başlangıç moduna (RUN veya STOP) geçer.

### Not

CPU'yu RUN moduna ayarlamadan önce transfer kartını çıkarmanız gerekir.

#### 4.5.4 Program kartı

**Dikkat**

**Elektrostatik boşalma, hafıza kartına ya da CPU üzerindeki yuvaya zarar verebilir.**

Bellek kartıyla çalışırken topraklayıcı bir altlık ile temas kurunuz ve/veya topraklı bir bileklik takınız. Bellek kartını iletken bir muhafaza içinde saklayınız.



Bellek kartının yazmaya karşı korumalı olmadığından emin olunuz. Koruma anahtarını "Lock" konumundan alınız.

Program kartına herhangi bir program elemanı kopyalamadan önce, daha önce kaydedilmiş programları bellek kartından siliniz.

#### Bir program kartı oluşturmak

Bir program kartı olarak kullanıldığında bellek kartı CPU'nun harici yükleme belleğidir. Program kartını çıkarırsanız, CPU'nun dahili yükleme belleği boşalır.

**Not**

CPU'ya boş bir bellek kartı takıp CPU'nu enerjisini kapatıp açarak, STOP – RUN geçişi uygulayarak veya bir bellek sıfırlaması (MRES) yaparak bir hafıza kartı değerlendirmesi gerçekleştirirseniz, CPU'nun dahili yükleme belleğindeki program ve zorlama değerleri bellek kartına kopyalanır. (Hafıza kartı artık bir program kartıdır.). Kopyalama tamamlandıktan sonra CPU'nun dahili yükleme belleğindeki program silinir. CPU bundan sonra konfigüre edilmiş olduğu başlangıç moduna (RUN veya STOP) geçer.

Program kartına bir proje kopyalamadan önce CPU'nun başlangıç parametresini konfigüre etmeyi daima hatırlayınız (Sayfa 127). Bir program kartı oluşturmak için şu adımları takip ediniz:

1. Bilgisayarınıza bağlı bir SD kart okuyucusuna/yazıcısına boş bir SIMATIC bellek kartı takınız. (Kart yazma korumalı ise koruma anahtarını "Lock" pozisyonundan çekiniz.)

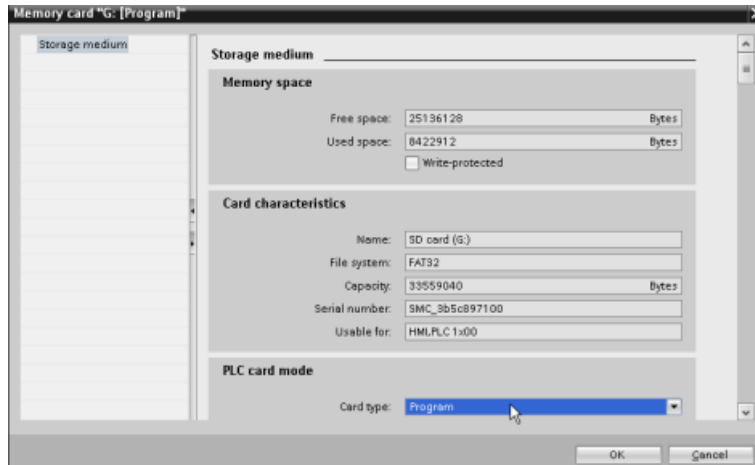
Bir kullanıcı programı veya bir donanım yazılımı güncellemesi içeren bir SIMATIC bellek kartı kullanıyorsanız, kartı yeniden kullanmadan önce program dosyalarını silmelisiniz. Windows Gezginini kullanarak bellek kartının içeriğini görüntüleyiniz ve "S7\_-OB.S7S" dosyasını siliniz ve mevcut "Data Logs" klasörlerini ve her hangi bir klasör dizinini ("SIMATIC.S7S" veya "FWUPDATE.S7S" gibi) de silebilirsiniz.

**UYARI**

**Bellek kartından "\_LOG\_" ve "crdinfo.bin" saklı dosyalarını silmeyiniz.**

"\_LOG\_" ve "crdinfo.bin" dosyaları bellek kartı için gereklidir. Bu dosyaları silerseniz, bellek kartını CPU ile kullanamazsınız.

2. Proje ağacında (Proje görünümü), "SIMATIC Kart Okuyucu" klasörünü genişleterek kart okuyucunuzu seçiniz.
3. Kart okuyucunun içindeki bellek kartının üzerine sağ tıklayarak ve içerik menüsünden "Özellikler"i seçerek "Bellek kartı" diyalogunu açınız.
4. "Bellek kartı" diyalogundaki açılır menü listesini kullanarak "Program"ı seçiniz.



5. Proje ağacından CPU cihazını (PLC\_1 [CPU 1214 DC/DC/DC] gibi) seçip CPU cihazını bellek kartına sürükleyerek programı ekleyiniz. (Başka bir yöntem de CPU cihazını kopyalayarak bellek kartına yapıştırmaktır). CPU cihazını bellek kartına kopyalamak, "Yükleme ön izleme" diyalogunu açar.
6. "Yükleme ön izleme" diyalogunda CPU cihazını bellek kartına kopyalamak için "Yükle" düğmesine tıklayınız.
7. Diyalog CPU cihazının (program) hatasız bir şekilde yüklendiğini belirten bir mesaj görüntülediğinde, "Bitir" düğmesine tıklayınız.

**CPU'nuz için program kartını yükleme belleği olarak kullanmak****⚠ UYARI****Bir program kartı takılması ile ilişkili riskler**

Bellek kartını takmadan önce CPU'nun aktif bir şekilde çalışıyor olmadığından emin olunuz.

Bir bellek kartının takılması CPU'nun STOP moduna geçmesine yol açar. Bu da bir çevrimiçi prosesin ya da makinenin çalışmasını etkileyebilir. Bir proseste ya da makinede yaşanabilecek beklenmeyen işleyişler, kişisel yaralanmalara veya ölümlere ve/veya mal hasarlarına yol açabilir.

Bellek kartını takmadan önce CPU'nun offline ve güvenli bir durumda olduğundan emin olunuz.

CPU'nuzla bir program kartını kullanmak için şu adımları takip ediniz:

1. Program kartını CPU'ya takınız. CPU, RUN modunda ise STOP moduna geçecektir. Bellek kartının değerlendirmeye ihtiyaç duyduğunu göstermek için bakım (MAINT) LED'i yanıp söner.
2. Bellek kartını değerlendirmek için CPU'nun enerjisini kapatıp açınız. CPU'nun yeniden yüklenmesi için kullanılacak alternatif yöntemler bir STOP'tan RUN'a geçişi ya da STEP 7'den bir bellek resetlemesi (MRES) yapmaktır.
3. CPU yeniden yüklendikten ve program kartını değerlendirdikten sonra, CPU'nun dahili yükleme belleğini CPU siler

CPU bundan sonra CPU için yapılandırılmış olduğunuz başlangıç moduna (RUN veya STOP) geçer.

Program kartı CPU'da kalmalıdır. Program kartı çıkarıldığında CPU'nun dahili yükleme belleğinde hiçbir program kalmaz.

**⚠ UYARI****Bir program kartı çıkarılması ile ilişkili riskler**

Program kartını çıkarırsanız, CPU kendisine ait harici yükleme belleğini kaybeder ve bir hata üretir. CPU, STOP moduna geçer ve hata LED'i yanıp söner.

Kontrol cihazları güvensiz durumda kalır ve kontrollü ekipmanın beklenmeyen operasyonu meydana gelir. Böyle beklenmeyen operasyonlar personelin ölümü veya ciddi yaralanmalara ve/veya ekipmanın zarar görmesine yol açar.

CPU'dan bir program kaldırmayı anlamadan program kartını çıkarmayınız.

#### 4.5.5 Donanım yazılımı güncelleme

Donanım yazılımı güncellemesi için bir bellek kartı kullanabilirsiniz. Alternatif yöntemler; bir donanım yazılımı güncellemesi yapmak için Web sunucunun modül bilgi sayfasını (Sayfa 616) kullanmayı veya donanım yazılımı güncellemesi (Sayfa 791) yapmak için STEP 7'nin çevrimiçi ve diyagnostik fonksiyonlarını kullanmayı içerir. Bu bölüm bir bellek kartını kullanan yöntemi açıklar.

##### UYARI

**Bellek kartını koruyunuz ve elektrostatik boşalmadan muhafaza ediniz.**

Elektrostatik boşalma, bellek kartına veya CPU üzerindeki yuvaya zarar verebilir.

Bellek kartıyla çalışırken topraklayıcı bir altlık (pad) ile temas sağlayınız ve/veya topraklı bir bileklik takınız. Bellek kartını iletken bir muhafaza içinde saklayınız.

Müşteri desteğinden (<http://www.siemens.com/automation/>) donanım yazılımı güncellemesi yüklerken bir bellek kartı kullanırsınız. Bu Web sitesinden **Automation Technology > Automation Systems > SIMATIC Industrial Automation Systems > PLC > Modular controllers SIMATIC S7 > SIMATIC S7-1200** adresine gidiniz. Oradan güncellemek istediğiniz özel tip modül için aramaya devam ediniz. "Support" altında, devam etmek üzere "Software Downloads" için link'i tıklayınız.

Alternatif olarak S7-1200 yüklemeler Web sayfasına doğrudan erişebilirsiniz (<http://support.automation.siemens.com/WW/view/en/34612486/133100>).

##### Not

Donanım yazılımı güncellemesiyle bir S7-1200 CPU V3.0 veya S7-1200 V4.0 için önceki versiyonları güncelleyemezsiniz.

##### UYARI

**Bellek kartını yeniden formatlamak için Windows formatlayıcı hizmetini veya diğer formatlama hizmetini kullanmayınız.**

Siemens bellek kartı Microsoft Windows formatlayıcı hizmeti kullanılarak yeniden formatlanırsa, bellek kartı bir S7-1200 CPU tarafından artık kullanılabilir olmayacaktır.

Bellek kartınıza bir donanım yazılımı güncellemesi yüklemek için şu adımları takip ediniz:

1. Bilgisayarınıza bağlı bir SD kart okuyucusuna/yazıcısına yazma koruması olmayan bir SIMATIC bellek kartı takınız. (Kart yazma korumalı ise koruma anahtarını "Lock" pozisyonundan çekiniz.)

Bir kullanıcı programı veya başka bir donanım yazılımı güncellemesi içeren bir SIMATIC bellek kartını yeniden kullanabilirsiniz, ancak kartı yeniden kullanmadan önce bazı program dosyalarını silmelisiniz.

Bellek kartını yeniden kullanmak için donanım yazılım güncellemesini yüklemeye başlamadan önce "S7\_OB.S7S" dosyası ve mevcut "Data Logs" klasörleri ve klasör dizinlerini de ("SIMATIC.S7S" veya "FWUPDATE.S7S" gibi) silmelisiniz. Bellek kartının içeriğini görüntülemek, dosya ve dizinleri silmek için Windows Gezgini kullanınız.

**UYARI**

**Bellek kartından "\_LOG\_" ve "crdinfo.bin" saklı dosyalarını silmeyiniz.**

"\_LOG\_" ve "crdinfo.bin" dosyaları bellek kartı için gereklidir. Bu dosyaları silerseniz bellek kartını CPU ile kullanamazsınız.

2. Modülünüze karşılık gelen donanım yazılımı güncelleştirme için kendini-açabilen dosyayı (.exe) seçiniz ve bilgisayarınıza yükleyiniz. Güncelleştirme dosyasına çift-tıklayınız ve çıkarma işlemini başlatınız. Çıkarma işlemi tamamlandıktan sonra bellek kartının kök dizini (klasör) bir "FWUPDATE.S7S" klasörü ve "S7\_JOB.S7S" dosyası bulunduracaktır.
3. Kart okuyucu/yazıcıdan kartı güvenli bir şekilde çıkartınız.

Donanım yazılımı güncellemesini kurmak için şu adımları takip ediniz:

 **WARNING**

**Donanım yazılımı güncellemesini kurmadan önce CPU'nun aktif olarak bir proses yürütmediğini doğrulayınız.**

Donanım yazılımı güncellemesini kurmak CPU'nun STOP moduna geçmesine neden olacaktır. Bu işlem, çevrimiçi bir proses veya makinanın çalışmasını etkileyebilecektir. Bir proses veya makinanın beklenmeyen çalışması personelin ölmesi veya ciddi yaralanmaları ve/veya mala zarar ile sonuçlanabilecektir.

Bellek kartını takmadan önce CPU'nun çevrim dışı (offline) ve güvenli durumda olduğundan daima emin olunuz.

1. Bellek kartını CPU'ya takınız. CPU, RUN modunda ise CPU sonra STOP moduna geçer. Bakım (MAINT) LED bellek kartının değerlendirilmesi ihtiyacını göstermek için yanar/söner.
2. Donanım yazılımı güncelleştirmesini başlatmak için CPU gücünü kesip geri veriniz. CPU'yu yeniden yüklemenin alternatif yöntemleri ya STOP'tan RUN'a geçiş ya da STEP 7'den bir kart resetlemesidir (MRES).

---

**Not**

Modül için donanım yazılım güncellemesini tamamlamak için 24 VDC harici gücün bu modülü beslemeye devam etmesinden emin olunuz.

CPU yeniden yüklendikten sonra, donanım yazılımı güncellemesi başlar. RUN/STOP LED, güncellenmenin kopyalanmakta olduğunu göstermek için alternatifli olarak yeşil ve sarı renginde yanar. RUN/STOP LED (koyu sarı) yandığında, MAINT LED yanıp söner ve kopyalama işlemi bitmiştir. Sonra bellek kartını çıkarabilirsiniz.

3. Bellek kartını çıkardıktan sonra, yeni donanım yazılımı güncellemesi için CPU'yu tekrar yeniden yükleyiniz (yeniden enerjileyerek veya yeniden yüklemenin alternatif yöntemleriyle)

Donanım yazılımı güncellemesinden kullanıcı programı ve donanım konfigürasyonu etkilenmez. CPU enerjilendiğinde, CPU konfigüre edildiği başlangıç durumuna girer. (CPU'nuz için başlangıç modu "Warm restart - mode before POWER OFF" olarak konfigüre edilmiş ise, CPU'nun son durumu STOP olduğundan CPU STOP modunda olacaktır.)

---

**Not****CPU'ya bağlı çoklu modüllerin güncellenmesi**

Donanım konfigürasyonunuz bellek kartı üzerinde tek bir donanım yazılım güncellemesine karşılık çoklu modülleri içerirse, konfigürasyon sırasına göre CPU bütün uygulanabilir modüllere (CM, SM ve SB) güncellemeleri uygular. Sıralama işlemi STEP'deki Cihaz Konfigürasyonundaki modül konumu sırasına göre artırılarak yapılır.

Çoklu modüller için bellek kartına çoklu donanım yazılım güncellemeleri yüklemişseniz, CPU güncellemeleri sizin bellek kartına onları yükleme sırasına göre uygular.

---

## 4.6 Kayıp bir şifrenin kurtarılması

Şifre korumalı bir CPU için şifrenizi kaybettiyseniz, şifre korumalı programı silmek için boş bir transfer kartı kullanınız. Boş transfer kartı CPU'nun dahili yükleme belleğini siler. Bundan sonra STEP 7'den CPU'ya yeni bir kullanıcı programı yükleyebilirsiniz.

Boş bir transfer kartının oluşturulması ve kullanımıyla ilgili bilgi almak için transfer kartları bölümüne (Sayfa 127) bakınız.

### UYARI

**Bir bellek kartını takmadan önce CPU'nun bir prosesi yürütmediğini doğrulayınız.**

Bir transfer kartını çalışmakta olan bir CPU'ya takarsanız, CPU STOP moduna geçer. Kontrol cihazları güvensiz bir duruma düşebilirler ve bunun sonucunda kontrollü ekipmanın beklenmeyen çalışması meydana gelebilir. Bu beklenmeyen işleyişler, kişisel yaralanmalara veya ölümlere ve/veya ekipman hasarlarına yol açabilir.

Bir transfer kartını takmadan önce CPU'nun STOP modunda ve prosesinizin güvenli bir durumda olduğundan daima emin olunuz.

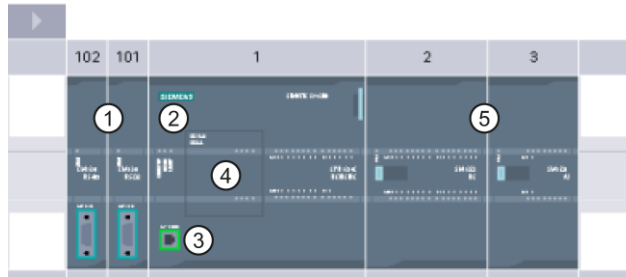
CPU'yu RUN moduna ayarlamadan önce transfer kartını güvenli bir şekilde çıkartmalısınız.



## S7Cihaz Konfigürasyonu

# 5

Projenize bir CPU ve ilave modüller ekleyerek PLC'niz için cihaz konfigürasyonu oluşturursunuz.



- ① Haberleşme modülü (CM) veya haberleşme işlemcisi (CP): 3 adete kadar, slot 101, 201 ve 103'e takılır
- ② CPU: Slot 1
- ③ CPU Ethernet portu
- ④ Sinyal kartı (SB), haberleşme kartı (CB) veya batarya kartı (BB): 1 adete kadar, CPU'ya takılır
- ⑤ Dijital veya analog I/O için sinyal modülü (SM): 8 adete kadar, slot 2'den slot 9'a kadar takılır (CPU 1214C, CPU 1215C ve CPU 1217C 8 slota izin verir, CPU 1212C 2 slota izin verir, CPU 1211C hiç birine izin vermez)

Cihaz konfigürasyonunu oluşturmak için projenize bir cihaz ekleyiniz.

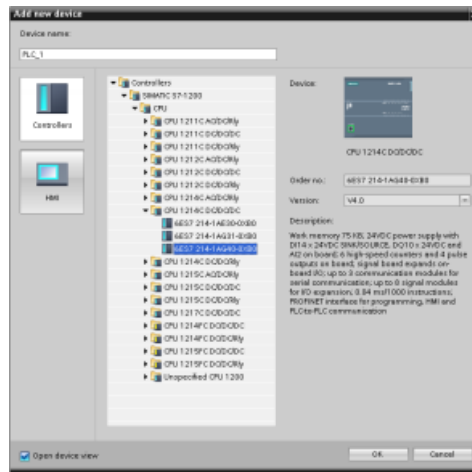
- Portal görünümünde, “Devices & Networks”ı seçiniz ve “Add new device” üzerine tıklayınız.
- Proje görünümünde, proje ismi altındaki “Add new device” üzerine çift tıklayınız.



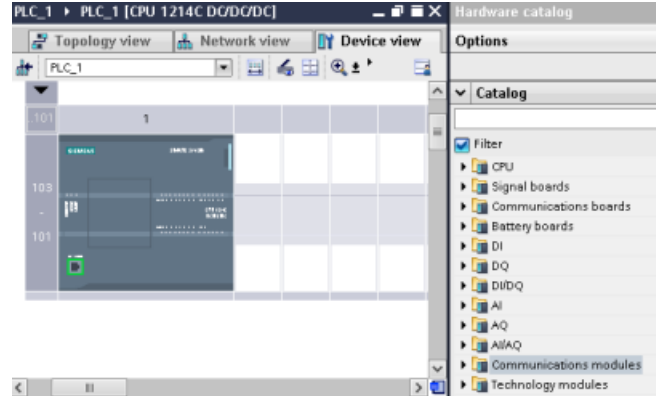
## 5.1 Bir CPU eklemek

Projenize bir CPU ekleyerek cihaz konfigürasyonunuzu oluşturursunuz. Listedeki doğru modül ve donanım yazılımı versiyonunu taktığınızdan emin olunuz ve “Add new device” diyalogundan CPU seçildiğinde şasi ve CPU oluşturulur.

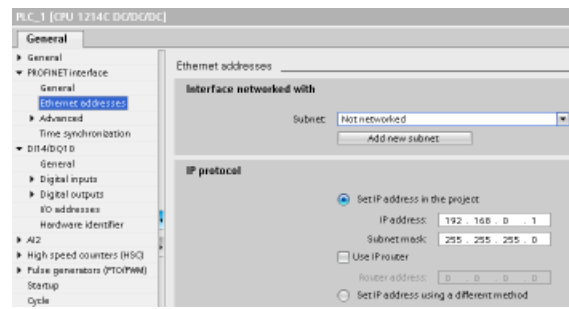
“Add new device” diyaloğu



Donanım konfigürasyonunun cihaz görünümü



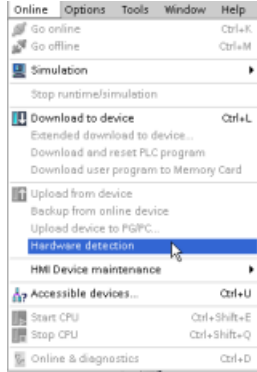
Cihaz görünümünden CPU seçildiğinde denetçi penceresinde CPU özellikleri görüntülenir.



### Not

CPU önceden yapılandırılmış bir IP adresine sahip değildir. Cihaz konfigürasyonu sırasında CPU için manuel olarak bir IP adresi atamanız gerekir. CPU'nuz ağ üzerindeki bir yönlendiriciye (router) bağlı ise, yönlendirici için de IP adresi girmeniz gerekir.

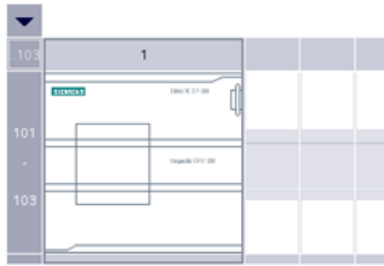
## 5.2 Belirtilmemiş bir CPU için konfigürasyon tespiti



Bir CPU'ya bağlıysanız, tüm modüllerle birlikte bu CPU'nun konfigürasyonunu projenize yükleyebilirsiniz. Basit olarak yeni bir proje oluşturunuz ve belirli bir CPU seçmek yerine "unspecified CPU"yu seçiniz. ("First steps"den "Create a PLC program"ı seçerek cihaz konfigürasyonunu tamamen atlayabilirsiniz). STEP 7 bundan sonra otomatik olarak belirtilmemiş bir CPU oluşturur.

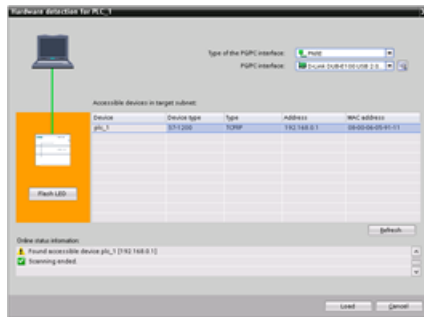
Program editöründeki "Online" menüsünden "Hardware detection" komutunu seçiniz.

Cihaz konfigürasyonu editöründen, bağlı cihazın konfigürasyonunu algılama (detect) seçeneği seçiniz.



The device is not specified.  
 → Please use the [hardware catalog](#) to specify the CPU.  
 → or [detect](#) the configuration of the connected device.

Online diyalogundan CPU'yu seçip yükle düğmesine tıkladıktan sonra STEP 7, tüm modüller de (SM, SB veya CM) dahil olmak üzere donanım konfigürasyonunu CPU'dan yükler. Bundan sonra CPU ve modüller için parametreleri konfigüre edebilirsiniz.



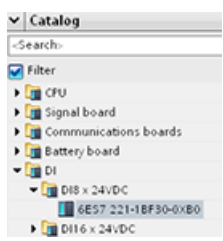


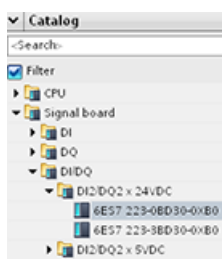


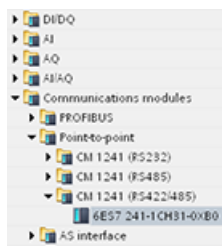


## 5.3 Konfigürasyona modüller eklemek

CPU'ya modül eklemek için donanım kataloğunu kullanınız:

- Sinyal modülü (SM) ilave dijital veya analog I/O noktaları sağlar. Bu modüller CPU'nun sağ tarafına bağlanırlar.
- Sinyal kartı (SB), CPU için birkaç ilave I/O noktası sağlar. SB, CPU'nun ön kısmına takılır.
- Batarya kartı 1297 (BB), gerçek zaman saatinin uzun-sürelili desteklenmesini sağlar. BB, CPU'nun ön kısmına takılır.
- Haberleşme kartı (CB), ilave bir haberleşme portu sağlar (RS485 gibi). CB, CPU'nun ön kısmına takılır.
- Haberleşme modülü (CM) ve haberleşme işlemcisi (CP), PROFIBUS veya GPRS gibi ilave bir haberleşme portu sağlar. Bu modüller CPU'nun sol tarafına bağlanırlar.

Donanım konfigürasyonuna bir modül eklemek için modülü donanım kataloğundan seçiniz ve modülün üzerine çift tıklayınız veya modülü (highlighted) vurgulu yuvaya sürükleyiniz. Modülleri cihaz konfigürasyonuna eklemelisiniz ve fonksiyonel olacak modüller için donanım konfigürasyonunu CPU'ya yüklemelisiniz.

Çizelge 5- 1 Bir modülü cihaz konfigürasyonuna eklemek

Modül	Modül seçme	Modül takma	Sonuç
SM			
SB, BB veya CB			
CM veya CP			

## 5.4 Bir cihazın değiştirilmesi

Bir CPU veya modülü konfigüre etmek için cihaz tipini değiştirebilirsiniz. Cihaz konfigürasyonundan cihaz üzerine sağ-tıklayınız ve içerik menüsünden Cihaz değiştirmeyi ("Change device") seçiniz. Diyalogdan değiştirmek istediğiniz CPU veya modülü seçiniz. Cihaz değiştirme diyalogu iki cihaz arasında size uygunluk bilgisini gösterir.

### Not

#### Cihaz değiştirme: Bir V3.0 CPU'yu bir V4.0 CPU ile değiştirme

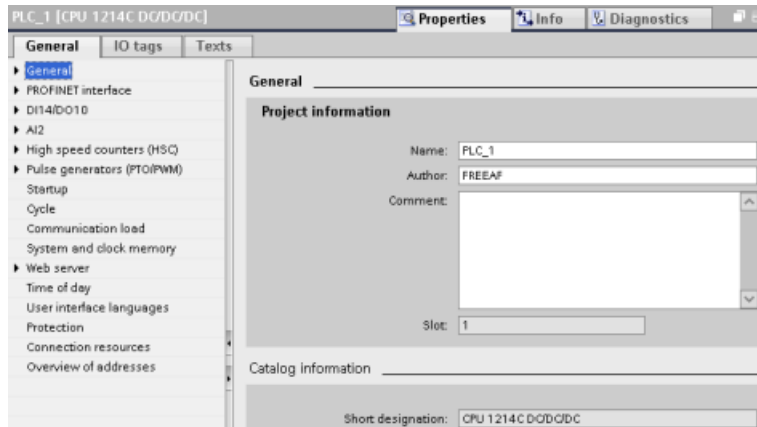
STEP 7 V13'de bir STEP 7 projesini açabilirsiniz ve V3.0 CPU'ları V4.0 CPU'lar ile değiştirebilirsiniz. V3.0 versiyonundan eski CPU'ları değiştiremezsiniz. V3.0 CPU'ları V4.0 CPU'lar ile değiştirirseniz özelliklerdeki ve iki versiyon arasındaki davranışlardaki farklılıklara (Sayfa 995) ve almanız gereken aksiyonlara dikkat ediniz.

V3.0'dan daha eski bir CPU versiyonu için bir projeye sahipseniz bu projeyi önce V3.0'e ve sonra V3.0'ı V4.0'a güncelleyebilirsiniz.

## 5.5 CPU'nun çalışmasını konfigüre etmek

### 5.5.1 Genel bakış

CPU'nun çalışma parametrelerini konfigüre etmek için cihaz görünümünden CPU'yu seçiniz (tüm CPU etrafında mavi bir çerçeve) ve denetçi penceresinin "Properties" sekmesini kullanınız.



Çizelge 5- 2 CPU özellikleri

Özellik	Açıklama
PROFINET ara yüzü	CPU ve zaman senkronizasyonu için IP adresini ayarlar
DI, DO ve AI	Lokal dijital ve analog I/O'nun davranışını konfigüre eder (örneğin dijital filtre süresini ve bir CPU'ya durması için dijital çıkış tepkisini)
Yüksek hızlı sayıcılar (Sayfa 409) ve darbe jeneratörleri (Sayfa 365)	Darbe katarı işlemleri (PTO) ve darbe genişlik modülasyonu (PWM) için kullanılan yüksek hızlı sayıcıları (HSC) ve darbe jeneratörlerini etkinleştirir ve yapılandırır. CPU'nun çıkışlarını ya da sinyal kartlarının çıkışlarını darbe jeneratörleri olarak (PWM veya temel hareket kontrolü komutları ile kullanım için) yapılandırırken, karşılık gelen çıkış adresleri (Q0.0, Q0.1, Q4.0 ve Q4.1) Q hafızasından çıkarılır ve kullanıcı programınızda başka amaçlar için kullanılmaz. Kullanıcı programınız darbe jeneratörü olarak kullanılan bir çıkışa bir değer yazarsa, CPU bu değeri fiziksel çıkışa yazmaz.
Başlangıç (Sayfa 81)	<p><b>Enerjilendikten sonraki başlangıç modu:</b> STOP modunda başlamak veya yeniden sıcak başlatmadan sonra RUN moduna geçmek gibi CPU'nun davranışını seçer.</p> <p><b>Desteklenen donanım uyumluluğu:</b> Hangi sistem bileşenlerinin yedekle değiştirilebilmesini konfigüre eder (SM, SB, CM, CP ve CPU):</p> <ul style="list-style-type: none"> <li>Kabul edilebilir yedeğe izin ver</li> <li>Herhangi bir yedeğe izin ver (varsayılan)</li> </ul> <p>Her modül; I/O sayısına, elektriksel uyumluluğa ve diğer karşılık gelen kıyaslama noktalarına dayalı değiştirme uyumluluğu kurallarını dahili olarak içerir. Örneğin bir 16-kanal SM bir 8-kanal SM için kabul edilebilir bir yedek olabilir, ancak bir 8-kanal SM bir 16-kanal SM için kabul edilebilir bir yedek olamaz. "Allow acceptable substitute" işlemini seçerseniz, STEP 7 değiştirme kurallarını dayatır, diğer durumlarda her hangi bir değişikliğe izin verir.</p> <p><b>Dağıtılmış I/O için I/O parametre ataması:</b> Dağıtılmış I/O'nun çevrimiçi hale getirilmesi için gerekli olan maksimum zaman miktarını yapılandırır (varsayılan 60000 ms). (CM'ler ve CP'ler güç ve haberleşme parametrelerini başlangıç sırasında CPU'dan alır). Bu atama süresi, çevrimiçi hale getirilecek CM'ye veya CP'ye bağlı I/O için süre sağlar.</p> <p>CPU, atama zamanından bağımsız olarak dağıtılmış I/O çevrimiçi olur olmaz RUN moduna geçer. Dağıtılmış I/O bu sürede çevrimiçi hale getirilmezse, CPU dağıtılmış I/O olmadan da RUN moduna geçer.</p> <p><b>Not:</b> Konfigürasyonunuz bir CM1243-5 (PROFIBUS master) kullanıyorsa, modülün çevrimiçi hale getirilebilmesini temin etmek için bu parametreyi 15 saniyenin (15000 ms) altına ayarlamayınız.</p>
Döngü (Sayfa 97)	Maksimum döngü süresi veya sabit bir minimum döngü süresi tanımlar.
Haberleşme yükü	Haberleşme görevlerine tahsis edilecek CPU süresinin yüzdesini ayarlar.
Sistem ve saat belleği (Sayfa 101)	"Sistem belleği" fonksiyonları için bir bayt ve "saat belleği" fonksiyonları için bir bayt etkinleştirir (her bir bit açık ve kapalı durumları arasında önceden belirlenmiş bir frekansta geçiş yapar)
Web sunucu (Sayfa 601)	Web sunucusu özelliğini etkinleştirir ve yapılandırır.
Günün saati	Saat dilimini seçer ve yaz saati zamanını yapılandırır.
Kullanıcı ara yüz dilleri	Proje diline karşılık gelen Web sunucu ve CPU display'i için dili ayarlar. İki proje diline kadar Web sunucu ve CPU display'i için dil ataması yapabilirsiniz.
Koruma (Sayfa 190)	Okuma/yazma koruması ile CPU'ya erişim şifresini ayarlar
Bağlantı kaynakları (Sayfa 611)	CPU için kullanılabilir olan haberleşme bağlantılarının bir özetini ve yapılandırılmış olan bağlantıların sayısını sunar.
Adreslerin özeti	CPU için yapılandırılan I/O adreslerinin bir özetini sunar.

## 5.5.2 Dijital giriş filtrelerinin konfigürasyonu

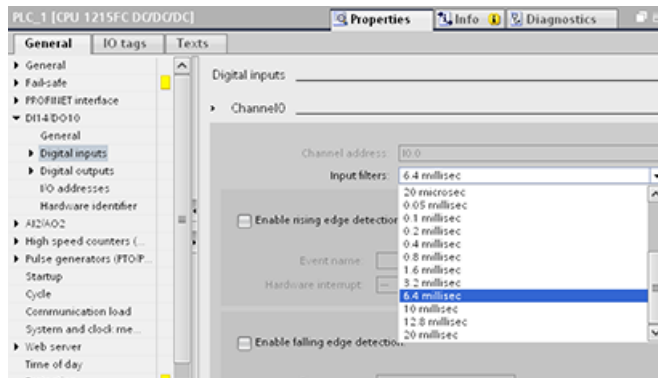
Dijital giriş filtreleri, programınızın giriş sinyallerindeki istenmeyen hızlı değişimlere tepki göstermesine karşı koruma görevi yaparlar. Hızlı değişimler anahtar kontaklarından veya elektrik gürültülerinden ortaya çıkabilir. Varsayılan filtre süresi 6.4 ms'dir ve mekanik kontaklardan kaynaklanan istenmeyen geçişleri bloka eder. Uygulamanızdaki farklı noktalar hızlı sensörlerden gelen girişleri algılamak ve cevap vermek için daha kısa filtre sürelerine veya yavaş kontak açma kapaması veya daha uzun süreli darbe gürültüsü için daha uzun filtre süresine gerek duyabilirler

Bir giriş filtreleme süresinin 6.4 ms olması, tek bir sinyalin '0' dan '1' e veya '1' den '0' a değişiminin algılanabilmesi için sinyaldeki değişimin yaklaşık 6.4 ms sürmesi ve tek bir yüksek veya düşük palsın yaklaşık olarak 6.4 ms 'den daha kısa olması o palsın algılanamayacağı anlamına gelir.

Giriş sinyalinin '0' ve '1' arasındaki anahtarlama süresi filtre süresinden daha hızlı değişirse, eski değerli palslar üzerindeki yeni değerli palsların biriken süresi filtre süresini aştığında kullanıcı programındaki giriş noktası değeri değişir.

Dijital giriş filtresi çalışması:

- Giriş "1" olduğunda, filtre süresine kadar sayar. Sayma filtre süresine ulaştığında, görüntü kayıt noktası "0" dan "1" e değişir.
- Giriş "0" olduğunda, "0" da duruncaya kadar azalarak sayar. Sayma "0" a ulaştığında, görüntü kayıt noktası "1" den "0" a değişir.
- Giriş ileri ve geri değişiyorsa, sayıcı bazı artan sayma işlemi ve bazı aşağı sayma işlemi yapacaktır. Net biriken saymalar filtre süresine veya "0" a ulaştığında, görüntü kaydedici değişecektir.
- Hızlı olarak, sinyalin "0'lardaki" değişimi "1'lerdeki" değişiminden hızlı ise sonuçta "0" a gidecektir. Sinyalin "1'lerdeki" değişimi "0'lardaki" değişiminden hızlı ise sonuçta "1" e değişecektir.



Her giriş noktası, tek bir filtre konfigürasyonuna sahiptir ve proses girişler, interrupt'lar, pals yakalama ve HSC girişlerinin kullanımına uygulanır. Giriş filtre sürelerini konfigüre etmek için "Digital Inputs" seçilir.



Dijital girişler için varsayılan filtre süresi 6.4 ms'dir. Aşağıdaki filtre süresi değerlerinden herhangi birini seçebilirsiniz:

- 0.1 us
- 0.2 us
- 0.4 us
- 0.8 us
- 1.6 us
- 3.2 us
- 6.4 us
- 10.0 us
- 12.8 us
- 20.0 us
- 0.05 ms
- 0.1 ms
- 0.2 ms
- 0.4 ms
- 0.8 ms
- 1.6 ms
- 3.2 ms
- 6.4 ms
- 10.0 ms
- 12.8 ms
- 20.0 ms

 **UYARI****Dijital giriş kanalı için filtre süresindeki değişikliklerin riskleri**

Bir dijital giriş kanalı için filtre süresi önceki bir ayardan değiştirilirse, yeni bir "0" seviye giriş değeri 20.0 ms'ye kadar sunulması gerekebilir. Bu, filtre yeni girişlere tam tepkili duruma gelme öncesine kadar biriken süredir. Bu süre boyunca, 20.0 ms'den kısa süreli "0" pals olayları algılanmayabilir veya sayılamayabilir.

Filtre sürelerindeki bu değişiklik beklenmeyen makine veya proses işlemleriyle sonuçlanabilir. Bu durum, personelin ölümüne veya ciddi yaralanmalarına ve/veya ekipmana zarara yol açabilir.

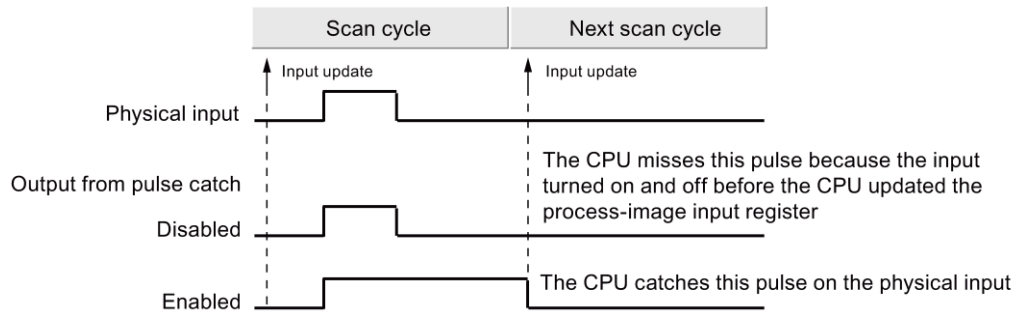
Yeni filtre süresinin hemen etkin olmasını temin etmek için CPU'nun bir güç çevrimi (kapama açma) uygulanmalıdır.

### 5.5.3 Pals yakalama

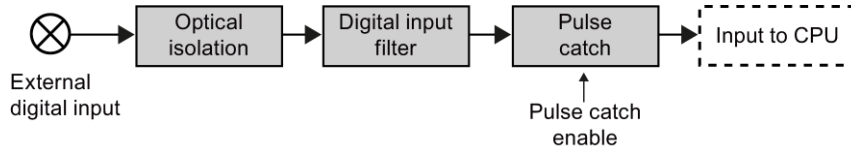
S7-1200 CPU, dijital giriş noktaları için bir pals yakalama özelliği sağlar. Pals yakalama özelliği yüksek-değişen palsları veya düşük-değişen palsları yakalamanıza imkan sağlar. Bu palsların süreleri öyle kısadır ki CPU tarama döngüsünün başlangıcındaki dijital girişleri okurken bu palsları daima göremeyebilir.

Bir giriş için pals yakalama etkinleştirildiğinde, girişin durumundaki bir değişiklik yakalanır ve gelecek giriş döngü güncellemesine kadar tutulur. Bu durum, kısa süreli bir periyotta sona eren bir palsın yakalanacağını ve CPU girişleri okuyuncaya kadar tutulacağını temin eder.

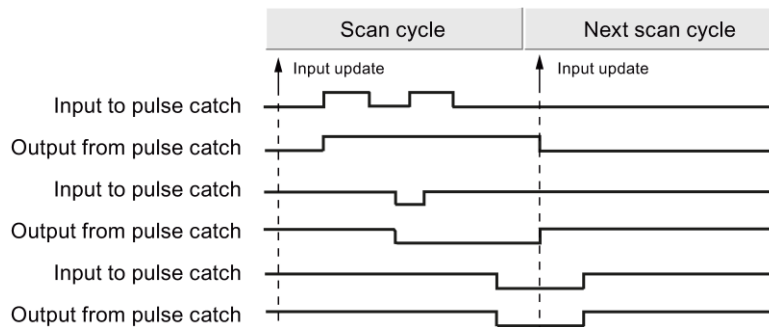
Aşağıdaki şekil pals yakalaması etkin olan ve etkin olmayan S7-1200 CPU'nun temel operasyonunu göstermektedir:



Pals yakalama fonksiyonu giriş üzerindeki işlemi giriş filtresini geçtikten sonra yaptığı için giriş filtre süresini pals filtre tarafından kaldırılmayacak şekilde ayarlamalısınız. Aşağıdaki şekil dijital giriş devresinin bir blok diyagramını göstermektedir:



Aşağıdaki şekil değişik giriş durumları için etkinleştirilmiş pals yakalama fonksiyonu tepkisini göstermektedir. Verilen bir taramada birden fazla palslar var ise, sadece ilk pals okunur. Verilen bir taramada çoklu palslar var ise, yükselen/azalan kenar interrupt olaylarını kullanmalısınız:



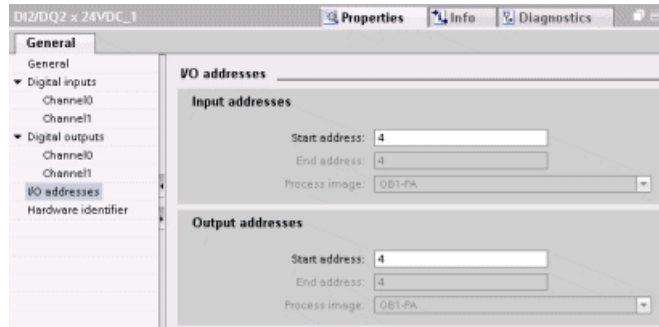
## 5.6 Modüllerin parametrelerini konfigüre etmek

Modüller için işletimsel parametreleri yapılandırmak için Cihaz görünümündeki modülü seçiniz ve modül için parametreleri konfigüre etmek üzere denetçi penceresinin "Properties" sekmesini kullanınız.

### Bir sinyal modülünü (SM) veya bir sinyal kartını (SB) konfigüre etmek

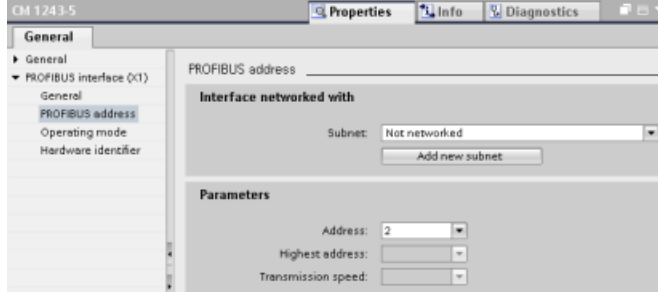
Sinyal modülleri ve sinyal kartları için cihaz konfigürasyonu aşağıdakileri konfigüre etmek için vasıta sağlar:

- Dijital I/O: Girişler yükselen kenar algılama veya düşen kenar algılama için (her birini bir olay veya donanım interrupt'ı ile ilişkilendirerek) veya giriş proses görüntüsünün bir sonraki güncellemesi aracılığıyla "darbe yakalama" (bir anlık darbeden sonra girişin "on" olarak kalması) için konfigüre edilebilir. Çıkışlar bir donma ya da yedek değer kullanabilir.
- Analog I/O: Bağımsız girişler için parametreleri konfigüre ediniz. Bu parametreler ölçme tipi (gerilim veya akım), aralık ve düzleştirme, alt taşma veya taşma diyagnostiğinin etkinleştirilmesi için olabilir. Analog çıkışlar; çıkış tipi (gerilim veya akım) ve diyagnostik kısa devre (gerilim çıkışları için) ya da üst/alt limit diyagnostikleri için parametreler sağlar. Analog girişlerin ve çıkışların mühendislik birimlerine dönüşümü için aralıkları Özellikler (Properties) diyalogundan konfigüre etmeyiniz. Bunu, "Analog değerlerin işlenmesi (Sayfa 110)." konusunda açıklandığı gibi kullanıcı programı lojiğinde yapmalısınız
- I/O adresleri: Modülün girişleri ve çıkışları setinin başlama adresini konfigüre ediniz.
- Girişleri ve çıkışları da bir proses görüntü bölmesine (PIP0, PIP1, PIP2, PIP3, PIP4) veya otomatik olarak güncellemeye veya proses görüntü bölmesi kullanılmaya atama yapabilirsiniz. Proses görüntü ve proses görüntü bölmelerinin bir açıklaması için "Execution of the user program"e (Sayfa 77) bakınız.



**Bir haberleşme ara yüzünü konfigüre etmek (CM, CP veya CB)**

Haberleşme ara yüzünün tipine bağlı olarak ağ için parametreleri konfigüre ediniz.

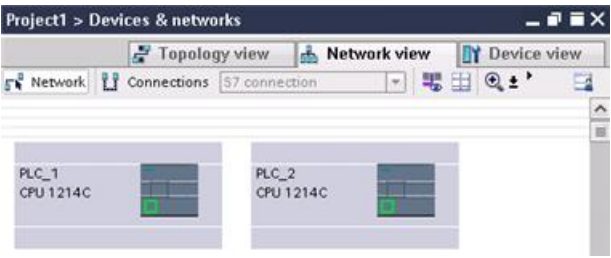
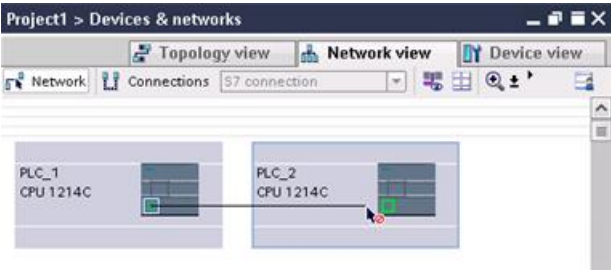
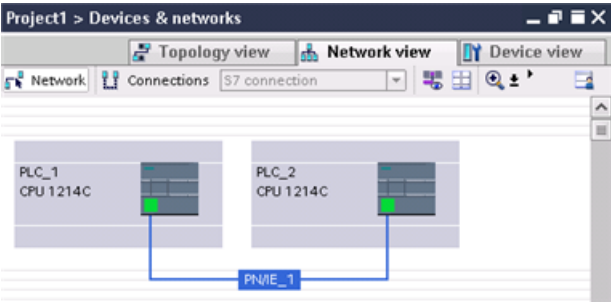


## 5.7 Haberleşme için CPU'yu konfigüre etmek

### 5.7.1 Bir ağ bağlantısı oluşturmak

Projenizde yer alan cihazlar arasında ağ bağlantıları oluşturmak için Cihaz konfigürasyonunun "Network view" ini kullanınız. Ağ bağlantısını oluşturduktan sonra, ağın parametrelerini konfigüre etmek için denetçinin "Properties" sekmesini kullanınız.

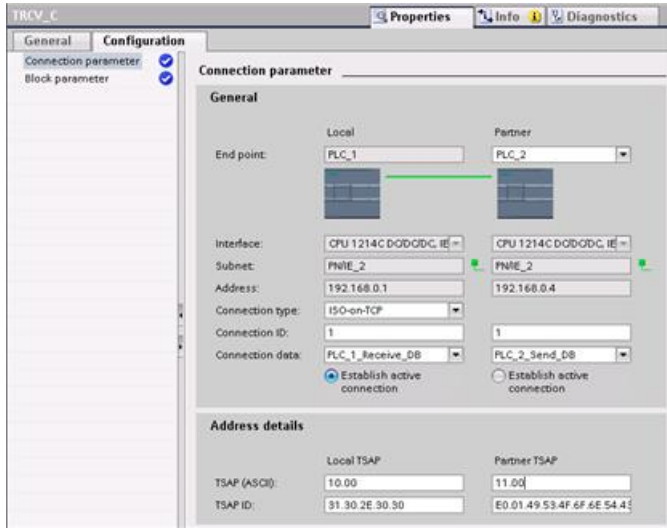
Çizelge 5- 3 Bir ağ bağlantısının oluşturulması

Aksiyon	Sonuç
Bağlanacak cihazları göstermek için "Network view" i seçiniz.	
Cihaz üzerindeki portu seçiniz ve bağlantıyı ikinci cihaz üzerindeki porta sürükleyiniz.	
Ağ bağlantısını oluşturmak için fare düğmesini bırakınız.	

## 5.7.2 Lokal/Partner bağlantı yolunun konfigürasyonu

Bir TSEND\_C, TRCV\_C veya TCON komutunun kullanıcı programına yerleştirilmesi sonrası, komutun parçalarından herhangi birinin seçişinizde denetçi penceresi bağlantının özelliklerini görüntüler. Haberleşme komutu için "Özellikler" ("Properties") sekmesindeki "Configuration" dan haberleşme parametrelerini belirleyiniz.

Çizelge 5- 4 Bağlantı yolunun konfigürasyonu (komutun özelliklerini kullanarak)

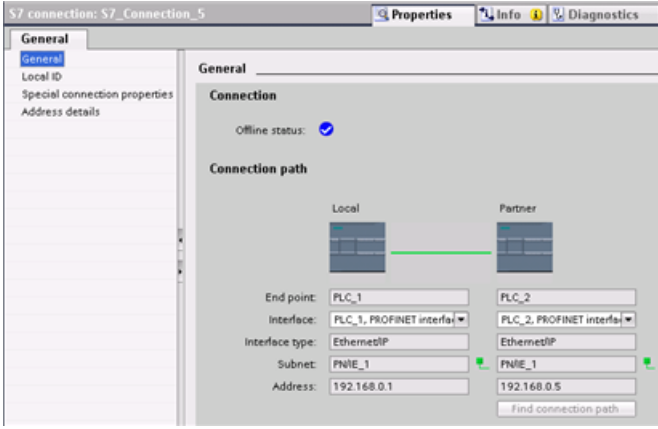
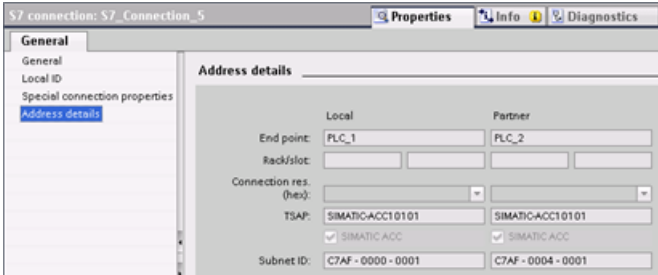
TCP, ISO-on-TCP ve UDP	Bağlantı özellikleri
<p>TCP, ISO-on-TCP ve UDP Ethernet protokolleri için, "Lokal/Partner" bağlantılarını yapılandırmak üzere komutun "Özelliklerini" ("Properties") kullanınız (TSEND_C, TRCV_C veya TCON). Resimde, bir ISO-on-TCP bağlantısı için "Konfigürasyon sekmesinin" ("Configuration tab") bağlantı özellikleri gösterilmektedir.</p>	

### Not

Bir CPU için bağlantı özelliklerini yapılandırırken, STEP 7 size partner CPU'daki belirli bir bağlantı DB'sini seçmenize (eğer bir tane varsa) veya partner CPU için bağlantı DB'si oluşturmanıza izin verir. Partner CPU proje için oluşturulmuş olmalıdır; bir "unspecified" CPU olamaz.

Yine de partner CPU'nun kullanıcı programına bir TSEND\_C, TRCV\_C veya TCON komutu eklemeniz gerekir. Komutu eklerken konfigürasyon tarafından üretilen bağlantı DB'sini seçiniz.

Çizelge 5- 5 S7 haberleşmesi için bağlantı yolunun konfigürasyonu (Cihaz konfigürasyonu)

S7 haberleşme (GET ve PUT)	Bağlantı özellikleri
<p>S7 haberleşmesi için, Lokal/Partner bağlantılarını yapılandırmak için ağın "Devices &amp; Networks" editörünü kullanın. Özelliklere girmek için "İşaretlenmiş Bağlantı" üzerine tıklayabilirsiniz.</p> <p>"Genel" ("General") sekmesi birkaç özellik sunar:</p> <ul style="list-style-type: none"> <li>"Genel" (gösterilen)</li> <li>"Lokal ID"</li> <li>"Özel bağlantı özellikleri"</li> <li>"Adres ayrıntıları" ("Address details") (gösterilen)</li> </ul>	 

Daha fazla bilgi ve kullanılabilir haberleşme komutlarının bir listesini almak için "S7 haberleşmesi" ("S7 communication") bölümündeki "Bir S7 bağlantısı oluşturmak" ("Creating an S7 connection") (Sayfa 591) ya da "PROFINET" bölümündeki "Protokoller" (Sayfa 517) bölümüne bakınız.

## S7Cihaz Konfigürasyonu

### 5.7 Haberleşme için CPU'yu konfigüre etmek

Çizelge 5-6 Çoklu CPU bağlantısı için parametreler

Parametre		Açıklama
Adres		Atanmış IP adresleri
Genel	Son nokta	Partner (alıcı) CPU'ya atanan isim
	Ara yüz	Ara yüzlere atanan isim
	Alt ağ	Alt ağlara atanan isim
	Ara yüz tipi	<i>Sadece S7 haberleşmesi:</i> Ara yüzün tipi
	Bağlantı tipi	Ethernet protokolünün tipi
	Bağlantı ID	Kimlik numarası
	Bağlantı verisi	Lokal ve Partner CPU veri depolama konumu
	Aktif bağlantı gerçekleştir	Aktif bağlantı olarak Lokal veya Partner CPU seçimi için radyo butonu
Adres ayrıntıları	Son nokta	<i>Sadece S7 haberleşmesi:</i> Partner (alıcı) CPU'ya atanan isim
	Şasi/slot	<i>Sadece S7 haberleşmesi:</i> Şasi ve slot konumu
	Bağlantı kaynağı	<i>Sadece S7 haberleşmesi:</i> Bir S7-300 veya S7-400 CPU ile bir S7 bağlantısı konfigüre edilirken kullanılan TSAP bileşeni
	Port (decimal):	TCP ve UDP: Ondalık formatta partner CPU portu
	TSAP <sup>1</sup> ve Subnet ID:	ISO on TCP (RFC 1006) ve S7 haberleşmesi: ASCII ve hexadecimal formatlarda lokal ve partner CPU TSAP'leri

<sup>1</sup> ISO-on-TCP için S7-1200'le bir bağlantı konfigüre edilirken, pasif haberleşme partnerleri için sadece TSAP genişlemesindeki ASCII karakterlerini kullanınız.

### Taşıma Hizmet Erişim Noktaları (TSAP'ler)

TSAP'ler, ISO on TCP protokolü ve S7 haberleşme kullanılması tek bir IP adresine çoklu bağlantı yapılmasına izin verir (64K bağlantılara kadar). TSAP'ler, bu haberleşme son nokta bağlantılarını bir IP adresine özel olarak tanımlarlar.

Bağlantı Parametreleri diyalogunun "Adres Ayrıntıları" ("Address Details") bölümünde, kullanılacak TSAP'leri tanımlarsınız. CPU'daki bir bağlantının TSAP'i, "Lokal TSAP" alanına girilir. Partner CPU'nuzdaki bağlantı için atanan TSAP, "Partner TSAP" alanı altında girilir.

### Port Numaraları

TCP ve UDP protokolleriyle, Lokal (aktif) bağlantı CPU'sunun bağlantı parametre konfigürasyonu uzak IP adresini ve Partner (pasif) bağlantı CPU'sunun port numarasını belirtmelidir.

Bağlantı Parametreleri diyalogunun "Adres Ayrıntıları" ("Address Details") bölümünde, kullanılacak portları tanımlarsınız. CPU'daki bir bağlantının portu, "Lokal Port" alanına girilir. Partner CPU'nuzdaki bağlantı için atanan port, "Partner Port" alanı altında girilir.



### 5.7.3 PROFINET bağlantısı için parametreler

TSEND\_C, TRCV\_C ve TCON komutları, partner cihaza bağlanmak için bağlantıyla ilgili parametrelerin belirlenmesini gerektirir. TCP, ISO on TCP ve UDP protokolleri için bu parametreler TCON\_Param yapısı tarafından atanır. Bu parametreleri belirlemek için komutun "Özelliklerindeki" ("Properties") "Konfigürasyon" sekmesi kullanılır. Konfigürasyon sekmesi erişilebilir değilse, TCON\_Param yapısını programsal olarak belirlemeniz gerekir.

#### TCON\_Param

Çizelge 5- 7 Bağlantı açıklamasının yapısı (TCON\_Param)

Bayt	Parametre ve veri tipi	Açıklama
0 ... 1	block_length UInt	Uzunluk: 64 bayt (sabit)
2 ... 3	id CONN_OUC (Word)	Bu bağlantıya referans: Değerlerin aralığı: 1 (varsayılan) - 4095. TSEND_C, TRCV_C veya ID altındaki TCON komutu için bu parametre değerini belirleyiniz.
4	connection_type USInt	Bağlantı tipi: <ul style="list-style-type: none"> <li>• 17: TCP (varsayılan)</li> <li>• 18: ISO-on-TCP</li> <li>• 19: UDP</li> </ul>
5	active_est Bool	ID, bağlantı tipi için: <ul style="list-style-type: none"> <li>• TCP ve ISO-on-TCP: <ul style="list-style-type: none"> <li>- FALSE: Pasif bağlantı</li> <li>- TRUE: Aktif bağlantı (varsayılan)</li> </ul> </li> <li>• UDP: FALSE</li> </ul>
6	local_device_id USInt	ID, lokal PROFINET veya endüstriyel Ethernet ara yüzü için: 1 (varsayılan)
7	local_tsap_id_len USInt	Kullanılan parametre uzunluğu local_tsap_id, bayt; mümkün değerler: <ul style="list-style-type: none"> <li>• TCP: 0 (aktif, varsayılan) veya 2 (pasif)</li> <li>• ISO-on-TCP: 2 - 16</li> <li>• UDP: 2</li> </ul>
8	rem_subnet_id_len USInt	Bu parametre kullanılmaz.
9	rem_staddr_len USInt	Partner son noktasının adres uzunluğu, bayt: <ul style="list-style-type: none"> <li>• 0: belirlenmemiş (parametre rem_staddr ilgili değil)</li> <li>• 4 (varsayılan): Geçerli IP adresi, rem_staddr parametresinde (sadece TCP ve ISO-on-TCP için)</li> </ul>
10	rem_tsap_id_len USInt	Kullanılan parametre uzunluğu rem_tsap_id, bayt; mümkün değerler: <ul style="list-style-type: none"> <li>• TCP: 0 (pasif) veya 2 (aktif, (varsayılan))</li> <li>• ISO-on-TCP: 2 -16</li> <li>• UDP: 0</li> </ul>
11	next_staddr_len USInt	Bu parametre kullanılmaz.

## S7Cihaz Konfigürasyonu

### 5.7 Haberleşme için CPU'yu konfigüre etmek

Bayt	Parametre ve veri tipi		Açıklama
12 ... 27	local_tsap_id	Bayt dizini [1..16]	<p>Bağlantının lokal adres bileşeni:</p> <ul style="list-style-type: none"> <li>TCP ve ISO-on-TCP: lokal port no. (muhtemel veriler: 1 - 49151; önerilen değerler: 2000...5000): <ul style="list-style-type: none"> <li>local_tsap_id[1] = port numarasının yüksek baytı, hexadecimal notasyonda;</li> <li>local_tsap_id[2] = port numarasının düşük baytı, hexadecimal notasyonda;</li> <li>local_tsap_id[3-16] = ilgisiz</li> </ul> </li> <li>ISO-on-TCP: local TSAP-ID: <ul style="list-style-type: none"> <li>local_tsap_id[1] = B#16#E0;</li> <li>local_tsap_id[2] = lokal son noktalar için şasi ve slot (bit 0 - 4: slot numarası, bit 5 * 7: şasi numarası);</li> <li>local_tsap_id[3-16] = TSAP genişleme, opsiyonel</li> </ul> </li> <li>UDP: Bu parametre kullanılmaz.</li> </ul> <p>Not: local_tsap_id için her değer CPU'da özeldir.</p>
28 ... 33	rem_subnet_id	USInt için dizin [1..6]	Bu parametre kullanılmaz.
34 ... 39	rem_staddr	USInt için dizin [1..6]	<p>TCP ve ISO-on-TCP sadece: partner son nokta için IP adres (pasif elemanlar için ilgili değildir) Örneğin, IP adres 192.168.002.003 dizinin aşağıdaki elemanlarında depolanır:</p> <p>rem_staddr[1] = 192  rem_staddr[2] = 168  rem_staddr[3] = 002  rem_staddr[4] = 003  rem_staddr[5-6]= geçersiz</p>
40 ... 55	rem_tsap_id	Bayt dizini [1..16]	<p>Bağlantının Partner adres bileşeni:</p> <ul style="list-style-type: none"> <li>TCP: partner port numarası. Aralık: 1 - 49151; Önerilen değerler: 2000 - 5000: <ul style="list-style-type: none"> <li>rem_tsap_id[1] = port numarasının yüksek baytı, hexadecimal notasyonda;</li> <li>rem_tsap_id[2] = port numarasının düşük baytı, hexadecimal notasyonda;</li> <li>rem_tsap_id[3-16] = ilgisiz</li> </ul> </li> <li>ISO-on-TCP: partner TSAP-ID: <ul style="list-style-type: none"> <li>rem_tsap_id[1] = B#16#E0</li> <li>rem_tsap_id[2] = partner son noktası için şasi ve slot numarası (bit 0 - 4: Slot numarası, bit 5 - 7: şasi numarası)</li> <li>rem_tsap_id[3-16] = TSAP genişleme, opsiyonel</li> </ul> </li> <li>UDP: Bu parametre kullanılmaz.</li> </ul>
56 ... 61	next_staddr	Bayt dizini [1..16]	Bu parametre kullanılmaz.
62 ... 63	spare	Word	Rezerv edilmiş: W#16#0000

**Ayrıca bakınız**

Lokal/partner bağlantı yolunun konfigürasyonu (Sayfa 150)

**5.7.4 İnternet Protokol (IP) adreslerinin atanması****5.7.4.1 Programlama ve ağ cihazları için IP adreslerinin atanması**

Programlama cihazınız tesis LAN'ınıza (ve muhtemelen internete) bağlı bir yerleşik adaptör kartı kullanıyorsa, CPU'nuzun ve programlama cihazının yerleşik adaptör kartının IP Adresi Ağ Kimliği ve alt ağ maskesi birebir aynı olmalıdır. Ağ Kimliği ID, IP adresinin ilk kısmıdır (ilk üç sekizli) (örneğin 211.154.184.16) ve hangi IP ağında olduğunuzu belirler. Alt ağ maskesi normal olarak 255.255.255.0 değerine sahiptir; ancak bilgisayarınız bir tesis LAN'ında olduğundan, özel alt ağlar kurmak için alt ağ maskesi çeşitli değerler alabilir (örneğin 255.255.254.0). Alt ağ maskesi, bir matematiksel AND işleminde cihaz IP adresi ile birleştirildiğinde, bir IP alt ağının sınırlarını tanımlar.

**Not**

Programlama cihazlarınızın, ağ cihazlarının ve IP yönlendiricilerin dünya ile haberleşme sağladığı bir www senaryosunda, diğer ağ kullanıcıları ile çakışmadan kaçınmak için özel IP adresleri tahsis edilmelidir. IP adreslerinin tahsis edilmesiyle ilgili olarak, şirketinizin tesis ağlarınız hakkında bilgi sahibi olan BT bölümü personeliyle iletişim kurunuz.

 **UYARI****Web sunucu yoluyla CPU'ya yetkisiz erişimi**

CPU'ya yetkisiz erişim veya PLC değişkenlerini geçersizize değiştirme proses işlemini bozabilir ve personel ölümü, yaralanması ve/veya mala hasarla sonuçlanabilir.

Web sunucunun etkinleştirilmesi yetkisiz kullanıcıların işletim modu değişiklikleri, PLC'ye veri yazma ve donanım yazılımı güncellemesi yapmasına izin verir. Siemens aşağıdaki güvenlik uygulamalarını yerine getirmenizi önerir:

- Web sunucuya erişimi etkinleştirmeyi HTTPS protokolü ile yapınız.
- Web sunucu kullanıcı ID (Sayfa 604) şifre-korumasını sadece güçlü bir şifre ile yapınız. Güçlü şifre en azında 8 karakter uzunluğunda, karışık harfler, sayılar ve özel karakterlerden oluşmalıdır ve bir sözlükte bulunmamalıdır ve kişisel bilgilerden, isimler veya tanımlayıcılarda türetilmiş olmamalıdır. Şifreniz saklı tutunuz ve sıkça değiştiriniz.
- "Herkes" kullanıcısının varsayılan minimum ayrıcalığı genişletmeyiniz.
- Web sayfası kullanıcıları PLC değişkenlerini geçersizize değiştirebilecekleri için programlama lojiginizde değişkenleriniz üzerine hata-kontrolünü ve aralık kontrolünü yapınız.

## S7Cihaz Konfigürasyonu

### 5.7 Haberleşme için CPU'yu konfigüre etmek

Programlama cihazınız izole edilmiş bir ağa bağlantı için bir Ethernet-USB adaptör kartı kullanıyorsa, IP Adres Ağ ID ve CPU'nuzun alt ağ maskesi ve Ethernet-USB adaptör kartı programlama cihazı tamamen aynı olmalıdır. Ağ ID, IP adresinin ilk kısmıdır (ilk üç grup) (örneğin, **211.154.184.16**) ve sizin hangi IP ağı üzerinde olduğunuzu belirler. Alt ağ maskesi normalde **255.255.255.0** değerine sahiptir. Alt ağ maskesi bir matematiksel AND işleminde cihaz IP adresi ile birleştirildiğinde bir IP alt ağın sınırlarını belirler.

#### Not

Bir Ethernet-USB adaptör kartı, programlama kartınızın şirket LAN'ı üzerinde olmasını istemediğiniz durumda kullanışlıdır. Başlangıç denemesi veya işletmeye alma denemesi süresince bu düzenleme özellikle kullanışlıdır.

Çizelge 5- 8 Ethernet adreslerinin atanması

Programlama Cihazı Adaptör kartı	Ağ tipi	İnternet Protokol (IP) Adresi	Alt ağ maskesi
Yerleşik adaptör kartı	Tesis LAN'ınıza (ve muhtemelen internete) bağlıdır	CPU'nuzun ve programlama cihazınızın yerleşik adaptör kartının ağ kimliği birbiriyle aynı olmalıdır <sup>1</sup> .	CPU'nuzun ve yerleşik adaptör kartının alt ağ maskesi birbiriyle aynı olmalıdır. Alt ağ maskesi normal olarak 255.255.255.0 değerine sahiptir; ancak bilgisayarınız bir tesis LAN'ında olduğundan, özel alt ağlar kurmak için alt ağ maskesi çeşitli değerler alabilir (örneğin 255.255.254.0) <sup>2</sup> .
Ethernet-ile-USB adaptör kartı	İzole edilmiş bir ağa bağlıdır	CPU'nuzun ve programlama cihazınızın Ethernet - USB adaptör kartının ağ kimliği birbiriyle aynı olmalıdır <sup>1</sup> .	CPU'nuzun ve Ethernet - USB adaptör kartının alt ağ maskesi birbiriyle aynı olmalıdır. Alt ağ maskesi normal olarak <b>255.255.255.0</b> değerine sahiptir.

<sup>1</sup> Ağ ID, IP adresinin ilk kısmıdır (ilk üç kısım) (örneğin, **211.154.184.16**) ve hangi IP ağı üzerinde olduğunuzu belirler.

<sup>2</sup> Alt ağ maski bir matematiksel AND işlemi içindeki cihaz IP adresi ile birleştirildiğinde bir IP alt ağın sınırlarını belirler.

### Masaüstünüzde bulunan "Ağ Bağlantılarım" simgesini kullanarak programlama cihazınızın IP adreslerini atama ve kontrol etme

Programlama cihazınızın IP adreslerini atama ve kontrol etme işlemlerini izleyen menü seçimleri ile yapabilirsiniz:

- (Sağ-tıkla) "Ağ Bağlantılarım"
- "Özellikler"
- (Sağ-tıkla) "Lokal alan bağlantısı"
- "Özellikler"

"Local Area Connection Properties" diyalogunda, "This connection uses the following items:" alanında, aşağı doğru kayarak "Internet Protocol (TCP/IP)" a ininiz. "Internet Protocol (TCP/IP)" tıklayınız ve "Properties" butonunu tıklayınız. "Obtain an IP address automatically (DHCP)" veya "Use the following IP address" i seçiniz (bir statik IP adresi girmek için).

#### Not

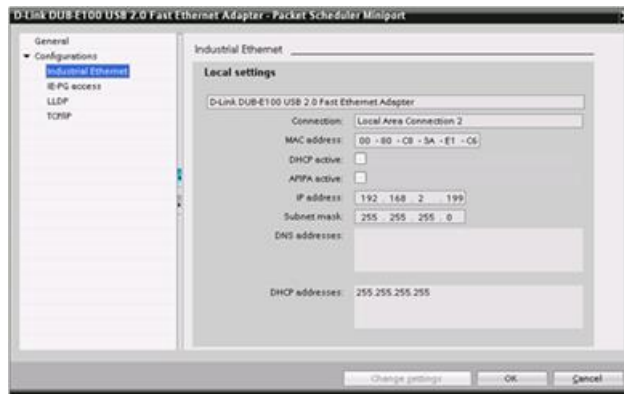
Dinamik Ana Bilgisayar Konfigürasyon Protokolü (DHCP), DHCP sunucusunun enerjilenmesinden sonra programlama cihazınıza otomatik olarak bir IP adresi atar.

#### 5.7.4.2 Programlama cihazınızın IP adresini kontrol etme

Programlama cihazının MAC ve IP adreslerini izleyen menü seçimlerini kullanarak kontrol edebilirsiniz:

4. "Proje ağacı" ndan "Online erişimi" seçiniz.
5. İstedığınız ağa sağ tıklayınız ve "Özellikleri" seçiniz.
6. Ağ diyalogundan "Konfigürasyonları" genişletiniz ve "Endüstriyel Ethernet" i seçiniz.

Programlama cihazının MAC ve IP adresleri görüntülenir.



### 5.7.4.3 CPU'ya çevrimiçi olarak IP Adresi Atama

Bir ağ cihazına çevrimiçi olarak bir IP adresi atayabilirsiniz. Bu fonksiyon özellikle cihazda ilk yapılan konfigürasyonda kullanışlıdır.

1. Proje ağacında, aşağıdaki menü seçimleri kullanılarak, CPU'ya hiçbir IP adresinin atanmamış olduğundan emin olunuz:

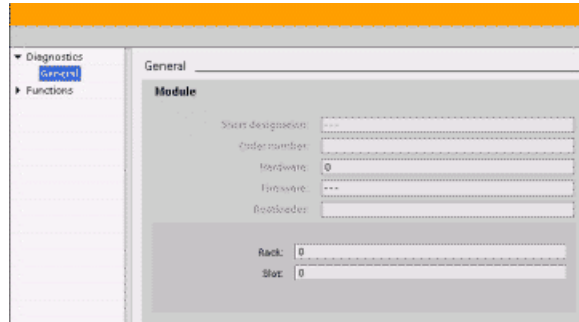
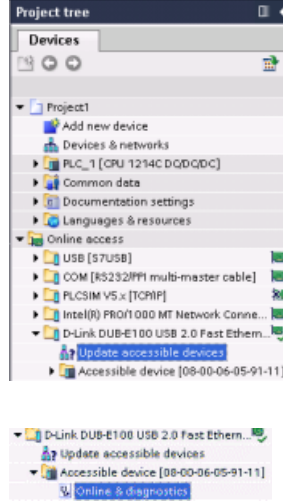
- "Online erişim"
- < Cihazın bulunduğu ağ için adaptör kartı >
- "Erişilebilir cihazları güncelle"

NOT: Bir IP adresi yerine bir MAC adresi gösterilirse, o zaman hiçbir IP adresi atanmamış demektir.

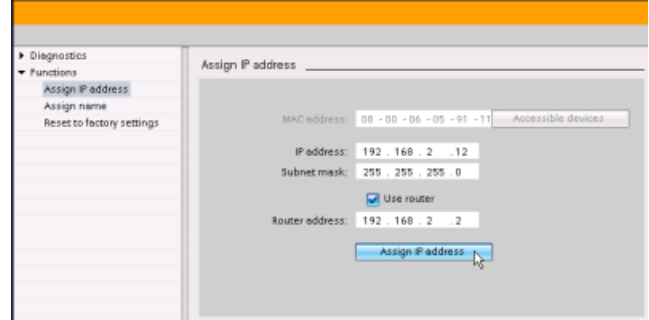
2. Gerekli erişilebilir cihazların altından "Online & Diagnostics"e çift tıklayınız.

3. "Online & diagnostics" diyalogundan şu menü seçimlerini yapınız:

- "Functions"
- "Assign IP address"

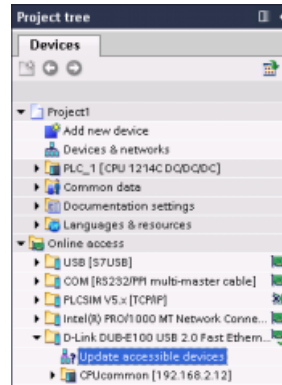


4. "IP adresleri" alanından yeni IP adresinizi girin ve "IP adresi ata" düğmesine tıklayınız.



5. Proje ağacından, CPU'ya yeni IP adresinin atandığından izleyen menü seçimleriyle emin olunuz:

- "Online erişim"
- < Cihazın bulunduğu ağ için adaptör >
- "Erişilebilir cihazları güncelle"



#### 5.7.4.4 Projenizdeki bir CPU için bir IP adresinin konfigürasyonu

##### PROFINET ara yüz konfigürasyonu

PROFINET ara yüzü parametrelerini konfigüre etmek için CPU üzerindeki yeşil PROFINET kutusunu seçiniz. Denetçi penceresindeki "Özellikler" sekmesi, PROFINET portunu görüntüler.



① PROFINET portu

## IP adreslerinin konfigürasyonu

**Ethernet (MAC) adresi:** Bir PROFINET ağında, her bir cihaz tanımlama amacıyla üretici tarafından bir Medya Erişim Kontrol adresine (MAC) atanmıştır. Bir MAC adresi, iki hexadecimal hanenin altılı grubundan oluşur ve iletim sırasına göre tirelerle (-) veya iki nokta üst üste işaretleriyle (:) ayrılır (örneğin 01-23-45-67-89-AB veya 01:23:45:67:89:AB).

**IP adresi:** Her bir cihaz, ayrıca bir İnternet Protokolü (IP) adresine sahip olmalıdır. Bu adresler cihazın daha karmaşık, yönlendirilmiş ağlara veri dağıtmasına izin verir.

Her IP adresi, dört adet 8 bitlik segmente bölünmüştür ve bunlar noktalı ondalık formatta ifade edilir (örneğin 211.154.184.16). IP adresinin ilk kısmı Ağ Kimliği için kullanılır (hangi ağ üzerinde olduğunuzu belirtir). İkinci kısım ise cihaz kimliği içindir (ağdaki her bir cihaz için özeldir). 192.168.x.y şeklindeki bir IP adresi, internete yönlendirilmemiş özel bir ağın parçası olarak tanınan standart bir atamadır.

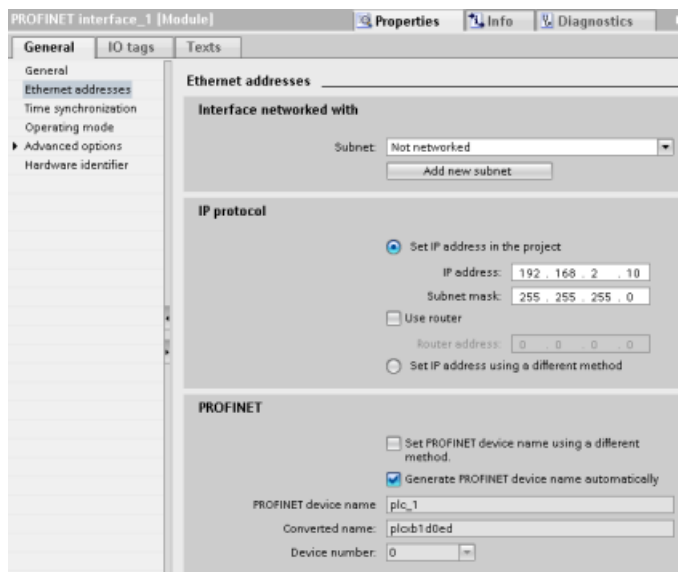
**Alt ağ maskesi:** Bir alt ağ, bağlı ağ cihazlarının mantıksal bir gruplamasıdır. Bir alt ağ üzerindeki düğümler (nodes), bir Yerel Alan Ağı (LAN) üzerinde birbirlerine fiziksel olarak yakın olarak konumlandırılma eğilimindedir. Bir maske (ağ maskesi ya da alt ağ maskesi olarak bilinir), bir IP alt ağının sınırlarını tanımlar.

255.255.255.0 şeklindeki bir alt ağ maskesi, genellikle küçük bir Yerel Ağ için uygundur, yani bu ağdaki tüm IP adreslerinin aynı ilk 3 sekizliğe sahip olması gerektiği ve de bu ağdaki çeşitli cihazların son sekizlik (8 bitlik alan) tarafından tanımlandığı anlamına gelir. Küçük bir Yerel Ağ üzerindeki cihazlara 192.168.2.0 ile 192.168.2.255 arasında bir IP adresi ile 255.255.255.0 alt ağ maskesini atamak buna bir örnek olarak verilebilir.

Farklı alt ağlar arasındaki tek bağlantı, bir yönlendirici vasıtasıyla olur. Alt ağlar kullanılırsa, bir IP yönlendiricisi kullanılmalıdır.

**IP yönlendirici (router):** Yönlendiriciler, LAN'lar arasındaki linklerdir. Bir yönlendirici kullanarak, bir ağ içindeki bir bilgisayar, arkasında başka LAN'lar da olabilecek diğer ağlara mesajlar gönderebilir. Verilerin gideceği yer LAN içinde değilse, yönlendirici verileri hedefe dağıtımın yapılabileceği başka bir ağa ya da ağ grubuna iletir.

Yönlendiricilerin veri paketlerini dağıtması ve alması IP adreslerine dayanır.



### IP adresi özellikleri:

Özellikler penceresinde, "Ethernet adresleri" konfigürasyon girişini seçiniz. STEP 7, yazılım projesini bu projeyi alacak CPU'nun IP adresi ile ilişkilendiren Ethernet adresinin konfigürasyon diyalogunu görüntüler




Çizelge 5- 9 IP adresi için parametreler

Parametre	Açıklama	
Alt ağ	Cihazın bağlı olduğu Alt ağın ismidir. Yeni bir alt ağ oluşturmak için "Yeni alt ağ ekle" butonuna tıklayınız. "Bağlı değil" varsayılan değerdir. İki bağlantı tipi mümkündür: <ul style="list-style-type: none"> <li>Varsayılan "Bağlı değil" lokal bir bağlantı sağlar.</li> <li>Ağınızda iki veya daha fazla cihaz varsa, bir alt ağ gereklidir.</li> </ul>	
IP protokolü	IP adresi	CPU için atanan IP adresi
	Alt ağ maskesi	Atanan alt ağ maskesi
	IP yönlendirici kullan	Bir IP yönlendiricinin kullanıldığını belirtmek için kontrol kutucuğunu işaretleyiniz.
	Yönlendirici adresi	Uygulanabilirse, yönlendirici için atanan IP adresi

### Not

Projeyi yüklediğinizde tüm IP adresleri konfigüre edilir. CPU önceden konfigüre edilmiş bir IP adresine sahip değilse, projeyi hedef cihazın MAC adresi ile ilişkilendirmelisiniz. CPU'nuz ağ üzerindeki bir yönlendiriciye bağlı ise, yönlendiricinin IP adresini de girmeniz gerekir.

"IP adresini farklı bir yöntem kullanarak ayarla" düğmesi, IP adresini çevrimiçi olarak ya da program yüklendikten sonra "T\_CONFIG (Sayfa 541)" komutunu kullanarak değiştirmenize izin verir. Bu IP adresi atama yöntemi sadece CPU için geçerlidir.

 <b>UYARI</b>
<p><b>"Set IP address using different method" ile bir donanım konfigürasyonu yükleme</b></p> <p>"Set IP address using a different method" opsiyonu etkin durumdayken bir donanım konfigürasyonu yüklemesinden sonra, CPU işletim modunun RUN'dan STOP'a veya STOP'tan RUN'a geçişi mümkün değildir.</p> <p>Bu koşullarda kullanıcı ekipmanı çalışmaya (run) devam eder ve beklenmedik makine çalışmalarına neden olabilir ve uygun önlemler alınmazsa bu durum personelin ölümü, ciddi yaralanması ve/veya mal hasarı ile sonuçlanabilir.</p> <p>CPU'nuzu gerçek bir otomasyon ortamında kullanmadan önce CPU IP adres(ler)inin ayarlandığından emin olunuz. Bu işlem STEP 7 programlama paketiniz (S7-1200 Tool veya T_CONFIG komutuyla birleşik HMI cihazı) kullanılarak yapılabilir.</p>


**UYARI**
**PROFINET ağının durabileceği koşullar**

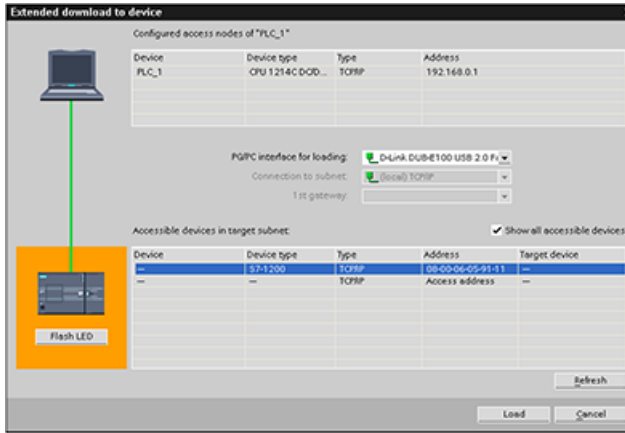
Bir CPU'nun IP adresini online veya kullanıcı programından değiştirirken, PROFINET ağının durabileceği bir durum oluşturmak da mümkündür.

Bir CPU'nun IP adresi alt ağın dışındaki bir IP adresiyle değiştirilirse, PROFINET ağı haberleşmesini kaybedecek ve tüm veri alışverişini duracaktır. Kullanıcı ekipmanı, bu koşullar altında çalışmaya devam edecek şekilde konfigüre edilebilir. PROFINET haberleşmesinin kaybedilmesi, beklenmedik makine veya proses işlemlerine neden olabilir. Bunlar da, gerekli önlemler alınmadığı takdirde ölüme, ciddi kişisel yaralanmalara veya malın hasar görmesine neden olabilir.

Bir IP adresi el ile değiştirilmek zorunda kalınırsa, yeni IP adresinin alt ağ içinde kaldığından emin olunuz.

## 5.7.5 PROFINET ağının test edilmesi

Konfigürasyon tamamlandıktan sonra projeyi CPU'ya yükleyiniz (Sayfa 196). Projeyi yüklediğinizde tüm IP adresleri konfigüre edilir.



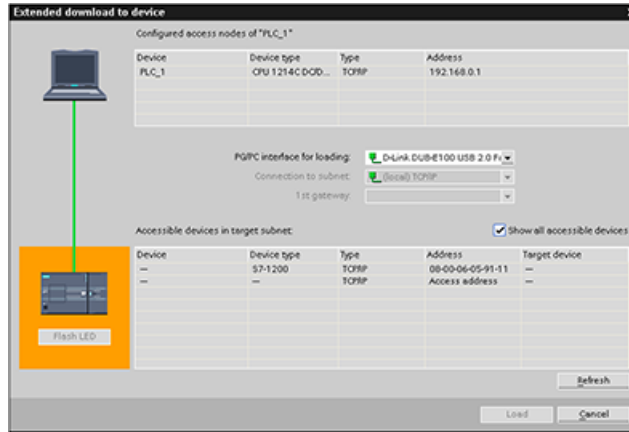
### Bir IP adresini cihaza çevrimiçi atama

S7-1200 CPU önceden konfigüre edilmiş IP adresine sahip değildir. CPU için el ile bir IP adresi atamalısınız:

- Bir cihaza online olarak bir IP adresi atama işleminin adım adım açıklaması için "Cihaz konfigürasyonu: Bir CPU'ya çevrimiçi olarak bir IP adresi atamak" (Sayfa 158) bölümüne bakınız.
- Projenize bir IP adresi atamak için Cihaz konfigürasyonunda IP adresini yapılandırılmalı, konfigürasyonu kaydetmeli ve bunu PLC'ye yüklemelisiniz. Daha fazla bilgi için "Cihaz yapılandırması: Projenizdeki bir CPU için bir IP adresi yapılandırmak" (Sayfa 159) bölümüne bakınız.

### Bağlı ağ cihazlarının test edilmesi için "Extended download to device" diyalogunun kullanımı

S7-1200 CPU "Cihaza yükle" fonksiyonu ve onun "Extended download to device" diyalogu, tüm erişilebilir ağ cihazlarını ve özel IP adresleri atanıp atanmamasından bağımsız olarak tüm cihazları gösterebilir. Tüm erişilebilir ve kullanılabilir cihazları, atanmış MAC veya IP adresleriyle görüntülemek için "Tüm erişilebilir cihazları göster" kontrol kutucuğunu işaretleyiniz.



Gerekli ağ cihazı bu listede yer almıyorsa, bu cihazla haberleşme bir nedenle kesintiye uğramıştır. Cihaz ve ağ, donanım ve/veya konfigürasyon hataları için incelenmelidir.

### 5.7.6 CPU'daki Ethernet (MAC) adresini konumlandırmak

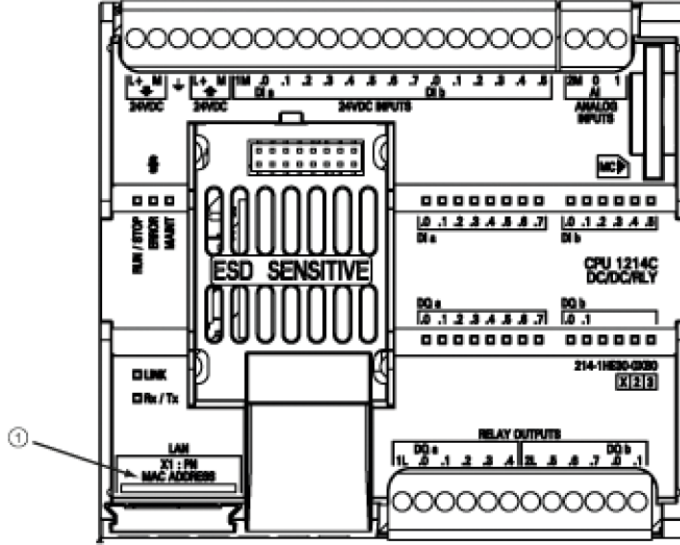
PROFINET ağında, bir Medya Erişim Kontrolü adresi (MAC adresi), üretici tarafından tanımlama amacıyla ağ ara yüzüne atanmış bir tanımlayıcıdır. Bir MAC adresi genellikle üreticinin kayıtlı tanımlama numarasını kodlar.

MAC adreslerinin kolay okunabilecek şekilde basılabilmesi için standart (IEEE802.3) formatı altılı gruplardan oluşur ve iletim sırasına göre tirelerle (-) veya iki nokta üst üste işaretleriyle (: ) ayrılır (örneğin 01-23-45-67-89-ab veya 01:23:45:67:89:ab)

#### Not

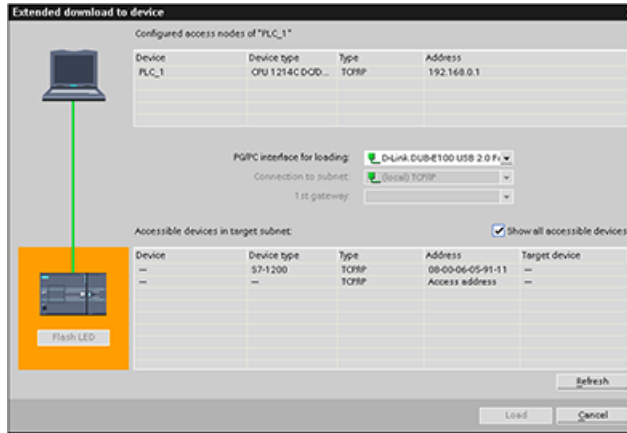
Her CPU'ya fabrikada özel ve kalıcı bir MAC adresi yüklenir. Bir CPU'nun MAC adresini değiştiremezsiniz.

MAC adresi CPU'nun ön tarafında, sol alt kısımda basılı olarak bulunur. MAC adresi bilgilerini görmek için alt kapakları kaldırmamız gerektiğine dikkat ediniz.



① MAC adresi

İlk olarak, CPU'nun bir IP adresi yoktur, sadece fabrikada atanmış bir MAC adresi vardır. PROFINET haberleşmeleri, tüm cihazlara özel bir IP adresi atanmasını gerektirir.



Erişilebilir tüm ağ cihazlarını göstermek ve tüm cihazlara özel IP adreslerinin atandığından emin olmak için CPU "Cihaza yükle" fonksiyonunu ve "Extended download to device" diyalogunu kullanınız.

Bu diyalog tüm erişilebilir ve hazır cihazları atanmış MAC veya IP adresleriyle beraber gösterir.

MAC adresleri, gerekli özel IP adreslerine sahip olmayan cihazların tanımlanması için çok önemlidir.

## 5.7.7 Ağ Zaman Protokolü senkronizasyonunun konfigüre edilmesi

**⚠ UYARI**

**Bir saldırgan ağınıza Network Time Protocol (NTP) senkronizasyon yoluyla ulaşabilir, saldırgan CPU sistem saatini kaydırarak prosesinizin sınırlı kontrolünü muhtemelen alabilir.**

S7-1200 CPU'nun NTP istemci özelliği varsayılan olarak etkisizdir ve etkinleştirildiğinde konfigüre edilen IP adresinin bir NTV sunucu olarak davranmasına izin verir. CPU bu özelliği varsayılan olarak etkisiz yapar ve uzaktan-kontrollü CPU sistem zaman düzeltmeleri için bu özelliği konfigüre etmelisiniz.

S7-1200 CPU "günün saati" interrupt'larını ve saat komutlarını destekler. Bunlar doğru CPU sistem zamanına bağlıdır. NTP'yi konfigüre eder ve bir sunucudan zaman senkronizasyonunu alırsanız, sunucunun güvenilir bir kaynak olduğundan emin olunuz. Aksi durumda güvenliğe uymamanız sonucunda bilinmeyen kullanıcı CPU sistem saatini kaydırarak prosesinizin sınırlı kontrolünü muhtemelen alabilir.

Güvenlik bilgisi ve öneriler için Siemens Servis ve Destek sitesindeki "Operational Guidelines for Industrial Security" ([http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational\\_guidelines\\_industrial\\_security\\_en.pdf](http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf)) adresimizi ziyaret ediniz.

Ağ Zaman Protokolü (NTP), bilgisayar sistemlerinin saatlerini internet zamanı sunucularıyla senkronize etmek için yaygın olarak kullanılmaktadır. NTP modunda, CP günün saati sorgularını düzenli aralıklarla (istemci modunda) alt ağdaki (LAN) NTP sunucusuna gönderir. Sunucudan gelen cevaplara göre, en güvenilir ve en doğru zaman hesaplanır ve istasyondaki günün saati senkronize edilir.

Bu modun avantajı, alt ağlar uçlarında zamanın senkronize edilebilmesidir.

Dörde kadar NTP sunucusunun IP adresleri konfigüre edilebilmektedir. Güncelleme aralıkları, zaman sorguları arasındaki aralığı (saniye) tanımlar. Aralığın değeri 10 saniye ile 1 gün arasında değişir.

NTP modunda, transfer edilen genellikle UTC'dir (Koordineli Evrensel Zaman) ve Greenwich Ortalama Zamanına (GMT) karşılık gelir.

Özellikler penceresinden, "Zaman senkronizasyonu" konfigürasyon girişini seçiniz. STEP 7, Zaman senkronizasyonu konfigürasyon diyalogunu görüntüler:

**Time synchronization**

Enable time-of-day synchronization using NTP mode

Server 1: 192 . 168 . 0 . 21

Server 2: 192 . 168 . 0 . 22

Server 3: 192 . 168 . 0 . 23

Server 4: 192 . 168 . 0 . 24

Update interval: 10 sec

**Not**

Projeyi yüklediğinizde bütün IP adresleri konfigüre edilir.

Çizelge 5- 10 Senkronizasyon süresi için parametreler

Parametre	Açıklama
Şebeke zaman protokol (NTP) sunucuları kullanarak günün saatini senkronizasyonunu etkinleştirmek	NTP sunucuları kullanarak günün saatini senkronizasyonunu etkinleştirmek için kontrol kutusuna tıklayınız.
Sunucu 1	Şebeke zaman sunucu 1 için atanan IP adresi
Sunucu 2	Şebeke zaman sunucu 2 için atanan IP adresi
Sunucu 3	Şebeke zaman sunucu 3 için atanan IP adresi
Sunucu 4	Şebeke zaman sunucu 4 için atanan IP adresi
Süre senkronizasyon aralığı	Aralık değeri (saniye)

### 5.7.8 PROFINET cihazı başlangıç süresi, isimlendirme ve adres ataması

PROFINET IO'ları sisteminizin başlangıç süresini uzatabilir (yapılandırılabilir zaman aşımı). Cihaz sayısının fazlalığı ve yavaş cihazlar RUN moduna geçiş için gereken sürenin miktarını etkiler. S7-1200 PROFINET ağında maksimum 8 IO cihazı olabilir.

Her istasyon (veya IO cihazı), başlangıçta bağımsız olarak başlar ve bu toplam CPU başlatma süresini etkiler. Yapılandırılabilir zaman aşımını çok düşük bir değere ayarlarsanız, tüm istasyonların başlangıcı tamamlaması için yeterli bir toplam CPU başlangıç süresi olmayabilir. Böyle bir durum oluşursa, yanlış istasyon hataları ortaya çıkacaktır.

"Startup" altında CPU özelliklerinde, "Parameter assignment time for distributed I/O" (time-out) bulabilirsiniz. Varsayılan yapılandırılabilir zaman aşımı 60,000 ms (1 dakika) olup kullanıcı bu süreyi değiştirebilir.

### STEP 7'de PROFINET cihazı isimlendirme ve adresleme

Tüm PROFINET cihazlarının bir Cihaz Adı ve bir IP Adresi olmalıdır. Cihaz Adlarını tanımlamak ve IP adreslerini yapılandırmak için STEP 7'yi kullanınız. Cihaz adları, IO cihazlarına PROFINET DCP (Keşif ve Yapılandırma Protokolü) kullanılarak yüklenir.

**Sistem başlangıcında PROFINET adres ataması**

Kontrolör ağ üzerinden cihazların adlarını yayınlar ve cihazlar MAC adresleriyle yanıt verir. Daha sonra kontrolör PROFINET DCP protokolünü kullanarak cihaza bir IP adresi atar:

- MAC adresinin yapılandırılmış bir IP adresi varsa, istasyon başlatmayı gerçekleştirir.
- MAC adresinin yapılandırılmış bir IP adresi yoksa, STEP 7 projede yapılandırılmış adresi atar ve istasyon başlatmayı bundan sonra gerçekleştirir.
- Bu proseste bir sorun varsa bir istasyon hatası meydana gelir ve başlangıç gerçekleşmez. Bu durum, yapılandırılabilir zaman aşımı değerinin aşılmasına neden olur.

*S7Cihaz Konfigürasyonu*

*5.7 Haberleşme için CPU'yu konfigüre etmek*

---



## Programlama konseptleri

### 6.1 PLC sisteminin tasarımı için kılavuzlar

Bir PLC sistemi tasarlarken, çeşitli yöntemler ve kriterler arasından seçim yapabilirsiniz. Aşağıdaki genel kılavuzlar çoğu tasarım projesi için uygulanabilir. Bu aşamada elbette ki kendi şirketinizin direktiflerine (prosedürlerine), eğitiminize ve bulunduğunuz ülkenin kabul görmüş uygulamalarına da uymanız gerekir.

Çizelge 6- 1 Bir PLC sistemi tasarımı için kılavuzlar

Tavsiye edilen adımlar	Görevler
Prosesinizin veya makinenizin bölünmesi	Prosesinizi veya makinenizi, birbirlerine göre bir bağımsızlık seviyelerine sahip olan bölümlere ayırınız. Bu bölümler kontrolörler arasındaki sınırları belirler ve fonksiyonel açıklama özelliklerini ve kaynakların atamasını etkiler.
Fonksiyonel spesifikasyonların oluşturulması	I/O noktaları, çalışmanın fonksiyonel açıklaması, her bir aktuatör için eyleme izin verilmeden önce ulaşılması gereken durum (bir selenoid, bir motor ya da bir sürücü), operatör ara yüzü ve prosesin ya da makinenin diğer bölümleri ile birlikte bütün ara yüzler için bir açıklama yazınız.
Emniyet devrelerinin tasarlanması	Emniyet için donanım mantığını gerektirebilen her bir cihazı belirleyiniz. Kontrol cihazlarının emniyetsiz bir durumda arızalanabileceğini unutmayınız. Bu tip durumlarda makinelerde beklenmeyen başlatmalar ya da makinenin çalışmasında değişiklikler yaşanabilir. Makinelerde ortaya çıkan beklenmeyen durumlar ya da hatalı işleyişler insanlara zarar verebilir ya da ciddi maddi zararlar yaşanabilir. Emniyetsiz işleyişleri önlemek için elektromanyetik bastırma devreleri (PLC'den bağımsız olarak çalışan) tatbik etmeyi düşünebilirsiniz. Emniyet devrelerinin tasarımında aşağıdaki görevler yer almalıdır: <ul style="list-style-type: none"> <li>Tehlikeli olabilecek her türlü hatalı veya beklenmedik aktuatör işleyişini tanımlayınız.</li> <li>Tehlikeli olmayan işleyiş sağlayacak şartları tanımlayın ve bu şartların PLC'den bağımsız olarak nasıl tespit edileceğini belirleyiniz.</li> <li>Enerji uygulandığında ve kaldırıldığında PLC'nin prosesi nasıl etkilediğini tanımlayın. Ayrıca hataların nasıl ve ne zaman tespit edildiğini de tanımlayınız. Bu bilgileri sadece normal ve beklenen anormal işleyişler tasarlamak için kullanınız. Emniyet amacıyla bu "en iyi durum" senaryosuna bel bağlamamanız gerekir.</li> <li>PLC'den bağımsız olarak tehlikeli işleyiş engelleyen manuel ya da elektromanyetik bastırmaları tasarlayınız.</li> <li>Programın ve operatör ara yüzlerinin gerekli bilgilere sahip olabilmeleri için bağımsız devrelerden PLC'ye uygun durum bilgilerini sağlayınız.</li> <li>Prosesin emniyetli işleyiş için gerekli olan diğer emniyetle ilgili gereklilikleri tanımlayın.</li> </ul>
Sistem güvenliğinin planlanması	Prosesinize erişim için ihtiyaç duyduğunuz koruma seviyesini (Sayfa 190) belirleyin. CPU'ları ve program bloklarını yetkisiz erişimlerden şifre (password) ile koruyabilirsiniz.

Tavsiye edilen adımlar	Görevler
Operatör istasyonlarının belirlenmesi	<p>Fonksiyonel spesifikasyonların gerekliliklerini temel alarak operatör istasyonlarının şu çizimlerini oluşturunuz:</p> <ul style="list-style-type: none"> <li>Her bir operatör istasyonunun proses ya da makineye göre konumunu gösteren bir genel bakış çizimi.</li> <li>Operatör istasyonunun cihazlarının görünüm, anahtarlar ve ışıklar gibi mekanik yerleşim çizimi; görünüm, anahtarlar ve ışıklar gibi.</li> <li>PLC ve sinyal modüllerinin ilgili I/O'larını da kapsayan elektriksel çizimler.</li> </ul>
Konfigürasyon çizimlerinin oluşturulması	<p>Fonksiyonel spesifikasyonların gerekliliklerini temel alarak, kontrol ekipmanlarının şu konfigürasyon çizimlerini oluşturunuz:</p> <ul style="list-style-type: none"> <li>Her bir PLC'nin proses ya da makineye göre konumunu gösteren bir genel bakış çizimi.</li> <li>Her bir PLC'nin ve I/O modülünün, tüm kabinler ve diğer ekipmanlar da dahil olmak üzere mekanik yerleşim çizimi.</li> <li>Her bir PLC ve tüm I/O modülleri için cihaz model numaralarını, haberleşme adreslerini ve I/O adreslerini içeren elektriksel çizimler.</li> </ul>
Sembolik isimlerin bir listesini oluşturmak	<p>Mutlak adresler için sembolik isimlerin bir listesini oluşturun. Sadece fiziksel I/O sinyallerini değil, aynı zamanda programınızda kullanılacak diğer elemanları da (tag(etiket) isimleri gibi) dahil ediniz.</p>

## 6.2 Kullanıcı programınızın yapılandırılması

Otomasyon görevleri için bir kullanıcı programı oluşturduğunuzda, program için komutları kod bloklarının içine ekleyiniz:

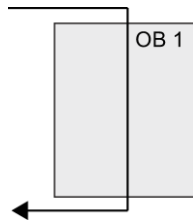
- Bir organizasyon bloğu (OB), CPU'daki belirli bir olaya tepki verir ve kullanıcı programının yürütülmesini kesintiye uğratabilir. Kullanıcı programının (OB 1) döngüsel yürütmesi için varsayılan, kullanıcı programınız için temel yapıyı sunar. Eğer programınıza başka OB'ler eklerseniz, bu OB'ler OB 1'in yürütülmesini kesintiye uğratır. Bu diğer OB'ler, hataların ve kesintilerin işlenmesi için ya da belirli zaman aralıklarında belirli program kodlarının yürütülmesi için başlatma görevleri gibi belirli görevleri uygular.
- Bir fonksiyon bloğu (FB), başka bir kod bloğundan (OB, FB veya FC) çağrıldığında yürütülen bir alt rutindir. Çağırın blok, parametreleri FB'ye aktarır ve ayrıca bu FB'nin belirli çağırısı ya da olayı için verileri saklayan özel bir data bloğu (DB) tanımlar. Instance DB'sinin değiştirilmesi, genel bir FB'nin bir cihaz grubunun işletimini kontrol etmesine olanak tanır. Örneğin bir FB, farklı instance DB'lerine sahip olan ve her pompa ya da vana için özel işletimsel parametreleri içeren birkaç vanayı veya pompayı kontrol edebilir.
- Bir fonksiyon (FC), başka bir kod bloğundan (OB, FB veya FC) çağrıldığında yürütülen bir alt rutindir. FC'nin ilişkilendirilmiş bir instance DB'si bulunmaz. Çağırın blok, parametreleri FC'ye aktarır. FC'den gelen çıkış verileri, bir bellek adresine ya da bir global DB'ye yazılmalıdır.

### Kullanıcı programınız için yapı tipinin seçilmesi

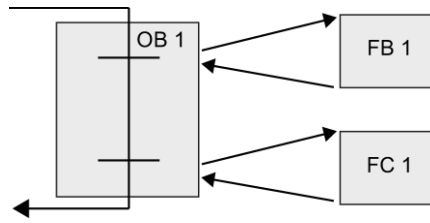
Kullanıcı programınızın oluşturulması için, uygulamanızın gereklilikleri esas alınarak bir doğrusal yapı ya da modüler yapı seçebilirsiniz:

- Bir doğrusal program, otomasyon görevleriniz için tüm komutları birbiri ardına sıralı olarak yürütür. Doğrusal program, tüm program komutlarını programın döngüsel yürütmesi için (OB 1) OB'ye koyar.
- Bir modüler program, özel görevleri gerçekleştiren özel kod bloklarını çağırır. Modüler bir yapı oluşturmak için karmaşık otomasyon görevini, prosesin teknolojik fonksiyonlarına karşılık gelen daha küçük alt görevlere bölebilirsiniz. Her bir kod bloğu, her bir alt görev için program segmenti sunar. Programınızı, başka bir bloktan kod bloklarından birini çağırarak yapılandırabilirsiniz.

Doğrusal yapı:



Modüler yapı:



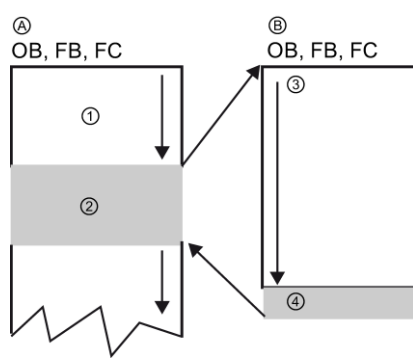
Kullanıcı programı içinde tekrar kullanılabilir genel kod blokları oluşturarak kullanıcı programının tasarımını ve uygulamasını basitleştirebilirsiniz. Genel kodların kullanılmasında şu faydalar vardır:

- Bir pompanın ya da motorun kontrol edilmesi gibi standart görevler için tekrar kullanılabilir bloklar oluşturabilirsiniz. Bu genel kod bloklarını, farklı uygulamalar ya da çözümler tarafından kullanılabilir bir kütüphanede saklayabilirsiniz.
- Kullanıcı programını fonksiyonel görevlerle ilgili modüler bileşenler şeklinde yapılandırdığınız zaman, programınızın tasarımını hem anlaşılma hem de yönetim açısından daha kolay olabilir. Modüler bileşenler sadece program tasarımının standartlaştırılmasına yardımcı olmakla kalmaz, aynı zamanda program kodunun güncellenmesinin ve değiştirilmesinin daha hızlı ve kolay olmasını sağlar.
- Modüler bileşenlerin oluşturulması, programınızdaki hata ayıklama işlemlerini de basitleştirir. Tüm programı bir program segmentleri seti olarak yapılandırdığınızda, her bir kod bloğunun işlevselliğini geliştirildiğinde test edebilirsiniz.
- Özel teknolojik fonksiyonlarla ilgili modüler bileşenler oluşturmak, tamamlanan uygulamanın devreye alınma süresinin kısaltılmasına ve sürecin basitleştirilmesine yardımcı olur.

## 6.3 Programınızı yapılandırmak için blokları kullanmak

Genel görevleri uygulaması için FC'ler ve FB'ler tasarlayarak modüler kod blokları oluşturmuş olursunuz. Sonra programınızı, bu tekrar kullanılabilir modülleri çağıran başka kod blokları ekleyerek yapılandırabilirsiniz. Çağırılan blok, cihaza özel parametreleri çağırılan bloğa aktarır.

Bir kod bloğu başka bir kod bloğunu çağırdığında, CPU çağırılan bloktaki program kodunu çalıştırır. Çağırılan bloktaki yürütme tamamlandıktan sonra CPU çağırılan bloğun yürütmesine devam eder. İşlem, blok çağırmasını takip eden komutun yürütülmesiyle devam eder.



A Çağırılan blok

B Çağırılan (ya da kesintiye uğratılan) blok

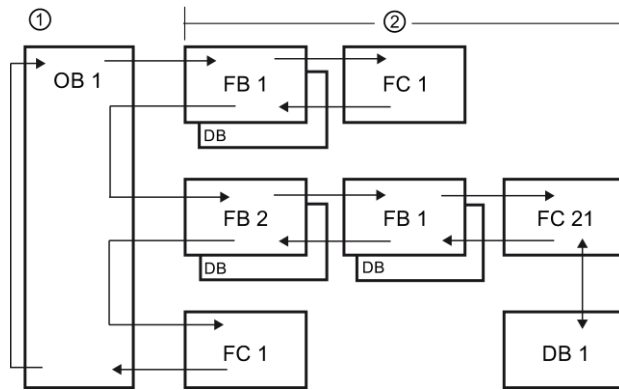
① Program yürütmesi

② Başka bir bloğun yürütmesini başlatan komut ya da olay

③ Program yürütmesi

④ Blok sonu (çağırılan bloğa döner)

Blok çağrılarını, birden fazla modüler yapı için iç içe yerleştirebilirsiniz (yuvalama). Bu örnekte yuvalama derinliği 4'tür: program döngüsü OB'si artı 3 kod bloğu katmanı.



① Döngünün başlangıcı

② Yuvalama derinliği

### 6.3.1 Organizasyon bloğu (OB)

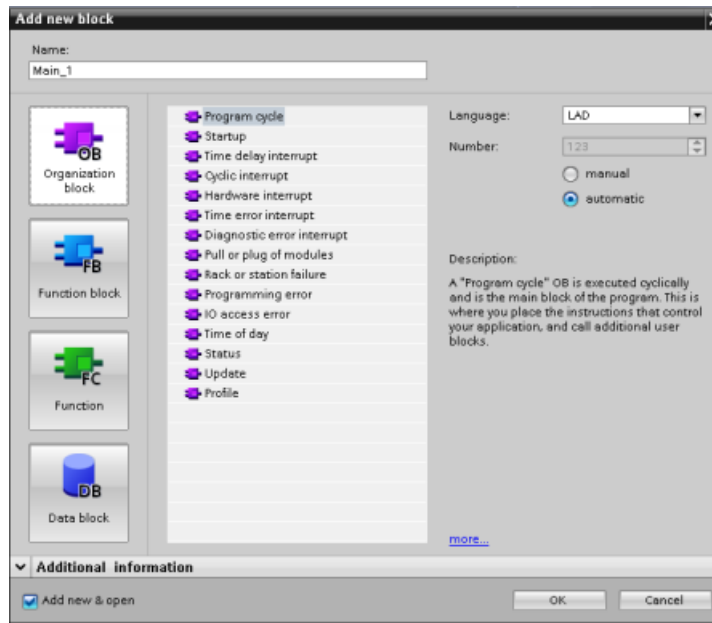
Organizasyon blokları, programınız için yapı sağlar. Bunlar, işletim sistemi ile kullanıcı programı arasındaki ara yüz olarak hizmet verirler. OB'ler olaya dayalıdır. Bir diyagnostik interrupt'ı ya da bir zaman aralığı gibi bir olay, CPU'nun bir OB'yi yürütmesine neden olacaktır. Bazı OB'lerin önceden tanımlanmış başlangıç olayları ve davranışları vardır.

Program döngüsü OB'si, ana programınızı içerir. Kullanıcı programınızda birden fazla program OB'si bulunabilir. RUN modunda iken, program döngüsü OB'leri en düşük öncelik seviyesinde yürütülür ve bunlar tüm diğer program işleme türleri tarafından kesintiye uğratılabilir. Başlangıç OB'leri program döngüsü OB'lerini kesintiye uğratmaz çünkü CPU başlangıç OB'lerini RUN moduna girmeden önce yürütür.

Program döngüsü OB'lerinin işlenmesi bittikten sonra CPU derhal program döngüsü OB'lerini tekrar çalıştırır. Bu döngüsel işleme, programlanabilir lojik kontrolörleri için kullanılan "normal" işleme türüdür. Çoğu uygulama için tüm kullanıcı programı tek bir program döngüsü OB'sinin içinde yer alır.

Hataları ve kesintileri işlemek ya da belirli zaman aralıklarında özel bir program kodunu çalıştırmak gibi özel fonksiyonları gerçekleştirmek için başka OB'ler de oluşturabilirsiniz. Bu OB'ler, program döngüsü OB'lerinin yürütülmesini kesintiye uğratır.

Kullanıcı programınızda yeni OB'ler oluşturmak için "Add new block" diyalogunu kullanın.



Kesme işlemesi her zaman olaya dayalıdır. Böyle bir olay meydana geldiğinde, CPU kullanıcı programının yürütülmesini keser ve bu olayı işlemek için yapılandırılmış OB'yi çağırır. Kesme OB'sinin yürütülmesi bittikten sonra CPU kullanıcı programının yürütülmesine kesme noktasından itibaren devam eder.

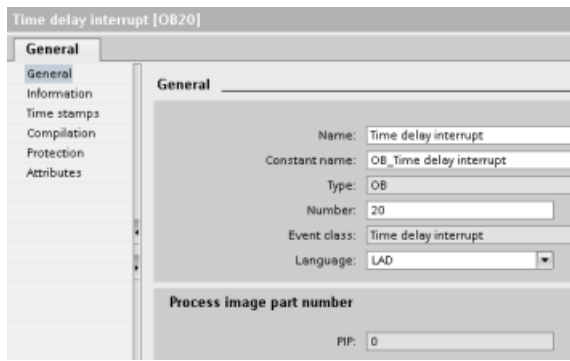
CPU, kesme olaylarının işlenme sırasını bir önceliğe göre belirler. Çoklu kesme olaylarını, aynı öncelik sınıfına atayabilirsiniz. Daha fazla bilgi almak için organizasyon blokları (Sayfa 85) ile kullanıcı programının yürütülmesi konusundaki başlıklara bakınız (Sayfa 77).

### İlave OB'lerin oluşturulması

Kullanıcı programınız için program döngüsü ve başlangıç OB'si olayları için bile birden fazla OB oluşturmanız mümkündür. Bir OB oluşturmak için "Add new block" diyalogunu kullanınız.

Eğer kullanıcı programınız için birden fazla program döngüsü OB'si oluşturursanız, CPU her bir program döngüsü OB'sini en düşük numaradan (OB1 gibi) başlamak üzere numara sırasına göre yürütür. Örneğin: ilk program döngüsü OB'si (OB1 gibi) bittikten sonra, CPU bir sonraki en büyük numaralı program döngüsü OB'sine geçer.

### Bir OB'nin özelliklerinin yapılandırılması



Bir OB özelliklerini yapılandırabilirsiniz. Örneğin OB numarasını veya programlama dilini yapılandırmanız mümkündür.

#### Not

PIP0, PIP1, PIP2, PIP3 veya PIP4'e karşılık gelen bir OB'ye proses görüntü bölümü numarasını atayabileceğinizi dikkate alınız. Eğer proses görüntü bölümü numarasına bir sayı girerseniz, CPU, proses görüntü bölümlerini oluşturur. Proses görüntü bölümlerinin açıklaması için "kullanıcı programının yürütülmesi" başlığına (Sayfa 77) bakınız.

### 6.3.2 Fonksiyon (FC)

Bir fonksiyon (FC), bir grup giriş değeri üzerinde belirli bir işlemi gerçekleştiren bir kod bloğudur. FC bu işlemin sonuçlarını bellek konumlarında saklar. Örneğin standart veya tekrar kullanılabilir işlemleri (matematiksel hesaplamalar gibi) ya da teknolojik fonksiyonları (bit lojik işlemlerini kullanarak bağımsız kontroller için olanlar gibi) gerçekleştirmek için FC'leri kullanınız. Bir FC, bir programın farklı noktalarında birkaç kez çağrılabilir. Bu tekrar kullanım sıkça tekrarlayan görevlerin programlamasını basitleştirir.

Bir FC'nin ilişkilendirilmiş bir instance veri bloğu (DB) yoktur. FC, işlemin hesaplanması için kullanılan geçici veriler için yerel veri alanını kullanır. Geçici veriler kaydedilmez. Verileri kalıcı olarak kaydetmek için çıkış değerini M hafızası gibi bir global bellek konumuna ya da bir global DB'ye atayınız.

### 6.3.3 Fonksiyon bloğu (FB)

Bir fonksiyon bloğu (FB), parametreleri ve statik verileri için bir instance veri bloğu kullanan bir kod bloğudur. FB'ler, bir veri bloğunda (DB) ya da bir "instance" veri bloğunda bulunan değişken bir belleğe sahiptir. Instance DB'si, FB'in bu instance (veya çağırısı) ile ilişkilendirilmiş olan ve de FB sona erdikten sonra verileri kaydeden bir bellek bloğu sunar. Farklı instance DB'lerini farklı FB sınıfları ile ilişkilendirebilirsiniz. Instance DB'leri, birden fazla cihazın kontrolü için bir genel FB kullanmanıza olanak tanır. Programınızı, bir FB'ye ve bir instance DB'sine çağrı gerçekleştirmek için bir kod bloğu ile yapılandırabilirsiniz. CPU bundan sonra bu FB'deki program kodunu yürütür ve instance DB'sindeki blok parametreleri ile statik lokal verileri kaydeder. FB'nin yürütmesi sona erdikten sonra CPU FB'yi çağırın kod bloğuna döner. Instance DB'si, FB'nin bu instance'ı için değerleri tutar. Bu değerler, aynı tarama döngüsünde ya da başka tarama döngülerinde fonksiyon bloğuna yapılan müteakip çağrılar için kullanılabilir.

#### İlişkilendirilmiş bellekli tekrar kullanılabilir kod blokları

İşlemlerini bir tarama döngüsü içinde bitirmeyen görevlerin ya da cihazların işletimini kontrol etmek için tipik olarak bir FB kullanılır. Bir taramadan sonraki taramaya hızlı bir şekilde erişilebilmek için işletim parametrelerini saklamak amacıyla kullanıcı programınızdaki her bir FB'nin bir veya birden fazla instance DB'si vardır. Bir FB'yi çağırdığınızda, FB'nin bu çağırısı için blok parametrelerini ve statik lokal verilerini tutan bir instance DB'sini de belirlersiniz. Instance DB'si bu değerleri FB yürütmesi sona erdikten sonra tutar.

Genel kontrol görevleri için FB tasarlamak suretiyle FB'yi birden fazla cihaz için tekrar tekrar kullanabilirsiniz. Bunun için FB'nin farklı çağrıları için farklı instance DB'leri seçilir.

Bir FB'nin Input, Output ve InOut ve Static parametreleri instance DB'sinde tutulur.

Ayrıca, RUN modundaki fonksiyon bloğu ara yüzünü modifiye edebilir ve indirebilirsiniz.

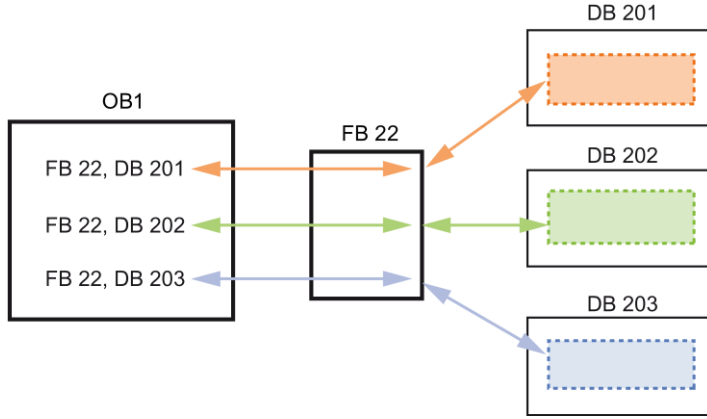
#### Instance DB'sindeki başlangıç değerini atamak

Instance DB'si, her bir parametre için hem bir varsayılan değeri hem de bir başlangıç değerini içerir. Başlangıç değeri, FB çalıştırıldığında kullanılacak değeri sağlar. Başlangıç değeri bundan sonra kullanıcı programınızın yürütülmesi sırasında değiştirilebilir.

FB ara yüzü aynı zamanda bir "varsayılan değer" sütunu sunar. Bu sütun, program kodunu yazarken parametre için yeni bir başlangıç değeri atamanıza izin verir. FB'deki bu varsayılan değer daha sonra ilişkilendirilmiş instance DB'sindeki başlangıç değerine transfer edilir. Eğer FB ara yüzündeki bir parametre için bir başlangıç değeri atamazsanız, instance DB'sindeki varsayılan değer başlangıç değeri olarak kopyalanır.

### DB'lerle tek bir FB kullanmak

Aşağıdaki şekil, her bir çağrı için farklı bir veri bloğu kullanarak bir FB'yi üç kez çağırarak bir OB'yi göstermektedir. Bu yapı, farklı cihazlar için her bir çağrı için farklı bir instance veri bloğu atamak suretiyle bir genel FB'nin, motorlar gibi birkaç benzer cihazı kontrol etmesine izin verir. Her bir instance DB'si, bağımsız bir cihaz için verileri saklar (hız, parça süresi ve toplam işletim süresi gibi).



Bu örnekte, FB 22 üç ayrı cihazı kontrol eder; DB 201 ilk cihaz için işletimsel verileri saklar, DB 202 ikinci cihaz için işletimsel verileri saklar ve DB 203 üçüncü cihaz için işletimsel verileri saklar.

### 6.3.4 Veri bloğu (DB)

Kod blokları için veri kaydetmek amacıyla kullanıcı programınızda veri blokları (DB) oluşturun. Kullanıcı programındaki tüm program blokları bir global DB'deki verilere erişebilir ancak instance DB'si özel bir fonksiyon bloğuna (FB) ait bilgileri saklar.

Bir DB'de depolanan veriler, ilgili kod bloğunun yürütmesi sonlana kadar silinmez. İki tür DB vardır:

- Bir global DB programınızdaki kod blokları için olan verileri saklar. Herhangi bir OB, FB ya da FC global DB'deki verilere erişebilir.
- Bir instance DB'si, belirli bir FB için olan verileri saklar. Instance DB'sindeki verilerin yapısı parametreleri (Giriş, Çıkış ve GirişÇıkış) ve FB için olan statik verileri yansıtır. (FB için Temp belleği, instance DB'sinde saklanmaz.)

#### Not

Instance DB'si belirli bir FB için olan verileri yansıtır olsa da herhangi bir kod bloğu bir instance DB'sindeki verilere erişebilir.

Ayrıca, RUN modundaki fonksiyon bloğu ara yüzünü modifiye edebilir ve indirebilirsiniz. (Sayfa 808).



### Salt-okunur veri blokları

Bir DB'yi salt okunur olarak yapılandırabilirsiniz:

1. Proje navigasyonunda DB'nin üzerine sağ tıklayınız ve açılan içerik menüsünden Properties seçiniz.
2. "Properties" (Özellikler) diyalogundan "Attributes" (Nitelikleri) seçiniz.
3. Cihazdaki "Data block write-protected in the device" seçeneğini seçiniz. ve "OK" tıklayınız.

### Optimize edilmiş ve standart veri blokları

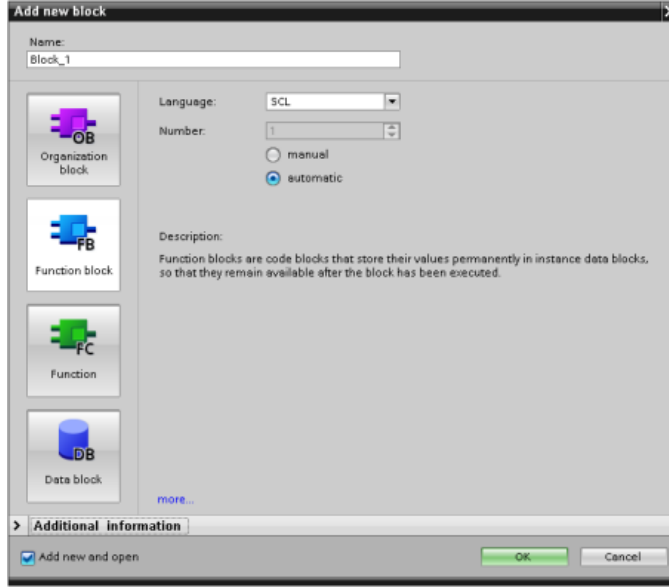
Bir veri bloğunu standart veya optimize edilmiş olarak ta yapılandırabilirsiniz. Standart bir DB, STEP 7 klasik programlama araçları ve klasik S7-300 ve S7-400 CPU'ları ile uyumludur. Optimize edilmiş erişimi bulunan veri blokları, sabit tanımlanmış bir yapıya sahip değildir. Veri elemanları, beyanın içinde sadece sembolik bir isim içerir ve blok içerisinde sabit bir adres bulunmaz. CPU, bu elemanları, bloğun kullanılabilir bir bellek alanında bellekte herhangi bir boşluk olmayacak şekilde saklar. Bu ise bellek kapasitesinin optimal bir şekilde kullanılmasını sağlar.

Veri bloğu için optimize edilmiş erişim kurmak için aşağıdaki adımları takip ediniz:

1. STEP 7 proje ağacında program blokları klasörünü genişletiniz.
2. Veri bloğu üzerine sağ tıklayınız ve açılan içerik menüsünden Özellikleri seçiniz.
3. "Attributes" için "Optimized block access" seçiniz.

Optimize edilmiş blok erişiminin yeni veri blokları için varsayılan değer olduğunu dikkate alınız. "Optimized block access" seçimini kaldırdığınızda, blok, standart erişimleri kullanır.

### 6.3.5 Tekrar kullanılabilir kod blokları oluşturmak



OB'ler, FB'ler, FC'ler ve global DB'ler oluşturmak için Proje navigatöründeki "Program blocks" " (Program blokları) altındaki "Add new block" (Yeni blok ekle) diyalogunu kullanınız. Bir kod bloğu oluşturduğunuzda, blok için programlama dilini de seçiniz. Sadece verileri sakladığından bir DB için bir dil seçmeyiniz. "Add new and open" (Yeni ekle ve aç) onay kutusunun seçilmesi, Project view'deki (proje görünümü) kod bloğunu açar.

Tekrar kullanmak istediğiniz nesnelere kütüphanede saklayabilirsiniz. Her bir proje için projeye bağlı bir proje kütüphanesi bulunur. Birkaç proje boyunca kullanılacak çok sayıda genel kütüphaneler oluşturabilirsiniz. Kütüphaneler birbiriyle uyumlu olduğundan kütüphane elemanları, bir kütüphaneden diğer bir kütüphaneye kopyalanabilir ve taşınabilir.

Kütüphaneler örneğin, öncelikle proje kütüphanesine yapıştırılacak ve daha sonra burada daha fazla geliştirilecek bloklar için şablonları oluşturmak üzere kullanılır. Son olarak, blokları proje kütüphanesinden genel kütüphaneye kopyalarsınız. Genel kütüphaneyi, projenizde çalışan diğer meslektaşlarınız için kullanılabilir yapabilirsiniz. Meslektaşlarınız, bu blokları kullanır ve gerekli olduğunda bunları kendi münferit gereksinimlerine uyumlu hale getirirler.

Kütüphane çalışmaları hakkındaki ayrıntılar için, STEP 7 çevrimiçi yardım kütüphane konularına bakınız.

## 6.4 Veri tutarlılığının (uyumluluğunun) anlaşılması

CPU, tüm temel veri tipleri ve tanımlı sistem yapıları (örneğin IEC\_TIMERS veya DTL) için veri bütünlüğünü korur (Word ve DWord). Değerin okunması veya yazılması kesintiye uğratılamaz. (Örneğin CPU bir DWord değerine erişimi DWord'ün dört baytı da okunana ya da yazılana kadar korur). Program döngüsü OB'lerinin ve interrupt OB'lerinin aynı bellek konumuna aynı anda yazmamasını sağlamak için CPU, program döngüsü OB'sindeki okuma veya yazma işlemi tamamlanana kadar bir interrupt yürütmez.

Kullanıcı programınız bir program döngüsü OB'si ve bir interrupt OB'si arasındaki çeşitli değerleri paylaşıyorsa, kullanıcı programınız aynı zamanda bu değerlerin de tutarlı bir şekilde değiştirilmesini veya okunmasını sağlamalıdır. Paylaşılan değerlere erişimi engellemek için program döngüsü OB'nizdeki DIS\_AIRT (alarm interrupt'ını devre dışı bırak) ve EN\_AIRT (alarm interrupt'ını etkinleştir) komutlarını da kullanabilirsiniz.

- Bir interrupt OB'sinin okuma veya yazma işlemi sırasında yürütülmemesini sağlamak için kod bloğuna bir DIS\_AIRT komutu ekleyiniz.
- Bir interrupt OB'si tarafından değiştirilebilecek değerleri okuyan veya yazan komutları ekleyiniz.
- DIS\_AIRT komutunu iptal etmek ve interrupt OB'sinin yürütülmesine izin vermek için dizinin sonuna bir EN\_AIRT komutu ekleyiniz.

Bir HMI cihazından ya da başka bir CPU'dan gelen haberleşme talebi aynı zamanda program döngüsü OB'sinin yürütülmesini kesintiye uğratabilir. Haberleşme talepleri ayrıca veri tutarlılığıyla ilgili sorunlara da neden olabilir. CPU, temel veri türlerinin kullanıcı programı komutları tarafından her zaman tutarlı bir şekilde okunup yazılmasını sağlar. Kullanıcı programı haberleşmeler tarafından periyodik olarak kesintiye uğratıldığından, CPU'daki çoklu değerlerin HMI ile aynı anda güncellenmesini garanti etmesi mümkün değildir. Örneğin bir HMI ekranında görüntülenen değerler, CPU'nun farklı tarama döngülerinden olabilir.

PtP (Noktadan noktaya) komutları, PROFINET komutları (TSEND\_C ve TRCV\_C gibi), PROFINET Dağıtılmış I/O komutları, PROFIBUS Dağıtılmış I/O komutları (Sayfa 314), kesintiye uğratılabilecek veri tamponlarını transfer eder. Hem program döngüsü OB'sindeki hem de bir interrupt OB'sindeki tamponlarda herhangi bir yazma veya okuma işleminden kaçınarak veri tamponları için veri tutarlılığını sağlayınız. Bir interrupt OB'sindeki bu komutlar için tampon değerlerini değiştirmek gerekir. Herhangi bir interrupt'ı geciktirmek için, bir EN\_AIRT komutu yürütülene kadar DIS\_AIRT komutunu (bir interrupt OB'si veya bir HMI'dan ya da başka bir CPU'dan gelen bir haberleşme interrupt'ı) kullanınız.

#### Not

DIS\_AIRT komutunun kullanılması, EN\_AIRT komutu kullanılana kadar interrupt OB'lerinin işlemlerini geciktirir. Bu, kullanıcı programınızın interrupt gecikmesini etkiler (bir olaydan interrupt OB'sinin çalıştırıldığı zamana kadar geçen zaman).

## 6.5 Programlama dili

STEP 7, S7-1200 için şu standart programlama dillerini sunar:

- LAD (merdiven mantığı), grafiksel bir programlama dilidir. Gösterim, devre şemaları tabanlıdır (Sayfa 180).
- FBD (Fonksiyon Blok Şeması), Boole cebirinde kullanılan grafiksel sembolleri temel alan bir programlama dilidir (Sayfa 181).
- SCL (Yapısal Kontrol Dili) metin tabanlı, yüksek düzeyli programlama dilidir (Sayfa 181).

Bir kod bloğu oluşturduğunuzda, bu blok tarafından kullanılacak programlama dilini seçersiniz.

### 6.5.1 Merdiven mantığı (LAD)

Bir devre şemasının normal olarak kapalı ve normal olarak açık kontaklar gibi elemanları ve bobinler, ağlar oluşturmak üzere birbirine bağlanır.



Mantık veya karmaşık işlemler oluşturmak amacıyla paralel devreler oluşturmak için kol devreleri ekleyebilirsiniz. Paralel kol devreleri aşağı doğru açılır ya da direkt olarak güç rayına bağlanır. Kol devreleri yukarı doğru sonlandırılır.

LAD, matematik, zamanlayıcı, sayıcı ve taşıma gibi çeşitli fonksiyonlar için "kutu" komutları sunar.

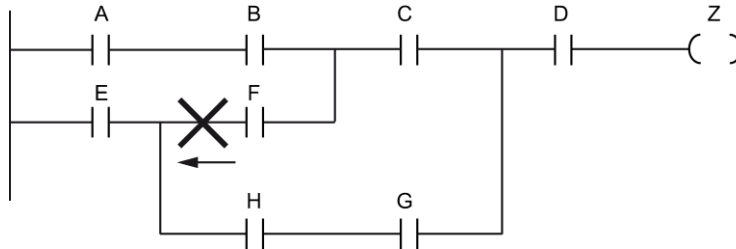
STEP 7, bir LAD ağındaki komutların sayısını (sütunlar ve satırlar) sınırlamaz.

#### Not

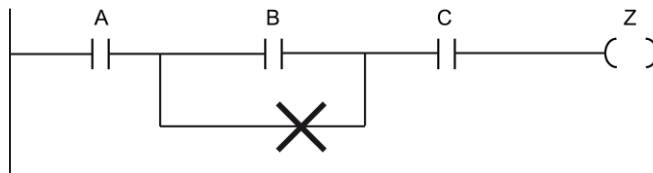
Her bir LAD ağı, bir bobin ya da bir kutu komutu ile sonlandırılmalıdır.

Bir LAD ağı oluştururken şu kuralları dikkate alınız:

- Ters yönde güç akışına neden olabilecek bir kol devresi oluşturamazsınız.

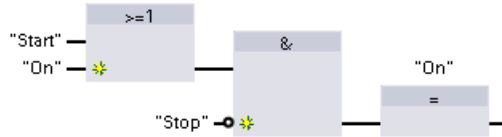


- Kısa devreye neden olabilecek bir kol devresi oluşturamazsınız.



## 6.5.2 Fonksiyon bloğu şeması (FBD)

LAD gibi FBD de grafiksel bir programlama dilidir. Mantığın sunumu Boole cebirinde kullanılan grafiksel mantık simgeleri tabanlıdır.



Karmaşık işlemler için mantık oluşturmak için kutular arasına paralel kol devreleri ekleyiniz.

Matematiksel ve diğer karmaşık fonksiyonlar, mantık kutuları ile direkt olarak temsil edilirler.

STEP 7, bir FBD ağındaki komutların sayısını (sütunlar ve satırlar) sınırlamaz.

## 6.5.3 SCL

Yapısal Kontrol Dili (SCL), SIMATIC S7 CPU'lar için PASCAL-tabanlı, yüksek düzeyli programlama dilidir. SCL, STEP 7 (Sayfa 172)'ye ait blok yapısını destekler. Projeniz takip eden üç programdan herhangi birini içerebilir: SCL, LAD ve FBD.

SCL komutları, değer verme (:=), matematiksel fonksiyonlarda (toplama için +, çıkarma için -, çarpma için \* ve bölme için /) gibi standart programlama operatörlerini kullanılır. SCL ayrıca as IF-THEN-ELSE, CASE, REPEAT-UNTIL, GOTO ve RETURN gibi standart PASCAL program kontrol işlemlerini kullanır. Ayrıca, SCL programlama diline ait söz dizimi elemanları için herhangi bir PASCAL referansını kullanabilirsiniz. SCL için Zamanlayıcı ve sayıcı, matematik, LAD ve FBD komutları gibi diğer birçok komut bulunur. Özel komutlar hakkında daha fazla bilgi için Temel komutlar (Sayfa 203) ve Genişletilmiş komutlar (Sayfa 283) bölümlerindeki özel komutlara bakınız.

Bir blok oluşturduğunuz zaman, SCL programlama dilini kullanmak için herhangi bir blok tipini (OB, FB veya FC) belirleyebilirsiniz. STEP 7, aşağıdaki elemanları içeren SCL program editörünü sağlar:

- Kod bloğunun parametrelerini tanımlayan ara yüz bölümünü
- Program kodu için kod bölümünü
- CPU tarafından desteklenen SCL komutlarını içeren komut ağacını.

Komutunuz için SCL kodunu doğrudan kod bölümüne girersiniz. Editör, ortak kod yapıları ve açıklamaları için düğmeler içerir. Daha karmaşık komutlar için, komut ağacından SCL komutlarını basitçe sürükleyin ve programınızın içine bırakın. Ayrıca bir SCL programı oluşturmak için herhangi bir metin editörünü kullanabilir ve sonra bu dosyayı STEP 7'nin içine taşıyabilirsiniz.

Interface			
	Name	Data type	Comment
1	Input		
2	StartStopSwitch	Bool	
3	Output		
4	RunYesNo	Bool	
5	InOut		
6	<Add new>		
7	Temp		
8	<Add new>		
9	Return		
10	Ret_Val	Void	

IF...	CASE... OF...	FOR... TO DO...	WHILE... DO...	(*...*)
1	IF	condition	THEN	
2		// Statement section IF		
3		;		
4	END_IF;			

SCL kod bloğunun Ara yüz bölümünde aşağıdaki parametrelerin tiplerini beyan edebilirsiniz:

- Input, Output, InOut ve Ret\_Val: Bu parametreler, kod bloğu için girdi etiketlerini, çıktı etiketlerini ve döndürülen değeri tanımlar. Buraya girdiğiniz etiket ismi, kod bloğunun yürütülmesi esnasında yerel olarak kullanılır. Tipik olarak etiket çizelgesinde genel etiket ismini kullanmazsınız.
- Static (sadece FB'ler; yukarıdaki gösterim bir FC içindir): Bu kod bloğu, instance veri bloğundaki statik ara sonuçların saklanması için statik etiketleri kullanır. Bu blok, birkaç döngü sonra olabilecek üstüne yazılmaya kadar statik verileri tutar. Multi-instance (birden çok kopyalı) olarak ta adlandırılan bu blokların isimleri, statik yerel veri içerisinde de saklanır.
- Temp: Bu parametreler, kod bloğunun yürütülmesi esnasında kullanılan geçici etiketlerdir.

Başka bir kod bloğundan SCL kod bloğunu çağırırsanız, SCL kod bloğunun bu parametreleri, girdiler veya çıktılar olarak görünür.



Bu örnekte, "Start" ve "On" için etiketler (proje etiket çizelgesinden gelen), SCL programına ait beyan çizelgesinde "StartStopSwitch" ve "RunYesNo"ya karşılık gelir.

## Bir SCL ifadesini bloklama

Bir SCL ifadesi, bir değer hesaplamak için kullanılan bir formüldür. Bu ifade, operandlardan ve operatörlerden oluşur ( \*, /, + veya - gibi). Bu operandlar, etiketler, sabitler veya ifadeler olabilir. Bu ifadenin değerlendirilmesi, aşağıdaki faktörlerle tanımlanan belirli bir düzen içerisinde gerçekleşir:

- Her bir operatör, en yüksek öncelik çalışması önce gerçekleştirilecek şekilde ön tanımlı bir önceliğe sahiptir.
- Eşit öncelikli operatörler için, bu operatörler soldan sağa bir sıralama ile işlenirler.
- Birlikte değerlendirilmesi gereken bir operatörler dizisini tayin etmek için parantez kullanırsınız.

Bu ifadenin sonucu, ya bir kontrol ifadesi ile kullanılacak bir şart olarak veya diğer SCL komutu için parametreler olarak programınızda kullanılan bir etikete bir değer atamak için ya da bir kod bloğunu çağırarak kullanılabılır.

Çizelge 6- 2 SCL'deki operatörler

Tipi	İşlem	Operatör	Öncelik
Parantez	(İfade)	( , )	1
Math	Güç	**	2
	İşaret (tekil artı)	+	3
	İşaret (tekil eksi)	-	3
	Çarpma	*	4
	Bölme	/	4
	Modulo	MOD	4
	Toplama	+	5
	Çıkarma	-	5
Karşılaştırma	-den az	<	6
	-den az veya eşit	<=	6
	-den büyük	>	6
	-den büyük veya eşit	>=	6
	-e eşit	=	7
	-e eşit değil	<>	7
Bit lojik	Olumsuzlama-değili (tekil)	NOT	3
	AND mantık işlemi	AND or &	8
	Exclusive OR (dışlamalı veya) mantık işlemi	XOR	9
	OR mantık işlemi	OR	10
Atama (Değer verme)	Atama	:=	11

Yüksek düzeyli bir programlama dili olarak, SCL, temel görevler için standard komutları kullanır:

- Atama ifadesi :=
- Matematiksel fonksiyonlar: +, -, \*, and /

- Global değişkenleri adresleme (etiketler): "<tag name>" (çift tırnak ile kapatılmış etiket ismi veya veri bloğu ismi )
- Yerel değişkenleri adresleme: #<variable name> ("#" sembolünden önce gelen değişken ismi)

Aşağıdaki örnekler, farklı kullanımlar için farklı ifadeleri gösterir.

"C" := #A+#B;	İki yerel değişkenin toplamını bir etikete atar
"Data_block_1".Tag := #A;	Bir veri bloğu etiketine atama
IF #A > #B THEN "C" := #A;	IF-THEN ifadesi için şart
"C" := SQRT (SQR (#A) + SQR (#B));	SQRT talimatı için parametreler

Aritmetik operatörler, çeşitli nümerik veri tiplerini işleyebilir. Bu sonucun veri tipi, en anlamlı operandlara ait veri tipi ile belirlenir. Örneğin, INT operandını ve REAL operandını kullanan bir çarpma işlemi, sonuç için REAL değerini verir.

## Kontrol komutları

Bir kontrol komutu, aşağıdaki görevleri gerçekleştiren özel bir SCL ifadesi tipidir:

- Program dallanması
- SCL program kodunun tekrarlanan bölümleri
- SCL programının diğer bölümlerine atlama
- Şartlı yürütme

SCL kontrol komutları, IF-THEN, CASE-OF, FOR-TO-DO, WHILE-DO, REPEAT-UNTIL, CONTINUE, GOTO ve RETURN'ü içerir.

Tek bir komut tipik olarak kodun bir satırını işgal eder. Bir satır üzerinde çoklu komutları girebilirsiniz veya kodun okunmasını daha kolaylaştırmak için bir komutu kodun farklı satırlarına bölebilirsiniz. Ayırıcılar (separators) (sekmeler, satır sonları ve ekstra boşluklar) sözdizimi denetimi sırasında ihmal edilir. Bir END komutu, kontrol komutunu sonlandırır.

Aşağıdaki örnekler, bir FOR-TO-DO kontrol komutunu gösterir. (kodlamanın her iki biçimi sözdizimsel olarak geçerlidir.)

```
FOR x := 0 TO max DO sum := sum + value(x); END_FOR;
FOR x := 0 TO max DO
    sum := sum + value(x);
END_FOR;
```

Bir kontrol komutu ayrıca bir etiketle sağlanabilir. Bir etiket, komutun başlangıcında bir sütunla oluşturulur:

```
Label: <Statement>;
```

STEP 7 çevrimiçi yardımı komple bir SCL programlama dili referansını sağlar.



## Koşullar

Bir koşul, sonucu BOOL türünden (TRUE veya FALSE değerli) olan bir karşılaştırma veya mantık ifadesidir. Aşağıdaki örnek, çeşitli tiplerden koşulları göstermektedir.

<code>#Temperature &gt; 50</code>	İlişkisel ifade
<code>#Counter &lt;= 100</code>	
<code>#CHAR1 &lt; 'S'</code>	
<code>(#Alpha &lt;&gt; 12) AND NOT #Beta</code>	Karşılaştırma ve mantık ifadesi
<code>5 + #Alpha</code>	Aritmetik ifade

Bir koşul aritmetik ifadeler kullanabilir:

İfadenin koşulu eğer sonuç sıfırdan farklı herhangi bir değer ise TRUE olur

İfadenin koşulu eğer sonuç sıfıra eşit ise FALSE olur

## Adresleme

LAD ve FBD'de olduğu gibi, SCL etiketler (sembolik adresleme) veya mutlak adresler kullanmanıza izin verir. SCL aynı zamanda bir değişkeni bir dizin endeksi olarak kullanmanıza izin verir.

### Mutlak adresleme

`%I0.0`  
`%MB100`

Mutlak adresten önce “%” sembolünü koyunuz  
“%” sembolü olmadan, STEP 7 derleme zamanında tanımlanmamış etiket hatası üretir.

### Sembolik adresleme

`"PLC_Tag_1"`  
`"Data_block_1".Tag_1`  
`"Data_block_1".MyArray[#i]`

PLC etiket çizelgesinde etiket  
Bir veri setinde etiket  
Bir veri bloğu dizininde dizin elemanı

## PEEK ve POKE talimatları ile endeksli adresleme

SCL, veri bloklarından, I/O'dan veya hafızadan okumanıza veya buralara yazmanıza izin veren PEEK ve POKE talimatlarını sağlamaktadır. İşlem için belirli byte veya bit ofsetlerine yönelik parametreler sağlarsınız.

### Not

PEEK ve POKE talimatlarını veri blokları ile kullanmak için standart (optimize edilmemiş) veri bloklarını kullanmanız gerekir. Diğer taraftan, PEEK ve POKE talimatlarının sadece veri ilettiğini dikkate alınız. Bu talimatlar adreslerdeki veri türleri hakkında herhangi bir bilgi içermez.

```
PEEK(area:=_in_,
      dbNumber:=_in_,
      byteOffset:=_in_);
```

```
PEEK_WORD(area:=_in_,
           dbNumber:=_in_,
           byteOffset:=_in_);
```

```
PEEK_DWORD(area:=_in_,
            dbNumber:=_in_,
            byteOffset:=_in_);
```

```
PEEK_BOOL(area:=_in_,
          dbNumber:=_in_,
          byteOffset:=_in_,
          bitOffset:=_in_);
```

```
POKE(area:=_in_,
      dbNumber:=_in_,
      byteOffset:=_in_,
      value:=_in_);
```

Referanslı veri bloğu, I/O veya hafıza alanının byteOffset ile referans yapılan byte'nı okur.

Veri bloğuna referans verme örneği

```
%MB100 := PEEK(area:=16#84,
               dbNumber:=1, byteOffset:=#i);
```

IB3 girişe atıf yapan örnek:

```
%MB100 := PEEK(area:=16#81,
               dbNumber:=0, byteOffset:=#i); // when
#i = 3
```

Referanslı veri bloğu, I/O veya hafıza alanının byteOffset ile referans yapılan byte'nı okur.

Örnek:

```
%MW200 := PEEK_WORD(area:=16#84,
                    dbNumber:=1, byteOffset:=#i);
```

Referanslı veri bloğu, I/O veya hafıza alanının byteOffset ile referans yapılan double word'unu okur.

Örnek:

```
%MD300 := PEEK_DWORD(area:=16#84,
                     dbNumber:=1, byteOffset:=#i);
```

Referanslı veri bloğu, I/O veya hafıza alanının byteOffset ve bitOffset ile referans yapılan bir Boolean'ını okur.

Örnek:

```
%MB100.0 := PEEK_BOOL(area:=16#84,
                      dbNumber:=1, byteOffset:=#ii,
                      bitOffset:=#j);
```

Referanslı veri bloğu, I/O veya hafıza alanının referans yapılan byteOffset'ine değer (Byte, Word, DWord) yazar.

Veri bloğuna referans verme örneği

```
POKE(area:=16#84, dbNumber:=2,
      byteOffset:=3, value:"Tag_1");
```

QB3 çıkışına atıf yapan örnek:

```
POKE(area:=16#82, dbNumber:=0,
      byteOffset:=3, value:"Tag_1");
```

```
POKE_BOOL(area:=_in_,
           dbNumber:=_in_,
           byteOffset:=_in_,
           bitOffset:=_in_,
           value:=_in_);
```

```
POKE_BLK(area_src:=_in_,
          dbNumber_src:=_in_,
          byteOffset_src:=_in_,
          area_dest:=_in_,
          dbNumber_dest:=_in_,
          byteOffset_dest:=_in_,
          count:=_in_);
```

Referanslı veri bloğu, I/O veya hafıza alanının referans yapılan byteOffset ve bitOffset'ine Boolean değerini yazar

Örnek:

```
POKE_BOOL(area:=16#84, dbNumber:=2,
           byteOffset:=3, bitOffset:=5,
           value:=0);
```

Referanslı veri bloğu, I/O veya hafıza alanının referans yapılan Offset byte'ında başlayan byte'ların "say" sayısını referanslı destinasyon veri bloğu, I/O veya hafıza alanının referans yapılan byteOffset'ine yazar

Örnek:

```
POKE_BLK(area_src:=16#84,
          dbNumber_src:=#src_db,
          byteOffset_src:=#src_byte,
          area_dest:=16#84,
          dbNumber_dest:=#src_db,
          byteOffset_dest:=#src_byte,
          count:=10);
```

PEEK ve POKE talimatlarında, "area", "area\_src" ve "area\_dest" parametreleri için aşağıdaki değerler uygulanabilir. Veri blokları haricindeki alanlar için dbNumber parametresi 0 olmalıdır.

16#81	I
16#82	Q
16#83	M
16#84	DB

## SCL programınızdan diğer kod bloklarının çağırılması

Kullanıcı programınızda diğer bir kod bloğunu çağırmak için basitçe FB'nin veya FC'nin adını (veya mutlak adres) parametrelerle birlikte giriniz. Bir FB için FB ile çağırılacak olan DB instance'ını sağlamanız gerekir.

<DB name> (Parameter list)

Tekil bir instance olarak çağırınız

<#Instance name> (Parameter list)

Çoklu-instance olarak çağırınız

```
"MyDB" (MyInput:=10, MyInOut:="Tag1");
```

<FC name> (Parameter list)

Standart çağırma

<Operand>:=<FC name> (Parameter list)

İfade içerisinde çağırma

```
"MyFC" (MyInput:=10, MyInOut:="Tag1");
```

Diğer taraftan, gezinti ağacından (navigation tree) veri bloklarını SCL program editörüne sürükleyebilir ve parametre atamalarını tamamlayabilirsiniz.

### SCL koduna blok yorumlar eklenmesi

SCL kodunuza, yorum metnini (\* ve \*) arasına koyarak bir blok yorum koyabilirsiniz. (\* ve \*) arasına koyarak herhangi bir sayıda yorum satırınız olabilir. SCL programınız birçok blok yorum içerebilir. Programlama kolaylığı açısından, SCL editörü ortak kontrol komutlarının yanında bir blok yorum düğmesi içermektedir:

IF...	CASE... OF...	FOR... TO DO...	WHILE... DO...	(*...*)
-------	------------------	--------------------	-------------------	---------

## 6.5.4 LAD, FBD ve SCL için EN ve ENO

### Bir komut için “güç akışının” (EN ve ENO) belirlenmesi

Bazı komutlar (Matematik ve taşıma komutları gibi) EN ve ENO için parametreler sunarlar. Bu parametreler LAD ve FBD içindeki güç akışı ile ilgilidir ve komutun bu tarama sırasında yürütülüp yürütülmediğini belirler. SCL ayrıca bir kod bloğu için ENO parametresini tayin etmenize imkân verir.

- EN (Enable In-Girişi Etkinleştir), bir Boole girişidir. Kutu komutunun yürütülebilmesi için enerji akışı (EN = 1) bulunmalıdır. Bir LAD kutusunun EN girişi direkt olarak sol enerji rayına bağlı ise, komut her zaman yürütülecektir.
- ENO (Enable Out-Çıkışı Etkinleştir) bir Boole çıkışıdır. Eğer kutu, EN girişinde enerji akışına sahipse ve kutu fonksiyonunu hatasız bir şekilde yürütürse, o zaman ENO çıkışı enerji akışını (ENO = 1) bir sonraki elemana geçirir. Eğer kutu komutunun yürütülmesi sırasında bir hata tespit edilirse, o zaman enerji akışı hatayı üreten kutu komutunda sonlandırılır (ENO = 0).

Çizelge 6-3 EN ve ENO için operandlar

Program editörü	Girdiler/Çıktılar	Operandlar	Veri Tipi
LAD	EN, ENO	Güç akışı	Boole
FBD	EN	I, I:P, Q, M, DB, Temp, Power Flow	Boole
	ENO	Power Flow	Boole
SCL	EN <sup>1</sup>	TRUE, FALSE	Boole
	ENO <sup>2</sup>	TRUE, FALSE	Boole

<sup>1</sup> EN'nin kullanımı sadece FB'ler için geçerlidir.

<sup>2</sup> SCL kod bloğu ile birlikte ENO'nun kullanımı isteğe bağlıdır. Kod bloğu bittiğinde, ENO'yu tayin etmek için SCL derleyicisini konfigüre etmelisiniz.

## ENO'yu tayin etmek için SCL'nin konfigüre edilmesi

ENO'nun belirlenmesi için SCL derleyicisini konfigüre etmek üzere aşağıdaki adımları takip ediniz:

1. "Options" menüsünden "Settings" komutunu seçiniz.
2. "PLC programming" özelliklerini genişletin ve SCL'yi (Yapısal Kontrol Dili) seçiniz.
3. Set ENO automatically" seçeneğini seçiniz.

## Program kodunda ENO kullanılması

Program kodunuzda ayrıca ENO'yu kullanabilirsiniz, örneğin, ENO'yu bir PLC etiketine atayarak veya bir yerel blokta ENO değerlendirilerek.

### Örnekler:

```
"MyFunction"  
( IN1 := ... ,  
  IN2 := ... ,  
  OUT1 => #myOut,  
  ENO => #statusFlag ); // PLC tag statusFlag holds the value of  
ENO
```

```
"MyFunction"  
( IN1 := ...  
  IN2 := ... ,  
  OUT1 => #myOut,  
  ENO => ENO ); // block status flag of "MyFunction"  
                // is stored in the local block
```

```
IF ENO = TRUE THEN  
  // execute code only if MyFunction returns true ENO
```

## ENO'da Effect of Ret\_Val veya Status parametreleri

Haberleşme komutları veya dizi (string) dönüşümü komutları gibi bazı komutlar, komutun işlenmesi hakkında bilgiler içeren bir çıktı parametresi sağlar. Örneğin,

Örneğin, bazı talimatlar tipik olarak -32768 ile + 32767 aralığında bir durum bilgisi içeren bir Int veri türü olan bir Ret\_Val (dönüş değeri) sağlar. Diğer talimatlar tipik olarak 16#0000 ile 16#FFFF arasında onaltılık bir aralıktaki durum bilgisi saklayan bir Word veri türü sağlar. Ret\_Val veya Status parametresi içerisinde saklanan sayısal değer bu talimat için ENO durumunu belirler.

- Ret\_Val: 0 ile 32767 arasında bir değer tipik olarak ENO=1 (veya TRUE) yapar. -32768 ile -1 arasında bir değer tipik olarak ENO=0 (veya FALSE) yapar. Ret\_Val'i değerlendirmek için gösterimi onaltılık düzene çeviriniz.
- Status: 16#000 ile 16#7FFF arasında bir değer ENO=1 (veya TRUE) yapar. 16#800 ile 16#FFFF arasında bir değer tipik olarak ENO=0 (veya FALSE) yapar.

Yürütme için birden fazla tarama gerektiren talimatlar sıklıkla, talimatın aktif olduğunu ama yürütmenin tamamlanmadığını işaret eden bir Busy parametresi (Boole) sağlar. Bu talimatlar ayrıca sıklıkla bir Done parametresi (Bool) ve Error parametresi (Boole) sağlar. Done talimatın hatasız tamamlandığını, Error ise talimatın bir hata durumu ile birlikte tamamlandığını işaret eder.

- Busy = 1 (or TRUE) olduğunda, ENO = 1 (veya TRUE).
- Done = 1 (or TRUE) olduğunda, ENO = 1 (veya TRUE).
- Error = 1 (or TRUE) olduğunda, ENO = 0 (veya FALSE).

### Ayrıca Bakınız

OK (Geçerliliği kontrol et) ve NOT\_OK (Geçersizliği kontrol et) talimatları (Sayfa 228)

## 6.6 Koruma

### 6.6.1 CPU için erişim koruması

CPU, belirli fonksiyonlara erişimi kısıtlamak için dört güvenlik seviyesi sunar. Bir CPU için güvenlik seviyesi ve şifre yapılandırıldığında, bir şifre girilmeden erişilebilecek fonksiyonları ve bellek alanlarını sınırlamış olursunuz.

Her seviye, bir şifre olmadan erişilebilecek belirli fonksiyonlara izin verir. CPU için varsayılan koşul, kısıtlama ve şifre korumasının olmamasıdır. Bir CPU'ya erişimi kısıtlamak için CPU'nun özelliklerini yapılandırmalı ve şifreyi girmelisiniz.

Şifrenin bir ağ üzerinden girilmesi, CPU için şifre korumasını tehlikeye atmaz. Şifre koruması, haberleşme fonksiyonları da dahil olmak üzere kullanıcı programınızın yürütmesine uygulanmaz. Doğru şifrenin girilmesiyle bu seviyedeki tüm fonksiyonlara erişim sağlanır.

PCL-PLC haberleşmesi (kod bloklarındaki haberleşme komutları kullanılarak), CPU'daki güvenlik seviyesi tarafından kısıtlanmaz.

Çizelge 6- 4 CPU için güvenlik seviyesi

Güvenlik seviyesi	Erişim kısıtlamaları
Tam erişim (Koruma yok)	Şifre koruması olmadan tam erişime izin verir.
Okuma erişimi	HMI erişimine ve her türlü PLC-PLC haberleşmesine şifre koruması olmaksızın izin verir. CPU'nun değiştirilmesi (üzerine yazılması) ve CPU modunun (RUN/STOP) değiştirilmesi için şifre gerekir.
HMI erişimi	HMI erişimine ve her türlü PLC-PLC haberleşmesine şifre koruması olmaksızın izin verir. CPU'da veri okunmasında, CPU'nun değiştirilmesi (üzerine yazılması) ve CPU modunun (RUN/STOP) değiştirilmesi için şifre gerekir.
Erişim yok (Tam koruma)	Şifre koruması olmadan hiçbir erişime izin vermez. HMI erişimi için, CPU'da veri okunmasında, CPU'nun değiştirilmesi (üzerine yazılması) ve CPU modunun (RUN/STOP) değiştirilmesi için şifre gerekir.

Şifreler büyük-küçük harfe duyarlıdır. Koruma seviyesini ve şifreleri konfigüre etmek için aşağıdaki adımları takip ediniz:

1. "Device configuration" içerisinde CPU'yu seçiniz.
2. Denetçi penceresinde "Properties" (özellikleri) sekmesini seçiniz.
3. Koruma seviyesini seçmek ve şifreleri girmek için "Protection" (koruma) özelliğini seçiniz.

**Protection**

Select the access level for the PLC.

Access level	Access			Access permission	
	HMI	Read	Write	Password	Confirmation
<input type="radio"/> Full access (no protection)	✓	✓	✓	*****	*****
<input type="radio"/> Read access	✓	✓		*****	*****
<input checked="" type="radio"/> HMI access	✓				
<input type="radio"/> No access (complete protection)					

Bu konfigürasyonu CPU'ya indirdiğinizde, kullanıcının HMI girişi vardır ve şifre kullanmadan HMI işlevlerine erilebilir. Veri okumak için, kullanıcı "Read Access" veya "Full Access (koruma yok)" için konfigüre edilmiş şifreyi girmelidir. Veri yazmak için, kullanıcı "Full Access (koruma yok)" için konfigüre edilmiş şifreyi girmelidir.

 **UYARI****Korumalı CPU'ya yetkisiz erişim**

Tam erişim yetkileri olan kullanıcılar PLC değişkenlerini okuma ve yazma yetkisine sahiptir. CPU'nun erişim seviyesinden bağımsız olmak üzere, Web kullanıcıları PLC değişkenlerini okuma ve yazma yetkisine sahip olabilirler. Yetkisiz şekilde CPU'ya erişim veya PLC değişkenlerini geçersiz değerler ile değiştirmek sürecin çalışmasını bozabilir ve ölüm, ciddi kişisel yaralanma ve/veya mal hasarı ile sonuçlanabilir.

Yetkili kullanıcılar çalışma modu değişimi, PLC verisine yazma ve donanım yazılımı güncellemeleri yapabilir. Siemens aşağıdaki güvenlik tedbirlerini dikkate almanızı tavsiye eder.

CPU erişim seviyelerini ve Web kullanıcı ID'lerini (Sayfa 604) güçlü şifrelerle koruyunuz. Güçlü şifreler; en azından 8 karakter uzunluğundadır, harf-sayı-özel karakter karışımından oluşur, sözlükte bulunabilen kelimeler değildir ve kişisel bilgilerden türetilebilecek isimler veya tanımlayıcılar değildir. Şifreyi gizli tutunuz ve sıklıkla değiştiriniz.

Web sunucuya sadece HTTPS protokolü ile erişime izin veriniz.

Web sunucusu "Everybody" kullanıcısının ön tanımlı minimum yetkilerini genişletmeyiniz.

- Programla mantığınızdaki değişkenleriniz için hata-kontrolü ve aralık-kontrolü yapınız. Çünkü web sayfası kullanıcıları PLC değişkenlerini geçersiz değerlerle değiştirebilirler.

**Bağlantı mekanizmaları**

Ortaklara PUT/GET talimatları için uzaktan erişim vermek için, kullanıcının aynı zamanda izni olmalıdır.

"Permit access with PUT/GET communication" seçeneği ön tanımlı olarak etkin değildir. Bu durumda, CPU verisini okuma ve yazma erişimi sadece hem yerel CPU hem de iletişim ortağı için konfigürasyon veya programlama gerektiren iletişim bağlantıları için mümkündür. Örneğin, BSEND/BRCV talimatları aracılığıyla erişim mümkündür.

Bu nedenle, yerel CPU'nun sadece bir sunucu olduğu (yani yerel CPU'da iletişim ortağı ile iletişim için herhangi bir konfigürasyon/programlama yok anlamına gelen) bağlantılar CPU'nun çalışması sırasında mümkün değildir. Örneğin;

- İletişim modülü aracılığıyla PUT/GET, FETCH/WRITE veya FTP erişimi
- Diğer s/ CPU'larından PUT/GET erişimi
- PUT/GET iletişimi aracılığıyla HMI erişimi



CPU'ya istemci tarafından erişime imkân tanımak isterseniz, yani, CPU'nun haberleşme servislerini kısıtlamak istemezseniz aşağıdaki adımları takip ediniz:

1. "No access (complete protection)" (Erişim yok (Tam koruma)) dışındaki herhangi bir seviyeye olabilecek koruma erişim seviyesini konfigüre ediniz.
2. "Permit access with PUT/GET communication" kontrol kutusunu seçiniz.



Bu konfigürasyonu CPU'ya indirdiğiniz zaman, CPU, uzak ortaklardan PUT/GET haberleşmesine izin verir.

## 6.6.2 Know-how koruması

Know-how koruması, programınızdaki bir veya birden fazla kod bloğunu (OB, FB, FC veya DB) izinsiz erişimden korur. Kod bloğuna erişimi sınırlamak için bir şifre oluşturulur. Bu şifre, kod bloğunda yapılacak okuma ve değişiklik (yazma) işlemlerine karşı şifre koruması sunar. Şifre olmadan, kod bloğu hakkında sadece şu bilgileri okuyabilirsiniz:

- Blok başlığı, blok yorumu ve blok özellikleri
- Transfer parametreleri (IN, OUT, IN\_OUT, Return ---GİRİŞ, ÇIKIŞ, GİRİŞ\_ÇIKIŞ, Dönüş)
- Programın çağrı yapısı
- Çapraz referanslardaki global etiketler (kullanım noktası bilgileri olmadan), ancak lokal etiketler gizlenir

Know-how koruması için bir blok yapılandırıldığında, şifre girişi haricinde blok içindeki koda erişilemez.

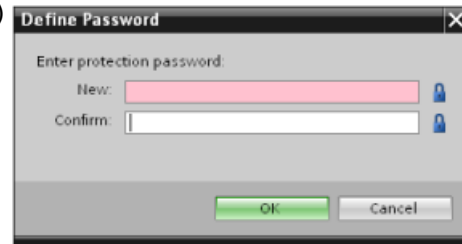
Bu blok için know-how korumasını yapılandırmak için kod bloğunun "Properties" (Özellikler) görev kartını kullanın. Kod bloğunu açtıktan sonra "Properties"den (Özelliklerden) "Protection"ı (Koruma'yı) seçiniz.



1. "Know-how protection" (Know-how koruması) diyalogunu açmak için kod bloğunun Özelliklerinden "Protection" (Koruma) düğmesine basınız.
2. Şifreyi girmek için "Define" (Tanımla) düğmesine tıklayınız.



Şifreyi girip onayladıktan sonra "OK" (tamam) düğmesine tıklayınız.



### 6.6.3 Kopya koruması

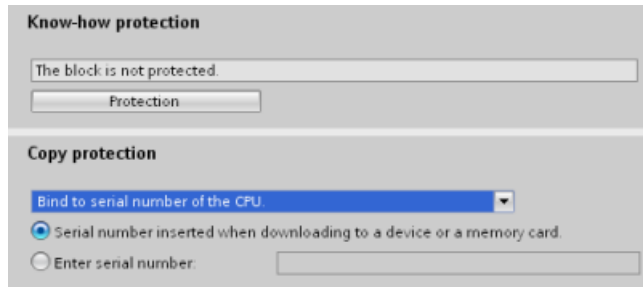
Ek bir güvenlik özelliği, programı ya da kod bloklarını özel bir hafıza kartı ya da CPU ile kullanmanıza izin verir. Bu özellik bilhassa fikri mülkiyetinizin korunmasında oldukça faydalıdır. Bir programı ya da bloğu belirli bir cihaza bağladığınızda, programı ya da kod bloğunu sadece belirli bir hafıza kartı ya da CPU ile kullanılabilir şekilde kısıtlamış olursunuz. Bu özellik sayesinde bir programı veya kod bloğunu elektronik olarak dağıtabilir (mesela e-posta aracılığıyla internet üzerinden) ya da bir depolama cihazı içinde gönderebilirsiniz. Kopya koruması, OB'ler (Sayfa 172), FB'ler (Sayfa 175) ve FC'ler (Sayfa 174) için mevcuttur.

Bloğu belirli bir CPU'ya veya hafıza kartına bağlamak için kod bloğunun "Properties" (Özellikler) görev kartını kullanınız.

1. Kod bloğunu açtıktan sonra "Protection"ı (Koruma'yı) seçiniz.



2. "Copy protection" (Kopya koruma) görevinin altındaki açılır listeden kod bloğunu bir hafıza kartına ya da belirli bir CPU'ya bağlama seçeneğini seçiniz.



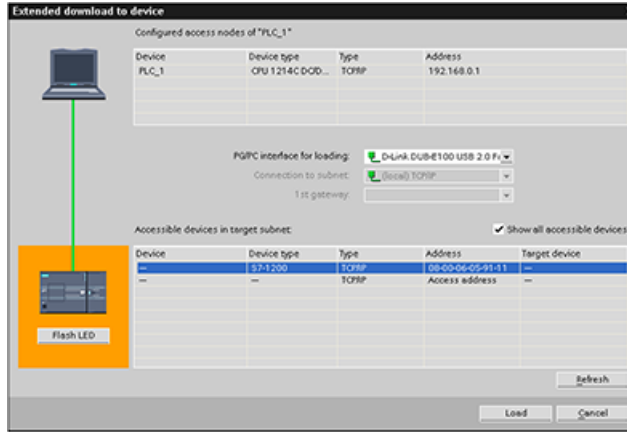
3. Kopya koruma tipini seçiniz. ve hafıza kartının veya CPU'nun seri numarasını giriniz.

#### Not

Seri numarası büyük ve küçük harfe duyarlıdır.

## 6.7 Programınızın elemanlarını yüklemek

Projenizin elemanlarını programlama cihazından CPU'ya yükleyebilirsiniz. Bir proje yüklediğinizde, CPU kullanıcı programını (OB'ler, FC'ler, FB'ler ve DB'ler) kalıcı bellekte saklar.



Projenizi, programlama cihazından CPU'nuzaya aşağıdaki konumlardan herhangi birinden yükleyebilirsiniz:

- "Project tree" (Proje ağacı): Program elemanına sağ tıklayınız ve sonra bağlam duyarlı "Download" (yükleme) seçimine tıklayınız.
- "Online" menüsü: "Download to device" (Cihaza yükle) seçimine tıklayınız.
- Toolbar (Araç çubuğu): "Download to device" (Cihaza yükle) simgesine tıklayınız.

### Not

Program yükleme, silmez veya kuvvetli hafızadaki mevcut değerlere herhangi bir değişiklik yapmaz. Bir yükleme öncesinde kuvvetli hafızayı silmek isterseniz, bu durumda programı yükleme öncesinde CPU'nuzu fabrika ayarlarına getiriniz (resetleyiniz).

Ayrıca, Temel HMI panelleri (Sayfa 32) için TIA Portalından S7-1200 CPU'sundaki bir hafıza kartına bir panel projesini de yükleyebilirsiniz.

## 6.8 CPU'dan yükleme

### 6.8.1 Projenin elemanlarının kopyalanması

Program bloklarını bir çevrimiçi CPU'dan ya da programlama cihazınıza bağlı bir hafıza kartından da kopyalayabilirsiniz.

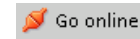
Offline projeyi, kopyalanan program blokları için hazırlayınız:

1. Online CPU'ya uygun bir CPU cihazı ekleyiniz.
2. CPU nodunu "Program blocks" (Program blokları) klasörü görünür olacak şekilde genişletiniz.



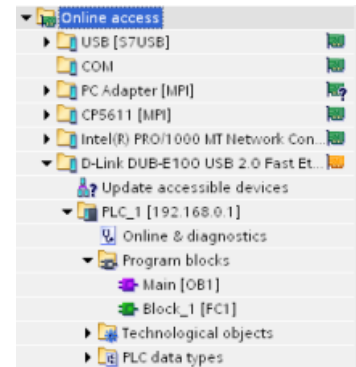
Online CPU'dan offline projeye program bloklarını yüklemek için aşağıdaki adımları takip ediniz:

1. Offline projedeki "Program blocks" (program blokları) klasörüne tıklayınız.
2. "Go online" (çevrimiçiye git) düğmesine tıklayınız.
3. "Upload" (yükle) düğmesine tıklayınız.
4. Kararınızı yükle diyalogundan (Sayfa 786) teyit ediniz.

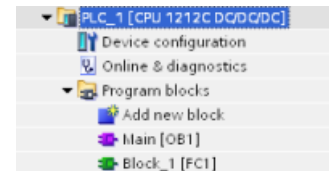


Önceki yöntemle bir alternatif olarak aşağıdaki adımları takip ediniz:

1. Online CPU'daki program bloklarını seçmek için proje navigatöründe "Online access" (Online erişim) nodunu genişletiniz:
2. Ağ için olan nodu genişletin ve Update accessible devices" (Erişilebilir cihazları güncelle) üzerine çift tıklayınız.
3. CPU için olan nodu genişletiniz.
4. Online CPU'dan "Program blocks" (Program blokları) klasörünü sürükleyin ve klasörü offline projenizin "Program blocks" (Program blokları) klasörünün içine bırakınız.
5. "Upload preview" (Yükleme önizleme) diyalogundan "Continue" (Devam et) kutusunu seçiniz. ve sonra "Upload from device" (Cihazdan yükle) düğmesine tıklayınız.



Yükleme tamamlandığında, tüm program blokları, teknoloji blokları ve tag'ler offline alanda görüntülenir.



### Not

Program bloklarını çevrimiçi CPU'dan mevcut bir programa kopyalayabilirsiniz. Offline projenin "Program-blocks" (Program blokları) klasörü boş olmak zorunda değildir. Ancak mevcut program silinecek ve online CPU'dan gelen kullanıcı programı ile değiştirilecektir.

## 6.8.2 Karşılaştırma fonksiyonunu kullanmak

Online ve offline projeleri arasındaki farkları bulmak için STEP 7'deki "Karşılaştırma" editörünü (Sayfa 794) kullanabilirsiniz. CPU'dan yükleme öncesinde bunu kullanışlı bulabilirsiniz.

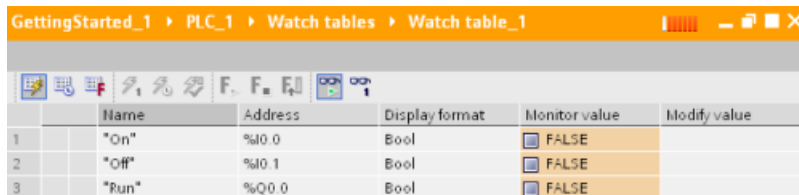
## 6.9 Programda hata ayıklama ve test işlemleri

### 6.9.1 CPU'daki verilerin izlenmesi ve değiştirilmesi

Aşağıdaki çizelgeye göre çevrimiçi CPU'daki değerleri izleyebilir ve değiştirebilirsiniz.

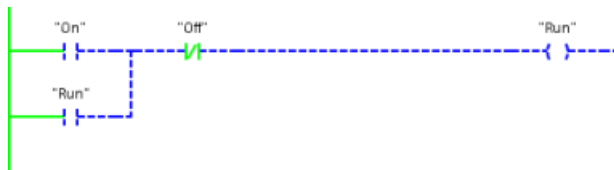
Çizelge 6- 5 Verilerin STEP 7 ile izlenmesi ve değiştirilmesi

Editör	İzle	Değiştir	Zorla
İzleme çizelgesi	Evet	Evet	Hayır
Zorla çizelgesi	Evet	Hayır	Evet
Program editörü	Evet	Evet	Hayır
Etiket çizelgesi	Evet	Hayır	Hayır
DB editörü	Evet	Hayır	Hayır



Name	Address	Display format	Monitor value	Modify value
"On"	%I0.0	Bool	<input type="checkbox"/> FALSE	
"Off"	%I0.1	Bool	<input type="checkbox"/> FALSE	
"Run"	%Q0.0	Bool	<input type="checkbox"/> FALSE	

Bir izleme çizelgesi ile izleme



LAD editörü ile izleme

CPU'daki verilerin izlenmesi ve değiştirilmesiyle ilgili daha fazla bilgi için "Online ve diyagnostikler" bölümüne bakabilirsiniz (Sayfa 795).

## 6.9.2 İzleme tabloları ve zorlama tabloları

İzleme tabloları, online CPU tarafından yürütülen bir kullanıcı programının değerlerini izlemek ve değiştirmek için kullanılır. Çeşitli test ortamlarını desteklemek için projenizde farklı izleme tabloları oluşturabilir ve bunları kaydedebilirsiniz. Bu size devreye alma ya da servis ve bakım amaçlı testleri tekrarlama olanağı sunar.

Bir izleme çizelgesi ile CPU kullanıcı programını yürütürken onunla etkileşime geçebilir ve izleme yapabilirsiniz. Kod bloklarının ve data blokların etiketlerinin değerlerini izleme ve değiştirmenin dışında bu işlemleri CPU'nun bellek alanları için de gerçekleştirebilirsiniz. Buna girdiler ve çıktılar (I ve Q), çevresel girişler (I:P), bit belleği (M) ve data blokları (DB) dahildir.

İzleme çizelgesi ile STOP modundaki bir CPU'nun fiziksel çıktıları (Q:P) devreye alabilirsiniz. Örneğin CPU için kabloları test ederken çıkışlara özel değerler atayabilirsiniz.

STEP 7 ayrıca bir etiketi özel bir değere zorlamak için zorla (force) çizelgesi sunar. Zorlama (Forcing) hakkında daha fazla bilgi almak için "Online ve diyagnostik" bölümündeki CPU'daki zorla (force) değerleri kısmına bakabilirsiniz (Sayfa 802).

---

### Not

Zorla (Force) değerleri izleme çizelgesinde değil CPU'da saklanır.

Bir girişte (ya da "I" adresinde) zorla (force) yapamazsınız. Ancak bir çevresel girdide zorla (force) yapılabilir. Bir çevresel girişi zorla (force) yapmak için adrese ":P" ekleyin (örnek: "On:P") "On:P".

---

STEP 7 ayrıca, tetikleme koşullarına dayalı olarak program değişkenlerini izleme kayıt etme yeteneği sağlar (Sayfa 81).

## 6.9.3 Kullanımı göstermek için çapraz başvuru

Denetçi penceresi, seçilen objenin tüm projede nasıl kullanıldığı hakkındaki çapraz başvuru bilgilerini görüntüler (kullanıcı programı, CPU ve herhangi bir HMI cihazı gibi). Çapraz başvuru" sekmesi, halihazırda kullanılmakta olan ve kullanılan objenin projede kullanıldığı yerleri görüntüler. Denetçi penceresi aynı zamanda sadece çapraz başvurularda çevrimiçi olarak bulunan blokları da içerir. Çapraz başvuruları görüntülemek için "Show cross-references" (Çapraz başvuruları göster) komutunu seçiniz. (Proje görünümünde, "Tools" (Araçlar) menüsünden çapraz başvuruları bulunuz.)

---

### Not

Çapraz başvuru bilgilerini görmek için editörü kapatmak zorunda değilsiniz.

---

Çapraz başvurudaki girişleri sıralayabilirsiniz. Çapraz başvuru listesi, kullanıcı programı içindeki etiketleri ve bellek adreslerinin kullanımının bir genel bakışını sunar.

- Bir program oluştururken ve programı değiştirirken, kullandığınız blok çağrılarının, tag'lerin ve operandların bir genel bakışını alırsınız.
- Çapraz başvurulardan, operandların ve etiketlerin kullanım noktasına direkt olarak atlayabilirsiniz.
- Bir program testi ya da sorun giderme sırasında, hangi bellek konumunun hangi bloktaki hangi komut tarafından işlendiği, hangi ekranda hangi tag'in kullanılmakta olduğu ve hangi blok tarafından hangi diğer bloğun çağrıldığı hakkında bilgilendirilirsiniz.

Çizelge 6- 6 Çapraz başvurunun elemanları

Sütun	Açıklama
Obje	Alt seviye objelerini kullanan ya da alt seviye objeleri tarafından kullanılan objenin adı
Miktar	Kullanımların sayısı
Konum	Her bir kullanımın konumu, mesela ağ
Özellik	Referans objelerin belirli özellikleri, mesela multi- instance beyanlarındaki etiket isimleri
Ne olarak	Objeye hakkındaki ilave bilgileri gösterir, mesela instance DB'sinin şablon olarak mı yoksa çoklu- instance olarak mı kullanıldığı gibi
Erişim	Erişimin tipi, operanda erişimin okuma erişimi mi (R) yoksa yazma erişimi mi (W) ya da bunların ikisi birden mi olduğu
Adres	Operandın adresi
Tip	Objenin oluşturulması için kullanılan tip ve dil hakkındaki bilgiler
Yol	Objenin proje ağacındaki yolu



#### 6.9.4 Çağırma hiyerarşisini incelemek için çağrı yapısı

Çağrı hiyerarşisi, kullanıcı programınız içindeki bloğun çağrı hiyerarşisini açıklar. Kullanılan blokların, diğer bloklara yapılan çağrılarının, bloklar arasındaki ilişkilerin, her bir blok için veri gerekliliklerinin ve blokların durumlarının genel bir bakışını sunar. Program editörünü açabilir ve çağrı yapısından blokları düzenleyebilirsiniz.

Çağrı yapısının görüntülenmesi, size kullanıcı programında kullanılan blokların bir listesini sağlar. STEP 7 çağrı yapısının ilk seviyesini vurgular ve programdaki başka bir blok tarafından çağrılmayan tüm blokları görüntüler. Çağrı yapısının ilk seviyesi OB'leri ve bir OB tarafından çağrılmayan tüm FC'leri, FB'leri ve DB'leri görüntüler. Eğer bir kod bloğu başka bir bloğu çağırırsa, çağrılan blok çağırılan bloğun altında bir girinti olarak gösterilir. Çağrı yapısı, sadece bir kod bloğu tarafından çağrılan blokları görüntüler.

Sadece çağrı yapısı ile çakışmaya neden olan blokları seçerek görüntüleyebilirsiniz. Aşağıdaki koşullar çakışmaya neden olur:

- Daha eski veya daha yeni kod zamanı damgalarına sahip çağrıları yürüten bloklar
- Bir bloğu değiştirilen ara yüzle çağırılan bloklar
- Bir etiketi değiştirilen adresle ve/veya veri tipi ile kullanan bloklar
- Bir OB tarafından ne doğrudan ne de dolaylı olarak çağrılmayan bloklar
- Olmayan ya da kayıp bir bloğu çağırılan bloklar

Birkaç blok çağrısını ve data bloklarını gruplayarak bir grup haline getirebilirsiniz. Çeşitli çağrı konumlarına linkleri görmek için açılır menü listesini kullanınız.

Zaman damgası çakışmalarını göstermek için bir tutarlılık kontrolü de gerçekleştirebilirsiniz. Bir bloğun zaman damgasının program oluşturulurken ya da program oluşturulduktan sonra değiştirilmesi, çağırılan ve çağrılan bloklar arasında tutarsızlıklara neden olabilir.

- Çoğu zaman damgası ve ara yüz çakışması, kod bloklarının tekrar derlenmesiyle düzeltilebilir.
- Eğer derleme tutarsızlıkların ortadan kaldırılmasında başarısız olursa, sorunun program editöründe kaynağına inmek için "Ayrıntılar" sütunundaki linki kullanın. Her türlü tutarsızlığı bundan sonra manüel olarak ortadan kaldırabilirsiniz.
- Kırmızıyla işaretlenmiş tüm bloklar tekrar derlenmelidir



## Temel komutlar



### 7.1 Bit lojik alıřmaları

#### 7.1.1 Bit lojik komutları

LAD ve FBD, Boole lojik iřlenmesi iin hayli etkilidir. SCL karmařık matematiksel hesaplama ve proje kontrol yapıları iin zellikle etkili olduėundan Boole lojik iin kullanabilirsiniz.

#### LAD kontakları

izelge 7- 1 Normal olarak aık ve normal olarak kapalı kontaklar

LAD	SCL	Aıklama
"IN" 	<pre>IF in THEN     Statement; ELSE     Statement; END_IF;</pre>	Normal olarak aık ve normal olarak kapalı kontaklar: Kontakları diėer kontaklara baėlayabilir ve kendi kombinasyon mantıėını oluřturabilirsiniz. Eėer belirlediėiniz giriř biti I (giriř) ya da Q (ıkıř) hafızası tanımlayıcısını kullanıyorsa o zaman bit deėeri proses imajı kaydından okunur. Kontrol prosesinizdeki fiziksel kontak sinyalleri, PLC zerindeki I terminallerine baėlıdır. CPU kablo ile baėlanan giriř sinyallerini tarar ve proses imajı giriř kaydındaki karřılık gelen durum deėerlerini srekli olarak gnceller.
"IN" 	<pre>IF NOT (in) THEN     Statement; ELSE     Statement; END_IF;</pre>	Ofsetten sonra ":P" kullanarak fiziksel giriřin anında okunmasını gerekleřtirebilirsiniz (rnek: "%I3.4:P"). Bir anında okuma iin bit veri deėerleri, proses imajı yerine direkt olarak fiziksel giriřten okunur. Bir direkt olarak okuma proses imajını gncellemez.

izelge 7- 2 Parametreler iin veri tipleri

Parametre	Veri tipi	Aıklama
IN	Bool	Atanmıř bit

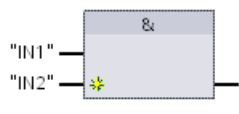
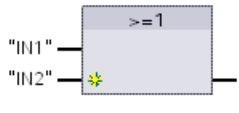
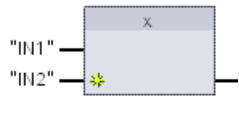
- Normal Olarak Aık kontak, atanan bit deėeri 1'e eřit olduėunda kapanır.
- Normal Olarak Kapalı kontak, atanan bit deėeri 0'a eřit olduėunda kapanır.
- Seri olarak baėlanan kontaklar AND mantıklı aėlar oluřturur.
- Paralel olarak baėlanan kontaklar OR mantıklı aėlar oluřturur.

### FBD AND, OR ve XOR kutuları

FBD programlamasında, LAD kontak ağları AND (&), OR (>=1) ve exclusive OR (x) kutusu ağlarına dönüştürülür. Burada kutu girişleri ve çıkışları için bit değerlerini belirleyebilirsiniz. Diğer mantık kutularıyla da bağlantı kurabilir ve kendi mantık kombinasyonlarınızı oluşturabilirsiniz. Kutu ağınızda konumlandırıldıktan sonra, "Favorites" (Favoriler) araç çubuğundan ya da komut ağacından "Insert input" (giriş ekle) aracını sürükleyip kutunun giriş tarafına bırakarak daha fazla giriş ekleyebilirsiniz. Ayrıca kutu giriş konektörüne sağ tıklayıp "Insert input" (Giriş ekle) seçimini de yapmanız mümkündür.

Kutu girişleri ve çıkışları, başka bir mantık kutusuna bağlayabilir ya da bağlı olmayan bir giriş için bir bit adresi ya da bit sembol ismi girebilirsiniz. Kutu komutu yürütüldüğünde geçerli giriş durumları binary kutu mantığına uygulanır ve eğer true ise kutu çıkışı da true olacaktır.

Çizelge 7-3 AND, OR ve XOR kutuları

FBD	SCL <sup>1</sup>	Açıklama
	<pre>out := in1 AND in2;</pre>	Çıkışın TRUE olması için bir AND kutusunun tüm girişleri TRUE olmalıdır.
	<pre>out := in1 OR in2;</pre>	Çıkışın TRUE olması için bir OR kutusunun herhangi bir girişi TRUE olmalıdır.
	<pre>out := in1 XOR in2;</pre>	Çıkışın TRUE olması için bir XOR kutusunun tek sayı miktarındaki girişleri TRUE olmalıdır.


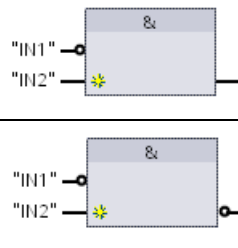
<sup>1</sup> SCL için: İşlem sonucu başka bir komut için kullanılacak olan bir değişkene atamalısınız.

Çizelge 7-4 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN1, IN2	Bool	Giriş biti

## NOT (DEĞİL) mantık çeviricisi

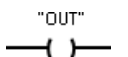

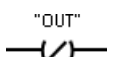
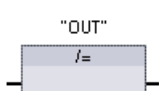
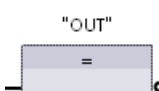
Çizelge 7- 5 Invert RLO (Mantık İşleminin Sonucu)

LAD	FBD	SCL	Açıklama
		<p>NOT (DEĞİL)</p>	<p>FBD programlaması için, "Favorites" (Favoriler) araç çubuğundan ya da komut ağacından "Negate binary input" (Binary girişi reddet) aracını sürükleyebilir ve sonra bu kutu konektörü üzerinde bir mantık çeviricisi oluşturmak için bir çıkış ya da giriş üzerine bırakabilirsiniz.</p> <p>LAD NOT kontağı, güç akış girişinin mantıksal durumunu tersine çevirir.</p> <ul style="list-style-type: none"> <li>Eğer NOT kontağına hiç enerji akışı yoksa, o zaman dışarı enerji akışı vardır.</li> <li>Eğer NOT kontağına enerji akışı varsa, o zaman dışarı enerji akışı yoktur.</li> </ul>

## Çıkış bobini ve atama kutusu

Bobin çıkış komutu, bir çıkış biti için bir değer yazar. Eğer belirlediğiniz çıkış biti Q hafızası tanımlayıcısını kullanırsa, o zaman CPU proses imajındaki çıkış bitini açıp ya da kapatarak belirtilen biti güç akış durumuna eşit olarak ayarlar. Kontrol aktuatörleriniz için çıkış sinyalleri, CPU'nun Q terminallerine bağlanır. RUN modunda CPU sistemi sürekli olarak giriş sinyallerinizi tarar, giriş durumlarındaki prosesleri program mantığınıza göre işler ve sonra proses imajı çıkışı kaydındaki yeni çıkış durumu değerlerini ayarlayarak tepki verir. CPU sistemi, proses imajı kaydındaki yeni çıkış durumu tepkisini bağlı olunan çıkış terminallerine transfer eder.

Çizelge 7- 6 Atama ve red ataması

LAD	FBD	SCL	Açıklama
		<p>out := &lt;Boolean expression&gt;;</p>	<p>FBD programlamasında, LAD bobinleri atama kutularına (= ve /=), çıkış kutusu için bir bit adresi belirlediğiniz yerde dönüştürülür. Kutu girişleri ve çıkışları diğer kutu mantığına bağlanabilir ya da siz bir bit adresi girebilirsiniz.</p>
		<p>out := NOT &lt;Boolean expression&gt;;</p>	<p>Q ofsetinden sonra ":"P" kullanarak fiziksel çıkışın anında yazılmasını sağlayabilirsiniz (örnek: "%Q3.4:P"). Bir anında yazma için, bit veri değerleri proses imajı çıkışına ve direkt olarak fiziksel çıkışa yazılır.</p>
			

## Temel komutlar

### 7.1 Bit lojik çalışmaları

Çizelge 7-7 Parametreler için veri tipleri

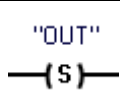
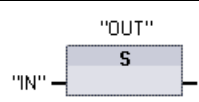
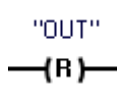
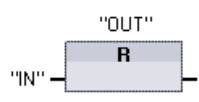
Parametre	Veri Tipi	Açıklama
OUT	Bool	Atanmış bit

- Eğer bir çıkış bobini üzerinden enerji akışı varsa ya da bir FBD “=” kutusu etkinleştirilmişse, o zaman çıkış biti 1 olarak ayarlanır.
- Eğer bir çıkış bobini üzerinden enerji akışı yoksa ya da bir FBD “=” atama kutusu üzerinden etkinleştirilmemişse, o zaman çıkış biti 0 olarak ayarlanır.
- Eğer çevrilmiş bir çıkış bobini üzerinden enerji akışı varsa ya da bir FBD “/=” kutusu etkinleştirilmişse, o zaman çıkış biti 0 olarak ayarlanır.
- Eğer çevrilmiş bir çıkış bobini üzerinden enerji akışı yoksa ya da bir FBD “/=” kutusu üzerinden etkinleştirilmemişse, o zaman çıkış biti 1 olarak ayarlanır.

### 7.1.2 Set ve reset komutları

#### Set ve Reset 1 biti

Çizelge 7-8 S ve R komutları

LAD	FBD	SCL	Açıklama
		Mevcut değil	Çıkışı Set edilmesi: S (Set) etkinleştirildiğinde, ÇIKIŞ adresindeki veri değeri 1 olarak ayarlanır. S etkinleştirilmediğinde, ÇIKIŞ değiştirilmez.
		Mevcut değil	Çıkışın Reset edilmesi: R (Reset) etkinleştirildiğinde, ÇIKIŞ adresindeki veri değeri 0 olarak ayarlanır. R etkinleştirilmediğinde, ÇIKIŞ değiştirilmez.

<sup>1</sup> LAD ve FBD için: Bu komutlar ağda herhangi bir yere yerleştirilebilir.

<sup>2</sup> SCL için: Bu fonksiyonu uygulamanızda kopyalamak için kod yazmalısınız.

Çizelge 7-9 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN (GİRİŞ) (ya da kontak/kapı mantığına bağlantı)	Bool	İzlenecek konumun bit etiketi
OUT (ÇIKIŞ)	Bool	Set veya reset olarak ayarlanacak konumun bit etiketi

## Set ve Reset Biti Alanı

Çizelge 7- 10 SET\_BF ve RESET\_BF komutları

LAD <sup>1</sup>	FBD	SCL	Açıklama
<pre> "OUT" --(SET_BF)-- "n" </pre>	<pre> "OUT" SET_BF EN N </pre>	Mevcut değil	Bit alanının Set edilmesi: SET_BF etkinleştirildiğinde, ÇIKIŞ adresinde başlayan 'n' bitleri, 1 veri değeri olarak atanır. SET_BF etkinleştirilmediğinde, ÇIKIŞ değiştirilmez
<pre> "OUT" --(RESET_BF)-- "n" </pre>	<pre> "OUT" RESET_BF EN N </pre>	Mevcut değil	Bit alanının Reset edilmesi: RESET_BF, ÇIKIŞ adresinde başlayan 'n' bitleri, 0 veri değeri olarak yazar. RESET_BF etkinleştirilmediğinde, ÇIKIŞ değiştirilmez

- 1 LAD ve FBD için: Bu komutlar bir kolda en sağ komutlar olmalıdır.
- 2 SCL için: Bu fonksiyonu uygulamanızda kopyalamak için kod yazmalısınız.

Çizelge 7- 11 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
OUT	Bool	Bir alanının başlangıç elemanı set veya reset olarak ayarlanır (Örnek : #MyArray[3])
n	Constant (Sabit) (UInt)	Yazılacak bitlerin sayısı

## Set ve Reset öncelikli RS Flip Flop

Çizelge 7- 12 RS ve SR komutları

LAD / FBD	SCL	Açıklama
<pre> "INOUT" RS R S1 Q </pre>	Mevcut Değil	Flip-flop'un resetlenmesi/set edilmesi: RS, Set öncelikli flipflop, setin resete karşı önceliği olduğu kilitleme elemanıdır. Eğer set (S1) ve reset (R) sinyallerinin ikisi de TRUE ise, çıkış adresi INOUT 1 olacaktır.
<pre> "INOUT" SR S R1 Q </pre>	Mevcut Değil	Flip-flop'un set edilmesi/ resetlenmesi: SR, Reset öncelikli flipflop, resetin sete karşı önceliği olduğu kilitleme elemanıdır. Eğer set (S) ve reset (R1) sinyallerinin ikisi de TRUE ise, çıkış adresi INOUT 0 olacaktır.

- 1 LAD ve FBD için: Bu komutlar, bir koldaki en sağ konumda olmalıdır.
- 2 SCL için: Uygulamanızın içerisindeki bu fonksiyonun tekrarlanması için kod yazmalısınız.

## Temel komutlar

### 7.1 Bit lojik çalışmaları

Çizelge 7- 13 Parametreler için veri tipleri

Parametre	Veri Tipi	Açıklama
S, S1	Bool	Set girişi; 1 dominantlığı belirtir
R, R1	Bool	Reset girişi; 1 dominantlığı belirtir
INOUT	Bool	Atanan bit çıkışı "INOUT"
Q	Bool	"INOUT" bitinin durumunu izler

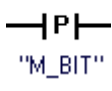
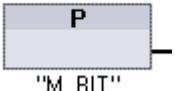
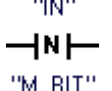
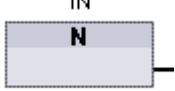
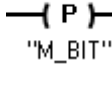
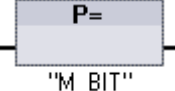
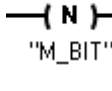
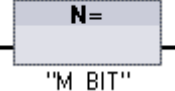
INOUT parametresi, set veya reset olan bit adresini belirtir. Opsiyonel Q çıktısı, "INOUT" adresinin sinyal durumunu takip eder.

Komut	S1	R	"INOUT" biti
RS	0	0	Önceki durum
	0	1	0
	1	0	1
	1	1	1
SR	<b>S</b>	<b>R1</b>	
	0	0	Önceki durum
	0	1	0
	1	0	1
	1	1	0



### 7.1.3 Pozitif Geçiş (Yükselen Kenar) ve Negatif Geçiş (Düşen Kenar) Komutları

Çizelge 7- 14 Pozitif ve negatif geçiş algılama


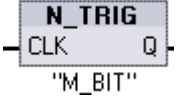
LAD	FBD	SCL	Açıklama
<p>"IN"</p>  <p>"M_BIT"</p>	<p>"IN"</p>  <p>"M_BIT"</p>	Yok <sup>1</sup>	<p>Pozitif işaret ayrıtı için operandı tarayın.</p> <p>LAD: Bu kontağın durumu, atanan "INPUT" biti üzerinde bir pozitif geçiş (OFF'dan ON'a) algılandığında TRUE'dur. Kontak lojik durumu bundan sonra, enerji akışı çıkış durumunu ayarlamak için enerji akışı giriş durumu ile birleştirilir. P kontağı, bir kol devresinin ucu haricinde ağdaki herhangi bir yerde olabilir.</p> <p>FBD: Çıkış lojik durumu, atanan giriş biti üzerinde bir pozitif geçiş (OFF'dan ON'a) algılandığında TRUE'dur. P kutusu sadece bir devre kolunun başlangıcında konumlandırılabilir.</p>
<p>"IN"</p>  <p>"M_BIT"</p>	<p>"IN"</p>  <p>"M_BIT"</p>	Yok <sup>1</sup>	<p>Negatif işaret ayrıtı için operandı tarayın.</p> <p>LAD: Bu kontağın durumu, atanan giriş biti üzerinde bir negatif geçiş (ON'dan OFF'a) algılandığında TRUE'dur. Kontak lojik durumu bundan sonra, enerji akışı çıkış durumunu ayarlamak için enerji akışı giriş durumu ile birleştirilir. N kontağı, bir kol devresinin ucu haricinde ağdaki herhangi bir yerde olabilir.</p> <p>FBD: Çıkış lojik durumu, atanan giriş biti üzerinde bir negatif geçiş (ON'dan OFF'a) algılandığında TRUE'dur. N kutusu sadece bir devre kolunun başlangıcında konumlandırılabilir.</p>
<p>"OUT"</p>  <p>"M_BIT"</p>	<p>"OUT"</p>  <p>"M_BIT"</p>	Yok <sup>1</sup>	<p>Pozitif işaret ayrıtı için operandı.</p> <p>LAD: Atanan "OUT" biti, bobine giren enerji akışı üzerinde bir pozitif geçiş (OFF'dan ON'a) algılandığında TRUE'dur. Enerji akışı giriş durumu, bobin üzerinden her zaman enerji akışı çıkış durumu gibi geçer. P bobini ağda herhangi bir yerde konumlandırılabilir.</p> <p>FBD: Atanan bit "OUT" kutu giriş bağlantısında ya da eğer kutu bir devre kolunun başlangıcında konumlandırılmışsa giriş bit atamasında bir pozitif geçiş (OFF'dan ON'a) algılandığında TRUE'dur. Giriş lojik durumu kutu üzerinden her zaman çıkış lojik durumu gibi geçer. P= bobini devre kolunda herhangi bir yerde konumlandırılabilir.</p>
<p>"OUT"</p>  <p>"M_BIT"</p>	<p>"OUT"</p>  <p>"M_BIT"</p>	Yok <sup>1</sup>	<p>Negatif işaret ayrıtı için operandı.</p> <p>LAD: Atanan "OUT" biti, bobine giren enerji akışı üzerinde bir negatif geçiş (ON'dan OFF'a) algılandığında TRUE'dur. Enerji akışı giriş durumu, bobin üzerinden her zaman enerji akışı çıkış durumu gibi geçer. N bobini ağda herhangi bir yerde konumlandırılabilir.</p> <p>FBD: Atanan bit "OUT" kutu giriş bağlantısında ya da eğer kutu bir devre kolunun başlangıcında konumlandırılmışsa giriş bit atamasında bir negatif geçiş (ON'dan OFF'a) algılandığında TRUE'dur. Giriş lojik durumu kutu üzerinden her zaman çıkış lojik durumu gibi geçer. N= bobini devre kolunda herhangi bir yerde konumlandırılabilir.</p>

<sup>1</sup> SCL için: Uygulamanızın içerisindeki bu fonksiyonun tekrarlanması için kod yazmalısınız.

## Temel komutlar



### 7.1 Bit lojik çalışmaları

Çizelge 7- 15 P\_TRIG ve N\_TRIG

LAD / FBD	SCL	Açıklama
	Yok <sup>1</sup>	<p>Pozitif işaret ayrıtı için RLO'u (mantık işleminin sonucu) tarayın CLK giriş durumu (FBD) veya CLK güç akışında (LAD) pozitif geçiş (OFF-to-ON) tespit edildiğinde Q çıkış güç akışı veya lojik durumu TRUE olur.</p> <p>LAD'de, P_TRIG komutu bir ağın başında ya da sonunda konumlandırılmaz. FBD'de, P_TRIG komutu bir kol devresinin sonu hariç her yerde konumlandırılabilir.</p>
	Yok <sup>1</sup>	<p>Negatif işaret ayrıtı için RLO'u tarayın Q çıkış enerji akışı ya da lojik durumu, CLK giriş durumunda (FBD) ya da CLK enerji akışında (LAD) bir negatif geçiş (ON'dan OFF'a) algılandığında TRUE'dur.</p> <p>LAD'de, N_TRIG komutu bir ağın başında ya da sonunda konumlandırılmaz. FBD'de, N_TRIG komutu bir kol devresinin sonu hariç her yerde konumlandırılabilir.</p>

<sup>1</sup> SCL için: Uygulamanızın içerisindeki bu fonksiyonun tekrarlanması için kod yazmalısınız.

Çizelge 7- 16 R\_TRIG ve F\_TRIG komutları

LAD / FBD	SCL	Açıklama
	<p>"R_TRIG_DB" (CLK:=_in_, Q=&gt; _out_</p>	<p>Pozitif sinyal kenarında etiket ayarı.</p> <p>Atanan instance DB, CLK girişinin önceki durumunu saklamak için kullanılır. CLK giriş durumu (FBD) veya (LAD)'daki güç akışında bir pozitif geçiş (OFF-ON) algılandığında, Q çıkış güç akışı veya lojik durumu TRUE olur.</p> <p>LAD'da, R_TRIG komutu bir ağın başında veya sonunda yer alamaz. FBD'de, R_TRIG komutu bir kolun (branşın) dışında herhangi bir yerde yer alabilir.</p>
	<p>"F_TRIG_DB" (CLK:=_in_, Q=&gt; _out_</p>	<p>Negatif sinyal kenarında etiket ayarı.</p> <p>Atanan instance DB, CLK girişinin önceki durumunu saklamak için kullanılır. CLK giriş durumu (FBD) veya (LAD)'daki güç akışında bir negatif geçiş (ON-OFF) algılandığında, Q çıkış güç akışı veya lojik durumu TRUE olur.</p> <p>LAD'da, F_TRIG komutu bir ağın başında veya sonunda yer alamaz. FBD'de, F_TRIG komutu bir kolun (branşın) dışında herhangi bir yerde yer alabilir.</p>

R\_TRIG ve F\_TRIG için, programa bir komut yerleştirdiğinizde, "Call options" diyalogu otomatik olarak açılır. Bu diyalogda kenar belleği bitinin kendi veri bloğunda (tek instance) veya lokal bir etiket olarak blok ara yüzünde olmasına göre atama yapabilirsiniz. Aynı bir veri bloğu oluşturursanız, onu proje ağacında "Program blocks > System blocks" altında "Program resources" klasöründe bulacaksınız.

Çizelge 7- 17 Parametreler için veri tipleri (P ve N kontakları/bobinleri, P=, N=, P\_TRIG ve N\_TRIG)

Parametre	Veri tipi	Açıklama
M_BIT	Bool	Girişin önceki durumunun kaydedildiği bellek biti
IN	Bool	Geçiş kenarı algılanacak giriş biti
OUT	Bool	Bir geçiş kenarının algılandığını belirten çıkış biti
CLK	Bool	Geçiş kenarı algılanacak enerji akışı ya da giriş biti
Q	Bool	Bir kenarın algılandığını belirten çıkış

Tüm kenar komutları, izlenen giriş sinyalinin önceki durumunu saklamak için bir bellek biti kullanır (M\_BIT: P/N kontakları/bobinleri, P\_TRIG/N\_TRIG ya da örneğin DB bit: R\_TRIG, F\_TRIG). Bir kenar girişinin durumu bellek bitinin önceki durumu ile karşılaştırılarak tespit edilir. Eğer durumla ilgili yönün girişinde bir değişim belirtirse, o zaman bir kenar çıkışın TRUE olarak yazılmasıyla raporlanır. Aksi halde çıkış FALSE olarak yazılacaktır.

#### Not

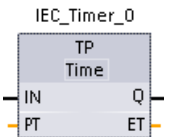
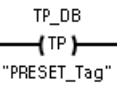
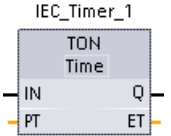
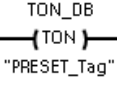
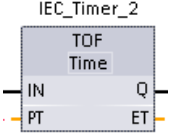
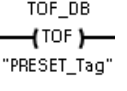
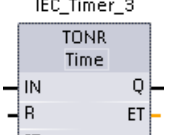
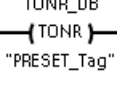

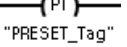
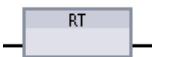
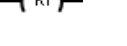
Kenar komutları, giriş ve bellek biti değerlerini her yürütüldüklerinde değerlendirir. Bu değerlendirmeye ilk yürütme de dâhildir. İlk tarama üzerinde kenar algılamasına izin verilmesi ya da verilmemesi konusunda program tasarımınızdaki bellek bitinin ve girişin ilk durumlarının sorumluluğu size aittir.

Bellek biti bir yürütmeden diğerine korunmak zorunda olduğundan, her bir kenar komutu için benzersiz bir bit kullanmalı ve bu bit programınızda herhangi başka bir yerde kullanılmamalıdır. Geçici hafıza ve bir I/O güncellemesi gibi diğer sistem fonksiyonları tarafından etkilenebilecek hafızalardan kaçınmalısınız. M\_BIT hafıza atamaları için sadece M, global DB veya Statik hafızayı (bir DB'sinde) kullanın.

## 7.2 Zamanlayıcı işlemleri

Programlanmış zaman gecikmeleri oluşturmak için zamanlayıcı komutları kullanılır. Kullanıcı programında kullanabileceğiniz zamanlayıcıların sayısı, sadece CPU'daki bellek miktarı ile sınırlıdır. Her bir zamanlayıcı, kutusunun ya da bobin komutunun üzerinde belirtilmiş olan, zamanlayıcı verilerini saklamak için 16 bayt IEC\_Timer veri tipi DB yapısını kullanır. Siz komutu eklediğinizde, STEP 7 DB'yi otomatik olarak oluşturur.

Çizelge 7- 18 Zamanlayıcı komutları

LAD / FBD kutuları	LAD bobinleri	SCL	Açıklama
	 "PRESET_Tag"	<pre>"IEC_Timer_0_DB".TP (   IN:= _bool_in_,   PT:= _time_in_,   Q=&gt; _bool_out_,   ET=&gt; _time_out_);</pre>	TP zamanlayıcısı, ön ayarlanan süre boyunca bir darbe üretir.
	 "PRESET_Tag"	<pre>"IEC_Timer_0_DB".TON (   IN:= _bool_in_,   PT:= _time_in_,   Q=&gt; _bool_out_,   ET=&gt; _time_out_);</pre>	TON zamanlayıcısı, ayarlanan süre gecikmesinden sonra Q çıkışı ON yapar.
	 "PRESET_Tag"	<pre>"IEC_Timer_0_DB".TOF (   IN:= _bool_in_,   PT:= _time_in_,   Q=&gt; _bool_out_,   ET=&gt; _time_out_);</pre>	TOF zamanlayıcısı, ayarlanan süre gecikmesinden sonra Q çıkışı OFF olarak resetler.
	 "PRESET_Tag"	<pre>"IEC_Timer_0_DB".TONR (   IN:= _bool_in_,   R:= _bool_in_,   PT:= _time_in_,   Q=&gt; _bool_out_,   ET=&gt; _time_out_);</pre>	TONR zamanlayıcısı, ayarlanan süre gecikmesinden sonra Q çıkışı ON olarak ayarlar. Geçen süre, R girişi geçen zamanı resetlemek için kullanılabilecek kadar geçen çoklu zamanlama periyotlarının toplamıdır.
FBD sadece: 	 "PRESET_Tag"	(SCL eşdeğeri değil)	PT (Ayarlanan zamanlayıcısı) bobini, yeni PRESET (ayarlanan) zaman değerini belirtilen IEC_Timers'ına yükler.
FBD sadece: 	 "PRESET_Tag"	(SCL eşdeğeri değil)	RT (Reset zamanlayıcısı) bobini belirtilen IEC_Timers'ı resetler.

<sup>1</sup> Siz komutu eklediğinizde, STEP 7 DB'yi otomatik olarak oluşturur.

<sup>2</sup> SCL örneklerinde, "IEC\_Timer\_0\_DB", DB olayının (instance DB) ismidir.

Çizelge 7- 19 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
Kutu: IN Bobin: Güç akışı	Bool	TP, TON ve TONR: Kutu (Box) 0=Zamanlayıcıyı devre dışı bırak, 1=Zamanlayıcıyı etkinleştir Bobin (Coil): Güç akışı yok= Zamanlayıcıyı devre dışı bırak, Güç akışı= Zamanlayıcıyı etkinleştir TOF: Kutu (Box): 0= Zamanlayıcıyı etkinleştir, 1= Zamanlayıcıyı devre dışı bırak Bobin (Coil): Güç akışı yok = Zamanlayıcıyı etkinleştir, Güç akışı= Zamanlayıcıyı devre dışı bırak
R	Bool	Sadece TONR Kutu (Box)'su: 0=resetleme yok 1= Reset'in aldığı süre ve 0 için Q bit'i
Kutu: PT Bobin: "PRESET_Tag"	Time	Zamanlayıcı Kutu (Box)'su veya Bobini (Coil): Önceden tanımlı zaman girişi
Kutu: Q Bobin: DBdata.Q	Bool	Zamanlayıcı Kutu (Box)'su: DB veri zamanlayıcısındaki Q kutu çıkışı veya Q bit'i Zamanlayıcı Bobini (Coil): Sadece DB veri zamanlayıcısındaki Q bit'i adresleyebilirsiniz.
Kutu: ET Bobin: DBdata.ET	Time	Zamanlayıcı Kutu (Box)'su: DB veri zamanlayıcısındaki ET (geçen zaman) kutu girişi veya ET zaman değeri Zamanlayıcı Bobini (Coil): Sadece DB veri zamanlayıcısındaki ET zaman değerini adresleyebilirsiniz.

Çizelge 7- 20 PT ve IN parametrelerindeki değer değişimlerinin etkisi

Zamanlayıcı	PT ve IN kutusu parametrelerindeki ve karşılık gelen bobin parametrelerindeki değişiklikler
TP	<ul style="list-style-type: none"> <li>PT'yi değiştirmek zamanlayıcı çalışırken herhangi bir etkiye neden olmaz.</li> <li>IN'i değiştirmek zamanlayıcı çalışırken herhangi bir etkiye neden olmaz.</li> </ul>
TON	<ul style="list-style-type: none"> <li>PT'yi değiştirmek zamanlayıcı çalışırken herhangi bir etkiye neden olmaz.</li> <li>Zamanlayıcı çalışırken IN'i FALSE olarak değiştirmek, zamanlayıcıyı resetler ve durdurur.</li> </ul>
TOF	<ul style="list-style-type: none"> <li>PT'yi değiştirmek zamanlayıcı çalışırken herhangi bir etkiye neden olmaz.</li> <li>Zamanlayıcı çalışırken IN'i TRUE olarak değiştirmek, zamanlayıcıyı resetler ve durdurur.</li> </ul>
TONR	<ul style="list-style-type: none"> <li>PT'yi zamanlayıcı çalışırken değiştirmenin hiçbir etkisi yoktur ancak zamanlayıcı tekrar çalıştığında bir etkisi vardır.</li> <li>IN'i zamanlayıcı çalışırken FALSE olarak değiştirmek, zamanlayıcıyı durdurur ancak zamanlayıcıyı resetlemez. IN'i tekrar TRUE olarak değiştirmek, zamanlayıcının zamanlamayı toplanan zaman değerinden başlatmasına neden olur.</li> </ul>

PT (ayarlanan zaman) ve ET (geçen zaman) değerleri, belirtilen IEC\_TIMER DB verilerinde saklanır ve zamanı milisaniye olarak veren çift tamsayılar olarak işaretlenir. TIME verileri T# tanımlayıcısını kullanır ve tek bir zaman birimi (T#200ms veya 200) ve T#2s\_200ms gibi bileşik zaman birimleri olarak girilebilir.

## Temel komutlar

### 7.2 Zamanlayıcı işlemleri

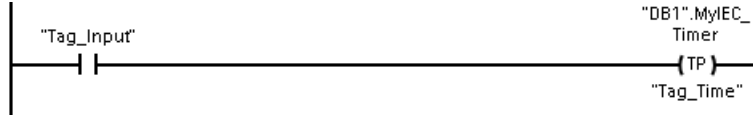
Çizelge 7- 21 TIME veri tipinin boyutu ve aralığı

Veri tipi	Boyut	Geçerli sayı aralıkları <sup>1</sup>
TIME	32 bit, DInt verileri olarak saklanır	T#-24d_20h_31m_23s_648ms, T#24d_20h_31m_23s_647ms arasında -2,147,483,648 ms, +2,147,483,647 ms arasında saklanır

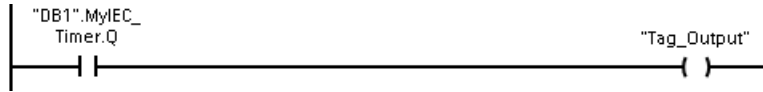
<sup>1</sup> TIME veri tipinin yukarıda gösterilen negatif aralığı, zamanlayıcı komutlarıyla birlikte kullanılamaz. Negatif PT (geçerli zaman) değerleri, zamanlayıcı komutu yürütüldüğünde sıfıra ayarlanır. ET (geçen zaman) her zaman pozitif bir değerdir.

### Zamanlayıcı bobini örneği

-(TP)-, -(TON)-, -(TOF)- ve -(TONR)- zamanlayıcı bobinleri bir LAD ağındaki son komut olmalıdır. Zamanlayıcı örneğinde de gösterildiği gibi, müteakip bir ağıdaki kontak komutu, bir zamanlayıcı bobininin IEC\_Zamanlayıcı DB verilerindeki Q bitini değerlendirir. Aynı şekilde, eğer programınızda geçen zaman değerini kullanmak istiyorsanız IEC\_Timer DB verilerindeki ELAPSED (geçen) elemanını da adreslemelisiniz.



Darbe zamanlayıcısı, Tag\_Input bit değerinin bir 0'dan 1'e geçişi üzerinde başlar. Zamanlayıcı, Tag\_Time zaman değeri tarafından belirlenen süre boyunca çalışır.



Zamanlayıcı çalıştığı sürece DB1.MyIEC\_Timer.Q=1 ve Tag\_Output value=1'dir. Tag\_Time değeri geçtiğinde, DB1.MyIEC\_Timer.Q=0 ve Tag\_Output value=0 olur.

### Reset zamanlayıcısı -(RT)- ve Preset (Ayarlanan) -(PT)- bobinleri

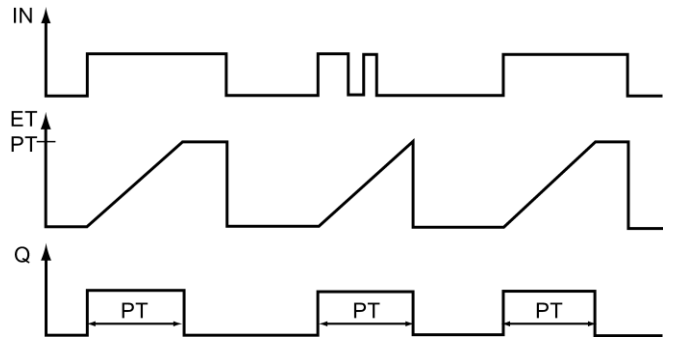
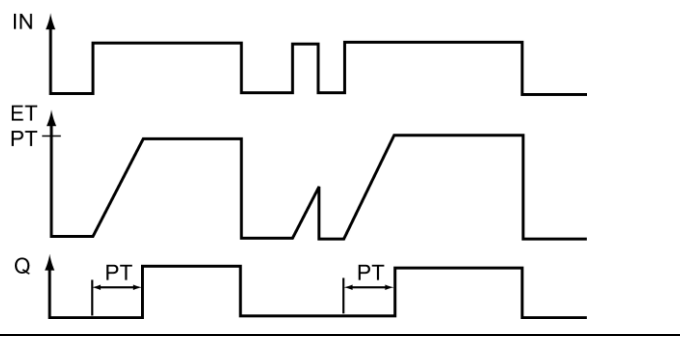
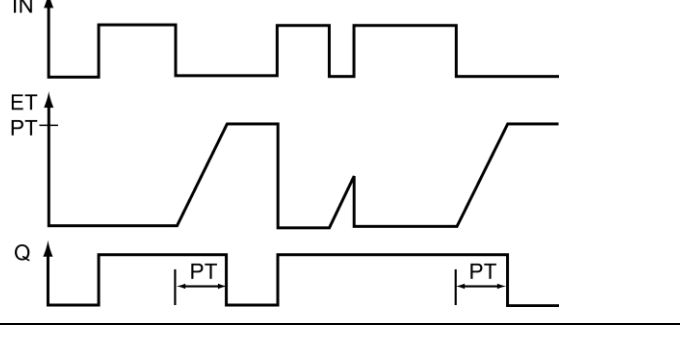
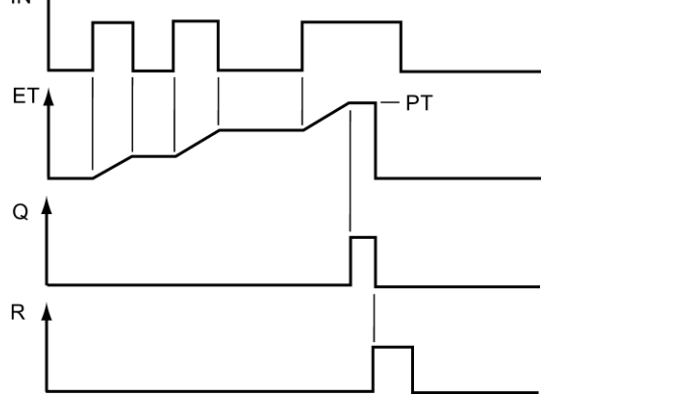
Bu bobin komutları, kutu veya bobin zamanlayıcıları ile kullanılabilir ve orta hat konumuna yerleştirilebilir. Bobin çıkış gücü akış durumu her zaman bobin giriş durumu ile aynıdır. (RT)-bobini etkinleştirildiğinde, belirtilen IEC\_Timer DB verilerinin ELAPSED (geçen) zaman elemanı 0'a resetlenir. -(PT)- bobini etkinleştirildiğinde, belirtilen IEC\_Timer DB verilerinin PRESET (ön ayarlanan) zaman elemanı 0'a resetlenir.

#### Not

Zamanlayıcı komutlarını bir FB'ye yerleştirirken, "Çoklu instance veri bloğu" seçeneğini seçebilirsiniz. Zamanlayıcı yapısı, ayrı veri yapıları ile farklı olabilir ancak zamanlayıcı verileri tek bir veri bloğunda yer alır ve her bir zamanlayıcı için ayrı bir veri bloğu gerektirmez. Bu işlem süresini ve zamanlayıcıların işlenmesi için gereken veri depolama alanını azaltır. Paylaşılan çoklu instance DB'sindeki zamanlayıcı veri yapıları arasında hiçbir etkileşim yoktur.

## Zamanlayıcıların çalışması

Çizelge 7- 22 IEC zamanlayıcıların tipleri

Zamanlayıcı	Zamanlayıcı şeması
<b>TP:</b> Darbe oluşturucu TP zamanlayıcısı, ayarlanan genişlik zamanı ile bir darbe üretir.	
<b>TON:</b> ON-gecikme oluşturucu TON zamanlayıcısı, ayarlanan bir zaman gecikmesinden sonra Q çıkışını ON olarak ayarlar.	
<b>TOF:</b> OFF-gecikme oluşturucu TOF zamanlayıcısı ayarlanan bir zaman gecikmesinden sonra Q çıkışını OFF olarak resetler.	
<b>TONR:</b> Zaman biriktirici TONR zamanlayıcısı, ayarlanan bir zaman gecikmesinden sonra Q çıkışını ON olarak ayarlar. Geçen süre, çoklu zamanlama periyotları üzerinden R girişi geçen zamanı resetlemek için kullanılına kadar geçen süredir.	

**Not**

CPU'da, özel herhangi bir zamanlayıcı komutuna hiçbir özel kaynak tahsisi yapılmaz. Bunun yerine her bir zamanlayıcı, zamanlamayı gerçekleştirmek için kendi zamanlayıcı yapısını ve sürekli çalışan dâhili CPU zamanlayıcısını kullanır.

Bir TP, TON, FOF veya TONR komutunun girişi üzerindeki bir kenar değişimine bağlı olarak bir zamanlayıcı başlatıldığında, sürekli çalışan CPU zamanlayıcısının değeri bu zamanlayıcı komutu için tahsis edilen DB yapısının START üyesine kopyalanır. Bu başlangıç değeri, zamanlayıcı çalışmaya devam ederken değişmeden kalır ve daha sonra zamanlayıcının her güncellenişinde kullanılır. Zamanlayıcının her başlayışında, zamanlayıcı yapısına dâhili CPU zamanlayıcısından yeni bir başlangıç değeri yüklenir.

Bir zamanlayıcı güncellendiğinde, yukarıda açıklanan başlangıç değeri geçen zamanı belirlemek için dâhili CPU zamanlayıcısının geçerli değerinden çıkarılır. ELAPSED (geçen) ve Q üyeleri bundan sonra, bu zamanlayıcı için tahsis edilen DB yapısında güncellenir. Geçen süre bundan sonra, zamanlayıcı Q bitinin durumunu belirlemek için ayarlanan süre ile karşılaştırılır. ELAPSED (geçen) ve Q üyeleri bundan sonra, bu zamanlayıcı için tahsis edilen DB yapısında güncellenir. Geçen sürenin ayarlanan süre değerine kenetlendiğini unutmayın (ayarlanan süreye ulaşıldıktan sonra zamanlayıcı geçen süreyi biriktirmeye devam etmez).

Bir zamanlayıcı güncellemesi, sadece ve sadece şu zamanlarda gerçekleştirilir:

- Bir zamanlayıcı komutu (TP, TON, TOF veya TONR) yürütüldüğünde
- Zamanlayıcı yapısının DB'deki "ELAPSED" (geçen) üyesine bir komut tarafından direkt olarak başvurulduğunda kullanıldığında
- Zamanlayıcı yapısının DB'deki "Q" üyesine bir komut tarafından direkt olarak başvurulduğunda kullanıldığında



## Zamanlayıcı programlaması

Kullanıcı programınızı planlarken ve oluştururken, zamanlayıcı işleyişinin şu sonuçları dikkate alınmalıdır:

- Aynı aramada bir zamanlayıcının çeşitli güncellemelerine sahip olabilirsiniz. Zamanlayıcı, zamanlayıcı komutunun (TP, TON, TOF, TONR) her yürütülüşünde ve zamanlayıcı yapısının ELAPSED veya Q üyesinin yürütülen başka bir komutun bir parametresi olarak her kullanılışında güncellenir. Bu, eğer en son zaman verilerini (özellikle zamanlayıcının anında okumasını) istiyorsanız avantajlıdır. Ancak eğer bir program taraması boyunca tutarlı değerler istiyorsanız, o halde zamanlayıcı komutunuzu bu değerlere ihtiyaç duyan tüm diğer komutların öncesine yerleştiriniz ve zamanlayıcı DB yapısının ELAPSED ve Q üyeleri yerine zamanlayıcı komutundan Q ve ET çıkışlarından etiketleri kullanın.
- Hiçbir zamanlayıcının güncellemesi gerçekleşmiyorken taramalar yapabilirsiniz. Zamanlayıcınızı bir fonksiyon içinde başlatmanız ve sonra bir veya daha fazla tarama için bu fonksiyonu çağırılmayı sona erdirmeniz mümkündür. Eğer zamanlayıcı yapısının ELAPSED ya da Q üyelerine başvuran başka hiçbir komut yürütülüyorsa, o zaman zamanlayıcı da güncellenmez. Zamanlayıcı komutu tekrar yürütülene ya da zamanlayıcı yapısından bir parametre olarak ELAPSED ya da Q kullanılarak başka bir komut yürütülene kadar yeni bir güncelleme gerçekleşmez.
- Genelde kullanılmasa da aynı DB zamanlayıcı yapısını birden fazla zamanlayıcı komutuna atayabilirsiniz. Genel olarak, beklenmedik bir etkileşimden kaçınmak için, her bir DB zamanlayıcı yapısı başına bir zamanlayıcı komutu (TP, TON, TOF, TONR) kullanmalısınız.
- Kendi kendini resetleyen zamanlayıcılar, periyodik olarak meydana gelmesi gereken eylemlerin tetiklenmesinde kullanışlıdır. Tipik olarak, kendi kendini resetleyen zamanlayıcılar zamanlayıcı bitine başvuran normal olarak kapalı bir kontağı zamanlayıcı komutunun önüne yerleştirmek suretiyle oluşturulur. Zamanlayıcı ağı tipik olarak eylemleri tetiklemek için zamanlayıcı bitini kullanan bir veya birden fazla bağımlı ağı üzerine konumlandırılır. Zamanlayıcının süresi dolduğunda (geçen zaman ayarlanan zamana ulaştığında), bir tarama için zamanlayıcı biti açıktır ve bu, zamanlayıcı biti tarafından kontrol edilen bağımlı ağ lojisinin yürütülmesine izin verir. Zamanlayıcı ağının bir sonraki yürütülmesinde, normal olarak kapalı olan kontak kapalıdır, böylece zamanlayıcıyı resetler ve zamanlayıcı bitini temizler. Bir sonraki taramada normal olarak kapalı kontak açıktır, böylece zamanlayıcıyı resetler. Bunun gibi kendi kendini resetleyen zamanlayıcılar oluştururken zamanlayıcı DB yapısının "Q" üyesini, zamanlayıcı komutunun önünde normal olarak kapalı kontak için parametre olarak kullanmayın. Bunun yerine bu amaçla zamanlayıcı komutunun "Q" çıkışına bağlı etiketi kullanın. Zamanlayıcı DB yapısının Q üyesine erişimden kaçınılmasının sebebi bu erişimin zamanlayıcıda bir güncellemeye neden olmasıdır. Ayrıca eğer zamanlayıcı normal olarak kapağı kontağa bağlı olarak güncellenirse, o zaman kontak zamanlayıcı komutunu derhal resetler. Zamanlayıcı komutunun Q çıkışı bir tarama için açık olmayacak ve bağımlı ağlar yürütülmeyecektir.

## Bir RUN-STOP-RUN geçişinden ya da bir CPU enerji döngüsünden sonra zaman verilerinin tutulması

Eğer bir RUN modu oturumu STOP modu ile ya da bir CPU enerji döngüsü ile sonlandırılırsa ve yeni bir RUN modu oturumu başlatılırsa, zamanlayıcı veri yapısı kalıcı olarak ayarlanmadığı sürece (TP, TON, TOF ve TONR zamanlayıcıları), önceki RUN modu oturumunda saklanan zamanlayıcı verileri kaybolur.

Program editörüne bir zamanlayıcı komutu yerleştirdikten sonra çağrı seçenekleri diyalogundaki varsayılanları kabul ettiğinizde, kalıcı yapılamayacak bir instance DB'sine atanmış olursunuz. Zamanlayıcı verilerini kalıcı yapmak için bir global DB ya da bir çoklu instance DB'si kullanmanız gerekir.

### Zamanlayıcı verilerini kalıcı veriler olarak saklamak için bir global DB atayın

Bu seçenek, zamanlayıcının nerede kullanıldığından (OB, FC veya FB) bağımsız olarak çalışır.

1. Bir global DB oluşturma:
  - Proje ağacından “Yeni blok ekle” üzerine çift tıklayınız
  - Data bloğu (DB) simgesine tıklayınız
  - Tip için global DB'yi seçiniz
  - Eğer bu DB'deki veri elemanlarından bazılarını kalıcı olarak seçebilmek istiyorsanız, DB tipi “Optimize” kutusunun işaretlendiğinden emin olun. Diğer DB tipi seçeneği “Standart S7-300/400 ile uyumlu”, tüm DB veri elemanlarının kalıcı ya da kalıcı olmayan şekilde ayarlanmasına olanak tanır.
  - TAMAM'a tıklayınız
2. DB'ye zamanlayıcı yapısı/yapıları ekleme:
  - Global DB'de, IEC\_Timer veri tipini kullanarak yeni bir statik etiket ekleyiniz.
  - “Tut” sütunundaki kutucuğu işaretleyin, böylece bu yapı kalıcı olacaktır.
  - Bu DB'de saklamak istediğiniz tüm zamanlayıcılar için yapılar oluşturmak için bu işlemi tekrarlayınız. Her zamanlayıcı yapısını benzersiz bir global DB'ye yerleştirebilir ya da çoklu zamanlayıcı yapılarını aynı global DB'ye yerleştirebilirsiniz. Zamanlayıcılar dışında diğer statik etiketleri de bu global DB'ye yerleştirebilirsiniz. Aynı global DB'ye birden fazla zamanlayıcı yapısı yerleştirmek, genel blok sayınızı azaltmanıza yarar.
  - İsterseniz zamanlayıcı yapılarının isimlerini değiştirebilirsiniz.
3. Bir kalıcı zamanlayıcı (OB, FC veya FB) yerleştirmek istediğiniz yerde düzenleme için program bloğunu açınız.
4. Zamanlayıcı komutunu istediğiniz konuma yerleştiriniz.
5. Çağrı diyalogu belirlediğinizde iptal düğmesine basınız.
6. Yeni zamanlayıcı komutunun üstüne, global DB'nin ismini (tarama için yardımcı kullanmayın) ve yukarıda oluşturmuş olduğunuz zamanlayıcı yapısını giriniz (örnek: "Data\_block\_3.Static\_1").

## Zamanlayıcı verilerini kalıcı veriler olarak saklamak için bir çoklu instance DB'si atayın

Bu seçenek ancak zamanlayıcıyı bir FB içine yerleştirirseniz çalışır.

Bu seçenek FB'nin "Optimize" blok erişimi ile yapılandırılıp yapılandırılmadığına bağlıdır (sadece sembolik erişime izin verir). Erişim davranışının mevcut bir FB için nasıl yapılandırıldığını doğrulamak için, Proje ağacından FB üzerine sağ tıklayınız, özellikleri seçiniz ve sonra nitelikleri seçiniz.

Eğer FB "Optimize" kutusu işaretli bir şekilde oluşturulmuşsa (sadece sembolik erişime izin verir):

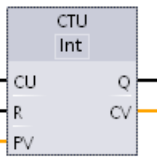
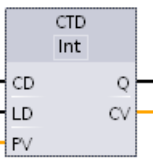
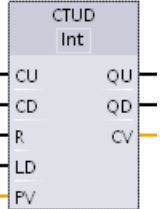
1. Düzenleme için FB'yi açınız.
2. Zamanlayıcı komutunu FB içinde istediğiniz konuma yerleştiriniz.
3. Çağrı seçenekleri diyalogu belirlediğinde Çoklu instance simgesine tıklayınız. Çoklu instance seçeneği ancak eğer komut bir FB içine yerleştirilmişse vardır.
4. Eğer isterseniz çağrı seçenekleri diyalogundan zamanlayıcının ismini değiştirebilirsiniz.
5. TAMAM'a tıklayınız. Editörde zamanlayıcı komutu görüntülenir ve IEC\_TIMER yapısı Statik altındaki FB ara yüzünde belirir.
6. Eğer gerekirse FB ara yüz editörünü açın (görünümü genişletmek için küçük oka tıklanması gerekebilir).
7. Statik altında, sizin için oluşturulmuş olan zamanlayıcı yapısını konumlandırınız.
8. Bu zamanlayıcı yapısı için olan Tut sütunundan, seçimi "Tut" olarak değiştirin. Bu FB bundan sonra başka bir program bloğundan ne zaman çağrılırsa, kalıcı olarak işaretlenmiş zamanlayıcı yapısını içeren bu ara yüz tanımlaması ile bir instance DB'si oluşturulacaktır. Bu FB bundan sonra başka bir program bloğundan ne zaman çağrılırsa, kalıcı olarak işaretlenmiş zamanlayıcı yapısını içeren bu ara yüz tanımlaması ile bir instance DB'si oluşturulacaktır.

Eğer FB optimize blok erişimi olarak atanmadığında, blok erişimi klasik S7-300/400 yapılandırması ile uyumlu ve sembolik ve direk erişimi standart olarak sağlamalıdır. Standart bir blok erişim olan FB için çoklu instance atamak için şu adımları izleyiniz:

1. Düzenleme için FB'yi açınız.
2. Zamanlayıcı komutunu FB içinde istediğiniz konuma yerleştiriniz.
3. Çağrı seçenekleri diyalogu belirlediğinde çoklu instance simgesine tıklayınız. Çoklu instance seçeneği ancak eğer komut bir FB içine yerleştirilmişse vardır.
4. Eğer isterseniz çağrı seçenekleri diyalogundan zamanlayıcının ismini değiştirebilirsiniz.
5. TAMAM'a tıklayınız. Editörde zamanlayıcı komutu görüntülenir ve IEC\_TIMER yapısı Statik altındaki FB ara yüzünde belirir.
6. Bu FB'yi kullanacak bloğu açınız.
7. Bu FB'yi istediğiniz konuma yerleştiriniz. Böylece bu FB için bir instance data bloğunun oluşturulması tamamlanır.
8. FB'yi editöre yerleştirdiğinizde oluşturduğunuz instance data bloğunu açınız.
9. Statik altından, ilgili zamanlayıcı yapısını konumlandırın. Bu yapıyı kalıcı yapmak için bu zamanlayıcı için olan Tut sütunundaki kutucuğu işaretleyiniz.

## 7.3 Sayıcı İşlemleri

Çizelge 7- 23 Sayıcı komutları

LAD / FBD	SCL	Açıklama
<p>"Counter name"</p> 	<pre>"IEC_Counter_0_DB".CTU (     CU:=_bool_in,     R:=_bool_in,     PV:=_int_in,     Q=&gt;_bool_out,     CV=&gt;_int_out);</pre>	<p>İç program olaylarını ve dış işlem olaylarını saymak için sayıcı komutlarını kullanın. Her sayıcı, sayıcı, verilerini tutmak için bir data bloğunda saklanan bir yapıyı kullanır. Bir sayıcı komutu editöre yerleştirildiğinde data bloğunu atamış olursunuz.</p> <ul style="list-style-type: none"> <li>• CTU yukarı sayıcı</li> <li>• CTD aşağı sayıcı</li> <li>• CTUD aşağı/yukarı sayıcı</li> </ul>
<p>"Counter name"</p> 	<pre>"IEC_Counter_0_DB".CTD (     CD:=_bool_in,     LD:=_bool_in,     PV:=_int_in,     Q=&gt;_bool_out,     CV=&gt;_int_out);</pre>	
<p>"Counter name"</p> 	<pre>"IEC_Counter_0_DB".CTUD D(     CU:=_bool_in,     CD:=_bool_in,     R:=_bool_in,     LD:=_bool_in,     PV:=_int_in,     QU=&gt;_bool_out,     QD=&gt;_bool_out,     CV=&gt;_int_out);</pre>	

- 1 LAD ve FBD için: Komut adının altındaki açılır menü listesinden sayıcı değeri veri tipini seçiniz.
- 2 Siz komutu eklediğinizde, STEP 7 DB'yi otomatik olarak oluşturur.
- 3 SCL örneğinde, DB olayının ismi "IEC\_Counter\_0\_DB"dir.

Çizelge 7- 24 Parametreler için veri tipleri

Parametre	Veri Tipi <sup>1</sup>	Açıklama
CU, CD	Bool	Bir sayım kadar yukarı veya aşağı sayım
R (CTU, CTUD)	Bool	Sayım değerini sıfıra resetlemek
LD (CTD, CTUD)	Bool	Ayarlanan değer için yükleme kontrolü
PV	SInt, Int, DInt, USInt, UInt, UDIInt	Sayım değeri için ayarlanan değer
Q, QU	Bool	Eğer CV >= PV ise True
QD	Bool	Eğer CV <= 0 ise True
CV	SInt, Int, DInt, USInt, UInt, UDIInt	Geçerli sayım değeri

- 1 Sayım değerlerinin nümerik aralığı, seçtiğiniz veri tipine bağlıdır. Eğer sayım değeri işaretli tamsayı tipinde ise, sıfıra kadar aşağı ya da aralık sınırına kadar ileri sayım yapabilirsiniz. Eğer sayım değeri işaretli bir tamsayı ise, negatif tamsayı sınırına kadar aşağı ve pozitif tamsayı limitine kadar da ileri sayım yapabilirsiniz.

Kullanıcı programında kullanabileceğiniz sayıcıların sayısı, sadece CPU'daki bellek miktarı ile sınırlıdır. Sayıcılar, şu bellek miktarlarını kullanırlar:

- SInt ve USInt veri tipleri için sayıcı komutu 3 bayt kullanır.
- Int ve UInt veri tipleri için sayıcı komutu 6 bayt kullanır.
- DInt ve UDInt veri tipleri için sayıcı komutu 12 bayt kullanır.

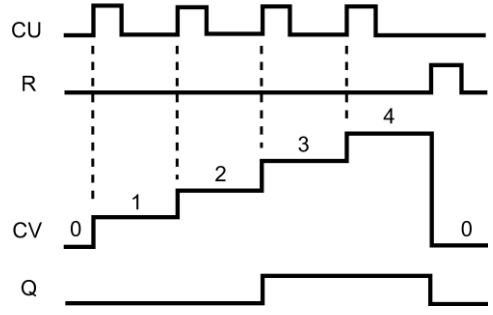
Bu komutlar, maksimum sayım hızları yerleştirildikleri OB'nin yürütme hızıyla sınırlandırılmış yazılım sayıcılarını kullanırlar. Komutların yerleştirildiği OB, CU veya CD girişlerinin tüm geçişlerini algılamaya yetecek kadar sık yürütülmelidir. Daha hızlı sayım işlemleri için CTRL\_HSC komutuna bakınız (Sayfa 3).

### Not

Bir FB'ye sayıcı komutları yerleştirdiğinizde, çoklu instance DB'si seçeneğini seçebilirsiniz, sayıcı yapısı isimleri ayrı veri yapıları ile farklı olabilir, ancak bir tekli DB içindeki sayıcı verileri her sayıcı için ayrı bir DB gerektirmez. Bu işlem süresini ve sayıcılar için gereken veri depolama alanını azaltır. Paylaşılan çoklu instance DB'sindeki sayıcı veri yapıları arasında hiçbir etkileşim yoktur.

## Sayıcıların çalışması

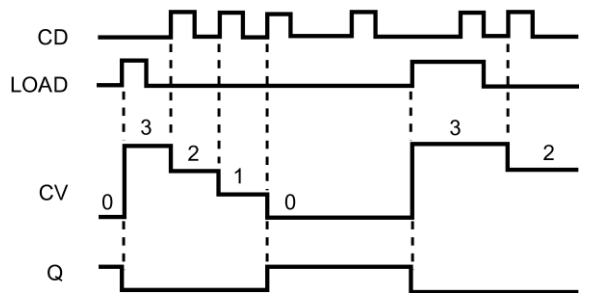
Çizelge 7- 25 CTU (yukarı sayıcının) sayıcısının çalışması

Sayıcı	Çalışma
CTU sayıcısı, CU parametresinin değeri 0'dan 1'e değiştiğinde 1 yukarı sayar. CTU zamanlama diyagramı, işaretsiz bir tamsayı sayım değeri için çalışmayı göstermektedir (PV = 3 iken). <ul style="list-style-type: none"> <li>• Eğer CV (geçerli sayım değeri) parametresinin değeri PV (ön ayarlanan sayım değeri) parametresinin değerinden büyükse ya da buna eşitse, o zaman sayıcı çıkış parametresi Q = 1'dir.</li> <li>• Eğer reset parametresi R'nin değeri 0'dan 1'e değişirse, o zaman geçerli sayım değeri de 0'a resetlenir.</li> </ul>	

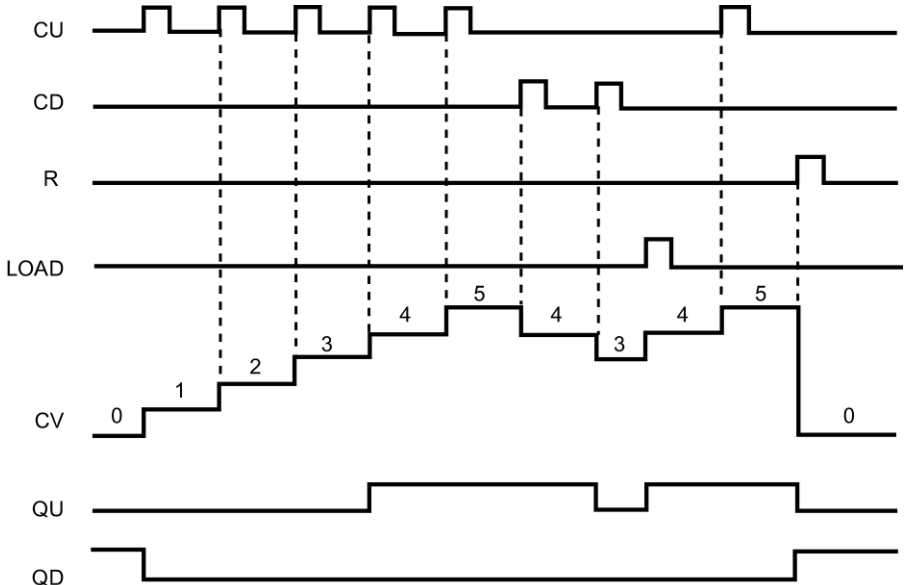
## Temel komutlar

### 7.3 Sayıcı İşlemleri

Çizelge 7- 26 CTD sayıcısının çalışması (aşağıya sayıcı)

Sayıcı	Çalışma
<p>CTD sayıcısı, CD parametresinin değeri 0'dan 1'e değiştiğinde 1 aşağı sayar. CTD zamanlama diyagramı, işaretli bir tamsayı sayım değeri için çalışmayı göstermektedir (PV = 3 iken).</p> <ul style="list-style-type: none"> <li>Eğer CV (geçerli sayım değeri) parametresinin değeri 0'a eşit ya da 0'ın altında ise, sayıcı çıkış parametresi Q = 1'dir.</li> <li>Eğer LOAD parametresinin değeri 0'dan 1'e değişirse, PV (ayarlanan sayım değeri) parametresindeki değer yeni CV (geçerli sayım değeri) olarak sayıcıya yüklenir.</li> </ul>	

Çizelge 7- 27 CTUD sayıcısının çalışması (aşağı/yukarı sayıcı)

Sayıcı	Çalışma
<p>CTUD sayıcısı, yukarı sayım (CU) ya da aşağı sayım (CD) girişlerinin 1'den 0'a geçişlerinde 1 birim yukarı veya aşağı sayar. CTUD zamanlama diyagramı, işaretli bir tamsayı sayım değeri için çalışmayı göstermektedir (PV = 4 iken).</p> <ul style="list-style-type: none"> <li>Eğer CV parametresinin değeri PV parametresinin değerinden büyükse ya da buna eşitse, o zaman sayıcı çıkış parametresi QU = 1'dir.</li> <li>Eğer CV parametresinin değeri sıfırdan küçük ya da sıfıra eşitse, o zaman sayıcı çıkış parametresi QD = 1'dir.</li> <li>Eğer LOAD parametresinin değeri 0'dan 1'e değişirse, o zaman PV parametresindeki değer yeni CV olarak sayıcıya yüklenir.</li> <li>Eğer reset parametresi R'nin değeri 0'dan 1'e değişirse, geçerli sayım değeri de 0'a resetlenir.</li> </ul>	

## Bir RUN-STOP-RUN geçişinden ya da bir CPU enerji döngüsünden sonra sayıcı verilerinin tutulması

Eğer bir RUN modu oturumu STOP modu ile ya da bir CPU enerji döngüsü ile sonlandırılırsa ve yeni bir RUN modu oturumu başlatılırsa, sayıcı veri yapısı kalıcı olarak belirlenmediği sürece (CTU, CTD ve CTUD sayıcıları) önceki RUN modu oturumunda saklanan sayıcı verileri kaybolur.

Program editörüne bir sayıcı komutu yerleştirdikten sonra çağrı seçenekleri diyalogundaki varsayılanları kabul ettiğinizde, kalıcı yapılamayacak bir instance DB'sine atanmış olursunuz. Sayıcı verilerini kalıcı yapmak için bir global DB ya da bir çoklu instance DB'si kullanmanız gerekir.

## Zamanlayıcı verilerini kalıcı veriler olarak saklamak için bir global DB atayın

Bu seçenek, sayıcının nerede kullanıldığından (OB, FC veya FB) bağımsız olarak çalışır.

1. Bir global DB oluşturunuz:
  - Proje ağacından “Yeni blok ekle” üzerine çift tıklayınız.
  - Data bloğu (DB) simgesine tıklayınız
  - Tip için global DB'yi seçiniz
  - Eğer bu DB'deki veri öğelerinden bazılarını kalıcı olarak seçebilmek istiyorsanız, sadece sembolik erişim kutusunun işaretlendiğinden emin olunuz.
  - Tamam'a tıklayınız.
2. DB'ye sayıcı yapısı/yapıları ekleyiniz:
  - Global DB'de, sayıcı veri tiplerinden birini kullanarak yeni bir statik etiket ekleyiniz. Ayarlanan ve Sayım değerleriniz için kullanmak istediğiniz Tipe dikkat ediniz.
  - Tut" sütunundaki kutucuğu işaretleyin, böylece bu yapı kalıcı olacaktır.
  - Bu DB'de saklamak istediğiniz tüm sayıcılar için yapılar oluşturmak için bu işlemi tekrarlayın. Her sayıcı yapısını benzersiz bir global DB'ye yerleştirebilir ya da çoklu sayıcı yapılarını aynı global DB'ye yerleştirebilirsiniz. Sayıcılar dışında diğer statik etiketleri de bu global DB'ye yerleştirebilirsiniz. Aynı global DB'ye birden fazla sayıcı yapısı yerleştirmek, genel blok sayınızı azaltmanıza yarar.
  - İsterseniz sayıcı yapılarının isimlerini değiştirebilirsiniz.
3. Bir kalıcı sayıcı (OB, FC veya FB) yerleştirmek istediğiniz yerde düzenleme için program bloğunu açın.
4. Sayıcı komutunu istediğiniz konuma yerleştiriniz.
5. Çağrı diyalogu belirlediğinde iptal düğmesine basın. Komut adının hemen üstünde ve hemen altında "???" ibaresine sahip yeni bir sayıcı komutu görmelisiniz.
6. Yeni sayıcı komutunun üstüne, global DB'nin ismini (tarama için yardımcıyı kullanmayın) ve yukarıda oluşturmuş olduğunuz sayıcı yapısını girin (örnek: "Data\_block\_3.Static\_1").Bu karşılık gelen ayarlanan ve sayım değeri tipinin doldurulmasına neden olur (örnek: Bir IEC\_UCounter yapısı için UInt).

Sayıcı Veri Tipi	Ayarlanan ve Sayım Değerleri için Karşılık Gelen Tip
IEC_Counter	INT
IEC_SCounter	SINT
IEC_DCounter	DINT
IEC_UCounter	UINT
IEC_USCounter	USINT
IEC_UDCounter	UDINT

### Sayıcı verilerini kalıcı veriler olarak saklamak için bir çoklu instance DB'si atayın

Bu seçenek ancak sayıcıyı bir FB içine yerleştirirseniz çalışır.

Bu seçenek, FB'nin sadece sembolik erişim olarak oluşturulup oluşturulmadığına bağlıdır (sadece sembolik erişime izin verilmesi). Bu erişimin niteliğinin mevcut bir FB için nasıl yapılandırıldığını görmek için, Proje ağacından FB üzerine sağ tıklayınız, özellikleri seçiniz ve sonra nitelikleri seçiniz.

Eğer FB'yi "Sadece sembolik erişim" atanmışsa:

1. Düzenleme için FB'yi açın.
2. Sayıcı komutunu FB içinde istediğiniz konuma yerleştiriniz.
3. Çağrı seçenekleri diyalogu belirlediğinde Çoklu instance simgesine tıklayınız. Çoklu oluş seçeneği ancak eğer komut bir FB içine yerleştirilmişse vardır.
4. Eğer isterseniz çağrı seçenekleri diyalogundan sayıcının ismini değiştirebilirsiniz.
5. Tamam'a tıklayınız. Ayarlanan ve sayım değerleri için sayıcı komutu editörde INT tipi ile belirir ve IEC\_COUNTER yapısı Statik altındaki FB Ara yüzünde belirir.
6. İsterseniz INT'den sayıcı komutundaki tipi diğer tiplerden birine değiştirebilirsiniz. Sayıcı yapısı buna göre değişecektir.
7. Eğer gerekirse FB ara yüz editörünü açın (görünümü genişletmek için küçük oka tıklanması gerekebilir).
8. Statik altında, sizin için oluşturulmuş olan sayıcı yapısını konumlandırın.
9. Bu sayıcı yapısı için olan Tut sütunundan, seçimi "Tut" olarak değiştirin. Bu FB bundan sonra başka bir program bloğundan ne zaman çağrılırsa, kalıcı olarak işaretlenmiş sayıcı yapısını içeren bu ara yüz tanımlaması ile bir instance DB'si oluşturulacaktır.

FB'nin "optimize edilmiş blok değerlendirmesi"ni belirtmemesi halinde, o zaman blok erişim tipi, S7-300/400 klasik konfigürasyonlarla uyumlu ve sembolik ve doğrudan erişimlere imkân verecek şekilde standarttır. Standard blok erişimi FB'ye çoklu bir olay (instance) atamak için aşağıdaki adımları takip ediniz.

1. Düzenleme için FB'yi açın.
2. Sayıcı komutunu FB içinde istediğiniz konuma yerleştiriniz.
3. Çağrı seçenekleri diyalogu belirlediğinde çoklu instance simgesine tıklayınız. Çoklu instance seçeneği ancak eğer komut bir FB içine yerleştirilmişse vardır.
4. Eğer isterseniz çağrı seçenekleri diyalogundan sayıcının ismini değiştirebilirsiniz.



5. Tamam'a tıklayınız. Ayarlanan ve sayım değeri için sayıcı komutu editörde INT tipi ile belirir ve IEC\_COUNTER yapısı Statik altındaki FB Ara yüzünde belirir.
6. İsterseniz INT'den sayıcı komutundaki tipi diğer tiplerden birine değiştirebilirsiniz. Sayıcı yapısı buna göre değişecektir.
7. Bu FB'yi kullanacak bloğu açınız.
8. Bu FB'yi istediğiniz konuma yerleştiriniz. Böylece bu FB için bir instance data bloğunun oluşturulması tamamlanır.
9. FB'yi editöre yerleştirdiğinizde oluşturduğunuz instance data bloğunu açınız.
10. Statik altından, ilgili sayıcı yapısını konumlandırın. Bu yapıyı kalıcı yapmak için bu sayıcı için olan Tut sütunundaki kutucuğu işaretleyiniz.

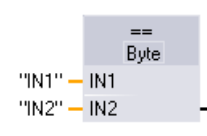
Sayıcı komutunda gösterilen tip (ön ayarlanan FB ara yüzündeki karşılık gelen yapı tipi ve sayım değerleri için)

INT	IEC_Counter
SINT	IEC_SCounter
DINT	IEC_DCounter
UINT	IEC_UCounter
USINT	IEC_USCounter
UDINT	IEC_UDCounter

## 7.4 Karşılaştırıcı işlemleri

### 7.4.1 Karşılaştırma değer komutları

Çizelge 7- 28 Karşılaştırma komutları

LAD	FBD	SCL	Açıklama
<pre>   "IN1"   ==   Byte   "IN2"   </pre>		<pre> out := in1 = in2; or IF in1 = in2   THEN out := 1;   ELSE out := 0; END IF; </pre>	<p>Aynı veri tipinin iki değerini karşılaştırır. LAD kontak karşılaştırması TRUE olduğunda, kontak aktive edilir.</p> <p>FBD kutusu karşılaştırması TRUE olduğunda, kutu çıkışı TRUE'dur.</p>

- <sup>1</sup> LAD ve FBD için: Açılır menü listesinden karşılaştırma tipini değiştirmek için komut adına ("==" gibi) tıklayınız. "???" üzerine tıklayınız ve açılır menü listesinden veri tipini seçiniz.

Çizelge 7- 29 Parametreler için veri tipleri

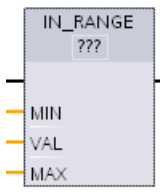
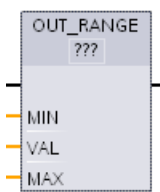
Parametre	Veri tipi	Açıklama
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, String, Char, Time, DTL, Constant	Karşılaştırılacak değerler

Çizelge 7- 30 Karşılaştırma açıklamaları

İlişki tipi	Şu durumda karşılaştırma TRUE'dur
=	IN1, IN2'ye eşit olduğunda
<>	IN1, IN2'ye eşit olmadığına
>=	IN1, IN2'den büyük ya da IN2'ye eşit olduğunda
<=	IN1, IN2'den küçük ya da IN2'ye eşit olduğunda
>	IN1, IN2'den büyük olduğunda
<	IN1, IN2'den küçük olduğunda

## 7.4.2 IN\_Range (Aralık içi) ve OUT\_Range (aralık dışı) komutlar

Çizelge 7- 31 Aralık içi ve aralık dışı değer komutlar

LAD / FBD	SCL	Açıklama
	<pre>out := IN_RANGE (min, val, max);</pre>	Bir giriş değerinin belirtilen değer aralığının içinde mi yoksa dışında mı olduğunu test eder. Eğer karşılaştırma TRUE ise, kutu çıkışı TRUE'dur.
	<pre>out := OUT_RANGE (min, val, max);</pre>	

<sup>1</sup> LAD ve FBD için: "???" üzerine tıklayınız ve açılır menü listesinden veri tipini seçiniz.

Çizelge 7- 32 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
MIN, VAL, MAX	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Constant	Karşılaştırmalı girişleri

<sup>1</sup> Giriş parametreleri MIN, VAL ve MAX aynı veri tipinden olmalıdır.

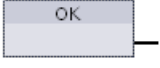
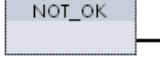
- Şu durumda IN\_RANGE karşılaştırması TRUEdur:  $MIN \leq VAL \leq MAX$
- Şu durumda OUT\_RANGE karşılaştırması TRUEdur:  $VAL < MIN$  or  $VAL > MAX$

## Temel komutlar

### 7.4 Karşılaştırıcı işlemleri

#### 7.4.3 OK (Tamam) ve NOT OK (Tamam değil) komutları

Çizelge 7- 33 OK (geçerliliği kontrol et) ve Not OK (geçersizliği kontrol et) komutları

LAD	FBD	SCL	Açıklama
"IN" ├─OK┤	"IN" 	Uygun değil	Bir giriş veri referansının IEEE spesifikasyonu 754'e göre gerçek bir sayı olup olmadığını test eder.
"IN" ├─NOT_OK┤	"IN" 	Uygun değil	

- <sup>1</sup> LAD ve FBD için: LAD kontağı TRUE olduğunda, kontak aktive edilir ve enerji akışını geçirir. FBD kutusu TRUE olduğunda, kutu çıkışı TRUE'dur.

Çizelge 7- 34 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	Real, LReal	Giriş verileri

Çizelge 7- 35 Çalışma

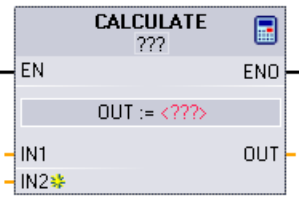
Komut	Şu durumda Reel sayı testi TRUE'dur:
OK	Giriş değeri geçerli bir reel sayıdır <sup>1</sup>
NOT_OK	Giriş değeri geçerli bir reel sayı değildir <sup>1</sup>

- <sup>1</sup> Bir Real ya da LReal değeri, eğer +/- INF (sonsuz), NaN (bir sayı değil) ya da denormalize edilmiş bir değerse geçersizdir. Denormalize edilmiş bir değer, sifıra çok yakın bir sayıdır. Hesaplamalarda CPU, denormalize bir değer için sıfır kullanır.

## 7.5 Matematik fonksiyonları

### 7.5.1 CALCULATE (Hesapla) komutu

Çizelge 7- 36 CALCULATE komutu

LAD / FBD	SCL	Açıklama
	Denklem oluşturmak için standart SCL ifadelerini kullanın.	<ul style="list-style-type: none"> <li>CALCULATE komutu, tanımladığınız eşitliğe göre girişler (IN1, IN2, ... INn) üzerinde çalışan ve OUT'ta sonuç üreten bir matematik fonksiyonu oluşturmanıza izin verir.</li> <li>Önce bir veri tipi seçiniz. Tüm girişler ve çıkış, aynı veri tipinden olmalıdır.</li> <li>Başka bir giriş eklemek için son girişteki simgeye tıklayınız.</li> </ul>

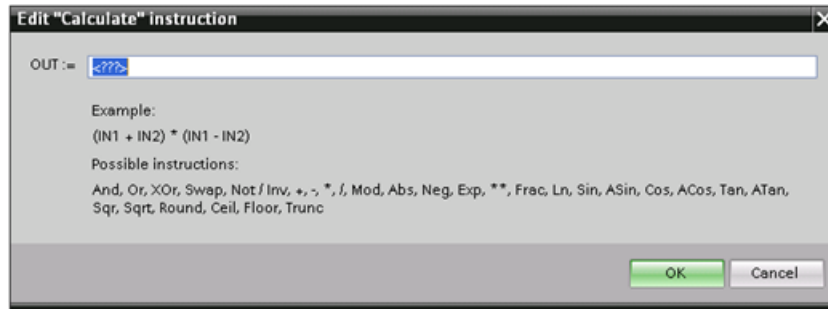
Çizelge 7- 37 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>
IN1, IN2, ..INn	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord

<sup>1</sup> IN ve OUT parametreleri aynı veri tipinden olmalıdır (giriş parametrelerinin kesin dönüşümleri ile). Örneğin, Bir giriş için bir SINT değeri, eğer OUT bir INT veya REAL ise bir INT veya REAL değerine dönüştürülebilir.

Diyalogu açmak ve matematik fonksiyonunuzu tanımlamak için hesap makinesi simgesine tıklayınız. Eşitliğinizi girişler (IN1 ve IN2 gibi) ve işlemler olarak girin. Fonksiyonu kaydetmek için Tamam'a tıkladığınızda, diyalog otomatik olarak CALCULATE komutu için girişleri oluşturur.

Editörün alt kısmında bir örnek ve kullanabileceğiniz matematiksel işlemler listelenmiştir.



### Not

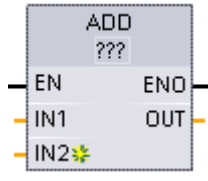
Fonksiyonunuzdaki tüm sabitler için bir giriş oluşturmalsınız. Sabit değer bundan sonra CALCULATE komutu için olan ilgili girişe girilmelidir.

Sabitleri girişler olarak girmek suretiyle, CALCULATE komutunu fonksiyonu değiştirmeye gerek kalmadan kullanıcı programınızdaki diğer konumlara kopyalayabilirsiniz. Bundan sonra komut için girişlerin değerlerini veya etiketleri, fonksiyonu değiştirmeden değiştirebilirsiniz

CALCULATE yürütüldüğünde ve hesaplamadaki tüm bağımsız işlemler başarılı bir şekilde tamamlandığında, ENO = 1'dir. Aksi halde ENO = 0'dır.

## 7.5.2 Toplama, çıkarma, çarpma ve bölme komutları

Çizelge 7- 38 Toplama, çıkarma, çarpma ve bölme komutları

LAD / FBD	SCL	Açıklama
	<pre>out := in1 + in2; out := in1 - in2; out := in1 * in2; out := in1 / in2;</pre>	<ul style="list-style-type: none"> <li>• ADD: Toplama (IN1 + IN2 = OUT)</li> <li>• SUB: Çıkarma (IN1 - IN2 = OUT)</li> <li>• MUL: Çarpma (IN1 * IN2 = OUT)</li> <li>• DIV: Bölme (IN1 / IN2 = OUT)</li> </ul> <p>Bir tamsayı bölme işlemi, bir tamsayı çıkışı almak için bölümün kesirli kısmını keser.</p>

<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz

Çizelge 7- 39 Parametreler için veri tipleri (LAD ve FBD)

Parametre	Veri tipi <sup>1</sup>	Açıklama
IN1, IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Constant	Matematik işlemi girişleri
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Matematik işlemi girişleri

<sup>1</sup> IN1, IN2 ve OUT parametreleri, aynı veri tipinden olmalıdır.



Bir ADD veya MUL girişi eklemek için “Oluştur” simgesine tıklayınız ya da mevcut IN parametrelerinden biri için bir girişe sağ tıklayınız ve “Giriş ekle” komutunu seçiniz.

Bir girişi kaldırmak için, mevcut IN parametrelerinden biri için kökün üzerine sağ tıklayınız (orijinal iki girişten fazlası olduğunda) ve “Sil” komutunu seçiniz.

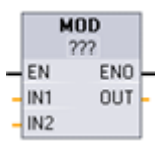
Etkinleştirildiğinde (EN = 1), matematik komutu giriş değeri üzerinde belirtilen işlemi gerçekleştirir (IN1 ve IN2) ve sonucu çıkış parametresi (OUT) tarafından belirlenen bellek adresine yazar. İşlemin başarılı bir şekilde tamamlanmasından sonra komut ENO = 1 olarak ayarlanır.

Çizelge 7- 40 ENO durumu

ENO	Açıklama
1	Hata yok
0	Matematik işleminin sonuç değeri, seçilen veri tipinin geçerli sayı aralığının dışında olacaktır. Sonucun hedef boyutuna uyan en küçük anlamlı bölümü döndürülecektir.
0	0 ile bölme (IN2 = 0): Sonuç tanımsızdır ve sıfır döndürülür.
0	Real/LReal: Eğer giriş değerlerinden biri NaN (bir sayı değil) ise, NaN döndürülür.
0	ADD Real/LReal: Eğer her iki IN değeri de farklı işaretlerle INF ise, bu kuraldışı bir işlemdir ve NaN döndürülür.
0	SUB Real/LReal: Eğer her iki IN değeri de aynı işaretle INF ise, bu kuraldışı bir işlemdir ve NaN döndürülür.
0	MUL Real/LReal: Eğer bir IN değeri sıfır değeri ise INF ise, bu geçersiz bir işlemdir ve NaN döndürülür.
0	DIV Real/LReal: Eğer her iki IN değeri de sıfır veya INF ise, bu geçersiz bir işlemdir ve NaN döndürülür.

### 7.5.3 MOD (bölmenin kalanına dön) komutu

Çizelge 7- 41 Modulo (bölmenin kalanına dön) komutu

LAD / FBD	SCL	Açıklama
	<pre>out := in1 MOD in2;</pre>	MOD komutunu, bir tamsayı bölme işleminin kalanını vermek için kullanabilirsiniz. IN1 girişindeki değer IN2 girişindeki değere bölünür ve kalan OUT çıkışına döndürülür.

1 LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

Çizelge 7- 42 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
IN1 ve IN2	SInt, Int, DInt, USInt, UInt, UDInt, Constant	Modulo girişleri
OUT	SInt, Int, DInt, USInt, UInt, UDInt	Modulo girişleri

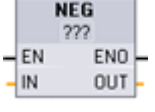
1 IN1, IN2 ve OUT parametreleri, aynı veri tipinden olmalıdır.

Çizelge 7- 43 ENO değerleri

ENO	Açıklama
1	Hata yok
0	IN2 = 0 değeri, OUT sıfır değerine atanır

## 7.5.4 NEG (ikinin tamamlayıcısını oluştur) komutu

Çizelge 7- 44 NEG (ikinin tamamlayıcısını oluştur) komutu

LAD / FBD	SCL	Açıklama
	- (in) ;	NEG komutu, IN parametresinin aritmetik işaretini tersine çevirir ve sonucu OUT parametresinde saklar.

<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

Çizelge 7- 45 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
IN	SInt, Int, DInt, Real, LReal, Constant	Matematik işlemi girişi
OUT	SInt, Int, DInt, Real, LReal	Matematik işlemi girişi


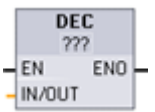
<sup>1</sup> IN ve OUT parametreleri, aynı veri tipinden olmalıdır.

Çizelge 7- 46 ENO durumu

ENO	Açıklama
1	Hata yok
0	Sonuç değeri seçilen veri tipinin geçerli sayı aralığının dışındadır. SInt için örnek: NEG (-128), +128 ile sonuçlanır ki bu da veri tipi maksimumunu aşar.

## 7.5.5 INC (Arttır) ve DEC (Azalt) komutları

Çizelge 7- 47 INC ve DEC komutları

LAD / FBD	SCL	Açıklama
	in_out := in_out + 1;	İşaretili veya işaretsiz bir tamsayı değerini artırır: IN_OUT değeri +1 = IN_OUT değeri
	in_out := in_out - 1;	İşaretili veya işaretsiz bir tamsayı değerini azaltır: IN_OUT değeri -1 = IN_OUT değeri

<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz.



Çizelge 7- 48 Parametreler için veri tipleri

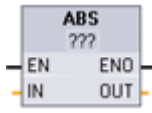
Parametre	Veri tipi	Açıklama
IN/OUT	SInt, Int, DInt, USInt, UInt, UDIInt	Matematik işlemi girişi ve çıkışı

Çizelge 7- 49 ENO durumu

ENO	Açıklama
1	Hata yok
0	Sonuç değeri seçilen veri tipinin geçerli sayı aralığının dışındadır. SInt için örnek: INC (+127), +128 ile sonuçlanır ki bu da veri tipi maksimumunu aşar.

## 7.5.6 ABS (Mutlak değer oluştur) komutu

Çizelge 7- 50 ABS (mutlak değer) komutu

LAD / FBD	SCL	Açıklama
	<pre>out := ABS(in);</pre>	IN parametresindeki işaretli veya reel bir sayının mutlak değerini hesaplar ve sonucu OUT parametresinde saklar.

1 LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

Çizelge 7- 51 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
IN	SInt, Int, DInt, Real, LReal	Matematik işlem girişi
OUT	SInt, Int, DInt, Real, LReal	Matematik işlem çıkışı

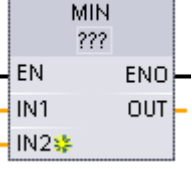
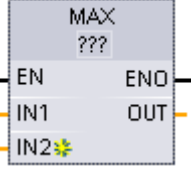
1 IN ve OUT parametreleri, aynı veri tipinden olmalıdır.

Çizelge 7- 52 ENO durumu

ENO	Açıklama
1	Hata yok
0	Matematik işleminin sonuç değeri seçilen veri tipinin geçerli sayı aralığının dışındadır. SInt için örnek: ABS (-128), +128 ile sonuçlanır ki bu da veri tipi maksimumunu aşar.

## 7.5.7 MIN (Minimum) ve MAX (Maksimum) komutları

Çizelge 7- 53 MIN (minimumu al) ve MAX (maksimumu al) komutları

LAD / FBD	SCL	Açıklama
	<pre>out:= MIN(     in1:=_variant_in_,     in2:=_variant_in_     [,...in32]);</pre>	<p>MIN komutu IN1 ve IN2 parametrelerinin değerlerini karşılaştırır ve minimum (az olan) değeri OUT parametresine atar.</p>
	<pre>out:= MAX(     in1:=_variant_in_,     in2:=_variant_in_     [,...in32]);</pre>	<p>MAX komutu IN1 ve IN2 parametrelerinin değerlerini karşılaştırır ve maksimum (çok olan) değeri OUT parametresine atar.</p>

<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz

Çizelge 7- 54 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
IN1, IN2 [...IN32]	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Constant	Matematik işlem girişleri (32 girişe kadar)
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Matematik işlemi çıkışı

<sup>1</sup> IN1, IN2 ve OUT parametreleri, aynı veri tipinden olmalıdır.



Bir giriş eklemek için “Oluştur” simgesine tıklayınız ya da mevcut IN parametrelerinden biri için bir girişe sağ tıklayınız ve “Giriş ekle” komutunu seçiniz.


Bir girişi kaldırmak için, mevcut IN parametrelerinden biri için kökün üzerine sağ tıklayınız (orijinal iki girişten fazlası olduğunda) ve “Sil” komutunu seçiniz.

Çizelge 7- 55 ENO durumu

ENO	Açıklama
1	Hata yok
0	Sadece Reel veri tipi için: <ul style="list-style-type: none"> <li>Bir veya birden fazla giriş bir reel sayı değildir (NaN).</li> <li>Sonuç OUT” +/- INF (sonsuz) şeklindedir.</li> </ul>

## 7.5.8 LIMIT (Limit değeri ver) komutu

Çizelge 7- 56 LIMIT (limit değeri ver) komutu

LAD / FBD	SCL	Açıklama
	<pre>LIMIT(MN:=_variant_in_,       IN:=_variant_in_,       MX:=_variant_in_,       OUT:=_variant_out_);</pre>	Limit komutu, IN parametresinin değerinin MIN ve MAX parametreleri tarafından belirlenen değer aralığının içinde olup olmadığını test eder ve eğer değer bu aralıkta değilse değeri MIN veya MAX'a kilitler.

1 LAD ve FBD için: "???" üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

Çizelge 7- 57 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
MN, IN ve MX	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Constant	Matematik işlemi girişleri
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Matematik işlemi çıkışı

1 MIN, IN, MAX ve OUT parametreleri, aynı veri tipinden olmalıdır.

Eğer IN parametresinin değeri belirtilen aralıkta ise, o zaman IN değeri OUT parametresinde saklanır. Eğer IN parametresinin değeri belirtilen aralığın dışında ise, o zaman OUT değeri MIN (eğer IN değeri MIN değerinden küçükse) parametresinin değeridir ya da MAX (eğer IN değeri MAX değerinden büyükse) parametresinin değeridir.

Çizelge 7- 58 ENO durumu

ENO	Açıklama
1	Hata yok
0	Eğer MIN, IN ve MAX için bir veya birden fazla değer NaN (bir sayı değil) ise, o zaman NaN döndürülür.
0	Eğer MIN değeri MAX değerinden büyükse, IN değeri OUT'a atanır.

SCL örnekleri:

- MyVal := LIMIT(MN:=10,IN:=53, MX:=40); //Result: MyVal = 40
- MyVal := LIMIT(MN:=10,IN:=37, MX:=40); //Result: MyVal = 37
- MyVal := LIMIT(MN:=10,IN:=8, MX:=40); //Result: MyVal = 10

### 7.5.9 Üslü, logaritma ve trigonometri komutları



Kayan nokta komutlarını, Real veya LReal veri tipi kullanarak matematiksel işlemleri programlamak için kullanabilirsiniz:

- SQR: Kare oluştur ( $IN^2 = OUT$ )
- SQRT: Karekök oluştur ( $\sqrt{IN} = OUT$ )
- LN: Doğal logaritma oluştur ( $LN(IN) = OUT$ )
- EXP: Üslü değer oluştur ( $e^{IN} = OUT$ ), taban  $e = 2.71828182845904523536$  olduğu yerde
- EXPT: Üs almak ( $IN1^{IN2} = OUT$ )

EXPT parametreleri IN1 ve IN2, çoğunlukla Real veya LReal olarak seçmeniz gereken aynı veri tipleridir. Üslü parametre için IN2'yi birçok veri tipinin arasından seçebilirsiniz.

- FRAC: Kesiri getir (Kayan nokta sayısının kesirli bölümü  $IN = OUT$ )
- SIN: sinüs değeri oluştur ( $\sin(IN \text{ radians}) = OUT$ )
- ASIN: arcsin değeri oluştur ( $\sin(OUT \text{ radians}) = IN$  olduğu yerde  $\arcsin(IN) = OUT$  radians),
- COS: Kosinüs oluştur ( $\cos(IN \text{ radians}) = OUT$ )
- ACOS: arccos değeri oluştur ( $\cos(OUT \text{ radians}) = IN$  olduğu yerde  $\arccos(IN) = OUT$  radians),
- TAN: Tanjant değeri oluştur ( $\tan(IN \text{ radians}) = OUT$ )
- ATAN: Arctanjant değeri oluştur ( $\tan(OUT \text{ radians}) = IN$  olduğu yerde  $\arctan(IN) = OUT$  radians),

Çizelge 7- 59 Kayan nokta matematik komutlarının örnekleri

LAD / FBD	SCL	Açıklama
	<pre>out := SQR(in); or out := in * in;</pre>	<p>Kare: <math>IN^2 = OUT</math> Örneğin: Eğer <math>IN = 9</math>, sonra <math>OUT = 81</math>.</p>
	<pre>out := in1 ** in2;</pre>	<p>Genel üstel: <math>IN1^{IN2} = OUT</math> Örneğin: Eğer <math>IN1 = 3</math> ve <math>IN2 = 2</math>, sonra <math>OUT = 9</math>.</p>

<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız (komut adı olarak) ve açılır menüden veri tipini seçiniz.

<sup>2</sup> For SCL: matematiksel ifadeleri oluşturmak için temel SCL matematik operatörlerini de kullanabilirsiniz.

Çizelge 7- 60 Parametreler için veri tipleri

Parametre	Data type	Açıklama
IN, IN1	Real, LReal, Constant	Girdiler
IN2	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Constant	EXPT üst girdisi
OUT	Real, LReal	Çıktılar

Çizelge 7- 61 ENO durumu

ENO	Komut	Durum	Sonuç (OUT)
1	All	Hata yok	Geçerli sonuç
0	SQR	Sonuç geçerli Real/LReal aralığını aşar	+INF
		IN is +/- NaN (sayı değil)	+NaN
	SQRT	IN negatif	-NaN
		IN +/- INF (sonsuz) veya +/- NaN	+/- INF veya +/- NaN
	LN	IN 0.0, negatif, -INF, veya -NaN	-NaN
		IN +INF veya +NaN	+INF veya +NaN
	EXP	Sonuç geçerli Real/LReal aralığını aşar	+INF
		IN is +/- NaN	+/- NaN
	SIN, COS, TAN	IN is +/- INF or +/- NaN	+/- INF veya +/- NaN
	ASIN, ACOS	IN geçerli -1.0'den +1.0 aralığının dışında	+NaN
		IN +/- NaN	+/- NaN
	ATAN	IN +/- NaN	+/- NaN
	FRAC	IN +/- INF veya +/- NaN	+NaN
	EXPT	IN1 +INF ve IN2 -INF değil	+INF
		IN1 negatif veya -INF	+NaN eğer IN2 ise Real/LReal, diğer türlü -INF
		IN1 veya IN2 +/- NaN	+NaN
IN1 0.0 ve IN2 Real/LReal (sadece)		+NaN	

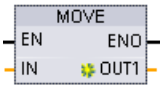
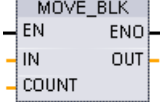
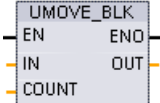
## 7.6 Taşı komutları

### 7.6.1 MOVE (değeri taşı), MOVE\_BLK (blok taşı), ve UMOVE\_BLK (bloğu kesintisiz taşı) komutları

Taşı komutlarını veri elemanlarını yeni bir bellek adresine kopyalamak ve bir veri tipinden diğerine dönüşüm için kullanın. Taşıma işleminde kaynak verileri değişmez.

- MOVE komutu, tek bir veri elemanını IN parametresi tarafından tanımlanan kaynak adresinden OUT parametresi tarafından tanımlanan hedef adresine kopyalar.
- MOVE\_BLK ve UMOVE\_BLK komutlarının ek bir COUNT parametresi vardır. COUNT, kaç tane veri elemanının kopyalandığını belirtir. Kopyalaman eleman başına baytların sayısı, PLC etiket tablosundaki IN ve OUT parametre etiket isimlerine atanan veri tiplerine bağlıdır.

Çizelge 7- 62 MOVE, MOVE\_BLK ve UMOVE\_BLK komutları

LAD / FBD	SCL	Açıklama
	<pre>out1 := in;</pre>	Belirli bir adreste saklanan bir veri elemanını yeni bir adrese ya da birden fazla adrese kopyalar. <sup>1</sup>
	<pre>MOVE_BLK (     in:=_variant_in,     count:=_uint_in,     out=&gt;_variant_out);</pre>	Bir veri elemanları bloğunu yeni bir adrese kopyalayan kesintiye uğratılabilir taşıma.
	<pre>UMOVE_BLK (     in:=_variant_in,     count:=_uint_in,     out=&gt;_variant_out);</pre>	Bir veri elemanları bloğunu yeni bir adrese kopyalayan kesintiye uğratılmaz taşıma.

<sup>1</sup> MOVE komutu: LAD veya FBD'ye başka bir çıkış eklemek için, çıkış parametresinin yanındaki "Oluştur" simgesine tıklayınız.

Çizelge 7- 63 MOVE komutu için veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Char, Array, Struct, DTL, Time	Kaynak adresi
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Char, Array, Struct, DTL, Time	Hedef adresi



MOVE çıkışları eklemek için "Oluştur" simgesine tıklayınız ya da mevcut OUT parametrelerinden biri için bir girişe sağ tıklayınız ve "Çıkış ekle" komutunu seçiniz.

Bir çıkışı kaldırmak için, mevcut OUT parametrelerinden biri için kökün üzerine sağ tıklayınız (orijinal iki çıkıştan fazlası olduğunda) ve "Sil" komutunu seçiniz.

Çizelge 7- 64 MOVE\_BLK ve UMOVE\_BLK komutları için veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal Byte, Word, DWord	Kaynak başlangıç adresi
COUNT	UInt	Kopyalanacak veri elemanlarının sayısı
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord	Hedef başlangıç adresi

#### Not

##### Veri kopyalama işlemleri için kurallar

- Boole veri tipini kopyalamak için SET\_BF, RESET\_BF, R, S veya çıkış bobinini (LAD) kullanın (sayfa 3).
- Tek bir temel veri tipini kopyalamak için MOVE komutunu kullanın.
- Bir temel veri tipinin bir dizisini kopyalamak için MOVE\_BLK veya UMOVE\_BLK komutunu kullanın.
- Bir yapıyı kopyalamak için MOVE komutunu kullanın.
- Bir string'i kopyalamak için S\_MOVE komutunu kullanın (sayfa 3)
- Bir string'deki tek bir karakteri kopyalamak için MOVE komutunu kullanın.
- MOVE\_BLK ve UMOVE\_BLK komutları, I, Q veya M hafızası alanlarına diziler veya yapılar kopyalamak için kullanılamaz.

MOVE\_BLK ve UMOVE\_BLK komutları interrupt'ların işlenmesinde farklıdır:

- Interrupt olayları, MOVE\_BLK yürütmesi sırasında **kuyruklanır ve işlenir**. MOVE\_BLK komutunu, taşımanın hedef adresindeki veriler bir interrupt OB alt programı içinde kullanılmıyorsa ya da eğer hedef verilerinin tutarlı olması gerekmiyorsa kullanın. Eğer bir MOVE\_BLK işlemi kesintiye uğratılırsa, o zaman taşınan son veri elemanı tamamlanır ve hedef adreste tutarlıdır. MOVE\_BLK işlemi, interrupt OB yürütmesi tamamlandıktan sonra devam eder.
- Interrupt olayları **kuyruklanır ancak UMOVE\_BLK yürütmesi tamamlanana kadar işlenmez**. Taşıma işlemi tamamlanması gerektiğinde ve hedef verilerinin tutarlı olması gerektiğinde, bir interrupt OB alt programının yürütülmesinden önce UMOVE\_BLK komutunu kullanın. Daha fazla bilgi için veri tutarlılığıyla ilgili bölüme bakabilirsiniz (Sayfa 3)

MOVE komutunun yürütülmesini takiben ENO her zaman TRUE'dur.

## Temel komutlar

### 7.6 Taşı komutları

Çizelge 7- 65 ENO durumu

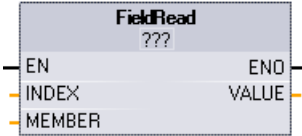

ENO	Durum	Sonuç
1	Hata yok	Tüm COUNT elemanları başarılı bir şekilde kopyalandı.
0	Kaynak (IN) aralığı ya da hedef (OUT) aralığı, kullanılabilir bellek alanını aşıyor.	Uygun olan elemanlar kopyalandı. Hiç kısmi eleman kopyalaması yapılmadı.

### 7.6.2 FieldRead (alanı oku) ve FieldWrite (alana yaz) komutları

#### Not

STEP 7 V10.5, bir dizi indeksi ya da çok boyutlu diziler olarak bir değişken referansını desteklemiyordu. FieldRead ve FieldWrite komutları, bir tek boyutlu dizi için değişken dizi indeks işlemleri sunmak için kullanılıyordu. STEP 7 V11, bir dizi indeksi ya da çok boyutlu diziler olan bir değişkeni destekliyor. FieldRead ve FieldWrite, STEP 7 V11'ye, bu komutları kullanan programlar için geriye dönük uyumluluk sağlamak için dahil edilmiştir.

Çizelge 7- 66 FieldRead ve FieldWrite komutları

LAD / FBD	SCL	Açıklama
	<pre>value := member[index];</pre>	FieldRead, dizi elemanını, ilk elemanı MEMBER parametresi tarafından belirtilen diziden INDEX indeks değeri ile okur. Dizi elemanının değeri VALUE parametresinde belirtilen konuma transfer edilir.
	<pre>member[index] := value;</pre>	FieldWrite, VALUE parametresi tarafından belirlenen konumdaki değeri, ilk elemanı MEMBER parametresi tarafından belirlenen diziyeye transfer eder. Değer, dizi indeksi INDEX parametresi tarafından belirlenen dizi elemanına transfer edilir.

1 LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz.



Çizelge 7- 67 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
Dizin	Giriş	DInt	Yazılacak veya okunacak dizi elemanının indeks
Üye <sup>1</sup>	Giriş	Array eleman tipleri: Bool, Byte, Word, DWord, Char, SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Bir global veri bloğunda ya da blok ara yüzünde tanımlanan tek boyutlu bir dizideki ilk elemanın konumu. Örneğin eğer dizi [-2..4] olarak belirtilmişse, o zaman ilk elemanın indeksi -2'dir, 0 değildir.
Değer <sup>1</sup>	Çıkış	Bool, Byte, Word, DWord, Char, SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Belirtilen dizi elemanının kopyalanacağı konum (FieldRead)  Belirtilen dizi elemanına kopyalanacak değerın konumu (FieldWrite)

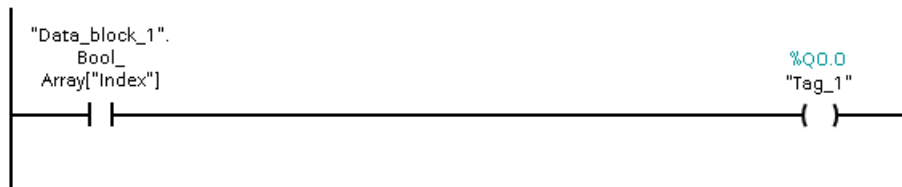
<sup>1</sup> MEMBER parametresi ve VALUE parametresi tarafından belirtilen dizi elemanının veri tipleri aynı olmalıdır.

Eğer şu durumlardan biri geçerli ise ENO = 1 çıkışını etkinleştirin:

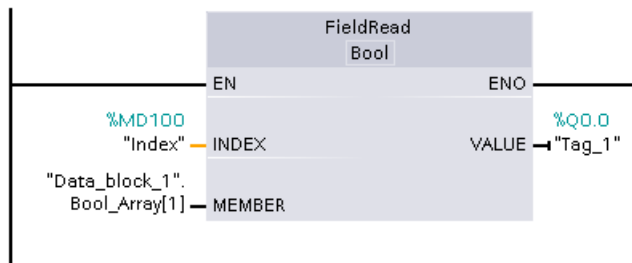
- EN girişi "0" sinyal durumuna sahipse
- INDEX parametresinde belirtilen dizi elemanı, MEMBER parametresinde referans verilen dizide tanımlanmamışsa
- İşleme sırasında bir taşma benzeri hatalar meydana gelmişse

## Dizi indeksleme ile verilere erişme

Bir değişkenli bir dizi öğelerine erişmek için değişkeni sadece program mantığınızda bir dizi indeksi olarak kullanın. Örneğin, aşağıdaki devre, "İndeks" PLC etiketi ile referanslanan "Data\_block\_1" deki Boole bir dizinin Boole değerine dayalı bir çıkış ayarlar.



Değişken dizi indeksi olan mantık, FieldRead komutunu kullanan eski yöntemle eşdeğerdir.




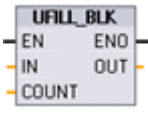
FieldWrite ve FieldRead komutları değişken dizi indeksleme mantığı ile değiştirilebilir.

SCL, hiçbir FieldRead veya FieldWrite komutuna sahip değildir ancak bir değişkenli bir dizinin dolaylı olarak adreslenmesini destekler:

```
#Tag_1 := "Data_block_1".Bool_Array[#Index];
```

### 7.6.3 FILL\_BLK (bloğu doldur) ve UFILL\_BLK (bloğu kesintisiz doldur) talimatları

Çizelge 7- 68 FILL\_BLK ve UFILL\_BLK talimatları

LAD / FBD	SCL	Açıklama
	<pre>FILL_BLK(     in:=_variant_in,     count:=int,     out=&gt;_variant_out);</pre>	Kesilebilir doldur komutu: belirtilen bir veri elemanının kopyaları ile birlikte bir adres aralığı doldurulur.
	<pre>UFILL_BLK(     in:=_variant_in,     count:=int     out=&gt;_variant_out);</pre>	Kesintisiz doldur komutu: belirtilen bir veri elemanının kopyaları ile birlikte bir adres aralığı doldurulur.

Çizelge 7- 69 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord	Veri kaynağı adresi
COUNT	USInt, UInt	Kopyalamak üzere veri elemanlarının sayısı
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Byte, Word, DWord	Veri hedefi adresi

#### Not

##### Veri doldurma işlemleri için kurallar

- BOOLE veri tipi ile doldurmak için SET\_BF, RESET\_BF, R, S veya çıkış bobini (LAD) kullanılır
- Tekli temel veri tipi ile doldurmak için MOVE kullanılır
- Temel veri tipi ile bir diziyi doldurmak için FILL\_BLK veya UFILL\_BLK kullanılır
- Bir string'de tek bir karakteri doldurmak için MOVE kullanılır
- I, Q veya M bellek alanlarındaki dizileri doldurmak için FILL\_BLK ve UFILL\_BLK komutları kullanılmaz

FILL\_BLK ve UFILL\_BLK komutları, başlangıç adresi OUT parametresi ile belirtilen hedefe kaynak veri ögesini (IN) kopyalar. Kopyalama işlemi tekrar eder ve kopyaların sayısı COUNT parametresine eşit olana kadar bir blok komşu adres doldurulur.

FILL\_BLK ve UFILL\_BLK komutları kesmelerin nasıl işleneceği hususunda farklılık gösterir:

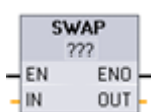
- Kesme olayları, FILL\_BLK yürütmesi sırasında sıraya konur ve işlenir. Taşıma hedef adresindeki veri, kesme OB alt program içerisinde kullanılmadığında FILL\_BLK komutunu kullanın ya da bu veri kesme OB alt program içerisinde kullanılırsa, hedef veri tutarlı olmak zorunda değildir.
- Kesme olaylar sıraya konur ancak UFILL\_BLK yürütmesi tamamlanana kadar işlenmez. Kesme OB alt programın yürütülmesinden önce taşıma işlemi tamamlandığında ve hedef veri tutarlı olduğunda UFILL\_BLK komutunu kullanın.

Çizelge 7- 70 ENO durumu

ENO	Şart	Sonuç
1	Hata yok	IN elemanı, Tüm COUNT hedeflerine başarılı bir şekilde kopyalandı.
0	Hedef (OUT) aralığı, mevcut bellek alanını aşar	Eşleşen elemanlar kopyalanır. Kısmi (eksik) elemanlar kopyalanmaz.

## 7.6.4 SWAP (bitleri deęiş tokuş et) komutu

Çizelge 7- 71 SWAP komutu

LAD / FBD	SCL	Açıklama
	<pre>out := SWAP(in);</pre>	<p>İki bayt ve dört bayt veri elemanları için bayt sırasını tersine çevirir. Baytların içindeki bit sırasında herhangi bir deęişiklik yapılmaz. SWAP komutunun yürütülmesini takiben ENO her zaman TRUE'dur.</p>

<sup>1</sup> LAD ve FBD için: "???" üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

## Temel komutlar

### 7.7 Dönüştürme işlemleri

Çizelge 7- 72 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	Word, DWord	Sıralanan veri tipleri IN
OUT	Word, DWord	Ters sıralanan veri tipleri OUT

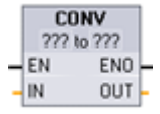
Örnek 1	Parametre IN = MB0 (yürütmeden önce)	Parametre OUT = MB4, (yürütmeden sonra)
Adres	MW0 MB1	MW4 MB5
W#16#1234	12 34	34 12
WORD	MSB LSB	MSB LSB

Örnek 2	Parametre IN = MB0 (yürütmeden önce)	Parametre OUT = MB4, (yürütmeden sonra)
Adres	MD0 MB1 MB2 MB3	MD4 MB5 MB6 MB7
DW#16# 12345678	12 34 56 78	78 56 34 12
DWORD	MSB LSB	MSB LSB

## 7.7 Dönüştürme işlemleri

### 7.7.1 CONV (değeri dönüştür) komutu

Çizelge 7- 73 Dönüştür (CONV) komutu

LAD / FBD	SCL	Açıklama
	<pre>out := &lt;data type in&gt;_TO_&lt;data type out&gt;(in);</pre>	<p>Bir veri elemanını bir veri tipinden başka bir veri tipine dönüştürür.</p>

- LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tiplerini seçiniz.
- SCL için: Giriş parametresi (in) ve çıkış parametresi (out) için veri tipini tanımlayarak dönüşüm komutunu oluşturun. Örneğin, DWORD\_TO\_REAL bir DWord değeri gerçek (Real) değere dönüştürür.

Çizelge 7- 74 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	Bit string <sup>1</sup> , SInt, USInt, Int, UInt, DInt, UDInt, Real, LReal, BCD16, BCD32	Giriş değeri
OUT	Bit string <sup>1</sup> , SInt, USInt, Int, UInt, DInt, UDInt, Real, LReal, BCD16, BCD32	Giriş değeri yeni bir veri tipine dönüştürüldü

<sup>1</sup> Komut, Bit string'leri (Byte, Word, DWord) seçmenize izin vermez. Komutun bir parametresi için Byte, Word ve DWord veri tipinin operandını girmek amacıyla, aynı bit uzunluğunda bir işaretli tamsayı (integer) seçiniz. Örneğin, bir Byte için USInt, bir Word için UInt veya bir Dword için UDInt seçiniz.

Veri tipini seçtikten sonra (dönüştürülecek olan veri tipi), açılır menü listesinde olası dönüşümlerin bir listesi (hangi veri tipine dönüştürülebileceği) görüntülenir. BCD16 veri tipinden ve bu veri tipine yapılacak dönüşümler Int veri tipi ile sınırlıdır. BCD32 veri tipinden ve bu veri tipine yapılacak dönüşümler DInt veri tipi ile sınırlıdır.

Çizelge 7- 75 ENO durumu

ENO	Açıklama	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	IN, +/- INF veya +/- NaN	+/- INF veya +/- NaN
0	Sonuç, OUT veri tipi için olan geçerli aralığı aşıyor	OUT, IN'in en az anlamlı baytlarına ayarlanmıştır

## 7.7.2 SCL için dönüşüm komutları

### SCL için dönüşüm komutları

Çizelge 7- 76 Bir Boole, Byte, Word veya DWord'den dönüşüm

Veri tipi	Komut	Sonuç
Bool	BOOL_TO_BYTE, BOOL_TO_WORD, BOOL_TO_DWORD, BOOL_TO_INT, BOOL_TO_DINT	Bu değer, hedef veri tipinin en az anlamlı bit'ine transfer edilir.
Byte	BYTE_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	BYTE_TO_WORD, BYTE_TO_DWORD	Bu değer, hedef veri tipinin en az anlamlı byte'ına transfer edilir.
	BYTE_TO_SINT, BYTE_TO_USINT	Bu değer, hedef veri tipine transfer edilir.
	BYTE_TO_INT, BYTE_TO_UINT, BYTE_TO_DINT, BYTE_TO_UDINT	Bu değer, hedef veri tipinin en az anlamlı byte'ına transfer edilir.
Word	WORD_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	WORD_TO_BYTE	Kaynak değerinin en az anlamlı byte'ı, hedef veri tipine transfer edilir.
	WORD_TO_DWORD	Bu değer, hedef veri tipinin en az anlamlı word'üne transfer edilir.

## Temel komutlar

## 7.7 Dönüştürme işlemleri

Veri tipi	Komut	Sonuç
	WORD_TO_SINT, WORD_TO_USINT	Kaynak değerinin en az anlamlı byte'ı, hedef veri tipine transfer edilir.
	WORD_TO_INT, WORD_TO_UINT	Bu değer, hedef veri tipine transfer edilir.
	WORD_TO_DINT, WORD_TO_UDINT	Bu değer, hedef veri tipinin en az anlamlı word'üne transfer edilir
DWord	DWORD_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	DWORD_TO_BYTE, DWORD_TO_WORD, DWORD_TO_SINT	Kaynak değerinin en az anlamlı byte'ı, hedef veri tipine transfer edilir.
	DWORD_TO_USINT, DWORD_TO_INT, DWORD_TO_UINT	Kaynak değerinin en az anlamlı word'ü, hedef veri tipine transfer edilir.
	DWORD_TO_DINT, DWORD_TO_UDINT, DWORD_TO_REAL	Bu değer, hedef veri tipine transfer edilir.

Çizelge 7- 77 Kısa tamsayıdan (integer ) dönüşüm (SInt veya USInt)

Veri Tipi	Komut	Sonuç
SInt	SINT_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	SINT_TO_BYTE	Bu değer, hedef veri tipine transfer edilir.
	SINT_TO_WORD, SINT_TO_DWORD	Bu değer, hedef veri tipinin en az anlamlı byte'na transfer edilir.
	SINT_TO_INT, SINT_TO_DINT, SINT_TO_USINT, SINT_TO_UINT, SINT_TO_UDINT, SINT_TO_REAL, SINT_TO_LREAL, SINT_TO_CHAR, SINT_TO_STRING	Bu değer dönüştürülür.
USInt	USINT_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	USINT_TO_BYTE	Bu değer, hedef veri tipine transfer edilir.
	USINT_TO_WORD, USINT_TO_DWORD, USINT_TO_INT, USINT_TO_UINT, USINT_TO_DINT, USINT_TO_UDINT	Bu değer, hedef veri tipinin en az anlamlı byte'na transfer edilir.
	USINT_TO_SINT, USINT_TO_REAL, USINT_TO_LREAL, USINT_TO_CHAR, USINT_TO_STRING	Bu değer dönüştürülür.

Çizelge 7- 78 Bir tamsayıdan (integer ) dönüşüm (Int veya UInt)

Veri tipi	Komut	Sonuç
Int	INT_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	INT_TO_BYTE, INT_TO_DWORD, INT_TO_SINT, INT_TO_USINT, INT_TO_UINT, INT_TO_UDINT, INT_TO_REAL, INT_TO_LREAL, INT_TO_CHAR, INT_TO_STRING	Bu değer dönüştürülür.
	INT_TO_WORD	Bu değer, hedef veri tipine transfer edilir.
	INT_TO_DINT	Bu değer, hedef veri tipinin en az anlamlı byte'na transfer edilir.
UInt	UINT_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	UINT_TO_BYTE, UINT_TO_SINT, UINT_TO_USINT, UINT_TO_INT, UINT_TO_REAL, UINT_TO_LREAL, UINT_TO_CHAR, UINT_TO_STRING	Bu değer dönüştürülür.
	UINT_TO_WORD, UINT_TO_DATE	Bu değer, hedef veri tipine transfer edilir.
	UINT_TO_DWORD, UINT_TO_DINT, UINT_TO_UDINT	Bu değer, hedef veri tipinin en az anlamlı byte'na transfer edilir.

Çizelge 7- 79 Çift tamsayıdan (integer ) dönüşüm (Dint veya UInt)

Veri tipi	Komut	Sonuç
DInt	DINT_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	DINT_TO_BYTE, DINT_TO_WORD, DINT_TO_SINT, DINT_TO_USINT, DINT_TO_INT, DINT_TO_UINT, DINT_TO_UDINT, DINT_TO_REAL, DINT_TO_LREAL, DINT_TO_CHAR, DINT_TO_STRING	Bu değer dönüştürülür.
	DINT_TO_DWORD, DINT_TO_TIME	Bu değer, hedef veri tipine transfer edilir.
UDInt	UDINT_TO_BOOL	En az anlamlı bit, hedef veri tipine transfer edilir.
	UDINT_TO_BYTE, UDINT_TO_WORD, UDINT_TO_SINT, UDINT_TO_USINT, UDINT_TO_INT, UDINT_TO_UINT, UDINT_TO_DINT, UDINT_TO_REAL, UDINT_TO_LREAL, UDINT_TO_CHAR, UDINT_TO_STRING	Bu değer dönüştürülür.
	UDINT_TO_DWORD, UDINT_TO_TOD	Bu değer, hedef veri tipine transfer edilir.

## Temel komutlar

### 7.7 Dönüştürme işlemleri

Çizelge 7- 80 Gerçek sayıdan (reel sayı) dönüşüm (Real veya LReal)

Veri tipi	Komut	Sonuç
Real	REAL_TO_DWORD, REAL_TO_LREAL	Bu değer, hedef veri tipine transfer edilir.
	REAL_TO_SINT, REAL_TO_USINT, REAL_TO_INT, REAL_TO_UINT, REAL_TO_DINT, REAL_TO_UDINT, REAL_TO_STRING	Bu değer dönüştürülür.
LReal	LREAL_TO_SINT, LREAL_TO_USINT, LREAL_TO_INT, LREAL_TO_UINT, LREAL_TO_DINT, LREAL_TO_UDINT, LREAL_TO_REAL, LREAL_TO_STRING	Bu değer dönüştürülür.

Çizelge 7- 81 Zaman(Time), DTL, TOD veya tarihten (Date) dönüşüm

Veri tipi	Komut	Sonuç
Zaman (Time)	TIME_TO_DINT	Bu değer, hedef veri tipine transfer edilir.
DTL	DTL_TO_DATE, DTL_TO_TOD	Bu değer dönüştürülür.
TOD	TOD_TO_UDINT	Bu değer dönüştürülür.
Tarih (Date)	DATE_TO_UINT	Bu değer dönüştürülür.



Çizelge 7- 82 Char (Karakter) veya String'den dönüşüm

Veri tipi	Komut	Sonuç
Char	CHAR_TO_SINT, CHAR_TO_USINT, CHAR_TO_INT, CHAR_TO_UINT, CHAR_TO_DINT, CHAR_TO_UDINT	Bu değer dönüştürülür.
	CHAR_TO_STRING	Bu değer, string'in ilk karakterine transfer edilir.
String	STRING_TO_SINT, STRING_TO_USINT, STRING_TO_INT, STRING_TO_UINT, STRING_TO_DINT, STRING_TO_UDINT, STRING_TO_REAL, STRING_TO_LREAL	Bu değer dönüştürülür.
	STRING_TO_CHAR	String'in ilk karakteri, Char'a transfer edilir.



### 7.7.3 ROUND (sayısal değeri yuvarla) ve TRUNC (sayısal değeri kes) komutları

Çizelge 7- 83 ROUND ve TRUNC komutları

LAD / FBD	SCL	Açıklama
	<code>out := ROUND (in);</code>	<p>Bir reel sayıyı bir tamsayıya çevirir. LAD / FBD de, çıkış için veri tipini seçmek amacıyla örneğin "DInt", komut kutusundaki "???" tıklayınız.</p> <p>SCL de, ROUND komut çıkışı için varsayılan veri tipi DINT tir. Başka bir çıkış veri tipine yuvarlamak için komut adını veri tipinin açık adı ile giriniz, örneğin, ROUND_REAL veya ROUND_LREAL..</p> <p>Reel sayının kesri, en yakın tamsayıya yuvarlanır (IEEE- en yakına yuvarlama). Eğer sayı iki tamsayı arasında tam ortada ise (mesela 10,5), o zaman sayı en yakındaki çift tamsayıya yuvarlanır. Örneğin,</p> <ul style="list-style-type: none"> <li>• ROUND (10.5) = 10</li> <li>• ROUND (11.5) = 12</li> </ul>
	<code>out := TRUNC (in);</code>	<p>TRUNC, bir reel sayıyı bir tamsayıya çevirir. Reel sayının kesir kısmı, sıfır yapılır. (IEEE – sıfıra yuvarla)</p>

<sup>1</sup> LAD ve FBD için: "???" (komut ismine göre) üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

Çizelge 7- 84 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	Real, LReal	Kayan nokta girişi
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Yuvarlanmış veya kesilmiş çıkış

Çizelge 7- 85 ENO durumu

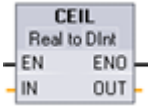

ENO	Açıklama	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	IN, +/- INF veya +/- NaN	+/- INF veya +/- NaN

## Temel komutlar

### 7.7 Dönüştürme işlemleri

#### 7.7.4 CEIL ve FLOOR komutları (Kayan nokta sayısı komutlarından sonraki daha yüksek ve daha düşük tamsayı üretme)

Çizelge 7- 86 CEIL ve FLOOR komutları

LAD / FBD	SCL	Açıklama
	<pre>out := CEIL(in);</pre>	Bir reel sayı (Real veya LReal), seçilen reel sayıya eşit olan ya da bu reel sayıdan büyük en yakın tamsayıya dönüştürür (IEEE "+ sonsuza yuvarla")
	<pre>out := FLOOR(in);</pre>	Bir reel sayı (Real veya LReal), seçilen reel sayıya eşit olan ya da bu reel sayıdan küçük en yakın tamsayıya dönüştürür (IEEE "- sonsuza yuvarla").

<sup>1</sup> LAD ve FBD için: "???" (komut ismine göre) üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

Çizelge 7- 87 Parametreler için veri tipleri

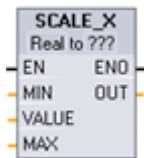
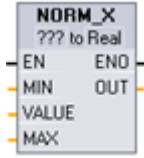
Parametre	Veri tipi	Açıklama
IN	Real, LReal	Kayan nokta girişi
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Dönüştürülmüş çıktı

Çizelge 7- 88 ENO durumu

ENO	Açıklama	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	IN, +/- INF veya +/- NaN	+/- INF veya +/- NaN

## 7.7.5 SCALE\_X (Ölçek) and NORM\_X (Normalize) komutları

Çizelge 7- 89 SCALE\_X ve NORM\_X komutları

LAD / FBD	SCL	Açıklama
	<pre>out :=SCALE_X(min:=_in_,                value:=_in_,                max:=_in_);</pre>	<p>Normalize edilmiş reel VALUE parametresini (0,0 &lt; = VALUE &lt; = 1,0 ) ölçekler; burada veri tipi ve değer aralığı MIN ve MAX parametreleri tarafından tanımlanmıştır:</p> $OUT = VALUE (MAX - MIN) + MIN$
	<pre>out :=NORM_X(min:=_in_,               value:=_in_,               max:=_in_);</pre>	<p>MIN ve MAX parametreleri tarafından belirlenen değer aralığı içindeki VALUE parametresini normalize eder:</p> $OUT = (VALUE - MIN) / (MAX - MIN),$ <p>burada ( 0.0 &lt;= OUT &lt;= 1.0 )</p>

1 LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tipini seçiniz.

Çizelge 7- 90 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
MIN	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Aralık için minimum değeri girin
VALUE	SCALE_X: Real, LReal NORM_X: SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Ölçekleme veya normalize için değer girin
MAX	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Aralık için maksimum değeri girin
OUT	SCALE_X: SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal NORM_X: Real, LReal	Ölçeklenen veya normalize edilen çıkış değeri

1 SCALE\_X için: MIN, MAX ve OUT parametreleri, aynı veri tipinden olmalıdır.

2 NORM\_X için: MIN, VALUE ve MAX parametreleri, aynı veri tipinden olmalıdır.

**Not**

Eğer VALUE parametresi 0,0'dan küçükse ya da 1,0'dan büyükse, SCALE\_X parametresi VALUE kısıtlanmalıdır (  $0,0 < = VALUE < = 1,0$  ).

- Lineer ölçekleme işlemi, OUT veri tipinin değer aralığına uygun OUT değerleri için MIN parametre değerinden düşük ya da MAX değerinden yüksek OUT çıktıları üretebilir. SCALE\_X yürütmesi, bu durumlar için ENO = TRUE ayarını yapar.
- OUT veri tipinin aralığı içinde olmayan ölçeklenmiş sayılar üretilmesi de mümkündür. Bu durumlar için OUT parametre değeri, OUT veri tipine son dönüştürme öncesinde ölçeklenen reel sayının en küçük anlamlı kısmına eşit bir orta değere ayarlanır. Bu durumda SCALE\_X yürütmesi ENO = FALSE ayarını yapar.

**NORM\_X parametresi VALUE kısıtlanmalıdır (  $MIN < = VALUE < = MAX$  )**

Eğer VALUE parametresi MIN altında ya da MAX üzerinde ise, lineer ölçekleme işlemi 0,0'dan küçük ya da 1,0'dan büyük normalize edilmiş OUT değerleri üretebilir. Bu durumda NORM\_X yürütmesi ENO = TRUE ayarını yapar.

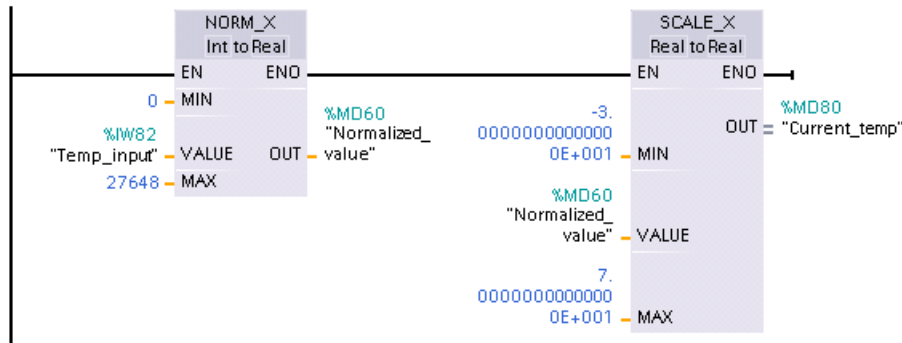
Çizelge 7- 91 ENO durumu

ENO	Durum	Sonuç OUT
1	Hata yok	Geçerli sonuç
0	Sonuç, OUT veri tipi için olan geçerli aralığı aşıyor	Ara sonuç: Bir reel sayının OUT veri tipine dönüştürülmesinden önceki en küçük anlamlı kısmı.
0	Parametreler $MAX < = MIN$	SCALE_X: VALUE Reel sayısının en küçük anlamlı kısmı, OUT boyutunu doldurur. NORM_X: VALUE veri tipindeki VALUE, bir çift word boyutunu doldurmak için genişletilir
0	Parametre VALUE = +/- INF veya +/- NaN	VALUE, OUT'a yazılır

### Örnek (LAD): Bir analog giriş değerini normalize etme ve ölçekleme

Bir analog sinyal modülünden veya akımlardaki girişi kullanan sinyal kartından gelen bir analog giriş, geçerli değerler için 0 ile 27648 aralığındadır. Diyelim ki analog girişin 0 değerinin -30 derecelik, 27648'in de 70 derecelik bir sıcaklığı temsil ettiği bir analog giriş olsun.

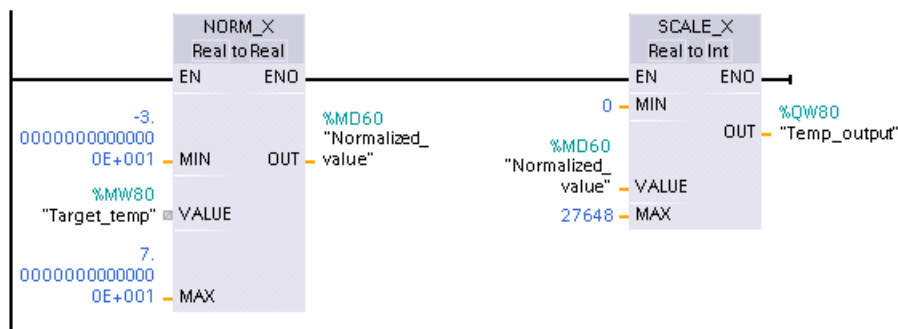
Analog değeri karşılık gelen mühendislik birimlerine dönüştürmek için girişi 0,0 ile 1,0 arasındaki bir değere normalize edin ve sonra bunu -30,0 ile 70,0 arasına ölçekleyiniz. Sonuç değer, analog giriş tarafından santigrat derece olarak simgelenen sıcaklıktır.



Eğer analog giriş bir sinyal modülünden ya da gerilim kullanan sinyal kartından alınmışsa, NORM\_X komutu için MIN değeri 0 yerine -27648 olacaktır.

### Örnek (LAD): Bir analog çıkış değerini normalize etme ve ölçekleme

Bir analog sinyal modülünde ya da sinyal kartında akım içindeki çıkış kullanılarak ayarlanacak bir analog çıkış, geçerli değerler için 0 ile 27648 aralığında olmalıdır. Diyelim ki analog çıkışın 0 değerinin -30 derecelik, 27648'in de 70 derecelik bir sıcak ayarını temsil ettiği bir analog giriş olsun. Bellekteki, -30,0 ile 70,0 arasında olan bir sıcaklığı 0 ile 27648 aralığındaki bir analog çıkış için bir değere dönüştürmek için mühendislik birimlerindeki değeri 0,0 ile 1,0 arasındaki bir değere normalize etmeli ve sonra bunu analog çıkışın aralığına ölçeklemelisiniz( 0 ile 27648):



Eğer analog çıkış bir sinyal modülü ya da gerilim kullanan sinyal kartı içinse, SCALE\_X komutu için MIN değeri 0 yerine -27648 olacaktır.

Analog giriş sunumları (sayfa 607) ve analog çıkış sunumları (sayfa 607) ile ilgili olarak hem gerilim hem de akımla ilgili ek bilgileri Teknik Özellikler bölümünde bulunabilir.

## 7.8 Program kontrol işlemleri

### 7.8.1 JMP (Jump if RLO = 1), JMPN (Jump if RLO = 0) ve Label (Jump label) komutları

Çizelge 7- 92 JMP, JMPN ve LABEL komutu

LAD	FBD	SCL	Açıklama
Label_name —{JMP}—	Label_name JMP	GOTO (Sayfa 274) komutuna bakınız.	Jump if RLO (mantık işleminin sonucu) = 1: Eğer bir JMP bobinine güç akışı varsa (LAD) ya da JMP kutusu girişi TRUE ise (FBD), o zaman program yürütmesi belirtilen etiketi takip eden ilk komutla devam eder.
Label_name —{JMPN}—	Label_name JMPN		Jump if RLO = 0: Eğer bir JMPN bobinine güç akışı yoksa (LAD) ya da JMPN kutusu girişi FALSE ise (FBD), o zaman program yürütmesi belirtilen etiketi takip eden ilk komutla devam eder.
Label_name	Label_name		Bir JMP veya JMPN atlama komutu için hedef etiketi.

- <sup>1</sup> Etiket isimlerinizi LABEL komutuna direkt olarak yazarak oluşturabilirsiniz. JMP ve JMPN etiket adı alanı için kullanılabilir etiket isimlerini seçmek için parametre yardımcısı simgesini kullanabilirsiniz. JMP veya JMPN komutuna direkt olarak da bir etiket ismi yazabilirsiniz.

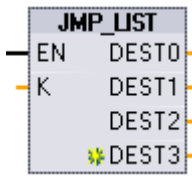
Çizelge 7- 93 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
Label_name	Label identifier (Etiket tanımlayıcı)	Atlama komutları ve karşılık gelen atlama hedef program etiketi için tanımlayıcı.

- Bir kod bloğu içindeki her etiket özgün (benzersiz) olmalıdır.
- Bir kod bloğu içinde atlama yapabilirsiniz, ancak bir kod bloğundan başka bir kod bloğuna atlama yapamazsınız.
- İleri veya geri sıçrayabilirsiniz.
- Aynı kod bloğu içinde birden fazla yerden aynı etikete sıçrayabilirsiniz.

## 7.8.2 JMP\_LIST (atlama listesini tanımla) komutu

Çizelge 7- 94 JMP\_LIST komutu

LAD / FBD	SCL	Açıklama
	<pre> CASE k OF   0: GOTO dest0;   1: GOTO dest1;   2: GOTO dest2;   [n: GOTO destn;] END_CASE; </pre>	<p>JMP_LIST komutu, program bölümlerinin yürütülmesini kontrol etmek için bir program atlama dağıtıcısı olarak davranır. K girişinin değerine bağlı olarak karşılık gelen program etiketine bir atlama meydana gelir. Program yürütmesi hedef atlama etiketini izleyen program komutları ile devam eder. Eğer K girişinin değeri etiketlerin sayısının bir eksiğini geçerse, o zaman hiçbir atlama meydana gelmez ve işlem bir sonraki program ağı ile sürer.</p>

Çizelge 7- 95 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
K	UInt	Atlama dağıtıcısı kontrol değeri
DEST0, DEST1, ..., DESTn.	Program Labels (Program Etiketleri)	<p>Belirli K parametresi değerlerine karşılık gelen atlama hedef etiketleri: Eğer K'nın değeri 0'a eşitse, o zaman DEST0 çıkışına atanan program etiketine bir atlama gerçekleştirilir. Eğer K'nın değeri 1'e eşitse, o zaman DEST1 çıkışına atanan program etiketine bir atlama gerçekleştirilir ve bu böyle devam eder. Eğer K girişinin değeri etiketlerin sayısının bir eksiğini (etiketlerin sayısı -1) geçerse, o zaman hiçbir atlama meydana gelmez ve işlem bir sonraki program ağı ile sürer.</p>

LAD ve FBD için: JMP\_LIST kutusu programınıza ilk yerleştirildiğinde iki atlama etiketi çıkışı vardır. Atlama hedefleri ekleyebilir ve silebilirsiniz.



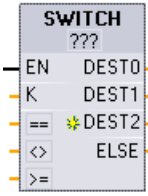
Atlama etiketleri için yeni çıkışlar eklemek için kutunun içindeki Oluştur simgesine tıklayınız (son DEST parametresinin solunda).



- Bir çıkış stub'nda sağ tıklayınız ve "Insert output" komutunu seçiniz.
- Bir çıkış stub'nda sağ tıklayınız ve "Delete" komutunu seçiniz.

### 7.8.3 SWITCH (atlama dağıtıcısı) komutu

Çizelge 7- 96 SWITCH komutu

LAD / FBD	SCL	Açıklama
	Mevcut değil	SWITCH komutu, program bölümlerinin yürütülmesini kontrol etmek için bir program atlama dağıtıcısı olarak davranır. K giriş değeri ile belirtilen karşılaştırma girişlerine atanmış değerler arasındaki karşılaştırmanın sonucuna bağlı olarak TRUE olan ilk karşılaştırma testine karşılık gelen program etiketine bir atlama meydana gelir. Eğer karşılaştırmaların hiçbiri TRUE değilse, o zaman ELSE'e atanan etikete bir atlama gerçekleşir. Program yürütmesi hedef atlama etiketini izleyen program komutları ile devam eder.

- 1 LAD ve FBD için: Kutu isminin altına tıklayınız ve açılır menüden veri tipini seçiniz.
- 2 SCL için: IF-THEN karşılaştırma dizisini kullanınız.



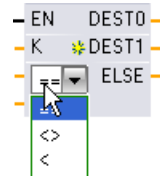
Çizelge 7- 97 Parametreler için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
K	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, TOD, Date	Ortak karşılaştırma değeri girişi
==, <>, <, <=, >, >=	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, TOD, Date	Özel karşılaştırma tipleri için ayrı karşılaştırma değeri girişleri
DEST0, DEST1, ..., DESTn. ELSE	Program Etiketleri	Belirli karşılaştırmalara karşılık gelen atlama hedefi etiketleri: K girişinin yanındaki ve altındaki karşılaştırma girişi ilk önce işlenir ve eğer K değeri ile bu giriş arasındaki karşılaştırma TRUE ise DEST0'a atanmış etikete bir atlamaya neden olur. Bir sonraki karşılaştırma testi bir sonraki aşağıdaki girişi kullanır ve eğer karşılaştırma TRUE ise DEST1'e atanan etikete bir atlamaya neden olur. Kalan karşılaştırmalar da buna benzer olarak gerçekleştirilir ve eğer karşılaştırmaların hiçbiri TRUE değilse, o zaman ELSE çıkışına atanan etikete bir atlama gerçekleştirilir.

<sup>1</sup> K girişi ve karşılaştırma girişleri (==, <>, <, <=, >, >=) aynı veri tipinden olmalıdır.

### Girişler eklemek, girişler silmek ve karşılaştırma tipleri belirlemek

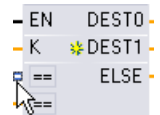
LAD veya FBD SITCH kutusu programınıza ilk yerleştirildiğinde, iki karşılaştırma kutusu vardır. Aşağıda gösterildiği gibi karşılaştırma tipleri ataması yapabilir ve atlama hedefleri ekleyebilirsiniz.



Kutunun içindeki bir karşılaştırma operatörüne tıklayınız ve açılır menü listesinden yeni bir operatör seçiniz.



Yeni karşılaştırma-hedef parametreleri eklemek için kutunun içindeki "create" (oluştur) simgesine tıklayınız (son DEST parametresinin solunda).



- Bir girişe sağ tıklayınız ve "Insert input" (Giriş ekle) komutunu seçiniz.
- Bir girişe sağ tıklayınız ve "Delete" (Sil) komutunu seçiniz.

Çizelge 7- 98 SWITCH kutusu veri tipi seçimi ve izin verilen karşılaştırma işlemleri.

Veri tipi	Karşılaştırma	Operatör yazımı
Byte, Word, DWord	Eşit	==
	Eşit değil	<>
SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal, Time, TOD, Date	Eşit	==
	Eşit değil	<>
	Büyük veya eşit	>=
	Küçük veya eşit	<=
	Büyük	>
	Küçük	<

### SWITCH kutusu değiştirme kuralları

- Hiçbir LAD/FBD komut bağlantısının karşılaştırmanın önüne geçmesine izin verilmez.
- Hiçbir ENO çıkışı yoktur, yani bir ağda sadece bir SWITCH komutuna izin verilir ve SWITCH komutu bir ağdaki son işlem olmalıdır.

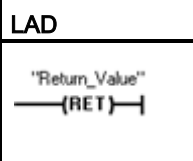
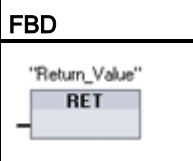
### 7.8.4 RET (Geri dönüş) talimatı

Opsiyonel RET komutu, geçerli bloğun yürütmesini sonlandırmak için kullanılır. Ancak ve ancak RET bobinine (LAD) enerji akışı varsa ya da RET kutusu girişi TRUE ise (FBD), o zaman geçerli bloğun program yürütmesi bu noktada sona erecek ve RET komutundan sonraki komutlar yürütülmeyecektir. Eğer geçerli blok bir OB ise, "Return\_Value" parametresi yok sayılır Eğer geçerli blok bir FC ya da FB ise, "Return\_Value" parametresinin değeri çağrılan kutunun ENO değeri olarak çağırılan rutine geri aktarılır.

RET komutunu bir blokta son komut olarak kullanmanız gerekmez; bu sizin için otomatik olarak yapılır. Tek bir bloğun içinde birden fazla RET komutuna sahip olabilirsiniz.

SCL için, RETURN (Sayfa 274) komutuna bakınız.

Çizelge 7- 99 Return\_Value (RET) yürütme kontrol komutu

LAD	FBD	SCL	Açıklama
		RETURN ;	Geçerli bloğun yürütmesini sonlandırır

Çizelge 7- 100 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
Return_Value	Boole	RET komutunun "Return_value" parametresi, çağırılan bloktaki blok çağrı kutusunun ENO çıkışına atanır.

RET komutunu bir FC kod bloğu içinde kullanmak için örnek adımlar:

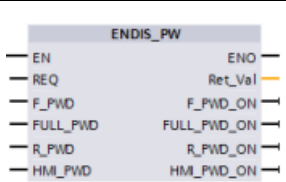
1. Yeni bir proje oluşturunuz ve bir FC ekleyiniz:
2. FC'yi düzenleyiniz:
  - Komut ağacından komutlar ekleyiniz.
  - Return\_value” (Geri dönüş değeri) parametresi için şunlardan biri de dahil olmak üzere RET komutunu ekleyiniz:
    - TRUE, FALSE veya gerekli döndürülen değeri belirten bir bellek konumu.
  - Komutlar eklemeye devam ediniz.
3. MAIN'den FC'yi çağırın [OB1].

MAIN kod bloğundaki FC kutusunun EN girişi, FC'nin yürütülmesine başlamak için TRUE olmalıdır.

RET komutu tarafından FC'de belirtilen değer MAIN kod bloğundaki FC kutusunun ENO çıkışı üzerinde, RET komutuna enerji akışının TRUE olduğu FC'nin yürütülmesini takiben sunulacaktır.

## 7.8.5 ENDIS\_PW (CPU şifrelerini etkinleştirme/ Enable/geçersiz kılma) talimatı

Çizelge 7- 101 ENDIS\_PW talimatı

LAD / FBD	SCL	Açıklama
	<pre> ENDIS_PW(   req:=_bool_in_,   f_pwd:=_bool_in_,   full_pwd:=_bool_in_,   r_pwd:=_bool_in_,   hmi_pwd:=_bool_in_,   f_pwd_on=&gt;_bool_out_,   full_pwd_on=&gt;_bool_out_,   r_pwd_on=&gt;_bool_out_,   hmi_pwd_on=&gt;_bool_out_, </pre>	<p>ENDIS_PW talimatı, istemci doğru şifreyi girebilse dahi, istemcinin bir S7-1200 CPU'suna bağlantılarına izin verebilir veya izin vermeyebilir. Bu talimat, Web sunucusu şifrelerini reddetmemelidir.</p>

Çizelge 7- 102 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	REQ=1 ise fonksiyonu gerçekleştir
F_PWD	IN	Bool	Arıza güvenlik şifresi: İzin ver (=1) veya izin vermeme (=0)
FULL_PWD	IN	Bool	Tam erişim şifresi: İzin ver (=1) veya izin vermeme (=0) tam erişim şifresi
R_PWD	IN	Bool	Okuma erişimli şifre: İzin ver (=1) veya izin vermeme (=0)
HMI_PWD	IN	Bool	HMI şifresi: İzin ver (=1) veya izin vermeme (=0)
F_PWD_ON	OUT	Bool	Arıza güvenlik şifresi durumu: İzin ver (=1) veya izin vermeme (=0)
FULL_PWD_ON	OUT	Bool	Tam erişim şifresi durumu: İzin ver (=1) veya izin vermeme (=0)

## Temel komutlar

### 7.8 Program kontrol işlemleri

Parametre ve tipi	Veri tipi	Açıklama
R_PWD_ON	OUT	Bool
HMI_PWD_ON	OUT	Bool
Ret_Val	OUT	Word

REQ=1 ile ENDIS\_PW çağırma, karşılık gelen şifre giriş parametreleri FALSE olduğunda şifre tiplerine izin vermez. Her bir şifre tipine, birbirinden bağımsız olarak izin verilebilir veya izin verilmeyebilir. Örneğin, eğer arıza güvenlik şifresine izin verilir ve diğer bütün şifrelere izin verilmez ise bu durumda küçük çalışan gruplarının CPU'ya erişimini kısıtlayabilirsiniz.

ENDIS\_PW, bir program taramasında eş zamanlı olarak yürütülür ve şifre çıkış parametreleri her zaman REQ giriş parametresinden bağımsız şifre toleransının mevcut durumunu gösterir. İzin vermek üzere girdiğiniz bütün şifreler, izin vermeme/izin verme durumuna değiştirilebilir olmalıdır. Aksi takdirde, bir hata geri döner ve ENDIS\_PW yürütmesinden önce izin verilmiş olan tüm şifrelere izin verilir. Bu ise standart CPU (Arıza güvenlik şifresinin yapılandırılmadığı yerde) F\_PWD'nin, 0'ın bir geri dönüş değerine yol açmak üzere her zaman 1'e ayarlanması gerektiği anlamına gelir.

#### Not

- HMI şifresine izin verilmezse ENDIS\_PW yürütmesi, HMI cihazlarının erişimini engelleyebilir.
- ENDIS\_PW yürütmesi öncesinde yetkilendirilmiş istemci oturumları ENDIS\_PW yürütmesi ile değişmeksizin aynen devam eder.

Bir enerjilendirmeden sonra, düzenli CPU koruma konfigürasyonunda önceden tanımlanan şifreler ile CPU erişimi sınırlandırılır. Geçerli bir şifreye izin verilememeye yeteneği, yeni bir ENDIS\_PW yürütmesi ile yeniden tesis edilmelidir. Ancak, eğer ENDIS\_PW hemen yürütülür ise ve gerekli şifrelere izin verilmemesi halinde bu durumda TIA portal erişiminde eşzamanlama yitimi olabilir. ENDIS\_PW yürütmesini geciktirmek için bir zamanlayıcı komutu kullanılabilir ve şifreler izin verilmeme durumuna gelmeden önce şifrelerin girilmesi için zaman verebilirsiniz.

#### Not

#### TIA portal haberleşmesinde eş zamanlama yitimi olan bir CPU'nun geri yüklenmesi

Bir hafıza kartı kullanılarak bir PLC dâhili yükleme belleğinin nasıl silineceği hakkındaki ayrıntılar ile ilgili "kayıp şifreden toparlanma (Sayfa 3)" başlığına bakınız.

Bir çalışma modunun hatalar, STP yürütmesi veya STEP 7 nedeniyle STOP' a (DURMA) değişmesi, korumayı iptal etmez. CPU güç döngüsü gerçekleşene kadar koruma geçerlidir. Ayrıntılar için aşağıdaki çizelgeye bakınız.

İşlem	Çalışma modu	ENDIS_PW şifre kontrolü
STEP 7'den gelen bellek resetlemesinden sonra	STOP	Active (Aktif): İzin verilmeyen şifreler, izin verilmeyen olarak devam eder.
Açma veya bir hafıza kartının değişiminden sonra	STOP	Off (Kapalı): Hiçbir şifre reddedilmez. (İzin verilmeyen hiçbir şifre yoktur.)

İşlem	Çalışma modu	ENDIS_PW şifre kontrolü
Bir program döngüsü veya OB başlatmasında ENDIS_PW yürütmesinden sonra	STARTUP, RUN	Active (Aktif): Şifrelere ENDIS_PW parametrelerine göre izin verilmez.
Çalışma modunun STP talimatı, hata, or STEP 7 vasıtasıyla RUN veya STARTUP'tan değişiminden sonra	STOP	Active (Aktif): izin verilmeyen şifreler, izin verilmeyen olarak devam eder.

**Not**


Şifre, güçlü şifreler ile CPU erişim seviyelerini korur. Güçlü şifreler en azından sekiz karakter uzunluğunda, harflerin, rakamların ve özel karakterlerin karışımından oluşur ve sözlükte bulunabilen kelimeler değildir ve bunlar ayrıca kişisel bilgilerden elde edilebilen isimler veya tanımlayıcılar olmazlar. Şifrenizi gizli tutunuz ve sıkça değiştiriniz.

## Çizelge 7- 103 Durum kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
80D0	Arıza güvenlik için şifre yapılandırılmaz.
80D1	Okuma/yazma erişimi için şifre yapılandırılmaz.
80D2	Okuma erişimi için şifre yapılandırılmaz.
80D3	HMI erişimi için şifre yapılandırılmaz.

## 7.8.6 RE\_TRIGR (yeniden başlatma (Restart) çevrimi izleme zamanı) talimatı

Çizelge 7- 104 RE\_TRIGR talimatı

LAD / FBD	SCL	Açıklama
	RE_TRIGR ( ) ;	RE_TRIGR (Tekrar tetikleme tarama zamanı gözetleyicisi), tarama döngüsü gözetleyicisi zamanlayıcısının bir hata üretmesinden önce izin verilen maksimum süreyi uzatmak için kullanılır.

Tek bir tarama döngüsü sırasında tarama döngüsü izleme zamanlayıcısını yeniden başlatmak için. Bunun etkisi, izin verilen maksimum tarama süresinin RE\_TRIGR fonksiyonunun son yürütülmesinden sonra bir maksimum döngü süresi kadar daha uzatılmasıdır.

### Not

S7-1200 CPU firmware sürümü 2.2 öncesinde, RE\_TRIGR, bir OB program döngüsünden gelen yürütme ile kısıtlanmıştır ve PLC tarama zamanını süresiz olarak uzatmak için kullanılabilir. RE\_TRIGR döngüsü bir OB başlatması, bir interrupt (kesme) veya bir OB hatasına istinaden yürütüldüğü zaman gözetleyici zamanlayıcısının resetlenmemesi ve ENO = EN demektir.

Firmware sürümü 2.2 ve sonrası için RE\_TRIGR, herhangi bir OB'ye (başlatma, kesme ve hata OB'leri dahil olmak üzere) istinaden yürütülebilir. Ancak, PLC taraması sadece konfigüre edilen en yüksek döngü süresinin azami 10 katına kadar genişletilebilir.

## PLC azami döngü süresinin ayarlanması

"Cycle time" (Döngü süresi) için Cihaz konfigürasyonundan azami tarama süresi için değeri yapılandırın

Çizelge 7- 105 Döngü süresi değerleri

Döngü süresi monitörü	Asgari değer	Azami değer	Varsayılan değer
Azami döngü süresi	1 ms	6000 ms	150 ms

## Gözetleyici zaman aşımı

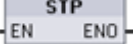
Azami tarama döngüsü zamanlayıcısı, tarama döngüsü tamamlanmadan önce sonra ererse, bir hata üretilir. Eğer kullanıcı programı, bir OB (OB 80) zaman hata kesmesi ihtiva ederse, CPU, özel bir reaksiyon oluşturmak için program lojji içerebilen OB zaman hata kesmesini yürütür.

Kullanıcı programı bir zaman hata kesmesi ihtiva etmezse, ilk zaman aşımı şartı yok sayılır ve CPU, RUN modunda kalmaya devam eder. Aynı program taramasında ikinci azami tarama süresi zaman aşımı meydana gelirse (azami döngü süresi değerinin iki katı), bu durumda bir hata tetiklenir ve bu hata CPU'nun STOP moduna geçmesine neden olur.

STOP modunda, program yürütmeleriniz durur ancak CPU sistem iletişimleri ve sistem teşhisleri devam eder.

## 7.8.7 STP (Çıkış programı) talimatı

Çizelge 7- 106 STP talimatı

LAD / FBD	SCL	Açıklama
	<code>STP ( ) ;</code>	STP komutu CPU'yu STOP moduna sokar. CPU STOP modunda iken programınızın yürütülmesi ve proses imajından gelen fiziksel güncellemeler durur.

Daha fazla bilgi için bkz: Bir RUN – STOP geçişindeki çıkışların yapılandırılması (Sayfa 3).

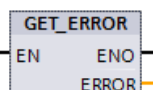
Eğer EN = TRUE ise, o zaman CPU STOP modunda geçer, program yürütmeleri durur ve ENO durumu anlamsız olur. Aksi halde EN = ENO = 0.

## 7.8.8 GET\_ERROR ve GET\_ERROR\_ID (yerel olarak hata ve hata ID al) komutları

Hata al komutları, program blok yürütme hataları hakkında bilgi sağlar Eğer bir GetError veya GetErrorID komutunu kod bloğunuza eklerseniz, program hatalarınızı program bloğunuzun içinde işleyebilirsiniz.

### GET\_ERROR

Çizelge 7- 107 GET\_ERROR talimatı

LAD / FBD	SCL	Açıklama
	<code>GET_ERROR( _out_ ) ;</code>	Bir lokal program bloğu yürütme hatasının meydana geldiğini ve önceden tanımlanmış bir hata veri yapısını ayrıntılı hata bilgileriyle doldurduğunu belirtir.

Çizelge 7- 108 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
ERROR	ErrorStruct	Hata veri yapısı: Yapıyı yeniden adlandırabilirsiniz ama yapının içindeki üyeleri yeniden adlandıramazsınız.

## Temel komutlar

## 7.8 Program kontrol işlemleri

Çizelge 7- 109 ErrorStruct veri yapısının elemanları

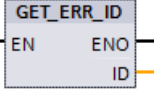
Yapı bileşenleri		Veri tipi	Açıklama																																																						
ERROR_ID		Word (kelime)	Error ID (Hata kimliği)																																																						
FLAGS		Byte (Bayt)	Bir blok çağrısı sırasında bir hatanın meydana gelip gelmediğini gösterir. <ul style="list-style-type: none"> <li>16#01: Bir blok çağrısı sırasında hata var.</li> <li>16#00: Bir blok çağrısı sırasında hata yok.</li> </ul>																																																						
REACTION		Byte (Bayt)	Varsayılan reaksiyon: <ul style="list-style-type: none"> <li>0: Yok say (yazma hatası),</li> <li>1: Yedek değer "0" ile devam et (okuma hatası),</li> <li>2: Komutu atla (sistem hatası)</li> </ul>																																																						
CODE_ADDRESS		CREF	Adres ve blok tipi hakkında bilgiler																																																						
	BLOCK_TYPE	Byte (Bayt)	Hatanın meydana geldiği bloğun tipi: <ul style="list-style-type: none"> <li>1: OB</li> <li>2: FC</li> <li>3: FB</li> </ul>																																																						
	CB_NUMBER	UInt	Kod bloğunun numarası																																																						
	OFFSET	UDInt	Dahili hafızaya başvuru																																																						
MODE		Byte (Bayt)	Erişim modu: Erişimin tipine bağlı olarak şu bilgiler çıkış olabilir:																																																						
			<table border="1"> <thead> <tr> <th>Mode</th> <th>(A)</th> <th>(B)</th> <th>(C)</th> <th>(D)</th> <th>(E)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>1</td> <td></td> <td></td> <td></td> <td></td> <td>Offset</td> </tr> <tr> <td>2</td> <td></td> <td></td> <td>Alan</td> <td></td> <td></td> </tr> <tr> <td>3</td> <td>Konum</td> <td>Kapsam</td> <td></td> <td>Numara</td> <td></td> </tr> <tr> <td>4</td> <td></td> <td></td> <td>Alan</td> <td></td> <td>Ofset</td> </tr> <tr> <td>5</td> <td></td> <td></td> <td>Alan</td> <td>DB no.</td> <td>Ofset</td> </tr> <tr> <td>6</td> <td>PtrNo. / Acc</td> <td></td> <td>Alan</td> <td>DB no.</td> <td>Ofset</td> </tr> <tr> <td>7</td> <td>PtrNo. / Acc</td> <td>Slot No. / Scope</td> <td>Alan</td> <td>DB no.</td> <td>Ofset</td> </tr> </tbody> </table>	Mode	(A)	(B)	(C)	(D)	(E)	0						1					Offset	2			Alan			3	Konum	Kapsam		Numara		4			Alan		Ofset	5			Alan	DB no.	Ofset	6	PtrNo. / Acc		Alan	DB no.	Ofset	7	PtrNo. / Acc	Slot No. / Scope	Alan	DB no.	Ofset
Mode	(A)	(B)	(C)	(D)	(E)																																																				
0																																																									
1					Offset																																																				
2			Alan																																																						
3	Konum	Kapsam		Numara																																																					
4			Alan		Ofset																																																				
5			Alan	DB no.	Ofset																																																				
6	PtrNo. / Acc		Alan	DB no.	Ofset																																																				
7	PtrNo. / Acc	Slot No. / Scope	Alan	DB no.	Ofset																																																				
OPERAND_NUMBER		UInt	Makine komutunun Operand numarası																																																						
POINTER_NUMBER_LOCATION		UInt	(A) Dahili pointer																																																						
SLOT_NUMBER_SCOPE		UInt	(B) Dahili bellekteki depolama alanı																																																						
DATA_ADDRESS		NREF	Bir operandın adresi hakkındaki bilgiler																																																						
	AREA	Byte (Bayt)	(C) Bellek alanı: <ul style="list-style-type: none"> <li>L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE</li> <li>I: 16#81</li> <li>Q: 16#82</li> <li>M: 16#83</li> <li>DB: 16#84, 85, 8A, 8B</li> </ul>																																																						



Yapı bileşenleri	Veri tipi	Açıklama
DB_NUMBER	UInt	(D) Data bloğunun numarası
OFFSET	UDInt	(E) Operandin göreceli adresi

## GET\_ERROR\_ID

Çizelge 7- 110 GetErrorID talimatı

LAD / FBD	SCL	Açıklama
	GET_ERR_ID ( ) ;	Bir program bloğu yürütme hatasının meydana geldiğini belirtir ve hatanın kimliğini (tanımlama kodu) raporlar.

Çizelge 7- 111 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
ID	Word	ErrorStruct ERROR_ID üyesi için hata tanımlayıcı değerleri

Çizelge 7- 112 Error\_ID değerleri

ERROR_ID hexadecimal (onaltılık düzende)	ERROR_ID decimal (Ondalık)	Program bloğu yürütme hatası
0	0	Hata yok
2520	9504	Bozulmuş string (dize)
2522	9506	Operand aralık dışında okuma hatası
2523	9507	Operand aralık dışında yazma hatası
2524	9508	Geçersiz alan okuma hatası
2525	9509	Geçersiz alan yazma hatası
2528	9512	Veri yerleşimi okuma hatası (FALSE bit yerleşimi)
2529	9513	Veri yerleşimi yazma hatası (FALSE bit yerleşimi)
252C	9516	Başlatılmamış pointer hatası
2530	9520	DB yazma korumalı
253A	9530	Global DB yok
253C	9532	Hatalı versiyon ya da FC yok
253D	9533	Komut yok
253E	9534	Hatalı versiyon ya da FB yok
253F	9535	Komut yok
2575	9589	Program yuvalama derinliği hatası
2576	9590	Lokal veri atama hatası
2942	10562	Fiziksel giriş noktası yok
2943	10563	Fiziksel çıkış noktası yok

## Çalışma

Varsayılan olarak CPU bir blok yürütme hatasına diyagnostik ara belleğine bir hata kaydederek yanıt verir. Ancak eğer bir kod bloğunun içine bir veya birden fazla GetError veya GetErrorID komutu yerleştirirseniz, bu blok artık blok içindeki hataları işlemek için ayarlanmış olur. Bu durumda CPU diyagnostik ara belleğindeki bir hatanın kaydını tutmaz. Bunun yerine, hata bilgileri GetError veya GetErrorID komutunun çıkışında raporlanır. GetError komutu ile ayrıntılı hata bilgilerini okuyabilir ya da GetErrorID komutu ile sadece hata tanımlayıcısını okuyabilirsiniz. Normalde ilk hata en önemlisidir. Sonrakiler ise sadece ilk hatanın sonuçlarıdır.

GetError veya GetErrorID komutunun bir blok içerisinde ilk çalıştırılması, blok yürütülmesi sırasında tespit edilen ilk hatayı verir. Bu hata GetError veya GetErrorID komutunun yürütülmesi ile bloğun başlangıcı arasında herhangi bir yerde meydana gelmiş olabilir. GetError veya GetErrorID komutunun sonraki yürütmeleri, GetError veya GetErrorID komutlarının önceki yürütülmesinden bu yana meydana gelen ilk hatayı verir. Hataların tarihçesi kaydedilmez ve her iki komutun da çalıştırılması PLC sistemini bir sonraki hatayı yakalamak üzere hazır konuma getirir.

GetError komutu tarafından kullanılan ErrorStruct veri tipi data bloğu editörüne ve blok ara yüzü editörlerine eklenebilir, böylece program lojiğiniz bu değerlere erişebilir. Bu yapıyı eklemek için veri tipi açılır menü listesinden ErrorStruct'ı seçiniz. Benzersiz isimler kullanarak çeşitli ErrorStruct elemanları oluşturabilirsiniz. Bir ErrorStruct'ın üyelerinin isimleri sonradan değiştirilemez.

## ENO tarafından belirtilen hata durumu

Eğer EN = TRUE ve GetError veya GetErrorID yürütülürse, o zaman:

- ENO = TRUE, bir kod bloğu yürütme hatasının meydana geldiğini ve hata verilerinin mevcut olduğunu belirtir
- ENO = FALSE, hiçbir kod bloğu yürütme hatasının meydana gelmediğini belirtir

Hata reaksiyon program lojiğini, bir hata meydana geldikten sonra aktive olan ENO'ya bağlayabilirsiniz. Eğer bir hata varsa, o zaman çıkış parametresi hata verilerini programınızın ona eriştiği yerde saklar.

GetError ve GetErrorID komutları, hata bilgilerinin o anda yürütülen bloktan (çağrılan blok) bir çağırılan bloğa gönderilmesi için kullanılabilir. Çağrılan bloğun son yürütme durumunu raporlamak için komutu çağrılan blok programının son ağına yerleştiriniz.

## 7.8.9 SCL program kontrol komutları

### 7.8.9.1 SCL program kontrol komutlarına genel bakış

Yapısal Kontrol dili (SCL), kullanıcı programınızı yapılandırmak için üç tip program kontrol komutu sağlar:

- Seçici komutlar: Seçici bir komut, komutların alternatif dizisinin içerisine program yürütmesine yönelmenizi mümkün kılar.
- Loops (döngüler): Tekrarlama komutlarını kullanarak döngü yürütmesini kontrol edebilirsiniz. Bir tekrarlama komutu, belirli şartlara bağlı olarak hangi program bölümlerinin tekrarlanması gerektiğini belirtir.
- Program atlamaları: Bir program atlama, belirtilen bir atlama hedefine ve dolayısıyla aynı blok içerisindeki farklı bir komuta anında bir atlama manasına gelir.

Bu program kontrol komutları, PASCAL programlama dilinin sözdizimini kullanır.

Çizelge 7- 113 SCL program kontrol komutlarının tipleri

Program kontrol komutu	Açıklama	
Selective (Seçici)	IF-THEN komutu (Sayfa 268)	Durumun TRUE veya FALSE olmasına bağlı olarak program yürütmesini iki alternatif daldan birine yönlendirmenizi mümkün kılar.
	CASE komutu (Sayfa 269)	Değişken değer bağlı olarak, <i>n alternatif</i> daldan 1'ine seçici yürütmeyi mümkün kılar.
Loop (Döngü)	FOR komutu (Sayfa 270)	Kontrol değişkeni belirtilen değer aralığı içerisinde kalmaya devam ettiği sürece komut dizisini tekrarlar.
	WHILE-DO komutu (Sayfa)	Bir yürütme şartının karşılanması gerekirken komut dizisini tekrarlar.
	REPEAT-UNTIL komutu (Sayfa)	Bir sonlanma şartı karşılanana kadar komut dizisini tekrarlar.
Program jump (Program atlama)	CONTINUE ifadesi (Sayfa 273)	Mevcut döngü tekrarının yürütmesini durdurur.
	EXIT komutu (Sayfa 273)	Bir sonlanma şartı karşılanıp veya karşılanmadığına bakılmaksızın herhangi bir noktadaki döngüyü çıkarır
	GOTO komutu (Sayfa 274)	Programın belirtilen bir etikete anında atlamasına neden olur
	RETURN komutu (Sayfa 274)	Programın, halihazırda yürütülmekte olduğu bloktan çıkmasına ve çağırılan bloğa dönmesine neden olur.

### 7.8.9.2 IF-THEN komutu

IF-THEN komutu, mantıksal (lojik) bir ifadenin Boole değerinin değerlendirilmesine bağlı olarak bir komut grubunun yürütülmesi ile program akışını kontrol eden şartlı bir komuttur. Ayrıca, çoklu IF-THEN komutlarının yürütülmesinin yapılandırmak veya yuvarlatmak için araçları kullanabilirsiniz.

Çizelge 7- 114 IF-THEN komutunun elemanları

SCL	Açıklama
<pre>IF "condition" THEN     statement_A;     statement_B;     statement_C; ;</pre>	<p>If "şartı", TRUE veya 1 ise bu durumda END_IF komutu ile karşılaşılana kadar aşağıdaki komutlar yürütülür.</p> <p>If "şartı", FALSE veya 0 ise bu durumda END_IF komutu atlanır (program, isteğe bağlı ELSIF veya ELSE komutlarını içermedikçe)</p>
<pre>[ELSIF "condition-n" THEN     statement_N; ;]</pre>	<p>İsteğe bağlı ELSIF<sup>1</sup> komutu, değerlendirilmesi gereken ilave şartları sağlar. Örneğin: IF-THEN komutunun içindeki "şart", FALSE ise bu durumda program "condition-n" (şart-n)'yi değerlendirir. "condition-n" nin TRUE olması halinde bu durumda "statement_N" yürütülür.</p>
<pre>[ELSE     statement_X; ;]</pre>	<p>İsteğe bağlı ELSE komutu, IF-THEN komutunun içindeki "şart", FALSE olduğunda değerlendirilmesi gereken ilave şartları sağlar.</p>
<pre>END_IF;</pre>	<p>END_IF komutu, IF-THEN komutunu sonlandırır.</p>

<sup>1</sup> Bir IF-THEN komutu içerisindeki çoklu ELSIF komutlarını dahil edebilirsiniz.

Çizelge 7- 115 IF-THEN komutu için değişkenler

Değişkenler	Açıklama
"condition"	Gerekli. Lojik ifadesi ya TRUE (1) ya da FALSE (0)'dir.
"statement_A"	İsteğe bağlı. "şart", TRUE olduğunda bir veya daha fazla komutun yürütülmesi gerekir.
"condition-n"	İsteğe bağlı. Lojik ifadesi ifadesinin, isteğe bağlı ELSIF komutu ile değerlendirilmesi gerekir.
"statement_N"	İsteğe bağlı. ELSIF komutunun "şart-n" si TRUE olduğunda bir veya daha fazla komutun yürütülmesi gerekir.
"statement_X"	İsteğe bağlı. IF-THEN komutunun "şart" ı FALSE olduğunda bir veya daha fazla komutun yürütülmesi gerekir.

Bir IF komutu, aşağıdaki kurallara göre yürütülür.

- Lojik ifadesi = TRUE olan komutların ilk dizisi yürütülür. Komutların kalan dizileri yürütülmez.
- Boole ifadesi = TRUE ise, ELSE ile başlayan komutların dizileri yürütülür (ELSE kolu mevcut olmadığında komutların hiçbir dizisi)
- ELSIF komutlarının herhangi bir sayısı mevcut olabilir.

#### Not

Bir veya daha fazla ELSIF kolunun, geçerli bir ifadeyi takip eden lojik ifadelerin, IF komutlarının bir dizisinin aksine daha fazla değerlendirilmemesi avantajı vardır. Programın işleyiş süresi bundan dolayı azaltılabilir.

### 7.8.9.3 CASE komutu

Çizelge 7- 116 CASE komutunun elemanları

SCL	Açıklama
<pre> CASE "Test_Value" OF   "ValueList": Statement[; Statement, ...]   "ValueList": Statement[; Statement, ...] [ELSE Else-statement[; Else-statement, ...]] END CASE;</pre>	<p>CASE komutu, bir ifadenin değerine bağlı olarak komutların birkaç grubunun birini yürütür.</p>

Çizelge 7- 117 Parametreler

Parametre	Açıklama
"Test_Value"	Gerekli. Int veri tipinin herhangi bir nümerik ifadesi
"ValueList"	Gerekli. Tek bir değer veya değerlerin veya değerlerin aralığının virgülle ayrılmış listesi. (bir değer aralığını tanımlamak için iki periyot kullanılır: 2..8) Aşağıdaki örnekte, değer listesinin farklı değişkenleri gösterilmiştir: 1: Statement_A; 2, 4: Statement_B; 3, 5..7,9: Statement_C;
Statement	Gerekli. "Test_Value", değer listesinde herhangi değerle eşleştiği zaman, bir veya daha fazla komut yürütülür.
Else-statement	İsteğe bağlı. Belirtilen "Test_Value"nin değeri hiçbir eşleşme olmaması halinde , bir veya daha fazla komut yürütülür.

Bir CASE komutu, aşağıdaki kurallara göre yürütülür.

- Test\_value ifadesi, Int veri tipinin bir değerine geri dönmelidir.
- Bir CASE komutu işlem gördüğü zaman, program, Test\_value ifadesinin değerinin, belirtilen değerler listesi içerisinde olup olmadığını kontrol eder. Eşleşme bulunması halinde, bu listeye atanan komut bileşeni yürütülür.
- Eşleşme bulunmaması halinde, ELSE'yi takip eden program bölümü yürütülür veya ELSE kolunun mevcut olmaması halinde hiçbir komut yürütülmez.

CASE komutları iç içe düzenlenebilir (yuvalanabilir). Her bir iç içe düzenlenen case komutu END\_CASE komutu ile ilişkili olmalıdır.

```

CASE "var1" OF
  1 : #var2 := 'A';
  2 : #var2 := 'B';
ELSE
  CASE "var3" OF

    65..90: #var2 := 'UpperCase';
    97..122: #var2 := 'LowerCase';

  ELSE

    #var2:= 'SpecialCharacter';

  END_CASE;
END_CASE;

```

#### 7.8.9.4 FOR komutu

Çizelge 7- 118 FOR komutunun elemanları

SCL	Açıklama
<pre> FOR "control_variable" := "begin" TO "end" [BY "increment"] DO   statement; ; END_FOR; </pre>	<p>Bir FOR komutu, kontrol değişkeni, belirtilen değer aralığının içerisinde olduğu sürece, bir komut dizisinin tekrarlanması için kullanılır. FOR ile bir döngünün tanımı, bir başlangıç ve bir bitiş değerinin özelliğini içerir. Her iki değer ise kontrol değişkeni olarak aynı tipte olmalıdır.</p> <p>FOR döngülerini iç içe düzenleyebilirsiniz. END_FOR komutu, en son yürütülen FOR komutuna başvurur.</p>

Çizelge 7- 119 Parametreler

Parametre	Açıklama
"control_variable"	Gerekli. Bir döngü sayıcısı olarak hizmet veren bir tamsayı (Int or DInt)
"begin"	Gerekli. Kontrol değişkenlerinin başlangıç değerini belirten basit ifade
"end"	Gerekli. Kontrol değişkenlerinin nihai değerini belirleyen basit ifade
"increment"	İsteğe bağlı. "Kontrol değişkeninin" (Control variable) her bir döngüden sonra değiştiği miktardır. "Artış" (increment), "kontrol değişkeni olarak aynı veri tipine sahiptir. "Artış" (increment) değerinin belirtilmemesi halinde, çalıştırma etiketlerinin değeri, her bir döngüden sonra 1 ile artırılabilecektir. "Artış" (increment)'ı FOR komutunun yürütmesi sırasında değiştiremezsiniz.

FOR komutu, aşağıdaki gibi yürütülür:

- Döngünün başlamasında, kontrol değişkeni başlangıç değerine (başlangıç ataması) ayarlanır ve her defasında bu döngü tekrarlar, nihai değer ulaşılan kadar belirtilen artış (pozitif artış) ile veya azaltma (negatif artış) ile artırılır.
- Döngünün her bir çalıştırmasını müteakiben, bu hususun karşılanıp karşılanmadığını tespit etmek için şart kontrol edilir (nihai değere ulaşma). Bitiş şartının karşılanmaması halinde komutların dizisi tekrar yürütülür, aksi takdirde, döngü sonlanır ve yürütme, bu döngüyü anında takip eden komutla devam eder.

FOR komutlarını formülize eden kurallar aşağıdadır:

- Kontrol değişkeni yalnızca bir Int veya DInt veri tipi olabilir.
- Komut BY (artış)'yi yoksayabilirsiniz. Artış belirtilmemesi halinde bunu otomatik olarak +1 olacağı varsayılır. .

“Şart” ifadesinin durumuna bakılmaksızın döngüyü bitirmek için, EXIT komutu kullanılır (Sayfa 3). EXIT komutu, END\_FOR komutunun hemen ardından gelen komutu yürütür.

Bir FOR döngüsünün ardından gelen komutları atlamada ve sonlandırma için şartın karşılanıp karşılanmadığının incelenmesi ile birlikte döngüyü devam ettirmede CONTINUE komutu kullanılır.

### 7.8.9.5 WHILE-DO komutu

Çizelge 7- 120 WHILE komutu

SCL	Açıklama
<pre>WHILE "condition" DO     Statement;     Statement;     ...; END_WHILE;</pre>	<p>WHILE komutu, verilen bir şart TRUE olana kadar komutların bir dizisini gerçekleştirir.</p> <p>WHILE döngülerini iç içe düzenleyebilirsiniz. END_WHILE komutu, en son yürütülen WHILE komutuna başvurur.</p>

Çizelge 7- 121 Parametreler

Parametre	Açıklama
"condition"	Gerekli. TRUE veya FALSE için değerlendirme yapan bir mantıksal (lojik) ifadedir. (Bir "null" (boş) şartı, FALSE olarak yorumlanır.
Statement	İsteğe bağlı. Bu şart TRUE olarak değerlendirilene kadar bir veya daha fazla komut yürütülür.

#### Not

WHILE komutu, herhangi bir komutu yürütmeden önce “şart”ın durumunu değerlendirir. “şart”ın durumuna bakılmaksızın en az bir kez komutları yürütmek için, REPEAT komutu kullanılır (Sayfa 3).

WHILE komutu, aşağıdaki kurallara göre yürütülür:

- Her bir döngü gövdesi tekrarı öncesinde, yürütme şartı değerlendirilir.
- Yürütme şartı, "TRUE" değerine sahip olduğu sürece DO'yu takip eden döngü gövdesi tekrarlanır.
- FALSE değeri bir kez gerçekleştiğinde, döngü atlanır ve döngüyü takip eden komut yürütülür.

"Şart" ifadesinin durumuna bakılmaksızın döngüyü bitirmek için, EXIT komutu kullanılır (Sayfa 3). EXIT komutu, END\_WHILE komutunun hemen ardından gelen komutu yürütür.

Bir WHILE döngüsünün ardından gelen komutları atlamada ve sonlandırma için şartın karşılanıp karşılanmadığının incelenmesi ile birlikte döngüyü devam ettirmede CONTINUE komutu kullanılır.

### 7.8.9.6 REPEAT-UNTIL komutu

Çizelge 7- 122 REPEAT komutu

SCL	Açıklama
<pre> REPEAT     Statement; ; UNTIL "condition" END_REPEAT;</pre>	<p>REPEAT komutu, verilen bir şart TRUE olana kadar komutların bir grubunu yürütür.</p> <p>REPEAT döngülerini iç içe düzenleyebilirsiniz. END_REPEAT komutu, en son yürütülen Repeat komutuna başvurur</p>

Çizelge 7- 123 Parametreler

Parametre	Açıklama
Statement	İsteğe bağlı. Şart, TRUE olana kadar bir veya daha fazla komut yürütülür
"condition"	Gerekli. Aşağıdaki yollara ait bir veya daha fazla ifade: TRUE veya FALSE için değerlendirme yapan sayısal bir ifade veya bir dize ifadesi. Bir "null" (boş) şartı, FALSE olarak yorumlanır.

#### Not

Şartın durumu değerlendirilmeden önce, REPEAT komutu, döngünün ilk tekrarı sırasındaki ("şart", FALSE olsa bile) komutları yürütür. Bu komutları yürütmeden önce şartın durumunu gözden geçirmek için, WHILE komutu kullanılır (Sayfa 3).

"Şart" ifadesinin durumuna bakılmaksızın döngüyü bitirmek için, EXIT komutu kullanılır (Sayfa 3). EXIT komutu, END\_REPEAT komutunun hemen ardından gelen komutu yürütür.

Bir REPEAT döngüsünün ardından gelen komutları atlamada ve sonlandırma için şartın karşılanıp karşılanmadığının incelenmesi ile birlikte döngüyü devam ettirmede CONTINUE komutu kullanılır.



### 7.8.9.7 CONTINUE komutu

Çizelge 7- 124 CONTINUE komutu

SCL	Açıklama
<code>CONTINUE Statement; ;</code>	CONTINUE komutu, Bir program döngüsünün (FOR, WHILE, REPEAT) ardından gelen komutları atlar ve sonlandırma için şartın karşılanıp karşılanmadığının incelenmesi ile birlikte döngüyü devam ettirir. Bu durum gerçekleşmemişse (sonlandırma için şart), döngü devam eder.

CONTINUE komutu, aşağıdaki kurallara göre yürütülür:

- Bu komut, döngü gövdesi yürütmesini anında sonlandırır.
- Döngünün tekrarlanması için şartın karşılanıp karşılanmadığına bağlı olarak gövde tekrar yürütülür veya tekrarlama komutundan çıkılır ve ardından gelen komut yürütülür.
- FOR komutu için, kontrol değişkeni, bir CONTINUE komutunun hemen sonrasında belirtilen artışla artırılır.

CONTINUE komutu, sadece bir döngü içerisinde kullanılır. İç içe düzenlenmiş döngülerde, CONTINUE, hemen bunu içeren bir döngüye başvurur. CONTINUE genellikle bir IF komutu ile bağlantılı olarak kullanılır.

Döngüden, sonlandırma deneyine bakılmaksızın çıkış yapılması gerekmesi halinde EXIT komutu kullanılır.

Aşağıdaki örnekte, bir değer yüzdesi hesaplandığı zaman sıfıra bölme hatasından kaçınmak için CONTINUE komutunun kullanılması gösterilmiştir:

```
FOR i := 0 TO 10 DO  
  IF value[i] = 0 THEN CONTINUE; END_IF;  
    p := part / value[i] * 100;  
    s := INT_TO_STRING(p);  
    percent := CONCAT(IN1:=s, IN2:="%");  
END_FOR;
```

### 7.8.9.8 EXIT komutu

Çizelge 7- 125 EXIT komutu

SCL	Açıklama
<code>EXIT;</code>	Bir EXIT komutu, sonlandırma şartının karşılanıp karşılanmadığına bakılmaksızın, herhangi bir noktada bir döngüden (FOR, WHILE, REPEAT) çıkmak için kullanılır.

EXIT komutu, aşağıdaki kurallara göre yürütülür:

- Bu komut, anında çıkılması gereken çıkış komutunun hemen yanındaki komutun tekrarlamasına neden olur.
- Programın yürütülmesi, döngünün bitiminden sonra devam eder (örneğin END\_FOR sonrasında) EXIT komutu, sadece bir döngü içerisinde kullanılır.

İç içe düzenlenmiş döngülerde, EXIT komutu, bir sonraki en yüksek iç içe düzenlenmiş seviyenin işlenmesine geri döner.

```
FOR i := 0 TO 10 DO
CASE value[i, 0] OF
  1..10: value [i, 1]:="A";
  11..40: value [i, 1]:="B";
  41..100: value [i, 1]:="C";
ELSE
EXIT;
END_CASE;
END_FOR;
```

### 7.8.9.9 GOTO komutu

Çizelge 7- 126 GOTO komutu

SCL	Açıklama
<pre>GOTO JumpLabel; Statement; ... ; JumpLabel: Statement;</pre>	<p>GOTO komutu, aynı bloktaki bir etikete sıçrayarak komutlar üzerinden atlar. Atlama etiketi ("JumpLabel") ve GOTO komutu aynı blokta olmalıdır. Bir atlama etiketinin adı, bir blok içerisinde sadece bir kez atanabilir. Her bir atlama etiketi, birkaç GOTO komutunun hedefi olabilir.</p>

Bir döngü bölümüne FOR, WHILE veya REPEAT) atlama mümkün değildir. Bir döngü içerisinde atlama mümkündür.

Takip eden örnekte: "Tag\_value" operandının değerine bağlı olarak programın yürütmesi, karşılık gelen atlama etiketi ile tanımlanan bir noktada sürdürülür. Tag\_value" nun 2'ye eşit olması halinde program yürütmesi, "MyLabel2" atlama etiketinde sürdürülür ve "MyLabel1" atlanır.

```
CASE "Tag_value" OF
  1 : GOTO MyLabel1;
  2 : GOTO MyLabel2;
ELSE GOTO MyLabel3;
END_CASE;
MyLabel1: "Tag_1" := 1;
MyLabel2: "Tag_2" := 1;
MyLabel3: "Tag_4" := 1;
```

### 7.8.9.10 RETURN komutu

Çizelge 7- 127 RETURN komutu

SCL	Açıklama
<pre>RETURN;</pre>	<p>RETURN komutu, şartlar olmaksızın yürütülmekte olan kod bloğunu çıkış yapar. Program yürütmesi, çağırılan bloğa veya işletim sistemine (bir OB'den çıktığında) geri döner.</p>

RETURN komutunun bir örneği:

```
IF "Error" <> 0 THEN
RETURN;
END_IF;
```

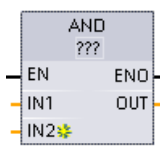
### Not

En son komutun yürütmesinden sonra, kod bloğu otomatik olarak çağırılan bloğa geri döner. Bu bloğun bitiminde RETURN komutunu eklemeyin.

## 7.9 Word lojik işlemleri

### 7.9.1 AND, OR ve XOR lojik işlem komutları

Çizelge 7- 128 AND, OR ve XOR lojik işlem komutları

LAD / FBD	SCL	Açıklama
	<code>out := in1 AND in2;</code>	AND: Mantıksal AND
	<code>out := in1 OR in2;</code>	OR: Mantıksal OR
	<code>out := in1 XOR in2;</code>	XOR: Mantıksal özel OR

1 LAD ve FBD için: "???" üzerine tıklayınız ve açılır menüsünden veri tipini seçiniz.



Bir girişi eklemek için "Create" (Oluştur) simgesine tıklayınız ya da mevcut IN parametrelerinden biri için bir girişine sağ tıklayınız ve "Insert input" (Giriş ekle) komutunu seçiniz.

Bir girişi kaldırmak için, mevcut IN parametrelerinden birinin üzerine sağ tıklayınız (orijinal iki girişten fazlası olduğunda) ve "Delete" (Sil) komutunu seçiniz.

Çizelge 7- 129 Parametreler için veri tipleri


Parametre	Veri tipi	Açıklama
IN1, IN2	Byte, Word, DWord	Mantıksal girişler
OUT	Byte, Word, DWord	Mantıksal çıkış

1 Veri tipi seçimi IN1, IN2 ve OUT parametrelerini aynı tipe ayarlar.

IN1 ve IN2'nin karşılıklı gelen bit değerleri OUT parametresinde bir ikili lojik sonucu oluşturmak için birleştirilir. Bu komutların yürütülmesini takiben ENO her zaman TRUE'dur.

## 7.9.2 INV (birin tamamlayıcısını oluştur) komutu

Çizelge 7- 130 INV komutu

LAD / FBD	SCL	Açıklama
	Mevcut değil	IN parametresinin ikili birin tamamlayıcısını hesaplar. Birin tamamlayıcısı, IN parametresinin her bir bit değerinin tersine çevrilmesiyle oluşturulur (her 0 1'e ve her 1 de 0'a çevrilir). Bu komutun yürütülmesini takiben ENO her zaman TRUE'dur.

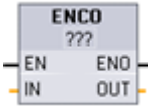
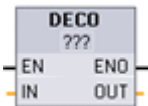
<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüsünden veri tipini seçiniz.

Çizelge 7- 131 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord	Ters çevirmek için veri elemanı
OUT	SInt, Int, DInt, USInt, UInt, UDInt, Byte, Word, DWord	Ters çevrilen çıkış

## 7.9.3 DECO (Kodla) ENCO (Kod çöz) komutları

Çizelge 7- 132 ENCO ve DECO komutu

LAD / FBD	SCL	Açıklama
	<code>out := ENCO(_in_);</code>	Bir bit paternini bir ikili sayıya kodlar ENCO komutu IN parametresini, IN parametresinin en küçük anlamlı setinin bit konumuna karşılık gelen ikili sayıya dönüştürür ve sonucu OUT parametresine döndürür. Eğer IN parametresi 0000 0001 veya 0000 0000 ise, o zaman OUT parametresine 0 değeri döndürülür. Eğer IN parametresi 0000 0000 değerine sahipse ENO FALSE olarak ayarlanır.
	<code>out := DECO(_in_);</code>	Bir ikili sayıyı bir bit paternine deşifre eder (çözer) DECO komutu IN parametresinden gelen bir ikili sayıyı, OUT parametresindeki karşılık gelen bit konumunu 1'e ayarlayarak (tüm diğer bitler 0'dır) deşifre eder. DECO komutunun yürütülmesini takiben ENO her zaman TRUE'dur. Not: DECO komutu için varsayılan veri tipi DWORD'tur. SCL'de bir byte veya Word değerini çözmek için komutun adını DECO_BYTE veya DECO_WORD'a değiştirin ve bir byte'ı veya word etiketini veya adresi atayın.

<sup>1</sup> LAD ve FBD için : “???” üzerine tıklayınız ve açılır menüsünden veri tipini seçiniz.

Çizelge 7- 133 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	ENCO: Byte, Word, DWord DECO: UInt	ENCO: Kodlanacak bit paterni DECO: kodu çözülecek değer
OUT	ENCO: Int DECO: Byte, Word, DWord	ENCO: Kodlanacak değer DECO: Deşifre (çözülen) edilen değer

Çizelge 7- 134 ENO durumları

ENO	Şart	Sonuç (OUT)
1	Hata yok	Geçerli bit numarası
0	IN sıfırdır	OUT sıfır olarak ayarlanır

Bir Bayt, Word veya DWord'ün DECO parametresi OUT veri tipi seçimi, IN parametresinin kullanılabilir aralığını kısıtlar. Eğer IN parametresinin değeri kullanılabilir aralığı aşarsa, o zaman bir modulo işlemi gerçekleştirilerek en küçük anlamlı bitler aşağıda gösterilir.

DECO parametresi IN aralığı:

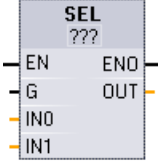
- 3 bit (değerler 0-7) IN, bir Bayt OUT'ta 1 bit konumunu ayarlamak için kullanılır
- 4 bit (değerler 0-15) IN, bir Word OUT'ta 1 bit konumunu ayarlamak için kullanılır
- 5 bit (değerler 0-31) IN, bir DWord OUT'ta 1 bit konumunu ayarlamak için kullanılır

Çizelge 7- 135 Örnekler

DECO IN değeri			DECO OUT değeri (Tek bir konumunu deşifre et)
Byte OUT 8 bit	Min. IN	0	00000001
	Maks. IN	7	10000000
Word OUT 16 bit	Min. IN	0	0000000000000001
	Maks. IN	15	1000000000000000
DWord OUT 32 bit	Min. IN	0	00000000000000000000000000000001
	Maks. IN	31	10000000000000000000000000000000

### 7.9.4 SEL (Seç), MUX (Çokla), and DEMUX (Çoklama çöz) komutları

Çizelge 7- 136 SEL (seç) komutu

LAD / FBD	SCL	Açıklama
	<pre>out := SEL(   g:=_bool_in,   in0:=_variant_in,   in1:=_variant_in);</pre>	SEL, G parametresinin değerine göre OUT parametresine iki giriş değerinden birini atar.

<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüsünden veri tipini seçiniz.

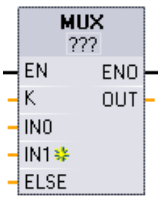
Çizelge 7- 137 SEL komutu için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
G	Bool	<ul style="list-style-type: none"> <li>• 0, IN0’i seçer</li> <li>• 1, IN1’i seçer</li> </ul>
IN0, IN1	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Girişler
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Çıkış

<sup>1</sup> Giriş değişkenleri ve çıkış değişkenleri, aynı veri tipinden olmalıdır.

**Şart kodları:** SEL komutunun yürütülmesini takiben ENO her zaman TRUE’dur.

Çizelge 7- 138 MUX (çokla) komutu

LAD / FBD	SCL	Açıklama
	<pre>out := MUX(   k:=_unit_in,   in1:=variant_in,   in2:=variant_in,   [...in32:=variant_in,]   inelse:=variant_in);</pre>	MUX, K parametresinin değerine göre OUT parametresine çeşitli giriş değerlerinden birini kopyalar. Eğer K parametresinin değeri (INn – 1) üzerinde ise, o zaman OUT parametresine ELSE parametresinin değeri kopyalanır.

<sup>1</sup> LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüsünden veri tipini seçiniz.



Bir giriş eklemek için "Create" (Oluştur) simgesine tıklayınız ya da mevcut IN parametrelerinden biri için bir girişe sağ tıklayınız ve "Insert input" (Giriş ekle) komutunu seçiniz.

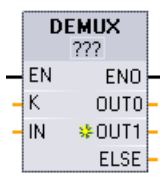
Bir girişi kaldırmak için, mevcut IN parametrelerinden biri için üzerine sağ tıklayınız (orijinal iki girişten fazlası olduğunda) ve "Delete"(Sil) komutunu seçiniz.

Çizelge 7- 139 MUX komutu için veri tipleri

Parametre	Veri tipi	Açıklama
K	UInt	<ul style="list-style-type: none"> <li>0, IN0'ı seçer</li> <li>1, IN1'i seçer</li> <li>n, INn'yi seçer</li> </ul>
IN0, IN1, .. INn	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Girişler
ELSE	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Giriş yedek değeri (opsiyonel)
OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Çıkış

<sup>1</sup> Giriş değişkenleri ve çıkış değişkeni aynı veri tipinden olmalıdır.

Çizelge 7- 140 DEMUX (Çoklama çöz) komutu

LAD / FBD	SCL	Açıklama
	<pre> DEMUX (     k:=_unit_in,     in:=variant_in,     out1:=variant_in,     out2:=variant_in,      [...out32:=variant_in,]      outelse:=variant_in);         </pre>	<p>DEMUX, IN parametresine atanan konum değerini çeşitli çıkışlardan birine atar. K parametresinin değeri, hangi çıkışın IN değerinin hedefi olarak seçileceğini belirler Eğer K değeri (OUTn – 1) üzerinde ise, o zaman IN değeri ELSE parametresine atanan konuma kopyalanır.</p>

<sup>1</sup> LAD ve FBD için: "???" üzerine tıklayınız ve açılır menüsünden veri tipini seçiniz.



Bir çıkış eklemek için "Create" (Oluştur) simgesine tıklayınız ya da mevcut OUT parametrelerinden biri için bir çıkışa sağ tıklayınız ve "Insert output" (Çıkış ekle) komutunu seçiniz.

Bir çıkışı kaldırmak için, mevcut OUT parametrelerinden biri için üzerine sağ tıklayınız (orijinal iki çıkıştan fazlası olduğunda) ve "Delete" (Sil) komutunu seçiniz.

## Temel komutlar

### 7.9 Word lojik işlemleri

Çizelge 7- 141 DEMUX komutu için veri tipleri

Parametre	Veri tipi <sup>1</sup>	Açıklama
K	UInt	Seçici değer: <ul style="list-style-type: none"> <li>• 0, OUT0'ı seçer</li> <li>• 1, OUT1'i seçer</li> <li>• n, OUTn'i seçer</li> </ul>
IN	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Giriş
OUT0, OUT1, .. OUTn	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	Çıktılar
ELSE	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal, Byte, Word, DWord, Time, Char	K, (OUTn – 1) üzerinde olduğundaki yedek çıkış

<sup>1</sup> Giriş değişkeni ve çıkış değişkenleri, aynı veri tipinden olmalıdır.

Çizelge 7- 142 MUX ve DEMUX komutları için ENO durumu


ENO	Şart	Sonuç OUT
1	Hata yok	MUX: Seçilen IN değeri OUT'a kopyalanır DEMUX: IN değeri seçilen OUT'a kopyalanır
0	MUX: K, girişlerin sayısı - 1'den büyük	<ul style="list-style-type: none"> <li>• Hiçbir ELSE sunulmuştur: OUT değiştirilmemiştir,</li> <li>• ELSE sunulmuştur, ELSE değeri OUT'a atanmıştır</li> </ul>
	DEMUX: K, girişlerin sayısı - 1'den büyük	<ul style="list-style-type: none"> <li>• Hiçbir ELSE sunulmuştur: çıktılar değişmemiştir,</li> <li>• ELSE sunulmuştur, IN değeri ELSE'e kopyalanmıştır</li> </ul>



## 7.10 Kaydır ve Döndür

### 7.10.1 SHR (Sağa kaydır) ve SHL (Sola kaydır) komutları

Çizelge 7- 143 SHR ve SHL komutları

LAD / FBD	SCL	Açıklama
	<pre> out := SHR(   in:=_variant_in_,   n:=_uint_in); out := SHL(   in:=_variant_in_,   n:=_uint_in);           </pre>	IN parametresinin bit paternini kaydırmak için kaydır komutlarını (SHL ve SHR) komutlarını kullanın. Sonuç, OUT parametresine atanır. N parametresi, kaydırılan bit konumlarının sayısını belirtir: <ul style="list-style-type: none"> <li>• SHR: Bit paternini sağa kaydır</li> <li>• SHL: Bit paternini sola kaydır</li> </ul>

1 LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tiplerini seçiniz.

Çizelge 7- 144 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	Byte, Word, DWord	Kaydırılacak bit paterni
N	UInt	Kaydırılacak bit konumlarının sayısı
OUT	Byte, Word, DWord	Kaydırma işleminden sonraki bit paterni

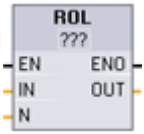
- N=0 için, hiçbir kaydırma yapılmaz. IN değeri OUT’a atanır.
- Kaydırma işlemi tarafından boşaltılan bit komutlarına sıfır atanır.
- Eğer kaydırma sayısı (N) hedef değerdeki bitlerin sayısını aşarsa (Bayt için 8, Word için 16, DWord için 32), bu durumda tüm orijinal bit değerleri kaydırılır ve yerlerine sıfır getirilir (OUT’a sıfır atanır).
- Kaydırma işlemleri için ENO her zaman TRUE’dur.

Çizelge 7- 145 Word verileri için SHL örneği

Bir Word’ün bitlerini, sağdan sıfırlar ekleyerek sola kaydırın (N = 1)			
IN	1110 0010 1010 1101	İlk kaydırmadan önceki OUT değeri:	1110 0010 1010 1101
		İlk sola kaydırmadan sonra:	1100 0101 0101 1010
		İkinci sola kaydırmadan sonra:	1000 1010 1011 0100
		Üçüncü sola kaydırmadan sonra:	0001 0101 0110 1000

## 7.10.2 ROR (Sağa döndür) ve ROL (Sola döndür) komutları

Çizelge 7- 146 ROR ve ROL komutları

LAD / FBD	SCL	Açıklama
	<pre> out := ROL(   in:=_variant_in_,   n:=_uint_in_); out := ROR(   in:=_variant_in_,   n:=_uint_in_); </pre>	<p>IN parametresinin bit paternini döndürmek için döndür komutlarını (ROR ve ROL) kullanın. Sonuç, OUT parametresine atanır. N parametresi, döndürülen bit konumlarının sayısını tanımlar.</p> <ul style="list-style-type: none"> <li>• ROR: Bit paternini sağa döndür</li> <li>• ROL: Bit paternini sola döndür</li> </ul>

1 LAD ve FBD için: “???” üzerine tıklayınız ve açılır menüden veri tiplerini seçiniz.

Çizelge 7- 147 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	Byte, Word, DWord	Döndürülecek bit paterni
N	UInt	Döndürülecek bit konumlarının sayısı
OUT	Byte, Word, DWord	Döndürme işleminden sonraki bit paterni

- N=0 için, hiçbir döndürme yapılmaz. IN değeri OUT'a atanır.
- Bit verileri, hedef değer bir tarafından hedef değer diğer tarafına döndürülür, böylece hiçbir orijinal bit değeri kaybolmaz.
- Eğer döndürülecek bit konumlarının sayısı (N) hedef değerdeki bitlerin sayısını aşıyorsa (Bayt için 8, Word için 16, DWord için 32), bu durumda döndürme yine de gerçekleştirilir.
- Döndürme komutlarının yürütülmesini takiben ENO her zaman TRUE'dur.

Çizelge 7- 148 Word verileri için ROR örneği

Bitleri, sağ taraftan sol tarafa döndürme (N = 1)			
IN	0100 0000 0000 0001	İlk döndürmeden önceki OUT değeri:	0100 0000 0000 0001
		İlk sağa döndürmeden sonrası:	1010 0000 0000 0000
		İkinci sağa döndürmeden sonrası:	0101 0000 0000 0000

# 8

## Genişletilmiş komutlar

### 8.1 Tarih, gün saati ve saat fonksiyonları

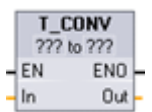
#### 8.1.1 Tarih ve gün saati komutları

Takvim ve saat hesaplamaları için date ve clock komutlarını kullanınız.

- T\_CONV bir değeri, (tarih ve saat veri tipleri) ve (byte, word ve dword boyut veri tipleri) tiplere dönüştürür veya bu tiplerden dönüştürür.
- T\_ADD Time ve DTL değerlerini toplar: (Time + Time = Time) veya (DTL + Time = DTL)
- T\_SUB Time ve DTL değerlerini çıkarır: (Time - Time = Time) veya (DTL - Time = DTL)
- T\_DIFF iki DTL değeri arasındaki farkı bir zaman değeri olarak sağlar: DTL - DTL = Time
- T\_COMBINE bir DTL değeri yaratmak için bir tarih değeri ile Date \_and\_Time değerini birleştirir.

DTL ve Time veri biçimi hakkında daha fazla bilgi için Date ve Time veri tipleri bölümüne bakınız (Sayfa 113).

Çizelge 8- 1 T\_CONV (zamanları dönüştür ve çıkar) komutu

LAD / FBD	SCL örneği	Açıklama
 <p>T_CONV ??? to ??? EN END In Out</p>	<pre>out := DINT_TO_TIME ( in:= _variant_in);  out := TIME_TO_DINT ( in:= _variant_in);</pre>	<p>_CONV bir değeri (Tarih ve zaman veri tipleri) ve (byte, word ve dword boyut veri tipleri)' ne veya bunlardan bir değere dönüştürür.</p>

<sup>1</sup> LAD ve FBD kutuları için: "???" tıklayınız ve aşağı açılır menüden kaynak/hedef veri tiplerini seçiniz.

<sup>2</sup> SCL için: Komut ağacından T\_CONV sürükleyiniz ve program editörüne bırakınız, sonra kaynak/hedef veri tiplerini seçiniz.

Çizelge 8- 2 T\_CONV dönüşümleri için geçerli veri tipleri

Veri tipi IN (veya OUT)	Veri tipleri OUT (veya IN)
TIME (milisaniye)	DInt, Int, SInt, UDInt, UInt, USInt, TOD SCL yalnızca: Byte, Word, Dword
DATE (1 Ocak 1990'dan beri gün sayısı)	DInt, Int, SInt, UDInt, UInt, USInt, DTL SCL yalnızca: Byte, Word, Dword
TOD (gece yarısından 24:00:00.000 beri milisaniye)	DInt, Int, SInt, UDInt, UInt, USInt, TIME, DTL SCL yalnızca: Byte, Word, Dword

## Genişletilmiş komutlar

### 8.1 Tarih, gün saati ve saat fonksiyonları

#### Not

**Daha büyük bir veri boyutunu daha küçük bir veri boyutuna dönüştürmek için T\_CONV kullanma**

Daha fazla baytlı daha büyük bir veri tipini daha az baytlı daha küçük bir veri tipine dönüştürdüğümüzde, veri değerleri kesilebilir. Bu hata oluşursa, ENO 0'a ayarlanır.


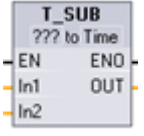
#### DTL veri tipine veya DTL veri tiplerinden dönüşüm

DTL (Tarih ve Zaman Uzunluğu) yıl, ay, gün ve zaman verisi içerir. DTL verisi DATE ve TOD veri tiplerine veya bu veri tiplerinden DTL verisine dönüştürülebilir.

Ancak, DATE verili DTL dönüşümü yalnızca yıl, ay ve gün değerlerini etkiler. TOD verili DTL dönüşümü yalnızca saat, dakika ve saniye değerlerini etkiler.

T\_CONV, DTL'ye dönüştürüldüğünde, DTL formatındaki etkilenmemiş veri öğeleri değiştirilmeden bırakılmıştır.

Çizelge 8-3 T\_ADD (zaman ekleme) ve T\_SUB (zaman çıkarma) komutları

LAD / FBD	SCL	Açıklama
	<pre>out := T_ADD(   in1:=_variant_in,   in2:=_time_in);</pre>	<p>T_ADD, IN2 giriş zaman değeri ile IN1 giriş değerini (DTL veya zaman veri tipleri) toplar. OUT Parametresi DTL veya Zaman değeri sonucu sağlar. İki veri tipi işlemi mümkündür:</p> <ul style="list-style-type: none"> <li>Time + Time = Time</li> <li>DTL + Time = DTL</li> </ul>
	<pre>out := T_SUB(   in1:=_variant_in,   in2:=_time_in);</pre>	<p>T_SUB, IN1 den (DTL veya zaman verisi) IN2 zaman değerini çıkarır. OUT Parametresi DTL veya Zaman veri tipi olarak fark değeri sağlar. İki veri tipi işlemi mümkündür:</p> <ul style="list-style-type: none"> <li>Time - Time = Time</li> <li>DTL - Time = DTL</li> </ul>

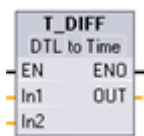
<sup>1</sup> LAD ve FBD için: "???" tıklayınız ve aşağı açılır menüden veri tiplerini seçiniz.

Çizelge 8-4 T\_ADD ve T\_SUB parametreleri için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN1 <sup>1</sup>	IN	DTL, Time
IN2	IN	Time
OUT	OUT	DTL, Time
		DTL veya Zaman toplamı veya farkı

<sup>1</sup> Komut ismi altında mevcut açılır listeden IN1 veri tipini seçiniz. IN1 veri tipi seçimi OUT parametresinin veri tipini belirler.

Çizelge 8- 5 T\_DIFF (Zaman farkı) komutu

LAD / FBD	SCL	Açıklama
	<pre>out := T_DIFF(     in1:=_DTL_in,     in2:=_DTL_in);</pre>	<p>T_DIFF, (IN1) DTL deęerinden (IN2) DTL deęerini çıkarır. OUT Parametresi zaman veri tipi olarak fark deęer sağlar.</p> <ul style="list-style-type: none"> <li>DTL - DTL = Time</li> </ul>

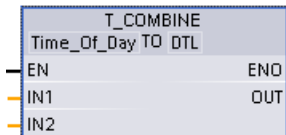
Çizelge 8- 6 T\_DIFF parametreleri için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN1	IN	DTL
IN2	IN	DTL
OUT	OUT	Time

**Durum kodları:** ENO = 1 hiç bir hatanın oluşmadığı anlamına gelir. ENO = 0 ve parametre OUT = 0 hataları:

- Geçersiz DTL deęeri
- Geçersiz zaman deęeri

Çizelge 8- 7 T\_COMBINE (Sreleri birleřtir) komutu

LAD / FBD	SCL	Açıklama
	<pre>out := CONCAT_DATE_TOD(     In1 := _date_in,     In2 := _tod_in);</pre>	<p>T_COMBINE, DTL deęeri oluşturmak için Date deęeri ile Time_of_Day deęerini birleřtirir.</p>

<sup>1</sup> Geniřletilmiř komutlardaki T\_COMBINE komutunun SCL'deki CONCAT\_DATE\_TOD fonksiyonuna eřit olduęu unutulmamalıdır.

Çizelge 8- 8 T\_COMBINE parametreleri için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN1	IN	Date
IN2	IN	Time_of_Day
OUT	OUT	DTL

## 8.1.2 Saat fonksiyonları

**! WARNING**

**Bir saldırgan Ağ Zaman Protokolü (NTP) senkronizasyonu yoluyla ağlara erişebilirse, saldırgan muhtemelen CPU sistem saatini kaydırarak yaptığı işlemi sınırlı olarak kontrolü altına alabilir.**


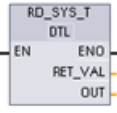
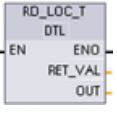
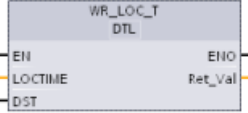
S7-1200 CPU'nun NTP istemci özelliği varsayılan olarak devre dışıdır ve etkinleştirildiğinde, sadece yapılandırılmış IP adreslerinin NTP sunucusu olarak davranmasına izin verir. CPU varsayılan olarak bu özelliği devre dışı bırakır ve uzaktan kontrollü CPU sistem saati düzeltmelerine izin vermek için bu özelliği yapılandırmanız gerekir.

S7-1200 CPU doğru CPU sistem saatine bağlı olan "günün saati" kesmeleri ve saat komutlarını destekler. NTP'yi yapılandırır ve bir sunucudan zaman senkronizasyonu kabul ederseniz, sunucunun güvenilir bir kaynak olduğundan emin olmalısınız. Aksi takdirde bilinmeyen bir kullanıcının CPU sistem saatini kaydırarak yaptığınız işlemi sınırlı olarak kontrolü altına alabilmesine izin veren bir güvenlik ihlaline sebep olabilir.

Güvenlik bilgisi ve öneriler için Siemens Servis ve destek sitesindeki ([http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational\\_guidelines\\_industrial\\_security\\_en.pdf](http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf)) "Sanayi Güvenliği ile ilgili operasyonel Kurallar" a bakılmalıdır.

CPU sistem saatini avarlamak ve okumak için saat komutlarını kullanınız. DTL veri türü (Sayfa 113) tarih ve zaman değerlerini sağlamak için kullanılır.

Çizelge 8- 9 Sistem zaman komutları

LAD / FBD	SCL	Açıklama
	<pre>ret_val := WR_SYS_T ( in:= DTL in );</pre>	WR_SYS_T ( günün saatini avarla) IN parametresindeki bir DTL değeri ile CPU'nun günün saatini avarlar. Bu zaman değeri, verel saat dilimini veya gün ışığı tasarruf süresi kaymalarını içermez.
	<pre>ret_val := RD_SYS_T ( out=&gt;_DTL_out);</pre>	RD_SYS_T (günün saati değerini oku) CPU' dan mevcut sistem zamanını okur. Bu zaman değeri, verel saat dilimini veya gün ışığı tasarruf süresi kaymalarını içermez.
	<pre>ret_val := RD_LOC_T ( out=&gt;_DTL_out);</pre>	RD_LOC_T (yerel saati oku) CPU'nun mevcut verel saatini, DTL veri tipi olarak sağlar. Bu zaman değeri, günün ışığı tasarruf süresi için uygun bir şekilde ayarlanan yerel saat dilimini yansıtır (Yapılandırılırsa).
	<pre>ret_val := WR_LOC_T ( LOCTIME:=DTL_in_, DST: _in_;</pre>	WR_LOC_T (verel saati vaz) CPU saatinin tarih ve zamanını avarlar. DTL veri tipi ile LOCTIME'da verel zaman olarak tarih ve zaman bilgilerini atavınız. Komut sistem zamanını hesaplamak için "ZamanDönüřtürmeKuralı (Savfa 289)" DB yapısını kullanır. Yerel zaman ve sistem zamanı için zaman bilgisinin öe boyu ürüne özündür ve en az bir milisaniyedir. CPU tarafından desteklenenden daha az olan ve LOCTIME parametresindeki girdi değerleri sistem zaman hesaplaması sırasında yuvarlanır. <b>Not:</b> "gün zamanı" özelliklerini (zaman dilimi, DST aktivasyonu, DST başlatma ve DST durma) ayarlamak için CPU cihaz yapılandırması kullanmanız gerekir. Aksi takdirde, WR_LOC_T DST saat deęişiklięini yorumlayamazsınız.

Çizelge 8- 10 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
IN	IN	DTL	CPU sistem saatinde ayarlanacak gün zamanı
OUT	OUT	DTL	RD_SYS_T: Mevcut CPU sistem zamanı RD_LOC_T: Mevcut yerel zaman, yaz saati için herhangi bir ayarlama dahil, yapılandırılırsa.
LOCTIME	IN	DTL	WR_LOC_T: Yerel zaman
DST	IN	BOOL	WR_LOC_T: Saatler yaz saati ayarlaması deęiřtirildiğinde yaz saati sadece "çift saat" boyunca deęerlendirilir. • TRUE = gün ışığından yararlanma süresi (birinci saat) • FALSE = standart süre (ikinci saat)
RET_VAL	OUT	Int	Yürütme durum kodu

- Yerel saat, Cihaz Yapılandırmasının Genel sekmesinde "Günün saati" parametrelerinde ayarlanan zaman dilimi ve yaz saati farkı kullanılarak hesaplanır.
- Zaman dilimi konfigürasyonu, UTC veya GMT saati kaymasıdır.

## Genişletilmiş komutlar

### 8.1 Tarih, gün saati ve saat fonksiyonları

- Yaz saati konfigürasyonunda, yaz saati uygulaması başladığında ay, hafta, gün ve saat belirtilir.
- Standart zaman konfigürasyonunda standart zaman başladığında ay, hafta, gün ve saat belirtilir.
- Zaman dilimi farkı, sistem zaman değeri için her zaman uygulanır. Yaz saati farkı, yaz saati uygulaması etkin olduğunda uygulanır.

#### Not

#### Yaz saati uygulaması ve standart zaman konfigürasyonunu başlatmak

" CPU Cihaz Yapılandırmasının "yaz saatini başlat" ile ilgili "Günün saati" Özellikleri sizin yerel saatiniz olmalıdır.

**Durum kodları:** ENO = 1 herhangi bir hata oluşmadığı anlamına gelir. ENO = 0 bir yürütme hatasının oluştuğu anlamına gelir ve RET\_VAL çıkışında bir durum kodu sağlanır.

RET_VAL (W#16#...)	Açıklama
0000	Mevcut yerel saat, standart zaman içindedir.
0001	Yaz saati uygulaması yapılandırılmıştır ve mevcut yerel saat, yaz saati uygulaması içindedir.
8080	Yerel saat kullanılabilir değil veya LOCTIME değeri geçersiz.
8081	Kural dışı yıl değeri veya LOCTIME parametresiyle verilen zaman değeri geçersiz.
8082	Kural dışı ay değeri (DTL formatında 2 bayt )
8083	Kural dışı gün değeri (DTL formatında 3 bayt )
8084	Kural dışı saat değeri (DTL formatında 5 bayt )
8085	Kural dışı dakika değeri (DTL formatında 6 bayt )
8086	Kural dışı saniye değeri (DTL formatında 7 bayt )
8087	Kural dışı nanosaniye değeri (DTL formatında 8 ila 11 bayt )
8089	Zaman değeri mevcut değil (saat değişimle zaten yaz saati uygulamasına geçirilmiştir)
80B0	Gerçek zaman saati bozuldu
80B1	"TimeTransformationRule" yapısı tanımlanmamıştır.



### 8.1.3 TimeTransformationRule veri yapısı

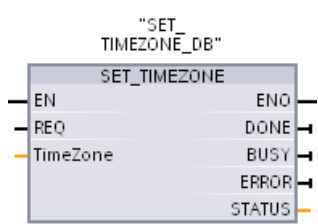
#### Açıklama

Standart zaman ve yaz saati uygulamasıyla ilgili deęişim kuralları TimeTransformationRule yapısında tanımlanır. Yapı ařağıdaki gibidir:

İsim	Veri tipi	Açıklama
TimeTransformationRule	STRUCT	
Bias	INT	Yerel saat ile UTC arasındaki zaman farkı [dakika]
DaylightBias	INT	Yaz saati ile standart saat arasındaki zaman farkı [dakika]
DaylightStartMonth	USINT	Yaz saatine geçiř ayı
DaylightStartWeek	USINT	Yaz saatine geçiř haftası 1 = Ay içindeki haftanın gününde ilk meydana gelme , ..., 5 = Ay içindeki haftanın gününde son meydana gelme
DaylightStartWeekday	USINT	Yaz saati deęişiminin yapıldığı gün: 1 = Pazar
DaylightStartHour	USINT	Yaz saati deęişiminin yapıldığı saat
DaylightStartMinute	USINT	Yaz saati deęişiminin yapıldığı dakika
StandardStartMonth	USINT	Yaz saati deęişiminin yapıldığı ay
StandardStartWeek	USINT	Standart saate geçiř haftası 1 = Ay içindeki haftanın gününde ilk meydana gelme, ..., 5 = Ay içindeki haftanın gününde son meydana gelme
StandardStartWeekday	USINT	Standart saat deęişiminin yapıldığı hafta: 1 = Pazar
StandardStartHour	USINT	Standart saat deęişiminin yapıldığı saat
StandardStartMinute	USINT	Yaz saati deęişiminin yapıldığı dakika
TimeZoneName	STRING[80]	Zaman diliminin ismi: "(GMT+01:00) Amsterdam, Berlin, Bern, Roma, Stockholm, Viyena"

### 8.1.4 SET\_TIMEZONE (Zaman dilimini ayarla) komutu

Çizelge 8- 11 SET\_TIMEZONE komutu

LAD / FBD	SCL	Açıklama
	<pre>"SET_TIMEZONE_DB" (   REQ:=_bool_in,   Timezone:=_struct_in,   DONE=&gt;_bool_out_,   BUSY=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_);</pre>	<p>CPU sistem saatini yerel saate dönüştürmek için kullanılan yerel saat dilimi ve yaz saati parametrelerini ayarlar.</p>

1 SCL örneğinde, "SET\_TIMEZONE\_DB" kopya DB ismidir.

Çizelge 8- 12 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	REQ=1: fonksiyonu yürüt
Timezone	IN	TimeTransformationRule	Sistem saatinden yerel saate dönüşüm kuralları
DONE	OUT	Bool	Fonksiyon tamamlandı
BUSY	OUT	Bool	Fonksiyon meşgul
ERROR	OUT	Bool	Hata algılandı
STATUS	OUT	Word	Fonksiyon sonucu / hata mesajı

CPU için zaman dilimi parametrelerini elle yapılandırmak üzere Cihaz Yapılandırmasının "Genel" sekmesinin özelliklerindeki "Günün saati" kullanılır.

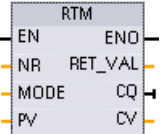
Programlanabilir bir biçimde yerel saat yapılandırmasını avarlamak için SFT TIMEZONE komutu kullanılır. "TimeTransformationRule (Savfa 289)" varısı standart saat ile yaz saati arasında otomatik geçiş için yerel saat dilimini ve zamanlamayı tahsis eder.

Durum kodları: FNO = 1 hatanın oluşmadığı anlamına gelir. FNO = 0 bir yürütme hatasının olduğu anlamına gelir ve STATUS çıkışında bir durum kodu sağlanır.

STATUS (W#16#....)	Açıklama
0	Hata yok
7000	Aktif bir iş prosesi yok
7001	İş prosesi başlangıcı. Parametre BUSY = 1, DONE = 0
7002	Ara çağrı (REQ ilgisiz): Komut çoktan aktif; BUSY "1" değerine sahip.
808x	x-nci bileşende hata: Örneğin 8084, DaylightStartWeekif'in 1 ile 5 arasında bir değer olmadığını gösterir.

## 8.1.5 RTM (Çalışma zamanı ölçer) komutu

Çizelge 8- 13 RTM komutu

LAD / FBD	SCL	Açıklama
	<pre>RTM(NR:=_uint_in_,      MODE:=_byte_in_,      PV:=_dint_in_,      CQ=&gt;_bool_out_,      CV=&gt;_dint_out_);</pre>	<p>RTM (çalışma zamanı ölçer) komutu CPU'daki çalışma zamanı ölçülerini ayarlayabilir, başlatabilir, durdurabilir ve okuyabilir.</p>

Çizelge 8- 14 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
NR	IN	UInt
MODE	IN	Byte
PV	IN	DInt
RET_VAL	OUT	Int
CQ	OUT	Bool
CV	OUT	DInt

Kritik kontrol alt sistemlerin çalışma saatlerini izlemek için CPU, 10 çalışma saati ölçüsüne kadar çalışır. Münferit saat ölçer, her zamanlayıcı için bir RTM yürütmesiyle birlikte başlatılmalıdır. CPU run-stop geçişi yaptığında tüm çalışma zamanı saat ölçer durdurulur. RTM yürütme modu 2 ile bireysel Zamanlayıcıları da durdurabilirsiniz.

Bir CPU bir stop-run geçişi yaptığında, saat zamanlayıcılar, başlatılan her zamanlayıcı için bir RTM yürütmesi ile birlikte yeniden başlatılmalıdır. Bir çalışma zamanı ölçer değeri 2147483647 saatten büyük olduktan sonra sayma durur ve "Overflow" hatası gönderilir. Zamanlayıcı değiřtirmek veya sıfırlamak için her zamanlayıcı için bir kez RTM komutu yürütülmelidir.

CPU güç arızası veya güç çevrimi, mevcut çalışma zamanı ölçer değerlerini tutucu belleğe kaydeden bir güç kapanma prosesine neden olur. CPU yeniden enerjilenmesi durumunda, saklanan çalışma zamanı ölçer değerleri zamanlayıcıya yeniden yüklenir ve önceki çalışma süresi saat toplamları kaybolmaz. İlave çalışma zamanını depolaması için çalışma zamanı ölçerin yeniden başlatılması gerekir.

Programınız, bir bellek kartında alıřma zamanı ler deęerlerini kaydetmek iin RTM yrtme modu 7'yi kullanabilirsiniz. RTM modu 7'nin yrtldę anda tm zamanlayıcıların durumları bellek kartı iinde saklanır. Saat zamanlayıcıları program alıřtırma blm sırasında bařlatılıp durdurulduka saklanan bu deęerler, zaman iinde hatalı hale gelebilir. nemli alıřma zamanı olaylarını yakalamak iin bellek kartı deęerleri dzenli olarak gncellenmelidir. RTM deęerlerinin bellek kartında saklanmasından dolayı elde edilecek avantaj, bellek kartının, programınızın ve kaydedilmiş RTM deęerlerinin kullanılabileceęi yedek CPU'ya yerleřtirilebilir olmasıdır. RTM deęerlerini bellek kartına kaydetmediyseniz, zamanlayıcı deęerleri (yedek CPU iinde) kaybolabilir.

**Not**

**Bellek kartı yazma iřlemleri iin ařırı program aęrılarından kaınınız.**

Bellek kartının mrn uzatmak iin flař bellek kartı yazma iřlemlerini en aza indiriniz.

izelge 8- 15 Durum kodları

RET_VAL (W#16#....)	Aıklama
0	Hata yok
8080	Yanlıř alıřma zamanı ler numarası
8081	PV parametresine negatif bir deęer geirildi
8082	alıřma saati sayacı tařması
8091	Giriř parametresi MODE geersiz bir deęer ieriyor
80B1	Deęer MC'ye kaydedilemez (MODE=7)

## 8.2 String ve karakter

### 8.2.1 String (karakter dizisi) veriye genel bakıř

#### String (karakter dizisi) veri tipi

String verisi, 254 karakter baytına kadar ASCII karakter kodlarından nce 2 baytlık bařlık olarak depolanır. Bir string bařlıęı iki uzunluk ierir. İlk bayt, bir string deęer atandıęında křeli parantez iinde verilen azami uzunluk veya varsayılan olarak 254'tr. İkinci bařlık baytı string'deki geerli karakter sayısı olan mevcut uzunluktur. Mevcut uzunluk azami uzunluęa eřit veya daha kk olmalıdır. String biimde yer alan saklanan bayt sayısı azami uzunluktan daha byk olan 2 bayttir.

#### String verinizin bařlangı ayarı

String giriř ve ıkıř verileri herhangi bir string komutun yrtlmesinden nce, bellekteki geerli string'ler olarak bařlatılması gerekmektedir.

## Geçerli String verisi

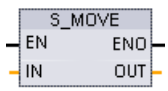
Geçerli string sıfırdan büyük ancak 255'ten daha küçük olması gereken azami uzunluğa sahiptir. Mevcut uzunluk azami uzunluktan daha az veya eşit olmalıdır.

String'ler I veya Q bellek alanlarına atanamaz.

Daha fazla bilgi için bakınız: String veri tipinin biçimlendirilmesi (Sayfa 114).

## 8.2.2 S\_MOVE (Karakter string'i kaydır) komutu

Çizelge 8- 16 String kaydır komutu

LAD / FBD	SCL	Açıklama
 <p>The image shows a ladder logic symbol for the S_MOVE instruction. It is a rectangular box with 'S_MOVE' written inside. On the left side, there are two terminals: 'EN' (top) and 'IN' (bottom). On the right side, there are two terminals: 'END' (top) and 'OUT' (bottom).</p>	<pre>out := in;</pre>	<p>Kaynak IN string'ini OUT konumuna kopyalayınız. S_MOVE'un yürütülmesi kaynak string'in içeriğini etkilemez.</p>

Çizelge 8- 17 Parametreler için veri tipleri

Parametre	Veri tipi	Açıklama
IN	String	Kaynak string
OUT	String	Hedef adres

IN Girişinde string'in gerçek uzunluğu OUT çıkışında saklanan bir string'in azami uzunluğunu aşarsa, o zaman OUT string'in içine sığabilecek IN string bölümü kopyalanır.

## 8.2.3 String dönüşüm komutları

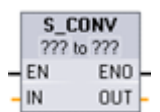
### 8.2.3.1 S\_CONV, STRG\_VAL ve VAL\_STRG (Karakter string'e ve sayıya veya bunlardan dönüştürme) komutları

Bu komutlarla sayı karakter string'leri sayı değerlerine veya sayı değerlerini sayı karakter string'lere dönüştürebilirsiniz:

- S\_CONV (sayı string'i sayı değerine) veya (sayı değerini sayı string'e) dönüştürür
- STRG\_VAL bir sayı string'i bir sayı değerine format seçenekleriyle dönüştürür
- VAL\_STRG bir sayı değerini bir sayı string'e bir format seçenekleriyle dönüştürür

## S\_CONV (karakter string'i dönüştür)

Çizelge 8- 18 String dönüşüm komutu

LAD / FBD	SCL	Açıklama
	<pre>out := &lt;Type&gt;_TO_&lt;Type&gt;(in) ;</pre>	<p>Karakter string'i karşılık gelen değere veya bir değeri karşılık gelen karakter string'e dönüştürür. S_CONV komutunun çıkış formatlama seçenekleri yoktur. Bu, S_CONV komutunu basit hale getirir, ancak STRG_VAL ve VAL_STRG komutlarından daha az esneklik.</p>

- LAD/FBD için: "???" tıklayınız ve aşağı açılır listeden veri tipini seçiniz.
- SCL için: Genişletilmiş komutlardan S\_CONV'u seçiniz ve dönüşümle ilgili veri tipleri için isteklere cevap veriniz. Daha sonra STEP 7 uygun dönüşüm komutunu sağlar.

Çizelge 8- 19 Veri tipleri (string'ten değere)

Parametre ve tipi		Veri tipi	Açıklama
IN	IN	String	Giriş karakter string
OUT	OUT	String, Char, SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Çıkış sayı değeri

IN string parametresi dönüşümü ilk karakterde başlar ve string'in sonuna kadar devam eder veya 0-"9", "+", "-", veya "." olmayan ilk karakterle karşılaşılana kadar devam eder. Sonuç değer OUT parametresinde belirtilen konumda sağlanır. Çıkış sayı değeri OUT veri tipi aralığına girmiyorsa, OUT parametresi 0 olarak ve ENO ise FALSE olarak ayarlanır. Aksi takdirde, OUT parametresi geçerli bir sonuç içerir ve ENO TRUE olarak ayarlanır.

Giriş String format kuralları:

- IN string'de bir ondalık işareti kullanılırsa, "." karakteri kullanmalısınız.
- Ondalık işaretinin solunda biner işareti olarak virgöl karakteri "," kullanılmasına izin verilir ve ihmal edilir.
- Öndeki boşluklar ihmal edilir.

## S\_CONV (değerden string'e dönüşüm)

Çizelge 8- 20 Veri tipleri (değerden string'e)

Parametre ve tipi		Veri tipi	Açıklama
IN	IN	String, Char, SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal	Giriş sayı değeri
OUT	OUT	String	Çıkış karakter string

Bir integer, unsigned integer veya IN kayan nokta değeri OUT'ta karşılık gelen karakter string'e dönüştürülür. OUT parametresi dönüşüm yapılmadan önce geçerli bir string'e atıfta bulunmalıdır. Geçerli bir string ilk bayt'ta azami string uzunluğundan, ikinci bayt'ta mevcut string uzunluğundan, sonraki bayt'larda mevcut string karakterlerden oluşur. Dönüştürülmüş string ilk karakterden başlayarak OUT string'deki karakterlerin yerini alır ve OUT string'in mevcut uzunluk bayt'ını ayarlar. OUT string'in azami uzunluk bayt'ı değişmez.

Kaç tane karakterin değiştirileceği IN parametresinin veri tipine ve sayı değerine bağlıdır. Değiştirilecek karakter sayısı OUT parametresi string uzunluğunu aşmamalıdır. OUT string'inin azami string uzunluğu (ilk bayt) dönüştürülen karakterlerin azami beklenen sayısına eşit veya daha büyük olmalıdır. Aşağıdaki Çizelgede, S\_CONV değeri string'e dönüştürme örnekleri gösterilmiştir.

Çıkış String format kuralları:

- OUT parametresine yazılan değerlerin önünde "+" işareti kullanılmaz.
- Sabit nokta gösterimi kullanılır (üstel gösterim yok).
- IN parametresi gerçek veri tipi olduğunda periyot karakteri "." ondalık işaretini temsil etmek için kullanılır.
- Değerler, çıkış string'inde sağa yaslanmış ve boş karakter pozisyonlarını dolduracak boşluk karakterleri bunların öncesinde bulunur.


Çizelge 8- 21 Her veri tipi için azami string uzunlukları

IN veri tipi	S_CONV ile tahsis edilen karakter pozisyonları	Dönüştürülen string örneği <sup>1</sup>	Azami ve mevcut uzunluk bayt'ları dahil toplam string'in uzunluğu
USInt	4	"x255"	6
SInt	4	"-128"	6
UInt	6	"x65535"	8
Int	6	"-32768"	8
UDInt	11	"x4294967295"	13
DInt	11	"-2147483648"	13
Real	14	"x-3.402823E+38" "x-1.175495E-38" "x+1.175495E-38" "x+3.402823E+38"	16
LReal	21	"-1.7976931348623E+308" "-2.2250738585072E-308" "+2.2250738585072E-308" "+1.7976931348623E+308"	23

<sup>1</sup> "x" karakterleri dönüştürülmüş değeri için ayrılan sağa yaslanmış alanda boş pozisyonları dolduran boşluk karakterlerini temsil eder.

## STRG\_VAL (karakter string'i sayısal değere dönüştürür)

Çizelge 8- 22 String-değer komutu

LAD / FBD	SCL	Açıklama
	<pre>"STRG_VAL" (   in:=_string_in,   format:=_word_in,   p:=uint_in,   out=&gt;_variant_out);</pre>	<p>Sayı karakter string'i karşılık gelen tamsayıya veya kayan nokta gösterimine dönüştürür.</p>

<sup>1</sup> LAD / FBD için: "???" tıklayınız ve açılır listeden veri tipini seçiniz.

Çizelge 8- 23 STRG\_VAL komutu için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN	IN	String
FORMAT	IN	Word
P	IN	UInt, Byte, USInt
OUT	OUT	SInt, Int, DInt, USInt, UInt, UDIInt, Real, LReal

Dönüşüm P karakter kaydırmadaki IN string'inde başlar ve string'in sonuna kadar devam eder veya "+", "-", ".", ",", "e", "E" veya "0" ile "9" olmayan ilk karakterle karşılaşana kadar devam eder. Sonuç OUT parametresinde belirtilen konuma yerleştirilir.

String verinin, yürütülmeden önce, hafızada geçerli bir string olarak başlatılması gerekmektedir.

STRG\_VAL komutu için FORMAT parametresi aşağıda tanımlanmıştır. Kullanılmayan bit pozisyonları sıfır olarak ayarlanmalıdır.

Çizelge 8- 24 STRG\_VAL komutunun formatlanması

Bit 16								Bit 8	Bit 7						Bit 0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	f r

f = Gösterim formatı

r = Ondalık işaret formatı

1= Üstel gösterim

0 = Sabit nokta gösterimi

1 = "," (virgül karakteri)

0 = "." (periyot karakteri)



Çizelge 8- 25 FORMAT parametresi değerleri

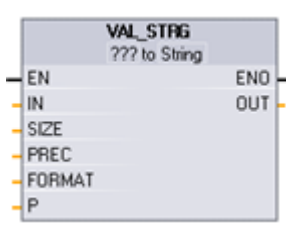
FORMAT (W#16#)	Gösterim formatı	Ondalık işaret gösterimi
0000 (varsayılan)	Sabit nokta	"."
0001		" "
0002	Üstel	"."
0003		" "
0004 - FFFF	Geçersiz değerler	

STRG\_VAL dönüşüm kuralları:

- Periyot karakteri "." ondalık işareti için kullanılırsa, ondalık işaretin solundaki virgüller "," binler ayraç karakteri olarak yorumlanır. Virgül karakterlerine izin verilir ve ihmal edilir.
- Virgül karakteri "," ondalık işareti için kullanılırsa, ondalık işaretin solundaki periyotlar "." binler ayraç karakteri olarak yorumlanır. Bu periyot karakterlerine izin verilir ve ihmal edilir.
- Öndeki boşluklar ihmal edilir.

## VAL\_STRG (sayısal değeri string'e dönüştürme)

Çizelge 8- 26 Değer-string işlemi

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a ladder logic instruction box labeled 'VAL_STRG' with the text '??? to String'. It has an 'EN' (Enable) input on the left, an 'ENO' (Enable Out) output on the right, an 'IN' (Input) on the left, and an 'OUT' (Output) on the right. Below the box, the parameters 'SIZE', 'PREC', 'FORMAT', and 'P' are listed.</p>	<pre>"VAL_STRG" (     in:=_variant_in,     size:=_usint_in,     prec:=_usint_in,     format:=_word_in,     p:=_uint_in,     out=&gt;_string_out);</pre>	<p>Bir tam sayı, işaretsiz tam sayı veya kayan nokta değerini karşılık gelen karakter string gösterimine dönüştürür.</p>

1 LAD / FBD için: "???" tıklayınız ve açılır listeden veri tipini seçiniz.

Çizelge 8- 27 VAL\_STRG komutu için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN	IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal
SIZE	IN	USInt
PREC	IN	USInt
FORMAT	IN	Word
P	IN	UInt, Byte, USInt
OUT	OUT	String

## Geniřletilmiř komutlar

### 8.2 String ve karakter

IN parametresiyle temsil edilen deęer, OUT parametresiyle bařvurulan bir string'e dnřtrlr. Dnřtrme iřlemi yrtlmeden nce OUT parametresinin geerli bir string olması gerekir.

Dnřtrlen string, P karakter ofset sayısından bařlayarak, SIZE parametresiyle belirtilen karakter sayısına kadar OUT iindeki karakterlerin yerini alacaktır. P karakter pozisyonundan itibaren sayarak SIZE iindeki karakterlerin sayısı OUT string uzunluęuyla eřleřmelidir. Bu komut, bir metin string iine sayı karakterlerinin yerleřtirilmesinde yararlıdır. rneęin, "Pompa basıncı = 120 psi" string'i iine "120" sayılarını koyabilirsiniz.

PREC parametresi string'in kesirsel blm iin duyarlıęı veya basamak sayısını belirtir. IN parametre deęeri bir integer ise, PREC ondalık iřaretin konumunu belirtir. rneęin veri deęeri 123 ve PREC= 1 ise, o zaman sonu "12.3" olur. Gerek veri tipi iin desteklenen azami duyarlılık 7 basamaktan oluřur.

P parametresi OUT string'inin mevcut boyutundan daha byk ise, o zaman P konumuna kadar bořluk eklenir ve sonu string'in sonuna ilave edilir. Azami OUT string uzunluęuna ulařılırsa dnřtrme iřlemi biter.

VAL\_STRG komutu iin FORMAT parametresi ařaęıda tanımlanmıřtır. Kullanılmayan bit pozisyonları sıfır olarak ayarlanmalıdır.

izelge 8- 28 VAL\_STRG komutu formatı

Bit 16								Bit 8	Bit 7							Bit 0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	s	f	r

s = Sayı iřaret karakteri

f = Gsterim formatı

r = Ondalık iřaret formatı

1= "+" ve "-" iřaret karakterini kullanınız

0 = yalnızca "-" iřaret karakterini kullanınız

1= stel gsterim

0 = Sabit nokta gsterimi

1 = "," (virgl karakteri)

0 = "." (periyot karakteri)

izelge 8- 29 FORMAT parametre deęerleri

FORMAT (WORD)	Sayı iřaret karakteri	Gsterim formatı	Ondalık iřaret gsterimi
W#16#0000	"." sadece	Sabit nokta	"."
W#16#0001			"."
W#16#0002		stel	"."
W#16#0003			"."
W#16#0004	"+" ve "-"	Sabit nokta	"."
W#16#0005			"."
W#16#0006		stel	"."
W#16#0007			"."
W#16#0008 - W#16#FFFF	Geersiz deęerler		

OUT Parametresi string format kuralları:

- Dönüřtürülen string, belirtilen boyuttan küçük olduėunda öndeki boşluk karakterleri string'in en soluna ilave edilir.
- FORMAT parametresi işaret biti FALSE olduėunda, unsigned ve signed integer veri tipi deėerleri, önünde "+" işareti olmaksızın çıkış ara belleėine yazılır. "-" işareti gerekli olduėunda kullanılır.  
< öndeki boşluklar >< öndeki sıfırlar olmaksızın basamaklar >'<PREC basamakları>
- İşaret biti TRUE olduėunda, unsigned ve signed integer veri tipi deėerleri, önündeki işaret karakterleri ile daima çıkış ara belleėine yazılır.  
<öndeki boşluklar><işaret><öndeki sıfırlar olmaksızın basamaklar >'<PREC basamakları>
- FORMAT üstel gösterim olarak ayarlandıėında, Gerçek veri tipi deėerleri ařaėıdaki gibi çıkış ara belleėine yazılır:  
< öndeki boşluklar >< işaret >< basamak > ' ' <PREC basamaklar >'E' < işaret >< öndeki sıfırlar olmaksızın basamaklar >
- FORMAT sabit nokta gösterimine ayarlandıėında, integer, unsigned integer ve real veri tipi deėerleri ařaėıdaki gibi çıkış ara belleėine yazılır:  
< öndeki boşluklar >< işaret >< öndeki sıfırlar olmaksızın basamaklar >'<PREC basamaklar>
- Ondalık işaretin solundaki sıfırlar (Ondalık işarete komřu basamak hariç) kaldırılır.
- Ondalık işaretin saėındaki deėerler PREC parametresiyle belirtilen ondalık işaretin saėına uyacak şekilde basamak sayısına uygun olarak yuvarlatılır.
- Çıkış string'inin boyutu, ondalık işaretin saėındaki basamak sayısından asgari üç bayt fazla olmalıdır.
- Deėerler çıkış string'inde saėa yaslamalı olacaktır.

## ENO ile raporlanan kořullar

Dönüřtürme işlemi sırasında bir hata ile karřılařıldığında, ařaėıdaki sonuçlar döndürülür:

- ENO 0'a ayarlanır.
- OUT 0'a ayarlanır veya string – deėer dönüřümü için örneklerde gösterildiėi gibi.
- OUT deėiřmez veya OUT bir string olduėunda örneklerde gösterildiėi gibi.

Çizelge 8- 30 ENO durumu

ENO	Açıklama
1	Hata yok
0	Kural dışı veya geçersiz parametre; örneėin mevcut olmayan DB'ye erişim
0	Kural dışı string, burada azami string uzunluėu 0 veya 255
0	Kural dışı string, burada mevcut uzunluk azami uzunluktan büyüktür
0	Dönüřtürülen sayı deėeri belirtilen OUT veri tipi için çok büyük.
0	P parametresi karakter pozisyonundan başlamak üzere, OUT parametresi azami string boyutu SIZE parametresiyle belirtilen karakter sayısını kabul edecek kadar geniř olmalıdır.

## Geniřletilmiř komutlar

### 8.2 String ve karakter

ENO	Açıklama
0	Kural dıřı P deęeri, burada P=0 veya P mevcut string uzunluęundan büyük
0	SIZE Parametresi, PREC parametresinden büyük olmalıdır.

Çizelge 8- 31 S\_CONV string – deęer dönüşümü ile ilgili örnekler

IN string	OUT veri tipi	OUT deęeri	ENO
"123"	Int veya DInt	123	TRUE
"-00456"	Int veya DInt	-456	TRUE
"123.45"	Int veya DInt	123	TRUE
"+2345"	Int veya DInt	2345	TRUE
"00123AB"	Int veya DInt	123	TRUE
"123"	Real	123.0	TRUE
"123.45"	Real	123.45	TRUE
"1.23e-4"	Real	1.23	TRUE
"1.23E-4"	Real	1.23	TRUE
"12,345.67"	Real	12345.67	TRUE
"3.4e39"	Real	3.4	TRUE
"-3.4e39"	Real	-3.4	TRUE
"1.17549e-38"	Real	1.17549	TRUE
"12345"	SInt	0	FALSE
"A123"	N/A	0	FALSE
""	N/A	0	FALSE
"++123"	N/A	0	FALSE
"-123"	N/A	0	FALSE

Çizelge 8- 32 S\_CONV deęer-string dönüşümü ile ilgili örnekler

Veri tipi	IN deęeri	OUT string <sup>1</sup>	ENO
UInt	123	"xxx123"	TRUE
UInt	0	"xxxxx0"	TRUE
UDInt	12345678	"xxx12345678"	TRUE
Real	+9123.456	"xx+9.123456E+3"	TRUE
LReal	+9123.4567890123	"xx+9.1234567890123 E+3"	TRUE
Real	-INF	"xxxxxxxxxxxINF"	FALSE
Real	+INF	"xxxxxxxxxxxINF"	FALSE
Real	NaN	"xxxxxxxxxxxNaN"	FALSE

<sup>1</sup> "x" karakterleri, dönüřtürülen deęer için ayrılan saęa yaslanmıř alanda boş pozisyonları dolduran boşluk karakterlerini temsil eder.

Çizelge 8- 33 STRG\_VAL dönüşümüyle ilgili örnekler

IN string	FORMAT (W#16#...)	OUT veri tipi	OUT değeri	ENO
"123"	0000	Int veya DInt	123	TRUE
"-00456"	0000	Int veya DInt	-456	TRUE
"123.45"	0000	Int veya DInt	123	TRUE
"+2345"	0000	Int veya DInt	2345	TRUE
"00123AB"	0000	Int veya DInt	123	TRUE
"123"	0000	Real	123.0	TRUE
"-00456"	0001	Real	-456.0	TRUE
"+00456"	0001	Real	456.0	TRUE
"123.45"	0000	Real	123.45	TRUE
"123.45"	0001	Real	12345.0	TRUE
"123.45"	0000	Real	12345.0	TRUE
"123.45"	0001	Real	123.45	TRUE
".00123AB"	0001	Real	123.0	TRUE
"1.23e-4"	0000	Real	1.23	TRUE
"1.23E-4"	0000	Real	1.23	TRUE
"1.23E-4"	0002	Real	1.23E-4	TRUE
"12,345.67"	0000	Real	12345.67	TRUE
"12,345.67"	0001	Real	12.345	TRUE
"3.4e39"	0002	Real	+INF	TRUE
"-3.4e39"	0002	Real	-INF	TRUE
"1.1754943e-38" (ve daha küçük)	0002	Real	0.0	TRUE
"12345"	N/A	SInt	0	FALSE
"A123"	N/A	N/A	0	FALSE
""	N/A	N/A	0	FALSE
"++123"	N/A	N/A	0	FALSE
"+-123"	N/A	N/A	0	FALSE

VAL\_STRG dönüşümleriyle ilgili izleyen örnekler aşağıdaki gibi başlatılmış bir OUT string'e dayanır:

"Mevcut Sıcaklık = xxxxxxxxxxx C"

Burada "x" karakteri, dönüştürülmüş değer için ayrılan boşluk karakterlerini temsil eder.

## Geniştirilmiş komutlar

### 8.2 String ve karakter

Çizelge 8- 34 VAL\_STRG dönüşümüyle ilgili örnekler

Veri tipi	IN değeri	P	BOYUT	FORMAT (W#16#....)	PREC	OUT string	ENO
UInt	123	16	10	0000	0	Mevcut Sıcaklık = xxxxxxxx123 C	TRUE
UInt	0	16	10	0000	2	Mevcut Sıcaklık = xxxxxx0.00 C	TRUE
UDInt	12345678	16	10	0000	3	Mevcut Sıcaklık = x12345.678 C	TRUE
UDInt	12345678	16	10	0001	3	Mevcut Sıcaklık = x12345,678 C	TRUE
Int	123	16	10	0004	0	Mevcut Sıcaklık = xxxxxx+123 C	TRUE
Int	-123	16	10	0004	0	Mevcut Sıcaklık = xxxxxx-123 C	TRUE
Real	-0.00123	16	10	0004	4	Mevcut Sıcaklık = xxx-0.0012 C	TRUE
Real	-0.00123	16	10	0006	4	Mevcut Sıcaklık = -1.2300E-3 C	TRUE
Real	-INF	16	10	N/A	4	Mevcut Sıcaklık = xxxxxx-INF C	FALSE
Real	+INF	16	10	N/A	4	Mevcut Sıcaklık = xxxxxx+INF C	FALSE
Real	NaN	16	10	N/A	4	Mevcut Sıcaklık = xxxxxxxNaN C	FALSE
UDInt	12345678	16	6	N/A	3	Mevcut Sıcaklık = xxxxxxxxxx C	FALSE

#### 8.2.3.2 Strg\_TO\_Chars ve Chars\_TO\_Strg (karakter string'e ve CHAR dizisine veya bunlardan dönüşüm) komutları

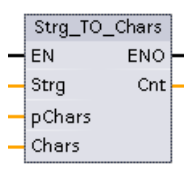
Strg\_TO\_Chars, bir ASCII karakter string'i karakter byte dizisine kopyalar.

Chars\_TO\_Strg, ASCII karakter byte dizisini karakter string'e kopyalar.

#### Not

Sadece sıfır tabanlı dizi tipleri (Karakter Dizisi [0 .. n]) veya (Byte Dizisi [0 .. n]), Chars\_TO\_Strg komutu için giriş parametresi karakteri olarak ya da Strg\_TO\_Chars komutu için IN\_OUT parametre karakteri olarak izin verilir.

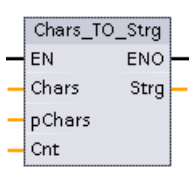
Çizelge 8- 35 Strg\_TO\_Chars komutu

LAD / FBD	SCL	Açıklama
	<pre>Strg_TO_Chars (     Strg:=_string_in_,     pChars:=_dint_in_,     Cnt=&gt;_uint_out_,      Chars:=_variant_inout_);</pre>	<p>Komple giriş string'i Strg, IN_OUT parametresi Chars' taki bir karakter dizisine kopyalanır.</p> <p>İşlem, pChars parametresi ile belirtilen dizi eleman sayısından başlayarak bayt'ların üzerine yazar.</p> <p>Tüm desteklenen azami uzunluklardaki (1 .. 254) String'ler kullanılabilir.</p> <p>Bir son ayraç yazılmaz; bu sizin sorumluluğunuzdur. Yazılan son dizi karakterinden hemen sonra bir son ayraç koymak istiyorsanız, bir sonraki dizi eleman sayısını kullanınız [pChars+Cnt].</p>

Çizelge 8- 36 Parametreler için veri tipleri (Strg\_TO\_Chars)

Parametre ve tipi		Veri tipi	Açıklama
Strg	IN	String	Kaynak string
pChars	IN	DInt	Hedef diziye yazılan ilk string karakter için dizi elemanı sayısı
Chars	IN_OUT	Variant	Chars parametresi, giriş string'inden kopyalanan sıfır tabanlı karakter dizisini [0 .. n] gösteren bir işaretçidir. Dizi, blok ara yüzündeki yerel değişkenler olarak veya bir DB içinde beyan edilebilir. Örnek: "DB1".MyArray, DB1'deki Char eleman değerlerinden oluşan MyArray [0..10]'i gösterir.
Cnt	OUT	UInt	Kopyalanan karakter sayısı

Çizelge 8- 37 Chars\_TO\_Strg komutu

LAD / FBD	SCL	Açıklama
	<pre>Chars_TO_Strg(   Chars:=_variant_in_,   pChars:=_dint_in_,   Cnt:=_uint_in_,   Strg=&gt;_string_out_);</pre>	<p>Karakterler dizisinin tamamı veya bir kısmı bir string'e kopyalanır.</p> <p>Chars_TO_Strg yürütülmeden önce çıkış string'i beyan edilmelidir. String daha sonra Chars_TO_Strg işlemi tarafından üzerine yazılır.</p> <p>Tüm desteklenen azami uzunluklardan (1 .. 254) oluşan string'ler kullanılabilir.</p> <p>String azami uzunluk değeri Chars_TO_Strg işlemi ile değiştirilmez. Azami dizi uzunluğuna ulaşıldığında diziden string'e kopyalama durur.</p> <p>Karakter dizisindeki bir nul karakteri '\$00' veya 16#00 değer bir ayraç olarak çalışır ve karakterlerin string içine kopyalanma işlemini bitirir.</p>

Çizelge 8- 38 Parametreler için veri tipleri (Chars\_TO\_Strg)

Parametre ve tipi	Veri tipi	Açıklama
Chars	IN	Chars parametresi, string'e dönüştürülecek sıfır tabanlı karakter dizisini [0 .. n] gösteren bir işaretçidir. Dizi, blok ara yüzündeki yerel değişkenler olarak veya bir DB içinde beyan edilebilir. Örnek: "DB1".MyArray, DB1'deki Char eleman değerlerinden oluşan MyArray [0..10]'i gösterir.
pChars	IN	Dizideki ilk karakter için kopyalanacak eleman sayısı. Dizi elemanı [0] varsayılan değerdir.
Cnt	IN	Kopyalanacak karakter sayısı: 0 hepsi demektir
Strg	OUT	Hedef string

## Geniřletilmiř komutlar

### 8.2 String ve karakter

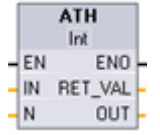
Çizelge 8- 39 ENO durumu

ENO	Açıklama
1	Hata yok
0	Chars_TO_Strg: Sting bildiriminde azami uzunluk bayt'ı ile izin verilenden daha fazla karakter bayt'ı çıkış string'ine kopyalamaya çalışır
0	Chars_TO_Strg: nul karakteri (16#00) değeri giriş karakter bayt dizisinde bulunmuřtur.
0	Strg_TO_Chars: sınırlı eleman sayısı ile izin verilenden daha fazla karakter bayt'ı çıkış dizisine kopyalamaya çalışır

#### 8.2.3.3 ATH ve HTA (ASCII string ve onaltılık sayıya veya bunlardan dönüşüm) komutu

ATH (ASCII'den onaltılık sisteme) ve HTA (onaltılık sistemden ASCII'ye) komutları, ASCII karakter bayt'lar (yalnızca 0 ila 9 karakterleri ve A ila F büyük harfleri) ile karşılık gelen 4-bit onaltılık sistem arasındaki dönüşüm için kullanılır.

Çizelge 8- 40 ATH komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := ATH(     in:=_variant_in_,     n:=_int_in_,     out=&gt;_variant_out_);</pre>	ASCII karakterlerini paketlenmiş onaltılık sayıya dönüřtürür.

Çizelge 8- 41 ATH komutu için veri tipleri

Parametre tipi	Veri tipi	Açıklama	
IN	IN	Variant	ASCII karakter byte dizisini gösteren işaretçi
N	IN	UInt	Dönüřtürülecek ASCII karakter bayt'ların sayısı
RET_VAL	OUT	Word	Yürütme durumu kodu
OUT	OUT	Variant	Dönüřtürülen onaltılık bayt dizisini gösteren işaretçi

Dönüřüm, IN parametresiyle belirtilen konumda başlar ve N bayt için devam eder. Sonuç, OUT'la belirtilen konuma yerleřtirilir. Sadece geçerli ASCII karakterler 0-9, küçük harfler a-f ve büyük harf A-F dönüřtürülebilir. Bařka bir karakter sıfıra dönüřtürülür.

8-bitlik ASCII kodlu karakterler 4-bit onaltılık sisteme dönüřtürülür. İki ASCII karakteri iki 4-bit onaltılık sistem içeren tek bir bayt'a dönüřtürülebilir.

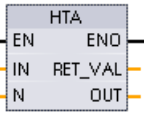
IN ve OUT parametreleri onaltılık String verileri deęil bayt dizileri belirtir. Okunduęu gibi ASCII karakterleri aynı sırayla onaltılık çıktıya dönüřtürülür ve yerleřtirilir. ASCII karakter bir tek sayı ise, o zaman sıfırlar son dönüřtürülen onaltılık sayının en saę dört bitine konur.



Çizelge 8- 42 ASCII-onaltılık (ATH) dönüşüm ile ilgili örnekler

IN karakter baytları	N	OUT değeri	ENO
'0a23'	4	W#16#0A23	TRUE
'123AFx1a23'	10	16#123AF01023	FALSE
'a23'	3	W#16#A230	TRUE

Çizelge 8- 43 HTA komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := HTA(   in:=_variant_in_,   n:=_uint_in_,   out=&gt;_variant_out_);</pre>	Paketlenmiş onaltılık sayıyı bunlara karşılık gelen ASCII karakter baytlarına dönüştürür.

Çizelge 8- 44 HTA komutu için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN	IN	Variant
N	IN	UInt
RET_VAL	OUT	Word
OUT	OUT	Variant

Dönüşüm, IN parametresiyle belirtilen konumda başlar ve N bayt için devam eder. Her 4-bit yarım bayt tek bir 8-bitlik ASCII karakterine dönüştürülür ve çıkışta 2N ASCII karakter bayt üretir. Çıkıştaki tüm 2N baytlar ASCII karakterleri 0-9 ve A-F olarak yazılır. OUT parametresi bir string'i değil bir bayt dizisini belirtir.

Onaltılık baytın her yarım baytı okunduğu gibi aynı sırada bir karaktere dönüştürülür (ilk bir onaltılık sayının en sol yarım baytı ardından aynı baytın en sağ yarım baytı dönüştürülür).

Çizelge 8- 45 Onaltılık sayı-ASCII (HTA) dönüşümüyle ilgili örnekler

IN değeri	N	OUT karakter baytları	ENO (HTA komutunu yürütmeden sonra ENO daima TRUE olur)
W#16#0123	2	'0123'	TRUE
DW#16#123AF012	4	'123AF012'	TRUE

Çizelge 8- 46 ATH ve HTA durum kodları

RET_VAL (W#16#...)	Açıklama	ENO
0000	Hata yok	TRUE
0007	Geçersiz ATH giriş karakteri: Bir ASCII karakter (0-9 büyük harf A-F ya da küçük harf a-f) olmayan bir karakter bulundu.	FALSE

## Genişletilmiş komutlar

### 8.2 String ve karakter


RET_VAL (W#16#....)	Açıklama	ENO
8101	Kuraldışı veya geçersiz giriş işaretçisi, örneğin mevcut olmayan bir DB için bir erişim.	FALSE
8120	Giriş string'i geçersiz format, başka bir deyişle, max= 0, max=255, mevcut>max, veya işaretçideki onaylanan (rant) uzunluk < max	FALSE
8182	Giriş ara belleği N için çok küçük	FALSE
8151	Giriş ara belleği N için izin verilmeyen veri tipi	FALSE
8301	Kuraldışı veya geçersiz çıkış işaretçisi, örneğin mevcut olmayan bir DB için bir erişim	FALSE
8320	Çıkış string'i geçersiz format, başka bir deyişle, max= 0, max=255, mevcut>max, veya işaretçideki onaylanan (rant) uzunluk < max	FALSE
8382	Çıkış ara belleği N için çok küçük	FALSE
8351	Çıkış ara belleği N için izin verilmeyen veri tipi	FALSE

### 8.2.4 String işlem komutları

Kontrol programınız, operatör ekranının ve proses kütükleri için mesajlar oluşturmak üzere aşağıdaki string ve karakter komutlarını kullanabilir.

#### 8.2.4.1 MAX\_LEN (Bir karakter string'in maksimum uzunluğu) komutu

Çizelge 8- 47 Maksimum uzunluk komutu


LAD / FBD	SCL	Açıklama
	<pre>out := MAX_LEN(in);</pre>	<p>MAX_LEN (maksimum string uzunluğu), OUT çıkışında IN string'e atanan maksimum uzunluk değeri sağlar. Komutların yürütülmesi sırasında hatalar ortaya çıkarsa, o zaman boş bir string uzunluğu çıkış olacaktır.</p> <p>STRING veri tipi iki uzunluk içerir: ilk bayt maksimum uzunluğu verir ve ikinci bayt mevcut uzunluğu verir (bu geçerli karakterlerin mevcut sayısı).</p> <ul style="list-style-type: none"> <li>Karakter string'in maksimum uzunluğu köşeli parantez içinde her STRING tanımı için atanır. Bir string tarafından işgal edilen bayt sayısı maksimum uzunluktan 2 bayt daha büyüktür.</li> <li>Mevcut uzunluk gerçekte kullanılan karakter yerlerinin sayısını temsil eder. Mevcut uzunluk azami uzunluğa eşit ya da daha az olmalıdır.</li> </ul> <p>Mevcut bir string uzunluğu almak için LEN komutunu, karakter string'in maksimum uzunluğunu almak için MAX_LEN komutunu kullanınız.</p>

Çizelge 8- 48 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN	IN	String
OUT	OUT	DInt

### 8.2.4.2 LEN (Bir karakter string'in uzunluğunu belirleme) komutu

Çizelge 8- 49 Uzunluk komutu

LAD / FBD	SCL	Açıklama
	<code>out := LEN(in);</code>	LEN (uzunluk), çıkış OUT'daki IN string'in mevcut uzunluğunu sağlar. Boş bir string sıfır uzunluğa sahiptir.

Çizelge 8- 50 Parametreler için veri tipleri

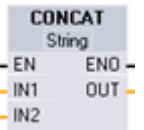
Parametre ve tipi	Veri tipi	Açıklama
IN	IN	String
OUT	OUT	Int, DInt, Real, LReal
		IN string'inin geçerli karakter sayısı

Çizelge 8- 51 ENO durumu

ENO	Durum	OUT
1	Geçersiz string durumu yok	Geçerli string uzunluk
0	Mevcut IN uzunluğu maksimum IN uzunluğunu aşar	Mevcut uzunluk sıfıra ayarlanır
	Maksimum IN uzunluğu ayrılan bellek aralığıyla uyumlu değil	
	Maksimum IN uzunluğu 255 (kuraldışı uzunluk)	

### 8.2.4.3 CONCAT ( karakter string'lerini birleştirme) komutu

Çizelge 8- 52 String'leri birleştirme komutu

LAD / FBD	SCL	Açıklama
	<code>out := CONCAT(in1, in2);</code>	CONCAT (string'leri birleştir), OUT'da sağlanan tek string'i oluşturacak şekilde IN1 ve IN2 string parametrelerini birleştirir. Birleştirme sonrası, String IN1 birleştirilen string'in sol bölümünü ve string IN2 ise sağ bölümünü oluşturur.

Çizelge 8- 53 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN1	IN	String
IN2	IN	String
OUT	OUT	String
		Birleştirilmiş string (string 1 + string 2)

## Geniřletilmiř komutlar

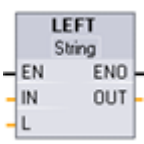
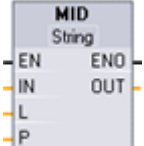
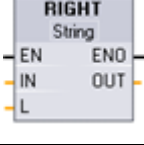
### 8.2 String ve karakter

Çizelge 8- 54 ENO durumu

ENO	Durum	OUT
1	Hiç bir hata tespit edilmedi	Geçerli karakterler
0	Birleřtirmeden sonra oluřan string, maksimum OUT string uzunluęundan daha büyüktür.	Sonuçta oluřan string karakterler maksimum OUT uzunluęuna ulařilana kadar kopyalanır
	Mevcut IN1uzunluęu maksimum IN1 uzunluęunu ařar, mevcut IN2 uzunluęu maksimum IN2 uzunluęunu ařar veya mevcut OUT uzunluęu maksimum OUT uzunluęunu ařar (geçersiz string)	Mevcut uzunluk 0'a ayarlanır
	IN1, IN2 veya OUT maksimum uzunluęu ayrılan bellekle uyumlu deęil	
	IN1 veya IN2 nin maksimum uzunluęu 255 veya maksimum OUT uzunluęu 0 veya 255	

#### 8.2.4.4 LEFT, RIGHT ve MID (bir karakter string'deki substringleri okuma) komutları

Çizelge 8- 55 Sol, saę ve orta substring iřlemleri

LAD / FBD	SCL	Açıklama
 <p>LEFT String ladder logic diagram showing EN, ENO, IN, OUT, and L terminals.</p>	<pre>out := LEFT(in, L);</pre>	<p>SOL (Sol substring), IN string parametresinin ilk L karakterinden oluřan bir substring saęlar.</p> <ul style="list-style-type: none"> <li>L, mevcut IN string uzunluęundan büyükse, tüm IN string'i OUT'a döndürülür.</li> <li>Boř bir string, giriş ise, OUT'a boř bir string döndürülür.</li> </ul>
 <p>MID String ladder logic diagram showing EN, ENO, IN, OUT, L, and P terminals.</p>	<pre>out := MID(in, L, p);</pre>	<p>MID (Orta substring) bir string'in orta bölümünü saęlar. Orta substring L karakter uzunluęundadır ve P karakter konumunda bařlar (bu konum dahil).</p> <p>L ve P toplamı IN string parametrenin mevcut uzunluęunu ařarsa, o zaman P karakter konumunda bařlayan ve IN string sonuna kadar devam eden bir substring döndürülür.</p>
 <p>RIGHT String ladder logic diagram showing EN, ENO, IN, OUT, and L terminals.</p>	<pre>out := RIGHT(in, L);</pre>	<p>RIGHT (Saę substring), bir string'in son L karakterini saęlar.</p> <ul style="list-style-type: none"> <li>L, mevcut IN string uzunluęundan büyükse, tüm IN string'i OUT'a döndürülür.</li> <li>Boř bir string, giriş ise, OUT'a boř bir string döndürülür.</li> </ul>

Çizelge 8- 56 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
IN	IN	String	Giriş string'i
L	IN	Int	Oluşturulacak substring uzunluğu: <ul style="list-style-type: none"> <li>• LEFT, string'deki karakterlerin en sol karakter sayısını kullanır</li> <li>• RIGHT, string'deki karakterlerin en sağ karakter sayısını kullanır</li> <li>• MID, string içinde P pozisyonundan başlayan karakter sayısını kullanır</li> </ul>
P	IN	Int	Yalnızca MID: Kopyalanacak ilk substring karakter pozisyonu P= 1, IN string'in başlangıç karakter konumu için
OUT	OUT	String	Çıkış string

Çizelge 8- 57 ENO durum

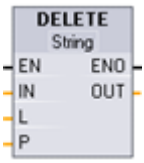
ENO	Durum	OUT
1	Hiç bir hata tespit edilmedi	Geçerli karakterler
0	<ul style="list-style-type: none"> <li>• L ya da P, sıfırdan az veya sıfıra eşittir</li> <li>• P, maksimum IN uzunluğundan büyüktür</li> <li>• Mevcut IN uzunluğu, maksimum IN uzunluğunu aşar veya mevcut OUT uzunluğu maksimum OUT uzunluğunu aşar</li> <li>• Maksimum IN veya OUT uzunluğu ayrılan bellek aralığına uymaz</li> <li>• Maksimum IN veya OUT uzunluğu 0 veya 255'dir</li> </ul>	Mevcut uzunluk 0' a ayarlanır
	Kopyalanacak Substring uzunluğu (L) maksimum OUT string uzunluğundan büyük.	Karakterler maksimum OUT uzunluğuna ulaşılan kadar kopyalanır
	MID yalnızca: L ya da P, 0'a eşit ya da daha azdır	Mevcut uzunluk 0' a ayarlanır
	MID yalnızca: P, maksimum IN uzunluğundan büyüktür	
	Mevcut IN1 uzunluğu maksimum IN1 uzunluğunu aşar veya mevcut IN2 uzunluğu maksimum IN2 uzunluğunu aşar (geçersiz string)	Mevcut uzunluk 0' a ayarlanır
	Maksimum IN1, IN2 veya OUT uzunluğu ayrılan belleğe uymaz	
	Maksimum IN1, IN2 veya OUT uzunluğu 0 veya 255 (kuraldışı uzunluk)	

## Genişletilmiş komutlar

### 8.2 String ve karakter

#### 8.2.4.5 DELETE (bir karakter string'deki karakterleri silme) komutu

Çizelge 8- 58 Substring silme komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a 'DELETE String' instruction. It has four inputs: EN (Enable), IN (Input string), L (Number of characters to delete), and P (Starting position). It has two outputs: ENO (Enable Out) and OUT (Output string).</p>	<pre>out := DELETE(in, L, p);</pre>	<p>IN string'den L karakterlerini siler. Karakter silme P karakter pozisyonunda (dahil) başlar ve geriye kalan substring OUT parametresinde sağlanır.</p> <ul style="list-style-type: none"> <li>L sıfıra eşitse, giriş string'i OUT'a döndürülür.</li> <li>L ve P toplamı giriş string'i uzunluğundan daha büyükse, o zaman string sonuna kadar silinir.</li> </ul>

Çizelge 8- 59 Parametreler için veri tipleri

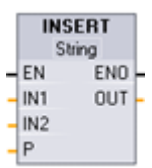
Parametre ve tipi	Veri tipi	Açıklama
IN	String	Giriş string'i
L	Int	Silinecek karakter sayısı
P	Int	Silinecek ilk karakter pozisyonu: IN string'in ilk karakterinin pozisyon numarası 1
OUT	String	Çıkış string'i

Çizelge 8- 60 ENO durumu

ENO	Durum	OUT
1	Hiç bir hata tespit edilmedi	Geçerli karakterler
0	P mevcut IN uzunluğundan büyük	Karakterler silinmeden IN, OUT'a kopyalanır.
	Karakterler silindikten sonra ortaya çıkan string maksimum OUT string uzunluğundan daha büyüktür	Sonuçta oluşan karakterler maksimum OUT uzunluğuna ulaşılan kadar kopyalanır
	L, 0'dan küçük veya P, 0'dan küçük veya eşit	Mevcut uzunluk 0'a ayarlanır
	Mevcut IN uzunluğu maksimum IN uzunluğunu aşar veya mevcut OUT uzunluğu maksimum OUT uzunluğunu aşar	
	Maksimum IN veya OUT uzunluğu ayrılan belleğe uymaz	
	Maksimum IN veya OUT uzunluğu 0 veya 255	

### 8.2.4.6 INSERT (karakterleri bir karakter string'ine yerleřtirme) komutu

Çizelge 8- 61 Substring komutunu yerleřtirme

LAD / FBD	SCL	Açıklama
	<pre>out := INSERT(in1, in2, p);</pre>	String IN2'yi string IN1 içine yerleřtirir. Yerleřtirme P pozisyonundaki karakterden sonra bařlar.

Çizelge 8- 62 Parametreler için veri tipleri

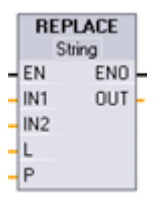
Parametre ve tipi	Veri tipi	Açıklama	
IN1	IN	String	Giriř string'i 1
IN2	IN	String	Giriř string'i 2
P	IN	Int	String IN2 için yerleřtirme noktasından önce string IN1'deki son karakter pozisyonu String IN1'in ilk karakterinin pozisyon numarası 1.
OUT	OUT	String	Sonuç string

Çizelge 8- 63 ENO durumu

ENO	Durum	ÇIKIŞ (OUT)
1	Hiç bir hata tespit edilmedi	Geçerli karakterler
0	P, IN1uzunluęundan büyük	IN2, son IN1 karakterinden hemen sonra IN1 ile birleřtirilir
	P, 0'dan küçük	Mevcut uzunluk 0'a ayarlanır
	Yerleřtirmeden sonra sonuçta oluřan string, maksimum OUT string uzunluęundan büyük	Sonuçta oluřan string karakterler maksimum OUT uzunluęuna ulařilana kadar kopyalanır
	Mevcut IN1 uzunluęu maksimum IN1 uzunluęunu ařar, mevcut IN2 uzunluęu maksimum IN2 uzunluęunu ařar veya mevcut OUT uzunluęu maksimum OUT uzunluęunu ařar (geçersiz string)	Mevcut uzunluk 0'a ayarlanır
	IN1, IN2 veya OUT'un maksimum uzunluęu ayrılan bellek aralıęına uymaz	
IN1 veya IN2'nin maksimum uzunluęu 255 veya maksimum OUT uzunluęu 0 veya 255		

### 8.2.4.7 REPLACE (bir karakter string'indeki karakterleri deęiřtirme) komutu

Çizelge 8- 64 Substring deęiřtirme komutu

LAD / FBD	SCL	Açıklama
 <p>REPLACE String EN ENO IN1 OUT IN2 L P</p>	<pre>out := REPLACE (     in1:=_string_in_,     in2:=_string_in_,     L:=_int_in_,     p:=_int_in_);</pre>	<p>IN1 string parametresi içindeki L karakterlerini deęiřtirir. Deęiřtirme, IN2 string parametresinden gelen deęiřtirme karakterleri ile, P pozisyonundaki (dahil) string IN1 karakterinden başlar.</p>

Çizelge 8- 65 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
IN1	IN	String
IN2	IN	String
L	IN	Int
P	IN	Int
OUT	OUT	String

Parametre L sıfıra eřitse string IN2, string IN1'den hiçbir karakter silmeksizin IN1 string'inin P pozisyonuna yerleřtirilir.

P bire eřit ise, o zaman string IN1'in ilk L karakteri string IN2 karakteriyle deęiřtirilir.


Çizelge 8- 66 ENO durumu

ENO	Durum	ÇIKIř
1	Hiç bir hata tespit edilmedi	Geçerli karakterler
0	P, IN1uzunluęundan büyük	IN2, son IN1 karakterinden hemen sonra IN1 ile birleřtirilir
	P, IN1 içini iřaret eder, ancak IN1'de kalan L karakterlerinden azdır	IN2, P pozisyonundan başlayarak IN1'in son karakterlerinin yerine geçer
	Deęiřtirmeden sonrasında oluřan string, maksimum OUT string uzunluęundan büyüktür	Sonuçta oluřan string karakterler maksimum OUT uzunluęuna ulařılana kadar kopyalanır
	Maksimum IN1 uzunluęu 0'dır	IN2 karakterler OUT'a kopyalanır
	L, 0'dan küçük veya P, 0'dan küçük veya eřit	Mevcut uzunluk 0'a ayarlanır
	Mevcut IN1 uzunluęu maksimum IN1 uzunluęunu ařar, mevcut IN2 uzunluęu maksimum IN2 uzunluęunu ařar veya mevcut OUT uzunluęu maksimum OUT uzunluęunu ařar	
	Maksimum IN1, IN2 veya OUT uzunluęu ayrılan bellek aralıęına uymaz	
Maksimum IN1 veya IN2 uzunluęu 255, veya maksimum OUT uzunluęu 0 veya 255		



### 8.2.4.8 FIND (karakter string içindeki karakterleri bulma) komutu

Çizelge 8- 67 Substring'i bulma komutu

LAD / FBD	SCL	Açıklama
	<pre>out := FIND(     in1:=_string_in_,     in2:=_string_in);</pre>	<p>String IN1 içinde IN2 tarafından belirtilen substring karakter konumunu sağlar. Arama soldan başlar. IN2 string ilk oluştuğu karakter konumu OUT'a döndürülür. String IN2, string IN1 içinde bulunmazsa, sıfır döndürülür.</p>

Çizelge 8- 68 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
IN1	IN	String	Bu string'in içinde ara
IN2	IN	String	Bu string'i ara
OUT	OUT	Int	İlk arama eşleşmesinin IN1 string'i içindeki karakter pozisyonu

Çizelge 8- 69 ENO durumu

ENO	Durum	ÇIKIŞ
1	Hiç bir hata tespit edilmedi	Geçerli karakter pozisyonu
0	IN2, IN1'den büyük	Karakter pozisyonu 0'a ayarlanır
	Mevcut IN1 uzunluğu maksimum IN1 uzunluğunu aşar, mevcut IN2 uzunluğu maksimum IN2 uzunluğunu aşar (geçersiz string)	
	Maksimum IN1 veya IN2 uzunluğu ayrılan bellek aralığına uymaz	
	Maksimum IN1 veya IN2 uzunluğu 255	

## 8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-i)

### 8.3.1 Dağıtılmış I/O Komutları

Aşağıdaki dağıtılmış I/O komutları PROFINET, PROFIBUS veya AS-I ile birlikte kullanılabilir:

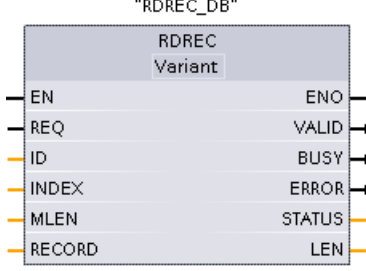
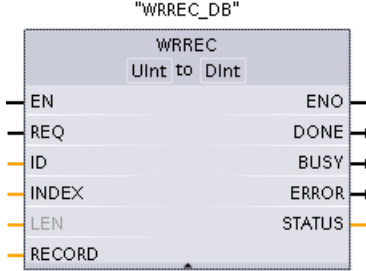
- RDREC komutu (Sayfa 315): Bir modül veya cihazdan INDEX numarası ile bir veri kaydını okuyabilirsiniz.
- WRREC komutu (Sayfa 315): ID ile tanımlanan bir modül veya cihaza INDEX numarası ile bir veri kaydı aktarabilirsiniz.
- RALRM komutu (Sayfa 318): Bir modül veya cihazdan alınan tüm karşılık gelen bilgilerle bir kesme alabilir ve bu bilgileri bunun çıkış parametrelerine temin edebilirsiniz.
- DPRD\_DAT komutu (Sayfa 326): Bir modül ya da cihazdan gelen 64 bayttan daha fazla tutarlı veri alanlarını DPRD\_DAT komutu ile okumanız gerekir.
- DPWR\_DAT komutu (Sayfa 326): Bir modül ya da cihazdan gelen 64 bayttan daha fazla tutarlı veri alanlarını DPRW\_DAT komutu ile yazmanız gerekir.

DPNRM\_DG komutu (Sayfa 326) sadece PROFIBUS ile kullanılabilir. EN 50 170 Volume 2 PROFIBUS tarafından belirtilen formatta DP slave'in mevcut tanı verilerini okuyabilirsiniz.

### 8.3.2 RDREC ve WRREC (Veri kaydını okuma/yazma) komutları

PROFINET, PROFIBUS ve AS-i ile RDREC (veri kaydı oku) ve WRREC (veri kaydı yaz) komutlarını kullanabilirsiniz.

Çizelge 8- 70 RDREC ve WRREC komutları

LAD / FBD	SCL	Açıklama
<p>"RDREC_DB"</p> 	<pre>"RDREC_DB" (   req:=_bool_in_,   ID:=_word_in_,   index:=_dint_in_,   mlen:=_uint_in_,   valid=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_dword_out_,   len=&gt;_uint_out_,  record:=_variant_inout_);</pre>	<p>Merkezi şasiye veya dağıtılmış bileşen (PROFIBUS DP veya PROFINET IO) gibi ID ile adreslenen bileşenden INDEX numarası ile bir veri kaydını okumak için RDREC komutu kullanınız.</p> <p>Okuma yapmak için maksimum bayt sayısını MLEN'e atayınız. Hedef alan kaydının seçilen uzunluğu en azından MLEN bayt uzunluğuna sahip olmalıdır.</p>
<p>"WRREC_DB"</p> 	<pre>"WRREC_DB" (   req:=_bool_in_,   ID:=_word_in_,   index:=_dint_in_,   len:=_uint_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_dword_out_,  record:=_variant_inout_);</pre>	<p>Merkezi şasiye veya dağıtılmış bileşendeki (PROFIBUS DP veya PROFINET IO) bir modül gibi ID ile adreslenen bir DP slave/PROFINET IO cihaz bileşenine, INDEX kayıt numarası ile bir veri kaydı aktarmak için WRREC komutunun kullanınız.</p> <p>İletilecek veri kaydının bayt uzunluğunu atayınız. Kaynak alan kaydının seçilen uzunluğu, bu nedenle, en azından LEN bayt uzunluğuna sahip olmalıdır.</p>

- 1 Komutu yerleřtirdiğinizde STEP 7 otomatik olarak DB oluşturur.
- 2 SCL örneklerinde "RDREC\_DB" ve "WRREC\_DB", kopya DB'lerin isimleridir.

Geniřletilmiř komutlar

8.3 Dađıtılmıř I/O (PROFINET, PROFIBUS veya AS-I)

Çizelge 8- 71 RDREC ve WRREC parametreleri için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
REQ	IN	Bool	REQ = 1: veri kaydı transferi
ID	IN	HW_IO (Word)	<p>DP slave / PROFINET IO bileřen mantıksal adresi (modül veya alt modül):</p> <ul style="list-style-type: none"> <li>Bir çıkıř modülü için bit 15 ayarlanmıř olmalıdır (örneğin, adres 5 için: ID:= DW#16#8005).</li> <li>Bir kombinasyon modülü için iki adresten küçük olanı belirtilmelidir.</li> </ul> <p><b>Not:</b> V3.0'da, cihaz ID iki řekilde belirlenebilir:</p> <ul style="list-style-type: none"> <li>Ařađıdaki "Network view" seçimlerini yaparak: <ul style="list-style-type: none"> <li>Cihaz (gri kutu)</li> <li>Cihazın "özellikleri"</li> <li>"Donanım kimliđi"</li> </ul> <p><b>Not:</b> Ancak tüm cihazlar Donanım kimliklerini görüntüler.</p> </li> <li>Ařađıdaki "Project tree" menü seçimlerini yaparak: <ul style="list-style-type: none"> <li>PLC etiketleri</li> <li>Varsayılan etiket çizelgesi</li> <li>Sistem sabitleri sekmesi</li> </ul> <p>Tüm yapılandırılan cihaz donanım kimlikleri görüntülenir.</p> </li> </ul> <p><b>Not:</b> V4.0'da, ara yüz modülü için kullanılan cihaz ID'si (donanım kimliđi) etiket çizelgesine giderek ve Sistem sabitleri altında "Device Name [HEAD]" parametresini yerleřtirerek belirlenir.</p>
INDEX	IN	Byte, Word, USInt, UInt, SInt, Int, DInt	Veri kayıt numarası
MLEN	IN	Byte, USInt, UInt	Bayt cinsinden getirilecek maksimum uzunluktaki veri kayıt bilgisi (RDREC)
VALID	OUT	Bool	Yeni veri kaydı alındı ve geçerli (RDREC) oldu. Son istek hatasız olarak tamamlandıktan sonra VALID biti, bir tarama için TRUE olur.
DONE	OUT	Bool	Veri kaydı (WRREC) transfer edildi. Son istek hatasız tamamlandıktan sonra DONE biti, bir tarama için TRUE olur.
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>BUSY = 1: Oku (RDREC) veya yaz (WRREC) iřlemi henüz sonlanmadı.</li> <li>BUSY = 0: Veri kaydı transferi tamamlandı.</li> </ul>
ERROR	OUT	Bool	ERROR = 1: Bir okuma (RDREC) veya yazma (WRREC) hatası oluřtu. Son istek bir hata ile sonlandıktan sonra ERROR biti, bir tarama için TRUE olur. STATUS parametresinde hata kodu deđeri, yalnızca ERROR= TRUE olduđunda tekli tarama sırasında geçerlidir.
STATUS	OUT	Dword	Blok durum veya hata bilgisi

Parametre ve tipi		Veri tipi	Açıklama
LEN	OUT (RDREC) IN (WRREC)	UInt	<ul style="list-style-type: none"><li>Alınan veri kayıt bilgilerinin uzunluğu (RDREC)</li><li>Transfer edilecek veri kaydının maksimum bayt uzunluğu (WRREC)</li></ul>
RECORD	IN_OUT	Variant	<ul style="list-style-type: none"><li>Getirilen veri kaydı için hedef alan (RDREC)</li><li>Veri kaydı (WRREC)</li></ul>

RDREC ve WRREC komutları asenkron olarak çalışır, yani işlem çoklu komut çağrılarını kapsar. REQ = 1 ile RDREC veya WRREC çağırarak işi başlatınız.

İş durumu, BUSY çıkış parametresi üzerinden ve STATUS çıkış parametresinin iki merkez baytı üzerinden gösterilir. BUSY çıkış parametresi FALSE olarak ayarlandığında veri kaydının transferi tamamlanır.

VALID (RDREC) veya DONE ( WRREC ) çıkış parametresi üzerindeki TRUE değeri (yalnızca bir tarama ) veri kaydının başarıyla hedef alan KAYIT ( RDREC ) içine ya da hedef cihaza transfer edildiğini doğrular. RDREC durumunda, çıkış parametresi LEN bayt olarak getirilen verilerin uzunluğunu içerir.

Çıkış parametresi ERROR (ERROR = TRUE olduğunda sadece bir tarama için) bir veri kaydı transfer hatası oluştuğunu gösterir. Böyle bir durumda, çıkış parametresi STATUS (ERROR = TRUE olduğunda sadece bir tarama için) hata bilgilerini içerir.

Veri kayıtları donanım aygıt üreticisi tarafından tanımlanır. Bir veri kaydı hakkında detaylı bilgi için donanım üreticisinin cihaz dokümantasyonuna bakılmalıdır.

#### Not

Bir DPV1 slave, GSD dosyası üzerinden yapılandırılmış ise (GSD rev. 3 ve üstü) ve DP master'ın DP ara yüzü "S7 uyumlu" olarak ayarlandı ise, daha sonra "RDREC" ile kullanıcı programında I / O modüllerinden herhangi bir veri kaydı okunamayabilir veya "WRREC" ile I/O modüllerine yazılamayabilir. Bu durumda, DP master yanlış slotu adresler (yapılandırılmış slot + 3).

Çözüm: DP master ara yüzünü "DPV1" e ayarlamak.

#### Not

"RDREC" ve "WRREC" komut ara yüzleri "IEC 61131-3 göre PROFIBUS Rehber PROFIBUS Haberleşme ve Proxy Fonksiyon Blokları" olarak tanımlanan "RDREC" ve "WRREC" FB'lerle aynıdır.

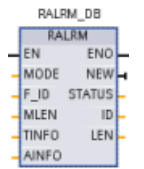
#### Not

PROFINET IO için bir veri kaydını okumak veya yazmak üzere "RDREC" veya "WRREC" kullanırsanız INDEX, MLEN ve LEN parametrelerindeki negatif değerler unsigned 16-bit integer olarak yorumlanır.

### 8.3.3 RALRM (Kesme alma) komutu

PROFINET ve PROFIBUS ile RALRM (alarm okuma) komutunu kullanabilirsiniz.

Çizelge 8- 72 RALRM komutu

LAD / FBD	SCL	Açıklama
	<pre>"RALRM_DB" (     mode:=_int_in_,     f_ID:=_word_in_,     mlen:=_uint_in_,     new=&gt;_bool_out_,     status=&gt;_dword_out_,     ID=&gt;_word_out_,     len=&gt;_uint_out_,     tinfo:=_variant_inout_,     ainfo:=_variant_inout_);</pre>	<p>PROFIBUS veya PROFINET I / O modülleri / cihazlarından gelen tanılama kesme bilgilerini okumak için RALRM (alarm okuma) komutunu kullanınız.</p> <p>Çıkış parametrelerindeki bilgi, kesme kaynak bilgilerinin yanı sıra OB denilen başlat bilgilerinin de içerir.</p> <p>Kesmeye neden olay (lar) hakkında bilgi dönüşü için bir OB kesmesi içinde RALRM çağırınız. S7-1200 içinde. Aşağıdaki tanılama OB kesmeleri desteklenir: Durum, güncelleme, profil, tanılama hata kesmesi, çek veya çıkar modülleri, şasi veya istasyon arızası.</p>

- 1 Komutu yerleřtirdiğinizde STEP 7 otomatik olarak DB oluşturur.
- 2 SCL örneğinde, "RALRM\_DB ", kopya DB'lerin ismidir.

Çizelge 8- 73 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
MODE	IN	Byte, USInt, SInt, Int
F_ID	IN	HW_IO (Word)
MLEN	IN	Byte, USInt, UInt
NEW	OUT	Bool
STATUS	OUT	Dword
ID	OUT	HW_IO (Word)

Çalışma modu

Kesmelerin alınacağı bileşenin (modülün) mantıksal başlangıç adresi

**Not:** Cihaz ID'si iki şekilde belirlenebilir:

- Aşağıdaki "Network view" seçimlerini yaparak:
  - Cihaz (gri kutu)
  - Cihazın "özellikleri"
  - "Donanım kimliği"**Not:** Ancak tüm cihazlar Donanım kimliklerini görüntüler.
- Aşağıdaki "Project tree" menü seçimlerini yaparak:
  - PLC etiketleri
  - Varsayılan etiket çizelgesi
  - Sistem sabitleri sekmesi
  - Tüm yapılandırılan cihaz donanım kimlikleri görüntülenir.

Alınacak veri kesme bilgisinin bayt cinsinden maksimum uzunluğu. 0'ın MLEN'i, AINFO Hedef Bölgesindeki kadar çok veri kesme bilgisi alınmasına izin verecektir.

Yeni bir kesme alındı.

RALRM komutu durumu. Daha fazla bilgi için " RDREC, WRREC ve RALRM için STATUS parametresi " ne (Sayfa 322) bakınız.

Tanılayıcı kesmeye neden olan I/O modülüne ait donanım Tanımlayıcı.

**Not:** Cihaz ID'sinin nasıl belirleneceğiyle ilgili açıklama için F\_ID parametresine bakınız.

Parametre ve tipi		Veri tipi	Açıklama
LEN	OUT	Dword, UInt, UInt, DInt, Real, LReal	Alınan AINFO kesme bilgisinin uzunluğu
TINFO	IN_OUT	Variant	Görev bilgisi: OB başlatması için hedef aralık ve yönetim bilgisi. TINFO uzunluğu her zaman 32 bayt.
AINFO	IN_OUT	Variant	Kesme bilgisi: Başlık bilgisi için hedef alanı ve ek kesme bilgisi. MLEN 0'dan büyük ise, AINFO için, en azından MLEN bayt uzunluğunu sağlayınız. AINFO uzunluğu deęiřkendir.

#### Not

Başlatma olayı bir I / O kesmesi olmayan bir OB'de "RALRM" komutunu çağırırsanız, komut çıkışlarında buna baęlı olarak azaltılmış bilgi sağlayacaktır.

Farklı OB'lerde "RALRM" komutunu çağırduğunuzda farklı kopya DB'ler kullandığınızdan emin olunuz. İlişkili kesme OB dışında bir "RALRM" çağırısından ortaya çıkan verileri deęerlendirirseniz, OB başlatma olayı başına ayrı bir kopya DB kullanmalısınız.

#### Not

"RALRM" komut ara yüzü "IEC 61131-3'e uygun olarak PROFIBUS Kılavuz PROFIBUS Haberleşme ve Proxy Fonksiyon Blokları" olarak tanımlanan " RALRM " FB ile aynıdır.

## RALRM Çaęırma

Üç farklı çalışma modunda (MODE) RALRM komutunu çağırabilirsiniz.

Çizelge 8- 74 RALRM komutu çalışma modları

MOD	Açıklama
0	<ul style="list-style-type: none"> <li>ID, kesmeyi tetikleyen I / O modülünün donanım kimliğini içerir.</li> <li>NEW Çıkış parametresi TRUE olarak ayarlanır.</li> <li>LEN, 0 çıkış üretir.</li> <li>AINFO ve TINFO herhangi bir bilgiyle güncelleştirilmez.</li> </ul>
1	<ul style="list-style-type: none"> <li>ID, kesmeyi tetikleyen I / O modülünün donanım kimliğini içerir.</li> <li>NEW çıkış parametresi TRUE olarak ayarlanır.</li> <li>LEN bayt cinsinden döndürülen AINFO veri miktarı çıktısı üretir.</li> <li>AINFO ve TINFO kesme ilişkili bilgiyle güncelleştirilir.</li> </ul>
2	<p>F_ID giriş parametresine atanan donanım kimliği kesmeyi tetiklediyse:</p> <ul style="list-style-type: none"> <li>ID kesmeyi tetikleyen I / O modülünün donanım kimliğini içerir. F_ID'deki deęer ile aynı olmalıdır.</li> <li>Çıkış parametresi NEW TRUE olarak ayarlanmalıdır.</li> <li>LEN, döndürülen AINFO veri bayt miktarı olarak bir çıktı üretir.</li> <li>AINFO ve TINFO kesmeyle ilişkili bilgiyle güncellenir.</li> </ul>

---

**Not**

Çok kısa olan TINFO veya AINFO için bir hedef alan atarsanız, RALRM tam bilgiyi döndüremez.

MLEN döndürülen AINFO veri miktarını sınırlayabilir.

AINFO ve TINFO verilerinin nasıl yorumlanacağı hakkında bilgi için STEP 7 çevrimiçi bilgi sistemindeki AINFO ve TINFO parametrelerine bakınız.

---



### TInfo organizasyon blok verisi

Aşağıdaki çizelge TInfo verilerinin RALRM komutu için nasıl düzenlendiğini gösterir:

OB'ler için aynı: Durum, Güncelleme, Profil, Tanılama hata kesmesi, Çek veya çıkar modülleri, Şasi veya istasyon arızası	0	SI_Format	OB_Class	OB_Nr	
	4	LADDR			
TI_Submodule – OB'ler: Durum, Güncelleme, Profil	4			Slot	
	8	Belirtici		0	
TI_DiagnosticInterrupt - OB: Tanılama hata kesmesi	4			IO_State	
	8	Kanal		MultiError	0
TI_PlugPullModule - OB: Çek veya çıkar modülleri	4			Event_Class	Fault_ID
	8	0		0	
TI_StationFailure - OB: Şasi veya istasyon arızası	4			Event_Class	Fault_ID
	8	0		0	
OB'ler için aynı: Durum, Güncelleme, Profil, Tanılama hata kesmesi, Çek veya çıkar modülleri, Şasi veya istasyon arızası	12	0			
	16				
	20	adres		slv_prfl	intr_type
	24	flags1	flags2	id	
	28 <sup>1</sup>	imalatçı		kopya	

<sup>1</sup> 28 - 31 Baytları (imalatçı ve kopya) PROFIBUS ile kullanılmaz.

### Not

TINFO verileri hakkında ayrıntılı bilgi için STEP 7 çevrimiçi bilgi sistemine bakınız.

## Geniştirilmiş komutlar

### 8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-I)

#### 8.3.4 RDREC, WRREC ve RALRM için STATUS parametresi

Çıkış parametresi STATUS, BYTE DİZİSİ [1 ... 4] olarak yorumlanan hata bilgilerini aşağıdaki yapıyla içerir:

Çizelge 8- 75 STATUS çıkış dizisi

Dizi elemanı	İsim	Açıklama
STATUS [1]	Function_Num	<ul style="list-style-type: none"> <li>B#16#00, hata yoksa</li> <li>DPV1-PDU' den gelen Fonksiyon ID: Hata olursa, B#16#80 VEYA'lanır (veri kaydını okumak için: B#16#DE; veri kaydını yazmak için: B#16#DF). DPV1 protokol elemanı kullanılmazsa, B#16#C0 çıkış olacaktır.</li> </ul>
STATUS [2]	Error Decode	Hata ID'sinin konumu
STATUS [3]	Error_Code_1	Hata ID'si
STATUS [4]	Error_Code_2	İmalatçıya-özümlenmiş hata ID genişlemesi

Çizelge 8- 76 STATUS [2] değerleri

Hata_kod çözme (B#16#...)	Kaynak	Açıklama
00 - 7F	CPU	Hata veya uyarı yok
80	DPV1	IEC 61158-6'ya göre hata
81 - 8F	CPU	B#16#8x komutun "x." çağrı parametresinde bir hata gösterir.
FE, FF	DP Profile	Profile-özümlenmiş hata

Çizelge 8- 77 STATUS [3] değerleri

Hata_kod çözme (B#16#...)	Hata_kod_1 (B#16#...)	Tarif (DVP1)	Açıklama
00	00		Hata, uyarı yok
70	00	Ayrılmış, ret	Başlangıç çağrısı; etkin veri kayıt transferi yok
	01	Ayrılmış, ret	Başlangıç çağrısı; veri kayıt transferi başladı
	02	Ayrılmış, ret	Acil çağrı; veri kayıt transferi çoktan aktif
80	90	Ayrılmış, geçti	Geçersiz mantıksal başlangıç adresi
	92	Ayrılmış, geçti	Varyant işaretçi için kural dışı tip
	93	Ayrılmış, geçti	ID veya F_ID yoluyla adreslenen DP bileşeni yapılandırılmamış.

## 8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-i)

Hata_kod çözme (B#16#....)	Hata_kod_1 (B#16#....)	Tarif (DVP1)	Açıklama
	96		"RALRM (Sayfa 318)"; OB başlat bilgisi, yönetim bilgisi, başlık bilgisi veya ek kesme bilgisi sağlayamaz. Aşağıdaki OB'ler için (OB başlat bilgisinden gelen adres bilgileri) asenkron bir şekilde ilgili DP slave'in mevcut tanılayıcı mesaj çerçevesini okumak üzere "DPNRM_DG (Sayfa 328)" komutunu kullanabilirsiniz: <ul style="list-style-type: none"> <li>• Donanım kesmesi (Sayfa 87)</li> <li>• Durum (Sayfa 993), Güncelleme (Sayfa 93) veya Profil (Sayfa 94)</li> <li>• Tanılayıcı hata kemesi (Sayfa 89)</li> <li>• Çek veya çıkar modülleri (Sayfa 91)</li> </ul>
	A0	Okuma hatası	Modülden okurken olumsuz onay
	A1	Yazma hatası	Modülden yazarken olumsuz onay
	A2	Modül arızası	Katman 2'de DP protokol hatası (örneğin, slave hatası veya veri yolu problemleri için)
	A3	Ayrılmış, geçti	<ul style="list-style-type: none"> <li>• PROFIBUS DP: Doğrudan-Veri-Link-Eşleyici veya Kullanıcı-Ara yüz/Kullanıcı ile DP protokol hatası</li> <li>• PROFINET IO: Genel CM hatası</li> </ul>
	A4	Ayrılmış, geçti	İletişim veri yolu üzerindeki iletişim bozulur
	A5	Ayrılmış, geçti	-
	A7	Ayrılmış, geçti	DP slave veya modüller işgal ediliyor (geçici hata)
	A8	Versiyon çakışması	DP slave veya modül uyumlu olmayan versiyonları bildiriyor.
	A9	Desteklenmeyen özellik	DP slave veya modül tarafından desteklenmeyen özellik
	AA - AF	Kullanıcıya özgü	DP slave veya modül kendi uygulamasında üreticiye-özel bir hata bildiriyor. DP slave veya modül üreticisinden gelen belgeleri lütfen kontrol ediniz.
	B0	Geçersiz indeks	Modülde bilinmeyen veri kaydı; kuraldışı veri kayıt numarası $\geq 256$
	B1	Yazma uzunluk hatası	RECORD parametresinde uzunluk bilgisi yanlıştır. <ul style="list-style-type: none"> <li>• "RALRM" ile: AINFO da uzunluk hatası</li> </ul> <p><b>Not:</b> "AINFO" döndürülen arabelleklerin nasıl yorumlanacağı hakkında bilgilere anında erişim için STEP 7 çevrimiçi bilgi sistemine bakınız.</p> <ul style="list-style-type: none"> <li>• "RDREC (Sayfa 31515)" ve "WRREC (Sayfa 31515)" ile: "MLEN" de uzunluk hatası</li> </ul>
	B2	Geçersiz slot	Yapılandırılmış slot işgal edilmemiş.
	B3	Tip çakışması	Gerçek modül tipi belirtilen modül tipi ile eşleşmiyor.
	B4	Geçersiz alan	DP slave veya modül, geçersiz alana erişim yapıldığını bildiriyor.
	B5	Durum çakışması	DP slave veya modül hazır değil
	B6	Erişim reddedildi	DP slave veya modül erişimi reddediyor.

## Genişletilmiş komutlar

## 8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-I)

Hata_kod çözme (B#16#...)	Hata_kod_1 (B#16#...)	Tarif (DVP1)	Açıklama
	B7	Geçersiz aralık	DP slave veya modül bir parametre veya değer için geçersiz aralık bildiriyor.
	B8	Geçersiz parametre	DP slave veya modül geçersiz parametre bildiriyor.
	B9	Geçersiz tip	DP slave veya modül geçersiz tip bildiriyor: <ul style="list-style-type: none"> <li>"RDREC (Sayfa 31515)" ile: Arabellek çok küçük (altkümeler okunamıyor)</li> <li>"WRREC (Sayfa 31515)" ile: Arabellek çok küçük (altkümeler yazılmıyor)</li> </ul>
	BA ila BF	Kullanıcıya özgü	Erişim yapılırken DP slave veya modülü üreticiye-özü bir hata bildiriyor. DP slave veya modül üreticisinden gelen belgeleri kontrol ediniz.
	C0	Sınır çakışmasını oku	<ul style="list-style-type: none"> <li>"WRREC (Sayfa 31515)" ile: CPU STOP modunda olduğunda veriler sadece yazılabilir. <b>Not:</b> Bu durum, verilerin kullanıcı programı tarafından yazılmayacağı anlamına gelir. Sadece bir PG / PC ile çevrimiçi olan bir veriyi yazabilirsiniz.</li> <li>"RDREC (Sayfa 31515)" ile: Modül veri kaydını gönderir, ancak hiçbir veri mevcut değildir veya veri sadece CPU STOP modunda olduğunda okunabilir. <b>Not:</b> Veriler sadece CPU STOP modunda olduğunda okunabiliyor ise, kullanıcı programı tarafından herhangi bir değerlendirme mümkün değildir. Bu durumda, yalnızca bir PG / PC ile çevrimiçi olan veriyi okuyabilirsiniz.</li> </ul>
	C1	Sınır çakışmasını yaz	Aynı veri kaydı için modüle, önceki yazma isteği verisi modül tarafından henüz işlenmemiştir.
	C2	Kaynak meşgul	Modül şu anda bir CPU için olası maksimum sayıda iş işliyor.
	C3	Kaynak kullanılmıyor	Gerekli işletim kaynakları şu anda işgal edilmiştir.
	C4		İç geçici hata. İş yürütülemez. İş tekrarlayınız. Bu hata sık sık oluşursa, elektriksel girişim kaynakları için kurulumu kontrol ediniz.
	C5		DP slave veya modül kullanılamaz.
	C6		Veri kaydı transferi, öncelikli sınıf iptali nedeniyle iptal edildi.
	C7		İş, DP master'daki ılık veya soğuk yeniden başlatma nedeniyle durduruldu.
	C8- CF		DP slave veya modül, üreticiye özü kaynak hatası bildiriyor. DP slave veya modül üreticisinden gelen belgeleri lütfen kontrol ediniz.
	Dx	Kullanıcıya özgü	DP Slave'e özü. DP Slave açıklamasına bakınız.
81	00 - FF		Başlangıç çağrı parametresinde hata ("RALRM (Sayfa 31818)" ile: MODE)
	00		Kuraldışı çalışma modu
82	00 - FF		İkinci çağrı parametresinde hata

## 8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-i)

Hata_kod çözme (B#16#....)	Hata_kod_1 (B#16#....)	Tarif (DVP1)	Açıklama
88	00 - FF		Sekizinci çağrı parametresinde hata ("RALRM (Sayfa 31818)": TINFO) <b>Not:</b> "TINFO" döndürülen arabelleklerin nasıl yorumlanacağı hakkında bilgilere anında erişim için STEP 7 çevrimiçi bilgi sistemine bakınız.
	01		Yanlış sözdizimi ID
	23		Miktar yapısı aşıldı ya da hedef alan çok küçük
	24		Yanlış aralık ID
	32		Kullanıcı aralığı dışında DB/DI numarası
	3A		Alan ID DB/DI için DB/DI numarası NULL veya belirtilen DB/DI mevcut değil.
89	00-FF		Dokuzuncu çağrı parametresinde hata ("RALRM (Sayfa 3186)" ile: AINFO) <b>Not:</b> "AINFO" döndürülen arabelleklerin nasıl yorumlanacağı hakkında bilgilere anında erişim için STEP 7 çevrimiçi bilgi sistemine bakınız.
	01		Yanlış sözdizimi ID
	23		Miktar yapısı aşıldı ya da hedef alan çok küçük
	24		Yanlış aralık ID
	32		Kullanıcı aralığı dışında DB/DI numarası
	3A		Alan ID DB/DI için DB/DI numarası NULL veya belirtilen DB/DI mevcut değil.
8A	00 - FF		10. çağrı parametresinde hata
8F	00 - FF		15. çağrı parametresinde hata
FE, FF	00 - FF		Profile-özü hata

**Dizi elemanı STATUS [4]**

DPV1 hataları ile, CPU ve komut için DP Master, STATUS [4]'de geçer. Bir DPV1 hatası olmadan, bu değer RDREC için aşağıdaki istisnalar dışında, 0'a ayarlanır:

- STATUS [4] RECORD'dan gelen hedef alan uzunluğunu içerir, MLEN > RECORD'dan gelen hedef alan uzunluğu ise.
- STATUS [4]=MLEN, gerçek veri kayıt uzunluğu < MLEN < RECORD'dan gelen hedef alan uzunluğu ise.
- STATUS [4]=0, STATUS [4] > 255; ayarlanması gerekecek ise.

PROFINET IO'da, STATUS [4] değeri 0 olur.



### DPRD\_DAT işlemleri

Hedef alan, STEP 7 ile yapılandırılan seçilen modülün uzunluğuyla aynı olmalıdır. Veri transferi sırasında hata meydana gelirse, RECORD ile belirlenen hedef alana okunan veri girilir.

Modüler bir tasarıma veya birkaç DP kimliğine sahip bir DP standart slave'den okursanız, sadece yapılandırılmış başlangıç adresini belirterek her DPRD\_DAT komut çağrısı için bir modül/DP kimlik verisine erişebilirsiniz.

### DPWR\_DAT işlemleri

Sürekli olarak adreslenen DP standart slave/PROFINET IO'suna RECORD'daki veriyi transfer edersiniz. Veri senkron olarak transfer edilir, yani yazma süreci komut tamamlandığında tamamlanır.

Kaynak alanı, STEP 7 ile yapılandırılan seçilen modülün uzunluğunda olmalıdır.

DP standart slave modüler bir tasarıma sahipse, yalnızca bir adet DP slave modülüne erişebilirsiniz.

Çizelge 8- 80 DPRD\_DAT ve DPWR\_DAT hata kodları

Hata kodu	Açıklama
0000	Hata oluşmadı
8090	Aşağıdaki durumlardan biri uygulanır: <ul style="list-style-type: none"> <li>• Belirtilen mantıksal taban adresi için bir modül yapılandırmadınız.</li> <li>• Tutarlı veri uzunluğuna ilişkin kısıtlamayı göz ardı ettiniz.</li> <li>• Onaltılık formatta LADDR parametresinde başlangıç adresi girmediniz.</li> </ul>
8092	RECORD parametresi aşağıdaki veri tiplerini destekler: Byte, Char, Word, Dword, Int, UInt, USInt, SInt, DInt, UDInt ve bu tiplerdeki diziler.
8093	LADDR'da belirtilen mantıksal adreste var olan tutarlı veriyi okuyabileceğiniz (DPRD_DAT) veya yazabileceğiniz (DPWR_DAT) DP modülü / PROFINET IO cihazı yok.
80A0	I/O cihazlarına erişilirken erişim hatası algılandı (DPRD_DAT).
80A1	I/O cihazlarına erişilirken erişim hatası algılandı (DPWR_DAT).
80B0	Harici DP ara yüz modülünde (DPRD_DAT) ve (DPWR_DAT) Slave arızası
80B1	Belirtilen hedef (DPRD_DAT) veya kaynak (DPWR_DAT) alanı uzunluğu, STEP 7 Basic ile yapılandırılmış kullanıcı veri uzunluğuyla aynı değildir.
80B2	Harici DP ara yüz modülünün (DPRD_DAT) ve (DPWR_DAT) Sistem hatası
80B3	Harici DP ara yüz modülünün (DPRD_DAT) ve (DPWR_DAT) Sistem hatası
80C0	Veri henüz modül tarafından okunmadı (DPRD_DAT).
80C1	Modüldeki önceki yazma işi verisi henüz modül tarafından işlenmedi (DPWR_DAT).
80C2	Harici DP ara yüz modülünün (DPRD_DAT) ve (DPWR_DAT) Sistem hatası
80Fx	Harici DP ara yüz modülünün (DPRD_DAT) ve (DPWR_DAT) Sistem hatası
85xy	Harici DP ara yüz modülünün (DPWR_DAT) Sistem hatası
87xy	Harici DP ara yüz modülünün (DPWR_DAT) Sistem hatası

## Genişletilmiş komutlar

### 8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-i)

Hata kodu	Açıklama
808x	Harici DP ara yüz modülü ile sistem hatası (DPRD_DAT)
8xyy	Genel hata bilgisi Genel hata kodları hakkında daha fazla bilgi amacıyla (Sayfa 407) "Genişletilmiş' komutlar için Ortak hata kodları" na bakınız.

x = parametre numarası

y = olay numarası


#### Not

DPV1 slave'lere erişerseniz, bu slave'lerden gelen hata bilgileri DP master'dan komuta yönlendirilebilir.

### 8.3.6 DPNRM\_DG (Bir DP slave'den gelen tanılama (diyagnostik) verilerini okuma) komutu

PROFIBUS ile DPNRM\_DG (Tanılama verileri okuma) komutunu kullanabilirsiniz.

Çizelge 8- 81 DPNRM\_DG komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := DPNRM_DG(   req:= bool_in_,   laddr:= word_in_,   record=&gt;_variant_out_,   busy=&gt; bool_out );</pre>	<p>EN 50 170 Cilt 2, PROFIBUS tarafından belirtilen formatta bir DP slave'in mevcut Tanılama verilerini okumak için DPNRM_DG komutunu kullanınız. Okunan veriler, hatasız veri transferinin ardından RECORD ile belirtilen hedef alana girildi.</p>

Çizelge 8- 82 Parametreler için veri tipleri DPNRM\_DG komutu

Parametre ve tipi	Veri tipi	Açıklama
REQ	IN	Bool
LADDR	IN	HW_DPSLAVE
RET_VAL	OUT	Int

REQ=1: Okuma isteği

DP slave'in yapılandırılmış tanılama adresi: I / O cihazına göre değil istasyon adresi olmalıdır. Tanılama adresi belirlemek için " Cihaz Yapılandırma" nın "Network" görünümünde istasyonu (ve cihazın görüntüsü değil) seçiniz.

Onaltılık biçimde adresleri giriniz. Örneğin, 1022' nin tanılama adresi LADDR: = W#16#3FE anlamına gelir.

Fonksiyon aktif iken bir hata oluşursa, dönüş değeri bir hata kodu içerir. Hiçbir hata oluşmazsa, aslında aktarılan verinin uzunluğu RET\_VAL içine aktarılır.



8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-i)

Parametre ve tipi		Veri tipi	Açıklama
RECORD	OUT	Variant	Okunan tanılama verileri için hedef alanı. Okunacak veri kaydının asgari uzunluğu (veya hedef bölge) 6 bayttır. Gönderilecek veri kaydının maksimum uzunluğu 240 bayttır. Standart slave'ler, maksimum 244 bayta kadar olmak üzere 240 bayttan fazla tanılama verisi sağlayabilir. Bu durumda, ilk 240 bayt hedef alana transfer edilir ve taşma biti veride ayarlanır.
BUSY	OUT	Bool	BUSY=1: Okuma işi henüz tamamlanmadı.

DPNRM\_DG komut çağrısında giriş parametresi REQ için 1 atayarak okuma işini başlatırsınız. Okuma işi asenkron olarak yürütülür, diğer bir deyişle bu iş birkaç DPNRM\_DG komut çağrısı gerektirir. İşin durumu RET\_VAL ve BUSY çıkış parametreleri ile gösterilir.

Çizelge 8- 83 Slave tanılama veri yapısı

Bayt	Açıklama
0	İstasyon durumu 1
1	İstasyon durumu 2
2	İstasyon durumu 3
3	Master istasyon numarası
4	Satıcı ID (yüksek bayt)
5	Satıcı ID (düşük bayt)
6 ...	Ek slave-özgü tanılama bilgileri

Çizelge 8- 84 DPNRM\_DG komutu hata kodları

Hata kodu	Açıklama	Kısıtlama
0000	Hata yok	-
7000	REQ=0 ile ilk çağrı: aktif veri transferi yok; BUSY = 0.	-
7001	REQ =1 ile ilk çağrı: aktif veri transferi yok; BUSY = 1.	Dağıtılmış I/O'lar
7002	(REQ alakalı değil) ara çağrı: veri transferi çoktan aktif; BUSY =1.	Dağıtılmış I/O'lar
8090	Belirtilen mantıksal taban adresi geçersiz: Hiçbir taban adresi yok.	-
8092	RECORD parametresi aşağıdaki veri tiplerini destekler: Byte, Char, Word, Dword, Int, UInt, USInt, Sint, DInt, UInt ve bu tiplerden oluşan dizi.	-
8093	<ul style="list-style-type: none"> <li>LADDR tarafından belirtilen modül için bu komuta izin verilmez (S7-1200 için S7-DP modüllerine izin verilir).</li> <li>LADDR, istasyonu belirtmek yerine I/O cihazını belirtir. Tanılama adresi belirlemek için " Cihaz Yapılandırma" nın "Ağ" görünümünde istasyonu (ve cihazın görüntüsü değil) seçiniz.</li> </ul>	-
80A2	<ul style="list-style-type: none"> <li>Katman 2'de DP protokol hatası (Örneğin, slave arızası veya veri yolu sorunları için)</li> <li>ET200S için, veri kaydı DPV0 modunda okunamıyor.</li> </ul>	Dağıtılmış I/O'lar
80A3	Kullanıcı ara yüzü/kullanıcı ile DP protokol hatası	Dağıtılmış I/O'lar

### Genişletilmiş komutlar

#### 8.3 Dağıtılmış I/O (PROFINET, PROFIBUS veya AS-i)

Hata kodu	Açıklama	Kısıtlama
80A4	İletişim veri yolu üzerindeki iletişim sorunu	Hata CPU ve harici DP ara yüz modülü arasında meydana gelir.
80B0	<ul style="list-style-type: none"> <li>Komut modül tipi için mümkün değildir.</li> <li>Bu modül, veri kaydını tanımıyor.</li> <li>Veri kayıt numarası 241'e izin verilmez.</li> </ul>	-
80B1	RECORD parametresinde belirtilen uzunluğu yanlış.	Belirtilen uzunluk > kayıt uzunluğu
80B2	Yapılandırılmış slot işgal edilmemiş.	-
80B3	Gerçek modül tipi gerekli modül tipiyle eşleşmiyor.	-
80C0	Tanımlama bilgisi yok.	-
80C1	Modüldeki aynı veri kaydı için önceki yazma işinin verisi henüz modül tarafından işlenmemiştir.	-
80C2	Modül şu anda bir CPU için mümkün olan maksimum sayıda iş yapıyor.	-
80C3	Gerekli kaynaklar (bellek, vb.) şu anda işgal ediliyor.	-
80C4	Dahili geçici hata. İş yapılamadı. İş tekrarlayınız. Bu hata sık sık oluşursa, elektriksel gürültü kaynakları için sisteminizi kontrol ediniz.	-
80C5	Dağıtılmış I/O'lar kullanılamaz.	Dağıtılmış I/O'lar
80C6	Veri kayıt transferi, öncelikli sınıf iptali nedeniyle (yeniden başlatma veya arka plan) durduruldu.	Dağıtılmış I/O'lar
8xyy <sup>1</sup>	Genel hata kodları	

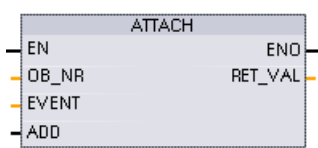
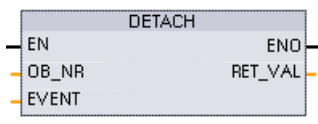
Genel hata kodları hakkında daha fazla bilgi için (Sayfa 322) " Genişletilmiş komutlar, Dağıtılmış I / O: RDREC, WRREC ve RALRM için hata bilgisi" ne bakınız.

## 8.4 Kesmeler (Interrupts)

### 8.4.1 ATTACH ve DETACH (OB ve kesme olayını baęlama/ayırma) komutları

ATTACH ve DETACH komutları ile kesme olayına odaklı altprogramları aktifleřtirebilir ve devre dıřı bırakabilirsiniz.

Çizelge 8- 85 ATTACH ve DETACH komutları

LAD / FBD	SCL	Açıklama
 <p>ATTACH instruction LAD diagram showing EN, END, OB_NR, RET_VAL, EVENT, and ADD inputs/outputs.</p>	<pre>ret_val := ATTACH(     ob_nr:=_int_in_,     event:=_event_att_in_,     add:=_bool_in_);</pre>	ATTACH bir donanım kesme olayı için kesme OB alt programı yürütülmesine imkan verir.
 <p>DETACH instruction LAD diagram showing EN, END, OB_NR, RET_VAL, and EVENT inputs/outputs.</p>	<pre>ret_val := DETACH(     ob_nr:=_int_in_,     event:=_event_att_in_);</pre>	DETACH bir donanım kesme olayı için kesme OB alt programı yürütülmesini iptal eder.

Çizelge 8- 86 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
OB_NR	IN	OB_ATT
		Organizasyon bloęu belirleyici: " Add new block " özellięi kullanılarak oluşturulan mevcut donanım kesmesi OB'lerden seçim yapınız. Parametre alanına çift tıklayınız, sonra mevcut OB'leri görmek için yardım simgesine tıklayınız.
EVENT	IN	EVENT_ATT
		Olay belirleyici: Dijital giriřleri veya yüksek-hız sayacı için PLC'de Cihaz Yapılandırmasında etkinleřtirilen mevcut donanım kesmesinden seçim yapınız. Parametre alanına çift tıklayınız, sonra mevcut olayları görmek için yardım simgesine tıklayınız.
ADD (ATTACH yalnızca)	IN	Bool
		<ul style="list-style-type: none"> <li>ADD = 0 (varsayılan): Bu olay, bu OB için bütün önceki olay eklerini deęiřtirir.</li> <li>ADD = 1: Bu olay, bu OB için önceki olay eklerine ilave edilir.</li> </ul>
RET_VAL	OUT	Int
		Yürütme durum kodu

### Donanım kesme olayları

Ařaęıdaki donanım kesme olayları CPU tarafından desteklenmektedir:

Pozitif kenar olayları: İlk 12 yerleřik CPU dijital giriřleri (DIa.0 ila DIb.3) ve tüm SB dijital giriřleri

- Dijital giriř, giriře baęlı bir saha cihazından gelen sinyaldeki bir deęiřikliğe tepki olarak OFF'dan ON'a geçiř yaptığında pozitif kenar oluřur.

- Düşen kenar olayları: ilk 12 yerleřik CPU dijital giriřleri (DIa.0 ila DIb.3) ve tüm SB dijital giriřleri
  - Dijital giriř, ON 'dan OFF'a geçiř yaptığında düşen kenar oluřur.
- Yüksek hızlı sayıcı (HSC) mevcut deęeri = referans deęer (CV = RV) olayları (HSC 1 ila 6).
  - Mevcut sayı, bitişik deęerden daha önce oluřturulmuř bir referans deęerle tam olarak eřleşen deęere geçiř yaptığında, bir HSC için CV = RV kesmesi üretilir.
- HSC yön deęiřtirme olayları (HSC 1 ila 6)
  - HSC'nin, artıřtan azalmaya veya azalmadan artıřa geçtięi tespit edildiğinde, bir yön deęiřtirme olayı meydana gelir.
- HSC harici reset olayları (HSC 1 ila 6)
  - Belirli HSC modları, HSC sayma deęerini sıfırlamak için kullanılan harici bir reset olarak bir dijital giriřin atanmasına izin verir. Böyle bir SC için harici reset olayı, bu giriř OFF'dan ON'a geçiř yaptığında meydana gelir.

### **Donanım kesme olaylarını cihaz yapılandırmasında etkinleřtirme**

Donanım kesmeleri aygıt yapılandırması sırasında etkin olmalıdır. Bu olayı yapılandırma sırasında veya çalıřma zamanında eklemek isterseniz, bir dijital giriř kanal veya bir HSC için cihaz yapılandırmasındaki olay-etkinleřtirme kutusunu kontrol etmelisiniz.

PLC aygıt yapılandırması içindeki kutu seęeneklerini kontrol ediniz:

- Dijital giriř
  - Pozitif kenar algılamayı etkinleřtir
  - Düşen kenar algılamayı etkinleřtir
- Yüksek hızlı sayıcı (HSC)
  - Kullanım için bu yüksek hızlı sayıcıyı etkinleřtiriniz
  - Referans deęer sayısına eřit sayıcı deęeri için kesme üret
  - Harici reset olayı için kesme üret
  - Yön deęiřtirme olayı için kesme üret

## Programınıza yeni donanım kesme OB kod blokları ekleme

Olay ilk etkinleştirildiğinde, varsayılan olarak, hiçbir OB bir olaya bağlanmaz. Bu "HW kesme:" Cihaz Yapılandırması "<not connected>" etiketi ile gösterilir. Sadece donanım kesmesi OB'ler bir donanım kesme olayına bağlanabilir. Mevcut tüm donanım kesme OB'leri "HW kesme:" açılır listede görünür. Hiçbir OB listede yoksa, o zaman aşağıdaki gibi "Donanım kesme" tipinde bir OB oluşturmanız gerekir. Proje ağacı "Program blokları" dalı altında:

4. "yeni blok ekle" çift tıklayınız, Organizasyon bloğu (OB)" seçiniz ve "Donanım kesme" seçiniz.
5. İsteğe bağlı olarak, OB'yi yeniden adlandırabilirsiniz, programlama dilini (LAD, FBD veya SCL) ve blok numarasını (manuel'e seçiniz ve önerilenden farklı bir blok numarası seçiniz) seçiniz.
6. OB'yi düzenleyin ve olay meydana geldiğinde yürütmek istediğiniz programlanmış tepkiyi ekleyiniz. Altılı bir yuvalama derinliğine kadar bu OB'den FC'ler ve FB'ler çağırabilirsiniz.

## OB\_NR parametresi

Mevcut tüm donanım-kesme OB isimleri Cihaz Yapılandırmasında "HW kesme" açılır liste ve ATTACH / DETACH parametresi OB\_NR açılır listesinde görünür.

## EVENT parametresi

Bir donanım kesmesi olayı etkin olduğunda, benzersiz bir varsayılan olay adı, bu özel olay için atanır. "Olay adı:" düzenleme kutusunu düzenleyerek bu olay adını değiştirebilirsiniz, ancak benzersiz bir ad olmalıdır. Bu olay isimleri "sabitler" etiket çizelgesinde etiket isimleri haline gelir ve ATTACH ve DETACH komut kutuları için EVENT parametresi açılır listesinde görünür. Etiket değeri, olayı tanımlamak için kullanılan bir iç sayıdır.

## Genel işlem

Her donanım olayı, donanım kesme olayı meydana geldiğinde yürütme için sıraya konacak bir donanım kesme OB'sine bağlanabilir. OB-olay bağlantısı, yapılandırma sırasında veya çalışma zamanında ortaya çıkabilir.

Yapılandırma sırasında etkin bir olaya bağlama veya ayırma seçeneğiniz vardır. Yapılandırma sırasında bir OB'yi bir olaya bağlamak için "HW kesme:" açılır liste (sağda aşağı oka tıklayınız) kullanmanız ve mevcut donanım kesme OB'leri listesinden bir OB seçimi yapmanız gerekir. "Bu listeden uygun OB adını seçiniz veya bağlantıyı kaldırmak için "<not connected>" seçiniz.

Ayrıca çalışma zamanında etkin bir donanım kesme olayını bağlayabilir veya ayırabilirsiniz. Uygun OB için etkin bir kesme olayı bağlamak veya ayırmak için (isterseniz birden çok kez), çalıştırma zamanında ATTACH veya DETACH program kodlarını kullanınız. Hiçbir OB bağlı değilse (Cihaz Yapılandırmasında bir "<not connected>" seçiminden ya da bir DETACH komutunun bir sonucu olarak), etkin donanım kesme olayı göz ardı edilir.

## DETACH iřlemi

Belirli bir olay veya belirli bir OB'den kaynaklı tm olayları ayırmak iin DETACH komutu kullanınız. Bir EVENT belirtilirse, o zaman sadece bu olay, belirtilen OB\_NR'den ayrılır; geerli bu OB\_NR'ye baėlı bařka tm olaylar baėlı kalmaya devam edecektir. Hibir EVENT belirtilmemiře, o zaman geerli OB\_NR'ye baėlı tm olaylar ayrılacaktır.

## Durum kodları

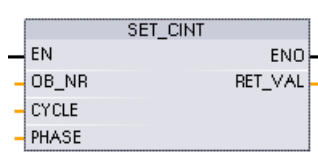
izelge 8- 87 Durum kodları

RET_VAL (W#16#....)	ENO	Aıklama
0000	1	Hata yok
0001	1	Ayrılacak bir řey yok (DETACH yalnızca)
8090	0	OB mevcut deėil
8091	0	OB yanlıř tiptedir
8093	0	Olay bulunmuyor

## 8.4.2 Dngsel kesmeler

### 8.4.2.1 SET\_CINT (Dngsel kesme parametrelerini ayarlama) komutu

izelge 8- 88 SET\_CINT (Dngsel kesme parametrelerini ayarlama)

LAD / FBD	SCL	Aıklama
	<pre>ret_val := SET_CINT(   ob_nr:=_int_in_,   cycle:=_udint_in_,   phase:=_udint_in_);</pre>	<p>Program taramayı kesen dngsel yrtmeyi bařlatmak iin belirtilen kesme OB'yi ayarlayınız.</p>

izelge 8- 89 Parametreler iin veri tipleri

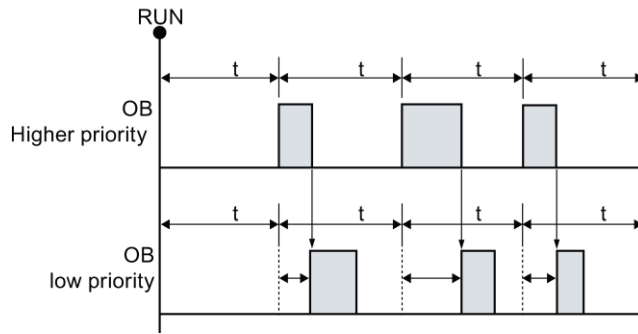
Parametre ve tipi	Veri tipi	Aıklama
OB_NR	IN	OB_CYCLIC
CYCLE	IN	UDInt
PHASE	IN	UDInt
RET_VAL	OUT	Int

Zaman parametre rnekleri:

- CYCLE zamanı = 100 us ise, OB\_NR tarafından bařvurulan kesme OB, dngsel program taramasını her 100 us’de keser. Kesme OB yrtlr ve sonra kesme noktasında yrtme kontroln program taramaya dndrr.
- CYCLE zamanı = 0 ise, kesme olayı devre dıřı bırakılır ve kesme OB yrtlmez.
- PHASE (faz kayması) zamanı, CYCLE zaman aralıęı bařlamadan nce meydana gelen bir belirtilen gecikme sresidir. Daha dřk ncelikli OB’lerin yrtme zamanlamasını kontrol etmek iin faz kaymasını kullanabilirsiniz.

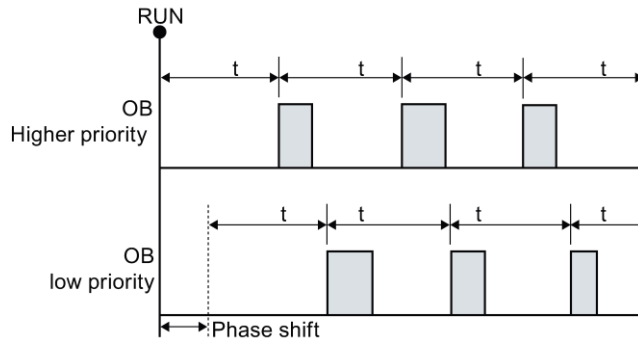
Dřk ve yksek ncelikli OB’ler aynı zaman aralıęında aęrılırsa, dřk ncelikli OB yalnızca, yksek ncelikli OB iřlemine tamamlandıktan sonra aęrılır. Dřk ncelikli OB iin yrtme bařlangı zamanı, yksek ncelikli OB’lerin iřlem sresine baęlı olarak deęiřebilir.

OB call without phase shift



Bir sabit zaman evriminde dřk ncelikli OB yrtlmesini bařlatmak istiyorsanız, o zaman faz kayma sresi yksek ncelikli OB’lerin iřlem sresinden daha byk olmalıdır.

OB-call with phase shift



izelge 8- 90 Durum kodları

RET_VAL (W#16#....)	Aıklama
0000	Hata yok
8090	OB bulunmuyor veya yanlıř tiptedir
8091	Geersiz evrim sresi

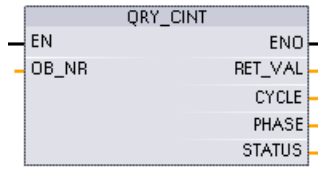
## Geniştirilmiş komutlar

### 8.4 Kesmeler (Interrupts)

RET_VAL (W#16#....)	Açıklama
8092	Geçersiz faz kayma süresi
80B2	OB'nin bağlantılı bir olayı yok.

#### 8.4.2.2 QRY\_CINT (Döngüsel kesme parametrelerini sorgulama) komutu

Çizelge 8- 91 QRY\_CINT (Döngüsel kesmeyi sorgula)

LAD / FBD	SCL	Açıklama
 <p>The diagram shows the QRY_CINT instruction with inputs EN and OB_NR, and outputs END, RET_VAL, CYCLE, PHASE, and STATUS.</p>	<pre>ret_val := QRY_CINT(   ob_nr:=_int_in_,   cycle=&gt;_udint_out_,   phase=&gt;_udint_out_,   status=&gt;_word_out_);</pre>	<p>Döngüsel bir kesme OB'den parametre ve yürütme durumunu alınız. Döndürülen değerler QRY_CINT'ın yürütüldüğü zamanda mevcuttur.</p>

Çizelge 8- 92 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
OB_NR	IN	OB_CYCLIC
RET_VAL	OUT	Int
CYCLE	OUT	UDInt
PHASE	OUT	UDInt
STATUS	OUT	Word
		<p>Döngüsel kesme durum kodu:</p> <ul style="list-style-type: none"> <li>Bitler 0 ila 4, aşağıdaki STATUS Çizelgesine bakınız</li> <li>Diğer bitler, daima 0</li> </ul>

Çizelge 8- 93 STATUS parametresi

Bit	Value	Açıklama
0	0	CPU RUN esnasında
	1	Başlatma esnasında
1	0	Kesme etkinleştirilir.
	1	Kesme DIS_IRT komutuyla devre dışı bırakılır.
2	0	Kesme aktif değil veya durdu.
	1	Kesme aktif.
4	0	OB_NR tarafından tanımlanan OB yok.
	1	OB_NR tarafından tanımlanan OB var.
Diğer bitler		Daima 0

Bir hata oluşursa, RET\_VAL uygun hata kodunu gösterir ve parametre STATUS = 0 olur.



Çizelge 8- 94 RET\_VAL parametresi

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
8090	OB bulunmuyor veya yanlış tiptedir
80B2	OB'ye baėlı bir olay yok

### 8.4.3 Günün saati kesmeleri

**⚠ WARNING**

**Bir saldırgan Ağ Zaman Protokolü (NTP) senkronizasyonu yoluyla aėlarınıza erişebilirse, saldırgan muhtemelen CPU sistem saatini kaydırarak yaptığınız işlemi sınırlı olarak kontrolü altına alabilir.**

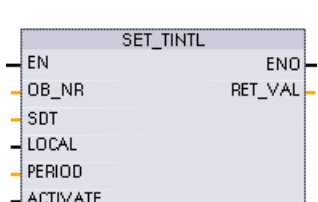
S7-1200 CPU'nun NTP istemci özelliėi varsayılan olarak devre dıřıdır ve etkinleřtirildiėinde, sadece yapılandırılmıř IP adreslerinin NTP sunucusu olarak davranmasına izin verir. CPU varsayılan olarak bu özelliėi devre dıřı bırakır ve uzaktan kontrollü CPU sistem saati düzeltmelerine izin vermek için bu özelliėi yapılandırmanız gerekir.

S7-1200 CPU doėru CPU sistem saatine baėlı olan "günün saati" kesmeleri ve saat komutlarını destekler. Eėer NTP'yi yapılandırır ve bir sunucudan zaman senkronizasyonu kabul ederseniz, sunucunun güvenilir bir kaynak olduėundan emin olmalısınız. Aksi takdirde bilinmeyen bir kullanıcının CPU sistem saatini kaydırarak yaptığınız işlemi sınırlı olarak kontrolü altına alabilmesine izin veren bir güvenlik ihlaline sebep olabilir.

Güvenlik bilgisi ve öneriler için, Siemens Servis ve destek sitesindeki ([http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational\\_guidelines\\_industrial\\_security\\_en.pdf](http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf)) "Sanayi Güvenliėi ile ilgili operasyonel Kurallar" a bakılmalıdır.

#### 8.4.3.1 SET\_TINTL (günün saati kesmesi ayarlama)

Çizelge 8- 95 SET\_TINTL (DTL veri tipiyle tarih ve günün saati kesmesi ayarlama)

LAD / FBD	SCL	Açıklama
	<pre>ret_val := SET_TINTL(     OB_NR:=_int_in_,     SDT:=_dtl_in_,     LOCAL:=_bool_in_     PERIOD:=_word_in_     ACTIVATE:=_bool_in_);</pre>	<p>Bir tarih ve günün saati kesmesi ayarlayınız. Program kesintisi OB, bir yürütme veya atanmıř bir zaman periyodu ile yinelenen yürütme için ayarlanabilir.</p>

## Geniřletilmiř komutlar

### 8.4 Kesmeler (Interrupts)

Çizelge 8- 96 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)	OB numarası (sembolik ismi kabul eder)
SDT	IN	DTL	Tarih ve saati başlat: Saniye ve milisaniye göz ardı edilir ve 0'a ayarlanabilir.
LOCAL	IN	Bool	0 = Sistem saatini kullanınız 1 = Yerel saati kullanınız (CPU yerel saat için yapılandırıldığı takdirde sistem saatini kullanabilirsiniz)
PERIOD	IN	Word	Başlangıç tarihi ve saatinden itibaren yinelenen kesme olayı için periyot. <ul style="list-style-type: none"> <li>• W#16#0000 = Bir kere</li> <li>• W#16#0201 = Her dakika</li> <li>• W#16#0401 = Her saat</li> <li>• W#16#1001 = Günlük</li> <li>• W#16#1201 = Haftalık</li> <li>• W#16#1401 = Aylık</li> <li>• W#16#1801 = Yıllık</li> <li>• W#16#2001 = Ay sonu</li> </ul>
ACTIVATE	IN	Bool	0 = ACT_TINT kesme olayını etkinleřtirmek için yürütülmesi gerekir. 1 = Kesme olayı etkinleřtirilir.
RET_VAL	OUT	Int	Yürütme durum kodu

Programınız, atanan kesme OB'yi yürütecek tarih ve günün saati kesmesi olayını ayarlamak için SET\_TINTL kullanabilirsiniz. Başlangıç tarihi ve saati (örneğin günlük veya haftalık) SDT parametresiyle ve tekrarlayan kesmeler için zaman periyodu PERIOD parametresiyle belirlenir. Aylık olarak tekrarlama periyodunu ayarlarsanız, 1 ila 28 arası bir güne başlangıç tarihini ayarlamanız gerekir. Bunlar Şubat ayında meydana gelmediği için 29 ila 31 arası günler kullanılamayabilir. Her ayın sonunda bir kesme olayı istiyorsanız, PERIOD parametresi için ayın sonunu kullanınız.

SDT parametresi içinde DTL verisi hafta günü değeri göz ardı edilir. Bir Çevrimiçi CPU'nun "Online & diagnostics" görünümünde "Set günün saati" fonksiyonunu kullanarak bir CPU'nun geçerli tarih ve saatini ayarlayınız. Ay, ayın günü ve yılı ayarlamak gerekir. STEP 7, CPU'nun tarih ve zaman saatine göre kesme periyodunu hesaplar.

#### Not

Yazdan kışa geçildiğinde (yaz saati uygulaması) günün ilk saati yoktur. İkinci saat içinde bir başlangıç zamanı kullanınız veya ilk bir saat içinde bir ek zaman gecikme kesmesi kullanınız.

Çizelge 8- 97 Durum kodu

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz OB_NR parametresi
8091	Geçersiz SDT başlangıç zamanı parametresi: (Örneğin, yaz saati başlangıcında atlanan saat içinde bir başlangıç zamanı)
8092	Geçersiz PERIOD parametresi
80A1	Başlangıç saati geçmiştir. (Bu hata kodu yalnızca PERIOD = W #16#0000 ile meydana gelir.)

### 8.4.3.2 CAN\_TINT (günün saati kesmesini iptal etme)

Çizelge 8- 98 CAN\_TINT (tarih ve günün saati kesmesini iptal etme)

LAD / FBD	SCL	Açıklama
	<pre>ret_val:=CAN_TINT(_int_in);</pre>	Belirtilen OB kesmesi için başlangıç tarih ve günün saati kesme olayını iptal et.

Çizelge 8- 99 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)
RET_VAL	OUT	Int

Çizelge 8- 100 Durum kodları

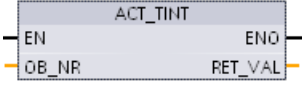
RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz OB_NR parametresi
80A0	Belirtilen OB kesmesi için başlangıç tarih / saat ayarı yok

## Geniřletilmiř komutlar

### 8.4 Kesmeler (Interrupts)

#### 8.4.3.3 ACT\_TINT (gnn saati kesmesini etkinleřtirme)

Çizelge 8- 101 ACT\_TINT (tarih ve gnn saati kesmesini etkinleřtirme)

LAD / FBD	SCL	Açıklama
	<pre>ret_val:=ACT_TINT(_int_in_);</pre>	Belirtilen OB kesmesi için bařlangıç tarih ve gnn saati kesme olayını etkinleřtir.

Çizelge 8- 102 Parametreler için veri tipleri

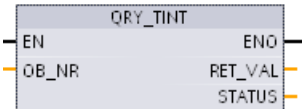
Parametre ve tipi	Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)
RET_VAL	OUT	Int

Çizelge 8- 103 Durum kodları

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz OB_NR parametresi
80A0	İlgili gnn saati OB kesmesi için bařlangıç tarih / saat ayarı yok
80A1	Etkinleřtirilen saat geçmiřtedir. Hata yalnızca, kesme OB sadece bir kez çalıřtırmak için ayarlandığında olur.

#### 8.4.3.4 QRY\_TINT (gnn saati kesme durumunu sorgulama)

Çizelge 8- 104 QRY\_TINT (tarih ve gnn saatini ayarlakesmesini sorgulama)

LAD / FBD	SCL	Açıklama
	<pre>ret_val:=QRY_TINT(   OB_NR:=_int_in_,   STATUS=&gt;_word_out_);</pre>	Belirtilen OB kesmesi için tarih ve gnn saatini ayarlakesme durumunu sorgular.

Çizelge 8- 105 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
OB_NR	IN	OB_TOD (INT)
RET_VAL	OUT	Int
STATUS	OUT	Word

Çizelge 8- 106 STATUS parametresi

Bit	Değer	Açıklama
0	0	Çalışmada
	1	Başlangıçta
1	0	Kesme etkinleştirilir.
	1	Kesme devre dışı bırakılır.
2	0	Kesme aktif değil veya süresi geçti.
	1	Kesme aktif.
4	0	Atanan OB_NR yok.
	1	Atanan OB_NR ile OB yok.
6	1	tarih ve günün saati kesmesi yerel saati kullanır.
	0	tarih ve günün saati kesmesi sistem saatini kullanır.
Diğerleri		Daima 0

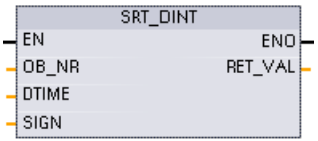
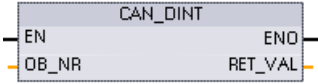

Çizelge 8- 107 Durum kodu

RET_VAL (W#16#....)	Açıklama
0000	Hata yok
8090	Geçersiz OB_NR parametresi

### 8.4.4 Time delay (Zaman gecikme) kesmeleri

Bařlangıç ve SRT\_DINT ve CAN\_DINT komutları ile zaman gecikmesi kesme iřlemine bařlatabilir ve iptal edebilirsiniz veya QRY\_DINT komutu ile kesme durumunu sorgulayabilirsiniz. Her zaman gecikme kesmesi, belirtilen gecikme suresinden sonra meydana gelen bir defalık bir olaydır. Gecikme zamanı dolmadan zaman gecikmesi olayı iptal edilirse, program kesintisi meydana gelmez.

Çizelge 8- 108 SRT\_DINT, CAN\_DINT ve QRY\_DINT komutları

LAD / FBD	SCL	Açıklama
	<pre>ret_val := SRT_DINT(   ob_nr:=_int_in_,   dtime:=_time_in_,   sign:=_word_in_);</pre>	SRT_DINT, DTime parametresi tarafından belirlenen gecikme süresi dolduğunda bir OB'yi yürüten bir zaman gecikme kesmesi bařlatır.
	<pre>ret_val := CAN_DINT(   ob_nr:=_int_in_);</pre>	CAN_DINT, çoktan bařlamıř olan bir zaman gecikme kesmesini iptal eder. Zaman gecikme kesmesi OB bu durumda yürütülmez.
	<pre>ret_val := QRY_DINT(   ob_nr:=_int_in_,   status=&gt;_word_out_);</pre>	QRY_DINT, OB_NR parametresi tarafından belirlenen zaman gecikme kesmesinin durumunu sorgular.

Çizelge 8- 109 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
OB_NR	IN	OB_DELAY	Bir zaman gecikmesinden sonra bařlatılacak Organizasyon bloęu (OB): "Yeni blok ekle" projesi ağaç özellięi kullanılarak oluřturulan kullanılabilir zaman-gecikmesi OB'lerden seęiniz. Parametre alanına çift tıklayınız, ardından mevcut OB'leri görmek için yardımcı ikonuna tıklayınız.
DTIME <sup>1</sup>	IN	Time	Zaman gecikme deęeri (1- 60000 ms)
SIGN <sup>1</sup>	IN	Word	S7-1200 tarafından kullanılmaz: Herhangi bir deęer kabul edilir. Hataları engellemek için bir deęer atanmalıdır.
RET_VAL	OUT	Int	Yürütme durum kodu
STATUS	OUT	Word	QRY_DINT komutu: Belirtilen bir zaman gecikme kesmesi OB durumu, ařaęıdaki çizelgeye bakınız.

<sup>1</sup> Yalnızca SRT\_DINT için

## Operasyon

SRT\_DINT komutu, bir zaman gecikmesi belirtir, dahili zaman gecikme zamanlayıcısını başlatır ve bir zaman gecikme kesmesi OB alt programı ile zaman gecikmesi zaman aşımı olayını ilişkilendirir. Belirtilen zaman gecikmesi dolduğunda, ilişkili zaman gecikme kesmesi OB'nin çalıştırılmasını tetikleyen bir program kesmesi oluşturulur. CAN\_DINT komutu yürütülerek belirtilen zaman gecikmesi oluşturulmadan önce bir işlemdeki zaman gecikme kesmesini iptal edebilirsiniz. Etkin zaman gecikmesi kesme olaylarının toplam sayısı dördü geçmemelidir.

## Projenize zaman gecikme kesmesi OB alt programlarının eklenmesi

SRT\_DINT ve CAN\_DINT komutlarına sadece zaman gecikme kesme OB'leri atanabilir. Yeni bir projede zaman gecikme kesme OB'si bulunmaz. Projenize zaman gecikme kesme OB'leri eklemeniz gerekir. Zaman gecikme kesme OB'si oluşturmak için şu basamakları izleyiniz:

7. Proje ağacının "Program bloklar" dalında "yeni blok ekle" ögesini çift tıklayınız, "Organizasyon bloğu ( OB )" seçeneğini seçiniz ve "Zaman gecikme kesmesi" ni seçiniz.
8. OB'yi adlandırmak, programlama dilini seçmek veya blok numarasını seçmek için seçeneğiniz vardır. Otomatik olarak atanmış sayıdan farklı bir blok numarası atamak isterseniz manuel numaralandırmaya geçiniz.
9. OB zaman kesme gecikmesi alt programını düzenleyiniz ve zaman gecikmesi zaman aşımı olayı gerçekleştiğinde çalıştırmak istediğiniz programlanmış tepki oluşturunuz. Maksimum altı adet yuvalama derinliği ile OB zaman gecikme kesmesinden diğer FC ve FB kod bloklarını çağırabilirsiniz.
10. SRT\_DINT ve CAN\_DINT komutlarının OB\_NR parametresini düzenlediğinizde yeni atanan OB zaman gecikme kesme isimleri kullanılabilir olacaktır.

## QRY\_DINT için STATUS parametresi

Çizelge 8- 110 Hata varsa (REL\_VAL <> 0), STATUS = 0.

Bit	Değer	Açıklama
0	0	Çalışıyor
	1	Başlangıçta
1	0	Kesme etkinleştirilir
	1	Kesme devre dışı bırakılır.
2	0	Kesme aktif değil veya süresi geçti.
	1	Kesme aktif.
4	0	OB_NR'de verilen OB numaralı OB mevcut değil.
	1	OB_NR'de verilen OB numaralı OB mevcut.
Diğer bitler		Daima 0

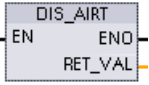
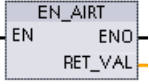
## Durum kodları

Çizelge 8- 111 SRT\_DINT, CAN\_DINT ve QRY\_DINT için durum kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata meydana gelmedi
8090	Yanlış OB_NR parametresi
8091	Yanlış DTIME parametresi
80A0	Zaman gecikme kesmesi başlatılmadı.

## 8.4.5 DIS\_AIRT ve EN\_AIRT (Yüksek öncelikli kesmelerin yürütülmesini geciktirme/ etkinleştirme ve asenkron hata olayları) komutları

Alarm kesme işlemlerini devre dışı bırakmak ve etkinleştirmek için DIS\_AIRT ve EN\_AIRT komutlarını kullanınız.

LAD / FBD	SCL	Açıklama
	DIS_AIRT ( ) ;	DIS_AIRT yeni kesme olaylarının işlenmesini geciktirir. Bir OB'de birden çok DIS_AIRT yürütebilirsiniz.
	EN_AIRT ( ) ;	EN_AIRT, önceden DIS_AIRT komutu ile devre dışı bıraktığınız kesme olaylarının işlenmesini etkinleştirir. Her DIS_AIRT yürütmesi, bir EN_AIRT yürütmesi tarafından iptal edilmelidir. EN_AIRT yürütmeleri, aynı OB içinde oluşması gerekir ya da kesmeler bu OB için tekrar etkinleştirilmeden önce, herhangi FC veya FB aynı OB'den çağrılır.

Çizelge 8- 112 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
RET_VAL	OUT	Int
		Gecikme sayısı = sıradaki DIS_AIRT yürütme sayısı.

DIS\_AIRT yürütmeleri işletim sistemi tarafından sayılır. Bunların her biri, bir EN\_AIRT komutu ile tekrar özellikle iptal edilinceye kadar ya da mevcut OB kadar tamamen işlenene kadar etkin kalır. Örneğin: Kesmeleri, beş adet DIS\_AIRT yürütmesi ile beş kez devre dışı bıraktığınızda, kesmeler tekrar etkin hale gelmeden önce bunları beş adet EN\_AIRT yürütmesi ile iptal etmeniz gerekir.

Kesme olayları tekrar etkinleştirildikten sonra, DIS\_AIRT yürürlükte iken meydana gelen kesmeler işlenir ya da kesmeler mevcut OB yürütülür yürütülmez işlenir.

RET\_VAL parametresi kesme işleminin iptal edilme sayısını gösterir. Bu da sıralanmış DIS\_AIRT yürütmelerin sayısına eşittir. Kesme işlemi sadece parametre RET\_VAL = 0 olduğunda etkinleştirilir.



## 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

### 8.5.1 Tanılama (diyagnostik) komutları

Ařağıdaki tanılama komutları PROFINET veya PROFIBUS ile birlikte kullanılabilir:

- GET\_DIAG komutu (Sayfa 358): Belirtilen cihazdan tanılama bilgilerini okuyabilirsiniz.
- DeviceStates komutu (Sayfa 347): Bir I/O alt sistemi içinde dağıtılmış I/O cihazı için çalışma durumlarını alabilirsiniz.
- ModuleStates komutu (Sayfa 353): Bir dağıtılmış I/O cihazındaki modüller için çalışma durumlarını alabilirsiniz.
- LED komutu (Sayfa 34646): Bir dağıtılmış I/O cihazı için LED'lerin durumunu okuyabilirsiniz.

### 8.5.2 Dağıtılmış I/O için tanılama (diyagnostik) olayları

#### Not

Bir PROFIBUS IO sistemi ile, bir indirmeden ya da güç çevriminden sonra, donanım uyumluluęu, kabul edilebilir yedek modüllere izin verecek şekilde ayarlanmadıkça (Sayfa 142) ve bir ya da daha fazla modül eksik veya yapılandırılmış modül için kabul edilebilir bir yedek deęilse CPU, RUN moduna geçecektir.

Ařağıdaki çizelgede gösterildięi gibi, CPU, dağıtılmış I / O sistemi bileřenleri için yapılandırılabilir tanılamayı destekler. Bu hataların her biri tanılama ara belleęinde bir kütük kaydı oluşturur.

Çizelge 8- 113 PROFINET ve PROFIBUS için tanılama olaylarının ele alınması

Hata tipi	İstasyon için tanılama bilgisi var mı?	Tanılama arabelleęine giriş?	CPU çalışma modu
Tanılama hatası	Evet	Evet	RUN modunda kalır
řasi veya istasyon arızası	Evet	Evet	RUN modunda kalır
I/O erişim hatası <sup>1</sup>	Hayır	Evet	RUN modunda kalır
Çevresel erişim hatası <sup>2</sup>	Hayır	Evet	RUN modunda kalır
Çek/tak olayı	Evet	Evet	RUN modunda kalır

<sup>1</sup> I/O erişim hatasına örnek neden: Bir modül çıkarılmıştır.

<sup>2</sup> Çevresel erişim hatası örnek nedeni: İletişim kurmayan bir alt modülle çevrimsiz iletişim.

Tanı bilgilerini elde etmek üzere her istasyon için GET\_DIAG komutunu (Sayfa 358) kullanınız. Bu, cihazda karşılaşılan hataları programlı olarak işlemenize izin verecektir ve istenirse CPU'yu STOP moduna alacaktır. Bu yöntem, durum bilgilerinin okunacaęı donanım cihazını belirtmenizi gerektirir.

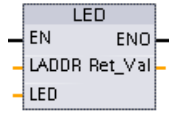
## Geniřletilmiř komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

GET\_DIAG komutu bütn istasyonun sađlamlık durumunu elde etmek için istasyonun "L adresi" (LADDR) ni kullanır. Bu L Adres, Ađ Yapılandırma görünümü içinde bulunabilir ve tüm istasyon rafını (bütn gri alan) seçerek, L Adresi, istasyonun özellikler sekmesinde gösterilmiştir. CPU için modl özelliklerinde veya varsayılan etiket çizelgesinde her modl için LADDR (Cihaz Yapılandırmasında) bulabilirsiniz.

### 8.5.3 LED (LED durumu okuma) komutu

Çizelge 8- 114 LED komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := LED(     laddr:=_word_in_,     LED:=_uint_in_);</pre>	<p>Bir CPU veya ara yüz üzerindeki LED'lerin durumunu okumak için LED komutu kullanınız. Belirtilen LED durumu RET_VAL çıkışıyla döndürlr.</p>

Çizelge 8- 115 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
LADDR	IN	HW_IO
		CPU veya ara yüzn tanıtım numarası <sup>1</sup>
LED	IN	UInt
		LED kimlik numara
		1 RUN/STOP
		Renk 1 = yeřil, renk 2 = sarı
		2 Hata
		Renk 1 = kırmızı
		3 Bakım
		Renk 1 = sarı
		4 Yedek
		uygulanmaz
		5 Bađlantı
		Renk 1 = yeřil
		6 Tx/Rx
		Renk 1 = sarı
RET_VAL	OUT	Int
		LED in durumu

<sup>1</sup> rneđin, açılır parametre listesinden CPU ("PLC\_1" gibi) veya PROFINET ara yüzn seçebilirsiniz.


Çizelge 8- 116 RET\_VAL Durumu

RET_VAL (W#16#...)	Açıklama	
0 - 9 LED durumu	0	LED bulunmuyor
	1	Kapalı
	2	Renk 1 açık (dolu)
	3	Renk 2 açık (dolu)
	4	Renk 1 2 Hz'de yanıp sönüyor
	5	Renk 2 2 Hz'de yanıp sönüyor
	6	Renk 1 & 2 2 Hz'de deęişkenli olarak yanıp sönüyor
	7	Renk 1 açık (Tx/Rx)
	8	Renk 2 açık (Tx/Rx)
	9	LED'in durumu mevcut deęil
8091	LADDR ile tanımlanan cihaz bulunmuyor	
8092	LADDR ile tanımlanan cihaz LED'leri desteklemiyor	
8093	LED kimlięi tanımlı deęil	
80Bx	LADDR ile tanımlanan CPU LED komutunu desteklemiyor	

#### 8.5.4 DeviceStates komutu

Daęıtılmış I/O Master baęlantılı daęıtılmış I/O slave cihazlarının durumlarını döndürmek için DeviceStates komutunu kullanabilirsiniz.

Çizelge 8- 117 DeviceStates komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := DeviceStates(     laddr:=hw_io_in_,     mode:=uint_in_,     state:=_variant_inout_);</pre>	<p>DeviceStates, bir I/O alt sisteminin I/O cihazı çalışma durumlarını alır. Yürütme sonrasında, STATE parametresi (atanmış LADDR ve MODE için) bit listesindeki her bir I/O cihazının hata durumunu içerir. Bu bilgiler, STEP 7 tanılama görünümünde görülen cihaz durumuna karşılık gelir.</p> <p>DeviceStates'in LADDR giriři, daęıtılmış I/O ara yüzü donanım kimlięini kullanır. TIA portalda, bir PLC için donanım kimlikleri PLC etiket çizelgesinde sistem sabitleri sekmesindeki "Hw_IoSystem" veri tipleri aranarak bulunabilir.</p>

Genişletilmiş komutlar

8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Çizelge 8- 118 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
LADDR	IN	HW_IOSYSTEM	Mantıksal adres: (I/O sistem için belirleyici)
MODE	IN	UInt	Beş adet operasyon modunu destekler. MODE girişi, hangi verilerin STATE bilgisi için belirtilen yere döndürüleceğini belirler. Modlar aşağıda verilmiştir: <ul style="list-style-type: none"> <li>1: Cihaz yapılandırması aktif</li> <li>2: Cihaz kusurlu</li> <li>3: Cihaz devre dışı</li> <li>4: Cihaz var</li> <li>5: Cihazda sorun var</li> </ul>
RET_VAL	OUT	Int	Yürütme durum kodu
STATE <sup>1</sup>	InOut	Variant	Her cihazın hata durumunu alan arabellek: STATE parametresi için seçtiğiniz veri tipi herhangi bir bit tipi (Bool, Byte, Word veya DWord) ya da bit tiplerinden oluşan bir dizi olabilir: <ul style="list-style-type: none"> <li>Döndürülen STATE verilerinin ilk baytının 0 biti özet bittir. TRUE olarak ayarlandığında, diğer verilerin mevcut olduğunu gösterir.</li> <li>STATE parametresi tarafından döndürülen veriler, bit konumu ile dağıtılmış I/O adresi arasında bire bir ilişkiyi göstermektedir. Bu cihaz adreslemesi, PROFIBUS ve PROFINET için TRUE olur. Örneğin, ilk baytın 4. biti PROFIBUS adresi 4 veya PROFINET cihaz numarası 4 ile ilişkilidir.</li> </ul>

<sup>1</sup> PROFIBUS-DP için durum bilgilerinin uzunluğu 128 bittir. PROFINET I/O için uzunluk 1024 bittir.

Yürütme sonrasında, STATE parametresi bit listesi olarak her I/O cihazının hata durumunu içerir (atanmış LADDR ve MODE için).

Çizelge 8- 119 Durum kodları

RET_VAL (W#16#...)	Açıklama
0	Hata yok
8091	LADDR yok.
8092	LADDR bir I/O sistemi adreslemiyor.
8093	STATE parametresi için atanan geçersiz veri tipi: Geçerli veri tipleri (Bool, Byte, Word, veya Dword), veya (Bools, Bytes, Words, veya Dwords) dizileri.
80Bx	DeviceStates komutu bu LADDR için CPU tarafından desteklenmiyor.
8452	Tüm durum verisi, atanan STATE parametresi için çok büyük. STATE arabelleği kısmi bir sonuç içeriyor.

### 8.5.4.1 DeviceStates rnek yapılandırması

#### PROFIBUS rneęi

PROFIBUS rneęi ařaęıdakilerden oluřur:

- 16 PROFIBUS cihazları "DPSlave\_10" ila "DPSlave\_25" olarak isimlendirilir.
- 16 PROFIBUS cihazı sırasıyla 10 ila 25 PROFIBUS adreslerini kullanır.
- Her slave cihazı oklu I/O modlleriyle yapılandırılır.
- Dndrlen STATE parametre bilgisinin ilk drt baytı grntlenir.

MOD	rnek 1: Hatasız normal alıřma	rnek 2: Tek modl  ekilli PROFIBUS slave cihazı DPSlave_12	rnek 3: Baęlantısız PROFIBUS slave cihazı DPSlave_12
1: Cihaz yapılandırması aktif	0x01FC_FF03	0x01FC_FF03	0x01FC_FF03
2: Cihaz kusurlu	0x0000_0000	0x0110_0000	0x0110_0000
3: Cihaz devre dıřı	0x0000_0000	0x0000_0000	0x0000_0000
4: Cihaz var	0x01FC_FF03	0x01FC_FF03	0x01EC_FF03
5: Cihazda sorun var	0x0000_0000	0x0110_0000	0x0110_0000

Ařaęıdaki drt izelgede, analiz edilen verilerin drt baytının ikili dkm gsterilmiřtir:

izelge 8- 120 rnek 1: Hata yok: 0x01FC\_FF03 deęeri MODE 1 iin dndrlr (Cihaz yapılandırması aktif).

Deęere sahip bayt	Deęere sahip bit yapısı	Notlar
Byte 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xFC	Bit 15 1111-1100 Bit 8	
Byte 3 0xFF	Bit 23 1111-1111 Bit 16	
Byte 4 0x03	Bit 31 0000-0011 Bit 24	

Cihazlar, 10 (Bit 10) ila 25 (Bit 25) arasındaki adreslerle yapılandırılır.

Hibir cihaz 1 ile 9 arasındaki adreslerle yapılandırılmaz.

MODE 4 (Cihaz var) verisi ile MODE 1 (Cihaz yapılandırması aktif) eřleřir, bu nedenle yapılandırılmıř cihazlar mevcut cihazlarla eřleřir.

## Geniřletilmiř komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Çizelge 8- 121 Örnek 2: PROFIBUS slave cihazından "DPSlave\_12" bir modül çekilmiştir.  
 0x0110\_0000 deęeri MODE 2 için döndürülür (Cihaz kusurlu).

Deęere sahip bayt	Deęere sahip bit yapısı	Notlar
Byte 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0x10	Bit 15 0001-0000 Bit 8	
Byte 3 0x00	Bit 23 0000-0000 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Cihaz 12 (Bit 12) kusurlu olarak iřaretlenir.

MOD 5 (Cihazda sorun var), MOD 2 (Cihaz kusurlu) ile aynı bilgiyi döndürür.

Çizelge 8- 122 Örnek 2 (devamı): PROFIBUS slave cihazından "DPSlave\_12" bir modül çekilmiştir.  
 0x01FC\_FF03 deęeri MODE 4 (Cihaz var) için döndürülür.

Deęere sahip bayt	Deęere sahip bit yapısı	Notlar
Byte 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xFC	Bit 15 1111-1100 Bit 8	
Byte 3 0xFF	Bit 23 1111-1111 Bit 16	
Byte 4 0x03	Bit 31 0000-0011 Bit 24	

Cihaz 12 (Bit 12), yukarıda MODE 2'de (Cihaz kusurlu) gösterildięi gibi bir hataya sahip olsa bile, cihazı "mevcut cihaz " olarak göstermek için cihaz yine de MODE 4'e (cihaz var) neden olan aę üzerinde çalıřıyordu.

Çizelge 8- 123 Örnek 3: PROFIBUS slave cihazı "DPSlave\_12" PROFIBUS aędan (kablo baęlantısı veya güç kaybı) kesilir. "DPSlave\_12" hala, kusurlu cihazın yanı sıra cihaz içinde bir hata var olarak algılanır. Fark, "DPSlave\_12" artık var olmayan bir cihaz olarak algılanır olmasıdır. 0x01EC\_FF03 deęeri MOD 4 (cihaz var) için döndürülür.

Deęere sahip bayt	Deęere sahip bit yapısı	Notlar
Byte 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xEC	Bit 15 1110-1100 Bit 8	
Byte 3 0xFF	Bit 23 1111-1111 Bit 16	
Byte 4 0x03	Bit 31 0000-0011 Bit 24	

Cihaz 12 (Bit 12) bulunmuyor olarak iřaretlenir. İstisnai olarak, 10 ile 25 arasındaki cihazlar yine de bulunuyor olarak iřaretlenir.

## PROFINET örneği

PROFINET örneği aşağıdakilerden oluşur:

- 16 PROFINET slave cihazları "et200s\_1" ile "et200s\_16" arasında isimlendirilir.
- 16 PROFINET cihazları sırasıyla 1 ile 16 PROFINET adreslerini kullanır.
- Her slave cihazı çoklu I/O modülleriyle yapılandırılır.
- Döndürülen STATE parametre bilgisinin ilk dört baytı görüntülenir.

MOD	Örnek 1: Hatasız normal çalışma	Örnek 2: Çekilen PROFINET et200s_1 slave modülü	Örnek 3: Bağlantısız PROFINET slave et200s_1
1: Cihaz yapılandırması aktif	0xFFFF_0100	0xFFFF_0100	0xFFFF_0100
2: Cihaz kusurlu	0x0000_0000	0x0300_0000	0x0300_0000
3: Cihaz devre dışı	0x0000_0000	0x0000_0000	0x0000_0000
4: Cihaz var	0xFFFF_0100	0xFFFF_0100	0xFDFF_0100
5: Cihazda sorun var	0x0000_0000	0x0300_0000	0x0300_0000

Aşağıdaki dört çizelgede analiz edilen verilerin dört baytının ikili dökümü gösterilmiştir:

Çizelge 8- 124 Örnek 1: Hata yok: 0xFFFF\_0100 değeri MODE 1 için döndürülür (Cihaz yapılandırması aktif).

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xFF	Bit 15 1111-1111 Bit 8	
Byte 3 0x01	Bit 23 0000-0001 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Cihazlar, 1 (Bit 1) ile 16 (Bit 16) arasındaki adreslerle yapılandırılır.

Hiçbir cihaz 1 ile 9 arasındaki adreslerle yapılandırılmaz.

MODE 4 (Cihaz var) verisi ile MODE 1 (Cihaz yapılandırması aktif) eşleşir, bu nedenle yapılandırılmış cihazlar mevcut cihazlarla eşleşir.

## Geniřletilmiř komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Çizelge 8- 125 Örnek 2: PROFINET slave cihazından "et200s\_1" bir modül çekilmiřtir. 0x0300\_0000 deęeri MODE 2 için döndürölür (Cihaz kusurlu).

Deęere sahip bayt	Deęere sahip bit yapısı	Notlar
Byte 1 0x03	Bit 7 0000-0011 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0x00	Bit 15 0000-0000 Bit 8	
Byte 3 0x00	Bit 23 0000-0000 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Cihaz 1 (Bit 1) kusurlu olarak iřaretlenir. Cihaz yine de varken, MODE 4 (Cihaz var) normal olarak çalıřtıęıyla aynı bilgiyi gösterir.

MOD 5 (Cihazda sorun var) MOD 2 (Cihaz kusurlu) ile aynı bilgiyi döndürölür.

Çizelge 8- 126 Örnek 2 (devamı): PROFIBUS slave cihazından "et200s\_1" bir modül çekilmiřtir. 0xFFFF\_0100 deęeri MODE 4 (Cihaz var) için döndürölür.

Deęere sahip bayt	Deęere sahip bit yapısı	Notlar
Byte 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xFF	Bit 15 1111-1111 Bit 8	
Byte 3 0x01	Bit 23 0000-0001 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Cihaz 1 (Bit 1), yukarıda MODE 2 (Cihaz kusurlu) de gösterildięi gibi bir hataya sahip olsa bile, cihazı "mevcut cihaz " olarak göstermek için cihaz yine de MODE 4'e (cihaz var) neden olan aę üzerinde çalıřıyordu.

Çizelge 8- 127 Örnek 3: PROFINET slave cihazı "et200s\_1" PROFINET aędan (kablo baęlantısı veya güç kaybı) kesilir. 0xFDFD\_0100 deęeri MODE 4 (Cihaz var) için döndürölür.

Deęere sahip bayt	Deęere sahip bit yapısı	Notlar
Byte 1 0xFD	Bit 7 1111-1101 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xFF	Bit 15 1111-1111 Bit 8	
Byte 3 0x01	Bit 23 0000-0001 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

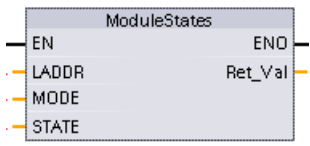
Cihaz 1 (Bit 1) bulunmuyor. Cihaz 2 (Bit 2) ila Cihaz 16 (Bit 16) arası bulunur.



## 8.5.5 ModuleStates komutu

Bir PROFIBUS veya PROFINET istasyon modüllerinin tümünün durumunu döndürmek için ModuleStates komutunu kullanabilirsiniz.

Çizelge 8- 128 ModuleStates komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := ModuleStates(   laddr:=_word_in_,   mode:=_uint_in_,   state:=_variant_inout);</pre>	<p>ModuleStates I/O modüllerinin çalışma durumlarını alır. Yürütme sonrasında, STATE parametresi (atanmış LADDR ve MODE için) bit listesindeki her bir I/O modülünün hata durumunu içerir. Bu bilgiler STEP 7 tanılama görünümünde görülen modül durumuna karşılık gelir.</p> <p>ModuleStates'in LADDR giriři, baş modülün kendisinin değil dağıtılmış I/O istasyonunun bir donanım belirleyicisidir. Donanım belirleyici, ağ görünümündeki tüm istasyonu seçip özellikler altında donanım belirleyici bölümüne bakarak bulunabilir. Aynı zamanda PLC etiket çizelgesinde sistem sabitler sekmesinde "Hw_Device" ve "Hw_DpSlave" veri tiplerine bakarak bulunabilir.</p>

Çizelge 8- 129 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
LADDR	IN	HW_DEVICE
MODE	IN	UInt
		<p>Mantıksal adres (I/O modülleri için belirleyici)</p> <p>Beş adet operasyon modunu destekler. MODE giriři, hangi verilerin STATE bilgisi için belirtilen yere döndürüleceğini belirler. Modlar aşağıda verilmiştir:</p> <ul style="list-style-type: none"> <li>• 1: Modül yapılandırması aktif</li> <li>• 2: Modül kusurlu</li> <li>• 3: Modül devre dışı</li> <li>• 4: Modül var</li> <li>• 5: Modülde sorun var</li> </ul>

## Geniřletilmiř komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Parametre ve tipi		Veri tipi	Açıklama
RET_VAL	OUT	Int	Durum (durum kodu)
STATE <sup>1</sup>	InOut	Variant	<p>Her modülün hata durumunu alan arabellek: STATE parametresi için seçtiđiniz veri tipi herhangi bir bit tipi (Bool, Byte, Word veya DWord) ya da bir bit tiplerinden oluřan bir dizi olabilir.</p> <ul style="list-style-type: none"> <li>Döndürülen STATE verilerinin ilk baytının 0 Biti özet bittir. TRUE olarak ayarlandığında, diđer verilerin mevcut olduđunu gösterir.</li> <li>STATE parametresi tarafından döndürülen veriler, bit konumu ile dağıtılmıř I / O adresi arasında bire bir iliřkiyi göstermektedir. Bu slot adreslemesi PROFIBUS ve PROFINET için TRUE olur. Örneđin, Bař modül, güç modülü ve bir çift I/O modülüne sahip ET 200S için, sırasıyla ilk Bayttaki Bit 1 bař modülüyle, Bit 2 güç modülüyle, Bit 3 ve Bit 4 I/O modülleriyle iliřkilidir</li> </ul>

<sup>1</sup> Maksimum 128 bit atanabilir. Gerekli bit sayısı, I / O modülü kullanımına bađlıdır.

Çizelge 8- 130 Durum kodları

RET_VAL ( W#16#...)	Açıklama
0	Hata yok
8091	LADDR tarafından tanımlanan modül yok.
8092	LADDR tarafından tanımlanan modül I/O cihazını adreslemiyor.
8093	STATE parametresi için atanan geçersiz veri tipi: Geçerli veri tipleri (Bool, Byte, Word veya Dword), veya (Bools, Bytes, Words veya Dwords) dizileri.
80Bx	ModuleStates komutu bu LADDR için CPU tarafından desteklenmiyor.
8452	Tüm durum verisi, atanan STATE parametresi için çok büyük. STATE arabelleđi kısmi bir sonuç içeriyor.

#### 8.5.5.1 ModuleStates örnek konfigürasyonları

##### PROFIBUS örneđi

PROFIBUS örneđi ařađıdakilerden oluřur:

- 16 PROFIBUS cihazları DPSlave\_10" ile "DPSlave\_25" olarak isimlendirilir.
- 16 PROFIBUS cihazı sırasıyla 10 ila 25 PROFIBUS adreslerini kullanır.
- Her slave cihazı çoklu I/O modülleriyle yapılandırılır.
- Bu örnekte; bař modülü, güç modülü ve iki adet I/O modülü içeren PROFIBUS slave "DPSlave\_12" nin LADDR parametresi kullanılır.
- Döndürülen STATE parametre bilgisinin ilk dört baytı görüntülenir.

MOD	Örnek 1: Hatasız normal çalışma	Örnek 2: Çekilen tek modüllü PROFIBUS slave cihazı DPSlave_12	Örnek 3: Bağılantısız PROFIBUS slave cihazı DPSlave_12
1: Modül yapılandırması aktif	0x1F00_0000	0x1F00_0000	0x1F00_0000
2: Modül kusurlu	0x0000_0000	0x0900_0000	0x1F00_0000
3: Modül devre dışı	0x0000_0000	0x0000_0000	0x0000_0000
4: Modül var	0x1F00_0000	0x1700_0000	0x0000_0000
5: Modülde sorun var	0x0000_0000	0x0900_0000	0x1F00_0000

Aşağıdaki dört çizelgede analiz edilen verilerin dört baytının ikili dökümü gösterilmiştir:

Çizelge 8- 131 Örnek 1: Hata yok: 0x1F00\_0000 değeri MODE 1 için döndürülür (Cihaz yapılandırması aktif).

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0x1F	Bit 7 0001-1111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0x00	Bit 15 0000-0000 Bit 8	
Byte 3 0x00	Bit 23 0000-0000 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Slot 1 (Bit 1) ila Slot 4 (Bit 4), modüller içerir. Slot 5 (Bit 5) ve sonrası modül içermez.

MODE 4 (Modül var) verisi ile MODE 1 (Modül yapılandırması aktif) eşleşir, bu nedenle yapılandırılmış modüller mevcut modüllerle eşleşir.

Çizelge 8- 132 Örnek 2: PROFIBUS "DPSlave\_12" slave cihazından bir modül çekilmiştir. 0x0900\_0000 değeri MODE 2 için döndürülür (Modül kusurlu).

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0x09	Bit 7 0000-1001 Bit 0	Bit 0 true olur; veri mevcuttur
Byte 2 0x00	Bit 15 0000-0000 Bit 8	
Byte 3 0x00	Bit 23 0000-0000 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Yalnızca modül 3 (Bit 3) kusurlu olarak işaretlenir. Diğer tüm modüller fonksiyoneldir.

## Genişletilmiş komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Çizelge 8- 133 Örnek 2 (devamı): PROFIBUS "DPSlave\_12" slave cihazından bir modül çekilmiştir. 0x1700\_0000 değeri MODE 4 (Modül var) için döndürülür.

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0x17	Bit 7 0001-0111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0x00	Bit 15 0000-0000 Bit 8	
Byte 3 0x00	Bit 23 0000-0000 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Modül 3 (Bit 3) kayıp olarak gösterilir. Modül 1, 2 ve 4 (Bit 1, 2 ve 4) mevcut olarak gösteriliyor.

Çizelge 8- 134 Örnek 3: PROFIBUS slave cihazı "DPSlave\_12", PROFIBUS ağdan (kablo bağlantısı veya güç kaybı) kesilir. 0x1F00\_0000 değeri MOD 2 (Modül kusurlu) için döndürülür.

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0x1F	Bit 7 0001-1111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0x00	Bit 15 0000-0000 Bit 8	
Byte 3 0x00	Bit 23 0000-0000 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Slot 1 ile slot 4'teki (Bit 1 ile Bit 4) modüller kusurlu olarak işaretlenir, çünkü cihaz kayıptır. MOD 5 (Modülde sorun var), MOD 2 (Modül kusurlu) ile aynı bilgiyi gösterir.

## PROFINET örneği

PROFINET örneği aşağıdakilerden oluşur:

- 16 PROFINET slave cihazları "et200s\_1" ile "et200s\_16" olarak isimlendirilir.
- 16 PROFINET cihazı sırasıyla 1 ile 16 PROFINET adreslerini kullanır.
- Her slave cihazı çoklu I/O modülleriyle yapılandırılır.
- Bu örnekte; baş modülü, güç modülü ve 18 adet I/O modülü içeren PROFINET slave "et200s\_1" kullanılır.
- Döndürülen STATE parametre bilgisinin ilk dört baytı görüntülenir.

MOD	Örnek 1: Hatasız normal çalışma	Örnek 2: Çekilen PROFINET et200s_1 slave modülü	Örnek 3: Bağlantısız PROFINET et200s_1 slave
1: Modül yapılandırması aktif	0xFFFF_1F00	0xFFFF_1F00	0xFFFF_1F00
2: Modül kusurlu	0x0000_0000	0x0180_0000	0xFFFF_1F00
3: Modül devre dışı	0x0000_0000	0x0000_0000	0x0000_0000
4: Modül var	0xFFFF_1F00	0xFF7F_1F00	0x0000_0000
5: Modülde sorun var	0x0000_0000	0x0180_0000	0xFFFF_1F00

## 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Aşağıdaki dört çizelgede analiz edilen verilerin dört baytının ikili dökümü gösterilmiştir:

Çizelge 8- 135 Örnek 1: Hata yok: MODE 1 için 0xFFFF\_1F00 değeri döndürülür (Cihaz yapılandırması aktif).

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xFF	Bit 15 1111-1111 Bit 8	
Byte 3 0x1F	Bit 23 0001-1111 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Slot 1 (Bit 1) ila Slot 20 (Bit 20), modüller içerir. Slot 21 (Bit 21) ve sonrası modül içermez. MODE 4 (Modül var) verisi ile MODE 1 (Modül yapılandırması aktif) eşleşir, bu nedenle yapılandırılmış Modüller mevcut Modüllerle eşleşir.

Çizelge 8- 136 Örnek 2: PROFINET "et200s\_1" slave cihazından bir modül çekilmiştir. 0x0180\_0000 değeri MODE 2 için döndürülür (Modül kusurlu).

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0x01	Bit 7 0000-0001 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0x80	Bit 15 1000-0000 Bit 8	
Byte 3 0x00	Bit 23 0000-0000 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Yalnızca modül 15 (Bit 15) kusurlu olarak işaretlenir. Diğer tüm modüller fonksiyoneldir.

Çizelge 8- 137 Örnek 2 (devamı): PROFIBUS "et200s\_1" slave cihazından bir modül çekilmiştir. 0xFF7F\_1F00 değeri MODE 4 (Modül var) için döndürülür.

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0x7F	Bit 15 0111-1111 Bit 8	
Byte 3 0x1F	Bit 23 0001-1111 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	

Modül 15 (Bit 15) kayıp olarak gösteriliyor. Modül 1 ila 14 (Bit 1 ila 14) ve Modül 16 ila 20 (Bit 16 ila 20) mevcut olarak gösteriliyor.

## Genişletilmiş komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Çizelge 8- 138 Örnek 3: PROFINET slave cihazı "et200s\_1", PROFINET ağından (kablo bağlantısı veya güç kaybı) kesilir. 0xFFFF\_1F00 değeri MOD 2 (Modül kusurlu) için döndürülür.

Değere sahip bayt	Değere sahip bit yapısı	Notlar
Byte 1 0xFF	Bit 7 1111-1111 Bit 0	Bit 0 true olur; veri mevcuttur.
Byte 2 0xFF	Bit 15 1111-1111 Bit 8	
Byte 3 0x1F	Bit 23 0001-1111 Bit 16	
Byte 4 0x00	Bit 31 0000-0000 Bit 24	


Slot 1 ila slot 20'deki (Bit 1 ila Bit 20) modüller kusurlu olarak işaretlenir, çünkü cihaz kayıptır. MOD 5 (Modülde sorun var) MOD 2 (Modül kusurlu) ile aynı bilgiyi gösterir.

## 8.5.6 GET\_DIAG (Tanılama bilgisini oku) komutu

### Açıklama

Bir donanım cihazının tanılama bilgisini okumak için "GET\_DIAG" komutunu kullanabilirsiniz. Donanım cihazı LADDR parametresi ile seçilir. MODE parametresi ile, hangi tanılama bilgisinin okunacağını seçiniz.

Çizelge 8- 139 GET\_DIAG komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := GET_DIAG(     mode:=_uint_in_,     laddr:=_word_in_,     cnt_diag=&gt;_uint_out_,     diag:=_variant_inout_,     detail:=_variant_inout_);</pre>	Atanmış bir donanım cihazının tanılama bilgisini okur.

### Parametreler

Aşağıdaki çizelgede "GET\_DIAG" komutunun parametreleri gösterilmiştir:

Çizelge 8- 140 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
MODE	IN	UInt	Hangi tanılama verisinin çıkış olacağını seçmek için MODE parametresini kullanınız.
LADDR	IN	HW_ANY (Word)	Cihazın donanım ID'si
RET_VAL	OUT	Int	Komut durumu
CNT_DIAG	OUT	UInt	Çıkış tanılama detaylarının sayısı
DIAG	InOut	Variant	Seçilen modla tanılama bilgilerinin depolanması için veri alanını gösteren işaretçi
DETAILS	InOut	Variant	Seçilen moda göre tanılama detaylarının depolanması için veri alanını gösteren işaretçi

### MODE parametresi

MODE parametresindeki değere bağlı olarak, farklı tanılama verileri DIAG, CNT\_DIAG ve DETAILS çıkış parametrelerindeki çıkışıdır.

Çizelge 8- 141 MODE parametresi

MODE	Açıklama	DIAG	CNT_DIAG	DETAILS
0	X modunun desteklendiğini gösteren Bit X = 1 durumunda, DWord olarak bir modül için tüm desteklenen tanılama bilgilerinin çıkışı.	X modunun desteklendiğini gösteren Bit X = 1 durumunda, DWord olarak desteklenen bir modül için tüm modların bit string'i.	0	-
1	Adreslenen donanım nesnesinin doğal durumunun çıkışı.	Tanılama durumu: DIS yapısına uygun çıkış. (Not: Bölümün sonunda aşağıdaki "DIS yapısı" bilgisi ve GET_DIAG komutu örneğine bakınız.)	0	-
2	Adreslenen donanım nesnenin tüm alt modüllerinin durum çıkışı	DNN yapısına uygun tanılama verilerinin çıkışı. (Not: Bölümün sonunda aşağıdaki "DNN yapısı" bilgisine ve GET_DIAG komutu örneğine bakınız.)	0	-

## Geniřletilmiř komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

#### DIS yapısı

MODE parametresi = 1 ile, DIS yapısına gre tanılama bilgileri ıkıřtır. Ařađıdaki izelgede mnferit parametre deęerlerinin anlamı gsterilmektedir:

izelge 8- 142 Tanılama bilgi kaynađının yapısı (DIS)

Parametre	Veri tipi	Deęer	Aıklama
MaintenanceState	Dword	Enum	
		0	Bakım gerekli deęil
		1	Modl veya cihaz devre dıřı.
		2	-
		3	-
		4	-
		5	Bakım gerekli
		6	Bakım isteniyor
		7	Hata
		8	Durum bilinmiyor / alt modlde hata
		9	-
10	Giriřler/ıkıřlar kullanılabilir deęil.		
Componentstate Detail	Dword	Bit array	Modln alt modllerinin durumu: <ul style="list-style-type: none"> <li>Bit 0 ila 15: Modln durum mesajı</li> <li>Bit 16 ila 31: CPU nun durum mesajı</li> </ul>
		0 ila 2 (enum)	İlave bilgi: <ul style="list-style-type: none"> <li>Bit 0: İlave bilgi yok</li> <li>Bit 1: Transfere izin yok</li> </ul>
		3	Bit 3 = 1: Tanılamalar iin en az bir kanal niteleyicileri destekliyor.
		4	Bit 4 = 1: En az bir kanal ya da bir bileřen iin bakım gerekli
		5	Bit 5 = 1: En az bir kanal ya da bir bileřen iin bakım isteniyor
		6	Bit 6 = 1: En az bir kanal ya da bir bileřende hata
		7 ila 10	Ayrılmıř (daima = 0)
		11 ila 14	Bit 11 = 1: PNIO – alt modl doęru Bit 12 = 1: PNIO – yedek modl Bit 13 = 1: PNIO – yanlıř modl Bit 14 = 1: PNIO – modl baęlı deęil
		15	Ayrılmıř (daima = 0)



## 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Parametre	Veri tipi	Değer	Açıklama
		16 ila 31	Modüller için CPU tarafından üretilen durum bilgisi Bit 16 = 1: Modül devre dışı Bit 17 = 1: CiR işlemi aktif Bit 18 = 1: Giriş kullanımında değil Bit 19 = 1: Çıkış kullanımında değil Bit 20 = 1: Taşma tanılayıcı arabellek Bit 21 = 1: Tanılama kullanılabilir değil Bit 22 - 31: Ayrılmış (daima 0)
OwnState	Uint16	Enum	OwnState parametresinin değeri modülün bakım durumunu açıklar.
		0	Arıza yok
		1	Modül veya cihaz devre dışı.
		2	Bakım gerekli
		3	Bakım isteniyor
		4	Hata
		5	Cihaz veya modüle CPU tarafından ulaşılamıyor (CPU altındaki modüller ve cihazlar için geçerlidir).
		6	Giriş/çıkışlar kullanımında değil.
IO State	Uint16	Bit array	Modülün I/O durumu
		0	Bit 0 = 1: Bakım gerekli değil
		1	Bit 1 = 1: Modül veya cihaz devre dışı.
		2	Bit 2 = 1: Bakım gerekli
		3	Bit 3 = 1: Bakım isteniyor
		4	Bit 4 = 1: Hata
		5	Bit 5 = 1: Cihaz veya modüle CPU tarafından ulaşılamıyor (CPU altındaki modüller ve cihazlar için geçerlidir).
		6	Giriş/çıkışlar kullanımında değil.
		7	Niteleyici; bit 7 = 1, bit 0, 2, veya 3 ayarlanırsa
		8 ila 15	Ayrılmış (daima = 0)
OperatingState	UInt16	Enum	
		0	-
		1	In STOP / firmware güncellemesi
		2	In STOP / sıfırlama hafızası
		3	In STOP / kendinden başlatma
		4	In STOP
		5	Bellek sıfırlama
		6	In START
		7	In RUN
		8	-
		9	In HOLD
		10	-
		11	-
12	Modül kusurlu		

## Geniřletilmiř komutlar

### 8.5 Tanılama (Diyagnostik) (PROFINET veya PROFIBUS)

Parametre	Veri tipi	Deęer	Açıklama
		13	-
		14	Güç yok
		15	CiR
		16	In STOP / DIS olmaksızın
		17	In
		18	
		19	
		20	

### DNN'ların yapısı

MOD = 2 parametresi ile, tanılama bilgileriyle ilgili ayrıntılar DNN yapısına uygun çıktır. Ařaęıdaki çizelgede tek tek parametre deęerlerinin anlamı gösterilmektedir:

Çizelge 8- 143 Tanılama Gezinti Düęüm Yapısı (Diagnostic Navigation Node, DNN)

Parametre	Veri tipi	Deęer	Açıklama
SubordinateState	UINT	Enum	Alt modülün Durumu (DIS yapısının OwnState parametresine bakınız.)
SubordinateIOState	WORD	Bitarray	Alt modülün giriş ve çıkıřlarının durumu (DIS yapısının IO State parametresine bakınız.)
DNNmode	WORD	Bitarray	<ul style="list-style-type: none"> <li>Bit 0 = 0: Tanılama etkinleřtirildi</li> <li>Bit 0 = 1: Tanılama devre dıřı bırakıldı</li> <li>Bit 1 ila 15: Ayrıldı</li> </ul>

### RET\_VAL parametresi

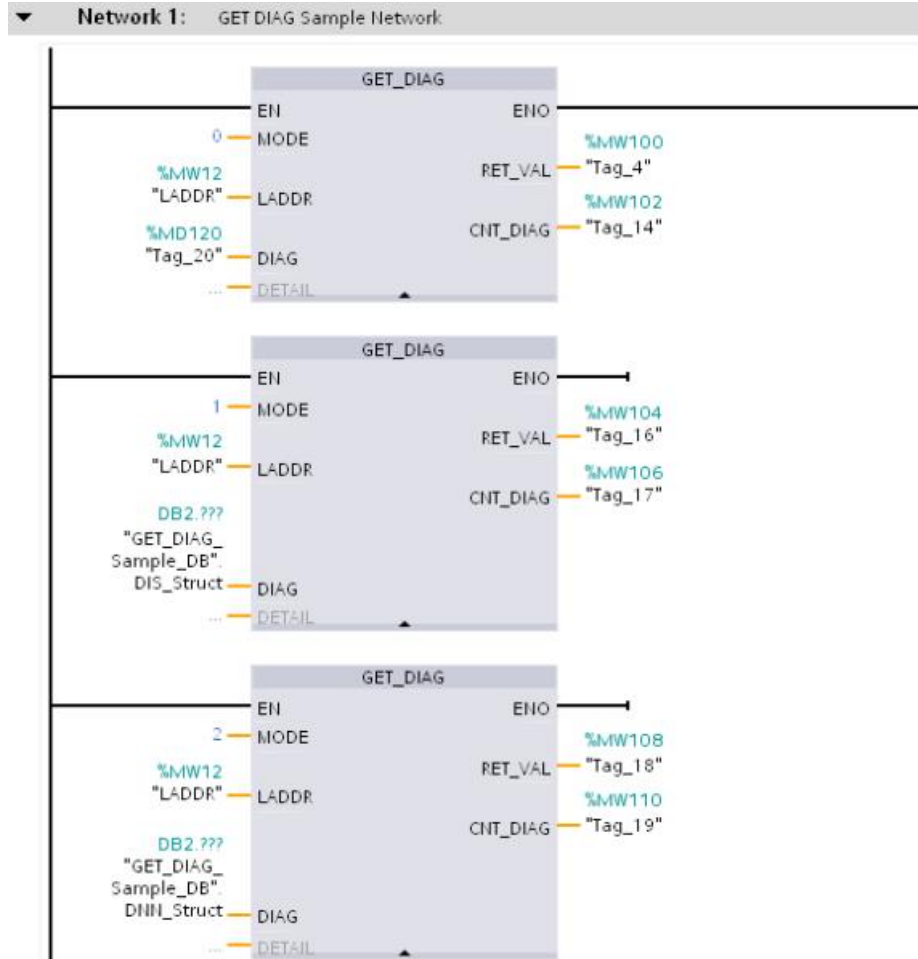
Çizelge 8- 144 RET\_VAL parametresi hata kodları

Hata kodu (W#16#...)	Açıklama
0	Hata yok
8080	MODE parametresi deęeri desteklenmiyor.
8081	DIAG parametresindeki tip seęilen modla (MOD parametre) desteklenmiyor.
8082	DETAILS parametresindeki tip seęilen modla (MOD parametre) desteklenmiyor.
8090	LADDR mevcut deęil
8091	KANAL parametresinde seęilen kanal yok.
80C1	Paralel yürütme için yetersiz kaynaklar

## Örnek

Aşağıdaki merdiven mantık devresi ile DB, üç yapılı üç modun nasıl kullanılacağını gösterir:

- DIS
- DNN



GET_DIAG_Sample_DB							
	Name	Data type	Offset	Start value	Retain	Visible in HMI	Comment
1	Static						
2	▼ DNN_Struct	DNN	0.0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	
3	SubordinateState	UInt	0.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
4	SubordinateIOState	Word	2.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
5	DNNImode	Word	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
6	▼ DIS_Struct	DIS	6.0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	
7	MaintainanceState	DWord	0.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
8	ComponentStateDetail	DWord	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
9	OwnState	UInt	8.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
10	IOState	Word	10.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	
11	OperatingState	UInt	12.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	

- ① DNN  
 ② DIS

### Not


DB'de, üç yapının her birine erişmek için veri tipini elle yazmanız gerekir; Hiçbir açılan liste seçimi yoktur. Veri tiplerindeki tip tam olarak aşağıda gösterildiği gibidir:

- DNN
- DIS

## 8.6 Darbe (pals)

### 8.6.1 CTRL\_PWM (Darbe genişlik modülasyonu) komutu

Çizelge 8- 145 CTRL\_PWM (Darbe genişlik modülasyonu) komutu

LAD / FBD	SCL	Açıklama
	<pre>"CTRL_PWM_DB" (     PWM:=_word_in_,     enable:=_bool_in_,     busy=&gt;_bool_out_,     status=&gt;_word_out_);</pre>	<p>Değişken bir görev çevrimi ile, sabit bir çevrim süresi çıkışı sağlar. PWM çıkışı, belirtilen frekansta (döngü süresi) başlatıldıktan sonra sürekli çalışır. Darbe genişliği istenen kontrole etki etmek için gerektiği gibi değiştirilir.</p>

- 1 Komutu yerleştirdiğinizde STEP 7 otomatik olarak DB oluşturur.
- 2 SCL örneğinde, "CTRL\_PWM\_DB" kopya DB'nin ismidir.

Çizelge 8- 146 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
PWM	IN	HW_PWM (Word) PWM belirtici: Etkinleştirilmiş darbe jeneratörü isimleri "sabit" etiket çizelgesindeki etiketler olacaktır ve PWM parametresi olarak kullanıma hazır olacaktır. (Varsayılan değer: 0)
ENABLE	IN	Bool 1 = Darbe jeneratörünü başlat 0 = Darbe jeneratörünü durdur
BUSY	OUT	Bool Fonksiyon meşgul (Varsayılan değer: 0)
STATUS	OUT	Word Yürütme durum kodu (Varsayılan değer: 0)

CTRL\_PWM komutu DB'deki parametre bilgilerini depolar. Veri bloğu parametreleri kullanıcı tarafından ayrı ayrı değiştirilmez, ancak CTRL\_PWM komutu ile kontrol edilir.

PWM parametresi için etiket adını kullanarak, kullanılacak etkinleştirilmiş darbe jeneratörünü belirtiniz.

EN girişi TRUE olduğunda, PWM\_CTRL komutu, ENABLE girişindeki değere göre tanımlı PWM'yi başlatır veya durdurur. Darbe genişliği, ilişkili Q word çıktı adresindeki değerle belirtilir.

CTRL\_PWM komutu çalıştırıldığında, CPU isteği işlediği için, BUSY parametresi daima FALSE raporu verecektir. Bir hata tespit edilirse, ENO FALSE olarak ayarlanır ve STATUS parametresi bir durum kodu içerir.

CPU, ilk olarak RUN moduna girdiğinde darbe genişliği Cihaz Yapılandırmasında yapılandırılan başlangıç değerine ayarlanır. Darbe genişliğini değiştirmek gerektiğinde ("Çıkış adresleri " / " Başlangıç adresi ") Cihaz Yapılandırmasında belirtilen Q - word konumuna değerleri yazınız. İstenen darbe genişliğini uygun Q word'e yazmak için move, convert, math veya PID box gibi bir komut kullanmalısınız. Q - word değeri (yüzde, binde, on binde ya da S7 analog formatta ) için geçerli bir aralık kullanmanız gerekir.

#### Not

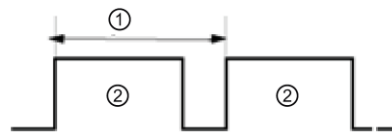
#### PWM ve PTO'ya atanan dijital I/O noktaları zorlanamaz

Darbe genişlik modülasyonu ( PWM ) ve darbe katarı çıkış cihazı ( PTO ) tarafından kullanılan dijital I/O noktaları cihaz yapılandırması sırasında atanır. Dijital I/O nokta adresleri bu cihazlar için atandığı zaman, atanan I/O nokta adresleri değerleri "Watch table force" fonksiyonu tarafından değiştirilemez.

Çizelge 8- 147 STATUS parametre değeri

STATUS	Açıklama
0	Hata yok
80A1	PWM belirteci geçerli bir PWM adreslemez.

## 8.6.2 Darbe çıkışları işlemi



- ① Çevrim süresi
- ② Darbe genişliği

Darbe genişliği, çevrim süresinin yüzde biri (0-100) binde biri (0-1000), on binde biri (0-10000) ya da analog S7 biçimi olarak ifade edilebilir. Darbe genişliği 0'dan (darbe yok, daima kapalı) tam ölçeğe (darbe yok, daima açık) kadar değişebilir.

PWM çıkış, 0'dan tam ölçeğe kadar değiştirilebilir olduğu için, çeşitli şekillerde bir analog çıkış ile aynı olan bir dijital çıkış sağlar. Örneğin, PWM çıkış bir motorun durur vaziyetten tam hıza kadar hızını kontrol etmek için kullanılabilir, ya da bir valfin pozisyonunu kapalı durumdan tam olarak açılmış duruma kadar kontrol etmek için kullanılabilir.

Dört darbe jeneratörü yüksek hızlı darbe çıkış fonksiyonlarını kontrol etmek için kullanılabilir: PWM ve Darbe katarı çıkışı (PTO). PTO hareket kontrol komutları tarafından kullanılır. İki de aynı anda olmamak kaydıyla her bir darbe jeneratörünü PWM veya PTO'ya atayabilirsiniz.

Yerleşik CPU çıkışları kullanılabilir veya isteğe bağlı sinyal kart çıkışlarını kullanabilirsiniz. Çıkış noktası numaraları (varsayılan çıkış yapılandırmasını kabul edilirse) aşağıdaki çizelgede gösterilmiştir. Çıkış noktası numaralandırmasını değiştirirseniz, o zaman bir çıkış noktası numaraları atadığınız numaralar olacaktır. PTO isteğe bağlı olarak kanal başına iki çıkış kullanırken PWM'in tek çıkış gerektirdiğini unutmayınız. Bir çıkış, darbe fonksiyonu için gerekli değilse, diğer kullanımlar için kullanılabilir. I/O atama için aşağıdaki çizelgeye bakınız.

Aşağıdaki çizelgede varsayılan I/O atamaları gösterilmiştir; Ancak, dört darbe jeneratörü herhangi bir yerleşik CPU veya SB dijital çıkış için yapılandırılabilir. Farklı çıkış noktaları farklı gerilimleri ve hızları destekler, bu nedenle PWM/PTO yerleri atanırken bunlar dikkate alınmalıdır.

#### Not

**Darbe-katarı çıkışları kullanıcı programında diğer komutlar tarafından kullanılamaz.**

Darbe jeneratörleri olarak CPU veya sinyal kartının çıkışlarını yapılandırıyorsanız (PWM veya hareket kontrolü PTO komutları ile kullanım için), karşılık gelen çıkış adresleri Q bellekten kaldırılır ve kullanıcı programında başka amaçlar için kullanılamaz. Kullanıcı programı bir darbe jeneratörü olarak kullanılan bir çıkışa bir değer yazarsa, CPU fiziksel çıkışa o değeri yazmaz.

#### Not

**PTO yön çıkışları programınızın başka yerinde kullanım için serbest olabilir.**

Her PTO, iki çıkışın atanmasına ihtiyaç duyar: birisi, bir darbe çıkışı olarak ve diğeri bir yön çıkışı olarak. Yön çıkışını değil sadece darbe çıkışını kullanabilirsiniz. Daha sonra kullanıcı programında diğer amaçlar için yön çıkışını serbest bırakabilirsiniz.

Çizelge 8- 148 Darbe jeneratörler için varsayılan çıkış atamaları <sup>3</sup>

Açıklama	Darbe	Yön
PTO1		
Yerleşik I/O	Q0.0	Q0.1
SB I/O	Q4.0	Q4.1
PWM1		
Yerleşik çıkışlar	Q0.0	-
SB çıkışlar	Q4.0	-
PTO2		
Yerleşik I/O	Q0.2	Q0.3
SB I/O	Q4.2	Q4.3
PWM2		
Yerleşik çıkışlar	Q0.2	-
SB çıkışlar	Q4.2	-
PTO3		
Yerleşik I/O	Q0.4 <sup>1</sup>	Q0.5 <sup>1</sup>
SB I/O	Q4.0	Q4.1

Açıklama	Darbe	Yön
PWM3		
Yerleşik çıkışlar	Q0.4 <sup>1</sup>	-
SB çıkışlar	Q4.1	-
PTO4		
Yerleşik I/O	Q0.6 <sup>2</sup>	Q0.7 <sup>2</sup>
SB I/O	Q4.2	Q4.3
PWM4		
Yerleşik çıkışlar	Q0.6 <sup>2</sup>	-
SB çıkışlar	Q4.3	-

- 1 CPU 1211C Q0.4, Q0.5, Q0.6 veya Q0.7 çıkışlarına sahip değildir. Bu nedenle, bu çıkışlar CPU 1211C'de kullanılamaz.
- 2 CPU 1212C Q0.6 veya Q0.7 çıkışlarına sahip değildir. Bu nedenle, bu çıkışlar CPU 1212C'de kullanılamaz.
- 3 Bu Çizelge CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C ve CPU 1217C PTO/PWM fonksiyonlarına uygulanır.

### 8.6.3

#### PWM için bir darbe kanalı yapılandırması

PWM işlemine hazırlamak için ilk olarak CPU'yu sonra Darbe Jeneratörünü (PTO / PWM) ve sonra PWM1 ila PWM4'ü seçerek aygıt yapılandırmasında bir darbe kanalı yapılandırınız. Darbe jeneratörünü (onay kutusu) etkinleştiriniz. Bir darbe jeneratörü etkinse, benzersiz varsayılan bir isim bu özel darbe jeneratörüne atanır. "Name:" düzenleme kutusunda bunu düzenleyerek bu adı değiştirebilirsiniz, ancak benzersiz bir ad olmalıdır. Etkin darbe jeneratörleri isimleri " constant " etiket çizelgesindeki etiketler olacak ve CTRL\_PWM komutunun PWM parametresi olarak kullanıma hazır olacaktır.

Çizelge 8- 149 CPU çıkış: maksimum frekans

CPU	CPU çıkış kanalı	Darbe ve yön çıkışı	A/B, dördün, yukarı/aşağı ve darbe/yön
1211C	Qa.0 ila Qa.3	100 kHz	100 kHz
1212C	Qa.0 ila Qa.3	100 kHz	100 kHz
	Qa.4, Qa.5	20 kHz	20 kHz
1214C ve 1215C	Qa.0 ila Qa.4	100kHz	100kHz
	Qa.5 ila Qb.1	20 kHz	20 kHz
1217C	DQa.0 ila DQa.3 (.0+, .0- ila .3+, .3-)	1 MHz	1 MHz
	DQa.4 ila DQb.1	100 kHz	100 kHz

Çizelge 8- 150 SB sinyal kartı çıkıřı: maksimum frekans (opsiyonel kart)

SB sinyal kartı	SB çıkıř kanalı	Darbe ve yön çıkıř	A/B, dördün, yukarı/ařađı ve darbe/yön
SB 1222, 200 kHz	DQe.0 ila DQe.3	200kHz	200 kHz
SB 1223, 200 kHz	DQe.0, DQe.1	200kHz	200 kHz
SB 1223	DQe.0, DQe.1	20 kHz	20 kHz

**Not**

Darbe çıkıř jeneratörünün maksimum darbe frekansı; CPU 1217C için 1 MHz ve 1211C CPU, 1212C, 1214C, 1215C için 100 KHz; (Standart SB için) 20 KHz; ya da (yüksek hızlı SB için) 200 KHz. Ancak bu donanım sınırını ařan maksimum hıza veya frekansa sahip bir eksen yapılandırdığınızda, STEP 7 sizi uyarır. Sorunlar sizin uygulamanızla ortaya çıkabilir, bu yüzden donanımın maksimum darbe frekansını ařmadığınızı daima kontrol ediniz.

Ařađıdaki gibi darbe jeneratörünü yeniden adlandırmak, bir yorum eklemek ve parametreleri atama seçeneğiniz vardır:

- Darbe jeneratörü řöyle kullanılır: PWM veya PTO (PWM seçiniz)
- Çıkıř kaynađı: yerleřik CPU veya SB
- Zaman esası: milisaniyeler veya mikrosaniyeler
- Darbe geniřlik biçimi:
  - Yüzde birler (0 -100)
  - Binde birler (0 - 1000)
  - On binde birler (0 - 10000)
  - S7 analog format (0 - 27648)
- Çevrim süresi (aralık 0 ila 16,777,215 ): Çevrim süresi deđerini giriniz. Bu deđer yalnızca Cihaz Yapılandırmasında deđiřtirilebilir.
- Bařlangıç darbe geniřliđi: Bařlangıç darbe geniřlik deđerini giriniz. Darbe geniřliđi deđeri çalışma sırasında deđiřtirilebilir.

Çıkıř adreslerini yapılandırmak için bařlangıç adresini giriniz. Darbe geniřlik deđerini bulmak istediğinizde Q word adresi giriniz.

**Not****Darbe-katarı çıkıřları diđer komutlar tarafından kullanıcı programında kullanılmaz**

Darbe jeneratörleri olarak CPU veya sinyal kartının çıkıřlarını yapılandırdığınızda (PWM veya hareket kontrol komutları ile kullanım için), karřılık gelen çıkıř adresleri Q bellekten kaldırılır ve kullanıcı programında bařka amaçlar için kullanılmaz. Kullanıcı programı bir darbe jeneratörü olarak kullanılan bir çıkıřa bir deđer yazarsa, CPU fiziksel çıkıřa o deđer yazmaz.



Darbe geniřlik deęerleri iin varsayılan konumlar ařaęıdaki gibidir:

- PWM1: QW1000
- PWM2: QW1002
- PWM3: QW1004
- PWM4: QW1006

Bu konumdaki deęer, darbenin geniřlięini kontrol eder ve CPU'nun STOP'tan RUN moduna her geiřinde yukarıda belirtilen "İlk darbe geniřlięi:" deęeri bařlatılır. Darbe geniřlięinde bir deęiřiklięe yol amak iin alıřma zamanı sırasında bu Q-word deęerini deęiřtirirsiniz.

## 8.7 Tarifler ve veri gnlkleri

### 8.7.1 Tarifler

#### 8.7.1.1 Tarife genel bakıř

##### Tarif verisi depolama

- Projenizde oluřturduęunuz bir tarif veri bloęu CPU yk belleęinde saklanmalıdır. Dahili CPU belleęi ya da harici bellek "Program" kartı kullanılabilir.
- Oluřturmanız gereken bařka bir DB, aktif tarif veri bloęudur. Bu DB bir adet aktif tarif kaydı okunan ya da program mantıęınız ile yazılan alıřma hafızasında olmalıdır.

##### Tarif veri ynetimi

DB tarifi, rn tarifi kayıtlarından oluřan bir dizi kullanır. Tarif dizisinin her bir oęesi ortak bileřenler grubuna dayalı farklı bir nitelik tarifini temsil eder.

- tek bir tarif kaydındaki tm bileřenleri tanımlayan bir PLC veri tipi veya yapı oluřturunuz. Bu veri tipi řablonu, tm tarif kayıtları iin yeniden kullanılır. rn tarifleri, tarif bileřenlerine atanan bařlangı deęerlerine gre deęiřir.
- Bir READ\_DBL komutu kullanılarak herhangi bir zamanda DB tarifi (yk belleęindeki tm tarifler) aktif DB tarifine(alıřma belleęindeki bir tarif) transfer edilebilir. Bir tarif kaydı alıřma hafızasına tařındıktan sonra, program mantıęınız, bileřen deęerlerini okur ve bir retim alıřtırması bařlatabilir. Bu transfer, tarif veriler iin gerekli CPU alıřma bellek miktarını en aza indirir.
- Aktif tarif bileřen deęerleri, bir retim alıřtırması sırasında bir HMI cihazı tarafından ayarlanırsa, WRIT\_DBL komutu kullanılarak, deęiřtirilen deęerleri tekrar DB tarifine yazabilirsiniz.

**Tarifi dışı aktarılması (DB tarifiinden CSV dosyasına)**

Komple tarif kayıtları grubu RecipeExport komutu kullanılarak bir CSV dosyası olarak oluşturulabilir. Kullanılmayan tarif kayıtları da dışı aktarılmaktadır.

**Tarifi içe aktarılması (CSV dosyasından DB tarifine)**

Bir tarif verme işlemini tamamlandıktan sonra, bir veri yapısı şablonu olarak oluşturulan CSV dosyasını kullanabilirsiniz.

- 11.CPU'dan PC'ye mevcut CSV tarif dosyasını indirmek için CPU web sunucusundaki dosya tarayıcı sayfasını kullanınız.
- 12.Bir ASCII metin editörü ile CSV tarif dosyasını değiştirin. Veri tipleri veya veri yapısını değil ancak bileşenlere atanan başlangıç değerlerini değiştirebilirsiniz.
- 13.Değiştirilmiş CSV dosyasını PC'den CPU'ya geri yükleyiniz. Ancak CPU web sunucusu yükleme çalışmasına izin vermeden önce, CPU yükleme belleğindeki eski CSV dosyası (aynı adlı) silinmeli veya yeniden adlandırılmalıdır.
- 14.Değiştirilmiş CSV dosyası CPU'ya yüklendikten sonra , yeni başlangıç değerlerini değiştirilmiş CSV dosyasından (CPU yükleme belleğinde) DB tarifine (CPU yükleme belleğinde) aktarmak için RecipelImport komutunu kullanabilirsiniz.

**8.7.1.2 DB tarifi örneği****Örnek tarifler**

Aşağıdaki çizelgede, bir DB tarifiinde kullanılmak üzere tarif bilgilerinin nasıl hazırlanacağı gösterilmiştir. Örnek DB tarifi üçü kullanılan beş kayıt saklar. Dördüncü ve beşinci kayıtları daha sonraki açılımlar için serbesttir. Her bir Çizelge satırı, tarifi adı, bileşen veri tipleri ve bileşen değerlerini saklayan bir kaydı temsil eder.

ürün ismi	su	arpa	buğday	Şerbetçi otu	maya	su sıcaklığı	karışım sıcaklığı	karışım süresi	QTest
Bira	10	9	3	280	39	40	30	100	0
Hafif bira	10	9	3	150	33	50	30	120	0
Koyu bira	10	9	3	410	47	60	30	90	1
Kullanılmaz	0	0	0	0	0	0	0	0	0
Kullanılmaz	0	0	0	0	0	0	0	0	0

## Bir tarif veri bloęu oluřturma

### Not

#### Tarif veri blokları iin kurallar

- DB tarifi, PLC veri tipi veya bir yapıdan oluřan tek boyutlu dizi iermelidir. Tarif rneęi bir PLC veri tipine sahip bir DB tarifinin nasıl oluřturulacaęını gsterir.
- Bu rnekte, bileřen maddelerinin veri tipi, UINT veri tipidir. Bileřen veri tipleri de yapılar dıřında herhangi bir veri tipi karıřımı olabilir. Bir DB tarifi dizi elemanı iinde, bir yapı iindeki bir yapı veya bir PLC veri tipi iindeki bir yapıya izin verilmez.

## İlk olarak, yeni bir PLC veri tipi oluřturma

İsmi tarif tipi olan yeni bir PLC veri tipi ekleyiniz. Ařaęıdaki grntde, "Beer\_Recipe" basit veri tipleri dizisi saklayan yeni karmařık PLC veri tipidir."Beer\_Recipe" PLC veri tipi, her DB tarif kaydında ve aynı zamanda aktif DB tarifinde yeniden kullanılan bir veri řablonudur. Tm rnek tariflerde ortak olan bileřen isimleri ve veri tiplerini giriniz. Mnferit bileřen deęerleri DB Tarifinde daha sonra ilave edilir.

Beer_Recipe			
	Name	Data type	Default value
1	productname	String[20]	'Beer_Recipe'
2	water	UInt	0
3	barley	UInt	0
4	wheat	UInt	0
5	hops	UInt	0
6	yeast	UInt	0
7	waterTmp	UInt	0
8	mashTmp	UInt	0
9	mashTime	UInt	0
10	QTest	UInt	0

### İkinci olarak, bir tarif veri bloęu oluřturmak

- Etkinleřtirilen "Ykleme belleęinde saklayın" DB özellięine sahip global bir veri bloęu olarak DB tarifinizi oluřturunuz.
- Bir tarif veri bloęunun ismi, karřılık gelen CSV dosyasının dosya adı olarak kullanılır. Atadığınız DB isim karakterleri, Windows dosya sistemi isimlendirme kısıtlamalarına uymalıdır. \ / : \* ? " < > | karakterlerine ve bořluk karakterine izin verilmez.
- Tarif dizisi atama "Beer\_Recipe"dan oluřan dizi olarak "rnler" [1 .. 5]'dir. Dizi boyutu 5, mmkn olan maksimum sayıdaki lezzetler tarifidir.
- Tarif bileřenleriyle ilgili deęerler DB bařlangıç deęerleri olarak ilave edilir.

Ařaęıdaki grntde, "blackbeer" tarifi tarif kaydının tm bileřenlerini gsterecek řekilde geniřletilmiřtir.

Recipe_DB				
	Name	Data type	Offset	Start value
1	Static			
2	Products	Array [1 .. 5] of "Beer_Recipe"	...	
3	Products[1]	"Beer_Recipe"	...	
4	Products[2]	"Beer_Recipe"	...	
5	Products[3]	"Beer_Recipe"	...	
6	productname	String[20]	...	'BlackBeer'
7	water	UInt	...	10
8	barley	UInt	...	9
9	wheat	UInt	...	3
10	hops	UInt	...	410
11	yeast	UInt	...	47
12	waterTmp	UInt	...	60
13	mashTmp	UInt	...	30
14	mashTime	UInt	...	90
15	QTest	UInt	...	1
16	Products[4]	"Beer_Recipe"	...	
17	Products[5]	"Beer_Recipe"	...	

### Tarifi dıřa aktarma (DB tarifinden CSV dosyasına)

"RecipeExport (Sayfa 374)" yrtmesi, ařaęıdaki metin dosyasında gsterildięi gibi DB tarif verisinden bir CSV dosyasına transfer iřlemini yapar.

Recipe\_DB.csv

```
index,productname,water,barley,wheat,hops,yeast,waterTmp,
mashTmp,mashTime,QTest
1,"Pils",10,9,3,280,39,40,30,100,0
2,"Lager",10,9,3,150,33,50,30,120,0
3,"BlackBeer",10,9,3,410,47,60,30,90,1
4 "Not_used",0,0,0,0,0,0,0,0,0,0
5 "Not_used",0,0,0,0,0,0,0,0,0,0
```

### Tarifi ie aktarma (CSV dosyasından DB tarifine)

1. Mevcut CSV tarif dosyasını CPU ykleme belleğinden PC'ye indirmek iin CPU web sunucusu dosya tarayıcı sayfasını kullanınız.
2. Bir ASCII metin editr ile CSV tarif dosyasını deėiřtiriniz. Veri tipleri veya veri yapısını deėil ancak bileřenlere atanan bařlangı deėerlerini deėiřtirebilirsiniz.
3. Deėiřtirilmiş CSV dosyasını PC'den CPU'ya geri ykleyiniz. Ancak CPU web sunucusu ykleme alıřmasına izin vermeden nce, CPU ykleme belleğindeki eski CSV dosyası (aynı adlı) silinmeli veya yeniden adlandırılmalıdır.
4. Deėiřtirilmiş CSV dosyası CPU'ya ykledikten sonra , yeni bařlangı deėerlerini deėiřtirilmiş CSV dosyasından (CPU ykleme belleğinde) DB tarifine (CPU ykleme belleğinde) aktarmak iin RecipelImport komutunu kullanabilirsiniz.

### CSV dosyaları karřılık gelen tarif DB yapısıyla tamamen eřleřmelidir

- CSV dosyasındaki deėerleri deėiřtirilebilirsiniz, ancak yapısının deėiřtirilmesine izin verilmez . RecipelImport komutunda, kayıtlar ve bileřenlerin tam sayısının hedef DB tarifi yapısıyla eřleřmesi gerekir . Aksi takdirde, RecipelImport yrtmesi bařarısız olur. rneėin, DB tarifinde 10 tarif tanımlanmış ancak sadece 6'sı kullanımda ise, CSV dosyasındaki 7 ila 10 satırları DB'ye aktarılır. Bu verinin geerli olup olmadıėı kontrol edilmelidir. rneėin, kullanılmayan tarif kayıtlarında rn ismi iin bir " Not\_used " deėiřkeni atayabilirsiniz.
- Metin dosyasına veri kaydı eklemek ve deėiřtirilmiş dosyayı ie aktarmak isterseniz, atadıėınız DB dizi limitinin tm tarif kayıtları iin yeterli geye sahip olduėundan emin olunuz.
- CSV dosyasına aktarım yapılması sırasında otomatik olarak bir indeks numarası oluřturulur . Ek veri kayıtları oluřturmak isterseniz, buna uygun olarak ardışık dizin numaraları ekleyebilirsiniz.
- RecipelImport yrtmesi, CSV dosyası verilerini doėru yapı ynnden ve deėerlerin ilgili DB tarifinde atanan veri tiplerine uygun olup olmaması bakımından kontrol eder. rneėin, bir Bool veri tipi bir tamsayı deėeri saklar ve RecipelImport yrtmesi bařarısız olur.

### Excel'de CSV tarif verilerini grntlemek


CSV dosyası, okuma ve dzenlemeyi kolaylařtırmak iin Excel'de aılabilir. Virgller ondalık ayrılar olarak tanınmazsa, yapılandırılmış biimdeki veri ıkışı iin Excel import iřlevini kullanınız.

	A	B	C	D	E	F	G	H	I	J	K
1	index	product	water	barley	wheat	hops	yeast	waterTmp	mashTmp	mashTime	QTest
2	1	"Pils"	10	9	3	280	39	40	30	100	0
3	2	"Lager"	10	9	3	150	33	50	30	120	0
4	3	"BlackBeer"	10	9	3	410	47	60	30	90	1
5	4	"Not_used"	0	0	0	0	0	0	0	0	0
6	5	"Not_used"	0	0	0	0	0	0	0	0	0

### 8.7.1.3 Tarif verisini aktaran program komutları

#### RecipeExport (Recipe export) komutu

Çizelge 8- 151 RecipeExport komutu

LAD/FBD	SCL	Açıklama
<p>"RecipeExport_DB"</p> 	<pre>"RecipeExport_DB" (     req=_bool_in_,     done=&gt;_bool_out_,     busy=&gt;_bool_out_,     error=&gt;_bool_out_,     status=&gt;_word_out_,      Recipe_DB:=_variant_inout_);</pre>	<p>"RecipeExport" komutu CSV dosya formatında bir tarif veri bloğunun tüm tarif kayıtlarını dışa aktarır. CSV dosyası ürün adları, bileşen isimleri ve başlangıç değerlerini içerir. Opsiyonel harici "program" bellek kartı takılı ise CSV dosyası, dahili yükleme belleği ya da harici yükleme belleğinde depolanır.</p> <p>Dışa aktarım işlemi " REQ " parametresi ile tetiklenir. BUSY parametresi, dışa aktarım işlemi sırasında "1" olarak ayarlanır. RecipeExport yürütülmesi durduktan sonra, BUSY "0" olarak sıfırlanır ve işlemin tamamlanması DONE parametresinde "1" ile gösterilir. Yürütme sırasında bir hata oluşursa, ERROR ve STATUS parametreleri sonuç gösterir.</p>

Bir tarifin dışa aktarımı mümkün olmadan önce bir DB tarifinin oluşturulması gerekir. Bir tarif veri bloğunun ismi, yeni bir CSV dosyasının dosya adı olarak kullanılır. Aynı ada sahip bir CSV dosyası zaten varsa, o aktarma işlemi sırasında yazılır.

CSV tarif dosyasına erişmek için CPU'nun yerleşik web sunucusu Dosya Tarayıcı sayfasını (Sayfa 622) kullanabilirsiniz. Dosya, CPU yükleme belleğinin kök dizinindeki tarif klasörüne konulur.

Çizelge 8- 152 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Kontrol parametresi REQUEST: Pozitif kenarda aktarımı aktifleştirilir.
RECIPE_DB	In/Out	Variant	Tarif veri bloğu işaretçisi. Ayrıntılar için " DB tarif örneği (Sayfa 370)"e bakınız. DB isim karakterleri, Windows dosya sistemi adlandırma kısıtlamalarına uymalıdır. \/: *? "<>   karakterleri ve boşluk karakterine izin verilmez.
DONE	OUT	Bool	Son istek hatasız tamamlandıktan sonra DONE biti, bir tarama için TRUE olur. (Varsayılan değer: False)
BUSY	OUT	Bool	RecipeExport yürütmesi <ul style="list-style-type: none"> <li>0: Yapılan işlem yok</li> <li>1: İşlem devam ediyor</li> </ul>

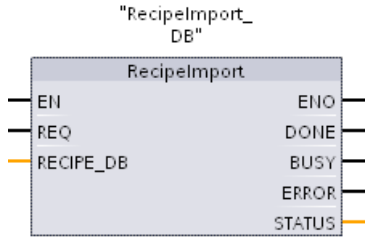
Parametre ve tipi		Veri tipi	Açıklama
ERROR	OUT	Bool	Son istek hatasız tamamlandıktan sonra ERROR biti, bir tarama için TRUE olur. STATUS parametresindeki hata kodu değeri yalnızca ERROR = TRUE olan tekli tarama sırasında geçerli olur. <ul style="list-style-type: none"> <li>0: Uyarı veya hata yok</li> <li>1: Hata oluştu. STATUS parametresi hata tipi hakkında bilgi sağlar.</li> </ul>
STATUS	OUT	Word	Yürütme durum kodu

Çizelge 8- 153 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#...)	Açıklama
0	0000	Hata yok
0	7000	REQ kenarı olmayan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarı olan ilk çağrı (çalışıyor): BUSY = 1, DONE = 0
0	7002	N. çağrı (çalışıyor): BUSY = 1, DONE = 0
1	8070	Tm kopya kaydı kullanımda.
1	8090	Dosya ismi geçersiz karakterler içeriyor.
1	8091	RECIPE_DB ile referans gösterilen veri yapısı işlenemez.
1	8092	RECIPE_DB'de belirtilen veri yapısı 5000 baytı aşıyor
1	80B3	MC veya dahili yükleme belleğinde yeterli alan yok
1	80B4	MC yazmaya karşı korumalı
1	80B6	DB tarif niteliđi "Sadece yükleme belleğinde sakla" etkin deđil.
1	80C0	CSV dosyası geçici olarak kilitli
1	80C1	DB geçici olarak kilitli

## RecipeImport (Recipe Import) (tarif alma) komutu

Çizelge 8- 154 RecipeImport komutu

LAD/FBD	SCL	Açıklama
	<pre>"RecipeImport_DB" (     req:=_bool_in_,     done=&gt;_bool_out_,     busy=&gt;_bool_out_,     error=&gt;_bool_out_,     status=&gt;_word_out_,      Recipe_DB:=_variant_inout_);</pre>	<p>"RecipeImport" komutu RECIPE_DB parametresi tarafından başvuru bir tarif veri bloğuna, CPU yüklemeye belleğindeki bir CSV dosyasından tarif verilerini aktarır. Tarif veri bloğundaki başlangıç değerleri alma işlemi tarafından üzerine yazılır. Alma işlemi "REQ" parametresi ile tetiklenir. BUSY parametresi, alma işlemleri sırasında "1" olarak ayarlanır. RecipeImport yürütülmesi durduktan sonra, BUSY "0" olarak ayarlanır ve işlemin tamamlanması DONE parametresinde "1" ile gösterilir. Yürütme sırasında bir hata oluşursa, ERROR ve STATUS parametreleri sonuç gösterir.</p>

Çizelge 8- 155 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	REQUEST kontrol parametresi: pozitif kenarda aktarım aktifleştirilir.
RECIPE_DB	In/Out	Variant	Tarif veri bloğu işareti. Ayrıntılar için " DB Tarifi örneği (Sayfa 88)" e bakınız. DB isim karakterleri, Windows dosya sistemi adlandırma kısıtlamalarına uymalıdır. \/: *? "<>   karakterleri ve boşluk karakterine izin verilmez.
DONE	OUT	Bool	Son istek hatasız tamamlandıktan sonra DONE biti, bir tarama için TRUE olur. (Varsayılan değer: False)
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>0 - Yapılan işlem yok</li> <li>1 - İşlem devam ediyor</li> </ul>
ERROR	OUT	Bool	Son istek hatasız tamamlandıktan sonra ERROR biti, bir tarama için TRUE olur. STATUS parametresindeki hata kodu değeri yalnızca ERROR = TRUE olan tekli tarama sırasında geçerli olur.
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)



Bir tarif alma işlemi mümkün olmadan önce CSV dosyası veri yapısı ile tutarlı bir yapı içeren bir DB tarifi bulunmalıdır.

CSV dosyası kuralları:

- Opsiyonel harici "program" bellek kartı takılı ise CSV dosyası , dahili yükleme belleğinde ya da harici yükleme belleğindeki " Tarifler " kök dizini klasöründe yer almalıdır.
- CSV dosyasının adı, RECIPE\_DB parametresindeki veri bloğunun adı ile eşleşmelidir.
- CSV dosyasının ilk satırı (başlık) tarif bileşenlerinin adını içerir . İlk satır, alma işlemi sırasında göz ardı edilir. CSV dosyasında ve veri bloğundaki tarif bileşenlerinin adları alma işlemi sırasında karşılaştırılmaz.
- Her durumda CSV dosyasının her satırındaki ilk değer tarifin indeks numarasıdır . Münferit tarifler indeks sırasına göre alınır. Bunun için CSV dosyasındaki dizin, artan sırada olmalı ve hiç boşluk (durum böyle değilse , STATUS parametresinde 80B0 hata mesajı çıkışı verilir) içermeyebilir.
- CSV dosyası, tarif veri bloğu için sağlanandan daha fazla tarif veri kaydı içermeyebilir . Maksimum veri kaydı sayısı, veri bloğundaki dizi sınırları ile belirtilir.

Çizelge 8- 156 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#....)	Açıklama
0	0000	Hata yok
0	7000	REQ kenarı olmayan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarı olan ilk çağrı(çalışıyor): BUSY = 1, DONE = 0
0	7002	N. çağrı (çalışıyor): BUSY = 1, DONE = 0
1	8070	Tüm kopya kaydı kullanımda.
1	8090	Dosya ismi geçersizler karakter içeriyor.
1	8092	Alma işlemi sırasında eşleşen herhangi bir CSV dosyası yok. Olası nedeni: CSV dosyasının ismi DB tarif ismiyle eşleşmiyor.
1	80C0	CSV dosyası geçici olarak kilitli
1	80C1	Veri bloğu geçici olarak kilitli.
1	80B0	CSV dosyasının indeksindeki numaralandırma, sürekli değil, artan değil veya veri bloğundaki maksimum sayıyı aşıyor (dizi sınırı).
1	80B1	Tarif veri bloğunun ve CSV dosyasının yapısı eşleşmiyor: CSV dosyası çok fazla alan içeriyor.
1	80B2	Tarif veri bloğunun ve CSV dosyasının yapısı eşleşmiyor: CSV dosyası çok az alan içeriyor.
1	80B6	DB Tarifinin "Sadece yükleme belleğinde sakla" niteliği etkin değil..
1	80D0 +n	Tarif veri bloğunun ve CSV dosyasının yapısı eşleşmiyor: n alanındaki veri tipi (n<46) eşleşmiyor.
1	80FF	Tarif veri bloğunun ve CSV dosyasının yapısı eşleşmiyor: n alanındaki veri tipi (n>46) eşleşmiyor.

### 8.7.1.4 Örnek tarif programı

#### Örnek tarif programı için önkoşullar

- Tüm tarif kayıtlarını depolayan bir tarif DB. Tarif DB yükleme belleğinde depolanır.
- Çalışma belleğinde bir tarifin kopyasını saklayan bir aktif tarif DB.

Tarif DB ve karşılık gelen CSV dosyası hakkında ayrıntılı bilgi için "Tarif DB Örneği (Sayfa 370)" bakınız.

#### Aktif tarif DB oluşturma

"Yeni blok ekle" penceresinde:

- "Yeni blok ekle" penceresindeki "Veri bloğu" düğmesini seçiniz
- "Tip" açılır menüsünden, önceden oluşturduğunuz "Beer\_recipe" PLC veri tipini seçiniz.

Başlangıç değerleri gerekli değildir. Bir tarif, tarif DB'den aktif tarif DB'ye transfer edildiğinde DB veri değerleri ayarlanır. Bu örnekte, aktif tarif DB, READ\_DBL verileri için bir hedeftir ve WRITE\_DBL için kaynak verileri sağlar. Aşağıdaki görüntü Active\_Recipe DB'yi gösterir.

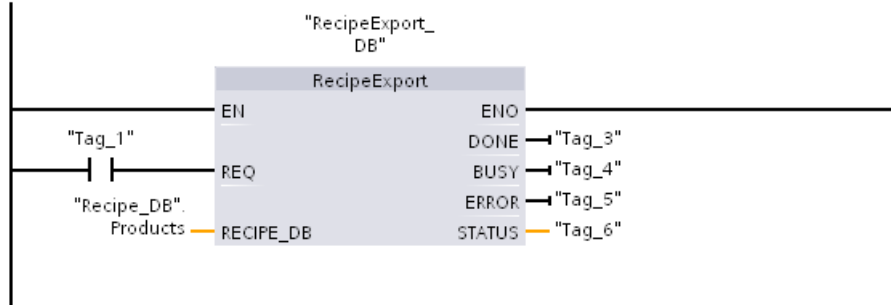
Active_Recipe			
	Name	Data type	Start value
1	Static		
2	productname	String[20]	'Beer_Recipe'
3	water	UInt	0
4	barley	UInt	0
5	wheat	UInt	0
6	hops	UInt	0
7	yeast	UInt	0
8	waterTmp	UInt	0
9	mashTmp	UInt	0
10	mashTime	UInt	0
11	QTest	UInt	0

#### Kopya DB'ler

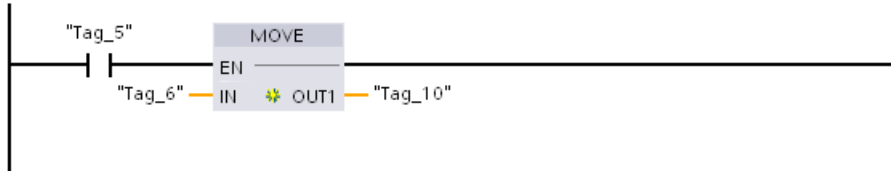
Programınızda komutları yerleştirdiğinizde, RecipeExport ("RecipeExport\_DB") ve RecipeImport ("RecipeImport\_DB") komutları tarafından kullanılan kopya DB'ler otomatik olarak oluşturulur. Kopya DB'ler komut yürütülmesini kontrol etmek için kullanılır ve program mantığında referans gösterilmez.

## Örnek tarif programı

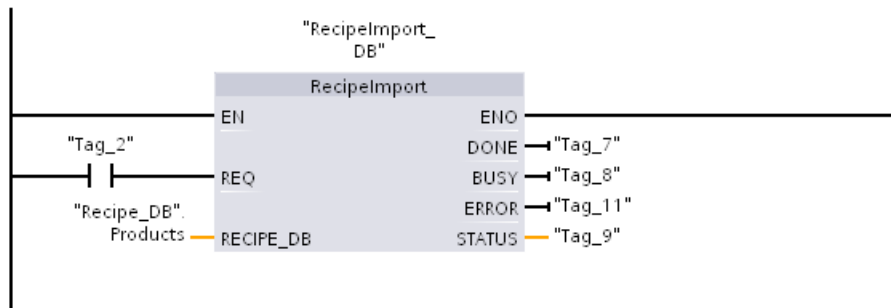
**Devre 1** REQ üzerindeki pozitif bir kenar, aktarma sürecini başlatır. Bir CSV dosyası, tarif DB verilerinden oluşturulur ve CPU bellek tarifleri klasörüne yerleştirilir.



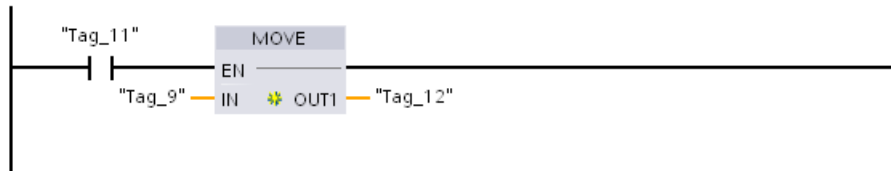
**Devre 2** Bu yalnızca bir tarama için geçerli olduğundan RecipeExport yürütmesinden ortaya çıkan STATUS çıkışı yakalayabilir.



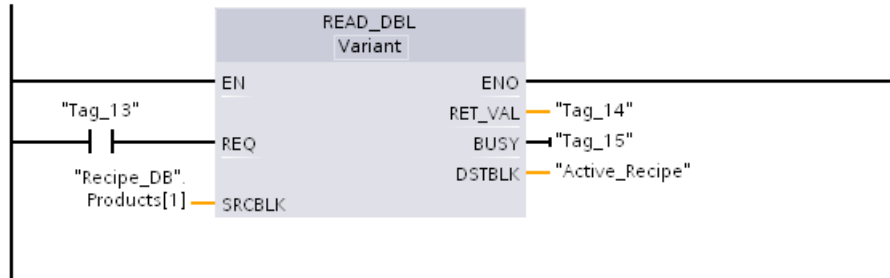
**Devre 3** REQ üzerindeki pozitif bir kenar alma işlemini başlatır. Mevcut tarif DB, CPU bellek tarifleri klasöründe bulunan karşılık gelen CSV dosyasından okunan tüm tarif verileriyle birlikte yüklenir



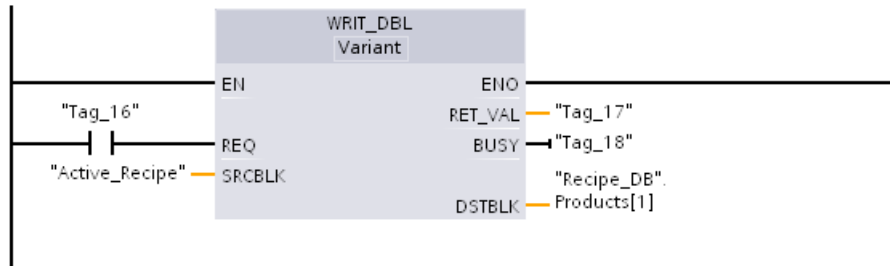
**Devre 4** Bu yalnızca bir tarama için geçerli olduğundan RecipeImport yürütmesinden ortaya çıkan STATUS çıkışı yakalayabilir.



**Devre 5** READ\_DBL, bir adet tarif "Recipe\_DB". Products [1] (CPU ykleme belleğinde) gelen bařlangıç deęerlerini mevcut Active\_Recipe DB deęerlerine (CPU alıřma belleğinde) kopyalar. READ\_DBL yrtmesinden sonra, program mantıđınız Active\_Recipe DB yerleri adresleyerek tarif bileřen deęerlerine eriřebilirsiniz. rneđin, sembolik adresler ("Active\_Recipe". Productname) ve ("Active\_Recipe.water) mevcut tarif adı ve su miktarı ile program mantıđınızı saęlar.



**Devre 6** alıřma sırasında, bir HMI cihazı Active\_Recipe DB'de saklanan bir bileřen deęerini deęiřtirebilir. Geliřtirilmiř tarif verileri WRIT\_DBL'yi yrterek saklanabilir. rnekte, tek tarif "Recipe\_DB". Products[1] iin tm Recipe\_DB bařlangıç deęerleri zerine "Active\_Recipe" DB den gelen mevcut deęerler yazılır.



## 8.7.2 Veri gnlkleri (logları)

Kontrol programınız kalıcı log dosyalarındaki alıřma zamanı veri deęerlerini depolamak iin veri log komutlarını kullanabilir. Veri log dosyaları flash bellekte ( CPU veya bellek kartı) saklanır. Log dosya verisi, standart CSV (Virglle Ayrılmıř Deęer) formatında saklanır . Veri kayıtları, nceden belirlenmiř bir boyuttaki bir dngsel log dosyası olarak dzenlenir.

log dosyaları oluřturmak, amak, kayıt yazmak ve kapatmak iin programınızda veri log komutları kullanılır. Tek bir log kaydı tanımlayan bir veri ara belleđi oluřturarak hangi program deęerlerinin log kaydı olarak girileceđine karar verirsiniz. Veri ara belleđiniz, yeni bir log kaydı iin geici saklama yeri olarak kullanılır. Yeni mevcut deęerlerinin alıřma zamanı sırasında ara bellek iine programlı olarak tařınması gerekir. Mevcut veri deęerlerinin tm gncel olduđunda, ara bellekten gelen verinin bir veri log kaydına aktarılması iin DataLogWrite komutunu yrtebilirsiniz.

Web Sunucusunun Dosya Tarayıcı sayfasından veri log dosyalarını aabilir, dzenleyebilir, kaydedebilir, yeniden adlandırabilir ve silebilirsiniz. Dosya tarayıcısını grntlemek iin okuma ayrıcalıđına sahip olmalısınız ve veri log dosyalarını dzenlemek, silmek veya yeniden adlandırmak iin deęiřtirme ayrıcalıklarına sahip olmalısınız.

### 8.7.2.1 Veri log kayıt yapısı

DataLogCreate komutunun DATA ve HEADER parametreleri, log kaydındaki tüm veri elemanlarının veri tipini ve sütun başlığı açıklamasını atar.

#### DataLogCreate komutu için DATA parametresi

DATA parametresi, yeni bir log kaydı için geçici ara bellek olarak kullanılan belleği işaret eder ve M veya DB konumuna atanması gerekir .

Bir bütün DB (DB oluşturulduğunda atadığınız PLC veri tipinden türetilmiş) veya bir DB parçası (belirtilen DB elemanı yalnızca herhangi bir veri tipi, veri tipi yapısı, PLC veri tipi veya veri dizisi olabilir) atayabilirsiniz.

Yapı veri tipleri, tekli yuvalama düzeyiyle sınırlıdır. Beyan edilen veri elemanlarının toplam sayısı, başlık parametresinde belirtilen sütun sayısına karşılık gelmelidir. Atadığınız veri elemanlarının maksimum sayısı 253 ( zaman etiketli) veya 255' tir (zaman etiketsiz ). Bu kısıtlama, kaydınızı bir Excel sayfasının 256 sütun sınırı içinde tutar.

DATA parametresi, bir "Standart" ( S7-300/400 ile uyumlu) veya "Optimize" DB tipinde kalıcı veya kalıcı olmayan veri elemanlarını atayabilir.

Veri log kaydı yazmak için önce yeni bir süreç değerlerine sahip geçici DATA kaydını yüklemek ve ardından veri log dosyasındaki yeni kayıt değerlerini kaydeden DataLogWrite komutunu yürütmeniz gerekir.

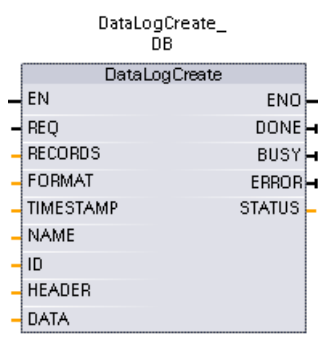
#### DataLogCreate komutu için HEADER parametresi

HEADER parametresi CSV dosyasında kodlanmış veri matrisinin üst satırı için sütun başlığı isimlerini işaret eder. HEADER verisi, DB veya M hafızada yer almalı ve karakterler her bir sütun adının virgül ile ayrıldığı standart CSV formatı kurallarına uymalıdır. Veri tipi bir string, bayt dizisi ya da karakter dizisi olabilir. Karakter/byte dizileri string'lerin maksimum 255 bayt ile sınırlandırıldığı yerde artırılmış boyuta izin verir. HEADER parametresi opsiyoneldir. HEADER atanmış değilse, o zaman hiçbir başlık satırı Veri log dosyasında oluşturulmaz.

### 8.7.2.2 Veri gnlklerini (log'larını) kontrol eden program komutları

#### DataLogCreate (veri gnlg oluřturma) komutu

Çizelge 8- 157 DataLogCreate komutu

LAD/FBD	SCL	Açıklama
	<pre>"DataLogCreate_DB" (     req:=_bool_in_,     records:=_udint_in_,     format:=_uint_in_,     timestamp:=_uint_in_,     done=&gt;_bool_out_,     busy=&gt;_bool_out_,     error=&gt;_bool_out_,     status=&gt;_word_out_,     name:=_string_inout_,     ID:=_dword_inout_,     header:=_variant_inout_,     data:=_variant_inout_);</pre>	<p>Bir veri log dosyası oluřturur ve bařlatır. Dosya NAME parametresi tarafından adlandırılmıř PLC\datalogs dizininde oluřturulur ve yazma iřlemleri iin dolaylı olarak aılır. alıřma zamanı iřlem verilerini programlı olarak CPU'nun flash belleğinde depolamak iin veri log komutlarını kullanabilirsiniz.</p> <p>Komutu yerleřtirdiđinizde STEP 7 otomatik olarak iliřkili kopya DB oluřturur.</p>

<sup>1</sup> SCL rneđinde, "DataLogCreate\_DB" kopya DB'nin adıdır.

Çizelge 8- 158 Parametreler iin veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Dřkten ykseđe geen (pozitif kenar) sinyal, alıřmayı bařlatır. (Varsayılan deđer: False)
RECORDS	IN	UDint	En eski giriřin zerine yazılmadan nce dngsel veri gnlgnn ierebileceđi maksimum veri kayıt sayısı: Bařlık kaydı dahil deđildir. Bařarılı bir Őekilde veri gnlg oluřturmak iin yeterli mevcut PLC ykleme belleđi var olmalıdır. (Varsayılan deđer - 1)
FORMAT	IN	UInt	Veri log formatı: <ul style="list-style-type: none"> <li>0 - Dahili format (desteklenmiyor)</li> <li>1 - Virglle ayrılmıř deđerler "csv-eng" (Varsayılan deđer)</li> </ul>
TIMESTAMP	IN	UInt	Veri zaman etiket formatı: Tarih ve saat alanları iin stun bařlıkları gerekli deđildir. Zaman etiketinde yerel saat deđil sistem saati (UTC - Koordineli Evrensel Zaman) kullanılır. <ul style="list-style-type: none"> <li>0 - Zaman etiketi yok</li> <li>1 - Tarih ve saat etiketi (Varsayılan deđer)</li> </ul>

Parametre ve tipi		Veri tipi	Açıklama
NAME	IN	Variant	Veri log adı: İsmi siz sağlarsınız. Bu değişken, sadece bir string veri tipini destekler ve yalnızca yerel, DB veya M bellekte bulunabilir. (Varsayılan değer: ' ')  String referans, ayrıca veri log dosyasının adı olarak kullanılır. İsim karakterleri, Windows dosya sistemi adlandırma kısıtlamalarına uymalıdır. \/: *? "<>   karakterlerine ve boşluk karakterine izin verilmez.
ID	In/Out	Dword	Veri kütüğü sayısal belirteç: Bu üretilen değeri diğer veri log komutları ile kullanmak için depolarsınız. ID parametresi sadece DataLogCreate komutu ile bir çıkış olarak kullanılır. (Varsayılan değer: 0)  Bu parametre için sembolik isim erişimine izin verilmez.
HEADER	In/Out	Variant	CSV dosyasında kodlanmış veri matrisinin üst satırı için sütun başlığı isimlerini işaret eder. (Varsayılan değer: null).  HEADER verisi, DB veya M bellekte yer almalıdır. Karakterler, her bir sütun adının virgül ile ayrıldığı standart CSV formatı kurallarına uymalıdır. Veri tipi bir string, bayt dizisi ya da karakter dizisi olabilir. Karakter/bayt dizileri string'lerin maksimum 255 bayt ile sınırlandırıldığı yerde artırılmış boyuta izin verir.  HEADER parametresi opsiyoneldir. HEADER atanmış değilse, o zaman hiçbir başlık satırı veri log dosyasında oluşturulmaz.
DATA	In/Out	Variant	Kayıt veri yapısını, kullanıcı tanımlı tipi (UDT) ya da diziyi işaret eder. Kayıt verisi DB veya M bellekte yer almalıdır.  DATA parametresi, veri günlük kaydının münferit veri elemanlarını (sütunlar) ve bunların veri tipini belirtir. Yapı veri tipleri, tekli yuvalama düzeyiyle sınırlıdır. Beyan edilen veri elemanlarının toplam sayısı, başlık parametresinde belirtilen sütun sayısına karşılık gelmelidir. Atadığınız veri elemanlarının maksimum sayısı 253 (zaman etiketli) veya 255'tir (zaman etiketsiz). Bu kısıtlama, kaydınızı bir Excel sayfasının 256 sütun sınırı içinde tutar.
DONE	OUT	Bool	DONE biti, son istek hatasız tamamlandıktan sonra, bir tarama için TRUE olur. (Varsayılan değer: False)
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>0 – Yapılan işlem yok</li> <li>1 – İşlem devam ediyor</li> </ul>
ERROR	OUT	Bool	Son istek hatasız tamamlandıktan sonra ERROR biti, bir tarama için TRUE olur. STATUS parametresindeki hata kodu değeri yalnızca ERROR = TRUE olan tekli tarama sırasında geçerlidir.
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)

RECORDS ve DATA parametrelerini esas alan önceden belirlenmiş sabit boyutlu bir veri günlük dosyası oluşturulur. Veri kayıtları döngüsel bir log dosyası olarak düzenlenmektedir. RECORDS parametresi tarafından belirtilen maksimum sayıda kayıt saklanana kadar yeni kayıtlar, veri log dosyasına eklenir. Yazılmış bir sonraki kayıt eski kayıt üzerine yazılacaktır. Başka bir kayıt yazma işlemi sonraki en eski veri kaydı üzerine yazma yapar ve böyle devam eder.

Bellek kaynak kullanımı:

- Veri kütükleri, yalnızca yükleme belleğini tüketir.
- Veri günlüklerinin toplam sayısı için belirlenen bir sınır yoktur. Birleştirilmiş tüm veri günlüklerinin boyutu yükleme belleğinin mevcut kaynakları ile sınırlıdır. Sadece sekiz adet veri günlüğü aynı anda açık olabilir.
- RECORDS parametresi için mümkün olan maksimum sayı, bir UDint sayısı (4,294,967,295) için sınırdır. RECORD parametresi için gerçek sınır, tek bir kayıt boyutuna, diğer veri günlükleri boyutuna ve yükleme belleğinin mevcut kaynaklarının büyüklüğüne bağlıdır. Ayrıca, Excel, Excel sayfasında izin verilen satır sayısını sınırlar.

---

**Not**

**Veri günlüğü yazma işlemi başlamadan önce veri günlüğü oluşturma yürütmesi tamamlanmalıdır**

- DataLogCreate ve DataLogNewFile günlük dosyası oluşturma işlemleri birçok program tarama çevrimleri boyunca uzar. Log dosyası oluşturulması için gerekli gerçek zaman kayıt yapısı ve kayıtların sayısına bağlıdır. Program mantığınız, DONE bitinin, bir günlük dosyası oluşturulmasının tamamlandığını sinyal veren TRUE durumuna geçişini izlemeli ve yakalamalıdır. Veri günlüğü oluşturma işlemi tamamlanmadan önce bir DataLogWrite komutu icra edilirse, o zaman yazma işlemi beklendiği gibi yeni bir veri kaydına yazmada başarısız olur.
- Bazı durumlarda çok hızlı bir program taraması çalıştığında, veri günlüğü oluşturma uzun bir zaman alabilir. Uzun oluşturma süresi çok yavaş ise, döngüsel OB'lerde minimum çevrim süresi için onay kutusunun aktif olduğundan emin olmanız gerekir ve minimum çevrim süresi, bir ms veya daha büyük olarak ayarlanır. Daha fazla bilgi için çevrim süresi ve iletişim yükünün (Sayfa 97) yapılandırılmasına bakınız.

---

**Not**

**DataLogNewFile komutu mevcut bir veri günlüğünün kayıt yapısını kopyalar**

Herhangi bir veri kaydı üzerine yazmayı önlemek istiyorsanız, mevcut veri günlüğü maksimum sayıda kayıt depoladıktan sonra, mevcut veri günlüğüne dayalı yeni bir veri günlüğü oluşturmak için DataLogNewFile komutunu kullanabilirsiniz. Yeni veri kayıtları yeni veri günlük dosyasında saklanır. Eski veri günlük dosyası ve kayıt verileri flash bellekte depolu kalır.

---

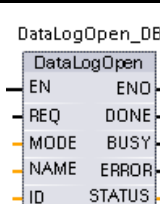


Çizelge 8- 159 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#....)	Açıklama
0	0000	Hata yok
0	7000	REQ kenarı olmayan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarı olan ilk çağrı(çalışıyor): BUSY = 1, DONE = 0
0	7002	N. çağrı (çalışıyor): BUSY = 1, DONE = 0
1	8070	Tüm dahili kopya kaydı kullanımda.
1	807F	Dahili hata
1	8090	Geçersiz dosya ismi
1	8091	İsim parametresi bir String referans değil.
1	8093	Bir veri günlüğü önceden o isimdedir. Farklı bir ad kullanınız, mevcut veri günlüğünün .csv dosyasının açık olmadığından emin olunuz ve sonra mevcut veri kaydını silmek için Web Sunucu Dosya tarayıcı sayfasını (Sayfa 622) kullanınız.
1	8097	İstenen dosya uzunluğu dosya sisteminin maksimumunu aşıyor.
1	80B3	Kullanılabilir yükleme belleği yetersiz
1	80B4	MC (Bellek kartı) yazmaya karşı korumalı.
1	80C1	Açık dosya sayısı çok fazla: Açılmış sekizden fazla veri günlük dosyasına izin verilmez.
1	8253	Geçersiz kayıt sayısı
1	8353	Geçersiz format seçimi
1	8453	Geçersiz zaman etiketi seçimi
1	8B24	Geçersiz HEADER alan ataması: Örneğin, yerel belleği işaret ediniz
1	8B51	Geçersiz HEADER parametresi veri tipi
1	8B52	Çok fazla HEADER parametresi veri elemanı
1	8C24	Geçersiz DATA alan ataması: Örneğin, yerel hafızayı işaret ediniz
1	8C51	Geçersiz DATA parametresi veri tipi
1	8C52	Çok fazla DATA parametresi veri elemanı

## DataLogOpen (veri günlüğü açma) komutu

Çizelge 8- 160 DataLogOpen komutu

LAD / FBD	SCL	Açıklama
 <p>DataLogOpen_DB</p> <p>DataLogOpen</p> <p>EN ENO</p> <p>REQ DONE</p> <p>MODE BUSY</p> <p>NAME ERROR</p> <p>ID STATUS</p>	<pre>"DataLogOpen_DB" (     req:=_bool_in_,     mode:=_uint_in_,     done=&gt;_bool_out_,     busy=&gt;_bool_out_,     error=&gt;_bool_out_,     status=&gt;_word_out_,     name:=_string_inout_,     ID:=_dword_inout_);</pre>	<p>Önceden var olan bir veri günlük dosyası açar. Günlüğe yeni kayıtları yazabilmeden önce bir veri günlüğü açılmalıdır. Veri günlükleri münferit olarak açılabilir ve kapatılabilir. Maksimum sekiz adet veri günlüğü aynı anda açık olabilir.</p> <p>Komutu yerleştirdiğinizde STEP 7 otomatik olarak ilişkili kopya DB oluşturur.</p>

<sup>2</sup> SCL örneğinde, "DataLogOpen\_DB" kopya DB'nin ismidir.

## Geniřletilmiř komutlar

### 8.7 Tarifler ve veri gnlkleri

Çizelge 8- 161 Parametreler iin veri tipleri

Parametre ve tipi		Veri tipi	Aıklama
REQ	IN	Bool	Dřkten ykseęe (pozitif kenar) sinyali, iřlemi bařlatır. (Varsayılan deęer: False)
MODE	IN	UInt	İřlem modu: <ul style="list-style-type: none"> <li>0 - Mevcut verilere eklenecek (Varsayılan deęer)</li> <li>1 - Tm mevcut kayıtları temizler</li> </ul>
NAME	IN	Variant	Mevcut bir veri gnlę adı: Bu varyant sadece bir string veri tipini destekler ve yalnızca yerel, DB veya M bellekte yer alabilir. (Varsayılan deęer: '')
ID	In/Out	Dword	Veri gnlęnn sayısal belirteci. (Varsayılan deęer: 0) <b>Not:</b> Bu parametre iin Sembolik isim eriřimine izin verilmez.
DONE	OUT	Bool	Son istek hatasız tamamlandıktan sonra DONE biti, bir tarama iin TRUE olur. (Varsayılan deęer: False)
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>0 - Yapılan iřlem yok</li> <li>1 - İřlem devam ediyor</li> </ul>
ERROR	OUT	Bool	Son istek hatasız tamamlandıktan sonra ERROR biti, bir tarama iin TRUE olur. STATUS parametresindeki hata kodu deęeri yalnızca ERROR = TRUE olan tekli tarama sırasında geerli olur.
STATUS	OUT	Word	Yrtme durum kodu (Varsayılan deęer: 0)

nceden var olan bir veri gnlęyle ilgili NAME veya bir ID (bir girdi olarak ID parametresi) saęlayabilirsiniz. Her iki parametreyi saęlarsanız ve geerli bir ID, NAME veri gnlęne karřılık gelmezse, o zaman ID kullanılır ve NAME gz ardı edilir.

NAME, DataLogCreate komutu tarafından oluřturulan bir veri gnlę adı olmalıdır. Sadece NAME saęlanırsa ve NAME geerli bir veri gnlę belirtirse, karřılık gelen ID (bir ıkıř olarak ID parametresi) dndrlr.

#### Not

##### Veri log dosyalarının genel kullanımı

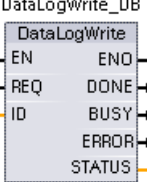
- Veri log dosyaları DataLogCreate ve DataLogNewFile iřlemlerinden sonra otomatik olarak aılır.
- PI C'nin alıřmadan durmava deęiři veya bir PLC g evriminden sonra Veri log dosyaları otomatik olarak kapatılır.
- Yeni DataLogWrite iřleminin mmkn olmasından nce veri log dosyasının aık olması gerekir.
- Aynı anda maksimum sekiz adet veri log dosyası aık olabilir. Sekizden fazla veri log dosyası olabilir, ancak bazılarının kapalı olması gerektięinden en fazla sekizi aıktır.

Çizelge 8- 162 ERROR ve STATUS deęerleri

ERROR	STATUS (W#16#)	Açıklama
0	0000	Hata yok
0	0002	Uyarı: Veri gnlk dosyası, bu uygulama programı tarafından zaten açık
0	7000	REQ kenarı olmayan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarı olan ilk çağrı(çalışıyor): BUSY = 1, DONE = 0
0	7002	N. çağrı (çalışıyor): BUSY = 1, DONE = 0
1	8070	Tm dahili kopya bellek kullanımdadır.
1	8090	Veri gnlę tanımı mevcut veri gnlk dosyası ile uyumlu deęil.
1	8091	Name parametresi String referans deęil.
1	8092	Veri gnlę mevcut deęil.
1	80C0	Veri gnlk dosyası kilitle.
1	80C1	Açık dosya sayısı çok fazla: Açılmış sekizden fazla veri gnlk dosyasına izin verilmez.

## DataLogWrite (veri gnlę yazma) komutu

Çizelge 8- 163 DataLogWrite komutu

LAD / FBD	SCL	Açıklama
	<pre>"DataLogWrite_DB" (     req:=_bool_in_,     done=&gt;_bool_out_,     busy=&gt;_bool_out_,     error=&gt;_bool_out_,     status=&gt;_word_out_,     ID:=_dword_inout_);</pre>	<p>Belirtilen veri gnlęne bir veri kaydı yazar. Bir DataLogWrite işlemine izin verilmeden önce daha önceden var olan hedef veri gnlę açık olmalıdır.</p> <p>Komutu yerleřtirdiğinizde STEP 7 otomatik olarak iliřkili kopya DB oluşturur.</p>

<sup>2</sup> SCL örneğinde, "DataLogWrite\_DB" kopya DB'nin ismidir.

Çizelge 8- 164 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
REQ	IN	Bool Dřkten ykseęe (pozitif kenar) sinyali, işlemi bařlatır. (Varsayılan deęer: False)
ID	In/Out	Dword Veri gnlęnn sayısal belirteci. Yalnızca DataLogWrite komutu için bir giriş olarak kullanılır. (Varsayılan deęer: 0) <b>Not:</b> Bu parametre için Sembolik isim erişimine izin verilmez.
DONE	OUT	Bool Son istek hatasız tamamlandıktan sonra DONE biti, bir tarama için TRUE olur.
BUSY	OUT	Bool <ul style="list-style-type: none"> <li>0 - Yapılan işlem yok</li> <li>1 - İşlem devam ediyor</li> </ul>

## Genişletilmiş komutlar

### 8.7 Tarifler ve veri günlükleri

Parametre ve tipi		Veri tipi	Açıklama
ERROR	OUT	Bool	Son istek hatasız tamamlandıktan sonra ERROR biti, bir tarama için de TRUE olur. STATUS parametresindeki hata kodu değeri yalnızca ERROR = TRUE olan tekli tarama sırasında geçerli olur.
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)

Ara belleğin rekor bellek adresi ve veri yapısı bir DataLogCreate komutunun DATA parametresi tarafından yapılandırılır. Programlı olarak mevcut çalışma zamanı işlem değerleri ile kayıt ara belleğini yüklemeniz ve daha sonra yeni kayıt verisini ara bellekten veri günlüğüne taşımak için DataLogWrite komutunu çalıştırmanız gerekir.

ID parametresi bir veri günlüğü ve veri kaydı yapılandırmasını tanımlar. Veri günlüğü oluşturulduğunda ID numarası üretilir.

Döngüsel veri günlük dosyasında boş kayıtlar varsa, o zaman bir sonraki boş kayıt yazılır. Tüm kayıtlar doluyorsa, eski kayıt üzerine yazılır.

#### DİKKAT

##### Veri günlük yazma işlemi başlamadan önce veri günlüğü oluşturma işlemleri tamamlanmalıdır

DataLogCreate ve DataLogNewFile günlük dosyası oluşturma işlemleri birçok program tarama çevrimleri boyunca uzanır. Log dosyası oluşturulması için gerekli gerçek zaman, kayıt yapısı ve kayıtların sayısına bağlıdır. Program mantığınız, DONE bitinin, bir günlük dosyası oluşturulmasının tamamlandığını sinyal veren TRUE durumuna geçişini izlemeli ve yakalamalıdır. Veri günlüğü oluşturma işlemi tamamlanmadan önce bir DataLogWrite komutu yürütülürse, o zaman yazma işlemi beklendiği gibi yeni bir veri kaydına yazmada başarısız olur.

#### DİKKAT

##### CPU güç kesintisi sırasında veri günlüğündeki veriler kaybolabilir

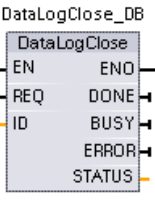
Tamamlanmamış DataLogWrite işlemi sırasında bir elektrik kesintisi olması durumunda, veri günlüğüne aktarılan veri kaydı kaybolabilir.

Çizelge 8- 165 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#)	Açıklama
0	0000	Hata yok
0	0001	Veri günlüğünün dolu olduğunu gösterir: Her bir veri günlüğü, kayıtların belirlenen maksimum sayısı ile oluşturulur. Maksimum sayının son kaydı yazılmıştır. Sonraki yazma işlemi, en eski kayıt üzerine yazacaktır.
0	7000	REQ kenarı olmayan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarı olan ilk çağrı (çalışıyor): BUSY = 1, DONE = 0
0	7002	N. çağrı (çalışıyor): BUSY = 1, DONE = 0
1	8070	Tüm dahili kopya kaydı kullanımda.
1	8092	Veri günlüğü mevcut değil.
1	80B0	Veri günlük dosyası açık değil (Yalnızca belirgin açma modu için).

## DataLogClose (Veri günlüğünü kapatma) komutu

Çizelge 8- 166 DataLogClose komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a function block named 'DataLogClose_DB'. It has an 'EN' (Enable) input and an 'END' (End) output. It has three outputs: 'REQ' (Boolean), 'DONE' (Boolean), 'BUSY' (Boolean), 'ID' (Dword), 'ERROR' (Boolean), and 'STATUS' (Word).</p>	<pre>"DataLogClose_DB" (   req:=_bool_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_word_out_,   ID:=_dword_inout_);</pre>	<p>Açık bir veri günlük dosyasını kapatır. Kapalı bir veri günlüğü için DataLogWrite işlemleri sonucunda bir hata oluşur. Başka bir DataLogOpen işlemi gerçekleştirilene kadar bu veri günlüğüne hiçbir yazma işlemi için izin verilmez.</p> <p>STOP moduna geçiş, tüm açık veri log dosyalarını kapatacaktır.</p> <p>Komutu yerleştirdiğinizde STEP 7 otomatik olarak ilişkili kopya DB oluşturur.</p>

<sup>2</sup> SCL örneğinde, "DataLogClose\_DB" kopya DB'nin ismidir.

Çizelge 8- 167 Parametreler için veri tipleri

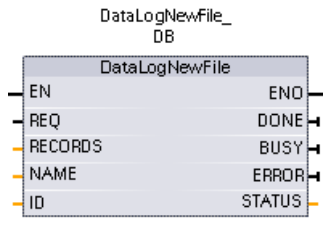
Parametre ve tipi	Veri tipi	Açıklama
REQ IN	Bool	Düşükten yükseğe geçen (pozitif kenar) sinyal çalışmayı başlatır. (Varsayılan değer: False)
ID In/Out	Dword	Veri günlüğünün sayısal belirteci. Yalnızca DataLogClose komutu için bir giriş olarak kullanılır. (Varsayılan değer: 0) <b>Not:</b> Bu parametre için Sembolik isim erişimine izin verilmez.
DONE OUT	Bool	Son istek hatasız tamamlandıktan sonra DONE biti, bir tarama için TRUE olur.
BUSY OUT	Bool	<ul style="list-style-type: none"> <li>0 - Yapılan işlem yok</li> <li>1- İşlem devam ediyor</li> </ul>
ERROR OUT	Bool	Son istek hatasız tamamlandıktan sonra ERROR biti, bir tarama için TRUE olur. STATUS parametresindeki hata kodu değeri yalnızca ERROR = TRUE olan tekli tarama sırasında geçerli olur.
STATUS OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)

Çizelge 8- 168 ERROR ve STATUS değerleri

ERROR	STATUS (W#16#)	Açıklama
0	0000	Hata yok
0	0001	Veri günlüğü açık değil
0	7000	REQ kenarı olmayan çağrı: BUSY = 0, DONE = 0
0	7001	REQ kenarı olan ilk çağrı(çalışıyor): BUSY = 1, DONE = 0
0	7002	N. çağrı (çalışıyor): BUSY = 1, DONE = 0
1	8092	Veri günlüğü mevcut değil.

## DataLogNewFile (yeni dosyada veri gnlg) komutu

Çizelge 8- 169 DataLogNewFile komutu

LAD / FBD	SCL	Açıklama
	<pre>"DataLogNewFile_DB" (   req:= _bool_in_,   records:= _udint_in_,   done=&gt; _bool_out_,   busy=&gt; _bool_out_,   error=&gt; _bool_out_,   status=&gt; _word_out_,   name:= _DataLog_out_,   ID:= _dword_inout_ );</pre>	<p>Programınızın mevcut veri gnlk dosyasına dayalı yeni bir veri gnlk dosyası oluřturmasına olanak saęlar.</p> <p>Komutu yerleřtirdiğinizde STEP 7 otomatik olarak iliřkili kopya DB oluřturur.</p>

<sup>2</sup> SCL örneğinde, "DataLogNewFile\_DB" kopya DB'nin ismidir.

Çizelge 8- 170 Parametreler için veri tipleri

Parametre ve tipi	Ver tipi	Açıklama
REQ	IN	Bool Dřikten ykseęe geen (pozitif kenar) sinyal alıřmayı bařlatır. (Varsayılan deęer: False)
RECORDS	IN	UDInt En eski giriřin zerine yazılmadan nce dngsel veri gnlgnn ierebileceęi maksimum veri kayıt sayısı. (Varsayılan deęer: 1) Bařlık kaydı dahil deęildir. Bařarılı bir Őekilde veri gnlg oluřturmak iin yeterli mevcut PLC ykleme belleęi var olmalıdır.
NAME	IN	Variant Bir veri gnlg adı: Bu varyant sadece bir string veri tipini destekler ve yalnızca yerel, DB veya M bellekte yer alabilir. (Varsayılan deęer: ' ') String referans, ayrıca veri log dosyasının adı olarak kullanılır. İsim karakterleri, Windows dosya sistemi adlandırma kısıtlamalarına uymalıdır. \ /: *? "<>   karakterlerine ve bořluk karakterine izin verilmez.
ID	In/Out	Dword Veri gnlg sayısal belirteci (Varsayılan deęer: 0): <ul style="list-style-type: none"> <li>Yrtmede, ID giriř, geerli bir veri gnlg tanımlar. Yeni veri gnlg yapılandırması bu veri kaydından kopyalanır.</li> <li>Yrtme sonrasında, ID parametresi yeni oluřturulan veri gnlk dosyasının ID'sini dndren bir ıkıř olur.</li> </ul> <b>Not:</b> Bu parametre iin sembolik isim eriřimine izin verilmez.
DONE	OUT	Bool DONE biti, son istek hatasız tamamlandıktan sonra, bir tarama iin TRUE olur.
BUSY	OUT	Bool <ul style="list-style-type: none"> <li>0 - Yapılan iřlem yok</li> <li>1 - İřlem devam ediyor</li> </ul>
ERROR	OUT	Bool Son istek hatasız tamamlandıktan sonra ERROR biti, bir tarama iin TRUE olur. STATUS parametresindeki hata kodu deęeri yalnızca ERROR = TRUE olan tekli tarama sırasında geerli.
STATUS	OUT	Word Yrtme durum kodu (Varsayılan deęer: 0)

Bir veri gnlg dolduđu veya tamamlanmış olduđu kabul edildiđinde ve veri gnlgnde depolanan tm verileri kaybetmek istemediđinizde DataLogNewFile komutunu yrtebilirsiniz. Yeni boş bir veri gnlk dosyası dolu veri gnlk dosya yapısına dayalı olarak oluřturulabilir . Bařlık kaydı, orijinal veri gnlg özelliklerine ( DATA kayıt ara belleđi, veri formatı ve zaman etiketi ayarları) sahip orijinal veri gnlgnden çođaltılacaktır. Orijinal Veri gnlk dosyası, dolaylı olarak kapatılır ve yeni veri gnlk dosyası dolaylı olarak aılır.

DataLogWrite parametre tetikleyici: Programınız her DataLogWrite iřleminin ERROR ve STATUS parametrelerini izlemesi gerekir. Son kayıt yazıldıđında ve veri gnlg dolduđunda, DataLogWrite ERROR biti = 1 ve DataLogWrite STATUS word = 1 olur. Bu ERROR ve STATUS deđerleri sadece bir tarama iin geerlidir. Bylece izleme mantıđınız STATUS deđerini yakalamak iin zaman kapısı olarak ERROR = 1 kullanması gerekir ve sonra STATUS = 1 iin test edilmelidir (veri gnlg dolu ).

DataLogNewFile iřlemi: veri gnlgn alan program mantıđınız dolu sinyal olduđunda bu durum bir DataLogNewFile iřlemini etkinleřtirmek iin kullanılır. Mevcut (genellikle dolu) ve aık veri gnlgnn ID'sine sahip ancak yeni bir benzersiz NAME parametresiyle DataLogNewFile komutu yrtlmelidir. DataLogNewFile iřlemi yapıldıktan sonra, yeni veri gnlg ismine karřılık gelen yeni bir veri gnlk ID deđeri dndrlr ( bir ıkıř parametresi olarak ). Yeni veri gnlk dosyası dolaylı olarak aılır ve yeni kayıtları saklamak iin hazırdır. Yeni veri gnlk dosyasına ynlendirilen yeni DataLogWrite iřlemleri, DataLogNewFile iřlemi tarafından dndrlen ID deđerini kullanmanız gerekir.

**DİKKAT****Veri gnlk yazma iřlemi bařlamadan nce veri gnlg oluřturma iřlemleri tamamlanmalıdır**

DataLogCreate ve DataLogNewFile gnlk dosyası oluřturma iřlemleri, birok program tarama evrimleri boyunca devam eder. Log dosyası oluřturulması iin gerekli gerek zaman, kayıt yapısı ve kayıtların sayısına bađlıdır. Program mantıđınız, DONE bitinin, bir gnlk dosyası oluřturulmasının tamamlandıđını sinyal veren TRUE durumuna geiřini izlemeli ve yakalamalıdır. Veri gnlg oluřturma iřlemi tamamlanmadan nce bir DataLogWrite komutu icra edilirse, o zaman yazma iřlemi beklendiđi gibi yeni bir veri kaydına yazmada bařarısız olur.

izelge 8- 171 ERROR ve STATUS deđerleri

ERROR	STATUS (W#16#)	Aıklama
0	0000	Hata yok
0	7000	REQ kenarı olmayan ađrı: BUSY = 0, DONE = 0
0	7001	REQ kenarı olan ilk ađrı (alıřıyor): BUSY = 1, DONE = 0
0	7002	N. ađrı (alıřıyor): BUSY = 1, DONE = 0
1	8070	Tm dahili kopya belleđi kullanımda.
1	8090	Geersiz dosya ismi
1	8091	İsim parametresi String referans deđil.
1	8092	Veri gnlg mevcut deđil.
1	8093	Veri gnlg zaten var.
1	8097	İstenen dosya uzunluđu dosya sisteminin maksimumunu ařıyor.
1	80B3	Kullanılabilir ykleme belleđi yetersiz.

ERROR	STATUS (W#16#)	Açıklama
1	80B4	MC yazmaya karşı korumalı.
1	80C1	Açık dosya sayısı çok fazla.

### 8.7.2.3 Veri günlükleriyle çalışma

Veri günlük dosyaları kalıcı flash bellekte, virgülle ayrılmış değer biçimi (\*.csv) olarak saklanır. PLC web sunucu özelliğini kullanarak veya PLC bellek kartını çıkarmadan bunu standart bir PC kart okuyucuya takarak veri günlüklerini görüntüleyebilirsiniz.

### PLC Web sunucu özelliği ile veri günlükleri görüntüleme

PLC PROFINET portu ve PC bir ağa bağlı olduğunda dahili PLC Web sunucusuna erişmek için Microsoft Internet Explorer veya Mozilla Firefox gibi bir PC web tarayıcısı kullanabilirsiniz. PLC web sunucusunu çalıştırdığınızda PLC çalışma modunda veya durma modunda olabilir. PLC çalışma modunda ise, PLC web sunucusu ağ üzerinden log verisini transfer ederken kontrol programınız çalışmaya devam eder.

Web sunucusuna erişim:

5. Hedef CPU (Sayfa 603) için Cihaz Yapılandırmasındaki Web sunucusunu etkinleştiriniz.
6. Bilgisayarınızı PROFINET ağ (Sayfa 606) üzerinden PLC'ye bağlayınız.
7. Gömülü Web sunucusu (Sayfa 609) üzerinden CPU'ya erişiniz.
8. "File Browser" standart Web sayfası (Sayfa 622) ile veri log dosyalarını indiriniz, düzenleyiniz ve siliniz.
9. Microsoft Excel gibi bir elektronik Çizelge uygulaması ile .csv dosyasını açınız.

### PLC bellek kartındaki veri günlüklerini görüntüleme

S7-1200 CPU, takılı bir "Program" tipi S7-1200 bellek kartına sahipse, o zaman bellek kartını çıkarınız ve bu kartı bir PC veya PG üzerindeki standart SD (Dijital olarak güvenli) veya MMC (Multi medya kartı) kart yuvasına yerleştiriniz. PLC stop modundayken bellek kartı çıkarıldığında kontrol programınız yürütülmez.

Windows dosya gezginini kullanınız ve bellek kartında \DataLog dizinine gidin. Tüm \\*.csv veri log dosyalarınız bu dizinde bulunur.



Veri gnlk dosyalarının bir kopyasını alınız ve bunu PC'nizin yerel srcsne koyunuz. Sonra, bir \*.csv dosyasının bellek kartında saklanan orijinal dosyasını deęil yerel bir kopyasını amak iin Excel'i kullanabilirsiniz.

**DİKKAT**

**Bir PC kart okuyucu kullanarak S7-1200 bellek kartındaki veri log dosyalarını kopyalayabilirsiniz. Ancak bu dosyaları deęiřtirmeyiniz veya silmeyiniz.**

Standart Web sunucusu Dosya Tarayıcı sayfası log dosyalarını grntleme, indirme (kopyalama) ve silmek iin nerilen aratır.

Windows gezgini tarafından bellek kartı dosya sistemini doęrudan tarama iřlemi risk ierir. nk veri gnlęn veya bir dosyayı bozabilen ya da bellek kartını kullanılamaz hale getiren dięer sistem dosyalarını yanlıřlıkla silebilir/deęiřtirebilirsiniz.

**Web tarayıcısından veri gnlklerinin grntlenmesi**

Web sunucu zellięini kullanmazsanız bile, veri kayıtlarını Internet Explorer veya Mozilla Firefox gibi bir Web tarayıcısından doęrudan grebilirsiniz. CPU IP adresini ve "MyDataLog" yerine STEP 7'de saęlanan veri gnlk dosyasının gerek adını kullanarak tarayıcınızın adres ubuęuna sadece ařaęıdaki metni giriniz:

**<http://192.168.0.1/FileBrowser/Download?Path=/DataLogs/MyDataLog.csv>**

Veri gnlk dosyalarının sabit adresleri, nc parti dosya toplama araları vasıtasıyla onlara eriřimi mmkn kılar.

**8.7.2.4 Veri gnlk dosyalarının boyutunu sınırlama**

Veri gnlk dosyaları, program, program verileri, yapılandırma verileri, kullanıcı-tanımlı Web sayfaları ve PLC sistem verileri ile PLC ykleme bellek alanını paylařır. Dahili ykleme belleęini kullanan byk boyutlu bir program byk kapasitede ykleme belleęi gerektirir. Veri gnlk dosyaları iin yeterli boř alan olmayabilir. Bu durumda, ykleme bellek boyutunu artırmak iin bir "Program kart" kullanabilirsiniz. S7-1200 CPU'lar tek seferde her ikisi olmamak zere dahili veya harici ykleme belleęini kullanabilir.

Bir "Program" kartı oluřturma hakkında ayrıntılar iin bellek kartı blmne bakınız (Sayfa 130).

### Veri günlük dosyaları için maksimum boyut kuralı

Bir veri günlük dosyasının maksimum boyutu, serbest yükleme bellek boyutunu veya 500 MB'tan küçük olanını geçemez. Program birden fazla veri günlük dosyası kullanıyorsa, o zaman tüm veri günlük dosyalarının toplam boyutu serbest yükleme bellek boyutunu aşamaz.

Çizelge 8- 172 Yükleme bellek boyutu

Veri alanı	CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C, CPU 1217C	Veri depolama
<b>Dahili yükleme belleği</b> flash bellek	1 MB	1 MB	4 MB	4 MB	Kullanıcı programı ve program verileri, yapılandırma verileri, Veri günlükleri, kullanıcı-tanımlı Web sayfaları ve PLC sistem verileri
<b>Harici yükleme belleği</b> Opsiyonel "Program kart" flash bellek	4 MB, 12 MB, 24 MB, 256 MB, 2 GB veya 32 GB SD kart boyutuna bağlı olarak				

### Boş yükleme bellek alanını belirleme

Boş yükleme belleği alan miktarı, normal işlemler sırasında, işletim sisteminin belleği kullanması ve serbest bırakmasına göre değişir. Yükleme belleğinin bellek boyutunu görmek için aşağıdaki basamakları kullanınız.

1. STEP 7 ile hedef S7-1200 PLC arasında bir çevrimiçi bağlantı kurunuz.
2. Veri günlük işlemlerini kontrol eden programı indiriniz.
3. Size gereken herhangi bir opsiyonel kullanıcı-tanımlı Web sayfaları oluşturunuz. Veri günlüklerine erişen standart Web sayfaları PLC cihaz yazılımında saklanır ve yükleme belleğini kullanmaz.
4. Toplam yükleme bellek boyutunu ve boş alanı (Sayfa 783) görmek için Çevrimiçi ve Tanılama araçları kullanınız.

### Veri günlük dosyasının boyutunun hesaplanması (tüm veri kayıtları)

Veri günlük dosyası oluşturulduğunda maksimum bellek boyutu tahsis edilir. Tüm veri kayıtları için gerekli boyuta ek olarak, veri günlük başlığı (eğer kullanılıyorsa), zaman etiketi başlığı (eğer kullanılıyorsa), kayıt indeks başlığı ve bellek tahsisine yönelik minimum blok boyutu için depolama alanını dahil etmeniz gerekir.

Veri günlük dosyalarınızın boyutunu belirlemek için aşağıdaki formülü kullanınız ve maksimum boyut kuralını ihlal etmediğinizden emin olunuz.

Veri günlük dosyası boyutu = başlık baytı + veri baytı + 512 bayt

## Başlık

Veri günlüğü başlık baytı = başlık karakteri baytı + 2 bayt

### Başlık karakter baytları

- Veri başlığı yok ve zaman etiketleri yok = 7 bayt
- Veri başlığı yok ve zaman etiketleri var (bir zaman etiketi başlığına sahip) = 21 bayt
- Veri başlıkları var ve zaman etiketleri yok = ayırıcı virgüller dahil tüm sütun başlığı metnindeki karakter baytların sayısı
- Veri başlıkları var ve zaman etiketleri var (bir zaman etiketi başlığına sahip) = ayırıcı virgüller dahil tüm sütun başlığı metnindeki karakter baytların sayısı + 21 bayt

## Veri

Veri günlüğü veri baytları = ((bir kayıttaki veri baytları + zaman etiketi baytları) \* kayıt sayısı) + 12 bayt

### Bir veri kaydındaki veri baytları

DataLogCreate DATA parametresi, veri alanlarının sayısını ve bir veri günlük kaydı için her veri alanının veri tipini atayan bir yapıyı işaret eder.

Gerekten bayt sayısı, belirli bir veri tipine ait olaylar sayısını çarpınız. Bir kayıttaki her veri tipi için işlemi tekrarlayınız ve bir kayıttaki tüm veri elemanlarının toplamını almak için tüm veri baytlarını toplayınız.

### Münferit veri elemanlarının boyutu

Log verisi, CSV dosya formatındaki (virgülle ayrılmış değerler) karakter baytlar olarak saklanır. Aşağıdaki Çizelge, her bir veri ögesini saklamak için gerekli olan bayt sayısını göstermektedir.

Veri tipi	Bayt sayısı (veri + bir virgül baytı dahil)
Bool	2
Byte	5
Word	7
Dword	12
Char	4
String	<b>Örnek 1:</b> MyString String[10] Maksimum string boyutu 10 karakter olarak atanır. Metin karakterleri + Boş karakterlere sahip otomatik doldurma = 10 bayt Açılış ve kapanış çift tırnağı + virgül karakterleri = 3 bayt 10 + 3 = 13 toplam bayt <b>Örnek 2:</b> Mystring2 String Köşeli parantez ile hiçbir boyut atanmazsa, daha sonra 254 bayt varsayılan olarak tahsis edilir. Metin karakterler + Boş karakterlere sahip otomatik doldurma = 254 bayt Açılış ve kapanış çift tırnağı + virgül karakterleri = 3 bayt 254 + 3 = 257 toplam bayt
USInt	5

UInt	7
UDInt	12
SInt	5
Int	7
DInt	12
Real	16
LReal	25
Time	15
DTL	24

**Veri gnlk dosyasındaki kayıtların sayısı**

DataLogCreate komutunun RECORDS parametresi veri gnlk dosyasındaki kayıtların maksimum sayısını ayarlar.

**Bir veri kaydındaki zaman etiketi baytları**

- Zaman etiketi yok = 0 bayt
- Zaman etiketi = 20 bayt

**Ayrıca bakınız**

evrim sresi ve bellek kullanımını izleme (Sayfa 792)

SIMATIC bellek kartları (Sayfa 978)

**8.7.2.5 Veri gnlg rnek program**

Bu veri gnlg rnek programı, dinamik bir sreten rnek deęerleri almak iin gerekli tm program mantıęını gstermez, ancak Veri gnlk komutlarının anahtar iřlemlerini gsterir. Kullandığınız gnlk dosyalarının yapısı ve sayısı, proses kontrol gereksinimlerine baęlıdır.

**Not****Veri log dosyalarının genel kullanımı**

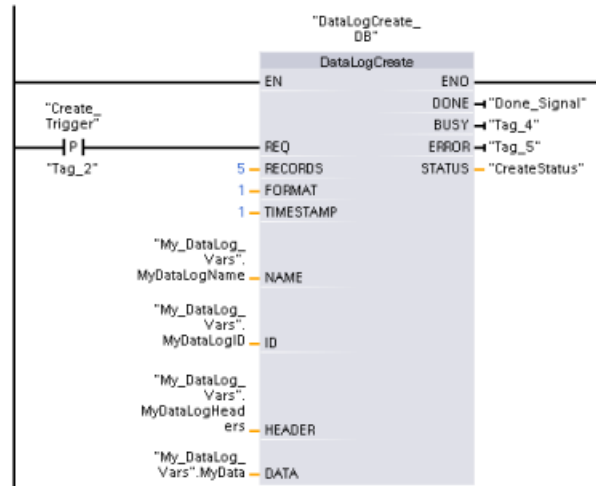
- Veri log dosyaları DataLogCreate ve DataLogNewFile iřlemlerinden sonra otomatik olarak aılır.
- PI C'nin alıřmadan durmava deęiři veya bir PLC g evriminden sonra Veri log dosyaları otomatik olarak kapatılır.
- Yeni DataLogWrite iřleminin mmkn olmasından nce veri log dosyasının aık olması gerekir.
- Aynı anda maksimum sekiz adet veri log dosyası aık olabilir. Sekizden fazla veri log dosyası olabilir, ancak bazılarının kapalı olması gerektięinden en fazla sekizi aıktır.

## Örnek veri günlüğü programı

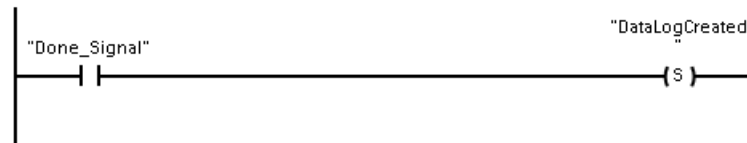
Örnek veri günlük isimleri, başlık metni ve MvData varısı, veri bloğunda oluşturulur. Üç adet MvData değişkeni öncici olarak yeni örnek değerleri depolar. Bu DB konumlarındaki proses örnek değerleri, DataLogWrite komutu yürütülerek bir veri günlük dosyasına aktarılır.

My_Datalog_Vars			
	Name	Data type	Start value
1	Static		
2	MyNewDataLogName	String	'MyNEWDatLog'
3	MyDataLogName	String	'MyDataLog'
4	MyDataLogID	DWord	0
5	MyDataLogHeaders	String	'Count, Temperature, Pressure'
6	MyData	Struct	
7	MyCount	Int	0
8	MyTemperature	Real	0.0
9	MyPressure	Real	0.0

Devre 1 REQ pozitif kenar, veri günlüğü oluşturma prosesini başlatır.



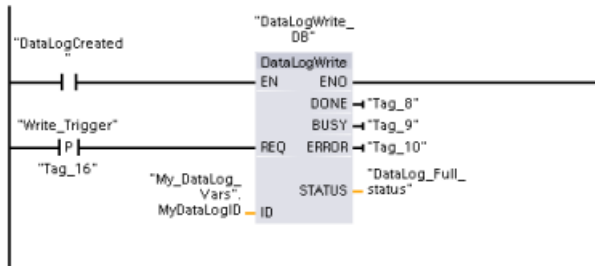
Devre 2 DataLogCreate'ten gelen DONE çıkışını yakalar, çünkü bu sadece bir tarama için geçerlidir.



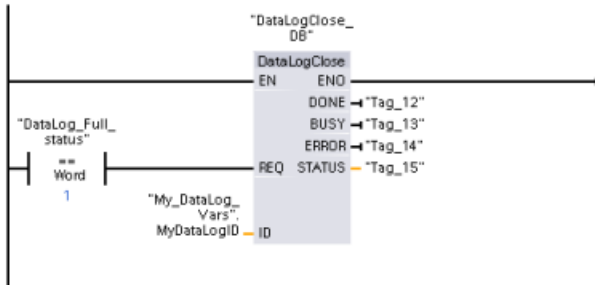
**Devre 3** MyData yapısında yeni proses deęerleri depolanacaęı zaman pozitif bir kenar sinyali tetiklenir.



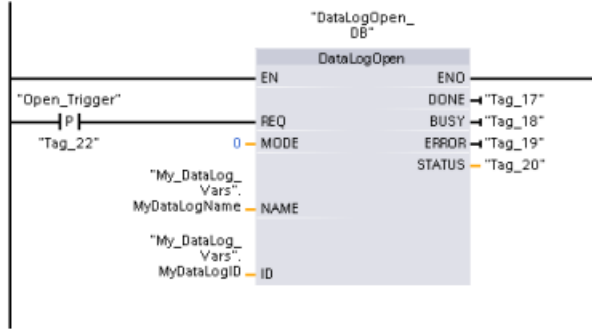
**Devre 4** EN giriř durumu DataLogCreate iřlemi tamamlandıęında esas alınır. Bir oluřturma iřlemi, birok tarama dngleri boyunca devam eder ve bir yazma iřlemini alıřtırmadan nce tamamlanmış olmalıdır. REQ giriřindeki pozitif kenar sinyali, etkinleřtirilmiş bir yazma iřlemini tetikleyen bir olaydır.



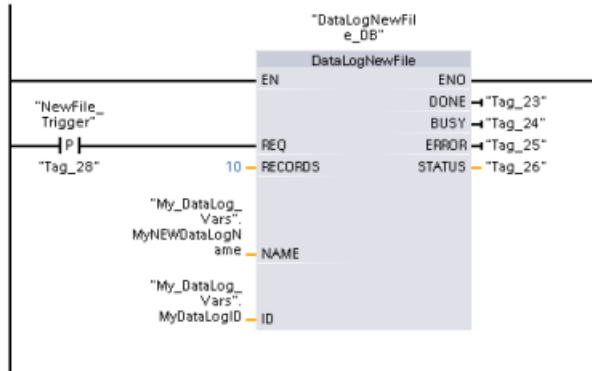
**Devre 5** Son kavıt vazıldıktan sonra veri gnllğini kapatınız. DataLogWrite STATUS cıkıs = 1 olduęunda, son kavdı vazan DataLogWrite iřlemi alıřtırıldıktan sonra, gnlk dosyasının dolu durumuyla ilgili sinyal verilir.



**Devre 6** Pozitif sinval kenarı DataLogOpen RFO dirisi. veri aünlük dosvası acan bir HMI üzerindeki bir düömeve basarak kullanıcıyı simüle eder. Proses verileri ile dolu olan tüm kavıtlara sahip bir veri loo dosvası açarsanız. o zaman bir sonraki DataLogWrite işlemi eski kavıt üzerine vazacaktır. Devre 7'de dösterildiđi gibi, eski veri günlüđünü korumak ve yerine yeni bir veri günlüđü oluşturmak isteyebilirsiniz.



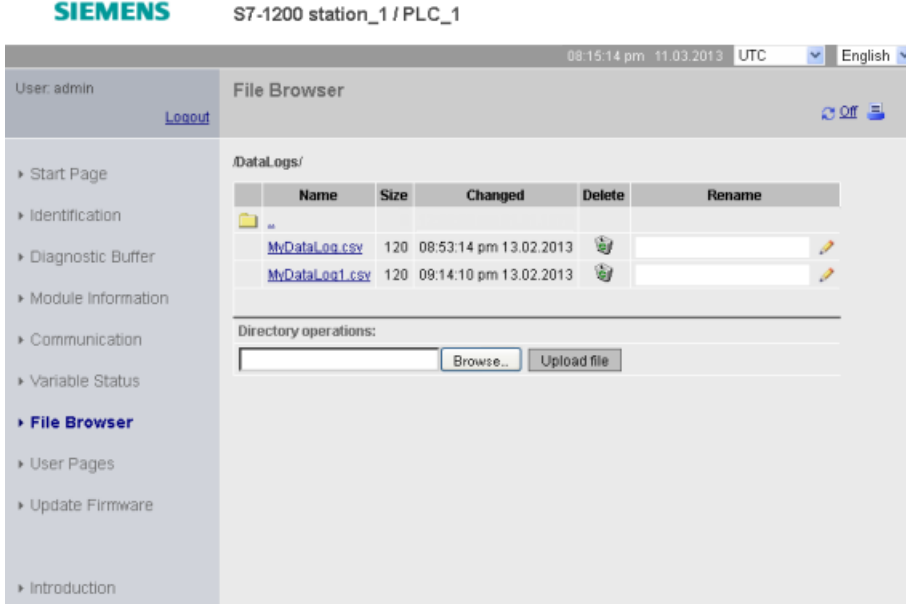
**Devre 7** ID parametresi bir IN / OUT tipidir. Öncelikle, yapısını kopyalamak istediđiniz mevcut veri günlüđünün ID deđerini sađlarsınız. DataLogNewFile işlemi tamamlandıktan sonra, yeni veri günlüđü için yeni ve benzersiz bir ID deđerini, ID referans konumuna geri yazılır. Gerekli DONE biti = TRUE yakalaması gösterilmemiştir, DONE bit mantıđı örneđi için devre 1, 2 ve 4'e bakınız.



## Geniřletilmiř komutlar

### 8.7 Tarifler ve veri gnlkleri

#### S7-1200 CPU Web sunucusu ile grntlenen rnek program tarafından oluřturulan veri log dosyaları



- ① Modify ayrıcalıkları ile oturum amadıysanız "Delete" seeneęi mevcut olmaz.
- ② Modify ayrıcalıkları ile oturum amadıysanız "Rename" seeneęi mevcut olmaz.

izelge 8- 173 Excel ile grntlenen indirilmiř .csv dosyası rnekleri

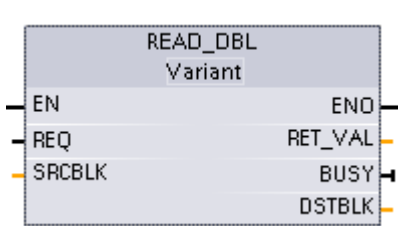
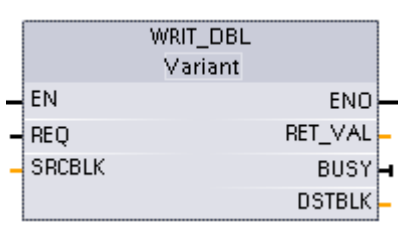
Maksimum beř kayıtlı dosyaya yazılmıř iki kayıt	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>UTC Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>1</td> <td>9/29/2010</td> <td>21:01:46</td> <td>5</td> <td>5.00E+00</td> <td>5.00E+00</td> </tr> <tr> <td>3</td> <td>2</td> <td>9/29/2010</td> <td>21:01:47</td> <td>5</td> <td>5.00E+00</td> <td>5.00E+00</td> </tr> <tr> <td>4</td> <td colspan="6">//END</td> </tr> <tr> <td>5</td> <td colspan="6"></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	UTC Time	Count	Temperature	Pressure	2	1	9/29/2010	21:01:46	5	5.00E+00	5.00E+00	3	2	9/29/2010	21:01:47	5	5.00E+00	5.00E+00	4	//END						5																				
	A	B	C	D	E	F																																																			
1	Record	Date	UTC Time	Count	Temperature	Pressure																																																			
2	1	9/29/2010	21:01:46	5	5.00E+00	5.00E+00																																																			
3	2	9/29/2010	21:01:47	5	5.00E+00	5.00E+00																																																			
4	//END																																																								
5																																																									
Maksimum beř kayıtlı bir veri gnlk dosyasındaki beř kayıt	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>UTC Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>1</td> <td>9/30/2010</td> <td>20:26:56</td> <td>1</td> <td>9.86E+01</td> <td>3.52E+01</td> </tr> <tr> <td>3</td> <td>2</td> <td>9/30/2010</td> <td>20:28:43</td> <td>2</td> <td>1.00E+02</td> <td>3.73E+01</td> </tr> <tr> <td>4</td> <td>3</td> <td>9/30/2010</td> <td>20:29:03</td> <td>3</td> <td>9.99E+01</td> <td>3.68E+01</td> </tr> <tr> <td>5</td> <td>4</td> <td>9/30/2010</td> <td>20:29:21</td> <td>4</td> <td>9.95E+01</td> <td>3.64E+01</td> </tr> <tr> <td>6</td> <td>5</td> <td>9/30/2010</td> <td>20:30:19</td> <td>5</td> <td>9.92E+01</td> <td>3.74E+01</td> </tr> <tr> <td>7</td> <td colspan="6"></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	UTC Time	Count	Temperature	Pressure	2	1	9/30/2010	20:26:56	1	9.86E+01	3.52E+01	3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01	4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01	5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01	6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01	7						
	A	B	C	D	E	F																																																			
1	Record	Date	UTC Time	Count	Temperature	Pressure																																																			
2	1	9/30/2010	20:26:56	1	9.86E+01	3.52E+01																																																			
3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01																																																			
4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01																																																			
5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01																																																			
6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01																																																			
7																																																									
Bir ilave kayıt, dolu olan yukarıdaki dosyaya yazıldıktan sonra, altıncı yazma iřlemi, kayıt altıya sahip en eski kayıtlı olanı zerine yazar. Bařka bir yazma iřlemi, kayıt yediye sahip kayıt ikiyi zerine yazacaktır ve bu iřlemler byle devam edecektir.	<table border="1"> <thead> <tr> <th></th> <th>A</th> <th>B</th> <th>C</th> <th>D</th> <th>E</th> <th>F</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Record</td> <td>Date</td> <td>UTC Time</td> <td>Count</td> <td>Temperature</td> <td>Pressure</td> </tr> <tr> <td>2</td> <td>6</td> <td>9/30/2010</td> <td>20:32:03</td> <td>6</td> <td>9.86E+01</td> <td>3.58E+01</td> </tr> <tr> <td>3</td> <td>2</td> <td>9/30/2010</td> <td>20:28:43</td> <td>2</td> <td>1.00E+02</td> <td>3.73E+01</td> </tr> <tr> <td>4</td> <td>3</td> <td>9/30/2010</td> <td>20:29:03</td> <td>3</td> <td>9.99E+01</td> <td>3.68E+01</td> </tr> <tr> <td>5</td> <td>4</td> <td>9/30/2010</td> <td>20:29:21</td> <td>4</td> <td>9.95E+01</td> <td>3.64E+01</td> </tr> <tr> <td>6</td> <td>5</td> <td>9/30/2010</td> <td>20:30:19</td> <td>5</td> <td>9.92E+01</td> <td>3.74E+01</td> </tr> <tr> <td>7</td> <td colspan="6"></td> </tr> </tbody> </table>		A	B	C	D	E	F	1	Record	Date	UTC Time	Count	Temperature	Pressure	2	6	9/30/2010	20:32:03	6	9.86E+01	3.58E+01	3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01	4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01	5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01	6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01	7						
	A	B	C	D	E	F																																																			
1	Record	Date	UTC Time	Count	Temperature	Pressure																																																			
2	6	9/30/2010	20:32:03	6	9.86E+01	3.58E+01																																																			
3	2	9/30/2010	20:28:43	2	1.00E+02	3.73E+01																																																			
4	3	9/30/2010	20:29:03	3	9.99E+01	3.68E+01																																																			
5	4	9/30/2010	20:29:21	4	9.95E+01	3.64E+01																																																			
6	5	9/30/2010	20:30:19	5	9.92E+01	3.74E+01																																																			
7																																																									



## 8.8 Veri blok kontrolü

### 8.8.1 READ\_DBL ve WRIT\_DBL (Yükleme hafızasındaki bir veri bloğunu okuma/yazma) komutları

Çizelge 8- 174 READ\_DBL ve WRIT\_DBL komutları

LAD / FBD	SCL	Açıklama
	<pre>READ_DBL (     req:=_bool_in_,     srcblk:=_variant_in_,     busy=&gt;_bool_out_,     dstblk=&gt;_variant_out_);</pre>	<p>DB başlangıç değerlerini veya bu değerlerden bir bölümünü yükleme belleğinden çalışma belleğindeki bir hedef DB'ye kopyalar.</p> <p>Yükleme bellek içeriği kopyalama işlemi sırasında değiştirilmez.</p>
	<pre>WRIT_DBL (     req:=_bool_in_,     srcblk:=_variant_in_,     busy=&gt;_bool_out_,     dstblk=&gt;_variant_out_);</pre>	<p>DB mevcut değerlerini veya bu değerlerden bir bölümünü çalışma belleğinden yükleme belleğindeki bir hedef DB'ye kopyalar.</p> <p>Çalışma bellek içeriği kopyalama işlemi sırasında değiştirilmez.</p>

Çizelge 8- 175 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
REQ IN	BOOL	Yüksek bir sinyal işlemi başlatır, BUSY = 0 ise.
SRCBLK IN	VARIANT	READ_DBL: Yükleme belleğindeki kaynak veri bloğunu işaret eder. WRIT_DBL: Çalışma belleğindeki kaynak veri bloğunu işaret eder
RET_VAL OUT	INT	Yürütme durum kodu
BUSY OUT	BOOL	BUSY = 1 okuma/yazma prosesinin tamamlanmadığını sinyal verir.
DSTBLK OUT	VARIANT	READ_DBL: Çalışma belleğinde hedef veri bloğunu işaret eder WRIT_DBL: Yükleme belleğinde hedef veri bloğunu işaret eder

Tipik olarak, bir DB y¼kleme belleęi (flash) ve alıřma belleęinin (RAM) her ikisinde saklanır. Bařlatma deęerleri (bařlangı deęerleri) her zaman y¼kleme belleęinde depolanır ve mevcut deęerler her zaman alıřma belleęinde depolanır. READ\_DBL, y¼kleme belleęindeki bařlatma deęerleri k¼mesini, program tarafından bařvurulan alıřma belleęindeki bir DB'nin mevcut deęerlerine kopyalamak iin kullanılabilir. Siz alıřma belleęindeki mevcut deęerlerinden dahili y¼kleme belleęinde ya da bellek kartında depolanan bařlatma deęerlerini g¼ncellemek iin WRIT\_DBL kullanabilirsiniz.

---

**Not****Ařırı WRIT\_DBL flash bellek yazma iřlemlerinden kaınınız**

WRIT\_DBL komutu, flash bellekte (dahili y¼kleme belleęi veya bellek kartı) yazma iřlemleri gerekleřtirir. WRIT\_DBL bir ¼retim s¼reci deęiřiklikleri gibi nadir g¼ncellemeler iin kullanılmalıdır.

---

Bu komutları kullanmadan ¼nce READ\_DBL ve WRIT\_DBL tarafından kullanılan veri blokları daha ¼nce STEP 7 tarafından oluřturulmuř olması gerekir. Kaynak DB bir "standart" tip olarak oluřturulursa, hedef DB'de "standart" tipi olmak zorundadır. Kaynak veri bloęu "optimize edilmiř" tip olarak oluřturulursa, hedef veri bloęu da "optimize edilmiř" tip olmak zorundadır.

DB'ler standart ise, o zaman bir etiket adı veya P # deęer belirtebilirsiniz. P # deęeri, herhangi bir sayıda, belirtilen boyuttaki ¼geleri (Byte , Word veya DWord ) belirtmenize ve kopyalamanıza izin verir. B¼ylece, t¼m DB'yi veya bir kısmını kopyalayabilirsiniz. DB'ler optimize edilmiř ise, yalnızca bir etiket adını belirtebilirsiniz; P # operat¼r¼n¼ kullanamazsınız. Standart veya optimize DB'ler (veya dięer alıřma - bellek tipleri iin) iin bir etiket adı belirtirseniz, o zaman bu etiket adı tarafından referans g¼sterilen her řey kopyalanır. Bu bir kullanıcı tanımlı tip, bir dizi ya da temel bir eleman olabilir. DB optimize deęil ve standart ise Tip Yapısı sadece bu komutlar tarafından kullanılabilir. Bir kullanıcı tanımlı tip (UDT) optimize bellekteki bir yapı ise bunu kullanmanız gerekir. " veri tipleri " nin kaynak ve hedef yapıların her ikisi iin de tam olarak aynı olmasını sadece kullanıcı tanımlı tip saęlar.

---

**Not****Bir "optimize" DB'de bir yapı (veri tipi Struct) kullanma**

"Optimize" DB'ler ile birlikte bir Struct veri tipini kullanarak, ¼nce Struct iin bir kullanıcı tanımlı veri tip (UDT) oluřturmanız gerekir. Daha sonra kaynak ve hedef DB'lerin her ikisini UDT ile yapılandırınız. UDT, Struct iindeki veri tiplerinin her iki DB iin tutarlı kalmasını saęlar.

"Standart" veri tabanları iin bir UDT yaratmadan Struct kullanınız.

---

READ\_DBL ve WRIT\_DBL d¼ng¼sel program taramayla asenkron olarak y¼r¼t¼l¼r. iřlem oklu READ\_DBL ve WRIT\_DBL aęıruları boyunca devam eder. DB aktarma iřini REQ = 1 ile aęırarak bařlatırsınız ve daha sonra veri aktarımının ne zaman tamamlandıęını ve doęru olduęunu belirlemek iin BUSY ve RET\_VAL sonularını izlersiniz.

Veri tutarlılıęını saęlamak iin (bařka bir deyiřle BUSY parametresi TRUE olduęu s¼rece) READ\_DBL iřlemi sırasında hedef alanı veya WRIT\_DBL iřlemi sırasında kaynak alanı deęiřtirmeyiniz.

SRCBLK ve DSTBLK parametre kısıtlamaları:

- Referans gösterilmeden önce bir veri bloğunun önceden oluşturulmuş olması gerekir.
- BOOL tipi bir VARIANT işaretçi uzunluğu 8 ile bölünebilir olmalıdır.
- STRING tipi bir VARIANT işaretçi uzunluğu kaynak ve hedef işaretçilerindekiyle aynı olmalıdır.

### Tarifler ve makine kurulum bilgileri

Tarifler veya makine kurulum bilgilerini yönetmek için READ\_DBL ve WRIT\_DBL komutlarını kullanabilirsiniz. Flaş belleğin erkenden tükenmesini önlemek için yazma sayısını sınırlamak isterseniz bile, bu esas olarak sık değişmeyen değerler için kalıcı veriyi başka şekilde elde etme yöntemi haline gelir. Bu yöntem, en azından sık değişmeyen değerler için normal güç-kesintisinde tutulan veriler için desteklenenin ötesinde kalıcı bellek miktarını etkili biçimde artırmanıza olanak sağlar. WRIT\_DBL komutunu kullanarak yükleme belleğine çalışma belleğindeki tarif bilgileri veya makine-kurulum bilgilerini kaydedebilir ve yükleme belleğindeki bu bilgileri READ\_DBL komutunu kullanarak tekrar çalışma belleğine alabilirsiniz.

Çizelge 8- 176 Durum kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
0081	Uyarı: Kaynak alanı hedef alandan daha küçüktür. Kaynak veri değişmeden hedef alandaki ekstra bayt ile tamamen kopyalanır.
7000	REQ = 0: BUSY = 0 ile çağır
7001	REQ = 1 (çalışıyor): BUSY = 1 ile ilk çağrı
7002	N. çağrı (çalışıyor): BUSY = 1
8051	Veri blok tipi hatası
8081	Kaynak alan hedef alandan daha büyüktür. Hedef alan, tamamen dolu ve kaynaktaki geri kalan baytlar göz ardı edilir.
8251	Kaynak veri blok tipi hatası
82B1	Eksik kaynak veri bloğu
82C0	Kaynak DB başka ifade ile ya da bir iletişim fonksiyonu tarafından düzenleniyor
8551	Hedef veri blok tipi hatası
85B1	Eksik hedef veri bloğu
85C0	Hedef DB başka ifade ile veya bir iletişim fonksiyonu tarafından düzenleniyor
80C3	50'den fazla READ_DBL veya 50 WRIT_DBL ifadeleri şu anda yürütme için sıradadır


Ayrıca bakınız Tarifler (Sayfa 369)

## 8.9 Adres işleme

### 8.9.1 LOG2GEO (slotu donanım kimliğinden belirleme) komutu

Bir donanım tanımlayıcıya ait mantıksal adresten coğrafi adresi (modül yuvası) belirlemek için LOG2GEO komutunu kullanınız.

Çizelge 8- 177 LOG2GEO komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := LOG2GEO(     laddr:=_word_in_,     GEOADDR:=_variant_in_out_);</pre>	<p>Bir donanım kimliğine ait modül yuvasını belirlemek için LOG2GEO komutunu kullanınız.</p>

LOG2GEO komutu donanım kimliğine dayalı mantıksal adresin coğrafi adresini belirler:

- Donanım kimliğine dayalı mantıksal adresi seçmek için LADDR parametresini kullanınız.
- GEOADDR, LADDR girişinde verilen mantıksal adresin coğrafi adresini içerir.

#### Not

HW tipinin bir bileşeni desteklemediği durumlarda, modül 0 için bir alt yuva numarası döndürülür.

LADDR girişi HW nesnesini adreslemez ise bir hata sağlanır.

Çizelge 8- 178 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
LADDR	IN	HW_ANY
RET_VAL	OUT	Int
GEOADDR	IN_OUT	Variant

Çizelge 8- 179 GEOADDR sistem veri tipi

Parametre adı	Veri tipi	Açıklama
GEOADDR	STRUCT	
HWTYPE	UNIT	Donanım tipi: <ul style="list-style-type: none"> <li>1: IO sistem (PROFINET/PROFIBUS)</li> <li>2: IO cihaz/DP slave</li> <li>3: Şasi</li> <li>4: Modül</li> <li>5: Alt modül</li> </ul> Bir donanım tipi komut tarafından desteklenmiyorsa, HWTYPE "0" çıktısıdır.
AREA	UNIT	Alan ID: <ul style="list-style-type: none"> <li>0 = CPU</li> <li>1 = PROFINET IO</li> <li>2 = PROFIBUS DP</li> </ul>
IOSYSTEM	UNIT	PROFINET IO sistem (0 = merkezi cihaz, 0-3 şasisinde)
STATION	UNIT	<ul style="list-style-type: none"> <li>Şasi numarası, alan belirteç AREA = 0 ise.</li> <li>İstasyon numarası, alan belirteç AREA &gt; 0 ise.</li> </ul>
SLOT	UNIT	Yuva numarası
SUBSLOT	UNIT	Alt modül numarası. Herhangi bir alt modül yerleştirilebilirse, bu Parametre değeri "0" olur.

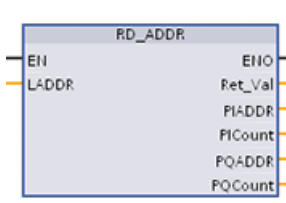
Çizelge 8- 180 Durum kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
8090	LADDR parametresinde belirtilen adres geçersiz.

## 8.9.2 RD\_ADDR (Donanım belirtecindeki IO adreslerini belirleme) komutu

Bir alt modülün I/O adreslerini almak için RD\_ADDR komutunu kullanınız.

Çizelge 8- 181 RD\_ADDR komutu

LAD / FBD	SCL	Açıklama
	<pre>ret_val := RD_ADDR(   laddr:=_word_in_,   PIADDR=&gt;_uint_out_,   PCount=&gt;_uint_out_,   PQADDR=&gt;_uint_out_,   PQCount=&gt;_uint_out_,);</pre>	Alt modülün I/O adreslerini almak için RD_ADDR komutunu kullanınız.

## Genişletilmiş komutlar

### 8.10 "Genişletilmiş" komutlar için ortak hata kodları

RD\_ADDR komutu, bir alt modülün donanım belirtecine dayalı giriş veya çıkışların başlatma adresini ve uzunluğunu belirler:

- Donanım belirtecine dayalı giriş veya çıkış modülünü seçmek için LADDR parametresini kullanınız.
- Aşağıdaki çıkış parametreleri bir giriş modülü veya çıkış modülü olmasına bağlı olarak kullanılır:
  - Bir giriş modülü durumunda, belirlenen değerler PIADDR ve PICOUNT parametrelerinde çıkıştır.
  - Bir çıkış modülü durumunda, belirlenen değerler PQADDR ve PQCOUNT parametrelerinde çıkıştır.
- PIADDR ve PQADDR parametrelerinin her biri Modülün I/ O adreslerinin başlatma adreslerini içerir.
- PICOUNT ve PQCOUNT parametrelerinin her biri giriş ve çıkışların bayt sayısını (8 giriş/çıkış için 1 bayt, 16 giriş/çıkış için 2 bayt) içerir.

Çizelge 8- 182 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
LADDR	IN	HW_IO
RET_VAL	OUT	Int
PIADDR	OUT	UDInt
PICOUNT	OUT	UInt
PQADDR	OUT	UDInt
PQCOUNT	OUT	UInt

Çizelge 8- 183 Durum kodları

RET_VAL (W#16#...)	Açıklama
0000	Hata yok
8090	LADDR parametresinde modülünün donanım belirteci geçersiz.

## 8.10 "Genişletilmiş" komutlar için ortak hata kodları

Çizelge 8- 184 "Genişletilmiş" komutlar için ortak durum hata kodları

Durum kodu (W#16#...) <sup>1</sup>	Açıklama
8x22 <sup>2</sup>	Alan, giriş için çok küçük
8x23	Alan, çıkış için çok küçük
8x24	Kuraldışı giriş alanı
8x25	Kuraldışı çıkış alanı

*8.10 "Geniřletilmiř" komutlar iin ortak hata kodlar*

Durum kodu (W#16#....) <sup>1</sup>	Aıklama
8x28	Kuraldışı giriř bit ataması
8x29	Kuraldışı ıkıř bit ataması
8x30	ıkıř alanı salt-okunur DB' dir.
8x3A	DB mevcut deęil.

- <sup>1</sup> Bir kod bloęu alıřtırıldıęında bu hatalardan biri oluřursa, o zaman CPU RUN'da (varsayılan) kalır veya STOP'a gitmek iin yapılandırılabilir. Opsiyonel olarak, hatayı yerel biimde iřlemek (CPU RUN'da kalır) ve bu hataya karřı programlanmıř bir tepki oluřturmak iin bu kod bloęu iinde GetError veya GetErrorID komutlarını kullanabilirsiniz.
- <sup>2</sup> "X" hata ile parametre numarasını gsterir. Parametre numaraları 1 ile bařlar.

*Geniřletilmiř komutlar*

*8.10 "Geniřletilmiř" komutlar iin ortak hata kodlar*

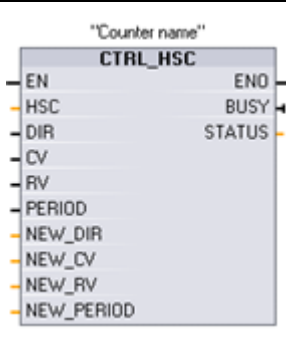
---



## Teknoloji komutları

### 9.1 Yüksek hızlı sayıcı

Çizelge 9- 1 CTRL\_HSC komutu

LAD / FBD	SCL	Açıklama
	<pre>"CTRL_HSC_1_DB" (   hsc:=_hw_hsc_in_,   dir:=_bool_in_,   cv:=_bool_in_,   rv:=_bool_in_,   period:=_bool_in_,   new_dir:=_int_in_,   new_cv:=_int_in_,   new_rv:=_dint_in_,   new_period:=_int_in_,   busy:=_bool_out_,   status:=_word_out_);</pre>	<p>Her CTRL_HSC (Kontrol yüksek hızlı sayıcı) komutu sayıcı verilerini korumak için DB'de saklanan bir yapı kullanır. CTRL_HSC komutu editör içine yerleştirildiği zaman DB'yi atayınız.</p>

- 1 Komutu yerleştirdiğinizde STEP 7 otomatik olarak DB oluşturur.
- 2 SCL örneğinde, "CTRL\_HSC\_1\_DB " kopya DB'nin ismidir.

Çizelge 9- 2 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
HSC	IN	HW_HSC	HSC kimliği
DIR <sup>1,2</sup>	IN	Bool	1 = Yeni yön isteği
CV <sup>1</sup>	IN	Bool	1 = Yeni sayıcı değeri ayarlama isteği
RV <sup>1</sup>	IN	Bool	1= Yeni referans değeri ayarlama isteği
PERIOD <sup>1</sup>	IN	Bool	1 = Yeni periyot değeri ayarlama isteği (yalnızca frekans ölçme modunda)
NEW_DIR	IN	Int	Yeni yön: 1= ileri, -1= geri
NEW_CV	IN	DInt	Yeni sayıcı değeri
NEW_RV	IN	DInt	Yeni referans değeri
NEW_PERIOD	IN	Int	(Sadece frekans ölçme modu için) saniye cinsinden yeni periyot değeri: 1= 1 s 2 = .1 s 3 = 0.1 s

Parametre ve tipi	Veri tipi	Açıklama
BUSY <sup>3</sup>	OUT	Bool
STATUS	OUT	Word

- <sup>1</sup> Bir parametre değeri güncellemesi talep edilmediyse, o zaman karşılık gelen giriş değerleri göz ardı edilir.
- <sup>2</sup> Yapılandırılmış sayma yönü "Kullanıcı programı (iç yön kontrol)" olarak ayarlanırsa, DIR parametresi geçerlidir. HSC cihaz yapılandırmasında bu parametrenin nasıl kullanılacağını belirlemelisiniz.
- <sup>3</sup> CPU veya SB üzerinde bir HSC için, BUSY parametresi her zaman 0 değerini alır.

Sayma / frekans fonksiyonu, sıfırlama seçeneklerini, kesme olay yapılandırması, donanım I/O'u ve sayma değeri adresiyle ilgili CPU için cihaz yapılandırmasında her HSC için parametreleri yapılandırınız.

HSC için parametrelerin bazıları sayma sürecinin program kontrolünü sağlamak için kullanıcı programınız tarafından değiştirilebilir:

- Sayma yönünü bir NEW\_DIR değere ayarlayınız
- Geçerli sayma değerini bir NEW\_CV değere ayarlayınız
- Referans değeri bir NEW\_RV değere ayarlayınız
- (Frekans ölçüm modu için) periyot değerini bir NEW\_PERIOD değere ayarlayınız

CTRL\_HSC komut çalıştırıldığında aşağıdaki Boole bayrak değerleri 1'e ayarlanırsa, gelen NEW\_xxx değeri sayıcıya yüklenir. Çoklu istekleri (birden fazla bayrak aynı anda ayarlanır) CTRL\_HSC komutu tek bir yürütme işlenir.

- DIR= 1, bir NEW\_DIR değerini yüklemek için bir istek, 0 = değişme yok
- CV = 1, bir NEW\_CV değerini yüklemek için bir istek, 0 = değişme yok
- RV = 1, bir NEW\_RV değerini yüklemek için bir istek, 0 = değişme yok
- PERIOD = 1, bir NEW\_PERIOD değerini yüklemek için bir istek, 0 = değişme yok

CTRL\_HSC komutu, genellikle sayıcı donanım kesme olayı tetiklendiğinde yürütülen bir donanım kesme OB'sine yerleştirilir. Örneğin CV = RV olayı sayıcı kesmesini tetiklerse, bir donanım kesmesi OB kod bloğu CTRL\_HSC komutunu yürütür ve bir NEW\_RV değeri yükleyerek referans değerini değiştirebilirsiniz.

Mevcut sayma değeri, CTRL\_HSC parametrelerinde mevcut değildir. Mevcut sayma değerini saklayan proses görüntü adresi, yüksek hızlı sayıcı donanım yapılandırması sırasında atanır. Doğrudan sayma değerini okumak için bir program mantığı kullanabilirsiniz. Programınıza dönen değer, sayıcının okunduğu an için doğru bir sayma olacaktır. Sayıcı, yüksek hızlı olayları saymaya devam edecektir. Bu nedenle Programınız eski bir sayma değerini kullanarak bir süreci tamamlanmadan önce, gerçek sayma değeri değiştirebilir.

#### HSC mevcut sayma değeri: Program erişim, değer aralığı ve devrilme davranışı

CPU, bir giriş (I) adresinde her bir HSC'nin mevcut değerini saklar. Aşağıdaki çizelge, her bir HSC'nin mevcut değerine atanan varsayılan adresleri göstermektedir. Cihaz Yapılandırmasında CPU özelliklerini değiştirerek mevcut değeri için I adresini değiştirebilirsiniz.

Yüksek hızlı sayıcılar mevcut sayma değerini saklamak için bir DInt değerini kullanınız. DInt sayma değeri -2147483648 ila +2147483647 aralığında bir diziye sahiptir. Sayıcı yukarı doğru sayarken maksimum pozitif değerden maksimum negatif değere ve aşağı sayarken maksimum negatif değerden maksimum pozitif değere devrilir.

HSC	Mevcut değeri veri tipi	Varsayılan mevcut değeri adresi
HSC1	DInt	ID1000
HSC2	DInt	ID1004
HSC3	DInt	ID1008
HSC4	DInt	ID1012
HSC5	DInt	ID1016
HSC6	DInt	ID1020

Bir hata oluşursa, ENO = 0 yapılır ve STATUS çıkışı bir durum kodunu gösterir.

Çizelge 9- 3 Yürütme durum kodları

STATUS (W#16#)	Açıklama
0	Hata yok
80A1	HSC belirteci bir HSC adresleme
80B1	NEW_DIR'da kuraldışı değer
80B2	NEW_CV'de kuraldışı değer
80B3	NEW_RV'de kuraldışı değer
80B4	NEW_PERIOD'da kuraldışı değer
80C0	Yüksek hızlı sayıcıya çoklu erişim
80D0	Yüksek hızlı sayıcı (HSC), CPU donanım yapılandırmasında etkin değil

### 9.1.1 Yüksek hızlı sayıcının çalışması

Yüksek hızlı sayıcılar ( HSC ) döngüsel OB yürütme hızından daha hızlı meydana gelen olayları sayabilir. Sayılacak olaylar OB yürütme oranından daha yavaş oluşursa, CTU, CTD veya CTUD standart sayıcı komutlarını kullanabilirsiniz. Olaylar, OB yürütme hızından daha hızlı ortaya çıkarsa, o zaman daha hızlı HSC cihazı kullanınız. CTRL\_HSC komutu, programınızın, HSC parametrelerinin bazılarını programlı olarak değiştirmesine izin verir.

Örneğin: Artan bir shaft kodlayıcı için bir girdi olarak HSC kullanabilirsiniz. Shaft kodlayıcı tur başına saymaların belirtilen sayısını ve bir turda bir kere oluşan bir sıfırlama darbesi sağlar. Shaft kodlayıcıdan gelen saat (ler) ve sıfırlama darbesi HSC için girişler sağlar.

HSC, ön ayarlardan birincisiyle yüklenir ve çıkışlar mevcut sayısının mevcut ön ayarından az olduğu durumda, zaman periyodu için etkinleştirilir. HSC, sıfırlama meydana geldiğinde, mevcut sayısı ön ayar değerine eşit olduğunda ve ayrıca yön değişimi olduğunda bir kesme sağlar.

Her bir mevcut - sayma - değeri – ön ayar değerine - eşit kesme olayı oluştuğunda, yeni bir ön ayar yüklenir ve çıkışlar için bir sonraki durum ayarlanır. Sıfırlama kesme olayı olduğunda, ilk önayar ve ilk çıkış durumları ayarlanır ve çevrim tekrarlanır.

Kesmeler, HSC'nin sayma oranından çok daha düşük hızda ortaya çıktığı için yüksek hızlı işlemlerin hassas kontrolü, CPU'nun tarama çevrimine nispeten küçük etkisi ile uygulanabilir. Kesme ekleme yöntemi, her yeni bir önayar yüklemesinin, kolay durum kontrolü için ayrı bir kesme rutini içinde yapılmasına imkan sağlar. Alternatif olarak, tüm kesme olayları tek bir kesme rutini içinde işlenebilir.

### HSC giriş kanalı seçimi

Aşağıdaki çizelgeyi kullanınız, CPU ve bağlandığınız SB giriş kanallarının proses sinyallerinizdeki maksimum darbe hızlarını desteklediğinden emin olunuz.

#### Not

**CPU ve SB giriş kanalları (V4 veya üstü firmware) yapılandırılabilir giriş filtresi sürelerine sahiptir**

Önceki firmware sürümleri sabit HSC giriş kanallarına ve değiştirilemeyen sabit filtre sürelerine sahiptir.

V4 veya sonraki sürümler, giriş kanalları ve filtre sürelerini atamanıza izin verir. 6.4 ms'lik varsayılan giriş filtresi ayarı, proses sinyalleriniz için çok yavaş olabilir. HSC girişleri için filtre süreleri, HSC uygulamanız için optimize edilmiş olmalıdır. Bakınız "Dijital giriş filtre sürelerini yapılandırma" ayrıntıları için" (Sayfa 144)

Çizelge 9- 4 CPU giriş: maksimum frekans

CPU	CPU Giriş kanalı	1 veya 2 faz modu	A/B Dördün faz modu
1211C	Ia.0 ila Ia.5	100 kHz	80 kHz
1212C	Ia.0 ila Ia.5	100 kHz	80 kHz
	Ia.6, Ia.7	30 kHz	20 kHz
1214C ve 1215C	Ia.0 ila Ia.5	100kHz	80kHz
	Ia.6 ila Ib.5	30 kHz	20 kHz
1217C	Ia.0 ila Ia.5	100 kHz	80 kHz
	Ia.6 ila Ib.1	30 kHz	20 kHz
	Ib.2 ila Ib.5 (.2+, .2- ila .5+, .5-)	1 MHz	1 MHz

Çizelge 9- 5 B sinyal kart girişi: maksimum frekans (opsiyonel kart)

SB sinyal kartı	SB giriş kanalı	1 veya 2 faz modu	A/B dördün faz modu
SB 1221, 200 kHz	Ie.0 ila Ie.3	200kHz	160 kHz
SB 1223, 200 kHz	Ie.0, Ie.1	200kHz	160 kHz
SB 1223	Ie.0, Ie.1	30 kHz	20 kHz

### HSC için işlevsellik seçimi

Tüm HSC'ler, aynı sayıcı çalışma modu için aynı şekilde işlev sağlar. HSC işlev özellikleri için sayıcı modu, yön kontrolü ve başlangıç yönü CPU cihaz yapılandırmasında atanır.

Dört temel tip HSC vardır:

- Dahili yön kontrollü tek fazlı sayıcı
- Harici yön kontrollü tek fazlı sayıcı

- 2 saat girişli iki fazlı sayıcı
- A/B faz dördün sayıcı

Bir reset airisi olan veya olmayan her HSC tipini kullanabilirsiniz. Reset airisi aktif edildiğinde (bazı kısıtlamalarla birlikte, aşağıdaki çizelgeye bakınız) mevcut değeri temizlenir ve reset girişini devre dışı bırakana kadar açık tutulur.

- Frekans fonksiyonu: Bazı HSC modları, HSC'nin, bir mevcut darbe sayısı yerine frekansı rapor (sayma tipi) edecek şekilde yapılandırılmasına izin verir. Üç farklı frekans ölçme periyodu kullanılabilir: 0.01, 0.1 ya da 1.0 saniye.
- Frekans Ölçme periyodu, HSC'nin hangi sıklıkta hesaplama yapacağını ve yeni bir frekans değerini belirler. Raporlanan frekans, son ölçme periyodunda saymaların toplam sayısına göre belirlenen ortalama bir değerdir. Frekans hızla değişiyor ise, raporlanan değer bu ölçme periyodu içerisinde meydana gelen en yüksek ile en düşük frekans arasında bir ara değer olacaktır. Frekans, her zaman frekans ölçme periyodu ayarına bakılmaksızın Hertz (saniyede darbeler) cinsinden rapor edilir.
- Sayıcı modları ve girişler: Aşağıdaki çizelge, saat için kullanılan girişleri, yön kontrolünü ve HSC ile ilgili reset fonksiyonlarını gösterir.

Aynı giriş, iki farklı fonksiyon için kullanılamaz, fakat HSC'sinin mevcut modu ile kullanılan tüm girişler başka bir amaç için kullanılabilir. Örneğin HSC1, iki gömülü giriş kullanan bir modda ise ancak üçüncü harici reset girişini (I0.3' te varsayılan atama) kullanmıyor ise, o zaman I0.3, kenar kesmeleri veya HSC 2 için kullanılabilir.

Çizelge 9-6 HSC için sayma modları

Tip	Giriş 1	Giriş 2	Giriş 3	Fonksiyon
Dahili yön kontrollü tek fazlı sayıcı	Saat	-	-	Say veya frekans
			Reset	Say
Harici yön kontrollü tek fazlı sayıcı	Saat	Yön	-	Say veya frekans
			Reset	Say
2 saat girişleri ile iki fazlı sayıcı	Saat artır	Saat azalt	-	Say veya frekans
			Reset	Say
A/B-faz dördün sayıcı	Faz A	Faz B	-	Say veya frekans
			Reset <sup>1</sup>	Say

<sup>1</sup> Kodlayıcı için: Faz Z, Başlangıç

## HSC için giriş adresleri

CPU yapılandırılırken, her HSC için "Donanım girişleri" ni etkinleştirmek ve yapılandırmak için seçeneğiniz vardır.

Tüm HSC girişleri, CPU modülünün ön kısmına takılan opsiyonel sinyal kartı veya CPU modülü üzerindeki terminallere bağlı olmalıdır.

### Not

Aşağıdaki çizelgelerde gösterildiği gibi, farklı HSC'ler için opsiyonel sinyallerin varsayılan atamaları üst üste gelir. Örneğin, HSC 1 için opsiyonel harici reset, HSC 2 için girişlerden biri ile aynı girişi kullanır.

V4 CPU'lar veya sonraki sürümler için CPU yapılandırma sırasında HSC girişlerini yeniden atayabilirsiniz. Varsayılan giriş atamalarını kullanmak zorunda değilsiniz.

Herhangi bir giriş iki HSC tarafından kullanılıyor olmayacak şekilde, her zaman VHS'lerinizi yapılandırmış olduğunuzu kontrol ediniz.

Aşağıdaki çizelgeler, yerleşik işlemciler ve opsiyonel SB I / O için HSC giriş varsayılan atamalarını göstermektedir. (Seçilen SB modelin sadece 2 girişi varsa, sadece 4.0 ve 4.1 girişleri kullanılabilir.)

### HSC giriş çizelge tanımları

- **Tek-faz:** C Saat girişi, [d] yön girişi (opsiyonel) ve [R] (opsiyonel) harici reset girişi (Reset, sadece "Sayma" modu için kullanılabilir.)
- **İki faz:** CU, **Clock Up (Saat artır)** girişi, CD **Clock Down (Saat azalt)** girişi ve [R] harici reset girişi (isteğe bağlı). (Reset, sadece "Sayma" modu için kullanılabilir.)
- **AB-faz dördün:** A Saat A girişi, B Saat B girişi ve [R] (opsiyonel) harici reset girişidir. (Reset, sadece "Sayma" modu için kullanılabilir.)

Çizelge 9-7 CPU 1211C: HSC varsayılan adres atamaları

HSC sayıcı modu		CPU yerleşik giriş (varsayılan 0.x)						Opsiyonel SB girişi (varsayılan 4.x) <sup>1</sup>			
		0	1	2	3	4	5	0	1	2	3
HSC 1	1-faz	C	[d]		[R]			C	[d]		[R]
	2- faz	CU	CD		[R]			CU	CD		[R]
	AB- faz	A	B		[R]			A	B		[R]
HSC 2	1-faz		[R]	C	[d]				[R]	C	[d]
	2- faz		[R]	CU	CD				[R]	CU	CD
	AB- faz		[R]	A	B				[R]	A	B
HSC 3	1-faz					C	[d]	C	[d]		[R]
	2- faz										
	AB- faz										
HSC4	1-faz					C	[d]	C	[d]		[R]
	2- faz					CU	CD				
	AB- faz					A	B				

HSC sayıcı modu		CPU yerleşik giriş (varsayılan 0.x)						Opsiyonel SB girişi (varsayılan 4.x) <sup>1</sup>			
		0	1	2	3	4	5	0	1	2	3
HSC 5	1-faz							C	[d]		[R]
	2- faz							CU	CD		[R]
	AB- faz							A	B		[R]
HSC 6	1-faz								[R]	C	[d]
	2- faz								[R]	CU	CD
	AB- faz								[R]	A	B

<sup>1</sup> Sadece 2 dijital girişi olan bir SB sadece 4.0 ve 4.1 girişleri sağlar.

Çizelge 9- 8 CPU 1212C: HSC varsayılan adres atamaları

HSC sayıcı modu		CPU yerleşik giriş (varsayılan 0.x)								Opsiyonel SB girişi (varsayılan 4.x) <sup>1</sup>			
		0	1	2	3	4	5	6	7	0	1	2	3
HSC 1	1-faz	C	[d]		[R]					C	[d]		[R]
	2- faz	CU	CD		[R]					CU	CD		[R]
	AB- faz	A	B		[R]					A	B		[R]
HSC 2	1-faz		[R]	C	[d]						[R]	C	[d]
	2- faz		[R]	CU	CD						[R]	CU	CD
	AB- faz		[R]	A	B						[R]	A	B
HSC 3	1-faz					C	[d]		[R]	C	[d]		[R]
	2- faz					CU	CD		[R]				
	AB- faz					A	B		[R]				
HSC4	1-faz						[R]	C	[d]	C	[d]		[R]
	2- faz						[R]	CU	CD				
	AB- faz						[R]	A	B				
HSC 5	1-faz									C	[d]		[R]
	2- faz									CU	CD		[R]
	AB- faz									A	B		[R]
HSC 6	1-faz										[R]	C	[d]
	2- faz										[R]	CU	CD
	AB- faz										[R]	A	B

<sup>1</sup> Sadece 2 dijital girişleri olan bir SB sadece 4.0 ve 4.1 girişleri sağlar.

Çizelge 9- 9 CPU 1214C, CPU 1215C ve CPU1217C:

HSC varsayılan adres atamaları

(yalnızca yerleşik girişler, opsiyonel SB adresleri için bir sonraki çizelgeye bakınız)

HSC sayıcı modu		Dijital giriş baytı 0 (varsayılan: 0.x)							Dijital giriş baytı 1 (varsayılan: 1.x)						
		0	1	2	3	4	5	6	7	0	1	2	3	4	5
HSC 1	1-faz	C	[d]		[R]										
	2- faz	CU	CD		[R]										
	AB- faz	A	B		[R]										
HSC 2	1-faz		[R]	C	[d]										
	2- faz		[R]	CU	CD										
	AB- faz		[R]	A	B										
HSC 3	1-faz					C	[d]		[R]						
	2- faz					CU	CD		[R]						
	AB- faz					A	B		[R]						
HSC4	1-faz						[R]	C	[d]						
	2- faz						[R]	CU	CD						
	AB- faz						[R]	A	B						
HSC 5	1-faz									C	[d]	[R]			
	2- faz									CU	CD	[R]			
	AB- faz									A	B	[R]			
HSC 6	1-faz												C	[d]	[R]
	2- faz												CU	CD	[R]
	AB- faz												A	B	[R]

Çizelge 9- 10 Yukarıdaki çizelgedeki CPU'larda opsiyonel SB: HSC varsayılan adres atamaları

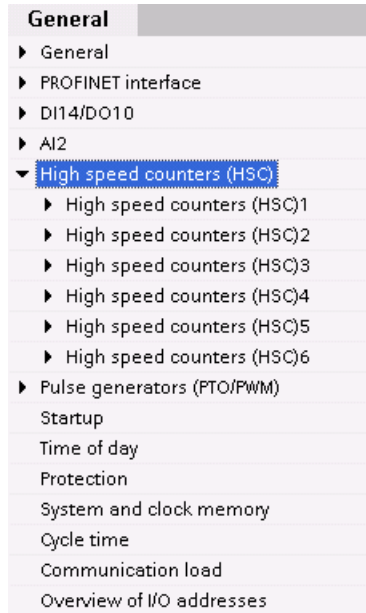
HSC		Opsiyonel SB girişleri (varsayılan: 4.x) <sup>1</sup>			
		0	1	2	3
HSC 1	1-faz	C	[d]		[R]
	2- faz	CU	CD		[R]
	AB- faz	A	B		[R]
HSC 2	1-faz		[R]	C	[d]
	2- faz		[R]	CU	CD
	AB- faz		[R]	A	B
HSC 5	1-faz	C	[d]		[R]
	2- faz	CU	CD		[R]
	AB- faz	A	B		[R]
HSC 6	1-faz		[R]	C	[d]
	2- faz		[R]	CU	CD
	AB- faz		[R]	A	B

<sup>1</sup> Sadece 2 dijital girişi olan bir SB sadece 4.0 ve 4.1 girişleri sağlar.



**Not**

Yüksek hızlı sayıcı cihazlar tarafından kullanılan dijital I/O noktaları CPU cihaz yapılandırması sırasında atanır. Dijital I/O nokta adresleri HSC cihazlara atandığı zaman, atanan I/O nokta adresleri değerleri bir izleme çizelgesindeki kuvvet fonksiyonu tarafından değiştirilemez.

**9.1.2 HSC yapılandırması**

6 adete kadar yüksek hızlı sayıcıyı yapılandırabilirsiniz. CPU Cihaz Yapılandırmasını düzenleyiniz ve her münferit HSC için HSC özelliklerini atayınız.

Bir HSC için "Enable" seçeneğini seçerek o HSC'yi etkinleştiriniz.

HSC çalışmasını kontrol etmek için kullanıcı programınızda CTRL\_HSC komutunu kullanınız.

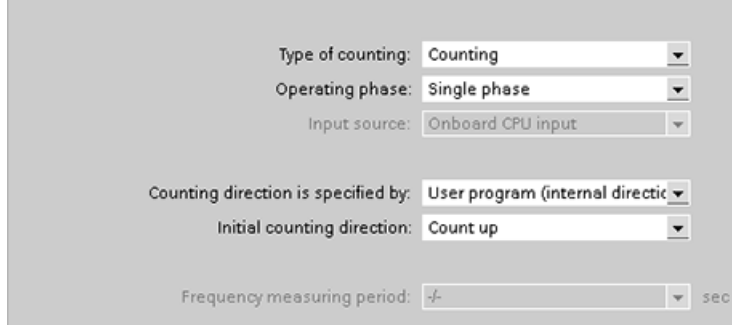
**Enable** Enable this high speed counter for use **UYARI****Dijital giriş kanalları için filtre süre ayar değişikliklerindeki riskler**

Bir dijital giriş kanalı için filtre süresi, bir önceki ayarın değiştirilmesi durumunda, filtre yeni girişlere tamamen duyarlı hale gelmeden önce, yeni bir "0" seviye giriş değeri, toplam 20.0 ms süreye kadar sunulması gerekebilir. Bu süre boyunca, 20.0 ms'den daha az olan kısa "0" darbe olayları algılanamayabilir veya sayılamayabilir.

Filtre sürelerindeki bu değişiklik, personelin ölüm veya ciddi yaralanmalarına neden olabilen beklenmeyen makine veya proses çalışmalarıyla ve/veya ekipmana zarar gelmesiyle sonuçlanabilir.

Yeni bir filtre zamanının hemen devreye girdiğinden emin olmak için, CPU'yu güç çevrimine sokun.

HSC'yi etkinleştirdikten sonra, sayıcı fonksiyonu, başlangıç değerleri, reset seçenekleri ve kesme olayları gibi diğer parametreleri yapılandırınız.



HSC yapılandırması hakkında ek bilgi için CPU yapılandırma hakkındaki bölüme bakınız (Sayfa 142).

## 9.2 PID kontrol

STEP 7, S7-1200 CPU için aşağıdaki PID komutlarını sağlar:

- PID\_Compact komutu sürekli giriş-çıkış değişkenleri ile teknik süreçleri kontrol etmek için kullanılır.
- PID\_3Step komutu, açık ve kapalı tahrik için ayrı sinyaller gerektiren vanalar gibi motor tahrikli cihazları kontrol etmek için kullanılır.

---

### Not

PID yapılandırmasında yaptığınız değişiklikler ve RUN modunda yükleme, CPU STOP'tan RUN moduna geçene kadar etkili olmaz. "Başlangıç değeri kontrolü" kullanan "PID parametreleri" diyalogunda yaptığınız değişiklikler hemen etkili olacaktır.

Her iki PID komutu (PID\_3Step ve PID\_Compact) ile yol verme sırasında P-, I- ve D- bileşenlerini ("pre-tuning" için yapılandırılmış ise) hesaplayabilirsiniz. Ayrıca parametreleri optimize etmenize izin vermek için "ince ayar" ile ilgili komutu yapılandırabilirsiniz. Parametreleri el ile belirlemeniz gerekmez.

---

### Not

**Örnekleme süresinin sabit aralıklarında PID komutunu yürütünüz (tercihen döngüsel OB'de)**

PID döngüsü, kontrol değerindeki değişikliklere tepki vermek için belirli bir zamana ihtiyacı olduğundan, her çevrimde çıkış değerinin hesaplanması gerekmez. Ana program çevrim OB'sinde (OB 1 gibi) PID komutunu yürütmeyiniz.

PID algoritmasının örnekleme süresi, iki adet çıkış değeri (kontrol değeri) hesaplaması arasındaki süreyi temsil eder. Çıkış değeri, öz- ayarlama sırasında hesaplanır ve çevrim süresinin bir katına yuvarlanır. PID komutunun diğer tüm fonksiyonları her çağırma yürütülür.

## PID algoritması

PID ( Oransal /integral /Türev ) kontrolör, iki çağrı arasındaki zaman aralığını ölçer ve sonra örnekleme zamanını izlemek için sonuçları değerlendirir. Ortalama bir örnekleme zamanı değeri, her mod geçişinde ve ilk yol verme sırasında oluşturulur. Bu değer, izleme işlevi için referans olarak ve hesaplama amacıyla kullanılır. İzleme, iki çağrı arasındaki mevcut ölçme süresi ve tanımlı kontrolörün örnekleme süresinin ortalama değerinden oluşur.

PID kontrol için çıkış değeri üç bileşenden oluşmaktadır:

- P (oransal) : "P" bileşeni ile hesaplandığında, çıkış değeri, ayar noktası ile proses değeri (giriş değeri) arasındaki fark ile orantılıdır.
- I (integral) : "I" bileşeni ile hesaplandığında, nihai olarak farkı düzeltmek için çıkış değeri, ayar noktası ile proses değeri (giriş değeri) arasındaki farkın süresiyle orantılı olarak artar.
- D (türevi) : "D" bileşeni ile hesaplandığında, ayar noktası ile proses değeri (giriş değeri) arasındaki farkın artan değişim hızının bir fonksiyonu olarak çıkış değeri artar. Çıkış değeri mümkün olduğunca hızlı bir şekilde ayar noktasına düzeltilir.

PID kontrolör PID\_Compact komutu çıkış değerini hesaplamak için aşağıdaki formülü kullanır.

$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

y	Çıkış değeri	x	Proses değeri
w	Ayar noktası değeri	s	Laplace operatörü
K <sub>p</sub>	Oransal kazanç (P bileşeni)	a	Türevsel gecikme katsayısı (D bileşeni)
T <sub>i</sub>	İntegral eylem zamanı (I bileşeni)	b	Oransal eylem ağırlıklandırma (P bileşeni)
T <sub>D</sub>	Türevsel eylem zamanı (D bileşeni)	c	Türevsel eylem ağırlıklandırma (D bileşeni)

PID kontrolör PID\_3Step komutu çıkış değerini hesaplamak için aşağıdaki formülü kullanır.

$$\Delta y = K_p \cdot s \cdot \left[ (b \cdot w - x) + \frac{1}{T_i \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

y	Çıkış değeri	x	Proses değeri
w	Ayar noktası değeri	s	Laplace operatörü
K <sub>p</sub>	Oransal kazanç (P bileşeni)	a	Türevsel gecikme katsayısı (D bileşeni)
T <sub>i</sub>	Türevsel eylem zamanı (I bileşeni)	b	Oransal eylem ağırlıklandırma (P bileşeni)
T <sub>D</sub>	Türevsel eylem zamanı (D bileşeni)	c	Türevsel eylem ağırlıklandırma (D bileşeni)

### 9.2.1 PID komutu ve teknoloji nesnesinin yerleştirilmesi

PID kontrolü için STEP 7 iki komut sağlar:

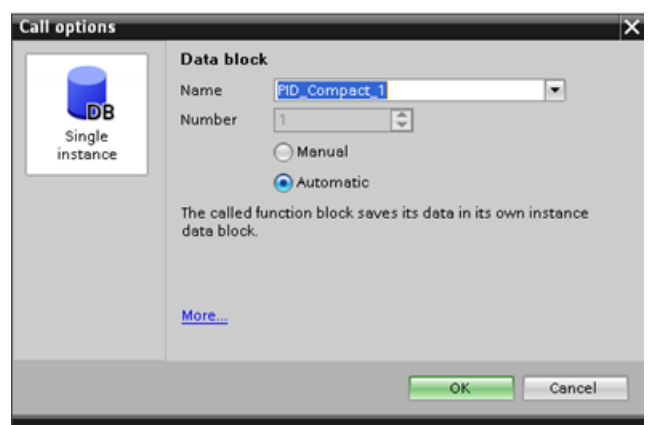
- PID\_Compact komutu ve ilgili teknoloji nesnesi, ayarlaması olan evrensel bir PID kontrolör sağlar. Teknoloji nesnesi, kontrol döngüsü için tüm ayarları içerir.
- PID\_3Step komutu ve ilgili teknoloji nesnesi, motor tahrikli vanalar için özel ayarlara sahip bir PID kontrolör sağlar. Teknoloji nesnesi kontrol döngüsü için tüm ayarları içerir. PID\_3Step kontrolör iki ilave Boolean çıkışı sağlar.

Teknoloji nesnesini oluşturduktan sonra, parametreleri (Sayfa 437) yapılandırmanız gerekir. Ayrıca PID kontrolörün (Sayfa 439) çalışmasını başlatmak için otomatik ayar parametrelerini (yol verme sırasında "pre-tuning" veya manuel "ince ayar") ayarlayınız.

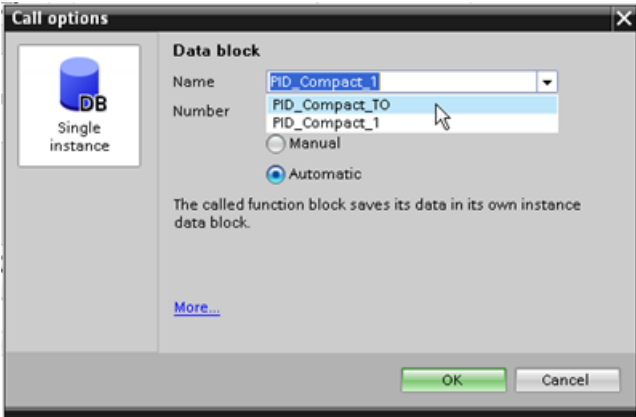
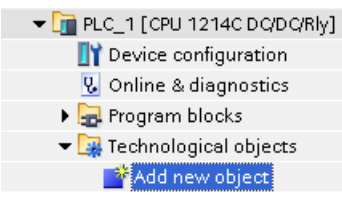
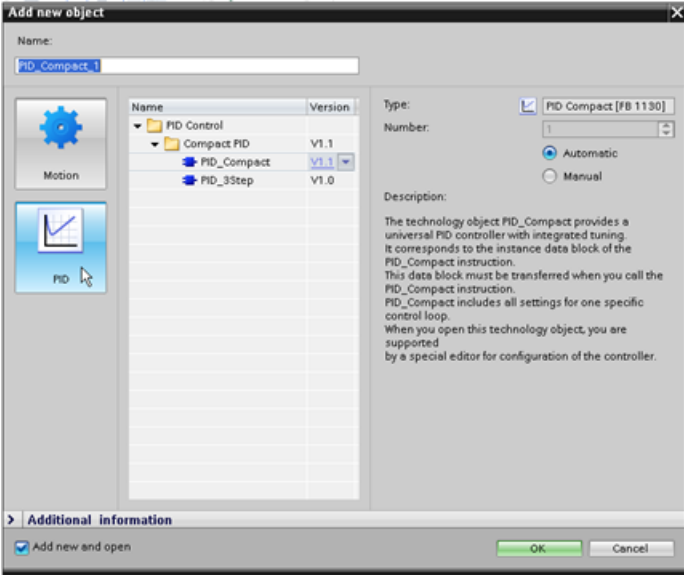
Çizelge 9- 11 PID komutu ve teknoloji nesnesinin yerleştirilmesi

Kullanıcı programı içine bir PID komutu yerleştirdiğinizde, STEP 7 otomatik olarak komut için bir teknoloji nesnesi ve bir kopya DB oluşturur. Kopya DB, PID komutu tarafından kullanılan tüm parametreleri içerir. Her PID komutunun düzgün çalışması için kendisinin benzersiz bir kopya DB'si olmalıdır.

PID komutu yerleştirilip, teknoloji nesnesi ve kopya DB oluşturduktan sonra, teknoloji nesnesi (Sayfa 437) için parametreleri yapılandırınız.



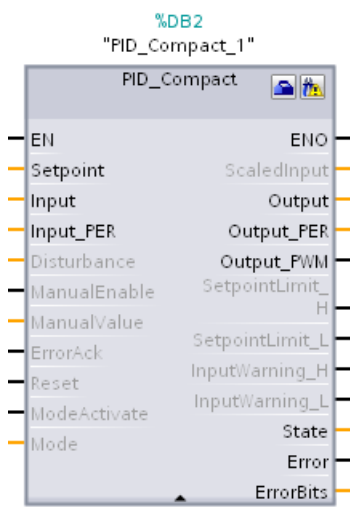
Çizelge 9- 12 (Opsiyonel) proje gezgininden bir teknoloji nesnesi oluşturma

<p>Ayrıca PID komutunu yerleştirmeden önce projeniz için teknoloji nesnelere oluşturabilirsiniz. Kullanıcı programına bir PID komutu yerleştirmeden önce teknoloji nesnesi oluşturarak PID komutu yerleştirdiğinizde, teknoloji nesnesini seçebilirsiniz.</p>	
<p>Bir teknoloji nesnesi oluşturmak için proje gezgininde "yeni nesne ekle" ikonuna çift tıklayınız.</p>	
<p>"Control" simgesine tıklayınız ve PID kontrolör (PID_Compact veya PID_3Step) tipi için teknoloji nesnesini seçiniz. Teknoloji nesnesi için opsiyonel bir ad oluşturabilirsiniz. Teknoloji nesnesi oluşturmak için "OK" ı tıklayınız.</p>	

## 9.2.2 PID\_Compact komutu

PID\_Compact komutu otomatik ve manuel mod için entegre edilmiş öz-ayarlanabilir evrensel bir PID kontrolör sağlar.

Çizelge 9- 13 PID\_Compact komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows the PID_Compact block with the following inputs and outputs:</p> <ul style="list-style-type: none"> <li>EN (Enable)</li> <li>Setpoint (ScaledInput)</li> <li>Input</li> <li>Input_PER (Output_PER)</li> <li>Disturbance (Output_PWM)</li> <li>ManualEnable (SetpointLimit_H)</li> <li>ManualValue (SetpointLimit_L)</li> <li>ErrorAck (InputWarning_H)</li> <li>Reset (InputWarning_L)</li> <li>ModeActivate (State)</li> <li>Mode (Error)</li> <li>ErrorBits</li> </ul>	<pre>"PID_Compact_1" (     Setpoint:=_real_in_,     Input:=_real_in_,     Input_PER:=_word_in_,     Disturbance:=_real_in_,     ManualEnable:=_bool_in_,     ManualValue:=_real_in_,     ErrorAck:=_bool_in_,     Reset:=_bool_in_,     ModeActivate:=_bool_in_,     Mode:=_int_in_,     ScaledInput=&gt;_real_out_,     Çıkış=&gt;_real_out_,     Output_PER=&gt;_word_out_,     Output_PWM=&gt;_bool_out_,     SetpointLimit_H=&gt;_bool_out_,     SetpointLimit_L=&gt;_bool_out_,      InputWarning_H=&gt;_bool_out_,      InputWarning_L=&gt;_bool_out_,     State=&gt;_int_out_,     Error=&gt;_bool_out_,     ErrorBits=&gt;_dword_out_ );</pre>	<p>PID_Compact otomatik ve manuel mod için öz-ayarlanabilir bir PID kontrolör sağlar. PID_Compact, ağırlıklı P- ve D- bileşenine sahip ve sarma engelleyicili bir PID T1 kontrolörüdür.</p>

- 1 Komutu yerleştirdiğinizde STEP 7 otomatik olarak teknoloji nesnesi ve kopya DB oluşturur. Kopya DB teknoloji nesnesinin parametrelerini içerir.
- 2 SCL örneğinde, "PID\_Compact\_1" kopya DB'nin ismidir.

Çizelge 9- 14 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
Setpoint	IN	Real
Input	IN	Real

Otomatik modda PID kontrolör ayar noktası değeri. (Varsayılan değer: 0.0)

Kullanıcı programının bir etiketi, proses değerinin kaynağı olarak kullanılır. (Varsayılan değer: 0.0)

Giriş parametresi kullanıyorsanız, aşağıdaki şekilde ayar yapmanız gerekir.

Config.InputPerOn = FALSE.

Parametre ve tipi		Veri tipi	Açıklama
Input_PER	IN	Word	Bir Analog giriş, proses değerinin kaynağı olarak kullanılır. (Varsayılan değer: W # 16 # 0) Input_PER parametresini kullanıyorsanız, aşağıdaki şekilde ayar yapmanız gerekir Config.InputPerOn = TRUE.
Disturbance	IN	Real	Bozucu değişken veya ön kontrol değeri
ManualEnable	IN	Bool	Manuel çalışma modunu etkinleştirir veya devre dışı bırakır. (Varsayılan değer: FALSE): <ul style="list-style-type: none"> <li>State = 4 iken, bir FALSE-TRUE kenarı, "manuel mod" u aktif hale getirir. Mod değişmeden kalır.</li> </ul> ManualEnable = TRUE olduğu sürece ModeActivate'te bir pozitif kenar kullanılarak çalışma modunu değiştiremezsiniz veya devreye alma diyalogunu kullanamazsınız. <ul style="list-style-type: none"> <li>TRUE-FALSE kenarı Mod tarafından atanan çalışma modunu etkinleştirir.</li> </ul> Not: Sadece ModeActivate kullanarak çalışma modunu değiştirmenizi öneririz.
ManualValue	IN	Real	Manüel çalışma için çıkış değeri. (Varsayılan değer: 0.0) Config.OutputLowerLimit'ten Config.OutputUpperLimit'e kadar olan değerleri kullanabilirsiniz.
ErrorAck	IN	Bool	ErrorBits ve uyarı çıkışlarını resetler. FALSE-TRUE kenarı
Reset	IN	Bool	Kontrolör yeniden başlatılır. (Varsayılan değer: FALSE): <ul style="list-style-type: none"> <li>FALSE-TRUE kenarı:               <ul style="list-style-type: none"> <li>"inaktif" moduna geçer</li> <li>ErrorBits ve uyarı çıkışları sıfırlar</li> <li>Integral eylemini temizler</li> <li>PID parametrelerini sürdürür</li> </ul> </li> <li>Reset = TRUE olduğu sürece, PID_Compact "Inactive" modunda (State = 0) kalır.</li> <li>TRUE-FALSE kenarı:               <ul style="list-style-type: none"> <li>PID_Compact, Mode parametresinde kaydedilen çalışma moduna geçer.</li> </ul> </li> </ul>
ModeActivate	IN	Bool	PID_Compact Mode parametresinde kaydedilen çalışma moduna geçer. FALSE-TRUE kenarı:
Mode	IN	Int	Arzu edilen PID modu; Aktivate Modunun ön kenarında etkinleştirilir.
ScaledInput	OUT	Real	Ölçekli proses değeri. (Varsayılan değer: 0.0)
Output <sup>1</sup>	OUT	Real	REAL formatında çıkış değeri. (Varsayılan değer: 0.0)
Output_PER <sup>1</sup>	OUT	Word	Analog çıkış değeri. (Varsayılan değer: W#16#0)
Output_PWM <sup>1</sup>	OUT	Bool	Darbe genişlik modülasyonu için çıkış değeri. (Varsayılan değer: FALSE) On ve Off zamanları çıkış değerini oluşturur.
SetpointLimit_H	OUT	Bool	Ayar noktası yüksek sınırı. (Varsayılan değer: FALSE) SetpointLimit_H = TRUE ise, mutlak ayar noktası üst sınırına ulaşılır (Setpoint ≥ Config.SetpointUpperLimit). Ayar noktası Config.SetpointUpperLimit ile sınırlıdır.

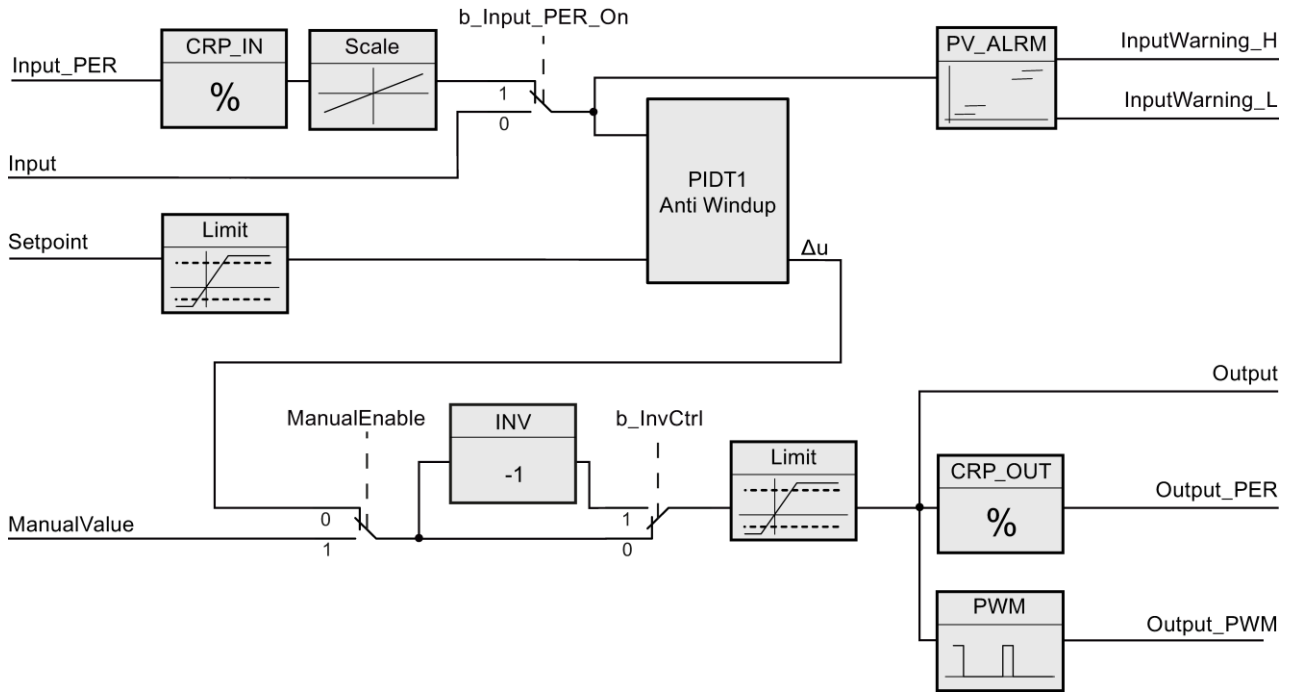
## Teknoloji komutları

## 9.2 PID kontrol

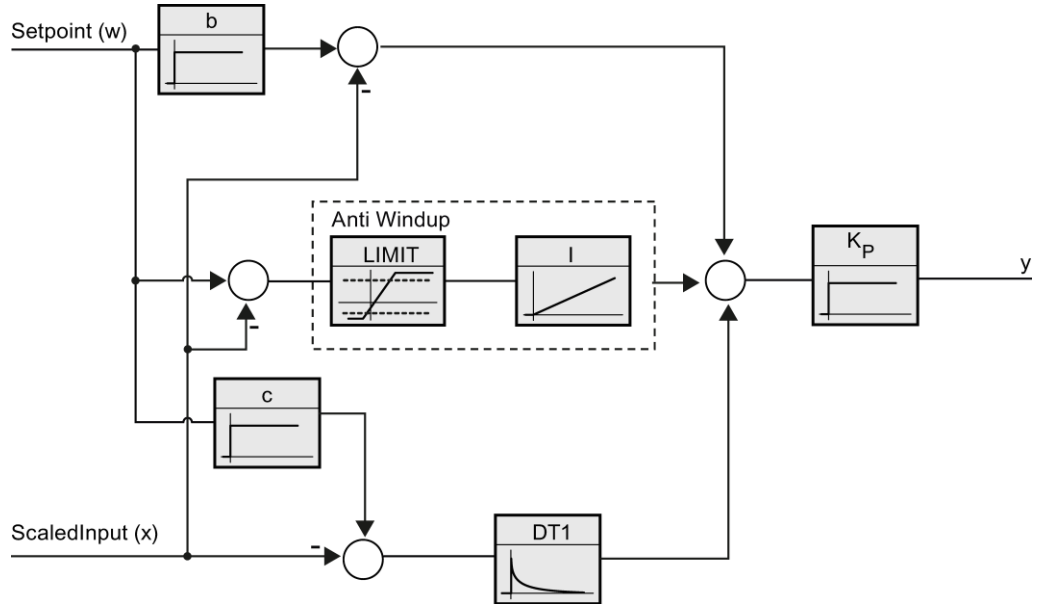
Parametre ve tipi	Veri tipi	Açıklama
SetpointLimit_L	OUT	Bool Ayar noktası alt sınırı. (Varsayılan değer: FALSE) SetpointLimit_L = TRUE, ise, mutlak ayar noktası alt sınırına ulaşılır (Setpoint ≤ Config.SetpointLowerLimit). Ayar noktası Config.SetpointLowerLimit ile sınırlıdır.
InputWarning_H	OUT	Bool InputWarning_H = TRUE ise proses değerine ulaştı veya uyarı yüksek sınırını aştı. (Varsayılan değer: FALSE)
InputWarning_L	OUT	Bool InputWarning_L = TRUE ise proses değerine ulaştı veya uyarı alçak sınırın altına düştü. (Varsayılan değer: FALSE)
State	OUT	Int PID kontrolörün mevcut çalışma modu. (Varsayılan değer: 0) Mode giriş parametresini kullanarak çalışma modunu ve ModeActivate' te pozitif kenarı değiştirebilirsiniz. <ul style="list-style-type: none"> <li>State = 0: Inactive (aktif değil)</li> <li>State = 1: Önayar</li> <li>State = 2: Manuel ince ayar</li> <li>State = 3: Otomatik mod</li> <li>State = 4: Manuel mod</li> <li>State = 5: Hata izlemeli yedek çıkış değeri</li> </ul>
Error	OUT	Bool Hata = TRUE ise, en az bir hata mesajı bu döngüde beklemektedir. (Varsayılan değer: FALSE) Not: V1.x PID'deki hata parametresi, hata kodlarını içeren ErrorBits alanıydı. Şimdi bir hata oluştuğunu belirten bir Boolean bayrağıdır.
ErrorBits	OUT	Dword PID_Compact komutu ErrorBits parametre çizelgesi (Sayfa 426) bekleyen hata mesajlarını tanımlar. (Varsayılan değer: DW # 16 # 0000 (hata yok)). ErrorBits kalıcıdır ve Reset veya ErrorAck'te pozitif bir kenar üzerine sıfırlanır. Not: V1.x'te ErrorBits parametresi Hata parametresi olarak tanımlanmıştı ve yoktu.

<sup>1</sup> Output, Output\_PER ve Output\_PWM parametrelerinin çıkışlarını paralel olarak kullanabilirsiniz.



**PID\_Compact kontrolörün çalışması**


Şekil 9-1 PID\_Compact kontrolörün çalışması



Şekil 9-2 Sarma engelleyicili PIDT1 kontrolör olarak PID\_Compact kontrolörünün çalışması

### 9.2.3 PID\_Compact komutu ErrorBit parametreleri

Birkaç hata bekliyorsa, hata kodları değerleri ikili ekleme vasıtasıyla görüntülenir. 0003 hata kodu görüntüsü, örneğin, 0001 ve 0002 hatalarının beklediğini gösterir.

Çizelge 9- 15 PID\_Compact komutu ErrorBit parametreleri

ErrorBit (DW#16#...)	Açıklama
0000	Hata yok
0001 <sup>1,2</sup>	Giriş parametresi proses değer sınırları dışında. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 <sup>2,3</sup>	İnce ayar sırasında hata. Proses değer salınımı sürdürülemedi.
0004 <sup>4</sup>	Ön ayar sırasında hata. Proses değer salınımı sürdürülemedi.
0008 <sup>4</sup>	Ön ayar başlangıcında hata. Proses değeri ayar noktası değerine çok yakın. İnce ayarı başlatınız.
0010 <sup>4</sup>	Ayar noktası ayar sırasında değiştirildi. Not: CancelTuningLevel etiketindeki ayar noktası üzerinde izin verilen dalgalanmayı ayarlayabilirsiniz.
0020	Ön-ayara, ince ayar sırasında izin verilmez. Not: Hata oluşmadan önce ActivateRecoverMode = TRUE olursa, PID_Compact ince ayar modunda kalır.
0080 <sup>4</sup>	Ön ayar sırasında hata. Çıkış değeri sınırlarının yanlış konfigürasyonu. Çıkış değeri sınırlarının doğru yapılandırıldığını kontrol ediniz ve kontrol mantığını eşleştiriniz.
0100 <sup>4</sup>	İnce ayar sırasında meydana gelen hata geçersiz parametrelere neden oldu.
0200 <sup>2,3</sup>	Giriş parametresinde geçersiz değer: Değer, geçersiz sayı biçimine sahip.
0400 <sup>2,3</sup>	Çıkış değerinin hesaplanması başarısız oldu. PID parametrelerini kontrol ediniz.
0800 <sup>1,2</sup>	Örnekleme zamanı hatası: PID_Compact döngüsel kesme OB'sinin örnekleme süresi içinde çağrılmaz.
1000 <sup>2,3</sup>	Setpoint parametresinde geçersiz değer: Değer geçersiz sayı biçimine sahip.
10000	ManualValue parametresinde geçersiz değer: Değer, geçersiz sayı biçimine sahip. Not: Hata oluşmadan önce ActivateRecoverMode = TRUE olursa, PID_Compact çıkış değeri olarak SubstituteOutput kullanır. En kısa sürede ManualValue parametresinde geçerli bir değer atadığınızda, PID_Compact bunu çıkış değeri olarak kullanır.

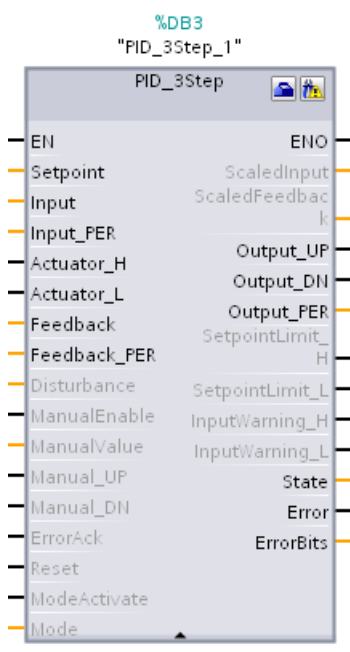
ErrorBit (DW#16#...)	Açıklama
20000	<p>SubstituteValue etiketinde geçersiz değer: Değer, geçersiz sayı biçimine sahip.</p> <p>PID_Compact, çıkış değeri olarak çıkış değeri alt sınırı kullanır.</p> <p>Not: Hata oluşmadan önce otomatik mod aktif ise. ActivateRecoverMode = TRUE ve artık hata beklemez, PID_Compact otomatik moda geri döner.</p>
40000	<p>Disturbance parametresinde geçersiz değer: Değer, geçersiz bir sayı biçimine sahip.</p> <p>Not: Hata oluşmadan önce Otomatik mod aktif ve ActivateRecoverMode = FALSE ise Disturbance sıfıra ayarlanır. PID_Compact otomatik modda kalır.</p> <p>Not: Hata oluşmadan önce Ön-ayar veya ince ayar modu etkin ve ActivateRecoverMode = TRUE ise, PID_Compact, Mode parametresinde kaydedilen çalışma moduna geçer. Mevcut fazda Disturbance, çıkış değeri üzerinde herhangi bir etkiye sahip değildir, ayar iptal edilmez.</p>

- <sup>1</sup> Not: Hata oluşmadan önce otomatik mod aktif ve ActivateRecoverMode = TRUE ise PID\_Compact otomatik modda kalır.
- <sup>2</sup> Not: Hata oluşmadan önce Ön-ayar veya ince ayar modu aktif ve ActivateRecoverMode = TRUE ise PID\_Compact, Mode parametresinde kaydedilen çalışma moduna geçer.
- <sup>3</sup> Not: Hata oluşmadan önce otomatik mod aktif ve ActivateRecoverMode = TRUE ise PID\_Compact yapılandırılmış yedek çıkış değerini verir. Hatanın bekleme işlemi biter bitmez, PID\_Compact otomatik moduna geri döner.
- <sup>4</sup> Not: Hata oluşmadan önce ActivateRecoverMode = TRUE ise, PID\_Compact, ayarlamayı iptal eder ve Mode parametresinde kaydedilen çalışma moduna geçer.

### 9.2.4 PID\_3Step komutu

PID\_3Step komutu, motor kontrollü vanalar ve tahrik mekanizmaları için optimize edilmiş öz-ayar yeteneklerine sahip bir PID kontrolörü yapılandırır.

Çizelge 9- 16 PID\_3Step komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows the PID_3Step block with various inputs and outputs. Inputs include EN, Setpoint, Input, Input_PER, Actuator_H, Actuator_L, Feedback, Feedback_PER, Disturbance, ManualEnable, ManualValue, Manual_UP, Manual_DN, ErrorAck, Reset, ModeActivate, and Mode. Outputs include ENO, ScaledInput, ScaledFeedback, Output_UP, Output_DN, Output_PER, SetpointLimit_H, SetpointLimit_L, InputWarning_H, InputWarning_L, State, Error, and ErrorBits.</p>	<pre>"PID_3Step_1" (   SetpoInt:=_real_in_,   Input:=_real_in_,   ManualValue:=_real_in_,   Feedback:=_real_in_,   InputPer:=_word_in_,   FeedbackPer:=_word_in_,   Disturbance:=_real_in_,   ManualEnable:=_bool_in_,   ManualUP:=_bool_in_,   ManualDN:=_bool_in_,   ActuatorH:=_bool_in_,   ActuatorL:=_bool_in_,   ErrorAck:=_bool_in_,   Reset:=_bool_in_,   ModeActivate:=_bool_in_,   Mode:=_int_in_,   ScaledInput=&gt;_real_out_,   ScaledFeedback=&gt;_real_out_,   ErrorBits=&gt;_dword_out_,   OutputPer=&gt;_word_out_,   State=&gt;_int_out_,   OutputUP=&gt;_bool_out_,   OutputDN=&gt;_bool_out_,   SetpoIntLimitH=&gt;_bool_out_,   SetpoIntLimitL=&gt;_bool_out_,   InputWarningH=&gt;_bool_out_,   InputWarningL=&gt;_bool_out_,   Error=&gt;_bool_out_,   ErrorBits=&gt;_dword_out_);</pre>	<p>PID_3Step motor kontrollü vanalar ve tahrik mekanizmaları için optimize edilmiş öz-ayar yeteneklerine sahip bir PID kontrolörü yapılandırır. Bu iki Boolean çıkışı sağlar.</p> <p>PID_3Step sarma engelleyici ve ağırlıklı P- ve D-bileşenlerine sahip bir PID kontrolördür.</p>

- 1 Komutu yerleştirdiğinizde STEP 7 otomatik olarak teknoloji nesnesi ve kopya DB oluşturur. Kopya DB teknoloji nesnesinin parametrelerini içerir.
- 2 SCL örneğinde, "PID\_3STEP\_1" kopya DB'nin ismidir.

Çizelge 9- 17 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
Setpoint	IN	Real
Input	IN	Real

PID kontrolörün otomatik modda ayar noktası. (Varsayılan değer: 0.0)

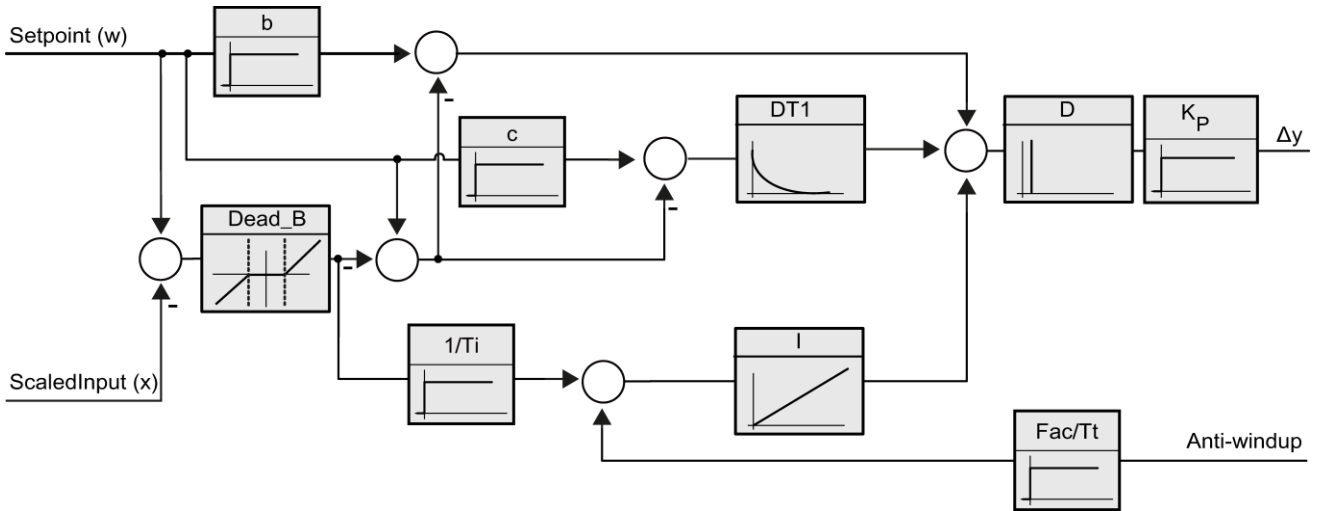
Kullanıcı programının bir etiketi, proses değerinin kaynağı olarak kullanılır. (Varsayılan değer: 0.0)

Giriş parametresi kullanıyorsanız, Config.InputPerOn = FALSE olarak ayarlamalısınız.

Parametre ve tipi		Veri tipi	Açıklama
Input_PER	IN	Word	Bir Analog giriş, proses değerinin kaynağı olarak kullanılır. (Varsayılan değer: W # 16 # 0) Input_PER parametresini kullanıyorsanız, Config.InputPerOn = TRUE olarak ayar yapmalısınız
Actuator_H	IN	Bool	Yüksek son durağı için vananın dijital pozisyon geri beslemesi Actuator_H = TRUE ise, vana yüksek son durağındadır ve artık bu doğrultuda hareket etmez. (Varsayılan değer: FALSE)
Actuator_L	IN	Bool	Alçak son durağı için vananın dijital pozisyon geri beslemesi Actuator_L = TRUE ise, vana alçak son durağındadır ve artık bu doğrultuda hareket etmez. (Varsayılan değer: FALSE)
Feedback	IN	Real	Vananın pozisyon geri beslemesi. (Varsayılan değer: 0.0) Feedback parametresi kullanıyorsanız, Config.FeedbackPerOn = FALSE ayarlamanız gerekir.
Feedback_PER	IN	Int	Vananın pozisyonunun analog geri beslemesi. (Varsayılan değer: W#16#0) Feedback_PER parametresi kullanıyorsanız, Config.FeedbackPerOn = TRUE ayarlamanız gerekir. Feedback_PER aşağıdaki etiketlere göre ölçeklendirilir: <ul style="list-style-type: none"> <li>• Config.FeedbackScaling.LowerPointIn</li> <li>• Config.FeedbackScaling.UpperPointIn</li> <li>• Config.FeedbackScaling.LowerPointOut</li> <li>• Config.FeedbackScaling.UpperPointOut</li> </ul>
Disturbance	IN	Real	Bozucu değişken veya ön kontrol değeri
ManualEnable	IN	Bool	Manuel çalışma modunu etkinleştirir veya devre dışı bırakır. (Varsayılan değer: FALSE): <ul style="list-style-type: none"> <li>• State = 4 iken, bir FALSE-TRUE kenarı, "manuel mod" u aktif hale getirir. Mod değişmeden kalır.</li> </ul> ManualEnable = TRUE olduğu sürece ModeActivate'te bir pozitif kenar kullanılarak çalışma modunu değiştiremezsiniz veya devreye alma diyalogunu kullanamazsınız. <ul style="list-style-type: none"> <li>• TRUE-FALSE kenarı Mod tarafından atanan çalışma modunu etkinleştirir.</li> </ul> Not: Sadece ModeActivate kullanarak çalışma modunu değiştirmenizi öneririz.
ManualValue	IN	Real	Manuel çalışma için proses değeri. (Varsayılan değer: 0.0) Manuel modunda, vananın mutlak konumunu belirtiniz. ManualValue, yalnızca OutputPer kullanmanız durumunda ya da pozisyon geri besleme varsa değerlendirilir.

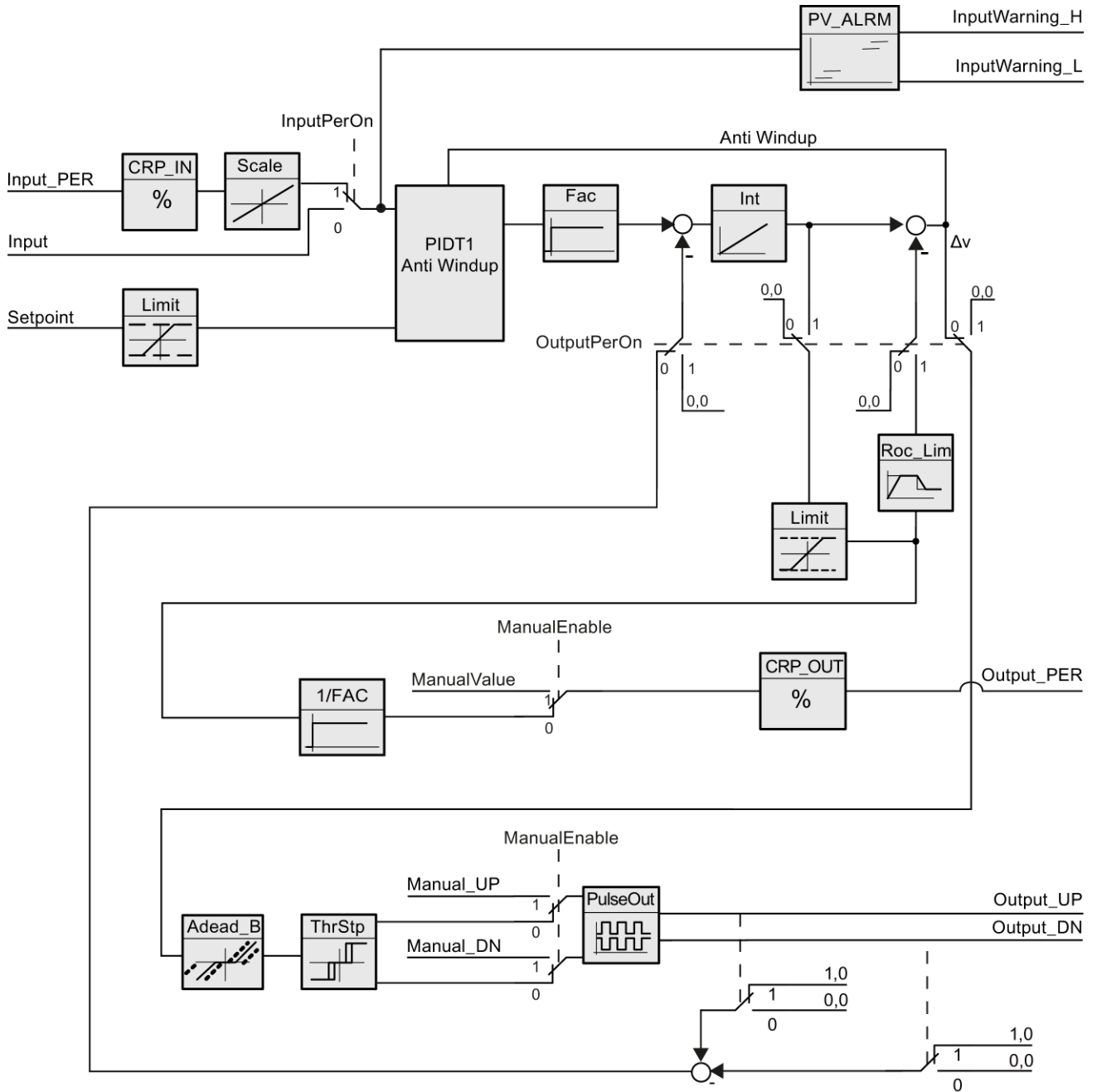
Parametre ve tipi		Veri tipi	Açıklama
ManualUP	IN	Bool	<ul style="list-style-type: none"> <li>Manual_UP = TRUE: <ul style="list-style-type: none"> <li>Output_PER veya pozisyon geri besleme kullansanız bile vana açılır. Üst son durağına ulaşıldığında valf artık hareket etmez.</li> <li>Ayrıca bakınız Config.VirtualActuatorLimit</li> </ul> </li> <li>Manual_UP = FALSE: <ul style="list-style-type: none"> <li>Output_PER veya pozisyon geri besleme kullansanız bile, vana ManualValue'ya hareket eder. Aksi takdirde, vana artık hareket etmez.</li> </ul> </li> </ul> <p>Not: Manual_UP ve Manual_DN aynı anda TRUE ayarlanmış ise, vana hareket etmez.</p>
ManualDN	IN	Bool	<ul style="list-style-type: none"> <li>Manual_DN = TRUE: <ul style="list-style-type: none"> <li>Output_PER veya pozisyon geri besleme kullansanız bile vana açılır. Üst son durağına ulaşıldığında valf artık hareket etmez.</li> <li>Ayrıca bakınız Config.VirtualActuatorLimit</li> </ul> </li> <li>Manual_DN = FALSE: <ul style="list-style-type: none"> <li>Output_PER veya pozisyon geri besleme kullansanız bile, vana ManualValue'ya hareket eder. Aksi takdirde, vana artık hareket etmez.</li> </ul> </li> </ul>
ErrorAck	IN	Bool	ErrorBits ve uyarı çıkışlarını sıfırlar. FALSE-TRUE kenarı
Reset	IN	Bool	<p>Kontrolörü yeniden başlatır. (Varsayılan değer: FALSE):</p> <ul style="list-style-type: none"> <li>FALSE-TRUE kenarı: <ul style="list-style-type: none"> <li>"inaktif" moduna geçer</li> <li>ErrorBits ve uyarı çıkışlarını sıfırlar</li> <li>Integral eylemini temizler</li> <li>PID parametrelerini sürdürür</li> </ul> </li> <li>Reset = TRUE olduğu sürece, PID_3Step "Inactive" modunda (State = 0) kalır.</li> <li>TRUE-FALSE kenarı: <ul style="list-style-type: none"> <li>PID_3Step, Mode parametresinde kaydedilen çalışma moduna geçer.</li> </ul> </li> </ul>
ModeActivate	IN	Bool	The PID_3Step, Mode parametresinde kaydedilen moda geçer. FALSE-TRUE kenarı:
Mode	IN	Int	Arzu edilen PID modu; Aktivate Input Modunun ön kenarında etkinleştirilir.
ScaledInput	OUT	Real	Ölçekli proses değeri
ScaledFeedback	OUT	Real	Ölçekli vana pozisyon geri beslemesi Not: Konum geri beslemesi olmayan bir tahrik mekanizması için ScaledFeedback tarafından gösterilen tahrik mekanizması konumu, çok kesin değildir. ScaledFeedback bu durumda sadece mevcut konumun kaba tahmini için kullanılabilir.
Output_UP	OUT	Bool	Vanayı açmak için dijital çıkış değeri. (Varsayılan değer: FALSE) Config.OutputPerOn = FALSE ise, Output_UP parametresi kullanılır.
Output_DN	OUT	Bool	Vanayı kapatmak için dijital çıkış değeri. (Varsayılan değer: FALSE) Config.ÇıkışPerOn = FALSE ise, Output_DN parametresi kullanılır.

Parametre ve tipi		Veri tipi	Açıklama
Output_PER	OUT	Word	Analog çıkış değeri. Config.ÇıkışPerOn = TRUE, Output_PER parametresi kullanılır.
SetpointLimitH	OUT	Bool	Ayar noktası yüksek sınırı. (Varsayılan değer: FALSE) SetpointLimit_H = TRUE ise, mutlak ayar noktası üst sınırına ulaşılır (Setpoint $\geq$ Config.SetpointUpperLimit). Not Ayar noktası (Setpoint $\geq$ Config.SetpointUpperLimit) ile sınırlıdır
SetpointLimitL	OUT	Bool	Ayar noktası alt sınırı. (Varsayılan değer: FALSE) SetpointLimit_L = TRUE ise, mutlak ayar noktası alt sınırına ulaşılır (Setpoint $\leq$ Config.SetpointLowerLimit). Not Ayar noktası (Setpoint $\geq$ Config.SetpointLowerLimit) ile sınırlıdır
InputWarningH	OUT	Bool	InputWarning_H = TRUE ise proses değerine ulaştı veya uyarı yüksek sınırını aştı. (Varsayılan değer: FALSE)
InputWarningL	OUT	Bool	InputWarning_L = TRUE ise proses değerine ulaştı veya uyarı alçak sınırın altına düştü. (Varsayılan değer: FALSE)
State	OUT	Int	PID kontrolörün mevcut çalışma modu. (Varsayılan değer: 0) Mode giriş parametresini kullanarak çalışma modunu ve ModeActivate te pozitif kenarı değiştirebilirsiniz: <ul style="list-style-type: none"> <li>• State = 0: Inactive(aktif değil)</li> <li>• State = 1: Önayar</li> <li>• State = 2: Manuel ince ayar</li> <li>• State = 3: Otomatik mod</li> <li>• State = 4: Manuel mod</li> <li>• State = 5: Yedek çıkış değeri yaklaşımı</li> <li>• State = 6: Geçiş süresi ölçümü</li> <li>• State = 7: Hata izleme</li> <li>• State = 8: Hata izlemeli yedek çıkış değeri yaklaşımı \</li> <li>• State = 10: Son durak sinyalleri olmaksızın manuel mod</li> </ul>
Error	OUT	Bool	Hata = TRUE ise, en az bir hata mesajı beklemektedir. (Varsayılan değer: FALSE) Not: V1.x PID'deki hata parametresi, hata kodlarını içeren ErrorBits alanıydı. Şimdi bir hata oluştuğunu belirten bir Boolean bayrağıdır.
ErrorBits	OUT	Dword	The PID_3Step komutu ErrorBits parametre çizelgesi (Sayfa 435) bekleyen hata mesajlarını tanımlar. (Varsayılan değer: DW # 16 # 0000 (hata yok)). ErrorBits kalıcıdır ve Reset veya ErrorAck'te pozitif bir kenar üzerine sıfırlanır. Not: V1.x'te ErrorBits parametresi Hata parametresi olarak tanımlanmıştı ve yoktu.

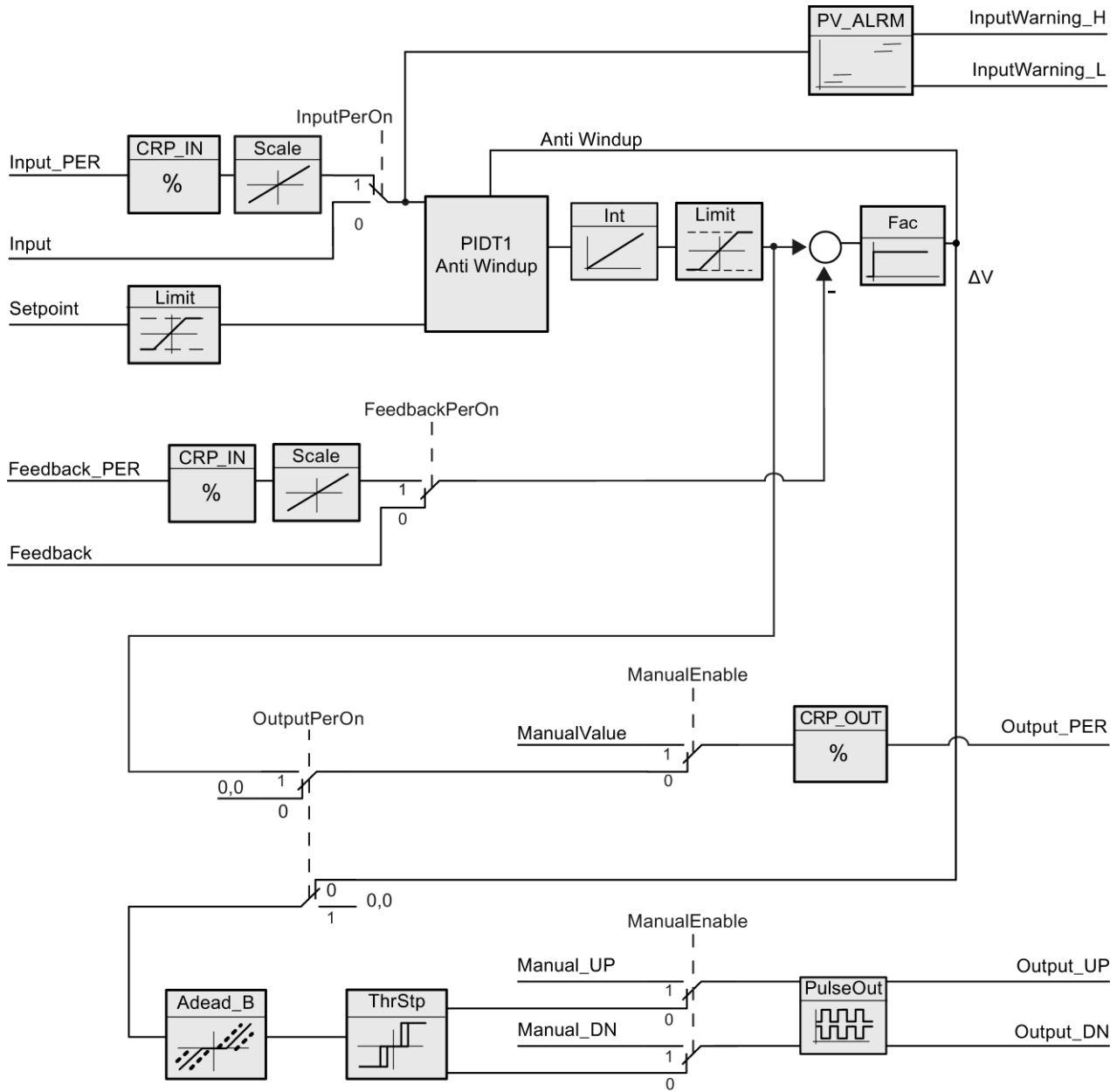


Şekil 9-3 Sarma engelleyicili PID T1 kontrolör olarak PID\_3Step kontrolörünün çalışması





Şekil 9-4 PID\_3Step denetleyicinin pozisyon geri beslemesiz olarak çalışması



Şekil 9-5 PID\_3Step kontrolörün etkinleştirilmiş pozisyon geri beslemesiyle çalışması

## 9.2.5 PID\_3Step komutu ErrorBit parametreleri

Birkaç hata bekliyorsa, hata kodları değerleri ikili ilave vasıtasıyla görüntülenir. 0003 hata kodu görüntüsü, örneğin, 0001 ve 0002 hatalarının beklediğini gösterir.

Çizelge 9- 18 PID\_3STEP komutu ErrorBit parametreleri

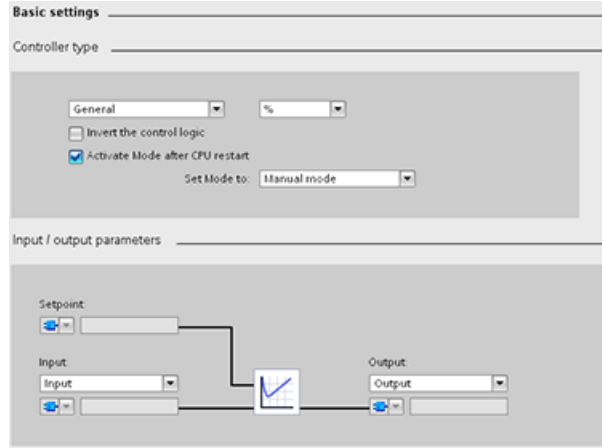
ErrorBit (DW#16#...)	Açıklama
0000	Hata yok
0001 <sup>1,2</sup>	Giriş parametresi proses değer sınırları dışında. Input > Config.InputUpperLimit Input < Config.InputLowerLimit
0002 <sup>2,3</sup>	Input_PER parametresinde geçersiz değer. Hatanın analog girişte bekleyip beklemediğini kontrol ediniz.
0004 <sup>4</sup>	İnce ayar sırasında hata. Proses değer salınımı sürdürülemedi.
0010 <sup>4</sup>	Ayar noktası ayar sırasında değiştirildi. Not: CancelTuningLevel etiketindeki ayar noktası üzerinde izin verilen dalgalanmayı ayarlayabilirsiniz.
0020	Ön-ayara, ince ayar sırasında izin verilmez. Not: Hata oluşmadan önce ActivateRecoverMode = TRUE olursa, PID_3Step ince ayar modunda kalır.
0080 <sup>4</sup>	Ön ayar sırasında hata. Çıkış değeri sınırlarının yanlış konfigürasyonu. Çıkış değeri sınırlarının doğru yapılandırıldığını kontrol ediniz ve kontrol mantığını eşleştiriniz.
0100 <sup>4</sup>	İnce ayar sırasında meydana gelen hata geçersiz parametrelere neden oldu.
0200 <sup>2,3</sup>	Giriş parametresinde geçersiz değer: Değer, geçersiz sayı biçimine sahip.
0400 <sup>2,3</sup>	Çıkış değerinin hesaplanması başarısız oldu. PID parametrelerini kontrol ediniz.
0800 <sup>1,2</sup>	Örnekleme zamanı hatası: PID_3Step döngüsel kesme OB'sinin örnekleme süresi içinde çağrılmaz.
1000 <sup>2,3</sup>	Setpoint parametresinde geçersiz değer: Değer geçersiz sayı biçimine sahip.
2000 <sup>1,2,5</sup>	Feedback_PER parametresinde geçersiz değer. Hatanın analog girişte bekleyip beklemediğini kontrol ediniz.
4000 <sup>1,2,5</sup>	Feedback_PER parametresinde geçersiz değer: Değer geçersiz sayı biçimine sahip.
8000 <sup>1,2</sup>	Dijital pozisyon geri beslemesi sırasında hata. Actuator_H = TRUE ve Actuator_L = TRUE. Tahrik edici, yedek çıkış değerine hareket edemez ve mevcut konumda kalır. Manuel mod bu durumda mümkün değildir. Bu durumdan tahrik ediciyi taşımak için "Tahrik edici son durağı" (Config.ActuatorEndStopOn = FALSE)' nı devre dışı bırakmanız gerekir veya son durak sinyalleri olmadan manuel moda (Mod = 10) geçmeniz gerekir.

ErrorBit (DW#16#...)	Açıklama
10000	ManualValue parametresinde geçersiz değer: Değer, geçersiz sayı biçimine sahip. Tahrik edici manuel değere taşınamaz ve mevcut konumunda kalır. ManualValue'da geçerli bir değer atavınız veya Manual_UP ve Manual_DN ile manuel modda tahrik ediciyi taşıyınız.
20000	SavePosition etiketi geçersiz değer: Değer geçersiz sayı biçimine sahip. Tahrik edici, yedek çıkış değerine hareket edemez ve mevcut konumda kalır.
40000	Disturbance parametresinde geçersiz değer: Değer geçersiz sayı biçimine sahip. Not: Hata oluşmadan önce Otomatik mod aktif ve ActivateRecoverMode = FALSE ise Disturbance sıfıra ayarlanır. PID_3Step otomatik modda kalır. Not: Hata oluşmadan önce Ön-ayar veya ince ayar modu etkin ve ActivateRecoverMode = TRUE ise, PID_3Step, Mod parametresinde kaydedilen çalışma moduna geçer. Mevcut fazda Disturbance, çıkış değeri üzerinde herhangi bir etkiye sahip değildir, ayar iptal edilmez. Hatanın, geçiş süresi ölçme sırasında hiçbir etkisi yoktur.

- 1 Not: Hata oluşmadan önce otomatik mod aktif ve ActivateRecoverMode = TRUE ise PID\_3Step otomatik modda kalır.
- 2 Not: Hata oluşmadan önce Ön-ayar, ince ayar veya geçiş süresi ölçüm modu aktif ve ActivateRecoverMode = TRUE ise PID\_3Step, Mode parametresinde kaydedilen çalışma moduna geçer.
- 3 Not: Hata oluşmadan önce otomatik mod aktif ve ActivateRecoverMode = TRUE ise PID\_3Step "hata izleme ile yaklaşım yedek çıkış değeri" veya "Hata izleme" moduna geçer. Hatanın bekleme işlemi biter bitmez, PID\_3Step otomatik moduna geri döner.
- 4 Not: Hata oluşmadan önce ActivateRecoverMode = TRUE ise, PID\_3Step, ayarlamayı iptal eder ve Mode parametresinde kaydedilen çalışma moduna geçer.
- 5 Tahrik edici yedek çıkış değerine hareket edemez ve mevcut pozisyonunda kalır. Manuel modda, Tahrik edici pozisyonunu ManualValue ile değil yalnızca Manual\_UP ve Manual\_DN ile değiştirebilirsiniz.

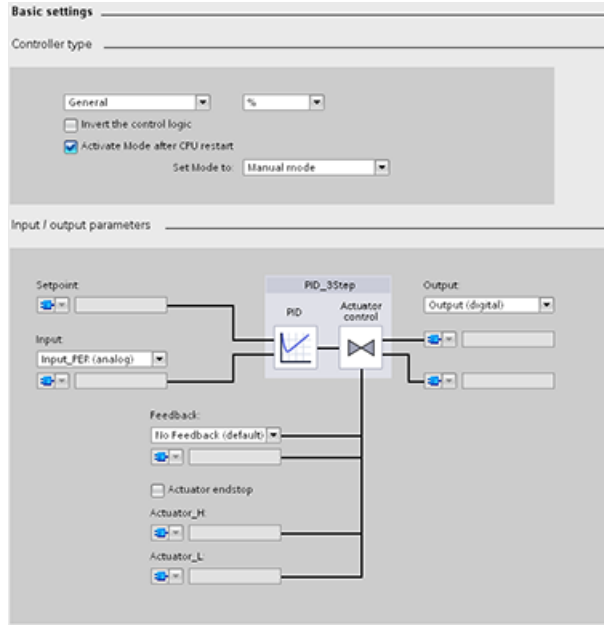
## 9.2.6 PID kontrolörün yapılandırılması

Teknoloji nesnesi parametreleri, PID kontrolör çalışmasını belirler. Yapılandırma editörünü açmak için yandaki simgeyi kullanınız.

Çizelge 9- 19 PID\_Compact komut için örnek yapılandırma ayarları

Ayarlar	Açıklama	
Temel	Kontrolör tipi	Mühendislik birimlerini seç.
	Kontrol mantığını çevir	Bir ters-aksiyon PID döngüsünün seçimini sağlar. <ul style="list-style-type: none"> <li>Seçili değilse, PID döngü doğrudan-aksiyon modundadır ve eğer giriş değeri &lt;ayar noktası ise PID döngüsünün çıkışı artar.</li> <li>Seçili ise, giriş değeri &gt; ayar noktası ise PID döngüsünün çıkışı artar.</li> </ul>
	CPU'yu yeniden başlattıktan sonra son modunu etkinleştiriniz	Resetten sonra veya bir giriş limitini aştığınızda ve yeniden geçerli aralığa döndüğünde PID döngüsünü yeniden başlatınız.
	Giriş	Proses değeri için giriş parametresini veya (analog için) Input_PER parametresini seçiniz. Input_PER bir analog giriş modülünden doğrudan gelebilir.
	Çıkış	Proses değeri için çıkış parametresini veya (analog için) Output_PER parametresini seçiniz. Output_PER bir analog çıkış modülüne doğrudan gidebilir.
Proses değer		Proses değerinin hem aralığını hem de sınırlarını ölçekler. Proses değeri alt sınırın altında veya üst sınırın üstüne çıkarsa, PID döngü aktif olmayan (inactive) moda geçer ve çıkış değerini 0 olarak ayarlar. Input_PER kullanmak için analog proses değerini (giriş değeri) ölçeklemeniz gerekir.



Çizelge 9- 20 PID\_3Step komut için örnek yapılandırma ayarları

Ayarlar	Açıklama	
Temel	Kontrolör tipi	Mühendislik birimlerini seçiniz.
	Kontrol mantığını çevir	Bir ters-aksiyon PID döngüsünün seçimini sağlar. <ul style="list-style-type: none"> <li>Seçili değilse, PID döngü doğrudan-aksiyon modundadır ve eğer giriş değeri &lt;ayar noktası ise PID döngüsünün çıkışı artar.</li> <li>Seçili ise, giriş değeri &gt; ayar noktası ise PID döngüsünün çıkışı artar.</li> </ul>
	CPU'yu yeniden başlattıktan sonra modu etkinleştiriniz	Resetten sonra veya bir giriş limitini aştığınızda ve yeniden geçerli aralığa döndüğünde PID döngüsünü yeniden başlatınız. Mod ayarlayınız: Tekrar başlamadan sonra kullanıcı PID'nin gitmesini istediği modu tanımlar.
	Giriş	Proses değeri için giriş parametresini veya (analog için) Input_PER parametresini seçiniz. Input_PER bir analog giriş modülünden doğrudan gelebilir.
	Çıkış	Proses değeri için çıkış parametresini veya (analog için) Output_PER parametresini seçiniz. Output_PER bir analog çıkış modülüne doğrudan gidebilir.
	Geri besleme	PID döngüsüne geri dönen cihaz durumu tipini seçer: <ul style="list-style-type: none"> <li>Geri besleme yok (varsayılan)</li> <li>Geri besleme</li> <li>Feedback_PER</li> </ul>
Proses değeri		Proses değerinin hem aralığını hem de sınırlarını ölçekler. Proses değeri alt sınırın altında veya üst sınırın üstüne çıkarsa, PID döngü aktif olmayan (inactive) moda geçer ve çıkış değerini 0 olarak ayarlar. Input_PER kullanmak için analog proses değerini (giriş değeri) ölçeklemeniz gerekir.
Tahrik edici	Motor geçiş süresi	Vana için açmadan kapamaya geçiş süresini ayarlar. (Bu değeri veri föyüne veya vananın plakası üzerine yerleştiriniz.)
	Minimum ON zamanı	Vana için minimum hareket süresini ayarlar. (Bu değeri veri föyüne veya vananın plakası üzerine yerleştiriniz.)

Ayarlar	Açıklama	
Minimum OFF süresi	Vana için minimum durma süresini ayarlar. (Bu değeri veri fôyüne veya vananın plakası üzerine yerleştiriniz.)	
Hataya tepki	PID döngü reset olduğunda veya bir hata algılandığında vananın davranışını tanımlar. Bir vedek konum kullanmayı seçerseniz. "Güvenlik pozisyonu" nu diriniz. Analog seri besleme veya analog çıkış için çıkışla ilgili üst veya alt sınır arasında bir değer seçiniz. Dijital çıkışlar için yalnızca % 0 (kapalı) veya (açık) % 100 seçebilirsiniz.	
Feedback Pozisyonunu ölçekle <sup>1</sup>	<ul style="list-style-type: none"> <li>"Yüksek son durağı" ve "Alt son durağı" maksimum pozitif konumunu (tam açık) ve maksimum negatif konumunu (tam kapalı) tanımlar. "Yüksek son durağı" "Alt son durağı" ndan daha büyük olmalıdır.</li> <li>"Yüksek limit proses değeri" ve "Düşük limit proses değeri", otomatik mod veya ayarlama sırasında vana konumlarının üst ve alt pozisyonlarını tanımlar.</li> <li>"FeedbackPER" ("düşük" ve "yüksek"), vana pozisyonunun analog seri beslemesini tanımlar. "FeedbackPER Yüksek", "FeedbackPER Düşük" ten daha büyük olmalıdır.</li> </ul>	
Gelişmiş	Proses değerini izle	Proses değeri için uyarılacak yüksek ve düşük limitleri belirler.
	PID parametreleri	Kullanıcı isterse, bu pencerede kendi PID ayar parametrelerini girebilir. " Enable Manual Entry " onay kutusu, buna izin vermek için kontrol edilmelidir.

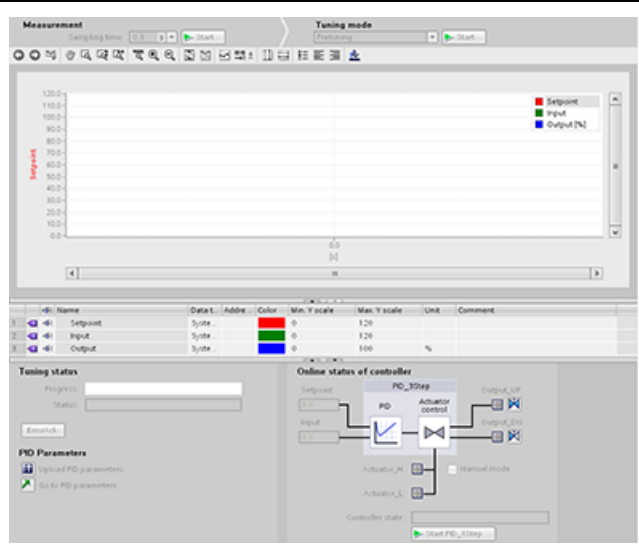
<sup>1</sup> " Feedback Pozisyonunu ölçekle ", yalnızca "Temel" ayarlarda "Geri besleme" yi etkinleştirirseniz düzenlenebilir.

## 9.2.7 PID kontrolörün devreye alınması

Yol verme anında oto-ayarlama için veya operasyon sırasında otomatik ayarlama için PID kontrolörünü yapılandırmak üzere devreye alma editörünü kullanınız. Devreye alma editörünü açmak için komut ya da proje gezgini üzerindeki simgeyi tıklayınız.



Çizelge 9- 21 Başlatma ekranı (PID\_3Step) örneği



- Ölçme: gerçek-zamanlı eğilimde ayar noktasını, proses değeri (giriş değeri) ve çıkış değerini görüntülemek için örnek süresini giriniz ve "Başlat" düğmesini tıklayınız.
- Ayarlama modu: PID döngüyü ayarlamak için "Pre-tuning" ya da " Fine tuning" (manuel) den birini seçiniz ve Başlat düğmesini tıklayınız. PID kontrolör sistem tepkisini hesaplamak ve süreleri güncellemek için çoklu aşamalardan geçer. Uygun ayarlama parametreleri bu değerlerden hesaplanır.

Ayarlama sürecinin tamamlanmasından sonra, devreye alma editörü "PID Parameters" bölümündeki "Upload PID parameters" düğmesini tıklayarak yeni parametreleri saklayabilirsiniz.

Ayarlama sırasında bir hata oluşursa, PID çıkış değeri 0'a gider. PID modu daha sonra " inactive " moduna ayarlanır. Durum, hata gösterir.

## PID başlatma değeri kontrolü

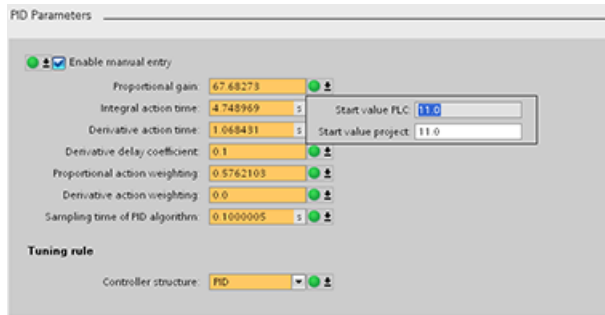
PID kontrolör davranışını çevrimiçi modda optimize edilebilecek şekilde PID yapılandırma parametrelerinin gerçek değerlerini düzenleyebilirsiniz.

PID kontrolörünüz ve bunun "Yapılandırma" nesnesi için "Teknoloji nesnelere" ni açınız. Başlatma değeri kontrolüne erişmek için diyalog kutusunun sol üst köşesindeki "gözlük simgesini" tıklayınız:



Aşağıdaki şekilde gösterildiği gibi şimdi PID kontrolörünüzün yapılandırma parametrelerinin herhangi birinin değerini değiştirebilirsiniz.

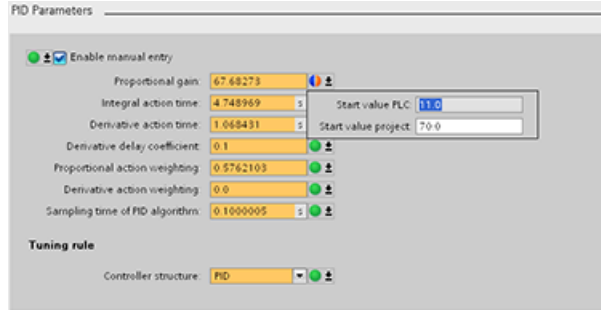
Her bir parametrenin PLC başlatma değeri (çevrimiçi) ve proje başlatma değerini (çevrimdışı) gerçek değeri ile karşılaştırabilirsiniz. Bu Teknoloji nesnesi veri bloğunun (TO-DB) çevrimiçi / çevrimdışı farklılıklarını karşılaştırmak ve PLC'nin sonraki Dur-Başlat geçişindeki mevcut değerler olarak kullanılacak değerler hakkında bilgi sahibi olmak için gereklidir. Buna ek olarak, bir karşılaştırma simgesi, çevrimiçi / çevrimdışı farklılıkların kolayca tanımlanmasına yardımcı olmak için görsel bir gösterge verir:



Yukarıdaki şekil, çevrimiçi ve çevrimdışı projeler arasında hangi değer farklı olduğunu gösteren karşılaştırma simgeleri ile PID parametre ekranını gösterir. Yeşil bir simge, değerlerin aynı olduğunu gösterir; mavi / turuncu simge değerlerin farklı olduğunu gösterir.



Avrıcı, her bir parametrenin proje (çevrimdışı) başlatma değeri ve PI C (çevrimci) başlatma değerini gösteren küçük bir pencere açmak için aşağı ok düğmesine sahip parametre butonuna tıklayınız:



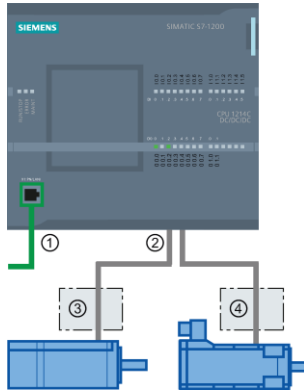
## 9.3 Hareket kontrolü

CPU, step motorlar ve servo motorların darbe ara yüzü ile çalışması için hareket kontrol işlevselliği sağlar. Hareket kontrol işlevselliği sürücülerin kontrol ve izleme işlemini devralır.

- "Eksen" teknoloji nesnesi, mekanik sürücü verileri, sürücü ara yüzü, dinamik parametreler ve diğer sürücü özelliklerini yapılandırır.
- Sürücüyü kontrol etmek için CPU'nun darbe ve yön çıkışlarını yapılandırınız.
- Kullanıcı programınız, eksen kontrol etmek ve hareket görevlerini başlatmak için hareket kontrol komutlarını kullanır.
- CPU ve programlama cihazı arasında çevrimiçi bağlantı kurmak için PROFINET ara yüzünü kullanınız. CPU'nun çevrimiçi işlevlerine ek olarak, hareket kontrolü için ilave devreye alma ve teşhis fonksiyonları kullanılabilir.

### Not

Hareket kontrol yapılandırmasında yaptığınız değişiklikler ve RUN modunda yükleme, CPU STOP'tan RUN moduna geçene kadar etkili olmaz.



- ① PROFINET
- ② Darbe ve yön çıkışları
- ③ Step motor için güç bölümü
- ④ Servo motor için güç bölümü

CPU S7-1200'ün DC/DC/DC varyantları, sürücüleri doğrudan kontrol için yerleşik çıkışlara sahiptir. CPU'nun röle varyantları, sürücü kontrolü için DC çıkışlara sahip sinyal kartı gerektirir.

Bir sinyal kartı (SB) birkaç ek I/O noktası ilave etmek için yerleşik I/O'yu genişletir. İki dijital çıkışa sahip bir SB, bir motoru kontrol etmek için darbe ve yön çıkışları olarak kullanılabilir. Dört dijital çıkışa sahip bir SB, iki motoru kontrol etmek için darbe ve yön çıkışları olarak kullanılabilir. Gömülü röle çıkışları, motorları kontrol etmek için darbe çıkışları olarak kullanılmaz. İster yerleşik I/O veya ister SB I/O veya her ikisinin kombinasyonunu kullanınız, maksimum dört darbe jeneratörüne sahip olabilirsiniz.

Dört darbe jeneratörü, varsayılan I/O atamalarına sahiptir; ancak CPU veya SB üzerinde herhangi bir dijital çıkış için yapılandırılabilir. CPU üzerindeki darbe jeneratörleri, SM'lere ya da dağıtılmış I/O'ya atanamaz.

#### Not

##### Bir Darbe-katari çıkışları kullanıcı programında diğer komutlar tarafından kullanılmaz

CPU veya sinyal kartı çıkışlarını darbe jeneratörleri (PWM veya hareket kontrol komutları ile kullanım için) olarak yapılandırdığınızda, ilgili çıkış adresleri artık çıkışları kontrol etmez. Kullanıcı programınız bir darbe jeneratörü olarak kullanılan bir çıkışa bir değer yazarsa, CPU bu değeri fiziksel çıkışa yazmaz.

Çizelge 9- 22 Kontrol edilebilir maksimum sürücü sayısı

CPU tipi		Yerleşik I/O; SB takılmadı		Bir SB ile (2 x DC çıkışları)		Bir SB ile (4 x DC çıkışları)	
		Yönlü	Yönsüz	Yönlü	Yönsüz	Yönlü	Yönsüz
CPU 1211C	DC/DC/DC	2	4	3	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4
CPU 1212C	DC/DC/DC	3	4	3	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4
CPU 1214C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4

CPU tipi		Yerleşik I/O; SB takılmadı		Bir SB ile (2 x DC çıkışları)		Bir SB ile (4 x DC çıkışları)	
		Yönlü	Yönsüz	Yönlü	Yönsüz	Yönlü	Yönsüz
CPU 1215C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	0	0	1	2	2	4
	DC/DC/RLY	0	0	1	2	2	4
CPU 1217C	DC/DC/DC	4	4	4	4	4	4

**Not**

**Maksimum darbe jeneratörü sayısı dördür.**

İster yerleşik I/O veya ister SB I/O ya da her ikisini bir arada kullanınız, maksimum dört darbe jeneratörüne sahip olabilirsiniz.

Çizelge 9- 23 CPU çıkışı: maksimum frekans

CPU	CPU çıkış kanalı	Darbe ve yön çıkışı	A/B, dördün, yukarı/aşağı ve darbe/yön
1211C	Qa.0 ila Qa.3	100 kHz	100 kHz
1212C	Qa.0 ila Qa.3	100 kHz	100 kHz
	Qa.4, Qa.5	20 kHz	20 kHz
1214C ve 1215C	Qa.0 ila Qa.3	100kHz	100kHz
	Qa.4 ila Qb.1	20 kHz	20 kHz
1217C	DQa.0 ila DQa.3 (.0+, .0- ila .3+, .3-)	1 MHz	1 MHz
	DQa.4 ila DQb.1	100 kHz	100 kHz

Çizelge 9- 24 SB sinyal kartı çıkışı: maksimum frekans (opsiyonel kart)

SB sinyal kartı	SB çıkış kanalı	Darbe ve yön çıkışı	A/B, dördün, yukarı/aşağı ve darbe/yön
SB 1222, 200 kHz	DQe.0 ila DQe.3	200kHz	200 kHz
SB 1223, 200 kHz	DQe.0, DQe.1	200kHz	200 kHz
SB 1223	DQe.0, DQe.1	20 kHz	20 kHz

Çizelge 9- 25 Darbe çıkışlarının sınır frekansları

Darbe çıkışı	Frekans
Yerleşik	4 PTO: $2 \text{ Hz} \leq f \leq 1 \text{ MHz}$ , 4 PTO: $2 \text{ Hz} \leq f \leq 100 \text{ KHz}$ , veya 4 PTO için bu değerlerin herhangi bir kombinasyonu. <sup>1,2</sup>
Standart SB	$2 \text{ Hz} \leq f \leq 20 \text{ KHz}$
Yüksek hızlı SB'ler	$2 \text{ Hz} \leq f \leq 200 \text{ KHz}$

<sup>1</sup> Dört olası CPU 1217C çıkış hızı kombinasyonları için aşağıdaki çizelgeye bakınız.

<sup>2</sup> Dört olası CPU 1211C, 1212C CPU, CPU 1214C veya CPU 1215C çıkış hızı kombinasyonları için aşağıdaki çizelgeye bakınız.

### CPU 1217C örnek çıkış hızı konfigürasyonları

#### Not

CPU 1217C onboard diferansiyel çıkışlarını kullanarak, 1 MHz'e kadar darbe çıkışları üretebilir.

Aşağıdaki örneklerde dört olası çıkış hızı kombinasyonları gösterilmiştir:

- Örnek 1: 4 - 1 MHz PTO'lar, yön çıkışı yok
- Örnek 2: 1 - 1 MHz, 2 - 100 KHz ve 1 - 20 KHz PTO'lar, tümü yön çıkışlı
- Örnek 3: 4 - 200 KHz PTO'lar, yön çıkışı yok
- Örnek 4: 2 - 100 KHz PTO'lar ve 2 - 200 KHz PTO'lar, tümü yön çıkışlı

P = Darbe D = Yön	CPU yerleşik çıkışlar										Yüksek hızlı SB çıkışlar				Standart SB çıkışlar	
	1 MHz Çıkışlar (Q)				100 KHz Çıkışlar (Q)						200 KHz Çıkışlar (Q)				20 KHz Çıkışlar (Q)	
	0.0+	0.1+	0.2+	0.3+	0.4	0.5	0.6	0.7	1.0	1.1	4.0	4.1	4.2	4.3	4.0	4.1
	0.0-	0.1-	0.2-	0.3-												
Ex. 1: 4 - 1 MHz (yön çıkışı yok)	PTO1	P														
	PTO2		P													
	PTO3			P												
	PTO4				P											
Ex. 2: 1 - 1 MHz; 2 - 100 ve 1 - 20 KHz (tümü yön çıkışlı)	PTO1	P	D													
	PTO2				P	D										
	PTO3						P	D								
	PTO4													P	D	

P = Darbe D = Yön	CPU yerleşik çıkışlar											Yüksek hızlı SB çıkışlar				Standart SB çıkışlar											
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17										
Ex. 3: 4 - 200 KHz (yön çıkışı yok)	PTO1																										
	PTO2																										
	PTO3																										
	PTO4																										
Ex. 4: 2 - 100 KHz; 2 - 200 KHz (tümü yön çıkışlı)	PTO1						P	D																			
	PTO2								P	D																	
	PTO3																										
	PTO4																										

### CPU 1211C, CPU 1212C, CPU 1214C ve CPU 1215C örnek çıkış hızı konfigürasyonları

Aşağıdaki örneklerde dört olası çıkış hızı kombinasyonları gösterilmiştir:

- Örnek 1: 4 - 100 KHz PTO'lar, yön çıkışı yok
- Örnek 2: 2 - 100 KHz PTO'lar ve 2 - 20 KHz PTO'lar, tümü yön çıkışlı
- Örnek 3: 4 - 200 KHz PTO'lar, yön çıkışı yok
- Örnek 4: 2 - 100 KHz PTO'lar ve 2 - 200 KHz PTO'lar, tümü yön çıkışlı

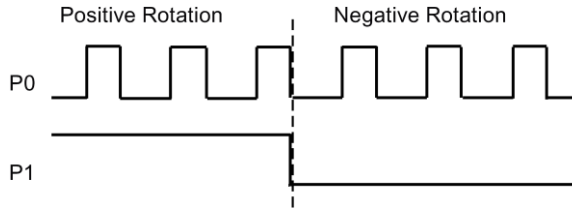
P = Darbe D = Yön	CPU yerleşik çıkışlar											Yüksek hızlı SB çıkışlar				Düşük-hızlı SB çıkışlar	
	100KHz Çıkışlar (Q)				20KHz Çıkışlar (Q)							200KHz Çıkışlar (Q)				20Kz Çıkışlar (Q)	
	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	1.0	1.1	4.0	4.1	4.2	4.3	4.0	4.1	
	CPU 1211C																
	CPU 1212C				CPU 1212C												
	CPU 1214C				CPU 1214C			CPU 1214C									
	CPU 1215C				CPU 1215C			CPU 1215C									
Ex. 1: 4 - 100 KHz (yön çıkışı yok)	PTO1	P															
	PTO2		P														
	PTO3			P													
	PTO4				P												
Ex. 2: 2 - 100 KHz; 2 - 20 KHz (tümü yön çıkışlı)	PTO1	P	D														
	PTO2			P	D												
	PTO3					P	D										
	PTO4							P	D								

P = Darbe D = Yön	CPU yerleşik çıkışlar										Yüksek hızlı SB çıkışlar				Düşük-hızlı SB çıkışlar	
	PTO1	PTO2	PTO3	PTO4	PTO5	PTO6	PTO7	PTO8	PTO9	PTO10	SB1	SB2	SB3	SB4	SB5	SB6
Ex. 3: 4 - 200 KHz (yön çıkışı yok)	PTO1										P					
	PTO2											P				
	PTO3												P			
	PTO4													P		
Ex. 4: 2 - 100 KHz; 2 - 200 KHz (tümü yön çıkışlı)	PTO1	P	D													
	PTO2			P	D											
	PTO3									P	D					
	PTO4											P	D			

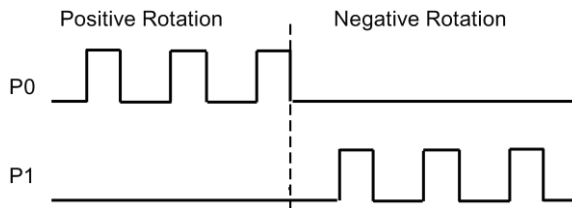
### 9.3.1 Safhalama

Step / servo sürüş için "Safhalama" ara yüzüne ait dört seçeneğiniz vardır. Bu seçenekler şunlardır:

- PTO (darbe A ve yön B): Bir PTO (darbe A ve yön B) seçeneğini seçerseniz, o zaman bir çıkış (P0) darbeyi ve bir çıkış (P1) yönü kontrol eder. Darbe pozitif yönde ise P1 (aktif) yüksektir. Darbe negatif yönde ise P1 (aktif değil) düşüktür:

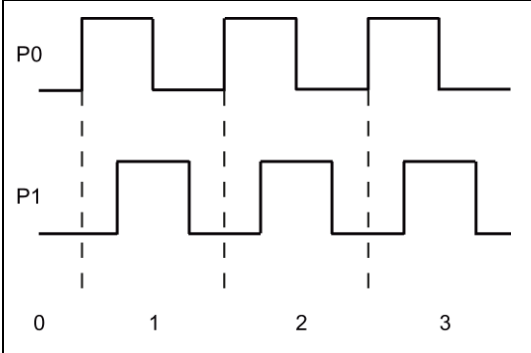
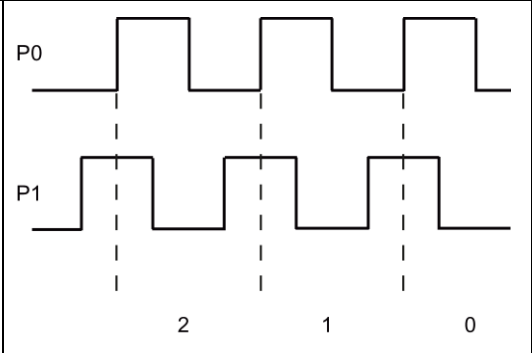


- PTO (A'ya kadar ileri say ve B'ye kadar geri say): Eğer bir PTO (A'ya kadar ileri say ve B'ye kadar geri say) seçeneği seçerseniz, bir çıkış (P0) pozitif yönler için ve farklı bir çıkış (P1) negatif yönler için darbe üretir:



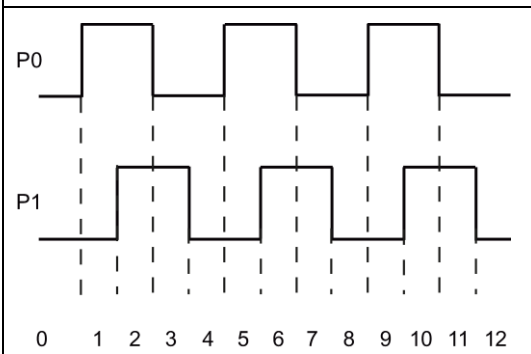
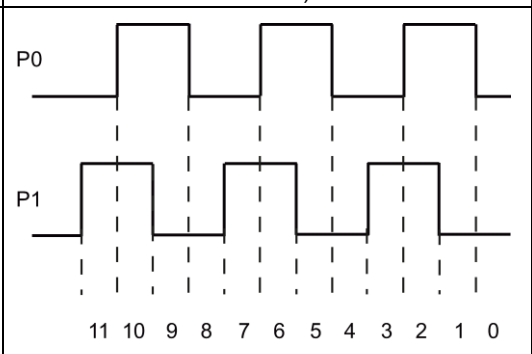
- PTO (A/B faz-kaymalı): Bir PTO (A/B faz-kaymalı) seçeneği seçerseniz, her iki çıkış belirtilen hızda ancak faz dışı 90 derecelik darbe üretir. Bu bir 1X yapılandırma olup, bir darbenin P0 pozitif geçişleri arasındaki bir miktar zaman olduğu anlamına gelir. Bu durumda yön, ilk olarak hangi çıkış geçişlerinin yüksek olduğuna göre belirlenir. Pozitif yön için P0 P1'den önde başlar. Negatif yön için P1, P0'dan önde başlar.

Üretilen darbelerin sayısı, Faz A'nın 0-1 geçiş sayısına dayanmaktadır. Faz ilişkisi hareketin yönünü belirler:

PTO (A/B faz-kaymalı)	
Faz A, faz B'den önde başlar (pozitif hareket)	Faz A, faz B'nin gerisinde başlar (negatif hareket)
	
Darbe sayısı	Darbe sayısı

- PTO (A/B faz-kaymalı - dört kat): Bir PTO (A/B faz-kaymalı - dört kat) seçeneğini seçerseniz, her iki çıkış belirtilen hızda ancak 90 derecelik faz dışı darbe üretir. Dört kat bir 4X yapılandırma olup, bir darbenin her bir çıkışın (pozitif ve negatif) geçişi anlamına gelir. İlk olarak hangi çıkış geçişlerinin yüksek olduğuna göre belirlenir. Pozitif yön için P0 P1'den önde başlar. Negatif yön için P1, P0'dan önde başlar.

Dört kat, Faz A ve Faz B'nin her ikisinin pozitif ve negatif geçişlerine dayanmaktadır. Geçişler sayısını yapılandırabilirsiniz. Faz ilişkisi (A, B'den önde veya B A'dan önde) hareketin yönünü belirler.

PTO (A/B faz-kaymalı – dört kat)	
Faz A, faz B'den önde başlar (pozitif hareket)	Faz A, faz B'nin gerisinde başlar (negatif hareket)
	
Darbe sayısı	Darbe sayısı

- PTO (darbe ve yön (yön seçimini kaldır)): Bir PTO'da (darbe ve yön (yön seçimini kaldır)) yön çıkışı seçimini kaldırırsanız, çıkış (P0) darbeyi kontrol eder. Çıkış P1 kullanılmaz ve diğer program kullanımları için emre amade durumdadır. Sadece pozitif hareket komutları bu modda CPU tarafından kabul edilir. Bu modu seçtiğinizde hareket kontrolü, kuraldışı negatif yapılandırma yapmanızı engeller. Hareket uygulamanız, yalnızca bir yönde ise, bir çıkışı kaydedebilirsiniz. Aşağıdaki şekilde tek faz (bir çıkış) gösterilmiştir: (pozitif polarite olduğu varsayılıyor)



### 9.3.2 Darbe jeneratörünün yapılandırılması

#### 5. Teknoloji nesnesi ekleyiniz:

- Proje ağacında "Teknoloji nesnelere" düğümünü genişletin ve "yeni bir nesne ekle" seçiniz.
- "Eksen" simgesini (gerekirse yeniden adlandırın) seçiniz ve eksen nesnesi yapılandırma editörünü açmak için "Tamam" a tıklayınız.
- "Temel parametreler" başlığı altında " Eksen Kontrolü için PTO seçiniz" özelliklerini görüntüleyiniz ve istenen darbeyi seçiniz.

---

#### Not

PTO, CPU özelliklerinde daha önceden yapılandırılmamış ise, PTO yerleşik çıkışlardan birini kullanmak için yapılandırılır.

Bir çıkış sinyali kartı kullanırsanız, CPU özelliklerine gitmek için "Cihaz yapılandırma" düğmesini seçiniz. "Parametre atama" altında, "Darbe seçenekleri" içinde, bir sinyal kartı çıkışına giden çıkış kaynağını yapılandırınız.

---

- Kalan Temel ve Genişletilmiş parametrelerini yapılandırınız.

#### 6. Uygulamanızı programlayınız: Bir kod bloğu içine MC\_Power komutunu yerleştiriniz.

- Eksen girişi için oluşturduğunuz ve yapılandırdığınız eksen teknoloji nesnesini seçiniz.
- Enable girişini TRUE yapma diğer hareket komutlarının çalışmasını sağlar.
- Enable girişini FALSE yapma diğer hareket komutlarını iptal eder.

---

#### Not

Eksen başına sadece bir MC\_Power komutunu dahile ediniz.

---

#### 7. Gerekli hareketi üretmek için başka hareket komutlarını yerleştiriniz.



**Not**

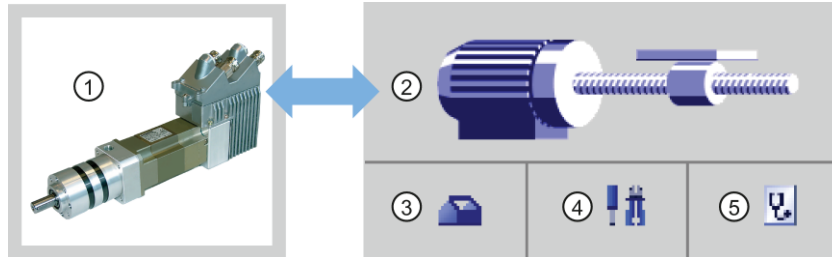
Sinyal kartı çıkışları için bir darbe jeneratörü yapılandırma: bir CPU (Cihaz Yapılandırmasında) için "Darbe jeneratörleri (PTO / PWM)" özelliklerini seçiniz ve bir darbe jeneratörünü etkinleştiriniz. Her S7-1200 CPU V1.0, V2.0, V2.1, V2.2 için iki darbe jeneratörü kullanılabilir. S7-1200 CPU V3.0 ve V4.0 CPU'lar kullanılabilir dört darbe jeneratörüne sahiptir. "darbe seçenekleri" altında bu aynı yapılandırma alanında "PTO" olarak kullanılan darbe jeneratörünü seçiniz.

**Not**

CPU, "dilimler" veya 10 ms'lik segmentlerde hareket görevlerini hesaplar. Bir dilim yürütülürken, bir sonraki dilim yürütülecek kuyrukta bekler. Bir eksen (bu eksen için yeni bir hareket görevi yürüterek) üzerinde hareket görevini keserseniz, yeni hareket görevi maksimum 20 ms (mevcut dilimin kalanı artı kuyruktaki dilim) için yürütülemez.

**9.3.3****Eksenin yapılandırılması**

STEP 7 "Eksen" teknoloji nesnesi için yapılandırma araçları, devreye alma araçları ve tanılama araçları sağlar.



① Sürücü

② Teknoloji nesnesi

③ Konfigürasyon

④ Devreye alma

⑤ Tanılama

**Not**

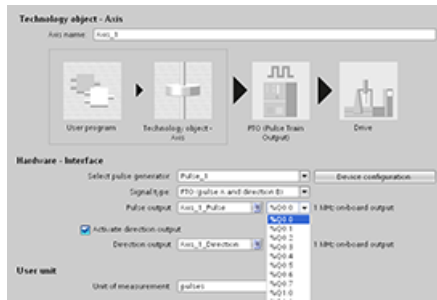
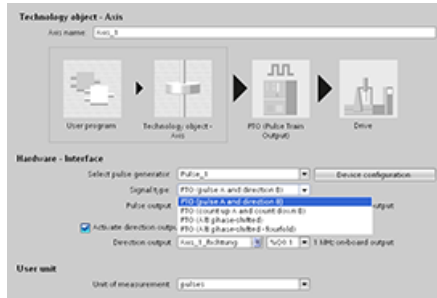
CPU firmware sürümleri V2.2 ve öncekiler için PTO, yüksek hızlı sayıcının(HSC) iç işlevselliğine ihtiyaç duyar. Bu, ilgili HSC'nin başka yerde kullanılmayacağı anlamına gelir.

PTO ve HSC arasındaki atama sabittir. PTO1 aktif ise, bu HSC1'e bağlanacaktır. PTO2 aktif ise, bu HSC2'ye bağlanacaktır. Darbeler meydana gelirken mevcut değerini (örneğin ID1000'de) izleyemezsiniz.

S7-1200 V3.0 ve üstü CPU'larda bu kısıtlama yoktur; darbe çıkışları bu CPU'larda yapılandırıldığında tüm HSC'ler program kullanımı için emre amade kalmaya devam edecektir.

Çizelge 9- 26 Hareket kontrolü için STEP 7 araçları

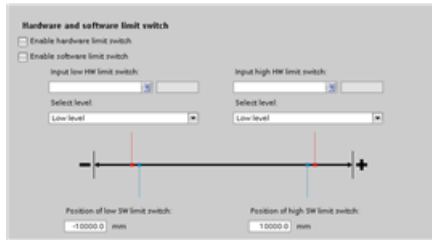
Araç	Açıklama
Konfigürasyon	"Eksen" teknoloji nesnesinin aşağıdaki özelliklerini yapılandırır: <ul style="list-style-type: none"> <li>• Kullanılacak PTO seçimi ve sürücü ara yüzü yapılandırması</li> <li>• Sürücünün (veya makine ya da sistemin) mekanik özellikleri ve aktarma oranı</li> <li>• Pozisyon limitleri, dinamik ve homing için özellikler</li> </ul> Teknoloji nesnenin veri bloğundaki yapılandırmasını kaydediniz.
Devreye alma	Bir kullanıcı programı oluşturmak zorunda kalmadan, ekseninizin fonksiyonunu test eder. Araç başlatıldığında, kontrol paneli görüntülenir. Aşağıdaki komutlar kontrol panelinde mevcuttur: <ul style="list-style-type: none"> <li>• Ekseni etkinleştir ve devre dışı bırak</li> <li>• Adım modunda ekseni hareket ettir</li> <li>• Mutlak ve nispi olarak ekseni konumlandır</li> <li>• Ekseni başlangıç konumuna al</li> <li>• Hataları kabul et</li> </ul> Hareket komutları için hız ve hızlanma/yavaşlama belirlenebilir. Kontrol paneli, geçerli eksen durumunu da gösterir.
Tanımlama	Eksen ve sürücü için mevcut durum ve hata bilgilerini izler.



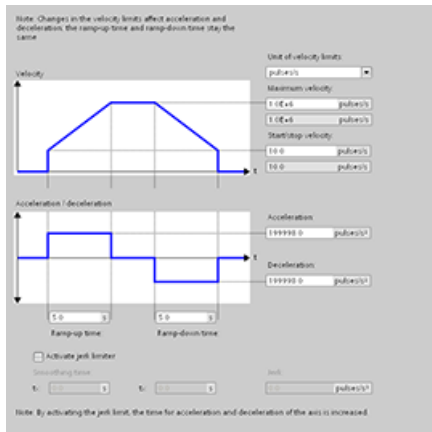
Eksen için teknoloji nesnesi oluşturduktan sonra, PTO ve sürücü ara yüzü yapılandırması gibi temel parametreleri tanımlayarak ekseni yapılandırınız. Ayrıca, pozisyon limitleri, dinamikler ve homing gibi eksenin diğer özelliklerini yapılandırınız.

### Not

Kullanıcı programında hareket kontrol komutlarının giriş parametrelerinin değerlerini yeni boyut birimine adapte etmeniz gerekebilir.



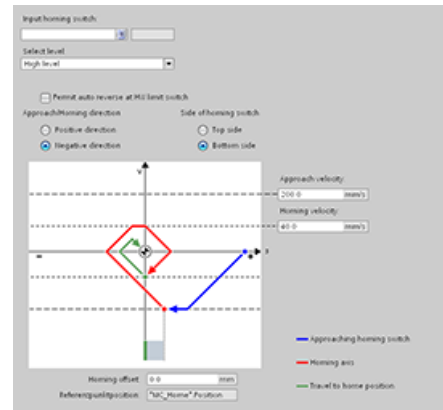
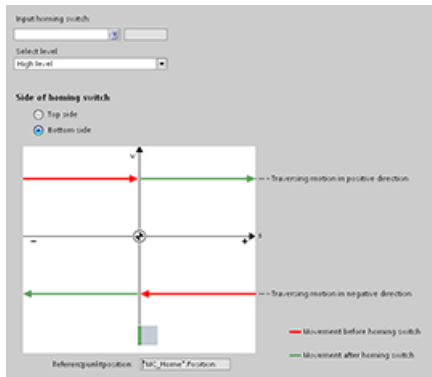
Sürücü sinyalleri, sürücü mekaniği ve pozisyon izleme için özellikleri (donanım ve yazılım limit anahtarları) yapılandırınız.



Hareket dinamiklerini ve acil durdurma komutunun davranışını yapılandırınız.



Ayrıca homing davranışını (pasif ve aktif) yapılandırınız.



Kullanıcı programından bağımsız olarak işlevselliği sınamak için "Devreye Alma" kontrol panelini kullanınız.



Ekseni devreye almak için "Başlangıç" simgesine tıklayınız.

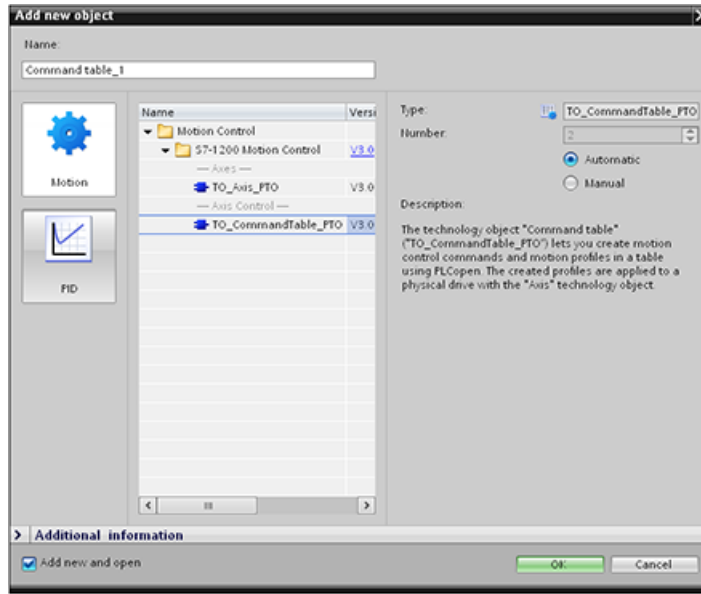
Kontrol paneli, ekseninin mevcut durumunu gösterir. Ekseni etkinleştirme ve devre dışı bırakmanın yanı sıra, (mutlak ve nispi olarak) ekseninin konumlandırılmasını test edebilir ve hız, hızlanma ve yavaşlamayı belirtebilirsiniz. Ayrıca homing ve jogging (yavaş koşma) görevlerini test edebilirsiniz. Kontrol paneli ayrıca hataları kabul etmenize izin verir.

### 9.3.4 TO\_CommandTable\_PTO yapılandırması

Teknoloji nesnelere kullanarak bir CommandTable komutunu yapılandırabilirsiniz.

#### Teknoloji nesne ekleme

8. Proje ağacında, "Teknoloji nesnelere" düğümünü genişletiniz ve "yeni nesne Ekle" seçiniz.
9. "CommandTable" simgesini (gerekirse yeniden adlandırınız) seçiniz ve CommandTable nesnesi yapılandırma editörünü açmak için "Tamam" a tıklayınız.



#### Uygulamanız için adımları planlama

"Komut Çizelgesi" yapılandırma penceresinde istenilen hareket sırasını oluşturabilir ve eğilim şemasında grafik görünümüne göre sonucu kontrol edebilirsiniz.

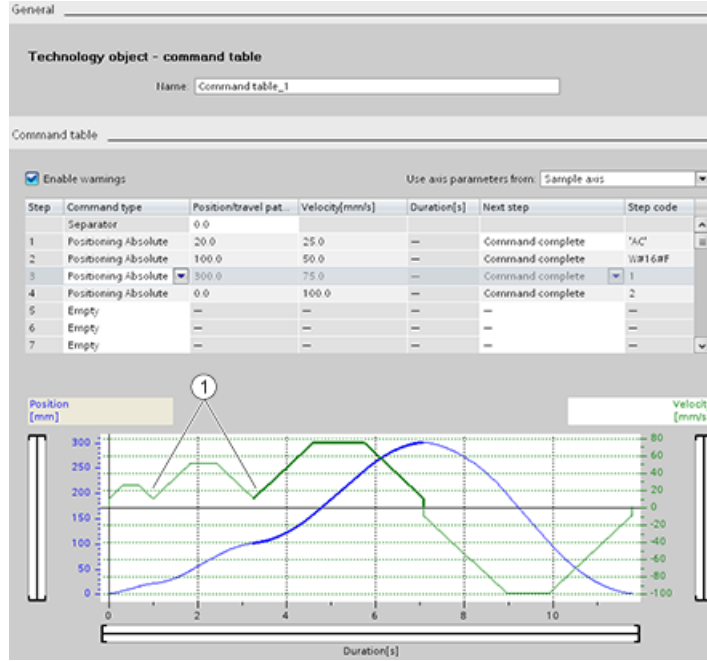
Kumanda çizelgesini işlemek için kullanılacak komut tiplerini seçebilirsiniz. 32 adıma kadar girilebilir. Komutlar sırayla işlenir ve kolayca karmaşık bir hareket profili üretilebilir.

Çizelge 9- 27 MC\_CommandTable komut tipleri

Komut tipi	Açıklama
Boşalt	Boşalt, eklenecek tüm komutlar için bir yer tutucu olarak hizmet verir. Boşalt girişi komut çizelgesi işlendiğinde yok sayılır.
Durdur	Ekseni durdurur. Not: Komut, yalnızca bir "Hız ayar noktası" komutundan sonra gerçekleşir.
Bağıl konumlandırma	Eksenini mesafeye dayalı olarak konumlandırır. Komut, verilen mesafe ve hıza göre ekseni hareket ettirir.
Mutlak konumlandırma	Ekseni konuma dayalı olarak konumlandırır. Komut, belirtilen hız kullanılarak verilen konuma göre ekseni hareket ettirir.
Hız ayar noktası	Ekseni verilen hızda hareket ettirir.

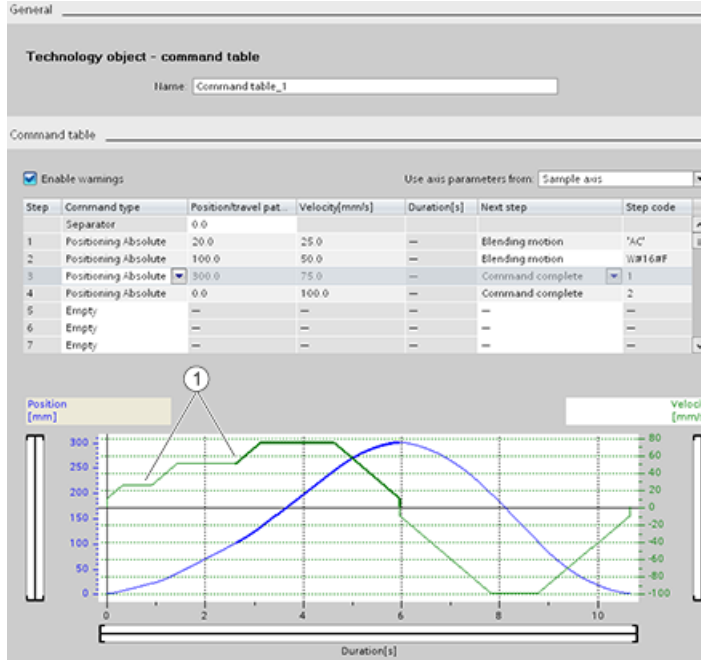
Komut tipi	Açıklama
Bekle	Verilen süre bitene kadar bekler. "Wait" aktif bir dönme hareketini durdurmaz.
Ayırıcı	Secilen çizgi üzerinde. "ayırıcı" bir çizgi ekler. Ayırıcı çizgi, birden fazla profilin, tek bir komut çizelgesinde tanımlanmasına izin verir.

Aşağıdaki şekilde, "Komut tamamlandı" bir sonraki aşamaya geçiş olarak kullanılır. Bu geçiş türü cihazınızın start/stop hızına yavaşlamasına ve ardından bir sonraki adım başında bir kez daha hızlanmasına izin verir.



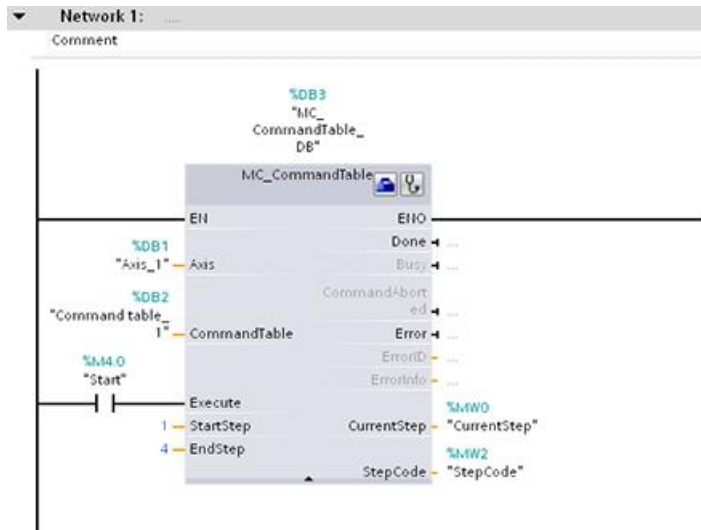
① Eksen, adımlar arasında start/stop hızına yavaşlar.

Aşağıdaki şekilde, "Karıştirma hareketi" bir sonraki adıma geçiş olarak kullanılır. Bu tür bir geçiş, cihazın bir adımdan bir sonraki adıma yumuşak geçişi ile sonuçlanan, cihazınızın bir sonraki adımın başlangıcında hızını korumasına imkan sağlar. Karıştirma kullanımı işlemin tamamen gerçekleşmesi için bir profile ilgili gerekli toplam süreyi kısaltabilir. Karıştirma olmadan, örneği çalıştırmak için yedi saniye gerekir. Karıştirma ile işletim süresi bir saniye azaltılarak toplam altı saniye olur.



- ① Eksen hareket etmeye devam eder ve sonraki adım hızına göre hızlanır veya yavaşlar böylece zamandan ve mekanik yıpranmadan tasarruf eder.

CommandTable komutunuzun çalışması aşağıda gösterildiği gibi bir MC\_CommandTable komutu ile kontrol edilir:



## 9.3.5 Hareket kontrol komutları

### 9.3.5.1 MC komutuna genel bakış

Hareket kontrol komutları, bir eksen üzerinde hareketi kontrol etmek için ilişkili bir teknoloji veri bloğu ve CPU'nun özel PTO'sunu (darbe çıkışları) kullanır.

- MC\_Power (Sayfa 456) hareket kontrol eksenini etkinleştirir ve devre dışı bırakır.
- MC\_Reset (Sayfa 459), tüm hareket kontrol hatalarını sıfırlar. Kabul edilebilen tüm hareket kontrol hataları kabul edilir.
- MC\_Home ( Sayfa 460) eksen kontrol programı ile eksen mekanik konumlandırma sistemi arasındaki ilişkiyi kurar.
- MC\_Halt (Sayfa 462) tüm hareket proseslerini iptal eder ve eksen hareketinin durmasına neden olur. Durma pozisyonu tanımlı değildir.
- MC\_MoveAbsolute (Sayfa 464) hareketi mutlak bir pozisyonda başlatır. Hedef pozisyona ulaşıldığında iş biter.
- MC\_MoveRelative (Sayfa 466) hareketi başlangıç pozisyonuna göre konumlandırmayı başlatır.
- MC\_MoveVelocity (Sayfa 468) eksenin belirtilen hız ile dönmesine neden olur.
- MC\_MoveJog (Sayfa 1471) test ve yol verme amaçları için adım modunu yürütür.
- MC\_CommandTable ( Sayfa 473 ) bir hareket dizisi olarak eksen komutlarını çalıştırır.
- MC\_ChangeDynamic (Sayfa 476 ) eksen için Dinamik ayarları değiştirir.
- MC\_WriteParam (Sayfa 478) kullanıcı programından eksenin işlevselliğini değiştirmek için belirli bir sayıda parametre yazar.
- MC\_ReadParam (Sayfa 480) Eksen girişinde tanımlanan mevcut konum, hız ve benzerini işaret eden seçilen sayıda parametreyi okur.

### CPU firmware seviyeleri

V4.0 firmware'e sahip bir S7-1200 CPU'nuz varsa, her hareket komutunun V4.0 sürümünü seçiniz.

V3.0 veya daha önceki firmware sahip bir S7-1200 CPU'nuz varsa, her bir hareket komutunun uygulanabilir V3.0, V2.0 ya da V1.0 sürümünü seçiniz.

---

#### Not

CPU, "dilimler" veya 10 ms'lik segmentlerde hareket görevlerini hesaplar. Bir dilim yürütülürken, bir sonraki dilim yürütülecek kuyrukta bekler. Bir eksen (bu eksen için yeni bir hareket görevi yürüterek) üzerinde hareket görevini keserseniz, yeni hareket görevi maksimum 20 ms (mevcut dilimin kalanı artı kuyruktaki dilim) için yürütülemez.

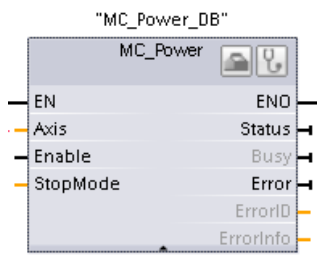
---

### 9.3.5.2 MC\_Power (Ekseni serbest bırakma/engelleme) komutu

#### Not

Eksen bir hata nedeniyle durdurulursa, bu hata ortadan kaldırılıp onaylandıktan sonra otomatik olarak tekrar aktif olacaktır. Bu, "Enable input" parametresinin bu proses sırasında TRUE değerini korumasını gerektirir.

Çizelge 9- 28 MC\_Power komutu

LAD / FBD	SCL	Açıklama
 <p>"MC_Power_DB"</p> <p>MC_Power</p> <p>EN END</p> <p>Axis Status</p> <p>Enable Busy</p> <p>StopMode Error</p> <p>ErrorID</p> <p>ErrorInfo</p>	<pre>"MC_Power_DB" (     Axis:= _multi_fb_in_,     Enable:= _bool_in_,     StopMode:= _int_in_,     Status=&gt; _bool_out_,     Busy=&gt; _bool_out_,     Error=&gt; _bool_out_,     ErrorID=&gt; _word_out_,     ErrorInfo=&gt; _word_out_ );</pre>	<p>MC_Power hareket kontrol komutu, bir eksenin etkinleştirir veya devre dışı bırakır. Eksen etkinleştirmeden veya devre dışı bırakmadan önce, aşağıdaki koşulları sağlayınız:</p> <ul style="list-style-type: none"> <li>• Teknoloji nesnesi düzgün yapılandırılmıştır.</li> <li>• Etkinleştirmeyi engelleyici bekleyen hiçbir hata yoktur.</li> </ul> <p>MC_Power yürütmesi, bir hareket kontrol görevi tarafından iptal edilemez. Eksenin devre dışı bırakılması (giriş parametresi Enable = FALSE) ilgili teknoloji nesnesi için tüm hareket kontrol görevlerini iptal eder.</p>

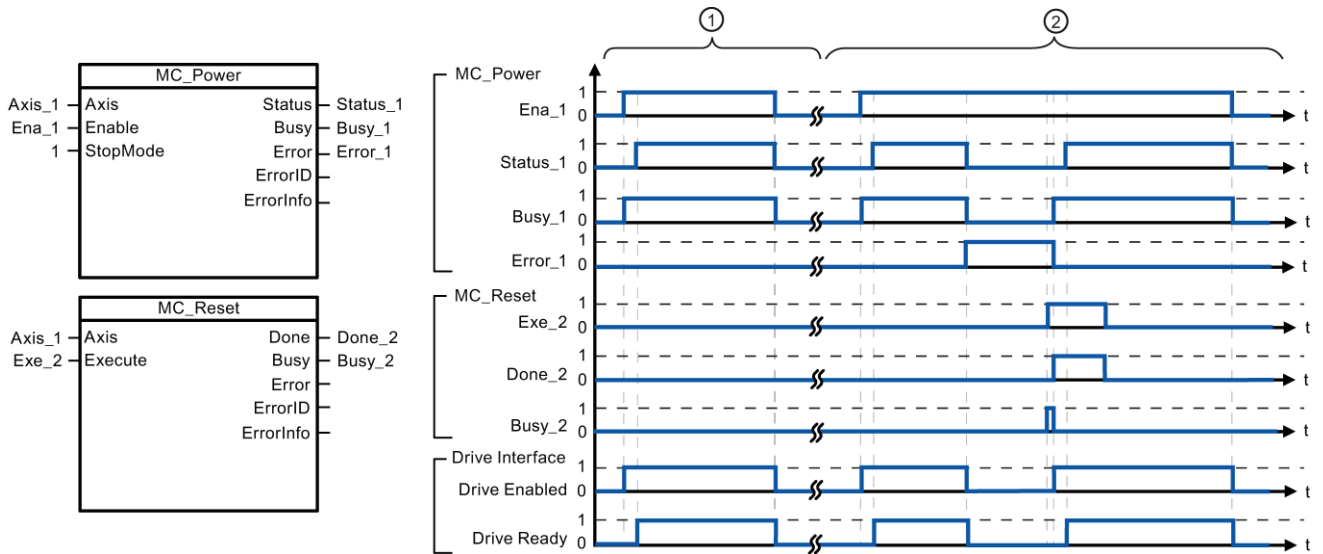
- 1 Komutu yerleştirdiğinizde STEP 7 otomatik DB oluşturur.
- 2 SCL örneğinde, "MC\_Power\_DB" kopya DB'nin ismidir.

Çizelge 9- 29 MC\_Power komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama
Axis	IN	TO_Axis_1 Eksen teknoloji nesnesi
Enable	IN	Bool <ul style="list-style-type: none"> <li>• FALSE (varsayılan): Tüm aktif görevler parametrelendirilen "StopMode" a göre iptal edilir ve eksen durdurulur.</li> <li>• TRUE: Hareket kontrolü, eksenin etkinleştirmeye çalışır.</li> </ul>
StopMode	IN	Int <ul style="list-style-type: none"> <li>• 0: Acil durdurma: Eksenin devre dışı bırakmak için bir istek beklemede ise, yapılandırılmış acil yavaşlamada eksen frenler. Eksen, duraklama haline geldikten sonra devre dışı bırakılır.</li> <li>• 1: Derhal durdurma: Eksenin devre dışı bırakmak için bir istek beklemede ise, bu eksen yavaşlama olmadan devre dışı bırakılır. Darbe çıkışı derhal durdurulur.</li> <li>• Sarsıntı kontrolü ile acil durdurma: Eksenin devre dışı bırakmak için bir istek beklemede ise, yapılandırılmış acil durdurma yavaşlamasında eksen frenler. Sarsıntı kontrolü aktif ise, yapılandırılmış sarsıntı dikkate alınır. Eksen duraklama haline geldikten sonra devre dışı bırakılır.</li> </ul>



Parametre ve tipi	Veri tipi	Açıklama
Durum	OUT	Bool Eksen durumu etkin: <ul style="list-style-type: none"> <li>FALSE: Eksen devre dışı bırakılır: <ul style="list-style-type: none"> <li>Eksen, hareket kontrol görevlerini yerine getirmez ve yeni görevler (istisna: MC_Reset görevi) kabul etmez.</li> <li>Eksen başlangıç konumuna gelmez.</li> <li>Devre dışı bırakılması üzerine eksen duraklama haline gelene kadar, durum FALSE haline geçmez.</li> </ul> </li> <li>TRUE: Eksen etkindir: <ul style="list-style-type: none"> <li>Eksen, hareket kontrol görevlerini yürütmek için hazırdır.</li> <li>Eksenin etkinleştirilmesi üzerine "Sürücü hazır" sinyali beklenene kadar, durum TRUE haline geçmez. "Sürücü hazır" sürücü ara yüzü, eksen konfigürasyonunda yapılandırılmamışsa, durum derhal TRUE haline geçer</li> </ul> </li> </ul>
Busy	OUT	Bool FALSE: MC_Power aktif değildir. TRUE: MC_Power aktif.
Error	OUT	Bool FALSE: Hata yok TRUE: "MC_Power" hareket kontrol komutunda veya ilgili teknoloji nesnesinde bir hata oluştu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir.
ErrorID	OUT	Word "Error" parametresi için Error ID
ErrorInfo	OUT	Word "ErrorID" parametresi için Error info ID



- ① Bir eksen etkinleştirilir ve daha sonra tekrar devre dışı bırakılır. Sürücü CPU'ya "Sürücü hazır" sinyalini tekrar gönderdikten sonra, başarılı etkinleştirme "Status\_1" üzerinden okunabilir.
- ② Bir eksenin etkinleştirilmesinin ardından, eksenin devre dışı kalmasına neden olan bir hata oluştu. Hata "MC\_Reset" ile ortadan kaldırılır ve onaylanır. Eksen sonra tekrar etkinleştirilir.

Yapılandırılmış sürücü ara yüzü ile bir eksenin etkinleştirmek için şu adımları izleyiniz:

10. Yukarıda belirtilen kuralları kontrol ediniz.

11. İstenen değer ile giriş parametresi "StopMode" u başlatınız. Giriş parametresi "Enable" TRUE olarak ayarlayınız.

Sürücüye giden gücü etkinleştirmek için " Drive enabled" için etkinleştirme çıkışı TRUE haline geçer.

CPU, sürücünün "Sürücü hazır" sinyali için bekler.

"Sürücü hazır" sinyali CPU'nun yapılandırılmış hazır girişinde kullanılabilir olduğunda, eksen etkin hale gelir. Çıkış parametresi " Status " ve teknoloji nesne etiketi <Axis name>. StatusBits.Enable değeri TRUE gösterir.

Yapılandırılmış sürücü ara yüzü olmadan bir eksenin etkinleştirmek için şu adımları izleyiniz:

12. Yukarıda belirtilen kuralları kontrol ediniz.

13. İstenen değer ile giriş parametresi "StopMode" u başlatın. Giriş parametresi "Enable" TRUE olarak ayarlayınız. Eksen etkindir. Çıkış parametresi " Status " ve teknoloji nesne etiketi <Axis name>. StatusBits.Enable değeri TRUE gösterir.

Bir eksenin devre dışı bırakmak için şu adımları izleyiniz:

14. Eksenin durma haline getiriniz.

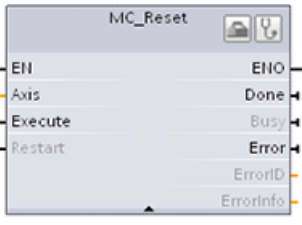
Eksen, teknoloji nesne etiketi <Axis name >. StatusBits.StandStill in ne zaman durduğunu saptayabilirsiniz.

15. Durma noktasına ulaşıldıktan sonra giriş parametresi "Enable" FALSE olarak ayarlayınız.

16. Çıkış parametreleri " Busy" ve "Status" ve teknoloji nesne etiketi <Axis. StatusBits.Enable FALSE değerini gösterirse, eksenin devre dışı edilmesi tamamlandı demektir.

### 9.3.5.3 MC\_Reset (Hatayı onayla) komutu

Çizelge 9- 30 MC\_Reset komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows the MC_Reset DB with the following connections:      - EN: EHO      - Axis: Done      - Execute: Busy      - Restart: Error      - Done: ErrorID      - Busy: ErrorInfo</p>	<pre>"MC_Reset_DB" (   Axis:=_multi_fb_in_,   Execute:=_bool_in_,   Restart:=_bool_in_,   Done=&gt;_bool_out_,   Busy=&gt;_bool_out_,   Error=&gt;_bool_out_,   ErrorID=&gt;_word_out_,   ErrorInfo=&gt;_word_out_);</pre>	<p>"Operating error with axis stop" ve "Configuration error" kabul etmek için MC_Reset komutunu kullanınız. Onay gerektiren hatalar "Remedy" başlığı altında "ErrorID'ler ve ErrorInfo'lar Listesi" nde bulunabilir.</p> <p>MC_Reset komutunu kullanmadan önce, onay gerektiren bekleyen bir yapılandırma hatasının nedenini ortadan kaldırmalısınız. (örneğin "Eksen" teknoloji nesnesinde geçersiz bir ivme değerini geçerli bir değerine değiştirerek)</p> <p>V3.0 ve sonrası itibarıyla, Restart komutu, eksen konfigürasyonunun RUN çalışma modunda çalışma hafızasına yüklenmesini sağlar.</p>

- 1 Komutu yerleştirdiğinizde STEP 7 otomatik DB oluşturur.
- 2 SCL örneğinde, "MC\_Reset\_DB" kopya DB'nin ismidir.

MC\_Reset görevi başka bir hareket kontrol görevi tarafından iptal edilemez. Yeni MC\_Reset görevi başka bir aktif hareket kontrol görevlerini iptal etmez.

Çizelge 9- 31 MC\_Reset komutu parametreleri

Parametre ve tipi	Veri tipi	Açıklama	
Axis	IN	TO_Axis_1	Eksen teknoloji nesnesi
Execute	IN	Bool	Bir pozitif kenar ile görev başlatınız
Restart	IN	Bool	TRUE = çalışma hafızasına yük bellekten eksen yapılandırmasını indiriniz. Eksen devre dışı bırakıldığında komutu yalnızca idam edilebilir.
			FALSE = Bekleyen hataları onaylar
Done	OUT	Bool	TRUE = Hata onaylandı.
Busy	OUT	Bool	TRUE = Görev yürütülüyor.
Error	OUT	Bool	TRUE = görevin yürütülmesi sırasında bir hata oluştu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir.
ErrorID	OUTP	Word	"Error" parametresi için Error ID
ErrorInfo	OUT	Word	"ErrorID" parametresi için Error info ID

MC\_Reset ile bir hatayı onaylamak için şu adımları izleyiniz:

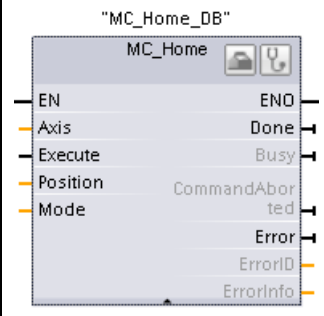
17.Yukarıda belirtilen kuralları kontrol ediniz.

18.Execute giriş parametresinde bir pozitif kenarı ile hata onayını başlatınız.

19.Done TRUE'ya eşit ve teknoloji nesne etiketi <Axis name >. StatusBits.Error FALSE'a eşit olduğunda hata onaylanmıştır.

### 9.3.5.4 MC\_Home (Home axis) komutu

Çizelge 9- 32 MC\_Home komutu

LAD / FBD	SCL	Açıklama
	<pre>"MC_Home_DB" (   Axis:=_multi_fb_in_,   Execute:=_bool_in_,   Position:=_real_in_,   Mode:=_int_in_,   Done=&gt;_bool_out_,   Busy=&gt;_bool_out_,   CommandAborted=&gt;_bool_out_,   Error=&gt;_bool_out_,   ErrorID=&gt;_word_out_,   ErrorInfo=&gt;_word_out_);</pre>	<p>Eksenin koordinatlarını gerçek, fiziksel sürücü pozisyonu ile eşleştirmek için MC_Home komutunu kullanınız. Homing, eksenin mutlak konumu için gereklidir: MC_Home komutunu kullanmak için eksen ilk önce etkinleştirilmelidir.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP 7 otomatik olarak DB oluşturur.

<sup>2</sup> SCL örneğinde, "MC\_Home\_DB " kopya DB'nin ismidir.

Aşağıdaki Homing tipleri mevcuttur:

- Direct homing absolute (Mod = 0): Geçerli eksen konumu "Position " parametre değerine ayarlanır.
- Direct homing relative (Mod = 1): Geçerli eksen konumu "Position " parametre değeriyle kaydırılır.
- Passive homing (Mod = 2): Passive homing sırasında, MC\_Home komutu herhangi homing hareketi gerçekleştirmez. Bu adım için gerekli olan dönme hareketi, diğer hareket kontrol komutları aracılığıyla kullanıcı tarafından uygulanmalıdır. Referans noktası değişimi tespit edildiğinde, eksen özgüdümlenir.
- Active homing (Mod = 3): Homing prosedürü otomatik olarak yürütülür.

Çizelge 9- 33 MC\_Home komut parametreleri

Parametre ve tipi		Veri tipi	Açıklama
Axis	IN	TO_Axis_PTO	Eksen teknoloji nesnesi
Execute	IN	Bool	Bir pozitif kenar ile görevi başlatın
Position	IN	Real	<ul style="list-style-type: none"> <li>Mod = 0, 2 ve 3 (homing işleminin tamamlanmasından sonra ekseninin mutlak pozisyonu)</li> <li>Mod = 1 (Mevcut eksen pozisyonu için düzeltme değeri)</li> </ul> Limit değerler: $-1.0e^{12} \leq \text{Pozisyon} \leq 1.0e^{12}$
Mode	IN	Int	Homing modu <ul style="list-style-type: none"> <li>0: Direct homing mutlak Yeni eksen pozisyonu, " Position " parametresinin pozisyon değeridir.</li> <li>1: Direct homing bağlı Yeni eksen pozisyonu, "Position" parametresinin pozisyon değeri + geçerli eksen pozisyonu.</li> <li>2: Passive homing Eksen yapılandırılmaya göre homing. Homing ardından, "Position " parametre değeri yeni eksen pozisyonu olarak ayarlanır.</li> <li>3: Active homing Eksen konfigürasyonuna uygun referans noktası yaklaşımı. Homing ardından, "Position" parametre değeri yeni eksen pozisyonu olarak ayarlanır.</li> </ul>
Done	OUT	Bool	TRUE = Görev tamamlandı
Busy	OUT	Bool	TRUE = Görev yürütülüyor.
CommandAborted	OUT	Bool	TRUE = Yürütülmesi sırasında görev başka bir görev tarafından iptal edildi.
Error	OUT	Bool	TRUE = görevin yürütülmesi sırasında bir hata oluştu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir.
ErrorID	OUT	Word	"Error" parametresi için Error ID
ErrorInfo	OUT	Word	"ErrorID" parametresi için Error info ID

#### Not

##### Eksen homing aşağıdaki koşullarda kaybolur

- MC\_Power komutuyla ekseni devre dışı bırakma
- Otomatik kontrol ve manuel kontrol arasındaki geçiş
- Active homing başlaması üzerine (homing işleminin başarıyla tamamlanmasından sonra, eksen homing yeniden yapılabilir.)
- CPU güç-çevriminden sonra
- CPU yeniden başlatıldıktan sonra (RUN-STOP veya STOP-RUN)

Ekseni özgüdümlmek için şu adımları izleyiniz:

20. Yukarıda belirtilen kuralları kontrol ediniz.

21. Gerekli giriş parametrelerini değerleri ile başlat ve giriş parametresi "Execute" da pozitif kenarı ile homing işlemini başlatınız.

22. Çıkış parametresi " Done " ve teknoloji nesne etiketi <Axis name >.

StatusBits.HomingDone TRUE değerini gösteriyorsa, Homing tamamlanmıştır.

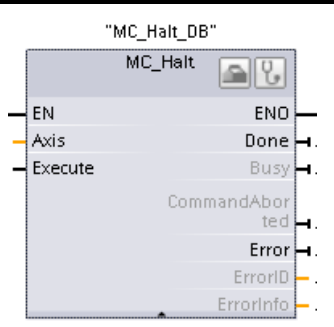
Çizelge 9- 34 Geçersiz kılma tepkisi

Mod	Açıklama
0 veya 1	MC_Home görevi başka bir hareket kontrol görevi tarafından iptal edilemez. Yeni MC_Home görevi herhangi bir aktif hareket kontrol görevlerini iptal etmez. Pozisyon ile ilgili hareket görevleri ("Position" giriş parametresindeki değer) yeni homing konumuna göre homing sonrası kaldığı yerden devam eder.
2	MC_Home görevi aşağıdaki hareket kontrol görevleri tarafından iptal edilebilir: MC_Home görevi Modu = 2, 3: Yeni MC_Home görevi aşağıdaki aktif hareket kontrol görevini iptal eder. MC_Home görevi Modu = 2: Pozisyon ilgili hareket görevleri yeni homing konumuna ("Position" giriş parametresinde değer) göre homing sonrası kaldığı yerde devam eder.
3	MC_Home görevi aşağıdaki hareket kontrol görevleri tarafından iptal edilebilir: <ul style="list-style-type: none"> <li>MC_Home Mode = 3</li> <li>MC_Halt</li> <li>MC_MoveAbsolute</li> <li>MC_MoveRelative</li> <li>MC_MoveVelocity</li> <li>MC_MoveJog</li> </ul>

Yeni MC_Home görevi aşağıdaki aktif hareket kontrol görevlerini iptal: <ul style="list-style-type: none"> <li>MC_Home Mode = 2, 3</li> <li>MC_Halt</li> <li>MC_MoveAbsolute</li> <li>MC_MoveRelative</li> <li>MC_MoveVelocity</li> <li>MC_MoveJog</li> </ul>
---

### 9.3.5.5 MC\_Halt komutu

Çizelge 9- 35 MC\_Halt komutu

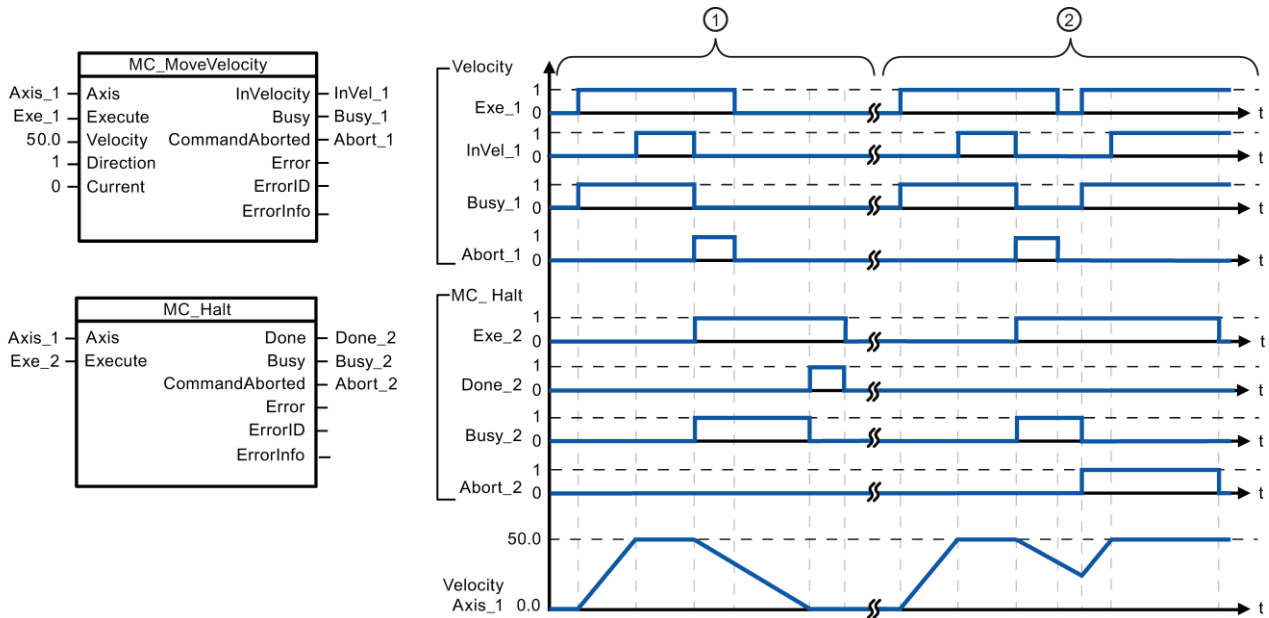
LAD / FBD	SCL	Açıklama
	<pre>"MC_Halt_DB" (   Axis:= _multi_fb_in_,   Execute:= _bool_in_,   Done=&gt; _bool_out_,   Busy=&gt; _bool_out_,   CommandAborted=&gt; _bool_out_,   Error=&gt; _bool_out_,   ErrorID=&gt; _word_out_,   ErrorInfo=&gt; _word_out_);</pre>	<p>Tüm hareketi durdurmak ve ekseni hareketsiz hale getirmek için MC_Halt komutunu kullanınız. Hareketsiz pozisyon tanımlı değildir:</p> <p>MC_Halt komutunu kullanmak için eksen ilk önce etkinleştirilmelidir.</p>

1 Komutu yerleştirdiğinizde STEP 7 otomatik olarak DB oluşturur.

2 SCL örneğinde, "MC\_Halt\_DB " kopya DB'nin ismidir.

Çizelge 9- 36 MC\_Halt komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama	
Axis	IN	TO_Axis_1	Eksen teknoloji nesnesi
Execute	IN	Bool	Görevi yükselen kenarla başlatınız
Done	OUT	Bool	TRUE = Sıfır hıza ulaşıldı
Busy	OUT	Bool	TRUE = Görev yürütülüyor.
CommandAborted	OUT	Bool	TRUE = Yürütme sırasında görev başka bir görev tarafından iptal edildi.
Error	OUT	Bool	TRUE = Yürütme sırasında bir hata oluştu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir.
ErrorID	OUT	Word	"Error" parametresi için Error ID.
ErrorInfo	OUT	Word	"ErrorID" parametresi için Error info ID.



Aşağıdaki değerler "Dynamics> General" yapılandırma penceresinde yapılandırılmıştır: Hızlanma = 10,0 ve Yavaşlama = 5.0

- ① Hareketsiz kalana kadar eksen bir MC\_Halt görevi tarafından frenlenir. Eksen hareketsizliği "Done\_2" üzerinden bildirilir.
- ② Bir MC\_Halt görevi eksenini frenlerken, bu görev başka bir hareket görevi tarafından iptal edilir. İptal "Abort\_2" üzerinden bildirilir.

### Geçersiz kılma tepkisi

MC\_Halt görevi aşağıdaki hareket kontrol görevleri tarafından iptal edilebilir:

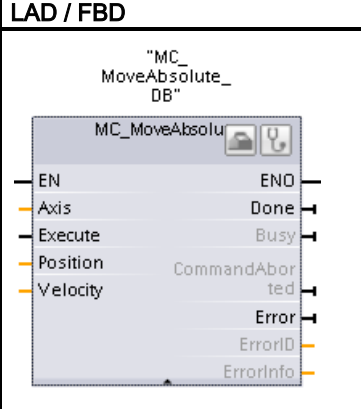
- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

Yeni MC\_Halt görevi aşağıdaki aktif hareket kontrol görevlerini iptal eder:

- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

### 9.3.5.6 MC\_MoveAbsolute (Ekseni mutlak konumla) komutu

Çizelge 9- 37 MC\_MoveAbsolute komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a function block named 'MC_MoveAbsolute_DB'. It has several input and output ports: EN (Enable), Axis (Axis), Execute (Execute), Position (Position), Velocity (Velocity), END (End), Done (Done), Busy (Busy), CommandAborted (CommandAborted), Error (Error), ErrorID (ErrorID), and ErrorInfo (ErrorInfo).</p>	<pre>"MC_MoveAbsolute_DB" (   Axis:=_multi_fb_in_,   Execute:=_bool_in_,   Pozisyonu:=_real_in_,   Velocity:=_real_in_,   Done=&gt;_bool_out_,   Busy=&gt;_bool_out_,   CommandAborted=&gt;_bool_out_,   Error=&gt;_bool_out_,   ErrorID=&gt;_word_out_,   ErrorInfo=&gt;_word_out_);</pre>	<p>Eksenin mutlak bir pozisyona konumlandırma hareketini başlatmak için MC_MoveAbsolute komutunu kullanınız.</p> <p>MC_MoveAbsolute komutunu kullanmak için, eksen ilk önce etkinleştirilmeli ve özgüdümlenmelidir.</p>

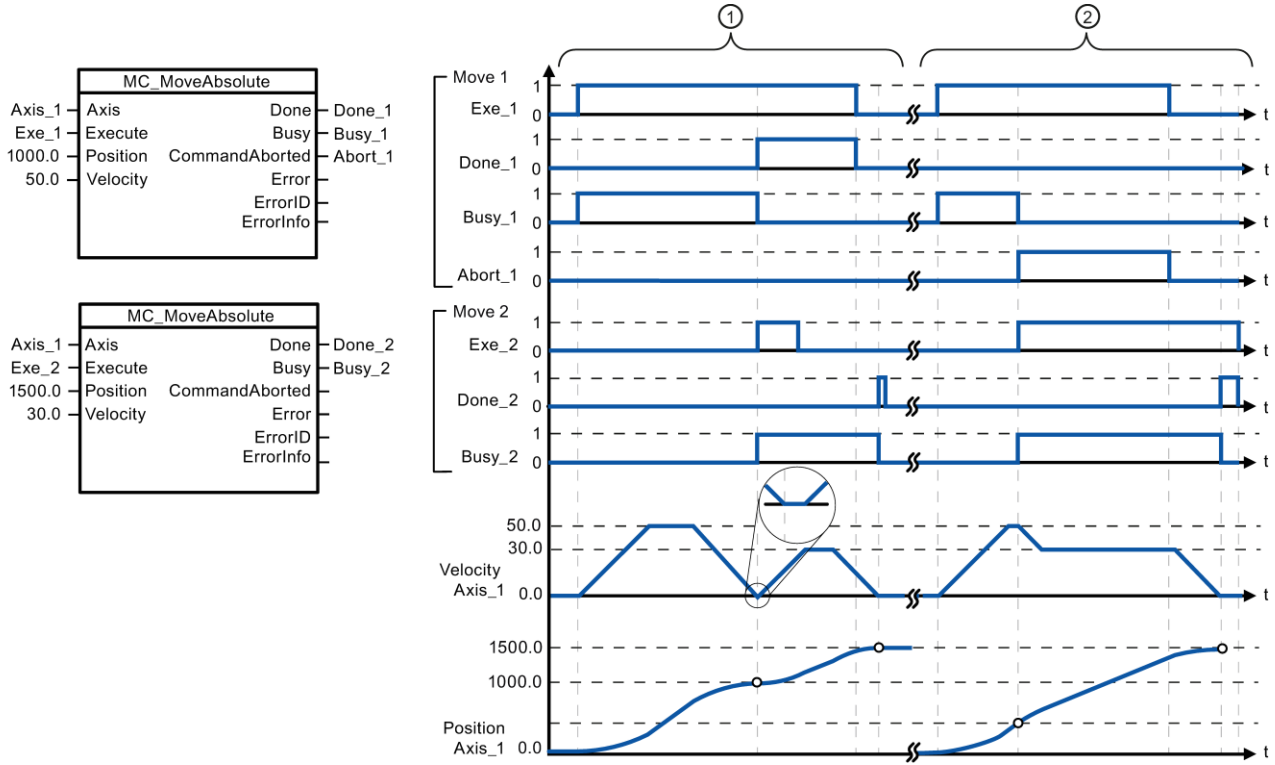
- 1 Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.
- 2 SCL örneğinde , "MC\_MoveAbsolute\_DB" kopya DB'nin ismidir.

Çizelge 9- 38 MC\_MoveAbsolute komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama
Axis	IN	TO_Axis_1 Eksen teknoloji nesnesi
Execute	IN	Bool Görevi pozitif kenar ile başlatma (Varsayılan değer: False)
Pozisyonu	IN	Real Mutlak hedef pozisyonu (Varsayılan değer: 0.0) Limit değerler: $-1.0e^{12} \leq \text{Pozisyon} \leq 1.0e^{12}$
Velocity	IN	Real Eksen hızı (Varsayılan değer: 10.0) Yapılandırılmış hızlanma ve vavaşlama ve vaklasılacak hedef pozisyon nedeniyle bu hıza her zaman ulaşılmaz. Limit değerler: $\text{Start/stop hızı} \leq \text{Hız} \leq \text{maksimum hız}$
Done	OUT	Bool TRUE = Mutlak hedef pozisyonuna ulaşıldı
Busy	OUT	Bool TRUE = Görev yürütülüyor.



Parametre ve tipi	Veri tipi	Açıklama	
CommandAborted	OUT	Bool	TRUE = Yürütme sırasında görev başka bir görev tarafından iptal edildi.
Error	OUT	Bool	TRUE = Yürütme sırasında bir hata oluştu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir.
ErrorID	OUT	Word	"Error" parametresi için Error ID. (Varsayılan değer: 0000)
ErrorInfo	OUT	Word	"ErrorID" parametresi için Error info ID. (Varsayılan değer: 0000)



Aşağıdaki değerler "Dynamics> General" yapılandırma penceresinde yapılandırılmıştır: Hızlanma = 10.0 ve Yavaşlama = 10.0

- ① Bir eksen MC\_MoveAbsolute görevi ile 1000,0 mutlak pozisyonuna taşınır. Eksen hedef konuma ulaştığında, bu "Done\_1" üzerinden bildirilir. "Done\_1" = TRUE olduğunda, 1500,0 hedef pozisyonu ile başka MC\_MoveAbsolute görevi başlatılır. Tepki süreleri (örneğin, kullanıcı programı cevrim süresi vb.) nedeniyle, eksen (zoomed-in'e ayrıntılı bakınız) kısa sürede hareketsiz kalır. Eksen yeni hedef konuma ulaştığında, bu "Done\_2" ile sinyellenir.
- ② Bir aktif MC\_MoveAbsolute görevi başka MC\_MoveAbsolute görevi tarafından iptal edilir. İptal "Abort\_1" üzerinden bildirilir. Eksen sonra 1500,0 yeni hedef konumuna yeni bir hızda taşınır. Yeni hedef pozisyonuna ulaştığında, bu "Done\_2" üzerinden bildirilir.

### Geçersiz kılma tepkisi

MC\_MoveAbsolute görevi aşağıdaki hareket kontrol görevleriyle iptal edilebilir:

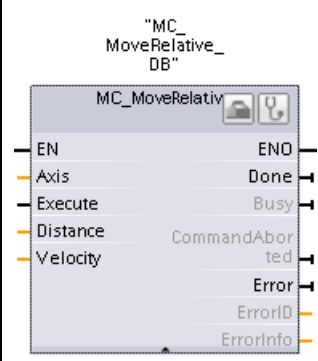
- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

MC\_MoveAbsolute görevi aşağıdaki hareket kontrol görevlerini iptal eder:

- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

### 9.3.5.7 MC\_MoveRelative (Eksen bağıl olarak konumla) komutu

Çizelge 9- 39 MC\_MoveRelative komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a function block named 'MC_MoveRelative_DB'. It has several inputs: EN, Axis, Execute, Distance, and Velocity. It has several outputs: Done, Busy, CommandAborted, Error, ErrorID, and ErrorInfo. The block is connected to a network with EN and Execute inputs, and Done, Busy, CommandAborted, Error, ErrorID, and ErrorInfo outputs.</p>	<pre>"MC_MoveRelative_DB" (   Axis:=_multi_fb_in_,   Execute:=_bool_in_,   Distance:=_real_in_,   Velocity:=_real_in_,   Done=&gt;_bool_out_,   Busy=&gt;_bool_out_,   CommandAborted=&gt;_bool_out_,   Error=&gt;_bool_out_,   ErrorID=&gt;_word_out_,   ErrorInfo=&gt;_word_out_);</pre>	<p>Başlatma pozisyonuna göre hareketi konumlandırmayı başlatmak için MC_MoveRelative komutunu kullanınız.</p> <p>MC_MoveRelative komutunu kullanmak için, eksen ilk önce etkinleştirilmelidir.</p>

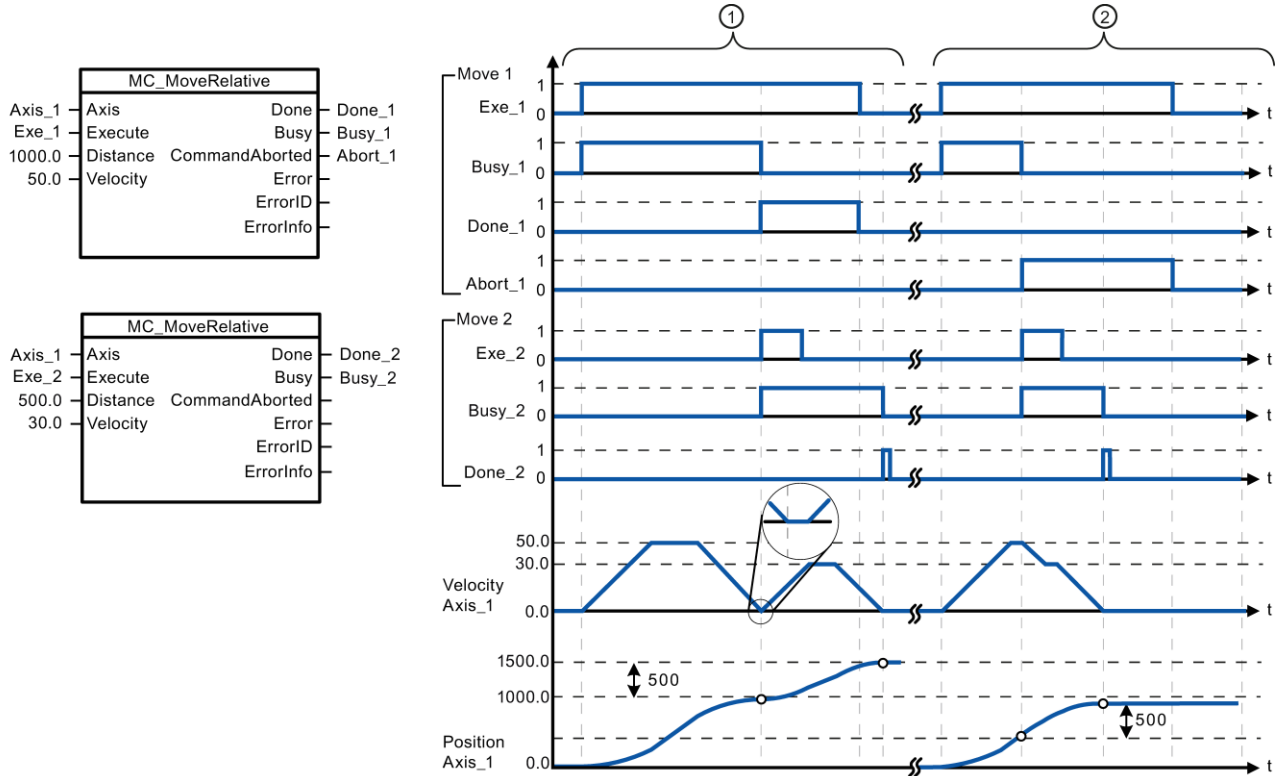
<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

<sup>2</sup> SCL örneğinde, "MC\_MoveRelative\_DB" kopya DB'nin ismidir.

Çizelge 9- 40 MC\_MoveRelative komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama
Axis	IN	TO_Axis_1
Execute	IN	Bool
Distance	IN	Real
Velocity	IN	Real
Done	OUT	Bool
Busy	OUT	Bool

Parametre ve tipi	Veri tipi	Açıklama	
CommandAborted	OUT	Bool	TRUE = Yürütme sırasında görev başka bir görev tarafından iptal edildi.
Error	OUT	Bool	TRUE = Yürütme sırasında bir hata oluştu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir.
ErrorID	OUT	Word	"Error" parametresi için Error ID. (Varsayılan değer: 0000)
ErrorInfo	OUT	Word	"ErrorID" parametresi için Error info ID. (Varsayılan değer: 0000)



Aşağıdaki değerler "Dynamics> General" yapılandırma penceresinde yapılandırılmıştır: Hızlanma = 10.0 ve Yavaşlama = 10.0

- ① Bir eksen MC\_MoveRelative görevi ile 1000,0 mesafeye ("Distance") taşınır. Eksen hedef konuma ulaştığında, bu "Done\_1" üzerinden bildirilir. "Done\_1" = TRUE olduğunda, alınan mesafe 500.0 ile başka MC\_MoveRelative görevi başlatılır. Tepki süreleri (örneğin, kullanıcı programı cevrim süresi vb.) nedeniyle, eksen (zoomed-in'e ayrıntılı bakınız) kısa sürede hareketsiz kalır. Eksen yeni hedef konuma ulaştığında, bu "Done\_2" ile sinyallerin.
- ② Bir aktif MC\_MoveRelative görevi başka MC\_MoveAbsolute görevi tarafından iptal edilir. İptal "Abort\_1" üzerinden bildirilir. Eksen sonra 500,0 yeni mesafeye ("Distance") yeni bir hızda taşınır. Yeni hedef pozisyonuna ulaşıldığında, bu "Done\_2" üzerinden bildirilir.

### Geçersiz kılma tepkisi

MC\_MoveRelative görevi aşağıdaki hareket kontrol görevleriyle iptal edilebilir:

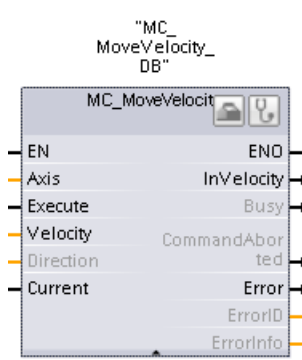
- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

Yeni MC\_MoveRelative görevi aşağıdaki hareket kontrol görevlerini iptal eder:

- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

### 9.3.5.8 MC\_MoveVelocity (Ekseni önceden belirlenen hızda hareket ettir) komutu

Çizelge 9- 41 MC\_MoveVelocity komutu

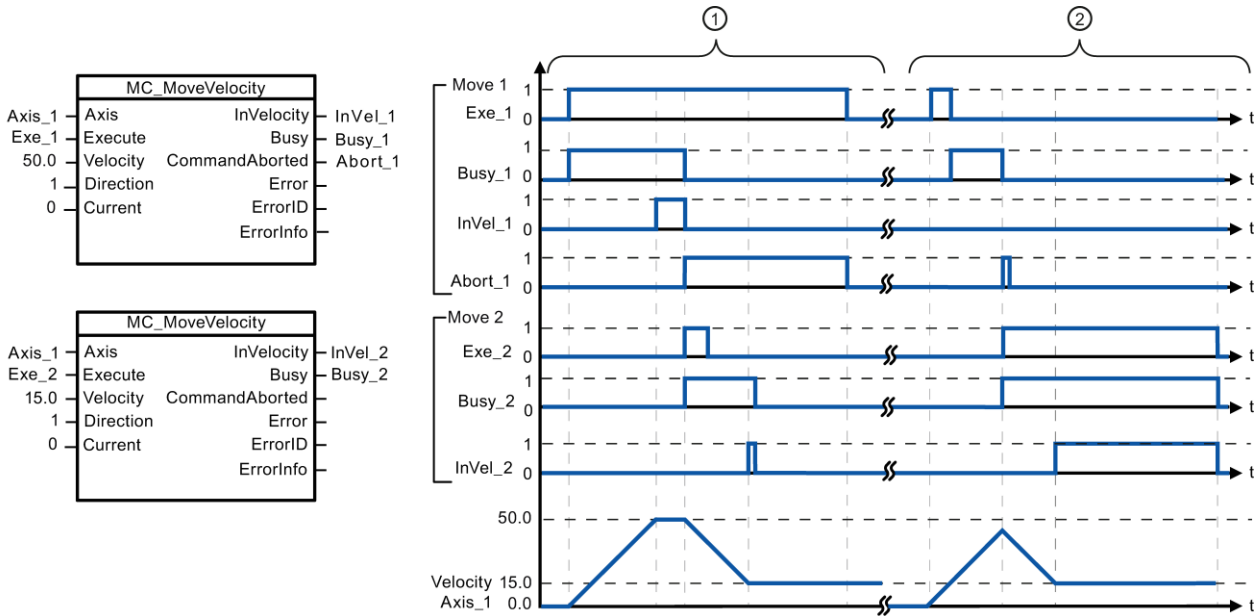
LAD / FBD	SCL	Açıklama
	<pre>"MC_MoveVelocity_DB" (   Axis:= _multi_fb_in_,   Execute:= _bool_in_,   Velocity:= _real_in_,   Direction:= _int_in_,   Current:= _bool_in_,   InVelocity=&gt; _bool_out_,   Busy=&gt; _bool_out_,   CommandAborted=&gt; _bool_out_,   Error=&gt; _bool_out_,   ErrorID=&gt; _word_out_,   ErrorInfo=&gt; _word_out_ );</pre>	<p>Belirtilen hızda sürekli olarak eksen hareket ettirmek için MC_MoveVelocity komutunu kullanınız.</p> <p>MC_MoveVelocity komutunu kullanmak için eksen ilk önce etkinleştirilmelidir.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

<sup>2</sup> SCL örneğinde , "MC\_MoveVelocity\_DB " kopya DB'nin ismidir.

Çizelge 9- 42 MC\_MoveVelocity komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama	
Axis	IN	TO_Axis_1	Eksen teknoloji nesnesi
Execute	IN	Bool	Görevi pozitif kenar ile başlatma (Varsayılan değer: False)
Velocity	IN	Real	Eksen hareketi için hız özelliği (Varsayılan değer: 10.0) Limit değerler: Start/stop hızı $\leq$  Hız  $\leq$ maksimum hız (Hız = 0.0 izin verilir)
Direction	IN	Int	Yön özelliği: <ul style="list-style-type: none"> <li>0: Dönme yönü "Velocity" parametre değerinin (Varsayılan değer) işaretine karşılık gelir</li> <li>1: Pozitif dönme yönü (Velocity" parametre değerinin işareti göz ardı edilir)</li> <li>2: Negatif dönme yönü (Velocity" parametre değerinin işareti göz ardı edilir.)</li> </ul>
Current	IN	Bool	Geçerli hızı koruma: <ul style="list-style-type: none"> <li>FALSE: "Maintain current velocity" devre dışı edilir. "Velocity" ve "Direction" parametrelerinin değerleri kullanılır. (Varsayılan değer)</li> <li>TRUE: "Maintain current velocity" aktifleştirilir. "Velocity" vd "Direction" parametrelerinin değerleri hesaba katılmaz.</li> </ul> Eksen geçerli hızında hareketine devam ettiğinde, "InVelocity" parametre değeri TRUE ya döner.
InVelocity	OUT	Bool	TRUE: <ul style="list-style-type: none"> <li>"Current" = FALSE ise: "Velocity" parametresinde belirtilen hıza ulaşıldı.</li> <li>"Current" = TRUE ise: Başlangıç zamanında geçerli hızda eksen hareketi.</li> </ul>
Busy	OUT	Bool	TRUE = Görev yürütülüyor.
CommandAborted	OUT	Bool	TRUE = Yürütme sırasında görev başka bir görev tarafından iptal edildi.
Error	OUT	Bool	TRUE = Yürütme sırasında bir hata oluştu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir.
ErrorID	OUT	Word	"Error" parametresi için Error ID. (Varsayılan değer: 0000)
ErrorInfo	OUT	Word	"ErrorID" parametresi için Error info ID. (Varsayılan değer: 0000)



Aşağıdaki değerler "Dynamics> General" yapılandırma penceresinde yapılandırılmıştır: Hızlanma = 10.0 ve Yavaşlama = 10.0

- ① Etkin bir MC\_MoveVelocity görevi hedef hıza ulaştığını "InVel\_1" aracılığıyla sinyaller. Daha sonra başka bir MC\_MoveVelocity görevi tarafından iptal edilir. İptal "Abort\_1" üzerinden bildirilir. Yeni hedef hıza 15.0 ulaşıldığında, bu "InVel\_2" üzerinden bildirilir. Eksen sonra yeni sabit bir hızla hareket etmeye devam eder.
- ② Bir aktif MC\_MoveVelocity görevi, hedef hıza ulaşmadan önceki başka MC\_MoveVelocity görevi tarafından iptal edilir. İptal "Abort\_1" üzerinden bildirilir. Yeni hedef hıza 15.0 ulaşıldığında, bu "InVel\_2" üzerinden bildirilir. Eksen sonra yeni sabit bir hızla hareket etmeye devam eder.

#### Geçersiz kılma tepkisi

MC\_MoveVelocity görevi aşağıdaki hareket kontrol görevleriyle iptal edilebilir:

- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

Yeni MC\_MoveVelocity görevi aşağıdaki hareket kontrol görevlerini iptal eder:

- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

**Not**
**Sıfır ayar hızı (Hız = 0.0) ile davranış**

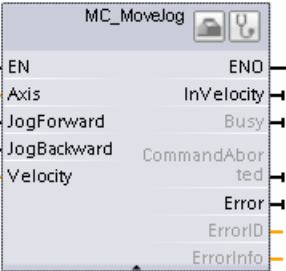
"Velocity" = 0.0 (MC\_Halt görevi gibi) ile MC\_MoveVelocity görevi aktif hareket görevlerini iptal eder ve yapılandırılmış yavaşlama ile eksen durdurur. Eksen, hareketsiz hale geldiğinde, çıkış parametresi "InVelocity" en azından bir program çevrimi için TRUE gösterir.

"Busy" yavaşlama işlemi sırasında değer TRUE belirtir ve " InVelocity" ile birlikte FALSE durumuna değişir. Parametre " Execute " = TRUE ayarlı ise, "InVelocity" ve " Busy " oturtulur.

MC\_MoveVelocity görevi başlatıldığında, teknoloji nesnesinde durum biti "SpeedCommand" ayarlanır. Eksenin hareketsiz kalması üzerine Durum biti "ConstantVelocity" ayarlanır. Yeni bir hareket görev başlatıldığında iki bit yeni duruma adapte edilir.

**9.3.5.9 MC\_MoveJog (Jog modunda eksen hareket ettir) komutu**

Çizelge 9- 43 MC\_MoveJog komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows the MC_MoveJog block with the following inputs and outputs:</p> <ul style="list-style-type: none"> <li>Inputs: EN, Axis, JogForward, JogBackward, Velocity</li> <li>Outputs: InVelocity, Busy, CommandAborted, Error, ErrorID, ErrorInfo</li> </ul>	<pre>"MC_MoveJog_DB" (   Axis:=_multi_fb_in_,   JogForward:=_bool_in_,   JogBackward:=_bool_in_,   Velocity:=_real_in_,   InVelocity=&gt;_bool_out_,   Busy=&gt;_bool_out_,   CommandAborted=&gt;_bool_out_,   Error=&gt;_bool_out_,   ErrorID=&gt;_word_out_,   ErrorInfo=&gt;_word_out_);</pre>	<p>Jog modunda belirtilen hızda eksen sürekli hareket ettirmek için MC_MoveJog komutunu kullanınız. Bu komut genellikle test amaçlı ve devreye alma için kullanılır. MC_MoveJog komutunu kullanmak için eksen ilk önce etkinleştirilmelidir.</p>

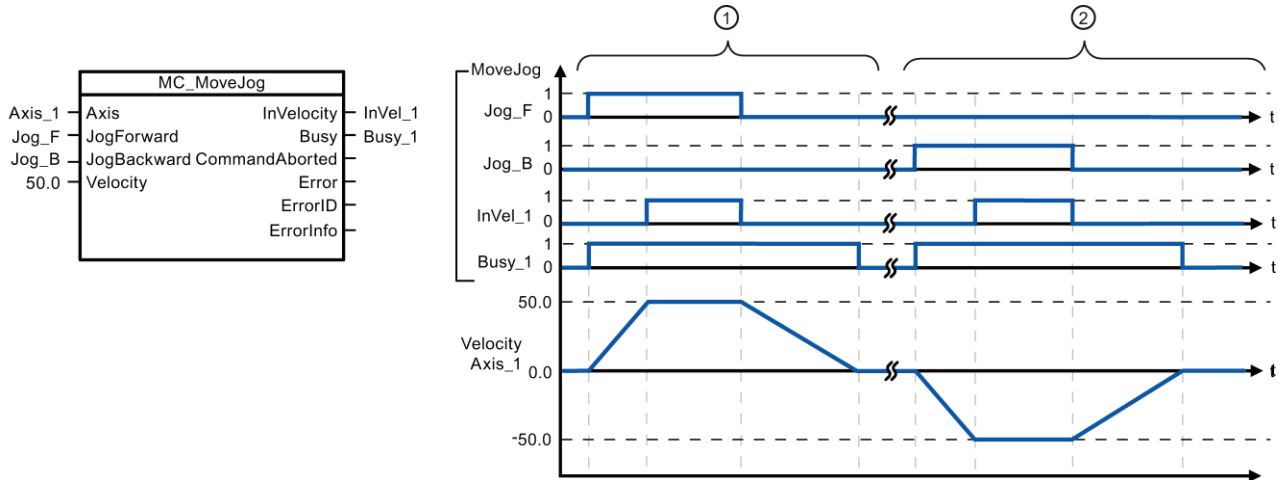
<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

<sup>2</sup> SCL örneğinde , "MC\_MoveJog\_DB " kopya DB'nin ismidir.

Çizelge 9- 44 MC\_MoveJog komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama
Axis	IN	TO_Axis_1
JogForward <sup>1</sup>	IN	Bool
JogBackward <sup>1</sup>	IN	Bool
Velocity	IN	Real
InVelocity	OUT	Bool
Busy	OUT	Bool
CommandAborted	OUT	Bool
Error	OUT	Bool
ErrorID	OUT	Word
ErrorInfo	OUT	Word

<sup>1</sup> Hem JogForward hemde JogBackward parametreleri aynı anda TRUE ise, eksen yapılandırılmış yavaşlama ile durur. "Error", "ErrorID" ve "ErrorInfo" parametrelerinde bir hata belirtilir.



Aşağıdaki değerler "Dynamics> General" yapılandırma penceresinde yapılandırılmıştır: Hızlanma = 10.0 ve Yavaşlama = 5.0

- ① Eksen, "Jog\_F" üzerinden adım modunda pozitif yönde hareket eder. Hedef hız 50.0'a ulaştığında, bu, "InVelo\_1" üzerinden bildirilir. Jog\_F sıfırlandıktan sonra eksen tekrar hareketsiz hale doğru frenler.
- ② Eksen, "Jog\_B" üzerinden adım modunda negatif yönde hareket eder. Hedef hız 50.0'a ulaştığında, bu, "InVelo\_1" üzerinden bildirilir. Jog\_B sıfırlandıktan sonra eksen tekrar hareketsiz hale doğru frenler.



### Geçersiz kılma tepkisi

MC\_MoveJog görevi aşağıdaki hareket kontrol görevleriyle iptal edilebilir:

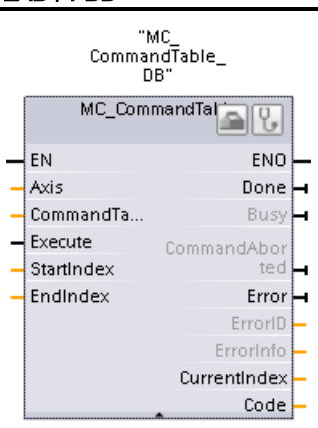
- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

Yeni bir MC\_MoveJog görevi aşağıdaki hareket kontrol görevlerini iptal eder:

- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog

### 9.3.5.10 MC\_CommandTable (Eksen komutlarını hareket dizisi olarak çalıştırma) komutu

Çizelge 9- 45 MC\_CommandTable komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows the MC_CommandTable block with the following inputs and outputs:</p> <ul style="list-style-type: none"> <li>Inputs: EN, Axis, CommandTable, Execute, StartIndex, EndIndex</li> <li>Outputs: END, Done, Busy, CommandAborted, Error, ErrorID, ErrorInfo, CurrentIndex, Code</li> </ul>	<pre>"MC_CommandTable_DB" (   Axis:= _multi_fb_in_,   CommandTable:= _multi_fb_in_,   Execute:= _bool_in_,   StartIndex:= _uint_in_,   EndIndex:= _uint_in_,   Done=&gt; _bool_out_,   Busy=&gt; _bool_out_,   CommandAborted=&gt; _bool_out_,   Error=&gt; _bool_out_,   ErrorID=&gt; _word_out_,   ErrorInfo=&gt; _word_out_,   CurrentIndex=&gt; _uint_out_,   Code=&gt; _word_out_ );</pre>	<p>Bir hareket dizisi içine birleşebilen bir motor kontrol eksenini için bireysel hareketler serisini yürütür. Bireysel hareketler darbe katarı çıkışı için bir teknoloji nesnesi komut çizelgesinde yapılandırılır (TO_CommandTable_PTO).</p>

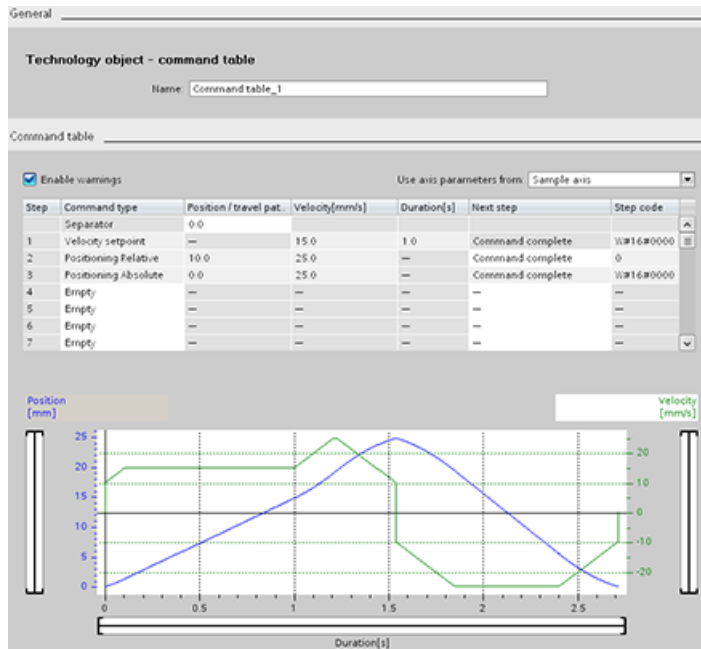
- 1 Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.
- 2 SCL örneğinde , "MC\_CommandTable\_DB " kopya DB'nin ismidir.

Çizelge 9- 46 MC\_CommandTable komutu parametreleri

Parametre ve tipi	Veri tipi	Başlangıç değeri	Açıklama
Axis	IN	TO_Axis_1	Eksen teknoloji nesnesi
Table	IN	TO_CommandTable_1	Komut çizelgesi teknoloji nesnesi
Execute	IN	Bool	İşi yükselen kenar ile başlat
StartIndex	IN	Int	Komut çizelgesi prosesini bu adımla başlat Limitler: $1 \leq \text{StartIndex} \leq \text{EndIndex}$
EndIndex	IN	Int	Komut çizelgesi prosesini bu adımla bitir Limitler: $\text{StartIndex} \leq \text{EndIndex} \leq 32$

Parametre ve tipi	Veri tipi	Başlangıç değeri	Açıklama	
Done	OUT	Bool	FALSE	MC_CommandTable prosesi başarıyla tamamlandı
Busy	OUT	Bool	FALSE	İşlem devam ediyor
CommandAborted	OUT	Bool	FALSE	Görev, proses sırasında başka bir görev ile iptal edildi.
Error	OUT	Bool	FALSE	Proses sırasında hata oluştu. Hatanın nedeni ErrorID ve ErrorInfo parametreleriyle gösterilir
ErrorID	OUT	Word	16#0000	Hata tanımlayıcı
ErrorInfo	OUT	Word	16#0000	Hata bilgisi
Step	OUT	Int	0	Adım hali hazırda devam ediyor
Code	OUT	Word	16#0000	Adımın kullanıcı tanımlı tanılayıcısı hali hazırda devam ediyor

"Komut Çizelgesi" yapılandırma penceresinde istenilen hareket dizisini oluşturabilirsiniz ve eğilim şemasında grafik görünümüne göre sonucu kontrol edebilirsiniz.



Komut çizelgesini işlemek için kullanılacak komut tiplerini seçebilirsiniz. Maksimum 32'ye kadar iş girilebilir. Komutlar sırayla işlenir.

Çizelge 9- 47 MC\_CommandTable komut tipleri

Komut tipi	Açıklama
Empty	Empty, eklenecek tüm komutlar için bir yer tutucu olarak hizmet verir. Empty girişi komut çizelgesi işlendiğinde yok sayılır
Halt	Eksenı durdurur. Not: Komut, yalnızca bir "Velocity setpoint" komutundan sonra gerçekleşir.
Positioning Relative	Eksenini mesafeye dayalı olarak konumlandırır. Komut, verilen mesafe ve hıza göre eksenı hareket ettirir.

Komut tipi	Açıklama
Positioning Absolute	Ekseni konuma dayalı olarak konumlandırır. Komut, belirtilen hız kullanılarak verilen konuma göre eksen hareket ettirir.
Velocity setpoint	Ekseni verilen hızda hareket ettirir.
Wait	Verilen süre bitene kadar bekler. "Wait" aktif bir dönme hareketini durdurmaz.
Separator	Secilen çizgi üzerinde bir "ayırıcı" bir çizgi ekler. Ayırıcı çizgi, birden fazla profilin, tek bir komut çizelgesinde tanımlanmasına izin verir.

MC\_CommandTable yürütmesi için önkoşullar:

- Teknoloji nesnesi TO\_Axis\_PTO V2.0 doğru yapılandırılmalıdır.
- Teknoloji nesnesi TO\_CommandTable\_PTO doğru yapılandırılmalıdır.
- Eksen serbest bırakılmalıdır.

#### Geçersiz kılma tepkisi

MC\_CommandTable görevi aşağıdaki hareket kontrol görevleriyle iptal edilebilir:

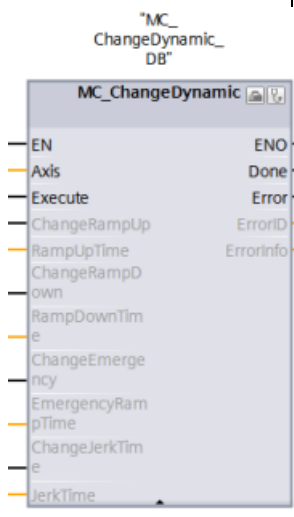
- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog
- MC\_CommandTable

Yeni MC\_CommandTable görevi aşağıdaki hareket kontrol görevlerini iptal eder:

- MC\_Home Mode = 3
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative
- MC\_MoveVelocity
- MC\_MoveJog
- MC\_CommandTable
- İlk " Positioning Relative ", " Positioning Absolute ", " Velocity setpoint " ya da "Halt" komutunu başlatarak geçerli hareket kontrol işi

### 9.3.5.11 MC\_ChangeDynamic (Eksenin dinamik ayarlarını değiştir) komutu

Çizelge 9- 48 MC\_ChangeDynamic komutu

LAD / FBD	SCL	Açıklama
	<pre>"MC_ChangeDynamic_DB" (   Execute:=_bool_in_,   ChangeRampUp:=_bool_in_,   RampUpTime:=_real_in_,   ChangeRampDown:=_bool_in_,   RampDownTime:=_real_in_,   ChangeEmergency:=_bool_in_,   EmergencyRampTime:=_real_in_,   ChangeJerkTime:=_bool_in_,   JerkTime:=_real_in_,   Done=&gt;_bool_out_,   Error=&gt;_bool_out_,   ErrorID=&gt;_word_out_,   ErrorInfo=&gt;_word_out_);</pre>	<p>Hareket kontrol ekseninin dinamik ayarlarını değiştirir:</p> <ul style="list-style-type: none"> <li>ramp-up zaman değerini değiştir (hızlanma)</li> <li>ramp-down zaman değerini değiştir (yavaşlama)</li> <li>Acil durdurma ramp-down zaman değerini değiştir (acil durum yavaşlaması)</li> <li>Yumuşatma zaman değerini değiştir (sarsıntı)</li> </ul>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

<sup>2</sup> SCL örneğinde , "MC\_ChangeDynamic\_DB " kopya DB'nin ismidir.

Çizelge 9- 49 MC\_ChangeDynamic komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama
Axis	IN	TO_Axis_1
Execute	IN	Bool
ChangeRampUp	IN	Bool
RampUpTime	IN	Real
ChangeRampDown	IN	Bool
RampDownTime	IN	Real
ChangeEmergency	IN	Bool

Parametre ve tipi		Veri tipi	Açıklama
EmergencyRampTime	IN	Real	Fkseni acil durdurma modunda sarsıntı sınırlayıcı olmadan, vabılandırılmıs maksimum hızdan hareketsiz duruma yavařlatma zamanı (saniye). Varsayılan deęer: 2.00 Deęiřiklik <Axis name>. Config.DynamicDefaults. EmergencyDeceleration etiketini etkileyecektir. Deęiřiklięin etkisi bu etiketin aıklamasında gsterilmiřtir.
ChangeJerkTime	IN	Bool	TRUE = "JerkTime"parametresiyle aynı doęrultuda yumuřatma zamanını deęiřtir. Varsayılan deęer: FALSE
JerkTime	IN	Real	eksenin hızlanma ve yavařlama rampaları iin kullanılan yumuřatma zamanı (saniye). Varsayılan deęer: 0.25 Deęiřiklik <Axis name>. Config.DynamicDefaults. Jerk etiketini etkileyecektir. Deęiřiklięin etkisi bu etiketin aıklamasında gsterilmiřtir.
Done	OUT	Bool	TRUIF = Deęiřtirilen deęerler teknoloii veri blođuna vazılmıřtır. etiketlerin aıklamasında Deęiřiklięin ne zaman yrrlęe girdięi gsterilecektir. Varsayılan deęer: FALSE
Error	OUT	Bool	TRUE = Komutun yrtlmesi sırasında bir hata oluřtu. Hatanın nedeni "ErrorID" ve "ErrorInfo" parametrelerinde bulunabilir. Varsayılan deęer: FALSE
ErrorID	OUT	Word	Hata tanımlayıcı. Varsayılan deęer: 16#0000
ErrorInfo	IN	Word	Error bilgisi. Varsayılan deęer: 16#0000

MC\_ChangeDynamic yrtmesi ile ilgili n kořullar:

- Teknoloji nesnesi TO\_Axis\_PTO V2.0 doęru biimde yapılandırılmalıdır.
- Eksen serbest bırakılmalıdır.

### Geersiz kılma tepkisi

MC\_ChangeDynamic komutu bařka bir hareket kontrol komutu tarafından iptal edilemez.

Yeni bir MC\_ChangeDynamic komutu bařka bir hareket kontrol komutunu iptal etmez.

### Not

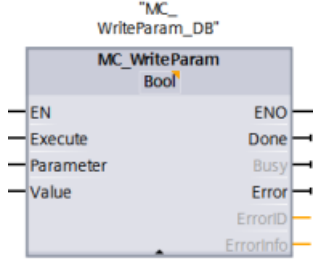
"RampUpTime", "RampDownTime", "EmergencyRampTime" ve "RoundingOffTime" giriř parametreleri, " acceleration ", " delay ", " emergency stop-delay " ve "sarsıntı" sonu eksen parametrelerini izin verilen sınırlar dıřına atan deęerler ile belirtilebilir.

MC\_ChangeDynamic parametrelerini Emin eksen teknolojisi nesnesi iin dinamik yapılandırma ayarlarına ait sınırlar iinde tuttuęunuzdan emin olunuz.

### 9.3.5.12 MC\_WriteParam (teknolojisi nesnesi parametrelerini yaz) komutu

Kullanıcı programından eksenin işlevselliğini değiştirecek seçilen parametre sayısını yazmak için MC\_WriteParam komutunu kullanınız.

Çizelge 9- 50 MC\_WriteParam komutu

LAD / FBD	SCL	Açıklama
	<pre>"MC_WriteParam_DB" (     Parameter:=_variant_in_,     Value:=_variant_in_,     Execute:=_bool_in_,     Done:=_bool_out_,     Error:=_real_out_,     ErrorID:=_word_out_,     ErrorInfo:=_word_out_);</pre>	<p>(Örneğin, hızlanma ve kullanıcı DB değerleri) genel parametrelere yazmak için MC_WriteParam komutunu kullanınız.</p>

- 1 Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.
- 2 SCL örneğinde , "MC\_WriteParam\_DB" kopya DB'nin ismidir.

Genel parametrelere yazabilirsiniz. "MotionStatus" ve "StatusBits" parametrelerine yazamazsınız. Geçerli parametreler aşağıdaki çizelgede listelenmektedir:

Yazılabilir parametre ismi	Yazılabilir parametre ismi
Actor.InverseDirection	DynamicDefaults.Acceleration
Actveya.DirectionMode	DynamicDefaults.Deceleration
Actor.DriveParametre.PulsesPerDriveRevolution	DynamicDefaults.Jerk
Sensor[1].ActiveHoming.Mode	DynamicDefaults.EmergencyDeceleration
Sensor [1].ActiveHoming.SidelInput	PozisyonuLimitsHW.Active
Sensor [1].ActiveHoming.Offset	PozisyonuLimitsHW.MaxSwitchedLevel
Sensor [1].ActiveHoming.SwitchedLevel	PozisyonuLimitsHW.MinSwitchedLevel
Sensor [1].PassiveHoming.Mode	PozisyonuLimitsSW.Active
Sensveya[1].PassiveHoming.SidelInput	PozisyonuLimitsSW.MinPozisyonu
Sensor [1].PassiveHoming.SwitchedLevel	PozisyonuLimitsSW.MaxPozisyonu
Units.UzunlukUnit	Homing.Autveyaeversal
Mechanics.LeadScrew	Homing.ApproachDirection
DynamicLimits.MinVelocity	Homing.ApproachVelocity
DynamicLimits.MaxVelocity	Homing.ReferencingVelocity

Çizelge 9- 51 MC\_WriteParam komut parametreleri

Parametre ve tipi		Veri tipi	Açıklama
PARAMNAME	IN	Variant	Değerin yazıldığı parametre ismi
VALUE	IN	Variant	Atanmış parametreye yazılacak değer
EXECUTE	IN	Bool	Komutu başlat. Varsayılan değer: FALSE
DONE	OUT	Bool	Değer yazıldı. Varsayılan değer: FALSE
BUSY	OUT	Bool	TRUE ise, komut çalışıyor. Varsayılan değer: FALSE
ERROR	OUT	Real	TRUE ise, bir hata oluştu. Varsayılan değer: FALSE
ERRORID	OUT	Word	Hata ID'si
ERRORINFO	OUT	Word	ERRORID ile ilgili bilgi

Çizelge 9- 52 ERRORID ve ERRORINFO için durum kodları

ERRORID (W#16#...)	ERRORINFO (W#16#...)	Açıklama
0	0	Eksen TO-DB başarılı parametre değişimi
8410 <sub>[1]</sub>	0028 <sub>[1]</sub>	Geçersiz parametre ayarla (yanlış uzunlukta Eksen TO-DB parametresi )
8410 <sub>[1]</sub>	0029 <sub>[1]</sub>	Geçersiz parametre ayarla (Axis TO-DB parametresi yok)
8410 <sub>[1]</sub>	002B <sub>[1]</sub>	Geçersiz parametre ayarla (read-only Axis TO-DB parametre)
8410 <sub>[1]</sub>	002C <sub>[1]</sub>	Geçerli parametre ayarla, ancak eksen devre dışı bırakılmamış
Config Error <sub>[2]</sub>	Config Error <sub>[2]</sub>	Geçerli parametre ayarla (genel salt okunur Eksen TO-DB parametresi) aralık dışı
Config Error <sub>[3]</sub>	Config Error <sub>[3]</sub>	Geçerli parametre ayarla (genel Eksen TO-DB parametresi) aralık dışı

[1] MC\_WriteParam' da hata

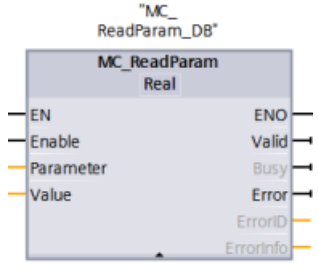
[2] MC\_Power da hata

[3] MC\_Power ve MC\_MoveXXX veya MC\_CommandTable da hata

### 9.3.5.13 MC\_ReadParam (teknoloji nesnesi parametrelerini oku) komutu

Eksen girişinde tanımlanan eksenin mevcut konumunu, hızını ve benzerlerini gösteren seçilen sayıda parametreyi okumak için MC\_ReadParam komutunu kullanınız.

Çizelge 9- 53 MC\_ReadParam komutu

LAD / FBD	SCL	Açıklama
	<pre>"MC_ReadParam_DB" (   Enable:=_bool_in_,   Parameter:=_variant_in_,   Value:=_variant_in_out_,   Valid:=_bool_out_,   Busy:=_bool_out_,   Error:=_real_out_,   ErrorID:=_word_out_,   ErrorInfo:=_word_out_);</pre>	<p>Çevrim kontrol noktasından bağımsız olarak tekli durum değerlerini okumak için MC_ReadParam komutunu kullanınız.</p>

- 1 Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.
- 2 SCL örneğinde , "MC\_ReadParam\_DB " kopya DB'nin ismidir.

MC\_ReadParam komutu "enable" davranışı üzerinde çalışır. "Enable" girişi true olduğu sürece komutu belirtilen "Parametre" yi "Value" depolama konumuna okur.

Her Çevrim Kontrol Noktasında (CCP) "MotionStatus" " Position " değeri, geçerli HSC değerine dayalı olarak güncellenir.

"MotionStatus", " Velocity " değeri, mevcut segment (güncellendi ~10 ms) sonundaki komut hızıdır. MC\_ReadParam da bu değeri okuyabilir.

Bir hata oluşursa, komut, giriş "Enable" da sadece yeni yükselen bir kenar ile sıfırlanabilen bir hata durumuna geçer.

Çizelge 9- 54 MC\_ReadParam komut parametreleri

Parametre ve tipi	Veri tipi	Açıklama	
ENABLE	IN	Bool	Komutu başlat. Varsayılan değer: FALSE
PARAMETER	IN	Variant	Okunacak TO-parametresine işaretçi
VALID	OUT	Bool	TRUE ise, değer okundu. Varsayılan değer: FALSE
BUSY	OUT	Bool	TRUE ise, komut çalışıyor. Varsayılan değer: FALSE
ERROR	OUT	Real	TRUE ise, bir hata oluştu. Varsayılan değer: FALSE
ERRORID	OUT	Word	Hata ID'si. Varsayılan değer: 0
ERRORINFO	OUT	Word	ERRORID ile ilgili bilgi. Varsayılan değer: 0
VALUE	INOUT	Variant	Okunan değer saklanacağı konuma işaretçi



Çizelge 9- 55 ERRORID ve ERRORINFO için durum kodları

ERRORID (W#16#...)	ERRORINFO (W#16#...)	Açıklama
0	0	Başarılı parametre okuması
8410	0028	Geçersiz parametre (yanlış uzunluk)
8410	0029	Geçersiz parametre (TO-DB yok)
8410	0030	Geçersiz parametre (okunabilir değil)
8411	0032	Geçersiz parametre (yanlış değer)

## TO parametreleri

Eksen "MotionStatus" dört değerden oluşur. Program çalışırken okunabilir bu değerlerdeki değişiklikleri izlemek isteyeceksiniz:

Değişken ismi	Veri tipi	MC_ReadParam aracılığıyla okunabilir
Hareket durumu:	Structure	Hayır
• Pozisyon	REAL	Evet
• Hız	REAL	Evet
• Mesafe	REAL	Evet
• Hedef Pozisyon	REAL	Evet

## 9.3.6 S7-1200 için hareket kontrolünün çalışması

### 9.3.6.1 Hareket kontrolü için kullanılan CPU çıkışları

CPU dört darbe çıkış jeneratörü sağlar. Her darbe çıkış jeneratörü bir adet darbe çıkışı ve bir step motor sürücüsü veya darbe ara yüzü bir servo motor sürücüsü kontrol için bir adet yön çıkışı sağlar. Darbe çıkışı motor hareketi için gerekli darbelerle sahip sürücü sağlar. Yön çıkışı, sürücünün hareket yönünü kontrol eder.

PTO çıkışı değişken frekanslı bir kare dalga çıkışı üretir. Darbe üretimi ve H / W vanılandırma ve / veya SFC'ler / SFB'ler üzerinden sağlanan yapılandırma veya yürütme bilgisiyle kontrol edilir.

CPU RUN modunda iken, kullanıcının seçimine davalı olarak değerler ve döndürü vuzmacında saklanır veya darbe jeneratörü çıkışları dijital çıkışlarını sürer. STOPmodunda, PTO jeneratör, çıkışları kontrol etmez.

Yerleşik CPU çıkışları ve bir sinval kartı çıkışları darbe ve yön çıkışları olarak kullanılabilir. "Özellikler" sekmesinde Darbe jeneratörleri (PTO / PWM) altında Cihaz Yapılandırması sırasında sinval kartının çıkışları ile onboard CPU çıkışları arasında seçim yaparsınız. Sadece PTO (Darbe Katar Çıkışı) hareket kontrolü için geçerlidir.

Aşağıdaki çizelge varsayılan I/O atamalarını gösterir; Bununla birlikte, dört pals jeneratörü herhangi bir dijital çıkış için yapılandırılabilir.

**Not**

**Darbe-katarı çıkışları kullanıcı programında diğer komutlar tarafından kullanılamaz.**

Darbe jeneratörleri olarak CPU veya sinyal kartının çıkışlarını yapılandırdığınızda (PWM veya hareket kontrol komutları ile kullanım için), karşılık gelen çıkış adresleri artık çıkışları kontrol etmez. Kullanıcı programınız bir darbe jeneratörü olarak kullanılan bir çıkışa bir değer yazarsa, CPU, fiziksel çıkışa o değeri yazmaz.

**Not**

**PTO yön çıkışları programınızda başka bir yerde kullanılmak üzere serbest olabilir.**

Her PTO, iki çıkış ataması gerektirir: biri bir darbe çıkışı olarak ve diğeri bir yön çıkışı olarak. Yön çıkışını değil sadece darbe çıkışını kullanabilirsiniz. Daha sonra kullanıcı programınızda diğer amaçlar için yön çıkışını serbest bırakabilirsiniz. Çıkış, her iki PTO yön çıkışı için ve kullanıcı programında aynı anda kullanılamaz.

Çizelge 9- 56 Darbenin ve yön çıkışlarının varsayılan adres atamaları

Hareket kontrolü için çıkışların kullanımı		
	Darbe	Yön
PTO1		
Gömülü I/O	Q0.0	Q0.1
SB I/O	Q4.0	Q4.1
PTO2		
Gömülü I/O	Q0.2	Q0.3
SB I/O	Q4.2	Q4.3
PTO3		
Gömülü I/O	Q0.4 <sup>1</sup>	Q0.5 <sup>1</sup>
SB I/O	Q4.0	Q4.1
PTO4		
Gömülü I/O	Q0.6 <sup>2</sup>	Q0.7 <sup>2</sup>
SB I/O	Q4.2	Q4.3

<sup>1</sup> CPU 1211C, Q0.4, Q0.5, Q0.6 veya Q0.7 çıkışlarına sahip değildir. Bu nedenle, bu çıkışlar CPU 1211C de kullanılamaz.

<sup>2</sup> CPU 1212C, Q0.6 veya Q0.7 çıkışlarına sahip değildir. Bu nedenle, bu çıkışlar CPU 1212C de kullanılamaz.

<sup>3</sup> Bu Çizelge CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C ve CPU 1217C PTO fonksiyonları için geçerlidir.

## Sürücü ara yüzü

Hareket kontrolü için "Sürücü etkin" ve "Sürücü hazır" için sürücü ara yüzünü isteğe bağlı olarak yapılandırabilirsiniz. Sürücü ara yüzünü kullanırken, "sürücü hazır" için sürücü etkinleştirme ve dijital girişle ilgili dijital çıkış serbestçe seçilebilir.

### Not

PTO (Darbe Katarı Çıkışı) seçilmiş ve bir eksene atanmışsa firmware, karşılık gelen darbe ve yön çıkışları aracılığıyla kontrolü ele alacaktır.

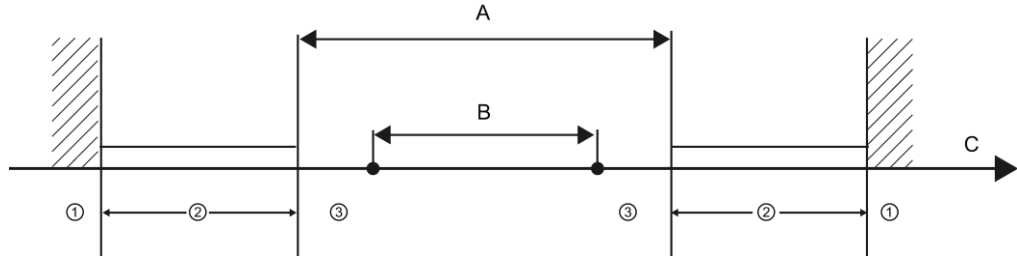
Kontrol fonksiyonunun ele alınmasıyla birlikte, proses görüntü ile I/O çıkışı arasındaki bağlantı kesilir. Kullanıcı, kullanıcı programı veya izleme çizelgesi aracılığıyla proses darbe görüntüsünü ve yön çıkışlarını yazma olasılığına sahipken, bu asla I/O çıkışına aktarılmaz. Buna göre, kullanıcı programı veya izleme çizelgesi aracılığıyla I/O çıkışını izlemek mümkün değildir. Okunan bilgi sadece proses görüntü değerini yansıtır ve herhangi bir bakımından I/O çıkışının gerçek durumuyla eşleşmez.

CPU firmware tarafından kalıcı olarak kullanılmayan bütün diğer CPU çıkışları için I/O çıkışının durumu, her zamanki gibi proses görüntüsü vasıtasıyla kontrol edilebilir veya izlenebilir.

### 9.3.6.2

#### Hareket kontrolü için donanım ve yazılım sınır anahtarları

Ekseninizin "İzin verilen hareket aralığı" ve "çalışma aralığını" sınırlamak için donanım ve yazılım sınır anahtarlarını kullanınız.



① Mekanik durdurucu

② Alt ve üst donanım sınırları

③ Alt ve üst yazılım sınırları

A Eksen için izin verilen hareket aralığı

B Eksenin çalışma aralığı

C Mesafe

Donanım ve yazılım sınır anahtarları, konfigürasyonda veya kullanıcı programında kullanmadan önce aktif edilmelidir. Yazılım sınır anahtarları sadece eksen özgüdümlendikten sonra aktif olur.

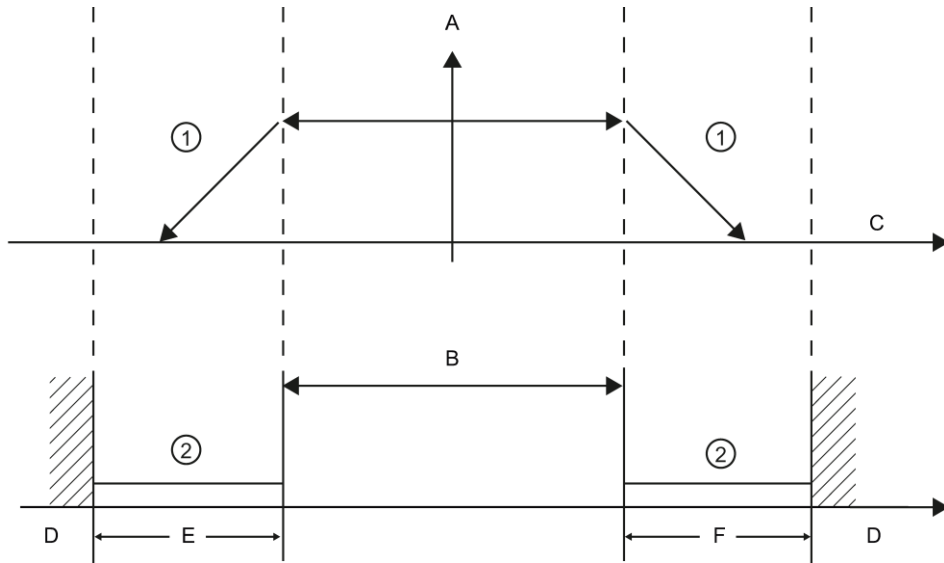
### Donanım sınır anahtarları

Donanım sınır anahtarları, eksenin maksimum hareket aralığını belirler. Donanım sınır anahtarları, CPU'nun kesme yeteneğine sahip girişlerine bağlı olması gereken fiziksel anahtarlama elemanlarıdır. Yaklaşıldıktan sonra sadece kalıcı olarak anahtarlanmış şekilde kalan donanım sınır anahtarlarını kullanınız. Bu anahtarlama durumu, yalnızca izin verilen hareket aralığına döndükten sonra iptal edilebilir.

Çizelge 9- 57 Donanım sınırları için kullanılabilir girişler

Açıklama	RPS	LIM-	LIM+
Gömülü I/O		10.0 - 11.5	
SB I/O		14.0 - 14.3	

Donanım sınır anahtarlarına yaklaşıldığında, yapılandırılan acil yavaşlamada eksen, hareketsiz duruma frenleme yapar. Belirtilen acil yavaşlama, mekanik durdurucudan önce eksenin güvenilir şekilde durduracak kadar yeterli olmalıdır. Aşağıdaki diyagram ekseninin, donanım sınır anahtarlarına yaklaştıktan sonraki davranışını göstermektedir.



- ① Yapılandırılmış acil yavaşlamada eksen, hareketsiz duruma frenler.
- ② Donanım sınır anahtarlarının "yaklaştı" durumunu sinyallediği aralık
- A [Hız]
- B İzin verilen hareket aralığı
- C Mesafe
- D Mekanik durdurucu
- E Alt donanım sınır anahtarı
- F Üst donanım sınır anahtarı

**⚠ UYARI****Dijital giriş kanalı için filtre süresi değişiklikleri ile riskler**

Bir dijital giriş kanalı için filtre süresi bir önceki ayara göre değiştirilmesi durumunda, filtre yeni girişlere tamamen duyarlı hale gelmeden önce, yeni bir "0" seviye giriş değerinin toplam 20.0 ms'ye kadar bir süre için sunulması gerekebilir. Bu süre boyunca, 20.0 ms'den daha az süreli kısa "0" darbeleri olaylar algılanamayabilir veya sayılamayabilir.

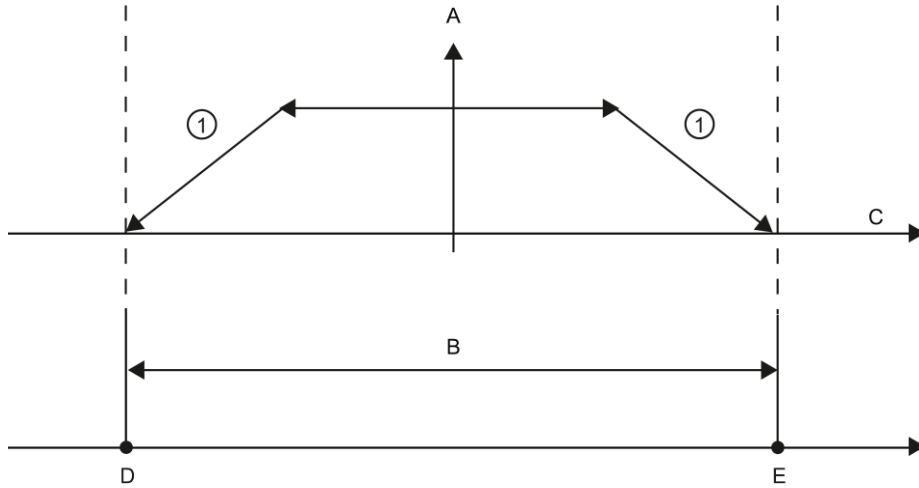
Filtre süresi değişikliği, personelin ölüm veya ciddi yaralanmalarına neden olabilen ve/veya ekipmana zararı olabilen beklenmeyen makine veya işlem operasyonuna neden olabilir.

Yeni bir filtre süresinin derhal etkili olduğundan emin olmak için CPU'nun güç çevrimi uygulanmalıdır.

**Yazılım sınır anahtarları**

Yazılım sınır anahtarları eksenin "çalışma aralığını" sınırlar. Bu anahtarlar hareket aralığına göre donanım sınır anahtarlarının içinde kalmalıdır. Yazılım sınır anahtarlarının pozisyonları esnek bir şekilde ayarlanabildiği için eksen çalışma aralığı geçerli dönme profiline bağlı olarak bireysel bazda sınırlı olabilir. Donanım sınır anahtarlarının aksine, yazılım sınır anahtarları yalnızca yazılım vasıtasıyla uygulanabilir ve kendi anahtarlama elemanlarına ihtiyaç duymaz.

Yazılım sınır anahtarları aktif ise, aktif bir hareket yazılım sınır anahtarının konumunda durdurulur. Eksen, yapılandırılmış yavaşlamada frenlenir. Eksen yazılım sınırı anahtarlarına ulaşana kadar aşağıdaki diyagram, ekseninin davranışını göstermektedir.



① yapılandırılmış acil yavaşlamada eksen hareketsiz duruma frenler.

A [Hız]

B Çalışma aralığı

C Mesafe

D Alt donanım sınır anahtarı

E Üst donanım sınır anahtarı

Mekanik kapama, yazılım sınır anahtarlarından sonra konumlandırılmışsa ve mekanik hasar riski varsa, ek donanım sınır anahtarları kullanınız.

## İlave bilgi

Kullanıcı programınız hem donanım hem de yazılım limitleri işlevselliğini etkinleştirerek veya devre dışı bırakarak donanım veya yazılım pozisyon limitlerini geçersiz kılabilirsiniz. Seçim Eksen DB'den yapılır.

- Donanım sınır işlevselliğini etkinleştirmek veya devre dışı bırakmak için, DB yolu "<axis name> / Config / PositonLimits\_HW" de "Aktif" etiketine (Bool) erişin. "Aktif" etiketinin durumu, donanım pozisyon limitlerinin kullanımını etkinleştirir veya devre dışı bırakır.
- Yazılım sınır işlevselliğini etkinleştirmek veya devre dışı bırakmak için DB yolu "<axis name> / Config / PositonLimits\_SW" de "Aktif" etiketine (Bool) erişiniz. "Aktif" etiketinin durumu, yazılım pozisyon limitlerinin kullanımını etkinleştirir veya devre dışı bırakır.

Ayrıca (örneğin makine kurulumuna esneklik katmak için veya makine değişim zamanını kısaltmak için) kullanıcı programı ile yazılım pozisyon sınırlarını değiştirebilirsiniz. Kullanıcı programınız DB "<axis adı> / Config / PositionLimits\_SW" de "MinPosition" ve "MaxPosition" etiketlerine (Real formatında mühendislik birimleri) yeni değerler yazabilirsiniz.

### 9.3.6.3 Homing (özüdümlleme)

Homing, eksen koordinatlarının gerçek, fiziksel sürücü konumuyla eşleşmesi anlamına gelir. (Sürücü, hali hazırda x pozisyonunda ise, eksen, x pozisyonunda olacak şekilde ayarlanacaktır.) Pozisyon kontrollü eksenler için pozisyona ait kayıtlar ve görüntüler bu eksen koordinatlarıyla tam olarak uyuşur.

---

#### Not

Eksen koordinatları ve gerçek durum arasındaki uyuşma son derece önemlidir. Bu adım, eksenin mutlak hedef pozisyonunun, ayrıca sürücü ile tam olarak elde edilmesinin sağlanması açısından gereklidir.

---

MC\_Home komutu, eksenin homing işlemini başlatır.

4 farklı homing işlevi vardır. İlk iki fonksiyon kullanıcının, ekseninin mevcut konumunu ve ikinci iki pozisyon eksenini Başlangıç referans sensörüne göre ayarlamasını sağlar.

- Mod 0 - Mutlak Doğrudan Referans: Çalıştırıldığında bu mod eksenin tam olarak nerede olduğunu söyler. Bu, dahili pozisyon değişkenini, Homing komutunun Pozisyon giriş değerine ayarlar. Bu, makine kalibrasyonu ve kurulumu için kullanılır.

Eksen pozisyonu, referans noktası anahtarından bağımsız olarak ayarlanır. Aktif döndürme hareketleri iptal edilmez. MC\_Home komutu pozisyon giriş parametresinin değeri eksenin referans noktası olarak hemen ayarlanır. Tam mekanik bir konuma referans noktasını atamak için, eksen homing işlemi sırasında bu pozisyonda hareketsiz olarak durmalıdır.

- Mod 1 - Bağlı Doğrudan Referans: Çalıştırıldığında bu mod dahili pozisyon değişkeni kullanır ve buna Homing komutunun üzerindeki Pozisyon giriş değerini ekler. Bu, genellikle makine ofsetini hesaba katmak için kullanılır.

Eksen pozisyonu, referans noktası anahtarından bağımsız olarak ayarlanır. Aktif döndürme hareketleri iptal edilmez. Aşağıdaki ifade homing sonrası eksen pozisyonu için geçerlidir: Yeni eksen pozisyonu = geçerli eksen konumunu + MC\_Home komutunun pozisyonu parametre değeri.

- Mod 2 - Pasif Referanslama: eksen hareket ederken ve Referans Noktası Anahtarını geçerken geçerli konum başlangıç konum olarak ayarlanır. Bu özellik, normal makine aşınmasının ve dişli tepkisinin hesaba katılmasına yardımcı olur ve aşınma için manuel telafi ihtiyacını engeller. Homing komutunun üzerindeki Pozisyon girişi, daha önce olduğu gibi, kolay Ev pozisyon ofsetine izin veren Referans Noktası Anahtarı tarafından belirtilen konuma ekler.

Pasif homing sırasında MC\_Home komutu herhangi bir homing hareketini gerçekleştirmez. Bu adım için gerekli olan dönme hareketi diğer hareket kontrol komutları aracılığıyla kullanıcı tarafından uygulanmalıdır. Referans noktası anahtarı algılandığında, eksen, konfigürasyona göre özgüdümlenir. Aktif döndürme hareketleri pasif homing işleminin başlatılması üzerine iptal edilmez.

- Mod 3 - Aktif Referanslama: Bu mod eksenin homing yapmanın en hassas yöntemidir. Başlangıç yönü ve hareketin hızı Teknoloji Nesne Konfigürasyonu Genişletilmiş Parametreler-Homing de yapılandırılır. Bu makine konfigürasyonuna bağlıdır. Referans Noktası Anahtarı sinyalinin yükselen kenarı veya düşen kenarı Ana pozisyon olup olmadığını belirleme yeteneği de vardır. Hemen hemen tüm sensörler aktif bir aralığa sahiptir ve Steady State On pozisyonu Ana sinyal olarak kullanılması durumunda, On sinyali aktif aralığı, mesafe aralığını kapsayacağı için homing pozisyonunda hata olma olasılığı olacaktır. Sinyalin yükselen ya da düşen kenarını kullanarak çok daha hassas Ana konum sonuçları elde edilir. Diğer tüm modlarda olduğu gibi Homing komutu üzerinde Pozisyon giriş değeri, Donanım referanslı pozisyona ilave edilir.

Aktif homing modunda MC\_Home komutu gerekli referans noktası yaklaşımını gerçekleştirir. Referans noktası anahtarı algılandığında, eksen, yapılandırmaya göre özgüdümlenir. Aktif döndürme hareketleri iptal edilir.

Modu 0 ve 1, hiç bir suretle eksenin hareket ettirilmesini gerektirmez. Bunlar genellikle kurulum ve kalibrasyonda kullanılmaktadır. Mod 2 ve 3, eksenin hareket etmesini ve Referans Noktası Anahtarı olarak "Eksen" teknoloji nesnesinde yapılandırılmış bir sensörü geçmesini gerektirir. Referans noktası, eksenin çalışma alanı içine veya hareket aralığı içinde kalmak kaydıyla normal çalışma alanının dışına yerleştirilebilir.

### Homing parametrelerinin yapılandırılması

"Homing" yapılandırma penceresinde aktif ve pasif homing için parametreleri yapılandırınız. Homing yöntemi hareket kontrol komutunun "Mode" giriş parametresi kullanılarak ayarlanır. Burada, Mode = 2 pasif homing, Mode = 3 aktif homing anlamına gelir.

#### Not

Makinenin yön değiştirme olayında mekanik durducuya kadar hareket etmediğinden emin olmak için aşağıdaki önlemlerden birini kullanınız:

- Yaklaşım hızını düşük tutunuz
- Yapılandırılmış hızlanma / yavaşlamayı artırınız
- Donanım sınır anahtarı ve mekanik durak arasındaki mesafeyi artırınız

Çizelge 9- 58 Eksenin homing yapılması için yapılandırma parametreleri

Parametre	Açıklama
Giriş referans noktası anahtarı (Aktif ve Pasif homing)	Açılır liste kutusundan referans noktası anahtarı için dijital giriş seçiniz. Giriş kesme özelliğine sahip olmalıdır. Yerleştirilen bir sinyal kartının girişleri ve yerleşik CPU girişleri referans noktası anahtarı girişi olarak seçilebilir.  Dijital girişler için varsayılan filtre süresi 6.4 ms. Dijital girişler, referans noktası anahtarı olarak kullanıldığı zaman, bu, istenmeyen yavaşlamalara ve dolayısıyla hatalara sebep olabilir. Referans noktası anahtarının kapsamına ve azaltılmış hıza bağlı olarak, referans noktası algılanamayabilir. Filtre süresi, dijital girişlerin cihaz yapılandırmasında "Giriş filtresi" altında ayarlanabilir.  Belirtilen filtre süresi, referans noktası anahtarındaki giriş sinyalinin süresinden daha az olmalıdır.
Donanım sınır anahtarlarına ulaştıktan sonra otomatik geriye çevirme (yalnızca Aktif homing)	Referans noktası yaklaşımında bir yönlendirici kam olarak Donanım sınır anahtarını kullanmak için onay kutusunu etkinleştiriniz. Donanım sınır anahtarları yapılandırılmalı ve yön değiştirme için etkinleştirilmelidir.  Donanım sınır anahtarına aktif homing sırasında ulaşılması durumunda, eksen yapılandırılmış yavaşlamada (acil yavaşlama ile değil) frenler ve yön değiştirir. Referans noktası anahtarı sonra ters yönde algılanır.  Yön değiştirme aktif değilse ve eksen aktif homing sırasında donanım sınır anahtarına ulaşırsa, referans noktası yaklaşımı bir hata ile iptal edilir ve eksen acil yavaşlamada frenler.
Yaklaşım yönü (Aktif ve Pasif homing)	Yön seçimi ile referans noktası anahtarını aramak için aktif homing sırasında kullanılan "yaklaşım yönü" nün yanı sıra homing yönünü de belirlersiniz. Homing yönü, homing işlemini yürütmek amacıyla referans noktası anahtarının yapılandırılmış tarafına yaklaşım için eksenin kullandığı hareket yönünü belirtir.



Parametre	Açıklama
Referans nokta anahtarı (Aktif ve Pasif homing)	<ul style="list-style-type: none"> <li>Aktif homing: Eksenin, referans noktası anahtarının sol veya sağ tarafında referanslanacak olup olmadığını belirleyiniz. Eksenin başlangıç pozisyonuna ve homing parametrelerinin yapılandırılmasına bağlı olarak, referans noktası yaklaşım dizisi yapılandırma penceresindeki şemadan farklı olabilir.</li> <li>Pasif homing: Pasif homing ile homing amaçları bakımından dönme hareketleri hareket komutları ile kullanıcı tarafından uygulanması gerekir. Referans noktası anahtarının Homing'in olduğu tarafı, aşağıdaki faktörlere bağlıdır:               <ul style="list-style-type: none"> <li>"Yaklaşım yönü" yapılandırması</li> <li>"Referans noktası anahtarı" yapılandırması</li> <li>Pasif homing sırasında geçerli hareket yönü</li> </ul> </li> </ul>
Yaklaşım hızı (Yalnızca Aktif homing)	Referans noktası yaklaşımı sırasında referans noktası anahtarının aranacağı hızı belirtiniz. Sınır değerleri (seçilen kullanıcı ünitesinden bağımsız): Start / Stop hızı ≤ yaklaşım hızı ≤ maksimum hız
Azaltılmış hız (Yalnızca Aktif homing)	Eksenin, homing için referans noktası anahtarına yaklaştığı hızı belirtiniz. Sınır değerleri (seçilen kullanıcı ünitesinden bağımsız): Start / stop hız ≤ azaltılmış hız ≤ maksimum hız
Home pozisyon ofseti (Yalnızca Aktif homing)	İstenilen referans pozisyonu, referans noktası anahtarının pozisyonundan saparsa, home pozisyon ofseti bu alanda belirtilebilir. Değer 0'a eşit değilse, eksen, referans noktası anahtarında homing sonrasında aşağıdaki eylemleri yürütür: 79. Azaltılmış hızda eksen home pozisyon ofset değeri kadar taşıyınız. 80. Home pozisyon ofset pozisyonuna ulaşıldığında, eksen pozisyonu mutlak referans konumuna ayarlanır. Mutlak referans konumu "MC_Home" hareket kontrol komutunun "pozisyon" parametresi aracılığıyla belirtilir. Sınır değerleri (seçilen kullanıcı ünitesinden bağımsız): -1.0e12 ≤ home pozisyon ofset ≤ 1.0e12

Çizelge 9- 59 Homing'i etkileyen faktörler

Etki eden faktörler:			Sonuç:
Konfigürasyon Yaklaşım yönü	Konfigürasyon Referans nokta anahtarı	Geçerli hareket yönü	Referans nokta anahtarı üzerinde Homing
Pozitif	"Left (Negatif) kenar"	Pozitif yön	Sol
		Negatif yön	Sağ
Pozitif	"Right (Pozitif) kenar "	Pozitif yön	Sağ
		Negatif yön	Sol
Negatif	"Sol (Negatif) kenar "	Pozitif yön	Sağ
		Negatif yön	Sol
Negatif	"Sağ (Pozitif) kenar "	Pozitif yön	Sol
		Negatif yön	Sağ

### Aktif homing dizisi

Hareket kontrol komutu "MC\_Home" ile aktif homing başlatırsınız (giriş parametresi Mode = 3). Bu durumda Giriş parametresi "Position" mutlak referans noktası koordinatlarını belirir. Alternatif olarak, test amaçları için kontrol panelinde aktif homing başlatabilirsiniz.

Aşağıdaki diyagram aşağıdaki yapılandırma parametreleri ile aktif bir referans noktası yaklaşımı için bir karakteristik eğrinin bir örneğini göstermektedir:

- " Yaklaşım yönü " = "Pozitif Yaklaşım yönü"
- " Referans noktası anahtarı " = " Sağ (pozitif) tarafı "
- " home pozisyon ofset" değeri > 0

Çizelge 9- 60 MC homing için hız karakteristikleri

İşlem		Notlar	
		A	Yaklaşım hızı
		B	Azaltılmış hız
		C	Home pozisyon koordinatı
		D	Home pozisyon ofseti
①	Arama fazı (mavi eğri segment): Aktif hedef başladığında, eksen, yapılandırılmış "yaklaşım hızı" na hızlanır ve referans noktası anahtarı için bu hızda arama yapar.		
②	Referans noktası yaklaşımı (kırmızı eğri bölümü): Referans noktası anahtarı algılandığında, yapılandırılmış "azaltılmış hız" da referans noktası anahtarının yapılandırılmış tarafında özgüdümlenmek üzere bu örnekte eksen frenler ve tersine döner.		
③	Referans nokta pozisyonuna (yeşil eğri segmenti) hareket: Referans noktası anahtarındaki homing işleminden sonra, eksen "azaltılmış hız" da "Referans noktası koordinatları" na gider. "Referans noktası koordinatları" na ulaşıldığında, eksen "MC_Home" komutunun pozisyon giriş parametresinde belirtilmiş olan konum değerinde durdurulur.		

### Not

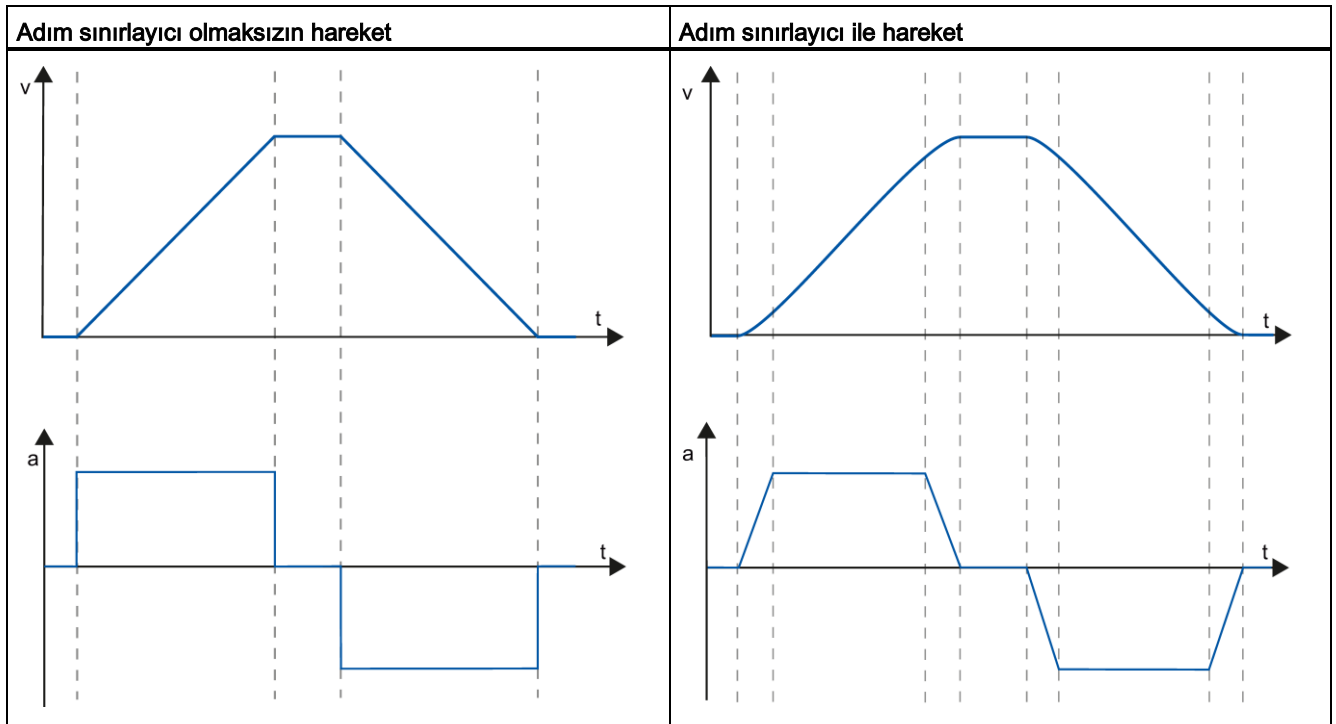
Beklendiği gibi homing arama çalışmazsa, donanım sınırlarına veya referans noktasına atanan girişleri kontrol ediniz. Bu girişler, bunların kenar kesmelerini cihaz yapılandırmasında devre dışı etmiş olabilir.

"HW Low Limit Switch Input", "HW High Limit Switch Input" ve "Input reference point switch" için hangi girişlerin atandığını (varsa) görmek için söz konusu eksen teknoloji nesnesi için yapılandırma verilerini inceleyiniz. Sonra CPU için cihaz yapılandırmasını açınız ve atanan girişlerin her birini inceleyiniz. "Enable rising edge detection" ve "Enable falling edge detection" her ikisinin seçili olduğunu doğrulayınız. Bu özellikler seçili değilse, eksen yapılandırmasında belirtilen girişleri silin ve onları tekrar seçiniz.

### 9.3.6.4 Sarsıntı sınırı

Sarsıntı sınırı ile bir hızlanma ve yavaşlama rampası sırasında mekaniklerinizin üzerindeki gerilmeleri azaltabilirsiniz. Adım sınırlayıcı aktif olduğunda hızlanma ve yavaşlama değeri birdenbire değişmez; bir geçiş aşamasına uyarlanır. Aşağıdaki şekil sarsıntı sınırı olan veya olmayan hız ve ivme eğrisini göstermektedir.

Çizelge 9- 61 Sarsıntı sınırı



Sarsıntı sınırı, eksen hareketinin bir "yumuşatılmış" hız profilini verir. Bu, örneğin bir konveyör kayışının yumuşak yol verilmesi ve frenlemesini sağlar.

### 9.3.7 Devreye alma

#### "Status and error bits" tanılama fonksiyonu

Eksen için en önemli durum ve hata mesajlarını izlemek için " Status and error bits " tanılama işlevini kullanınız. Eksen aktif olduğunda tanılama fonksiyonu, ekran "Manuel kontrol" modunda ve "otomatik kontrol" de çevrimiçi modda kullanılabilir.

Çizelge 9- 62 Eksenin durumu

Durum	Açıklama
Etkin	Eksen etkinleştirildi ve hareket kontrol görevleri üzerinden kontrol edilmeye hazırdır. (Teknoloji nesnenin Etiket: <Axis name> StatusBits.Enable)
Homed	Eksen, özgüdümlendi ve "MC_MoveAbsolute" hareket kontrol komutunun mutlak konumlandırma görevlerini yürütme yeteneğine sahiptir. Eksen, bağlı homing için özgüdümlenme zorunda değildir. Özel durumlar: <ul style="list-style-type: none"> <li>Aktif homing sırasında, durum FALSE.</li> <li>Özgüdümlenmiş bir eksen pasif homing olmuşsa, durum pasif homing sırasında TRUE olarak ayarlanır.</li> </ul> (Teknoloji nesnesi etiketi: <Axis name> StatusBits.HomingDone)
Hata	"Eksen" teknoloji nesnesinde bir hata oluştu. Hata hakkında daha fazla bilgi, hareket kontrol komutlarının ErrorID ve ErrorInfo parametrelerinde otomatik kontrolde mevcuttur. Manuel modda, kontrol panelinin " Last error " alanı hata nedeni hakkında detaylı bilgiler gösterir. (Teknoloji nesne etiketi: <Axis name>. StatusBits.Error)
Aktif kontrol paneli	"Manuel kontrol" modu, kontrol panelinde etkinleştirildi. Kontrol panelinin, "Eksen" teknoloji nesnesine göre kontrol önceliği vardır. Eksen, kullanıcı programından kontrol edilemez. (Teknoloji nesne etiketi: <Axis name>. StatusBits.ControlPanelActive)

Çizelge 9- 63 Sürücü durumu

Durum	Açıklama
Sürücü hazır	Sürücü çalışmaya hazır. (Teknoloji nesne etiketi: <Axis name>. StatusBits.DriveReady)
Hata	Sürücü, hazır sinyal hatasından sonra bir hata bildirdi. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.DriveFault)

Çizelge 9- 64 Eksen hareketinin durumu

Durum	Açıklama
Hareketsiz	Eksen hareketsiz. (Teknoloji nesne etiketi: <Axis name>.StatusBits.StandStill)
Hızlanıyor	Eksen hızlanıyor. (Teknoloji nesne etiketi: <Axis name>.StatusBits.Acceleration)

Durum	Açıklama
Sabit hız	Eksen sabit hızda hareket ediyor. (Teknoloji nesne etiketi: <Axis name>.StatusBits.ConstantVelocity)
Yavaşlıyor	Eksen yavaşlıyor (hızı azalıyor). (Teknoloji nesne etiketi: <Axis name>.StatusBits.Deceleration)

Çizelge 9- 65 Hareket modunun durumu

Durum	Açıklama
Konumlama	Eksen, hareket kontrol komutu "MC_MoveAbsolute" veya "MC_MoveRelative" ya da kontrol panelinin bir konumlandırma görevini yürütür. (Teknoloji nesne etiketi: <Axis name>.StatusBits.Positioning Command)
Hız komutu	Eksen hareket kontrol komutu "MC_MoveVelocity" veya "MC_MoveJog" ya da kontrol panelinin ayarlanan hızında bir görevi yürütür. (Teknoloji nesne etiketi: <Axis name>.StatusBits.SpeedCommand)
Homing	Eksen hareket kontrol komutu "MC_Home" ya da kontrol panelinin bir homing görevini yürütür. (Teknoloji nesne etiketi: <Axis name>.StatusBits.Homing)

Çizelge 9- 66 Hata bitleri

Hata	Açıklama
Min yazılım sınırına ulaşıldı	Alt yazılım sınır anahtarına ulaşıldı. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.SwLimitMinReached)
Min yazılım sınırı aşıldı	Alt yazılım sınır anahtarı aşıldı. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.SwLimitMinExceeded)
Max yazılım sınırına ulaşıldı	Üst yazılım sınır anahtarına ulaşıldı. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.SwLimitMaxReached)
Max yazılım sınırı aşıldı	Üst yazılım sınır anahtarı aşıldı. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.SwLimitMaxExceeded)
Negatif donanım sınırı	Alt donanım sınır anahtarına yaklaşıldı. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.HwLimitMin)
Pozitif donanım sınırı	Üst donanım sınır anahtarına yaklaşıldı. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.HwLimitMax)
PTO çoktan kullanılıyor	İkinci eksen aynı PTO kullanıyor ve "MC_Power" ile etkinleştirildi. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.HwUsed)
Yapılandırma hatası	"Eksen" teknoloji nesnesi, yanlış yapılandırılmış veya düzenlenebilir yapılandırma verileri kullanıcı programının çalışma zamanı sırasında yanlış değiştirilmiştir. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.ConfigFault)
Genel Hata	Dahili bir hata oluştu. (Teknoloji nesne etiketi: <Axis name>.ErrorBits.SystemFault)

**"Motion status" tanılama fonksiyonu**

Eksenin hareket durumunu izlemek için " Motion status " tanılama işlevini kullanınız. Eksen aktif olduğunda tanılama fonksiyon ekranı, "Manuel kontrol" modunda ve "otomatik kontrol" de çevrimiçi modda kullanılabilir.

Çizelge 9- 67 Hareket durumu

Durum	Açıklama
Hedef pozisyon	" Hedef pozisyon " alanı hareket kontrolü komutu "MC_MoveAbsolute" veya "MC_MoveRelative" ya da kontrol panelinin etkin bir konumlandırma görevinin geçerli hedef konumunu gösterir. "Target position" değeri yalnızca bir konumlandırma görevinin yürütülmesi sırasında geçerlidir. (Teknoloji nesne etiketi: <Axis name>. MotionStatus. TargetPosition)
Geçerli pozisyon	" Geçerli pozisyon " alanı mevcut eksen konumunu gösterir. Eksen özgüdümlenmemişse, bu değer, eksenin etkinleştirilmiş konumuna göreli konum değerini gösterir. (Teknoloji nesne etiketi: <Axis name>. MotionStatus. Position)
Geçerli hız	" Geçerli hız " alanı gerçek eksen hızını gösterir. (Teknoloji nesne etiketi: <Axis name>. MotionStatus.Velocity)

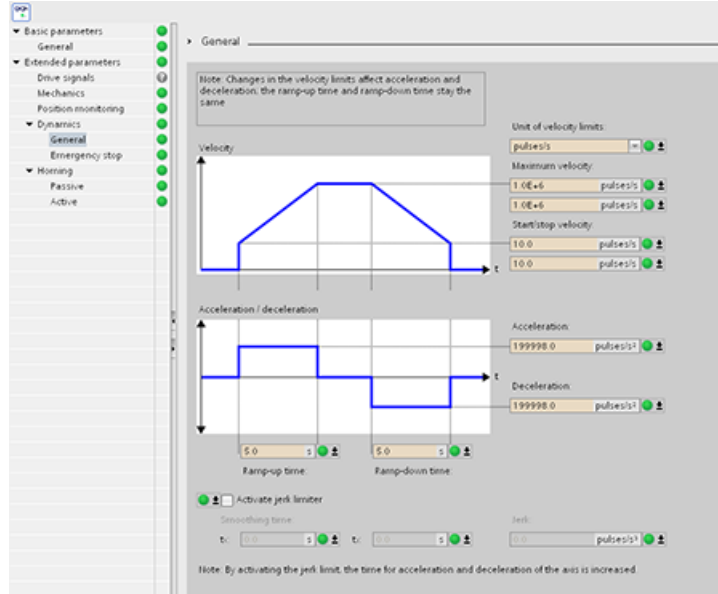
Çizelge 9- 68 Dinamik sınırlar

Dinamik sınır	Açıklama
Hız	" Hız " alanı eksenin yapılandırılmış maksimum hızını gösterir. (Teknoloji nesne etiketi: <Axis name>.Config.DynamicLimits.MaxVelocity)
Hızlanma	" Hızlanma " alanı eksenin hali hazırda yapılandırılmış hızlanmasını gösterir. (Teknoloji nesne etiketi: <Axis name>.Config.DynamicDefaults.Acceleration)
Yavaşlama	" Yavaşlama " alanı eksenin hali hazırda yapılandırılmış yavaşlamasını gösterir. (Teknoloji nesne etiketi: <Axis name>.Config.DynamicDefaults.Deceleration)

## Hareket başlatma değeri kontrolü

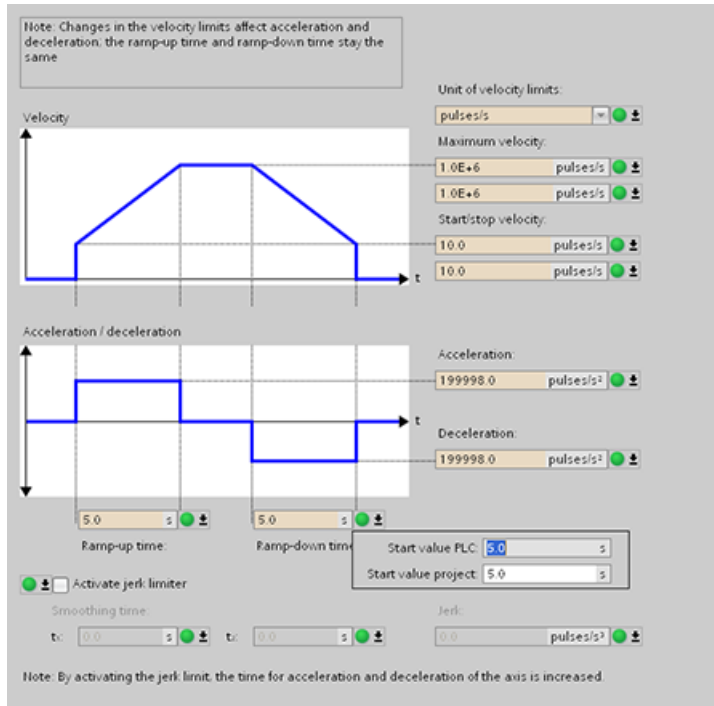
Proses davranışı çevrimiçi modda optimize edilebilecek şekilde hareket yapılandırma parametrelerinin gerçek değerlerini düzenleyebilirsiniz.

Hareketinizin kontrolü için "Teknoloji nesnelere" ni ve "Yapılandırma" nesnesini açınız. Başlatma değeri kontrolüne erişmek için iletişim kutusunun sol üst köşesindeki "gözlük simgesini" tıklayınız:



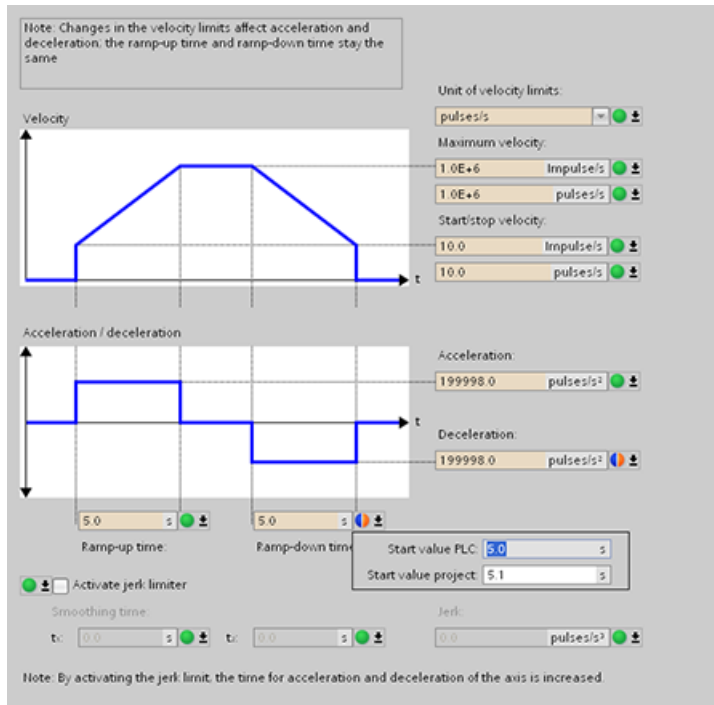
Aşağıdaki şekilde gösterildiği gibi şimdi hareket kontrol yapılandırma parametrelerinin herhangi birinin değerini değiştirebilirsiniz.

Gerçek değeri, her parametrenin PLC (çevrimiçi) başlangıç değeriyle ve proje (çevrimdışı) başlangıç değeriyle karşılaştırabilirsiniz. Bu Teknoloji nesne veri bloğunun (TO-DB) çevrimiçi/çevrimdışı farklılıklarını karşılaştırmak ve PLC'nin sonraki Stop-Start geçişiyle ilgili geçerli değerler olarak kullanılacak olan değerler hakkında bilgi sahibi olmak için gereklidir. Buna ek olarak, bir karşılaştırma simgesi, çevrimiçi/çevrimdışı farklılıkların kolayca tanımlanmasına yardımcı olmak için görsel bir gösterge verir.



Yukarıdaki şekil, çevrimiçi ve çevrimdışı projeler arasında hangi değerlerin farklı olduğunu gösteren karşılaştırma simgeleriyle birlikte hareket parametre ekranını gösterir. Yeşil simge, değerlerin aynı olduğunu gösterir; mavi/turuncu simge değerlerin farklı olduğunu gösterir.

Ayrıca, her bir parametrenin proje (çevrimdışı) başlangıç değeri ve PLC (çevrimiçi) başlangıç değerini gösteren küçük bir pencere açmak için aşağı ok ile parametre butonuna tıklayınız.





## 9.3.8 Aktif komutların izlenmesi

### 9.3.8.1 MC komutların "Done" çıkış parametresiyle izlenmesi

"Done" çıkış parametrelili hareket kontrol komutları, giriş parametresi "Execute" tarafından başlatılır ve tanımlanmış bir sonuca sahip olur. (örneğin "MC\_Home" hareket kontrol komutu ile Homing başarılı oldu) görev tamamlandı ve eksen hareketsizdir.

- Görev başarıyla tamamlanmış ise "Done" çıkış parametresi, TRUE değeri gösterir.
- Görevin hala işleniyor olduğu çıkış parametreleri "Busy", "CommandAborted" ve "Error" sinyali iptal edildi veya bir hata bekliyor. Hareket kontrol komutu "MC\_Reset" iptal edilemez ve böylece "CommandAborted" çıkış parametresine sahip olamaz.
  - Hareket kontrol görevinin işlenmesi sırasında, çıkış parametresi "Busy" TRUE değeri gösterir. Görev tamamlanmış, iptal edilmiş ya da bir hata ile durdurulmuş ise, çıkış parametresi "Busy" FALSE değerine değişir. Bu değişiklik "Execute" giriş parametresinde sinyale bakılmaksızın oluşur.
  - Çıkış parametreleri "Done", "CommandAborted" ve "Error" en az bir çevrim için TRUE değeri gösterir. Giriş parametresi "Execute" TRUE olarak ayarlanırken bu durum mesajları ortaya çıkar.

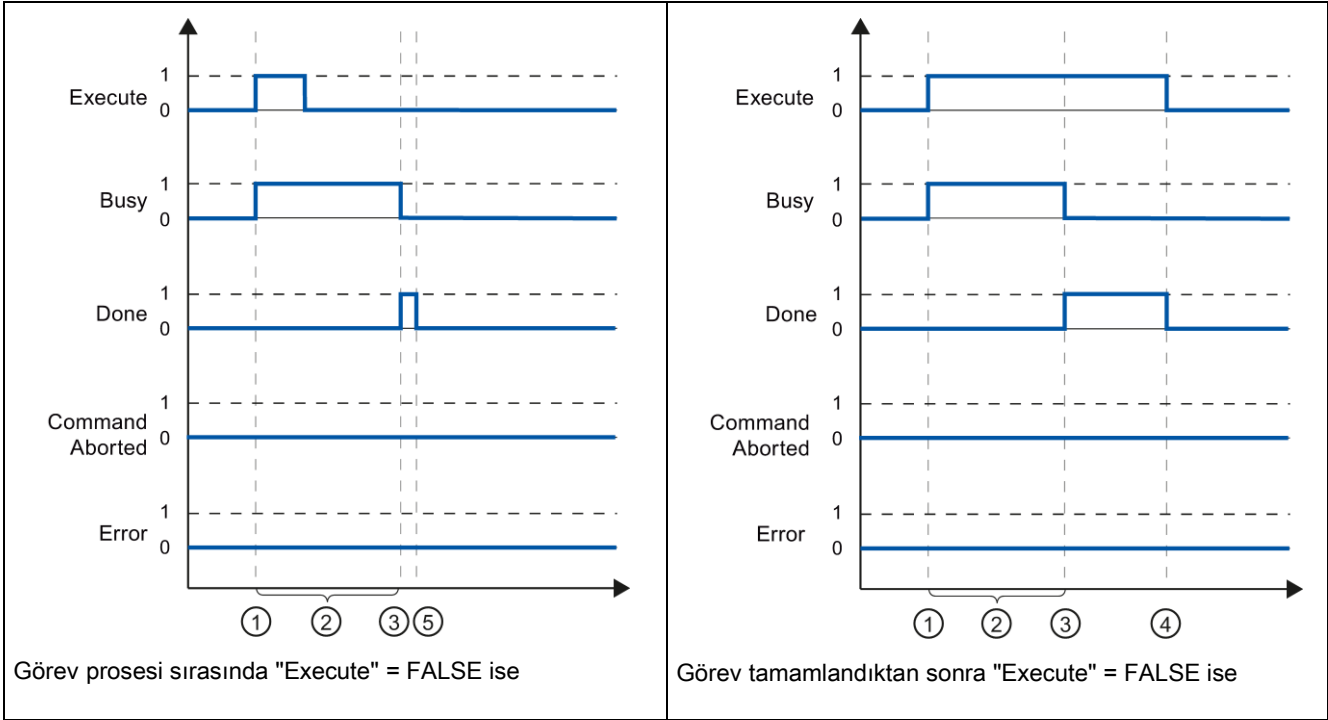
Aşağıdaki hareket kontrol komutlarının görevleri tanımlanmış bir sonuca sahiptir:

- MC\_Reset
- MC\_Home
- MC\_Halt
- MC\_MoveAbsolute
- MC\_MoveRelative

Durum bit davranışı çeşitli örnek durumlar için aşağıda sunulmuştur.

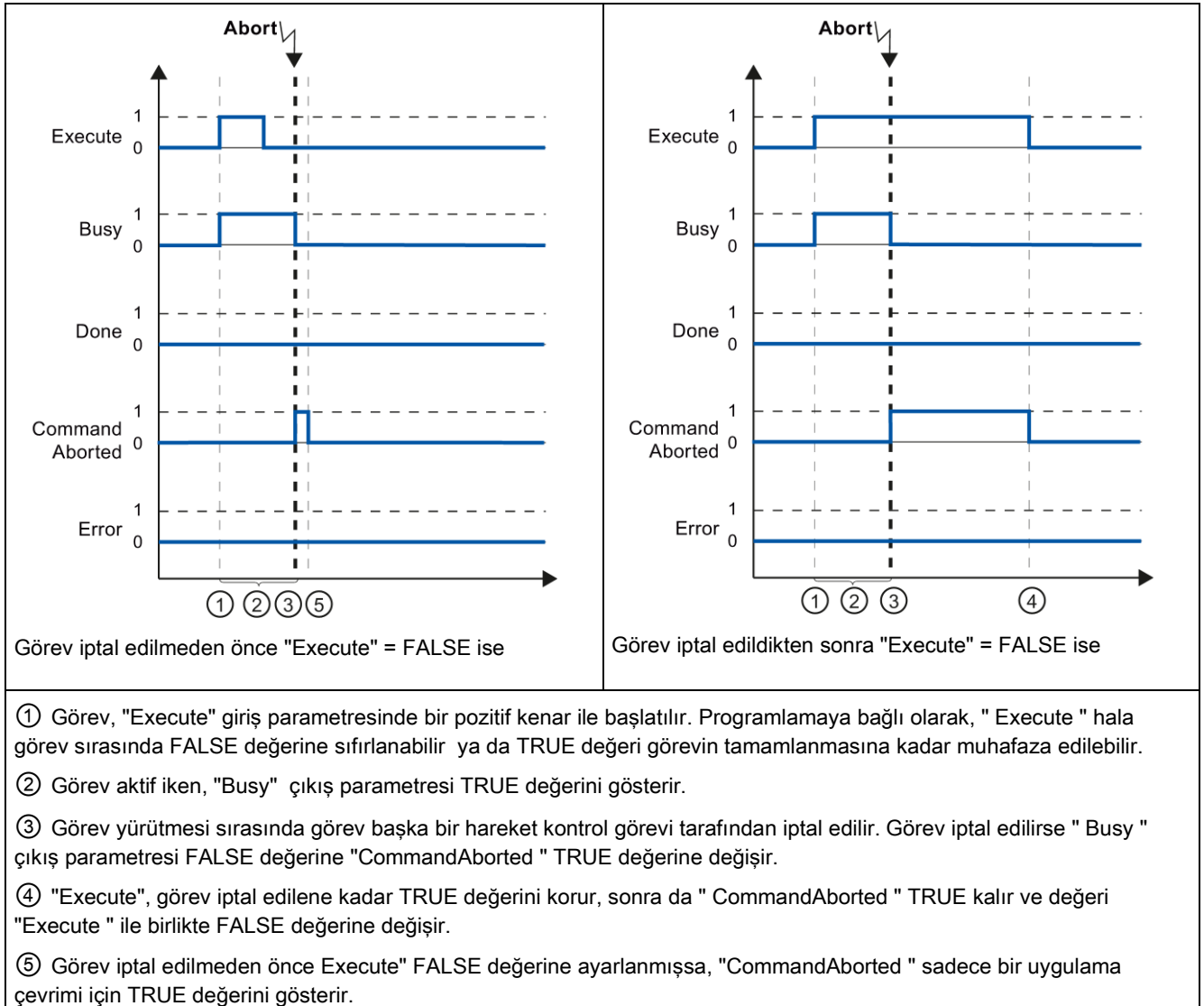
- İlk örnek, tamamlanmış bir görev için eksen davranışını gösterir. Hareket kontrol görevi sonucunun zamanıyla birlikte tamamen yürütülür ise, bu "Done" çıkış parametresinde TRUE değeri ile belirtilir. "Execute" Giriş parametresinin sinyal durumu " Done " çıkış parametresinde görüntüleme süresini etkiler.
- İkinci örnek ise, iptal edilen bir görev için eksen davranışını gösterir. Hareket kontrol görevi yürütme sırasında iptal edilirse, bu "CommandAborted çıkış parametresinde " TRUE değeriyle belirtilir. "Execute" Giriş parametresi sinyal durumu, "CommandAborted" çıkış parametresinde ekran süresini etkiler.
- Üçüncü örnek bir hata oluşması durumunda ekseninin davranışını gösterir. Hareket kontrol görevinin yürütülmesi sırasında bir hata ortaya çıkarsa, bu "Error" çıkış parametresinde TRUE değeri ile belirtilir. "Execute" Giriş parametresinin sinyal durumu "Error" çıkış parametresinin görüntüleme süresini etkiler.

Çizelge 9- 69 Örnek 1 – Görevin tam yürütülmesi

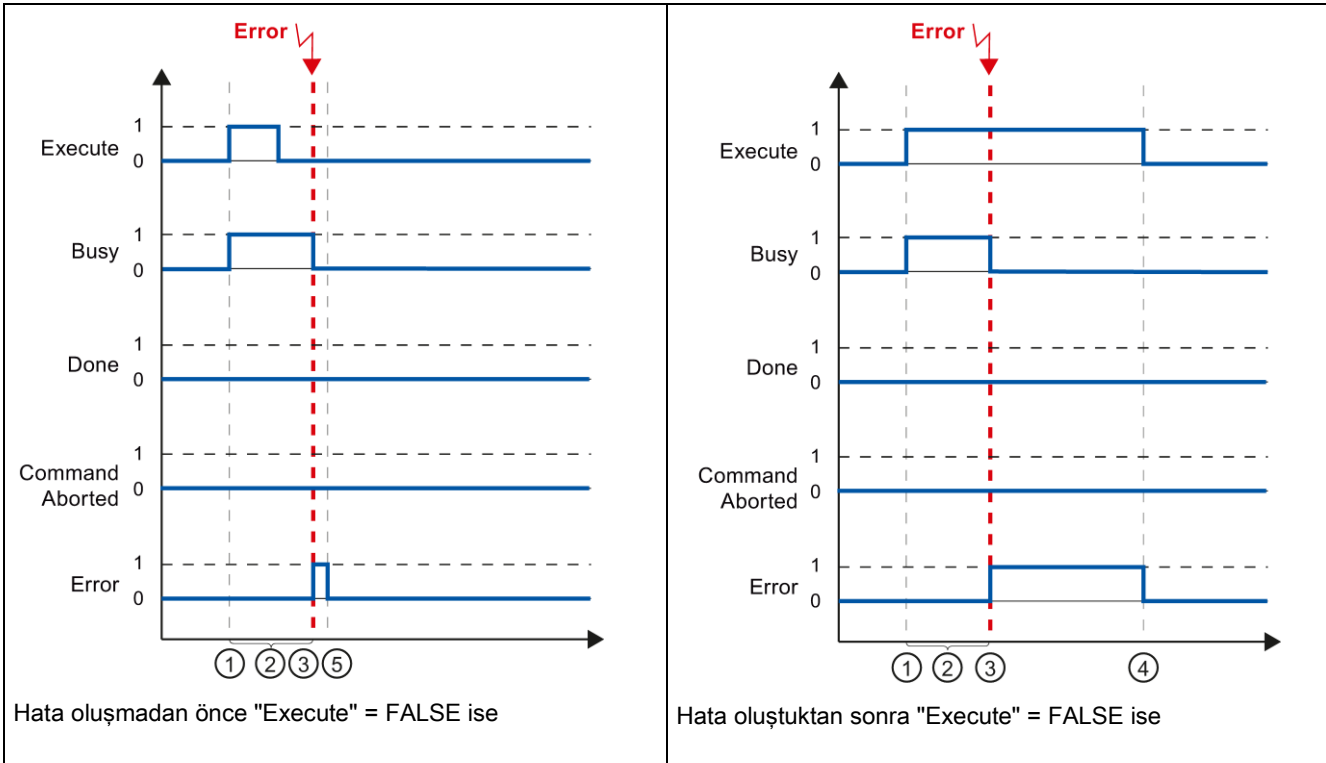


- ① Görev, "Execute" giriş parametresinde bir pozitif kenar ile başlatılır. Programlamaya bağlı olarak, "Execute" hala görev sırasında FALSE değerine sıfırlanabilir ya da TRUE değeri görevin tamamlanmasına kadar muhafaza edilebilir.
- ② Görev aktif iken, "Busy" çıkış parametresi TRUE değerini gösterir.
- ③ Görevin sonuçlanmasıyla beraber (örneğin "MC\_Home": hareket kontrol komutu için homing başarılı oldu) "Busy" çıkış parametresi FALSE değerine "Done" TRUE değerine değişir.
- ④ "Execute", görev tamamlanana kadar TRUE değerini korur, sonra da "Done" TRUE kalır ve değeri "Execute" ile birlikte FALSE değerine değişir.
- ⑤ Görev tamamlanmadan önce "Execute" FALSE değerine ayarlanmışsa, "Done" sadece bir uygulama çevrimi için TRUE değerini gösterir.

Çizelge 9- 70 Örnek 2 – Görevi iptal etme



Çizelge 9- 71 Örnek 3 – Görevin yürütülmesi sırasında hata



- ① Görev, "Execute" giriş parametresinde bir pozitif kenar ile başlatılır. Programlamaya bağlı olarak, "Execute" hala görev sırasında FALSE değerine sıfırlanabilir ya da TRUE değeri görevin tamamlanmasına kadar muhafaza edilebilir.
- ② Görev aktif iken, "Busy" çıkış parametresi TRUE değerini gösterir.
- ③ Görev yürütmesi sırasında bir hata oluştu. Hata oluştuğunda "Busy" çıkış parametresi FALSE değerine "Error" TRUE değerine değişir.
- ④ "Execute", hata oluşana kadar TRUE değerini korur, sonra da "Error" TRUE kalır ve değeri "Execute" ile birlikte FALSE değerine değişir.
- ⑤ Hata oluşmadan önce "Execute" FALSE değerine ayarlanmışsa, "Error" sadece bir uygulama çevrimi için TRUE değerini gösterir.

### 9.3.8.2 MC\_Velocity komutunun izlenmesi

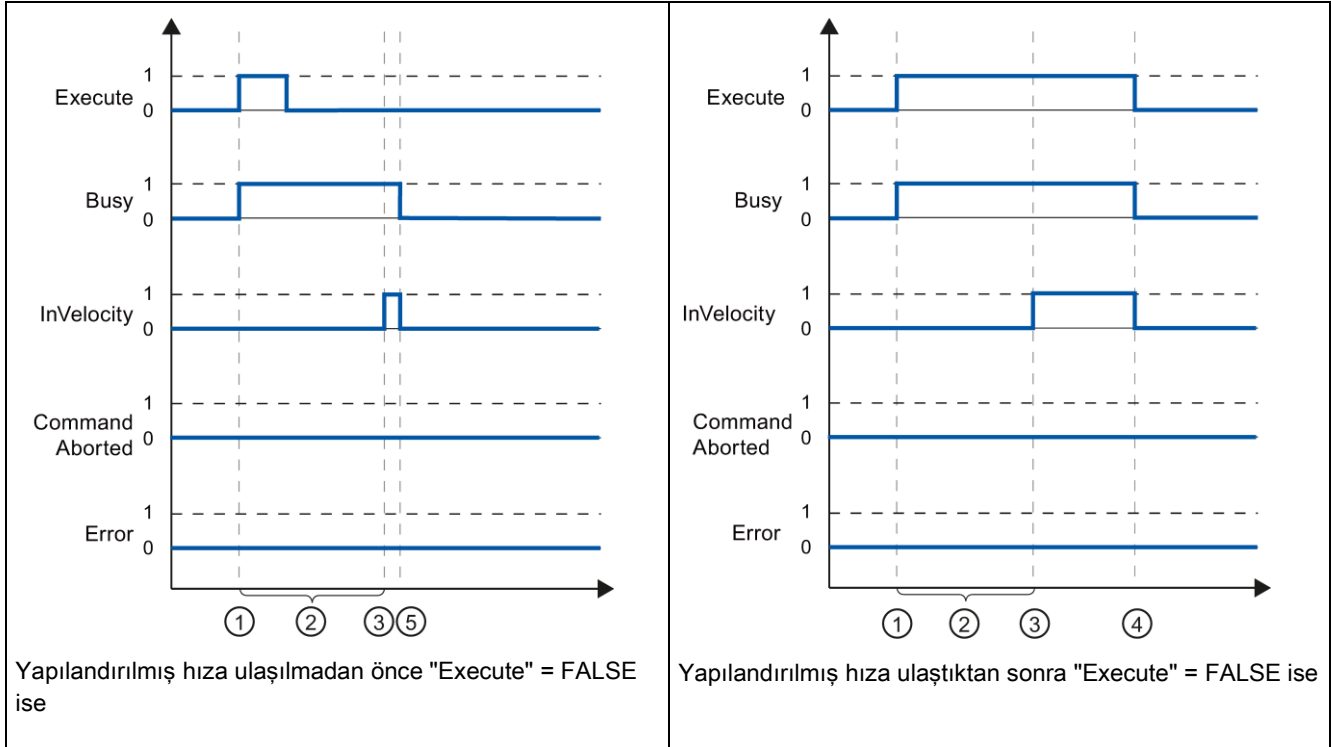
Hareket kontrol komutu "MC\_MoveVelocity"nin görevleri, belirtilen hızda bir hareket uygulamaktır:

- Hareket kontrol komutu "MC\_MoveVelocity" görevleri için tanımlanmış bir son yoktur. Görev amacı, parametrelendirilmiş hıza ilk kez ulaşıldığında yerine getirilmiş olur ve eksen sabit bir hızla hareket eder. Parametrelendirilmiş hıza ulaşıldığında, bu, "InVelocity" çıkış parametresinde TRUE değeriyle belirtilir.
- Parametrelendirilmiş hıza ulaşıldığında ve "Execute" giriş parametresi değeri False olarak ayarlandığında görev tamamlanır. Ancak, eksen hareketi, görevin tamamlanmasıyla henüz tamamlanmamıştır. Örneğin, eksen hareketi, hareket kontrol görevi "MC\_Halt" ile durdurulabilir.
- "Busy", "CommandAborted" ve "Error" çıkış parametreleri, görevin hala işleniyor olduğunu, iptal edildiğini veya bir hata beklediğini sinyaller.
  - Hareket kontrol görevinin işlenmesi sırasında, çıkış parametresi "Busy" TRUE değeri gösterir. Görev tamamlanmış, iptal edilmiş ya da bir hata ile durdurulmuş ise, çıkış parametresi "Busy" FALSE değerine değişir. Bu değişiklik "Execute" giriş parametresinde sinyale bakılmaksızın oluşur.
  - Şartları karşılandığında çıkış parametreleri "InVelocity", "CommandAborted" ve "Error" en az bir çevrim için TRUE değeri gösterir. Giriş parametresi "Execute" TRUE olarak ayarlanırken bu durum mesajları ortaya çıkar.

Durum bit davranışı çeşitli örnek durumlar için aşağıda sunulmuştur:

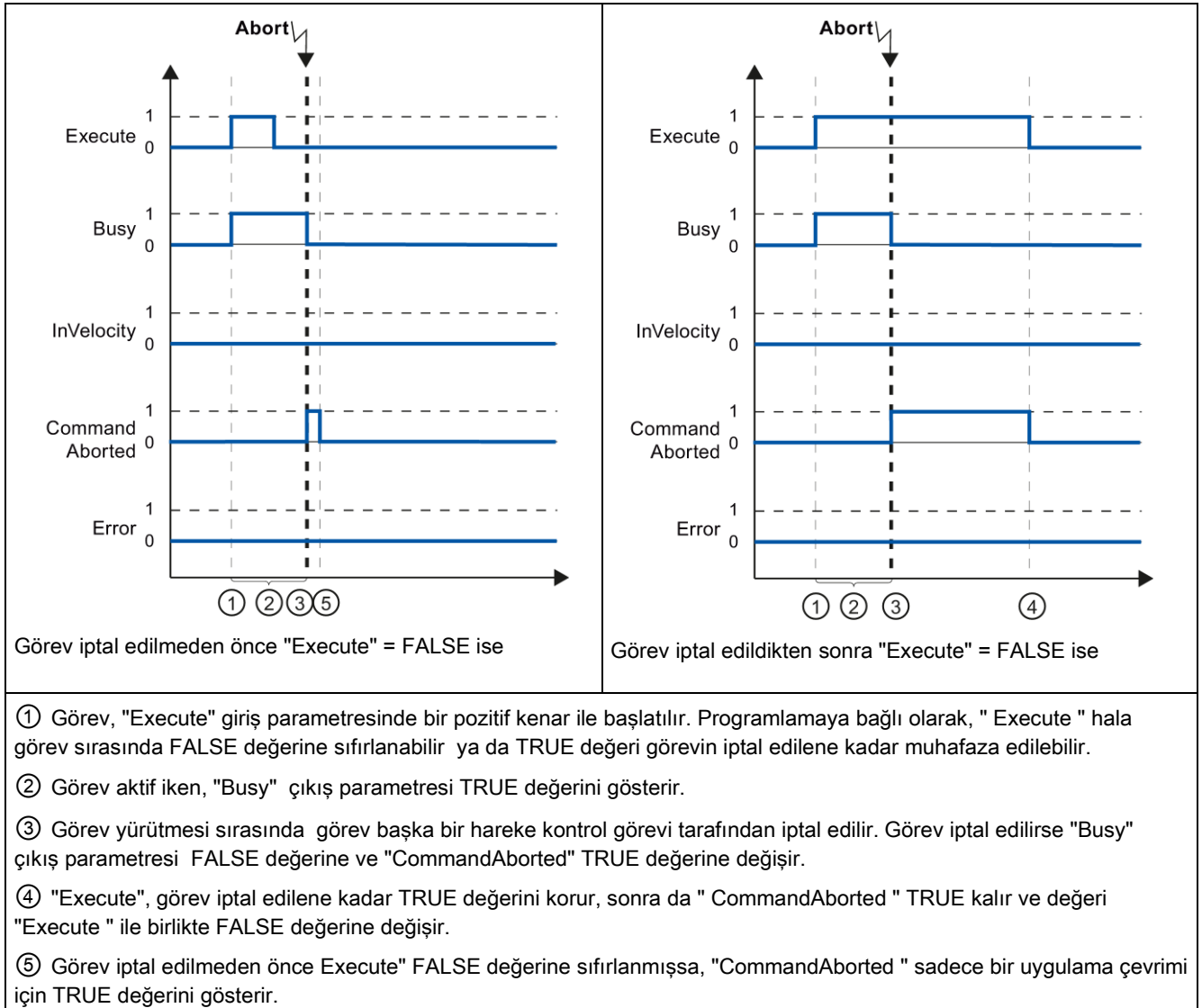
- İlk örnek, eksenin parametrelendirilmiş hıza ulaştığındaki davranışını gösterir. Hareket kontrol görevi parametrelendirilmiş hıza ulaşıldığında yürütülmüş ise, bu "InVelocity" çıkış parametresinde TRUE değeri ile belirtilir. "Execute" Giriş parametresinin sinyal durumu "InVelocity" çıkış parametresinde görüntüleme süresini etkiler.
- İkinci örnek ise, görev iptal edilirse parametrelendirilmiş hıza ulaşılmadan önceki davranışı gösterir. Hareket kontrol görevi parametrelendirilmiş hıza ulaşılmadan önce iptal edilirse, bu "CommandAborted" çıkış parametresinde TRUE değeriyle belirtilir. "Execute" Giriş parametresi sinyal durumu, "CommandAborted" çıkış parametresinde ekran süresini etkiler.
- Üçüncü örnek bir hata oluşması durumunda parametrelendirilmiş hıza ulaşılmadan önceki eksen davranışını gösterir. Hareket kontrol görevinin yürütülmesi sırasında parametrelendirilmiş hıza ulaşılmadan önce bir hata ortaya çıkarsa, bu, "Error" çıkış parametresinde TRUE değeri ile belirtilir. "Execute" giriş parametresinin sinyal durumu "Error" çıkış parametresinin görüntüleme süresini etkiler.

Çizelge 9- 72 Örnek 1 – Parametrelendirilen hızı ulaşırsa



- ① Görev, "Execute" giriş parametresinde bir pozitif kenar ile başlatılır. Programlamaya bağlı olarak, "Execute" hala görev sırasında parametrelendirilmiş hızı ulaşılmadan önce ya da alternatif olarak yalnızca bu hızı ulaşıldıktan sonra FALSE değerine sıfırlanabilir.
- ② Görev aktif iken, "Busy" çıkış parametresi TRUE değerini gösterir.
- ③ Parametrelendirilmiş hızı ulaşıldığında "InVelocity" çıkış parametresi TRUE değerine değişir.
- ④ "Execute", parametrelendirilmiş hızı ulaşıldıktan sonra bile TRUE değerini korur, görev aktif kalır. "InVelocity" ve "Busy" TRUE kalır ve durumu yalnızca "Execute" ile birlikte FALSE değerine değişir.
- ⑤ Parametrelendirilmiş hızı ulaşılmadan önce "Execute" FALSE değerine ayarlanmışsa, Parametrelendirilmiş hızı ulaşıldığında görev tamamlanmıştır. "InVelocity" sadece bir uygulama çevrimi için TRUE değerini gösterir ve "Busy" ile birlikte FALSE değerine değişir.

Çizelge 9- 73 Örnek 2 - Görev parametrelendirilmiş hıza ulaşılmadan önce iptal edilirse

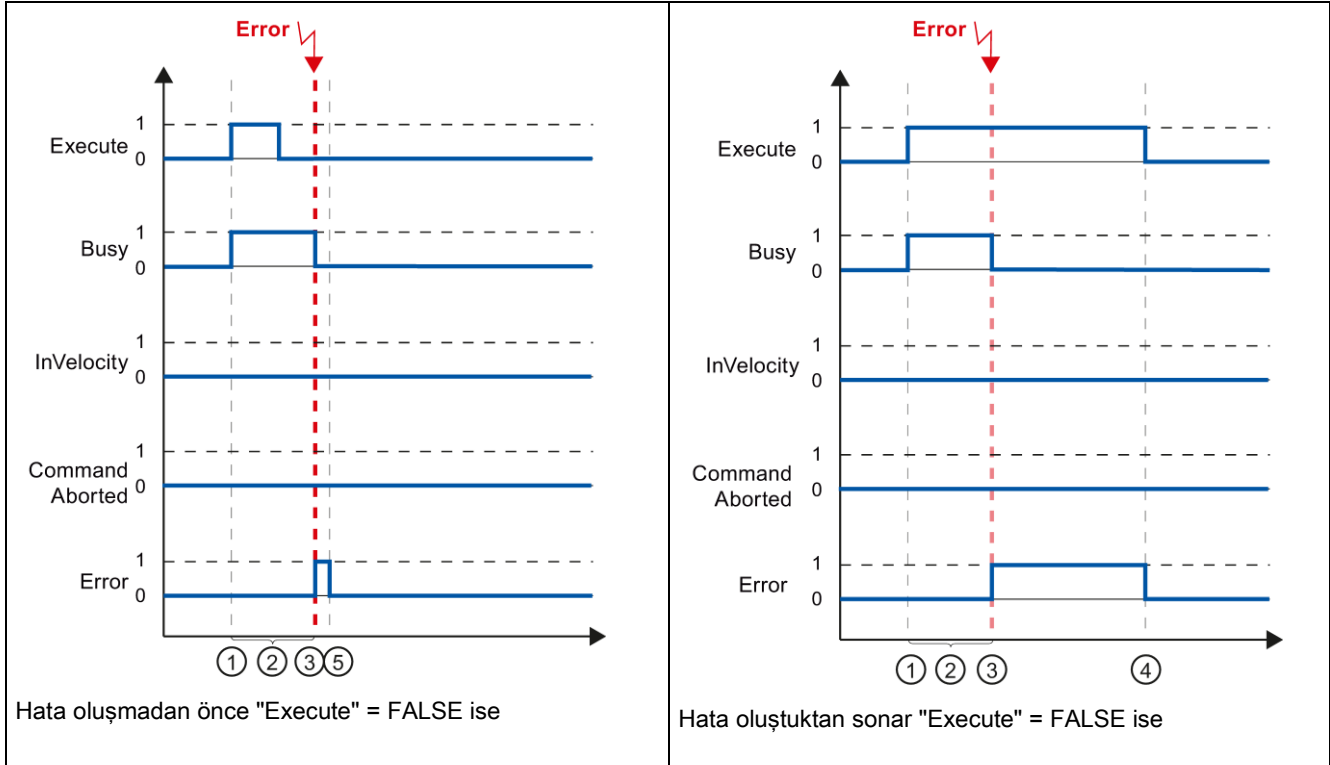


### Not

Aşağıdaki koşullar altında bir iptal, "CommandAborted" çıkış parametresinde belirtilmez:

- Parametrelendirilmiş hıza ulaşıldı, "Execute" giriş parametresi FALSE değerine sahiptir ve yeni bir hareket kontrol görevi başlatılır.
- Parametrelendirilmiş hıza ulaşıldığında ve "Execute" giriş parametresi FALSE değerine sahip olduğunda görev tamamlanır. Bu nedenle, yeni görevin başlangıcı iptal olarak gösterilmez.

Çizelge 9- 74 Örnek 3 - Parametrelendirilmiş hıza ulaşılmadan önce bir hata oluşursa



- ① Görev, "Execute" giriş parametresinde bir pozitif kenar ile başlatılır. Programlamaya bağlı olarak, "Execute" hala görev sırasında FALSE değerine sıfırlanabilir ya da TRUE değeri hatanın oluşmasına kadar muhafaza edilebilir.
- ② Görev aktif iken, "Busy" çıkış parametresi TRUE değerini gösterir.
- ③ Görev yürütmesi sırasında bir hata oluştu. Hata oluştuğunda "Busy" çıkış parametresi FALSE değerine "Error" TRUE değerine değişir.
- ④ "Execute", hata oluşana kadar TRUE değerini korur, sonra da "Error" TRUE kalır ve yalnızca durumu "Execute" ile birlikte FALSE değerine değişir.
- ⑤ Hata oluşmadan önce "Execute" FALSE değerine ayarlanmışsa, "Error" sadece bir uygulama çevrimi için TRUE değerini gösterir.

#### Not

Aşağıdaki koşullar altında, bir hata, "Error" çıkış parametresinde belirtilmez:

- Parametrelendirilmiş hıza ulaşıldı, "Execute" giriş parametresi FALSE değerine sahiptir ve eksen hatası oluşur (örneğin yazılım sınır anahtarına yaklaşıldı).
- Parametrelendirilmiş hıza ulaşıldığında ve "Execute" giriş parametresi FALSE değerine sahip olduğunda görev tamamlanır. Görevin tamamlanmasından sonra eksen hatası yalnızca "MC\_Power" hareket kontrol komutunda belirtilir.



### 9.3.8.3 MC\_MoveJog komutunun izlenmesi

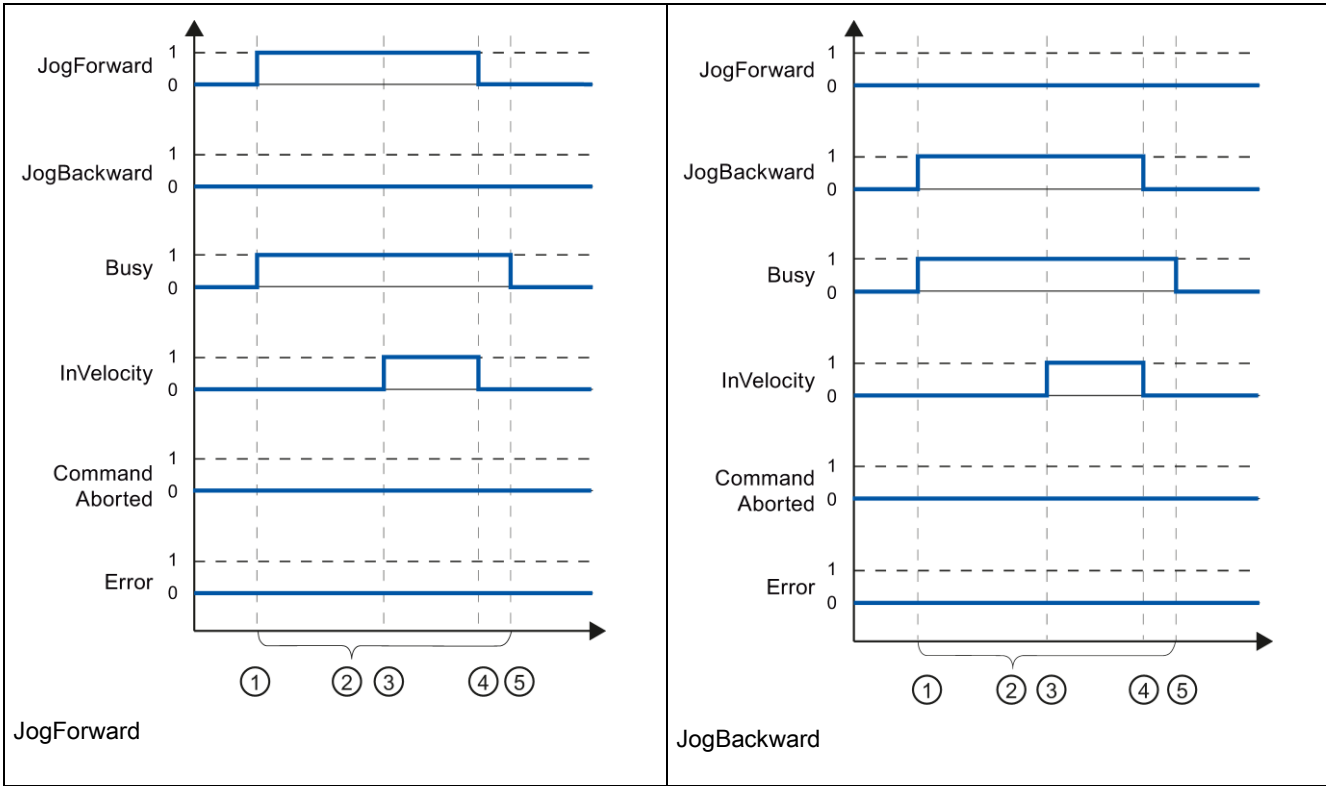
Hareket kontrol komutu "MC\_MoveJog" görevleri, bir adım işlemi uygular:

- Hareket kontrol komutu "MC\_MoveJog" görevleri için tanımlanmış bir son yoktur. Görev amacı, parametrelendirilmiş hıza ilk kez ulaşıldığında yerine getirilmiş olur ve eksen sabit bir hızla hareket eder. Parametrelendirilmiş hıza ulaşıldığında, bu, "InVelocity" çıkış parametresinde TRUE değeriyle belirtilir.
- Giriş parametresi "JogForward" veya "JogBackward" FALSE değerine ayarlandığında ve eksen durma noktasına geldiği zaman komut tamamlanmıştır.
- "Busy", "CommandAborted" ve "Error" çıkış parametreleri, görevin hala işleniyor olduğunu, iptal edildiğini veya bir hata beklediğini sinyaller.
  - Hareket kontrol görevinin işlenmesi sırasında, çıkış parametresi "Busy" TRUE değeri gösterir. Görev tamamlanmış, iptal edilmiş ya da bir hata ile durdurulmuş ise, çıkış parametresi "Busy" FALSE değerine değişir.
  - Eksen parametrelendirilmiş hızda hareket ettiği sürece çıkış parametresi "InVelocity" TRUE durumunu gösterir. Çıkış parametreleri "InVelocity", "CommandAborted" ve "Error" en az bir çevrim için TRUE değeri gösterir. Giriş parametresi "JogForward" veya "JogBackward" TRUE olarak ayarlandığı sürece bu durum mesajları kilitlenir.

Durum bit davranışı çeşitli örnek durumlar için aşağıda sunulmuştur:

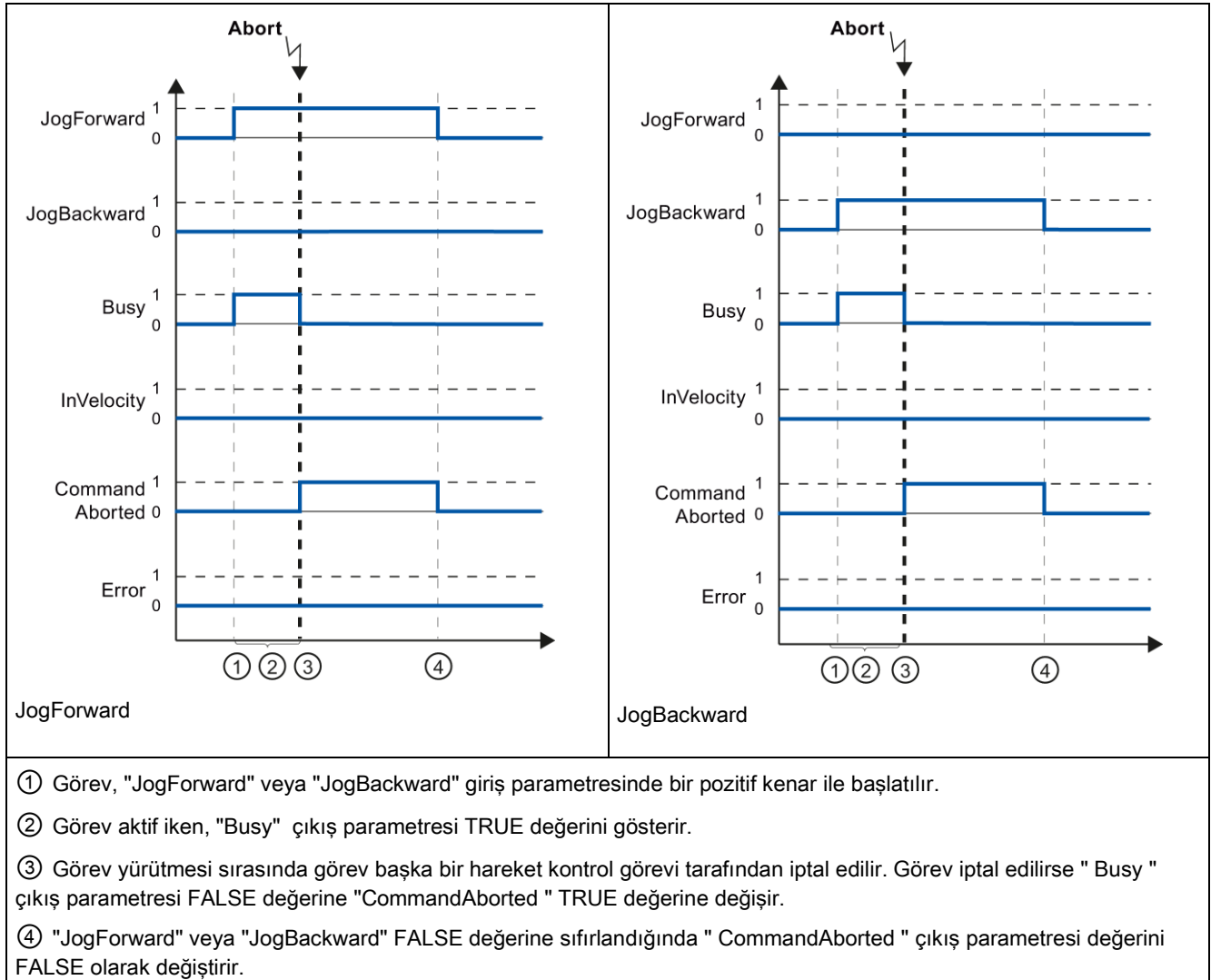
- İlk örnek, eksenin parametrelendirilmiş hıza ulaşıldığında ve bunu sürdürdüğündeki davranışını gösterir. Hareket kontrol görevi parametrelendirilmiş hıza ulaşıldığında yürütülmüş ise, bu "InVelocity" çıkış parametresinde TRUE değeri ile belirtilir.
- İkinci örnek ise, görev iptal edilirse eksenin parametrelendirilmiş hıza ulaşılmadan önceki davranışını gösterir. Bu durum "CommandAborted" çıkış parametresinde TRUE değeriyle belirtilir. Davranış, parametrelendirilmiş hıza ulaşıp ulaşılmamadan bağımsızdır.
- Üçüncü örnek bir hata oluşması durumunda eksen davranışını gösterir. Hareket kontrol görevinin yürütülmesi sırasında bir hata ortaya çıkarsa, bu, "Error" çıkış parametresinde TRUE değeri ile belirtilir. Davranış, parametrelendirilmiş hıza ulaşıp ulaşılmamadan bağımsızdır.

Çizelge 9- 75 Örnek 1 - Parametrelendirilmiş hıza ulaşıp sürdürülüyorsa



- ① Görev, "JogForward" veya "JogBackward" giriş parametresinde bir pozitif kenar ile başlatılır.
- ② Görev aktif iken, "Busy" çıkış parametresi TRUE değerini gösterir.
- ③ Parametrelendirilmiş hıza ulaşıldığında "InVelocity" çıkış parametresi FALSE değerine değişir.
- ④ "JogForward" veya "JogBackward" giriş parametreleri FALSE değerine sıfırlandığında eksen hareketi durur. Eksen yavaşlamaya başlar. Sonuç olarak eksen artık sabit hızda hareket etmez ve "InVelocity" çıkış parametresi durumunu FALSE olarak değiştirir.
- ⑤ Eksen hareketsiz duruma gelirse hareket kontrol komutu tamamlanır ve "Busy" çıkış parametresi "Error" değerini FALSE olarak değiştirir.

Çizelge 9- 76 Örnek 2 – Yürütme sırasında görev iptal edilirse

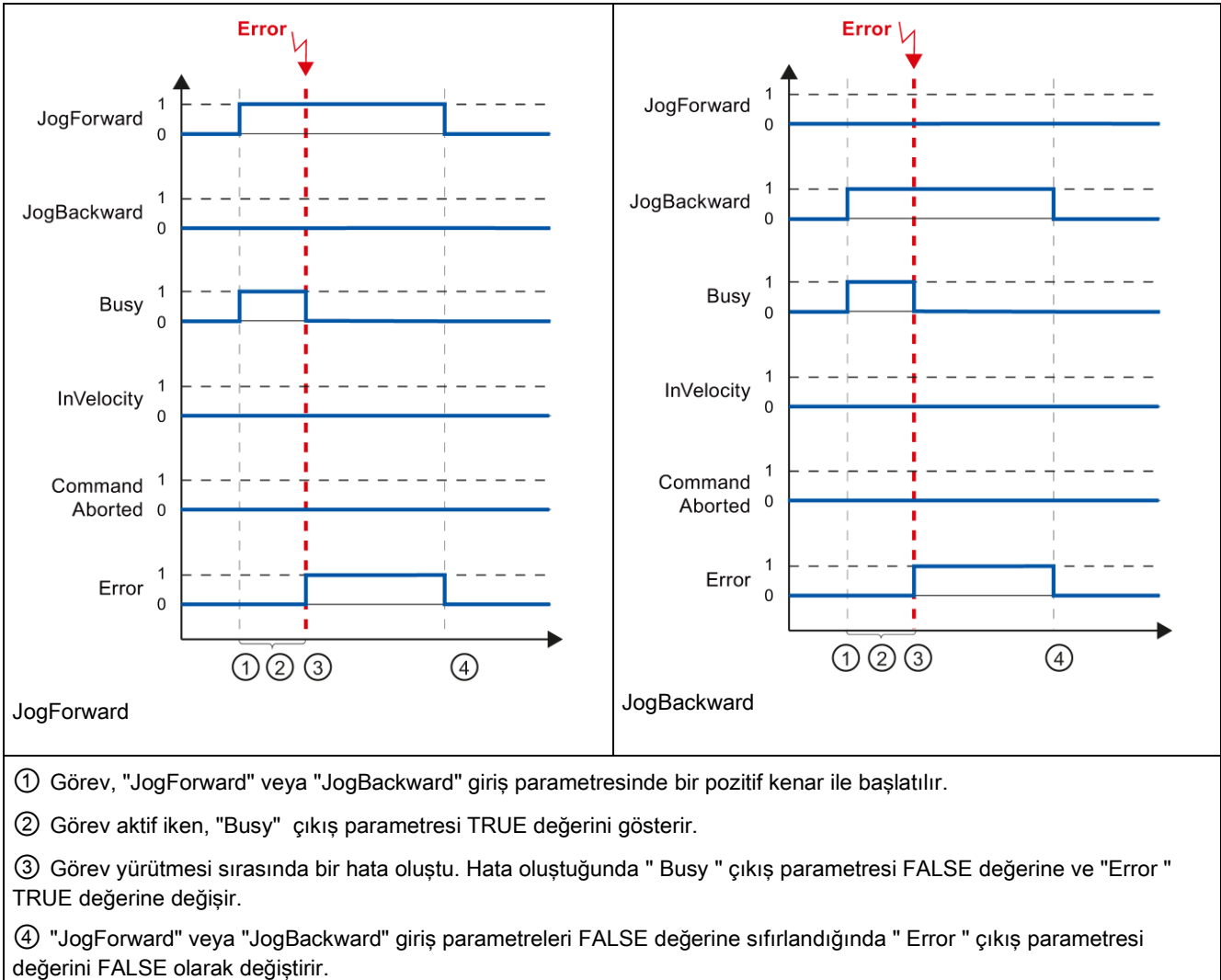


### Not

Aşağıdaki tüm koşullar yerine getirildiği takdirde görev iptali, sadece bir yürütme çevrimi için "CommandAborted" çıkış parametresinde gösterilir:

Giriş parametreleri "JogForward" ve "JogBackward" FALSE değerine sahiptir (ama eksen hala yavaşlıyor) ve bir yeni hareket kontrol görevi başlatılır.

Çizelge 9- 77 Örnek 3 – Görev yürütmesi sırasında bir hata oluşmuşsa




### Not

Aşağıdaki tüm koşullar yerine getirildiği takdirde hata olayı, sadece bir yürütme çevrimi için "Error" çıkış parametresinde gösterilir:

Giriş parametreleri "JogForward" ve "JogBackward" FALSE değerine sahiptir (ama eksen hala yavaşlıyor) ve bir yeni hata oluşur (örneğin yazılım sınır anahtarına yaklaşıldı.)

## Haberleşme

S7-1200, CPU'lar ve programlama cihazları, HMI ve diğer CPU'lar arasında çeşitli haberleşme tipleri sunar.

 <b>UYARI</b>
<p><b>Bir saldırgan ağlarınıza fiziksel olarak erişebilirse, saldırgan muhtemelen verileri okuyup yazabilir.</b></p> <p>(GET / PUT kullanarak Hm'ler hariç) TIA Portal, CPU ve HMI'ler tekrar oynatmaya ve "man-in-the-middle" (iki bağlantı noktası arasındaki bağlantıyı izinsiz izleme) saldırılarına karşı koruyan güvenli haberleşme kullanır. Haberleşme etkinleştirildiğinde, imzalı mesajların değişimi açık metinde yer alır, bu da bir saldırganın veri okumasına izin verir, ancak verilerin yetkisiz yazılmasına karşı korur. Haberleşme süreci değil TIA Portalı know-how korunan blok verilerini şifreler.</p> <p>Diğer tüm haberleşme biçimlerinin (PROFIBUS, PROFINET, AS-i ya da diğer I/O bus, GET / PUT, T-Blok ve haberleşme modülleri (CM) üzerinden I/O değişimi) hiçbir güvenlik özelliği yoktur. Fiziksel erişimi kısıtlayarak bu haberleşme biçimlerini korumak zorundasınız. Bir saldırgan bu haberleşme biçimlerini kullanarak fiziksel olarak ağlarınıza erişebilirse, saldırgan muhtemelen veri okuyup yazabilir.</p> <p>Güvenlik bilgisi ve öneriler için, Siemens Servis ve destek sitesindeki (<a href="http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf">http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf</a>) "Sanayi Güvenliği ile ilgili operasyonel Kurallar" a bakılmalıdır.</p>

## PROFINET

PROFINET, Ethernet üzerindeki diğer haberleşme ortakları ile kullanıcı programı vasıtasıyla veri alışverişi için kullanılır:

- PROFINET, maksimum 256 alt modüle sahip 16 IO cihazını destekler. PROFIBUS, DP-Master başına maksimum 512 modül ile DP master başına 32 slave destekleyen, 3 bağımsız PROFIBUS DP Master'a izin verir.
- S7 haberleşmesi
- Kullanıcı Veri Paketi Protokolü (UDP)
- ISO on TCP (RFC 1006)
- İletim Kontrol Protokolü (TCP)

## PROFINET RT IO kontrolör

PROFINET RT kullanan bir IO kontrolör olarak CPU yerel PN ağda ya da bir PN / PN Kuplör (link) üzerinden 16 PN cihazla iletişim kurar. Daha fazla bilgi için Uluslararası PROFIBUS ve PROFINET, PI ([www.us.profinet.com](http://www.us.profinet.com)) bakınız.

## PROFIBUS

PROFIBUS, PROFIBUS ağı üzerinden diğer haberleşme ortakları ile kullanıcı programı vasıtasıyla veri alışverişi için kullanılır:

- CM 1242-5 ile CPU bir PROFIBUS DP slave olarak çalışır.
- CM 1243-5 ile CPU bir PROFIBUS DP master olarak çalışır.
- PROFIBUS DP Slave'leri, PROFIBUS DP Master'ları, AS-i (3 sol-kenar haberleşme modülleri) ve PROFINET birbirini sınırlamayan ayrı haberleşme ağlarıdır.

## AS-i

S7-1200 CM 1243-2 AS-i Master, bir AS-i ağın bir S7-1200 CPU'ya eklenmesine izin verir.

## CPU-ile-CPU S7 haberleşmesi

Bir partner istasyon ile bir haberleşme bağlantısı oluşturabilir ve S7 CPU ile iletişim kurmak için GET ve PUT komutlarını kullanabilirsiniz.

## Teleservis haberleşmesi

GPRS aracılığıyla teleservis içinde STEP 7' nin tesis edildiği mühendislik istasyonu, GSM şebekesi ve internet üzerinden CP 1242-7'ye sahip bir SIMATIC S7-1200 istasyonu ile iletişim kurar. Bağlantı, bir arabulucu olarak hizmet veren ve internete bağlı olan bir telekontrol sunucusu üzerinden çalışır.

## IO-Link

S7-1200 SM 1278 4xIO-Link Master, IO-Link cihazlarının bir S7 1200 CPU'ya bağlanmasına imkan verir.

## 10.1 Desteklenen asenkron haberleşme bağlantılarının sayısı

PROFINET ve PROFIBUS için CPU aşağıdaki maksimum sayıda eş zamanlı, asenkron haberleşme bağlantılarını destekler:

- Açık Kullanıcı Haberleşmeleri için 8 bağlantı (aktif veya pasif): TSEND\_C, TRCV\_C, TCON, TDISCON, TSEND ve TRCV.
- 3 CPU-ile-CPU S7 bağlantıları, GET/PUT veri sunucusu için
- 8 CPU-ile-CPU S7 bağlantıları, GET/PUT veri istemcisi için

---

### Not

CPU-ile-CPU S7 haberleşmesi için S7-1200, S7-300 ve S7-400 CPU'lar GET ve PUT komutlarını kullanır. CPU-ile-CPU S7 haberleşmesi için bir S7-200 CPU ETHx\_XFER komutlarını kullanır.

- 
- HMI bağlantıları: TIA Portal aracılığıyla uygulamanız için gerekli HMI bağlantı sayısını yapılandırabilirsiniz.
  - PG bağlantıları: CPU, 1 programlama cihazı (PG) desteklemek için bağlantılar sağlar.
  - Web sunucusu (HTTP) bağlantıları: CPU, Web sunucusu için bağlantı sağlar.

---

### Not

Açık Kullanıcı Haberleşmeleri, S7 bağlantısı, HMI, programlama cihazı ve Web sunucusu (HTTP) haberleşme bağlantıları hali hazırda kullanılan özelliklere dayalı çoklu bağlantı kaynaklarını kullanabilir.

---

## 10.2 PROFINET

### 10.2.1 Yerel/Partner bağlantı

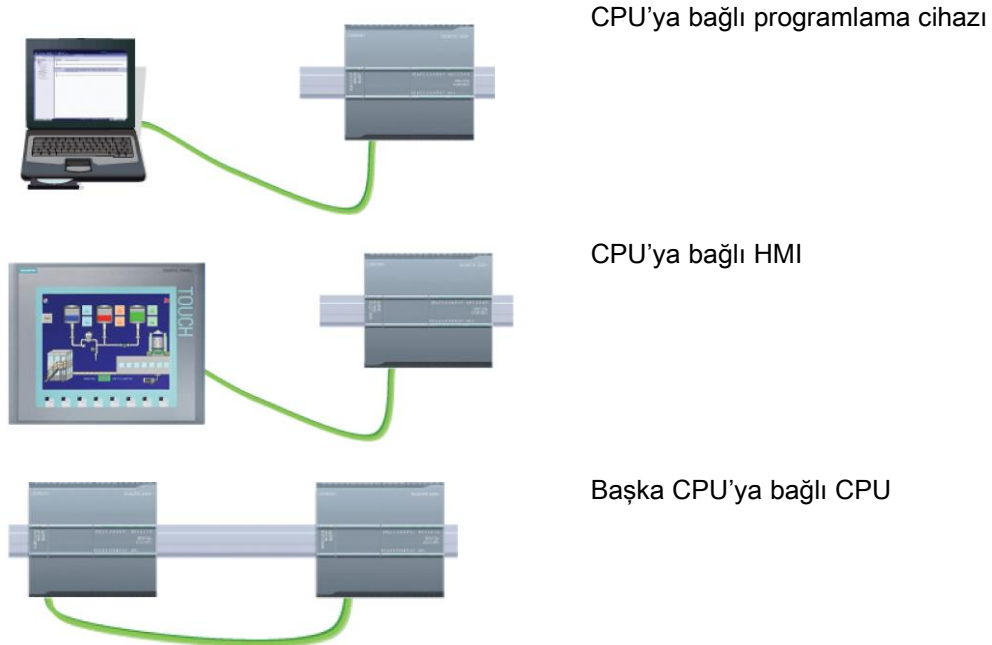
Bir Yerel / Partner (uzak) bağlantı, haberleşme hizmetlerini oluşturmak için iki haberleşme partnerinin mantıksal bir atamasını tanımlar. Bir bağlantı aşağıdakileri tanımlar:

- Katılan haberleşme partnerleri ( Bir aktif, bir pasif )
- Bağlantı tipi (Örneğin, bir PLC , HMI veya cihaz bağlantısı)
- Bağlantı yolu

Haberleşme partnerleri, haberleşme bağlantısını kuracak ve oluşturacak komutları yürütür. Aktif ve pasif haberleşme uç noktası partnerlerini belirlemek için parametreleri kullanınız. Bağlantı kurduktan ve oluşturduktan sonra, bağlantı otomatik olarak sürdürülür ve CPU tarafından izlenir. Bağlantıyla ilgili parametreleri yapılandırma hakkında bilgi için (Sayfa 150 ) "Yerel / Partner bağlantısının yapılandırılması" bölümüne bakınız.

Bağlantının sonlandırılması halinde ( örneğin, hat kopması nedeniyle), aktif partner yapılandırılmış bağlantıyı yeniden oluşturmaya çalışır. Haberleşme komutunu tekrar çalıştırmak zorunda değilsiniz.

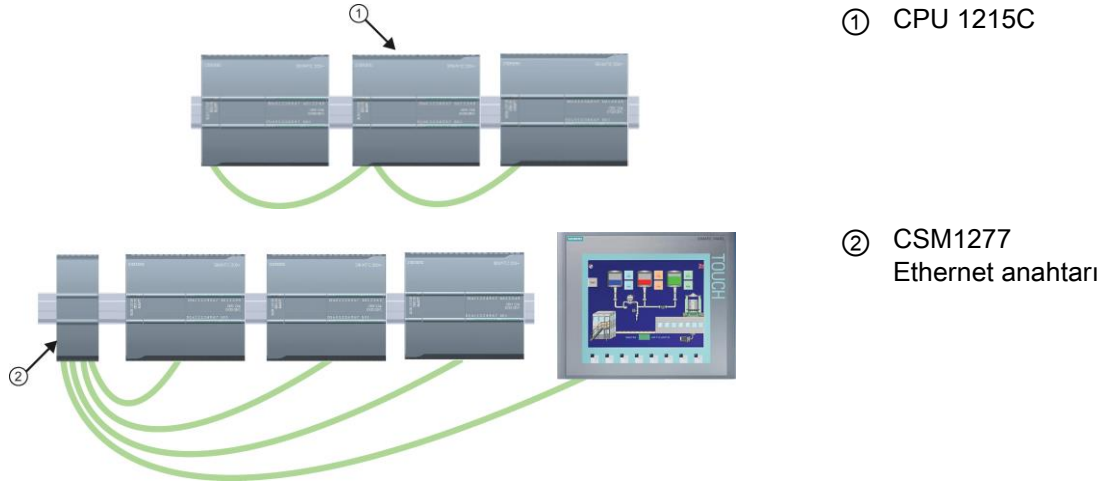
CPU; HMI cihazları, programlama cihazları, diğer CPU'lar ve standart TCP haberleşme protokolleri kullanan Siemens olmayan cihazlar ile haberleşebilir.





## Ethernet anahtarlama

CPU 1211C, 1212C ve 1214C üzerindeki PROFINET portu, bir Ethernet anahtarlama cihazı içermez. Bir programlama cihazı veya HMI ile CPU arasındaki doğrudan bağlantı, Ethernet anahtarı gerektirmez. Ancak ikiden fazla CPU veya HMI cihazına sahip bir ağ bir Ethernet anahtarı gerektirmektedir.



CPU 1215C ve CPU 1217C, dahili 2-port Ethernet anahtarına sahiptir. Bir CPU 1215C ve diğer iki S7-1200 CPU arasında bir ağ kurabilirsiniz. Ayrıca, çoklu CPU ve HMI cihazlarının bağlanması amacıyla rafa monteli CSM1277 4-port Ethernet anahtarı kullanabilirsiniz.

## 10.2.2 Açık kullanıcı haberleşmesi

### 10.2.2.1 Açık kullanıcı haberleşme komutları için bağlantı ID'leri

Kullanıcı programı içine TSEND\_C, TRCV\_C veya TCON PROFINET komutlarını eklediğinizde, STEP 7 cihazlar arasındaki haberleşme kanalını (ya da bağlantı) yapılandırmak için bir kopya DB oluşturur. Bağlantı amacıyla parametreleri yapılandırmak için komutun "Özellikler" ini kullanınız. Parametreler arasında, o bağlantı için bağlantı ID'sidir.

- Bağlantı ID'si CPU için benzersiz olmalıdır. Oluşturduğunuz her bağlantı, farklı DB ve bağlantı ID'sine sahip olmalıdır.
- Yerel CPU ve partner CPU'nun her ikisi, aynı bağlantı için aynı bağlantı ID numarasını kullanabilir, ancak bağlantı ID numaraları eşleşme için gerekli değildir. Bağlantı ID numarası sadece münferit CPU kullanıcı programı içinde PROFINET komutları için geçerlidir.
- CPU bağlantı ID'si için herhangi bir numara kullanabilirsiniz. Ancak, bağlantı ID'lerini "1" den başlayarak sırayla yapılandırmak, belirli bir CPU için kullanılan bağlantıların numarasını izlemek için kolay bir yöntem sağlar.

---

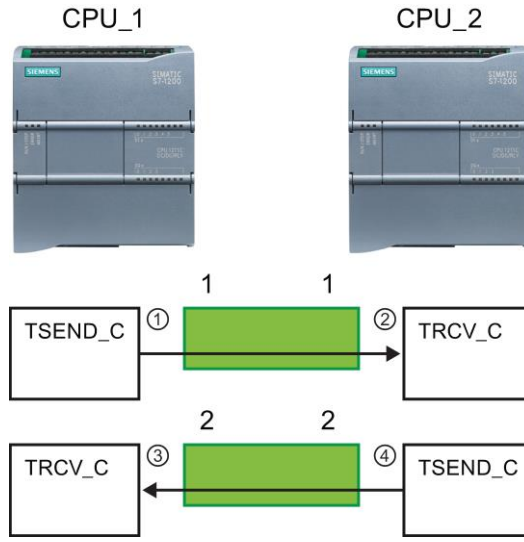
#### Not

Kullanıcı programınızda her TSEND\_C, TRCV\_C veya TCON komutunu yeni bir bağlantı oluşturur. Her bağlantı için doğru bağlantı ID'sini kullanmak önemlidir.

---

Asağıdaki örnek, veri gönderme ve alma için 2 ayrı bağlantı kullanan iki CPU arasındaki haberleşmeyi göstermektedir.

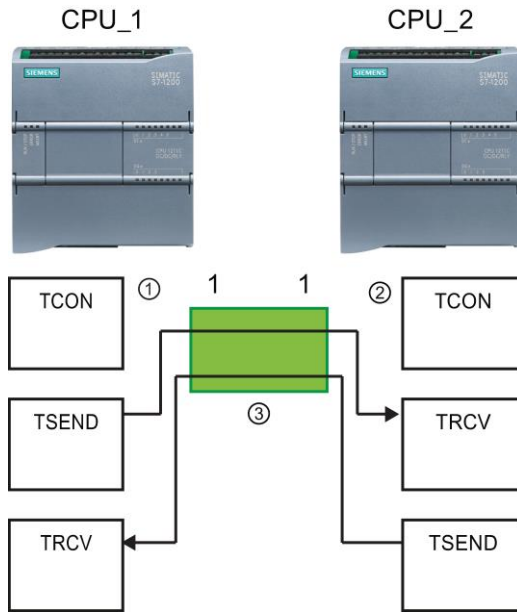
- CPU\_1 deki TSEND\_C komutu ilk bağlantı üzerinden CPU\_2 deki TRCV\_C ile bağlantı sağlar (CPU\_1 ve CPU\_2' nin her ikisi üzerinde "bağlantı ID 1").
- CPU\_1 deki TRCV\_C komutu ikinci bağlantı üzerinden CPU\_2 deki TSEND\_C ile bağlantı sağlar (CPU\_1 ve CPU\_2' nin her ikisi üzerinde "bağlantı ID 2").



- ① CPU\_1 üzerinde TSEND\_C, bir bağlantı oluşturur ve o bağlantıya bir bağlantı ID'si (CPU\_1 için bağlantı ID 1) atar.
- ② CPU\_2 üzerinde TRCV\_C, CPU\_2 için bir bağlantı oluşturur ve o bağlantıya bir bağlantı ID'si (CPU\_2 için bağlantı ID 1) atar.
- ③ CPU\_1 üzerinde TRCV\_C, CPU\_1 için ikinci bir bağlantı oluşturur ve o bağlantıya farklı bir bağlantı ID'si (CPU\_1 için bağlantı ID 2) atar.
- ④ CPU\_2 üzerinde TSEND\_C, ikinci bir bağlantı oluşturur ve o bağlantıya farklı bir bağlantı ID'si (CPU\_2 için bağlantı ID 2) atar.

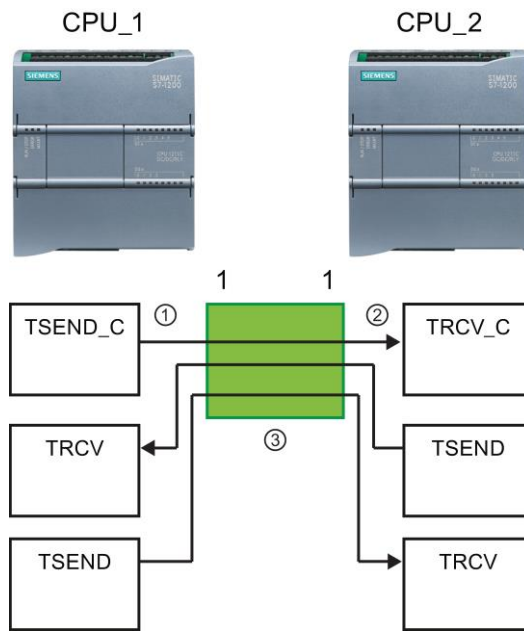
Aşağıdaki örnek, veri alma ve göndermenin her ikisi için 1 bağlantı kullanan iki CPU arasındaki haberleşmeyi göstermektedir.

- Her CPU iki CPU arasındaki bağlantıyı yapılandırmak için bir TCON komutu kullanır.
- CPU\_1 deki TSEND komutu, CPU\_1 deki TCON komutu ile yapılandırılan bağlantı ID'si ("bağlantı ID 1") kullanılarak CPU\_2 deki TRCV komutuna bağlantı yapar. CPU\_2 deki TRCV komutu, CPU\_2 deki TCON komutu ile yapılandırılan bağlantı ID'si ("bağlantı ID 1") kullanılarak CPU\_1 deki TSEND komutuna bağlantı sağlar.
- CPU\_2 deki TSEND komutu, CPU\_2 deki TCON komutu ile yapılandırılan bağlantı ID'si ("bağlantı ID 1") kullanılarak CPU\_1 deki TRCV komutuna bağlantı yapar. CPU\_1 deki TRCV komutu, CPU\_1 deki TCON komutu ile yapılandırılan bağlantı ID'si ("bağlantı ID 1") kullanılarak CPU\_2 deki TSEND komutuna bağlantı sağlar.



- ① CPU\_1 üzerinde TCON, bir bağlantı oluşturur ve CPU\_1 üzerindeki o bağlantıya bir bağlantı ID'si (ID=1) atar.
- ② CPU\_2 üzerinde TCON, bir bağlantı oluşturur ve CPU\_2 üzerindeki o bağlantıya bir bağlantı ID'si (ID=1) atar.
- ③ CPU\_1 üzerindeki TSEND ve TRCV, CPU\_1 üzerindeki TCON ile oluşturulan bağlantı ID'sini kullanır (ID=1). CPU\_2 üzerindeki TSEND ve TRCV, CPU\_2 üzerindeki TCON ile oluşturulan bağlantı ID'sini kullanır (ID=1).

Aşağıdaki örnekte gösterildiği gibi, ayrıca bir TSEND\_C veya TRCV\_C komutu ile oluşturulan bir bağlantı üzerinden haberleşme için bireysel TSEND ve TRCV komutu kullanabilirsiniz. Tsend ve TRCV komutları kendi kendilerine yeni bir bağlantı oluşturmaz, bu nedenle bir TSEND\_C, TRCV\_C veya TCON komutu ile oluşturulan DB ve bağlantı ID'si kullanmak gerekir.



- ① CPU\_1 üzerinde TSEND\_C, bir bağlantı oluşturur ve o bağlantıya bir bağlantı ID'si (ID=1) atar.
- ② CPU\_2 üzerinde TRCV\_C, bir bağlantı oluşturur ve CPU\_2 üzerindeki o bağlantıya bir bağlantı ID'si (ID=1) atar.
- ③ CPU\_1 üzerindeki TSEND ve TRCV, CPU\_1 üzerindeki TSEND\_C ile oluşturulan bağlantı ID'sini kullanır (ID=1). CPU\_2 üzerindeki TSEND ve TRCV, CPU\_2 üzerindeki TRCV\_C ile oluşturulan bağlantı ID'sini kullanır (ID=1).

### Ayrıca bakınız

Yerel / Partner bağlantı yolunu yapılandırma (Sayfa 150)

### 10.2.2.2 Protokoller

CPU entegre PROFINET portu, bir Ethernet ağ üzerinden birden fazla iletişim standartlarını destekler:

- İletim Kontrol Protokolü (TCP)
- TCP üzerinde ISO (RFC 1006)
- Kullanıcı Veri Paketi Protokolü (UDP)

Çizelge 10- 1 Her biri için protokoller ve haberleşme komutları

Protokol	Kullanım örnekleri	Alıcı alanda veri girişi	Haberleşme komutları	Adresleme tipi
TCP	CPU-CPU haberleşme Çerçeve iletimi	Ad hoc modu	Yalnızca TRCV_C ve TRCV	Yerel (aktif) ve Partner (pasif) cihazlar için port numaralarını atar
		Belirtilen uzunlukta veri alımı	TSEND_C, TRCV_C, TCON, TDISCON, TSEND, ve TRCV	
TCP üzerinde ISO	CPU-CPU haberleşme Mesaj bölümlenme ve yeniden birleştirme	Ad hoc modu	Yalnızca TRCV_C ve TRCV	Yerel (aktif) ve Partner (pasif) cihazlar için TSAP'lar atar
		Protokol-kontrollü	TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV	
UDP	CPU-CPU haberleşme Kullanıcı program haberleşmeleri	Kullanıcı Veri Paket Protokolü	TUSEND ve TURCV	Yerel (aktif) ve Partner (pasif) cihazlar için port numaralarını atar, ancak özel bir bağlantı değil
S7 haberleşme	CPU-CPU haberleşmesi CPU'ya veya CPU'dan veri okuma/yazma	Belirtilen uzunlukta veri iletimi ve alımı	GET ve PUT	Yerel (aktif) ve Partner (pasif) cihazlar için TSAP'lar atar
PROFINET RT	CPU-PROFINET IO cihaz haberleşmesi	Belirtilen uzunlukta veri iletimi ve alımı	Dahili	Dahili

### 10.2.2.3 Ad hoc modu

Tipik olarak, TCP ve ISO-on-TCP, 1 ile 8192 bayt arasında değişen, belirli bir uzunlukta veri paketlerini alır. Ancak, TRCV\_C ve TRCV haberleşme komutları 1 ila 1472 bayt arasında değişken uzunlukta veri paketlerini alabilen bir "ad hoc" haberleşme modu da sağlar.

#### Not

Verileri "optimize" bir DB'de (sadece sembolik) depolarsanız sadece Byte, Char, UInt ve Sint veri tipleri dizilerinde veri alabilirsiniz.

Ad hoc modunda TRCV\_C veya TRCV komutunu yapılandırmak için ADHOC komutu giriş parametresini ayarlayınız.

Sık olarak ad hoc modunda TRCV\_C veya TRCV komutunu çağırmasanız, bir çağrıda birden fazla paket alabilirsiniz. Örneğin: bir çağrı ile beş adet 100-bayt paket alsaydınız ISO-on-TCP, paketleri, beş adet 100-bayt paketler halinde yeniden oluştururken, TCP, bu beş paketi bir adet 500-bayt paket olarak teslim ederdi.

#### 10.2.2.4 TCP ve ISO on TCP

Taşıma Kontrol Protokolü (TCP), RFC 793: İletim Kontrol Protokolü tarafından tanımlanan standart bir protokoldür. TCP' nin birincil amacı, süreç çiftleri arasında güvenilir, güvenli bağlantı hizmeti sunmaktır. Bu protokol, aşağıdaki özelliklere sahiptir:

- Donanıma yakın bir biçimde bağlı olduğu için etkili bir haberleşme protokolü
- Orta-büyük ölçekli veri miktarına uygun (en fazla 8192 bayt)
- Uygulamalar için (özellikle hata kurtarma, akış kontrolü ve güvenilirlik) oldukça fazla imkanlar sağlar
- Bağlantı yönelimli protokol
- Sadece TCP'yi destekleyen üçüncü parti sistemler ile çok esnek olarak kullanılabilir
- Yönlendirme özelliği
- Yalnızca statik veri uzunlukları geçerlidir.
- Mesajlar kabul edilmiştir.
- Uygulamalar port numaraları kullanılarak adreslenmektedir.
- Telnet ve FTP gibi kullanıcı uygulama protokollerinin çoğu TCP kullanır.
- Programlama çalışması, GÖNDERME/ALMA programlama ara yüzü nedeniyle veri yönetimi için gereklidir.

Taşıma Denetim Protokolüyle (TCP) (RFC 1006) (TCP on ISO) ilgili Uluslararası Standartlar Organizasyonu (ISO), ISO uygulamalarının TCP / IP ağına yönlendirilmesini sağlayan bir mekanizmadır. Bu protokol, aşağıdaki özelliklere sahiptir:

- Donanıma yakın bir biçimde bağlı bir haberleşme protokolü
- Orta-büyük ölçekli veri miktarına uygun (en fazla 8192 bayt)
- TCP aksine, mesajlar, veri sonu tanımlama özellikte ve mesaj yönlendirmelidir.
- Yönlendirme özelliği; WAN kullanılabilen
- Dinamik veri uzunlukları mümkündür.
- Programlama çalışması, GÖNDERME/ALMA programlama ara yüzü nedeniyle veri yönetimi için gereklidir.

Taşıma Hizmeti Erişim Noktaları (TSAP'lar) kullanılarak, TCP protokolü tek bir IP adresi (64K bağlantılara kadar) için birden fazla bağlantı sağlar. RFC 1006 ile TSAP'lar, bir IP adresi için bu haberleşme uç nokta bağlantılarını benzersiz olarak tanımlar.

**10.2.2.5 TSEND\_C ve TRCV\_C (Ethernet üzerinden veri gönderme ve alma) komutları**

TSEND\_C komutu, TCON, TDISCON ve Tsend komutlarının fonksiyonlarını birleştiriyor. TRCV\_C komutu, TCON, TDISCON ve TRCV komutlarının fonksiyonlarını birleştiriyor. (Bu komutlar hakkında daha fazla bilgi için "TCON, TDISCON, TSEND ve TRCV (TCP haberleşme) komutlarına (Sayfa 527)" bakınız.)

İletebileceğiniz (TSEND\_C) veya (TRCV\_C) alabileceğiniz minimum veri boyutu bir bayt; maksimum boyut 8192 bayttır. TSEND\_C, Boole konumlardan veri iletimini desteklemez ve TRCV\_C, Boole konumlar içine verileri almayacaktır. Bu komutlar ile veri aktarımı hakkında bilgi için, veri tutarlılığı bölümüne bakınız (Sayfa 178).

---

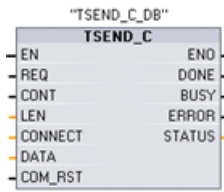
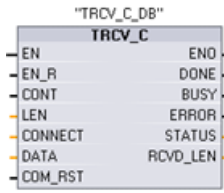
**Not****Haberleşme parametreleri başlatılıyor**

TSEND\_C veya TRCV\_C komutu eklendikten sonra, haberleşme parametrelerini (Sayfa 150) yapılandırmak için komutun "Özellikler" (Sayfa 153) kısmını kullanınız. Kontrolör penceresinde haberleşme partnerleri için parametreler girerken, STEP 7, komut için DB de ilgili verileri girer.

Çoklu kopya DB kullanmak istiyorsanız, her iki CPU'daki DB'yi elle yapılandırmanız gerekir.



Çizelge 10- 2 TSEND\_C ve TRCV\_C komutları

LAD / FBD	SCL	Açıklama
 <p>"TSEND_C_DB" TSEND_C</p> <p>EN, REQ, CONT, LEN, CONNECT, DATA, COM_RST, END, DONE, BUSY, ERROR, STATUS</p>	<pre>"TSEND_C_DB" (   req:=_bool_in_,   cont:=_bool_in_,   len:=_uint_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_word_out_,   connect:=_struct_inout_,   data:=_variant_inout_,   com rst:=_bool_inout_);</pre>	<p>TSEND_C. partner istasyonunu ile TCP veya ISO on TCP haberleşme bağlantısı kurar. veri gönderir ve bağlantı kesebilir. Bağlantı kurulup oluşturulduktan sonra bu bağlantı otomatik olarak sürdürülür ve CPU tarafından izlenir.</p>
 <p>"TRCV_C_DB" TRCV_C</p> <p>EN, EN_R, CONT, LEN, CONNECT, DATA, COM_RST, END, DONE, BUSY, ERROR, STATUS, RCVD_LEN</p>	<pre>"TRCV_C_DB" (   en_r:=_bool_in_,   cont:=_bool_in_,   len:=_uint_in_,   adhoc:=_bool_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_word_out_,   rcvd_len=&gt;_uint_out_,   connect:=_struct_inout_,   data:=_variant_inout_,   com rst:=_bool_inout_);</pre>	<p>TRCV_C. partner CPU ile TCP veya ISO on TCP haberleşme bağlantısı kurar. veri alır ve bağlantı kesebilir. Bağlantı kurulup oluşturulduktan sonra bu bağlantı otomatik olarak sürdürülür ve CPU tarafından izlenir.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Çizelge 10- 3 TSEND\_C ve TRCV\_C Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ (TSEND_C)	IN	Bool	Kontrol parametresi REQ, yükselen kenarda CONNECT de açıklanan bağlantı ile gönderme işini başlatır.
EN_R (TRCV_C)	IN	Bool	Kontrol parametresi almak için etkinleştirildi: EN_R = 1 olduğunda, TRCV_C almaya hazırdır. Alma işi işlenir.
CONT	IN	Bool	<ul style="list-style-type: none"> <li>0: Bağlantı kes</li> <li>1: Bağlantı oluştur ve tut</li> </ul>
LEN	IN	UInt	<p>Gönderilecek (TSEND_C) veya alınacak (TRCV_C) maksimum bayt sayısı:</p> <ul style="list-style-type: none"> <li>Varsayılan = 0: DATA parametresi (TSEND_C) gönderilecek veya (TRCV_C) alınacak verilerin uzunluğunu belirler.</li> <li>Ad hoc modu = 65535: verilerin değişken uzunluğu, alım (TRCV_C) için ayarlanır.</li> </ul>
CONNECT	IN_OUT	TCON_Param	Bağlantı açıklamasını işaret eder (Sayfa153)

Parametre ve tipi		Veri tipi	Açıklama
DATA	IN_OUT	Variant	<ul style="list-style-type: none"> <li>Gönderilecek (TSEND_C) verilerin adres ve uzunluğunu içerir</li> <li>Alınan verilerin (TRCV_C) başlangıç adresi ve maksimum uzunluğunu içerir.</li> </ul>
COM_RST	IN_OUT	Bool	<p>Komutun yeniden başlatılmasına izin verir:</p> <ul style="list-style-type: none"> <li>0: Alakalı değil</li> <li>1: Fonksiyon bloğunun yeniden başlatılmasını tamamla, mevcut bağlantı sona erdirilecek.</li> </ul>
DONE	OUT	Bool	<ul style="list-style-type: none"> <li>0: İş hala başlamadı veya çalışmaya devam ediyor.</li> <li>1: İş hatasız tamamlandı.</li> </ul>
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>0: İş tamamlandı.</li> <li>1: İş henüz tamamlanmadı. Yeni bir iş tetiklenemiyor.</li> </ul>
ERROR	OUT	Bool	<p>Aşağıdaki dereceye sahip durum parametreleri:</p> <ul style="list-style-type: none"> <li>0: Hata yok</li> <li>1: Proses sırasında hata oluştu. STATUS, hata tipi hakkında ayrıntılı bilgi sağlar.</li> </ul>
STATUS	OUT	Word	Hata bilgileri de dahil olmak üzere durum bilgileri. (Aşağıdaki "Hata ve Durum Parametreleri" çizelgesine bakınız.)
RCVD_LEN (TRCV_C)	OUT	Int	Gerçekte alınan veri miktarı, bayt olarak

#### Not

TSEND\_C komutu, gönderme işini başlatmak için REQ giriş parametresinde bir geçiş gerektirir. BUSY parametresi, daha sonra proses sırasında 1'e ayar edilir. Gönderme işinin tamamlanması, tek tarama için 1'e ayarlanan DONE veya ERROR parametrelerinden biriyle belirtilir. Bu süre boyunca, REQ giriş parametresinde herhangi bir geçiş göz ardı edilir.

#### Not

LEN parametresinin varsayılan değeri (LEN=0) iletilen verinin uzunluğunu belirlemek için DATA parametresini kullanır. TSEND\_C komutu tarafından iletilen DATA'nın TRCV\_C komutunun DATA parametresiyle aynı boyutta olduğundan emin olunuz.

## TSEND\_C işlemleri

Aşağıdaki fonksiyonlar TSEND\_C komutunun çalışmasını açıklar:

- Bir bağlantı kurmak için CONT = 1 ile TSEND\_C yürütülür.
- Bağlantı başarılı bir şekilde oluşturulduktan sonra TSEND\_C bir çevrim için DONE parametresini ayarlar.
- Haberleşme bağlantısını sonlandırmak için CONT = 0 ile TSEND\_C yürütülür. Bağlantı derhal iptal edilecektir. Bu, aynı zamanda, alıcı istasyonunu etkiler. Bağlantı, orada kapatılacaktır ve alma arabelleği içinde veri kaybı olabilir.
- Kurulmuş bir bağlantı üzerinden veri göndermek için REQ üzerinde yükselen kenarı ile TSEND\_C yürütünüz. Başarılı bir gönderme işleminden sonra, TSEND\_C bir çevrim için DONE parametresini ayarlar.
- Bir bağlantı kurmak ve veri göndermek için CONT = 1 ve REQ = 1 ile TSEND\_C yürütülür. Başarılı gönderme işleminden sonra, TSEND\_C bir çevrim için DONE parametresini ayarlar.

## TRCV\_C işlemleri

Aşağıdaki fonksiyonlar TRCV\_C komutunun çalışmasını açıklar:

- Bir bağlantı kurmak için parametre CONT = 1 ile TRCV\_C yürütünüz.
- Veri almak için parametre EN\_R = 1 ile TRCV\_C yürütünüz. Parametreler EN\_R = 1 ve CONT = 1 olduğunda TRCV\_C sürekli veri alır.
- Bağlantıyı sonlandırmak için parametre CONT = 0 ile TRCV\_C yürütünüz. Bağlantı hemen iptal edilecektir ve veri kaybı olabilir.

TRCV\_C, TRCV komutuyla aynı alma modlarını işler. Aşağıdaki çizelgede, verilerin, alma alanına nasıl girildiğini gösterir:

Çizelge 10- 4 Verilerin alma alanına girilişi

Protokol değişkeni	Verilerin alma alanına girilişi	Parametre "connection_type"	LEN parametre değeri	RCVD_LEN parametre değeri (bayt)
TCP	Ad hoc modu	B#16#11	65535	1 ila 1472
TCP	Belirtilen uzunlukta veri alımı	B#16#11	0 (tavsiye edilen) veya 1 ila 8192, 65535 hariç	1 ila 8192
ISO on TCP	Ad hoc modu	B#16#12	65535	1 ila 1472
ISO on TCP	Protokol-kontrollü	B#16#12	0 (tavsiye edilen) veya 1 ila 8192, 65535 hariç	1 ila 8192

**Not**
**Ad hoc modu**

"Ad hoc modu", TCP ve ISO on TCP protokol varyantlarında bulunur. "ad hoc modu" nu LEN parametresine "65535" atayarak ayarlayınız. Alma alanı, DATA ile biçimlendirilen alanla özdeşdir. Alınan verilerin uzunluğu RCVD\_LEN parametresine çıkış olacaktır.

Verileri "optimize" bir DB'de (sadece sembolik) depolarsanız sadece Byte, Char, USInt ve Sint veri tiplerindeki dizilerde veri alabilirsiniz.

**Not**
**"ad hoc modu" içeren S7-300/400 STEP 7 projelerini S7-1200 içine aktarma**

S7-300/400 STEP 7 projelerde, "ad hoc mode", LEN parametresine "0" atayarak seçilir. S7-1200'de, LEN parametresine "65535" atayarak "ad hoc modu" nu ayarlarsınız.

"ad hoc modu" içeren bir S7-300/400 STEP 7 projesini S7-1200 içine aktarırsanız, LEN parametresini "65535" olarak değiştirmeniz gerekir.

**Not**
**DONE parametresi veya ERROR parametresi TRUE değerini alana kadar gönderici alanındaki verileri uyumlu tutmalısınız**

TSEND\_C asenkron proses nedeniyle, DONE parametresi veya ERROR parametresi TRUE değerini alana kadar, gönderici alanındaki verileri uyumlu tutmalısınız.

TSEND\_C için DONE parametresindeki TRUE durumu, verinin başarılı gönderildiği anlamına gelir. Bu bağlantı partneri CPU aslında alınan arabelleği okuduğu anlamına gelmez.

TRCV\_C asenkron proses nedeniyle alıcı alandaki veriler sadece parametre DONE = 1 olduğunda uyumlu olur.

Çizelge 10- 5 TSEND\_C ve TRCV\_C komutları BUSY, DONE ve ERROR parametreleri

BUSY	DONE	ERROR	Açıklama
TRUE	irrelevant	irrelevant	İş işlemeye devam ediyor.
FALSE	TRUE	FALSE	İş başarılı bir şekilde tamamlandı.
FALSE	FALSE	TRUE	İş bir hata ile sonlandı. Hatanın sebebi STATUS parametresinde bulunabilir.
FALSE	FALSE	FALSE	Yeni bir iş atanmadı.

**TSEND\_C ve TRCV\_C Error ve Status durum kodları**

ERROR	STATUS	Açıklama
0	0000	İş hatasız yürütüldü
0	7000	İşlenen aktif iş yok
0	7001	İş prosesine başla, bağlantı kur, bağlantı partnerini bekle

ERROR	STATUS	Açıklama
0	7002	Gönderilen veya alınan veri
0	7003	Bağlantı sonlandırıldı
0	7004	Bağlantı oluşturuldu ve izleniyor, işlenen aktif iş yok
1	8085	LEN parametresi izin verilen en büyük değerden büyük.
1	8086	CONNECT parametresi izin verilen aralığın dışında.
1	8087	Maksimum bağlantı sayısına ulaşıldı; İlave bağlantı mümkün değil.
1	8088	DATA'da belirtilen bellek alanı için LEN parametresi geçerli değil.
1	8089	CONNECT parametresi veri bloğunu işaret etmez.
1	8091	Maksimum yuvalama derinliği aşıldı.
1	809A	CONNECT parametresi, bağlantı açıklama uzunluğuyla eşleşmeyen bir alanı işaret ediyor.
1	809B	Bağlantıda açıklamasındaki local_device_id CPU ile eşleşmiyor.
1	80A1	Haberleşme hatası: <ul style="list-style-type: none"> <li>• Belirtilen bağlantı henüz kurulmamıştır</li> <li>• Belirtilen bağlantı şu anda sonlandırılıyor; Bu bağlantı üzerinden iletimi mümkün değildir</li> <li>• Ara yüz yeniden başlatılıyor</li> </ul>
1	80A3	Var olmayan bir bağlantıyı sonlandırmak için yapılan girişimi
1	80A4	Uzak ortak bağlantının IP adresi geçersiz. Örneğin, uzak ortak bir IP adresi yerel ortak IP adresi olarak aynı.
1	80A5	Connection ID (Sayfa 514) zaten kullanımda.
1	80A7	Haberleşme hatası: TSEND_C tamamlanmadan önce TDISCON çağırınız.
1	80B2	CONNECT parametresi Unlinked anahtar kelime ile oluşturulan bir veri bloğunu işaret eder.
1	80B3	Tutarsız parametreler: <ul style="list-style-type: none"> <li>• Bağlantı açıklamasında hata</li> <li>• Yerel port (parametre local_tsap_id) zaten başka bir bağlantı açıklamasında mevcuttur.</li> <li>• Parametre olarak belirtilen ID den farklı bağlantı açıklamasındaki ID</li> </ul>
1	80B4	Pasif bir bağlantı kurmak için ISO on TCP kullanırken (connection_type = B # 16 # 12) girilen TSAP'ın aşağıdaki adres gereksinimlerinden birine uymadığı hususunda 80B4 durum kodu sizi uyarır: <ul style="list-style-type: none"> <li>• Uzunluğu 2 olan yerel TSAP için ve ilk baytı E0 veya E1 (onaltılık) olan TSAP ID değeri için, ikinci bayt 00 veya 01 olmalıdır.</li> <li>• Uzunluğu 3 veya daha büyük olan yerel TSAP için ve ilk baytı E0 veya E1 (onaltılık) olan bir TSAP ID değeri için, ikinci byte 00 veya 01 olmalıdır ve diğer tüm baytlar geçerli ASCII karakteri olmalıdır.</li> <li>• Uzunluğu 3 veya daha büyük olan yerel TSAP ve ilk bayt değeri E0 veya E1 (onaltılık) olmayan TSAP ID için TSAP ID'nin tüm baytları geçerli ASCII karakter olmalıdır.</li> </ul> Geçerli ASCII karakterleri 20 ila 7E (onaltılık) bayt değerlerinde olmalıdır.
1	80B7	Veri tipi ve/veya iletilen verilerin uzunluğu yazılacak olduğu partner CPU alanına uymuyor.
1	80C3	Tüm bağlantı kaynaklar kullanılmaktadır.

ERROR	STATUS	Açıklama
1	80C4	Geçici haberleşme hatası: <ul style="list-style-type: none"><li>Bağlantı bu sefer kurulamaz</li><li>Ara yüzü yeni parametreleri alıyor</li><li>Yapılandırılmış bağlantı şu anda TDISCON tarafından kaldırılıyor.</li></ul>
1	8722	Parametre CONNECT: Kaynak alanı geçersiz: alan, DB de yok.
1	873A	Parametre CONNECT: Bağlantı açıklamasına erişim mümkün değildir (örneğin, DB mevcut değil)
1	877F	Parametre CONNECT: geçersiz ANY referans gibi dahili hata:
1	893A	Parametre, yüklü olmayan bir DB numarası içerir.

### Bağlantı Ethernet protokolleri

Her CPU'nun standart PROFINET haberleşmesini destekleyen tümleşik bir PROFINET portu vardır. TSEND\_C ve TRCV\_C ve TSEND ve TRCV komutlarının hepsi TCP ve ISO on TCP Ethernet protokollerini destekler.

Daha fazla bilgi için "Cihaz Yapılandırma Yerel / Partner bağlantı yolunun (Sayfa 150) yapılandırılması" na bakılmalıdır.

#### 10.2.2.6 TCON, TDISCON, TSEND ve TRCV (TCP haberleşme) komutları

#### TCP ve ISO on TCP protokollerini kullanan Ethernet haberleşme

##### Not

##### TSEND\_C ve TRCV\_C komutları

PROFINET/Ethernet haberleşme programlamayı kolaylaştırmaya yardımcı olmak için TSEND\_C komutu ile TRCV\_C komutu; TCON, TDISCON, TSEND ve TRCV komutlarının işlevselliğini birleştirir:

- TSEND\_C; TCON, TDISCON ve TSEND komutlarını birleştirir.
- TRCV\_C; TCON, TDISCON ve TRCV komutlarını birleştirir.

Aşağıdaki komutlar haberleşme sürecini kontrol eder:

- TCON, istemci ve sunucu ( CPU ) PC arasında TCP/IP bağlantısı kurar.
- TSEND ve TRCV veri alır ve gönderir.
- TDISCON bağlantıyı keser.

İletilebileceğiniz (TSEND) veya alabileceğiniz (TRCV) minimum veri boyutu bir bayt; maksimum boyutu 8192 bayttır. TSEND Boole konumlardan veri iletimini desteklemez ve TRCV Boole konumlar içine verileri almayacaktır. Bu komutlar ile veri aktarımıyla ilgili bilgi için, veri tutarlılığı (Sayfa 178 ) bölümüne bakınız.

TCON, TDISCON, TSEND ve TRCV asenkron çalışır. Bu da iş prosesinin çoklu komut yürütmesi boyunca devam ettiği anlamına gelir. Örneğin, bir bağlantı kurmak ve oluşturmak için parametre REQ = 1 ile bir TCON komutunu yürüterek bir iş başlatırsınız. Sonra parametre DONE ile işin tamamlandığını test etmek ve iş ilerlemesini izlemek için ek TCON yürütmelerini kullanırsınız.

Aşağıdaki Çizelge BUSY, DONE ve ERROR arasındaki ilişkileri gösterir. Geçerli iş durumunu belirlemek için bu çizelgeyi kullanınız:

Çizelge 10- 6 BUSY, DONE ve ERROR parametreler arasındaki etkileşim

BUSY	DONE	ERROR	Açıklama
TRUE	irrelevant	irrelevant	İş işlenmeye devam ediyor.
FALSE	TRUE	FALSE	İş başarılı şekilde tamamlandı.
FALSE	FALSE	TRUE	İş bir hata ile sonlandı. Hatanın nedeni STATUS parametresinde bulunabilir.
FALSE	FALSE	FALSE	Yeni bir iş atanmadı.

## TCON ve TDISCON

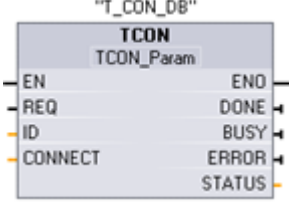
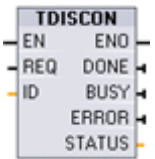
### Not

#### Haberleşme parametrelerinin başlatılması

TCON komutunu yerleştirdikten sonra, haberleşme parametrelerini (Sayfa 150) yapılandırmak için komutun "Özellikler" (Sayfa 153) ini kullanınız. Kontrolör penceresinde haberleşme partnerleri için parametreleri girerken, STEP 7, komut için kopya DB'de karşılık gelen verileri girer.

Çoklu kopya DB kullanmak istiyorsanız, her iki CPU'daki DB'yi elle yapılandırmanız gerekir.

Çizelge 10- 7 TCON ve TDISCON komutları

LAD / FBD		Açıklama
	<pre>"TCON_DB" (   req:=_bool_in_,   ID:=_undef_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status =&gt;_word_out_,   connect:= struct inout );</pre>	TCP ve ISO on TCP: TCON, CPU' dan haberleşme ortağına bir haberleşme bağlantısı başlatır.
	<pre>"TDISCON_DB" (   req:=_bool_in_,   ID:=_word_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status =&gt;_word_out_);</pre>	TCP ve ISO on TCP: TDISCON, CPU' dan haberleşme ortağına olan haberleşme bağlantısını keser.

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Çizelge 10- 8 TCON ve TDISCON parametreleri için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
REQ	IN	Bool
ID	IN	CONN_OUC (Word)
CONNECT (TCON)	IN_OUT	TCON_Param
DONE	OUT	Bool
BUSY	OUT	Bool
ERROR	OUT	Bool
STATUS	OUT	Word

Kontrol parametresi REQ ID tarafından belirtilen bağlantı kurarak işini başlatır. İş yükselen kenarında başlar.

(TCON) kurulmuş veya uzak partnerinizde, ya da kullanıcı programı ve işletim sisteminin haberleşme tabakası arasında (TDISCON) sona erdirilecek bağlantı referans. ID yerel bağlantı açıklamasında ilişkili parametre ID aynı olmalıdır. Değer aralığı: W#16#0001 ila W#16#0FFF

Bağlantı açıklamasına işaretçi (Sayfa 153)

- 0: İş henüz başlamadı veya hala çalışıyor.
- 1: İş hatasız tamamlandı.

- 0: İş tamamlandı.
- 1: İş henüz tamamlanmadı. Yeni bir iş tetiklenemiyor.

Aşağıdaki değerlere sahip durum parametreleri:

- 0: Hata yok
- 1: Proses sırasında hata oluştu. STATUS, hata tipi hakkında ayrıntılı bilgi sağlar.

Hata bilgisi içeren durum bilgisi. (Aşağıda verilen hata ve durum kodlarına bakılmalıdır.)

Her iki haberleşme partneri haberleşme bağlantısını kırmak ve oluşturmak için TCON komutunu yürütür. Aktif ve pasif haberleşme son nokta partnerlerini belirlemek için parametreleri kullanınız. Bağlantı kırıp oluşturduktan sonra, bu bağlantı otomatik olarak sürdürülür ve CPU tarafından izlenir.



Hat konması veya uzak haberleşme partneri nedeniyle bağlantı sonlandırılırsa örneğin, aktif partner vanılandırılmış bağlantıyı yeniden oluşturmaya çalışır. TCON komutunu tekrar yürütmek zorunda değilsiniz.

TDISCON komutu yürütüldüğünde veya CPU STOP moduna geçtiği zaman var olan bir bağlantı sonlandırılır ve kurulum bağlantısı kaldırılır. Bağlantı kurmak ve yeniden oluşturmak için, TCON komutunu tekrar yürütmeniz gerekir.

Çizelge 10- 9 TCON ve TDISCON için ERROR ve STATUS durum kodları

ERROR	STATUS	Açıklama
0	0000	Bağlantı başarılı şekilde oluşturuldu.
0	7000	Devam eden aktif iş yok
0	7001	İş prosesini başlat; bağlantı oluştur (TCON) veya bağlantı sonlandır (TDISCON)
0	7002	Çağrıyı takip et (REQ irrelevant); bağlantı oluştur (TCON) veya bağlantı sonlandır (TDISCON)
1	8086	ID parametresi izin verilen adres aralığının dışında.
1	8087	TCON: Maksimum bağlantı sayısına ulaşıldı; ilave bağlantı mümkün değil.
1	809B	TCON: bağlantıda açıklamasındaki local_device_id CPU ile eşleşmiyor.
1	80A1	TCON: Bağlantı veya port kullanıcı tarafından çoktan kullanılmış durumda.
1	80A2	TCON: Yerel veya uzak port sistem tarafından kullanılıyor.
1	80A3	Mevcut bağlantıyı yeniden oluşturmak (TCON) için yapılan girişim veya var olmayan bir bağlantıyı sonlandırmak (TDISCON).
1	80A4	TCON: Uzak bağlantı uç noktasının IP adresi geçersiz; Bu yerel IP adresiyle eşleşebilir.
1	80A5	TCON: Bağlantı ID (Sayfa 514) kullanılmaktadır.
1	80A7	TCON: Haberleşme hatası: TCON tamamlanmadan önce TDISCON yürüttünüz. TDISCON ilk olarak ID tarafından referanslanan bağlantıyı tamamen sonlandırmalıdır.
1	80B4	TCON: Pasif bir bağlantı kurmak için ISO on TCP kullanırken (connection_type = B # 16 # 12) girilen TSAP' ın aşağıdaki adres gereksinimlerinden birine uymadığı hususunda 80B4 durum kodu sizi uyarır: <ul style="list-style-type: none"> <li>• Uzunluğu 2 olan yerel TSAP için ve ilk baytı E0 veya E1 (onaltılık) olan TSAP ID değeri için ikinci bayt 00 veya 01 olmalıdır.</li> <li>• Uzunluğu 3 veya daha büyük olan yerel TSAP için ve ilk baytı E0 veya E1 (onaltılık) olan bir TSAP ID değeri için ikinci byte 00 veya 01 olmalı ve diğer tüm baytlar geçerli ASCII karakteri olmalıdır.</li> <li>• Uzunluğu 3 veya daha büyük olan yerel TSAP için ve ilk bayt değeri E0 veya E1 (onaltılık) olmayan TSAP ID için TSAP ID'nin tüm baytları geçerli ASCII karakter olmalıdır.</li> </ul> Geçerli ASCII karakterleri 20 ila 7E (onaltılık) bayt değerlerinde olmalıdır.
1	80B6	TCON: Parametre connection_type' da parametre atama hatası
1	80B7	TCON: Veri tipi ve/veya iletilen verilerin uzunluğu yazılacak olduğu partner CPU alanına uymuyor.
1)	80B8	TCON: Yerel bağlantı açıklamasındaki Parametre ve Parametre ID farklı.

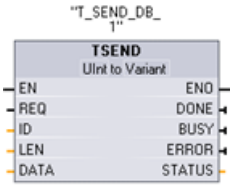
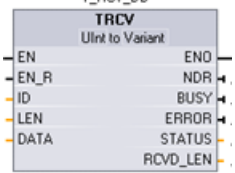
ERROR	STATUS	Açıklama
1	80C3	TCON: Tüm bağlantı kaynaklar kullanılmaktadır.
1	80C4	Geçici haberleşme hatası: <ul style="list-style-type: none"> <li>Bağlantı bu sefer kurulamaz (TCON).</li> <li>Yapılandırılmış bağlantı şu anda TDISCON tarafından kaldırılıyor (TCON).</li> <li>Bağlantı hali hazırda oluşturuluyor (TDISCON).</li> <li>Ara yüzü yeni parametreleri alıyor (TCON ve TDISCON).</li> </ul>

## TSEND ve TRCV

### Not

PROFINET Açık Kullanıcı haberleşmesi kullanılırken, uzak cihazda karşılık gelen TRCV komutunu yürütmeksizin bir TSEND komutunu yürütürseniz, TSEND komutu, "Meşgul durum" da süresiz durabilir ve TRCV komutunun verileri almasını bekler. Bu durumda, TSEND komutu "Busy" çıkışı ayarlanır ve "Status" çıkışı "0x7002" bir değere sahip olur. 4096 bayttan fazla veri aktardığınız takdirde bu durum oluşabilir. Sorun TRCV komutunun bir sonraki yürütülmesinde çözümlenir.

Çizelge 10- 10 TSEND ve TRCV komutları

LAD / FBD	SCL	Açıklama
 <p>"T_SEND_DB_1"</p> <p>TSEND UInt to Variant</p> <p>EN, REQ, ID, LEN, DATA, ENQ, DONE, BUSY, ERROR, STATUS</p>	<pre>"TSEND_DB" (     req:=_bool_in_,     ID:=_word_in_,     len:=_udint_in_,     done=&gt;_bool_out_,     busy=&gt;_bool_out_,     error=&gt;_bool_out_,     status =&gt;_word_out_,     data:=_variant_inout_);</pre>	TCP ve ISO on TCP: TSEND, CPU' dan partner istasyona bir haberleşme bağlantısı üzerinden veri gönderir.
 <p>"T_RCV_DB"</p> <p>TRCV UInt to Variant</p> <p>EN, EN_R, ID, LEN, DATA, ENQ, NDR, BUSY, ERROR, STATUS, RCVD_LEN</p>	<pre>"TRCV_DB" (     en_r:=_bool_in_,     ID:=_word_in_,     len:=_udint_in_,     ndr=&gt;_bool_out_,     busy=&gt;_bool_out_,     error=&gt;_bool_out_,     status=&gt;_word_out_,     rcvd_len=&gt;_udint_out_,     data:=_variant_inout_);</pre>	TCP ve ISO on TCP: TRCV, partner istasyonundan CPU' ya haberleşme bağlantısı üzerinden veri alır.

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Çizelge 10- 11 TSEND ve TRCV parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
REQ	IN	Bool TSEND: Bir yükselen kenarda gönderme işini başlatır. Veriler, DATA ve LEN ile belirtilen alandan aktarılır.
EN_R	IN	Bool TRCV: Alacak CPU'yu etkinleştirir; EN_R = 1 ile TRCV almaya hazırdır. Alınan iş işlenir.
ID	IN	CONN_OUC (Word) İlgili bağlantıya referans. ID yerel bağlantı açıklamasında ilişkili parametre ID ile aynı olmalıdır. Değer aralığı: W#16#0001 - W#16#0FFF
LEN	IN	UInt Gönderilecek (TSEND) veya alınacak (TRCV) maksimum bayt sayısı: <ul style="list-style-type: none"> <li>Varsayılan = 0: DATA parametresi gönderilecek (TSEND) veya alınacak (TRCV) veri uzunluğunu belirler</li> <li>Ad hoc modu = 65535: değişken uzunluktaki veriler alımı için ayarlanır (TRCV).</li> </ul>
DATA	IN_OUT	Variante Gönderilen (TSEND) veya alınan (TRCV) veri alanına işaret edici; Veri alanı adres ve uzunluk içerir. Adres, I belleği, Q belleği, M belleği veya bir DB' yi belirtir.
DONE	OUT	Bool TSEND: <ul style="list-style-type: none"> <li>0: İş henüz başlatılmadı veya hala çalışıyor.</li> <li>1: İş hatasız yürütüldü.</li> </ul>
NDR	OUT	Bool TRCV: <ul style="list-style-type: none"> <li>NDR = 0: İş henüz başlatılmadı veya hala çalışıyor.</li> <li>NDR = 1: İş başarılı tamamlandı.</li> </ul>
BUSY	OUT	Bool <ul style="list-style-type: none"> <li>BUSY = 1: İş henüz tamamlanmadı. Yeni bir iş tetiklenemiyor.</li> <li>BUSY = 0: İş tamamlandı.</li> </ul>
ERROR	OUT	Bool ERROR = 1: Proses sırasında hata oluştu. STATUS, hata tipi hakkında ayrıntılı bilgi sağlar
STATUS	OUT	Word Hata bilgisi içeren durum bilgisi. (Aşağıda verilen hata ve durum kodlarına bakılmalıdır.)
RCVD_LEN	OUT	Int TRCV: Gerçekte alınan veri miktarı, bayt olarak

### Not

TSEND komutu, gönderme işini başlatmak için REQ giriş parametresinde bir geçiş gerektirir. BUSY parametresi, daha sonra proses sırasında 1'e ayar edilir. Gönderme işinin tamamlanması, tek tarama için 1'e ayarlanan DONE veya ERROR parametrelerinden biriyle belirtilir. Bu süre boyunca, REQ giriş parametresinde herhangi bir geçiş göz ardı edilir.

**TRCV İşlemleri**

TRCV komutu aşağıdaki iki değişken tarafından belirtilen bir alma alanına alınan verileri yazar:

- Alanının başlangıcına işaretçi
- Alanının uzunluğu veya LEN girişinde sağlanan değeri (0 değilse)

---

**Not**

LEN parametresinin varsayılan değeri (LEN=0) iletilen verinin uzunluğunu belirlemek için DATA parametresini kullanır. TSEND komutu tarafından iletilen DATA'nın TRCV komutunun DATA parametresiyle aynı boyutta olduğundan emin olunuz.

---

Tüm iş verisi alınır alınmaz, TRCV bu verileri alma alanına aktarır ve NDR'yi 1 olarak ayarlar.

Çizelge 10- 12 Verilerin alma alanına girişi

Protokol değişkeni	Verilerin alma alanına girişi	Parametre "connection_type"	LEN parametre değeri	RCVD_LEN parametre değeri (bayt)
TCP	Ad hoc modu	B#16#11	65535	1 ila 1472
TCP	Belirtilen uzunlukta veri alımı	B#16#11	0 (tavsiye edilen) veya 1 ila 8192, 65535 hariç	1 ila 8192
ISO on TCP	Ad hoc modu	B#16#12	65535	1 ila 1472
ISO on TCP	Protokol-kontrollü	B#16#12	0 (tavsiye edilen) veya 1 ila 8192, 65535 hariç	1 ila 8192

**Not****Ad hoc modu**

"Ad hoc modu", TCP ve ISO on TCP protokol varyantları ile bulunur. "ad hoc modu" nu LEN parametresine "65535" atayarak ayarlayınız. Alma alanı, DATA ile biçimlendirilen alanla özdeştir. Alınan verilerin uzunluğu RCVD\_LEN parametresine çıkış olacaktır. Bir veri bloğunun alınmasından hemen sonra, TRCV verileri alma alanına girer ve NDR 1 olarak ayarlanır.

Verileri "optimize" bir DB (sadece sembolik) de depolarsanız sadece Byte, Char, USInt ve Sint veri tiplerindeki dizilerde veri alabilirsiniz.

**Not****"ad hoc modu" içeren S7-300/400 STEP 7 projelerini S7-1200 içine aktarma**

S7-300/400 STEP 7 projelerde, "ad hoc mode", LEN parametresine "0" atayarak seçilir. S7-1200 de, LEN parametresine "65535" atayarak "ad hoc modu" nu ayarlarsınız.

"ad hoc modu" içeren bir S7-300/400 STEP 7 projesini S7-1200 içine aktarırsanız, LEN parametresini "65535" olarak değiştirmeniz gerekir.

### TSEND ve TRCV Error ve Status durum kodları

ERROR	STATUS	Açıklama
0	0000	<ul style="list-style-type: none"> <li>Gönderme işi hatasız tamamlandı (TSEND)</li> <li>Yeni veri kabul edildi: Alınan verilerin (TRCV) mevcut uzunluğu RCVD_LEN de gösterilmektedir.</li> </ul>
0	7000	<ul style="list-style-type: none"> <li>Devam eden aktif iş yok (TSEND)</li> <li>Blok almaya hazır değil (TRCV)</li> </ul>
0	7001	<ul style="list-style-type: none"> <li>İş prosesinin başlangıcı, gönderilen veri: Bu proses sırasında işletim sistemi DATA gönderme alanındaki verilere erişir (TSEND).</li> <li>Blok almaya hazır, alınan iş aktiveştirildi (TRCV).</li> </ul>
0	7002	<ul style="list-style-type: none"> <li>Komut yürütülmesini izle (REQ irrelevant), iş işleniyor: Bu proses sırasında işletim sistemi DATA gönderme alanındaki verilere erişir (TSEND).</li> <li>Komut yürütülmesini izle, alınan iş işleniyor: Bu proses sırasında veri alma alanına yazılıyor. Bu nedenle, Bir hata alma alanında tutarsız veriye neden olur (TRCV).</li> </ul>
1	8085	<ul style="list-style-type: none"> <li>LEN parametresi izin verilen en büyük değerden daha büyük (TSEND) ve (TRCV).</li> <li>İlk komut yürütüldüğünden beri LEN veya DATA parametresi değişti (TRCV).</li> </ul>
1	8086	ID parametresi izin verilen adres aralığında değil.
1	8088	LEN parametresi DATA'da belirtilen bellek alanından daha büyük.
1	80A1	<p>Haberleşme hatası:</p> <ul style="list-style-type: none"> <li>Belirtilen bağlantı henüz kurulmamıştır (TSEND ve TRCV).</li> <li>Belirtilen bağlantı şu anda sonlandırılıyor; Bu bağlantı üzerinden iletimi mümkün değildir (TSEND ve TRCV).</li> <li>Ara yüz yeniden başlatılıyor (TSEND).</li> <li>Ara yüz yeni parametreler alıyor (TRCV).</li> </ul>
1	80C3	Dahili kaynak eksikliği: Bu ID ile bir blok zaten farklı bir öncelik sınıfında işleniyor.
1	80C4	<p>Geçici haberleşme hatası:</p> <ul style="list-style-type: none"> <li>Haberleşme partneriyle bağlantı bu sefer oluşturulamaz.</li> <li>Ara yüz yeni parametre ayarlarını alıyor, ya da bağlantı hali hazırda kuruluyor.</li> </ul>

### Bağlantı Ethernet protokolleri

Her CPU'nun standart PROFINET haberleşmesini destekleyen tümleşik bir PROFINET portu vardır. TSEND\_C TRCV\_C TSEND ve TRCV komutlarının hepsi TCP ve ISO on TCP Ethernet protokollerini destekler.

Daha fazla bilgi için "Cihaz Yapılandırması Yerel/Partner bağlantı yolunun (Sayfa 150) yapılandırılması" bakılmalıdır.

### 10.2.2.7 UDP

UDP, RFC 768 ile tanımlanan standart bir protokoldür: Kullanıcı Veri Paket Protokolü. UDP bir uygulamadan diğerine veri paketi göndermek için bir mekanizma sağlar; Ancak, verilerin teslimi garantili değildir. Bu protokol, aşağıdaki özelliklere sahiptir:

- Hızlı haberleşme protokolü. Çünkü donanımla çok yakın ilişkilidir.
- Küçük-Orta ölçekli veri miktarına uygun (en fazla 1472 bayt)
- UDP, TCP'den daha basit bir taşıma kontrol protokolüdür ve düşük işletim yükü üreten ince bir tabakaya sahiptir
- Birçok üçüncü parti sistemler ile çok esnek olarak kullanılabilir
- Yönlendirme özelliği
- Veri paketlerini yönlendirmek için port numaraları kullanır
- Mesajlar kabul edilmiştir: Uygulamanın, hata kurtarma ve güvenlik sorumluluğunu alması gereklidir
- Programlama çalışması, GÖNDERME/ALMA programlama ara yüzü nedeniyle veri yönetimi için gereklidir

UDP yayın haberleşmesini destekler. Yayını kullanmak için, ADDR yapılandırmasının IP adresi kısmını yapılandırmanız gerekir. Örneğin: Bir 192.168.2.10 IP adresi ve 255.255.255.0 alt ağ maskesine sahip CPU, 192.168.2.255 bir yayın adresini kullanabilir.

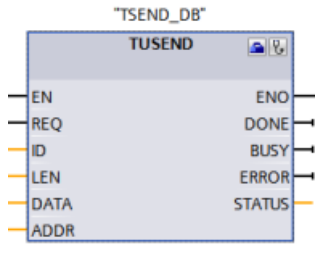
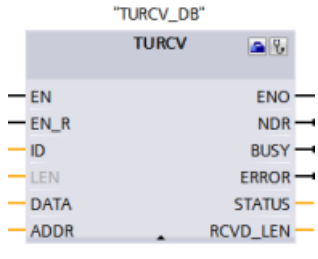
### 10.2.2.8 TUSEND ve TURCV

Aşağıdaki komutlar haberleşme sürecini kontrol eder:

- TCON, istemci ve sunucu ( CPU ) PC arasında haberleşme sağlar.
- TUSEND ve TURCV veri alır ve gönderir.
- TDISCON istemci ve sunucu arasındaki bağlantıyı keser.

" TCON ve TDISCON haberleşme komutları hakkında daha fazla bilgi için TCP ve ISO-on-TCP" bölümündeki TCON, TDISCON, TSEND ve TRCV (Sayfa 527) bakınız.

Çizelge 10- 13 TUSEND ve TURCV komutları

LAD / FBD	SCL	Açıklama
 <p>"TSEND_DB" TUSEND</p> <p>EN, REQ, ID, LEN, DATA, ADDR (Inputs) ENO, DONE, BUSY, ERROR, STATUS (Outputs)</p>	<pre>"TUSEND_DB" (   req:=_bool_in_,   ID:=_word_in_,   len:=_udint_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_word_out_,   data:=_variant_inout_);</pre>	<p>TUSEND komutu adres parametresi tarafından belirtilen uzak bir partnere UDP aracılığıyla veri gönderir.</p> <p>Veri gönderme amacıyla işi başlatmak için, REQ = 1 ile TUSEND komutunu çağırın diyoruz.</p>
 <p>"TURCV_DB" TURCV</p> <p>EN, EN_R, ID, LEN, DATA, ADDR (Inputs) ENO, NDR, BUSY, ERROR, STATUS, RCVD_LEN (Outputs)</p>	<pre>"TURCV_DB" (   en_r:=_bool_in_,   ID:=_word_in_,   len:=_udint_in_,   ndr=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status =&gt;_word_out_,   rcvd_len=&gt;_udint_out_,   data:=_variant_inout_);</pre>	<p>TURCV komutu UDP üzerinden verileri alır. ADDR Parametresi gönderenin adresini gösterir. TURCV in başarıyla tamamlanmasından sonra, ADDR parametresi uzak partnerinizin (gönderici) adresini içerir.</p> <p>TURCV ad hoc modunu desteklemez.</p> <p>Veri alma amacıyla işi başlatmak için, EN_R = 1 ile TURCV komutunu çağırınız.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

TCON, TDISCON, TUSEND ve TURCV komutları asenkron olarak çalışır, yani iş prosesi çoklu komut yürütmeleri boyunca devam eder.

Çizelge 10- 14 TUSEND ve TURCV Parametreleri için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ (TUSEND)	IN	Bool	Bir yükselen kenarda gönderme işini başlatır. Veriler, DATA ve LEN ile belirtilen alandan aktarılır.
EN_R (TURCV)	IN	Bool	<ul style="list-style-type: none"> <li>0: CPU alamaz.</li> <li>1: CPU nun almasını sağlar. TURCV komutu almaya hazır ve alınan iş işlenir.</li> </ul>
ID	IN	Word	Kullanıcı programı ve işletim sisteminin haberleşme düzeyi arasındaki ilişkili bağlantıya Referans. ID yerel bağlantı açıklamasında ilişkili parametre ID ile aynı olmalıdır. Değerler aralığı: W#16#0001 - W#16#0FFF.
LEN	IN	UDInt	Gönderilecek (TUSEND) veya alınacak (TURCV) bayt sayısı. <ul style="list-style-type: none"> <li>Varsayılan = 0. DATA parametresi gönderilecek veya alınacak veri uzunluğunu belirler.</li> <li>Aksi taktirde, değerler aralığı: 1 - 1472</li> </ul>
DONE (TUSEND)	IN	Bool	Durum parametresi DONE (TUSEND): <ul style="list-style-type: none"> <li>0: İş henüz başlatılmadı veya hala çalışıyor.</li> <li>1: İş hatasız tamamlandı.</li> </ul>



Parametre ve tipi		Veri tipi	Açıklama
NDR (TURCV)	OUT	Bool	Durum parametresi NDR (TURCV): <ul style="list-style-type: none"> <li>0: İş henüz başlatılmadı veya hala çalışıyor.</li> <li>1: İş başarıyla tamamlandı.</li> </ul>
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>1: İş henüz tamamlanmadı. Yeni bir iş tetiklenemiyor.</li> <li>0: İş tamamlandı.</li> </ul>
ERROR	OUT	Bool	Aşağıdaki değerlere sahip durum parametreleri: <ul style="list-style-type: none"> <li>0: Hata yok</li> <li>1: Proses sırasında hata oluştu. STATUS, hata tipi hakkında ayrıntılı bilgi sağlar.</li> </ul>
STATUS	OUT	Word	Hata bilgisi içeren durum bilgisi. (Aşağıda verilen hata ve durum kodlarına bakılmalıdır.)
RCVD_LEN	OUT	UDInt	Alınan bayt sayısı (TURCV)
DATA	IN_OUT	Variant	Gönderici alanın (TURCV) veya alıcı alanın adresi (TUSEND): <ul style="list-style-type: none"> <li>Proses görüntü giriş çizelgesi</li> <li>Proses görüntü çıkış çizelgesi</li> <li>Bir bellek biti</li> <li>Bir veri bloğu</li> </ul>
ADDR	IN_OUT	Variant	Alıcı (TURCV için) veya göndericinin (TUSEND için) adresine işaretçi (örneğin, P # DB100.DBX0.0 bayt 8) pointer. İşaretçi herhangi bir bellek alanına işaret edebilir. Aşağıdaki 8 baytlık bir yapı gereklidir: <ul style="list-style-type: none"> <li>İlk 4 bayt uzak IP adresini içerir.</li> <li>Sonraki 2 bayt uzak port numarasını belirtir.</li> <li>Son 2 bayt rezerve edilmiştir.</li> </ul>

İş durumu BUSY ve STATUS çıkış parametrelerinde belirtilir. STATUS asenkron olarak çalışan komutların RET\_VAL çıkış parametresine karşılık gelir.

Aşağıdaki çizelge BUSY, DONE (TUSEND), NDR (TURCV) ve ERROR arasındaki ilişkileri gösterir. Bu çizelgeyi kullanarak, (TUSEND veya TURCV) komutunun geçerli durumunu veya gönderme (iletim)/alma işleminin ne zaman tamamlandığını belirleyebilirsiniz.

Çizelge 10- 15 BUSY, DONE (TUSEND) / NDR (TURCV), ve ERROR parametrelerinin durumu

BUSY	DONE / NDR	ERROR	Açıklama
TRUE	irrelevant	irrelevant	İş işleniyor.
FALSE	TRUE	FALSE	İş başarılı bir şekilde tamamlandı.
FALSE	FALSE	TRUE	İş bir hata ile sonlandı. Hatanın nedeni STATUS parametresinde bulunabilir.
FALSE	FALSE	FALSE	Yeni bir iş için komut atanmadı.

<sup>1</sup> Komutların asenkron çalışması nedeniyle: DONE parametresi veya ERROR parametresi TRUE değerini alana kadar TUSEND için gönderici alanındaki verileri tutarlı yapmak zorundasınız. TURCV için alıcı alandaki veriler sadece NDR parametresi TRUE değerini aldığı zaman tutarlı olur.

Çizelge 10- 16 ERROR ve STATUS için TUSEND ve TURCV durum kodları

ERROR	STATUS	Açıklama
0	0000	<ul style="list-style-type: none"> <li>Gönderme işi hatasız tamamlandı (TUSEND).</li> <li>Yeni veriler kabul edildi. Alınan verilerin mevcut uzunluğu gösterilmektedir RCVD_LEN (TURCV).</li> </ul>
0	7000	<ul style="list-style-type: none"> <li>Devam eden aktif iş yok (TUSEND)</li> <li>Blok almaya hazır değil (TURCV)</li> </ul>
0	7001	<ul style="list-style-type: none"> <li>İş prosesinin başlangıcı, gönderilen veri (TUSEND): Bu proses sırasında, işletim sistemi DATA gönderme alanındaki verilere erişir.</li> <li>Blok almaya hazır, alınan iş aktiveştirildi (TURCV).</li> </ul>
0	7002	<ul style="list-style-type: none"> <li>Komut yürütülmesini izle (REQ irrelevant), iş işleniyor (TUSEND): Bu proses sırasında, işletim sistemi DATA gönderme alanındaki verilere erişir.</li> <li>Komut yürütülmesini izle, iş işleniyor: Bu proses sırasında, TURCV komutu alıcı Alana veri yazar. Bu nedenle, bir hata alma alanında tutarsız veriye neden olur.</li> </ul>
1	8085	LEN parametresi izin verilen en büyük değerden daha büyük ve 0 değerine sahip (TUSEND), veya ilk komut yürütüldüğünden beri LEN veya DATA parametresinin değerini değiştirdiniz (TURCV).
1	8086	ID parametresi izin verilen adres aralığında değil.
1	8088	<ul style="list-style-type: none"> <li>LEN parametresi bellek alanından veya DATA da belirtilen alıcı alandan (TURCV) büyük (TUSEND).</li> <li>Alıcı alan çok küçük (TURCV).</li> </ul>
1	8089	ADDR parametresi bir veri bloğunu işaret etmez.
1	80A1	Haberleşme hatası: <ul style="list-style-type: none"> <li>Kullanıcı programı ile işletim sisteminin haberleşme katmanı arasında belirtilen bağlantı henüz oluşturulmadı.</li> <li>Kullanıcı programı ile işletim sisteminin haberleşme katmanı arasında belirtilen bağlantı hali hazırda sonlandırılıyor. Bu bağlantı üzerinden iletim (TUSEND) veya alma işi (TURCV) mümkün değildir.</li> <li>Ara yüz yeniden başlatılıyor.</li> </ul>
1	80A4	Uzak bağlantı uç noktasının IP adresi geçersiz; bu IP adresinin yerel IP adresiyle eşleşmesi mümkündür (TUSEND).
1	80B3	<ul style="list-style-type: none"> <li>Set protokol varyantı (bağlantı açıklamasında connection_type parametresi) UDP değildir. Lütfen TSEND veya TRCV komutunu kullanınız.</li> <li>ADDR parametresi: port numarası (TUSEND) için geçersiz ayarlar</li> </ul>
1	80C3	<ul style="list-style-type: none"> <li>Bu ID ile bir blok zaten farklı bir öncelik sınıfında işleniyor.</li> <li>Dahili kaynak eksikliği</li> </ul>
1	80C4	Geçici Haberleşme hatası: <ul style="list-style-type: none"> <li>Kullanıcı programı ile işletim sisteminin haberleşme düzeyi arasındaki bağlantı e sefer oluşturulamadı (TUSEND).</li> <li>Ara yüz yeni parametreler alıyor (TUSEND).</li> <li>Bağlantı hali hazırda yeniden başlatılıyor (TURCV).</li> </ul>

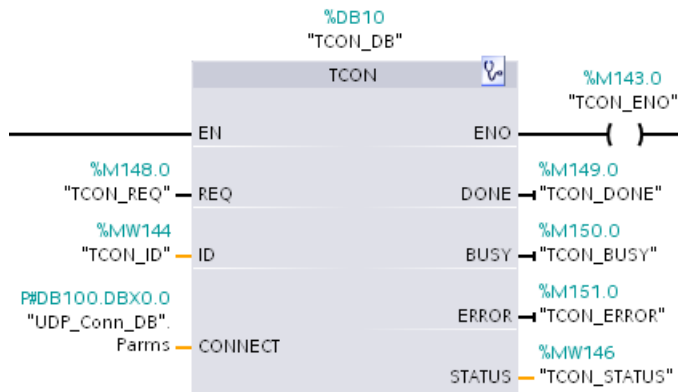
## Ethernet protokollerinin bağlantısı

Her CPU'nun standart PROFINET haberleşmesini destekleyen tümlşik bir PROFINET portu vardır. TSEND ve TURCV komutlarının hepsi UDP Ethernet protokolünü destekler.

Daha fazla bilgi için "Cihaz Yapılandırması" bölümünde "Yerel / Partner bağlantı yolunun (Sayfa 150) yapılandırılması" na bakılmalıdır.

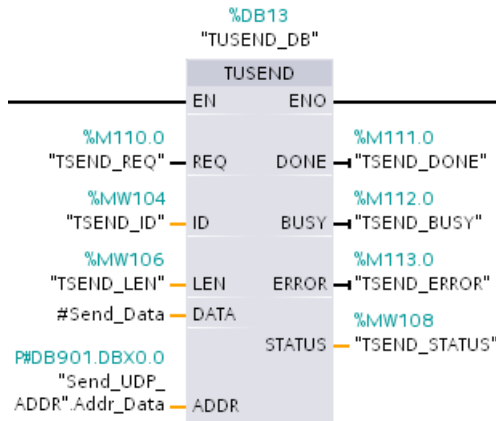
## Operasyonlar

Her iki partner, UDP haberleşmesinde pasiftir. "TCON\_Param" veri tipi için tipik parametre başlangıç değerleri aşağıdaki şekillerde gösterilmiştir. Port numaraları (LOCAL\_TSAP\_ID) 2-bayt biçiminde yazılır. 161, 34962, 34963 ve 34964 dışındaki tüm portlara izin verilir.



UDP_Conn_DB					
	Name	Data type	Offset	Initial value	Comment
1	Static				
2	Parms	TCON_Param	0.0		
3	BLOCK_LENGTH	UInt	0.0	64	byte length of SDT
4	ID	CONN_OUC	2.0	1	reference to the connection
5	CONNECTION_TYPE	USInt	4.0	19	17: TCP/IP, 18: ISO on TCP
6	ACTIVE_EST	Bool	5.0	false	active/passive connection establishment
7	LOCAL_DEVICE_ID	USInt	6.0	1	1: local IE interface
8	LOCAL_TSAP_ID_L...	USInt	7.0	2	byte length of local TSAP id/port number
9	REM_SUBNET_ID_...	USInt	8.0	0	byte length of remote subnet id
10	REM_STADDR_LEN	USInt	9.0	0	byte length of remote IP address
11	REM_TSAP_ID_LEN	USInt	10.0	0	byte length of remote port/TSAP id
12	NEXT_STADDR_LEN	USInt	11.0	0	byte length of next station address
13	LOCAL_TSAP_ID	Array[1..16] of Byte	12.0		TSAP id/local port number
14	LOCAL_TSAP_I...	Byte		B#16#07	
15	LOCAL_TSAP_I...	Byte		B#16#D0	

TUSEND komutu "TADDR\_Param" veri tipinde belirtilen uzak partnere UDP üzerinden veri gönderir. TURCV komutu UDP üzerinden veri alır. Aşağıdaki şekillerde gösterildiği gibi TURCV komutunun başarılı bir şekilde yürütülmesinden sonra, "TADDR\_Param" veri tipi, uzak partner (gönderici) adresini gösterir.



Send_UDP_ADDR					
	Name	Data type	Offset	Initial value	Comment
1	Static				
2	Addr_Data	TADDR_Param	0.0		
3	REM_IP_ADDR	Array[1..4] of USint	0.0		remote station address
4	REM_IP_ADDR[1]	USint		192	
5	REM_IP_ADDR[2]	USint		168	
6	REM_IP_ADDR[3]	USint		2	
7	REM_IP_ADDR[4]	USint		10	
8	REM_PORT_NBR	UInt	4.0	2000	remote port number
9	RESERVED	Word	6.0	0	unused; has to be 0

### 10.2.2.9 T\_CONFIG

T\_CONFIG komut aşağıdaki özelliklerin kalıcı bir değişiklik veya ayarı sağlayan, kullanıcı programı PROFINET portu IP yapılandırma parametrelerini değiştirir:

- İstasyon adı
- IP adresi

- Alt ağ maskesi
- Yönlendirici adresi

### Not

CPU "özellikleri" "Ethernet adresi" sayfasında bulunan farklı bir yöntem kullanarak IP adresini ayarla" (Sayfa 546) radyo düğmesi, IP adresini Çevrimiçi olarak değiştirmenize izin verir ya da program indirildikten sonra "T\_CONFIG" komutunu kullanarak. Bu IP adresi atama yöntemi yalnızca CPU içindir.

CPU "özellikleri" "Ethernet adresi" sayfasında bulunan "farklı bir yöntem kullanarak IP adresini ayarla" (Sayfa 547) radyo düğmesi, PROFINET cihaz ismini Çevrimiçi olarak değiştirmenize izin verir ya da program indirildikten sonra "T\_CONFIG" komutunu kullanarak. Bu PROFINET cihaz ismini atama yöntemi yalnızca CPU içindir.

### ⚠ UYARI

#### T\_CONFIG ile IP yapılandırma parametresinin değiştirilmesi CPU nun yeniden başlamasına neden olur.

Bir IP yapılandırma parametresini değiştirmek için T\_CONFIG kullanıldıktan sonra CPU yeniden başlatılır. CPU, STOP moduna geçer, sıcak yeniden başlatma gerçekleştirir ve RUN moduna döner.

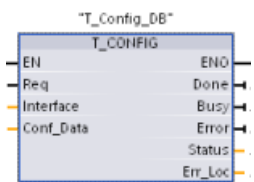
Bir üretim ortamında T\_CONFIG komutunu kullanmayınız. Kontrol cihazları, güvensiz bir durumda arızalanabilir. Bu durum kontrol edilen donanımın beklenmeyen çalışmasına yol açar. Bu tarz beklenmeyen çalışmalar, personelin ölümüne veya ciddi yaralanmasına ve/veya donanım hasarına neden olabilir.

T\_CONFIG komutu yürütmesi sonucu olarak CPU sıcak yeniden başlatma gerçekleştirdiğinde prosesinizin güvenli bir duruma gideceğinden emin olunuz. Sıcak yeniden başlatma, belleği sıfırlamaz. Sıcak yeniden başlatma, bütün kalıcı olmayan sistemi ve kullanıcı verilerini başlatır ve tüm kalıcı kullanıcı verilerinin değerlerini tutar.

### Not

Bir seferde birden fazla T\_CONFIG komutunu yürütmeye çalışmayınız.

Çizelge 10- 17 T\_CONFIG komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a function block named "T_CONFIG" within a data block "T_Config_DB". It has several inputs and outputs: EN (Enable), Req (Request), Interface, Conf_Data, Status, and Err_Loc (Error Location). The outputs are ENO (Enable Out), Done, Busy, Error, and Err_Loc.</p>	<pre>"T_CONFIG_DB" (   req:= bool_in_,   interface:=_word_in_,   conf_Data:=_variant_in_,   done=&gt;_bool_out_,   busy=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_dword_out_,   err_loc=&gt;_word_out_);</pre>	<p>Kullanıcı programınızdan IP yapılandırma parametrelerini değiştirmek için T_CONFIG komutunu kullanınız.</p> <p>T_CONFIG asenkron çalışır. Yürütme birden fazla çağrı boyunca devam eder.</p>

Çizelge 10- 18 T\_CONFIG Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
REQ	Input	Bool	Komut yükselen kenarda başlar.
INTERFACE	Input	HW_Interface	Şebeke ara yüzü ID'si
CONF_DATA	Input	Variant	Konfigürasyon verileri yapısına referans; CONF_DATA Bir Sistem Veri tipine (SDT) göre tanımlanır.
DONE	Çıkış	Bool	<ul style="list-style-type: none"> <li>0: İş henüz başlamadı veya hala çalışıyor.</li> <li>1: İş hatasız yürütüldü.</li> </ul>
BUSY	Çıkış	Bool	<ul style="list-style-type: none"> <li>0: İş tamamlandı.</li> <li>1: İş henüz tamamlanmadı. Yeni bir iş tetiklenemiyor.</li> </ul>
ERROR	Çıkış	Bool	Aşağıdaki değerlere sahip durum parametreleri: <ul style="list-style-type: none"> <li>0: Hata yok</li> <li>1: Proses sırasında hata oluştu. STATUS, hata tipi hakkında ayrıntılı bilgi sağlar.</li> </ul>
STATUS	Çıkış	Dword	Hata bilgisi içeren durum bilgisi. (Aşağıda verilen hata ve durum kodlarına bakılmalıdır.)
ERR_LOC	Çıkış	Dword	Arıza yeri bulma (hata parametresinin saha ID ve alt saha ID'si)

IP yapılandırma bilgileri yukarıda başvuru CONF\_DATA parametresi üzerinde bir Variant işaretçi ile birlikte, CONF\_DATA veri bloğuna yerleştirilir. T\_CONFIG komutunun başarılı bir şekilde yürütülmesi, IP yapılandırma verilerinin ağ ara yüzüne teslim edilmesi ile sona erer. Hatalar STATUS çıkış parametresine atanır.

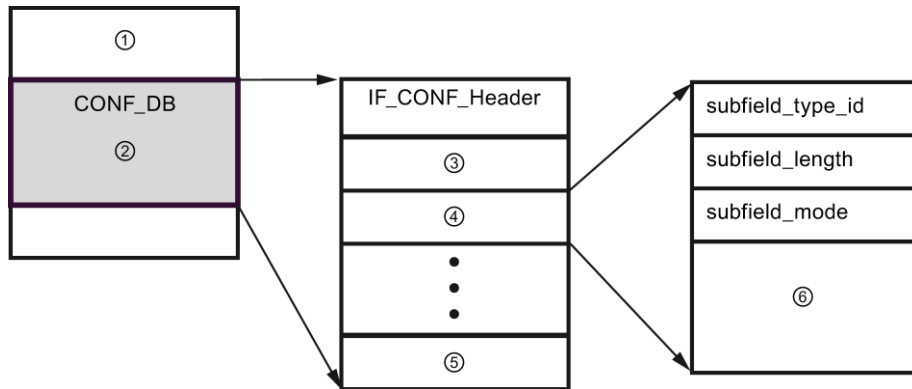
Çizelge 10- 19 ERROR ve STATUS Durum kodları

ERROR	STATUS (DW#16#...)	Açıklama
0	00000000	Hata yok Not: Komut başarılı şekilde yürütülürse, "Hata yok" durumu döndürülemeyebilir.
0	00700000	İş bitmedi (BUSY = 1).
0	00700100	İş yürütümünün başlangıcı
0	00700200	Ara çağrı (REQ irrelevant)
1	C08xyy00	Genel arıza
1	C0808000	Ara yüzünün tanımlanması için LADDR parametreleri geçersiz.
1	C0808100	Ara yüzün tanıtımı için LADDR parametreleri, desteklenmeyen bir donanım ara yüzü için atanmıştır.
1	C0808200	CONF_DATA parametre hatası: Varyant işaretçi veri tipi, veri tipi baytıyla eşleşmiyor.
1	C0808300	CONF_DATA parametre hatası: Alan işaretçi, variant işaretçisinin DB sinde değil.
1	C0808400	CONF_DATA parametre hatası: Alan işaretçi yanlış uzunlukta
1	C0808600	Rezerve edildi
1	C0808700	IP yapılandırması için CONF_DATA veri bloğu uzunluğunda tutarsızlık
1	C0808800	CONF_DATA veri bloğu field_type_id parametreleri geçersiz. (yalnızca field_type_id = 0 izin verilir)

ERROR	STATUS (DW#16#...)	Açıklama
1	C0808900	CONF_DATA veri bloğu field_type_id parametreleri geçersiz veya bir kaç defa kullanıldı.
1	C0808A00	IP yapılandırma parametrelerinin LEN uzunluğu veya subfield_cnt hataları
1	C0808B00	IP yapılandırma ID parametresi geçersiz veya desteklenmiyor.
1	C0808C00	IP yapılandırmasının Alt bloğu yanlış (Alt-blok yanlış, yanlış emir veya birden çok kez kullanılmış) yerleştirilmiş.
1	C0808D00	LEN Alt-bloklar ifadesinin uzunluğu geçersiz.
1	C0808E00	Alt-blok modunda parametre değeri geçersiz.
1	C0808F00	IP yapılandırması ve bir önceki alt-blok arasındaki alt-blok çakışması.
1	C0809000	Alt alan parametreleri yazma-korumalı (örneğin: parametreler yapılandırma tarafından belirtilir veya PNIO modu etkinleştirilir).
1	C0809100	Rezerve edildi
1	C0809400	Alt-blok IP yapılandırmasında bir parametre tanımlı değil ya da kullanılamaz.
1	C0809500	Bir alt-blok IP yapılandırma parametresi ile diğer parametreler arasında bir tutarsızlık var.
1	C080C200	Komut yürütülemez. Örneğin, ara yüz ile bağlantı kesildiğinde bu hata oluşabilir.
1	C080C300	Yeterli kaynak yok. Örneğin, komut farklı parametrelerle birden çok kez çağrılırsa bu hata oluşabilir
1	C080C400	Haberleşme hatası. Hata geçici olarak oluşabilir ve kullanıcı programının tekrarını gerektirir.
1	C080D200	Komutun Yürütülmesi, PROFINET ara yüzü tarafından desteklenmiyor.

## CONF\_DATA Veri bloğu

Aşağıdaki diyagram, aktarılabacak yapılandırma verilerinin yapılandırma DB'sinde nasıl depolandığını gösterir.



- |   |                     |   |                             |
|---|---------------------|---|-----------------------------|
| ① | Yapılandırma DB     | ④ | Alt alan 2                  |
| ② | Yapılandırma verisi | ⑤ | Alt alan n                  |
| ③ | Alt alan 1          | ⑥ | Alt alana özel parametreler |

CONF\_DB ye ait konfigürasyon verileri bir başlık (IF\_CONF\_Header) ve çok sayıda alt alan içeren bir alandan oluşur. IF\_CONF\_Header aşağıdaki elemanları sağlar:

- field\_type\_id (Veri tipi UInt): Sıfır
- field\_id (Veri tipi UInt): Sıfır
- subfield\_cnt (Veri tipi UInt): Alt alan sayısı

Her alt alan bir başlık (subfield\_type\_id, subfield\_length, subfield\_mode) ve alt alana özel parametrelerden oluşur. Her alt alan, çift sayıda baytlardan oluşmalıdır. Subfield\_mode 1 değerini destekler.

#### Not

Şu anda sadece bir alana (IF\_CONF\_Header) izin verilir. Onun field\_type\_id ve field\_id parametrelerinin değeri sıfır olmalıdır. Field\_type\_id ve field\_id için farklı değerlere sahip diğer alanlar gelecekteki ilavelere maruz kalır.

IF\_CONF\_Header alanında, sadece iki alt alana, "addr" (IP adresi) ve "nos" (istasyon adı) şu anda izin verilir.

Çizelge 10- 20 Desteklenen alt alanlar

subfield_type_id	Veri tipi	Açıklama
30	IF_CONF_V4	IP parametresi: IP adresi, Alt ağ maskesi, router adresi
40	IF_CONF_NOS	PROFINET IO cihaz ismi (istasyon ismi)

Çizelge 10- 21 IF\_CONF\_V4 Veri tipi elemanları

İsim	Veri tipi	Başlangıç değeri	Açıklama
Id	UInt	30	subfield_type_id
len	UInt	18	subfield_length
mode	UInt	1	subfield_mode (1: kalıcı)
InterfaceAddress	IP_V4	-	Ara yüz adresi
ADDR	Dizi [1..4] Bayt		
	ADDR[1]	Bayt	b#16#C8 IP adresi yüksek bayt: 200
	ADDR[2]	Bayt	b#16#0C IP adresi yüksek bayt: 12
	ADDR[3]	Bayt	b#16#01 IP adresi düşük bayt: 1
	ADDR[4]	Bayt	b#16#90 IP adresi düşük bayt: 144
SubnetMask	IP_V4	-	Alt ağ maskesi
ADDR	Array [1..4] of Byte		
	ADDR[1]	Bayt	b#16#FF Alt ağ maskesi yüksek bayt: 255
	ADDR[2]	Bayt	b#16#FF Alt ağ maskesi yüksek bayt: 255
	ADDR[3]	Bayt	b#16#FF Alt ağ maskesi düşük bayt : 255
	ADDR[4]	Bayt	b#16#00 Alt ağ maskesi düşük bayt: 0
DefaultRouter	IP_V4	-	Varsayılan router
ADDR	Dizi [1..4] Bayt		
	ADDR[1]	Bayt	b#16#C8 Router yüksek bayt: 200



İsim	Veri tipi	Başlangıç değeri	Açıklama
ADDR[2]	Bayt	b#16#0C	Router yüksek bayt: 12
ADDR[3]	Bayt	b#16#01	Router düşük bayt: 1
ADDR[4]	Bayt	b#16#01	Router düşük bayt: 1

Çizelge 10- 22 IF\_CONF\_NOS Veri tipi elemanları

İsim	Veri tipi	Başlangıç değeri	Açıklama
id	UInt	40	subfield_type_id
len	UInt	246	subfield_length
mode	UInt	1	subfield_mode (1: kalıcı)
Nos (İstasyon ismi)	Dizi[1..240] Bayt	0	İstasyon ismi: İlk bayttan başlayarak ARRAY doldurulmalıdır. ARRAY, atanacak istasyon adından daha uzun ise, gerçek istasyon adından (IEC 61158-6-10 ile uyumlu) sonra bir sıfır bayt girmelisiniz. Aksi takdirde, nos reddedilir ve "T_CONFIG (Sayfa 541)" komutu STATUS içinde DW # 16 # C0809400 hata kodunu girer. İlk bayt sıfır ile doldurulursa, istasyon ismi silinir.

İstasyon adı, aşağıdaki sınırlamalara tabidir:

- İstasyon adının içinde bir isim bileşeni, yani, iki nokta arasında bir karakter string, 63 karakteri geçmemelidir.
- Üst üste iki nokta, parantez, alt çizgi, eğik çizgi, boşluk gibi özel karakterler yoktur. İzin verilen tek özel karakter tire (-) çizgisidir.
- İstasyon adı "-" karakteri ile başlamaz veya bitemez.
- İstasyon adı bir sayı ile başlamamalıdır.
- İstasyon adı biçimi n.n.n.n (n = 0, ... 999) izin verilmez.
- İstasyon adı, "port-xyz" veya "port-xyz-abcde" string ile başlamalıdır (a, b, c, d, e, x, y, z = 0, ... 9).

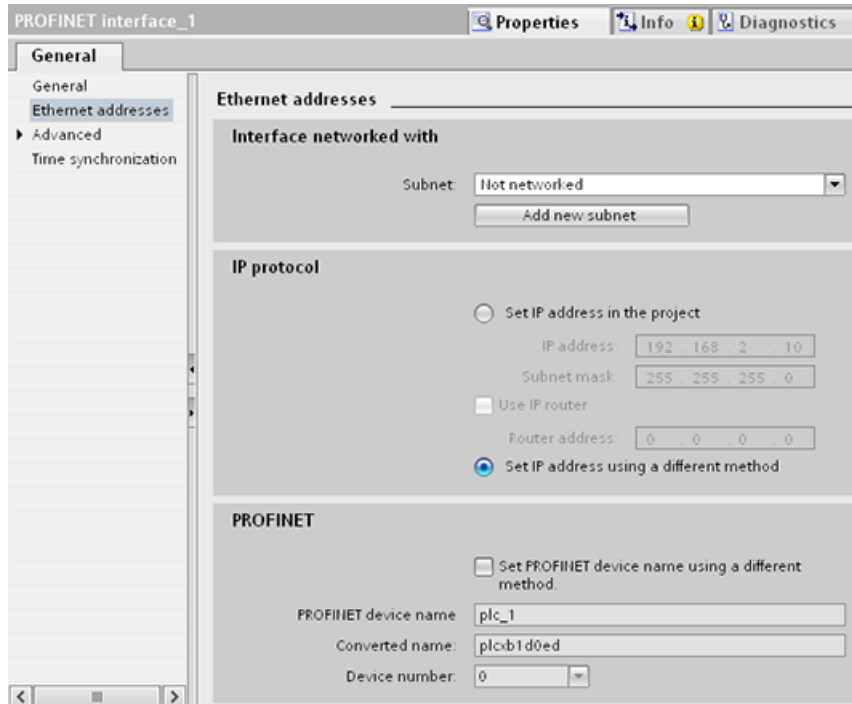
### Not

Ayrıca en az 2 bayt olmak şartıyla 240 bayttan kısa bir ARRAY "Nos" yaratabilirsiniz. Bu durumda, buna göre "len" (alt alanın uzunluğu) etiketini ayarlamanız gerekir.

## IP parametreleri nasıl değiştirilir

Aşağıdaki örnekte, "addr" alt alanında, "InterfaceAddress" (IP adresi) "SubnetMask" ve "DefaultRouter" (IP yönlendirici) değiştirilir. CPU "Properties", "Ethernet address" sayfasında, program yüklendikten sonra "T\_CONFIG" komutunu kullanarak IP adresini değiştirmenize imkan vermek için "Set IP address using a different method" radyo düğmesi tıklanmalıdır.

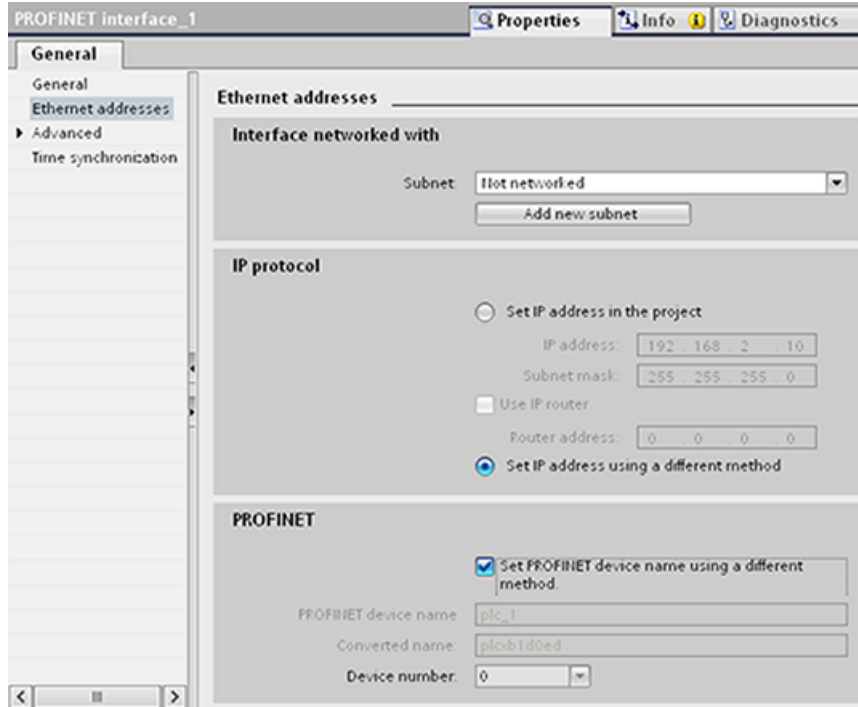
CONF_DATA_1			
	Name	Data type	Start value
1	Static		
2	Conf_data	Struct	
3	header	IF_CONF_Header	
4	FieldType	UInt	0
5	FieldId	UInt	0
6	SubfieldCount	UInt	1
7	addr	IF_CONF_v4	
8	Id	UInt	30
9	Length	UInt	18
10	Mode	UInt	1
11	InterfaceAddress	IP_V4	
12	ADDR	array [1..4] of Byte	
13	ADDR[1]	Byte	192
14	ADDR[2]	Byte	168
15	ADDR[3]	Byte	2
16	ADDR[4]	Byte	30
17	SubnetMask	IP_V4	
18	ADDR	array [1..4] of Byte	
19	ADDR[1]	Byte	255
20	ADDR[2]	Byte	255
21	ADDR[3]	Byte	255
22	ADDR[4]	Byte	0
23	DefaultRouter	IP_V4	
24	ADDR	array [1..4] of Byte	
25	ADDR[1]	Byte	192
26	ADDR[2]	Byte	168
27	ADDR[3]	Byte	2
28	ADDR[4]	Byte	1



## IP parametreleri ve PROFINET IO cihaz isimleri nasıl değiştirilir

Aşağıdaki örnekte, "addr" ve "nos" (istasyon adı) alt alanlarının her ikisi de değiştirilir. CPU "Properties", " Ethernet address" sayfasında, program yüklendikten sonra "T\_CONFIG" komutunu kullanarak PROFINET cihaz ismini değiştirmenize imkan vermek için "Set PROFINET device name using a different method" radyo düğmesi tıklanmalıdır.

CONF_DATA_2			
	Name	Data type	Start value
1	Static		
2	Conf_data	Struct	
3	header	IF_CONF_Header	
4	FieldType	UInt	0
5	Fieldid	UInt	0
6	SubfieldCount	UInt	2
7	addr	IF_CONF_v4	
8	Id	UInt	30
9	Length	UInt	18
10	Mode	UInt	1
11	InterfaceAddress	IP_V4	
12	ADDR	array [1..4] of Byte	
13	SubnetMask	IP_V4	
14	ADDR	array [1..4] of Byte	
15	DefaultRouter	IP_V4	
16	ADDR	array [1..4] of Byte	
17	nos	IF_CONF_NOS	
18	Id	UInt	40
19	Length	UInt	246
20	Mode	UInt	1
21	NOS	array [1..240] of Byte	



### 10.2.2.10 Komutlar için ortak parametreler

#### REQ giriş parametresi

Açık Kullanıcı Haberleşme komutlarının birçoğu, geçiş üzerinde çalışma başlatmak için REQ girişini kullanır. REQ girişi, komutun bir kere yürütülmesi için yüksek (TRUE) olmalıdır, ama REQ girişi, istediğiniz sürece TRUE kalabilir. Komut, REQ girişinin geçmiş durumunu sıfırlayabilecek şekilde, REQ girişi FALSE ile yürütülene kadar komut başka bir işlem başlatmaz. Sonraki işlemi başlatmak için komutun geçişini tespit edebilmesi için bu gereklidir.

Programınıza bu komutlardan birini yerleştirdiğinizde, STEP 7, kopya DB'yi tanımlamanızı ister. Her komut çağırısı için benzersiz bir DB kullanınız. Bu, her komutun REQ gibi girişleri doğru bir şekilde işlemlerini sağlar.

#### ID giriş parametresi

Bu STEP 7 "Devices and networks" ün "Network view" indeki "Local ID (hex)" için bir referanstır ve bu haberleşme bloğu için kullanmak istediğiniz ağ ID'sidir. ID, yerel bağlantı açıklamasında ilişkili parametre ID'siyle aynı olmalıdır.

## DONE, NDR, ERROR ve STATUS çıkış parametreleri

Bu komutlar tamamlanma durumunu açıklayan çıkışları sağlar:

Çizelge 10- 23 Açık Kullanıcı Haberleşme çıkış komutu parametreleri

Parametre	Veri tipi	Varsayılan	Açıklama
DONE	Bool	FALSE	Son isteğin hatasız tamamlandığını belirtmek amacıyla bir yürütme için TRUE olarak ayarlanır; aksi halde, FALSE olur.
NDR	Bool	FALSE	İstenen eylem hatasız tamamlandığını ve yeni veri alındığını belirtmek amacıyla bir yürütme için TRUE olarak ayarlanır; aksi halde, FALSE olur.
BUSY	Bool	FALSE	Aşağıdakileri belirtmek amacıyla aktif olduğunda TRUE olarak ayarlanır: <ul style="list-style-type: none"><li>• İş henüz tamamlanmadı.</li><li>• Yeni bir iş tetiklenemiyor.</li></ul> İş tamamlandığında FALSE olarak ayarlanır.
ERROR	Bool	FALSE	STATUS' de geçerli hata kodu ile son isteğin hatalar ile tamamlandığını belirtmek amacıyla bir yürütme için TRUE olarak ayarlanır aksi halde, FALSE olur.
STATUS	Word	0	Sonuç durum: <ul style="list-style-type: none"><li>• DONE veya NDR bit ayarlanırsa, STATUS 0 a veya bilgi içerikli bir koda ayarlanır.</li><li>• ERROR bit ayarlanırsa, STATUS bir hata koduna ayarlanır.</li><li>• Yukarıdaki bitlerin hiçbiri ayarlanmazsa, komut, fonksiyonun mevcut durumunu açıklayacak durum sonuçları döndürür.</li></ul> STATUS, fonksiyonun yürütülmesi süresince değerini korur.

### Not

DONE, NDR ve ERROR'un yalnızca bir yürütme için ayarlanacağı unutulmamalıdır.

## Pasif ISO ve TCP haberleşmesi için kısıtlı TSAP'lar ve port numaraları

Pasif bir haberleşme bağlantısı kurmak ve oluşturmak için "TCON" komutunu kullanırsanız, aşağıdaki port adresleri kısıtlanmıştır ve kullanılmamalıdır:

- ISO TSAP (pasif):
  - 01.00, 01.01, 02.00, 02.01, 03.00, 03.01
  - 10.00, 10.01, 11.00, 11.01, ... BF.00, BF.01
- TCP port (pasif): 5001, 102, 123, 20, 21, 25, 34962, 34963, 34964, 80
- UDP port (pasif): 161, 34962, 34963, 34964

### 10.2.3 Programlama cihazıyla haberleşme



Bir CPU bir ağ üzerinde bir STEP 7 programlama cihazı ile iletişim kurabilir.

Bir CPU ve bir programlama cihazı arasındaki iletişimi kurarken aşağıdakileri dikkate alınız:

- Konfigürasyon / Kurulum: Donanım yapılandırma gereklidir.
- Ethernet anahtarı bire-bir haberleşme için gerekli değildir; bir Ethernet anahtarı bir ağda ikiden fazla cihaz için gereklidir.

#### 10.2.3.1 Donanım haberleşme bağlantısının kurulması

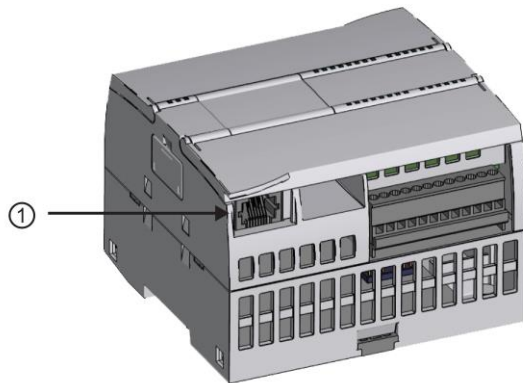
PROFINET ara yüzleri bir programlama cihazı ve CPU arasındaki fiziksel bağlantıları kurar. Auto-Cross-Over işlevselliği CPU içine yerleştirilmiş olduğundan, bir standart veya çapraz Ethernet kablosu ara yüz için kullanılabilir. Bir Ethernet anahtarı bir programlama cihazını bir CPU'ya doğrudan bağlamak için gerekli değildir.

Bir programlama cihazı ve CPU arasındaki donanım bağlantısı oluşturmak için aşağıdaki adımları izleyiniz:

23.CPU monte ediniz (Sayfa 56).

24.Aşağıda gösterilen PROFINET portuna Ethernet kablosunu takınız.

25.Programlama cihazına Ethernet kablosunu bağlayınız.



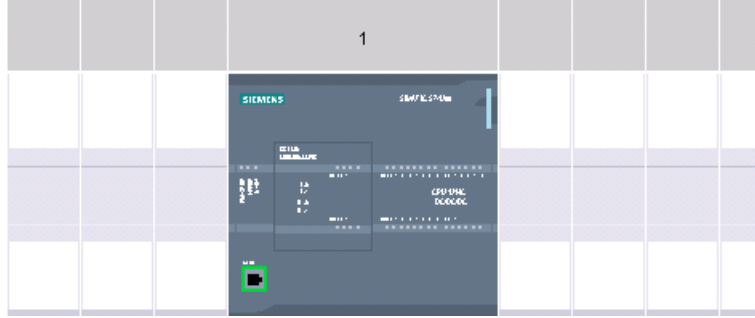
① PROFINET portu

PROFINET bağlantısını güçlendirmek için bir opsiyonel gerilme rahatlatıcı vardır.

### 10.2.3.2 Cihazların yapılandırılması

Bir CPU ile bir proje oluşturduysanız, STEP 7'de projenizi açınız.

Deñilse bir proje oluşturunuz ve şasinin içine bir CPU (Sayfa 138) takınız. Aşağıdaki projede, bir CPU "Cihaz görünümü" de gösterilir.



### 10.2.3.3 İnternet Protokol (IP) adreslerinin atanması

#### IP adreslerinin atanması

Bir PROFINET ağındaki her cihazın aynı zamanda bir İnternet Protokolü (IP) adresine sahip olması gerekir. Bu adres cihazın daha karmaşık, yönlendirilmiş ağdaki veriyi göndermesini sağlar:

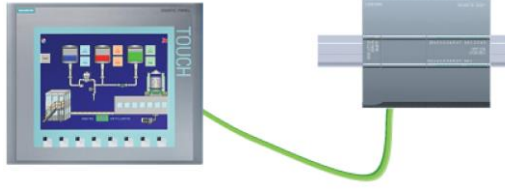
- Tesisinizin LAN'ına bağlı bir yerleşik adaptör kartı ya da yalıtılmış bir ağa bağlı bir Ethernet-USB adaptör kartı kullanan programlama cihazlarına veya diğer ağ cihazlarına sahipseniz, onlara IP adresleri atamanız gerekir. Daha fazla bilgi için (Sayfa 155) "programlama ve ağ cihazlarına IP adreslerinin atanması" konusuna bakınız.
- Ayrıca, çevrimiçi bir CPU veya ağ cihazına bir IP adresi atayabilirsiniz. Bu, özellikle, bir başlangıç cihaz konfigürasyonunda yararlıdır. Daha fazla bilgi için (Sayfa 155) "çevrimiçi bir CPU için bir IP adresi atama" konusuna bakınız.
- Eğer projenizde CPU veya ağ cihazını yapılandırdıysanız, onun IP adresini eklemek için PROFINET ara yüz parametrelerini yapılandırabilirsiniz. Daha fazla bilgi için (Sayfa 158) "projenizde bir CPU için bir IP adresinin yapılandırılması" bölümüne bakınız.

### 10.2.3.4 PROFINET ağınızın test edilmesi

Yapılandırmasını tamamladıktan sonra, projenizi CPU'ya yüklemeniz gerekir. Projeyi yüklediğinizde tüm IP adresleri yapılandırılır.

CPU nun "cihaza yükle" Fonksiyonu ve onun "cihaza genişletilmiş yükleme" diyalogu, tüm erişilebilir ağ cihazlarını ve benzersiz IP adreslerinin tüm cihazlara atanmış olup olmadığını gösterebilir. Daha fazla bilgi için (Sayfa 162) "PROFINET ağının test edilmesi" bölümüne bakınız.

### 10.2.4 HMI-PLC haberleşmesi



CPU, HMI'lere yapılan (Sayfa 32) PROFINET haberleşme bağlantılarını destekler. CPU'lar ve HMI'ler arasındaki haberleşmeyi kurarken aşağıdaki gereksinimler dikkate alınmalıdır:

Yapılandırma/Kurulum:

- CPU'nun PROFINET portu, HMI ile bağlantı yapmak için yapılandırılmış olması gerekir.
- HMI kurulumu yapılmalı ve yapılandırılmalıdır.
- HMI yapılandırma bilgileri, CPU projesinin bir parçasıdır ve yapılandırılabilir ve proje içinde indirebilirsiniz.
- Ethernet anahtarı bire-bir haberleşme için gerekli değildir; bir Ethernet anahtarı bir ağda ikiden fazla cihaz için gereklidir.

#### Not

Şasiye monteli CSM1277 4 portlu Ethernet anahtarı, sizin CPU ve HMI cihazlarınızı bağlamak için kullanılabilir. CPU üzerindeki PROFINET portu, bir Ethernet anahtarlama cihazı içermez.

Desteklenen fonksiyonlar:

- HMI, CPU'ya okuma/yazma verilerini okuyabilir.
- Mesajlar, CPU'dan alınan bilgilere dayalı olarak tetiklenebilir.
- Sistem tanılama

Çizelge 10- 24 HMI ve CPU arasındaki iletişimi yapılandırmak için gerekli adımlar

Adım	Görev
1	Donanım haberleşme bağlantısının kurulması Bir PROFINET ara yüzü, HMI ve CPU arasındaki fiziksel bağlantıyı kurar. Auto-Cross-Over işlevselliği CPU' ya gömüldüğünden, ara yüz için bir standart veya çapraz Ethernet kablosunu kullanabilirsiniz. Bir Ethernet anahtarı bir HMI ile bir CPU yu bağlamak için gerekli değildir. Daha fazla bilgi için (Sayfa 550): " Bir programlama cihazı ile haberleşme: Donanım haberleşme bağlantısının kurulması " bölümüne bakınız.
2	Cihazların yapılandırılması Daha fazla bilgi için (Sayfa 551): " Bir programlama cihazı ile haberleşme: Cihazların yapılandırılması " bölümüne bakınız.
3	Bir HMI ile bir CPU arasındaki mantıksal ağ bağlantılarının yapılandırılması Daha fazla bilgi için "HMI-PLC haberleşmesi: İki cihaz arasındaki mantıksal ağ bağlantılarının yapılandırılması " (Sayfa 553) bölümüne bakılmalıdır.



Adım	Görev
4	Projenizdeki bir IP adresinin yapılandırılması Aynı yapılandırma işlemi kullanınız; Ancak, HMI ve CPU için IP adreslerini yapılandırmanız gerekli. Daha fazla bilgi için (Sayfa 159): " Cihaz yapılandırması: Projenizde bir CPU için bir IP adresi yapılandırma " bölümüne bakınız.
5	PROFINET ağının test edilmesi Her CPU ve HMI cihazı için yapılandırma karşıdan yüklemeniz gerekir. Daha fazla bilgi için (Sayfa 162): " Cihaz yapılandırması: PROFINET ağını test edilmesi " bölümüne bakınız.

#### 10.2.4.1 İki cihaz arasındaki mantıksal ağ bağlantılarının yapılandırılması

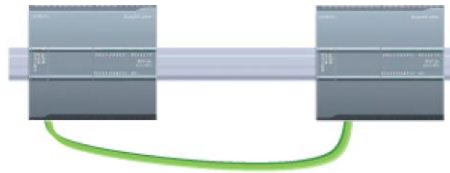
CPU ile rafı yapılandırdıktan sonra, artık ağ bağlantılarınızı yapılandırmak için hazırsınız.

Cihazlar ve Ağlar portalında, projenizde cihazlar arasındaki ağ bağlantıları oluşturmak için "Ağ görünümü" nü kullanınız. Öncelikle, "Bağlantılar" sekmesini tıklayınız ve sonra açılan liste ile bağlantı tipini seçiniz, sadece sağa (örneğin, bir ISO on TCP bağlantısı).

PROFINET bağlantısı oluşturmak için ilk cihazda yeşil (PROFINET) kutuyu tıklayınız ve ikinci cihazda PROFINET kutusuna bir çizgi sürükleyiniz. Farenin düğmesini bırakınız ve PROFINET bağlantınız birleştirilir.

Daha fazla bilgi için (Sayfa 149): "Cihaz yapılandırma: Bir ağ bağlantısı oluşturma " bölümüne bakınız.

#### 10.2.5 PLC-PLC haberleşmesi



Bir CPU, TSEND\_C ve TRCV\_C komutları ile bir ağ üzerindeki başka bir CPU ile iletişim kurabilir.

İki işlemci arasında haberleşme kurarken aşağıdakileri dikkate alınız:

- Yapılandırma / Kurulum: Donanım yapılandırma gereklidir.
- Desteklenen fonksiyonlar: Eşdüzey CPU' ya veri Okuma/Yazma
- Ethernet anahtar bire-bir iletişim için gerekli değildir; bir Ethernet anahtarı bir ağda ikiden fazla cihaz için gereklidir.

Çizelge 10- 25 İki CPU arasında haberleşmeyi yapılandırmak için gerekli adımlar

Basamak	Görev
1	Donanım haberleşme bağlantısının kurulması Bir PROFINET ara yüzü, iki CPU arasındaki fiziksel bağlantıyı kurar. Auto-Cross-Over işlevselliği CPU' ya gömüldüğünden ara yüz için bir standart veya çapraz Ethernet kablosunu kullanabilirsiniz. Bir Ethernet anahtarı iki CPU yu bağlamak için gerekli değildir. Daha fazla bilgi için (Sayfa 268): " Bir programlama cihazı ile haberleşme: Donanım haberleşme bağlantısının kurulması " bölümüne bakınız.
2	Cihazların yapılandırılması Projenizde iki CPU'yu yapılandırmanız gerekir. Daha fazla bilgi için (Sayfa 269): " Bir programlama cihazı ile haberleşme: Cihazların yapılandırılması " bölümüne bakınız.
3	İki CPU arasındaki mantıksal ağ bağlantılarının yapılandırılması Daha fazla bilgi için "PLC-PLC haberleşmesi: İki cihaz arasındaki mantıksal ağ bağlantılarının yapılandırılması " (Sayfa 554) bölümüne bakılmalıdır.
4	Projenizdeki bir IP adresinin yapılandırılması Aynı yapılandırma işlemi kullanınız; Ancak, iki CPU için IP adreslerini yapılandırmanız gerekir (örneğin, PLC_1 ve PLC_2). Daha fazla bilgi için (Sayfa 159): " Cihaz yapılandırması: Projenizde bir CPU için bir IP adresi yapılandırma " bölümüne bakınız.
5	İletim (gönderme) ve alma parametrelerini yapılandırma İki CPU arasında haberleşme sağlamak için TSEND_C ve TRCV_C komutlarını yapılandırmanız gerekir. Daha fazla bilgi için "iki CPU arasında haberleşmenin yapılandırılması: İletim (gönderme) ve alma parametrelerinin yapılandırılması " bölümüne bakınız (Sayfa 555).
6	PROFINET ağının test edilmesi Her CPU için yapılandırmayı karşıdan yüklemeniz gerekir. Daha fazla bilgi için (Sayfa 162): " Cihaz yapılandırması: PROFINET ağının test edilmesi " bölümüne bakınız.

### 10.2.5.1 İki cihaz arasındaki mantıksal ağ bağlantılarının yapılandırılması

CPU ile şasi yapılandırdıktan sonra, artık ağ bağlantılarınızı yapılandırmak için hazırsınız.

Cihazlar ve Ağlar portalında, projenizde cihazlar arasındaki ağ bağlantıları oluşturmak için "Ağ görünümü" nü kullanınız. Öncelikle, "Bağlantılar" sekmesini tıklayınız ve sonra açılan liste ile bağlantı tipini seçiniz, sadece sağa (örneğin, bir ISO on TCP bağlantısı).

PROFINET bağlantısı oluşturmak için ilk cihazda yeşil (PROFINET) kutuyu tıklatınız ve ikinci cihazda PROFINET kutusuna bir çizgi sürükleyiniz. Farenin düğmesini bırakınız ve PROFINET bağlantınız birleştirilir.

Daha fazla bilgi için (Sayfa 149): "Cihaz yapılandırma: Bir ağ bağlantısı oluşturma " bölümüne bakınız.

## 10.2.5.2 İki cihaz arasındaki Yerel / Partner bağlantı yolunu yapılandırma

### Genel parametrelerin yapılandırılması

Haberleşme komutunun "Özellikler" yapılandırma diyalogunda haberleşme parametrelerini belirtiniz. Komutun herhangi bir bölümünü seçtiğiniz zaman bu diyalog sayfanın altına yakın yerde ortaya çıkar.

Daha fazla bilgi için: "Cihaz yapılandırması: Yerel / Partner bağlantı yolunun yapılandırılması" (Sayfa 152) bakınız.

Bağlantı parametreleri diyalogunun "Adres detayları" bölümünde, kullanılacak TSAP'lar veya bağlantı noktalarını tanımlayınız. CPU da bir bağlantı port veya TSAP, "Yerel TSAP" alanına girilir. Partner CPU' nuzda bağlantı için atanan TSAP veya port "Partner TSAP" alanının altına girilir.

## 10.2.5.3 İletim (gönderme) ve alma parametrelerini yapılandırma

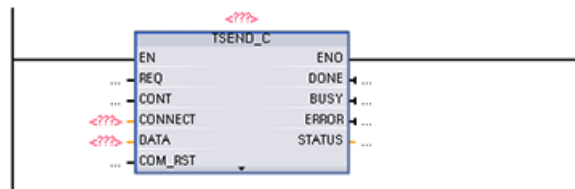
Haberleşme blokları (örneğin, TSEND\_C ve TRCV\_C) iki CPU arasında bağlantı oluşturmak için kullanılır. CPU'lar PROFINET haberleşmesi kurabilmeden önce, iletim (veya gönderme) mesajları ve alma mesajları için parametreleri yapılandırmanız gerekir. Bu parametreler mesajlar hedef cihaza iletileceği ya da bu cihazdan alınacağı zaman haberleşmelerin nasıl çalıştığını belirtir.

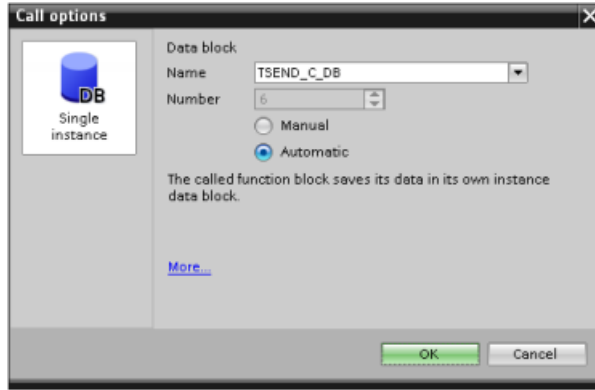
### TSEND\_C komutunun iletim (gönderme) parametrelerini yapılandırma

#### TSEND\_C komutu

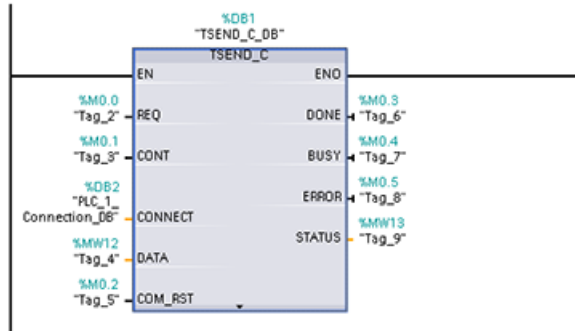
TSEND\_C komutu (Sayfa 520) partner istasyonuyla bir haberleşme bağlantısı oluşturur. Bu komut ile bağlantıyı kesmek için komut verilene kadar bağlantı kurulur, oluşturulur ve otomatik olarak izlenir. TSEND\_C komutu TCON, TDISCON ve TSEND komutlarının fonksiyonlarını birleştirir.

STEP 7'de Cihaz yapılandırmadan, bir TSEND\_C komutunun nasıl veri aktardığını yapılandırabilirsiniz. Başlamak için " Instructions " görev kartında "Communications" klasöründen programa komutu yerleştiriniz. TSEND\_C komutu, komut parametrelerini depolamak için bir DB atadığınız çağrı seçenekleri diyalogu ile birlikte görüntülenir.





Aşağıdaki şekilde gösterildiği gibi, giriş ve çıkışlar için etiket bellek konumlarını atayabilirsiniz:



### Genel parametreleri yapılandırma

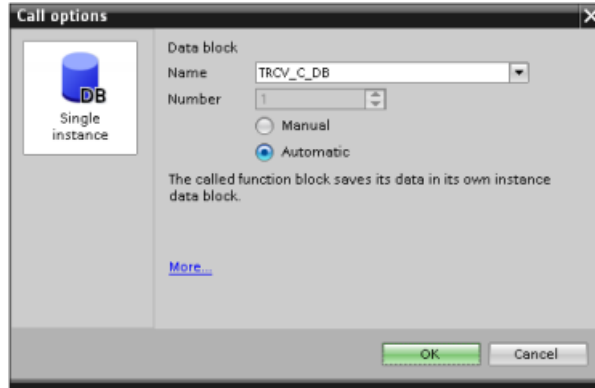
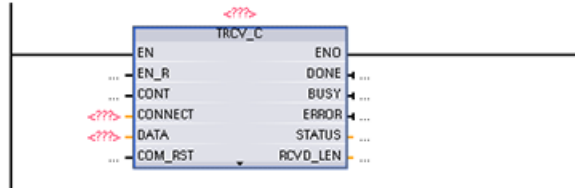
TSEND\_C komutunun "Özellikler" yapılandırma diyalogunda parametreleri belirtiniz. TSEND\_C komutunun herhangi bir bölümünü seçtiğiniz zaman bu diyalog sayfanın altına yakın yerde ortaya çıkar.

### TRCV\_C komutu alma parametrelerini yapılandırma

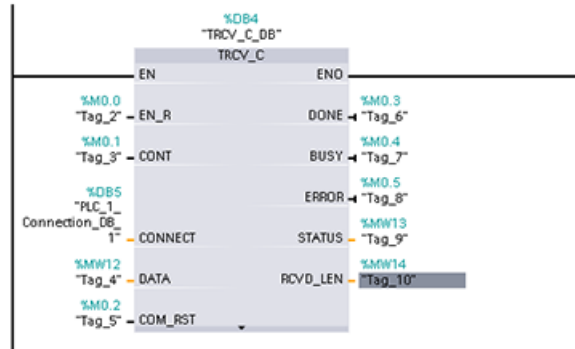
#### TRCV\_C komutu

TRCV\_C komutu (Sayfa 520) partner istasyonu ile bir haberleşme bağlantısı oluşturur. Bu komut ile bağlantıyı kesmek için komut verilene kadar bağlantı kurulur, oluşturulur ve otomatik olarak izlenir. TRCV\_C komutu TCON, TDISCON ve TRCV komutlarının fonksiyonlarını birleştirir.

STEP 7'de Cihaz yapılandırmadan, bir TRCV\_C komutunun nasıl veri aldığını yapılandırabilirsiniz. Başlamak için "Instructions" görev kartında "Communications" klasöründen programa komutu yerleştiriniz. TRCV\_C komutu, komut parametrelerini depolamak için bir DB atadığınız çağrı seçenekleri diyalogu ile birlikte görüntülenir.



Aşağıdaki şekilde gösterildiği gibi, giriş ve çıkışlar için etiket bellek konumlarını atayabilirsiniz:



## Genel parametreleri yapılandırma

TRCV\_C komutunun "Özellikler" yapılandırma diyalogunda parametreleri belirtiniz. TRCV\_C komutunun herhangi bir bölümünü seçtiğiniz zaman bu diyalog sayfanın altına yakın yerde ortaya çıkar.

## 10.2.6 Bir CPU ve PROFINET IO cihazını yapılandırma

### 10.2.6.1 Bir PROFINET IO cihazı ekleme

#### Bir PROFINET IO cihazı ekleme


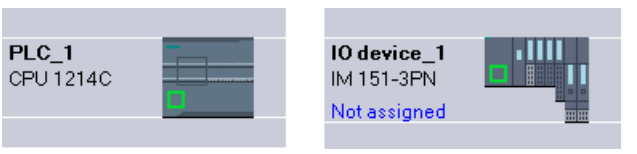
"Cihazlar ve ağlar" portalında, PROFINET IO cihazları eklemek için donanım kataloğunu kullanınız.

#### Not

PROFINET IO cihaz eklemek için STEP 7 Profesyonel veya Temel, V11 veya daha üstü kullanabilirsiniz.

Örneğin bir ET200S IO cihazı: Dağıtılmış I / O, ET200S, ara yüz modülleri ve PROFINET, eklemek için donanım kataloğunda aşağıdaki kutuları genişletin. Daha sonra ET200S cihazlar (parça numarasına göre sıralanır) listesinden ara yüz modülünü seçebilir ve ET200S IO cihazı ekleyebilirsiniz.

Çizelge 10- 26 Cihaz yapılandırmasına bir ET200S IO cihazı ekleme

IO cihazı yerleştir	Sonuç
	

Şimdi CPU PROFINET IO cihazı bağlayabilirsiniz:

26.Cihazda " Not assigned " bağlantısını sağ tıklayınız ve "Select IO controllers " iletişim kutusunu görüntülemek için bağlam menüsünden "Assign new IO controller " seçeneğini seçiniz.

27.Projede IO kontrolörleri listesinden S7-1200 CPU (bu örnekte, "PLC\_1") seçiniz.

28.Ağ bağlantısı oluşturmak için "OK" u tıklayınız.

### 10.2.6.2 CPU ve PROFINET IO cihazı arasındaki mantıksal ağ bağlantılarını yapılandırma

#### Mantıksal ağ bağlantılarını yapılandırma

CPU ile şasiyi yapılandırdıktan sonra, artık ağ bağlantılarınızı yapılandırmak için hazırsınız.

Cihazlar ve Ağlar portalında, projenizde cihazlar arasındaki ağ bağlantıları oluşturmak için "Ağ görünümü" nü kullanınız. PROFINET bağlantısı oluşturmak için ilk cihazda yeşil (PROFINET) kutuyu tıklayınız ve ikinci cihazda PROFINET kutusuna bir çizgi sürükleyiniz. Farenin düğmesini bırakınız ve PROFINET bağlantınız birleştirilir.

Daha fazla bilgi için (Sayfa 149): "Cihaz yapılandırma: Bir ağ bağlantısı oluşturma " bölümüne bakınız.

### 10.2.6.3 CPU'lar ve cihaz isimlerini atama

#### CPU'lar ve cihaz isimlerini atama

Cihazlar arasındaki ağ bağlantıları, CPU' ya PROFINET IO cihazını atar ki bu CPU'nun cihazı kontrol etmesi için gereklidir. Bu atamayı değiştirmek için PROFINET IO cihaz üzerinde gösterilen PLC adını tıklatınız. Bir iletişim kutusu, PROFINET IO cihazının mevcut CPU' dan bağlantısının kesilmesine ve istenirse, yeniden atanmasına veya atanmadan bırakılmasına izin verir.

CPU' ya bağlanmadan önce PROFINET ağındaki cihazların atanmış bir adı olmalıdır. cihazlara henüz bir isim atanmış değil ise veya cihazın ismi değiştirilecek ise PROFINET cihazlara isimleri atamak için "Ağ görünümünü " kullanınız . PROFINET IO cihazını sağ tıklayınız ve bunu yapmak için "cihaz ismi ata " seçeneğini seçiniz.

Her PROFINET IO cihazı için, her iki STEP 7 projesindeki cihaza ve PROFINET IO cihaz yapılandırma hafızasına "Çevrimiçi & diagnostics" aracını ( örneğin, bir ET200 S ara yüz modülü konfigürasyon belleği) kullanarak aynı adı atamanız gerekir. Bir isim eksikse veya her iki yerde de eşleşmiyorsa, PROFINET veri alışveriş modu çalışmayacaktır. Daha fazla bilgi için "Çevrimiçi ve tanılama araçları: Bir PROFINET cihazına çevrimiçi (Sayfa 787 ) bir isim atama".

### 10.2.6.4 İnternet Protokol (IP) adresleri atama

#### IP adresleri atama

Bir PROFINET ağdaki her cihazın aynı zamanda bir İnternet Protokolü (IP) adresine sahip olması gerekir. Bu adres cihazın, daha karmaşık, yönlendirilmiş ağdaki veriyi göndermesini sağlar:

- Tesisinizin LAN'ına bağlı bir yerleşik adaptör kartı ya da yalıtılmış bir ağa bağlı bir Ethernet-USB adaptör kartı kullanan programlama cihazlarına veya diğer ağ cihazlarına sahipseniz, onlara IP adresleri atamanız gerekir. Daha fazla bilgi için (Sayfa 155) "programlama ve ağ cihazlarına IP adreslerinin atanması" konusuna bakınız.
- Ayrıca, çevrimiçi bir CPU veya ağ cihazına bir IP adresi atayabilirsiniz. Bu, özellikle, bir başlangıç cihaz konfigürasyonunda yararlıdır. Daha fazla bilgi için (Sayfa 158) "çevrimiçi bir CPU için bir IP adresi atama" konusuna bakınız.
- Eğer projenizde CPU veya ağ cihazını yapılandırdıysanız, onun IP adresini eklemek için, PROFINET ara yüz parametrelerini yapılandırabilirsiniz. Daha fazla bilgi için (Sayfa 159) "projenizde bir CPU için bir IP adresinin yapılandırılması" bölümüne bakınız.

### 10.2.6.5 IO çevrim süresinin yapılandırılması

#### IO çevrim süresinin yapılandırılması

Bir PROFINET IO cihazına bir "IO çevrim" süresi içinde CPU' dan yeni veri sağlanır. Güncelleme süresi ayrı ayrı her cihaz için yapılandırılabilir ve verinin CPU' dan ve cihazdan iletiildiği zaman aralığını belirleyebilir.

Değiştirilecek veri miktarını ve bu kontrolöre atanan cihazların sayısını hesaba katarak STEP 7, PROFINET ağdaki her bir cihaz için varsayılan ayarda "IO çevrimi" güncelleme süresini otomatik olarak hesaplar. Güncelleme zamanının otomatik olarak hesaplanmasını istemiyorsanız, bu ayarı değiştirebilirsiniz.

PROFINET IO cihazının "Özellikler" yapılandırma diyalogunda "IO çevrim" parametrelerini belirtirsiniz. Komutun herhangi bir bölümünü her ne zaman seçerseniz bu diyalog sayfanın altına yakın görünür.

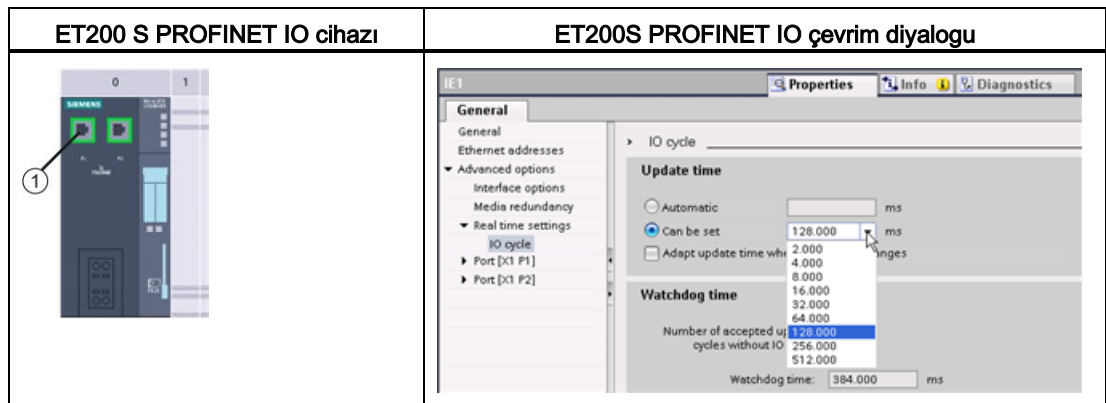
PROFINET IO cihazının "Device view" de, PROFINET portuna tıklayınız. "PROFINET Interface" diyalogunda, aşağıdaki menü seçimleri ile "IO çevrim" parametrelerine erişirsiniz:

- "Gelişmiş seçenekler"
- "Gerçek zaman ayarları"
- "IO çevrim"

IO çevrimi "Güncelle zamanı" nı aşağıdaki seçimler ile tanımlayınız:

- Otomatik olarak hesaplanan uygun bir güncelleme zamanına sahip olmak için "Automatic" seçeneğini seçiniz.
- Güncellemeyi kendiniz ayarlamak istiyorsanız " Can be set " seçeneğini seçiniz ve ms cinsinden gerekli güncelleme zamanını giriniz.
- Gönderme saati ve güncelleme zamanı arasında tutarlılığı sağlamak için "Adapt update time when send clock changes" seçeneğini etkinleştiriniz. Bu seçenek, güncelleştirme zamanının gönderme saatinden daha az ayarlanmış olmadığını sağlar.

Çizelge 10- 27 ET200S PROFINET IO çevrim süresinin yapılandırılması



① PROFINET port

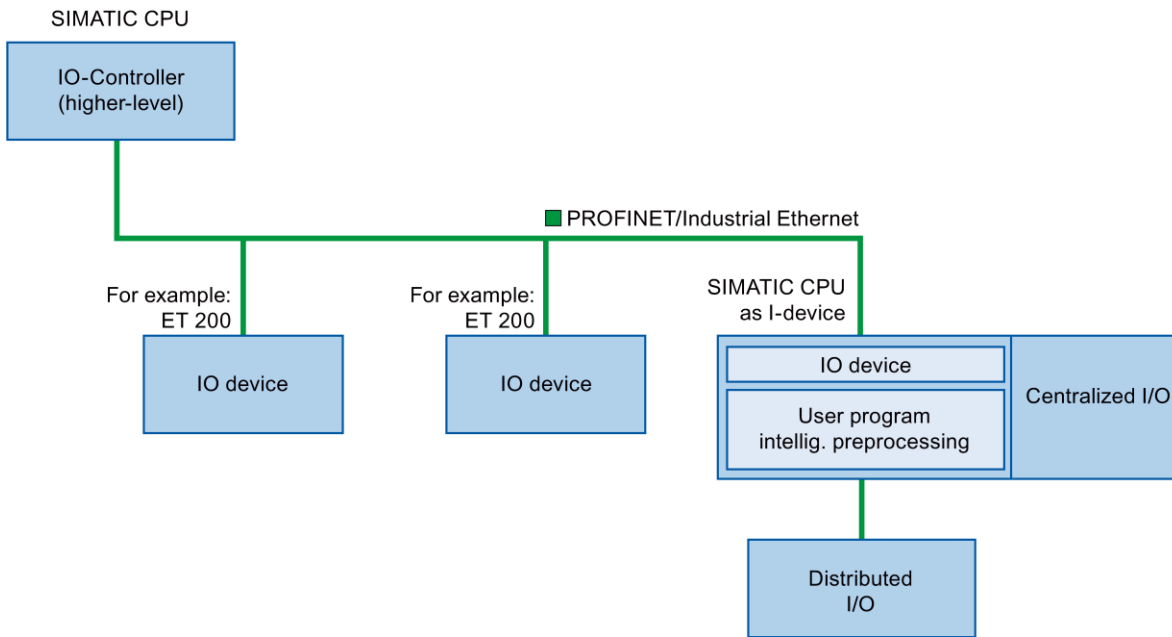


## 10.2.7 Bir CPU ve PROFINET I-Cihazın yapılandırılması

### 10.2.7.1 I-cihaz işlevselliği

Bir CPU'nun "I-cihaz" (akıllı IO cihazı) işlevi, örneğin alt süreçlerin akıllı ön işlem birimi olarak CPU'nun çalışmasıyla ve bir IO kontrolörüyle veri alışverişini kolaylaştırır. I-cihaz "üst seviye" IO kontrolörüne bir IO cihazı olarak bağlantılıdır.

Ön-işleme, kullanıcı programı tarafından CPU üzerinde işlenir. Merkezi veya (PROFINET IO veya PROFIBUS DP) I / O da elde edilen proses değerler, kullanıcı programı tarafından ön işleme tabi tutulur ve üst-seviye istasyonun CPU'su için bir PROFINET IO ara yüz aracılığıyla kullanılabilir hale gelir.



### "I-cihaz" isimlendirme kuralı

Bu açıklamanın geri kalanında, bir CPU veya I-cihaz işlevselliği olan bir CP basitçe bir "I-cihaz" olarak adlandırılır.

### 10.2.7.2 I-cihazın özellikleri ve avantajları

#### Uygulama alanları

I-cihazının uygulama alanları:

- Dağıtık proses:

Karmaşık bir otomasyon görevi daha küçük birimlere / alt proseslere ayrılabilir. Bu basitleştirilmiş alt görevlere yol açan yönetilebilir proseslerle sonuçlanır.

- Alt proseslere ayırma:

Karmaşık, yaygın olarak dağıtılan ve kapsamlı prosesler, I-cihazları kullanarak yönetilebilir ara yüzleri ile birkaç alt prosese ayrılabilir. Bu alt prosesler, sonra bir master proje oluşturmak için birleştirilebilen bireysel STEP 7 projelerde gerekirse saklanabilir.

- Know-how koruması:

Bileşenler, bir STEP 7 projesi yerine I-cihaz ara yüz açıklaması için sadece bir GSD dosyası ile teslim edilebilir. Artık yayınlanması gerekmediğinden kullanıcı kendi programını koruyabilir.

#### Özellikler

I-cihazın özellikleri:

- STEP 7 proje bağlantısını kesmek:

I-cihazın yaratıcıları ve kullanıcıları tamamen ayrı STEP 7 otomasyon projelerine sahip olabilir. GSD dosyası STEP 7 projeler arasında ara yüz oluşturur. Bu standart bir ara yüz üzerinden standart IO kontrolörleri için bir bağlantı sağlar.

- Gerçek zamanlı haberleşme:

I-cihaz, PROFINET IO ara yüzü üzerinden bir belirleyici PROFINET IO sistemi ile sağlar ve bu nedenle RT (gerçek-zamanlı haberleşme) ve IRT'yi (izokron gerçek zaman) destekler.

#### Avantajları

I-cihaz aşağıdaki avantajlara sahiptir:

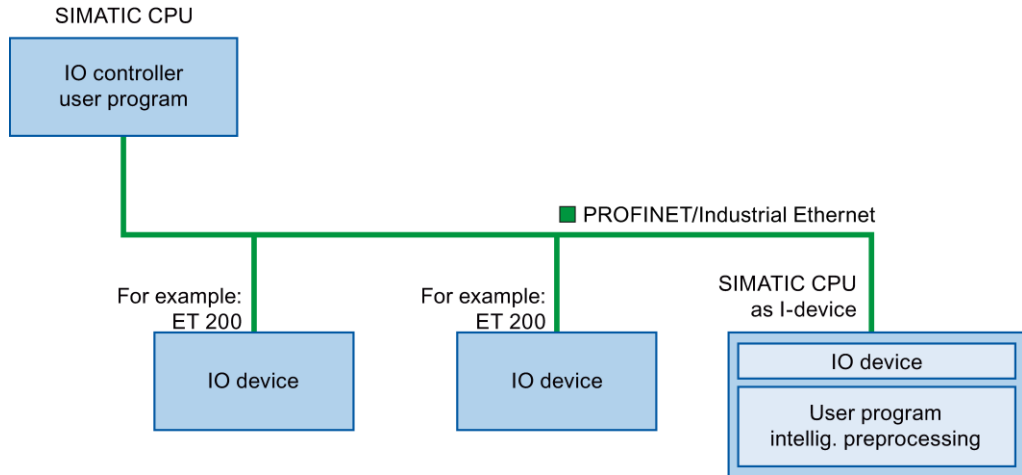
- IO kontrolörlerinin basit bağlantısı
- IO kontrolörleri arasındaki gerçek zamanlı haberleşme
- I-cihazlara işlem kapasitesini dağıtarak IO kontrolörlerini rahatlatma.
- Yerel olarak proses verilerini işleyerek düşük haberleşme yükü.
- Ayrı STEP 7 projelerde alt görevlerin işlenmesi nedeniyle yönetilebilir

### 10.2.7.3 Bir I-cihazın karakteristiği

I-cihaz standart IO cihaz gibi bir IO sistemine dahil edilir.

### Alt düzey PROFINET IO sistemi olmadan I-cihaz

I-cihazın kendi dağıtılmış I/O 'u yoktur. Bir IO cihazın rolünde I-cihazların yapılandırma ve parametre ataması, dağıtılmış I/O sistemi (örneğin, ET 200) ile aynıdır.



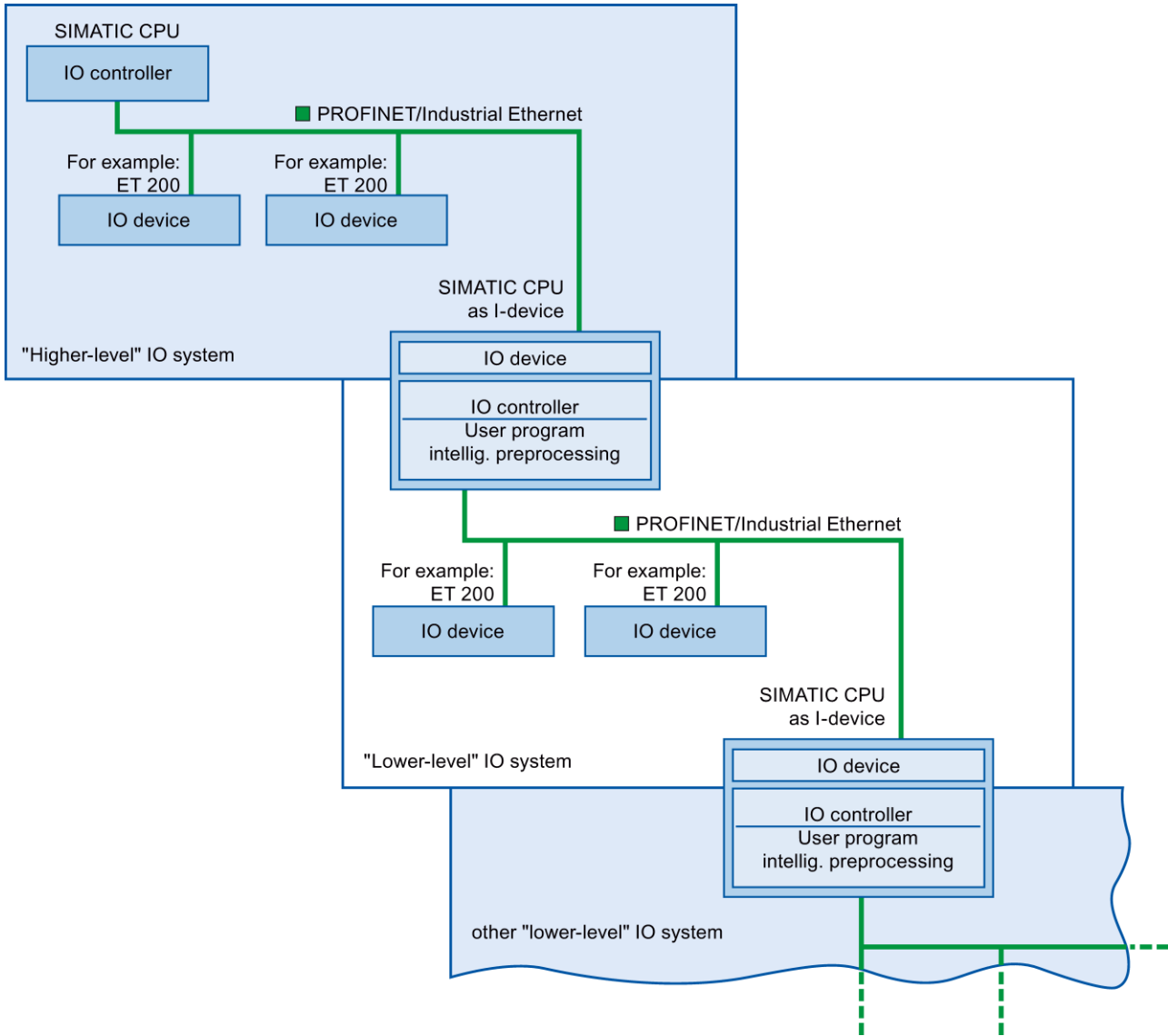
### Alt seviye PROFINET IO sistemine sahip I-cihaz

Yapılandırmasına bağlı olarak, bir I-Cihaz, bir IO cihaz rolüne sahip olmasının yanında bir PROFINET ara yüzü üzerinde bir IO kontrolörü de olabilir.

Bu, I-Cihazın, PROFINET ara yüzü üzerinden bir üst seviye IO sisteminin parçası olabileceği ve IO kontrolör olarak, kendi alt seviye IO sistemini destekleyebileceği anlamına gelir.

Alt seviye IO sistemi, sırayla, I-cihazlar (aşağıdaki şekle bakınız) içerebilir. Bu, hiyerarşik olarak yapılandırılmış IO sistemlerini mümkün kılar.

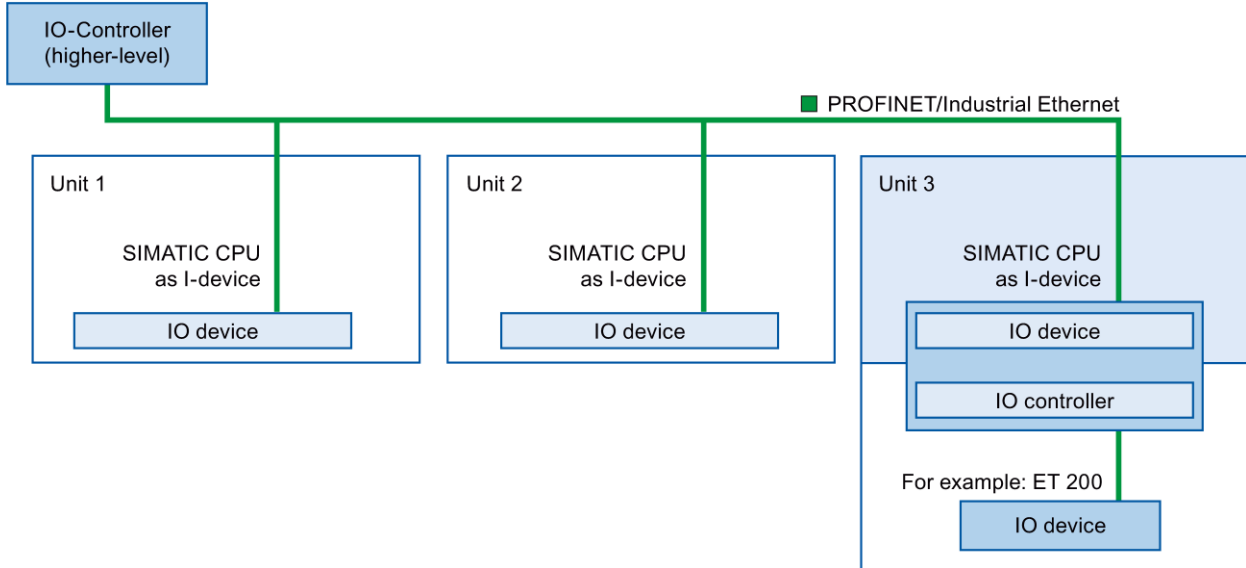
IO kontrolörü olarak rolüne ek olarak, bir I-cihaz aynı zamanda alt seviye PROFIBUS sistemi için DP-Master olarak bir PROFIBUS ara yüzü aracılığıyla kullanılabilir.



### Örnek - IO cihazı ve IO kontrolörü olarak I-cihaz

IO cihazı ve IO kontrolörü olarak I-cihaz, bir yazdırma prosesi örneğine dayalı olarak anlatılmıştır. I-cihaz, bir birimi (bir alt prosesi) kontrol eder. Bir birim, örneğin basılı malzeme paketi içinde el ilanları veya broşür gibi ek dokümanlar eklemek için kullanılır.

For example: S7mEC



Ünite 1 ve ünite 2' nin her biri merkezi I / O ya sahip bir I-cihazından oluşur. Dağıtılmış I / O sistemi ile birlikte I-cihaz (örneğin, ET 200) ünite 3'ü oluşturur.

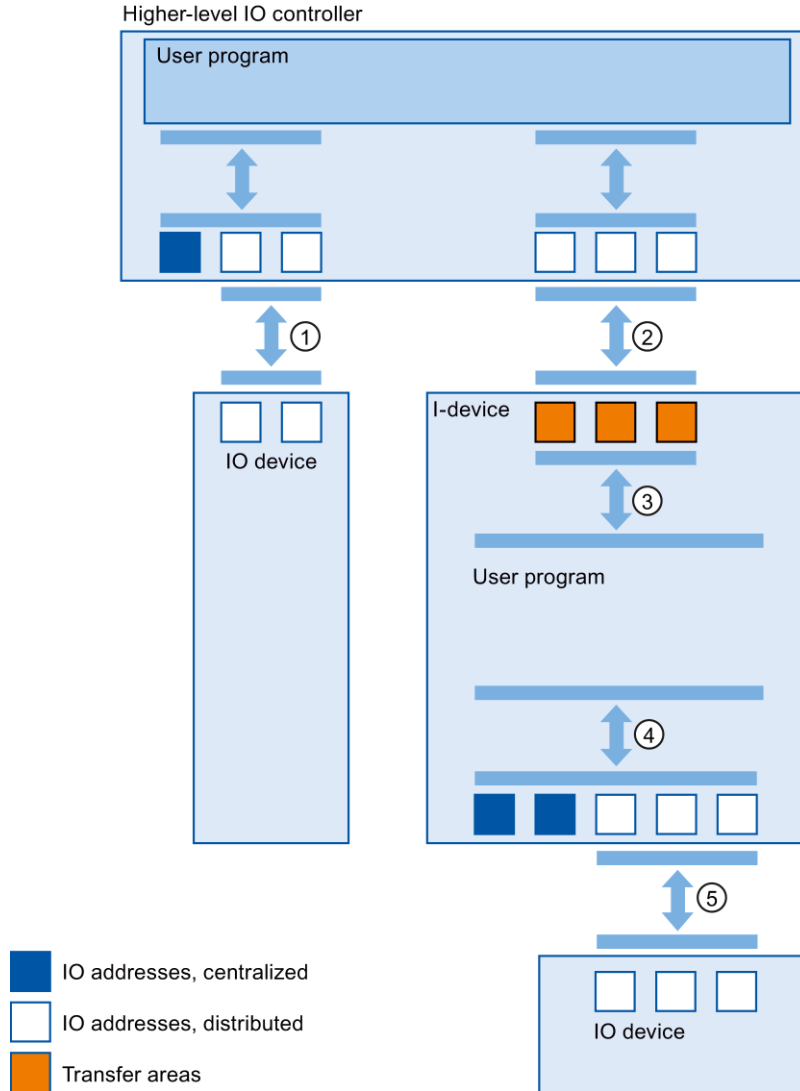
I-cihazda kullanıcı programı, proses verilerinin ön işlemeden sorumludur. Bu görev için, I-cihazın kullanıcı programı, üst seviye IO kontrolöründen gelen varsayılan ayarlara (örneğin, kontrol verileri) ihtiyaç duyar. I-cihaz, sonuçlar ile birlikte yüksek seviye IO kontrolörü (örneğin kendi alt görevin durumu) sağlamaktadır.

#### 10.2.7.4 Daha yüksek ve daha düşük seviyeli IO sistemi arasında veri değişimi

Transfer alanlar, I-cihaz CPU'su kullanıcı programı için bir ara yüzdür. Girişler kullanıcı programında işlenir ve çıkışlar kullanıcı programında prosesin sonucudur.

IO kontrolörü ve I-cihaz arasında haberleşme verisi, transfer alanlarında kullanılabilir. Bir transfer alanı IO kontrolörü ve I-cihaz arasında sürekli değiştirilen bir bilgi birimi içerir. "I-cihaz yapılandırması" bölümündeki transfer alanların kullanımı ve yapılandırma hakkında daha fazla bilgi bulabilirsiniz.

Sonraki şekil yüksek ve düşük seviyeli IO sistemi arasındaki veri alışverişini gösterir. Bireysel haberleşme ilişkileri numaralara dayalı olarak aşağıda açıklanmıştır:



- ① **Üst seviye IO kontrolörü ve normal IO - cihaz arasında veri alışverişi**  
Bu şekilde, PROFINET yoluyla IO kontrolörü ve IO cihazları arasında veri değişimi.
- ② **Üst seviye IO kontrolörü ve IO - cihaz arasında veri alışverişi**  
Bu şekilde, PROFINET yoluyla IO kontrolörü ve IO cihazları arasında veri değişimi.  
Bir üst seviye IO kontrolörü ve bir I - cihaz arasında veri değişimi geleneksel IO kontrolörü / IO cihaz ilişkisine dayanmaktadır.  
Üst seviye IO kontrolörü için, I- cihazların transfer alanları önceden yapılandırılmış istasyonun alt modüllerini temsil eder.  
IO kontrolörü çıkış verileri, I-cihazının giriş verisidir. Benzer şekilde, IO kontrolör giriş verileri I-cihazın çıkış verileridir.
- ③ **Kullanıcı programı ve transfer alanı arasında transferi ilişkisi**  
Bu şekilde, kullanıcı programı ve transfer alanı giriş ve çıkış veri alışverişi yapar.
- ④ **Kullanıcı programı ve I-cihazının I / O su arasındaki veri alışverişi**  
Bu şekilde kullanıcı programı ve merkezi / dağıtılmış I / O su giriş ve çıkış veri alışverişi yapar.
- ⑤ **Alt seviye IO kontrolörü ve IO - cihaz arasında veri alışverişi**  
Bu şekilde, PROFINET yoluyla I-cihaz ve IO cihazları arasında veri alışverişi olur.

### 10.2.7.5 I-cihaz yapılandırması

Yapılandırma için temelde iki olasılık vardır:

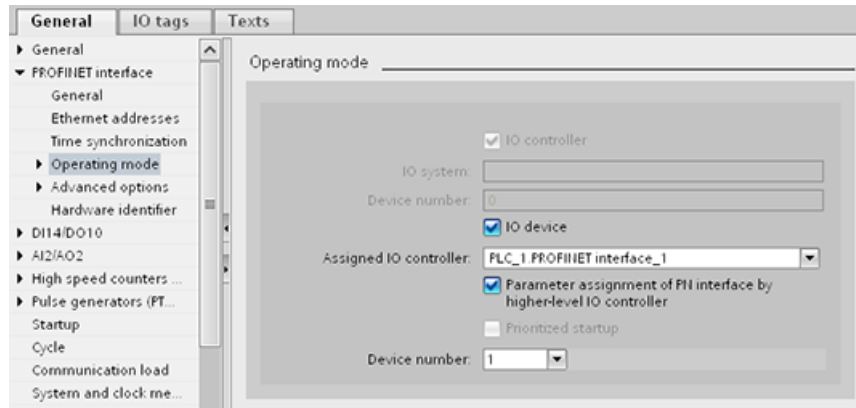
- Bir proje içinde bir I-cihaz yapılandırması
- Başka bir projede veya başka bir mühendislik sisteminde kullanılan bir I-cihaz yapılandırması.

STEP 7, GSD dosyasına yapılandırılmış I-cihaz aktarımı yaparak başka bir proje için veya başka bir mühendislik sistemi için I-cihazını yapılandırmanıza izin verir. Diğer GSD dosyaları gibi diğer projeler veya mühendislik sistemlerinde GSD dosyasını alırsınız. Diğer veriler arasında veri alışverişi için transfer alanları bu GSD dosyasında saklanır.

#### Bir proje içinde bir I-cihaz yapılandırması

29. Bir PROFINET CPU'yu donanım kataloğundan ağ görünümüne sürükleyiniz ve bırakınız.
30. Bir IO cihazı olarak yapılandırılabilir bir PROFINET CPU yu da donanım kataloğundan ağ görünümüne sürükleyiniz ve bırak. Bu cihaz (örneğin, CPU 1215C), bir I-cihaz olarak yapılandırılır.
31. I-cihaz için PROFINET ara yüzünü seçiniz.
32. Alan gezintisi denetim penceresinde "Çalışma modu" nu ve sonra " IO cihazı" onay kutusu seçeneğini seçiniz.
33. Şimdi " Atanmış IO kontrolörü" açılır listesinde IO kontrolörü seçme seçeneği vardır.

IO kontrolörünü bir kere seçtiğinizde her iki cihaz arasındaki, ağ ve IO sistemi ağ görünümünde görüntülenir.

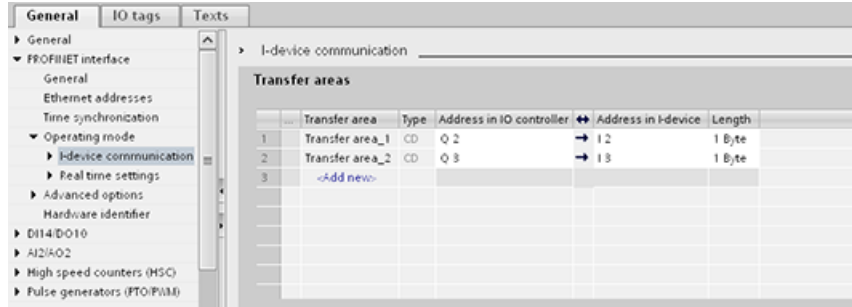


34. "Üst seviye IO kontrolörü tarafından PN ara yüzünün Parametre atanması" Onay kutusu ile ara yüz parametrelerinin I-cihazın kendisi tarafından ya da daha yüksek seviyeli IO kontrolörü tarafından atanıp atanmayacağını belirtiniz.

Bir alt seviye IO sistemi ile I-cihazını işletirseniz, daha sonra I-Cihaz PROFINET ara yüzü parametreleri (örneğin, port parametresi) üst seviyeli IO denetçisiyle atanamaz.

35. Transfer alanlarını yapılandırınız. Transfer alanları "I-cihaz haberleşmesi" alan gezinti bölümünde bulunur:

- "Transfer alanı" sütununun ilk alanına tıklayınız. STEP 7, değiştirebileceğiniz varsayılan bir isim atar.
- Haberleşme ilişki tipini seçiniz: Hali hazırda sadece CD veya F-CD' yi seçebilirsiniz.
- Adresler otomatik olarak ayarlanmıştır; Eğer gerekirse adresleri düzeltebilir ve sürekli transfer edilecek transfer alanının uzunluğunu belirleyebilirsiniz.



36. Her transfer alanı için gezinti alanında ayrı bir giriş oluşturulur. Bu girişlerden birini seçerseniz, transfer alanının detaylarını ayarlayabilir ya da bunları düzeltebilir ve onlara yorum yapabilirsiniz.

## GSD dosyası ile bir I-cihaz yapılandırma

Başka bir projede bir I-cihaz kullanırsanız veya I-cihaz başka bir mühendislik sisteminde kullanılırsa, o zaman üst seviye IO kontrolörünü ve yukarıda açıklandığı gibi I-cihazını yapılandırınız.

Ancak, aktarım alanlarını yapılandırdıktan sonra "Export" düğmesine tıklayınız böylece I-cihazdan yeni bir GSD dosyası oluşturulur. Bu GSD dosyası diğer projelerde yapılandırılmış I-cihazını temsil eder.

"Export" düğmesi, Denetim penceresinin "I-cihaz haberleşmesi" bölümünde bulunur.

Donanım yapılandırması derlenir ve "dışa aktar" iletişim penceresi açılır.

I-cihaz proksi için bir ad ve bunun yanı sıra sağlanan alanlarda bir açıklama atayınız. İşleminizi tamamlamak için "Export" düğmesine tıklayınız.

Son olarak, örneğin, başka bir projede, GSD dosyasını import ediniz.

## 10.2.8 Tanılama (diyagnostik)

Bu haberleşme ağları ile tanılama için organizasyon bloklarının (OBS) nasıl kullanılacağı hakkında bilgi için "Organizasyon blokları (OB'ler)" (Sayfa 85) bölümüne bakınız.

## 10.2.9 Dağıtılmış I/O komutları

Bu haberleşme ağları ile dağıtılmış I/O komutlarının nasıl kullanılacağı hakkında bilgi için (Sayfa 314) "Dağıtılmış I/O (PROFINET, PROFIBUS, ya da AS-i)" bölümüne bakınız.



### 10.2.10 Tanılama (diyagnostik) komutları

Bu iletişim ağları ile bu komutların nasıl kullanılacağı hakkında bilgi için "Tanılama (PROFINET veya PROFIBUS)": "Tanılama komutları" (Sayfa 345) bölümüne bakınız.

### 10.2.11 Dağıtılmış I/O için tanılama olayları

Bu haberleşme ağları ile bu tanılama bilgilerinin nasıl kullanılacağı hakkında bilgi için "Tanılama (PROFINET veya PROFIBUS)": "Dağıtılmış I / O için Tanılama olayları" (Sayfa 345) bölümüne bakınız.

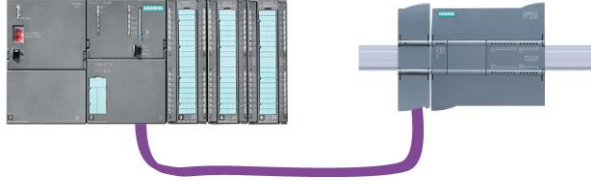
## 10.3 PROFIBUS

Bir PROFIBUS sistemi, RS485 seri veri yolu üzerindeki bir multi-drop modda dağıtılan slave cihazlarını yoklamak için bir veri yolu master'ı kullanır. Bir PROFIBUS slave, bilgileri işleyen ve master'a çıkış gönderen herhangi bir çevresel cihazdır (I/O transdüser, valf, motor sürücü, ya da diğer ölçme cihazları). Slave, veri yolu erişim haklarına sahip olmadığı için ağ üzerinde pasif bir istasyon oluşturur ve sadece alınan mesajları kabul edebilir veya talep üzerine master'a tepki mesajları gönderebilir. Tüm PROFIBUS slave'ler aynı önceliğe sahiptir ve tüm ağ haberleşmesi master'dan kaynaklanmaktadır.

Bir PROFIBUS-Master, ağ üzerinde bir "aktif istasyon" oluşturur. PROFIBUS DP, iki master sınıfı tanımlar. Sınıf 1 master ( normal olarak bir merkezi programlanabilir kontrolör (PLC) veya özel bir yazılım çalıştıran bir PC), normal haberleşmeyi ya da kendisine atanan slave ile veri alışverişini yönetir. Sınıf 2 master (Örneğin genellikle devreye alma, bakım veya tanılama amaçlı kullanılan bir dizüstü bilgisayar veya programlama konsolu gibi yapılandırma cihazı) öncelikle slave'leri devreye alma ve tanılama amaçlı kullanılan özel bir cihazdır.

S7 -1200CM, 1242-5 haberleşme modülü ile bir DP slave olarak bir PROFIBUS ağına bağlanır. CM 1242-5 ( DP slave ) modülü, DP V0/V1 master'ların haberleşme partneri olabilir. Bir üçüncü - parti sistemindeki modülü yapılandırmak istiyorsanız, modül ile gönderilen CD'de ve Internet üzerinde Siemens Otomasyon Müşteri Destek sayfalarında (<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=6GK72425DX300XE0&caller=view>) CM 1242-5 ( DP slave ) için kullanılabilir bir GSD dosyası vardır.

Aşağıdaki şekilde, S7-1200, S7-300 kontrolör için bir DP slave'dir:



S7-1200, CM 1243-5 haberleşme modülü ile bir DP-Master olarak bir PROFIBUS ağına bağlanır. CM 1243-5 (DP master) modülü DP V0/V1 slave'lerin haberleşme partneri olabilir. Aşağıdaki şekilde, S7-1200 ET200S DP slave'i kontrol eden master'dır:



CM 1242-5 ve CM 1243-5 birlikte kurulu ise, bir S7-1200 aynı anda, hem üst seviye bir DP master sisteminin bir slave'i hem de bir alt seviye DP slave sisteminin bir Master'ı olarak çalışabilir:



V4.0 için, DP master veya DP slave CM'lerin herhangi bir kombinasyonunun olabildiği, istasyon başına maksimum üç PROFIBUS CMS yapılandırabilirsiniz. V3.0 veya daha üst CPU firmware uygulanmasında DP master'ların her biri maksimum 32 slave kontrol edebilir.

PROFIBUS CM'lerin yapılandırma verileri, yerel CPU üzerinde saklanır. Bu, gerektiğinde bu haberleşme modüllerinin basitçe yedeğiyle değiştirilmesini sağlar.

S7-1200 V4.0 CPU'lar ile PROFIBUS'u kullanmak için PROFIBUS Master CM firmware'ini V1.3 e yükseltmeniz gerekir.

Dijital olarak güvenli (SD) bir kart kullanılarak bu yükseltmeyi yapabilirsiniz.

#### Not

PROFIBUS CM firmware'i, her zaman Siemens Servis ve Destek web sitesinde (<http://support.automation.siemens.com/WW/view/en/42131407> update) bulunan en son sürüme güncelleniz tavsiye edilir.

### 10.3.1 PROFIBUS CM'lerin haberleşme hizmetleri

PROFIBUS CM'ler PROFIBUS DP-V1 protokolü kullanır.

#### DP-V1 ile haberleşme tipleri

Aşağıdaki haberleşme tipleri DP-V1 ile kullanılabilir:

- Döngüsel haberleşme (CM 1242-5 ve 1243-5 CM)

Her iki PROFIBUS modülü, DP Slave ve DP-Master arasında proses veri transferi için döngüsel haberleşmeyi destekliyor.

Döngüsel haberleşme, CPU işletim sistemi tarafından işlenir. Bunun için hiçbir yazılım bloklarına gerek yoktur. I/O verisi, doğrudan CPU proses görüntüsünden okunur veya buradan yazılır.

- Döngüsel olmayan haberleşme (sadece CM 1243-5)

DP master modülü, yazılım blokları kullanılarak, döngüsel olmayan haberleşmeyi destekler:

- "RALRM" komutu kesme işleme için kullanılabilir.
- "RDREC" ve "WRREC" komutları, yapılandırma ve tanılama verilerinin aktarımı için kullanılabilir.

İşlevler, CM 1243-5: SYNC / FREEZE ve Get\_Master\_Diag tarafından desteklenmez.

#### CM 1243-5 in diğer haberleşme hizmetleri

CM 1243-5 DP master modülü aşağıdaki ek haberleşme hizmetlerini destekler:

- S7 haberleşme

- PUT/GET hizmetler

DP master, PROFIBUS üzerinden diğer S7 kontrolörlerinden veya bilgisayarlardan sorgu yapmak için bir istemci ve sunucu olarak çalışır.

- PG/OP haberleşme

PG fonksiyonları, PG den yapılandırma verilerinin ve kullanıcı programlarının indirilmesini ve bir PG için tanılama verilerinin aktarımını sağlar.

OP haberleşme için olası haberleşme Partnerleri HMI paneller, S7 iletişimini destekleyen esnek WinCC' li SIMATIC panel PC'ler veya SCADA sistemleridir.

## 10.3.2 PROFIBUS CM kullanıcı kılavuzlarına referans

### Daha fazla bilgi

Cihazlar için kılavuzlarda PROFIBUS CM'ler hakkında detaylı bilgi bulabilirsiniz. Aşağıdaki giriş ID'ler altında Siemens Endüstriyel Otomasyon Müşteri Destek sayfalarında internette bunları bulabilirsiniz:

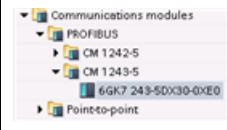
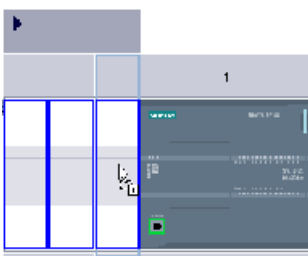
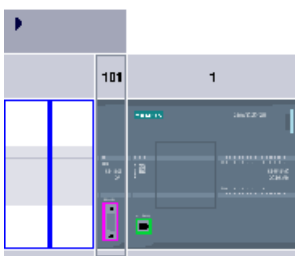
- CM 1242-5 (<http://support.automation.siemens.com/WW/view/en/49852105>)
- CM 1243-5 (<http://support.automation.siemens.com/WW/view/en/49851842>)

## 10.3.3 DP master ve slave cihaz yapılandırma

### 10.3.3.1 CM 1243-5 (DP master) modülü ve bir DP slave ekleme

"Cihazlar ve ağlar" portalda, PROFIBUS modüllerini CPU'ya eklemek için donanım kataloğu kullanınız. Bu modüller CPU'nun sol tarafına bağlanır. Donanım yapılandırması için bir modül eklemek için donanım kataloğunda modülü seçiniz ve çift tıklayınız ya da vurgulanan slota modülü sürükleyiniz.

Çizelge 10- 28 Cihaz Yapılandırması için bir PROFIBUS CM 1243-5 (DP master) modülü ekleme

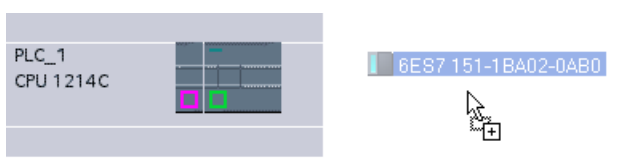
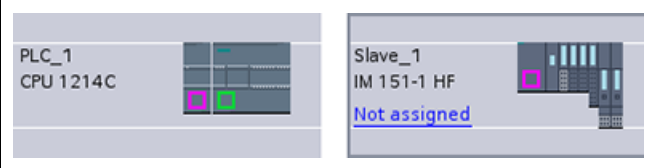
Modül	Modülü seç	Modül ekle	Sonuç
CM 1243-5 (DP master)			

Yanı sıra DP köle eklemek için donanım katalog kullanınız. Örneğin, bir ET200 S DP slave eklemek için, donanım kataloğunda, aşağıdaki kapları genişletiniz:

- Dağıtılmış I/O
- ET200 S
- Ara yüz modülleri
- PROFIBUS

Sonra, parça numaraları listesinden "6ES7 151-1BA02-0AB0" (IM151-1 HF) seçiniz ve aşağıda gösterildiği gibi ET200 S DP slave ekleyiniz.

Çizelge 10- 29 Cihaz Yapılandırması için ET200 S DP slave ekleme

DP slave ekle	Sonuç
	

### 10.3.3.2 İki PROFIBUS cihazı arasındaki mantıksal ağ bağlantılarının yapılandırılması

CM 1243-5 (DP master) modülünü yapılandırdıktan sonra, artık ağ bağlantılarınızı yapılandırmak için hazırsınız.

Cihazlar ve Ağlar portalında, projenizde cihazlar arasındaki ağ bağlantıları oluşturmak için "Ağ görünümü" nı kullanınız. PROFIBUS bağlantısı oluşturmak için, ilk cihazda mor (PROFIBUS) kutuyu seçiniz ve ikinci cihazda PROFIBUS kutusuna bir çizgi sürükleyiniz. Farenin düğmesini bırakın ve PROFIBUS bağlantınız birleştirilir.

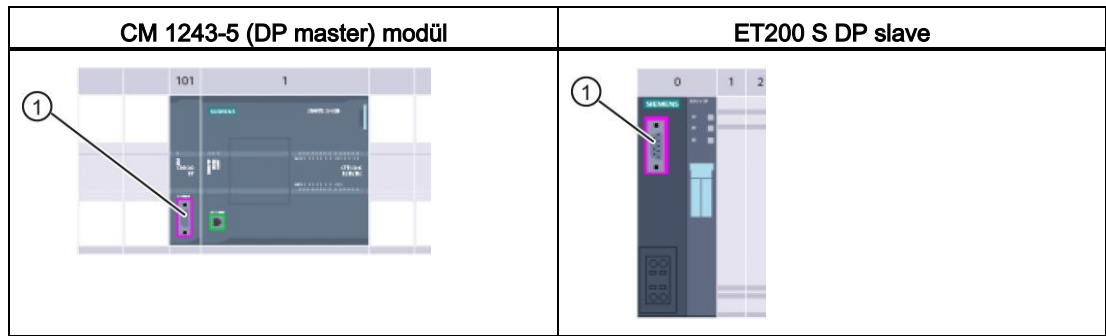
Daha fazla bilgi için (Sayfa 149): "Cihaz yapılandırma: Bir ağ bağlantısı oluşturma" bölümüne bakınız.

### 10.3.3.3 CM 1243-5 modülü ve DP slave'e PROFIBUS adreslerini atama

#### PROFIBUS ara yüz yapılandırması

İki PROFIBUS cihazı arasındaki mantıksal ağ bağlantılarını yapılandırdıktan sonra, PROFIBUS ara yüzler için parametreleri yapılandırabilirsiniz. Bunu yapmak için, CM 1243-5 modülü üzerindeki mor PROFIBUS kutusunu tıklayınız ve denetim penceresinde "Özellikler" sekmesi PROFIBUS ara yüzünü görüntüler. DP slave PROFIBUS ara yüzü aynı şekilde yapılandırılır.

Çizelge 10- 30 CM 1243-5 (DP master) modülünü ve ET200 S DP slave PROFIBUS ara yüzleri yapılandırmak



① PROFIBUS port

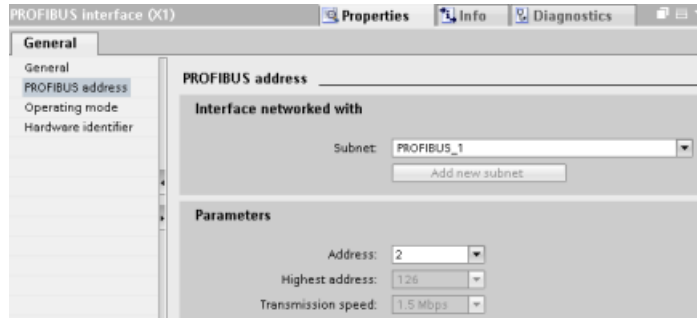
#### PROFIBUS adresi atama

PROFIBUS ağında her cihaza bir PROFIBUS adresi atanır. Bu adres, aşağıdaki istisnalar dışında, 0 ile 127 arasında olabilir:

- Adres 0: Ağ yapılandırması ve/veya veri yoluna bağlı programlama araçları için rezerve edilmiştir.
- Adres 1: İlk master için Siemens tarafından rezerve edilmiştir.
- Adres 126: Bir anahtar ayarı olmayan ve ağ üzerinden yeniden adreslenmesi gereken, fabrikadan gelen cihazlar için rezerve edilmiştir.
- Adres 127: Ağdaki tüm cihazlara yayın mesajları göndermek için rezerve edilmiştir ve operasyonel cihazlara atanamaz.

Bu nedenle, PROFIBUS işletme cihazları için kullanılabilen adresler 2 ile 125 arasındadır.

Özellikler penceresinde, "PROFIBUS adresi" yapılandırma girişini seçiniz. STEP 7, cihazın PROFIBUS adresini atamak için kullanılan PROFIBUS adres yapılandırma iletişim kutusunu görüntüler.



Çizelge 10- 31 PROFIBUS adresi için parametreler

Parametre	Açıklama	
Ara ağ	Cihazın bağlı olduğu alt ağ adı. Yeni bir alt ağ oluşturmak için " Add new subnet " düğmesine tıklayınız. "Bağlı değil" varsayılandır. İki bağlantı tipi mümkündür: <ul style="list-style-type: none"> <li>"Bağlı değil" varsayılanı, yerel bağlantı sağlar.</li> <li>Ağda iki veya daha fazla cihaz olduğunda bir alt ağ gereklidir.</li> </ul>	
Parametreler	Adres	Cihaz için Atanan PROFIBUS adresi
	En yüksek adres	En yüksek PROFIBUS adresi, PROFIBUS (örneğin, DP master) üzerindeki aktif istasyonlarına dayanmaktadır. En yüksek PROFIBUS adresi örneğin 15'e ayarlanmış olsa bile, pasif DP slave'ler bağımsız olarak, 1 ila 125 PROFIBUS adreslerine sahip olur. En yüksek PROFIBUS adresi, belirteç yönlendirme (gönderme haklarını yönlendirme) için geçerlidir ve belirteç sadece aktif istasyonlarına yönlendirilir. En yüksek PROFIBUS adresini belirtmek veri yolunu optimize eder.
	İletim hızı	Yapılandırılmış PROFIBUS ağ iletim hızı: PROFIBUS iletim hızları 9.6 kbit/s ile 12 Mbit/s arasında değişir. İletim hızı ayarı, kullanılan PROFIBUS düğümlerin özelliklerine bağlıdır. İletim hızı en yavaş düğüm tarafından desteklenen hızdan daha büyük olmamalıdır. İletim hızı normalde, aynı iletim hızını (otomatik baud) otomatik olarak kullanan tüm DP slave'lerle birlikte, PROFIBUS ağ üzerindeki master için ayarlanır.

### 10.3.4 Dağıtılmış I/O komutları

Bu haberleşme ağları ile dağıtılmış I/O komutlarının nasıl kullanılacağı hakkında bilgi için (Sayfa 314) " Dağıtılmış I/O (PROFINET, PROFIBUS, ya da AS-i) " bölümüne bakınız.

### 10.3.5 Tanılama (diyagnostik) komutları

Bu iletişim ağları ile bu komutların nasıl kullanılacağı hakkında bilgi için "Tanılama (PROFINET veya PROFIBUS)": "Tanılama komutları" (Sayfa 345) bölümüne bakınız.

### 10.3.6 Dağıtılmış I/O için tanılama olayları

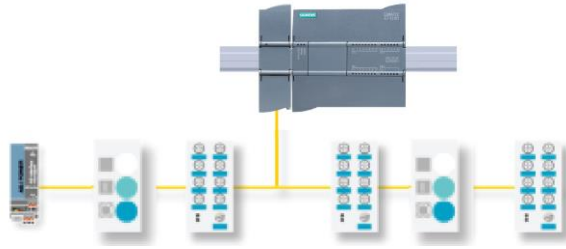
Bu haberleşme ağları ile bu tanılama bilgilerinin nasıl kullanılacağı hakkında bilgi için "Tanılama (PROFINET veya PROFIBUS)": "Dağıtılmış I/O için Tanılama olayları" (Sayfa 345) bölümüne bakınız.

## 10.4 AS-i

S7-1200 AS-i master CM 1243-2, bir AS-i ağının bir S7-1200 CPU'ya eklenmesine izin verir.

Harekete geçirici / sensör ara yüzü, ya da AS-i, otomasyon sistemlerinde düşük seviyede bir tekli master ağ bağlantı sistemidir. CM 1243-2, ağ için AS-i master olarak hizmet vermektedir. Tekli AS-i kablo kullanarak, sensörler ve harekete geçiriciler (AS-i slave cihazları), CM 1243-2 üzerinden CPU bağlanabilir. CM 1243-2, CM 1243-2 ye atanan I/O adresleri aracılığıyla harekete geçiriciler ve sensörlerden gelen bütün AS-i ağ koordinasyon ve röle verilerini ve durum bilgilerini CPU'ya işler. Slave tipine bağlı olarak ikili veya analog değerlere erişebilirsiniz. AS-i slave'ler, AS-i sisteminin giriş ve çıkış kanallarıdır ve sadece CM 1243-2 tarafından çağrıldığında aktif olur.

Aşağıdaki şekilde, S7-1200, AS-i operatör paneli ve I/O modülü analog/dijital slave cihazlar kontrol eden AS-i master'dır.



AS-i' yi S7-1200 V4.0 CPU'larla birlikte kullanmak için AS-i Master CM firmware'i V1.1'e yükseltmeniz gerekir.

Dijital olarak güvenli bir (SD) kart kullanarak bu yükseltmeyi yapabilirsiniz.

### Not

AS-i CM firmware i son sürümüne güncellenmeniz için Siemens Servis ve Destek web sitesini mevcuttur (<http://support.automation.siemens.com/WW/view/en/43416171>) kullanmanız tavsiye edilir.



## 10.4.1 Bir AS-i master ve slave cihaz yapılandırma

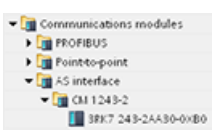

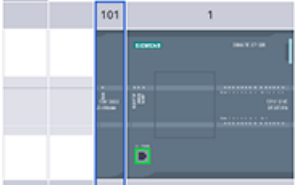
AS-i master CM 1243-2, S7-1200 otomasyon sistemine bir haberleşme modülü olarak entegre edilmiştir.

AS-i master CM 1243-2 hakkında detaylı bilgi bulabilirsiniz " SIMATIC S7-1200 için AS-i master CM 1243-2 ve AS-i veri ayırma ünitesi DCM 1271 " Kılavuzu (<http://support.automation.siemens.com/WW/view/en/50414115/133300>).

### 10.4.1.1 AS-i master CM 1243-2 ve AS-i slave ekleme

CPU' ya AS-i master CM1243-2 modül eklemek için donanım kataloğu kullanınız. Bu modüller CPU nun sol tarafına bağlanır ve maksimum üç AS-i master CM1243-2 modül kullanılabilir. Donanım yapılandırması içine bir modül eklemek için donanım kataloğunda modülü seçiniz ve çift tıklayınız ya da vurgulanan slota modülü sürükleyiniz.

Çizelge 10- 32 Cihaz yapılandırması için AS-i master CM1243-2 modülü ekleme

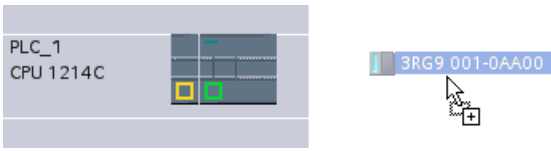
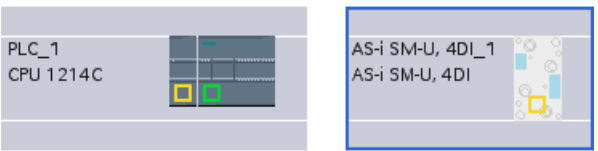
Modül	Modülü seç	Modül ekle	Sonuç
CM 1243-2 AS-i Master			

AS-i slave'leri eklemek için donanım kataloğunu kullanınız. Örneğin, Donanım Kataloğunda bir "I/O modülü, kompakt, dijital, giriş" slave eklemek için aşağıdaki kapları genişletiniz:

- Alan cihazları
- AS-Ara yüz slave

Sonra, parça numaraları listesinden "3RG9 001-0AA00" seçiniz (AS-i SM-U, 4DI), ve aşağıdaki şekilde gösterildiği gibi "I / O modülü, kompakt, dijital, giriş" slave ekleyiniz.

Çizelge 10- 33 Cihaz yapılandırması için AS-i slave ekleme

AS-i slave ekleme	Sonuç
	

### 10.4.1.2 İki AS-i cihazı arasında bağlantı için lojik devrenin konfigürasyonu

AS-i master CM1243-2 yapılandırıldıktan sonra, artık ağ bağlantılarınızı yapılandırmak için hazırsınız.

Cihazlar ve Ağlar portalında, projenizde cihazlar arasındaki ağ bağlantıları oluşturmak için "Ağ görünümü" nı kullanınız. AS-i bağlantısı oluşturmak için, ilk cihazda yeşil (AS-i) kutuyu tıklayınız ve ikinci cihazda AS-i kutusuna bir çizgi sürükleyin. Farenin düğmesini bırakın ve AS-i bağlantınız birleştirilir.

Daha fazla bilgi için (Sayfa 149): "Cihaz yapılandırma: Bir ağ bağlantısı oluşturma " bölümüne bakınız.

### 10.4.1.3 AS-i master CM1243-2 özelliklerinin yapılandırılması

AS-i ara yüz parametrelerini yapılandırmak için, AS-i master CM1243-2 modülünde sarı AS-i kutusuna tıklayınız ve denetim penceresindeki "Özellikler" sekmesi AS-i ara yüzünü görüntüler.

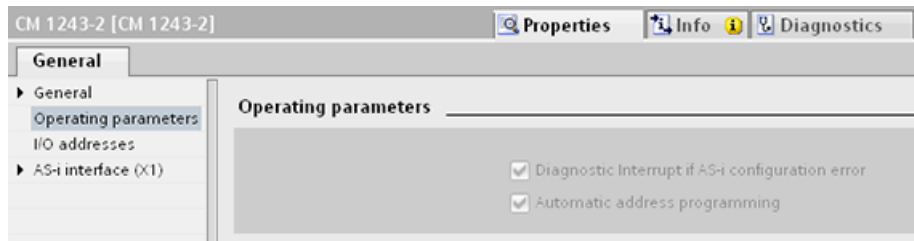
STEP 7 denetim penceresinde genel bilgileri, adresleri ve çalışma parametrelerini görüntüleyebilir, yapılandırabilir ve değiştirebilirsiniz:

Çizelge 10- 34 AS-i master CM1243-2 modül özellikleri

Özellik	Açıklama
Genel	AS-i master CM 1243-2 ismi
Çalışma parametreleri	AS-i master tepki parametreleri
I/O adresleri	Slave I / O adresleri için Adres alanı
AS-i ara yüzü (X1)	Atanmış AS-i ağı

#### Not

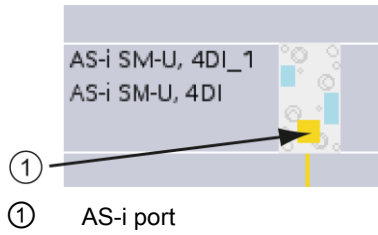
"AS-i yapılandırmasında arızalar için Tanılama kesmesi" ve "Otomatik adres programlama" her zaman aktiftir ve bu nedenle gri gösterilmiştir.



#### 10.4.1.4 Bir AS-i adresinin bir AS-i slave'e atanması

##### AS-i slave ara yüzünü yapılandırma

AS-i ara yüzü parametrelerini yapılandırmak için AS-i slave üzerinde sarı AS-i kutusuna tıklayınız ve denetim penceresinde "Özellikler" sekmesi AS-i ara yüzünü görüntüler.



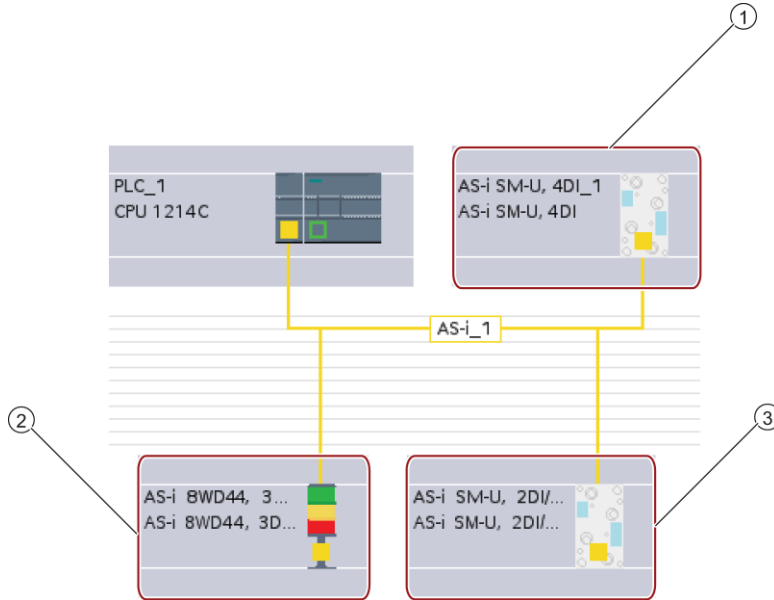
##### AS-i slave adresinin atanması

AS-i ağda, her cihaza bir AS-i slave adresi atanır. Bu adres 0 ile 31 arasında değişir; Ancak, adres 0, yalnızca yeni slave cihazlar için ayrılmıştır. Toplam 62 ye kadar slave cihazına 1 (A ya da B) ile 31 (A veya B) arasından Slave adresleri atanır.

"Standart" AS-i cihazları A veya B gösterilişi olmayan bir dizi adrese sahip, tüm adresi kullanır. "A / B düğüm" AS-i cihazları, 31 adresin her birinin iki kez kullanılmasına imkan sağlayarak, her adresin A veya B kısmını kullanır. Adres alanı aralığı 1A ile 31A ve 1B ile 31B'dir.

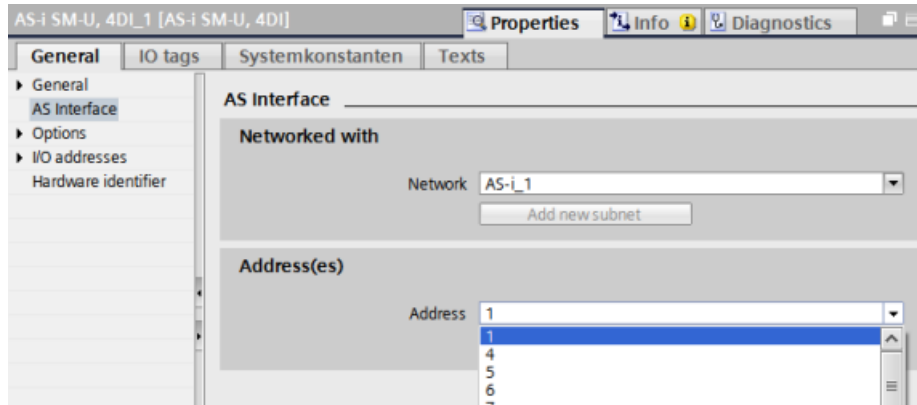
1- 31 aralığında herhangi bir adres, AS-i slave cihaza atanabilir; diğer bir deyişle, ilk slave'e aslında 1 adresinin veriliş verilmemesi ya da slave'lerin 21 adresi ile başlayıp başlamaması fark etmez.

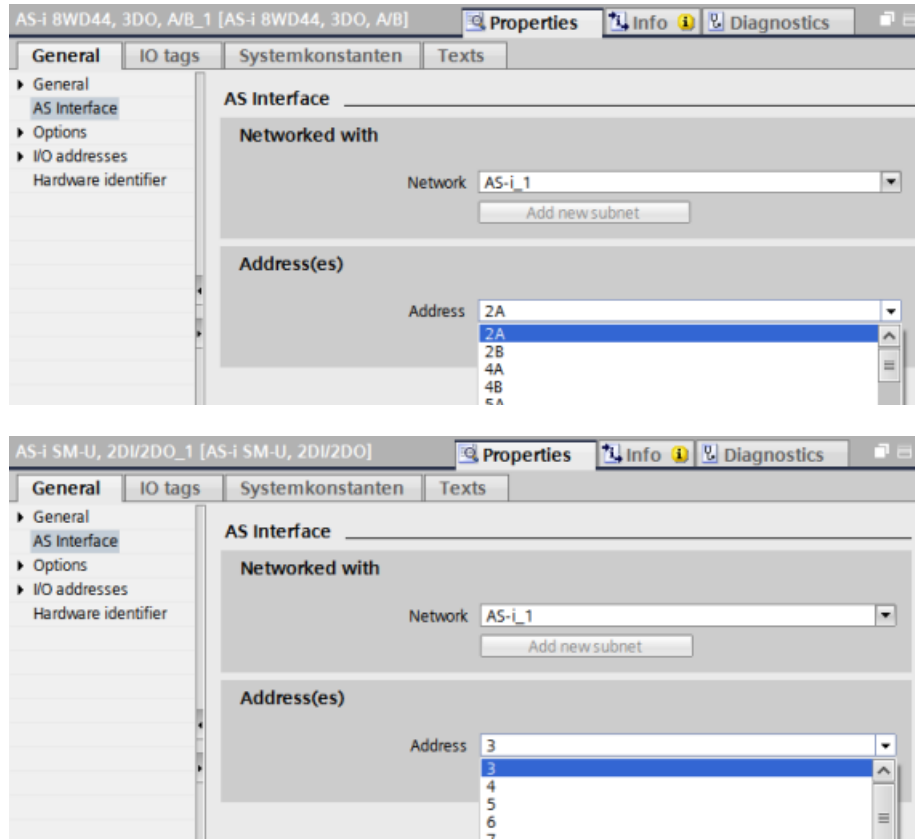
Aşağıdaki örnekte, üç AS-i cihazının "1" (bir standart tip cihaz), "2A" (bir A / B düğüm tipi cihaz) ve "3" (standart tip bir cihaz) olarak adreslenmiştir:



- ① AS-i slave adresi 1; Cihaz: AS-i SM-U, 4DI; sipariş numarası: 3RG9 001-0AA00
- ② AS-i slave adresi 2A; Cihaz: AS-i 8WD44, 3DO, A/B; sipariş numarası: 8WD4 428-0BD
- ③ AS-i slave adresi 3; Cihaz: AS-i SM-U, 2DI/2DO; sipariş numarası: 3RG9 001-0AC00

Burada AS-i slave adresinizi giriniz:





Çizelge 10- 35 AS-i ara yüz parametreleri

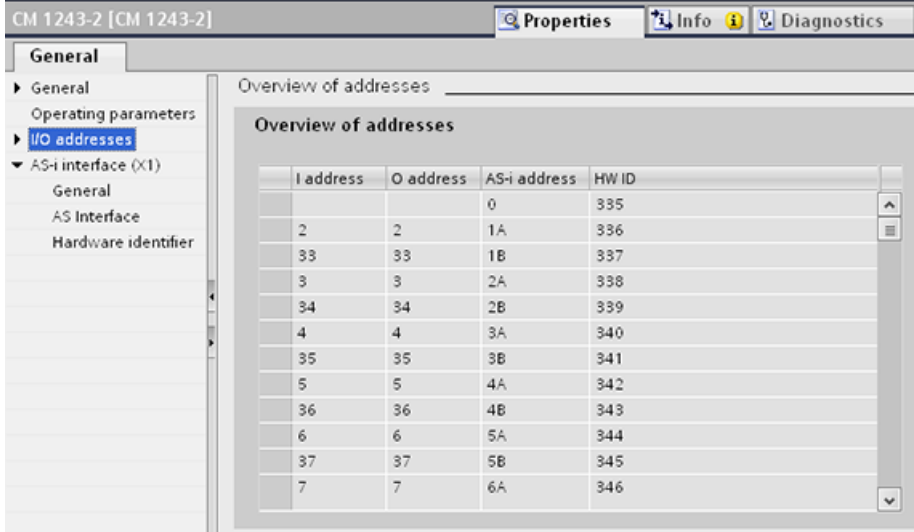
Parametre	Açıklama
Ağ	Cihazın bağlı olduğu ağın adı
Adres(ler)	Toplam 62 adede kadar slave cihazı için slave cihaza 1 (A veya B) ile 31 (A veya B) aralığında AS-i adresi atanır

## 10.4.2 Kullanıcı programı ve AS-i slave'ler arasında veri alışverişi

### 10.4.2.1 STEP 7 temel yapılandırma

AS-i master, CPU'nun I/O alanında bir 62 bayt veri alanını tutar. Dijital verilere erişim, burada bayt cinsinden yapılır; Her Slave için, bir bayt giriş ve bir bayt çıkış verileri vardır.

AS-i dijital slave'lerin AS-i bağlantılarının, atanan bayt veri bitleri için ataması AS-i master CM 1243-2'nin denetim penceresinde gösterilir.



The screenshot shows the 'Overview of addresses' window in the SIMATIC Manager. The window title is 'CM 1243-2 [CM 1243-2]'. The left sidebar shows the 'General' tab selected, with 'I/O addresses' highlighted. The main area displays a table with the following data:

I address	O address	AS-i address	HW ID
		0	335
2	2	1A	336
33	33	1B	337
3	3	2A	338
34	34	2B	339
4	4	3A	340
35	35	3B	341
5	5	4A	342
36	36	4B	343
6	6	5A	344
37	37	5B	345
7	7	6A	346

(Örneğin, "AND") uygun bit mantık işlemlerine ya da bit atamalarına sahip görüntülenen I / O adreslerini kullanarak kullanıcı programında AS-i slave'lerin verilerine erişebilirsiniz.

#### Not

STEP 7 ile AS-i slave'ler yapılandırılmazsa "Sistem atama" otomatik olarak devreye girer.

Herhangi bir slave yapılandırmazsanız, AS-i master CM1243-2'yi "ACTUAL > EXPECTED" çevrimiçi işlevini kullanarak gerçek veri yolu yapılandırması hakkında bilgilendirmeniz gerekir.

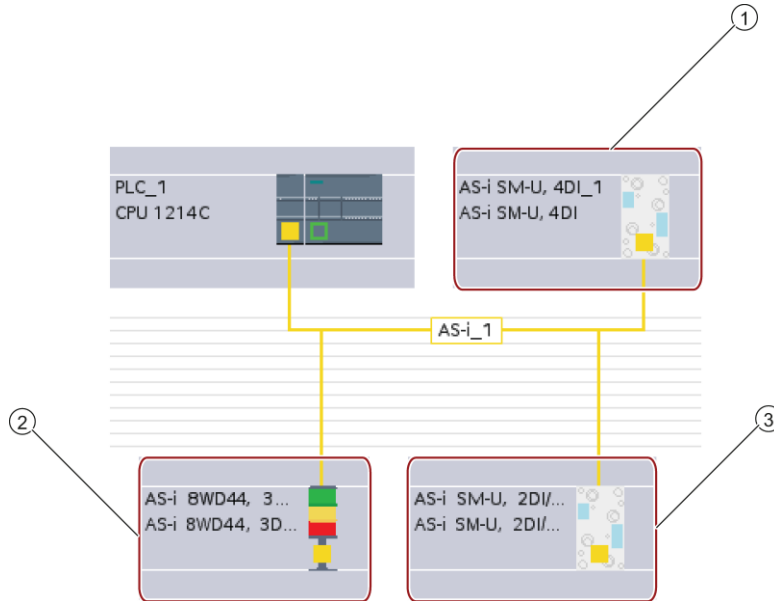
#### Ayrıntılı bilgi

AS-i master CM 1243-2 hakkında detaylı bilgi bulabilirsiniz " SIMATIC S7-1200 için AS-i master CM 1243-2 ve AS-i veri ayırma ünitesi DCM 1271 " Kılavuzu (<http://support.automation.siemens.com/WW/view/en/50414115/133300>).

## 10.4.2.2 STEP 7 ile slave'leri yapılandırma

### AS-i dijital değerlerin aktarılması

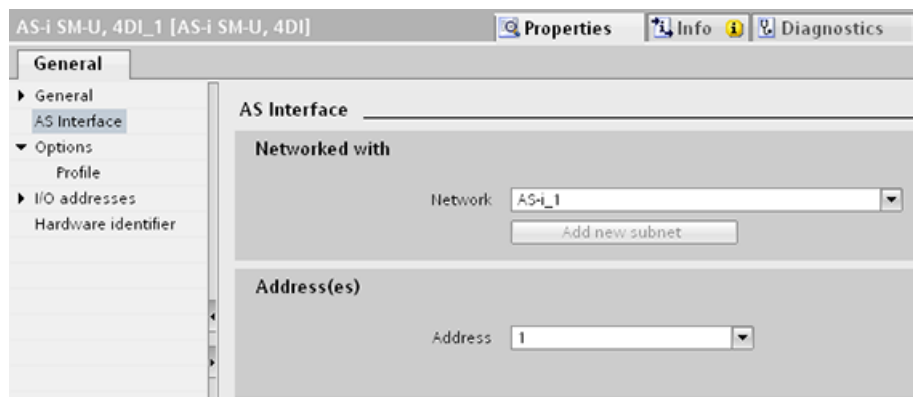
CPU, döngüsel olmayan çalışmada AS-i master CM1243-2 üzerinden AS-i slave'lerin dijital girişler ve çıkışlarına erişir. Veriye, I / O adresleri üzerinden veya veri kayıt transferi yoluyla erişilir.



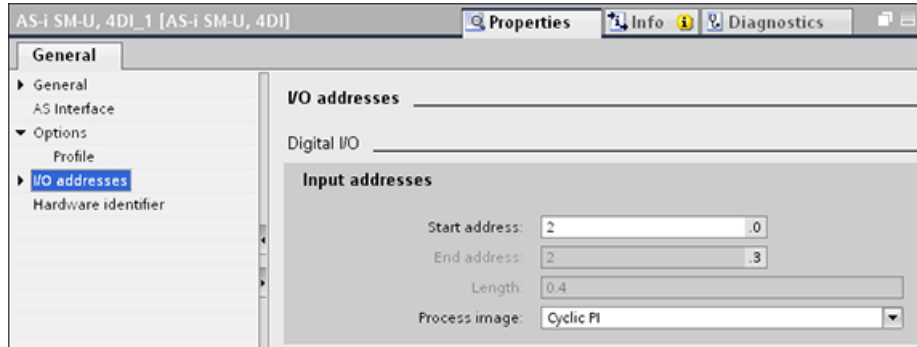
- ① AS-i slave adresi 1
- ② AS-i slave adresi 2A
- ③ AS-i slave adresi 3

Dijital verilere erişim (diğer bir deyişle, her AS-i dijital slave'e bir bayt atanır) burada bayt cinsinden yapılır. STEP 7'de AS-i slave'leri yapılandırdığınızda, kullanıcı programdan gelen verilere erişim için kullanılan I/O adresi, ilgili AS-i slave için denetim penceresinde görüntülenir.

Yukarıda AS-i ağdaki Dijital giriş modülüne (AS-i SM-U, 4DI), slave adresi 1 atanmıştır. Dijital giriş modülünü tıklayarak, cihazın " Properties " inde "AS interface " sekmesi aşağıda gösterildiği gibi slave adresini görüntüler:

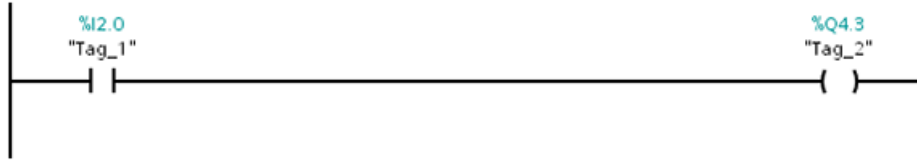


Yukarıda AS-i ağdaki Dijital giriş modülüne (AS-i SM-U, 4DI), I/O adresi 2 atanmıştır. Dijital giriş modülünü tıklayarak, cihazın " Properties " inde "I/O addresses" sekmesi aşağıda gösterildiği gibi I/O adresini görüntüler:



(Örneğin, "AND") uygun bit mantık işlemlerine ya da bit atamalarına sahip olan I/O adreslerini kullanarak kullanıcı programında AS-i slave'lerin verilerine erişebilirsiniz. Aşağıdaki basit program atamanın nasıl çalıştığını göstermektedir:

Giriş 2.0, bu programda toplanır. AS-i sistemde, bu giriş (Giriş bayt 2, bit 0) slave1 e aittir. Daha sonra ayarlanan çıkış 4.3, AS-i slave 3'e (Çıkış bayt 4, bit 3) karşılık gelir.



### AS-i analog değerlerin aktarılması

Bu AS-i slave' i STEP 7' de analog slave olarak yapılandırmışsanız bir AS-i slave in analog verisine CPU proses görüntüsü üzerinden erişebilirsiniz.

STEP 7 de analog slave yapılandırmadıysanız, döngüsel olmayan işlevler (veri kayıt ara yüzü) ile sadece AS-i slave' in verilerine erişebilirsiniz. CPU kullanıcı programında, AS-i çağruları RDREC (veri kaydı okuma) ve WRREC (veri kaydı yazma) dağıtılmış I / O komutları kullanılarak okuma ve yazma yapılır.

#### Not

STEP 7 üzerinden belirtilen ve S7 istasyonuna indirilen AS-i slave'lerin bir yapılandırması, S7 istasyonu başlatması sırasında AS-i master CM1243-2 deki CPU tarafından aktarılır. "Sistem atama" Çevrimiçi fonksiyonu (Sayfa 582) ("ACTUAL -> EXPECTED") ile belirlenen mevcut herhangi bir yapılandırma üzerine yazılacaktır.

### Ayrıntılı bilgi

AS-i master CM 1243-2 hakkında detaylı bilgi bulabilirsiniz " SIMATIC S7-1200 için AS-i master CM 1243-2 ve AS-i veri ayırma ünitesi DCM 1271 " Kılavuzu (<http://support.automation.siemens.com/WW/view/en/50414115/133300>).



### 10.4.3 Dağıtılmış I / O komutları

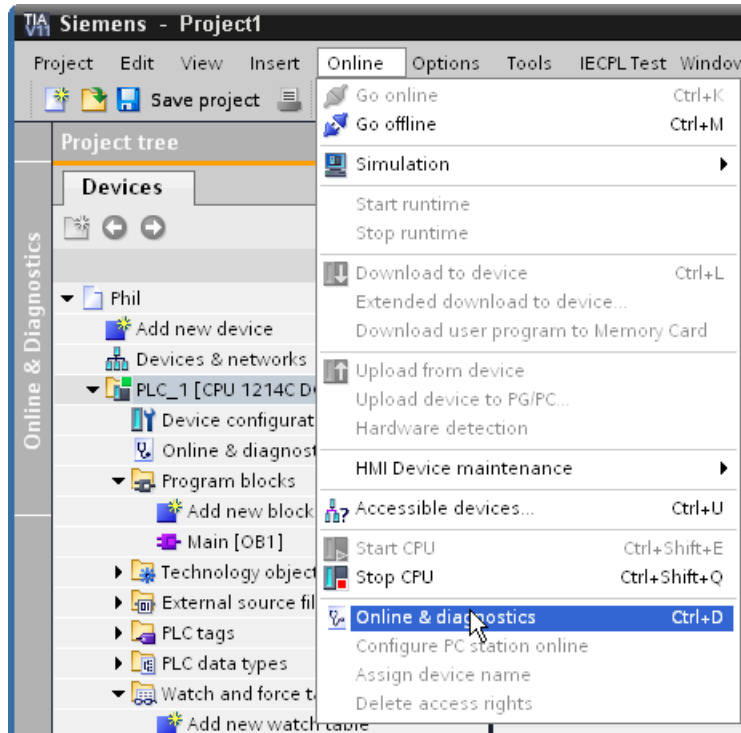
Bu haberleşme ağları ile dağıtılmış I/O komutlarının nasıl kullanılacağı hakkında bilgi için (Sayfa 314) " Dağıtılmış I/O (PROFINET, PROFIBUS, ya da AS-i) " bölümüne bakınız.

### 10.4.4 AS-i çevrimiçi araçlarla çalışma

#### AS-i çalışma modlarının çevrimiçi değiştirilmesi

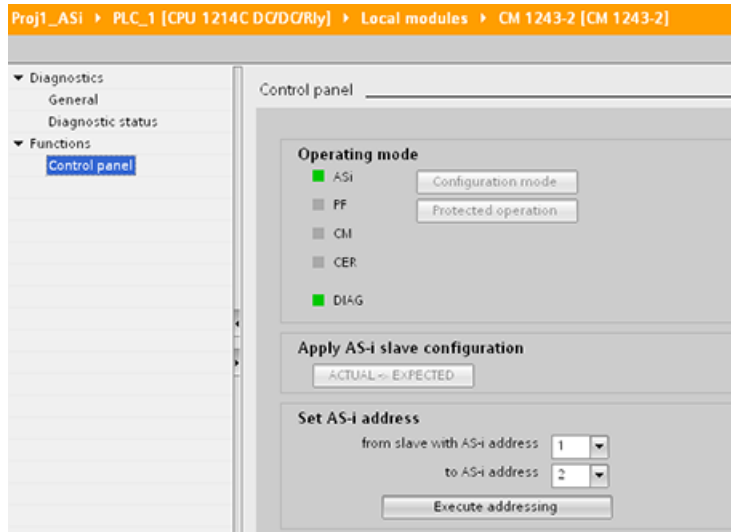
AS-i çalışma modları görüntülemek ve değiştirmek için çevrimiçi olmanız gerekir.

Çevrimiçi olmanız için ilk olarak seçilen AS-i master CM1243-2 modülü ile "Cihaz yapılandırması" nda olmanız gerekir ve sonra araç çubuğundaki "Go Çevrimiçi " düğmesine tıklayınız. Sonra, "Çevrimiçi" menüsünden "Çevrimiçi ve diyagnostikler " komutunu seçiniz.



İki AS-i çalışma modu vardır:

- Koruma modu:
  - AS-i slave cihazını ve CPU I/O adreslerini değiştiremezsiniz.
  - Yeşil "CM" LED OFF olur.
- Yapılandırma modu:
  - AS-i slave cihazınız ve CPU I/O adreslerinde gerekli değişiklikleri yapabilirsiniz.
  - Yeşil "CM" LED ON olur.



"Set AS-i adresi" alanında, AS-i slave adresini değiştirebilirsiniz. Bir adres atanmamış yeni bir slave'in, her zaman 0 adresi vardır. Bir adres atanması olmadan bu yeni bir slave olarak master tarafından tespit edilir ve bir adres atanana kadar normal bir haberleşme içine dahil değildir.

### Yapılandırma hatası

Sarı "CER" LED'i ON olduğunda, AS-i slave cihaz yapılandırmasında bir hata var demektir. AS-i alan ağ slave cihaz yapılandırması ile AS-i master CM1243-2 modül slave cihaz yapılandırması üzerine yazmak için "ACTUAL > EXPECTED" düğmesini seçiniz.

## 10.5 S7 haberleşme

### 10.5.1 GET ve PUT (uzak CPU' dan oku ve yaz) komutları

PROFINET ve PROFIBUS bağlantıları aracılığıyla S7 CPU'lar ile haberleşmek için GET ve PUT komutları kullanabilirsiniz. Bu sadece "PUT/GET haberleşmesi ile İzin erişimi" fonksiyonu yerel CPU özelliklerinin "Koruma" özelliğinde partner CPU için aktif olduğunda mümkündür:

- Uzak CPU verilerine erişme: Bir S7-1200 CPU uzak CPU'ların (S7-200/300/400/1200) değişkenlerine adres atamak için sadece ADDR\_x giriş alanındaki mutlak adreslerini kullanabilirsiniz.
- Standart DB verilerine erişme: Bir S7-1200 CPU, uzak S7 CPU nun standart DB sinde DB değişkenlerine adres atamak için sadece ADDR\_x giriş alanındaki mutlak adreslerini kullanabilirsiniz.
- Optimize DB verilerine erişme: Bir S7-1200 CPU, uzak S7-1200 CPU nun optimize edilmiş bir DB de DB değişkenlerine erişemez.
- Yerel bir CPU verilerine erişme: Bir S7-1200 CPU, sırasıyla, GET veya PUT komutunun RD\_x veya SD\_x giriş alanlarına girişler olarak mutlak veya sembolik adreslerini kullanabilir.

---

#### Not

#### V4.0 CPU programı GET / PUT işlemi, otomatik olarak etkinleştirilmez

Bir V3.0 CPU programı GET/PUT işlemi, V4.0 CPU'da otomatik olarak etkinleştirilir.

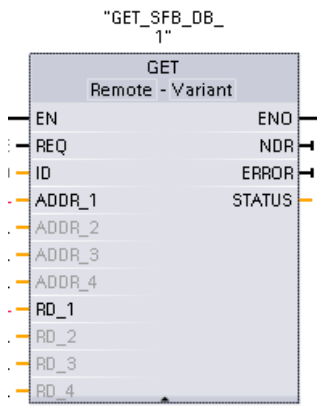
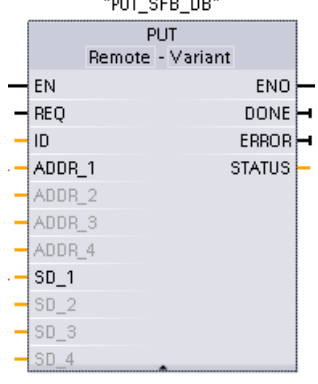
Ancak, V4.0 CPU'daki bir V4.0 CPU programı GET/PUT işlemi, otomatik olarak etkinleştirilmez. GET/PUT erişimi sağlamak için CPU "Cihaz yapılandırması", denetim penceresi "Özellikler" sekmesinde, "Koruma" özelliğine gitmelisiniz (Sayfa 190).

---

## Haberleşme

### 10.5 S7 haberleşme

Çizelge 10- 36 GET ve PUT komutları

LAD / FBD	SCL	Açıklama
	<pre>"GET_DB" (   req:=_bool_in_,   ID:=_word_in_,   ndr=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_word_out_,   addr_1:=_remote_inout_,   [...addr_4:=_remote_inout_,]   rd_1:=_variant_inout_   [...rd_4:=_variant_inout_]);</pre>	<p>Uzak bir S7 CPU' dan veri okumak için GET komutunu kullanınız. Uzak CPU, RUN veya STOP modunda olabilir.</p> <p>Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.</p>
	<pre>"PUT_DB" (   req:=_bool_in_,   ID:=_word_in_,   done=&gt;_bool_out_,   error=&gt;_bool_out_,   status=&gt;_word_out_,   addr_1:=_remote_inout_,   [...addr_4:=_remote_inout_,]   sd_1:=_variant_inout_,   [...sd_4:=_variant_inout_]);</pre>	<p>Uzak bir S7 CPU' ya veri yazmak için PUT komutunu kullanınız. Uzak CPU, RUN veya STOP modunda olabilir.</p> <p>Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.</p>

Çizelge 10- 37 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
REQ	Input	Bool
ID	Input	CONN_PRG (Word)
NDR (GET)	Output	Bool
DONE (PUT)	Output	Bool

Parametre ve tipi		Veri tipi	Açıklama
ERROR STATUS	Output Output	Bool Word	<ul style="list-style-type: none"> <li>• ERROR=0 STATUS değer: – 0000H: Ne uyarı nede hata var – &lt;&gt; 0000H: Uyarı, STATUS detaylı bilgi sağlar</li> <li>• ERROR=1 Bir hata var. Hata yapısı hakkında STATUS detaylı bilgi sağlar.</li> </ul>
ADDR_1	InOut	Remote	Okunacak (GET) veya gönderilecek (PUT) verileri depolayan uzak CPU da bellek alanlarını işaret eder
ADDR_2	InOut	Remote	
ADDR_3	InOut	Remote	
ADDR_4	InOut	Remote	
RD_1 (GET) SD_1 (PUT)	InOut	Variante	Okunacak (GET) veya gönderilecek (PUT) verileri depolayan uzak CPU da bellek alanlarını işaret eder.
RD_2 (GET) SD_2 (PUT)	InOut	Variante	İzin verilen veri tipleri: Bool (izin verilen yalnızca tek bit), Byte, Char, Word Int, DWord, DINT veya Real.
RD_3 (GET) SD_3 (PUT)	InOut	Variante	Not: İşaretçi bir DB'ye erişirse, aşağıdaki gibi mutlak adresi belirtmeniz gerekir:
RD_4 (GET) SD_4 (PUT)	InOut	Variante	P# DB10.DBX5.0 Bayt 10 Bu durumda, 10, GET veya PUT da kullanılacak bayt sayısını gösterir.

ADDR\_x (uzak CPU) ve RD\_x veya SD\_x (yerel CPU) parametreleri için uzunluk (bayt sayısı) ve veri tiplerinin eşleştiğinden emin olmalısınız. Tanımlayıcı "byte" tan sonraki sayı, ADDR\_x, RD\_x veya SD\_x parametresi tarafından referanslanan bayt sayısıdır.

### Not

Bir GET komutunda alınan toplam bayt sayısı veya bir PUT komutunda gönderilen toplam bayt sayısı sınırlıdır. Sınırlamalar, dört olası adres ve bellek alanlarından kaçını kullandığınıza bağlıdır:

- Sadece ADDR\_1 ve RD\_1/SD\_1 kullanırsanız, bir GET komutu 222 bayt alabilir ve bir PUT komutu 212 bayt gönderebilir.
- ADDR\_1, RD\_1/SD\_1, ADDR\_2 ve RD\_2/SD\_2 kullanırsanız, bir GET komutu toplam 218 bayt alabilir ve bir PUT komutu toplam 196 bayt gönderebilirsiniz.
- ADDR\_1, RD\_1/SD\_1, ADDR\_2, RD\_2/SD\_2, ADDR\_3 ve RD\_3/SD\_3 kullanırsanız GET komutu toplam 214 bayt alabilir ve bir PUT komutu toplam 180 bayt gönderebilir.
- ADDR\_1, RD\_1/SD\_1, ADDR\_2, RD\_2/SD\_2, ADDR\_3, RD\_3/SD\_3, ADDR\_4, RD\_4/SD\_4 kullanırsanız GET komutu toplam 210 bayt alabilir ve bir PUT komutu toplam 164 bayt gönderebilir.

Adres ve bellek alan parametrelerinin her birinin bayt sayısının toplamı belirlenmiş sınırlara eşit veya daha az olmalıdır. Bu sınırları aşarsanız, GET veya PUT komutu bir hata döndürür.

REQ parametresinin yükselen kenarında, okuma işlemi (GET) veya yazma işlemi (PUT), ID, ADDR\_1 ve RD\_1 (GET) veya SD\_1 (PUT) parametrelerini yükler.

- GET için: Uzak CPU, bir sonraki taramayla başlayarak, istenen veriyi alma alanlarına (RD\_x) döndürür. Okuma işlemi hatasız tamamlandığında, NDR parametresi 1 olarak ayarlanır. Yeni bir operasyon sadece önceki işlem tamamlandıktan sonra başlatılabilir.
- PUT için •: Yerel CPU, verileri, uzak CPU'daki bellek konumuna (ADDR\_x) göndermeye (SD\_x) başlar. Yazma işlemi hatasız tamamlandığında, uzak CPU bir yürütme onayı döndürür. PUT komutunun DONE parametresi 1 olarak ayarlanır. Yeni bir yazma işlemi yalnızca önceki işlem tamamlandıktan sonra başlatılabilir.

### Not

Veri tutarlılığını sağlamak için, verilere erişmeden veya başka okuma veya yazma işlemi başlatılmadan önce, daima operasyon tamamlandığında (GET için NDR = 1 veya PUT için DONE = 1) değerlendirme yapılır.

ERROR ve STATUS parametreleri (PUT) (GET) okuma veya yazma işleminin durumu hakkında bilgi sağlar.

Çizelge 10- 38 Hata bilgisi

ERROR	STATUS (ondalık)	Açıklama
0	11	<ul style="list-style-type: none"> <li>• Önceki iş henüz tamamlanmadığı için yeni iş gerçekleşemez.</li> <li>• İş artık daha düşük önceliğe sahip bir öncelik sınıfında işleniyor.</li> </ul>
0	25	Haberleşme başladı. İş işleniyor.
1	1	Aşağıdaki gibi haberleşme problemleri bulunmaktadır: <ul style="list-style-type: none"> <li>• Bağlantı açıklaması yüklenmedi (yerel veya uzak)</li> <li>• Bağlantı kesildi (örneğin: kablo, CPU devre dışı bırakıldı veya CM/CB/CP STOP modunda)</li> <li>• Partnerle olan bağlantı henüz oluşturulmadı.</li> </ul>
1	2	Partner cihazdan olumsuz onay. Görev yürütülemez.
1	4	Veri uzunluğu ya da veri tipi içeren gönderme alanı işaretçilerinde (GET için RD_x veya PUT için SD_x) Hatalar.
1	8	Partner CPU da erişim hatası
1	10	Yerel kullanıcı belleğin erişim mümkün değil (örneğin silinmiş bir DB ye erişim girişimi)
1	12	SFB çağrıldığında: <ul style="list-style-type: none"> <li>• GET veya PUT a ait olmayan Kopya DB belirtilmiştir</li> <li>• Kopya DB belirtilmedi, daha ziyade paylaşılan DB belirtildi</li> <li>• Kopya DB bulunmadı (yeni bir kopya DB yükleniyor)</li> </ul>
1	20	<ul style="list-style-type: none"> <li>• Paralel işleri / örneklerinin sayısını aştı</li> <li>• Kopyalar CPU-RUN da aşırı yüklendi</li> </ul> Bu durum, GET veya PUT komutunun ilk yürütmesi için mümkün
1	27	CPU da karşılık gelen hiçbir GET veya PUT komutu yok.

## 10.5.2 S7 bağlantısı oluşturma

### Bağlantı mekanizmaları

PUT / GET komutları ile uzak bağlantı partnerlerine erişmek için, kullanıcının izni olması gerekir.

Varsayılan olarak, "PUT/ GET haberleşmesi ile erişime izin ver " seçeneği etkin değildir. Bu durumda, CPU okuma ve yazma erişimi, yalnızca yerel CPU ve haberleşme partnerinin her ikisi için yapılandırma veya programlama gerektiren haberleşme bağlantıları için mümkündür. Örneğin BSEND / BRCV komutları ile erişim mümkündür.

Yerel CPU nun (yerel CPU da haberleşme partneri ile hiçbir haberleşme yapılandırması / programlamasının olmadığı anlamına gelir) sadece bir sunucu olarak kullanıldığı bağlantılar, CPU çalışması sırasında bundan dolayı mümkün değildir, örneğin:

- Haberleşme modülleri aracılığıyla PUT / GET, FETCH / WRITE veya FTP erişimi
- Diğer S7 CPU'lardan PUT / GET erişimi
- PUT / GET haberleşme yoluyla HMI erişimi

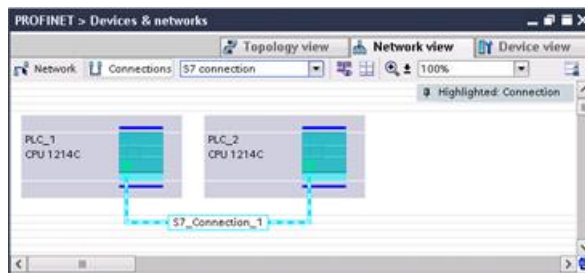
Eğer istemci tarafında CPU verilerine erişime izin vermek istiyorsanız, yani, CPU haberleşme hizmetlerini kısıtlamak istemiyorsanız daha fazla bilgi için "S7-1200 CPU için Erişim koruma" (Sayfa 190) bölümüne bakınız.

### Bağlantı tipleri

Seçtiğiniz bağlantı tipi partner istasyonuna haberleşme bağlantı oluşturur. Bağlantı kurulur, çalıştırılır, otomatik olarak izlenir.

Cihazlar ve Ağlar portalında, projenizde cihazlar arasındaki ağ bağlantılarını oluşturmak için "Ağ görünümü" nü kullanınız. Öncelikle, "Bağlantılar" sekmesini tıklayınız ve sonra açılan listeden bağlantı tipini seçiniz, sadece sağa (örneğin, bir S7 bağlantısı). İlk cihazda yeşil (PROFINET) kutuyu tıklayınız ve ikinci cihazda PROFINET kutusuna bir çizgi sürükleyiniz. Farenin düğmesini bırakın ve PROFINET bağlantınız birleştirilir.

Daha fazla bilgi için (Sayfa 149): "Bir ağ bağlantısı oluşturma " bölümüne bakınız.



Haberleşme komutunun "Properties" yapılandırma diyaloguna erişmek için: " Highlighted: Connection " düğmesine tıklayınız.

### 10.5.3 İki cihaz arasındaki Yerel / Partner bağlantı yolunu yapılandırma

#### Genel parametreleri yapılandırma

Haberleşme komutunun " Properties" yapılandırma diyalogunda haberleşme parametrelerini belirtiniz. Komutun herhangi bir bölümünü seçtiğiniz zaman bu diyalog sayfanın altına yakın görünür.

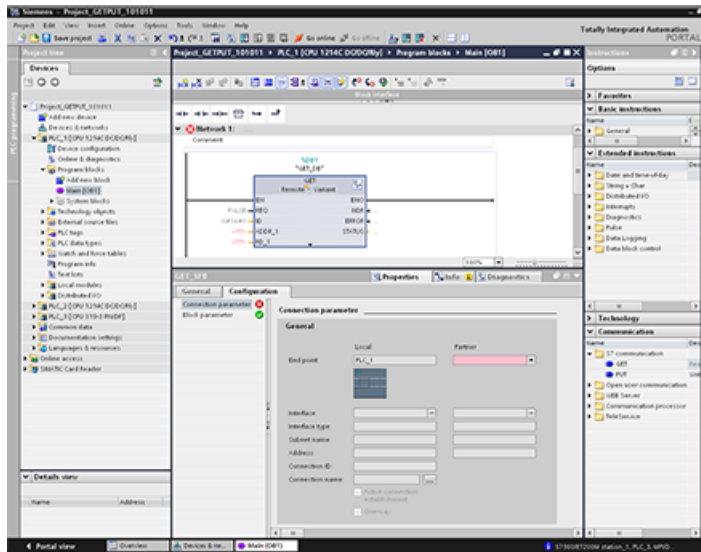
Daha fazla bilgi için: " Cihaz yapılandırması: Yerel / Partner bağlantı yolunu yapılandırma (Sayfa 150) " bölümüne bakınız.

Bağlantı parametreleri iletişim penceresinde "Adres Detayları" bölümünde, kullanılacak TSAP'lar veya bağlantı noktalarını tanımlayınız. CPU'daki TSAP veya bir bağlantı portu "Yerel TSAP" alanına girilir. Partner CPU da bağlantı için atanan TSAP veya port "Partner TSAP" alanı altında girilir.

### 10.5.4 GET / PUT bağlantı parametre atama

GET / PUT komutları bağlantı parametre ataması, CPU-CPU S7 haberleşme bağlantılarını yapılandırmak için bir kullanıcı yardımcısıdır.

Bir GET veya PUT bloğunu yerleştirdikten sonra, GET / PUT komutları bağlantı parametre ataması başlatılır:





Komutun herhangi bir bölümünü seçtiğiniz zaman denetim penceresi bağlantı özelliklerini görüntüler. Haberleşme komutu için "Properties" " Configuration " sekmesinde haberleşme parametrelerini belirleyebilirsiniz.

### Not

#### V4.0 CPU programı GET / PUT işlemi, otomatik olarak etkinleştirilmez

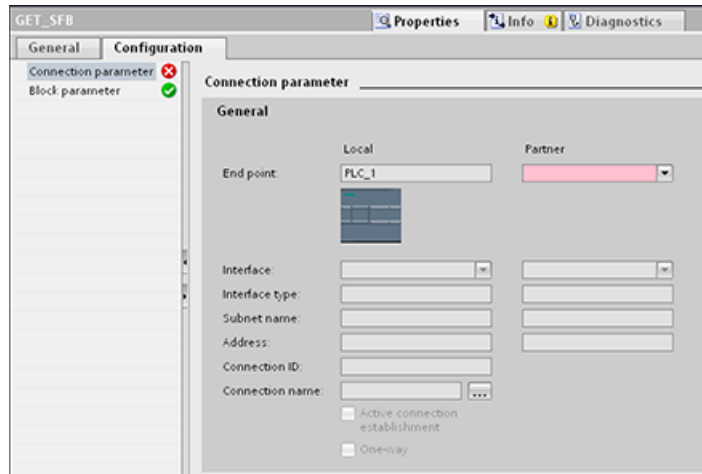
Bir V3.0 CPU programı GET / PUT işlemi, V4.0 CPU da otomatik olarak etkinleştirilir.

Ancak, V4.0 CPU'daki bir V4.0 CPU programı GET / PUT işlemi, otomatik olarak etkinleştirilmez. GET / PUT erişimi sağlamak için CPU "Cihaz yapılandırması", denetim penceresi "Properties" sekmesinde, "Koruma" özelliğine gitmelisiniz (Sayfa 190).

### 10.5.4.1 Bağlantı parametreleri

"Bağlantı parametreleri" sayfası, gerekli S7 bağlantısını yapılandırmanıza ve GET/PUT blok parametresi "ID" tarafından başvuru "Bağlantı ID" parametresini yapılandırmanıza izin verir. Sayfanın içeriğinde, yerel uç nokta hakkında bilgiler vardır ve yerel ara yüzü tanımlamanızı sağlar. Ayrıca partner uç noktasını tanımlayabilirsiniz.

"Blok parametreleri" sayfası, ek blok parametrelerini yapılandırmanıza izin verir.



Çizelge 10- 39 Bağlantı parametresi: Genel tanımlar

Parametre	Tanım	
Bağlantı parametresi: Genel	Uç nokta	"Yerel uç noktası": Yerel işlemciye atanan isim "Partner uç noktası": Partner (uzak) CPU' ya atanan isim Not: "Partner uç noktası" açılır listesinde, sistem geçerli projenin tüm olası S7 bağlantı Partnerlerinin yanı sıra "belirtilmemiş" seçeneğini görüntüler. Belirtilmemiş bir partner (örneğin, üçüncü taraf cihaz haberleşme partneri) hali hazırda STEP 7 projesinde olmayan haberleşme partnerini temsil eder.
	Ara yüz	Ara yüzleri atanan isim Not: Yerel ve partner ara yüzleri değiştirerek bağlantıyı değiştirebilirsiniz
	Ara yüz tipi	Ara yüz tipi
	Alt ağ ismi	Alt ağlara atanan isim

Parametre	Tanım
Adres	Atanan IP adresleri Not: Bir "belirtilmemiş" haberleşme partneri için üçüncü taraf cihazın uzak adresini belirtebilirsiniz.
Bağlantı ID	ID numarası: GET / PUT bağlantı parametre ataması tarafından otomatik olarak oluşturulur
Bağlantı ismi	Yerel ve Partner CPU veri saklama yeri: GET / PUT bağlantı parametre ataması tarafından otomatik oluşturulur
Aktif bağlantı kurulumu	Aktif bağlantı olarak Yerel CPU yu seçmek için onay kutusu
Tek yön	Tek yönlü veya iki yönlü bir bağlantı belirtmek için onay kutusu; salt okunur Not: Bir PROFINET GET / PUT bağlantısında, yerel ve partner cihazların her ikisi, bir sunucu ya da istemci olarak görev yapabilir. Bu, iki yönlü bir bağlantı sağlar ve "tek yönlü" onay kutusu işareti kaldırılır. Bir PROFIBUS GET / PUT bağlantısında, bazı durumlarda, Partner cihaz sadece bir sunucu (örneğin, bir S7-300) olarak görev yapabilir ve "tek yönlü" onay kutusu işaretlenir.

### Bağlantı ID parametresi

Sistem tanımlı bağlantı ID'lerini değiştirmek için üç yol vardır:

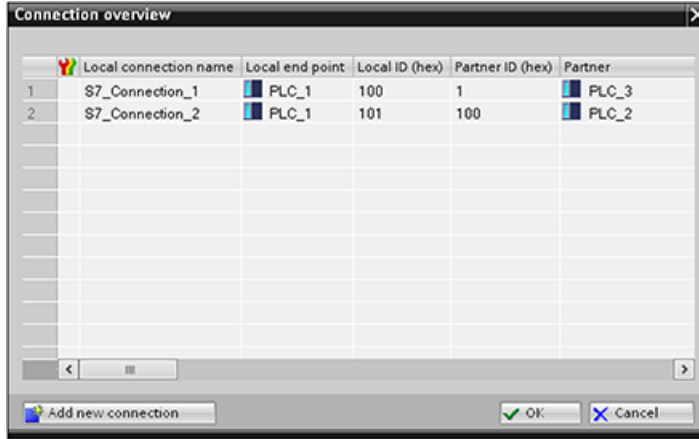
37. Kullanıcı, GET / PUT bloğunda geçerli ID'yi doğrudan değiştirebilir. Yeni ID var olan bağlantıya aitse, bağlantı değiştirilir.
38. Kullanıcı GET / PUT bloğunda geçerli ID'yi doğrudan değiştirebilir, ama yeni ID zaten yok. Yeni S7 bağlantısı sistem tarafından oluşturulur.
39. Kullanıcı "Connection overview " iletişim kutusu yoluyla geçerli ID'yi değiştirebilirsiniz: Kullanıcı girişi karşılık gelen GET / PUT bloğunda ID parametresi ile senkronize edilir.

#### Not

GET / PUT bloğunun "ID" parametresi, bir bağlantı ismi değildir, ancak aşağıdaki örnekteki gibi bir sayısal ifade yazılır: W#16#1

## Bağlantı adı parametresi

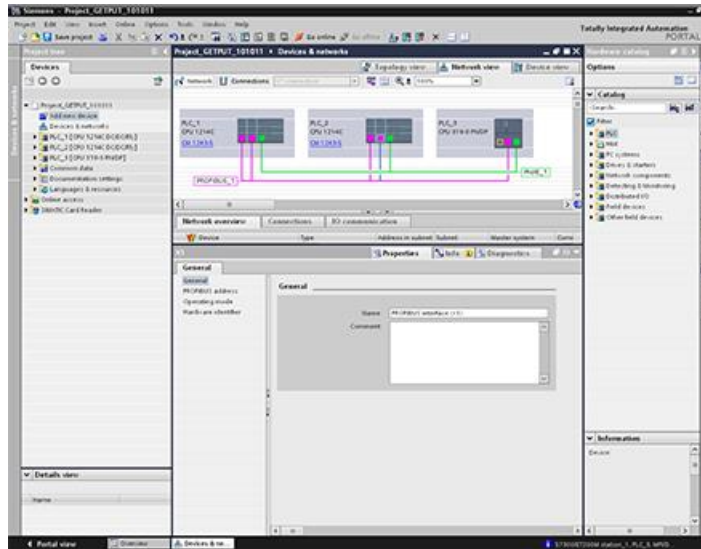
Bağlantı adı özel bir kullanıcı kontrolü, "Connection overview" iletişim penceresi yoluyla düzenlenebilir. Bu iletişim penceresi mevcut GET / PUT haberleşme için bir alternatif olarak seçilebilen tüm kullanılabilir S7 bağlantıları sunmaktadır. Kullanıcı, bu çizelgede tamamen yeni bir bağlantı oluşturabilir. "Connection overview" iletişim penceresini başlatmak için "Connection name" alanının sağındaki düğmeye tıklatınız.



	Local connection name	Local end point	Local ID (hex)	Partner ID (hex)	Partner
1	S7_Connection_1	PLC_1	100	1	PLC_3
2	S7_Connection_2	PLC_1	101	100	PLC_2

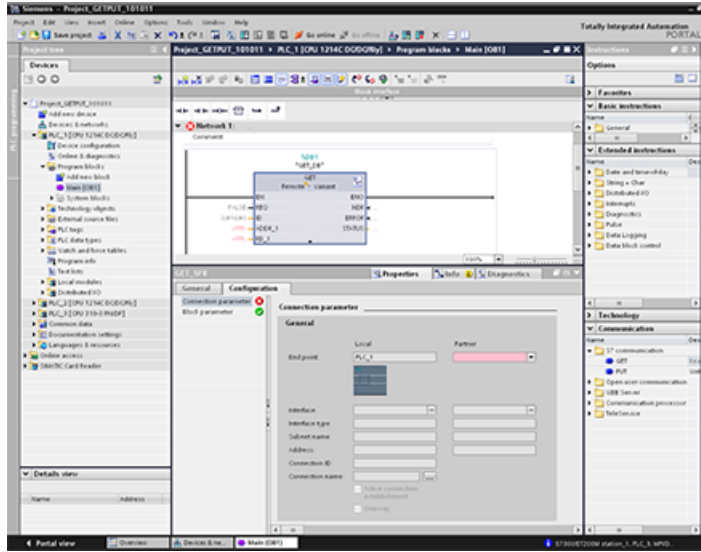
## 10.5.4.2 CPU-CPU S7 bağlantı yapılandırma

Aşağıdaki şekilde gösterildiği gibi PLC\_1, PLC\_2 ve PLC\_3 yapılandırma göz önüne alındığında, "PLC\_1" için GET veya PUT blokları ekleyiniz.



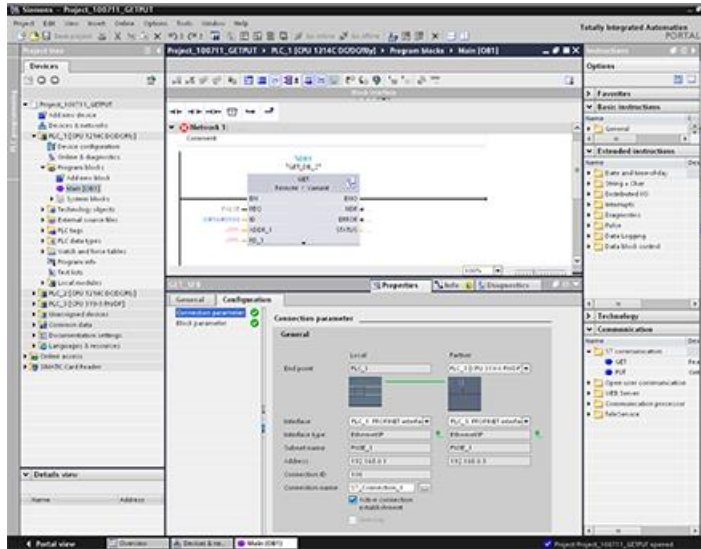
GET veya PUT komutu için, aşağıdaki menü seçimleri ile "Properties" sekmesi, otomatik olarak denetim penceresinde görüntülenir.

- "Configuration"
- "Connection parameters"



## PROFINET S7 bağlantı yapılandırma

"Partner uç noktası" için "PLC\_3" seçiniz.



Sistem aşağıdaki değişikliklere tepki verir:

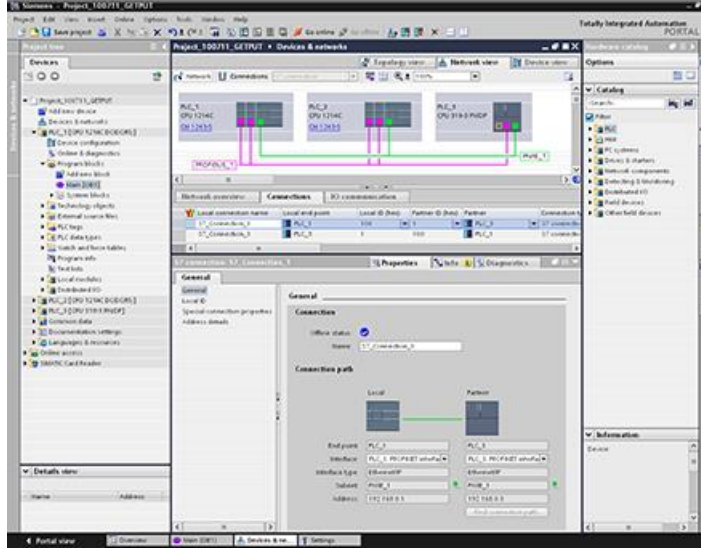
Çizelge 10- 40 Bağlantı parametresi: Genel değerler

Parametre	Tanım
Bağlantı parametresi: Genel	Uç nokta "Yerel uç nokta" salt okunur olarak "PLC_1" içerir. "Partner uç nokta " alanı" PLC_3[CPU319-3PN/DP]" içerir: <ul style="list-style-type: none"> <li>• Kırmızıdan beyaza renk değişimi</li> <li>• "Partner" cihaz görüntüsü gösterilir.</li> <li>• Bir bağlantı hattı PLC_1 ve PLC_3 cihaz görüntüleri (yeşil Ethernet hat) arasında görünür.</li> </ul>
	Ara yüz "Yerel ara yüz", "CPU1214C DC/DC/DC, PROFINET ara yüz (R0/S1)" içerir. "Partner ara yüz ", "CPU319-3PN/DP, PROFINET ara yüz (R0/S2)" içerir.
	Ara yüz tipi "Yerel ara yüz tip", "Ethernet/IP" içerir; kontrol salt okunur. "Partner ara yüz tip", "Ethernet/IP " içerir; kontrol salt okunur. Ara yüz tipi görüntüleri Yerel ve Partner "Ara yüz tipi" (yeşil Ethernet simgesi) yanında sağda gösterilir.
	Alt ağ ismi "Yerel alt ağ ismi", "PN/IE_1" içerir; kontrol salt okunur. "Partner alt ağ ismi "PN/IE_1" içerir; kontrol salt okunur.
	Adres "Yerel Adres", Yerel IP adresi içerir; kontrol salt okunur. "Partner Adres", Partner IP adresi içerir; kontrol salt okunur.
	Bağlantı ID "Bağlantı ID", "100" içerir. Program editöründe, Ana [OB1] de, GET/PUT bloğu "Bağlantı ID" değeri de "100" içeriyor.
	Bağlantı ismi "Bağlantı ismi" varsayılan bağlantı adını (örneğin, "S7_Connection_1 ") içerir; kontrolü etkindir.
	Aktif bağlantı kurulumu Aktif bağlantı olarak yerel CPU yu seçmek için kontrol yapın ve etkinleştiriniz.
	Tek yön Salt okunur ve işaretlenmemiş. Not: "PLC_1" (bir S7-1200 CPU 1214CDC/DC/Rly) ve "PLC_3" (bir S7-300 CPU 319-3PN/DP) PROFINET GET/PUT bağlantısında hem bir sunucu ve hemde bir istemci olarak davranabilir ve iki yönlü bağlantıya izin verir.

Özellik Görünüm ağacında GET/PUT simgesi kırmızıdan yeşile döner.

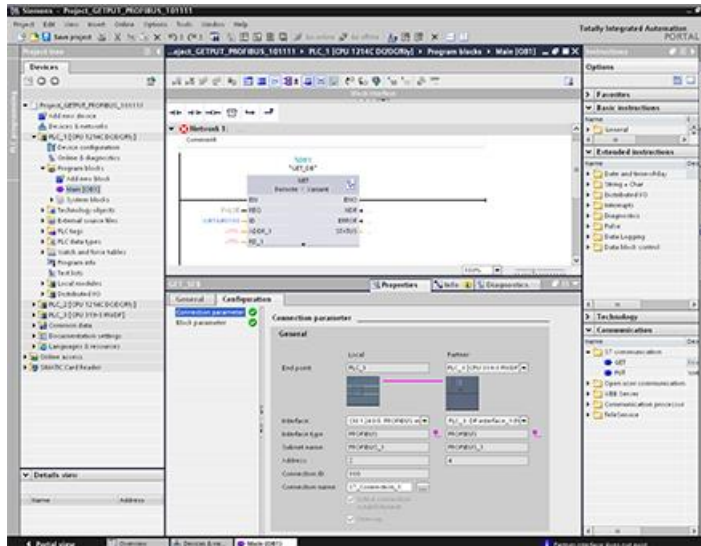
## Tamamlanmış PROFINET S7 bağlantısı

"Network view" de, iki yönlü S7 bağlantısı "PLC\_1" ve "PLC\_3" arasındaki "Connections" çizelgesinde gösterilmiştir.



## PROFIBUS S7 bağlantısı yapılandırma

"Partner uç noktası" için "PLC\_3" seçiniz.



Sistem aşağıdaki değişikliklere tepki verir:

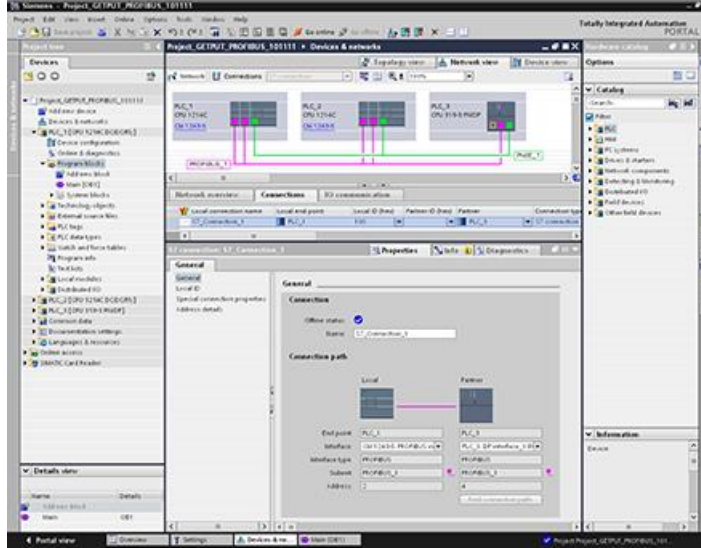
Çizelge 10- 41 Bağlantı parametresi: Genel değerler

Parametre	Tanımlama
Bağlantı parametresi: Genel	Uç nokta "Yerel uç nokta" salt okunur olarak "PLC_1" içerir. Partner uç nokta " alanı" PLC_3 [CPU319-3PN/DP]" içerir: <ul style="list-style-type: none"> <li>• Kırmızıdan beyaza renk değişimi</li> <li>• "Partner" cihaz görüntüsü gösterilir.</li> <li>• Bir bağlantı hattı PLC_1 ve PLC_3 cihaz görüntüleri (mor PROFIBUS hat) arasında görünür.</li> </ul>
	Ara yüz "Yerel ara yüz", "CPU1214C DC/DC/DC, PROFIBUS ara yüz (R0/S1)" içerir. Partner ara yüz ", "CPU319-3PN/DP, PROFIBUS ara yüz (R0/S2)" içerir.
	Ara yüz tipi "Yerel ara yüz tip", "PROFIBUS" içerir; kontrol salt okunur. "Partner ara yüz tip", "PROFIBUS" içerir; kontrol salt okunur. Ara yüz tipi görüntüleri Yerel ve Partner "Ara yüz tipi" (mor PROFIBUS simgesi) yanında sağda gösterilir.
	Alt ağ ismi "Yerel alt ağ ismi", "PROFIBUS_1" içerir; kontrol salt okunur. "Partner alt ağ ismi "PROFIBUS_1" içerir; kontrol salt okunur.
	Adres "Yerel Adres", Yerel IP adresi içerir; kontrol salt okunur. "Partner Adres", Partner IP adresi içerir; kontrol salt okunur.
	Bağlantı ID "Bağlantı ID", "100" içerir. Program editöründe, Ana [OB1] de, GET / PUT bloğu "Bağlantı ID" değeri de "100" içeriyor.
	Bağlantı ismi "Bağlantı ismi" varsayılan bağlantı adını (örneğin, "S7_Connection_1 ") içerir; kontrolü etkindir.
	Aktif bağlantı kurulumu Aktif bağlantı olarak yerel CPU yu seçmek için kontrol yapın ve etkinleştiriniz.
	Tek yön Salt okunur ve işaretlenmiş. Not: "PLC_3" (bir S7-300 CPU 319-3PN/DP) PROFINET GET/PUT bağlantısında bir sunucu (bir istemci olarak davranamaz) ve yalnızca tek yönlü bağlantıya izin verir.

Özellik Görünüm ağacında GET/PUT simgesi kırmızıdan yeşile döner.

**Tamamlanmış PROFIBUS S7 bağlantısı**

"Network view "de, iki yönlü S7 bağlantısı "PLC\_1" ve "PLC\_3" arasındaki " Connections " çizelgesinde gösterilmiştir.





## Web sunucu

S7-1200 için Web sunucusu CPU ve proses verisi hakkındaki veriye Web sayfası erişimi sağlar.

Bir PC'den veya mobil cihazdan S7-1200 Web sayfalarına erişebilirsiniz. Web sunucusu Web sayfalarına erişmek için kullandığınız cihaz ile uyumlu bir boyutta ve formatta sayfaları görüntüler. Web sunucusu asgari 240 x 240 piksel çözünürlüğü destekler.



### Standart Web sayfaları

S7-1200, PC'nizin Web Tarayıcısı (Sayfa 606) veya bir mobil cihazdan (Sayfa 607) erişebileceğiniz standart Web sayfaları içerir:

- Giriş ( Sayfa 613) - Standart Web sayfalarına giriş noktası
- Başlangıç Sayfası ( Sayfa 614) - CPU hakkında genel bilgiler
- Kimlik ( Sayfa 615) - Seri, sipariş ve sürüm numaraları dahil işlemci hakkında detaylı bilgi
- Modül Bilgisi ( Sayfa 616) - Yerel raftaki modüller hakkında bilgi ve firmware güncelleme yeteneği
- Haberleşme ( Sayfa 620) - Ağ adresleri, fiziksel haberleşme ara yüz özellikleri ve haberleşme istatistikleri hakkında bilgiler,
- Tanılama arabelleği ( Sayfa 616) - Tanılama arabelleği
- Değişken Durumu ( Sayfa 620) - Adres veya PLC etiket adına göre erişilebilir CPU değişkenleri ve I/O,
- Dosya Gezgini ( Sayfa 622) - CPU veya bellek kartında dahili olarak depolanan dosyalar (örneğin, veri kayıtlarını ve tarifler) için gezgin,
- Giriş ( Sayfa 610) - Farklı bir kullanıcı olarak oturum açınız veya oturum kapatınız.

Bu sayfalar S7-1200 CPU'ya gömülüdür ve İngilizce, Almanca, Fransızca, İspanyolca, İtalyanca ve Basitleştirilmiş Çince mevcuttur. Bazı sayfalar, sayfayı görüntülemek için STEP 7'de yapılandırduğunuz ilave kullanıcı ayrıcalıklarını (Sayfa 604) gerektirir. Standart Web sayfaları ve onlara nasıl erişileceği hakkında ayrıntılı bilgi için Standart Web sayfaları (Sayfa 609) bölümüne bakınız.

**Not**

S7-1200 V4.0 CPU'lar ayrı bir firmware güncelleme standart Web sayfası içermez. Firmware güncelleme özelliği modül bilgi sayfasına dahil edilmiştir.

---

**Kullanıcı tanımlı Web sayfaları**

S7-1200'de CPU veri erişebilirsiniz kullanıcı-tanımlı Web sayfaları oluşturmak için destek sağlar. Seçtiğiniz HTML yazma yazılımı ile bu sayfaları geliştirmek ve CPU verilere erişmek için HTML kodu komutları ön tanımlı "awp" (Otomasyonu Web Programlama) içerebilir. Kullanıcı tanımlı Web sayfaları (Sayfa 625) kullanıcı-tanımlı Web sayfalarının geliştirilmesi üzerinde spesifik bilgi için bölüm ve ilişkili yapılandırma ve STEP 7 programlama bakınız.

Sen standart Web sayfalarından bir PC veya mobil cihaz ya gelen kullanıcı tanımlı sayfalara erişebilirsiniz.

**Web tarayıcısı gereksinimi**

Web sunucusu aşağıdaki PC Web tarayıcılarını destekler:

- Internet Explorer 8.0
- Internet Explorer 9.0
- Mozilla Firefox 17.0.1
- Google Chrome 23.0
- Apple Safari 5.1.7 (Windows)
- Apple Safari 6.0.2 (Mac)

Web sunucusu aşağıdaki mobil cihaz Web tarayıcılarını destekler:

- Internet Explorer 6.0 ve öncesi, HMI paneller için
- Mobile Safari 7534.48.3 (iOS 5.0.1)
- Mobile Android Browser 2.3.4
- Mobile Google Chrome 23.0

Standart veya kullanıcı-tanımlı Web sayfalarının ekran ile müdahale Kısıtlamaları (Sayfa 664) bölümünde görebilirsiniz tarayıcı ilgili kısıtlamalar.

## 11.1 Web sunucusunu etkinleştirme

Bağlanmak niyetinde olduğunuz CPU için Cihaz Yapılandırma STEP 7'de Web sunucusunu etkinleştiriniz.

Web sunucusunu etkinleştirmek için şu adımları izleyiniz:

40.Cihaz Yapılandırma görünümünde CPU seçiniz.

41.Denetçisi penceresinde, CPU özelliklerinden "Web sunucusu" seçeneğini seçiniz.

42.Bu modül üzerinde etkinleştiriniz web sunucusu" için onay kutusunu seçiniz.

43.Web sunucusuna güvenli erişim istemek için, onay kutusunu "yalnızca HTTPS ile izni erişimi" seçeneğini seçiniz.

### WARNING

#### Web sunucusu aracılığıyla CPU yetkisiz erişim

CPU veya değerleri geçersiz PLC değişkenleri olabilir değişen yetkisiz erişim süreç operasyon bozmaya ve ölüm, ciddi yaralanma ve/veya maddi hasara neden olabilir.

Web sunucusunu sağlayan yetkili kullanıcıların çalışma modu değişiklikleri yapmasını sağlar, çünkü PLC veri yazar ve firmware güncellemeleri, Siemens aşağıdaki güvenlik uygulamaları gözlemlenmelidir:

- Sadece HTTPS protokolü ile Web sunucusuna erişimi etkinleştiriniz.
- Şifre koruması Web sunucusu kullanıcı kimlikleri güçlü bir parola ile (Sayfa 604) . Güçlü parolalar, uzunluğu en az on karakterler harf, rakam ve özel karakter karıştırınız, bir sözlükte bulunabilecek sözcükler değildir ve isimleri veya kişisel bilgiler elde edilebilir tanımlayıcı değildir. Şifreyi gizli ve sık değiştiriniz.
- " Herkes " kullanıcı varsayılan minimum ayrıcalıkları uzatmayınız.
- Web sayfası kullanıcıların geçersiz değerleri PLC değişkenleri değiştirebilirsiniz, çünkü program mantığı içinde değişkenler üzerinde hata denetimi ve aralık - kontrolü gerçekleştiriniz.
- S7 -1200 PLC Web'e bağlanmak için güvenli bir Sanal Özel Ağ (VPN) kullanınız korumalı ağ dışında bir yerden sunucu.

Cihaz yapılandırmasını yükledikten sonra, CPU'ya erişmek için standart Web sayfalarını kullanabilirsiniz. "Otomatik güncelleme" için "Enable" seçeneğini seçerseniz, standart Web sayfaları her on saniyede bir yenileme yapar.

Kullanıcı-tanımlı Web sayfaları (Sayfa 625) oluşturur ve etkinleştirirseniz, standart Web sayfası menüsünden onlara erişebilirsiniz.

**Not****Cihaz değiştirme: Bir V3.0 CPU'yu bir V4.0 CPU ile değiştirme ve proje dönüşümü**

Mevcut bir V3.0 CPU'yu bir V4.0 CPU (Sayfa 995) ile değiştirir ve V3.0 projenizi V4.0 projeye dönüştürürseniz, STEP 7 ve V4.0 CPU ile Web sunucusu ayarlarını korumak için aşağıdakilere dikkat ediniz:

- "Bu modüldeki web sunucusunu etkinleştirme"
- "İzin erişimi sadece HTTPS ile"

**Not**

Bir "Download in RUN" (Sayfa 803) devam ediyor ise yükleme tamamlanana kadar, standart ve kullanıcı tanımlı Web sayfaları veri değerlerini güncellemek veya veri değerlerini yazmak için izin vermez. Bir doküman yüklemesi devam ederken Web sunucusu veri değerlerini yazmak girişimlerini boşa çıkarır.

**Web sunucusu için diğer dilleri etkinleştirme**

Ayrıca isteğe bağlı standart Web sayfalarının görüntülenmesi için diğer dilleri seçebilirsiniz. Cihazınızın yapılandırma Özellikler sekmesinden "Kullanıcı ara yüz dili" ni seçiniz ve sonra Web sunucusu destekleyen altı dillerden birine bir STEP 7 projesi dil atayabilirsiniz. Cihaz Yapılandırmasını indirdikten sonra, standart Web sayfaları kullanıcı ara yüzü dili için bir seçici sağlar. Bir dil seçmezseniz, varsayılan dil İngilizcedir.

## 11.2 Web sunucusu kullanıcılarını yapılandırma

Web sunucusu aracılığıyla CPU erişim için çeşitli ayrıcalık seviyeleri ile kullanıcılara yapılandırabilirsiniz.

Web sunucusu kullanıcıları ve ilişkili ayrıcalıklarını yapılandırmak için şu adımları izleyiniz:

1. Cihaz Yapılandırma görünümünde CPU seçiniz.
2. Denetçisi penceresinde, CPU özelliklerinden "Web sunucusu" seçiniz ve Web sunucusu (Sayfa 603) etkinleştiriniz.
3. Web sunucusu özelliklerinde "Kullanıcı yönetimi" seçeneğini seçiniz.
4. Sağlamak istediğiniz kullanıcı girişi için kullanıcı adları, erişim düzeylerini ve şifreleri giriniz.

Yapılandırmayı CPU'ya yükledikten sonra, sadece ayrıcalıklara sahip olan yetkili kullanıcılar Web sunucusu işlevlerini erişebilirler.

## Web sunucusu erişim seviyeleri

STEP 7, şifresiz olarak "Herkes" adlı varsayılan bir kullanıcı sağlar. Varsayılan olarak, bu kullanıcı hiçbir ek ayrıcalığa sahip değildir ve sadece giriş yaptığında Web sayfalarını görebilir. Ancak, diğer kullanıcılar gibi "Herkes" kullanıcılarına ilave ayrıcalıklar verebilir ve yapılandırabilirsiniz:

- Sorgu tanılamaları
- Okuma etiketleri
- Yazma etiketleri
- Okuma etiketi durumu
- Yazma etiketi durumu
- Açık kullanıcı-tanımlı sayfaları
- Kullanıcı-tanımlı sayfalara yazma
- Okuma dosyaları
- Yazma/silme dosyaları
- Çalışma modunu değiştirme
- Flaş LED'ler
- Firmware güncellemesi yapma

Ek ayrıcalıklar eklemeksizin "Herkes" kullanıcısı sadece Başlat (Sayfa 614) ve Giriş (Sayfa 613) sayfalarını görebilir.



### WARNING

#### Web sunucusuna erişim

"Herkes" kullanıcısı için ayrıcalıklar tanınması Web sunucusunda şifresiz olarak oturum açılmasını mümkün kılar. CPU'ya yetkisiz erişim veya PLC değişkenlerini geçersiz değerlere değiştirme proses işlemini bozabilir ve ölüm, ağır yaralanma ve / veya maddi hasara neden olabilir.

Yeterli ayrıcalıklar verildiğinde "Herkes" kullanıcısının şifresiz olarak çalışma modunu değiştirmesine, PLC verilerine yazmasına ve firmware güncellemelerini yapmasına izin verildiği için Siemens aşağıdaki güvenlik uygulamalarını gözlemlemenizi önerir:

- Sadece HTTPS protokolü ile Web sunucusuna erişimi etkinleştiriniz.
- Web sunucusu kullanıcı ID'lerini güçlü bir parola ile şifre korumalı hale getirin. Güçlü parolalar; harf, rakam ve özel karakterin karışımından meydana gelen en az on karakter olmalı ve bir sözlükte bulunabilecek sözcükler olmamalı ve kişisel bilgilerden elde edilebilen tanımlayıcılar veya isimler olmamalıdır. Şifreyi gizleyin ve sık sık değiştiriniz.
- "Herkes" kullanıcısının varsayılan minimum ayrıcalıklarını artırmayınız.
- Web sayfası kullanıcıları PLC değişkenlerini geçersiz değerlere değiştirebildiği için program mantığınız içinde değişkenleriniz üzerinde hata kontrolü ve aralık kontrolü yapınız.
- Korunmalı ağınız dışında bir yerden S7 -1200 PLC Web sunucusuna bağlanmak için güvenli bir Sanal Özel Ağ (VPN) kullanınız.

## 11.3 PC'den Web sayfalarına erişim

Bir PC'den veya mobil cihazdan S7-1200 standart Web sayfalarına erişebilirsiniz.

Bir PC'den S7-1200 standart Web sayfalarına erişmek için şu adımları izleyiniz:

5. S7-1200 ve PC ortak bir Ethernet ağ üzerinde veya standart Ethernet kablosu ile birbirine doğrudan bağlı olduğundan emin olunuz.
6. "ww.xx.yy.zz" S7-1200 CPU' nun IP adresine karşılık geldiği yerde bir Web tarayıcısı açınız ve <http://ww.xx.yy.zz> URL giriniz.

Web tarayıcısı Giriş sayfasını açar.

### Not

Korumalı ağ dışında bir yerden S7-1200 PLC Web sunucusuna bağlanmak için güvenli bir Sanal Özel Ağ (VPN) kullanınız. Web ortamı veya işletim sisteminizin empoze edebileceği herhangi bir kısıtlamanın (Sayfa 664) farkında olunuz.

Alternatif olarak, Web tarayıcınızı belirli bir standart Web sayfası için adresleyebilirsiniz. Bunu yapmak için, <http://ww.xx.yy.zz/<Sayfa>.html> biçiminde URL giriniz. Burada <Sayfa> Standart Web sayfalarından birine karşılık gelir:

- Başla ( Sayfa 614) - CPU hakkında genel bilgiler
- Kimlik ( Sayfa 3615) - seri, sipariş ve sürüm numaraları dahil CPU hakkında detaylı bilgi
- Modül ( Sayfa 616) -yerel raftaki modülleri ve firmware güncelleme yeteneği hakkında bilgi
- Haberleşme (Sayfa 620) - ağ adresleri, haberleşme ara yüzleri fiziksel özellikleri ve haberleşme istatistikleri hakkında bilgi
- Tanılama (Sayfa 616) – tanılama arabelleği
- Değişken (Sayfa 620) - CPU değişkenleri ve I/O, adres veya PLC etiketi adıyla erişilebilir
- Dosya tarayıcı (Sayfa 622) - CPU veya bellek kartında dahili olarak depolanan tarif dosyalarına veya veri günlüğüne erişmek için kullanılan tarayıcı
- İndeks (Sayfa 613) - standart Web sayfaları girmek için giriş sayfası
- Oturum (Sayfa 610) - Farklı bir kullanıcı olarak oturum açma veya oturum kapama sayfası. (bir oturum penceresinin PC standart Web sayfalarının her sayfasında kullanılabilir olduğunu ancak oturum sayfasının bir mobil cihazdan oturum için gerekli olduğunu unutmayınız.)

Örneğin, "<http://ww.xx.yy.zz/communication.html>" girerseniz, tarayıcı iletişim sayfasını görüntüler.

## Güvenli erişim

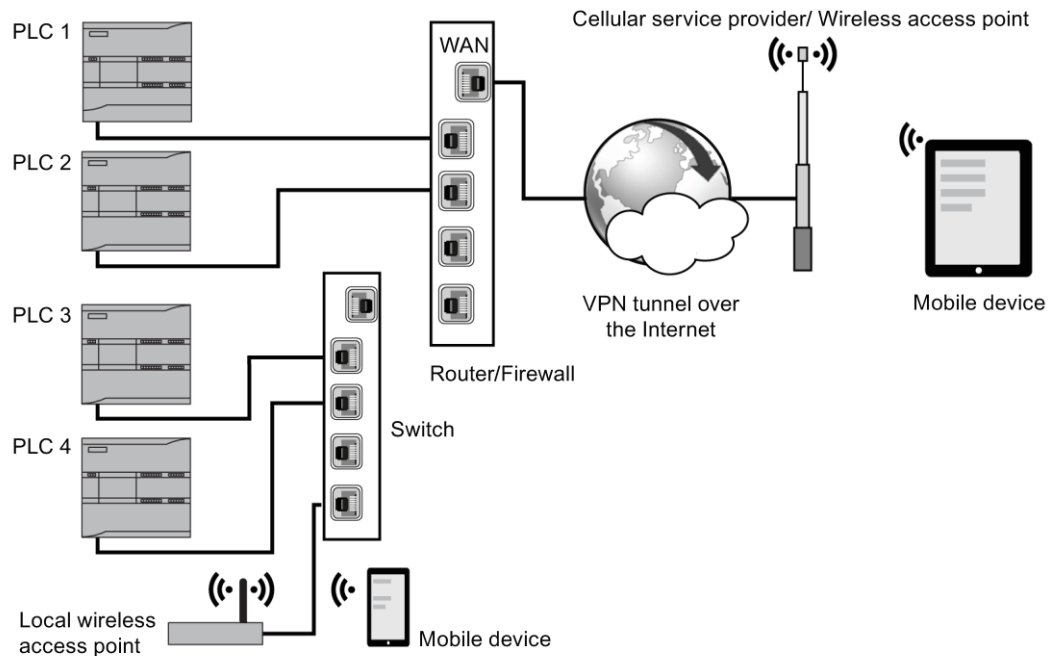
Korumalı ağınızın dışında bir yerden S7-1200 PLC Web sunucusuna bağlanmak için güvenli bir Sanal Özel Ağ (VPN) kullanınız. Standart Web sayfalarına güvenli erişim için <http://> yerine <https://> kullanabilirsiniz. <https://> ile S7-1200 bağlandığınızda, Web sitesi, bir dijital sertifika ile oturumu şifreler. Web sunucusu veri güvenli şekilde iletir ve herkesin görüntülemesi için erişilebilir değildir. Genellikle standart Web sayfalarına geçmek için "Evet" ile teyit ettiğiniz bir güvenlik uyarısı alırsınız. Her bir güvenli erişime sahip güvenlik uyarısını önlemek için, Web tarayıcınıza (Sayfa 666) Siemens yazılım sertifikasını import edebilirsiniz.

İsterseniz STEP 7 (Sayfa 603) deki PLC'yi sadece güvenli erişim ([https](https://)) ile erişilebilir olacak şekilde yapılandırabilirsiniz.

## 11.4 Bir mobil cihazdan Web sayfalarına erişme

Bir mobil cihazdan S7-1200'e erişmek için PLC'nizi internete veya yerel kablosuz erişim noktasına bağlayan bir ağa bağlamanız gerekir. Bir mobil cihazı S7 1200 PLC Web sunucusuna bağlamak için güvenli bir Sanal Özel Ağ (VPN) kullanınız. Bir PLC'nin IP adresini, mobil cihazın buna internetten erişebileceği adresle eşleştirmek için kablosuz yönlendiricideki port yönlendirmeyi kullanabilirsiniz. Port yönlendirmeyi yapılandırmak için, yönlendiricinizin yazılım yapılandırması yönergelerini izleyiniz. Yönlendiricinizin desteklediği kadar çok PLC'leri ve anahtarlama cihazlarını bağlayabilirsiniz.

Port yönlendirme olmadan, bir PLC ye sadece yerel olarak ve kablosuz sinyalin kapsamındayken bağlanabilirsiniz.



Bu örnekte, yerel kablosuz erişim noktası menziline olan bir mobil cihaz, PLC 3 ve PLC 4'e IP adresleri üzerinden bağlanabilir. Yerel kablosuz sinyalin kapsamı dışında kalan internetten, mobil cihaz PLC 1 ve PLC 2'ye her PLC için port yönlendirmesi yapılmış adresi kullanarak bağlanabilir.

Standart Web sayfalarına erişmek için bir hücresel hizmet veya kablosuz erişim noktasına erişiminiz olmalıdır. İnternet'ten bir PLC'ye erişmek için mobil cihazınızın Web tarayıcısındaki port yönlendirmesi yapılmış adresi giriniz. Örneğin <http://ww.xx.yy.zz:pppp> veya <https://ww.xx.yy.zz:pppp>. Burada ww.xx.yy.zz yönlendirici adresi ve pppp belirli bir PLC için port atamasıdır.

Bir yerel kablosuz erişim noktası üzerinden erişim için CPU'nun IP adresini giriniz: <http://ww.xx.yy.zz> veya <https://ww.xx.yy.zz>. Bir PC'den Web sayfalarına erişimde açıklandığı gibi belirli bir Web sayfasına isimle geçebilirsiniz (Sayfa 606).

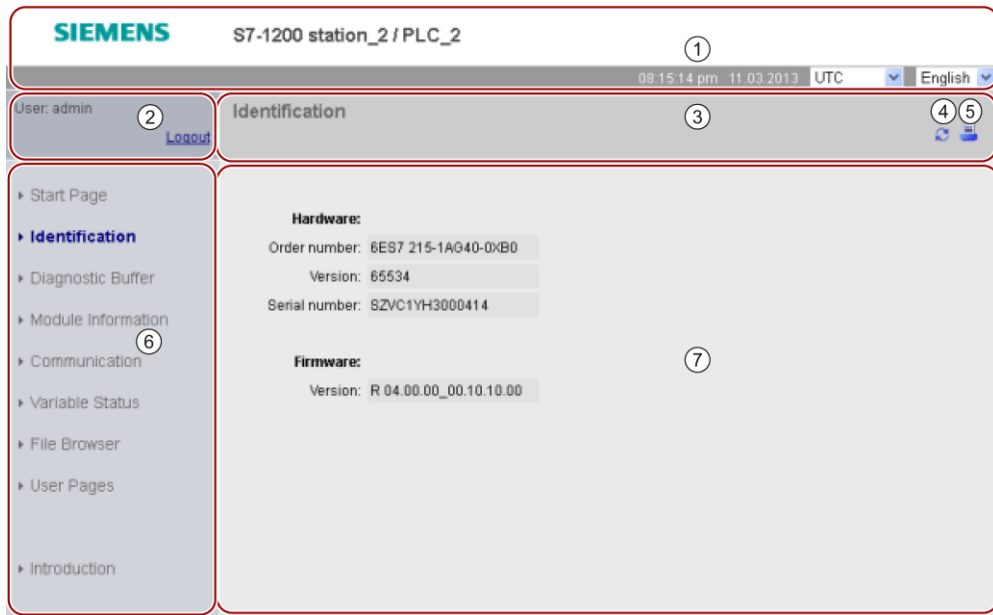
Eğer bunu yapmayı seçerseniz unutmayınız ki yalnızca güvenli erişim (https) tarafından erişilebilir olması için PLC'deki STEP 7'yi (Sayfa 603) yapılandırabilirsiniz.



## 11.5 Standart Web sayfaları

### 11.5.1 Standart Web sayfalarının yerleşim düzeni

Standart Web sayfalarının her birinin gezinme bağlantıları ve sayfa kontrolleri ile ortak bir yerleşim düzenleri vardır. Bir Sayfayı ister PC den ister mobil cihazdan görüntüleyin, her sayfa aynı içerik alanına sahiptir, ancak yerleşim düzeni ve geçiş kontrolleri cihazın ekran boyutu ve çözünürlüğüne göre değişir. Standart bir Web sayfasının standart bir PC'deki veya geniş mobil cihazdaki düzeni şu şekilde görünür:



- ① PLC Yerel saat veya UTC zamanını görüntülemek için seçicili Web sunucu başlığı ve görüntüleme dili için bir seçici (Sayfa 142)
- ② Oturum açma veya kapama
- ③ Görüntülediğiniz sayfa adı ile Standart Web sayfası başlığı. Bu örnek, CPU tanılama sayfasıdır. Modül bilgileri gibi standart Web sayfalarının bazıları, bu tür çoklu ekrana ulaşılabilirse burada da bir gezinti yolu gösterir.
- ④ Yenile simgesi: otomatik yenilemesi olan sayfalar için, otomatik yenileme fonksiyonunu devreye sokar veya devre dışı bırakır; otomatik yenilemesi olmayan sayfalar için, sayfanın mevcut verilerle güncellenmesine neden olur
- ⑤ Yazdır simgesi: görüntülenen sayfadaki mevcut bilgilerin yazdırılabilir sürümünü hazırlar ve görüntüler
- ⑥ Başka bir sayfaya geçmek için gezinti alanı
- ⑦ Görüntülediğiniz belirli bir standart Web sayfası için içerik alanı. Bu örnek CPU tanılama sayfasıdır.

## Mobil cihaz düzeni

Geniřlięi 768 pikselden az olan bir cihaz üzerinde, Web sunucusu her sayfanın bir mobil sürümünü görüntüler. Sayfa gezinti alanını, giriş alanını ve başlık alanını atlar ve geriye ilerleyen ve Web sayfaları aracılığıyla iletmek için düğmeleri ve Navigasyon sayfasına götüren bir Ana sayfa düğmesi içerir. Ayrıca gezinti için mobil cihazınızla birlikte verilen gezinme denetimlerini kullanabilirsiniz. Örneęin, ekran geniřlięi 768 pikselden az olan bir mobil cihazda tanılama sayfası ařaęıdaki gibi düřey yönde gözükür:



Unutmayınız ki bu bölümdeki standart Web sayfası çizimleri standart PC Web sayfası görünümünü temsil eder. Her standart Web sayfası eşdeęer bir mobil sayfa görünümüne sahiptir.

### 11.5.2 Oturum açma ve kullanıcı ayrıcalıkları

PC standart Web sayfalarının her biri gezinti bölmesinin üstünde bir giriş penceresi sağlar. Alan probleminin dolaylı, mobil Web sayfaları ayrı bir giriş sayfası sağlar.S7-1200 çeřitli erişim seviyeleri (ayrıcalıklar) ile birden fazla kullanıcı giriři destekler:

- Sorğu tanılama
- Okuma etiketi
- Yazma etiketi
- Okuma etiketi durumu
- Yazma etiketi durumu
- Açık kullanıcı tanımlı sayfalar
- Kullanıcı tanımlı sayfalara yazma
- Dosyaları okuma
- Dosya yazma/silme
- Çalışma modunu deęiřtirme
- Flash LED'leri
- Firmware güncellemesi yapma

Erişim düzeyleriyle (ayrıcılıklar) ilişkili kullanıcı rollerini ve CPU'nun aygıt yapılandırması STEP 7'deki Web sunucusu kullanıcı yönetimi özelliklerindeki şifreleri (Sayfa 604) yapılandırınız.

## Oturum açma

Genellikle, standart Web sayfalarında verileri görüntülemek için oturum açmak gerekmez. STEP 7'nin bir "herkes" kullanıcısına varsayılan olarak sağladığı, standart Web sayfalarının tümünü görüntüleme ayrıcalıklarına sahiptir. Örneğin, deneticinin çalışma modunu değiştirme, belleğe değerleri yazma ve CPU firmware güncellemesi gibi belirli eylemleri gerçekleştirmek için gereken ayrıcalıklara sahip olmanız gerekir. Unutmayınız ki Eğer CPU'nun koruma düzeyini "tam koruma (erişim yok)" olarak ayarlarsanız, ardından "herkes" kullanıcısı Web sunucusuna erişemez.



Oturum açma çerçevesi bir PC ya da geniş bir mobil cihazdan görüntülendiğinde her standart Web sayfasının sol üst köşesine yakındır.

Oturum açma sayfası küçük mobil cihazlarda ayrı bir sayfadır ve ana sayfadan seçilebilir.

Giriş yapmak için şu adımları izleyiniz:

7. Kullanıcı adı alanına kullanıcı adını giriniz.
8. Şifre alanına kullanıcı şifresini giriniz.

Otuz dakikalık faaliyetsizlikten sonra oturumunuz zaman aşımına uğrar. O anki yüklü sayfa sürekli yenileniyor ise, oturumunuz zaman aşımına uğramaz.

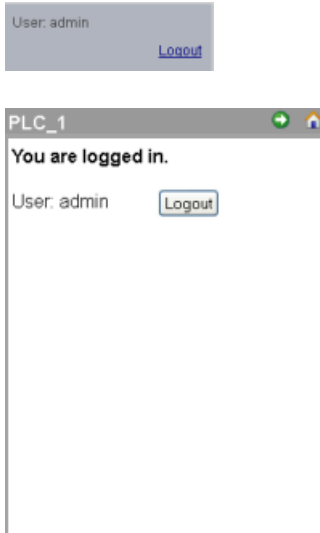
**! UYARI****Web sunucusuna CPU aracılığıyla yetkisiz erişim**

CPU'ya yetkisiz erişim veya PLC değişkenlerini geçersiz değerlerle değiştirmek çalışma sürecini bozabilir ve ölüm, ağır yaralanma ve/veya maddi hasara neden olabilir.

Web sunucusunu etkinleştirme, yetkili kullanıcıların işletim mod değişikliklerini gerçekleştirmesine, PLC veri yazdırmasına ve firmware güncellemelerini gerçekleştirmesine izin verdiği için, Siemens aşağıdaki güvenlik uygulamalarını gözlemlemenizi önerir:

- Yalnızca HTTPS protokolü ile Web sunucusuna erişimi etkinleştirme.
- Güçlü bir parola ile Şifre koruması Web sunucusu kullanıcı kimlikleri (Sayfa 190). Güçlü parolalar, uzunluğu en az sekiz karakter olan karışık harfler, sayılar ve özel karakterlerdir; bir sözlükte bulunabilecek sözcükler, isimler veya kişisel bilgilerden elde edilebilecek tanımlamalar değildir. Şifreyi gizli tutun ve sık sık değiştiriniz.
- Program mantığınızdaki değişkenler üzerinde hata denetimi ve aralık-kontrolü yapın çünkü Web sayfası kullanıcıları PLC değişkenlerini geçersiz değerlerle değiştirebilir.

Oturum açarken herhangi bir hata ile karşılaşırsanız, giriş sayfasına (Sayfa 613) dönünüz ve Siemens güvenlik sertifikasını (Sayfa 666) indiriniz. Daha sonra hatasız giriş yapabilirsiniz.

**Oturumu kapatma**

PC veya geniş bir mobil cihazdan sayfayı görüntülerken herhangi bir sayfadan çıkış yapmak için sadece "Oturum kapa" bağlantısına tıklayınız.

Küçük bir mobil cihazdan, Ana sayfadan giriş / çıkış sayfasına gidin ve "Çıkış" düğmesine dokununuz.

"Herkes" kullanıcı yetkilerine göre, giriş yapmadığınız zaman bile standart Web sayfalarına erişmeye ve onları görüntülemeye devam edebilirsiniz, ancak yüksek ayrıcalıklar gerektiren eylemleri gerçekleştiremezsiniz. Standart Web sayfası açıklamalarının her biri ek ayrıcalıklar gerektiren eylemleri (varsa) tanımlar.

**Not****Web sunucuyu kapatmadan önce oturum kapatma**

Web sunucusuna oturum açtıysanız, Web tarayıcınızı kapatmadan önce oturumu kapattığınızdan emin olunuz. Web sunucusu eşzamanlı maksimum yedi oturumu destekler.

### 11.5.3 Giriş

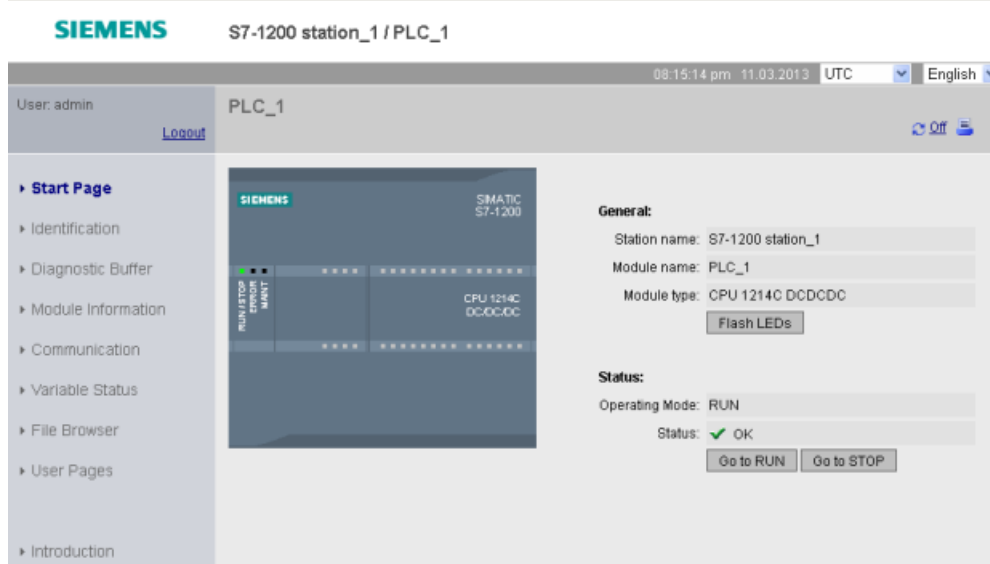
Giriş sayfası S7-1200 standart Web sayfalarına giriş için karşılama ekranıdır.



Bu sayfadan, S7-1200 standart Web sayfalarına erişmek için "Enter" e tıklayınız. Ekranın üst kısmında yararlı Siemens Web siteleri bağlantıları olduğu gibi Siemens güvenlik sertifikası (Sayfa 666) da indirebileceğiniz bir bağlantı vardır.

## 11.5.4 Başlangıç

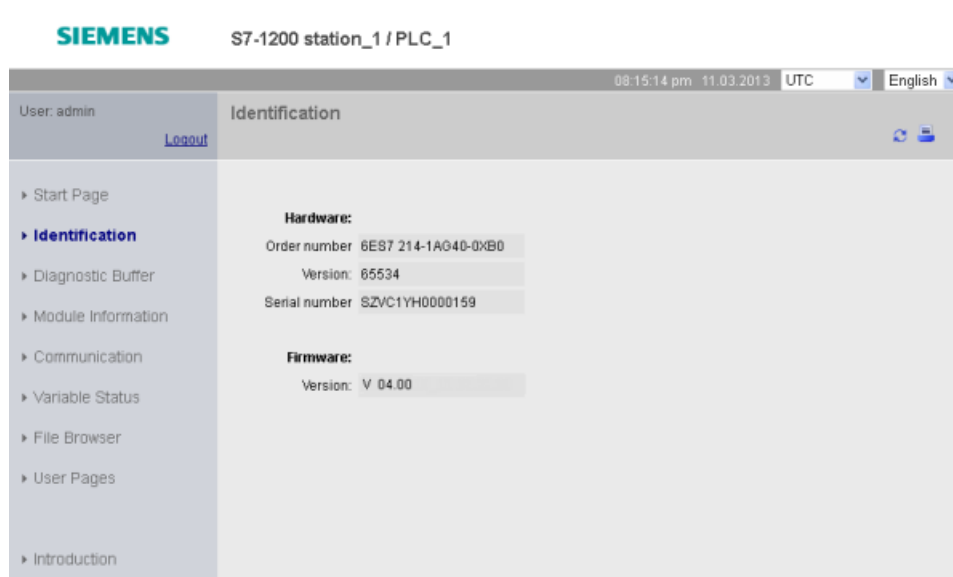
Başlangıç sayfası, bağlandığınız CPU'nun bir temsilini görüntüler ve o CPU ile ilgili genel bilgileri listeler. Bu özel ayrıcalıkları ile (Sayfa 610) oturum açtıysanız, CPU'nun çalışma modunu değiştirmek ve LED'leri yakmak için butonları kullanabilirsiniz.



## 11.5.5 Tanıtım (Kimlik)

Tanıtım sayfası, CPU'yu tanımlayan karakteristikleri görüntüler:

- Seri numarası
- Sipariş numarası
- Versiyon bilgisi



Tanıtım sayfasını görüntüleme "sorgu tanılama" ayrıcalığı gerektirir. Varsayılan olarak "Herkes" kullanıcısı oturum açmadan bu ayrıcalığa sahiptir.

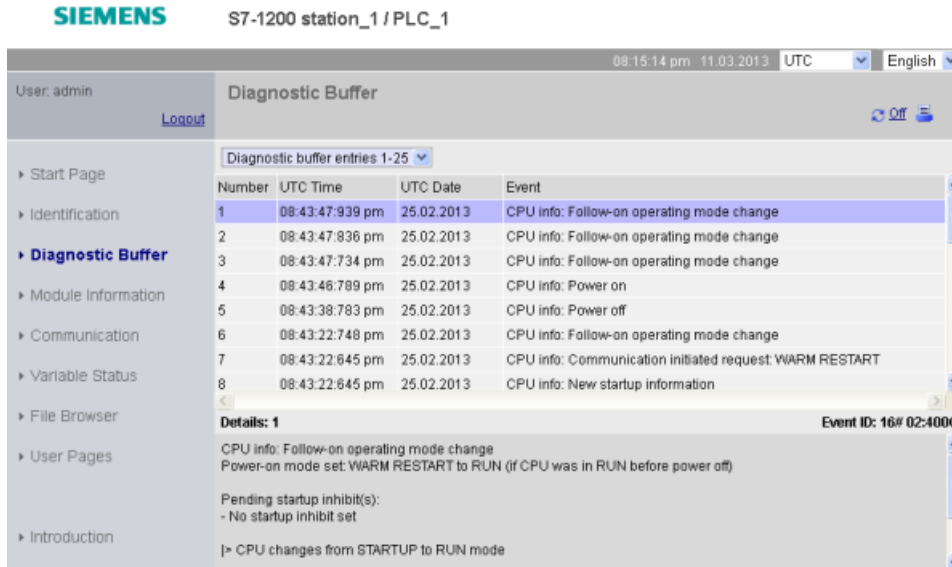
**Ayrıca bakınız**

Web sunucusu kullanıcılarını yapılandırma (Sayfa 604)

## 11.5.6 Tanılama arabelleği

Tanılama arabellek sayfası tanılama olaylarını görüntüler. Soldaki seçiciden, ya 1'den 25'e ya da 26'dan 50'ye, tanılama arabellek girişlerinin hangi aralıkta görüntüleneceğini seçebilirsiniz. Sağdaki seçiciden, zamanı, UTC zamanları veya PLC yerel zamanları olarak görüntülemeyi seçebilirsiniz. Sayfanın üst kısmı tanılama girdilerini olayın meydana geldiği saat ve tarihle görüntüler.

Sayfanın alt kısmındaki girdi hakkında detaylı bilgi göstermek üzere sayfanın üst kısmından herhangi bir girdi seçebilirsiniz.



SIEMENS S7-1200 station\_1 / PLC\_1

08:15:14 pm 11.03.2013 UTC English

User: admin Logout

Diagnostic Buffer

Diagnostic buffer entries 1-25

Number	UTC Time	UTC Date	Event
1	08:43:47:939 pm	25.02.2013	CPU info: Follow-on operating mode change
2	08:43:47:836 pm	25.02.2013	CPU info: Follow-on operating mode change
3	08:43:47:734 pm	25.02.2013	CPU info: Follow-on operating mode change
4	08:43:46:789 pm	25.02.2013	CPU info: Power on
5	08:43:38:783 pm	25.02.2013	CPU info: Power off
6	08:43:22:748 pm	25.02.2013	CPU info: Follow-on operating mode change
7	08:43:22:645 pm	25.02.2013	CPU info: Communication initiated request: WARM RESTART
8	08:43:22:645 pm	25.02.2013	CPU info: New startup information

Details: 1 Event ID: 16# 02:400C

CPU info: Follow-on operating mode change  
Power-on mode set: WARM RESTART to RUN (if CPU was in RUN before power off)

Pending startup inhibit(s):  
- No startup inhibit set

> CPU changes from STARTUP to RUN mode

Modül bilgisi sayfasını görüntülemek "sorgu tanılama" ayrıcalığı (Sayfa 604) gerektirir. Varsayılan olarak "Herkes" kullanıcısı oturum açmadan bu ayrıcalığa sahiptir.

## 11.5.7 Modül bilgisi


Modül bilgi sayfası, yerel raftaki tüm modüller hakkında bilgi sağlar. Ekranın üst bölümü modüllerin bir özetini gösterir ve alt bölüm seçilen modülün durumunu, kimliğini ve firmware bilgilerini gösterir. Modül Bilgi sayfası, bir firmware güncellemesi gerçekleştirme kabiliyeti sağlar.

Modül Bilgi sayfasını görüntülemek, "sorgu tanılama" ayrıcalığı (Sayfa 604) gerektirir. Varsayılan olarak "Herkes" kullanıcısı oturum açmadan bu ayrıcalığa sahiptir.



## Modül bilgisi: Durum sekmesi

Modül bilgi sayfasının alt bölümündeki durum sekmesi üst bölümde seçili olan modülün mevcut durumunun bir açıklamasını görüntüler.



**SIEMENS** S7-1200 station\_1 / PLC\_1

User: admin Logout

05:08:02 am 18.01.1970 UTC English

Module Information

S7-1200 station\_1 - S7-1200 station\_1 Slot Filter

Slot	Status	Name	Order number	I address	Q address	Comment
1	✓	PLC_1	Details 6ES7 214-1AG40-0XB0	0	0	
2	✓	DI16/DO16 x 24VDC_1	Details 6ES7 223-1BL30-0XB0	8	8	

[Status](#) | [Identification](#) | [Firmware](#)

The module is parameterized, and it exchanges data with the CPU. The module did not detect any parameter assignment errors, channel errors, or hardware related errors.

## Not

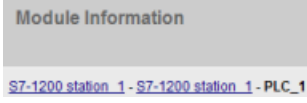
Mobil cihaz modül bilgi sayfası "I adresi", "Q adresi" ve "Yorum" bilgilerini kimlik sekmesinden ziyade ana modül bilgileri çizelgesinde sütun olarak görüntüler.

## Alt veriye ulaşma

Belirli bir modülün, modül bilgisinin ayrıntısına ulaşmak için üst bölümden bir bağlantı seçebilirsiniz. Alt modülleri olan modüllerin her alt modül için ayrı bağlantıları vardır. Görüntülenen bilginin türü seçilen modüle göre değişir. Örneğin, modül bilgi iletişimi başlangıçta SIMATIC 1200 istasyonunun adını, durum göstergesini ve açıklamasını görüntüler. Eğer CPU'nun detayına inerseniz, modül bilgisi CPU modelinin sağladığı dijital ve analog giriş ve çıkışların isimlerini (örneğin, "DI14/DO10", "AI2"), I / O için adres bilgilerini, durum göstergelerini, slot numaralarını ve açıklamaları görüntüler.

Slot	Status	Name	Order number	I address	Q address	Comment
1.16	✓	HSC_1	Details 6ES7 214-1AG40-0XB0	1000	---	
1.17	✓	HSC_2	Details 6ES7 214-1AG40-0XB0	1004	---	
1.18	✓	HSC_3	Details 6ES7 214-1AG40-0XB0	1008	---	
1.19	✓	HSC_4	Details 6ES7 214-1AG40-0XB0	1012	---	
1.20	✓	HSC_5	Details 6ES7 214-1AG40-0XB0	1016	---	
1.21	✓	HSC_6	Details 6ES7 214-1AG40-0XB0	1020	---	
1.2	✓	AI2_1	Details 6ES7 214-1AG40-0XB0	64	---	
1.1	✓	DI14/DO10_1	Details 6ES7 214-1AG40-0XB0	0	0	
1.32	✓	Pulse_1	Details 6ES7 214-1AG40-0XB0	---	1000	

Siz detaya indikçe, modül bilgi sayfası sizin takip ettiğiniz yolu gösterir. Bir üst seviyeye dönmek için bu yolda herhangi bir bağlantıyı tıklayabilirsiniz.



## Sıralama alanları

Listede, birden çok modül görüntülediği zaman, bir alanı azalan ya da artan şekilde sıralamak için bu alanın sütun başlığına tıklayabilirsiniz.

Not: Bu özellik, Çince Modül Bilgi sayfası için henüz mevcut değil.

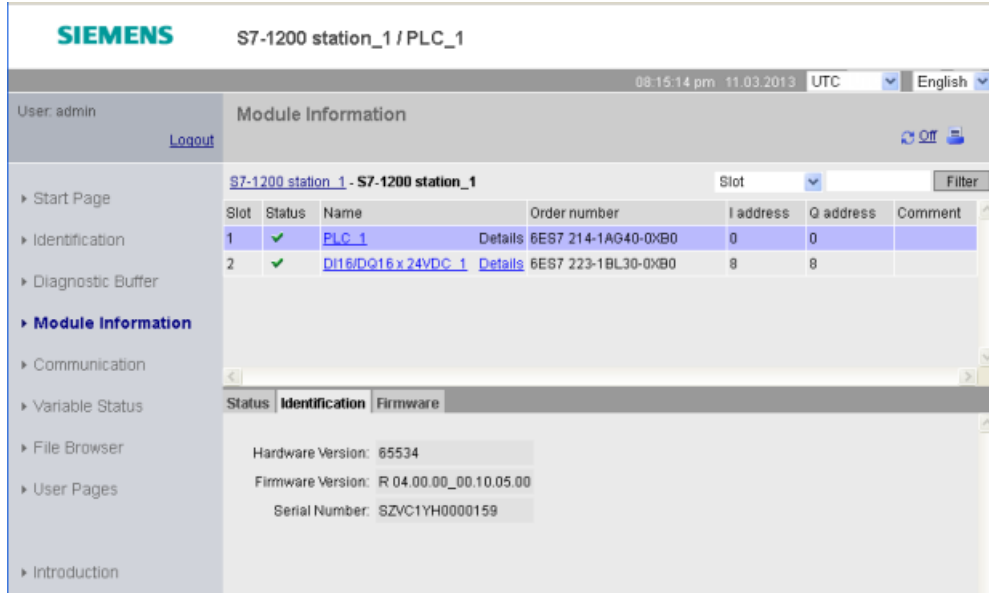
Name	
AI2_1	<a href="#">Details</a>
DI14/DO10_1	<a href="#">Details</a>
HSC_1	<a href="#">Details</a>
HSC_2	<a href="#">Details</a>
HSC_3	<a href="#">Details</a>
HSC_4	<a href="#">Details</a>
HSC_5	<a href="#">Details</a>
HSC_6	<a href="#">Details</a>
Pulse_1	<a href="#">Details</a>

## Modül bilgisini filtreleme

Modül bilgi listesindeki herhangi bir alanı filtre edebilirsiniz. Açılır listeden, filtrelemek istediğiniz verilerin alan adını seçiniz. İlgili metin kutusuna metin giriniz ve Filtre bağlantısını tıklayınız. Liste, sizin filtreleme kriterinize uygun modülleri göstermek için güncellenir.

## Modül bilgisi: Kimlik sekmesi

Kimlik sekmesi, seçilen modülün seri numarası ve sürüm numaralarını görüntüler.



Slot	Status	Name	Order number	I address	Q address	Comment
1	✓	<a href="#">PLC_1</a>	6ES7 214-1AG40-0XB0	0	0	
2	✓	<a href="#">DI16/DO16x24VDC_1</a>	6ES7 223-1BL30-0XB0	8	8	

Hardware Version: 65534  
 Firmware Version: R 04.00.00\_00.10.05.00  
 Serial Number: SZVC1YH0000159

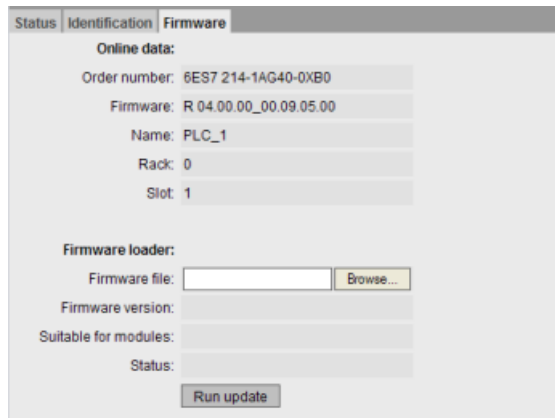
## Modül bilgisi: Firmware sekmesi

Modül bilgi sayfasının firmware sekmesi, seçilen modülün firmware i hakkındaki bilgisini görüntüler. Eğer "firmware güncelleştirmeyi gerçekleştirme" ayrıcalığına (Sayfa 604) sahipseniz, firmware güncellemesini destekleyen CPU veya diğer modüllerin firmware güncellemesini yapabilirsiniz.

### Not

Sadece Firmware Güncelleme özelliğine sahip olan ve sürüm 3.0 ve üzeri olan S7-1200 CPU'ları güncelleyebilirsiniz.

Web sunucusu firmware güncelleştirmesini gerçekleştirmek için "https" protokolünü kullanır.



CPU, firmware güncellemesini gerçekleştirmek için STOP modunda olmalıdır. CPU STOP modunda olduğunda, geçiş yapmak için Gözet düğmesini tıklayınız ve bir firmware dosyasını seçiniz. Firmware güncellemeleri müşteri destek Web sitesinde (<http://support.automation.siemens.com>) mevcuttur.

Güncelleştirme sırasında, sayfa, güncellenmenin ilerlediğini gösteren bir mesaj görüntüler. Güncelleme tamamlandıktan sonra, sayfa güncellenmiş firmware sipariş numarası ve sürüm numarasını görüntüler. Eğer CPU veya bir sinyal kartı için firmware güncellenmiş ise, Web sunucusu CPU'yu yeniden başlatır.

### Not

STEP 7'den (Sayfa 791) ya da bir bellek kartı kullanarak bir firmware güncellemesi yapabilirsiniz (Sayfa133).

## 11.5.8 İletişim

İletişim sayfası MAC adresi, IP adresi ve CPU'nun IP ayarları da dahil olmak üzere bağlı olan CPU'nun parametrelerini görüntüler.



İletişim sayfasını görüntülemek "sorgu tanılama" ayrıcalığı gerektirir.

## 11.5.9 Değişken Durumu

Değişken Durumu sayfası, size CPU'nuzdaki herhangi bir I/O veya bellek verisini görüntülemenize olanak sağlar. Adresi doğrudan (örneğin I0.0 gibi), bir PLC etiket adıyla, ya da belirli bir veri bloğundaki etiketle girebilirsiniz. Veri bloğu etiketleri için, veri blok adını çift tırnak içine alabilirsiniz. Her izleme değeri için bir veri görüntüleme biçimi seçebilirsiniz. Sayfanın sınırları dahilinde istediğiniz kadar çok değer girerek ve o değerleri belirterek devam edebilirsiniz. Sayfanın sağ üst köşesindeki "Off" ikonuna tıklamadığınız sürece izleme değerleri otomatik olarak gösterilir ve varsayılan olarak yenilenir. Yenileme devre dışı bırakıldığında, otomatik yenilemeyi yeniden etkinleştirmek için "On"a tıklayabilirsiniz.

Değişken Durum sayfasını görüntülemek "değişken durumu okuma" ayrıcalığı gerektirir.

"Değişken durumu yazma" ayrıcalığı bulunan bir kullanıcı olarak oturum açarsanız, veri değerlerini de değiştirebilir. Ayarlamak istediğiniz herhangi bir değeri, uygun "Modify Value" alanına giriniz. Bir değeri CPU'ya yazmak için o değer yanındaki "Go" butonuna tıklayınız. Ayrıca birden değerleri giriniz ve CPU tüm değerleri yazmak için "Bütün Değerleri Değiştir" tıklayabilirsiniz. Eğer "değişken durumu yazma" ayrıcalığınız varsa değiştirmek için düğmeleri ve sütun etiketleri sadece görünür.

SIEMENS S7-1200 station\_1 / PLC\_1

08:15:14 pm 11.03.2013 UTC English

User: admin Logout

Variable Status

Enter the address of a tag here which you want to monitor/modify

Address / Display Format	Monitor Value / Modify Value	Modify
Q0.1 BOOL	true	Go
I0.1 BOOL	false	Go
Conveyor_speed DEC	145	Go
Mixer_On BOOL	false	Go
*Data_block_1*.location CHARACTER	16#00000000	Go
Tag_1 FLOATING_POINT	17.2	Go
New variable BIN		

Monitor Value Modify All Values

Değişken Durum sayfasından ayrılır ve geri dönerseniz, Değişken Durumu sayfası girdilerinizi tutmaz. O sayfayı yer imlerine ekleyebilirsiniz ve aynı girdileri görmek için o yer imine dönebilirsiniz. Eğer sayfayı yer imlerine eklemediyseniz, değişkenleri yeniden girmeniz gerekir.

## Not

Standart Değişken Durumu sayfasını kullanırken, aşağıdaki konuların farkında olunuz:

- Tüm string değişikliklerini tek tırnak içine alınız.
- Değişken Durum sayfası, aşağıdaki karakterlerden herhangi birini içeren etiketleri izleyemez ya da değiştiremez: &, <, (, +, ,(comma), ., [, ], \$, veya %. Örneğin, şu etiketi izleyemezsiniz "Clock\_2.5Hz".
- Değişken Durumu sayfası 198 karakterden daha uzun bir string'i değiştirmenize izin vermez.
- Değişken Durum sayfasında bir Real veya LReal veri tipi için bir değer girmek için üstel gösterim kullanılırken:
  - Pozitif üs (+3,402823e+25 gibi) ile bir gerçek sayı değeri (Real veya LReal) girmek için, değeri aşağıdaki biçimlerden birinde giriniz:
    - +3.402823e25
    - +3.402823e+25
  - Gerçek sayı değerini (Real veya LReal) negatif bir üs ile (+3,402823e-25 gibi) girmek için değeri aşağıdaki gibi giriniz:
    - +3.402823e-25
  - Üstel gösterimde gerçek değerın mantis kısmının bir ondalık noktası içerdiğinden emin olunuz. Beklenmedik bir tamsayı değerine değer değişiklik bir ondalık nokta sonuçlarını içerir başarısızlık. Örneğin, -1e8'den ziyade -1.0e8 giriniz.
- Değişken Durumu sayfası bir LReal değeri (ondalık noktanın konumuna bakılmaksızın) sadece 15 basamak destekler. 15'den fazla basamak girmek bir yuvarlama hatası oluşturur.

Değişken durum sayfasında sınırlamalar:

- Sayfa başına değişken girişlerin sayısı maksimum 50'dir.
- Değişken durum sayfasıyla ilişkili URL için maksimum karakter sayısı 2083'dür. Tarayıcınızın adres çubuğunda, o anki değişken sayfasını temsil eden URL'yi görebilirsiniz.
- Gerçek CPU değerleri, tarayıcı tarafından yorumlanan geçerli ASCII karakterleri değilse karakter görüntüleme formatında, sayfa onaltılık değerleri görüntüler.

---

**Not**

Bir etiket ismi, değişken durum sayfasında girdi olarak reddedilen özel karakterler görüntülense, etiket ismini çift tırnak içine alabilirsiniz. Çoğu durumda, Değişken durum sayfası daha sonradan etiket adını tanıyacaktır.

---

**Ayrıca bakınız**

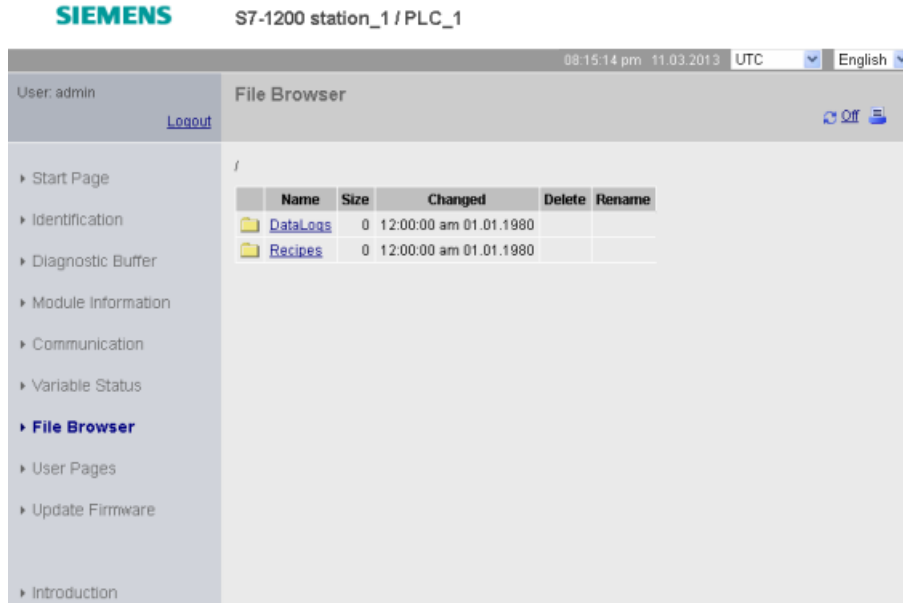
Oturum açma ve kullanıcı ayrıcalıkları (Sayfa 610)

**11.5.10 Dosya tarayıcısı**

Dosya Tarayıcı sayfası CPU'nun dahili yükleme belleğindeki veya bellek kartındaki (harici yükleme belleğindeki) dosyalara erişim sağlar. Dosya tarayıcısı sayfası başlangıçta yük belleğindeki "Veri günlükleri" ve "Tarifler" klasörlerini içeren kök klasörünü görüntüler; aynı zamanda eğer bir bellek kartı kullanıyorsanız, oluşturmuş olabileceğiniz diğer klasörleri de görüntüler.

Dosyalara ve klasörlere erişmek için sahip olduğunuz dosya erişimi türü sizin kullanıcı ayrıcalıklarınıza (Sayfa 604) bağlıdır. "dosyaları okuma" ayrıcalığına sahip herhangi bir kullanıcı dosya ve klasörleri dosya tarayıcısı ile görüntüleyebilir. Oturum açma ayrıcalığınız ne olursa olsun veri günlüğü veya Tarifler klasörünü silemezsiniz, ancak bellek kartında özel klasörler oluşturduysanız, bu klasörleri "dosyaları yazma / silme" ayrıcalığına sahip bir kullanıcı olarak oturum açtığınızda silebilirsiniz.

Klasördeki bireysel dosyalara ulaşmak için bir klasöre tıklayınız.



## Veri günlükleri

"Veri Günlükleri" klasöründen, veri günlük dosyalarından herhangi birini açabilirsiniz. "dosyaları sil/yaz" ayrıcalığı (Sayfa 604) ile oturum açtıysanız, dosyaları silebilir, yeniden adlandırabilir ve yükleyebilirsiniz. Veri günlük dosyaları, virgülle ayrılmış değerler (CSV) dosya biçimindedir. Başka bir program veya Microsoft Excel'de (varsayılan) onları açabilir veya bilgisayara kaydedebilirsiniz. PLC programının açık bir veri günlüğü varsa bu açık iken, dosya tarayıcı sayfasından bunun silmeyi unutmayınız. Dosya zaten varsa, bir veri günlük dosyası yükleyemezsiniz.

### Not

#### Veri günlükleri için zaman etiketleri

Web sunucusu, sayfanın üstündeki seçiminize bağlı olarak UTC zamanı veya PLC yerel saati ya da veri günlükleri için zaman etiketlerini görüntüler.

**SIEMENS** S7-1200 station\_1 / PLC\_1

08:15:14 pm 11.03.2013 UTC English

User: admin [Logout](#)

**File Browser**

/Data1.ogs/

Name	Size	Changed	Delete	Rename
MyDataLog.csv	120	08:53:14 pm 13.02.2013		<input type="text"/>
MyDataLog1.csv	120	09:14:10 pm 13.02.2013		<input type="text"/>

Directory operations:

**Not:** "dosyaları sil/ yaz" ayrıcalığı ile oturum açmadıysanız "Sil" ve "Yeniden adlandır" seçenekleri kullanılamaz.

### Not

Veri günlük dosyası ABD / İngiltere virgülle ayrılmış değerler (CSV) biçimindedir. ABD/İngiltere'ye göre olmayan sistemlerde, bunu belirli ayarlara sahip Excel'e aktarmanız gerekir (Sayfa 667).

## Tarif dosyaları

Veri günlükleri klasörü gibi, tarif klasörü, yükleme belleğinde bulunan herhangi bir tarif dosyasını görüntüler. Tarif dosyaları CSV formatındadır ve Microsoft Excel veya başka bir programda açabilirsiniz. Veri günlükleri gibi, tarif dosyaları, silme, değiştirme ve kaydetme, yeniden adlandırmak ya da yükleme yapmak için ayrıcalıkları değiştirmeniz gerekir.

## Dosya yükleme ve otomatik sayfa yenileme

Bir dosya yükleme başlar ise, yükleme işlemi, Dosya Tarayıcı Web sayfasında kaldığınız sürece devam eder. Web sunucusu sayfalarını on saniyede bir yenilemek için otomatik güncellemeyi etkinleştirdiyseniz her bir sayfa yenilemesi meydana geldiğinde dosya yükleme işleminin ilerleme sürecini görürsünüz. Örneğin 2 MB dosya yüklüyorsanız, dosya yükleme ilerledikçe bayt cinsinden 2500, 5000, 10000, 15000 ve 20000 de dosya boyutunu gösteren güncellemeleri görebilirsiniz.

Yükleme tamamlanmadan önce Dosya Tarayıcı sayfasından ayrılırsanız, tam dosyayı alamazsınız. Geriye döndüğünüzde, Dosya Tarayıcı sayfası, dosya adını ve yüklemenin durduğu zamandaki dosya boyutunu görüntüler. Bunun bir eksik dosya olduğunu belirten başka hiçbir belirti görmezsiniz. Tam dosyayı yüklediğinizden emin olmak için Görüntülenen dosya boyutu dosyanın gerçek boyutuna ulaşıncaya kadar, Dosya Tarayıcı sayfasında kalınız.




## Ek bilgiler

Veri günlük komutları ile programlama ve tarifleri içe aktarma (Sayfa 376)ve dışa aktarma (Sayfa 374) hakkında bilgi için Tarifler ve Veri günlükleri (Sayfa 369) bölümüne bakınız.

## 11.6 Kullanıcı tanımlı Web sayfaları

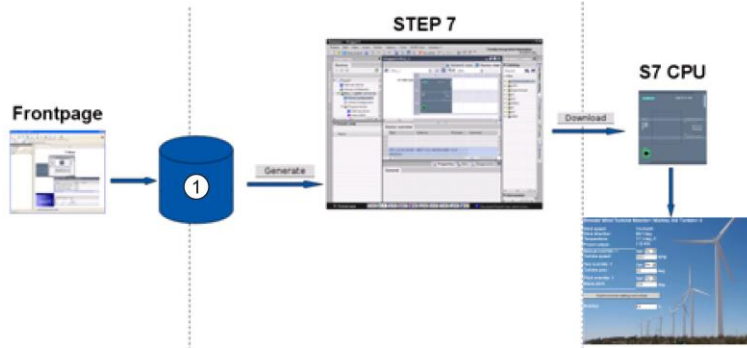
S7-1200 Web sunucusu, PLC den verileri dahil eden kendi uygulamanıza özgü HTML sayfaları oluşturmanız için araçlar sağlar.

 <b>UYARI</b>
<b>Kullanıcı-tanımlı Web sayfaları aracılığıyla CPU' ya yetkisiz erişim</b>
Kullanıcı-tanımlı Web sayfaları aracılığıyla CPU' ya yetkisiz erişim ölüm, ağır yaralanma ve / veya maddi hasara yol açabilecek proses çalışmasını bozabilir.
Kullanıcı-tanımlı Web sayfalarının Güvensiz kodlaması, siteler arası betik çalıştırma (XSS), kod enjeksiyonu ve diğerleri gibi güvenlik açıkları oluşturur.
Endüstriyel Güvenlik Web sitesinde bulunan Çalışma Kılavuzunda belirtildiği gibi, S7 1200 CPU'nuzu güvenli bir şekilde yükleyerek yetkisiz erişimden koruyunuz.

Seçtiğiniz HTML editörü kullanarak kullanıcı tanımlı Web sayfaları oluşturur ve standart Web sayfası menüsünden erişilebilen bu sayfaları CPU'ya indirebilirsiniz. Bu süreç çeşitli görevleri kapsar:

- Microsoft FrontSayfa (Sayfa 626) gibi bir HTML editör ile HTML sayfaları oluşturma
- HTML kodunda (Sayfa 627) HTML yorumlarına AWP komutlarını dahil etme: AWP komutlar Siemens in, CPU bilgilerine erişmek için sağladığı sabit bir komut grubu vardır.
- HTML sayfalarını okumak ve işlemek için STEP 7 yapılandırma (Sayfa 641)
- HTML sayfalarından bloklar oluşturma (Sayfa 641)
- HTML sayfalarını (Sayfa 642) kullanımını kontrol etmek için STEP 7 Programlama
- Derleme ve CPU blokları indirirken (Sayfa 643)
- PC'niz (Sayfa 644) kullanıcı-tanımlı Web sayfalarına erişim

Bu işlem aşağıda gösterilmiştir:



① Gömülü AWP komutlarına sahip HTML sayfaları

### Ayrıca bakınız

Endüstriyel Güvenlik Web sitesi (<http://www.siemens.com/industrialsecurity>)

### 11.6.1 HTML sayfası oluşturma

Web sunucusu ile kullanım amacıyla kendi HTML sayfalarınızı oluşturmak için seçtiğiniz yazılım paketini kullanabilirsiniz. HTML kodunuzun W3C'nin (Dünya Çapında Ağ Birliği) HTML standartlarına uyumlu olduğundan emin olunuz. STEP 7, HTML sözdiziminizin herhangi bir doğrulamasını yapmaz.

WYSIWYG de veya tasarım düzen modunda tasarım yapmanızı sağlayan bir yazılım paketi kullanabilirsiniz, ancak saf HTML biçiminde HTML kodunuzu düzenlemeniz gerekir. Çoğu Web yazarlık araçları bu tip bir düzenleme sağlar; Aksi takdirde, her zaman HTML kodunu düzenlemek için basit bir metin editörü kullanabilirsiniz. UTF-8 için sayfanın karakter setini ayarlamak için HTML sayfasına aşağıdaki satırı giriniz:

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

Ayrıca UTF-8 karakter kodlamada editörden dosyayı kaydettiğinizden emin olunuz.

HTML sayfalarındaki her şeyi STEP 7 veri blokları halinde derlemek için STEP 7 kullanınız. Bu veri blokları, derlenmiş Web sayfalarını içeren bir veya daha fazla fragman veri bloklarından ve Web sayfaları ekranını yönlendiren bir kontrol veri bloğundan oluşur. Kapsamlı HTML sayfaları grubunun, özellikle resimleri çok olanların, fragman DB'ler için önemli miktarda yükleme bellek alanına (Sayfa 644) ihtiyaç duyduğunu unutmayınız. CPU'nuzun dahili yükleme belleği kullanıcı-tanımlı Web sayfaları için yeterli değilse, harici yükleme belleği sağlamak için bir bellek kartı (Sayfa 124) kullanınız.

S7-1200 den gelen verileri kullanmak amacıyla HTML kodu programlamak için, AWP komutlarını HTML yorumları olarak (Sayfa 627) dahil ediniz. Bittiğinde, PC'nize HTML sayfalarını kaydediniz ve bunları kaydettiğiniz klasör yolunu not ediniz.

---

**Not**

AWP komutu içeren HTML dosyaları için dosya boyutu limiti 64 kilobayttır. Dosya boyutunuzu bu sınırın altında tutmalısınız.

---

### Kullanıcı-tanımlı Web sayfalarını yenileme

Kullanıcı tanımlı Web sayfaları otomatik olarak yenilenmez. Bu sayfayı yenileyecek şekilde HTML programlamak sizin tercihinizdir. PLC verileri görüntüleyen sayfaları için periyodik olarak yenileme veriyi güncel tutar. Veri girişi için formlar olarak hizmet veren HTML sayfaları için yenileme, verileri giren kullanıcıya engel olur. Tüm sayfayı otomatik olarak yenilemek istiyorsanız, HTML başlığına bu satırı ekleyebilirsiniz (Burada "10" yenilemeler arasındaki saniye değeri):

```
<meta http-equiv="Refresh" content="10">
```

Ayrıca, sayfayı veya veri yenilemeyi denetlemek için JavaScript veya diğer HTML tekniklerini kullanabilirsiniz. Bunun için, HTML ve JavaScript belgelerine bakınız.

### 11.6.2 S7-1200 Web sunucusu tarafından desteklenen AWP komutları

S7-1200 Web sunucusu, size aşağıdaki amaçlar için HTML yorumlar olarak kullanıcı-tanımlı Web sayfalarına gömdüğünüz AWP komutları sağlar:

- Değişkenleri Okuma (Sayfa 629)
- Değişkenleri yazma (Sayfa 630)
- Özel değişkenler okuma (Sayfa 631)
- Özel değişkenler yazma (Sayfa 633)
- Enum tipleri tanımlama (Sayfa 635)
- Enum tiplerine değişkenleri atama (Sayfa 635)
- Fragman veri bloklarını oluşturma (Sayfa 637)

## Genel sözdizimi

Bir değişkeni okuyan komutu dışında, AWP komutları aşağıdaki sözdizimi şeklindedir:  
`<!-- AWP_ <komut ismi ve parametreler> -->`

CPU'daki değişkenlere yazmak için tipik bir HTML formu komutları ile birlikte AWP komutlarını kullanabilirsiniz.

Aşağıdaki sayfalarda AWP komutlarının açıklamaları aşağıdaki yöntemleri kullanır:

- Parantez [ ] içindeki öğeler opsiyoneldir.
- Açılı parantez <> içindeki öğeler belirtilmesi gereken parametre değerleridir.
- Tırnak işaretleri, komutun değişmez bir parçasıdır. Bunlar gösterildiği gibi mevcut olması gerekir.
- Etiket veya veri bloğu adlarındaki Özel karakterlerden, kullanıma bağlı olarak, kaçınılmalı veya tırnak işaretleri (Sayfa 639) içinde olmalıdır.

Sayfalarınıza AWP komutları eklemek için bir metin editörü veya HTML düzenleme modunu kullanınız.

---

### Not

#### AWP komutlarının b eklenen sözdizimi

Bir AWP komuta formülasyonunda "`<!--`" den sonraki boşluk ve "`-->`" den önceki boşluk komutunun düzgün derlenmesi için gereklidir. Boşluk karakterlerinin çıkarılması derleyicinin doğru kod üretememesine neden olabilir. Derleyici bu durumda bir hata göstermez.

---

## AWP komut özeti

Her bir AWP komutunun kullanımıyla ilgili bilgiler takip eden konularda vardır, ama burada komutların kısa bir özeti verilmiştir:

### Değişkenleri okuma

`:=<Varname>:`

### Değişkenleri Yazma

`<!-- AWP_In_Variable Name='<Varname1>' [Use='<Varname2>'] ... -->`

Bu AWP komutu sadece Name ifadesindeki değişkenin yazılabilir olduğunu bildirir. HTML kodunuz, bir HTML formu içinde `<input>`, `<select>` veya diğer HTML ifadelerinden isme göre değişkene yazmaları gerçekleştirir.

### Özel değişkenler Okuma

`<!-- AWP_Out_Variable Name='<Type>:<Name>' [Use='<Varname>'] -->`

### Özel değişkenler Yazma

`<!-- AWP_In_Variable Name='<Type>:<Name>' [Use='<Varname>'] -->`

### Enum tipleri tanımlama

`<!-- AWP_Enum_Def Name='<Enum type name>' Values='<Value>', <Value>, ... ' -->`

### Enum tiplerini referanslama

`<!-- AWP_Enum_Ref Name='<VarName>' Enum='<EnumType>' -->`

**Fragmanlar oluştur**

```
<!-- AWP_Start_Fragment Name='<Name>' [Type=<Type>] [ID=<id>] -->
```

**Fragmanları içe aktarma**

```
<!-- AWP_Import_Fragment Name='<Name>' -->
```

**11.6.2.1 Değişkenleri okuma**

Kullanıcı tanımlı Web sayfaları değişkenleri (PLC etiketleri) CPU' dan okuyabilir.

**Sözdizimi**

```
:=<Varname>:
```

**Parametreler**

<Varname>	STEP 7 programınızdan bir PLC etiketi adı olabilen okunacak değişken, bir veri bloğu etiketi I / O veya adreslenebilir bellek. Bellek veya I / O adresleri veya diğer adlar (Sayfa 639) için etiket adı etrafında çift tırnak işaretleri kullanmayınız. PLC etiketleri için etiket adı etrafında çift tırnak işaretleri kullanınız. Veri bloğu etiketleri için, blok adını yalnızca çift tırnak içine alınız. Etiket adı, tırnak işaretleri dışındadır. Eğer veri bloğu adı ve bir veri blok numarası kullandığınızı unutmayınız.
-----------	---

**Örnekler**

```
:= "Conveyor_speed" : := "My_Data_Block".flag1 :  
:= I0.0 :  
:= MW100 :
```

**Örnek bir başka ad değişkeni okuyor**

```
<!-- AWP_Out_Variable Name='flag1' Use=' "My_Data_Block".flag1' -->  
:= flag1 :
```

**Not**

PLC etiketleri ve veri bloğu etiketleri için diğer adların tanımlanması, “değişken bir referans için bir takma ad (Sayfa 634) kullanma” konusu içinde açıklanmıştır.

Bir etiket adı veya veri bloğu adı özel karakterler içeriyorsa, ek tırnak işareti ya da özel karakterler (Sayfa 639) içeren ve “etiket isimlerinin işlenmesi” konusunda açıklanan çıkış karakterleri kullanmanız gerekir.

**11.6.2.2 Değişkenleri Yazma**

Kullanıcı tanımlı sayfalar CPU'ya veri yazabilir. Bu, HTML sayfasından, CPU içindeki bir değişkenin yazılabilir olacak şekilde tanımlamak için bir AWP komutu kullanılarak gerçekleştirilir. Değişkenin, PLC etiket adı veya veri bloğu etiket adıyla belirtilmesi gerekir. Bir ifadede birden çok değişken adlarını bildirebilirsiniz. CPU' ya veri yazmak için, standart HTTP POST komutlarını kullanınız.

Tipik bir kullanım, yazılabilir CPU değişkenlerine karşılık gelen seçim listesi seçenekleri veya metin giriş alanları ile HTML sayfası içinde bir form tasarlamaktır. Tüm kullanıcı-tanımlı sayfalarda olduğu gibi, STEP 7 programınıza dahil edecek şekilde STEP 7 blok oluşturursunuz. Arka arkaya değişkenlerini değiştirmek için ayrıcalıklara sahip bir kullanıcı bu sayfaya eriştiğinde ve giriş alanları içine veri girdiğinde veya bir seçme listesinden bir seçim yaptığında, Web sunucusu, girdiyi, değişken için uygun veri türüne dönüştürür ve CPU'daki değişkene değer yazar. HTML giriş alanları ve HTML seçme listeleri için ad (name) ifadesinin AWP\_In\_Variable komutun ad ifadesi için tipik bir sözdizimini kullandığını unutmayınız. Genellikle adı tek tırnak işaretleri içine alınız ve bir veri bloğuna başvurursanız, veri bloğu adını çift tırnak içine alınız.

Form yönetimiyle ilgili Ayrıntılar için, HTML belgelerine bakınız.

### Sözdizimi

```
<!-- AWP_In_Variable Name='<Varname1>' [Use='<Varname2>'] ... -->
```

### Parametreler

<Varname1>	Use ibaresi sağlanmıyorsa, Varname1 yazılacak olan değişkendir. Bu STEP 7 programından bir PLC etiketi adı ya da belirli bir veri bloğundan bir etiket olabilir. Bir Use ibaresi sağlanırsa, Varname1, <Varname2> (Sayfa 634) de başvuru alan değişken için alternatif bir isimdir. Bu HTML sayfası içinde bir yerel addır.
<Varname2>	Use ibaresi sağlanmıyorsa, Varname2 yazılacak olan değişkendir. Bu STEP 7 programından bir PLC etiketi adı ya da belirli bir veri bloğundan bir etiket olabilir.

Name ifadeleri ve Use ibarelerinin her ikisi için tam ad tek tırnak içine alınmalıdır. Tek tırnak içinde, bir PLC etiketi ve bir veri bloğu ismi etrafında çift tırnak kullanınız. Veri bloğu etiket adı değil veri bloğu ismi çift tırnak içindedir. Veri bloğu etiketleri için veri bloğu numarasını değil blok adını kullandığınızı unutmayınız.

### HTML giriş alanının kullanıldığı Örnekler

```
<!-- AWP_In_Variable Name='\"Target_Level\"' -->
<form method="post">
<p>Input Target Level: <input name='\"Target_Level\"' type="text" />
</p>
</form>

<!-- AWP_In_Variable Name='\"Data_block_1\".Braking' -->
<form method="post">
<p>Braking: <input name='\"Data_block_1\".Braking' type="text" />
%</p>
</form>

<!-- AWP_In_Variable Name='\"Braking\"' Use='\"Data_block_1\".Braking' -
->
<form method="post">
<p>Braking: <input name='\"Braking\"' type="text" /> %</p>
</form>
```

### HTML seçme listesinin kullanıldığı Örnekler

```
<!-- AWP_In_Variable Name='\"Data_block_1\".ManualOverrideEnable'-->
<form method="post">
<select name='\"Data_block_1\".ManualOverrideEnable'>
<option value=:"Data_block_1\".ManualOverrideEnable:> </option>
<option value=1>Yes</option>
<option value=0>No</option>
</select><input type="submit" value="Submit setting" /></form>
```

#### Not

Değişkenleri değiştirmek için ayrıcalıklara sahip olan bir kullanıcı CPU'ya veri yazabilir. Kullanıcı değiştirme ayrıcalıklarına sahip değilse, Web sunucusu komutları yok sayar.

Bir etiket adı veya veri bloğu adı özel karakterler içeriyorsa, ek tırnak işareti ya da "özel karakter (Sayfa 639) içeren etiket isimlerini işleme" konusunda açıklanan çıkış karakterleri kullanmanız gerekir.

### 11.6.2.3 Özel değişkenler okuma

Web sunucusu, HTTP cevap başlığında özel değişkenleri saklamak için PLC den değerleri okuma yeteneği sağlar. Örneğin HEADER: Yer Özel değişkeni, kullanarak URL'yi başka bir yere yönlendirmek için bir PLC etiketinden bir yol adı okumak isteyebilirsiniz.

#### Sözdizimi

```
<!-- AWP_Out_Variable Name='<Type>:<Name>' [Use='<Varname>'] -->
```

## Parametreler

<Type>	Özel değişken tipi aşağıdakilerden biridir: HEADER COOKIE_VALUE COOKIE_EXPIRES
<Name>	HEADER değişkenlerin tüm isimlerinin listesi için HTTP belgelerine bakınız. Birkaç örnek, aşağıda sıralanmıştır: Durum: yanıt kodu Yer: yönlendirme için yol Tekrar dene-sonra: İstekte bulunan istemci için hizmetin Ne kadar uzun bir süre kullanılamaz olması bekleniyor COOKIE_VALUE ve COOKIE_EXPIRES tipleri için, <Name> belirli bir çerez adıdır. COOKIE_VALUE: name: adlandırılmış çerez değeri COOKIE_EXPIRES:name: adlandırılmış çerezin saniye cinsinden zaman aşımı süresi Name ifadesi, tek veya çift tırnak içine alınmalıdır. Hiçbir Use ifadesi belirtilmemişse, özel değişken adı bir PLC etiketi adına karşılık gelir. Çift tırnak içinde PLC etiketi ve tek tırnak içinde tam Name ifadesi bulunmalıdır. Özel değişken adı ve PLC etiketi adı tam olarak eşleşmelidir.
<Varname>	Okunacak değişken için PLC etiketi veya veri bloğu etiketinin adı VarName tek tırnak işareti içine alınmalıdır. Tek tırnak içinde, bir PLC etiketi veya veri bloğu ismi etrafında çift tırnak işaretlerini kullanınız. Veri bloğu etiket adı değil Veri bloğu ismi çift tırnak içindedir. Veri bloğu etiketleri için bir veri bloğu numarası değil bloğun adını kullanabildiğinizi unutmayınız.

Bir etiket adı veya veri bloğu adı özel karakterler içeriyorsa, ek tırnak işareti ya da özel karakterler (Sayfa 639) içeren ve "etiket isimlerinin işlenmesi" konusunda açıklanan çıkış karakterleri kullanmanız gerekir.

### Örnek: Use ibaresi olmadan özel bir değişken okuma

```
<!-- AWP_Out_Variable Name='HEADER:Status' -->
```

Bu örnekte, HTTP özel değişkeni "HEADER:Status", "HEADER:Status" PLC etiketi değerini alır. Hiçbir Use ibaresi belirtilmiyorsa PLC etiketi çizelgesindeki isim, özel değişkenin adıyla tam olarak eşleşmesi gerekir.

### Örnek: Bir Use ibaresi ile özel bir değişken okuma

```
<!-- AWP_Out_Variable Name='HEADER:Status Use='Status' -->
```

Bu örnekte, HTTP özel değişkeni "HEADER:Status", PLC etiketi "Status" değerini alır.

## 11.6.2.4 Özel değişkenler yazma

Web sunucusu HTTP istek başlığında özel değişkenlerden CPU'ya değerler yazma yeteneği sağlar. Örneğin, bir kullanıcı-tanımlı Web sayfası, bir sayfaya erişen kullanıcı ile ilişkili çerez hakkında STEP 7'deki bilgiyi veya başlık bilgilerini saklayabilir. Web sunucusu, değişkenleri değiştirme ayrıcalıklarına sahip bir kullanıcı olarak oturum açıldığında CPU'ya yazabileceğiniz belirli özel değişkenlere erişim sağlar.



## Sözdizimi

```
<!-- AWP_In_Variable Name='<Type>:<Name>' [Use='<Varname>']-->
```

## Parametreler

<Type>	Özel değişken tipi aşağıdakilerden biridir: HEADER SERVER COOKIE_VALUE
<Name>	Bu örneklerde gösterildiği gibi, yukarıda tarif edilen tiplerde olan özel değişken: HEADER:Accept: kabul edilebilir içerik tipleri HEADER>User-Agent: isteğin sahibi kullanıcı hakkında bilgiler. SERVER:current_user_id: geçerli kullanıcı id; hiçbir kullanıcı oturum açmamışsa 0 SERVER:current_user_name: Geçerli kullanıcı adı COOKIE_VALUE:<name>: adlandırılmış çerez değeri Name ibaresini tek tırnak içine alınız. Use ibaresi belirtilmemişse, özel değişken adı bir PLC etiketi adına karşılık gelir. Çift tırnak içinde PLC etiketi ve tek tırnak içinde tam Name ibaresi bulunmalıdır. Özel değişken adı ve PLC etiketi adı tam olarak eşleşmelidir. HEADER değişkenlerin tüm isimlerinin listesi için HTTP belgelerine bakınız.
<Varname>	Bir PLC etiket adı veya bir veri bloğu etiketi olabilen özel değişkene, yazmak istediğiniz STEP 7 programınızdaki değişken adı. VarName tek tırnak işareti içine alınmalıdır. Tek tırnak içinde, bir PLC etiketi veya veri bloğu ismi etrafında çift tırnak işaretlerini kullanınız. Veri bloğu etiket adı değil veri bloğu ismi çift tırnak içindedir. Veri bloğu etiketleri için bir veri bloğu numarası değil bloğun adını kullanabildiğinizi unutmayınız.

## Örnekler

```
<!-- AWP_In_Variable Name='"SERVER:current_user_id"' -->
```

Bu örnekte, Web sayfası HTTP özel değişkenin değerini yazar

"SERVER:current\_user\_id" ni PLC etiket isimli "SERVER:current\_user\_id" ye.

```
<!-- AWP_In_Variable Name=SERVER:current_user_id' Use='"my_userid"' -->
```

Bu örnekte, Web sayfası HTTP özel değişkenin değerini yazar

"SERVER:current\_user\_id" ni PLC etiket isimli "my\_userid" ye.

## Not

Değişkenleri değiştirmek için ayrıcalıklara sahip olan bir kullanıcı CPU' ya veri yazabilir. Kullanıcı değiştirme ayrıcalıklarına sahip değilse, Web sunucusu komutları yok sayar.

Bir etiket adı veya veri bloğu adı özel karakterler içeriyorsa, ek tırnak işareti ya da "özel karakter (Sayfa 639) içeren etiket isimlerini işleme" konusunda açıklanan çıkış karakterleri kullanmanız gerekir.

### 11.6.2.5 Değişken bir referans için bir diğer ad (takma ad) kullanma

Bir In\_Variable veya Out\_Variable için kullanıcı-tanımlı Web sayfasında bir takma ad kullanabilirsiniz. Örneğin, HTML sayfanızda CPU da kullanıldan farklı bir sembolik bir ad kullanılabilir ya da CPU'daki bir değişkeni özel bir değişken ile eşitleyebilirsiniz. AWP Use ifadesi bu yeteneği sağlar.

#### Sözdizimi

```
<-- AWP_In_Variable Name='<Varname1>' Use='<Varname2>' -->
<-- AWP_Out_Variable Name='<Varname1>' Use='<Varname2>' -->
```

#### Parametreler

<Varname1>	Takma ad veya özel değişken adı Varname1 tek veya çift tırnak içine alınmalıdır.
<Varname2>	Bir takma ad atamak istediğiniz PLC değişkenin adı. Değişken bir PLC etiketi, bir veri bloğu etiketi, ya da özel bir değişken olabilir. Varname2 tek tırnak işareti içine alınmalıdır. Tek tırnak içinde, bir PLC etiketi, özel değişken veya veri blok adını çift tırnak içinde kullanınız. Veri bloğu etiket adı değil veri bloğu ismi çift tırnak içindedir. Veri bloğu etiketleri için, bir veri bloğu numarası değil bloğun adını kullanabildiğinizi unutmayınız.

#### Örnekler

```
<-- AWP_In_Variable Name='SERVER:current_user_id'
Use=' "Data_Block_10".server_user' -->
```

Bu örnekte, özel değişken SERVER: current\_user\_id, "Data\_Block\_10" veri bloğundaki "server\_user" etiketine yazılır.

```
<-- AWP_Out_Variable Name='Weight'
Use=' "Data_Block_10".Tank_data.Weight' -->
```

Bu örnekte, veri bloğu yapı elemanı Data\_Block\_10.Tank\_data.Weight içindeki değer kullanıcı-tanımlı Web sayfasının geri kalanı boyunca "Ağırlık" la basitçe başvurulabilir.

```
<-- AWP_Out_Variable Name='Weight' Use=' "Raw_Milk_Tank_Weight"' -->
```

Bu örnekte, PLC etiketi "Raw\_Milk\_Tank\_Weight" içindeki değer, kullanıcı-tanımlı Web sayfasının geri kalanı boyunca "Ağırlık" la basitçe başvurulabilir.

Bir etiket adı veya veri bloğu adı özel karakterler içeriyorsa, ek tırnak işareti ya da özel karakterler (Sayfa 639) içeren ve "etiket isimlerinin işlenmesi" konusunda açıklanan çıkış karakterleri kullanmanız gerekir.

### 11.6.2.6 Enum tipleri tanımlama

Kullanıcı-tanımlı sayfalarda enum tipleri tanımlayabilir ve bir AWP komutunda elemanları atayabilirsiniz.

#### Sözdizimi

```
<!-- AWP_Enum_Def Name='<Enum type name>' Values='<Value>,
<Value>,... ' -->
```

## Parametreler

<Enum tip ismi>	Tek veya çift tırnak işareti içine alınmış numaralandırılmış tip adı.
<Değer>	<sabit>:<isim> Sabit, enum tipi ataması için sayısal değeri gösterir. Toplam sayı sınırlanmamıştır. İsim enum elemanına atanan değerdir.

Enum değeri atamalarının tüm string'inin tek tırnak içine alındığını ve her münferit enum tipi elemanı atamasının, çift tırnak içine alındığını unutmayınız. Bir enum tipi tanımının kapsamı kullanıcı-tanımlı Web sayfaları için geneldir. Eğer dil klasörlerinde (Sayfa 656) kullanıcı-tanımlı Web sayfalarını kurduysanız, enum tipi tanımı dil klasöründeki tüm sayfalar için geneldir.

## Örnek

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is full", 2:"Tank is empty"' -->
```

### 11.6.2.7 Bir enum tipli CPU değişkenlerine başvurma

Bir enum tipi için CPU içinde bir değişken atayabilirsiniz. Bu değişken bir okuma (Sayfa 629) veya yazma işleminde (Sayfa 630), kullanıcı-tanımlı Web sayfasında başka bir yerde kullanılabilir. Bir okuma işlemi üzerinde, Web sunucusu karşılık gelen enum metin değeri ile CPU' dan okunan sayısal değerini yerini alacaktır. Bir yazma işlemi üzerinde, Web sunucusu CPU' ya değer yazmadan önce metin değeri yerine metne karşılık gelen numaralandırma tamsayı değerini kullanacaktır.

## Sözdizimi

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<EnumType>" -->
<!-- AWP_Out_Variable Name='<Varname>' Enum="<EnumType>" -->
```

## Parametreler

<Varname>	Enum tipi ile ilişkilendirilen PLC etiketi veya veri bloğu etiketi adı ya da beyan edilirse bir PLC etiketi için takma adın Adı (Sayfa 634). VarName tek tırnak işareti içine alınmalıdır. Tek tırnak içinde, bir PLC etiketi veya veri bloğu ismi etrafında çift tırnak işaretlerini kullanınız. Veri bloğu etiket adı değil Veri bloğu ismi çift tırnak içindedir. Veri bloğu etiketleri için, bir veri bloğu numarası değil bloğun adını kullanabildiğinizi unutmayınız.
<EnumType>	Tek veya çift tırnak içine alınması gereken numaralandırılmış tip adı

Bir enum tipi referans kapsamı, geçerli fragmandır.

## Değişken bir okumada Örnek kullanım

```
<!-- AWP_Out_Variable Name='Alarm' Enum="AlarmEnum" -->...
<p>The current value of "Alarm" is := "Alarm":</p>
```

CPU'da "Alarm" değeri 2 ise, enum tipi tanımı (Sayfa 635) "Tank boş" metin string için 2 sayısal değeri atadığı için HTML sayfası "Alarm"ın geçerli değeri Tank boş' ifadesini görüntüler.

### Değişken bir yazmada Örnek kullanım

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is full", 2:"Tank is empty"' -->
<!-- AWP_In_Variable Name='Alarm' Enum='AlarmEnum' -->...
<form method="POST">
<p><input type="hidden" name="Alarm" value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /><p>
</form>
```

Enum tipi tanımı (Sayfa 635) "Tank dolu" yu sayısal değer 1 olarak atadığı için CPU da "Alarm" adlı PLC etikete 1 değeri yazılır.

AWP\_In\_Variable bildiriminde Name ifadesinin, AWP\_Enum\_Ref bildiriminde name ifadesine tam olarak karşılık geldiğini unutmayınız.

### Bir takma ad kullanımı ile değişken yazmada Örnek kullanım

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is full", 2:"Tank is empty"' -->
<!-- AWP_In_Variable Name='Alarm' Enum='AlarmEnum'
Use='Data_block_4.Motor1.Alarm'-->...
<form method="POST">
<p><input type="hidden" name="Alarm" value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /><p>
</form>
```

Enum tipi tanımı (Sayfa 635) "Tank dolu" yu sayısal değer 1 e atadığı için 1 değeri, CPU'da "Data\_Block\_4" veri bloğunda "Motor1.Alarm" adlı PLC etikete karşılık gelen "Alarm" takma adına yazılır.

Bir etiket adı veya veri bloğu adı özel karakterler içeriyorsa, ek tırnak işareti ya da özel karakterler (Sayfa 639) içeren ve "etiket isimlerinin işlenmesi" konusunda açıklanan çıkış karakterleri kullanmanız gerekir.

---

### Not

Önceki sürümler, tanımlanmış bir enum tipi ile bir değişkeni ilişkilendirmek için ayrı bir AWP\_Enum\_Ref bildirimine ihtiyaç duyar. STEP 7 ve S7-1200, AWP\_Enum\_Ref bildirimleri ile mevcut kodu destekler; Ancak, bu komut artık gerekli değildir.

---

### 11.6.2.8 Fragmanları oluşturma

Web sunucusu için CPU Özellikleri "Blok Oluştur" tıkladığınızda, STEP 7 bir kontrol DB ve fragman DB'ler olarak kullanıcı-tanımlı Web sayfalarını dönüştürür ve depolar. Belirli sayfaları için veya belirli sayfaların bölümleri için özel fragmanlar kurabilirsiniz. "Start\_Fragment" AWP komutu ile bir Ad ve numaraya göre bu fragmanları tanımlayabilirsiniz. Başka AWP\_Start\_Command verilene kadar veya dosyanın sonuna ulaşıncaya kadar AWP\_Start\_Fragment komutu ardından sayfadaki her şey bu fragmana aittir.

#### Sözdizimi

```
<!-- AWP_Start_Fragment Name='<Name>'
[Type=<Type>] [ID=<id>] [Mode=<Mode>] -->
```

#### Parametreler

<İsim>	Metin string: fragman DB adı Fragman adları bir harfle ya da alt çizgi ile başlamalı ve harfler, rakamlar ve alt çizgilerden oluşmalıdır. Fragman ismi formun düzenli bir ifadesidir: [a-zA-Z_][a-zA-Z_0-9]*
<Tip>	"manuel" veya "otomatik" Manuel: STEP 7 programının, bu fragmanı istemesi gerekir ve buna göre yanıt verebilir. Fragman işlemi, STEP 7 ve kontrol DB değişkenleriyle kontrol edilmelidir. Otomatik: Web sunucusu otomatik olarak fragmanı işler. Eğer tip parametresi belirtmezseniz, varsayılan "otomatik" olur.
<id>	Integer kimlik numarası. ID parametresi belirtmezseniz, Web sunucusu varsayılan olarak bir numara atar. Manuel fragmanlar için ID olarak düşük bir sayı ayarlar. ID STEP 7 programının manuel fragmanı kontrol ettiği araçtır.
<Mod>	"görünür" veya "gizli" Görünür: fragmanın içeriği kullanıcı-tanımlı Web sayfasında görüntülenir. Gizli: fragmanın içeriği kullanıcı-tanımlı Web sayfasında görüntülenmez. Tip parametresi belirtmezseniz, varsayılan "görünür" olur.

#### Manuel fragmanları

Kullanıcı tanımlı bir Web sayfası veya bir sayfa bölümü için bir manuel fragman oluşturursanız, STEP 7 programı fragmanın gönderildiği zamanı kontrol etmelidir. STEP 7 programı manuel kontrol altında kullanıcı tanımlı bir sayfa için kontrol DB sindeki uygun parametreleri ayarlamalı ve daha sonra değiştirilmiş olan kontrol DB ile WWW komutunu çağırmanız gerekir. Kontrol DB yapısını anlamak ve bireysel sayfaları ve fragmanların nasıl işlendiğini görmek için Gelişmiş kullanıcı-tanımlı Web sayfası kontrolü (Sayfa 660) konu bölümüne bakınız.

### 11.6.2.9 Fragmanları içeri aktarma (import)

HTML kodunuzun bir bölümünde adlandırılmış bir fragman oluşturabilir ve sonra bu fragmanı kullanıcı tanımlı Web sayfaları grubunun başka bir yerinde import edebilirsiniz. Örneğin, başlangıç sayfasına sahip kullanıcı-tanımlı Web sayfaları grubu ve daha sonra bir başlangıç sayfasındaki bağlantılardan erişilebilir olan birkaç diğer HTML sayfalarını düşününüz. Ayrı sayfaların her birinin sayfadaki şirket logosunu görüntülediğini varsayınız. Siz şirket logosunun resmini yükleyen fragmanı (Sayfa 637) oluşturarak bunu uygulayabilirsiniz. Her münferit HTML sayfası, sonra şirket logosunu görüntülemek için bu fragmanı içe aktarabilir. Bu amaçla AWP Import\_Fragment komutunu kullanınız. Fragman için HTML kodu sadece bir fragmanda vardır, ancak seçebildiğiniz kadar çok web sayfalarında gerekli olduğu kadar çok bu fragmanı DB'yi import edebilirsiniz.

#### Sözdizimi

```
<!-- AWP_Import_Fragment Name='<Name>' -->
```

#### Parametreler

<İsim>	Metin dizesi: import edilecek fragman DB ismi
--------	---

#### Örnek

Bir resmi görüntülemek için bir fragman oluşturan HTML kodundan alıntı:

```
<!-- AWP_Start_Fragment Name='My_company_logo' --><p></p>
```

Logosu resmini görüntüleyen fragmanı import eden başka .html dosyasındaki HTML kodundan alıntı:

```
<!-- AWP_Import_Fragment Name='My_company_logo' -->
```

Her iki .html dosyası (biri fragmanı oluşturur diğeri onu import eder) STEP 7 DE kullanıcı-tanımlı sayfaları yapılandırdığınız zamanı tanımladığınız klasör yapısındadır (Sayfa 641).

### 11.6.2.10 Tanımları birleştirme

Kullanıcı-tanımlı Web sayfalarında kullanılmak üzere değişkenleri beyan ettiğiniz zaman, değişken için bir değişken bildirim ile bir diğer adı (Sayfa 634) birleştirebilirsiniz. Ayrıca, bir ifadede çoklu Out\_Variables ve bir ifadede çoklu In\_Variables beyan edebilirsiniz.

#### Örnekler

```
<!-- AWP_In_Variable Name='Level', Name='Weight', Name='Temp'  
-->  
<--! AWP_Out_Variable Name='HEADER:Status', Use='Status',  
Name='HEADER:Location', Use='Location',  
Name='COOKIE_VALUE:name', Use='my_cookie' -->  
<!-- AWP_In_Variable Name='Alarm' Use='Data_block_10'.Alarm' -->
```

### 11.6.2.11 Özel karakterler içeren etiket adlarını işleme

Kullanıcı-tanımlı Web sayfalarında değişken adlarını belirtirken etiket adları özel anlamlara sahip karakterler içeriyorsa, özel olarak dikkat göstermeniz gerekir.

#### Değişkenleri Okuma

Bir değişkeni (Sayfa 629) okumak için aşağıdaki sözdizimini kullanınız:  
 :=<Varname>:

Aşağıdaki kurallar okuma değişkenleri için geçerlidir:

- PLC etiketi çizelgesinden değişken isimleri için, etiket adını çift tırnak içine alınız.
- Veri bloğu etiketleri olan değişken isimleri için, veri bloğu adını çift tırnak için alınız. Etiket tırnak işaretleri dışına alınız.
- Direk I / O adresleri, bellek adresleri, ya da diğer isimler olan değişken adları için, okuma değişkenini tırnak işareti içine almayınız.
- Bir ters eğik çizgi içeren etiket isimleri veya veri bloğu etiket isimleri için, başka bir ters eğik çizgi ile birlikte ters eğik çizgi koyunuz.
- Bir etiket adı veya veri bloğu etiket adı bir iki nokta üst üste işareti, küçük işareti, büyük işareti veya & işareti içeriyorsa okuma değişkeni için hiçbir özel karakteri olmayan bir takma ad tanımlayınız ve takma ad kullanan değişkeni okuyunuz. Bir ters eğik çizgi ile Use ifadesi içinde etiket adlarında öncesinde iki nokta üst üste koyunuz.

Çizelge 11- 1 Okuma değişkenlerine ait örnekler

Veri bloğu ismi	Etiket ismi	Okuma komutu
n/a	ABC:DEF	<!--AWP_Out_Variable Name='special_tag' Use = "ABC:DEF" --> :=special_tag:
n/a	T\	:= "T\\":
n/a	A \B 'C :D	<!--AWP_Out_Variable Name='another_special_tag' Use=' "A \\B \'C :D" --> :=another_special_tag:
n/a	a<b	<!--AWP_Out_Variable Name='a_less_than_b' Use=' "a<b" --> :=a less than b:
Data_block_1	Tag_1	:= "Data_block_1".Tag_1:
Data_block_1	ABC:DEF	<!-- AWP_Out_Variable Name='special_tag' Use=' "Data_block_1".ABC\ :DEF' --> :=special_tag:
DB A' B C D\$ E	Tag	:= "DB A' B C D\$ E".Tag:
DB:DB	Tag:Tag	<!--AWP_Out_Variable Name='my_tag' Use = "DB:DB".Tag\ :Tag' --> :=my_tag:

## Name ve Use ifadeleri

AWP komutları (AWP\_In\_Variable, AWP\_Out\_Variable, AWP\_Enum\_Def, AWP\_Enum\_Ref, AWP\_Start\_Fragment ve AWP\_Import\_Fragment) Name ifadeleri içerir. <input> ve <select> gibi HTML form komutları da isim ifadelerine sahiptir var. AWP\_In\_Variable ve AWP\_Out\_Variable ayrıca Use ifadelerine sahip olabilir. Komuttan bağımsız olarak, özel karakterler işlenmesiyle ilgili Name ve Use ifadeleri için sözdizimi ayndır:

- Bir Name veya Use ifadesi için sağladığınız metin tek tırnak içine alınmalıdır. Kapatılan isim PLC etiketi veya veri bloğu ismi ise, tam ifade için tek tırnak işaretleri kullanınız.
- Bir Name veya Use ifadesinde, veri blok adları ve PLC etiket adları çift tırnak içine alınmalıdır.
- Bir etiket adı veya veri bloğu adı tek tırnak karakteri ya da ters eğik çizgi içeriyorsa, bir ters eğik çizgi ile bu karakterden çık. Ters eğik çizgi AWP komut derleyicisinde çıkış karakteridir.

Çizelge 11- 2 Name ifadeleriyle ilgili örnekler

Veri bloğu ismi	Etiket ismi	Name ifade seçenekleri
n/a	ABC'DEF	Name=' "ABC\ 'DEF" '
n/a	A\B'C:D	Name=' "A \\B \'C :D" '
Data_block_1	Tag_1	Name=' "Data_block_1".Tag_1'
Data_block_1	ABC'DEF	Name=' "Data_block_1".ABC\ 'DEF'
Data_block_1	A\B'C:D	Name=' "Data_block_1".A \\B \'C :D'
DB A' B C D\$ E	Tag	Name=' "DB A\ ' B C D\$ E".Tag'

Use ifadeleri Name ifadeleri ile aynı kuralları takip eder.

### Not

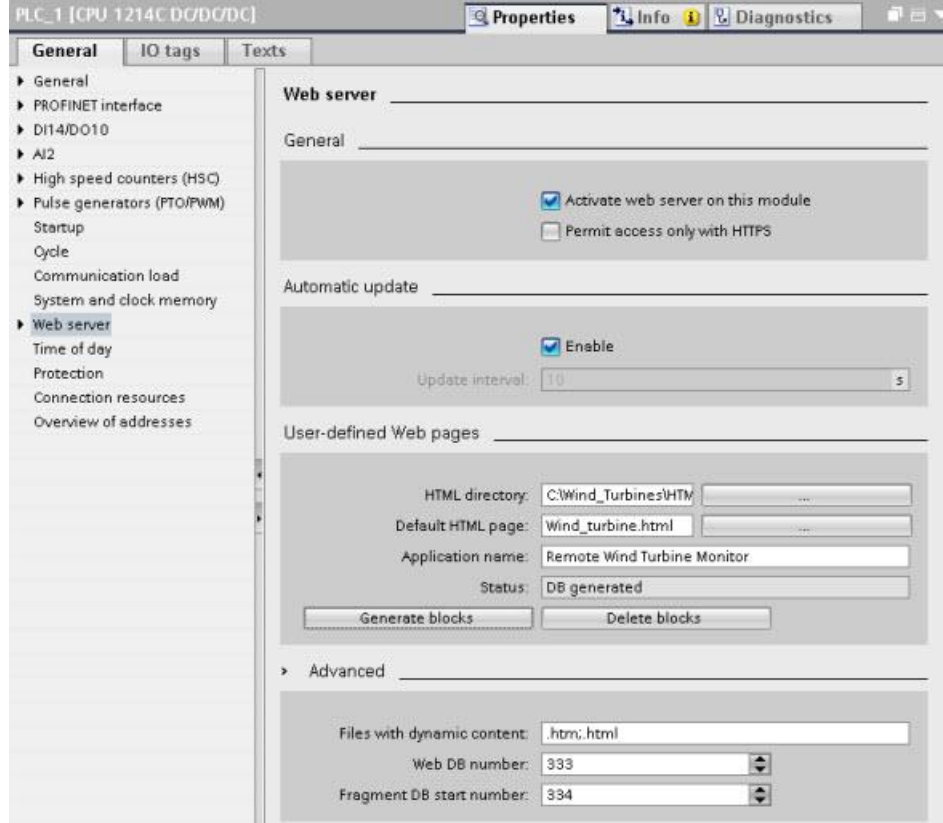
HTML sayfasında kullandığınız karakterler ne olursa olsun, HTML sayfasının karakter setini UTF-8 olarak ayarlayınız ve UTF-8 karakter kodlamalı editör ile kaydediniz.



### 11.6.3 Kullanıcı-tanımlı Web sayfalarının kullanımını yapılandırma

STEP 7 den kullanıcı-tanımlı Web sayfalarını yapılandırmak için şu adımları izleyiniz:

1. Cihaz Yapılandırma görünümünde CPU yu seçiniz.
2. CPU için denetim penceresinde "Web sunucusu" özelliklerini görüntüler.



3. Zaten seçili değilse, " Web sunucusunu bu modül üzerinde etkinleştiriniz " onay kutusunu seçiniz.
4. Web sunucusunun şifreli haberleşme kullandığından emin olmak ve Web'ten erişilebilir CPU nun güvenliğini artırmak için "sadece HTTPS ile erişime izin ver " onay kutusunu seçiniz.
5. PC'nizde varsayılan HTML sayfasını (başlangıç sayfası) kaydettiğiniz klasör adına göz atın yada yazarak giriniz.
6. Varsayılan sayfanın adını giriniz.
7. Uygulamanız için bir ad verin (Opsiyonel). Web sunucusu, ayrıca web sayfalarını alt kategorilere ayırmak veya gruplamak için uygulama adını kullanır. Bir uygulama adı var olduğunda URL şu biçimde görünür: http://ww.xx.yy.zz/awp/ <uygulama adı> / <sayfa adı> .html.

Uygulama adında özel karakterlerden kaçınınız. Bazı karakterler, Web sunucusunun kullanıcı tanımlı sayfaları görüntüleyememesine neden olabilir.

8. AWP komutlarının varlığı için incelenecek dosya uzantıları belirtiniz. Varsayılan olarak .htm, .html ya da .Js uzantılı dosyaları STEP 7 analiz eder. Ek dosya uzantıları varsa, bunları ekler.

### 11.6 Kullanıcı tanımlı Web sayfaları

9. Web DB numarası için varsayılan tutun, ya da seçtiğiniz bir sayı giriniz. Bu Web sayfalarının ekranını kontrol eden kontrol DB sinin DB sayısıdır.
10. Fragman DB başlangıç numarası için varsayılan tutun, ya da seçtiğiniz bir sayı giriniz. Bu Web sayfalarını içeren fragman DB'lerinin ilkidir.

#### Program blokları oluşturma

"Bloklar Oluştur" butonuna tıkladığınızda, STEP 7 belirttiğiniz HTML kaynak dizinindeki veri bloklarını ve Web sayfalarının çalışması için bir kontrol veri bloğu oluşturur. Uygulamanız (Sayfa 642) için gerekli olan bu öznitelikleri ayarlayabilirsiniz. STEP 7, ayrıca HTML sayfalarının tümünün temsilini tutmak için fragman veri blokları grubu oluşturur. Veri bloklarını oluşturduğunuzda, STEP 7 kontrol verileri blok numarasını ve fragman veri bloklarının ilkinin sayısını görüntülemek için özellikleri günceller. Veri blokları oluşturduktan sonra, kullanıcı-tanımlı Web sayfaları STEP 7 programının bir parçasıdır. Bu sayfalara karşılık gelen bloklar, proje gezinti ağacında Program bloklarının altında sistem blokları klasöründe olan Web sunucusu klasöründe görünür.

#### Program bloklarını silme

Daha önce oluşturmuş veri bloklarını silmek için "veri bloklarını sil" düğmesine tıklayınız. STEP 7, kullanıcı-tanımlı Web sayfalarına karşılık gelen projenizden kontrol veri bloğunu ve fragman veri bloklarının tümünü siler.


### 11.6.4 Kullanıcı-tanımlı web sayfaları için WWW komutu programlama

STEP 7 kullanıcı programınız, standart Web sayfalarından erişilebilir olacak şekilde kullanıcı-tanımlı Web sayfaları için WWW komutunu içermeli ve sırayla yürütmelidir. Kontrol veri bloğu WWW komutunun giriş parametresidir ve fragman veri bloklarında temsil edilen sayfaların içeriğinin yanı sıra durum kontrol bilgisini belirtir. Kullanıcı-tanımlı Web sayfalarının konfigürasyonunda (Sayfa 664) "blok oluştur" butonuna tıkladığınızda STEP 7 kontrol veri bloğu oluşturur.

#### WWW komutu programlama

Standart Web sayfalarından erişilebilir olacak şekilde kullanıcı-tanımlı Web sayfaları için STEP 7 programı WWW komutunu yürütmelidir. Uygulama gereksinimleriniz ve tercihlerinizle dikte edilen sadece belirli koşullar altında mevcut olan kullanıcı-tanımlı Web sayfalarını isteyebilirsiniz. Bu durumda program mantığınız, WWW komutunu ne zaman çağıracağınızı kontrol edebilir.

Çizelge 11- 3 WWW komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a function block named 'WWW'. It has an 'EN' (Enable) input on the left and an 'ENO' (Enable Out) output on the right. Below the block, there are two data inputs: 'CTRL_DB' and 'RET_VAL'. The 'RET_VAL' output is connected to a variable 'RET_VAL' in the SCL code.</p>	<pre>ret_val := WWW(     ctrl_db:=_uint_in_);</pre>	<p>Standart Web sayfalarından kullanıcı-tanımlı Web sayfalarına erişim sağlar</p>

Kontrol DB'sinin integer DB numarasına karşılık gelen kontrol veri blok giriş parametresi (CTRL\_DB) sağlamanız gerekir. Kullanıcı-tanımlı Web sayfaları için blokları oluşturduktan sonra CPU nun Web Sunucusu özelliklerinde (Web DB sayısı denir) bu kontrol DB blok numarasını bulabilirsiniz. WWW komutunun CTRL\_DB parametresi olarak integer DB numarası giriniz. Dönüş değeri (RET\_VAL) fonksiyon sonucunu içerir. WWW komutunun asenkron olarak yürütüldüğünü ve daha sonra bir hata oluşabilmesine rağmen RET\_VAL çıkışının 0 başlangıç değerine sahip olabileceğini unutmayınız. Program, uygulamanın başarıyla başladığını sağlamak için kontrol DB durumunu veya WWW için bir sonraki çağrı ile RET\_VAL'i kontrol edebilirsiniz.

Çizelge 11- 4 Dönüş değeri

RET_VAL	Açıklama
0	Hata yok
16#00yx	x: ilgili bit ile temsil edilen istek bekleme durumundadır: x=1: istek 0 x=2: istek 1 x=4: istek 2 x=8: istek 3 x değerleri, çoklu isteklerin bekleyen durumlarını temsil etmesi için mantıksal olarak OR-ed olabilir. Örneğin x = 6 ise, 1 ve 2 istekleri bekler. y: 0: Hata yok; 1: hata var ve "last_error" kontrol DB'sinde (Sayfa 660) ayarlanmıştır.
16#803a	Kontrol DB yüklü değil.
16#8081	Denetimi DB yanlış türü, biçimi veya sürüm olduğunu.
16#80C1	Hiçbir kaynak web uygulaması başlatmak için kullanılabilir.

## Kontrol DB Kullanımı

STEP 7, "blok oluştur" tıkladığınızda kontrol veri bloğu oluşturur ve kullanıcı-tanımlı Web sayfaları özelliklerinde kontrol DB sayısını görüntüler. Proje gezinti ağacında Program blokları klasöründe kontrol DB bulabilirsiniz.

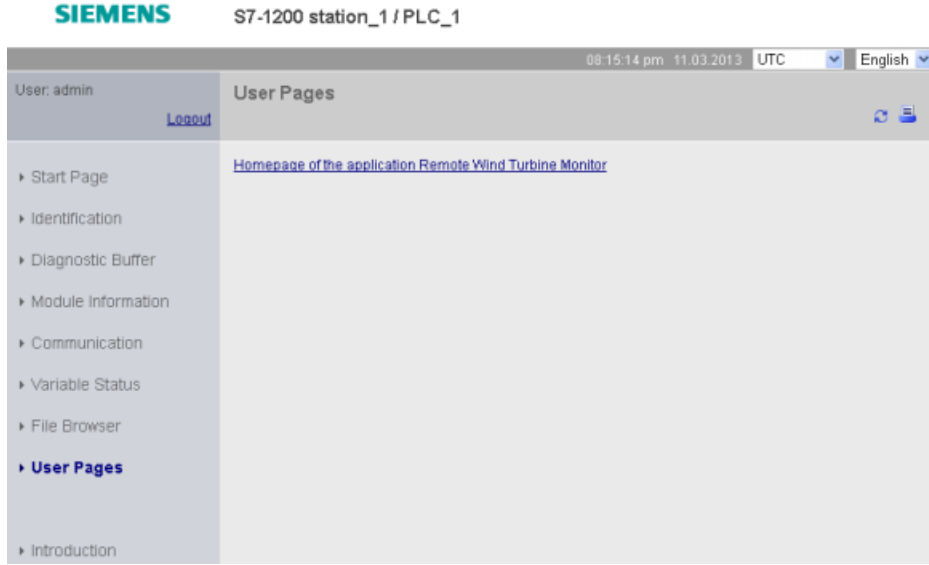
Tipik olarak STEP 7 programı, hiçbir ek manipülasyon olmaksızın "bloklar Oluştur" işlemi tarafından oluşturulan kontrol DB'yi doğrudan kullanır. Ancak, STEP 7 kullanıcı programı web sunucusunu devre dışı bırakmak için ya da sonradan yeniden etkinleştirmek için kontrol DB sindeki global komutları ayarlayabilirsiniz. Ayrıca, manuel fragman DB'leri (Sayfa 641) olarak oluşturduğunuz kullanıcı-tanımlı sayfalar için STEP 7 kullanıcı programı kontrol DB sinde istek çizelgesi aracılığıyla bu sayfaların davranışını kontrol etmelidir. Bu gelişmiş görevler hakkında bilgi için Gelişmiş kullanıcı-tanımlı Web sayfası kontrolü (Sayfa 660) konu bölümüne bakınız.

### 11.6.5 CPU' ya program bloklarını yükleme

Kullanıcı-tanımlı Web sayfaları için bloklar oluşturduktan sonra, onlar sadece başka program blokları gibi STEP 7 programının bir parçası olur. CPU' ya program blokları yüklemek için normal süreci izleyiniz. CPU STOP modunda olduğunda sadece kullanıcı tanımlı Web sayfası program bloklarını indirebileceğinizi unutmayınız.

### 11.6.6 Kullanıcı-tanımlı Web sayfalarına erişim

Standart Web sayfalarından (Sayfa 606) kullanıcı-tanımlı Web sayfalarına erişirsiniz. Standart Web sayfaları, diğer sayfalara olan bağlantıların görüldüğü sol taraftaki menüden "Kullanıcı Sayfaları" için bir bağlantı görüntüler. Mobil cihaz gezinme sayfası da "Kullanıcı Sayfaları" na bir bağlantı sağlar. "Kullanıcı Sayfaları" linkine tıkladığınızda, Web tarayıcınız varsayılan sayfa için bir bağlantı sağlayan sayfaya gider. Kullanıcı-tanımlı sayfalar içinden gezinti size özel tasarlanmış sayfalara göredir.



### 11.6.7 Kullanıcı-tanımlı Web sayfalarına özgü kısıtlamalar

Standart Web sayfaları (Sayfa 664) için kısıtlamalar kullanıcı-tanımlı Web sayfaları içinde geçerlidir. Buna ek olarak, kullanıcı-tanımlı Web sayfaları belirli bazı hususlara sahiptir.

#### Yükleme bellek alanı

"bloklar oluştur"a tıkladığınızda kullanıcı-tanımlı Web sayfaları veri blokları haline gelir. Bu da yükleme bellek alanı gerektirir. Eğer bir bellek kartı takılı ise, kullanıcı-tanımlı Web sayfaları için harici yükleme bellek alanı olarak bellek kartının kapasitesine kadar sahipliğiniz vardır.

Eğer bir bellek kartı takılı değilse, bu bloklar CPU modeline göre sınırlı olan dahili yükleme bellek alanı kaplar.

Kullanılan Yükleme bellek alanı miktarını ve STEP 7 de Çevrimiçi ve Tanılama araçlarından erişilebilir olan miktarı kontrol edebilirsiniz. Ayrıca bu STEP 7'nin kullanıcı-tanımlı Web sayfalarından ürettiği bireysel bloklar için özelliklere ve yükleme bellek tüketimine bakabilirsiniz.

### Not

Kullanıcı-tanımlı Web sayfaları için gerekli alanı azaltmanız gerekirse görüntülerin kullanımını azaltırsınız.

## 11.6.8 Kullanıcı tanımlı bir web sayfası örneği

### 11.6.8.1 Bir rüzgar türbininin izlenmesi ve kontrol edilmesi için Web sayfası

Bir kullanıcı-tanımlı Web sayfası örneği olarak, bir rüzgar türbinini uzaktan izlemek ve kontrol etmek için kullanılan bir Web sayfası düşününüz:



### Açıklama

Bu uygulamada, bir rüzgar türbini çiftliğindeki her bir rüzgar türbine, türbinin kontrolü için bir S7-1200 ile donatılmıştır. STEP 7 program kapsamında, her bir rüzgar türbininin bu rüzgar türbinine özgü veriye sahip olan bir veri bloğu vardır.

Kullanıcı tanımlı Web sayfası, bir PC'den uzak türbin erişimi sağlar. Bir kullanıcı belirli bir rüzgar türbininin CPU sunun standart web sayfalarına bağlanabilir ve bu türbine ait verileri görmek için kullanıcı-tanımlı "Uzak Rüzgar Türbini İzleme" Web sayfasına erişebilirsiniz. Değişkenleri değiştirme ayrıcalıkları olan bir kullanıcı da türbini manuel moda koyabilir ve Web sayfasından türbin hızı, sapma ve eğim için değişkenleri kontrol edebilir. Değişkenlerini değiştirme ayrıcalıkları olan bir kullanıcı, türbinin manuel veya otomatik kontrol altında olup olmadığından bağımsız olarak bir frenleme değerini ayarlayabilirsiniz.

STEP 7 Programı, otomatik kontrolü geçersiz kılmak için Boole değerleri kontrol eder ve ayarlanırsa, türbin hızı, sapma ve eğim için kullanıcı tarafından girilen değerleri kullanır. Aksi takdirde, program bu değerleri göz ardı eder.

## Kullanılan Dosyalar

Bu kullanıcı-tanımlı Web sayfası örneği üç dosyadan oluşur:

- **Wind\_turbine.html:** Bu, kontrolör veriye erişmek için AWP komutlarını kullanarak, yukarıda gösterilen ekranı uygulayan bir HTML sayfasıdır.
- **Wind\_turbine.css:** Bu, HTML sayfası için biçimlendirme stilleri içeren kaskad stil sayfasıdır. Bir kaskad stil sayfası kullanımı isteğe bağlıdır, ancak HTML sayfası geliştirmeyi kolaylaştırabilirsiniz.
- **Wind\_turbine.jpg:** Bu, HTML sayfasının kullandığı arka plan görüntüsü. Kullanıcı tanımlı Web sayfalarında görüntülerin kullanımı, tabii ki, isteğe bağlıdır ve CPU ek alan gerektirir.

Bu dosyalar kurulumunuz ile birlikte sağlanmaz, ancak bir örnek olarak tarif edilmektedir.

## Uygulama

HTML sayfası, kullanıcı girişinden gelen veriler için PLC (Sayfa 629) ye değerler yazmak ve ekran alanları için PLC (Sayfa 630) den değerleri okumak için AWP komutlarını kullanır. Bu sayfa aynı zamanda ON / OFF ayarlarını işlemek için enum tip tanımı (Sayfa 635) ve referans (Sayfa 635) amacıyla AWP komutlarını kullanır.

Sayfanın ilk bölümü rüzgar türbini numarasını içeren bir başlık satırı görüntüler.

**Remote Wind Turbine Monitor: Turbine #5**

Sayfanın bir sonraki bölümü rüzgâr türbininin atmosferik koşullarını görüntüler. Türbin sahasındaki I/O, rüzgar hızı, rüzgar yönü ve mevcut sıcaklığı verir.

Wind speed:	7.5 km/h
Wind direction:	23.5 deg.
Temperature:	17.2 deg. C

Sonra, sayfa, S7-1200 den okunan türbinin çıkış gücünü gösterir.

Power output:	1000 KW
---------------	---------

Aşağıdaki bölümler S7-1200 ile normal bir otomatik kontrolü geçersiz kılan, türbinin manuel kontrolüne izin verir. Bu tipler aşağıda verilmiştir:

- Manuel kumanda: Türbinin manuel kumanda edilmesini sağlar. Türbin hızı, sapma veya eğim için manuel ayarların herhangi birinin kullanımını etkinleştirmeden önce STEP 7 kullanıcı programı, manuel kumanda ayarının doğru olmasını gerektirir.

Manual override: On	Set: Yes
Turbine speed:	15 RPM

- Sapma kumandası: Sapma ayarının manuel kumandasını ve yaw (yön) için bir manuel ayarı etkinleştirir. Sapma ayarını uygulamak için STEP 7 kullanıcı programı, manuel kumanda ve sapma kumandasının her ikisinin de doğru olmasını gerektirir.

Yaw override: On	Set: Yes
Turbine yaw:	52 deg.

- Eğim kumandası: Kanatların eğiminin manuel kumanda edilmesini sağlar. Kanat eğim ayarını uygulamak için STEP 7 kullanıcı programı, manuel kumanda ve eğim kumandasının her ikisinin de (true) doğru olmasını gerektirir.

Pitch override: On	Set: Yes
Blade pitch:	4.5 deg.

HTML sayfası kontrolöre kumanda ayarlarını göndermek için Gönder düğmesini içerir.

Submit override settings and values
-------------------------------------

Frenleme kullanıcı girişi alanı, bir frenleme yüzdesi için bir manuel ayar sağlar. STEP 7 kullanıcı programı bir frenleme değerini kabul etmek için manuel kumanda gerektirmez.

Braking:	2.5 %
----------	-------

Buna ek olarak, HTML sayfası, PLC etiketi çizelgesindeki bir etikete, sayfaya erişen kullanıcının kullanıcı ID sini içeren özel değişken (Sayfa 633) yazmak için bir AWP komutu kullanır.

### 11.6.8.2 Kontrolör verisi okuma ve görüntüleme

"Uzak Rüzgar Türbini İzleme" HTML sayfası, kontrolörden (Sayfa 629) gelen verileri okumak ve sayfa üzerinde görüntülemek için çok sayıda AWP komutları kullanır. Örneğin, örnek Web sayfasının bu kısmında gösterildiği gibi, güç çıkışını gösteren HTML kodunu göz önünde bulundurunuz:

Power output:	1000 KW
---------------	---------

### Örnek HTML kodu

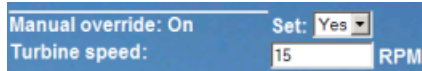
"Uzak Rüzgar Türbini İzleme" HTML sayfasından alınan aşağıdaki alıntı, bir Çizelge satırının sol hücrelerinde "Güç Çıkışı" metnini görüntüler ve güç çıkışı için değişkeni okur ve bunu Çizelge satırının sağ hücrelerinde kilovat metni için kW kısaltmasıyla birlikte görüntüler.

AWP komutu: = "Data\_block\_1" PowerOutput: Okuma işlemini gerçekleştirir. Veri bloklarının veri bloğu numarasıyla değil (yani, "Data\_block\_1" ve "DB1" değil), isimle başvurulduğunu unutmayınız.

```
<tr style="height:2%;">
<td>
<p>Power Output:</p>
</td>
<td>
<p style="margin-bottom:5px;"> := "Data_block_1".PowerOutput: kW</p>
</td>
</tr>
```

### 11.6.8.3 Bir enum tipini kullanma

"Uzak Rüzgar Türbini İzleme" HTML sayfası, HTML sayfasının bir Boolean değeri için "ON" ya da "OFF" görüntülediği yerde üç kopya için ve kullanıcının bir Boolean değeri ayarladığı yer için enum tiplerini kullanır. "ON" için enum tipi 1 değeri ile sonuçlanır ve "OFF" için enum tipi 0 değeriyle sonuçlanır. Örneğin, bir enum tipi kullanarak "Data\_block\_1".ManualOverrideEnable değerinde Manuel Override Etkinleştirme ayarını okumak ve yazmak için HTML kodunu göz önüne alınız:



### Örnek HTML kodu

"Uzak Rüzgar Türbini İzleme" HTML sayfasından alınan alıntılar "Off" için 0 ve "On" için 1 değerleri ile "OverrideStatus" olarak adlandırılan bir enum tipinin nasıl beyan edildiğini gösterir ve daha sonra "Data\_block\_1" adlı veri bloğunda ManualOverrideEnable Boole etiketi için OverrideStatus a bir enum tipi başvurusu ayarlar.

```
<!-- AWP_In_Variable Name="Data_block_1".ManualOverrideEnable'
Enum="OverrideStatus" -->

<!-- AWP_Enum_Def Name="OverrideStatus" Values='0:"Off",1:"On"' -->
```

HTML sayfası, ManualOverrideEnable'in mevcut durumu için bir Çizelge hücrelerinde bir görüntü alanını içerdiği yerde, sadece normal bir okuma değişkeni komutunu kullanır, ancak önceden beyan edilen ve başvuru alan enum tipi kullanımı ile sayfa, 0 veya 1 den ziyade "Off" ya da "On" görüntüler.

```
<td style="width:24%; border-top-style: Solid; border-top-width:
2px; border-top-color: #ffffff;">
<p>Manual override: := "Data_block_1".ManualOverrideEnable:</p>
</td>
```



HTML sayfası ManualOverrideEnable değerini değiştirmek için kullanıcı için açılan seçim listesi içerir. Seçim listesi, seçim listelerinde gösterilecek "Evet" ve "Hayır" metnini kullanır. Enum tipi kullanımı ile "Evet", enum tipinin "On" değeriyle ilişkili ve "Hayır", "Off" değeriyle ilişkilidir. Boş seçim, ManualOverrideEnable değerini olduğu gibi bırakır.

```
<select name=' "Data_block_1".ManualOverrideEnable'>
<option value=' : "Data_block_1".ManualOverrideEnable: ' > </option>
<option value="On">Yes</option>
<option selected value="Off">No</option>
</select>
```

Seçim listesi HTML sayfasında bir form içinde yer almaktadır. Kullanıcı Gönder düğmesini tıkladığında, sayfa, Data\_block\_1 de Boolean ManualOverrideEnable için kullanıcı "Hayır" seçseydi "0" değerini, "Evet" seçseydi "1" değerini yazan formu açar.

#### 11.6.8.4 Kontrolöre kullanıcı girişini yazma

"Uzak Rüzgar Türbini İzleme" HTML sayfası, kontrolöre (Sayfa 630) verileri yazma için birkaç AWP komutları içerir. Değişkenleri değiştirme ayrıcalıklarına sahip bir kullanıcı rüzgar türbinini manuel kontrol altına koyabilecek şekilde ve türbin hızı, sapma kumandası ve / veya Kanat açısı ayarı için manuel kumandayı etkinleştirebilecek şekilde HTML sayfası Boole değişkenleri için AWP\_In\_Variables bildirir. Sayfa, türbin hızı, sapma, eğim ve fren yüzdesi için kayan nokta değerlerini sonradan ayarlamak için değişkenleri değiştirme ayrıcalıklarına sahip bir kullanıcıya izin vermek için AWP\_In\_Variables kullanır. Sayfa kontrolöre AWP\_In\_Variables yazmak için bir HTTP form sonrası komutu kullanır.

Örneğin, frenleme değerinin manuel ayarlanması için HTML kodunu göz önüne alınız:



#### Örnek HTML kodu

"Uzak Rüzgar Türbini İzleme" HTML sayfasından alınan aşağıdaki alıntı ilk olarak, veri bloğu "Data\_block\_1" de herhangi etiketi yazmak için HTML sayfasını etkinleştiren "Data\_block\_1" için bir AWP\_In\_Variable bildirir. Sayfa bir Çizelge satırının sol hücrelerinde "Frenleme:" metnini görüntüler. Çizelge satırının sağ hücrelerinde Data\_block\_1 in "Frenleme" etiketi için kullanıcı girişini kabul eden alan vardır. Bu kullanıcı giriş değeri, girilen metin verisini CPU' ya göndermek için HTTP yöntemi "POST" u kullanan bir HTML formu içindedir. Sayfa daha sonra kontrolörden gelen gerçek frenleme değerini okur ve veri giriş alanında görüntüler.

Değişkenleri değiştirme ayrıcalıklarına sahip bir kullanıcı daha sonra frenlemeyi kontrol eden CPU'daki veri bloğuna bir frenleme değeri yazmak için bu sayfayı kullanabilir.

```
<!-- AWP_In_Variable Name=' "Data_block_1"' -->
...
<tr style="vertical-align: top; height: 2%;">
<td style="width: 22%;"><p>Braking:</p></td>
<td>
<form method="POST">
<p><input name=' "Data_block_1".Braking' size="10" type="text"> %</p>
</form>
</td>
</tr>
```

**Not**

Bir kullanıcı tanımlı sayfa, bir string veri tipi olan yazılabilir bir veri bloğu etiketi için bir veri giriş alanına sahipse, alana string değeri girilirken, kullanıcının string'i tek tırnak işareti içine alması gerektiğini unutmayınız.

**Not**

<!--AWP\_In\_Variable Name = "Data\_block\_1"--> gibi bir AWP\_In\_Variable bildiriminde tüm bir veri bloğu bildirirseniz, o veri bloğu içindeki her etiket kullanıcı-tanımlı Web sayfasından yazılabilir olduğunu unutmayınız. Bir veri bloğundaki tüm etiketlerin yazılabilir olmasını amaçlıyorsanız bunu kullanınız. Aksi takdirde Sadece belirli bir veri bloğu etiketlerinin kullanıcı-tanımlı Web sayfasından yazılabilir olmasını istiyorsanız, özellikle <!--AWP\_In\_Variable Name = "Data\_block\_1".Braking'--> gibi bir bildirim ile bunu beyan ediniz.

**11.6.8.5 Özel bir değişken yazma**

Kullanıcı değiştirme ayrıcalıklarına sahip olması şartıyla "Uzak Rüzgar Türbini İzleme" Web sayfası, özel değişken SERVER:current\_user\_id'yi CPU'daki bir PLC etiketine yazar. Bu durumda, PLC etiket değeri "Uzak Rüzgar Türbini İzleme" Web sayfasına erişen kullanıcının kullanıcı ID sini içerir.

Web sayfası PLC ye özel değişkeni yazar ve hiçbir kullanıcı ara yüzü gerektirmez.

**Örnek HTML kodu**

```
<!-- AWP_In_Variable Name="SERVER:current_user_id" Use="User_ID"-->
```

**11.6.8.6 Referans: Uzak rüzgar türbini izleme Web sayfasının HTML listesi****Wind\_turbine.html**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<!--
Bu test programı Rüzgar Türbini izlemek ve kontrol etmek için bir
Web sayfasını simüle eder
STEP 7 de Gerekli PLC etiketleri ve Veri Bloğu Etiketleri:
```

```
PLC Tag:
User_ID: Int
```

```
Data Blocks:
Data_block_1
```

```
Tags in Data_Block_1:
```

```
TurbineNumber: Int
WindSpeed: Real
WindDirection: Real
```

```
Temperature: Real
PowerOutput: Real
ManualOverrideEnable: Bool
TurbineSpeed: Real
YawOverride: Bool
Yaw: Real
PitchOverride: Bool
Pitch: Real
Braking: Real
```

Kullanıcı tanımlı Web sayfası PLC verileri için mevcut değerlerini görüntüler ve numaralandırılmış tip ataması kullanan üç Boole değerini ayarlamak için bir seçim listesi sunar. "Gönder" butonu seçilen Boole değerlerinin yanı sıra Türbin hızı, Sapma ve Eğim için veri giriş alanlarını açar. Frenleme değeri "Gönder" butonunu kullanmadan ayarlanabilir.

Gerçek STEP 7 programının bu sayfayı kullanmasına gerek yoktur. İlişkili Booleans ayarlanmış olsaydı teorik olarak, STEP 7 programı sadece, Türbin hızı, Sapma ve Eğim değerleri üzerinde hareket edecekti. Sadece STEP 7 gereksinimi, bu sayfa için oluşturulan veri bloklarının DB numarası ile WWW komutunu çağırmaaktır.

```
-->
<!-- AWP_In_Variable Name="Data_block_1" -->
<!-- AWP_In_Variable Name="Data_block_1".ManualOverrideEnable'
Enum="OverrideStatus" -->
<!-- AWP_In_Variable Name="Data_block_1".PitchOverride'
Enum="OverrideStatus" -->
<!-- AWP_In_Variable Name="Data_block_1".YawOverride'
Enum="OverrideStatus" -->
<!-- AWP_In_Variable Name="SERVER:current_user_id" Use="User_ID"-->
<!-- AWP_Enum_Def Name="OverrideStatus" Values='0:"Off",1:"On"' -->
```

```
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8"><link rel="stylesheet" href="Wind_turbine.css">
<title>Remote Wind Turbine Monitor</title>
</head>
<body>
<Table cellpadding="0" cellspacing="2">
<tr style="height: 2%;">
<td colspan="2">
<h2>Remote Wind Turbine Monitor: Turbine
#:="Data_block_1".TurbineNumber:</h2>
</td>

<tr style="height: 2%;"><td style="width: 25%;"><p>Wind
speed:</p></td>
<td><p> :="Data_block_1".WindSpeed: km/h</p></td>
</tr>

<tr style="height: 2%;">
<td style="width: 25%;"><p>Wind direction:</p></td>
<td><p> :="Data_block_1".WindDirection: deg.</p></td>
</tr>
```

```
<tr style="height: 2%;"><td style="width:
25%;"><p>Temperature:</p></td>
<td><p> := "Data_block_1".Temperature: deg. C</p></td>
</tr>

<tr style="height: 2%;">
<td style="width: 25%;"><p>Power çıkış:</p></td>
<td><p style="margin-bottom:5px;"> := "Data_block_1".PowerOutput:
kW</p>
</td>
</tr>

<form method="POST" action="">
<tr style="height: 2%;" >
<td style="width=25%; border-top-style: Solid; border-top-width:
2px; border-top-color: #ffffff;">
<p>Manual override: := "Data_block_1".ManualOverrideEnable:</p>
</td>
<td class="Text">Set:

<select name=' "Data_block_1".ManualOverrideEnable'>
<option value=' := "Data_block_1".ManualOverrideEnable:'> </option>
<option value="On">Yes</option>
<option value="Off">No</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;"><td style="width:
25%;"><p>Turbine speed:</p></td>
<td>
<p style="margin-bottom:5px;"><input
name=' "Data_block_1".TurbineSpeed' size="10"
value=' := "Data_block_1".TurbineSpeed:' type="text"> RPM</p>
</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 25%;">
<p>Yaw override: := "Data_block_1".YawOverride: </p>
</td>
<td class="Text">Set:

<select name=' "Data_block_1".YawOverride'>
<option value=' := "Data_block_1".YawOverride:'> </option>
<option value="On">Yes</option>
<option value="Off">No</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
```

```
<td style="width: 25%;">
<p>Turbine yaw:</p>
</td>
<td>
<p style="margin-bottom:5px;"><input name=' "Data_block_1".Yaw'
size="10" value=' :="Data_block_1".Yaw:' type="text"> deg.</p>
</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 25%;">
<p>Pitch override: :="Data_block_1".PitchOverride: </p>
</td>
<td class="Text">Set:

<select name=' "Data_block_1".PitchOverride'>
<option value=' :="Data_block_1".PitchOverride:'> </option>
<option value="On">Yes</option>
<option value="Off">No</option>
</select>

</td>
</tr>

<tr style="vertical-align: top; height: 2%;">
<td style="width=25%; border-bottom-style: Solid; border-bottom-
width: 2px; border-bottom-color: #ffffff;">
<p>Blade pitch:</p>
</td>
<td>
<p style="margin-bottom:5px;"><input name=' "Data_block_1".Pitch'
size="10" value=' :="Data_block_1".Pitch:' type="text"> deg.</p>
</td>

</tr>
<tr style="height: 2%;">
<td colspan="2">
<input type="submit" value="Submit override settings and values">
</td>
</tr>
</form>

<tr style="vertical-align: top; height: 2%;">
<td style="width: 25%;"><p>Braking:</p></td>
<td>
<form method="POST" action="">
<p> <input name=' "Data_block_1".Braking' size="10"
value=' :="Data_block_1".Braking:' type="text"> %</p>
</form>
</td>
</tr>
<tr><td></td></tr>
```

```
</Table>
</body>
</html>
```

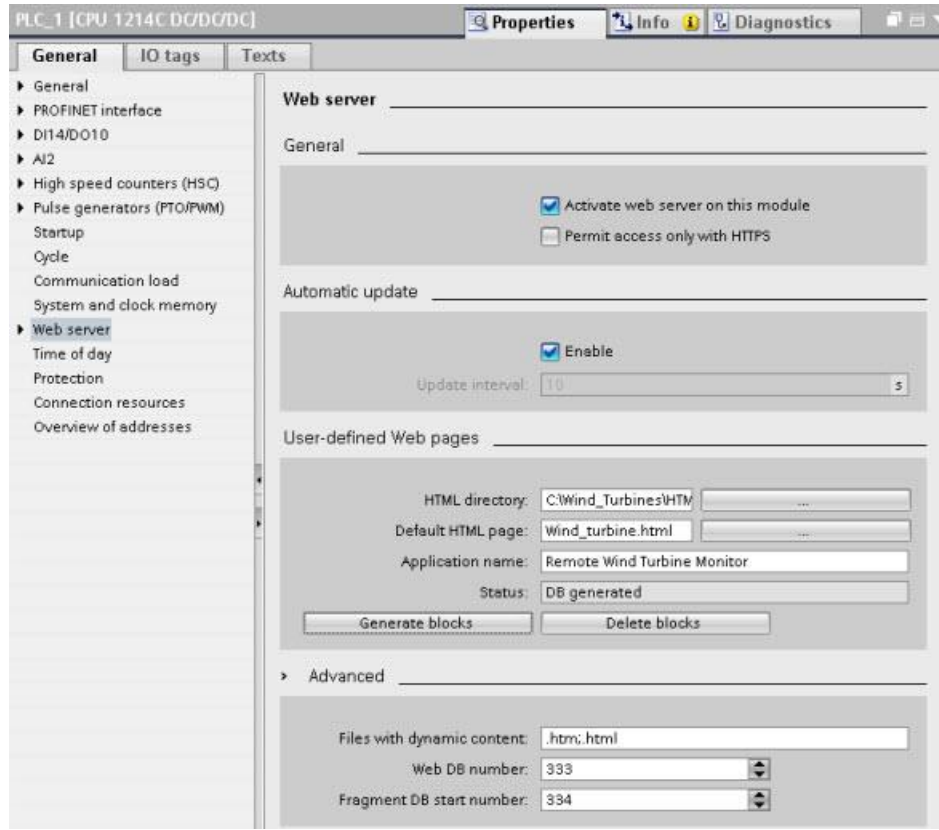
### Wind\_turbine.css

```
BODY {
  background-image: url('./Wind_turbine.jpg');
  background-pozisyonu: 0% 0%;
  background-repeat: no-repeat;
  background-size: cover;
}
H2 {
  font-family: Arial;
  font-weight: bold;
  font-size: 14.0pt;
  color: #FFFFFF;
  margin-top:0px;
  margin-bottom:10px;
}
P {
  font-family: Arial;
  font-weight: bold;
  color: #FFFFFF;
  font-size: 12.0pt;
  margin-top:0px;
  margin-bottom:0px;
}
TD.Text {
  font-family: Arial;
  font-weight: bold;
  color: #FFFFFF;
  font-size: 12.0pt;
  margin-top:0px;
  margin-bottom:0px;
}
```

#### 11.6.8.7 Örnek Web sayfasının STEP 7'de yapılandırması

"Uzak Rüzgar Türbini İzleme" HTML sayfasını S7-1200 için bir kullanıcı-tanımlı Web sayfası olarak eklemek için STEP 7'de HTML sayfası hakkındaki verileri yapılandırır ve HTML sayfasından veri bloklarını oluşturursunuz.

Rüzgar türbinini kontrol eden S7-1200 için CPU özelliklerine erişin ve Web Sunucusunun Kullanıcı tanımlı web sayfaları özelliklerinde yapılandırma bilgilerini giriniz:



## Yapılandırma alanları

- HTML dizin: Bu alan varsayılan sayfanın (giriş sayfası ya da başlangıç sayfası) bilgisayarda bulunduğu klasörün tam nitelikli yol adını belirtir. "..." butonu ihtiyacınız olan klasöre göz atmanızı sağlar.
- Varsayılan HTML sayfası: Bu alan, varsayılan sayfanın veya HTML uygulamasının ana sayfasının dosya adını belirtir. "..." butonu ihtiyacınız olan dosyayı seçmenize izin verir. Bu örnek için WindTurbine.html varsayılan bir HTML sayfasıdır. Uzaktan Rüzgar Türbini İzleme örneği sadece tek bir sayfadan oluşur, ancak diğer kullanıcı-tanımlı uygulamalarda varsayılan sayfa, varsayılan sayfadaki bağlantılardan ek sayfaları çağırabilir. HTML kodu içinde, varsayılan sayfa HTML kaynak klasörüne göre diğer sayfaları referanslamalıdır.
- Uygulama adı: Bu opsiyonel alan, sayfa görüntülenirken Web tarayıcısının adres alanına dahil ettiği adı içerir. Bu örnek için "Uzaktan Rüzgar Türbini İzleme" dir, ancak herhangi bir ad kullanabilirsiniz.

Diğer alanların yapılandırılmasına gerek yoktur.

## Nihai adımlar

Yapılandırılan Uzaktan Rüzgar Türbini İzlemeyi kullanmak için blokları oluşturun, giriş parametresi olarak oluşturulan kontrol DB'nin numarası ile WWW komutunu (Sayfa 642) programlayın, program bloklarını indirin ve CPU'yu çalışma moduna koyunuz.

Bir operatör daha sonra rüzgar türbinini kontrol eden S7-1200 için standart Web sayfalarına eriştiğinde, "Uzaktan Rüzgar Türbini İzleme" Web sayfası, gezinme çubuğundaki "Kullanıcı Sayfaları" linkinden erişilebilir. Bu sayfa artık rüzgar türbinini izlemek ve kontrol etmek için araçlar sağlar.

### 11.6.9 Kullanıcı-tanımlı Web sayfalarını çoklu dilde ayarlama

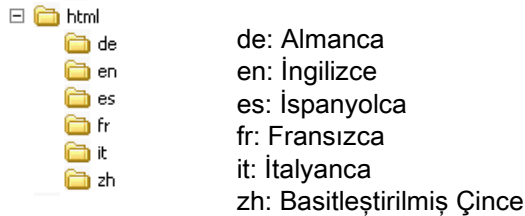
Web sunucusu aşağıdaki dillerde kullanıcı-tanımlı Web sayfaları sağlamak için araçlar sağlar:

- Almanca (de)
- İngilizce (en)
- İspanyolca (es)
- Fransızca (fr)
- İtalyanca (it)
- Basitleştirilmiş Çince (zh)

Dillere karşılık gelen bir klasör yapısındaki (Sayfa 656) HTML sayfalarınızı kurarak ve sayfalarınızdan "siemens\_automation\_language" adlı özel bir çerezi (Sayfa 657) ayarlayarak bunu yapabilirsiniz. Web sunucusu bu çereze yanıt verir ve karşılık gelen dil klasöründe varsayılan sayfaya geçer.

#### 11.6.9.1 Klasör yapısını oluşturma

Kullanıcı-tanımlı Web sayfalarını çoklu dilde sağlamak için HTML dizini altında bir klasör yapısı kurunuz. İki harfli klasör isimleri özeldir ve aşağıda gösterildiği gibi adlandırılması gerekir:



Aynı düzeyde, örneğin sayfalarınızın görüntü veya komutlar (scripts) için klasörlere ihtiyaç duyduğu başka diğer klasörleri de içerebilir.

Dil klasörlerinin herhangi bir kısmını dahil edebilirsiniz. Tüm altı dili de dahil etmek zorunda değilsiniz. Dil klasörleri içinde, uygun dilde HTML sayfaları oluşturur ve programlarsınız.



### 11.6.9.2 Dil geçişini programlama

Web sunucusu "siemens\_automation\_language" adında bir çerez kullanımı ile diller arasında geçiş yapar. Bu, bir tanımlanmış çerezdir ve HTML sayfalarında belirlenir ve aynı adlı dil klasöründen uygun dildeki bir sayfayı görüntülemek için Web sunucusu tarafından yorumlanır. HTML sayfası, önceden tanımlanmış dil kimliklerden ("de", "en", "es", "fr", "o" veya "zh") biri için bu çerezi ayarlamak amacıyla bir JavaScript içermelidir.

Örneğin, HTML sayfası, çerezi "de" olarak ayarlarsa Web sunucusu "de" klasörüne geçer ve STEP 7 yapılandırmasında (Sayfa 659) tanımlandığı gibi varsayılan HTML sayfası adıyla sayfayı görüntüler.

### Örnek

Aşağıdaki örnek, dil klasörlerinin her birinde "langswitch.html" adında varsayılan bir HTML sayfası kullanır. Ayrıca HTML dizininde "script" adlı bir klasör vardır. Script klasörü, "lang.js" adlı bir JavaScript dosyası içerir. Her bir langswitch.html sayfası, dil çerezini, "siemens\_automation\_language" olarak ayarlamak için bu JavaScript'i kullanır.

### "en" klasöründe "langswitch.html" için HTML

HTML sayfasının başlığı, dili İngilizce olarak, karakter setini UTF-8 olarak ayarlar ve yolu JavaScript dosyası lang.js. olarak ayarlar.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english Sayfa</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

Dosya gövdesi bir kullanıcının Almanca ve İngilizce arasında seçim yapması için bir seçim listesini kullanır. İngilizce ("en") önceden seçilen dildir. Kullanıcı dili değiştirdiğinde, sayfa, seçilen opsiyonun değeri ile DoLocalLanguageChange()JavaScript işlevini çağırır.

```
<!-- Language Selection -->
<Table>
  <tr>
    <td align="right" valign="top" nowrap>
      <!-- change language immediately on selection change -->
      <select name="Language"
        onchange="DoLocalLanguageChange(this)"
        size="1">
        <option value="de" >German</option>
        <option value="en" selected >English</option>
      </select>
    </td>
  </tr>
</Table><!-- Language Selection End-->
```

### "de" klasöründeki "langswitch.html" için HTML

Dilin Almanca ayarlanması dışında, Almanca langswitch.html sayfası için başlık, İngilizce ile aynıdır.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de"><meta http-
equiv="Content-Type" content="text/html; charset=utf-8">
<title>Sprachumschaltung Deutsche Seite</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

Almanca sayfasındaki HTML, seçilen dilin varsayılan değerinin Almanca ("de") olması dışında, İngilizce sayfasınıninkine özdeştir.

```
<!-- Language Selection -->
<Table>
  <tr>
    <td align="right" valign="top" nowrap>
      <!-- change language immediately on change of the selection -
->
      <select name="Language"
        onchange="DoLocalLanguageChange(this)"
        <size="1">
        <option value="de" selected >Deutsch</option>
        <option value="en" >Englisch</option>
      </select>
    </td>
  </tr>
</Table><!-- Language Selection End-->
```

### "Script" klasöründeki JavaScript "lang.js"

"DoLocalLanguageChange()" Fonksiyonu lang.js dosyasındadır. Bu fonksiyon, "SetLangCookie ()" işlevini çağırır ve daha sonra HTML sayfasını görüntüleyen pencereyi yeniden yükler.

"SetLangCookie ()" Fonksiyonu belgenin "siemens\_automation\_language" çerezine seçim listesinden gelen değeri atayan bir atama oluşturur. İstenen sayfa değil değiştirilen sayfa çerez değerini alacak şekilde uygulama yolunu ayarlar.

İsteğe bağlı olarak, yorumlanan bölümünde, sayfa, çerez için bir zaman aşımı değeri ayarlayabilir.

```
function DoLocalLanguageChange (oSelect) {
    SetLangCookie (oSelect.value);
    top.window.location.reload();
}
function SetLangCookie (value) {
    var strval = "siemens_automation_language=";
    // Bu, Web sunucusunun istenen dili algıladığı
    // çerezidir
    // Bu ad, Web sunucusu tarafından gereklidir.
    strval = strval + value;
    strval = strval + "; path=/ ";
    // uygulama yolunu ayarlayınız
    // Aksi takdirde yol isteyen sayfaya ayarlanmış olacağından,
    // bu sayfa çerezi almaz.
    /* OPTIONAL
    Bu çerez artık geçerli tarayıcı oturumundan daha uzun
    yaşarsa expiration (süre sonu) kullanınız:
    var now = new Date();
    var endttime = new Date(now.getTime() + expiration);
    strval = strval + "; expires=" +
        endttime.toGMTString() + ";";
    */
    document.cookie = strval;
}
```

### 11.6.9.3 Çoklu dil sayfa yapısını kullanmak için STEP 7 yapılandırma

Çoklu dil kullanıcı-tanımlı Web sayfalarını yapılandırmak için gereken prosedür kullanıcı-tanımlı Web sayfalarını (Sayfa 641) yapılandırmak için gereken genel işleme benzer. Eğer diller için ayarlanmış klasörleriniz varsa, münferit dil klasörleri içeren klasör için HTML dizin ayarını yapınız. HTML dizinini dil klasörlerinden biri olarak ayarlamayınız.

Varsayılan HTML sayfasını seçtiğinizde, dil klasörüne gidin ve HTML sayfasını başlangıç sayfası yapmak üzere seçiniz. Sonradan blokları oluşturduğunuzda ve o blokları CPU'ya indirdiğiniz zaman, Web sunucusu yapılandırılmış dil klasöründe başlangıç sayfasını görüntüler.

Örneğin, burada gösterilen klasör yapısı C:\'de olsaydı, HTML dizin ayarı C:\html olacaktı ve eğer İngilizce ilk sayfa görüntülemesi olsaydı, varsayılan HTML sayfası ayarı için en\langswitch.html'e gidebilirdiniz.



### 11.6.10 Gelişmiş kullanıcı-tanımlı Web sayfası kontrolü

Kullanıcı-tanımlı Web sayfalarınız için veri blokları oluşturduğunuzda, STEP 7 bu kullanıcı-tanımlı sayfaları ile ekran ve etkileşimi kontrol etmek için kullandığı bir kontrol DB oluşturur. STEP 7 ayrıca bireysel sayfaları temsil eden bir grup fragman DB oluşturur. Normal şartlar altında, kontrol DB yapısını veya bunun nasıl kullanıldığını bilmeniz gerekmez.

Örneğin, bir web uygulamasını açmak ve kapatmak, ya da parçaları manuel olarak işlemek istiyorsanız, bunu yapmak için kontrol DB etiketlerini ve WWW komutunu kullanınız.

### Kontrol DB Yapısı

Kontrol DB kapsamlı bir veri yapısıdır ve STEP 7 kullanıcı programını programladığınızda erişilebilir. Sadece kontrol veri bloğu etiketlerinden bazıları burada açıklanmıştır.

### Commandstate yapısı

"Commandstate" Web sunucusu için küresel komutları ve küresel durumları içeren bir yapıdır.

#### "Commandstate" yapıda küresel komutlar

Küresel komutlar genel olarak Web sunucusuna uygulanır. Web sunucusunu kontrol DB parametrelerinden devre dışı bırakabilir veya yeniden başlatabilirsiniz.

Blok etiketi	Veri tipi	Açıklama
init	BOOL	Kontrol DB değerlendirin ve Web uygulamasını başlatınız
deactivate	BOOL	Web uygulamasını devre dışı bırakınız

#### Commandstate yapısında global durumlar

Global durumlar genel olarak Web sunucusuna uygulanır ve Web uygulaması ile ilgili durum bilgilerini içerir.

Blok etiketi	Veri tipi	Açıklama
initializing	BOOL	Web uygulama kontrol DB okuyor
error	BOOL	Web uygulaması başlatılmadı
deactivating	BOOL	Web uygulaması sona eriyor
deactivated	BOOL	Web uygulaması sonlandırıldı
initialized	BOOL	Web uygulaması başlatıldı
last_error	INT	WWW dönüş kodu 16#0010 olduğunda son hata WWW komut çağrısından (Sayfa 642) dönen son hata: 16#0001: Fragman DB yapısı tutarsız 16#0002: uygulama adı zaten var 16#0003: hiçbir kaynak yok (bellek) 16#0004: kontrol DB yapısı tutarsız 16#0005: Fragman DB kullanılamaz 16#0006: Fragman DB, AWP için değildir 16#0007: Numaralandırma verisi tutarsız 16#000D: kontrol DB boyutu çelişkili

### İstek çizelgesi

İstek Çizelgesi, bireysel fragman DB'lere uygulanan komutları ve durumları içeren yapı dizisidir. Eğer fragmanları AWP\_Start\_Fragment (Sayfa 637) "manuel" komut tipi ile oluşturduysanız, STEP 7 kullanıcı programı kontrol DB ile bu sayfaları kontrol etmelidir. İstek durumları salt okunurdur ve anlık fragman hakkında bilgi sağlamaktadır. Anlık fragmanı kontrol etmek için istek komutları kullanınız.

Blok etiketi	Veri tipi	Açıklama
requesttab	ARRAY [ 1 .. 4 ] OF STRUCT	Bireysel fragman DB kontrolü için yapıların dizisi. Web sunucusu, aynı anda dört fragmana kadar işleyebilir. Web sunucusu birden çok tarayıcı oturumundaki fragmanları veya birden çok fragmanı işlerken belirli bir fragman için dizi indeksi keyfidir.

### Requesttab yapısındaki Struct üyeleri

Blok etiketi	Veri tipi	Açıklama
page_index	UINT	Mevcut web sayfası sayısı
fragment_index	UINT	Mevcut fragman sayısı - farklı bir fragmana ayarlanabilir
// İstek komutları		
continue	BOOL	Geçerli sayfanın / parçanın gönderilmesine olanak sağlar ve bir sonraki parça ile devam eder
repeat	BOOL	Geçerli sayfanın / fragmanın yeniden gönderilmesine olanak sağlar ve aynı parça ile devam eder
abort	BOOL	http bağlantısını göndermeden kapatır
finish	BOOL	Bu parçayı gönder; sayfa tamamlandı - herhangi bir ek parça işleme
// İstek durumları		
idle	BOOL	Yapacak bir şey yok, ama aktif
waiting	BOOL	Fragmanın etkin olması bekleniyor
sending	BOOL	Fragman gönderiliyor
aborting	BOOL	Kullanıcı mevcut isteği iptal etti

### Operasyon

Programınız ne zaman kontrol DB'de değişiklik yaparsa yapsın, değiştirilmiş kontrol DB numarası ile WWW komutunu kendi parametresi olarak çağırması gerekir. STEP 7 kullanıcı programı WWW komutunu (Sayfa 642) çalıştırdığında küresel komutlar ve istek komutları etkin olur.

STEP 7 kullanıcı programı parça\_indeksini açıkça ayarlayabilir, bu da Web sunucusunun bir istek komutuyla belirtilen parçanın işlenmesine neden olur. Aksi takdirde WWW komutu çalıştırıldığında Web sunucusu mevcut sayfa için geçerli fragmanı işler.

Fragment\_index kullanan olası teknikler şunlardır:

- Mevcut fragmanı işleme: Fragment\_index'i değiştirmeden bırakın ve devam komutunu ayarlayınız.
- Mevcut fragmanı atlama: Fragment\_index' i 0 olarak ayarlayınız ve devam komutunu ayarlayınız.
- Mevcut fragmanı farklı bir parça ile değiştirme: Fragment\_index' i yeni fragman kimliğine ayarlayınız ve devam komutunu ayarlayınız.

Değiştirilebilen global durumları veya istek durumlarını kontrol etmek için STEP 7 kullanıcı programının bu durumların mevcut değerlerini değerlendirmek için WWW komutunu çağırması gerekir. Tipik bir kullanım, belirli bir durum oluşuncaya kadar periyodik olarak WWW komutunu çağırarak olabilir.

---

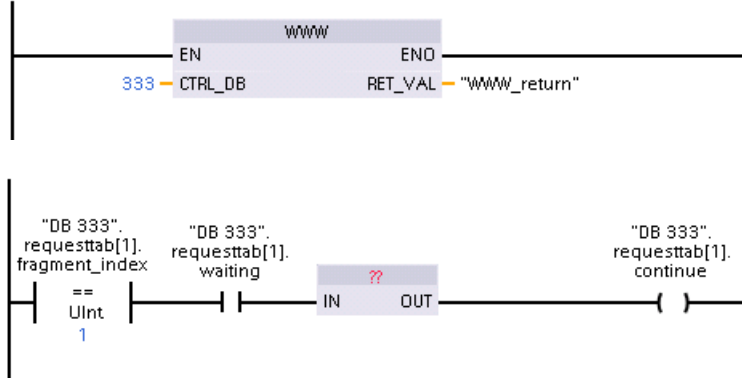
**Not**

STEP 7 kullanıcı programı birden fazla istek komutu ayarlarsa, WWW komutu sadece bu öncelik sırasıyla işler: abort, finish, repeat, continue. WWW komutu işleminden sonra istek komutlarının tümünü temizler.

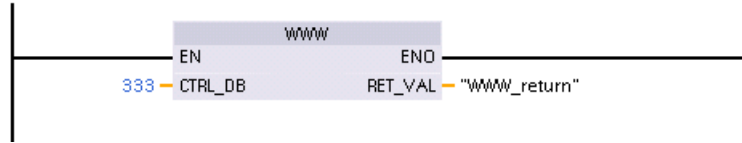
---

## Örnekler

Aşağıdaki örnek, WWW komutunun bir önceki çağrısının ardından, ID si 1 olan bir fragmanın bekleme durumunda olduğunu kontrol eden bir STEP 7 kullanıcı programını gösterir. Diğer uygulamaya özgü koşulların gerçekleşmesi için de beklenebilir. Sonra veri bloğu etiketlerini ayarlama, hesaplamalar yapma veya diğer uygulamaya özgü görevler gibi fragman için ne gerekiyorsa gerçekleştirmelidir. Daha sonra Web sunucusu bu fragmanı yürütecek şekilde devam bayrağını ayarlar.



Program bu değiştirilmiş kontrol DB ile WWW komutunu çağırdığında, bu fragmana sahip kullanıcı-tanımlı Web sayfası Web tarayıcısından görüntülenebilir.



Bunun basitleştirilmiş bir örnek olduğuna dikkat ediniz; kontrol edilecek fragman dizideki dört requesttab yapıardan herhangi birinde olabilir.

## 11.7 Kısıtlamalar

Aşağıdaki IT hususları Web sunucusu kullanımını etkileyebilir:

- Genellikle, standart Web sayfaları ya da kullanıcı-tanımlı Web sayfalarına erişmek için CPU nun IP adresini ya da bir port numarası ile bir kablosuz router'ın IP adresini kullanmanız gerekir. Web tarayıcınız bir IP adresine doğrudan bağlanmaya izin vermezse, BT yöneticinize başvurun. Yerel politikalarınız DNS i destekliyorsa, bu adresin DNS girişi üzerinden IP adresine bağlanabilirsiniz.
- Güvenlik duvarları, proxy ayarları ve diğer sahaya özgü kısıtlamalar da CPU erişimini kısıtlayabilir. Bu sorunları çözmek için IT yöneticinize başvurunuz.
- Standart Web sayfaları JavaScript ve çerezleri kullanır. Web tarayıcı ayarlarınız JavaScript veya çerezleri devre dışı bırakırsa, bunları etkinleştiriniz. Bunları etkinleştiremiyorsanız bazı özellikler (Sayfa 665) kısıtlanır. Kullanıcı-tanımlı Web sayfalarında JavaScript ve çerezlerin kullanımı isteğe bağlıdır. Kullandıysanız, tarayıcınızda bunları etkinleştirmeniz gerekir.

- Web sunucusu Güvenli Yuva Katmanını (SSL) destekler. "ww.xx.yy.zz" CPU'nun IP adresini gösterdiği yerde <http://ww.xx.yy.zz> veya <https://ww.xx.yy.zz> şeklindeki bir URL ile standart Web sayfaları ve kullanıcı-tanımlı Web sayfalarına erişebilirsiniz.
- Siemens, Web sunucusuna güvenli erişim için bir güvenlik sertifikası sağlar. Giriş standart Web sayfasından (Sayfa 613) itibaren sertifikayı indirebilir ve Web tarayıcınızın (Sayfa 666) Internet seçenekleri içine aktarabilirsiniz. Sertifikayı içeri aktarmamayı seçerseniz, <https://> ile Web sunucusuna her erişiminizde güvenlik doğrulama istemiyle karşılaşacaksınız.

### Bağlantı sayısı

Web sunucusu, maksimum 30 aktif HTTP bağlantısını destekler. Kullandığınız Web tarayıcısına ve sayfa başına farklı nesnelerin (.css dosyaları, resimler, ek .html dosyaları) sayısına bağlı olarak, çeşitli eylemler 30 bağlantıyı kullanır. Web sunucusu bir sayfayı görüntülerken bazı bağlantılar devam eder; diğer bağlantılar ilk bağlantıdan sonra devam etmez.

Örneğin, maksimum altı adet kalıcı bağlantıyı destekleyen Mozilla Firefox 8 kullanıyorsanız, Web sunucusu bağlantılardan düşmeye başlamadan önce, beş tarayıcı ya da tarayıcı sekmesi kopyalarını kullanabilirsiniz. Bir sayfanın altı bağlantısının tümünü kullanmadığı durumda, ek tarayıcı veya tarayıcı sekmesi kopyalarına sahip olabilirsiniz.

Ayrıca aktif bağlantıların sayısı sayfa performansını etkileyebileceğinin farkında olunuz.

### Not

#### Web sunucusunu kapatmadan önceki oturum kapama

Web sunucusuna oturum açtıysanız, Web tarayıcınızı kapatmadan önce oturumu kapattığınızdan emin olunuz. Web sunucusu eşzamanlı maksimum yedi oturumu destekler.

### 11.7.1 İnternet seçenekleri JavaScript'i devre dışı bıraktığında özellik kısıtlamaları

Standart Web sayfaları HTML, JavaScript ve çerezleri kullanır. Siteniz JavaScript ve çerezlerin kullanımını kısıtlıyorsa, sayfaların düzgün çalışması için bunları etkinleştiriniz. Web tarayıcınız için JavaScript'i etkinleştiremezseniz, JavaScript kontrollerini kullanan özellikleri çalıştıramazsınız.

### Genel

Sayfalar dinamik olarak güncellenmez. Taze verileri görüntülemek için Yenile simgesi (Sayfa 609) ile sayfayı manuel olarak yenilemeniz gerekir.

### Tanılama arabellek sayfası

Tanılama arabellek sayfası aşağıdaki gibi JavaScript kullanır:

- Olay ayrıntılarını gösterme: JavaScript ile, alt bölümde detayları görmek için tanılama arabelleğinde bir satır seçiniz. JavaScript olmadan, alt bölümdeki olay verilerini görmek için bir tanılama arabellek girişinin olay alanı köprüsünü tıklamanız gerekir.



- Görüntülemek için tanılama arabellek girişlerinin aralığını değiştirme: JavaScript ile görüntülemek için tanılama arabellek girişlerinin aralığını seçmek için en üstte açılır listeyi kullanınız ve sayfa otomatik olarak günceller. JavaScript olmadan, görüntülemek için tanılama arabellek girişlerinin aralığını seçmek için en üstte açılır listeyi kullanınız, ancak o zaman açılır listeden seçtiğiniz aralık ile tanılama arabellek sayfasını güncellemek için "git" bağlantısına tıklamanız gerekir.

"Git" ve olay alanı köprülerinin yalnızca JavaScript etkin olmadığında görünür olduğunu unutmayınız. Onlar gerekli değildir ve bu nedenle JavaScript etkin olduğunda mevcut olmaz.

### Modül Bilgi sayfası

Etkinleştirilmiş JavaScript olmadan, aşağıdaki kısıtlamalar geçerlidir:

- Verileri süzemezsiniz.
- Alanları sıralayamazsınız.

### Değişken Durum sayfası

Etkinleştirilmiş JavaScript olmadan, aşağıdaki kısıtlamalar geçerlidir:

- Her bir değişkeni girdikten sonra, yeni bir değişken girmek için odağı "Yeni Değişken" satırına manuel olarak ayarlamamız gerekir.
- Bir ekran formatını seçme, veri değeri ekranını seçilen formata otomatik olarak değiştirmez. Yeni format ile ekranı yenilemek için "Değeri izle" butonuna tıklamanız gerekir.

## 11.7.2 İnternet seçenekleri çerezlere izin vermediğinde özellik kısıtlamaları

Web tarayıcınızda çerezleri devre dışı bırakırsanız, aşağıdaki kısıtlamalar geçerlidir:

- Oturum açamazsınız.
- Dil ayarını değiştiremezsiniz.
- UTC saatinden PLC saatine geçemezsiniz. Çerezler olmadan, tüm saatler UTC saatindedir.

## 11.7.3 Siemens güvenlik sertifikasını içeri aktarma

"ww.xx.yy.zz" CPU nun IP adresini gösterdiği yerde Web tarayıcınıza <https://ww.xx.yy.zz> girdiğinizde güvenlik doğrulaması istenmeyecek şekilde İnternet seçenekleri içine Siemens güvenlik sertifikasını aktarabilirsiniz. Bir <https://> URL değil de <http://> URL kullanıyorsanız, o zaman sertifikayı indirip yüklemenize gerek kalmaz.

### Sertifikayı yükleme

PC'nize Siemens güvenlik sertifikasını yüklemek için giriş sayfasından "sertifikayı yükle" linkini (Sayfa 613) kullanınız. Prosedür, kullandığınız Web tarayıcısına göre değişir:

**Internet Explorer için sertifikayı aktarma (import)**

1. Giriş sayfasından "sertifikayıyükle" linkini tıklayınız. Bir "Dosya Yükleme - Güvenlik Uyarısı" iletişim kutusu açılır.
2. "Dosya Yükleme - Güvenlik Uyarısı" iletişim kutusundan dosyayı açmak için "Aç" a tıklayınız. Bir "Sertifika" iletişim kutusu görüntülenir.
3. "Sertifika" iletişim kutusundan, Sertifikayı Import Etme Sihirbazı'nı başlatmak için "Sertifikayı kur" butonuna tıklayınız.
4. İşletim sisteminin otomatik olarak sertifika deposunu seçmesine izin vererek, sertifikayı import etmek için "Sertifika Alma Sihirbazı" nın diyaloglarını izleyiniz.

**Mozilla Firefox için sertifikayı aktarma**

1. Giriş sayfasından "sertifikayı yükle" linkini tıklayınız. "MiniWebCA\_Cer.cr açma" iletişim kutusu açılır.
2. "MiniWebCA\_Cer.crt açma" penceresinden " Dosyayı Kaydet " tıklayınız. Bir "Yüklemeler" penceresi görüntülenir.
3. " Yüklemeler " penceresinden, "MiniWebCA\_Cer.crt" çift-tıklayınız. Eğer bir kereden fazla indirme girişiminde bulduysanız, çoklu kopyalar gösterilir. Sadece "MiniWebCA\_Cer.crt" girdilerinden birini çift tıklayınız.
4. Yürütülebilir bir dosyayı açmak isterseniz "Tamam" ı tıklayınız.
5. Eğer görünürse "Dosya aç – Güvenlik Uyarısı" penceresi üzerinde "Aç" a tıklayınız. Bir "Sertifika" penceresi görüntülenir.
6. "Sertifika" penceresinde, "Sertifikayı kur" butonuna tıklayınız.
7. İşletim sisteminin otomatik olarak sertifika deposunu seçmesine izin vererek, sertifikayı indirmek için "Sertifika Alma Sihirbazı"nın diyaloglarını izleyiniz.
8. "Güvenlik Uyarısı" penceresi belirirse, sertifika yüklemeyi onaylamak için "Evet" i tıklayınız.

**Diğer tarayıcılar**

Siemens sertifikasını aktarmak ve kurmak için Web tarayıcınızın kurallarına uyunuz.

Web tarayıcınızın içeriği için internet seçeneklerinde Siemens güvenlik sertifikası "S7-Kontrolör Ailesi" yükledikten sonra [https:// ww.xx.yy.zz](https://ww.xx.yy.zz) ile Web sunucusuna eriştiğinizde, bir güvenlik istemini doğrulamanıza gerek kalmayacaktır.

**Not**

Güvenlik sertifikası CPU nun yeniden başlatılması yoluyla sabit kalır; Ancak, CPU nun IP adresini değiştirirseniz eğer yeni bir sertifika yüklemek için yukarıdaki prosedürleri takip etmeniz gerekir.

#### 11.7.4 Microsoft Excel'in USA/UK dışındaki sürümleri için CSV formatında veri günlüklerini aktarma

Veri günlük dosyaları, virgülle ayrılmış değerler (CSV) dosya formatındadır. Sisteminiz Excel in ABD veya İngiltere sürümünde çalışırken bu dosyaları veri kayıtları sayfasından doğrudan Excel'de açabilirsiniz. Virgül sayısal gösterimde sık olduğu için diğer ülkelerde, bu format yaygın olarak kullanılmamaktadır.

Kaydettiğiniz bir veri günlük dosyasını açmak için Excel in ABD/İngiltere dışındaki sürümleri için şu adımları izleyiniz:

1. Excel i açınız ve boş bir çalışma kitabı oluşturun.
2. "Veri> Dış Veri Al" menüsünden, "Veri Alma" komutunu seçiniz.
3. Açmak istediğiniz veri günlük dosyasına gidin ve seçiniz. Metin Alma Sihirbazı başlar.
4. Metin Alma Sihirbazından, "Orijinal veri tipi" için varsayılan seçeneği "Sabit genişlik" ten "Ayrılmış" a değiştirin.
5. İleri düğmesine tıklayınız.
6. 2. Adım penceresinde, "Virgül" onay kutusunu seçiniz. ayırma türünü "Sekme" den "Virgül"e değiştirmek için
7. İleri düğmesine tıklayınız.
8. 3. Adım penceresinde, isteğe bağlı olarak Tarih biçimini MDY (ay / gün / yıl) dan başka bir biçime değiştirebilirsiniz.
9. Dosyasını almak için Metin Alma Sihirbazının kalan adımlarını tamamlayınız.



## Haberleşme işlemcisi ve Modbus TCP

### 12.1 Seri haberleşme ara yüzleri kullanarak

İki haberleşme modülü (CM'ler) ve bir haberleşme kartı (CB) PtP haberleşme için ara yüz sağlar:

- CM 1241 RS232 (Sayfa 975)
- CM 1241 RS422/485 (Sayfa 976)
- CB 1241 RS485 (Sayfa 973)

Üç (herhangi bir tip) kadar CM ve bir de CB olmak üzere toplam dörde kadar haberleşme ara yüzü bağlayabilirsiniz. CPU soluna veya başka CM ye CM monte ediniz. CPU ön tarafına CB monte ediniz. S7-1200 Sistemi Kılavuzunda (Sayfa 62) Modül montajı ve kaldırılması hakkında ayrıntılı talimatlar için "Montaj" bölümüne bakınız.

Seri haberleşme ara yüzleri aşağıdaki özelliklere sahiptir:

- İzole port var
- Noktadan Noktaya protokolleri destekler
- Noktadan-noktaya haberleşme işlemci komutları aracılığıyla yapılandırılır ve programlanır
- LED'ler yoluyla gönderme ve alma aktivitesini görüntüler
- Tanılama LED Ekranı (sadece CM'ler)
- CPU tarafından enerjilendirilir: Harici güç bağlantısı gerekli değildir.

Haberleşme ara yüzleri (Sayfa 964) için teknik özelliklerine bakınız.

### LED göstergeler

Haberleşme modüllerinde üç LED göstergesi vardır:

- Tanılama LED (DIAG): Bu LED, CPU tarafından adreslenene kadar kırmızı yanıp söner. CPU enerjilendikten sonra, CM'leri denetler ve onları adresler. Tanılama LED'i yeşil yanıp sönmeye başlar. Bu durum, CPU nun CM'yi adreslediği, ancak henüz kendisine yapılandırma sunmadığı anlamına gelir. Program CPU'ya indirildiği zaman CPU, yapılandırılmış CM'lere yapılandırmayı yükler. CPU' ya indirme işleminden sonra, haberleşme modülü üzerindeki tanılama LED i sabit yeşil renkte olmalıdır.
- İletim LED (Tx): Veri, haberleşme portuna iletilirken iletim LED i yanar.
- Alma LED (Rx): Bu LED veriler haberleşme portu tarafından alınırken yanar

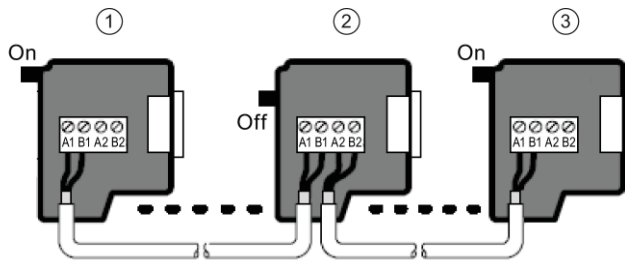
Haberleşme kartı, (TxD) iletim ve (RxD) alma LED'leri sağlar. Hiçbir tanılama LED i yoktur.

## 12.2 RS485 ağ konektörü öngerilimleme ve sonlandırma

Siemens, bir RS485 ağına kolayca birden fazla cihaz bağlamak için kullanabileceğiniz bir RS485 ağ konektörü (Sayfa 991) sağlar. Konektörün, gelen ve giden ağ kablolarını takmanıza izin veren iki takım terminali vardır. Konektör, de seçici olarak ağı öngerilimleyen ve sonlandıran anahtarlar içerir.

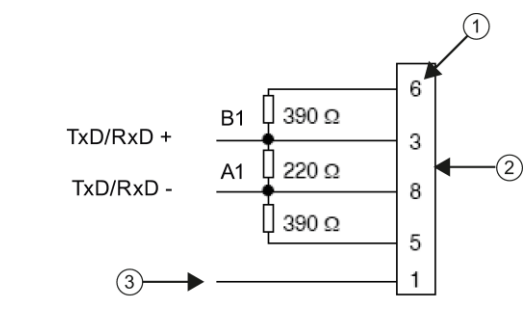
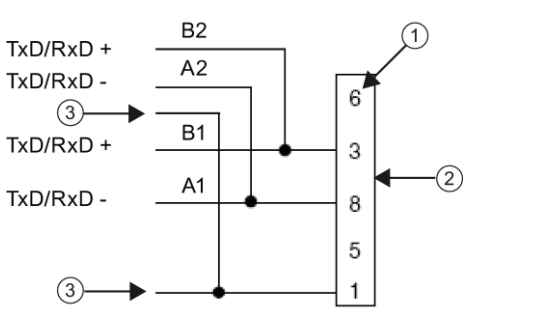
### Not

RS485 ağın sadece iki ucunu sonlandırır ve öngerilimlersiniz. İki uç cihaz arasındaki cihazlar, sonlandırılmaz veya öngerilimlenmez. Çıplak kablo zırhı: Yaklaşık 12 mm (1/2 inç) tüm yerlerin metal kılavuzlarına temas etmelidir.



- ① Anahtar pozisyonu = On: Sonlandırılmış ve öngerilimlenmiş
- ② Anahtar pozisyonu = Off: Sonlandırma veya öngerilimleme yok
- ③ Anahtar pozisyonu = On: Sonlandırılmış ve öngerilimlenmiş

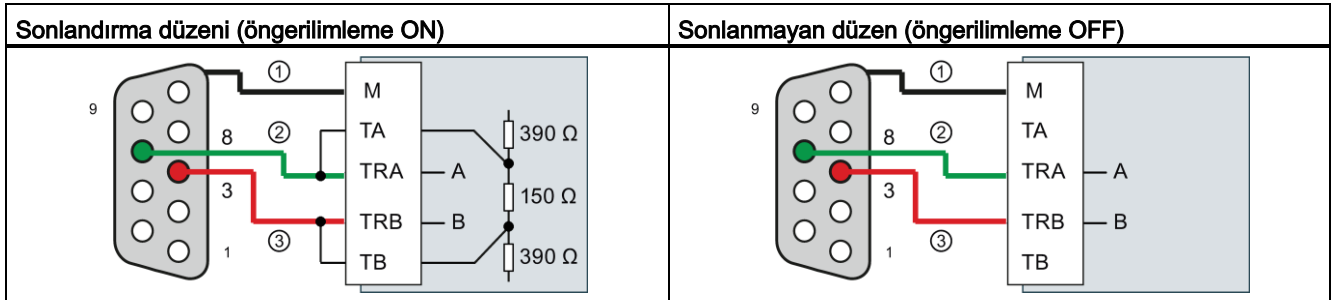
Çizelge 12- 1 RS485 konektör için sonlandırma ve öngerilimleme

Sonlandırma düzeni (öngerilim ON)	Sonlanmayan düzen (öngerilim OFF)
	

- ① Pin numarası
- ② Ağ konektörü
- ③ Kablo zırhı

CB 1241 ağı sonlandırma ve öngerilimleme için iç dirençler sağlar. Bağlantıyı sonlandırma ve öngerilimleme için, iç dirençleri devreye dahil etmek için TRA'yı TA'ya ve TRB'yi TB'ye bağla. CB 1241, 9-pin konektöre sahip değil. Aşağıdaki çizelgede, haberleşme partneri üzerinde bir 9-pin konektör ile yapılan bağlantılar gösterilmiştir.

Çizelge 12- 2 CB 1241 için sonlandırma ve öngerilimleme



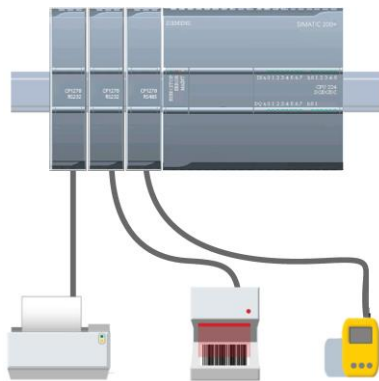
- ① M kablo zırhına bağlanacak
- ② A = TxD/RxD - (Yeşil tel / Pin 8)
- ③ B = TxD/RxD + (Kırmızı tel / Pin 3)

## 12.3 Noktadan noktaya (PtP) haberleşme

CPU karakter tabanlı seri protokoller için aşağıdaki Noktadan Noktaya haberleşmeyi (PtP) destekler:

- PtP
- USS (Sayfa 712)
- Modbus (Sayfa 727)

PtP maksimum özgürlük ve esneklik sağlar, ancak kullanıcı programında kapsamlı uygulama gerektirir.



PtP, çok çeşitli bir olanak sağlar:

- Yazıcı gibi harici bir cihaza doğrudan bilgi gönderme yeteneği
- Barkod okuyucular, RFID okuyucular, üçüncü taraf kamera ya da görüntü sistemleri ve diğer birçok aygıt tipleri gibi diğer cihazlardan bilgi alma yeteneği
- GPS cihazları, üçüncü taraf kamera ya da görüntü sistemleri, radyo modemler ve daha birçok diğer cihazlar ile veri gönderme ve alma, bilgi alışverişi yeteneği

PtP'nin bu haberleşme türü, çeşitli baud hızları ve parite seçeneklerini desteklemek için standart UART'lar kullanan seri haberleşmedir. RS232 ve RS422/485 haberleşme modülleri (CM 1241) ve RS485 haberleşme kartı (CB 1241) PtP haberleşme gerçekleştirmek için elektrik ara yüzleri sağlar.

### 12.3.1 Haberleşme portları yapılandırma

Haberleşme ara yüzlerini yapılandırmak için aşağıdaki yöntemlerden birini kullanabilirsiniz:

- Port parametrelerini (baud ve parite), gönderme parametreleri ve alma parametrelerini yapılandırmak için STEP 7'de cihaz yapılandırmasını kullanınız. CPU, Cihaz yapılandırma ayarlarını depolar ve bir güç çevriminden ve RUN-STOP geçişinden sonra ayarları uygular.
- Parametreleri ayarlamak için PORT\_CFG (Sayfa 686), SEND\_CFG (Sayfa 688), ve RCV\_CFG (Sayfa 689) komutlarını kullanınız. CPU, RUN modundayken komutlar tarafından belirlenen port ayarları geçerlidir. Port ayarları STOP geçişi ya da güç çevriminden sonra cihaz yapılandırma ayarlarına döner.

Donanım cihazlarını yapılandırdıktan sonra (Sayfa 137), şasi veya CB'nizdeki CM'lerden birini (yapılandırılmış ise) seçerek haberleşme ara yüzleri için parametreleri yapılandırınız.



Denetim penceresinin "Properties" sekmesi, seçilen CM veya CB parametrelerini görüntüler. Aşağıdaki parametreleri düzenlemek için "Port yapılandırma" seçeneğini seçiniz:

- Baud hızı
- Parite
- Karakter başına veri bitleri
- Dur bit sayısı
- Akış kontrolü (RS232 için)
- Bekleme zamanı

CM 1241 RS232 ve CB RS485 için (akış kontrolü haricinde(Sayfa 674), sadece CM 1241 RS232'nin desteklediği), bir RS232 veya RS485 haberleşme modülü veya RS485 haberleşme kartını yapılandırıp yapılandırmamasından bağımsız olarak port yapılandırma parametreleri aynıdır. Parametre değerleri farklı olabilir.

CM 1241 RS422/485 için, aşağıda gösterildiği gibi port yapılandırma için ek seçenekler vardır. CM 1241 RS422/485 modülünün 422 modu da yazılım akış kontrolünü destekler.





Aşağıdaki RS422/485 parametrelerini düzenlemek için "Port yapılandırma" seçeneğini seçiniz:

- " Çalışma modu ":
  - Tam çift yönlü (RS422) dört kablo modu (noktadan noktaya bağlantı)
  - Tam çift yönlü (RS422) dört kablo modu (çok noktalı master)
  - Tam çift yönlü (RS422) dört kablo modu (çok noktalı slave)
  - Yarım çift yönlü (RS485) iki kablo modu
- "Hat başlangıç durumunu alınız ":
  - Yok
  - İleri besleme (Sinyal R (A) 0V, sinyal R (B) 5V)

STEP 7 kullanıcı programı da portu yapılandırabilir veya PORT\_CFG komutu (Sayfa 686) ile mevcut yapılandırmasını değiştirebilir.

#### Not

Kullanıcı programındaki PORT\_CFG komutundan ayarlanan parametre değerleri cihaz yapılandırmasından ayarlanan port yapılandırma ayarlarını geçersiz kılar. S7-1200 ün, güç kesilmesi durumunda PORT\_CFG komutundan ayarlanan parametreleri tutmadığını unutmayınız.

Parametre	Tanım
Baud hızı	Baud hızı için varsayılan değer saniyede 9,6 Kbits. Geçerli seçimler şunlardır: 300 baud, 600 baud, 1.2 Kbits, 2.4 Kbits, 4.8 Kbits, 9.6 Kbits, 19.2 Kbits, 38.4 Kbits, 57.6 Kbits, 76.8 Kbits ve 115.2 Kbits.
Parite	Parite için varsayılan değer, parite yok tur. Geçerli seçenekler şunlardır: Parite yok, çift, tek, işaret (parite biti her zaman 1 olarak ayarlanır), ve boşluk (parite biti her zaman 0'a ayarlanır).
Karakter başına veri biti	Bir karakterdeki veri bitlerinin sayısı. Geçerli seçenekler 7 ya da 8.
Durma bitlerinin sayısı	Durma bit sayısı bir veya iki olabilir. Varsayılan biridir.
Akış kontrolü	RS232 haberleşme modülü için, donanım veya yazılım akış kontrolü (Sayfa 674) seçebilirsiniz. Eğer donanım akış kontrolünü seçerseniz, RTS sinyali her zaman açık olup olmayacağını ya da RTS'nin anahtarlanacağını seçebilirsiniz. Yazılım akış kontrolü seçerseniz, XON ve XOFF karakterlerini tanımlayabilirsiniz. RS485 haberleşme ara yüzleri, akış kontrolünü desteklemez. CM 1241 RS422/485 modülünün 422 modu, yazılım akış kontrolünü destekler.
Bekleme süresi	Bekleme süresi, akış kontrolü tipine bağlı olarak, bir XOFF aldıktan sonra bir XON almak için yada RTS belirttikten sonra CTS almak için CM veya CB'nin bekleyeceği süreyi belirtir. Haberleşme ara yüzü beklenen bir CTS veya XON almadan önce bekleme süresi dolarsa CM veya CB gönderme işlemi iptal eder ve kullanıcı programı için bir hata döndürür. Milisaniye cinsinden bekleme süresini belirtiniz. Aralık 0 ile 65535 milisaniye arasındadır.
Çalışma modu	Bu, çalışma modu RS422 veya RS485 ve ağ yapılandırmalarını seçer.
Alma hattı başlangıç durumu	Bu öngerilim seçeneklerini seçer. Geçerli değerler yok, ileri besleme ve ters besleme. Ters besleme, kablo kopması algılamasını sağlamak için kullanılır.

### 12.3.1.1 Akış kontrolü yönetme

Akış kontrolü, hiçbir veri kaybolmayacak şekilde veri iletimiyle ilgili gönderme ve almayı dengeleme için bir mekanizmaya başvurur. Akış kontrolü, iletilen cihaz, alıcı cihazın işleyebileceğinden daha fazla bilgi göndermemesini sağlar. Akış kontrolü, donanım veya yazılım aracılığıyla gerçekleştirilebilir. RS232 CM donanım ve yazılım akış kontrolünü destekler. RS485 CM ve CB'nin akış kontrolü desteği yoktur. CM 1241 RS422/485 modülünün 422 modu, yazılım akış kontrolünü destekler. Portu yapılandırıldığında (Sayfa 672) veya PORT\_CFG komutu ile (Sayfa 686) akış kontrolü türünü belirtiniz.

Donanım akış kontrolü, istek üzerine gönderme (RTS) ve temizleme için gönderme (CTS) haberleşme sinyalleri üzerinden çalışır. RS232 CM ile, RTS sinyali, pin 7 den gönderilir ve CTS sinyali pin 8 üzerinden alınır. RS232 CM, bir giriş olarak CTS'yi izleyen ve bir çıkış olarak RTS belirten bir DTE (Veri terminal donanımı) cihazıdır.

#### Donanım akış kontrolü : RTS anahtarlamalı

RS232 CM için RTS anahtarlamalı donanım akış kontrolünü etkinleştirirseniz, modül, veri göndermek için RTS sinyalini aktif olarak ayarlar. Bu, alıcı cihazın veriyi kabul edip edemeyeceğini belirlemek için CTS sinyalini izler. CTS sinyali aktif olduğu sürece, modül veri iletebilir. CTS sinyali pasif olursa, o zaman iletim durdurulmalıdır.

CTS sinyali aktif olduğunda iletim devam eder. CTS sinyali yapılandırılmış bekleme süresi içinde aktif hale gelmezse, modül iletimi durdurur ve kullanıcı programı için bir hata döndürür. Port yapılandırmasında ( Sayfa 672) bekleme süresini belirtiniz.

RTS anahtarlamalı akış kontrolü, iletimin aktif olduğuna dair bir sinyal gerektiren cihazlar için yararlıdır. Radyo vericisini enerjilendirmek için bir " anahtar " sinyal olarak RTS kullanan bir radyo modem Bir örnek olarak verilir. RTS anahtarlamalı akış kontrolü, standart telefon modemlerle çalışmaz. Telefon modemler için her zaman seçim üzerine RTS kullanınız.

#### Donanım akış kontrolü: RTS her zaman açık

Her zaman açık modda olan RTS'de, CM 1241, varsayılan olarak RTS'yi aktif ayarlar. Telefon modem gibi bir cihaz CM den gelen RTS sinyalini izler ve bir gönderme-için-temizleme olarak bu sinyali kullanır. Modem yalnızca, RTS aktif olduğunda yani telefon modem aktif CTS'yi gördüğünde CM'ye iletir. RTS aktif değilse, telefon modülü CM ye iletmez.

Modemin herhangi bir zamanda CM'ye veri göndermesine izin vermek için donanım akış kontrolü "her zaman RTS açık" olarak yapılandırılır. CM böylece RTS sinyalini her zaman aktif olarak ayarlar. Modül karakterleri kabul edemiyor olsa bile CM, RTS'yi pasif olarak ayarlamayacaktır. İletim cihazı, bunun, CM'nin alma arabelleğinden taşmamasını sağlar.

#### Veri Terminali Hazır (DTR) ve Veri Seti Hazır (DSR) sinyali kullanımı

CM, donanım akış kontrolünün iki türü için DTR'yi aktif olarak ayarlar. Modül, yalnızca DSR sinyali aktif hale geldiğinde iletir. DSR durumu sadece, gönderme işleminin başlangıcında değerlendirilir. DSR, iletim başladıktan sonra pasif hale gelirse, iletim durdurulmayacaktır.

#### Yazılım akış kontrolü

Yazılım akış kontrolü, akış kontrolü sağlamak için mesajlarda özel karakterler kullanmaktadır. XON ve XOFF temsil eden Hex karakterleri yapılandırınız.

XOFF, bir iletimin durdurulması gerektiğini gösterir. XON bir iletimin devam edebileceğini gösterir. XOFF ve XON aynı karakter olmamalıdır.

Verici cihaz, alıcı cihazdan bir XOFF karakteri aldığı anda, verici iletimi durur. Verici cihaz bir XON karakteri aldığı anda iletim devam eder. Bu, port yapılandırmasında ( Sayfa 672 ) belirtilen bekleme süresi içinde bir XON karakteri almazsa, CM iletimi durdurur ve kullanıcı programı için bir hata döndürür.

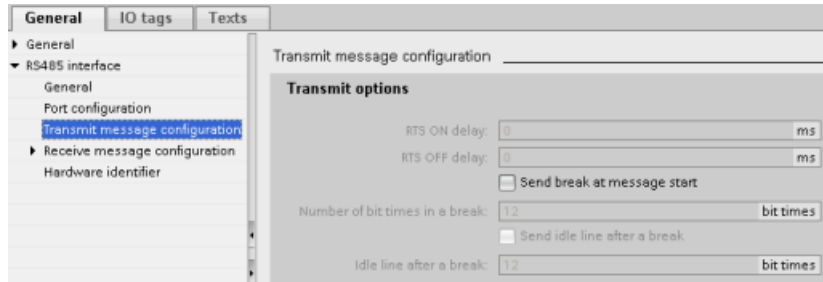
Alıcı partnerin, bir iletim sürerken verici partnere XOFF gönderebilmesi gerektiği için yazılım akış kontrolü, tam çift yönlü haberleşme gerektirir. Yazılım akış kontrolü yalnızca ASCII karakterleri içeren mesajlar ile mümkündür. İkili protokoller yazılım akış kontrolünden yararlanamazlar.

## 12.3.2 İletim (gönderme) ve alma parametrelerini yapılandırma

CPU, PtP haberleşmelerini kurabilmeden önce, iletilen (veya gönderme) mesajlar ve alınan mesajlar için parametreleri yapılandırmanız gerekir. Bu parametreler, mesajlar iletilirken ya da bir hedef cihazdan alınıyorken haberleşmenin nasıl işlediğini belirtir.

### 12.3.2.1 İletim (gönderme) parametrelerini yapılandırma

CPU cihaz yapılandırmasından, bir haberleşme ara yüzünün, seçilen ara yüz için "İletim mesajı yapılandırması" özelliklerini ayarlayarak verileri nasıl ileteceğini yapılandırabilirsiniz.



Ayrıca SEND\_CFG (Sayfa 688) komutu ile kullanıcı programından iletim mesajı parametrelerini dinamik olarak yapılandırabilir veya değiştirebilirsiniz.

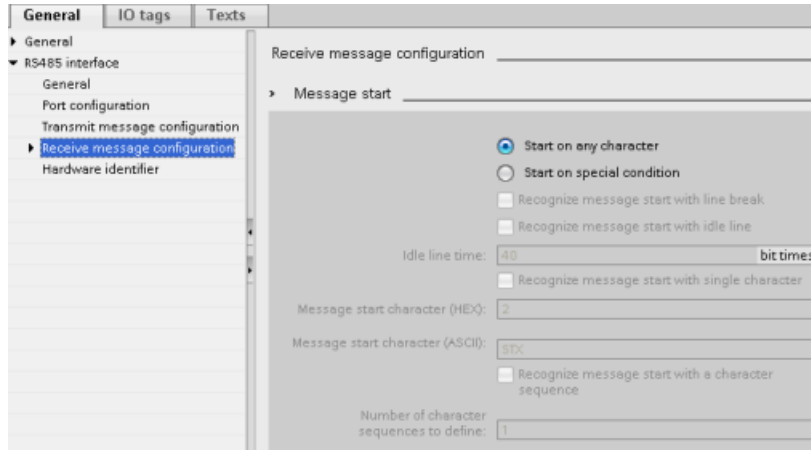
#### Not

Kullanıcı programında SEND\_CFG komutundan ayarlanan parametre değerleri "İletim mesajı yapılandırması" özelliklerini geçersiz kılar. CPU nun, güç kesilmesi durumunda SEND\_CFG komutundan ayarlanan parametreleri tutmayacağını unutmayınız.

Parametre	Tanımlama
RTS ON gecikme	İletim başlamadan önce RTS aktif edildikten sonra beklenecek süreyi belirtir. 0 varsayılan değer olmak üzere aralık 0 ile 65535 ms'dir. Bu parametre yalnızca Port yapılandırma (Sayfa 672) donanım akış kontrolünü belirttiğinde geçerlidir. RTS açık gecikme süresi dolduktan sonra CTS değerlendirilir. Bu parametre sadece RS232 modülleri için geçerlidir.
RTS OFF gecikme	İletimin tamamlanmasının ardından RTS pasif edilmeden önce beklenecek süreyi belirtir. 0 varsayılan değer olmak üzere Aralık 0 ile 65535 ms'dir. Bu parametre yalnızca Port yapılandırma (Sayfa 672) donanım akış kontrolünü belirttiğinde geçerlidir. Bu parametre sadece RS232 modülleri için geçerlidir.
Mesaj başlangıcında gönderme kesmesi Bir kesmedeki bit sürelerinin sayısı	Her mesajın başlamasından sonra, RTS açık gecikme (yapılandırıldığı takdirde) dolduktan ve CTS aktif olduktan sonra bir kesme gönderilecek olduğunu belirtir. Kaç tane bit süresinin, satırın bir boşlukta tutulduğu yerde bir kesme oluşturduğunu belirtir. Varsayılan değer 12 ve maksimum 65535' tir, sekiz saniyelik bir limite kadar.
Kesmeden sonra boş satır gönder Kesmeden sonra boş satır	Mesaj başlamadan önce Boş satır gönderileceğini belirtir. Kesme yapılandırılırsa eğer, bu kesmeden sonra gönderilir. " Kesmeden sonra boş satır " Parametreleri, satırın işaretleme durumunda tutulduğu yerde, kaç tane bit süresinin boş satır oluşturacağını belirtir. Varsayılan değer 12 ve maksimum 65535 tir, sekiz saniyelik bir limite kadar.

### 12.3.2.2 Alma parametrelerini yapılandırma

CPU'nun cihaz yapılandırmasından, bir haberleşme ara yüzünün nasıl veri alacağını ve bir mesajın hem başlangıcını ve hemde sonunu nasıl bir tanıyacağını yapılandırabilirsiniz. Seçilen ara yüzü için "mesaj yapılandırmasını al" özelliklerinde bu parametreleri ayarlarsınız.



Ayrıca RCV\_CFG (Sayfa 689) komutu ile kullanıcı programından alma mesajı parametrelerini dinamik olarak yapılandırabilir veya değiştirebilirsiniz.

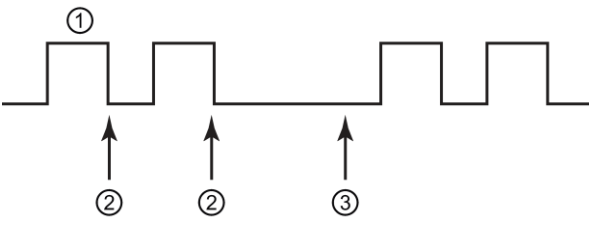
#### Not

Kullanıcı programında RCV\_CFG komutundan ayarlanan parametre değerleri "mesaj yapılandırmasını al" özelliklerini geçersiz kılar. CPU nun, güç kesilmesi durumunda RCV\_CFG komutundan ayarlanan parametreleri tutmayacağını unutmayınız.

## Mesaj başlangıç koşulları

Haberleşme ara yüzünün, bir mesajın başlangıcını nasıl tanıdığı belirlebilirsiniz. Yanılandırılmış bir uç kosulu sağlanana kadar başlangıç karakterleri ve mesaj içeren karakterleri alıcı ara belleğine gider.

Çoklu başlangıç koşulları belirlebilirsiniz. Birden fazla başlangıç kosulu belirtirseniz, mesaj başlamış kabul edilmeden önce, başlangıç koşullarının tümünün yerine getirilmesi gerekir. Örneğin eğer boş hat süresini ve belirli bir başlangıç karakterini yanılandırırırsanız, CM veya CB önce karşılanması gereken boş satır süre gereksinimini aravacaktır ve ardından CM belirtilen başlangıç karakterini arar. Bazı diğer karakterler (belirtilen başlangıç karakteri değil) alınırsa, CM veya CB yine boş hat süresine bakarak mesajın başlangıcını yeniden başlatacaktır.

Parametre	Tanım
Herhangi Karakterden başla	Herhangi karakter koşulu, başarıyla alınan herhangi bir karakterin bir mesajın başlangıcını gösterdiğini belirtir. Bu karakter, bir mesajın içindeki ilk karakterdir.
Satır kesmesi	Satır kesmesi koşulu, bir mesaj alımı işleminin bir kesme karakteri alındıktan sonra başladığını belirtir.
Boş satır	Boş satır koşulu, alıcı satır, bir kere, belirtilen bit sürelerinin sayısı için boşta veya sessiz olduğunda mesaj alımının başladığını belirtir. Bu durum gerçekleştiğinde, bir mesajın başlangıcı başlar.   <p>① Karakterler            ② Boş satır zamanlayıcıyı yeniden başlatır            ③ Boş satır tespit edilir ve mesaj alımı başlatılır</p>
Özel durum: Tek bir karakter ile mesaj başlangıcını tanımasına	Özel bir karakterin bir mesajın başlangıcını gösterdiğini belirtir. Bu karakter bir mesaj içindeki ilk karakterdir. Bu özel karakterden önce, alınan herhangi bir karakter atılır. Varsayılan karakter STX'tir.
Özel durum: Bir karakter dizisiyle mesaj başlangıcını tanıma	Dörde kadar yapılandırılmış dizilerden belirli bir karakter dizisinin bir mesajın başlangıcını gösterdiğini belirtir. Her dizi için, beş karaktere kadar belirtebilirsiniz. Her karakter konumu için, belirli bir onaltılık karakter veya karakterin, dizi eşleştirmede (joker karakter) göz ardı edildiğini belirtirsiniz. Bir karakter dizisinin son belirli bir karakteri, o başlatma koşulu dizisini sona erdirir.  Bir başlatma koşulu karşılanan kadar, gelen diziler yapılandırılmış başlatma koşullarına karşı değerlendirilir. Başlatma dizisi karşılandıktan sonra, uç koşulların değerlendirilmesi başlar.  Dörde kadar belirli karakter dizilerini yapılandırabilirsiniz. Farklı karakterler dizisi bir mesajın başlangıcını gösterebildiği zaman bir çoklu-dizi başlatma koşulunu kullanınız. Karakter dizilerinin herhangi biri eşleştirilirse, mesaj başlatılır.

Başlangıç koşullarını kontrol sırası:

- Boş satır
- Satır kesmesi
- Karakterler veya karakter dizileri

Çoklu başlangıç koşullarını kontrol ederken, koşullardan biri yerine getirilmediği takdirde, CM veya CB, ilk gerekli koşul ile kontrolü yeniden başlatır. CM veya CB, başlangıç koşullarının karşılanmış olduğunu sağladıktan sonra, uç koşullarının değerlendirmesini başlatır.

### Örnek yapılandırma - iki karakter dizisinin birinde mesajı başlatma

Aşağıdaki başlangıç mesajı koşulu yapılandırmasını göz önüne alınız:

Recognize message start with a character sequence

Number of character sequences to define:

**5-character message start sequences**

**Message start sequence 1**

Inspect character 1

Character value (HEX):

Character value (ASCII):

Inspect character 2

Character value (HEX):

Character value (ASCII):

Inspect character 3

Character value (HEX):

Character value (ASCII):

Inspect character 4

Character value (HEX):

Character value (ASCII):

Inspect character 5

Character value (HEX):

Character value (ASCII):

**Message start sequence 2**

Inspect character 1

Character value (HEX): 0

Character value (ASCII): any

Inspect character 2

Character value (HEX): 6A

Character value (ASCII):

Inspect character 3

Character value (HEX): 6A

Character value (ASCII):

Inspect character 4

Character value (HEX): 0

Character value (ASCII): any

Inspect character 5

Character value (HEX): 0

Character value (ASCII): any

Bu yapılandırmayla aşağıdaki iki yapıdan biri oluştuğunda, başlangıç koşulu karşılanır:

- İlk karakter 0x6A ve beşinci karakter 0x1C olduğu yerde beş karakter dizisi alındığında. Pozisyon 2, 3 ve 4 deki karakterler bu yapılandırma ile herhangi bir karakter olabilir. Beşinci karakter alındıktan sonra, uç koşulların değerlendirilmesi başlar.
- Herhangi bir karakter öncesinde iki ardışık 0x6A karakteri alındığında. Bu durumda, ikinci 0x6A (3 karakter) alındıktan sonra, uç koşulların değerlendirilmesi başlar. İlk 0x6A dan önceki karakter, başlangıç koşulunda dahil edilir.

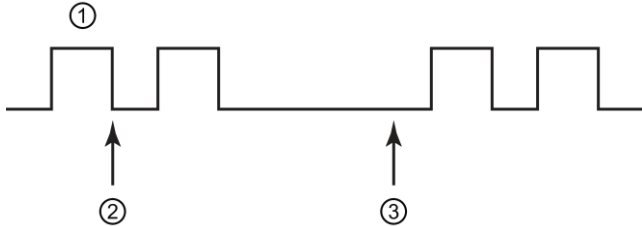
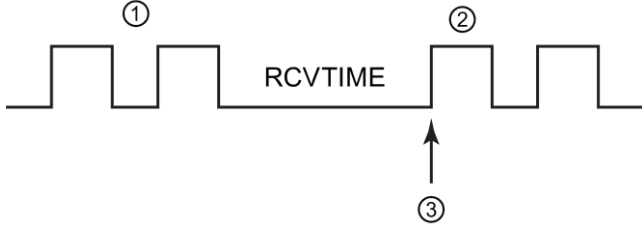
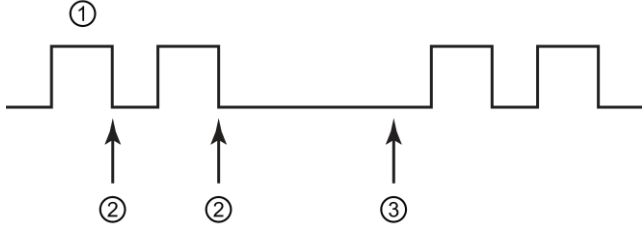
Bu başlangıç koşulunu karşılayacak örnek diziler aşağıda verilmiştir:

- <Herhangi bir karakter> 6A 6A
- 6A 12 14 18 1C
- 6A 44 A5 D2 1C

## Mesaj sonu koşulları

Haberleşme ara yüzünün, bir mesajın sonunu nasıl tanıdığı yapılandırılır. Çoklu mesaj bitiş koşullarını yapılandırabilirsiniz. Yapılandırılmış koşullardan herhangi biri oluşursa, mesaj biter.

Örneğin, 300 milisaniyelik mesaj sonu aşımı, 40 bit süreli ara karakter zaman aşımı ve maksimum 50 bayt uzunluk ile bir bitiş koşulu belirtebilirsiniz. Mesaj alımı 300 milisaniyeden daha uzun sürerse ya da herhangi iki karakter arasındaki boşluk 40 bit süresini aşarsa ya da 50 bayt alınırsa mesaj biter.

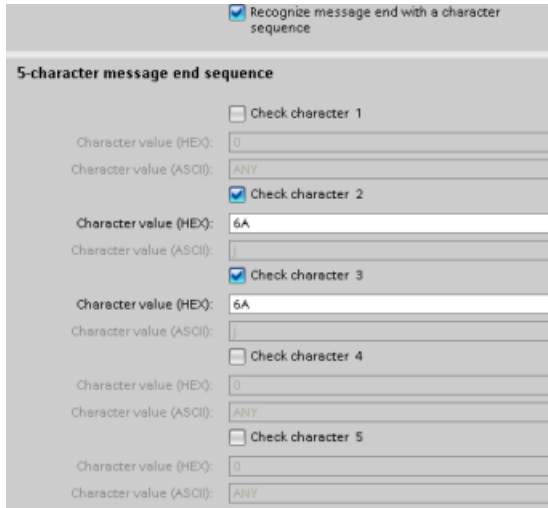
Parametre	Tanımlama
Mesaj zaman aşımıyla mesaj sonunu tanı	<p>Mesajı sonunu beklemek için yapılandırılmış zaman miktarı dolduğunda, mesaj sonu oluşur. Bir başlatma koşulu karşılandığında mesaj zaman aşımı periyodu başlar. Varsayılan değer 200 ms ve aralık 0 ile 65535 ms.</p>  <p>① Alınan karakterler          ② Başlangıç Mesaj koşulu yerine getirildi: Mesaj zamanlayıcısı başlar          ③ Mesaj zamanlayıcı doldu ve mesajı sonlandırır</p>
Tepki zaman aşımıyla mesaj sonunu tanı	<p>Geçerli bir başlangıç dizisi alınmadan önce bir yanıt için beklenecek yapılandırılmış zaman miktarı dolduğunda mesaj sonu oluşur. Bir iletim bittiğinde yanıt zaman aşımı süresi başlar ve CM veya CB alma işlemini başlatır. Varsayılan yanıt zaman aşımı 200 ms ve aralık 0 ile 65535 ms'dir. Bir karakter yanıt periyodu içinde, (RCVTIME) alınmasa, o zaman karşılık gelen RCV_PTP komutu için bir hata döndürülür. Yanıt zaman aşımı, belirli bir bitiş koşulu tanımlamaz. Bu sadece, bir karakterin belirtilen süre içinde başarıyla alınmış olması gerektiğini belirtir. Bir mesajın gerçek sonunu belirtmek için başka bir bitiş koşulu yapılandırmanız gerekir.</p>  <p>① İletilen karakterler          ② Alınan karakterler          ③ İlk karakter bu sefer başarılı şekilde alınmalı</p>
Karakterler arası boşlukla mesaj sonunu tanı	<p>Bir mesajın herhangi iki ardışık karakter arasındaki maksimum yapılandırılan zaman aşımı süresi dolduğu zaman mesaj sonu oluşur. Karakterler arası boşluk için varsayılan değer 12 bit süresi ve maksimum sayı 65535 bit süresi, en fazla sekiz saniyeye kadar.</p>  <p>① Alınan karakterler          ② Karakterler arası zamanlayıcıyı yeniden başlatır          ③ Karakterler arası zamanlayıcı doldu ve mesajı sonlandırdı.</p>



Parametre	Tanımlama
Sabit sayıda bir karakter alarak mesaj sonunu tanı	Belirtilen sayıda karakter alındığında mesaj sonu oluşur. Sabit uzunluk için geçerli aralık 1-4096'dır. S7-1200 için unutmayınız, bu mesaj sonu koşulu V4.0 CPU'lar veya üstü için geçerlidir.
Maksimum uzunluk ile mesaj sonunu tanı	Maksimum sayıda yapılandırılmış karakter alındığında mesaj sonu oluşur. Maksimum uzunluk için geçerli aralık 1-1023'tür. Bu koşul bir mesaj arabellek taşma hatasını önlemek için kullanılabilir. Bu mesaj sonu koşulu zaman aşımı sonu koşullarıyla birleştirildiğinde ve zaman aşımı koşulu oluştuğunda, herhangi bir geçerli alınan karakterler, maksimum uzunluğa ulaşılmış olsa bile sağlanmaktadır. Bu sayede yalnızca maksimum uzunluk bilindiği zaman değişen uzunluk protokolleri için desteğe izin verilir.
Mesajdan mesaj uzunluğunu oku	Mesajın kendisi mesajın uzunluğunu belirtir. Belirtilen uzunlukta bir mesaj alındığında mesaj sonu oluşur. Mesajın uzunluğunu belirtme ve yorumlama yöntemi aşağıda tarif edilmiştir.
Bir karakter ile mesaj sonunu tanı	Belirtilen karakter alındığında mesajı sonu oluşur.
Bir karakter dizisi ile mesaj sonunu tanı	Belirtilen karakter dizisi alındığında mesaj sonu oluşur. En fazla beş karakter dizisi belirtebilirsiniz. Her karakter konumu için, belirli bir onaltılık karakter veya karakterin dizi eşleşmesinde göz ardı edildiğini belirtiniz. Karakterleri göz ardı edilen öndeki karakterler mesaj sonu koşulunun bir parçası değildir. Karakterleri göz ardı edilen izleyen karakterler mesaj sonu koşulunun bir parçasıdır.

### Örnek Yapılandırma - bir karakter dizisi ile son mesaj

Aşağıdaki son mesaj koşulu yapılandırmasını göz önüne alınız:



Bu durumda ardından herhangi iki karakter gelen iki ardışık 0x7A karakter alındığında, son koşulu karşılanır. 0x7A 0x7A yapısından önce gelen karakter, son karakter dizisinin parçası değildir. aşağıdaki 0x7A 0x7A yapısından sonra gelen iki karakter son karakter dizisini sonlandırmak için gereklidir. 4 ve 5 karakter pozisyonlarda alınan değerler alakasızdır, ancak son koşulunu karşılamak için alınması gerekir.

**Mesajın içinde mesaj uzunluğu özelliği**

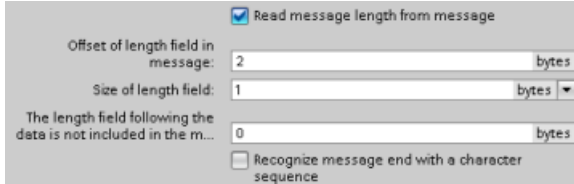
Mesaj uzunluğu, mesajın içinde bulunan bir özel durum seçtiğinizde, mesaj uzunluğu hakkında bilgi tanımlayan üç parametre sağlamanız gerekir.

Gerçek Mesaj yapısı kullanımdaki protokole göre değişir. Aşağıdaki gibi üç parametre şunlardır:

- n: uzunluk belirticini başlatan mesajın içindeki karakter konumu ( 1 tabanlı)
- Uzunluk ölçüsü: uzunluk belirtici bayt sayısı ( bir, iki veya dört )
- Uzunluk m: uzunluk sayımına dahil edilmeyen uzunluk belirticinin ardından gelen karakter sayısı

Biten karakterlerin bitişik olması gerekmez. "Uzunluk m" değeri, boyutu uzunluk alanına dahil olmayan bir sağlama alanının uzunluğunu belirtmek için kullanılabilir.

Bu alanlar cihaz özelliklerinin alınan mesaj yapılandırmasında görünür:



The screenshot shows a configuration window with the following settings:

- Read message length from message
- Offset of length field in message: 2 bytes
- Size of length field: 1 bytes
- The length field following the data is not included in the m...: 0 bytes
- Recognize message end with a character sequence

**Örnek 1:** Aşağıdaki protokole göre yapılandırılmış bir mesajı göz önünde bulundurun:

STX	Len (n)	Uzunluk ile sayılan karakterler 3 ila 14											
		ADR	PKE		INDEX		PWD		STW		HSW		BCC
1	2	3	4	5	6	7	8	9	10	11	12	13	14
STX	0x0C	xx	xxxx		xxxx		xxxx		xxxx		xxxx		xx

Aşağıdaki gibi, bu mesajın için aldığınız mesaj uzunluğu parametrelerini yapılandırınız:

- $n = 2$  (mesaj uzunluğu Bayt 2 ile başlar.)
- Uzunluk ölçüsü = 1 (mesaj uzunluğu bir bayt olarak tanımlanır.)
- Uzunluk  $m = 0$  (uzunluk sayımında sayılmayan uzunluk belirticinin ardından ek karakterler yoktur. Uzunluk belirticiyi oniki karakter izler.)

Bu örnekte, 3 ila 14 (dahil) arasındaki karakter Len (n) ile sayılan karakterlerdir.

**Örnek 2:** Aşağıdaki protokole göre yapılandırılmış bir mesaj göz önünde bulundurun:

SD1	Len (n)	Len (n)	SD2	Uzunluk ile sayılan karakterler 5 ila 10						FCS	ED
				DA	SA	FA	Veri birimi=3 bytes				
1	2	3	4	5	6	7	8	9	10	11	12
xx	0x06	0x06	xx	xx	xx	xx	xx	xx	xx	xx	xx

Aşağıdaki gibi, bu mesajın için aldığınız mesaj uzunluğu parametrelerini yapılandırınız:

- $n = 3$  (mesaj uzunluğu Bayt 3 ile başlar.)
- Uzunluk ölçüsü = 1 (mesaj uzunluğu bir bayt olarak tanımlanır.)
- Uzunluk  $m = 3$  (uzunluk sayımında sayılmayan uzunluk belirticinin ardından gelen üç karakter vardır. Bu örneğe ait protokolde, karakterler SD2, FCS ve ED uzunluk sayımında sayılmaz. Diğer altı karakter uzunluk sayımında sayılır; Bu nedenle uzunluk belirticinin ardından gelen toplam karakter sayısı dokuz.)

Bu örnekte, 5 ila 10 (dahil) arasındaki karakter Len (n) ile sayılan karakterlerdir.

### 12.3.3 Noktadan noktaya komutlar

#### 12.3.3.1 Noktadan noktaya komutlar için ortak parametreler

Çizelge 12- 3 PTP komutlar için ortak giriş parametreleri

Parametre	Açıklama
REQ	PtP komutlarının çoğu düşük-yüksek geçiş üzerinde çalışmasını başlatmak için REQ girişini kullanır. REQ girişi, komutun bir yürütmesi için (TRUE) yüksek olmalıdır, ama REQ girişi istediğiniz sürece TRUE kalabilir. REQ girişinin geçmiş durumunu sıfırlayabilecek şekilde REQ girişi FALSE ile çağrılana kadar Komut başka bir işlem başlatmaz. Komut, sonraki işlemi başlatmak için düşük-yüksek geçişi tespit edebilecek şekilde gereklidir. Programda bir PtP komutu yerleştirdiğinizde, STEP 7, kopya DB tanımlamanızı ister. Her PtP komut çağrı için benzersiz bir DB kullanınız. Bu, her bir komutun REQ gibi girişleri düzgün bir şekilde işlenmesini sağlar.
PORT	Port adresi. haberleşme cihazı yapılandırılması sırasında atanır. Yapılandırmadan sonra. varsayılan sembolik port adı. parametre yardımcılı açılan listeden seçilebilir. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dir. Port sembolik ismi, PLC etiket çizelgesinin "Sabitler" sekmesinde atanır.
Bit süre çözünürlüğü	Pek çok parametre, yapılandırılmış baud hızında bit sürelerinin sayısı ile belirtilir. Bit sürelerinde parametre belirtmek parametrenin baud hızından bağımsız olmasını sağlar. Bit süre birimleriyle belirtilen tüm parametreler, maksimum 65535 e kadar belirtilebilir. Bununla birlikte, bir CM ya da CB'nin ölçebileceği maksimum zaman miktarı sekiz saniyedir.

PtP komutlarının DONE, NDR, ERROR ve STATUS çıkış parametreleri, PtP işlemleri için yürütmenin tamamlanma durumunu sağlar.

Çizelge 12- 4 DONE, NDR, ERROR ve STATUS çıkış parametreleri

Parametre	Veri tipi	Varsayılan	Açıklama
DONE	Bool	FALSE	Son isteğin hatasız tamamlandığını belirtmek amacıyla bir yürütme için TRUE ayarlayınız; aksi halde, FALSE.
NDR	Bool	FALSE	Son isteğin hatasız tamamlandığını ve yeni bir verinin alındığını belirtmek amacıyla bir yürütme için TRUE ayarlayınız; aksi halde, FALSE.
ERROR	Bool	FALSE	Son isteğin STATUS deki uyandırılabilir bir hata koduna sahip bir hata ile tamamlandığını belirtmek amacıyla bir yürütme için TRUE ayarlayınız; aksi halde, FALSE.
STATUS	Word	0	Sonuç durum: <ul style="list-style-type: none"> <li>DONE veya NDR biti varsa, o zaman STATUS 0'a veya bir bilgi koduna ayarlanır.</li> <li>ERROR biti varsa o zaman STATUS bir hata koduna ayarlanır.</li> <li>Yukarıdaki bitlerin hiçbiri yoksa komut, işlevin mevcut durumunu açıklayan durum sonuçları döndürür.</li> </ul> STATUS, fonksiyonun yürütülmesi süresince değerini korur.

**Not**

DONE, NDR ve FRROR parametreleri tek bir yürütme için avarlardır. Sizin program mantığınız veri mandallarında decici çıkış durum değerlerini kavdetmesi gerekir, böylece ardışık program taramalarında durum değişikliklerini algılayabilir.

Çizelge 12- 5 Ortak durum kodları

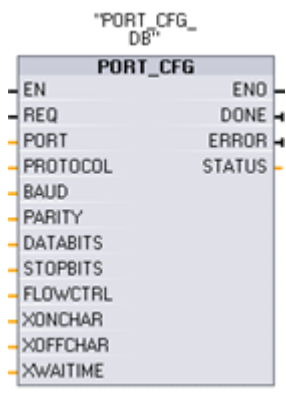
STATUS (W#16#....)	Açıklama
0000	Hata yok
7000	Fonksiyon meşgul değil
7001	Fonksiyon ilk çağrı ile meşgul.
7002	Fonksiyon ardışık çağrılar ile meşgul (ilk çağrıdan sonra yoklar).
8x3A	Parametre x' de kuraldışı işaretçi
8070	Tüm dahili kopya bellek kullanımında, çok sayıda eşzamanlı komut çalışıyor
8080	Port numarası kuraldışı.
8081	Zaman aşımı, modül hatası veya diğer dahili hatalar
8082	Parametrezasyon arka planda devam, çünkü parametrezasyon başarısız oldu.
8083	Arabellek taşması: CM veya CB, izin verilen uzunluk parametrelerinden daha büyük bir uzunluğu olan bir alınan bir mesajı döndürdü.
8090	Dahili hata: Yanlış mesaj uzunluğu, yanlış alt modül veya kural dışı mesaj
8091	Dahili hata: parametrezasyon mesajında yanlış sürüm
8092	Dahili hata: parametrezasyon mesajında yanlış kayıt uzunluğu

Çizelge 12- 6 Ortak hata sınıfları

Sınıf Açıklama	Hata sınıfları	Açıklama
Port yapılandırma	80Ax	Ortak port yapılandırma hatalarını tanımlamak için kullanılır
Verici yapılandırma	80Bx	Ortak verici yapılandırma hatalarını tanımlamak için kullanılır
Alma yapılandırma	80Cx	Ortak Alma yapılandırma hatalarını tanımlamak için kullanılır
İletim çalışma zamanı	80Dx	Ortak İletim çalışma zamanı hatalarını tanımlamak için kullanılır
Alma çalışma zamanı	80Ex	Ortak Alma çalışma zamanı hatalarını tanımlamak için kullanılır
Sinyal işleme	80Fx	Tüm sinyal işlemlerle ilişkili ortak hataları tanımlamak için kullanılır

### 12.3.3.2 PORT\_CFG (haberleşme parametrelerini dinamik olarak yapılandırınız) komutu

Çizelge 12- 7 PORT\_CFG (Port Yapılandırma) komutu

LAD / FBD	SCL	Açıklama
	<pre>"PORT_CFG_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,   PROTOCOL:=_uint_in_,   BAUD:=_uint_in_,   PARITY:=_uint_in_,   DATABITS:=_uint_in_,   STOPBITS:=_uint_in_,   FLOWCTRL:=_uint_in_,   XONCHAR:=_char_in_,   XOFFCHAR:=_char_in_,   WAITTIME:=_uint_in_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_);</pre>	<p>PORT_CFG programınızdan baud hızı gibi port parametrelerini değiştirmenize izin verir.</p> <p>Cihaz yapılandırma özelliklerinde portun ilk statik yapılandırmasını kurabilirsiniz, ya da sadece varsayılan değerleri kullanabilirsiniz. Yapılandırmayı değiştirmek için programınızda PORT_CFG komutunu yürütebilirsiniz.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

PORT\_CFG yapılandırma değişiklikleri CPU da kalıcı olarak saklanmaz. CPU, RUN-STOP moduna geçtiğinde ve güç çevrimi sonrasında Cihaz Yapılandırmasında yapılandırılan parametreler geri yüklenir. Daha fazla bilgi için haberleşme portlarını yapılandırma (Sayfa 672) ve akış kontrolünü Yönetme (Sayfa 674) bölümüne bakınız.

Çizelge 12- 8 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
REQ	IN	Bool Bu girişin yükselen kenarında yapılandırma değişikliğini etkinleştiriniz. (Varsayılan değer: False)
PORT	IN	PORT CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dır. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır. (Varsayılan değer: 0)
PROTOCOL	IN	UInt 0 - Noktadan Noktaya haberleşme protokolü (Varsayılan değer) 1 .. n - özel protokoller için gelecek tanımlı
BAUD	IN	UInt Port baud hızı (Varsayılan değer: 0): 1 = 300 baud, 2 = 600 baud, 3 = 1200 baud, 4 = 2400 baud, 5 = 4800 baud, 6 = 9600 baud, 7 = 19200 baud, 8 = 38400 baud, 9 = 57600 baud, 10 = 76800 baud, 11 = 115200 baud
PARITY	IN	UInt Port paritesi (Varsayılan değer: 0): 1 = Parite yok, 2 = Çift Parite, 3 = Tek Parite, 4 = İşaret Paritesi, 5 = Boşluk Paritesi
DATABITS	IN	UInt Karakter başına bitler (Varsayılan değer.): 1 = 8 veri bitleri, 2 = 7 veri bitleri
STOPBITS	IN	UInt Dur bitleri (Varsayılan değer: 0): 1 = 1 dur biti, 2 = 2 Dur bitleri

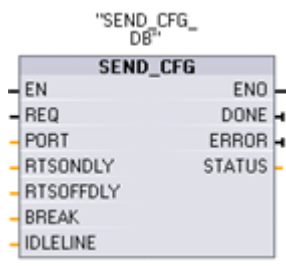
Parametre ve tipi		Veri tipi	Açıklama
FLOWCTRL	IN	UInt	Akış kontrol (Varsayılan değer: 0): 1 = Akış kontrolü yok, 2 = XON/XOFF, 3 = Donanım RTS her zaman açık, 4 = Donanım RTS anahtarlamalı
XONCHAR	IN	Char	XON karakteri olarak kullanılan karakteri belirtir. Bu genellikle bir DC1 karakter (11H)'dir. Bu parametre yalnızca akış kontrolü etkinse değerlendirilir. (Varsayılan değer: 0)
XOFFCHAR	IN	Char	XOFF karakteri olarak kullanılan karakteri belirtir. Bu genellikle bir DC3 karakter (13H)'dir. Bu parametre yalnızca akış kontrolü etkinse değerlendirilir. (Varsayılan değer: 0)
XWAITIME	IN	UInt	Bir XOFF karakteri aldıktan sonra, bir XON karakteri için ne kadar bekleyeceğini ya da RTS'yi (0 ile 65535 ms) etkinleştirdikten sonra CTS sinyalini ne kadar uzun bekleyeceğini belirtir. Bu parametre yalnızca akış kontrolü etkinse değerlendirilir. (Varsayılan değer: 2000)
DONE	OUT	Bool	Son istek hatasız tamamlandıktan sonra bir yürütme için TRUE
ERROR	OUT	Bool	Son istek hata ile tamamlandıktan sonra bir yürütme için TRUE
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)

Çizelge 12- 9 Durum kodları

STATUS (W#16#....)	Açıklama
80A0	Özel protokol yok.
80A1	Özel baud hızı yok.
80A2	Özel parite seçeneği yok.
80A3	Belirli sayıda veri bitleri yok.
80A4	Belirli sayıda stop bitleri yok.
80A5	Belirli tipte akış kontrolü yok.
80A6	Bekleme süresi 0 ve akış kontrolü etkinleştirildi
80A7	XON ve XOFF kuraldışı değerler (Örneğin, aynı değer)

### 12.3.3.3 SEND\_CFG (Seri iletim parametrelerini dinamik olarak yapılandırınız) komutu

Çizelge 12- 10 SEND\_CFG (Yapılandırmayı gönder) komutu

LAD / FBD	SCL	Açıklama
	<pre>"SEND_CFG_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,   RTSONDLY:=_uint_in_,   RTSOFFDLY:=_uint_in_,   BREAK:=_uint_in_,   IDLELINE:=_uint_in_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS =&gt;_word_out_ );</pre>	<p>SEND_CFG, bir PtP haberleşme portu için seri iletim parametrelerinin dinamik yapılandırılmasını sağlar. SEND_CFG çalıştırıldığında herhangi bir CM veya CB içinde mesaj sıraya atılır.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Cihaz yapılandırma özelliklerinde portun ilk statik yapılandırmasını kurabilirsiniz, ya da sadece varsayılan değerleri kullanabilirsiniz. Yapılandırmayı değiştirmek için programınızda SEND\_CFG komutunu yürütebilirsiniz.

SEND\_CFG yapılandırma değişiklikleri CPU da kalıcı olarak saklanmaz. CPU, RUN-STOP moduna geçtiğinde ve güç çevrimi sonrasında Cihaz Yapılandırmasında yapılandırılan parametreler geri yüklenir. Daha fazla bilgi için verici (gönderme) parametrelerini yapılandırma bölümüne bakınız (Sayfa 675).

Çizelge 12- 11 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Bu girişin yükselen kenarında yapılandırma değişikliğini etkinleştiriniz. (Varsayılan değer: False)
PORT	IN	PORT	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dir. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır. (Varsayılan değer: 0)
RTSONDLY	IN	UInt	Herhangi Tx veri iletimi gerçekleşmeden önce RTS'vi etkinleştirdikten sonra beklenecek milisanive sayısı. Bu parametre yalnızca donanım akış kontrolü etkinleştirildiğinde geçerlidir. Geçerli aralık 0-65535 ms'dir. 0 değeri, özelliği devre dışı bırakır. (Varsayılan değer: 0)
RTSOFFDLY	IN	UInt	Herhangi Tx veri iletimi gerçekleştikten sonra RTS'vi etkinleştirmeden önce beklenecek milisanive sayısı. Bu parametre yalnızca donanım akış kontrolü etkinleştirildiğinde geçerlidir. Geçerli aralık 0-65535 ms'dir. 0 değeri, özelliği devre dışı bırakır. (Varsayılan değer: 0)
BREAK	IN	UInt	Bu parametre, belirtilen sayıda bit süresi için her mesajın başlangıcında bir kesmenin gönderileceğini belirtir. Maksimum 65535 bit süresi, maksimum sekiz saniyeye kadar. 0 değeri özelliği devre dışı bırakır. (Varsayılan değer: 12)
IDLELINE	IN	UInt	Bu parametre, belirtilen sayıda bit süresi için her mesajın başlangıcında bir satırın boş kalacağını belirtir. Maksimum 65535 bit süresi, maksimum sekiz saniyeye kadar. 0 değeri özelliği devre dışı bırakır. (Varsayılan değer: 12)
DONE	OUT	Bool	Son istek hatasız tamamlandıktan sonra bir yürütme için TRUE



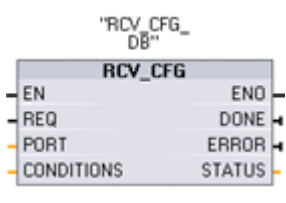
Parametre ve tipi	Veri tipi	Açıklama
ERROR	OUT	Bool
STATUS	OUT	Word

Çizelge 12- 12 Durum kodları

STATUS (W#16#....)	Açıklama
80B0	İletim kesme yapılandırmasına izin verilmez.
80B1	Kesme süresi, izin verilen maksimum değerden daha büyüktür.
80B2	Boş süre izin verilen maksimum değerden daha büyüktür.

### 12.3.3.4 RCV\_CFG (Seri alma parametrelerini dinamik olarak yapılandırınız) komutu

Çizelge 12- 13 RCV\_CFG (yapılandırmayı al) komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a function block named 'RCV_CFG' with inputs: EN, REQ, PORT, and CONDITIONS. It has outputs: END, DONE, ERROR, and STATUS.</p>	<pre>"RCV_CFG_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,   CONDITIONS:=_struct_in_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS =&gt; word_out );</pre>	<p>RCV_CFG, bir PtP haberleşme portu için seri alma parametrelerinin dinamik yapılandırılmasını sağlar. Bu komut alınan bir mesajın başlangıç ve bitiş sinyali koşullarını yapılandırır. RCV_CFG çalıştırıldığında bir CM veya CB içinde sıraya atılan mesajlar göz ardı edilir.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Cihaz yapılandırma özelliklerinde haberleşme portunun ilk statik yapılandırmasını kurabilirsiniz, ya da sadece varsayılan değerleri kullanabilirsiniz. Yapılandırmayı değiştirmek için programınızda RCV\_CFG komutunu yürütebilirsiniz.

RCV\_CFG yapılandırma değişiklikleri CPU da kalıcı olarak saklanmaz. CPU, RUN-STOP moduna geçtiğinde ve güç çevrimi sonrasında Cihaz Yapılandırmasında yapılandırılan parametreler geri yüklenir. Daha fazla bilgi için alma (gönderme) parametrelerini yapılandırma bölümüne bakınız (Sayfa 675).

Çizelge 12- 14 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Bu girişin yükselen kenarında yapılandırma değişikliğini etkinleştiriniz. (Varsayılan değer: False)
PORT	IN	PORT	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcılı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dir. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır. (Varsayılan değer: 0)
CONDITIONS	IN	CONDITIONS	Koşullar veri yapısı başlangıç ve bitiş mesaj koşullarını aşağıda açıklandığı gibi belirler.
DONE	OUT	Bool	Bir tarama için TRUE, son istek hatasız tamamlandıktan sonra
ERROR	OUT	Bool	Bir tarama için TRUE, son istek bir hata ile tamamlandıktan sonra
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)

### RCV\_PTP komutu için başlangıç durumları

RCV\_PTP komutu noktadan-noktaya haberleşme mesajlarının başlangıcını ve sonunu belirlemek için RCV\_CFG Komutu ile belirtilen yapılandırmayı kullanır. Bir mesajın başlangıcı, başlangıç koşullarına göre belirlenir. Bir mesajın başlangıcı, bir adet ya da başlangıç koşullarının bir kombinasyonu ile tespit edilebilir. Birden fazla başlatma koşulu belirtilmişse, mesaj başlamadan önce, tüm koşulların yerine getirilmesi gerekir.

Mesajının başlatma koşullarının açıklaması için." alma parametrelerini yapılandırma(Sayfa 676) " konusuna bakınız.

### CONDITIONS parametreleri veri tipi yapısı bölüm 1 (başlangıç koşulları)

Çizelge 12- 15 START koşulları için CONDITIONS yapısı

Parametre ve tipi	Veri tipi	Açıklama	
STARTCOND	IN	UInt	Başlangıç durumunu belirtir (Varsayılan değer: 1) <ul style="list-style-type: none"> <li>• 01H – Başlangıç karakteri</li> <li>• 02H – Herhangi bir karakter</li> <li>• 04H – Satır kesmesi</li> <li>• 08H – Boş satır</li> <li>• 10H - Dizi 1</li> <li>• 20H - Dizi 2</li> <li>• 40H - Dizi 3</li> <li>• 80H - Dizi 4</li> </ul>
IDLETIME	IN	UInt	Boş satır zaman aşımı için gerekli bit sayısıdır. (Varsayılan değer: 40). Sadece bir boş satır koşulu ile kullanılabilir. 0 ila 65535
STARTCHAR	IN	Byte	Başlangıç karakteri koşulu ile kullanılabilir başlangıç karakteri. (Varsayılan değer: B#16#2)

Parametre ve tipi		Veri tipi	Açıklama
SEQ[1].CTL	IN	Byte	Her karakter için Dizi 1 göz ardı et/ karşılaştır kontrolü: (Varsayılan değer: B # 16 # 0) Bu, başlangıç dizisinde her karakter için etkinleştirme bitleridir <ul style="list-style-type: none"> <li>• 01H - Karakter 1</li> <li>• 02H - Karakter 2</li> <li>• 04H - Karakter 3</li> <li>• 08H - Karakter 4</li> <li>• 10H - Karakter 5</li> </ul> Bir karakteri ile ilişkili bitin devre dışı bırakılması, herhangi bir karakterin bu dizi pozisyonunda, eşleşeceği anlamına gelir.
SEQ[1].STR	IN	Char[5]	Dizi 1 başlangıç karakterleri (5 karakter). Varsayılan değer: 0
SEQ[2].CTL	IN	Byte	Her karakter için Dizi 2 göz ardı et/ karşılaştır kontrolü: (Varsayılan değer: B # 16 # 0)
SEQ[2].STR	IN	Char[5]	Dizi 2 başlangıç karakterleri (5 karakter). Varsayılan değer: 0
SEQ[3].CTL	IN	Byte	Her karakter için Dizi 3 göz ardı et/ karşılaştır kontrolü: (Varsayılan değer: B#16#0)
SEQ[3].STR	IN	Char[5]	Dizi 3 başlangıç karakterleri (5 karakter). Varsayılan değer: 0
SEQ[4].CTL	IN	Byte	Her karakter için Dizi 4 göz ardı et/ karşılaştır kontrolü: (Varsayılan değer: B#16#0)
SEQ[4].STR	IN	Char[5]	Dizi 4 başlangıç karakterleri (5 karakter). Varsayılan değer: 0

## Örnek

Alınan onaltılık kodlu mesajı "68 10 aa bb 68 10 aa 16" ve aşağıdaki çizelgede gösterilen yapılandırılmış başlangıç dizileri düşününüz. Başlangıç dizileri ilk 68H karakteri başarıyla alındığında değerlendirilmeye başlar. Dördüncü karakter (ikinci 68H) başarıyla alındıktan sonra, başlangıç koşulu 1 karşılanır. Başlangıç koşullarını karşılandığında, bitiş koşullarının değerlendirilmesi başlar.

Başlangıç dizisi işlemesi, çeşitli parite, çerçeveleme, ya da karakterler arası zamanlama hataları nedeniyle sona erdirilebilir. Başlangıç koşulu karşılanmadığından dolayı, bu hatalar, mesajın alınmamasına neden olur.

Çizelge 12- 16 Start durumu

Başlangıç durum	İlk Karakter	İlk Karakter +1	İlk Karakter +2	İlk Karakter +3	İlk Karakter +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

### RCV\_PTP komutu için bitiş durumları

Bir mesajın sonu, bitiş koşullarının özelliği ile belirlenir. Bir mesajın sonu, bir ya da daha çok yapılandırılmış bitiş koşullarının ilk oluşumu ile belirlenmektedir. "alma parametrelerini yapılandırma" (Sayfa 676) konusunda "Mesaj bitiş koşulları" bölümü, RCV\_CFG komutunda yapılandırabildiğiniz bitiş koşullarını açıklar.

Cihaz yapılandırmasında haberleşme ara yüzü özelliklerinde ya da RCV\_CFG komutundan gelenlerle bitiş koşullarını yapılandırabilirsiniz. CPU'nun STOP-RUN geçişlerinde, alma parametreleri (başlangıç ve bitiş koşullarının her ikisi) cihaz yapılandırma ayarlarına geri döner. STEP 7 kullanıcı programı RCV\_CFG yürütürse, ayarlar RCV\_CFG koşullarına değiştirilir.

### CONDITIONS Parametreleri veri tipi yapısı bölüm 2 (bitiş durumları)

Çizelge 12- 17 END durumları için CONDITIONS yapısı

Parametre	Parametre tipi	Veri tipi	Açıklama
ENDCOND	IN	UInt 0	Bu parametre mesaj bitiş durumunu belirtir: <ul style="list-style-type: none"> <li>• 01H – Tepki süresi</li> <li>• 02H – Mesaj süresi</li> <li>• 04H – Karakterler arası boşluk</li> <li>• 08H - Maksimum uzunluk</li> <li>• 10H - N + LEN + M</li> <li>• 20H - Dizi</li> </ul>
MAXLEN	IN	UInt 1	Maksimum mesaj uzunluğu: Yalnızca Maksimum uzunluk bitiş koşulu seçildiğinde kullanılır. 1 ila 1024 bayt
N	IN	UInt 0	Uzunluk alan mesajının içindeki bayt konumu. Sadece N + LEN + M bitiş koşulu ile kullanılır. 1 ila 1022 bayt
UZUNLUKSIZE	IN	UInt 0	Uzunluk alanının boyutu (1, 2, veya 4 bayt). Sadece N + LEN + M bitiş koşulu ile kullanılır.
UZUNLUKM	IN	UInt 0	Uzunluk alan değerine dahil edilmeyen uzunluk alanı ardından gelen karakter sayısını belirtir. Sadece N + LEN + M bitiş koşulu ile kullanılır. 0 ila 255 bayt
RCVTIME	IN	UInt 200	Alınacak ilk karakter için ne kadar bekleyeceğini belirtir. Bir karakter belirtilen süre içinde başarıyla alınmadıysa alma işlemi bir hata ile sona erdirilecek. Bu sadece tepki süresi koşulu ile kullanılır. (0-65535 bit süresi, maksimum 8 saniye)  Bir yanıtın ilk karakteri alındığında değerlendirme sona erdiği için bu parametre, bir mesaj bitiş koşulu değildir. Bir yanıt beklendiği zaman hiçbir yanıt alınmadığı için Sadece bir alıcı işleminin sonlandırılması anlamında bu bir bitiş koşuludur. Ayrı bir bitiş koşulu seçmelisiniz.

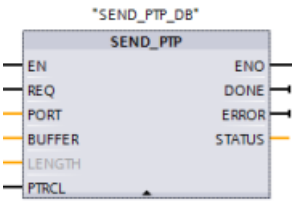
Parametre	Parametre tipi	Veri tipi	Açıklama
MSGTIME	IN	UInt 200	İlk karakter alındıktan sonra tamamen alınacak tüm mesaj için ne kadar bekleyeceğini belirtir. Bu parametre yalnızca Mesaj zaman aşımı koşulu seçildiğinde kullanılır. (0 ile 65535 milisaniye)
CHARGAP	IN	UInt 12	Karakterler arasındaki bit sayısını belirtiniz. Karakterler arasında bit süre sayısı belirtilen değeri aşarsa, o zaman bitiş koşulu karşılanacaktır. Bu sadece karakterler arası boşluk koşulu ile kullanılır. (0 ile 65535 bit süresi, maksimum 8 saniyeye kadar)
SEQ.CTL	IN	Byte B#16#0	Her karakter için Dizi 1 göz ardı et/ karşılaştır kontrolü: Bu bitiş dizisi için her karakter için etkinleştirme bitleri vardır. Karakter 1 bit 0, karakter 2 bit 1, ..., karakter 5 bit 4'tür. Bir karakter ile ilişkili bitin devre dışı bırakılması bu dizi pozisyonunda, herhangi bir karakterin eşleşeceği anlamına gelir.
SEQ.STR	IN	Char[5] 0	Dizi 1 Başlangıç karakterleri (5 karakter)

Çizelge 12- 18 Durum kodları

STATUS (W#16#...)	Açıklama
80C0	Kuraldışı başlangıç koşulu seçildi
80C1	Kuraldışı bitiş koşulu seçildi, hiç bir bitiş koşulu seçilmedi
80C2	Alma kesmesi etkinleştirildi ve bu mümkün değil.
80C3	Maksimum uzunluk bitiş koşulu etkinleştirildi ve maksimum uzunluk 0 veya > 1024.
80C4	Hesaplanan uzunluk etkinleştirildi ve N >= 1023.
80C5	Hesaplanan uzunluk etkinleştirildi ve uzunluk 1, 2 veya 4 değil.
80C6	Hesaplanan uzunluk etkinleştirildi ve M değeri > 255.
80C7	Hesaplanan uzunluk etkinleştirildi ve hesaplanan uzunluk > 1024.
80C8	Tepki zaman aşımı etkinleştirildi ve tepki zaman aşımı sıfır.
80C9	Karakterler arası boşluk zaman aşımı etkinleştirildi ve bu sıfır.
80CA	Boş satır zaman aşımı etkinleştirildi ve bu sıfır.
80CB	Bitiş dizisi etkinleştirildi ancak tüm karakterler "don't care".
80CC	Başlangıç dizisi (4 ten herhangi biri) etkinleştirildi ancak tüm karakterler "don't care".

### 12.3.3.5 SEND\_PTP (Gönderilen ara bellek verisini ilet) komutu

Çizelge 12- 19 SEND\_PTP (Noktadan noktaya veri gönder) komutu

LAD / FBD	SCL	Açıklama
	<pre>"SEND_PTP_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,    BUFFER:=_variant_in_,   LENGTH:=_uint_in_,   PTRCL:=_bool_in_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_);</pre>	<p>SEND_PTP veri iletimini başlatır ve haberleşme ara yüzü için atanmış arabelleği aktarır. CM veya CB atanan baud hızında veri gönderirken CPU programı devam eder. Sadece bir gönderme işlemi, belirli bir zamanda bekleyebilir. CM veya CB zaten bir mesaj iletirken, ikinci bir SEND_PTP yürütülürse CM veya CB bir hata döndürür.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Çizelge 12- 20 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	İletimi etkinleştirilmiş bu girişin yükselen kenarında istenen iletimi aktifleştirir. B, arabellek içeriğinin Noktadan Noktaya haberleşme ara yüzüne aktarımını başlatır. (Varsayılan değer: False)
PORT	IN	PORT	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcılı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dır. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır. (Varsayılan değer: 0)
BUFFER	IN	Variant	Bu parametre iletim arabelleğinin başlangıç konumunu işaret eder. (Varsayılan değer: 0) <b>Not:</b> Boolean veri veya Boolean dizileri desteklenmez.
LENGTH <sup>1</sup>	IN	UInt	Bayt cinsinden iletilen çerçeve uzunluğu (Varsayılan değer: 0) Karmaşık bir yapı iletirken, her zaman 0 uzunluğunu kullanınız.
PTRCL	IN	Bool	Bu parametre, eklenen CM veya CB içinde uygulanan normal noktadan noktaya veya belirli Siemens tarafından sağlanan protokoller olarak arabelleği seçer. (Varsayılan değer: False) FALSE = Kullanıcı programı noktadan-noktaya işlemleri kontrol eder. (sadece geçerli bir seçenek)
DONE	OUT	Bool	Bir tarama için TRUE, son istek hatasız tamamlandıktan sonra
ERROR	OUT	Bool	Bir tarama için TRUE, son istek bir hata ile tamamlandıktan sonra
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)

<sup>1</sup> Opsiyonel parametre: LAD / FBD Kutusunu genişletmek için bu kutunun altındaki oku tıklayınız ve bu parametreyi dahil ediniz.

Bir iletim işlemi yapılırken, DONE ve ERROR çıkışları FALSE olur. Bir iletim işlemi tamamlandığında, DONE veya ERROR çıkışı iletim işleminin durumunu göstermek için TRUE olarak ayarlanacaktır. DONE veya ERROR TRUE iken, STATUS çıkışı geçerlidir.

Haberleşme ara yüzü iletim verilerini kabul ederse komut, 16#7001 durumunu döndürür. CM veya CB hala iletlemeyle meşgul ise sonraki SEND\_PTP yürütümleri, 16 # 7002 döndürür. İletim işlemi tamamlandığında, CM veya CB 16 # 0000 (hata oluşmazsa) olarak iletim işleminin durumunu döndürür. Düşük REQ ile SEND\_PTP'nin sonraki yürütümleri 16#7000 (meşgul değil) durumunu döndürür.

Aşağıdaki diyagramlar REQ için çıkış değerlerinin ilişkisini gösterir. Burada, komutun iletim işleminin durumunu periyodik olarak kontrol etmek için çağrıldığı varsayılır. Aşağıdaki şemada, komutun (STATUS değerleri ile temsil edilen) her taramada çağrıldığı varsayılır.

REQ							
DONE							
ERROR							
STATUS	7000H	7001H	7002H	7002H	7002H	0000H	7000H

Aşağıdaki şemada REQ satırı iletim işlemini başlatmak için (bir tarama için) darbeli ise DONE ve STATUS parametrelerinin sadece bir tarama için geçerli olduğu gösterilmiştir.

REQ							
DONE							
ERROR							
STATUS	7000H	7001H	7002H	7002H	7002H	0000H	7000H

Bir hata olduğunda, aşağıdaki diyagram DONE, ERROR ve STATUS parametrelerinin ilişkisini gösterir.

REQ							
DONE							
ERROR							
STATUS	7000H	7001H	7002H	7002H	7002H	80D1H	7000H

DONE, ERROR ve STATUS değerleri yalnızca SEND\_PTP, aynı kopya DB ile tekrar yürütülene kadar geçerlidir.

Çizelge 12- 21 Durum kodları

STATUS (W#16#....)	Açıklama
80D0	Yeni istek, verici aktif iken
80D1	Bekleme süresi içinde hiçbir bir CTS olmadığı için iletim iptal edildi
80D2	DCE cihazından hiçbir DSR gelmediği için iletim iptal edildi
80D3	Kuyruk taşması nedeniyle iletim iptal edildi (1024'ten fazla bayt iletildi)
80D5	Ters besleme sinyali (Tel kopma durumu)
833A	BUFFER parametreleri için DB mevcut değil.

### LENGTH ve BUFFER parametrelerinin etkileşimi

SEND\_PTP komut ile iletilebilen verinin minimum boyutu bir bayttır. BUFFER parametreleri iletilecek veri boyutunu belirler. BUFFER parametreleri için Bool veri tipini veya Bool dizilerini kullanamazsınız.

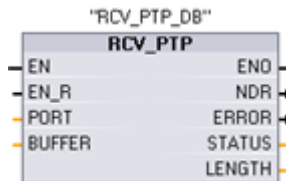
Her zaman LENGTH parametrelerini 0 olarak ayarlayınız ve SEND\_PTP'nin, BUFFER parametreleri tarafından temsil edilen tüm veri yapısını gönderdiğinden emin olunuz. BUFFER parametrelerinde bir veri yapısının sadece bir kısmını göndermek isterseniz aşağıdaki gibi LENGTH parametrelerini ayarlayabilirsiniz:

Çizelge 12- 22 LENGTH ve BUFFER parametreleri

UZUNLUK	BUFFER	Açıklama
= 0	Kullanılmıyor	BUFFER parametrelerinde tanımlanan komple veri gönderilir. LENGTH = 0 olduğunda aktarılan bayt sayısını belirtmek gerekmez.
> 0	Temel Veri tipi	LENGTH değeri bu veri tipi bayt sayısını içermelidir. Örneğin, bir Word değeri için, LENGTH iki olmalıdır. Dword veya Real için, LENGTH dört olmalıdır. Aksi takdirde, hiçbir şey aktarılmaz ve hata 8088H döndürülür.
	Yapı	LENGTH değeri, yapının birinci LENGTH baytlarının sadece BUFFER dan gönderildiği durumda, yapının tam bayt uzunluğundan daha az sayıda bayt içerebilir. Bir yapının iç bayt organizasyonu, her zaman belirlenemediğinden, beklenmeyen sonuçlar alabilirsiniz. Bu durumda, tam yapıyı göndermek için 0 LENGTH kullanınız.
	Dizi	LENGTH değeri, dizinin tam bayt uzunluğunda daha az olan ve veri ögesi bayt sayısının katı olması gereken bayt sayısı içermelidir. Örneğin, LENGTH parametreleri, Word dizisi için ikinin katı ve Real dizisi için dördün katı olmalıdır. LENGTH belirtildiğinde, LENGTH baytlarda bulunan dizi elemanlarının sayısı aktarılır. BUFFER, örneğin, 15 Dword (60 toplam bayt) lük bir dizi içerir ve 20'lik LENGTH belirtirseniz, o dizideki ilk beş Dword aktarılır.  LENGTH değeri, veri ögesi bayt sayısının katı olmalıdır. Aksi takdirde, STATUS = 8088H, ERROR = 1 olur ve hiçbir iletim olmaz.
	String	LENGTH parametreleri, iletilecek karakter sayısını içerir. Sadece String karakterler iletir. String in maksimum ve gerçek uzunluk baytı iletmez.

### 12.3.3.6 RCV\_PTP (Alma mesajlarını etkinleştirme) komutu

Çizelge 12- 23 RCV\_PTP (Noktadan noktaya alma) komutu

LAD / FBD	SCL	Açıklama
	<pre>"RCV_PTP_DB" (   EN_R:=_bool_in_,   PORT:=_uint_in_,   BUFFER:=_variant_in_,   NDR=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_,   LENGTH=&gt;_uint_out_);</pre>	<p>RCV_PTP, CM veya CB de alınan mesajları denetler. Bir mesaj varsa, o mesaj CM veya CB den CPU' ya transfer edilecektir. Bir hata, uygun STATUS değerini döndürür.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.



Çizelge 12- 24 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
EN_R	IN	Bool	Bu giriş TRUE ise ve bir mesaj mevcut olduğunda, mesaj BUFFER a CM veya CB den aktarılır. EN_R, FALSE olduğunda, CM veya CB mesajları kontrol edilir ve NDR, ERROR ve STATUS çıkışı güncellenir, ancak mesaj BUFFER a aktarılmaz. (Varsayılan değer: 0)
PORT	IN	PORT	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dır. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır. (Varsayılan değer: 0)
BUFFER	IN	Variante	Bu parametre alma arabelleğinin başlangıç konumunu işaret eder. Bu arabellek maksimum uzunlukta mesaj almak için yeterince büyük olmalıdır. Boolean veri veya Boolean diziler desteklenmez. (Varsayılan değer: 0).
NDR	OUT	Bool	Yeni veri hazır olduğunda ve işlem hatasız tamamlandığında bir yürütme için TRUE
ERROR	OUT	Bool	Bir yürütme için TRUE, İşlem hata ile tamamlandıktan sonra.
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)
LENGTH	OUT	UInt	Bayt cinsinden döndürülen mesajın uzunluğu (Varsayılan değer: 0)

NDR veya ERROR TRUE olduğunda STATUS değeri geçerlidir. STATUS değeri CM veya CB de alma işleminin sona erdirmeye nedenini sağlar. Bu genellikle, alım işleminin başarılı olduğunu ve alma sürecinin normal olarak sonlandırıldığını belirten pozitif bir değerdir. STATUS değeri (onaltılık değerlerin en önemli biti ayarlanır) negatif ise, alma işlemi; parite, çerçeveleme veya taşma hataları gibi bir hata koşulu için sonlandırılır.

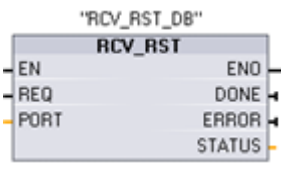
Her PtP haberleşme ara yüzü maksimum 1024 bayt a kadar boşaltılabilir. Bu, büyük bir mesaj ya da birkaç küçük mesaj olabilir. Birden fazla mesaj CM veya CB de mevcut ise, RCV\_PTP komutu mevcut eski mesajı döndürür. Bir sonraki RCV\_PTP komut yürütmesi, mevcut sonraki en eski mesajı döndürür.

Çizelge 12- 25 Durum kodları

STATUS (W#16#...)	Açıklama
0000	Mevcut arabellek yok
0094	Alınan maksimum karakter uzunluğu nedeniyle sonlandırılan mesaj
0095	Mesaj zaman aşımı nedeniyle sonlandırılan mesaj
0096	Karakterler arası zaman aşımı nedeniyle sonlandırılan mesaj
0097	Tepki zaman aşımı nedeniyle sonlandırılan mesaj
0098	Mesaj sonlandırıldı Çünkü "N+LEN+M" uzunluk durumu karşılandı
0099	Mesaj sonlandırıldı Çünkü bitiş dizisi karşılandı
80E0	Mesaj sonlandırıldı Çünkü alıcı arabellek dolu
80E1	Parite hatası nedeniyle mesaj sonlandırıldı
80E2	Çerçeve hatası nedeniyle mesaj sonlandırıldı
80E3	Taşma hatası nedeniyle mesaj sonlandırıldı
80E4	Mesaj sonlandırıldı. Çünkü hesaplanan uzunluk arabellek boyutunu aşıyor
80E5	Ters besleme sinyali (Kablo kopma durumu)
833A	BUFFER parametreleri için DB mevcut değil.

### 12.3.3.7 RCV\_RST (Alıcı arabelleği sil) komutu

Çizelge 12- 26 RCV\_RST (Alıcı sıfırlaması) komutu

LAD / FBD	SCL	Açıklama
	<pre>"RCV_RST_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_);</pre>	RCV_RST, CM veya CB deki alıcı arabellekleri temizler.

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Çizelge 12- 27 Parametreler için veri tipleri

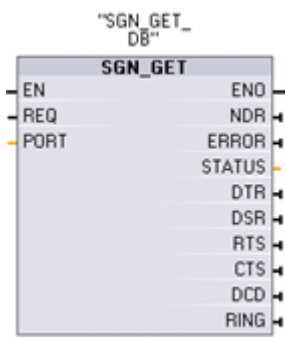
Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Bu etkinleştirme girişinin yükselen kenarında alıcı sıfırlamasını aktifleştirir (Varsayılan değer: False)
PORT	IN	PORT	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcılı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dır. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır. (Varsayılan değer: 0)
DONE	OUT	Bool	Bir tarama için TRUE olduğunda, son isteğin hatasız tamamlandığını gösterir.
ERROR	OUT	Bool	TRUE olduğunda, son isteğin hata ile tamamlandığını gösterir. Ayrıca, çıkış TRUE olduğunda, STATUS çıkışı ilgili hata kodlarını içerir.
STATUS	OUT	Word	Hata kodu (Varsayılan değer: 0) Haberleşme durum kodları için Noktadan Noktaya komutlara (Sayfa 664) ait ortak parametrelere bakınız.

#### Not

Haberleşme hatasının ardından mesaj arabelleklerinin temiz olduğundan emin olmak için ya da baud hızı gibi haberleşme parametrelerini değiştirdikten sonra RCV\_RST komutunu kullanmak isteyebilirsiniz. RCV\_RST Yürütülmesi, modülün, dahili mesaj arabelleklerinin tümünü temizlemesine neden olur. Mesaj arabellekleri temizlendikten sonra, program bir sonraki alma komutunu çalıştırdığında döndürülen mesajların yeni mesajlar olduğundan ve RCV\_RST çağırısından biraz zaman önce olan eski mesajlar olmadığından emin olabilirsiniz.

### 12.3.3.8 SGN\_GET (RS-232 sinyallerini sorgulama) komutu

Çizelge 12- 28 SGN\_GET (RS232 sinyallerini alma) komutu

LAD / FBD	SCL	Açıklama
	<pre> "SGN_GET_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,   NDR=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_,   DTR=&gt;_bool_out_,   DSR=&gt;_bool_out_,   RTS=&gt;_bool_out_,   CTS=&gt;_bool_out_,   DCD=&gt;_bool_out_,   RING=&gt;_bool_out_ ); </pre>	<p>SGN_GET RS232 haberleşme sinyallerinin mevcut durumlarını okur.</p> <p>Bu fonksiyon sadece RS232 CM için geçerlidir.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Çizelge 12- 29 Parametreler için veri tipleri


Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Bu girişin yükselen kenarında RS232 sinyal durum değerlerini alınız (Varsayılan değer: False)
PORT	IN	PORT	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcılı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dir. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır.
NDR	OUT	Bool	Bir tarama için TRUE, Yeni veri hazır olduğunda ve işlem hatası tamamlandığında
ERROR	OUT	Bool	Bir tarama için TRUE, İşlem hata ile tamamlandıktan sonra
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)
DTR	OUT	Bool	Veri terminali hazır, modül hazır (çıkış). Varsayılan değer: False
DSR	OUT	Bool	Veri seti hazır, Haberleşme partneri hazır (giriş). Varsayılan değer: False
RTS	OUT	Bool	Gönderme isteği, gönderilecek modül hazır (çıkış). Varsayılan değer: False
CTS	OUT	Bool	Gönderme için temizle, Haberleşme partneri veriyi alabilir (giriş). Varsayılan değer: False
DCD	OUT	Bool	Veri taşıyıcı tespiti, alıcı sinyal seviyesi (daima False, desteklenmiyor)
RING	OUT	Bool	Zil göstergesi, Gelen çağrı gösterimi (daima False, desteklenmiyor)

Çizelge 12- 30 Durum kodları

STATUS (W#16#...)	Açıklama
80F0	CM veya CB RS485 tir ve hiç bir sinyal mevcut değil

### 12.3.3.9 SGN\_SET (RS-232 sinyallerini ayarlama) komutu

Çizelge 12- 31 SGN\_SET (RS232 sinyallerini ayarlama) komutu

LAD / FBD	SCL	Açıklama
	<pre>"SGN_SET_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,   SIGNAL:=_byte_in_,   RTS:=_bool_in_,   DTR:=_bool_in_,   DSR:=_bool_in_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_);</pre>	<p>SGN_GET RS232 haberleşme sinyallerinin durumlarını ayarlar.</p> <p>Bu fonksiyon sadece RS232 CM için geçerlidir.</p>

<sup>1</sup> Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.

Çizelge 12- 32 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
REQ	IN	Bool	Bu girişin yükselen kenarında set RS232 sinyal çalışmasını başlatır (Varsayılan değer: False)
PORT	IN	PORT	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dır. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır. (Varsayılan değer: 0)
SIGNAL	IN	Byte	Ayarlanacak sinyali seçer: (çoklu izin verme). Varsayılan değer: 0 <ul style="list-style-type: none"> <li>• 01H = Set RTS</li> <li>• 02H = Set DTR</li> <li>• 04H = Set DSR</li> </ul>
RTS	IN	Bool	Gönderme isteği, ayarlanacak veriyi göndermek için modül hazır (true veya false), Varsayılan değer: False
DTR	IN	Bool	Veri terminali hazır, ayarlanacak veriyi göndermek için modül hazır (true veya false). Varsayılan değer: False
DSR	IN	Bool	Veri seti hazır (yalnızca DCE tip ara yüzlerine uygulanır), Kullanılmıyor.
DONE	OUT	Bool	Son istek hatasız tamamlandıktan sonra bir yürütme için TRUE
ERROR	OUT	Bool	Son istek hata ile tamamlandıktan sonra bir yürütme için TRUE
STATUS	OUT	Word	Yürütme durum kodu (Varsayılan değer: 0)

Çizelge 12- 33 Durum kodları

STATUS (W#16#....)	Açıklama
80F0	CM veya CB RS485 tir ve hiç bir sinyal ayarlanamıyor
80F1	Donanım akış kontrolü nedeniyle sinyaller ayarlanamıyor
80F2	DSR'yi ayarlayamıyor, çünkü modül DTE
80F3	DTR'yi ayarlayamıyor, çünkü modül DCE

### 12.3.4 PtP haberleşmelerini Programlama

STEP 7, kullanıcı programının, kullanıcı programında tasarlanmış ve uygulanan bir protokol ile Noktadan Noktaya haberleşme gerçekleştirmesini sağlamak için genişletilmiş komutlar sağlar. Bu komutlar iki kategoriye ayrılır:

- Yapılandırma komutları
- Haberleşme komutları

#### Yapılandırma komutları

Kullanıcı programınız PtP Haberleşmesiyle meşgul olmadan önce, haberleşme ara yüzü portunu ve veri gönderme ve veri alma parametrelerini yapılandırmanız gerekir.

Eğer kullanıcı programında cihaz yapılandırma veya bu komutlar yoluyla her CM veya CB için port yapılandırmasını ve mesaj yapılandırmasını gerçekleştirebilirsiniz:

- PORT\_CFG (Sayfa 686)
- SEND\_CFG (Sayfa 688)
- RCV\_CFG (Sayfa 689)

#### Haberleşme komutları

PtP haberleşme komutları, kullanıcı programının, haberleşme ara yüzlerinden mesajları almasına buraya göndermesine imkan verir. Bu komutlar ile veri aktarımı hakkında bilgi için veri tutarlılığı (Sayfa 178 ) hakkındaki bölüme bakınız.

Tüm PtP fonksiyonları, asenkron olarak çalışır. Kullanıcı programı iletim ve alımların durumunu belirlemek için bir yoklama mimarisini kullanabilir. SEND\_PTP ve RCV\_PTP eşzamanlı yürütülebilir. Haberleşme modülleri ve haberleşme kartı, maksimum 1024 baytlık arabellek boyutuna kadar gerekli ölçüde iletim ve alma mesajlarını arabellekte tutar.

CM'ler ve CB, gerçek noktadan noktaya cihazlardan gelen mesajları alır veya buraya mesajları gönderir. Belirli bir haberleşme portuna gönderilen veya buradan alınan mesaj protokolü, bir arabellektedir. Arabellek ve port, gönderme ve alma komutlarının parametreleridir:

- SEND\_PTP (Sayfa 694)
- RCV\_PTP (Sayfa 697)

Ek komutlar, alıcı arabelleği sıfırlama ve belirli RS232 sinyallerini alma ve ayarlama yeteneği sağlar:

- RCV\_RST (Sayfa 698)
- SGN\_GET (Sayfa 699)
- SGN\_SET (Sayfa 700)

#### 12.3.4.1 Yoklama mimarisi

STEP 7 kullanıcı programınızın, alınan mesajları kontrol etmek için döngüsel / periyodik olarak S7-1200 noktadan-noktaya komutlarını çağırması gerekir. Göndermeyi yoklama kullanıcı programınıza iletimin ne zaman tamamlandığını söyler.

### **Yoklama mimarisi: master**

Bir master için tipik dizi aşağıda verilmiştir:

10. Bir SEND\_PTP komutu CM veya CB için iletimi başlatır.
11. SEND\_PTP komutu, iletimin tamamlanma durumu yoklamak için sonraki taramalarda yürütülür.
12. SEND\_PTP komutu iletimin tamamlandığını gösterdiği zaman kullanıcı kodu yanıt almak için hazırlanabilir.
13. RCV\_PTP komutu yanıtı kontrol etmek için tekrar tekrar yürütülür. CM veya CB yanıt mesajını topladığı zaman, RCV\_PTP komutu yanıtı CPU'ya kopyalar ve yeni verinin alındığını gösterir.
14. Kullanıcı programı yanıtı işleyebilir.
15. 1. adımına gidin ve çevrimi tekrarlayınız.

### **Yoklama mimarisi: slave**

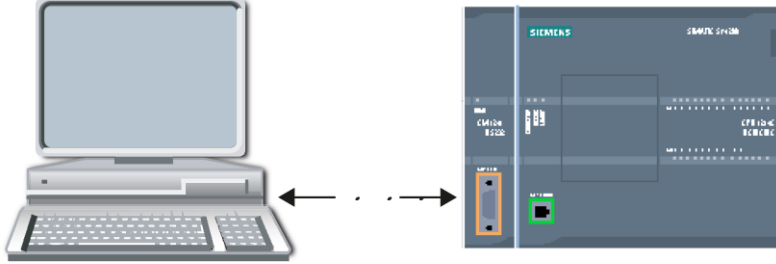
Bir slave için tipik dizi aşağıda verilmiştir:

16. Kullanıcı programı RCV\_PTP komutunu her taramada yürütür.
17. CM veya CB istek aldığı anda RCV\_PTP komutu yeni verinin hazır olduğunu ve isteğin CPU içine kopyalandığını gösterir.
18. Kullanıcı programı isteği servis eder ve bir yanıt oluşturur.
19. Master'a yanıt göndermek için bir SEND\_PTP komutunu kullanınız.
20. İletimin gerçekleştiğinden emin olmak için SEND\_PTP komutunu tekrar tekrar yürütün.
21. 1. adımına gidin ve çevrimi tekrarlayınız.

Slave yanıt beklerken master süreleri zaman aşımına uğramadan master'dan iletim almak için yeterli sıklıkta RCV\_PTP çağırma sorumludur. Bu görevi başarmak için çevrim süresinin aşım süresi sona ermeden master'dan iletimi almak için yeterli olduğu yerde kullanıcı programı bir döngüsel OB den RCV\_PTP'yi çağırabilir. Eğer master zaman aşımı süresi içinde iki yürütme için sağlanacak OB için çevrim süresini ayarlarsanız, kullanıcı programı herhangi bir kayıp olmaksızın iletimler alabilir.

### 12.3.5 Örnek: Noktadan Noktaya haberleşme

Bu örnekte, bir S7-1200 CPU, CM 1241 RS232 modülü aracılığıyla bir terminal emülatörüne sahip bir PC ile haberleşir. Noktadan noktaya yapılandırma ve STEP 7 programı bu örnekte CPU'nun PC'den nasıl bir mesaj aldığını ve geri PC'ye nasıl yansıttığını göstermektedir.



Normalde COM1 olan PC'nin RS232 ara yüzü ile CM 1241 RS232 modülünün haberleşme ara yüzünü bağlamanız gerekir. Bu portların her ikisi Veri Terminali Ekipmanı (DTE) olduğu için iki port bağlantı yapıldığı zaman, alma ve gönderme pinlerini (2 ve 3) değiştirmeniz gerekir: Bunu da aşağıdaki yöntemlerden biri ile yapabilirsiniz:

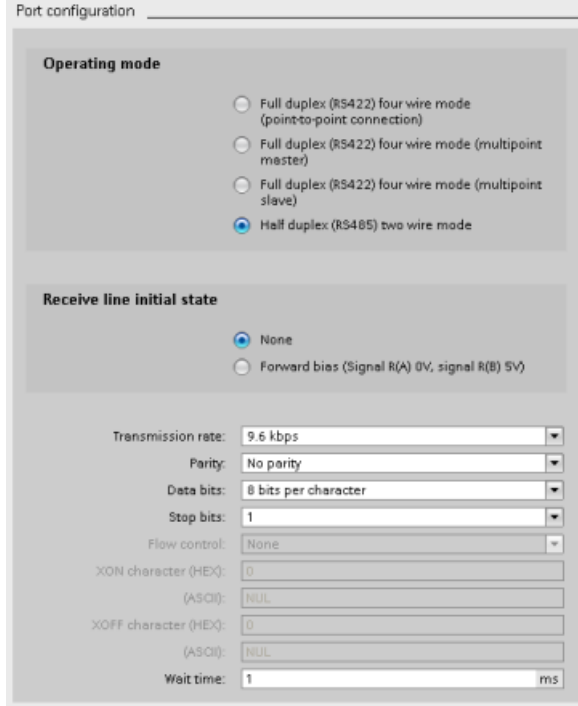
- Standart bir RS232 kablosu ile birlikte 2 ve 3 pinlerini değiştirmek için bir NULL modem adaptörü kullanınız.
- Zaten değiştirilmiş 2 ve 3 pinlerine sahip olan bir NULL modem kablosu kullanınız. Genellikle bir NULL modem kablosunu, iki adet dişi 9-pin D konektör uçlarına sahip olarak tespit edebilirsiniz.



### 12.3.5.1 Haberleşme modülünün yapılandırılması

STEP 7 deki Cihaz yapılandırmasından veya kullanıcı programı komutları ile CM 1241 i yapılandırabilirsiniz. Bu örnek Cihaz yapılandırma yöntemini kullanır.

- Port yapılandırma: Cihaz yapılandırmasından CM modülünün haberleşme portunu tıklayınız ve gösterildiği gibi portu yapılandırınız:



#### Not

"Çalışma modu" ve " hat başlangıç durumunu al" için yapılandırma ayarları yalnızca CM 1241 (RS422/RS485) modülü için geçerlidir. Diğer CM 1241 modüllerinin bu port yapılandırma ayarları yoktur. RS422 ve RS485 (Sayfa 707) yapılandırmaya bakınız.

- Mesajı yapılandırması İletimi: mesaj yapılandırması iletimi için varsayılanı kabul ediniz. Mesajı başlangıcında hiç bir kesme gönderilmeyecektir.

- Mesajı başlangıç yapılandırmasını alma: haberleşme hattı en az 50 bit süre için pasif kaldığında (9600 baud hızında yaklaşık 5 milisaniye =  $50 * 1/9600$ ) bir mesaj almaya başlamak için CM 1241 yapılandırınız:

> Message start

Start on any character  
 Start on special condition  
 Recognize message start with broken line  
 Recognize message start with idle line  
 Idle line time: 50 Bit times  
 Recognize message start with single character  
 Message start character (HEX): 2  
 Message start character (ASCII): STX  
 Recognize message start with a character sequence  
 Number of character sequences to define: 1

- Mesajı sonu yapılandırma alma: maksimum 100 bayt ya da bir satır besleme karakteri (10 ondalık veya onaltılık) aldığıında mesajı sona erdirmek için CM 1241 yapılandırınız. Bitiş dizisi, dizide en fazla beş bitiş karakterine izin verir. Dizinin beşinci karakteri satır besleme karakteridir. Ondan önceki dört bitiş dizisi karakterleri "don't care" veya seçilmemiş karakterlerdir. CM 1241, "don't care" karakterlerini değerlendirmez ancak mesaj sonunu belirtmek için daha fazla "don't care" karakteri ya da önünde sıfır olan bir satır besleme karakteri arar.

> Message end

Define message end conditions

Recognize message end by message timeout  
 Message timeout: 200 ms  
 Recognize message end by response timeout  
 Response timeout: 200 ms  
 Recognize message end by inter-character timeout  
 Inter-character gap timeout: 12 Bit times  
 Recognize message end by maximum length  
 Maximum length of message: 100 bytes  
 Read message length from message  
 Offset of length field in message: 1 bytes  
 Size of length field: 1 bytes  
 The length field following the data is not included in the m...  
 Recognize message end with a character sequence

5-character message end sequence

<input type="checkbox"/> Check character 1
Character value (HEX): 0
Character value (ASCII): ANY
<input type="checkbox"/> Check character 2
Character value (HEX): 0
Character value (ASCII): ANY
<input type="checkbox"/> Check character 3
Character value (HEX): 0
Character value (ASCII): ANY
<input type="checkbox"/> Check character 4
Character value (HEX): 0
Character value (ASCII): ANY
<input checked="" type="checkbox"/> Check character 5
Character value (HEX): A
Character value (ASCII): LF

### 12.3.5.2 RS422 ve RS485 Çalışma modları

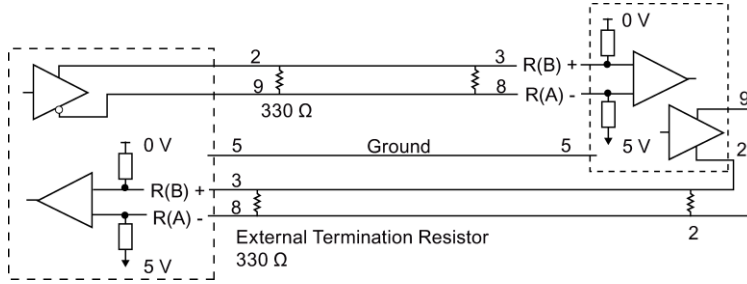
#### RS422 yapılandırma

RS422 modu için, ağ yapılandırmasına bağlı olarak üç çalışma modu vardır. Ağınızdaki cihazlara göre bu çalışma modundan birini seçiniz. Hat başlangıç durumunu alma için farklı seçimlerde, daha fazla bilgi için aşağıda gösterilen durumlara bakılır.

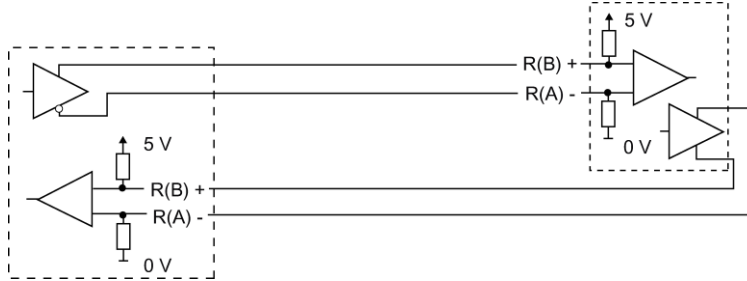
- Tam çift yönlü (RS422) dört kablo modu (noktadan noktaya bağlantı): ağınızdaki iki cihaz varken, bu seçeneği seçiniz. Alma hat başlangıç durumunda:
  - Besleme ve sonlandırma sağladığınız zaman hiçbiri seçeneğini seçiniz (Durum 3).
  - İç besleme ve sonlandırma kullanmak için ileri beslemeyi seçiniz (Durum 2).
  - İç besleme ve sonlandırma kullanmak ve her iki cihaz için kablo kopması algılamasını etkinleştirmek için ters beslemeyi seçiniz (Durum 1).
- Tam çift yönlü (RS422) dört kablo modu (çoklu master): Bir master ve birden fazla slave e sahip bir ağınız varsa master cihaz için bu seçeneği seçiniz. Alma hat başlangıç durumunda:
  - Besleme ve sonlandırma sağladığınız zaman hiçbiri seçeneğini seçiniz (Durum 3).
  - İç besleme ve sonlandırma kullanmak için ileri beslemeyi seçiniz (Durum 2).
  - Bu modda kablo kopması algılaması mümkün değil.
- Tam çift yönlü (RS422) dört kablo modu (çoklu master): Bir master ve birden fazla slave e sahip bir ağınız varsa slave cihazların tümü için bu seçeneği seçiniz. Alma hat başlangıç durumunda:
  - Besleme ve sonlandırma sağladığınız zaman hiçbiri seçeneğini seçiniz (Durum 3).
  - İç besleme ve sonlandırma kullanmak için ileri beslemeyi seçiniz (Durum 2).
  - İç besleme ve sonlandırma kullanmak ve slave cihazlar için kablo kopması algılamasını etkinleştirmek için ters beslemeyi seçiniz (Durum 1).

**Durum 1: Kablo kopması algılanmalı RS422**

- Çalışma modu: RS422
- Alma hattı başlangıç durumu: Ters besleme ( $R(A) > R(B) > 0V$  şeklinde besleme)
- Kablo kopması: Kablo kopması algılaması etkinleştirildi (Verici daima aktif)

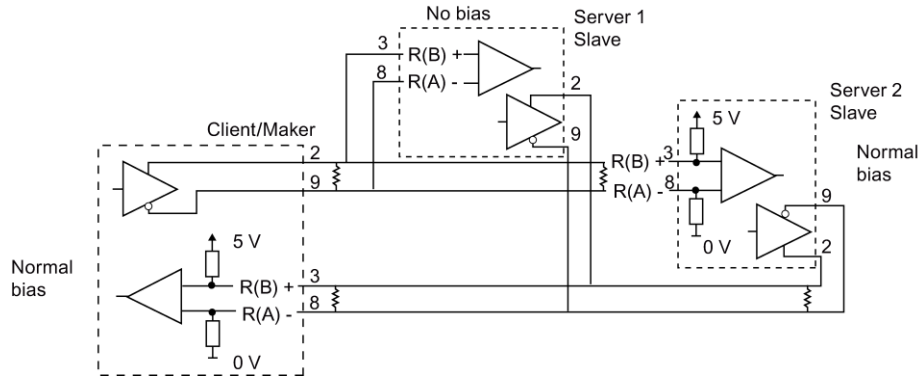

**Durum 2: RS422 kablo kopma algılaması yok, ileri besleme**

- Çalışma modu: RS422
- Alma hattı başlangıç durumu: İleri besleme ( $R(B) > R(A) > 0V$  şeklinde besleme)
- Kablo kopması: Kablo kopması algılaması yok (Verici yalnızca iletim yapılırken etkinleştirilir)


**Durum 3: RS422: kablo kopma algılaması yok, besleme yok**

- Çalışma modu: RS422
- Alma hattı başlangıç durumu: besleme yok
- Kablo kopması: Kablo kopması algılaması yok (Verici yalnızca iletim yapılırken etkinleştirilir)

Besleme ve sonlandırma ağın uç düğümlerinde kullanıcı tarafından eklenir.



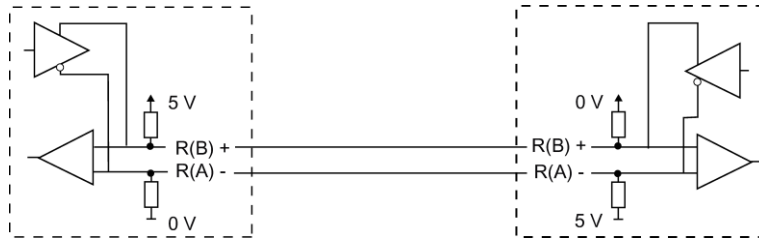
## RS485 Yapılandırma

RS485 modu için sadece tek bir çalışma modu vardır. Hat başlangıç durumunu alma için farklı seçimlerde daha fazla bilgi için, aşağıda gösterilen durumlara bakılır.

- Yarım çift yönlü (RS485) iki kablo modu. Alma hattı başlangıç durumunda:
  - Besleme ve sonlandırma sağladığınız zaman hiçbiri seçeneğini seçiniz (Durum 5).
  - İç besleme ve sonlandırma kullanmak için ileri beslemeyi seçiniz (Durum 4).

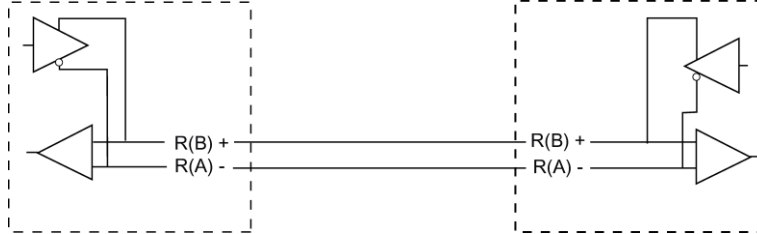
### Durum 4: RS485: İleri besleme

- Çalışma modu: RS485
- Alma hattı başlangıç durumu: İleri besleme ( $R(B) > R(A) > 0 V$  şeklinde besleme)



### Durum 5: RS485: besleme yok (Harici besleme)

- Çalışma modu: RS485
- Alma hattı başlangıç durumu: besleme yok (harici besleme gerekir)

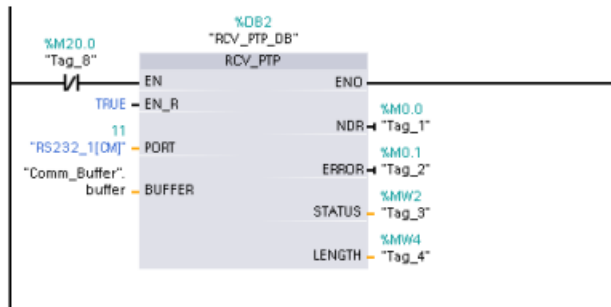


### 12.3.5.3 STEP 7 programı programlama

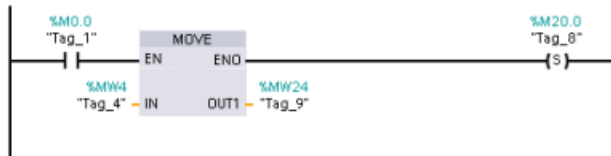
Örnek program, haberleşme arabelleği için küresel bir veri bloğunu, terminal emülatöründen veri almak için bir RCV\_PTP (Sayfa 697) komutunu ve arabelleği tekrar terminal emülatörüne yansıtmak için bir SEND\_PTP komutunu (Sayfa 694) kullanır. Örneği programlamak için, veri bloğu yapılandırmayı ve aşağıda açıklandığı gibi ana program bloğu OB 1'i ekleyiniz.

**Küresel veri bloğu "Comm\_Buffer:"** Küresel bir veri bloğu (DB) oluşturun ve "Comm\_Buffer" olarak isimlendiriniz. "dizi [0 .. 99] Byte" bir veri türü ile "buffer" olarak adlandırılan veri bloğunda bir değer oluşturunuz.

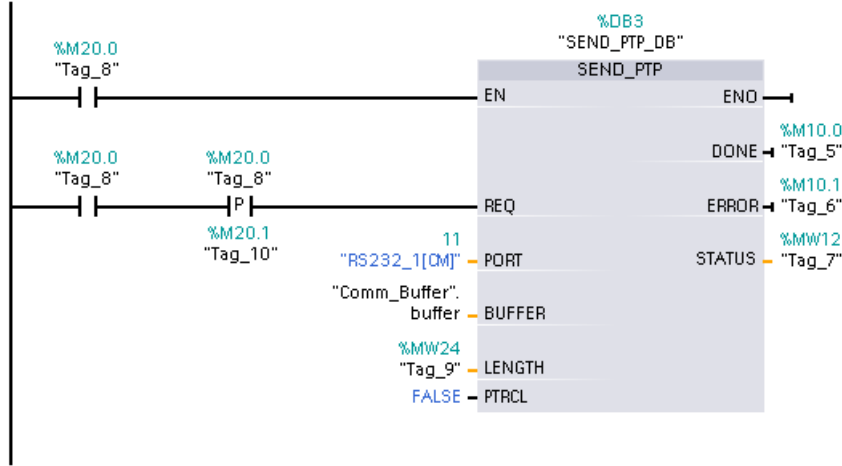
**Devre 1:** SEND\_PTP aktif olmadığında RCV\_PTP komutunu etkinleştiriniz. MW20.0 deki Tag\_8 devre 4'de göndermenin tamamlanma ve haberleşme modülünün böylece mesaj almak için hazır olma zamanını gösterir.



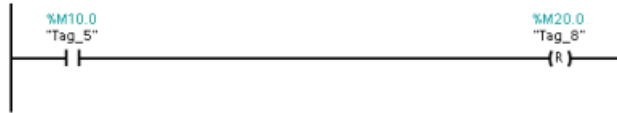
**Devre 2:** SEND\_PTP komutunu tetiklemek amacıyla bir bayrak (Tag\_8'de M20.0) ayarlamak için ve alınan bayt sayısının bir kopyasını yapmak için RCV\_PTP komutuyla belirlenen NDR değerini (Tag\_1 de M0.0) kullanınız.



**Devre 3:** M20.0 bayrağı ayarlandığında SEND\_PTP komutunu etkinleştiriniz. Ayrıca bir tarama için REQ girişini TRUE olarak ayarlamak için bu bayrağı kullanınız. REQ girişi, SEND\_PTP komutuna yeni bir isteğin aktarılacak olduğunu söyler. REQ girişinin sadece bir adet SEND\_PTP yürütmesi için TRUE olarak ayarlanması gerekir. İletim tamamlanıncaya kadar SEND\_PTP komutu her taramada yürütülür. Mesajın son baytı CM 1241 den aktarıldığında iletim tamamlanır. İletim tamamlandığında, DONE çıkışı (Tag\_5'de M10.0) bir adet SEND\_PTP yürütmesi için TRUE olarak ayarlanır.



**Devre 4:** SEND\_PTP'nin DONE çıkışı izlenir ve iletim işlemi tamamlandığında iletim bayrağını Tag\_8'de M20.0) sıfırlanır. İletim bayrağı sıfırlandığında, Devre 1 de ki RCV\_PTP komutu sonraki mesaj almak için etkinleştirilir.



#### 12.3.5.4 Terminal emülatörünü yapılandırma

Örnek programı desteklemek için bir terminal emülatörü kurmalısınız. Bilgisayarınızda HyperTerminal gibi herhangi bir terminal emülatörünü kullanabilirsiniz. Terminal emülatörünün aşağıdaki gibi ayarlarını düzenlemeden önce bağlantısının kesik modda olduğundan emin olunuz:

22. Terminal emülatörünü, PC (normalde COM1) üzerindeki RS232 portunu kullanacak şekilde ayarlayınız.
23. 9600 baud, 8 veri biti, parite yok (hiç), 1 stop biti ve akış kontrolü yok olacak şekilde portu yapılandırınız.
24. ANSI terminalinin emülasyonunu yapmak için Terminal emülatör ayarlarını değiştirin.
25. (Kullanıcı Enter tuşuna bastığında sonra) her satırdan sonra bir satır beslemesi göndermek için terminal emülatörü ASCII kurulumunu yapılandırınız.
26. Terminal emülatörü yazdıklarınızı görüntüleyecek şekilde karakterleri yerel olarak yansıtır.

### 12.3.5.5 Örnek programı çalıştırma

Örnek program alıştırması için şu adımları izleyiniz:

27.CPU' ya STEP 7 programını indirin ve RUN modunda olduğundan emin olunuz.

28.Yapılandırma değişikliklerini uygulamak ve CM 1241 de terminal oturumu açmak için terminal emülatörü üzerindeki "connect" düğmesine tıklayınız.

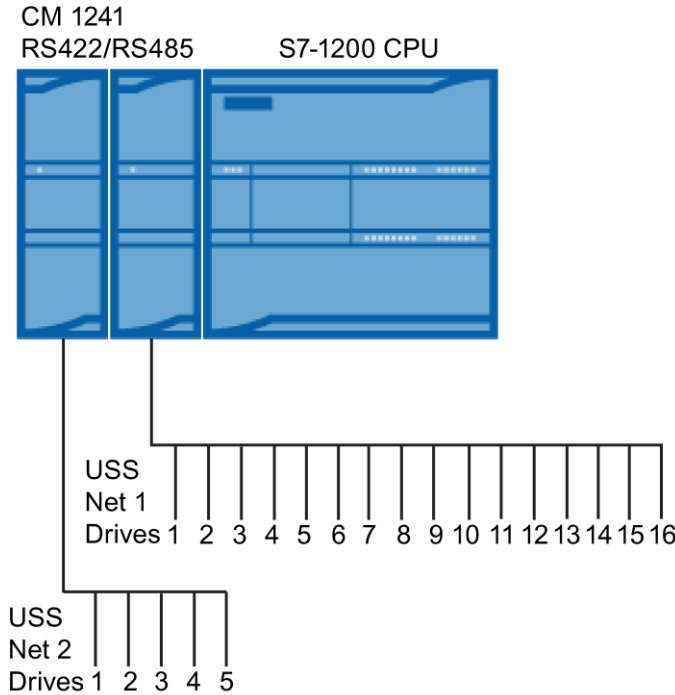
29.PC'de karakterleri yazınız ve Enter tuşuna basınız.

Terminal emülatörü CM 1241 ve CPU' ya karakterleri gönderir. CPU Programı daha sonra tekrar terminal emülatörüne karakterleri yansıtır.

## 12.4 Evrensel seri ara yüz (USS) haberleşmesi

USS komutları evrensel seri ara yüz (USS) protokolünü destekleyen motor sürücülerinin çalışmasını kontrol eder. CM 1241 RS485 haberleşme modüllerine veya CB 1241 RS485 haberleşme kartına RS485 bağlantıları aracılığıyla çoklu sürücüler ile iletişim kurmak için USS komutlarını kullanabilirsiniz. Üçe kadar CM 1241 RS422/RS485 modülü ve bir CB 1241 RS485 kartı, bir S7-1200 CPU' ya monte edilebilir. Her RS485 portu, on altı sürücüye kadar çalışabilir.

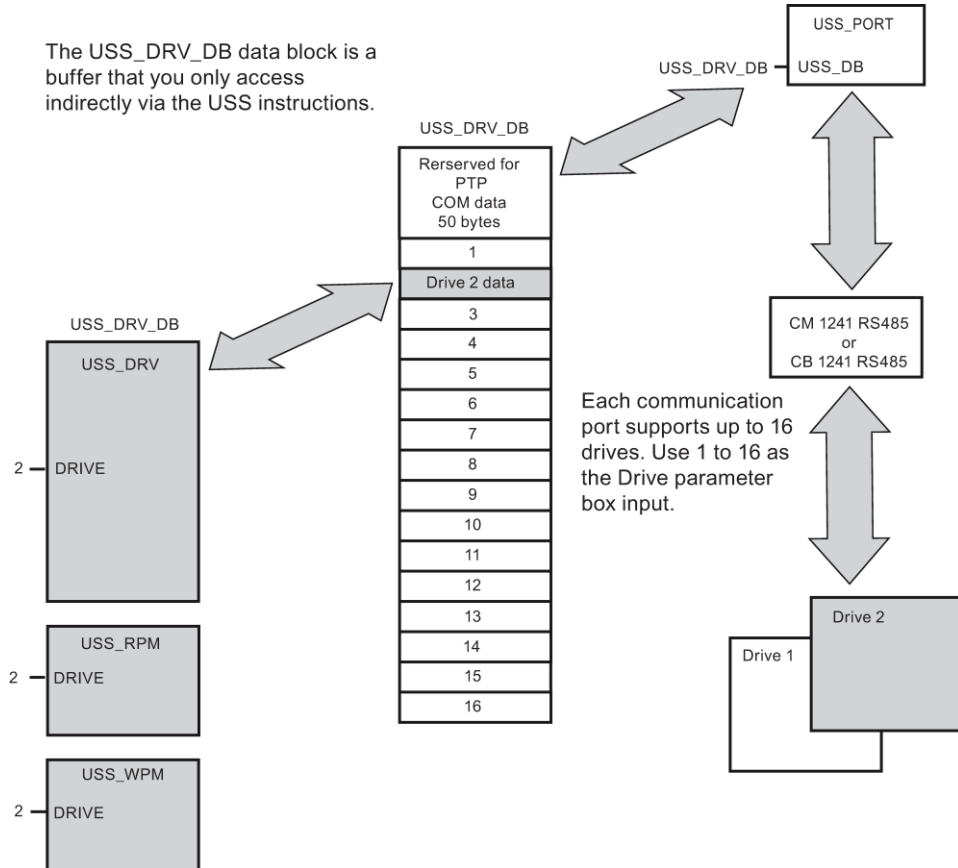
USS protokolü bir seri veri yolu üzerinden haberleşme için bir master-slave devresi kullanır. Master, seçilen slave e bir mesaj göndermek için adres parametrelerini kullanır. Bir slave in kendisi bunu yapmak için bir isteği ilk olarak almadan iletim yapamaz. Bireysel slave'ler arasında doğrudan bir mesaj aktarımı mümkün değildir. USS haberleşme yarı çift yönlü modda çalışır. Aşağıdaki USS gösterimi bir örnek sürücü uygulaması için bir devre şemasını gösterir.





### 12.4.1 USS protokolünü kullanma gereksinimleri

Dört adet USS komutu, USS protokolünü desteklemek için 1 FB ve 3 FC kullanır. Bir USS\_PORT kopya veri bloğu (DB) Her USS devresi için kullanılır. USS\_PORT kopya veri bloğu o USS devresindeki tüm sürücüler için geçici depolama ve arabellekler içerir. USS komutları bu veri bloğundaki bilgiyi paylaşır.



Tek bir RS485 portuna bağlı ( 16 kadar ) tüm sürücüler aynı USS devresinin bir parçasıdır. Farklı bir RS485 portuna bağlı tüm sürücüler, farklı bir USS devresinin bir parçasıdır. Her USS devresi benzersiz bir veri bloğu kullanılarak yönetilir. Tek bir USS devresi ile ilgili tüm komutların bu veri bloğunu paylaşması gerekir. Bu, tekli USS devresindeki tüm sürücülerini kontrol etmek için kullanılan tüm USS\_DRV , USS\_PORT , USS\_RPM ve USS\_WPM komutlarını içerir.

USS\_DRV komutu Fonksiyon Bloğu ( FB ) 'dur. USS\_DRV komutunu Program editörü içine yerleştirdiğinizde, bu FB ye yönelik bir DB atamak için "Çağrı seçenekleri" iletişim penceresiyle zorlanırsınız. Bu, USS devresi için bu programda ilk USS\_DRV komutu ise o zaman, varsayılan DB atamasını kabul edebilir ( ya da isterseniz adını değiştirin) ve yeni DB sizin için oluşturulur. Ancak bu kanal için ilk USS\_DRV komutu değilse, o zaman önce bu USS devresi için önceden atanan DB adını seçmek için " Çağrı seçenekleri " iletişim penceresinde açılır listeyi kullanmanız gerekir.

USS\_PORT , USS\_RPM ve USS\_WPM Komutlarının tümü fonksiyondu (FC ). Editöre bu FC'leri yerleştirdiğinizde hiçbir DB atanmaz. Bunun yerine, bu komutların " USS\_DB " girişine uygun DB referansını atamanız gerekir. Parametre alanına çift tıklayınız ve ardından mevcut DB adlarını görmek için parametre yardımcı ikonuna tıklayınız).

USS\_PORT işlevi noktadan noktaya (PtP) RS485 haberleşme portu aracılığıyla CPU ve sürücüler arasındaki gerçek iletişimi yönetir. Bu işlev için her çağrı bir sürücü ile bir iletişimi yönetir. Programınız sürücüler tarafından bir iletişim zaman aşımı önlemek için yeterince hızlı bu fonksiyonu çağırarak gerekir. Sen bir ana program döngüsü OB veya herhangi bir kesme OB bu işlevi çağırabilir.

Genellikle, bir döngüsel kesmeden OBUSS\_PORT işlevini çağırmanız. Döngüsel kesme OB'nin döngü süresi asgari arama aralığı (bir örnek olarak 1200 baud haberleşme, 350 ms ya da daha az bir döngüsel süre kullanılmalıdır) yaklaşık çağrı aralığının yarısı kadar ayarlanmalıdır.

USS\_DRV fonksiyon bloğu USS ağdaki bir belirtilen sürücüye program erişim sağlar. Onun girişler ve çıkışlar sürücü için statü ve kontroller bulunmaktadır. Ağda 16 sürücüler varsa, program en az 16 USS\_DRV aramaları, her bir sürücü için bir tane olması gerekir. Bu bloklar, sürücünün çalışmasını kontrol etmek için gerekli olan bir oranda aranmalıdır.

Sen yalnızca bir ana program döngüsü OB gelen USS\_DRV fonksiyon bloğu çağırabilir.

**⚠ DİKKAT****OB'lerden gelen USS komutlarının çağırılma hususları**

Sadece bir ana program çevrim OB sinden USS\_DRV, USS\_RPM ve USS\_WPM çağırın. USS\_PORT işlevi herhangi OB den genellikle döngüsel bir kesme OB sinden çağırılabilir.

Karşılık gelen USS\_PORT komutundan daha yüksek öncelikli OB de USS\_DRV, USS\_RPM veya USS\_WPM komutlarını kullanmayın. Örneğin, ana programda USS\_PORT ve döngüsel bir kesme OB sinde USS\_RPM komutunu yerleştirmeyin. USS\_PORT yürütme kesmesini engelleyecek bir arıza yaralanmalara neden olabilecek beklenmedik hatalar üretebilir.

USS\_RPM ve USS\_WPM fonksiyonları, uzaktan sürücü işletim parametrelerini okur ve yazar. Bu parametreler, sürücünün iç çalışmasını kontrol eder. Bu parametrelerin tanımı için Sürücü kılavuzuna bakınız. Programınız gerekli olduğu kadar çok bu fonksiyonları içerebilir, ancak herhangi bir zamanda sürücü başına yalnızca bir okuma veya yazma isteği etkin olabilir. Sadece bir ana program çevrim OB sinden USS\_RPM ve USS\_WPM işlevlerini çağırabilirsiniz.

**Sürücüsü ile iletişim kurmak için gerekli olan zamanı hesaplama**

Sürücü ile haberleşmeler S7-1200 tarama çevrimiyle asenkronudur. Bir sürücü haberleşme işlemi tamamlanmadan önce S7-1200 genellikle bir kaç tarama tamamlar.

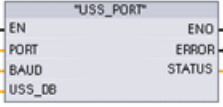
USS\_PORT aralığı, bir sürücü işlemi için gereken zamandır. Aşağıdaki çizelgede, her haberleşme baud hızı için minimum USS\_PORT aralığını gösterilmiştir. USS\_PORT aralığından daha sık olarak USS\_PORT işlevi çağırması işlemlerin sayısını artırmaz. Haberleşme hataları işlemi tamamlamak için 3 denemeye neden olduysa, Sürücü zaman aşımı aralığı, bir işlem için alınabilecek süre miktarıdır. Varsayılan olarak, USS protokol kütüphanesi otomatik olarak her bir işlem için 2'ye kadar deneme yapar.

Çizelge 12- 34 Zaman gereksinimlerini hesaplama

Baud hızı	Hesaplanan minimum USS_PORT çağrı aralığı (milisaniye)	Sürücü başına sürücü mesaj aralığı aşımı (milisaniye)
1200	790	2370
2400	405	1215
4800	212.5	638
9600	116.3	349
19200	68.2	205
38400	44.1	133
57600	36.1	109
115200	28.1	85

## 12.4.2 USS\_PORT (USS devre üzerinden haberleşmeyi düzenleme) komutu

Çizelge 12- 35 USS\_PORT komutu

LAD / FBD	SCL	Açıklama
	<pre> USS_PORT (   PORT:=_uint_in_,   BAUD:=_dint_in_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_,   USS_DB:=_fbtref_inout_); </pre>	USS_PORT komutu USS devre üzerinden haberleşmeyi yönetir.

Çizelge 12- 36 Parametreler için veri tipleri



Parametre ve tipi	Veri tipi	Açıklama
PORT	IN	Port
BAUD	IN	DInt
USS_DB	INOUT	USS_BASE
ERROR	OUT	Bool
STATUS	OUT	Word

Tipik olarak programınızda PtP haberleşme portu başına sadece bir USS\_PORT komutu vardır ve bu fonksiyonun her çağrılışı tek bir sürücüden veya buraya yapılan bir iletimi işler. Bir USS devresi ve PtP haberleşme portu ile ilişkili tüm USS fonksiyonlarının aynı kopya DB'yi kullanması gerekir.

Programınız, sürücü zaman aşımını önlemek için USS\_PORT komutunu yeterince sık yürütmesi gerekir. USS\_PORT genellikle sürücü zaman aşımı önlemek ve USS\_DRV çağrılarına yönelik mevcut en son USS veri güncellemelerini tutmak için bir döngüsel kesme OB sinden çağrılır.

### 12.4.3 USS\_DRV (Sürücü ile verileri değiştirme) komutu

Çizelge 12- 37 USS\_DRV komutu

LAD / FBD	SCL	Açıklama
<p>Varsayılan görünüm</p> 	<pre>"USS_DRV_DB" (   RUN:=_bool_in_,   OFF2:=_bool_in_,   OFF3:=_bool_in_,   F_ACK:=_bool_in_,   DIR:=_bool_in_,   DRIVE:=_usint_in_,   PZD_LEN:=_usint_in_,   SPEED_SP:=_real_in_,   CTRL3:=_word_in_,   CTRL4:=_word_in_,   CTRL5:=_word_in_,   CTRL6:=_word_in_,   CTRL7:=_word_in_,   CTRL8:=_word_in_,   NDR=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_,   RUN_EN=&gt;_bool_out_,   D_DIR=&gt;_bool_out_,   INHIBIT=&gt;_bool_out_,   FAULT=&gt;_bool_out_,   SPEED=&gt;_real_out_,   STATUS1=&gt;_word_out_,   STATUS3=&gt;_word_out_,   STATUS4=&gt;_word_out_,   STATUS5=&gt;_word_out_,   STATUS6=&gt;_word_out_,   STATUS7=&gt;_word_out_,   STATUS8=&gt;_word_out_ );</pre>	<p>USS_DRV komut istek mesajları oluşturarak ve sürücü tepki mesajlarını yorumlayarak bir sürücü ile veri değişimleri yapar. Her bir sürücü için Ayrı bir fonksiyon bloğunun kullanılması gerekir, ancak bir USS devresi ve PtP haberleşme portu ile ilgili tüm USS fonksiyonları aynı kopya veri bloğunu kullanması gerekir. İlk USS_DRV komut yerleştirdiğinizde DB adı oluşturmanız gerekir sonra ilk komut kullanımı ile oluşturulan DB'yi referans gösterirsiniz.</p> <p>Komutu yerleştirdiğinizde STEP7 otomatik olarak DB oluşturur.</p>
<p>Genişletilmiş görünüm</p> 		

<sup>1</sup> LAD ve FBD: Tüm parametreleri ortaya çıkarmak için kutunun altını tıklatarak kutuyu genişletiniz. Gri olan parametre pinleri isteğe bağlıdır ve parametre ataması gerekli değildir.

Çizelge 12- 38 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
RUN	IN	Bool	Başlangıç bitini sür: True olduğunda, bu giriş sürücünün önceden ayarlanmış hızda çalıştırmasını sağlar. Bir sürücü çalışırken RUN false olduğunda, motor durma noktasına gidecektir. Bu davranış, güç düşmesi (OFF2) veya motorun frenlenmesinden (OFF3) farklıdır.
OFF2	IN	Bool	Elektrik stop biti: false olduğunda, bu bit sürücünün frenleme yapmaksızın salınarak durmasına neden olur.
OFF3	IN	Bool	Hızlı stop biti: false olduğunda, bu bit, sürücünün salınarak durmasına izin vermesinden ziyade frenleyerek, hızlı şekilde durmasına neden olur.
F_ACK	IN	Bool	Arıza kabul biti: Bu bit, bir sürücüdeki arıza bitini sıfırlamak için ayarlanır. Artık bitin önceki arızayı belirtmesine gerek olmadığını sürücüye göstermek için Arıza temizlendikten sonra bit ayarlanır.
DIR	IN	Bool	Sürücü Yön kontrolü: Bu bit, yönün (pozitif SPEED_SP için) ileri olduğunu göstermek için ayarlanır.
DRIVE	IN	USInt	Sürücü Adresi: Bu giriş, USS sürücünün adresidir. Geçerli aralık sürücü 1 ila sürücü 16' dır.
PZD_LEN	IN	USInt	Word uzunluğu: Bu PZD verinin word sayısıdır. Geçerli değerler 2, 4, 6 veya 8 Word'dür. Varsayılan değer 2'dir.
SPEED_SP	IN	Real	Hız ayar noktası: Bu, yapılandırılmış frekansın bir yüzdesi olarak sürücünün hızıdır. Pozitif bir değer, ileri yönü (DIR doğru olduğunda) belirtir. Geçerli aralık 200.00 ila -200.00'dür.
CTRL3	IN	Word	Kontrol word 3: Sürücü üzerinde bir kullanıcı tarafından yapılandırılabilen parametreye yazılan bir değer. Bunu sürücü üzerinde yapılandırmanız gerekir. (isteğe bağlı parametre)
CTRL4	IN	Word	Kontrol word 4: Sürücü üzerinde bir kullanıcı tarafından yapılandırılabilen parametreye yazılan bir değer. Bunu sürücü üzerinde yapılandırmanız gerekir. (isteğe bağlı parametre)
CTRL5	IN	Word	Kontrol word 5: Sürücü üzerinde bir kullanıcı tarafından yapılandırılabilen parametreye yazılan bir değer. Bunu sürücü üzerinde yapılandırmanız gerekir. (isteğe bağlı parametre)
CTRL6	IN	Word	Kontrol word 6: Sürücü üzerinde bir kullanıcı tarafından yapılandırılabilen parametreye yazılan bir değer. Bunu sürücü üzerinde yapılandırmanız gerekir. (isteğe bağlı parametre)
CTRL7	IN	Word	Kontrol word 7: Sürücü üzerinde bir kullanıcı tarafından yapılandırılabilen parametreye yazılan bir değer. Bunu sürücü üzerinde yapılandırmanız gerekir. (isteğe bağlı parametre)
CTRL8	IN	Word	Kontrol word 8: Sürücü üzerinde bir kullanıcı tarafından yapılandırılabilen parametreye yazılan bir değer. Bunu sürücü üzerinde yapılandırmanız gerekir. (isteğe bağlı parametre)
NDR	OUT	Bool	Yeni veriler hazır: True olduğunda, Bit, çıkışların yeni bir haberleşme isteğinden gelen verileri içerdiğini gösterir.
ERROR	OUT	Bool	Hata oluştu: True olduğunda, bu, bir hata oluştuğunu gösterir ve STATUS çıkışı geçerlidir. Diğer tüm çıkışlar bir hata üzerine sıfıra ayarlanır. Haberleşme hataları sadece USS_PORT komutu ERROR ve STATUS çıkışları üzerinde raporlanır.
STATUS	OUT	Word	İsteğin durum değeri taramanın sonucunu gösterir. Bu sürücüden döndürülen bir durum kelimesi değildir.
RUN_EN	OUT	Bool	Çalıştırma etkinleştirildi: Bu bit sürücünün çalışıp çalışmadığını gösterir.

Parametre ve tipi	Veri tipi	Açıklama	
D_DIR	OUT	Bool	Sürücü yönü: Bu bit, sürücünün ileri çalışır durumda olup olmadığını gösterir.
INHIBIT	OUT	Bool	Sürücü engellendi: Bu bit sürücü üzerindeki engelleme bitinin durumunu gösterir.
FAULT	OUT	Bool	Sürücü Arızası: Bu bit sürücüde bir arıza kaydı olduğunu gösterir. Sorunu çözmeniz ve daha sonra ayarlandığında F_ACK bitini bu biti temizleyecek şekilde ayarlamamız gerekir.
SPEED	OUT	Real	Sürücünün mevcut hızı (sürücü durum word 2 ölçekli değeri): yapılandırılan hızın bir yüzdesi olarak sürücünün hız değeri.
STATUS1	OUT	Word	Durum Word 1 Sürücü: Bu değer, bir sürücünün sabit durum bitlerini içerir.
STATUS3	OUT	Word	Durum Word 3 Sürücü: Bu değer, sürücü üzerinde bir kullanıcı tarafından yapılandırılabilir durum word içerir.
STATUS4	OUT	Word	Durum Word 4 Sürücü: Bu değer, sürücü üzerinde bir kullanıcı tarafından yapılandırılabilir durum word içerir.
STATUS5	OUT	Word	Durum Word 5 Sürücü: Bu değer, sürücü üzerinde bir kullanıcı tarafından yapılandırılabilir durum word içerir.
STATUS6	OUT	Word	Durum Word 6 Sürücü: Bu değer, sürücü üzerinde bir kullanıcı tarafından yapılandırılabilir durum word içerir.
STATUS7	OUT	Word	Durum Word 7 Sürücü: Bu değer, sürücü üzerinde bir kullanıcı tarafından yapılandırılabilir durum word içerir.
STATUS8	OUT	Word	Durum Word 8 Sürücü: Bu değer, sürücü üzerinde bir kullanıcı tarafından yapılandırılabilir durum word içerir.

İlk USS\_DRV yürütme gerçekleştiğinde, USS adresi (parametre SÜRÜCÜ ) tarafından belirtilen sürücü kopya DB de başlatılır. Bu başlatmadan sonra, USS\_PORT un sonraki yürütmeleri bu sürücü numarasındaki sürücüyle haberleşmeyi başlatabilir.

Sürücü numarasını değiştirme, kopya DB'yi başlatan CPU nun STOP-RUN mod geçişini gerektirir. Giriş parametreleri, USS\_TX mesaj arabelleği içine yapılandırılır ve çıkışlar, varsa bir " önceki" geçerli bir yanıt arabelleğinden okunur. USS\_DRV yürütülmesi sırasında herhangi bir veri iletimi yoktur. USS\_PORT çalıştırıldığında sürücüler iletişime geçer. USS\_DRV sadece gönderilecek mesajları yapılandırır ve bir önceki istekten alınmış olabilen verileri yorumlar.

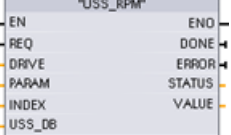
DIR girişini ( Bool ) kullanarak ya da işareti (pozitif veya negatif ) SPEED\_SP girişi ile (Gerçek ) kullanarak sürücünün dönme yönünü kontrol edebilirsiniz. Aşağıdaki Çizelge, motorun ileri dönüş için kablajının yapıldığı varsayılarak, bu girişlerin sürücü yönünü belirlemek için birlikte nasıl çalıştıklarını gösterir.

Çizelge 12- 39 SPEED\_SP ve DIR parametrelerin etkileşimi

SPEED_SP	DIR	Sürücü dönüş yönü
Değer > 0	0	Geri
Değer > 0	1	İleri
Değer < 0	0	İleri
Değer < 0	1	İleri

## 12.4.4 USS\_RPM (Sürücüden gelen parametreleri okuma) komutu

Çizelge 12- 40 USS\_RPM komutu

LAD / FBD	SCL	Açıklama
	<pre>USS_RPM(REQ:=_bool_in_, DRIVE:=_usint_in_, PARAM:=_uint_in_, INDEX:=_uint_in_, DONE=&gt;_bool_out_, ERROR=&gt;_bool_out_, STATUS=&gt;_word_out_, VALUE=&gt;_variant_out_, USS_DB:=_fbtref_inout_);</pre>	USS_RPM komutu bir sürücüden gelen bir parametreyi okur. Bir USS devresi ve PtP haberleşme portu ile ilişkili tüm USS fonksiyonları aynı veri bloğunu kullanmalıdır. USS_RPM bir ana program döngü OB'den çağrılmalıdır.

Çizelge 12- 41 Parametreler için veri tipleri

Parametre tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Gönderme isteği: True olduğunda, REQ, yeni bir okuma isteğinin arzu edildiğini gösterir. Bu parametre için istek zaten beklemede ise, bu göz ardı edilir.
DRIVE	IN	USInt	Sürücü adresi: DRIVE, USS sürücünün adresidir. Geçerli aralık sürücü 1 ila sürücü 16'dır.
PARAM	IN	UInt	Parametre numarası: PARAM, sürücü parametrelerinin yazıldığını belirtir. Bu parametrenin aralığı 0-2047'dir. Bazı sürücülerde, en önemli bayt 2047 den fazla olan PARAM değerlerine erişebilir. Genişletilmiş aralığa nasıl erişim sağlanacağıyla ilgili ayrıntılar için sürücü kılavuzuna bakınız.
INDEX	IN	UInt	Parametre indeksi: INDEX hangi Sürücü Parametre indeksinin yazılacağını belirtir. En önemsiz baytın (0-255) aralığındaki gerçek indeks değeri olan bir 16-bit değeri. En Önemli Byte sürücü tarafından da kullanılabilir ve sürücüye özgü olabilir. Ayrıntılar için sürücü el kitabına bakınız.
USS_DB	INOUT	USS_BASE	Bir USS_DRV komutunu programınızda yerleştirdiğinizde oluşturulan ve başlatılan kopya DB'nin ismi.
VALUE	IN	Word, Int, UInt, Dword, DInt, UDIInt, Real	Bu okunan parametrenin değeri olup DONE biti true olduğunda geçerlidir.
DONE <sup>1</sup>	OUT	Bool	True olduğunda, VALUE çıkışının önceden istenen okuma parametre değerini tuttuğunu gösterir. Bu bit USS_DRV sürücüden gelen okuma yanıt verilerini gördüğünde ayarlanır. Bu bit, başka USS_RPM yoklaması yoluyla veya USS_DRV için sonraki iki çağrıdan ikincisinde yanıt verilerini istediğinizde sıfırlanır

Parametre tipi		Veri tipi	Açıklama
ERROR	OUT	Bool	Hata oluştu: True olduğunda, ERROR bir hata oluştuğunu gösterir ve STATUS çıkışı geçerlidir. Diğer tüm çıkışlar bir hata üzerine sıfıra ayarlanır. Haberleşme hataları sadece USS_PORT komutu ERROR ve STATUS çıkışları üzerinde raporlanır.
STATUS	OUT	Word	STATUS okuma isteğinin sonucunu gösterir. Ek bilgiler bazı durum kodları için "USS_Extended_Error" değişkeninde mevcuttur.

- <sup>1</sup> DONE biti, geçerli verinin referans alınan motor sürücüsünde okunduğunu ve CPU' ya teslim edildiğini gösterir. Bu, USS kütüphanesinin hemen ardına başka bir parametre okuma yeteneğine sahip olduğunu göstermez. Boş PKW isteği, motor sürücüsüne gönderilmesi gerekir ve belirli bir sürücüye yönelik parametre kanalı, kullanım için mevcut hale gelmeden önce komut tarafından kabul edilmelidir. Hemen ardına belirtilen motor sürücüsü için bir USS\_RPM veya USS\_WPM FC çağrısı, bir 0x818A hataya neden olur.

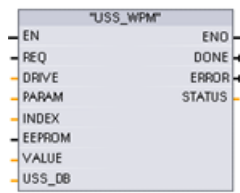
## 12.4.5 USS\_WPM (Sürücüdeki parametreleri değiştirme) komutu

### Not

#### EEPROM yazma işlemleri (USS sürücü içindeki EEPROM için)

EEPROM kalıcı yazma işlemini aşırı kullanmayınız. EEPROM ömrünü uzatmak için EEPROM yazma işlemlerinin sayısını en aza indiriniz.

Çizelge 12- 42 USS\_WPM komutu

LAD / FBD	SCL	Açıklama
	<pre>USS_WPM (REQ:=_bool_in_, DRIVE:=_usint_in_, PARAM:=_uint_in_, INDEX:=_uint_in_, EEPROM:=_bool_in_, VALUE:=_variant_in_, DONE=&gt;_bool_out_, ERROR=&gt;_bool_out_, STATUS=&gt;_word_out_, USS_DB:=_fbtref_inout_);</pre>	<p>USS_WPM komutu sürücüde bir parametre değiştirir. Bir USS devresi ve PtP haberleşme portu ile ilişkili tüm USS fonksiyonları aynı veri bloğunu kullanmalıdır. USS_WPM bir ana program çevrim OB'sinden çağrılmalıdır.</p>



Çizelge 12- 43 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Gönderme isteği: True olduğunda, REQ, yeni bir okuma isteğinin arzu edildiğini gösterir. Bu parametre için istek zaten beklemede ise, bu göz ardı edilir.
DRIVE	IN	USInt	Sürücü adresi: DRIVE, USS sürücünün adresidir. Geçerli aralık sürücü 1 ila sürücü 16'dır.
PARAM	IN	UInt	Parametre numarası: PARAM, sürücü parametrelerinin yazıldığını belirtir. Bu parametrenin aralığı 0-2047'dir. Bazı sürücülerde, en önemli bayt 2047 den fazla olan PARAM değerlerine erişebilir. Genişletilmiş aralığa nasıl erişim sağlayıcıyla ilgili ayrıntılar için sürücü kılavuzuna bakınız.
INDEX	IN	UInt	Parametre indeksi: INDEX hangi Sürücü Parametre indeksinin yazılacağını belirtir. En önemsiz baytın (0-255) aralığındaki gerçek indeks değeri olan bir 16-bit değeri. En Önemli Bayt sürücü tarafından da kullanılabilir ve sürücüye özgü olabilir. Ayrıntılar için sürücü el kitabına bakınız.
EEPROM	IN	Bool	EEPROM Sürücüde saklamak: true olduğunda, bir yazma sürücü parametre işlemi, sürücü EEPROM unda saklanır. False ise, yazma geçicidir ve sürücü güç çevriminde ise tutulamayacaktır.
VALUE	IN	Word, Int, UInt, Dword, DInt, UInt, Real	Bu yazılacak parametrenin değeri olup REQ biti true olduğunda geçerlidir.
USS_DB	INOUT	USS_BASE	Bir USS_DRV komutunu programınızda yerleştirdiğinizde oluşturulan ve başlatılan kopya DB'nin ismi.
DONE <sup>1</sup>	OUT	Bool	True olduğunda, VALUE çıkışının önceden istenen okuma parametre değerini tuttuğunu gösterir. Bu bit USS_DRV sürücünden gelen okuma yanıt verilerini gördüğünde ayarlanır. Bu bit, başka USS_WPM yoklaması yoluyla veya USS_DRV için sonraki iki çağrıdan ikincisinde yanıt verilerini istediğinizde sıfırlanır.
ERROR	OUT	Bool	True olduğunda, ERROR bir hata oluştuğunu gösterir ve STATUS çıkışı geçerlidir. Diğer tüm çıkışlar bir hata üzerine sıfıra ayarlanır. Haberleşme hataları sadece USS_PORT komutu ERROR ve STATUS çıkışları üzerinde raporlanır.
STATUS	OUT	Word	STATUS yazma isteğinin sonucunu gösterir. Ek bilgiler bazı durum kodları için "USS_Extended_Error" değişkeninde mevcuttur.

<sup>1</sup> DONE biti, geçerli verinin referans alınan motor sürücüsünde okunduğunu ve CPU' ya teslim edildiğini gösterir. Bu, USS kütüphanesinin hemen ardına başka bir parametre okuma yeteneğine sahip olduğunu göstermez. Boş PKW isteği, motor sürücüsüne gönderilmesi gerekir ve belirli bir sürücüye yönelik parametre kanalı, kullanım için mevcut hale gelmeden önce komut tarafından kabul edilmelidir. Hemen ardına belirtilen motor sürücüsü için bir USS\_RPM veya USS\_WPM FC çağırısı, bir 0x818A hataya neden olur.

## 12.4.6 USS durum kodları

USS komutu durum kodları USS fonksiyonlarının STATUS çıkışında döndürülür.

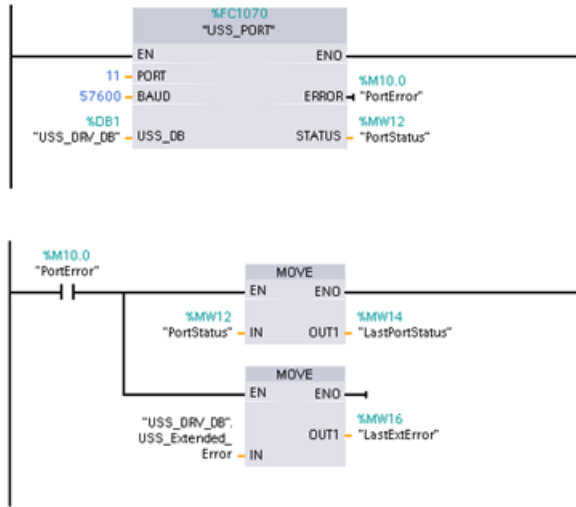
Çizelge 12- 44 STATUS kodları <sup>1</sup>

STATUS (W#16#....)	Açıklama
0000	Hata yok
8180	Sürücü yanıt uzunluğu ile sürücüden alınan karakterler eşleşmiyor. Hatanın oluştuğu sürücü numarası "USS_Extended_Error" değişkeninde döndürülür. Bu çizelgenin altındaki genişletilmiş hata açıklamasına bakınız.
8181	VALUE parametreleri bir Word, Real veya Dword veri tipi değildir.
8182	Kullanıcı, bir parametre değeri için Word sağlar ve yanıt olarak sürücüden bir DWord veya Real alır.
8183	Kullanıcı, bir parametre değeri için DWord veya Real sağlar ve yanıt olarak sürücüden bir Word alır
8184	Sürücüden gelen yanıt telegramın kötü sağlaması vardı. Hatanın oluştuğu sürücü numarası "USS_Extended_Error" değişkeninde döndürülür. Bu çizelgenin altındaki genişletilmiş hata açıklamasına bakınız.
8185	Kural dışı sürücü adresi (geçerli sürücü adresi aralığı: 1-16)
8186	Hız ayar noktası, geçerli aralığın dışında (geçerli hız SP aralığı: - % 200 ila % 200).
8187	Yanlış sürücü numarası gönderilen isteğe yanıt verdi. Hatanın oluştuğu sürücü numarası "USS_Extended_Error" değişkeninde döndürülür. Bu çizelgenin altındaki genişletilmiş hata açıklamasına bakınız.
8188	Belirtilen Kuraldışı PZD word uzunluğu (geçerli aralık = 2, 4, 6 veya 8 word)
8189	Kuraldışı Baud Hızı belirtildi.
818A	Parametre istek kanalı bu sürücü için başka bir istek tarafından kullanılıyor.
818B	Sürücü, isteklere ve denemelere cevap vermedi. Hatanın oluştuğu sürücü numarası "USS_Extended_Error" değişkeninde döndürülür. Bu çizelgenin altındaki genişletilmiş hata açıklamasına bakınız.
818C	Sürücü, parametre istek işlemi üzerine genişletilmiş bir hata döndürdü. Bu çizelgenin altındaki genişletilmiş hata açıklamasına bakınız.
818D	Sürücü, parametre istek işlemi üzerine kuraldışı erişim hatası döndürdü. Parametre erişiminin sınırlı olabilme nedeniyle ilgili bilgi için sürücü el kitabına bakınız.
818E	Sürücü başlatılmadı. USS_DRV, bu sürücü için, en az bir kez çağrılmadığında USS_RPM veya USS_WPM için bu hata kodu döndürülür. Bu, yeni bir giriş olarak sürücüyü başlattığı için USS_DRV'nin ilk taramadaki başlatmasının, bekleyen parametre okuma veya yazma isteği üzerine yazmasını engeller. Bu hatayı düzeltmek için, bu sürücü sayısı için USS_DRV çağrılır.
80Ax-80Fx	USS Kütüphanesi tarafından çağrılan PtP haberleşme FB'lerinden belirli hatalar döndürülür - Bu hata kodu değerleri USS kütüphanesi tarafından değiştirilmez ve PtP komut açıklamalarında tanımlanmıştır.

<sup>1</sup> Yukarıda listelenen USS komut hatalarına ek olarak, hatalar temel PtP haberleşme komutlarından döndürülebilir.

Birkaç STATUS kodu için, ek bilgiler, USS\_DRV Kopya DB'nin "USS\_Extended\_Error" değişkeninde sağlanır. Onaltılık 8180, 8184, 8187 ve 818B STATUS kodları için USS\_Extended\_Error, haberleşme hatasının oluştuğu sürücü numarasını içerir. STATUS kod onaltılık 818C için USS\_Extended\_Error, bir USS\_RPM veya USS\_WPM komutunu kullanırken sürücüden dönen bir sürücü hata kodunu içerir.

Haberleşme hataları (STATUS = 16 # 818B) sadece USS\_PORT komutun üzerinde değil USS\_DRV komutunun üzerinde raporlanır. Devre düzgün sonlandırılmadıysa bir sürücünün RUN a gitmesi mümkündür ancak USS\_DRV komutu, çıkış parametreleri için tüm 0'ları gösterecektir. Bu durumda, yalnızca USS\_PORT komutunun üzerinde haberleşme hatasını tespit edebilirsiniz. Bu hata sadece bir tarama için görünür olduğundan, aşağıdaki örnekte gösterildiği gibi bazı yakalama mantığı eklemek gerekir. USS\_PORT komutunun hata biti TRUE olduğunda, bu örnekte, daha sonra STATUS ve USS\_Extended\_Error değerleri M belleğe kaydedilir. STATUS kod değeri onaltılık 8180, 8184, 8187 veya 818B olduğunda sürücü numarası, USS\_Extended\_Error değişkeni içine yerleştirilir.



**Devre 1** "PortStatus" port durumu ve "USS\_DRV\_DB".

USS\_Extended\_Error genişletilmiş hata kodu değerleri yalnızca bir program taraması için geçerlidir. Bu değerler daha sonra işlenmek için yakalanmalıdır.

**Devre 2** "PortError" kontağı "LastPortStatus" da "PortStatus" değerinin ve LastExtError' de "USS\_DRV\_DB".

USS\_Extended\_Error değerinin depolanmasını tetikler.

USS sürücüler, bir sürücünün dahili parametrelerine okuma ve yazma desteği verir. Bu özellik sürücünün uzaktan kontrol ve yapılandırmasını sağlar. Bir sürücünün mevcut mod için aralık dışı değerler veya kuraldışı istekler gibi hatalar nedeniyle sürücü parametreleri erişim işlemleri başarısız olabilir. Sürücü, "USS\_Extended\_Error" değişkeninde döndürülen bir hata kodu değerini oluşturur. Bu hata kodu değeri yalnızca bir USS\_RPM veya USS\_WPM komutunun son yürütülmesi için geçerlidir. STATUS kod değeri onaltılık 818C olduğunda sürücü hata kodu USS\_Extended\_Error değişkenine konur. "USS\_Extended\_Error" ün bir hata kodu değeri, sürücü modeline bağlıdır. Parametre işlemlerini okuma ve yazmayla ilgili genişletilmiş hata kodlarının açıklaması için sürücünün el kitabına bakınız.

## 12.4.7 Genel sürücü kurulum bilgisi


### Genel sürücü kurulum gerekleri

- Sürücüler 4 PKW word kullanmak üzere ayarlanmış olması gerekir.
- Sürücüler 2, 4, 6 veya 8 PZD word için yapılandırılabilir.
- Sürücüde PZD word sayısı o sürücü için USS\_DRV komutu üzerinde PZD\_LEN girişiyle eşleşmelidir.
- Tüm sürücülerdeki baud hızı USS\_PORT komutu üzerinde BAUD girişiyle eşleşmesi gerekir.
- Sürücü uzaktan kumanda için ayarlanmış olması gerekir.
- Sürücü COM bağlantısı üzerinde USS ye frekans ayar noktası için ayarlanmış olması gerekir.
- Sürücü adresi, 1-16 ayarlanmış olması ve bu sürücü için USS\_DRV blokta DRIVE girişiyle eşleşmesi gerekir.
- Sürücü yön kontrolü, sürücü ayar-noktasının polaritesini kullanmak üzere ayarlanmış olması gerekir.
- RS485 devresi düzgün sonlandırılmalıdır.

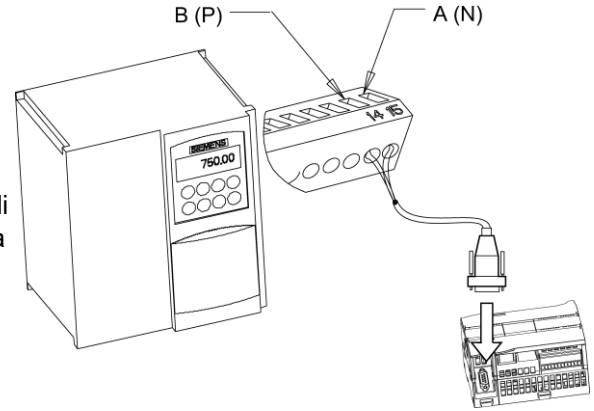
### MicroMaster sürücüsünün bağlanması

SIEMENS, MicroMaster sürücüler hakkındaki bu bilgileri bir örnek olarak vermiştir. Diğer sürücüler için kurulum yönergeleri için sürücünün el kitabına bakınız.

MicroMaster Serisi 4 (MM4) sürücüsüne bağlantı yapmak için RS485 kablo uçlarını USS işlemi için sağlanan iki kafesli-kelepçeye, vidasız terminaller içine yerleştiriniz. Standart PROFIBUS kablosu ve konektörler S7-1200 bağlantısı için kullanılabilir.

 <b>DIKKAT</b>
<b>Farklı referans potansiyellerine sahip ara bağlantı ekipmanları, istenmeyen akımların ara bağlantı kablosu üzerinden akmasına neden olabilir</b>
Bu istenmeyen akımlar, haberleşme hatalarına veya ekipman hasarına neden olabilir. Eğer bir haberleşme kablosu ile bağlantı yapmak üzere olduğunuz tüm ekipmanlar ortak devre referansını paylaştığından veya istenmeyen akımlara karşı izole edildiğinde emin olunuz. Zırh, 9-pin konektör üzerinde şase toprağına veya pin 1 e bağlı olmalıdır. MicroMaster üzerindeki 2 ila -0 V kablaj terminalini şase toprağına bağlamanız önerilir.

RS485 kablosunun karşı ucundaki iki tel MM4 Sürücü terminal bloklarına takılmalıdır. Bir MM4 sürücüde kablo bağlantısı yapmak amacıyla, terminal bloklarına erişmek için sürücü kapağını/kapaklarını çıkarın. Ayrıntılar özel sürücü kapaklarını çıkarmakla ilgili ayrıntılar için MM4 kullanım kılavuzuna bakınız.



Terminal bağlantıları sayısal olarak etiketlidir. S7-1200 tarafında bir PROFIBUS konektörü kullanarak, kablonun A terminalini, sürücü terminal 30 (MM440) veya sürücü terminal 15'e (bir MM420 için) bağlayınız. B (P) A (N) kablo konektörünün B terminalini Terminal 14 (MM420) veya terminal 29'a (MM440) bağlayınız.

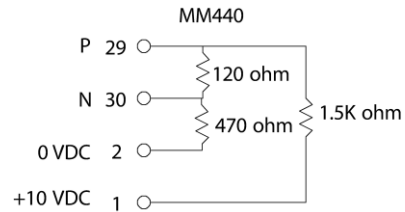
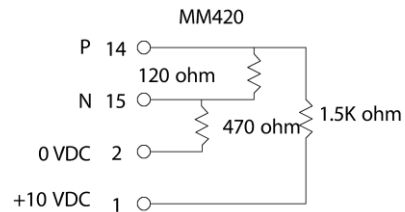
S7-1200 devredeki bir sonlandırıcı düğüm ise veya bağlantı noktadan-noktaya ise, sonlandırma ayarlarının ayarlanmasına izin verdikleri için konektörün A1 ve B1 (A2 ve B2 değil) terminallerini kullanmak gereklidir (örneğin 6ES7 972 -- 0BA40 -- 0x40 DP konektör tipi ile).

**⚠ DİKKAT**

**Güç vermeden önce sürücü kapaklarını uygun biçimde değiştiriniz**

Sürücü kapaklarının, üniteye güç vermeden önce uygun biçimde değiştirildiğinden emin olunuz.

Sürücü devrede sonlandırma düğümü olarak yapılandırılırsa, sonlandırma ve besleme dirençleri de uygun terminal bağlantılarına kablayı yapılmalıdır. Bu diyagram sonlandırma ve besleme için gerekli MM4 sürücü bağlantılarıyla ilgili örnekleri gösterir.



**MM4 sürücüyü ayarlama**

S7-1200 için bir sürücü bağlamadan önce, sürücünün aşağıdaki sistem parametrelerine sahip olduğundan emin olmalısınız. Parametreleri ayarlamak için sürücü üzerindeki tuş takımını kullanınız:


1. Fabrika ayarlarına (isteğe bağlı) sürücüyü sıfırlayınız.	P0010=30 P0970=1
1. adımı atlarsanız, daha sonra bu parametreler belirtilen değerlere ayarlanır emin olunuz.	USS PZD uzunluk = P2012 İndeks 0=(2, 4, 6, veya 8) USS PKW uzunluk = P2013 İndeks 0=4
2. Tüm parametrelere (Uzman modu) okuma / yazma erişimini etkinleştiriniz.	P0003=3
3. Sürücünüzün Motor ayarlarını kontrol ediniz. Bu ayarlar kullanılan motor(lar) a göre değişir. P304, P305, P307, P310, P311 Parametrelerini ayarlamak için önce P010 parametrelerini 1 (hızlı devreye alma modu) olarak ayarlamamız gerekir. Parametreleri ayarlamayı tamamladığınızda P010 parametrelerini 0 olarak ayarlayınız. P304, P305, P307, P310 ve P311 Parametreleri sadece hızlı devreye alma modunda değiştirilebilir.	P0304 = Beyan motor gerilimi (V) P0305 = Beyan motor akımı (A) P0307 = Beyan motor gücü (W) P0310 = Beyan motor frekansı (Hz) P0311 = Beyan motor hızı
4. Lokal / uzaktan kontrol modunu ayarlayınız.	P0700 İndeks 0=5
5. COM bağlantısı üstündeki USS frekans ayar noktasının seçimini yapınız.	P1000 İndeks 0=5
6. Hızlanma süresi (opsiyonel) Bu motorun maksimum frekansa hızlanması için gereken saniye cinsinden süredir.	P1120=(0 ila 650.00)
7. Yavaşlama süresi (opsiyonel) Bu motorun tam durana kadar yavaşlaması için gereken saniye cinsinden süredir.	P1121=(0 ila 650.00)
8. Seri bağlantı referans frekansını ayarlayınız:	P2000=(1 ila 650 Hz)
9. USS normalleşmeyi ayarlayınız:	P2009 İndeks 0=0
10. RS485 ara yüzünün baud hızını ayarlayınız:	P2010 İndeks 0= 4 (2400 baud) 5 (4800 baud) 6 (9600 baud) 7 (19200 baud) 8 (38400 baud) 9 (57600 baud) 12 (115200 baud)
11. Slave adresini giriniz. Her sürücü (maksimum 31) veri yolu üzerinden kumanda edilebilir.	P2011 İndeks 0=(0 ila 31)
12. Seri bağlantı zaman aşımı ayarlayınız. Bu, iki adet gelen veri telegramları arasındaki izin verilen maksimum periyottur. Bu özellik, bir haberleşme kesintisi durumunda invertörü kapatmak için kullanılır. Geçerli bir veri telegramı alındıktan sonra zamanlama başlar. Bir başka veri telegram, belirtilen süre içinde alınmaz ise, invertör açma yapacak ve F0070 hata kodunu gösterecektir. Değeri sıfır olarak ayarlama kontrol cihazını kapatır.	P2014 İndeks 0=(0 ila 65,535 ms) 0=zaman aşımı devre dışı bırakıldı
13. RAM 'den EEPROM veri aktarımı:	P0971=1 (Transferi başlat) EEPROM parametre ayarlarında değişiklikleri kaydediniz

## 12.5 Modbus haberleşme

### 12.5.1 Modbus RTU ve TCP haberleşme Modbus TCP komutları V13'e Genel Bakış

#### Modbus fonksiyon kodları

- Modbus RTU master (veya Modbus TCP istemci) olarak çalışan bir CPU uzak bir Modbus RTU slave (veya Modbus TCP sunucusu) deki veriyi ve I / O durumlarını okuyabilir/yazabilir. Uzak veri, okunabilir ve sonra program mantığı içinde işlenebilir.
- Modbus RTU slave (veya Modbus TCP sunucusu) olarak çalışan bir CPU, bir denetim cihazının, CPU bellekteki veriyi ve I / O durumlarını okuyup yazmasına imkan sağlar. Bir RTU master (veya Modbus TCP istemci) program mantığınız için kullanılabilir olan slaver/sunucu CPU belleğe yeni değerler yazabilir.

 <b>UYARI</b>
<b>Bir saldırgan ağlarınıza fiziksel olarak erişebilirse, saldırgan muhtemelen verileri okuyup yazabilir.</b>
(GET / PUT kullanarak HMI'ler hariç) TIA Portal, CPU ve HMI'ler tekrar oynatmaya ve "man-in-the-middle (iki bağlantı noktası arasındaki bağlantıyı izinsiz izleme)" saldırılarına karşı koruyan güvenli haberleşme kullanır. Haberleşme etkinleştirildiğinde, imzalı mesajların değişimi açık metinde yer alır bu da bir saldırganın veri okumasına izin verir, ancak verilerin yetkisiz yazılmasına karşı korur. Haberleşme süreci değil TIA Portal, know-how korunan blok verilerini şifreler.
Diğer tüm haberleşme biçimlerinin (PROFIBUS, PROFINET, AS-i, ya da diğer I/O bus, GET/PUT, T-Blok ve haberleşme modülleri (CM) üzerinden I/O değişimi) hiçbir güvenlik özelliği yoktur. Fiziksel erişimi kısıtlayarak bu haberleşme biçimlerini korumak zorundasınız. Bir saldırgan bu haberleşme biçimlerini kullanarak fiziksel olarak ağlarınıza erişebilirse, saldırgan muhtemelen veri okuyup yazabilir.
Güvenlik bilgisi ve öneriler için Siemens Servis ve destek sitesindeki ( <a href="http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf">http://www.industry.siemens.com/topics/global/en/industrial-security/Documents/operational_guidelines_industrial_security_en.pdf</a> ) "Sanayi Güvenliği ile ilgili operasyonel Kurallar" a bakılmalıdır.

Çizelge 12- 45 Veri fonksiyonlarını oku: uzaktan I/O ve program verisi okuma

Modbus fonksiyon kodu	slave (sunucu) fonksiyonlarını oku - standart adresleme
01	Çıkış bitlerini oku: istek başına 1 ila 2000 bit
02	Giriş bitlerini oku: istek başına 1 ila 2000 bit
03	Tutma kayıtlarını oku: istek başına 1 ila 125 word
04	Giriş word'larını oku: istek başına 1 ila 125 word

Çizelge 12- 46 Veri fonksiyonları yazınız: uzak I/O yaz ve program verilerini değiştirmek

Modbus fonksiyon kodu	slave (sunucu) fonksiyonlarını yaz - standart adresleme
05	Bir çıkış biti yaz: istek başına 1 bit
06	Bir tutma kaydı yaz: istek başına 1 word
15	Bir veya daha fazla çıkış biti yaz: istek başına 1 ila 1968 bit
16	Bir veya daha fazla tutma kaydı yaz: istek başına 1 ila 123 word

- Modbus fonksiyon kodları 08 ve 11 slave cihazı haberleşme tanılama bilgisi sağlar.
- Modbus fonksiyon kodu 0 (köle tepki ile) tüm köleler bir mesaj yayınlar. Haberleşme tabanlı bir bağlantı olduğundan yayın fonksiyonu, Modbus TCP için geçerli değildir.

Çizelge 12- 47 Modbus ağ istasyonu adresleri

İstasyon		Adres
RTU istasyon	Standart istasyon adresi	1 ila 247
	Genişletilmiş istasyon adresi	1 ila 65535
TCP istasyon	İstasyon adresi	IP adresi ve port numarası

### Modbus bellek adresleri

Mevcut Modbus bellek adreslerinin gerçek sayısı CPU modeline, ne kadar bellek bulunduğu ve ne kadar CPU belleğinin diğer program verileri tarafından kullanılıyor olduğuna bağlıdır. Aşağıdaki Çizelge, adres aralığının nominal değerini verir.

Çizelge 12- 48 Modbus bellek adresleri

İstasyon		Adres aralığı
RTU istasyon	Standart bellek adresi	10K
	Genişletilmiş bellek adresi	64K
TCP istasyon	Standart bellek adresi	10K

### Modbus RTU haberleşme

Modbus RTU (Uzak Terminal Birimi) Modbus ağ cihazları arasında seri veri transferi için RS232 veya RS485 elektrik bağlantısını kullanan standart bir ağ haberleşme protokolüdür. Bir RS232 veya RS485 CM veya bir RS485 CB ile bir CPU' ya PTP (Noktadan Noktaya) ağ portları ekleyebilirsiniz.

Modbus RTU, tüm haberleşmelerin Tek Master cihazı tarafından başlatıldığı bir master / slave ağı kullanır ve slave sadece bir master'ın isteğine yanıt verebilir. Master bir slave adresine bir istek gönderir ve sadece o slave adresi komuta yanıt verir.



## Modbus TCP haberleşme

Modbus TCP (iletim kontrol protokolü), TCP / IP haberleşme için CPU üzerindeki PROFINET konektörünü kullanan standart bir ağ haberleşme protokolüdür. Hiçbir ek haberleşme donanım modülü gerekli değildir.

Modbus TCP, Modbus haberleşme yolu olarak Açık Kullanıcı Haberleşme (OUC) bağlantıları kullanır. STEP 7 ve CPU arasındaki bağlantıya ek olarak çoklu istemci-sunucu bağlantıları mevcut olabilir. Karışık istemci ve sunucu bağlantıları, CPU modeli (Sayfa 511) tarafından izin verilen maksimum bağlantı sayısına kadar desteklenir.

Her MB\_SERVER bağlantısı, benzersiz bir kopya DB ve IP port numarasını kullanmalıdır. IP port başına sadece 1 bağlantı desteklenir. Her MB\_SERVER (benzersiz kopya DB ve IP port ile) her bağlantı için ayrı ayrı yürütülmelidir.

Bir Modbus TCP istemcisi (master), DISCONNECT parametreleri ile istemci-sunucu bağlantısını kontrol etmelidir. Temel Modbus istemci faaliyetleri aşağıda gösterilmiştir.

30.Özel bir sunucu (slave) IP adresi ve IP port numarası için bir bağlantı başlatınız

31.Modbus mesajlarının istemci iletimini başlatın ve sunucu yanıtlarını alınız

32.İstedığınız zaman, Farklı bir sunucu ile bağlantıyı sağlamak için istemci ve sunucu arasındaki bağlantı kesmesini başlatınız.

## Programınızdaki Modbus RTU komutları

- MB\_COMM\_LOAD: Bir MB\_COMM\_LOAD yürütmesi baud hızı, parite ve akış kontrolü gibi PtP port parametrelerini ayarlamak için kullanılır. CPU port, Modbus protokolü için yapılandırıldıktan sonra, sadece MB\_MASTER veya MB\_SLAVE komutlarından biri tarafından kullanılabilir.
- MB\_MASTER: Modbus master komutu CPU nun Modbus RTU master cihaz olarak davranmasını ve bir veya birden fazla Modbus slave cihazlar ile iletişim kurmasını sağlar.
- MB\_SLAVE: Modbus slave komutu CPU nun Modbus RTU slave cihaz olarak davranmasını ve bir Modbus slave cihaz ile iletişim kurmasını sağlar.

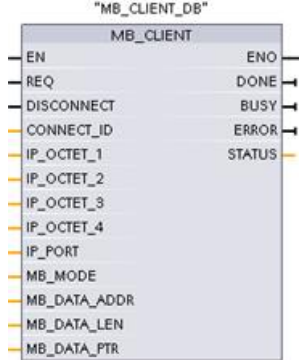
## Programınızdaki Modbus TCP komutları

- MB\_CLIENT: Sunucu-istemci TCP bağlantısını yapar, komut mesajı gönderir, yanıt alır ve sunucudan bağlantı kopmasını kontrol eder.
- MB\_SERVER: İstek üzerine bir Modbus TCP istemciye bağlantı sağlar, Modbus mesajını alır ve yanıt verir.

## 12.5.2 Modbus TCP

### 12.5.2.1 MB\_CLIENT (PROFINET aracılığıyla Modbus TCP istemci olarak haberleşme) komutu

Çizelge 12- 49 MB\_CLIENT komutu

LAD / FBD	SCL	Açıklama
 <p>"MB_CLIENT_DB" MB_CLIENT</p> <p>EN, ENO, REQ, DONE, DISCONNECT, BUSY, CONNECT_ID, ERROR, IP_OCTET_1, STATUS, IP_OCTET_2, IP_OCTET_3, IP_OCTET_4, IP_PORT, MB_MODE, MB_DATA_ADDR, MB_DATA_LEN, MB_DATA_PTR</p>	<pre>"MB_CLIENT_DB" (   REQ:= bool_in_,   DISCONNECT:= bool_in_,   CONNECT_ID:= uint_in_,   IP_OCTET_1:= byte_in_,   IP_OCTET_2:= byte_in_,   IP_OCTET_3:= byte_in_,   IP_OCTET_4:= byte_in_,   IP_PORT:= uint_in_,   MB_MODE:= usint_in_,   MB_DATA_ADDR:= udint_in_,   MB_DATA_LEN:= uint_in_,   DONE=&gt; bool_out_,   BUSY=&gt; bool_out_,   ERROR=&gt; bool_out_,   STATUS=&gt; word_out_,   MB_DATA_PTR:= variant_inout );</pre>	<p>MB_CLIENT, S7-1200 CPU üzerinde PROFINET konektör ile Modbus TCP istemcisi olarak iletişim kurar. Hiçbir ek haberleşme donanım modülü gerekli değildir.</p> <p>MB_CLIENT, bir istemci-sunucu bağlantısı yapar, bir Modbus fonksiyon isteği gönderir, bir yanıt alır ve bir Modbus TCP sunucudan bağlantı kopukluğunu kontrol edebilir.</p>

Çizelge 12- 50 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama
REQ	In	Bool FALSE = Modbus haberleşme isteği yok TRUE = Bir Modbus TCP sunucusu ile iletişim kurma isteği
DISCONNECT	IN	Bool DISCONNECT parametreleri, programınızın Modbus sunucu cihazla bağlantısını ve kopukluğu kontrol etmenizi sağlar. DISCONNECT= 0 ise ve bir bağlantı yoksa o zaman MB_CLIENT, atanan IP adresi ve port numarası ile bağlantı kurmaya çalışır. DISCONNECT= 1 ise ve bir bağlantı varsa, o zaman bir kesme işlemi denir. Bu giriş etkin olduğunda, başka hiçbir işlem denemeyecektir.
CONNECT_ID	IN	UInt CONNECT_ID parametrelerinin, PLC içindeki her bağlantıyı benzersiz olarak belirlemesi gerekir. MB_CLIENT veya MB_SERVER komutunun her benzersiz kopyası, benzersiz CONNECT_ID parametre içermelidir.
IP_OCTET_1	IN	USInt Modbus TCP sunucu IP adresi: Octet 1 İstemcinin bağlantı yapacağı ve Modbus TCP protokolü kullanarak iletişim sağlayacağı Modbus TCP sunucunun 32-bitlik IPv4 IP adresinin 8-bitlik bölümü.
IP_OCTET_2	IN	USInt Modbus TCP sunucu IP adresi: Octet 2
IP_OCTET_3	IN	USInt Modbus TCP sunucu IP adresi: Octet 3
IP_OCTET_4	IN	USInt Modbus TCP sunucu IP adresi: Octet 4

Parametre ve tipi		Veri tipi	Açıklama
IP_PORT	IN	UInt	Varsayılan değer = 502: İstemci bağlantı kurmaya çalışacağı ve sonuçta TCP / IP protokolünü kullanarak iletişim kuracağı sunucunun IP port numarası.
MB_MODE	IN	USInt	Mod Seçimi: İstek türünü (okuma, yazma veya tanılama) atar. Ayrıntılar için aşağıdaki Modbus fonksiyonları çizelgesine bakınız.
MB_DATA_ADDR	IN	UDInt	Modbus Başlangıç Adresi: MB_CLIENT tarafından erişilecek verilerin başlangıç adresini atar. Geçerli adresler için aşağıdaki Modbus fonksiyonları çizelgesine bakınız.
MB_DATA_LEN	IN	UInt	Modbus veri uzunluğu: Bu istekte erişilebilir bit ya da word sayısını atar. Geçerli uzunluklar için aşağıdaki Modbus fonksiyonları çizelgesine bakınız
MB_DATA_PTR	IN_OUT	Variant	Modbus veri kaydı işaretçisi: Yazmaç, bir Modbus sunucudan gelen veya giden veriyi saklar. İşaretçi, standart bir küresel DB veya bir M bellek adresi atamalıdır.
DONE	OUT	Bool	DONE biti bir tarama için TRUE olur, son istek hatasız tamamlandıktan sonra.
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>0 - MB_CLIENT işlemi devam etmiyor</li> <li>1 - MB_CLIENT işlemi devam ediyor</li> </ul>
ERROR	OUT	Bool	MB_CLIENT yürütmesi, bir hata ile sona erdirildikten sonra ERROR biti, bir tarama için TRUE olur. STATUS parametrelerindeki hata kodu değeri yalnızca ERROR = TRUE olan tek çevrim esnasında geçerli olur
STATUS	OUT	Word	Yürütme durum kodu

## REQ parametreleri

FALSE = Modbus haberleşme isteği yok

TRUE = Bir Modbus TCP sunucu ile haberleşme isteği

Hiçbir kopya MB\_CLIENT aktif değil ve parametre DISCONNECT = 0 ise, REQ = 1 olduğunda yeni Modbus isteği başlayacaktır. Eğer bağlantı zaten kurulmuş değilse, yeni bir bağlantı yapılacaktır.

MB\_CLIENT aynı kopyası DISCONNECT = 0 ve REQ= 1 ile tekrar çalıştırılırsa geçerli isteğin tamamlanmasından önce, sonraki Modbus iletimi yapılmayacaktır. Ancak mevcut istek tamamlanır tamamlanmaz MB\_CLIENT REQ= 1 ile çalıştırılırsa yeni bir istek işlenebilir.

Geçerli MB\_CLIENT haberleşme isteği tamamlandığında, DONE biti, bir çevrim için geçerlidir. DONE biti, çoklu MB\_CLIENT isteklerini sıralamak için bir zaman kapısı olarak kullanılabilir.

#### Not

#### MB\_CLIENT prosesi sırasında giriş veri tutarlılığı

Bir Modbus istemci Modbus işlemini bir kere başlattığında, tüm giriş durumları dahili olarak kaydedilir ve daha sonra her bir ardışık çağrı ile karşılaştırılır. Bu özel çağrının, aktif istemci isteğinin yaratıcısı olup olmadığını belirlemek için karşılaştırma kullanılır. Birden fazla MB\_CLIENT çağrısı, ortak bir kopya DB kullanılarak yapılabilir.

Bunun sonucu olarak, bu girişler bir MB\_CLIENT işleminin aktif olarak işlendiği zaman periyodu boyunca değiştirilmemesi önemlidir. Bu kurala uyulmazsa, bir MB\_CLIENT onun aktif kopya olduğunu belirleyemez.

### MB\_MODE ve MB\_DATA\_ADDR parametreleri Modbus haberleşme fonksiyonunu seçer

MB\_DATA\_ADDR, erişilecek verilerin başlangıç Modbus adresini atar. MB\_CLIENT komutu bir fonksiyon kod girişini yerine MB\_MODE girişini kullanır.

MB\_MODE ve MB\_DATA\_ADDR değerlerinin kombinasyonu, gerçek Modbus mesajında kullanılan fonksiyon kodunu belirler. Aşağıdaki çizelgede MB\_MODE parametreleri, Modbus fonksiyonu ve Modbus adres aralığı arasındaki ilişkiyi gösterir.

Çizelge 12- 51 Modbus fonksiyonları

MB_MODE	Modbus fonksiyon	Veri uzunluğu	İşlem ve veri	MB_DATA_ADDR
0	01	1 ila 2000	Çıkış bitlerini oku: istek başına 1 ila 2000 bit	1 ila 9999
0	02	1 ila 2000	Giriş bitlerini oku: 1 ila 2000 bit, istek başına	10001 ila 19999
0	03	1 ila 125	Tutma kayıtlarını oku: istek başına 1 ila 125 word	40001 ila 49999 veya 400001 ila 465535
0	04	1 ila 125	Giriş word'larını oku: istek başına 1 ila 125 word	30001 ila 39999
1	05	1	Bir çıkış biti yazdır: istek başına bir bit	1 ila 9999
1	06	1	Bir tutma kaydı yazdır: istek başına 1 word	40001 ila 49999 veya 400001 ila 465535
1	15	2 ila 1968	Çoklu çıkış bitleri yazdır: istek başına 2 ila 1968 bit	1 ila 9999
1	16	2 ila 123	Çoklu tutma kaydı yazdır: istek başına 2 ila 123 word	40001 ila 49999 veya 400001 ila 465535
2	15	1 ila 1968	Bir veya daha fazla çıkış biti yazdır: istek başına 1 ila 1968 bit	1 ila 9999
2	16	1 ila 123	Bir veya daha fazla tutma kaydı yazdır: istek başına 1 ila 123 word	40001 ila 49999 veya 400001 ila 465535

MB_MODE	Modbus fonksiyon	Veri uzunluğu	İşlem ve veri	MB_DATA_ADDR
11	11	0	Sunucu haberleşme durum word ünü ve olay sayacını okur. Durum word (0-meşgul değil, 0xFFFF - meşgul) meşgul gösterir. Olay sayacı, bir mesajın her başarılı tamamlanmasında artırılır. MB_CLIENT'in MB_DATA_ADDR ve MB_DATA_LEN parametrelerinin her ikisi bu işlev için göz ardı edilir.	
80	08	1	Veri tanılama kodu 0x0000 kullanılarak, sunucu durumunu kontrol ediniz (Loopback testi - sunucu isteği yansıtır) istek başına 1 word	
81	08	1	Veri tanılama kodu 0x000a kullanılarak, sunucu olay sayacını sıfırlar istek başına 1 word	
3 ila 10, 12 ila 79, 82 ila 255			Rezerve edildi	

**Not**

**MB\_DATA\_PTR bir Modbus TCP sunucusundan okunan veya buraya yazılan verileri depolamak için bir arabellek atar**

Veri arabelleği standart bir küresel DB veya M bellek adresinde olabilir.

M bellekteki bir arabellek için, standart herhangi işaretçi biçimini kullanınız. Bu P # "Bit Adresi" "Veri tipi" "Uzunluk" biçimindedir örnek olarak P # M1000.0 WORD 500.

**MB\_DATA\_PTR haberleşme arabelleği atar**

- MB\_CLIENT haberleşme fonksiyonları:
  - Modbus sunucu adreslerinde 1-bitlik veri okuma ve yazma (00001 ila 09999)
  - Modbus sunucu adreslerinde 1-bitlik veri okuma (10001 ila 19999)
  - Modbus sunucu adreslerinde 16-bit word verilerini okuma (30.001-39.999) ve (40.001-49.999)
  - Modbus sunucu adreslerinde 16-bit word veri yazma (40001 ila 49999)
- Word veya bit büyüklüğündeki veriler, MB\_DATA\_PTR tarafından atanan DB veya M bellek arabelleğinde aktarılır.
- DB, MB\_DATA\_PTR tarafından arabellek olarak atanmış ise, o zaman tüm DB veri elemanları için veri tiplerini atamanız gerekir.
  - 1-bitlik Bool veri tipi, bir Modbus bit adresini temsil eder

- WORD, Uint ve Int gibi 16-bit tekli bir word veri tipleri, bir adet Modbus word adresini temsil eder
- DWORD, DInt ve Real gibi 32-bit çift word veri tipleri iki Modbus word adreslerini temsil eder
- Aşağıdaki gibi karmaşık DB elemanları, MB\_DATA\_PTR tarafından atanabilir
  - Standart diziler
  - Her elemanın benzersiz olduğu adlandırılmış yapılar.
  - Her elemanın benzersiz bir adı ve bir 16 veya 32 bitlik veri tipine sahip olduğu adlandırılmış karmaşık yapılar.
- MB\_DATA\_PTR veri alanlarının aynı küresel veri bloğunda (veya M bellek alanı) olması gibi bir gereklilik yoktur. Modbus okumalar için bir veri bloğu Modbus yazmalar için başka bir veri bloğu veya her MB\_CLIENT istasyonu için bir veri bloğu atayabilirsiniz.

### Çoklu istemci bağlantıları

Bir Modbus TCP istemci, PLC tarafından izin verilen Açık Kullanıcı haberleşme bağlantılarının maksimum sayısına kadar eşzamanlı bağlantıları destekler. Modbus TCP İstemciler ve Sunucular içeren bir PLC için bağlantıların toplam sayısı, desteklenen Açık Kullanıcı Haberleşme bağlantı sayısını (Sayfa 511) geçmemelidir. Modbus TCP bağlantıları istemci ve / veya Sunucu tipi bağlantıları arasında paylaşılabilir.

Bireysel istemci bağlantıları bu kurallara uymalıdır:

- Her MB\_CLIENT bağlantısı farklı bir kopya DB kullanmalıdır
- Her MB\_CLIENT bağlantısı benzersiz bir sunucu IP adresi belirtmelidir
- Her MB\_CLIENT bağlantısı benzersiz bir bağlantı ID'si belirtmelidir
- Benzersiz IP port numaraları, sunucu yapılandırmasına bağlı olarak gerekli olabilir veya olmayabilir

Bağlantı ID si her münferit bağlantı için benzersiz olmalıdır. Bu; tekli, eşsiz Bağlantı ID sinin yalnızca her bir münferit kopya DB ile kullanılması gerekir demektir. Özetle, kopya DB ve Bağlantı ID si birlikte eşleştirilmiş ve her bağlantı için benzersiz olmalıdır.

Çizelge 12- 52 MB\_CLIENT kopya veri bloğu kullanıcı erişimli statik değişkenler

Değişken	Veri tipi	Varsayılan	Açıklama
Blocked_Proc_Timeout	Real	3.0	ACTIVE olarak bu kopyayı kaldırmadan önce engellenen bir Modbus istemci kopyasının (Saniye cinsinden) beklenme zaman miktarı. Bu, örneğin, Bir istemci isteği verildiğinde oluşabilir ve tamamen isteği bitirmeden önce uygulama, istemci işlevini yürütmesini durdurur. Maksimum S7-1200 sınırı 55 saniyedir.
MB_Unit_ID	Word	255	Modbus birim tanımlayıcı: Bir Modbus TCP sunucusu, IP adresini kullanarak adreslenir. Bunun bir sonucu olarak, MB_UNIT_ID parametreleri Modbus TCP adresleme için kullanılmaz. MB_UNIT_ID parametreleri Modbus RTU protokolündeki slave adresine karşılık gelir. Bir Modbus TCP sunucu bir Modbus protokolüne doğru bir ağ geçidi için kullanılırsa, MB_UNIT_ID seri ağda bağlı slave cihazını tanımlamak için kullanılabilir. MB_UNIT_ID, doğru Modbus RTU slave adresine isteği iletmek için kullanılacaktır. Bazı Modbus TCP cihazlar, MB_UNIT_ID parametrelerinin kısıtlı aralık değerleri içinde başlatılması gerekebilir.
RCV_TIMEOUT	Real	2.0	MB_CLIENT bir sunucunun bir isteğe yanıt vermesi bekleyeceği saniye cinsinden süre.
Bağlı	Bool	0	1 = Bağlı 0 = bağlı değil: atanan sunucuya yapılan bağlantının bağlı veya bağlı değil olup olmadığını gösterir

Çizelge 12- 53 MB\_CLIENT protokol hataları

STATUS (W#16#)	Modbus istemciye yanıt kodu (B#16#)	Modbus protokol hataları
8381	01	Fonksiyon kodu desteklenmiyor
8382	03	Veri uzunluk hatası
8383	02	Veri adres hatası veya MB_HOLD_REG adres alanı sınırları dışında erişim
8384	03	Veri değer hatası
8385	03	Veri tanımlama kod değeri desteklenmiyor (Fonksiyon kodu 08)

Çizelge 12- 54 MB\_CLIENT yürütme durum kodları <sup>1</sup>

STATUS (W#16#)	MB_CLIENT parametre hataları
7001	MB_CLIENT, Modbus sunucunun atanan TCP port üzerinde bir bağlantı veya bağlantı kesme isteğine yanıt vermesini bekliyor. Bu, yalnızca bir bağlantı veya bağlantı kesme operasyonun ilk defa yürütülmesi için rapor edilir.
7002	MB_CLIENT, Modbus sunucunun atanan TCP port üzerinde bir bağlantı veya bağlantı kesme isteğine yanıt vermesini bekliyor. Bir bağlantı ya da bağlantı kesme işleminin tamamlanması için beklerken, bu, herhangi bir sonraki yürütmeler için rapor edilecektir.
7003	Bir bağlantı kesme işlemi başarıyla (Sadece geçerli bir PLC tarama için) tamamladı.

STATUS (W#16#)	MB_CLIENT parametre hataları
80C8	Sunucu, atanmış zaman içerisinde yanıt vermedi. MB_CLIENT, atanan süre içinde başlangıçta transfer edilen işlem ID sini kullanarak bir yanıtı alması gerekir veya bu hata döndürülür. Modbus sunucu cihazına olan bağlantıyı kontrol ediniz. Bu hata sadece yapılandırılmış tüm yeniden denemeler (uygulanabilirse) yapıldıktan sonra rapor edilir.
8188	Geçersiz mod değeri
8189	Geçersiz veri adres değeri
818A	Geçersiz veri uzunluk değeri
818B	DATA_PTR alanına geçersiz işaretçi. Bu MB_DATA_ADDRESS + MB_DATA_LEN'in kombinasyonu olabilir.
818C	Optimize edilmiş DATA_PTR alanına (standart DB alanı veya M bellek alanı olmalıdır) geçersiz işaretçi
8200	Port, mevcut Modbus isteğini işlemekle meşgul.
8380	Alınan Modbus çerçevesi, yanlış biçimlenmiş veya çok az bayt alınmıştır.
8387	Atanan Bağlantı ID parametreleri önceki istekler için kullanılan ID den farklıdır. Sadece her bir MB_CLIENT kopya DB içinde kullanılan tekli bağlantı ID si olabilir. Bir sunucudan alınan Modbus TCP protokolü ID si 0 değilse, bu da bir iç hata olarak kullanılır.
8388	Bir Modbus sunucu, istenenden farklı bir veri miktarı geri döndürdü. Bu, Modbus fonksiyonu 15 ya da 16 için geçerlidir.

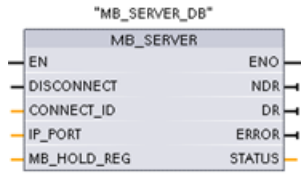
<sup>1</sup> Yukarıda listelenen MB\_CLIENT hatalarına ek olarak, temel T blok haberleşme komutlarından (TCON, TDISCON, TSEND ve TRCV) hatalar döndürülebilir.

### Ayrıca bakınız

TCON, TDISCON, TSEND, ve TRCV (TCP haberleşme) komutları (Sayfa 527)

### 12.5.2.2 MB\_SERVER (Modbus TCP sunucu olarak PROFINET aracılığıyla haberleşme) komutu

Çizelge 12- 55 MB\_SERVER komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a block named "MB_SERVER" with inputs: EN, DISCONNECT, CONNECT_ID, IP_PORT, and MB_HOLD_REG. It has outputs: ENO, NDR, DR, ERROR, and STATUS.</p>	<pre>"MB_SERVER_DB" (     DISCONNECT:=_bool_in_,     CONNECT_ID:=_uint_in_,     IP_PORT:=_uint_in_,     NDR=&gt;_bool_out_,     DR=&gt;_bool_out_,     ERROR=&gt;_bool_out_,     STATUS=&gt;_word_out_,     MB_HOLD_REG:=_variant_inout_);</pre>	<p>MB_SERVER, S7-1200 CPU üzerinde PROFINET konektör ile Modbus TCP sunucusu olarak iletişim kurar. Hiçbir ek haberleşme donanım modülü gerekli değildir.</p> <p>MB_SERVER, bir Modbus TCP istemciyle bağlantı isteğini kabul edebilir, bir Modbus fonksiyon isteği alır, bir yanıt mesajı gönderir.</p>



Çizelge 12- 56 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
DISCONNECT	IN	Bool	MB_SERVER bir partner cihaz ile pasif" bağlantı kurmaya çalışır. Bu, sunucusunun herhangi bir istek IP adresinden gelen bir TCP bağlantı isteğini pasif olarak dinlemesi demektir. DISCONNECT= 0 ise ve bir bağlantı yoksa o zaman pasif bir bağlantı başlatılır. DISCONNECT= 1 ise ve bir bağlantı varsa, o zaman bir kesme işlemi başlatılır. Bu programınızın bağlantı Kabul edildiğinde kontrol etmesine izin verir. Bu giriş etkin olduğunda, başka hiçbir işlem denenmeyecektir.
CONNECT_ID	IN	UInt	CONNECT_ID parametrelerinin, PLC içindeki her bağlantıyı benzersiz olarak belirler. MB_CLIENT veya MB_SERVER komutunun her benzersiz kopyası, benzersiz CONNECT_ID parametrelerini içermelidir.
IP_PORT	IN	UInt	Varsayılan değer = 502: Modbus İstemciden bağlantı isteği için izlenecek IP port numarasını belirler. Bu TCP port numaraları bir MB_SERVER pasif bağlantı için izin verilmez: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 ve 34964.
MB_HOLD_REG	IN_OUT	Variant	Modbus veri tutma yazmacı işaretçisi: Tutma yazmacı, standart bir küresel DB veya bir M bellek adresi olmalıdır. Bu bellek alanı Modbus yazmaç fonksiyonları 3 (okuma), 6 (yazma) ve 16 (yazma) kullanarak bir Modbus istemcinin erişmesine izin verilen değerleri tutmak için kullanılır.
NDR	OUT	Bool	Yeni Veri Hazır: 0 = Yeni veri yok, 1 = Yeni verilerin bir Modbus istemci tarafından yazılmış olduğunu gösterir
DR	OUT	Bool	Veri okuma: 0 = Okunacak veri yok, 1 = Verinin Modbus istemci tarafından okunmuş olduğunu gösterir.
ERROR	OUT	Bool	MB_SERVER yürütmesi, bir hata ile sona erdirildikten sonra ERROR biti, bir tarama için TRUE olur. STATUS parametrelerindeki hata kodu değeri yalnızca ERROR =.TRUE olan tek çevrim esnasında geçerli olur.
STATUS	OUT	Word	Yürütme durum kodu

MB\_SERVER, gelen Modbus fonksiyon kodlarının (1, 2, 4, 5 ve 15) S7-1200 CPU nun giriş proses görüntü ve çıkış proses görüntüsünde doğrudan bit ve Word'ları okumasına veya yazmasına izin verir. Veri aktarım fonksiyonu kodları (3, 6 ve 16) için, MB\_HOLD\_REG parametreleri, bir bayttan daha büyük bir veri tipi olarak tanımlanmalıdır. Aşağıdaki çizelgede CPU'daki proses görüntüsü için Modbus adreslerinin haritalanması gösterilmiştir.

Çizelge 12- 57 Proses görüntü için Modbus adreslerinin haritalama

Modbus fonksiyonları						S7-1200	
Kodlar	Fonksiyon	Veri alanı	Adres aralığı			Veri alanı	CPU adresi
01	Bitleri oku	Çıkış	1	ila	8192	Çıkış Proses Görüntüsü	Q0.0 ila Q1023.7
02	Bitleri oku	Giriş	10001	ila	18192	Giriş Proses Görüntüsü	I0.0 ila I1023.7
04	Word'leri oku	Giriş	30001	ila	30512	Giriş Proses Görüntüsü	IW0 ila IW1022
05	Bit yaz	Çıkış	1	ila	8192	Çıkış Proses Görüntüsü	Q0.0 ila Q1023.7
15	Bitleri yaz	Çıkış	1	ila	8192	Çıkış Proses Görüntüsü	Q0.0 ila Q1023.7

Bir M bellek adresi aralığı veya bir veri bloğu olabilen Modbus tutma yazmacı. Gelen Modbus mesajı fonksiyon kodları (3, 6 ve 16) word'ları okur veya yazar. Tutma yazmacının tipi MB\_HOLD\_REG parametreleri tarafından belirtilir.

#### Not

#### MB\_HOLD\_REG parametre ataması

Modbus Tutma Yazmacı, standart bir küresel DB veya bir M bellek adresinde olabilir.

M bellekteki A Modbus tutma yazmacı için, standart tüm İşaretçi biçimini kullanır. Bu, P # "Bit Adresi" "Veri tipi" "Uzunluk" biçimindedir. Bir örnek P # M1000.0 WORD 500 olabilir.

Aşağıdaki Çizelge Modbus fonksiyon kodları 03 (word okuma), 06 (word yazma) ve 16 (word yazma) için kullanılan tutma yazmacı haritalama için Modbus adresi örneklerini göstermektedir. DB adreslerinin gerçek üst sınırı, her bir CPU modeli için maksimum çalışma bellek sınırı ve M bellek sınırı tarafından belirlenir.

Çizelge 12- 58 CPU bellek adresi için Modbus adresinin haritalama örnekleri

Modbus Adresi	MB_HOLD_REG parametre örnekleri		
	P#M100.0 Word 5	P#DB10.DBx0.0 Word 5	"Tarif".içerik
40001	MW100	DB10.DBW0	" Tarif ". içerik [1]
40002	MW102	DB10.DBW2	" Tarif ". içerik [2]
40003	MW104	DB10.DBW4	" Tarif ". içerik [3]
40004	MW106	DB10.DBW6	" Tarif ". içerik [4]
40005	MW108	DB10.DBW8	" Tarif ". içerik [5]

#### Çoklu sunucu bağlantıları

Çoklu sunucu bağlantıları oluşturulabilir. Bu, tek bir PLC'nin çoklu Modbus TCP istemcileri için eşzamanlı bağlantı kurmasına izin verir.

Bir Modbus TCP sunucu, PLC tarafından izin verilen Açık Kullanıcı haberleşme bağlantılarının maksimum sayısına kadar eşzamanlı bağlantıları destekler. Modbus TCP İstemciler ve Sunucular içeren bir PLC için bağlantıların toplam sayısı, desteklenen Açık Kullanıcı Haberleşme bağlantı sayısını (Sayfa 511) geçmemelidir. Modbus TCP bağlantıları istemci ve / veya Sunucu tipi bağlantıları arasında paylaşılabilir.

Bireysel istemci bağlantısı bu kurallara uymalıdır:

- Her MB\_SERVER bağlantısı farklı bir kopya DB kullanmalıdır.
- Her MB\_SERVER bağlantısı benzersiz bir IP port adresi oluşturmalıdır. Port başına yalnızca 1 bağlantı desteklenir.
- Her MB\_SERVER bağlantısı benzersiz bir bağlantı ID'si kullanmalıdır.
- MB\_SERVER her bir bağlantı için münferit olarak çağrılmalıdır (ilgili kopya DB si ile birlikte).

Bağlantı ID si her münferit bağlantı için benzersiz olmalıdır. Bu; tekli, eşsiz Bağlantı ID sinin yalnızca her bir münferit kopya DB ile kullanılması gerektiği anlamına gelir. Özetle, kopya DB ve Bağlantı ID si birlikte eşleştirilmiştir ve her bağlantı için benzersiz olmalıdır.

Çizelge 12- 59 Modbus tanılama fonksiyon kodları

MB_SERVER Modbus tanılama fonksiyonları		
Kodlar	Alt-fonksiyon	Açıklama
08	0x0000	Sorgu verisi yansıma testini döndürür: MB_SERVER, alınan word veriyi Modbus istemciye yansıtır.
08	0x000A	Haberleşme olay sayacını temizler: MB_SERVER, Modbus işlevi 11 için kullanılan haberleşme olay sayacını temizleyecektir.
11		Haberleşme olay sayacını alınız: MB_SERVER, Modbus sunucuya gönderilen başarılı Modbus okuma ve yazma isteklerinin kaydetmek için bir iç haberleşme olay sayacı kullanır. Sayıcı herhangi Fonksiyon 8 veya Fonksiyon 11 isteklerinde artmaz. Aynı zamanda bir haberleşme hatasına neden olan herhangi bir istekte de artmaz. Yalnızca tek bir istemci-sunucu bağlantısı herhangi bir anda var olduğu için yayın fonksiyonu, Modbus TCP için kullanılabilir değildir.

## MB\_SERVER değişkenleri

Bu Çizelge programınızda kullanılacak MB\_SERVER kopya veri bloğunda saklanan yerel statik değişkenleri gösterir.

Çizelge 12- 60 MB\_SERVER genel statik değişkenler

Değişken	Veri tipi	Varsayılan değer	Açıklama
HR_Start_Offset	Word	0	Modbus Tutma Yazmacının başlangıç adresini atar
Request_Count	Word	0	Bu sunucu tarafından alınan tüm isteklerin sayısı.
Server_Message_Count	Word	0	Bu özel sunucu tarafından alınan isteklerin sayısı.
Xmt_Rcv_Count	Word	0	Bir hatayla karşılaşan iletimler veya alımların sayısı. Bir mesaj, geçersiz bir Modbus mesajı alırsa artırılır.
Exception_Count	Word	0	Döndürülmüş istisna gerektiren Modbus a özel hatalar
Success_Count	Word	0	Protokol hatası olmayan bu özel sunucu için alınan isteklerin sayısı.
Connected	Bool	0	Atanmış istemciyle olan bağlantının bağlı veya ayrı olup olmadığını belirtir: 1 = bağlı, 0 = ayrı

Programınız, HR\_Start\_Offset'e değerler yazabilir ve Modbus sunucu işlemlerini kontrol edebilir. Diğer değişkenler Modbus durumunu izlemek için okunabilir.

## HR\_Start\_Offset

Modbus tutma yazmacı adresleri 40001 'den başlar. Bu adresler tutma yazmacının başlangıcı PLC bellek adresine karşılık gelir. Ancak, bunun 40001 yerine başka değerdeki başlangıç Modbus tutma yazmacı adresini başlatmak için "HR\_Start\_Offset" değişkenini yapılandırabilirsiniz.

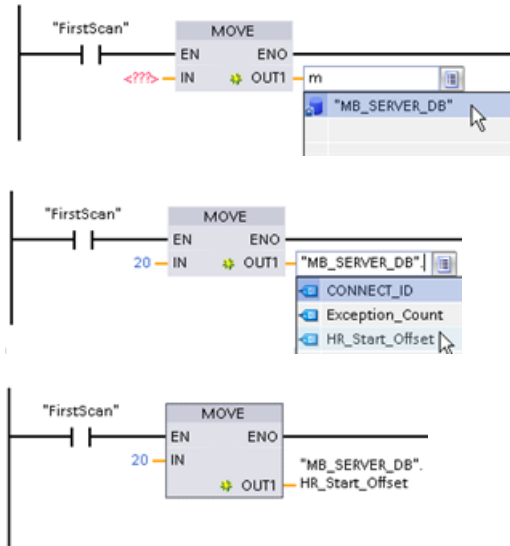
Örneğin, tutma yazmacı MW100 de başlatmak için yapılandırılırsa ve 100 word uzunluğundadır. 20 ofset, bir 40001 yerine 40021 başlangıç tutma yazmacı adresini belirtir. 40021 altında ve 40119 üstünde herhangi adres bir adresleme hatasına neden olur.

Çizelge 12- 61 Modbus tutma yazmacı adresleme örneği

HR_Start_Offset	Adres	Minimum	Maksimum
0	Modbus adresi (Word)	40001	40099
	S7-1200 adresi	MW100	MW298
20	Modbus adresi (Word)	40021	40119
	S7-1200 adresi	MW100	MW298

HR\_Start\_Offset Modbus tutma kaydının başlangıç adresini belirten bir word değeridir ve MB\_SERVER kopya veri bloğunda saklanır. MB\_SERVER programınıza yerleştirildikten sonra, parametre yardımcı açılan listesini kullanarak bu yerel statik değişken değerini ayarlayabilirsiniz.

Örneğin MB\_SERVER, LAD devresine yerleştirildikten sonra, bir önceki devreye gidebilir ve HR\_Start\_Offset değerini atayabilirsiniz. Değer, MB\_SERVER yürütmesinden önce atanmalıdır.



Varsayılan DB ismini kullanarak Modbus sunucu değişkeni girme:

6. Parametre alanına imleci getirin ve bir m karakteri yazınız.
7. DB adları açılan listeden "MB\_SERVER\_DB" seçeneğini seçiniz.
8. DB değişkenlerin açılan listeden "MB\_SERVER\_DB.HR\_Start\_Offset" seçeneğini seçiniz.

Çizelge 12- 62 MB\_SERVER yürütme durum kodları <sup>1</sup>

STATUS (W#16#)	Modbus sunucuya yanıt kodu (B#16#)	Modbus protokol hataları
7001		MB_SERVER, Modbus sunucunun atanmış TCP porta bağlantı yapmasını bekliyor. Bu, yalnızca bir bağlantı veya bağlantı kesme operasyonunun ilk defa yürütülmesi üzerine rapor edilir.
7002		MB_SERVER, Modbus sunucunun atanmış TCP porta bağlantı yapmasını bekliyor. Bir bağlantı ya da bağlantı kesme işleminin tamamlanması için beklerken, bu kod, herhangi bir sonraki yürütmeler için rapor edilecektir.
7003		Bir bağlantı kesme işlemi başarıyla (Sadece geçerli bir PLC tarama için) tamamladı.
8187		MB_HOLD_REG için Geçersiz işaretçi: alan çok küçük
818C		Optimize edilmiş MB_HOLD_REG alanına (standart DB alanı veya M bellek alanı olmalıdır) işaretçi veya Bloke edilmiş işlem zaman aşımı, 55 saniye sınırını aşıyor. (S7-1200 e özel)
8381	01	Desteklenmeyen fonksiyon kodu
8382	03	Veri uzunluk hatası
8383	02	Veri adres hatası veya MB_HOLD_REG adres alanının sınırları dışında erişim
8384	03	Veri değer hatası
8385	03	Desteklenmeyen veri tanımlama kodu değeri (fonksiyon kodu 08)

<sup>1</sup> Yukarıda listelenen MB\_SERVER hatalarına ek olarak, temel T blok haberleşme komutlarından (TCON, TDISCON, TSEND ve TRCV) hatalar döndürülebilir.

### Ayrıca bakınız

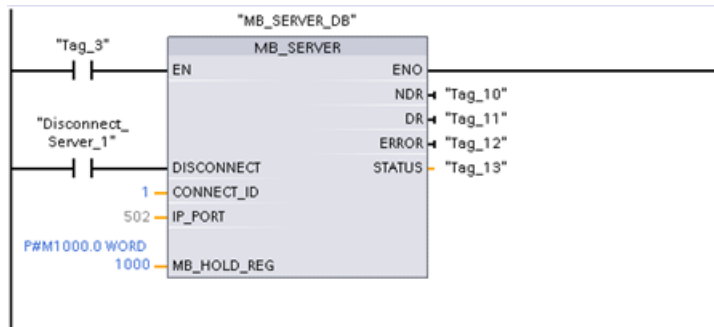
TCON, TDISCON, TSEND, ve TRCV (TCP haberleşme) komutları (Sayfa 527)

### 12.5.2.3 MB\_SERVER örnek: Çoklu TCP bağlantıları

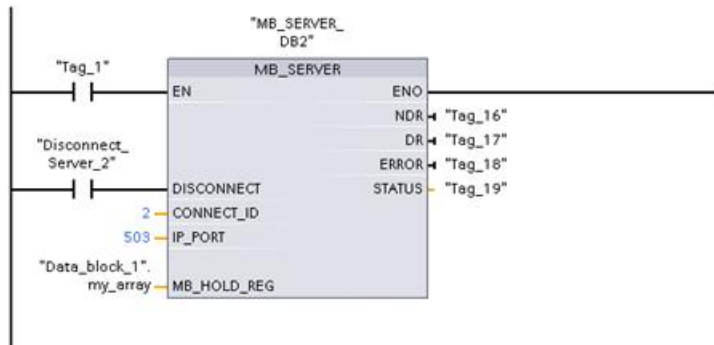
Çoklu Modbus TCP sunucu bağlantılarına sahip olabilirsiniz. Bunu gerçekleştirmek için, MB\_SERVER bağımsız bir şekilde, her bağlantı için yürütülmelidir. Her bağlantı bağımsız bir kopya DB, bağlantı ID si ve IP port kullanılmalıdır. S7-1200, IP port başına sadece bir bağlantı sağlar.

En iyi performans için MB\_SERVER her bağlantıya göre her program çevriminde yürütülmelidir.

**Devre 1:** Bağımsız IP\_PORT, bağlantı ID' si ve kopya DB ile bağlantı # 1



**Devre 2:** Bağımsız IP\_PORT, bağlantı ID'si ve kopya DB ile bağlantı # 2



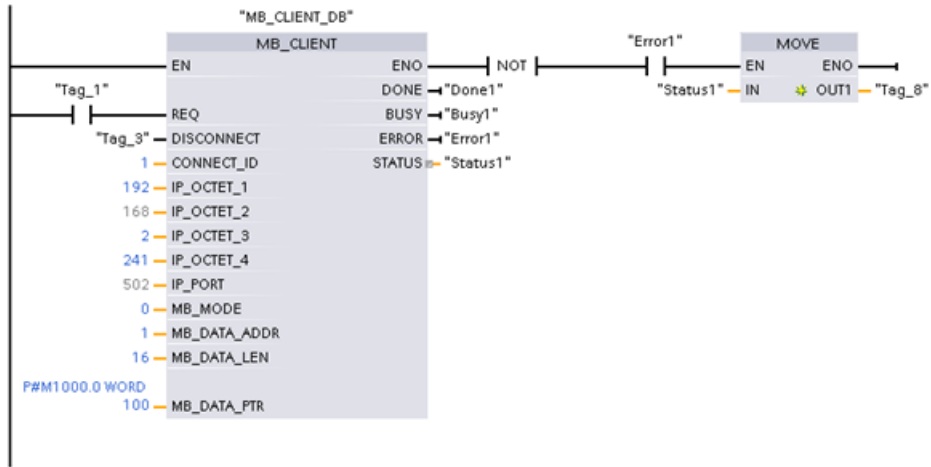
### 12.5.2.4 MB\_CLIENT örnek 1: Ortak TCP bağlantısı ile çoklu istek

Çoklu Modbus istemci istekleri aynı bağlantı üzerinden gönderilebilir. Bunu gerçekleştirmek için, aynı kopya DB, bağlantı ID si ve port numarasını kullanınız.

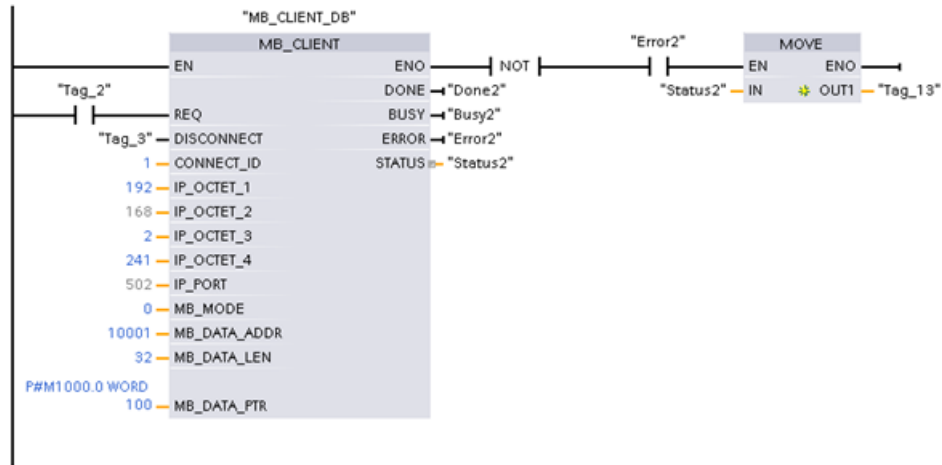
Sadece 1 istemci herhangi bir zamanda aktif olabilir. Bir istemci, yürütmesini tamamladığında, bir sonraki istemci yürütmesini başlatır. Programınız yürütme sırasından sorumludur.

Örnekte aynı bellek alanına yazan iki istemci gösterilmiştir. Ayrıca, döndürülen bir hata isteğe bağlı olarak yakalanır.

**Devre 1:** Modbus fonksiyon 1 - 16 çıkış görüntü bitlerini oku



**Devre 2:** Modbus fonksiyon 2 - 32 çıkış görüntü bitlerini oku



### 12.5.2.5 MB\_CLIENT örnek 2: farklı TCP bağlantıları ile çoklu istek

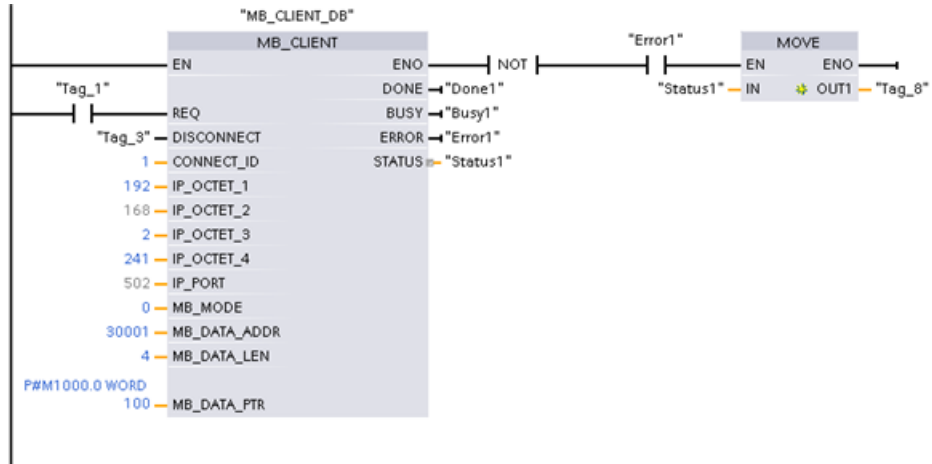
Modbus istemci istekleri farklı bağlantılar üzerinden gönderilebilir. Bunu gerçekleştirmek için, farklı kopya DB'ler, IP adresleri ve bağlantı ID'leri kullanılmalıdır.

Bağlantılar, aynı Modbus sunucuya kurulmuş ise port numarası farklı olmalıdır. Bağlantılar farklı sunucularda ise, hiçbir port numarası kısıtlaması yoktur.

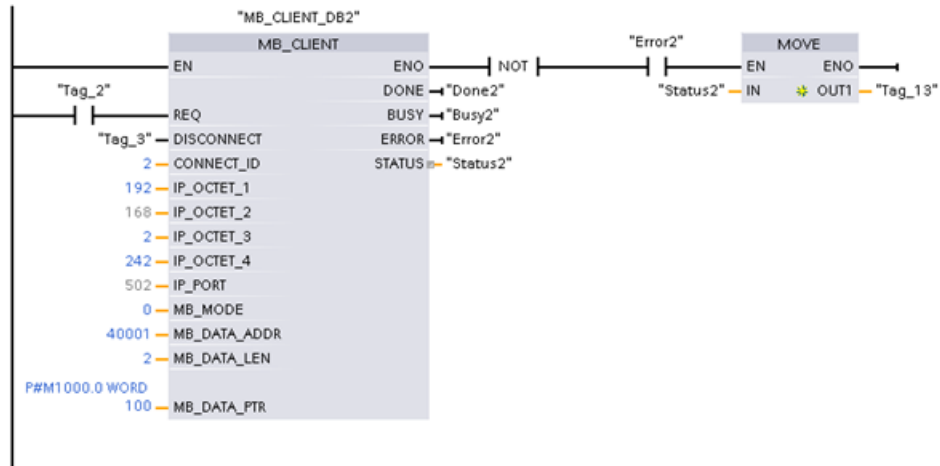
Örnekte aynı bellek alanına yazan iki istemci gösterilmiştir. Ayrıca, döndürülen bir hata isteğe bağlı olarak yakalanır.

#### Devre 1:

Modbus fonksiyon 4 - (S7-1200 belleğinde) giriş Word'lerini oku



**Devre 2:** Modbus fonksiyon 3 - Bir Modbus TCP sunucusundan tutma yazmacı Word'lerini oku.

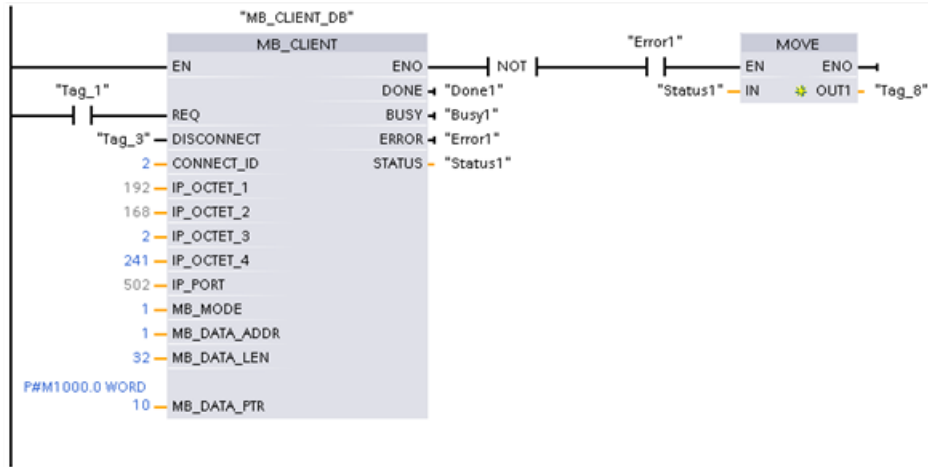


### 12.5.2.6 MB\_CLIENT örnek 3: Çıkış görüntü yazma isteği

Bu örnek, S7-1200 çıkış görüntüsünü yazmak için Modbus istemci isteğini gösterir.

**Devre 1:** Modbus fonksiyonu 15 - S7-1200 çıkış görüntü bitlerini yaz



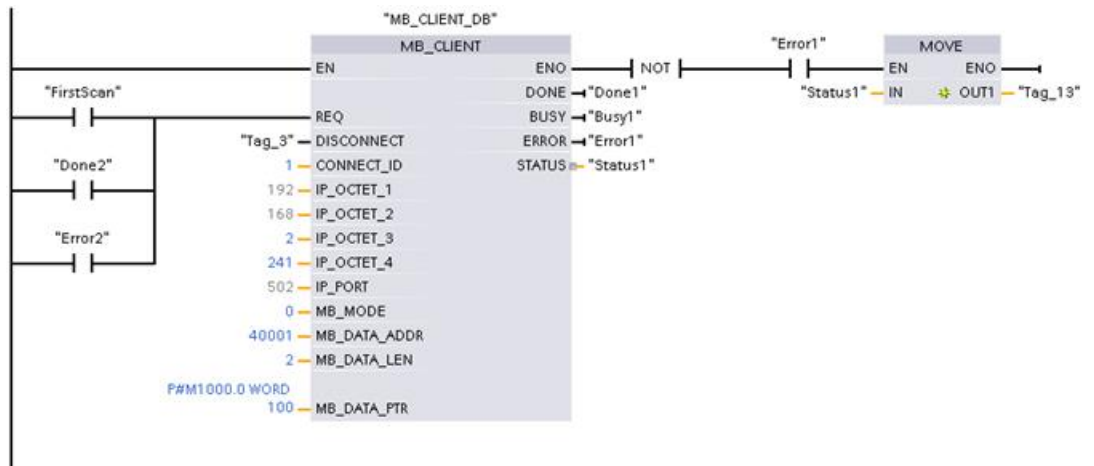


### 12.5.2.7 MB\_CLIENT örnek 4: Çoklu isteklerin koordinasyonu

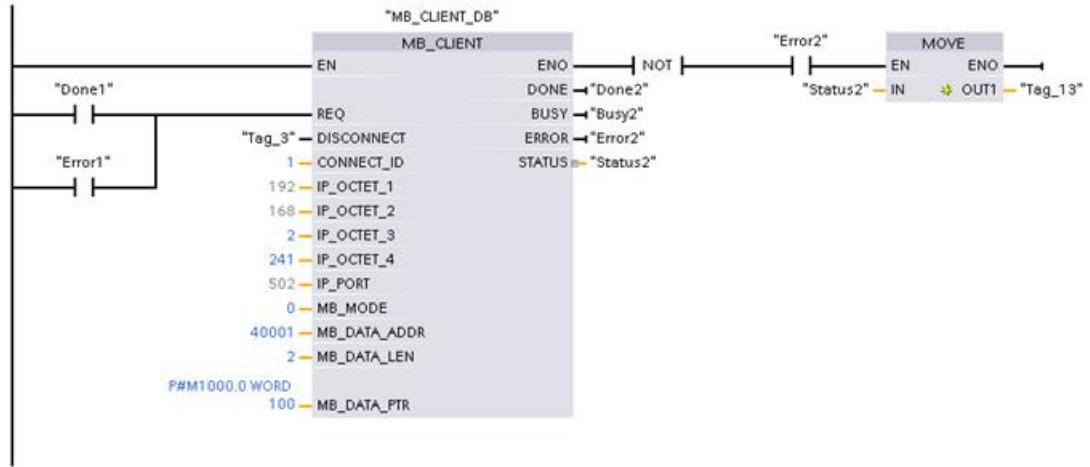
Her münferit Modbus TCP isteğinin yürütmesini tamamladığından emin olmalısınız. Bu koordinasyon, programınız tarafından sağlanmalıdır. Aşağıdaki örnek, birinci ve ikinci istemci isteklerinin çıkışlarının yürütmeyi koordine etmek için nasıl kullanılabileceğini göstermektedir.

Örnekte aynı bellek alanına yazan iki istemci gösterilmiştir. Ayrıca, döndürülen bir hata isteğe bağlı olarak yakalanır.

**Devre 1:** Modbus fonksiyon 3 – Tutma yazmaç Word'lerini oku



**Devre 2: Modbus fonksiyon 3 – Tutma yazmaç Word'lerini oku**



### 12.5.3 Modbus RTU

STEP 7 de mevcut Modbus RTU komutlarının iki versiyonu vardır:

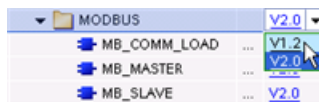
- Versiyon 1, STEP 7 Basic V10.5 başlangıçta mevcut idi.
- Versiyon 2, STEP 7 Basic / Profesyonel V11 de mevcuttur. Versiyon 2 tasarımı, MB\_COMM\_LOAD için REQ ve DONE parametrelerini ekler. Ayrıca, MB\_MASTER ve MB\_SLAVE için MB\_ADDR parametreleri şimdi genişletilmiş adresleme için bir UInt değerine izin verir.

Uyumluluk ve Geçiş kolaylığı için, hangi komut versiyonunu kullanıcı programınız içine ekleyeceğinizi seçebilirsiniz.

Aynı CPU programında 1.x ve 2.y komut versiyonlarının ikisini birden kullanmayınız. Programınızın Modbus komutları, aynı ana versiyon numarasına (1.x, 2.y veya V.z) sahip olması gerekir. Ana versiyon grubu içindeki bireysel komutlar farklı küçük versiyonlara sahip olabilir (1.x).



Komut ağacının başlıklarını ve sütunlarını etkinleştirmek için komut ağaç görev kartındaki simgeyi tıklayınız.



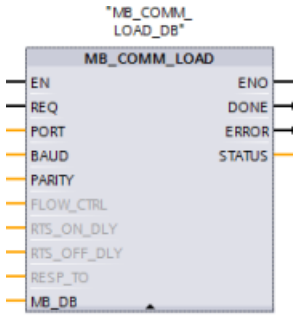
Modbus komutlarının versiyonunu değiştirmek için, açılan listeden versiyonu seçiniz. Siz grup veya bireysel komutları seçebilirsiniz.

Programınızda bir Modbus komutu yerleştirmek için komut ağacı kullandığınızda, proje ağacında yeni bir FB kopyası oluşturulur. PLC\_x> Program blokları> Sistem blokları> Program kaynakları altında proje ağacında yeni FB kopya görebilirsiniz.

Bir programda bir Modbus komut versiyonunun doğrulanması için program editöründe görüntülenen bir kutunun özelliklerini değil, proje ağaç özelliklerini incelemeniz gerekir. Bir proje ağacı Modbus FB kopyasını seçiniz, sağ tıklayınız ve "Özellikler" i seçiniz ve Modbus komut versiyon numarasını görmek için "Bilgi" sayfasını seçiniz.

### 12.5.3.1 MB\_COMM\_LOAD (Modbus RTU için PtP modülündeki portu yapılandırma) komutu

Çizelge 12- 63 MB\_COMM\_LOAD komutu

LAD / FBD	SCL	Açıklama
	<pre>"MB_COMM_LOAD_DB" (   REQ:=_bool_in_,   PORT:=_uint_in_,   BAUD:=_udint_in_,   PARITY:=_uint_in_,   FLOW_CTRL:=_uint_in_,   RTS_ON_DLY:=_uint_in_,   RTS_OFF_DLY:=_uint_in_,   RESP_TO:=_uint_in_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_,   MB_DB:=_fbtref_inout_);</pre>	<p>MB_COMM_LOAD komutu Modbus RTU protokolü haberleşmesi için bir PtP noktasını yapılandırır. Modbus port donanım seçenekleri: Üç CMS (RS485 veya RS232), artı bir CB ye (R4845) kadar yükleyiniz. Programınızda MB_COMM_LOAD komutunu yerleştirdiğinizde bir kopya veri bloğu otomatik olarak atanır.</p>

Çizelge 12- 64 Parametreler için veri tipleri

Parametre ve tipi	Veri tipi	Açıklama	
REQ	IN	Bool	Düşük-yüksek (pozitif kenar) sinyali çalışmayı başlatır. (Versiyon 2.0 sadece)
PORT	IN	Port	CM veya CB haberleşme cihazını yükleyip yapılandırdıktan sonra, port tanımlayıcı, PORT kutusu bağlantısında mevcut parametre yardımcı açılan listede görünür. Atanan CM veya CB port değeri cihaz yapılandırma özelliği "donanım tanımlayıcı" dır. Port sembolik ismi PLC etiket çizelgesinin "Sistem sabitleri" sekmesinde atanır.
BAUD	IN	UDInt	Baud hızı seçimi: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200, diğer tüm değerler geçersiz
PARITY	IN	UInt	Parite seçimi: <ul style="list-style-type: none"> <li>0 – hiç</li> <li>1 – tek</li> <li>2 – çift</li> </ul>
FLOW_CTRL <sup>1</sup>	IN	UInt	Akış kontrol seçimi: <ul style="list-style-type: none"> <li>0 – (varsayılan) akış kontrolü yok</li> <li>1 – Donanım akış kontrolü, RTS daima ON (RS485 portları için geçerli değildir)</li> <li>2 – Donanım akış kontrolü, RTS anahtarlmalı</li> </ul>
RTS_ON_DLY <sup>1</sup>	IN	UInt	RTS ON gecikme seçimi: <ul style="list-style-type: none"> <li>0 – (varsayılan) mesajın ilk karakteri iletilene kadar Aktif RTS den hiçbir gecikme yok</li> <li>1 ila 65535 – mesajın ilk karakteri (RS485 portu için geçerli değildir) iletilene kadar Aktif RTS den milisaniye olarak gecikme. RTS gecikmeleri FLOW_CTRL seçiminden bağımsız olarak uygulanmalıdır.</li> </ul>

Parametre ve tipi	Veri tipi	Açıklama
RTS_OFF_DLY <sup>1</sup>	IN	UInt RTS OFF gecikme seçimi: <ul style="list-style-type: none"> <li>0 – (varsayılan) RTS pasif olana kadar iletilen son karakterden hiçbir gecikme yok</li> <li>1 ila 65535 – RTS pasif olana kadar (RS485 portu için geçerli değildir) iletilen son karakterden milisaniye olarak gecikme. RTS gecikmeleri FLOW_CTRL seçiminin bağımsız olarak uygulanmalıdır.</li> </ul>
RESP_TO <sup>1</sup>	IN	UInt Tepki zaman aşımı: Slave in yanıt vermesi için MB_MASTER tarafından izin verilen milisaniye olarak zaman. Slave, bu süre içinde yanıt vermezse, MB_MASTER isteği yeniden denecek veya belirtilen sayıda deneme gönderildiğinde bir hata ile isteği sonlandırılacak. 5 ms ila 65535 ms (Varsayılan değer = 1000 ms).
MB_DB	IN	Variant MB_MASTER veya MB_SLAVE komutları tarafından kullanılan kopya veri bloğuna bir referans. MB_SLAVE veya MB_MASTER Programda yerleştirildikten sonra, DB tanımlayıcı MB_DB kutu bağlantısında mevcut parametre yardımcı açılan listesinde görünür.
DONE	OUT	Bool DONE biti Bir tarama için TRUE olur, son istek hatasız tamamlandıktan sonra. (Versiyon 2.0 yalnızca)
ERROR	OUT	Bool ERROR biti Bir tarama için TRUE olur, son istek bir hata ile sonlandırıldıktan sonra. STATUS parametrelerinde hata kodu değeri yalnızca ERROR = TRUE olan tek tarama sırasında geçerli olur.
STATUS	OUT	Word Yürütme durum kodu

<sup>1</sup> MB\_COMM\_LOAD (V 2.x veya sonrası) için Opsiyonel parametreler. Kutuyu genişletmek ve tüm parametreleri dahil etmek için LAD/FBD kutusunun altındaki oku tıklayınız.

MB\_COMM\_LOAD, bir portu yapılandırmak amacıyla Modbus protokolü için yürütülür. Bir port, Modbus protokolü için yapılandırıldığında, sadece MB\_MASTER veya MB\_SLAVE komutları tarafından kullanılabilir.

Bir MB\_COMM\_LOAD yürütmesi, Modbus haberleşme için kullanılan her bir haberleşme portunu yapılandırmak için kullanılmalıdır. Kullandığınız her port için benzersiz bir MB\_COMM\_LOAD kopya DB atayınız. CPU da, üç haberleşme modülü ( RS232 veya RS485 ) ve bir haberleşme kartına ( RS485 ) kadar montaj yapabilirsiniz. Bir başlatma OB sinden MB\_COMM\_LOAD u çağırın ve bunu bir kez çalıştırınız ya da bunu bir kez çalıştırmak için çağırın ve bunu bir kez çalıştırınız ya da bunu bir kez çalıştırmak için çağırın ve bunu bir kez çalıştırınız. Baud hızı veya değer eşitliği gibi haberleşme parametrelerini değiştirmek zorunda iseniz sadece MB\_COMM\_LOAD u tekrar çalıştırınız.

Programınıza bu komutları yerleştirdiğinizde MB\_MASTER veya MB\_SLAVE için bir kopya veri bloğu atanır. Bu kopya veri bloğu MB\_COMM\_LOAD komutu için MB\_DB parametrelerini belirtirken başvurulmaktadır.

## MB\_COMM\_LOAD veri bloğu değişkenleri

Aşağıdaki çizelgede, programınızda kullanılabilir MB\_COMM\_LOAD için kopya DB de saklanan yerel statik değişkenleri gösterilmiştir.

Çizelge 12- 65 Kopya DB de statik değişkenler

Değişken	Veri tipi	Açıklama
ICHAR_GAP	UInt	Karakterler arasında ara-karakter boşluğu için gecikme. Bu parametre, milisaniye cinsinden belirtilir ve alınan karakterler arasındaki beklenen zaman miktarını artırmak için kullanılır. Bu parametre için bit zamanlarının karşılık gelen sayısı, varsayılan 35 bit zamanlık (3.5 karakter zamanı) Modbus eklenir.
RETRIES	UInt	0x80C8 yanıt hata kodu yok döndürmeden önce master'ın yapmaya çalışacağı deneme sayısı.
STOP_BITS	USInt	Her bir karakteri çerçevelemede kullanılan stop bitlerinin sayısı. Geçerli değerler 1 ve 2.

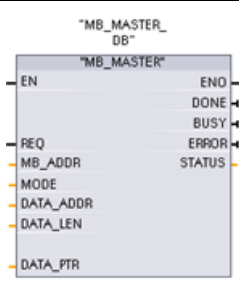
Çizelge 12- 66 MB\_COMM\_LOAD yürütme durum kodları <sup>1</sup>

STATUS (W#16#)	Açıklama
0000	Hata yok
8180	Geçersiz port ID değeri (haberleşme modülü için yanlış port / donanım tanımlayıcı)
8181	Geçersiz baud hızı değeri
8182	Geçersiz parite değeri
8183	Geçersiz akış kontrol değeri
8184	Geçersiz tepki zaman aşımı değeri (tepki zaman aşımı minimum 5 ms den küçük)
8185	MB_DB parametreleri bir MB_MASTER veya MB_SLAVE komutunun kopya veri bloğu değildir.

<sup>1</sup> Yukarıda listelenen USS komut hatalarına ek olarak, hatalar temel PtP haberleşme komutlarından döndürülebilir.

### 12.5.3.2 MB\_MASTER (Modbus master olarak PtP portu üzerinden haberleşme) komutu

Çizelge 12- 67 MB\_MASTER komutu

LAD / FBD	SCL	Açıklama
 <p>The diagram shows a ladder logic network for the MB_MASTER DB. It includes inputs: EN, MB_ADDR, MODE, DATA_ADDR, DATA_LEN, and DATA_PTR. It includes outputs: ENO, DONE, BUSY, ERROR, and STATUS.</p>	<pre>"MB_MASTER_DB" (   REQ:=_bool_in_,   MB_ADDR:=_uint_in_,   MODE:=_usint_in_,   DATA_ADDR:=_udint_in_,   DATA_LEN:=_uint_in_,   DONE=&gt;_bool_out_,   BUSY=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_,   DATA_PTR:=_variant_inout_);</pre>	MB_MASTER komutu MB_COMM_LOAD komutunun bir önceki yürütmesi tarafından yapılandırılmış bir portu kullanarak bir Modbus master olarak iletişim kurar. Programınızda MB_MASTER komutunu yerleştirdiğinizde bir kopya veri bloğu otomatik olarak atanır. MB_COMM_LOAD komutu için MB_DB parametrelerini belirtirken bu MB_MASTER kopya veri bloğu, kullanılır.

Çizelge 12- 68 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
REQ	IN	Bool	0=İstek yok 1= Modbus slave e veri iletim isteği
MB_ADDR	IN	V1.0: USInt V2.0: UInt	Modbus RTU istasyon adresi: Standart adresleme aralığı (1 ila 247) Genişletilmiş adresleme aralığı (1 ila 65535) 0 değeri, tüm Modbus slave'lere mesaj yayını için ayrılmıştır. Modbus fonksiyon kodları 05, 06, 15 ve 16, yayını için desteklenen tek fonksiyon kodlarıdır.
MODE	IN	USInt	Mod Seçimi: İstek türünü (okuma, yazma veya tanılama) belirtir. Ayrıntılar için aşağıdaki Modbus fonksiyonları çizelgesine bakınız.
DATA_ADDR	IN	UDInt	Slave deki başlangıç Adresi: Modbus slave de erişilecek verilerin başlangıç adresini belirtir. Geçerli adresler için aşağıdaki Modbus fonksiyonları çizelgesine bakınız.
DATA_LEN	IN	UInt	Veri uzunluğu: Bu istekte erişilebilir bit ya da word sayısını belirtir. Geçerli uzunluklar için aşağıdaki Modbus fonksiyonları çizelgesine bakınız.
DATA_PTR	IN	Variant	Veri işaretçisi: Yazılan ve okunan veri için (standart bir DB tipi) DB veya bir M bellek adresini işaret eder.
DONE	OUT	Bool	DONE biti bir tarama için TRUE olur, son istek hatasız tamamlandıktan sonra.
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>0 – MB_MASTER işlemi devam etmiyor</li> <li>1 – MB_MASTER işlemi devam ediyor</li> </ul>
ERROR	OUT	Bool	Bir hata ile sona erdirildikten sonra ERROR biti, bir tarama için TRUE olur. STATUS parametrelerindeki hata kodu değeri yalnızca ERROR =. TRUE olan tek çevrim esnasında geçerli olur .
STATUS	OUT	Word	Yürütme durum kodu

### Modbus master haberleşme kuralları

- Bir MB\_MASTER komutu bir port ile haberleşmeden önce bu portu yapılandırmak için MB\_COMM\_LOAD yürütülmelidir.
- Bir Modbus master isteklerini başlatmak için bir port kullanılacak ise, bu port MB\_SLAVE tarafından kullanılmamalıdır. MB\_MASTER yürütmesinin bir veya daha fazla kopyası bu port ile kullanılabilir, ancak tüm MB\_MASTER yürütmesi, o port için aynı MB\_MASTER kopya DB sini kullanması gerekir.
- Modbus komutları haberleşme sürecini kontrol etmek için haberleşme kesme olaylarını kullanmaz. Programınız, tüm koşulların iletimi ve alınması için MB\_MASTER komutunu yoklar.
- Bir program çevrim OB sinden belirli bir port için tüm MB\_MASTER yürütmesini çağırmanız önerilir. Modbus master komutları sadece bir program çevrimi veya döngüsel / zaman gecikme yürütme düzeylerinden birinde yürütebilir. Bunların her iki yürütme öncelik düzeylerinde yürütülmemesi gerekir. Daha yüksek öncelikli yürütmenin öncelik seviyesinde başka bir Modbus master komutu ile bir Modbus Master komutunun çağrı önceliği yanlış çalışmaya neden olur. Modbus master komutlarının, başlangıç, tanılama veya zaman hatası yürütme öncelik seviyelerinde yürütülmesi gerekir.
- Bir master komutu bir iletimi başlattığında, bu kopya bir DONE = 1 durumu veya ERROR = 1 durumu dönene kadar etkinleştirilmiş EN girişi ile sürekli olarak yürütülmelidir. Özel bir MB\_MASTER kopyası, bu iki olaydan biri oluşana kadar aktif kabul edilir. Orijinal kopya aktif iken, etkinleştirilmiş REQ girişi ile herhangi bir diğer kopya için yapılan herhangi bir çağrı, bir hataya neden olur. Orijinal kopyanın sürekli yürütmesi durursa, istek durumu statik değişken Blocked\_Proc\_Timeout tarafından belirtilen bir süre boyunca aktif kalır. Bu süre dolduktan sonra, etkin bir REQ girişi ile adlandırılan bir sonraki master komutu aktif kopya olacaktır. Bu Modbus master kopyasının, bir portu tekelleştirmesi veya kilitleme erişimini önler. Orijinal aktif kopya, statik değişken " Blocked\_Proc\_Timeout " tarafından belirtilen süre içinde etkin değilse, o zaman bu kopya tarafından bir sonraki yürütme (REQ ile set değil )aktif durumu temizleyecektir. REQ ayarlanırsa başka bir kopya aktifmiş gibi, bu yürütme yeni bir master isteğini başlatır.

### REQ parametreleri

0 = İstek yok; 1 = Modbus slave e veri iletim isteği

Ya bir seviye kullanımı ya da kenar tetiklemeli temas yoluyla bu girişi kontrol edebilirsiniz. Bu giriş etkin olduğunda, geçerli istek tamamlanıncaya kadar, aynı kopya DB'yi kullanan başka hiçbir MB\_MASTER'ın isteği yayınlamasına izin verilmediğinden emin olmak için bir durum makinesi başlatılır. Yanıt alınıncaya kadar ya da bir hata tespit edilene kadar diğer tüm giriş durumları yakalanır ve geçerli istek için dahili olarak tutulur.

MB\_MASTER aynı kopyası, geçerli isteğin tamamlanmasından önce REQ girişi = 1 ile tekrar çalıştırılır, daha sonra sonraki yayınlar yapılmaz. Ancak istek tamamlandığında MB\_MASTER REQ girişi = 1 ile yeniden yürütüldüğünde, yeni bir istek verilir.

**DATA\_ADDR ve MODE parametreleri Modbus fonksiyon tipini seçiniz**

DATA\_ADDR (slave de başlayan Modbus adresi): Modbus slave de erişilecek verinin başlangıç adresini belirtir.

MB\_MASTER komutu Fonksiyon Kodu girişi yerine MOD girişini kullanır. MODE ve Modbus adresi birleşimi gerçek Modbus mesajında kullanılan Fonksiyon Kodunu belirler. Aşağıdaki çizelgede MOD parametreleri, Modbus fonksiyon kodu ve Modbus adres aralığı arasındaki ilişkiyi gösterir.

Çizelge 12- 69 Modbus fonksiyonları

MOD	Modbus Fonksiyonu	Veri uzunluğu	İşlem ve veri	Modbus Adresi
0	01	1 ila 2000 1 ila 1992 <sup>1</sup>	Çıkış bitlerini oku: istek başına 1 ila (1992 veya 2000) bit	1 - 9999
0	02	1 ila 2000 1 ila 1992 <sup>1</sup>	Giriş bitlerini oku: istek başına 1 ila (1992 veya 2000) bit	10001- 19999
0	03	1 ila 125 1 ila 124 <sup>1</sup>	Tutma yazmaçlarını oku: istek başına 1 ila (124 veya 125) word	40001 - 49999 veya 400001 - 465535
0	04	1 ila 125 1 ila 124 <sup>1</sup>	Giriş Word'lerini oku: istek başına 1 ila (124 veya 125) word	30001 ila 39999
1	05	1	Bir çıkış biti yaz: istek başına bir bit	1 ila 9999
1	06	1	Tutma yazmacı yaz: istek başına 1 word	40001 ila 49999 veya 400001 ila 465535
1	15	2 ila 1968 2 ila 1960 <sup>1</sup>	Çoklu çıkış bitleri yaz: istek başına 2 ila (1960 veya 1968) bit	1 ila 9999
1	16	2 ila 123 2 ila 122 <sup>1</sup>	Çoklu tutma yazmacı yaz: istek başına 2 ila (122 veya 123) word	40001 ila 49999 veya 400001 ila 465535
2	15	1 ila 1968 2 ila 1960 <sup>1</sup>	Bir veya daha fazla çıkış biti yaz: istek başına 1 ila (1960 veya 1968) bit	1 ila 9999
2	16	1 ila 123 1 ila 122 <sup>1</sup>	Bir veya daha fazla tutma yazmacı yaz: istek başına 1 ila (122 veya 123) word	40001 ila 49999 veya 400001 ila 465535
11	11	0	Sunucu haberleşme durum word ünü ve olay sayacını okur. Durum word (0-meşgul değil, 0xFFFF - meşgul) meşgul gösterir. Olay sayacı, bir mesajın her başarılı tamamlanmasında artırılır. MB_MASTER'ın MB_DATA_ADDR ve MB_DATA_LEN parametrelerinin her ikisi bu işlev için göz ardı edilir.	
80	08	1	Veri tanılama kodu 0x0000 kullanılarak, sunucu durumunu kontrol ediniz (Loopback testi - slave isteği yansıtır) İstek başına 1 word	



MOD	Modbus Fonksiyonu	Veri uzunluğu	İşlem ve veri	Modbus Adresi
81	08	1	Veri tanımlama kodu 0x000A kullanılarak, slave olay sayacını sıfırlar. İstek başına 1 word	
3 ila 10, 12 ila 79, 82 ila 255			Rezerve edildi	

<sup>1</sup> " Genişletilmiş adresleme " modu için maksimum veri uzunlukları, fonksiyon tarafından kullanılan veri tipine bağlı olarak 1 bayt veya 1 word azalır.

### DATA\_PTR parametreleri

DATA\_PTR parametreleri yazılır veya okunur DB veya M adrese puan. Bir veri bloğu kullanırsanız, o zaman okur ve Modbus slave yazar için veri depolama sağlayan bir global veri bloğu oluşturmanız gerekir.

#### Not

#### DATA\_PTR veri blok tipi doğrudan adresleme izin vermesi gerekir

Veri bloğu (mutlak) doğrudan hem de sembolik adresleme izin vermelidir. Eğer veri blok oluşturduğunuzda "Standart" erişim özelliği seçili olması gerekir.

### DATA\_PTR parametre için veri bloğu yapıları

- Bu veri tipleri, Sözcük, 30.001-39.999, 40001-49999 ve 400.001-465.536 Modbus adreslerinin word okumaları için ve aynı zamanda 40.001-49.999 ve 400.001-465.536 Modbus adreslerinin word yazmaları için geçerlidir.
  - WORD, UINT, ya da INT veri tiplerindeki standart dizi.
  - Her bir elemanın benzersiz bir ada ve 16 bit veri tipine sahip olduğu İsimlendirilmiş WORD, UINT ya da INT yapısı.
  - Her bir elemanın benzersiz bir ada ve 16 veya 32 bit veri tipine sahip olduğu İsimlendirilmiş karmaşık yapı.
- 00001-09999 Modbus adreslerinin Bit okumaları ve yazmaları için ve 10.001-19.999 Modbus adreslerinin okumaları için.
  - Boolean veri tipinde Standart dizi.
  - Benzersiz isimli Boole değişkenlerinin isimlendirilmiş Boole yapısı.
- Gerekli olmasa da, her MB\_MASTER komutunun kendine ait ayrı bir bellek alanı olması önerilir. Bu önerinin sebebi, çoklu MB\_MASTER komutu, aynı bellek alanına yazar ve buradan okursa veri bozulma olasılığının daha büyük olmasıdır.
- DATA\_PTR veri alanları aynı küresel veri bloğunda olma gereksinimi yoktur. Modbus okumaları için çoklu kullanım alanlara sahip bir veri bloğu, Modbus yazmalar için bir veri bloğu, ya da her bir slave istasyonu için bir veri bloğu oluşturabilir

### Modbus master veri bloğu değişkenleri

Aşağıdaki çizelgede, programda kullanılan olabilir MB\_MASTER için kopya DB saklanan yerel statik değişkenleri gösterir.

Çizelge 12- 70 Kopya DB de Statik değişkenler

Değişken	Veri tipi	Başlangıç değeri	Açıklama
Blocked_Proc_Timeout	Real	3.0	ACTIVE olarak bu kopyayı kaldırmadan önce engellenen bir Modbus istemci kopyasının (Saniye cinsinden) beklenme zaman miktarı. Bu, örneğin, Bir Master isteği verildiğinde oluşabilir ve program tamamen isteği bitirmeden önce uygulama, master işlevini çağırılmayı durdurur. Zaman değeri 0'dan büyük ve 55 saniyeden daha az olmalıdır, yoksa bir hata oluşur. Varsayılan değer 0,5 saniyedir.
Extended_Addressing	Bool	False	Tek veya çift bayt slave adreslemeyi yapılandırır. Varsayılan değer = 0. (0 = tek bayt adresi, 1 = çift bayt adresi)

Programınız, Modbus master işlemleri kontrol etmek için Blocked\_Proc\_Timeout ve Extended\_Addressing değişkenlere değerler yazabilirsiniz. Modbus (Sayfa 755) genişletilmiş adresleme hakkında detaylar ve program editöründeki bu değişkenlerin nasıl kullanılacağıyla ilgili bir örnek için HR\_Start\_Offset ve Extended\_Addressing ile ilgili MB\_SLAVE konu açıklamasına bakınız.

### Durum kodları

 Çizelge 12- 71 MB\_MASTER yürütme durum kodları (haberleşme ve yapılandırma hataları) <sup>1</sup>

STATUS (W#16#)	Açıklama
0000	Hata yok
80C8	Slave zaman aşımı. slave baud hızı, parite ve kablağı kontrol et.
80D1	Alıcı, etkin bir iletimi askıya almak için bir akış kontrol isteği yayınladı ve belirtilen bekleme süresince iletim yeniden etkinleştirilmedi. Alıcı belirtilen bekleme süresi içinde CTS belirtmezse, bu hata, donanım akış kontrolü sırasında oluşturulur.
80D2	DCE'den DSR sinyali alınmadığı için iletim isteği iptal edildi.
80E0	Mesaj sonlandırıldı Çünkü alıcı arabellek dolu.
80E1	Parite hatasının sonucu olarak mesaj sonlandırıldı.
80E2	Çerçeve hatasının sonucu olarak mesaj sonlandırıldı.
80E3	Taşma hatasının sonucu olarak mesaj sonlandırıldı.
80E4	Bu mesaj, belirtilen uzunluğun toplam ara bellek boyutunu aşmasının sonucu olarak sonlandırıldı.
8180	MB_COMM_LOAD komutu ile Geçersiz port ID değeri veya hata
8186	Geçersiz Modbus istasyon adresi
8188	Yayın isteği için belirtilen geçersiz mod
8189	Geçersiz veri Adres değeri
818A	Geçersiz veri Uzunluk değeri
818B	Yerel veri kaynağı / hedefi için geçersiz işaretçi: Boyut doğru değil

STATUS (W#16#)	Açıklama
818C	DATA_PTR veya geçersiz Blocked_Proc_Timeout için geçersiz işaretçi: veri alanı, bir DB (sembolik ve doğrudan erişimin her ikisine de izin verir) veya M bellek olmalıdır.
8200	Port bir gönderme isteğini işlemeye meşgul.


Çizelge 12- 72 MB\_MASTER yürütme durum kodları (Modbus protokol hataları) <sup>1</sup>

STATUS (W#16#)	Slave den yanıt kodu	Modbus protokol hataları
8380	-	CRC hatası
8381	01	Fonksiyon kodu desteklenmiyor
8382	03	Veri uzunluk hatası
8383	02	Veri adres hatası veya DATA_PTR geçerli adres alanı sınırları dışında
8384	03 ten büyük	Veri değer hatası
8385	03	Veri tanılama kod değeri desteklenmiyor (Fonksiyon kodu 08)
8386	-	Tepkideki fonksiyon kodu, istekteki kodla eşleşmiyor.
8387	-	Yanlış slave yanıt verdi
8388	-	Slave in yazma isteğine yanıtı yanlış. Slave tarafından döndürülen yazma isteği Master'ın gerçekte gönderdiği ile eşleşmiyor.

<sup>1</sup> Yukarıda listelenen MB\_MASTER komut hatalarına ek olarak, hatalar temel PtP haberleşme komutlarından döndürülebilir.

### 12.5.3.3 MB\_SLAVE (Modbus slave olarak PtP portu üzerinden haberleşme) komutu

Çizelge 12- 73 MB\_SLAVE komutu

LAD / FBD	SCL	Açıklama
	<pre>"MB_SLAVE_DB" (     MB_ADDR:= uint_in ,     NDR=&gt; bool_out ,     DR=&gt; bool_out ,     ERROR=&gt; bool_out ,     STATUS=&gt; word_out ,     MB_HOLD_REG:= _variant_inout ) ;</pre>	MB_SLAVE komutu, programınızın CM (RS485 veya RS232) ve CB (RS485) de bir PtP portu üzerinden bir Modbus slave olarak iletişim kurmasını sağlar. Uzak bir Modbus RTU master, bir isteği yayımladığında, kullanıcı programınız MB_SLAVE yürütmesi tarafından isteğe cevap verir. Komutu yerleştirdiğinizde STEP 7 otomatik olarak bir kopya DB oluşturur. MB_COMM_LOAD komutu için MB_DB parametrelerini belirttiğinizde bu MB_SLAVE_DB ismini kullanınız.

Çizelge 12- 74 Parametreler için veri tipleri

Parametre ve tipi		Veri tipi	Açıklama
MB_ADDR	IN	V1.0: USInt V2.0: UInt	Modbus slave istasyon adresi: Standart adresleme aralığı (1 ila 247) Genişletilmiş adresleme aralığı (0 ila 65535)
MB_HOLD_REG	IN	Variant	Modbus Tutma yazmacı DB ye İşaretçi: Modbus tutma yazmacı M bellek veya bir veri bloğu olabilir.
NDR	OUT	Bool	Yeni Veri Hazır: <ul style="list-style-type: none"> <li>0 – Yeni veri yok</li> <li>1 – Yeni verinin Modbus master tarafından yazıldığını gösterir</li> </ul>
DR	OUT	Bool	Veri okuma: <ul style="list-style-type: none"> <li>0 – Okunan veri yok</li> <li>1 – Yeni verinin Modbus master tarafından okunduğunu gösterir</li> </ul>
ERROR	OUT	Bool	Son istek bir hata ile sona erdirildikten sonra ERROR biti, bir tarama için TRUE olur. STATUS parametrelerindeki hata kodu değeri yalnızca ERROR = TRUE olan tek çevrim esnasında geçerli olur.
STATUS	OUT	Word	Yürütme Hata kodu

Modbus fonksiyon kodlarının (1, 2, 4, 5 ve 15) CPU nun giriş proses görüntü ve çıkış proses görüntüsünde doğrudan bit ve Word'leri okumasına veya yazmasına izin verir. Bu fonksiyon kodları için, MB\_HOLD\_REG parametreleri, bir bayttan daha büyük bir veri tipi olarak tanımlanmalıdır. Aşağıdaki çizelgede CPU'daki proses görüntüsü için Modbus adreslerinin örnek haritalanması gösterilmiştir.

Çizelge 12- 75 Proses görüntü için Modbus adreslerinin haritalama

Modbus fonksiyonları						S7-1200	
Kodlar	Fonksiyon	Veri alanı	Adres aralığı			Veri alanı	CPU adresi
01	Bitleri oku	Çıkış	1	ila	8192	Çıkış Proses Görüntüsü	Q0.0 ila Q1023.7
02	Bitleri oku	Giriş	10001	ila	18192	Giriş Proses Görüntüsü	I0.0 ila I1023.7
04	Word'leri oku	Giriş	30001	ila	30512	Giriş Proses Görüntüsü	IW0 ila IW1022
05	Bit yaz	Çıkış	1	ila	8192	Çıkış Proses Görüntüsü	Q0.0 ila Q1023.7
15	Bitleri yaz	Çıkış	1	ila	8192	Çıkış Proses Görüntüsü	Q0.0 ila Q1023.7

Modbus haberleşme fonksiyon kodları (3, 6 ve 16) Bir M bellek adresi aralığını veya bir veri bloğu olabilen Modbus tutma yazmacısını kullanır. Tutma yazmacısının tipi MB\_SLAVE komutundaki MB\_HOLD\_REG parametreleri tarafından belirtilir.

#### Not

#### MB\_HOLD\_REG veri blok tipi

Modbus tutma yazmacı veri bloğu, hem doğrudan (mutlak) hem de sembolik adreslemeye izin vermelidir. Veri bloğu oluşturduğunuzda "Standart" erişim özelliği seçili olması gerekir.

Aşağıdaki Çizelge Modbus fonksiyon kodları 03 (word okuma), 06 (word yazma) ve 16 (word yazma) için kullanılan tutma yazmacı haritalama için Modbus adresi örneklerini göstermektedir. DB adreslerinin gerçek üst sınırı, her bir CPU modeli için maksimum çalışma bellek sınırı ve M bellek sınırı tarafından belirlenir.

Çizelge 12- 76 CPU bellek için Modbus adreslerinin haritalanması

Modbus Master Adresi	MB_HOLD_REG parametre örnekleri				
	MW100	DB10.DBW0	MW120	DB10.DBW50	"Tarif".içerik
40001	MW100	DB10.DBW0	MW120	DB10.DBW50	" Tarif ". içerik [1]
40002	MW102	DB10.DBW2	MW122	DB10.DBW52	" Tarif ". içerik [2]
40003	MW104	DB10.DBW4	MW124	DB10.DBW54	" Tarif ". içerik [3]
40004	MW106	DB10.DBW6	MW126	DB10.DBW56	" Tarif ". içerik [4]
40005	MW108	DB10.DBW8	MW128	DB10.DBW58	" Tarif ". içerik [5]

Çizelge 12- 77 Tanılama fonksiyonları

S7-1200 MB_SLAVE Modbus tanılama fonksiyonları		
Kodlar	Alt-fonksiyon	Açıklama
08	0000H	Sorgu verisi yansıma testini döndürür: MB_SLAVE, alınan word veriyi Modbus master'a yansıtır.
08	000AH	Haberleşme olay sayacını temizler: MB_SLAVE, Modbus işlevi 11 için kullanılan haberleşme olay sayacını temizleyecektir.
11		Haberleşme olay sayacını alınız: MB_SLAVE, Modbus sunucuya gönderilen başarılı Modbus okuma ve yazma isteklerinin kaydetmek için bir iç haberleşme olay sayacı kullanır. Sayıcı herhangi Fonksiyon 8 veya Fonksiyon 11 isteklerinde artmaz. Aynı zamanda bir haberleşme hatasına neden olan herhangi bir istekte de artmaz.

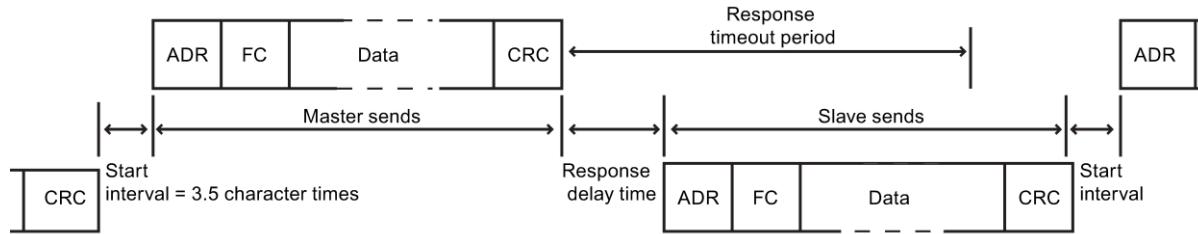
İstek geçerli adreslere erişmek için olduğu sürece MB\_SLAVE komutu, herhangi bir Modbus master'dan yayın yazma isteklerini destekler. MB\_SLAVE yayında desteklenmeyen fonksiyon kodları için 0x8188 hata kodunu üretecektir.

### Modbus slave haberleşme kuralları

- Bir MB\_SLAVE komutu bir port ile haberleşmeden önce bu portu yapılandırmak için MB\_COMM\_LOAD yürütülmelidir.
- Bir port bir slave olarak Modbus master'a yanıt verecek ise, bu portu MB\_MASTER komutu ile programlamayın.
- Yalnızca bir kopya MB\_SLAVE, belirli bir port ile birlikte kullanılabilir aksi taktirde düzensiz davranış oluşabilir.
- Modbus komutları haberleşme sürecini kontrol etmek için haberleşme kesme olaylarını kullanmaz. Programınız, tüm koşulların iletimi ve alınması için MB\_SLAVE komutunu yoklayarak haberleşme sürecini kontrol etmelidir.
- MB\_SLAVE komutu, Modbus master'dan gelen isteklere zamanında cevap vermesine imkan veren bir hızda periyodik olarak yürütülmesi gerekir. MB\_SLAVE' i, bir program çevrimi OB den her taramada yürütmeniz önerilir. Döngüsel bir kesme OB sinden MB\_SLAVE yürütülmesi mümkündür, ama diğer kesme rutinlerinin yürütülmesini geçici olarak engellemek için kesme rutinindeki aşırı zaman gecikme potansiyeli nedeniyle tavsiye edilmez.

### Modbus sinyal zamanlaması

MB\_SLAVE Modbus Master'dan her isteği almak ve gerektiği gibi yanıt vermek için periyodik olarak yürütülmelidir. MB\_SLAVE için yürütme sıklığı Modbus master yanıt zaman aşımı süresi bağlıdır. Bu, aşağıdaki şemada gösterilmiştir.



Yanıt zaman aşımı süresi RESP\_TO, bir Modbus slave'den bir yanıt başlangıcı için bir Modbus master'ın bekleyeceği süre miktarıdır. Bu süre Modbus protokolü tarafından tanımlanmaz, ancak her Modbus master'ın bir parametresidir. MB\_SLAVE (bir yürütme ile bir sonraki yürütme arasındaki zaman) yürütme sıklığı, Modbus master'ınızın özel parametrelerine dayalı olmalıdır. En azından, Modbus master'ın yanıt zaman aşımı süresi içinde MB\_SLAVE'i iki kez yürütmek gerekir.

## Modbus slave değişkenleri

Bu Çizelge, programınızda kullanılacak MB\_SLAVE kopya veri bloğunda saklanan yerel statik değişkenleri gösterir.

Çizelge 12- 78 Modbus slave değişkenleri

Değişken	Veri tipi	Açıklama
Request_Count	Word	Bu slave tarafından alınan tüm isteklerin sayısı
Slave_Message_Count	Word	Bu özel slave için alınan isteklerin sayısı
Bad_CRC_Count	Word	CRC hatası olan alınan isteklerin sayısı
Broadcast_Count	Word	Alınan yayın isteklerinin sayısı
Exception_Count	Word	Bir döndürülen istisna gerektiren Modbus a özgü hatalar
Success_Count	Word	Hiçbir protokol hatası olmayan bu özel slave için alınan isteklerin sayısı
HR_Start_Offset	Word	Modbus Tutma kaydının başlangıç adresi (varsayılan= 0) belirtir
Extended_Addresssing	Bool	Adresleme tek veya çift bayt slave yapılandırır (0= tek bayt adresi, 1=çift bayt adresi, Varsayılan = 0)

Programınız, HR\_Start\_Offset ve Extended\_Addresssing değişkenlerine değerler yazabilir ve Modbus slave işlemlerini kontrol edebilirsiniz. Diğer değişkenler Modbus durumunu izlemek için okunabilir.

## HR\_Start\_Offset

Modbus tutma register adresleri, 40001 veya 400001 'den başlar. Bu adresler tutma kaydının başlangıç PLC bellek adresine karşılık gelir. Ancak, 40001 veya 400001 yerine başka değerlerde başlangıç Modbus tutma kayıt adresini başlatmak için "HR\_Start\_Offset" değişkenini yapılandırabilirsiniz.

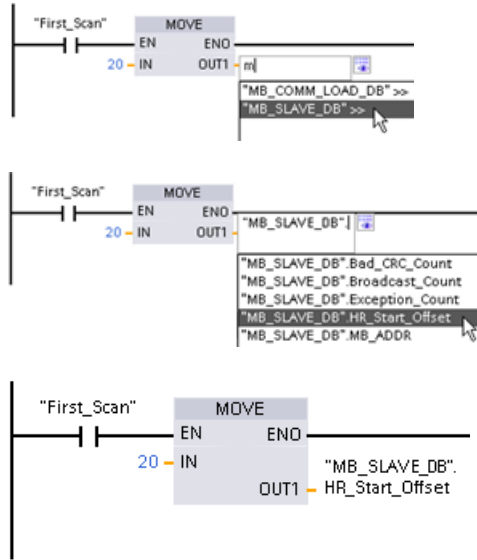
Örneğin, tutma register MW100'de başlayacak şekilde yapılandırılırsa, tutma register 100 word uzunluğundadır. 20 ofset değeri, 40001 yerine başlangıç tutma register adresini 40021 olarak belirtir. 40021 altındaki ve 400119 üstündeki herhangi bir adres bir adresleme hatasına neden olur.

Çizelge 12- 79 Adresleme Modbus tutma register örneği

HR_Start_Offset	Adres	Minimum	Maksimum
0	Modbus adresi (Word)	40001	40099
	S7-1200 adresi	MW100	MW298
20	Modbus adresi (Word)	40021	40119
	S7-1200 adresi	MW100	MW298

HR\_Start\_Offset, Modbus tutma register başlangıç adresini belirten bir word değeridir ve MB\_SLAVE kopya veri bloğunda saklanır. MB\_SLAVE programınıza yerleştirildikten sonra, parametre yardımcı açılan liste kullanılarak bu yerel statik değişkenin değerini ayarlayabilirsiniz.

Örneğin MB\_SLAVE, LAD devresine yerleştirildikten sonra, bir önceki devreye gidebilir ve HR\_Start\_Offset değerini atayabilirsiniz. Değer, MB\_SLAVE yürütmesinden önce atanmalıdır.



Varsayılan DB ismini kullanarak Modbus slave değişkeni girme:

9. Parametre alanına imleci getirin ve bir m karakteri yazınız.

10. Açılan listeden "MB\_SLAVE\_DB" seçeneğini seçiniz.

11. (Tırnak karakterinden sonra) DB adının sağ tarafına imleci getirin ve bir periyot karakteri giriniz.

12. Açılan listeden "MB\_SLAVE\_DB.HR\_Start\_Offset" seçeneğini seçiniz.

## Extended Addressing

Extended Addressing değişkenine, Extended Addressing değişkeninin bir Boolean değer olması dışında yukarıda tartışılan HR\_Start\_Offset referans ile benzer bir şekilde erişilir. Boole değeri bir hareket kutusu değil bir çıkış bobini tarafından yazılmış olmalıdır.

Modbus slave adresleme, tek bir bayt (Modbus standart) veya çift bayt olarak yapılandırılabilir. Genişletilmiş adresleme, tek bir devre içinde 247 den fazla cihazı adreslemek için kullanılır. Genişletilmiş adresleme seçimi, maksimum 64000 adresi adreslemenize izin verir. Modbus fonksiyon 1 çerçeve bir örnek olarak aşağıda gösterilmiştir.

Çizelge 12- 80 Tek bayt slave adresi (bayt 0)

Fonksiyon 1	Bayt 0	Bayt 1	Bayt 2	Bayt 3	Bayt 4	Bayt 5	
İstek	Slave adresi	F kodu	Başlangıç adresi		Bobinlerin Uzunluğu		
Geçerli yanıt	Slave adresi.	F kodu	Uzunluk	Bobin verisi			
Hata yanıtı	Slave adresi.	0x81	E kodu				

Çizelge 12- 81 Çift bayt slave adresi (bayt 0 ve bayt 1)

	Bayt 0	Bayt 1	Bayt 2	Bayt 3	Bayt 4	Bayt 5	Bayt 6
İstek	Slave adresi		F kodu	Başlangıç adresi		Bobinlerin Uzunluğu	
Geçerli yanıt	Slave adresi.		F kodu	Uzunluk	Bobin verisi		
Hata yanıtı	Slave adresi.		0x81	E kodu			



## Durum kodları

 Çizelge 12- 82 MB\_SLAVE yürütme durum kodları (Haberleşme ve yapılandırma hataları) <sup>1</sup>

STATUS (W#16#)	Açıklama
80D1	Alıcı, etkin bir iletimi askıya almak için bir akış kontrol isteği yayınladı ve belirtilen bekleme süresince iletim yeniden etkinleştirilmedi. Alıcı belirtilen bekleme süresi içinde CTS belirtmezse, bu hata, donanım akış kontrolü sırasında oluşturulur.
80D2	DCE'den DSR sinyali alınmadığı için iletim isteği iptal edildi.
80E0	Mesaj sonlandırıldı Çünkü alıcı arabellek dolu.
80E1	Parite hatasının sonucu olarak mesaj sonlandırıldı.
80E2	Çerçeve hatasının sonucu olarak mesaj sonlandırıldı.
80E3	Taşma hatasının sonucu olarak mesaj sonlandırıldı.
80E4	Bu mesaj, belirtilen uzunluğun toplam ara bellek boyutunu aşmasının sonucu olarak sonlandırıldı.
8180	MB_COMM_LOAD komutu ile Geçersiz port ID değeri veya hata
8186	Geçersiz Modbus istasyon adresi
8187	MB_HOLD_REG DB ye geçersiz işaretçi: Alan çok küçük
818C	M bellek veya DB ye geçersiz MB_HOLD_REG işaretçi (DB alanı sembolik ve doğrudan adresin ikisine de izin vermelidir)

 Çizelge 12- 83 MB\_SLAVE yürütme durum kodları (Modbus protokol hataları) <sup>1</sup>

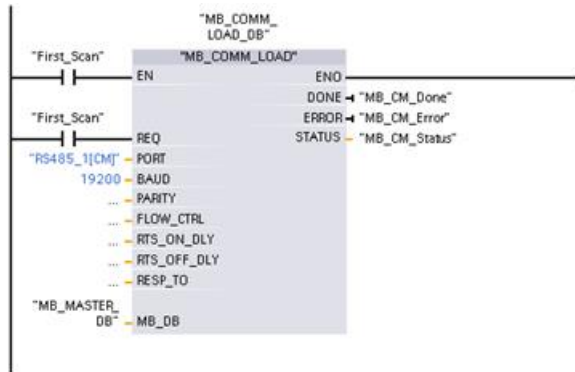
STATUS (W#16#)	Slave'den yanıt kodu	Modbus protokol hataları
8380	Yanıt yok	CRC hatası
8381	01	Fonksiyon kodu desteklenmiyor veya yayınlar içerisinde desteklenmiyor
8382	03	Veri uzunluk hatası
8383	02	Veri adres hatası veya DATA_PTR geçerli adres alanı sınırları dışında
8384	03	Veri değer hatası
8385	03	Veri tanılama kod değeri desteklenmiyor (Fonksiyon kodu 08)

<sup>1</sup> Yukarıda listelenen MB\_SLAVE komut hatalarına ek olarak, hatalar temel PtP haberleşme komutlarından döndürülebilir.

### 12.5.3.4 Modbus RTU master örnek program

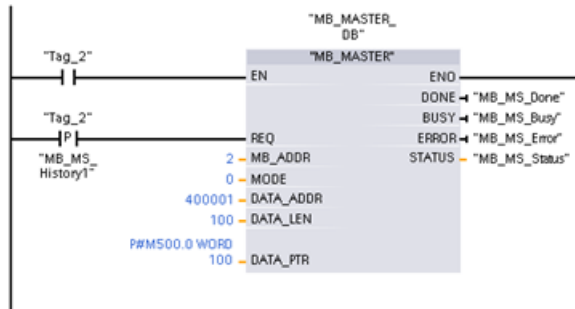
MB\_COMM\_LOAD ilk tarama bayrağını kullanarak yol verme sırasında başlatıldı. Seri port yapılandırması çalışma zamanında değişmediğinde MB\_COMM\_LOAD Yürütmesi yalnızca bu şekilde yapılmalıdır.

**Devre 1** ilk tarama sırasında RS485 modülü parametrelerini sadece bir kez başlat.

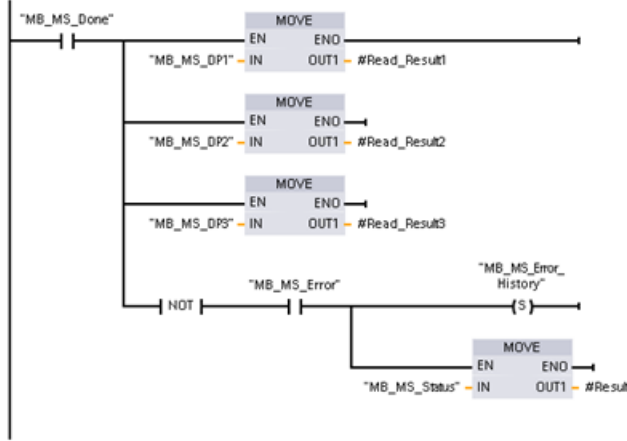


Bir MB\_MASTER komutu, tek bir slave ile iletişim kurmak için bir program çevrimi OB sinde kullanılır. Ek MB\_MASTER komutları, diğer slave'ler ile iletişim kurmak için bir program çevrimi OB sinde kullanılabilir ya da bir MB\_MASTER FB, ek slave'ler ile iletişim kurmak için yeniden kullanılabilir.

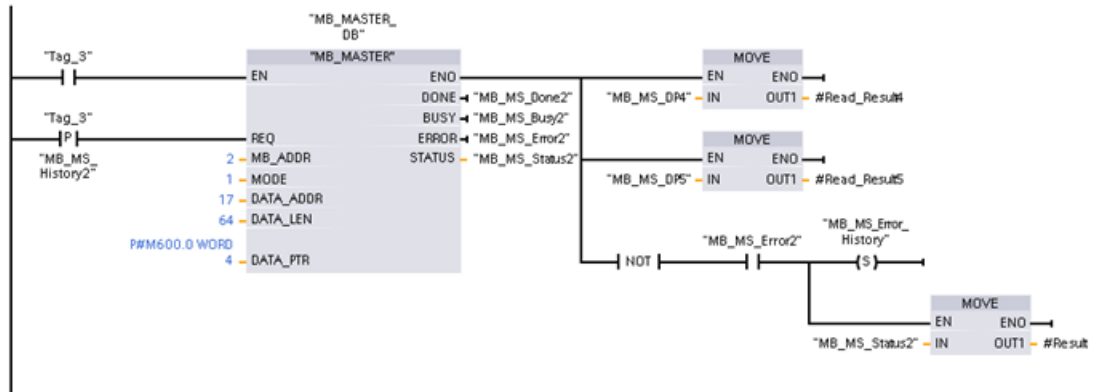
**Network 2** Slave tutma register'den 100 word okuyunuz.



**Network 3** Okuma işlemi bir kez yapıldığında bu sadece ilk 3 word değerini gösteren opsiyonel devredir.



**Devre 4** Slave adresi Q2.0 dan başlayarak çıkış görüntü register'ine 64 bit yazınız

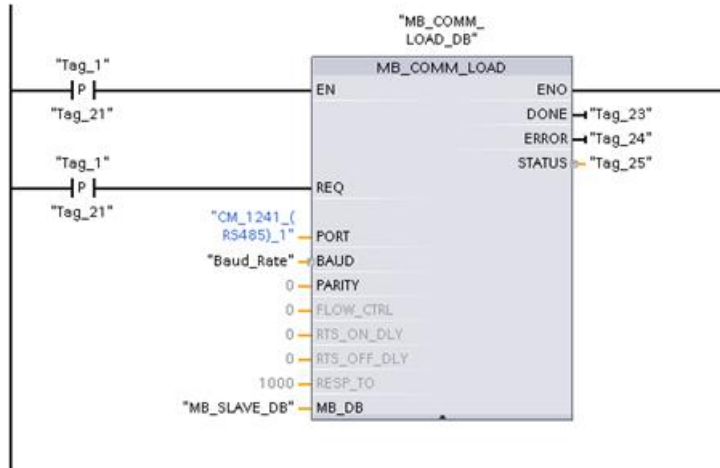


### 12.5.3.5 Modbus RTU slave örnek program

Aşağıda gösterilen MB\_COMM\_LOAD, "Tag\_1" in her etkinleştirilmesinde başlatılır.

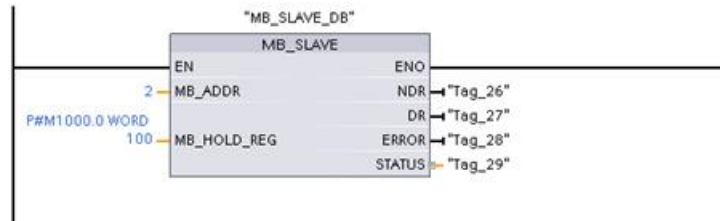
HMI yapılandırma sonucunda seri port yapılandırma çalışma zamanında değiştiğinde, MB\_COMM\_LOAD Yürütmesi sadece bu şekilde yapılmalıdır.

**Devre 1** RS485 modülü parametreleri HMI cihazı tarafından her değiştirildiğinde başlatılır.



Aşağıda gösterilen MB\_SLAVE her 10ms de yürütülen döngüsel bir OB ye yerleştirilir. Bu slave ile mutlak en hızlı tepki vermez iken, kısa mesajlar (istekte 20 bayt veya daha az) için 9600 baud'da iyi performans sağlamaz.

**Devre 2** Her tarama sırasında Modbus master isteklerini kontrol ediniz. Modbus tutma register, MW1000 de başlayan 100 word için yapılandırılmıştır.



## 12.6 CP 1242-7 ile telekontrol ve teleservis

### 12.6.1 GSM şebekesine bağlantı

#### GPRS üzerinden IP tabanlı WAN haberleşme

CP 1242-7 haberleşme işlemcisini kullanarak, S7-1200 GSM şebekelerine bağlanabilir. CP 1242-7, bir kontrol merkezi ve istasyonlar arası haberleşme ile uzak istasyonlardan WAN haberleşmesi sağlar.

İstasyonlar arası haberleşme, sadece bir GSM şebekesi üzerinden mümkündür. Bir uzak istasyon ile bir kontrol odası arasındaki haberleşme için kontrol merkezinde, internet erişimi olan bir bilgisayar olmalıdır.

CP 1242-7, GSM şebekesi üzerinden haberleşme için aşağıdaki hizmetleri destekler:

- GPRS (Genel Paket Radyo Servisi)

Veri iletimi "GPRS" için paket-odaklı hizmet, GSM şebekesi üzerinden işlenir.

- SMS (Kısa Mesaj Servisi)

CP 1242-7, SMS mesajları alır ve gönderebilir. Haberleşme partneri, bir cep telefonu veya bir S7-1200 olabilir.

CP 1242-7, dünya çapında endüstride kullanım için uygundur ve aşağıdaki frekans bandlarını destekler:

- 850 MHz
- 900 MHz
- 1 800 MHz
- 1 900 MHz

**Gereksinimler**

İstasyonlarda veya kontrol merkezinde kullanılan ekipman özel uygulamaya bağlıdır.

- Merkezi kontrol odası ile ya da bunun aracılığıyla haberleşme için, kontrol merkezi internet erişimi olan bir bilgisayar gerektirir.
- İstasyon donanımları dışında, bir CP 1242-7 ye sahip uzak S7-1200 istasyonu, GSM şebekesi üzerinden iletişim kurabilmek için aşağıdaki gereksinimleri karşılaması gerekir:
  - Uygun bir GSM şebeke sağlayıcısı ile bir sözleşme  
GPRS kullanılırsa, sözleşme GPRS hizmet kullanımına izin vermesi gerekir.  
Sadece GSM şebekesi üzerinden istasyonlar arasında doğrudan iletişim olacaksa, GSM şebeke sağlayıcısı CP'lere sabit bir IP adresi ataması gerekir. Bu durumda, istasyonlar arasındaki haberleşme kontrol merkezi üzerinden yapılmaz.
  - Sözleşmesine ait SIM kart  
CP 1242-7 ye SIM kart eklenir.
  - İstasyon kapsamı içinde bir GSM şebekesinin yerel erişilebilirliği

## 12.6.2 CP 1242-7 uygulamaları

CP 1242-7 aşağıdaki uygulamalar için kullanılabilir:

### Telekontrol uygulamaları

- SMS mesajları gönderme

CP 1242-7 üzerinden, uzak S7-1200 istasyonunun CPU su, GSM şebekesinden SMS mesajları alabilir veya yapılandırılmış bir cep telefonu veya bir S7-1200 e SMS ile mesaj gönderebilirsiniz.

- Bir kontrol merkezi ile haberleşme

Uzak S7-1200 istasyonları Master istasyonunda bir telekontrol sunucusu ile GSM şebekesi ve internet üzerinden haberleşir. GPRS kullanarak yapılan veri transferi için "TELECONTROL SERVER BASIC" uygulaması Master istasyonundaki telekontrol sunucusu üzerine kurulur. Telekontrol sunucusu, bütünleşmiş OPC sunucusu işlevini kullanarak daha yüksek seviyeli bir merkezi kontrol sistemi ile iletişim kurar.

- Bir GSM şebekesi üzerinden S7-1200 istasyonları arasındaki haberleşme

Bir CP 1242-7 ile uzak istasyonlar arasındaki haberleşme, iki farklı yoldan ele alınabilir:

- Bir master istasyonu aracılığı ile istasyonlar arası haberleşme

Bu yapılandırmada, birbirleri ile ve telekontrol sunucusu ile iletişim kuran S7-1200 istasyonları arasında kalıcı güvenli bir bağlantı, Master istasyonda kurulur. İstasyonlar arasındaki haberleşme telekontrol sunucusu üzerinden yapılır. CP 1242-7 "Telekontrol" modunda çalışır.

- İstasyonları arası doğrudan haberleşme

Master istasyonu üzerinden dolambaçlı olmadan istasyonlar arası doğrudan haberleşme için, istasyonların birbirleriyle doğrudan adreslemesine izin veren sabit bir IP adresi olan SIM kartlar kullanılır. Olası haberleşme hizmetleri ve güvenlik fonksiyonları (örneğin VPN) şebeke sağlayıcısı tarafından sunulana bağlıdır. CP 1242-7 "GPRS direct" modunda çalışır.

### GPRS üzerinden TeleServis

Bir TeleServis bağlantısı, GSM şebekesi ve internet üzerinden, STEP 7'li bir mühendislik istasyonu ve CP 1242-7'li uzak S7-1200 istasyonu arasında kurulabilir. Bağlantı, bir aracı yönlendirme çerçeveleri gibi davranan ve yetki kuran bir TeleServis ağ geçidi veya bir telekontrol sunucusu üzerinden mühendislik istasyonunda çalışır. Bu bilgisayarlar "TELECONTROL SERVER BASIC" uygulaması işlevlerini kullanır.

Aşağıdaki amaçlar için TeleServis bağlantısını kullanabilirsiniz:

- İstasyona STEP 7 projesinde yapılandırma veya program verileri indirme
- İstasyonda tanılama verilerini sorgulama

### 12.6.3 CP'nin diğer özellikleri

#### CP 1242-7'nin diğer hizmetleri ve fonksiyonları

- İnternet üzerinden CP'nin günün saati senkronizasyonu  
Aşağıdaki gibi CP üzerindeki saati ayarlayabilirsiniz:
  - "Telekontrol" modunda, günün saati telekontrol sunucusu tarafından aktarılır. CP zamanı ayarlamak için bunu kullanır.
  - "GPRS direct" modunda ,CP, SNTP kullanarak zaman talep edebilir.  
CPU zamanını senkronize etmek için bir blok kullanılarak CP'den geçerli zamanı okuyabilirsiniz.
- Bağlantı sorunları varsa gönderilecek mesajlar geçici arabelleğe alınır
- Yedek telekontrol sunucusuna bağlanma seçeneği sayesinde artan emre amadelik
- Optimize veri hacmi ( geçici bağlantı)  
Telekontrol sunucusuna kalıcı bir bağlantı için bir alternatif olarak, CP telekontrol sunucusuna yapılan geçici bağlantı ile STEP 7'de yapılandırılabilir. Bu durumda sadece gerektiğinde, telekontrol sunucusuyla bir bağlantı kurulur.
- Veri hacmiyle ilgili günlük oluşturma  
Aktarılan veri hacimleri kaydedilir ve belirli amaçlar için değerlendirilebilir.

#### Yapılandırma ve modül değiştirme

Modülü yapılandırmak için aşağıdaki yapılandırma aracı gereklidir:

STEP 7 sürüm V11.0 SP1 veya daha yüksek

STEP 7 V11.0 SP1 için, ayrıca "CP 1242-7" (HSP0003001) destek paketine ihtiyacınız olur.

GPRS kullanarak yapılan proses veri transferi için istasyonun kullanıcı programındaki telekontrol haberleşme komutlarını kullanınız.

CP 1242-7'nin yapılandırma verileri yerel CPU üzerinde saklanır. Bu, gerekli olduğunda, CP'nin basitçe değiştirilmesini sağlar.

S7-1200 başına CP 1242-7 tipinde üç modüle kadar ekleyebilirsiniz. Bu, örneğin, yedek iletişim yollarının kurulmasını sağlar.

#### Elektrik bağlantıları

- CP 1242-7 güç kaynağı  
CP, harici 24 VDC güç kaynağı için ayrı bir bağlantısı vardır.
- GSM şebekesi için kablosuz ara yüz  
GSM haberleşme için ekstra bir anten gereklidir. Bu, CP'nin SMA soketine bağlanır.



## Daha fazla bilgi

CP 1242-7 kılavuzu ayrıntılı bilgi içermektedir. Aşağıdaki giriş ID si altında bunu Siemens Endüstriyel Otomasyon Müşteri Destek sayfalarında internette bulabilirsiniz:

45605894 (<http://support.automation.siemens.com/WW/view/en/45605894>)

## 12.6.4 Yardımcı donanımlar

### ANT794-4MR GSM/GPRS anteni

Aşağıdaki antenler, GSM / GPRS şebekelerinde kullanıma uygundur ve hem bina içi hem de bina dışı olarak monte edilebilir:

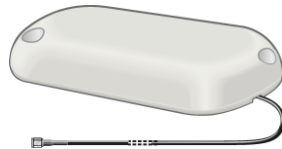
- Quadband anten ANT794-4MR



Şekil 12-1 ANT794-4MR GSM/GPRS anten

Kısa ismi	Sipariş no.	Açıklama
ANT794-4MR	6NH9 860-1AA00	Quadband anten (900, 1800/1900 MHz, UMTS); bina içi ve bina dışı alanlarda hava şartlarına dayanıklı; Antene sürekli bağlı 5 m bağlantı kablosu; Montaj aparatı, vidalar, duvar prizleriyle birlikte SMA konektör

- Yassı anten ANT794-3M



Şekil 12-2 Yassı anten ANT794-3M

Kısa ismi	Sipariş no.	Açıklama
ANT794-3M	6NH9 870-1AA00	Bina içi ve bina dışı alanlarda hava şartlarına dayanıklı; antene sürekli bağlı 1,2 m bağlantı kablosu; Yapışkanlı pad dahil SMA konektör, vidalarla montaj olanağı

Antenler ayrı ayrı sipariş verilmelidir.

### Daha fazla bilgi

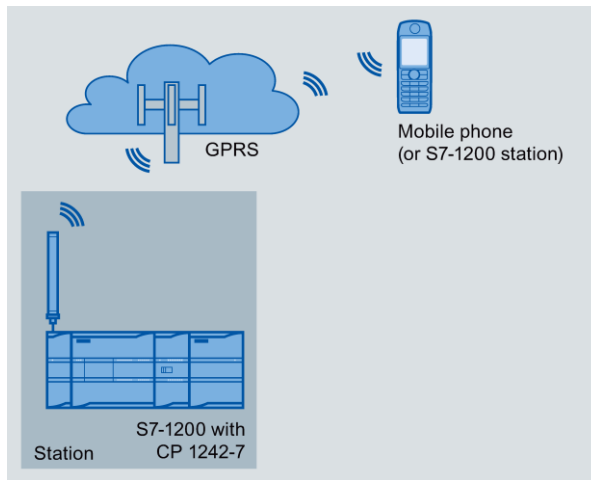
Cihaz kılavuzu ayrıntılı bilgi içermektedir. Aşağıdaki giriş ID si altında bunu Siemens Endüstriyel Otomasyon Müşteri Destek sayfalarında internette bulabilirsiniz:

23119005 (<http://support.automation.siemens.com/WW/view/en/23119005>)

### 12.6.5 Telekontrol için yapılandırma örnekleri

Aşağıda, bir CP 1242-7 ile istasyonlar için çeşitli yapılandırma örnekleri bulacaksınız.

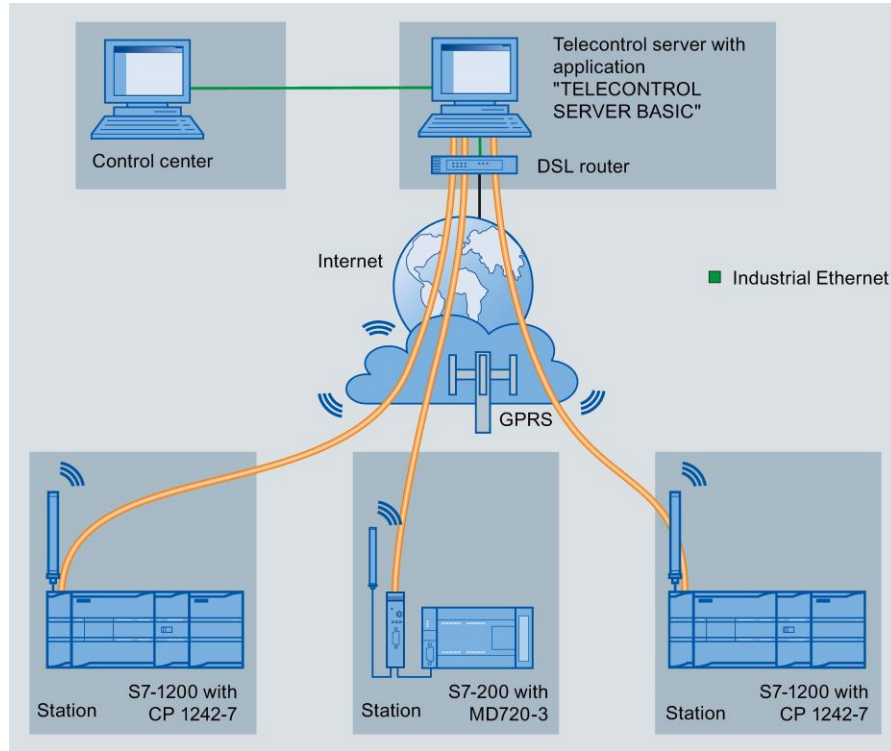
### SMS ile mesaj gönderme



Şekil 12-3 S7-1200 istasyonundan SMS ile mesajları gönderme

Bir CP 1242-7'ye sahip bir SIMATIC S7-1200, bir cep telefonu ya da yapılandırılmış bir S7-1200 istasyonuna SMS ile mesaj gönderebilir.

## Bir kontrol merkezi tarafından Telekontrol



Şekil 12-4 S7 -1200 istasyonları ve kontrol merkezi arasındaki haberleşme

Telekontrol uygulamalarında, bir CP 1242-7'ye sahip SIMATIC S7 -1200 istasyonları GSM şebekesi ve internet üzerinden bir kontrol merkezi ile haberleşir. " TELECONTROL SERVER BASIC " (TCSB) uygulaması, master istasyonundaki telekontrol sunucusu üzerine kurulur. Bu aşağıdaki kullanım durumlarına neden olur:

- İstasyon ve kontrol merkezi arasındaki telekontrol haberleşmesi

Bu kullanım durumunda, sahadan gelen veri, GSM şebekesi ve internet üzerinden master istasyonundaki telekontrol sunucusu istasyonlar tarafından gönderilir. Telekontrol sunucusu uzak istasyonları izlemek için kullanılır.

- Bir istasyon ile OPC istemcisine sahip bir kontrol odası arasında haberleşme

İlk durumda olduğu gibi, istasyonlar telekontrol sunucusu ile iletişim kurarlar. Entegre OPC sunucusunu kullanarak, kontrol odasının OPC istemcisi ile telekontrol sunucusu veri alışverişi yapar.

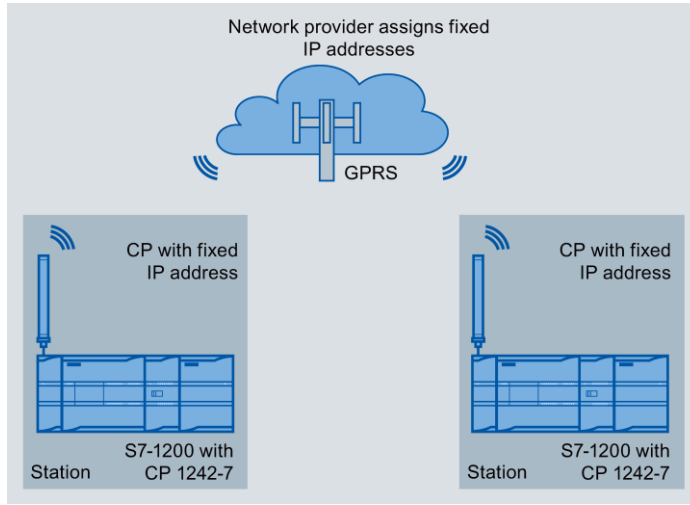
Örneğin TCSB, WinCC'ye sahip bir kontrol merkezi bilgisayarına yüklendiğinde OPC istemci ve telekontrol sunucusu, tek bir bilgisayar üzerinde yer alabilir.

- Bir kontrol merkezi üzerinden istasyonlar arası haberleşme

İstasyonlar arası haberleşme, CP 1242-7 ile donatılmış S7 istasyonları ile mümkündür.

İstasyonlar arası haberleşmeyi sağlamak için telekontrol sunucusu, gönderen istasyondaki mesajları alıcı istasyonuna iletir.

## İstasyonlar arası doğrudan haberleşme



Şekil 12-5 İki S7-1200 istasyonu arasında doğrudan haberleşme

Bu konfigürasyonda, iki SIMATIC S7-1200 istasyonu, GSM şebekesi üzerinden CP 1242-7 kullanarak birbiriyle doğrudan iletişim kurar. Her CP 1242-7'nin sabit bir IP adresi vardır. GSM şebeke sağlayıcınızın ilgili servisi buna izin vermelidir.

## GPRS üzerinden TeleServis

GPRS üzerinden TeleServis içinde, STEP 7'nin kurulu olduğu bir mühendislik istasyonu, S7-1200'de CP 1242-7 ile GSM şebekesi ve internet üzerinden iletişim kurar.

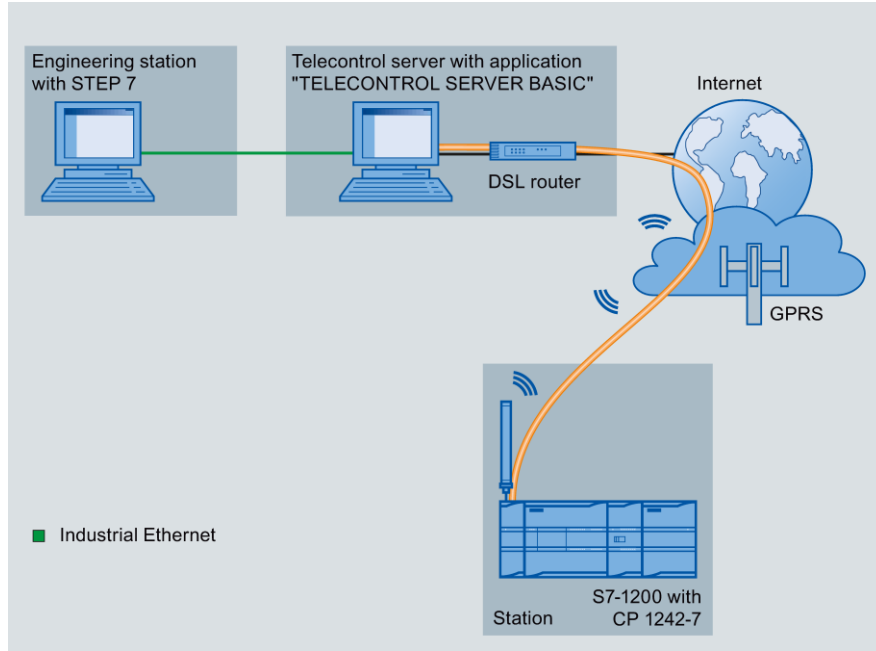
Bir güvenlik duvarı normalde dışarıdan gelen bağlantı istekleri için kapalı olduğundan, uzak istasyon ve mühendislik istasyonu arasında bir geçiş istasyonu gereklidir. Yapılandırmada hiçbir Telekontrol sunucusu yoksa bu geçiş istasyonu, bir TeleServis geçidi ya da bir tele sunucu olabilir.

## Telekontrol sunucuya sahip Teleservis

Bağlantı telekontrol sunucusu üzerinden çalışır.

- Mühendislik istasyonu ve uzak sunucu Intranet (LAN) veya Internet üzerinden bağlanır.
- Telekontrol sunucu ve uzak istasyon internet üzerinden ve GSM şebekesi üzerinden bağlanır.

Mühendislik istasyonu ve uzak sunucu aynı bilgisayar olabilir; diğer bir deyişle, STEP 7 ve TCSB aynı bilgisayarda yüklüdür.

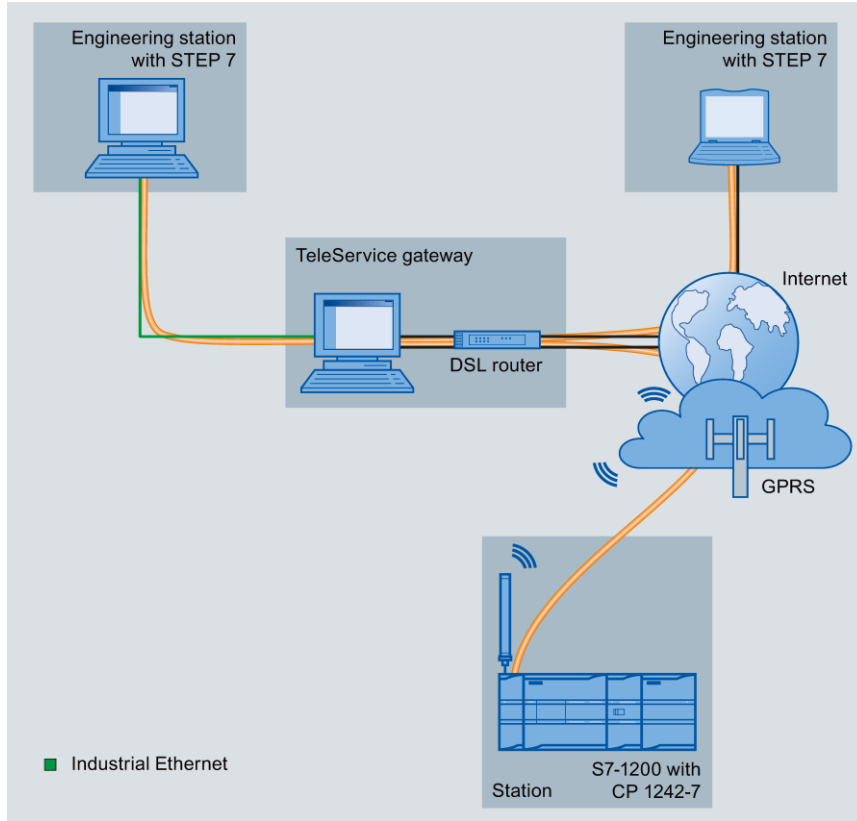


Şekil 12-6 Telekontrol sunucusu bir konfigürasyonda GPRS üzerinden TeleServis

### Bir telekontrol sunucusu olmadan TeleServis

Bağlantı TeleServis geçidi üzerinden çalışır.

Mühendislik istasyonu ve TeleServis geçidi arasındaki bağlantı, bir LAN veya internet üzerinden yerel olabilir.



Şekil 12-7 TeleServis geçitli bir konfigürasyonda GPRS üzerinden TeleServis

## Teleservis haberleşme (SMTP email)

### 13.1 TM\_Mail (email gönderme) komutu

Çizelge 13- 1 TM\_MAIL komutu

LAD / FBD	SCL	Açıklama																		
<div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">"TM_MAIL_DB"</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th colspan="2" style="text-align: center;">TM_MAIL</th> </tr> </thead> <tbody> <tr> <td>EN</td> <td>ENO</td> </tr> <tr> <td>REQ</td> <td>BUSY</td> </tr> <tr> <td>ID</td> <td>DONE</td> </tr> <tr> <td>TO_S</td> <td>ERROR</td> </tr> <tr> <td>CC</td> <td>STATUS</td> </tr> <tr> <td>SUBJECT</td> <td></td> </tr> <tr> <td>TEXT</td> <td></td> </tr> <tr> <td>ATTACHMENT</td> <td></td> </tr> </tbody> </table> </div>	TM_MAIL		EN	ENO	REQ	BUSY	ID	DONE	TO_S	ERROR	CC	STATUS	SUBJECT		TEXT		ATTACHMENT		<pre>"TM_MAIL_DB" (   REQ:=_bool_in_,   ID:=_int_in_,   TO_S:=_string_in_,   CC:=_string_in_,   SUBJECT:=_string_in_,   TEXT:=_string_in_,   ATTACHMENT:=_variant_in_,   BUSY=&gt;_bool_out_,   DONE=&gt;_bool_out_,   ERROR=&gt;_bool_out_,   STATUS=&gt;_word_out_,);</pre>	<p>TM_MAIL komutu CPU Endüstriyel Ethernet bağlantısı aracılığıyla TCP / IP üzerinden SMTP (Basit Posta Aktarım Protokolü) kullanarak bir e-posta mesajı gönderir. Ethernet-tabanlı internet bağlantısının mevcut olmadığı durumlarda, opsiyonel Teleservis adaptör, telefon kara hatları ile bağlantı için kullanılabilir. TM_MAIL asenkron olarak yürütülür ve iş çoklu TM_MAIL çağrılar boyunca uzanır. TM_MAIL i aradığınızda, bir kopya DB atanması gerekir. <b>Kopya DB'nin tutucu niteliği ayarlanmış olmamalıdır.</b> Bu, CPU nun STOP'tan RUN'a geçişinde kopya DB'nin başlatılmasını ve yeni bir TM_MAIL işleminin tetiklenebilir olmasını sağlar.</p>
TM_MAIL																				
EN	ENO																			
REQ	BUSY																			
ID	DONE																			
TO_S	ERROR																			
CC	STATUS																			
SUBJECT																				
TEXT																				
ATTACHMENT																				

1 KOMUT yerleştirdiğinizde, STEP 7 otomatik olarak kopya DB oluşturur.

Giriş parametresi REQ' de, 0 dan 1'e pozitif bir kenar değişikliği ile bir e-posta göndermeye başlayabilirsiniz. Aşağıdaki çizelge, BUSY DONE ve ERROR arasındaki ilişkiyi gösterir. Ardışık Aramalarda bu parametreleri değerlendirerek TM\_MAIL yürütmesinin ilerlemesini izleyebilir ve tamamlanmayı algılayabilirsiniz.

BUSY çıkış parametrelerinin durumu 1'den 0'a değiştiğinde DONE, ERROR, STATUS ve SFC\_STATUS Çıkış parametreleri yalnızca tek bir çevrim için geçerlidir. Sonraki program yürütme çevrimlerindeki durum değişikliklerini algılayabilmeniz için program mantığınız geçici çıkış durum değerlerini kaydeder.

Çizelge 13- 2 Done, Busy ve Error parametrelerinin etkileşimi

DONE	BUSY	ERROR	Açıklama
İlgisiz	1	İlgisiz	İş devam ediyor.
1	0	0	İş başarılı bir şekilde tamamlandı.
0	0	1	İş bir hata ile sonlandırıldı. Hatanın nedeni için STATUS parametrelerine bakınız.
0	0	0	Devam eden iş yok

## Teleservis haberleşme (SMTP email)

### 13.1 TM\_Mail (email gönderme) komutu

CPU, TM\_MAIL aktif iken STOP moduna değiştirilirse, o zaman e-posta sunucusu için haberleşme bağlantısı sonlandırılır. Endüstriyel Ethernet veri yolunda CPU iletişimde sorunlar meydana gelirse, e-posta sunucusuyla olan haberleşme bağlantısı kaybolur. Bu durumlarda, gönderme süreci askıya alınır ve e-posta alıcıya ulaşmaz.

#### DİKKAT

##### Kullanıcı programlarını değiştirme

Program bloklarını silme ve değiştirme, TM\_MAIL için çağrılar veya TM\_MAIL' in kopya DB'leri için çağrılar, programı blokları arasındaki bağlantıyı koparabilir. Bağlantılı program bloklarını korumayı başaramazsanız, TPC / IP haberleşme fonksiyonları tanımsız bir duruma girebilir bu da muhtemelen maddi hasar ile sonuçlanabilir. Değiştirilmiş bir program bloğunu aktardıktan sonra, bir CPU yeniden başlatma (sıcak) veya soğuk başlatma yapmak zorunda kalacaktır.

Program blokları bağlantısını koparmaktan kaçınmak için yalnızca aşağıdaki durumlarda TM\_MAIL çağrılarını doğrudan etkileyen kullanıcı programınızın parçalarını değiştiriniz:

- STOP modda CPU
- email gönderilmiyor (REQ ve BUSY = 0)

#### Veri tutarlılığı

İşlem başladığında giriş parametreleri ADDR\_MAIL\_SERVER okunur. Geçerli çalışma tamamlanana kadar ve yeni bir TM\_MAIL işlemi başlayana kadar yeni bir değer etki etmez.

Buna karşılık, WATCH\_DOG\_TIME, to\_s, CC, FROM, SUBJECT, TEXT, ATTACHMENT, USERNAME ve PASSWORD parametreleri, TM\_MAIL yürütülmesi sırasında okunur ve sadece iş bittiğinde değişebilir (BUSY = 0).

#### Çevirmeli bağlantı: TS adaptör IE parametrelerini yapılandırma

İnternet Servis Sağlayıcınızın çevirmeli sunucusuna bağlanmak için giden çağrılara yönelik Teleservis adaptörü IE parametrelerini yapılandırmanız gerekir. Çağrıyı " on demand " öznitelik olarak ayarlarsanız, daha sonra bağlantı sadece bir e-posta gönderildiğinde kurulur. Analog modem bağlantısında, bağlantı işlemi için daha fazla zaman (yaklaşık bir dakika daha uzun) gereklidir. WATCH\_DOG\_TIME değerinde, ekstra zaman ilave etmelisiniz.

Çizelge 13- 3 Parametreler için veri tipleri

Parametre ve tipi	Veri tipleri	Açıklama	
REQ	IN	Bool	Düşük-yüksek (pozitif kenar) sinyali çalışmayı başlatır.
ID	IN	Int	Bağlantı tanımlayıcı: TCON, TDISCON, TSEND ve TRCV komutlarının ID parametrelerine bakınız Kullanıcı programında bu komutun herhangi bir ek kopyası için kullanılmayan bir numara kullanılmalıdır.
TO_S	IN	String	Alıcı adresleri: maksimum uzunluğu 240 karakter olan STRING veri
CC	IN	String	CC Alıcı adreslerine kopyalar (isteğe bağlı): maksimum 240 karakter uzunluğunda STRING veri:



Parametre ve tipi		Veri tipleri	Açıklama
SUBJECT	IN	String	email konu ismi: Maksimum 240 karakter uzunluğunda STRING veri
TEXT	IN	String	(Opsiyonel) e-posta metin mesajı: maksimum 240 karakter uzunluğunda STRING veri. Bu parametre boş bir string ise, e-posta, mesaj metni olmadan gönderilecektir.
ATTACHMENT	IN	Variant	e-posta eki verilerine işaretçi: Maksimum 65534 bayt uzunlukta Bayt, word veya double word veriler: Hiçbir değer atanmamış ise, e-posta, eksiz gönderilir.
DONE	OUT	Bool	<ul style="list-style-type: none"> <li>0 – İş henüz başlatılmadı veya hala yürütülüyor.</li> <li>1 – İş hatasız yürütüldü.</li> </ul>
BUSY	OUT	Bool	<ul style="list-style-type: none"> <li>0 - İşlem devam etmiyor</li> <li>1- İşlem devam ediyor</li> </ul>
ERROR	OUT	Bool	Son istek bir hata ile sona erdirildikten sonra ERROR biti, bir tarama için TRUE olur. STATUS parametrelerindeki hata kodu değeri yalnızca ERROR = 1 olan tek çevrim esnasında geçerli olur.
STATUS	OUT	Word	TM_MAIL komutunun değerini veya hata bilgisini döndürür
ADDR_MAIL_SERVER	<sup>1</sup> Static	Dword	<p>Posta sunucusunun IP adresi: Her IP adresi bölümünü, iki adet 4-bit onaltılık karakterden oluşan bir sekizli (octet) olarak atamanız gerekir. IP adresi bölümü, onaltılık değer A ya eşit olan ondalık değere 10 eşitse, o zaman o sekizli için "0A" girmeniz gerekir.</p> <p>Örneğin: IP adresi = 192.168.0.10 ADDR_MAIL_SERVER = DW#16#C0A8000A, burada:</p> <ul style="list-style-type: none"> <li>192 = 16#C0,</li> <li>168 =16#A8</li> <li>0 = 16#00</li> <li>10 = 16#0A</li> </ul>
WATCH_DOG_TIME	<sup>1</sup> Static	Time	<p>SMTP bağlantı başlangıcından itibaren SMTP iletim sonuna kadar olan ve tüm SMTP işlemini tamamlamak amacıyla TM_MAIL için izin verilen maksimum süre. Bu süre aşılsa, TM_MAIL yürütmesi bir hata ile sona erer.</p> <p>TM_MAIL sona erene kadar ve hata yayınlanana kadarki gerçek zaman gecikmesi, bağlantı kesme işlemi için gerekli olan ilave zaman nedeniyle, WATCH_DOG_TIME ı aşabilir.</p> <p>İlk başta 2 dakikalık bir zaman ayarlamamız gerekir. Bu süre bir ISDN telefon bağlantısı için daha küçük olabilir.</p>
USERNAME	<sup>1</sup> Static	String	E-posta hesabı kullanıcı adı: maksimum uzunluğu 180 karakter olan STRING veri.
PASSWORD	<sup>1</sup> Static	String	Posta sunucu şifresi: maksimum uzunluğu 180 karakter olan STRING veri.

## Teleservis haberleşme (SMTP email)

### 13.1 TM\_Mail (email gönderme) komutu

Parametre ve tipi	Veri tipleri	Açıklama	
FROM	<sup>1</sup> Static	String	Gönderici adresi: maksimum uzunluğu 240 karakter olan STRING
SFC_STATUS	<sup>1</sup> Static	Word	Çağrılan haberleşme bloklarının yürütme durum kodu

<sup>1</sup> Bu parametrelerin değerleri TM\_MAIL her çağrılmasında değiştirilmez. Değerler TM\_MAIL kopya veri bloğu içinde atanır ve TM\_MAIL in ilk çağrılışında, sadece bir kez başvurulur.

### SMTP kimlik doğrulaması

TM\_MAIL, SMTP AUTH LOGIN kimlik doğrulama yöntemini destekler. Bu kimlik doğrulama yöntemi hakkında bilgi için posta sunucusu el kitabı veya internet servis sağlayıcınızın web sitesine bakınız.

AUTH LOGIN kimlik doğrulama yöntemi, posta sunucusuna bağlanmak için TM\_MAIL USERNAME ve PASSWORD parametrelerini kullanır. Kullanıcı adı ve şifrenin, bir e-posta sunucusundaki bir e-posta hesabına önceden ayarlanması gerekir.

USERNAME parametreleri için hiçbir değer atanmazsa, AUTH LOGIN kimlik doğrulama yöntemi kullanılmaz ve e-posta, kimlik doğrulaması olmadan gönderilir.

### TO\_S:, CC: ve FROM: parametreleri

TO\_S:, CC: ve FROM parametreleri: string'lerdir, aşağıdaki örneklerde gösterildiği gibi:

TO: <wenna@mydomain.com>, <ruby@mydomain.com>,

CC: <admin@mydomain.com>, <judy@mydomain.com>,

FROM: <admin@mydomain.com>

Bu karakter string'leri girilirken aşağıdaki kurallar kullanılmalıdır:

- "TO:", "CC:" ve "FROM:" karakterleri kolon karakteri dahil girilmelidir.
- Bir boşluk karakteri ve bir küçük işareti "<" her adresten önce gelmelidir. Örneğin "TO:" ve <e-posta adresi> arasında bir boşluk karakteri olmalıdır.
- Bir büyüktür işareti ">" her adresten sonra girilmelidir.
- Bir virgöl karakteri "," TO\_S: ve CC: adresleri için her e-posta adresinden sonra girilmelidir. Tek e-posta adresinden sonra virgöl gereklidir. Örneğin "TO:<e-posta Adresi> ,".
- Sonunda hiçbir virgöl olmadan FROM: giriş için Sadece bir e-posta adresi kullanılabilir.

Çalışma zamanı modu ve bellek kullanımı nedeniyle, TM\_MAIL TO\_S:, CC: ve FROM: veri üzerinde bir sözdizimi kontrolü gerçekleştirilmez. Yukarıda biçim kurallarına tam olarak uyulmazsa SMTP e-posta sunucu işlemi başarısız olur.

### STATUS ve SFC\_STATUS parametreleri

TM\_MAIL tarafından döndürülen yürütme koşulu kodları aşağıdaki gibi sınıflandırılabilir:

- W#16#0000: TM\_MAIL işlemi başarıyla tamamlandı
- W#16#7xxx: TM\_MAIL işleminin durumu
- W#16#8xxx: haberleşme cihazına ya da posta sunucusuna dahili çağrı yaparken bir hata

Aşağıdaki çizelgede, dahili olarak çağrılan haberleşme modüllerinden gelen hata kodları hariç TM\_MAIL yürütme koşulu kodları gösterilmektedir.

---

**Not****E-posta sunucu kuralları**

TM\_MAIL, sadece port 25 üzerinden SMTP kullanarak bir e-posta sunucusu ile iletişim kurabilir. Atanan port numarası değiştirilemez.

Çoğu IT bölümleri ve harici e-posta sunucuları artık virüs bulaşmış bir PC'nin sahte e-posta üretici haline gelmesini önlemek için port 25'i engeller.

SMTP üzerinden bir dahili posta sunucusuna bağlanabilir ve dahili sunucunun, harici bir posta sunucusuna Internet üzerinden e-posta atmak için gerekli olan mevcut güvenlik geliştirmelerini yönetmesine izin verebilir.

---

**Dahili e-posta sunucusu yapılandırma örneği**

Bir dahili posta sunucusu olarak Microsoft Exchange kullanıyorsanız, o zaman S7 -1200 PLC'ye atanan IP adresinden SMTP erişimi sağlayacak şekilde sunucuyu yapılandırabilirsiniz. Exchange Yönetim konsolunu yapılandırınız: Server configuration > Hub transport > Receive connectors > IP relay. Ağ sekmesinde , " bu IP adreslere sahip uzak sunuculara e-posta al " adında bir kutu vardır. Burası TM\_MAIL komutunu yürüten PLC'nin IP adresini koyduğunuz yerdir. Dahili Microsoft Exchange sunucusu ile bu tür bağlantılı için hiçbir kimlik doğrulama gerekli değildir.

**E-posta sunucusu yapılandırma**

TM\_MAIL, sadece port 25 haberleşmesi, SMTP ve AUTH LOGIN kimlik doğrulamaya (opsiyonel) izin veren bir e-posta sunucusu kullanabilir.

Uzak SMTP oturum açmayı kabul etmek için bir uyumlu e-posta sunucu hesabını yapılandırınız. e -posta hesabınız ile bağlantınızın kimlik doğrulaması için kullanılan TM\_MAIL USERNAME ve PASSWORD karakter string'lere koymak amacıyla TM\_MAIL için kopya DB'yi düzenleyiniz.

## Teleservis haberleşme (SMTP email)

## 13.1 TM\_Mail (email gönderme) komutu

Çizelge 13- 4 Durum kodları

STATUS (W#16#...):	SFC_STATUS (W#16#...):	Açıklama
0000	-	TM_MAIL işlemi hatasız tamamladı. Bu sıfır STATUS kodu, e-postanın gerçekten (bu çizelgeden sonar verilen nottaki ilk ögeye bakınız) gönderildiğini garanti etmez.
7001	-	TM_MAIL aktif (BUSY = 1).
7002	7002	TM_MAIL aktif (BUSY = 1).
8xxx	xxxx	TM_MAIL işlemi, dahili haberleşme komut çağrılarında bir hata ile tamamlandı. SFC_STATUS parametreleri hakkında daha fazla bilgi için, temel PROFINET açık kullanıcıli haberleşme komutlarının STATUS parametreleriyle ilgili açıklamalarına bakınız.
8010	xxxx	Bağlanamadı: SFC_STATUS parametreleri hakkında daha fazla bilgi için, TCON komutunun STATUS parametrelerine bakınız.
8011	xxxx	Veri gönderme hatası: SFC_STATUS parametreleri hakkında daha fazla bilgi için, TSEND komutunun STATUS parametrelerine bakınız.
8012	xxxx	Veri alma hatası: SFC_STATUS parametreleri hakkında daha fazla bilgi için, TRCV komutunun STATUS parametre açıklamalarına bakınız.
8013	xxxx	Bağlanamadı: SFC_STATUS parametrelerini değerlendirmekle ilgili daha fazla bilgi için, TCON ve TDISCON komutlarının STATUS parametre açıklamalarına bakınız.
8014	-	Bağlanamadı: Yanlış bir posta sunucu IP adresi (ADDR_MAIL_SERVER) girmiş olabilirsiniz ya da Bağlantı için çok az zaman (WATCH_DOG_TIME) vardır. Bu CPU nun ağ bağlantısının olmaması veya CPU yapılandırmasının yanlış olması mümkündür.
8015	-	ATTACHMENT parametreleri için geçersiz bir işaretçi: Bir veri tipi ve uzunluk atanan bir değişken işaretçi kullanınız. Örneğin, "P # DB.DBX0.0" yanlış ve "P # DB.DBX0.0 bayt 256" doğrudur.
82xx, 84xx, 85xx	-	Hata mesajı posta sunucusundan gelir ve SMTP protokolünün "8" hata numarasına karşılık gelir. Bu çizelgeden sonra verilen notta ikinci ögeye bakınız.
8450	-	İşlem yapılmıyor: Posta kutusu mevcut değildir; Lütfen daha sonra tekrar deneyiniz.
8451	-	İşlem iptal edildi: Yerel hata işleniyor, daha sonra tekrar deneyiniz.
8500	-	Komut sözdizimi hatası: Nedeni, e-posta sunucusunun LOGIN kimlik doğrulama işlemini desteklememesi olabilir. TM_MAIL parametrelerini kontrol ediniz. Kimlik doğrulaması olmadan bir e-posta göndermeyi deneyin. Boş bir string ile USERNAME parametrelerini değiştirmeyi deneyiniz.
8501	-	Sözdizimi hatası: Yanlış parametre veya argüman; TO_S veya CC parametrelerinde yanlış adres yazmış olabilirsiniz.
8502	-	Komut uygulanamadı veya bilinmiyor: Girişlerinizi özellikle FROM parametrelerini kontrol ediniz. Belki de bu eksik ve "@" veya "." karakterlerini çıkardınız.
8535	-	SMTP kimlik doğrulaması eksik. Hatalı kullanıcı adı veya şifre girmiş olabilirsiniz.
8550	-	Posta sunucusuna ulaşılamıyor, ya da hiçbir erişim hakkınız yok. Hatalı kullanıcı adı veya şifre girmiş olabilirsiniz veya posta sunucusu oturum açma erişimini desteklemiyor. Bu hatanın bir başka nedeni TO_S veya CC parametrelerinde "@" karakterinden sonra alan adı girişi hatalı olabilir.
8552	-	İşlem iptal edildi: Ayrılan bellek boyutu aşıldı; Lütfen daha sonra tekrar deneyin.
8554	-	İletim başarısız oldu: Daha sonra tekrar deneyiniz.

**Not****Olası bildirilmemiş e-posta iletim hataları**

- Bir alıcı adresinin yanlış girilmesi, TM\_MAIL için bir STATUS hatası oluşturmaz. Bu durumda, ilave alıcıların (Doğru e-posta adresleri ile), e-postayı alacağını hiçbir garantisi yoktur.
  - SMTP hata kodları hakkında daha fazla bilgi, internet üzerinde veya posta sunucusu için hata belgelerinde bulunabilir. Ayrıca posta sunucusundan gelen son hata mesajını okuyabilirsiniz. Hata mesajı TM\_MAIL için kopya DB buffer1 parametrelerinde saklanır.
-

*Teleservis haberleşme (SMTP email)*

*13.1 TM\_Mail (email gönderme) komutu*

---

## Çevrimiçi ve tanılama araçları

### 14.1 Durum LED'leri

CPU ve I/O modülleri, modülün veya I/O'un çalışma durumu hakkında bilgi sağlamak için LED'ler kullanır.

#### Bir CPU'daki durum LED'leri

CPU aşağıdaki durum göstergelerini sağlar:

- STOP/RUN
  - Dolu sarı STOP modunu gösterir
  - Dolu yeşil RUN modunu gösterir
  - Yanıp sönme (alternatif yeşil ve sarı ) CPU'nun STARTUP modunda olduğunu gösterir
- ERROR
  - Yanıp sönen kırmızı örneğin CPU'daki bir iç hatayı, bellek kartıyla ilgili bir hatayı ya da bir yapılandırma hatasını (eşleşmeyen modüller) gösterir
  - Arızalı durum:
    - Dolu kırmızı kusurlu donanımı gösterir
    - Kusur firmware de tespit edilirse, tüm LED'ler yanıp söner
- Bir bellek kartı taktığınızda MAINT ( Bakım ) yanıp söner. CPU sonra STOP moduna geçer. CPU STOP moduna geçtikten sonra, bellek kartının değerlendirmesini başlatmak için aşağıdaki işlevlerden birini gerçekleştiriniz:
  - CPU yu RUN moduna geçiriniz
  - Belleği sıfırlayınız ( MRES )
  - CPU yu güç çevrimine tabi tutunuz

Ayrıca LED'lerin durumunu belirlemek için LED komutunu (Sayfa 346) kullanabilirsiniz.

Çizelge 14- 1 Bir CPU için durum LED' leri

Açıklama	STOP/RUN Sarı / Yeşil	ERROR Kırmızı	MAINT Sarı
Güç kesik	Kapalı	Kapalı	Kapalı
Yol verme, oto-test veya firmware güncelleme	Yanıp sönüyor (sarı ve yeşil şeklinde değişiyor)	-	Kapalı
Dur modu	Açık (sarı)	-	-
Çalışma modu	Açık yeşil)	-	-
Bellek kartını çıkart	Açık (sarı)	-	Yanıp sönüyor
Hata	Açık (sarı veya yeşil)	Yanıp sönüyor	-

Açıklama	STOP/RUN Sarı / Yeşil	ERROR Kırmızı	MAINT Sarı
Bakım isteniyor <ul style="list-style-type: none"> <li>Zorlamalı I/O</li> <li>Bataryanın değişmesi gerek (batarya kartı takılı ise)</li> </ul>	Açık (sarı veya yeşil)	-	Açık
Kusurlu donanım	Açık (sarı)	Açık	Kapalı
LED test veya kusurlu CPU firmware	Yanıp sönüyor (sarı ve yeşil şeklinde değişiyor)	Yanıp sönüyor	Yanıp sönüyor
Bilinmeyen veya uyumsuz CPU yapılandırma versiyonu	Açık (sarı)	Yanıp sönüyor	Yanıp sönüyor

### Not

#### " Bilinmeyen ya da uyumsuz CPU yapılandırma sürümü" hatası

S7-1200 V3.0 programını S7-1200 V4.0 CPU' ya indirme girişimi CPU hatasına neden olur ve CPU tanılama arabelleğinde ilgili bir hata mesajı görüntüler. Geçersiz bir sürüm program transfer kartı (Sayfa 127) kullanarak bu duruma ulaşırsanız, kartı çıkarınız, STOP'tan RUN'a geçiniz, bir belleği sıfırlayın (MRES) veya güç çevrimini gerçekleştiriniz. Geçersiz bir program indirmesiyle bu duruma ulaşırsanız, CPU yu fabrika ayarlarına (Sayfa 789) sıfırlayınız. CPU yu hata durumundan kurtardıktan sonra geçerli V4.0 CPU programı indirebilirsiniz.

CPU aynı zamanda PROFINET haberleşme durumunu gösteren iki LED sağlar. PROFINET LED'lerini görüntülemek için alt terminal kapağını açınız.

- Bağlantı (yeşil), Başarılı bir bağlantı yapıldığını göstermek için yanar
- Rx / Tx (sarı), iletim faaliyetini göstermek için yanar

CPU ve her dijital sinyal modülü (SM), dijital giriş ve çıkışların her biri için bir I/O Kanal LED i sağlar. I/O Kanal (yeşil) münferit giriş veya çıkış durumunu göstermek için açık veya kapalı olur.


### Bir ölümcül hatanın ardından S7-1200 davranışı

CPU firmware, ölümcül bir hata tespit ederse bu bir kusur-modunu yeniden başlatmayı dener ve başarılı olursa, STOP/RUN, ERROR ve MAINT LED'lerini sürekli yakıp söndürerek arızalı modu bildirir. Kullanıcı programı ve donanım yapılandırması, arıza modunun yeniden başlatılmasının ardından yüklenmez.

CPU, arıza modunun yeniden başlatılmasını başarıyla tamamlarsa, CPU ve sinyal kartı çıkışları 0'a ayarlanır ve merkezi şasi sinyal modüllerinin çıkışları ve dağıtılmış I/O yapılandırılmış "CPU STOP Reaksiyonu"na ayarlanır.



Kusur modunu yeniden başlatma başarısız olursa, (örneğin, bir donanım arızası nedeniyle), STOP ve ERROR LED'leri AÇIK ve MAINT LED i KAPALI olur.

 <b>UYARI</b>
<p><b>Kusurlu durumda çalışma garanti edilemez</b></p> <p>Kontrol cihazları, kontrol edilen ekipmanın beklenmeyen çalışmasıyla sonuçlanan, güvensiz bir durumda arızalanabilir. Bu tarz beklenmeyen çalışmalar ölüme veya personelin ciddi şekilde yaralanmasına ve/veya ekipman hasarına neden olabilir.</p> <p>Acil durdurma fonksiyonunu, elektromekanik kilitlemeyi veya PLC den bağımsız diğer yedek emniyet önlemlerini kullanınız.</p>

### Bir SM üzerindeki durum LED'leri

Buna ek olarak, her bir dijital SM, modülün durumunu gösteren bir DIAG LED sağlamaktadır:

- Yeşil, modülün çalışır olduğunu gösterir
- Kırmızı, modülün arızalı veya çalışmadığını gösterir

Her bir analog SM, analog giriş ve çıkışların her biri için bir I/O Kanal LED sağlar.

- Yeşil, kanalın konfigüre edildiğini ve aktif olduğunu belirtir
- Kırmızı, münferit analog giriş veya çıkışın hata durumunu gösterir

Buna ek olarak her bir analog SM, modülün durumunu gösteren bir DIAG LED sağlamaktadır:

- Yeşil, modülün çalışır olduğunu gösterir
- Kırmızı, modülün arızalı veya çalışmadığını gösterir

SM, modülü besleyen gücün varlığını veya yokluğunu tespit eder (gerekirse, saha tarafındaki güç)


Çizelge 14- 2 Bir sinyal modülü (SM) için durum LED'leri

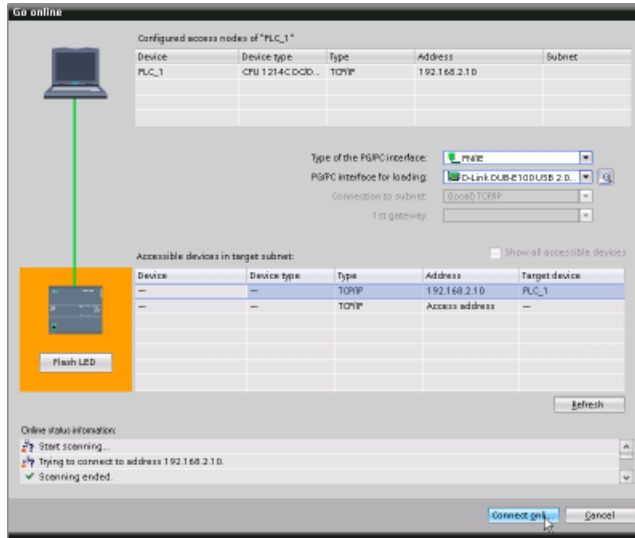
Açıklama	DIAG (Kırmızı / Yeşil)	I/O Kanalı (Kırmızı / Yeşil)
Saha tarafındaki güç kapalı	Yanıp sönen kırmızı	Yanıp sönen kırmızı
Yapılandırılmamış veya güncelleme devam ediyor	Yanıp sönen yeşil	Kapalı
Modül hatasız yapılandırıldı	Açık (yeşil)	On (yeşil)
Hata durumu	Yanıp sönen kırmızı	-
I/O hatası (etkinleştirilmiş tanılamayla)	-	Yanıp sönen kırmızı
I/O hatası (devre dışı bırakılmış tanılamayla)	-	Açık (yeşil)

## 14.2 Çevrimiçi olma ve bir CPU bağlantısı

Yükleme programları ve proje mühendislik verilerinin yanı sıra aşağıdaki gibi faaliyetler için Programlama cihazı ve CPU arasında bir çevrimiçi bağlantı gereklidir:

- Kullanıcı programlarını test etme
- CPU çalışma modunun görüntülenmesi ve değiştirilmesi(Sayfa 792)
- CPU nun tarih ve saatinin görüntülenmesi ve ayarlanması(Sayfa 789)
- Modül bilgilerini görüntüleme
- Çevrimiçi ve çevrimdışı program bloklarının karşılaştırılması ve senkronize edilmesi (Sayfa 794)
- Program blokları yükleme ve indirme
- Tanıların ve tanılama arabelleğinin görüntülenmesi (Sayfa 793)
- Değerleri izleyerek ve değiştirerek kullanıcı programını test etmek için bir izleme çizelgesi kullanma(Sayfa 798)
- CPU da değerleri zorlamak için bir zorlama çizelgesini kullanma (Sayfa 801)

Yapılandırılmış bir CPU ile çevrimiçi bağlantı kurmak için Proje  Go online Gezinti ağacından CPU tıklayınız ve Proje görünümünden " Go Çevrimiçi " düğmesine tıklayınız:



İlk defa CPU ile çevrimiçi olunuyorsa, Go Çevrimiçi iletişim kutusundan özel PG / PC ara yüzü ve PG / PC ara yüz türünü, bu ara yüz üzerinde bulunan bir CPU ile çevrimiçi bağlantı kurmadan önce seçmelisiniz .

Programlama cihazınız artık CPU'ya bağlıdır. Turuncu renk çerçeveler çevrimiçi bağlantıyı gösterir. Şimdi Proje ağacı ve çevrimiçi araçlar görev kartından çevrimiçi & tanılama araçlarını kullanabilirsiniz.

## 14.3 PROFINET IO cihazınıza çevrimiçi bir isim atama

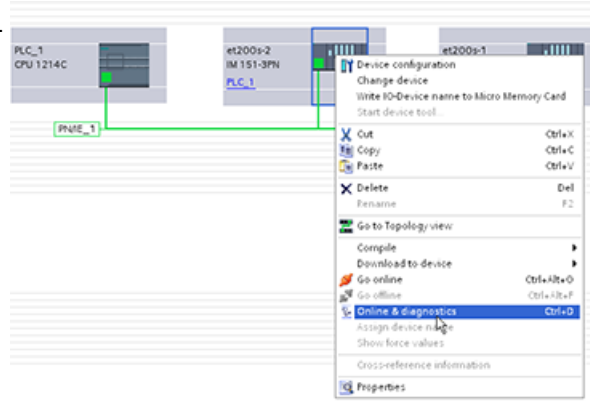
CPU ile bağlantı kurmadan önce PROFINET ağdaki cihazların atanmış bir adı olmalıdır. Cihazın ismi değiştirilecek ise veya henüz cihazlara bir isim atanmamışsa, PROFINET cihazlara isimleri atamak için "Cihazlar & ağlar" editörünü kullanınız.

Her PROFINET IO cihazı için, her iki STEP 7 projesindeki cihaza ve "Çevrimiçi ve tanılama" aracını kullanarak, PROFINET IO cihaz yapılandırma hafızasına (örneğin, bir ET200 S ara yüz modülü konfigürasyon belleği) aynı adı atamanız gerekir. Bir isim eksik ise veya her iki yerde de uymuyorsa, PROFINET IO veri alışverişi modu çalışmaz.

## Çevrimiçi ve tanılama araçları

### 14.3 PROFINET IO cihazınıza çevrimiçi bir isim atama

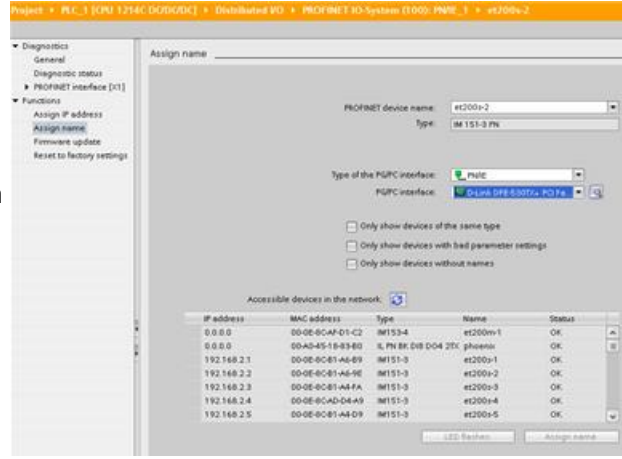
1. "Cihazlar & ağlar" editöründe, gerekli PROFINET IO cihaz üzerinde sağ tıklayınız ve "Çevrimiçi & tanılama" seçiniz.



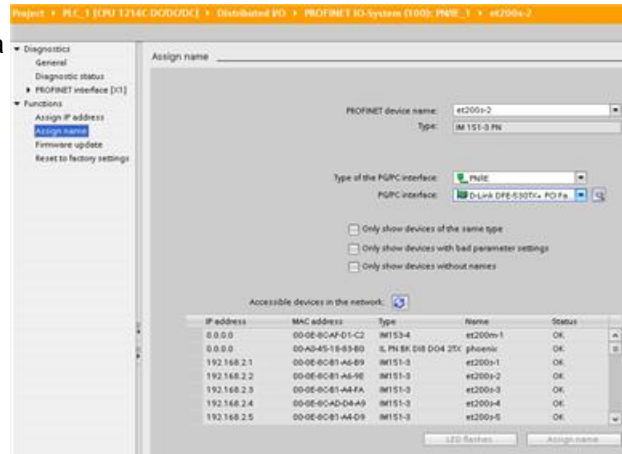
2. "Çevrimiçi & tanılama" iletişim kutusunda, aşağıdaki menü seçimlerini yapın:

- " Fonksiyonlar "
- " İsim atama "

Ağda PROFINET IO cihazların hepsini görüntülemek için "ağdaki Erişilebilir cihazlar" simgesine tıklayınız.



3. Görüntülenen listede, gerekli PROFINET IO cihazına tıklayınız ve PROFINET IO cihaz yapılandırma hafızasına isim yazmak için "İsim atama" butonuna tıklayınız.



## 14.4 IP adresini ve günün saatini ayarlama

Çevrimiçi CPU da, IP adresini (Sayfa 158) ve saati ayarlayabilirsiniz. Bir Çevrimiçi CPU için Proje ağacından "Çevrimiçi ve tanılama" eriştikten sonra, IP adresini görüntüleyebilir veya değiştirebilirsiniz. Ayrıca çevrimiçi CPU nun saat ve tarih parametrelerini görüntüleyebilir veya ayarlayabilirsiniz.



The image shows two screenshots of a web-based configuration interface. The left screenshot is titled 'Set time of day' and contains two sections: 'PG/PC time' and 'Module time'. The 'PG/PC time' section has a dropdown menu for '(GMT-05:00) Eastern Time (US & Canada)', a date selector set to 'September 01, 2009', and a time selector set to '09:47:00 AM'. The 'Module time' section has a date selector set to 'December 31, 1969', a time selector set to '07:42:56 PM', and a checked checkbox labeled 'Take from PG/PC'. An 'Apply' button is at the bottom right. The right screenshot is titled 'Assign IP address' and has a 'MAC address' field with the value '08-00-06-05-91-E1' and an 'Accessible devices' button. Below that are 'IP address' (192.168.0.1) and 'Subnet mask' (255.255.255.0) fields. There is a 'Use router' checkbox and a 'Router address' field with the value '0.0.0.0'. An 'Assign IP address' button is at the bottom.

### Not

Bu özellik yalnızca bir MAC adresi (henüz bir IP adresi atanmış değil) olan ya da fabrika ayarlarına sıfırlanan bir CPU için kullanılabilir.

## 14.5 Fabrika ayarlarına sıfırlama

Bir S7-1200 ü aşağıdaki koşullarda orijinal fabrika ayarlarına sıfırlayabilirsiniz:

- Bellek kartı CPU'ya yerleştirilmez.
- CPU nun Çevrimiçi bir bağlantısı vardır.
- CPU, STOP modunda.

### Not

CPU RUN modunda ise ve sıfırlama işlemini başlatırsanız, bir onay istemini kabul ettikten sonra bunu STOP moduna yerleştirebilirsiniz.

## Prosedür

Bir CPU yu fabrika ayarlarına sıfırlamak için şu adımları izleyiniz:

1. CPU nun Çevrimiçi ve Tanılama görünümünü açınız.
2. "Fonksiyonlar" klasöründeki "fabrika ayarlarına sıfırla" seçeneğini seçiniz.
3. Eğer IP adresini silmek istiyorsanız "IP adresini sıfırla" onay kutusunu veya IP adresini korumak istiyorsanız "IP adresini sakla" onay kutusunu seçiniz.
4. "Sıfırla" düğmesine tıklayınız.
5. "Tamam" ile onay istemini onaylayın.

## Sonuç

Modül, gerekirse STOP moduna geçer ve fabrika ayarlarına sıfırlanır:

- Çalışma belleği ve dahili yükleme belleği ve tüm işlenen alanlar silinir.
- Tüm parametreler varsayılan değerlerine sıfırlanır.
- Tanılama arabelleği temizlenir.
- Günün saati sıfırlanır.
- IP adresi, yaptığınız ayara dayalı olarak tutulur veya silinir. (MAC adresi sabittir ve asla değişmez.)

## 14.6 Firmware güncellemesi

STEP 7 çevrimiçi ve tanılama araçlarından bağlı CPU'nun firmware'ini güncelleyebilirsiniz.

Bir firmware güncellemesi yapmak için şu adımları izleyiniz:

1. Bağlı CPU nun Çevrimiçi ve Tanılama görünümünü açınız.
2. "Fonksiyonlar" klasöründeki "Firmware güncellemesi" seçeneğini seçiniz.
3. Gözet düğmesine tıklayınız ve firmware güncelleme dosyasını içeren konuma gidin. Bu, hizmet ve destek Web sitesinden (<http://www.siemens.com/automation/>) bir S7-1200 firmware güncelleme dosyasını (<http://support.automation.siemens.com/WW/view/en/34612486/133100>) indirdiğiniz sabit disk üzerindeki bir yer olabilir.
4. Sizin modülünüz ile uyumlu bir dosya seçiniz. Seçilen bir dosya için, Çizelge, uyumlu modülleri görüntüler.
5. "Güncellemeyi çalıştır" butonuna tıklayınız. Gerekirse CPU'nuzun çalışma modunu değiştirmek için, diyalogları izleyiniz.

firmware güncellemesi yüklenirken STEP 7 ilerleme diyaloglarını görüntüler.  
Tamamlandığında, bu yeni firmware ile modülü başlatmanız istenir.

---

### Not

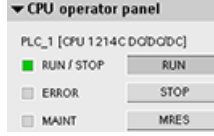
Yeni firmware ile modülü başlatmayı tercih etmezseniz, örneğin güç çevrimiyle modülü sıfırlayana kadar, önceki firmware etkin kalır. Yeni firmware, ancak modülü sıfırladıktan sonra aktif olur.

---

Ayrıca aşağıdaki ek yöntemlerden biri ile bir firmware güncellemesi yapabilirsiniz:

- Bellek kartını kullanarak (Sayfa 133)
- Web sunucusu "Modül Bilgisi" standart Web sayfasını (Sayfa 616) kullanarak

## 14.7 Çevrimiçi CPU için CPU operatör paneli



"CPU operatör paneli" Çevrimiçi CPU nun çalışma modunu (STOP veya RUN) görüntüler. Panel ayrıca CPU nun bir hatasının olup olmadığını ya da değerlerin zorlanıp zorlanmadığını gösterir.

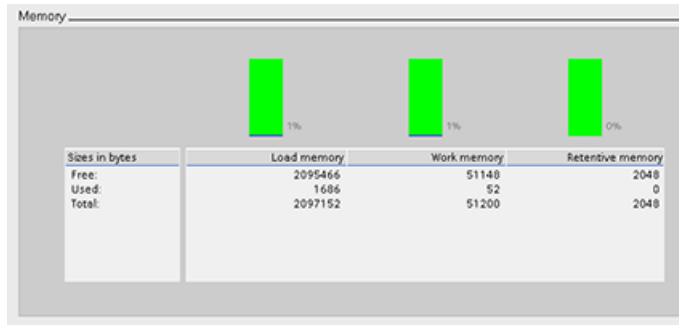
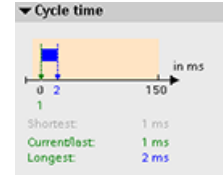
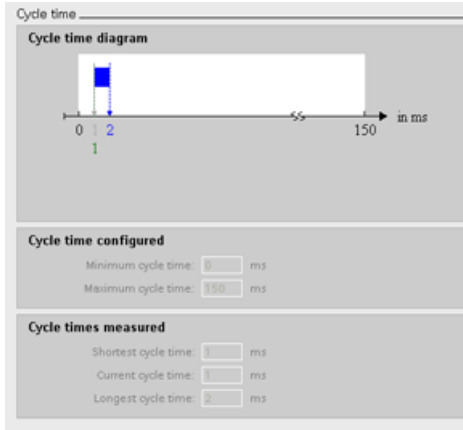
Bir Çevrimiçi CPU nun çalışma modunu değiştirmek için Çevrimiçi Araçlar görev kartının CPU işletim panelini kullanınız. CPU Çevrimiçi olduğunda Çevrimiçi Araçlar görev kartı erişilebilir olur.

## 14.8 Çevrim süresi ve bellek kullanımını izleme

Çevrimiçi CPU nun çevrim süresini ve bir bellek kullanımını izleyebilirsiniz.

Çevrimiçi CPU' ya bağlandıktan sonra, aşağıdaki ölçümleri görüntülemek için Çevrimiçi araçlar görev kartını açınız:

- Çevrim süresi
- Bellek kullanımı

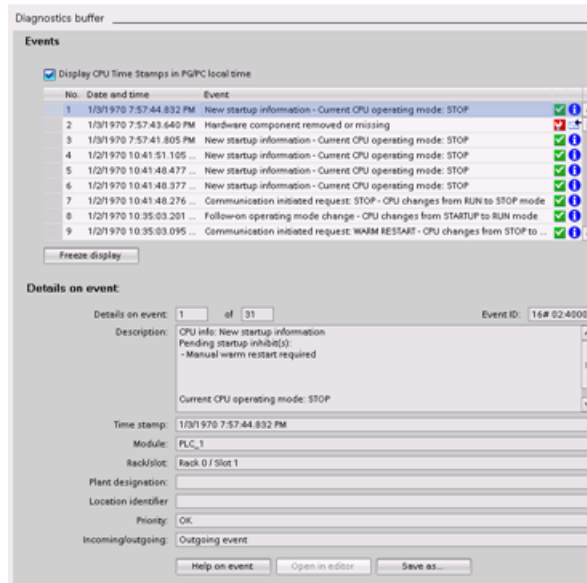




## 14.9 CPU'daki tanılama olaylarını görüntüleme

CPU'daki son faaliyeti incelemek için tanılama arabelleğini kullanınız. Tanılama arabelleği Proje ağacında bir Çevrimiçi CPU için "Çevrimiçi & Tanılama" dan erişilebilir olur. Bu aşağıdaki girdileri içerir:

- Tanılama olayları
- CPU çalışma modunda değişiklikler (STOP veya RUN için geçişler)



The screenshot shows the 'Diagnostics buffer' window. It contains a table of events with columns for 'No.', 'Date and time', and 'Event'. The events listed are:

No.	Date and time	Event
1	1/21/970 7:57:44.832 PM	New startup information - Current CPU operating mode: STOP
2	1/21/970 7:57:43.640 PM	Hardware component removed or missing
3	1/21/970 7:57:41.805 PM	New startup information - Current CPU operating mode: STOP
4	1/21/970 10:41:51.105 ...	New startup information - Current CPU operating mode: STOP
5	1/21/970 10:41:48.477 ...	New startup information - Current CPU operating mode: STOP
6	1/21/970 10:41:48.377 ...	New startup information - Current CPU operating mode: STOP
7	1/21/970 10:41:48.276 ...	Communication initiated request: STOP - CPU changes from RUN to STOP mode
8	1/21/970 10:35:03.201 ...	Follow-on operating mode change - CPU changes from STARTUP to RUN mode
9	1/21/970 10:35:03.095 ...	Communication initiated request: WARM RESTART - CPU changes from STOP to ...

Below the table, there is a 'Details on event' section for the selected event (Event ID: 16# 02-4000). The details include:

- Description: CPU info: New startup information  
Pending startup inhibit(s)  
- Manual warm restart required  
Current CPU operating mode: STOP
- Time stamp: 1/21/970 7:57:44.832 PM
- Module: PLC\_1
- Rack/slot: Rack 0 / Slot 1
- Plant designation: [empty]
- Location identifier: [empty]
- Priority: OK
- Incoming/outgoing: Outgoing event

İlk giriş en son olayı içerir. Tanılama arabelleğindeki her giriş, olayın kaydedildiği tarih ve saatini ve bir açıklamasını içerir.

Girişlerin maksimum sayısı CPU' ya bağlıdır. Maksimum 50 giriş desteklenmektedir.

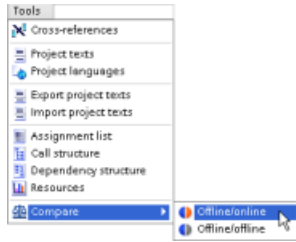
Tanılama arabelleği Sadece 10 en son olaylar saklanır

Sadece Tanılama arabelleğindeki son 10 olay kalıcı olarak saklanır. CPU yu Fabrika ayarlarına sıfırlama, girdileri silerek tanılama arabelleğini sıfırlar.

Ayrıca tanılama bilgilerini toplamak için GET\_DIAG komutunu (Sayfa 358) kullanabilirsiniz.

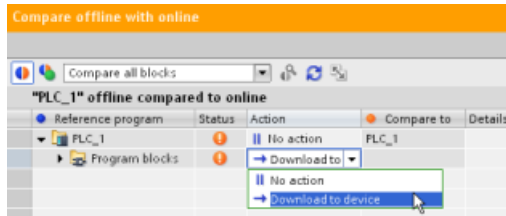
## 14.10 Çevrimdışı ve çevrimiçi CPU'ların karşılaştırılması

Projenizdeki kod blokları ile bir çevrimiçi CPU'daki kod bloklarını karşılaştırabilirsiniz. Projenizin kod blokları, çevrimiçi CPU kod bloklarına uymuyorsa, "karşılaştır" editörü proje kod bloklarını CPU'ya indirerek, ya da Çevrimiçi CPU da olmayan blokları projeden silerek çevrimiçi CPU ile projenizi senkronize etmenize olanak verir.



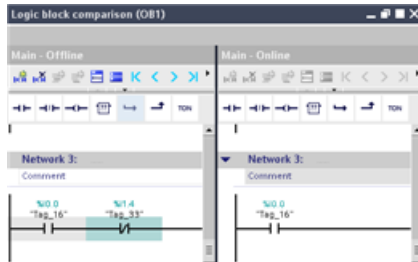
Projenizde CPU seçiniz.

"Karşılaştırma" editörünü açmak için "Çevrimdışı / Çevrimiçi karşılaştırma" komutunu kullanınız. (Komuta "Araçlar" menüsünden veya projenizde CPU'ya sağ tıklayarak erişin.)



Nesneyi silmek, hiçbir eylemde bulunmamak veya cihaza nesneyi indirmek gibi seçimleri seçmek için bir nesnedeki "Eylem" sütununa tıklayınız.

Kod bloklarını yüklemek için "Senkronla" düğmesini tıklayınız.



"karşılaştır" sütununda bir nesneyi sağ tıklayınız ve kod bloklarını yan yana göstermek için "detaylı karşılaştırmayı başlat" düğmesini seçiniz.

Detaylı karşılaştırma, Çevrimiçi CPU kod blokları ile projenizdeki CPU kod blokları arasındaki farklılıkları vurgular.

## 14.11 CPU değerlerini izleme ve değiştirme

STEP 7 CPU'nun izlenmesi için çevrimiçi araçlar sağlar:

- Bu etiketlerin geçerli değerlerini görüntüleyebilir veya izleyebilirsiniz. İzleme fonksiyonu programın sırasını değiştirmez. Bu fonksiyon, program sırası ve CPU'daki program verileri hakkında bilgileri size sunar.
- Ayrıca kullanıcı programının sırasını ve verilerini kontrol etmek için diğer işlevleri kullanabilirsiniz:
  - Kullanıcı programının nasıl tepki vereceğini görmek için çevrimiçi CPU etiketlerine yönelik değeri değiştirebilirsiniz.
  - Bir çevresel çıkışı (Q0.1:P ya da "Start":P gibi) belirli bir değere zorlayabilirsiniz.
  - STOP modundayken çıkışları etkinleştirebilirsiniz.

### Not

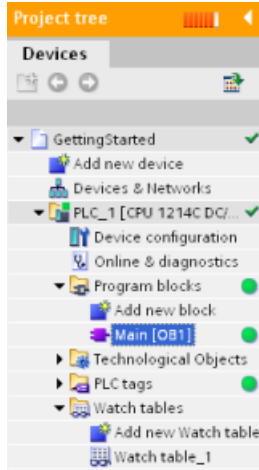
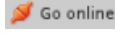
Kontrol fonksiyonlarını kullanırken her zaman dikkatli olunuz. Bu işlevler kullanıcı / sistem programının yürütülmesini ciddi şekilde etkileyebilir.

Çizelge 14- 3 STEP 7 editörlerinin Çevrimiçi yetenekleri

Editör	İzleme	Değiştirme	Zorlama
İzleme çizelgesi	Evet	Evet	Hayır
Zorlama Çizelgesi	Evet	Hayır	Evet
Program editör	Evet	Evet	Hayır
Etiket Çizelgesi	Evet	Hayır	Hayır
DB editör	Evet	Hayır	Hayır

### 14.11.1 CPU'daki değerleri izlemek için çevrimiçi olma

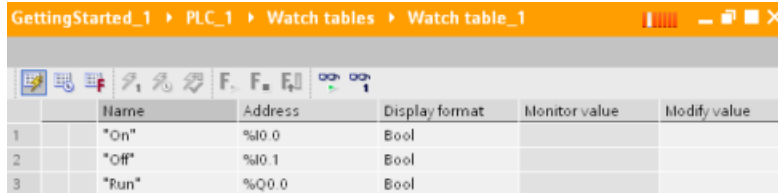
Etiketleri izlemek için, CPU ile çevrimiçi bağlantı olması gerekir. Sadece araç çubuğundaki "Go online" düğmesine tıklayınız.



CPU' ya bağladığınızda, STEP 7, çalışma alanlarının başlıklarını turuncuya döndürür.

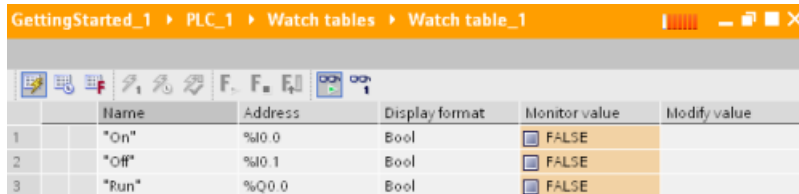
Proje ağacı, çevrimdışı bir proje ile çevrimiçi CPU'nun bir karşılaştırmasını görüntüler. Bir yeşil daire, CPU ve projenin senkronize olduğu bununda her ikisinin aynı yapılandırma ve kullanıcı programına sahip olduğu anlamına geldiğini belirtir.

Etiket çizelgeleri etiketleri gösterir. İzleme çizelgeleri de etiketlerin yanı sıra doğrudan adresleri gösterebilir.



	Name	Address	Display format	Monitor value	Modify value
1	*On	%I0.0	Bool		
2	*Off	%I0.1	Bool		
3	*Run	%Q0.0	Bool		

Kullanıcı programının yürütülmesini izlemek ve etiketlerin değerlerini görüntülemek için araç çubuğundaki "tümünü izle" butonuna tıklayınız.



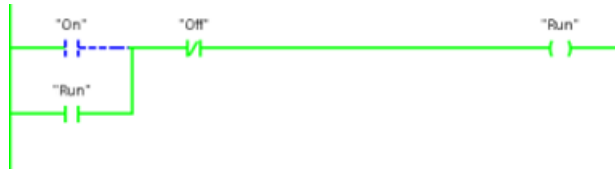
	Name	Address	Display format	Monitor value	Modify value
1	*On	%I0.0	Bool	FALSE	
2	*Off	%I0.1	Bool	FALSE	
3	*Run	%Q0.0	Bool	FALSE	

"Değeri izle" alanı her etiket için değerini gösterir.

### 14.11.2 Program editöründe durumu gösterme

LAD ve FBD program editörlerinde etiketlerin durumunu izleyebilirsiniz. LAD editörünü görüntülemek için editör çubuğunu kullanınız. Editör çubuğu editörleri açmak veya kapamak zorunda kalmadan açık editörler arasında görünümü değiştirmenize izin verir.

Program editörünün araç çubuğunda, kullanıcı programının durumunu görüntülemek için "açık / kapalı izleme" butonuna tıklayınız.



Program editöründeki devre enerji akışını yeşil gösterir.

Ayrıca komut değerini değiştirmek için komutta ya da parametre üzerinde sağ tıklayabilirsiniz.

### 14.11.3 Başlangıç değerlerini sıfırlamak için bir DB çevrimiçi değerlerini yakalama

Bir küresel DB ye yönelik başlangıç değerleri haline gelmesi için bir çevrimiçi CPU da izlenmekte olan mevcut değerleri yakalayabilirsiniz.

- CPU ile çevrimiçi bağlantınız olması gerekir.
- CPU RUN modunda olmalıdır.
- STEP 7 de DB'yi açmış olmanız gerekir.



DB de seçilen etiketlerin geçerli değerlerini yakalamak için "izlenen değerlerin anlık durum görüntüsünü (snapshot) göster" düğmesini kullanınız. Daha sonra DB'nin "başlangıç değeri" sütununa bu değerleri kopyalayabilirsiniz.

1. DB editöründe, "Tüm etiketleri izle" butonuna tıklayınız. "Değeri izle" sütunu mevcut veri değerlerini görüntüler.
2. "Snapshot" sütununda geçerli değerleri görüntülemek için " izlenen değerlerin anlık durum görüntüsünü (snapshot) göster " düğmesine tıklayınız.
3. CPU'daki verileri izlemeyi durdurmak için " tümünü izle" düğmesini tıklayınız.
4. Bir etiket için "Snapshot" sütunundaki bir değeri kopyalayınız.
  - Kopyalanacak bir değer seçiniz.
  - İçerik menüsünü görüntülemek için seçilen değeri sağ tıklayınız.
  - "Kopyala" komutunu seçiniz.
5. Etiket için karşılık gelen "Başlangıç değeri" sütununa kopyalanan değeri yapıştırın. (Hücreyi sağ tıklayınız ve içerik menüsünden "Yapıştır" seçeneğini seçiniz.)

6. DB için yeni başlangıç değerleri olarak kopyalanan değerlerini yapılandırmak için projeyi kaydediniz.
7. DB'yi derleyin ve CPU'ya indirin. CPU RUN moduna geçtikten sonra DB yeni başlangıç değerlerini kullanır.

#### Not

"Değeri izle" sütununda gösterilen değerler daima CPU'dan kopyalanır. STEP 7, tüm değerlerin CPU nun aynı tarama çevriminden gelip gelmediğini kontrol etmez.

### 14.11.4 CPU değerlerini izlemek ve değiştirmek için bir izleme çizelgesi kullanma

Bir izleme çizelgesi, CPU programınızı çalıştırdıkça veri noktaları üzerinde izleme ve kontrol fonksiyonlarını gerçekleştirmenize olanak sağlar. Bu veri noktaları, izleme veya kontrol fonksiyonuna bağlı olarak süreci görüntüsü ( I veya Q ), M, DB veya fiziksel girişler ( I\_ : P ) olabilir. Fiziksel çıkışları ( Q\_ : P ) doğru bir şekilde izleyemezsiniz. Çünkü izleme fonksiyonu, sadece Q bellekten yazılan son değeri görüntüleyebilir ve fiziksel çıkışlarından gelen gerçek değeri okumaz.

İzleme fonksiyonu programın sırasını değiştirmez. Bu, CPU'daki program sırası ve program verileri hakkındaki bilgileri size sunar.

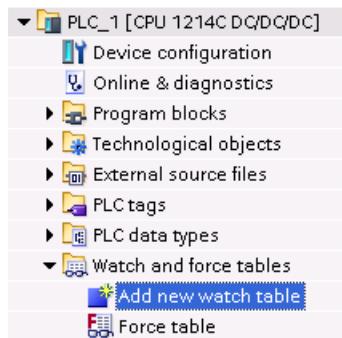
Kontrol fonksiyonları, kullanıcının program sırası ve verileri kontrol etmesini sağlar. Kontrol fonksiyonlarını kullanırken dikkatli olmalısınız. Bu fonksiyonlar kullanıcı / sistem programının yürütülmesini ciddi şekilde etkileyebilir. Üç kontrol fonksiyonu STOP'taki Değiştir, Zorla ve Çıkışları etkinleştir olarak belirtilir.

İzleme çizelgesi ile aşağıdaki çevrimiçi işlevleri gerçekleştirebilirsiniz:

- Etiketlerin durumunu izleme
- Münferit etiketler için değerleri değiştirme

Etiketlin izlenme veya değiştirilme zamanını seçiniz:

- Tarama çevriminin başlangıcı: tarama çevriminin başında değeri okur ya da yazar
- Tarama çevriminin sonu: tarama çevriminin sonunda değeri okur ya da yazar
- Durmaya geçer



Bir izleme çizelgesi oluşturmak için:

1. Yeni bir izleme çizelgesi açmak için "yeni izleme çizelgesi ekle" çift tıklayınız.
2. İzlemek çizelgesine bir etiket eklemek için etiket adını giriniz. Aşağıdaki seçenekler etiketlerin izlenmesi için kullanılabilir:
  - Tümünü izle: Bu komut aktif izleme çizelgesinde görünür etiketleri izlemeye başlar.
  - Şimdi izle: Bu komut aktif izleme çizelgesinde görünür etiketleri izlemeye başlar. İzleme çizelgesi derhal ve sadece bir kez etiketleri izler.

Aşağıdaki seçenekler etiketleri değiştirmek için kullanılabilir:

- "0'a Değiştir" seçilen adresin değerini "0" olarak ayarlar.
- "1'e Değiştir" seçilen adresin değerini "1" olarak ayarlar.
- "Şimdi değiştir" bir tarama çevrimi için seçilen adreslerin değerini derhal değiştirir.
- "Tetikleyici ile değiştirin" seçilen adreslerin değerlerini değiştirir.

Bu fonksiyon, seçilen adreslerin aslında değiştirildiğini belirtmek için geri bildirim sağlamaz. Değişimin geri bildirim gerekli ise, "Şimdi değiştir" işlevini kullanınız.

- "Çevresel çıkışları etkinleştir" çıkışı devre dışı bırak komutunu devre dışı bırakır ve yalnızca CPU STOP modunda olduğunda kullanılabilir.

Etiketleri izlemek için, CPU ile çevrimiçi bağlantı olması gerekir.

Name	Address	Display format	Monitor value	Monitor with trigger	Modify with trigger	Modify value
1	"Start"	%I.0	Bool	Permanent	Permanent	
2	"Stop"	%I.1	Bool	Permanent	Permanent	
3	"Running"	%M.0	Bool	Permanent	Permanent	

Çeşitli fonksiyonları seçmek için izleme çizelgesinin üstündeki düğmeleri kullanınız.

Açılan seçimden bir görüntü formatını izlemek ve seçmek için etiket adını giriniz. CPU ile çevrimiçi bağlantı yaparak, " Değeri izle " alanındaki veri noktasının gerçek değerini görüntülemek için "İzle" butonuna tıklayınız.

#### 14.11.4.1 PLC etiketlerini izlerken veya değiştirirken bir tetikleyici kullanma

Tetikleme, tarama çevriminin hangi noktasında seçilen adresin izleneceği ya da değiştirileceğini belirler.

Çizelge 14- 4 Tetikleme tipleri

Tetikleme	Açıklama
Kalıcı	Sürekli veri toplar
Tarama çevrim başlangıcında	Kalıcı: CPU, girişleri okuduktan sonra, tarama çevriminin başında verileri sürekli toplar Bir kez: CPU girişleri okuduktan sonra, tarama çevriminin başında verileri toplar
Tarama çevrimi sonunda	Kalıcı: CPU, çıkışları yazmadan önce, tarama çevriminin sonunda verileri sürekli toplar Bir kez: CPU, çıkışları yazmadan önce, tarama çevriminin sonunda verileri bir kez toplar
STOP a geçişte	Kalıcı: CPU nun STOP a geçişlerinde sürekli veri toplar Bir kez: CPU nun STOP geçişlerinden sonra bir kez veri toplar

Belirli bir tetikleyicide bir PLC etiketini değiştirmek için çevrimin başlangıç veya sonunu seçiniz.

- Bir çıkışı değiştirme: Bir çıkış değiştirmek için en iyi tetikleyici olay, CPU çıkışları yazıyor yazmadan hemen önce, tarama çevriminin sonunda yer almaktadır.

Fiziksel çıkışlara hangi değer yazılacağını belirlemek için tarama çevriminin başında çıkışların değerini izleyiniz. CPU, program mantığını kontrol etmek ve gerçek I / O davranışıyla karşılaştırmak amacıyla fiziksel çıkışlara değerleri yazmadan önce ayrıca çıkışları izleyiniz.

- Bir girişi değiştirme: bir giriş değiştirmek için en iyi tetikleyici olay, CPU girişleri okuduktan hemen sonra ve kullanıcı programı giriş değerleri okumadan önce çevrimin başlangıcında yer almaktadır.

Değerlerin tarama sırasında değiştiğinden şüpheleniyorsanız, tarama çevriminin sonundaki giriş değerinin, tarama çevriminin başındaki değerden değişmediğinden emin olmak için tarama çevriminin sonundaki girişlerin değerini izlemek isteyebilirsiniz. Değerlerde bir fark varsa, kullanıcı programınız yanlılıkla girişlere yazabilir.

CPU nun STOP a neden gitmiş olabileceğini tanılamak için son proses değerlerini yakalamak için "STOP a Geçiş" tetiklemesini kullanınız.

#### 14.11.4.2 STOP modunda çıkışları etkinleştirme

İzleme çizelgesi, CPU STOP modunda olduğunda, size çıkışlara yazmanıza izin verir. Bu işlevsellik çıkışların bağlantılarını kontrol etmenize ve bir çıkış pinine bağlı kablonun, bağlı olduğu proses cihazının terminaline yüksek ya da düşük sinyali başlattığını doğrulamanıza izin verir.

#### UYARI

##### STOP modunda fiziksel çıkışlara yazma riskleri

CPU STOP modunda olsa bile, fiziksel bir çıkışı etkinleştirme, bağlı olduğu proses noktasını aktive edebilir bu da muhtemelen beklenmeyen ekipman çalışmasına neden olur. Beklenmedik ekipman çalışması ölüme ya da ciddi yaralanmalara neden olabilir.

İzleme çizelgesinden bir çıkışa yazmadan önce, fiziksel çıkışın değiştirilmesinin, beklenmeyen ekipman çalışmasına neden olmayacağından emin olunuz. Her zaman proses ekipmanları için güvenlik önlemlerine uyunuz.

Çıkışlar etkin olduğunda STOP modunda çıkışların durumunu değiştirebilirsiniz. Çıkışlar devre dışı ise, STOP modunda çıkışları değiştiremezsiniz. İzleme çizelgesinden çıkışların STOP modunda değiştirilmesini etkinleştirmek için şu adımları izleyiniz:

1. "Online" menüsünden "Genişletilmiş mod" menü komutunu seçiniz.
2. İzleme çizelgesinin satırını sağ tıkladıktan sonra "Online" menüsünden veya içerik menüsünden "Değiştir" komutunun "çevresel çıkışları etkinleştir" seçeneğini seçiniz.

Dağıtılmış I / O yu yapılandırdıysanız, STOP modunda çıkışları etkinleştiremezsiniz. Bunu yapmaya çalıştığınızda bir hata döndürülür. CPU yu RUN moduna ayarlama, "çevresel çıkışları etkinleştir" seçeneğini devre dışı bırakır. Herhangi bir giriş veya çıkış zorlanırsa, CPU'nun, STOP modunda iken çıkışları etkinleştirilmesine izin verilmez. İlk olarak zorlama fonksiyonu iptal edilmelidir.



## 14.11.5 CPU'da zorlama değerleri

### 14.11.5.1 Cebir çizelgesini kullanarak

Bir cebir çizelgesi çevresel giriş veya çevresel çıkış adresi için belirtilen bir değere bir giriş veya çıkış noktası için değer yazan bir "cebiri" fonksiyonu sağlar. CPU bu cebiri değeri, kullanıcı programının yürütülmesi önce giriş proses görüntüsüne ve çıkışlar modüllere yazılmadan önce çıkış proses görüntüsüne uygular.

#### Not

Cebiri değerleri cebiri çizelgesinde değil CPU'da saklanır.

Bir girişi (veya "I" adresi) veya bir çıkışı (veya "Q" adresi) zorlayamazsınız. Ancak, bir çevresel giriş veya çevresel çıkışı zorlayabilirsiniz. Zorlama çizelgesi ":P" yi adrese otomatik olarak ekler (örneğin: "On": P veya "Run": P).

	Name	Address	Display format	Monitor value	Force value	F
1	"On" P	%I0.0 P	Bool		TRUE	<input checked="" type="checkbox"/>
2	"Off" P	%I0.1 P	Bool			<input type="checkbox"/>
3	"Run" P	%Q0.1 P	Bool			<input type="checkbox"/>

"Cebiri değer" hücrelerinde, zorlanacak giriş veya çıkış için değeri giriniz. Daha sonra giriş veya çıkışın zorlanmasını etkinleştirmek için "Cebiri" sütunundaki onay kutusunu kullanabilirsiniz.



Cebiri çizelgesindeki etiketlerin değerini zorlamak için "Cebiri başlatınız veya değiştiriniz" düğmesini kullanınız. Etiketlerinin değerini sıfırlamak için "zorlamayı durdur" düğmesini tıklayınız.

Cebiri çizelgesinde, bir giriş için cebiri değerinin durumunu izleyebilirsiniz. Ancak, bir çıkışın zorlanmış değerini izleyemezsiniz.

Ayrıca program editöründe zorlanmış değer durumunu görebilirsiniz.



#### Not

Bir giriş ya da çıkış, cebiri çizelgesinde zorlandığında, cebiri eylemleri proje yapılandırma parçası haline gelir. STEP 7'yi kapatırsanız, zorlanan elemanlar temizlenene kadar CPU programında aktif kalır. Bu zorlanan elemanları temizlemek için, çevrimiçi CPU ile bağlanmak için STEP 7'yi kullanmanız gerekir ve daha sonra bu elemanlar için cebiri fonksiyonunu kapatmak ya da durdurmak için cebiri çizelgesini kullanınız.

### 14.11.5.2 Cebir fonksiyonunun çalışması

CPU, Cebir çizelgesinde fiziksel giriş veya çıkış adresini (I\_P: veya Q\_P) belirterek ve daha sonra cebir işlevini başlatarak size giriş ve çıkış noktasını ( noktalarını ) zorlamanıza izin verir.

Programda, fiziksel giriş okuma değerleri zorlanan değerler tarafından üzerine yazılır. Program işlem sırasında zorlanan değeri kullanır. Program fiziksel bir çıkışa yazdığı anda, çıkış değeri cebri değer tarafından üzerine yazılır. Zorlama değeri fiziksel çıkışta görünür ve proses tarafından kullanılır.

Bir giriş veya çıkış, cebir çizelgesinde zorlandığında, cebir eylemleri kullanıcı programının bir parçası haline gelir. Programlama yazılımı kapalı olsa bile onlar programlama yazılımı ile çevrimiçi olarak ve cebir fonksiyonunu durdurarak temizlenene kadar, cebir seçimleri CPU çalışma programında aktif kalır. Bir bellek kartından başka bir CPU' ya yüklenen zorlama noktalarına sahip Programlar programda seçilen noktaları zorlamaya devam edecektir.

CPU bir yazma korumalı bellek kartından kullanıcı programını yürütürse, bir izleme çizelgesinden I / O zorlamasını başlatamaz veya değiştiremezsiniz çünkü yazma korumalı kullanıcı programında değerleri geçersiz kılamazsınız. Yazma korumalı değerleri zorlamak için herhangi bir girişim bir hata üretir. Bir kullanıcı programını aktarmak için bir bellek kartı kullanıyorsanız, bu bellek kartındaki herhangi bir zorlanmış elemanlar CPU' ya transfer edilecektir.

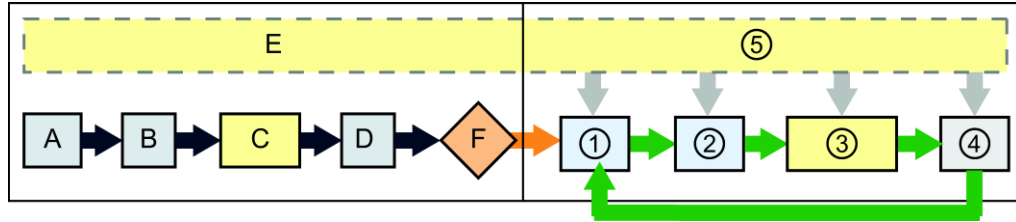
---

#### **Not**

#### **HSC, PWM ve PTO ya atanan dijital I / O noktaları zorlanamaz**

Yüksek hızlı sayıcı (HSC), darbe genişlik modülasyonu (PWM) ve darbe katarı çıkışı cihazları tarafından kullanılan dijital I / O noktaları (PTO) cihaz yapılandırması sırasında atanır. Dijital I / O nokta adresleri bu cihazlara atandığı zaman, atanan I / O nokta adreslerinin değerleri cebir çizelgesinin cebir fonksiyonu tarafından değiştirilemez.

---



#### Yol verme

- A I bellek alanının temizlenmesi cebir fonksiyonu tarafından etkilenmez.
- B Çıktıları değerlerinin başlatılması cebir fonksiyonu tarafından etkilenmez.
- C Kullanıcı programı, fiziksel girişe eriştiğinde başlangıç OB'lerinin yürütülmesi sırasında, CPU cebir değerini uygular.
- D Kesme olaylarının kuyrukta depolanması etkilenmez.
- E Çıktılara yazmanın etkinleştirilmesi etkilenmez.

#### RUN

- ① Fiziksel çıkışlara Q belleği yazarken çıkışlar güncellenirken, CPU, cebir değerini uygular.
- ② Fiziksel girişleri okurken, CPU girişleri I belleğe kopyalamadan hemen önce cebir değerlerini uygular.
- ③ Kullanıcı programı fiziksel girişe eriştiğinde veya fiziksel çıkışa yazdığı anda kullanıcı programı (program çevrim OB'leri) yürütülmesi sırasında, CPU cebir değerini uygular.
- ④ Haberleşme istekleri ve oto-test tanılamamın işlenmesi cebir fonksiyonu tarafından etkilenmez.
- ⑤ Tarama çevriminin herhangi bir bölümü sırasında kesmelerin işlenmesi etkilenmez.

## 14.12 RUN modunda yükleme

CPU "RUN modunda yükle" meyi destekler. Bu yeteneğin, size, program tarafından kontrol edilen süreci minimal rahatsızlık ile bir kullanıcı programında küçük değişiklikler yapmanıza izin vermesi amaçlanmıştır. Ancak, bu yeteneğin uygulanması da yıkıcı ve hatta tehlikeli olabilen büyük program değişikliklerine izin verir.

### UYARI

#### RUN modunda yükleme ile Riskler

RUN modunda değişiklikleri CPU'ya indirdiğinizde, değişiklikler proses çalışmasını hemen etkileyebilir. RUN modunda programın değiştirilmesi personelin ölüm veya ciddi yaralanmalarına neden olabilen ve / veya ekipmana zarar verebilecek beklenmedik sistem çalışmasına neden olabilir.

Sadece, RUN modu değişikliklerinin sistem çalışması üzerindeki etkilerini anlayan yetkili personel, RUN modunda bir yükleme yapmalıdır.

" RUN modunda yükle " Özelliği bir programda değişiklikler yapmanıza ve STOP moduna geçmeden CPU'nuza onları indirmenize izin veriyor:

- Kapatmaya gerek kalmadan mevcut süreçte küçük değişiklikler yapabilirsiniz. (Örneğin, bir parametre değerini değiştirmek )
- Bu özelliği ile programda daha hızlı hata bulma. (Örneğin, normalde açık veya normalde kapalı anahtarı için mantığı ters çevir )

Aşağıdaki program bloğu ve etiket değişikliklerini yapabilir ve RUN modda indirebilirsiniz:

- Fonksiyonları ( FC ), Fonksiyon Bloklarını ( FB ) , Etiket çizelgeleri oluşturma, üzerine yazma ve silme.
- Fonksiyon Blokları ( FB ) için Veri Blokları ( DB ) ve kopya veri blokları oluşturma, silme ve üzerine yazma. DB yapılarına ekleme yapabilir ve onları RUN modunda indirebilirsiniz. CPU, mevcut blok etiketleri değerlerini koruyabilir ve onların başlangıç değerlerine yeni veri bloğu etiketlerini başlatabilir veya CPU, yapılandırma ayarlarınıza (Sayfa 608 ) bağlı olarak, başlangıç değerleri için tüm veri bloğu etiketlerini ayarlayabilirsiniz. RUN modunda bir web sunucu DB (kontrol veya fragmanı) indiremezsiniz.
- Organizasyon Blokları( OB ) Üstüne yazma; Ancak, OB'ler oluşturamaz veya silemezsiniz.

Tek seferde RUN modunda maksimum yirmi blok indirebilirsiniz. Yirmiden fazla blok indirmeniz gerekiyorsa, CPU yu STOP modunda yerleştirmeniz gerekir.

Gerçek bir işleme (bir programda hata ayıklama sırasında yapabileceğiniz simüle edilmiş sürecin aksine) değişiklikleri indirirseniz, bunu indirmeden önce makine ve makine operatörleri için olası güvenlik sonuçları ile düşünmek önemlidir.

---

**Not**

CPU RUN modunda ise ve program değişiklikleri yapılmış ise STEP 7 her zaman ilk RUN'da indirmeye çalışır. Eğer bunun olmasını istemiyorsanız, CPU yu STOP a yerleştirmek gerekir.

Yapılan değişiklikler "RUN da yükle" de desteklenmiyorsa, STEP 7 kullanıcıya CPU nun STOP a gitmek zorunda olduğunu söyler.

---

### 14.12.1 "RUN modunda yükle" için Önkoşullar

RUN modundaki bir CPU' ya program değişikliklerini yükleyebilmek için, bu önkoşulları yerine getirmelisiniz:

- CPU sürümünüz V3.0 veya üstü

---

**Not**

Var olan blokları değiştirmek ve RUN modunda genişletilmiş blok ara yüzünü yüklemek için CPU versiyonu V4.0 veya üstü olmalıdır. (Sayfa 808)

---

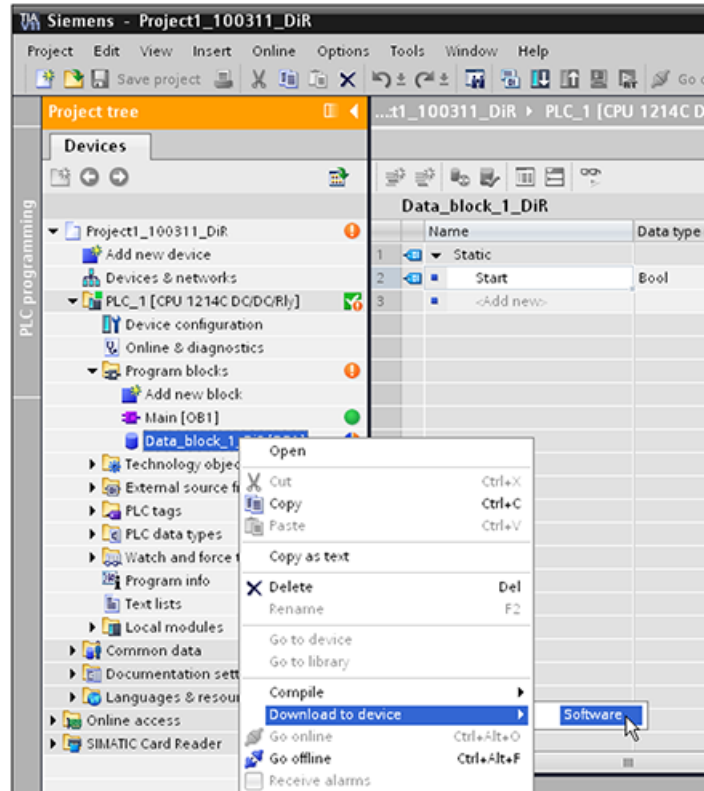
- Programınız başarıyla derlenmelidir.
- STEP 7'yi çalıştırdığınız programlama cihazı ile CPU arasındaki iletişimi başarıyla kurmuş olmanız gerekir.

## 14.12.2 RUN modunda programınız değiştirme

RUN modunda programınızı değiştirmek için ilk olarak CPU ve programın önkoşulları (Sayfa 804) karşıladığını sağlamanız gerekir ve sonra aşağıdaki adımları izleyiniz:

3. RUN konumunda programı indirmek için aşağıdaki yöntemlerden birini seçiniz:

- "Online" menüsünden "cihaza indirin" komutunu seçiniz.
- Araç çubuğundaki "cihaza yükle" düğmesini tıklayınız.
- "Proje ağacı" nda, "Program blokları" nı sağ tıklayınız ve "Cihaza yükle> Yazılım" komutunu seçiniz.



Programı başarıyla derlerse, STEP 7 programı CPU' ya yüklemeye başlar.

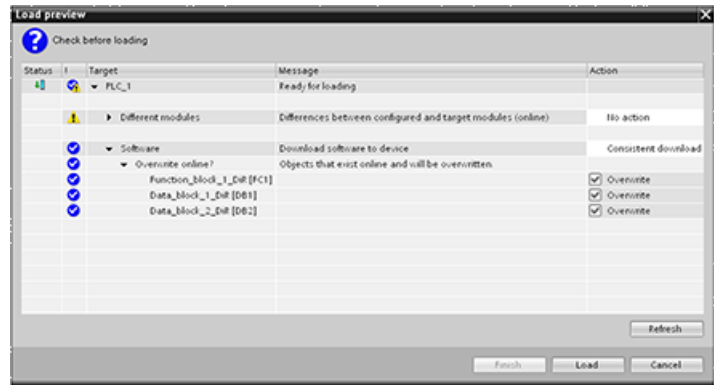
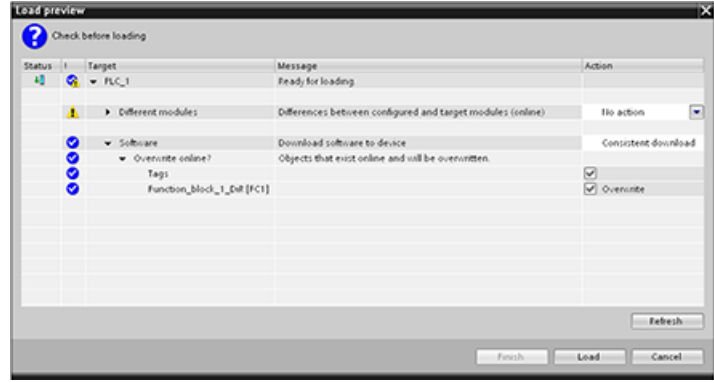
4. STEP 7 programı yüklemenizi ya da işlemi iptal etmenizi istediğinde, CPU' ya programı yüklemek için "Load" tıklayınız.

### 14.12.3 Seçilen blokları yükleme

Program blokları klasöründen, yüklemek için tek bir blok veya blokların seçimini seçebilirsiniz.

Yüklemek için tek bir blok seçerseniz, daha sonra "Eylem" sütunundaki tek seçenek "Tutarlı yükleme" olur. Hangi blokların yükleneceğinden emin olmak için kategori satırını genişletebilirsiniz. Bu örnekte, çevrimdışı blok için küçük bir değişiklik yapıldı ve başka hiçbir bloğun yüklü olması gerekmez.

Bu örnekte, yüklemek için birden fazla blok gereklidir.



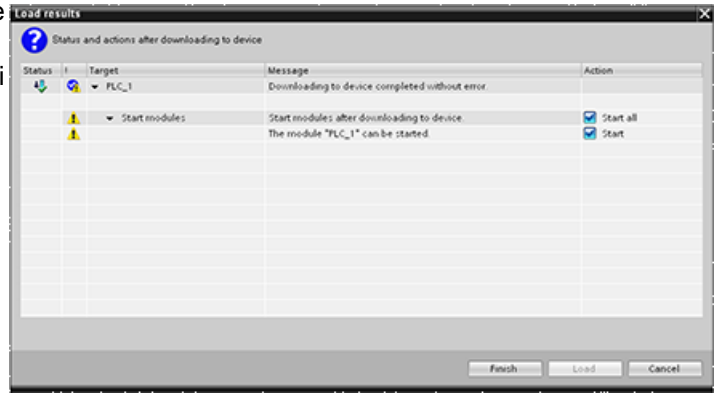
#### Not

Tek seferde RUN modunda maksimum yirmi blok yükleyebilirsiniz. Yirmiden fazla blok yüklemeniz gerekirse, CPU'yu STOP modunda yerleştirmeniz gerekir.

RUN'da yükleme girişiminde bulunur, ancak sistem bunun gerçek yüklemeye başlamadan önce mümkün olmadığını tespit ederse, Stop modülleri kategorisi satırı iletişim kutusunda görüntülenir.



"Load" düğmesine tıklayınız ve "sonuçları yükle" iletişim kutusu görüntülenir. Yükleme tamamlamak için "Finish" düğmesine tıklayınız.



#### 14.12.4 Başka bir bloktaki bir derleme hatası ile seçilen tek bir bloğu yükleme

Başka bir bloktaki bir derleme hatası ile tutarlı yükleme girişiminde bulunursanız, o zaman diyalog bir hata gösterir ve "Load" düğmesi devre dışı bırakılır.



Diğer blokta derleme hatasını düzeltmeniz gerekir. Ardından, "Load" butonu aktif olur.

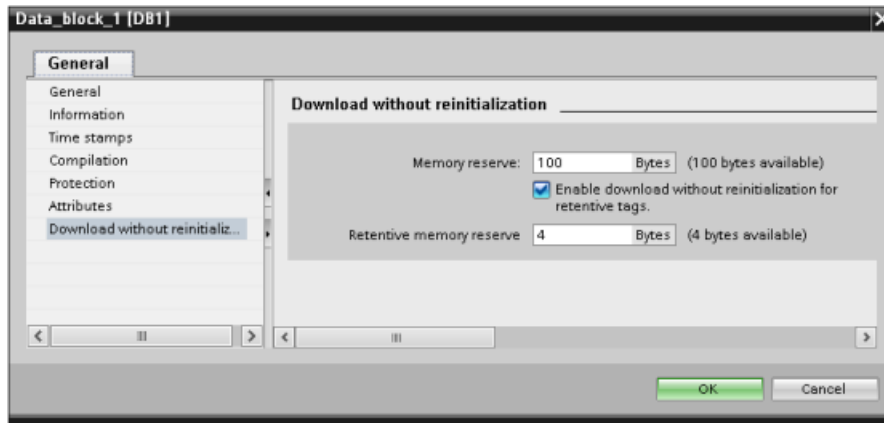


### 14.12.5 RUN modunda mevcut blokları değiştirme ve yükleme

Run modunda yükle özelliği size veri blokları ve fonksiyon bloklarında etiketleri eklemenize ve değiştirmenize ve sonra RUN modunda CPU'ya değiştirilmiş bloğu yüklemenize izin verir.

#### Yeniden başlatma olmaksızın yükleme


Her DB ve FB, size sonradan RUN modunda yükleyebileceğiniz bloğa etiketleri eklemeniz için kullanabileceğiniz rezerve edilmiş bellek miktarına sahiptir. Varsayılan olarak, rezerve edilen belleğin başlangıç boyutu 100 bayttır. Rezerve edilen bellek boyutuna kadar verinize ilave etiketler ekleyebilirsiniz ve RUN modunda CPU'ya genişletilmiş bloğu yükleyebilirsiniz. Bloğunuzda ilave etiketler için daha fazla bellek gerekiyorsa da bellek rezervini artırabilirsiniz. Tahsis ettiğiniz bellek miktarından daha fazla etiket eklerseniz, RUN modunda CPU'ya genişletilmiş blok yükleyemezsiniz.




"Yeniden başlatma olmaksızın yükle" özelliği, daha fazla veri bloğu etiketleri ekleyerek bir veri bloğunu genişletmenize ve RUN modunda genişletilmiş veri bloğunu yüklemenize izin verir. Bu şekilde, bir veri bloğuna etiketler ekleyebilir ve program yeniden başlatma olmadan yükleyebilirsiniz. CPU mevcut veri bloğu etiket değerlerini korur ve yeni eklenen etiketleri başlangıç değerlerine atar.



RUN modunda CPU ile bir çevrimiçi proje için bu işlevi etkinleştirmek amacıyla şu adımları izleyiniz:

1. STEP 7 proje ağacında Program blokları klasöründen bloğu açınız.
2. Fonksiyonunu etkinleştirmek için blok editöründe "Yeniden başlatma olmaksızın yükle" anahtar düğmesine tıklayınız ( bunu etkinleştirdiğinizde simge etrafında bir kutu meydana gelir:  )
3. Tercihinizi onaylamak için OK tıklayınız.
4. Blok ara yüzüne etiketler ekleyiniz ve RUN modunda blok yükleyiniz. Bellek rezervinin izin verdiği ölçüde yeni etiketler ekleyebilir ve yükleyebilirsiniz.

Bloğunuza, bellek rezervi için yapılandırduğunuzdan daha fazla bayt eklerseniz RUN modunda blok yüklemeye çalıştığınızda STEP 7 bir hata görüntüler. Miktarı artırmak için blok özelliklerini düzenlemeniz gerekir. "Yeniden başlatmaksızın yükle" fonksiyon etkin iken mevcut girişleri silemezsiniz veya bloğun "Bellek rezervi" ni değiştiremezsiniz. "Yeniden başlatmaksızın yükle" işlevini devre dışı bırakmak için şu adımları izleyiniz:

1. İşlevi devre dışı bırakmak için blok editöründe "Yeniden başlatmaksızın yükle" anahtar düğmesine tıklayınız. (bu devre dışı olduğunda simge etrafında bir kutu olmaz:  )
2. Tercihinizi onaylamak için OK tıklayınız.
3. Bloğu yükleyiniz. Yükle iletişim penceresi üzerinde, genişletilmiş blok yüklemek için "yeniden başlat" seçmelisiniz.

Yükleme daha sonra başlangıç değerleri için tüm mevcut ve yeni blok etiketlerini yeniden başlatır.

## Tutucu blok etiketleri yükleme

RUN modunda tutucu blok etiketleri yükleme bir tutucu bellek rezerv tahsisi gerektirir. Bu tutucu bellek rezervini yapılandırmak için şu basamakları izleyiniz:

1. STEP 7 proje ağacında Program bloklar klasöründen, bloğa sağ tıklayınız ve içerik menüsünden "Özellikler" i seçiniz.
2. "Yeniden başlatmaksızın yükle" özelliğini seçiniz.
3. "Tutucu etiketleri için Yeniden başlatmaksızın yüklemeyi etkinleştir" onay kutusunu seçiniz.
4. Kuvvetli bellek rezervi için kullanılabilir bayt sayısını yapılandırınız.
5. Değişiklikleri kaydetmek için Tamam'a tıklayınız.
6. Veri bloğuna tutucu veri bloğu etiketleri ekleyiniz ve RUN modunda veri bloğu yükleyebilirsiniz. Kuvvetli bellek rezervinin izin verdiği ölçüde birçok yeni tutucu veri bloğu etiketleri ekleyebilirsiniz ve yükleyebilirsiniz.

Veri bloğunuza kuvvetli bellek rezervi için yapılandırduğunuzdan daha fazla tutucu bayt eklediyseniz, RUN modunda blok yüklemeye çalıştığınızda, STEP 7 bir hata görüntüler. Sadece RUN modunda onları yükleyebilmek için kuvvetli bellek rezervine kadar tutucu blok etiketleri ekleyebilirsiniz.

Genişletilmiş tutucu blok etiketleri yüklediğinizde etiketler mevcut değerleri içerir.

### Yeni bloklar için rezerve bellek miktarını yapılandırma

Yeni veri blokları için varsayılan bellek rezerv boyutu 100 bayttır. Yeni bir blok oluşturduğunuzda, bu rezervde kullanılabilecek 100 bayt vardır. Bellek rezerv büyüklüğünün yeni bloklar için farklı olmasını istiyorsanız, PLC programlama ayarlarında ayarı değiştirebilirsiniz:

7. STEP 7'den, Seçenekler> Ayarlar menü komutunu seçiniz.
8. Ayarlar iletişim kutusunda, "PLC programlama" genişletip "Genel" i seçiniz.
9. "Yeniden başlatmaksızın yükle" bölümünde, bellek rezervi için bayt sayısını giriniz.

Siz yeni bloklar oluşturduğunuzda STEP 7, yeni bloklar için girilen bellek rezerv yapılandırmasını kullanır.

### Kısıtlamalar

Aşağıdaki kısıtlamalar, blokları RUN modunda düzenleme ve yükleme için geçerlidir:

- RUN modunda yeni etiketleri ekleyerek ve yükleyerek bloğu ara yüzü genişletme, yalnızca optimize bloklar (Sayfa 176) için kullanılabilir.
- RUN modunda bir bloğun yapısını değiştiremezsiniz ve yeniden başlatma olmaksızın değiştirilen bloğu yükleyemezsiniz. Etiket adlarını, dizi boyutlarını, veri tiplerini ya da tutucu durumunu değiştirme, bir Struct (Sayfa 117) etiketi yeni üye ekleme işlemi, bloğu RUN modunda yüklerseniz bloğu yeniden başlatmanızı gerektirir. Hala yeniden başlatma olmaksızın RUN modunda blok çalıştırabilmeniz ve yükleyebilmeniz, başlangıç değerleri (veri blokları) veya varsayılan değerler (fonksiyon blokları) veya yorumlarda yapılan değişiklikler mevcut blok etiketlerine yapılabilen değişikliklerdir.

---

#### Not

Blokları değiştirebilme ve bunları RUN modunda yükleyebilme yeteneği, S7-1200 CPU V4.0 ile yenidir. V4.0'dan önce, sadece değiştirilen blokları STOP modunda yükleyebiliyordunuz.

---

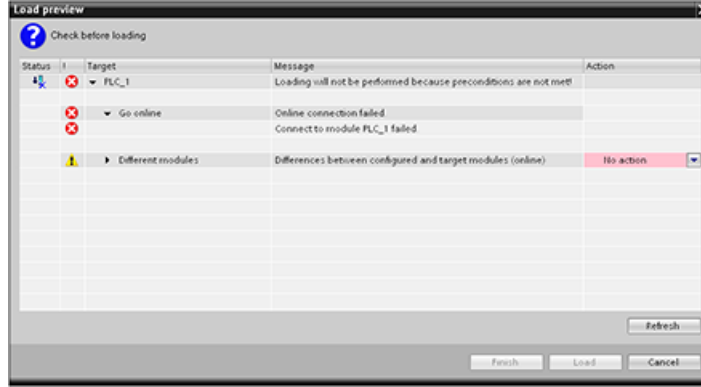
- RUN modunda Bellek rezervinin barındırabileceğinden daha fazla yeni blok etiketleri yükleyemezsiniz.
- RUN modunda kuvvetli bellek rezervinin barındırabileceğinden daha fazla yeni tutucu blok etiketleri yükleyemezsiniz.

### Ayrıca bakınız

Bir V4.0 CPU için bir V3.0 CPU değişimi (Sayfa 995)

### 14.12.6 Yükleme işleminin başarısız olması halinde sistem tepkisi

RUN işleminde ilk yükleme sırasında, bir ağ bağlantı hatası oluşursa, STEP 7 aşağıdaki "Load preview" iletişim penceresini görüntüler:



### 14.12.7 RUN modunda yüklerken dikkat edilecek hususlar

RUN modunda programı yüklemeye başlamadan önce, aşağıdaki durumlar için CPU çalışması üzerinde RUN modu değişikliğinin etkisini göz önünde bulundurunuz:

- Bir çıkış için kontrol mantığını sildiyseniz, CPU STOP moduna geçinceye veya sonraki enerji çevrimine kadar çıkışın son durumunu korur.
- Yüksek hızlı sayacı veya çalışan darbe çıkış fonksiyonlarını sildiyseniz, yüksek hızlı sayıcı darbe veya darbe çıkışı, STOP moduna geçinceye veya sonraki enerji çevrimine kadar çalışmaya devam eder.
- İlk tarama bitinin durumuna bağlı olan herhangi bir mantık, RUN'dan STOP moduna geçinceye veya sonraki enerji çevrimine kadar yürütülmez. İlk tarama biti yalnızca RUN moduna geçişle ayarlanır ve RUN modunda bir yüklemeyle etkilenmez.
- Veri blokları (DB) ve/veya etiketlerin geçerli değerleri üzerine yazılabilir.

---

**Not**

RUN modunda programı yüklemeyen önce, CPU nun RUN modunda değişiklikleri desteklemesi gerekir, programın hatasız olarak derlenmesi gerekir ve STEP 7 ve CPU arasındaki haberleşme hatasız olmalıdır.

Programı blokları ve etiketlerde aşağıdaki değişiklikleri yapabilir ve RUN modunda bunları yükleyebilirsiniz:

- Fonksiyonları ( FC ), Fonksiyon Bloklarını ( FB ) , Etiket çizelgelerini oluşturma, üzerine yazma ve silme.
- Veri Blokları (DB) oluşturma ve silme; Ancak, DB yapısı değişiklikleri üzerine yazılamaz. İlk DB değerler üzerine yazılabilir. RUN modunda bir web sunucusu DB (kontrol veya fragmanı) yükleyemezsiniz.
- Organizasyon Blokları ( OB ) üstüne yazma; Ancak, OB'ler oluşturamaz veya silemezsiniz.

Tek seferde RUN modunda maksimum yirmi blok yükleyebilirsiniz. Yirmiden fazla blok yüklemeniz gerekiyorsa, CPU yu STOP modunda yerleştirmeniz gerekir.

Siz bir yükleme başlattığınızda yükleme tamamlanana kadar, STEP 7 deki diğer görevleri gerçekleştiremezsiniz.

---

**"RUN modunda yükle" nedeniyle başarısız olabilen komutlar**

Run değişikliklerinde yükleme CPU da aktif edildiğinde aşağıdaki komutlar geçici bir hatayla karşılaşabilir. CPU yüklenen değişiklikleri etkinleştirmek için hazırlanıyor iken komut başlatıldığında hata oluşur. Bu süre boyunca CPU kullanıcı programının yükleme Belleğine erişimini tamamlarken CPU, kullanıcı programının bu Yükleme Belleğine erişimini askıya alır. Yüklenen değişiklikler sürekli olarak aktif edilebilmesi için bu yapılır.

Komut	Aktivasyon beklerken gösterilen Tepki
DataLogCreate	STATUS = W#16#80C0, ERROR = TRUE
DataLogOpen	STATUS = W#16#80C0, ERROR = TRUE
DataLogWrite	STATUS = W#16#80C0, ERROR = TRUE
DataLogClose	STATUS = W#16#80C0, ERROR = TRUE
DataLogNewFile	STATUS = W#16#80C0, ERROR = TRUE
READ_DBL	RET_VAL = W#16#82C0
WRIT_DBL	RET_VAL = W#16#82C0
RTM	RET_VAL = 0x80C0

Hata oluştuğunda tüm durumlarda komuttan RLO çıkışı yanlış olacaktır. Hata geçicidir. Bu oluşursa, komutun daha sonra denenmesi gerekmektedir.

---

**Not**

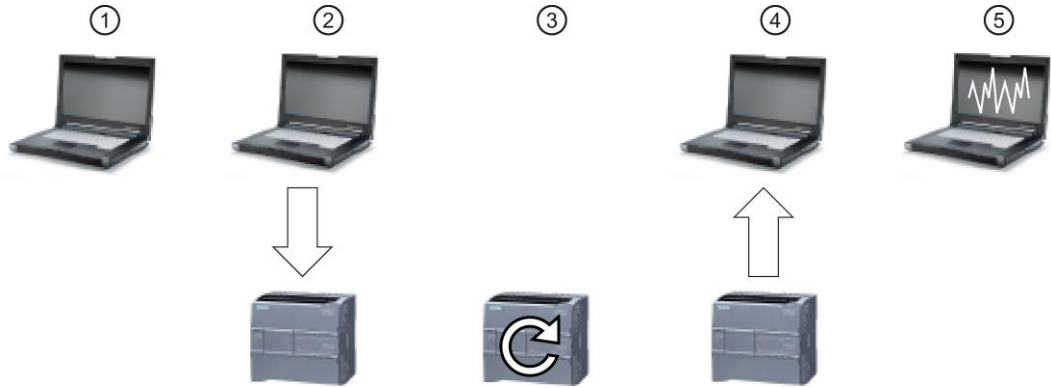
OB'nin mevcut yürütmesinde işlemi yeniden denememeniz gerekir.

---

## 14.13 Tetikleme koşullarında CPU verilerinin izlenmesi ve kaydedilmesi

STEP 7, izlemek ve kaydetmek için PLC değişkenlerini yapılandırabildiğiniz izleme ve mantık analizörü fonksiyonları sağlar. Daha sonra programlama cihazına kaydedilmiş izleme verilerini yükleyebilir ve verileri analiz etmek, yönetmek ve grafiğe dökmek için STEP 7 araçlarını kullanabilirsiniz. İzler oluşturmak ve yönetmek için STEP 7 proje ağacındaki İzler klasörünü kullanınız.

Aşağıdaki şekil izleme özelliğinin çeşitli aşamalarını göstermektedir:



- ① STEP 7'nin izleme editöründe izlemeyi yapılandırınız. Siz kaydedilecek veri değerlerini, kayıt süresini, kayıt sıklığını ve tetikleme durumunu yapılandırabilirsiniz.
- ② İzleme yapılandırmasını STEP 7'den PLC'ye aktarınız.
- ③ PLC, programı çalıştırır ve Tetikleme durumu oluştuğunda, izleme verilerini kaydetmeye başlar.
- ④ Kaydedilen değerleri PLC'den STEP 7'ye aktarınız.
- ⑤ Verileri analiz etmek, grafiksel olarak görüntülemek ve kaydetmek için STEP 7 araçlarını kullanınız.

İzleme özelliğinin yalnızca S7-1200 CPU V4.0 itibarıyla kullanılabilir olduğunu unutmayınız.

Bir izlemeyi programlamak, yapılandırmayı yüklemek, izleme verilerini yüklemek ve mantık analizöründeki verileri görüntülemekle ilgili ayrıntılı bilgi için STEP 7 bilgi sistemine bakınız.



## SM 1278 4xIO-Link Master

### 15.1 SM 1278 4xIO-Link Master genel bakış

#### 15.1.1 Genel bakış

SM 1278 4xIO-Link Master, bir sinyal modülü ve bir haberleşme modülü olarak çalışan 4-portlu bir modüldür. Her port IO-Link modunda, tek bir 24 VDC dijital giriş veya 24 VDC dijital çıkışla çalışabilir.

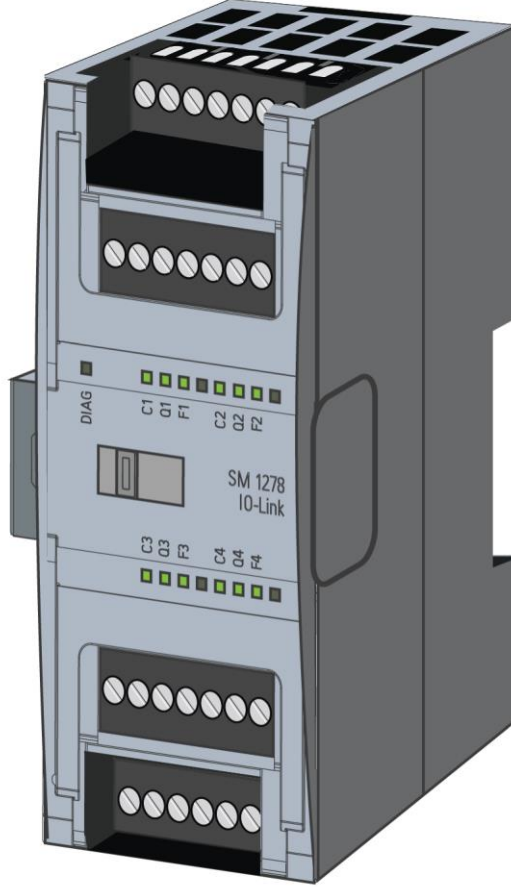
#### 15.1.2 IO-Link ve STEP 7 programınız

STEP 7 S7-1200 kontrolör programınız içinde IOL\_CALL fonksiyon bloğu (FB) kullanarak IO-Link cihazı ile IO-Link master programları döngüsel olmayan biçimde haberleşir. IOL\_CALL FB, programınızın kullandığı IO-Link master'ı ve veri alışverişi için master'ın hangi portları kullandığını gösterir.

IOL\_CALL FB ile çalışma hakkında ayrıntılar için Siemens Endüstri Online Destek web sitesini (<http://support.automation.siemens.com>) ziyaret ediniz. IO-Link ürünleri ve bunların kullanımı hakkında bilgilere erişmek için web sitenizin arama kutusuna "IO-Link" giriniz.

### 15.1.3 Özellikler

Modül görünüşü



### Özellikler

#### Teknik özellikleri

- IO-Link Master, IO-Link şartnamesi V1.1 göre (ayrıntılar için IO-Link Konsorsiyum web sitesine bakınız (<http://io-link.com/en/index.php>))
- Dört portlu (kanal) seri haberleşme modülü
- Veri iletim hızı COM1 (4.8 kbaud), COM2 (38.4 kbaud), COM3 (230.4 kbaud)
- SIO modu (standart IO modu)
- Dört adede kadar IO-Link cihazları (3 telli bağlantı) veya dört standart aktuatöre veya standart enkoderle bağlantı
- Portu ile programlanabilir tanılama fonksiyonu



**Desteklenen fonksiyonlar**

- I&M (kurulum ve bakım) tanıtım verileri
- Firmware güncellemesi
- S7-PCT port yapılandırma aracı, STEP 7 Profesyonel ve S7-1200 V4.0 veya daha yüksek CPU vasıtasıyla IO-Link parametre atama

**15.1.4 Fonksiyonlar**

IO-Link, bir master ve bir cihaz arasında bir noktadan-noktaya bağlantıdır. Hem konvansiyonel ve hemde akıllı sensörler/aktuatöre, kanıtlanmış 3-telli teknolojiyi kullanan zırhsız standart kablolar üzerinden IO-link cihazları olarak kullanılabilir. IO-Link, geleneksel dijital sensörler ve aktuatöre ile geriye dönük uyumludur. Devre durumu ve veri kanalı kanıtlanmış 24 VDC teknoloji ile tasarlanmıştır.

SIMATIC IO-Link teknolojisi hakkında ek bilgi için Siemens Endüstri Online Destek web sitesinde "IO-Link sistemi Fonksiyon Kılavuzu" (<http://support.automation.siemens.com>)'na bakılmalıdır.

**15.1.5 SM 4xIO-Link sinyal modülünün değiştirilmesi****Not****IO-Link parametre verileri**

SM 4xIO-Link Master'ı değiştirdiğinizde, parametre verileri otomatik olarak atanmaz.

**⚠ DİKKAT****Çıkarma ve takma**

Anahtarlanarak devreye alınan yükü olan SM 4xIO-Link Master'ı takarsanız, bu tesisinizde tehlikeli durumlara da yol açabilir.

Bunun sonucu olarak S7-1200 otomasyon sisteminde fiziksel hasar ortaya çıkabilir.

Sadece yük anahtarlanarak devre dışı yapıldığında, SM 4xIO-Link Master'ı çıkarınız veya takınız.

### 15.1.6 Modülü fabrika ayarlarına reset

#### Fabrika ayarlarına resetlemenin etkileri

S7-PCT ile yaptığınız parametre atamalarını teslimat durumuna geri getirmek için "fabrika ayarlarına resetle" işlevini kullanınız.

"Fabrika ayarlarına resetle" dikten sonra, SM 1278 4xIO-Link modülünün parametreleri aşağıdaki gibi atanır:

- Portları DI modunda
- Portlar 0.0-0.3bağlı adreslerle eşleştirilir
- PortQualifier devre dışı
- Bakım verileri 1-3 silinir

---

#### Not

Fabrika ayarlarına resetlediğinizde, cihaz parametreleri silinir ve teslimat durumuna geri döndürülür.

Bir SM 1278 4xIO-Link sinyal modülünü çıkarırsanız bu modülü depoya koymadan önce, fabrika ayarlarına resetleyiniz.

---

#### Prosedür


"Fabrika ayarlarına resetle" me için " Master Yapılandırması> 'Komutlar' sekmesi " altında S7-PCT çevrimiçi yardımında açıklandığı gibi devam ediniz.

## 15.2 Bağlantı

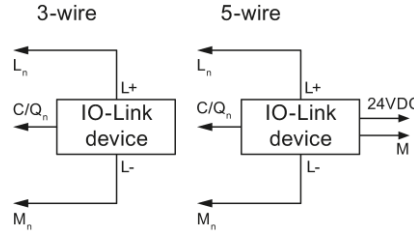
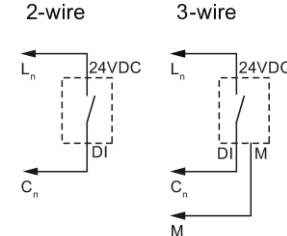
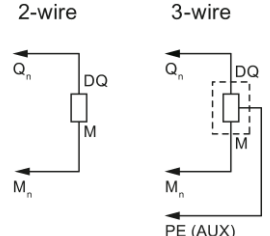
### 15.2.1 Pin ataması

Pin ataması hakkında ayrıntılı bilgi için Ek A'da, "IO-Link Master sinyal modülü" teknik özelliklerine (Sayfa 934) bakınız.

Aşağıdaki çizelge, SM 1278 4xIO-Link Master terminali atamalarını gösterir:

Pin	X10	X11	X12	X13	Notlar	Temel birimler
7	M <sub>1</sub>	M <sub>2</sub>	M <sub>3</sub>	M <sub>4</sub>	<ul style="list-style-type: none"> <li>M<sub>n</sub>: slave için toprak</li> <li>C/Q<sub>n</sub>: SDLC, DI veya DQ</li> <li>L<sub>n</sub>: 24 VDC slave için</li> <li>M: toprak</li> <li>L+: 24 VDC Master için</li> <li>RES: rezerve edilmiş; atanamaz</li> </ul>	A1
6	C/Q <sub>1</sub>	C/Q <sub>2</sub>	C/Q <sub>3</sub>	C/Q <sub>4</sub>		
5	L <sub>1</sub>	L <sub>2</sub>	L <sub>3</sub>	L <sub>4</sub>		
4	RES	RES	RES	RES		
3	 (Fonksiyonel toprak)	RES	RES	RES		
2	M	RES	RES	RES		
1	L+	RES	RES	RES		

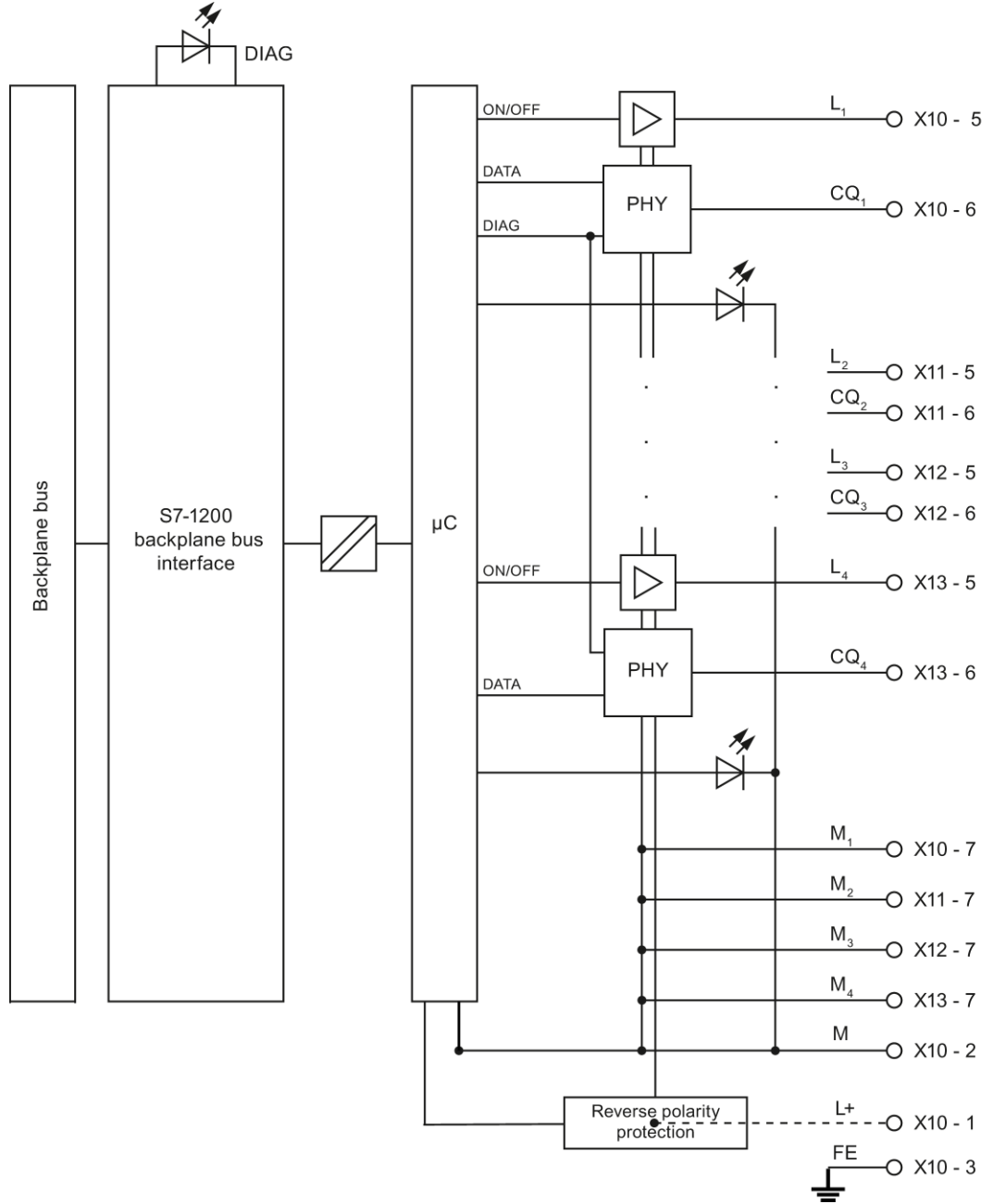
Aşağıdaki Çizelge bağlantı örnekleriyle ilgili çizimler içerir. Burada n = port numarası:

IO-Link Çalışma modu	Çalışma modu DI	Çalışma modu DQ
		

### Not

Bağlı sensörlerin, Master modül L<sub>n</sub> bağlantısı ile sağlanan cihaz beslemesini kullanması gerekir.

## 15.2.2 Blok diyagramı



## 15.3 Parametreler/adres alanı

### 15.3.1 Yapılandırma

#### SM 1278 4xIO-Link Master yapılandırma

Modül entegrasyonu için Siemens mühendislik aracı TIA Portal V12 veya üstü gerekir. Ayrıca IO-Link entegrasyonu için S7-PCT v3.2 veya üstü gerekir.

Devreye alma için parametre atama amacıyla bir mühendislik aracı ve S7-PCT V3.2 veya daha üstü gerekir.

### 15.3.2 Parametreler

Aşağıdaki Çizelge, SM 1278 4xIO-Link Master parametrelerini gösterir:

Parametreler	Değer aralığı	Varsayılan	RUN da yapılandırma	Verimlilik aralığı
Tanımlama port 1	<ul style="list-style-type: none"><li>Devre dışı</li><li>Etkin</li></ul>	Devre dışı	Evet	Port (kanal)
Tanımlama port 2	<ul style="list-style-type: none"><li>Devre dışı</li><li>Etkin</li></ul>	Devre dışı	Evet	Port (kanal)
Tanımlama port 3	<ul style="list-style-type: none"><li>Devre dışı</li><li>Etkin</li></ul>	Devre dışı	Evet	Port (kanal)
Tanımlama port 4	<ul style="list-style-type: none"><li>Devre dışı</li><li>Etkin</li></ul>	Devre dışı	Evet	Port (kanal)

#### Port 1-Port 4 parametre için tanılamayı etkinleştirme

Bu parametre, tanımlamanın, dört IO-Link bağlantısına ait belirli portlar için etkinleştirilmesini sağlar.

Port atamaları aşağıdaki gibidir:

Port 1 → kanal 1

Port 2 → kanal 2

Port 3 → kanal 3

Port 4 → kanal 4

### 15.3.3 Adres alanı

SM 4xIO-Link Master giriş ve çıkış adreslerinin maksimum boyutu her durumda 32 bayttır. S7-PCT port yapılandırma aracını kullanarak adres alanlarını atarsınız.

### 15.3.4 Parametre veri kaydı

#### Kullanıcı programında parametre ataması

Çalışma zamanında cihazınızı yapılandırabilirsiniz.

#### Çalışma zamanında parametreleri değiştirme

Modül parametreleri, veri kaydı 128' de dahil edilmiştir. WRREC komutu ile değiştirilebilir parametreleri modüle aktarabilirsiniz.

CPU sıfırladığınızda (güç çevrimi), CPU, parametreleştirme sürecinde WRREC komutuyla modüle gönderilen parametrelerin üzerine yazar.

#### Parametre atama komutu

Aşağıdaki komut, kullanıcı programında I / O modülüne parametre atamak için verilir:

Komut	Uygulama
SFB 53 WRREC	Değiştirilebilir parametrelerin modüle transferi.

#### Hata mesajı

Aşağıdaki dönüş değeri bir hata durumunda bildirilmiştir:

Hata kodu	Anlamı
80B1H	Veri uzunluğunda hata
80E0H	Başlık bilgisinde hata
80E1H	Parametre hatası

## Veri kayıt yapısı

Aşağıdaki Çizelge, IO-Link parametrelerini gösterir:

Ofset	Etiket	Tip	Varsayılan	Açıklama
0	Versiyon	1 bayt	0x02	IO-Link V1.1 uyarınca IO-Link Master kayıt 0x02 yapısını göstermektedir
1	Parametre uzunluğu	1 bayt	0x02	Parametre uzunluğu (2 bayt + 2 başlık)
IO-Link başlangıç parametreleri				
2	Port tanılama (Port1 1 ila n)	1 bayt	0x00	N port 1 için tanılama etkinleştirme
3	IOL özellikleri	1 bayt	0x00	Modül özellikleri

Aşağıdaki Çizelge, veri kaydı versiyonunu göstermektedir:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Rezerve edildi		Büyük versiyon (00)		Küçük versiyon (0010)			

Aşağıdaki Çizelge veri kaydı port tanılamayı gösterir:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Rezerve edildi				EN_Port4	EN_Port3	EN_Port2	EN_Port1

EN\_Portx:

0 = Tanılamalar devre dışı bırakıldı

1 = Tanılamalar aktifleştirildi

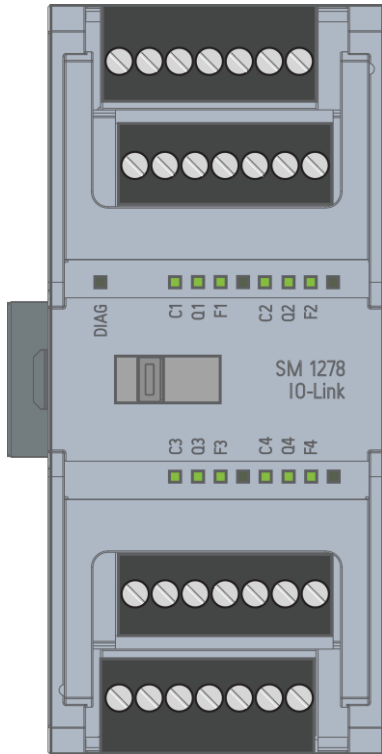
Aşağıdaki Çizelge, veri kaydı IOL özelliklerini gösterir:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Rezerve edildi							

## 15.4 Kesme, hata ve sistem alarmları

### 15.4.1 Durum ve hata göstergesi

#### LED ekran









## LED ekranların anlamı




Aşağıdaki Çizelge, durum ve hata görüntülerinin anlamını açıklar. "Tanı alarmları" bölümünde tanılama alarmları için düzeltici önlemleri bulabilirsiniz.

## LED DIAG

DIAG	Anlamı
 Açık (off)	S7-1200 için arka plan veri yolu besleme OK değil
 Flaşlar	Modül yapılandırılmamış
 Kapalı (on)	Modül parametrelendirilmiş ve modül tanılamalar yok
 Flaşlar	Modül parametrelendirilmiş ve modül tanılamalar VEYA L+ güç bağlı değil



## LED port durumu

IO-Link port modunda IO-Link port için geçerlidir.

COM/1 ... COM/4	Anlamı
 Açık (off)	Port devre dışı
 Flaşlar	Port aktif, cihaz bağlı değil veya Port, yapılandırılmış cihaza bağlı değil
 Kapalı (on)	Port aktif, cihaz bağlı

## LED kanal durumu

DI/Q modunda IO-Link port için geçerlidir.

DI/Q1 ... DI/Q4	Anlamı
 Açık (off)	Proses sinyali = 0
 Kapalı (on)	Proses sinyali = 1

## LED port hatası

F1 ... F4	Anlamı
□ Açık (off)	Hata yok
■ Kapalı (on)	Hata

### 15.4.2 Tanılama alarmları

Modül hataları, sadece IO-Link modunda tanılayıcı (modül durumu) olarak belirtilir.

Tanılayıcı alarmı	Hata kodu (ondalık)	STATU S (W#16#. ..)	Anlamı (IO-Link Hata kodu)	IO-Link master	IO-Link cihazı
Kısa devre	1	1804	IO-Link cihaz üzerindeki proses kablolarında kısa devre	X	
		7710	IO cihazda kısa devre		X
Düşük gerilim	2	5111 5112	Besleme gerilimi çok düşük		X
Aşırı gerilim	3	5110	Besleme gerilimi çok yüksek		X
Aşırı ısınma	5	1805	Sıcaklık master'da aştı	X	
		4000 4210	Sıcaklık cihazda aştı		X
Tel kopuğu	6	1800	<ul style="list-style-type: none"> <li>IO-Link cihazı bağlı değil</li> <li>IO-Link cihazına giden sinyal hattı üzerinde bir kopma var</li> <li>IO-Link cihazı, farklı bir hata nedeniyle iletişim kuramıyor</li> </ul>	X	
Taşma	7	8C10 8C20	Proses etiketi aralığı aşıldı		X
		8C20	Ölçüm aralığı aşıldı		
Aşağı taşma	8	8C30	Proses etiket aralığı çok düşük		X
Hata	9	---	Burada yer almayan tüm IO-Link hata kodları Bu PROFIBUS DP hatasıyla eşleştirilir		X
Parametre atama hatası	16	1882 1883	IO-Link master yapılandırılmadı	X	
		1802	Yanlış cihaz		
		1886	Depolama hatası		
		6320 6321 6350	Cihaz doğru yapılandırılmamış		X
Besleme gerilimi eksik	17	1806	Cihaz için L + besleme gerilimi eksik	X	
		1807	Cihaz için L + besleme gerilimi çok düşük (<20 V)		

Tanılayıcı alarmı	Hata kodu (ondalık)	STATU S (W#16#. ..)	Anlamı (IO-Link Hata kodu)	IO-Link master	IO-Link cihazı
Arızalı sigorta	18	5101	Cihaz üzerindeki sigorta arızalı		X
Emniyetli kapama	25	1880	Ciddi bir hata (master değiştirilmelidir)	X	
Harici hata	26	1809 180A 180B 180C 180D	Verileri saklama hatası	X	
		1808	6'dan fazla hata IO-Link cihazında aynı anda beklemede		



## Teknik özellikler

### A.1 Genel teknik özellikler

#### Standartlara uyumluluk

S7-1200 otomasyon Sistem tasarımı aşağıdaki standartlar ve deney özellikleri ile uyumludur. S7-1200 otomasyon Sistemi için deney kriteri bu standartlar ve deney özellikleri temeline dayanır.

Bütün S7-1200 modelleri bu standartlara göre belgelendirilebilir ve belgelendirme durumu bildirim yapmaksızın değişebilir. Ürün üzerinde işaretlenen değerlere ilişkin olarak uygulanabilir belgelendirmeyi belirleme bizim sorumluluğumuzdur. Parça numarasıyla tam onaylamaların en son ki listesine ilişkin ilave bilgiye ihtiyacınız olursa, lokal Siemens temsilcinize danışınız.

#### CE onayı



EC Directive 2006/95/EC (Düşük gerilim Direktifi) "Belirli gerilim sınırları içinde kullanılmak amacıyla tasarlanmış elektrikli ekipman" S7-1200 otomasyon sistemi aşağıda listelenen EC direktiflerine göre kuralları ve güvenliğe ilişkin hedefleri karşılar ve Avrupa Birliğinin yetkili yayınlarında listelenen programlanabilir kontrolörler için bir araya getirilmiş Avrupa standartlarını karşılar.

- EN 61131-2:2007 Programlanabilir kontrolörler – Ekipman ihtiyaçları ve deneyler
- EC Direktifi 2004/108/EC (EMC Direktifi) "Elektromanyetik uyumluluk"
  - Emisyon standardı  
EN 61000-6-4:2007: Endüstriyel çevre
  - Bağışıklık standardı  
EN 61000-6-2:2005: Endüstriyel çevre
- EC Direktifi 94/9/EC (ATEX) " Potansiyel olarak patlayıcı atmosferde kullanılmak için tasarlanmış ekipman ve koruyucu sistemler"
  - EN 60079-15:2005: Koruma tipi 'n'

CE uyumluluk beyanı yetkili otoriteler için hazır olarak dosyada aşağıdaki adreste tutulmaktadır:

Siemens AG  
IA AS RD ST PLC Amberg  
Werner-von-Siemens-Str. 50  
D92224 Amberg  
Germany

**cULus onayı**

Underwriters Laboratories Inc. aşağıdakilerle uyumludur:

- Underwriters Laboratories, Inc.: UL 508 Listed (Endüstriyel Kontrol Donanımı)
- Canadian Standards Association: CSA C22.2 Number 142 (Proses Kontrol Donanımı)

---

**Not**

The SIMATIC S7-1200 serisi CSA standardını karşılar.

cULus logosu, S7-1200'ün UL 508 ve CSA 22.2 No.142 standartlarına göre Underwriters Laboratories (UL) tarafından deneyden geçirildiğini ve belgelendiğini göstermektedir.

---

**FM onayı**

Factory Mutual Research (FM)

Onaylanan Standart Sınıf Numarası 3600 ve 3611

Onaylandığı kullanım yerleri:

Sınıf I, Kısım 2, Gaz Grup A, B, C, D, Sıcaklık Sınıfı T3C Ta = 60 °C

Sınıf I, Bölge 2, IIC, Sıcaklık Sınıfı T3 Ta = 60 °C

Kanada Sınıf I, Bölge 2 Tesis etme CEC 18-150 başına

**ÖNEMLİ İSTİSNA :** Aynı anda izin verilen girişlerin veya çıkışların sayısı için teknik özelliklere bakınız. Ta = 60 °C için bazı modellerin değerleri düşürülmüştür.



**Bileşenlerin değiştirilmesi Sınıf I, Bölüm 2 ve Bölge 2 için uyumluluğa zarar verebilir.  
Ünitelerin tamiri sadece yetkili Siemens Servis Merkezince yapılmalıdır.**

**ATEX onayı**

ATEX onayı DC modelleri için geçerlidir. ATEX onayı AC ve Röle modelleri için geçerli değildir.

EN 60079-0: 2009: Patlayıcı Ortamlar - Genel Koşullar

EN 60079-15: 2010: Potansiyel olarak Patlayıcı Ortamlarda Kullanılan Elektrikli Cihazlar;  
Koruma tipi 'nA' II 3 G Ex nA IIC T4 veya T3 Gc

EN 60529'a göre asgari IP54 koruma derecesi sağlayan uygun bir mahfazaya veya buna eşdeğer bir koruma derecesi sağlayan bir konuma modülleri takın.

Ekli kablolar ve iletkenler beyan şartları altında ölçülen gerçek sıcaklık için beyan değerlerde olmalıdır. Tesis geçici rejimlerin 119 V. tan daha az ile sınırlandırılacak şekilde olmalıdır. Bu bölümdeki Ani darbe bağışıklığına bakılmalıdır.

**ÖNEMLİ İSTİSNA:** Aynı anda izin verilen girişlerin veya çıkışların sayısı için teknik özelliklere bakınız. Ta = 60 °C için bazı modellerin değerleri düşürülmüştür.

**C-Tick onayı**

S7-1200 otomasyon sistemi AS/NZS 2064 (Sınıf A) standardlarındaki kuralları karşılar.

## Teknik özellikler

### A.1 Genel teknik özellikler

#### Korea Certification



S7-1200 otomasyon sistemi Kore Sertifikasyon (KC Mark) gereksinimlerini karşılar. Bu Sınıf A Donanımı olarak tanımlanmış ve endüstriyel uygulamalar için tasarlanmıştır ve ev kullanımı için kabul edilmemiştir.

#### Maritime onayı

S7-1200 ürünleri periyodik olarak Belirli pazarlarda ve uygulamalarla ilgili özel kuruluş onaylarına sunulur. Eğer parça numarasına göre son kesin onay listesi ile ilgili ek bilgi gerektiğinde yerel Siemens temsilcisine başvurun. Sınıflandırma kurumları:

- ABS (American Bureau of Shipping)
- BV (Bureau Veritas)
- DNV (Det Norske Veritas)
- GL (Germanischer Lloyd)
- LRS (Lloyds Register of Shipping)
- Sınıf NK (Nippon Kaiji Kyokai)
- Korean Register of Shipping

#### Endüstriyel çevreler

S7-1200 otomasyon sistemi endüstriyel çevrelerde kullanmak için tasarlanmıştır.

Çizelge A- 1 Endüstriyel çevreler

Uygulama alanı	Gürültü emisyon şartları	Gürültü bağışıklık kuralları
Endüstriyel	EN 61000-6-4:2007	EN 61000-6-2:2005

#### Elektromanyetik uyumluluk

Elektromanyetik uyumluluk (EMC), elektromanyetik ortamda tasarlandığı gibi ve çevredeki diğer elektrikli cihazlarda bozucu etki yapabilecek elektromanyetik girişimin emisyon seviyesine ulaşmaksızın bir elektrikli cihazın çalışma yeteneğidir.

Çizelge A- 2 EN 61000-6-2'ye göre bağışıklık

Elektromanyetik uyumluluk - EN 61000-6-2' ye göre bağışıklık	
EN 61000-4-2 Elektrostatik deşarj	8 kV bütün yüzeyler için hava deşarjı 6 kV patlayıcı iletken yüzeylere temas direnci
EN 61000-4-3 Işınımlı, radyo frekansı, elektromanyetik alan bağışıklık testi	80 - 1000 MHz, 10 V/m, 80% AM 1 kHz'de 1.4 - 2.0 GHz, 3 V/m, 80% AM 1 kHz'de 2.0 - 2.7 GHz, 1 V/m, 80% AM 1 kHz'de



Elektromanyetik uyumluluk - EN 61000-6-2' ye göre bağıışıklık	
EN 61000-4-4 Hızlı geçiş durumu patlamaları	2 kV, 5 kHz AC ve DC Sistem gücü kublaj devresi ile 2 kV, 5 kHz I/O için kublaj kenetlemesi ile
EN 6100-4-5 Darbe (aşırı gerilim) bağıışıklığı	AC sistemler - 2 kV ortak mod, 1 kV Diferansiyel mod DC sistemler - 2 kV ortak mod, 1 kV Diferansiyel mod DC sistemler için aşağıdaki Darbe bağıışıklığına bakınız
EN 61000-4-6 İletilen bozulmalar	150 kHz - 80 MHz, 10 V RMS, 80% AM 1kHz'de
EN 61000-4-11 Gerilim düşmeleri	AC sistemler 1 döngü için 0%, 12 döngü için 40% ve 30 döngü için 70%, 60 Hz'de

### Darbe (aşırı gerilim) bağıışıklığı

Yıldırım darbe kublajından kaynaklanan darbe gerilimlerine maruz kalan kablolama (kablaj) sistemleri harici koruma ile donatılmalıdır. Yıldırım tipi darbe gerilimlerinden korunmanın değerlendirilmesi için bir özellik EN 61000-4-5'de bulunabilir. Burada, S7-1200 DC CPU'lar ve sinyal modülleri tarafından gerçekleştirilen çalışma limitleri bu standart tarafından tanımlanan darbe gerilimlerine karşı güvenli çalışmayı sağlayacak harici korumayı gerektirir.

Gerekli darbe gerilimi bağıışıklığı korumasını destekleyici bazı cihazlar aşağıda listelenmiştir. Bu cihazlar imalatçının önerilerine göre tesis edilirse koruma sağlarlar. Aynı veya daha iyi özelliklerle başkaları tarafından üretilen cihazlar da kullanılabilir:

- Kaynak gerilimi, 24 VDC: DEHN, Inc., Tip BVT AD 24, Part Number 918 402
- Dijital girişler, 24 VDC: DEHN, Inc., Tip DCO RK E 24, Part Number 919 988
- Dijital çıkışlar, 24 VDC: DEHN, Inc., Tip DCO RK E 24, Part Number 919 988
- Analog girişler: DEHN, Inc., Tip BXT ML4 BD12, Part Number 920 342
- Analog çıkışlar: DEHN, Inc., Tip DCO RK E 12, Part Number 919 987
- Ethernet: DEHN, Inc., Tip DPA M CAT6 RJ45S 48, Part Number 929 100
- RS-232: DEHN, Inc., Tip DCO RK E 12, Part Number 919 987
- RS-485: DEHN, Inc., Tip BXT ML2 BE HFS 5, Part Number 920 270
- Röle çıkışları: Hiçbiri gerekmez

Çizelge A-3 İletilen ve ışıınım emisyonları, EN 61000-6-4'e göre

Elektromanyetik uyumluluk – İletilen ve ışıınım emisyonları, EN 61000-6-4'e göre		
İletilen emisyonlar EN 55011, Sınıf A, Grup 1	0.15 MHz - 0.5 MHz	<79dB (µV) pik gibi; <66 dB (µV) ortalama
	0.5 MHz - 5 MHz	<73dB (µV) pik gibi; <60 dB (µV) ortalama
	5 MHz - 30 MHz	<73dB (µV) pik gibi; <60 dB (µV) ortalama
İışıınım emisyonlar EN 55011, Sınıf A, Grup 1	30 MHz - 230 MHz	<40dB (µV/m) pik gibi; 10m'de ölçülen
	230 MHz - 1 GHz	<47dB (µV/m) pik gibi; 10m'de ölçülen

## Teknik özellikler

### A.1 Genel teknik özellikler

#### Çevresel koşullar

##### Çizelge A- 4 Taşıma ve depolama

Çevresel koşullar - Taşıma ve depolama	
EN 60068-2-2, Test Bb, Kuru sıcak ve EN 60068-2-1, Test Ab, Soğuk	-40 °C ila +70 °C
EN 60068-2-30, Test Db, Yaş sıcak	25 °C - 55 °C, 95% nemlilik
EN 60068-2-14, Test Na, sıcaklık şoku	-40 °C ila +70 °C, durma süresi 3 saat, 5 döngü
EN 60068-2-32, Serbest düşme	0.3 m, 5 kere, ürün paketlenme
Atmosferik basınç	1080 - 660h Pa (-1000 ila 3500m rakıma karşılık)

##### Çizelge A- 5 İşletim koşulları

Çevresel şartlar - İşletme	
Ortam sıcaklık aralığı (Giriş havası 25 mm, ünitenin altında)	-20 °C - 60 °C yatay montaj -20 °C - 50 °C dikey montaj 95% yoğuşmayan nem Başkaca belirtilmedikçe
Atmosferik basınç	1080 - 795 hPa (-1000 ila 2000m rakımına karşılık)
Kirlenmelerin yoğunlukları	SO <sub>2</sub> : < 0.5 ppm; H <sub>2</sub> S: < 0.1 ppm; RH < 60% yoğuşmayan
EN 60068-2-14, Test Nb, sıcaklık değişimi	5 °C - 55 °C, 3 °C/dakika
EN 60068-2-27 Mekanik şok	15 G, 11 ms darbe, her 3 eksenin her birinde şoklar 6
EN 60068-2-6 Sinüsoidal titreşim	DIN ray montaj: 3.5 mm 5-9 Hz'den 3.5 mm, 9 - 150 Hz'den 1G Panel montaj : 5-9 Hz'den 7.0 mm, 9 -150 Hz'den 2G 10 tarama her bir eksen, 1 oktav dakika başına

##### Çizelge A- 6 Yüksek potansiyelli izolasyon deneyi

Yüksek potansiyelli izolasyon deneyi	
24 VDC / 5 VDC nominal devreler	520 VDC (optik izolasyon sınırlarının tip deneyi)
115 VAC / 230 VAC devreler toprağa göre	1500 VAC
115 VAC / 230 VAC devreler 115 VAC / 230 VAC devrelere göre	1500 VAC
115 VAC / 230 VAC devreler 24 VDC / 5 VDC devrelere göre	1500 VAC (3000 VAC/4242 VDC tip deneyi)
Ethernet port 24 VDC / 5 VDC devreler ve toprağa <sup>1</sup> göre	1500 VAC (tip deneyi sadece)

<sup>1</sup> Ethernet port izolasyonu, zararlı gerilimler için kısa süreli devre hataları boyunca hasarı sınırlamak amacıyla tasarlanmıştır. Rutin AC hat gerilim izolasyonu için güvenlik kurallarını onaylamaz.

#### Koruma sınıfı

- EN 61131-2'ye göre koruma Sınıfı II (Koruyucu iletken gerekmez)

**Koruma derecesi**

- IP20 Mekanik Koruma, EN 60529
- Standart prob ile test ederken yüksek gerilimle parmak temaslarına karşı korumalar. Kir, toz, su ve 12.5mm çapından küçük yabancı nesnelere için harici koruma gerekir.

**Anma gerilimleri**

Çizelge A- 7 Anma gerilimleri

Anma gerilimi	Tolerans
24 VDC	20.4 VDC - 28.8 VDC
120/230 VAC	85 VAC - 264 VAC, 47- 63 Hz

**Not**

Mekanik anahtar CPU'ya veya herhangi bir dijital sinyal modülüne enerji verdiğinde yaklaşık 50 mikrosaniye boyunca dijital çıkışlara "1" sinyali gönderilir. Bu işlem, beklenmeyen makine veya proses çalışmasına neden olabilir ve sonuçta personelin ölümüne veya ciddi yaralanmasına ve/veya ekipmana zarara yol açabilir. Kısa süreli darbelere tepki veren cihazlar kullanıyorsanız bunun için bir plan yapmalısınız.

**Ters gerilim koruması**

Her terminal çifti veya CPU'ların, sinyal modüllerinin (SM'lerin) ve sinyal kartlarının (SB'lerin) kullanıcı güç girişi için +24 VDC ters gerilim koruma devresi sağlanmıştır. Farklı terminal çiftlerini ters polaritelere bağlayarak sisteme zarar vermek yine de mümkündür.

S7-1200 sisteminde bazı 24 VDC güç giriş portları birbiriyle çoklu M terminallerini bağlayan ortak lojik devresiyle bağlıdır. Örneğin, veri sayfasında "not isolated" (izole edilmemiş) olarak gösterilen şu devreler birbiriyle bağlıdır: CPU'nun 24 VDC güç kaynağı, CPU'nun sensör gücü, bir SM röle bobinin giriş gücü ve izole edilmemiş bir analog giriş girişin güç kaynağı. Bütün "not isolated" (izole edilmemiş) M terminalleri aynı harici referans potansiyeline bağlanmalıdır.

**⚠ UYARI**

**İzolesiz M terminallerini farklı referans potansiyellerine bağlamak amaçlanmayan akım akışlarına neden olur ve sonuçta PLC veya bağlı ekipmanda zarar veya öngörülemeyen çalışma meydana gelebilir.**

Bu kurallara uyulmaması personel ölümü veya ciddi yaralanma ve/veya mal zararı ile sonuçlanacak zarar ve öngörülemeyen çalışmaya yol açar.

Bir S7-1200 sisteminde izolesiz M terminallerinin aynı referans potansiyeline bağlı olduğundan daima emin olunuz.

**DC Çıkışlar**

CPU'lar, sinyal modülleri (SM'leri) ve sinyal kartlarında (SB'ları) DC çıkışları için kısa devre koruma devresi sağlanmaz,

**Röle elektrik servis ömrü**

Örnek deneylerden sağlanan tipik performans verisi aşağıda gösterilmiştir. Gerçek performans sizin özel uygulamanıza göre değişebilir. Yüke adapte edilmiş bir harici koruma devresi kontakların servis ömrünü iyileştirecektir. Endüktif ve lamba yük şartlarında, normalde kapalı (N.C.) kontaklar normalde açık (N.O.) kontakların ömrünün yaklaşık üçte biri tipik servis ömrüne sahiptirler.

Harici bir koruma devresi kontakların servis ömrünü artıracaktır.

Çizelge A- 8 Tipik performans verisi

Bir aktuatör seçimi için veri				
Sürekli termal akım		2 A max.		
Anahtarlama kapasitesi ve kontakların ömrü				
Omik yük için	Gerilim	Akım	Operasyon döngü sayısı (tipik)	
	24 VDC	2.0 A	0.1 milyon	
	24 VDC	1.0 A	0.2 milyon	
	24 VDC	0.5 A	1.0 milyon	
	48 VAC	1.5 A	1.5 milyon	
	60 VAC	1.5 A	1.5 milyon	
	120 VAC	2.0 A	1.0 milyon	
	120 VAC	1.0 A	1.5 milyon	
	120 VAC	0.5 A	2.0 milyon	
	230 VAC	2.0 A	1.0 milyon	
	230 VAC	1.0 A	1.5 milyon	
	230 VAC	0.5 A	12.0 milyon	
Endüktif yük için (IEC 947-5-1 DC13/AC15'e göre)	Gerilim	Akım	Operasyon döngü sayısı (tipik)	
	24 VDC	2.0 A	0.05 milyon	
	24 VDC	1.0 A	0.1 milyon	
	24 VDC	0.5 A	0.5 milyon	
	24 VAC	1.5 A	1.0 milyon	
	48 VAC	1.5 A	1.0 milyon	
	60 VAC	1.5 A	1.0 milyon	
	120 VAC	2.0 A	0.7 milyon	
	120 VAC	1.0 A	1.0 milyon	
	120 VAC	0.5 A	1.5 milyon	
	230 VAC	2.0 A	0.7 milyon	
	230 VAC	1.0 A	1.0 milyon	
230 VAC	0.5 A	1.5 milyon		
Bir dijital girişin aktifleştirilmesi		Mümkün		
Anahtarlama frekansı				
	Mekanik	Max. 10 Hz		
	Omik yükde	Max. 1 Hz		
	Endüktif yükde (IEC 947-5-1 DC13/AC15'e göre)	Max. 0.5 Hz		
	Lamba yükünde	Max. 1Hz		

## A.2 CPU 1211C

### A.2.1 Genel açıklamalar ve özellikler

Çizelge A- 9 Genel özellikler

Teknik veri	CPU 1211C AC/DC/Röle	CPU 1211C DC/DC/Röle	CPU 1211C DC/DC/DC
Sipariş numarası	6ES7 211-1BE40-0XB0	6ES7 211-1HE40-0XB0	6ES7 211-1AE40-0XB0
Boyutlar W x H x D (mm)	90 x 100 x 75	90 x 100 x 75	90 x 100 x 75
Nakliye ağırlığı	420 gram	380 gram	370 gram
Güç kaybı	10 W	8 W	8 W
Akım kullanılabilir(CM bus)	750 mA max. (5 VDC)	750 mA max. (5 VDC)	750 mA max. (5 VDC)
Akım kullanılabilir(24 VDC)	300 mA max. (sensör gücü)	300 mA max. (sensör gücü)	300 mA max.(sensör gücü)
Dijital giriş akımı çekimi (24 VDC)	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı

Çizelge A- 10 CPU özellikleri

Teknik veri	Açıklama	
Kullanıcı belleği	Çalışma	30 Kbayt
	Yükleme	1 Mbyte dahili, SD kart boyutuna kadar genişletilebilir
	Kalıcı	10 Kbayt
Yerleşik dijital I/O	6 giriş /4 çıkış	
Yerleşik analog I/O	2 giriş	
Proses görüntü boyutu	1024 girişlerin baytları (I) /1024 çıkışların baytları (Q)	
Bit belleği(M)	4096 bayt	
Geçici (lokal) bellek	<ul style="list-style-type: none"> <li>16 Kbayt, başlangıç ve program döngüsü için (FB'ler ve FC'ler ile ilişkili olanlar dahil)</li> <li>6 Kbayt, diğer interrupt önceliği seviyelerinin her birisi için (FB'ler ve FC'ler ile ilişkili olanlar dahil)</li> </ul>	
Sinyal modülleri genişlemesi	Hiçbiri	
SB, CB, BB genişlemesi	1 max.	
Haberleşme modülü genişlemesi	3 CM max.	
Yüksek-hızlı sayıcılar	Her hangi yerleşik veya SB girişlerini kullanmak için 6'ya kadar konfigüre edilmiştir. Çizelgeye bakınız, CPU 1211C: HSC varsayılan adres atamaları (Sayfa 411) 100/180 kHz (Ia.0 - Ia.5)	
Darbe çıkışları <sup>2</sup>	Her hangi yerleşik veya SB çıkışlarını kullanmak için 4'e kadar konfigüre edilmiştir. 100 kHz (Qa.0 - Qa.3)	
Darbe yakalama girişleri	6	

Teknik veri	Açıklama
Süre gecikme interrupt'ları	4 toplam, 1 ms çözünürlük ile
Döngüsel interrupt'lar	4 toplam, 1 ms çözünürlük ile
Kenar interrupt'ları	6 yükselme ve 6 düşme (10 ve 10 opsiyonel sinyal paneli ile)
Bellek kartı	SIMATIC Bellek kartı (opsiyonel)
Gerçek zaman saat doğruluğu	+/- 60 saniye/ay
Gerçek zaman saat tutma süresi	20 gün tipik/12 gün min. 40 °C'da (bakımsız Süper Kapasitör)

<sup>1</sup> HSC, quadrature (dördün) çalıştırma modu için konfigüre edildiğinde, daha düşük hız uygulanabilir.

<sup>2</sup> Röle çıkışları olan CPU modelleri için darbe çıkışlarını kullanmak üzere bir dijital sinyal panel tesis etmek gerekir.

Çizelge A- 11 Performans

Komut tipi	Yürütme hızı
Boolean	0.08 µs/komut
Move Word	1.7 µs/komut
Real math	2.3 µs/komut

## A.2.2 CPU 1211C tarafından desteklenen zamanlayıcılar, sayıcılar, kod blokları

Çizelge A- 12 CPU 1211C tarafından desteklenen zamanlayıcılar, sayıcılar ve kod blokları

Öge	Açıklama	
Bloklar	Tip	OB, FB, FC, DB
	Boyut	30 Kbayt
	Miktar	1024 bloğa kadar toplam (OB'ler + FB'ler + FC'ler + DB'ler)
	FB'ler, FC'ler ve DB'ler için adres aralığı	FB ve FC: 1 - 65535 (FB 1 - FB 65535 gibi) DB: 1 - 59999
	Gömme derinliği	16, program döngü veya başlangıç OB'den 6, her hangi bir interrupt olayı OB'den
	İzleme	2 kod bloklarının durumu aynı anda izlenebilir
	OB'ler	Program döngü
Başlangıç		Çoklu
Süre-geciktirme interrupt'ı		4 (1 olay başına)
Döngüsel interrupt'lar		4 (1 olay başına)
Donanım interrupt'ları		50 (1 olay başına)
Süre hatası interrupt'ları		1
Diagnostik hatası interrupt'ları		1
Çek veya tak modülleri		1
Şasi veya istasyon arızası		1
Günün saati		Çoklu

## Teknik özellikler

### A.2 CPU 1211C

Öge	Açıklama
	Durum
	1
	Güncelleme
	1
	Profil
	1
Zamanlayıcılar	Tip
	IEC
	Miktar
	Bellek boyutu ile sınırlıdır
	Depolama
	DB yapısı, zamanlayıcı başına 16 bayt
Sayıcılar	Tip
	IEC
	Miktar
	Bellek boyutu ile sınırlıdır
	Depolama
	DB içindeki yapı, akım tipine bağlı boyut
	<ul style="list-style-type: none"> <li>• SInt, USInt: 3 bayt</li> <li>• Int, UInt: 6 bayt</li> <li>• DInt, UDInt: 12 bayt</li> </ul>

### Çizelge A- 13 Haberleşme

Teknik veri	Açıklama
Portların numarası	1
Tip	Ethernet
HMI cihazı	3
Programlama cihazı (PG)	1
Bağlantılar	<ul style="list-style-type: none"> <li>• 8, Açık kullanıcı haberleşmesi (aktif veya pasif) için: TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV</li> <li>• 3, sunucu GET/PUT (CPU ile CPU) S7 haberleşme için</li> <li>• 8, müşteri GET/PUT (CPU ile CPU) S7 haberleşme için</li> </ul>
Veri hızları	10/100 Mb/s
İzolasyon (PLC lojiğe göre harici sinyal)	Transformatör yalıtımlı, 1500 VAC, sadece kısa süreli olay güvenliği için
Kablo tipi	CAT5e zırhlı

### Çizelge A- 14 Güç kaynağı

Teknik veri	CPU 1211C AC/DC/Röle	CPU 1211C DC/DC/Röle	CPU 1211C DC/DC/DC
Gerilim aralığı	85 - 264 VAC	20.4 VDC - 28.8 VDC	
Hat frekansı	47 - 63 Hz	--	--
Giriş akımı	CPU sadece max. yükde	60 mA 120 VAC'de 30 mA 240 VAC'de	300 mA 24 VDC'de
	Tüm genişleme donatıları ile CPU, max. yükde	180 mA 120 VAC'de 90 mA 240 VAC'de	900 mA 24 VDC'de
Baskın akım (max.)	20 A 264 VAC'de	12 A 28.8 VDC'de	12 A 28.8 VDC'de
İzolasyon (giriş gücü ile lojik arasında)	1500 VAC	İzole edilmemiş	İzole edilmemiş
Toprak kaçacağı, AC hat ile fonksiyonel toprak arasında	0.5 mA max.	--	--



Teknik veri	CPU 1211C AC/DC/Röle	CPU 1211C DC/DC/Röle	CPU 1211C DC/DC/DC
Tutma süresi (gücün kesilmesi)	20 ms 120 VAC'de 80 ms 240 VAC'de	10 ms 24 VDC'de	10 ms 24 VDC'de
Dahili sigorta, kullanıcı değiştiremez	3 A, 250 V, geç atmalı	3 A, 250 V, geç atmalı	3 A, 250 V, geç atmalı

Çizelge A- 15 Sensör gücü

Teknik veri	CPU 1211C AC/DC/Röle	CPU 1211C DC/DC/Röle	CPU 1211C DC/DC/DC
Gerilim aralığı	20.4 - 28.8 VDC	L+ eksi 4 VDC min.	
Çıkış akımı değeri (max.)	300 mA (kısa-devre korumalı)	300 mA (kısa-devre korumalı)	300 mA (kısa-devre korumalı)
Maksimum dalgalık gürültüsü(<10 MHz)	< 1 V tepeden- tepeye	Giriş hattı ile aynı	Giriş hattı ile aynı
İzolasyon (CPU lojik ile sensör gücüne göre)	İzole edilmemiş	İzole edilmemiş	İzole edilmemiş

### A.2.3 Dijital girişler ve çıkışlar

Çizelge A- 16 Dijital girişler

Teknik veri	CPU 1211C AC/DC/Röle, CPU 1211C DC/DC/Röle ve CPU 1211C DC/DC/DC
Giriş sayısı	6
Tip	Sink/Source (IEC Tip 1 sink)
Anma gerilimi	24 VDC 4 mA'de, nominal
Sürekli izin verilebilir gerilim	30 VDC, max.
Aşırı gerilim	35 VDC için 0.5 saniye
Lojik 1 sinyal (min.)	15 VDC 2.5 mA'de
Lojik 0 sinyal (max.)	5 VDC 1 mA'de
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika
İzolasyon grupları	1
Filtre süreleri	us ayarlar: 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0 ms ayarlar: 0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0
HSC saat giriş hızları (max.) (Lojik 1 Seviye = 15 - 26 VDC)	100/80 kHz (Ia.0 - Ia.5)
Aynı anlı girişlerin sayısı	6, 60 °C'da yatay, 50 °C dikey
Kablo uzunluğu (metre)	500 m zırlı, 300 m zırlısız, 50 m zırlı HSC girişler için

## Teknik özellikler

## A.2 CPU 1211C

Çizelge A- 17 Dijital çıkışlar

Teknik veri	CPU 1211C AC/DC/Röle ve CPU 1211C DC/DC/Röle	CPU 1211C DC/DC/DC
Giriş sayısı	4	4
Tip	Röle, kuru kontak	Yarı iletken- MOSFET (Source)
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	--	20 VDC min.
Lojik 0 sinyal, 10 KΩ yük ile	--	0.1 VDC max.
Akım (max.)	2.0 A	0.5 A
Lamba yük	30 W DC / 200 W AC	5 W
ON durum direnci	0.2 Ω max. yeni olduğunda	0.6 Ω max.
Nokta başına kaçak akım	--	10 μA max.
Ani aşırı akım	7 A kontakların kapanmasıyla	8 A için 100 ms max.
Aşırı yük koruması	Hayır	Hayır
İzolasyon (alan tarafı ile lojik)	1500 VAC için 1 dakika (bobin ve kontak) Hiçbiri (bobin ile lojik)	500 VAC için 1 dakika
İzolasyon direnci	100 MΩ min. yeni olduğunda	--
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika	--
İzolasyon grupları	1	1
İndüktif kenetleme gerilimi	--	L+ eksi 48 VDC, 1 W ısı olarak kayıp
Maksimum röle anahtarlama frekansı	1 Hz	--
Anahtarlama gecikmesi (Qa.0 - Qa.3)	10 ms max.	1.0 μs max., off - on geçişi 3.0 μs max., on - off geçişi
Darbe katırı çıkış hızı	Önerilmez <sup>1</sup>	100 kHz (Qa.0 - Qa.3) <sup>2</sup> , 2 Hz min.
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler	--
Anma yükünde ömür boyu temaslar	100,000 açık/kapalı döngüler	--
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	4, 60 °C yatayda, 50 °C dikey	
Kablo uzunluğu (metre)	500 m zırlı, 150 m zırlısız	500 m zırlı, 150 m zırlısız

<sup>1</sup> Röle çıkışları olan CPU modellerinde, darbe çıkışlarını kullanmak için bir dijital sinyal kartı (SB) takmanız gerekir.

<sup>2</sup> Darbe alıcı ve kablonuza bağlı olarak ilave bir yük direnci (anma akımının en az %10'nunda) darbe sinyal kalitesini ve gürültü bağışıklığını iyileştirebilir.

## A.2.4 Analog girişler

Çizelge A- 18 Analog girişler

Teknik veri	Açıklama
Giriş sayısı	2
Tip	Gerilim (tek-sonlu)
Tam-ölçek aralığı	0 - 10 V
Tam-ölçek aralığı (veri word)	0 - 27648
Aşma aralığı	10.001 - 11.759 V
Aşma aralığı (veri word)	27649 - 32511
Taşma aralığı	11.760 - 11.852 V
Taşma aralığı (veri word)	32512 - 32767
Çözünürlük	10 bit
Maksimum dayanma gerilimi	35 VDC
Düzleme	Hiçbiri, Zayıf, Orta veya Güçlü CPU'nun analog girişleri için basamak tepkisi (ms) çizelgeye bakınız (Sayfa 842)
Gürültü bastırma	10, 50 veya 60 Hz
Empedans	≥100 KΩ
İzolasyon (alan tarafı ile lojik)	Hiçbiri
Doğruluk (25 °C / 0 - 55 °C)	Tam-ölçeğe göre % 3.0 / %3.5
Kablo uzunluğu (metre)	100 m, zırlı bükülü çift

### A.2.4.1 CPU yerleşik analog girişleri için basamak tepkisi

Çizelge A- 19 Basamak tepkisi (ms), 0 V - 10 V %95'de ölçülen

Düzleme seçimi (örnek ortalama)	Bastırma frekansı (İntegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Hiçbiri (1 döngü): Ortalama alma yok	50 ms	50 ms	100 ms
Zayıf (4 döngü): 4 örnek	60ms	70 ms	200 ms
Orta (16 döngü): 16 örnek	200 ms	240 ms	1150 ms
Güçlü (32 döngü): 32 örnek	400 ms	480 ms	2300 ms
<b>Örnekleme süresi</b>	<b>4.17 ms</b>	<b>5 ms</b>	<b>25 ms</b>

**A.2.4.2 CPU yerleşik analog portları için örnekleme süresi**

Çizelge A- 20 CPU yerleşik analog girişleri için örnekleme süresi

Bastırma frekansı (İntegrasyon süresi seçimi)	Örnekleme süresi
60 Hz (16.6 ms)	4.17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

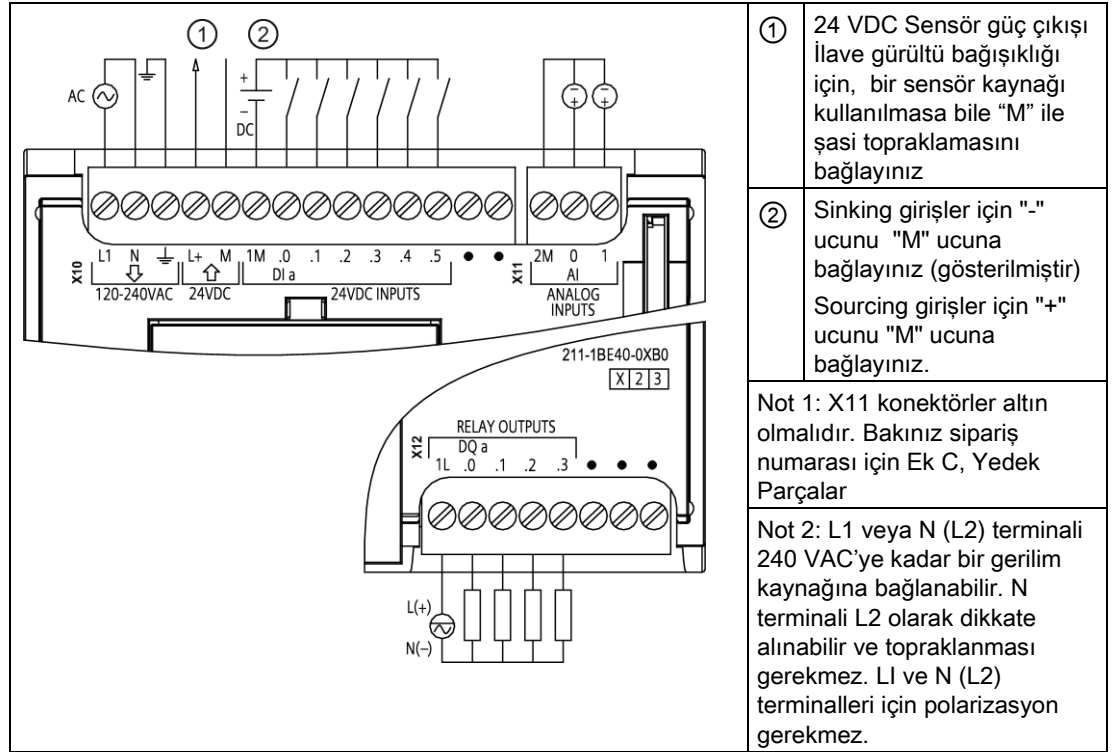
**A.2.4.3 Gerim (CPU) için yerleşik analog girişlerinin ölçme aralıkları**

Çizelge A- 21 Gerilim için analog giriş gösterimi (CPU'lar)

Sistem		Gerilim Ölçme Aralığı	
Decimal	Hexadecimal	0 - 10 V	
32767	7FFF	11.851 V	Taşma
32512	7F00		
32511	7EFF	11.759 V	Aşma aralığı
27649	6C01		Anma aralığı
27648	6C00	10 V	
20736	5100	7.5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

## A.2.5 CPU 1211 bağlantı diyagramları

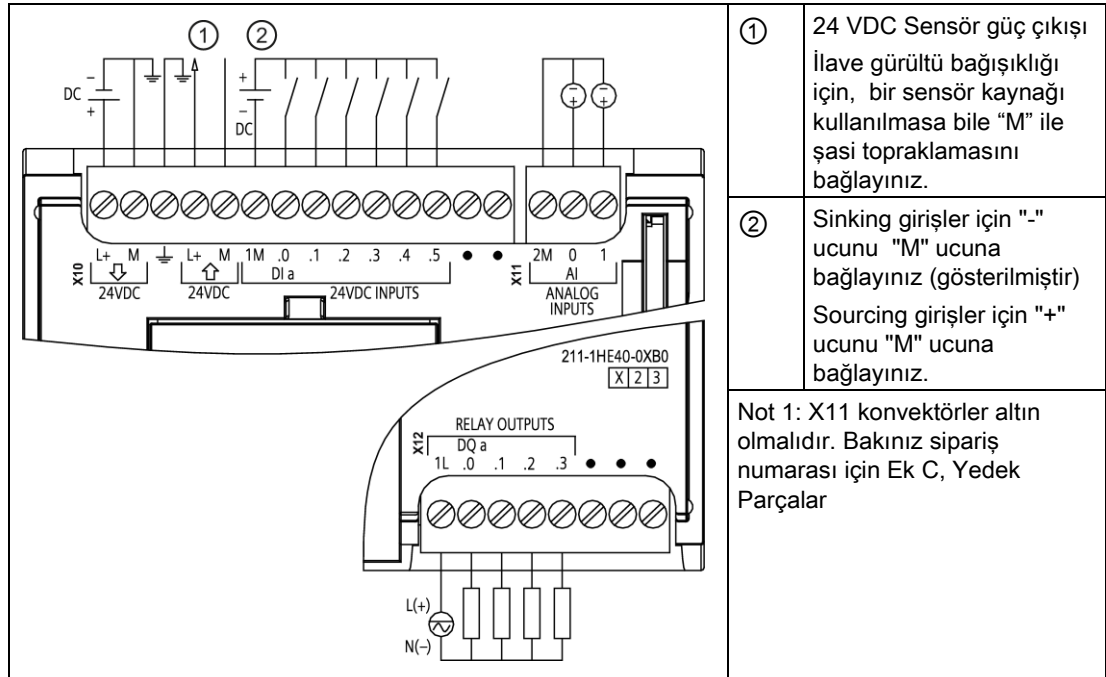
Çizelge A- 22 CPU 1211C AC/DC/Röle (6ES7 211-1BE40-0XB0)



Çizelge A- 23 CPU 1211C AC/DC/Röle (6ES7 211-1BE40-0XB0) için konektör pin yerleşimi

Pin	X10	X11 (altın)	X12
1	L1 / 120-240 VAC	2 M	1L
2	N / 120-240 VAC	AI 0	DQ a.0
3	Fonksiyonel toprak	AI 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.2
5	M / 24 VDC Sensör çıkışı	--	DQ a.3
6	1M	--	Bağlantı yok
7	DI a.0	--	Bağlantı yok
8	DI a.1	--	Bağlantı yok
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Bağlantı yok	--	--
14	Bağlantı yok	--	--

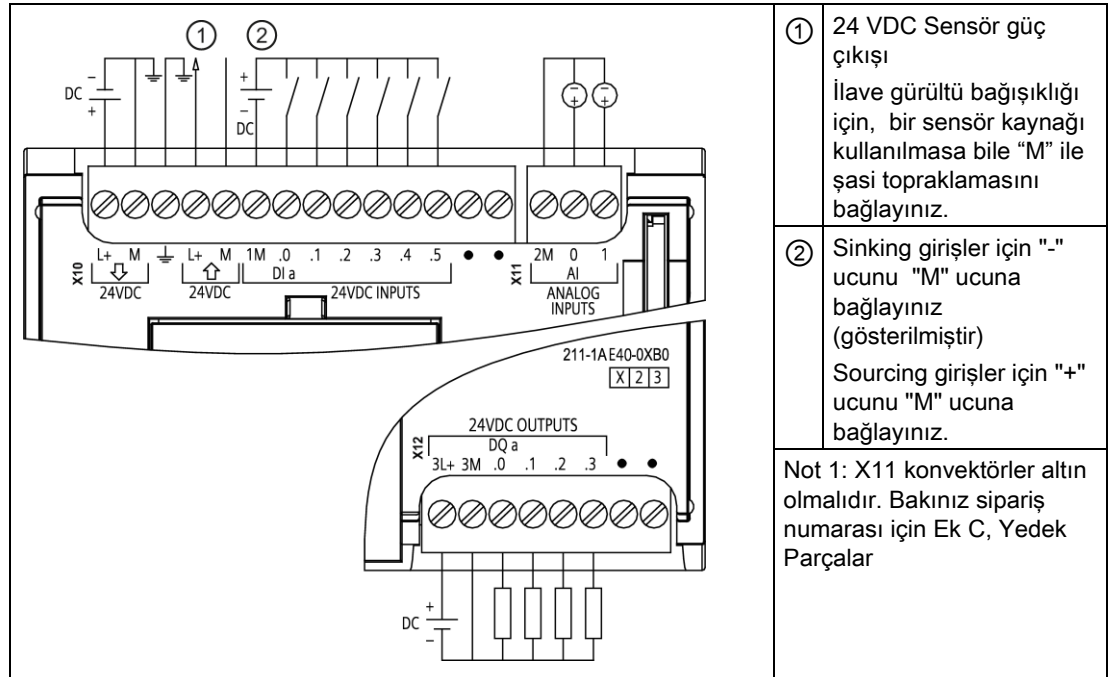
Çizelge A- 24 CPU 1211C DC/DC/Röle(6ES7 211-1HE40-0XB0)



Çizelge A- 25 CPU 1211C DC/DC/Röle (6ES7 211-1HE40-0XB0) için konektör pin yerleşimi

Pin	X10	X11 (altın)	X12
1	L+ / 24 VDC	2 M	1L
2	M / 24 VDC	AI 0	DQ a.0
3	Fonksiyonel toprak	AI 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.2
5	M / 24 VDC Sensör çıkışı	--	DQ a.3
6	1M	--	Bağlantı yok
7	DI a.0	--	Bağlantı yok
8	DI a.1	--	Bağlantı yok
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Bağlantı yok	--	--
14	Bağlantı yok	--	--

Çizelge A- 26 CPU 1211C DC/DC/DC (6ES7 211-1AE40-0XB0)



Çizelge A- 27 CPU 1211C DC/DC/DC (6ES7 211-1AE40-0XB0) için konektör pin yerleşimi

Pin	X10	X11 (altın)	X12
1	L+ / 24 VDC	2 M	3L+
2	M / 24 VDC	AI 0	3M
3	Fonksiyonel toprak	AI 1	DQ a.0
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.1
5	M / 24 VDC Sensör çıkışı	--	DQ a.2
6	1M	--	DQ a.3
7	DI a.0	--	Bağlantı yok
8	DI a.1	--	Bağlantı yok
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	Bağlantı yok	--	--
14	Bağlantı yok	--	--

### Not

Kullanılmayan analog girişler topraklanmalıdır.

## A.3 CPU 1212C

### A.3.1 Genel açıklamalar ve özellikler

Çizelge A- 28 Genel

Teknik veri	CPU 1212C AC/DC/Röle	CPU 1212C DC/DC/Röle	CPU 1212C DC/DC/DC
Sipariş numarası	6ES7 212-1BE40-0XB0	6ES7 212-1HE40-0XB0	6ES7 212-1AE40-0XB0
Boyutlar W x H x D (mm)	90 x 100 x 75	90 x 100 x 75	90 x 100 x 75
Nakliye ağırlığı	425 gram	385 gram	370 gram
Güç kaybı	11 W	9 W	9 W
Akım kullanılabilir (SM ve CM bus)	1000 mA max. (5 VDC)	1000 mA max. (5 VDC)	1000 mA max. (5 VDC)
Akım kullanılabilir(24 VDC)	300 mA max. (Sensör gücü)	300 mA max. (Sensör gücü)	300 mA max. (Sensör gücü)
Dijital girişin çektiği akım(24 VDC)	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı

Çizelge A- 29 CPU özellikleri

Teknik veri	Açıklama	
Kullanıcı belleği	Çalışma	50 Kbayt
	Yükleme	1 Mbyte dahili, SD kartı boyutuna kadar genişletilebilir
	Kalıcı	10 Kbayt
Yerleşik dijital I/O	8 giriş/6 çıkış	
Yerleşik analog I/O	2 giriş	
Proses görüntü boyutu	1024 girişlerin baytları (I)/1024 çıkışların baytları (Q)	
Bit belleği(M)	4096 bayt	
Geçici (lokal) bellek	<ul style="list-style-type: none"> <li>16 Kbayt, başlangıç ve program döngüsü için (ilgili FB'ler ve FC'ler dahil)</li> <li>6 Kbayt, diğer interrupt önceliği seviyelerinin her birisi için (FB'ler ve FC'ler dahil)</li> </ul>	
Sinyal modülleri genişlemesi	2 SM'ler max.	
SB, CB, BB genişlemesi	1 max.	
Haberleşme modülü genişlemesi	3 CM'ler max.	
Yüksek-hızlı sayıcılar	6'ya kadar, yerleşik veya SB girişlerinde kullanılmak üzere konfigüre edilmiş. Bakınız Çizelge, CPU 1212C: HSC varsayılan adres atamaları (Sayfa 411). <ul style="list-style-type: none"> <li>100/180 kHz (Ia.0 - Ia.5)</li> <li>30 /120 kHz (Ia.6 - Ia.7)</li> </ul>	
Darbe çıkışları <sup>2</sup>	4'e kadar, yerleşik veya SB çıkışlarında kullanılmak üzere konfigüre edilmiş <ul style="list-style-type: none"> <li>100 kHz (Qa.0 - Qa.3)</li> <li>30 kHz (Qa.4 - Qa.5)</li> </ul>	



Teknik veri	Açıklama
Darbe yakalama girişleri	8
Süre gecikme interrupt'ları	4 toplam 1 ms çözünürlük ile
Döngüsel interrupt'lar	4 toplam 1 ms çözünürlük ile
Kenar interrupt'ları	8 yükselme ve 8 düşme (12 ve 12 opsiyonel sinyal kartı ile)
Bellek kartı	SIMATIC Bellek kartı (opsiyonel)
Gerçek zaman saat doğruluğu	+/- 60 saniye/ay
Gerçek zaman saat tutma süresi	20 gün tipik/12 gün min. 40 °C'da (bakımsız Süper Kapasitör)

- HSC dördün çalışma moduna konfigüre edildiğinde daha düşük hız uygulanabilir.
- Röle çıkışları olan CPU modellerinde, darbe çıkışlarını kullanmak için bir dijital sinyal kartı (SB) takmanız gerekir.

Çizelge A- 30 Performans

Komut tipi	Yürütme hızı
Boolean	0.08 µs/komut
Move Word	1.7 µs/komut
Real math	2.3 µs/komut

### A.3.2 CPU 1212C tarafından desteklenen zamanlayıcılar, sayıcılar ve kod blokları

Çizelge A- 31 CPU 1212C tarafından desteklenen zamanlayıcılar, sayıcılar ve kod blokları

Öğe	Açıklama	
Bloklar	Tip	OB, FB, FC, DB
	Boyut	50 Kbayt
	Miktar	Toplam 1024 blok'a kadar (OB'ler + FB'ler + FC'ler + DB'ler)
	FB'ler, FC'ler ve DB'ler için adres aralığı	FB ve FC: 1 - 65535 (FB 1 - FB 65535 gibi) DB: 1 - 59999
	Gömme derinliği	16, program döngüsü veya başlangıç OB'sinden 6, herhangi bir interrupt olayı OB'sinden
	İzleme	2 kod bloklarının durumu aynı anda izlenebilir
OB'lar	Program döngüsü	Çoklu
	Başlangıç	Çoklu
	Süre-geciktirme interrupt'ı	4 (1 olay başına)
	Döngüsel interrupt'lar	4 (1 olay başına)
	Donanım interrupt'ları	50 (1 olay başına)
	Süre hatası interrupt'ları	1
	Diyagnostik hatası interrupt'ları	1
	Çek veya tak modülleri	1
	Şasi veya istasyon arızası	1
Günün saati	Çoklu	

## Teknik özellikler

### A.3 CPU 1212C

Öge	Açıklama
	Durum
	1
	Güncelleme
	1
	Profile
	1
Zamanlayıcılar	Tip
	IEC
	Miktar
	Bellek boyutu ile sınırlıdır
	Depolama
	DB yapısı, zamanlayıcı başına 16 bayt
Sayıcılar	Tip
	IEC
	Miktar
	Bellek boyutu ile sınırlıdır
	Depolama
	DB yapısı, sayma tipine göre boyut
	<ul style="list-style-type: none"> <li>• SInt, USInt: 3 bayt</li> <li>• Int, UInt: 6 bayt</li> <li>• DInt, UDInt: 12 bayt</li> </ul>

### Çizelge A- 32 Haberleşme

Teknik veri	Açıklama
Portların numarası	1
Tip	Ethernet
HMI cihazı	3
Programlama cihazı (PG)	1
Bağlantılar	<ul style="list-style-type: none"> <li>• 8, açık kullanıcı haberleşmesi için (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV</li> <li>• 3, sunucu GET/PUT (CPU-ile-CPU) S7 haberleşme için</li> <li>• 8, istemci GET/PUT (CPU-ile-CPU) S7 haberleşme için</li> </ul>
Veri hızı	10/100 Mb/s
İzolasyon (harici sinyal ile PLC lojik arasında)	Transformator izoleli, 1500 VAC, sadece kısa süre olay güvenliği için
Kablo tipi	CAT5e zırhlı

### Çizelge A- 33 Güç kaynağı

Teknik veri	CPU 1212C AC/DC/Röle	CPU 1212C DC/DC/Röle	CPU 1212C DC/DC/DC
Gerilim aralığı	85 - 264 VAC	20.4 VDC - 28.8 VDC	
Hat frekansı	47 - 63 Hz	--	--
Giriş akımı (max. CPU sadece yük)	80 mA 120 VAC'de 40 mA 240 VAC'de	400 mA 24 VDC'de	400 mA 24 VDC'de
	Tüm genişleme donatıları ile CPU	240 mA 120 VAC'de 120 mA 240 VAC'de	1200 mA 24 VDC'de 1200 mA 24 VDC'de
Baskın akım (max.)	20 A 264 VAC'de	12 A 28.8 VDC'de	12 A 28.8 VDC'de
İzolasyon (giriş gücü ile lojik arasında )	1500 VAC	İzole edilmemiş	İzole edilmemiş
Toprak kaçacağı, AC hat ile fonksiyonel toprak arasında	0.5 mA max.	--	--

Teknik veri	CPU 1212C AC/DC/Röle	CPU 1212C DC/DC/Röle	CPU 1212C DC/DC/DC
Tutma süresi (güç kesilmesi)	20 ms 120 VAC'de 80 ms 240 VAC'de	10 ms 24 VDC'de	10 ms 24 VDC'de
Dahili sigorta, kullanıcı değiştiremez	3 A, 250 V, geç atmalı	3 A, 250 V, geç atmalı	3 A, 250 V, geç atmalı

Çizelge A- 34 Sensör gücü

Teknik veri	CPU 1212C AC/DC/Röle	CPU 1212C DC/DC/Röle	CPU 1212C DC/DC/DC
Gerilim aralığı	20.4 - 28.8 VDC	L+ eksi 4 VDC min.	
Çıkış akımı değeri (max.)	300 mA (kısa-devre korumalı)	300 mA (kısa-devre korumalı)	300 mA (kısa-devre korumalı)
Maksimum dalgalık gürültüsü(<10 MHz)	< 1 V tepeden- tepeye	Giriş hattı ile aynı	Giriş hattı ile aynı
İzolasyon (CPU lojik ile sensör gücü arasında)	İzole edilmemiş	İzole edilmemiş	İzole edilmemiş

### A.3.3 Dijital girişler ve çıkışlar

Çizelge A- 35 Dijital girişler

Teknik veri	CPU 1212C AC/DC/Röle, DC/DC/Röle ve DC/DC/DC
Giriş sayısı	8
Tip	Sink/Source (IEC Tip 1 sink)
Anma gerilimi	24 VDC 4 mA'de, nominal
Sürekli izin verilebilir gerilim	30 VDC, max.
Aşırı gerilim	35 VDC için 0.5 saniye
Lojik 1 sinyal (min.)	15 VDC 2.5 mA'de
Lojik 0 sinyal (max.)	5 VDC 1 mA'de
İzolasyon (alan tarafı ile lojik arasında)	500 VAC için 1 dakika
İzolasyon grupları	1
Filtre süreleri	us ayarlar: 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0 ms ayarlar: 0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0
HSC saat giriş hızları (max.) (Lojik 1 seviye = 15 - 26 VDC)	100/80 kHz (Ia.0 - Ia.5) 30 /20 kHz (Ia.6 - Ia.7)
Aynı anlı girişlerin sayısı	4 (bitişik noktalar değil), 60 °C'da yatay veya 50 °C dikey 8, 55 °C'da yatay veya 45 °C dikey
Kablo uzunluğu (metre)	500 m zırlı, 300 m zırlısız, 50 m zırlı için HSC girişleri

## Teknik özellikler

## A.3 CPU 1212C

Çizelge A- 36 Dijital çıkışlar

Teknik veri	CPU 1212C AC/DC/Röle ve DC/DC/Röle	CPU 1212C DC/DC/DC
Giriş sayısı	6	6
Tip	Röle, kuru kontak	Yarı iletken- MOSFET (Source)
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	--	20 VDC min.
Lojik 0 sinyal, 10 KΩ yük ile	--	0.1 VDC max.
Akım (max.)	2.0 A	0.5 A
Lamba yük	30 W DC / 200 W AC	5 W
ON durum direnci	0.2 Ω max. yeni olduğunda	0.6 Ω max.
Nokta başına kaçak akım	--	10 µA max.
Ani aşırı akım	7 A kontakların kapanmasıyla	8 A için 100 ms max.
Aşırı yük koruması	Hayır	Hayır
İzolasyon (alan tarafı ile lojik arasında)	1500 VAC için 1 dakika (bobin ve kontak) Hiçbiri (bobin ile lojik arasında)	500 VAC için 1 dakika
İzolasyon direnci	100 MΩ min. yeni olduğunda	--
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika	--
İzolasyon grupları	2	1
İndüktif kenetleme gerilimi	--	L+ eksi 48 VDC, 1 W ısı olarak kayıp
Anahtarlama gecikmesi (Qa.0 - Qa.3)	10 ms max.	1.0 µs max., off - on geçişi 3.0 µs max., on - off geçişi
Anahtarlama gecikmesi (Qa.4 - Qa.5)	10 ms max.	50 µs max., off - on geçişi 200 µs max., on - off geçişi
Maksimum röle anahtarlama frekansı	1 Hz	--
Darbe katari çıkış hızı	Önerilmez <sup>1</sup>	100 kHz (Qa.0 - Qa.3) <sup>2</sup> , 2 Hz min. 20 kHz (Qa.4 - Qa.5) <sup>2</sup>
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler	--
Anma yükünde ömür boyu temaslar	100,000 açık/kapalı döngüler	--
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	3, (bitişik noktalar değil) 60 °C'da yatay veya 50 °C dikey 6, 55 °C'da yatay veya 45 °C dikey	
Kablo uzunluğu (metre)	500 m zırhlı, 150 m zırhsız	500 m zırhlı, 150 m zırhsız

<sup>1</sup> Röle çıkışları olan CPU modellerinde, darbe çıkışlarını kullanmak için bir dijital sinyal kartı (SB) takmanız gerekir.

<sup>2</sup> Darbe alıcı ve kablonuza bağlı olarak ilave bir yük direnci (anma akımının en az %10'nunda) darbe sinyal kalitesini ve gürültü bağışıklığını iyileştirebilir.

### A.3.4 Analog girişler

Çizelge A- 37 Analog girişler

Teknik veri	Açıklama
Giriş sayısı	2
Tip	Gerilim (tek-sonlu)
Tam-ölçek aralığı	0 - 10 V
Tam-ölçek aralığı (veri word)	0 - 27648
Aşma aralığı	10.001 - 11.759 V
Aşma aralığı (veri word)	27649 - 32511
Taşma aralığı	11.760 - 11.852 V
Taşma aralığı (veri word)	32512 - 32767
Çözünürlük	10 bit
Maksimum dayanma gerilimi	35 VDC
Düzleme	Hiçbiri, Zayıf, Orta veya Güçlü CPU'nun analog girişleri için basamak tepkisi çizelgesine bakınız (Sayfa 852).
Gürültü bastırma	10, 50 veya 60 Hz
Empedans	≥100 KΩ
İzolasyon (alan tarafı ile lojik)	Hiçbiri
Doğruluk (25 °C / 0 - 55 °C)	Tam-ölçeğe göre %3.0 / %3.5
Kablo uzunluğu (metre)	100 m, zırlı bükülü çift

#### A.3.4.1 CPU yerleşik analog girişlerinin basamak tepkisi

Çizelge A- 38 Basamak tepkisi (ms), 0 V -10 V, 95% 'deki ölçüm

Düzleme seçimi (örnek ortalama)	Bastırma frekansı (İntegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Hiçbiri (1 döngü): Ortalama alma yok	50 ms	50 ms	100 ms
Zayıf (4 döngü): 4 örnek	60ms	70 ms	200 ms
Orta (16 döngü): 16 örnek	200 ms	240 ms	1150 ms
Güçlü (32 döngü): 32 örnek	400 ms	480 ms	2300 ms
<b>Örnekleme süresi</b>	<b>4.17 ms</b>	<b>5 ms</b>	<b>25 ms</b>

**A.3.4.2 CPU'nun yerleşik analog portları için örnekleme süresi**

Çizelge A- 39 CPU'nun yerleşik analog portları için örnekleme süresi

Bastırma frekansı (İntegrasyon süresi seçimi)	Örnekleme süresi
60 Hz (16.6 ms)	4.17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

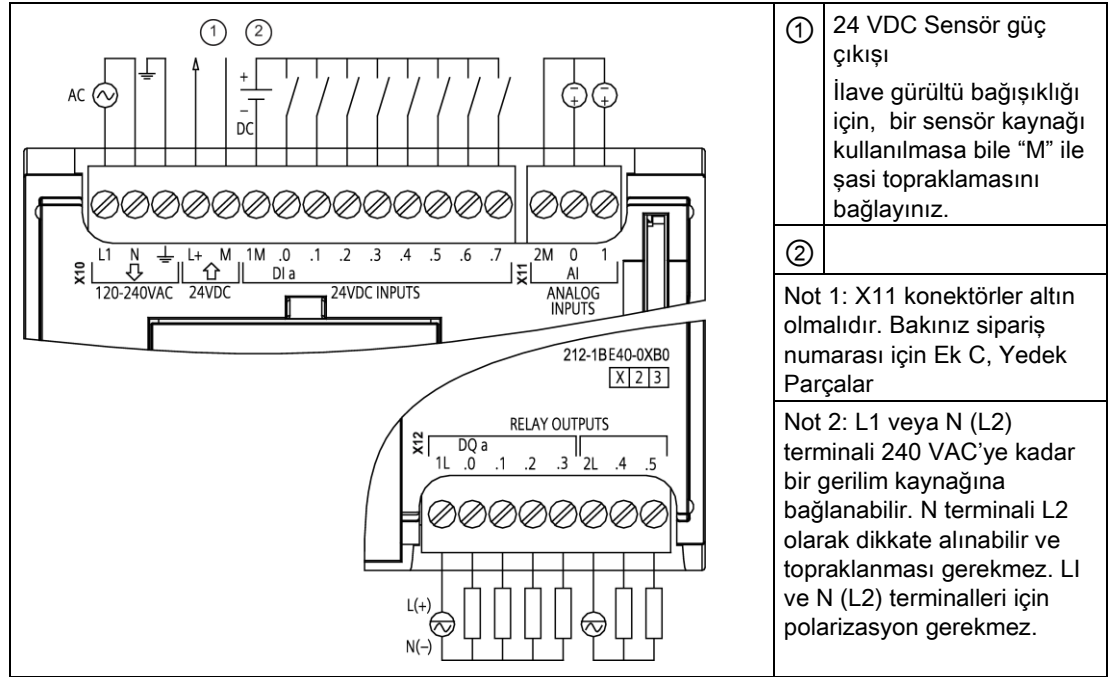
**A.3.4.3 Gerilim (CPU'lar) için analog giriş gösterilişi**

Çizelge A- 40 Gerilim (CPU'lar) için analog giriş gösterilişi

Sistem		Gerilim Ölçme Aralığı	
Decimal	Hexadecimal	0 - 10 V	
32767	7FFF	11.851 V	Taşma
32512	7F00		
32511	7EFF	11.759 V	Aşma aralığı
27649	6C01		
27648	6C00	10 V	Anma aralığı
20736	5100	7.5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

### A.3.5 CPU 1212C bağlantı diyagramları

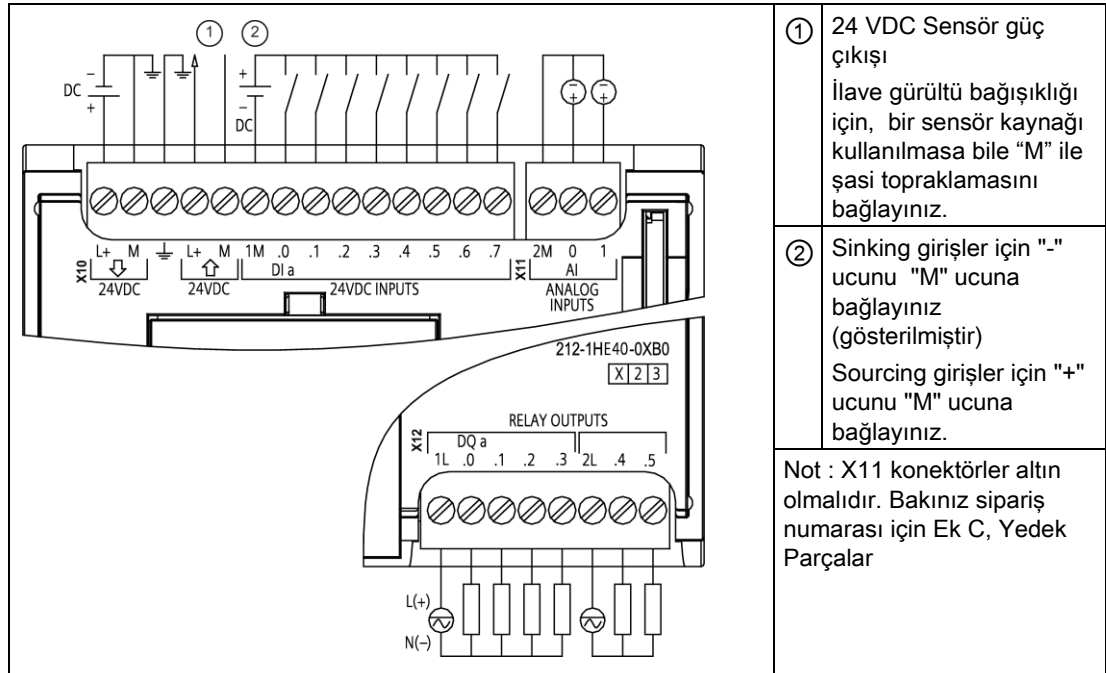
Çizelge A- 41 CPU 1212C AC/DC/Röle (6ES7 212-1BE40-0XB0)



Çizelge A- 42 CPU 1212C AC/DC/Röle (6ES7 212-1BE40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L1 / 120-240 VAC	2 M	1L
2	N / 120-240 VAC	AI 0	DQ a.0
3	Fonksiyonel toprak	AI 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.2
5	M / 24 VDC Sensör çıkışı	--	DQ a.3
6	1M	--	2L
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

Çizelge A- 43 CPU 1212C DC/DC/Röle (6ES7 212-1HE40-0XB0)

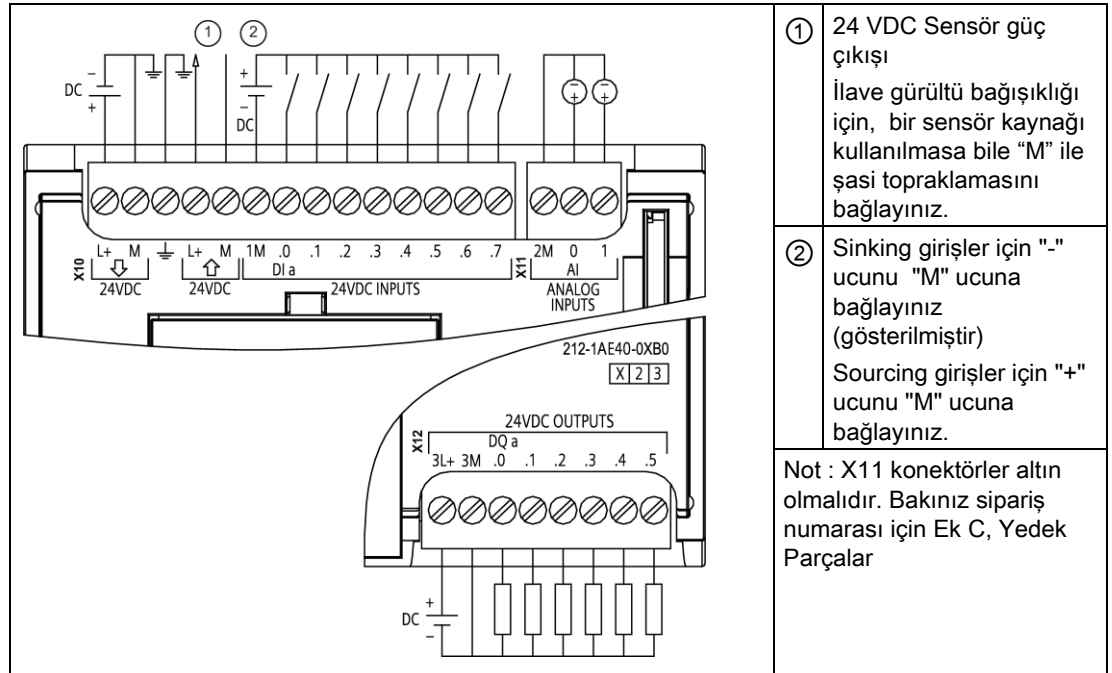


Çizelge A- 44 CPU 1212C DC/DC/Röle (6ES7 212-1HE40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L+ / 24 VDC	2 M	1L
2	M / 24 VDC	AI 0	DQ a.0
3	Fonksiyonel toprak	AI 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.2
5	M / 24 VDC Sensör çıkışı	--	DQ a.3
6	1M	--	2L
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--



Çizelge A- 45 CPU 1212C DC/DC/DC (6ES7-212-1AE40-0XB0)



Çizelge A- 46 CPU 1212C DC/DC/DC (6ES7 212-1AE40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L+ / 24 VDC	2 M	3L+
2	M / 24 VDC	AI 0	3M
3	Fonksiyonel toprak	AI 1	DQ a.0
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.1
5	M / 24 VDC Sensör çıkışı	--	DQ a.2
6	1M	--	DQ a.3
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	--
10	DI a.3	--	--
11	DI a.4	--	--
12	DI a.5	--	--
13	DI a.6	--	--
14	DI a.7	--	--

**Not**

Kullanılmayan analog girişler topraklanmalıdır.

## A.4 CPU 1214C

### A.4.1 Genel açıklamalar ve özellikler

Çizelge A- 47 Genel

Teknik veri	CPU 1214C AC/DC/Röle	CPU 1214C DC/DC/Röle	CPU 1214C DC/DC/DC
Sipariş numarası	6ES7 214-1BG40-0XB0	6ES7 214-1HG40-0XB0	6ES7 214-1AG40-0XB0
Boyutlar W x H x D (mm)	110 x 100 x 75	110 x 100 x 75	110 x 100 x 75
Nakliye ağırlığı	475 gram	435 gram	415 gram
Güç kaybı	14 W	12 W	12 W
Akım kullanılabilir(SM ve CM bus)	1600 mA max. (5 VDC)	1600 mA max. (5 VDC)	1600 mA max. (5 VDC)
Akım kullanılabilir(24 VDC)	400 mA max. (sensör gücü)	400 mA max. (sensör gücü)	400 mA max. (sensör gücü)
Dijital girişin çektiği akım (24 VDC)	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı

Çizelge A- 48 CPU özellikleri

Teknik veri	Açıklama	
Kullanıcı belleği	Çalışma	75 Kbayt
	Yükleme	4 Mbayt dahili, SD kartı boyutuna kadar genişletilebilir
	Kalıcı	10 Kbayt
Yerleşik dijital I/O	14 giriş/10 çıkış	
Yerleşik analog I/O	2 giriş	
Proses görüntü boyutu	1024 girişlerin baytları (I)/1024 çıkışların baytları (Q)	
Bit belleği(M)	8192 bayt	
Geçici (lokal) bellek	<ul style="list-style-type: none"> <li>16 Kbayt, başlangıç ve program döngüsü için (ilişkili FB'ler ve FC'ler dahil)</li> <li>6 Kbayt, diğer interrupt (FB'ler ve FC'ler dahil) öncelik seviyeleri için</li> </ul>	
Sinyal modülleri genişlemesi	8 SM max.	
SB, CB, BB genişlemesi	1 max.	
Haberleşme modülü genişlemesi	3 CM max.	
Yüksek-hızlı sayıcılar	6'ya kadar, yerleşik veya SB girişlerinde kullanılmak üzere konfigüre edilmiş. Bakınız Çizelge, CPU 1214C: HSC varsayılan adres atamaları (Sayfa 411). <ul style="list-style-type: none"> <li>100/180 kHz (Ia.0 - Ia.5)</li> <li>30/120 kHz (Ia.6 - Ib.5)</li> </ul>	

Teknik veri	Açıklama
Darbe çıkışları <sup>2</sup>	6'ya kadar, yerleşik veya SB çıkışlarında kullanılmak üzere konfigüre edilmiş. <ul style="list-style-type: none"> <li>• 100 kHz (Qa.0 - Qa.3)</li> <li>• 30 kHz (Qa.4 - Qb.1)</li> </ul>
Darbe yakalama girişleri	14
Süre gecikme interrupt'ları	Toplam 4, 1 ms çözünürlük ile
Döngüsel interrupt'lar	Toplam 4, 1 ms çözünürlük ile
Kenar interrupt'ları	12 yükselme ve 12 düşme (16 ve seçmeli sinyal kartlı 16)
Bellek kartı	SIMATIC Bellek kartı (opsiyonel)
Gerçek zaman saat doğruluğu	+/- 60 saniye/ay
Gerçek zaman saat tutma süresi	20 gün tipik/12 gün min. 40 °C'ta (bakımsız Süper Kapasitör)

<sup>1</sup> HSC dördün çalışma moduna konfigüre edildiğinde daha düşük hız uygulanabilir.

<sup>2</sup> Röle çıkışları olan CPU modellerinde, darbe çıkışlarını kullanmak için bir dijital sinyal kartı (SB) takmanız gerekir.

#### Çizelge A- 49 Performans

Komut tipi	Yürütme hızı
Boolean	0.08 µs/komut
Move Word	1.7 µs/komut
Real math	2.3 µs/komut

## A.4.2 CPU 1214C tarafından desteklenen zamanlayıcılar, sayıcılar ve kod blokları

#### Çizelge A- 50 CPU 1214C tarafından desteklenen bloklar, zamanlayıcılar ve sayıcılar

Öğe	Açıklama	
Bloklar	Tip	OB, FB, FC, DB
	Boyut	64 Kbayt
	Miktar	Tarafından desteklenen bloklar (OB'ler + FB'ler + FC'ler + DB'ler)
	FB'ler, FC'ler ve DB'ler için adres aralığı	FB ve FC: 1 - 65535 ( FB 1 - FB 65535 gibi) DB: 1 - 59999
	Gömme derinliği	16, program döngüsü veya başlangıç OB'sinden 6, herhangi bir interrupt olayı OB'sinden
	İzleme	2 kod bloklarının durumu aynı anda izlenebilir
OB'lar	Program döngüsü	Çoklu
	Başlangıç	Çoklu
	Süre-geciktirme interrupt'ları	4 (1 olay başına)
	Döngüsel interrupt'lar	4 (1 olay başına)
	Donanım interrupt'ları	50 (1 olay başına)
	Süre hatası interrupt'ları	1
	Diyagnostik hatası interrupt'ları	1

## Teknik özellikler

## A.4 CPU 1214C

Öge	Açıklama	
	Çek veya tak modülleri	1
	Şasi veya istasyon arızası	1
	Günün saati	Çoklu
	Durum	1
	Güncelleme	1
	Profile	1
Zamanlayıcılar	Tip	IEC
	Miktar	Bellek boyutu ile sınırlıdır
	Depolama	DB yapısı, zamanlayıcı başına 16 bayt
Sayıcılar	Tip	IEC
	Miktar	Bellek boyutu ile sınırlıdır
	Depolama	DB yapısı, sayma tipine göre boyut <ul style="list-style-type: none"> <li>• SInt, USInt: 3 bayt</li> <li>• Int, UInt: 6 bayt</li> <li>• DInt, UDInt: 12 bayt</li> </ul>

## Çizelge A- 51 Haberleşme

Teknik veri	Açıklama
Portların numarası	1
Tip	Ethernet
HMI cihazı	3
Programlama cihazı (PG)	1
Bağlantılar	<ul style="list-style-type: none"> <li>• 8 , açık kullanıcı haberleşmesi için (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV</li> <li>• 3, server GET/PUT (CPU-ile-CPU) S7 haberleşme için</li> <li>• 8, istemci GET/PUT (CPU-ile-CPU) S7 haberleşme için</li> </ul>
Veri hızları	10/100 Mb/s
İzolasyon (harici sinyal ile PLC lojiği arasında)	Transformator izoleli, 1500 VAC, sadece kısa süreli olay güvenliği için
Kablo tipi	CAT5e zırhlı

## Çizelge A- 52 Güç kaynağı

Teknik veri	CPU 1214C AC/DC/Röle	CPU 1214C DC/DC/Röle	CPU 1214C DC/DC/DC
Gerilim aralığı	85 - 264 VAC	20.4 VDC to 28.8 VDC	
Hat frekansı	47 - 63 Hz	--	
Giriş akımı (max. yük)	CPU sadece	100 mA 120 VAC'de 50 mA 240 VAC'de	500 mA 24 VDC'de
	Tüm genişleme donatıları ile CPU	300 mA 120 VAC'de 150 mA 240 VAC'de	1500 mA 24 VDC'de
Baskın akım (max.)	20 A 264 VAC'de	12 A 28.8 VDC'de	

Teknik veri	CPU 1214C AC/DC/Röle	CPU 1214C DC/DC/Röle	CPU 1214C DC/DC/DC
İzolasyon (giriş gücü ile lojik arasında)	1500 VAC	İzole edilmemiş	
Toprak kaçacağı, AC hat ile fonksiyonel toprak arasında	0.5 mA max.	-	
Tutma süresi (güç kesilmesi)	20 ms 120 VAC'de 80 ms 240 VAC'de	10 ms 24 VDC'de	
Dahili sigorta, kullanıcı değiştiremez	3 A, 250 V, geç atmalı		

Çizelge A- 53 Sensör gücü

Teknik veri	CPU 1214C AC/DC/Röle	CPU 1214C DC/DC/Röle	CPU 1214C DC/DC/DC
Gerilim aralığı	20.4 - 28.8 VDC	L+ eksi 4 VDC min.	
Çıkış akımı değeri (max.)	400 mA (kısa-devre korumalı)		
Maksimum dalgalık gürültüsü(<10 MHz)	< 1 V tepeden- tepeye	Giriş hattı ile aynı	
İzolasyon (CPU lojigi ile sensör gücü arasında)	İzole edilmemiş		

### A.4.3 Dijital girişler ve çıkışlar

Çizelge A- 54 Dijital girişler

Teknik veri	CPU 1214C AC/DC/Röle	CPU 1214C DC/DC/Röle	CPU 1214C DC/DC/DC
Giriş sayısı	14		
Tip	Sink/Source (IEC Tip 1 sink)		
Anma gerilimi	24 VDC 4 mA'de, nominal		
Sürekli izin verilebilir gerilim	30 VDC, max.		
Aşırı gerilim	35 VDC için 0.5 saniye		
Lojik 1 sinyal (min.)	15 VDC 2.5 mA'de		
Lojik 0 sinyal (max.)	5 VDC 1 mA'de		
İzolasyon (alan tarafı ile lojik arasında)	500 VAC için 1 dakika		
İzolasyon grupları	1		
Filtre süreleri	us ayarlar: 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0 ms ayarlar: 0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0		
HSC saat giriş hızları (max.) (Lojik 1 seviye = 15 - 26 VDC)	100/80 kHz (Ia.0 - Ia.5) 30/20 kHz (Ia.6 - Ib.5)		
Aynı anlı girişlerin sayısı	<ul style="list-style-type: none"> <li>7 (bitişik noktalar değil) 60 °C'ta yatay veya 50 °C dikey</li> <li>14 55 °C'ta yatay veya 45 °C dikey</li> </ul>		
Kablo uzunluğu (metre)	500 m zırhlı, 300 m zırhsız, 50 m zırhlı için HSC girişler		

## Teknik özellikler

## A.4 CPU 1214C

Çizelge A- 55 Dijital çıkışlar

Teknik veri	CPU 1214C AC/DC/Röle ve DC/DC/Röle	CPU 1214C DC/DC/DC
Giriş sayısı	10	10
Tip	Röle, kuru kontak	Yarı iletken- MOSFET (Source)
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	--	20 VDC min.
Lojik 0 sinyal, 10 KΩ yük ile	--	0.1 VDC max.
Akım (max.)	2.0 A	0.5 A
Lamba yük	30 W DC / 200 W AC	5 W
ON durum direnci	0.2 Ω max. yeni olduğunda	0.6 Ω max.
Nokta başına kaçak akım	--	10 µA max.
Ani aşırı akım	7 A kontakların kapanmasıyla	8 A için 100 ms max.
Aşırı yük koruması	Hayır	Hayır
İzolasyon (alan tarafı ile lojik)	1500 VAC için 1 dakika (bobin ve kontak) Hiçbiri (bobin ile lojik)	500 VAC için 1 dakika
İzolasyon direnci	100 MΩ min. yeni olduğunda	--
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika	--
İzolasyon grupları	2	1
İndüktif kenetleme gerilimi	--	L+ eksi 48 VDC, 1 W ısı olarak kayıp
Anahtarlama gecikmesi (Qa.0 - Qa.3)	10 ms max.	1.0 µs max., off - on geçişi 3.0 µs max., on - off geçişi
Anahtarlama gecikmesi (Qa.4 - Qb.1)	10 ms max.	50 µs max., off - on geçişi 200 µs max., on - off geçişi
Maksimum röle anahtarlama frekansı	1 Hz	--
Darbe katarı çıkış hızı	Önerilmez <sup>1</sup>	100 kHz (Qa.0 - Qa.3) <sup>2</sup> , 2 Hz min. 20 kHz (Qa.4 - Qb.1) <sup>2</sup>
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler	--
Anma yükünde ömür boyu temaslar	100,000 açık/kapalı döngüler	--
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	
Aynı andaki çıkış sayısı	<ul style="list-style-type: none"> <li>5 (bitişik noktalar değil), 60 °C'ta yatay veya 50 °C dikey</li> <li>10, 55 °C'ta yatay veya 45 °C dikey</li> </ul>	
Kablo uzunluğu (metre)	500 m zırhlı, 150 m zırhsız	

<sup>1</sup> Röle çıkışları olan CPU modellerinde, darbe çıkışlarını kullanmak için bir dijital sinyal kartı (SB) takmanız gerekir.

<sup>2</sup> Darbe alıcı ve kablonuza bağlı olarak ilave bir yük direnci (anma akımının en az %10'nunda) darbe sinyal kalitesini ve gürültü bağışıklığını iyileştirebilir.

#### A.4.4 Analog girişler

Çizelge A- 56 Analog girişler

Teknik veri	Açıklama
Giriş sayısı	2
Tip	Gerilim (tek-sonlu)
Tam-ölçek aralığı	0 - 10 V
Tam-ölçek aralığı (veri word)	0 - 27648
Aşma aralığı	10.001- 11.759 V
Aşma aralığı (veri word)	27649 - 32511
Taşma aralığı	11.760 - 11.852 V
Taşma aralığı (veri word)	32512 - 32767
Çözünürlük	10 bit
Maksimum dayanma gerilimi	35 VDC
Düzleme	Hiçbiri, Zayıf, Orta veya Güçlü CPU analog girişlerinin basamak tepkileri (ms) için Çizelgeye bakınız (Sayfa 862).
Gürültü bastırma	10, 50 veya 60 Hz
Empedans	≥100 KΩ
İzolasyon (alan tarafı ile lojik)	Hiçbiri
Doğruluk (25 °C / 0 - 55 °C)	Tam-ölçeğe göre %3.0 / %3.5
Kablo uzunluğu (metre)	100 m, zırlı bükülü çift

##### A.4.4.1 CPU yerleşik analog girişleri için basamak tepkisi

Çizelge A- 57 Basamak tepkisi (ms), 0 V - 10 V %95'e göre ölçülen

Düzleme seçimi (örnek ortalama)	Bastırma frekansı (İntegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Hiçbiri (1 döngü): Ortalama alma yok	50 ms	50 ms	100 ms
Zayıf (4 döngü): 4 örnek	60ms	70 ms	200 ms
Orta (16 döngü): 16 örnek	200 ms	240 ms	1150 ms
Güçlü (32 döngü): 32 örnek	400 ms	480 ms	2300 ms
<b>Örnekleme süresi</b>	<b>4.17 ms</b>	<b>5 ms</b>	<b>25 ms</b>

**A.4.4.2 CPU yerleşik analog girişleri için örnekleme süresi**

Çizelge A- 58 CPU yerleşik analog girişleri için örnekleme süresi

Bastırma frekansı (İntegrasyon süresi seçimi)	Örnekleme süresi
60 Hz (16.6 ms)	4.17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

**A.4.4.3 Gerilim (CPU'lar) için analog girişlerin ölçme aralıkları**

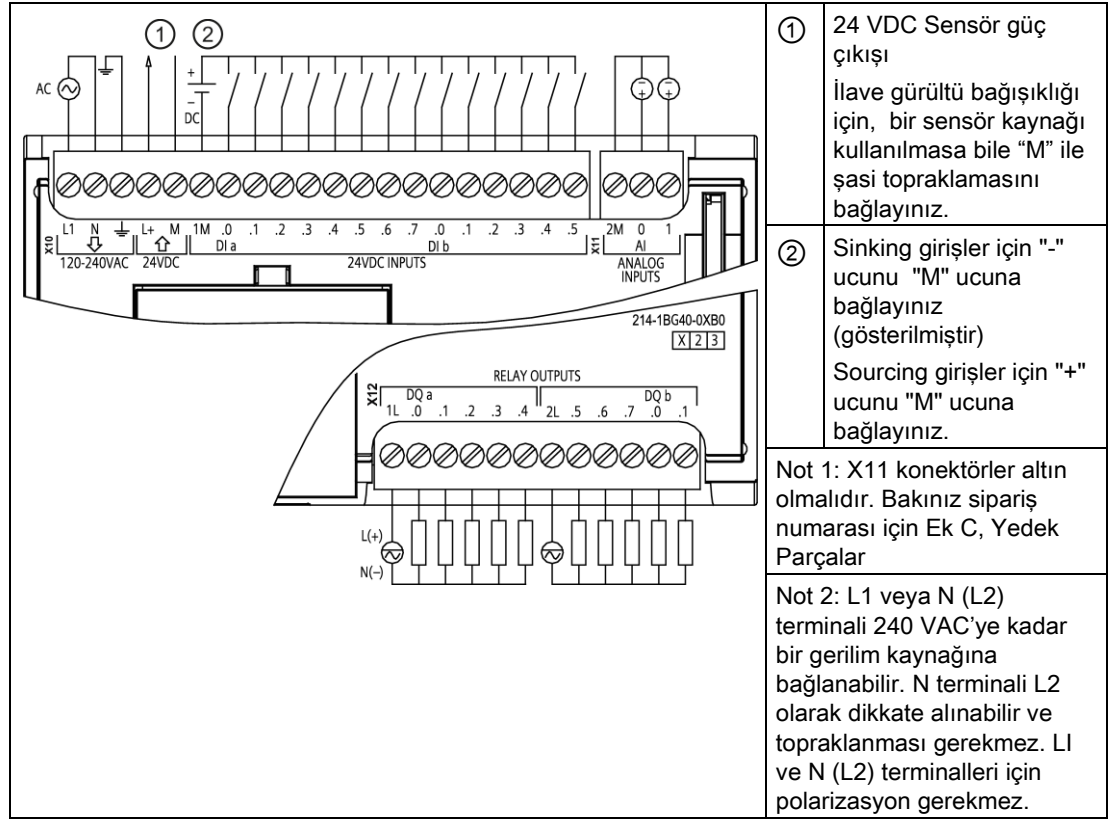
Çizelge A- 59 Gerilim (CPU'lar) için analog girişlerin ölçme aralıkları

Sistem		Gerilim Ölçme Aralığı	
Decimal	Hexadecimal	0 - 10 V	
32767	7FFF	11.851 V	Taşma
32512	7F00		
32511	7EFF	11.759 V	Aşma aralığı
27649	6C01		Anma aralığı
27648	6C00	10 V	
20736	5100	7.5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	



### A.4.5 CPU 1214C bağlantı diyagramları

Çizelge A- 60 CPU 1214C AC/DC/Röle (6ES7 214-1BG40-0XB0)

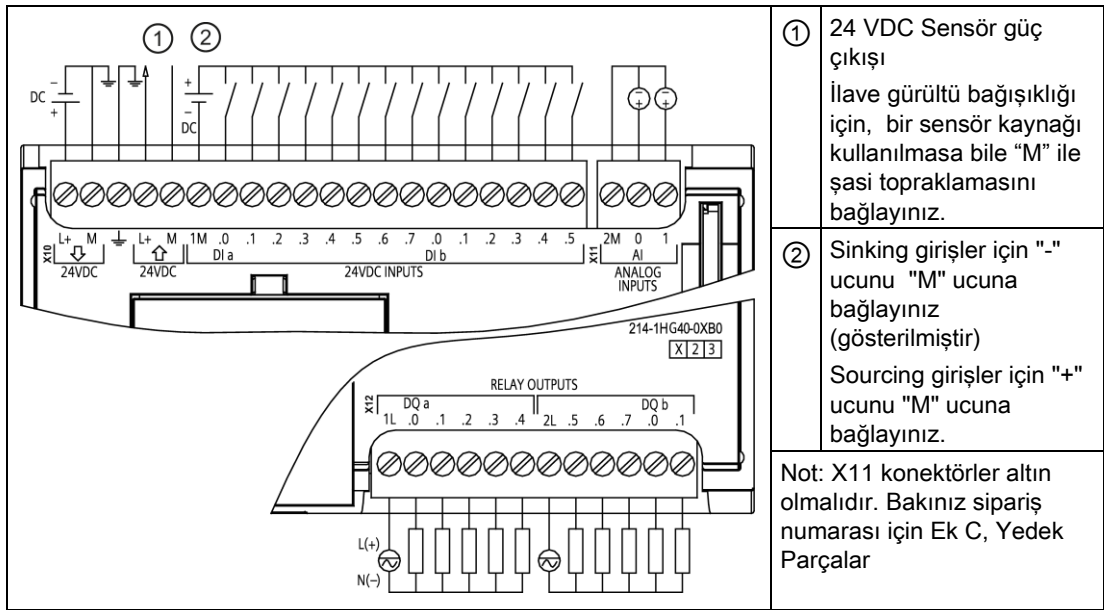


Çizelge A- 61 CPU 1214C AC/DC/Röle (6ES7 214-1BG40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L1 / 120-240 VAC	2 M	1L
2	N / 120-240 VAC	AI 0	DQ a.0
3	Fonksiyonel toprak	AI 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.2
5	M / 24 VDC Sensör çıkışı	--	DQ a.3
6	1M	--	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--

Pin	X10	X11 (altın)	X12
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Çizelge A- 62 CPU 1214C DC/DC/Röle (6ES7 214-1HG40-0XB0)

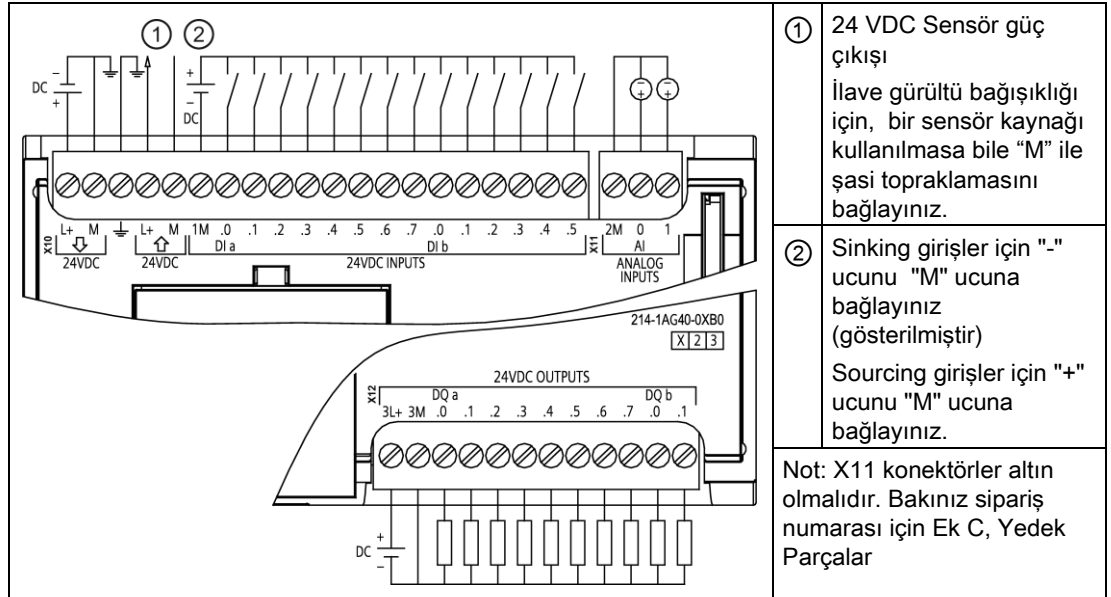


Çizelge A- 63 CPU 1214C DC/DC/Röle (6ES7 214-1HG40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L+ / 24 VDC	2 M	1L
2	M / 24 VDC	AI 0	DQ a.0
3	Fonksiyonel toprak	AI 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.2
5	M / 24 VDC Sensör çıkışı	--	DQ a.3
6	1M	--	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1

Pin	X10	X11 (altın)	X12
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Çizelge A- 64 CPU 1214C DC/DC/DC (6ES7 214-1AG40-0XB0)



Çizelge A- 65 CPU 1214C DC/DC/DC (6ES7 214-1AG40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L+ / 24 VDC	2 M	3L+
2	M / 24 VDC	AI 0	3M
3	Fonksiyonel toprak	AI 1	DQ a.0
4	L+ / 24 VDC Sensör çıkışı	--	DQ a.1
5	M / 24 VDC Sensör çıkışı	--	DQ a.2
6	1M	--	DQ a.3
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0

## Teknik özellikler

### A.5 CPU 1215C

Pin	X10	X11 (altın)	X12
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	-
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

#### Not

Kullanılmayan analog girişler topraklanmalıdır.

## A.5 CPU 1215C

### A.5.1 Genel açıklamalar ve özellikler

Çizelge A- 66 Genel

Teknik veri	CPU 1215C AC/DC/Röle	CPU 1215C DC/DC/Röle	CPU 1215C DC/DC/DC
Sipariş numarası	6ES7 215-1BG40-0XB0	6ES7 215-1HG40-0XB0	6ES7 215-1AG40-0XB0
Boyutlar W x H x D (mm)	130 x 100 x 75	130 x 100 x 75	130 x 100 x 75
Nakliye ağırlığı	585 gram	550 gram	520 gram
Güç kaybı	14 W	12 W	12 W
Akım kullanılabilir(SM ve CM bus)	1600 mA max. (5 VDC)	1600 mA max. (5 VDC)	1600 mA max. (5 VDC)
Akım kullanılabilir(24 VDC)	400 mA max. (Sensör gücü)	400 mA max. (Sensör gücü)	400 mA max. (Sensör gücü)
Dijital girişin çektiği akım(24 VDC)	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı	4 mA/giriş kullanıldı

Çizelge A- 67 CPU özellikleri

Teknik veri	Açıklama	
Kullanıcı belleği	Çalışma	100 Kbayt
	Yükleme	4 Mbayt, dahili, SD kartı boyutuna kadar genişletilebilir
	Kalıcı	10 Kbayt

Teknik veri	Açıklama
Yerleşik dijital I/O	14 giriş/10 Çıkış
Yerleşik analog I/O	2 giriş/2 Çıkış
Proses görüntü boyutu	1024 girişlerin baytları (I)/1024 çıkışların baytları (Q)
Bit belleği(M)	8192 bayt
Geçici (lokal) bellek	<ul style="list-style-type: none"> <li>16 Kbayt, başlangıç ve program döngüsü için (ilişkili FB'ler ve FC'ler dahil)</li> <li>6 Kbayt, diğer interrupt öncelik seviyelerinin her biri için (FB'ler ve FC'ler dahil)</li> </ul>
Sinyal modülleri genişlemesi	8 SM max.
SB, CB, BB genişlemesi	1 max.
Haberleşme modülü genişlemesi	3 CM max.
Yüksek-hızlı sayıcılar	6'ya kadar, yerleşik veya SB girişlerinde kullanılmak üzere konfigüre edilmiş. Bakınız Çizelge, CPU 1215C: HSC varsayılan adres atamaları. <ul style="list-style-type: none"> <li>100/180 kHz (Ia.0 - Ia.5)</li> <li>30/120 kHz (Ia.6 - Ib.5)</li> </ul>
Darbe çıkışları <sup>2</sup>	4'e kadar, her yerleşik veya SB çıkışını kullanmak için konfigüre edilen <ul style="list-style-type: none"> <li>100 kHz (Qa.0 - Qa.3)</li> <li>30 kHz (Qa.4 - Qb.1)</li> </ul>
Darbe yakalama girişleri	14
Süre gecikme interrupt'ları	Toplam 4, 1 ms çözünürlük ile
Döngüsel interrupt'lar	Toplam 4, 1 ms çözünürlük ile
Kenar interrupt'ları	12 yükselme ve 12 düşme (16 ve opsiyonel sinyal kartı olan16)
Bellek kartı	SIMATIC Bellek kartı (opsiyonel)
Gerçek zaman saat doğruluğu	+/- 60 saniye/ay
Gerçek zaman saat tutma süresi	20 gün tipik/12 gün min. 40 °C'ta (bakımsız Süper Kapasitör)

<sup>1</sup> HSC döndür çalışma moduna konfigüre edildiğinde daha düşük hız uygulanabilir.

<sup>2</sup> Röle çıkışları olan CPU modellerinde, darbe çıkışlarını kullanmak için bir dijital sinyal kartı (SB) takmanız gerekir.

#### Çizelge A- 68 Performans

Komut tipi	Yürütme hızı
Boolean	0.08 µs/komut
Move Word	1.7 µs/komut
Real math	2.3 µs/komut

## A.5.2 CPU 1215C tarafından desteklenen zamanlayıcılar, sayıcılar ve kod blokları

Çizelge A- 69 CPU 1215C tarafından desteklenen bloklar, zamanlayıcılar ve sayıcılar

Öge	Açıklama	
Bloklar	Tip	OB, FB, FC, DB
	Boyut	64 Kbayt
	Miktar	Tarafından desteklenen bloklar (OB'ler + FB'ler + FC'ler + DB'ler)
	FB'ler, FC'ler ve DB'ler için adres aralığı	FB ve FC: 1 - 65535 ( FB 1 - FB 65535 gibi) DB: 1 - 59999
	Gömme derinliği	16, program döngüsü veya başlangıç OB'sinden 6, herhangi bir interrupt olayı OB'sinden
	İzleme	2, kod bloklarının durumu aynı anda izlenebilir
	OB'lar	Program döngüsü
Başlangıç		Çoklu
Süre-geciktirme interrupt'ları		4 (1 olay başına)
Döngüsel interrupt'lar		4 (1 olay başına)
Donanım interrupt'ları		50 (1 olay başına)
Süre hatası interrupt'ları		1
Diyagnostik hatası interrupt'ları		1
Çek veya tak modülleri		1
Şasi veya istasyon arızası		1
Günün saati		Çoklu
Durum		1
Güncelleme		1
Profile		1
Zamanlayıcılar	Tip	IEC
	Miktar	Bellek boyutu ile sınırlıdır
	Depolama	DB yapısı, 16 bayt zamanlayıcı başına
Sayıcılar	Tip	IEC
	Miktar	Bellek boyutu ile sınırlıdır
	Depolama	DB yapısı, sayma tipine göre boyut <ul style="list-style-type: none"> <li>• SInt, USInt: 3 bayt</li> <li>• Int, UInt: 6 bayt</li> <li>• DInt, UDInt: 12 bayt</li> </ul>

## Çizelge A- 70 Haberleşme

Teknik veri	Açıklama
Portların numarası	2
Tip	Ethernet
HMI cihazı	3
Programlama cihazı (PG)	1
Bağlantılar	<ul style="list-style-type: none"> <li>8, açık kullanıcı haberleşmesi için (aktif veya pasif):TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV</li> <li>3, server GET/PUT (CPU-ile-CPU) S7 haberleşme için</li> <li>8, istemci GET/PUT (CPU-ile-CPU) S7 haberleşme için</li> </ul>
Veri hızları	10/100 Mb/s
İzolasyon (PLC lojiğe harici sinyal)	Transformatör izoleli, 1500 VAC, sadece kısa süreli olay güvenliği için
Kablo tipi	CAT5e zırlı

## Çizelge A- 71 Güç kaynağı

Teknik veri	CPU 1215C AC/DC/Röle	CPU 1215C DC/DC/Röle	CPU 1215C DC/DC/DC
Gerilim aralığı	85 - 264 VAC	20.4 VDC - 28.8 VDC	
Hat frekansı	47 - 63 Hz	--	
Giriş akımı (max. yük)	CPU sadece 100 mA 120 VAC'de 50 mA 240 VAC'de	500 mA 24 VDC'de	
	Tüm genişleme donatıları ile CPU 300 mA 120 VAC'de 150 mA 240 VAC'de	1500 mA 24 VDC'de	
Baskın akım (max.)	20 A 264 VAC'de	12 A 28.8 VDC'de	
İzolasyon (giriş gücü ile lojik arasında)	1500 VAC	İzole edilmemiş	
Toprak kaçacağı, AC hat ile fonksiyonel toprak arasında	0.5 mA max.	-	
Tutma süresi (güç kesilmesi)	20 ms 120 VAC'de 80 ms 240 VAC'de	10 ms 24 VDC'de	
Dahili sigorta, kullanıcı değiştiremez	3 A, 250 V, geç atmalı		

## Çizelge A- 72 Sensör gücü

Teknik veri	CPU 1215C AC/DC/Röle	CPU 1215C DC/DC/Röle	CPU 1215C DC/DC/DC
Gerilim aralığı	20.4 - 28.8 VDC	L+ eksi 4 VDC min.	
Çıkış akımı değeri (max.)	400 mA (kısa-devre korumalı)		
Maksimum ripil gürültüsü(<10 MHz)	< 1 V tepeden- tepeye	Giriş hattı ile aynı	
İzolasyon (CPU lojiği ile sensör gücü arasında)	İzole edilmemiş		

## Teknik özellikler

### A.5 CPU 1215C

#### A.5.3 Dijital girişler ve çıkışlar

Çizelge A- 73 Dijital girişler

Teknik veri	CPU 1215C AC/DC/Röle	CPU 1215C DC/DC/Röle	CPU 1215C DC/DC/DC
Giriş sayısı	14		
Tip	Sink/Source (IEC Tip 1 sink)		
Anma gerilimi	24 VDC 4 mA'de, nominal		
Sürekli izin verilebilir gerilim	30 VDC, max.		
Aşırı gerilim	35 VDC için 0.5 saniye		
Lojik 1 sinyal (min.)	15 VDC 2.5 mA'de		
Lojik 0 sinyal (max.)	5 VDC 1 mA'de		
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika		
İzolasyon grupları	1		
Filtre süreleri	us ayarlar: 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0 ms ayarlar: 0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0		
HSC saat giriş hızları (max.) (Lojik 1 seviyesi = 15 - 26 VDC)	100/80 kHz (Ia.0 - Ia.5) 30/20 kHz (Ia.6 - Ib.5)		
Aynı anlı girişlerin sayısı	<ul style="list-style-type: none"> <li>7 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li> <li>14, 55 °C yatayda veya 45 °C dikey</li> </ul>		
Kablo uzunluğu (metre)	500 m zırlı, 300 m zırlısız, 50 m zırlı için HSC girişler		

Çizelge A- 74 Dijital çıkışlar

Teknik veri	CPU 1215C AC/DC/Röle ve CPU 1215C DC/DC/Röle	CPU 1215C DC/DC/DC
Giriş sayısı	10	10
Tip	Röle, kuru kontak	Yarı iletken- MOSFET (Source)
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	--	20 VDC min.
Lojik 0 sinyal, 10 KΩ yük ile	--	0.1 VDC max.
Akım (max.)	2.0 A	0.5 A
Lamba yük	30 W DC / 200 W AC	5 W
ON durum direnci	0.2 Ω max. yeni olduğunda	0.6 Ω max.
Nokta başına kaçak akım	--	10 µA max.
Ani aşırı akım	7 A kontakların kapanmasıyla	8 A için 100 ms max.
Aşırı yük koruması	Hayır	Hayır
İzolasyon (alan tarafı ile lojik)	1500 VAC için 1 dakika (bobin ve kontak) Hiçbiri (bobin ile lojik)	500 VAC için 1 dakika
İzolasyon direnci	100 MΩ min. yeni olduğunda	--
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika	--



Teknik veri	CPU 1215C AC/DC/Röle ve CPU 1215C DC/DC/Röle	CPU 1215C DC/DC/DC
İzolasyon grupları	2	1
İndüktif kenetleme gerilimi	--	L+ eksi 48 VDC, 1 W ısı olarak kayıp
Anahtarlama gecikmesi (Qa.0 - Qa.3)	10 ms max.	1.0 µs max., off - on geçişi 3.0 µs max., on - off geçişi
Anahtarlama gecikmesi (Qa.4 - Qb.1)	10 ms max.	50 µs max., off - on geçişi 200 µs max., on - off geçişi
Maksimum röle anahtarlama frekansı	1 Hz	--
Darbe katarı çıkış hızı	Önerilmez <sup>1</sup>	100 kHz (Qa.0 - Qa.3) <sup>2</sup> , 2 Hz min. 20 kHz (Qa.4 - Qb.1) <sup>2</sup>
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler	--
Anma yükünde ömür boyu temaslar	100,000 açık/kapalı döngüler	--
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	
Aynı andaki çıkış sayısı	<ul style="list-style-type: none"> <li>• 5 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li> <li>• 10, 55 °C yatayda veya 45 °C dikey</li> </ul>	
Kablo uzunluğu (metre)	500 m zırlı, 150 m zırsız	

1 Röle çıkışları olan CPU modellerinde, darbe çıkışlarını kullanmak için bir dijital sinyal kartı (SB) takmanız gerekir.

2 Darbe alıcı ve kablonuza bağlı olarak ilave bir yük direnci (anma akımının en az %10'nunda) darbe sinyal kalitesini ve gürültü bağışıklığını iyileştirebilir.

## A.5.4 Analog girişler ve çıkışlar

Çizelge A- 75 Analog girişler

Teknik veri	Açıklama
Giriş sayısı	2
Tip	Gerilim (tek-sonlu)
Tam-ölçek aralığı	0 - 10 V
Tam-ölçek aralığı (veri word)	0 - 27648
Aşma aralığı	10.001 - 11.759 V
Aşma aralığı (veri word)	27649 - 32511
Taşma aralığı	11.760 - 11.852 V
Taşma aralığı (veri word)	32512 - 32767
Çözünürlük	10 bit
Maksimum dayanma gerilimi	35 VDC
Düzleme	Hiçbiri, Zayıf, Orta veya Güçlü CPU analog girişlerinin basamak tepkileri için Çizelgeye bakınız.
Gürültü bastırma	10, 50 veya 60 Hz
Empedans	≥100 KΩ
İzolasyon (alan tarafı ile lojik)	Hiçbiri

## Teknik özellikler

### A.5 CPU 1215C

Teknik veri	Açıklama
Doğruluk (25 °C / 0 - 55 °C)	Tam-ölçeğe göre % 3.0/ % 3.5
Kablo uzunluğu (metre)	100 m, zırlı bükülü çift

#### A.5.4.1 CPU yerleşik analog girişlerinin basamak tepkisi

Çizelge A- 76 Basamak tepkisi(ms), 0 V -10 V %95'de ölçülen

Düzleme seçimi (örnek ortalama)	Bastırma frekansı (İntegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Hiçbiri (1 döngü): Ortalama alma yok	50 ms	50 ms	100 ms
Zayıf (4 döngü): 4 örnek	60ms	70 ms	200 ms
Orta (16 döngü): 16 örnek	200 ms	240 ms	1150 ms
Güçlü (32 döngü): 32 örnek	400 ms	480 ms	2300 ms
<b>Örnekleme süresi</b>	<b>4.17 ms</b>	<b>5 ms</b>	<b>25 ms</b>

#### A.5.4.2 CPU yerleşik analog portlarının örnekleme süresi

Çizelge A- 77 CPU yerleşik analog girişlerinin örnekleme süresi

Bastırma frekansı (İntegrasyon süresi seçimi)	Örnekleme süresi
60 Hz (16.6 ms)	4.17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

#### A.5.4.3 Gerilime göre analog girişlerin ölçme aralıkları (CPU'lar)

Çizelge A- 78 Gerilime göre analog giriş gösterilişi (CPU'lar)

Sistem		Gerilim Ölçme Aralığı	
Decimal	Hexadecimal	0 - 10 V	
32767	7FFF	11.851 V	Taşma
32512	7F00		
32511	7EFF	11.759 V	Aşma aralığı
27649	6C01		
27648	6C00	10 V	Anma aralığı
20736	5100	7.5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

#### A.5.4.4 Analog çıkış özellikleri

##### Analog çıkışlar

Çizelge A- 79 Analog çıkışlar

Teknik veri	Açıklama
Giriş sayısı	2
Tip	Akım
Tam-ölçek aralığı	0 - 20 mA
Tam-ölçek aralığı (veri word)	0 t- 27648
Aşma aralığı	20.01 - 23.52 mA
Aşma aralığı (veri word)	27649 - 32511
Taşma aralığı	Bakınız dipnot <sup>1</sup>
Taşma aralığı veri word	32512- 32767
Çözünürlük	10 bit
Çıkış sürme empedansı	≤500 Ω max.
İzolasyon (alan tarafı ile lojik)	Hiçbiri
Doğruluk (25 °C / -20 ila 60 °C)	Tam-ölçeğe göre %3.0 / %3.5
Yatışma süresi	2 ms
Kablo uzunluğu (metre)	100 m, zırlı bükülü çift

<sup>1</sup> Bir aşırı akım durumunda, analog çıkışlar konfigürasyon özellikleri ayarına göre davranacaktır. "Reaction to CPU STOP" parametresinde "yedek değeri kullanınız" veya "son değerini tutunuz" den birisini seçiniz.

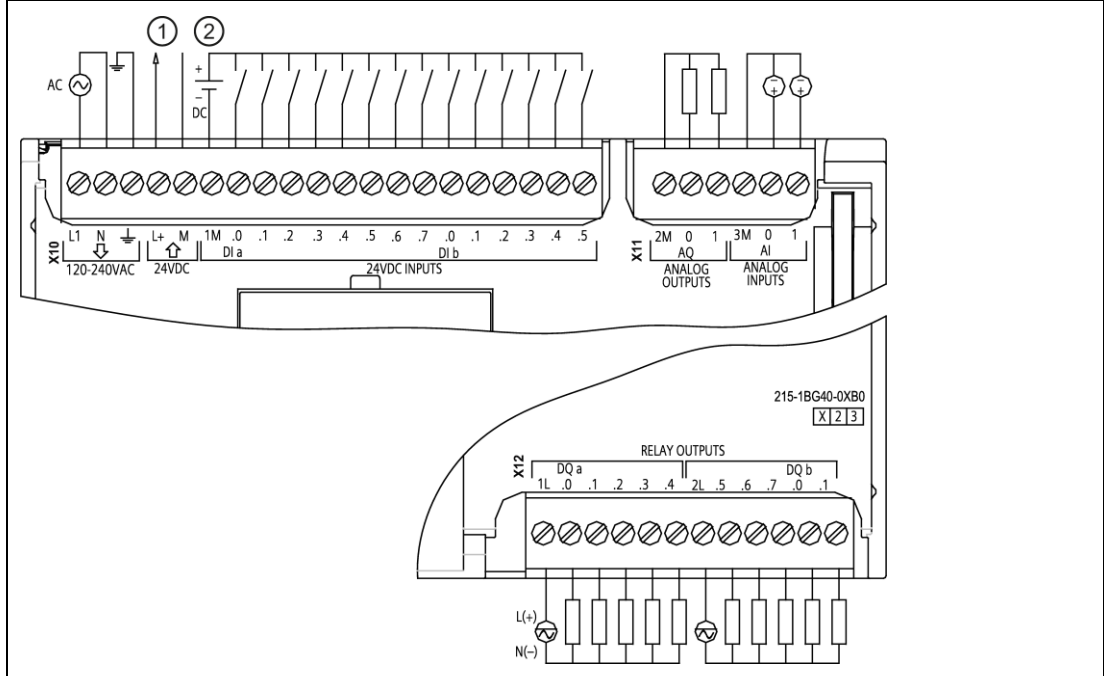
Çizelge A- 80 Akım için analog çıkış iş gösterilişi (CPU 1215C ve CPU 1217C)

Sistem		Akım çıkış aralığı	
Decimal	Hexadecimal	0 mA - 20 mA	
32767	7FFF	Bakınız not 1	Taşma
32512	7F00	Bakınız not 1	
32511	7EFF	23.52 mA	Aşma aralığı
27649	6C01		
27648	6C00	20 mA	Anma aralığı
20736	5100	15 mA	
34	22	12 mV	
0	0	0 mA	
Negatif değerler		Negatif değerler desteklenmez	

<sup>1</sup> Bir aşırı akım durumunda, analog çıkışlar konfigürasyon özellikleri ayarına göre davranacaktır. "Reaction to CPU STOP" parametresinde "yedek değeri kullanınız" veya "son değerini tutunuz" den birisini seçiniz.

## A.5.5 CPU 1215C bağlantı diyagramları

Çizelge A- 81 CPU 1215C AC/DC/Röle (6ES7 215-1BG40-0XB0)



- ① 24 VDC Sensör güç çıkışı  
İlave gürültü bağışıklığı için, bir sensör kaynağı kullanılmasa bile "M" ile şasi topraklamasını bağlayınız.
- ② sinking girişleri için "-" ile "M" bağlantısını yapınız (gösterildiği gibi).  
sourcing girişleri için "+" ile "M" bağlantısını yapınız.

Not 1: X11 konektörler altın olmalıdır. Bakınız sipariş numarası için Ek C, Yedek Parçalar

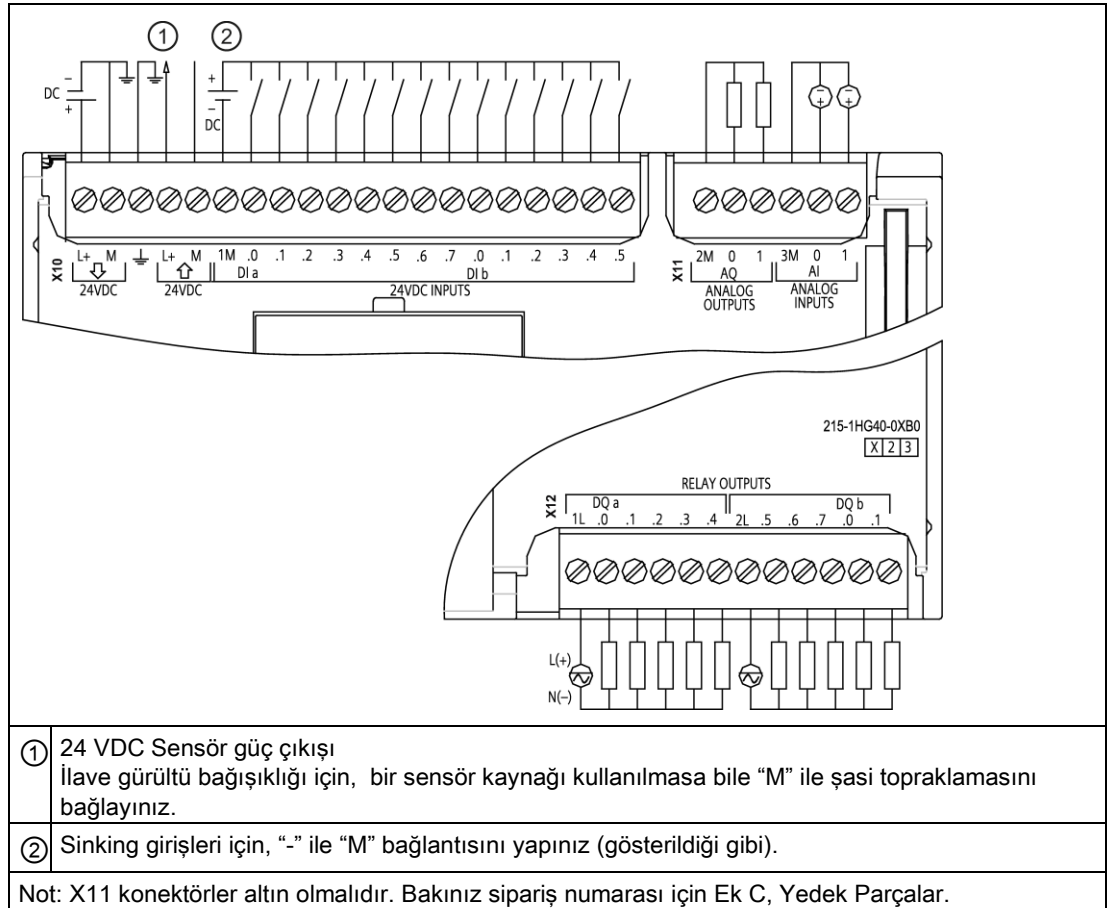
Not 2: L1 veya N (L2) terminali 240 VAC'ye kadar bir gerilim kaynağına bağlanabilir. N terminali L2 olarak dikkate alınabilir ve topraklanması gerekmez. L1 ve N (L2) terminalleri için polarizasyon gerekmez.

Çizelge A- 82 CPU 1215C AC/DC/Röle (6ES7 215-1BG40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L1 /120-240 VAC	2 M	1L
2	N / 120 - 240 VAC	AQ 0	DQ a.0
3	Fonksiyonel toprak	AQ 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	3M	DQ a.2
5	M / 24 VDC Sensör çıkışı	AI 0	DQ a.3
6	1M	AI 1	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7

Pin	X10	X11 (altın)	X12
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

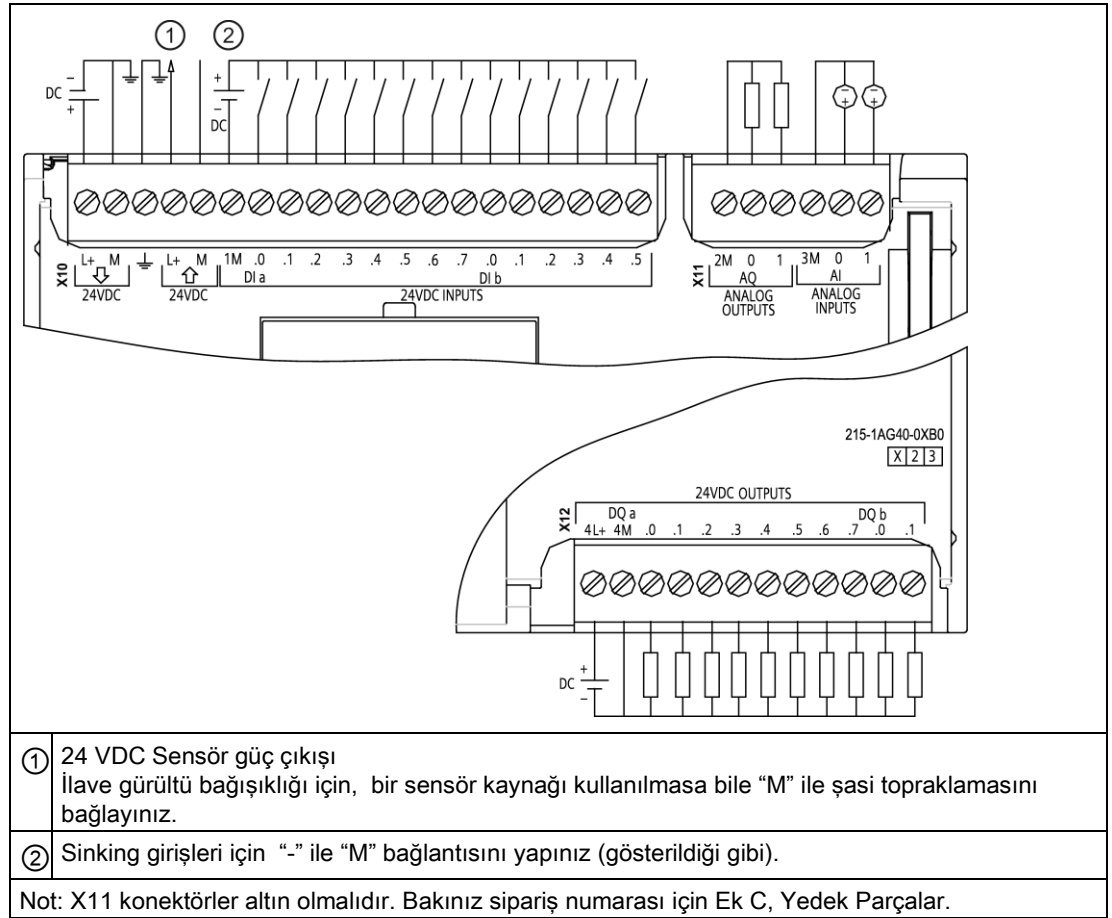
Çizelge A- 83 CPU 1215C DC/DC/Röle (6ES7 215-1HG40-0XB0)



Çizelge A- 84 CPU 1215C DC/DC/Röle (6ES7 215-1HG40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L+ / 24 VDC	2 M	1L
2	M / 24 VDC	AQ 0	DQ a.0
3	Fonksiyonel toprak	AQ 1	DQ a.1
4	L+ / 24 VDC Sensör çıkışı	3M	DQ a.2
5	M / 24 VDC Sensör çıkışı	AI 0	DQ a.3
6	1M	AI 1	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

Çizelge A- 85 CPU 1215C DC/DC/DC (6ES7 215-1AG40-0XB0)



Çizelge A- 86 CPU 1215C DC/DC/DC (6ES7 215-1AG40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11 (altın)	X12
1	L1 / 24 VDC	2 M	4L+
2	M / 24 VDC	AQ 0	4M
3	Fonksiyonel toprak	AQ 1	DQ a.0
4	L+ / 24 VDC Sensör çıkışı	3M	DQ a.1
5	M / 24 VDC Sensör çıkışı	AI 0	DQ a.2
6	1M	AI 1	DQ a.3
7	DI a.0	--	DQ a.4
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--

## Teknik özellikler

### A.6 CPU 1217C

Pin	X10	X11 (altın)	X12
15	DI b.0	--	--
16	DI b.1	--	--
17	DI b.2	--	--
18	DI b.3	--	--
19	DI b.4	--	--
20	DI b.5	--	--

#### Not

Kullanılmayan analog girişler topraklanmalıdır.

## A.6 CPU 1217C

### A.6.1 Genel açıklamalar ve özellikler

Çizelge A- 87 Genel

Teknik veri	CPU 1217C DC/DC/DC
Sipariş numarası	6ES7 217-1AG40-0XB0
Boyutlar W x H x D (mm)	150 x 100 x 75
Nakliye ağırlığı	530 gram
Güç kaybı	12 W
Akım kullanılabilir(SM ve CM bus)	1600 mA max. (5 VDC)
Akım kullanılabilir(24 VDC)	400 mA max. (sensör gücü)
Dijital girişin çektiği akım(24 VDC)	4 mA/giriş kullanıldı

Çizelge A- 88 CPU özellikleri

Teknik veri	Açıklama	
Kullanıcı belleği	Çalışma	125 Kbayt
	Yükleme	4 Mbayt, dahili, SD kartı boyutuna kadar genişletilebilir
	Kalıcı	10 Kbayt
Yerleşik dijital I/O	14 giriş/ 10 Çıkış	
Yerleşik analog I/O	2 giriş/ 2 Çıkış	
Proses görüntü boyutu	1024 girişlerin baytları (I) / 1024 çıkışların baytları (Q)	
Bit belleği(M)	8192 bayt	



Teknik veri	Açıklama
Geçici (lokal) bellek	<ul style="list-style-type: none"> <li>• 16 Kbayt, başlangıç ve program döngüsü için (ilişkili FB'ler ve FC'ler dahil)</li> <li>• 6 Kbayt, diğer interrupt öncelik seviyelerinin her biri için (FB'ler ve FC'ler dahil)</li> </ul>
Sinyal modülleri genişlemesi	8 SM max.
SB, CB, BB genişlemesi	1 max.
Haberleşme modülü genişlemesi	3 CM max.
Yüksek-hızlı sayıcılar	6'ya kadar, yerleşik veya SB girişleri kullanım için konfigüre edildiğinde (bakınız CPU 1217C Dijital giriş (DI) H/W konfigürasyon çizelgesi) (Sayfa 882) <ul style="list-style-type: none"> <li>• 1 MHz (Ib.2 - Ib.5)</li> <li>• 100/180 kHz (Ia.0 - Ia.5)</li> <li>• 30/120 kHz (Ia.6 - Ib.1)</li> </ul>
Darbe çıkışları	4'e kadar, yerleşik veya SB çıkışları kullanım için konfigüre edildiğinde (bakınız CPU 1217C Dijital çıkış (DQ) H/W konfigürasyon çizelgesi) (Sayfa 882) <ul style="list-style-type: none"> <li>• 1 MHz (Qa.0 - Qa.3)</li> <li>• 100 kHz (Qa.4 - Qb.1)</li> </ul>
Darbe yakalama girişleri	14
Süre gecikme interrupt'ları	Toplam 4, 1 ms çözünürlük ile
Döngüsel interrupt'lar	Toplam 4, 1 ms çözünürlük ile
Kenar interrupt'ları	12 yükselme ve 12 düşme (16 ve opsiyonel sinyal kartlı16)
Bellek kartı	SIMATIC Bellek kartı (opsiyonel)
Gerçek zaman saat doğruluğu	+/- 60 saniye/ay
Gerçek zaman saat tutma süresi	20 gün tipik/12 gün min. 40 °C'ta (bakımsız Süper Kapasitör)

<sup>1</sup> HSC dördün çalışma moduna konfigüre edildiğinde daha düşük hız uygulanabilir.

#### Çizelge A- 89 Performans

Komut tipi	Yürütme hızı
Boolean	0.08 µs/komut
Move Word	1.7 µs/komut
Real math	2.3 µs/komut

## A.6.2 CPU 1217C tarafından desteklenen zamanlayıcılar, sayıcılar ve kod blokları

Çizelge A- 90 CPU 1217C tarafından desteklenen zamanlayıcılar, sayıcılar ve kod blokları

Öge	Açıklama	
Bloklar	Tip	OB, FB, FC, DB
	Boyut	64 Kbayt
	Miktar	Toplam 1024 bloğa kadar (OB'ler + FB'ler + FC'ler + DB'ler)
	FB'ler, FC'ler ve DB'ler için adres aralığı	FB ve FC: 1 - 65535 ( FB 1 - FB 65535 gibi) DB: 1 - 59999

## Teknik özellikler

## A.6 CPU 1217C

Öge	Açıklama
	Gömme derinliği 16, program döngüsü veya başlangıç OB'sinden 6, herhangi bir interrupt olayı OB'sinden
	İzleme 2 kod bloğunun durumu aynı anda izlenebilir
OB'lar	Program döngüsü Çoklu
	Başlangıç Çoklu
	Süre-geciktirme interrupt'ları 4 (1 olay başına)
	Döngüsel interrupt'lar 4 (1 olay başına)
	Donanım interrupt'ları 50 (1 olay başına)
	Süre hatası interrupt'ları 1
	Diyagnostik hatası interrupt'ları 1
	Çek veya tak modülleri 1
	Şasi veya istasyon arızası 1
	Günün saati Çoklu
	Durum 1
	Güncelleme 1
	Profile 1
Zamanlayıcılar	Tip IEC
	Miktar Bellek boyutu ile sınırlıdır
	Depolama DB yapısı, zamanlayıcı başına 16 bayt
Sayıcılar	Tip IEC
	Miktar Bellek boyutu ile sınırlıdır
	Depolama DB yapısı, sayma tipine göre boyut <ul style="list-style-type: none"> <li>• SInt, USInt: 3 bayt</li> <li>• Int, UInt: 6 bayt</li> <li>• DInt, UDInt: 12 bayt</li> </ul>

## Çizelge A- 91 Haberleşme

Teknik veri	Açıklama
Portların numarası	2
Tip	Ethernet
HMI cihazı	6 max.
Programlama cihazı (PG)	1
Bağlantılar	<ul style="list-style-type: none"> <li>• 8, açık kullanıcı haberleşmesi için (aktif veya pasif): TSEND_C, TRCV_C, TCON, TDISCON, TSEND ve TRCV</li> <li>• 3 için sunucu GET/PUT (CPU-ile-CPU) S7 haberleşme</li> <li>• 8 için istemci GET/PUT (CPU-ile-CPU) S7 haberleşme</li> </ul>
Veri hızları	10/100 Mb/s
İzolasyon (PLC lojiğe harici sinyal)	Transformator izoleli, 1500 VAC, sadece kısa süreli olay güvenliği için
Kablo tipi	CAT5e zırhlı

Çizelge A- 92 Güç kaynağı

Teknik veri		CPU 1217C DC/DC/DC	
Gerilim aralığı		20.4 VDC - 28.8 VDC	
Hat frekansı		--	
Giriş akımı (max. yük)	CPU sadece	600 mA	24 VDC'de
	CPU, bütün genişleme donatıları ile	1600 mA	24 VDC'de
Baskın akım (max.)		12 A	28.8 VDC'de
İzolasyon (giriş gücü ile lojik arasında)		İzole edilmemiş	
Tutma süresi (gücün kaybindan)		10 ms	24 VDC'de
Dahili sigorta, kullanıcı değiştiremez		3 A, 250 V, geç atmalı	

Çizelge A- 93 Sensör gücü

Teknik veri		CPU 1217C DC/DC/DC	
Gerilim aralığı		L+ eksi 4 VDC min.	
Çıkış akımı değeri (max.)		400 mA (kısa-devre korumalı)	
Maksimum ripıl gürültüsü(<10 MHz)		Giriş hattı ile aynı	
İzolasyon (CPU lojigi ile sensör gücü arasında)		İzole edilmemiş	

### A.6.3 Dijital girişler ve çıkışlar

Çizelge A- 94 Dijital girişler

Teknik veri		CPU 1217C DC/DC/DC	
Giriş sayısı		14: total:	
		10: Sink/source (IEC Tip 1 sink)	
		4: 1.5 V diferansiyel	
<b>Tip: Sink/source (IEC Tip 1 sink)</b>		Ia.0 - Ia.7, Ib.0 - Ib.1	
Anma gerilimi		24 VDC 4 mA'de, nominal	
Sürekli izin verilebilir gerilim		30 VDC, max.	
Aşırı gerilim		35 VDC için 0.5 saniye	
Lojik 1 sinyal (min.)		15 VDC	2.5 mA'de
Lojik 0 sinyal (max.)		5 VDC	1 mA'de
İzolasyon (alan tarafı ile lojik)		500 VAC için 1 dakika (fonksiyonel izolasyon)	
İzolasyon grupları		1	
Filtre süreleri		us ayarlar: 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0	
		ms ayarlar: 0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0	

## Teknik özellikler

### A.6 CPU 1217C

Teknik veri	CPU 1217C DC/DC/DC
HSC saat giriş hızları (max.) (Lojik 1 seviye= 15 - 26 VDC)	100/80 kHz (Ia.0 - Ia.5) 30/20 kHz (Ia.6 - Ib.1)
<b>Tip: Diferansiyel giriş</b>	Ib.2 - Ib.5 (.2+ .2- ila .5+ .5-)
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 saniye, 3 VRMS sürekli (RS485 karakteristikler)
Sonlandırma ve polarlama	2M'e göre 390 Ω, Ib'-' üzerinde, +5 V'a göre 390 Ω, Ib'-' üzerinde, (OFF polarlama, T/B açık-devre olduğunda) 220 Ω, Ib'+-' ve Ib'-' arasında
Alıcı giriş empedansı	100 Ω, sonlandırma ve polarlama dahil
Diferansiyel alıcı eşik/duyarlılık	+/- 0.2 V min., 60 mV tipik histeresis
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika (fonksiyonel izolasyon)
İzolasyon grupları	1
Filtre süreleri	us ayarlar: 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0 ms ayarlar: 0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0
HSC saat giriş hızları (max.)	Tek faz: 1 MHz (Ib.2 - Ib.5) Dördün faz: 1 MHz (Ib.2 - Ib.5)
Diferansiyel giriş Kanal-kanal çarpık	40 ns max.
<b>Genel özellikler (bütün dijital girişler)</b>	
Aynı anlı girişlerin sayısı	5 Sink/source giriş (bitişik noktalar değil) ve 4 diferansiyel giriş, 60 °C yatayda veya 50 °C dikey 14, 55 °C yatayda veya 45 °C dikey
Kablo uzunluğu (metre)	500 m zırlı, 300 m zırlısız 50 m zırlı için HSC girişler (sink/source) 50 m zırlı, bütün diferansiyel girişler için bükümlü çiftler

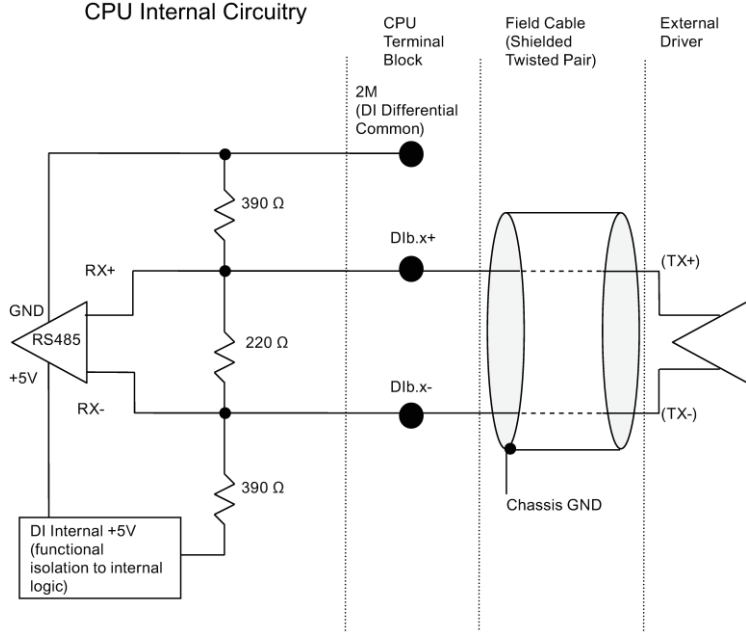
### Çizelge A- 95 Dijital çıkışlar

Teknik veri	CPU 1217C DC/DC/DC
Giriş sayısı	10 total 6: Yarı iletken- MOSFET (Source) 4: 1.5 V diferansiyel
<b>Tip: Yarı iletken- MOSFET (Source çıkışı)</b>	Qa.4 - Qb.1
Gerilim aralığı	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	20 VDC min.
Lojik 0 sinyal, 10 KΩ yük ile	0.1 VDC max.
Akım (max.)	0.5 A
Lamba yük	5 W
ON durum direnci	0.6 Ω max.
Nokta başına kaçak akım	10 µA max.
Ani aşırı akım	8 A için 100 ms max.
Aşırı yük koruması	Hayır

Teknik veri	CPU 1217C DC/DC/DC
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika (fonksiyonel izolasyon)
İzolasyon grupları	1
İndüktif kenetleme gerilimi	L+ eksi 48 VDC, 1 W ısı olarak kayıp
Anahtarlama gecikmesi (Qa.0 - Qa.3)	1.0 µs max., off - on geçişi 3.0 µs max., on - off geçişi
Anahtarlama gecikmesi (Qa.4 - Qb.1)	50 µs max., off - on geçişi 200 µs max., on - off geçişi
Maksimum röle anahtarlama frekansı	--
Darbe katırı çıkış hızı	100 KHz max. (Qa.4 - Qb.1) <sup>1</sup> , 2 Hz min.
<b>Tip: 1.5 V Diferansiyel çıkış</b>	Qa.0 - Qa.3 (.0+ 0- ila .3+ .3-)
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 saniye, 3 VRMS sürekli (RS485 karakteristikler)
Aktarıcı diferansiyel çıkış gerilimi	2 V min. RL = 100 Ω'da, 1.5 V min. RL = 54 Ω'da
Sonlandırma	100 Ω, Qa'+ ve Qa-' arasında
Sürücü çıkış empedansı	100 Ω sonlandırma dahil
İzolasyon	500 VAC, 1 dakika (fonksiyonel izolasyon)
İzolasyon grupları	1
Anahtarlama gecikmesi (DQa.0 - DQa.3)	100 ns max.
Diferansiyel çıkış kanal-kanal çapraz	40 ns max.
Darbe katırı çıkış hızı	1 MHz (Qa.0 - Qa.3), 2 Hz min.
<b>Genel özellikler (tüm dijital çıkışlar)</b>	
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	3, Yarı iletken- MOSFET (Source) çıkışları (bitişik noktalar değil) ve 4 diferansiyel çıkış 60 °C yatayda veya 50 °C dikeyde 10, 55 °C yatayda veya 45 °C dikeyde
Kablo uzunluğu (metre)	500 m zırlı, 150 m zırlısız

<sup>1</sup> Darbe alıcı ve kablonuza bağlı olarak ilave bir yük direnci (anma akımının en az %10'nunda) darbe sinyal kalitesini ve gürültü bağışıklığını iyileştirebilir.

### 1.5 V diferansiyel giriş ayrıntısı



**Not:**

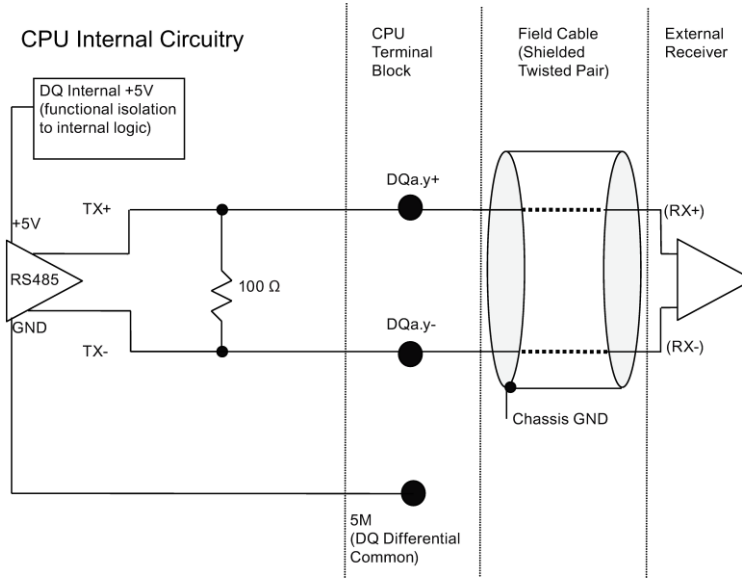
- Terminal bloklarının vidaları açık devre olduklarında, her bir diferansiyel DI, "OFF" olarak polarma yapılır.
- Sonlandırma ve polarlama = 100  $\Omega$ .

Çizelge A- 96 CPU 1217C Dijital giriş (DI) H/W konfigürasyon çizelgesi

Giriş	Tip ve değer
Dla.0	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 100 kHz max.
Dla.1	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 100 kHz max.
Dla.2	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 100 kHz max.
Dla.3	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 100 kHz max.
Dla.4	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 100 kHz max.
Dla.5	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: :100 kHz max.
Dla.6	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 30 kHz max.
Dla.7	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 30 kHz max.

Giriş	Tip ve değer
Dlb.0	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 30 kHz max.
Dlb.1	Tip: 24 V, source-sink Tip 1 giriş Yüksek-hızlı sayıcı giriş hızı: 30 kHz max.
Dlb.2+ .2-	Tip: 1.5 V Differential input Yüksek-hızlı sayıcı giriş hızı: 1 MHz max.
Dlb.3+ .3-	Tip: 1.5 V Differential input Yüksek-hızlı sayıcı giriş hızı: 1 MHz max.
Dlb.4+ .4-	Tip: 1.5 V Differential input Yüksek-hızlı sayıcı giriş hızı: 1 MHz max.
Dlb.5+ .5-	Tip: 1.5 V Differential input Yüksek-hızlı sayıcı giriş hızı: 1 MHz max.

### 1.5 V diferansiyel çıkış ayrıntısı



## Teknik özellikler

### A.6 CPU 1217C

Çizelgesi A- 97 CPU 1217C dijital çıkış (DQ) H/W konfigürasyon çizelgesi

Çıkış	Tip ve değer
DQa.0+ .0-	Tip: 1.5 V Diferansiyel çıkış Darbe katarı çıkış hızı: 1 MHz max., 2 Hz min.
DQa.1+ .1-	Tip: 1.5 V Diferansiyel çıkış Darbe katarı çıkış hızı: 1 MHz max., 2 Hz min.
DQa.2+ .2-	Tip: 1.5 V Diferansiyel çıkış Darbe katarı çıkış hızı: 1 MHz max., 2 Hz min.
DQa.3+ .3-	Tip: 1.5 V Diferansiyel çıkış Darbe katarı çıkış hızı: 1 MHz max., 2 Hz min.
DQa.4	Tip: 24 V Sourcing çıkışı Darbe katarı çıkış hızı: 100 kHz max., 2 Hz min.
DQa.5	Tip: 24 V Sourcing çıkışı Darbe katarı çıkış hızı: 100 kHz max., 2 Hz min.
DQa.6	Tip: 24 V Sourcing çıkışı Darbe katarı çıkış hızı: 100 kHz max., 2 Hz min.
DQa.7	Tip: 24 V Sourcing çıkışı Darbe katarı çıkış hızı: 100 kHz max., 2 Hz min.
DQb.0	Tip: 24 V Sourcing çıkışı Darbe katarı çıkış hızı: 100 kHz max., 2 Hz min.
DQb.1	Tip: 24 V Sourcing çıkışı Darbe katarı çıkış hızı: 100 kHz max., 2 Hz min.

## A.6.4 Analog girişler ve çıkışlar

### A.6.4.1 Analog giriş özellikleri

Çizelge A- 98 Analog girişler

Teknik veri	Açıklama
Giriş sayısı	2
Tip	Gerilim (tek-sonlu)
Tam-ölçek aralığı	0 - 10 V
Tam-ölçek aralığı (veri word)	0 - 27648
Aşma aralığı	10.001 - 11.759 V
Aşma aralığı (veri word)	27649 - 32511
Taşma aralığı	11.760 - 11.852 V
Taşma aralığı (veri word)	32512 - 32767
Çözünürlük	10 bit
Maksimum dayanma gerilimi	35 VDC
Düzleme	Hiçbiri, Zayıf, Orta veya Güçlü CPU analog girişleri için basamak tepkisi (ms) Çizelgesine bakınız.
Gürültü bastırma	10, 50 veya 60 Hz



Teknik veri	Açıklama
Empedans	≥100 KΩ
İzolasyon (alan tarafı ile lojik)	Hiçbiri
Doğruluk (25 °C / 0 - 55 °C)	Tam-ölçeğe göre %3.0 / %3.5
Kablo uzunluğu (metre)	100 m, zırlı bükülü çift

#### A.6.4.2 CPU yerleşik analog girişlerinin basamak tepkisi

Çizelge A- 99 Basamak tepkisi(ms), 0 V -10 V, %95'e göre ölçülen

Düzleme seçimi (örnek ortalama)	Bastırma frekansı (İntegrasyon süresi)		
	60 Hz	50 Hz	10 Hz
Hiçbiri (1 döngü): Ortalama alma yok	50 ms	50 ms	100 ms
Zayıf (4 döngü): 4 örnek	60ms	70 ms	200 ms
Orta (16 döngü): 16 örnek	200 ms	240 ms	1150 ms
Güçlü (32 döngü): 32 örnek	400 ms	480 ms	2300 ms
<b>Örnekleme süresi</b>	<b>4.17 ms</b>	<b>5 ms</b>	<b>25 ms</b>

#### A.6.4.3 CPU'nun gömülü analog girişleri için örnekleme süresi

Çizelge A- 100 CPU'nun gömülü analog girişleri için örnekleme süresi

Bastırma frekansı (İntegrasyon süresi seçimi)	Örnekleme süresi
60 Hz (16.6 ms)	4.17 ms
50 Hz (20 ms)	5 ms
10 Hz (100 ms)	25 ms

#### A.6.4.4 Analog girişlerin gerilim ölçme aralıkları (CPU'lar)

Çizelge A- 101 Analog giriş gerilim gösterimi (CPU'lar)

Sistem		Gerilim Ölçme Aralığı	
Decimal	Hexadecimal	0 - 10 V	
32767	7FFF	11.851 V	Taşma
32512	7F00		
32511	7EFF	11.759 V	Aşma aralığı
27649	6C01		
27648	6C00	10 V	Anma aralığı
20736	5100	7.5 V	
34	22	12 mV	
0	0	0 V	
Negatif değerler		Negatif değerler desteklenmez	

#### A.6.4.5 Analog çıkış özellikleri

##### Analog çıkışlar

Çizelge A- 102 Analog çıkışlar

Teknik veri	Açıklama
Giriş sayısı	2
Tip	Akım
Tam-ölçek aralığı	0 - 20 mA
Tam-ölçek aralığı (veri word)	0 - 27648
Aşma aralığı	20.01 - 23.52 mA
Aşma aralığı (veri word)	27649 - 32511
Taşma aralığı	Bakınız dipnot <sup>1</sup>
Taşma aralığı data word	32512 - 32767
Çözünürlük	10 bit
Çıkış sürme empedansı	≤500 Ω max.
İzolasyon (alan tarafı ile lojik)	Hiçbiri
Doğruluk (25 °C / -20 ila 60 °C)	Tam-ölçeğe göre 3.0% / 3.5%
Yatışma süresi	2 ms
Kablo uzunluğu (metre)	100 m, zırlı bükülü çift

1 Bir aşırı akım durumunda, analog girişler cihaz konfigürasyon özellikleri ayarına göre davranacaktır. "Reaction to CPU STOP" parametresinde, ya "Use substitute value" ya da "Keep last value" seçiniz.

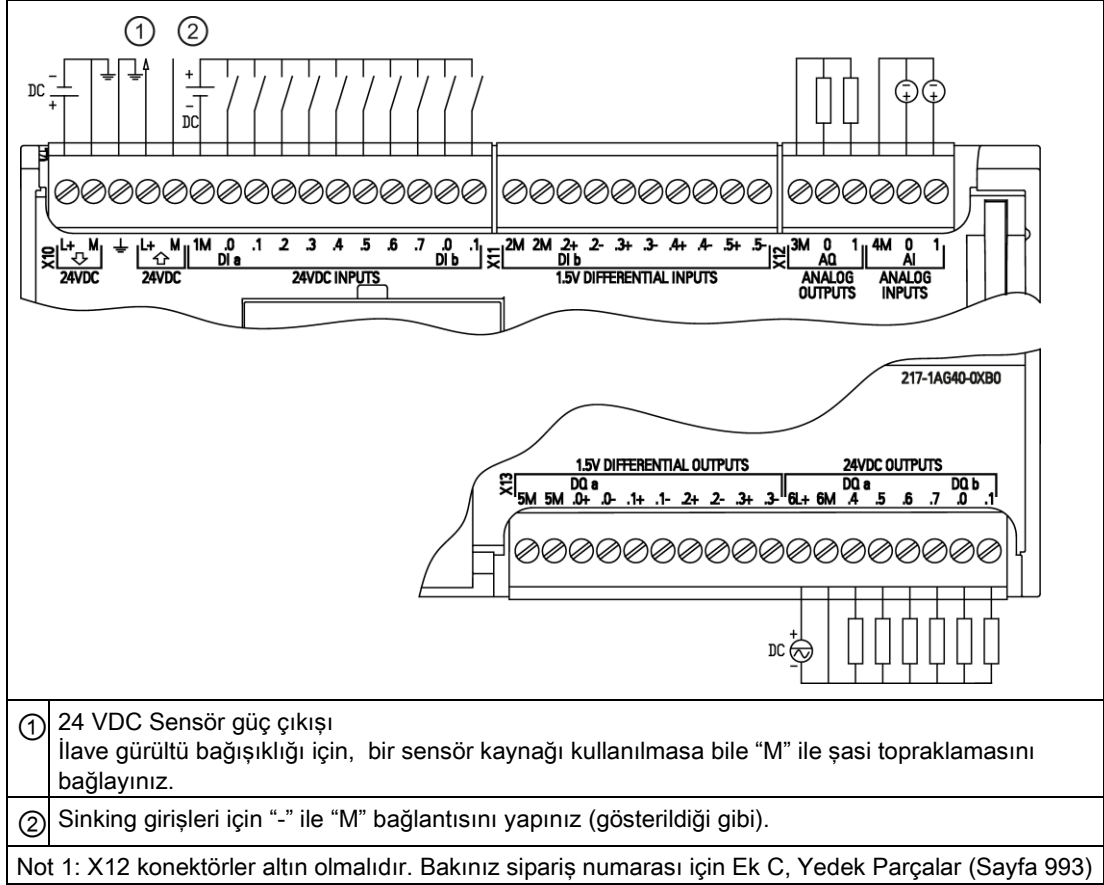
Çizelge A- 103 Geçerli (CPU 1215C ve CPU 1217C) için analog çıkış gösterimi

Sistem		Akım çıkış aralığı	
Decimal	Hexadecimal	0 mA - 20 mA	
32767	7FFF	Bakınız not 1	Taşma
32512	7F00	Bakınız not 1	
32511	7EFF	23.52 mA	Aşma aralığı
27649	6C01		
27648	6C00	20 mA	Anma aralığı
20736	5100	15 mA	
34	22	12 mV	
0	0	0 mA	
Negatif değerler		Negatif değerler desteklenmez	

<sup>1</sup> Bir aşırı akım durumunda, analog çıkışlar cihaz konfigürasyon özellikleri ayarına göre davranacaktır. "Reaction to CPU STOP" parametresinde, ya "Use substitute value" ya da "Keep last value" seçiniz.

## A.6.5 CPU 1217C bağlantı diyagramları

Çizelge A- 104 CPU 1217C DC/DC/DC (6ES7 217-1AG40-0XB0)



Çizelge A- 105 CPU 1217C DC/DC/DC (6ES7 217-1AG40-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12 (altın)	X13
1	L+ / 24 VDC	2M	3M	5M
2	M / 24 VDC	2M	AQ 0	5M
3	Fonksiyonel toprak	DI b.2+	AQ 1	DQ a.0+
4	L+ / 24 VDC Sensör çıkışı	DI b.2-	4M	DQ a.0-
5	M / 24 VDC Sensör çıkışı	DI b.3+	AI 0	DQ a.1+
6	1M	DI b.3-	AI 1	DQ a.1-
7	DI a.0	DI b.4+	--	DQ a.2+
8	DI a.1	DI b.4-	--	DQ a.2-
9	DI a.2	DI b.5+	--	DQ a.3+
10	DI a.3	DI b.5-	--	DQ a.3-
11	DI a.4	--	--	6L+
12	DI a.5	--	--	6M

Pin	X10	X11	X12 (altın)	X13
13	DI a.6	--	--	DQ a.4
14	DI a.7	--	--	DQ a.5
15	DI b.0	--	--	DQ a.6
16	DI b.1	--	--	DQ a.7
17	--	--	--	DQ b.0
18	--	--	--	DQ b.1

**Not**

Kullanılmayan analog girişler topraklanmalıdır.

## A.7 Dijital sinyal modülleri (SM'ler)

### A.7.1 SM 1221 dijital giriş özellikleri

Çizelge A- 106 Genel özellikler

Model	SM 1221 DI 8 x 24 VDC	SM 1221 DI 16 x 24 VDC
Sipariş numarası	6ES7 221-1BF32-0XB0	6ES7 221-1BH32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	45 x 100 x 75
Ağırlık	170 gram	210 gram
Güç kaybı	1.5 W	2.5 W
Çektiği akım(SM Bus)	105 mA	130 mA
Çektiği akım(24 VDC)	4 mA / giriş kullanıldı	4 mA / giriş kullanıldı

Çizelge A- 107 Dijital girişler

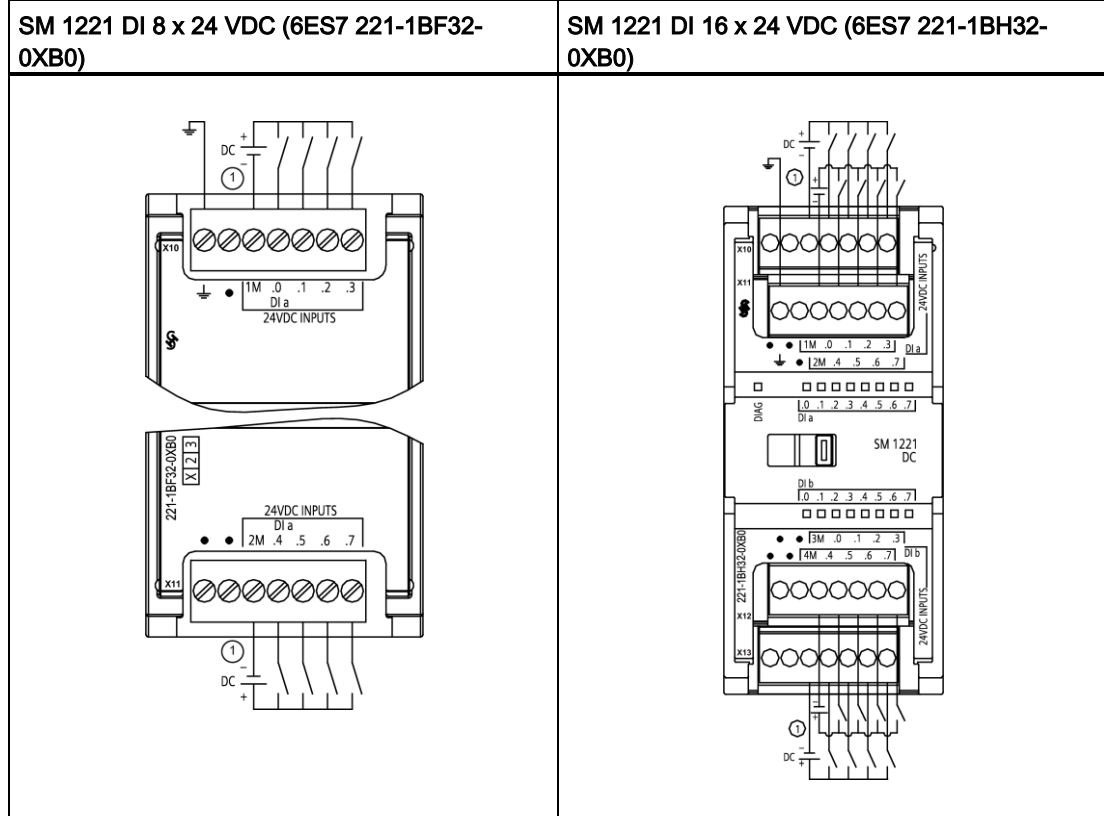
Model	SM 1221 DI 8 x 24 VDC	SM 1221 DI 16 x 24 VDC
Giriş sayısı	8	16
Tip	Sink/Source (IEC Tip 1 sink)	Sink/Source (IEC Tip 1 sink)
Anma gerilimi	24 VDC 4 mA'de, nominal	24 VDC 4 mA'de, nominal
Sürekli izin verilebilir gerilim	30 VDC, max.	30 VDC, max.
Aşırı gerilim	35 VDC için 0.5 saniye	35 VDC için 0.5 saniye
Lojik 1 sinyal (min.)	15 VDC 2.5 mA'de	15 VDC 2.5 mA'de
Lojik 0 sinyal (max.)	5 VDC 1 mA'de	5 VDC 1 mA'de
İzolasyon (alan tarafı-lojik)	500 VAC için 1 dakika	500 VAC için 1 dakika
İzolasyon grupları	2	4

## Teknik özellikler

### A.7 Dijital sinyal modülleri (SM'ler)

Model	SM 1221 DI 8 x 24 VDC	SM 1221 DI 16 x 24 VDC
Filtre süreleri	0.2, 0.4, 0.8, 1.6, 3.2, 6.4, ve 12.8 ms (grup 4'de seçilebilir)	0.2, 0.4, 0.8, 1.6, 3.2, 6.4, ve 12.8 ms (grup 4'de seçilebilir)
Aynı anlı girişlerin sayısı	8	16
Kablo uzunluğu (metre)	500 m zırlı, 300 zırsız	500 m zırlı, 300 zırsız

Çizelge A- 108 Dijital giriş SM'ler için bağlantı diyagramları



① sinking girişleri için "-" ile "M" bağlantısını yapınız (gösterilmiştir). sourcing girişleri için "+" ile "M" bağlantısını yapınız.

Çizelge A- 109 SM 1221 DI 8 x 24 VDC (6ES7 221-1BF32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11
1	GND	Bağlantı yok
2	Bağlantı yok	Bağlantı yok
3	1M	2M
4	DI a.0	DI a.4
5	DI a.1	DI a.5
6	DI a.2	DI a.6
7	DI a.3	DI a.7

Çizelge A- 110 SM 1221 DI 16 x 24 VDC (6ES7 221-1BH32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12	X13
1	GND	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	3 M	4 M
4	DI a.0	DI a.4	DI b.0	DI b.4
5	DI a.1	DI a.5	DI b.1	DI b.5
6	DI a.2	DI a.6	DI b.2	DI b.6
7	DI a.3	DI a.7	DI b.3	DI b.7

## A.7.2 SM 1222 8-nokta dijital çıkış özellikleri

Çizelge A- 111 Genel özellikler

Model	SM 1222 DQ 8 x Röle	SM 1222 DQ 8 RLY Değişirme	SM 1222 DQ 8 x 24 VDC
Sipariş numarası	6ES7 222-1HF32-0XB0	6ES7 222-1XF32-0XB0	6ES7 222-1BF32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	70 x 100 x 75	45 x 100 x 75
Ağırlık	190 gram	310 gram	180 gram
Güç kaybı	4.5 W	5 W	1.5 W
Çektiği akım(SM Bus)	120 mA	140 mA	120 mA
Çektiği akım(24 VDC)	11 mA / Röle bobini kullanıldı	16.7 mA/Röle bobini kullanıldı	50 mA

Çizelge A- 112 Dijital çıkışlar

Model	SM 1222 DQ 8 x Röle	SM 1222 DQ8 RLY Değişirme	SM 1222 DQ 8 x 24 VDC
Giriş sayısı	8	8	8
Tip	Röle, kuru kontak	Röle değişirme kontağı	Yarı iletken- MOSFET (sourcing)
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC	5 - 30 VDC veya 5 - 250 VAC	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	--	--	20 VDC min.
Lojik 0 sinyal, 10K $\Omega$ yük ile	--	--	0.1 VDC max
Akım (max.)	2.0 A	2.0 A	0.5 A
Lamba yük	30 W DC/200 W AC	30 W DC/200 W AC	5 W
ON durumu temas direnci	0.2 $\Omega$ max. yeni olduğunda	0.2 $\Omega$ max. yeni olduğunda	0.6 $\Omega$ max.
Nokta başına kaçak akım	--	--	10 $\mu$ A max.
Ani aşırı akım	7 A kontakların kapanmasıyla	7 A kontakların kapanmasıyla	8 A için 100 ms max.

## Teknik özellikler

### A.7 Dijital sinyal modülleri (SM'ler)

Model	SM 1222 DQ 8 x Röle	SM 1222 DQ8 RLY Değiştirme	SM 1222 DQ 8 x 24 VDC
Aşırı yük koruması	Hayır	Hayır	Hayır
İzolasyon (alan tarafı ile lojik)	1500 VAC için 1 dakika (bobin ve kontak) Hiçbiri (bobin ile lojik)	1500 VAC için 1 dakika (bobin ve kontak)	500 VAC için 1 dakika
İzolasyon direnci	100 MΩ min. yeni olduğunda	100 MΩ min. yeni olduğunda	--
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika	750 VAC için 1 dakika	--
İzolasyon grupları	2	8	1
Ortak nokta başına akım (max.)	10 A	2 A	4 A
İndüktif kenetleme gerilimi	--	--	L+ eksi 48 V, 1 W ısı olarak kayıp
Anahtarlama gecikmesi	10 ms max.	10 ms max	50 μs max. off - on geçişi 200 μs max. on - off geçişi
Maksimum röle anahtarlama frekansı	1 Hz	1 Hz	--
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler	10,000,000 açık/kapalı döngüler	--
Kontakların ömrü (N.A. kontak)	100,000 açık/kapalı döngüler	100,000 açık/kapalı döngüler	--
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	Son değer veya değişiklik değeri (varsayılan değer 0)	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	8	<ul style="list-style-type: none"> <li>4 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li> <li>8, 55 °C yatayda veya 45 °C dikey</li> </ul>	8
Kablo uzunluğu (metre)	500 m zırlı, 150 m zırlısız	500 m zırlı, 150 m zırlısız	500 m zırlı, 150 m zırlısız

### A.7.3 SM 1222 16-nokta dijital çıkış özellikleri

Çizelge A- 113 Genel özellikler

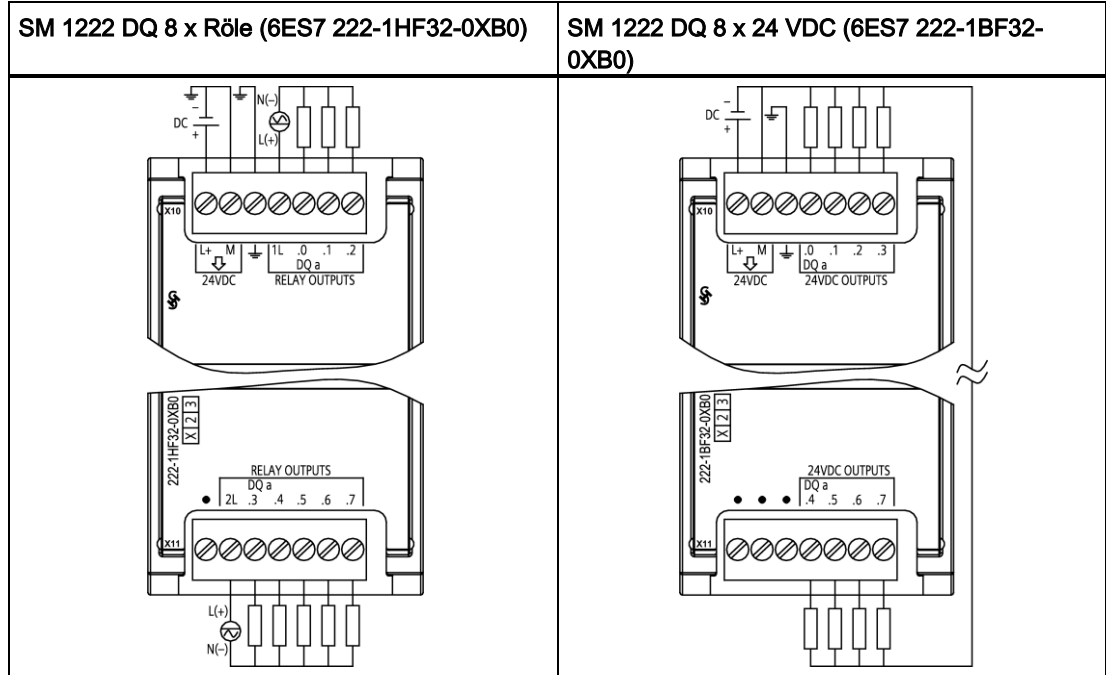
Model	SM 1222 DQ 16 x Röle	SM 1222 DQ 16 x 24 VDC
Sipariş numarası	6ES7 222-1HH32-0XB0	6ES7 222-1BH32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	45 x 100 x 75
Ağırlık	260 gram	220 gram
Güç kaybı	8.5 W	2.5 W
Çektiği akım(SM Bus)	135 mA	140 mA
Çektiği akım(24 VDC)	11 mA / Röle bobini kullanıldı	100 mA



Çizelge A- 114 Dijital çıkışlar

Model	SM1222 DQ 16 x Röle	SM1222 DQ 16 x 24 VDC
Giriş sayısı	16	16
Tip	Röle, kuru kontak	Yarı iletken- MOSFET (sourcing)
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	-	20 VDC min.
Lojik 0 sinyal, 10K $\Omega$ yük ile	-	0.1 VDC max.
Akım (max.)	2.0 A	0.5 A
Lamba yük	30 W DC/200 W AC	5 W
ON durumu temas direnci	0.2 $\Omega$ max. yeni olduğunda	0.6 $\Omega$ max.
Nokta başına kaçak akım	--	10 $\mu$ A max.
Ani aşırı akım	7 A kontakların kapanmasıyla	8 A için 100 ms max.
Aşırı yük koruması	Hayır	Hayır
İzolasyon (alan tarafı ile lojik)	1500 VAC için 1 dakika (bobin ve kontak) Hiçbiri (bobin ile lojik)	500 VAC için 1 dakika
İzolasyon direnci	100 M $\Omega$ min. yeni olduğunda	-
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika	-
İzolasyon grupları	4	1
Ortak nokta başına akım (max.)	10 A	8 A
İndüktif kenetleme gerilimi	-	L+ eksi 48 V, 1 W ısı olarak kayıp
Anahtarlama gecikmesi	10 ms max.	50 $\mu$ s max. off - on geçişi 200 $\mu$ s max. on - off geçişi
Maksimum röle anahtarlama frekansı	1 Hz	-
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler	-
Kontakların ömrü (N.A. kontak)	100,000 açık/kapalı döngüler	-
RUN'dan S-P'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	<ul style="list-style-type: none"> <li>8 (bitişik noktalar değil) 60 °C yatayda veya 50 °C dikey</li> <li>16, 55 °C yatayda veya 45 °C dikey</li> </ul>	16
Kablo uzunluğu (metre)	500 m zırlı, 150 m zırhsız	500 m zırlı, 150 m zırhsız

Çizelge A- 115 8-nokta dijital çıkış SM'ler için bağlantı diyagramları



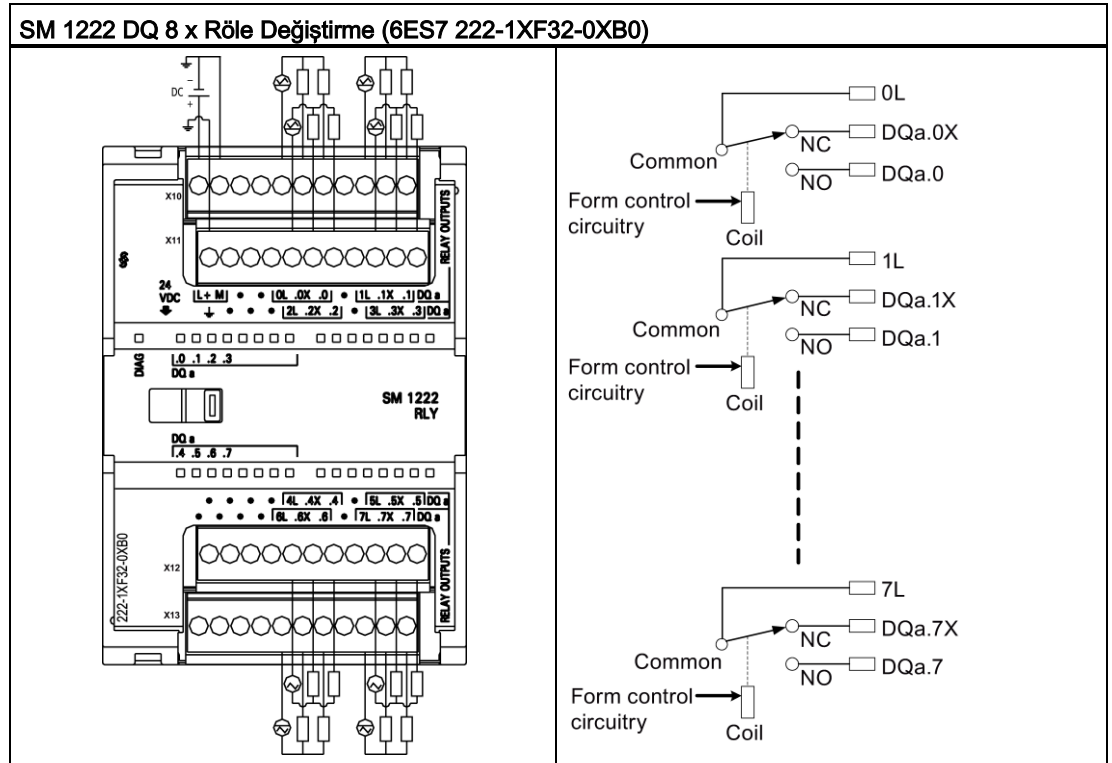
Çizelge A- 116 SM 1222 DQ 8 x Röle (6ES7 222-1HF32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11
1	L+ / 24 VDC	Bağlantı yok
2	M / 24 VDC	2L
3	Fonksiyonel toprak	DQ a.3
4	1L	DQ a.4
5	DQ a.0	DQ a.5
6	DQ a.1	DQ a.6
7	DQ a.2	DQ a.7

Çizelge A- 117 SM 1222 DQ 8 x 24 VDC (6ES7 222-1BF32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11
1	L+ / 24 VDC	Bağlantı yok
2	M / 24 VDC	Bağlantı yok
3	Fonksiyonel toprak	Bağlantı yok
4	DQ a.0	DQ a.4
5	DQ a.1	DQ a.5
6	DQ a.2	DQ a.6
7	DQ a.2	DQ a.7

Çizelge A- 118 8-nokta dijital çıkış rölesi konum değiştirme SM'ler için bağlantı diyagramı

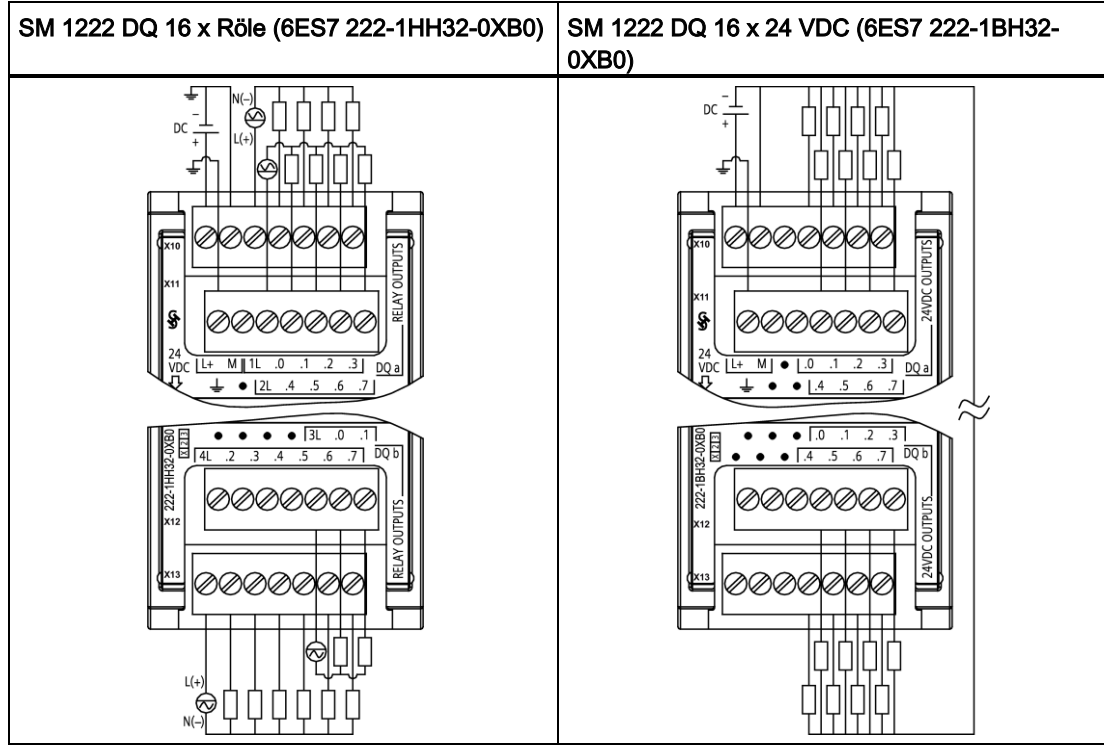


Bir değiştirme rölesi çıkışı bir ortak terminal kullanarak iki devreyi kontrol eder: Birisi normalde kapalı kontak, diğeri normalde açık kontak. Bir örnek olarak "0" çıkış kullanıldığında, çıkış noktası OFF olduğunda, ortak (OL) normalde kapalı kontakta (.0X) bağlanır ve normalde açık kontakta (.0) ayrılır. Çıkış noktası ON olduğunda, ortak (OL) normalde kapalı kontakta (.0X) ayrılır ve normalde açık kontakta (.0) bağlanır.

Çizelge A- 119 SM 1222 DQ 8 x Röle Değişirme (6ES7 222-1XF32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	Fonksiyonel toprak	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
5	0L	2L	4L	6L
6	DQ a.0X	DQ a.2X	DQ a.4X	DQ a.6X
7	DQ a.0	DQ a.2	DQ a.4	DQ a.6
8	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
9	1L	3L	5L	7L
10	DQ a.1X	DQ a.3X	DQ a.5X	DQ a.7X
11	DQ a.1	DQ a.3	DQ a.5	DQ a.7

Çizelge A- 120 16-nokta dijital çıkış SM'ler için bağlantı diyagramları



Çizelge A- 121 SM 1222 DQ 16 x Röle (6ES7 222-1HH32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	Fonksiyonel toprak	Bağlantı yok	4L
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	DQ b.2
3	1L	2L	Bağlantı yok	DQ b.3
4	DQ a.0	DQ a.4	Bağlantı yok	DQ b.4
5	DQ a.1	DQ a.5	3L	DQ b.5
6	DQ a.2	DQ a.6	DQ b.0	DQ b.6
7	DQ a.3	DQ a.7	DQ b.1	DQ b.7

Çizelge A- 122 SM 1222 DQ 16 x 24 VDC (6ES7 222-1BH32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	Fonksiyonel toprak	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	DQ a.0	DQ a.4	DQ b.0	DQ b.4
5	DQ a.1	DQ a.5	DQ b.1	DQ b.5
6	DQ a.2	DQ a.6	DQ b.2	DQ b.6
7	DQ a.3	DQ a.7	DQ b.3	DQ b.7

#### A.7.4 SM 1223 dijital giriş/çıkış VDC özellikleri

Çizelge A- 123 Genel özellikler

Model	SM 1223 DI 8 x 24 VDC, DQ 8 x Röle	SM 1223 DI 16 x 24 VDC, DQ 16 x Röle	SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC	SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC
Sipariş numarası	6ES7 223-1PH32-0XB0	6ES7 223-1PL32-0XB0	6ES7 223-1BH32-0XB0	6ES7 223-1BL32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	70 x 100 x 75	45 x 100 x 75	70 x 100 x 75
Ağırlık	230 gram	350 gram	210 gram	310 gram
Güç kaybı	5.5 W	10 W	2.5 W	4.5 W
Çektiği akım(SM Bus)	145 mA	180 mA	145 mA	185 mA
Çektiği akım(24 VDC)	4 mA / Giriş kullanıldı 11 mA / Röle bobini kullanıldı		150 mA	200 mA

Çizelge A- 124 Dijital girişler

Model	SM 1223 DI 8 x 24 VDC, DQ 8 x Röle	SM 1223 DI 16 x 24 VDC, DQ 16 x Röle	SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC	SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC
Giriş sayısı	8	16	8	16
Tip	Sink/Source (IEC Tip 1 sink)			
Anma gerilimi	24 VDC 4 mA'de, nominal			
Sürekli izin verilebilir gerilim	30 VDC max.			
Aşırı gerilim	35 VDC için 0.5 saniye			
Lojik 1 sinyal (min.)	15 VDC, 2.5 mA'de			
Lojik 0 sinyal (max.)	5 VDC, 1 mA'de			
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika			
İzolasyon grupları	2	2	2	2
Filtre süreleri	0.2, 0.4, 0.8, 1.6, 3.2, 6.4 ve 12.8 ms, grup 4 içinde seçilebilir			
Aynı anlı girişlerin sayısı	8	<ul style="list-style-type: none"> <li>8 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li> <li>16, 55 °C yatayda veya 45 °C dikey</li> </ul>	8	16
Kablo uzunluğu (metre)	500 m zırlı, 300 m zırsız			

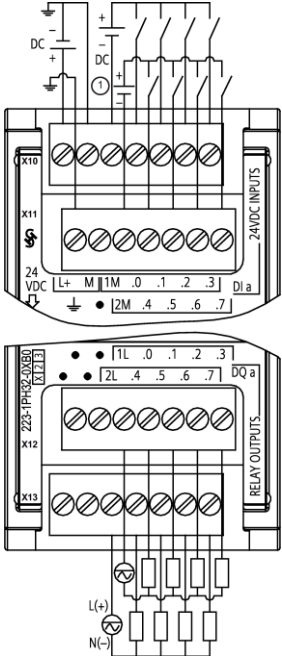
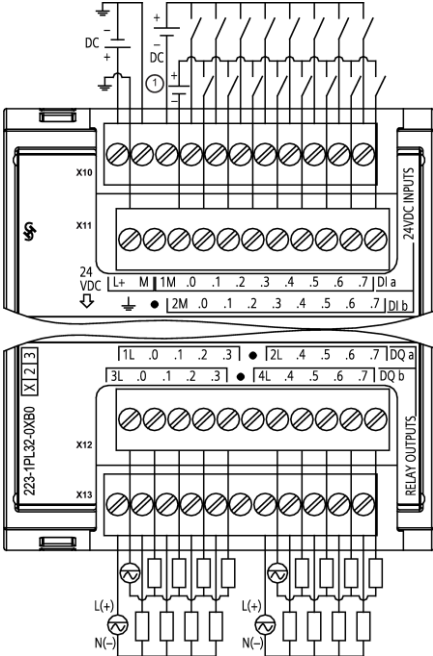
## Teknik özellikler

### A.7 Dijital sinyal modülleri (SM'ler)

Çizelge A- 125 Dijital çıkışlar

Model	SM 1223 DI 8 x 24 VDC, DQ 8 x Röle	SM 1223 DI 16 x 24 VDC, DQ 16 x Röle	SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC	SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC
Giriş sayısı	8	16	8	16
Tip	Röle, kuru kontak		Yarı iletken- MOSFET (sourcing)	
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC		20.4 - 28.8 VDC	
Lojik 1 sinyal max. akımda	--		20 VDC, min.	
Lojik 0 sinyal, 10 KΩ yük ile	--		0.1 VDC, max.	
Akım (max.)	2.0 A		0.5 A	
Lamba yük	30 W DC / 200 W AC		5 W	
ON durumu temas direnci	0.2 Ω max. yeni olduğunda		0.6 Ω max.	
Nokta başına kaçak akım	--		10 µA max.	
Ani aşırı akım	7 A kontakların kapanmasıyla		8 A için 100 ms max.	
Aşırı yük koruması	Hayır			
İzolasyon (alan tarafı ile lojik)	1500 VAC için 1 dakika (bobin ve kontak) Hiçbiri (bobin ile lojik)		500 VAC için 1 dakika	
İzolasyon direnci	100 MΩ min. yeni olduğunda		--	
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika		--	
İzolasyon grupları	2	4	1	1
Ortak nokta başına akım	10A	8 A	4 A	8 A
İndüktif kenetleme gerilimi	--		L+ eksi 48 V, 1 W ısı olarak kayıp	
Anahtarlama gecikmesi	10 ms max.		50 µs max. off - on geçişi 200 µs max. on - off geçişi	
Maksimum röle anahtarlama frekansı	1 Hz		--	
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler		--	
Kontakların ömrü (N.A. kontak)	100,000 açık/kapalı döngüler		--	
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)			
Aynı andaki çıkış sayısı	8	<ul style="list-style-type: none"> <li>8 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li> <li>16, 55 °C yatayda veya 45 °C dikey</li> </ul>	8	16
Kablo uzunluğu (metre)	500 m zırlı, 150 m zırsız			

Çizelge A- 126 Dijital giriş VDC/çıkış rölesi SM'ler için bağlantı diyagramları

SM 1223 DI 8 x 24 VDC, DQ 8 x Röle (6ES7 223-1PH32-0XB0)	SM 1223 DI 16 x 24 VDC, DQ 16 x Röle (6ES7 223-1PL32-0XB0)	Not
		<p>① Sinking girişler için "-" ucunu "M" ucuna bağlayınız (gösterilmiştir)</p> <p>Sourcing girişler için "+" ucunu "M" ucuna bağlayınız.</p>

Çizelge A- 127 SM 1223 DI 8 x 24 VDC, DQ 8 x Röle (6ES7 223-1PH32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	GND	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	1L	2L
4	DI a.0	DI a.4	DQ a.0	DQ a.4
5	DI a.1	DI a.5	DQ a.1	DQ a.5
6	DI a.2	DI a.6	DQ a.2	DQ a.6
7	DI a.3	DI a.7	DQ a.3	DQ a.7

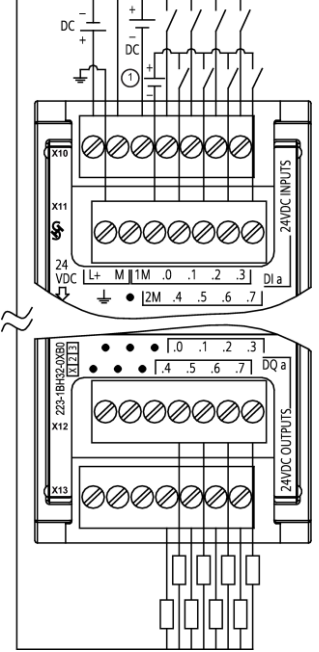
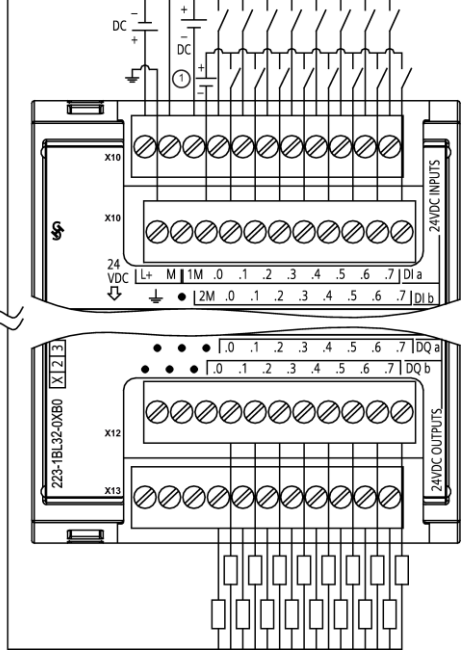
Teknik özellikler

A.7 Dijital sinyal modülleri (SM'ler)

Çizelge A- 128 SM 1223 DI 16 x 24 VDC, DQ 16 x Röle (6ES7 223-1PL32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	GND	1L	3L
2	M / 24 VDC	Bağlantı yok	DQ a.0	DQ b.0
3	1M	2M	DQ a.1	DQ b.1
4	DI a.0	DI b.0	DQ a.2	DQ b.2
5	DI a.1	DI b.1	DQ a.3	DQ b.3
6	DI a.2	DI b.2	Bağlantı yok	Bağlantı yok
7	DI a.3	DI b.3	2L	4L
8	DI a.4	DI b.4	DQ a.4	DQ b.4
9	DI a.5	DI b.5	DQ a.5	DQ b.5
10	DI a.6	DI b.6	DQ a.6	DQ b.6
11	DI a.7	DI b.7	DQ a.7	DQ b.7

Çizelge A- 129 Dijital giriş VDC/çıkış SM'ler için bağlantı diyagramları

SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC (6ES7 223-1BH32-0XB0)	SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC (6ES7 223-1BL32-0XB0)	Not
		<p>① Sinking girişler için "-" ucunu "M" ucuna bağlayınız (gösterilmiştir)</p> <p>Sourcing girişler için "+" ucunu "M" ucuna bağlayınız.</p>



Çizelge A- 130 SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC (6ES7 223-1BH32-0XB0) için konektör Pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	GND	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	Bağlantı yok	Bağlantı yok
4	DI a.0	DI a.4	DQ a.0	DQ a.4
5	DI a.1	DI a.5	DQ a.1	DQ a.5
6	DI a.2	DI a.6	DQ a.2	DQ a.6
7	DI a.3	DI a.7	DQ a.3	DQ a.7

Çizelge A- 131 SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC (6ES7 223-1BL32-0XB0) için konektör Pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	GND	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1M	2M	Bağlantı yok	Bağlantı yok
4	DI a.0	DI b.0	DQ a.0	DQ b.0
5	DI a.1	DI b.1	DQ a.1	DQ b.1
6	DI a.2	DI b.2	DQ a.2	DQ b.2
7	DI a.3	DI b.3	DQ a.3	DQ b.3
8	DI a.4	DI b.4	DQ a.4	DQ b.4
9	DI a.5	DI b.5	DQ a.5	DQ b.5
10	DI a.6	DI b.6	DQ a.6	DQ b.6
11	DI a.7	DI b.7	DQ a.7	DQ b.7

## A.7.5 SM 1223 dijital giriş/çıkış AC özellikleri

Çizelge A- 132 Genel özellikler

<b>Model</b>	<b>SM 1223 DI 8 x120/230 VAC / DQ 8 x Röle</b>
Sipariş numarası	6ES7 223-1QH32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75 mm
Ağırlık	190 gram
Güç kaybı	7.5 W
Çektiği akım(SM Bus)	120 mA
Çektiği akım(24 VDC)	11 mA, ON olduğunda çıkış başına

## Teknik özellikler

### A.7 Dijital sinyal modülleri (SM'ler)

Çizelge A- 133 Dijital girişler

Model	SM 1223 DI 8 x 120/230 VAC / DQ 8 x Röle
Giriş sayısı	8
Tip	IEC Tip 1
Anma gerilimi	120 VAC 6 mA'de, 230 VAC 9 mA'de
Sürekli izin verilebilir gerilim	264 VAC
Aşırı gerilim	--
Lojik 1 sinyal (min.)	79 VAC 2.5 mA'de
Lojik 0 sinyal (max.)	20 VAC 1 mA'de
Kaçak akım (max.)	1 mA
İzolasyon (alan tarafı ile lojik arasında)	1500 VAC için 1 dakika
İzolasyon grupları <sup>1</sup>	4
Giriş gecikme süreleri	Tipik olarak: 0.2 - 12.8 ms, kullanıcı seçim çizelgesi Maksimum: -
2-telli yaklaşım sensörü bağlantısı (Bero) (max.)	1 mA
Kablo uzunluğu	Zırhsız: 300 metre Zırhlı: 500 metre
Aynı anlı girişlerin sayısı	8

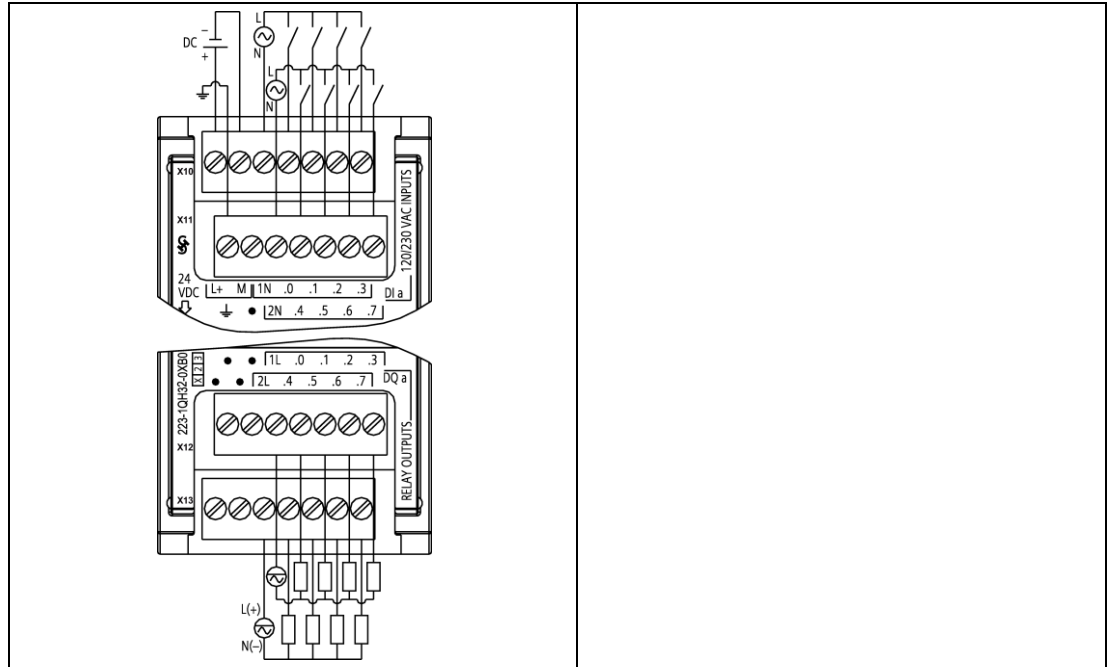
<sup>1</sup> Bir grup içindeki kanallar aynı faza ait olmalıdır.

Çizelge A- 134 Dijital çıkışlar

Model	SM 1223 DI 8 x 120/230 VAC / DQ 8 x Röle
Giriş sayısı	8
Tip	Röle, kuru kontak
Gerilim aralığı	5 - 30 VDC veya 5 - 250 VAC
Lojik 1 max. akımdaki sinyal	--
Lojik 0 sinyal, 10K $\Omega$ yük ile	--
Akım (max.)	2.0 A
Lamba yük	30 W DC / 200 W AC
ON durumu temas direnci	0.2 $\Omega$ max. yeni olduğunda
Nokta başına kaçak akım	--
Ani aşırı akım	7 A kontakların kapanmasıyla
Aşırı yük koruması	Hayır
İzolasyon (alan tarafı ile lojik arasında)	1500 VAC için 1 dakika (bobin ve kontak arasında) Hiçbiri (bobin ile lojik arasında)
İzolasyon direnci	100 M $\Omega$ min. yeni olduğunda
Açık kontaklar arasında izolasyon	750 VAC için 1 dakika
İzolasyon grupları	2
Ortak nokta başına akım (max.)	10 A
İndüktif kenetleme gerilimi	--

<b>Model</b>	<b>SM 1223 DI 8 x 120/230 VAC / DQ 8 x Röle</b>
Anahtarlama gecikmesi (max.)	10 ms
Maksimum röle anahtarlama frekansı	1 Hz
Mekanik ömür (yüksüz)	10,000,000 açık/kapalı döngüler
Anma yükünde ömür boyu temaslar	1000,000 açık/kapalı döngüler
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	<ul style="list-style-type: none"> <li>4 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li> <li>8, 55 °C yatayda veya 45 °C dikey</li> </ul>
Kablo uzunluğu (metre)	500 m zırhlı, 150 m zırhsız

Çizelge A- 135 SM 1223 DI 8 x 120/230 VAC, DQ 8 x Röle (6ES7 223-1QH32-0XB0)



Çizelge A- 136 SM 1223 DI 8 x 120/240 VAC, DQ 8 x Röle (6ES7 223-1QH32-0XB0) için konektör Pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	GND	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	1N	2N	1L	2L
4	DI a.0	DI a.4	DQ a.0	DQ a.4
5	DI a.1	DI a.5	DQ a.1	DQ a.5
6	DI a.2	DI a.6	DQ a.2	DQ a.6
7	DI a.3	DI a.7	DQ a.3	DQ a.7

## Teknik özellikler

## A.8 Analog sinyal modülleri (SM'ler)

## A.8 Analog sinyal modülleri (SM'ler)

## A.8.1 SM 1231 Analog sinyal modülleri (SM'ler)

Çizelge A- 137 Genel özellikler

Model	SM 1231 AI 4 x 13 bit	SM 1231 AI 8 x 13 bit	SM 1231 AI 4 x 16 bit
Sipariş numarası	6ES7 231-4HD32-0XB0	6ES7 231-4HF32-0XB0	6ES7 231-5ND30-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	45 x 100 x 75	45 x 100 x 75
Ağırlık	180 gram	180 gram	180 gram
Güç kaybı	2.2 W	2.3 W	2.0 W
Çektiği akım(SM Bus)	80 mA	90 mA	80 mA
Çektiği akım(24 VDC)	45 mA	45 mA	65 mA

Çizelge A- 138 Analog girişler

Model	SM 1231 AI 4 x 13 bit	SM 1231 AI 8 x 13 bit	SM 1231 AI 4 x 16 bit
Giriş sayısı	4	8	4
Tip	Gerilim veya akım (diferansiyel): 2 grubu seçim Çizelgesinde		Gerilim veya akım (diferansiyel)
Aralık	±10 V, ±5 V, ±2.5 V, 0 - 20 mA, veya 4 mA - 20 mA		±10 V, ±5 V, ±2.5 V, ±1.25 V, 0 - 20 mA veya 4 mA - 20 mA
Tam ölçek aralığı (veri word)	-27648 ila 27648 gerilim / 0 - 27648 akım		
Üst taşma /alt taşma aralığı (veri word) Akım ve gerilim için analog giriş aralıkları bölümüne bakınız (Sayfa 917).	Gerilim:32511 - 27649 / -27649 ila -32,512 Akım:32511 - 27649 / 0 ila -4864		
Üst taşma/alt taşma (veri word) Akım ve gerilim için analog giriş aralıkları bölümüne bakınız (Sayfa 916).	Gerilim:32767 - 32512 / -32513 ila -32768 Akım 0 - 20 mA: 32767 - 32512 / -4865 ila -32768 Akım 4 - 20 mA: 32767 - 32512 (-4864 altındaki değerler kablo kopukluğunu gösterir)		
Çözünürlük <sup>1</sup>	12 bit + işaret biti		15 bit + işaret biti
Maksimum dayanma gerilimi/akım	±35 V / ±40 mA		
Düzleme	Hiçbiri, zayıf, orta veya güçlü Basamak tepki süreleri üzerine seçime bakınız (Sayfa 9176).		
Gürültü bastırma	400, 60, 50 veya 10 Hz Örnekleme hızı üzerine seçime bakınız (Sayfa 9176).		
Giriş empedansı	≥ 9 MΩ (gerilim) / 280 Ω (akım)		≥ 1 MΩ (gerilim) / <315 Ω, >280 Ω (akım)

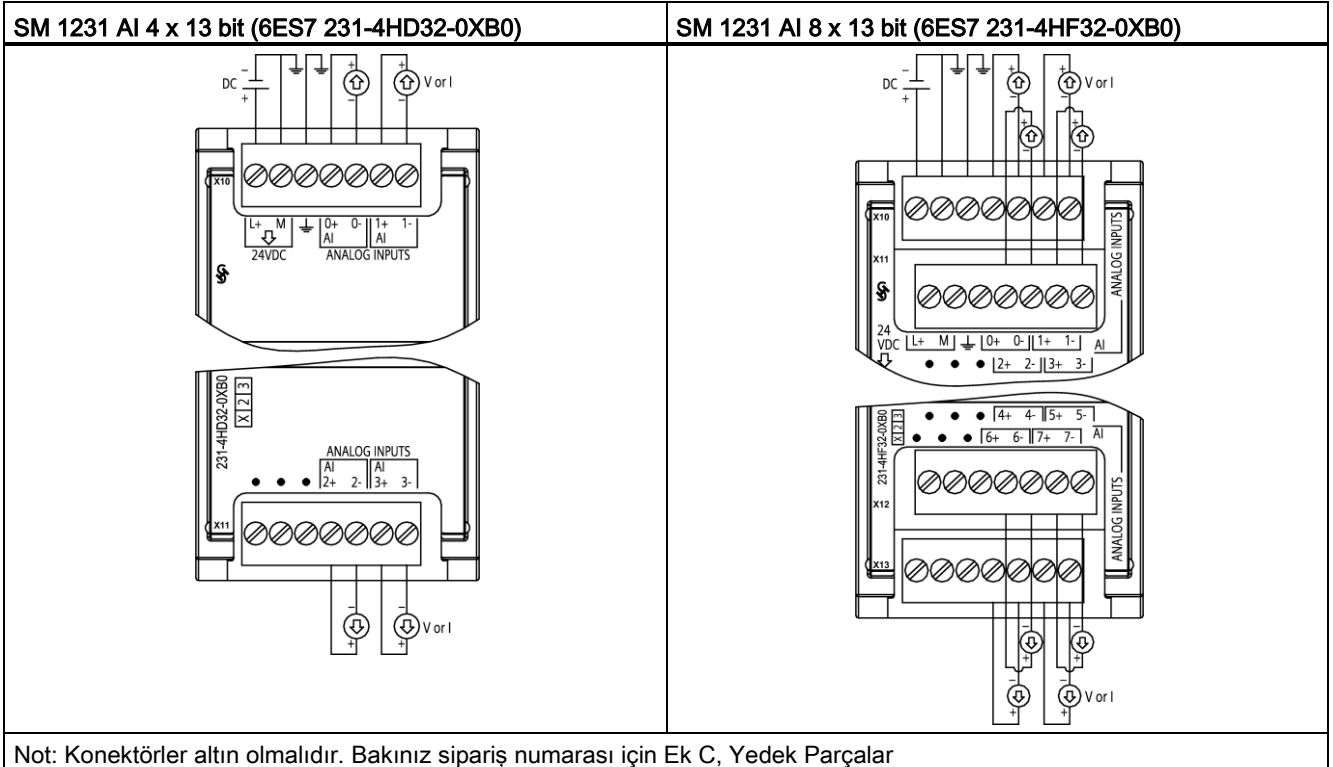
Model	SM 1231 AI 4 x 13 bit	SM 1231 AI 8 x 13 bit	SM 1231 AI 4 x 16 bit
İzolasyon	Hiçbiri		
Alan tarafı ile lojik arasında			500 VAC
Lojik ile 24 VDC arasında			500 VAC
Alan tarafı ile 24 VDC arasında			500 VAC
Kanal - kanal arasında			Hiçbiri
Doğruluk (25 °C / -20 - 60 °C)	Tam ölçüğe göre $\pm 0.1\%$ / $\pm 0.2\%$		Tam ölçüğe göre $\pm 0.1\%$ / $\pm 0.3\%$
Ölçme prensibi	Gerçek değer dönüşümü		
Ortak mod bastırma	40 dB, DC - 60 Hz		
Operasyonel tek aralık <sup>1</sup>	Sinyal artı ortak mod gerilimi +12 V'tan az ve -12 V'tan büyük olmalıdır.		
Kablo uzunluğu (metre)	100 m, bükülü ve zırlı		

<sup>1</sup> Bir kanal için gerilimin operasyonel aralığın dışında olması diğer kanallar üzerinde etkileşime neden olabilir.

Çizelge A- 139 Diyagnostikler

Model	SM 1231 AI 4 x 13 bit	SM 1231 AI 8 x 13 bit	SM 1231 AI 4 x 16 bit
Üst taşma/alt taşma	Evet	Evet	Evet
24 VDC düşük gerilim	Evet	Evet	Evet
Açık tel (kablo)	4- 20 mA aralığı sadece (-4164 altında ise; giriş 1.185 mA)		

Çizelge A- 140 Analog giriş SM'ler için bağlantı diyagramları



Teknik özellikler

A.8 Analog sinyal modülleri (SM'ler)

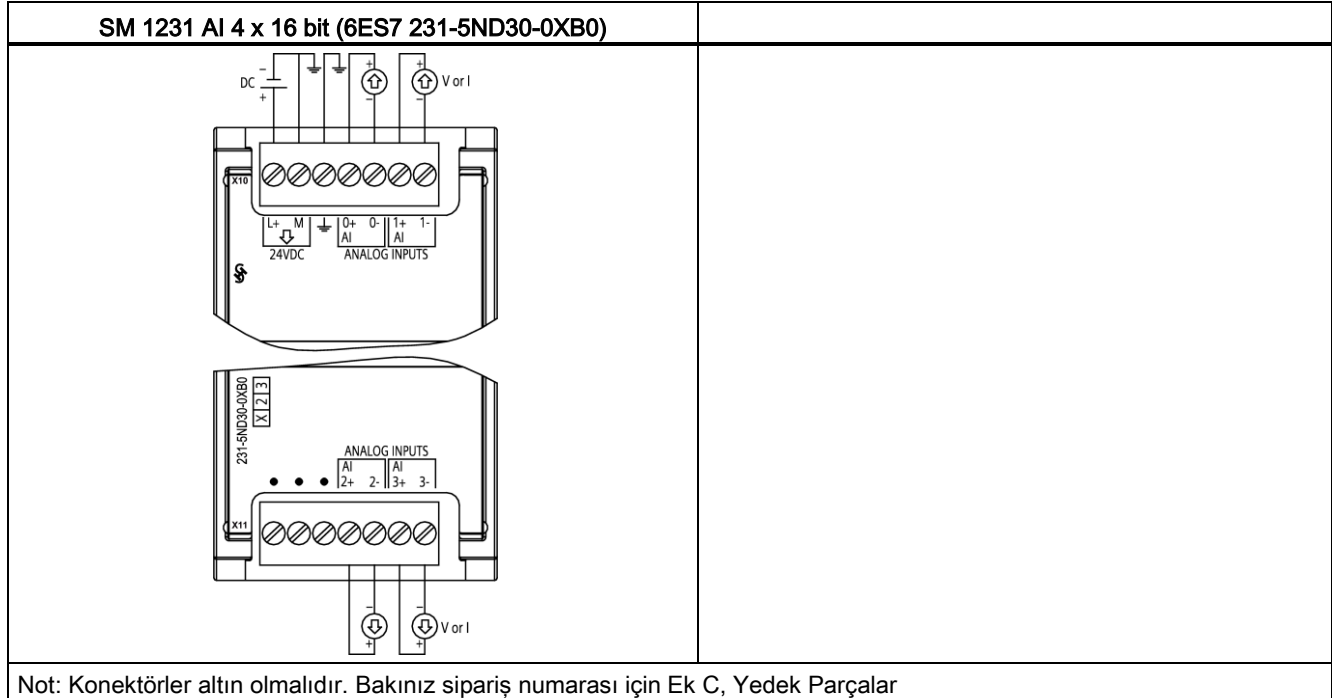
Çizelge A- 141 SM 1231 AI 4 x 13 bit (6ES7 231-4HD32-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)
1	L+ / 24 VDC	Bağlantı yok
2	M / 24 VDC	Bağlantı yok
3	GND	Bağlantı yok
4	AI 0+	AI 2+
5	AI 0-	AI 2-
6	AI 1+	AI 3+
7	AI 1-	AI 3-

Çizelge A- 142 SM 1231 AI 8 x 13 bit (6ES7 231-4HF32-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)	X12 (altın)	X13 (altın)
1	L+ / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	GND	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0+	AI 2+	AI 4+	AI 6+
5	AI 0-	AI 2-	AI 4-	AI 6-
6	AI 1+	AI 3+	AI 5+	AI 7+
7	AI 1-	AI 3-	AI 5-	AI 7-

Çizelge A- 143 Analog giriş SM'ler için bağlantı diyagramı



Çizelge A- 144 SM 1231 AI 4 x 16 bit (6ES7 231-5ND30-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)
1	L+ / 24 VDC	Bağlantı yok
2	M / 24 VDC	Bağlantı yok
3	GND	Bağlantı yok
4	AI 0+	AI 2+
5	AI 0-	AI 2-
6	AI 1+	AI 3+
7	AI 1-	AI 3-

**Not**

Kullanılmayan gerilim kanalları kısa devre edilmelidir.

Kullanılmayan akım giriş kanalları 0-20mA aralığına ve/veya etkisizleştirmeye (disable) (tel kopma hata raporlamasına) ayarlanmalıdır.

Geçerli mod için konfigüre edilen girişler modül enerjilenmedikçe veya konfigüre edilmedikçe döngü akışını gerçekleştirmeyecektir.

Akım giriş kanalları aktarıcıya harici güç sağlanmadıkça çalışmayacaktır.

## A.8.2 SM 1232 analog çıkış modülü özellikleri

Çizelge A- 145 Genel özellikler

Teknik veri	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Sipariş numarası	6ES7 232-4HB32-0XB0	6ES7 232-4HD32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	45 x 100 x 75
Ağırlık	180 gram	180 gram
Güç kaybı	1.8 W	2.0 W
Çektiği akım(SM Bus)	80 mA	80 mA
Çektiği akım(24 VDC)	45 mA (yüksüz)	45 mA (yüksüz)

Çizelge A- 146 Analog çıkışlar

Teknik veri	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Giriş sayısı	2	4
Tip	Gerilim veya akım	Gerilim veya akım
Aralık	±10 V, 0 - 20 mA veya 4 mA - 20 mA	±10 V, 0 - 20 mA veya 4 mA - 20 mA
Çözünürlük	Gerilim:14 bit Akım:13 bit	Gerilim:14 bit Akım:13 bit

## Teknik özellikler

### A.8 Analog sinyal modülleri (SM'ler)

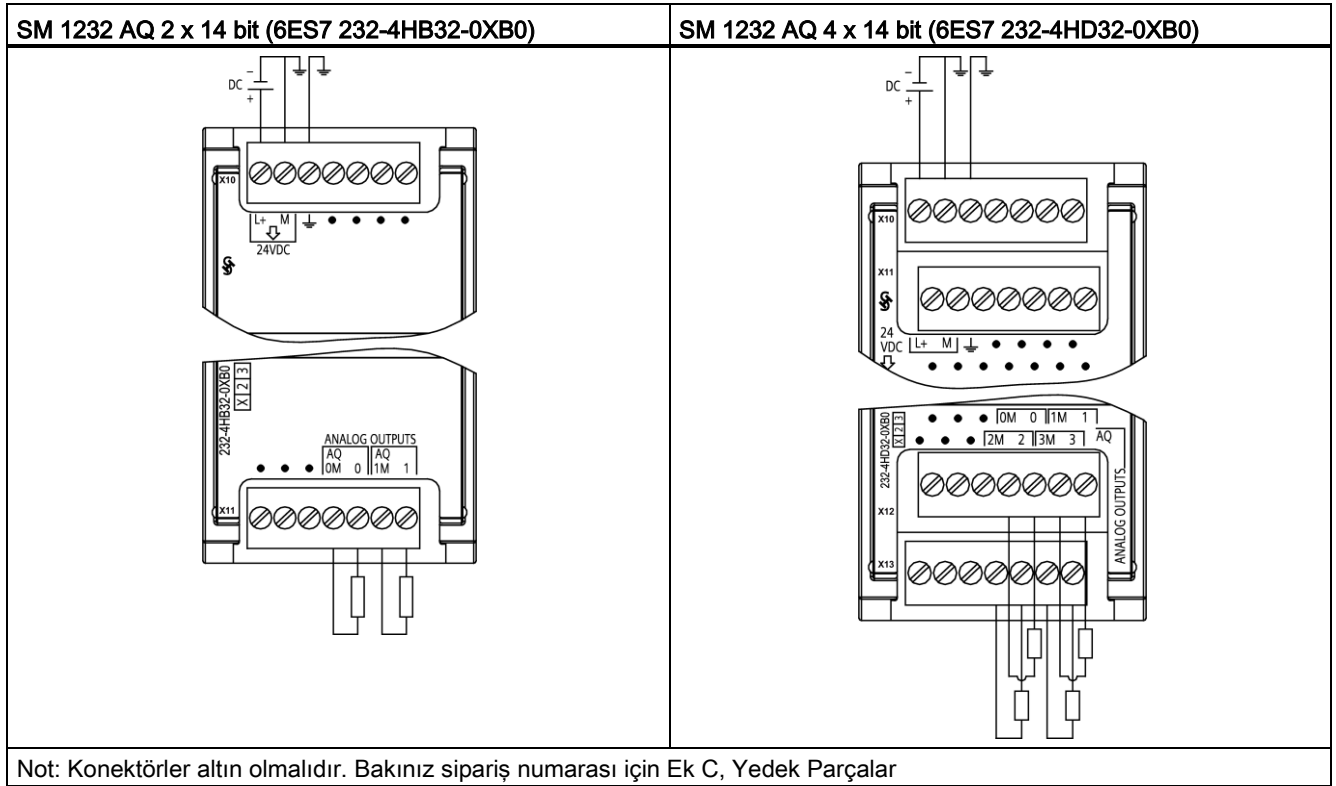
Teknik veri	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Tam-ölçek aralığı(veri word)	Gerilim:-27,648 ila 27,648 ; Akım: 0 - 27,648 Gerilim ve akım için çıkış aralıklarına bakınız (Sayfa 917).	
Doğruluk (25 °C / -20 ila 60 °C)	Tam ölçeğe göre $\pm 0.3\%$ / $\pm 0.6\%$	
Yatışma süresi (yeni değer için %95'i)	Gerilim:300 $\mu$ S (R), 750 $\mu$ S (1 uF) Akım:600 $\mu$ S (1 mH), 2 ms (10 mH)	
Yük empedansı	Gerilim: $\geq 1000 \Omega$ Akım: $\leq 600 \Omega$	
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	
İzolasyon (alan tarafı ile lojik)	hiçbiri	
Kablo uzunluğu (metre)	100 m bükülü ve zırlı	

### Çizelge A- 147 Diyagnostikler

Teknik veri	SM 1232 AQ 2 x 14 bit	SM 1232 AQ 4 x 14 bit
Taşma/alt taşma	Evet	Evet
Toprağa kısa devre (gerilim modu sadece)	Evet	Evet
Kablo kopması(akım modu sadece)	Evet	Evet
24 VDC düşük gerilim	Evet	Evet



Çizelge A- 148 Analog çıkış SM'ler için bağlantı diyagramları



Çizelge A- 149 SM 1232 AQ 2 x 14 bit (6ES7 232-4HB32-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)
1	L+ / 24 VDC	Bağlantı yok
2	M / 24 VDC	Bağlantı yok
3	GND	Bağlantı yok
4	Bağlantı yok	AQ 0M
5	Bağlantı yok	AQ 0
6	Bağlantı yok	AQ 1M
7	Bağlantı yok	AQ 1

## Teknik özellikler

### A.8 Analog sinyal modülleri (SM'ler)

Çizelge A- 150 SM 1232 AQ 4 x 14 bit (6ES7 232-4HD32-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)	X12 (altın)	X13 (altın)
1	L+ / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	GND	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	Bağlantı yok	Bağlantı yok	AQ 0M	AQ 2M
5	Bağlantı yok	Bağlantı yok	AQ 0	AQ 2
6	Bağlantı yok	Bağlantı yok	AQ 1M	AQ 3M
7	Bağlantı yok	Bağlantı yok	AQ 1	AIQ 3

### A.8.3 SM 1234 analog giriş/çıkış modülü özellikleri

Çizelge A- 151 Genel özellikler

Teknik veri	SM 1234 AI 4 x 13 bit / AQ 2 x 14 bit
Sipariş numarası	6ES7 234-4HE32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75
Ağırlık	220 gram
Güç kaybı	2.4 W
Çektiği akım(SM Bus)	80 mA
Çektiği akım(24 VDC)	60 mA (yüksüz)

Çizelge A- 152 Analog girişler

Model	SM 1234 AI 4 x 13 bit / AQ 2 x 14 bit
Giriş sayısı	4
Tip	Gerilim veya akım (diferansiyel): grup 2'd1 seçilebilir.
Aralık	$\pm 10$ V, $\pm 5$ V, $\pm 2.5$ V, 0 - 20 mA, veya 4 mA - 20 mA
Tam-ölçek aralığı(veri word)	-27648 - 27648
Aşma/alt aşma aralığı (veri word)	Gerilim: 32511 - 27649 / -27649 ila -32512 Akım: 32511 - 27649 / 0 ila -4864 Gerilim ve akım için giriş aralıkları kısmına bakınız (Sayfa 9176).
Taşma/alt taşma (veri word)	Gerilim: 32767 - 32512 / -32513 ila -32768 Akım: 32767 - 32512 / -4865 ila -32768 Gerilim ve akım için giriş aralıkları kısmına bakınız (Sayfa 9176).
Çözünürlük	12 bit + işaret biti
Maksimum dayanma gerilimi/akım	$\pm 35$ V / $\pm 40$ mA
Düzleme	Hiçbiri, zayıf, orta veya güçlü Basamak tepki süreleri kısmına bakınız (Sayfa 9176).
Gürültü bastırma	400, 60, 50 veya 10 Hz Örnekleme hızı kısmına bakınız (Sayfa 9176).

Model	SM 1234 AI 4 x 13 bit / AQ 2 x 14 bit
Giriş empedansı	≥ 9 MΩ (gerilim) / 280 Ω (akım)
İzolasyon (alan tarafı ile lojik)	Hiçbiri
Doğruluk (25 °C / -20 ila 60 °C)	Tam ölçüğe göre ±0.1% / ±0.2%
Analog-dijital dönüştürme süresi	625 µs (400 Hz bastırma)
Ortak mod bastırma	40 dB, DC ila 60 Hz
Operasyonel sinyal aralığı <sup>1</sup>	Sinyal artı ortak mod gerilimi +12 V'tan az ve -12 V'tan büyük olmalıdır.
Kablo uzunluğu (metre)	100 m, bükülü ve zırlı

<sup>1</sup> Bir kanala uygulanan gerilimin operasyonel aralığın dışında olması diğer kanallar üzerinde etkileşime neden olabilir.

Çizelge A- 153 Analog çıkışlar

Teknik veri	SM 1234 AI 4 x 13 bit / AQ 2 x 14 bit
Giriş sayısı	2
Tip	Gerilim veya akım
Aralık	±10 V veya 0 - 20 mA
Çözünürlük	Gerilim:14 bit ; Akım:13 bit
Tam ölçek aralığı (veri word)	Gerilim: -27648 ila 27648; Akım: 0 - 27648 Gerilim ve akım için çıkış aralıkları kısmına bakınız (Sayfa 9177).
Doğruluk (25 °C / -20 ila 60 °C)	Tam ölçüğe göre ±0.3% / ±0.6%
Yatışma süresi (yeni değer 95%)	Gerilim: 300 µS (R), 750 µS (1 uF) Akım: 600 µS (1 mH), 2 ms (10 mH)
Yük empedansı	Gerilim: ≥ 1000 Ω Akım: ≤ 600 Ω
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)
İzolasyon (alan tarafı ile lojik)	hiçbiri
Kablo uzunluğu (metre)	100 m bükülü ve zırlı

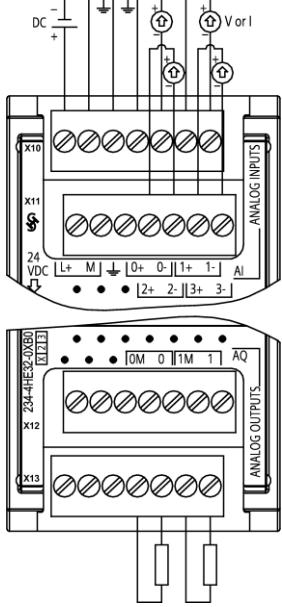
Çizelge A- 154 Diyagnostikler

Model	SM 1234 AI 4 x 13 bit / AQ 2 x 14 bit
Taşma/alt taşma	Evet
Toprağa kısa devre (gerilim modunda sadece)	Evet, çıkışlar üzerinde
Kablo kopması (akım modunda sadece)	Evet, çıkışlar üzerinde
24 VDC düşük gerilim	Evet

## Teknik özellikler

### A.8 Analog sinyal modülleri (SM'ler)

Çizelge A- 155 Analog giriş/çıkış SM'ler için bağlantı diyagramları

SM 1234 AI 4 x 13 Bit / AQ 2 x 14 bit (6ES7 234-4HE32-0XB0)	
	
<p>Not: Konektörler altın olmalıdır. Bakınız sipariş numarası için Ek C, Yedek Parçalar</p>	

Çizelge A- 156 SM 1234 AI 4 x 13 Bit / AQ 2 x 14 bit (6ES7 234-4HE32-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)	X12 (altın)	X13 (altın)
1	L+ / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	GND	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0+	AI 2+	Bağlantı yok	AQ 0M
5	AI 0-	AI 2-	Bağlantı yok	AQ 0
6	AI 1+	AI 3+	Bağlantı yok	AQ 1M
7	AI 1-	AI 3-	Bağlantı yok	AQ 1

#### Not

Kullanılmayan gerilim kanalları kısa devre edilmelidir.

Kullanılmayan akım giriş kanalları 0-20mA aralığına ve/veya etkisizleştirmeye (disable) (tel kopma hata raporlamasına) ayarlanmalıdır.

Geçerli mod için konfigüre edilen girişler modül enerjilenmedikçe veya konfigüre edilmedikçe döngü akışını gerçekleştirmeyecektir.

Akım giriş kanalları aktarıcıya harici güç sağlanmadıkça çalışmayacaktır.

### A.8.4 Analog girişlerin basamak tepkileri

Çizelge A- 157 Basamak tepkisi(ms), 0, %95'de tam-ölçekli ölçülen

Düzleme seçimi (örnek ortalama)	Gürültü azaltma/Bastırma frekansı (İntegrasyon süresi seçimi)			
	400 Hz (2.5 ms)	60 Hz (16.6 ms)	50 Hz (20 ms)	10 Hz (100 ms)
Hiçbiri (1 döngü): Ortalama alma yok	4 ms	18 ms	22 ms	100 ms
Zayıf (4 döngü): 4 örnek	9 ms	52 ms	63 ms	320 ms
Orta (16 döngü): 16 örnek	32 ms	203 ms	241 ms	1200 ms
Güçlü (32 döngü): 32 örnek	61 ms	400 ms	483 ms	2410 ms
<b>Örnekleme süresi</b>				
• 4 AI x 13 bit	• 0.625 ms	• 4.17 ms	• 5 ms	• 25 ms
• 8 AI x 13 bit	• 1.25 ms	• 4.17 ms	• 5 ms	• 25 ms
• 4 AI4 x 16 bit	• 0.417 ms	• 0.397 ms	• 0.400 ms	• 0.400 ms

### A.8.5 Analog girişler için örnekleme süresi ve güncelleme süresi

Çizelge A- 158 Örnekleme süresi ve güncelleme süresi

Bastırma frekansı (İntegrasyon süresi)	Örnekleme süresi	Bütün kanallar için modül güncelleme zamanı	
		4-kanal SM	8-kanal SM
400 Hz (2.5 ms)	<ul style="list-style-type: none"> <li>4-kanal SM: 0.625 ms</li> <li>8-kanal SM: 1.250 ms</li> </ul>	0.625 ms	1.250 ms
60 Hz (16.6 ms)	4.170 ms	4.17 ms	4.17 ms
50 Hz (20 ms)	5.000 ms	5 ms	5 ms
10 Hz (100 ms)	25.000 ms	25 ms	25 ms

### A.8.6 Gerilim ve akım (SB ve SM) için analog girişlerin ölçme aralıkları

Çizelge A- 159 Gerilim (SB ve SM) için analog giriş gösterilişi

Sistem		Gerilim Ölçme Aralığı				
Decimal	Hexadecimal	±10 V	±5 V	±2.5 V	±1.25 V	
32767	7FFF	11.851 V	5.926 V	2.963 V	1.481 V	Taşma
32512	7F00					
32511	7EFF	11.759 V	5.879 V	2.940 V	1.470 V	Aşma aralığı
27649	6C01					
27648	6C00	10 V	5 V	2.5 V	1.250 V	Anma aralığı
20736	5100	7.5 V	3.75 V	1.875 V	0.938 V	
1	1	361.7 µV	180.8 µV	90.4 µV	45.2 µV	

## Teknik özellikler

## A.8 Analog sinyal modülleri (SM'ler)

Sistem		Gerilim Ölçme Aralığı			
Decimal	Hexadecimal	$\pm 10$ V	$\pm 5$ V	$\pm 2.5$ V	$\pm 1.25$ V
0	0	0 V	0 V	0 V	0 V
-1	FFFF				
-20736	AF00	-7.5 V	-3.75 V	-1.875 V	-0.938 V
-27648	9400	-10 V	-5 V	-2.5 V	-1.250 V
-27649	93FF				
-32512	8100	-11.759 V	-5.879 V	-2.940 V	-1.470 V
-32513	80FF				
-32768	8000	-11.851 V	-5.926 V	-2.963 V	-1.481 V

Çizelge A- 160 Akım (SB ve SM) için analog giriş gösterilişi

Sistem		Akım ölçme aralığı	
Decimal	Hexadecimal	0 mA - 20 mA	4 mA - 20 mA
32767	7FFF	23.70 mA	22.96 mA
32512	7F00		
32511	7EFF	23.52 mA	22.81 mA
27649	6C01		
27648	6C00	20 mA	20 mA
20736	5100	15 mA	16 mA
1	1	723.4 nA	4 mA + 578.7 nA
0	0	0 mA	4 mA
-1	FFFF		
-4864	ED00	-3.52 mA	1.185 mA
-4865	ECFF		
-32768	8000		

## A.8.7 Gerilim ve akım (SB ve SM) için analog çıkışların ölçme aralıkları

Çizelge A- 161 Gerilim (SB ve SM) için analog çıkış gösterilişi

Sistem		Gerilim çıkış aralığı	
Decimal	Hexadecimal	$\pm 10$ V	
32767	7FFF	Bakınız not 1	Taşma
32512	7F00	Bakınız not 1	
32511	7EFF	11.76 V	Aşma aralığı
27649	6C01		
27648	6C00	10 V	Anma aralığı
20736	5100	7.5 V	
1	1	361.7 $\mu$ V	
0	0	0 V	

Sistem		Gerilim çıkış aralığı	
Decimal	Hexadecimal	± 10 V	
-1	FFFF	-361.7 µV	
-20736	AF00	-7.5 V	
-27648	9400	-10 V	
-27649	93FF		Alt taşma aralığı
-32512	8100	-11.76 V	Alt taşma
-32513	80FF	Bakınız not 1	
-32768	8000	Bakınız not 1	

<sup>1</sup> Aşırı akım veya alt taşma durumunda, analog çıkışlar STOP modun yedek değerini alacaktır.

Çizelge A- 162 Akım (SB ve SM) için analog çıkış gösterilişi

Sistem		Akım çıkış aralığı		
Decimal	Hexadecimal	0 mA - 20 mA	4 mA - 20 mA	
32767	7FFF	Bakınız not 1	Bakınız not 1	Taşma
32512	7F00	Bakınız not 1	Bakınız not 1	
32511	7EFF	23.52 mA	22.81 mA	Aşma aralığı
27649	6C01			Anma aralığı
27648	6C00	20 mA	20 mA	
20736	5100	15 mA	16 mA	
1	1	723.4 nA	4 mA + 578.7 nA	
0	0	0 mA	4mA	
-1	FFFF		4 mA - 578.7 nA	Alt taşma aralığı
-6912	E500		0 mA	Mümkün değil. Çıkış değeri sıfıra sınırlıdır.
-6913	E4FF			
-32512	8100			
-32513	80FF	Bakınız not 1	Bakınız not 1	Alt taşma
-32768	8000	Bakınız not 1	Bakınız not 1	

<sup>1</sup> Aşırı akım veya alt taşma durumunda, analog çıkışlar STOP modun yedek değerini alacaktır.

## A.9 Termokupl ve RTD sinyal modülleri (SM'ler)

### A.9.1 SM 1231 Termokupl

Çizelge A- 163 Genel özellikler

Model	SM 1231 AI 4 x 16 bit TC	SM 1231 AI 8 x 16 bit TC
Sipariş numarası	6ES7 231-5QD32-0XB0	6ES7 231-5QF32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	45 x 100 x 75
Ağırlık	180 gram	190 gram
Güç kaybı	1.5 W	1.5 W
Çektiği akım(SM Bus)	80 mA	80 mA
Çektiği akım(24 VDC) <sup>1</sup>	40 mA	40 mA

<sup>1</sup> 20.4 - 28.8 VDC (Sınıf 2, Sınırlı güç veya PLC'den sensör gücü)

Çizelge A- 164 Analog girişler

Model	SM 1231 AI 4 x 16 bit TC	SM 1231 AI 8 x 16 bit TC	
Giriş sayısı	4	8	
Aralık	Bakınız Termokupl seçim çizelgesi (Sayfa 924).	Bakınız Termokupl seçim çizelgesi (Sayfa 923).	
Nominal aralık (veri word)			
Üst aralık/alt aralık (veri word)			
Taşma/alt taşma (veri word)			
Çözünürlük	Sıcaklık	0.1 °C/0.1 °F	0.1 °C/0.1 °F
	Gerilim	15 bit artı işareti	15 bit artı işareti
Maksimum dayanma gerilimi	± 35 V	± 35 V	
Gürültü bastırma	85 dB, seçili filtre için (10 Hz, 50 Hz, 60 Hz veya 400 Hz)	85 dB, seçili filtre için (10 Hz, 50 Hz, 60 Hz veya 400 Hz)	
Ortak mod bastırma	> 120 dB 120 VAC'de	> 120 dB 120 VAC'de	
Empedans	≥ 10 MΩ	≥ 10 MΩ	
İzolasyon	Alan ile lojik arasında	500 VAC	500 VAC
	Alan ile 24 VDC arasında	500 VAC	500 VAC
	24 VDC ile lojik arasında	500 VAC	500 VAC
Kanal- kanal arasında	120 VAC	120 VAC	
Doğruluk	Bakınız Termokupl seçim çizelgesi (Sayfa 923).	Bakınız Termokupl seçim çizelgesi (Sayfa 923).	
Tekrarlanabilirlik	±0.05% FS	±0.05% FS	
Ölçme prensibi	İntegral alma	İntegral alma	
Modül güncelleme zamanı	Bakınız gürültü azaltma seçim çizelgesi (Sayfa 923/924).	Bakınız gürültü azaltma seçim çizelgesi (Sayfa 924).	
Soğuk birleşim hatası	±1.5 °C	±1.5 °C	



## A.9 Termokupl ve RTD sinyal modülleri (SM'ler)

Model	SM 1231 AI 4 x 16 bit TC	SM 1231 AI 8 x 16 bit TC
Kablo uzunluğu (metre)	100 sensöre max. 100 metre	100 sensöre max. 100 metre
İletken direnci	100 $\Omega$ max.	100 $\Omega$ max.

## Çizelge A- 165 Diyagnostikler

Model	SM 1231 AI 4 x 16 bit TC	SM 1231 AI 8 x 16 bit TC
Taşma/alt taşma <sup>1</sup>	Evet	Evet
Kablo kopması(akım modu sadece) <sup>2</sup>	Evet	Evet
24 VDC düşük gerilim <sup>1</sup>	Evet	Evet

<sup>1</sup> Aşırı akım, alt taşma ve düşük gerilim diyagnostik alarm bilgileri, modül konfigürasyonunda alarmlar etkisizleştirilse bile, analog veri değerleri içinde rapor edilecektir.

<sup>2</sup> Kablo kopukluğu alarmı etkisiz duruma getirildiğinde ve sensör bağlantısında açık kablo şartı oluştuğunda, modül rastgele değerler rapor edebilir.

SM 1231 Termokupl (TC) analog sinyal modülü modül girişlerine bağlanan gerilim değerini ölçer. Sıcaklık ölçme tipi "Termokupl" veya "Gerilim" olabilir.

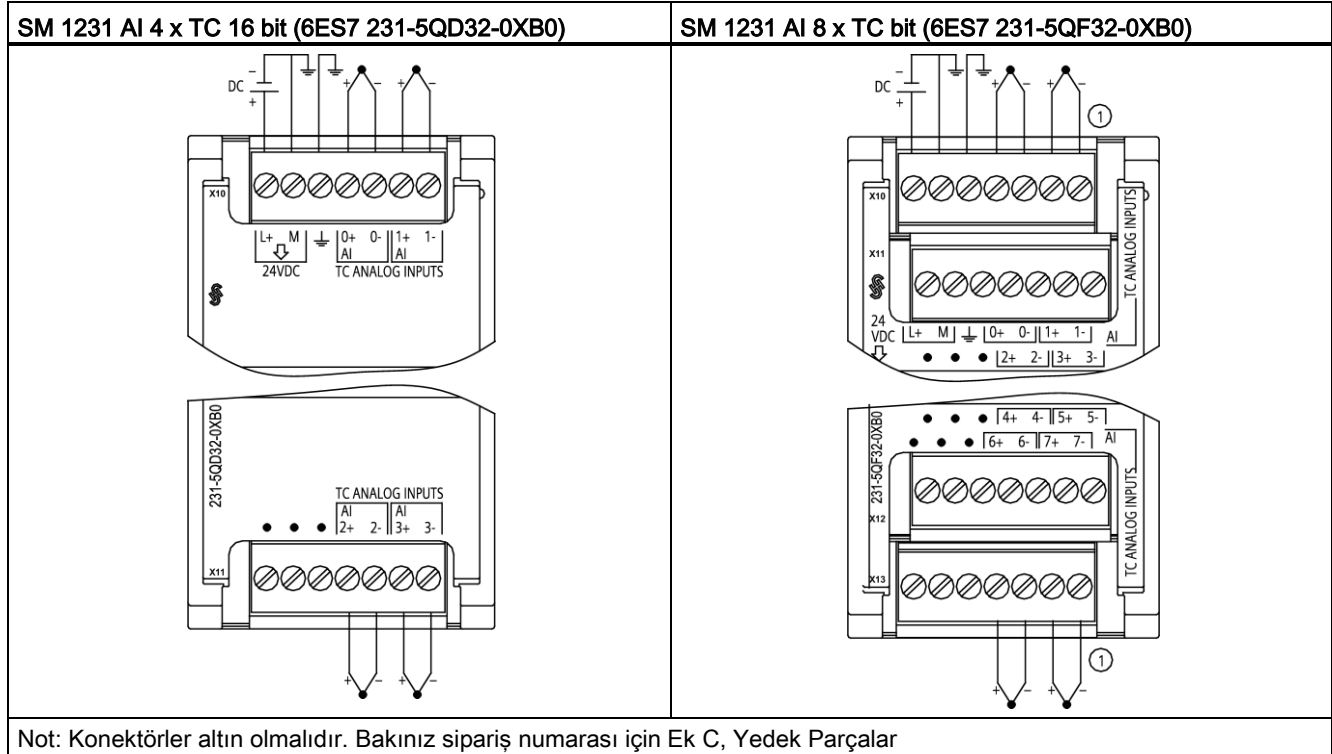
- "Termokupl": Değer, derecenin 10 ile çarpılmasıyla rapor edilecektir (örneğin, 25.3 derece decimal 253 olarak rapor edilecektir).

- "Gerilim": Nominal aralık tam ölçek değeri, decimal 27648 olacaktır.

Teknik özellikler

A.9 Termokupl ve RTD sinyal modülleri (SM'ler)

Çizelge A- 166 Termokupl SM'ler için bağlantı diyagramları



① TC 2, 3, 4 ve 5 açıklık için bağlı gösterilmemiştir.

Çizelge A- 167 SM 1231 AI 4 x TC 16 bit (6ES7 231-5QD32-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)
1	L+ / 24 VDC	Bağlantı yok
2	M / 24 VDC	Bağlantı yok
3	GND	Bağlantı yok
4	AI 0+ /TC	AI 2+ /TC
5	AI 0- /TC	AI 2- /TC
6	AI 1+ /TC	AI 3+ /TC
7	AI 1- /TC	AI 3- /TC

Çizelge A- 168 SM 1231 AI 8 x TC bit (6ES7 231-5QF32-0XB0) için konektör pin yerleşimleri

Pin	X10 (altın)	X11 (altın)	X12 (altın)	X13 (altın)
1	L+ / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	GND	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0+ /TC	AI 2+ /TC	AI 4 I- /TC	AI 6 I- /TC
5	AI 0- /TC	AI 2- /TC	AI 4 I+ /TC	AI 6 I+ /TC
6	AI 1+ /TC	AI 3+ /TC	AI 5 M- /TC	AI 7 M- /TC
7	AI 1- /TC	AI 3- /TC	AI 5 M+ /TC	AI 7 M+ /TC

### Not

Kullanılmayan analog girişler topraklanmalıdır.

Termokuplun kullanılmayan kanalları yeniden aktive edilebilir. Kullanılmayan bir kanal yeniden aktif duruma getirilirse, hata oluşmayacaktır.

#### A.9.1.1 Bir Termokuplun temel çalışması

Termokupllar, farklı iki metalin birbirine elektriksel olarak yapışık (bağlı) olduğunda oluşurlar. Birleşim sıcaklığına orantılı bir gerilim üretilir. Bu gerilim küçüktür; 1 mikro volt gerilim yüksek dereceleri temsil eder. Bir Termokupldan gerilim ölçmek, ilave birleşimler için temelini oluşturur.

Bir Termokuplu SM 1231 Termokupl modülüne bağladığınızda, iki birbirine benzemez metal modüle modül sinyal konektöründe bağlanır. Sensör Termokuplu şekillendirmek için birbirine benzemez iki metal birleştirilir.

Birbirine benzemeyen iki iletken (tel) sinyal konektörüne bağlandığında iki veya daha fazla Termokupl biçimlendirilir. Konektör sıcaklığı, sensör Termokuplundan üretilen gerilime eklenen bir gerilim dengelemek ve sonra sonuçların doğrusallaştırılması Termokupl kullanarak sıcaklık ölçmenin üretir. Bu gerilim için düzeltme yapılmazsa, ölçülen sıcaklık sensör sıcaklığından sapacaktır.

Soğuk birleşim dengelemesi konektör Termokuplda dengeleme için kullanılır. Termokupl çizelgesi referans birleşim sıcaklığı (genellikle sıfır derece Celsius) esasına göredir. Soğuk birleşim dengelemesi konektörü sıfır derece Celsius'a dengeler. Soğuk birleşim dengelemesi, konektör Termokuplları tarafından eklenen gerilimi geriye çıkarır.

Modülün sıcaklığı dahili olarak ölçülür ve sonra sensör dönüştürmesine eklenmek üzere bir değere dönüştürülür. Doğrultulan sensör dönüştürmesi sonra Termokupl çizelgesi kullanılarak doğrusallaştırılır.

Soğuk birleşim dengelemesinin optimum çalışması için Termokupl modülü termal olarak kararlı ortama yerleştirilmelidir. Ortam sıcaklığındaki yavaş değişim (0.1 °C/dakikadan az) modül özellikleri içerisinde doğru olarak dengelenir. Modül uçlarındaki hava akımı, soğuk birleşim dengeleme hatasına da yol açacaktır.

Daha iyi bir soğuk birleşim hata dengelemesi gerekirse, harici bir iso-ısı terminal bloğu kullanılabilir. Termokupl modülü 0 °C referanslı veya 50 °C referanslı terminal bloğu sağlar.

### A.9.1.2 SM 1231 Termokupl için seçim Çizelgesi

SM 1231 Termokupl sinyal modülü tarafından desteklenen farklı Termokupl tipleri için aralıklar ve doğruluk aşağıdaki Çizelgede gösterilmiştir.

Çizelge A- 169 SM 1231 Termokupl seçim Çizelgesi

Tip	Alt-aralık minimum <sup>1</sup>	Nominal aralık düşük limit	Nominal aralık yüksek limit	Üst-aralık maksimum <sup>2</sup>	Normal aralık <sup>3,4</sup> doğruluk @ 25 °C	Normal aralık <sup>1,2</sup> doğruluk -20 °C ila 60 °C
J	-210.0 °C	-150.0 °C	1200.0 °C	1450.0 °C	±0.3 °C	±0.6 °C
K	-270.0 °C	-200.0 °C	1372.0 °C	1622.0 °C	±0.4 °C	±1.0 °C
T	-270.0 °C	-200.0 °C	400.0 °C	540.0 °C	±0.5 °C	±1.0 °C
E	-270.0 °C	-200.0 °C	1000.0 °C	1200.0 °C	±0.3 °C	±0.6 °C
R & S	-50.0 °C	100.0 °C	1768.0 °C	2019.0 °C	±1.0 °C	±2.5 °C
B	0.0 °C	200.0 °C	800.0 °C	--	±2.0 °C	±2.5 °C
	--	800.0 °C	1820.0 °C	1820.0 °C	±1.0 °C	±2.3 °C
N	-270.0 °C	-200.0 °C	1300.0 °C	1550.0 °C	±1.0 °C	±1.6 °C
C	0.0 °C	100.0 °C	2315.0 °C	2500.0 °C	±0.7 °C	±2.7 °C
TXK/XK(L)	-200.0 °C	-150.0 °C	800.0 °C	1050.0 °C	±0.6 °C	±1.2 °C
Gerilim	-32512	-27648 -80mV	27648 80mV	32511	±0.05%	±0.1%

<sup>1</sup> Alt-aralık minimum değeri altındaki Termokupl değerleri -32768 olarak rapor edilir.

<sup>2</sup> Üst-aralık maksimum değeri üstündeki Termokupl değerleri 32767 olarak rapor edilir.

<sup>3</sup> Dahili soğuk birleşim hatası bütün aralıklar için ±1.5 °C. Çizelgedeki gibi hatalara bu durum eklenir. Bu özellikleri karşılamak üzere modülün en azından 30 dakika ısınma süresine ihtiyacı vardır. 970 MHz - 990 MHz ışınlı radyo frekansının bulunması, SM 1231 AI 4 x 16 bit TC doğruluğunu kötüleştirir.

#### Not

##### Termokupl kanalı

Termokupl sinyal modülü üstündeki her kanal bir diferansiyel tip olarak konfigüre edilebilir. (modül konfigürasyonu boyunca yazılım içerisinde seçilebilir).

Çizelge A- 170 SM 1231 Termokupl için gürültü azaltma ve güncelleme süreleri

Bastırma süresi seçimi	İntegrasyon süresi	4 kanallı modül güncelleme zamanı (saniye)	8 kanallı modül güncelleme zamanı (saniye)
400 Hz (2.5 ms)	10 ms <sup>1</sup>	0.143	0.285
60 Hz (16.6 ms)	16.67 ms	0.223	0.445
50 Hz (20 ms)	20 ms	0.263	0.525
10 Hz (100 ms)	100 ms	1.225	2.450

<sup>1</sup> 400 Hz bastırma frekansı seçildiğinde, modül çözünürlüğü ve doğruluğunu korumak için integrasyon süresi 10 ms seçilir. Bu seçim 100Hz ve 200 Hz gürültüyü de bastırır.

Termokuplların ölçmesinde 100 ms integrasyon süresi kullanılması tavsiye edilir. Daha küçük integrasyon süresi kullanımı sıcaklık okumalarının tekrarlanabilirlik hatasını artıracaktır.

### Not

Enerji uygulandıktan sonra, analog-dijital dönüştürücü için modül dahili kalibrasyonu yürütür. Bu sürede o kanal üzerinde geçerli bir veri mevcut oluncaya kadar modül her kanal için 32767 değerini rapor eder. Kullanıcı programınız bu başlangıç ayarı süresini sağlamaya gerek duyabilir. Bu modülün konfigürasyonu başlangıç ayarı süresinin uzunluğunu değiştirebileceğinden konfigürasyonunuzda bu modülün davranışını doğrulamanız gerekir. Gerekirse, modülün bu başlangıç ayarı süresini belirlemek için programınıza bir lojik dahil edebilirsiniz.

## Termokupl tipi J için analog değerlerin gösterilişi

Termokupl tipi J için analog değerlerin bir gösterilişi aşağıdaki Çizelgede verilmiştir.

Çizelge A- 171 Termokupl tipi J için analog değerlerin bir gösterilişi

Tip J, °C	Birimler		Tip J, °F	Birimler		Aralık
	Decimal	Hexadecimal		Decimal	Hexadecimal	
> 1450.0	32767	7FFF	> 2642.0	32767	7FFF	Taşma
1450.0	14500	38A4	2642.0	26420	6734	Aralık üstü
:	:	:	:	:	:	
1200.1	12001	2EE1	2192.2	21922	55A2	Anma aralığı
1200.0	12000	2EE0	2192.0	21920	55A0	
:	:	:	:	:	:	Alt taşma <sup>1</sup>
-150.0	-1500	FA24	-238.0	-2380	F6B4	
< -150.0	-32768	8000	< -238.0	-32768	8000	

<sup>1</sup> Hatalı kablay (örneğin, polarite ters çevrilmesi veya açık devre girişler) veya negatif aralıktaki sensör hatası (örneğin, Termokuplun yanlış tipte olması) Termokupl modül alt taşma sinyal üretmesine neden olabilir.

## Teknik özellikler

### A.9 Termokupl ve RTD sinyal modülleri (SM'ler)

#### A.9.2 SM 1231 RTD

#### SM 1231 RTD özellikleri

##### Çizelge A- 172 Genel özellikler

Teknik veri	SM 1231 AI 4 x RTD x 16 bit	SM 1231 AI 8 x RTD x 16 bit
Sipariş numarası	6ES7 231-5PD32-0XB0	6ES7 231-5PF32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75	70 x 100 x 75
Ağırlık	220 gram	270 gram
Güç kaybı	1.5 W	1.5 W
Çektiği akım(SM Bus)	80 mA	90 mA
Çektiği akım(24 VDC) <sup>1</sup>	40 mA	40 mA

<sup>1</sup> 20.4 - 28.8 VDC (Sınıf 2, Sınırlı güç veya PLC'den sensör gücü)

##### Çizelge A- 173 Analog girişler

Teknik veri	SM 1231 AI 4 x RTD x 16 bit	SM 1231 AI 8 x RTD x16 bit	
Giriş sayısı	4	8	
Tip	Modül referanslı RTD ve $\Omega$	Modül referanslı RTD ve $\Omega$	
Aralık	Bakınız RTD sensor seçim Çizelgesi (Sayfa 928).	Bakınız RTD sensor seçim Çizelgesi (Sayfa 928).	
Nominal aralık (veri word)			
Aşma/alt aşma aralığı (veri word)			
Taşma/alt taşma (veri word)			
Çözünürlük	Sıcaklık	0.1 °C/0.1 °F	0.1 °C/0.1 °F
	Direnç	15 bit artı işareti	15 bit artı işareti
Maksimum dayanma gerilimi	$\pm 35$ V	$\pm 35$ V	
Gürültü bastırma	85 dB, seçili gürültü azaltma için (10 Hz, 50 Hz, 60 Hz veya 400 Hz)	85 dB, seçili gürültü azaltma için (10 Hz, 50 Hz, 60 Hz veya 400 Hz)	
Ortak mod bastırma	> 120dB	> 120dB	
Empedans	$\geq 10$ M $\Omega$	$\geq 10$ M $\Omega$	
İzolasyon	Alan tarafı ile lojik	500 VAC	500 VAC
	Alan ile 2 VDC arasında	500 VAC	500 VAC
	24 VDC ile lojik arasında	500 VAC	500 VAC
Kanal-kanal arasında izolasyon	hiçbiri	hiçbiri	
Doğruluk	Bakınız RTD sensor seçim Çizelgesi (Sayfa 929).	Bakınız RTD sensor seçim Çizelgesi (Sayfa 929).	
Tekrarlanabilirlik	$\pm 0.05\%$ FS	$\pm 0.05\%$ FS	
Maksimum sensor enerji kaybı	0.5 m W	0.5 m W	
Ölçme prensibi	İntegralini alma	İntegralini alma	

## A.9 Termokupl ve RTD sinyal modülleri (SM'ler)

Teknik veri	SM 1231 AI 4 x RTD x 16 bit	SM 1231 AI 8 x RTD x16 bit
Modül güncelleme zamanı	Bakınız gürültü azaltma seçim Çizelgesi (Sayfa 929).	Bakınız gürültü azaltma seçim Çizelgesi (Sayfa 928).
Kablo uzunluğu (metre)	100 sensöre max. 100 metre	100 sensöre max. 100 metre
İletken direnci	20 Ω, 2.7 Ω için 10 Ω RTD max.	20 Ω, 2.7 Ω için 10 Ω RTD max.

## Çizelge A- 174 Diyagnostikler

Teknik veri	SM 1231 AI 4 x RTD x 16 bit	SM 1231 AI 8 x RTD x16 bit
Taşma/alt taşma <sup>1,2</sup>	Evet	Evet
Kablo kopması <sup>3</sup>	Evet	Evet
24 VDC düşük gerilim <sup>1</sup>	Evet	Evet

- <sup>1</sup> Aşırı akım, alt taşma ve düşük gerilim diyagnostik alarm bilgileri, modül konfigürasyonunda alarmlar etkisizleştirilse bile, analog veri değerleri içinde rapor edilecektir.
- <sup>2</sup> Direnç aralıkları için alt taşma algılaması asla etkin (enable) yapılmaz.
- <sup>3</sup> Kablo (tel) kopukluğu alarmı etkisizleştirildiğinde ve sensör kablağında açık devre şartı oluştuğunda modül rastgele değerler rapor edebilir.

SM 1231 RTD analog sinyal modülü modül girişlerine bağlanan direnç değerini ölçer. Ölçme tipi "Rezistör" veya "Termal Rezistör" olabilir.

- Rezistör": Nominal aralık tam ölçek değeri, decimal 27648 olacaktır.

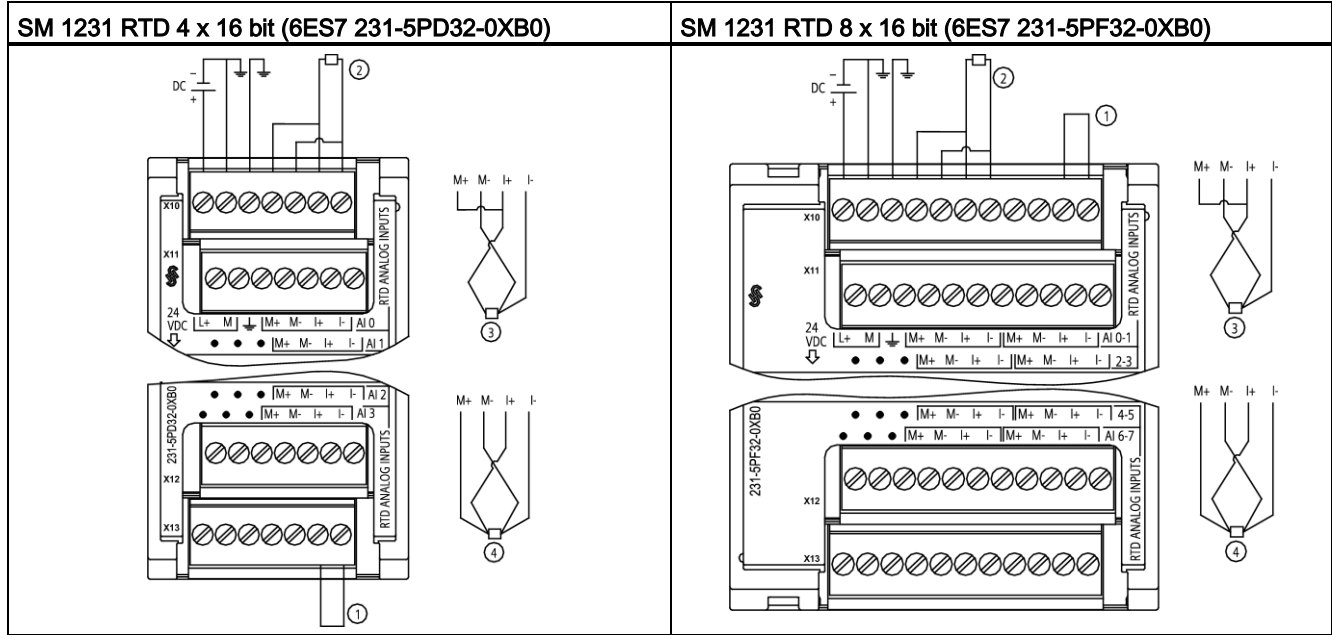
"Termal Rezistör": Değer, derecenin 10 ile çarpılmasıyla rapor edilecektir (örneğin, 25.3 derece decimal 253 olarak rapor edilecektir). Klimatik aralık değerleri derecenin 100 ile çarpılmasıyla rapor edilecektir (örneğin, 25.34 derece decimal 2534 olarak rapor edilecektir)

The SM 1231 RTD modülü, sensör dirençlerine 2-telli, 3- telli and 4- telli bağlantılarla yapılan ölçmeleri destekler.

## Teknik özellikler

### A.9 Termokupl ve RTD sinyal modülleri (SM'ler)

Çizelge A- 175 RTD SM'ler için bağlantı diyagramları



① Geri döngü kullanılmayan RTD girişleri

② 2-telli RTD ③ 3-telli RTD ④ 4-telli RTD

NOT: Konektörler altın olmalıdır. Bakınız sipariş numarası için Ek C, Yedek Parçalar

Çizelge A- 176 SM 1231 RTD 4 x 16 bit (6ES7 231-5PD32-0XB0) için konektör Pin yerleri

Pin	X10 (altın)	X11 (altın)	X12 (altın)	X13 (altın)
1	L+ / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	GND	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0 M+ /RTD	AI 1 M+ /RTD	AI 2 M+ /RTD	AI 3 M+ /RTD
5	AI 0 M- /RTD	AI 1 M- /RTD	AI 2 M- /RTD	AI 3 M- /RTD
6	AI 0 I+ /RTD	AI 1 I+ /RTD	AI 2 I+ /RTD	AI 3 I+ /RTD
7	AI 0 I- /RTD	AI 1 I- /RTD	AI 2 I- /RTD	AI 3 I- /RTD

Çizelge A- 177 SM 1231 RTD 8 x 16 bit (6ES7 231-5PF32-0XB0) için konvektör Pin yerleri

Pin	X10 (altın)	X11 (altın)	X12 (altın)	X13 (altın)
1	L+ / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	GND	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	AI 0 M+ /RTD	AI 2 M+ /RTD	AI 4 M+ /RTD	AI 6 M+ /RTD
5	AI 0 M- /RTD	AI 2 M- /RTD	AI 4 M- /RTD	AI 6 M- /RTD
6	AI 0 I+ /RTD	AI 2 I+ /RTD	AI 4 I+ /RTD	AI 6 I+ /RTD
7	AI 0 I- /RTD	AI 2 I- /RTD	AI 4 I- /RTD	AI 6 I- /RTD



Pin	X10 (altın)	X11 (altın)	X12 (altın)	X13 (altın)
8	AI 1 M+ /RTD	AI 3 M+ /RTD	AI 5 M+ /RTD	AI 7 M+ /RTD
9	AI 1 M- /RTD	AI 3 M- /RTD	AI 5 M- /RTD	AI 7 M- /RTD
10	AI 1 I+ /RTD	AI 3 I+ /RTD	AI 5 I+ /RTD	AI 7 I+ /RTD
11	AI 1 I- /RTD	AI 3 I- /RTD	AI 5 I- /RTD	AI 7 I- /RTD

**Not**

RTD kullanılmayan kanalları yeniden aktive edilebilir. Kullanılmayan bir kanal yeniden aktif duruma getirilirse, hata oluşmayacaktır.

RTD modülü, aktif yapılmayan kullanılmayan bir kanala otomatik olarak eklenen ilave stabilizasyon süresini yok etmek için geçerli döngünün devamına gerek duyar. Uygunluk için RTD modülü bir direnç bağlantısına ( 2-telli RTD bağlantısına benzer) gerek duyar.

**A.9.2.1 SM 1231 RTD için seçim Çizelgeleri**

Çizelge A- 178 RTD modüllerce desteklenen farklı sensörler için aralıklar ve doğruluk

Sıcaklık katsayısı	RTD tip	Alt aralık min. <sup>1</sup>	Nominal aralık düşük limit	Nominal aralık yüksek limit	Üst aralık max. <sup>2</sup>	Normal aralık doğruluk @ 25 °C	Normal aralık doğruluk -20 °C ila 60 °C
Pt 0.003850 ITS90 DIN EN 60751	Pt 100 klimatik	-145.00 °C	-120.00 °C	145.00 °C	155.00 °C	±0.20 °C	±0.40 °C
	Pt 10	-243.0 °C	-200.0 °C	850.0 °C	1000.0 °C	±1.0 °C	±2.0 °C
	Pt 50	-243.0 °C	-200.0 °C	850.0 °C	1000.0 °C	±0.5 °C	±1.0 °C
	Pt 100						
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0.003902 Pt 0.003916 Pt 0.003920	Pt 100	-243.0 °C	-200.0 °C	850.0 °C	1000.0 °C	± 0.5 °C	±1.0 °C
	Pt 200	-243.0 °C	-200.0 °C	850.0 °C	1000.0 °C	± 0.5 °C	±1.0 °C
	Pt 500						
	Pt 1000						
Pt 0.003910	Pt 10	-273.2 °C	-240.0 °C	1100.0 °C	1295 °C	±1.0 °C	±2.0 °C
	Pt 50	-273.2 °C	-240.0 °C	1100.0 °C	1295 °C	±0.8 °C	±1.6 °C
	Pt 100						
	Pt 500						
Ni 0.006720 Ni 0.006180	Ni 100	-105.0 °C	-60.0 °C	250.0 °C	295.0 °C	±0.5 °C	±1.0 °C
	Ni 120						
	Ni 200						
	Ni 500						
	Ni 1000						

## Teknik özellikler

## A.9 Termokupl ve RTD sinyal modülleri (SM'ler)

Sıcaklık katsayısı	RTD tip	Alt aralık min. <sup>1</sup>	Nominal aralık düşük limit	Nominal aralık yüksek limit	Üst aralık max. <sup>2</sup>	Normal aralık doğruluk @ 25 °C	Normal aralık doğruluk -20 °C ila 60 °C
LG-Ni 0.005000	LG-Ni 1000	-105.0 °C	-60.0 °C	250.0 °C	295.0 °C	±0.5 °C	±1.0 °C
Ni 0.006170	Ni 100	-105.0 °C	-60.0 °C	180.0 °C	212.4 °C	±0.5 °C	±1.0 °C
Cu 0.004270	Cu 10	-240.0 °C	-200.0 °C	260.0 °C	312.0 °C	±1.0 °C	±2.0 °C
Cu 0.004260	Cu 10	-60.0 °C	-50.0 °C	200.0 °C	240.0 °C	±1.0 °C	±2.0 °C
	Cu 50	-60.0 °C	-50.0 °C	200.0 °C	240.0 °C	±0.6 °C	±1.2 °C
	Cu 100						
Cu 0.004280	Cu 10	-240.0 °C	-200.0 °C	200.0 °C	240.0 °C	±1.0 °C	±2.0 °C
	Cu 50	-240.0 °C	-200.0 °C	200.0 °C	240.0 °C	±0.7 °C	±1.4 °C
	Cu 100						

<sup>1</sup> Alt-aralık minimum değeri altındaki RTD değerleri -32768 olarak rapor edilir.

<sup>2</sup> Üst-aralık maksimum değeri üstündeki RTD değerleri +32767 olarak rapor edilir.

## Çizelge A- 179 Direnç

Aralık	Alt aralık minimum	Nominal aralık düşük limit	Nominal aralık yüksek limit	Üst aralık maksimum <sup>1</sup>	Normal aralık doğruluk @ 25 °C	Normal aralık doğruluk -20 °C ila 60 °C
150 Ω	n/a	0 (0 Ω)	27648 (150 Ω)	176.383 Ω	±0.05%	±0.1%
300 Ω	n/a	0 (0 Ω)	27648 (300 Ω)	352.767 Ω	±0.05%	±0.1%
600 Ω	n/a	0 (0 Ω)	27648 (600 Ω)	705.534 Ω	±0.05%	±0.1%

<sup>1</sup> Üst-aralık minimum değeri üstündeki direnç değerleri +32768 olarak rapor edilir.

## Not

Aktif hale getirilen sensör bağlantısız bir kanal için modül 32767 raporunu verir. Açık-kablo algılama da etkinleştirilirse modül uygun LED'leri yakıp söndürür.

500 Ω ve 1000 Ω RTD aralıkları diğer düşük değerli dirençler ile kullanıldığında hata tanımlanan hatanın iki katına kadar yükselebilir.

10 Ω RTD aralığı için en iyi doğruluk 4-telli bağlantılar kullanılırsa elde edilecektir.

İki tel modda bağlantı tellerinin direnci sensör okumalarında hataya yol açar ve bundan dolayı doğruluk garanti edilemez.

Çizelge A- 180 RTD modülleri için gürültü azaltma ve güncelleme süresi

Bastırma süresi seçimi	İntegrasyon süresi	Güncelleme süresi (saniye)	
		4-kanal modül	8-kanal modül
400 Hz (2.5 ms)	10 ms <sup>1</sup>	4-/2-telli: 0.142 3-telli: 0.285	4-/2-telli: 0.285 3-telli: 0.525
60 Hz (16.6 ms)	16.67 ms	4-/2-telli: 0.222 3-telli: 0.445	4-/2-telli: 0.445 3-telli: 0.845
50 Hz (20 ms)	20 ms	4-/2-telli: 0.262 3-telli: .505	4-/2-telli: 0.524 3-telli: 1.015
10 Hz (100 ms)	100 ms	4-/2-telli: 1.222 3-telli: 2.445	4-/2-telli: 2.425 3-telli: 4.845

<sup>1</sup> 400 Hz bastırma frekansı seçildiğinde, modül çözünürlüğü ve doğruluğunu korumak için integrasyon süresi 10 ms seçilir. Bu seçim 100 Hz ve 200 Hz gürültüyü de bastırır.

### Not

Enerji uygulandıktan sonra, analog-dijital dönüştürücü için modül dahili kalibrasyonu yürütür. Bu sürede o kanal üzerinde geçerli bir veri mevcut oluncaya kadar modül her kanal için 32767 değerini rapor eder. Kullanıcı programınız bu başlangıç ayarı süresini sağlamaya gerek duyabilir. Bu modülün konfigürasyonu başlangıç ayarı süresinin uzunluğunu değiştirebileceğinden konfigürasyonunuzda bu modülün davranışını doğrulamanız gerekir. Gerekirse, modülün bu başlangıç ayarı süresini belirlemek için programınıza bir lojik dahil edebilirsiniz.

### RTD'ler için analog değerlerin gösterilişi

RTD standart sıcaklık aralık sensörleri için dijitalleştirilmiş ölçülen değerlerin bir gösterilişi aşağıdaki Çizelgede verilmiştir.

Çizelge A- 181 Direnç termometreler standart PT 100, 200, 500, 1000 ve PT 10, 50, 100, 500 GOST (0.003850) için değerlerin gösterilişi

Pt x00 °C biriminde standart (1 dijit = 0.1 °C)	Birimler		Pt x00 °F biriminde standart (1 dijit = 0.1 °F)	Birimler		Aralık
	Decimal	Hexadecimal		Decimal	Hexadecimal	
> 1000.0	32767	7FFF	> 1832.0	32767	7FFF	Taşma
1000.0	10000	2710	1832.0	18320	4790	Aralık dışı
:	:	:	:	:	:	
850.1	8501	2135	1562.1	15621	3D05	Anma aralığı
850.0	8500	2134	1562.0	15620	3D04	
:	:	:	:	:	:	
-200.0	-2000	F830	-328.0	-3280	F330	

## Teknik özellikler

### A.10 Teknoloji modülleri

Pt x00 °C biriminde standart (1 dijit = 0.1 °C)	Birimler		Pt x00 °F biriminde standart (1 dijit = 0.1 °F)	Birimler		Aralık
	Decimal	Hexadecimal		Decimal	Hexadecimal	
-200.1	-2001	F82F	-328.1	-3281	F32F	Aralık altı
:	:	:	:	:	:	
-243.0	-2430	F682	-405.4	-4054	F02A	
< -243.0	-32768	8000	< -405.4	-32768	8000	Alt taşma

## A.10 Teknoloji modülleri

### A.10.1 SM 1278 4xIO-Link Master SM

#### A.10.1.1 SM 1278 4xIO-Link Master sinyal modül özellikleri

Çizelge A- 182 Genel özellikler

Teknik veri	SM 1278 4xIO-Link Master sinyal modülü
Sipariş numarası	6ES7 278-4BD32-0XB0
Boyutlar W x H x D (mm)	45 x 100 x 75
Ağırlık	150 gram
Genel bilgi	
I&M veri	Evet; IM0 - IM3
Kaynak gerilimi	
Anma gerilimi (DC)	24 VDC
Geçerli aralık düşük sınır (DC)	19.2 V; 20.5 V, eğer IO-Link kullanılırsa (IMaster üzerindeki IO-Link cihazları için besleme kaynağı en azından 20 V)
Geçerli aralık yüksek sınır (DC)	28.8 VDC
Polarite ters çevirme koruması	Evet
Giriş akımı	
Akım tüketimi	65 mA; yüksüz
Enkoder beslemesi	
Giriş sayısı	4
Çıkış akımı, anma değer	200 mA
Güç kaybı	
Güç kaybı, tipik	1 W, port yükleme hariç
Dijital girişler/çıkışlar	
Kablo uzunluğu (metre)	20 m, zırhsız, max.
SDLC	

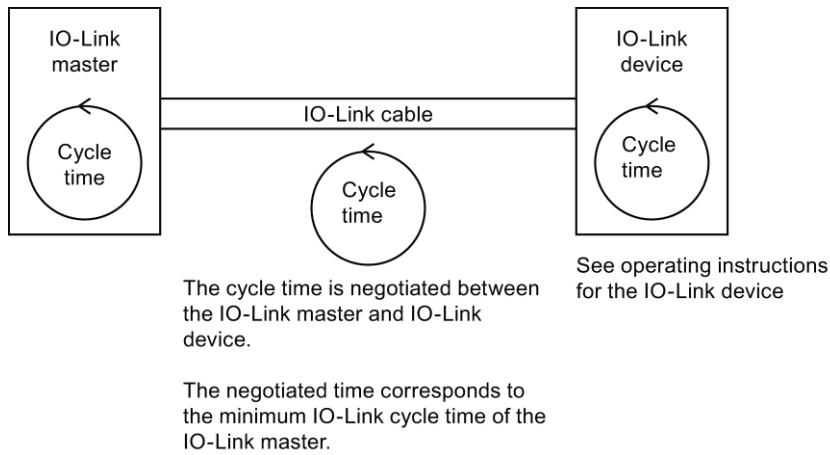
Teknik veri	SM 1278 4xIO-Link Master sinyal modülü
Kablo uzunluğu (metre)	20 m, zırhsız, max.
<b>IO-Link</b>	
Portların numarası	4
Aynı anda kontrol edilebilecek port sayısı	4
IO-Link protokol 1.0	Evet
IO-Link protokol 1.1	Evet
<b>Çalışma modu</b>	
IO-Link	Evet
DI	Evet
DQ	Evet; max. 100 mA
<b>I/O-Link cihazlarının bağlantısı</b>	
Port tipi A	Evet
Aktarım hızı	4.8 kBd (COM1) 38.4 kBd (COM2) 230.4 kBd (COM3)
Döngü süresi, min.	2 ms, dinamik, kullanıcı veri uzunluğuna bağlı
Proses verisi boyutu, port başına giriş	32 bayt; max.
Proses verisi boyutu, modül başına giriş	32 bayt
Proses verisi boyutu, port başına çıkış	32 bayt; max.
Proses verisi boyutu, modül başına çıkış	32 bayt
Cihaz parametreleri için bellek boyutu	2 Kbayt
Kablo uzunluğu zırhsız, max. (metre)	20 m
<b>Interrupt'lar/diyagnostikler/durum bilgisi</b>	
Durum gösterici	Evet
<b>Interrupt'lar</b>	
Diyagnostik interrupt	Evet; port diyagnostikler sadece IO-Link modda vardır
<b>Diyagnostik alarmlar</b>	
<b>Diyagnostikler</b>	
Kaynak geriliminin izlenmesi	Evet
Kısa devre	Evet
<b>Diyagnostik gösterici LED</b>	
Kaynak geriliminin izlenmesi	Evet; yanıp sönen LED DIAG LED
Kanal durum gösterici	Evet; kanal durumu Qn ve PORT durumu Cn (IO-Link modu) için kanal başına bir yeşil LED (SIO modu)
Kanal diyagnostikler için	Evet; kırmızı Fn LED
Modül diyagnostikler için	Evet; yeşil/kırmızı DIAG LED
<b>Elektriksel izolasyon</b>	
<b>Elektriksel izolasyon kanalları</b>	
Kanallar arasında	Hayır
Kanallar ve arka plan bus arasında	Evet
<b>İzin verilen potansiyel fark</b>	

## Teknik özellikler

### A.10 Teknoloji modülleri

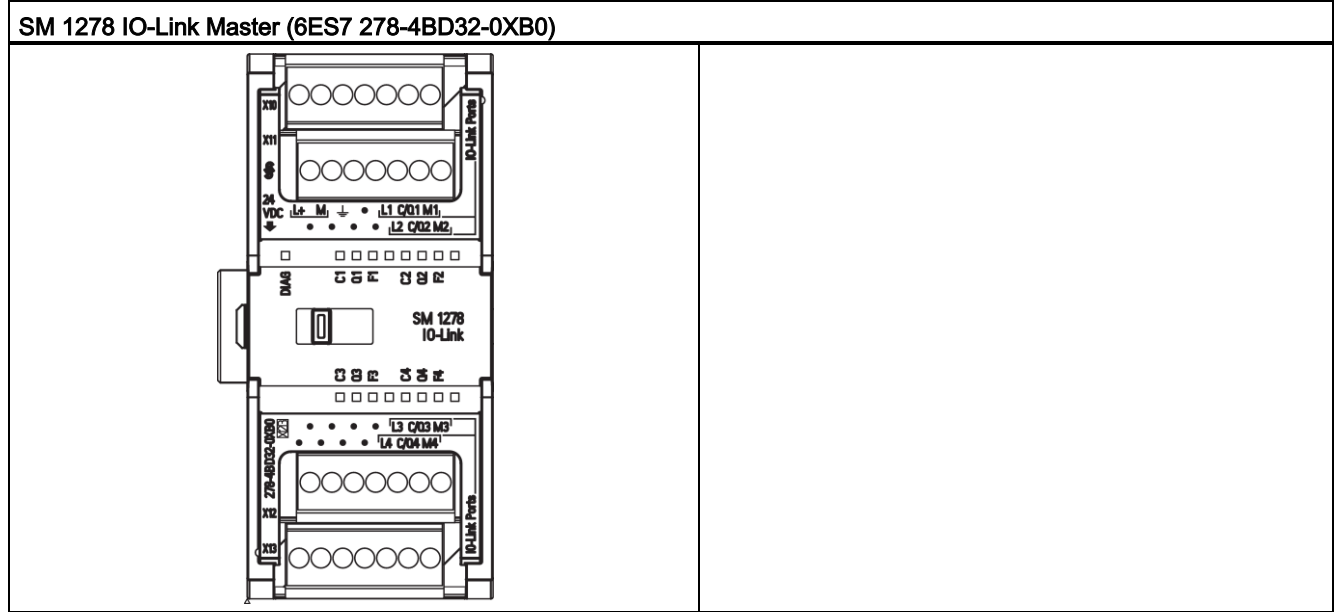
Teknik veri	SM 1278 4xIO-Link Master sinyal modülü
Farklı devreler arasında	75 VDC / 60 VAC (temel izolasyon)
Yalıtım	
Yalıtım denenir	707 VDC (tip deneyi)
Ortam koşulları	
Çalışma sıcaklığı	
Min.	-20 °C
Max.	60 °C
Yatay montaj, min.	-20 °C
Dikey montaj, max.	60 °C
Dikey montaj, min.	-20 °C
Dikey montaj, max.	50 °C

### Tepki süresinin genel değerlendirmesi



### A.10.1.2 SM 1278 4xIO-Link Master SM bağlantı diyagramları

Çizelgesi A- 183 SM 1278 IO-Link Master için bağlantı diyagramı



Çizelgesi A- 184 1278 IO-Link Master (6ES7 278-4BD32-0XB0) için konektör pin yerleşimleri

Pin	X10	X11	X12	X13
1	L+ / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
2	M / 24 VDC	Bağlantı yok	Bağlantı yok	Bağlantı yok
3	Fonksiyonel toprak	Bağlantı yok	Bağlantı yok	Bağlantı yok
4	Bağlantı yok	Bağlantı yok	Bağlantı yok	Bağlantı yok
5	L <sub>1</sub>	L <sub>2</sub>	L <sub>3</sub>	L <sub>4</sub>
6	C/Q <sub>1</sub>	C/QL <sub>2</sub>	C/Q <sub>3</sub>	C/QL <sub>4</sub>
7	ML <sub>1</sub>	ML <sub>2</sub>	M <sub>3</sub>	ML <sub>4</sub>

Teknik özellikler

A.11 Dijital sinyal kartları (SB'ler)

A.11 Dijital sinyal kartları (SB'ler)

A.11.1 SB 1221 200 kHz dijital giriş özellikleri

Çizelge A- 185 Genel özellikler

Teknik veri	SB 1221 DI 4 x 24 VDC, 200 kHz	SB 1221 DI 4 x 5 VDC, 200 kHz
Sipariş numarası	6ES7 221-3BD30-0XB0	6ES7 221-3AD30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 21	38 x 62 x 21
Ağırlık	35 gram	35 gram
Güç kaybı	1.5 W	1.0 W
Çektiği akım(SM Bus)	40 mA	40 mA
Çektiği akım(24 VDC)	7 mA / giriş + 20 mA	15 mA / giriş + 15 mA

Çizelge A- 186 Dijital girişler

Teknik veri	SB 1221 DI 4 x 24 VDC, 200 kHz	SB 1221 DI 4 x 5 VDC, 200 kHz
Giriş sayısı	4	4
Tip	Source	Source
Anma gerilimi	24 VDC 7 mA'de, nominal	5 VDC 15 mA'de, nominal
Sürekli izin verilebilir gerilim	28.8 VDC	6 VDC
Aşırı gerilim	35 VDC için 0.5 saniye	6 V
Lojik 1 sinyal (min.)	L+ eksi 10 VDC 2.9 mA'de	L+ eksi 2.0 VDC 5.1 mA'de
Lojik 0 sinyal (max.)	L+ eksi 5 VDC 1.4 mA'de	L+ eksi 1.0 VDC 2.2 mA'de
HSC saat giriş hızları (max.)	Tek faz: 200 kHz Dördün faz: 160 kHz	Tek faz: 200 kHz Dördün faz: 160 kHz
İzolasyon (alan tarafı ile lojik arasında)	500 VAC için 1 dakika	500 VAC için 1 dakika
İzolasyon grupları	1	1
Filtre süreleri	us ayarlar	0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0,12.8, 20.0
	ms ayarlar	0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0
Aynı anlı girişlerin sayısı	<ul style="list-style-type: none"> <li>2 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li> <li>4, 55 °C yatayda veya 45 °C dikey</li> </ul>	4
Kablo uzunluğu (metre)	50 zırlı bükülü çift	50 zırlı bükülü çift

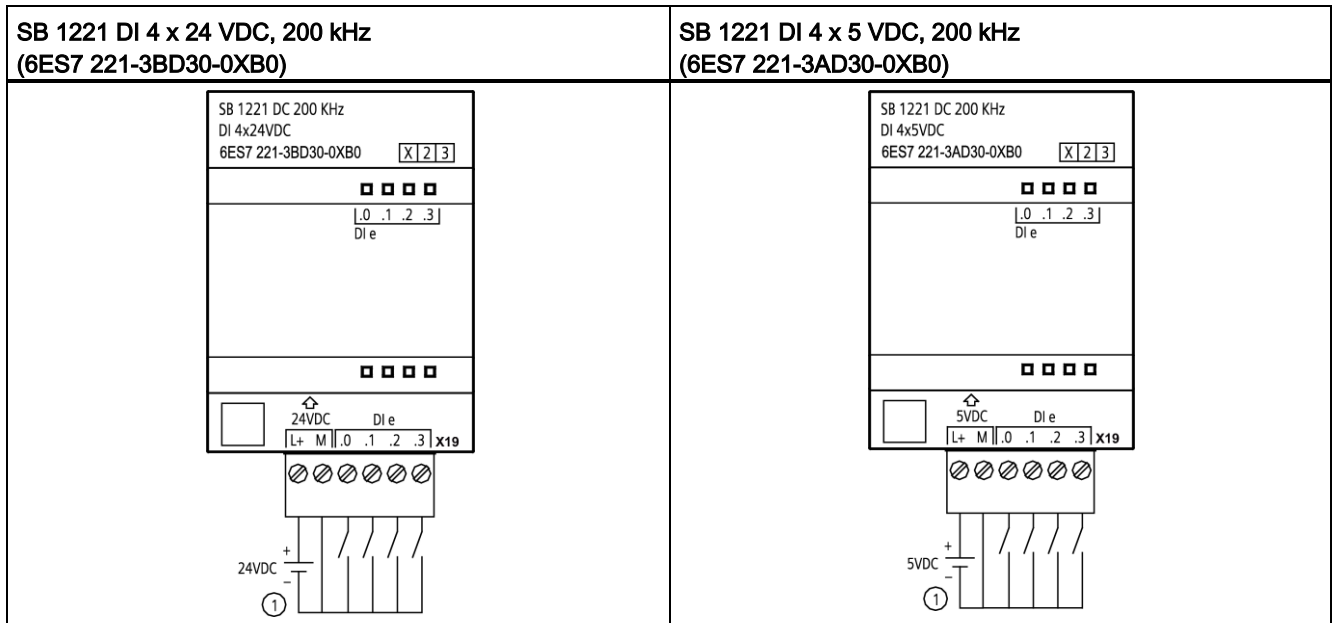


### Not

Anahtarlama frekansı 20 kHz üstünde olduğunda dijital girişler bir kare dalga alırlar. Bu girişlere sinyal kalitesini iyileştirmek için aşağıdaki opsiyonları dikkate alınız:

- Kablo uzunluğunu en kısa yapınız
- Bir sürücüyü sadece bir sink sürücüdən bir sinking ve sourcing sürücüyə deęiştiriniz
- Yüksek kaliteli bir kabloyla deęiştiriniz
- Devre/bileşenleri 24 V'dan 5 V'a düşürünüz
- Girişe bir harici yük ekleyiniz.

Çizelge A- 187 200 kHz dijital giriş SM'ler için bağlantı diyagramları



① Sadece sourcing girişleri destekler

Çizelge A- 188 SB 1221 DI 4 x 24 VDC, 200 kHz (6ES7 221-3BD30-0XB0) için konektör pin yerleşimleri

Pin	X19
1	L+ / 24 VDC
2	M / 24 VDC
3	DI e.0
4	DI e.1
5	DI e.2
6	DI e.3

## Teknik özellikler

### A.11 Dijital sinyal kartları (SB'ler)

Çizelge A- 189 SB 1221 DI 4 x 5 VDC, 200 kHz (6ES7 221-3AD30-0XB0) için konektör pin yerleşimleri

Pin	X19
1	L+ / 5 VDC
2	M / 5 VDC
3	DI e.0
4	DI e.1
5	DI e.2
6	DI e.3

### A.11.2 SB 1222 200 kHz dijital çıkış özellikleri

Çizelge A- 190 Genel özellikler

Teknik veri	SB 1222 DQ 4 x 24 VDC, 200 kHz	SB 1222 DQ 4 x 5 VDC, 200 kHz
Sipariş numarası	6ES7 222-1BD30-0XB0	6ES7 222-1AD30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 21	38 x 62 x 21
Ağırlık	35 gram	35 gram
Güç kaybı	0.5 W	0.5 W
Çektiği akım(SM Bus)	35 mA	35 mA
Çektiği akım(24 VDC)	15 mA	15 mA

Çizelge A- 191 Dijital çıkışlar

Teknik veri	SB 1222 DQ 4 x 24 VDC, 200 kHz	SB 1222 DQ 4 x 5 VDC, 200 kHz
Giriş sayısı	4	4
Çıkış tipi	Yarı iletken- MOSFET sink ve source <sup>1</sup>	Yarı iletken- MOSFET sink ve source <sup>1</sup>
Gerilim aralığı	20.4 - 28.8 VDC	4.25 - 6.0 VDC
Lojik 1 max. akımdaki sinyal	L+ eksi 1.5 V	L+ eksi 0.7 V
Lojik 0 max. akımdaki sinyal	1.0 VDC, max.	0.2 VDC, max.
Akım (max.)	0.1 A	0.1 A
Lamba yük	--	--
ON durumu temas direnci	11 Ω max.	7 Ω max.
Off durumu direnci	6 Ω max.	0.2 Ω max.
Nokta başına kaçak akım	--	--
Darbe katarı çıkış hızı	200 kHz max., 2 Hz min.	200 kHz max., 2 Hz min.
Ani aşırı akım	0.11 A	0.11 A
Aşırı yük koruması	Hayır	Hayır
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika	500 VAC için 1 dakika
İzolasyon grupları	1	1

Teknik veri	SB 1222 DQ 4 x 24 VDC, 200 kHz	SB 1222 DQ 4 x 5 VDC, 200 kHz
Ortak uç başına akımlar	0.4 A	0.4 A
İndüktif kenetleme gerilimi	Hiçbiri	Hiçbiri
Anahtarlama gecikmesi	1.5 $\mu$ s + 300 ns yükselme 1.5 $\mu$ s + 300 ns düşme	200 ns + 300 ns yükselme 200 ns + 300 ns düşme
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	<ul style="list-style-type: none"><li>2 (bitişik noktalar değil), 60 °C yatayda veya 50 °C dikey</li><li>4, 55 °C yatayda veya 45 °C dikey</li></ul>	4
Kablo uzunluğu (metre)	50 zırlı bükülü çift	50 zırlı bükülü çift

- <sup>1</sup> Hem sinking hem de sourcing konfigürasyonları aynı devre tarafından desteklendiği için sourcing yükün aktif durumu sinking yükünün tersidir. Bir source çıkışı pozitif lojik (yükten bir akım geçerken Q bit ve LED durumu ON olmaktadır) sergilerken bir sink çıkışı negatif lojik (yükten bir akım geçerken Q bit ve LED durumu OFF olmaktadır) sergiler. Kullanıcı programı olmadan modül takılırsa, bu modül için varsayılan 0 V'tur, bunun anlamı bir sinking yük ON durumuna geçecektir.

### Not

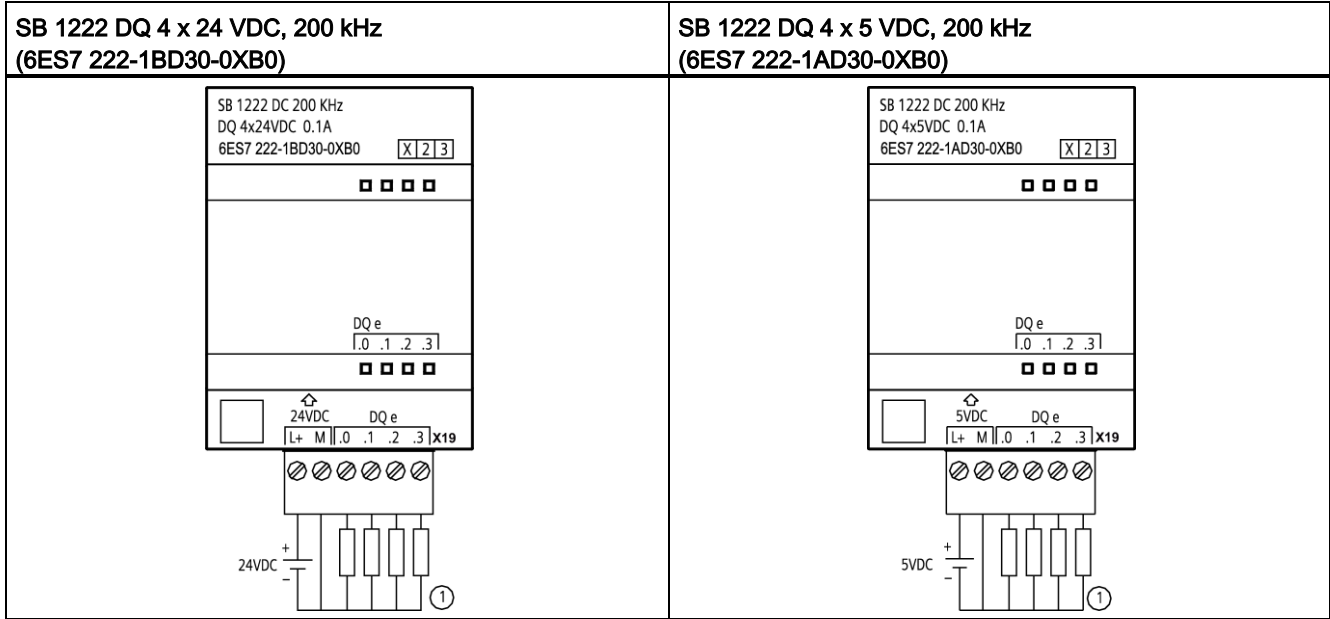
Anahtarlama frekansı 20 kHz üstünde olduğunda dijital girişler bir kare dalga alırlar. Bu girişlere sinyal kalitesini iyileştirmek için aşağıdaki opsiyonları dikkate alınız:

- Kablo uzunluğunu en kısa yapınız
- Bir sürücüyü sadece bir sink sürücünden bir sinking ve sourcing sürücüyeye değiştiriniz
- Yüksek kaliteli bir kabloyla değiştiriniz
- Devre/bileşenleri 24 V'dan 5 V'a düşürünüz
- Girişe bir harici yük ekleyiniz

## Teknik özellikler

### A.11 Dijital sinyal kartları (SB'ler)

Çizelge A- 192 200 kHz dijital çıkış SM'ler için bağlantı diyagramları



- ① Sourcing çıkışları için "Load" ucunu "-" ucuna bağlayınız (gösterilmiştir). Sinking çıkışları için "Load" ucunu "+" ucuna bağlayınız. Hem sinking hem de sourcing konfigürasyonları aynı devre tarafından desteklendiği için sourcing yükün aktif durumu sinking yükünün tersidir. Bir source çıkışı pozitif lojik (yükten bir akım geçerken Q bit ve LED durumu ON olmaktadır) sergilerken bir sink çıkışı negatif lojik (yükten bir akım geçerken Q bit ve LED durumu OFF olmaktadır) sergiler. Kullanıcı programı olmadan modül takılırsa, bu modül için varsayılan 0 V'tur, bunun anlamı bir sinking yük ON durumuna geçecektir.

Çizelge A- 193 SB 1222 DQ 4 x 24 VDC, 200 kHz (6ES7 222-1BD30-0XB0) için konektör pin yerleşimleri

Pin	X19
1	L+ / 24 VDC
2	M / 24 VDC
3	DQ e.0
4	DQ e.1
5	DQ e.2
6	DQ e.3

Çizelge A- 194 SB 1222 DQ 4 x 5 VDC, 200 kHz (6ES7 222-1AD30-0XB0) için konektör pin yerleşimleri

Pin	X19
1	L+ / 5 VDC
2	M / 5 VDC
3	DQ e.0
4	DQ e.1

Pin	X19
5	DQ e.2
6	DQ e.3

### A.11.3 SB 1223 200 kHz dijital giriş / çıkış özellikleri

Çizelge A- 195 Genel özellikler

Teknik veri	SB 1223 DI 2 x 24 VDC / DQ 2 x 24 VDC, 200 kHz	SB 1223 DI 2 x 5 VDC / DQ 2 x 5 VDC, 200 kHz
Sipariş numarası	6ES7 223-3BD30-0XB0	6ES7 223-3AD30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 21	38 x 62 x 21
Ağırlık	35 gram	35 gram
Güç kaybı	1.0 W	0.5 W
Çektiği akım(SM Bus)	35 mA	35 mA
Çektiği akım(24 VDC)	7 mA / giriş + 30 mA	15 mA / giriş + 15 mA

Çizelge A- 196 Dijital girişler

Teknik veri	SB 1223 DI 2 x 24 VDC / DQ 2 x 24 VDC, 200 kHz	SB 1223 DI 2 x 5 VDC / DQ 2 x 5 VDC, 200 kHz
Giriş sayısı	2	2
Tip	Source	Source
Anma gerilimi	24 VDC 7 mA'de, nominal	5 VDC 15 mA'de, nominal
Sürekli izin verilebilir gerilim	28.8 VDC	6 VDC
Aşırı gerilim	35 VDC için 0.5 saniye	6 V
Lojik 1 sinyal (min.)	L+ eksi 10 VDC 2.9 mA'de	L+ eksi 2.0 VDC 5.1 mA'de
Lojik 0 sinyal (max.)	L+ eksi 5 VDC 1.4 mA'de	L+ eksi 1.0 VDC 2.2 mA'de
HSC saat giriş hızları (max.)	Tek faz: 200 kHz Dördün faz: 160 kHz	Tek faz: 200 kHz Dördün faz: 160 kHz
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika	500 VAC için 1 dakika
İzolasyon grupları	1 (çıkışlar için izolasyon yok)	1 (çıkışlar için izolasyon yok)
Filtre süreleri	us ayarlar	0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0
	ms ayarlar	0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0
Aynı anlı girişlerin sayısı	2	2
Kablo uzunluğu (metre)	50 zırlı bükülü çift	50 zırlı bükülü çift

## Teknik özellikler

### A.11 Dijital sinyal kartları (SB'ler)

Çizelge A- 197 Dijital çıkışlar

Teknik veri	SB 1223 DI 2 x 24 VDC / DQ 2 x 24 VDC, 200 kHz	SB 1223 DI 2 x 5 VDC / DQ 2 x 5 VDC, 200 kHz
Giriş sayısı	2	2
Çıkış tipi	Yarı iletken- MOSFET sink ve source <sup>1</sup>	Yarı iletken- MOSFET sink ve source <sup>1</sup>
Gerilim aralığı	20.4 - 28.8 VDC	4.25 - 6.0 VDC
Anma değer	24 VDC	5 VDC
Lojik 1 max. akımdaki sinyal	L+ eksi 1.5 V	L+ eksi 0.7 V
Lojik 0 max. akımdaki sinyal	1.0 VDC, max.	0.2 VDC, max.
Akım (max.)	0.1 A	0.1 A
Lamba yük	--	--
ON durumu temas direnci	11 Ω max.	7 Ω max.
Off durumu direnci	6 Ω max.	0.2 Ω max.
Nokta başına kaçak akım	--	--
Darbe katarı çıkış hızı	200 kHz max., 2 Hz min.	200 kHz max., 2 Hz min.
Ani aşırı akım	0.11 A	0.11 A
Aşırı yük koruması	Hayır	Hayır
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika	500 VAC için 1 dakika
İzolasyon grupları	1 (girişler için izolasyon yok)	1 (girişler için izolasyon yok)
Ortak uç başına akımlar	0.2 A	0.2 A
İndüktif kenetleme gerilimi	Hiçbiri	Hiçbiri
Anahtarlama gecikmesi	1.5 µs + 300 ns yükselme 1.5 µs + 300 ns düşme	200 ns + 300 ns yükselme 200 ns + 300 ns düşme
RUN'dan STOP'a davranış	Son değer veya değişiklik (varsayılan değer 0)	Son değer veya değişiklik (varsayılan değer 0)
Aynı andaki çıkış sayısı	2	2
Kablo uzunluğu (metre)	50 zırlı bükülü çift	50 zırlı bükülü çift

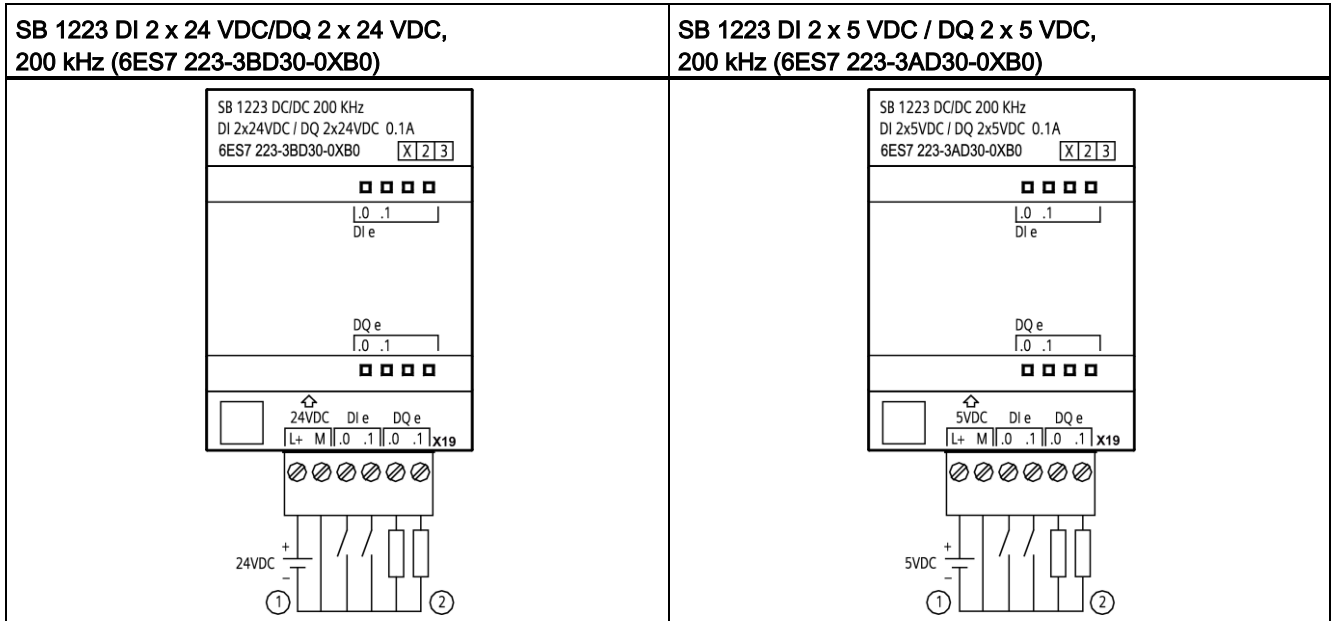
<sup>1</sup> Hem sinking hem de sourcing konfigürasyonları aynı devre tarafından desteklendiği için sourcing yükün aktif durumu sinking yükünün tersidir. Bir source çıkışı pozitif lojik (yükten bir akım geçerken Q bit ve LED durumu ON olmaktadır) sergilerken bir sink çıkışı negatif lojik (yükten bir akım geçerken Q bit ve LED durumu OFF olmaktadır) sergiler. Kullanıcı programı olmadan modül takılırsa, bu modül için varsayılan 0 V'tur, bunun anlamı bir sinking yük ON durumuna geçecektir.

#### Not

Anahtarlama frekansı 20 kHz üstünde olduğunda dijital girişler bir kare dalga alırlar. Bu girişlere sinyal kalitesini iyileştirmek için aşağıdaki opsiyonları dikkate alınız:

- Kablo uzunluğunu en kısa yapınız
- Bir sürücüyü sadece bir sink sürücünden bir sinking ve sourcing sürücüyeye değiştiriniz
- Yüksek kaliteli bir kabloyla değiştiriniz
- Devre/bileşenleri 24 V'dan 5 V'a düşürünüz
- Girişe bir harici yük ekleyiniz.

Çizelge A- 198 200 kHz dijital giriş/çıkış SM'ler için bağlantı diyagramları



① Sadece sourcing girişleri destekler.

② Sourcing çıkışları için "Load" ucunu "-" ucuna bağlayınız (gösterilmiştir). Sinking çıkışları için "Load" ucunu "+" ucuna bağlayınız. Hem sinking hem de sourcing konfigürasyonları aynı devre tarafından desteklendiği için sourcing yükün aktif durumu sinking yükünün tersidir. Bir source çıkışı pozitif lojik (yükten bir akım geçerken Q bit ve LED durumu ON olmaktadır) sergilerken bir sink çıkışı negatif lojik (yükten bir akım geçerken Q bit ve LED durumu OFF olmaktadır) sergiler. Kullanıcı programı olmadan modül takılırsa, bu modül için varsayılan 0 V'tur, bunun anlamı bir sinking yük ON durumuna geçecektir.

Çizelge A- 199 SB 1223 DI 2 x 24 VDC/DQ 2 x 24 VDC, 200 kHz (6ES7 223-3BD30-0XB0) için konektör pin yerleşimleri

Pin	X19
1	L+ / 24 VDC
2	M / 24 VDC
3	DI e.0
4	DI e.1
5	DQ e.0
6	DQ e.1

## Teknik özellikler

### A.11 Dijital sinyal kartları (SB'ler)

Çizelge A- 200 SB 1223 DI 2 x 5 VDC / DQ 2 x 5 VDC, 200 kHz (6ES7 223-3AD30-0XB0) için konektör pin yerleşimleri

Pin	X19
1	L+ / 5 VDC
2	M / 5 VDC
3	DI e.0
4	DI e.1
5	DQ e.0
6	DQ e.1

### A.11.4 SB 1223 2 X 24 VDC giriş / 2 X 24 VDC çıkış özellikleri

Çizelge A- 201 Genel özellikler

Teknik veri	SB 1223 DI 2 x 24 VDC, DQ 2 x 24 VDC
Sipariş numarası	6ES7 223-0BD30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 21
Ağırlık	40 gram
Güç kaybı	1.0 W
Çektiği akım(SM Bus)	50 mA
Çektiği akım(24 VDC)	4 mA / Giriş kullanıldı

Çizelge A- 202 Dijital girişler

Teknik veri	SB 1223 DI 2 x 24 VDC, DQ 2 x 24 VDC
Giriş sayısı	2
Tip	IEC Tip 1 sink
Anma gerilimi	24 VDC 4 mA'de, nominal
Sürekli izin verilebilir gerilim	30 VDC, max.
Aşırı gerilim	35 VDC için 0.5 saniye
Lojik 1 sinyal (min.)	15 VDC 2.5 mA'de
Lojik 0 sinyal (max.)	5 VDC 1 mA'de
HSC saat giriş hızları (max.)	Tek faz: 30 kHz (15 - 26 VDC) Dördün faz: 20 kHz (15 - 26 VDC)
İzolasyon (alan tarafı ile lojik arasında)	500 VAC için 1 dakika
İzolasyon grupları	1
Filtre süreleri	us ayarlar 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4,10.0, 12.8, 20.0 ms ayarlar 0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2, 6.4, 10.0, 12.8, 20.0
Aynı anlı girişlerin sayısı	2
Kablo uzunluğu (metre)	500 zırlı, 300 zırsız



Çizelge A- 203 Dijital çıkışlar

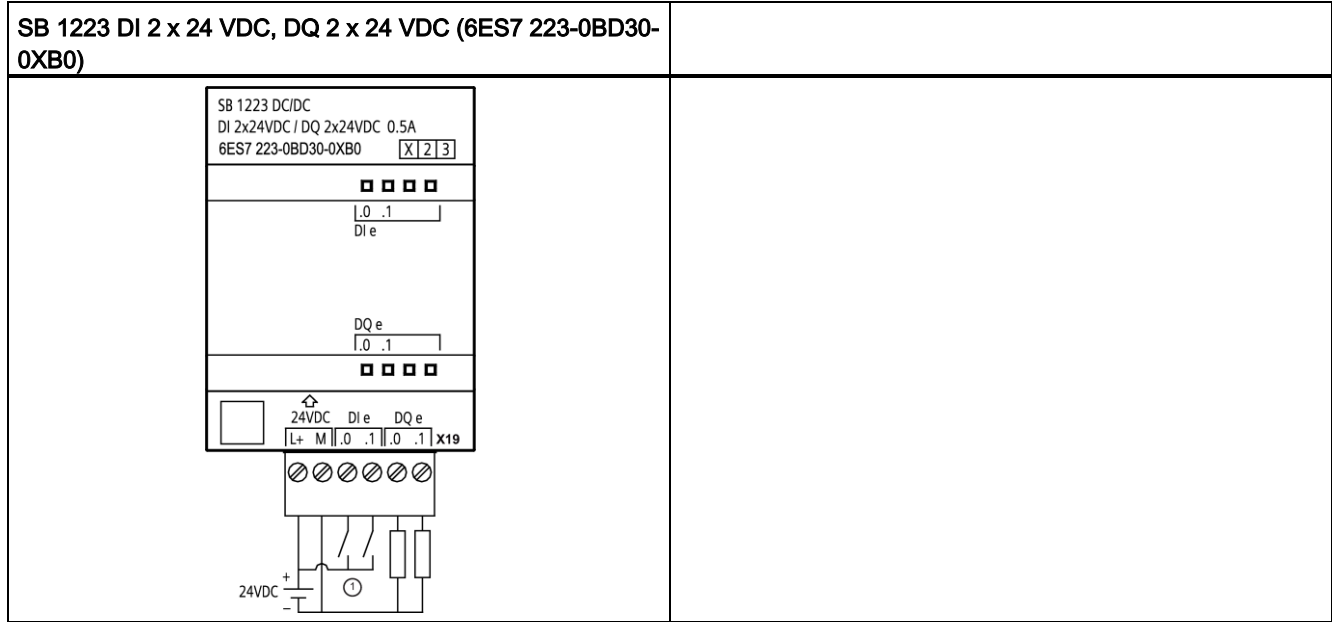
<b>Teknik veri</b>	<b>SB 1223 DI 2 x 24 VDC, DQ 2 x 24 VDC</b>
Giriş sayısı	2
Çıkış tipi	Yarı iletken- MOSFET (sourcing)
Gerilim aralığı	20.4 - 28.8 VDC
Lojik 1 max. akımdaki sinyal	20 VDC min.
Lojik 0 sinyal, 10K $\Omega$ yük ile	0.1 VDC max.
Akım (max.)	0.5 A
Lamba yük	5 W
ON durumu temas direnci	0.6 $\Omega$ max.
Nokta başına kaçak akım	10 $\mu$ A max.
Darbe katar çıkış (PTO) hızı	20 KHz max., 2 Hz min. <sup>1</sup>
Ani aşırı akım	5 A için 100 ms max.
Aşırı yük koruması	Hayır
İzolasyon (alan tarafı ile lojik)	500 VAC için 1 dakika
İzolasyon grupları	1
Ortak uç başına akımlar	1 A
İndüktif kenetleme gerilimi	L+ eksi 48 V, 1 W ısı olarak kayıp
Anahtarlama gecikmesi	2 $\mu$ s max. off - on geçişi 10 $\mu$ s max. on - off geçişi
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)
Aynı andaki çıkış sayısı	2
Kablo uzunluğu (metre)	500 m zırlı, 150 m zırlısız

<sup>1</sup> Darbe alıcı ve kablonuza bağlı olarak ilave bir yük direnci (anma akımının en az %10'nunda) darbe sinyal kalitesini ve gürültü bağışıklığını iyileştirebilir.

Teknik özellikler

A.11 Dijital sinyal kartları (SB'ler)

Çizelge A- 204 Analog giriş/çıkış SB'ler için bağlantı diyagramı



① Sinking girişlerini destekler sadece

Çizelge A- 205 SB 1223 DI 2 x 24 VDC, DQ 2 x 24 VDC (6ES7 223-0BD30-0XB0) için konektör pin yerleşimleri

Pin	X19
1	L+ / 24 VDC
2	M / 24 VDC
3	DI e.0
4	DI e.1
5	DQ e.0
6	DQ e.1

## A.12 Analog sinyal kartları (SB'ler)

### A.12.1 SB 1231 1 analog giriş özellikleri

#### Not

Bu SB'yi kullanmak için sizin CPU donanım yazılımı V2.0 veya üstü olmalıdır.

Çizelge A- 206 Genel özellikler

Teknik veri	SB 1231 AI 1 x 12 bit
Sipariş numarası	6ES7 231-4HA30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 21
Ağırlık	35 gram
Güç kaybı	0.4 W
Çektiği akım(SM Bus)	55 mA
Çektiği akım(24 VDC)	hiçbiri

Çizelge A- 207 Analog girişler

Teknik veri	SB 1231 AI 1x12 bit
Giriş sayısı	1
Tip	Gerilim veya akım (diferansiyel)
Aralık	±10 V, ±5 V, ±2.5 veya 0 - 20 mA
Çözünürlük	11 bit + işaret biti
Tam-ölçek aralığı(veri word)	-27648 ila 27648
Üst/alt aralık (veri word)	Gerilim:32511 ila 27649 / -27649 ila -32512 Akım:32511 - 27649 / 0 ila -4864 Analog gerilim ve akım girişi gösterilişi için bakınız
Taşma/Alt taşma (veri word)	Gerilim:32767 - 32512 / -32513 ila -32768 Akım:32767 - 32512 / -4865 ila -32768 Analog gerilim ve akım girişi gösterilişi için bakınız
Maksimum dayanma gerilimi / akım	±35 V / ±40 mA
Düzleme	Hiçbiri, zayıf, orta veya güçlü (Basamak tepki süresi için analog giriş basamak tepki süresine bakınız.)
Gürültü bastırma	400, 60, 50 veya 10 Hz (Örnekleme hızları için analog giriş tepki süresine bakınız)
Doğruluk (25 °C / -20 ila 60 °C)	Tam ölçeğe göre ±0.3% / ±0.6%
Giriş empedansı	
Diferansiyel	Gerilim:220 kΩ; Akım:250 Ω
Ortak mod	Gerilim:55 kΩ; Akım:55 kΩ

## Teknik özellikler

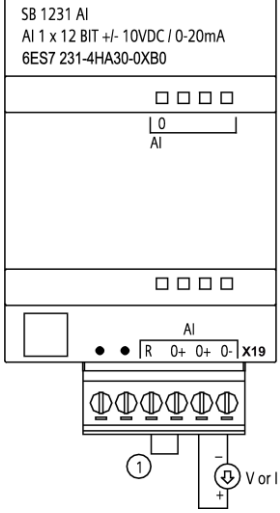
### A.12 Analog sinyal kartları (SB'ler)

Teknik veri	SB 1231 AI 1x12 bit
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)
Ölçme prensibi	Gerçek değer dönüştürme
Ortak mod bastırma	400 dB, DC - 60 Hz
Operasyonel sinyal aralığı	Sinyal artı ortak mod gerilimi +35 V'dan az ve -35 V'dan fazla olmamalıdır
İzolasyon (alan tarafı ile lojik arasında)	Hiçbiri
Kablo uzunluğu (metre)	100 m, bükülü ve zırlı

### Çizelge A- 208 Diyagnostikler

Teknik veri	SB 1231 AI 1 x 12 bit
Taşma/alt taşma	Evet
24 VDC düşük gerilim	hayır

### Çizelge A- 209 Analog giriş SB'ler için bağlantı diyagramı

SB 1231 AI x 12 bit (6ES7 231-4HA30-0XB0)	
 <p>SB 1231 AI AI 1 x 12 BIT +/- 10VDC / 0-20mA 6ES7 231-4HA30-0XB0</p> <p>□ □ □ □ 0 AI</p> <p>□ □ □ □ AI ● ● R 0+ 0+ 0- x19</p> <p>① - + V or I</p>	
① Akım için "R" ucunu "0+" ucuna bağlayınız.	
Not: Konektörler altın olmalıdır. Bakınız sipariş numarası için Ek C, Yedek Parçalar	

Çizelge A- 210 SB 1231 AI x 12 bit (6ES7 231-4HA30-0XB0) için konektör pin yerleşimleri

Pin	X19 (altın)
1	Bağlantı yok
2	Bağlantı yok
3	AI R
4	AI 0+
5	AI 0+
6	AI 0-

## A.12.2 SB 1232 1 Analog çıkış özellikleri

Çizelge A- 211 Genel özellikler

Teknik veri	SB 1232 AQ 1 x 12 bit
Sipariş numarası	6ES7 232-4HA30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 21
Ağırlık	40 gram
Güç kaybı	1.5 W
Çektiği akım(SM Bus)	15 mA
Çektiği akım(24 VDC)	40 mA (yüksüz)

Çizelge A- 212 Analog çıkışlar

Teknik veri	SB 1232 AQ 1 x 12 bit
Giriş sayısı	1
Tip	Gerilim veya akım
Aralık	±10 V veya 0 - 20 mA
Çözünürlük	Gerilim:12 bit Akım:11 bit
Tam-ölçek aralığı(veri word)	Gerilim:-27648 ila 27648
Gerilim ve akım için çıkış aralıklarına bakınız (Sayfa 951).	Akım:0 - 27648
Doğruluk (25 °C / -20 ila 60 °C)	Tam ölçeğe göre %±0.5 / %±1
Yatışma süresi (yeni değere göre 95%)	Gerilim:300 µS (R), 750 µS (1 uF) Akım:600 µS (1 mH), 2 ms (10 mH)
Yük empedansı	Gerilim:≥ 1000 Ω Akım:≤ 600 Ω
RUN'dan STOP'a davranış	Son değer veya değişiklik değeri (varsayılan değer 0)
İzolasyon (alan tarafı ile lojik arasında)	Hiçbiri
Kablo uzunluğu (metre)	100 m, bükülü ve zırlı

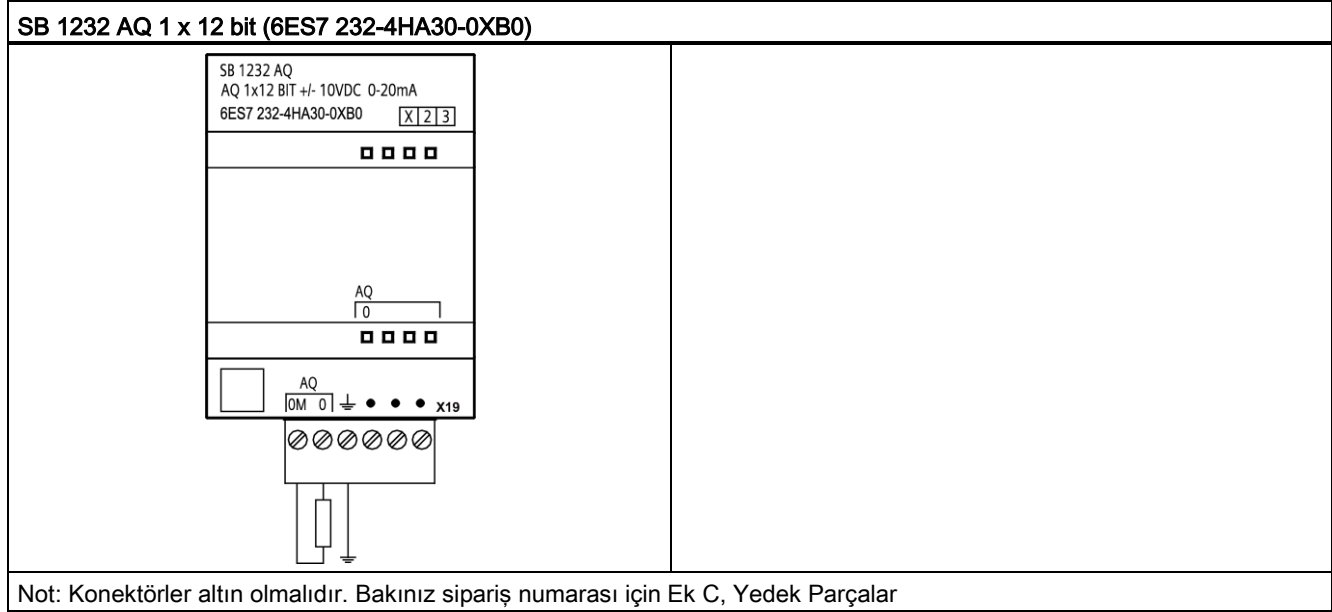
## Teknik özellikler

### A.12 Analog sinyal kartları (SB'ler)

#### Çizelge A- 213 Diyagnostikler

Teknik veri	SB 1232 AQ 1 x 12 bit
Taşma/alt taşma	Evet
Toprağa kısa devre (gerilim modu sadece)	Evet
Kablo kopması (akım modu sadece)	Evet

#### Çizelge A- 214 SB 1232 AQ 1 x 12 bit için bağlantı diyagramı



#### Çizelge A- 215 SB 1232 AQ 1 x 12 bit (6ES7 232-4HA30-0XB0) için konektör pin yerleşimleri

Pin	X19 (altın)
1	AQ 0M
2	AQ 0
3	GND
4	Bağlantı yok
5	Bağlantı yok
6	Bağlantı yok

## A.12.3 Analog girişler ve çıkışlar için ölçme aralıkları

### A.12.3.1 Analog girişlerin basamak tepkileri

Çizelge A- 216 Basamak tepkisi(ms), 0 V - 10 V, %95'de ölçülen

Düzleme seçimi (örnek ortalama)	İntegrasyon süresi seçimi			
	400 Hz (2.5 ms)	60 Hz (16.6 ms)	50 Hz (20 ms)	10 Hz (100 ms)
Hiçbiri (1 döngü): Ortalama alma yok	4.5 ms	18.7 ms	22.0 ms	102 ms
Zayıf (4 döngü): 4 örnek	10.6 ms	59.3 ms	70.8 ms	346 ms
Orta (16 döngü): 16 örnek	33.0 ms	208 ms	250 ms	1240 ms
Güçlü (32 döngü): 32 örnek	63.0 ms	408 ms	490 ms	2440 ms
<b>Örnekleme süresi</b>	<b>0.156 ms</b>	<b>1.042 ms</b>	<b>1.250 ms</b>	<b>6.250 ms</b>

### A.12.3.2 Analog girişler için örnekleme süresi ve güncelleme süresi

Çizelge A- 217 Örnekleme süresi ve güncelleme süresi

Seçim	Örnekleme süresi	SB güncelleme süresi
400 Hz (2.5 ms)	0.156 ms	0.156 ms
60 Hz (16.6 ms)	1.042 ms	1.042 ms
50 Hz (20 ms)	1.250 ms	1.25 ms
10 Hz (100 ms)	6.250 ms	6.25 ms

### A.12.3.3 Akım ve gerilim (SB ve SM) için analog girişlerin ölçme aralığı

Çizelge A- 218 Gerilim (SB ve SM) için analog giriş gösterilişi

Sistem		Gerilim Ölçme Aralığı				
Decimal	Hexadecimal	±10 V	±5 V	±2.5 V	±1.25 V	
32767	7FFF	11.851 V	5.926 V	2.963 V	1.481 V	Taşma
32512	7F00					
32511	7EFF	11.759 V	5.879 V	2.940 V	1.470 V	Aşma aralığı
27649	6C01					
27648	6C00	10 V	5 V	2.5 V	1.250 V	Anma aralığı
20736	5100	7.5 V	3.75 V	1.875 V	0.938 V	
1	1	361.7 µV	180.8 µV	90.4 µV	45.2 µV	
0	0	0 V	0 V	0 V	0 V	
-1	FFFF					
-20736	AF00	-7.5 V	-3.75 V	-1.875 V	-0.938 V	
-27648	9400	-10 V	-5 V	-2.5 V	-1.250 V	

## Teknik özellikler

## A.12 Analog sinyal kartları (SB'ler)

Sistem		Gerilim Ölçme Aralığı				
Decimal	Hexadecimal	$\pm 10$ V	$\pm 5$ V	$\pm 2.5$ V	$\pm 1.25$ V	
-27649	93FF					Alt taşma aralığı
-32512	8100	-11.759 V	-5.879 V	-2.940 V	-1.470 V	
-32513	80FF					Alt taşma
-32768	8000	-11.851 V	-5.926 V	-2.963 V	-1.481 V	

Çizelge A- 219 Akım (SB ve SM) için analog giriş gösterilişi

Sistem		Akım ölçme aralığı		
Decimal	Hexadecimal	0 mA - 20 mA	4 mA - 20 mA	
32767	7FFF	23.70 mA	22.96 mA	Taşma
32512	7F00			
32511	7EFF	23.52 mA	22.81 mA	Aşma aralığı
27649	6C01			
27648	6C00	20 mA	20 mA	Nominal aralık
20736	5100	15 mA	16 mA	
1	1	723.4 nA	4 mA + 578.7 nA	
0	0	0 mA	4 mA	
-1	FFFF			Alt taşma aralığı
-4864	ED00	-3.52 mA	1.185 mA	
-4865	ECFF			Alt taşma
-32768	8000			

## A.12.3.4 Akım ve gerilim (SB ve SM) için analog çıkışların ölçme aralığı

Çizelge A- 220 Gerilim (SB ve SM) için analog çıkış gösterilişi

Sistem		Gerilim çıkış aralığı	
Decimal	Hexadecimal	$\pm 10$ V	
32767	7FFF	Bakınız not 1	Taşma
32512	7F00	Bakınız not 1	
32511	7EFF	11.76 V	Aşma aralığı
27649	6C01		
27648	6C00	10 V	Anma aralığı
20736	5100	7.5 V	
1	1	361.7 $\mu$ V	
0	0	0 V	
-1	FFFF	-361.7 $\mu$ V	
-20736	AF00	-7.5 V	
-27648	9400	-10 V	



Sistem		Gerilim çıkış aralığı	
Decimal	Hexadecimal	± 10 V	
-27649	93FF		Alt taşma aralığı
-32512	8100	-11.76 V	
-32513	80FF	Bakınız not 1	Alt taşma
-32768	8000	Bakınız not 1	

<sup>1</sup> Bir aşırı akım veya alt taşma durumunda, analog çıkışlar STOP modunun yedek değerini alacaktır.

Çizelge A- 221 Akım (SB ve SM) için analog çıkış gösterilişi

Sistem		Akım çıkış aralığı		
Decimal	Hexadecimal	0 mA - 20 mA	4 mA - 20 mA	
32767	7FFF	Bakınız not 1	Bakınız not 1	Taşma
32512	7F00	Bakınız not 1	Bakınız not 1	
32511	7EFF	23.52 mA	22.81 mA	Aşma aralığı
27649	6C01			Anma aralığı
27648	6C00	20 mA	20 mA	
20736	5100	15 mA	16 mA	
1	1	723.4 nA	4 mA + 578.7 nA	
0	0	0 mA	4mA	
-1	FFFF		4 mA - 578.7 nA	
-6912	E500		0 mA	Alt taşma aralığı
-6913	E4FF			
-32512	8100			Mümkün değil. Çıkış değeri 0 mA'e sınırlıdır.
-32513	80FF	Bakınız not 1	Bakınız not 1	Alt taşma
-32768	8000	Bakınız not 1	Bakınız not 1	

<sup>1</sup> Bir aşırı akım veya alt taşma durumunda, analog çıkışlar STOP modunun yedek değerini alacaktır.

## Teknik özellikler

### A.12 Analog sinyal kartları (SB'ler)

#### A.12.4 Termokupl sinyal panelleri (SB'ler)

##### A.12.4.1 SB 1231 1 analog Termokupl giriş özellikleri

### Not

Bu SB'yi kullanmak için sizin CPU donanım yazılımı V2.0 veya üstü olmalıdır.

#### Çizelge A- 222 Genel özellikler

Teknik veri	SB 1231 AI 1 x 16 bit Termokupl
Sipariş numarası	6ES7 231-5QA30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 21
Ağırlık	35 gram
Güç kaybı	0.5 W
Çektiği akım(SM Bus)	5 mA
Çektiği akım(24 VDC)	20 mA

#### Çizelge A- 223 Analog girişler

Teknik veri	SB 1231 AI 1x16 bit Termokupl
Giriş sayısı	1
Tip	Kayan TC ve mV
Aralık	Bakınız Termokupl filtre seçim çizelgesi (Sayfa 954).
• Nominal aralık(veri word)	
• Üst aralık/alt aralık (veri word)	
• Taşma/alt taşma (veri word)	
Çözünürlük	Sıcaklık 0.1° C / 0.1° F
	Gerilim 15 bit artı işareti
Maksimum dayanma gerilimi	±35 V
Gürültü bastırma	85 dB, seçilen filtre ayarı için (10 Hz, 50 Hz, 60 Hz, 400 Hz)
Ortak mod bastırma	> 120 dB 120 VAC'de
Empedans	≥ 10 M Ω
Doğruluk	Bakınız Termokupl seçim çizelgesi (Sayfa 954).
Tekrarlanabilirlik	±0.05% FS
Ölçme prensibi	İntegralini alma
Modül güncelleme zamanı	Bakınız Termokupl filtre seçim çizelgesi (Sayfa 954).
Soğuk birleşme hatası	±1.5° C
İzolasyon (alan tarafı ile lojik arasında)	500 VAC

Teknik veri	SB 1231 AI 1x16 bit Termokupl
Kablo uzunluğu (metre)	100 m, sensöre max.
İletken direnci	100 Ω max.

## Çizelge A- 224 Diyagnostikler

Teknik veri	SB 1231 AI 1 x 16 bit Termokupl
Taşma/alt taşma <sup>1</sup>	Evet
Kablo kopması <sup>2</sup>	Evet

- <sup>1</sup> Aşırı akım, alt taşma ve düşük gerilim diyagnostik alarm bilgileri, modül konfigürasyonunda alarmlar etkisizleştirilse bile, analog veri değerleri içinde rapor edilecektir.
- <sup>2</sup> Kablo kopukluğu alarmı etkisiz duruma getirildiğinde ve sensör bağlantısında açık kablo şartı oluştuğunda, modül rastgele değerler rapor edebilir.

SM 1231 Termokupl (TC) analog sinyal modülü modül girişlerine bağlanan gerilim değerini ölçer. Sıcaklık ölçme tipi "Termokupl" veya "Gerilim" olabilir.

- "Termokupl": Değer, derecenin 10 ile çarpılmasıyla rapor edilecektir (örneğin, 25.3 derece decimal 253 olarak rapor edilecektir).
- "Gerilim": Nominal aralık tam ölçek değeri, decimal 27648 olacaktır.

#### A.12.4.2 Bir Termokupl için temel çalışma

Termokupllar birbirine benzemez iki metalin birbirine yapıştırılması sonucunda biçimlendirilirler. Üretilen gerilim birleşim sıcaklığına orantılıdır. Bu gerilim küçüktür; bir mikrovolt büyük dereceleri temsil edebilir. Termokupldan gerilimin ölçülmesi, ilave birleşimler için kompanse edilmesi ve sonra sonucun doğrusallaştırılması Termokupl kullanarak sıcaklık ölçmenin temellerini oluşturur.

Bir Termokuplu SM 1231'e bağladığınızda, iki farklı metal tel sinyal konektöründeki modüle bağlanır. İki farklı metalin birbirine bağlandığı yer sensör Termokuplu biçimlendirir. Konektör sıcaklığı bir gerime neden olur ve bu gerilim sensör Termokupldan gelen gerilime eklenir.

Sinyal konektörüne iki farklı telin bağlandığı durumda iki daha Termokupl oluşmaktadır. Konektör sıcaklığı bir gerilime neden olur ve bu gerilim sensör Termokupldan gelen gerilime ilave olur. Bu gerilim düzeltilmezse sonra rapor edilen bu sıcaklık sensör sıcaklığından sapacaktır.

Konektör Termokupl için soğuk sıcaklık birleşmesi dengeleyici olarak kullanılır. Termokupl çizelgesi genellikle sıfır derece Celsius referans sıcaklığını esas alır. Soğuk sıcaklık dengeleme konektörü sıfır derece Celsiusa dengeler. Soğuk sıcaklık dengeleme konektör Termokupllarca eklenen gerilimi eski haline getirir. Modülün sıcaklığı dahili olarak ölçülür ve sonra sensör dönüştürmesine ilave edilmek için bir değere dönüştürülür. Düzeltelen sensör dönüştürmesi sonra Termokupl düzeltme çizelgesi kullanılarak doğrusallaştırılır.

## Teknik özellikler

### A.12 Analog sinyal kartları (SB'ler)

Soğuk birleşim dengelemesinin optimum çalışması için Termokupl modülü termal olarak kararlı bir ortama yerleştirilmelidir. Ortam modül sıcaklığında yavaş değişim (0.1 °C/dakikadan az) modül özelliklerine göre doğru olarak dengelenir. Modül uçlarındaki hava hareketi soğuk birleşim dengeleme hatasına da neden olacaktır.

Daha iyi bir soğuk birleşim hata dengelemesi gerekirse, harici bir iso-termal terminal bloğu kullanılabilir. Termokupl modülü 0 °C referanslı veya 50 °C referanslı terminal bloğu sağlar.

### SB 1231 Termokupl için seçim çizelgesi

SB 1231 Termokupl sinyal kartı tarafından desteklenen farklı tip Termokupllar için aralıklar ve doğruluk aşağıdaki çizelgede verilmiştir.

Çizelge A- 225 SB 1231 Termokupl seçim Çizelgesi

Termokupl Tip	Alt aralık min. <sup>1</sup>	Nominal aralık düşük limit	Nominal aralık yüksek limit	Üst aralık max. <sup>2</sup>	Normal aralık <sup>3</sup> doğruluk @ 25 °C	Normal aralık <sup>3</sup> doğruluk -20 °C ila 60 °C
J	-210.0 °C	-150.0 °C	1200.0 °C	1450.0 °C	±0.3 °C	±0.6 °C
K	-270.0 °C	-200.0 °C	1372.0 °C	1622.0 °C	±0.4 °C	±1.0 °C
T	-270.0 °C	-200.0 °C	400.0 °C	540.0 °C	±0.5 °C	±1.0 °C
E	-270.0 °C	-200.0 °C	1000.0 °C	1200.0 °C	±0.3 °C	±0.6 °C
R & S	-50.0 °C	100.0 °C	1768.0 °C	2019.0 °C	±1.0 °C	±2.5 °C
B	0.0 °C	200.0 °C	800.0 °C	--	±2.0 °C	±2.5 °C
	--	800.0 °C	1820.0 °C	1820.0 °C	±1.0 °C	±2.3 °C
N	-270.0 °C	0.0 °C	1300.0 °C	1550.0 °C	±1.0 °C	±1.6 °C
C	0.0 °C	100.0 °C	2315.0 °C	2500.0 °C	±0.7 °C	±2.7 °C
TXK/XK(L)	-200.0 °C	-150.0 °C	800.0 °C	1050.0 °C	±0.6 °C	±1.2 °C
Gerilim	-32511	-27648 -80mV	27648 80mV	32511	±0.05%	±0.1%

<sup>1</sup> Alt-aralık minimum değeri altındaki Termokupl değerleri -32768 olarak rapor edilir.

<sup>2</sup> Üst-aralık maksimum değeri üstündeki Termokupl değerleri 32767 olarak rapor edilir.

Dahili soğuk birleşim hatası bütün aralıklar için ±1.5 °C. Çizelgedeki gibi hatalara bu durum eklenir. Bu özellikleri karşılamak üzere modülün en azından 30 dakika ısınma süresine ihtiyacı vardır.

Çizelge A- 226 SB 1231 Termokupl için filtre seçim Çizelgesi

Bastırma frekansı (Hz)	İntegrasyon süresi (ms)	Sinyal paneli güncelleme süresi (saniye)
10	100	0.306
50	20	0.066
60	16.67	0.056
400 <sup>1</sup>	10	0.036

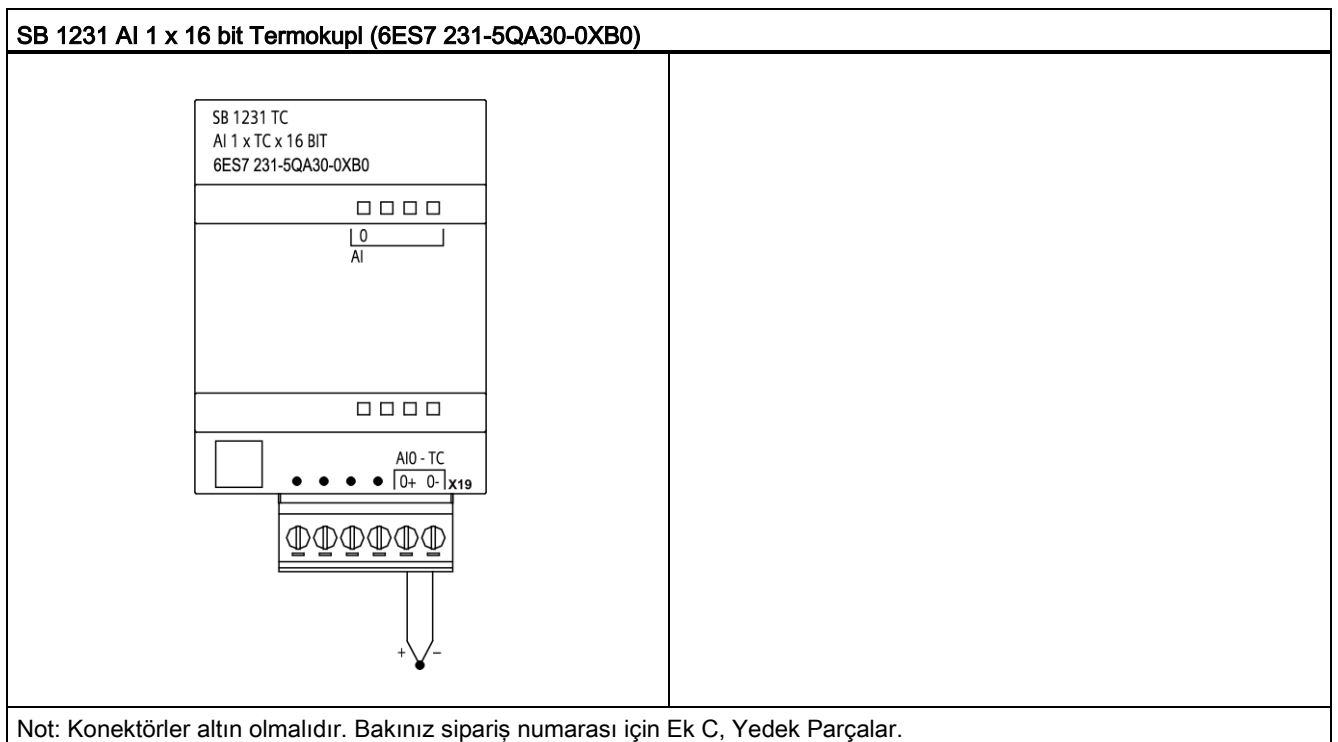
<sup>1</sup> 400 Hz bastırma frekansı seçildiğinde, modül çözünürlüğü ve doğruluğunu korumak için integrasyon süresi 10 ms seçilir. Bu seçim 100Hz ve 200 Hz gürültüyü de bastırır.

Termokuplların ölçmesinde 100 ms integrasyon süresi kullanılması tavsiye edilir. Daha küçük integrasyon süresi kullanımı sıcaklık okumalarının tekrarlanabilirlik hatasını artıracaktır.

### Not

Enerji uygulandıktan sonra, analog-dijital dönüştürücü için modül dahili kalibrasyonu yürütür. Bu sürede o kanal üzerinde geçerli bir veri mevcut oluncaya kadar modül her kanal için 32767 değerini rapor eder. Kullanıcı programınız bu başlangıç ayarı süresini sağlamaya gerek duyabilir.

Çizelge A- 227 SB 1231 AI 1 x 16 Termokupl için bağlantı diyagramı



Çizelge A- 228 SB 1231 AI 1 x 16 bit Termokupl (6ES7 231-5QA30-0XB0) için bağlantı pin yerleşimleri

Pin	X19 (altın)
1	Bağlantı yok
2	Bağlantı yok
3	Bağlantı yok
4	Bağlantı yok
5	AI 0- /TC
6	AI 0+ /TC

## Teknik özellikler

### A.12 Analog sinyal kartları (SB'ler)

#### A.12.5 RTD sinyal panelleri (SB'ler)

##### A.12.5.1 SB 1231 1 analog RTD giriş özellikleri

### Not

Bu SB'yi kullanmak için sizin CPU donanım yazılımı V2.0 veya üstü olmalıdır.

Çizelge A- 229 Genel özellikler

Teknik veri	SB 1231 AI 1 x 16 bit RTD
Sipariş numarası	6ES7 231-5PA30-0XB0
Boyutlar W x H x D (mm)	38 x 62 x 2
Ağırlık	35 gram
Güç kaybı	0.7 W
Çektiği akım(SM Bus)	5 mA
Çektiği akım(24 VDC)	25 mA

Çizelge A- 230 Analog girişler

Teknik veri	SB 1231 AI 1 x 16 bit RTD	
Giriş sayısı	1	
Tip	Modüle göre RTD ve Ohm	
Aralık	Bakınız seçim çizelgesi (Sayfa 960).	
	<ul style="list-style-type: none"> <li>Nominal aralık (veri word)</li> <li>Üst aralık/alt aralık (veri word)</li> <li>Taşma/alt taşma (veri word)</li> </ul>	
Çözünürlük	Sıcaklık	0.1 °C/ 0.1 °F
	Gerilim	15 bit artı işareti
Maksimum dayanma gerilimi	±35 V	
Gürültü bastırma	85 dB (10 Hz, 50 Hz, 60 Hz, 400 Hz)	
Ortak mod bastırma	> 120 dB	
Empedans	≥ 10 MΩ	
Doğruluk	Bakınız seçim çizelgesi (Sayfa 960).	
Tekrarlanabilirlik	±0.05% FS	
Maksimum sensor enerji kaybı	0.5 m W	
Ölçme prensibi	İntegralini alma	
Modül güncelleme zamanı	Bakınız seçim çizelgesi (Sayfa 961).	
İzolasyon (alan tarafı ile lojik)	500 VAC	

Teknik veri	SB 1231 AI 1 x 16 bit RTD
Kablo uzunluğu (metre)	100 m, sensör max. için
İletken direnci	20 $\Omega$ , 2.7 için 10 $\Omega$ RTD max.

## Çizelge A- 231 Diyagnostikler

Teknik veri	SB 1231 AI 1 x 16 bit RTD
Taşma/alt taşma <sup>1, 2</sup>	Evet
Kablo kopması <sup>3</sup>	Evet

<sup>1</sup> Aşırı akım, alt taşma ve düşük gerilim diyagnostik alarm bilgileri, modül konfigürasyonunda alarmlar etkisizleştirilse bile, analog veri değerleri içinde rapor edilecektir.

<sup>2</sup> Direnç aralıkları için alt taşma algılaması asla etkin (enable) yapılmaz.

<sup>3</sup> Kablo (tel) kopukluğu alarmı etkisizleştirildiğinde ve sensör kablolajında açık devre şartı oluştuğunda modül rastgele değerler rapor edebilir.

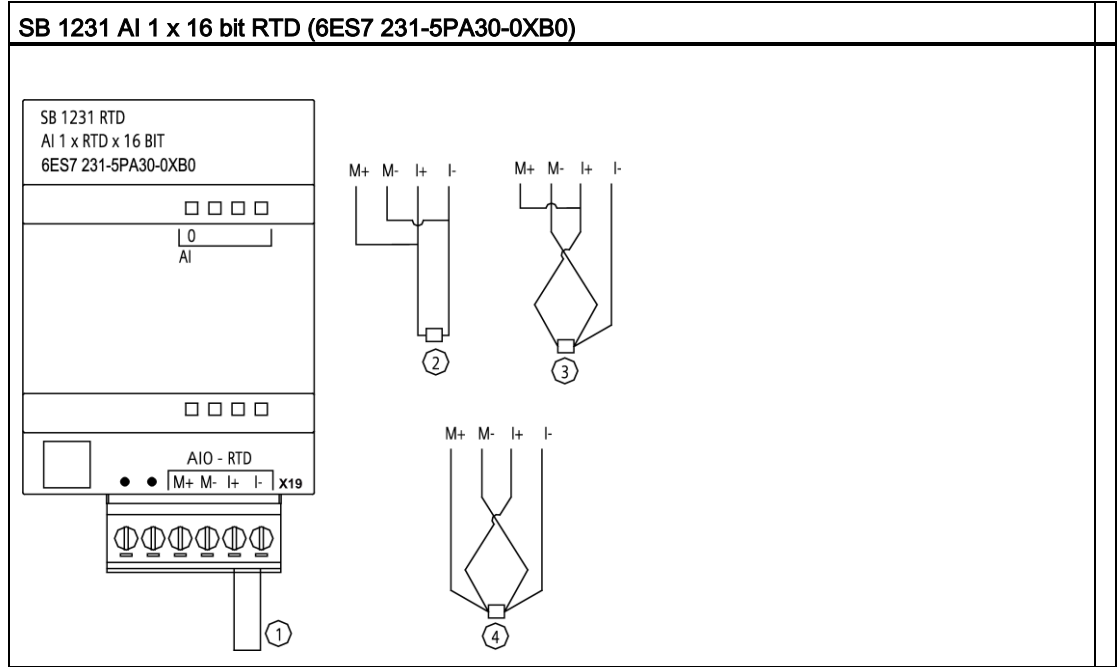
SM 1231 RTD analog sinyal modülü modül girişlerine bağlanan direnç değerini ölçer. Ölçme tipi "Rezistör" veya "Termal Rezistör" olabilir.

- Rezistör": Nominal aralık tam ölçek değeri, decimal 27648 olacaktır.

"Termal Rezistör": Değer, derecenin 10 ile çarpılmasıyla rapor edilecektir (örneğin, 25.3 derece decimal 253 olarak rapor edilecektir). Klimatik aralık değerleri derecenin 100 ile çarpılmasıyla rapor edilecektir (örneğin, 25.34 derece decimal 2534 olarak rapor edilecektir)

The SB 1231 RTD modülü, sensör dirençlerine 2-telli, 3- telli and 4- telli bağlantılarla yapılan ölçmeleri destekler.

Çizelge A- 232 SB 1231 AI 1 x 16 bit RTD için bağlantı diyagramı



- ① Geri-döngü kullanılmamış RTD girişi  
 ② 2-telli RTD  
 ③ 3-telli RTD  
 ④ 4-telli RTD

Not: Konektörler altın olmalıdır. Bakınız sipariş numarası için Ek C, Yedek Parçalar.

Çizelge A- 233 SB 1231 AI 1 x 16 bit RTD (6ES7 231-5PA30-0XB0) için bağlantı yerleşimleri

Pin	X19 (altın)
1	Bağlantı yok
2	Bağlantı yok
3	AI 0 M+ /RTD
4	AI 0 M- /RTD
5	AI 0 I+ /RTD
6	AI 0 I- /RTD



**A.12.5.2 SB 1231 RTD için seçim çizelgesi**

Çizelge A- 234 RTD modüllerince desteklenen farklı sensörler için aralıklar ve doğruluk

Sıcaklık katsayısı	RTD tip	Alt aralık min. <sup>1</sup>	Nominal aralık düşük limit	Nominal aralık yüksek limit	Üst aralık max. <sup>2</sup>	Normal aralık doğruluk @ 25 °C	Normal aralık doğruluk -20 °C ila 60 °C
Pt 0.003850 ITS90 DIN EN 60751	Pt 100 klimatik	-145.00 °C	-120.00 °C	-145.00 °C	-155.00 °C	±0.20 °C	±0.40 °C
	Pt 10	-243.0 °C	-200.0 °C	850.0 °C	1000.0 °C	±1.0 °C	±2.0 °C
	Pt 50	-243.0 °C	-200.0 °C	850.0 °C	1000.0 °C	±0.5 °C	±1.0 °C
	Pt 100						
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0.003902 Pt 0.003916 Pt 0.003920	Pt 100	-243.0 °C	-200.0 °C	850.0 °C	1000.0 °C	± 0.5 °C	±1.0 °C
	Pt 200						
	Pt 500						
	Pt 1000						
Pt 0.003910	Pt 10	-273.2 °C	-240.0 °C	1100.0 °C	1295 °C	±1.0 °C	±2.0 °C
	Pt 50	-273.2 °C	-240.0 °C	1100.0 °C	1295 °C	±0.8 °C	±1.6 °C
	Pt 100						
	Pt 500						
Ni 0.006720 Ni 0.006180	Ni 100	-105.0 °C	-60.0 °C	250.0 °C	295.0 °C	±0.5 °C	±1.0 °C
	Ni 120						
	Ni 200						
	Ni 500						
	Ni 1000						
LG-Ni 0.005000	LG-Ni 1000	-105.0 °C	-60.0 °C	250.0 °C	295.0 °C	±0.5 °C	±1.0 °C
Ni 0.006170	Ni 100	-105.0 °C	-60.0 °C	180.0 °C	212.4 °C	±0.5 °C	±1.0 °C
Cu 0.004270	Cu 10	-240.0 °C	-200.0 °C	260.0 °C	312.0 °C	±1.0 °	±2.0 °C
	Cu 50	-60.0 °C	-50.0 °C	200.0 °C	240.0 °C	±1.0 °C	±2.0 °C
	Cu 100						
Cu 0.004280	Cu 10	-240.0 °C	-200.0 °C	200.0 °C	240.0 °C	±1.0 °C	±2.0 °C
	Cu 50	-240.0 °C	-200.0 °C	200.0 °C	240.0 °C	±0.7 °C	±1.4 °C
	Cu 100						

<sup>1</sup> Alt-aralık minimum değeri altındaki RTD değerleri -32768 olarak rapor edilir.

<sup>2</sup> Üst-aralık maksimum değeri üstündeki RTD değerleri +32768 olarak rapor edilir.

## Teknik özellikler

## A.12 Analog sinyal kartları (SB'ler)

Çizelge A- 235 Direnç

Aralık	Alt aralık minimum	Nominal aralık düşük limit	Nominal aralık yüksek limit	Üst aralık max. <sup>1</sup>	Normal aralık doğruluk @ 25 °C	Doğruluk -20 °C ila 60 °C
150 Ω	n/a	0 (0 Ω)	27648 (150 Ω)	176.383 Ω	±0.05%	±0.1%
300 Ω	n/a	0 (0 Ω)	27648 (300 Ω)	352.767 Ω	±0.05%	±0.1%
600 Ω	n/a	0 (0 Ω)	27648 (600 Ω)	705.534 Ω	±0.05%	±0.1%

<sup>1</sup> Üst-aralık maksimum değeri üstündeki direnç değerleri 32768 olarak rapor edilir.

**Not**

Modül, sensör bağlı değilken aktif duruma getirilen bir kanal için 32767 değerini rapor eder. Açık kablo (tel) algılaması da etkin ise modül uygun kırmızı LED'leri yakıp söndürür.

4-telli bağlantılar kullanılırsa 10 Ω RTD aralıkları için en iyi doğruluk gerçekleştirilecektir.

2-telli moda bağlantı tellerinin direnci sensör okumasında bir hataya neden olacaktır. Bundan dolayı doğruluk garanti edilemez.

Çizelge A- 236 RTD modüller için gürültü azaltma ve güncelleme süreleri

Bastırma süresi seçimi	İntegrasyon süresi	4-/2-telli, 1-kanal modül Güncelleme süresi (saniye)	3-telli, 1-kanal modül Güncelleme süresi (saniye)
400 Hz (2.5 ms)	10 ms <sup>1</sup>	0.036	0.071
60 Hz (16.6 ms)	16.67 ms	0.056	0.111
50 Hz (20 ms)	20 ms	0.066	1.086
10 Hz (100 ms)	100 ms	0.306	0.611

<sup>1</sup> 400 Hz bastırma frekansı seçildiğinde, modül çözünürlüğü ve doğruluğunu korumak için integrasyon süresi 10 ms seçilir. Bu seçim 100Hz ve 200 Hz gürültüyü de bastırır.

**Not**

Enerji uygulandıktan sonra, analog-dijital dönüştürücü için modül dahili kalibrasyonu yürütür. Bu sürede o kanal üzerinde geçerli bir veri mevcut oluncaya kadar modül her kanal için 32767 değerini rapor eder. Kullanıcı programınız bu başlangıç ayarı süresini sağlamaya gerek duyabilir. Bu modülün konfigürasyonu başlangıç ayarı süresinin uzunluğunu değiştirebileceğinden konfigürasyonunuzda bu modülün davranışını doğrulamanız gerekir. Gerekirse, modülün bu başlangıç ayarı süresini belirlemek için programınıza bir lojik dahil edebilirsiniz.

## A.13 BB 1297 Batarya paneli

### BB 1297 Batarya paneli

S7-1200 BB 1297 Batarya paneli gerçek-zaman saatine uzun-sürelili besleme sağlamak için tasarlanir. Batarya paneli, S7-1200 CPU (donanım yazılımı 3.0 ve sonraki versiyonları) sinyal kartı yuvasına takılabilir. Cihaz konfigürasyonuna BB 1297'yi eklemelisiniz ve BB'nin fonksiyonel olabilmesi için CPU'ya donanım konfigürasyonunu yüklemelisiniz.

BB 1297 ile birlikte batarya (tip CR1025) bulunmamaktadır ve kullanıcı tarafından satın alınmalıdır.

#### Not

Donanım yazılımı 3.0 veya daha sonraki versiyonlarda olan CPU'lara uyması için BB 1297 mekanik olarak tasarlanir.

BB 1297 konektörü CPU'ya takılamayacağı için BB 1297'yi daha önceki versiyon CPU'lar ile kullanmayınız.

#### Çizelge A- 237 Genel özellikler

Teknik veri	BB 1297 Batarya paneli
Sipariş numarası	6ES7 297-0AX30-0XA0
Boyutlar W x H x D (mm)	38 x 62 x 21
Ağırlık	28 gram
Güç kaybı	0.5 W
Çektiği akım(SM Bus)	11 mA
Çektiği akım(24 VDC)	hiçbiri

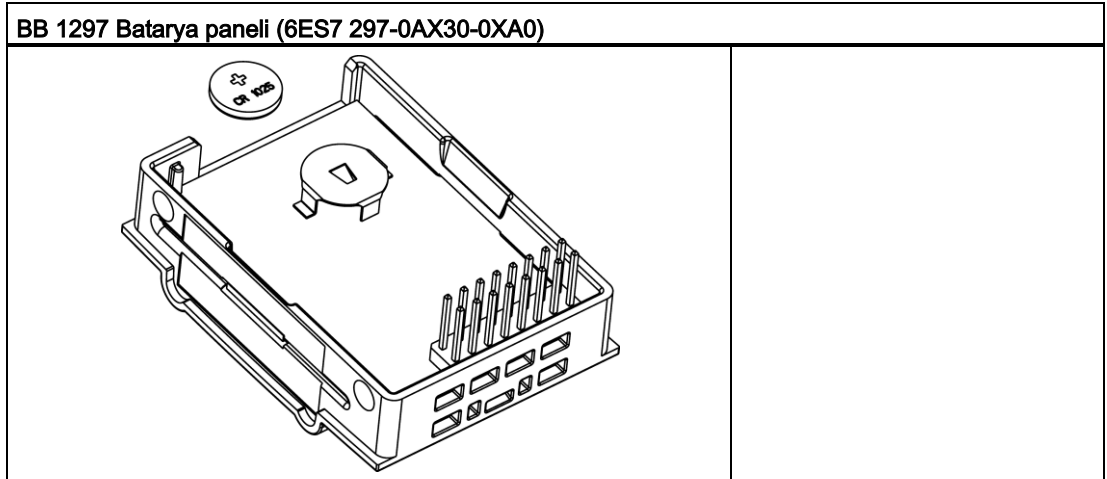
Batarya (dahil değildir)	BB 1297 batarya paneli
Tutma süresi	Yaklaşık 1 yıl
Batarya tipi	CR1025 BB 1297 batarya panelindeki batarya değiştirme veya takmaya bakınız (Sayfa 58 58)
Nominal gerilim	3 V
Nominal kapasite	En azından 30 mAH

Teknik özellikler

A.13 BB 1297 Batarya paneli

Diyagnostikler	BB 1297 batarya paneli
Kritik batarya seviyesi	< 2.5 V
Batarya diyagnostiği	Düşük gerilim göstereci: <ul style="list-style-type: none"> <li>Düşük batarya gerilimi, CPU MAINT LED'inin bal rengi (amber) ışıkla sürekli olarak ON ışıklandırmasına neden olacaktır.</li> <li>Diyagnostik Arabellek Olayı: 16#06:2700 "Alt modül bakımı talep edilir: En azından bir batarya bitmiştir (BATTFF)"</li> </ul>
Batarya durumu	Batarya durum biti vardır 0 = Batarya OK 1 = Batarya düşük
Batarya durumu güncelleme	Enerji uygulandığında, batarya durumu güncellenir ve sonra CPU RUN modunda iken günde bir defa güncellenir.

Çizelge A- 238 BB 1297 batarya paneli için takma şekli



## A.14 Haberleşme ara yüzleri

### A.14.1 PROFIBUS

#### A.14.1.1 CM 1242-5 PROFIBUS DP Slave

Çizelge A- 239 CM 1242-5 için teknik özellikler

<b>Teknik özellikler</b>	
Sipariş numarası	6GK7 242-5DX30-0XE0
<b>Ara yüzler</b>	
PROFIBUS'a bağlantı	9-pin D-sub dişi konektör
Ara yüz devre bileşenleri bağlandığında (örneğin optik devre bileşenleri) PROFIBUS üzerinden maksimum çekilen akım	15 mA, 5 V'da (sadece bus sonlandırması için) *)
<b>İzin verilen ortam sıcaklığı</b>	
Ortam sıcaklığı	
• depolama süresince	• -40 °C - 70 °C
• taşıma süresince	• -40 °C - 70 °C
• çalışma süresince, dikey montajlı (DIN ray yatay)	• 0 °C - 55 °C
• çalışma süresince, yatay montajlı (DIN ray dikey)	• 0 °C - 45 °C
Çalışma safhasında 25 °C'daki bağıl nem, yoğuşma yok, maksimum	%95
Koruma derecesi	IP20
<b>Güç kaynağı, Çektiği akım ve güç kayıpları</b>	
Güç kaynağının tipi	DC
Güç kaynağı, arka plan bus'dan	5 V
Çektiği akım(tipik)	150 mA
Etkin güç kaybı (tipik)	0.75 W
Elektriksel izolasyon	710 VDC için 1 dakika
• PROFIBUS ara yüz ile topraklama arasında	
• PROFIBUS ara yüz ile dahili devre arasında	
<b>Boyutlar ve ağırlıklar</b>	
• Genişlik	• 30 mm
• Yükseklik	• 100 mm
• Derinlik	• 75 mm

## Teknik özellikler

### A.14 Haberleşme ara yüzleri

#### Teknik özellikler

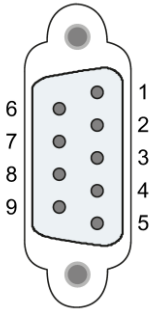
##### Ağırlık

- Net ağırlık
- 115 g
- Ağırlık, paketleme dahil
- 152 g

\*) VP (pin 6) ile DGND (pin 5) arasına bağlı harici bir geçerli yük bus sonlandırmasının maksimum değeri olan 15 mA'i (kısa devre korumalı) geçmemelidir.

## PROFIBUS ara yüz

Çizelge A- 240 D-sub soketi için Pin uçları



Pin	Açıklama	Pin	Açıklama
1	- kullanılmamış-	6	P5V2: +5V güç kaynağı
2	- kullanılmamış-	7	- kullanılmamış-
3	RxD/TxD-P: Veri hattı B	8	RxD/TxD-N: Veri hattı A
4	RTS	9	- kullanılmamış-
5	M5V2: Veri referans potansiyeli (toprak DGND)	Yuva	Topraklama konektörü

### A.14.1.2 CM 1243-5 PROFIBUS DP Master

Çizelge A- 241 CM 1243-5 için teknik özellikler

#### Teknik özellikler

Sipariş numarası 6GK7 243-5DX30-0XE0

#### Ara yüzler

PROFIBUS'a bağlantı 9-pin D-sub dişi konektör

PROFIBUS ara yüzüne devre bileşenleri bağlandığında maksimum çekilen akım (örneğin optik devre bileşenleri) 15 mA, 5 V 'da (sadece bus sonlandırması için) \*)

#### İzin verilen ortam sıcaklığı

##### Ortam sıcaklığı

- depolama süresince
- -40 °C ila 70 °C
- taşıma süresince
- -40 °C ila 70 °C
- çalışma süresince, dikey montajlı (DIN ray yatay)
- 0 °C - 55 °C
- çalışma süresince, yatay montajlı (DIN ray dikey)
- 0 °C - 45 °C

Çalışma safhasında 25 °C'daki bağıl nem, yoğuşma yok, maksimum 95 %

Koruma derecesi IP20

#### Güç kaynağı, çektiği akım ve güç kayıpları

Güç kaynağının tipi DC

### Teknik özellikler

Güç kaynağı / harici	24 V
• minimum	• 19.2 V
• maksimum	• 28.8 V
Çektiği akım (tipik)	
• 24 V DC kaynaktan	• 100 mA
• S7-1200 arka plan bus'tan	• 0 mA
Etkin güç kaybı (tipik)	
• 24 V DC kaynaktan	• 2.4 W
• S7-1200 arka plan bus'tan	• 0 W
Güç kaynağı 24 VDC / harici	
• Min. kablo kesiti	• min.: 0.14 mm <sup>2</sup> (AWG 25)
• Max. kablo kesiti	• max.: 1.5 mm <sup>2</sup> (AWG 15)
• Vida terminallerinin sıkma torqu	• 0.45 Nm (4 lb-in)
Elektriksel izolasyon	710 VDC için 1 dakika
• PROFIBUS ara yüz ile topraklama arası	
• PROFIBUS ara yüz ile dahili devre arası	

### Boyutlar ve ağırlıklar

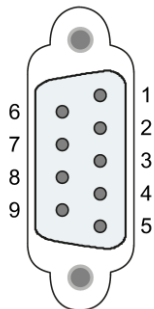
• Genişlik	• 30 mm
• Yükseklik	• 100 mm
• Derinlik	• 75 mm
Ağırlık	
• Net ağırlık	• 134 g
• Ağırlık, paketlenme dahil	• 171 g

\*) VP (pin 6) ile DGND (pin 5) arasına bağlı harici bir geçerli yük bus sonlandırmasının maksimum değeri olan 15 mA'yi (kısa-devre korumalı) geçmemelidir.

### Not

CM 1243-5 (PROFIBUS master modül), CPU'nun 24 VDC sensör beslemesinden enerji almalıdır.

## PROFIBUS ara yüz



Çizelge A- 242 D-sub soket için Pin uçları

Pin	Açıklama	Pin	Açıklama
1	- kullanılmamış -	6	VP: Güç kaynağı +5 V, bus sonlandırma rezistörleri için; harici cihazları beslemek için değil
2	- kullanılmamış -	7	- kullanılmamış -

kontrolör

/2014, A5E02486680

## Teknik özellikler

### A.14 Haberleşme ara yüzleri

Pin	Açıklama	Pin	Açıklama
3	RxD/TxD-P: Veri hattı B	8	RxD/TxD-N: Veri hattı A
4	CNTR-P: RTS	9	- Kullanılmamış -
5	DGND: veri sinyalleri ve VP için topraklama	Yuva	Topraklama konektörü

### PROFIBUS kablosu

#### Not

#### PROFIBUS kablosunun zırhının bağlantısı

PROFIBUS kablosunun zırhı temas halinde olmalıdır.

Bunu yapmak için PROFIBUS kablosunun sonundan yalıtımı soyunuz ve zırhı fonksiyonel toprağa bağlayınız.

### A.14.2 GPRS

#### Not

#### GPRS CP, denizcilik uygulamaları için onaylanmamıştır.

Aşağıdaki modül, denizcilik onayına sahip değildir:

- CP 1242-7 GPRS modülü

#### Not

Bu modülleri kullanmak için CPU donanım yazılımınız V2.0 veya üstü olmalıdır.

### A.14.2.1 CP 1242-7 GPRS

Çizelge A- 243 CP 1242-7 için teknik özellikler

Teknik özellikler	
Sipariş numarası	6GK7 242-7KX30-0XE0
Kablosuz ara yüz	
Anten konektör	SMA soket
Nominal empedans	50 ohm
Kablosuz bağlantı	



**Teknik özellikler**

Maksimum iletim gücü	<ul style="list-style-type: none"><li>• GSM 850, sınıf 4: +33 dBm ±2dBm</li><li>• GSM 900, sınıf 4: +33 dBm ±2dBm</li><li>• GSM 1800, sınıf 1: +30 dBm ±2dBm</li><li>• GSM 1900, sınıf 1: +30 dBm ±2dBm</li></ul>
GPRS	çoklu oluk sınıf 10 cihaz sınıfı B kodlama şeması 1...4 (GMSK)
SMS	Mod giden: MO servis: noktadan noktaya

**İzin verilen ortam sıcaklığı**

Ortam sıcaklığı	
<ul style="list-style-type: none"><li>• depolama süresince</li><li>• taşıma süresince</li><li>• çalışma süresince, dikey montajlı (DIN ray yatay)</li><li>• çalışma süresince, yatay montajlı (DIN ray dikey)</li></ul>	<ul style="list-style-type: none"><li>• -40 °C ila 70 °C</li><li>• -40 °C ila 70 °C</li><li>• 0 °C - 55 °C</li><li>• 0 °C - 45 °C</li></ul>
Çalışma safhasında 25 °C'daki bağıl nem, yoğuşma yok, maksimum	%95
Koruma derecesi	IP20

**Güç kaynağı, Çektiği akım ve güç kayıpları**

Güç kaynağının tipi	DC
Güç kaynağı / harici	24 V
<ul style="list-style-type: none"><li>• minimum</li><li>• maksimum</li></ul>	<ul style="list-style-type: none"><li>• 19.2 V</li><li>• 28.8 V</li></ul>
Çektiği akım(tipik)	
<ul style="list-style-type: none"><li>• 24 V DC'dan</li><li>• S7-1200 arka plan bus'tan</li></ul>	<ul style="list-style-type: none"><li>• 100 mA</li><li>• 0 mA</li></ul>
Etkin güç kaybı (tipik)	
<ul style="list-style-type: none"><li>• 24 V DC kaynaktan</li><li>• S7-1200 arka plan bus'tan</li></ul>	<ul style="list-style-type: none"><li>• 2.4 W</li><li>• 0 W</li></ul>
24 V DC güç kaynağı	
<ul style="list-style-type: none"><li>• Min. kablo kesiti</li><li>• Max. Kablo kesiti</li><li>• Terminal vidalarını sıkma torku</li></ul>	<ul style="list-style-type: none"><li>• min.: 0.14 mm<sup>2</sup> (AWG 25)</li><li>• max.: 1.5 mm<sup>2</sup> (AWG 15)</li><li>• 0.45 Nm (4 lb-in)</li></ul>
Elektriksel izolasyon	710 VDC için 1 dakika
Dahili devre için güç kaynağı	

**Boyutlar ve ağırlıklar**

<ul style="list-style-type: none"><li>• Genişlik</li><li>• Yükseklik</li><li>• Derinlik</li></ul>	<ul style="list-style-type: none"><li>• 30 mm</li><li>• 100 mm</li><li>• 75 mm</li></ul>
---	--

*Teknik özellikler**A.14 Haberleşme ara yüzleri***Teknik özellikler**

## Ağırlık

- Net ağırlık
- Ağırlık, paketleme dahil
- 133 g
- 170 g

**ANT794-4MR GSM/GPRS antenin teknik özellikleri****ANT794-4MR**

Sipariş numarası	6NH9860-1AA00
Mobil kablosuz şebekeler	GSM/GPRS
Frekans aralıkları	<ul style="list-style-type: none"><li>• 824 - 960 MHz (GSM 850, 900)</li><li>• 1 710 - 1 880 MHz (GSM 1 800)</li><li>• 1 900 - 2 200 MHz (GSM / UMTS)</li></ul>
Karakteristikler	Her yönlü
Anten kazancı	0 dB
Empedans	50 ohm
Duran dalga oranı(SWR)	< 2,0
Max. güç	20 W
Polarite	lineer dikey
Konektör	SMA
Anten kablosu uzunluğu	5 m
Harici materyal	Sert PVC, UV-dayanımlı
Koruma derecesi	IP20
İzin verilen ortam sıcaklığı	<ul style="list-style-type: none"><li>• Çalışma sıcaklığı</li><li>• Taşıma/depolama sıcaklığı</li><li>• Bağıl nem</li><li>• -40 °C ila +70 °C</li><li>• -40 °C ila +70 °C</li><li>• %100</li></ul>
Harici materyal	Sert PVC, UV-dayanımlı
Yapılış	Anten, 5 m sabit kablo ve SMA erkek konektöre sahip
Boyutlar (D x H) in mm	25 x 193
Ağırlık	<ul style="list-style-type: none"><li>• Anten, kablo dahil</li><li>• Bağlantı donatıları</li><li>• 310 g</li><li>• 54 g</li></ul>
Montaj	Sağlanan bağlantı ayağı ile

**ANT794-3M düz antenin teknik özellikleri**

Sipariş numarası	6NH9870-1AA00	
Mobil kablosuz şebekeler	<b>GSM 900</b>	<b>GSM 1800/1900</b>
Frekans aralığı	890 - 960 MHz	1710 - 1990 MHz
Duran dalga oranı(VSWR)	≤ 2:1	≤ 1,5:1
Yansıma kayıpları (Tx)	≈ 10 dB	≈ 14 dB
Anten kazancı	0 dB	
Empedans	50 ohm	
Max. güç	10 W	
Anten kablosu	HF kablo RG 174 (sabit), SMA erkek konektör ile	
Kablo uzunluğu	1.2 m	
Koruma derecesi	IP64	
İzin verilen sıcaklık aralığı	-40°C ila +75°C	
Tutuşabilirlik	UL 94 V2	
Harici materyal	ABS Polylac PA-765, açık gri (RAL 7035)	
Boyutlar (W x L x H), mm	70.5 x 146.5 x 20.5	
Ağırlık	130 g	

**A.14.3 CM 1243-2 AS-i master****A.14.3.1 AS-i master CM 1243-2 için teknik veri**

Çizelge A- 244 AS-i master CM 1243-2 için teknik veri

<b>Teknik veri</b>	
Sipariş numarası	3RK7243-2AA30-0XB0
Donanım yazılımı versiyonu	V1.0
Tarih	01.12.2011
<b>Ara yüzler</b>	
Maksimum çektiği akım	
S7-1200 arka plan bus'tan	Max. 250 mA, besleme gerilimi S7-1200 haberleşme bus 5 V DC
AS-i kablosundan	Max. 100 mA
ASI+/ASI- terminalleri arasında maksimum akım taşıma kapasitesi	8 A
Pin ataması	AS-i master CM1243-2 için elektrik bağlantıları bölümüne bakınız (Sayfa 972)
İletken kesiti	0.2 mm <sup>2</sup> (AWG 24) ... 3.3 mm <sup>2</sup> (AWG 12)
ASI konektör sıkma torku	0.56 Nm
<b>İzin verilebilir ortam koşulları</b>	

## Teknik özellikler

### A.14 Haberleşme ara yüzleri

#### Teknik veri

Ortam sıcaklığı	
Depolama süresince	-40 °C ... 70 °C
Taşıma süresince	-40 °C ... 70 °C
Çalışma fazı boyunca, dik montaj ile (yatay standart montaj rayı)	0 °C ... 55 °C
Çalışma fazı boyunca, yatay montaj ile (dik standart montaj rayı)	0 °C ... 45 °C
Çalışma safhasında 25 °C'daki bağıl nem, yoğuşma yok, maksimum	95 %
Koruma derecesi	IP20

#### Güç kaynağı, akım tüketimi, güç kaybı

Güç kaynağının tipi	DC
Çektiği akım (tipik olarak)	
S7-1200 arka plan bus'tan	200 mA
Toplam güç kaybı (tipik olarak):	
• S7-1200 arka plan bus'tan	1 W
• AS-i kablodan	2.4 W

#### Boyutlar ve ağırlıklar

Genişlik	30 mm
Yükseklik	100 mm
Derinlik	75 mm
Ağırlık	
Net ağırlık	122 g
Ağırlık, paketlenme dahil	159 g

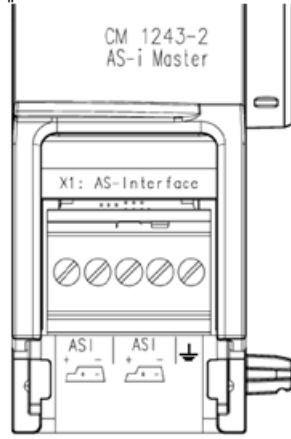
### A.14.3.2 AS-i master CM 1243-2 için elektrik bağlantıları


#### AS-i master CM 1243-2 için güç kaynağı

AS-i master CM 1243-2 beslemesi S7-1200'ün haberleşme yolu üzerinden yapılır. Bu, bir diyagnostik mesajının AS-i kaynak gerilimindeki bir arızanın ardından S7-1200'e yine de gönderilebildiği anlamına gelir. Haberleşme yoluna bağlantı AS-i master CM 1243-2'nin sağ tarafından yapılır.

## AS-Ara yüz terminalleri

AS-i kablosu bağlantısı için çıkarılabilir terminal, AS-i master CM 1243-2'nin ön tarafında alt kapağın arkasındadır.



AS-i biçimli kablo kullanılırsa,  sembolü vasıtasıyla kablonun doğru polaritesini tanıyabilirsiniz. Terminal bloğunun nasıl takılacağı ve çıkarılacağı ile ilgili bilgiyi sistem kılavuzunda "SIMATIC S7-1200 Programmable Controller" (Order No.: 6ES7298-8FA30 -8AH0) bulabilirsiniz.

### Not


#### Terminal kontaklarının maksimum akım taşıma kapasitesi

Bağlantı kontaklarının akım taşıma kapasitesi maksimum 8 A'dir. AS-i kablosu akımı bu değeri aşarsa AS-i master CM 1243-2, AS-i kablosuna döngü içerisinde olmamalıdır, ancak bunun yerine bir destek hattı yoluyla bağlanmalıdır (AS-i master CM 1243-2 üzerinde sadece bir bağlantı çifti ayrılmıştır).

AS-i master vasıtasıyla akım iletiliyor ve 4 A'den daha büyük akım bulunuyorsa, kullanılan kabloların en azından 75°C çalışma sıcaklığı için uygun olduğundan da lütfen emin olunuz.

AS-i kablo bağlantısı ile ilgili ilave bilgiye " SIMATIC S7-1200 için AS-i Master CM 1243-2 ve AS-i veri dekuplaj birimi DCM 1271 " kılavuzu "Modüllerin tesisi, bağlantısı ve devreye alınması" bölümünde bulabilirsiniz.

## Terminal ataması

Etiket	Anlamı
ASI+	AS- i bağlantısı - pozitif polarite
ASI-	AS i bağlantısı - negatif polarite
	Fonksiyonel topraklama

## Teknik özellikler

### A.14 Haberleşme ara yüzleri

#### A.14.4 RS232, RS422 ve RS485

##### A.14.4.1 CB 1241 RS485 özellikler

#### Not

Bu SB'yi kullanmak için sizin CPU donanım yazılımı V2.0 veya üstü olmalıdır.

Çizelge A- 245 Genel özellikler

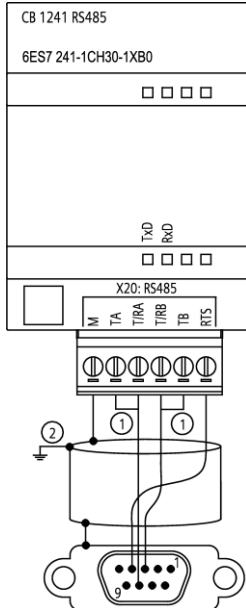
Teknik veri	CB 1241 RS485
Sipariş numarası	6ES7 241-1CH30-1XB0
Boyutlar W x H x D (mm)	38 x 62 x 21
Ağırlık	40 gram

Çizelge A- 246 Aktarıcı ve alıcı

Teknik veri	CB 1241 RS485
Tip	RS485 (2-tel yarı-duplex)
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 saniye, 3 VRMS sürekli
Aktarıcı diferansiyel çıkış gerilimi	2 V min. $R_L = 100 \Omega$ iken 1.5 V min. $R_L = 54 \Omega$ iken
Sonlandırma ve polarlama	10K ila +5 V, B üstünde, RS485 Pin 3 10K ile GND, A üstünde, RS485 Pin 4
Opsiyonel sonlandırma	Pin T/RB'ye kısa Pin TB, etkin sonlandırma empedansı 127 $\Omega$ , RS485 Pin 3'e bağlantı sağlar. Pin T/RA'ya kısa Pin TA, etkin sonlandırma empedansı 127 $\Omega$ , RS485 Pin 4'e bağlantı sağlar.
Alıcı giriş empedansı	5.4K $\Omega$ min. Sonlandırma dahil
Alıcı eşik/duyarlılık	+/- 0.2 V min., 60 mV tipik histeresis
İzolasyon RS485 sinyal ve şasi topraklaması arasında RS485 sinyal ve CPU lojik orta uç arasında	500 VAC, 1 dakika
Kablo uzunluğu, zırlı	1000 m max.
Baut hızı bağımlı	300 baud, 600 baud, 1.2 kbit, 2.4 kbit, 4.8 kbit, 9.6 kbit (varsayılan), 19.2 kbit, 38.4 kbit, 57.6 kbit, 76.8 kbit, 115.2 kbit,
Parite	Parite yok (varsayılan), çift, tek, işaret (parite biti daima 1'e ayarlı), Ara çubuğu (parite biti daima 0'a ayarlı)
stop bit sayısı	1 (varsayılan), 2
Akış kontrol	Desteklenmez
Bekleme süresi	0 - 65535 ms

Çizelge A- 247 Güç kaynağı

Teknik veri	CB 1241 RS485
Güç kaybı (harcaması)	1.5 W
Çektiği akım (SM Bus), max.	50 mA
Çektiği akım (24 VDC) max.	80 mA

CB 1241 RS485 (6ES7 241-1CH30-1XB0)	
	
<p>① Devreyi sonlandırmak için Connect "TA" ve "TB" bağlantısını gösterildiği gibi yapınız. (RS485 devresi üzerindeki sadece son cihazları sonlandırınız.)</p> <p>② Zırlı bükülü kablo çifti kullanınız ve kablo zırhını toprağa bağlayınız.</p>	

RS485 devresinin sadece iki ucunu sonlandırınız. İki uç arasındaki cihazlar sonlandırılmaz veya polarlanmaz. S7-1200 Sistem Kılavuzundaki "RS485 devre konnektörünü polarlama ve sonlandırma" bölümüne bakınız.

Çizelge A- 248 CB 1241 RS485 (6ES7 241-1CH30-1XB0) için konektör pin yerleşimleri

Pin	9-Pin konektör	X20
1	RS485 / Lojik GND	--
2	RS485 / Kullanılmaz	--
3	RS485 / TxD+	3 - T/RB
4	RS485 / RTS	1 - RTS
5	RS485 / Lojik GND	--
6	RS485 / 5 V Güç	--
7	RS485 / Kullanılmaz	--
8	RS485 / TxD-	4 - T/RA

## Teknik özellikler

### A.14 Haberleşme ara yüzleri

Pin	9-Pin konektör	X20
9	RS485 / Kullanılmaz	--
Hücre		7 - M

### Ayrıca bakınız

Bir RS485 devre konektörünü polarlama ve sonlandırma (Sayfa 670)

### A.14.4.2 CM 1241 RS232 özellikler

Çizelge A- 249 Genel özellikler

Teknik veri	CM 1241 RS232
Sipariş numarası	6ES7 241-1AH32-0XB0
Boyutlar (mm)	30 x 100 x 75
Ağırlık	150 gram

Çizelge A- 250 Aktarıcı ve alıcı

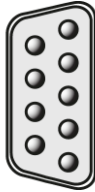
Teknik veri	CM 1241 RS232
Tip	RS232 (tam-duplex)
Aktarıcı çıkış gerilimi	+/- 5 V min., $R_L = 3K \Omega$ 'da
Aktarma çıkış gerilimi	+/- 15 VDC max.
Alıcı giriş empedansı	3 K $\Omega$ min.
Alıcı eşik/duyarlılık	0.8 V min. düşük, 2.4 max. yüksek 0.5 V tipik histeresis
Alıcı giriş gerilimi	+/- 30 VDC max.
İzolasyon RS 232 sinyali, şasi topraklamasına göre RS 232 sinyali, CPU lojik orta noktasına göre	500 VAC, 1 dakika
Kablo uzunluğu, zırlı	10 m max.
Baut hızı	300 baud, 600 baud, 1.2 kbit, 2.4 kbit, 4.8 kbit, 9.6 kbit (varsayılan), 19.2 kbit, 38.4 kbit, 57.6 kbit, 76.8 kbit, 115.2 kbit,
Parite	Parite yok (varsayılan), çift, tek, işaret (parite biti daima 1'e ayarlı), Ara çubuğu (parite biti daima 0'a ayarlı)
Stop bitlerinin numarası	1 (varsayılan), 2
Akış kontrol	Donanım, yazılım
Bekleme süresi	0- 65535 ms



Çizelge A- 251 Güç kaynağı

<b>Teknik veri</b>	<b>CM 1241 RS232</b>
Güç kaybı (harcaması)	1 W
+5 VDC'den	200 mA

Çizelge A- 252 RS232 konektör (erkek)

Pin	Açıklama	Konektör (erkek)	Pin	Açıklama
1 DCD	Veri taşıyıcı algılama: Giriş		6 DSR	Veri set hazır: Giriş
2 RxD	DCE'den alınan veri: Giriş		7 RTS	Göndermek için istem: Çıkış
3 TxD	DCE'ye aktarılan veri: Çıkış		8 CTS	Göndermek için silme: Giriş
4 DTR	Veri terminali hazır: Çıkış		9 RI	Zil gösterici (kullanılmaz)
5 GND	Lojik topraklama		SHELL	Şasi topraklama

#### A.14.4.3 CM 1241 RS422/485 özellikler

#### CM 1241 RS422/485 Özellikler

Çizelge A- 253 Genel özellikler

<b>Teknik veri</b>	<b>CM 1241 RS422/485</b>
Sipariş numarası	6ES7 241-1CH32-0XB0
Boyutlar W x H x H (mm)	30 x 100 x 75
Ağırlık	155 gram

Çizelge A- 254 Aktarıcı ve alıcı

<b>Teknik veri</b>	<b>CM 1241 RS422/485</b>
Tip	RS422 veya RS485, 9-pin sub D dişi konektör
Ortak mod gerilim aralığı	-7 V ila +12 V, 1 saniye, 3 VRMS sürekli
Aktarıcı diferansiyel çıkış gerilimi	2 V min. $R_L = 100 \Omega$ 'da 1.5 V min. $R_L = 54 \Omega$ 'da
Sonlandırma ve polarlama	10K $\Omega$ ila +5 V, B üstünde, PROFIBUS Pin 3 10K $\Omega$ ila GND, A üstünde, PROFIBUS Pin 8 Dahili polarlama opsiyonları sağlanır veya dahili polarlama yok. Bütün durumlarda, harici sonlandırma gerekir, S7-1200 Sistem Kılavuzundaki RS485 devre konektörünün (Sayfa 670707) polarlanması ve sonlandırılması ile RS422 ve RS485 konfigürasyonuna bakınız (Sayfa 707707).
Alıcı giriş empedansı	5.4K $\Omega$ min. sonlandırma dahil
Alıcı eşik/duyarlılık	+/- 0.2 V min., 60 mV tipik histeresis

## Teknik özellikler

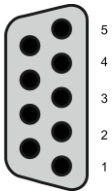
### A.14 Haberleşme ara yüzleri

Teknik veri	CM 1241 RS422/485
İzolasyon RS485 sinyali, şasi topraklamaya RS485 sinyali, CPU lojik ortak uca	500 VAC, 1 dakika
Kablo uzunluğu, zırlı	1000 m max. (baut hızı bağımlı)
Baut hızı bağımlı	300 baud, 600 baud, 1.2 kbit, 2.4 kbit, 4.8 kbit, 9.6 kbit (varsayılan), 19.2 kbit, 38.4 kbit, 57.6 kbit, 76.8 kbit, 115.2 kbit,
Parite	Parite yok (varsayılan), çift, tek, işaret (parite biti daima 1'e ayarlı), Ara çubuğu (parite biti daima 0'a ayarlı)
Stop bitlerinin numarası	1 (varsayılan), 2
Akış kontrol	RS422 modu için XON/XOFF desteklenir.
Bekleme süresi	0 - 65535 ms

### Çizelge A- 255 Güç kaynağı

Teknik veri	CM 1241 RS422/485
Güç kaybı (harcaması)	1.1 W
+5 VDC'den	220 mA

### Çizelge A- 256 RS485 veya RS422 konektör (dişi)

Pin	Açıklama	Konektör (dişi)	Pin	Açıklama
1	Lojik veya haberleşme topraklaması		6 PWR	+5 V, 100 ohm seri rezistör ile: Çıkış
2 TxD+ <sup>1</sup>	RS422 için bağlanır RS485 için kullanılmaz: Çıkış		7	Bağlı değil
3 TxD+	Sinyal A (RxD/TxD+): Giriş/Çıkış		8 TXD-	Sinyal A (RxD/TxD-): Giriş/Çıkış
4 RTS <sup>2</sup>	Çıkış gönderme (TTL seviyesi) sistemi		9 TXD- <sup>1</sup>	RS422 için bağlanır RS485 için kullanılmaz: Çıkış
5 GND	Lojik veya haberleşme topraklaması		SHELL	Şasi topraklama

<sup>1</sup> Pin 2 ve 9, RS422 için sadece aktarma sinyalleri olarak kullanılır.

<sup>2</sup> RTS bir TTL seviyesi sinyaldir ve bu sinyali esas alan diğer yar-m-dublex cihazın kontrolü için kullanılır. Aktarma yapıldığında aktif, diğer bütün zamanlarda pasiftir.

## A.15 TeleService (TS Adapter ve TS Adapter modular)

TS Adapter IE Basic ve TS Adapter modular için izleyen kılavuzlar teknik özellikleri içerir:

- Industrial Software Engineering Tools  
Modular TS Adapter
- Industrial Software Engineering Tools  
TS Adapter IE Basic

Bu ürün ve ürün dokümantasyonu hakkında daha fazla bilgi için web sitesinden ürün kataloğuna bakınız:

(<https://eb.automation.siemens.com/mall/en/de/Catalog/Search?searchTerm=TS%20Adapter%20IE%20basic&tab=>).

## A.16 SIMATIC bellek kartları

Sipariş numarası	Kapasite
6ES7 954-8LT02-0AA0	32 GB
6ES7 954-8LP02-0AA0	2 GB
6ES7 954-8LL02-0AA0	256 MB
6ES7 954-8LF02-0AA0	24 MB
6ES7 954-8LE02-0AA0	12 MB
6ES7 954-8LC02-0AA0	4 MB

## A.17 Giriş simülatörleri

Çizelge A- 257 Genel özellikler

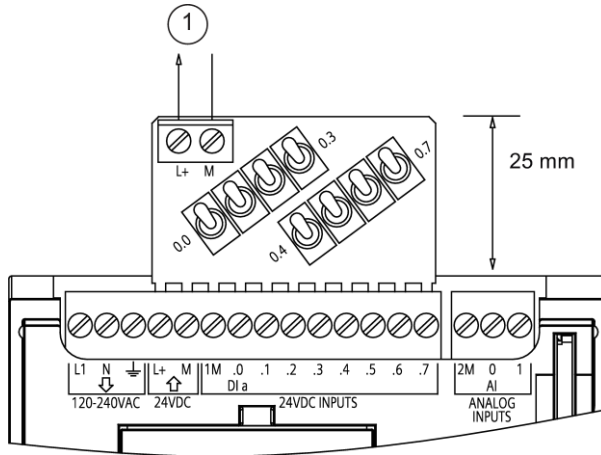
Teknik veri	8 Pozisyon Simülatörü	14 Pozisyon Simülatörü	CPU 1217C Simülatör
Sipariş numarası	6ES7 274-1XF30-0XA0	6ES7 274-1XH30-0XA0	6ES7 274-1XK30-0XA0
Boyutlar W x H x D (mm)	43 x 35 x 23	67 x 35 x 23	93 x 40 x 23
Ağırlık	20 gram	30 gram	43 gram
Noktalar	8	14	14
CPU ile kullanılan	CPU 1211C, CPU 1212C	CPU 1214C, CPU 1215C	CPU 1217C

**UYARI**

**Giriş simülatörlerinin güvenli kullanımı**

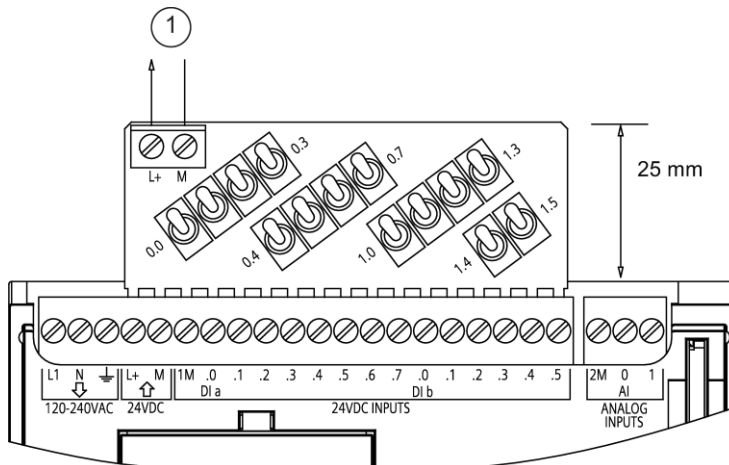
Bu giriş simülatörleri Sınıf I DIV 2 veya Sınıf I Bölge 2 tehlikeli yerlerde kullanmak için onaylanmamıştır. Anahtarlar, Sınıf I DIV 2 veya Sınıf I Bölge 2 tehlikeli yerlerde kullanılırsa, kıvılcım tehlikesi/patlama tehlikesi potansiyeli oluşturabilir. Onaylanmamış kullanım, personel ölümüne veya ciddi yaralanmasına ve/veya ekipmana zarara yol açabilir. Bu giriş simülatörlerini tehlikeli olmayan yerlerde kullanınız. Sınıf I DIV 2 veya Sınıf I Bölge 2 tehlikeli yerlerde kullanmayınız.

**8 Pozisyon Simülatörü (6ES7 274-1XF30-0XA0)**



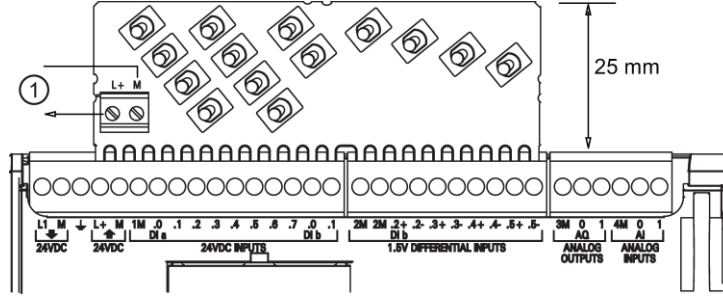
① 24 VDC sensör güç çıkışı

**14 Pozisyon Simülatörü (6ES7 274-1XF30-0XA0)**



① 24 VDC sensör güç çıkışı

CPU 1217C Simülator (6ES7 274-1XK30-0XA0)



① 24 VDC sensör güç çıkışı

## A.18 S7-1200 Potansiyometre modülü

S7-1200 Potansiyometre modülü S7-1200 CPU için bir donatıdır. Her potansiyometre konumuna orantılı olarak bir çıkış gerilimini CPU analog girişlerini 0 VDC - 10 VDC sürmek için oluşturur:

- 10.Devre kartı 'fingers'ı, S7-1200 CPU analog giriş terminal bloğuna takınız ve potansiyometre modülü üzerindeki 2-konum konektörüne bir harici DC güç kaynağı bağlayınız.
- 11.Bir küçük tornavidayı ayarlamalar yapmak için kullanınız: Gerilim çıkışını artırmak için tornavidayı saat yönünde (sağa) çeviriniz ve çıkış gerilimini azaltmak için tornavidayı saat yönünün tersine (sola) çeviriniz.

### Not

S7-1200 potansiyometre modülü ile işlem yaparken ESD kurallarını izleyiniz.

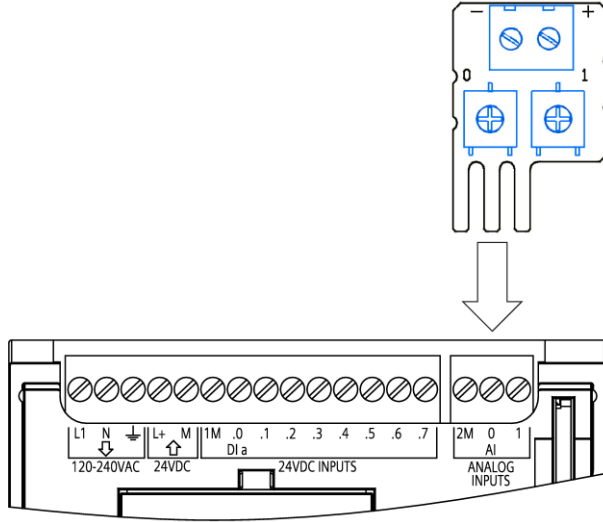
Teknik veri	Açıklama
Sipariş numarası	6ES7 274-1XA30-0XA0
CPU ile kullanılan	Bütün S7-1200 CPU'lar
Potansiyometre sayısı	2
Boyutlar W x H x D (mm)	20 x 33 x 14
Ağırlık	26 gram
2-pozisyonlu konektör <sup>1</sup> için kullanıcı-beslemeli gerilim (Sınıf 2, sınırlı güç veya PLC'den sensör gücü)	16.4 VDC - 28.8 VDC
Kablo uzunluğu (metre)/tip	<30 m, zırlı bükülü çift
Çekilen giriş akımı	10 mA max.
S7-1200 CPU Analog girişler <sup>1</sup> için potansiyometre gerilim çıkışı	0 VDC - 10.5 VDC min.

## Teknik özellikler

### A.19 I/O genişleme kablosu

Teknik veri	Açıklama
İzolasyon	İzole edilmemiş
Ortam sıcaklık aralığı	-20 °C - 60 °C

- <sup>1</sup> Potansiyometre modül çıkışı stabilitesi 2-konum konektöründeki kullanıcı tarafından sağlanan gerilim girişinin kalitesine bağlıdır-onu bir analog giriş gerilimi olarak dikkate alınız.

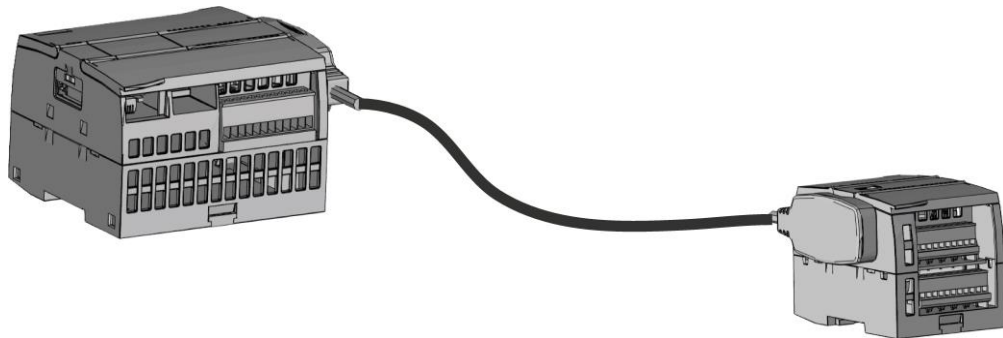


## A.19 I/O genişleme kablosu

Çizelge A- 258 Genişleme kabloları

Teknik veri	
Sipariş numarası	6ES7 290-6AA30-0XA0
Kablo uzunluğu	2 m
Ağırlık	200 g

S7-1200 genişleme kablosunun takılması ve çıkarılması hakkında bilgi için kurulum (montaj) bölümüne bakınız (Sayfa 6464).



## A.20 Eşlik eden ürünler

### A.20.1 PM 1207 güç modülü

PM 1207, SIMATIC S7-1200 için bir güç kaynağı modülüdür ve şu özelliklere sahiptir:

- Giriş 120/230 VAC, çıkış 24 VDC/2.5A
- Sipariş numarası 6ESP 332-1SH71-4AA0

Bu ürün ve ürün dokümantasyonu hakkında daha fazla bilgi için web sitesinde PM 1207 ürün kataloğuna bakınız

(<https://eb.automation.siemens.com/mall/en/de/Catalog/Product/6AG1332-1SH71-4AA0>).

### A.20.2 CSM 1277 kompakt anahtar modülü

CSM1277, bir endüstriyel Ethernet anahtar modülüdür. S7-1200 için Ethernet ara yüzünü çoğaltmak için kullanılabilir, böylece operatör paneli, programlama cihazları veya diğer konektörler ile aynı anda haberleşme sağlanır. Bu modül aşağıdaki özellikleri sağlar:

- 4 x RJ45 soket, endüstriyel Ethernet'e bağlantı için
- 3 kutuplu takmalı terminal ayar şeridi, harici 24 VDC kaynağın tepeye bağlantısı için
- LED'ler, endüstriyel Ethernet portlarının diyagnostik ve durum göstergesi
- Sipariş numarası 6GK7 277-1AA00-0AA0

Bu ürün ve ürün dokümantasyonu hakkında daha fazla bilgi için web sitesinde CSM 1277 ürün kataloğuna bakınız

(<https://eb.automation.siemens.com/mall/en/de/Catalog/Search?searchTerm=csm%201277&tab=>).

### A.20.3 CM CANopen modülü

CM CANopen modülü SIMATIC S7-1200 PLC ile CANopen'ı işleten her hangi bir cihaz arasında takılabilir bir modüldür. CM CANopen hem master hem de slave olacak şekilde konfigüre edilebilir. İki CM CANopen modülü vardır: CANopen modülü (sipariş numarası 021620-B) ve CANopen (sağlamlaştırılmış) modülü (sipariş numarası 021730-B).

CANopen modülü aşağıdaki özellikleri sağlar:

- CPU başına 3 modül bağlamaya imkan sağlar
- 16 CANopen slave düğümüne kadar bağlantı yapar
- Modül başına 256 bayt giriş ve 256 bayt çıkış
- 3 LED; modül, devre ve I/O durumları üzerine diyagnostik bilgi sağlar
- PLC içinde CANopen devre konfigürasyonunun depolanmasını sağlar.
- Modül, TIA Portal konfigürasyon takımının donanım kataloğuna entegre edilebilir.
- CANopen Configuration Studio (dahil) ile veya diğer bir harici CANopen konfigürasyon aracı ile CANopen konfigürasyonu
- CANopen haberleşme profilleri CiA 301 rev. 4.2 ve CiA 302 rev. 4.1 ile uyum sağlar
- Özel protokol yapmak için şeffaf CAN 2.0A'yı destekler.
- TIA portalında her bir PLC programlaması için önceden yapılmış bloklar vardır.
- CM CANopen modülü; alt devreler için vida terminalleri olan DSUB ile CM CANopen konfigürasyon stüdyo CD'si ve USB konfigürasyon kablosu içerir.

Bu ürün ve ürün dokümantasyonu hakkında daha fazla bilgi için web sitesinde CM CANopen ürün kataloğuna bakınız.



## Güç ihtiyacı hesabı

CPU, dahili bir güç kaynağına sahiptir. Bu güç kaynağı CPU'nun kendisinin, genişleme modüllerinin ve diğer 24 VDC kullanıcıların güç taleplerini karşılar.

Genişleme modülünün dört tipi vardır:

- Sinyal modülleri (SM), CPU'nun sağ tarafına takılır. Her CPU, güç ihtiyacını dikkate almaksızın mümkün olan en fazla sinyal modülüne izin verir.
  - CPU 1214C, CPU 1215C ve CPU 1217C, 8 sinyal modülüne izin verir.
  - CPU 1212C, 2 sinyal modülüne izin verir.
  - CPU 1211C, sinyal modülüne izin vermez.
- Haberleşme modülleri (CM) CPU'nun sol tarafına yerleştirilir. Bir CPU'ya, güç tüketimlerine bakılmaksızın en fazla 3 haberleşme modülünün takılmasına izin verilir.
- Sinyal kartları (SB), haberleşme kartları (CB) ve batarya kartları (BB) CPU'nun üst tarafına yerleştirilir. Bir CPU için en fazla bir sinyal kartına, haberleşme kartına veya batarya kartına izin verilir..

CPU'nun konfigürasyonunuz için ne kadar güç (veya akım) sağlayacağını belirlemek üzere bir yol gösterici olarak aşağıdaki bilgiyi kullanınız.

Her CPU hem 5 VDC hem de 24 VDC güç sağlar:

- Bir genişleme modülü bağlandığında, genişleme modülleri için CPU 5 VDC güç sağlar. Genişleme modülleri için 5 VDC güç ihtiyacı CPU güç kapasitesini aşarsa, genişleme modüllerini kaldırmaya CPU güç kapasitesi içinde kalıncaya kadar devam ediniz.
- Her CPU, lokal giriş noktaları veya genişleme modülleri üzerindeki röle bobinleri için 24 VDC sağlayan 24 VDC sensör kaynağına sahiptir. 24 VDC güç talebi CPU güç sağlama kapasitesinin üzerine çıkarsa, genişleme modülleri için harici bir 24 VDC güç kaynağı ekleyebilirsiniz. 24 VDC kaynağını giriş noktalarına veya röle bobinlerine el ile bağlayınız.

### UYARI

**DC sensör kaynağı ile harici 24 VDC güç kaynağını paralel bağlamak iki kaynak arasında bir uyumsuzlığa neden olabilir. Bu durumda her bir kaynak kendi çıkış gerilimi seviyesini üstün kılmaya çalışacaktır.**

Bu uyumsuzluğun sonucunda kaynaklardan birinin veya her ikisinin ömrü kısılacak veya anında arıza meydana gelecektir. Dolayısıyla PLC sisteminin öngörülemeyen çalışmasına yol açılabilecek ve ciddi personel yaralanması ve/veya malda zarar meydana gelebilecektir.

CPU üzerindeki sensör kaynağı ve harici güç kaynağı farklı noktalara güç sağlamalıdır. Ortak noktaların tek bir bağlantısına izin verilir.

PLC'de bazı 24 VDC güç giriş portları, çoklu M terminallerini bağlayan lojik ortak bir devreyle birbirleriyle bağlıdır. CPU 24 VDC güç kaynağı girişi, SM röle bobin güç giriş ve izolesiz analog güç kaynağı girişi bu devrelere örneklerdir. Bu devreler veri sayfalarında izolesiz olarak gösterilmişlerdir. Bütün izolesiz M terminalleri aynı harici referans potansiyeline bağlanmalıdır.

**! UYARI**

**İzolesiz M terminallerinin farklı referans potansiyellerine bağlanması istenmeyen akım akışlarına neden olabilir ve bunun sonucunda PLC ve bağlı ekipmanda zarar veya öngörülmeleyen çalışma oluşabilir.**

Böyle zarar veya öngörülemez çalışma ölüm, ciddi personel yaralanması ve/veya mala zarar ile sonuçlanabilir.

Bir PLC sisteminde bütün izolesiz M terminallerinin aynı referans potansiyeline bağlı olduklarından daima emin olunuz.

CPU'lar ve sinyal modüllerinin güç ihtiyaçlarının hesaplanması hakkında bilgi teknik özelliklerde verilmiştir (Sayfa 829).

**Not**

CPU'nun güç kapasitesinin aşılması CPU için izin verilen en fazla sayıdaki modülün bağlanamaması ile sonuçlanır.

**Örnek güç ihtiyacı hesabı**

İzleyen örnek bir konfigürasyon için güç ihtiyacının hesaplanmasına bir örnektir. Konfigürasyon; bir CPU 1214C AC/DC/Röle, bir SB 1223 2 x 24 VDC giriş/ 2 x 24 VDC çıkış, bir CM 124, üç SM 1223 8 DC In/8 Relay çıkış, bir SM 1221 8 DC In içermektedir. Bu örnek toplam 48 giriş ve 36 çıkışa sahiptir.

**Not**

CPU dahili röle bobinini sürmek için gerekli gücü önceden ayırmaktadır. Güç ihtiyacı hesabında dahili röle bobini güç ihtiyacını dahil etmenize gerek yoktur.

Bu örnekteki CPU SM'ler için yeterli 5 VDC akımı sağlar, ancak bütün girişler ve genişleme röle bobinleri için sensör kaynağından yeterli 24 VDC sağlamaz. I/O için 456 mA gerekir, CPU ancak sadece 400 mA sağlar. 24 VDC girişler ve çıkışların çalışması için 24 VDC'de en azından 56 mA ilave bir kaynağın tesisi gerekir.

Çizelge B- 1 Örnek güç ihtiyacı hesabı

CPU güç ihtiyacı hesabı	5 VDC	24 VDC
CPU 1214C AC/DC/Röle	1600 mA	400 mA
<i>Eksi</i>		
Sistem gerekleri	5 VDC	24 VDC
CPU 1214C, 14 giriş	-	14 * 4 mA = 56 mA
1 SB 1223 2 x 24 VDC Giriş/ 2 x 24 VDC Çıkış	50 mA	2 * 4 mA = 8 mA
1 CM 1241 RS422/485, 5 V güç	220 mA	
3 SM 1223, 5 V güç	3 * 145 mA = 435 mA	-
1 SM 1221, 5 V güç	1 * 105 mA = 105 mA	-
3 SM 1223, 8 giriş her birisi	-	3 * 8 * 4 mA = 96 mA
3 SM 1223, 8 Röle bobini her birisi	-	3 * 8 * 11 mA = 264 mA
1 SM 1221, 8 giriş her birisi	-	8 * 4 mA = 32 mA
<b>Toplam gerekleri</b>	810 mA	456 mA
<i>Eşittir</i>		
Akım balansı	5 VDC	24 VDC
Akım balansı toplam	790 mA	(56 mA)

### Güç ihtiyacı hesaplama formu

Konfigürasyonunuz için S7-1200 CPU'nun ne kadar güç (veya akım) sağlayabileceğini belirlemek için aşağıdaki çizelgeyi kullanınız. CPU modeliniz ve sinyal modülleriniz için güç ihtiyacını hesaplamak üzere teknik özelliklere (Sayfa 829) bakınız.

Çizelge B- 2 Güç ihtiyacı hesaplamaları

CPU güç ihtiyacı hesabı	5 VDC	24 VDC
<i>Eksi</i>		
Sistem gerekleri	5 VDC	24 VDC
<b>Toplam gerekleri</b>		
<i>Eşittir</i>		
Akım balansı	5 VDC	24 VDC
Akım balansı toplam		



## Sipariş numaraları

### C.1 CPU modelleri

Çizelge C- 1 S7-1200 CPU'lar

CPU modelleri		Sipariş numarası
CPU 1211C	CPU 1211C DC/DC/DC	6ES7 211-1AE40-0XB0
	CPU 1211C AC/DC/Röle	6ES7 211-1BE40-0XB0
	CPU 1211C DC/DC/Röle	6ES7 211-1HE40-0XB0
CPU 1212C	CPU 1212C DC/DC/DC	6ES7 212-1AE40-0XB0
	CPU 1212C AC/DC/Röle	6ES7 212-1BE40-0XB0
	CPU 1212C DC/DC/Röle	6ES7 212-1HE40-0XB0
CPU 1214C	CPU 1214C DC/DC/DC	6ES7 214-1AG40-0XB0
	CPU 1214C AC/DC/Röle	6ES7 214-1BG40-0XB0
	CPU 1214C DC/DC/Röle	6ES7 214-1HG40-0XB0
CPU 1215C	CPU 1215C DC/DC/DC	6ES7 215-1AG40-0XB0
	CPU 1215C AC/DC/Röle	6ES7 215-1BG40-0XB0
	CPU 1215C DC/DC/Röle	6ES7 215-1HG40-0XB0
CPU 1217C	CPU 1217C DC/DC/DC	6ES7 217-1AG40-0XB0

### C.2 Sinyal modülleri (SM'ler), sinyal kartları (SB'ler) ve batarya panelleri (BB)

Çizelge C- 2 Sinyal modülleri (SM'ler)

Sinyal modülleri		Sipariş numarası
Dijital giriş	SM 1221 8 x 24 VDC Giriş (Sink/Source)	6ES7 221-1BF32-0XB0
	SM 1221 16 x 24 VDC Giriş (Sink/Source)	6ES7 221-1BH32-0XB0
Dijital çıkış	SM 1222 8 x 24 VDC Çıkışı (Source)	6ES7 222-1BF32-0XB0
	SM 1222 16 x 24 VDC Çıkışı (Source)	6ES7 222-1BH32-0XB0
	SM 1222 8 x Röle Çıkışı	6ES7 222-1HF32-0XB0
	SM 1222 8 x Röle Çıkışı (Değiştirme)	6ES7 222-1XF32-0XB0
	SM 1222 16 x Röle Çıkışı	6ES7 222-1HH32-0XB0
Dijital giriş / çıkış	SM 1223 8 x 24 VDC Giriş (Sink/ Source) / 8 x 24 VDC Çıkışı (Source)	6ES7 223-1BH32-0XB0
	SM 1223 16 x 24 VDC Giriş (Sink/Source) / 16 x 24 VDC Çıkışı (Source)	6ES7 223-1BL32-0XB0
	SM 1223 8 x 24 VDC Giriş (Sink/ Source) / 8 x Röle Çıkışı	6ES7 223-1PH32-0XB0

Sipariş numaraları

C.2 Sinyal modülleri (SM'ler), sinyal kartları (SB'ler) ve batarya panelleri (BB)

Sinyal modülleri	Sipariş numarası	
SM 1223 16 x 24 VDC Giriş (Sink/ Source) / 16 x Röle Çıkışı	6ES7 223-1PL32-0XB0	
SM 1223 8 x 120/230 VAC Giriş (Sink/Source) / 8 x Röle Çıkışları	6ES7 223-1QH32-0XB0	
Analog giriş	SM 1231 4 x Analog Giriş	6ES7 231-4HD32-0XB0
	SM 1231 8 x Analog Giriş	6ES7 231-4HF32-0XB0
	SM 1231 4 x Analog Giriş x 16 bit (yüksek özellik)	6ES7 231-5ND32-0XB0
Analog çıkış	SM 1232 2 x Analog Çıkış	6ES7 232-4HB32-0XB0
	SM 1232 4 x Analog Çıkış	6ES7 232-4HD32-0XB0
Analog giriş / çıkış	SM 1234 4 x Analog Giriş / 2 x Analog çıkış	6ES7 234-4HE32-0XB0
RTD ve Termokupl	SM 1231 TC 4 x 16 bit	6ES7 231-5QD32-0XB0
	SM 1231 TC 8 x 16 bit	6ES7 231-5QF32-0XB0
	SM 1231 RTD 4 x 16 bit	6ES7 231-5PD32-0XB0
	SM 1231 RTD 8 x 16 bit	6ES7 231-5PF32-0XB0
Teknoloji modülleri	SM 1278 4xIO-Link Master	6ES7 278-4BD32-0XB0

Çizelge C- 3 Sinyal panelleri (SB) ve batarya panelleri (BB'ler)

Sinyal ve batarya panelleri	Sipariş numarası	
Dijital giriş	SB 1221 200 KHz 4 x 24 VDC Giriş (Source)	6ES7 221-3BD30-0XB0
	SB 1221 200 KHz 4 x 5 VDC Giriş (Source)	6ES7 221-3AD30-0XB0
Dijital çıkış	SB 1222 200 KHz 4 x 24 VDC Çıkış (Sink/ Source)	6ES7 222-1BD30-0XB0
	SB 1222 200 KHz 4 x 5 VDC Çıkış (Sink/ Source)	6ES7 222-1AD30-0XB0
Dijital giriş / çıkış	SB 1223 2 x 24 VDC Giriş (Sink) / 2 x 24 VDC Çıkışı (Source)	6ES7 223-0BD30-0XB0
	SB 1223 200 KHz 2 x 24 VDC Giriş (Source) / 2 x 24 VDC Çıkış (Sink/ Source)	6ES7 223-3BD30-0XB0
	SB 1223 200 KHz 2 x 5 VDC Giriş (Source) / 2 x 5 VDC Çıkış (Sink/ Source)	6ES7 223-3AD30-0XB0
Analog	SB 1232 1 Analog çıkış	6ES7 232-4HA30-0XB0
	SB 1231 1 Analog Giriş	6ES7 231-4HA30-0XB0
	SB 1231 1 Analog Giriş Termokupl	6ES7 231-5QA30-0XB0
	SB 1231 1 Analog Giriş RTD	6ES7 231-5PA30-0XB0
Batarya	BB 1297 Batarya Paneli (Batarya tipi CR1025 dahil değil)	6ES7 297-0AX30-0XA0

### C.3 Haberleşme

Çizelge C- 4 Haberleşme modülü (CM)

Haberleşme modülü (CM)			Sipariş numarası
RS232, RS422 ve RS485	CM 1241 RS232	RS232	6ES7 241-1AH32-0XB0
	CM 1241 RS422/485	RS422/485	6ES7 241-1CH32-0XB0
PROFIBUS	CM 1243-5	PROFIBUS Master	6GK7 243-5DX30-0XE0
	CM 1242-5	PROFIBUS Slave	6GK7 242-5DX30-0XE0
AS-i Master	CM 1243-2	AS-i Master	3RK7 243-2AA30-0XB0

Çizelge C- 5 Haberleşme paneli (CB)

Haberleşme paneli (CB)			Sipariş numarası
RS485	CB 1241 RS485	RS485	6ES7 241-1CH30-1XB0

Çizelge C- 6 Haberleşme işlemcisi (CP)

Haberleşme işlemcisi (CP)		Sipariş numarası
CP 1242-7	GPRS	6GK7 242-7KX30-0XE0

Çizelge C- 7 TeleService

TS Adaptör		Sipariş numarası
TS Adapter IE Basic		6ES7 972-0EB00-0XA0
TS Module GSM		6GK7 972-0MG00-0XA0
TS Module RS232		6ES7 972-0MS00-0XA0
TS Module Modem		6ES7 972-0MM00-0XA0
TS Module ISDN		6ES7 972-0MD00-0XA0

Çizelge C- 8 Donatılar

Donatı			Sipariş numarası
Anten	ANT794-4MR	GSM/GPRS anten	6NH9 860-1AA00
	ANT794-3M	Düz anten	6NH9 870-1AA00

Çizelge C- 9 Konektörler

Konektör tipi		Sipariş numarası
RS485	35-derece kablo çıkışı, vida-terminal bağlantısı	6ES7 972-0BA42-0XA0
	35-derece kablo çıkışı, hızlı bağlantı	6ES7 972-0BA60-0XA0

Sipariş numaraları

C.4 Diğer modüller

## C.4 Diğer modüller

Çizelge C- 10 Eşlik eden ürünler

Öğe		Sipariş numarası
Güç kaynağı	PM 1207 güç kaynağı	6EP1 332-1SH71-4AA0
Ethernet anahtar	CSM 1277 Ethernet anahtar - 4 port	6GK7 277-1AA10-0AA0
CM CANopen	CANopen, SIMATIC S7-1200 için	021620-B
	CANopen (Sağlamlaştırılmış), SIMATIC S7-1200 için	021730-B

## C.5 Bellek kartları

Çizelge C- 11 Bellek kartları

SIMATIC bellek kartları	Sipariş numarası
SIMATIC MC 4 MB	6ES7 954-8LC02-0AA0
SIMATIC MC 12 MB	6ES7 954-8LE02-0AA0
SIMATIC MC 24 MB	6ES7 954-8LF02-0AA0
SIMATIC MC 256 MB	6ES7 954-8LL02-0AA0
SIMATIC MC 2 GB	6ES7 954-8LP01-0AA0
SIMATIC MC 32 GB	6ES7 954-8LT02-0AA0

## C.6 Temel HMI cihazları

Çizelge C- 12 HMI cihazları

HMI Temel Paneller	Sipariş numarası
KTP400 Basic (Mono, PN)	6AV6 647-0AA11-3AX0
KTP600 Basic (Mono, PN)	6AV6 647-0AB11-3AX0
KTP600 Basic (Renkli, PN)	6AV6 647-0AD11-3AX0
KTP1000 Basic (Renkli, PN)	6AV6 647-0AF11-3AX0
TP1500 Basic (Renkli, PN)	6AV6 647-0AG11-3AX0



## C.7 Yedek parçalar ve diğer donanım

Çizelge C- 13 Genişleme kabloları, simülatörler, terminal blokları ve son tutucular

Eleman		Sipariş numarası	
I/O genişleme kablosu	I/O Genişleme kablosu, 2 m	6ES7 290-6AA30-0XA0	
I/O simülatör	Simülatör (1211C/1212C - 8 pozisyon)	6ES7 274-1XF30-0XA0	
	Simülatör (1214C/1215C - 14 pozisyon)	6ES7 274-1XH30-0XA0	
	Simülatör, CPU 1217C	6ES7 274-1XK30-0XA0	
Potansiyometre modülü	S7-1200 Potansiyometre modülü	6ES7 274-1XA30-0XA0	
Yedek kapı donatısı	CPU 1211C/1212C	6ES7 291-1AA30-0XA0	
	CPU 1214C	6ES7 291-1AB30-0XA0	
	CPU 1215C	6ES7 291-1AC30-0XA0	
	CPU 1217C	6ES7 291-1AD30-0XA0	
	Sinyal modülü, 45 mm	6ES7 291-1BA30-0XA0	
	Sinyal modülü, 70 mm	6ES7 291-1BB30-0XA0	
	Haberleşme modülü	6ES7 291-1CC30-0XA0	
Terminal blok	Tin	7 terminal, 4/pk	6ES7 292-1AG30-0XA0
		8 terminal, 4/pk	6ES7 292-1AH30-0XA0
		10 terminal, 4/pk	6ES7 292-1AK30-0XA0
		11 terminal, 4/pk	6ES7 292-1AL30-0XA0
		12 terminal, 4/pk	6ES7 292-1AM30-0XA0
		14 terminal, 4/pk	6ES7 292-1AP30-0XA0
		16 terminal, 4/pk	6ES7 292-1AR30-0XA0
		18 terminal, 4/pk	6ES7 292-1AT30-0XA0
	Altın	3 terminal, 4/pk (analog CPU için)	6ES7 292-1BC30-0XA0
		6 terminal, 4/pk (analog CPU için)	6ES7 292-1BF30-0XB0
		6 terminal, 4/pk (sinyal kartı için)	6ES7 292-1BF30-0XA0
		7 terminal, 4/pk (analog sinyal modülü için)	6ES7 292-1BG30-0XA0
		11 terminal, 4/pk (analog sinyal modülü için)	6ES7 292-1BL30-0XA0
	Tuş takımı sol, 7 kontak	6ES7 292-1AG40-0XA1	
	Tuş takımı sağ, 7 kontak	6ES7 292-1AG40-0XA0	
	Tuş takımı sağ, 8 kontak	6ES7 292-1AH40-0XA0	
	Tuş takımı sağ, 11 kontak	6ES7 292-1AL40-0XA0	
	Tuş takımı sağ, 12 kontak	6ES7 292-1AM40-0XA0	
	Tuş takımı sağ, 14 kontak	6ES7 292-1AP40-0XA0	
	Tuş takımı sağ, 20 kontak	6ES7 292-1AV40-0XA0	
Son tutucu	Son tutucu Termoplastik, 10 MM	8WA1808	
	Son tutucu çelik, 10.3 MM	8WA1805	

Sipariş numaraları

C.8 Programlama yazılımı

## C.8 Programlama yazılımı

Çizelge C- 14 Programlama yazılımı

SIMATIC yazılımı		Sipariş numarası
Programlama yazılımı	STEP 7 Basic V13	6ES7 822-0AA01-0YA0
	STEP 7 Professional V13	6ES7 822-1AA01-0YA5
Görselleştirme yazılımı	WinCC Basic V12 SP1	6AV2100-0AA01-0AA0
	WinCC Comfort V12 SP1	6AV2101-0AA01-0AA5
	WinCC Advanced V12 SP1	6AV2102-0AA01-0AA5
	WinCC Professional 512 PowerTags V12 SP1	6AV2103-0DA01-0AA5
	WinCC Professional 4096 PowerTags V12 SP1	6AV2103-0HA01-0AA5
	WinCC Professional max. PowerTags V12 SP1	6AV2103-0XA01-0AA5

## C.9 Dokümantasyon

Çizelge C- 15 S7-1200 dokümantasyon

Yazılı doküman	Dil	Sipariş numarası
S7-1200 Programlanabilir kontrolör sistem kılavuzu	Almanca	6ES7 298-8FA30-8AH0
	İngilizce	6ES7 298-8FA30-8BH0
	Fransızca	6ES7 298-8FA30-8CH0
	İspanyolca	6ES7 298-8FA30-8DH0
	İtalyanca	6ES7 298-8FA30-8EH0
	Çince	6ES7 298-8FA30-8KH0
S7-1200 Easy Book	Almanca	6ES7 298-8FA30-8AQ0
	İngilizce	6ES7 298-8FA30-8BQ0
	Fransızca	6ES7 298-8FA30-8CQ0
	İspanyolca	6ES7 298-8FA30-8DQ0
	İtalyanca	6ES7 298-8FA30-8EQ0
	Çince	6ES7 298-8FA30-8KQ0

## Bir V3.0 CPU'yu bir V4.0 CPU ile deęiřtirme

V3.0 CPU'nuzu V4.0 CPU (Sayfa 142) ile deęiřtirebilir ve V3.0 CPU için tasarladığınız mevcut STEP 7 projenizi kullanabilirsiniz. Baęlı modülünüz için donanın yazılımı güncellemelerini (Sayfa 133) kontrol edebilir ve uygulayabilirsiniz.

### Not

#### V4.0'den V3.0'e cihaz deęiřimi mümkün deęildir

V3.0 CPU'nuzu V4.0 CPU (Sayfa 3) ile deęiřtirebilirsiniz, ancak konfigürasyonu yükledikten sonra V3.0 CPU'nuzu V4.0 CPU ile deęiřtiremezsiniz. Mevcut STEP 7 V3.0 projenizi gözden geçirmek veya kullanmak isterseniz, cihaz deęiřimi yapmadan önce STEP 7 V3.0 projenizin bir arřivini oluřturunuz.

Deęiřim yapılan cihazın konfigürasyonunu henüz yüklediyseniz, onu geri alabilirsiniz. Yüklemeden sonra, V3.0'ten V4.0'e deęiřiklięi geri alamazsınız.

CPU versiyonları arasındaki bazı konfigürasyon ve operasyonel deęiřiklikler konusunda dikkatli olmalısınız.

### Organizasyon blokları

V4.0 ile, OB yürütmelerinin kesilebilir veya kesilemez (non-interruptible) olmasını konfigüre edebilirsiniz (Sayfa 94). Eski V3.0 CPU'lardan kalan projeler için STEP 7 bütün OB'leri kesilemez biçiminde varsayılan olarak ayarlar.

STEP 7 bütün OB'lerin önceliklerini (Sayfa 94), V3.0 CPU STEP 7 projesindeki deęerlerine ayarlar.

Seęerseniz, kesilebilirlięi veya öncelik ayarlarını sonradan deęiřtirebilirsiniz.

Diyagnostik hata interrupt'ı (OB) (Sayfa 89) başlama bilgisi, beklemede diyagnostik olayı yok ise, bir bütün olarak alt modüle iliřkindir.

### CPU řifre koruması

STEP 7 V4.0 CPU için bir řifre koruma seviyesini (Sayfa 190) V3.0 CPU için ayarlanan řifre koruma seviyesine eřdeęer olacak şekilde ayarlar.. STEP 7, V3.0 řifresini V4.0 CPU'ya "Full access (koruma yok)" řifresi olarak atar:

V3.0 koruma seviyesi	V4.0 eriřim seviyesi
Koruma yok	Tam eriřim(koruma yok)
Yazma koruması	Okuma eriřimi
Yazma/okuma koruması	HMI eriřimi

V4.0'deki "No access (tam koruma)" eriřim seviyesinin V3.0'da bulunmadığına dikkat ediniz.

## Web sunucu

3.0 CPU'nuzu V4.0 CPU ile deęiřtirirseniz, Web sunucuyu aktifleřtirmek iin Web sunucunuz proje ayarlaması (Sayfa 603) ve HTTPS eriřimi (gerekir veya gerekmez) V3.0'da nceden olduęu gibi aynı olacaktır. Sonra, Web sunucuyu kullanmak iin gerek duyulan kullanıcılar, ayrıcalıklar, řifreler (Sayfa 604) ve dilleri (Sayfa 603) konfigre edebilirsiniz. Kullanıcıları ilave ayrıcalıklar ile konfigre etmezseniz, sonra Standart Web sayfasından grebileceęiniz kadara sınırlandırılırsınız (Sayfa 609). S7-1200 V4.0 CPU nceden konfigre edilmiř eski "admin" kullanıcıyı ve řifreyi desteklemez.

nceleri, veri ktę Web sunucu sayfası bir "Download and Clear" operasyonu saęladı. Veri ktklerine ulařtıęınız V4.0 Web sunucu dosya tarayıcı (Sayfa 622) bu zellięi artık saęlayamaz. Bunun yerine, Web sunucu veri ktk dosyalarını ykleme, yeniden isimlendirme ve silme imkanı saęlar.

## Transfer kartı uyumsuzluęu

Bir V3.0 transfer kartını (Sayfa 125) bir V3.0 programını bir V4.0'de transfer iin kullanamazsınız. V3.0 projesini STEP 7'de amalı, cihazı V4.0 CPU'ya deęiřtirmeli ve STEP 7 projesini sizin V4.0 CPU'nuz transfer etmelisiniz (Sayfa 142). Projenizi V4.0 projesine deęiřtirdikten sonra sonraki program transferleri iin bir V4.0 transfer kartı yapabilirsiniz.

## GET/PUT haberleřme

Varsayılan olarak, GET/PUT haberleřmesi V3.0'de etkin durumdaydı. V3.0 CPU'yu V4.0 CPU ile deęiřtirdięinizde (Sayfa 142), uygunluk bilgi blmesinde GET/PUT'un etkin olduęunu belirten bir mesajı grebilirsiniz.

## Hareket kontrol desteęi

V1.0 ve V2.0 hareket ktphanelerini S7-1200 V4.0 CPU'lar desteklemez. V1.0 ve V2.0 hareket ktphaneleri olan bir STEP 7 projesi iin bir cihaz deęiřiklięi yaparsanız, cihaz deęiřimi derlemede V1.0 veya V2.0 hareket ktphane komutları yerine uyumlu olan V3.0 hareket kontrol komutlarını (Sayfa 455) yerleřtirir.

Farklı hareket komutları versiyonları (V3.0 ve V4.0) olan bir STEP 7 projesi iin bir cihaz deęiřiklięi yaparsanız, cihaz deęiřimi derlemede V4.0 hareket ktphane komutlarını yedekler.

V3.0'dan V4.0'a cihaz deęiřimi sırasında, hareket kontrol teknolojik nesnesi (TO) versiyonu otomatik olarak V4.0'e deęiřir. İlgili btn bloklar da deęiřir. Blok ara yzleri izleme izelgeleri ve izler yeni V4.0 versiyonuna gncellenir. İzleyen izelgede V3.0 ve V4.0 hareket kontrol eksen parametreleri arasındaki farkları bulabilirsiniz:

V3.0	V4.0
Config.General.LengthUnit	Units.LengthUnit
Config.Mechanics.PulsesPerDriveRevolution	Actor.DriveParameter.PulsesPerDriveRevolution
Config.Mechanics.LeadScrew	Mechanics.LeadScrew
Config.Mechanics.InverseDirection	Actor.InverseDirection
Config.DynamicLimits.MinVelocity	DynamicLimits.MinVelocity

V3.0	V4.0
Config.DynamicLimits.MaxVelocity	DynamicLimits.MaxVelocity
Config.DynamicDefaults.Acceleration	DynamicDefaults.Acceleration
Config.DynamicDefaults.Deceleration	DynamicDefaults.Deceleration
Config.DynamicDefaults.EmergencyDeceleration	DynamicDefaults.EmergencyDeceleration
Config.DynamicDefaults.Jerk	DynamicDefaults.Jerk
Config.PositionLimits_SW.Active	PositionLimitsSW.Active
Config.PositionLimits_SW.MinPosition	PositionLimitsSW.MinPosition
Config.PositionLimits_SW.MaxPosition	PositionLimitsSW.MaxPosition
Config.PositionLimits_HW.Active	PositionLimitsHW.Active
Config.PositionLimits_HW.MinSwitchedLevel	PositionLimitsHW.MinSwitchLevel
Config.PositionLimits_HW.MaxSwitchedLevel	PositionLimitsHW.MaxSwitchLevel
Config.Homing.AutoReversal	Homing.AutoReversal
Config.Homing.Direction	Homing.ApproachDirection
Config.Homing.SideActiveHoming	Sensör[1].ActiveHoming.SidelInput
Config.Homing.SidePassiveHoming	Sensör[1].PassiveHoming.SidelInput
Config.Homing.Offset	Sensör[1].ActiveHoming.HomePositionOffset
Config.Homing.FastVelocity	Homing.ApproachVelocity
Config.Homing.SlowVelocity	Homing.ReferencingVelocity
MotionStatus.Position	Position
MotionStatus.Velocity	Velocity
MotionStatus.Distance	StatusPositioning.Distance
MotionStatus.TargetPosition	StatusPositioning.TargetPosition
StatusBits.SpeedCommand	StatusBits.VelocityCommand
StatusBits.Homing	StatusBits.HomingCommand

Sadece "komut çizelgesi" nin yeniden isimlendirilmiş parametresi, komutlara sahip dizindir:

V3.0	V4.0
Config.Command[]	Command[]

Not: Dizin "Command[]" V3.0'da "TO\_CmdTab\_Config\_Command" ve V4.0'da "TO\_Struct\_Command" bir UDT tipidir.

## Komut deęiřiklikleri

Parametreler veya davranışlarda yer alan ařağıdaki komutlarda deęiřiklikler vardır:

- RDREC ve WRREC (Sayfa 315 315)
- CONV (Sayfa 244244)

## **HMI panel haberleřmesi**

S7-1200 V3.0 CPU'nuza baęlı bir veya daha fazla HMI paneline(Sayfa 32) sahip olduysanız, S7-1200 V4.0 CPU ile haberleřme HMI panelinin donanım yazılımı versiyonuna ve kullandıęınız haberleřme tipine baęlıdır. Projenizi yeniden derleyiniz ve CPU ve HMI'ye yükleyiniz ve/veya HMI donanım yazılımınız güncelleyiniz.

## **Program bloklarını derlemek için kurallar**

Bir V3.0 CPU'yu bir V4.0 CPU ile deęiřtirdięinizde, bütün program bloklarını V4.0 CPU'ya yüklemeyen önce yeniden derlemeniz gerekir. İlaveten her hangi bir blok uzman korumasına (Sayfa 193) veya PLC seri numarasına (Sayfa 195) baęlantılı kopyalama korumasına sahip ise, blokları derlemeden ve yüklemeyen önce korumayı kaldırmalısınız. (Bir bellek kartına baęlantılı kopyalama korumasının etkinlięini kaldırmanıza gerek yoktur). Yüklemeden sonra, uzman korumasını ve/veya PLC seri numarası kopyalama korumasını konfigüre edebilirsiniz. Projeniz OEM (Original Equipment Manufacturer) tarafından saęlanan uzman koruması olan blokları içeriyorsa, bu bloklar için V4.0 versiyonunu saęlaması için OEM ile temasa geçmelisiniz.

Genellikle Siemens, cihaz deęiřtirdikten sonra donanım konfigürasyonunu ve STEP 7'deki yazılımı yeniden derlemenizi ve projenizdeki bütün cihazlara yüklemenizi önerir.

## Dizin

### &

& kutu (FBD AND lojik operasyonu), 203

### (

-( )- (atama), 203  
 -( N )- (negatif sinyal kenarındaki operand ayarı, 209  
 -( P )- (pozitif sinyal kenarındaki set operandı), 209  
 -( / )- (atama retti), 203  
 -(R) (reset çıkışı), 206  
 -(RESET\_BF) (reset bit alanı), 207  
 -(S) (ayar çıkışı), 206  
 -(SET\_BF) (ayar bit alanı), 207

### /

/= kutu (FBD atama reddi), 203

### |

-|/|- (normalde kapalı kontak), 203  
 -||- (normalde açık kontak), 203  
 -|N|- (negatif sinyal için tarama operandı), 209  
 -|NOT|- (RLO'yu evirme), 203  
 -|P|- (pozitif sinyal için tarama operandı), 209

### =

= kutu (FBD atama), 203

### >

>=1 kutu (FBD VEYA lojik operasyon), 203

## A

ABS (biçim mutlak değeri), 233

AC

topraklama, 72  
 izolasyon kılavuzu, 71  
 kablaj kılavuzu, 70, 72

Erişim koruması, CPU, 190

Erişim

PC'den veri günlükleri, , 623

Kullanıcı-tanımlı Web Sayfaları, 644

ACOS (arccosine değeri biçimi), 236

ACT\_TINT (günün saati interrupt'ını aktif etme), 339

Aktif/pasif haberleşme

konfigüre edilme, 150, 593

bağlantı ID'ler, 514

parametreler, 153

Aktif/pasif bağlantı, 512

Ad hoc modu, TCP ve ISO on TCP, 518

ADD (topla), 230

Yeni cihaz ekleme

CPU, 138

mevcut donanımı algıla, 140

tanımsız CPU, 140

Ekleme komutları, LAD veya FBD'ye girişler veya çıkışlar için, 41

Adresleme

Boolean veya bit değerleri, 106

münferit girişler (I) veya çıkışlar (Q), 106

bellek alanları, 105

proses görüntüsü, 105

Hava akışı, 50

Takma adlar, kullanıcı-tanımlı Web sayfasında, 634

Analog I/O

konfigürasyon, 147

mühendislik birimine dönüştürme, 3939, 110, 253

giriş temsili (akım), 917, 951

giriş temsili (gerilim), 916, 950

çıkış temsili (akım), 918, 952

çıkış temsili (gerilim), 917, 951

durum gösterciler, 785

basamak tepki süreleri (CPU), 842, 852, 862, 873, 888

basamak tepki süreleri (SB), 950

basamak tepki süreleri (SM), 916

Analog sinyal kartları

SB 1231

SB 1231 RTD, 957

SB 1231 Termokupl, 953

SB 1232, 948

Analog sinyal modülleri

SM 1231, 907

SM 1231 RTD, 925

SM 1231 Termokupl, 919

SM 1232, 910

SM 1234, 913

AND (lojik operasyon), 275

Onaylar

ATEX, 830

CE, 829  
C-Tick, 831  
cULus, 830  
FM, 830  
Korea Certification, 831  
Maritime, 831  
Diziler, üyelere erişim, 241  
AS-i  
AS-i master CM1243-2 modülü ekleme, 577  
AS-i slave ekleme, 577  
adres, 579  
AS-i master CM 1243-2, 576  
dağıtılmış I/O komutları, 314  
ağ bağlantısı, 578  
RDREC (veri kaydı okuma), 315  
STEP 7 ile slave konfigürasyonu, 583  
STEP 7 olmadan slave konfigürasyonu, 582  
sistem ataması, 582  
slave adreslerin sistem ataması, 582  
analog değerlerin transferi, 583  
dijital değerlerin transferi, 583  
WRREC (veri kaydı yazma), 315  
ASIN (arcsine değeri biçimi), 236  
Enum tiplerin ataması, kullanıcı-tanımlı Web Sayfaları, 635  
ATEX onayı, 830  
ATH (ASCII string'i hexadecimal sayıya dönüştürme), 304  
ATTACH (bir interrupt olayına bir OB ekleme), 331  
AWP komutları, 627  
tanımların birleştirilmesi, 638  
bir enum tipin tanımı, 635  
fragmentların üretilmesi, 637  
fragmentların alınması, 638  
özel değişkenlerin okunması, 631  
bir enum tipin referanslanması, 635  
bir takma adın kullanımı, 634  
özel değişkenlerin yazılması, 633  
değişkenleri yazma, 630  
AWP\_Enum\_Def, 635  
AWP\_Enum\_Ref, 635  
AWP\_Import\_Fragment, 638  
AWP\_In\_Variable, 629, 633  
AWP\_Out\_Variable, 631  
AWP\_Start\_Fragment, 637

**B**

Basic paneller (HMI), 32  
Batarya kartı (BB)  
BB 1297, 962  
batarya takma, 963  
Baud hızı, 673

BB 1297, 962  
Bir CPU veya bellek kartına bağlama, 195  
Bit lojik, 203,206,209  
Bloklar  
blok çağrılmaları, 77  
SCL ile bir FB veya FC çağırma, 183  
uyum kontrolü, 201  
çevrim içi bir CPU'dan blokların kopyalanması, 197  
sayıcılar (miktar ve bellek ihtiyacı), 25, 839, 849, 859, 869, 881  
veri bloğu(DB), 77  
yükleme, 196  
olaylar, 94  
fonksiyon (FC), 77, 174  
fonksiyon bloğu (FB), 77, 175  
bir FB'nin başlangıç değeri, 175  
instance veri bloğu(DB), 175  
interrupt'lar, 25, 94, 838, 848, 858, 869, 881  
doğrusal ve yapışlı programlar, 170  
izleme, 25, 838, 848, 858, 869, 881  
gömme derinliği, 25, 77, 838, 848, 858, 869, 881  
kod bloklarının sayısı, 25, 838, 848, 858, 869, 880  
OB'lerin sayısı, 25, 94, 838, 848, 858, 869, 881  
organizasyon blokları (OB'ler), 25, 77, 85, 94, 838, 848, 858, 869, 881  
şifre koruması, 193  
tek instance veya çoklu-instance DB, 175  
kullanıcı programının boyutu, 25, 77, 838, 848, 858, 869, 880  
başlangıç OB'leri, 94  
zamanlayıcılar (miktar ve bellek ihtiyacı), 25, 839, 849, 859, 869, 881  
kod bloklarının tipleri, 77  
geçerli FC, FB ve DB sayıları, 77  
Boolean veya bit değerleri, 106  
Kesme, 676676, 677  
BUFFER parametresi, SEND\_PTP, 696  
Bus konektör, 29

**C**

Kablo  
uzatma,  
Ağ haberleşme, 670  
CALCULATE (hesapla), 229  
analogları ölçekleme, 39  
kompleks eşitlikler için kullanma, 39  
Takvim, 283  
Çağırma yapısı, 201  
Kullanıcı programı içerisinde kod bloklarının çağırılması, 172  
CAN\_DINT (süre geciktirme interrupt'larını iptal etme), 342



- CAN\_TINT (günün saati interrupt'larını iptal etme), 339
- CANopen modülleri  
021620-B, 021630-B, 983
- Bir çevrim içi DB'den değerlerin yakalanması, 797
- CB 1241  
sonlandırma ve ön gerilim, 671
- CB 1241 RS485, 974
- CE onayı, 829
- CEIL (kayan-nokta sayıdan sonraki en yüksek tamsayıyı üretme), 250
- Certificate Import Wizard, 666
- Cihaz değiştirme, 142  
STEP 7 için ayarları değiştirme, 43
- Char (karakter veri tipi), 115
- Karakter pozisyon, mesaj uzunluğu, 682
- Karakter sırası  
mesaj sonu, 681  
mesaj başlangıcı, 677
- Chars\_TO\_Strg (CHAR dizinini karakter string'ine dönüştürme), 302
- Açıklık, hava akış ve soğutma, 50
- Saat  
RD\_LOC\_T (yerel saati okuma), 287  
RD\_SYS\_T (günün saatini okuma), 287  
Günün-saati için saat, 104  
WR\_LOC\_T (yerel saati ayarla), 287  
WR\_SYS\_T (günün saatini ayarla), 287
- Saat bellek baytı, 103
- Kod bloğu  
bir CPU veya bellek kartına bağlama, 195  
blok çağrılmaları, 77  
kullanıcı programı içerisinde kod bloklarının çağrılması, 172  
kopya koruması, 195  
sayıcılar (miktar ve bellek ihtiyacı), 25, 839, 849, 859, 869, 881  
DB (veri bloğu), 77, 176  
FB (fonksiyon bloğu), 77, 175  
FC (fonksiyon), 77, 174  
bir FB'nin başlangıç değeri, 175  
instance veri bloğu(DB), 175  
interrupt'lar, 25, 838, 848, 858, 869, 881  
know-how koruma, 193  
doğrusal ve yapılandırılmış programlar, 170  
izleme, 25, 838, 848, 858, 869, 881  
gömme derinliği, 25, 838, 848, 858, 869, 881  
kod bloklarının sayısı, 25, 838, 848, 858, 869, 880  
OB'lerin sayısı, 25, 838, 848, 858, 869, 881  
organizasyon blokları (OB'ler), 25, 172, 838, 848, 858, 869, 881
- kullanıcı programının boyutu, 25, 838, 848, 858, 869, 880  
zamanlayıcılar (miktar ve bellek ihtiyacı), 25, 839, 849, 859, 869, 881  
kod bloklarının tipleri, 77  
geçerli FC, FB ve DB sayıları, 77
- Soğuk birleşim dengelemesi, Termokupl, 922, 955
- Görev kartlarında sütunlar ve başlıklar, 42
- Haberleşme  
aktif/pasif, 150, 153, 593  
AS-i adresi, 579  
haberleşme yükü, 98  
konfigürasyon, 150, 153, 593  
bağlantı ID'leri, 514  
döngü süresi, 98  
akış kontrol, 674  
donanım bağlantısı, 550  
IP adresi, 159  
MAC adresi, 159  
ağ, 550  
ağ bağlantısı, 149  
bağlantı sayısı (PROFINET), 511  
parametreler, 153  
yoklama yapısı, 703  
PROFIBUS adres, 574  
PROFINET ve PROFIBUS, 509  
Gönderme ve alma parametreleri, 675  
TCON\_Param, 153  
zaman senkronizasyon özelliği (PROFINET), 165
- Haberleşme kartı (CB)  
modüller ekleme, 141  
CB 1241 RS485, 974  
kıyaslama çizelgesi, 26  
parametrelerin konfigürasyonu, 147  
cihaz konfigürasyonu, 137  
kurulum, 58  
LED göstericiler, 669, 783  
genel bakış, 29  
programlama, 702  
sökme, 58  
RS485, 669
- Haberleşme ara yüzleri  
modüller ekleme, 141  
CB 1241 RS485, 974  
CM 1241 RS232, 975  
modüllerin kıyaslama çizelgesi, 26  
konfigürasyon, 672672  
cihaz konfigürasyonu, 137  
LED göstericiler, 783  
programlama, 702  
RS232 ve RS485, 669

- Haberleşme modülü (CM)  
AS-i master CM1243-2 modülü ekleme, 577  
CM 1243-5 (DP master) modülü ekleme, 572  
modüller ekleme, 141  
CM 1241 RS232, 975  
CM 1241 RS422/RS485, 976  
kıyaslama çizelgesi, 26  
PtP örnek program için konfigürasyon, 705705  
parametrelerin konfigürasyonu, 147  
veri alma, 697696  
cihaz konfigürasyonu, 137  
kurulum, 62  
LED göstericiler, 669, 783  
genel bakış, 29  
güç ihtiyacı, 985  
programlama, 702  
sökme, 62  
RS232 ve RS485, 669
- Haberleşme işlemcisi (CP)  
modüller ekleme, 141  
kıyaslama çizelgesi, 26  
parametrelerin konfigürasyonu, 147  
cihaz konfigürasyonu, 137  
genel bakış, 29
- Haberleşme standart Web sayfası, 620
- Kompakt anahtar modülü  
CSM 1277, 982
- Değerleri kıyasla, 226
- Kıyaslama ve senkronizasyon çevrim içi/dışı CPU'lar, 794794
- Kod bloklarını kıyaslama, 794
- Kıyaslama çizelgesi  
CPU modelleri, 24  
HMI cihazları, 32  
modüller, 26
- Bilgisayar gereksinimleri, 36
- CONCAT (karakter string'lerini birleştirme), 307
- Konfigürasyon  
modüller ekleme, 141  
AS-i, 579  
AS-i port, 578  
haberleşme ara yüzleri, 672  
haberleşme yükü, 98  
CPU parametreleri, 142  
döngü süresi, 97  
keşfetmek, 140  
yükleme, 196  
Ethernet portu, 159  
HSC (yüksek-hızlı sayıcı), 418  
IP adresi, 159  
MAC adresi, 159  
modüller, 147  
ağ bağlantısı, 149  
PLC- PLC haberleşmesi, 553  
portlar, 672  
PROFIBUS, 574  
PROFIBUS adresi, 574  
PROFINET, 159  
mesaj alma, 679  
RS422, çalışma modları, 707  
RS485 çalışma modları, 709  
başlangıç parametreleri, 127  
zaman senkronizasyon özelliği (PROFINET), 165
- Konfigürasyon, aktarılan mesajın, 675
- Konfigürasyon, kullanıcı-tanımlı Web Sayfaları  
çoklu dillerin ayarı, 659  
STEP 7 konfigürasyon, 641
- Bağlantı kontakları  
Maksimum akım taşıma kapasitesi, 972
- Bağlantılar  
konfigürasyon, 153  
bağlantı ID'leri, 514  
Ethernet protokolleri, 591  
bağlantı sayısı (PROFINET), 165  
partnerler, 150, 593  
S7 bağlantısı, 591  
haberleşme tipleri, 509  
tipler, çoklu-düğümlü bağlantılar, 591  
Web sunucu, 664
- Bağlantı, kurulum ve sökme, 63
- Uyum kontrolü, 201
- Sınırlamalar  
kullanıcı-tanımlı Web Sayfaları, 644  
Web sunucu, 664
- Kontakt bilgisi, 3, 142
- Kontrol DB, kullanıcı-tanımlı Web sayfaları için  
global komutlar, 660  
WWW komutu için parametre, 642  
İstek komutları ve durumları, 660
- CONV (değer dönüştürme), 245
- Dönüştürme (SCL komutları), 245
- Cookie sınırlamaları, standart Web sayfası, 666
- Cookie, Siemens\_automation\_language, 657
- Soğutma, 657
- Kopya koruması  
Bir CPU veya bellek kartına bağlama, 50
- Çevrim içi bir CPU'dan blokların kopyalanması, 197
- COS (cosine değer biçimi), 236
- Sayıcılar  
CTD (aşağı doğru sayma), 220  
CTRL\_HSC (yüksek-hızlı sayıcı kontrolü), 409  
CTU (yukarı doğru sayma), 220  
CTUD (yukarı ve aşağı doğru sayma), 220  
HSC konfigürasyonu, 418  
HSC operasyonu, 418  
operasyon (standart sayıcılar), 221  
miktar, 25, 839, 849, 859, 869, 881  
size, 25, 839, 849, 859, 869, 881

**CPU**

- erişim koruması 190190
- yeni cihaz ekleme, 138
- AS-i, 578
- AS-i adresi, 579
- AS-i port, 578
- çevrim içi bir CPU için bir IP adresi atama, 158
- bir DB'nin değerlerinin yakalanması, 797
- haberleşme, 512
- haberleşme yükü, 98
- kıyaslama ve senkronize etme blokları, 794
- çevrim içi bir CPU'dan blokların kopyalanması, 197
- döngü süresi konfigürasyonu, 98
- cihaz konfigürasyonu, 137
  - MAC ve IP adreslerini gösterme, 163
- yükleme, 196
- cihaza yükleme, 163
- boş transfer kartı, 136
  - STOP modunda çıkışları etkinleştirme, 800
- Ethernet portu, 159
- genişleme kablosu, 64
- kuvvet, 801 801, 802
- online duruma geçme, 786
- topraklama, 72
- HSC konfigürasyonu, 418
- standart Web sayfası belirleme, 615
- indüktif yükler, 74
- kurma, 55, 56
- IP adresi, 159
- izolasyon kılavuzları, 71
- know-how koruma, 193
- lambda yükler, 73
- LED göstericiler, 783
- kayıp şifre, 138
- MAC adresi, 153, 154
- izleme, 796
- ağ bağlantısı, 149
- online, 789, 796
- çalışma modları, 81
- operasyon paneli (online CPU), 792
- operatör paneli, 44
- şifre koruması, 190
- güç hesabı, 51
- güç ihtiyacı, 985
- OB'lerin prosesi, 172
- PROFIBUS, 574
- PROFIBUS adresi, 574
- PROFINET, 159
- PROFINET IO, 558
- program yürütme, 77
- darbe çıkışlar, 365
- kayıp bir şifreden kurtarma, 136
- fabrika ayarlarına reset, 789
- bir DB'nin başlama değerlerinin resetlenmesi, 797
- RTM (çalışma süresi ölçerler), 291
- RUN/STOP butonları, 44
- RUN/STOP modları, 792
- Güvenlik seviyeleri, 190
- başlangıç parametreleri, 127
- başlangıç işlemi, 84
- terminal blok konektör, 63
- termal bölge, 50 50, 54
- zaman senkronizasyon özelliği, 165
- haberleşme tipleri, 509
- tanımsız CPU, 140
- gözetleme çizelgesi, 798
- kablaj kılavuzları, 70, 72
- CPU konfigürasyon
  - HMI için haberleşme, 552
  - döngü süresi izleme, 97
  - modül özellikleri, 147
  - çoklu CPU'lar, 553
  - operasyonel parametreler, 142
  - darbe kanalları, 367
- CPU bellek kartı
  - yerleştirme, 125
  - program kartı, 131
  - transfer kartı, 128
  - kullanımlar, 124
- CPU özellikleri, kullanıcı-tanımlı Web Web sayfası
  - çoklu dillerin ayarı, 659
  - STEP 7 konfigürasyon, 641
- CPU'lar
  - modüller ekleme, 141
  - haberleşme kartları (CB), 29
  - kıyaslama çizelgesi, 24
  - CPU 1211C AC/DC/Röle, 837
  - CPU 1211C DC/DC/DC, 837
  - CPU 1211C DC/DC/Röle, 837
  - CPU 1212C AC/DC/Röle, 847
  - CPU 1212C DC/DC/DC, 847
  - CPU 1212C DC/DC/Röle, 847
  - CPU 1214C AC/DC/Röle, 857
  - CPU 1214C DC/DC/DC, 857
  - CPU 1214C DC/DC/Röle, 857
  - CPU 1215C AC/DC/Röle, 867
  - CPU 1215C DC/DC/DC, 867
  - CPU 1215C DC/DC/Röle, 867
  - CPU 1217C DC/DC/DC, 879
  - haberleşme bağlantı sayısı, 511
  - genel bakış, 23
  - sinyal kartları (SB), 29

basamak tepki süreleri, 842, 852, 862, 873, 888  
 PLC'ler arasında bir ağ haberleşmesi oluşturma, 149  
 Kullanıcı-tanımlı Web Sayfa DB'leri oluşturma, 642  
 Kullanıcı-tanımlı Web sayfaları oluşturma, 626  
 Kullanımı göstermek için Çapraz-referans, 200  
 CSM 1277 kompakt anahtar modülü, 982  
 CTD (aşağı doğru sayma), 220  
 C-Tick onayı, 831  
 CTRL\_HSC (yüksek-hızlı sayıcı kontrolü), 409  
 CTS, 674  
 CTU (yukarı doğru sayma), 220  
 CTUD (yukarı ve aşağı doğru sayma), 220  
 cULus onayı, 830  
 Müşteri desteği, 3  
 Döngü süresi  
   konfigürasyon, 98  
   izleme, 792  
   genel bakış, 97  
 Döngüsel interrupt OB, 87

## D

### Veri bloğu

Değerlerin yakalanması, 797  
 CONF\_DATA, 544  
 global veri bloğu, 105, 176  
 instance veri bloğu, 105  
 optimize edilmiş erişim, 177  
 organizasyon blokları (OB'ler), 172  
 genel bakış, 77, 176  
 READ\_DBL (yükleme belleğindeki veri bloğundan okuma), 401  
 başlama değerlerini resetleme, 797  
 çoklu instance DB'lere sahip tek FB ,176  
 standart erişim, 177  
 yapı, 77  
 WRIT\_DBL (yükleme belleğine veri bloğu yazma), 401  
 Veri bloğu, kullanıcı-tanımlı Web Web sayfaları için, fragmentların içeri aktarımı, 638  
 Veri değişimi, IO sistemleri arasında, 565  
 Veri işleme bloğu (DHB), 176  
 Veri günlüğü  
   veri günlüklerini gözden geçirme, 380  
   veri kayıt yapısı, 381  
   DataLogClose (veri günlüğünü kapama), 389  
   DataLogCreate (veri günlüğü oluşturma), 382  
   DataLogNewFile (yeni dosyada veri günlüğü), 390  
   DataLogOpen (veri günlüğü açma), 385  
   DataLogWrite (veri günlüğüne yazma), 387  
   örnek program, 396  
   veri günlüğüne boyutu limitleri, 393  
   veri günlüklerini gözden geçirme, 392

Veri günlükleri standart Web sayfaları, 623  
 Veri aktarımı, başlatma, 694  
 Veri tipleri, 110  
   Herhangi (işaretçi), 119  
   diziler, 116  
   Bool, Byte, Word ve DWord, 111  
   Char (karakter) ve string, 115  
   PLC veri tipi editörü, 117  
   Pointer (işaretçi), 118  
   işaretçi veri tipi gözden geçirme, 118  
   Real, LReal (kayan-nokta gerçek), 112  
   Struc, 117  
   Saat, Tarih, TOD (günün saati), DTL (tarih ve saat uzun), 113  
   USInt, SInt, UInt, Int, UDInt, Dint (integer), 112  
   Variant (işaretçi), 120

### Tarih

Tarih veri tipi, 113  
 DTL (tarih ve saat uzun veri tipi), 114  
 SET\_TIMEZONE (saat bölgesinin ayarı), 290  
 T\_ADD (süreleri toplama), 284  
 T\_COMBINE (süreleri birleştirme), 285  
 T\_CONV (süreleri dönüştür ve çıkar), 283  
 T\_DIFF (süre farkı), 285  
 T\_SUB (çıkarma süreleri), 284

Yaz saati TimeTransformationRule, 289

DB (veri bloğu), (veri bloğu)

### DC

topraklama, 72  
 indüktif yükler, 74  
 izolasyon kılavuzları, 71  
 çıkışlar, 835  
 kablaj kılavuzları, 70, 72

### Hata ayıklama

RUN modunda yükleme, 803, 811

DEC (azaltma), 232

DECO (kod çözme), 276

Enum tiplerin tanımlaması, kullanıcı-tanımlı Web sayfaları, 635

Koruma derecesi, 834

DELETE (bir karakter string'inde karakterleri silme), 169, 170

Bir PLC sistemi tasarlama, 169, 170

DETACH (bir interrupt olayından bir OB ayırma, 331

### Cihaz

PROFINET IO, 558

Cihaz konfigürasyonu, 137, 551

modüller ekleme, 141

yeni cihaz ekleme, 138

AS-i, 579

AS-i port, 579

bir cihaz tipini değiştirme, 142

CPU'yu konfigüre etme, 142

modülleri konfigüre etme, 147

- keşfetmek, 140
  - yükleme, 196
  - Ethernet portu, 159
  - ağ bağlantısı, 149
  - PROFIBUS, 574
  - PROFINET, 159
  - zaman senkronizasyon özelliği (PROFINET), 165
  - takılmamış modüller, 47
  - Cihaz değişimi, 142
  - Cihaz isimleri
    - PROFINET IO, 559
  - DeviceStates (bir I/O sisteminin modül durumunun okunması), 347
  - Hata interrupt OB diyagnostiği, 89
  - Standart Web sayfası diyagnostiği, 616
  - Diyagnostikler (tanılamalar)
    - ara bellek, 103
    - döngü süresi, 792
    - DeviceStates (bir I/O sisteminin modül durumunun okunması), 347
    - tanılama ara belleği, 793
    - GET\_DIAG (diyagnostik bilgisi okuma), 358
    - LED (LED durumunu okuma), 346
    - LED gösterciler, 783
    - bellek kullanımı, 792
    - ModuleStates (bir modülün modül durum bilgisini okuma), 353
    - durum göstercisi, 102
    - gözetleme çizelgesi, 798
  - Dijital I/O
    - konfigürasyon, 147
    - darbe yakalama, 147
    - durum gösterciler, 784
  - Dijital giriş filtre süresi, 144
  - Dijital sinyal kartları
    - SB 1221, 935
    - SB 1222, 937
    - SB 1223, 940, 943
  - Dijital sinyal modülleri
    - SM 1221, 892
    - SM 1222, 894, 895
    - SM 1223, 900,904
  - DIN rayı, 55
  - Kullanıcı-tanımlı Web sayfaları için klasörler, diller, 656
  - 656
  - DIS\_AIRT (yüksek öncelikli interrupt'ları ve asenkron hata olaylarını yürütmeyi etkisizleştirme), 344
  - Çevrim içi bir CPU'ya yüklemeyi keşfetmek, 140
  - MAC ve IP adreslerini gösterme, 163
  - DIV (böl), 230
  - Dokümantasyon, 4
  - Yükleme, RUN modunda
    - derleme hataları, 807
    - dikkate almalar, 811
    - ön ayarsız yükleme,808
    - seçili blokları yükleme, 806
    - genişletilmiş blok ara yüzü, 808
    - başarısız yükleme, 811
    - global bellek ayırma ayarlaması, 810
    - STEP 7'den başlatma, 805
    - bellek ayırma ve kalıcı bellek reserve, 808
    - genel bakış, 803
    - önşartlar, 804
    - sınırlamalar, 810
  - Yükleme
    - MAC ve IP adreslerini gösterme, 163
    - Firmware güncelleme, 133
    - Proje, 196
    - PC için Siemens güvenlik sertifikası, 613,666
    - kullanıcı programı, 196
    - kullanıcı-tanımlı Web Sayfa DB'ler, 643
  - DPNRM\_DG, 328
  - DPRD\_DAT (bir DP standart slave için uyumlu veriyi okuma), 326
  - DPWR\_DAT (bir DP standart slave için uyumlu veriyi yazma), 326326
  - Editörler arasında sürükle ve bırak, 43
  - Sürücüler
    - bağlantı, 724
    - genel sürücü kurma şartları, 724
    - MM4 sürücü kurma, 726
- ## E
- Düzenleme, RUN modunda, (RUN modunda yükleme)
  - Elektromanyetik uyumluluk, 833
  - Elektromanyetik uyumluluk (EMC), 832
  - EN ve ENO (güç akışı), 188188
  - EN\_AIRT (yüksek öncelikli interrupt'ları ve asenkron hata olaylarını yürütmeyi etkinleştirme), 344
  - ENCO (kodlama), 276
  - Bitiş koşulları, 679
  - Mesaj karakterini bitirme, 681
  - ENDIS\_PW (aktif olmayan şifreyi etkinleştirme), 259
  - Kullanıcı-tanımlı Web sayfasında enum tipleri, 634,635
  - Çevresel
    - işletim koşulları, 833
    - taşıma ve depolama koşulları, 833
  - Hata kodları
    - genişletilmiş komutlar için ortak hatalar, 407
    - RALRM, 322
    - RDREC, 322

- WRREC, 322
- Hatalar
  - diyagnostik hatalar, 90
  - zaman hataları, 88
- Ethernet
  - ad hoc modu, 518
  - bağlantı ID'leri, 514
  - CSM 1277 kompakt anahtar modülü, 982
  - DPNRM\_DG (bir DP slave'den tanılama verisi okumak), 328
  - DPRD\_DAT (bir DP standart slave için uyumlu veriyi okuma), 326
  - DPWR\_DAT (bir DP standart slave için uyumlu veriyi yazma), 326
  - GET (uzaktaki bir CPU'dan veri okuma), 587
  - IP adresi, 159
  - MAC adres, 159
  - ağ bağlantısı, 149
  - haberleşme bağlantılarının sayısı, 511
  - genel bakış, 517
  - PUT (uzaktaki bir CPU'ya veri yazma), 587
  - RALRM (interrupt almak), 318
  - RDREC (veri kaydı okuma), 315
  - T\_CONFIG (ara yüzü konfigüre etme), 541
  - TCON (haberleşme bağlantısı yapma), 527
  - TDISCON (haberleşme bağlantısı sonlandırma), 527
  - TRCV (haberleşme bağlantısı yoluyla veri alma), 527
  - TRCV\_C (Ethernet yoluyla veri alma(TCP)), 520
  - TSEND (haberleşme bağlantısı yoluyla veri gönderme), 527
  - TSEND\_C (Ethernet yoluyla veri gönderme(TCP)), 520
  - TURCV (Ethernet yoluyla veri alma(UDP)), 535
  - TUSEND (Ethernet yoluyla veri gönderme(UDP)), 535
  - haberleşme tipleri, 509
  - WRREC (veri kaydı yazma), 315
- Ethernet protokolleri, 517
  - çoklu-düğümlü bağlantılar, 591
- Olay yürütme, 94
- Örnek
  - Modbus slave, 764
  - PtP haberleşme, 704
  - PtP haberleşme, konfigürasyon, 705
  - PtP haberleşme, çalışma, 712
  - PtP haberleşme, STEP 7 programlama, 710
  - PtP haberleşme, terminal emülatör, 711
  - kullanıcı-tanımlı Web Sayfaları, 645
- EXP (üstel değer biçimi), 236
- Genişletilebilir komutlar, 42
- S7-1200'ün genişleme yeteneği, 26
- Uzatma kablosu, 981
- kurma, 64
- sökme, 64
- EXPT (üst almak), 236
- Genişletilmiş blok ara yüzü
  - RUN modunda yükleme, 808
- F**
- F\_TRIG (negatif sinyal kenarındaki etiket ayarı), 210
- Fabrika ayarlamalarına reset, 789
- FAQ'lar, 4
- Favori araçlar, 38
- FB (fonksiyon bloğu)
  - genel bakış, 77
- FBD (fonksiyon blok diyagramı), 181
- FC (fonksiyon), 77, 174
- FieldRead (alan okuma), 240
- FieldWrite (alan yazma), 240
- FILL\_BLK (blok doldurma), 242
- Filtre süresi, 144
- FIND (bir karakter string'inde karakterler bulma), 313
- Firmware güncelleme, 133
  - STEP 7'den, 791
  - Web sunucudan, 619
- İlk tarama göstericisi, 102
- Sabit uzunluk, 681
- FLOOR (kayan-nokta sayıdan sonraki en yüksek tamsayıyı üretme), 250
- Akış kontrol, 673, 674
  - konfigürasyon, 673
- FM onayı, 830
  - Klasörler, kullanıcı-tanımlı Web sayfaları için diller, 656
- Kuvvet, 801801, 802
  - l bellek, 801801, 802
  - girişler ve çıkışlar, 802
  - bellek kartı, 124
  - çevresel girişler, 801, 802
  - tarama döngüsü, 802
  - gözetleme çizelgesi, 798
- Kuvvet çizelgesi
  - çevresel girişleri adresleme, 801
  - kuvvet, 801
  - kuvvet operasyonu, 802
- FRAC (kesri döndür), 236
- Fragment DBs (kullanıcı-tanımlı Web sayfaları)
  - üretme, 642
- Fragmentler (kullanıcı-tanımlı Web sayfaları)
  - AWP komutundan oluşturma, 637
  - AWP komutu ile içeri aktarma, 638
- Freeport protokolü, 671
- Frekans, saat bitleri, 103
- Fonksiyon (FC)

- kullanıcı programı içerisinde kod bloklarının çağırılması, 172
  - know-how koruma, 193
  - doğrusal ve yapılandırılmış programlar, 170
  - genel bakış, 77, 174
  - geçerli FB numaraları, 77
  - Fonksiyon bloğu (FB)
    - kullanıcı programı içerisinde kod bloklarının çağırılması, 172
    - başlangıç değeri, 175
    - instance veri bloğu, 175
    - know-how koruma, 193
    - doğrusal ve yapılandırılmış programlar, 170
    - çıkış parametreleri, 175
    - genel bakış, 77, 175
    - çoklu instance DB'lere sahip tek FB, 176
    - geçerli FB sayıları, 77
  - Fonksiyonellik
    - I-cihaz, 561
- ## G
- Kullanıcı-tanımlı Web Sayfa DB'leri üretme, 642
  - GET
    - bağlantıyı konfigüre etme, 151
  - GET (uzaktaki bir CPU'dan veri okuma), 587
  - GET\_DIAG (diyagnostik bilgisi okuma), 358
  - GET\_ERROR (hatayı yerel olarak almak), 263
  - GET\_ERROR\_ID (hata ID'yi yerel olarak almak), 265
  - Global veri bloğu, 105, 176
  - Global kütüphane
    - USS protokolü gözden geçirme, 712
  - GSD dosya, 568
  - Kılavuzlar
    - CPU kurma, 56
    - topraklama, 7272
    - indüktif yükler, 74
    - kurma, 49
    - kurma işlemleri, 55
    - izolasyon, 71
    - lambda yükler, 73
    - kablaj kılavuzları, 70, 72
- ## H
- Donanım konfigürasyonu, 137
    - modüller ekleme, 141
    - yeni cihaz ekleme, 138
    - AS-i, 579
    - AS-i port, 579
    - CPU'yu konfigüre etme, 142
    - modülleri konfigüre etme, 147
    - keşfetmek, 140
    - yükleme, 196
    - Ethernet portu, 159
    - ağ bağlantısı, 149
    - PROFIBUS, 574
    - PROFINET, 159
  - Donanım akış kontrol, 674
  - Donanım interrupt OB, 87
  - Yüksek potansiyelli izolasyon deneyi, 834
  - Yüksek-hızlı sayıcılar, 409
    - zorlanamaz, 802
    - konfigürasyon, 418
    - operasyon, 411
  - HMI cihazları
    - PROFINET haberleşmesini konfigüre etme, 552
    - 552
    - ağ bağlantısı, 149
    - genel bakış, 32
  - Hotline (yardım hattı), 3
  - HSC (yüksek-hızlı sayıcılar)
    - konfigürasyon, 418418
    - operasyon, 411411
  - HTA ( hexadecimal sayıyı ASCII string'e dönüştürme), 304
  - HTML sayfaları
    - listeleme, kullanıcı-tanımlı Web Sayfa örneği, 650
    - kullanıcı-tanımlı, 625
  - HTML sayfaları, kullanıcı-tanımlı
    - S7-1200 veriye erişim, 627
    - geliştirme, 626
    - dil yerleşimleri, 659
    - Sayfa yerleşimleri, 641
    - yenileme, 627
  - HTTP bağlantıları, Web sunucu, 664
- ## I
- I bellek
    - kuvvet, 801
    - kuvvet operasyonu, 802
    - kuvvet Çizelgesi, 801
    - monitör, 796
    - monitör LAD, 797
    - çevresel giriş adresleri(kuvvet çizelgesi), 801
    - gözetleme çizelgesi, 796
  - I/O
    - adresleme, 109
    - analog çıkış temsili (akım), 917, 951
    - analog çıkış temsili (gerilim), 916, 950

- analog çıkış temsili (akım), 918, 952
- analog çıkış temsili (gerilim), 917, 951
- analog durum göstericiler, 785
- dijital durum göstericiler, 784
- kuvvet operasyonu, 802
- indüktif yükler, 74
- LAD'de durum izleme, 797
- bir gözetleme çizelgesiyle gözetleme, 798
- basamak tepki süreleri (CPU), 841, 852, 862, 873, 888
- basamak tepki süreleri (SB), 950
- basamak tepki süreleri (SM), 916
- Standart Web sayfası belirleme, 615
- I-cihaz
  - GSD dosya ile konfigüre etme, 568
- I-cihaz (zeki IO cihazı)
  - konfigüre etme, 567
  - fonksiyonellik, 561
  - düşük seviye PN IO sistem, 562
  - Özellikler, 562
- Boş hat, 676, 677
- Siemens güvenlik sertifikasını aktarma, 666
- IN\_Range (değer aralık içersinde), 227
- INC (artış), 232
- Uyumlu olmayan CPU versiyon hatası, 784
- değişkenlere sahip dizinlerin indekslenmesi, 241
- İndüktif yükler, 74
- Endüstriyel çevreler
  - onaylar, 831
- Bilgi kaynakları, 4
- Başlangıç değerleri
  - yakalama ve bir DB'nin başlaması değerlerinin resetlenmesi, 797
- Giriş filtre süresi, 144
- Giriş simülasyonları, 978
- Girişler
  - darbe yakalama bitleri, 146
- Girişler ve çıkışlar
  - izleme, 796
- INSERT (bir karakter string'ine karakterler yerleştirme), 311
- Bir cihaz yerleştirme
  - tanımsız CPU, 140
- Komutların yerleştirilmesi
  - sürükle ve bırak, 38
  - editörler arasında sürükle ve bırak, 43
  - favoriler, 38
- CPU'ya bellek kartı takma, 125
- Kurulum
  - hava akışı, 50
  - açıklık, 50
  - haberleşme kartı (CB), 58
  - haberleşme modülü (CM), 62
  - soğutma, 50
- CPU, 56
  - uzatma kablosu, 64
  - topraklama, 72
  - kılavuzlar, 49
  - indüktif yükler, 74
  - izolasyon kılavuzları, 71
  - lamba yükler, 73
  - montaj boyutları, 54
  - genel bakış, 49, 55
  - güç hesabı, 51
  - şartlar, 36
  - sinyal kartı(SB), 58
  - sinyal modülü(SM), 60 60
  - sinyal modülleri(SM), 29
  - terminal blok konektör, 63
  - termal bölge, 50, 54
  - TS Adaptör ve TS modül, 66
  - Bir DIN rayında TS Adaptör, 69
  - Bir duvarda TS Adaptör, 69
  - TS Adapter SIM kartı, 67
  - kablaj kılavuzları, 70, 72
- Instance veri bloğu, 105
- Komutlar
  - & kutu (FBD AND lojik operasyon), 203
  - ( )- (atama), 203
  - (/)- (atama reddi), 203
  - (N)- (negatif sinyal kenarındaki operand ayarı),
  - (P)- (pozitif sinyal kenarı için operand ayarı), 209
  - (R) (reset çıkışı), 209
  - (RESET\_BF) (reset bit alanı), 206
  - (S) (ayar çıkışı), 207
  - (SET\_BF) (bit alanı ayarı), 206
  - /= kutu (FBD atama reddi), 203
  - |/|- (normalde kapalı kontak), 203
  - ||- (normalde açık kontak), 203
  - |N|- (negatif sinyal kenarındaki operand taraması), 209
  - |NOT|- (invert RLO), 203
  - |P|- (pozitif sinyal kenarındaki operand taraması), 209
  - = kutu (FBD atama), 203
  - ABS (mutlak değer biçimi), 233
  - ACOS (arccosine değeri biçimi), 236
  - ACT\_TINT (günün saati interrupt'ını aktif etme), 339
  - ADD (ekle), 230
    - LAD veya FBD komutlarına girişler ve çıkışlar ekleme, 41
  - AND (lojik operasyon), 275
  - AS-i dağıtımlı I/O, 314
  - ASIN (arcsine değeri biçimi), 236
  - ATAN (tangent değeri biçimi), 236
  - ATH (ASCII string'i hexadecimal sayıya dönüştürme), 304
  - ATTACH (bir OB'yi bir interrupt olaya iliştir, 331



- CALCULATE, 39  
CALCULATE (hesapla), 229  
takvim, 283  
CAN\_DINT (zaman-gecikmesi interrupt'ını iptal et), 342  
CAN\_TINT (günün saati interrupt'ını iptal et), 339  
CASE (SCL), 269  
CEIL (kayan-nokta sayıdan sonraki en yüksek tamsayıyı üretme), 250  
Chars\_TO\_Strg (CHAR dizinini karakter string'ine dönüştürme), 302  
saat, 287  
sütunlar ve başlıklar, 42, 746  
ortak parametreler, 548  
değerleri kıyasla, 226  
CONCAT (karakter string'lerini birleştirme), 307  
CONTINUE (SCL), 273  
CONV (değer dönüştürme), 245  
COS (cosine değer biçimi), 236  
CTD (aşağı doğru sayma), 220  
CTRL\_HSC (yüksek-hızlı sayıcı kontrolü), 409  
CTRL\_PWM (darbe genişlik modülasyonu), 364  
CTU (yukarı doğru sayma), 220  
CTUD (yukarı ve aşağı doğru sayma), 220  
DataLogClose (veri günlüğü kapama), 389  
DataLogCreate (veri günlüğü oluşturma), 382  
DataLogNewFile (yeni dosyada veri günlüğü), 390  
DataLogOpen (veri günlüğü açma), 385  
DataLogWrite (veri günlüğüne yazma), 387  
tarih, 283  
DEC (azaltma), 232  
DECO (kod çözme), 276  
DELETE (bir karakter string'inde karakterleri silme), 310  
DETACH (bir interrupt olayından bir OB ayırmak), 331  
DeviceStates (bir I/O sisteminin modül durumunun okunması), 347  
DIS\_AIRT (yüksek öncelikli interrupt'ları ve asenkron hata olaylarını yürütmeyi etkisizleştirme), 344  
DIV (böl), 230  
DPNRM\_DG (bir DP slave'den tanılama verisi okumak), 328  
DPRD\_DAT (bir DP standart slave için uyumlu veriyi okuma), 326  
DPWR\_DAT (bir DP standart slave için uyumlu veriyi yazma), 326  
sürükle ve bırak, 38  
editörler arasında sürükle ve bırak, 43  
EN\_AIRT (yüksek öncelikli interrupt'ları ve asenkron hata olaylarını yürütmeyi etkinleştirme), 344  
ENCO (kodlama), 276  
ENDIS\_PW (aktif olmayan şifreyi etkinleştirme), 259  
EXIT (SCL), 273  
EXP (üstel değer biçimi), 236  
genişletilebilir komutlar, 42  
EXPT (üst almak), 236  
F\_TRIG (negatif sinyal kenarındaki etiket ayarı), 210  
favoriler, 38  
FieldRead (alan okuma), 240  
FieldWrite (alan yazma), 240  
FILL\_BLK (blok doldurma), 242  
FIND (bir karakter string'inde karakterler bulma), 313  
FLOOR (kayan-nokta sayıdan sonraki en yüksek tamsayıyı üretme), 250  
FOR (SCL), 270  
kuvvet, 801  
kuvvet operasyonu, 802  
FRAC (kesri döndür), 236  
GET (uzaktaki bir CPU'dan veri okuma), 587  
GET\_DIAG (diyagnostik bilgisi okuma), 358  
GET\_ERROR (hatayı yerel olarak almak), 263  
GET\_ERROR\_ID (hata ID'yi yerel olarak almak), 265  
GOTO (SCL), 274  
HSC (yüksek-hızlı sayıcılar), 411  
HTA (hexadecimal sayıyı ASCII string'e dönüştürme), 304  
IF-THEN (SCL), 268  
IN\_Range (değer aralık içerisinde), 227  
INC (artış), 232  
INSERT (bir karakter string'ine karakterler yerleştirme), 311  
takma, 38  
INV (birilerin tamamlayanını oluşturma), 276  
JMP (jump if RLO = 1), 254  
JMP\_LIST (atlama listesini tanımlama), 255  
JMPN (jump if RLO = 0), 254  
Label (atlama etiketi), 254  
LED (LED durumunu okuma), 346  
LEFT (bir karakter string'inin sol karakterlerini okuma), 308  
LEN (bir karakter string'inin uzunluğunu belirleme), 307  
LIMIT (sınır değeri ayarla), 235  
LN (doğal logaritma biçimi), 236  
LOG2GEO (donanım belirtecinden slot belirleme), 404

- MAX (maksimumu almak), 234  
MAX\_LEN (bir karakter string'inin maksimum uzunluğu), 306  
MB\_CLIENT ( Modbus TCP istemci olarak PROFINET yoluyla haberleşme), 730  
MB\_COMM\_LOAD (Modbus RTU için PtP modülünde port konfigüre etme ), 747  
MB\_MASTER (Modbus master olarak PtP port ile haberleşme), 749  
MB\_SERVER (Modbus TCP sunucu olarak PROFINET yoluyla haberleşme), 736  
MB\_SLAVE (Modbus slave olarak PtP port yoluyla haberleşme), 755  
MC\_ChangeDynamic (eksen için dinamik ayarlamayı değiştirme), 476  
MC\_Command Çizelgesi, 473  
MC\_Halt (ekseni duraklat), 462  
MC\_Home (ekseni yatağına getirme), 460  
MC\_MoveAbsolute (ekseni mutlak olarak konumlandırma), 464  
MC\_MoveJog (jog modunda eksen hareket ettirme), 471  
MC\_MoveRelative (ekseni bağıl olarak konumlandırma), 466  
MC\_MoveVelocity (ekseni ön tanımlı hızda hareket ettirme), 468  
MC\_Power (eksenleri serbest bırak/tut), 456  
MC\_ReadParam (bir teknoloji nesnesinin parametrelerini okuma), 480  
MC\_Reset (hata onaylama), 459  
MC\_WriteParam (bir teknoloji nesnesinin parametrelerini okuma), 478  
MID (karakter string'inin orta karakterlerini okuma), 308  
MIN (minimumu almak), 234  
MOD (bölme işleminin kalanını döndürme), 231  
ModuleStates (bir modülün modül durum bilgisini okuma), 353  
monitör, 796,797  
hareket kontrolü, 455  
MOVE (değeri taşıma), 238  
MOVE\_BLK (bloğu taşıma), 238  
MUL (çarp), 230  
MUX (çoğalt), 278  
N (negatif sinyal kenarındaki operand taraması), 209  
N\_TRIG (negatif sinyal kenarı için RLO tarama), 210  
N= kutu ve N bobin(negatif sinyal kenarındaki operand ayarı), 209  
NEG (ikinin tamamlayanını oluşturma), 232  
NORM\_X (normalize), 251  
NOT\_OK (geçersizliği kontrol etme), 228  
OK (geçerliliği kontrol etme), 228  
OR (lojik operasyon), 275  
OUT\_Range (değer aralık dışında), 227  
P (pozitif sinyal kenarındaki operand taraması), 209  
P\_TRIG (pozitif sinyal kenarı için RLO tarama), 210  
P= kutu ve P bobin (pozitif sinyal kenarı için operand ayarı), 209  
PID\_Compact (entegreli ayarlaması olan üniversal PID kontrolör), 422  
PORT\_CFG (haberleşme parametrelerini dinamik olarak konfigüre etme), 686  
PROFIBUS dağıtımlı I/O, 314  
PROFINET dağıtımlı I/O, 314  
program kontrol (SCL), 267  
PUT (uzaktaki bir CPU'ya veri yazma), 587  
QRY\_CINT (döngüsel interrupt parametrelerinin ayarı), 336  
QRY\_DINT (zaman-gecikmesi interrupt durumu sorgulama), 342  
QRY\_TINT (günün saati interrupt durumu sorgulama), 340  
R (reset çıkışı), 206  
R\_TRIG (pozitif sinyal kenarına etiket ayarı), 210  
RALRM (interrupt alma), 318  
RCV\_CFG (seri alma parametrelerini dinamik olarak konfigüre etme), 689  
RCV\_PTP (mesaj almayı etkinleştirme), 697696  
RCV\_RST (alıcı ara belleği silme), 698  
RD\_ADDR (donanım belirtecinden IO adreslerini tanımla), 406  
RD\_LOC\_T (yerel saati okuma), 287  
RD\_SYS\_T (günün saatini okuma), 287  
RDREC (veri kaydı okuma), 315  
RE\_TRIGR, 97  
RE\_TRIGR (döngü izleme süresini yeniden başlatma), 262  
READ\_DBL (yükleme belleğindeki veri bloğundan okuma), 401  
RecipeExport (tarif alma), 374  
RecipeImport (tarif alma), 376  
REPEAT (SCL), 272  
REPLACE (bir karakter string'indeki karakterleri değiştirme), 312  
RESET\_BF (reset bit alanı), 207  
RET (değer döndür), 258  
RETURN (SCL), 274  
RIGHT (karakter string'inin sağ karakterini okuma), 308  
ROL (sola döndür) ve ROR (sağa döndür), 282  
ROUND (sayısal değeri yuvarla), 249  
RS (reset/set flip-flop), 207  
RT (reset zamanlayıcı), 212  
RTM (çalışma süresi ölçerler), 291  
S (ayar çıkışı), 206  
S\_CONV (karakter string'ini dönüştürme), 293

- S\_MOV (karakter string'ini taşıma), 293  
SCALE\_X (ölçek), 251  
analog değerlerin ölçeklenmesi, 39  
SCL dönüştürme komutları, 245  
SEL (seç), 278  
SEND\_CFG (seri aktarma parametrelerin dinamik konfigürasyonu), 688  
SEND\_PTP (ara bellek veri gönderme aktarması), 694  
SET\_BF (bit alanı ayarı), 207  
SET\_CINT (döngüsel interrupt parametrelerinin ayarı), 334  
SET\_TIMEZONE (saat bölgesinin ayarı), 290  
SET\_TINTL (tarih ve gün interrupt'ı süresinin ayarı), 337  
SGN\_GET (RS232 sinyalleri sorgusu), 699  
SGN\_SET (RS-232 sinyallerinin ayarı), 700  
SHL (sola kaydır) ve SHR (sağa kaydır), 281  
SIN (sinüs değer biçimi), 236  
SQR (kare biçimi), 236  
SQRT (kare biçimi kökü), 236  
SR (set/reset flip-flop), 207  
SRT\_DINT (zaman gecikmesi interrupt'ını başlat), 342  
durum, 796, 797  
STP (programdan çıkış), 263  
Strg\_TO\_Chars (karakter string'ini CHAR dizisine dönüştür), 302  
STRG\_VAL (karakter string'ini sayısal değere dönüştür), 293  
SUB (çıkart), 230  
SWAP (baytları yer değiştir), 243  
SWITCH (atlama dağıtıcısı), 256  
T\_ADD (süreleri toplama), 284  
T\_COMBINE (süreleri birleştirme), 285  
T\_CONFIG (ara yüzü konfigüre etme), 541  
T\_CONV (süreleri dönüştür ve çıkar), 283  
T\_DIFF (süre farkı), 285  
T\_SUB (çıkarma süreleri), 284  
TAN (tangent değer biçimi), 236  
TCON (haberleşme bağlantısı yapma), 527  
TDISCON (haberleşme bağlantısı sonlandırma), 527  
zaman, 283  
zamanlayıcı, 212  
TM\_MAIL (email gönder), 775  
TOF (off-gecikme zamanlayıcı), 212  
TON (on-gecikme zamanlayıcı), 212  
TONR (on-gecikme kalıcı zamanlayıcı), 212  
TP (darbe zamanlayıcı), 212  
TRCV (haberleşme bağlantısı yoluyla veri alma), 527  
TRCV\_C, 556  
TRCV\_C (Ethernet yoluyla veri alma(TCP)), 520  
TRUNC (sayısal değeri kırpma), 249  
TSEND (haberleşme bağlantısı yoluyla veri gönderme), 527  
TSEND\_C, 555  
TSEND\_C (Ethernet yoluyla veri gönderme(TCP)), 520  
TURCV (Ethernet yoluyla veri alma(UDP)), 535  
TUSEND (Ethernet yoluyla veri gönderme(UDP)), 535  
UFILL\_BLK (doldurma bloğu kesintisiz), 242  
UMOVE\_BLK (bloğu taşıma kesintisiz), 238  
USS durum kodları, 722  
USS\_DRV (Sürücüyle veri yer değiştirme), 716  
USS\_PORT (USS ağla haberleşme düzenleme), 715  
USS\_RPM (sürücünden parametrelerin okunması), 719  
USS\_WPM (sürücüde parametreleri değiştirme), 720  
VAL\_STRG (sayısal değeri karakter string'ine dönüştürme), 293  
komutların versiyonları, 42, 746  
WHILE (SCL), 271  
WR\_LOC\_T (yerel saati ayarla), 287  
WR\_SYS\_T (günün saatini ayarla), 287  
WRIT\_DBL (yükleme belleğine veri bloğu yazma), 401  
WRREC (veri kaydı yazma), 315  
WWW (kullanıcı-tanımlı Web sayfaları senkronizasyonu), 642  
XOR (lojik operasyon), 275  
Karakter boşlukları arası, 681  
Interrupt'lar  
ATTACH (bir OB'yi bir interrupt olaya iliştir), 331  
CAN\_DINT (zaman-gecikmesi interrupt'ını iptal et), 342  
DETACH (bir interrupt olayından bir OB ayırmak), 331  
interrupt gecikmesi, 94  
genel bakış, 85  
QRY\_DINT (zaman-gecikmesi interrupt durumu sorgulama), 342  
SRT\_DINT (zaman gecikmesi interrupt'ını başlat), 342  
Standart Web sayfasına giriş, 613  
INV (birilerin tamamlayanını oluşturma), 276  
IO sistem

veri deęiřimi, 565  
IO-Link  
adres boşluęu, 822  
çalıma süresince parametreleri deęiřtirme, 822  
konfigüre etme, 821  
veri kaydı, 823  
cihaz profili, 815  
cihaz saklama, 818  
diyagnostikler (tanılamalar), 826  
diyagram, 820  
hata mesajları,822,825, 826  
fonksiyonlar, 817  
LED display, 825  
parametreler, 821  
pin atama, 819  
yer deęiřtirme, 817  
fabrika ayarlarına reset, 818  
IO-Link Master sinyal modülü,931  
IP adres,160  
atama,155, 162  
çevrim içi atama, 158  
konfigüre etme, 159  
çevrim içi CPU konfigürasyonu, 789  
cihaz konfigürasyonu, 142  
MAC adres, 159  
IP router, 159  
ISO on TCP  
ad hoc modu, 518  
ISO on TCP protokolü, 517  
İzolasyon kılavuzları, 71  
ISO-on-TCP  
baęlantı konfigürasyonu, 150  
baęlantı ID'leri, 514  
parametreler, 153

**J**

JavaScript sınırlamaları, standart Web sayfaları, 665  
JMP (jump if RLO = 1), 254  
JMP\_LIST (atlama listesini tanımlama), 255  
JMPN (jump if RLO = 0), 254

**K**

Know-how koruma  
şifre koruması, 193  
Korea Certification approval, 831

**L**

Label (atlama etiketi), 254  
LAD (merdiven lojji)

monitör,796, 797  
genel bakış, 180  
program editörü, 797  
durum, 796, 797,801  
Lamba yükler, 73  
Diller, kullanıcı-tanımlı Web sayfası, 656  
Gecikme, 94  
LED (LED durumunu okuma), 346  
LED göstericiler  
haberleşme ara yüzü, 669, 783  
CPU durumu, 783  
LEFT (bir karakter string'inin sol karakterlerini okuma), 308  
LEN (bir karakter string'inin uzunluęunu belirleme), 307  
Uzunluk  
mesaj, 682  
Uzunluk m, 682  
Uzunluk n, 682  
LENGTH parametresi, SEND\_PTP, 696  
LIMIT (sınır deęeri ayarla), 235  
Doęrusal programlama, 170  
LN (doęal logaritma biçimi), 236  
Yükleme belleęi, 24  
CPU 1211C, 837  
CPU 1212C, 847  
CPU 1214C, 857  
CPU 1215C, 867  
CPU 1217C, 879  
Hafıza kartı, 124  
program kartı, 124  
transfer kartı, 124  
kullanıcı-tanımlı Web sayfası, 644  
Yerel zaman  
RD\_LOC\_T (yerel saati okuma), 287  
WR\_LOC\_T (yerel saati ayarla), 287  
Lokal/Partner baęlantısı, 512  
LOG2GEO (donanım belirtecinden slot belirleme), 404  
Oturum açma/kapama  
standart Web sayfaları, 611  
Lojik analizör, 813  
Kayıp şifre, 136

**M**

MAC adres,159, 163  
Manuel fragment DB kontrolü, 660  
Kılavuzlar, 4  
Maritime onayı, 831  
Master yoklama yapısı, 703  
Math,39 **Hata! Yer işareti tanımlanmamış.**,229 229,  
230  
MAX (maksimumu almak), 234

- MAX\_LEN (bir karakter string'inin maksimum uzunluğu),306
- Maksimum mesaj uzunluğu, 681
- Maksimum Web sunucu bağlantıları, 664
- MB\_CLIENT ( Modbus TCP istemci olarak PROFINET yoluyla haberleşme), 730
- MB\_COMM\_LOAD (Modbus RTU için PtP modülünde port konfigüre etme ), 747
- MB\_MASTER (Modbus master olarak PtP port ile haberleşme), 749
- MB\_SERVER (Modbus TCP sunucu olarak PROFINET ile haberleşme), 736
- MB\_SLAVE (Modbus slave olarak PtP port yoluyla haberleşme), 755
- MC\_ChangeDynamic (eksen için dinamik ayarlamayı değiştirme), 476
- MC\_CommandTable, 473
- MC\_Halt (ekseni duraklat), 462
- MC\_Home (ekseni yatağına getirme), 460
- MC\_MoveAbsolute (ekseni mutlak olarak konumlandırma), 464
- MC\_MoveJog (jog modunda eksen hareket ettirme), 471
- MC\_MoveRelative (ekseni bağlı olarak konumlandırma), 466
- MC\_MoveVelocity (ekseni ön tanımlı hızda hareket ettirme, 468
- MC\_Power (eksenleri serbest bırak/tut), 456
- MC\_ReadParam (bir teknoloji nesnesinin parametrelerini okuma), 480
- MC\_Reset (hata onaylama), 459
- MC\_WriteParam (bir teknoloji nesnesinin parametrelerini okuma), 478
- Bellek
- saat belleği, 101
  - I (görüntü girişini işleme), 107
  - L (lokal bellek), 105
  - Yükleme belleği, 99
  - M (bit bellek), 108
  - bellek kullanımını izleme, 792
  - çevresel giriş adresleri (kuvvet çizelgesi), 801
  - Q (proses görüntü çıkışı), 107
  - kalıcı bellek, 99
  - sistem belleği, 101
  - geçici bellek, 108
  - iş belleği, 99
- Bellek alanları
- Boolean veya bit değerleri adresleme, 106
  - Anında erişim, 105
  - Proses görüntüsü, 105
- Bellek kartı, 978
- başlangıç parametrelerini konfigüre etme, 127
- kayıp bir şifre için boş transfer kartı, 136
- Firmware güncelleme, 133
- uyumsuzluk hatası, 784
- CPU'ya takma, 125
- kayıp şifre, 136
- operasyon, 124
- genel bakış, 124
- program kartı, 131
- transfer kartı, 128
- Bellek yerleşimleri,105, 106
- Mesaj
- sonu, 679
  - uzunluk, 681
  - başlangıcı, 677
- Mesaj konfigürasyonu
- komutları, 702
  - alma, 676
  - aktarma, 675
- MicroMaster sürücü
- bağlama, 724
- MID (karakter string'inin orta karakterlerini okuma), 308
- MIN (minimumu almak), 234
- Çeşitli PtP parametre hataları, 685
- Mobil cihazlar
- Web Sayfa düzeni, 610
- MOD (bölme işleminin kalanını döndürme), 231
- Modbus
- fonksiyon kodları, 727
  - MB\_CLIENT ( Modbus TCP istemci olarak PROFINET yoluyla haberleşme), 730
  - MB\_COMM\_LOAD (Modbus RTU için PtP modülünde port konfigüre etme ), 747
  - MB\_MASTER (Modbus master olarak PtP port ile haberleşme), 749
  - MB\_SERVER (Modbus TCP sunucu olarak PROFINET ile haberleşme), 736
  - MB\_SLAVE (Modbus slave olarak PtP port yoluyla haberleşme), 755
  - bellek adresleri, 728
  - Modbus slave örneği, 764
  - ağ istasyon adresleri, 728
  - RTU haberleşme, 728
  - versiyonlar,42 42, 746
- Düzenleme
- program editörü durumu, 797
  - Web sunucudan değişkenler, 620
  - gözetleme çizelgesi, 798
- Modül bilgi standart Web Sayfa, 616
- Modüller
- haberleşme kartları (CB), 29

haberleşme modülü (CM), 29  
 haberleşme işlemcisi (CP), 29  
 kıyaslama çizelgesi, 26  
   parametreleri konfigüre etme, 147  
 sinyal kartı(SB), 29  
 sinyal modülleri(SM), 29  
 termal bölge, 5050, 54  
 ModuleStates, 353  
 Monitör  
   bir DB'nin değerlerinin yakalanması, 797  
   bir DB'nin başlama değerlerinin resetlenmesi, 797  
 İzleme  
   döngü süresi, 792  
   kuvvet operasyonu, 802  
   kuvvet Çizelgesi, 801  
   LAD durumu,796, 797  
   bellek kullanımı, 792  
   gözetleme çizelgesi,796, 798  
 Programı izleme, 199  
 Web sunucudan değişkenleri izleme, 620  
 Hareket kontrolü  
   eksen konfigürasyonu, 449  
   donanım ve yazılım sınır anahtarları, 483  
   homing (aktif homing için sıra), 490  
   homing konfigürasyon parametreleri, 488  
   eksen homing, 486  
   MC\_ChangeDynamic (eksen için dinamik ayarlamayı değiştirme), 476  
   MC\_CommandTable, 473  
   MC\_Halt (ekseni duraklat), 462  
   MC\_Home (ekseni yatağına getirme), 460  
   MC\_MoveAbsolute (ekseni mutlak olarak konumlandırma), 464  
   MC\_MoveJog (jog modunda eksen hareket ettirme), 471  
   MC\_MoveRelative (ekseni bağıl olarak konumlandırma), 466  
   MC\_MoveVelocity (ekseni ön tanımlı hızda hareket ettirme, 468  
   MC\_Power (eksenleri serbest bırak/tut), 456  
   MC\_ReadParam (bir teknoloji nesnesinin parametrelerini okuma), 480  
   MC\_Reset (hata onaylama), 459  
   MC\_WriteParam (bir teknoloji nesnesinin parametrelerini okuma), 478  
   genel bakış, 441  
 Hareket kontrol komutları, 455  
 Montaj  
   hava akışı, 50  
   açıklık, 50  
   haberleşme kartı (CB), 58  
   haberleşme modülü (CM), 62  
   soğutma, 50  
   CPU, 56

boyutlar, 54  
 uzatma kablosu, 64  
 topraklama, 72  
 kılavuzlar, 49  
 indüktif yükler, 74  
 izolasyon, 71  
 lamba yükler, 73  
 genel bakış, 55  
 sinyal kartı(SB), 58  
 sinyal modülü(SM), 60  
 terminal blok konektör, 63  
 termal bölge, 50, 54  
   kablaj kılavuzları, 70,72  
 MOVE (değeri taşıma), 238  
 MOVE\_BLK (bloğu taşıma), 238  
 Hareket sırası (MC\_CommandTable), 473  
 MRES  
   operatör paneli, 44  
 MUL (çarp), 230  
 Çoklu-düğümülü bağlantılar  
   bağlantı tipleri, 591  
   Ethernet protokolleri, 591  
 Çoklu AWP değiştirilebilir tanımlamalar, 638  
 MUX (çoklama), 278  
 Dokümantasyon yöneticisi, 4

## N

N (negatif sinyal kenarındaki operand taraması), 209  
 N\_TRIG (negatif sinyal kenarı için RLO tarama), 210  
 N= kutu ve N bobin(negatif sinyal kenarındaki operand ayarı), 209  
 NEG (ikinin tamamlayanını oluşturma), 232  
 Gömme derinliği, 77  
 Ağ haberleşmesi, 550  
   Ön gerilim ve sonlandırma kablosu, 670  
 Ağ bağlantısı  
   konfigürasyonu, 149  
   çoklu CPU'lar,553, 554, 558, 573, 578  
 Ağ zaman protokolü(NTP), 165  
 Restart yok, 81  
 NORM\_X (normalize), 251  
 Analog değerlerin normalize edilmesi, 253  
 NOT\_OK (geçersizliği kontrol etme), 228  
 Sayılar  
   binary, 111  
   tam sayı, 112  
   gerçek, 112

## O

OB, (Organizasyon bloğu)  
 Off-gecikme (TOF),212

- OK (geçerliliği kontrol etme), 228  
On-gecikme gecikmesi (TON), 212  
On-gecikme kalıcı (TONR), 212  
Online (çevrim içi)  
bir IP adresi atama, 158  
bir DB'nin değerlerinin yakalanması, 797  
kıyaslama ve senkronize etme, 794  
döngü süresi, 792  
tanılama ara belleği, 793  
kuvvet, 801  
kuvvet operasyonu, 802  
online duruma geçiş, 786  
IP adresi, 789  
bellek kullanımı, 72  
monitör, 796  
çalıştırma paneli, 792  
operatör paneli, 44  
bir DB'nin başlama değerlerinin resetlenmesi, 797  
RUN/STOP butonları, 44  
durum, 796, 797  
günün saati, 789  
araçlar, 795795  
gözetleme çizelgesi, 796, 797, 798  
Online ve diyagnostik araçları  
RUN modunda yükleme, 803  
Online cihaz isimleri  
PROFINET IO, 787  
OPC, konfigürasyon, 771  
Açık Kullanıcı haberleşmesi  
TRCV\_C ile bir bağlantı kurma ve veri okuma, 520  
TSEND\_C ile bir bağlantı kurma ve veri gönderme, 520  
açık kullanıcı haberleşme komutları dönüş değerleri, 549  
Çalışma modu, 44  
STOP/RUN değişime, 792  
CPU çalışma modları, 81  
Operatör paneli, 44  
CPU çalışma modları, 81  
Operatör panelleri, 32  
Optimize edilmiş veri blokları, 177  
VEYA (lojik operasyon), 275  
Sipariş numaraları  
haberleşme ara yüzleri (CM, CB ve CP), 991  
konektör blokları, 993  
konektörler ve terminal bağlantıları, 991  
CPU'lar, 989  
CSM 1277 Ethernet anahtarı, 992  
dokümantasyon, 994  
uç tutucu, 993  
genişleme kabloları, 993  
HMI basic panelleri, 992  
bellek kartları, 992  
PM 1207 güç kaynağı, 992  
programlama yazılımı, 994  
sinyal kartları, batarya kartları, 990  
sinyal modülleri, 989  
simülatörler, 993  
STEP 7, 994  
görselleştirme yazılımı, 994  
WinCC, 994  
Organizasyon bloğu  
çağır, 85  
kullanıcı programı içerisinde kod bloklarının çağırılması, 172  
operasyon konfigüre etme, 174  
oluşturma, 174  
fonksiyon, 85  
know-how koruma, 193  
doğrusal ve yapılandırılmış programlama, 170  
çoklu döngüsel, 174  
genel bakış, 77  
öncelik sınıfları, 85  
işleme, 172172  
başlangıç işlemi, 84  
OUT\_Range (değer aralığı dışında), 227  
Çıkış parametreleri, 175  
darbe kanallarını konfigüre etme, 367  
darbe çıkışlar, 365  
**P**  
P (pozitif sinyal kenarındaki operand taraması), 209  
P\_TRIG (pozitif sinyal kenarı için RLO tarama), 210  
P= kutu ve P bobini (pozitif sinyal kenarı için operand ayarı, 209  
Paneller (HMI), 32  
Parametre atama, 175  
Parametrelerin konfigürasyonu  
LENGH ve BUFFER için SEND\_PTP, 696  
alma, 557  
aktarma, 556  
Parite, 673  
Aktif/pasif haberleşme  
partnerlerin konfigürasyonu, 150, 593  
bağlantı ID'leri, 514  
parametreler, 153  
Şifre koruması  
CPU'ya erişim, 190  
bir CPU veya bellek kartına bağlama, 195  
kod bloğu, 193  
kopya koruması, 195

- CPU, 190
  - boş transfer kartı, 136
  - ENDIS\_PW (aktif olmayan şifreyi etkinleştirme),259
  - kayıp şifre, 136
- PID
  - genel bakış, 419
  - PID\_3STEP (Valfler için ayarlamalı PID kontrolör), 428
  - PID\_3Step algoritma, 419
  - PID\_Compact (entegreli ayarlaması olan üniversal PID kontrolör), 422
  - PID\_Compact algoritma,419
- PLC
  - modüller ekleme, 141
  - çevrim içi bir CPU için bir IP adresi atama, 158
  - haberleşme yükü, 98
  - kıyaslama ve senkronize etme, 74
  - çevrim içi bir CPU'dan blokların kopyalanması, 197
  - döngü süresi,97, 98
  - döngü süresi,97, 98
  - cihaz konfigürasyonu, 137
  - yükleme, 196
  - uzatma kablosu, 64
  - kuvvet, 801
  - kuvvet operasyonu, 802
  - HSC konfigürasyonu, 418
  - kurulum, 55, 56
  - know-how koruma, 193
  - bellek kartı, 124
  - izleme, 796
  - çalışma modları, 81
    - CPU genel bakış,23
  - güç hesabı, 51
  - RTM (çalışma süresi ölçerler), 291
  - başlangıç işlemi, 84
  - sistem tasarımı, 169
  - etiketler, 105
  - terminal blok konektör, 63
  - zaman senkronizasyon özelliği, 165
  - blokları kullanarak, 170
  - gözetleme çizelgesi, 798
- PM 1207 güç modülü, 892
- Podcast'lar, 4
- İşaretçi
  - işaretçi gözden geçirme, 118
- İşaretçiler
  - Herhangi veri tipi, 119
  - İşaretçi veri tipi, 118
  - Variant veri tipi, 120
- Noktadan-Noktaya haberleşme, 671
- Noktadan-Noktaya programlama, 702
- Yoklama yapısı, 703
- Port konfigürasyonu,672
  - hatalar, 687
  - komutlar, 702
  - PtP örnek program, 705
- Port numarası, 517
- Port numaraları
  - sınırlandırılmış, 549
- PORT\_CFG (haberleşme parametrelerini dinamik olarak konfigüre etme), 686
- Portal görünüş, 37
- Potensiyometre modülü
  - özellikler, 980
- Güç hesabı, 51
  - örnek,986
  - hesaplamalar için form, 987
  - genel bakış, 985
- Güç ihtiyacı
  - güç ihtiyacının hesabı, 986
  - hesaplamalar için form, 987
  - güç hesabı, 985
- Güç kaynağı modülü
  - PM1207, 982
- Öncelik
  - öncelik sınıfı, 85
  - işlem sırasında öncelik, 94
- Proses görüntüsü
  - kuvvet, 801
  - kuvvet operasyonu, 802
  - monitör,796, 797
  - durum,796, 797, 801
- PROFIBUS
  - CM 1243-5 (DP master) modül ekleme, 572
  - DP slave ekleme, 572
  - adres, 574
  - adres, konfigüre etme, 574
  - CM 1242-5 (DP slave) module, 570
  - CM 1243-5 (DP master) module, 570
  - dağıtım I/O komutları, 314
  - DPNRM\_DG (bir DP slave'den tanılama veri okumak), 328
  - DPRD\_DAT (bir DP standart slave için uyumlu veriyi okuma), 326
  - DPWR\_DAT (bir DP standart slave için uyumlu veriyi yazma), 326
  - GET (uzaktaki bir CPU'dan veri okuma), 587
  - master, 569
  - ağ bağlantısı,149, 573
  - PROFIBUS adres, 574
  - PROFIBUS adres özellikleri,574
  - PUT (uzaktaki bir CPU'ya veri yazma), 587
  - RALRM (interrupt almak), 318
  - RDREC (veri kaydı okuma),315
  - S7 bağlantısı, 591
  - slave, 569
  - WRREC (veri kaydı yazma), 315
- PROFIBUS adresi, 574



- Profil OB, 94
- PROFINET
- ad hoc modu, 518
  - CPU ve HMI cihazı arasındaki haberleşmenin konfigürasyonu, 552
  - IP adresi konfigürasyonu, 142
  - bağlantı ID'leri, 514
  - CPU-CPU haberleşmesi, 553
  - cihaz isimlendirme ve adresleme, 166
  - dağıtımlı I/O komutları, 314
  - DPRD\_DAT (bir DP standart slave için uyumlu veriyi okuma), 326
  - DPWR\_DAT (bir DP standart slave için uyumlu veriyi yazma), 326
  - Ethernet adresi özellikleri, 160
  - GET (uzaktaki bir CPU'dan veri okuma), 587
  - IP adresi, 159
  - IP adresi atama, 167
  - MAC adresi, 159
  - ağ bağlantısı, 149, 553, 554, 558
  - haberleşme bağlantılarının sayısı, 511
  - genel bakış, 517
  - PLC-PLC haberleşme, 553
  - PUT (uzaktaki bir CPU'ya veri yazma), 587
  - RALRM (interrupt almak), 318
  - RDREC (veri kaydı okuma), 315
  - S7 bağlantısı, 591
  - sistem başlangıç süresi, 166
  - bir ağın testi, 162
  - zaman senkronizasyonu, 142
  - zaman senkronizasyon özelliği, 165
  - haberleşme tipleri, 509
  - WRREC (veri kaydı yazma), 315
- PROFINET komutları
- T\_CONFIG (ara yüzü konfigüre etme), 41
  - TCON (haberleşme bağlantısı yapma), 527
  - TDISCON (haberleşme bağlantısı sonlandırma), 527
  - TRCV (haberleşme bağlantısı yoluyla veri alma), 527
  - TRCV\_C, 556
  - TRCV\_C (Ethernet yoluyla veri alma(TCP)), 520
  - TSEND (haberleşme bağlantısı yoluyla veri gönderme), 527
  - TSEND\_C (Ethernet yoluyla veri gönderme(TCP)), 520
  - TURCV (Ethernet yoluyla veri alma(UDP)), 535
  - TUSEND (Ethernet yoluyla veri gönderme(UDP)), 535
- PROFINET IO
- Ekleme, bir cihazı, 558
  - Atama, bir CPU'ya, 559
  - Atama, cihaz isimlerini, 559
  - Atama, cihaz isimlerini online atama, 787
  - Cihaz isimleri, 559
  - Cihazlar, 558
  - Online cihaz isimleri, 787
- PROFINET RT, 517
- Program
- bir CPU veya bellek kartına bağlama, 195
  - kullanıcı programı içerisinde kod bloklarının çağırılması, 172
  - bir DB'nin değerlerinin yakalanması, 797
  - Çevrim içi bir CPU'dan blokların kopyalanması, 197
  - yükleme, 196
  - doğrusal ve yapılandırılmış programlar, 170
  - bellek kartı, 124
  - organizasyon blokları (OB'ler), 172
  - şifre koruması, 193
  - öncelik sınıfı, 85
  - bir DB'nin başlama değerlerinin resetlenmesi, 797
- Program kartı
- başlangıç parametrelerini konfigüre etme, 127
  - oluşturma, 131
  - CPU'ya takma, 125
  - operasyon, 124
  - genel bakış, 124
- Program kontrol (SCL), 267
- CASE, 269
  - CONTINUE, 273
  - EXIT, 273
  - FOR, 270
  - GO TO, 274
  - IF-THEN, 268
  - REPEAT, 272
  - RETURN, 274
  - WHILE, 271
- Program döngüsü OB, 86
- Program editörü
- bir DB'nin değerlerinin yakalanması, 797
  - monitör, 797
  - bir DB'nin başlama değerlerinin resetlenmesi, 797
  - durum, 797
- Program yürütme, 77
- Program bilgisi
- Çağırma yapısında, 201
- Program yapısı, 172
- kod bloklarını çağırma, 172
- Programlama
- LAD veya FBD komutlarına girişler ve çıkışlar ekleme, 4141
  - bir CPU veya bellek kartına bağlama, 195

blok çağrılmalar, 77  
kullanıcı programı içerisinde kod bloklarının çağrılması, 172  
kod bloklarını kıyaslama ve senkronize etme, 794  
veri bloğu (DB), 77  
editörler arasında sürükley ve bırak, 43  
genişletilebilir komutlar, 42  
favoriler, 38  
FBD (fonksiyon blok diyagramı), 181  
fonksiyon (FC), 174  
fonksiyon bloğu (FB), 77, 175  
bir FB'nin başlangıç değeri, 175  
komutların yerleştirilmesi, 38  
instance veri bloğu (DB), 175  
LAD (merdiven), 180  
doğrusal program, 170  
CPU çalışma modları, 81  
PID genel bakış, 419  
PID\_3STEP (valfler için ayarlanmalı PID kontrolör), 428  
PID\_3Step algoritma, 419  
PID\_Compact (entegreli ayarlaması olan universal PID kontrolör), 422  
PID\_Compact algoritma, 419  
güç akışı (EN ve ENO), 188  
öncelik sınıfı, 85  
PtP komutları, 702  
RTM (çalışma süresi ölçerler), 291  
SCL (yapısal kontrol dili), 181, 183  
yapılandırılmış program, 170  
sistem saati, 287  
kod bloklarının tipleri, 77  
takılmamış modüller, 47  
tanımsız CPU, 140  
geçerli FC, FB ve DB sayıları, 77  
Kullanıcı-tanımlı Web Sayfa dil anahtarı programlama, 657  
Proje  
erişim koruması, 190  
bir CPU veya bellek kartına bağlama, 195  
kıyaslama ve senkronize etme, 794  
yükleme, 196  
boş transfer kartı, 136  
kayıp şifre, 136  
program kartı, 131  
bir kod bloğunu koruma,  
bir CPU için sınırlandırılmış erişim, 193  
transfer kartı, 128  
Proje görünüşü, 37  
Koruma sınıfı, 834  
Koruma seviyesi  
bir CPU veya bellek kartına bağlama, 195  
kod bloğu, 193  
CPU, 190

kayıp şifre, 136  
Protokol  
haberleşme, 671  
boş port, 671  
ISO on TCP, 517  
Modbus, 671  
PROFINET RT, 517  
TCP, 517  
UDP, 517  
USS, 661  
PTO (darbe katar çıkışı)  
zorlanamaz, 802  
darbe kanallarını konfigüre etme, 367  
CTRL\_PWM (darbe genişlik modülasyonu), 364  
operasyon, 365  
PtP haberleşme, 671  
parametreleri konfigüre etme, 675  
portların konfigürasyonu, 672  
örnek program, 704  
örnek program konfigürasyonu, 705  
örnek program, çalışma, 712  
örnek program, STEP 7 programlama, 710  
programlama, 702  
terminal emülatör için örnek program, 711  
PtP hata sınıfları, 685  
PtP komut dönüş değerleri, 684  
OB modüllerini çek veya tak, 9191  
Darbe yakalama, 146, 147  
Darbe yakalama bitleri, dijital giriş konfigürasyonu, 146  
Darbe gecikmesi (TP), 212  
Darbe çıkışlar, 365  
PUT  
bağlantıyı konfigüre etme, 151  
PUT (uzaktaki bir CPU'ya veri yazma), 587  
PWM (darbe genişlik modülasyonu)  
zorlanamaz, 802  
darbe kanallarını konfigüre etme, 367  
CTRL\_PWM (darbe genişlik modülasyonu), 364  
operasyon, 365

## Q

Q bellek  
darbe kanallarını konfigüre etme, 367  
darbe çıkışlar, 365  
QRY\_CINT (döngüsel interrupt parametrelerinin ayarı), 336  
QRY\_DINT (zaman-gecikmesi interrupt durumu sorgulama), 342  
QRY\_TINT (günün saati interrupt durumu sorgulama), 340  
Sıraya koyma, 94  
Alıntı işaret teamülleri, Web sunucu, 639

**R**

R (çıkış resetleme ), 206  
R\_TRIG (pozitif sinyal kenarına etiket ayarı), 210  
Şasi veya istasyon arızası OB, 92  
RALRM, 322  
RALRM (interrupt almak), 318  
Anma gerilimleri, 834, 834  
RCV\_CFG (seri alma parametrelerini dinamik olarak konfigüre etme), 689  
RCV\_PTP (mesaj almayı etkinleştirme), 697  
RCV\_RST (alıcı ara belleği silme), 698  
RD\_ADDR (donanım belirtecinden IO adreslerini tanımla), 406  
RD\_LOC\_T (yerel saati okuma), 287  
RD\_SYS\_T (günün saatini okuma), 287  
RDREC, 322  
RDREC (veri kaydı okuma), 315  
RE\_TRIGR (döngü izleme süresini yeniden başlatma), 262  
READ\_DBL (yükleme belleğindeki veri bloğundan okuma), 401  
HTTP değişkenlerini okuma, 631  
Konfigürasyon hatalarını alma, 693  
Mesaj konfigürasyonunu alma, 676  
PtP örnek program, 706  
Parametrelerin konfigürasyonunu alma, 557  
Runtime dönüş değerlerini alma, 697  
Tarif  
DB yapısı, 370  
örnek program, 378  
genel bakış, 369  
RecipeExport (tarif alma), 374  
RecipeImport (tarif alma), 376  
Enum tiplerini referanslama, kullanıcı-tanımlı Web sayfası, 635  
Kullanıcı-tanımlı Web sayfası yenileme, 627  
Elektrik servis ömrünü aktarma, 835  
REPLACE (bir karakter string'indeki karakterleri değiştirme), 312  
Modülleri değiştirme, 47  
Gereksinimler, kurulum, 36  
Reset zamanlayıcı(RT), 212  
Fabrika ayarlarına reset, 789  
RESET\_BF (reset bit alanı), 207  
Bir DB'nin başlama değerlerinin resetlenmesi, 797  
Sınırlanmış TSAP'lar ve port numaraları, 549  
RET (döndür), 258  
Kalıcı blok etiketleri  
yükleme, RUN modunda, 809  
Kalıcı bellek, 24, 99  
CPU 1211C, 837

CPU 1212C, 847  
CPU 1214C, 857  
CPU 1215C, 867  
CPU 1217C, 879  
Dönüş değerleri  
Açık kullanıcı haberleşme komutları, 549  
PtP komutları, 684  
Ters gerilim koruması, 835  
RIGHT (karakter string'inin sağ karakterini okuma), 308  
ROL (sola döndür) ve ROR (sağa döndür), 282  
ROUND (sayısal değeri yuvarla), 249  
Router IP adresi, 160  
RS (reset/set flip-flop), 207  
RS232 ve RS485 haberleşme modülleri, 669  
RS485 konektör  
sonlandırma ve ön gerilim, 670  
RT (reset zamanlayıcı), 212  
RTS, 674  
RTS daima işlemde, 674  
RTS Off gecikme, 676  
RTS On gecikme, 676  
RTS anahtarlanır, 674  
Hareket sırasına göre eksen komutlarını RUN etme(MC\_CommandTable), 473  
RUN modu,81, 85, 792  
kuvvet operasyonu, 802  
operatör paneli, 44  
araç çubuğu butonları, 44  
RUN- STOP geçişi, 104  
RUN/STOP butonları, 44  
Çalışma süresi ölçer(RTM), 291

**S**

S (çıkış ayarı), 206  
S\_CONV (karakter string'ini dönüştürme), 293  
S\_MOV (karakter string'ini taşıma), 293  
S7 haberleşmesi  
bağlantıyı konfigüre etme, 151  
S7-1200  
erişim koruması, 190  
modüller ekleme, 141  
yeni cihaz ekleme, 138  
hava akışı, 50  
AS-i, 579  
AS-i adresi,579  
AS-i port, 578  
bir DB'nin değerlerinin yakalanması, 797  
açıklık, 50  
haberleşme yükü, 98  
kod bloklarını kıyaslama, 794

- CPU modellerin kıyaslama çizelgesi, 24  
CPU parametrelerini konfigüre etme, 142  
modülleri konfigüre etme, 147  
soğutma, 50  
CPU kurma, 56  
döngü süresi, 98  
cihaz konfigürasyonu, 137  
kayıp bir şifre için boş transfer kartı, 136  
Ethernet portu, 159  
uzatma kablosu, 64  
kuvvet, 801  
kuvvet operasyonu, 802  
topraklama, 72  
HSC konfigürasyonu, 418  
indüktif yükler, 74  
kurulumu, 55  
bir CB kurulumu, 58  
bir CM kurulumu, 62  
bir SB kurulumu, 58  
bir SM kurulumu, 60  
IP adresi, 159  
izolasyon kılavuzları, 71  
know-how koruma, 193  
lamba yükler, 73  
kayıp şifre, 136  
MAC adresi, 159  
modüller, 26  
izleme, 796  
montaj boyutları, 54  
ağ bağlantısı, 149  
çalışma modları, 81  
operasyon, 798  
operatör paneli, 44  
CPU genel bakış, 23  
şifre koruması, 190  
güç hesabı, 51  
PROFIBUS, 574  
PROFIBUS adresi, 574  
PROFINET, 159  
Program kartı, 131  
darbe çıkışlar, 365  
bir DB'nin başlama değerlerinin resetlenmesi, 797  
RUN/STOP butonları, 44  
başlangıç parametreleri, 127  
başlangıç işlemi, 84  
terminal blok konektör, 63  
termal bölge, 50, 54  
TS Adaptör, 26  
kablaj kılavuzları, 70, 72  
SCALE\_X (ölçek), 251  
Analog değerlerin ölçeklenmesi, 253  
Analogların ölçeklenmesi, 39  
Tarama döngüsü  
kuvvet, 801  
kuvvet operasyonu, 802  
genel bakış, 97  
SCL (Yapısal kontrol dili)  
ABS (mutlak değer biçimi), 233  
ACOS (arccosine değeri biçimi), 236  
ACT\_TINT (günün saati interrupt'ını aktif etme), 339  
ADD (ekle), 230  
adresleme, 183  
AND (lojik operasyon), 275  
ASIN (arcsine değeri biçimi), 236  
ATAN (tangent değeri biçimi), 236  
ATH (ASCII string'i hexadecimal sayıya dönüştürme), 304  
ATTACH (bir OB'yi bir interrupt olaya iliştir, 331  
Bit lojik, 203  
bir FB veya FC çağırma, 183  
çağırma blokları, 172  
CAN\_DINT (zaman-gecikmesi interrupt'ını iptal et), 342  
CAN\_TINT (günün saati interrupt'ını iptal et), 339  
CASE, 269  
CEIL (kayan-nokta sayıdan sonraki en yüksek tamsayıyı üretme), 250  
Chars\_TO\_Str (CHAR dizinini karakter string'ine dönüştürme), 302  
değerleri kıyasla, 226  
CONCAT (karakter string'lerini birleştirme), 307  
koşullar, 307  
CONTINUE, 273  
kontrol deyimleri, 183, 267, 268, 269, 270, 271, 272, 273, 274  
CONV (değer dönüştürme), 245  
Dönüştürme komutları, 245  
COS (cosine değeri biçimi), 236  
CTD (aşağı doğru sayma), 220  
CTRL\_HSC (yüksek hızlı sayıcı kontrolü), 409  
CTRL\_PWM (darbe genişlik modülasyonu), 364  
CTU (yukarı doğru sayma), 220  
CTUD (yukarı ve aşağı doğru sayma), 220  
DataLogClose (veri günlüğü kapama), 389  
DataLogCreate (veri günlüğü oluşturma), 382  
DataLogNewFile (yeni dosyada veri günlüğü), 390  
DataLogOpen (veri günlüğü açma), 385  
DataLogWrite (veri günlüğüne yazma), 387  
DEC (azaltma), 232  
DECO (kod çözme), 276  
DELETE (bir karakter string'inde karakterleri silme), 310  
DETACH (bir interrupt olayından bir OB ayırmak), 331  
DeviceStates (bir I/O sisteminin modül durumunun okunması), 347

- DIS\_AIRT (yüksek öncelikli interrupt'ları ve asenkron hata olaylarını yürütmeyi etkisizleştirme), 230
- DIV (böl), 230
- EN ve ENO (güç akışı), 188
- EN\_AIRT (yüksek öncelikli interrupt'ları ve asenkron hata olaylarını yürütmeyi etkinleştirme), 344
- ENCO (kodlama), 276276
- EXIT, 273
- EXP (üstel değer biçimi), 236
- ifadeler, 183
- EXPT (üst almak), 236
- F\_TRIG (negatif sinyal kenarındaki etiket ayarı), 210
- FILL\_BLK (blok doldurma), 242
- FIND (bir karakter string'inde karakterler bulma), 313
- FLOOR (kayan-nokta sayıdan sonraki en yüksek tamsayıyı üretme), 250
- FOR, 270
- FRAC (kesri döndür), 236
- GET (uzaktaki bir CPU'dan veri okuma), 587
- GET\_DIAG (diyagnostik bilgisi okuma), 358
- GOTO, 274
- HTA ( hexadecimal sayıyı ASCII string'e dönüştürme), 304
- IF-THEN, 268
- IN\_Range (değer aralık içerisinde), 227
- INC (artış), 232
- INSERT (bir karakter string'ine karakterler yerleştirme), 311
- INV (birilerin tamamlayanını oluşturma), 276
- JMP\_LIST (atlama listesini tanımlama), 255
- LED (LED durumunu okuma), 346
- LEFT (bir karakter string'inin sol karakterlerini okuma), 308
- LEN (bir karakter string'inin uzunluğunu belirleme), 307
- LIMIT (sınır değeri ayarla), 235
- LN (doğal logaritma biçimi), 236
- LOG2GEO (Donanım belirleyiciden slot tanımlama), 404
- MAX (maksimumu almak), 234
- MAX\_LEN (bir karakter string'inin maksimum uzunluğu), 306
- MB\_CLIENT ( Modbus TCP istemci olarak PROFINET yoluyla haberleşme), 730
- MB\_COMM\_LOAD (Modbus RTU için PtP modülünde port konfigüre etme ), 747
- MB\_MASTER (Modbus master olarak PtP port ile haberleşme), 749
- MB\_SERVER (Modbus TCP sunucu olarak PROFINET yoluyla haberleşme), 736
- MB\_SLAVE (Modbus slave olarak PtP port yoluyla haberleşme), 755
- MC\_ChangeDynamic (eksen için dinamik ayarlamayı değiştirme), 476
- MC\_CommandTable, 473
- MC\_Halt (ekseni duraklat), 462
- MC\_Home (ekseni yatağına getirme), 460
- MC\_MoveAbsolute (ekseni mutlak olarak konumlandırma), 464
- MC\_MoveJog (jog modunda eksen hareket ettirme), 471
- MC\_MoveRelative (ekseni bağıl olarak konumlandırma), 466
- MC\_MoveVelocity (ekseni ön tanımlı hızda hareket ettirme), 468
- MC\_Power (eksenleri serbest bırak/tut), 456
- MC\_ReadParam (bir teknoloji nesnesinin parametrelerini okuma), 480
- MC\_Reset (hata onaylama), 459
- MC\_WriteParam (bir teknoloji nesnesinin parametrelerini okuma), 478
- MID (karakter string'inin orta karakterlerini okuma), 308
- MIN (minimumu almak), 234
- MOD (bölme işleminin kalanını döndürme), 231
- ModuleStates (bir modülün modül durum bilgisini okuma), 353
- MOVE (değeri taşıma), 238
- MOVE\_BLK (bloğu taşıma), 238
- MUL (çarp), 230
- MUX (çoğalt), 278
- NEG (ikinin tamamlayanını oluşturma), 232
- NORM\_X (normalize), 251
- NOT\_OK (geçersizliği kontrol etme), 228
- OK (geçerliliği kontrol etme), 228
- operatörler, 183
- VEYA (lojik operasyon), 275
- OUT\_Range (değer aralık dışında), 227
- genel bakış, 181
- PID genel bakış, 419
- PID\_3STEP (Valfler için ayarlamalı PID kontrolör), 428428
- PID\_3Step algoritma, 419
- PID\_Compact (entegreli ayarlaması olan üniversal PID kontrolör), 419
- PORT\_CFG (haberleşme parametrelerini dinamik olarak konfigüre etme), 686
- operatörlerin önceliği, 183
- program kontrolü, 267

- program editörü, 181  
PUT (uzaktaki bir CPU'ya veri yazma), 587  
QRY\_CINT (döngüsel interrupt parametrelerinin ayarı), 336  
QRY\_DINT (zaman-gecikmesi interrupt durumu sorgulama), 342  
QRY\_TINT (günün saati interrupt durumu sorgulama), 340  
R\_TRIG (pozitif sinyal kenarına etiket ayarı), 210  
RCV\_CFG (seri alma parametrelerini dinamik olarak konfigüre etme), 689  
RCV\_PTP (mesaj almayı etkinleştirme), 697  
RCV\_RST (alıcı ara belleği silme), 698  
RD\_ADDR (donanım belirtecinden IO adreslerini tanımla), 406  
RD\_LOC\_T (yerel saati okuma), 287  
RD\_SYS\_T (günün saatini okuma), 287  
READ\_DBL (yükleme belleğindeki veri bloğundan okuma), 401  
RecipeExport (tarif alma), 374  
RecipeImport (tarif alma), 376  
REPEAT, 272  
REPLACE (bir karakter string'indeki karakterleri değiştirme), 312  
RETURN, 274  
RIGHT (karakter string'inin sağ karakterini okuma), 308  
ROL (sola döndür) ve ROR (sağa döndür), 282  
ROUND (sayısal değeri yuvarla), 249  
RTM (çalışma süresi ölçerler), 291  
S\_CONV (karakter string'ini dönüştürme), 293  
S\_MOV (karakter string'ini taşıma), 293  
SCALE\_X (ölçek), 251  
SEL (seç), 278  
SEND\_CFG (seri aktarma parametrelerin dinamik konfigürasyonu), 688  
SEND\_PTP (ara bellek veri gönderme aktarması), 694  
SET\_CINT (döngüsel interrupt parametrelerinin ayarı), 334  
SET\_TIMEZONE (saat bölgesinin ayarı), 290  
SET\_TINTL (tarih ve gün interrupt'ı süresinin ayarı), 337  
SGN\_GET (RS232 sinyalleri sorgusu), 699  
SGN\_SET (RS-232 sinyallerinin ayarı), 700  
SHL (sola kaydır) ve SHR (sağa kaydır), 281  
SIN (sinüs değer biçimi), 236  
SQR (kare biçimi), 236  
SQRT (karekök biçimi), 236  
SRT\_DINT (zaman gecikmesi interrupt'ını başlat), 342  
Strg\_TO\_Chars (karakter string'ini CHAR dizisine dönüştür), 302  
STRG\_VAL (karakter string'ini sayısal değere dönüştür), 293  
SUB (çıkarma), 230  
SWAP (baytları yer değiştir), 243  
SWITCH (atlama dağıtıcısı), 256  
T\_ADD (süreleri toplama), 284  
T\_COMBINE (süreleri birleştirme), 284  
T\_CONV (süreleri dönüştür ve çıkar), 283  
T\_DIFF (süre farkı), 285  
T\_SUB (çıkarma süreleri), 284  
TAN (tangent değer biçimi), 236  
zamanlayıcılar, 212  
TM\_MAIL (email gönder), 775  
TRUNC (sayısal değeri kırpma), 249  
UFILL\_BLK (doldurma bloğu kesintisiz), 242  
UMOVE\_BLK (hareket bloğu kesintisiz), 238  
USS\_DRV (sürücüyle veri yer değiştirme), 716  
USS\_PORT (USS ağla haberleşme düzenleme), 715  
USS\_RPM (sürücüden parametrelerin okunması), 719  
USS\_WPM (sürücüde parametreleri değiştirme), 720  
VAL\_STRG (sayısal değeri karakter string'ine dönüştürme), 293  
Değişken bölümü, 181  
WHILE, 271  
WR\_LOC\_T (yerel saati ayarla), 287  
WR\_SYS\_T (günün saatini ayarla), 287  
WRIT\_DBL (yükleme belleğine veri bloğu yazma), 401  
WWW (kullanıcı-tanımlı Web sayfasını senkronize etme), 642  
XOR (lojik operasyon), 275  
Güvenlik  
erişim koruması, 190  
bir CPU veya bellek kartına bağlama, 195  
kopya koruması, 195  
CPU, 190  
bir kod bloğu için know-how koruması, 193  
kayıp şifre,  
SEL (seç), 278  
Mesaj konfigürasyonu gönderme, 675  
Parametrelerin konfigürasyonunu gönderme, 150,556, 593  
SEND\_CFG (seri aktarma parametrelerin dinamik konfigürasyonu), 688  
SEND\_PTP (Noktadan-Noktaya veri gönderme)  
LENGH ve BUFFER parametreleri, 696  
SEND\_PTP (ara bellek veri gönderme aktarması), 694  
Seri haberleşme, 671  
Servis ve destek, 3  
SET\_BF (bit alanı ayarı), 207

- SET\_CINT (döngüsel interrupt parametrelerinin ayarı), 334
- SET\_TIMEZONE (saat bölgesinin ayarı), 290
- SET\_TINTL (tarih ve gün interrupt'ı süresinin ayarı), 337
- Ayarlamalar, 43
- SGN\_GET (RS232 sinyalleri sorgusu), 699
- SGN\_SET (RS-232 sinyallerinin ayarı), 700
- SHL (sola kaydır) ve SHR (sağa kaydır), 281
- Siemens güvenlik sertifikası, Web sayfaları, 613, 666
- Siemens teknik desteği, 3
- Siemens\_automation\_language Cookie, 657
- Sinyal kartları (SB)
- modüller ekleme, 141
  - analog çıkış temsili (akım), 918, 952
  - analog çıkış temsili (gerilim), 917, 951
  - parametrelerin konfigürasyonu, 147
  - giriş temsili (akım), 917, 951
  - giriş temsili (gerilim), 916, 950
  - kurulum, 58
  - genel bakış, 29
  - güç ihtiyacı, 985
  - sökme, 5858
  - SB 1221 DI 4 x 24 VDC, 200 kHz, 935
  - SB 1221 DI 4 x 5 VDC, 200 kHz, 935
  - SB 1222 DQ 4 x 24 VDC, 200 kHz, 937
  - SB 1222 DQ 4 x 5 VDC, 200 kHz, 937
  - SB 1223 DI 2 x 24 VDC, DQ 2 x 24 VDC, 937
  - SB 1223 DI 2 x 24 VDC/DQ 2 x 24 VDC, 200 kHz, 940
  - SB 1223 DI 2 x 5 VDC/DQ 2 x 5 VDC, 200 kHz, 940
  - SB 1231 AI 1 x 12 bit, 946
  - SB 1231 AI 1 x 16 bit RTD, 957
  - SB 1231 AI 1 x 16 bit Termokupl, 953
  - SB 1232 AQ 1 x 12 bit, 948
- Sinyal işleme hataları, 700, 701
- Sinyal modülleri(SM)
- modüller ekleme, 141
  - analog çıkış temsili (akım), 917, 951
  - analog çıkış temsili (gerilim), 916, 950
  - analog çıkış temsili (akım), 918, 952
  - analog çıkış temsili (gerilim), 917, 951
  - parametrelerin konfigürasyonu, 147
  - uzatma kablosu, 64
  - kurulum, 60
  - genel bakış, 29
  - güç ihtiyacı, 985
  - sökme, 61
  - SM 1221 DI 16 x 24 VDC, 892
  - SM 1221 DI 8 x 24 VDC, 892
  - SM 1222 DQ 16 x 24 VDC, 895
  - SM 1222 DQ 16 x Röle, 895
  - SM 1222 DQ 8 Röle Değiştirme, 894
  - SM 1222 DQ 8 x 24 VDC, 894
  - SM 1222 DQ 8 x Röle, 894
  - SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC, 900
  - SM 1223 DI 16 x 24 VDC, DQ 16 x Röle, 900
  - SM 1223 DI 8 x 120/230 VAC/DQ 8 x Röle, 904
  - SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC, 900
  - SM 1223 DI 8 x 24 VDC, DQ 8 x Röle, 900
  - SM 1231 AI 4 x 13 bit, 907
  - SM 1231 AI 4 x 16 bit, 907
  - SM 1231 AI 4 x 16 bit TC, 919
  - SM 1231 AI 4 x RTD x 16 bit, 925
  - SM 1231 AI 8 x 13 bit, 907
  - SM 1231 AI 8 x 16 bit TC, 919
  - SM 1231 AI 8 x RTD x 16 bit, 925
  - SM 1232 AQ 2 x 14 bit, 910
  - SM 1232 AQ 4 x 14 bit, 910
  - SM 1234 AI 4 x 13 bit/AQ 2 x 14 bit, 913
  - SM 1278 4xIO-Link Master, 931
  - basamak tepki süreleri, 916
- Simülatörler, 978
- SIN (sinüs değer biçimi), 236
- Slave yoklama yapısı, 703
- SM 1231 RTD
- seçim çizelgesi, 928, 960
- SM and SB
- kıyaslama çizelgesi, 26
  - cihaz konfigürasyonu, 137
- SMS, 770770
- İndüktif yükler için snubber devreleri, 74
- Yazılım akış kontrol, 675
- Özel karakterler
- Kullanıcı-tanımlı Web sayfaları, 639
- Özellikler
- analog çıkış temsili (akım), 917, 951
  - analog çıkış temsili (gerilim), 916, 950
  - analog çıkış temsili (akım), 918, 952
  - analog çıkış temsili (gerilim), 917, 951
  - onaylar, 829
  - BB 1297, 962
  - CB 1241 RS485, 974
  - CM 1241 RS232, 975
  - CM 1241 RS422/485, 976
  - CPU 1211C AC/DC/Röle, 837
  - CPU 1211C DC/DC/DC, 837
  - CPU 1211C DC/DC/Röle, 837
  - CPU 1212C AC/DC/Röle, 847
  - CPU 1212C DC/DC/DC, 847
  - CPU 1212C DC/DC/Röle, 847

- CPU 1214C AC/DC/Röle, 857  
CPU 1214C DC/DC/DC, 857  
CPU 1214C DC/DC/Röle, 857  
CPU 1215C AC/DC/Röle, 867  
CPU 1215C DC/DC/DC, 867  
CPU 1215C DC/DC/Röle, 867  
CPU 1217C DC/DC/DC, 879  
Elektromanyetik uyumluluk (EMC), 832  
çevre koşulları, 833  
genel teknik özellikler, 829  
endüstriyel çevreler, 831  
giriş simülatörleri, 978  
bellek kartları, 978  
potansiyometre modülü, 980  
Anma gerilimleri, 834  
SB 1221 DI 4 x 24 VDC, 200 kHz, 935  
SB 1221 DI 4 x 5 VDC, 200 kHz, 935  
SB 1222 DQ 4 x 24 VDC, 200 kHz, 937  
SB 1222 DQ 4 x 5 VDC, 200 kHz, 937  
SB 1223 DI 2 x 24 VDC, DQ 2 x 24 VDC, 943  
SB 1223 DI 2 x 24 VDC/DQ 2 x 24 VDC, 200 kHz, 943  
SB 1223 DI 2 x 5 VDC/DQ 2 x 5 VDC, 200 kHz, 940  
SB 1231 AI 1 x 12 bit, 946  
SB 1231 AI 1 x 16 bit RTD, 957  
SB 1231 AI 1 x 16 bit Termokupl, 953  
SB 1232 AQ 1 x 12 bit, 948  
SM 1221 DI 16 x 24 VDC, 892  
SM 1221 DI 8 x 24 VDC, 892  
SM 1222 DQ 16 x 24 VDC, 895  
SM 1222 DQ 16 x Röle, 895  
SM 1222 DQ 8 Röle Değişirme, 894  
SM 1222 DQ 8 x 24 VDC, 894  
SM 1222 DQ 8 x Röle, 894  
SM 1223 DI 16 x 24 VDC, DQ 16 x 24 VDC, 900  
SM 1223 DI 16 x 24 VDC, DQ 16 x Röle, 900  
SM 1223 DI 8 24 VDC, DQ 8 x Röle, 900  
SM 1223 DI 8 x 120/230 VAC/DQ 8 x Röle, 904  
SM 1223 DI 8 x 24 VDC, DQ 8 x 24 VDC, 900  
SM 1231 AI 4 x 13 bit, 907  
SM 1231 AI 4 x 16 bit, 907  
SM 1231 AI 4 x 16 bit TC, 919  
SM 1231 AI 4 x RTD x 16 bit sinyal modülü, 925  
SM 1231 AI 8 x 13 bit, 919  
SM 1231 AI 8 x 16 bit TC, 919  
SM 1231 AI 8 x RTD x 16 bit sinyal modülü, 925  
SM 1232 AQ 2 x 14 bit, 910  
SM 1232 AQ 4 x 14 bit, 910  
SM 1234 AI 4 x 13 bit/AQ 2 x 14 bit, 913  
SM 1278 4xIO-Link Master, 931  
basamak tepki süreleri (CPU), 842, 852, 862, 873, 888  
basamak tepki süreleri (SB), 950  
basamak tepki süreleri (SM), 916  
SQR (kare biçimi), 236  
SQRT (karekök biçimi), 236  
SR (set/reset flip-flop), 207  
SRT\_DINT (zaman gecikmesi interrupt'ını başlat), 342  
Standart veri blokları, 177  
Standart Web sayfaları, 601  
PC'den erişim, 606  
çalışma modunu değiştirme, 614  
haberleşme, 620  
Cookie sınırlamaları, 666  
Veri günlükleri, 623  
Diyagnostik (tanılama), 616  
Tanıma (kimlik), 615  
Giriş, 613  
JavaScript sınırlamaları, 665  
düzen, 609  
oturum açma ve çıkma, 611  
Modül bilgisi, 616  
erişim güvenliği, 607  
Başla, 614  
Değişken durum, 620  
Başlama koşulları, 677  
Mesaj karakteri başlatma, 677  
Standart Web sayfası başlatma, 614  
POWER ON'dan sonra başlatma, 81  
başlangıç işlemi, 84  
STARTUP modu  
kuvvet operasyonu, 802  
Başlangıç OB, 86  
Başlangıç parametreleri, 127  
Durum  
LED göstericiler, 783  
LED göstericiler (haberleşme ara yüzü), 669  
Durum OB, 93  
STEP 7  
modüller ekleme, 141  
yeni cihaz ekleme, 138  
bir PROFINET IO cihaz ekleme, 558  
bir LAD veya FBD komutunu girişler veya çıkışlara ekleme, 41  
AS-i, 579  
AS-i port, 578  
çevrim içi bir CPU için bir IP adresi atama, 158  
blok çağrılar, 77  
kullanıcı programı içerisinde kod bloklarının çağrılması, 172  
bir DB'nin değerlerinin yakalanması, 797  
ayarları değiştirme, 43  
haberleşme yükü, 98  
kıyaslama ve senkronize etme, 794  
CPU'yu konfigüre etme, 142  
modülleri konfigüre etme, 147  
Çevrim içi bir CPU'dan blokların kopyalanması, 197  
döngü süresi, 97, 98



döngü süresi, 97, 98  
veri bloğu(DB), 77  
cihaz konfigürasyonu, 137  
yükleme, 196  
editörler arasında sürükleyip bırak, 143  
Ethernet portu, 159  
genişletilebilir girişler ve çıkışlar, 42  
favoriler, 38  
kuvvet, 801  
kuvvet operasyonu, 802  
fonksiyon (FC), 174  
fonksiyon bloğu (FB), 77, 175  
HSC konfigürasyonu, 418  
bir FB'nin başlangıç değeri, 175  
komutların yerleştirilmesi, 38  
instance veri bloğu(DB), 175  
doğrusal ve yapılandırılmış programlar, 170  
bellek kartı, 124  
izleme, 796796, 797  
ağ bağlantısı, 149  
çalışma modları, 81  
operasyon, 798  
operatör paneli, 44  
şifre koruması, 193  
Portal görünüşü, 37  
öncelik sınıfı (OB), 85  
PROFIBUS, 574  
PROFINET, 159  
Program kartı, 124  
Proje görünüşü, 37  
bir DB'nin başlama değerlerinin resetlenmesi, 797  
RTM (çalışma süresi ölçer), 291  
RUN/STOP butonları, 44  
başlangıç işlemi, 84  
zaman senkronizasyon özelliği(PROFINET), 165  
kod bloklarının tipleri, 77  
takılmamış modüller, 47  
geçerli FC, FB ve DB sayıları, 77  
STEP 7 programlama  
PtP örnek program, 710  
kullanıcı-tanımlı Web sayfaları, 642  
STEP 7 Web sayfaları, 4  
Stop bitleri, 673673  
STOP modu, 81, 792  
STOP modunda çıkışları etkinleştirme, 800  
kuvvet operasyonu, 802  
operatör paneli, 44  
araç çubuğu butonları, 44  
STP (programdan çıkış), 263  
Strg\_TO\_Chars (karakter sitring'ini CHAR dizisine dönüştür), 302

STRG\_VAL (karakter sitring'ini sayısal değere dönüştür), 293  
String  
S\_MOVE (karakter string'i taşıma), 293  
string veriye genel bakış, 292  
String veri tipi, 115  
string operasyonlarına genel bakış, 306  
Yapılandırılmış programlama, 170, 172  
çağırma blokları, 172  
SUB (çıkart), 230  
Alt ağ maskesi, 160  
Destek, 3  
İndüktif yükler için bastırma devreleri, 74  
Gerilim darbesi ve bağışıklık, 832  
SWAP (baytları yer değiştir), 243  
SWITCH (atlama dağıtıcısı), 256  
Dil değiştirme, kullanıcı-tanımlı Web sayfaları, 656  
Senkronizasyon  
zaman senkronizasyon özelliği(PROFINET), 165  
Sistem saati  
RD\_SYS\_T (günün saatini okuma), 287  
WR\_LOC\_T (yerel saati ayarla), 287  
WR\_SYS\_T (günün saatini ayarla), 287  
Sistem bellek baytı, 102  
Sistem gereksinimleri, 36

## T

T\_ADD (süreleri toplama), 284  
T\_COMBINE (süreleri birleştirme), 285  
T\_CONFIG (ara yüzü konfigüre etme), 541  
T\_CONV (süreleri dönüştürme ve çıkarma), 283  
T\_DIFF (süre farkı), 285  
T\_SUB (çıkarma süreleri), 284  
Etiketler  
kuvvet, 801  
kuvvet operasyonu, 802  
monitör, 796  
durum, 796  
TAN (tangent değer biçimi), 236  
Görev kartları  
sütunlar ve başlıklar, 42, 746  
TCON  
konfigürasyon, 150  
bağlantı ID'leri, 514  
bağlantı parametreleri, 153  
TCON (haberleşme bağlantısı yapma), 527  
TCON\_Param, 153  
TCP  
ad hoc modu, 518  
bağlantı konfigürasyonu, 150

- bağlantı ID'leri, 514
- parametreler, 153
- Protokol, 517
- TCP/IP haberleşme, 517
- TDISCON (haberleşme bağlantısı sonlandırma), 527
- Teknik özellikler, 829
- Teknik destek, 3
- Teknoloji komutları, 409
- Teknoloji modülü
  - SM 1278, 931
- Teknoloji nesneleri
  - Hareket kontrolü, 448
  - PID, 420
- Telekontrol, 767
- TeleService adaptör ve modülü, 66
- Teleservice haberleşme
  - TM\_MAIL (email gönder), 775
- GPRS ile TeleService, 767
- Terminal blok konektör, 63
- PtP örnek program için Terminal emülatörü, 711
- Program testi, 199
- Termal bölge, 50, 54
- Termokupl
  - temel operasyon, 922, 955
  - soğuk birleşim dengelemesi, 953
  - SB 1231 AI 1 x 16 bit, 953
  - SB 1231 Filtre seçim çizelgesi, 955
  - SB 1231 Termokupl Filtre seçim çizelgesi, 955
  - SM 1231 Termokupl Filtre seçim çizelgesi, 923
  - SM 1231 Termokupl seçim çizelgesi, 923
- TIA Portal
  - Portal görünüş, 37
  - Proje görünüşü, 37
- Zaman
  - DTL (tarih ve saat uzun veri tipi), 114
  - RD\_LOC\_T (yerel saati okuma), 287
  - RD\_SYS\_T (günün saatini okuma), 287
  - SET\_TIMEZONE (saat bölgesinin ayarı), 290
  - T\_ADD (süreleri toplama), 284
  - T\_COMBINE (süreleri birleştirme), 285
  - T\_CONV (süreleri dönüştürme ve çıkarma), 283
  - T\_DIFF (süre farkı), 285
  - T\_SUB (çıkarma süreleri), 284
  - Zaman veri tipi, 113
  - TOD (günün saati tipi), 113
  - WR\_LOC\_T (yerel saati ayarla), 287
  - WR\_SYS\_T (günün saatini ayarla), 287
- Zaman gecikme OB, 87
- Zaman hata interrupt OB, 88
- Günün saati
  - çevrim içi CPU konfigürasyonu, 789
- Günün saati OB, 93
- Zaman senkronizasyon özelliği, 165
- Zaman-gecikme interrupt'ları, 342
- Zamanlayıcılar
  - operasyon, 215
  - nicelik, 25, 839, 849, 859, 869, 881
  - RT (reset zamanlayıcı), 212
  - boyut, 25, 839, 849, 859, 869, 881
  - TOF (off-gecikme zamanlayıcısı), 212
  - TON (on-gecikme zamanlayıcısı), 212
  - TONR (on-gecikme kalıcı) zamanlayıcı, 212
  - TP (darbe gecikme zamanlayıcısı), 212
- TimeTransformationRule, yaz saati için, 289
- TM\_MAIL (email gönderme), 775
- Özellikleri izleme, 813
- Transfer (program) kartları, 978
- Transfer kartı, 128
  - başlangıç parametrelerini konfigüre etme, 127
  - kayıp bir şifre için boş transfer kartı, 136
  - CPU'ya takma, 125
  - kayıp şifre, 136
  - operasyon, 124
  - genel bakış, 124
- Aktarma bloğu (T-block), 555
- Konfigürasyon hataları aktarma, 689
- Mesaj konfigürasyonu aktarma, 675
  - PtP örnek program, 705
- Çalışma süresi hatalarını aktarma, 695
- TRCV
  - ad hoc modu, 518
  - bağlantı ID'leri, 514
- TRCV (haberleşme bağlantısı yoluyla veri alma), 527
- TRCV\_C, 556
  - ad hoc modu, 518
  - konfigürasyon, 150
  - bağlantı ID'leri, 514
  - bağlantı parametreleri, 153
  - parametre konfigürasyonu, 557
- TRCV\_C (Ethernet yoluyla veri alma(TCP)), 520
- Gözetleme çizelgesindeki tetikleme değerleri, 799
- Sorun giderme
  - tanılama ara belleği, 793
  - LED gösterciler, 783
- TRUNC (sayısal değeri kırpma), 249
- TS Adaptör, 26
  - bir TS modül kurumu, 66
  - bir DIN rayına kurum, 69
  - bir duvara kurum, 69
  - SIM kartı, 67
- TSAP, 517
- TSAP (taşıma servisi erişim noktaları), 152, 519, 555, 592
- TSAP'lar
  - sınırlandırılmış, 549
- TSEND
  - bağlantı ID'leri, 514

- TSEND (haberleşme bağlantısı yoluyla veri gönderme), 527
- TSEND\_C, 555
- konfigürasyon, 150
  - bağlantı ID'leri, 514
  - bağlantı parametreleri, 153
- TSEND\_C (Ethernet yoluyla veri gönderme(TCP)), 520
- TSEND\_C komut konfigürasyonu, 556
- TURCV
- konfigürasyon, 150
  - bağlantı parametreleri, 153
- TURCV (Ethernet yoluyla veri alma(UDP)), 535
- TUSEND
- konfigürasyon, 150
  - parametreler, 153
- TUSEND (Ethernet yoluyla veri gönderme(UDP)), 535
- ## U
- UDP
- bağlantı konfigürasyonu, 150
  - parametreler, 153
- UDP Protokolü, 517
- UFILL\_BLK (doldurma bloğu kesintisiz), 242
- UMOVE\_BLK (hareket bloğu kesintisiz), 238
- Bilinmeyen CPU versiyon hatası, 784
- Takılmamış modüller, 47
- Tanımsız CPU, 140
- Güncelle OB, 93
- Updating firmware
- STEP 7'den, 791
  - Web sunucudan, 619
- Kullanıcı-tanımlı Web sayfasını güncelleme, 627
- Yükleme
- çevrim içi bir CPU'dan blokların kopyalanması, 197
  - kullanıcı programı, 197
- Kullanıcı ara yüzü
- STEP 7 proje ve portal görüntüleri, 37
- Kullanıcı programı
- LAD veya FBD komutlarına girişler ve çıkışlar ekleme, 41
  - bir CPU veya bellek kartına bağlama, 195
  - kullanıcı programı içerisinde kod bloklarının çağırılması, 172
  - çevrim içi bir CPU'dan blokların kopyalanması, 197
  - yükleme, 196
  - editörler arasında sürük ve bırak, 43
  - genişletilebilir komutlar, 42
  - favoriler, 38
  - komutların yerleştirilmesi, 38
  - doğrusal ve yapılandırılmış programlar, 170
  - bellek kartı, 124
  - organizasyon blokları (OB'ler), 172
  - şifre koruması, 193
  - Program kartı, 124
  - transfer kartı, 124
- Kullanıcı-tanımlı Web sayfaları, 602, 625
- PC'den erişim, 644
  - kontrol DB'den aktif ve devre dışı yapmak, 660
  - S7-1200 veriye erişim için AWP komutları, 627
  - konfigüre etme, 641
  - fragmentların oluşturulması, 637
  - HTML editör ile oluşturma, 626
  - program bloklarını silme, 642
  - yükleme karşılığı DB'ler, 643
  - WWW komutu ile etkinleştirme, 642
  - örnek, 645
  - program bloklarını üretme, 642
  - özel karakterlerin işlenmesi, 639
  - HTML listeleme, 650
  - fragmentların içeri aktarımı, 638
  - yükleme belleği sınırlamaları, 644
  - manuel fragment DB kontrolü, 660
  - çoklu dil konfigürasyonu, 659
  - çoklu dil, 656
  - STEP 7'de programlama, 642
  - özel değişkenleri okuma, 631
  - değişkenleri okuma, 629
  - yenileme, 627
  - özel değişkenleri yazma, 633
  - değişkenleri yazma, 630
- USS protokol kütüphanesi
- genel sürücü kurulum bilgisi, 724
  - genel bakış, 712
  - kullanım için kurallar, 713
  - durum kodları, 722
  - USS\_DRV (sürücüyle veri yer değiştirme), 716
  - USS\_PORT (USS ağ yoluyla haberleşme düzenleme), 715
  - USS\_RPM (sürücüden parametrelerin okunması), 719
  - USS\_WPM (sürücüde parametreleri değiştirme), 720
- ## V
- VAL\_STRG (sayısal değeri karakter string'ine dönüştürme), 293
- Vana PID ayarlama, 428
- Bir dizi için değiştirilebilir dizin, 24
- Değiştirilebilir durum standart Web sayfası, 620

Değişkenler, Web sayfasından izleme ve değiştirme, 620

Komutların versiyonları, 4242, 746

Görselleştirme

HMI cihazları, 32

## W

Bekleme süresi, 673

Sıcak restart, 81

Gözleme çizelgesi

STOP modunda çıkışları etkinleştirme, 800

kuvvet, 199

bellek kartı, 124

monitör, 796

operasyon, 798

tetikleme değerleri, 799

Watchdog (gözlemci) zamanlayıcı (RE\_TRIGR komutu), 262

Web sayfası

STEP 7 servis, destek ve dokümantasyon, 4

Web sunucu, 601

mobil cihazda görüntü, 610

sınırlamalar, 664

etkinleştirme, 603

maksimum HTTP bağlantıları, 664

alıntı işaret teamülleri, 639

standart Web sayfaları, 606

güncelleme hızı, 603

kullanıcı-tanımlı Web sayfaları, 625

Bağlantı diyagramları

CB 1241 RS 485, 974

CPU 1211C, 844

CPU 1212C, 854

CPU 1214C, 864

CPU 1215C, 875

CPU 1217C, 891

SB 1221, 936

SB 1222, 939

SB 1223, 942, 945

SB 1231, 947

SB 1231 RTD, 959

SB 1231 Termokupl, 956

SB 1232, 949

SM 1221, 893

SM 1222, 897

SM 1223, 902, 906

SM 1231, 908

SM 1231 RTD, 927

SM 1231 Termokupl, 921

SM 1232, 912

SM 1234, 915

SM 1278 IO-Link Master, 934

Kablaj kılavuzları, 72

Hava akış ve soğutma için açıklık, 50

topraklama, 72

önşartlar, 70

İş belleği, 24

CPU 1211C, 837

CPU 1212C, 847

CPU 1214C, 857

CPU 1215C, 867

CPU 1217C, 879

WR\_LOC\_T (yerel saati ayarla), 287

WR\_SYS\_T (günün saatini ayarla), 287

WRIT\_DBL (yükleme belleğine veri bloğu yazma), 401

WRREC, 322

WRREC (veri kaydı yazma), 315

WWW ( kullanıcı-tanımlı Web sayfası senkronizesi), 642

## X

x kutu (FBD özel veya lojik operasyon), 203

XON / XOFF, 675

XOR (lojik operasyon), 275