

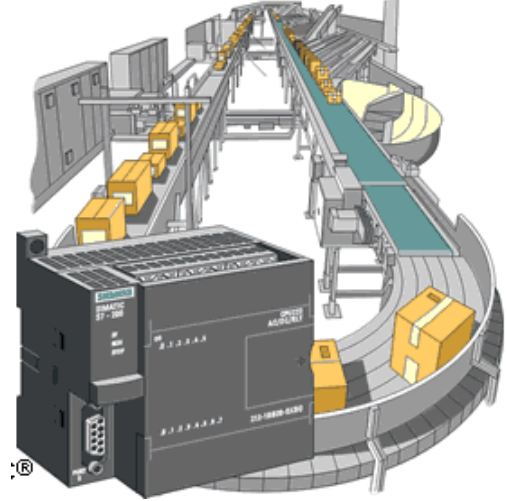
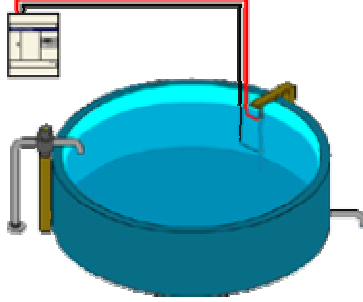
## Siemens S7 200 ve Microwin

### Otomasyon sistemi neye denir ?

Bir otomasyon cihazı, kendisine ulaştırılan bilgiler temelinde, önceden belirlenen hareketleri çevresiyle ilişkili olarak gerçekleştiren bir sistemdir. Hareketler, verilen ve hesaplanan bilgilere yada önceden belirlenmiş parametrelere bağlı kesin bir prosedüre göre gerçekleştirilir.

### Otomasyon sisteminin parçalarına örnekler

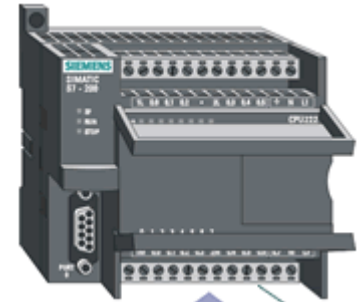
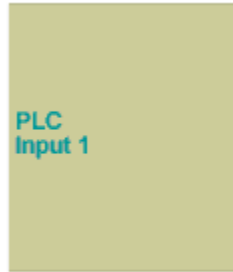
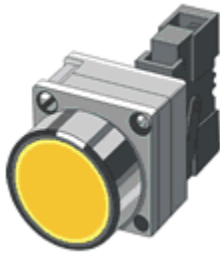
**Makine yada tesis :** Otomatize edilen sistemlerdir. Bu sistem çok karmaşık bir yapıda olabilir, örneğin montaj hattı, üretim atölyesi veya bir fabrika gibi. Daha basit cihazların otomasyonu da mümkün olabilir : trafik lambaları, garaj kapısı, yüzme havuzu veya bir sulama sistemi gibi.



### Giriş birimleri:

Bir otomasyon sisteminin de çevresinde onu bilgisiyle donatacak cihazları olmalıdır.

Bunlara giriş birimleri diyoruz Butonlar, sınır anahtarları, optik, endüktif, kapasitif sensörler ve diğer kontaktör veya röle kontakları gibi. Örneğin, otoyol geçiş gişelerinde otomobili algılamak için foto-elektrik sensörler kullanılır.



Input



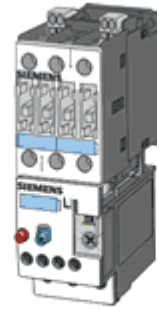
Pushbutton



Switch

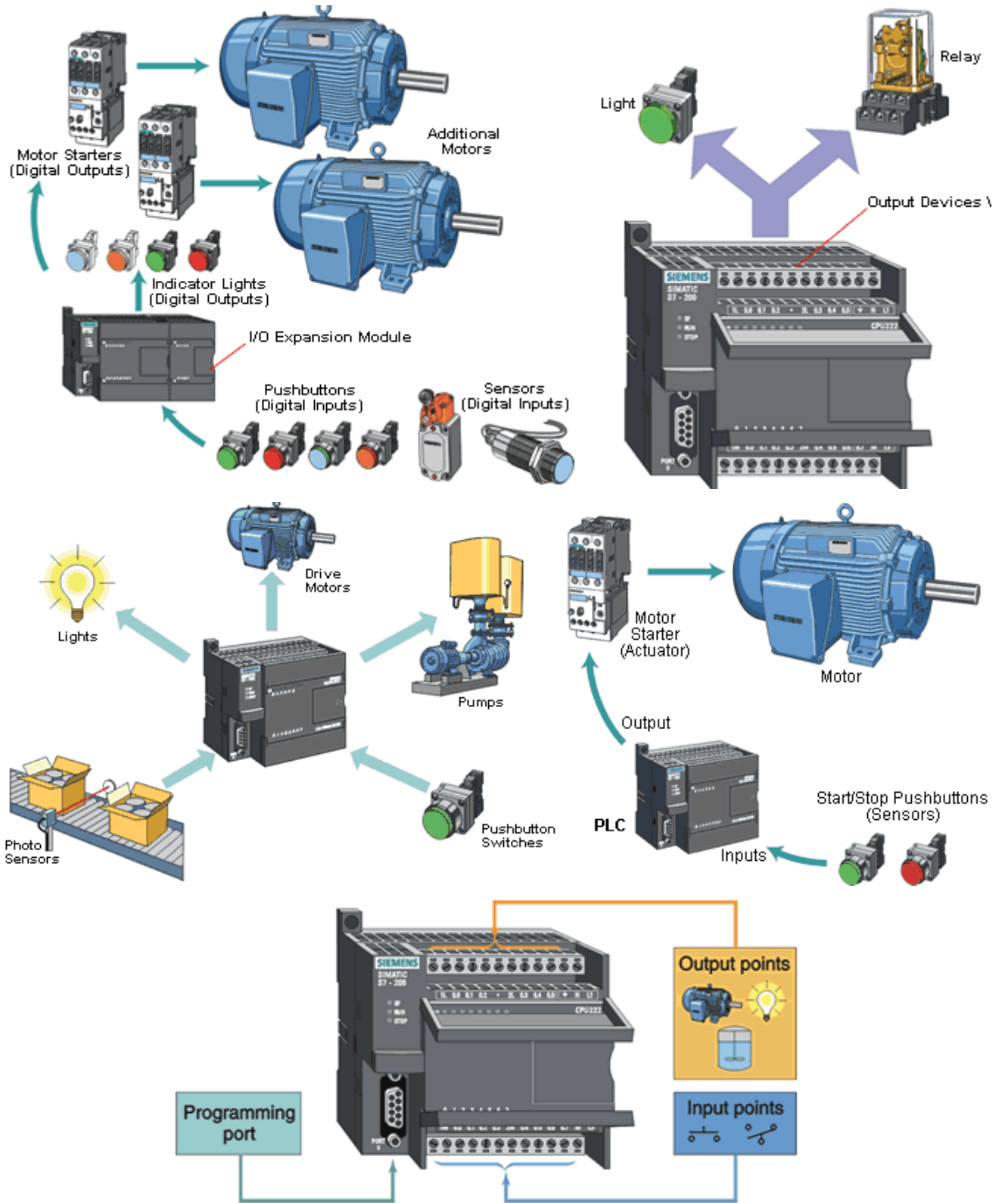
**Aktivatörler :** Aktivatörler sistemde hareketlerin oluşmasını sağlar. Pompalar, silindripler ve motorlar gibi.

**Güç kontrolü :** Aktivatörlere gerekli olan gücü sağlamak ve data işleme sistemiyle arasında aracı olmak için özel cihazlara ihtiyaç vardır. Bunlara güç kontrol sistemleri denir : Kontaktörler, devre kesiciler, röleler, valfler gibi.



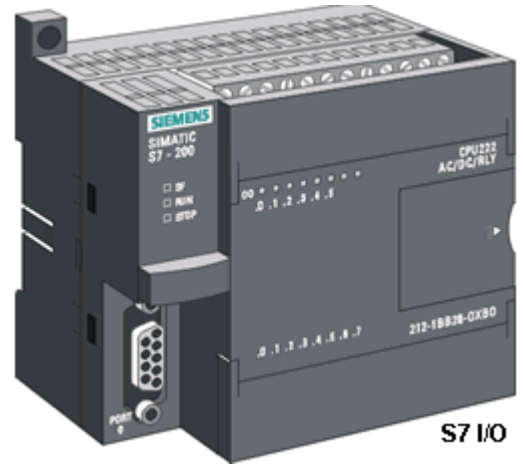
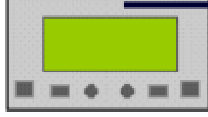
Motor Starter  
(Actuator)





**Data işleme sistemi** : Tesisin beyni data işleme sistemidir, önceleri röleler ve yardımcı kontaktörlerden oluşan bu sistem, şimdi daha güçlü ve kompakt yapıda olan Programlanabilir Kontrol cihazlarıyla kurulmaktadır.

**İnsan / Makina arabirimi** : Bütün otomasyon sistemleri bir operatör tarafından izlenmek ve kontrol edilmek zorundadır. Bunun için, aşağıdaki cihazlara gereksinim duyulur: Kumanda butonu, Arabirim terminalleri, Gösterge üniteleri ve bilgisayarlar.



### Kablolu teknoloji ya da program teknolojisi

#### Kablolu teknolojinin karakteristikleri

Tesisin çalışma şekli, farklı parçalar (röleler, zamanlayıcılar, saatler, vb) arasındaki kablolamayla tanımlanır. Tesis ne kadar kompleks ise, kablolama da o kadar komplekstir. Sonradan eklenen her yenilik için, kablolamayı değiştirmek gereklidir.

#### Program teknolojisinin karakteristikleri

Tesisin çalışma şekli, programlanabilir kontrol cihazı tarafından bir döngü içerisinde yürütülen programla tanımlanır. Sonradan eklenen her yenilik için yapılması gereken sadece programı modifiye etmektir. Artık yeni bir zamanlayıcı yada yardımcı röle ilave etmeğe gerek yoktur. Tek bir cihaz, modüller arası kablolama yoktur, sadece sensörler, aktivatörler ve güç kaynağı için bağlantılar gereklidir.

#### Program teknolojisinin avantajları

**Daha az parça**; Programlanabilir kontrol cihazı bütün zamanlayıcıların, yardımcı rölelerin ve saatlerin yerine geçer. Bu sadece hacimsel bir kazanç sağlamaz, aynı zamanda, otomasyon sisteminin beyninde artık mekanik parçalar kalmadığından güvenilirliği de artırır. Bağlanacak cihazlar sadece : Kontrol cihazı güç kaynağı, sensörler, aktivatörlerdir.

**Esnekliğin artması**: Bir program programlama terminali veya bilgisayar kullanılarak kolayca yazılabilen ve geliştirilebilen emirler dizisidir. Bu program aynı zamanda, benzer otomasyon sistemleri için kolayca kopya edilebilir.

**Daha kolay test edilebilme ve arıza giderme**: Kontrol cihazının ön yüzünde, aşağıdakileri gösteren led'ler göreceksiniz.: Sensörlerin çalışma durumları ( girişlerin açık/kapalı olması ), aktivatörlerin çalışma durumları ( çıkışların çalışıyor/duruyor olması ), Kontrol cihazının kendi çalışma durumu

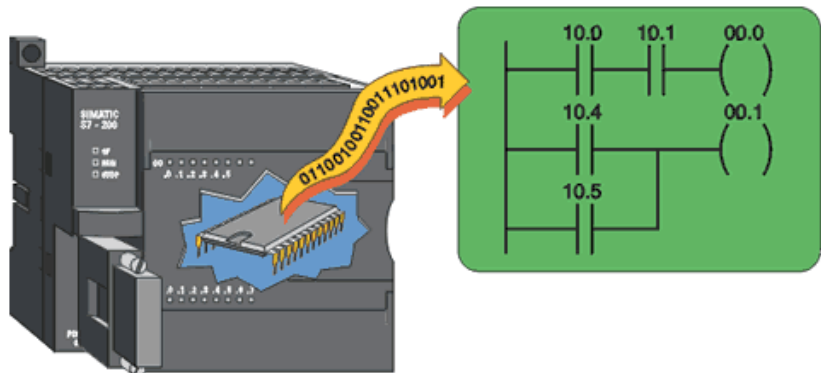
#### Programlanabilir Kontrol Cihazı neye denir?

Programlanabilir Kontrol Cihazı, girişleri vasıtasıyla veri alabilen ve çıkışları vasıtasıyla da komut gönderebilen gerçek bir bilgisayardır.

#### Programlanabilir Kontrol Cihazının yapısı

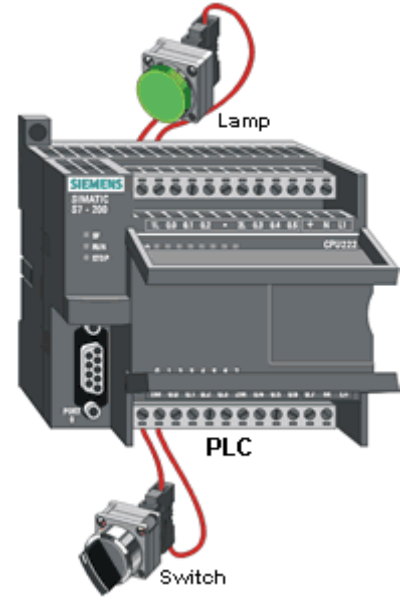
**CPU veya işlemci** : CPU, veya işlemci, kontrol cihazının beynini oluşturur. CPU giriş değerlerini okur, hafızada saklanan verilerle programı yürütür, aktivatörlerin bağlandığı çıkışları yazar

**Hafıza** : Kontrol cihazı hafızası yürütülecek programı ve aynı zamanda bu program tarafından kullanılan verileri de (zamanlayıcı değerleri, sayıcı değerleri, vb) tutar burası kontrol cihazının içerdiği bütün verilerin saklandığı yerdir. Hafıza olmadan, kontrol cihazı çalışmaz.



**Giriş/Çıkış Arabirimi** Giriş/Çıkış arabirimi, giriş elemanlarının durumlarını kontrol cihazına iletmek ve program sonucunda oluşan çıkış değerlerini aktivatörlere iletmek için kullanılır.

Giriş ve çıkışlar kontrol cihazlarının dış dünya ile fiziksel bağlantılarını oluştururlar. Bu sizin için basit bağlantı terminallerinden ibarettir, ancak kontrol cihazı içinde bir sistemi temsil ederler. Girişe gelen bir elektriksel sinyal, lojik duruma (0 veya 1) dönüştürülür Çıkıştaki bir lojik durum ise (0 veya 1) elektriksel sinyale dönüştürülür



**Güç kaynağı** Kontrol cihazının çalışması için gerekli dahili gerilimler üretir.

### Programlanabilir Kontrol Cihazı nasıl çalışır?

#### Kontrol döngüsü nedir?

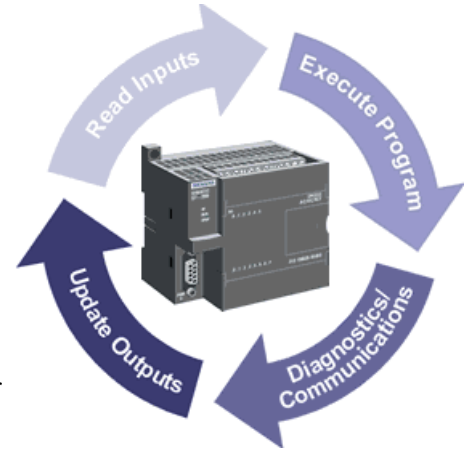
Programın çalışmasını başlattığımız anda, kontrol cihazı döngüsel olarak 3 fazı gerçekleştirecektir:

**Faz 1:** Girişlerinin o anki değerlerinin okunması

**Faz 2:** Programın yürütülmesi

**Faz 3:** Çıkışın aktif ya da deaktif edilmesi

Faz 3'ün sonunda kontrol cihazı Faz 1'e döner ve bu döngü bu şekilde devam eder.



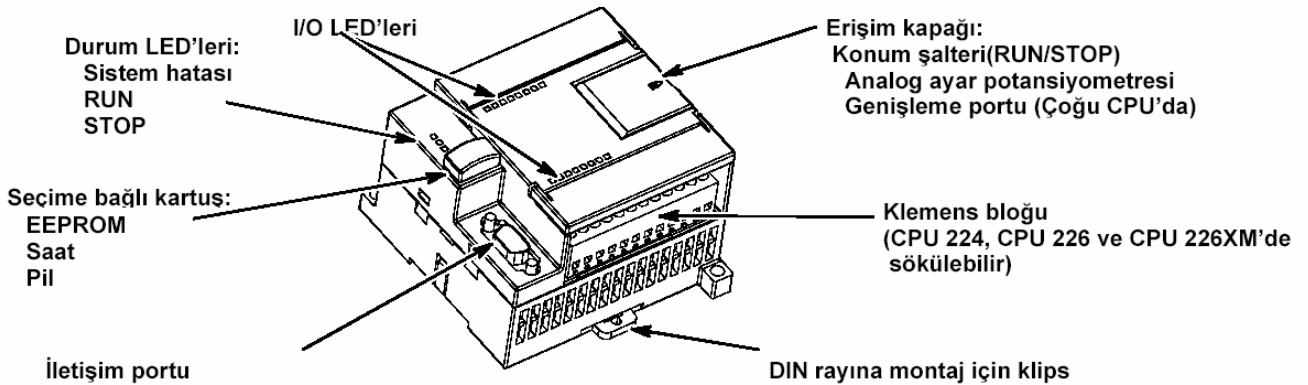
#### Program neye denir?

Program, gerçekleştirilmesi gereken operasyonları işlemciye ileten bir dizi sıralı komutlar bütünüdür.

#### Program dili nedir?

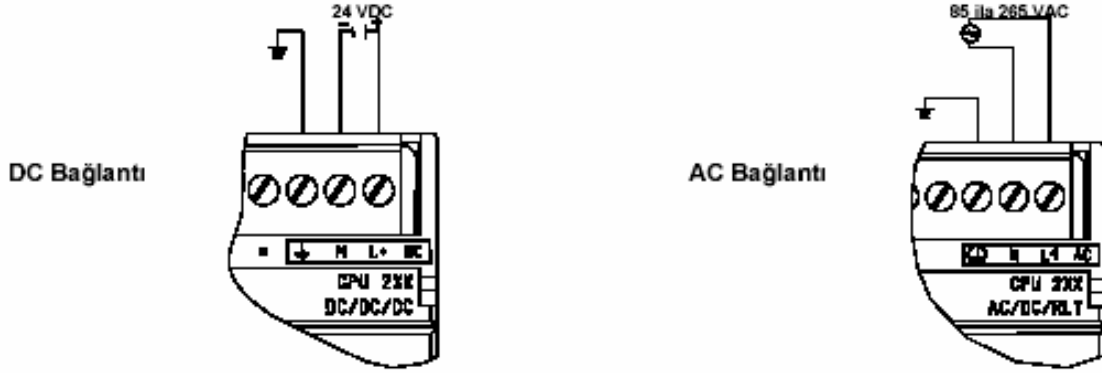
Program işlemci tarafından anlaşılabilir bir dilde yazılmalıdır. Buna programlama dili diyoruz. Gerçekleştirilecek işleme uygun olarak bir çok programlama dili olabilir

#### Siemens S7-200 PLC



#### S7-200 CPU Enerji Bağlantıları

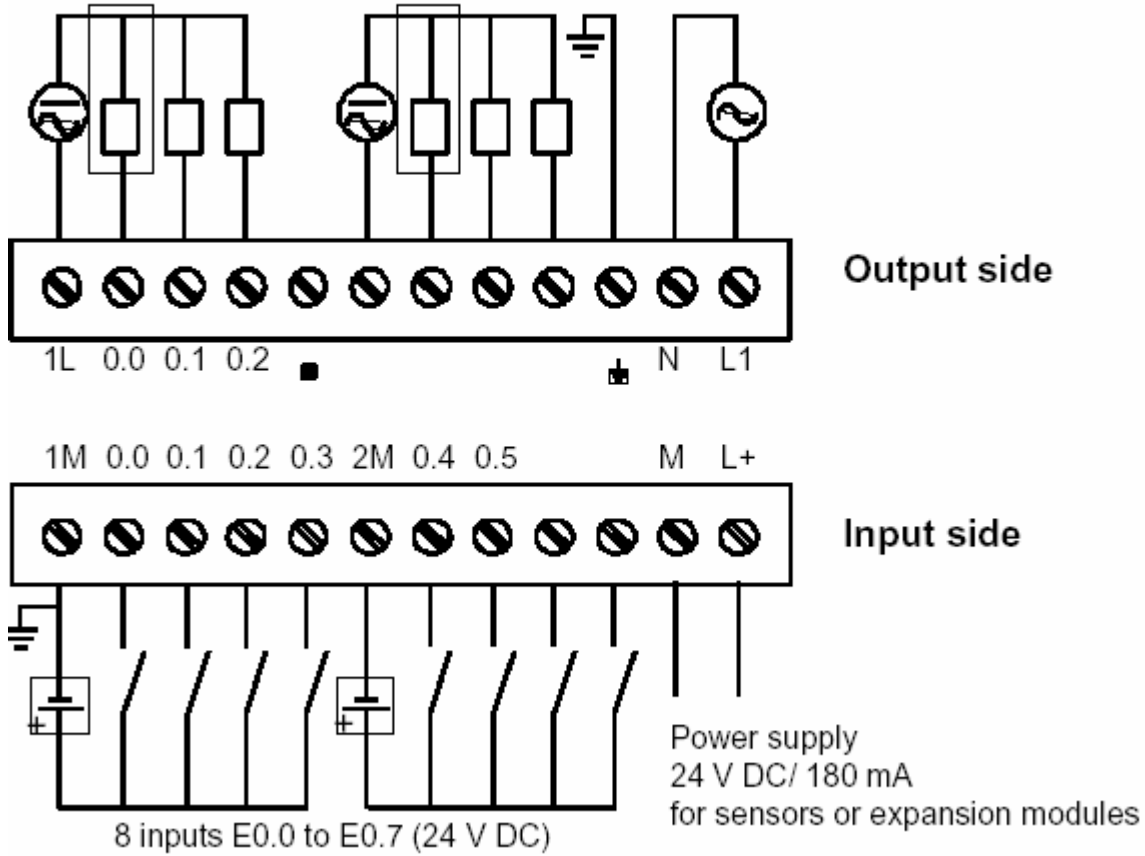
İlk adımda S7-200'ü bir güç kaynağına (veya şebekeye) bağlayacağız. Şekilde S7-200 CPU'nun DC ve AC modelleri için enerji bağlantıları gösterilmiştir. Herhangi bir elektrikli cihazı söker veya yerine takarken enerji bağlantısının kapalı olduğundan emin olun. Bu nedenle S7-200 cihazını da söker veya yerine takarken gerekli emniyet koşullarına uyum ve enerjinin bağlı olmadığından emin olunuz.

**Uyarı**

S7-200 veya diğer ilgili ekipmanı monte ederken veya kablağını yaparken enerji varsa, elektrik çarpması veya ekipmanın hatalı çalışması sonucu doğabilir. Sökme ve yerine takma esnasında S7-200 ve diğer ekipmanda enerji bulunması ölüme, ciddi yaralanmaya ve/veya ekipmana zarar gelmesiyle sonuçlanabilir. S7-200 cihazını söker veya yerine takarken gerekli emniyet koşullarına uyun ve enerjinin bağlı olmadığından kesinlikle emin olun.

6 relay outputs A0.0 to A0.5  
(24 V DC/ 24 to 230 V AC max. 2 A)

Power supply  
(85 to 264 V AC)

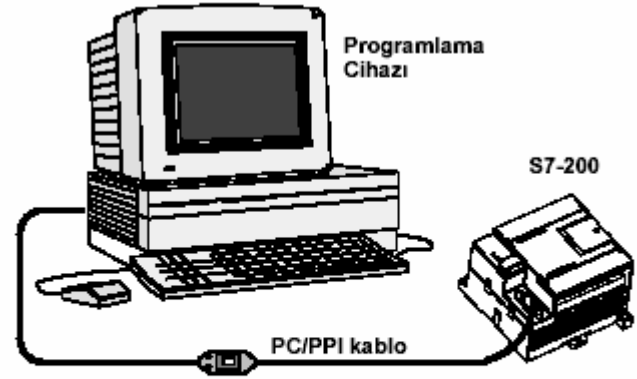




### PC/PPI Kablosunu Bağlama

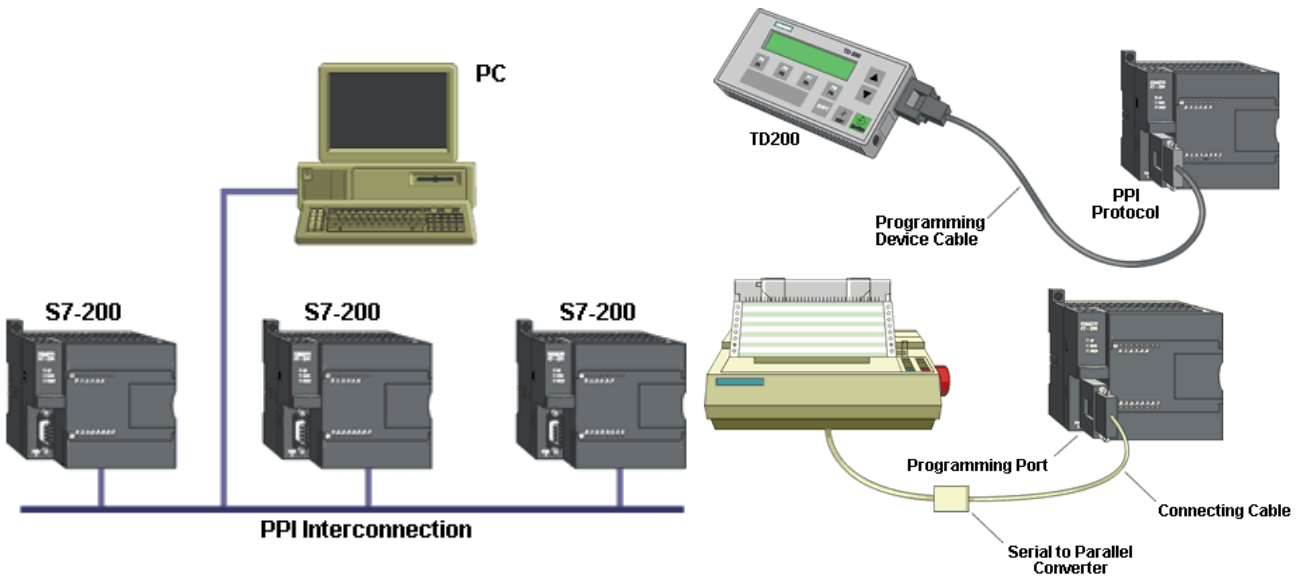
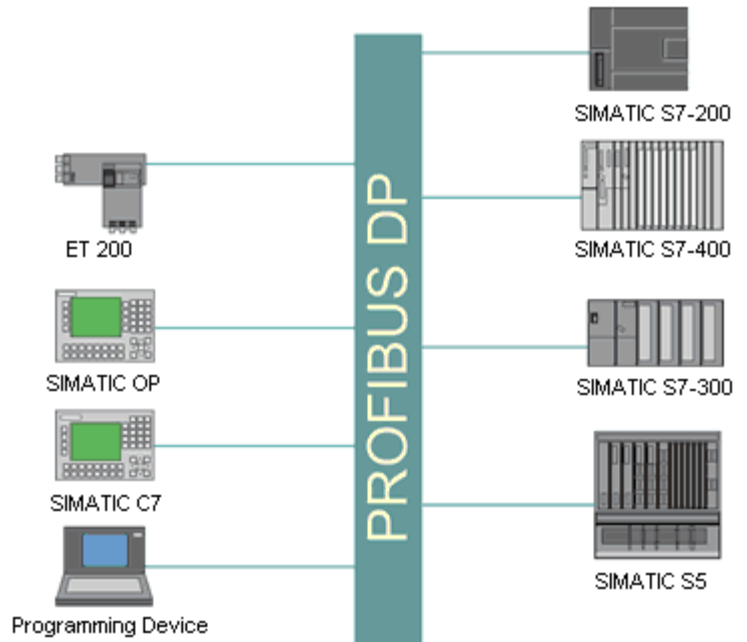
Şekilde PC/PPI kablosuyla S7-200 ve programlama cihazı(PC) arasındaki bağlantı görülmektedir. PC/PPI kablosunu bağlamak için:

1. RS-232 konvektörünü ( üzerinde "PC" yazar) programlama cihazının seri portuna (örneğimizde COM1'e) bağlayın.
2. RS-485 konvektörünü (üzerinde "PPI" yazar) S7-200'ün Port 0 veya Port 1'ine bağlayın.
3. PC/PPI kablosu üzerindeki DIP swiçlerin şekilde gösterildiği gibi olduğundan emin olun.



### İletişim Seçenekleri

Siemens, bilgisayarınızı S7-200'e bağlamak için iki ayrı iletişim seçeneği sunmaktadır: PC/PPI kablosu ile veya MPI ve PROFIBUS-DP iletişim ağlarında da kullanılabilecek bir Communication Processor (CP) kartı ve MPI kablosu ile. PC/PPI programlama kablosu bilgisayarınızı S7-200'e bağlamak için en yaygın olarak kullanılan ve en ekonomik seçenektir. Bu kablo, bir taraftan S7-200 portuna, diğer taraftan bilgisayarınızın seri (COM) portuna bağlanır. PC/PPI kablosu, sadece programlama amaçlı değil, S7-200'ün diğer cihazlara (örneğin modem) bağlantısı için bir çevirici olarak da kullanılabilir. MPI kablosunu kullanmak için, bilgisayarınıza bir CP kartı taktığınız. Bu CP kartı daha yüksek iletişim hızlarında bağlantı için gereken donanımı içerir ve yüksek hızda şebeke bağlantısına olanak tanır.

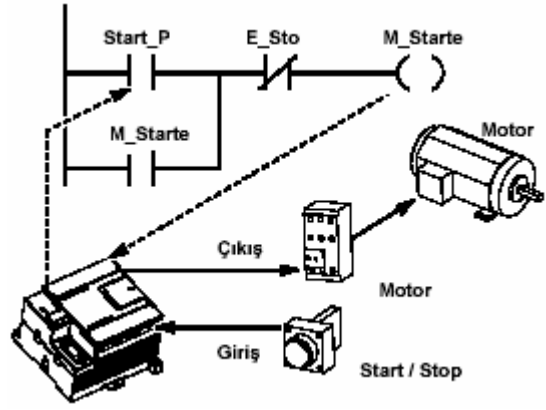


### S7-200 Kumanda Programınızı Nasıl Çalıştırıyor

S7-200 sürekli olarak kumanda programını tarar; verileri okur ve yazar.

#### S7-200, Programınızı Fiziksel Giriş ve Çıkışlarla İlişkilendirir

S7-200'ün temel çalışma şekli oldukça basittir: S7-200 girişlerin sinyal seviyesini okur. S7-200'deki program, değerlendirirken bu değerleri kullanır. Program çalıştıkça S7-200, verileri (sonuçları) günceller ve S7-200 verileri (sonuçları) çıkışlara yazar. Şekilde bir elektriksel devre şemasının nasıl S7-200 ile ilişkilendirildiğini göstermektedir. Bu örnekte, motoru çalıştıran anahtarın değeri diğer girişlerin durumuyla bir araya getirilmiştir. Bu durumların hesaplanması sonucunda motoru çalıştıran aktüatöre bilgi gönderilir.



#### S7-200, İşlemleri bir Tarama Döngüsünde Gerçekleştirir

S7-200 bir dizi işlemi sürekli olarak yapar. Bu işlemlerin döngüsel işletilmesine Tarama adı verilir. Şekilde de görüldüğü gibi, S7-200 görevlerinin tamamını veya çoğunu bir döngü içinde gerçekleştirir:

**Girişlerin okunması:** S7-200 fiziksel girişlerin durumunu PII kütüğü (Proses Giriş İmge Kütüğü) denilen alana kopyalar.

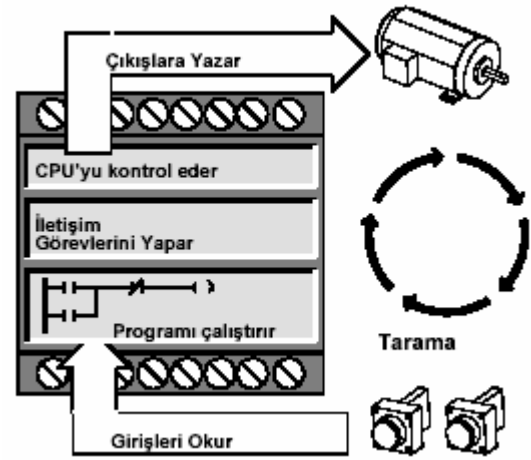
**Programdaki kumanda mantığının çalıştırılması:** S7-200 programdaki komutları çalıştırır ve sonuçları değişik hafıza alanlarına yazar. Bu sonuçlar fiziksel çıkışlarla ilgiliyse PIQ kütüğü (Proses Çıkış İmge Kütüğü) denilen alana yazılır

**İletişim taleplerinin yerine getirilmesi:** Bir PC veya operatör paneli ile iletişim gibi görevler yerine getirilir.

**Kendi kendinin kontrol edilmesi:** S7-200, işletim sisteminin, program hafızasının ve genişleme modüllerinin sağlam olup olmadığını kontrol eder.

**Çıkışların yazılması:** PIQ' da saklanan sonuçlar fiziksel çıkışlara aktarılır.

Taramanın işletilmesi S7-200'ün STOP veya RUN konumunda olmasına bağlıdır. RUN konumunda programınız çalıştırılır; STOP konumunda çalıştırılmaz.

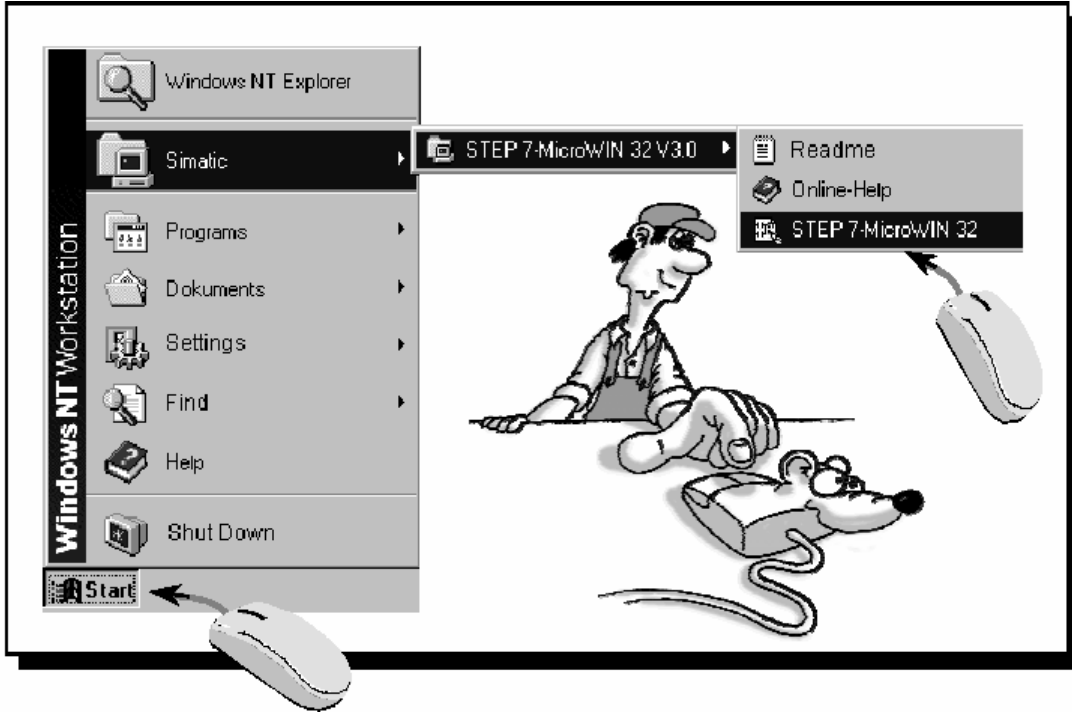
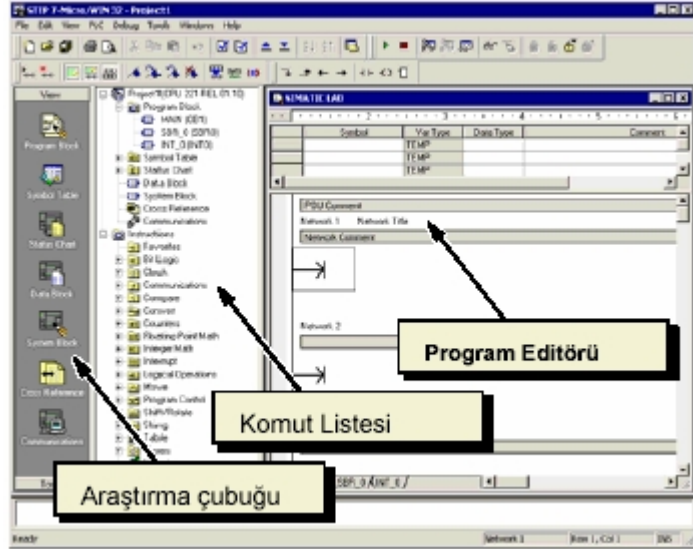


### STEP 7-Micro/WIN'in Program Oluşturmak için Kullanılması

STEP 7-Micro/WIN'i açmak için, STEP 7-Micro/WIN simgesine tıklanın veya **Start > SIMATIC >STEP 7 MicroWIN 3.2** menü komutunu seçin. Şekilde görüleceği gibi, STEP 7-Micro/WIN proje penceresi, programınızı oluşturmak için uygun bir çalışma alanı sağlar. Araç çubuğunda sıklıkla kullanılan menü komutları için kısayol butonları yer almaktadır. Araç çubuklarından istediğinizi gizleyebilir veya görüntüleyebilirsiniz.

Araştırma çubuğu, STEP 7-Micro/WIN'in değişik programlama olanaklarına erişim için simgeler içerir. Komut listesi, kumanda programınızı oluşturmak için gereken tüm proje bileşenlerini ve komutları gösterir. Seçtiğiniz komutu buradan programınıza sürükleyip bırakabilirsiniz. Ayrıca herhangi bir komutun üzerine çift tıklattığınızda imlecin bulunduğu yere o komut yerleştirilir. Program editörü program lojiğini ve lokal değişkenler tablosunu kapsar. Bu tabloda geçici lokal değişkenler için sembolik isimler tanımlayabilirsiniz. Altprogramlar, program editörü penceresinin alt kısmında bölmeler halinde görülür. Bu alanlara tıklayarak ana program ve altprogramlar arasında geçim yapabilirsiniz.

STEP 7-Micro/WIN, programınızı oluşturmak için üç ayrı editör sağlar: Ladder Logic (LAD), Komut Listesi (STL) ve Function Block Diyagram (FBD). Bir kaç sınırlama dışında herhangi bir editörle yazılan program, diğer editörlerle de görüntülenebilir ve üzerinde değişiklik yapılabilir.

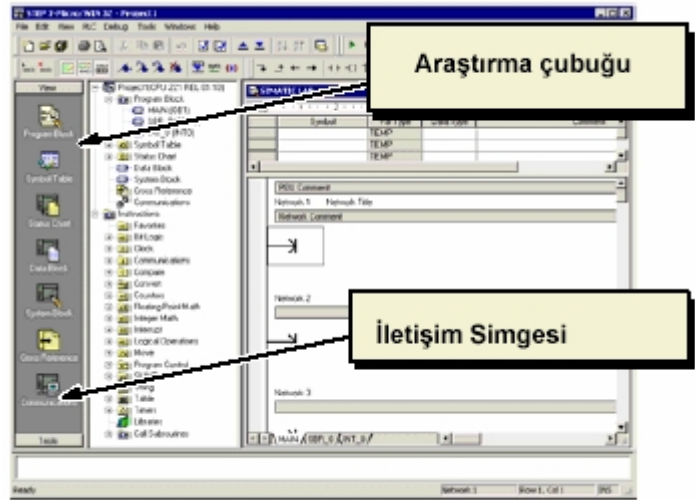




**STEP 7-Micro/WIN'i Açmak**

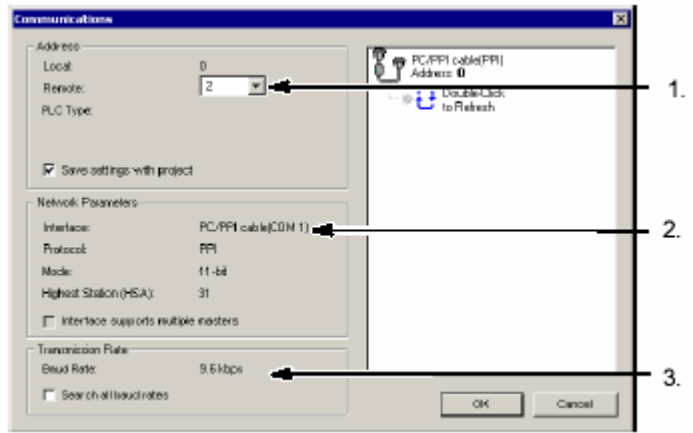
Yeni bir proje açmak için STEP 7-Micro/WIN simgesine tıklayınız. Resimde yeni bir proje gösterilmektedir.

Araştırma çubuğuna dikkat edin. Buradaki simgeleri kullanarak STEP 7-Micro/WIN projesi içerisindeki değişik elemanları açabilirsiniz. İletişim diyalog kutusunu açmak için araştırma çubuğu içinde yer alan İletişim Simgesini tıklayın. Bu diyalog kutusu ile STEP 7-Micro/WIN iletişim ayarları yapılır.

**STEP 7-Micro/WIN İletişim Ayarlarını Kontrol Etmek**

Örnek projemiz STEP 7-Micro/WIN ve PC/PPI kablo için standart ayarları kullanmaktadır. Bu ayarların doğruluğunu kontrol etmek için:

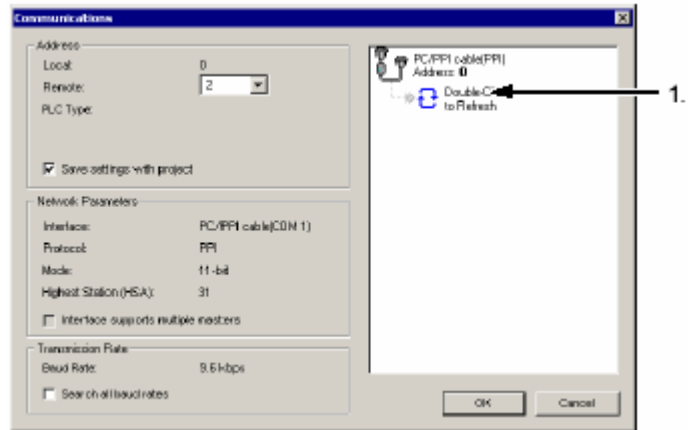
1. İletişim diyalog kutusunda PC/PPI kablosu için girilen adres ayarının 0 olduğunu doğrulayın.
2. Şebeke parametresinin PC/PPI kablo (COM1)'e ayarlı olduğunu doğrulayın.
3. İletişim hızının 9.6 kbps'e ayarlı olduğunu doğrulayın.

**S7-200 ile İletişim Kurmak**

S7-200 CPU ile iletişim kurmak için iletişim diyalog kutusunu kullanın:

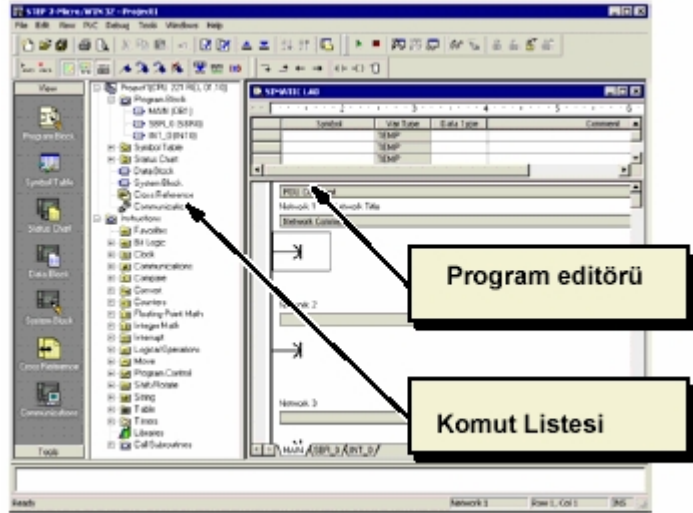
1. İletişim diyalog kutusundaki Yenile (Refresh) simgesini çift tıklayın. STEP 7-Micro/WIN, bağlı S7-200 cihazını araştırarak ve bulunduğu zaman bir CPU simgesi gösterecektir.
2. S7-200'ü seçin ve OK'ı tıklayın. Eğer STEP 7-Micro/WIN S7-200 CPU'yu bulamazsa, iletişim ayarlarını kontrol edin ve bu adımları tekrarlayın.

S7-200 ile iletişim kurduktan sonra, örnek programı oluşturmak ve yüklemek için hazırlıklarımız tamamlanmıştır.



### Program Editörünü Açmak

Program editörünü açmak için Program Blok simgesini tıklatın. Komut listesi ve program editörüne dikkat edin. Komut listesinden seçtiğimiz LAD komutlarını program editörüne “sürükle ve bırak” tekniğiyle yerleştirmektediriz. Araç çubuğu simgeleri menü komutlarına kısa yol erişimi sağlar. Programınızı girip kaydettikten sonra S7-200’e yükleyebilirsiniz.



\* Aşağıda adım adım verilen program sadece bir programın nasıl yapıldığını anlatmak için verilmiştir burada kullanılan elemanların görevlerinin ne olduğu ve programda nasıl kullanıldığı önümüzdeki derslerde anlatılacaktır.

### Örnek bir devreyi yazıp çalıştırmak

#### Zaman Rölesini Çalıştırmak

M0.0 yokken (0), bu kontak enerji akışına izin verir ve zaman rölesini çalıştırır. M0.0 için kontak girişi şöyle yapılacaktır:

1. Bit Logic simgesini çift tıklatın veya (+) işaretini tıklatın. Böylece karşınıza bit mantığı komutları gelecektir.
2. Normalde açık kontağı seçin. 3. Şimdi farenin sol butonuna basılı tutarak kontağı birinci devreye taşıyın ve bırakın.
4. Kontakın üzerinde görülen “???” alanına şu adresi yazın: M0.0
5. Enter tuşuna basarak girişi tamamlayın.

T33 için zamanlama komutu da şöyle Girilecektir

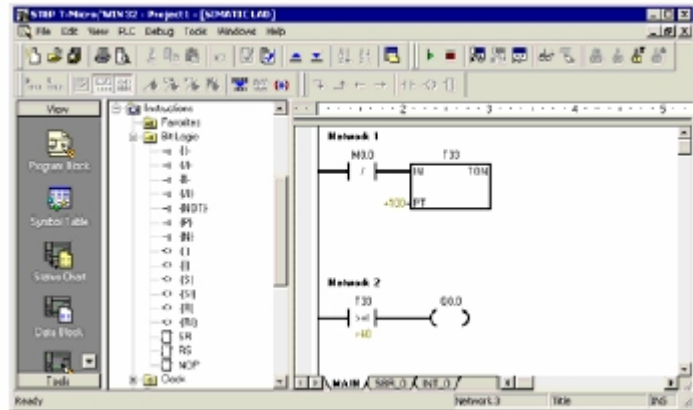
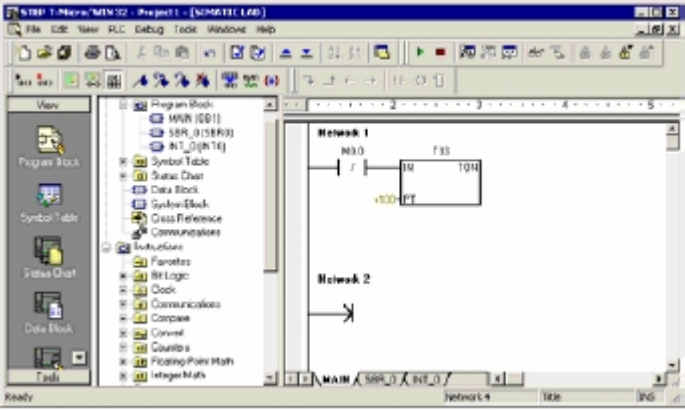
1. Zaman rölesi komutları görmek için Timers simgesini tıklatın.
2. TON'u (On-Delay Timer=Çekmede gecikmeli zaman rölesi) seçin.
3. Sol fare tuşunu basılı tutarak zaman rölesini birinci devreye taşıyın ve bırakın.
4. Zaman rölesi kutusunun üstündeki “???” alanına tıklatın ve rölenin numarasını girin: T33 (Burada T33 örnek olarak verilmiştir)
5. Enter tuşuna basınca imleç, zaman ayarı (PT) alanına gelecektir.
6. Zaman ayarı için şu değeri girin: 100
7. Enter tuşuyla değeri onaylayın.

T33'ün zaman değeri 40'a (40 çarpı 10 milisaniye, veya 0.4 saniye) eşit veya büyük olunca, kontak S7-200'ün Q0.0 çıkışını çalıştırmak üzere enerji akışı sağlar. Karşılaştırma komutunu girmek için:

1. Karşılaştırma simgesini çift tıklayın, böylece bu alandaki tüm komutlar listelenecektir. >=1 komutunu seçin (Büyük veya Eşit Tamsayı).
2. Sol fare tuşunu basılı tutarak karşılaştırma komutunu ikinci devreye taşıyın ve bırakın.
3. Kontakın üzerindeki “???” alanına tıklatın ve zaman rölesinin adresini girin: T33
4. Enter tuşuna basınca zaman rölesi numarası girilmiş olacak ve imleç ikinci karşılaştırma alanına gelecektir.
5. Zaman rölesi değeriyle karşılaştırılacak sabit değeri girin: 40
6. Enter tuşuyla değeri onaylayın.

Q0.0 çıkışını çalıştıracak komutu girmek için:

1. Bit Logic simgesine tıklayarak bu alandaki tüm komutların görülmesini sağlayın ve çıkış bobinini seçin.



2. Sol fare tuşuna basılı tutarak bobini ikinci devreye taşıyın ve bırakın.
3. Bobinin üzerindeki "???" alanına tıklatın ve şu adresi girin: Q0.0
4. Enter tuşuyla değeri onaylayın.

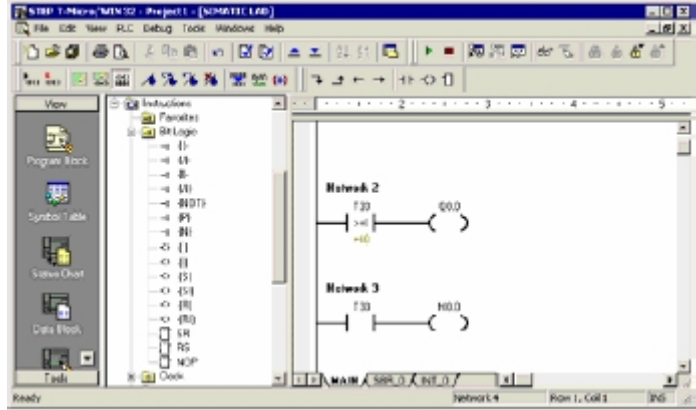
Zaman rölesi ayarlanan değere (100) ulaşmış zaman rölesi biti 1 olunca T33 kontağı da 1 olur. Bu kontakta M0.0 hafıza alanına (yardımcı rölesine) enerji akışı olur. Zaman rölesi M0.0'in normalde kapalı kontağıyla çalıştırıldığı için, M0.0'un 0'dan 1'e dönmesi zaman rölesini resetler.

Zaman rölesi T33'ün kontağı şu şekilde girilir:

1. Bit Logic komutları kısmından Normalde Açık kontağı seçin.
2. Sol fare butonuna basılı tutarak kontağı üçüncü devreye taşıyın ve bırakın.
3. Kontakın üzerindeki "???" alanına tıklatın ve zaman rölesi adresini girin: T33
4. Enter tuşuyla değeri onaylayın.

M0.0'i çalıştıracak bobini yerleştirmek için:

1. Bit logic komutları arasından çıkış bobinini seçin.
2. Sol fare butonuna basılı tutun ve çıkış bobinini üçüncü devreye taşıyıp bırakın.
3. Bobinin üzerindeki "???" alanına çift tıklatın ve şu adresi girin: M0.0
4. Enter tuşuyla girişi onaylayın.



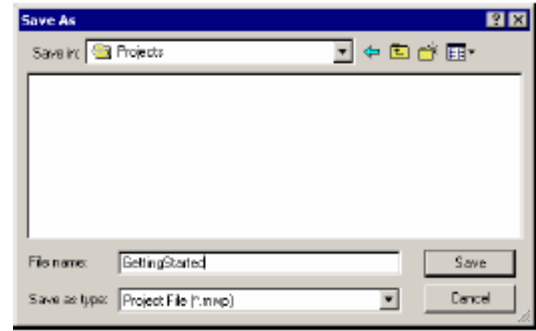
### Örnek Projeyi Kaydetmek

Üç devreyi girerek programımızı tamamlamış olduk. Programı kaydettiğinizde S7-200 CPU modelini ve diğer parametreleri içeren bir proje oluşturmuş olursunuz.

Projeyi kaydetmek için:

1. Menü çubuğundan **File > Save As** menü komutunu seçin.
2. Save As diyalog kutusuna bir proje ismi girin.
3. OK ile projeyi kaydedin.

Projeyi kaydettikten sonra programı S7-200'e yükleyebilirsiniz.

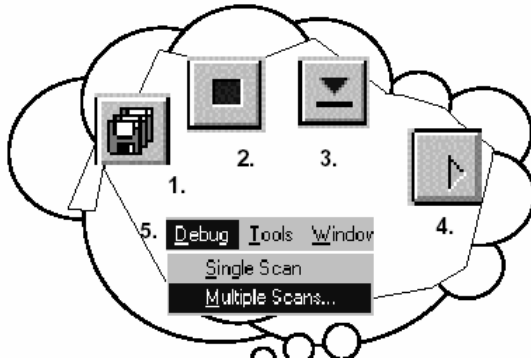
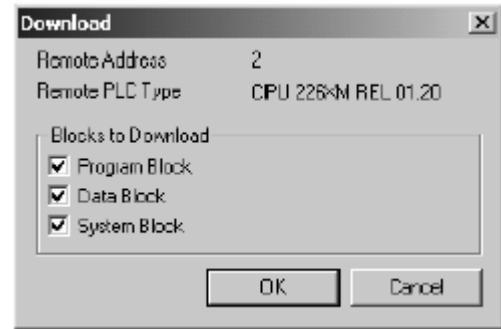


### Örnek Programı Yükleme



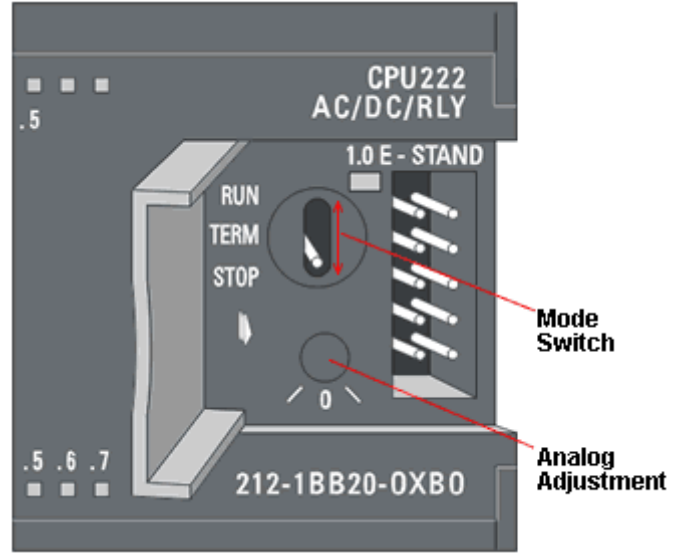
1. Araç çubuğundaki yükleme simgesini seçin veya **File > Download** menü komutunu seçerek yüklemeyi başlatın.

2. Program elemanlarını S7-200'e yüklemek için OK tuşuna basın. Eğer S7-200 RUN konumundaysa, bir diyalog kutusu görünecek ve S7-200'ü STOP konumuna geçirip geçirmemeyi soracaktır. Bu diyalog kutusunu onaylayın.



**S7-200 CPU'nun Çalışma Konumunu Seçmek**

S7-200'ün iki işlem durumu vardır: STOP ve RUN konumları. CPU'nun ön tarafındaki durum LED'leri mevcut çalışma konumunu gösterir ( Yeşil led Run, turuncu led stop konumunu ). STOP konumunda, S7-200 programı işletmez ve CPU'ya konfigürasyonu ve programı yükleyebilirsiniz. RUN konumunda, S7-200 programı çalıştırmaktadır. S7-200'ün üzerinde çalışma konumunu değiştirmek için bir siwiç yer alır. S7-200 erişim kapağının altında yer alan swiçi kullanarak çalışma konumunu manuel olarak değiştirebilirsiniz. Konum swiçini STOP pozisyonuna almak programın çalışmasını durdurur; RUN konumuna almak programın çalışmasını başlatır; TERM (terminal) konumuna almak mevcut çalışma durumunu değiştirmez. Bu konumda plc'nin RUN veya STOP konumu programlama yaptığımız pc tarafından online değiştirilebilir. Bizim çalışmalarımızda konum swiçi TERM pozisyonunda kalacaktır



. Konum swiçi STOP veya TERM pozisyonundayken enerji gidip geri gelirse S7-200 otomatik olarak STOP konumuna geçer. Eğer aynı olay swiç RUN pozisyonundayken olursa, S7-200, RUN konumuna geçer. Yazılımın çalışma konumunu değiştirmesine imkan vermek için konum swicinin TERM veya RUN pozisyonuna alınması gerekir. **PLC > STOP** veya **PLC > RUN** menü komutları veya araç çubuğundaki ilgili butonları kullanarak çalışma konumunu değiştirebilirsiniz. S7-200'ü STOP konumuna getirmek için programınızın içerisinde STOP komutunu da kullanabilirsiniz. Bu komut, program mantığına uygun olarak programın akışının durdurulmasını sağlar.

**S7-200'ü RUN Konumuna Almak**

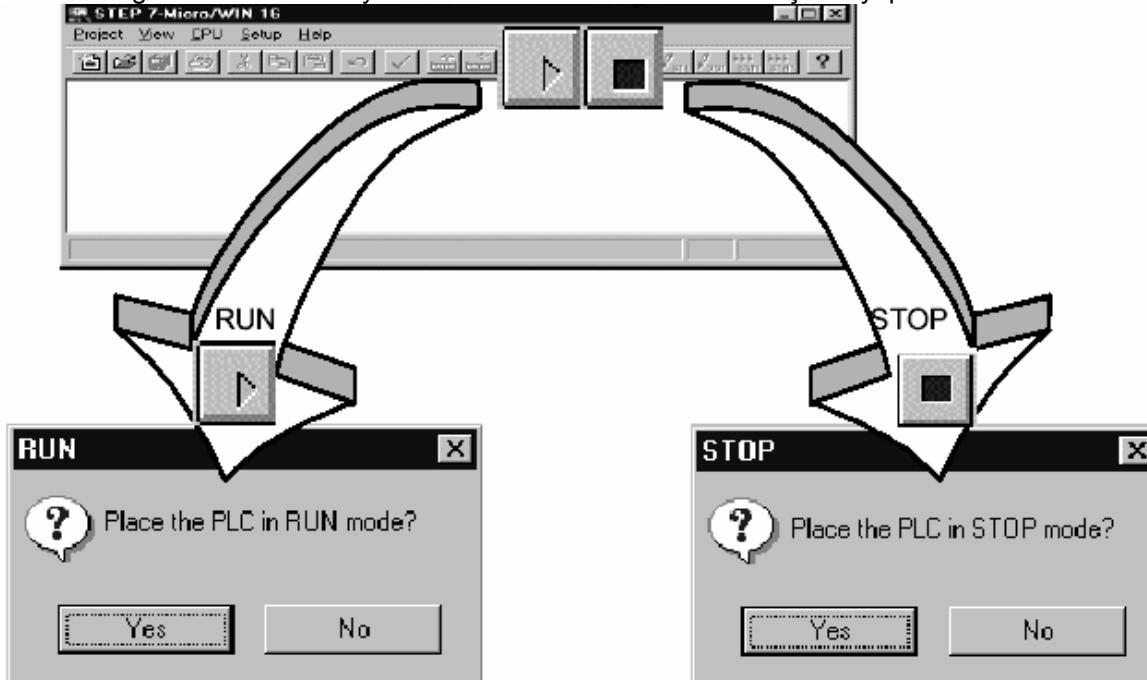
STEP 7-Micro/WIN'ün S7-200 CPU'yu RUN konumuna alması için cihazın üzerindeki konum şalterinin TERM veya RUN konumunda olması gereklidir. S7-200'ü RUN konumuna alınca S7-200 programı işletir:



1. RUN simgesini tıklatın veya **PLC > RUN** menü komutunu seçin.
2. S7-200'ün çalışma konumunu değiştirmek için OK'i tıklatın.

S7-200 RUN konumuna geçince CPU programı işletmeye başladığından programımıza bağlı olarak Q0.0 LED'i yanıp sönecektir.

Programın çalışmasını **Debug > Program Status** menü komutu sayesinde izleyebilirsiniz. STEP 7-Micro/WIN, komutların değerleri gösterir. Programı durdurmak için S7-200'ü STOP konumuna alınız. Bunu, STOP simgesine tıklatarak veya **PLC > STOP** mönü komutunu seçerek yapabilirsiniz.



## Programlama Dilleri

### STL Editörünün Özellikleri

STL editörü, programın metin olarak girilmesini sağlar. STL editörü, LAD veya FBD ile yazılamayacak bazı özel komutların girilebilmesini de sağlar. Şematik gösterimin oluşması için geçerli bazı kısıtlamalar STL'de söz konusu olmadığından ve S7-200'ün makine koduna en yakın gösterim şekli olduğundan, STL komutları en geniş imkanları sunar. Ancak, örnekten görülebileceği gibi, bu gösterim şeklinin kullanımı elektrik veya elektronik eğitimi almış kişilerden çok bilgisayar teknolojisine yatkın kişilere daha kolay gelmektedir.

S7-200, programda yazılan her satırı yukarıdan aşağıya doğru birer birer işler ve sonra tekrar başa döner. STL, ara sonuçları saklayabilmek için bir lojik 'yığın' kullanır. Böylece ardı ardına yapılan lojik işlemlerin gerçekleştirilmesi mümkün olur. STL editörünü seçerken aşağıdaki konuları dikkate alınız:

STL daha çok uzman programcı ve bilgisayar eğitimi almış kişiler için uygundur. STL bazı özel durumlarda LAD veya FBD editörü ile kolay olmayan çözümlere kolayca ulaşmanızı sağlar. Buna karşın izleme fonksiyonları STL'de daha zordur. STL editörünü sadece SIMATIC komut setiyle birlikte kullanabilirsiniz. LAD veya FBD editörüyle yazılmış programı her zaman STL ile izlemek mümkün olduğu halde bunun tersi doğru değildir. Bazı STL programları LAD veya FBD ile görüntülenemez.

### LAD Editörünün Özellikleri

LAD editörü, programı elektriksel bağlantı resmine çok yakın bir şekilde şematik olarak gösterir. Aslında LAD'de yazılmış program elektrik devre şemasının 90 derece döndürülmüş hali olarak düşünülebilir. Ladder programları tıpkı gerçek elektrik devrelerindeki gibi "bir enerji kaynağından kontaklar vasıtasıyla akan enerjiyi" sembolize etmek şeklinde, kullanıcıya kolay gelebilecek gösterim mantığına sahiptir.

LAD programında sol tarafta gösterilen dikey çizgi enerji kaynağını sembolize eder. 'Kapanmış' olan kontaklar enerji akışına izin verirken 'açılmış' kontaklar bu sembolik akışı bloke ederler.

Program devre (network) olarak tanımlanan parçalara bölünmüştür. Program akışı her devrede soldan sağa ve yukarıdan aşağıya doğrudur. Şekilde örnek bir LAD programını göstermektedir. Değişik komutlar grafik sembollerle gösterilmekte olup üç ayrı grupta incelenebilir.

**Kontaklar;** swiç, buton, şalter veya dahili koşullar gibi lojik girişlere işaret eder.

**Bobinler;** lamba, kontaktör veya dahili çıkış koşulları gibi lojik sonuçlara işaret eder.

**Kutular;** zaman rölesi, sayıcı, matematik fonksiyonlar gibi ek özelliklere işaret eder.

LAD editörünü seçerken şu gerçekleri dikkate alın:

Ladder mantığı daha çok elektrik eğitimi almış kişiler ve yeni başlayanlar için uygundur. Şematik gösterim şeklinin anlaşılması kolaydır ve tüm dünyada popülerdir. LAD editörü hem SIMATIC, hem de IEC 1131-3 komut setleriyle kullanılabilir. LAD editörüyle yazılmış bir program her zaman STL ile görüntülenebilir.

### Örnek Program

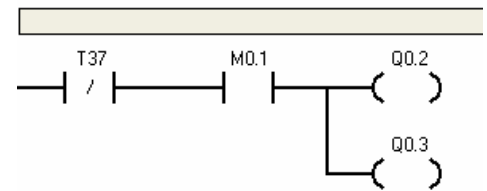
Program Satırları	Açıklamalar
LD I0.0	//Girişi oku
A I0.1	//Diğer girişle AND'le
= Q1.0	//Sonucu çıkışa yaz

I0.0 : 0. baytın 0. biti ile adreslenmiş giriş elemanı  
 Q1.0 : 1. baytın 0. biti ile adreslenmiş çıkış elemanı  
 Program üzerine yapılacak açıklamalar programdan // ile ayrılır.

### Örnek Program



### Network 5 Turn on Mixer Motor.



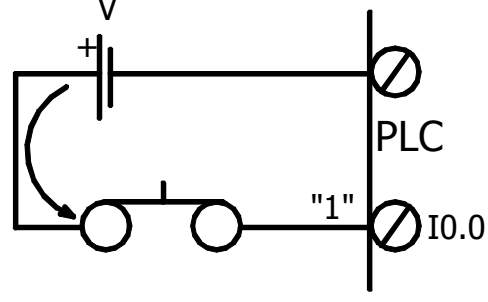
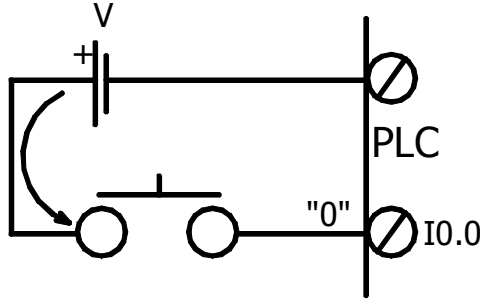






Plc girişine bağlanan açık veya kapalı anahtarın konum değişikliğine göre program içerisinde kullanılan normalde açık ve normalde kapalı kontak konum değişikliği:

**Açık anahtarın bağlanması :**



**Program içerisinde normalde açık kontak kullanımı:**

Buton açıkken plc'nin I0.0 girişinde 0 Volt vardır yani lojik 0 dır buna bağlı olarak program içerisindeki I0.0 a ait normalde açık

kontak açık konumunu koruyacak ve enerji geçişine müsaade etmeyecektir.

Butona basıldığında plc'nin I0.0 girişine buton üzerinden V gerilimi uygulanacak ve lojik 1 olacaktır. Buna bağlı olarak program içerisindeki I0.0 a ait normalde açık kontak kapanarak akım geçişine izin verecektir.

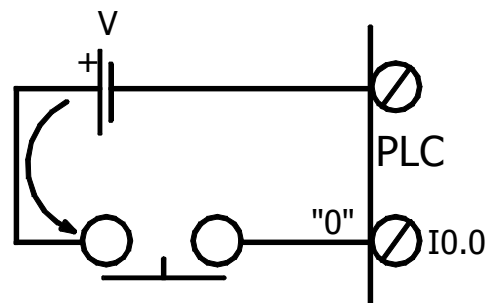
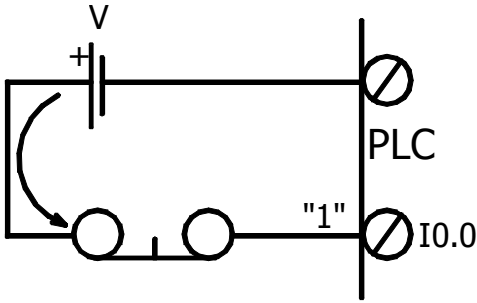
**Program içerisinde normalde kapalı kontak kullanımı:**

Buton açıkken plc'nin I0.0 girişinde 0 Volt vardır yani lojik 0 dır buna bağlı olarak program içerisindeki I0.0 a ait normalde

kapalı kontak kapalı konumunu koruyacak ve enerji geçişine izin verecektir.

Butona basıldığında plc'nin I0.0 girişine buton üzerinden V gerilimi uygulanacak ve lojik 1 olacaktır. Buna bağlı olarak program içerisindeki I0.0 a ait normalde kapalı kontak açılarak akım geçişine müsaade etmeyecektir.

**Kapalı anahtarın bağlanması :**



**Program içerisinde normalde açık kontak kullanımı:**

Buton kapalı olduğundan plc'nin I0.0 girişinde buton üzerinden V gerilimi uygulanacak yani lojik 1 olacaktır buna bağlı

olarak program içerisindeki I0.0 a ait normalde açık kontak konumunu değiştirecek ve kapanacaktır böylece enerji geçişine izin verecektir.

Butona basıldığında buton açılarak plc'nin I0.0 girişindeki gerilimin 0 Volta düşmesine neden olacaktır bu ise plc girişinin lojik 0 olması anlamına gelecektir. Buna bağlı olarak program içerisindeki I0.0 a ait normalde açık kontak açılarak akım geçişine müsaade etmeyecektir.

**Program içerisinde normalde kapalı kontak kullanımı:**

Buton kapalı olduğundan plc'nin I0.0 girişinde buton üzerinden V gerilimi uygulanacak yani lojik 1 olacaktır buna bağlı

olarak program içerisindeki I0.0 a ait normalde kapalı kontak konumunu değiştirecek ve açılacaktır böylece enerji geçişine müsaade etmeyecektir.

Butona basıldığında buton açılarak plc'nin I0.0 girişindeki gerilimin 0 Volta düşmesine neden olacaktır bu ise plc girişinin lojik 0 olması anlamına gelecektir. Buna bağlı olarak program içerisindeki I0.0 a ait normalde kapalı kontak normal haline dönerek kapanacak ve akım geçişine izin verecektir.

### Anında Kontaktlar

Anında (immediate) kontak, S7-200 tarama süresine bağımlı değildir; güncellemeyi anında yapar.

10.2 Anında Normalde Açık (LDI, AI ve OI) Anında Normalde Açık kontak, eğer fiziksel girişi 1 ise

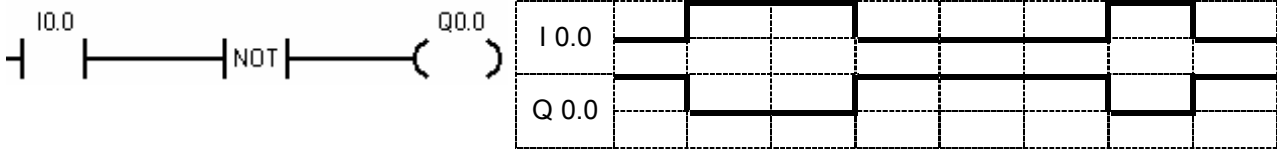
—| |— kapalıdır (akım geçirir)

10.3 Anında Normalde Kapalı kontak komutları (LDNI, ANI ve ONI) Anında Normalde Kapalı kontak eğer  $\neg / \mid$  fiziksel girişi 0 ise kapalıdır (akım geçirir).

\*Giriş kontaklarının isimlendirilmesinde küçük i veya büyük I kullanınız eğer büyük I veya küçük i kullanırsanız program bunu hata olarak algılayacak ve ismi kırmızı olacaktır.

### NOT Komutu

$\neg$  NOT Not (Değilleme) komutu (NOT) enerji akışının yönünü lojik olarak değiller, yani lojik yığının birinci biti 0 ise 1, 1 ise 0 yapar.

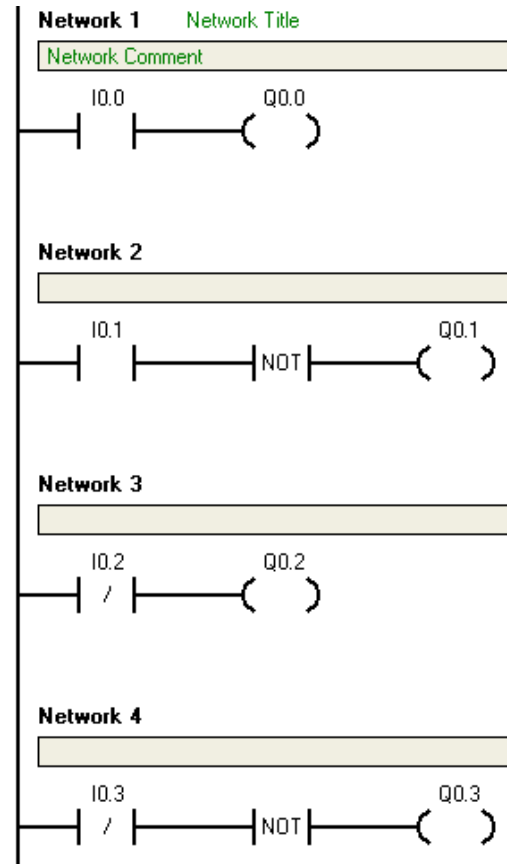


### Örnek Uygulama 1:

Yandaki devreyi kurarak girişlerin her "0" ve "1" konumu için çıkışları gözlemleyiniz

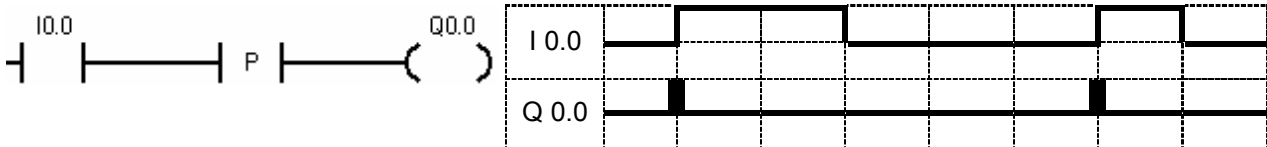
! NOT kontağının önündeki kontağı değillediğine dikkat ediniz.

? Kontakları açık veya kapalı kullanabildiğimize göre sizce NOT kontağına nerelerde ihtiyaç duyabiliriz?

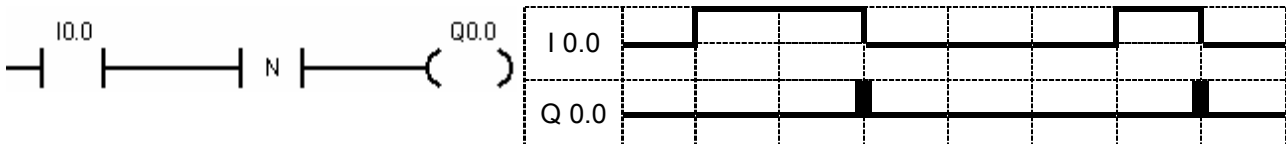


### Pozitif Geçiş (Yükselen Kenar) ve Negatif Geçiş (Düşen Kenar) Komutları

$\neg P$  Pozitif Geçiş (yükselen kenar) komutu (EU), her 0'dan 1'e dönüşümde sadece bir tarama için enerji akışına izin verir.



$\neg N$  Negatif Geçiş (düşen kenar) komutu (ED), her 1'den 0'a dönüşümde sadece bir tarama için enerji akışına izin verir.



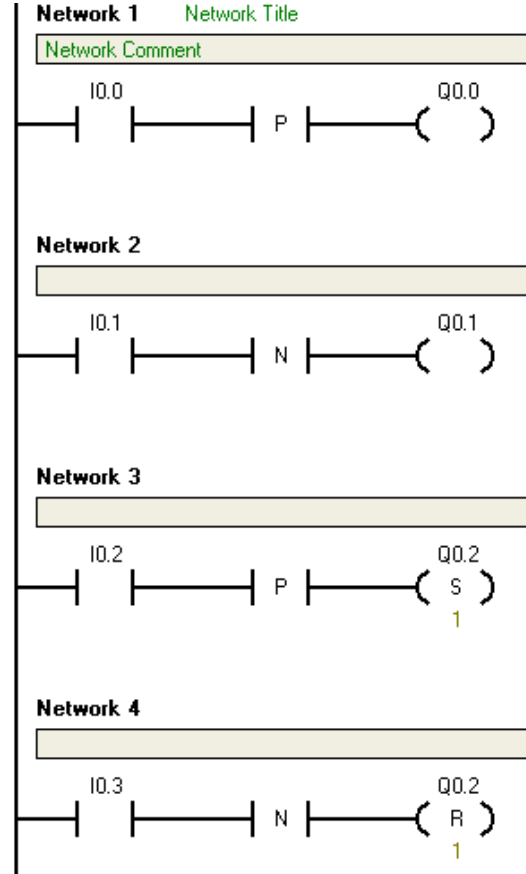
\*P ve N kontakları önümüzdeki derslerde göreceğimiz set ve reset çıkış elemanları ile birlikte kullanılırlar.

### Örnek Uygulama 2:

Yandaki devreyi kurarak girişlerin her "0" ve "1" konumu için çıkışları gözlemleyiniz

? Network1 ve network2 de çıkışları gözlemleyebildiniz mi? Neden?

! Network3 ve network4 de I0.2 ve I0.3 girişlerinin 0 dan 1 e veya 1 den 0 a geçişlerin de çıkışların değiştiğine dikkat ediniz.



### Bobinler

#### Çıkış

$\left( \begin{matrix} Q0.0 \\ S \end{matrix} \right)$  Çıkış komutu (=), çıkışın yeni değerini çıkış kütüğüne yazar. Komut işlendiğinde, S7-200 programa göre çıkış kütüğündeki biti "1" veya "0" yapar. LAD ve FBD'de, tanımlanan bit, enerji akışına eşit kılınır. STL'de lojik yığının tepe biti, belirtilen bite kopyalanır.

#### Anında Çıkış

$\left( \begin{matrix} Q0.1 \\ I \end{matrix} \right)$  Anında Çıkış komutu (=I), yeni değeri taramanın sona ermesine beklemeden hem çıkış kütüğüne, hem de fiziksel çıkışa yazar. Anında çıkış komutu işlendiğinde, fiziksel çıkış noktası, anında enerji akışına eşitlenir. Sonuç da aynı anda hem çıkış kütüğüne, hem de fiziksel çıkışa yazılır. Fiziksel çıkışa o anda yazmaması dolayısıyla normal çıkış komutundan ayrılır.

#### Set ve Reset

$\left( \begin{matrix} Q0.4 \\ S \\ 2 \end{matrix} \right)$  Set (S) ve Reset (R) komutları, belli bir başlangıç adresinden (bit) itibaren belli bir sayıdaki (N) bitin set (1 yapar) ve reset (0 yapar) işlemi gerçekleştirir. N, 1 ila 255 arasında olabilir. Eğer Reset işlemi zaman rölesi (T) veya sayıcı (C) bitine uygulanırsa, hem sayıcı veya zaman rölesi biti, hem de anlık değerler silinir.

$\left( \begin{matrix} Q0.5 \\ R \\ 3 \end{matrix} \right)$

**Bobinlerin ve kontakların kullanımı :** Program içerisinde fiziksel girişlere, bobinlere, sayıcılara ve zamanlayıcılara ait kontaklar birden fazla yerde kullanılabilirler. Bu kontaklar "1" veya "0" olmalarına göre program içerisinde aynı anda açılacak veya kapanacaklardır. Böylelikle kullanıldıkları her bir networkta birbirinden bağımsız işlemleri yerine getirebilirler. Her bir çıkış bobini ise program içerisinde yalnız bir yerde kullanılır. Birden fazla kullanılması durumunda ise program içerisinde diğerlerine göre en son satırda bulunan rölenin aldığı değer çıkışa yazılacaktır.

**Ana ve yardımcı bobinler :** Q??.? Olarak verilen bobinler fiziksel çıkışlar olup Lojik 1 olduklarında PLC nin o çıkışına ait klemensinde enerji olacaktır M??.? olarak verilen bobinler ise PLC içerisindeki sanal yardımcı röleler olup fiziksel çıkış vermezler sadece yazılım içerisinde açık veya kapalı kontakları kullanılır.

**Bobin veya kontakların isimlendirilmesi** : İsimlendirmede harf o elemanın ne olduğunu gösterir.

Örneğin

I : Giriş kontakları

M : Yardımcı röle veya kontağı

C : Sayıcı veya kontağı

Q : Çıkış veya kontağı

T : Zamanlayıcı veya kontağı

Gibi.

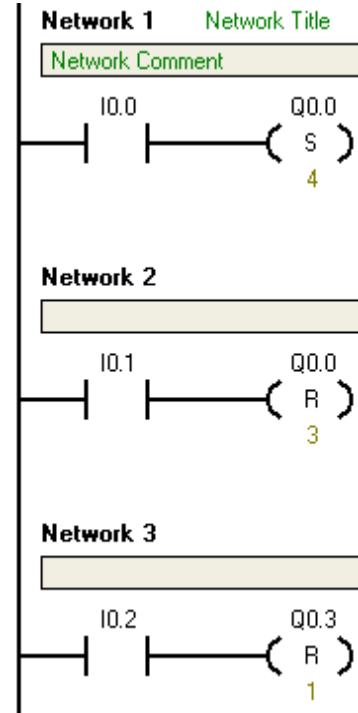
Sayılar ise byte ve bit adreslerini verir. Noktadan önceki sayı byte adresini noktadan sonraki sayı ise bit adresini verir. Bunları örneklerle açıklayacak olursak:

S7 224 plc nin giriş sayısı 14 adettir bunlar I0.0 adresi ile başlar ve I0.7 ye kadar gider 9. giriş ise isimlendirmesine I1.0 ile başlar ve I1.5 e kadar gider. Yine S7 222 plc de 6 adet çıkış vardır ve isimleri Q0.0 ile Q0.5 arasındadır bu plc ye 8 girişli EM223 modülü bağlanacak olursa bu modüldeki girişler ise Q1.0 ila Q1.7 arasında isimlendirilir.

### Örnek Uygulama 3:

Yandaki devreyi kurarak girişlerin her "0" ve "1" konumu için çıkışları gözlemleyiniz

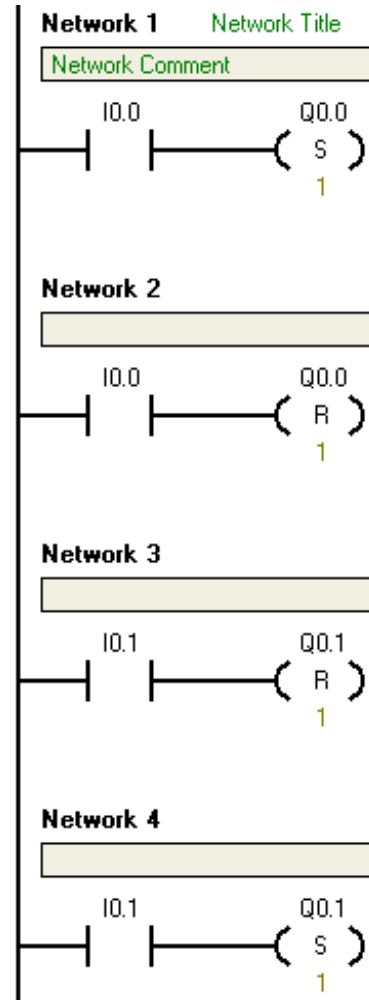
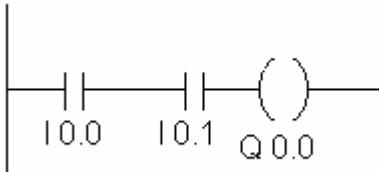
? Çıkışların set ve resetlenmesin de aşağıdaki sayı neyi temsil etmektedir?



**Örnek Uygulama 4:**

Yandaki devreyi kurarak girişlerin her "0" ve "1" konumu için çıkışları gözlemleyiniz

? Network1 ve network2 deki set-reset devresi ile Network3 ve network4 deki set-reset devresi arasındaki fark nedir? Bu çıkışı nasıl etkilemektedir?

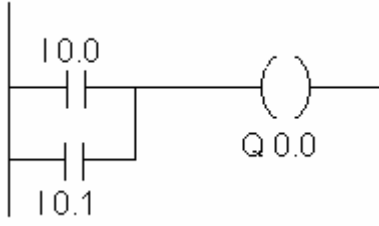
**VE (AND) İŞLEMİ**

Bu örnekte yapılan iş, I 0.0 olarak adlandırılan girişten gelen sinyalin değeri ile I 0.1 girişinden gelen sinyalin değerinin mantıksal VE işleminden geçirilmesidir. Ayrıca normalde açık kontak için seri bağlantı komutudur.

Bu diyagramın STL karşılığı ise:

```
LD I 0.0 //I0.0 Girişini oku
A I 0.1 //ve bu sonucu I0.1 girişi ile A(nd) yani VE işlemine tabi tut
= Q0.0 //And işleminin sonucuna göre Q0.0 çıkışını 1 yap
```

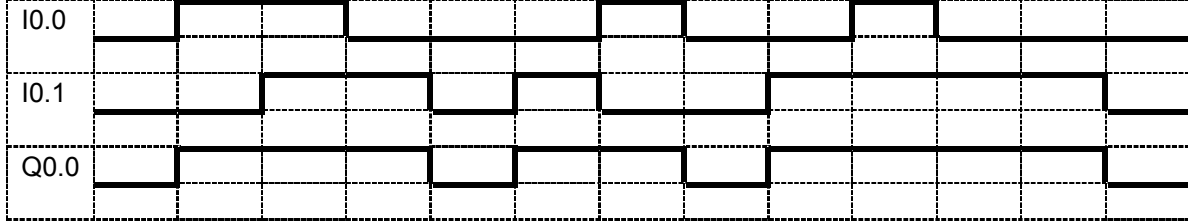
I0.0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
I0.1	0	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
Q0.0	0	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

**VEYA (OR) İŞLEMİ**

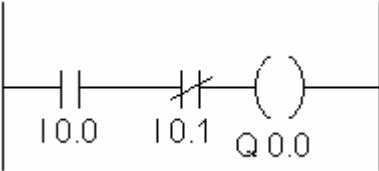
Bu örnekte I0.0 girişi ile I0.1 girişinin mantıksal OR işleminden geçirilmesidir. Normalde açık kontaklar için paralel bağlantı komutudur.

Bu diyagramın STL karşılığı;

```
LD I 0.0 //I0.0 Girişini oku
O I 0.1 //bu sonucu I0.1 girişiyle O(r) yani VEYA işlemine tabi tut
= Q0.0 //Or işleminin sonucuna göre Q0.0 çıkışını 1 yap
```



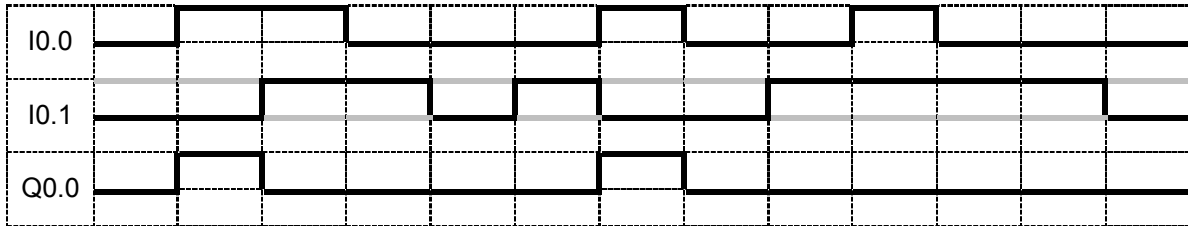
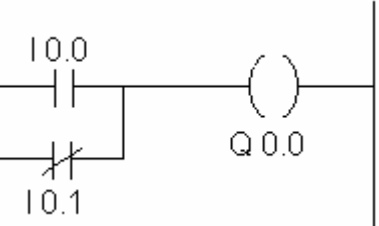
↕ ↕ ↕ ↕ Paralel eki almak için programın üst tarafındaki instruction menüsündeki line butonlarını kullanınız.

**VE DEĞİL (AND NOT) İŞLEMİ**

Normalde kapalı kontaklar için seri bağlantı komutudur

Bu LAD diyagramın STL karşılığı;

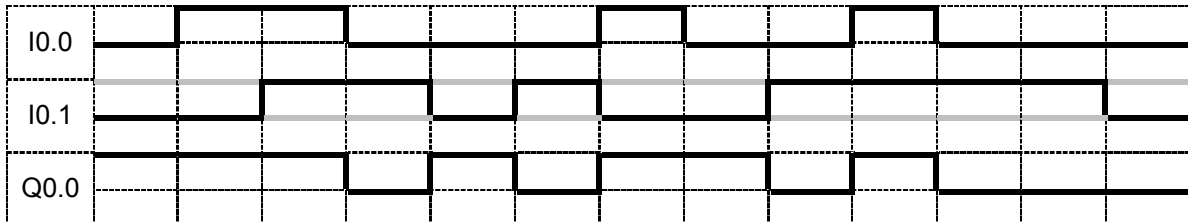
```
LD I 0.0 //I 0.0 Girişini oku
AN I 0.1 //I 0.0 ile I 0.1'i Ve Değil işlemine tabi tut
= Q0.0 //Ve Değil işleminin sonucuna göre Q0.0 çıkışını 1 yap
```

**VEYA DEĞİL (OR NOT) İŞLEMİ**

Normalde kapalı kontaklar için paralel bağlantı komutu.

Bu diyagramın STL karşılığı;

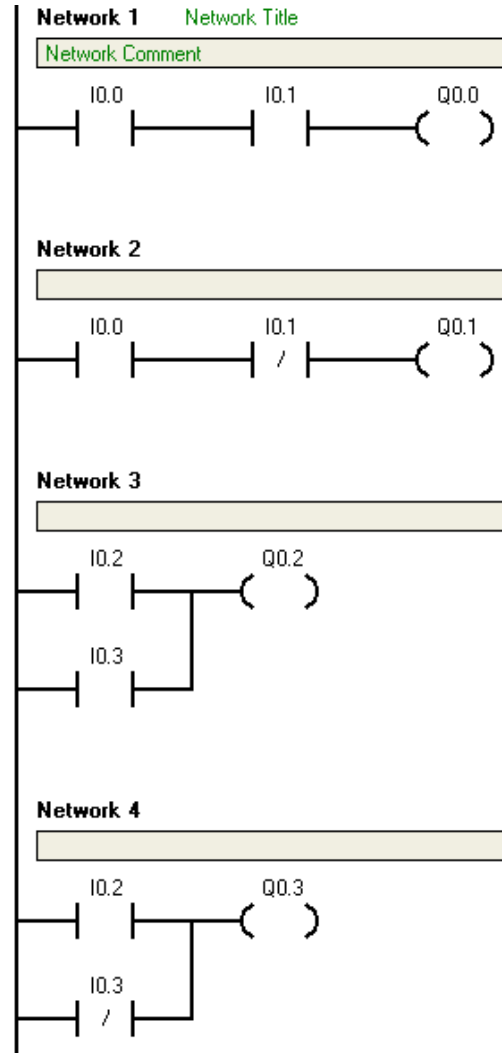
```
LD I 0.0 //I 0.0 girişini oku
OR I 0.1 //I 0.0 girişi ile I0.1 girişini Veya Değil işlemine tabi tut
= Q0.0 //Veya Değil işleminin sonucuna göre Q0.0 çıkışını 1 yap
```



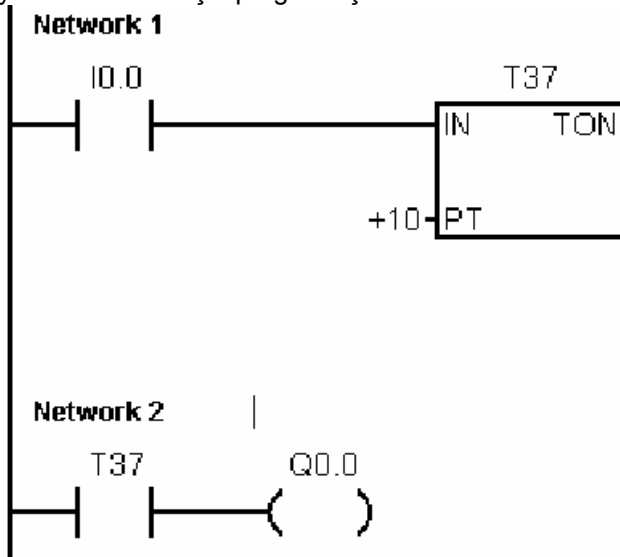


**Örnek Uygulama 5:**

Yandaki devreyi kurarak girişlerin her "0" ve "1" konumu için çıkışları gözlemleyiniz

**Zaman Rölesi Komutları****Çekmede Gecikmeli Zaman Rölesi ve Kalıcı Çekmede Gecikmeli Zaman Rölesi**

Çekmede gecikmeli zaman rölesi (TON) ve kalıcı çekmede gecikmeli zaman rölesi (TONR) komutları, giriş (IN) geldikten sonra çalışmaya başlar. PT değeri ile girilen süre sonunda değeri Lojik 1 olur. Aynı zamanda zaman rölesi numarası (Txx) zamanın katsayısını belirler. Zaman rölesinin değerinin "1" olmasının çıkışlara yansıtılabilmesi için program içerisindeki zaman rölesine ait açık veya kapalı kontaklar kullanılır.

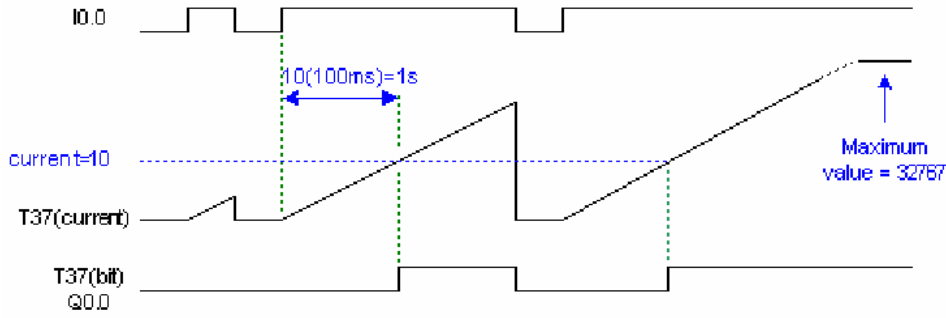


Network 1 //100 msn'lik zaman rölesi T37,  
I0.0  
// geldikten (10 x 100msn=1  
saniye)  
//sonra saymayı durdurur.

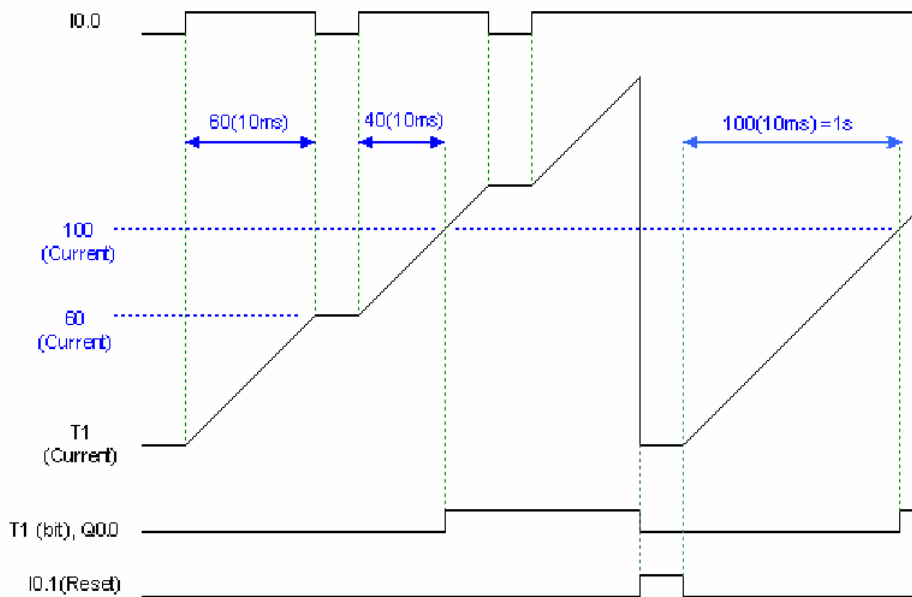
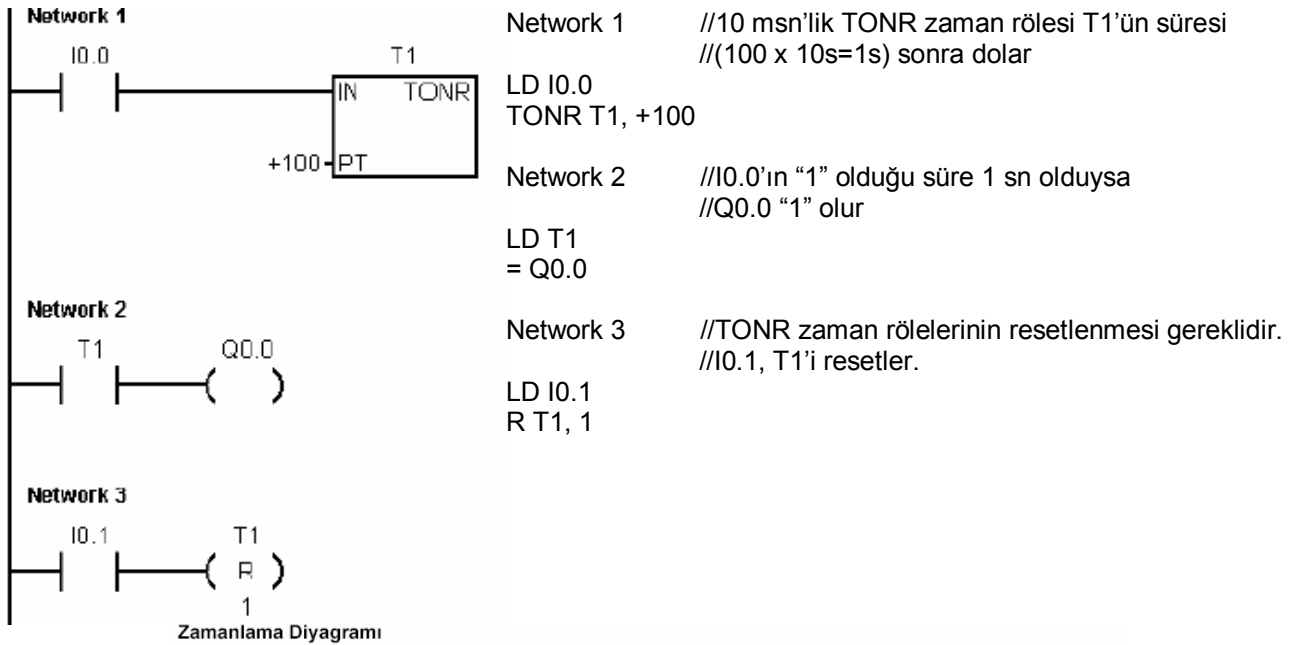
LD I0.0  
TON T37, +10

Network 2 //T37 biti, zaman rölesine bağlıdır  
LD T37  
= Q0.0

Zamanlama Diyagramı

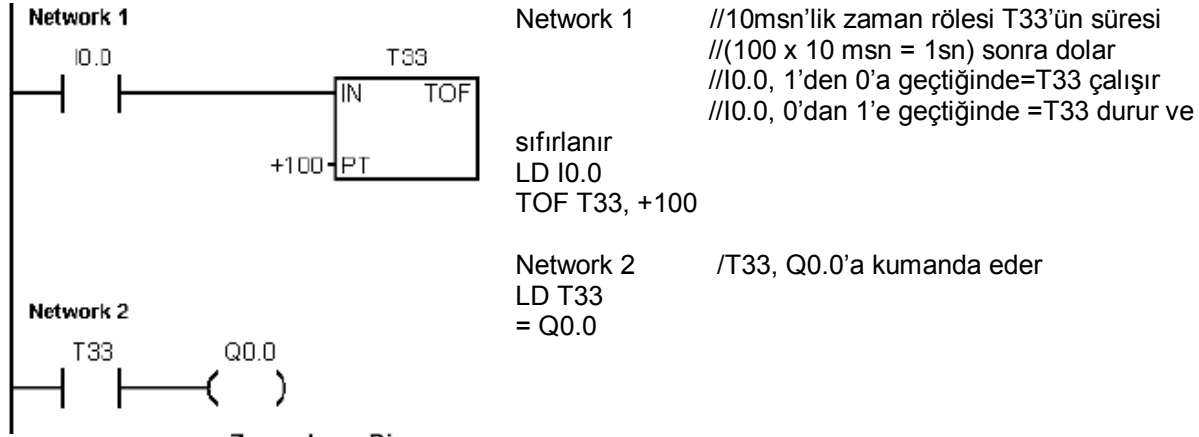
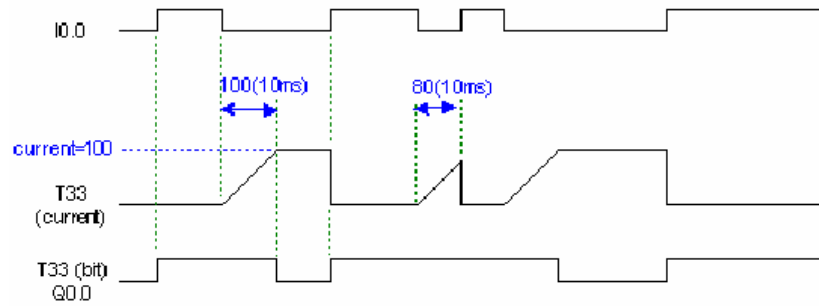


TON zaman rölesinde zaman sayımı in değeri olduğu sürece olmakta in girişi kesildiğinde TON zaman rölesinin değeri 0 olmakta idi. TONR zaman rölesinde ise zaman değeri bir ara hafızaya kaydedilir ve in girişi olduğu sürece zaman değeri toplanarak sayılır ve PT ile girilen zaman değerine gelindiğinde zaman rölesi "1" olur. TONR zaman rölesinde in değeri kesilse dahi zaman toplanarak gidildiğinden hiçbir zaman değeri 0 olmayacaktır. Bu nedenle zaman rölesinin değerini 0 yapmak için ayrıca resetleme elemanı kullanılmıştır.



**Düşmede Gecikmeli Zaman Rölesi**

Düşmede gecikmeli zaman rölesi (TOF) giriş (IN) gittikten sonra çıkışın girilen süre kadar bir süre daha çalışmasını sağlar. Zaman rölesi numarası (Txx) zamanın katsayısını belirler.

**Zamanlama Diyagramı**

TON'u belirli bir zaman aralığını tanımlamak için kullanabilirsiniz.

TONR'yi birden çok zaman aralığının toplamını almak için kullanabilirsiniz.

TOF'u bir işlemin bitmesini takiben zaman gecikmesi koymak amacıyla kullanabilirsiniz. Örneğin, bir motor durduktan sonra soğutma fanının bir süre daha çalışması için.

**Zaman röleleri için zamanın katsayısı**

Tip	Zaman Katsayısı	Maksimum Deger Zaman	Rölesi Numarası
TONR (kalıcı)	1 msn	32.767 sn (0.546 dk.)	T0 ve T64
	10 msn	327.67 sn (5.46 dk.)	T1 ila T4 arası ve T65 ila T68 arası
	100 msn	3276.7 sn (54.6 dk.)	T5 ila T31 arası ve T69 ila T95 arası
TON TOF	1 msn	32.767 sn (0.546 dk.)	T32 ve T96
	10 msn	327.67 sn (5.46 dk.)	T33 ila T36 arası ve T97 ila T100 arası
	100 msn	3276.7 sn (54.6 dk.)	T37 ila T63 arası ve T101 ila T255 arası

**Örnek Uygulama 6:**

Yandaki devreyi kurarak girişlerin her "0" ve "1" konumu için çıkışları gözlemleyiniz

T37 TON zaman rölesi için zaman hesabı :  
 $50 \times 100\text{ms} = 5000\text{ms} = 5\text{sn}$

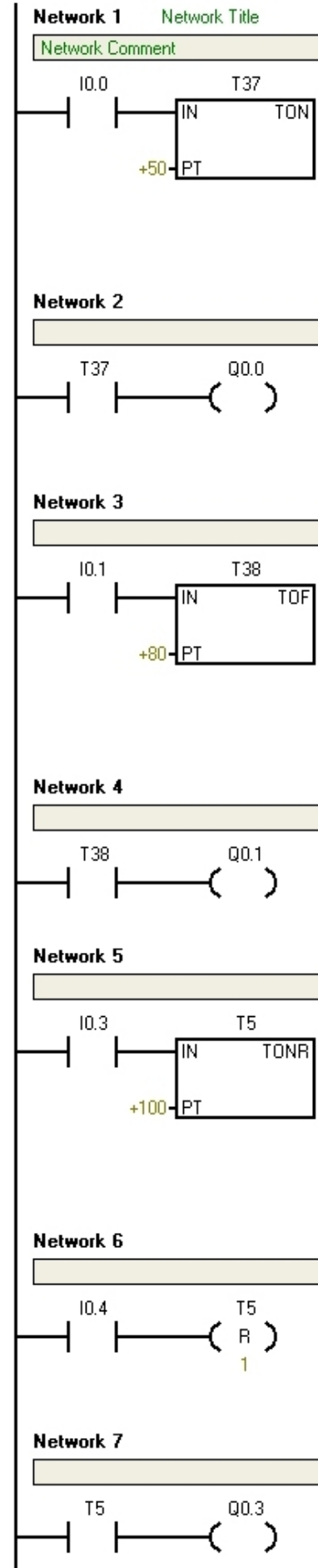
T38 TOF zaman röleleri için zaman hesabı :  
 $80 \times 100\text{ms} = 8000\text{ms} = 8\text{sn}$

T5 TONR zaman röleleri için zaman hesabı :  
 $100 \times 100\text{ms} = 10000\text{ms} = 10\text{sn}$

! T37 TON zaman rölesi girişini 5sn den önce keserek zaman rölesinin değerini gözlemleyiniz.

! T38 TOF zaman rölesi girişinin kesiminden itibaren 8sn den önce giriş vererek zaman rölesinin değerini gözlemleyiniz.

! T5 TONR zaman rölesi girişini zaman aralıkları ile verip keserek zaman rölesinin değerini gözlemleyiniz.



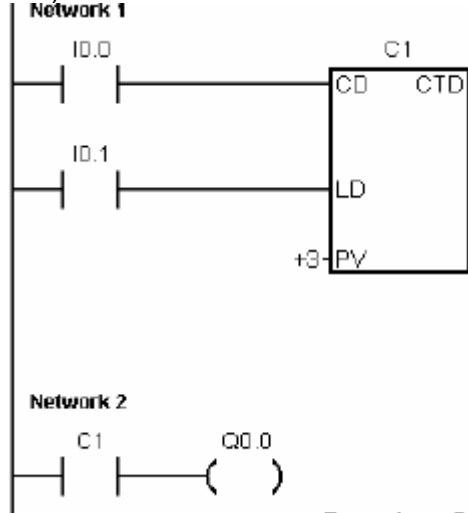
## Sayıcı Komutları

### Yukarı Sayıcı

Yukarı Say komutu (CTU), yukarı sayma girişinin (CU) her yükselen kenarında (0'dan 1'e dönüşümünde) anlık sayma değerini bir artırır. Cxx anlık değeri ayar değeri PV'ye eşit veya büyükse Cxx biti set olur. Reset (R) girişi geldiğinde ve Reset komutu işlendiğinde sayıcı değeri sıfırlanır. Maksimum değere (32767) ulaşıldığında sayma işlemi durur.

### Aşağı Sayıcı

Aşağı Say komutu (CTD), yukarı sayma girişinin (CD) her yükselen kenarında (0'dan 1'e dönüşümünde) anlık sayma değerini bir azaltır. Cxx anlık değeri 0'a eşitse Cxx biti set olur. LD (Load) girişi geldiğinde sayıcı biti sıfırlanır ve anlık değer PV değerine eşit yapılır. Sıfıra ulaşıldığında sayma işlemi durur (ve Cxx biti set olur).

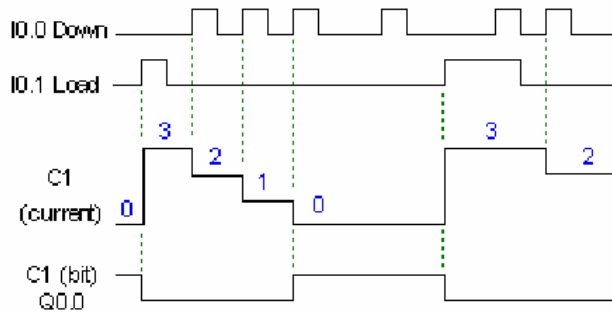


Network 1 //I0.1 yokken her I0.0 geldiğinde sayıcı C1, //3'den 0'a doğru geri sayar.  
//I0.1 geldiğinde sayıcı anlık değeri 3 olur

LD I0.0  
LD I0.1  
CTD C1, +3

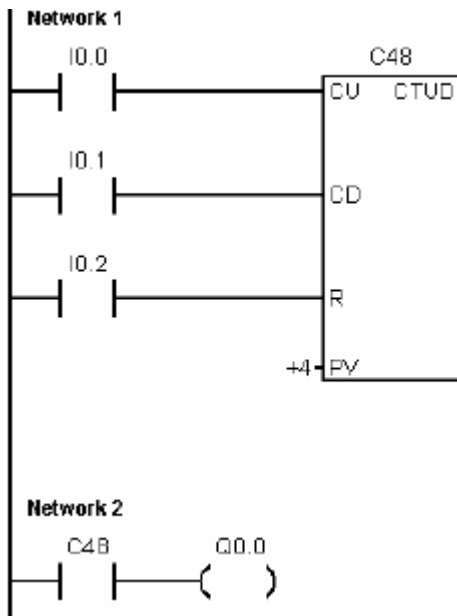
Network 2 //C1 biti anlık değer 0 iken 1'dir  
LD C1  
= Q0.0

Zamanlama Diyagramı



### Yukarı/Aşağı Sayıcı

Yukarı/Aşağı Say komutu (CTUD), yukarı sayma girişinin (CU) her yükselen kenarında yukarı sayar, aşağı sayma girişinin (CD) her yükselen kenarında ise aşağı sayar. Sayıcının anlık değeri Cxx o ana kadarki sayılan değeri saklar. Sayma işlemi yapıldığı anda anlık değer ile ayar değeri PV karşılaştırılır. Maksimum değere (32767) erişildiğinde yeni bir yukarı sayma girişi anlık değerinin minimum değere dönmesine neden olur (-32768). Aynı şekilde, minimum değere ulaşıldıktan sonraki aşağı sayma giriş sinyali anlık değerinin maksimum değer (32767) olmasına neden olur. Anlık değer Cxx ayar değeri PV'ye eşit veya büyükse Cxx biti set olur. Diğer durumda sıfırdır. Reset (R) girişi geldiğinde veya Reset komutu işlendiğinde sayıcı sıfırlanır. CTUD sayıcısı PV değerine ulaştığında sayma işlemi durur.

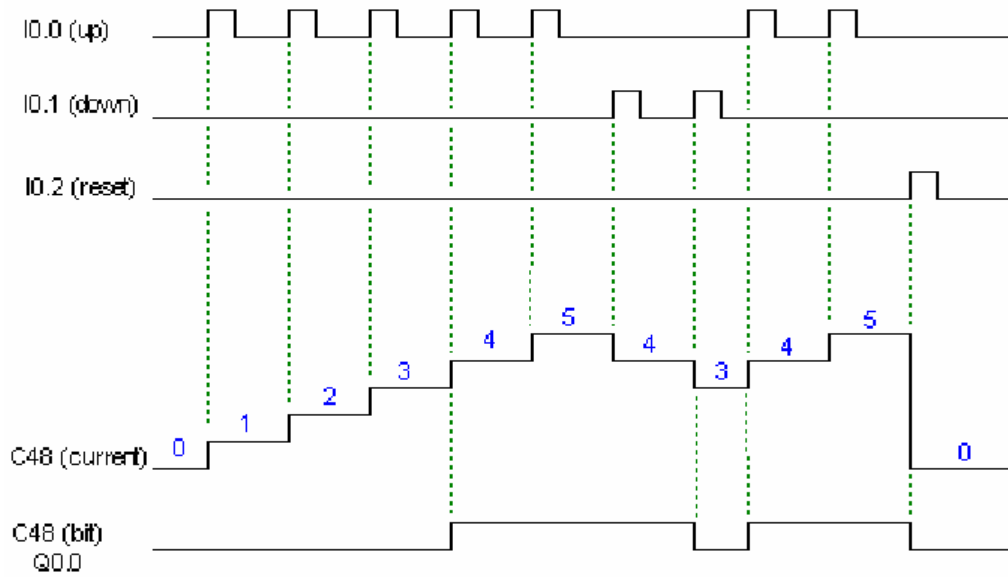


Network 1 //I0.0 yukarı sayar  
//I0.1 aşağı sayar  
//I0.2 sayıcı değerini sıfırlar

LD I0.0  
LD I0.1  
LD I0.2  
CTUD C48, +4

Network 2 //Anlık değer >=4 iken  
//C48 biti 1 olur

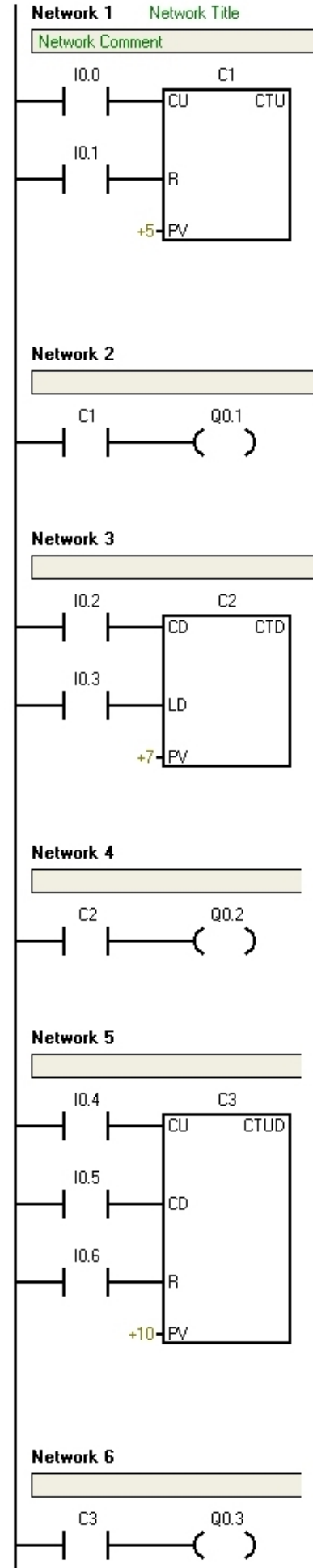
LD C48  
= Q0.0





**Örnek Uygulama 7:**

Yandaki devreyi kurarak girişlerin her "0" ve "1" konumu için çıkışları gözlemleyiniz



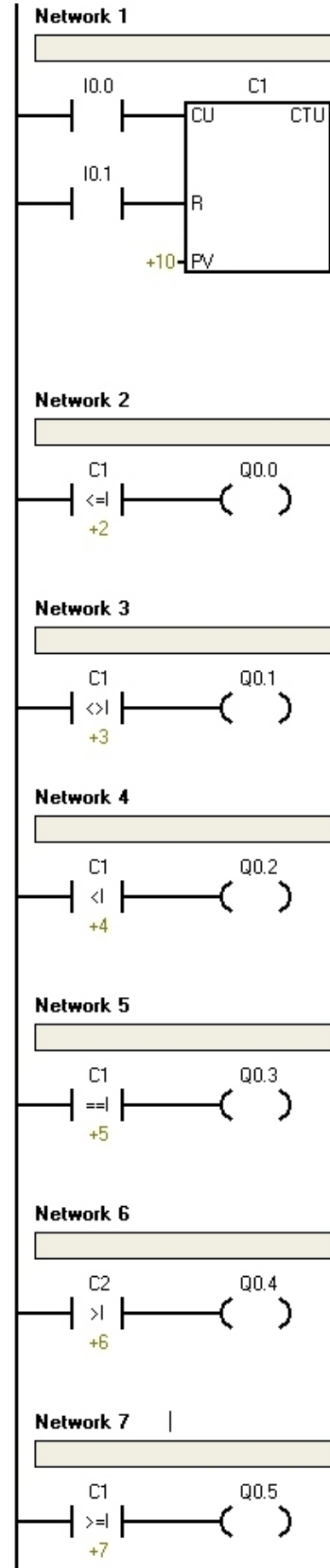
**Karşılaştırma Komutları****Nümerik Değerlerin Karşılaştırılması**

Karşılaştırma komutları iki değer arasındaki şartları sağlayıp sağlamadığına bakar:

$IN1 = IN2$     $IN1 \geq IN2$     $IN1 \leq IN2$     $IN1 > IN2$

$IN1 < IN2$     $IN1 \neq IN2$

Karşılaştırma sonucu doğru ise, kontak kapanır ("1" olur).

**Örnek Uygulama 8:**

## Sembolik Adresleme için Sembol Tablosunu Kullanmak



Sembol tablosu kullanarak değişkenlere isim verebilirsiniz. Böylece programınızda değişkenlerin

mutlak adresi değil, sembolik isimleri görülebilir. Birden çok sembol tablosu oluşturulabilir, ancak her bir tablo aynı adresleri içeremez. Aynı adres aynı tablo içinde birden fazla yer alamaz

Bunun dışında sembol tablosunda sistem tarafından tanıtilan sembollerle ilgili bir bölme de vardır. Sembol tablosu, global değişken tablosu olarak da isimlendirilir. Komutlarda kullanılan operandları mutlak veya sembolik olarak adresleyebilirsiniz. Mutlak adres, hafıza alanı ve bit ve bayt adresini içerir (I0.0 gibi). Sembolik adres ise alfa nümerik kombinasyonlardan oluşur (Start\_Butonu gibi). SIMATIC programları için bu adres atamaları sembol tablosunda, IEC programları için global değişken tablosunda yapılır. Bir adrese sembol atamak için:

1. Araştırma çubuğu üzerindeki sembol tablosu simgesini tıklayın.
2. Sembol ismini (azami 23 karakter) Symbol Name sütununa girin.
3. Adresi (örneğin, I0.0) Address sütununa girin.
4. IEC global değişken tablosu için, Data Type sütununa değer girin veya listeden seçin.

	Symbol	Address	Comment
1	AlwaysOn	SM0.0	Always on contact
2	Pump1	Q2.3	Pump 1 on/off
3	Pump1Limit	I1.1	Pump 1 pressure limit switch
4	Pump1Pressure	VD100	Pump 1 current pressure (real)
5	Pump1Rpm	VW200	Pump1 RPMs (integer)
6			

## Lokal Değişkenlerin Kullanımı

Program editöründeki lokal değişken tablosunu, o altprograma has değişkenlerin tanımlanması için kullanırsınız. Lokal değişkenler altprograma aktarılabilecek parametreler şeklinde de girilebilir ve altprogramın değişik parametrelerle birden çok kullanılabilmesine yarar.

Name	Var Type	Data Type	Comment
EN	IN	BOOL	
L0.0	IN	BOOL	First pass flag
LB1	IN	BYTE	Address of slave device
LW2	IN	INT	Data to write to slave
LB4	IN_OUT	BYTE	Status of write
L5.0	OUT	BOOL	Done flag
LW6	OUT	WORD	Error number (if any)
	OUT		
	TEMP		

## Programı İzlemek için Durum Tablosu Kullanmak



Durum tablosu (status chart) S7-200 programınızı çalıştırırken on-line olarak

proses değişkenlerini izlemenizi ve değiştirmenizi sağlar. Girişlerin, çıkışların ve diğer değişkenlerin anlık değerlerini izleyebilir, bazılarının değerlerini değiştirebilir veya force edebilirsiniz (sizin istediğiniz bir değere zorlayabilirsiniz).

Programınızın değişik kısımlarından değişik elemanları görüntülemek amacıyla birden çok durum tablosu oluşturabilir ve bunları kaydedebilirsiniz.

Durum tablosuna erişim için **View > Component > Status Chart** menü komutunu tıklayın veya araştırma çubuğundan Status Chart simgesini tıklayın. Durum tablosuna izlemek istediğiniz değişkenin adresini veya sembolik ismini girmelisiniz. Sabitlerin, akümülatörlerin ve lokal değişkenlerin durumunu izleyemezsiniz (bunları izlemeniz gerekli ise programın ilgili kısmında V adreslerine aktarabilirsiniz). Zaman rölesi ve sayıcıları hem bit olarak hem de anlık durumları açısından izleyebilirsiniz. Bir durum tablosu oluşturmak ve değişkenleri izlemek için:

1. İzlemeyi istediğiniz adresleri Address alanına girin.
2. Format sütunundan veri tipini seçin.
3. Değişkenlerin değerlerini görmek için **Debug > Chart Status** menü komutunu seçin.
4. Değerleri sürekli veya sadece bir tarama için izlemek amacıyla araç çubuğunda ilgili butonu tıklayın.

Durum tablosu değişkenlere yeni değerler vermek veya force etmek için de kullanılır. Durum tablosuna yeni satırlar girmek için **Edit > Insert > Row** menü komutunu seçin.

## Bilgi Notu

Birden çok durum tablosu oluşturup kendi içerisinde lojik bütünlüğe sahip olan değişkenleri aynı tabloya yerleştirmeniz ve sembolik adresleme kullanmanız, izleme işlemlerini oldukça kolaylaştıracaktır.

Address	Format	Current Value	New Value
1	Bit	0	
2	Bit	0	
3	Signed	0	
4	Signed	0	
5	Bit	0	
6	Hexadecimal	16#00	
7	Floating Point	0.0	
8	Signed		

**Lader Diyagramının On-line İzlenmesi**

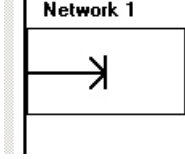
Plc nin çalışması sırasında plc ye on-line bağlanarak lader diyagramında kontak ve bobinlerin durumunu diyagram üzerinde görmemizi sağlar. Bunun için programın üst tarafındaki debug menüsündeki Program status butonunu kullanınız.

**Hataların Bulunması**

Yazmış olduğunuz programda yapmış olduğunuz olası hataları bulmak için kullanılır. Bunun için programın üst tarafındaki standart menüsündeki Compile ve Compile All butonlarını kullanınız.

### Kontak ve çıkış bobininin yerleştirilmesi

1- Fare ile network 'un başlangıcını seçiniz.

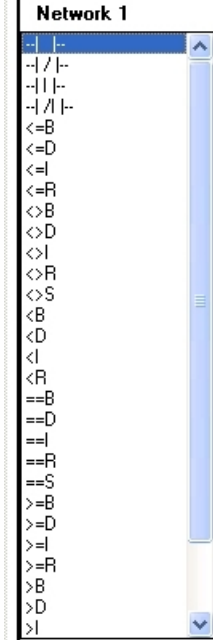


2- Yukarıdaki menüde bulunan kontak ikonuna fare ile tıklayınız veya F4 tuşuna basınız.

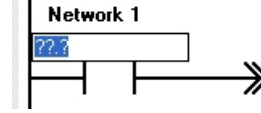
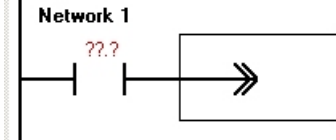


Contact - F4

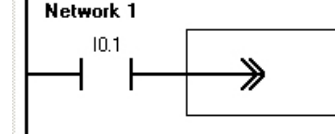
3- Açılan menüden eklemek istediğiniz kontak elemanını seçerek çift tıklayınız.



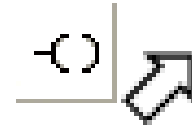
4- Kontak elemanının üzerindeki ??? üzerine tıklayarak kontak ismini klavyeden giriniz.



5- Fare ile kontağın sağ tarafındaki hattı seçiniz.

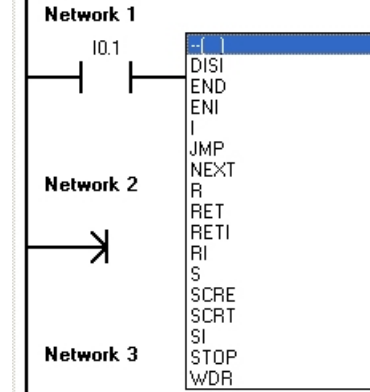


6- Yukarıdaki menüde bulunan bobin ikonuna fare ile tıklayınız veya F6 tuşuna basınız.

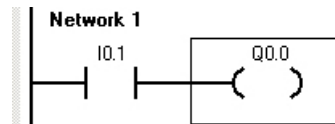
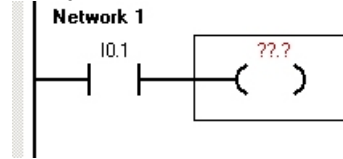


Coil - F6

7- Açılan menüden eklemek istediğiniz bobin elemanını seçerek çift tıklayınız.

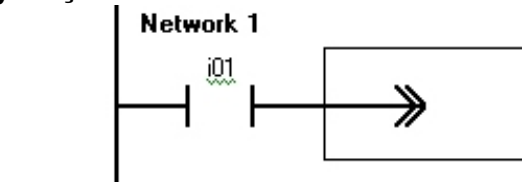


8- Bobin elemanının üzerindeki ??? üzerine tıklayarak bobin ismini klavyeden giriniz.



### Yazım yanlışı

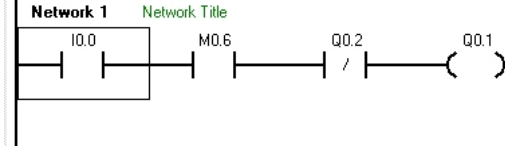
Eğer bir elemanın ismini yanlış yazacak olursanız verdiğiniz isim ya kırmızı renkli gözükecek veya altı yeşille çizilecektir.



**Paralel kontak ekleme :**

Aşağıda M0.6 kontağına paralel bir kontak eklenecektir.

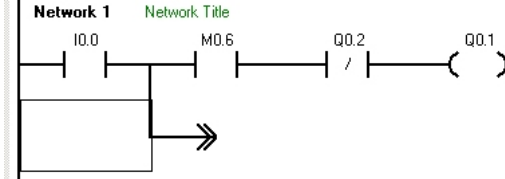
1- Paralel kontağı alacağınız hattın sol tarafını fare ile seçiniz.



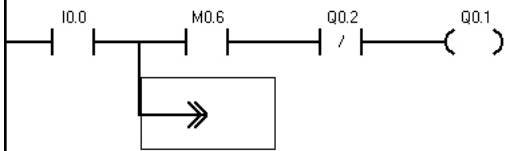
2- Yukarıdaki menüden aşağı ok ikonuna fare ile tıklayınız.



3- Şuanda seçili olan alan kontağı ekleyeceğiniz alan olmadığına dikkat ediniz.



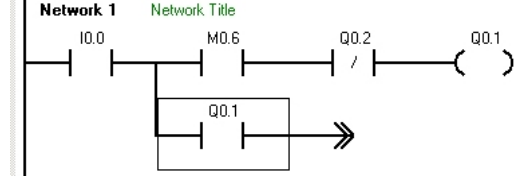
4- Kontakı eklemek istediğiniz yeri fare ile seçiniz.



5- Yukarıdaki menüde bulunan kontak ikonuna fare ile tıklayınız veya F4 tuşuna basınız.



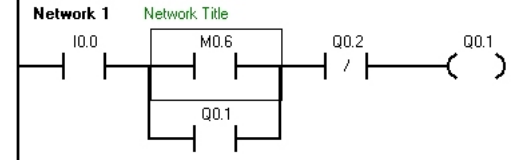
6- Yeni eklenen kontak elamanına isim veriniz ve seçili bırakınız.



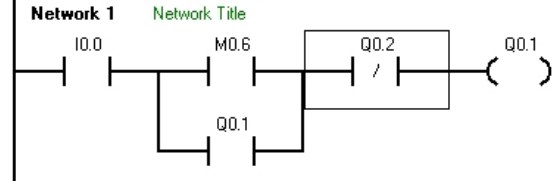
7- Yukarıdaki menüden yukarı ok ikonuna fare ile tıklayınız.



8- Paralel kontağınız eklenmiştir.

**Kontağı ileri öteleme**

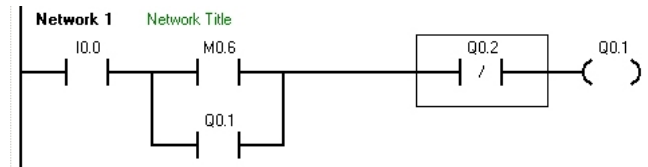
1- Devreye sonradan bir kontak ekleme için kontak ekleyeceğiniz yeri fare ile seçiniz.



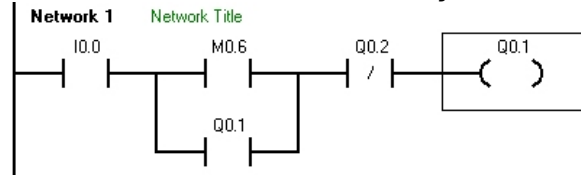
2- Yukarıdaki menüden ileri ok ikonuna fare ile tıklayınız.



3- Aranın açıldığını göreceksiniz.

**Çıkış elemanını ileri öteleme:**

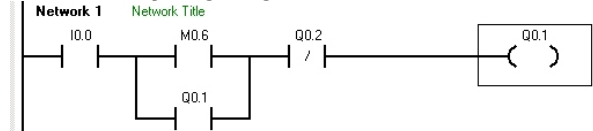
1-Ötelenecek bobin elemanını seçiniz.



2-Yukarıdaki menüden ileri ok ikonuna fare ile tıklayınız.

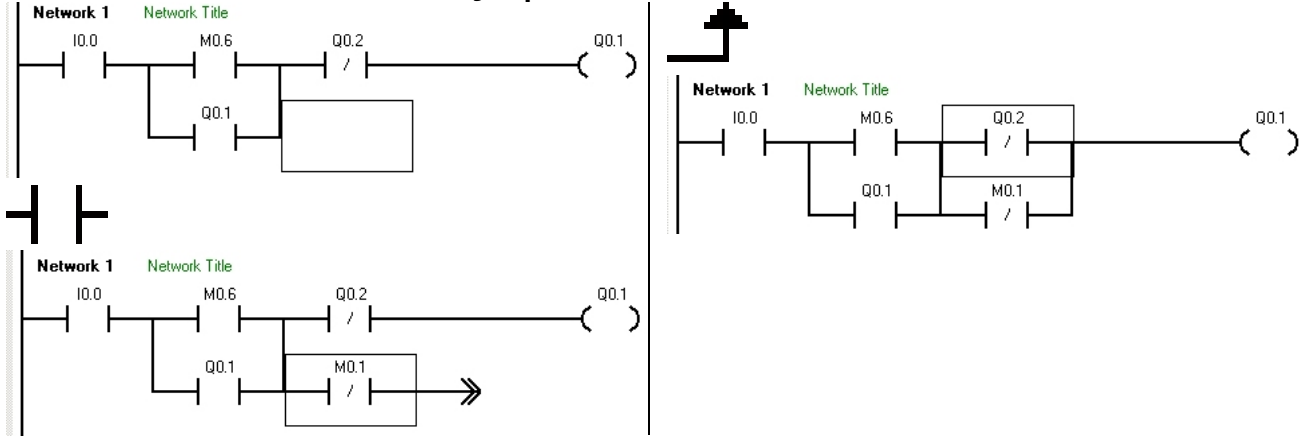


3-Aranın açıldığını göreceksiniz.



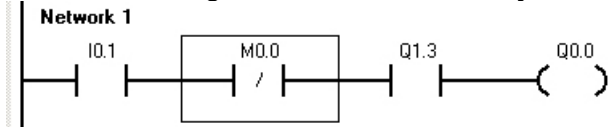


## Çift paralel kontak ekleme

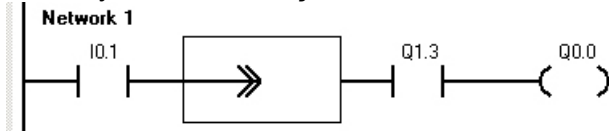


## Eleman Silmek

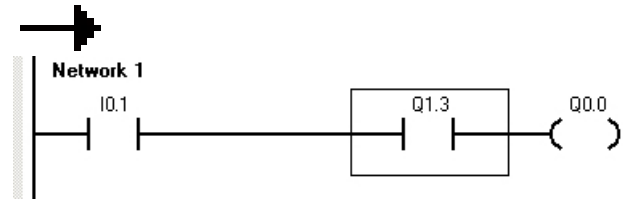
1- Silmek istediğiniz elemanı fare ile seçiniz



2- Klavyeden "DEL" tuşuna basınız.

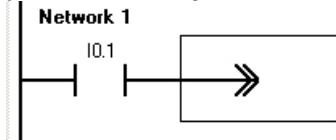


3-Arada oluşan boşluğu ileri öteleme ikonuna basarak kapatınız.

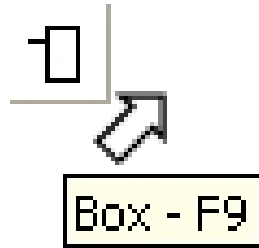


## Zaman rölesi ve sayıcı eklemek

1- Zaman rölesi veya sayıcı eklemek istediğiniz yeri fare ile seçiniz.



2- Yukarıdaki menüde bulunan kutu ikonuna fare ile tıklayınız veya F9 tuşuna basınız.



3- Açılan menüden eklemek istediğiniz zaman rölesini veya sayıcıyı seçiniz.

