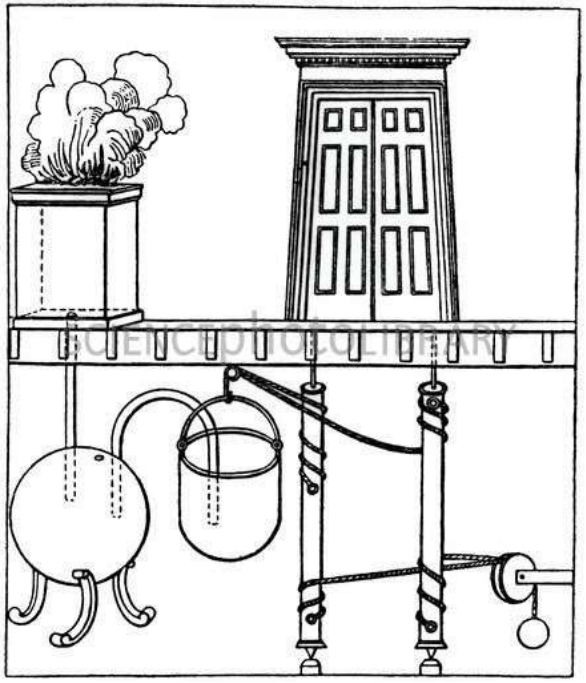
Principles of Mechatronic Systems
مبانی سیستم های مکاترونیکی (جلسه پنجم)

By: Reza Tikani
Mechanical Engineering Department
Isfahan University of Technology

سیستم‌های کنترل

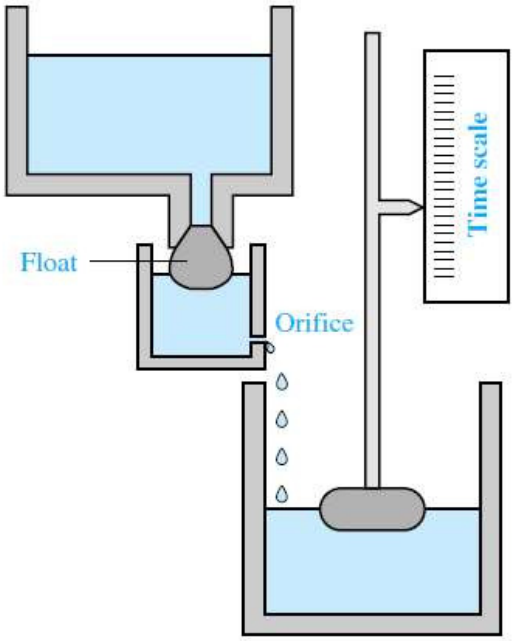
سیستمهای کنترل اتوماتیک تنها سیستمهای مدرن اطراف ما نیستند، این سیستمها از زمان‌های گذشته مورد استفاده قرار می‌گرفته‌اند.



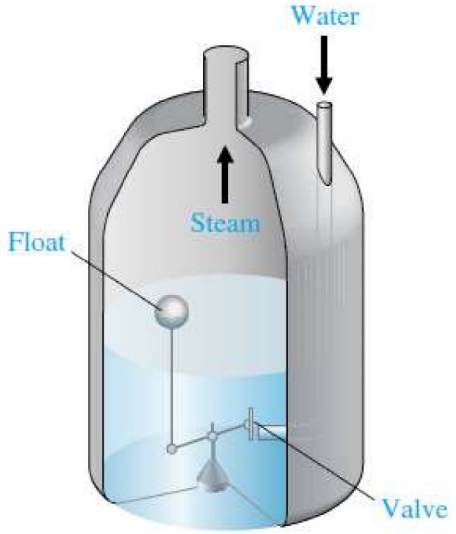
http://www.sciencephoto.com/media/364618/view

به نظر می رسد یکی از اولین سیستم های خودکار در کنترل باز و بسته کردن در معبدها به کار می رفته و مردم عادی را شگفت زده می نموده است . سیستم به گونه ای بود که وقتی آتش در معبد روشن می شده درهای معبد باز و با خاموش کردن آن درهای معبد بسته می شده است. با روشن شدن آتش ، هوا در لوله ی بسته ی زیر ان گرم شده و انبساط هوا آب را از مخزن به داخل سطل می راند و وزن آن را زیاد می کند تا اینکه بر وزنه تعادلی غلبه کرده و در را باز می کند.

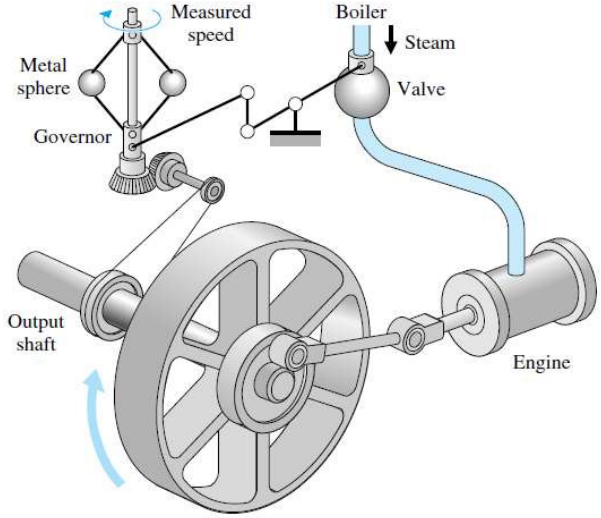
نمونه دیگری که از حدود چند دهه قبل از میلاد سراغ داریم ساعت آبی است که با تغییر سطح آب نمایشگر زمان جابه‌جا می‌شده است.



اولین کاربرد صنعتی سیستم های کنترل مکانیکی در قرن ۱۷ میلادی:



تنظیم کننده سطح



سیستم کنترل سرعت **Flyball governor**

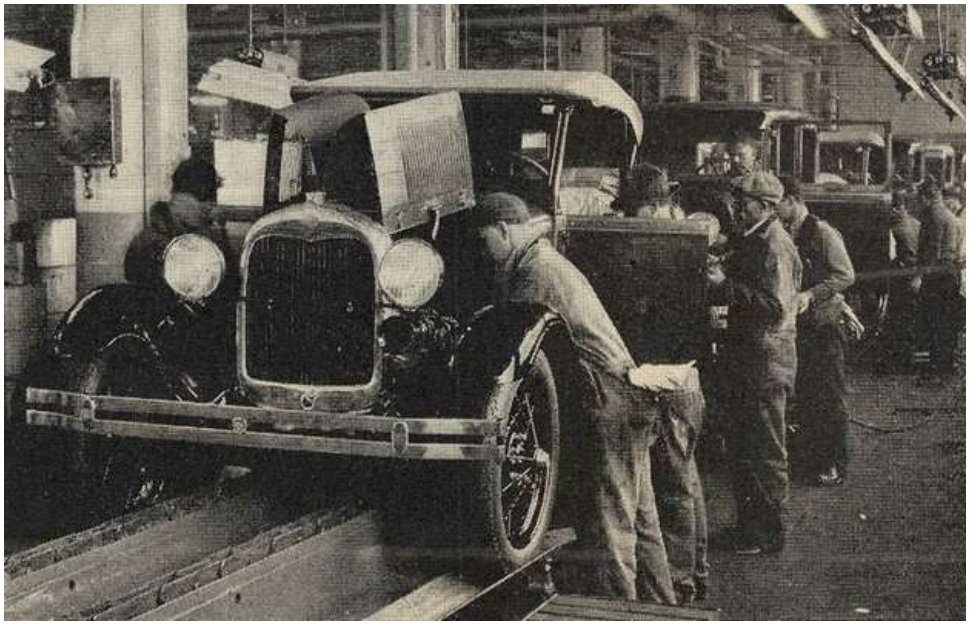
سیستم‌های کنترل

با ساخت مدارهای مجتمع **IC**، کنترل کننده های پیشرفته و قابل برنامه ریزی با هزینه کم عرضه شدند. سیستمهای کنترل امروزه در جنبه های مختلفی از زندگی روزمره ما مورد استفاده قرار می گیرند.

نمونه هایی از این کاربردها عبارتند از:

- کنترل کننده های خودکار در سیستم های تهویه مطبوع، دما و رطوبت هوای ساختمانها
- خط مونتاژ خودکار
- کنترل ماشین ابزار
- سیستم های حمل و نقل
- رباتها

سیستم‌های کنترل



سیستم‌های کنترل

مدارهای منطقی در سیستم های پیچیده قابل استفاده نیستند ضمن آنکه انعطاف ناپذیرند و برای تغییر در عملکرد بایستی سیستم جدیدی جایگزین شوند.

این مشکلات باعث توسعه سیستم های پردازنده با قابلیت برنامه ریزی گردید.

یک سیستم کنترلی مبتنی بر پردازنده دارای تعدادی ورودی و خروجی است که تمام پردازش هایی که قرار است روی ورودی ها انجام شود به صورت مجموعه ای از دستورهای نرم افزاری به سیستم داده شود.

میکروکنترلرها و **PLC**ها نمونه هایی از سیستمهای کنترل قابل برنامه ریزی می باشند.

سیستم‌های میکروپروسسور (ریزپردازنده)

سیستم های میکرو پروسسوری دارای سه بخش هستند:

۱- واحد پردازش مرکزی (**CPU**)

تشخیص دستورات و اجرای آنها، بر عهده این بخش است.

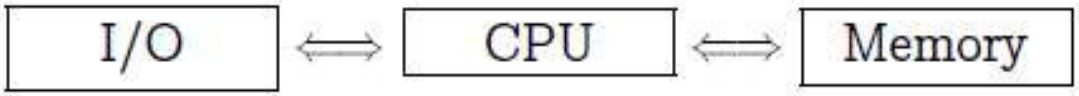
این بخش شامل ریزپردازنده و اجزای الکترونیکی وابسته به آن است.

۲- واسطه های ورودی و خروجی

برای برقراری ارتباط بین کامپیوتر و دنیای بیرون؛ عموماً از عبارت درگاه (**Port**) برای واسطه استفاده می‌شود.

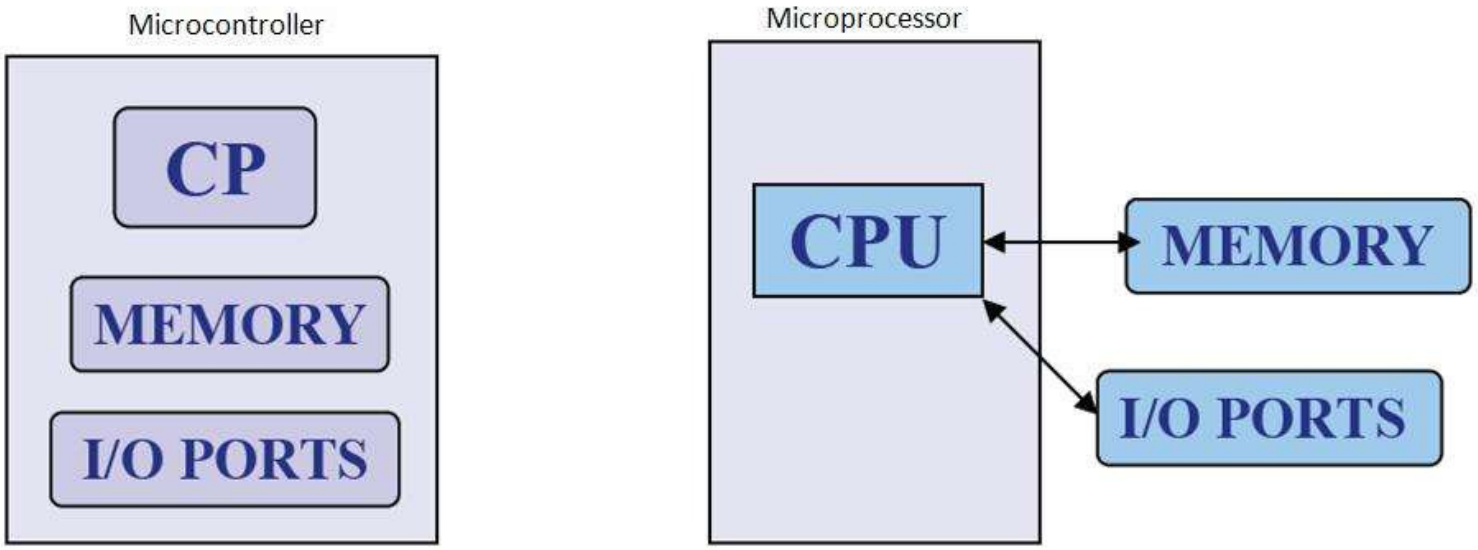
۳- حافظه

برای نگهداری دستورالعمل های برنامه و داده ها



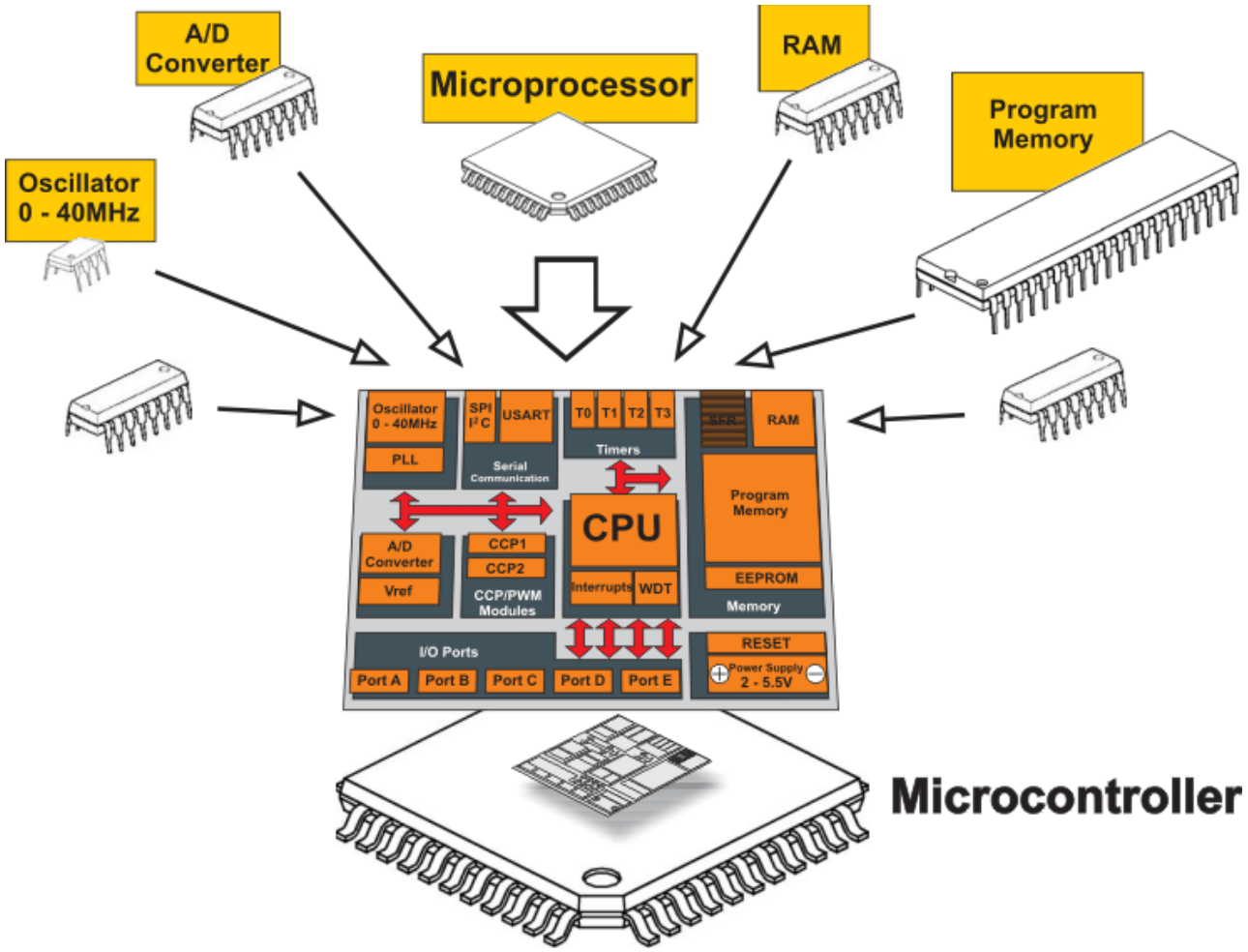
میکروکنترلر

ریزپردازنده هایی که حافظه و اجزای مختلف ورودی و خروجی را روی یک تراشه واحد دارند، میکروکنترلر (ریزکنترل کننده) می گویند.



http://studentsplaza.in/mobile.html

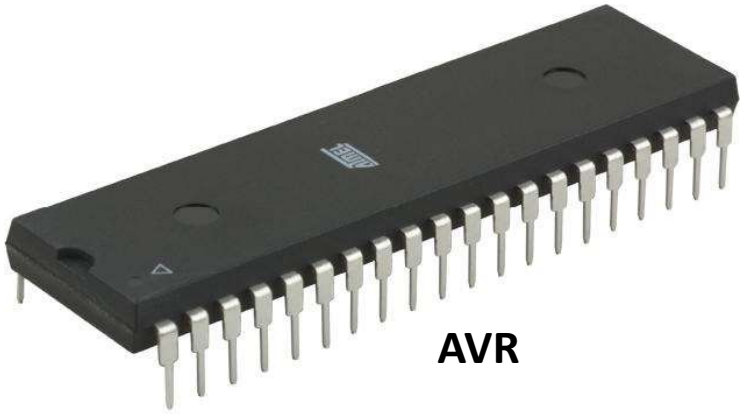
میکروکنترلر



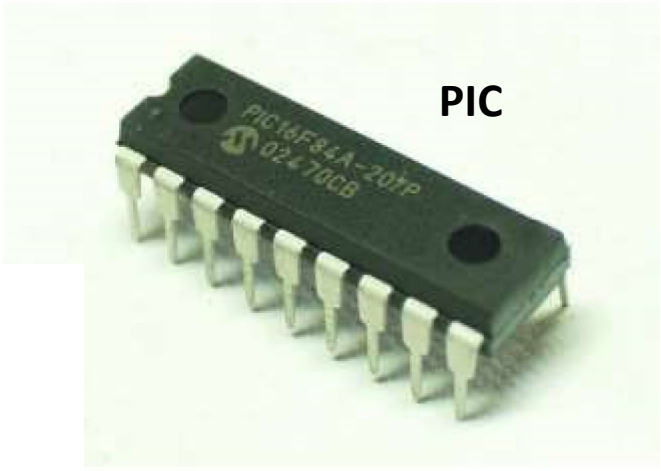
http://www.mikroe.com/chapters/view/1/introduction-world-of-microcontrollers/

میکروکنترلر

میکروکنترلرها در انواع متفاوت و در بسته بندی های متنوعی عرضه می شوند.



AVR

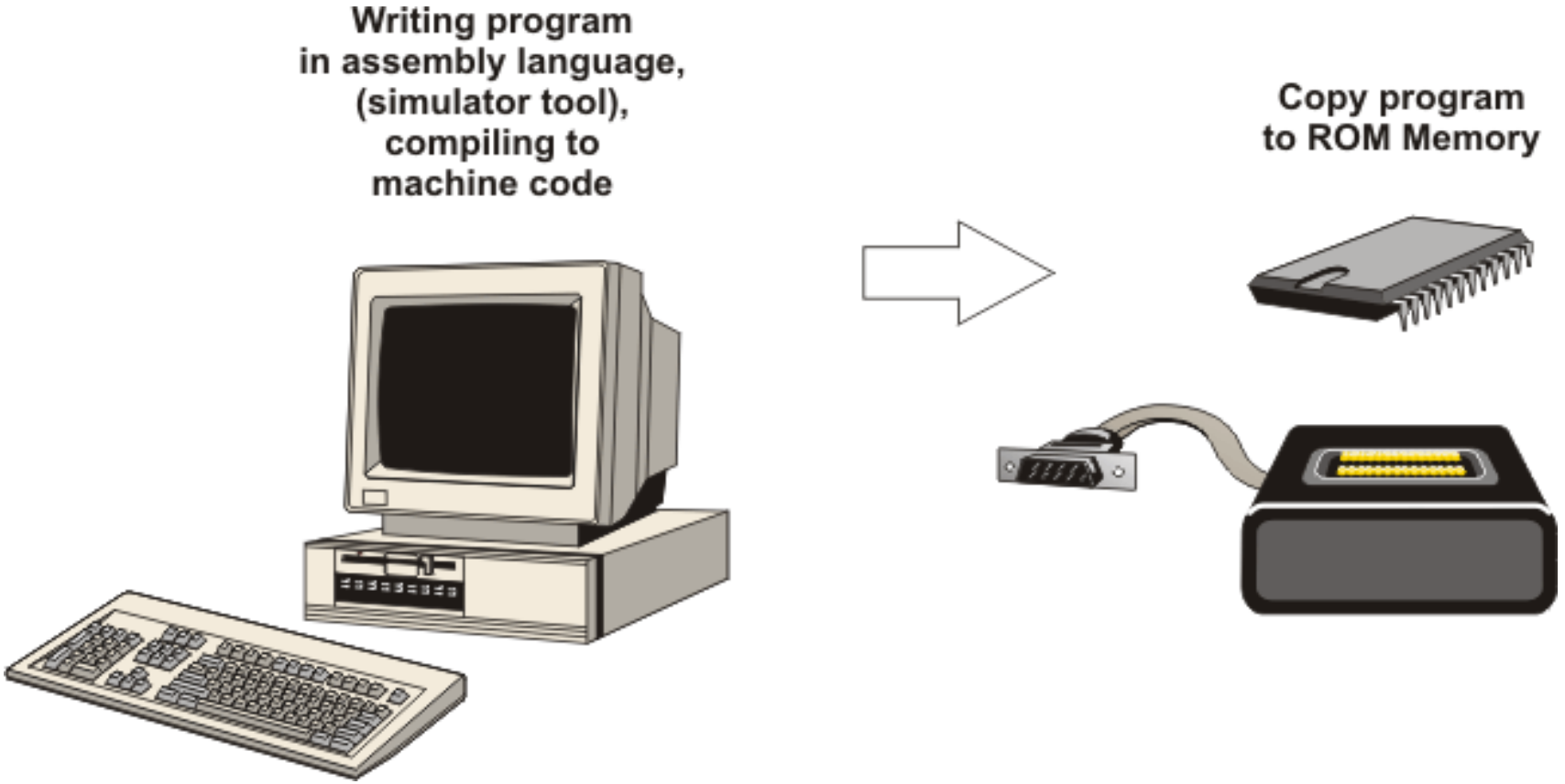


PIC

8051

میکروکنترلر

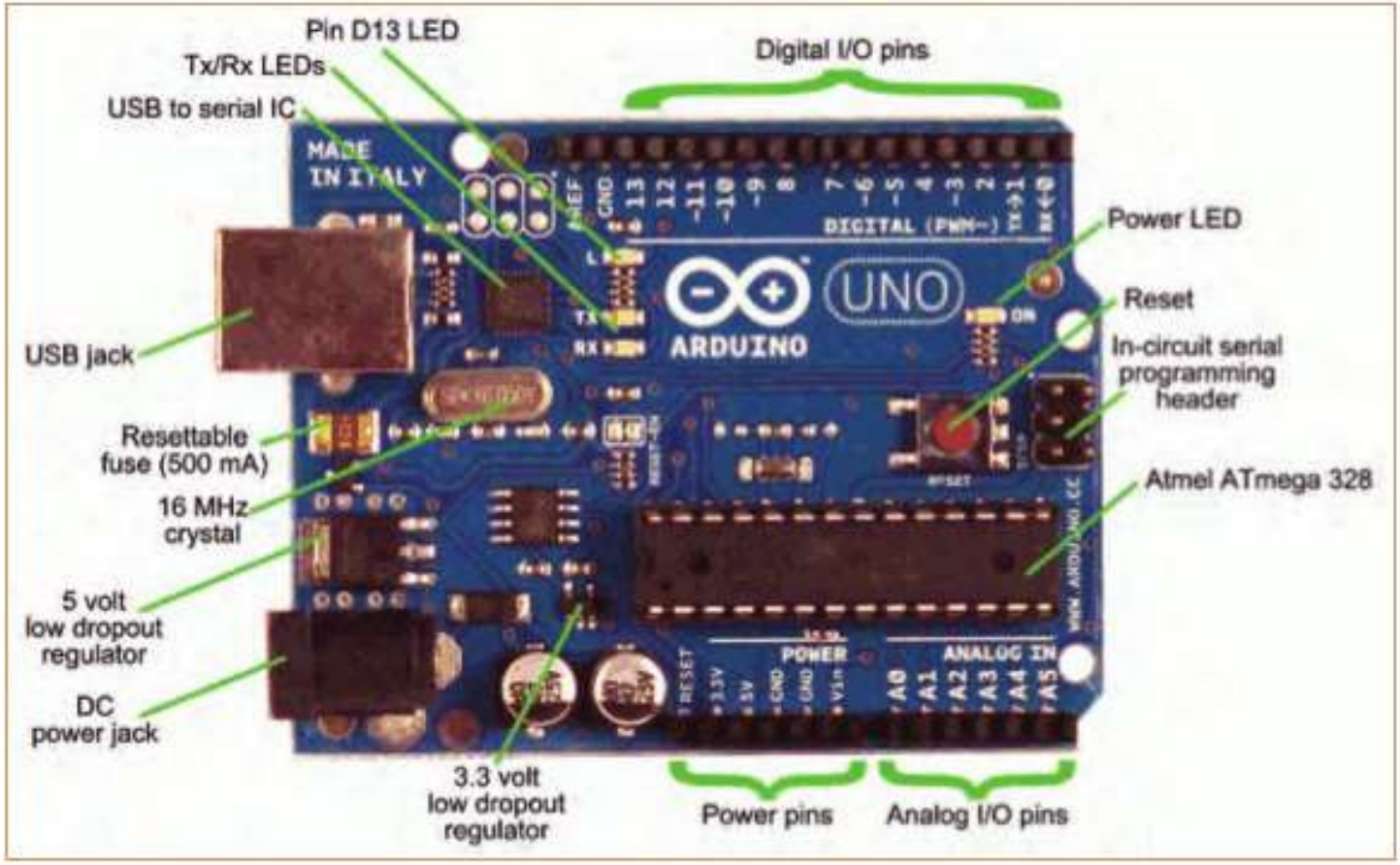
زبان برنامه نویسی اصلی میکروکنترلرها زبان اسمبلی است. هرچند با زبانهای دیگری مانند **Basic، C، Java** نیز قابل برنامه ریزی هستند.



http://www.mikroe.com/chapters/view/1/introduction-world-of-microcontrollers/

میکروکنترلر

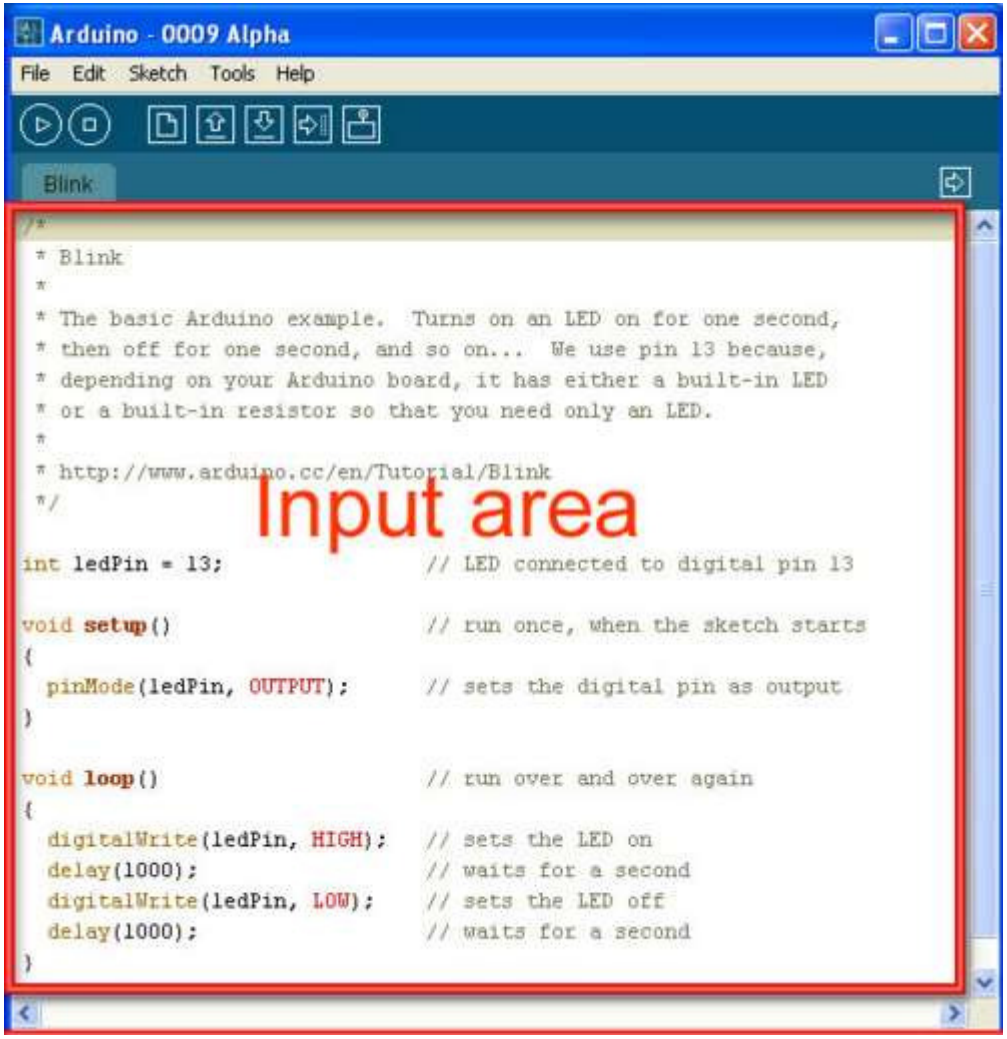
برد میکروکنترلر آردینو:



Making-robots-with-arduino.pdf

میکروکنترلر

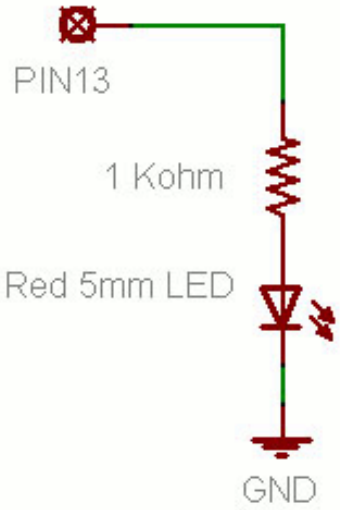
محیط برنامه نویسی:

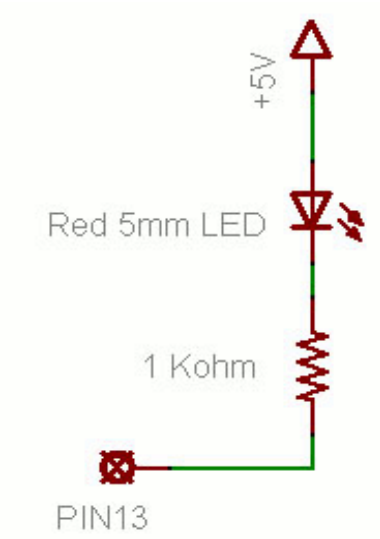


www.ladyada.net/learn/arduino

میکروکنترلر

برنامه نویسی:






```
int ledPin = 13;                  // LED connected to digital pin 13

void loop()                        // run over and over again
{
  digitalWrite(ledPin, HIGH);   // sets the LED on
  delay(500);                   // waits for a second
  digitalWrite(ledPin, LOW);    // sets the LED off
  delay(500);                   // waits for a second
}
```

www.ladyada.net/learn/arduino

تمرین و تحقیق

۱) در مورد فناوری خانه های هوشمند (Smart Home) تحقیق و چند کاربرد سیستمهای جاسازی شده در این خانه ها را ذکر نمایید.

۲) در مورد میکروکنترلرهای Ardiuno تحقیق نمایید و یک پروژه گزارش شده در اینترنت را با ذکر توضیح مراحل، به عنوان تکلیف تحویل دهید.

کنترل کننده منطقی قابل برنامه ریزی (PLC)

PLC، کنترل کننده ای نرم افزاری (قابل برنامه ریزی) است که اطلاعات را به صورت دودویی (باینری) دریافت نموده و آنها را طبق برنامه ای که در حافظه ذخیره شده است پردازش می کند.

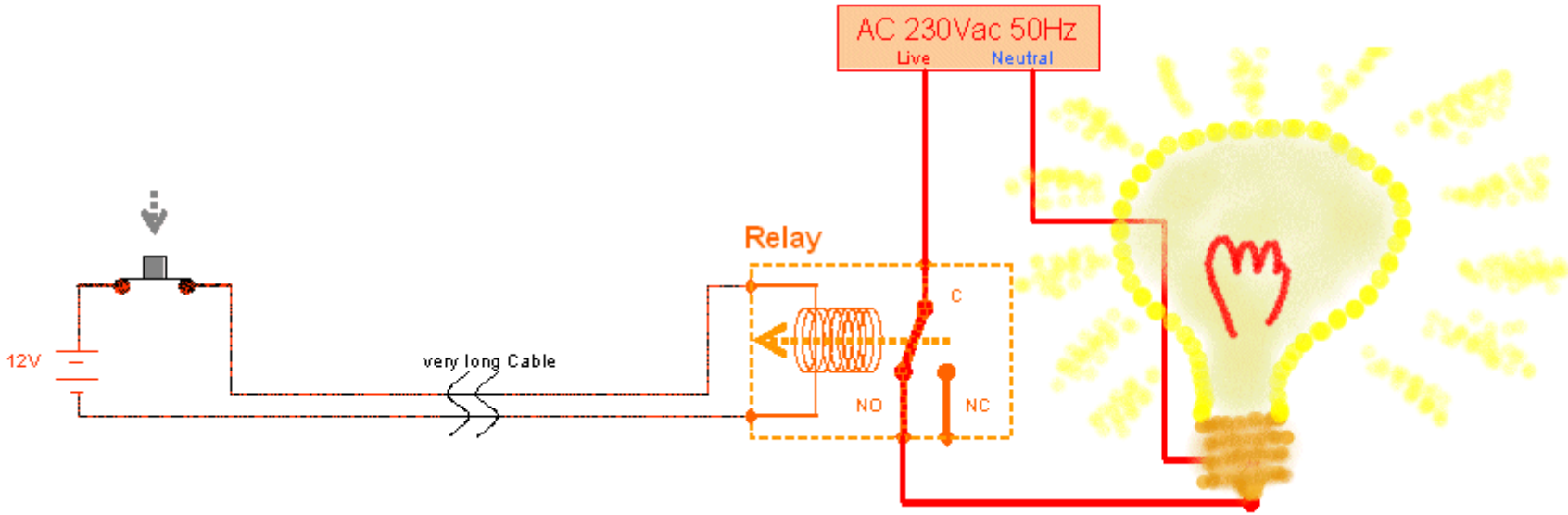
کنترل کننده منطقی قابل برنامه ریزی (PLC)

پیش از اختراع PLCها، مدارهای فرمان رله‌ای در محیط های صنعتی مورد استفاده قرار می گرفتند. اشکال این مدارها در آن است که با افزایش تعداد رله‌ها، حجم و وزن مدار فرمان بسیار زیاد شده و ضمناً افزایش قیمت را نیز به همراه خواهد داشت.

با استفاده از PLC تغییر در روند کنترل به راحتی و بدون نیاز به تغییر در سیم کشی و سیستم سخت افزاری صورت می پذیرد. با عوض کردن برنامه PLC این امر محقق می گردد.

رله ها

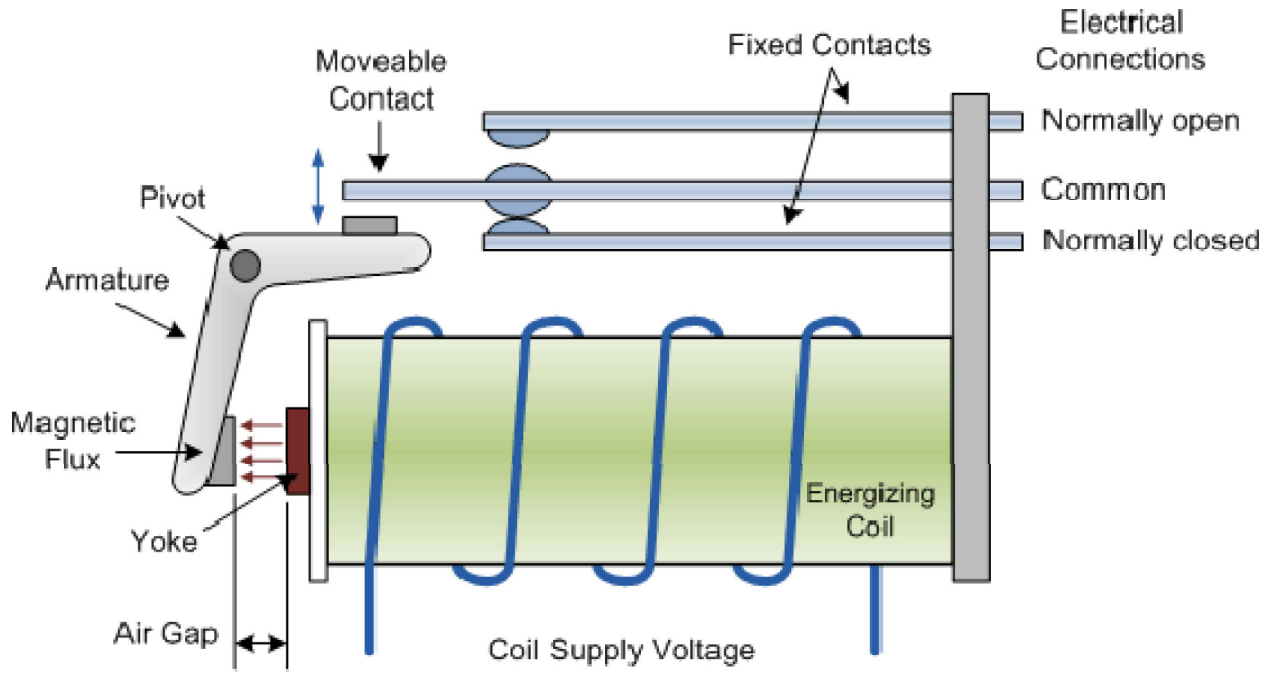
رله مکانیکی



http://www.siongboon.com/projects/2006-06-19_switch/

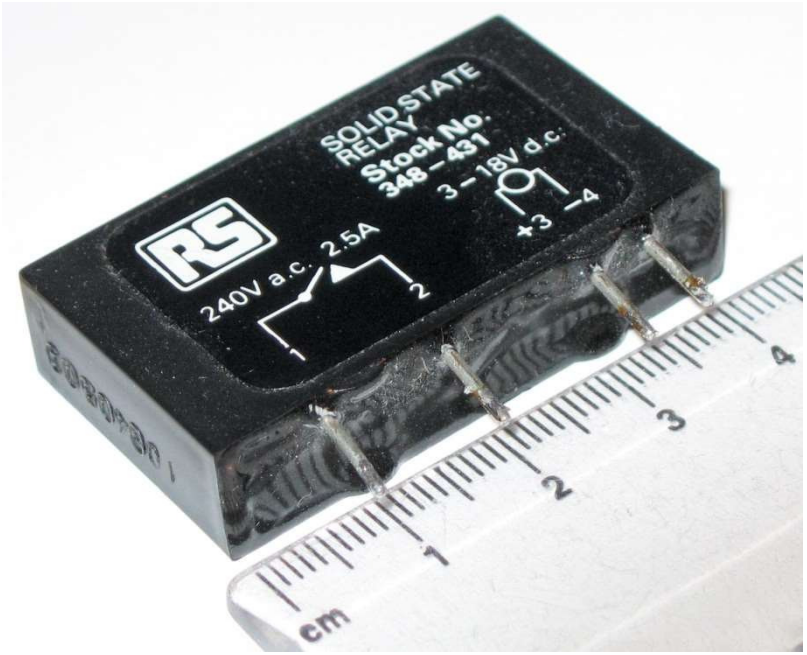
رله ها

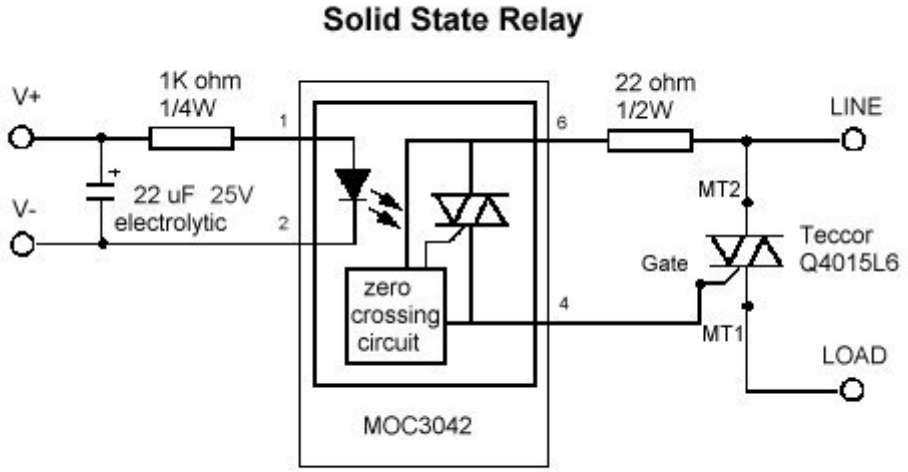
رله مکانیکی



رله ها

رله نیمه هادی

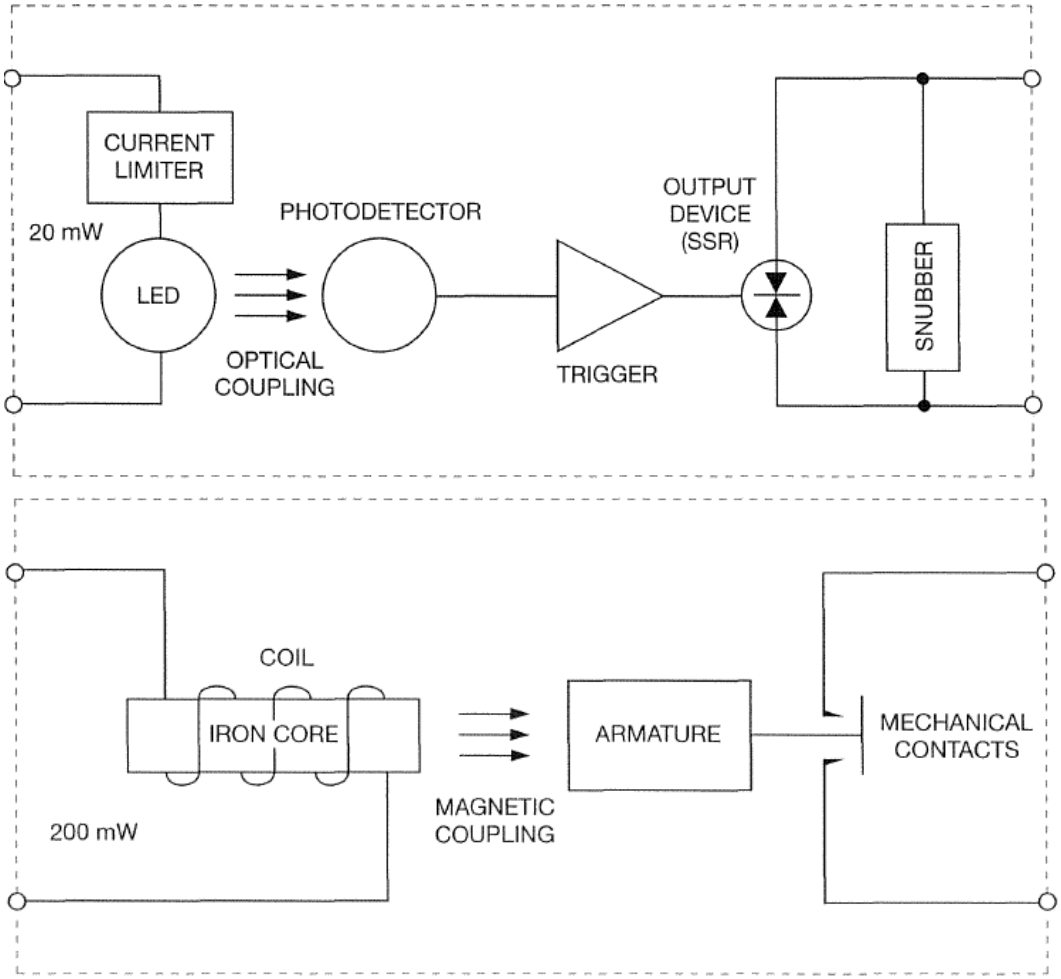




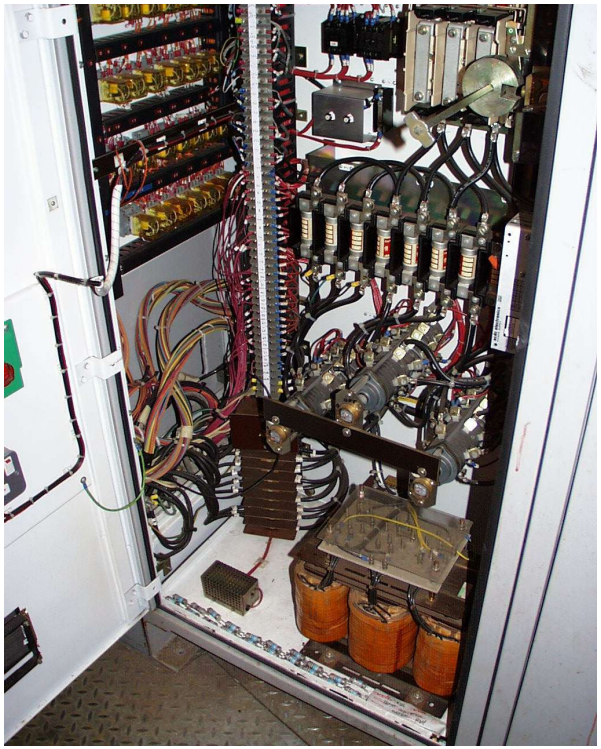
رله ها

مقایسه رله نیمه هادی با رله الکترومکانیکی:

۱. توان مورد نیاز
۲. سرعت
۳. نویز
۴. فرکانس کاری
۵. اتلاف انرژی
۶. قیمت

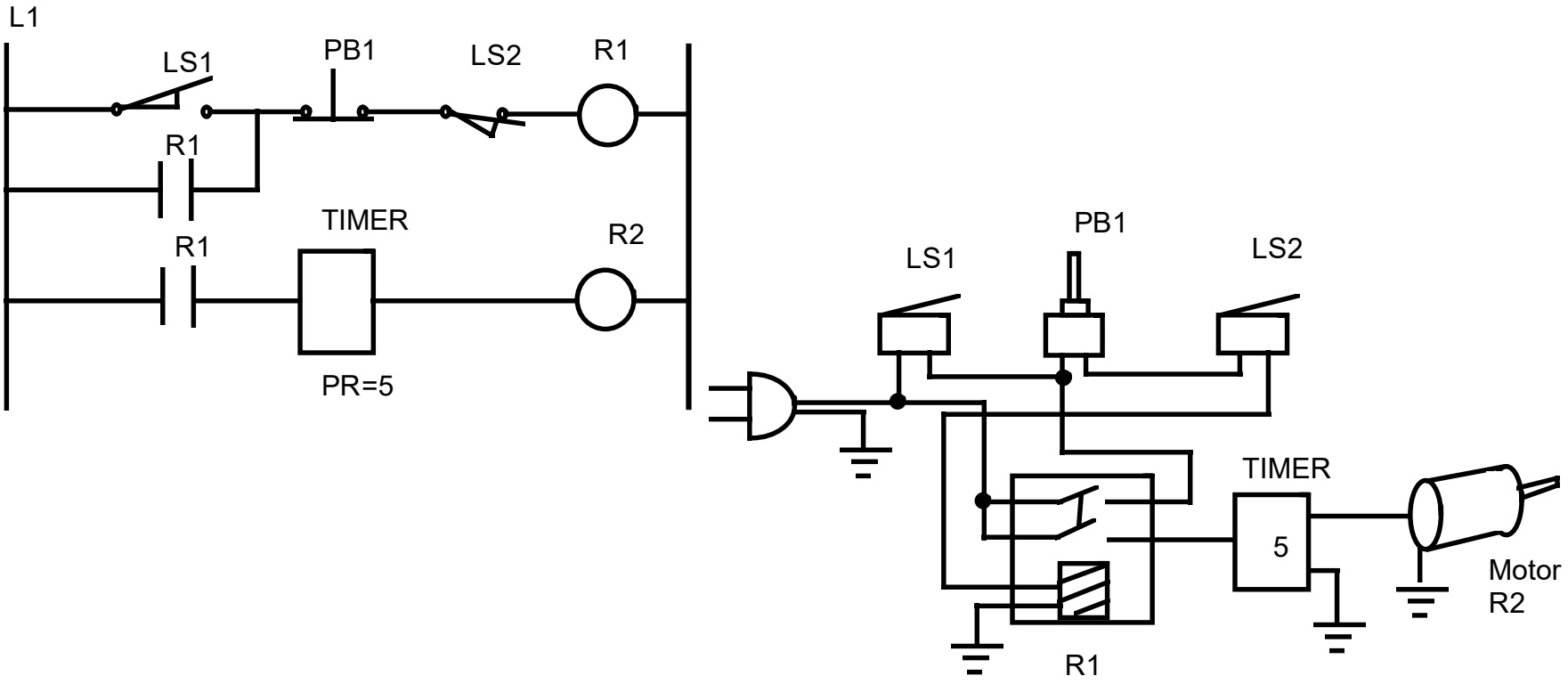


خودکار کردن ماشینها بر اساس منطق رله‌ای:



مثالی از یک سیستم رله‌ای

پروسه کنترل به منظور تنظیم و کنترل روشن شدن یک موتور است. موتور ۵ ثانیه پس از رسیدن قطعه کلید **LS1** روشن می شود و با رسیدن قطعه به **LS2** خاموش می شود. یک سوئیچ اضطراری نیز برای قطع پروسه در نظرگرفته شده است.



کنترل کننده منطقی قابل برنامه ریزی (PLC)

مقایسه با سیستم رله ای:

۱- سیم کشی کمتر

۲- تجهیزات کمتر

۳- فضای کمتر

۴- عیب یابی ساده تر

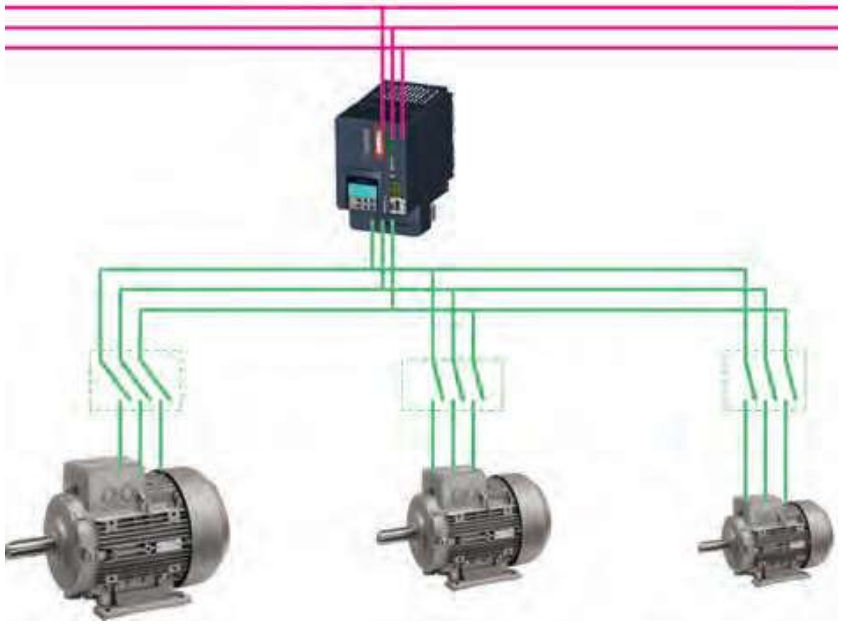
۵- عدم نیاز به سرویس و نگهداری

۶- طراحی ساده تر

۷- امکان اجرای محاسبات پیچیده تر

۸- تاخیر کمتر

۹- هزینه کمتر



کنترل کننده منطقی قابل برنامه ریزی (PLC)

مقایسه با سیستم کنترلی بر پایه کامپیوتر:

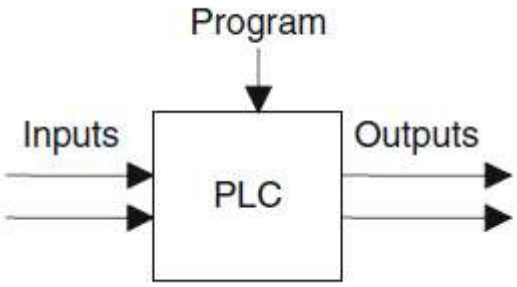
۱- نوشتن برنامه کنترل فرایند با PLC ساده می‌باشد در صورتی‌که برای نوشتن برنامه کنترل توسط کامپیوترهای صنعتی باید با یکی از زبان‌های برنامه‌نویسی کامپیوتر مثل پاسکال (Pascal)، C++ و... انجام شود که در برخی از موارد نیاز به تجربه و تخصص بالا داشته و زمان زیادی جهت برنامه‌نویسی احتیاج دارد.

۲- PLC به گونه‌ای طراحی شده که با تمامی نیازهای کنترل یک سیستم انطباق دارد فقط لازم است هنگام طراحی شرایط را پیش‌بینی کرده و بر اساس نیاز ماژول‌ها را انتخاب و استفاده کنیم ولی در کامپیوتر جهت برقرای ارتباط با سیستم‌های مختلف صنعت لازم است روی آن یک سری ماژول اضافه شود که بررسی و خرید تجهیزات خاص جهت انطباق با سیستم، کاری طاقت فرسا بوده و گاهی اوقات غیر ممکن است.

تاریخچه کنترل کننده منطقی قابل برنامه ریزی (PLC)

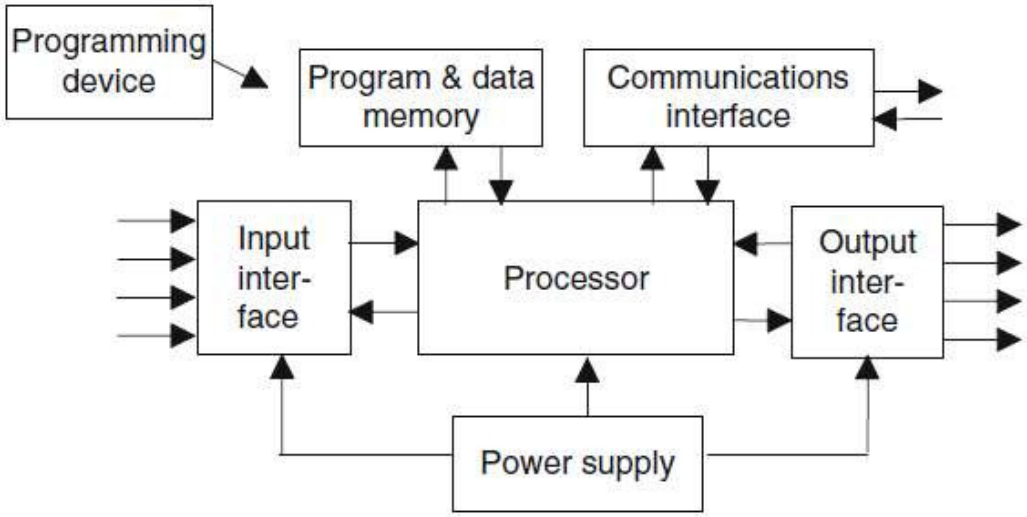
اولین کنترل‌کننده‌های منطقی برنامه‌پذیر در سال ۱۹۶۹ توسط توسط شرکت Modicon به سفارش جنرال‌موتور ساخته شد. در دهه ۷۰ امکان ارتباطات به آن اضافه شد در دهه ۸۰ رابط‌های استاندارد به آن‌ها اضافه شد و بالاخره در اواخر دهه ۸۰ استاندارد زبان‌های برنامه‌نویسی PLC یعنی استاندارد IEC61131 ارائه گردید که شرکت‌ها جهت تطابق آن لازم است نحوه ساخت PLC و زبان‌های برنامه‌نویسی خود را با این استاندارد هماهنگ کنند.

کنترل کننده منطقی قابل برنامه ریزی (PLC)



کنترل کننده منطقی قابل برنامه ریزی (PLC) نوع خاصی از کنترلرهای بر پایه میکروپروسسور است که از حافظه قابل برنامه ریزی برای ذخیره سازی دستورات و توابع اجرایی مانند منطقی، ترتیب دهی، زمانبندی، شمارش برای کنترل پروسه استفاده می کند.

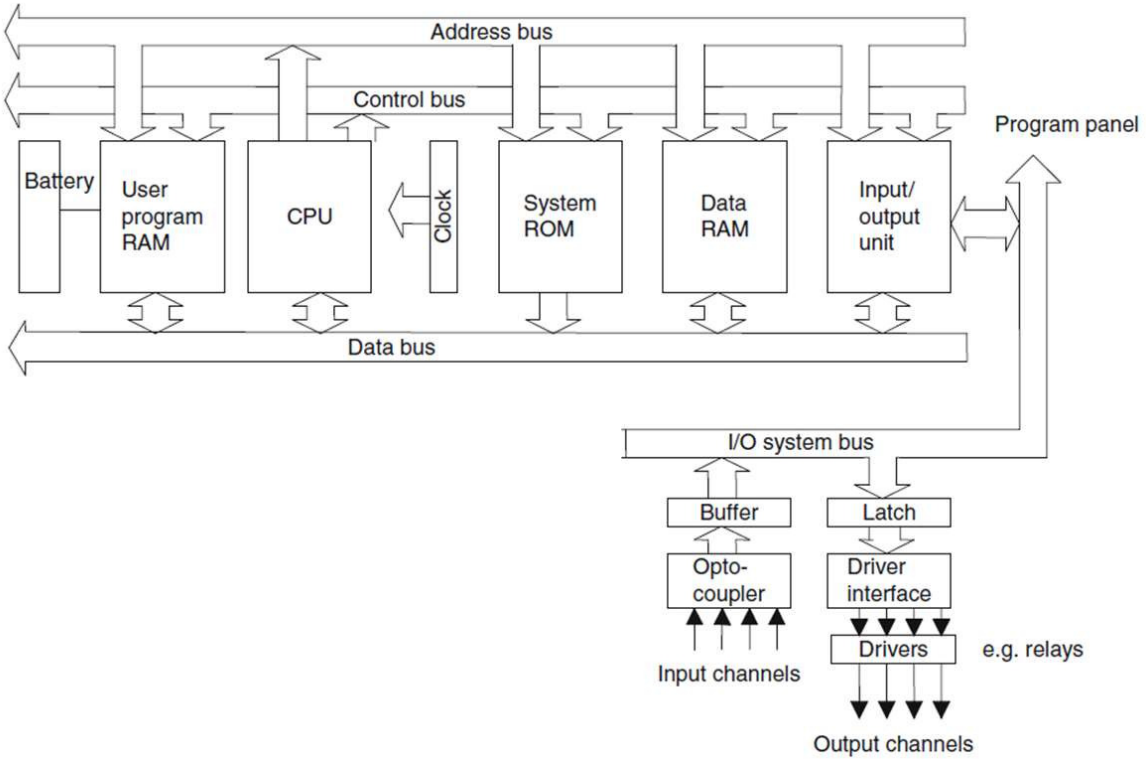
بخشهای مختلف یک PLC



- ✓ پردازشگر (CPU)
- ✓ بخش تغذیه انرژی
- ✓ بخش ارتباط با وسیله برنامه نویسی
- ✓ واحد حافظه
- ✓ ورودی و خروجی (دیجیتال و آنالوگ)
- ✓ ارتباط با سایر PLC ها

بخشهای مختلف یک PLC

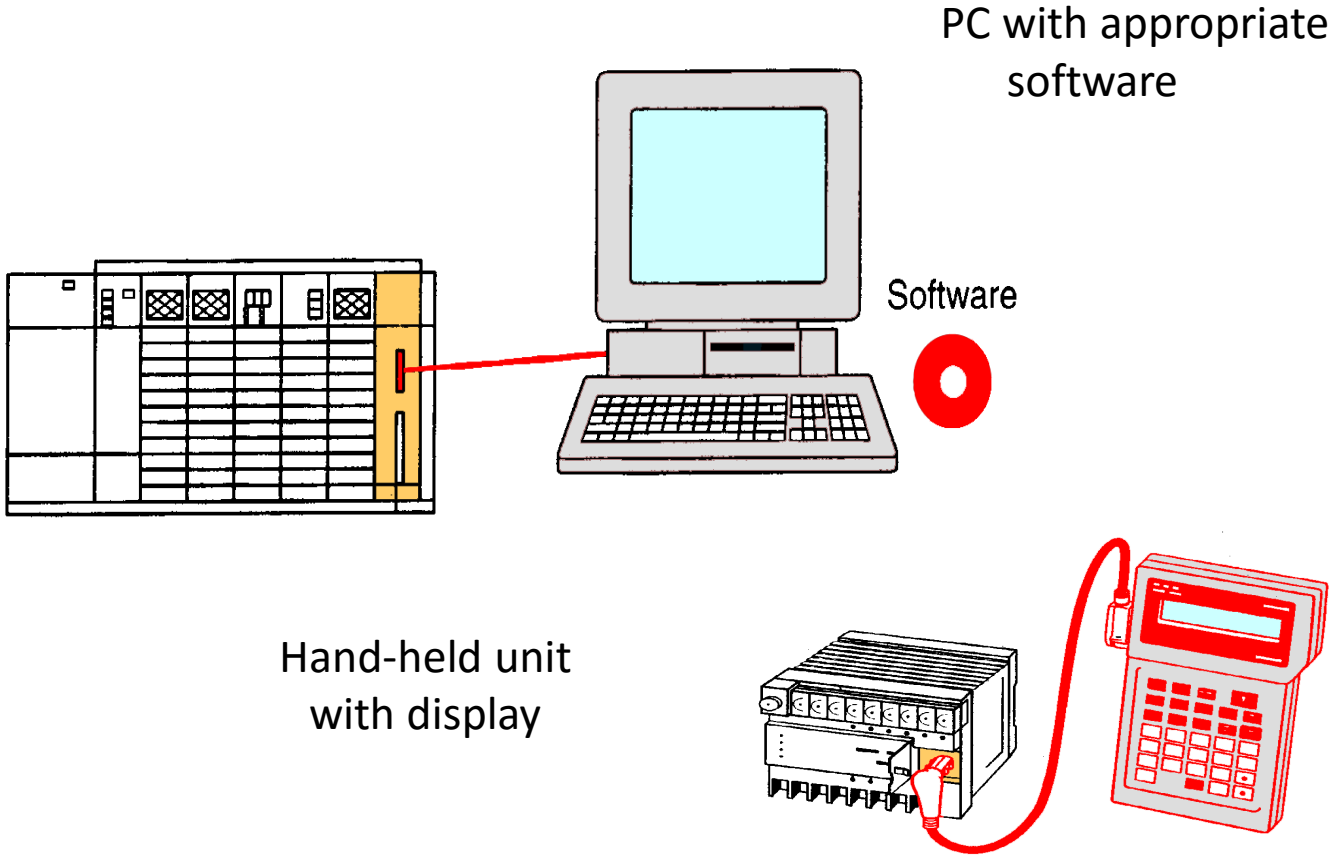
۱- پردازنده (**CPU**) : **قسمت مرکزی پردازنده شامل یک میکروپرسسور است و بر اساس سیگنالهای وارده عملیات کنترلی متناسب با برنامه ذخیره شده در حافظه را انجام می دهد. علاوه بر این CPU وظیفه مدیریت باس های ارتباطی که با ورودی و خروجی در ارتباط هستند یا باس های حافظه را نیز برعهده دارد.**



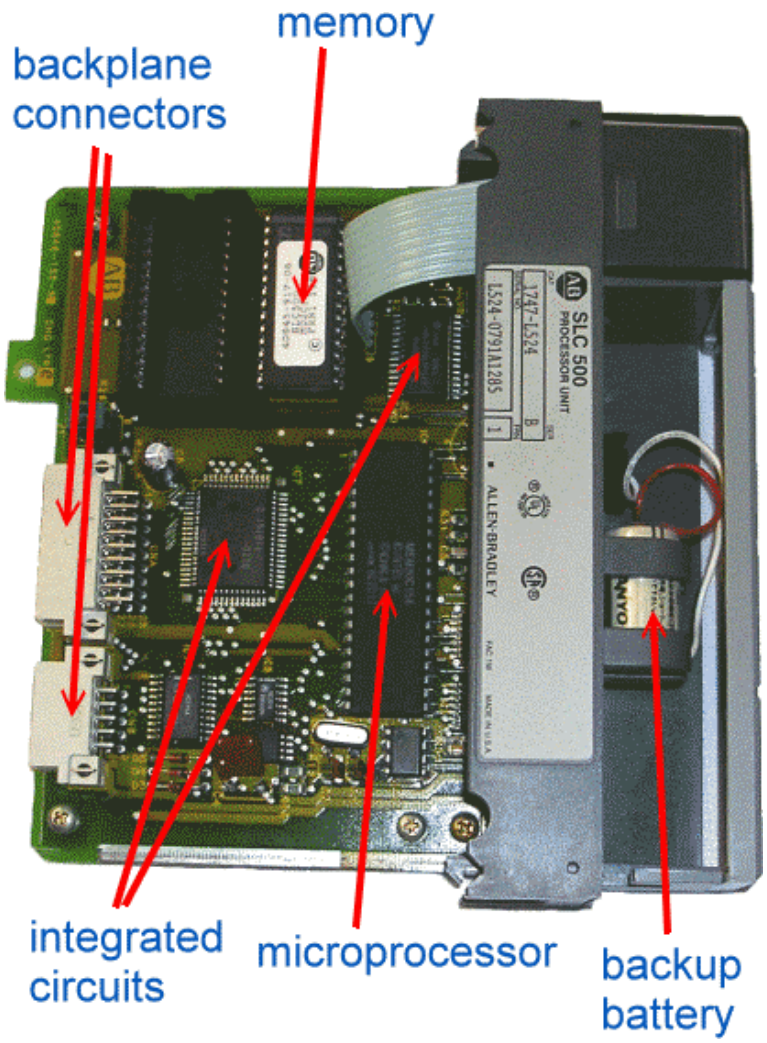
۲- تغذیه انرژی: **این واحد وظیفه تبدیل ولتاژ AC ورودی به ولتاژ DC (۵ ولت) برای پردازنده و همچنین تامین انرژی سایر اجزاء و مدارهای موجود را برعهده دارد.**

بخشهای مختلف یک PLC

۳- بخش ارتباط با وسیله برنامه نویسی: برای وارد نمودن برنامه مورد نیاز به حافظه پردازنده استفاده می‌شود. حافظه در این واحد توسعه داده شده و سپس به حافظه PLC منتقل می شود.



بخشهای مختلف یک PLC

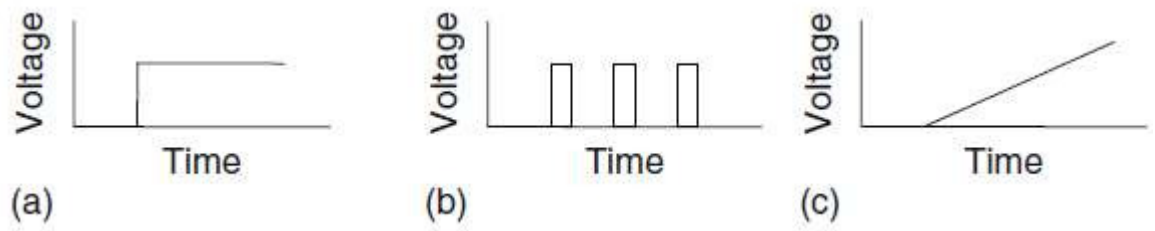


۴-واحد حافظه:

محل ذخیره برنامه کنترلی و داده های ورودی و خروجی

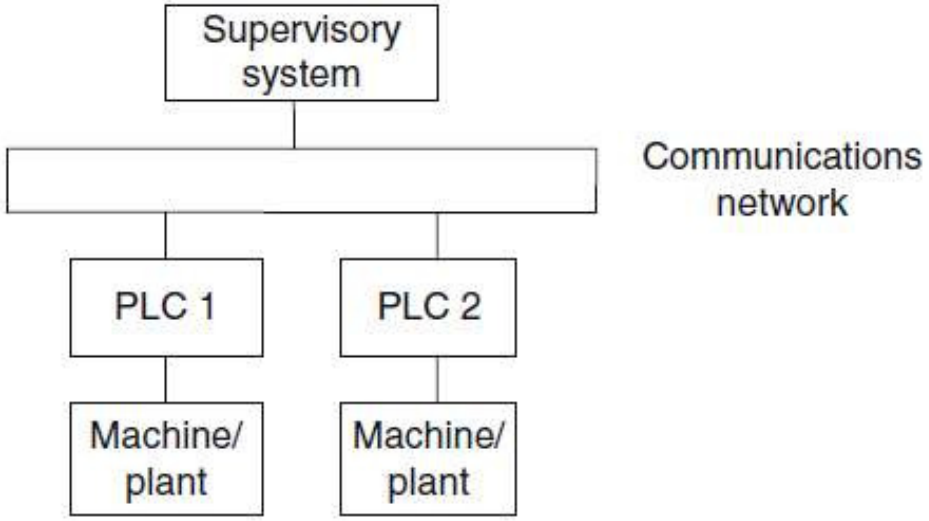
۵- ورودی و خروجی:

ارتباط پردازنده با دنیای بیرون و دریافت داده ها و ارسال اطلاعات. این ورودی ها می تواند از نوع سوئیچها، شمارشگرها و یا سنسورها باشد. خروجی ها می توانند شامل سیم پیچهای راه انداز موتور و یا شیرهای سولنوئیدی باشند.



۵- ارتباط با سایر PLCها:

ارتباط و انتقال و دریافت داده ها از طریق یک شبکه ارتباطی با سایر PLCها صورت می پذیرد.



نحوه اتصال انواع ورودی و خروجی به PLC

۱- اتصال PLC به منبع تغذیه:

PLCها توانایی کار با ولتاژهای ۲۴ ولت **DC** و ۱۲۰ و ۲۴۰ ولت **AC** را بسته به نوع مدل دارند.

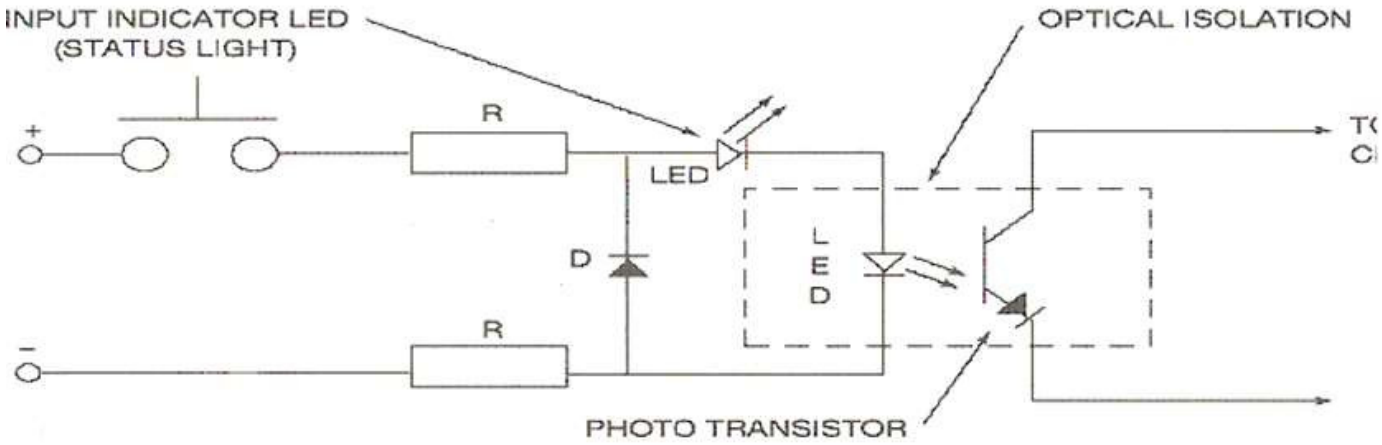
توجه به اتصال قطبهای مثبت و منفی در اتصال به ولتاژ **DC**

بهترین روش برای جلوگیری از آسیب رسیدن به **PLC** در اثر ولتاژ ورودی استفاده از چند فیوز است تا از عبور جریان بیش از حد جلوگیری کند.

نحوه اتصال انواع ورودی و خروجی به PLC

۲- اتصال انواع ورودی به PLC:

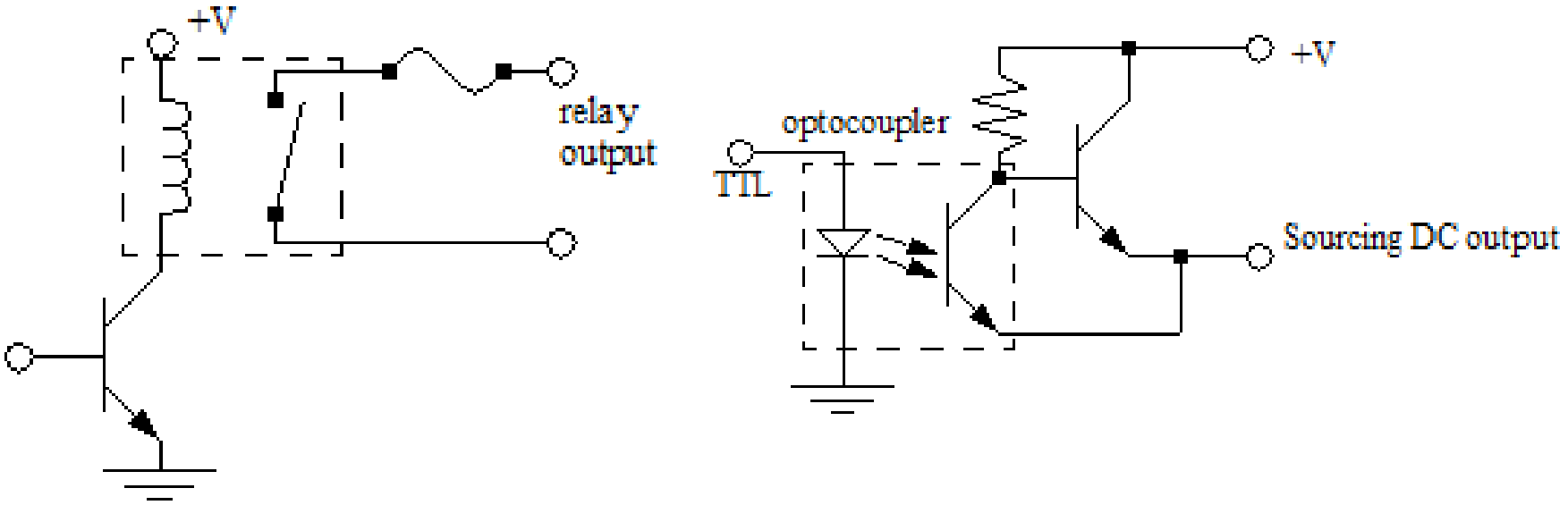
ورودی اکثر **PLC**های جدید به صورت اپتوکوپلر (optocoupler) می باشد.
در این نوع ورودی، ورودی شامل یک **LED** و یک فوتوترانزیستور است و به صورت یک ایزوله نوری عمل می کند.



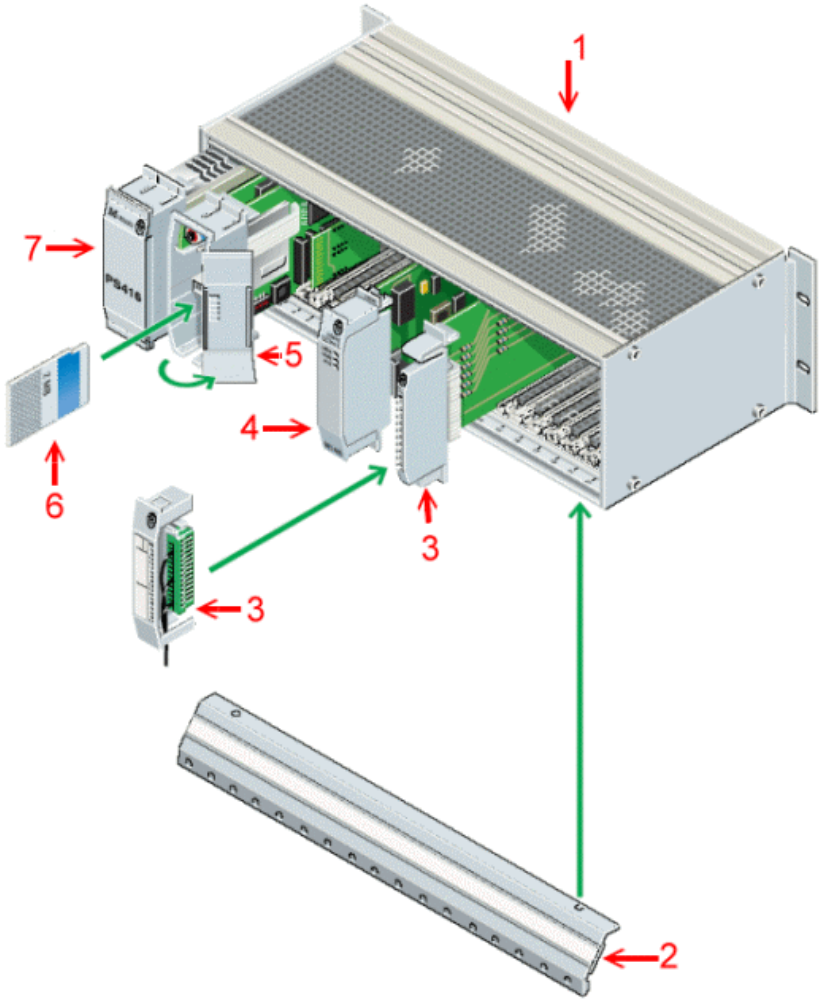
نحوه اتصال انواع ورودی و خروجی به PLC

۳- اتصال انواع خروجی به PLC:

به طور کلی خروجی **PLC** به دو صورت رله ای و ترانزیستوری است.



کنترل کننده منطقی قابل برنامه ریزی (PLC)



اجزای PLC به دو صورت ممکن است ارائه شوند:

۱- ساختار یکپارچه (Compact)

۲- ساختار مجزا و قابل توسعه (Modular)

کنترل کننده منطقی قابل برنامه ریزی (PLC)

شرکت های سازنده PLC:

کنترل کننده منطقی قابل برنامه ریزی (PLC)

معرفی سری های مختلف PLC شرکت زیمنس:

الف) PLCهای سری Simatic S5

ب) PLCهای سری S7

ب۱) Simatic S7 – 200

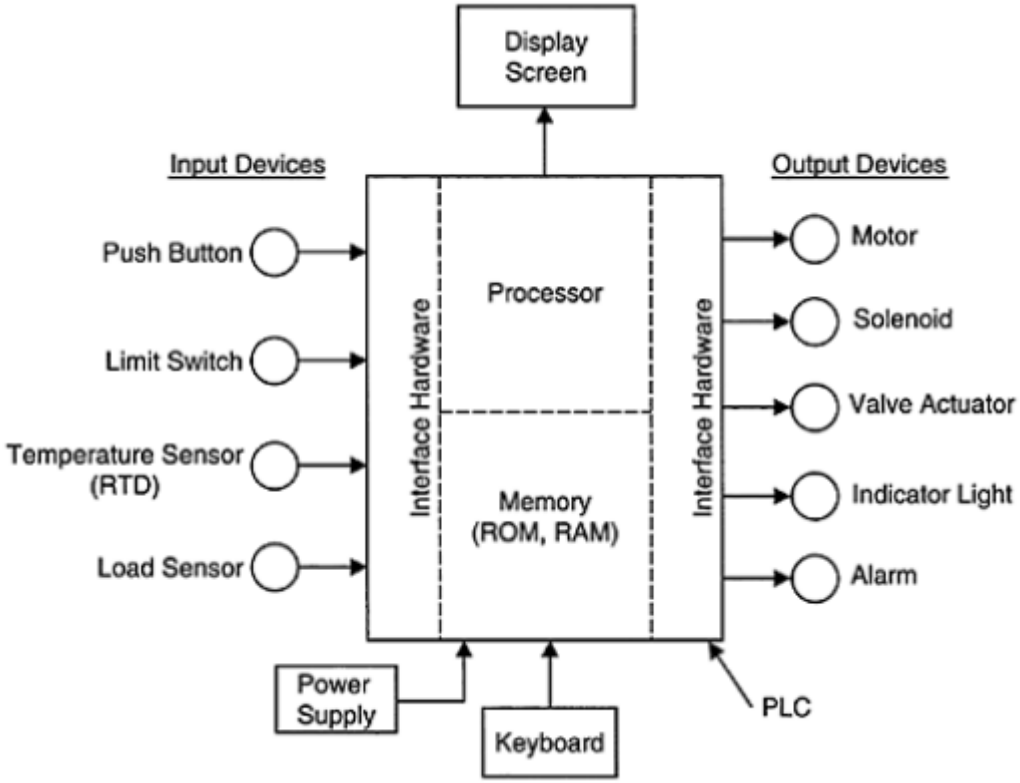
ب۲) Simatic S7 – 1200

ب۳) Simatic S7-300

ب۴) Simatic S7-400

ساختار PLC:

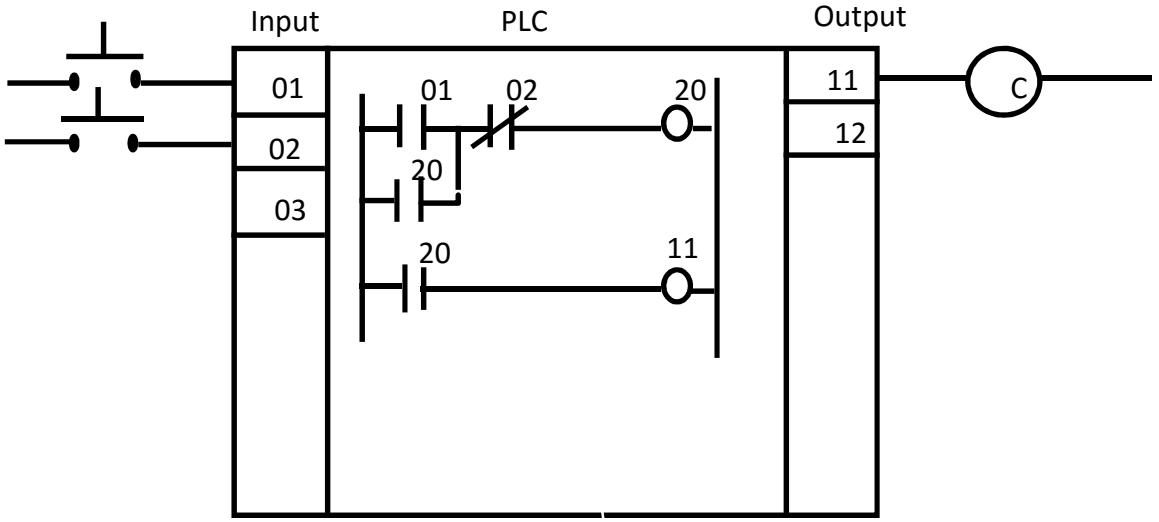
ساختار PLC:



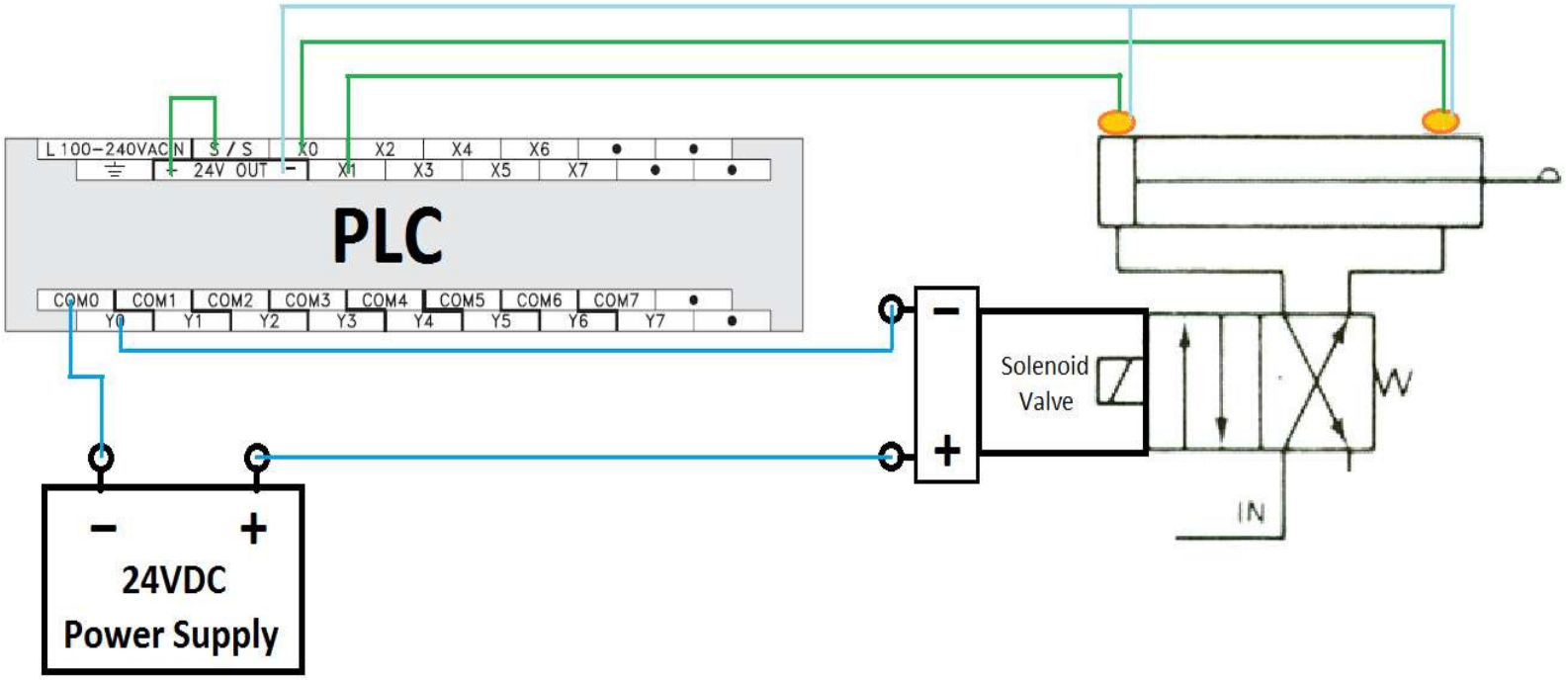
De Silva, C.W., "Mechatronics-An Integrated Approach", Taylor & Francis, CRC Press, 2005

سیم کشی در یک PLC:

سیم کشی در یک PLC:

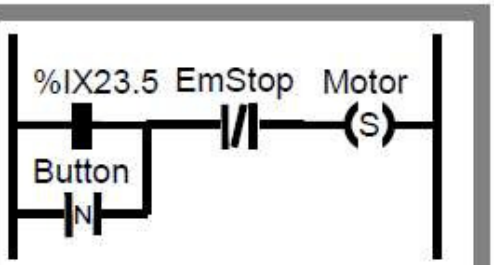


نحوه سیم کشی در PLC

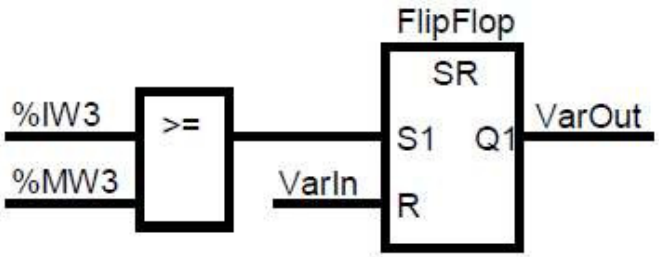


برنامه نویسی PLC

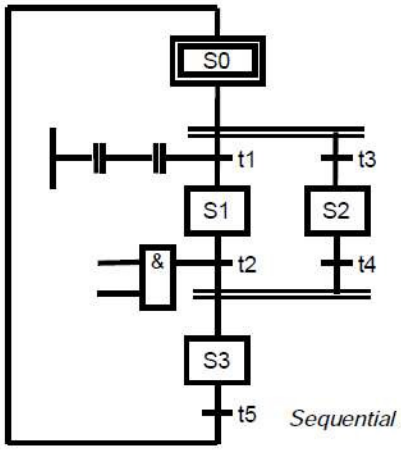
The IEC 61131 defined the standards for PLCs, with 61131-3 defining the programming languages: ladder diagrams (LAD), instruction list (IL), sequential function charts (SFC), structured text (ST), and function block diagrams (FBD).



Ladder Diagram



Function Block Diagram



Sequential Function Chart

```
LD      %IX23.5
ORN     Button
ANDN    EmStop
S       Motor
```

Instruction List

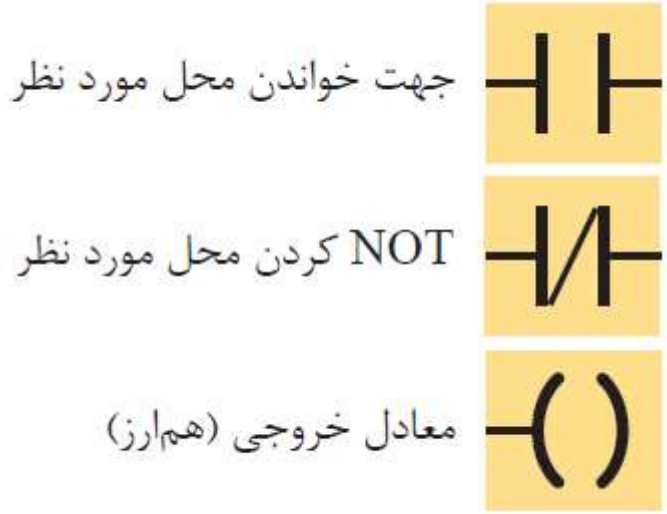
```
FlipFlop (   S1 := (%IW3 >= %MW3),
             R  := VarIn );
VarOut := FlipFlop.Q1;
```

Structured Text

دیاگرام نردبانی (Ladder Diagram):

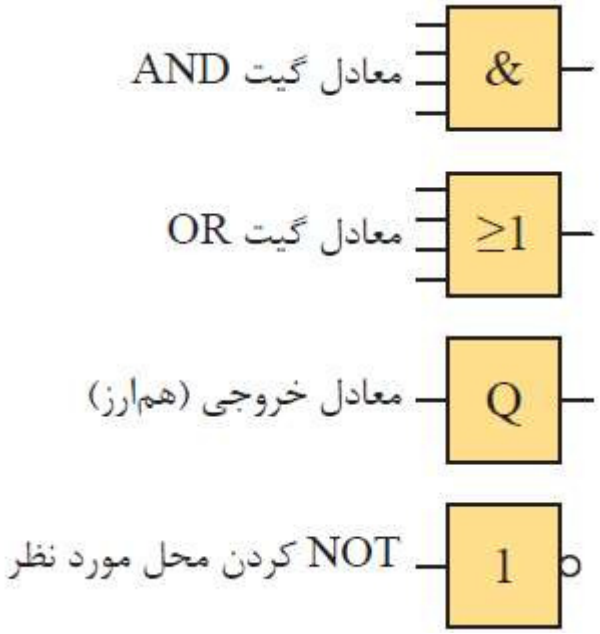
زبان برنامه نویسی گرافیکی است که از دیاگرام مدارهای کنترل رله ای نشات گرفته است.

این زبان به سادگی قابل درک بوده و نقشه ای که بر اساس آن ترسیم می شود مانند نقشه الکتریکی مدار فرمان است.



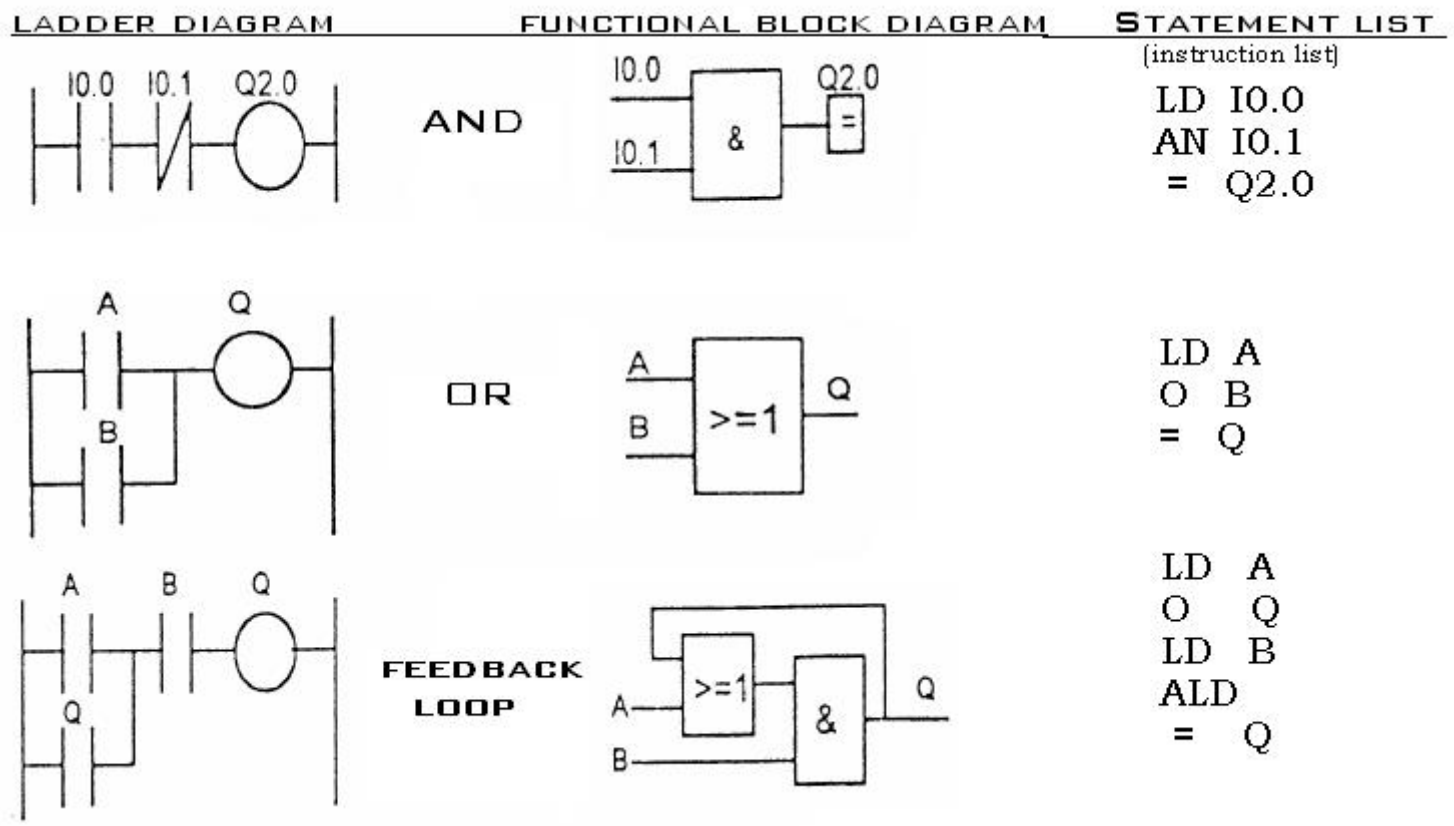
نمایش بلوکی (Function Block Diagram):

این زبان نیز نمایش گرافیکی است که در آن توابع به صورت جعبه ای نمایش داده می شوند.



نمایش دستوری (Instruction List):

این زبان یک زبان متنی است که دستورات آن شباهتهایی با زبان Assembly دارد.



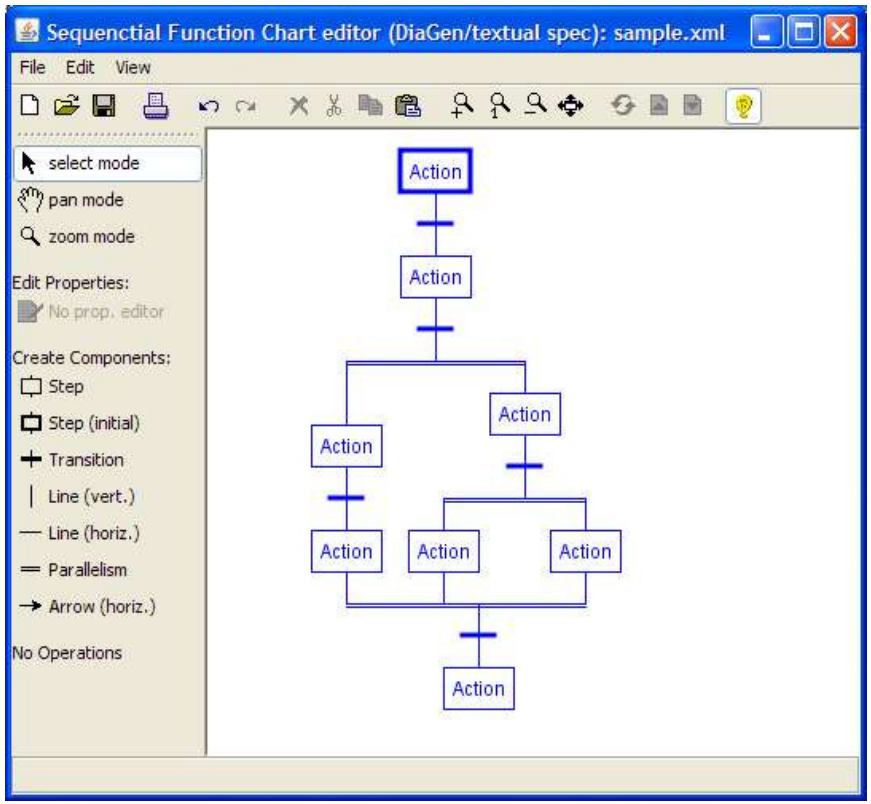
متن ساختاری (Structure Text):

این زبان یک زبان سطح بالا بر اساس زبان Pascal است که شامل تعاریف و دستورالعمل ها می باشد.

```
// PLC configuration
CONFIGURATION DefaultCfg
    VAR_GLOBAL
        b_Start_Stop  : BOOL;          // Global variable to represent a boolean.
        b_ON_OFF      : BOOL;          // Global variable to represent a boolean.
        Start_Stop AT %IX0.0:BOOL;     // Digital input of the PLC (Address 0.0)
        ON_OFF     AT %QX0.0:BOOL;     // Digital output of the PLC (Address 0.0). (Coil)
    END_VAR

    // Schedule the main program to be executed every 20 ms
    TASK Tick(INTERVAL := t#20ms);

    PROGRAM Main WITH Tick : Monitor_Start_Stop;
END_CONFIGURATION

PROGRAM Monitor_Start_Stop          // Actual Program
    VAR_EXTERNAL
        Start_Stop  : BOOL;
        ON_OFF      : BOOL;
    END_VAR
    VAR                             // Temporary variables for logic handling
        ONS_Trig    : BOOL;
        Rising_ONS  : BOOL;
    END_VAR

    // Start of Logic
    // Catch the Rising Edge One Shot of the Start_Stop input
    ONS_Trig    := Start_Stop AND NOT Rising_ONS;
```

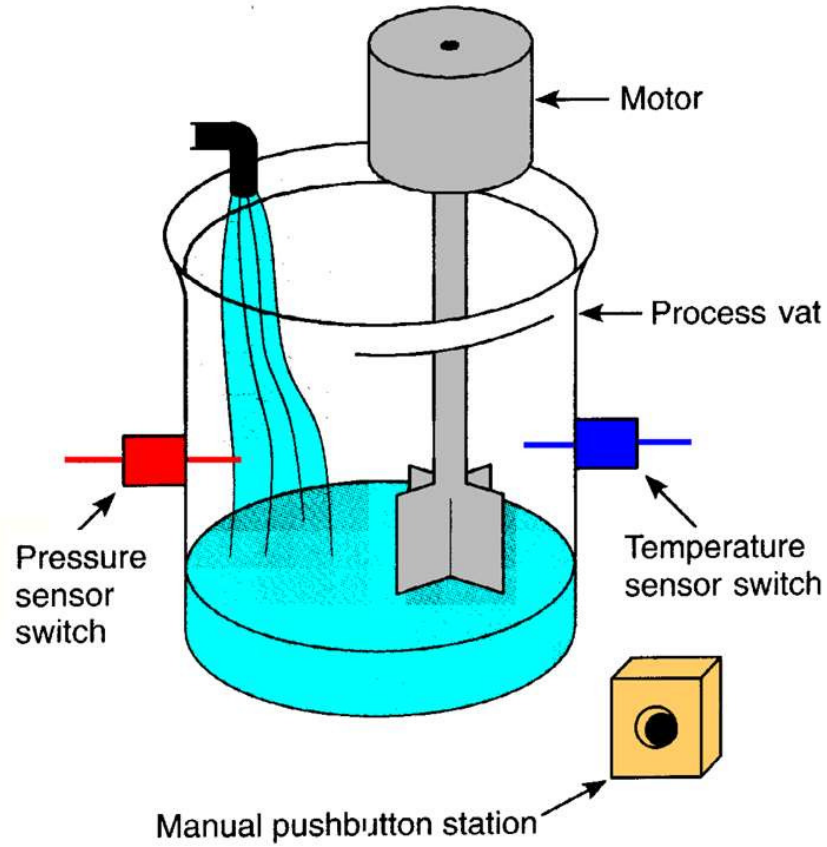
جدول توابع پیاپی (Sequential Function Chart):

این زبان یک روش نمایش تصویری عملکرد سیستم برای بیان ترتیب دستورالعمل هاست.



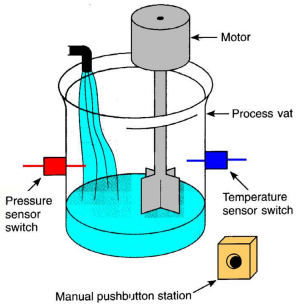
دیاگرام نردبانی (LD)

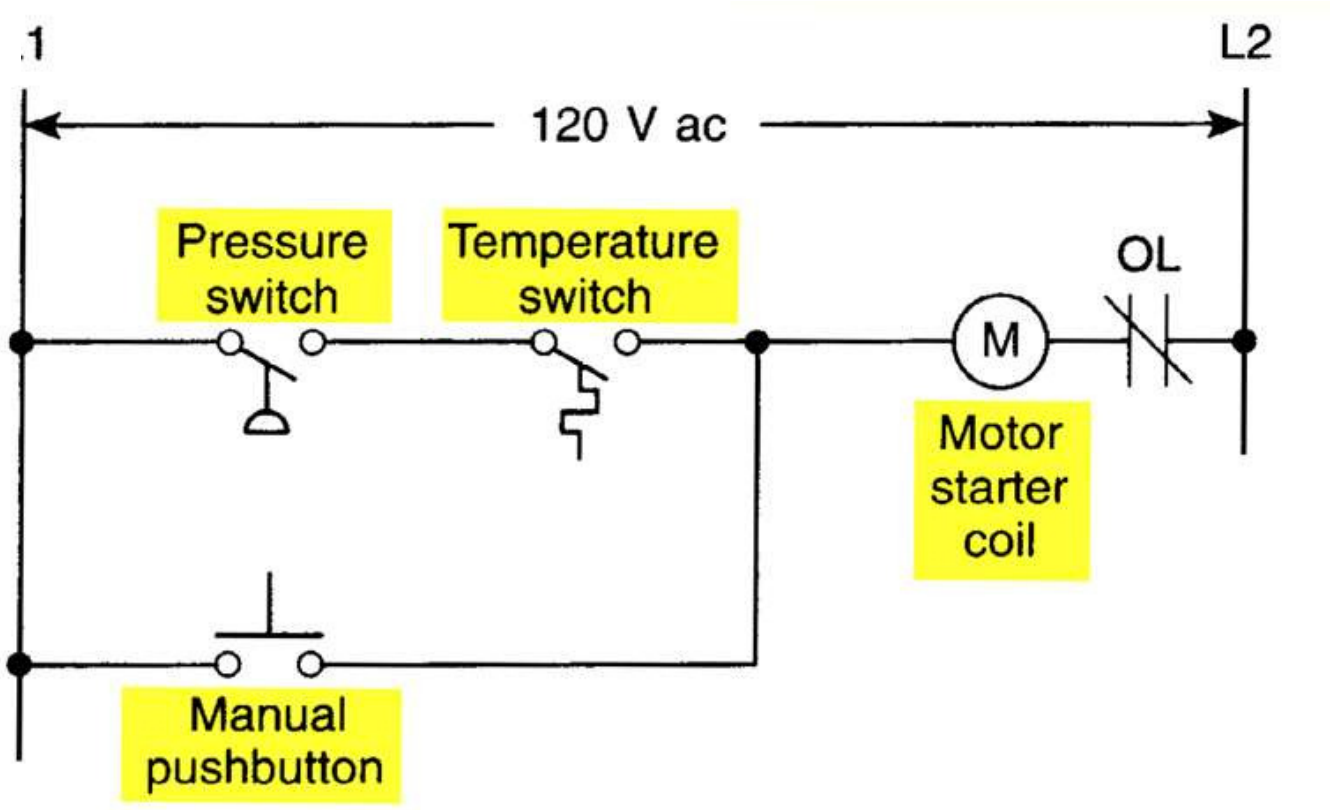
با یک مثال شروع می کنیم:



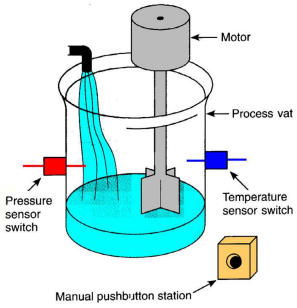
دیاگرام نردبانی (LD)

نمایش سیستم رله ای:

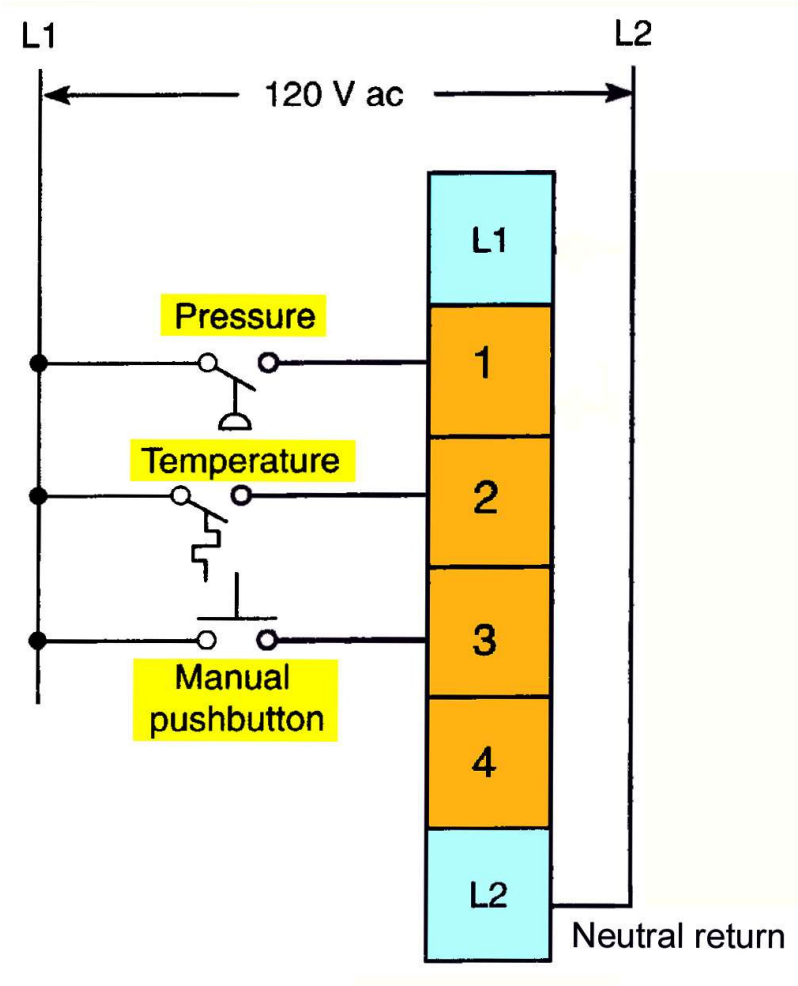




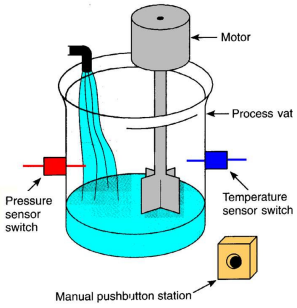
دیاگرام نردبانی (LD)



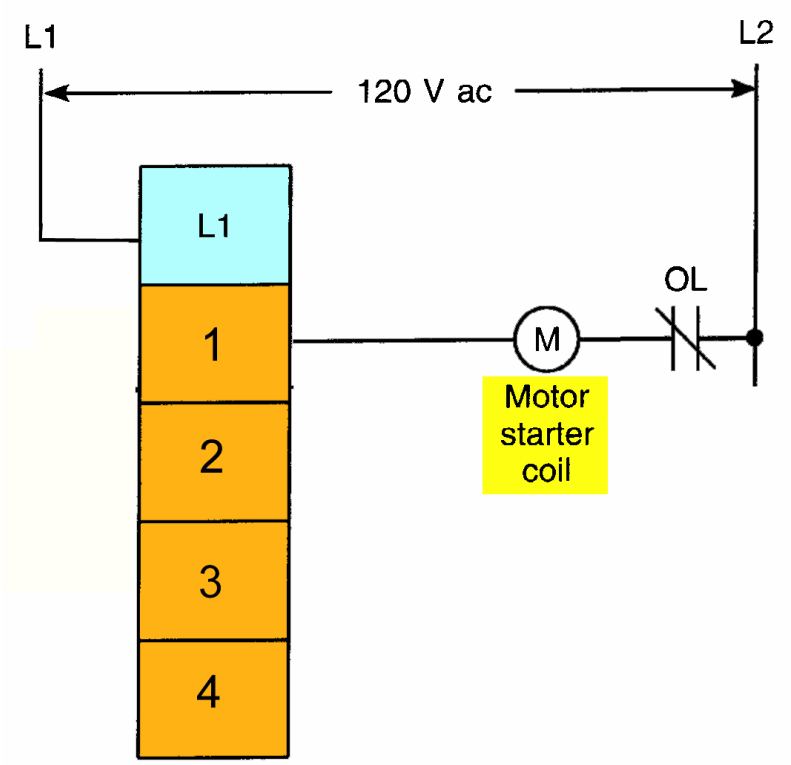
ورودی به PLC:



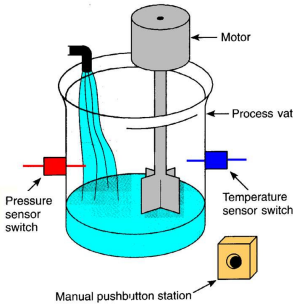
دیاگرام نردبانی (LD)



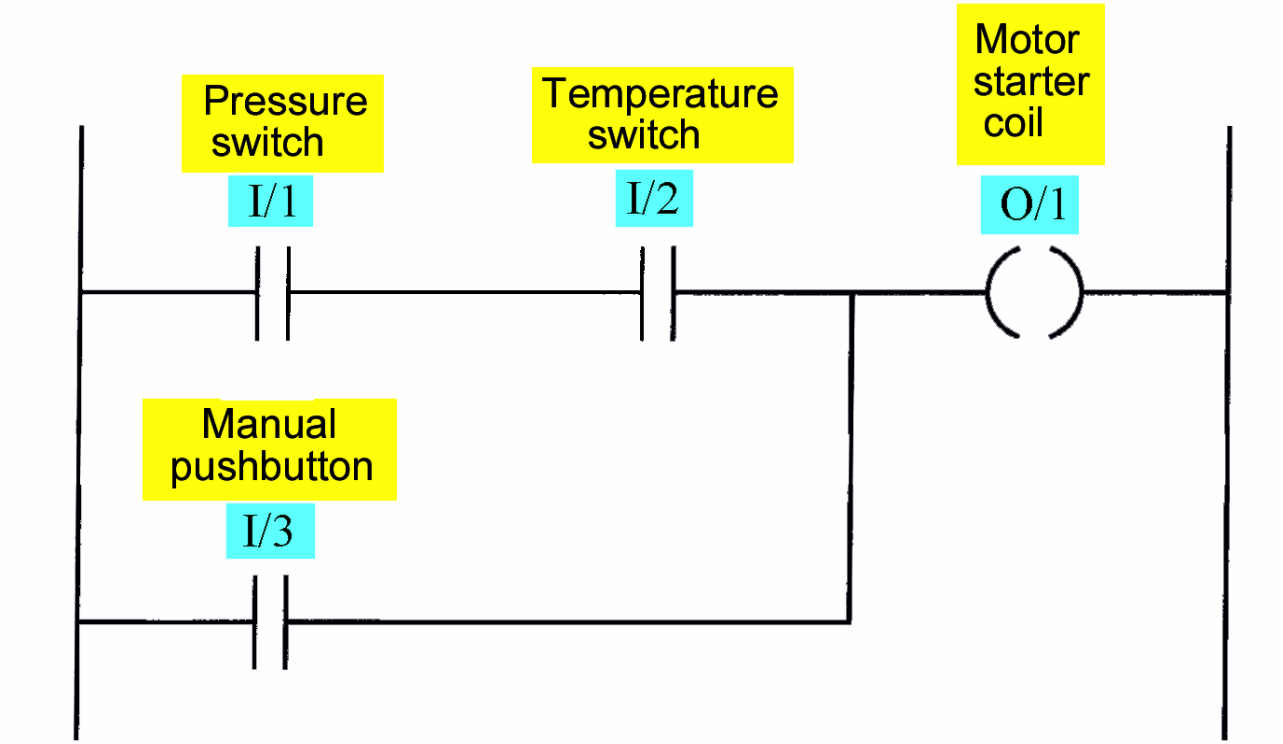
خروجی از PLC:



دیاگرام نردبانی (LD)

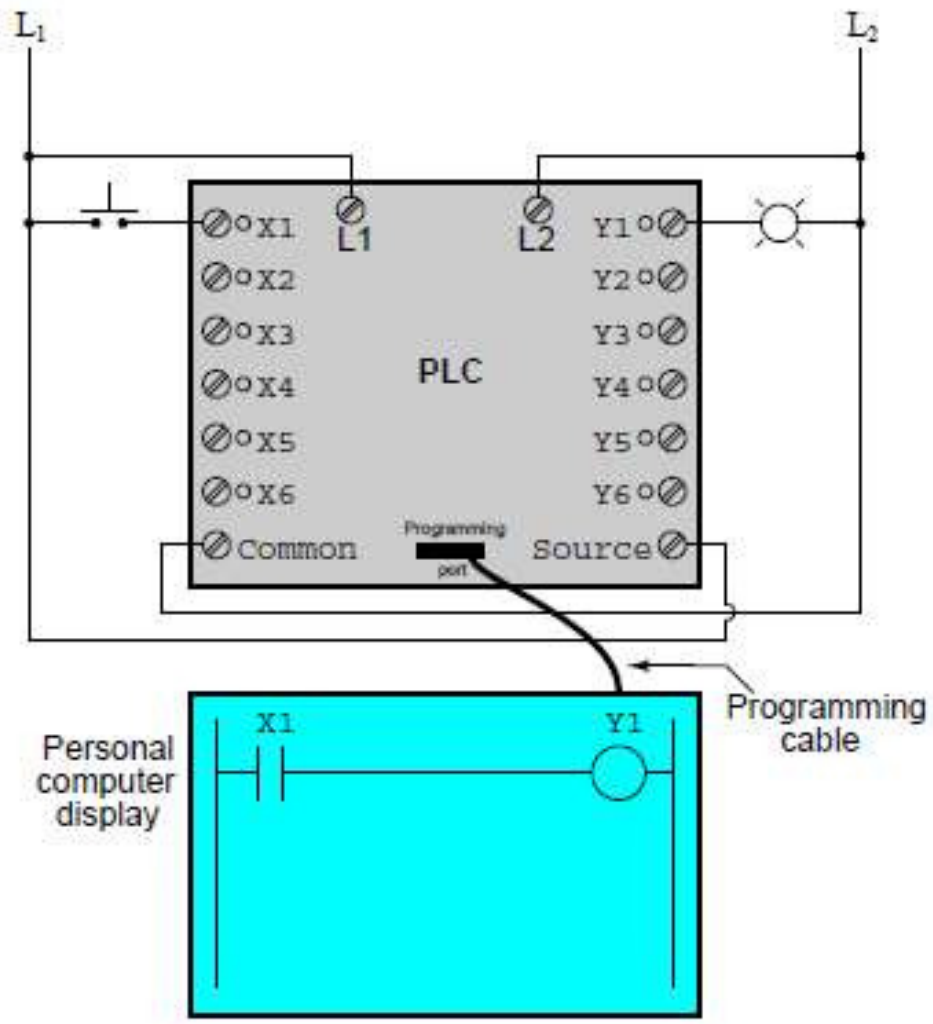


برنامه PLC:



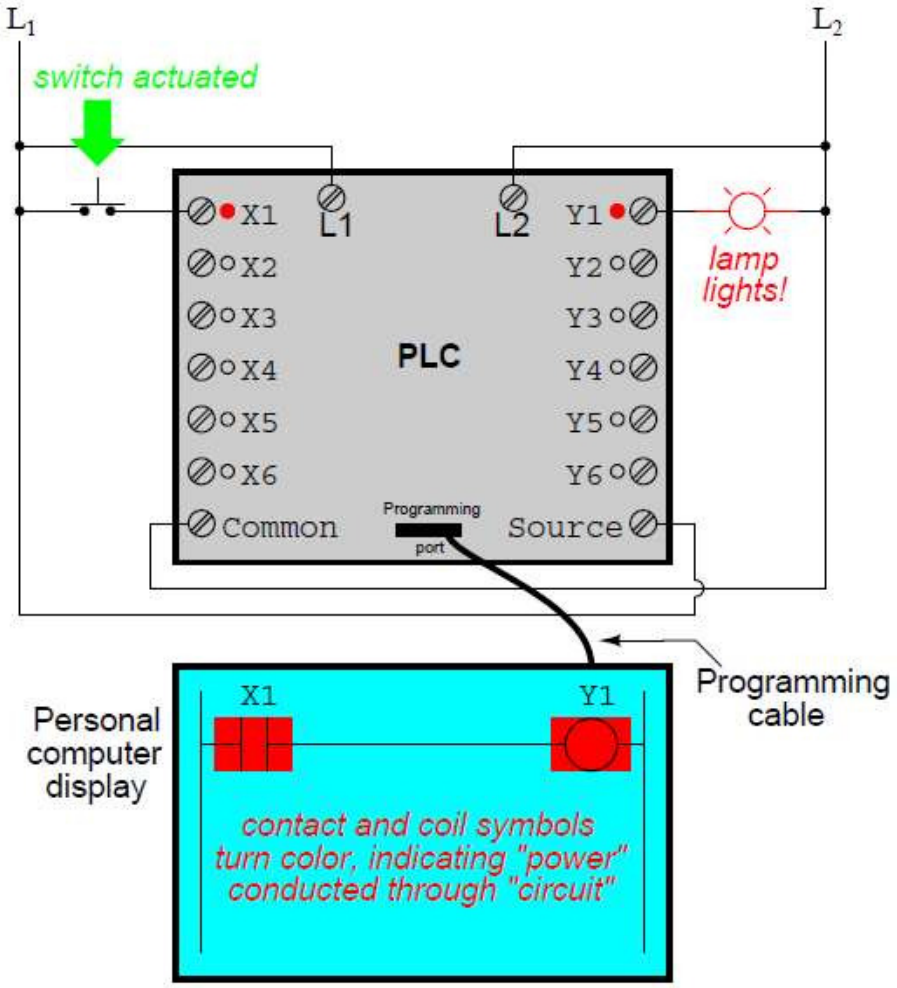
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



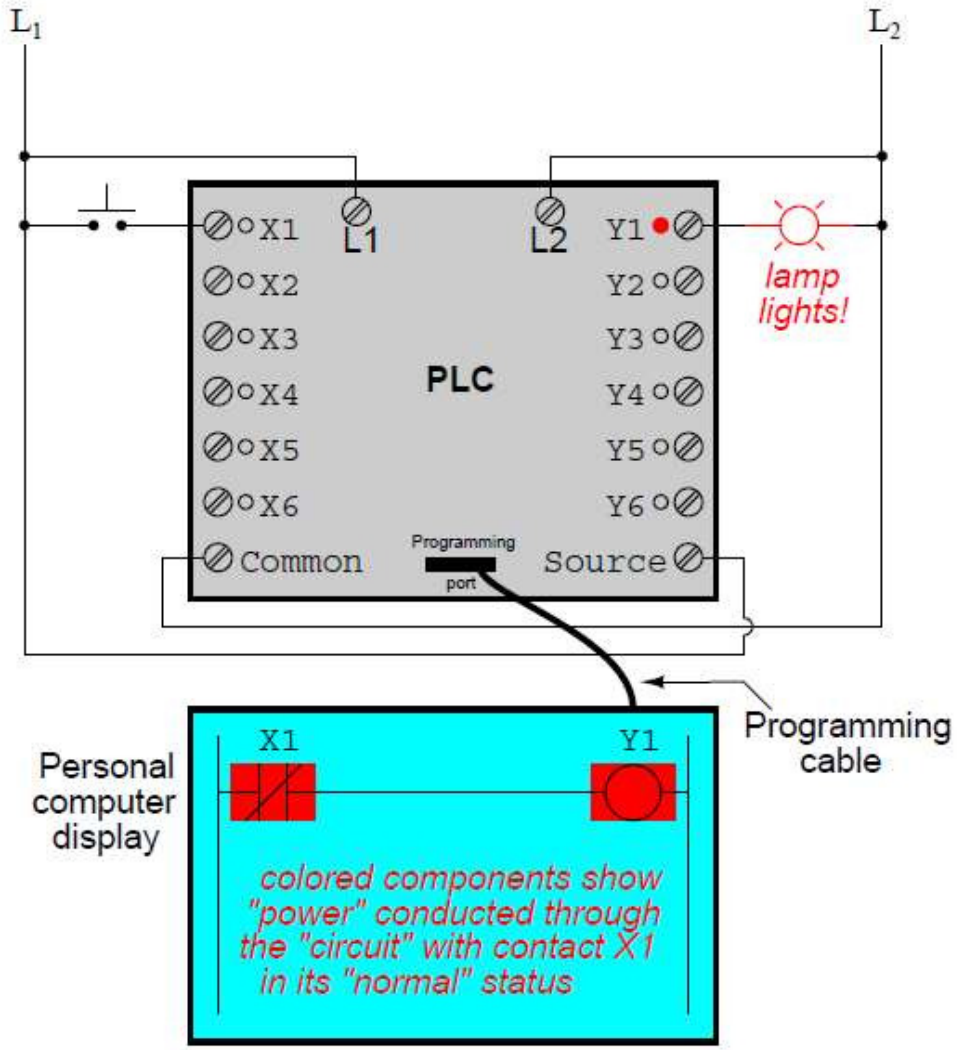
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



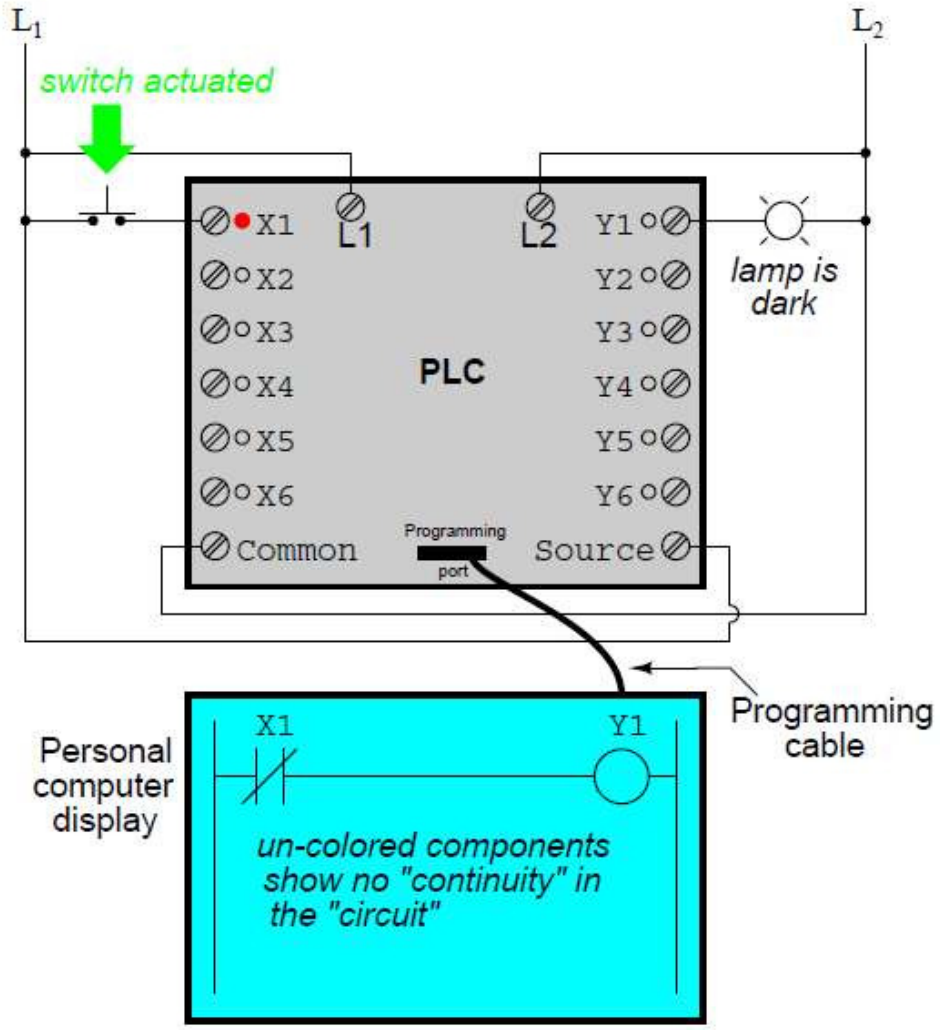
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



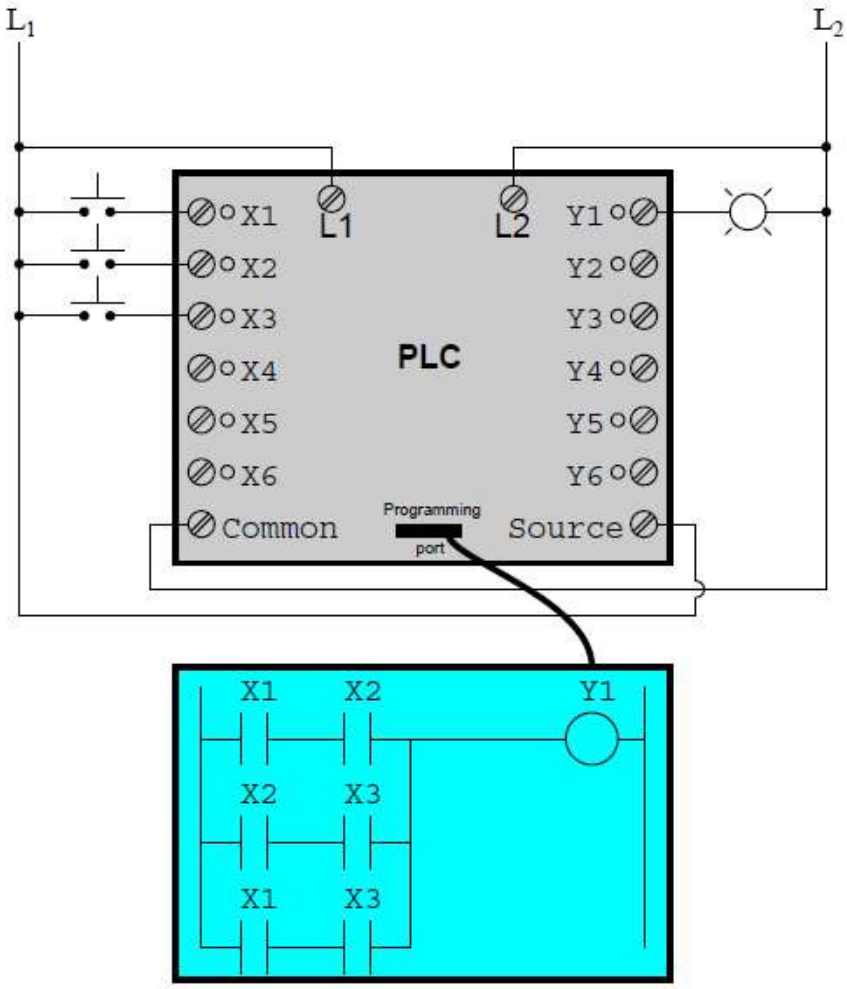
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



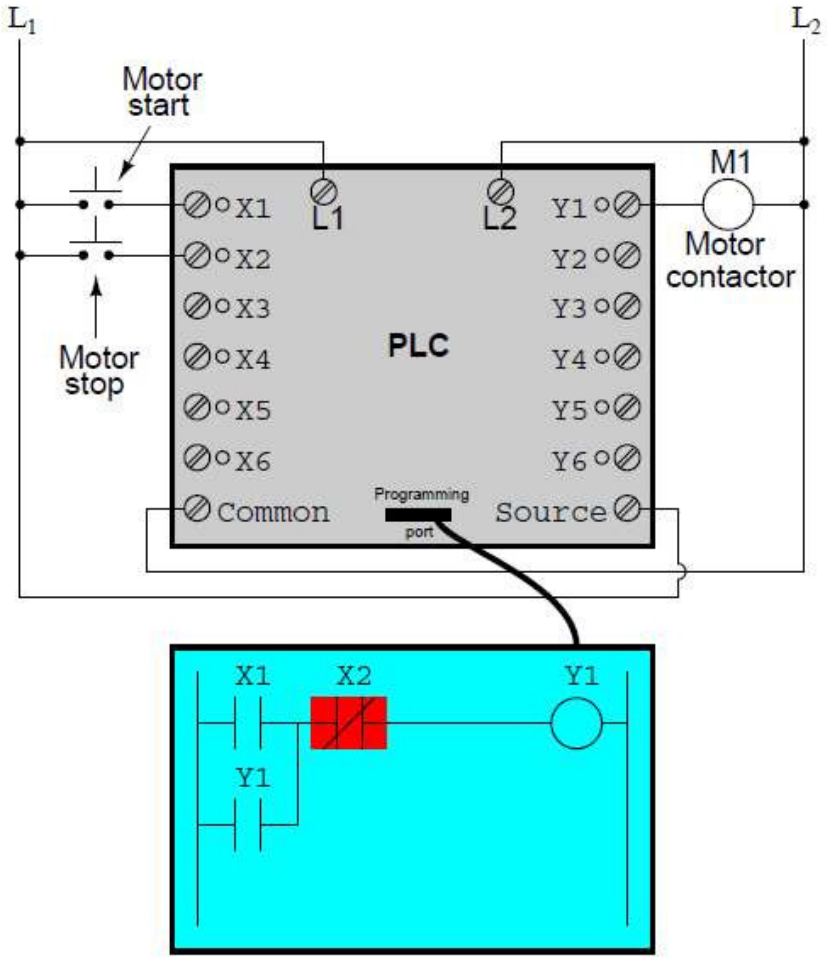
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



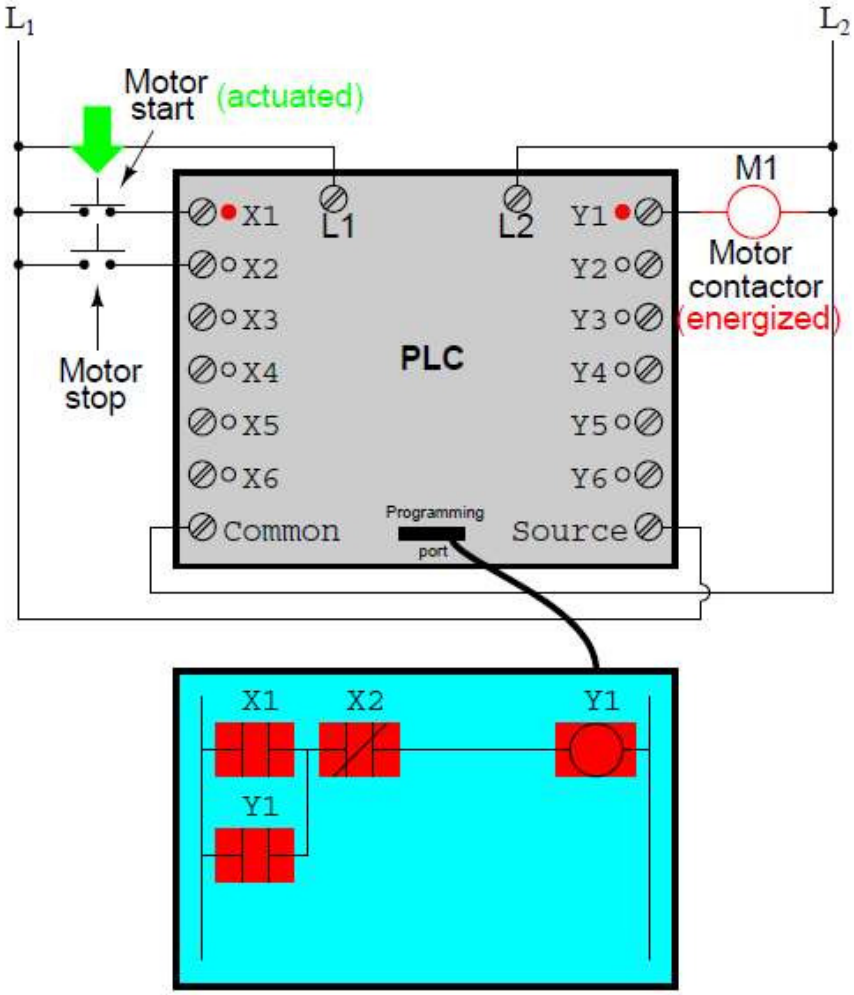
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:

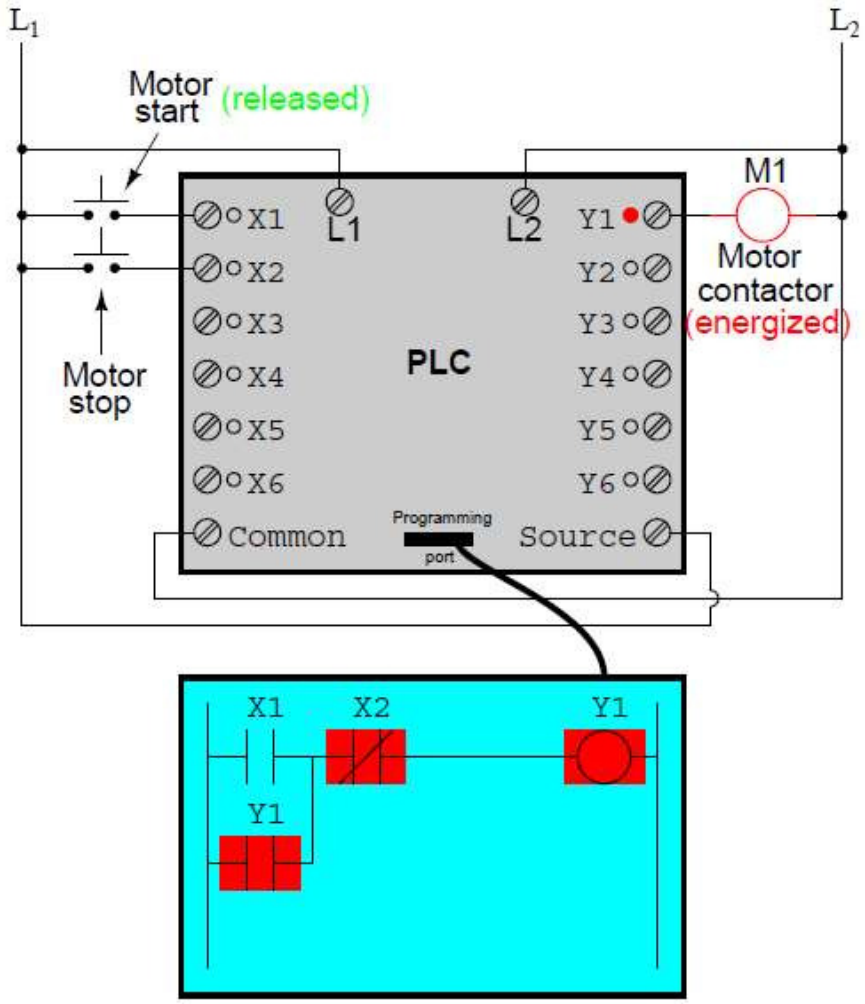


دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

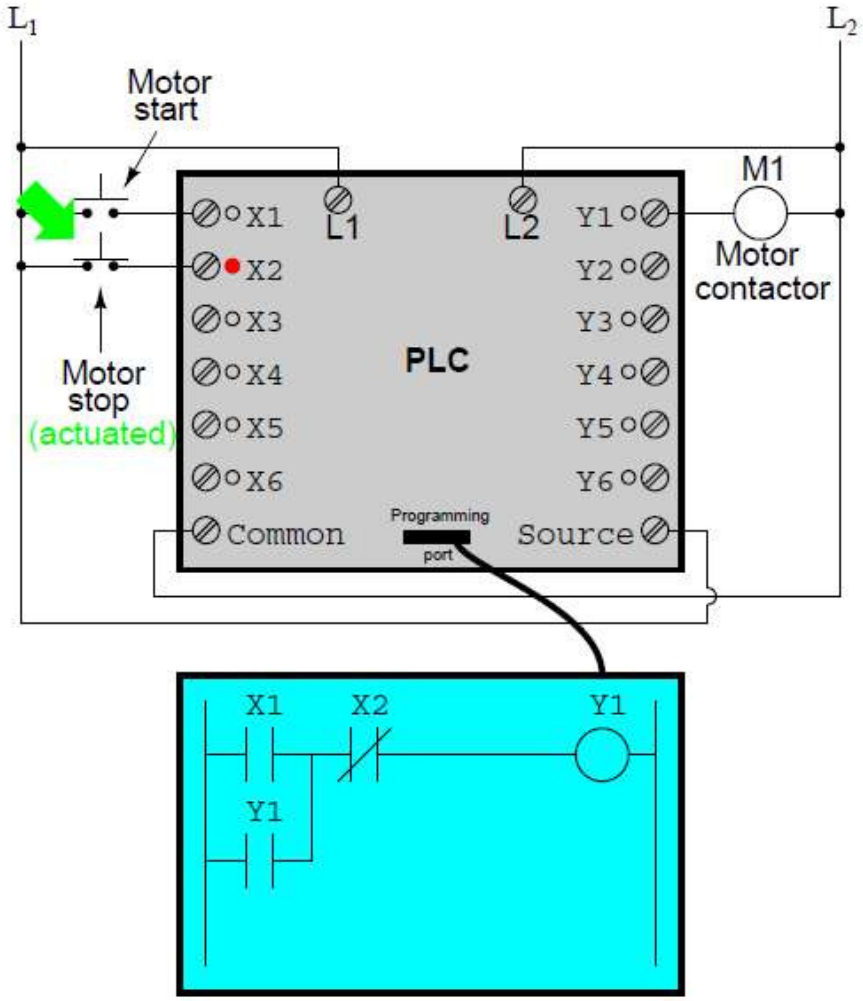


آشنایی با المانهای ساده:

(مدار حافظه، نگهدارنده یا **Latch**)

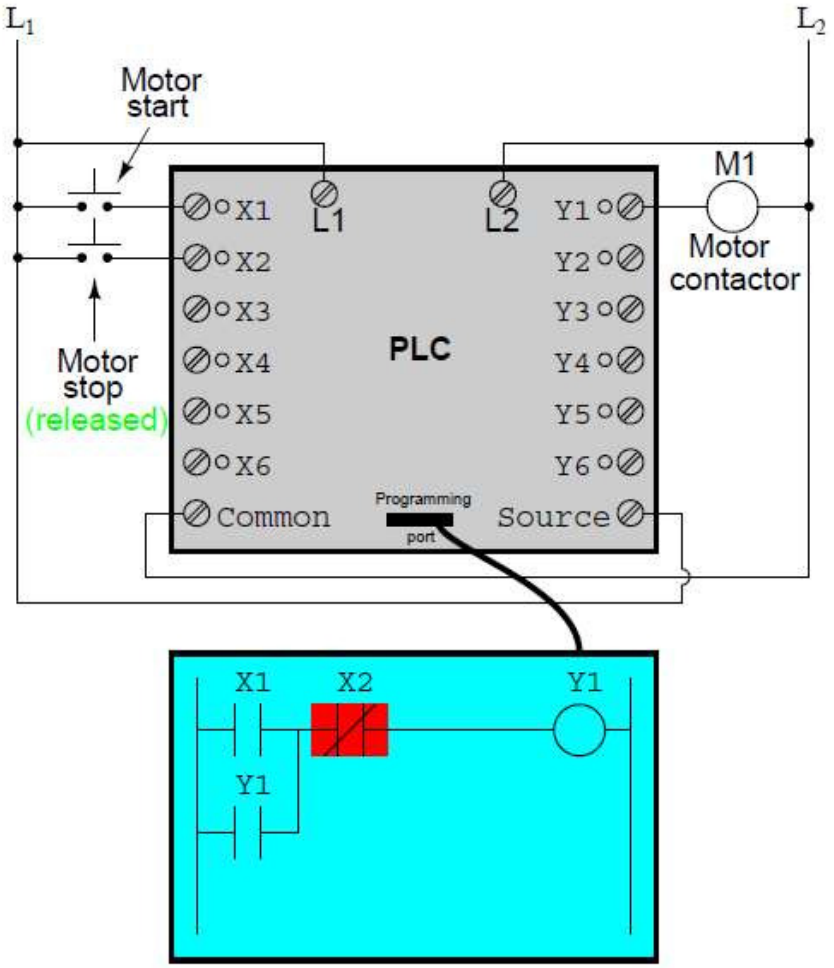
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

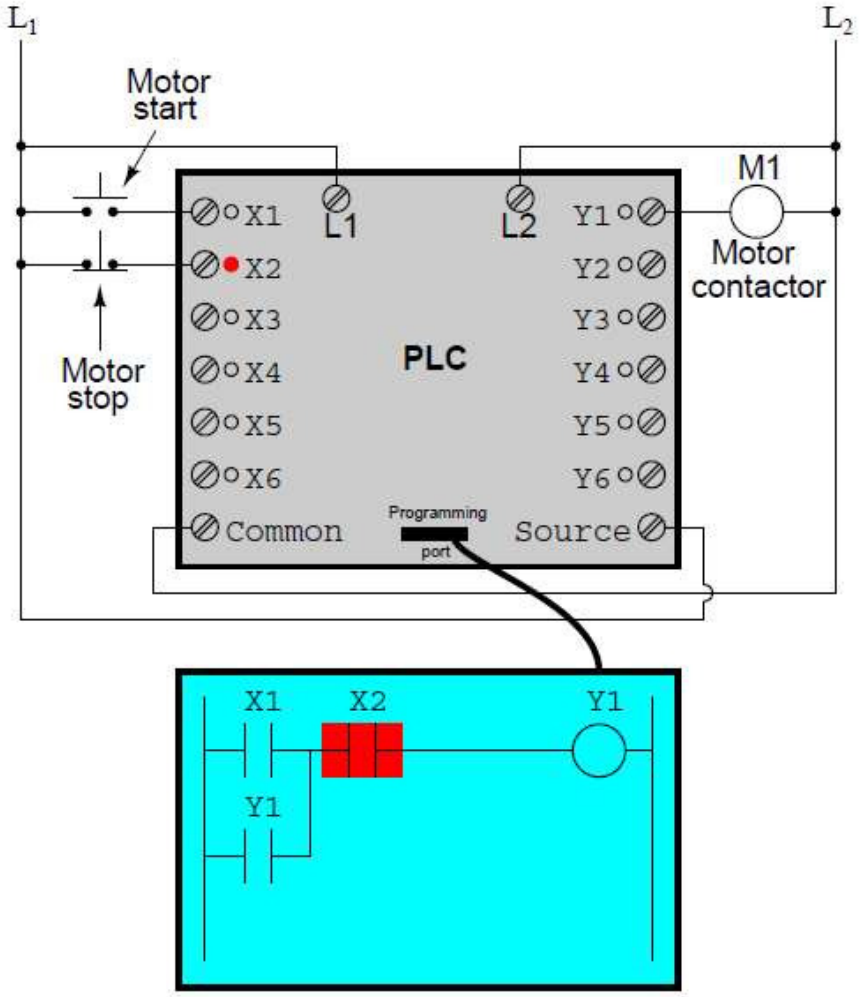
آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

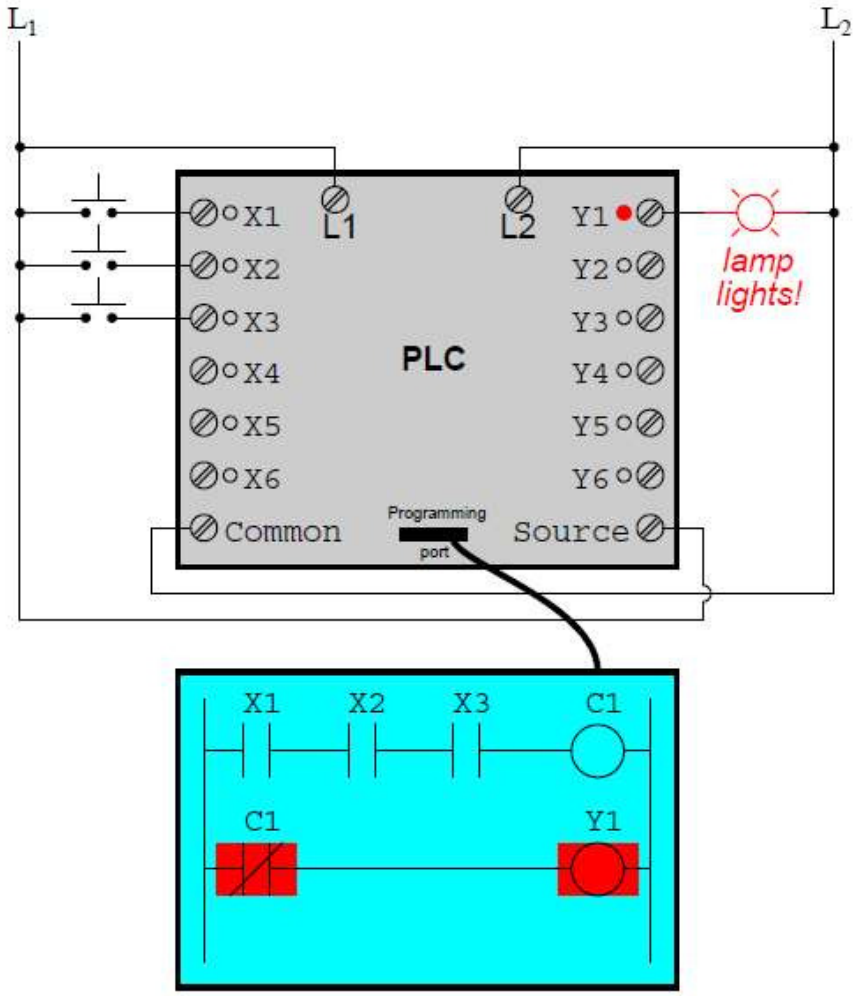
آشنایی با المانهای ساده:

- طراحی ایمن



دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

آشنایی با المانهای ساده: