

T.C.
MİLLÎ EĞİTİM BAKANLIĞI



MEGEP

(MESLEKÎ EĞİTİM VE ÖĞRETİM SİSTEMİNİN
GÜÇLENDİRİLMESİ PROJESİ)

ENDÜSTRİYEL OTOMASYON TEKNOLOJİLERİ

PLC PROGRAMLAMA

ANKARA 2007

Milli Eğitim Bakanlığı tarafından geliştirilen modüller;

- Talim ve Terbiye Kurulu Başkanlığının 02.06.2006 tarih ve 269 sayılı Kararı ile onaylanan, Mesleki ve Teknik Eğitim Okul ve Kurumlarında kademeli olarak yaygınlaştırılan 42 alan ve 192 dala ait çerçeve öğretim programlarında amaçlanan mesleki yeterlikleri kazandırmaya yönelik geliştirilmiş öğretim materyalleridir (Ders Notlarıdır).
- Modüller, bireylere mesleki yeterlik kazandırmak ve bireysel öğrenmeye rehberlik etmek amacıyla öğrenme materyali olarak hazırlanmış, denenmek ve geliştirilmek üzere Mesleki ve Teknik Eğitim Okul ve Kurumlarında uygulanmaya başlanmıştır.
- Modüller teknolojik gelişmelere paralel olarak, amaçlanan yeterliği kazandırmak koşulu ile eğitim öğretim sırasında geliştirilebilir ve yapılması önerilen değişiklikler Bakanlıkta ilgili birime bildirilir.
- Örgün ve yaygın eğitim kurumları, işletmeler ve kendi kendine mesleki yeterlik kazanmak isteyen bireyler modüllere internet üzerinden ulaşılabilir.
- Basılmış modüller, eğitim kurumlarında öğrencilere ücretsiz olarak dağıtılır.
- Modüller hiçbir şekilde ticari amaçla kullanılamaz ve ücret karşılığında satılamaz.

İÇİNDEKİLER

AÇIKLAMALAR	iii
GİRİŞ	1
ÖĞRENME FAALİYETİ- 1	3
1. TEMEL SEVİYE PLC KULLANIMI.....	3
1.1. PLC Hakkında Temel Bilgi	3
1.1.1. PLC'nin Tarihi	4
1.1.2. PLC'nin Yapısı	4
1.1.3. Röle ile PLC'nin Karşılaştırılması	7
1.1.4. PLC Çıkış Çeşitleri	9
1.2. PLC Programlama.....	11
1.2.1. Merdiven Diyagram ile Programlama	11
1.2.2. SFC ile Programlama	13
1.2.3. Çeşitli Marka PLC'lerin Röle Sayıları	14
1.3. Devre Dizaynı	15
1.3.1. Devre Dizaynının Akışı	15
1.3.2. Programlama Gereçleri.....	17
1.4. PLC için Komut Dili (Öğretici Dil)	18
1.4.1. Temel Bilgi	19
1.4.2. Programlama için Uyarıcı İfadeler	28
UYGULAMA FAALİYETLERİ.....	32
ÖLÇME VE DEĞERLENDİRME	35
ÖĞRENME FAALİYETİ- 2.....	37
2. İLERİ SEVİYE PLC KULLANIMI.....	37
2.1. Sayısal Veri Kavramı	37
2.1.1. İkilik Sayı Sistemi	37
2.1.2. Sekizlik ve Onaltılık Sayı Sistemi.....	37
2.1.3. BCD Sayı Sistemi	38
2.1.4. Veri Sistemi	39
2.1.5. 16 ve 32 Bitlik Verinin Yönetimi.....	40
2.2. Aygıt.....	41
2.2.1. Bit Aygıtı (Bit Device)	42
2.2.2. Kelime Aygıtı (Word Device)	42
2.2.3. Veri Taşıma	44
2.2.4. Verilerin Kodlarının Değiştirilerek Taşınması (BCD/BIN)	46
2.2.5. Artırma/Azaltma (Increment / Decrement).....	47
2.2.6. Karşılaştırma (Comparison).....	49
2.2.7. Toplama ve Çıkarma (Addition - Subtraction)	53
2.2.8. Çarpma ve Bölme (Multiplication and division)	54
2.3. Zaman Kontrol Komutları	55
2.3.1. Zamanlayıcı İşlemleri.....	55
2.3.2. DECO (Decode) Komutuyla Zamanın Set Edilmesi.....	58
2.3.3. TTMR Komutu ile Zamanın Set Edilmesi.....	60
2.3.4. Özel Zamanlayıcı	64
2.3.5. ALT Fonksiyonu	65
2.4. Sayıcı Fonksiyonu.....	67

2.4.1. 16 Bit Yukarı Sayıcı (up counter)	67
2.4.2. 32 Bit Yukarı / Aşağı Sayıcı	68
UYGULAMA FAALİYETLERİ	71
ÖLÇME VE DEĞERLENDİRME	76
MODÜL DEĞERLENDİRME	78
CEVAP ANAHTARLARI	79
KAYNAKÇA	80

AÇIKLAMALAR

KOD	523EO0322
ALAN	Endüstriyel Otomasyon Teknolojileri
DAL/MESLEK	Ortak Alan
MODÜLÜN ADI	PLC Programlama
MODÜLÜN TANIMI	Bu modül PLC yapısını, programlama mantığını ve genel data komutlarının kullanımına yönelik program yazımını anlatan öğrenme materyalidir.
SÜRE	40/32
ÖN KOŞUL	PLC'ye Hazırlık modülünü almış olmak
YETERLİK	PLC de programlama yapmak.
MODÜLÜN AMACI	Genel PLC programlamasını doğru olarak yapabileceksiniz. Amaçlar 1. Yapılacak işe uygun PLC seçimini doğru olarak yapabileceksiniz. 2. PLC'de merdiven diyagramı ile programlamayı doğru olarak yapabileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	Ortam: Arzılık kontrol laboratuvarı Donanım: Bilgisayar, PLC, kablo
ÖLÇME VE DEĞERLENDİRME	Her faaliyetin sonunda ölçme soruları ile öğrenme düzeyinizi ölçeceksiniz. Araştırmalarla, grup çalışmaları ve bireysel çalışmalarla öğretmen rehberliğinde ölçme ve değerlendirmeyi gerçekleştirebileceksiniz.



GİRİŞ

Sevgili Öğrenci,

PLC'ye Hazırlık modülünü tamamladıktan sonra bu modülde endüstriyel otomasyon alanında önemli yeri olan PLC'nin kullanımına yönelik temel çalışmalar yapmaya başlayacaksınız.

Bu amaçla öncelikle PLC yapısını tanımak ve programlanabilir kontrol mantığını anlamaya çalışmak gerekir. Programlanabilir mantıksal kontrol ile röleli kontrol uygulamaları arasında karşılaştırma yapabilmek için her iki kontrol yöntemini de yeterince kavramak gerekir. Bu modülde çeşitli algılayıcıların durumlarını dikkate alarak istenen şartlarda PLC çıkışına bağlı aktüatörlerin kontrolüne yönelik uygulamalar yapılacaktır. Başlangıçta verilen örnekler programlama mantığının anlaşılabilirliğini kolaylaştırmak için basit düzeyde seçilmiştir. Programlanabilir mantıksal kontrolün avantajları daha karmaşık uygulamalarda daha açık görülecektir.

Uygulama alanlarında çeşitli marka ve özelliklerde PLC'lerle karşılaşacaksınız. Fakat marka ve özellikleri farklı bile olsa genelde PLC'lerin programlama mantıkları birbirine yakındır. Bu modülde Mitsubishi marka PLC'ye uygun programlama yöntemlerini tanıyacak ve bunlarla ilgili uygulama yapma imkânı bulacaksınız.

Programlanabilir kontrol dendiğinde bilgi işleme ve bilginin iletimi söz konusu olur. Bunun için bu modülde sayı sistemlerini kısaca hatırladıktan sonra PLC üzerinde bilgi işlemeye yönelik uygulamalara da yer verilecektir.

Bu modül tamamlandığında programlanabilen mantıksal kontrol aygıtlarının temel düzeyde kullanımına ve programlanmasına yönelik yeterliklere sahip olacaksınız.



ÖĞRENME FAALİYETİ- 1

AMAÇ

Yapılacak işe uygun PLC seçimini doğru olarak yapabileceksiniz.

ARAŞTIRMA

Bu öğrenme faaliyetinden önce ;

- Ø Sanayide PLC'lerin kullanım alanları ile ilgili internet ortamında araştırma yapınız.
- Ø Röleli kontrolün sakıncalarını araştırınız.

1. TEMEL SEVİYE PLC KULLANIMI

1.1. PLC Hakkında Temel Bilgi

Programlanabilir mantık denetleyici (Programmable Logic Controller), PLC olarak kısaltılmaktadır. PLC, algılayıcılardan aldığı bilgiyi kendine verilen programa göre işleyen ve sonuçlarını iş elemanlarına aktaran bir mikrobilgisayar sistemidir. Röleli kontrol sisteminin olumsuz yönlerini gidermek amacıyla geliştirilmiştir. Zamanla özellikleri geliştirilerek ardışık kontrol, hareket denetimi (doğrusal ve döner hareket denetimi), süreç denetimi (sıcaklık, basınç, nem, hız), veri yönetimi (makine veya süreç hakkında veri toplama, izleme ve raporlama) gibi amaçlarla endüstriyel kontrol alanlarında kullanılabilir hale getirilmiştir.

Günümüzde MITSUBISHI, OMRON, SIEMENS gibi bir çok firma çeşitli özelliklerde PLC üretmektedir. Her firma, ürettiği PLC'ye uygun yazılımları da geliştirmektedir. Her ne kadar farklı firmalara ait PLC'lerin farklı özellikleri olsa da programlama mantığı birbirlerine çok benzemektedir.

Bu kitapta, MITSUBISHI FX2N PLC'nin bazı özellikleri açıklanacak ve kullanımına yönelik çalışma yapılacaktır.



Şekil 1.1: Çeşitli PLC görüntüleri

1.1.1. PLC'nin Tarihi

Programlanabilir mantık denetleyici bazen PC (Programmable controller) olarak da söylenilebilir. MITSUBISHI üretimi olan PLC'ler bazen "Sequencer" (art arda getiren) olarak da adlandırılır.

Tümüyle programlanabilen ilk denetleyici, 1968 yılında mühendislik alanında danışmanlık yapan Bedford Associates adlı bir firma tarafından General Motors firması için aşağıdaki tabloda belirtilen ihtiyaçlara tatmin edici cevap bulunabilmesi amacıyla geliştirilmiştir.

- Ø Program içeriğini değiştirmek ve bunu fabrika içerisinde yapabilmek mümkün ve kolay olmalıdır.
- Ø Programın sürdürülebilir olması, mümkün kılınmalı ve kurulan sistemin fişi prizine takıldığında çalışacak şekilde olmalıdır.
- Ø Fabrika içerisinde röleye göre çok daha yüksek güvenilirliğe sahip olmalıdır.
- Ø Röleli kontrol ile kıyaslandığında boyutları daha küçük olmalıdır.
- Ø Kontrol ünitesi, ana sisteme veri gönderebilmelidir.
- Ø Parasal yönden röleli sisteme göre pahalı olmamalıdır.
- Ø Giriş gerilimi olarak AC 115 Volt gerilimde kullanılabilmelidir.
- Ø Solenoid valf, motor sürücü gibi elemanların çalıştırılmasında, AC 115 Volt, 2 amper üzerindeki gerilim ve akım değerlerinde de kullanılabilmelidir.
- Ø Ana sistem, çalışan sistemin değişmesi durumunda genişleyebilmelidir.
- Ø En azından 4 Kilobayt'lık ve genişleyebilme imkânı veren, program yapılabilecek hafızaya sahip olmalıdır.

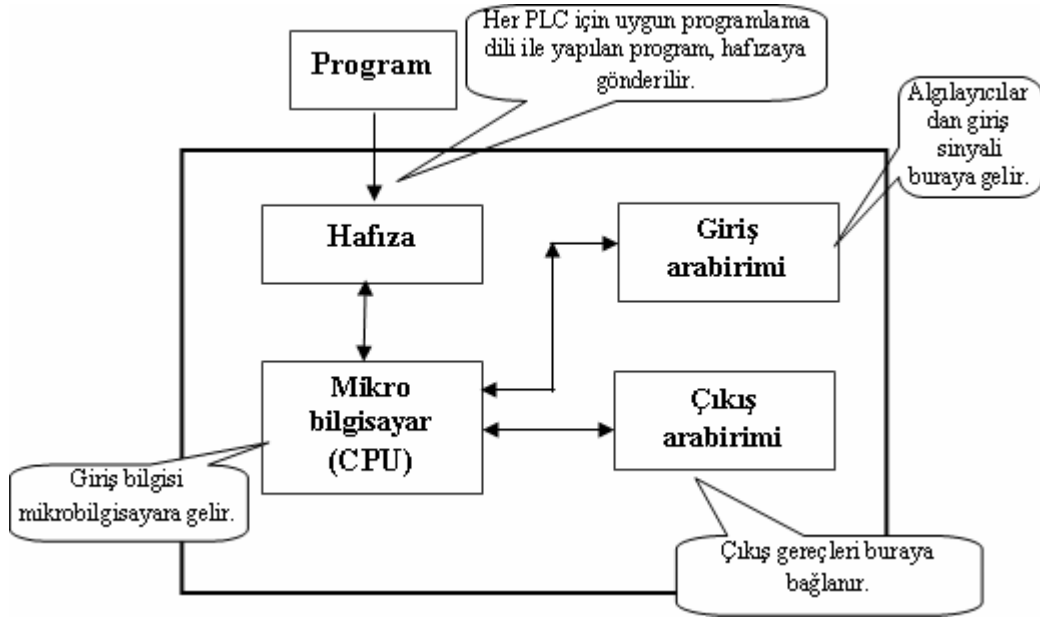
Zamanla PLC özellikleri geliştirilmiş ve 1978 yılında PLC'lerin birbirine veri aktarmasına imkân veren veri devresi tasarlanmıştır. 1980 yılında küçük, tek parça, düşük maliyetli, 64 giriş/çıkışlı ve daha güçlü bir PLC olan 84 Micro piyasaya sürülmüştür.

Başlangıçta yalnızca basit işlemler için kullanılan PLC'ler günümüzde çeşitli firmalar tarafından geliştirilerek çok daha karmaşık sorunları çözebilecek özelliklerle donatılmış ve endüstriyel kontrol alanlarında güvenle kullanılmaya başlanmıştır.

1.1.2. PLC'nin Yapısı

Aşağıdaki şekilde de görüleceği gibi PLC, içerisinde işlemci (CPU) , giriş-çıkış ara yüzleri ve hafıza (memory) bulunduran bir endüstriyel mikrobilgisayardır. PLC ile kontrolde, programlama konsolundan PLC'ye çeşitli giriş sinyalleri ve kontrol içerikleri gönderilir. Sonuç olarak da çıkış sinyalleri, çıkış elemanlarına yöneltilir.

Giriş, çıkış gereçleri ve PLC arasındaki sinyal değişiminde giriş arabirimi ve çıkış arabirimi olarak adlandırılan tekrarlayıcı (repeater) gereklidir.

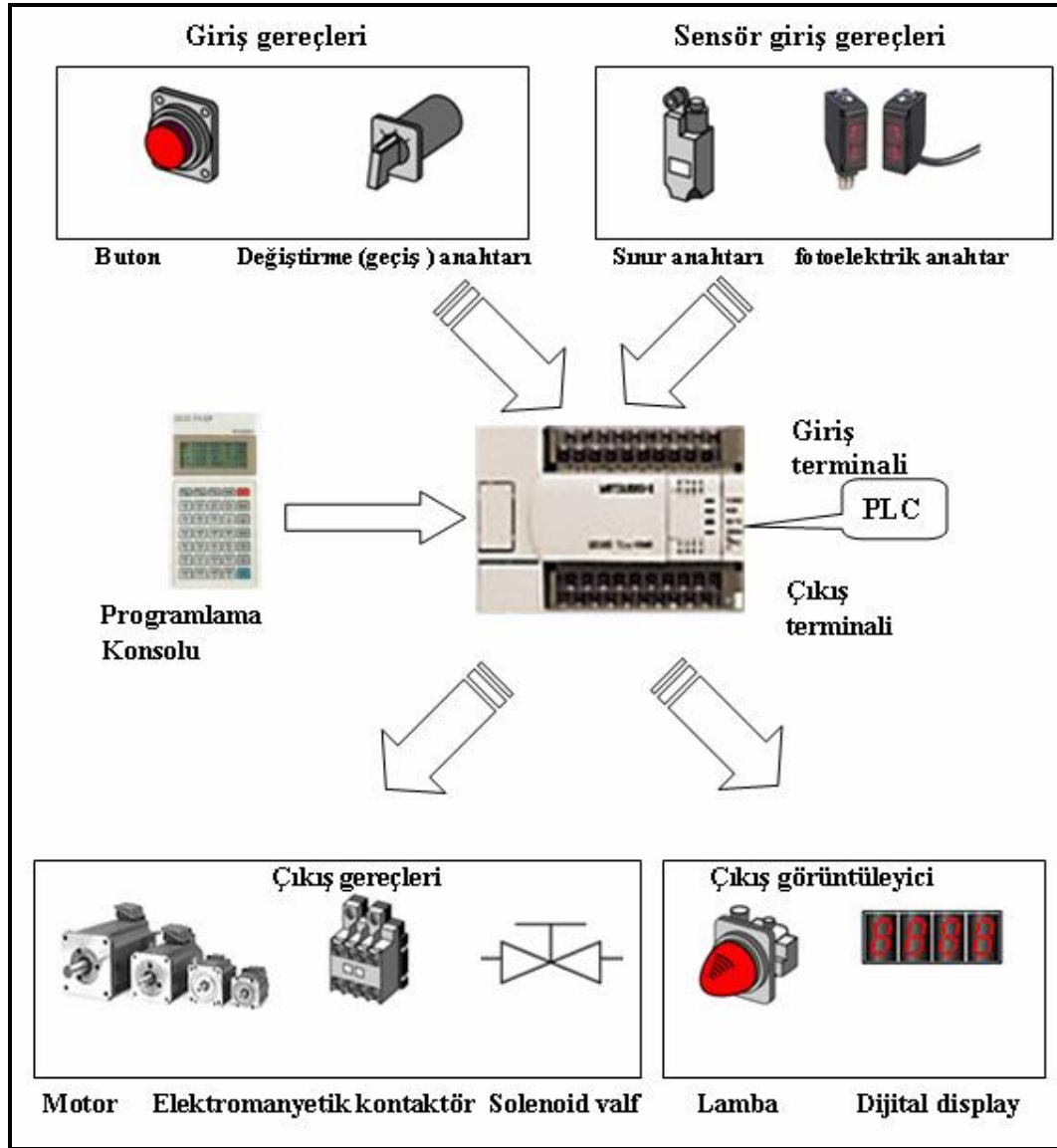


Şekil 1.2: PLC'nin iç yapısı

Şekil 1.3, PLC ile çevre birimleri arasındaki bağlantıyı göstermektedir. Giriş elemanı olarak buton, anahtar, seçici anahtar, dijital anahtar, sınır anahtarı, fotoelektrik anahtar, yaklaşım anahtarı vb. kullanılabilir. Bu elemanlardan gelen sinyaller PLC'nin işlemcisine giriş arabirimi üzerinden gönderilir.

İşlemci (CPU), hafızaya kaydedilmiş olan programın içeriğine ve giriş sinyallerine göre çıkış sinyallerini kontrol eder.

Çalıştırılması istenen donanımlar çıkış elemanı olarak isimlendirilir. Elektromanyetik valf, lamba, küçük güçlü motor vb. alıcılar elektriksel değerleri uygun olması durumunda doğrudan PLC'ye bağlanabileceği gibi transistör, röle vb. diğer kontrol elemanları üzerinden de kontrol edilebilir.



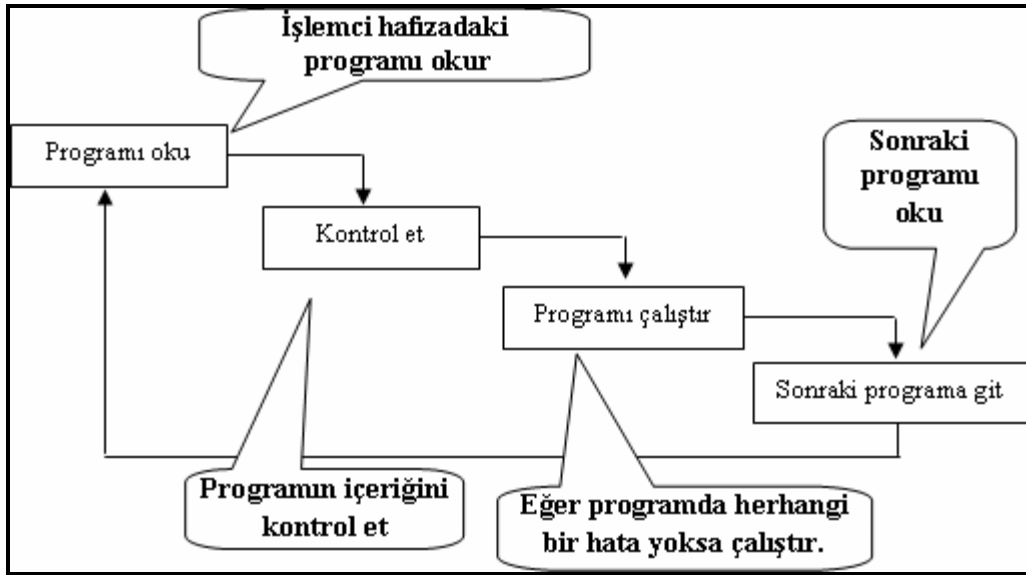
Şekil 1.3 : PLC ile giriş çıkış gereçleri arasındaki bağlantı

Referans: Programlama konsolu, komut listesindeki komutları kullanarak program yazmak ve bu programı PLC'ye göndererek kaydetmek için kullanılan bir programlama gerecidir.

Hafızaya kaydedilen program işlemci (CPU) tarafından okunur. Okunduktan sonra da programın içeriği yapılabirliğine göre kontrol edilir. Program uygulanabilir olduğunda, CPU tarafından çalıştırılır ve bir sonraki adıma geçilir. Hafızadaki tüm program satır satır işlenir.

PLC'nin hafıza kapasitesi, tipine bağlı olarak değişir. Örneğin MITSUBISHI firmasının küçük tip PLC'si olan FX2 serisinde, program kapasitesi 2000 adım (adım 0'dan adım 1999'a kadar) ve program kapasitesi 8000 adım (adım 0'dan adım 7999'a kadar) olan iki tipi vardır. Program kapasitesi, müşterinin ihtiyacını karşılayacak kapasitede olmalıdır.

OMRON firmasının da SYSMAC – S6 ve C500 gibi ürünleri ile müşterinin farklı ihtiyaçlarına cevap verebilecek program kapasitesine ve giriş-çıkış terminallerine sahip PLC'leri piyasaya sunmuştur.



Şekil 1.4: Programın işleyişi

1.1.3. Röle ile PLC'nin Karşılaştırılması

Aşağıda PLC kullanımının röleli kontrole göre çeşitli açılardan üstünlükleri belirtilmiştir.

Ø Ekonomi

Fabrikasyonda 10 ve üzeri elektromanyetik kontaktör veya anahtar kullanımı PLC kullanımına göre daha pahalıdır.

Ø Tasarımda iş gücü kazancı

Gereçlerin çevre planlaması basitleştirilebilir. PLC kullanıldığında kontrol devresinin dizaynı ve değiştirilmesi daha kolaydır. Aynı elemanlar için çalışma şeklini değiştirmek isterseniz PLC içerisindeki programı değiştirmek yeterli olacaktır.

Ø Üretim süresinde azalma

Gereçlerin teknik özelliklerinin değiştirilmesinde ve parçaların montajının basitleştirilmesinde esneklikler vardır. Kablolamanın basitleştirilmesi sayesinde üretim süresi azaltılır ve böylece genel maliyet de azaltılmış olur.

Ø Küçük boy ve standartlaştırma

PLC, röleli kontrole göre çok daha küçük boyutludur. Programın tekrar tekrar kullanımı sayesinde seri üretim mümkündür. (Program EPROM içerisinde saklanır.)

Ø Güvenilirliğin geliştirilmesi

Röleler ve zamanlayıcılardaki karışıklık azaltılabilir. (Kontaklarda bulunan toz ve rölenin titreşimi, kontakların “ON” ve “OFF” zamanlarında karışıklığa neden olur.)

Ø Bakım kolaylığının geliştirilmesi

PLC’nin ömrü, röleye göre daha uzundur. PLC’nin hata teşhis fonksiyonu kullanılarak sistemdeki hata kolaylıkla bulunup gerekli bakım kolaylıkla yapılabilir.

Röleli ve PLC ile yapılan kontrolün özellikleri aşağıdaki tabloda verilmiştir.

Maddeler	Röleli kontrol	PLC ile kontrol
1. Fonksiyon	Röle sayısını artırmak sureti ile karışık kontrol devreleri yapılabilir.	Programlama ile karışık kontrol devreleri yapılabilir.
2. Kontrolde değiştirilebilirlik	Kablolamayı değiştirmek gereklidir.	Programı değiştirerek kontrolde istenilen değişiklik yapılabilir.
3. Güvenilirlik	Eksik kontak yüzünden bazı kısıtlamaları vardır.	Yarı iletken teknolojisinden dolayı yüksek güvenilirliğe sahiptir.
4. Çok yönlülük	Diğer yerlerde kullanmada anlaşılmasının güç olduğu bilinmektedir.	Programın değiştirilmesi ile tekrar tekrar ve farklı yerlerde kullanılması mümkündür.
5. Genişleyebilirlik	Ekleme ve değişiklik yapmak zor ve karmaşıktır.	Çalışma limitine kadar genişlemesi mümkündür.
6. Bakım	Periyodik bakım yapmak ve aşınan parçaları değiştirmek gereklidir.	PLC üniteleri değiştirilebilir ve tamir edilebilir.
7. Teknik anlayış	Sıklıkla kullanıldığından anlaşılması kolaydır.	PLC donanımının içi kapalı kutudur.
8. Boyut	Genelde büyük boyutludur.	Çok farklı modellerde ve genelde de küçük boyutludur.
9. Dizayn süresi	Dizayn ve fabrikasyon için çok zamana ihtiyaç vardır.	Dizayn ve fabrikasyon için fazla zamana ihtiyaç yoktur.
10. Ekonomi	Karmaşık kontrol açısından pahalıdır. Elektromanyetik röle ve kontaktörler 10 defa kullanıma kadar ekonomiktir.	Karmaşık kontrollerde pahalı değildir.

1.1.4. PLC Çıkış Çeşitleri

Kontağa bağlı olarak çeşitli türde çıkışlar vardır. PLC bir yere kurulacağı zaman her bir çıkışın karakteristik özelliği dikkate alınmalıdır. PLC'nin şekline bakarak karar vermek zordur. Aşağıda her bir PLC'ye ait çıkış özellikleri açıklanmıştır.

Ø Röle çıkışlı PLC

Röle çıkışlı PLC'de çıkış kontak şeklindedir ve bu çok sık karşılaşılan çıkış şeklidir. Çıkış devresinin sadece kontakten oluşması nedeniyle bu PLC hem doğru hem de alternatif akımda kullanılabilir. Devreye 2 amper gibi yüksek değerli akım uygulanabilir. (Her bir kontağından 2 ampere kadar akım çekilebilir.) Kutupları olmadığı için çeşitli alanlarda kullanılması mümkündür. Örneğin DC motor, büyük güçlü elektromanyetik valf gibi. Cevap verme hızının düşük olması, mekanik hareket olmasından dolayı röle kontaklarının zamanla aşınması bu PLC'lerin dezavantajı olarak söylenilebilir.

Ø Triyak çıkışlı PLC

Bu tip PLC'lerde triyak veya tristör çıkış elemanı olarak kullanılır. Bu tip PLC'ler kontaksız tip PLC'lerdir. Triyak çıkışlı PLC'de çıkışa 85 – 242 Volt arasında alternatif gerilim uygulanmalıdır. Cevap verme süresi bakımından bu PLC röle çıkışlı PLC'ye göre çok daha hızlı fakat transistör çıkışlı PLC'ye göre ise daha yavaştır. Bu tip PLC'nin çıkışından alabileceğimiz akım ise 0,3 amper kadardır. Triyak çıkışlı PLC, uygulamada çok fazla yer bulmamasına rağmen, alternatif akım kullanılan ve kontrol panelinde röle çıkışlı PLC'ye sahip olan fabrikalarda yenileştirme yapmak amacı ile kullanılabilir.

Ø Transistör çıkışlı PLC

Transistör çıkışlı PLC'lerde Photo Coupler kullanılmaktadır. Çıkış akımı yaklaşık 0,5 amperdir. Bu PLC'lerde cevap verme süresi 0,2 ms gibi çok kısa bir süredir.

Transistör çıkışlı PLC'lerin bazılarında, özel bir ünite olmaksızın pozisyon kontrolü yapabilmek amacı ile pals çıkışı mevcuttur.

Transistör kullanıldığı için bu PLC'lerin çıkışında kutuplar vardır ve kablolama esnasında bu kutuplara özellikle dikkat edilmelidir. Otomasyona geçmiş olan fabrikalarda, son zamanlarda, çok miktarda küçük elektrik motoru ve valfler kullanılmaktadır. Bu PLC'ler bu tür cihazlara kumanda etmede kullanılmaktadır. Aynı zamanda robot ya da CNC kontrol ünitesinin kontrolü için de yüksek değerli akımlara ihtiyaç yoktur. Bu gibi alanlarda hızlı çalışması sebebi ile transistör çıkışlı PLC'ler tercih edilmektedir. Bu PLC'lerin mekanik kontağı olmadığı için aşınma ve gürültü olmamakta, böylece çıkış ünitesi çok daha uzun ömürlü olmakta ve sessiz çalışmaktadır.

Aşağıdaki tablo çeşitli çıkış tiplerine göre PLC'lerin özelliklerini göstermektedir.

Özellikler (MITSUBISHI FX2N)

Konu	Röle Çıkışlı	Triyak Çıkışlı	Transistör Çıkışlı
Anahtarlama Voltajı	-AC250V, -DC30V	AC85V-242V	DC5V-30V
Anma Akımı	2A / 1 kontak	0,3A / 1 kontak	0,5A / 1 kontak
Maksimum Yük	8A / com.	0,8A / com.	0,8A / com.
Maksimum endüktif yük	80VA	15VA/AC100V 30VA/AC200V	0,5A/24V DC (12W)
Açık devre akım sızıntısı	--	1mA/AC100V 2mA/AC200V	0,1mA/30V DC
Cevaplama süresi	OFF → ON 10ms ON → OFF 10ms	OFF → ON <1ms ON → OFF <10ms	OFF → ON <0.2ms ON → OFF <0.2ms
Akım izolasyonu	Röle ile	Photo tristör ile	Photo coupler ile
Çalışma göstergesi	Bobini enerjilendiğinde LED ışık verir.	Photo tristör sürüldüğünde LED ışık verir.	Photo coupler sürüldüğünde LED ışık verir.
Devre Dizaynı			

Şekil 1.5 : Çıkış tipine göre PLC'lerin özellikleri

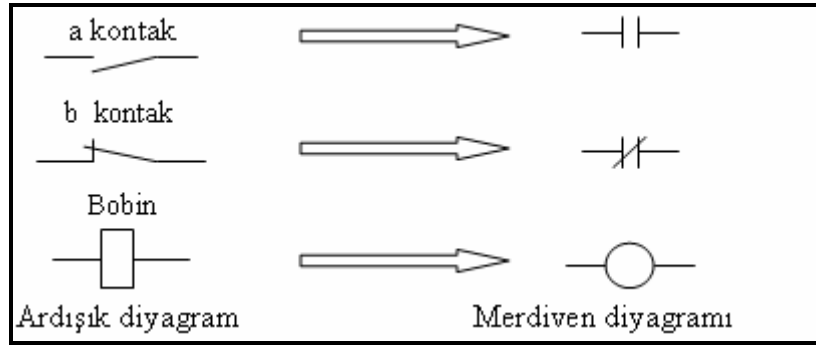
1.2. PLC Programlama

Kontrol içeriğini PLC hafızasında saklamak için programlama konsolunu kullanmak ya da kişisel bilgisayar kullanmak gibi farklı yollar vardır. Program, devre sembolleri (merdiven diyagram) kullanılarak yapılabileceği gibi doğrudan komutlar (öğretici dil) ile de yapılabilir.

1.2.1. Merdiven Diyagram ile Programlama

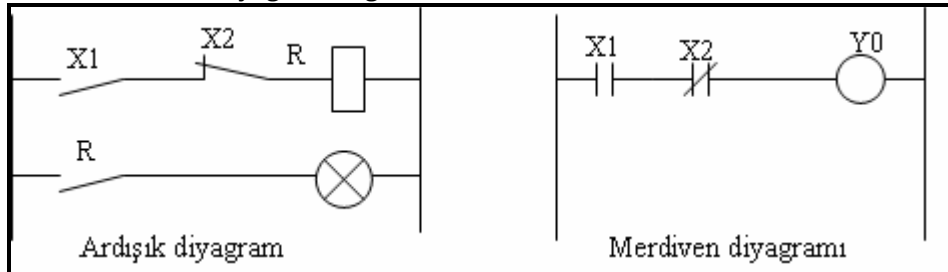
Merdiven (merdiven) diyagramı ile programlamada semboller kullanılır. Röleli kontrole alışık olanlar için merdiven diyagramı ile program yapmak daha kolaydır. Fakat komutların kullanımını da mutlaka öğrenmek gerekir.

Merdiven diyagram ile program yazmak röleli kontrolde çizilen ardışık diyagrama çok benzemektedir. Fakat ardışık diyagramda kullanılan semboller ile merdiven diyagramında kullanılan semboller farklıdır.



Aşağıda basit bir devre gösterilmektedir. Rölenin bobin ve kontağı ardışık diyagramda gösterilir. Fakat merdiven diyagramda, rölenin bobin ve kontağını çizmeye gerek yoktur. PLC’de hangi giriş şartlarında hangi çıkış ya da çıkışların aktif olacağını belirtmek yeterlidir. PLC, rölenin fonksiyonlarına da sahiptir. Bu nedenle röleyi çizmeye gerek yoktur.

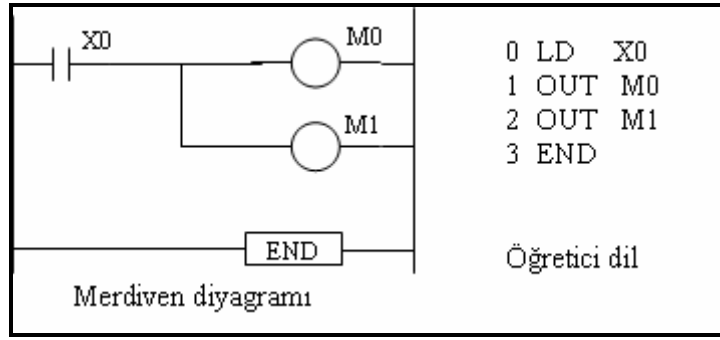
Aşağıdaki diyagram X1 (“a” kontak), X2 (“b” kontak) ve Y0’ın kullanıldığı bir ardışık diyagramı ve merdiven diyagramını göstermektedir.



Şekil 1.6 : PLC ardışık ve merdiven diyagramı

Merdiven diyagramı, özel öğretici dil kullanılarak da açıklanabilir. Öğretici dil ile yapılan programda, işlemin sırasını ve yöntemini görebiliriz. İşlemin sırası ve yöntemi, programın analizi açısından önemlidir. Programın, sol ana hattan başlayıp sağ ana hatta doğru çalışan her satırı “ünite” olarak tanımlanmaktadır. Bir satır, merdiven diyagramında bir ünite.

Bir ünitenin başlangıcı öğretici dilde LD komutu kullanılarak gösterilmelidir. Ünite sağ tarafta sona erdiğinde de öğretici dil için OUT komutu kullanılmalıdır. Aşağıdaki şekil buna bir örnek olarak verilmiştir. Öğretici dil her PLC üreticisi firma için farklı komutlarla yazılmaktadır. Fakat komutlar farklı olsa da mantık hemen hemen tüm ürünler için aynıdır. Bu kitapta, donanımımız MITSUBISHI firmasına ait olduğu için aynı firmaya ait olan öğretici dil kullanılmıştır.



Aşağıdaki tablo farklı firmalara ait öğretici dil ile ilgili bilgi vermektedir.

Mantıksal ifade	Sembol	Firma			
		MITSUBISHI	OMRON	SHARP	HITACHI
Mantıksal ifadeyi “a kontak” tan başlat (LoaD)		LD	LD	STR	ORG
Mantıksal ifadeyi “b kontak” tan başlat (LoaD Inverse)		LDI	LD NOT	STR NOT	ORG NOT
“a” kontağın mantıksal çoğaltılması, seri bağlantı (AND)		AND	AND	AND	AND
“b” kontağın mantıksal çoğaltılması, seri bağlantı (AND Inverse)		ANI	AND NOT	AND NOT	AND NOT

“a” kontağın paralel bağlanması, mantıksal ekleme (OR)		OR	OR	OR	OR
“b” kontağın paralel bağlanması, mantıksal ekleme (OR Inverse)		ORI	OR NOT	OR NOT	OR NOT
Mantıksal bloğun seri bağlantısı (AND Block)		ANB	AND LD	AND STR	AND ORG
Mantıksal bloğun paralel bağlantısı (OR Block)		ORB	OR LD	OR STR	OR ORG
Mantıksal işlemin çıkışı (OUT)		OUT	OUT	OUT	OUT
Mantıksal işlemde boşluk (Hiçbir şey yapılmaması)		NOP	NOP	NOP	NOP
Program sonunun tanımlanması (Bu programın bitmesi değil, işlemin sona gelmesi demektir.)		END	END	END	END

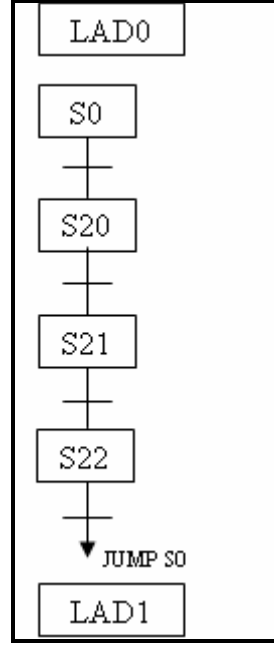
Şekil 1.7: Farklı firmalara ait öğretici dil komutları

1.2.2. SFC ile Programlama

SFC, Sequential Function Chart (Sıralı Fonksiyon Grafiği) ifadesinin kısaltılmış şeklidir. Durum geçiş grafiği anlamına da gelir. SFC, bir programlama metodudur ve her zaman değişen, gelişen adım işlemleri bu yöntemle gösterilir.

SFC ile, ardışık kontrol ya da lojik kontrol devreleri dizayn edilirken, geleneksel ve karmaşık dizayn yöntemleri kullanılmadan, çok kolay bir şekilde programlama yapılır. Bu sistem, kolay programlama yapılabildiğinden dolayı, IEC standardı için dikkate değerdir. Bütün bu olumlu yanlarına bakacak olursak, SFC sistemi hızla gelişecek gibi görünmektedir. SFC ile programlama ayrıca bir modülde açıklanacaktır.

Aşağıdaki şekil SFC diyagramına bir örnek olarak verilmiştir.



Şekil 1.8: Örnek SFC diyagram

1.2.3. Çeşitli Marka PLC'lerin Röle Sayıları

Üretici firmalar PLC içinde kullanılabilecek röle, zamanlayıcı, sayıcı gibi elemanları kendilerine özgü sayıda yapmakta ve numaralandırmaktadır. Programlama esnasında bu sınırlamalara dikkat etmek gerekir. Aşağıdaki tabloda bu duruma ait bilgiler verilmektedir.

	MITSUBISHI A1SH	MITSUBISHI FX2N	OMRON CQM1H	OMRON C200HE	OMRON CS1	HITACHI H-200
Giriş – Çıkış rölesi (Maksimum, toplam)	256 adet X00-XFF Y00-YFF (X ve Y üst üste gelmez)	256 adet X00-X267 Y00-Y267 (X ve Y üst üste gelir)	Input 256 adet 0CH-15CH Output 256 adet 100CH-115CH	640 adet 0CH-29CH 300CH-309CH (Giriş ve çıkış üst üste gelmez)	5120 adet 0CH-319CH	256 adet X00-X1515 Y00-Y1515 (X ve Y üst üste gelmez)
Yardımcı röle	2048 adet M0-M999 L1000-L2047	500 adet M0-M499	2368 adet 16CH-89 CH 116CH-189CH	6464 adet 30CH-231CH 310CH-511CH	42304 adet 1200CH- 1499CH 3800CH- 6143CH	1984 adet R0-R7BF
100 ms zamanlayıcı	200 adet T0-T199 M1000-L2047	200 adet T0-T199	512 adet (Zamanlayıcı ve sayıcının toplamı) Zamanlayıcı TIM0-TIM511 Sayıcı CNT0- CNT511 Yüksek hızlı zamanlayıcı TIM0-TM15 (Sayıcı ve zamanlayıcı üst üste gelmez)	512 adet (Zamanlayıcı ve sayıcının toplamı) Zamanlayıcı TIM0-TIM511 Sayıcı CNT0- CNT511 (Zamanlayıcı ve sayıcı üst üste gelmez)	4096 adet T0-T4095	512 adet (Zamanlayıcı ve sayıcının toplamı) Zamanlayıcı TD0-TD255 Sayıcı CU0-CU511 (Zamanlayıcı ve sayıcı üst üste gelmez)
10 ms zamanlayıcı	56 adet T200-T255	46 adet T200-T245			4096 adet C0-C4095	
Sayıcı	256 adet C0-C255	200 adet C0-C199				
			CH Kanal anlamına gelir. Eğer “0” dan “15” e kadar iki basamaklı bir sayı her bir kanalın sonuna eklenirse, bu röle numarası olarak gösterilecektir.			

Şekil 1.9: Çeşitli marka PLC’lerin röle sınırlamaları

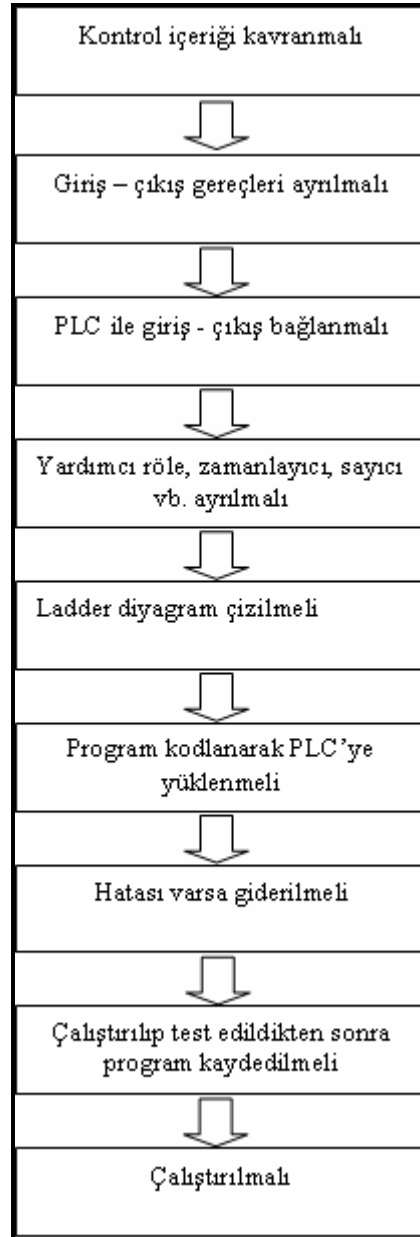
Bu kitaptaki uygulamalar ve örnekler MITSUBISHI - FX2N için hazırlanmıştır. FX2N için sınırlamalar aşağıda verilmiştir.

Giriş rölesi	X00-X07
Çıkış rölesi	Y00-Y07
Yardımcı röle	M00-M09 M10-M19 M20-M29.....
Zamanlayıcı	T00-T09 T10-T19 T20-T29.....
Sayıcı	C00-C09 C10-C19 C20-C29.....

1.3. Devre Dizaynı

1.3.1. Devre Dizaynının Akışı

Başlangıç seviyesinde olanlar PLC için bir program yazarken aşağıdaki işlemleri takip edebilir.



Şekil 1.10: Devre dizayn akışı

Kontrolün içeriği ve şartları kavranmalı, programın çözüm mantığı geliştirilmelidir.

Kullanılacak giriş ve çıkış eleman numaraları belirlenmelidir. Birbiriyle ilişkili giriş çıkışların numaraları da ilişkili seçilirse programın anlaşılabilirliği daha basit olacaktır. PLC ile giriş ve çıkış elemanları kablolaması yapılmalıdır.

Programda hangi elemanların kullanılacağına genellikle merdiven diyagram çizilirken karar verilir. Üst üste kullanımdan kaçınmak için tahsisat tablosu yapılmalıdır.

Devre, kontrol sırasına uygun olarak çizilmelidir. Bu durumda devrenin anlaşılması ve çizilmesi kolaylaşacaktır.

Merdiven diyagram üzerinde program yapılmasına kodlama (coding) denilmektedir.

Programın kontrol edilmesi, “debug” olarak adlandırılır. PLC ve yazılımı, yapılan programı kontrol etme fonksiyonuna sahiptir. Eğer programda bir hata varsa, hata mesajı ve hata kod numarası görüntülenmelidir. Program EPROM veya EEPROM içerisine kaydedilmelidir. Hatası giderilmiş program çalıştırılmalıdır.

1.3.2. Programlama Gereçleri

PLC için program iki farklı donanım ile yapılabilmektedir. Bunlar programlama konsolu ve kişisel bilgisayardır. Kişisel bilgisayar için özel yazılım gereklidir.

Ø Programlama konsolu ile programlama



Programlama konsolu, komut listesindeki komutları (Öğretici dil) kullanarak PLC’ye program yazmaya ve yüklemeye yarayan bir donanımdır. Üzerinde klavye ve ekranı vardır. Taşınabilir ve küçük olması avantajdır.

Ø Kişisel bilgisayar ile programlama



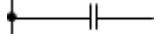
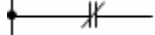
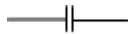
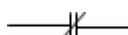
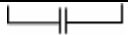
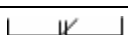
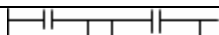
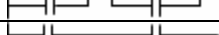
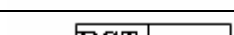
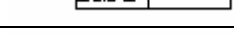
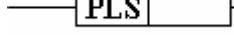
Bilgisayar ve özel bir yazılım kullanarak PLC programı yazmak mümkündür. Bu yöntem, PLC'ye program yazmak ve transfer etmek için kullanılan en yaygın yöntemdir. Fabrikalarda programları doğrudan değiştirmek için laptop kullanılmaktadır. Yukarıda bahsettiğimiz özel yazılımlardan bazıları aşağıda ifade edilmiştir.

MITSUBISHI	FX-PCS/WIN
	GX-Developer
OMRON	CX-Programmer
SIMENS	STEP7

1.4. PLC için Komut Dili (Öğretici Dil)

PLC üreticileri, kendi ürünleri için farklı programlama dili (öğretici dil - komut dili) kullanmaktadır. Bu farklı dil, PLC programlama metodunun farklılığından ya da PLC'nin fonksiyonlarının farklılığından kaynaklanmaktadır. Fakat temelde birbirlerine benzer.

Aşağıdaki tablo, MITSUBISHI ürünleri için, basit elektrik devrelerine göre komut kelimeleri hakkında açıklayıcı bilgi vermektedir.

Sembol adı	Fonksiyon	Devre, Eleman
 LD Load	İşlemi başlat (a kontak)	
 LDI Load inverse	İşlemi başlat (b kontak)	
 AND And	Seri bağlantı (a kontak)	
 ANI And Inverse	Seri bağlantı (b kontak)	
 OR Or	Paralel bağlantı (a kontak)	
 ORI Or Inverse	Paralel bağlantı (b kontak)	
 ANB And Block	Bloklar arasında seri bağlantı	
 ORB Or Block	Bloklar arasında paralel bağlantı	
 OUT Out	Bobine enerji verili	
 NOP Nop	İşlem yok	Boşluk
 SET Set	Bobine enerji verili (Mühürlemeli)	
 RST Reset	Bobin enerjisi kesik	
 PLS Pulse	Enerji verildiği anda bir seferlik yükselen sinyal	
 PLF Pulf	Enerji kesildiği anda bir seferlik yükselen sinyal	

MC Master control	Master Control Block'un başlaması	
MCR Master control reset	Master Control Block'un bitmesi	
MPS Push	Hafızanın set edilmesi	
MRD Read	Hafızanın okunması	
MPP Pop	Hafızanın okunması ve resetlenmesi	
END End	Programın tamamlanması	Programın tamamlanması

Şekil 1.11: Temel ardışık komutlar (öğretici dil komutları) (MITSUBISHI)

1.4.1.Temel Bilgi

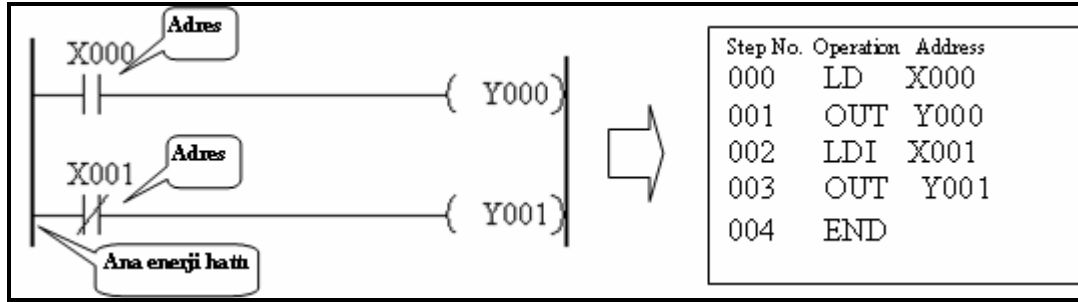
(1) LD

Load kelimesinin kısaltılmış halidir. Normalde açık olan “a kontak”ın ana enerji hattına bağlanmasını ifade eden komut kelimesidir.

(2) LDI

Load Inverse kelimelerinden alınmıştır. Normalde kapalı olan “b kontak”ın ana enerji hattına bağlanmasını ifade eden komut kelimesidir.

Aşağıda ardışık diyagram ve öğretici dil (komut kelimeleri) birlikte görülmektedir.



Ardışık diyagram

Öğretici dil

Şekil 1.12: LD ve LDI

(3) AND

Bu komut “VE” olarak açıklanır ve normalde açık olan bir kontağı (“a kontak”) başka bir kontağa seri olarak bağlamayı ifade eden komut kelimesidir.

(4) ANI

AND inverse komut kelimelerinin birleştirilmesi ile oluşturulmuştur. Normalde kapalı olan (“b kontak”) kontağı başka bir kontağa seri bağlamayı sağlayan komut kelimesidir.

(5) OR

Bu komut, “VEYA” olarak açıklanır ve normalde açık olan bir kontağı başka bir kantağı paralel bağlamayı sağlayan komut kelimesini ifade eder.

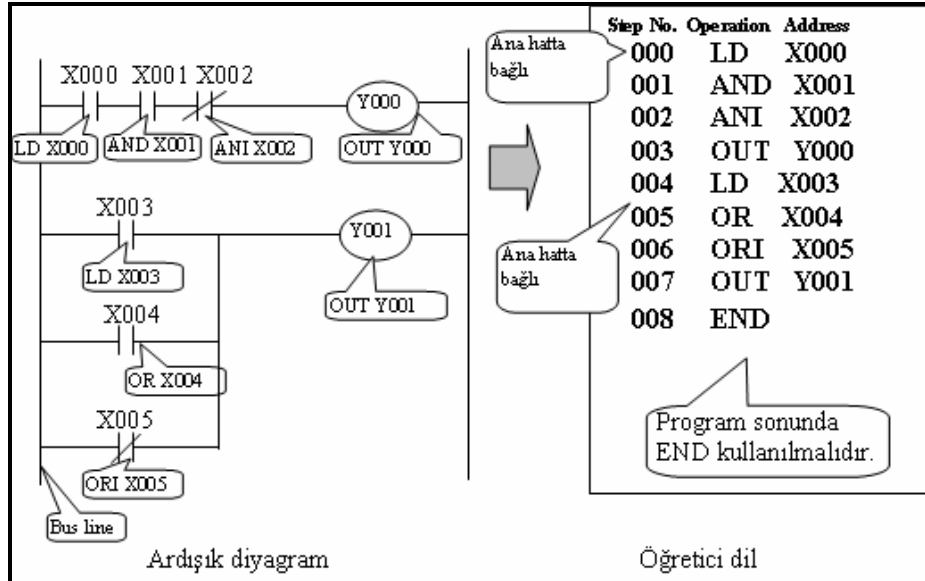
(6) ORI

OR inverse komut kelimelerinin birleştirilmesinden oluşturulmuştur. Normalde kapalı olan bir kantağı (“b kontak”) başka bir kantağı paralel bağlamayı ifade eden komut kelimesidir.

(7) OUT

Çıkış ya da bobini enerjilendirmeyi sağlayan komut kelimesidir.

Aşağıdaki ardışık diyagram ve öğretici dil, yukarıda anlatılan (1) – (7) arasındaki komutları göstermektedir.



Şekil 1.13: Temel komutların kullanımına örnek

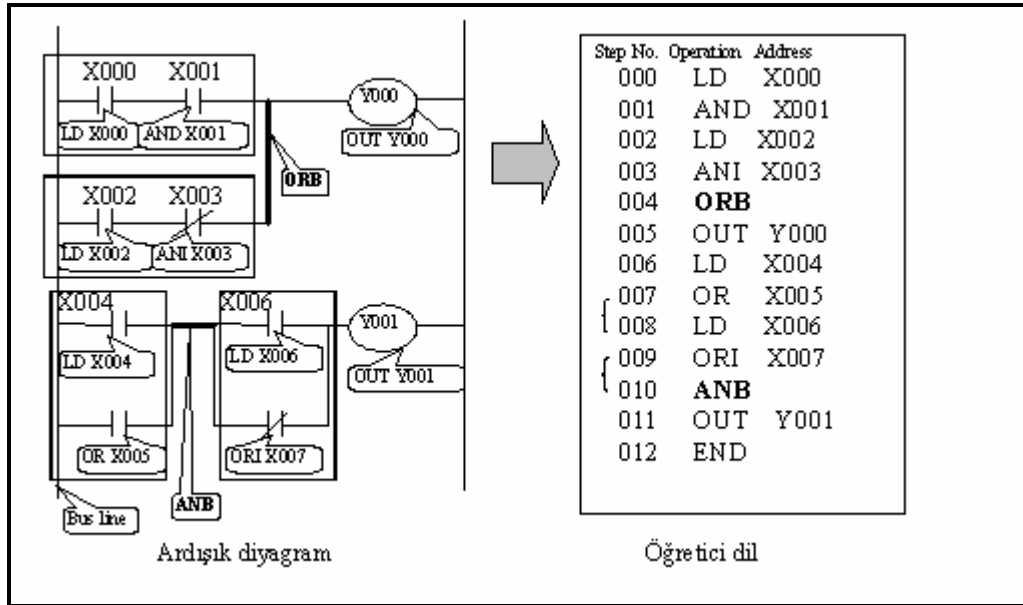
(8) ANB

Bu komut VE Blok (AND Block) olarak açıklanır. Blokları birbirine seri bağlamak için kullanılan komut kelimesidir.

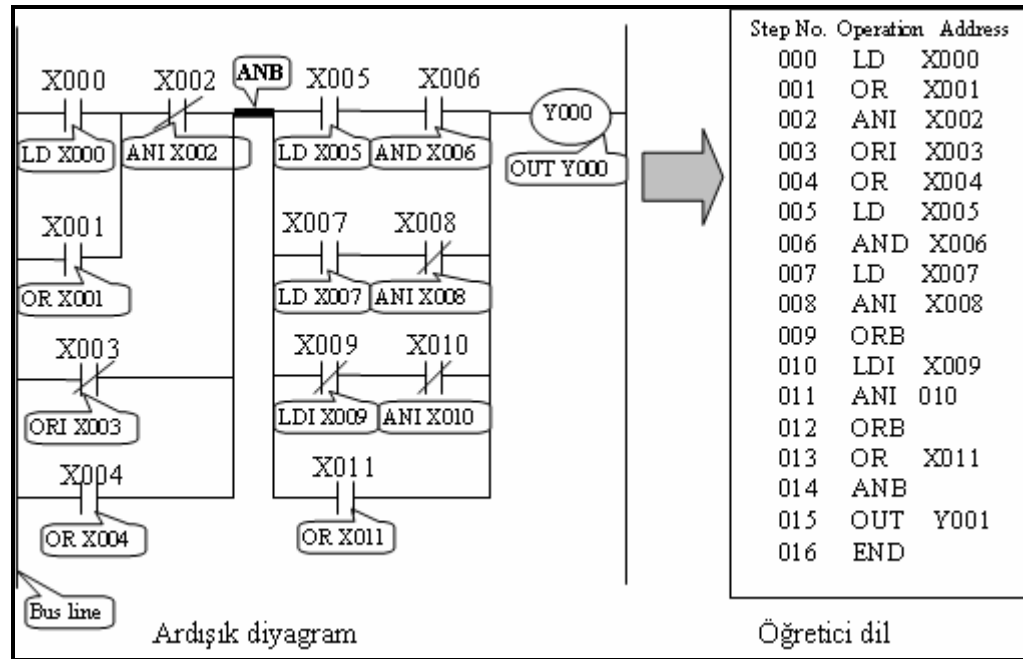
(9) ORB

Bu komut Veya Blok (OR Block) olarak açıklanır. Genellikle kendi arasında seri olarak bağlanmış olan kontakların blok halinde paralel bağlanması amacı ile kullanılır.

Aşağıdaki ardışık diyagram ve öğretici dil (1) – (9) arasındaki komutlara örnek olarak verilmiştir. İzleyen öğretici dilde, adım numarası 006 – 007 ve 008 – 009’da program paralel devreler halinde yapıldıktan sonra birbirlerine ANB komutu ile bağlanmalıdır.



Şekil 1.14: Çeşitli komutların kullanımına örnek (1)—(9)



Şekil 1.15: Çeşitli komutların kullanımına örnek (1)—(9)

X005 – X011 arasındaki program paralel olarak tamamlandıktan sonra ANB komutu kullanılarak önceden yapılan program ile birbirine seri olarak bağlanır.

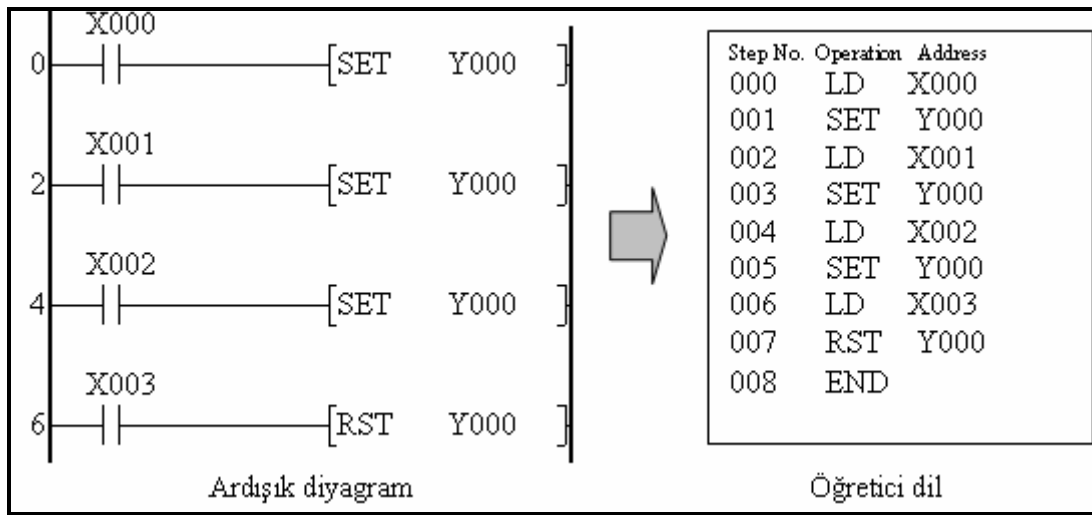
(10) SET

Bu komut bit aygıtını sürekli olarak “ON” konumuna getirir.

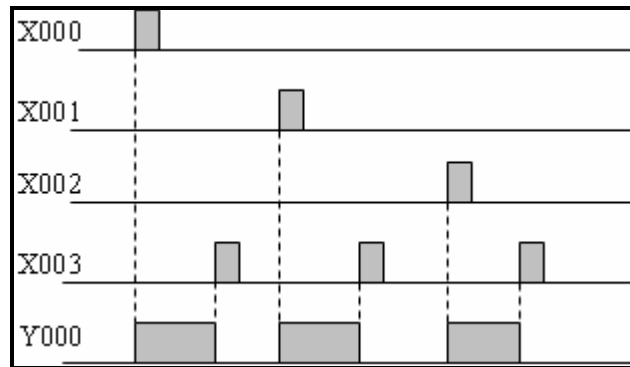
(11) RST

Bu komut bit aygıtını sürekli olarak “OFF” konumuna getirir. Aşağıdaki ardışık diyagram, öğretici dil ve zaman çizelgesi, SET ve RST komutlarının uygulamasını göstermektedir.

SET komutu, aynı çıkış aygıtı için birden fazla kullanılabilir. Karışık merdiven diyagramlarında SET komutu ve her bir giriş koşuluna uygun çıkış adresi, aynı çıkış adresi kullanılsa bile giriş koşullarının farklılığına bakılmaz.



Şekil 1.16: SET ve RST



Şekil 1.17 Zaman zaman çizelgesi

(12) MP

Bu komut, SPLC'nin o andaki aktif işleminin sonucunu hafızaya almaya yarar. MPS merdiven diyagramında bağlantı noktasına set edilir.

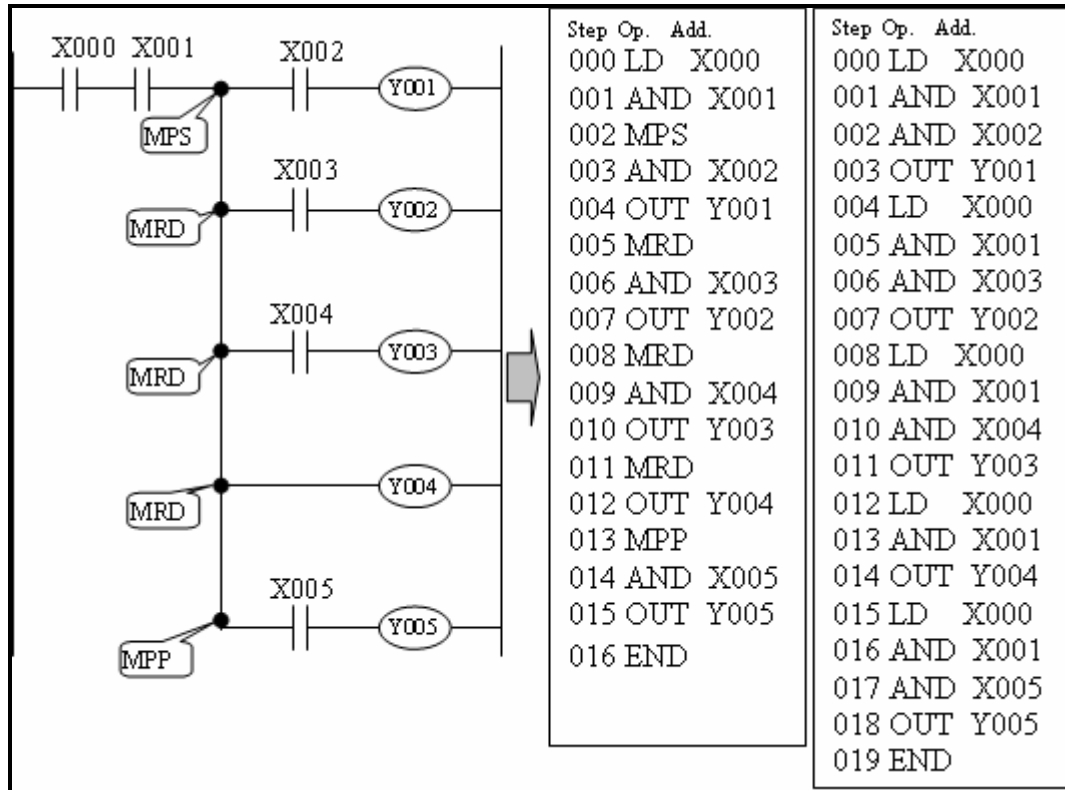
(13) MRD

Bu komut bir hafıza okuma komutudur ve kullanıldığı yerdeki aktif işlemin sonucunu okur. MRD komutu ile, MPS komutundan sonraki noktaya set edilir ve bir önce set edilen noktadaki veri okunur. Aynı zamanda bir sonraki bağlantı noktasını da bu nokta ile bağlantı kurmaya zorlar.

(14) MPP

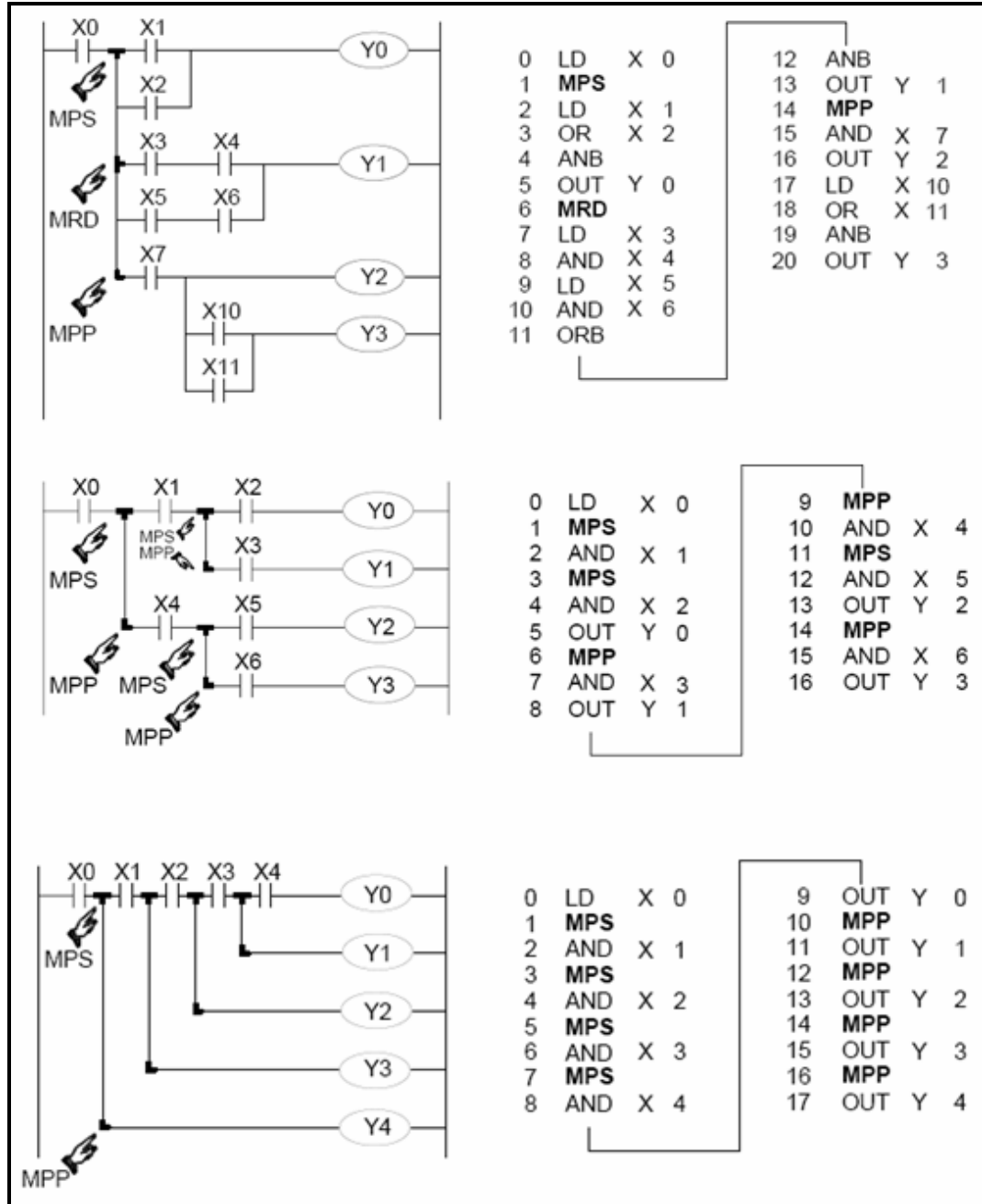
Bu komut kullanıldığı noktada, PLC'nin içerisindeki işlemlerin sonucunu okumaya ve reset etmeye yarar. MPP komutu, birden çok kola ayrılan devrede, en son çıkış noktasına set edilir. MPS komutu kullanılan bir devrede MPP komutu da son olarak kullanılmalıdır.

MPS, MRD ve MPP komutları, çıkış bobinlerini kontağın sol tarafına bağlamak için kullanılır. Eğer bunlar bu işlem içinde kullanılmazlarsa, sadece son kontağın sağ tarafı yapılabilir. Aşağıdaki uygulama, MPS, MRD ve MPP komutlarına örnek olarak verilmiştir. Aynı ardışık diyagram için iki adet öğretici dil kullanılarak yapılmış program verilmiştir. Bu programlardan sol taraftakinde MPS, MRD ve MPP komutları kullanılmış, sağ taraftakinde ise bu komutlar kullanılmamıştır. Bu iki program birbirinin aynı işlemi yapar. Fakat uzunlukları farklıdır.



Şekil 1.18: MPS, MRD ve MPP kullanımı

Şekil 1.13 gibi dallara ayrılan bir ardışık diyagramda, MPS, MRD, MPP gibi komutlar kullanılmazsa, “LD X000” ve “AND X001” komutları her bir dalın başına eklenmelidir. Bu durumda karışık bir devre yapılması gerektiği durumlarda, programın adım sayısı çok artacaktır. Buna bağlı olarak da programlama konsolu ile program yapmak çok daha zor bir hal alacaktır.



Şekil 1.19: Program konsolu

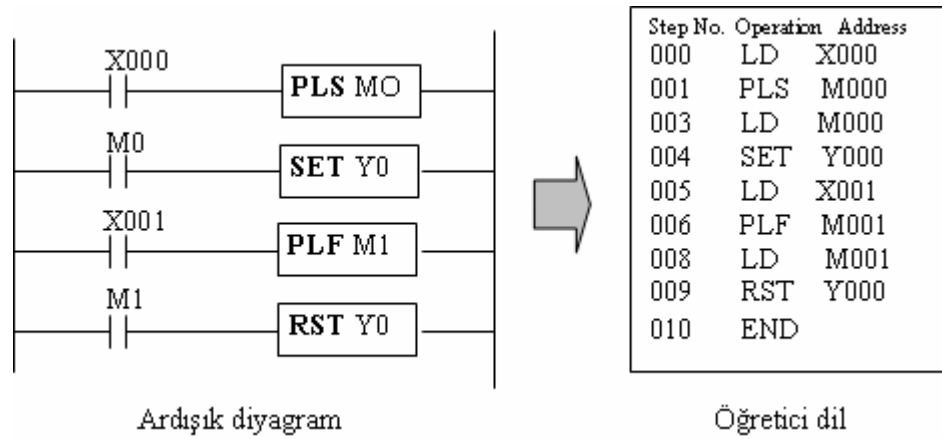
(15) PLS

Bu bir sinyal komutudur. Bu komut uygulandığında sadece ilk deęişim sinyali alınır.

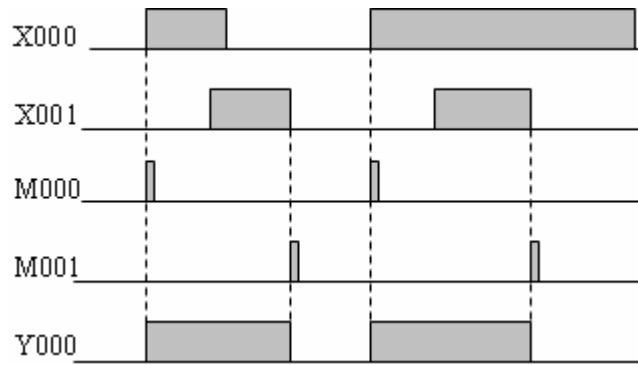
(16) PLF

Bu bir sinyal düşme komutudur. Bu komut uygulandığında sadece son deęişim sinyali alınır.

Aşağıdaki ardışık diyagram ve öğretici dil PLS ve PLF komutlarını göstermektedir.



Şekil 1.20: PLS ve PLF

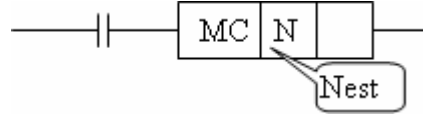


Şekil 1.21: Zaman grafięi

(17) MC

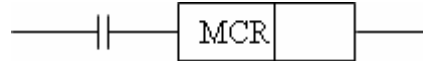
MC (Master Control), yönetici kontrol bloğunun başlangıcını ifade eder (Program adım 3).

Yönetici kontrol bloğunun grup seviyesi, (Nest) “0” dan “7” ye kadardır. (Toplam sekiz adet)

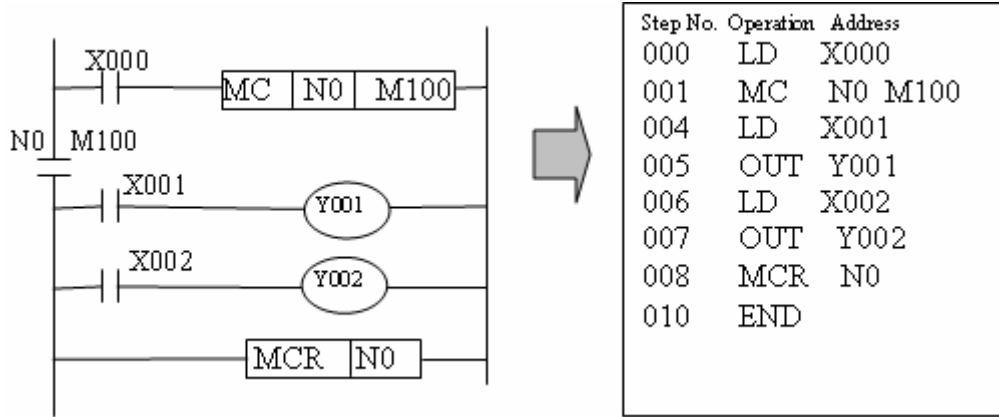


(18) MCR

MCR (Master Control Reset), yönetici kontrol bloğunun bitişini ifade eder (Program adım 2).



Aşağıdaki ardışık diyagram ve öğretici dil, MC ve MCR komutlarının kullanımına örnek olarak verilmiştir.



Şekil 1.22: MC ve MCR

X0 butonuna basıldığında (Master Control “ON” olduğunda) MC ve MCR komutları arasında kalan işlemler gerçekleştirilir. X0 butonu serbest bırakıldığında ise, bu iki komut arasında kalan işlemler gerçekleştirilmeden atlanır.

- Ø MC komutu, yardımcı röle (M) ve çıkış rölesi (Y) için etkilidir.
- Ø LD komutu, MC komutundan sonra uygulanır.
- Ø MC komutu uygulandığı anda, ana elektrik hattı (bus line) MC komutunun bir altına kayar. MCR komutu ile de orijinal noktasına döner.
- Ø Grup (Yuva) seviye numarası küçük numaradan büyük numaraya doğru (N0, N1..., N7) kullanılmalıdır.
- Ø MCR komutu kullanıldığında (Nest seviyesi ile birlikte), kullanıldığı grup seviyesi ve üzerindeki seviyeleri resetler. En sonunda N0 nest seviyesi de resetlenmelidir.

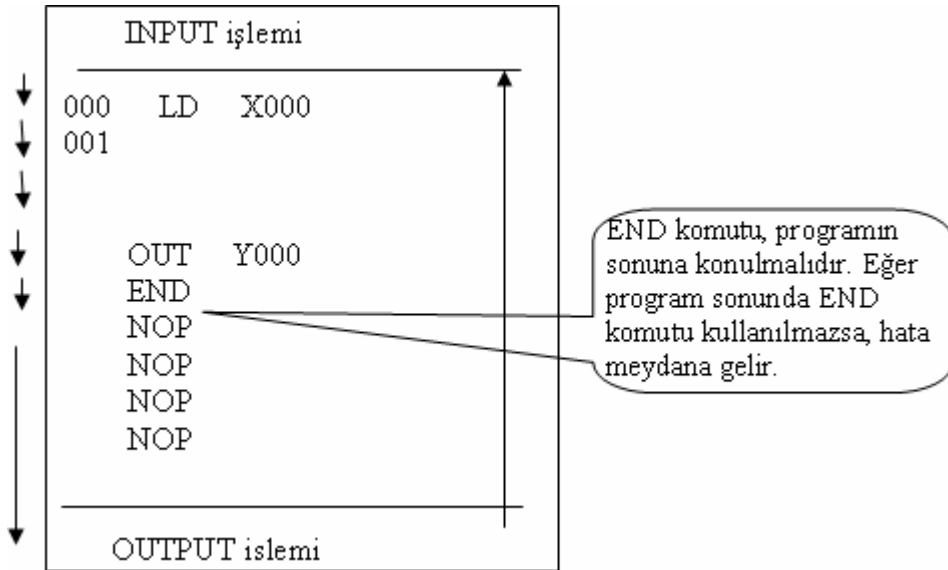
(19) NOP

İşlem yok ya da önemsiz satır anlamında bir komuttur. PLC içerisindeki program silindiğinde, tüm komutlar NOP şeklinde olur. Program içerisinde de NOP komutu kullanılabilir. Bu durumda program işletilirken bu satırlar atlatılır ve sonraki komutlar çalıştırılır. Fakat gerekli olmadığı durumlarda, program adımının daha da uzamasına meydan vermemek için bu satırların silinmesi daha uygundur.

(20) END

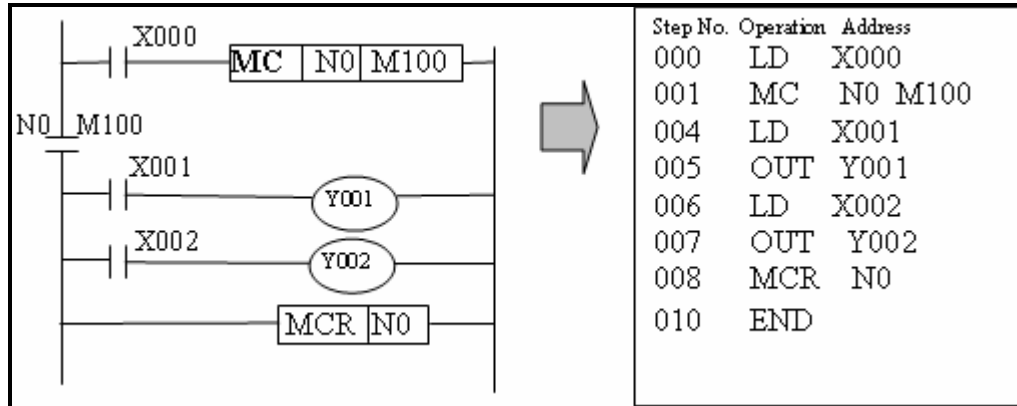
Programın en son satırına END komutunun yazılması durumunda PLC, bundan sonra yazılan komutları çalıştırmaz.

END komutu, test çalışmalarında programın çeşitli bölümlerine konularak programın parça parça kontrolü yapılabilir.



(21) SP

SP (Space, boşluk) boşluk tuşu, sayıcı ve zamanlayıcıya sabit sayı girmede, MC komutunda vb. yerlerde kullanılır. Bu bir komut olarak algılanmamalıdır.



Şekil 1.23: MC komutu kullanımı

Aşağıdaki çizelge, fonksiyon – komut sınıflandırmasını gösterir.

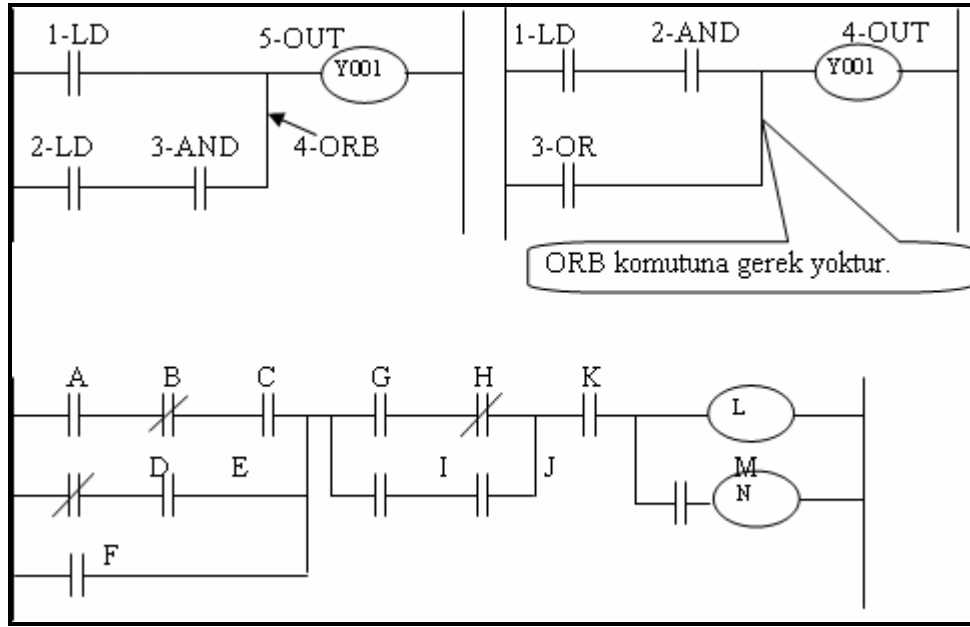
Fonksiyon (Görev)	Komut
Komut, kontak olarak ele alınır.	LD AND OR LDI ANI ORI
Komut, bobin olarak ele alınır.	OUT PLS SET PLF RST
Komut, bağlantı olarak ele alınır.	ANB MPS MPP ORB MRD
Diğer tür komutlar.	MC NOP MCR END

1.4.2. Programlama için Uyarıcı İfadeler

(1) Programın uygulama sırası

Kontak ve adım

Programlama sırası



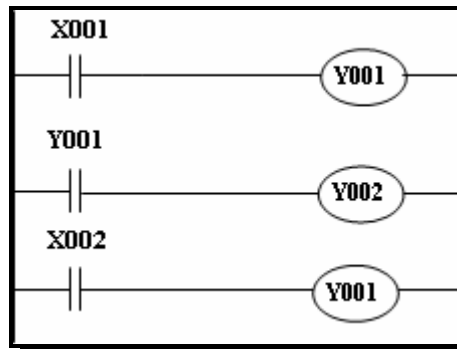
Şekil 1.24: Programlama sırası

Yukarıdaki program soldan sağa ve yukarıdan aşağıya doğru çalıştırılır.

(2) Aynı çıkış rölesinin birden fazla kullanılması

PLC programında aynı bobin birden fazla yerde çalıştırılacaksa, en son kullanılan kontak önceliğe sahip olur (Bobin kontakları bobinin en sondaki durumuna göre açık ya da kapalı olur).

Girişler X001=ON X002=OFF

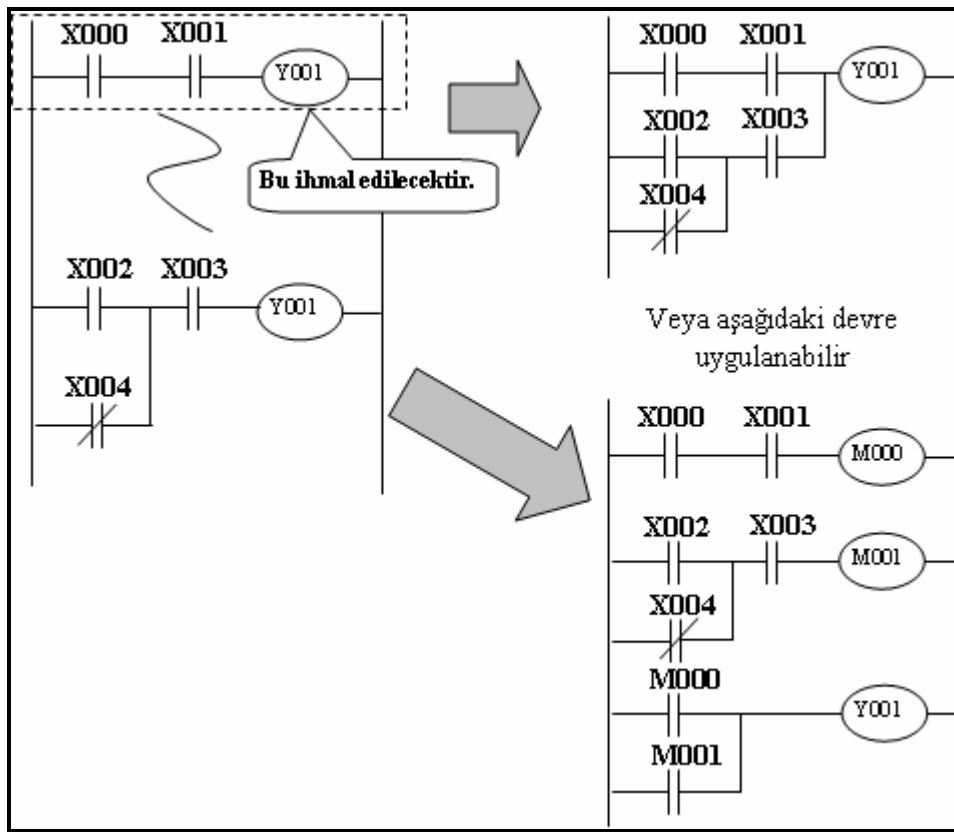


Çıkış durumu Y001=OFF Y002=ON

Yukarıdaki şekilde, aynı devre içerisinde Y001 rölesi iki kez kullanılmıştır. X1 “ON” olduğunda ilk Y001 bobini de “ON” olur. Dolayısıyla Y002’de “ON” olur. X2 kontağı “OFF” olduğu için ikinci kez kullanılan Y001 bobini de “OFF” durumuna gelecektir. Sonuç olarak Y001 bobininin son hali “OFF” olacaktır.

Aynı çıkış rölesi için hazırlık

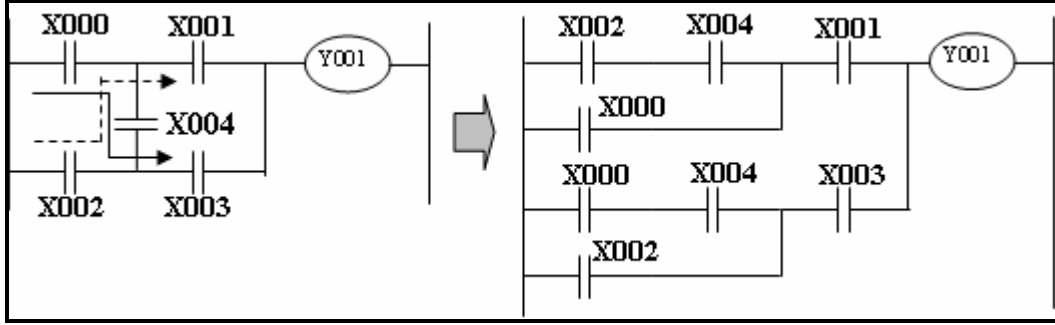
Program içerisinde aynı bobinin birden fazla kullanımı yasaklanmamıştır. Ancak bu kullanım ile işlemler karmaşık bir hal alacaktır. Bu sebeple aynı çıkış rölesini birden fazla kullanmak yerine, aşağıdaki gibi bir kullanım daha uygun olacaktır.



Şekil 1.25: Çift kontak kullanım kuralı

(3) Programlama için istenmeyen durum

Köprü devresi

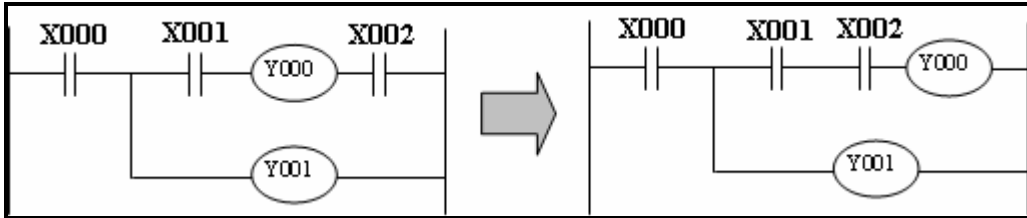


Şekil 1.26 : Programlama için istenmeyen durum

Programlamada, soldaki devrede olduğu gibi X004 kontağının bağlantı şekli mümkün değildir. Böyle bir devrenin kurulması gerektiğinde sağdaki devrede olduğu gibi bir bağlantı yapılabilir. Bu iki devre birbirisinin aynısıdır fakat soldaki devre programlama olarak hatalı bir devredir. Sağdaki devre, X001 ve X003'ü dikkate almazsak, diğer kontakların yalnız ya da blok halinde paralel bağlantısından meydana gelmiştir.

Ø Çıkış rölesinin bağlantısı

Merdiven diyagram çizilirken bobinin sağ tarafı doğrudan hatta bağlanmalıdır. Çıkış rölesinin sağına başka bir bobin ya da kontak çizilemez.



Şekil 1.27 : Bobin bağlantısı

UYGULAMA FAALİYETLERİ

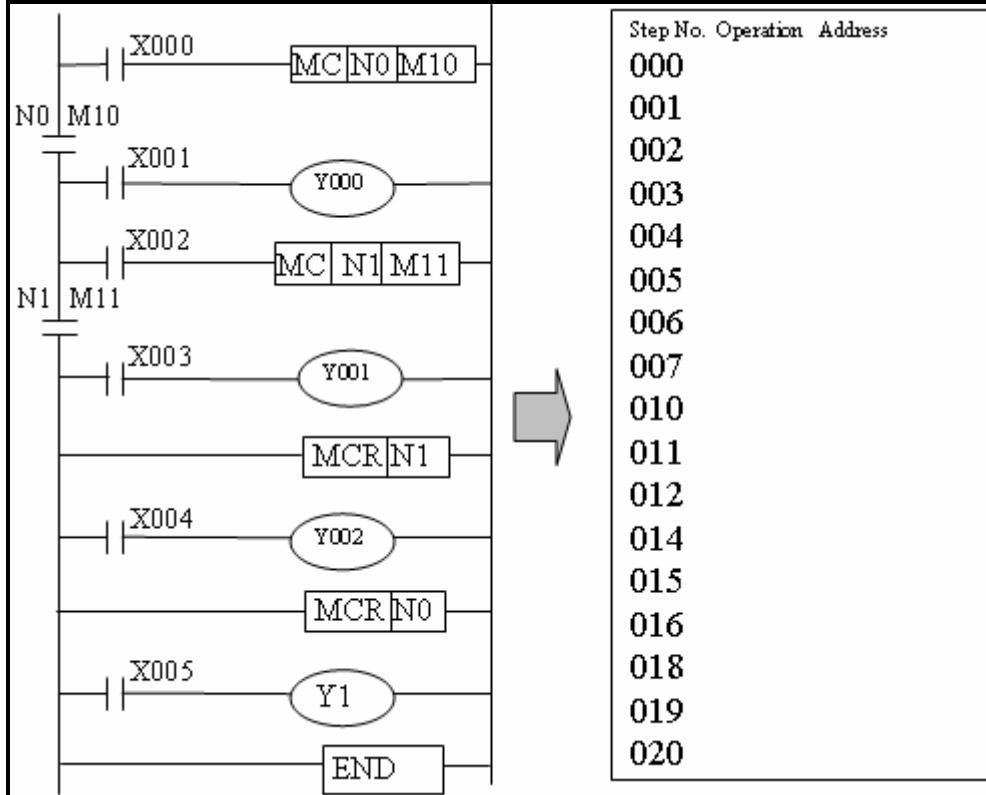
Aşağıdaki işlem basamaklarına göre uygulama faaliyetlerini yapınız.

İşlem Basamakları	Öneriler
Ø Öncelikle verilen problemde sizden istenenleri kavrayınız.	Ø Çalışma şartlarını kafanızda canlandırabilmeniz gerekir.
Ø Hangi komut ya da komutları kullanarak nasıl bir çözüm yolu izleyeceğinize karar veriniz.	Ø Size özel bir çözüm yolu geliştirebilirsiniz.
Ø Çözümünüzü öncelikle bir kâğıt üzerine çizerek kontrol ediniz.	Ø Önce kâğıt üzerinde çalışmak problem çözme yeteneğinizi geliştirecektir. Ø Enerji verildiği anda ne olması isteniyor? Sorusundan başlayınız.
Ø Giriş ve çıkış elemanları için gerekli kablolamayı yapınız.	Ø PLC enerjisiz <u>olmalıdır</u> .
Ø Merdiven diyagramını bilgisayar ortamında kurallara uygun çiziniz.	Ø Programınızı kaydetmeyi unutmayınız.
Ø Programınızı derleyerek PLC'ye yüklemeye hazır hale getiriniz.	
Ø PLC'ye programınızı yükleyiniz.	Ø PLC enerjili ve <u>STOP konumunda olmalıdır</u> .
Ø Giriş şartlarını sağlayarak programın çalışmasını kontrol ediniz.	Ø PLC besleme gerilimi 220 V ve giriş-çıkış ünitelerinin sinyal değeri 24 V olarak uygulama yapılmalıdır.

Uygulama- 1

Bu uygulamada merdiven diyagram ile yazılan programın öğretici dile çevrilmesine yönelik çalışma yapılacaktır.

Aşağıdaki ardışık diyagrama göre öğretici dilde program yazınız ve çalışmasını açıklayınız.



Şekil 1.28 : PLC programı

Uygulanama- 2

Bu uygulamada verilen şartları sağlayan program yazarak dc motor kontrolüne yönelik uygulama yapılacaktır.

PLC girişine bağlı butonlarla PLC'nin çıkışına bağlı olan bir dc motorun çalıştırılıp durdurulmasını saylayan programın merdiven diyagramıyla yapılmasını öğreneceksiniz.

Çalışma şartları:

- Enerji verildiğinde motor çalışmayacak
- X000 butonuna basıldığında motor çalışmaya başlayacak.
- X000 butonu bırakılsa bile çalışma devam edecek.
- X001 butonuna basıldığında motor duracak.
- Her iki buton, aynı anda basılı tutulursa motor çalışmayacak.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

- Algılayıcılar PLC’de nereye bağlanmalıdır?
A) Çıkış ünitesi
B) Giriş ünitesi
C) Hafıza
D) CPU
- Röleli çıkışlı **FX2N** PLC’nin çıkış uçlarından çekilebilecek en büyük akım değeri ne kadardır?
A) 2 A
B) 5 A
C) 10 A
D) 0,2 A
- Hangisi PLC’de bir çıkış şekli değildir?
A) Triyaklı
B) Transistörlü
C) Röleli
D) Dirençli
- Hangisi paralel iki açık kontağı ifade etmek için kullanılır?
A) AND
B) OR
C) ORI
D) ANI
- Hangisi seri iki bloğu birleştirmek için kullanılır?
A) MPS
B) SET
C) ANB
D) ANI
- Bit aygıtını sürekli olarak ‘ON’ durumunda tutan komut hangisidir?
A) SP
B) SET
C) MPS
D) LD

7. Paralel iki bloğu ifade etmek için hangisi kullanılmalıdır?
A) LD
B) NOP
C) END
D) ORB
8. Program sonunda kullanılması gereken komut hangisidir?
A) END
B) SET
C) MPS
D) LD
9. Y0001 çıkışını 'ON' yapmak için hangi komut kullanılmalıdır?
A) RST Y0001
B) LD Y0001
C) OUT Y0001
D) ORB Y0001

Değerlendirme

Cevaplarınızı cevap anahtarı ile karşılaştırınız. Doğru cevap sayınızı belirleyerek kendinizi değerlendiriniz. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt yaşadığınız sorular için ilgili faaliyete geri dönerek tekrar inceleyiniz.

ÖĞRENME FAALİYETİ- 2

AMAÇ

PLC’de merdiven diyagramı ile programlamayı doğru olarak yapabileceksiniz.

ARAŞTIRMA

Bu öğrenme faaliyetinden önce aşağıdaki hazırlıkları yapmalısınız.

- Ø Sayı sistemleri ve kontrol alanlarında kullanımına yönelik araştırma yapınız.
- Ø Sayıcı ve zamanlayıcıların kullanım alanlarını araştırınız.

2. İLERİ SEVİYE PLC KULLANIMI

2.1. Sayısal Veri Kavramı

2.1.1. İkilik Sayı Sistemi

Günlük hayatta varlıkların sayısını ifade etmek için onluk sayı sistemi kullanılmaktadır. Onluk sayı sisteminde 0,1,2,3,4,5,6,7,8,9 rakamları kullanılır.

Sayısal devrelerde sinyal ya da anahtarların durumları ikilik sayı sistemiyle açıklanabilmektedir. İkilik sayı sisteminde varlıkların sayısı yalnızca 0 ve 1 rakamları kullanılarak ifade edilir. Bu rakamların her birine “bit” denir.

Pozitif mantığa göre bir elektrik devresinde sinyalin varlığı veya bir anahtarın kapalı olması “1”, sinyalin yokluğu veya anahtarın açık olması ise “0” ile gösterilir. Benzer şekilde devreye enerji verip kesmek ya da PLC içerisindeki röleleri ya da hafızaları aktif ya da pasif yapmak için de ikilik sayı sistemi kullanılır. Devreye enerji verilmesi ya da herhangi bir elemanın aktif olması “1” , devre enerjisinin olmaması ya da herhangi bir elemanın pasif durumda olması da “0” sinyali ile gösterilir.

2.1.2. Sekizlik ve Onaltılık Sayı Sistemi

Octal (Sekizlik) Sayı Sistemi

Sekizlik sayı sisteminde varlıkların sayısı 0,1,2,3,4,5,6,7 rakamları kullanılarak ifade edilir. İkilik sistemdeki sayıyı en sağdan itibaren üçerli gruplara bölerek ve 0 ile 7 arasındaki rakamları kullanarak sekizlik sayı sistemine dönüştürebiliriz.

Hexadecimal (On altılık) sayı sistemi

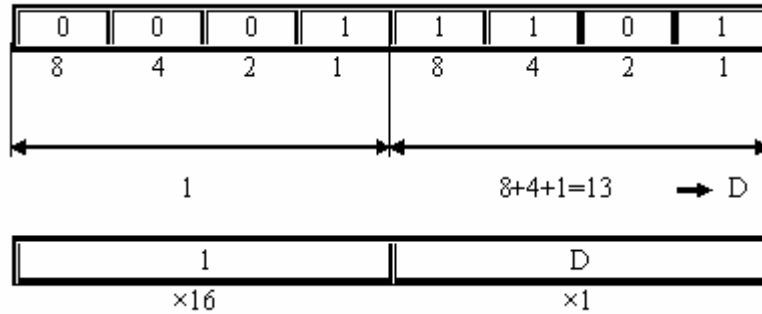
On altılık sayı sisteminde varlıkların sayısı 0.1.2.3.4.5.6.7.8.9,A,B,C,D,E,F karakterleriyle ifade edilir. On altılık sayı sisteminde her bir karakter dört bit ile ifade edilir. İkilik sayı, dörderli gruplara bölünerek ve her bir dördü grubun on altılık sayı sistemindeki karşılığı bulunarak dönüşüm işlemi yapılabilir.

Sayıların hangi sistemde olduğunu belirtmek için bazen sayının sağına indis şeklinde sayı sistemi yazılır. On altılık sayı sistemi programlarda çok sık kullanılmaktadır. Indis şeklinde yazmak bazen sorun yaratabilir. Bu nedenle on altılık sayılar genellikle sayının sonuna ya da başına “H” harfi eklenerek belirtilir. Örneğin; “H10” ya da “10H” (decimal karşılığı 16), “H2A” ya da “2AH” (decimal karşılığı 42).

Y007	Y006	Y005	Y004	Y003	Y002	Y001	Y000
0	0	0	1	1	1	0	1
128	64	32	16	8	4	2	1

Yukarıda, PLC’nin çıkış ünitesinde her bir rölenin sinyal durumu görülmektedir. Sekiz bit genişliğindeki bu sayının onluk sayı sistemindeki karşılığı $16+8+4+1=29$ ’dur.

Aynı sinyal durumunu on altılık sayı sisteminde ifade etmek için sağdan itibaren her bir dört bit bir karakter ile gösterilir.



“Y007 – Y000” arası H1D sayısı ile gösterilirse Y4, Y3, Y2 ve Y0 çıkışlarının aktif olduğunu ve diğerlerinin de pasif olduğunu anlamak gerekir. Bu durum ikilik sayı sisteminde 00011101 şeklinde gösterilir. Aynı sayı onluk sayı sisteminde 29’ a eşittir.

$$H1D_{\text{(Hexadecimal)}} = 00011101_{\text{(Binary)}} = 29_{\text{(Decimal)}}$$

2.1.3. BCD Sayı Sistemi

BCD, ikilik kodlanmış onluk sayı sistemidir (Binary Coded Decimal). Onluk sayı sistemindeki bir sayının her bir karakterinin ikilik sayı sisteminde dört bit ile gösterilmesi durumudur. Onluk sayı sisteminde en büyük karakter “9” olduğundan her bir karakter dört bit ile ifade edilir.

Aşağıda 15_{10} sayısının BCD karşılığının 00010101_{BCD} olduğu gösterilmiştir.

Y007	Y006	Y005	Y004	Y003	Y002	Y001	Y000
0	0	0	1	0	1	0	1
80	40	20	10	8	4	2	1

2.1.4. Veri Sistemi

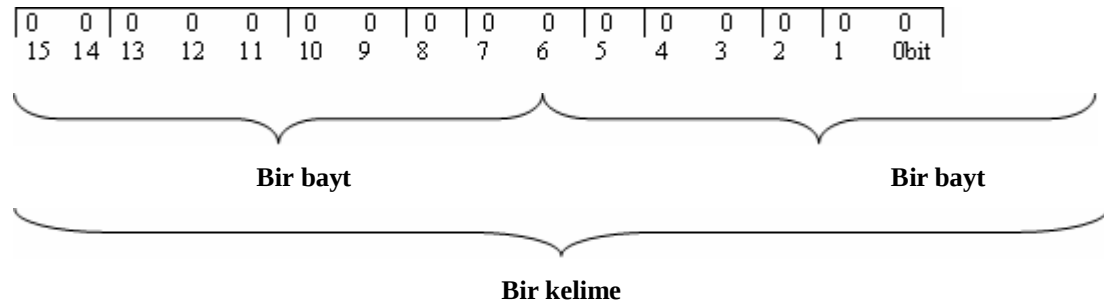
Binary, octal, hexadecimal ve BCD sayı sistemleri şu ana kadar açıklandı.

Aşağıdaki tabloda bazı sayıların farklı sayı sistemindeki karşılıkları verilmiştir.

Decimal	BCD		Hexadecimal	Binary		Octal
0	0000	0000	00	0000	0000	0
1	0000	0001	01	0000	0001	1
2	0000	0010	02	0000	0010	2
3	0000	0011	03	0000	0011	3
4	0000	0100	04	0000	0100	4
5	0000	0101	05	0000	0101	5
6	0000	0110	06	0000	0110	6
7	0000	0111	07	0000	0111	7
8	0000	1000	08	0000	1000	10
9	0000	1001	09	0000	1001	11
10	0001	0000	0A	0000	1010	12
11	0001	0001	0B	0000	1011	13
12	0001	0010	0C	0000	1100	14
13	0001	0011	0D	0000	1101	15
14	0001	0100	0E	0000	1110	16
15	0001	0101	0F	0000	1111	17
16	0001	0110	10	0001	0000	20
:	:	:	:	:	:	:
99	1001	1001	63	0110	0011	143

Şekil 1.29: Bazı sayıların farklı sayı sistemlerinde karşılıkları.

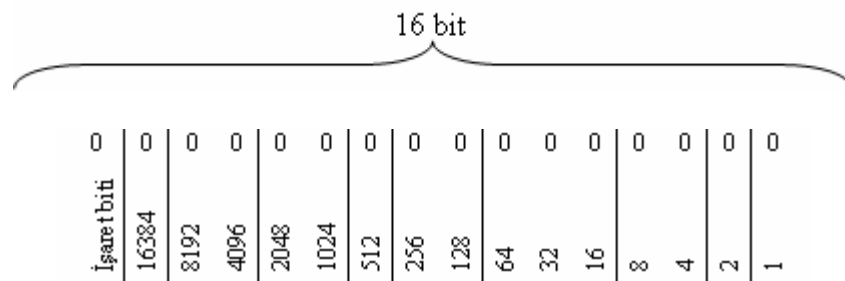
İkilik sayı sisteminin her bir rakamı “1”(ON) ya da “0”(OFF) ile gösterilir. 1 veya 0 ile gösterilen en küçük birime “**bit**” denir.



Sıfır ile yedinci bit arasındaki sekiz bitten oluşan veriye (one byte) bir “**bayt**” denir. On altı bitten meydana gelen veriye ise (one word) bir “**kelime**” adı verilir.

2.1.5. 16 ve 32 Bitlik Verinin Yönetimi

Aşağıdaki şekil 16 bitlik verinin durumunu göstermektedir. İşaret biti önemli özelliğe sahiptir. İşaret bitinin “0” olması, sayının “pozitif”, “1” olması ise “negatif” olduğu anlamına gelir.



PLC içinde “-” negatif sayılar binary olarak ve sayının ikiye tamamlayanı alınarak gösterilir. Örneğin, D10 veri kaydedicisinin (Data register) içeriği negatif bir sayı ise, bu veri kaydedici içerisinde sayının ikiye tamamlayanı kullanılır.

Negatif bir sayının ikiye tamamlayıcı her bir bitin tersinin alınıp sonuca “1” eklenmesi ile elde edilir. Aynı şekilde ikiye tamamlanmış bir sayının gerçek karşılığını bulmak için sayının tersi alınarak “1” ilave edilir.

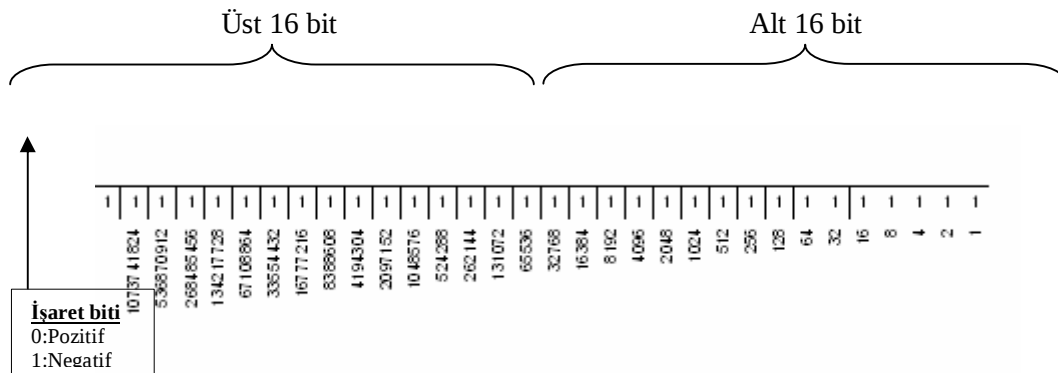
- 3'ün ikiye tamamlayanı ve gerçek sayının gösterimi aşağıdaki gibidir.

İkiye tamamlayan	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
Verinin tersinin alınması	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Bir eklenmesi	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Gerçek sayı	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Binary																Decimal
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	32,766
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	32,767
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	-2
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-32,767
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-32,768

Şekil 1.30: İkilik ve onluk sayı sistemindeki bazı pozitif ve negatif sayılar

65535'ten daha büyük sayıları ifade etmek için 16 bitlik veri genişliği yeterli değildir. -2147483648 ile +2147483647 arasındaki sayılar 32 bitlik veri ile ifade edilebilir. Bu alan genellikle ihtiyacı karşılamaktadır. 32 bitlik veri üst 16 bit ve alt 16 bitlik verilerin birleşmesinden oluşur. PLC'de 32 bitlik veri kullanarak işlem yapmak mümkündür.



Şekil 1.31: İkilik ve onluk sayı sistemindeki bazı pozitif ve negatif sayılar

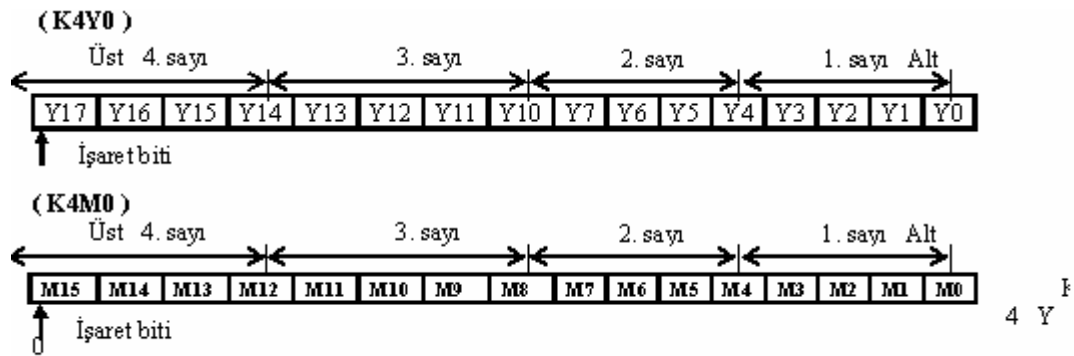
2.2. Aygıt

Sıralı kontrol teknolojisini anlamada aygıt çok önemlidir. Tüm veriler aygıt kullanılarak açıklanmalıdır. PLC'de bit ve kelime aygıtı olmak üzere iki tür aygıt vardır. Bit aygıtı sadece "1" ve "0" sinyallerini gösterebilir. (Örneğin, X, Y, M, S birer bit aygıtıdır.) Kelime aygıtı ise sayısal değerleri gösterir. (Örneğin T, C, D birer kelime aygıtıdır.)

2.2.1. Bit Aygıtı (Bit Device)

Bit aygıtı sadece “1” ve “0” bilgisini göstermek için kullanılır. Örneğin giriş rölesi (X), çıkış rölesi (Y), yardımcı röle (M), durum (S) vb. bit aygıtı sadece “1, kapalı” ya da “0, açık” bilgilerini taşımasına rağmen, bu aygıtların birlikte kullanımıyla sayısal bilgiler de taşınabilir.

Dörtlü gruplar (ünite) halindeki bit aygıtlarını yan yana kullanarak 32 bitlik veri işlemek mümkündür.



K: Basamak sayısının açıkça belirtilmesi gereklidir ve basamak sayısının önüne K harfi konur.

4: Basamak sayısı. (4 basamak ayrıldığı durumda)

16 bitlik işlemlerde basamak (digit) sayısı K1, K2, K3, K4 şeklinde belirtilir. 32 bitlik işlemlerde ise basamak sayısı K1,K8 şeklinde belirtilir. Burada bir basamak 4 bit anlamına gelmektedir.

Y: Çıkış aygıtını belirtir.

0: Belirtilen en küçük bit numarası.Y0’dan itibaren basamaklandırılacak anlamındadır.

Kısıtlamanın olmadığı durumlarda en küçük bit isteğe bağlı olarak belirtilebilir. (Örneğin, K4M11) Fakat bu numarayı 0 ya da 10’un katları olarak belirtmek daha iyidir. Burada K4, 4 basamaklı bir üniteyi gösterir. K2, 2 basamaklı bir üniteye karşılık gelir.

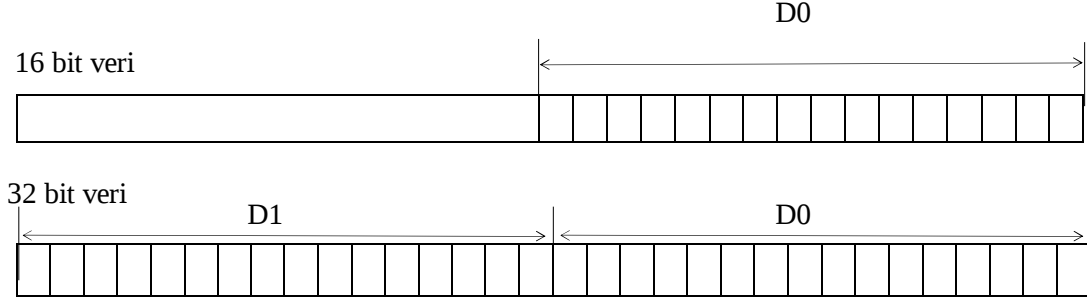
Taşıma, karşılaştırma, ekleme, çıkarma, çarpma ve bölme gibi veri işlemleri yapılırken 32 veya 16 bitlik işlem yapıldığına dikkat etmek gerekir. 32 bitlik işlem sonucunu 16 bitlik alana yazamazsınız.

2.2.2. Kelime Aygıtı (Word Device)

Kelime aygıtı 16 ya da 32 bitlik veriyi kaydetmek üzere tasarlanmıştır. PLC’de çeşitli kelime aygıtlarından bahsedilebilir.

(1) Veri kaydedicisi (Data register) (D)

Bu veri kaydedicisi ile 16 bitlik veri ya da iki adet 16 bitlik veri kaydedicisinin birlikte kullanılması ile 32 bitlik veri kontrol edilebilir.



Eğer düşük numaralı veri kaydedicisi (Örneğin D0) 32 bitlik data için ayarlanır ise, bir üst numaralı veri kaydedicisi otomatik olarak ikinci 16 bite ayrılır.

(2) Zaman kaydedicisinin güncel değeri (T)

Zamanlayıcıya, güncel değer olarak sayısal bir değer verilebilir. Bu aygıt 16 bitlik bir kaydedicidir ve 0 ile 32767 arasındaki değerleri alabilir. Zamanlayıcıya negatif değer girilemez.

Zamanlayıcının birimi ayarlanan zamanlayıcı numarasına göre 100 ms, 10 ms ve 1ms olarak değişiklik gösterir. Bu konuda daha geniş bilgi verilecektir.

(3) Sayıcının güncel değeri (C)

Sayıcıya güncel değer olarak sayısal bir veri girilebilir. Ayar numarası değiştirilerek yukarı sayıcı veya aşağı sayıcı elde edilir.

(Yukarı sayıcı için)

Üst bitteki “0” nedeniyle bu kaydedici 16 bitliktir ve 0 ile 32767 arasındaki sayıları değer olarak alabilir. (Eğer bu sayı negatif olarak girilirse, sayıcı çalışmaz.)

(Aşağı sayıcı için)

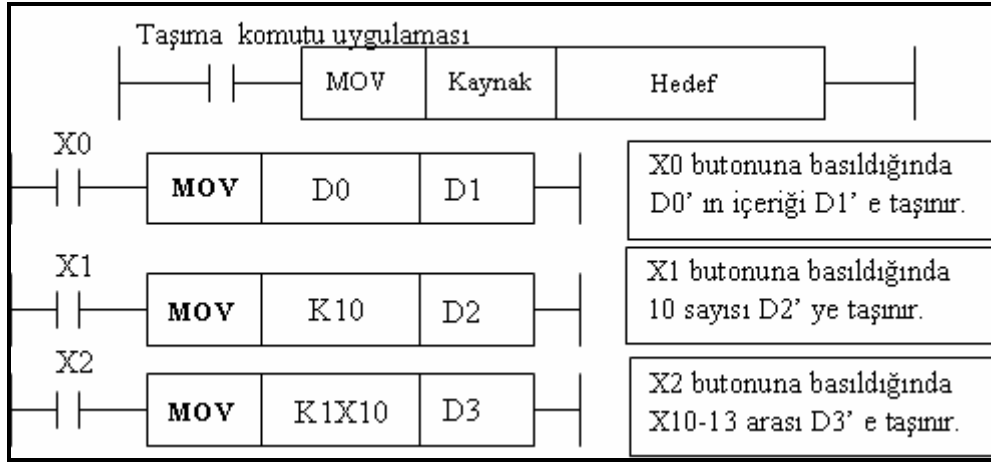
Bu kaydedici 32 bitlik bir kaydedicidir ve negatif sayıları da alabilir. Alabileceği değerler; -2147483648 ile +2147483647 arasındaki sayılardır.

2.2.3. Veri Taşıma

Sayısal değer, veri kaydedicide (D), X,Y,M,S gibi birleştirilmiş bit aygıtlarında, zamanlayıcı ya da sayıcıda geçerli değer olarak tutulur. Böyle bir veri bir kaydediciden başka bir kaydediciye taşınabilir. Bu işlem “veri taşıma” (data transfer) olarak adlandırılır. Veri taşıma işlemi “MOV” komutu ile yapılır.

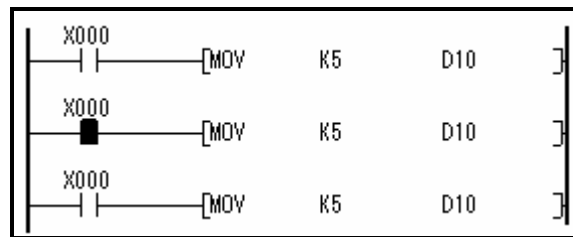
Taşıma işlemi yapılırken hangi verinin nereye taşınacağı uygun şekilde belirtilmelidir. Taşıma işleminde bit genişliğine de dikkat edilmelidir. 32 bitlik veri 16 bitlik alana taşınamaz.

Taşıma komutuyla birlikte onluk sayı sisteminde veri tanımlamak mümkündür. (Örneğin onluk sistemde 25 sayısını taşımak için kaynak, K25 şeklinde yazılır.)



Şekil 1.32: MOV Komutu

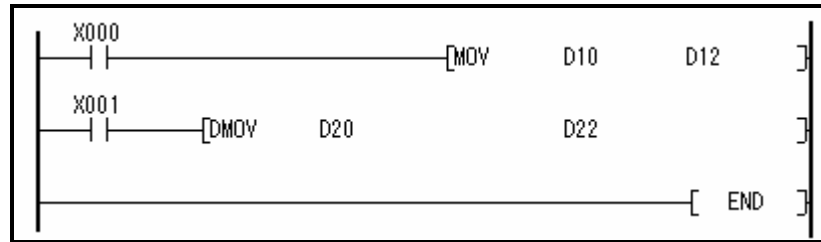
Taşıma butonu kapandıktan sonra tekrar açılmasına rağmen, hedefin içeriği eski durumuna dönmaz. Taşınan değer hedef içerisinde saklı tutulur. Butonun kapalı olması, elektrik akımının iletilmesi, açık olması ise elektrik akımının iletilmemesi anlamında kullanılmaktadır.



Şekil 1.33: MOV komut örneği

DMOV

Sayısal verinin uzunluğuna göre 16 bit ya da 32 bitlik veriler vardır. 32 bitlik veri ile ilgili çalışmalarda “D” (Double), komutun başına eklenmelidir. Burada kaynak ve hedef sayısı isteğe bağlı olarak ayrılabilir. Fakat herhangi bir karışıklığa meydan vermemek için, en düşük bit numarasını çift sayı olarak seçmek iyi olacaktır.



Şekil 1.34: DMOV komutu

D10’un içeriği D12’ye taşınır.
D20,D21’in içeriği D22,D23’e taşınır.

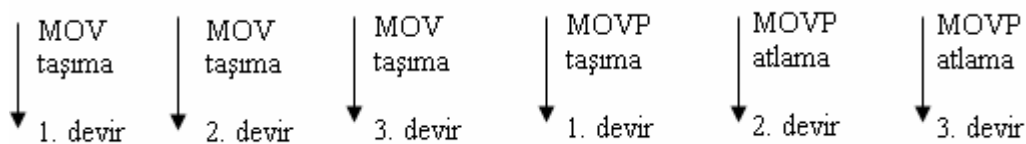
MOVP

Eğer taşıma işlemini, taşıma butonuna bastığımızda sadece bir seferlik yapmak istiyorsak bunu gerçekleştirebilmek için MOV komutunun sonuna P harfini eklemeliyiz. (MOVP) “P”nin anlamı Pulse, yani bir seferlik sinyal demektir.

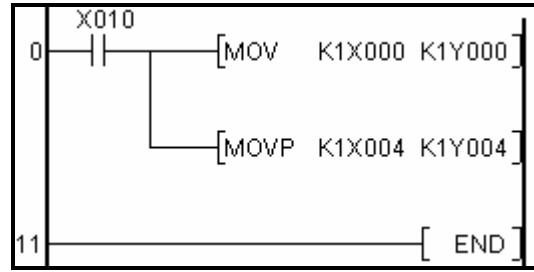


X0 butonuna basıldığı sürece D20’ deki bilgiler D30’a taşınır. Eğer bu esnada D20’nin içeriği değişirse, D30’un içeriği de değişecektir

X0 butonuna sürekli basılsa bile D20’ deki bilgiler sadece bir seferlik D30’a taşınır. Bu esnada D20’nin içeriği değişse bile D30’un içeriği değişmeyecektir. Bu komut kullanıldığında işlem süresi azalır



Şekil 1.35: MOVP komutu



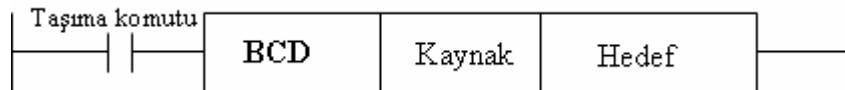
X10 butonuna basıldığı anda, X0-X3 arası Y0-Y3 arasına aktarılır. Aynı şekilde X4-X7 arası da Y4-Y7 arasına aktarılır. X10 girişi serbest bırakılmadan X0-X3 arasında yapılacak değişiklik Y0-Y3 arasına aynen yansıtılırken, X4-X7 arasında yapılan değişikliğin Y4-Y7 arasına yansımadağı görülür.

2.2.4. Verilerin Kodlarının Değıştirilerek Taşınması (BCD/BIN)

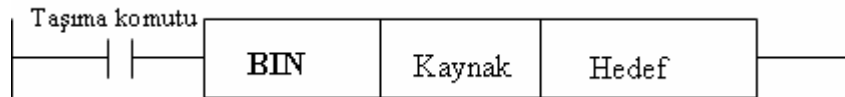
PLC içerisinde binary (BIN) kodlu ve binary kodlanmış decimal (BCD) işlemlerin aynı anda yapılması mümkün değildir. Çünkü bu kod sistemlerinin yapıları tamamen farklıdır.

BCD komutu ile 16 ya da 32 bitlik binary sayı, BCD'ye çevrilir. İşlem aralığı 16 bitlik işlemlerde 0 ile 9999 arasındadır. BCD komutu 7 segment display'e doğrudan veri çıkış komutu olarak kullanılabilir.

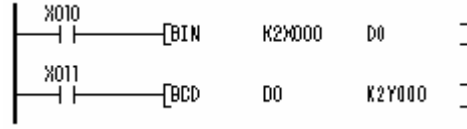
BIN komutu ile 16 ya da 32 bitlik BCD sayı, binary sayıya çevrilir. Eğer kaynak tarafından BCD biçiminde sayı sağlanmaz ise hata meydana gelir. Hata meydana geldiği zaman M8067 rölesinin kontağı kapanır. (Bu kontağı kullanarak hata durumunu anlayabiliriz.) BIN komutu, dijital anahtardan bilgileri doğrudan okumak için kullanılabilir.



Kaynak içindeki binary veri, BCD koduna çevrilerek hedef içinde saklanır.

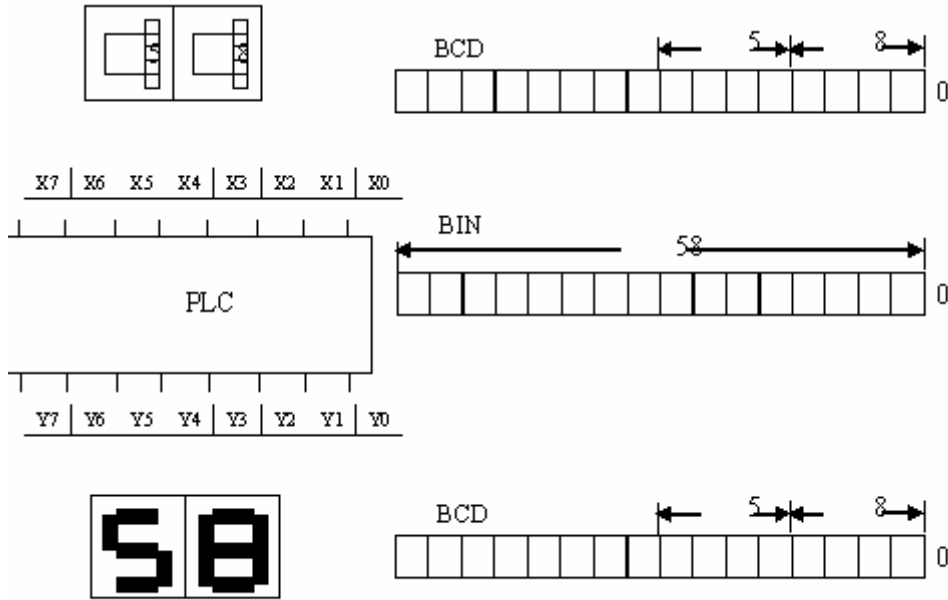


Kaynak içinde BCD kodundaki veri, binary koduna çevrilerek hedef içinde saklanır.



X0-X7 arası, binary sayıya çevrilerek D0 içerisinde kaydedilir.

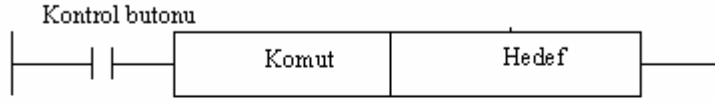
Binary olan D0'ın içeriği BCD koduna çevrilir ve Y0-Y7'ye çıkış olarak gönderilir.



Şekil 1.36: BCD-BIN komutu

2.2.5. Artırma/Azaltma (Increment / Decrement)

Bir kaydedici içindeki veriyi programın her devrinde bir artırmak ya da azaltmak mümkündür. INC komutu, ayarlanan hedefteki değeri programın her devrinde “1” artırır. DEC komutu ise hedefteki değeri programın her devrinde “1” azaltır.



Komut	Komutun içeriği
INC	$data + 1 = data$
DEC	$data - 1 = data$

Şekil 1.37: BCD-BIN komutu

INC

Kontrol butonuna basıldığı sürece programın her devrinde hedefe “1” eklenir. (Hedefteki sayı bir artar.)

Bu 16 bitlik bir komuttur. Bundan dolayı hedef +32767 sayısına ulaşınca, otomatik olarak tekrar -32768 sayısından itibaren artmaya başlayacaktır. 32 bitlik komut olması isteniyorsa, komut **INCD** şeklinde yazılmalıdır. Bu durumda hedefteki sayı +2147483647 değerine ulaştıktan sonra otomatik olarak -2147483648 sayısından itibaren tekrar artmaya başlayacaktır.

DEC

Kontrol butonuna basıldığı sürece programın her devrinde hedeften “1” çıkartılır. (Hedefteki sayı bir azalır.)

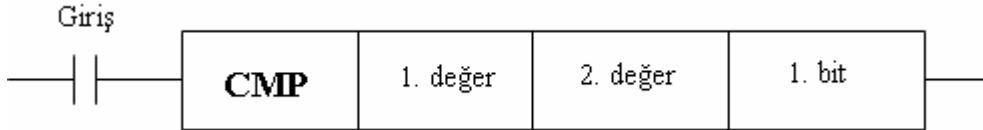
Bu 16 bitlik bir komuttur. Bundan dolayı hedef -32768 sayısına ulaşınca, otomatik olarak tekrar +32767 sayısından itibaren azalmaya başlayacaktır. 32 bitlik işlem yapmak için **DECD** şeklinde yazılmalıdır. Bu durumda hedefteki sayı -2147483648 değerine ulaştıktan sonra otomatik olarak +2147483647 sayısından itibaren tekrar azalmaya başlayacaktır.

INCP , DECP

Bir kaydedici içindeki değeri azaltma ya da artırma işlemini taşıma butonuna sürekli basılsa bile yalnızca bir kez yapmak için komutun sonuna **P** yazılır. Bu durumda işlem yalnızca butona basılan ilk program devrinde bir kez yapılır.

2.2.6. Karşılaştırma (Comparison)

Veri kaydedicide bulunan sayısal bir veri ile, zamanlayıcı ya da sayıcının aktif değerini, X, Y, M, S gibi bit aygıtlarının birleştirilmesinden oluşan sayının ya da K gibi sabit bir sayının karşılaştırılması mümkündür.



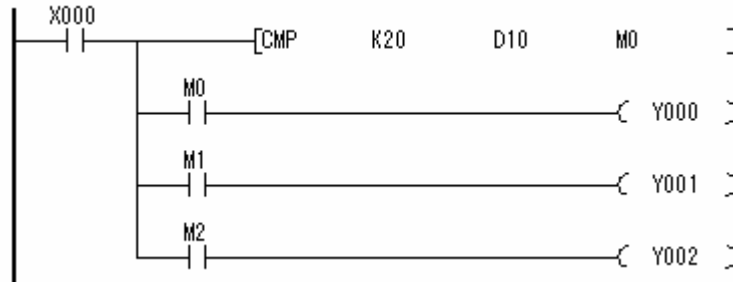
Karşılaştırmanın sonucuna göre; eğer 1. değer 2. değerden büyük ise, 1. bit ON olur. Eğer 1. değer 2. değere eşit ise, 2. bit (1. bitten sonra gelen bit) ON olur. Eğer 1. değer 2. değerden küçük ise, 3. bit ON olur.

- | | |
|--------------------|--|
| 1.değer > 2. değer | 1. bit ON |
| 1.değer = 2. değer | 2. bit ON (2. bit 1. bitten sonra gelen bittir.) |
| 1.değer < 2. değer | 3. bit ON (3. bit 2. bitten sonra gelen bittir.) |

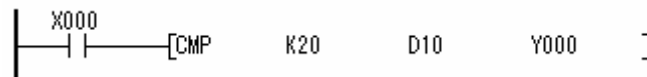
ÖRNEK

X0 butonuna basıldığında 20 değeri ile D10 veri kaydedicisinin içeriğini karşılaştıracak ve 20 sayısının D10'un içeriğinden büyük olması durumunda Y0, 20 sayısının D10'un içeriğine eşit olması durumunda Y1 ve 20 sayısının D10'un içeriğinden küçük olması durumunda Y2 çıkışını aktif yapacak olan programı yazınız.

1.



2.



Yukarıda gösterilen iki program aynı amaca yöneliktir. Birinci şekilde M0-M2 yardımcı röle olarak kullanılmış ve bu yardımcı rölelerin kontakları kullanılarak Y0-Y2 çıkışları çalıştırılmıştır. İkinci şekilde ise Y0-Y2 arası direkt olarak belirtilmiştir.

ÖRNEK

(1-1) X0 butonuna basıldığında 10(Decimal) sabit sayısı D10 veri kaydedicisine taşınacaktır.

(1-2) X1 butonuna basıldığında 20(Decimal) sabit sayısı D10 veri kaydedicisine taşınacaktır.

(1-3) X2 butonuna basıldığında 30(Decimal) sabit sayısı D10 veri kaydedicisine taşınacaktır.

(2) X3 butonuna basıldığında karşılaştırma yapılacaktır. D10 ile 20 sabit sayısının karşılaştırılmasından sonra ,

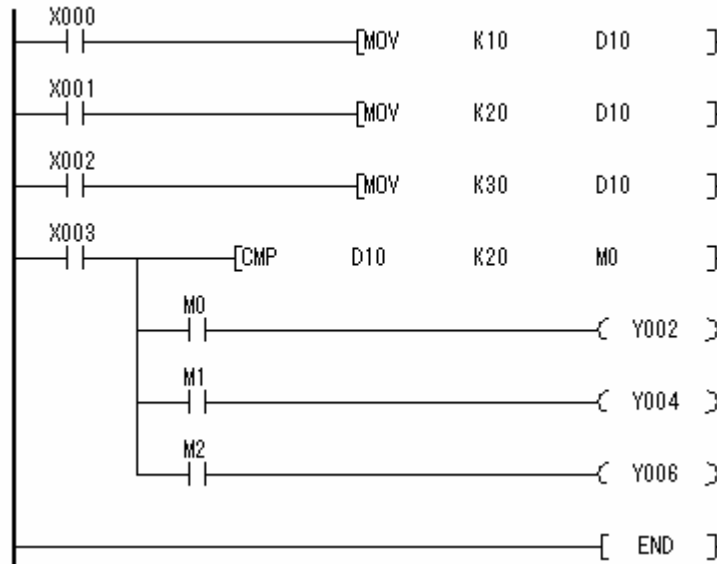
D10>20(decimal) ise Y2 ON,

D10=20(decimal) ise Y4 ON,

D10<20(decimal) ise Y6 ON olacaktır.

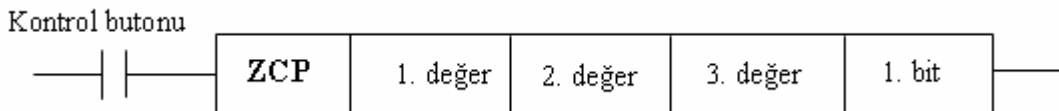
Yukarıdaki şartları sağlayan programın merdiven diyagramını çiziniz.

Cevap



ZCP

Alan karşılaştırma (Zone compare) komutudur. **ZCP** komutu 1.değer ve 2.değer ile belirlenen veri aralığı ile 3. değeri karşılaştırmak için kullanılır.



Şekil 1.38:ZCP komutu

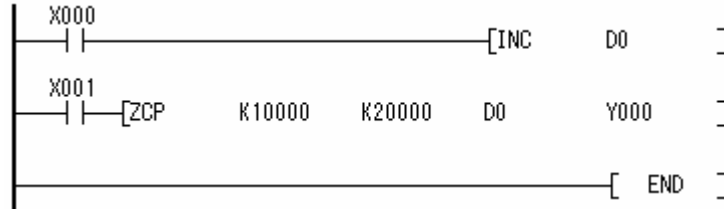
Karşılaştırmanın sonucuna göre eğer 3. değer, 1. ve 2. değerlerden küçük ise, 1. bit ON olur. Eğer 3. değer, 1. ve 2. değere eşit ya da iki değer arasında ise 2. bit ON olur. Eğer 3. değer, 1. ve 2. değerlerden daha büyük ise 3. bit ON olur.

- | | |
|--------------------------------|---|
| 1. değer, 2. değer > 3. değer | 1. bit ON |
| 1. değer ≤ 3. değer ≤ 2. değer | 2. bit ON (2. bit, 1. bitten sonra gelen bittir.) |
| 1. değer, 2. değer < 3. değer | 3. bit ON (3. bit, 2. bitten sonra gelen bittir.) |

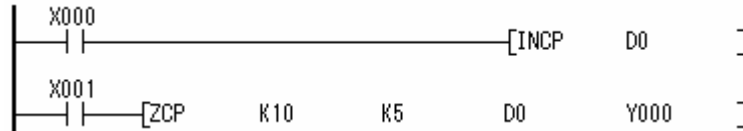
ÖRNEK

X0'a basıldığında D0'ın içeriği sürekli olarak artacak. X1'e basıldığında D0'ın değeri ile decimal 10000 ve 20000 sayıları karşılaştırılacak. Karşılaştırmanın sonucuna göre;

- Eğer D0'ın değeri 10000'den küçük ise Y0 çıkışı aktif olacak.
- Eğer D0'ın değeri 10000 ya da 20000'e eşit olursa veya bu iki değer arasında olursa Y1 çıkışı aktif olacak.
- Eğer D0'ın değeri 20000'den büyük olursa Y2 çıkışı aktif olacak.



2. değer 1. değerden küçük olması durumunda, 2. değer 1. değer olarak ayarlanır.



ÖRNEK

Aşağıdaki işlemleri gerçekleştirecek programı yapınız.

X1'e basıldığında zaman sayıcı saymaya başlayacak.

Eğer 20 saniye geçtiği düşünülüyorsa X1 butonu serbest bırakılacak.

20 saniye ile zaman sayıcının içeriği **Karşılaştırma komutu** kullanılarak karşılaştırılacak (Çıkış, Y0-Y2).

19-21 saniye arası ile zaman sayıcının içeriği **Alan Karşılaştırma komutu** kullanılarak karşılaştırılacak. (Çıkış Y10-Y12)

* 60 saniye sonra zaman sona erecek.

M8011 10ms clock

M8012 100ms clock

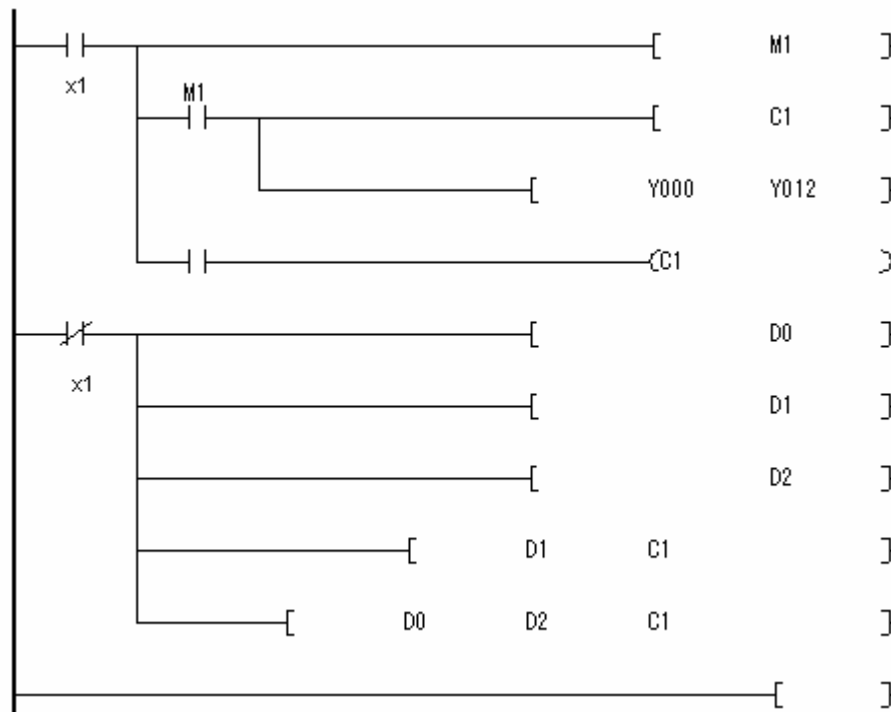
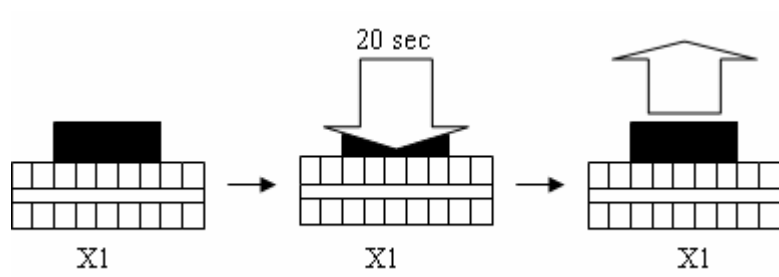
M8013 1 sec clock

M8014 1 min clock

ZRST Alan reset komutu

ZRST M0 M10

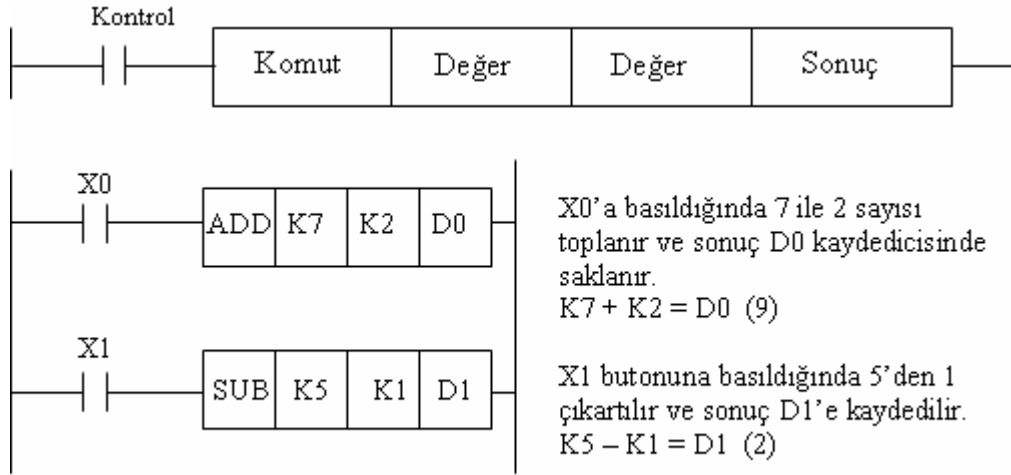
M0'dan M10'a kadar olan bölgeyi resetler.



2.2.7. Toplama ve Çıkarma (Addition - Subtraction)

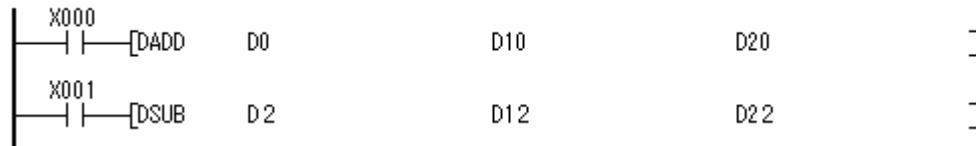
İki değer arasında toplama ve çıkarma işlemleri **ADD** ve **SUB** komutları ile yapılır.

Komut Yazılımı	Komut Adı
ADD	BINARY ekleme
SUB	BINARY çıkarma



Şekil 1.39:ADD ve SUB komutu

ADD ve **SUB** komutları 16 bitlik işlem yapar. 32 bitlik işlem yapmak için **DADD**, **DSUB** komutları kullanılmalıdır.



$$D1, D0 + D11, D10 \longrightarrow D21, D20$$

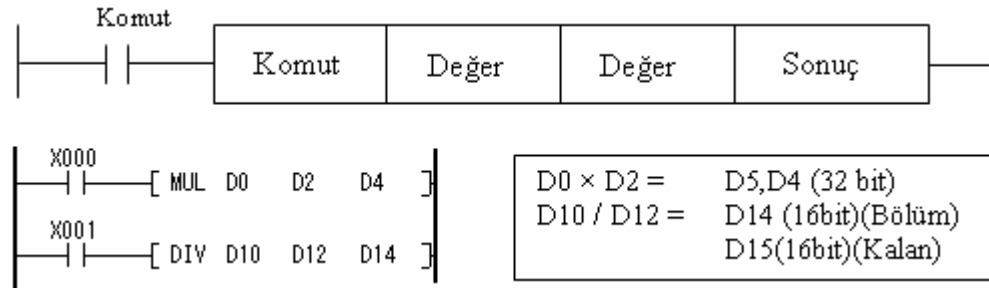
$$D3, D2 - D13, D12 \longrightarrow D23, S22$$

ğer hesaplanan sonucun "0" olması, özel yardımcı röle **M8020 (zero flag)** çalışır. Eğer hesaplanan sonucun 16 bitlik işlemlerde 32767'yi ve 32 bitlik işlemlerde de 2147483647'yi geçerse özel yardımcı röle **M8022 (carry flag)** çalışır. Aynı şekilde hesaplama sonucu, 16 bitlik işlemlerde -32768'i ve 32 bitlik işlemlerde de -2147483648'i geçecek olursa (negatif yönde), özel yardımcı röle **M8021 (borrow flag)** çalışır.

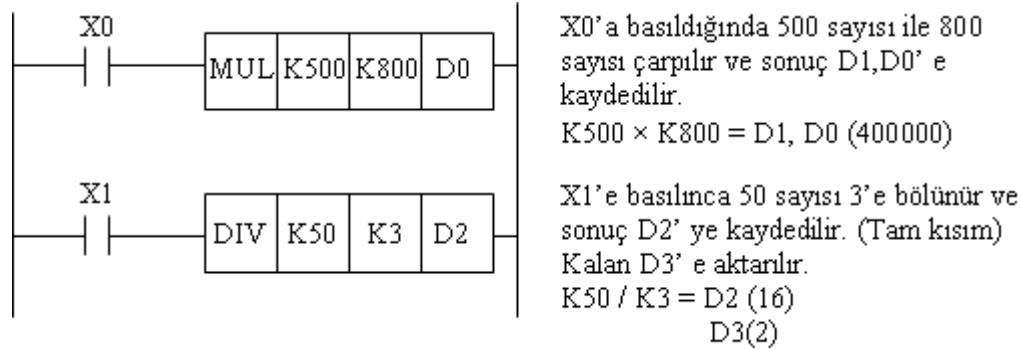
2.2.8. Çarpma ve Bölme (Multiplication and division)

Çarpma ve bölme işlemleri **MUL** ve **DIV** komutları kullanılarak yapılır. Aşağıdaki tablo çarpma ve bölme komutlarını göstermektedir.

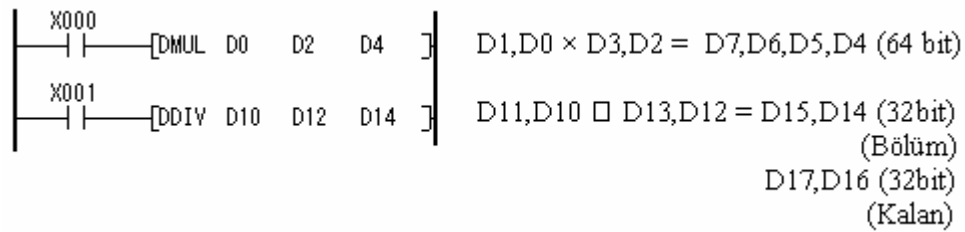
Komut Yazılımı	Komut Adı
MUL	BINARY çarpma
DIV	BINARY bölme



Şekil 1.40: MUL ve DIL Komutu



MUL ve **DIV** komutları 16 bitlik işlemler içindir. 32 bitlik işlem yapmak için **DMUL**, **DDIV** şeklinde kullanılmalıdır.



2.3. Zaman Kontrol Komutları

2.3.1. Zamanlayıcı İşlemleri

Belirli bir zaman gecikmesiyle bir işlemin yapılması için zamanlayıcılar kullanılır. Zamanlayıcının değeri, önceden ayarlanan değere ulaştığında çıkış kontakları konum değiştirir. Program hafızasındaki sabit bir sayı ya da veri kaydedicinin (D) içeriği zamanlayıcıya ön ayar değeri olarak verilebilir.

PLC'deki zamanlayıcı 1ms, 10 ms veya 100 ms'lik clock sinyaliyle sayma yapacak şekilde ayarlanabilir. 10 ms'lik bir zamanlayıcıya 457 ön değeri girildiğinde gerçek işlem zamanı aşağıdaki şekilde hesaplanır.

$$457 \times 10\text{ms}$$
$$457 \times 0,01\text{s} = 4,57\text{s}$$

Hatırlamayan zamanlayıcı

(Non-retentive timer)

100ms zamanlayıcı T0-199 (200p)

(0,1-3276,7s)

10ms zamanlayıcı T200-245(46p)

(0,01-327,67s)

(MITSUBISHI FX)

Hatırlayan Zamanlayıcı

(Retentive timer)

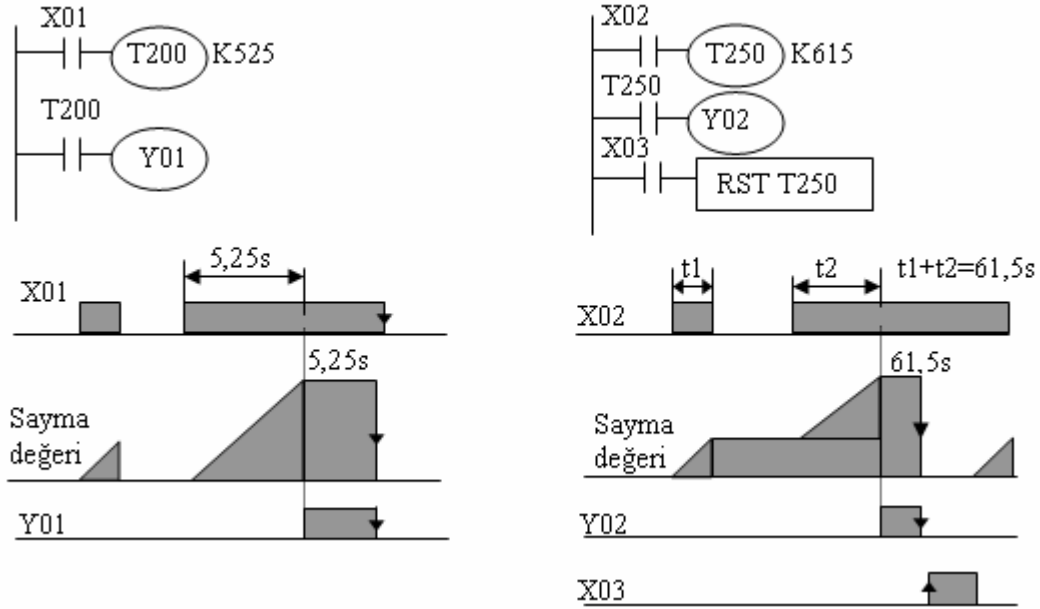
1ms hatırlayan zamanlayıcı T246-249(4p)

(0,001-32,767s)

100ms hatırlayan zamanlayıcı T250-255(6p)

(0,1-3276,7s)

(MITSUBISHI FX)

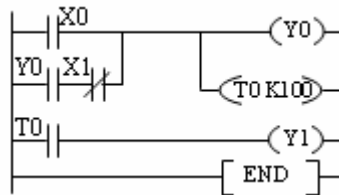


Şekil 1.41: Zamanlayıcı Çeşitleri

Hatırlayan zamanlayıcı (retentive timer) sayma değerini kontrol kontağı açılrsa bile hafızada tutar. Kontrol kontağı tekrar kapanınca kaldığı yerden saymaya devam eder. Hatırlayan zamanlayıcıyı resetlemek için bobin enerjisinin kesilmesi yeterli değildir. Reset için özel reset komutu kullanılmalıdır.

Zamanlayıcı ön değeri, sabit bir sayı olarak program yazılırken girilebilir.

ÖRNEK

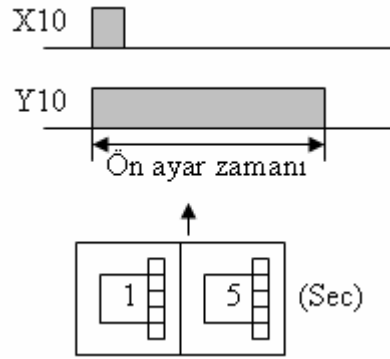


X0 butonuna basıldığında Y0 aktif olacak.
Y0'dan 10 saniye sonra ise Y1 aktif olacaktır.
T0 100 ms'lik zamanlayıcıdır. 100*100ms = 10 sn.
X1 basıldığında ise Y0 ve Y1 "OFF" olacaktır.

Bir veri kaydedicinin içeriği zamanlayıcı ön değeri olarak kullanılabilir. Harici bir veri girişi de zamanlayıcı ön değeri olarak set edilebilir. Sabit sayı veya veri kaydedicinin içeriği, program hafızasında bulunur. Eğer veri kaydedicinin değeri negatif bir sayı ise zamanlayıcı çalışmayacak ve işlem hatası verecektir.

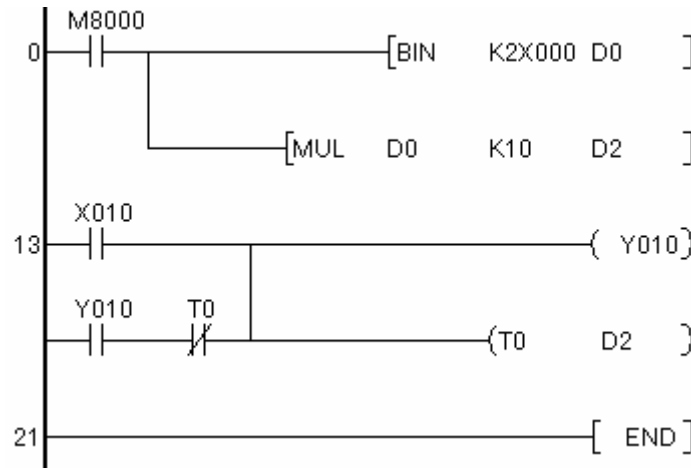
ÖRNEK

Bu örnekte iki dijitalik dijital anahtar ile zamanlayıcının ön ayar değeri saniye biriminde ayarlanacaktır.



Dijital anahtarda (thumbwheel) her bir dijit için dört bitlik çıkış vardır. İki dijit için PLC'nin X0-X7 girişleri kullanılacak. Bu örnekte 100ms zamanlayıcı kullanılacak.

X10'a basıldığında Y10 hemen ON olacak ve ayarlanmış süre kadar çıkışı aktif kalacak.



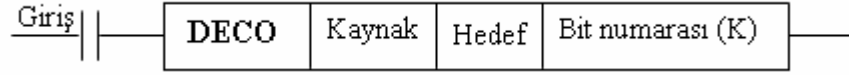
BCD olan X0-X7 arası BINARY sayıya çevrilir ve D0'da saklanır.

$D0 \times K10 = D3, D2$

T0, 100 ms'liktir. Saniyeye çevirmek için ön değer 10 ile çarpılmalıdır.

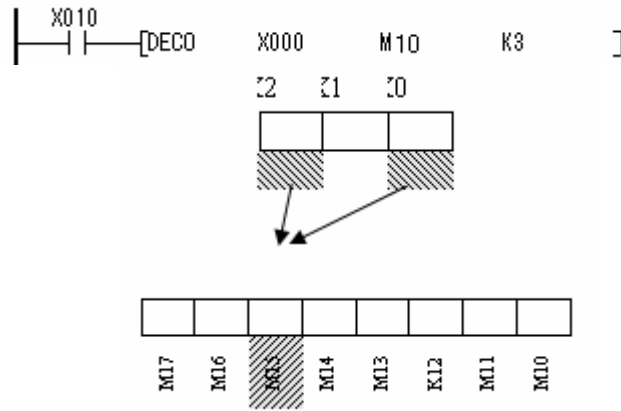
Mühürleme kontağı kullanılarak, ön ayar değerine ulaşıldığında Y10'un enerjisiz kalması sağlanır.

2.3.2. DECO (Decode) Komutuyla Zamanın Set Edilmesi



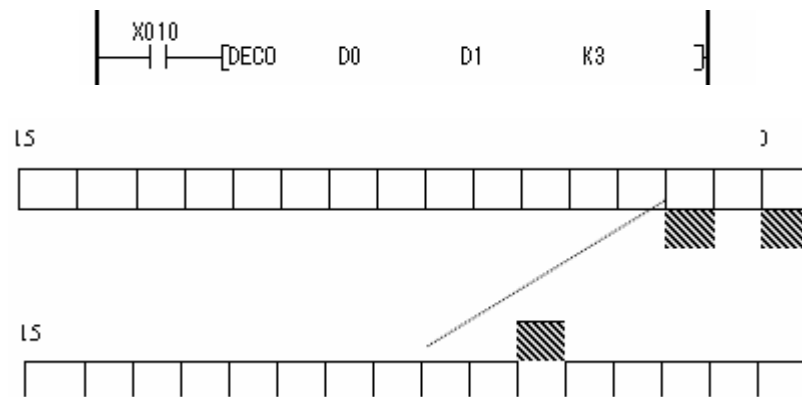
DECO, kod çözme komutudur. Kaynağın kodu hedefe sadece bit numarası olarak çözülür. Kaynak bit aygıtı olduğu durumda bit numarası 0 ile 8 arasında olmalıdır. Eğer kaynak kelime aygıtı ise, bu durumda bit numarası 0 ile 4 arasında olabilir. 0 ile 8 arasındaki bit numaralarını değiştirmek sureti ile kod çözme işlemi 0 ile 255 arasında yapılabilir.

(1) Hedefin bit aygıtı olduğu durumda;



Kaynak 1+4=5, bu nedenle M10'dan itibaren 5. bit (M15) ON olur. Eğer kaynağın hepsi "0" ise M10 ON olur. Bit numarası "0" ile "8" arasında bir sayı olmalıdır

(2) Hedefin kelime aygıtı olduğu durumda;

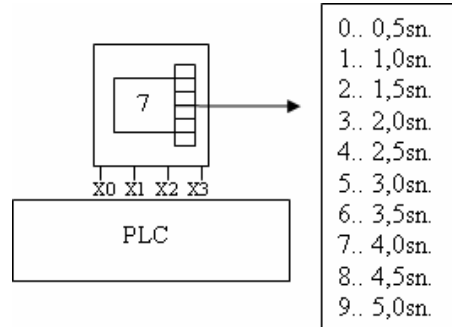


Şekil 1.42: DECO komutu

Hedefe, düşük bittten itibaren kod çözülür. (K3 ise B0-B2 arası bit numarası olarak alınır.) Bit numarası “0” ile “4” arasında olmalıdır. Bit numarası dörtten az ise, hedefin yüksek sekiz biti “0” olur.

Harici bir veri girişi DECO komutu kullanılarak örnek 7’de olduğu gibi zamanlayıcı ön değeri olarak set edilebilir.

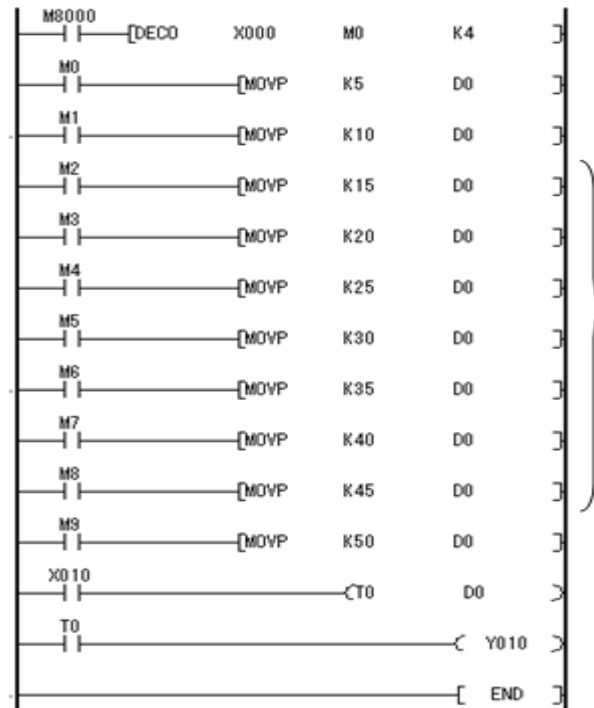
ÖRNEK



Bir haneli dijital anahtar kullanarak yanda gösterilen on farklı zaman değerini PLC içindeki T0 zamanlayıcısına set etmek için gerekli programı yazınız.

Böyle bir uygulama ile zamanlayıcı değeri, programa müdahale etmeden değiştirilebilir.

Bu uygulama içinde DECO komutu kullanılacaktır.



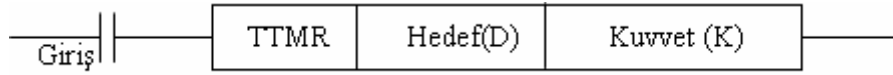
M8000: Çalışma habercisi
X3-0 arasının kodu M15-0 arasına çözülür.
X3-0 arası binary olarak ele alınır.
M0:ON(X3-X0:OFF) K5....D0
M1:ON(X3-X1:OFF X0:ON) K10....D0

→

→

Yukarıdaki düzen ile aynı.
X10:ON Zamanlayıcı çalışır. (T0) →
(Ön değer 5 ile 50 arasında olacaktır.) (0,5s. – 5s.)
T0:ON Y10:ON →

2.3.3. TTMR Komutu ile Zamanın Set Edilmesi

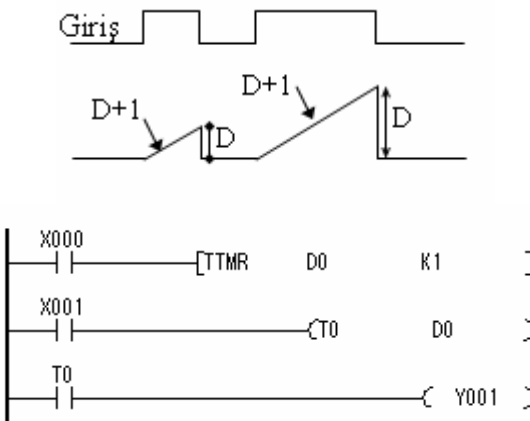


Girişin basılı tutulduğu süre ,K kuvvet değeriyle çarpılarak hedefe yazılır.

Bu komut kullanılarak zamanlayıcının ön değeri giriş tarafından ayarlanır. Girişe basma zamanı T(saniye) olarak belirlendiğinde, gerçek hedef aşağıdaki tabloya göre tespit edilir.

Kuvvet(K)	Hedef(D)
K0	T
K1	10T
K2	100T

Giriş ON olduğu süre D+1 içindeki değer artmaya devam eder.
Giriş OFF olduktan sonra, D+1 içeriği D'ye aktarılır ve D+1 resetlenir.

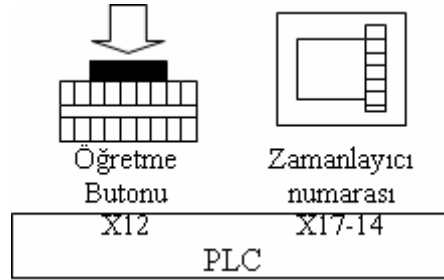


Şekil 1.43: TTMR komutu

X0'a basma zamanı sayılır. Bu süre 10 ile çarpılır. Sonuç D0 veri kaydedicisinin içeriği olarak saklanır. Bu değer zamanlayıcı ön ayar değeri olarak set edilir.

X1'e basıldığı andan itibaren zamanlayıcı saymaya başlar. Sayma değeri D0'ın içindeki değere ulaştığında T0'ın kontakları konum değiştirir ve Y1 çıkışı aktif olur.

ÖRNEK



Bu örnekte 10 farklı zamanlayıcıya (T0-T9) dışarıdan öğretme butonu ile zaman değeri ayrı ayrı girilecektir.

Öğretme butonuna basma süresi 10 ile çarpılarak D0'a kaydedilecek.

D0'daki değerin hangi zamanlayıcıya aktarılacağı dijital anahtar ile belirlenecektir. (Bunun için dizin kaydedici kullanılacak.)

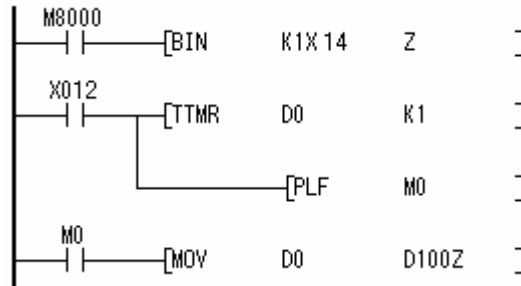
M8000: Çalışma habercisi

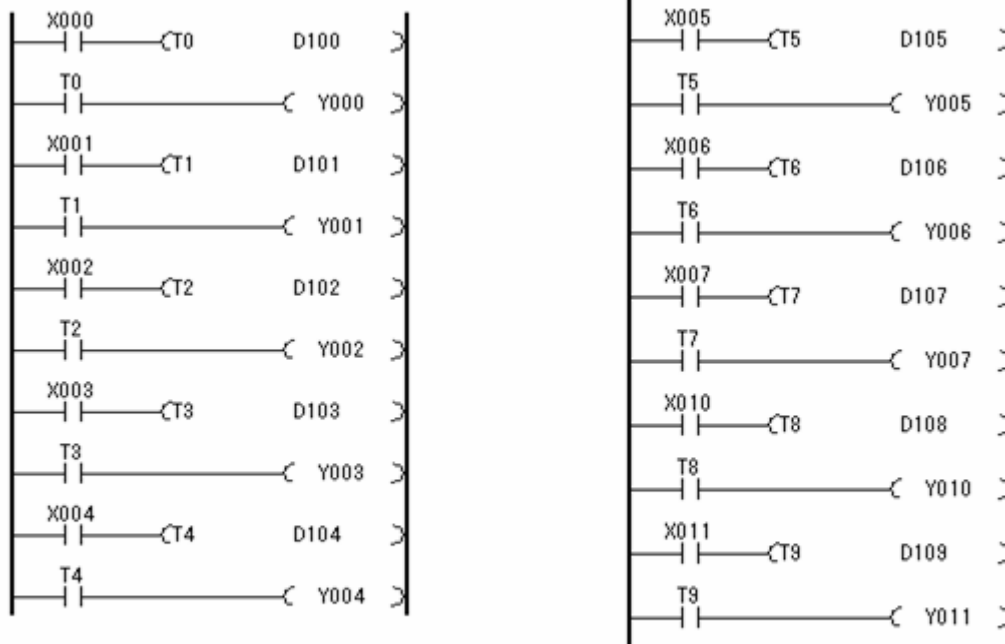
X17-14 arası binarye çevrilerek dizin kaydediciye kaydedilir (dizin kaydedici, Z).

Basma zamanı sayılır (X12) ve 10 ile çarpılarak D0'a aktarılır.

Öğretme butonunun bırakıldığı anlaşılır.

Öğretme butonu bırakıldığında D0'daki veri D (100+Z)'ye aktarılır.



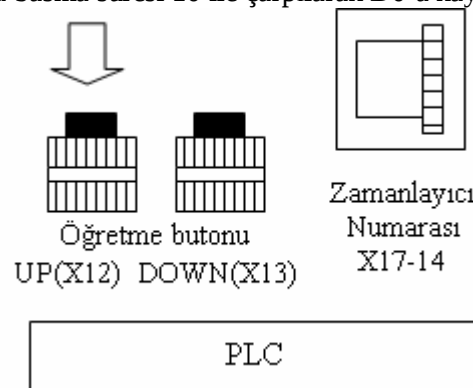


X0-X11 arasındaki butonlarla Y0-Y11 çıkışların kontrolü yapılır.

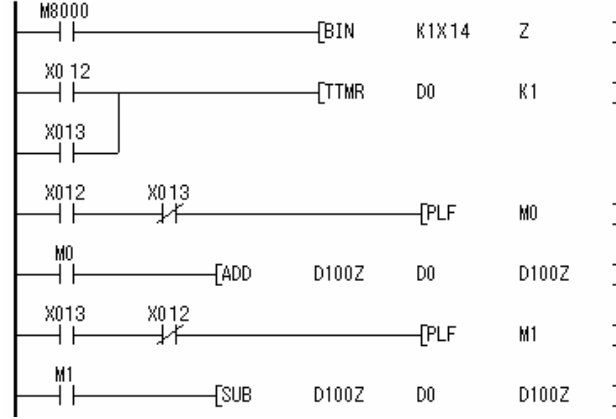
ÖRNEK

Bu örnekte 10 farklı zamanlayıcıya (T0-T9) dışarıdan öğretim butonu ile zaman değeri ayrı ayrı girilecektir. X12 basılı tutularak zaman artırılabilecek, X13 basılı tutularak da daha önce set edilmiş olan zaman azaltılabilecektir.

Öğretim butonuna basma süresi 10 ile çarpılarak D0'a kaydedilecek.



D0'daki değerin hangi zamanlayıcıya aktarılacağı dijital anahtar ile belirlenecektir. Bunun için izin kaydedici kullanılacaktır .



M8000: Çalışma habercisi

X17-14 arası binary sayıya çevrilir ve izin kaydediciye (Z) kaydedilir.

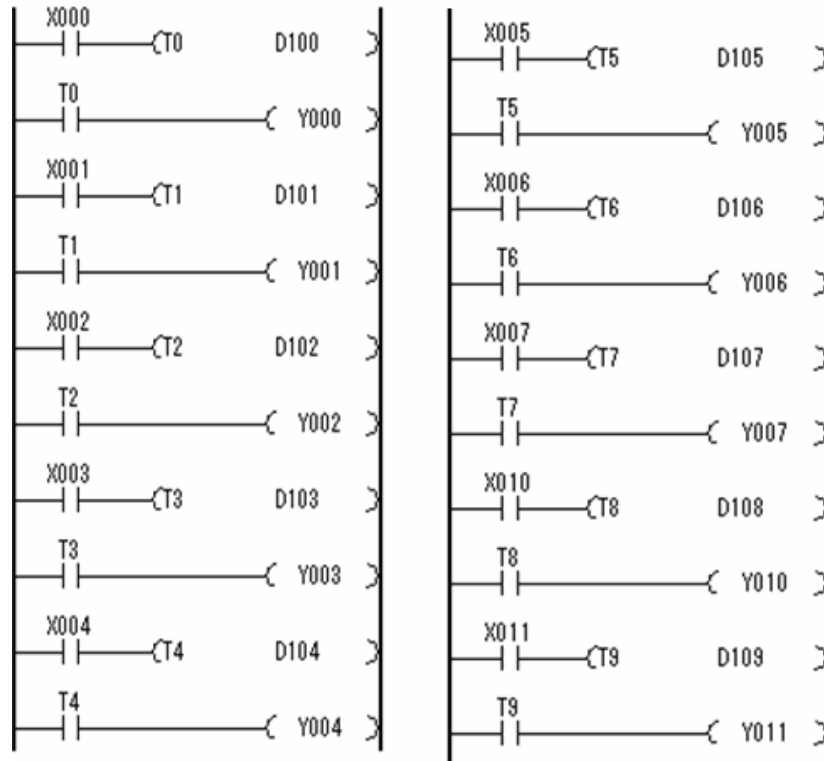
X12,13'e basma zamanı sayılır ve sonuç 10 ile çarpılarak D0'a aktarılır.

Öğretme zamanının bulunması (Yukarı)

$D100Z + D0 \rightarrow D100Z$

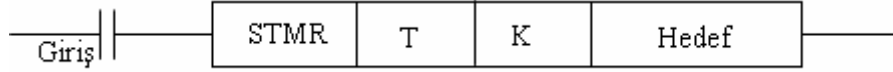
Öğretme zamanının bulunması (Aşağı)

$D100Z - D0 \rightarrow D100Z$

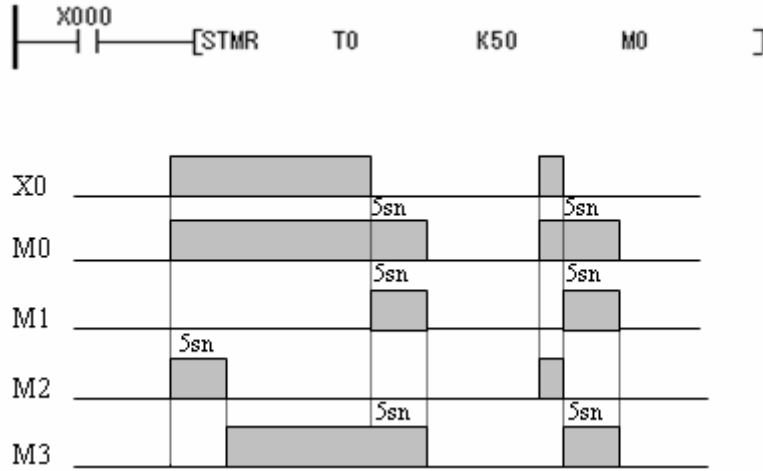


X0-X11 arasındaki butonlara basıldığında Y0-Y11 arası çıkışlar ayarlanmış zaman kadar sonra çıkış verir.

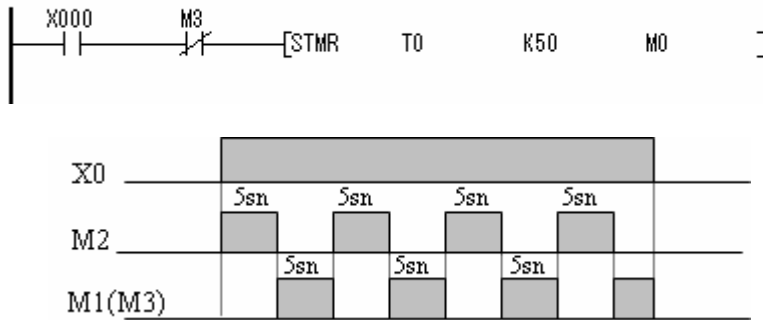
2.3.4. Özel Zamanlayıcı



STMR hedef D+0'dan D+3'e kadar olan alanı etkileyen, özel bir zamanlayıcıdır. D+0 ters zamanlayıcıdır. Giriş OFF olduktan sonra zamanlayıcı saymaya başlar ve ayarlanan sürenin sonunda kontağını OFF yapar. D+1 ise ters zamanlayıcı gibi çalışır fakat aradaki fark; giriş OFF olunca D+1 ON olur. Ayarlanan süre sonunda tekrar OFF olur. D+2 ve D+3 aşağıda zaman grafiğinde gösterildiği gibi çalışır.



D+3 aşağıdaki şekilde gösterildiği gibi kullanılabilir. Bu durumda D+1 ve D+2 titreşim devresini oluşturacaktır. (D+1 ve D+2 sırası ile ON, OFF olacaklardır.)



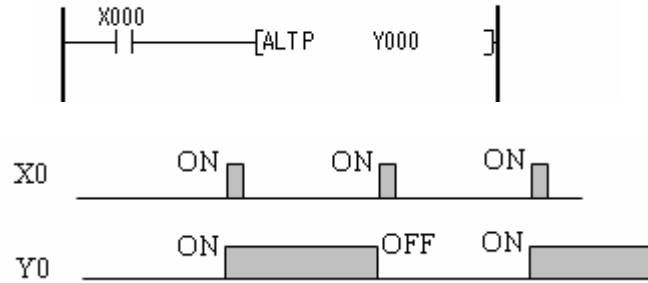
Şekil 1.44: Özel zamanlayıcı

2.3.5. ALT Fonksiyonu



Giriş sinyali verildiğinde, hedef ON olur ve giriş sinyali kesilse bile hedef durumunu korur. İkinci kez giriş sinyali verildiğinde, hedef OFF olur ve giriş sinyali kesilince yine bir önceki gibi hedef durumunu korur. (Hedef OFF olarak kalır.)

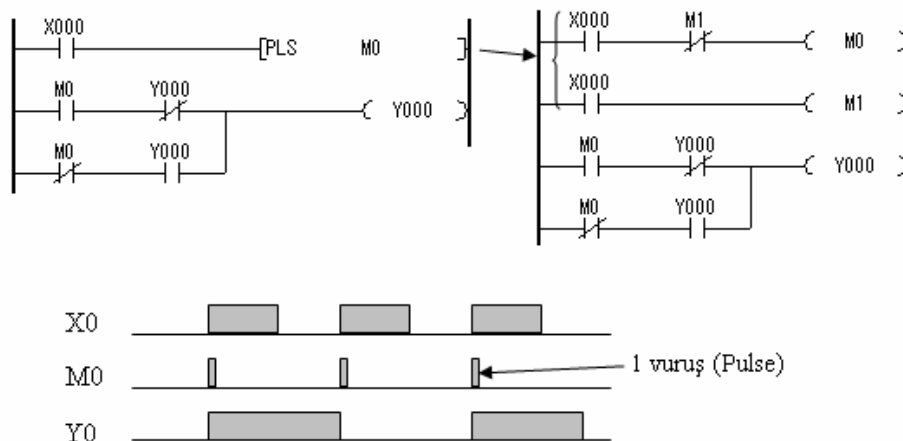
ALT komutu, iki koşullu işlemlerde geçiş için ideal bir komuttur. (Aynı girişten başlama ve durdurma gibi).



X0 ON olduğunda Y0 ON olur.
Y0 ON durumunu korur.

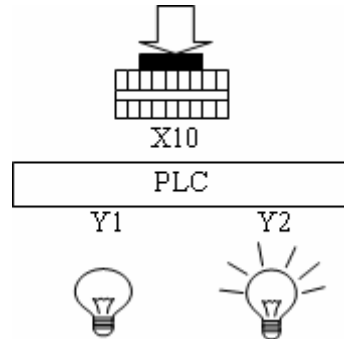
X0 tekrar ON olduğunda Y0 OFF olur ve OFF durumunu korur.

ALT komutunun işlevi aşağıdaki devre kurularak da sağlanabilir. Devreyi inceleyerek çalışmasını kavrayınız.



Şekil 1.45: Alt fonksiyonu

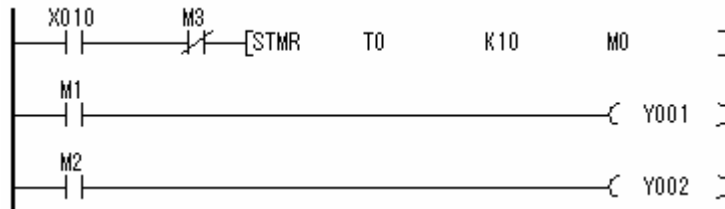
ÖRNEK



X10'a butonuna basılı tutulduğu sürece Y1 ve Y2 çıkışları bir saniye aralıklarla ON / OFF olacaktır.

(Çözüm 1)

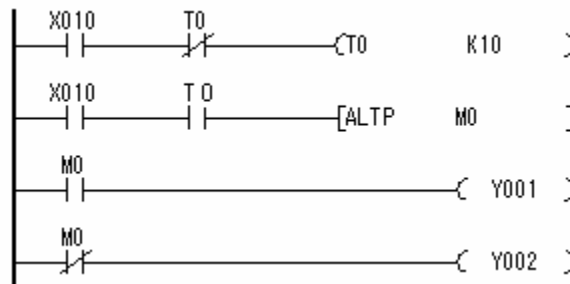
STMR komutu kullanılarak yapılabilir.



X10'a basıldığı sürece Y2, 1 saniye aktif olur. Daha sonra Y2 çıkış vermez ve 1 saniye boyunca Y1 aktif olur. Bu işlem sürekli olarak tekrar eder.

(Çözüm 2)

ALT komutu kullanılarak yapılabilir.



2.4. Sayıcı Fonksiyonu

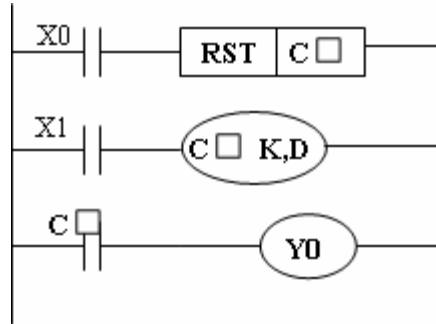
İşlemlerin belirli sayıda yapılmasını sağlamak için sayıcılar kullanılır. Sayıcı, girişine uygulanan sinyal sayısını sayar ve sayılan aktif değer, girilen ön değere ulaştığında kontakları konum değiştirir. Sayma işlemi yukarı ve aşağı yönde olabilir. Bu bölümde PLC’lerde bulunan iki çeşit sayıcıyı inceleyeceğiz.

1. 16 bit yukarı sayıcı
2. 32 bit yukarı / aşağı sayıcı

2.4.1. 16 Bit Yukarı Sayıcı (up counter)

16 bit yukarı sayıcı 1 ile 32767 arasında sayma yapabilir. Sayıcının bobinine her enerji verilişinde, aktif değeri bir artar. Sayıcının aktif değeri, önceden tanımlanan ön değere ulaştığı anda kontağı konum değiştirir. (Açık olan kontakları kapanır, kapalı olanlar açılır.) Bundan sonra, sayıcı reset edilene kadar aktif değeri sabit kalır. Reset işlemi, RST komutu ile yapılır. RST komutu uygulandığında, sayıcının değeri “0”a dönecektir. Aynı zamanda çıkış kontakları da resetlenir.

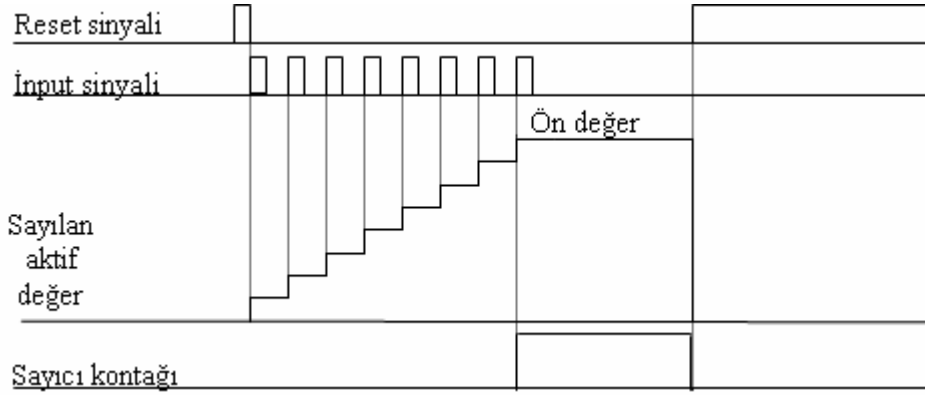
Sayıcının ön değeri doğrudan, sabit bir sayı kullanılarak ayarlanabilir veya dolaylı olarak, veri kaydedicide bulunan veri kullanılarak da ayar işlemi yapılabilir.



RST komutu, sayıcı işlemlerinin başında mutlaka kullanılmalıdır.

Giriş sinyalinin her “ON” olması durumunda, sayıcı değeri bir artacaktır.

Sayıcının değeri, ön değere ulaştığında (K,D) sayıcının kontağı konum değiştirir.



Şekil 1.46: Sayıcıların çalışma prensibi

FX2N (MITSUBISHI) model PLC’de 16 bit sayıcı numaralandırması aşağıdaki gibi yapılmıştır. Farklı PLC’ler için kullanım kitapçıklarından yararlanabilirsiniz.

16 bit genel yukarı sayıcı	C0-C99 (100 adet)
Hafızalı 16 bit yukarı sayıcı	C100-C199 (100 adet)

2.4.2. 32 Bit Yukarı / Aşağı Sayıcı

32 bit yukarı / aşağı sayıcı kullanılarak –2147483648 ile +2147483647 değerleri arasında sayma işlemi yapılabilir. Bu sayıcıya ön değer olarak negatif sayı da verilebilir. Sayıcının değeri artarak ön değere ulaştığında, diğer sayıcılarda olduğu gibi kontakları konum değiştirir.

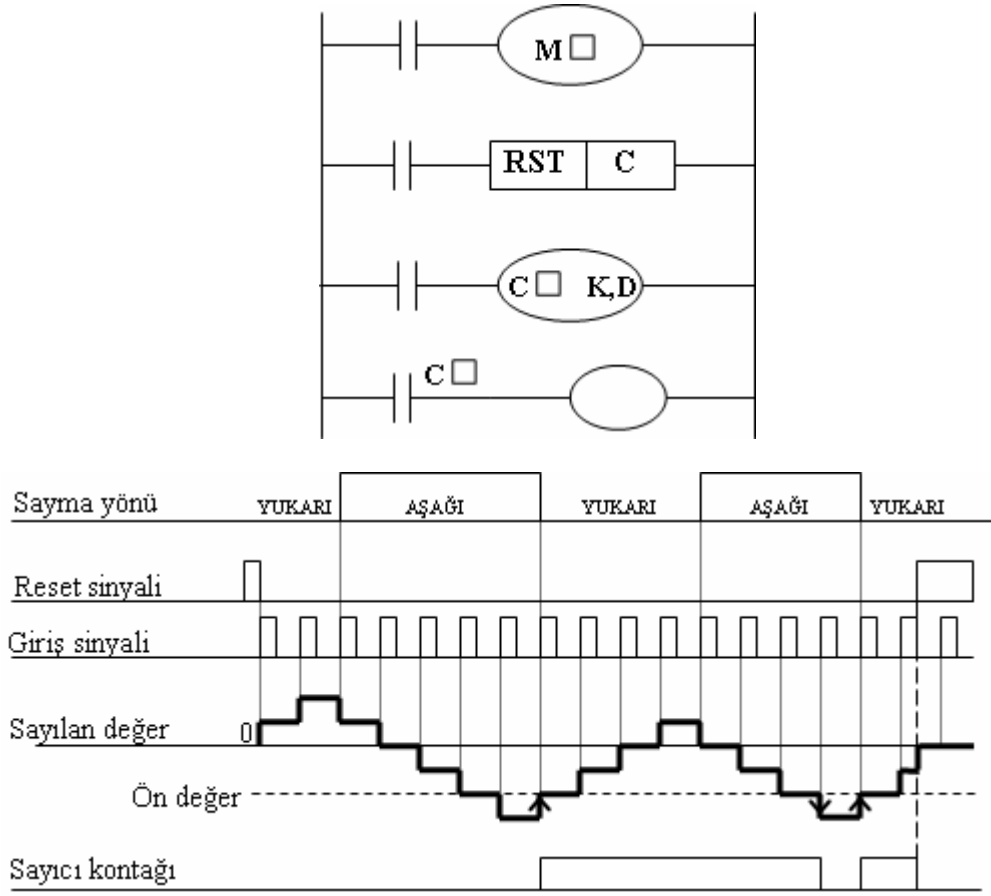
Sayıcının sayma yönü (sayma işleminin yukarı ya da aşağı olacağı), sayıcı numarasına (C) bağlı olarak kullanılacak yardımcı röle (M) ile belirlenir.

Sayıcının ön değeri, sabit bir sayı (K) kullanılarak ayarlanabileceği gibi, dolaylı yoldan, bir veri kaydedicinin içeriği kullanılarak da ayarlanabilir. Ön değer olarak veri kaydedicisi kullanıldığında, bir sonra gelen veri kaydedici otomatik olarak ayrılır. (Bu veri kaydedicisinin ayrılması, bir tek veri kaydedicisi yeterli olsa bile yapılır.)

Sayma yönü RST komutu, sayma işleminin başında mutlaka uygulanmalıdır.

Her giriş sinyalinde sayıcı değeri bir artar.

Sayıcının değeri ön değere (K, D) ulaştığında kontakları konum değiştirir.



Şekil 1.47: 32 Bitlik sayıcıların çalışma prensibi

Sayma değeri yukarı yönde artarak ön değere ulaştığında sayıcı bobini enerjilenir ve kontakları konum değiştirir. Sayma değeri sıfırdan set edilen negatif ön değere ulaştığında ise sayıcı bobini resetlenir. Ayrıca reset komutu (RST) uygulandığında sayılan değer ve sayıcının kontağı aynı anda resetlenir.

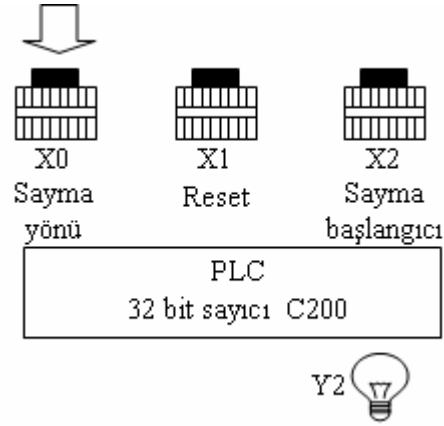
FX2N (MITSUBISHI) PLC’de 32 bit sayıcı numaralandırması aşağıdaki gibi yapılmıştır. Farklı PLC’ler için kullanım kitapçıklarından yararlanabilirsiniz.

32 bit genel yukarı / aşağı sayıcı	C200-C219 (20 adet)
Hafızalı 32bit yukarı / aşağı sayıcı	C220-C234 (15 adet)

Hafızalı sayıcılar pil ile desteklenir. PLC enerjisi kesilse bile sayıcının sayma değeri saklı tutulur. Enerji geldiğinde kaldığı yerden saymaya devam eder.

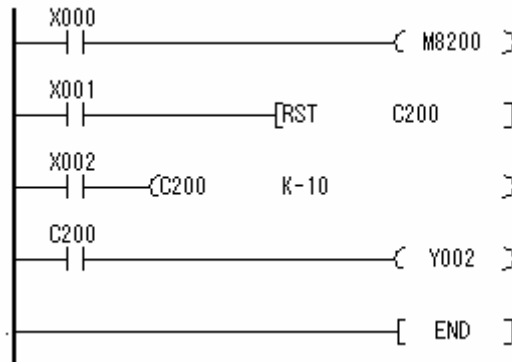
Sayma yönü, özel yardımcı röle (M8200 – M8234) kullanılarak belirlenir. C200 için M8200 ve C234 için M8234 kullanılır. Yardımcı röle ON olduğunda ilgili sayıcı aşağı sayma yapar. Yardımcı röle OFF olduğunda ilgili sayıcı yukarı sayma yapar.

ÖRNEK



32 bit yukarı / aşağı sayıcı kullanarak aşağıdaki işlemleri gerçekleştirecek devreyi yapınız.

Sayıcı değeri -10'a ulaştığında çıkış (Y2) aktif olacak. Ayarlanan değerler, soldaki şekle uygun olacak.



M8200 sayma yönü ON — aşağı, OFF—Yukarı sayma

Reset komutu 32bit Yukarı / Aşağı sayıcı.

C200'ün değeri artarak -10 değerine ulaştığında sayıcı kontağı aktif olur ve Y2 çıkış verir.

UYGULAMA FAALİYETLERİ

Aşağıdaki işlem basamaklarını her bir uygulama faaliyeti için yapınız.

İşlem Basamakları	Öneriler
Ø Öncelikle verilen problemde sizden istenenleri kavrayınız.	Ø Çalışma şartlarını kafanızda canlandırabilmeniz gerekir.
Ø Hangi komut ya da komutları kullanarak nasıl bir çözüm yolu izleyeceğinize karar veriniz.	Ø Size özel bir çözüm yolu geliştirebilirsiniz.
Ø Çözümünüzü kâğıt üzerine çizerek kontrol ediniz.	Ø Önce kâğıt üzerinde çalışmak problem çözme yeteneğinizi geliştirecektir. Ø Enerji verildiği anda ne olması isteniyor? Sorusundan başlayınız.
Ø Giriş ve çıkış elemanları için gerekli kablolamayı yapınız.	Ø PLC enerjisiz olmalıdır.
Ø Merdiven diyagramını bilgisayar ortamında kurallara uygun çiziniz.	Ø Programınızı kaydetmeyi unutmayınız.
Ø Programınızı derleyerek PLC'ye yüklemeye hazır hale getiriniz.	
Ø PLC'ye programınızı yükleyiniz.	Ø PLC enerjili ve STOP konumunda olmalıdır.
Ø Giriş şartlarını sağlayarak programın çalışmasını kontrol ediniz.	Ø PLC besleme gerilimi 220 V ve giriş-çıkış ünitelerinin sinyal değeri 24 V olarak uygulama yapılmalıdır.

Uygulama- 1

Aşağıdaki işlemleri gerçekleştiren bir program yazarak uygulamasını yapınız.

1. X0 butonuna basıldığı zaman 3 sayısı D0 kaydedicisine taşınacak,
2. X1 butonuna basıldığı zaman D1 kaydedicisinin içeriği D2 kaydedicisine taşınacak,
3. X2 butonuna basıldığı zaman X10-X13 arası D3 kaydedicisine taşınacak,
4. X3 butonuna basıldığı zaman 5 sayısı Y10-Y13 arasına taşınacak.

Uygulama- 2

Aşağıdaki işlemleri gerçekleştiren bir program yazarak uygulamasını yapınız.

1. X0 butonuna basılı olduğu sürece D0'ın değeri sürekli artacak.
2. X1 butonuna basılı olduğu sürece D1'in değeri sürekli azaltacak.
3. X2 butonuna basıldığında Y0-Y17 arası binary olarak sadece ilk devirde, bir kere artacak.
4. X3 butonuna basıldığında D2'nin değeri sadece bir kere azalacak.
5. X4 butonuna basıldığında Y0-Y3 arası BCD olarak ve sadece bir kere artacak.

Uygulama- 3

Aşağıdaki işlemleri gerçekleştiren bir program yazarak uygulamasını yapınız.

1. X0 butonuna basıldığında 10 (Decimal) sabit sayısı D10 veri kaydedicisine taşınacak.
2. X1 butonuna basıldığında 20 (Decimal) sabit sayısı D10 veri kaydedicisine taşınacak.
3. X2 butonuna basıldığında 30(Decimal) sabit sayısı D10 veri kaydedicisine taşınacak.

4. X3 butonuna basıldığında D10 ile 20 sabit sayısı karşılaştırılacak. Eğer;

D10>20(decimal) ise Y2 ON,

D10=20(decimal) ise Y4 ON,

D10<20(decimal) ise Y6 ON olacaktır.

Uygulama- 4

Bu uygulama sayıcı ve karşılaştırma komutunun kullanımına yöneliktir.

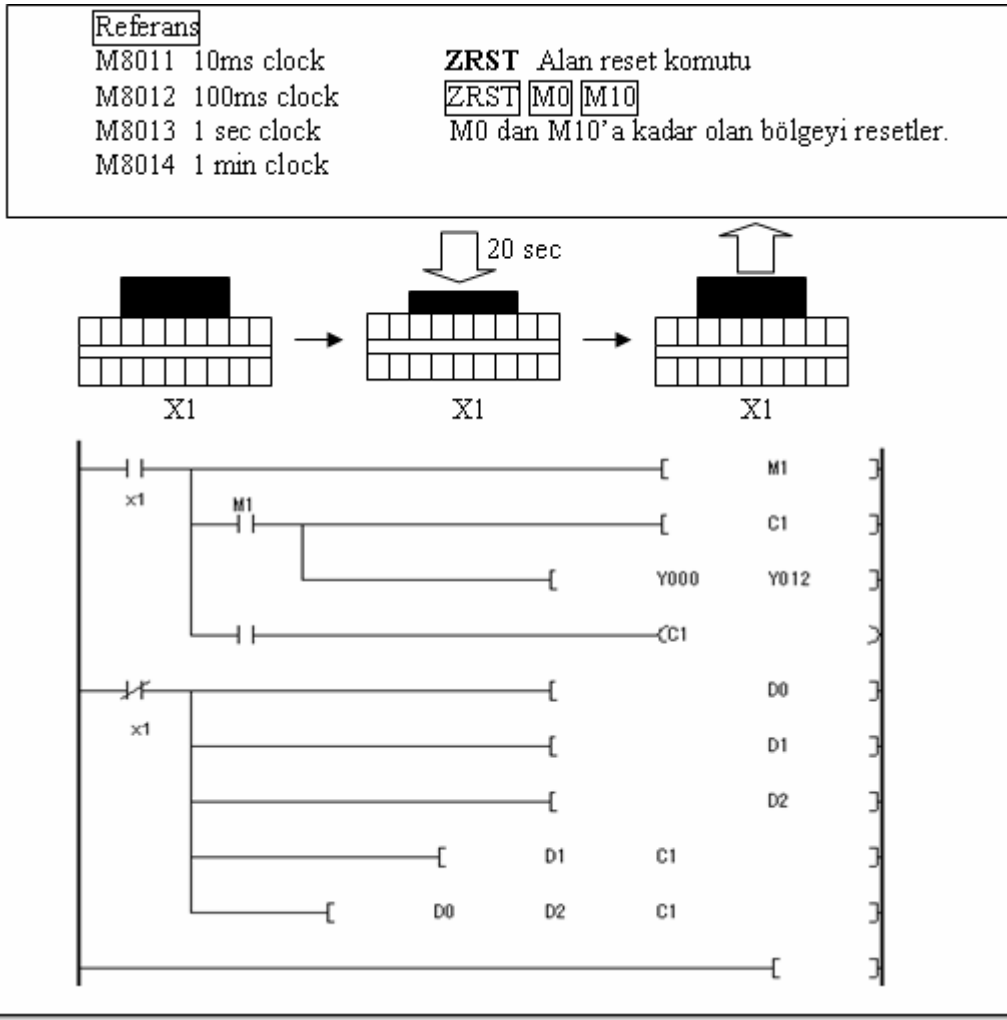
Aşağıdaki işlemleri gerçekleştiren bir program yazarak uygulamasını yapınız.

1. X1'e basıldığında zaman sayıcı saymaya başlayacak.
2. Yaklaşık 20 saniye sonra X1 butonu serbest bırakılacak (Süre tahmini olacak).

3. 20 saniye ile sizin tahmini sürenizi sayan zaman sayıcının içeriği **Karşılaştırma komutu** ile karşılaştırılacak (Çıkış, Y0-Y2).

19 saniyeden 21 saniyeye kadar olan zaman aralığı ile zaman sayıcının içeriğini **Alan Karşılaştırma** komutunu kullanarak karşılaştırınız (Çıkış Y10-Y12).

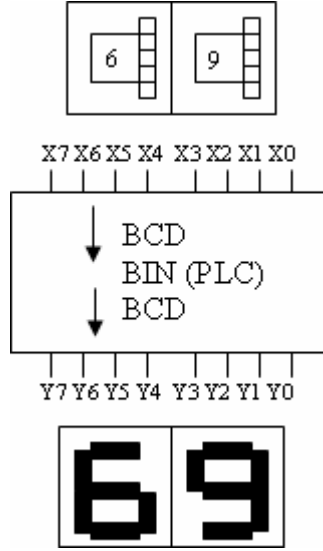
60 saniye sonra zaman sona erecek.



Uygulama- 5

Bu uygulama BCD kod ile binary sayıların PLC üzerinde kullanımı ve birbirine dönüştürülmesine yöneliktir.

Dijital ya da thumbwheel anahtar ve seven segment display kullanarak aşağıdaki şartları sağlayan programı yazınız, varsa uygun eğitim seti üzerinde uygulamasını yapınız.



1. Dijital anahtarın değerini displaye çevirecek programı yapınız (2 basamaklı).
2. X10'a basıldığında büyük sayı D1'e yerleştirilecek. X11'e basıldığında küçük sayı D2'ye yerleştirilecek.
3. X12'ye basıldığında D1 ile D2 toplanacak ve sonuç displayde gösterilecek.
4. X13'e basıldığında büyük sayıdan (D1), küçük sayı (D2) çıkarılacak ve sonuç displayde gösterilecek.
5. X14'e basıldığında küçük sayıdan (D2), büyük sayı (D1) çıkarılacak ve sonuç displayde gösterilecek.
6. X15'e basıldığında D1 ile D2'nin çarpımı yapılacak ve displayde gösterilecek.
7. (6)X16'ya basıldığında D1'in içeriği D2'ye bölünecek ve displayde gösterilecek. (Bölüm sonucu displayin onlar basamağında, kalan ise birler basamağında gösterilsin.)
8. (7)X17'ye basıldığında D2'nin içeriği D1'e bölünecek ve sonuç aynı şekilde displayde gösterilecek.

Uygulama- 6

Bu uygulama sayıcı kullanımına yöneliktir.

Aşağıdaki işlemleri gerçekleştiren bir program yazarak uygulamasını yapınız.

1. X1'e her basıldığında sayıcı değeri bir artacak,
2. Sayma değeri 5'e ulaştığında sayıcı kontağı aktif olacak, Y0 çıkışı verecektir.
3. X0'a basıldığında sayıcı değeri resetlenecektir.

Aşağıdaki işlemleri gerçekleştiren bir program yazarak uygulamasını yapınız.

Uygulama 6'ya aşağıdaki istenenleri ekleyiniz.

1. X2'ye basılı tutulduğunda programın her devrinde C1 sayıcısının değeri bir artacak.
2. Sayıcının değeri, ön değere (K10) ulaştığında, Y1 çıkışı aktif olacak.
3. X0'a basıldığında sayıcı değeri resetlenecektir.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

1. 10011111_2 sayısının karşılığı hangisidir?
A) 95_{10}
B) $9F_{16}$
C) $8E_{16}$
D) 95_{BCD}
2. 41_{10} sayısının karşılığı hangisidir?
A) 10000001_{BCD}
B) 01010001_{BCD}
C) 00010100_{BCD}
D) 01000001_{BCD}
3. D3'e 15 yazmak için hangi komut satırı yazılmalıdır?
A) MOV K15 D3
B) MOV D3 K15
C) MOV K1X15 D3
D) MOV D15 D3
4. 32 bit taşıma işlemi için hangi komut kullanılmalıdır ?
A) MOV
B) DMOV
C) MOVP
D) MOV K1X32
5. **BCD D0 K2Y000** komutu hangi uygulamayı yapar?
A) D0'ı BIN kodunda Y0-Y2 arasına yazar
B) D0'ı BCD kodunda Y0-Y2 arasına yazar
C) D0'ı BCD kodunda Y0-Y7 arasına yazar
D) D0'ı BIN kodunda Y0-Y7 arasına yazar
6. **CMP K15 D4 Y000** komutu uygulandığında D4 içinde 16 varsa hangi çıkış ON olur?
A) Y2
B) Y1
C) Y0
D) HİÇBİRİ
7. **DIV K20 K3 D2** komutu uygulanırsa D3 içinde hangisi vardır ?
A) 6
B) 2
C) 0
D) 3

8. Hangisi 10ms’lik zamanlayıcıdır ?

- A) T5
- B) T205
- C) T255
- D) T55

Değerlendirme

Cevaplarınızı cevap anahtarı ile karşılaştırınız. Doğru cevap sayınızı belirleyerek kendinizi değerlendiriniz. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt yaşadığınız sorularla ilgili konuları faaliyete geri dönerek tekrar inceleyiniz.

MODÜL DEĞERLENDİRME

Performans Testi (Yeterlik Ölçme)

Modülde yaptığınız uygulamaları tekrar yapınız. Yaptığınız bu uygulamaları aşağıdaki tabloya göre değerlendiriniz.

Açıklama: Aşağıda listelenen kriterleri uyguladıysanız “Evet” sütununa, uygulamadıysanız “Hayır” sütununa X işareti yazınız.		
Değerlendirme Ölçütleri	Evet	Hayır
Bilgisayar ortamında merdiven diyagram çizip derleyebildiniz mi?		
Derlenmiş bir programı PLC’ye yazdırabildiniz mi?		
İstenen şartları sağlayacak kablolama yapabildiniz mi?		
Temel kontrol komutlarıyla program yazabildiniz mi?		
BIN sayıyı PLC üzerinde BCD’ye çevirebildiniz mi?		
PLC üzerinden bir giriş elemanı ile bir çıkış elemanının kesik ya da sürekli kontrolünü yapabildiniz mi?		
32 bit veri taşıma işlemi yapabildiniz mi?		
İki farklı yazmaç içindeki verileri karşılaştırarak hangisinin büyük olduğunu belirleyebildiniz mi?		
Bir butona basılı tutulduğu sürece bir yazmaç içindeki verinin yalnızca bir kez artmasını sağlayabildiniz mi?		
Sabit sayılar ,yazmaçlar içindeki veriler ya da giriş uçlarındaki değerler arasında dört işlem yapabildiniz mi?		
Bir giriş şartı sağlandıktan belirli bir zaman sonra çıkış veren program yazabildiniz mi?		
Çalışma esnasında enerji kesilip tekrar gelse bile zamanın kaldığı yerden devam etmesini sağlayan program yazabildiniz mi?		
PLC girişine bağlı anahtarları kullanarak zamanlayıcı değerini ayarlayabildiniz mi?		
Bir giriş şartının belirli sayıda tekrarlanması durumunda çıkış şartlarını sağlayan program yazabildiniz mi?		
Sayma işlemini enerji kesilip tekrar geldiğinde kaldığı yerden devam ettiren program yazabildiniz mi?		

Değerlendirme

“Hayır” cevaplarınız var ise ilgili uygulama faaliyetini tekrar ediniz. Cevaplarınızın tümü evet ise bir sonraki modüle geçebilirsiniz.

CEVAP ANAHTARLARI

Öğrenme Faaliyeti 1 Objektif Testler (Ölçme Soruları) Cevap Anahtarı

1.	B
2.	A
3.	D
4.	B
5.	C
6.	B
7.	D
8.	A
9.	C

Öğrenme Faaliyeti 2 Objektif Testler (Ölçme Soruları) Cevap Anahtarı

1.	B
2.	D
3.	A
4.	B
5.	C
6.	A
7.	D
8.	B

KAYNAKÇA

- Ø Tetsuya OKUBO, Kahraman ÖNEY, **Ardışık Kontrol Teknolojisi Ders Kitabı**, M.E.B - JICA – Ağustos 2005.
- Ø **MITSUBISHI Programlama Kataloğu**
- Ø www.nagoya.melco.co.jp
- Ø Selim GÜLÇEN, **PLC Ders Notları**, JICA-2006.