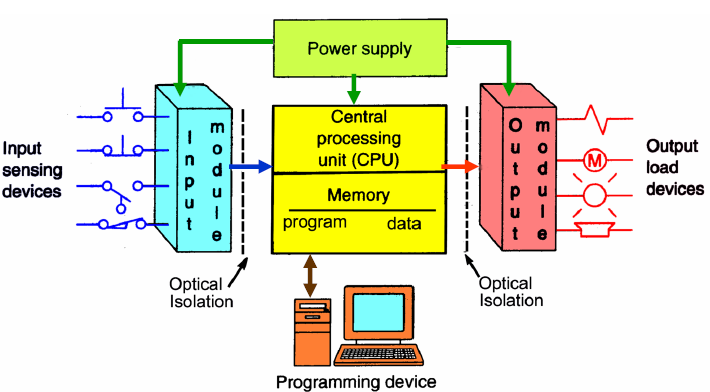
PLC (Programlanabilir Lojik Kontrolör = **P**rogrammable **L**ogic **C**ontroller), endüstriyel sistemlerin kontrol ve lojik kumanda işlevlerini gerçekleştirmek amacı ile modüler çevre birimlerine (dijital (Lojik) giriş/çıkış, analog giriş/çıkış, haberleşme, vb. ) sahip, programlanabilir, mikroişlemci tabanlı, endüstriyel uygulamalar için özelleştirilmiş bir bilgisayar olarak tanımlanabilir.



Geri besleme (ölçme sistemleri) yolu ile elde ettiği analog ve/veya lojik verileri hafızasına yüklenmiş olan program aracılığı ile değerlendirerek sistemin istenilen şekilde çalışması için gerekli olan kontrol/kumanda işaretlerini üretir.

Endüstride yaygın olarak kullanılmalarının temel sebepleri aşağıdaki gibi özetlenebilir.

i) Endüstriyel standartlara uygun olarak geliştirilmiş çevre birimleri (Dijital (On/Off) giriş/çıkış, Analog giriş/çıkış, Haberleşme, vb) ile endüstriyel uygulamar için gerekli donanımın çok hızlı ve kolay bir şekilde temin edilebilmesi/kurulabilmesi,

ii) Modüler yapı sayesinde sistemin analog/dijital giriş/çıkış sayılarının ihtiyaca göre tasarım sonrasında da rahatlıkla artırılabilmesi,

iii) Diğer sayısal işlemcilere (mikrodenetleyiciler, DSP, bilgisayar, vb) kıyasla program geliştirmenin daha kolay olması, dolayısıyla programlama için uzmanlık gerektirmemesi

iv) Endüstriyel uygulamalar için özel olarak üretildiklerinden çevresel şartlara (toz, sıcaklık, titreşim, vb.) karşı dayanıklılık

Ticari olarak ilk PLC sistemi 1969 yılında Modicon firması tarafından bir otomotiv fabrikasında uygulanmıştır.



PLC öncesinde gerekli sayı ve türdeki elektromekanik röleler istenilen lojik mantığa uygun kablo bağlantısı yapılarak kumanda işlemleri yerine getiriliyordu.

Üretilen ürün üzerinde değişiklik yapılmak istendiğinde kumanda mantığıda değişmesi gereketiğinden röleli sistemlerde bu işlem hem çok zaman alıcı hemde maliyet artırıcı olmakta idi.

Sayısal işlemci tabanlı sistemlerde (örneğin PLC) ise kontrol algoritmasında veya kumanda mantığında değişiklik yapmak yazılım aracılığı ile yerine getirildiğinden son derece kolay ve hızlı bir şekilde gerçekleştirilebilmektedir.

Günümüzde PLC’ler giriş-çıkış sayısı, program belleği, işlem yeteneği gibi özellikler açısından ilk üretilen PLC’ler ile kıyaslanamayacak bir düzeye ulaşmaştır. Dolayısı ile ilk kullanımlarında yerine getirdikleri işlevlerden esinlenerek verilen PLC (Programlanabilir Lojik Kontrolör) ismi mevcut cihazların özelliklerini yeterli derecede kapsamamaktadır.

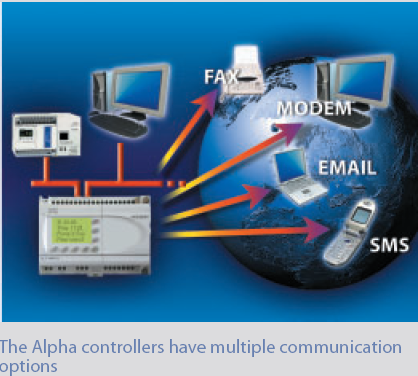
Bugün PLC tabanlı kumanda, ölçme, veri toplama ve geribeslemeli (kapalı çevrim) kontrol sistemleri bir çok endüstriyel alanda (otomotiv, petro-kimya, askeri, vb) yaygın olarak kullanılmaktadır.

Birçok firma (ABB, Allen-Bradley, Festo, Fuji Electric, GE-Fanuc, Hitachi, Honeywell, LG Industrial Systems, Mitsubishi, Omron, Rockwell Automation, Schneider Electric, Siemens, Toshiba, Yokogawa, …) tarafındam çeşitli türde PLC ticari olarak üretilmektedir.

PLC’ leri iki temel grupta sınıflandırılabilir.

i) Basit kumanda işlemleri için üretilmiş sınırlı-giriş çıkış sayısına sahip, kendi üzerinden programlanabilen akıllı röle olarakda adlandırılan mikro PLC’ler, Örnek; *Picocontroller –*Allen Bradley*, LOGO-*Siemens*, ALPHA-*Mitsubishi*, Zen-*Omron,*…*





ii) Üretici firma tarafından belirlenmiş sayıda çevre biriminin (Dijital giriş/çıkış, Analog giriş/çıkış, Haberleşme, vb) ilave edilebildiği **Modüler PLC**’ler. Modüler PLC’lerde maksimum çevre birimi (giriş/çıkış) sayısı, işlem hızı vb. Özelliklere gore küçük, orta ve üst düzey olarak üç grupta incelenebilir.







**PLC sistem bileşenleri**

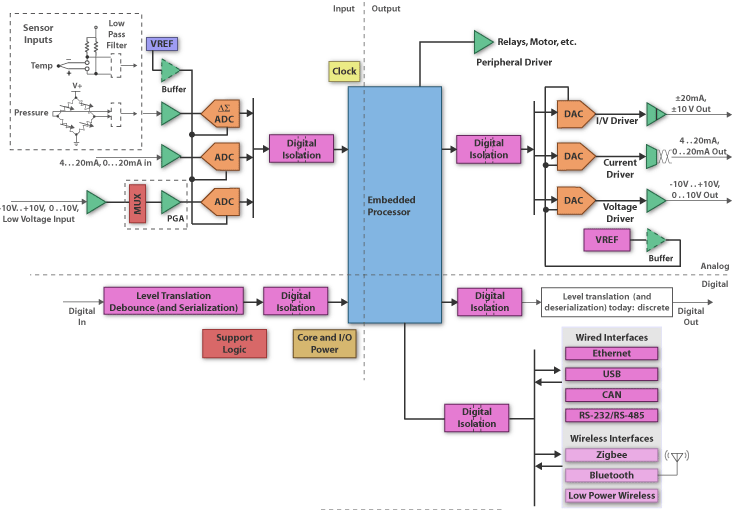
Genel bir PLC sistemi iki temel yapıdan oluşur.

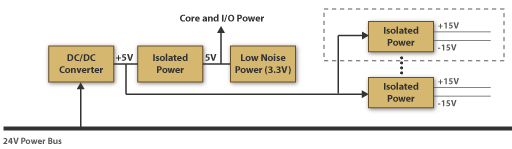
i) Merkezi İşlem Birimi (Central Processing Unit, CPU)

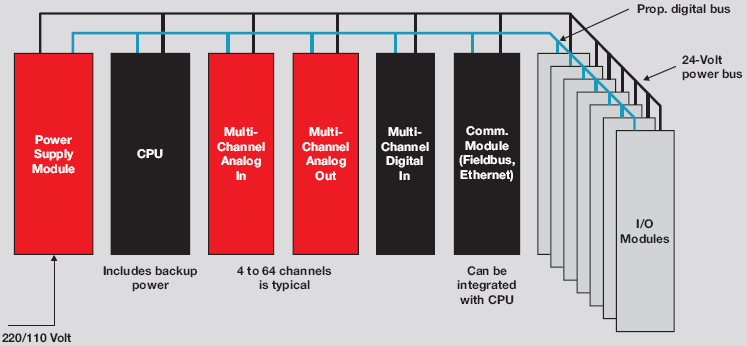
ii) Giriş / Çıkış Birimleri (Dijital giriş/çıkış, Analog giriş/çıkış, Yüksek-Hızlı Sayıcı (High-Speed Counter , HSC), Haberleşme birimleri, PID Modülleri, vb.)

Giriş / Çıkış Birimleri CPU’ ya modüler olarak bir veri kanalı (rack/bus) aracılığı ile bağlanır. CPU ile giriş/çıkış modülleri arasındaki veri iletişimi bu kanal aracılığı ile gerçekleştirilir.

**Genel PLC Mimarisi**

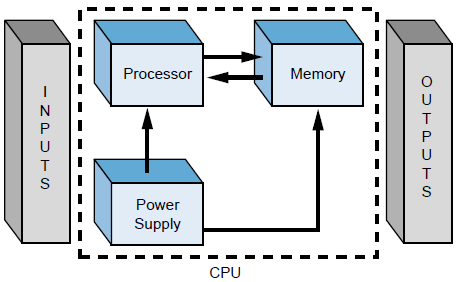






**PLC Merkezi İşlemci Birimi , CPU**

CPU PLC sisteminin belirli bir düzen içerisinde (girişlerin okunması, kullanıcı programın koşturulması, çıkışların güncellenmesi, sistemin hata denetiminin yapılması, vb.) çalışmasını sağlayan birimdir. Yerine getirdiği işlevler açısından PLC sisteminin en önemli elamanıdır.



(L.A.Bryan, E.A.Bryan)

CPU bünyesinde mikroişlemci (veya mikrodenetleyici) ve hafıza birimleri (program ve veri hafıza) bulunur. Bir üretici firmanın farklı özelliklere sahip PLC aileri olabildiği gibi (örneğin Siemens firması için S7-200/S7-1200, S7-300/ S7-1500 ve S7-400 ) aynı aile içerisinde farklı CPU türleride (S7-1200 ailesi 5 farklı CPU’ya sahiptir) olabilmektedir.

**CPU tanımlayan Temel Parametreler**

1-) Bağlanabilecek ilave modül sayısı ( DI/DO, AI/AO, Haberleşme, vb.)

2-) Çevrim Hızı (Execution Speed) , farklı komutlar (Lojik, Aritmetik, vs.) için koşturma süresi

3-) Program ve veri hafıza büyüklüğü

4-) **Floating Point** işlem kabiliyeti

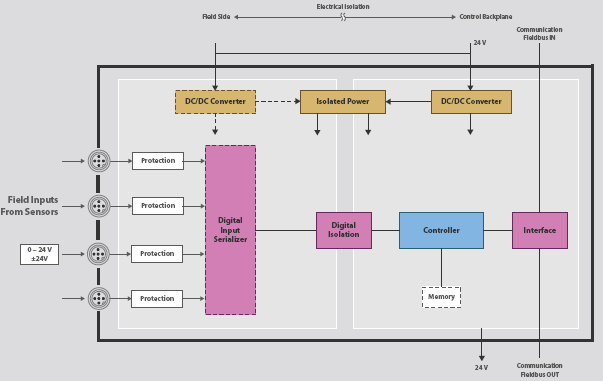
5-) Üzerinde mevcut haberleşme portları/ desteklediği haberleşme protokolleri

6-) Besleme Gerilimi, Güç tüketimi

7-) Çalışabileceği çevresel (sıcaklık, nem, titreşim, toz, vb.) şartlar

**Dijital Giriş (DI)**

Yaklaşım (endüktif, kapasitif, optik, vb.) anahtarları, limit anahtarlar, buton, vb. On/Off (Lojik) çıkış üreten sensörlerin bağlandığı PLC giriş birimleridir. Yaygın olarak 24V DC kullanılmaktadır.

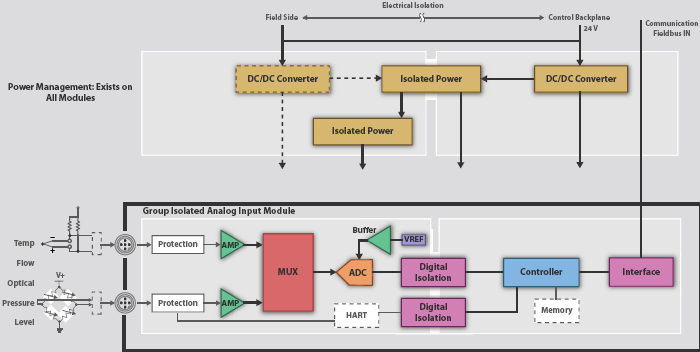


**Dijital Çıkış (DO)**

Röle, selenoid, uyarı lambaları, vb. ikili duruma (on/off) sahip aktüatörlerin konum değiştirmesini sağlayan çıkış birimleridir.

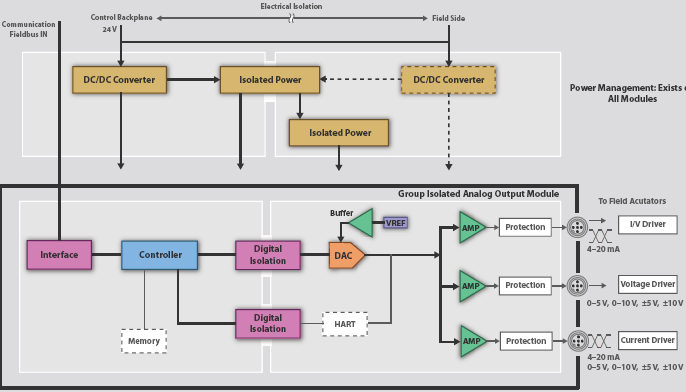
**Analog Giriş (AI)**

Sıcaklık, basınç, seviye, konum, vb. fiziksel büyüklüklerin uygun bir ölçme devresinde sonra elde edilen analog çıkış değerlerinin bağlandığı PLC giriş birimleridir. Analog giriş kanalları akım (0- 20 mA) ve gerilim girişli (0–5V, 0–10V, ±5V and ±10V) olabilmektedir.



**Analog Output Modules**

Analog çıkış birimleri kontrol algoritmasının üretmiş olduğu sayısal kontrol işaret değerini analoğa dönüştürerek sürücü sisteme aktaran çevre birimidir. Analog çıkış kanalları akım (4 - 20 mA) ve gerilim çıkışlı (0–5V, 0–10V, ±5V and ±10V) olabilmektedir.

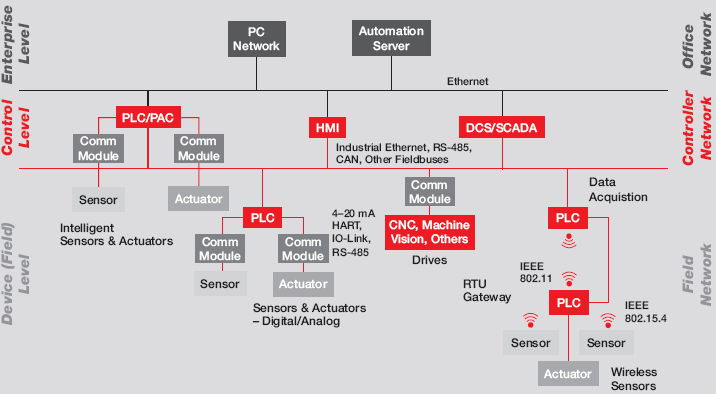


**Haberleşme (Arayüz) Birimleri**

PLC sistemlerinde teknolojik gelişmelere paralel olarak bir çok haberleşme seçeneği bulunmaktadır. Haberleşme arayüzlerini iki ana grupta toplamak mümkündür.

i) Kablolu haberleşme arayüzleri; RS-232/RS-485, USB, Profinet, Ethernet, CAN

ii) Kablosuz haberleşme arayüzleri; ZigBee, Bluetooth, GSM, …



**PLC İşletim Sistemi**

PLC’ ler kullanıcı programın yüklenmesi ve yürütülmesi, giriş/çıkış modülleri ile haberleşme ve hata denetimi gibi işlemleri gerçekleştiren temel bir işletim sistemine sahiptir.

**Kullanıcı Programın İcrası**

Normal durumda beslemesi verildikten sonra PLC program hafızada yüklü olan kullanıcı programını ilk komuttan başlayarak son komuta kadar, programda belirlenen akışa uygun olarak, sırayla icra eder. Son komut icra edildikten sonra programın akışı tekrar ilk komuta döner. Bu çalışma biçimi sonsuz veya sürekli çevrim olarak adlandırılır.

Her çevrimin belirli bir sürede tamamlanması gerekir. Çevrim (tarama) süresi için tanımlanan maksimum süre bazı modellerde sabit iken bazı modellerde kullanıcı tarafından ayarlanabilmektedir. Çevrim süresi “Watchdog Timer” tarafından sürekli olarak denetlenir ve çeşitli nedenlerle belirlenen süre içerisinde çevrim tamamlanamaz ise PLC hata konumuna geçer.

**PLC Çevrimi**

PLC “run” konumuna geçtiğinde bir dizi işlemi sürekli olarak yapar. Bu işlemlerin döngüsel icrasına tarama (çevrim) adı verilir. Bir çevrim esnasında sırasıyla aşağıda belirtilen işlemler yürütülür.

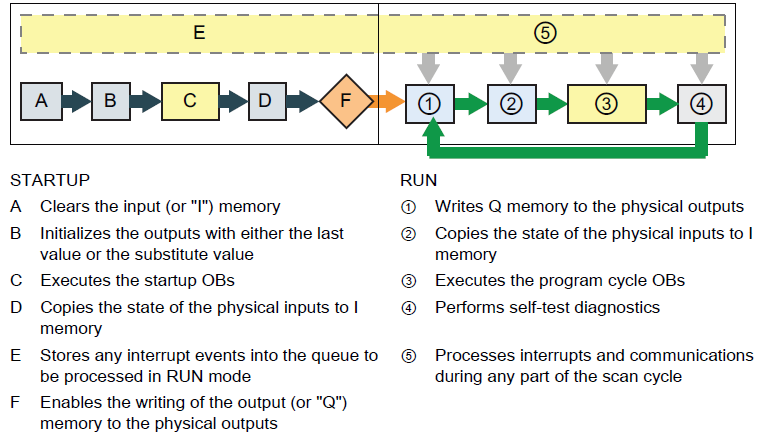
1-) Tanımlı tüm girişler okunarak “giriş görüntü belleği” olarak adlandırılan hafıza alanına yazılır. Girişlerin okunan bu değerleri bir sonraki çevrime kadar değişmez ve kullanıcı programın icrasında hafızaya yazılan bu değerler kullanılır.

2-) Kullanıcı programı icra edilir. Programın icrasında çıkış değerleri “çıkış görüntü belleği” olarak adlandırılan hafıza alanına yazılır.

3-) Varsa haberleşme taleplerini yerine getirir

4-) Çevre birimleri ile birlikte hata testi yapılır

5-)Hata yok ise çıkış görüntü belleğine aktarılan değerler fiziksel çıkışlara yazılarak çıkış değerleri güncellenmiş olır.

****

Bir çevrimin tamamlanma süresi işlemci hızına, kullanılan I/O sayısına ve kullanıcı programın uzunluğuna/hesap yüküne bağlı olarak değişir. Genel olarak bir çevrim 1-2 ms içerisinde tamamlanır.

Çalışma anında, çevrim süresi bir *“watchdog”* tarafından gözlemlenir ve tanımlı olan çevrim süresi aşıldığında *“hata”* oluşur.

PLC’ lerde 3 farklı işlem tanımlanabilir.

i-) herhangi bir koşula bağlı olmaksızın her çevrimde icra edilen işlemler

ii-) periyodik olarak icra edilen işlemler

iii-) olaya (duruma) bağlı olarak icra edilen işlemler

**Girişlerin Okunması**

*Dijital girişler:* Her çevrim, dijital girişlerin anlık durumlarının okunması ve PLC hafızasında giriş imge alanı (Process Input Image Area, PII) olarak adlandırılan ve bu iş için özel olarak ayrılmış hafıza alanına yazılmasıyla başlar. Kullanıcı programın yürütülmesinde dijital girişlerin değeri bu hafıza alanından okunur.

*Analog girişler:* Analog giriş filtreleme seçilmediyse, S7–1200 analog girişleri normal taramanın bir parçası

olarak güncellemez. Analog filtreleme seçeneği daha dengeli bir sinyal sağlamak için kullanılabilir. Analog

filtreyi her analog giriş kanalı için ayrı ayrı devreye sokabilirsiniz.

Analog filtreleme devreye sokulduğunda S7–1200 analog girişi her taramada bir kez okur, filtreleme

işlemini yapar ve filtre edilmiş değeri dahili olarak saklar. Program o analog girişe her erişmek istediğinde

en son filtrelenmiş değer sağlanır.

Analog filtreleme devreye sokulmadıysa, fiziksel modülden okunan değer, program o analog girişe eriştiği

zaman güncellenir. Bir başka deyişle, bu durumda analog girişler gerçek zaman (real time) olarak

değerlendirilir.

***Üretici Bilgi Notu***

Analog giriş filtreleme daha dengeli, anlık parazitlerin dikkate alınmadığı bir analog sinyal sağlamak üzere düşünülmüş olup seçilen tarama sayısında okunan değerlerin ortalamasının alınması prensibine dayanır. Yani, süratli değişmesi beklenen sinyaller için analog filtrelemenin kullanılması uygun değildir.

Ayrıca, alarm durumlarını veya dijital bazı bilgileri de gönderen modüllerde analog filtreleme kullanılmamalıdır. Bu nedenle RTD, Thermokupl ve AS–Interface Master modüllerinde analog filtrelemeyi devre dışı bırakınız.

**Kullanıcı Programın Koşturulması**

Kullanıcı programı ilk satırdan başlayıp son satıra kadar işlenerek çalıştırılır. Programın icrasında gerekli giriş değerleri PII’den okunur çıkış değerleride benzer şekilde PLC hafızasında çıkış imge alanı (Process Output Image Area, PIQ) olarak adlandırılan ve bu iş için özel olarak ayrılmış hafıza alanına yazılır.

Bazı PLC türlerinde “Anında giriş/çıkış” (Immediate I/O) komutları, program icrası esnasında fiziksel giriş ve çıkışlara doğrudan erişim sağlar. Bir başka deyişle bu komutlar PII ve PIQ’yu kullanmadan direkt olarak fiziksel alanlara erişir.

Eğer programınızda “interrupt”lar kullanıyorsanız, interrupt olgularıyla ilişkilendirilmiş interrupt (Interrupt=Yarıda Kesme) altprogramları, programınızın bir parçası olarak saklanır. Interrupt altprogramları normal taramanın bir parçası olarak değil, ilgili olduğu interrupt olgusu gerçekleştiğinde icra edilirler. Bu icra, taramanın herhangi bir noktasında ve normal tarama kesilerek o anda gerçekleştirilir. Bu nedenle, interrupt olguları, çok süratli gelişmesi ve/veya kısa sürmesi beklenen durumların izlenmesi ve kontrol edilmesi için kullanılır.

**Haberleşme Taleplerinin Yerine Getirilmesi**

Haberleşme portu veya akıllı giriş/çıkış modüllerinden gelen mesajlar değerlendirilir.

**Kendi Kendinin Kontrol Edilmesi (Self–test Diagnostics)**

Herhangi bir hataya karşı CPU, hafıza alanları ve genişleme modüllerinin durumu değerlendirilir.

**Çıkışların Yazılması**

Her taramanın sonunda PIQ’ya atanmış olan çıkış değerleri, fiziksel çıkışlara aktarılır (Analog çıkışların değeri, taramadan bağımsız olarak icra edildikleri anda fiziksel analog çıkışlara aktarılır).

PLC çevriminde her bir işlemin (girişlerin okunması, kullanıcı programın icra edilmesi, çıkış değerlerinin fiziksel olarak atanması, vs.) birbirinden ayrı ve bağımsız bir işlem olduğu bilinmelidir.

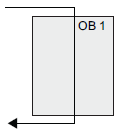
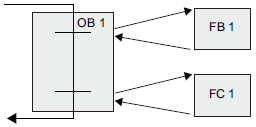
**Programlama**

Tüm sayısal işlemcilerde olduğu gibi PLC’lerdede donanımın herhangi bir işlevi ne zaman ve nasıl yerine getireceği komutlar aracılığı ile belirtilir. Komutların uygun bir editörde belirli kurallara uygun olarak yazılması ile elde edilen program PLC’nin kullanılacağı sistemde yerine getireceği fonksiyonları belirler.

PLC’ler (genel olarak sayısal işlemciler) için program geliştirme 3 farklı şekilde yapılabilir: Doğrusal (Linear), Bölünmüş (Divided), Yapısal (Structured)

**Linear program**

By a linear program the entire program has been created directly in OB1 and the CPU executes only the OB1. Thereby the instructions processed from the top to bottom and if is finished, the program starts again from the beginning.

**Lineer  Modular/Divided**

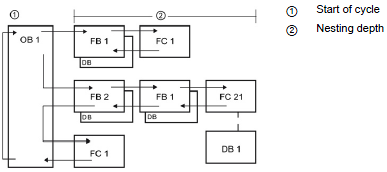
**Modular / Divided program**

A divided program is a program in which the program is stored in one or more functions or function blocks. Thereby the operating system calls OB1. OB1 in turn call other modules (functions, function blocks) according to a predetermined sequence. Each called block takes on a partial task. This method has compared to the linear program the advantage that a control task can be divided into several partial tasks. This preserves the clarity of the program and in case of errors and changes the focus has to be on this part of the program.

**Structured program**

A structured program is basically like a divided program. The only difference is that in a structured program the same function module is called several times and is parameterized each time with different parameters. Thus for a structured program you need parameterizable function modules. Because by calling only these can call different data blocks, which can contains different actual parameters.

A modular program calls specific code blocks that perform specific tasks. To create a modular structure, you divide the complex automation task into smaller subordinate tasks that correspond to the technological functions of the process. Each code block provides the program segment for each subordinate task. You structure your program by calling one of the code blocks from another block.



İç içe dallanma derinliği…

Program çevrim veya StartUP OB ‘ lerinden 16,

Herhangi bir kesme OB’ sinden 6

PLC’ler için tanımlanmış (IEC 61131-3 standartı) 5 ayrı program geliştirme tekniği mevcuttur.

i ) Merdiven diyagramı, (Ladder diagrams, LAD)

ii) Fonksiyonel blok diyagramı, (Functional Block Diagram, FBD)

iii) Komut kümesi, (Statement List, STL)

iv) Yapısal programlama, structured text (ST)

v) Ardışıl fonksiyon grafiği, (Sequential Function Charts, SFC)

Bu programlama tekniklerinden en yaygın olarak kullanılanlar göreceli olarak daha basit ve anlaşılır yapıda olmalarından dolayı LAD ve FBD’dir. Belirtilen tekniklerden herhangi birinde yazılan program istenildiğinde, bazı istisnai komutlar hariç, diğer editörlerdede gösterilebilmektedir.

Çoğu PLC’ler benzer komutlar içerir, ancak firmadan firmaya görünüş, işlem ve diğer açılardan ufak

farklılıklar olabilmektedir.

**LAD Editörü**

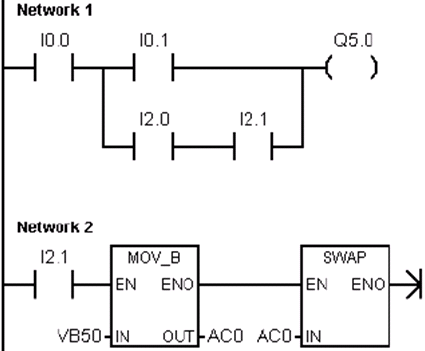
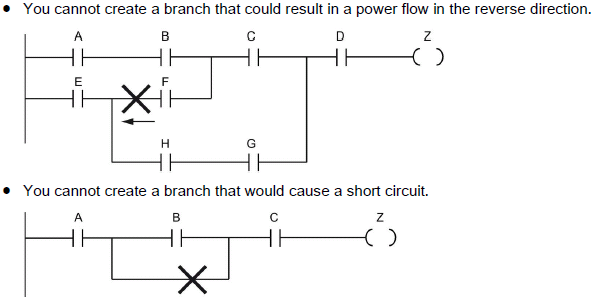
LAD editörü, programı elektriksel bağlantı resmine çok yakın bir şekilde şematik olarak gösterir. Şematik gösterim şeklinin anlaşılması kolay olduğundan yaygın olarak kullanılır. Aslında LAD’de yazılmış program elektrik devre şemasının 90 derece döndürülmüş hali olarak düşünülebilir.

Ladder programları tıpkı gerçek elektrik devrelerindeki gibi “bir enerji kaynağından kontaklar vasıtasıyla

akan enerjiyi” sembolize etmek şeklinde, kullanıcıya kolay gelebilecek gösterim mantığına sahiptir. LAD

programında sol tarafta gösterilen dikey çizgi enerji kaynağını sembolize eder. ‘Kapanmış’ olan kontaklar

enerji akışına izin verirken ‘açılmış’ kontaklar bu sembolik akışı bloke ederler.

Program devre (network) olarak tanımlanan parçalara bölünmüştür. Program akışı her devrede soldan sağa ve yukarıdan aşağıya doğrudur.

Değişik komutlar grafik sembollerle gösterilmekte olup üç ayrı grupta incelenebilir.

Kontaklar; sviç, buton, şalter veya dahili koşullargibi lojik girişlere işaret eder.

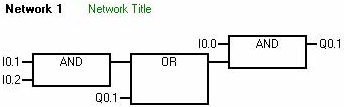
Bobinler; lamba, kontaktör veya dahili çıkış gibi lojik sonuçlara işaret eder.

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

**FBD Editörü**

FBD editörü, lojik kapıların kullanımına dayanan şematik bir gösterim şekli sunar. LAD editöründe olduğu

gibi kontaklar ve bobinler yer almaz, ancak eşdeğer kutular halinde lojik kapılar bulunur. Program lojiği bu kutular arasındaki bağlantıların bir sonucudur. Yani, bir komutun (örneğin AND kapısının) sonucu bir başka komutun (örneğin bir zaman rölesinin) girişi olarak kullanılabilir. Bu bağlantı kavramı pek çok lojik problemin çözümünü sağlar.



FBD’de bir enerji kaynağı ve nötr hattı sözkonusu değildir. Enerji akışının kaynağı ve sonuçta ulaştığı nokta

direkt olarak bir operanda atanabilir. FBD elemanları için “lojik 1” akım akışı anlamındadır.

**STL Editörü**

STL editörü, programın metin olarak girilmesini sağlar. STL editörü, LAD veya FBD ile yazılamayacak

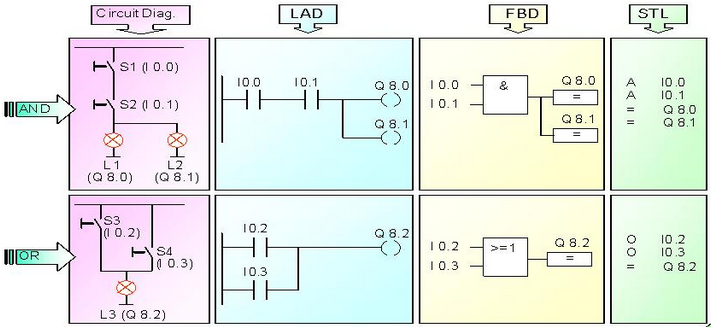
bazı özel komutların girilebilmesini de sağlar. Şematik gösterimin oluşması için geçerli bazı kısıtlamalar

STL’de sözkonusu olmadığından ve makine koduna en yakın gösterim şekli olduğundan, STL

komutları en geniş imkanları sunar. Ancak, bu programlama editörünün kullanımı ve mevcut programın takibi/anlaşılması göreceli olarak biraz daha zor olduğundan LAD ve FBD kadar yaygın değildir.

LAD veya FBD editörüyle yazılmış programları her zaman STL ile göstermek mümkün olduğu halde

bunun tersi doğru değildir. Bazı STL komutları LAD veya FBD ile görüntülenemez.



**ST Editörü**

Üst düzey programlama dilini (daha çok PASCAL) andıran bir yapıya sahiptir.

WHILE Counter<>0 DO  
Var1:=Var1\*2;  
Counter:=Counter-1;  
END\_WHILE

PLC üreticileri doğal olarak ürünlerinin programlanması için kendi özel yazılımlarınıda geliştirmişlerdir. Örneğin, Mitsubishi – MELSOFT, Siemens - SIMATIC STEP 7, TIA (Total Integrated Automation), Rockwell Automation – RSLogix.

• Typically, the program scanning takes place left to right across each rung and from the top to bottom rungs

*rung scanning*, it is the method used, for example, by Allen-Bradley

*column scanning* used by Modicon/Schneider

– Here the processor "looks" at the first contact at the top left corner and reads the first column from top to bottom – It next reads the second column from top to bottom, and so on

• Either method, rung or column, is appropriate.