

HAZIR BETON SANTRALİ OTOMASYONU

Gizem BEKAR

**YÜKSEK LİSANS TEZİ
ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ BÖLÜMÜ**

**GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ**

EKİM 2009

ANKARA

Gizem BEKAR tarafından hazırlanan HAZIR BETON SANTRALİ OTOMASYONU adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Prof. Dr. Nihal Fatma GÜLER
Tez Yöneticisi

Bu çalışma, jürimiz tarafından oy birliği / oy çokluğu ile Elektronik Eğitimi Anabilim Dalında Yüksek lisans tezi olarak kabul edilmiştir.

Başkan : Yrd. Doç. Dr. Mustafa BURUNKAYA _____

Üye(Danışman): Prof. Dr. Nihal Fatma GÜLER _____

Üye : Yrd. Doç. Dr. Derya YILMAZ _____

Tarih :/...../.....

Bu tez, Gazi Üniversitesi Bilişim Enstitüsü tez yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Gizem BEKAR

HAZIR BETON SANTRALİ OTOMASYONU**(Yüksek Lisans Tezi)****Gizem BEKAR****GAZİ ÜNİVERSİTESİ****BİLİŞİM ENSTİTÜSÜ****Ekim 2009****ÖZET**

Bu çalışmada gerçekleştirilen hazır beton santrali otomasyonunda röle, kontaktör gibi elektromekanik devre elemanlarıyla yapılan kontrol sistemleri yerine daha basit, ekonomik, hızlı, güvenilir ve sistemin tüm ihtiyaçlarına cevap verebilen bir sistem tasarlanmış ve gerçekleştirilmiştir. Hazır beton santrali birimlerinin fiziksel olarak çok büyük olması ve kullanılan hammaddelerin ölçümlerinin hassas bir şekilde yapılması gerektiğinden ölçme ve kontrol işlemleri insandan mümkün olduğunca bağımsız çalışır hale getirilmelidir. Ayrıca hazır beton santralinin fiziksel ortamının elektriksel gürültü ve toz gibi kötü şartlara sahip olması otomasyonun PLC ile gerçekleştirilmesiyle iyileştirilmeye çalışılmıştır. PLC ile yapılan bu otomasyon sistemiyle hazır beton santralinin üretim sürecinin sürekliliği sağlanmış ve sistem daha basit hale getirilmiştir.

Bilim Kodu : 704.3.013**Anahtar Kelime : Hazır beton santrali, Otomasyon****Sayfa Adedi : 115****Tez Yöneticisi : Prof. Dr. Nihal Fatma GÜLER**

READY MIXED CONCRETE PLANT AUTOMATION**(M.Sc. Thesis)****Gizem BEKAR****GAZI UNIVERSITY****INFORMATICS INSTITUTE****October 2009****ABSTRACT**

In this work ready mixed concrete plant automation, instead of control system which made with elektromechanic circuit component such as , relay, contactor, aim to projected a system which is easier, much economic, faster, safety and cover all of the system needs. Ready mixed concrete plants are very huge in physical area and row material measurements must be realized very sensitive. Because of this reason human do not have much responsibility as possible as, PC and automation have much responsibility for very sensitive measurement. On the other hand physical conditions have too much electrical noise and working conditions are very dusty, because of that automation made with PLC. Automation system which made with PLC, ready mixed concrete plant's manufacturing term become continuous, ready mixed concrete plant become easier for use.

Science Code : 704.3.013**Key Words : Ready mixed concrete plant, Automation****Page Number : 115****Adviser : Prof. Dr. Nihal Fatma GÜLER**

TEŐEKKÜR

Bu tezin hazırlanmasında yapıcı uyarı ve destekleri ile beni yönlendiren danışmanım Sayın Prof. Dr. Nihal Fatma GÜLER' e, çalışmalarımı sürekli destekleyen ve teşvik eden ailem ile benden yardımlarını ve desteğini hiç esirgemeyen sevgili eşim Atakan BEKAR' a teşekkürü bir borç bilirim.

Ekim 2009

Gizem BEKAR

İÇİNDEKİLER

	Sayfa
ÖZET.....	iv
ABSTRACT.....	v
İÇİNDEKİLER.....	vi
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	x
KISALTMALARIN LİSTESİ.....	xii
1. GİRİŞ.....	1
2. HAZIR BETONUN YAPISI.....	3
2.1. Beton.....	3
2.1.1. Betonun tanımı.....	3
2.1.2. Betonun oluşturan malzemeler.....	3
2.1.2.1. Çimento.....	3
2.1.2.2. Agregası.....	4
2.1.2.3. Karma suyu.....	5
2.1.2.4. Katkı maddeleri.....	5
2.2. Hazır Beton.....	6
2.3. Hazır Beton Santrali.....	6
2.3.1. Agregası ünitesi.....	6
2.3.1.1. Ventil ünitesi.....	7
2.3.1.2. Vibratör ünitesi.....	7
2.3.1.3. Nem ölçer.....	7
2.3.2. Tartı Bandı.....	7
2.3.2.1. Bant.....	7
2.3.3. Aktarma Bandı.....	9
2.3.4. Çimento Silosu.....	9
2.3.4.1. Helezon.....	9
2.3.5. Su Bunkeri.....	9
2.3.6. Katkı Bunkeri.....	10
2.3.7. Mikser.....	10

	Sayfa
2.3.8. Kompresör.....	10
3. PLC (PROGRAMLANABİLİR LOJİK KONTROL CİHAZI).....	12
3.1. Programlanabilir Lojik Kontrolör	12
3.2. Bilgisayar ile PLC Arasındaki Fark	14
3.3. PLC Mantığı ile Kontaklı Devre Mantığı Arasındaki Fark	15
3.4. PLC'nin İç Yapısı	17
3.4.1. Giriş birimi	20
3.4.2. Çıkış birimi.....	20
3.4.3. Diğer birimler	21
3.4.4. Programlayıcı birimi	22
3.5. Çalışma Birimi	22
3.5.1. Kullanıcı programının yürütülmesi	22
3.6. Programlama Dilleri.....	25
3.7. Programlama	26
3.7.1. Programlama yapıları	26
3.7.1.1. Doğrusal programlama.....	27
3.7.1.2. Yapısal programlama	28
3.7.2. MDG ile programlama tekniği.....	30
3.7.3. LKG ile programlama tekniği	32
3.7.4. Komut ile programlama	33
4. TASARLANAN OTOMASYON SİSTEMİ.....	36
4.1. Tasarlanan PLC Giriş Kartı.....	40
4.2. Tasarlanan PLC Çıkış Kartı	42
5. SONUÇ VE ÖNERİLER	45
KAYNAKLAR	47
EK-1	49
EK-2	105
ÖZGEÇMİŞ	115

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 1.1. Delta marka PLC ile diğer marka PLC'lerin Fiyat Karşılaştırması.....	1
Çizelge 3.1. Komutların Çeşitli PLC lerdeki karşılıkları.....	34
Çizelge 4.1. Birimlerin Elektrik Parametreleri	37
Çizelge 4.2. Beton Santrali Girdi Çıktıları.....	37

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Hazır Beton Santrali Şeması	11
Şekil 3.1. Kontaklı Kumanda Devresi	16
Şekil 3.2. Ladder Diyagramı	16
Şekil 3.3. PLC nin Yapısı.....	18
Şekil 3.4. PLC nin Blok Diyagramı	19
Şekil 3.5. 220V AC Gerilimle Uyarılan Bir Giriş Birimi Devresi.....	20
Şekil 3.6. Kontaktör Sürülen Röleli Çıkış Biriminin Bir Çıkış Noktasına İlişkin Devre	21
Şekil 3.7. PLC nin İşlev Evreleri	23
Şekil 3.8. $Y1=(X1+X2)*X3$ İfadesinin Komut, Ladder ve Lojik kapı gösterimi	26
Şekil 3.9. Doğrusal Programlama	27
Şekil 3.10. Yapısal Programlama.....	29
Şekil 3.11. Örnek Bir Kumanda Devresi	31
Şekil 3.12. PLC Giriş-Çıkış Bağlantıları ve MDG Programı	31
Şekil 3.13. Lojik Kapı Şekilleri	32
Şekil 3.14. Ladder Programı	33
Şekil 3.15. LKG programı.....	33
Şekil 4.1. Hazır Beton Santrali Blok Diyagramı.....	36
Şekil 4.2. Hazırlanan PLC Programının Akış Diyagramı	39
Şekil 4.3. PLC Giriş Kartının Şeması	40
Şekil 4.4. PLC Giriş Kartının Önden Görünüşü	41
Şekil 4.5. PLC Giriş Kartının Arkadan Görünüşü	41
Şekil 4.6. Tasarlanan Giriş Kartı.....	42
Şekil 4.7. PLC Çıkış Kartının Şeması.....	43
Şekil 4.8. PLC Çıkış Kartının Blok Şeması	43
Şekil 4.9. PLC Çıkış Kartının Önden Görünüşü	44
Şekil 4.10. PLC Çıkış Kartının Arkadan Görünüşü.....	44
Şekil 4.11. Tasarlanan Çıkış Kartı	44

Şekil	Sayfa
Şekil 4.12. Tasarlanan Cihazın İçten Görünüşü.....	45
Şekil 4.13. Sistemin Çalışma Anı	46

KISALTMALARIN LİSTESİ

Kısaltma	Açıklama
NEMA.....	Ulusal Elektriksel Üretim Derneği
PLC	Programlanabilir Lojik Kontrolör
SCADA.....	Veri Tabanlı Kontrol ve Gözetleme

1.GİRİŞ

Bu çalışmada, hazır beton santralının kontaktörlü ve röleli sistemler yerine PLC ile otomasyonunun gerçekleştirilmesi amaçlanmıştır. Bu nedenle beton santralının birimleri araştırılmış ve bu birimlere uygun bir otomasyon sistemi gerçekleştirilmiştir. Piyasadaki hazır beton santrallerinde kullanılan Siemens, Omron, Panasonic ve Toshiba gibi yüksek maliyetli PLC'ler yerine bu PLC'lerle karşılaştırıldığında çok daha ucuz olan Delta marka PLC ile hazır beton santrali otomasyonunun gerçekleştirilmesi tercih edilmiştir. Çizelge 1.1'de Delta marka PLC ile diğer marka PLC'lerin firma kataloglarından alınan fiyatları verilmiştir.

Çizelge 1.1. Delta marka PLC ile diğer marka PLC'lerin Fiyat Karşılaştırması

	CPU	Dijital Çıkış Modülleri	Analog Giriş	Dokunmatik Ekran
Siemens	S7-200 CPU224	4 çıkış	8 giriş	6 inç
Fiyat	309€	112€	178€	464€
Omron	CMP1A-40CDR-D	8 çıkış	2 giriş	190x110mm
Fiyat	606€	366€	315€	1023€
Delta	DVP125A	8 çıkış	8 giriş	5.7 inç
Fiyat	107\$	52\$	110\$	230\$

Günümüz modern üretim süreçlerinde yüksek verim ve kaliteli üretim için kaçınılmaz olan endüstriyel otomasyon sistemleri her geçen gün büyük bir hızla gelişmektedir. Endüstriyel otomasyon sistemleri, en küçük üretim biriminin amaca uygun çalışmasını düzenlediği gibi, bütün üretim sistemleri arasında veri iletişimi olanağı sağlayarak daha üst düzeyde yönetim ve planlama için gerekli veri tabanının oluşturulmasına olanak sağlar [1].

PLC son yirmi yıldan beri, endüstride yaygın olarak kullanılmasına rağmen, ülkemizde son yıllarda kullanım alanları giderek artmaya başlamıştır [2].

1978 yılında dört yıllık bir çalışmanın ürünü olan NEMA (National Electrical Manufactureres Association) kuruluşu tarafından standart PLC'ler piyasaya sürülmüştür [3].

Tesislerde röle ve kontaktör gibi elektromekanik devre elemanları ile yapılan süreç kontrolleri çok yetersiz kalmaktadır. Bu gibi geleneksel sistemlerle tesislerin emniyetini, üretim hızını, kalitesini ve maliyetini düşürmeyi sağlamak neredeyse imkansızdır. Bu yüzden PLC'li kontrol sistemlerine büyük ihtiyaç duyulmuştur. PLC ile otomasyon hakkında birçok eğitim dokümanı ve tez çalışmaları mevcuttur [4-5].

Kullanılan PLC'nin sekiz girişi ve dört çıkışı, RS-232 ve RS-485 portları ile sekiz adet özel modül bağlanabilme özelliği vardır. PLC ile birlikte sekiz giriş ve sekiz transistör çıkışlı iki adet modül ve bir adet dört girişli analog modül kullanılmıştır. Kullanıcı ara yüzü olarak Delta marka endüstriyel dokunmatik ekran kullanılmıştır. Ekran ile PLC RS-485 portu ile haberleştirilmiştir.

Giriş bölümünü izleyen 2. bölümde betonun yapısı ve beton santrallerinin birimleri, 3. bölümde PLC'nin yapısı 4. bölümde de tasarlanan otomasyon sistemi incelenmiştir. 5. bölümde ise sonuç ve önerilere yer verilmiştir.

2. HAZIR BETONUN YAPISI

2.1. Beton

2.1.1. Betonun tanımı

Beton; çimento, beton agregası (kum, çakıl gibi malzemeler), su ve gerektiğinde katkı maddelerinin homojen olarak karşılaştırılması ile elde edilen başlangıçta plastik kıvamında olup zamanla çimentonun hidrasyonu sebebiyle katılaşp sertleşen kompozit bir yapı malzemesidir [6]. Beton, çağımızda irili ufaklı birçok yapıda kullanılmakta olan en yaygın ve en popüler yapı malzemesidir. Çağdaş günlük yaşamda, insanların, betondan yapılmış yapıları kullandığı ya da betonla karşılaşmadığı tek yer dahi yoktur.

Her betonda olması gereken bazı özellikler şunlardır;

- İşlenebilme özelliği
- Basınç dayanımı
- Betonun dış ortam etkilerine dayanıklılığıdır [7].

2.1.2. Betonu oluşturan malzemeler

Betonu oluşturan üç ana madde; çimento, agrega ve sudur. Ayrıca gerektiğinde katkı maddeleri de kullanılmaktadır.

2.1.2.1. Çimento

Çimento, su ile karıştırıldığında hidrasyon reaksiyonları ve işlemler nedeniyle priz alan ve sertleşen bir hamur (pasta) oluşturan ve sertleşme sonrası suyun altında bile dayanımını ve kararlılığını koruyan, inorganik ve ince öğütülmüş hidrolik bağlayıcıdır [8]. Kırılmış kalker, kil ve gerekiyorsa demir cevheri ve/veya kum katılarak öğütölüp toz haline getirilir. Bu malzeme 1400-1500 °C'de döner fırınlarda

pişirilir. Meydana gelen ürüne “klinker” denir. Daha sonra klinkere bir miktar alçı taşı eklenip (%4-5 oranında) çok ince toz halinde öğütülerek Portland Çimentosu elde edilir. Katkılı çimento üretiminde; klinker ve alçı taşı dışında, çimento tipine göre tek veya birkaçı bir arada olmak üzere tras, yüksek fırın cürufu, uçucu kül, silis dumanı vb. katılır. Çimento birçok beton karışımında hacimce en küçük yeri işgal eden bileşendir; ancak beton bileşenleri içinde en önemlisidir [9].

En çok kullanılan çimento tipleri ve sınıfları: Portland kompoze çimento, katkılı çimento ve sülfata dayanıklı çimentodur. Bunun dışında özel amaçlar için beyaz Portland çimentosu ve diğer bazı tip çimentolar kullanılmaktadır. Normal betonda agrega taneleri en sağlam unsur olduğundan, diğer iki unsur (çimento hamuru ve aderans) mukavemeti belirlemektedir. Çimento hamurunun mukavemeti önemli ölçüde su/çimento oranına da bağlıdır [8]. Beton ile donatı meydana gelen etkiler nedeniyle şekil değiştirirler. Bu sırada iki malzeme arasında gerilmelerin geçişi meydana gelir. Arada kayma olmadan bu tür gerilme geçişinin ortaya çıkmasına aderans denir. Aderans, betonarmenin en önemli faydalı özelliklerinden biridir. Bu suretle iki malzemenin beraberce kullanılması ve birbirini tamamlaması mümkün olur.

2.1.2.2. Agrega

Beton üretiminde kullanılan kum, çakıl, kırma taş gibi malzemelerin genel adına agrega denir. Beton içinde hacimsel olarak %70-80 civarında yer işgal eden agrega önemli bir bileşendir. Agregalar, tane boyutlarına göre ince (kum, kırma kum vs.) ve kaba (çakıl, kırma taş vs.) agrega olarak ikiye ayrılır. Agregalarda aranan en önemli özellikler şunlardır:

- Sert ve dayanıklı olmaları,
- Zayıf taneler içermemeleri (deniz kabuğu, odun, kömür vs.)
- Basınca ve aşınmaya karşı mukavemetli olmaları,
- Toz, toprak ve betona zarar verebilecek maddeler içermemeleri,

- Yassı ve uzun taneler içermemeleri,
- Çimentoza zararlı reaksiyona girmemeleridir.

Agreganın kirli (kil, silt, mil, toz vs.) olması aderansı olumsuz etkilemekte, ayrıca bu küçük taneler su ihtiyacını da arttırmaktadır. Beton agregalarında elek analizi, yassılık, özgül ağırlık ve su emme gibi deneyler uygun aralıklarla yapılarak kalite sürekliliği takip edilmelidir. Betonda kullanılacak agregalar TS 706 EN 12620'ye uygun olmalıdır [10].

2.1.2.3. Karma suyu

Betonun bir diğer önemli bileşeni sudur. Su, çimento ile hidrasyon reaksiyonunu yaparak karışımın sertleşmesini sağlar. Su beton dayanımında önemli bir rol oynar. TS EN 206'ya göre karma suyu TS EN 1008'e uygun olmalıdır [11]. Beton üretiminde kullanılan karışım suyunun iki önemli işlevi vardır. Bunlar;

1. Kuru haldeki çimento ve agregayı plastik, işlenebilir bir kütle haline getirmek.
2. Çimento ile kimyasal reaksiyon yaparak plastik kütlelerin sertleşmesini sağlamak.

2.1.2.4. Katkı maddeleri

Betonun özelliklerini geliştirmek üzere üretim sırasında veya dökümden önce transmikserde az miktarda ilave edilen maddelere katkı adı verilir. Katkı maddelerini kökenine göre kimyasal ve mineral katkıları olarak ikiye ayırmak mümkündür [10].

2.2. Hazır Beton

Hazır beton, çimento, doğal veya yapay agrega, su ve istenilen betonun niteliğine göre kullanılan kimyasal katkı maddelerinden oluşan ve transmikserlerle nakledilerek beton pompası ile kalıba yerleştirilen bir karışımdır. Betonun mutlak hacmini yaklaşık olarak %75 oranında agrega (kum, çakıl, mıcır), %10 oranında çimento, %15 oranında su oluşturur. Gerektiğinde, çimento ağırlığının %2' sinden fazla olmamak kaydıyla, katkı malzemesi ilave edilebilir [7].

Hazır betonu, şantiyede elle ya da betoniyerlerle karıştırılarak hazırlanan betondan ayıran temel unsur, hazır betonun modern tesislerde, bilgisayar kontrolüyle üretilmesidir. Hazır beton kullanıcısının hazır betonda arayacağı nitelikler TS EN 206'da yer almaktadır. Hazır beton üretiminin su ölçme ve karıştırma işlemlerinin santralde veya transmikserde yapılmasına göre iki farklı şekli bulunmaktadır:

- Kuru Sistem
- Yaş Sistem

Kuru karışimli hazır beton, agrega ve çimentosu beton santralinde ölçülüp santralde veya transmikserde karıştırılan, suyu ve varsa kimyasal katkısı ise teslim yerinde ölçülüp karıştırılarak ilave edilen hazır betondur. Kuru karışimli hazır betonda şantiyede karışıma verilen su miktarına ve karıştırma süresine (homojen bir karışım için yeterli süre) özel itina gösterilmesi gerekmektedir. Yaş karışimli hazır beton, su dahil tüm bileşenleri beton santralinde ölçülen ve karıştırılan hazır betondur [11].

2.3. Hazır Beton Santrali

2.3.1. Agrega ünitesi

Betonu oluşturan malzemeler içerisinde en büyük orana (yaklaşık % 75) sahip olan agrega doğal kaynakları giderek tükenen ve standartlara uygun, temiz, kaliteli örneklerinin bulunması güç bir malzeme olarak, hazır beton sektöründeki stratejik

önemini her geçen gün artmaktadır. Agrega bunkerinde ventil, nem ölçer ve vibratör birimleri bulunur [10].

2.3.1.1. Ventil ünitesi

Elektrikle havaya yön vererek pistonu kontrol eder. Piston kapaklara bağlıdır ve hava ile ya da hidrolik olarak kapakların açılıp kapanmasını sağlar.

2.3.1.2. Vibratör ünitesi

Titreşim sağlayarak malzemenin kapaklardan daha düzenli akmasını ve nemden dolayı yüzeye yapışan kumların dökülmesini sağlar. Merkez kaç kuvveti ile çalışır.

2.3.1.3. Nem ölçer

Kumdaki suyun oranını ölçer ve PLC' ye ölçme bilgisini gönderir. PLC bu bilgiye bağlı olarak su bunkerinden miksera ne kadar su katılacağına karar verir.

2.3.2. Tartı Bandı

Kum, çakıl gibi malzemeler tartı bandında tek tek tartılarak aktarma bandına taşınır. Tartı bandı motor, redüktör ve banttandır. Tartı bandındaki motor bandı çevirmek için kullanılır 1500 devirle döner. Redüktör ise devri 12 de 1 oranında düşürür. Tartma işlemi loadcell kullanılarak gerçekleştirilir.

2.3.2.1. Bant

Malzemeyi nakleden bant, tamburlardan intikal eden çekme kuvvetlerine ve malzemedan ileri gelen ağırlık, sıcaklık ve aşınma etkilerine dayanıklı olmalıdır. Arızasız bir çalışma için bir konveyör bandının aşağıdaki özelliklere sahip olması istenir:

- Az nem çekme özelliği
- Yüksek mukavemet
- Düşük özgül ağırlık
- Az uzama
- Tamburlara sarılma ve oluklaşmadan dolayı meydana gelen eğilme tesirlerine mukavemet
- Eğrilik değişmelerinden meydana gelen alternatif gerilmelere mukavemet
- Malzemenin aşındırıcı tesirlerine mukavemet ve uzun ömür
- Özel şartlar: Sıcaklığa, soğuğa, iklim etkilerine, yağa, oksitlenmeye, ozona karşı mukavemet; statik elektrik toplamamak gibi özellikler.

Orta kısmında mukavemeti temin eden pamuk veya naylon gibi sentetik malzemeden yapılmış birkaç tabakalı bir dokuma bulunan lastik bantlar, yukarıdaki şartlara en iyi uyduklarından dolayı geniş bir tatbikat alanı bulmuşlardır. Bantlarda dokuma tabakalarının arası, üst, alt ve yanlar lastikle kaplanır.

Bandın ortasındaki dokuma tabakalarının hepsine birden karkas adı verilir. Karkas banttaki çekme kuvvetini nakleder ve bandın yük taşımalarını sağlar. Karkas tabakaları arasındaki lastik, bu tabakalarını birbirine bağlar ve aynı zamanda tabakaları su geçirmez hale getirir. Bandın alt yüzündeki lastik ise bandın makaralar üzerinde yükün uyguladığı darbe sonucu, ezilmesine engel olur, banda pisliklerin nüfuz etmesini önler ve tamburdan aldığı tahrik kuvvetini karkasa nakleder. Üst veya taşıyıcı yüz ise daha kalın ve daha yüksek evsafli bir lastik tabakası olup özellikle yüklenme bölgesinde malzemenin darbe etkilerine karşı bandı korur. Lastik kaplama kalınlığı 0.8 ile 12mm arasında değişir. Bazen 12 mm den kalın kaplamalar da kullanılır.

Çalışma şartlarının ağır olması halinde veya üst lastik kaplamanın aderansını arttırmak için birinci dokuma tabakasının üzerine ve üst kaplamanın arasına gevşek dokulu bir darbe tabakası yerleştirilir. Bu türlü yapılan birleşim çok kuvvetli olur ve

üst kaplamayı karkastan ayırmak, pratik olarak imkansız hale gelir böylece bantta bir yırtılma meydana geldiği zaman fazla büyümez [12].

2.3.3. Aktarma Bandı

Tartı bandından gelen malzemeyi mikser veya mikser üzerindeki ara bunkere bırakır. Aktarma bandı tartı bandı gibi motor, redüktör ve banttan oluşur. Buradaki motor bandın uzunluğuna ve açısına bağlıdır.

2.3.4. Çimento Silosu

Çimento, çimento silosunda depolanır ve helezon vasıtasıyla çimento tartım bunkerine gönderilir. Çimentoyu helezonla çekebilmek için siloda şoklama ve FRY bulunmaktadır.

Şoklama siloya hava vererek silonun içinde kalan çimentoyu boşaltmaya yarar. FRY ise filtre regülatör yağlayıcıdır. Filtre hava verirken çimentoya karışan suyu tutar. Regülatör ise havanın kaç barda verileceğini ayarlar ki bu değer genelde 2 bardır. Çevreyi kirletmemek için silonun üzerinde filtre emniyet klepesi vardır.

2.3.4.1. Helezon

Helezon çimento silosundaki çimentoyu çimento tartım bunkerine taşır. Helezonun motoru helezonun çapı ve boyuna göre değişir.

2.3.5. Su Bunkeri

Su deposunda saklanan su, hidrofor vasıtasıyla basınçlı şekilde gelerek loadcelllerle tartılır. İstenen değere ulaşıncı da klepeye (aktüatör vana) komuta ederek su bunkerinin dolmasını sağlar ve burada bekler [6].

2.3.6. Katkı Bunkeri

Betonun özelliklerini geliştirmek üzere üretim sırasında veya dökümden önce transmiksere az miktarda ilave edilen maddelere katkı adı verilir. Katkı maddelerini kökenine göre kimyasal ve mineral katkılar olarak ikiye ayırmak mümkündür: Yapılacak betonun cinsine göre kullanılacak olan kimyasal katkı malzemesi, katkı deposundan pompalar vasıtasıyla bırakılır ve loadcell ile tartılır. Tartılan katkı, katkı bunkerine doldurulur ve burada bekletilir ya da katkı su bunkerine aktarılır ki bu daha sağlıklıdır. Çünkü karışım daha homojen olur.

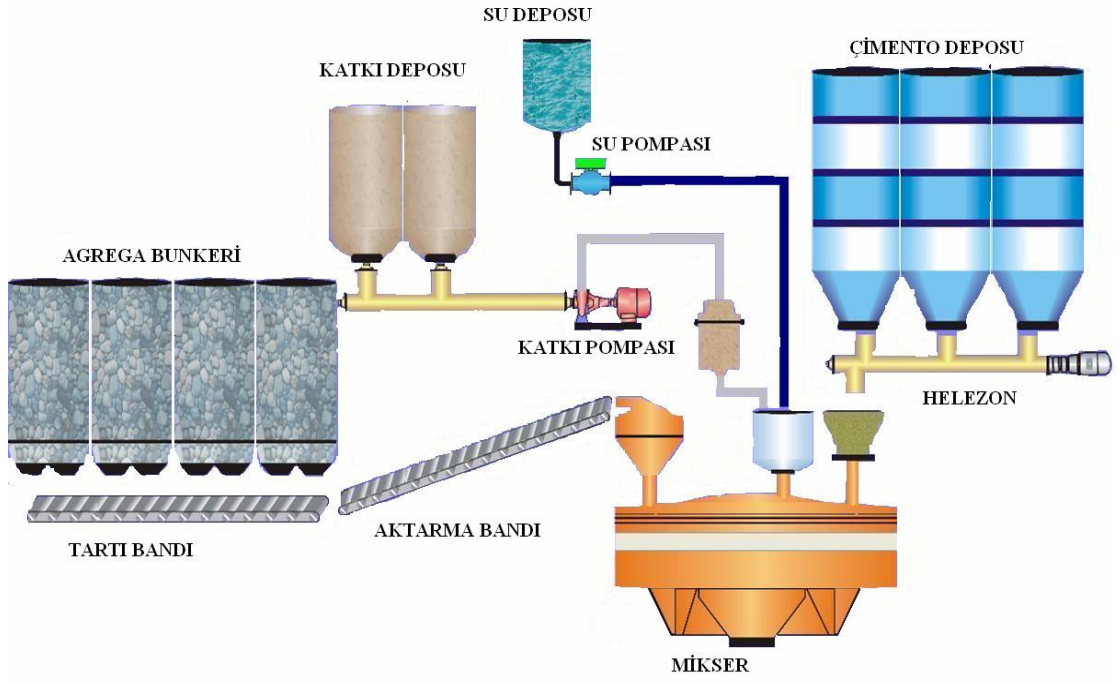
2.3.7. Mikser

Tartım bunkerinde, ara bunkerlerde bulunan çimento, su, katkı ve agregayı boşaltmak için klepeler bulunur. Bu klepelerde mekanik olarak ve hava ile çalışan sürücüler bulunur. Sürücüler de selenoid ventillerdir. Beton için gerekli tüm malzemeler kendi birimlerine dolduktan sonra otomasyonun vereceği komutla agrega takribi 2 sn arayla da sırasıyla katkı, su ve çimento bunkerleri kapaklarını açarak malzemelerini mikserine dökerler. Mikserin tipine göre karıştırma süreleri değişken olup ortalama 25 sn - 75 sn arasındadır. Karışan beton hidrolik ve pnömatik kapaklar açılarak transmiksere dökülür.

2.3.8. Kompresör

Sisteme gerekli olan hava basıncını sağlar. Agregası, mikser, su bunkerleri gibi yerlerdeki kapakların açılmasında kullanılan pnömatik sistemlerde kullanılır.

Şekil 2.1.'de hazır beton santralinin şeması görülmektedir.



Şekil 2.1. Hazır Beton Santrali Şeması

3. PLC (PROGRAMLANABİLİR LOJİK KONTROL CİHAZI)

Günümüz modern üretim süreçlerinde yüksek verim ve kaliteli üretim için kaçınılmaz olan endüstriyel otomasyon sistemleri her geçen gün büyük bir hızla gelişmektedir.

Endüstriyel otomasyon sistemleri tasarım açısından üç bölüm altında incelenebilir: Endüstriyel kumanda sistemleri, geri beslemeli kontrol sistemleri ve veri iletişim sistemleri. Endüstriyel kumanda sistemleri, en küçük üretim birimlerinin çalışma koşullarını (devreye girme ve devreden çıkma gibi) düzenleyen lojik temelli sistemlerdir. Geri beslemeli kontrol sistemleri, çeşitli üretim süreçlerinin her türlü bozucu etkiye karşı, sürecin istenen değerlerde çalışmasını sağlayan sistemlerdir. Veri iletişim sistemleri ise birimler arasında bilginin güvenilir ve hızlı akışını sağlayan donanım ve yazılım sistemleri olup bu amaçla günümüzde yaygın olarak SCADA (Supervisory Control and Data Acquisition) yazılımları kullanılır. Programlanabilir lojik kontrolör (PLC), günümüz endüstriyel otomasyon sistemlerinin her üç bölümünde de önemli işlevler yüklenen en önemli elemandır [1].

3.1. Programlanabilir Lojik Kontrolör

PLC endüstriyel otomasyon sistemlerinin kumanda ve kontrol devrelerini gerçeklemeye uygun yapıda giriş-çıkış birimleri ve iletişim arabirimleri ile donatılmış, kontrol yapısına uygun bir sistem programı altında çalışan bir endüstriyel bilgisayardır. Başlangıçta, röleli kumanda sistemlerinin yerine kullanılması düşünülmüş ve ilk ticari PLC, 1969 yılında Modicon firması tarafından geliştirilmiştir. O yıllarda, röleli kumanda devreleri yerine kullanılmak üzere geliştirilen bu cihaz yalnız temel lojik işlem komutları içerdiğinden programlanır (programlanabilir) lojik kontrolör adı ile sunulmuştur. İlk ticari PLC'nin endüstride başarıyla uygulanmasından sonra, Allen-Bradley, General Electric, GEC, Siemens ve Westinghouse gibi firmalar orta maliyette yüksek performanslı PLC'ler üretmişlerdir. Mitsubishi, Omron ve Toshiba gibi firmaların ucuz maliyette yüksek performanslı

PLC'lerin geliřtirmesinden sonra, bu aygıtlar endüstriyel otomasyon devrelerinde yaygın olarak kullanılmaya başlanmıştır.

Günümüzde üretilen PLC'ler ise lojik temelli işlemlere ek olarak aritmetik ve özel matematiksel işlemlerin yapılmasını sağlayan komutları da içerirler. Komut kümesinin genişlemesi sonucu daha karmaşık kumanda ve kontrol işlevleri gerçekleştirilmektedir. Bu kontrolörlerin geri beslemeli kontrol devrelerinde kullanılmaya başlanması, alışlagelmiş PLC adını tartışılmasına neden olmuştur. Birçok üretici firma, bu kontrolörlerin hem lojik temelli kumanda devrelerinde hem de geri beslemeli kontrol sistemlerinde kullanılmasını nedeniyle PLC yerine, programlanabilir ya da programlanır kontrolör adını kullanmayı daha uygun bulmuş ve kişisel bilgisayardan ayırma amacıyla, kısaca PCs olarak tanımlamıştır. Bazı üretici firmalar ise ilk sunulduğu andaki "programmable logic controller" kelimelerinin baş harflerinden oluşan PLC adını kullanmayı sürdürmektedirler. Yurdumuzda da bu aygıt yaygın olarak PLC olarak anılır [1].

PLC'lerin en yaygın olarak kullanıldığı alanlar, endüstriyel otomasyon sistemlerinin kumanda devreleridir. Bilindiği gibi kumanda devreleri, yardımcı röle veya kontaktör, zaman rölesi ve sayıcı gibi elemanlarla gerçekleştirilen devrelerdir. Günümüzde bu tür devrelerin yerlerini aynı işlevi sağlayan PLC'li kumanda sistemleri almıştır. Küçük boyutlu birkaç PLC modeli dışında, yeni üretilmekte olan bütün PLC modellerinde, bir kontrol algoritması yazmak için gerekli aritmetik işlemlerin yapılmasını sağlayan komut desteği bulunur. Bir PLC'nin geri beslemeli kontrol sistemlerinde sayısal kontrolör olarak kullanılması, analog işaretleri sayısal işaretlere (ADC) ve sayısal işaretleri analog işaretlere dönüştüren (DAC) giriş-çıkış birimleri ile sağlanır [13]. Endüstriyel otomasyon devrelerinde programlanabilir kontrol cihazlarının tercih edilme sebepleri ve avantajları şu şekilde sıralanabilir;

- Kumanda devresinin görevi yazılımla sağlandığından kumanda devresini tasarlamak kontaklı (röleli) bir devrenin tasarımından daha kolaydır.
- Bütün kumanda fonksiyonları yazılımla gerçekleştirildiğinden farklı bir uygulama için adaptasyon kolaydır.

- Kumanda devrelerine göre çok az yer kaplar.
- Güvenilirliği yüksek bakımı kolaydır.
- Bilgisayarla ve diğer kontrol cihazları ile haberleşme imkanları vardır. Bu özelliği bilgisayarlı otomasyona da imkan sağlar.
- Arıza ihtimali düşüktür, arızayı bulmak kolaydır.
- Kötü çevre koşullarında, özellikle tozlu ortamlarda röleli kumanda devrelerine göre daha güvenlidir.
- Parça adedinde azalma sağlar. Cihaz çok sayıda rölenin yerine kullanılabilir. Kumanda sisteminin fiziksel hacminde önemli bir azalma sağlar. Mekanik hareket içermemesi kumanda sisteminin güvenilirliğini artırır.
- Daha az kablo kullanılır. Yalnızca girişler, veri algılama cihazları ve çıkışlar ile şalt cihazlar arasında kablo bağlantısına gerek vardır.
- Kurulma kolaylığı sağlar, kablo bağlantısını değiştirmeye gerek kalmaksızın röleler ve kabloların yerine kullanılan programı değiştirmek yeterlidir.
- Kullanım esnekliği sağlar. Karmaşık sistemler ile özel makine kontrollerinde binaların enerji kontrolü vs. gibi uygulamalarda kolaylıkla kullanılabilir.
- İnsan makine diyalogunda kolay iletişimin yapılması mümkündür. Bu amaçla giriş çıkışların durumunu gösteren ledler mevcuttur. Ayrıca program ve ayar terminalleri aracılığıyla bilgi alışverişi yapılabilir.

Bunlardan başka PLC kumanda devresi tasarımını daha çabuk gerçekleştirdiğinden bu konuda çalışan teknik elemanlara zaman ve emek tasarrufu sağlar.

3.2. Bilgisayar ile PLC Arasındaki Fark

PLC'nin merkezi işlem ünitesinde mikroişlemci veya mikrokontrolcü ünite bulunur. Bu yüzden her PLC basit bir bilgisayardır. Fakat her bilgisayar bir PLC değildir. PLC'ler üretimin yapıldığı tozlu, kirli ve elektriksel gürültü gibi ağır şartlarda çalışacak şekilde tasarımı yapılmıştır. Bununla birlikte farklı bir programlama dili arıza bulma ve bakım kolaylıklarının olması gibi özellikleri ile sanayi uygulamalarında bilgisayarlardan farklıdır. Bilgisayarların arıza ve bakım servisi ile

programlama dillerinin öğrenilmesi için özel bir eğitime gerek vardır. PLC programlama dili klasik kumanda devrelerine uygunluk sağlayacak şekildedir. Bütün PLC'lerde hemen hemen aynı olan AND, OR, NOT (VE, VEYA, DEĞİL) gibi boolean ifadeleri kullanılır. Programlama klasik kumanda sistemini bilen birisi tarafından kolayca yapılabilir.

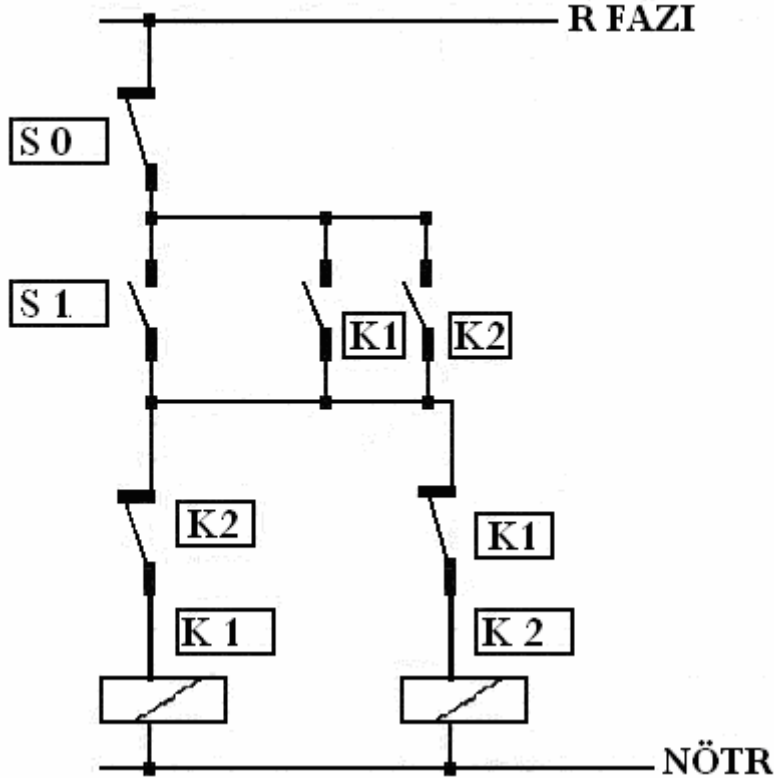
Büyük çaplı kontrol sistemleri için bilgisayarın mikroşlemcilerin kullanılması on adet röle kontaktör elemanından daha az eleman gerektiren kontrol devrelerinde de klasik kumanda devrelerinin kullanılması ekonomik açıdan daha avantajlı ve gereklidir. Sonuç olarak; küçük ve orta büyüklükte her türlü kumanda sisteminde küçük yapıları yüksek güvenilirlikli ve değişebilir beyin olarak PLC'ler otomasyon üretiminin vazgeçilmez bir elemanı olmuşlardır. PLC cihazı girişten alınan bilgi ve komutları işler.

Giriş sinyallerine karşılık çıkış sinyallerinin iletimi PLC'de yazılı olan programa bağlıdır. Selenoid valf sinyal lambası, röle, kontaktör gibi küçük yükler PLC tarafından direkt olarak sürülebilir. Fakat büyük kapasiteli selenoid valf üç fazlı motor gibi yükler kontaktör veya röle üzerinden sürülmelidir.

3.3. PLC Mantığı ile Kontaktlı Devre Mantığı Arasındaki Fark

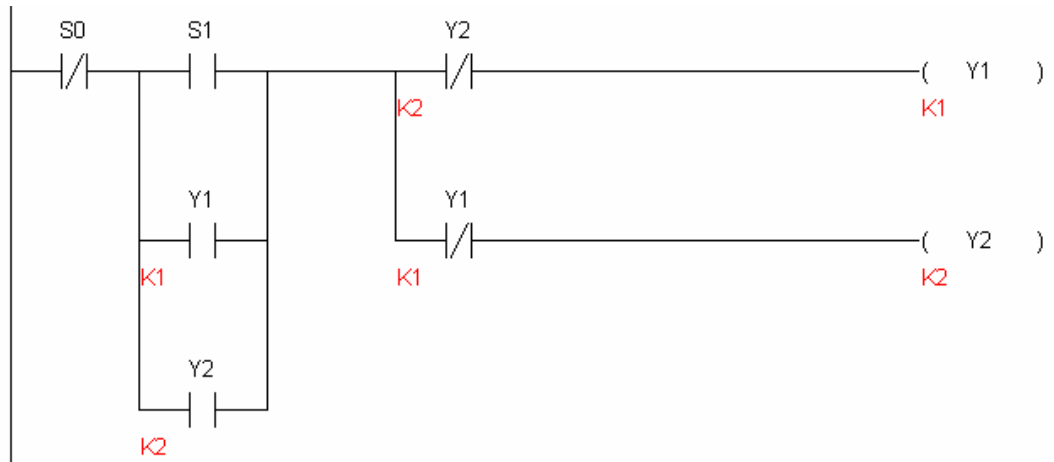
PLC'ler için geliştirilmiş olan programlama dilleri, kontaktlı (röleli) kumanda devreleri tasarımcıların kolayca anlayıp uygulayabileceği biçimdedir. Bir kontaktlı kumanda devresinden PLC'ye geçmek oldukça kolaydır. Bununla beraber kontaktlı kumanda devresi ile aynı amaçlı gerçekleştirilen PLC programı farklı sonuçlar verebilir. Bunun nedeni kontaktlı kumanda devresi ile PLC arasındaki yapısal farklılıklardır. Kontaktlı kumanda devresinde her röle ve kontaktör paralel çalışır. Yani aynı anda röle ve kontaktör bobinleri enerjilenir ve kontakları konum değiştirir. PLC'lerde ise mantık işlemler sırasıyla yapılır. Dolayısıyla bir kontaktörlü devrenin aynısı PLC programıyla gerçekleştirildiğinde PLC programının her zaman doğru sonucu vermediği aşağıdaki örnekle gösterilmiştir. Kontaktörlü devrelerde Şekil 3.1.'deki kumanda devresinde S1 butonuna basıldığında K1 ve K2 kontaktör

bobinleri aynı anda enerjilenir. Hangi kontaktör önce devreye girerse kontaktör devrede kalır. Diğeri devreye giremez. Bu ise kontaktörlerin zaman sabiteleri ve bağlantı iletkenlerinin direnç ve indüktansına bağlıdır.



Şekil 3.1. Kontaktlı Kumanda Devresi

Şekil 3.2'de, Şekil 3.1.'de görülen otomatik kumanda devresinin PLC için yazılan program parçası verilmiştir.



Şekil 3.2. Ladder Diyagramı

Bu programda S0 butonu 1 nolu PLC girişine, S1 butonu 2 nolu PLC girişine bağlanmıştır. Eğer bu program PLC ye yüklenir ve S1 butonuna basılırsa her zaman K1 kontaktörünün devreye girdiği K2 kontaktörünün hiçbir zaman devreye girmediği görülecektir. Bunun nedeni PLC'de programın adım adım işleme girmesidir. Çünkü 1. satırdaki K1 bobininin çıkış değeri mantık 1 olacağından ve bu değer 2. satıra ait mantık değeri hesaplanırken kullanılacağı için 2. satırın sonucu mantık 0 olacaktır. Kısaca PLC'deki program yazımında ilk önce yazılan komut işlem göreceğinden önce K1 bobini çeker ve kendisine ait kontağı açtığı için K2'nin devreye girmesi mümkün değildir.

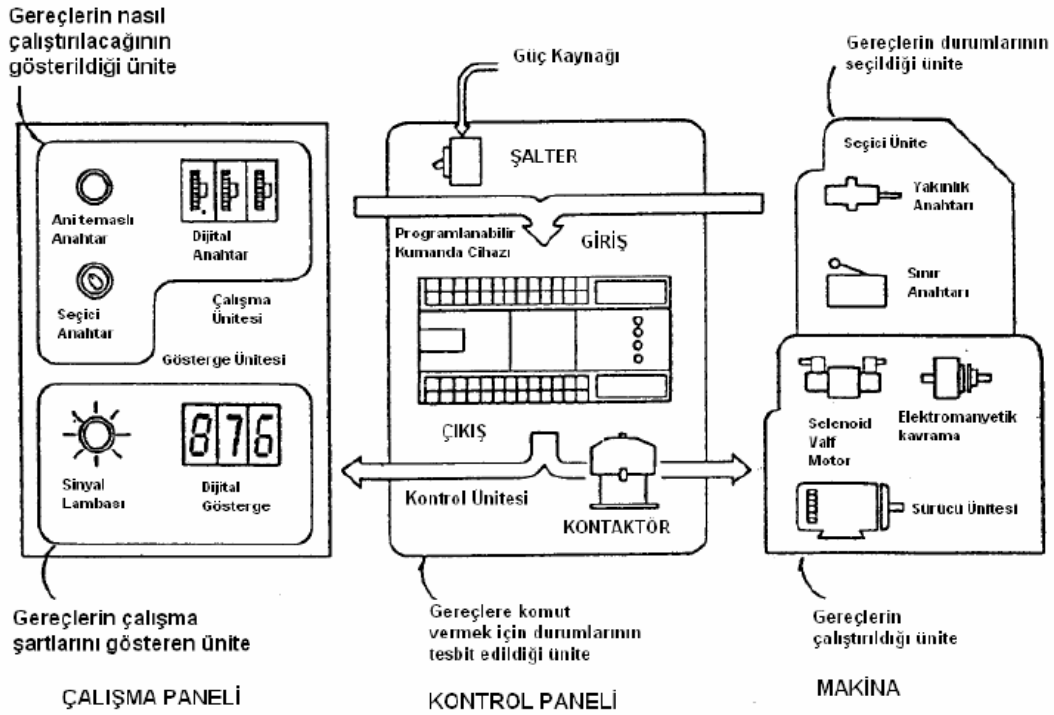
Ele alınan bu örnekte vurgulanmak istenen şudur ki, bir kontaklı kumanda devresinden yararlanılarak yazılan bir PLC programı amaca uygun çalışmayabilir. Bu nedenle yazılan her programın test edilmesi gerekir. Genellikle PLC programlayıcılarında veya kişisel bilgisayarlarda kullanılan derleyici programlarda bu imkan sağlanmıştır [3].

3.4. PLC'nin İç Yapısı

PLC'ler endüstriyel otomasyon devrelerinde doğrudan kullanıma uygun özel giriş ve çıkış birimleriyle donatılmıştır. Bu aygıtlara, basınç, seviye, sıcaklık algılayıcıları ve kumanda düğmesi gibi iki değerli lojik işaret bilgisi taşıyan elemanlar, kontaktör, selenoid gibi kumanda devrelerinin sürücü elemanları doğrudan bağlanabilir (Şekil 3.3). Bir PLC

- Bir sayısal işlemci ve bellek,
- Giriş ve çıkış birimleri,
- Programlayıcı birimi,
- Besleme güç kaynağı gibi temel kısımlardan oluşur.

Ayrıca programı yedeklemek ve başka bir PLC'ye aktarmak için kalıcı bellek birimi, giriş-çıkış sayısını arttırmak için genişleme birimi, enerji kesilmeleri durumunda PLC'yi besleyen yedek güç kaynağı ve iletişim arabirimi gibi elemanlar da bulunur.

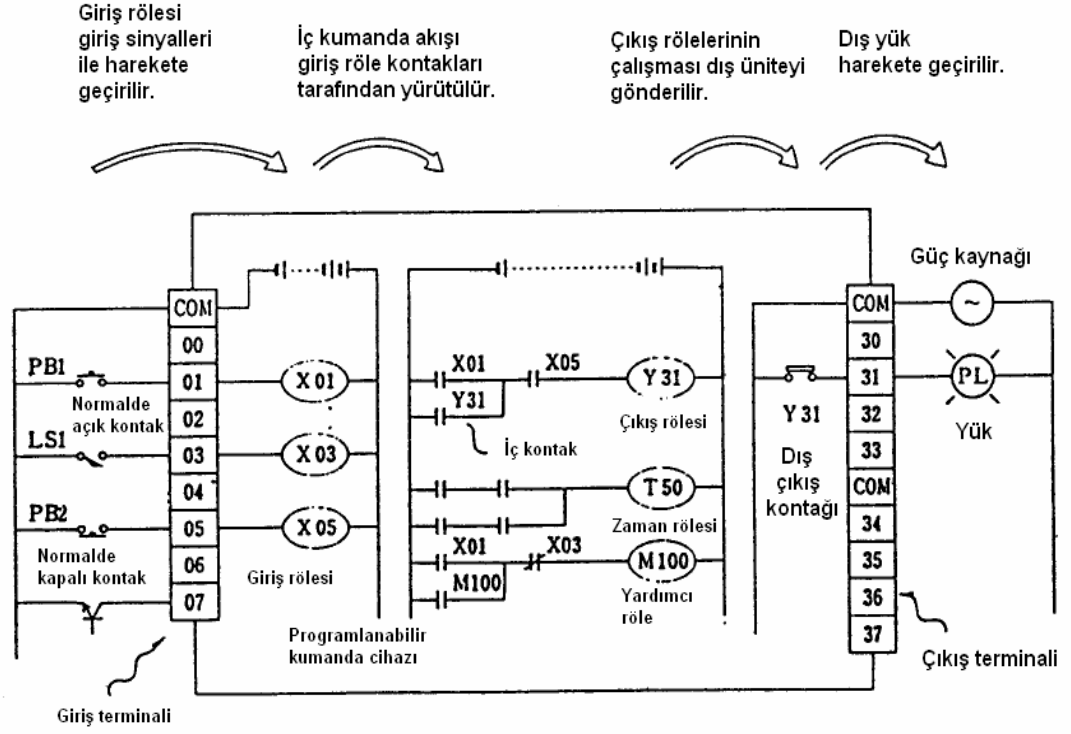


Şekil 3.3. PLC'nin Yapısı

Bir PLC'nin iç yapısı kabaca Şekil 3.4.'deki gibi verilebilir. Bütün sayısal bilgisayarlar gibi PLC; bir işlemci, bellek ve giriş-çıkış ara birimlerinden oluşur. Sayısal işlemci (mikrobilgisayar veya mikrokontrolör), PLC sistem programı altında kullanıcı programını yürütür, PLC'nin çalışmasını düzenleyen ve bu işlemleri yapmak için gerekli birimleri bulunan bir elemandır.

Bellek, sistem programının bulunduğu sistem belleği, kullanıcı programının bulunduğu program belleği ve veri belleği gibi bölümlerden oluşur. Sistem belleği ve PLC'ye ilişkin değiştirilmeyen veriler için salt okunur bellek (ROM); program belleği ve veri belleği için rasgele erişimli bellek (RAM) kullanılır. Veri belleği, giriş-çıkış işaret durumlarının tutulduğu giriş-çıkış görüntü belleği ve kullanıcıya ayrılmış bellek alanlarından oluşur.

Sistem belleği, üretici firmanın geliştirdiği PLC işletim sistemi programının yüklü olduğu bellek alanı, program belleği ise kullanıcı tarafından yazılan programın yüklendiği bellek alanıdır.

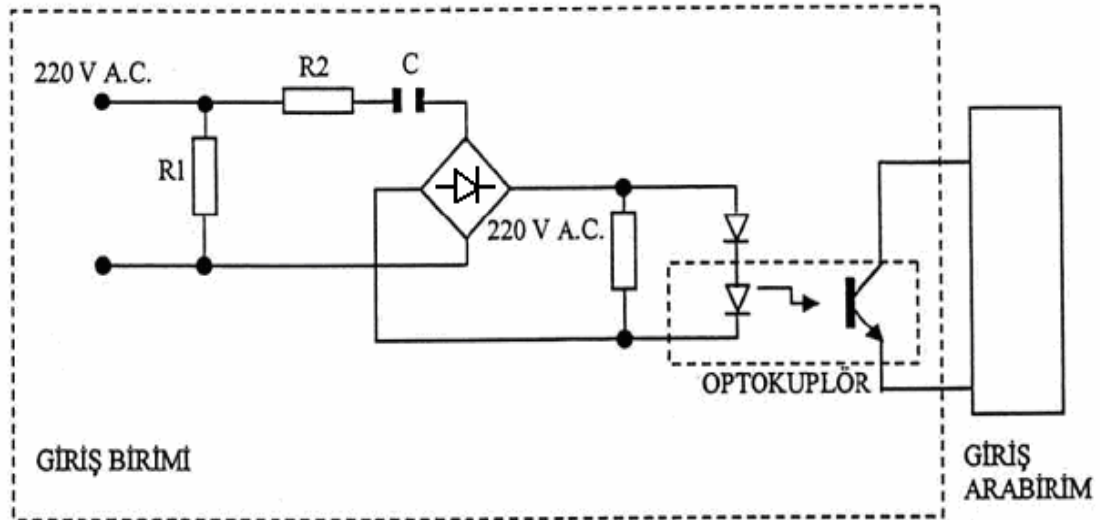


Şekil 3.4. PLC'nin Blok Diyagramı

Veri belleği alanında bulunan giriş görüntü belleği, programın yürütülmesi sürecinde, giriş birimindeki noktaların işaret durumlarının (var-yok; 1-0) saklandığı bellek alanı; çıkış görüntü belleği ise kullanıcı programının yürütülmesi sürecinde, çıkış noktalarına ilişkin hesaplanan değerlerin saklandığı bellek alanıdır. Kullanıcıya ayrılmış bellek alanına genellikle 1 bit, 8 bit, 16 bit ya da 32 bit'lik boyutlarda erişilebilir. 1 bit olarak erişilebilen bellek gözlerine marker, flag, internal output, auxiliary relay gibi isimler verilir. Bu alanlar kullanıcı programında hesaplanan 1 bitlik ara değerleri saklanması amacıyla kullanılır. Kontaklı kumanda devrelerindeki yardımcı rölenin işlevini görürler. Kullanıcıya ayrılmış bellek alanları- kalıcı (retentive) veya kalıcı olmayan (non-retentive) ve değişik özelliklerde değişken bellek bölümleri biçiminde düzenlenir [13].

3.4.1. Giriş birimi

Kontrol edilen sistemle ilgili algılama ve kumanda elemanlarından gelen elektriksel işaretleri lojik gerilim seviyelerine dönüştüren birimdir. Kontrol edilen sisteme ilişkin basınç, seviye, sıcaklık algılayıcıları, kumanda düğmeleri ve yaklaşım anahtarları gibi elemanlardan gelen iki değerli işaretler (var-yok, 0 veya 1) giriş birimi üzerinden alınır. Gerilim seviyesi 24V DC, 48V DC, 100V-120V AC veya 200V-240V AC değerlerinde olabilir. Şekil 3.5'te 200V-240V AC giriş gerilimi ile uyarılan bir giriş birimi devresi verilmiştir. PLC giriş birimi devresine gelen bir işaretin lojik 1 kabul edilebildiği bir alt sınır ve lojik 0 kabul edilebildiği bir üst sınır genlik değeri vardır. Giriş bilgisinin doğru olarak alınabilmesi için işaret gerilim seviyesinin bu sınır değerleri arasında olması gerekir.

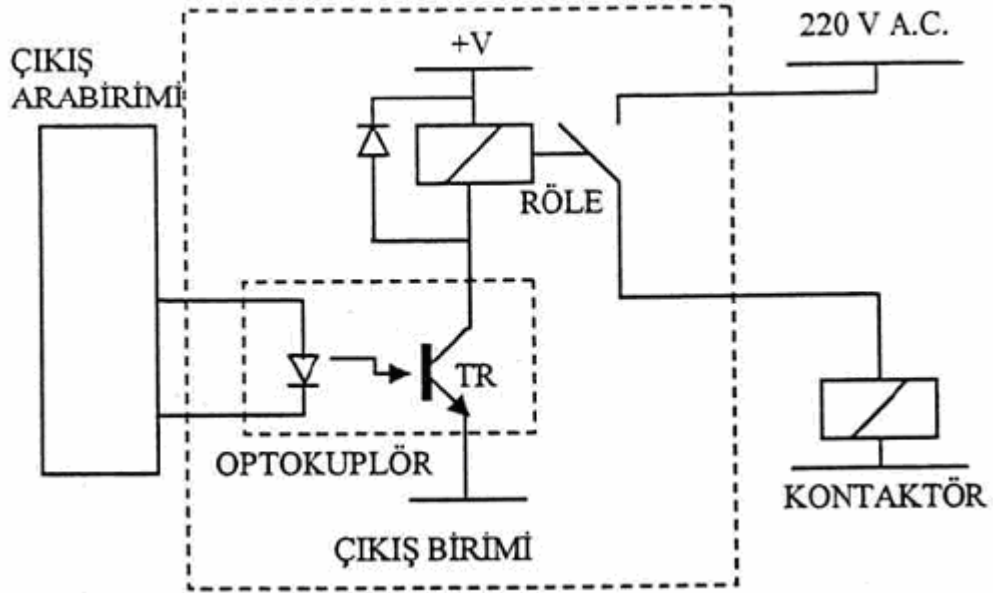


Şekil 3.5. 220V AC Gerilimle Uyarılan Bir Giriş Birimi Devresi

3.4.2. Çıkış birimi

PLC'de üretilen lojik gerilim seviyelerindeki işaretleri, kontrol edilen sistemdeki kontaktör, röle, selenoid gibi kumanda elemanlarını sürmeye uygun elektriksel işaretlere dönüştüren birimdir. Sürme elemanları için röle, triyak ya da transistör kullanılabilir. Çalışma sırasında çok sayıda yüksek hızlı açma-kapama gerektiren

durumlarda, doğru akımda transistörlü, alternatif akımlarda triyaklı çıkışlar kullanılır. PLC üzerindeki çıkış noktalarından çekilen alarm değerleri kontak çıkışlı birimler için 1A ile 8A arasında, triyak ve transistörlü birimler için 1A ya da 2A mertebesindedir. Uygulamada hangi çıkış biriminin kullanılacağı kumanda edilen elemanların özelliklerine bağlıdır. Örneğin, elektrik motorlarının kumandasında kullanılan kontaktörlerin sürülmesi için genellikle röle çıkışlı birimler kullanılır. Kontaktör sürülen kontak çıkışlı (röleli) bir birimin çıkış noktasına ilişkin devre Şekil 3.6.'da verilmiştir [1].



Şekil 3.6. Kontaktör Sürülen Röleli Çıkış Biriminin Bir Çıkış Noktasına İlişkin Devre

3.4.3. Diğer birimler

PLC'lerde yukarıda açıklanan giriş ve çıkış birimleri dışında, yüksek hız sayıcısı, kesme işareti girişi, analog giriş (ADC) ve analog çıkış (DAC) gibi giriş-çıkış birimleri de bulunur. Yüksek hız sayıcıları ve kesme işareti girişleri, PLC tarama çevrim süresinden daha hızlı değişen işaretlerin algılanıp değerlendirilmesi amacıyla kullanılır. Analog giriş ve çıkış birimleri ise geri beslemeli kontrol sistemlerinin

gerçeklenmesi için gereklidir. Küçük boyutlu PLC'lerde, genellikle besleme kaynağı ise giriş ve çıkış birimleri, işlemci birimi tümleşik olarak, büyük boyutlu PLC aynı birimler biçiminde bulunur.

3.4.4. Programlayıcı birimi

Kumanda ve kontrol amacıyla yazılan bir programın PLC program belleğine yüklenmesi bir programlayıcı birimi ile sağlanır. Programlayıcı birimi mikroişlemci tabanlı özel bir el aygıtı olabileceği gibi genel amaçlı kişisel bir bilgisayara yüklenmiş bir yazılım da olabilir. Bu birim; programın yazılması, PLC'ye aktarılması ve çalışma anında giriş/çıkış veya saklayıcı durumlarının gözlenmesi ya da değiştirilmesi gibi olanakları da sağlar. PLC'leri programlamak için geliştirilmiş olan yazılımlar, özellikle kumanda devreleri ile ilgili kişilerin kolayca kullanabilecekleri veya uyum sağlayabilecekleri editör-derleyici programlardır.

3.5. Çalışma Birimi

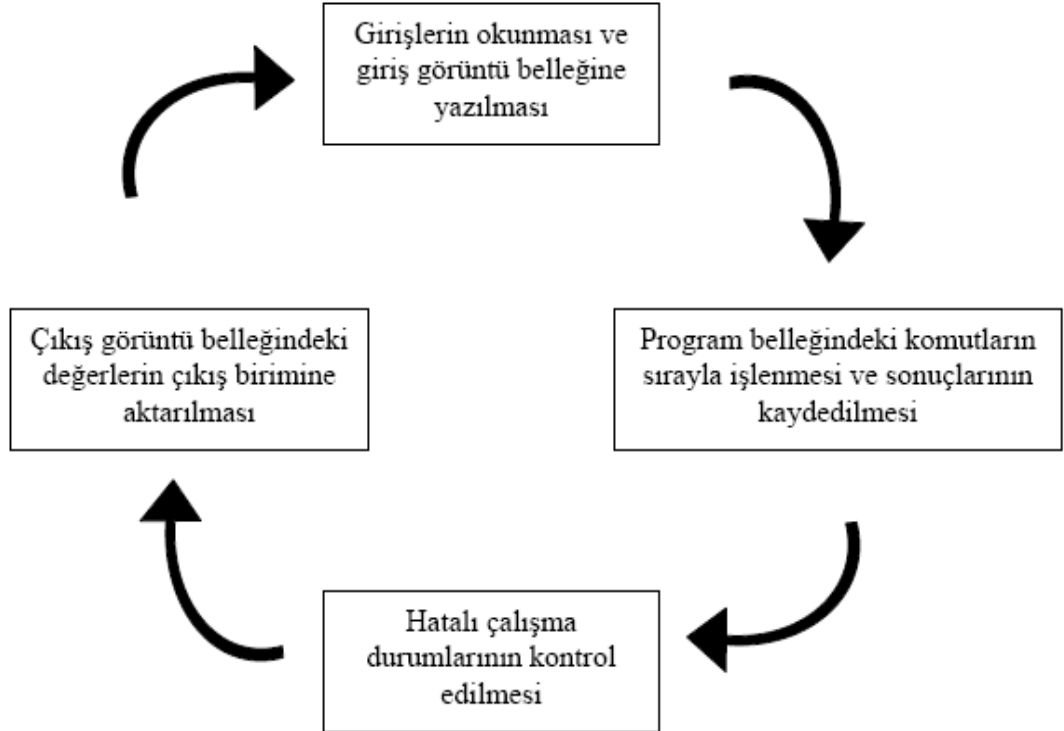
PLC'leri diğer mikroişlemci sistemlerden ayıran en önemli özelliklerden biri de çalışma biçiminin bir sistem programı ile düzenlenmesidir. Bütün PLC sistemlerinde birbirine çok benzeyen sistem; programları bulunur. Bu programlar üretim aşamasında kalıcı bir bellek alanına yüklenir. Genel olarak sistem programı şu işlevleri yerine getirir;

- Kullanıcı programını yürütür,
- Kesmeli çalışma ve iletişim olaylarını düzenler,
- Sistem çalışma durumlarını kontrol eder.

3.5.1. Kullanıcı programının yürütülmesi

PLC program belleğine yüklenmiş bir kullanıcı programı, birinci komuttan başlanarak son program komutuna kadar bütün komutların sırayla yürütülmesi biçiminde gerçekleşir. Program sonu komutuna erişildiğinde tekrar birinci komuta

dönülür. Bu çalışma biçimi sonsuz çevrime girmiş bir program parçası gibi düşünülebilir. Bu çalışma biçiminde komutların işleme sırası atlama, alt program çağırma gibi komutlar kullanıldığında ya da kesmeli çalışma durumlarında değişebilir. Ancak, her tarama çevriminin belirli bir sürede tamamlanması gerekir. Bir tarama işleminin belirli bir sürede tamamlanmaması durumunda ise sistem programı PLC'nin çalışmasını durdurur. Bu süre genellikle 300ms ile 1000ms arasında değişir. Bu işlem bir gözetleme zamanlayıcısı (Watchdog Timer) ile sağlanır. PLC'lerde bir çevrimin tamamlanması için geçen süreye tarama süresi denir. Bir PLC'nin tarama zamanı giriş-çıkış sayısına, programın içeriği ve uzunluğuna, işlemcinin çalışma frekansına bağlıdır. Genel olarak tarama hızı 1024 byte başına işlem hızı olarak verilir ve 0,5ms ile 200ms arasında değişebilir. Bir PLC'de alt program ve kesmeli çalışma yapılmadığı durumlarda kullanıcı programının yürütülmesi Şekil 3.7.'deki akış diyagramında verildiği gibi olur.



Şekil 3.7. PLC İşlem Evreleri

Bir PLC çalışma (RUN) durumuna getirildiğinde sırayla aşağıdaki işlemler gerçekleşir.

- Giriş birimindeki değerler giriş görüntü belleğine alınır ve saklanır. Bu değerler bir sonraki çevrime (taramaya) kadar değişmez.
- Yazılan programa göre program komutları adım adım sırayla işlenir. Ancak giriş değerleri için giriş görüntü belleğinden okudukları andaki değerler geçerlidir ve bir program çevrimi süresince bu değerler değişmez.
- Kullanıcı programının yürütülmesi tamamlandıktan sonra hesaplanan değerler çıkış görüntü belleğine yazılır ve çıkış birimine gönderilir. Çıkış birimine aktarma işlemi tamamlandıktan sonra tekrar birinci adıma dönülür. Çıkış görüntü belleği ve çıkış birimindeki değerler bir sonraki çevrime kadar değişmez.

Genel olarak bütün PLC'lerde programın yürütülmesi bu şekilde gerçekleşir. Ancak giriş görüntü belleğindeki değerlerin alınması ve çıkış görüntü belleğine değer yazılması işlemlerinin değişik yapıldığı PLC işletim sistemleri de vardır. Buna örnek olarak, hesaplanan çıkış değerlerinin doğrudan çıkış birimine gönderildiği (DPS. direct processing system) işletim sistemi verilebilir (Hitachi H-200). Bu işletim sisteminde, giriş adresini içeren bir komuta rastlanıldığında girişin o andaki değeri alınır. Çıkış adresine değer atayan bir komutun yürütülmesi tamamlandığında, sonuç doğrudan çıkış birimine iletilir.

Bazı PLC modellerinde giriş ve çıkışlara doğrudan erişmek için özel komutlar kullanılır. Bu komutlar ivedi (anında) giriş-çıkış komutları (immediate I/O instructions) olarak adlandırılır. Bu komutlarla işlenen giriş değerleri görüntü belleğindeki değerler olmayıp komutun yürütüldüğü andaki giriş değerleridir. Aynı anda çıkışa aktarılan değer, ivedi işlem komutunun yürütülmesi tamamlandığında hesaplanan değerdir. Bu tür komutların kullanım amacı hızlı değişen giriş işaretlerini yakalamak ve hesaplanan bir değeri anında çıkış birimine aktarmaktır. Bu komutların normal program çevrimi içinde kullanılması hızlı değişen işaretlerin algılanma

olasılığını arttırır. Ancak, hızlı değişen işaretleri değerlendirme olasılığını yüzde yüz güvence altına almaz. Çünkü komut yürütülmeden önce ya da komut yürütüldükten sonraki işaret değişimleri algılanamaz ve yine bilgi kaybı olabilir.

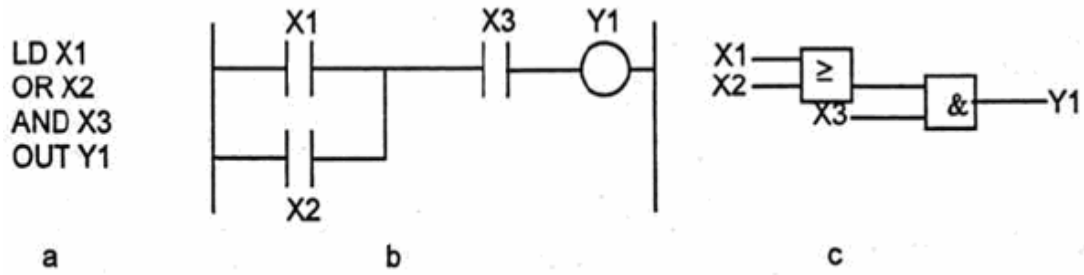
Yüksek hızlı değişen işaretleri algılamak ve değerlendirmek için en güvenceli yol kesmeli çalışma yöntemidir. Hesaplanan değerlerin anında çıkış birimine aktarılması, kesme alt programlarında ivedi işlem komutları kullanılarak sağlanır. Kesmeli çalışmada, kesme işareti geldiği anda, normal programın yürütülmesine ara verilir ve kesme olayı ile bağlantılı kesme alt programı yürütülmeye başlanır. Kesme alt programındaki komutların işlenmesi tamamlandığında tekrar normal programa dönülür. Bir kesme alt programında çıkışa ilişkin hesaplanan bir değer çıkış birimine aktarılması yine normal program çevrimi tamamlandığında (çıkış görüntü belleğinin çıkış birimine aktarılması aşamasında) olur. Kesme alt programının yürütülmesi anında, çıkışa ilişkin işaretlerin doğrudan çıkış birimine gönderilmesi için ivedi işlem komutları kullanılabilir. Böylece, hem giriş işaretinin anında değerlendirilmesi, hem de sonucun anında çıkış birimine aktarılması sağlanmış olur [1].

3.6. Programlama Dilleri

PLC'ler için geliştirilmiş olan programlama dilleri, kontaktörlü ve röleli kumanda devrelerinin tasarımı ile ilgili kişilerin kolayca anlayıp uygulayabileceği biçimde kullanıma sunulmuştur. Genel olarak üç türlü programlama biçimi kullanılır. Bunlar;

- Komut kullanımı ile programlama (Statement list, instruction list),
- Merdiven diyagramı gösterimi (MDG) ile programlama (Ladder programming),
- Diğer programlama yöntemleri (Grafcet, lojik kapı, kontrol akış gibi) olarak sınıflandırılabilir.

Şekil 3.8.'de bu programlama biçimlerine ilişkin bazı örnekler verilmiştir.



- a) Komut gösterimi (STL),
 b) Merdiven diyagramı gösterimi (MDG veya LAD),
 c) Lojik kapı gösterimi (LKG veya FBD) programı.

Şekil 3.8. $Y1=(X1+X2)*X3$ biçimindeki bir lojik fonksiyona ilişkin

Komut ile programlama, kumanda sistemine ilişkin çözümün mikro denetleyici ya da mikroişlemci sistemlerinde yapıldığı gibi, PLC komut kümesi ile gerçekleşmesidir. MDG ve LKG ile programlama ise, standart kontak ve lojik kapı şekilleri kullanılarak yapılan bir grafiksel programlama biçimidir.

3.7. Programlama

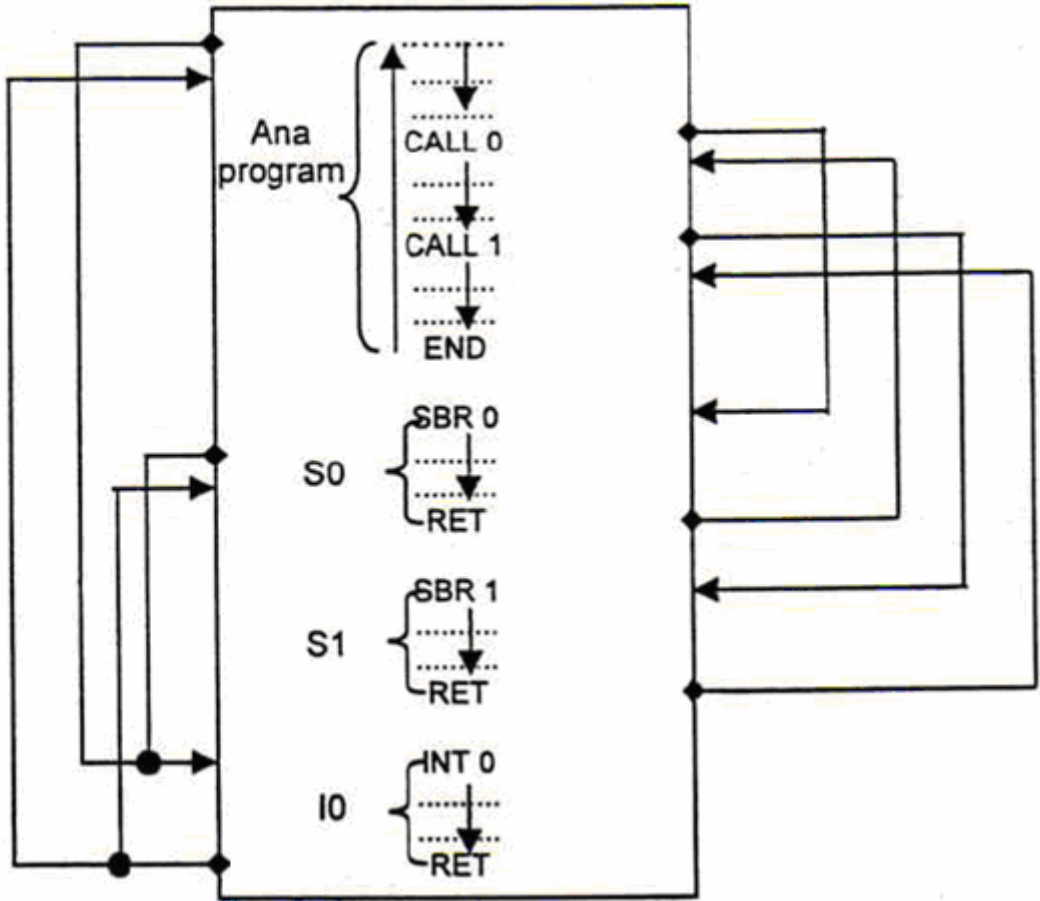
Programlama, belirli bir görevi yapmak üzere tasarlanmış bir kontrol ya da kumanda sisteminin PLC'de gerçekleşmesi için, yürütülmesi gereken işlemleri sağlayan, komut dizisinin yazılmasıdır. Günümüzde üretilen PLC'lerde, kumanda devrelerinin gerçekleşmesine ilişkin lojik işlem komutlarından kontrol algoritmalarının yazılmasına ilişkin matematik işlem komutlarına kadar geniş bir komut kümesi bulunur [13].

3.7.1. Programlama yapıları

Bütün kumanda ve kontrol sistemlerinin gerçekleşmesine ilişkin kuralların PLC'ye aktarılmasında kullanılan programlama yapıları iki ana başlık altında toplanabilir. Bunlardan doğrusal programlama olarak tanımlanan yapı genellikle küçük ölçekli, yapısal programlama olarak tanımlanan yapı ise orta ve büyük ölçekli PLC'lerin programlanmasında tercih edilir [1].

3.7.1.1. Doğrusal programlama

Doğrusal programlama, bütün komutların aynı program içinde yer aldığı ve komutların art arda yazıldığı bir programlama biçimidir. Komutlar yazılış sırasına göre yürütülür ve bir çevrim boyunca bütün komutlar işleme girer. Bu programlama biçiminde bütün komutlar ana programda bulunur. Şekil 3.9'da görüldüğü gibi alt programlar ve kesme alt programları, ana programın program sonu komutundan sonra, sırayla yazılır [13].



Şekil 3.9. Doğrusal programlama

Şekil 3.9.'da verilen programın işleyişi şu şekilde olur:

- Ana programa yazılan komutlar sırayla işlenir.

- Ana programdaki 0 nolu altprogramı (S0) çağırma komutu CALL 0 ile program akışı 0 nolu alt programa geçer. Alt program yürütüldükten sonra program akışı CALL 0 komutunu izleyen komutla sürer.
- Diğer alt programı çağırma komutuna kadar yazılan komutlar sırayla işlendikten sonra CALL 1 komutu ile program akışı 1 nolu altprograma (S1) geçer ve bu alt program yürütüldükten sonra tekrar ana programdaki CALL 1 komutunu izleyen komuta dönülür.
- Ana program sonu komutuna erişildikten sonra tekrar ana programın ilk komutuna dönülür ve aynı işlem tekrarlanır.
- Program yürütülürken herhangi bir anda kesme işareti oluştuğunda program akışı 0 nolu kesme alt programına geçer ve kesme alt programı yürütüldükten sonra kesme işaretinin geldiği andaki noktaya dönülür. Kesmeli çalışma aynı program çevrimi içinde birden çok olabilir.

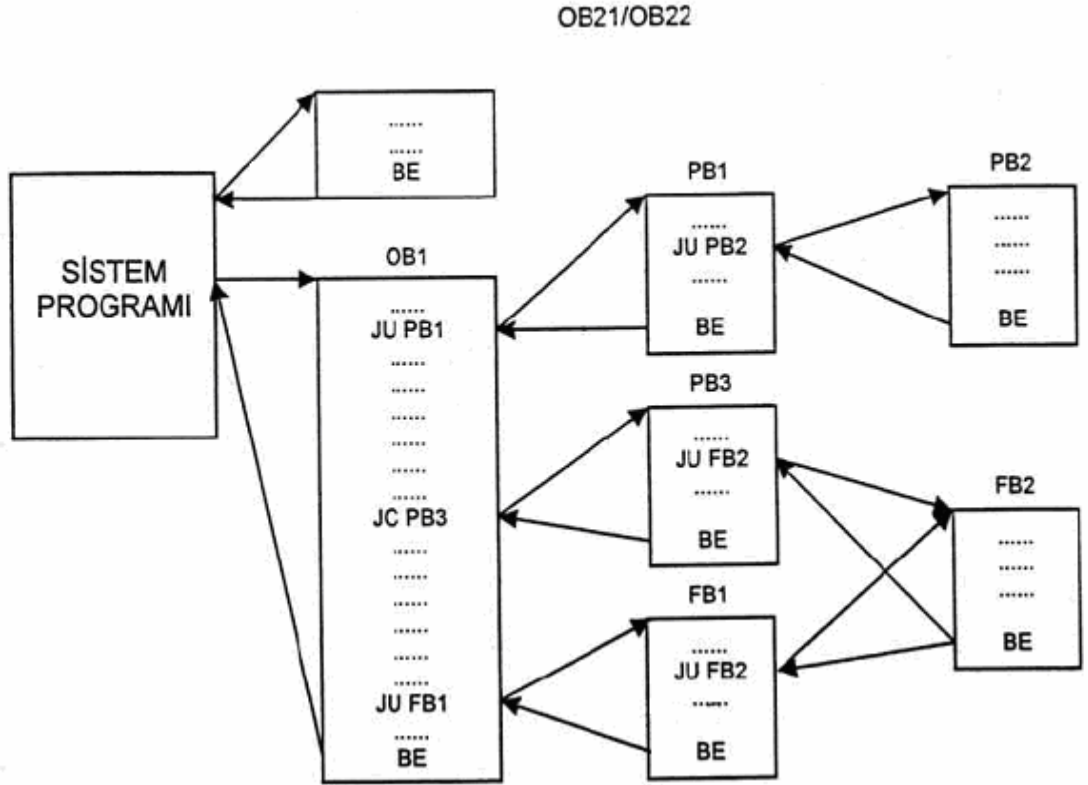
Doğrusal programlamada, bir alt programda başka bir alt programın çağrıldığı yapılar kullanılabilir. Ancak bu tür programlama kumanda sisteminin tasarımını ve izlenmesini zorlaştırdığı için genellikle tercih edilmez. Ana programa yazılan komutlar genellikle alt programlarda da kullanılabilir.

Kesme alt programlarının, PLC çalışma güvenliği açısından, kısa ve hızlı işlenmesi gereken program parçalarını içermesi gerektiğinden, kesme alt programları için komut kısıtlamaları konulabilir [13].

3.7.1.2. Yapısal programlama

Yapısal programlama, Şekil 4.9'da gösterildiği gibi organizasyon bloğu, program blokları, fonksiyon blokları biçiminde düzenlenmiş bir programlama biçimidir [1]. Bu programlamada her bir kontrol işlemi bloklar halinde bağımsız olarak oluşturulur ve bir icra programı yazılarak her bir çevrimde hangi bloğun ya da blokların yürütüleceği belirlenir. Yapısal programlamada bir çevrimde program belleğindeki programın bütün bloklarının işlenmesi gerekmez [2]. Bütün program blokları birer alt

program gibi düşünülebilir. Kesme alt programları içinde yine tanımlanmış özel organizasyon blokları kullanılır. Sistem programı organizasyon bloklarını işletir. Organizasyon blokları ise program ve fonksiyon bloklarını işletir. Organizasyon bloğuna yazılan atlama komutları ile bir program çevriminde hangi blokların yürütüleceği belirlenir. Şekil 3.10'da yapısal programlama biçimi gösterilmiştir.



Şekil 3.10. Yapısal Programlama

Şekil 3.10.'da verilen programlama biçiminde OB1 organizasyon bloğu, PB1, PB2, PB3 program blokları ve FB1, FB2 fonksiyon bloklarıdır. OB1 organizasyon bloğuna yazılan JU (koşulsuz atlama) ve JC (koşullu atlama) komutları blokların işleme girme sırasını belirler. Şekilden görüldüğü gibi, herhangi bir bloktan başka bir bloğa da atlama yapılabilir.

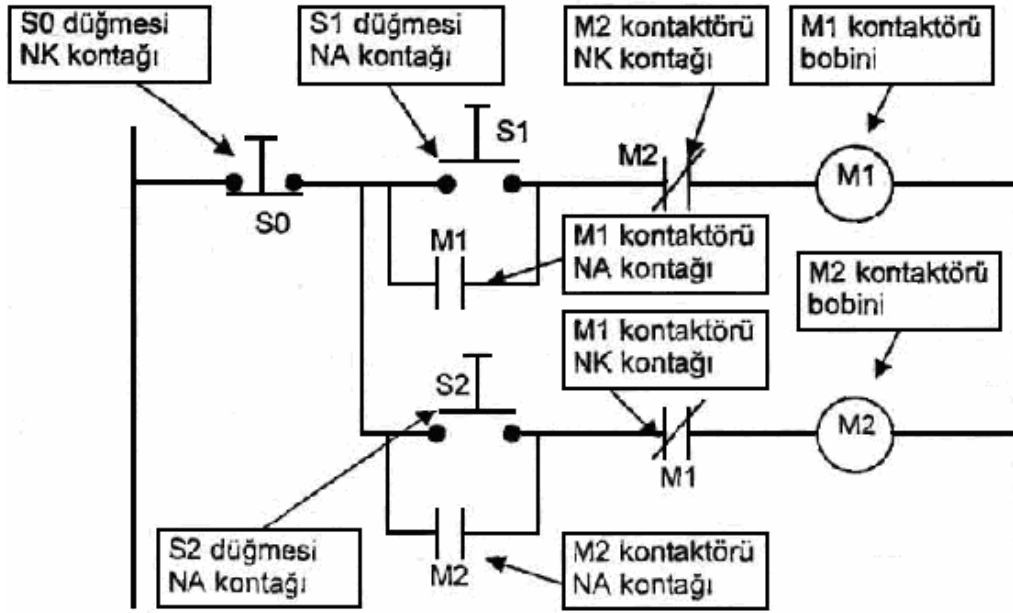
Bu programlama yönteminde, sistem çalışmasına ilişkin organizasyon blokları ve kesmeli çalışmaya ilişkin organizasyon blokları sistem programı tarafından işletilir. Örneğin, Şekil 3.10'da gösterilen OB21 organizasyon bloğu, bazı Siemens S5 serisi

PLC'lerde, elle durma (STOP) konumundan çalışma durumuna (RUN) geçildiğinde sistem programı tarafından işletilen bir organizasyon bloğudur. OB22 organizasyon bloğu ise PLC'ye enerji verildiğinde sistem programı tarafından işletilir. Kesmeli çalışmaya ilişkin benzer organizasyon blokları bulunur ve bu bloklar sistem programı tarafından işletilir.

Yukarıda açıklanan her iki programlama yapısında da doğrusal programlamaya ilişkin ana program ve alt programlar; yapısal programlamaya ilişkin organizasyon, program ve fonksiyon blokları doğrusal olarak programlanır. Programlamada, daha önce denildiği gibi komut, merdiven ya da lojik kapı gösterimi ile programlama biçimleri kullanılabilir. Bu programlama biçimlerinden merdiven diyagramı gösterimi (MDG) ve lojik kapı gösterimi (LKG) ile programlama biçimleri, bütün PLC'ler için, benzer şekilde işletilir. Komut ile programlama biçimi ise, bazı PLC modelleri için hem komut hem de komutların işlenmesi bakımından değişiklik gösterebilir [1].

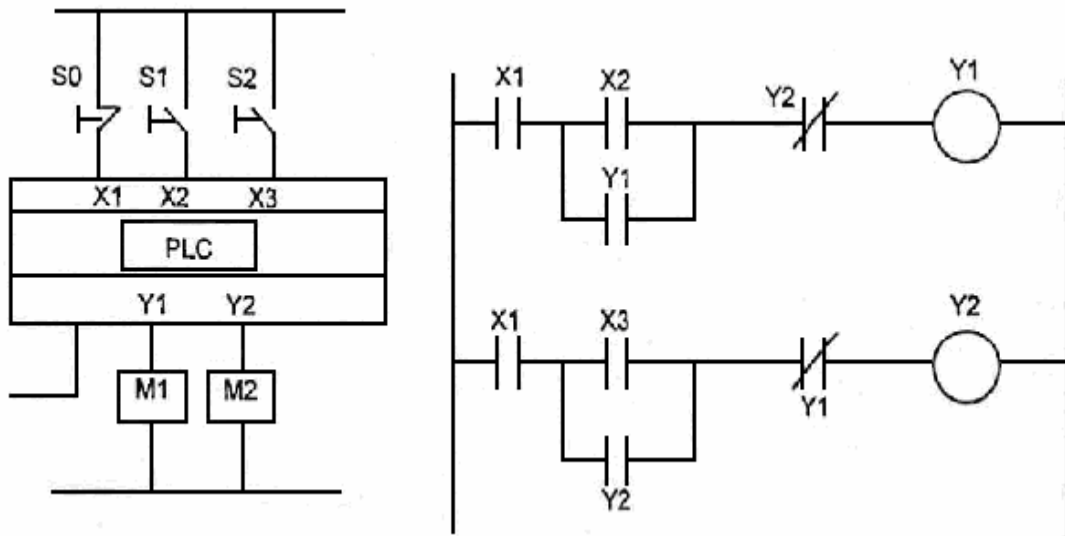
3.7.2. MDG ile programlama tekniği

MDG ile programlama tekniği, kontaklı (anahtarlamalı) kumanda devrelerinin ANSI (American National Standards Institute) standartları devre simgeleri ile gösterilişine benzeyen bir programlama biçimidir. Kontaklı kumanda devreleri normalde açık (NA), normalde kapalı (NK) kontaklar, kontaktör ve/veya yardımcı röle bobinlerinden oluşan devrelerdir. Şekil 3.11'de örnek bir kumanda devresi görülmektedir.



Şekil 3.11. Örnek Bir Kumanda Devresi

Şekil 3.11.'de verilen kumanda devresi ile aynı işleve sahip bir PLC'li kumanda devresi Şekil 3.12.'de görülmektedir. Burada, S0 durdurma düğmesi normalde kapalı kontak üzerinden PLC X1 girişine bağlandığından, MDG programında X1 normalde açık (NA) kontak olarak yer almıştır.

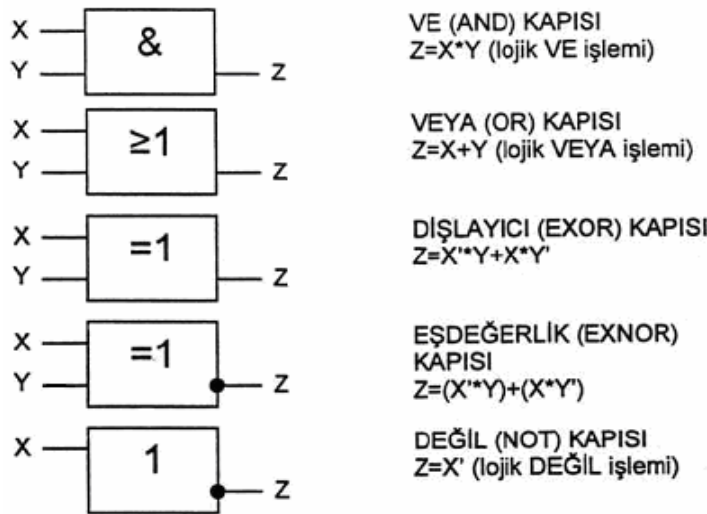


Şekil 3.12. PLC Giriş-Çıkış Bağlantıları ve MDG Programı

Şekil 3.12.'deki MDG programı, bir kontaklı kumanda devresi gibi, akım yolu izlenerek incelenebilir. Örneğin, birinci basamaktaki Y1 çıkışının uyarılması (etkin olması) için X1 kontağının kapalı, X2 veya Y1 kontağının kapalı Y2 kontağının kapalı olması gerekir. MDG programı, PLC giriş ve çıkış bağlantıları ile birlikte göz önüne alınarak incelenir. Örneğin, S0 durdurma düğmesi normalde kapalı kontağı üzerinden X1 girişine bağlandığından, bu düğmeye basılmadığı sürece X1 girişi etkindir ve merdiven diyagramında X1 değişkenini simgeleyen normalde açık kontak kapalı olarak değerlendirilir. X2 ve X3 kontakları ise S1 ve S2 düğmelerine basıldığında etkin olur ve bu düğmelere basılmadığı sürece açık kontak olarak değerlendirilir. Aynı şekilde, birinci basamaktaki Y2(NK) kontağı Y2 çıkışı etkin olmadığı sürece kapalı kontak olarak düşünülür. Belirli bir işlevi yerine getirmek için tasarlanmış geleneksel bir kontaklı kumanda devresi, aynı yapı kullanılarak bir PLC'li kumanda devresine dönüştürülebilir. Bu şekilde gerçekleştirilmiş bir PLC'li kumanda devresi, genellikle, kontaklı kumanda devresi ile aynı işlevi sağlar [13].

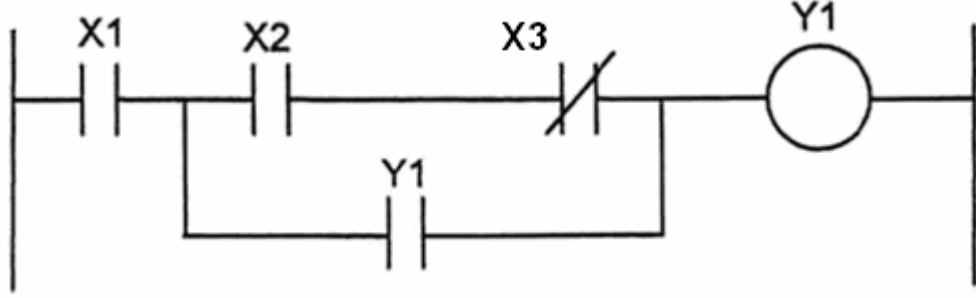
3.7.3. LKG ile programlama tekniği

LKG ile programlama, kumanda devresinin lojik işlemleri simgeleyen lojik kapı şekilleri ile programlanmasıdır. Bu programlama biçiminde lojik kapı şekilleri, art arda seri ya da paralel bağlanarak, program yazılır. Şekil 3.13.'de ISO, ANSI ve IEEE standartlarına göre belirlenmiş olan temel lojik kapı şekilleri gösterilmiştir.



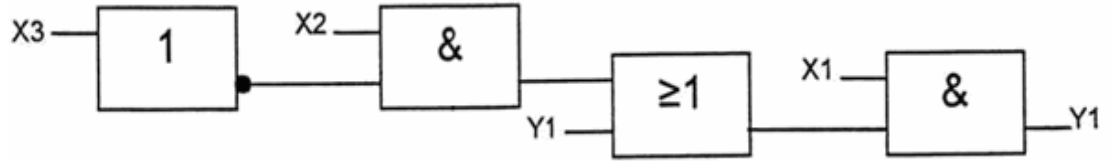
Şekil 3.13. Lojik Kapı Şekilleri

Şekil 3.13.'de gösterilen lojik kapı şekilleri kullanılarak herhangi bir lojik fonksiyon programlanabilir. Örneğin, Şekil 3.13.'te verilen MDG programı Şekil 3.14.'deki gibi programlanabilir.



Şekil 3.14. Ladder Programı

Şekil 3.14' deki programda Y1 çıkışına ilişkin lojik fonksiyon $Y1=X1*(X2*X3'+Y1)$ olup bu fonksiyon Şekil 3.15.'teki gibi grafiksel olarak programlanabilir.



Şekil 3.15. LKG programı

Lojik kapı şekilleri ile programlama, bütün PLC programlama yazılımları ile desteklenen bir yöntem olmasına karşın, lojik kapı elemanları ile devre tasarımı deneyimi olan kişilerin tercih ettiği bir programlama biçimidir.

3.7.4. Komut ile programlama

Komut ile programlama, PLC işlem komutlarının art arda yazılarak yapıldığı bir programlama biçimidir. Bütün PLC programlama yazılımları tarafından desteklenen bir yöntemdir. MDG ya da LKG ile yapılan programlama basit olmasına karşın, belirli kurallara uyulmadığı durumlarda programın derlenmeme olasılığı bulunur. Bu tür durumlarda, programın işleyişini incelemek ve sorunu çözmek için komut ile programlamanın bilinmesi zorunludur. Bunun yanı sıra bazı PLC programlama

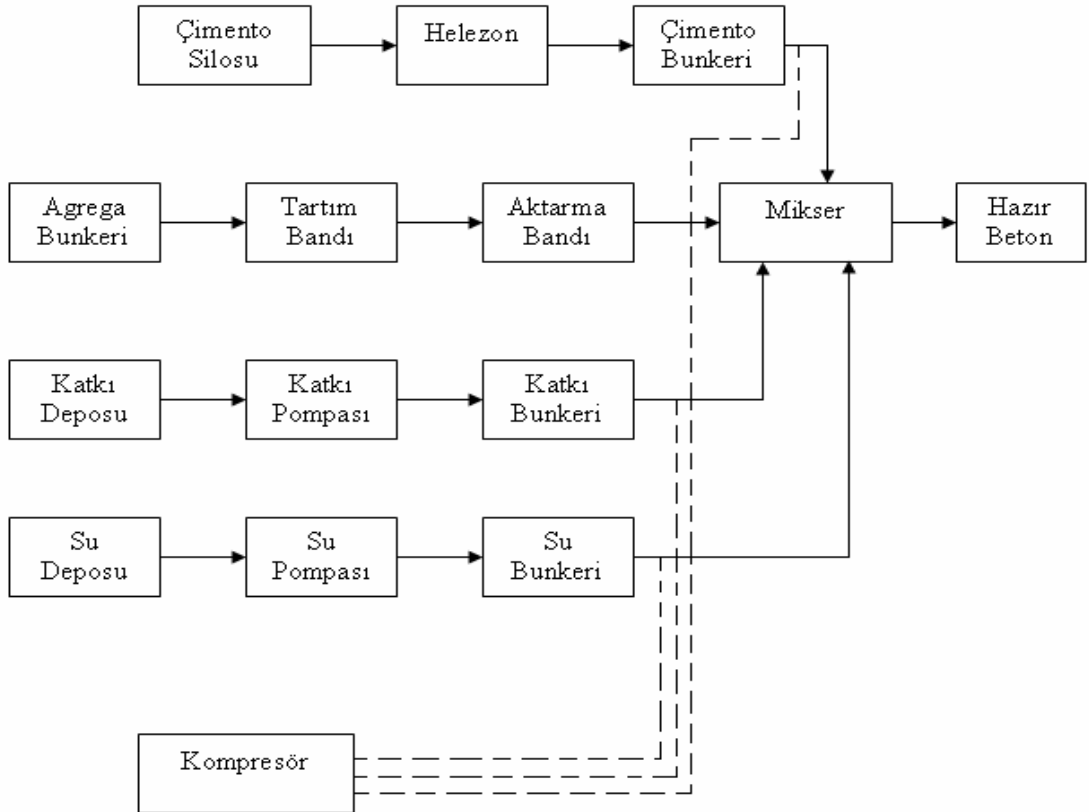
Çizelge 3.1.'de verilen komutlar ile her türlü lojik fonksiyon programlanabilir. Aynı satırdaki komutlar aynı işlevi sağlar ve benzer biçimde programlanır [13].

4.TASARLANAN OTOMASYON SİSTEMİ

Bir hazır beton santralinde üretime başlama şartlarını aşağıdaki gibi sıralamak mümkündür;

- i) Mikser çalışır durumda olmalı.
- ii) Hava basıncı normal olmalı.
- iii) Loadcell'lerin üzerinde tartımda dikkate alınacak yük olmamalıdır.

Üretim başlama şartları varsa başla butonuna basılarak üretim başlatılır. Üretim başladıktan sonra üretimle ilgili tüm tartım bilgileri ve çevrim detayları tartım analizden izlenebilir. Üretim de birinci çevrim mikserde karıştıktan sonra boşaltma süresini saymadan yeni çevrim başlamaz. Tasarlanan otomasyon sisteminde baz alınan hazır beton santralinin blok diyagramı Şekil 4.1'de gösterilmektedir.



Şekil 4.1. Hazır Beton Santrali Blok Diyagramı

Hazır beton santralının birimleri Çizelge 4.1.'de, girdi ve çıktıları ise Çizelge 4.2.'de gösterilmektedir.

Çizelge 4.1. Birimlerin Elektrik Parametreleri

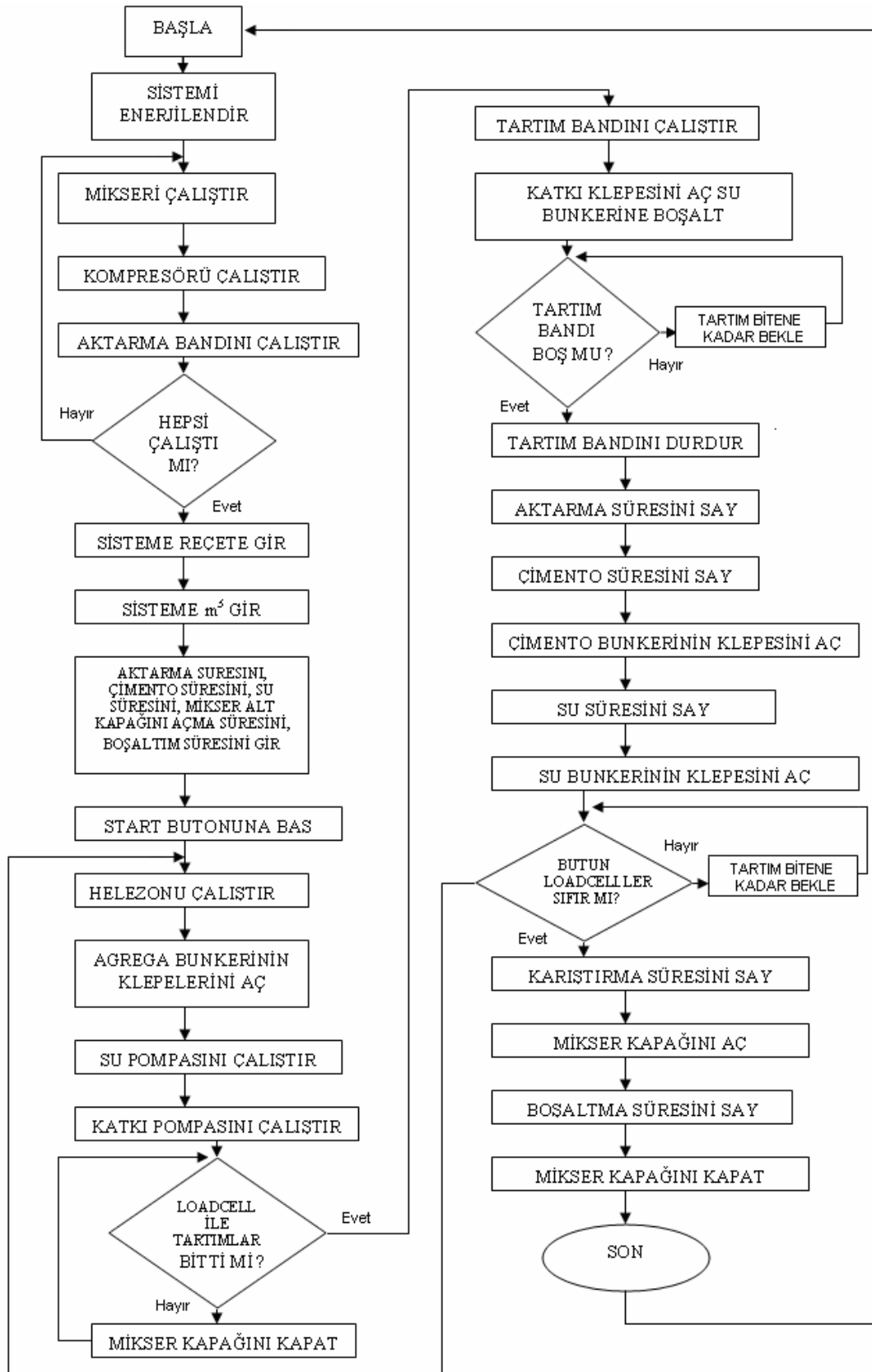
Birim	Akım	Gerilim	Güç
Ventil	10-90 mA	24V - 220 V	1,5-2 W
Vibratör	0.5-1 A	380 V	0,37 kW
Nem Ölçer	70-190 mA	9V – 30 V	1,5-2 W
Tartı Bandı	20-30 A	380 V	11 kW
Aktarma Bandı	40-80 A	380 V	22 kW
Helezon Motoru	20-40 A	380 V	7.5-15
Su Pompası	5-10 A	380 V	3 kW
Katkı Pompası	0.5-1 A	380 V	0.37 kW
Mikser	100-150 A	380 V	37-55 kW

Çizelge 4.2. Beton Santrali Girdi Çıktıları

Birim	Girdi	Çıktı
Agrega Bunkeri	Kum, Çakıl	Kum, Çakıl
Tartım Bandı	Kum, Çakıl	Kum, Çakıl
Aktarma Bandı	Kum, Çakıl	Kum, Çakıl
Çimento Silosu	Çimento	Çimento
Helezon	Çimento	Çimento
Çimento Bunkeri	Çimento	Çimento
Katkı Deposu	Katkı Maddesi	Katkı Maddesi
Katkı Pompası	Katkı Maddesi	Katkı Maddesi
Katkı Bunkeri	Katkı Maddesi	Katkı Maddesi
Su Deposu	Su	Su
Su Pompası	Su	Su
Su Bunkeri	Su	Su
Kompresör	Hava	Hava
Mikser	Kum, Çakıl, Çimento, Katkı Maddesi, Su	Hazır Beton

Tasarlanan otomasyon sisteminde geçerli olan genel PLC programının akış diyagramı Şekil 4.2 de gösterilmiştir. Akış diyagramından da görüleceği üzere önce mikser, aktarma bandı ve kompresör enerjilendirilerek sürekli çalışması sağlanır. Agreganın birinci klepesi açılarak malzeme tartım bandına dökülür. Burada malzemenin ağırlığı ölçülür ve operatörün girdiği kilo ile karşılaştırılır ve ölçüm referans kilodan 10kg az ise klepe kapanır. Klepe kapanana kadar geçen sürede, havada kalan malzemenin ağırlığı ölçülemeyeceğinden tartım referans kilodan azken

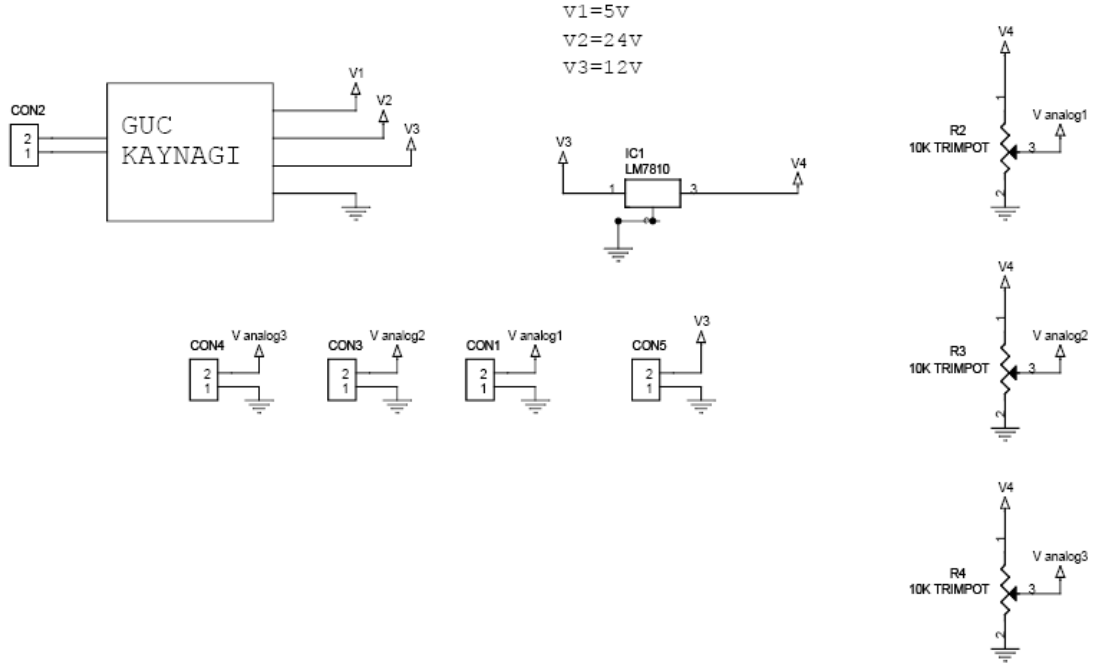
klepe kapanır. Bu tartım işlemi sırasıyla agreganın diğer üç malzemesi için de yapılır. Tartım işlemleri bittikten sonra tartım bandındaki malzemeler aktarma bandına buradan da mikserde gönderilir. Agreganın tartım işleri yapılırken aynı anda da su ile katkı ve çimento tartılır. Bu malzemeler de mikserde gönderilir. Bütün malzemeler mikserde toplandıktan sonra operatörün girdiği süre boyunca mikser çalışır. Süre bitiminde ise mikserin klepesi açılarak malzemeler transmiksere dökülür. Bütün bu işlemler ise üretilmek istenen betonun miktarına göre kaç çevrim çalışacağı hesaplanarak tekrarlanır.



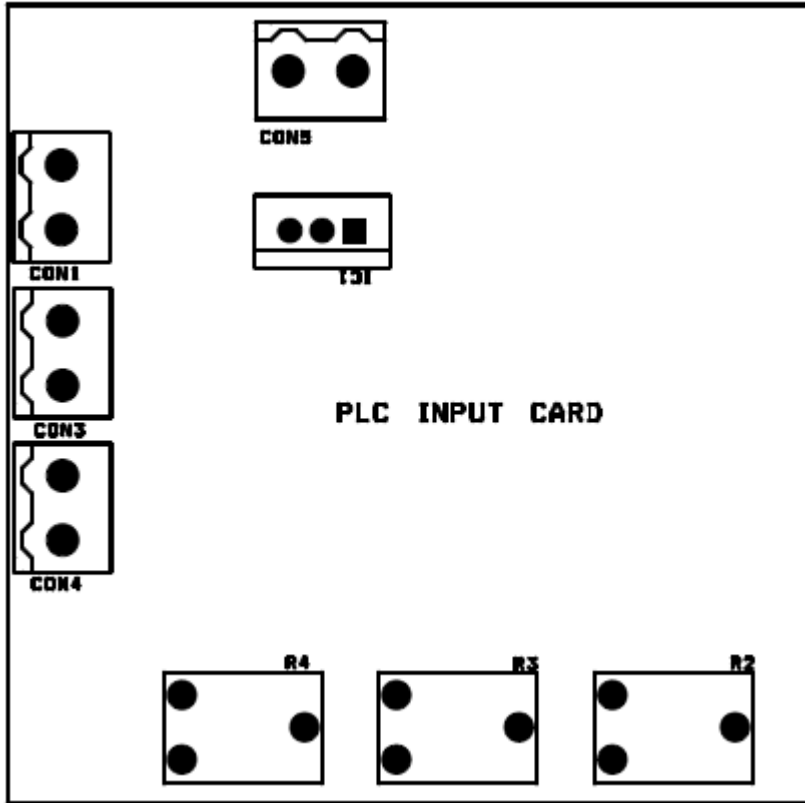
Şekil 4.2. Hazırlanan PLC Programının Akış Diyagramı

4.1 Tasarlanan PLC Giriş Kartı

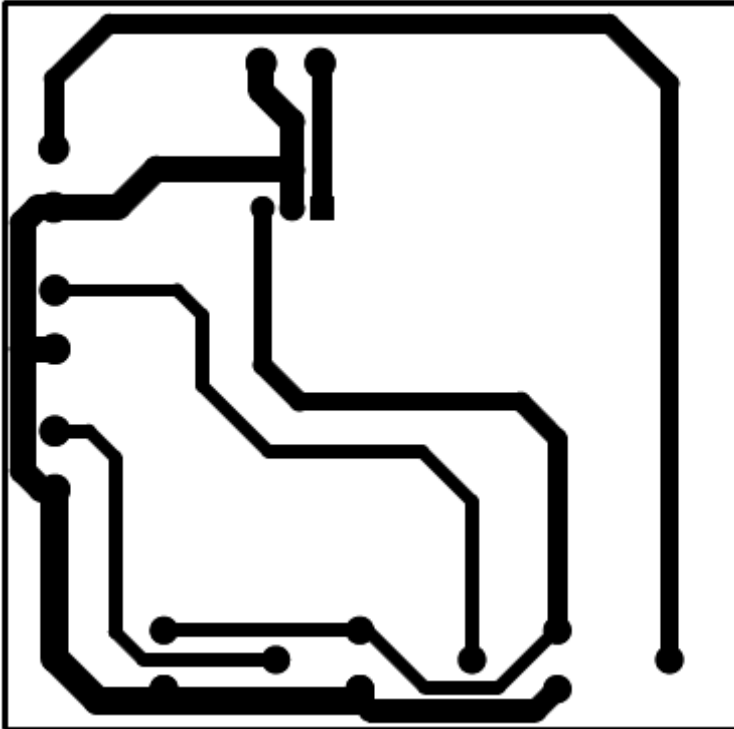
Tartım işleminin yapıldığı yük hücrelerinin (loadcell) çok pahalı olması nedeniyle loadcellerin yerine kullanılabilir bir giriş kartı tasarlanmıştır. Tasarlanan bu devre güç kaynağından elde edilen 12V ile beslenmekte ve 3 adet 0 – 10 V ayarlanabilir çıkış vermektedir. 0-10 V ayarlı çıkışlar loadcell yerine kullanılmıştır. Bu trimpotlarla ayarlanarak elde edilen değişken sonuçlar, PLC'nin analog modülüne aktararak giriş elde edilir.



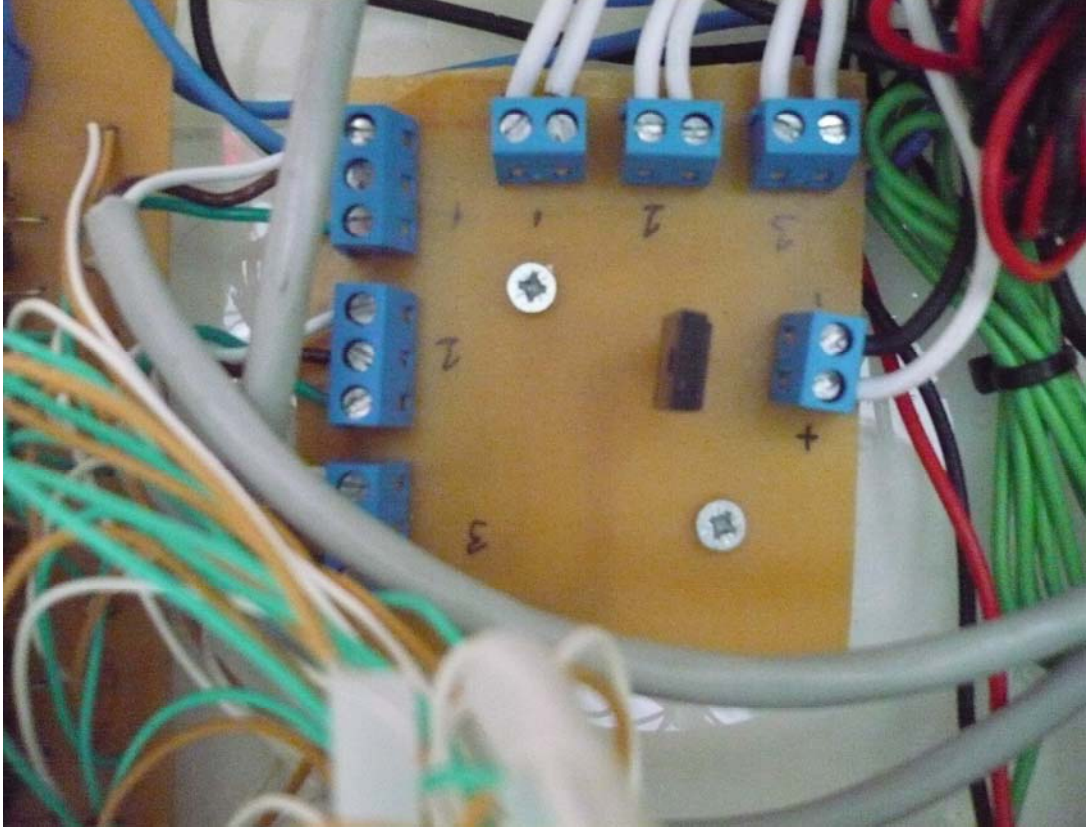
Şekil 4.3. PLC Giriş Kartının Şeması



Şekil 4.4. PLC Giriş Kartının Önden Görünüşü



Şekil 4.5. PLC Giriş Kartının Arkadan Görünüşü

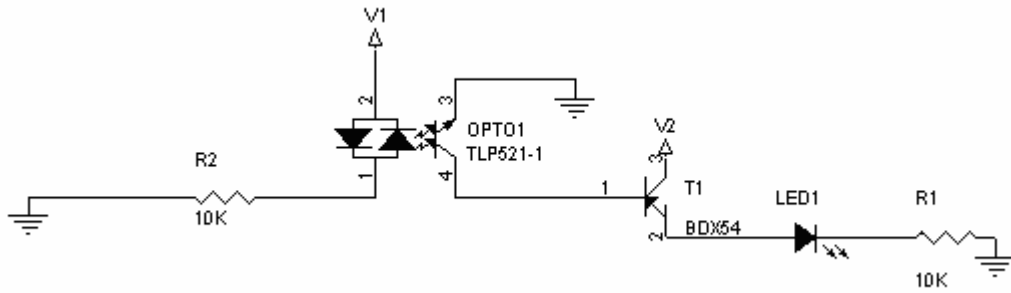


Şekil 4.6. Tasarlanan Giriş Kartı

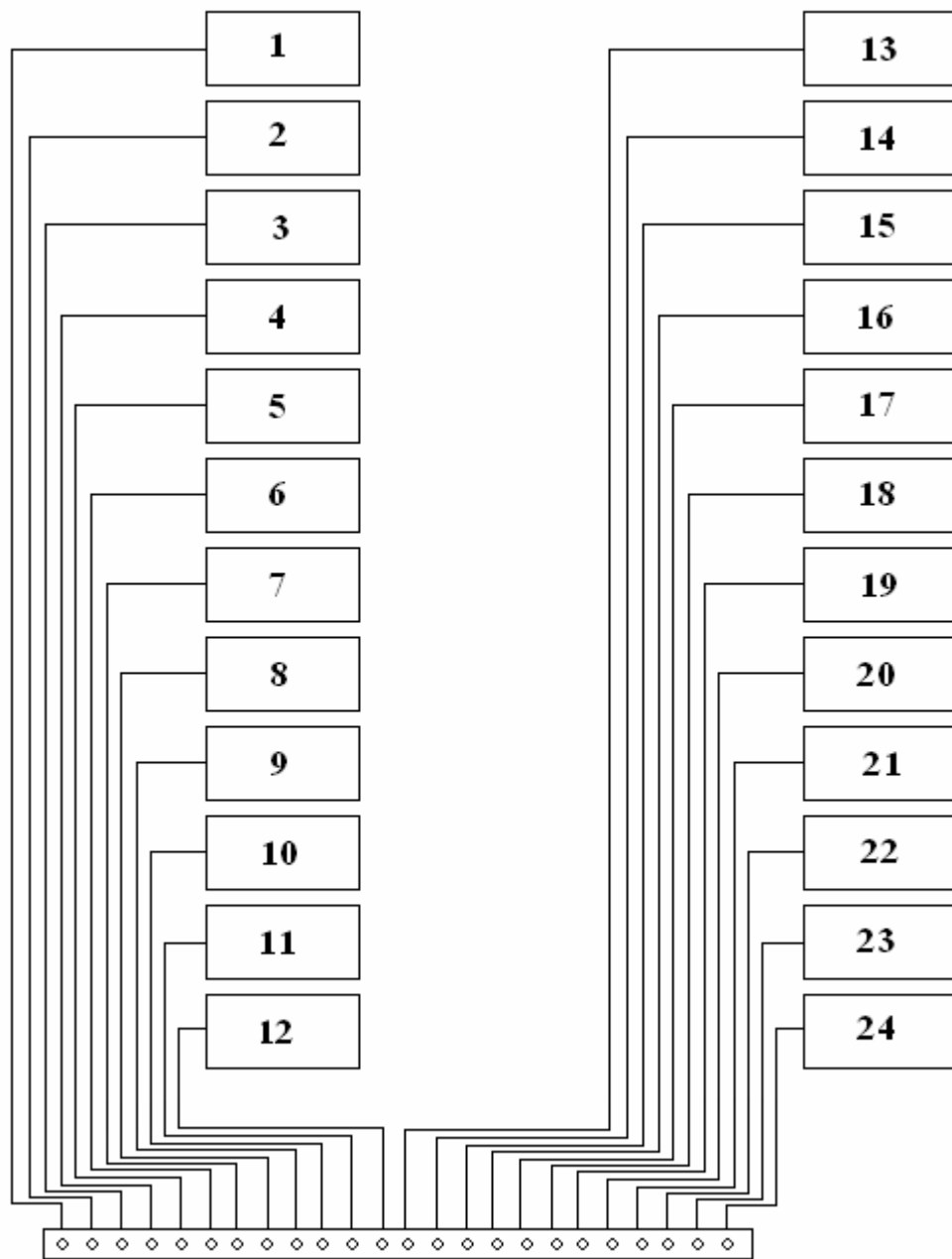
4.2 Tasarlanan PLC Çıkış Kartı

PLC'nin çıkışlarını korumak amacıyla çıkışları elektrik kumanda devresinden optik olarak ayıran bir PLC çıkış devresi tasarlanmıştır. Tasarlanan bu kartla aynı zamanda program çalışırken PLC'nin aktif olan çıkışlarını o çıkışa bağlı ledlerin yakılarak gösterilmesi de sağlanmıştır.

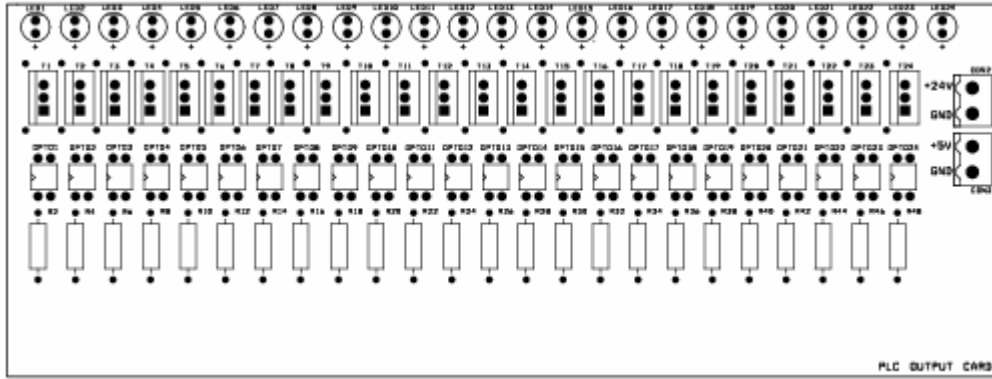
Devrede anahtarlama yapmak amacıyla TLP521-1 serili optocoupler kullanılmıştır. Bu optocouplerler güç kaynağı ile beslenen PLC'den alınan 5V ile sürülmektedir. Güç kaynağından alınan 24V gerilim ise BDX serisi pnp transistör ile çıkışlara verilmiştir. Elde edilen bu çıkışlar ledleri sürmektedir. Bu tasarımda ventil ve diğer tüm çıkışlar ledlerle simgelenmiştir. Şekil 4.6.'da PLC çıkış kartının bir bloğunun seması verilmiştir. Bu kart toplam yirmi dört çıkıştan meydana gelmektedir. Her bir blok özdeştir. Şekil 4.7'de ise PLC çıkış kartının blok şeması verilmiştir.



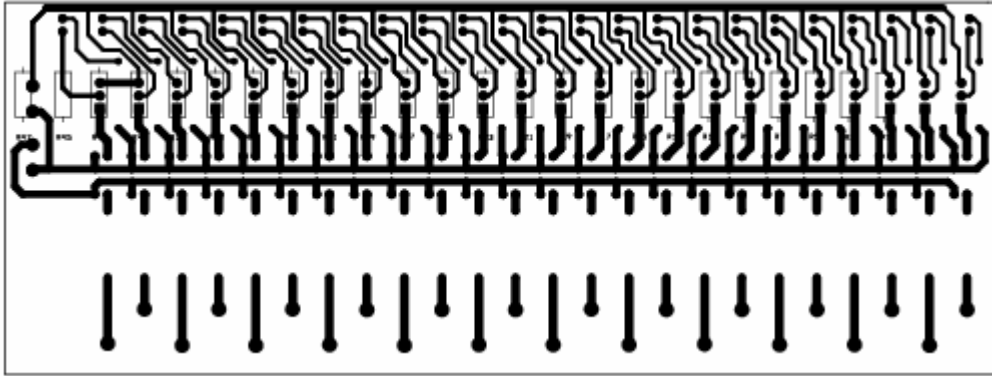
Şekil 4.7. PLC Çıkış Kartının Şeması



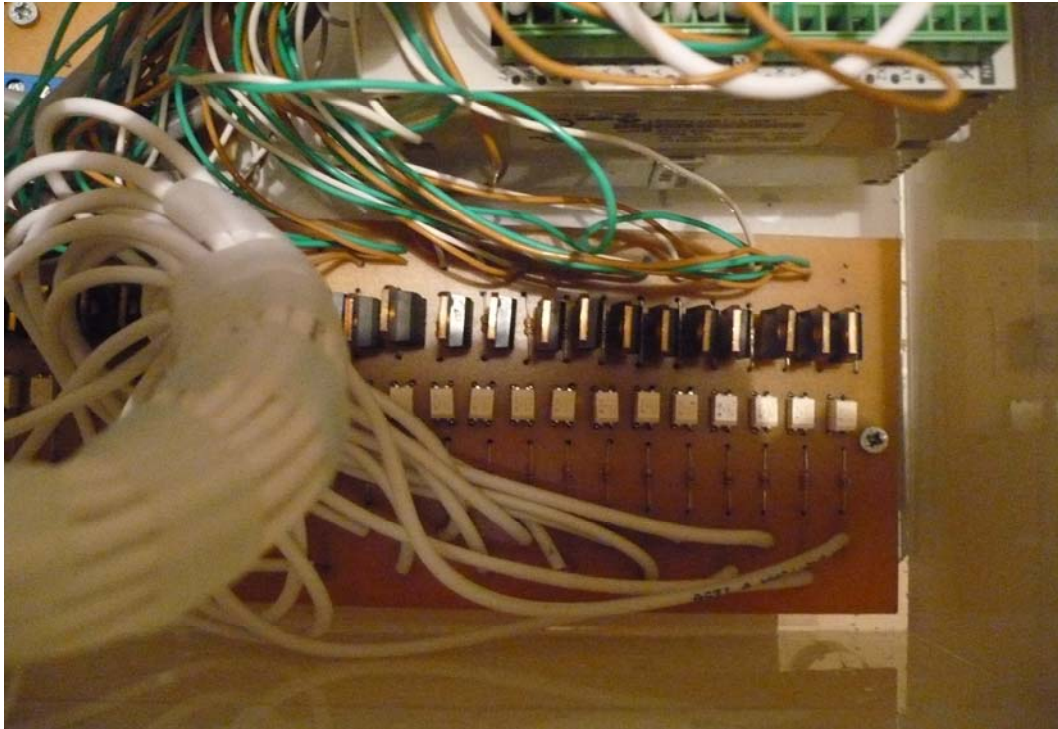
Şekil 4.8. PLC Çıkış Kartının Blok Şeması



Şekil 4.9. PLC Çıkış Kartının Önden Görünüşü



Şekil 4.10. PLC Çıkış Kartının Arkadan Görünüşü



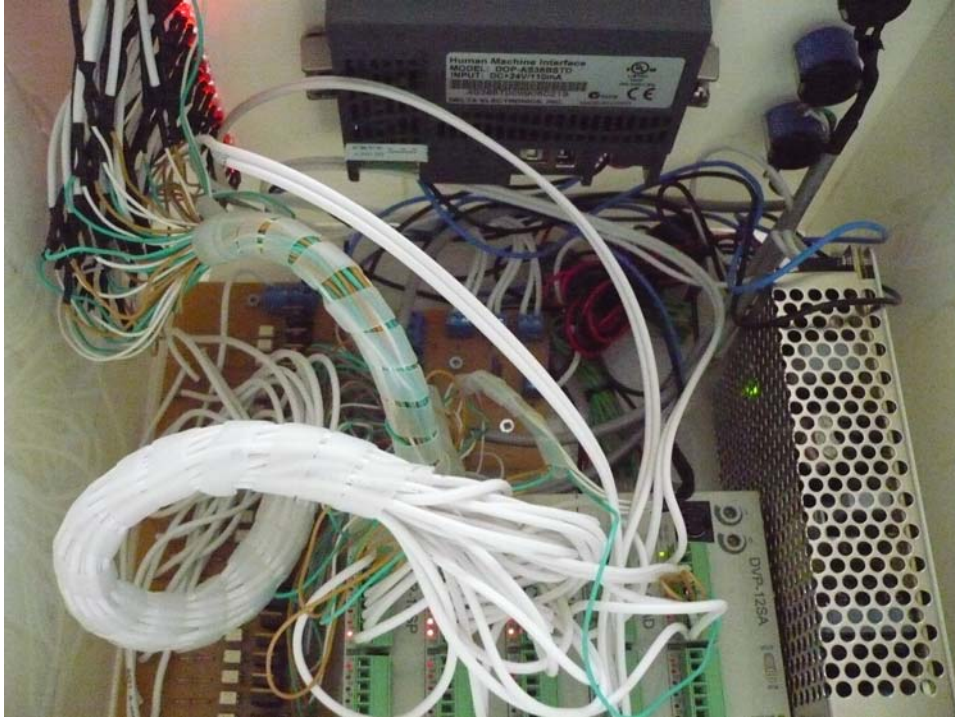
Şekil 4.11. Tasarlanan Çıkış Kartı

5.SONUÇ VE ÖNERİLER

Bu çalışmada hazır beton santrali otomasyonunun düşük maliyetli bir PLC olan Delta marka PLC ile gerçekleştirilmesi amaçlanmıştır. Hazır beton santrali otomasyonunu gerçekleştirirken giriş ve çıkış kartı olmak üzere iki baskı devre kartı tasarlanmıştır. Loadcell'in temini maliyetli olacağı ve sisteme yük uygulanamayacağı için PLC' nin analog girişlerini simüle edecek bir giriş kartı ile gerçekleştirilen otomasyon sisteminde PLC'den alınan çıkışları simüle etmek amacıyla bir çıkış kartı tasarlanmıştır.

Yapılan bu uygulamada giriş verileri dokunmatik ekran ile kullanıcı tarafından girilmektedir. Analog çıkışlar dokunmatik ekran üzerinden okunmakta, dijital çıkışlar ise tasarlanan çıkış kartına bağlı ledlerin yanmasıyla görülmektedir.

Şekil 4.12' de görüldüğü gibi tasarlanan cihazda bir adet güç kaynağı, Delta marka PLC ile dokunmatik ekran, giriş kartı ve çıkış kartı bulunmaktadır. Güç kaynağı 5V, 12V ve 24V olmak üzere üç farklı çıkış vermektedir. 12V giriş kartında, 5V ile 24V ise çıkış kartında kullanılmaktadır.



Şekil 4.12. Tasarlanan Cihazın İçten Görünüşü

PLC ile dokunmatik ekran 24V ile beslenmektedir. PLC, bilgisayar ile RS232 haberleşme protokolü ile haberleşirken dokunmatik ekran ile ise RS422 haberleşme protokolü ile haberleşmektedir.



Şekil 4.13. Sistemin Çalışma Anı

Bu tip çalışmalarda PLC, sensör gibi malzemeler tasarlanan projeye uygun seçilmelidir. Bu işlemlerin yapılmasında dikkat edilecek husus, kullanılacak giriş-çıkış sayısına uygun tasarım yapılması ve maliyeti en düşük elemanların seçilmesine dikkat edilmelidir.

Daha ileri bir çalışma olarak PLC'nin internet bağlantısı sağlanarak tespit edilen arızaları ya da oluşturulan raporları merkezi birime gönderen ya da hazırlanan bir SCADA programıyla bilgisayara gönderen bir sistem de geliştirilebilir.

KAYNAKLAR

1. Kurtulan, S., “PLC ile Endüstriyel Otomasyon Cilt”, *Birsen Yayınevi*, İstanbul, 1-17 (2003).
2. Özcan, M., “Otomasyon Sistemlerinde PLC Uygulamaları”, *Atlas Yayın Dağıtım*, İstanbul,1, 92 (2004).
3. Yılmaz, N., “PLC (Programlanabilir Mantık Kontrol Cihazı) İle Hareketin Üç Eksende Kontrolü”,Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 13-16 (1998).
4. Bayır, R., “Bir üretim bandında üretilen malın kalite kontrolünün PLC kullanılarak gerçekleştirilmesi”, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, (1998)
5. Erdem, F., “SCADA ve otomasyon sistemlerinin dağıtım şebekelerinde uygulanması”, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, (2005).
6. Türk Standartları Enstitüsü, “TS 802 Beton Karışımı Hesap Esasları”, *T.S.E, Ankara*, 3 (1985).
7. Kantar, E., “İzmir Ve Yöresinde Üretilen Hazır Betonların Sınıf Dayanımlarına Göre İstatistiksel İncelenmesi İle Kür Şartlarının Beton Dayanımlarına Olan Etkileri”, Yüksek Lisans Tezi, *Sakarya Üniversitesi Fen Bilimleri Enstitüsü*, Sakarya, 26-30 (1998).
8. Türk Standartları Enstitüsü, “TS EN 197-1 Çimento-Bölüm1: Genel çimentolar-Bileşim, Özellikler ve uygunluk kriterleri”, *T.S.E, Ankara*, 3 (2002).
9. Yardımcı, A., “Santral Çıkışı İle Şantiye Şartlarındaki C 20/25 Ve C 25/30 Hazır Beton Mukavemetinin Karşılaştırılması”, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 3-36 (2005).
10. İnternet: “Agrega”, <http://www.thbb.org>, (2006).
11. İnternet: “Beton”, <http://www.kalitekontrol.org/beton/beton-ve-bilesimi.html>, (2007).
12. Ergün, A.,“Bantlı Konveyörler Hesap ve Konstrüksiyon Esasları”, *T.M.M.O.B Makine Mühendisleri Odası Yayın no 98, Ankara*, 17-18 (1998).
13. Çilek, A., “PLC ve SCADA ile Endüstriyel Otomasyon Uygulaması”, Yüksek Lisans Tezi, *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 4-21 (2005).

14. Delta, "DVP-PLC Application Manual Programming", *Delta Electronics*, Thailand, 1-450 (2006).

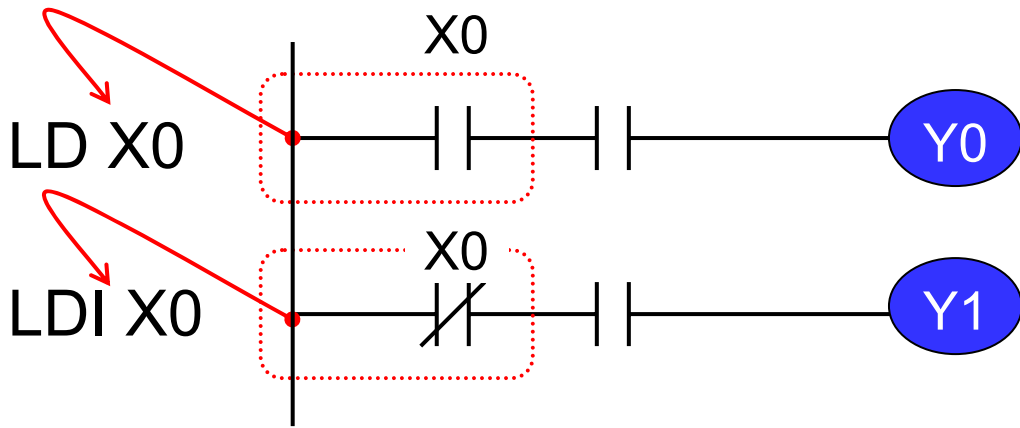
EK-1

EK(1)**1. DELTA PLClerde temel komutlar**

LD/LDI: Başlangıç Komutu NA kontak / Başlangıç Komutu NK kontak

LD normalde açık A kontağı (NA) uygulama başlangıç komutu.

LDI normalde kapalı B kontağı (NK) uygulama başlangıç komutu.

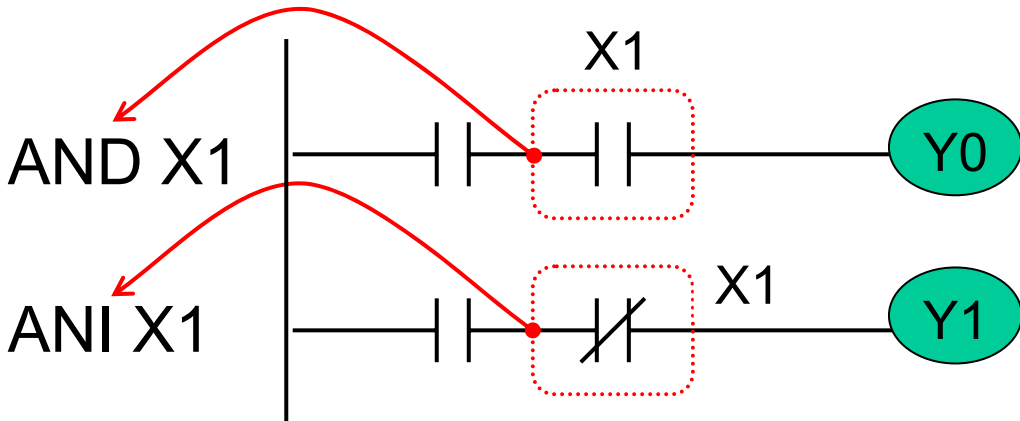


AND/ANI : NA Seri bağlantı komutu / NK Seri bağlantı Komutu

AND normalde açık (NA) seri bağlantı komutu (A kontak)

ANI normalde kapalı (NK) seri bağlantı komutu (B kontak)

Belirlenen datanın ON/OFF durumunu okur, AND uygulamasını gerçekleştirir ve diğer şartların durumuna göre çıkışı yönlendirir.

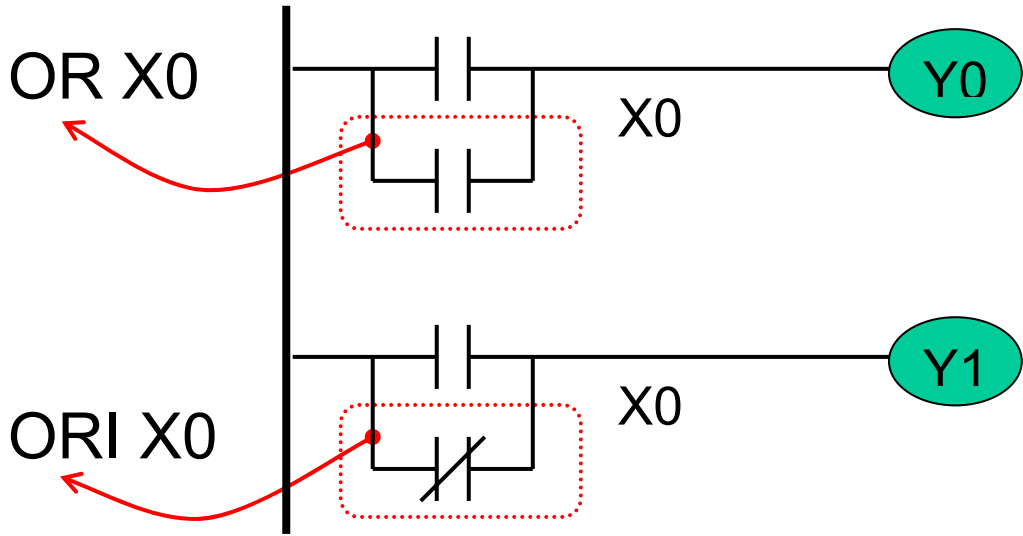


OR/ORI : NA Paralel bağlantı komutu / NK Paralel bağlantı komutu

OR normalde açık (NA) paralel bağlantı komutu (A kontak)

ORI normalde kapalı (NK) paralel bağlantı komutu (B kontak).

Belirlenen datanın ON/OFF durumunu okur, OR uygulamasını gerçekleştirir ve diğer şartların durumuna göre çıkışı yönlendirir.

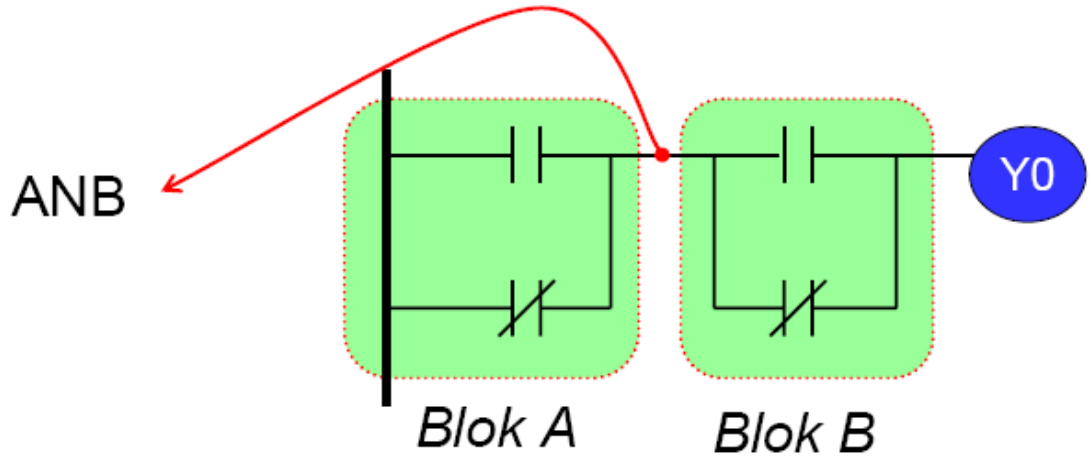


ANB : İki bloğun seri bağlantısı

A ve B bloklarının seri bağlantısını sağlayan çalışma komutudur.

ANB kontak sembolü değil bağlantı sembolüdür.

ANB komutu ile ardı ardına 8 tane blok yazılabilir. Eğer ard arda 8'den çok blok yazılırsa, PLC self-test yaparken bunu algılar ve arıza verir. İlgili arıza kodu D1004 özel data registerine kaydedilir.

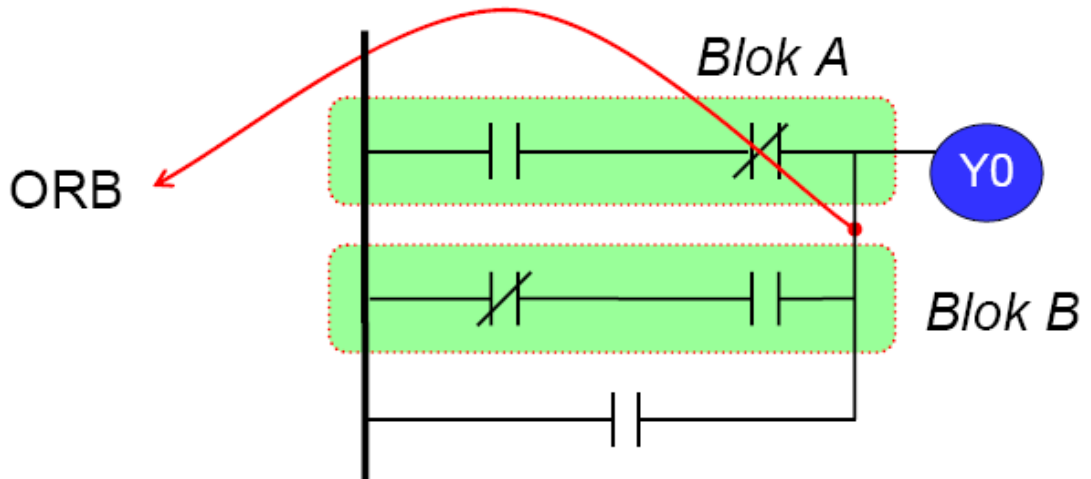


ORB : iki bloğun paralel bağlantısı

A ve B bloklarının paralel bağlantısını sağlayan çalışma komutudur.

ORB iki veya daha fazla kontak bulunan blokların birbirine paralel bağlantısında kullanılır. Sadece tek kontak bulunan bloklar bağlanırken OR ve ORI komutları kullanılır ORB komutuna ihtiyaç yoktur. ORB bir kontak değil bir bağlantı sembolüdür.

ORB komutu ile ardı ardına 8 tane blok yazılabilir. Eğer ar darda 8'den çok blok yazılırsa PLC Self-test yaparken bunu algılar ve arıza verir. İlgili arıza kodu özel data register D1004 e kaydedilir.



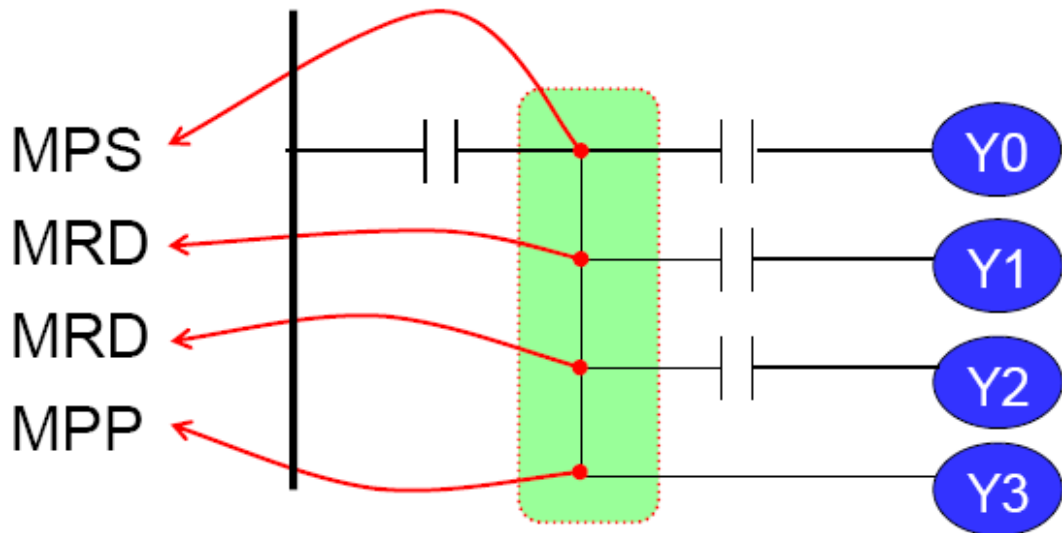
MPS/MRD/MPP:

MPS: Kendinden önce gelen dataların “ON/OFF” durumunu saklayan çalışma komutu, MRD: MPS tarafından saklanan datanın durumunu okur, MPP: MPS tarafından saklanan datanın durumunu okur, ve daha sonra bu datanın durumunu siler.

MPS; Kendisinden önce gelen datanın ON/OFF durumunu saklayan çalışma komutudur. MPS komutu 8 adede kadar kullanılabilir. Ayrıca, MPS komutları arasında MPP komutu Kullanılırsa MPS komutu kullanma sayısı 1 azalır. Bloğun ilk satırındadır.

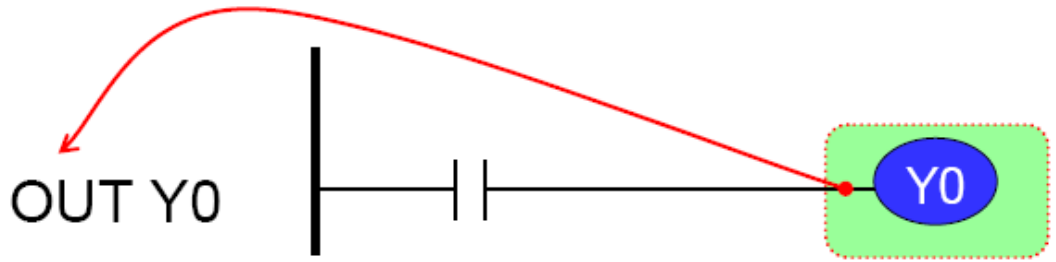
MRD: MPS komutu tarafından saklanan datanın durumunu okur ve bu durumu sonraki Step başlangıcında kullanır. Bloğun orta satırlarındadır.

MPP: MPS komutu tarafından saklanan datanın durumunu okur ve işlem bittikten sonra bu datanın durumunu siler.



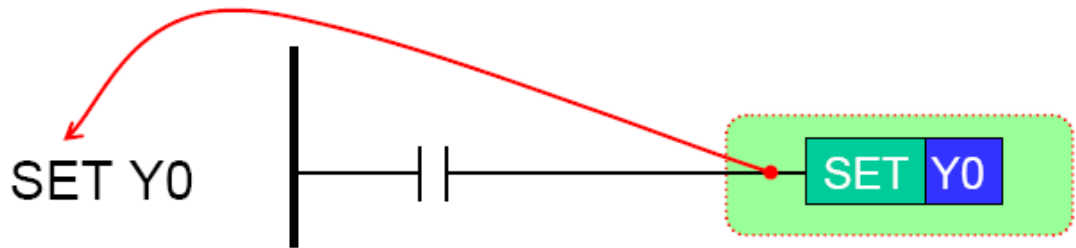
OUT : Çıkış komutu.

Bu komut kendinden önceki şartın sonucuna göre çıkış verir.



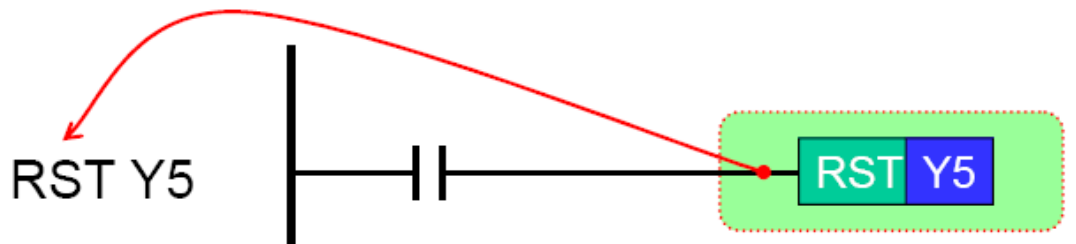
SET: Çıkış bobinini, yardımcı röleyi, veya özel röleyi ON yapar.

SET komutu şartı ON olduğunda, komutta kullanılan bit ON olur. Komut şartı OFF olduğu zaman ilgili bit ON kalmaya devam eder. Komutta kullanılan bit RST komutu kullanılmadan OFF olmaz.



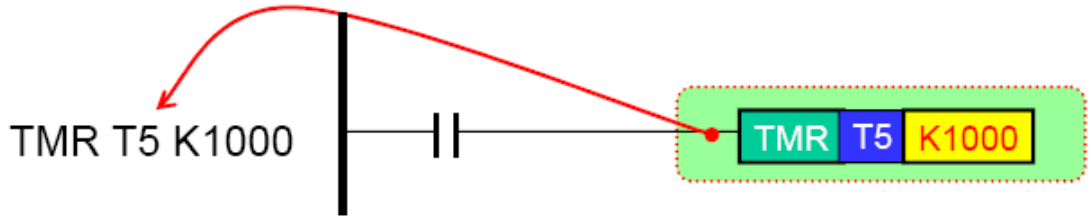
RST : Çıkış bobini, yardımcı röle veya özel röleyi RESET (OFF)' ler.

RST komutunun şartı ON olduğu zaman, komutta belirtilen bit OFF olur.



TMR : Timer

TMR komutunun önündeki şart ON olduğu zaman, zamanlayıcı bobini ON olur ve SET değerine doğru artmaya başlar. Zaman dolduğunda sayılan değer \geq set değeri, zamanlayıcı kontağı ON olur.

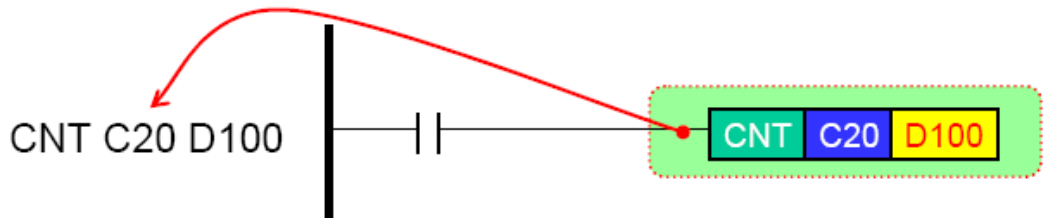


TMR komutunun önündeki şart OFF olduğu zaman zamanlayıcı değeri "0" olur ve ilgili zamanlayıcı kontağı OFF olur.

TMR zamanı dolduktan sonra, RST komutu kullanılmadan kontak durumu değişmez. A negatif sayılar -32768 den -1 e SET değeri olarak ayarlanamaz.

CNT : Sayıcı

CNT komutunun önündeki şart OFF'dan ON'a geçtiği zaman sayıcı mevcut değerinin üzerine 1 ekler. Sayma değeri sayıcı istenilen değere ulaştıktan sonra (sayılan değer = set değeri), ilgili sayıcı kontağı ON olur.

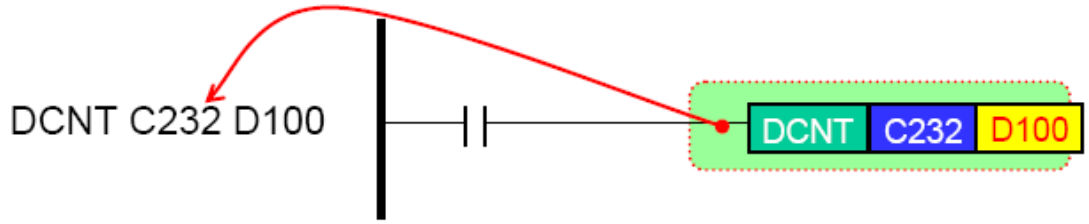


CNT komutunun önündeki şart sürekli geldiği zaman sayma işlemi sadece 1 artar sürekli artmaz. (Onun için sayma girişi olarak darbe kontağı kullanmaya gerek yoktur.)

Sayma işlemi tamamlandıktan sonra ilgili CNT kontağı ON olur ve RST komutu ile sıfırlanmadıkça OFF olmaz. SET değeri negatif sayı olamaz. -32768 ~ -1.

DCNT

C232 - C255 aralığındaki sayıcılar yüksek hızlı sayıcı olarak kullanılır. DCNT komutunun önündeki şart sağlandığı zaman (DCNT ON/OFF şartı) hızlı sayıcı aktif olur ve X00~X03 girişlerinden gelen tetikleme darbelerine göre sayıcının değeri artar. Tetikleme sinyali OFF'dan ON'a geçtiği zaman sayma işlemi gerçekleşir. Eğer sürekli ON olursa sayma olmaz.

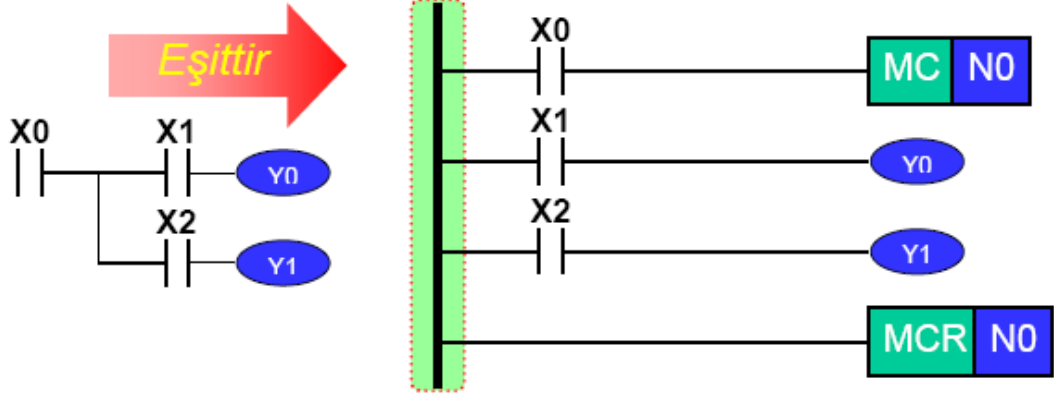


Sayıcayı sıfırlamak için RST C2XX komutu kullanılmalıdır.

MC / MCR : MC Master Kontrol başlangıç komutu. MCR Master Kontrol reset komutu.

MC master control, başlama komutudur. MC önündeki ON/OFF şartı durumuna göre MC MCR arası çalışma belirlenir. MC komutunun önündeki şart ON olduğu zaman MC ve MCR komutları arasındaki program PLC tarafından işlenir. Yani tarama zamanında bir kısalma olmaz. MC komutunun önündeki şart OFF olduğu zaman, MC - MCR arasındaki program işlenmez ve zamanlayıcı değeri "0" olur, çıkışların hepsi OFF olur SET ve RST durumlarını muhafaza eder, CNT bobin OFF olur mevcut sayma değerini korur.

MCR master kontrol reset olup master kontrol aralığının sonu olduğunu gösterir. MCR komutundan önce kontak kullanılmaz. MC komutları kümeleme içinde kullanılır. MC komutları aralığı komutlarla birlikte kullanılan numaralarla belirlenir. Kümeleme numaraları N0 - N7 arası girilebilir.



LDP/LDF ANDP/ANDF ORP/ORF :

LDP: Yükselen kenar algılama çalışma komutu

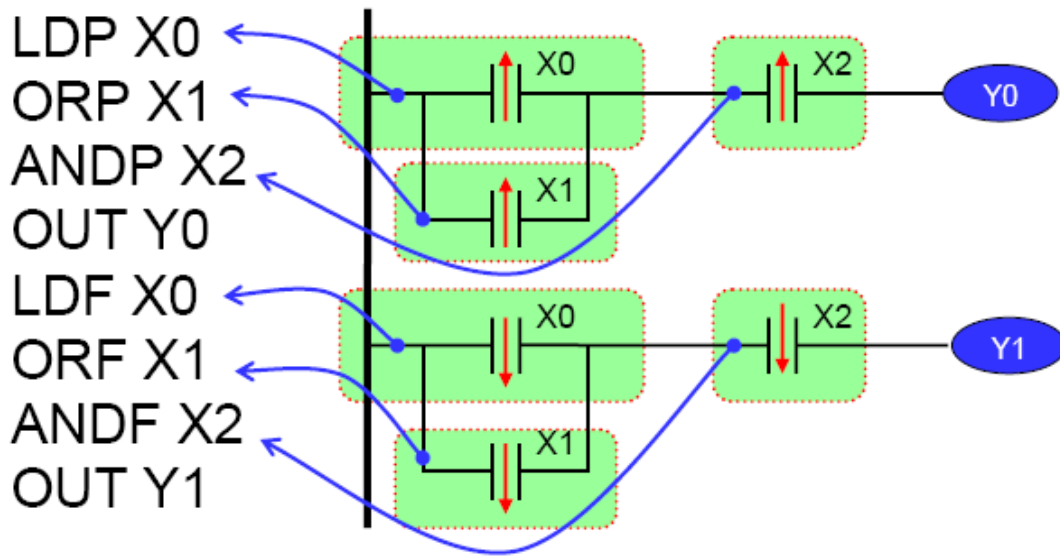
LDF: Düşen kenar algılama çalışma komutu

ANDP: Yükselen kenar algılama seri bağlantı komutu

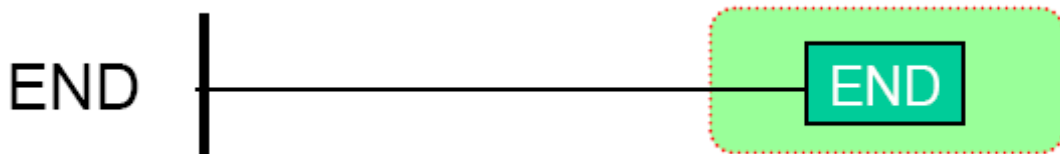
ANDF: Düşen kenar algılama seri bağlantı komutu

ORP: Yükselen kenar algılama paralel bağlantı komutu

ORF: Düşen kenar algılama paralel bağlantı komutu

**END :**

Bu komut programın sonu olduğunu gösterir. Buradan sonra tarama başa döner. (Step 0) END komutu normal çalışan programın içinde (ortalarda) bir yerde veya alt programın (subroutine) içinde kullanılamaz. Eğer programın içinde kullanılması gerekiyorsa FEND komutu kullanılmalıdır. (Subroutine).

**NOP :**

Bu komut uygulama olmadığı anlamına gelir ve programın çalışmasına hiçbir etki etmez. NOP aşağıdaki durumlarda kullanılır:

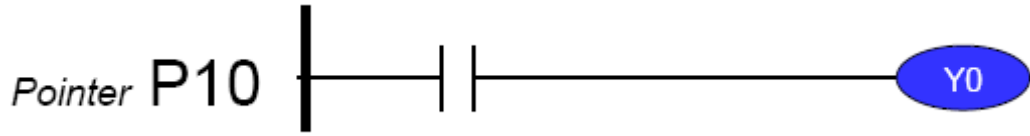
- Mevcut programı test etmek için boşluk sağlar.

- Step numaralarını deęiřtirmeden herhangi bir komutu deęiřtirmek için üzerine NOP konur.
- Geçici olarak komutu silmek için kullanılır.

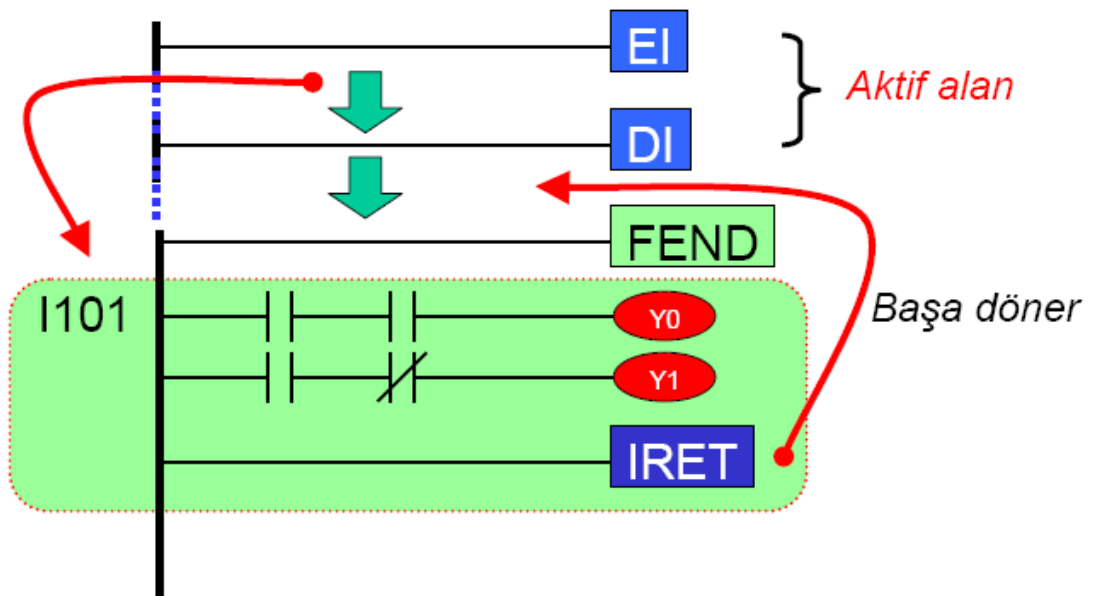
P / I : Pointerler (P) / Interrupt Pointerler

Pointerler atlama (JUMP) komutları ile birlikte (CJ, CALL) iki farklı şekilde kullanılabilir.

- CJ komutu ile P adresi belirtilen hedef program satırına atlayarak.
- CALL komutu P adresi belirtilen hedef alt program bloęuna atlayarak.

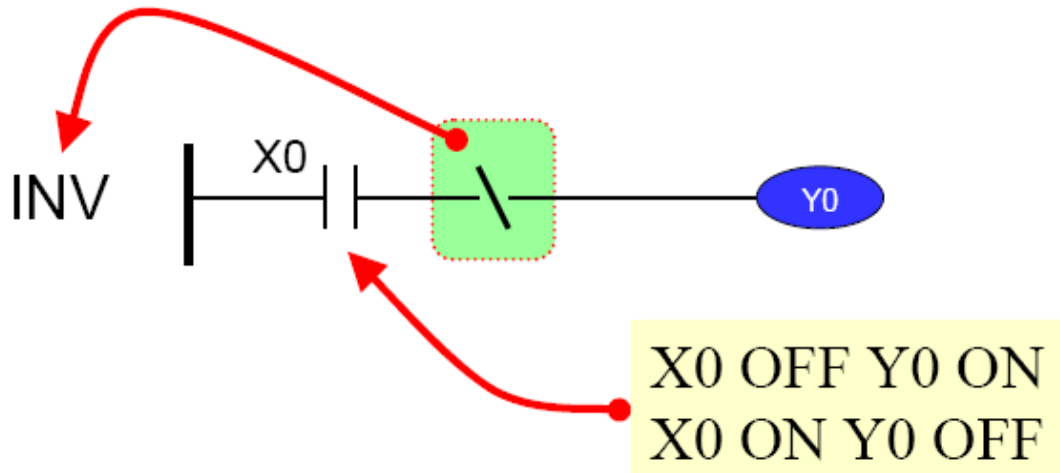


Interrupt pointerler interrupt programının başında numarası ile belirtilir. Her Interrupt programı interrupt pointer ile başlar ve IRET komutu kullanılarak sonlandırılır.

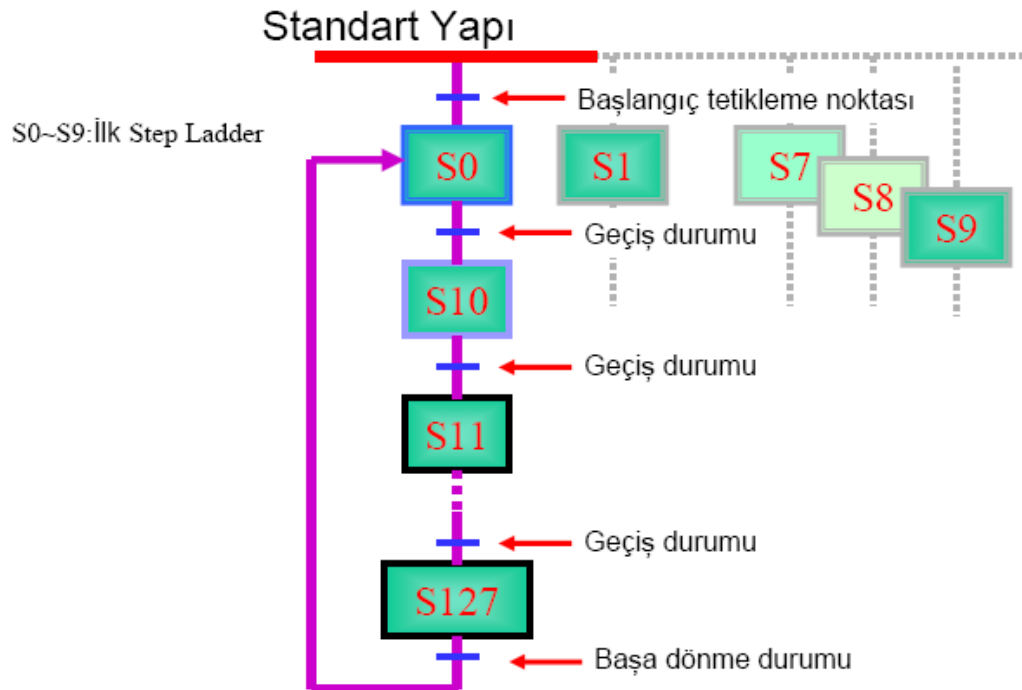


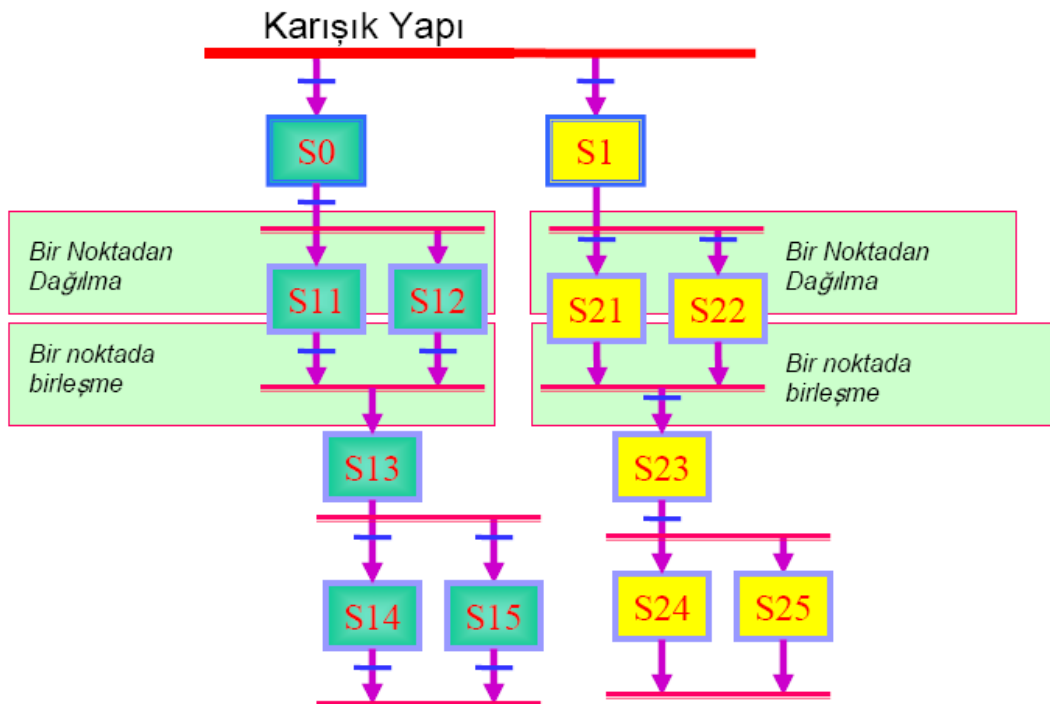
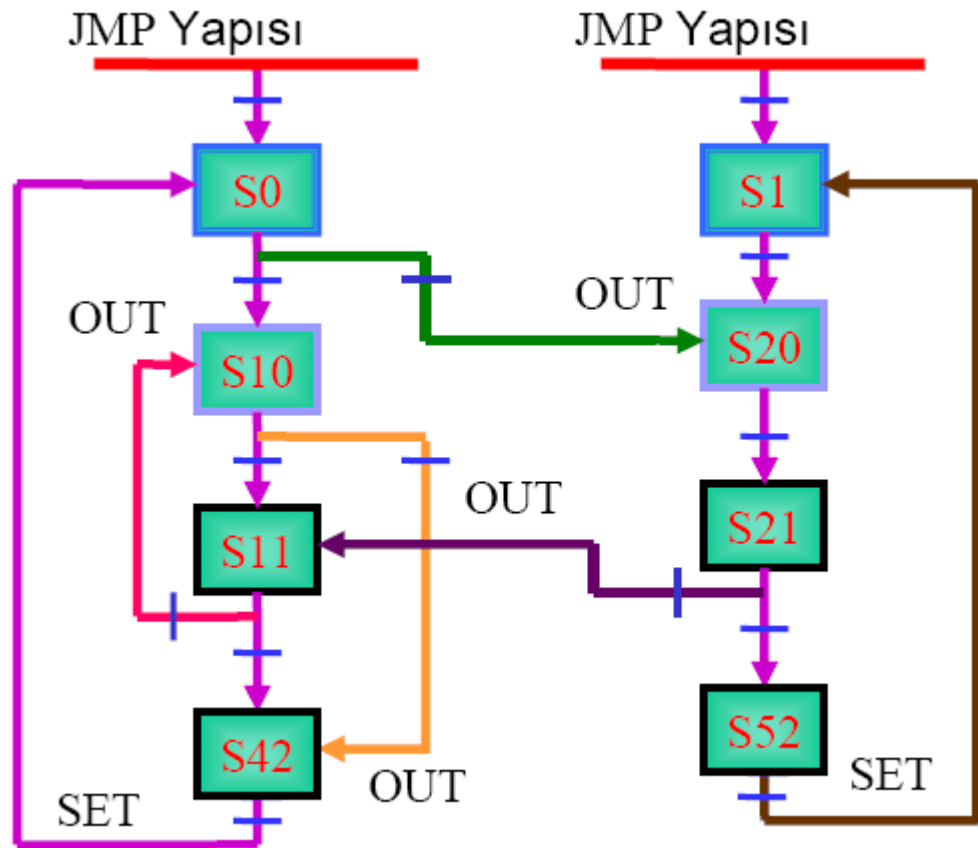
INV :

Kendinden önceki giriş şartını tersleyerek son durumu çıkışa iletir.



STL / RET: Step geçişi ladder başlangıř/bitiş komutları(SFC)

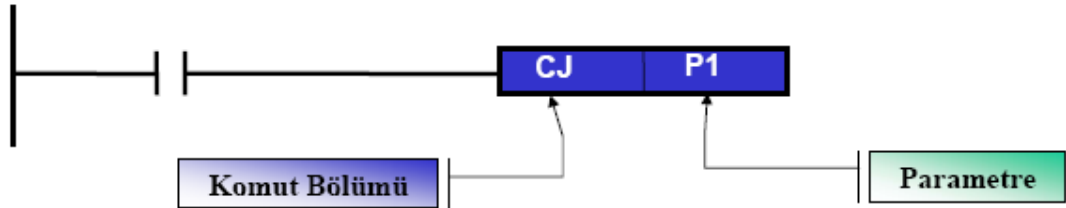




2. DELTA PLC lerde uygulama komutları

2.1 Uygulama komutları yapısı

Çoğu komutlar aşağıdaki gibi komut bölümü ve parametre olarak ayrılır:

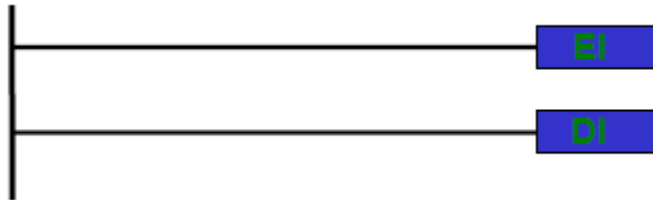


Komut Bölümü : Fonksiyonu gösterir.

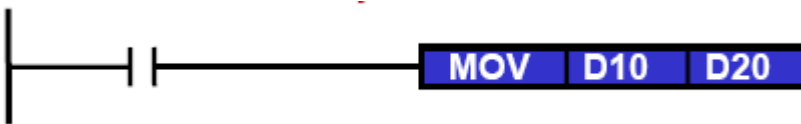
Parametre : Bu komut ile kullanılacak datayı gösterir.

Uygulama Komutları Yapısında Komut bölümü ve Parametre aşağıdakiler gibi birleştirilerek kullanılabilir:

Sadece Komut Bölümü



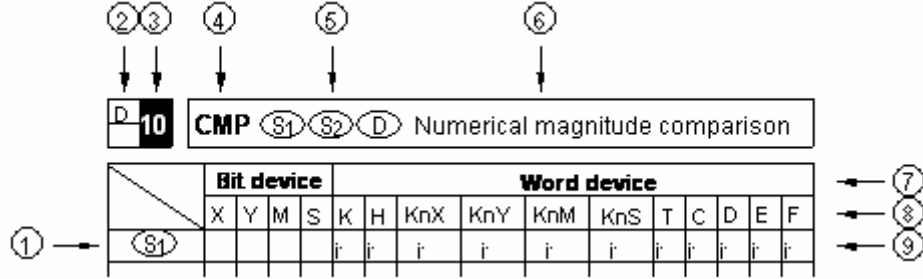
Komut Bölümü + Kaynak Parametre + Hedef Parametre



Komut Bölümü + Kaynak 1 Parametre + Kaynak 2 Parametre + Hedef 1 Parametre + Hedef 2 Parametre



2.2 Komut yapısı

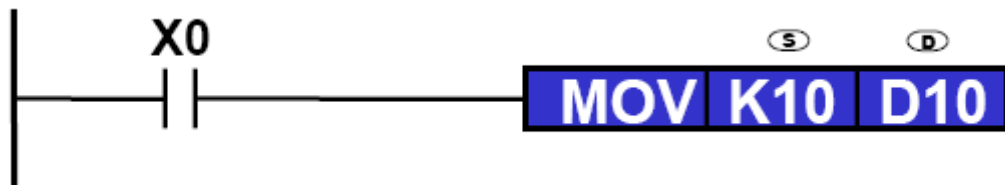


Açıklamalar:

- ① Komut Birimleri
- ② 32-bit komutu gösterir. Örnekte olduğu gibi komutun başına D harfi gelir.
- ③ API numarasını gösterir.
- ④ Komut sembolü
- ⑤ Komutla kullanılacak parametreleri gösterir.
- ⑥ Komut açıklaması
- ⑦ Parametre Tipleri
- ⑧ Parametre Adı
- ⑨ simgesi ile gösterilen parametreler bu komutla kullanılabilir.

2.3 Uygulama komutlarına giriş

PLC'ye HPP ile program yüklendiği zaman API numaraları kullanılır.



Bu komutla deki değer 'ye transfer olur.

(S) :

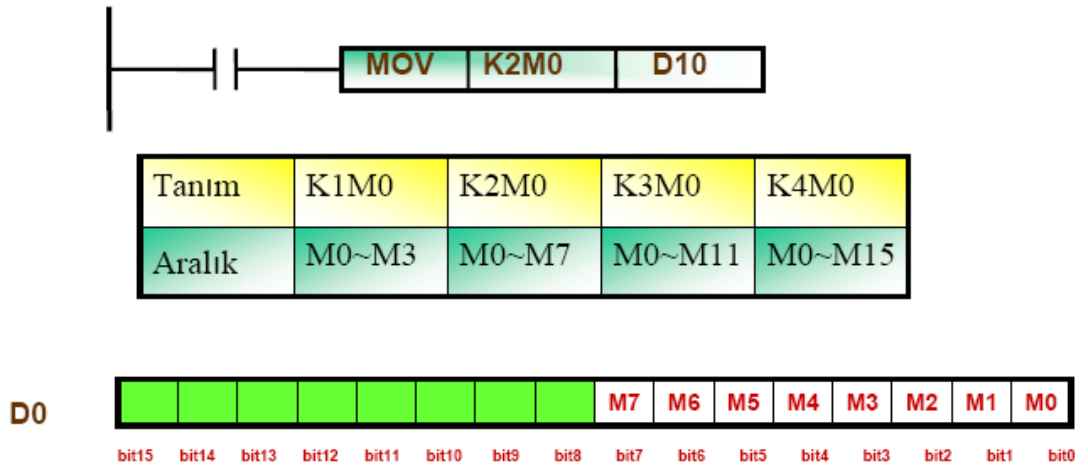
1. Uygulamada kullanılan kaynak datadır.
2. Kaynak data aşağıdakiler olabilir.
 - a. Sabit Sayı (K,H) : Program çalışırken değişmez.
 - b. Bit, Word (M,D) : Program çalışırken değişebilir.
3. 1'den fazla Kaynak Parametresi varsa S1, S2 kullanılır.

(D) :

1. Uygulama Sonucunun kaydedildiği hedef datadır.
2. 1'den fazla hedef parametre varsa D1, D2 kullanılır.

• Data Formatı

X, Y, M, S Sadece tek nokta ON/OFF olabilir ve BIT olarak tanımlanır; bununla birlikte, 16-bit (veya 32 bit) T, C, D data registerlar olup WORD olarak tanımlanır. Ayrıca X, Y, M ve S'nin önüne Kn ekleyerek WORD olarak tanımlanabilir, n=1 olduğu zaman 4-bit anlamına gelir. Böylece K1'den K4'e kadar kullanarak 16-bit tanımlanabilir ve K1'den K8'e kadar kullanılarak 32-bit tanımlanabilir. Örneğin K2M0 M0-M7 arası 8-Bit'i ifade eder.



• Bit İşleme

Bit parametreleri belirtildiğinde (X,Y,M,S) bit işleme yerine getirilir.

1-bit işleme : Temel Komutlarda genelde tek bit tanımlanabilir.

ÖR : LD X10, Out Y0

Digit işleme : Uygulama Komutlarında bitler için digit sayısı belirtilmelidir. 4 ve katları olarak 32-bite kadar tanımlanabilir.

1-) 16-bit Komut : K1-K4.

K1 (4 nokta) 0-15

K3 (12 nokta) 0-4095

K2 (8 nokta) 0-255

K4 (16 nokta) -32768 - 32768

2-) 32-bit Komut : K1-K8

K1 (4 nokta) 0-15

K5 (20 nokta) 0-1048575

K2 (8 nokta) 0-255

K6 (24 nokta) 0-167772165

K3 (12 nokta) 0-4095

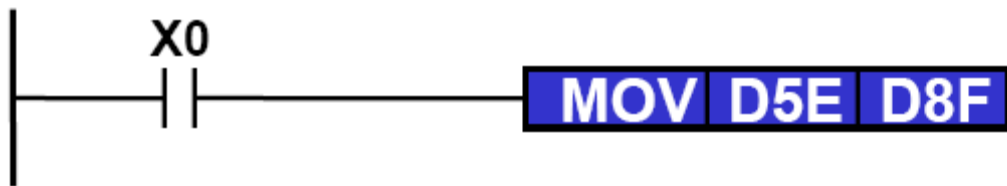
K7 (28 nokta) 0-268435455

K4 (16 nokta) 0-65535

K8 (32 nokta) -2147483648-2147483648

- **Dolaylı Adresleme Metodu**

E ve F komutlarda kullanılan sabitler olup transfer ve karşılaştırma işlemleri yapılabilir. Dolaylı adresleme WORD parametreleri ile kullanılabilirken (KnX, KnY, KnM, KnS, T, C, D) bit parametreleri ile kullanılmaz (X,Y,M,S). Dolaylı Adresleme için K ve H Sabitleri ile birlikte kullanılır.



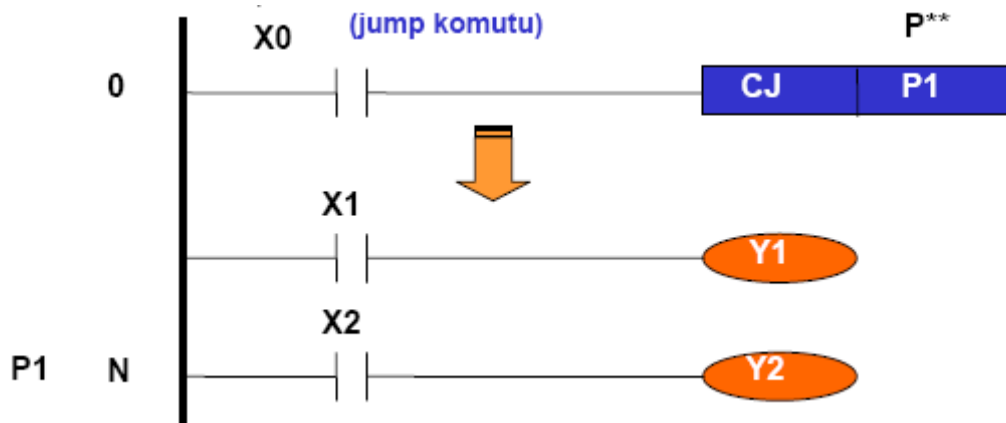
E = 8, F = 14 olduğu zaman
 $D5E = D(5+8) = D13$
 $D8F = D(8+14) = D22$

X0=ON olduğu zaman D13 Değerini
 D22'ye transfer eder.

Bazı komutlar dolaylı adresleme metodunu desteklemez. Bu metodu kullanırken komutun dolaylı adreslemeyi desteklediğinden emin olunmalıdır.

- **CJ (P**)**

Jump Komutu ON olduğu zaman programı komutta belirtilen Pointer numarasından itibaren çalıştırır. Jump Komutu OFF olduğu zaman program normal olarak Jump komutundan bir sonraki satırdan itibaren çalışmaya devam eder.

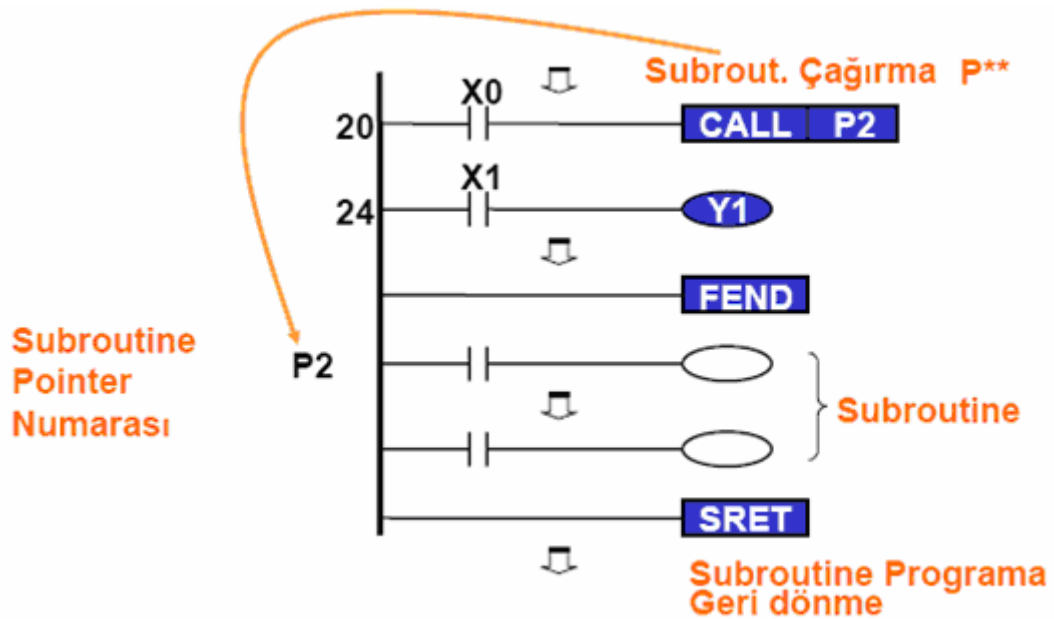


X0=ON olduğu zaman program adresi 0'dan N'ye atlar (P1) ve çalışmasına P1'den devam eder. Aradaki adresleri atlar. Eğer bu aradaki adreslerde bir TIMER varsa; TIMER saymayı durduracak. X0=OFF olduğu zaman program adres 0'dan itibaren işler CJ komutu aktivitesini yitirecek ve TIMER saymaya devam eder.

- **CALL (P**) / SRET**

Pointer (Pxx) tarafından belirlenen subroutine programı yerine getirir. CALL komutundan sonraki satırdan başlayarak normal programı yerine getirir.

- Beş kümeye kadar CALL komut kullanılabilir.
- Pointer (Pxx) tarafından belirlenen subroutine programı yerine getirir.



- SRET komutu gerçekleştiğinde CALL komutundan sonraki satırdan başlayarak normal programı yerine getirir.
- Subroutine programın sonu olduğunu gösterir.
- Örnek programda X0=ON olduğu zaman CALL Komutu çalışmaya başlar ve P2 satırına atlayarak Subroutine çalıştırır. SRET komutu çalışmaya başladığında program 24 numaralı adrese atlayarak çalışmasına devam eder.
- Subroutine program FEND Komutundan sonra çalışmasına devam eder.
- CALL Komutunda kullanılan P0 – P63 CJ Komutunda ikinci kez kullanılamaz.

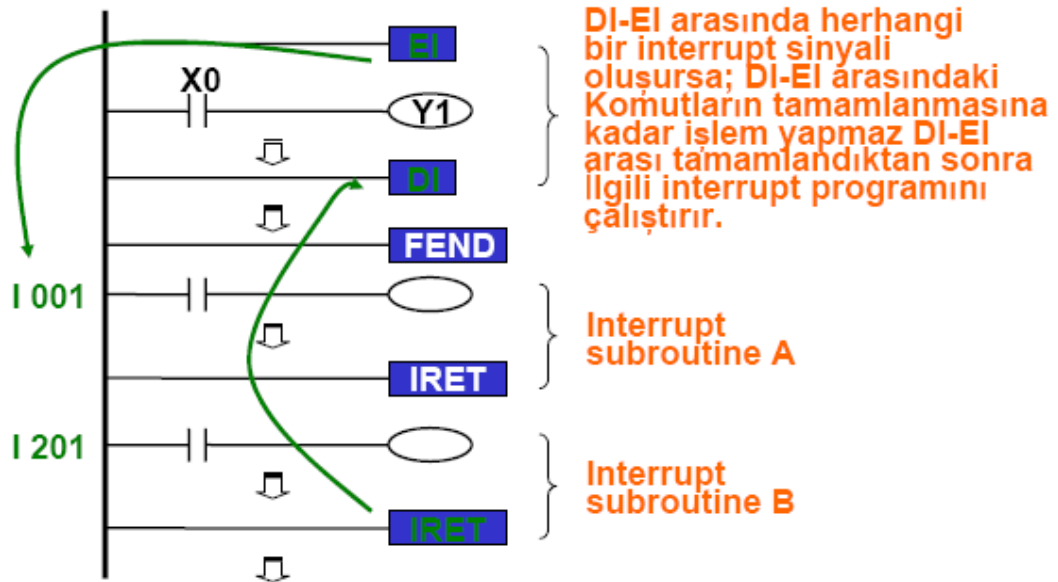
- **IRET**

Interrupt programı işleminin sonu olduğunu gösterir.

- **EI/DI (Enables / Disables the interrupt)**

PLC uygulamasında program EI ve DI arasındaki komutları tararken, eğer X001 ve X002=ON olursa programda Interrupt Service Subroutine I001 ve I002 aktif olur ve IRET komutuna ulaşıldığı zaman normal uygulamaya geri döner.

Interrupt özel yardımcı röleleri M1050-M1053 aktif olursa ilgili interrupt servisi aktivitesini yitirir. I001-I003 interrupt cursor'leri FEND komutundan sonra kullanılmalıdır.



- **FEND**

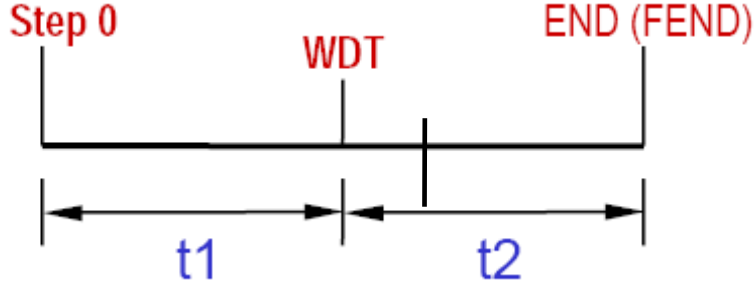
Normal Programı sonlandırır.

- PLC uygulamasında END komutu ile aynı fonksiyona sahiptir.
- CALL komutunda kullanılan Pointer (P) FEND komutundan sonra olmalı ve subroutine sonlandırmak için SRET komutu eklenmelidir.
- Interrupt komutlarında FEND komutundan sonra kullanılmalı ve interrupt sonlandırmak içinde IRET komutu eklenmelidir.
- Eğer birden fazla FEND komutu kullanılıyorsa subroutine ve interrupt servisleri en son FEND komutundan sonra programa işlenmelidir.

- **WDT**

Programdaki Watchdog Timer'ı resetler. Step 0 ile END (FEND) arasındaki zaman Watchdog Timer Set Değerini aştığı durumlarda kullanılır.

Step0 ile WDT arasındaki t_1 süresi ve WDT ile END (FEND) arasındaki t_2 süresi Watchdog Timer Set değerinden büyük olmamalıdır.

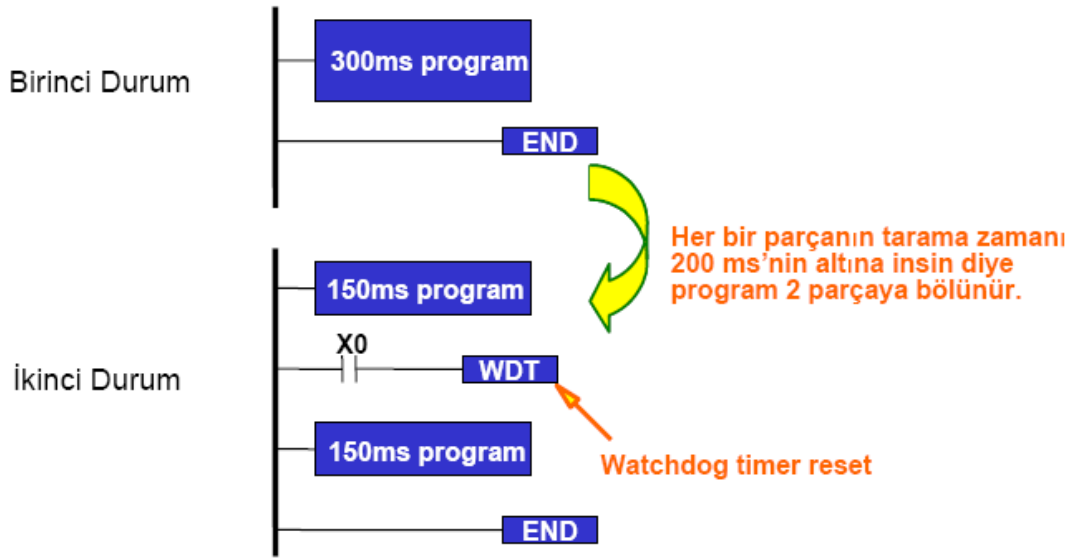


WDT komutu programda iki veya daha fazla kullanılabilir. Bununla birlikte test yapılırken dikkat edilmelidir, çünkü bir problem olursa çıkışlar aniden OFF olamayabilir.

PLC Tarama süresi 200 ms'yi aşarsa (Adres 0 ile END VEYA FEND komutu arasında) PLC ERROR LED'i ON olur. Bu durumda kullanıcı enerjiyi kesip tekrar vermelidir. Enerji geldiğinde PLC otomatik olarak STOP mod'a döner.

Eğer program tarama Zamanı 200ms üzerinde ise kullanıcı programı 2 parçaya bölebilir. Arasına Watchdog Timer koyarak herbir parçanın tarama zamanını 200 ms'nin altına indirebilir.

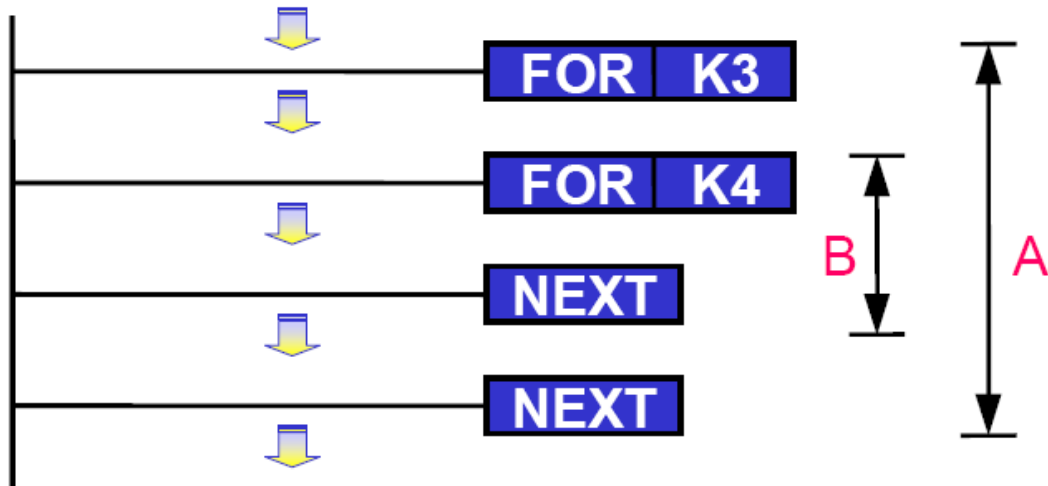
Watchdog Timer süresi 200 ms ve Adres 0 ile END (FEND) komutu arası 300 ms olduğu durumlarda kullanılır.



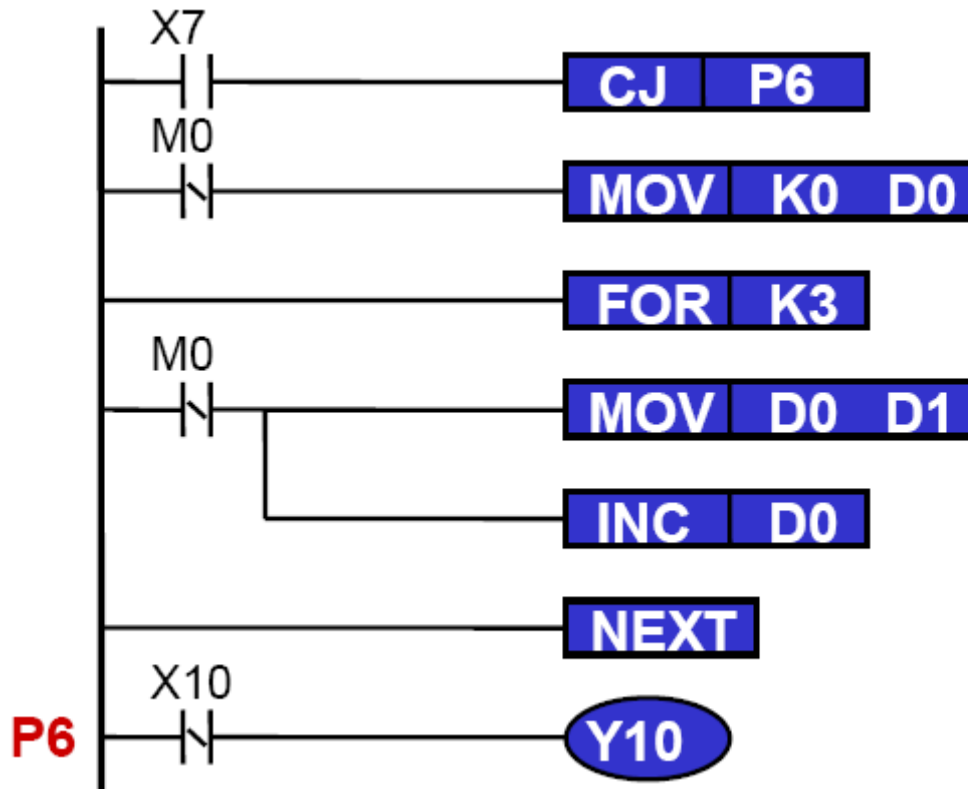
- **FOR / NEXT (Kümelenmiş Döngü Başlangıcı / Bitişi)**

FOR ve NEXT komutları arası kayıtsız şartsız FOR'da belirtilen "n" sayısı kadar işlendikten sonra NEXT komutundan sonraki adımları yerine getirir. Max. 4 Küme oluşturulabilir. "n" değeri 1-32767 arası girilebilir. "n" değeri -32767 – 0 arası girilirse n=1 olarak algılanır ve programda 1 olarak işlenir.

FOR – NEXT komutları aralığı işlenmesi gerekmediği zamanlarda CJ komutu kullanılarak programda bu komutlar atlanabilir.



Eğer programda çok fazla döngü varsa bu durum tarama zamanını uzatır. Tarama zamanının aşılmasına dikkat edilmelidir.



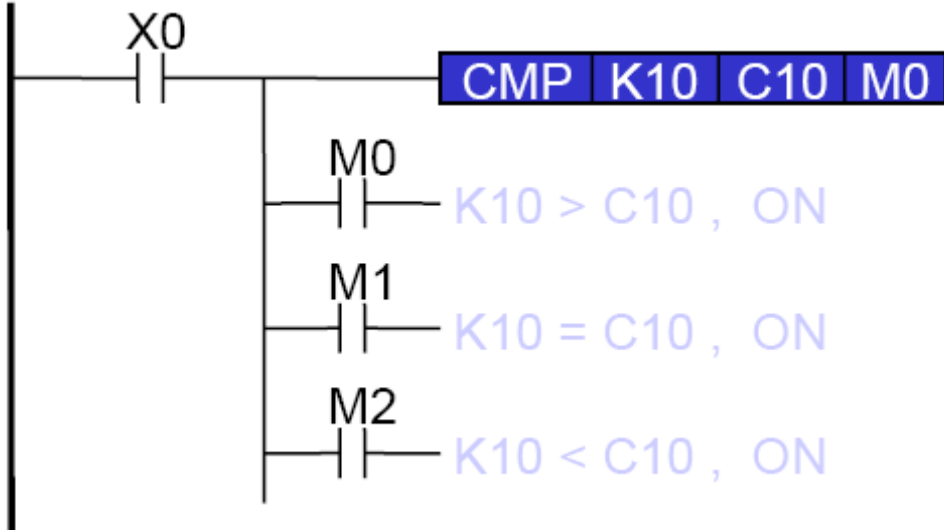
X7=OFF iken FOR-NEXT Komut aralığını yerine getiren X7=ON olduğunda FOR-NEXT komut aralığını işlemeyen programdır.

- **CMP (Karşılaştırma)**

Karşılaştırma uygulama komutu, iki data arasında sayısal büyüklük karşılaştırması yapar.

Bu komut BIN Değerlerini göz önünde tutarak iki data arasında karşılaştırma yapar. Eğer datanın en yüksek biti 1 ise karşılaştırma komutu bu değeri negatif olarak dikkate alır.

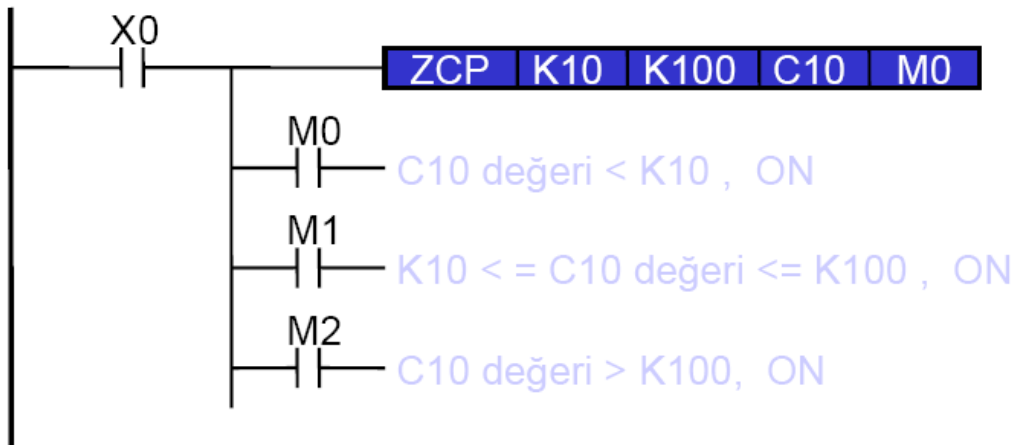
Karşılaştırma komutu $\textcircled{S1}$ de belirtilen data ile $\textcircled{S2}$ de belirtilen data arasında karşılaştırma yapar (=, >, ve <) ve aşağıda görüldüğü gibi kontaklar elde edilebilir:



Yukarıdaki örnekte Bit M0 olduğu zaman otomatik olarak M0,M1,M2 yi kapsar.

- **ZCP (Bölge Karşılaştırma)**

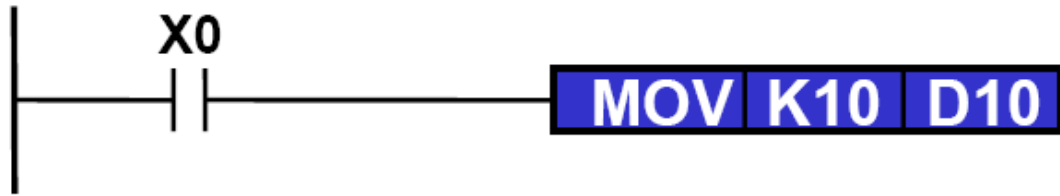
(S) değerini (S1) ve (S2) de belirtilen alt ve üst sınır değerleri ile karşılaştırarak ilgili çıkışı ON yapar. (D) değeri M0 olarak ayarlanırsa M0,M1 veya M2 çıkışlarından biri aktif olur).



- **MOV (Kaynak parametrede bulunan datayı hedef parametreye transfer eder.)**

(S) de belirtilmiş datayı (D) ye transfer eder.

Eğer hesaplama sonucu 32-bit ise (ÖR: MUL Uygulaması) ve data 32-bit hızlı sayıcı çıkışıysa DMOV komutu kullanılmalıdır.



X0=ON olduğunda 10 sayısını D10'un içine binary olarak kaydeden programdır.

- **CML (Ters Transfer)**

D	28	CML	(S)	(D)	Counter Transfer												
					Bit device				Word device								
					X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
(S)									☒	☒	☒	☒	☒	☒	☒	☒	☒
(D)												☒	☒	☒	☒	☒	☒

(S) de bulunan datanın içeriğini tersleyerek (1-0, 0-1) (D) ye transfer eder. Eğer

(S) içeriği K sabiti ise bu sabit direkt binary değere dönüşür.

Bu program ters faz çıkışı gerekli uygulamalarda kullanılır.

(S) de belirtilen parametre değerini (D) den itibaren (n) tane bölgeye transfer eder.



X0=ON olduğu zaman D10'dan itibaren 10 tane bölgeye (D10-D19) K10 bilgisini transfer eder.

- **XCH (Data Değişirme)**

D1 ve D2 datalarının içeriğini değiştirir. Puls kontağı ile kullanılması önerilir. Aksi takdirde durmadan değişme yapar.

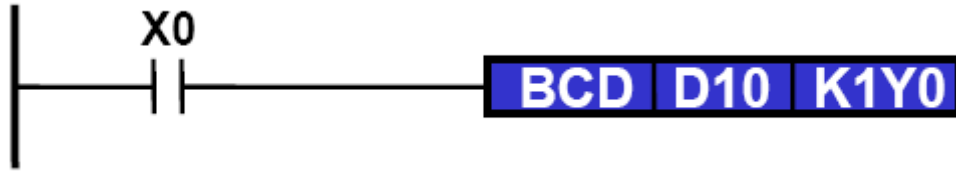


X0=ON olduğu zaman D10 daki bilgi ile D11 deki bilgi yer değiştirir.

- **BCD (Parametrede belirtilen BIN Datayı BCD'ye çevirir)**

(S) de tanımlanmış BIN datasını (0~9999) BCD ye çevirir ve sonucu (D) ye transfer eder.

Eğer (S) deki data 0 – 9999 arasında değilse uygulamada hata oluşur ve error bitleri (M1067 ve M1068) ON olur.

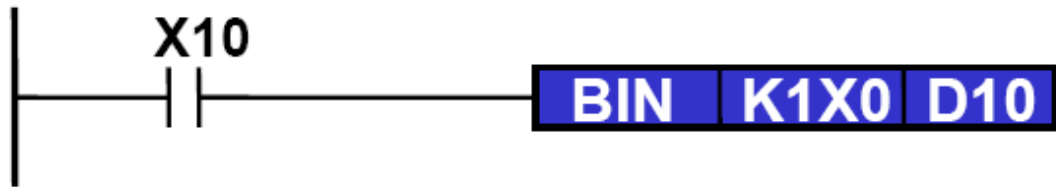


X0=ON olduğu zaman D10'un içindeki BIN değeri BCD'ye dönüşür ve sonucu K1Y0'a kaydeder.

- **BIN (Parametrede belirtilen BCD Datayı BIN'e çevirir)**

Ⓢ de tanımlanmış BCD datasını (0~9999) BIN ye çevirir ve sonucu Ⓣ ye transfer eder.

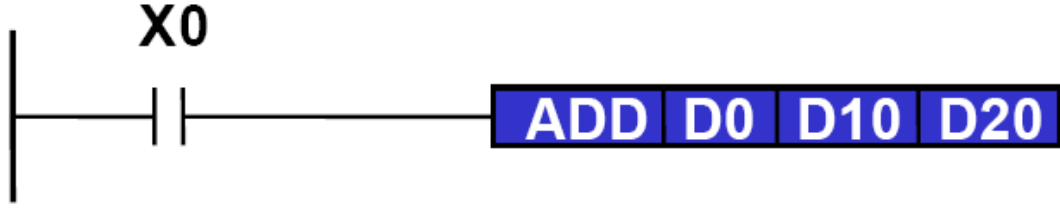
Eğer Ⓢ deki her bir digit 0 – 9 arasında değilse uygulamada hata oluşur ve error bitleri (M1067 ve M1068) ON olur.



X10=ON olduğu zaman X0-X3 BCD Datasını BIN Dataya çevirir ve sonucu D10'un içine kaydeder.

- **ADD (BIN Data toplama işlemini yerine getirir)**

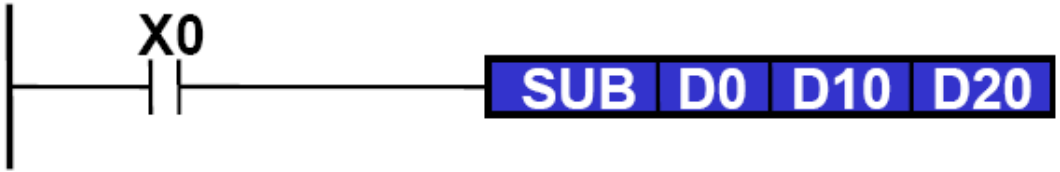
Ⓢ1 ve Ⓢ2 de belirtilen BIN dataları toplayarak sonucu Ⓣ de belirtilen parametreye kaydeder.



X0=ON olduğu zaman D0 ve D10'un içeriğini toplayarak D20'ye kaydeden programdır.

- **SUB (BIN Dataların Çıkartma işlemini yerine getirir)**

(S1) de belirtilen BIN datadan (S2) de belirtilen BIN data çıkartılır ve çıkartma sonucunu (D) de belirtilen parametreye kaydeder.



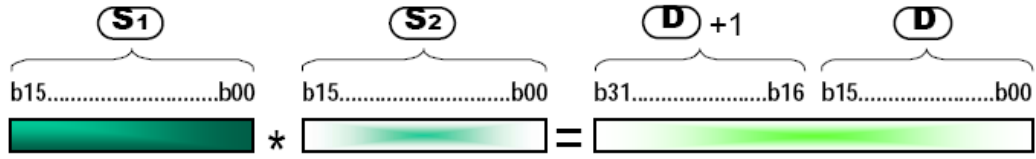
X0=ON olduğu zaman D0'ın içindeki değerden D10 içindeki değer çıkartılarak sonucu D20 içine kaydeden program.

- **MUL (BIN Datanın çarpma işlemini yerine getirir)**

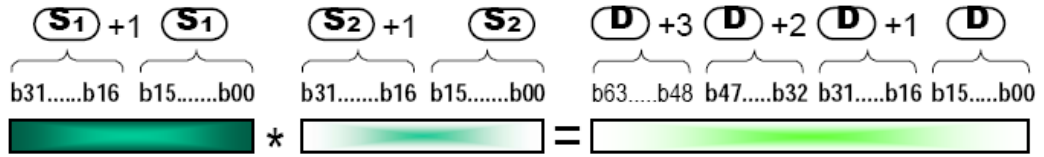
(S1) de belirtilen BIN Data ile (S2) de belirtilen BIN data çarpılır ve çarpım sonucu (D) de belirtilen parametreye kaydedilir.

(S1) ve (S2) datasının pozitif veya negatif kararı en üst bit olan b15'den (D) de ise b31'den yapılır.

16-bit uygulama



32-bit uygulama



Program Örneği (16-bit)



X10=ON olduğu zaman 5678 ve 1234 sayılarını çarparak sonucu BIN olarak D10 ve D11 içine kaydeden program. Yüksek 16-bit D11'de Düşük 16-bit D10'da kaydedilir.

Program Örneği (32-bit)



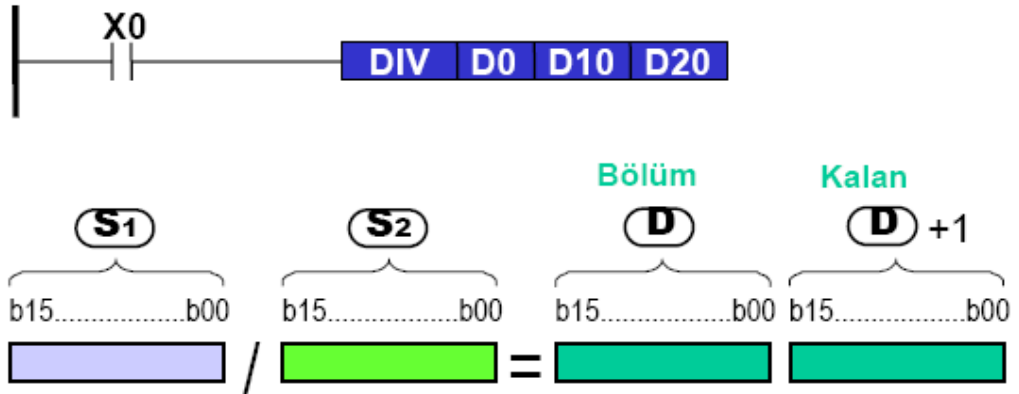
D0 ve D1'deki BIN Data ile D10 ve D11'deki BIN Datayı çarparak; 64 Bit Çarpma sonucunu BIN olarak D20-D23'e kaydeden programdır.

- **DIV (BIN Datanın Bölme işlemini yerine getirir)**

Uygulama sonucuna göre bölüm ile kalan word parametresi için 32-bit kullanılarak

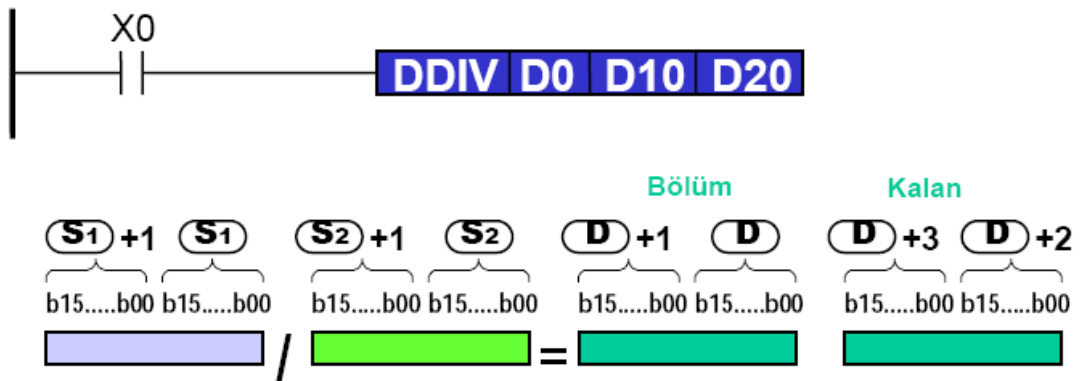
kaydedilir. Bit parametreleri içeren uygulamalarda ise sadece bölüm 16-bit olarak kaydedilir. $(S1)$ ve $(S2)$ datasının pozitif veya negatif kararı en üst bit olan b15'den (D) de ise b31'den yapılır.

16 bit uygulama



Bölüm : Düşük 16 Bite kaydedilir.
Kalan : Yüksek 16 Bite kaydedilir.
 (Sadece Word parametreleri durumunda kaydedilebilir)

32 bit uygulama



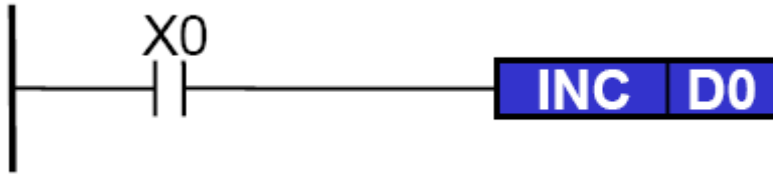
Bölüm : Düşük 32-Bite kaydedilir.
Kalan : Yüksek 32-Bite kaydedilir.
 (Sadece Word Parametreleri durumunda kaydedilir)

X0=ON olduğunda BIN olarak (S1) de belirlenen (D1,D0) değerini, BIN olarak (S2) de belirlenen (D11,D10) değerine böler; Bölüm (D) de belirtilen (D21,D20)'de kalan ise (D23,D22)'de kaydedilir.

- **INC (Belirtilen parametrenin değerini 1 arttırır)**

D 24	INC	(D)	Performs the addition of 1												
	Bit device				Word device										
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D		
(D)								☒	☒	☒	☒	☒	☒		

(D) de belirtilen parametrenin değerini 1 arttırır.

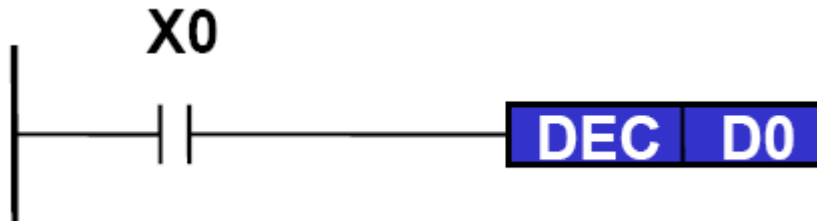


X0=ON olduğu zaman D0'ın içindeki değer 1 artar.

- **DEC (Belirtilen parametrenin değerini 1 azaltır)**

D 25	DEC	(D)	Performs the subtraction of 1												
	Bit device				Word device										
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D		
(D)								☒	☒	☒	☒	☒	☒		

(D) parametresinde belirtilen değerden 1 azaltma işlemini yerine getirir.



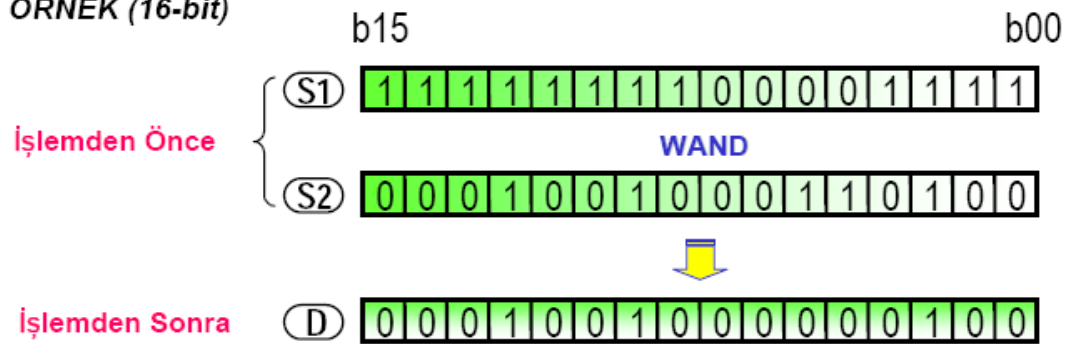
X0=ON olduğu zaman D0'ın içinde bulunan değer 1 azalır.

- **WAND (Parametrelerde belirtilen dataları AND işlemine tabi tutar)**

	D	26	WAND	S1	S2	D	Performs the logical product															
							Bit device				Word device											
				X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D						
								0	0	0	0	0	0	0	0	0						
								0	0	0	0	0	0	0	0	0						
											0	0	0	0	0	0						

(S1) ve (S2) de belirtilen 16-bit dataları bit başına lojik AND işlemine tabi tutarak sonucu (D) de belirtilen parametreye kaydeder.

ÖRNEK (16-bit)

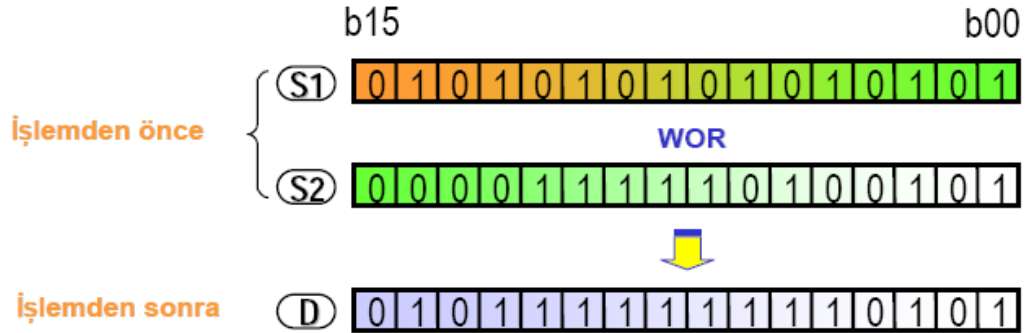


- **WOR (Parametrelerde belirtilen dataları OR işlemine tabi tutar)**

	D	27	WOR	S1	S2	D	Logic or (OR) operation															
							Bit device				Word device											
				X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D						
								0	0	0	0	0	0	0	0	0						
								0	0	0	0	0	0	0	0	0						
											0	0	0	0	0	0						

(S1) ve (S2) de belirtilen 16-bit dataları bit başına lojik OR işlemine tabi tutarak sonucu (D) de belirtilen parametreye kaydeder.

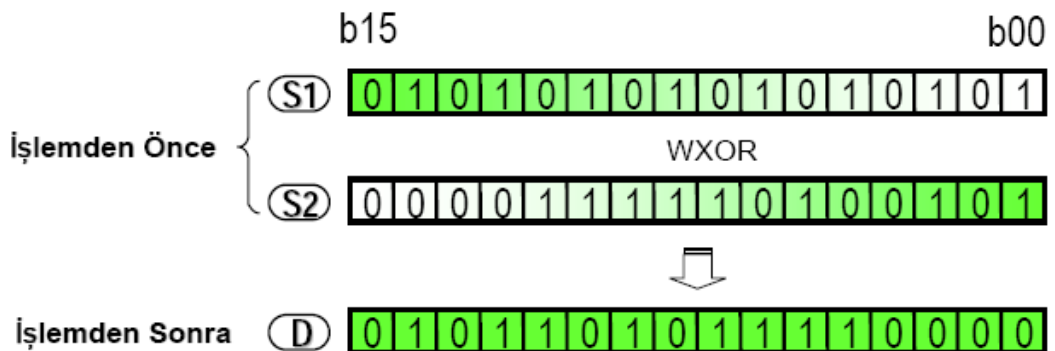
Örnek (16-bit)



- WXOR (Parametrelerde belirtilen dataları XOR işlemine tabi tutar)

D	28	WXOR	(S1)	(S2)	(D)	Performs the exclusive logical add											
						Bit device				Word device							
			X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D		
(S1)																	
(S2)																	
(D)																	

(S1) ve (S2) de belirtilen 16-bit dataları bit başına lojik XOR işlemine tabi tutarak sonucu (D) de belirtilen parametreye kaydeder.

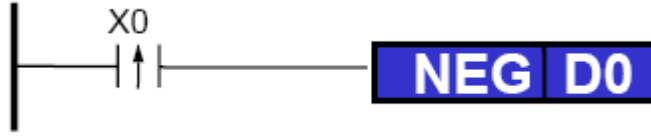


- **NEG (2'ye Tamamlama)**

D	29	NEG	(D)	Performs the addition of 1												
				Bit device					Word device							
				X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
				(D)							0	0	0	0	0	0

Komut işlemini yaptığında (D) de belirtilen parametrede 2'ye tamamlama işlemini yerine getirir.

Puls kontağı ile kullanılması önerilir. Aksi taktirde durmadan değişme yapar.



X0=ON olduğu zaman D0'ın içindeki değer terslenir. (0-1, 1-0 olur). 1 eklenerek sonuç D0 register'a kaydolur.

Bu komut Negatif BIN Değerini Pozitif olarak dikkate alır. (Mutlak değerini).

- **ROR (Sağa Döndürme)**

Koşul : $n \leq 16$ (16-Bit Komut) $n \leq 32$ (32-Bit Komut)

(D) ROR komutu Parametrede tanımlı bölge içindeki (32) 16-bit bilginin n - bit sağa döndürülmesini sağlar.

M,Y,S bitleri kullanılması gerekiyorsa sadece K4 (16-Bit) ve K8 (32-Bit) kullanılabilir. Örnek : K4M0, K8Y0.

Puls kontağı ile kullanılması önerilir. Aksi taktirde durmadan değişme yapar.



X0=ON olduğunda D10'un içindeki 16-bit data 4 bit sağa kayacak ve şekilde görüldüğü gibi b3 biti (CY) M1022'ye transfer olur.

- **ROL (Sola Kaydırma)**

Koşul : $n \leq 16$ (16-Bit Komut) $n \leq 32$ (32-Bit Komut)

(D) ROL komutu Parametrede tanımlı bölge içindeki (32) 16-bit bilginin n - bit sola döndürülmesini sağlar.

M,Y,S bitleri kullanılması gerekiyorsa sadece K4 (16-Bit) ve K8 (32-Bit) kullanılabilir. Örnek : K4M0, K8Y0.

Puls kontağı ile kullanılması önerilir. Aksi takdirde durmadan değişme yapar.



X0=ON olduğunda D10'un içindeki 16-bit data 4 bit sola kayacak ve şekilde görüldüğü gibi b12 biti (CY) M1022'ye transfer olur.

- **RCR (Carry Biti ile Sağa Döndürme)**

Koşul : $n \leq 16$ (16-Bit Komut) $n \leq 32$ (32-Bit Komut)

(D) RCR komutu Parametrede tanımlı bölge içindeki (32) 16-bit bilginin carry biti kullanılarak n - bit sağa döndürülmesini sağlar.

M,Y,S bitleri kullanılması gerekiyorsa sadece K4 (16-Bit) ve K8 (32-Bit) kullanılabilir. Örnek : K4M0, K8Y0.

Puls kontağı ile kullanılması önerilir. Aksi takdirde durmadan değişme yapar.



X0=ON olduğunda D10'un içindeki 16-bit data carry biti ile 4 bit sağa kayar ve şekilde görüldüğü gibi b3 biti (CY) M1022'ye transfer olur. Carry Bitinin orijinal içeriği ise b12'ye transfer olur.

- **RCL (Carry Biti ile Sola Döndürme)**

Koşul : $n \leq 16$ (16-Bit Komut) $n \leq 32$ (32-Bit Komut)

(D) RCL komutu Parametrede tanımlı bölge içindeki (32) 16-bit bilginin carry biti kullanılarak n - bit sola döndürülmesini sağlar.

M,Y,S bitleri kullanılması gerekiyorsa sadece K4 (16-Bit) ve K8 (32-Bit) kullanılabilir. Örnek : K4M0, K8Y0.

Puls kontağı ile kullanılması önerilir. Aksi taktirde durmadan değişme yapar.



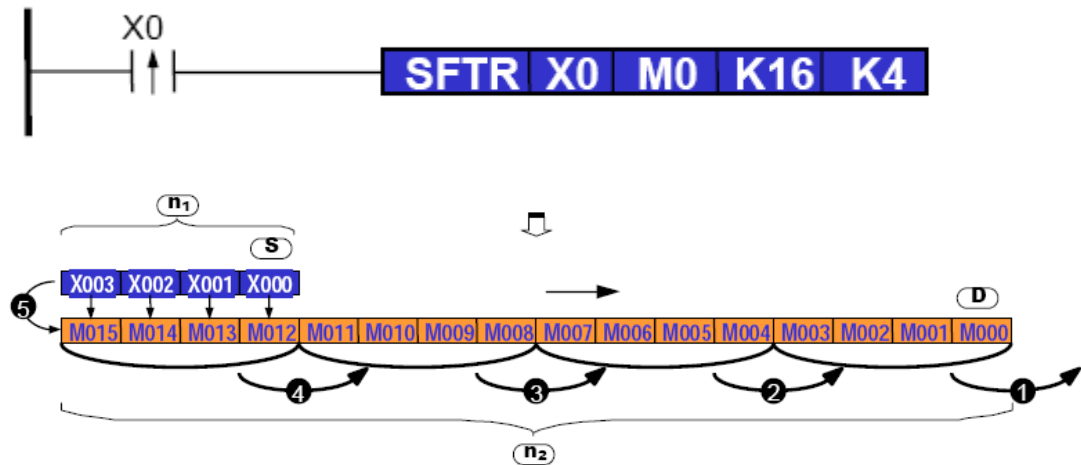
X0=ON olduğunda D10'un içindeki 16-bit data carry biti ile 4 bit sola kayacak ve şekilde görüldüğü gibi b12 biti (CY) M1022'ye transfer olur. Carry Bitinin orijinal içeriği ise b3'ye transfer olur.

- **SFTR (Parametrede belirtilen datayı sağa kaydırır.)**

34	SFTR	S	D	n1	n2	Shifts the data of device specified to the right											
						Bit device				Word device							
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D				
S	0	0	0	0													
D		0	0	0													
n1					0	0											
n2					0	0											

Koşul : $n_2 \leq n_1 \leq 512$ olmalı.

(S) den itibaren (n_2) tane biti (D) den itibaren (n_1) bit sağa kaydırır.

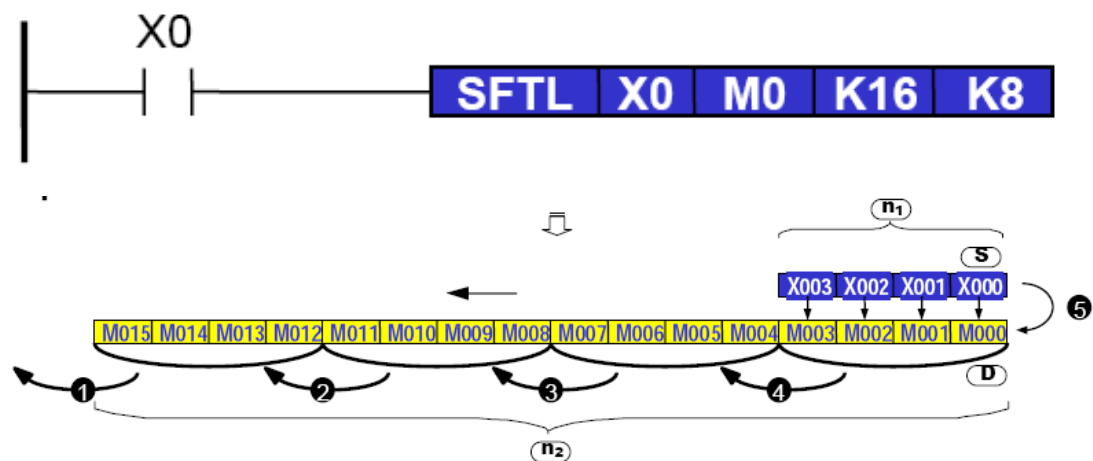


- SFTL (Parametrede belirtilen datayı sola kaydırır.)

	SFTL				Shifts the data of device specified to the left									
	S	D	n ₁	n ₂	Word device									
Bit device	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	
S	⊖	⊖	⊖	⊖										
D			⊖	⊖										
n ₁					⊖	⊖								
n ₂					⊖	⊖								

Koşul : $n_2 \leq n_1 \leq 512$ olmalı.

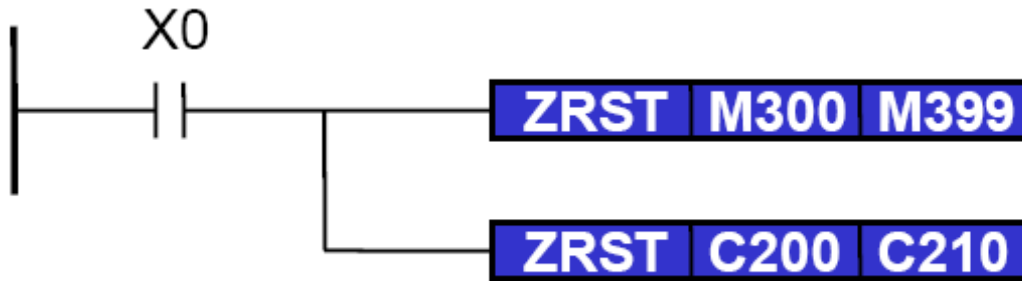
(S) den itibaren (n_2) tane biti (D) den itibaren (n_1) bit sola kaydırır.



- **ZRST (Belirtilen Parametre aralığını resetler.)**

40	ZRST	(D1)	(D2)	Resets a range of device specified.													
		Bit device				Word device											
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D			
(D1)		0	0	0								0	0	0			
(D2)		0	0	0								0	0	0			

Koşul : (D1) ve (D2) aynı kategoride olmalıdır.



X0=ON olduğu zaman M300-M399 arası OFF olur. C200-C210 arası sayıcı değerleri 0'a ayarlanır.

- **DECO (8 ⇒ 256 bit çözme)**

41	DECO	(S)	(D)	(n)	8 ⇒ 256 bit decode										
		Bit device				Word device									
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	
(S)		0	0	0	0	0	0					0	0	0	
(D)			0	0	0							0	0	0	
(n)						0	0								

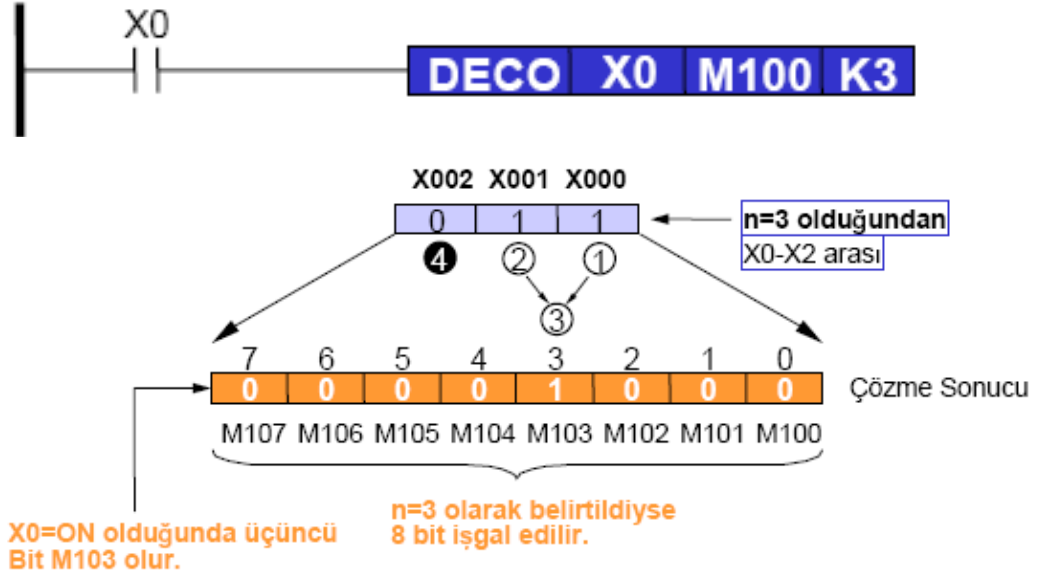
(S) de belirtilen değerden itibaren en düşük değerli (n) tane biti çözerek sonucu (D) den başlayarak 2ⁿ tane bite kaydeder.

“n”, 1 - 8 arası girilebilir.

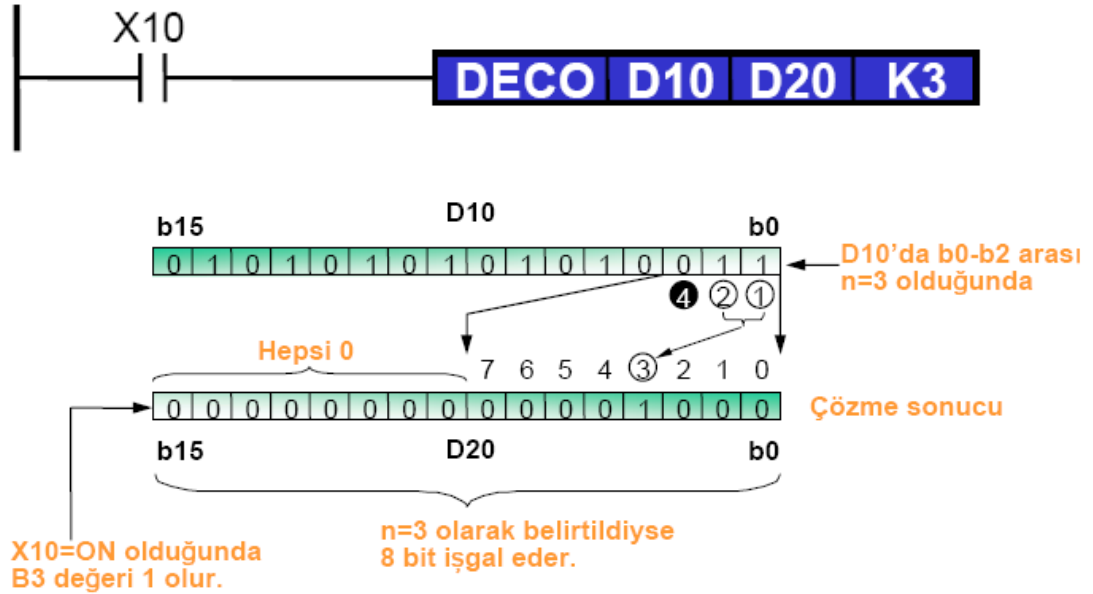
n = 0 olduğunda (D) içeriği değişmez.

Bit parametreleri 1 bit word parametreleri 16 bit olarak davranır.

Ⓓ bit parametresi olarak kullanıldığında $0 < n \leq 8$ olmalıdır.



Ⓓ word parametresi olarak kullanıldığında $0 < n \leq 4$ olmalıdır.



- ENCO (256 ⇒ 8 bit kodlama)

42	ENCO	(S)	(D)	(n)	256 ⇒ 8 bit encode									
		Bit device				Word device								
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
(S)		0	0	0	0							0	0	0
(D)												0	0	0
(n)						0	0							

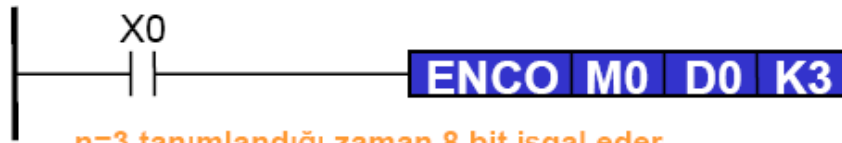
(S) den itibaren 2^n biti (D) ye kaydeder.

n ”, 1 - 8 arası girilebilir.

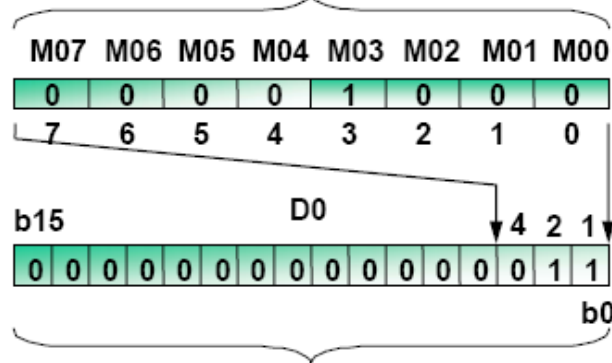
$n = 0$ olduğunda (D) içeriği değişmez.

İki veya daha fazla bit varsa sondaki bitin pozisyonu dikkate alınır.

(S) bit parametresi olarak kullanıldığında $0 < n \leq 8$ olmalıdır.



$n=3$ tanımlandığı zaman 8 bit işgal eder.

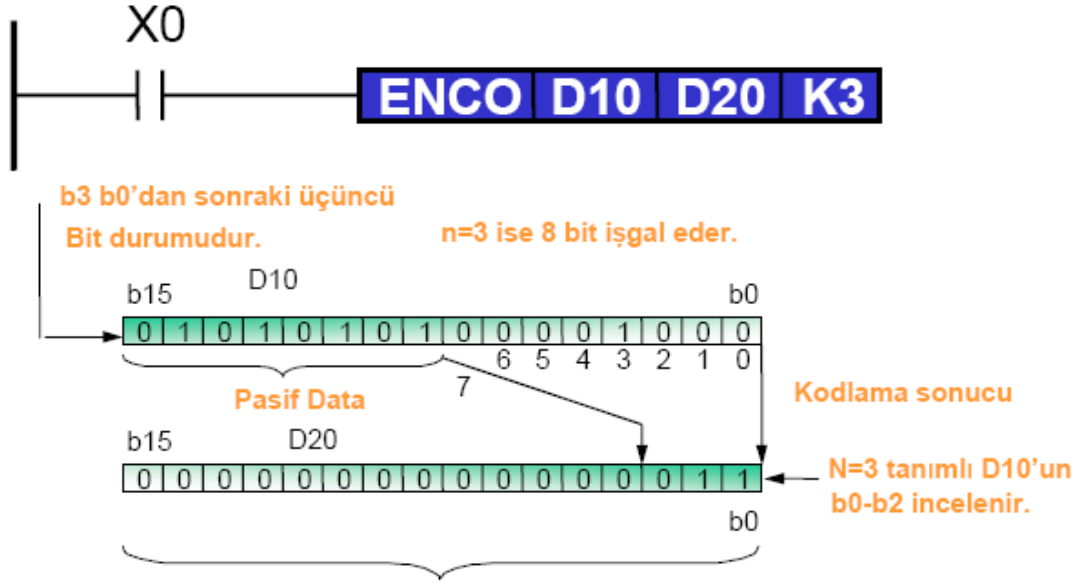


Kodlama sonuçları

M0'dan sonraki ilk aktif biti BIN olarak kaydeder.

(S) word parametresi olarak kullanıldığında $0 < n \leq 4$ olmalıdır.

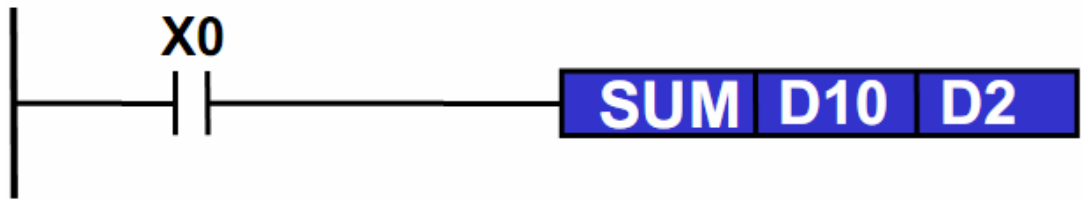
(S) word parametresi olarak kullanıldığında $0 < n \leq 4$ olmalıdır.



- SUM (ON olan Bitlerin Toplamı)

API	Mnemonic	Operands	Function													
43	D SUM P	(S) (D)	Sum of ON Bits													
Type	Bit Devices				Word devices											
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	E	F	
S					*	*	*	*	*	*	*	*	*	*	*	
D							*	*	*	*	*	*	*	*	*	

(S) de belirtilmiş data içindeki 1 lerin sayısını (D) ye transfer eder.



X0=ON olduğunda D10'un içindeki 1'lerin sayısını D2 içine kaydeden programdır.

Eğer D10'da hiç 1 yoksa Zero Flag (M1020) = ON olur.

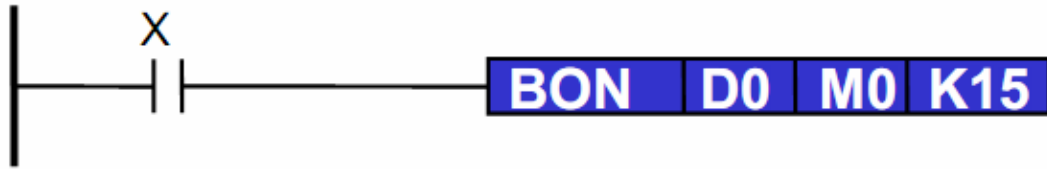
DSUM veya DSUMP 32-bit komutları kullanıldığında zaman D0 ve D1 içindeki 1 bitlerinin sayısını D2'ye kaydeder. D3'ün değeri 0 olur. (D3=0).

- **BON (ON olan Biti Belirleme)**

API	Mnemonic			Operands							Function					
44	D	BON	P	(S)	(D)	(n)						Check Specified Bit Status				
OP	Type				Word devices											
	Bit Devices				Word devices											
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	E	F	
S					*	*	*	*	*	*	*	*	*	*	*	
D		*	*	*							*	*	*	*	*	
n					*	*	*	*	*	*	*	*	*	*	*	

Koşul : 16-Bit Komutlarda n=0-15, 32-Bit Komutlarda n=0-31 olabilir.

(S) deki bölgenin n numaralı biti 1 ise (D) de tanımlanan bit = ON olur.



X0=ON olduğunda D0'nın 15 nolu Biti = 1 ise M0 = ON olur.

X0=OFF olduğu zaman M0 bir önceki ON/OFF durumunda kalır.

- **MEAN (Ortalama değer)**

API	Mnemonic			Operands							Function					
45	D	MEAN	P	(S)	(D)	(n)						Mean Value				
OP	Type				Word devices											
	Bit Devices				Word devices											
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D	E	F	
S							*	*	*	*	*	*	*	*	*	
D							*	*	*	*	*	*	*	*	*	
n					*	*	*	*	*	*	*	*	*	*	*	

(S) den itibaren n tane bölgenin içeriklerini toplayıp sonucu gene n ye bölerek bölge değerlerinin ortalamasını (D) ye kaydeder.

DHSCS ve DHSCR Komutlarında 16-Bit uygulama yoktur. (API 53 ve API 54).

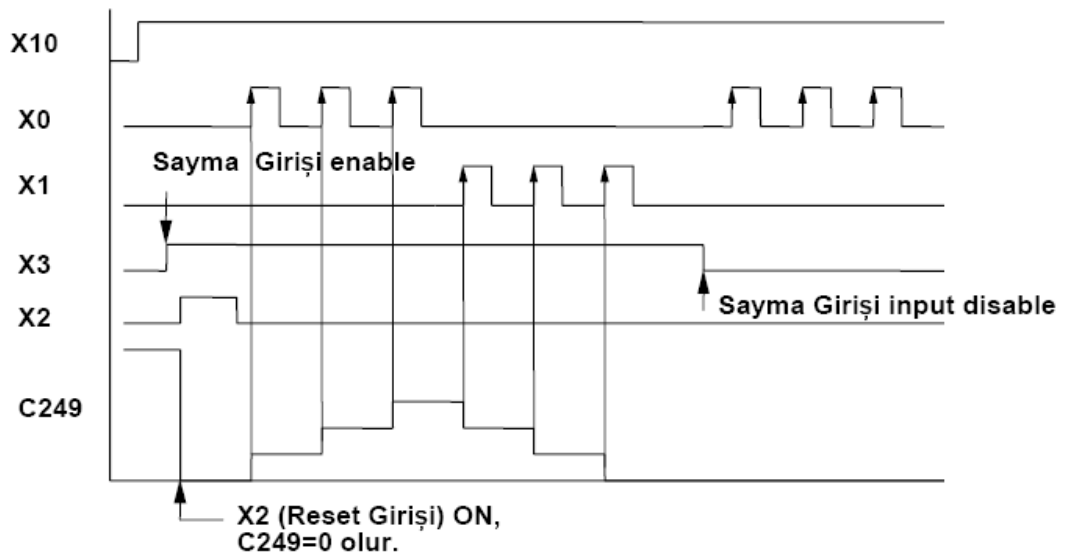
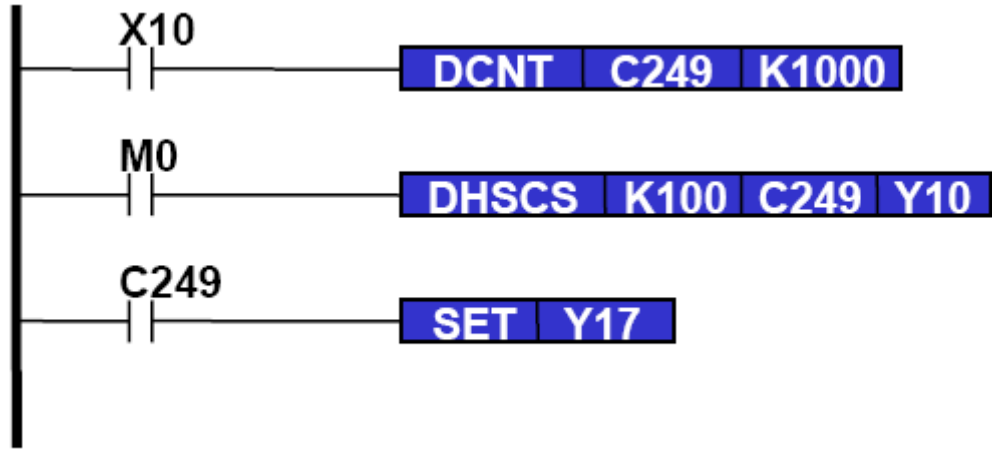
Uygulamadaki I/O ihtiyaçlarına göre ayarlanabilen (X0-X3) girişlere uygun yüksek hızlı sayıcı belirlenir.

Saymanın amacı (S_2) değeri (S_1) e ulaştığı zaman özel uygulama yapmaktır.

Counter değeri kullanıcının belirlediği değere ulaştınca interrupt routine atlar.

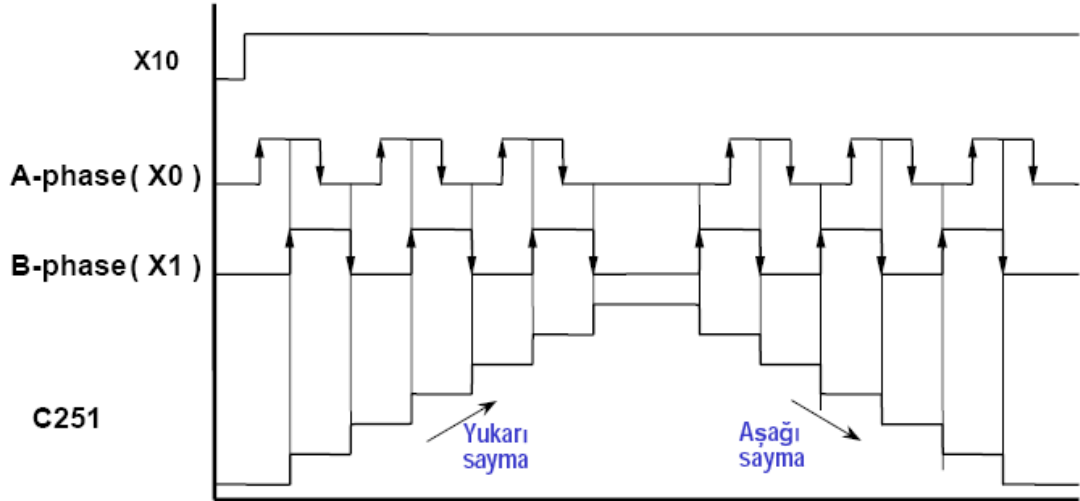
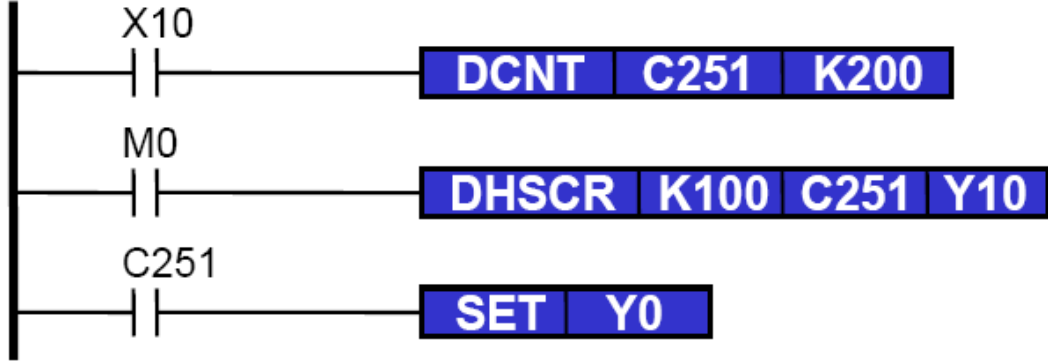
Interrupt routine içinde ani uygulamalar için (D) nin özel röle kullanılması önerilir.

Sayıcı mevcut değerini DHSCS ve DHSCR komutlarında belirtilen 4 set değerine kadar karşılaştırabilir. Eğer (D) Y ile kullanılacaksa Y00-Y17 arası bir değer seçilmelidir.



M0=ON olduğunda C249 Sayıcı Değeri 99'dan 100'e veya 101'den 100'e geçtiğinde Y10=ON olur.

C249 sayıcı Değeri 999'dan 1000'e veya 1001'den 1000'e geçtiğinde Y17=ON olur. Fakat program taramadan dolayı çıkışta gecikme olur.



M0=ON olduğunda C251 Hızlı Sayıcı Değeri 99'dan 100'e veya 101'den 100'e geçtiğinde Y10=OFF olur.

C251 sayıcı değeri 199'dan 200'e veya 201'den 200'e geçtiğinde C251 aktif olur aynı anda Y0 çıkışı ON yapar. Program taramadan dolayı çıkışta bir gecikme olur.

- **PLSY (Pulse Çıkışı)**

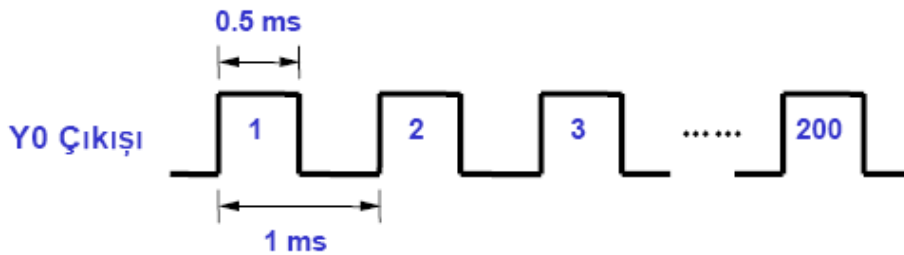
D 57	PLSY	S1	S2	D	Pulse output									
		Bit device				Word device								
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
S1						0	0	0	0	0	0	0	0	0
S2						0	0	0	0	0	0	0	0	0
D		Y00												

Belirlenmiş Frekans ve Pulse sayısı komutlarını meydana getirir.

S1 10 – 20 KHz Belirlenmiş Frekans.

S2 Belirlenmiş Pulse'ler. 16-bit: 1 - 32767, 32-bit: 214748364

D Belirlenmiş Y çıkışı olup Sadece Y0 çıkışı kullanılabilir. (Transistor çıkışlı modül kullanılmalıdır.)



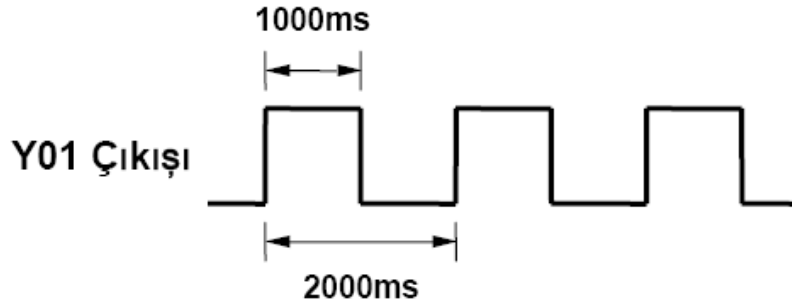
- **PWM (Pulse genişlik modülasyon çıkışı)**

58	PWM	S1	S2	D	Pulse width modulation output									
		Bit device				Word device								
		X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
S1						0	0	0	0	0	0	0	0	0
S2						0	0	0	0	0	0	0	0	0
D		Y01												

S1 0 – 32767 ms arası pulse genişlik zamanı belirlenir.

Ⓐ 1-32767 ms cycle zamanı belirlenir.

Ⓑ Çıkış pulsi kullanılacak Y çıkışı tanımlanır. Sadece Y01 çıkışı kullanılabilir.



- **PLSR (Hızlanma/Yavaşlama Komutları ile Puls Dalga Çıkışı)**

PLSR komutu sabit durumdan hedef hıza belirli bir zamanda ulaşılması gerektiği zamanlarda kullanılır. Hedef hıza ulaşıldığında motor daha hızlı hareket eder. Pulse dalga çıkışı hedef mesafeye ulaştığında durur.



S1 = Max. Hız (Hz). 10-10000 Hz arası ayarlanır. Max. hız 10 ve katları şeklinde olmalıdır. Eğer olmazsa parametre kullanılamaz. Hızlanma/Yavaşlama süreleri 1/10 max. hız adımları ile çalışarak step motor gürültüsünü engeller.

S2 = Pals dalga çıkışı Adedi. Ayarlar \rightarrow 110 - 32767 (16-bit komut) , 110 – 2147483647 (32-bit komut). Eğer bu ayar 110 dan aşağı olursa normal çıkış vermez. 32- Bit komut kullanıldığı zaman çıkış pulse dalga adedi D1 ve D0 'I içerir.

S3 = Hızlanma/Yavaşlama Zamanı (ms). Ayar 5000 ms altında olmalı ve aşağıdaki 3 şartı karşılamalıdır. Hızlanma ve yavaşlama zamanı aynı olup birbirinden bağımsız ayarlanamaz.

1-) Hızlanma/ Yavaşlama Zamanı Max. Scan Time (D1012) * 10'un üzerinde olmalıdır. Eğer bu değerın altında olursa Hızlanma/Yavaşlama doğrusu eğimi yanlış olur.

2-) Hızlanma/Yavaşlama Zamanı Minimum ayarı formülü $\rightarrow S3 \geq 90.000 / S1 * 1.22$

3-) Hızlanma/Yavaşlama Zamanı Maksimum Ayarı Formülü $\rightarrow S3 \leq S2/S1 * 818$

Sonuç olarak $90.000 / S1 * 1.22 \leq S3 \leq S2/S1 * 818$ olmalıdır.

Eğer Hızlanma/Yavaşlama süreleri ile yukarıdaki formüller sağlanamazsa Max. Hız düşürülerek formül sağlanır.

Sadece Y0 ve Y1 çıkışları puls çıkışı olarak kullanılabilir. Transistor çıkışlı PLC kullanılmalıdır.

X10 = OFF olduğunda çıkış kesilecek ve tekrar ON olduğunda Puls Dalga çıkışları 0'dan saymaya başlar.

S2'de ayarlanan puls sayısı iletildiğinde Y0 çıkışı tamamlanacak ve M1029=ON olacak. Aynı işlem Y1 çıkışı tamamlandığında M1030= ON olur.

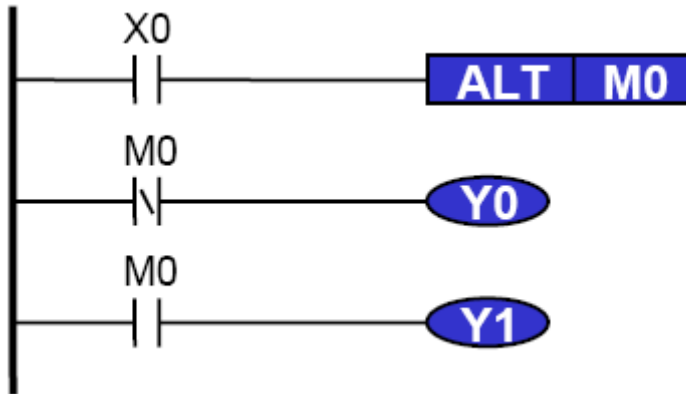
Komutların Kullanılma sayısı : PLSY (DPLSY), PWM ve PLSR (DPLSR) komutları her bir hızlı çıkış için 1 kere kullanılabilir.

- **IST (Manuel / Otomatik)**

API	Mnemonic	Operands	Function													
60	IST	(S) (D1) (D2)	Manual/Auto Control													
OP	Type	Bit Devices	Word devices													
		X Y M S K H KnX KnY KnM KnS T C D E														
	S	*	*	*												
	D ₁			*												
	D ₂			*												

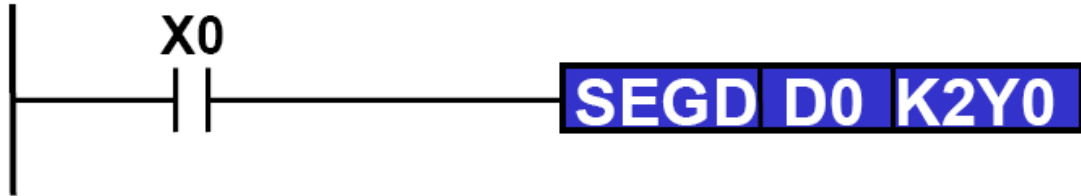
- **ALT (Karşılıklı ON/OFF Komutu)**

API	Mnemonic	Operands	Function													
66	ALT	(D)	ON/OFF Alternate command													
OP	Type	Bit device	Word device													
		X Y M S K H KnX KnY KnM KnS T C D														
	(D)															



X0 ilk kez aktif edildiği zaman M0=ON, Y1=ON; X0 ikinci kez aktif edildiği zaman M0=OFF, Y0=ON, =ON, Y1=OFF olur.

- **SEGD (7-Segment Display Kod Çözme)**



X0=ON olduğu zaman D0 bölgesinin içindeki en düşük 4 bitin değerini 7-Segment display tarafından okunabilecek hale getirerek K2Y0'a kaydeden programdır.

- **RS (Belirlenmiş Haberleşme Alanı içindeki data'ya göre data haberleşmesini yerine getirir.)**

	Bit device				Word device								
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
S													
m													
D													
n													

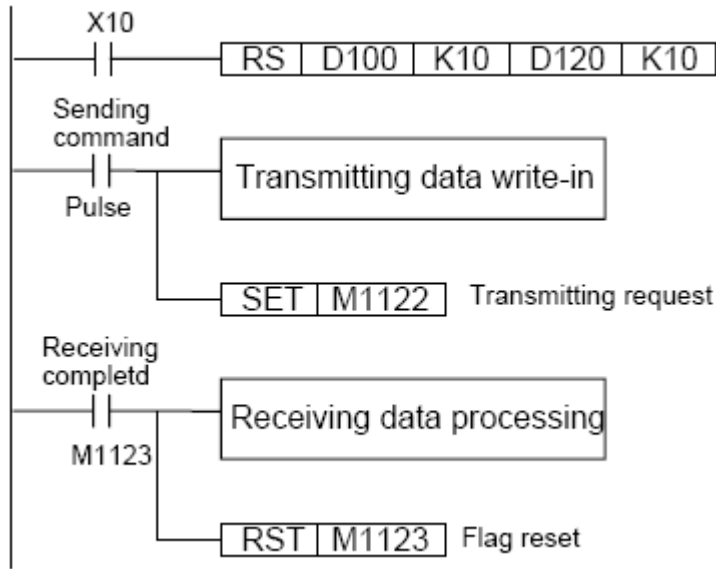
RS-485 seri haberleşme arabirimi içeren MPU (Main Processing Unit) için uygun bir komuttur. Kullanıcı datayı **S**'e kaydetmeden önce D1120 RS-485 Haberleşme Protokolüne kaydetmelidir.

RS Komutu kullanılacağı zaman; özel yardımcı röleler M1120-M1160 ve özel data register'ları D1120-D1131'e başvurulmalıdır.

		Content	0	1
b0		data length	7	8
b1 b2		parity	b2, b1	00 : None 01 : Odd 11 : Even
b3		stop bits	1 bit	2 bit
b4	0011 :	300		
b5	0100 :	600		
b6	0101 :	1200		
b7	0110 :	2400		
	0111 :	4800		
	1000 :	9600		
	1001 :	19200		
b8		STX setting	None	D1124
b9		ETX1 setting	None	D1125
b10		ETX2 setting	None	D1126
b15~b11 : don't care				

Kullanıcı Haberleşme komutları içine başlangıç ve bitiş bitlerini eklemelidir. DVP M1126 ve M1130 ile 2 kullanıcı tanımlama ünitesi seçeneği sağlar. Kullanıcı D1124,D1125 ve D1126 ile başlangıç ve bitiş bitlerini ayarlayabilir.

		M1130	
		0	1
M1126	0	D1124 : User defined D1125 : User defined D1126 : User defined	D1124 : H 0002 D1125 : H 0003 D1126 : H 0000 (No defined)
	1	D1124 : User defined D1125 : User defined D1126 : User defined	D1124 : H 003A (':') D1125 : H 000D (CR) D1126 : H 000A (LF)



- **ASCII (Belirtilmiş Hekzadesimal değeri ASCII koda çevirir.)**

	Converts HEX into ASCII												
	Bit device				Word device								
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
(S)					□	□	□	□	□	□	□	□	□
(D)								□	□	□	□	□	□
(n)					□	□							



(S) de belirtilmiş Hekzadesimal değeri ASCII koda çevirir ve sonuçları (D) den itibaren (n) tane noktaya kaydeder.

- **HEX (Belirtilmiş ASCII Kodu Hekzadesimal değere çevirir.)**

	Converts ASCII into HEX												
	Bit device				Word device								
	X	Y	M	S	K	H	KnX	KnY	KnM	KnS	T	C	D
(S)					⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗	⊗
(D)								⊗	⊗	⊗	⊗	⊗	⊗
(n)					⊗	⊗							

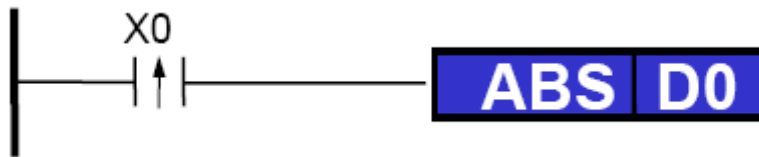


(S) de ASCII kod olarak tanımlanmış alt ve üst düzeydeki 8'er biti hegzadesimal'e çevirerek sonuçları (D) den itibaren (n) tane noktaya kaydeder.

- **ABS (Mutlak Değer)**

(D) de belirtilen parametrenin mutlak değerini alır.

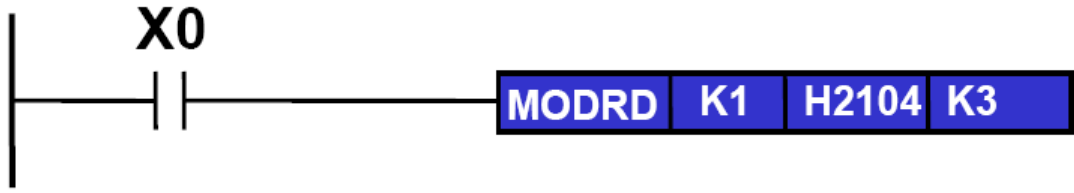
Puls kontağı ile kullanılması önerilir. Aksi taktirde durmadan değişme yapar.



X0=ON olduğu zaman D0'ın içindeki değerın mutlak değerini alır.

- **MODRD (MODBUS Data okuma)**

MODRD MODBUS ASCII haberleşme komutudur. (Version 3.3 ve sonrası RTU Mod içerir ve M1143 tarafından ayarlanır.) Delta VFD Serisi Sürücüler MODBUS Haberleşme içerir.



K1 = Haberleşme Adresi (K0-K254).

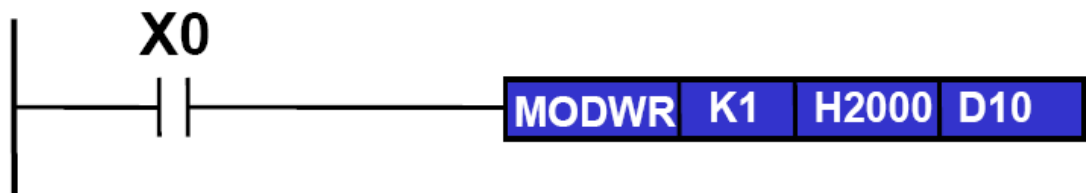
H2104 = Datası Okunacak Adres. Eğer adres ayarı yanlışsa kullanıcı bir hata kodu ile uyarılır. Bu kod D1130 da saklanır. Ayrıca M1141=ON olur. Örneğin 4000H VFD-S Serisi Driverlar için yanlış adrestir. Bu durumda M1141=ON ve D1130=2 olur.

K3 = Data Uzunluğu. $N \leq 6$ olmalıdır.

Geri besleme Datalar D1070-1085 de saklanır. Daha sonra PLC Dataları kontrol eder. Eğer hata varsa M1140=ON olur.

- **MODWR (MODBUS Data Yazma)**

MODWR MODBUS ASCII haberleşme komutudur. (Version 3.3 ve sonrası RTU Mod içerir ve M1143 tarafından ayarlanır.) Delta VFD Serisi Sürücüler MODBUS Haberleşme içerir.



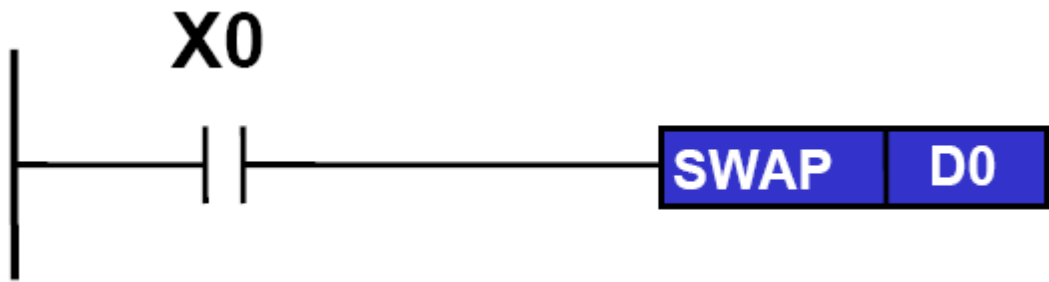
K1 = Haberleşme Adresi (K0-K254).

H2000 = Data Yazılacak Adres. Eğer adres ayarı yanlışsa kullanıcı bir hata kodu ile uyarılır. Bu kod D1130 da saklanır. Ayrıca M1141=ON olur.

D10 = Sürücü içine yazılacak Data.

Geri besleme Datalar D1070-1076 de saklanır. Daha sonra PLC Dataları kontrol eder. Eğer hata varsa M1140=ON olur.

- **SWAP (Üst ve Alt 8 bit yer değiştirir.)**

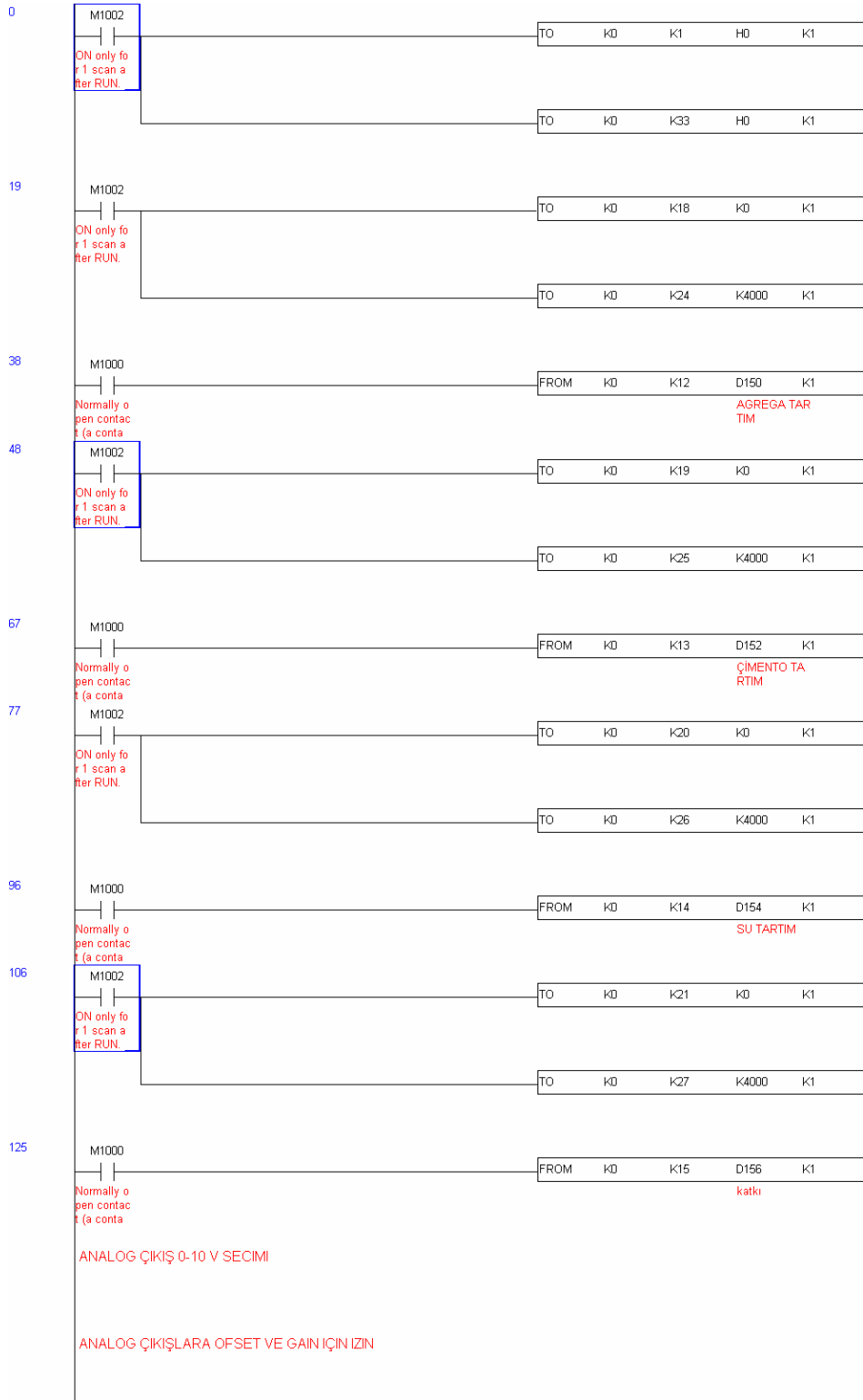


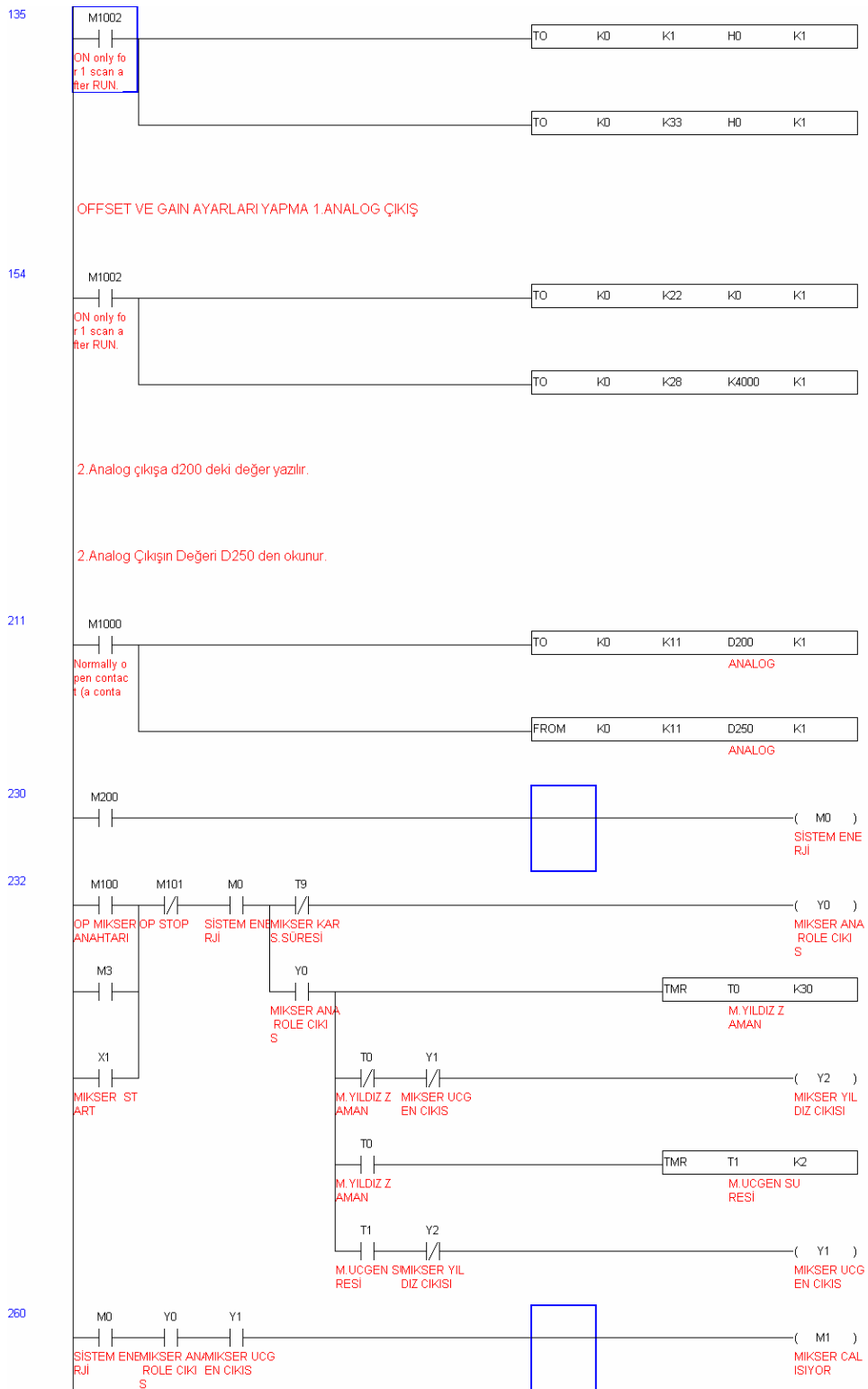
1. X0 = ON olduğu zaman D0 Bölgesinin Üst 8-biti ile Alt 8 biti yer değiştirir.
2. 16-Bit Komut kullanıldığında Kanalın ilk 8 biti ile son 8 biti kullanılır.
3. 32-Bit Komut kullanıldığında her iki kanalında ilk 8 biti ile son 8 biti yer değiştirir.
4. Puls komutu ile kullanılması uygundur. (SWAPP, DSWAPP) [14].

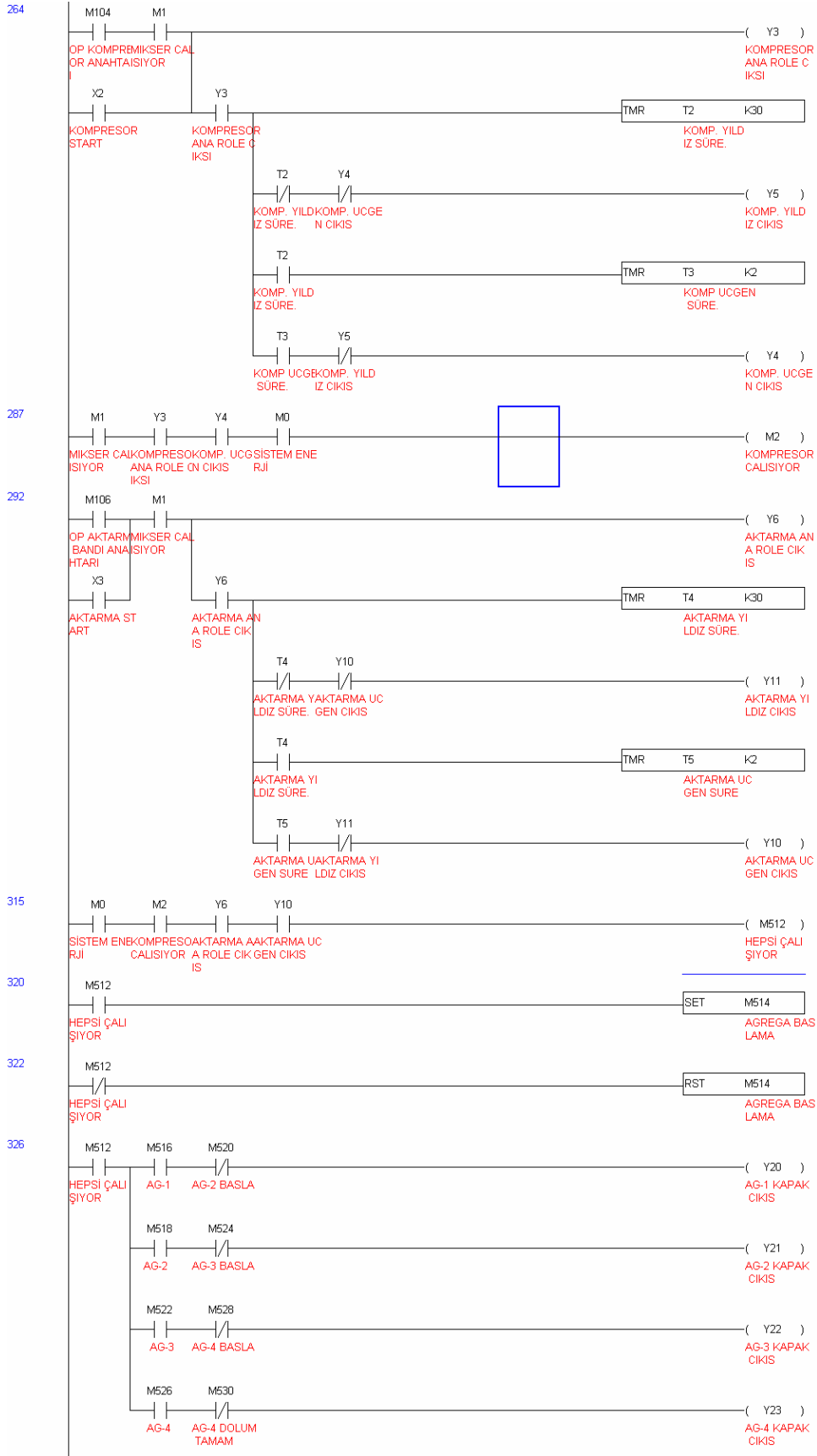
EK-2

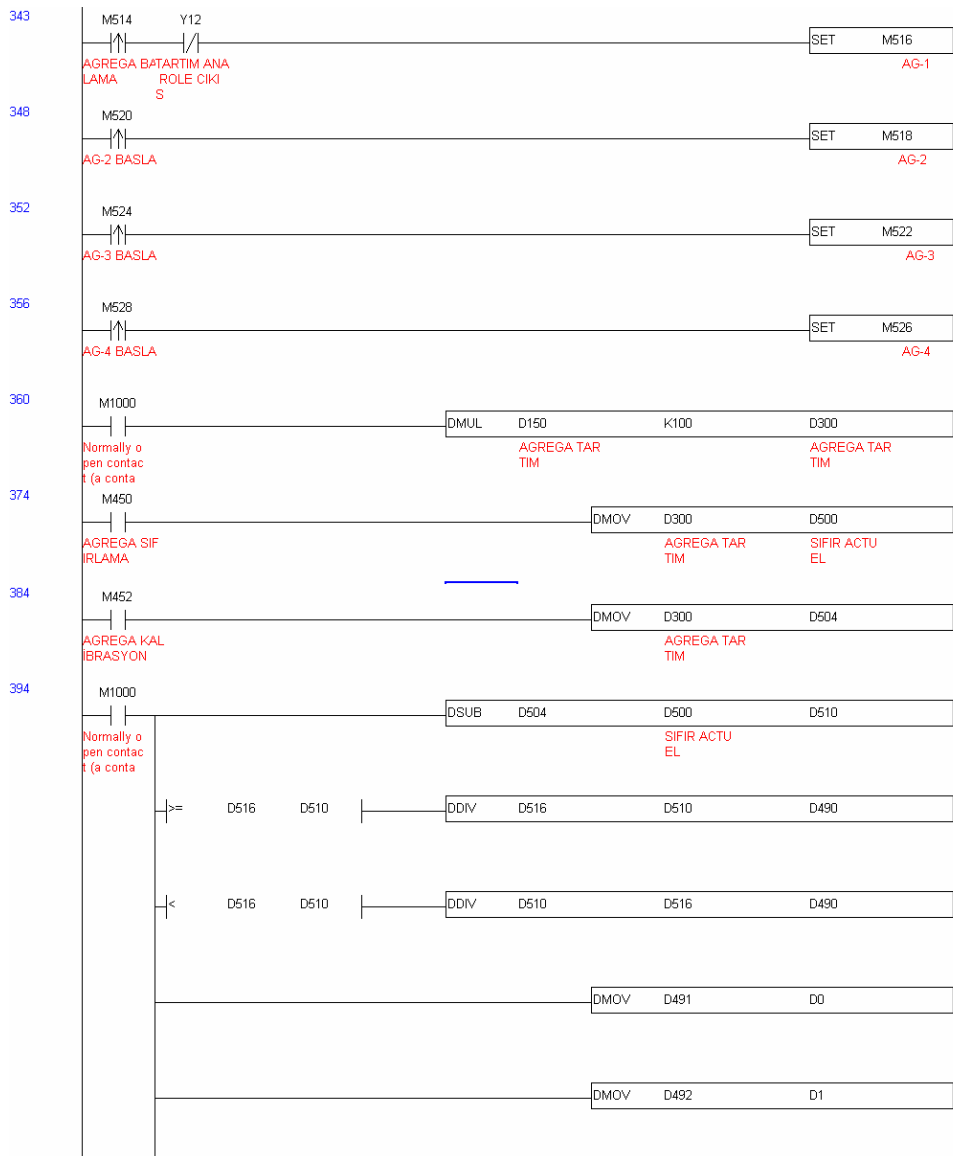
EK(2)

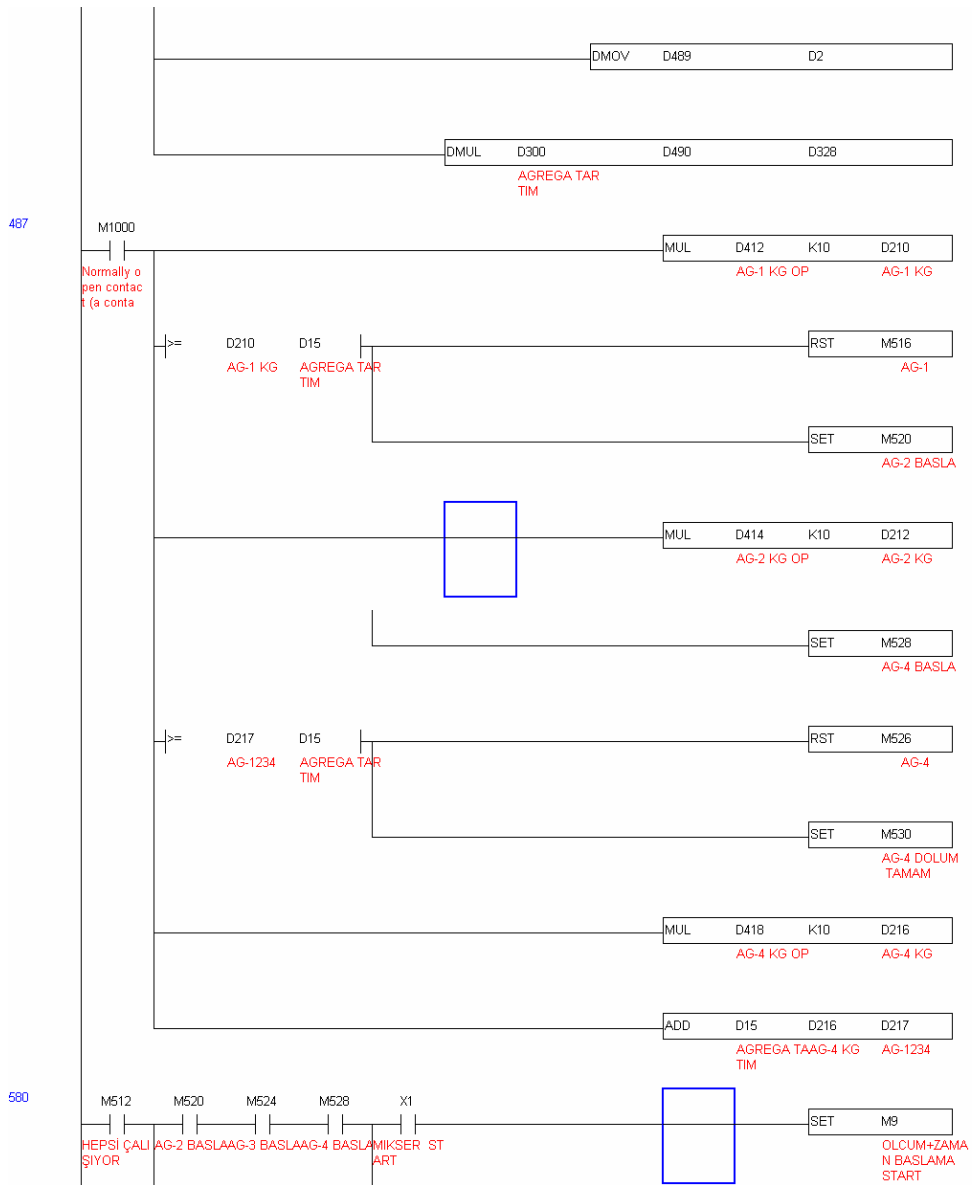
1. Ladder Yöntemiyle Yazılan PLC Programı

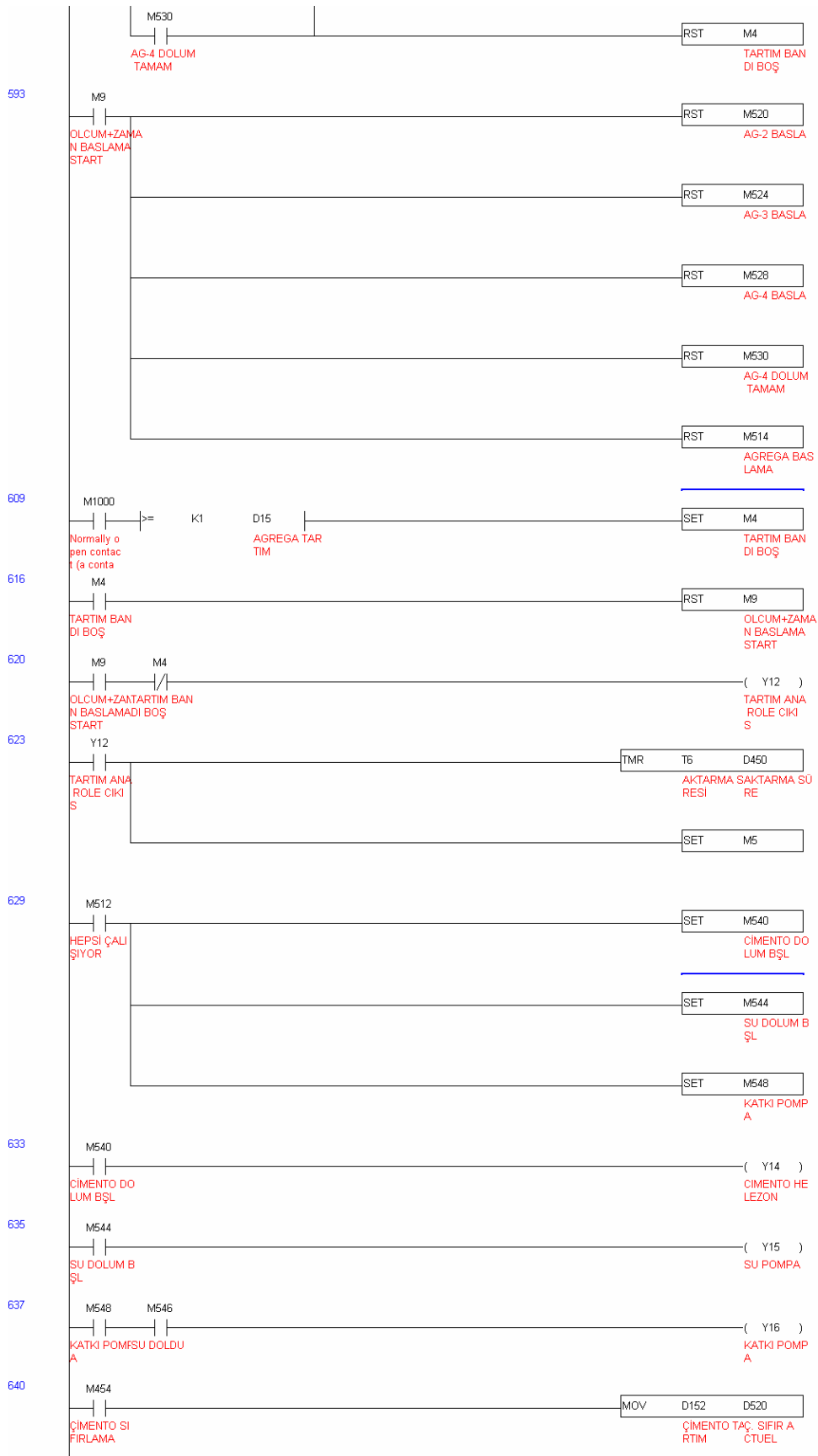


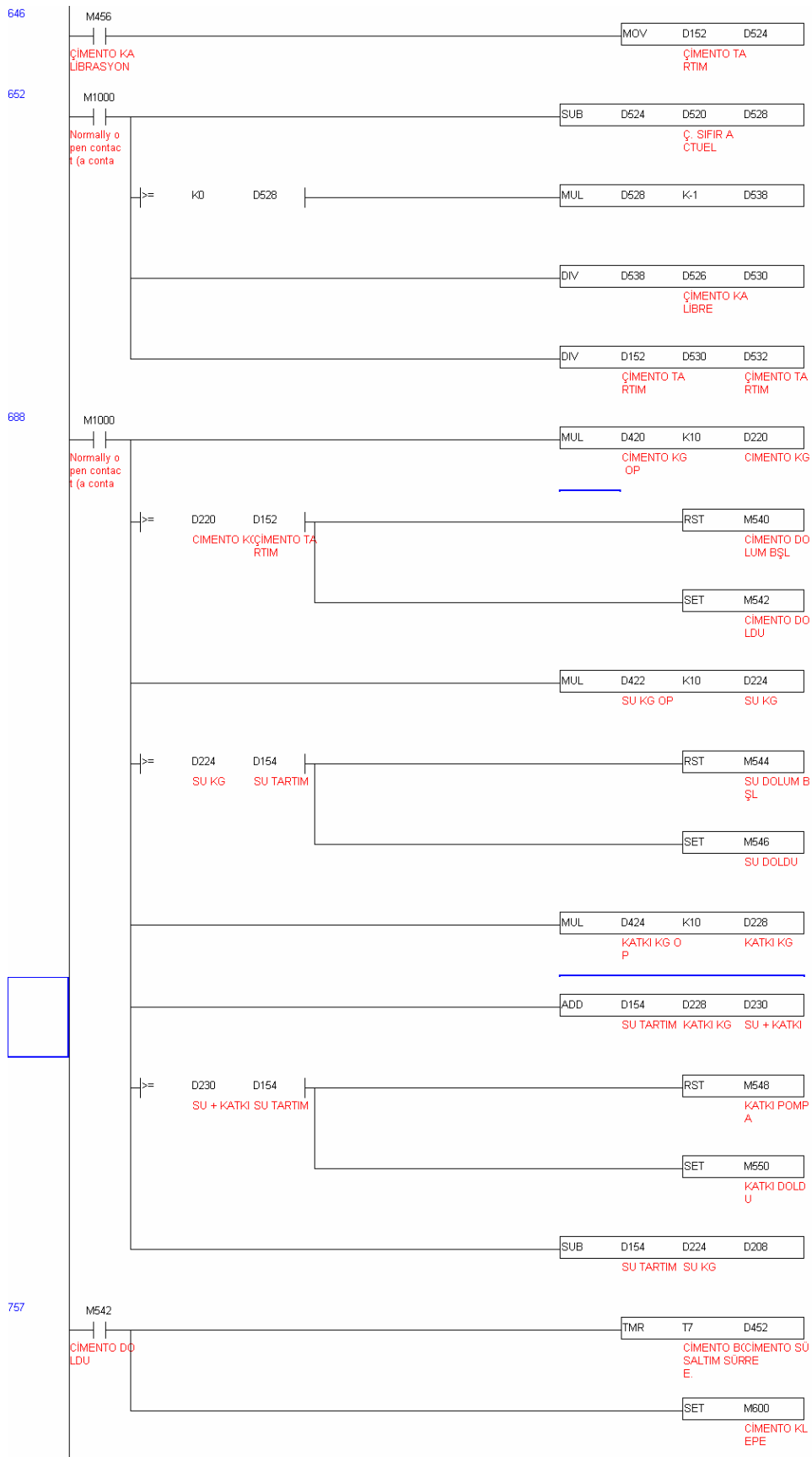


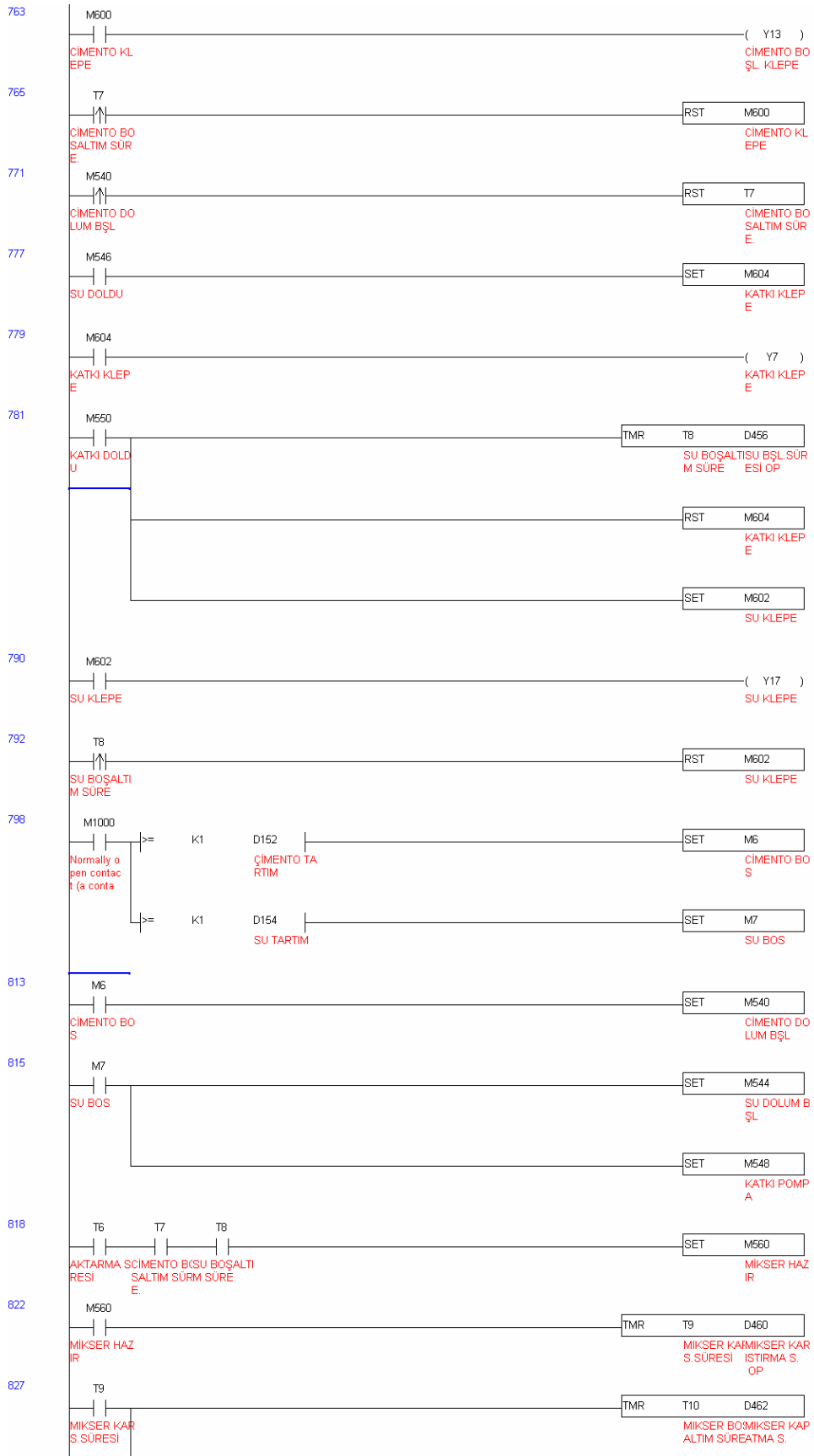


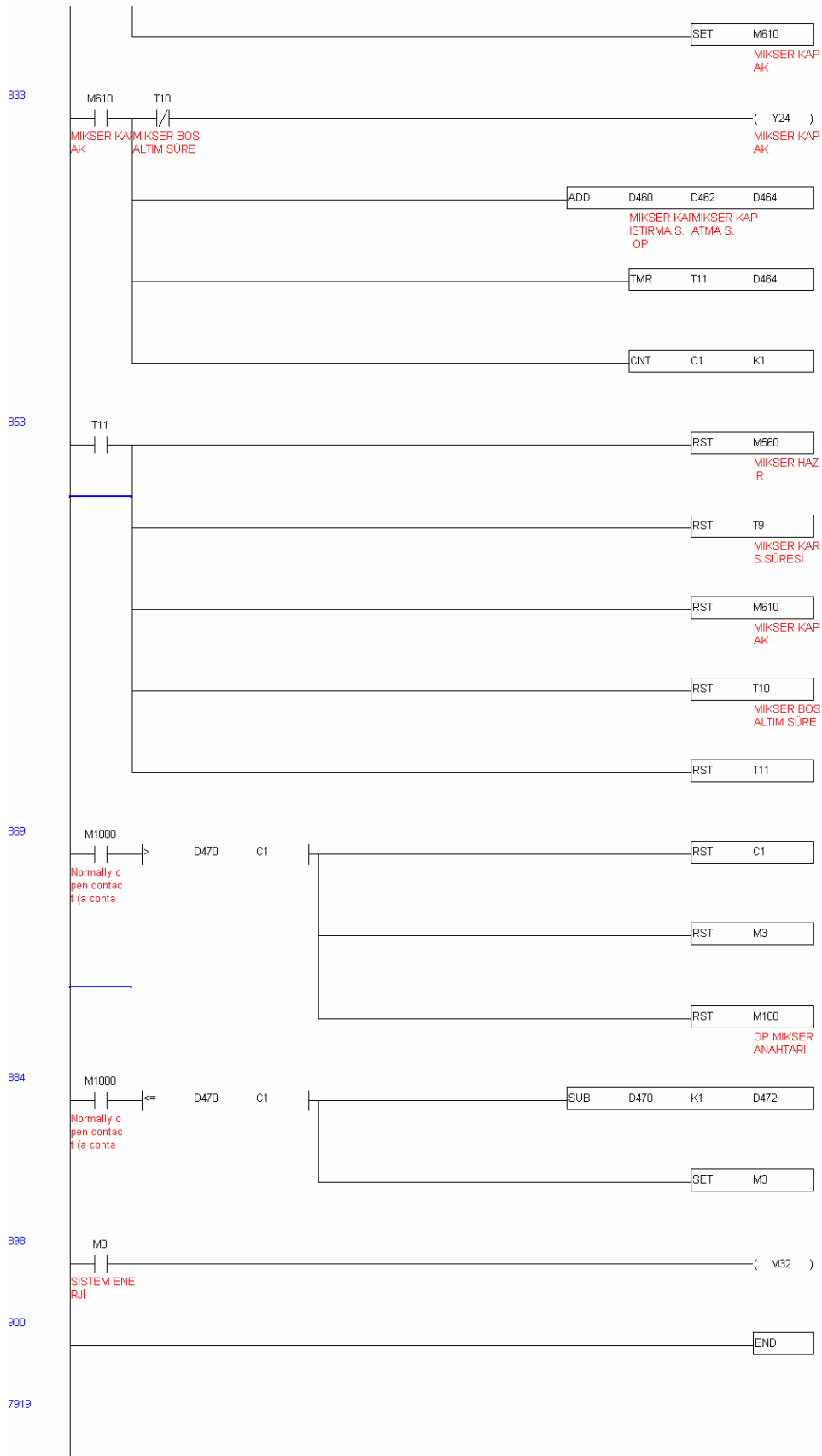












ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : BEKAR, Gizem
Uyruğu : T.C.
Doğum tarihi ve yeri : 27.11.1983 Ankara
Medeni hali : Evli
Telefon : 0535 232 25 81
e-mail : gizemcaglar@hotmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Lisans	Gazi Üniversitesi /Elektronik Öğretmenliği	2006
Lise	Türk Telekom Anadolu Meslek Lisesi	2001

İş Deneyimi

Yıl	Yer	Görev
Temmuz-Aralık 2007	DK Mühendislik	Kalite Kontrol Sorumlusu
2007-2009	Ortana Sistem Kurulum	Ar-Ge PCB Tasarımcısı
2009-.....	Sinop Üniv. Ayancık M.Y.O	Öğretim Görevlisi

Yabancı Dil

İngilizce

Hobiler

Yüzmek, Tiyatro izlemek