بسم الله الرحمن الرحیم

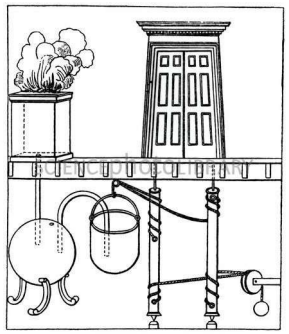
Principles of Mechatronic Systems
مبانی سیستم های مکاترونیکی

آشنایی با سیستمهای کنترلی
معرفی میکروکنترلر
کنترل کننده منطقی قابل برنامه ریزی (PLC)

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.

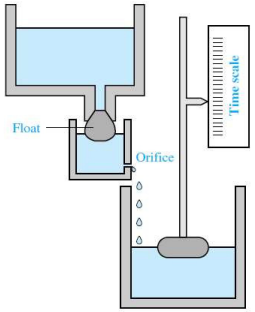


http://www.sciencephoto.com/media/364618/view

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.

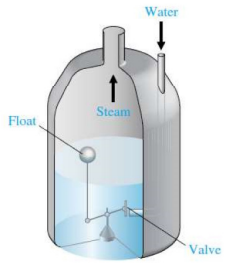
سیستم‌های کنترل

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.

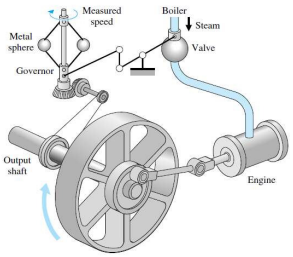


سیستم‌های کنترل

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

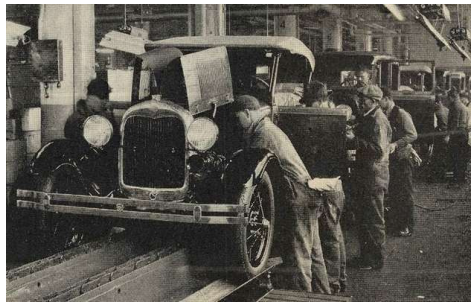


تنظیم کننده سطح



سیستم کنترل سرعت **Flyball governor**

سیستم‌های کنترل



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

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 PLCها 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.

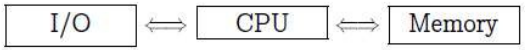
این بخش شامل ریزپردازنده و اجزای الکترونیکی وابسته به آن است.

۲- واسطه های ورودی و خروجی

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

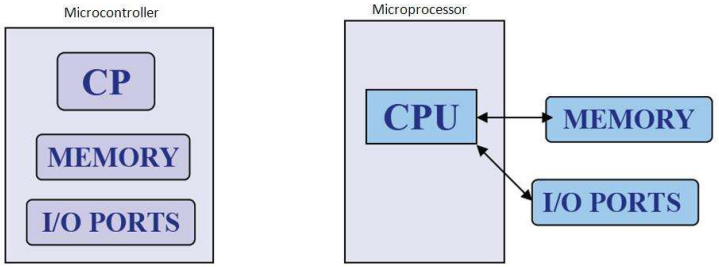
۳- حافظه

برای نگهداری دستورالعمل های برنامه و داده ها



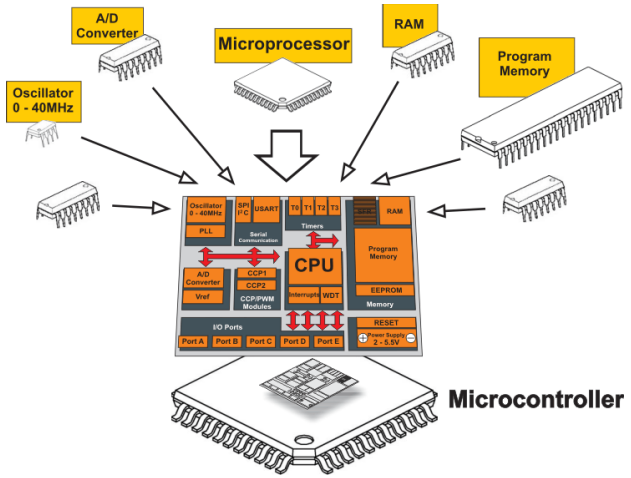
میکروکنترلر

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

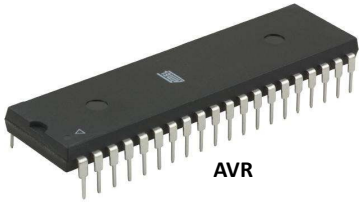
میکروکنترلر

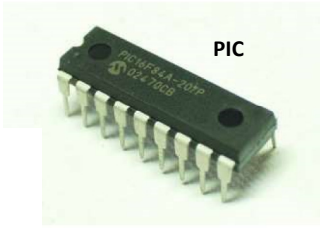


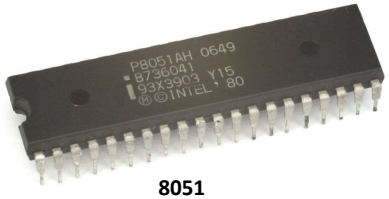
http://www.mikroe.com/chapters/view/1/introduction-world-of-microcontrollers/

میکروکنترلر

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


AVR

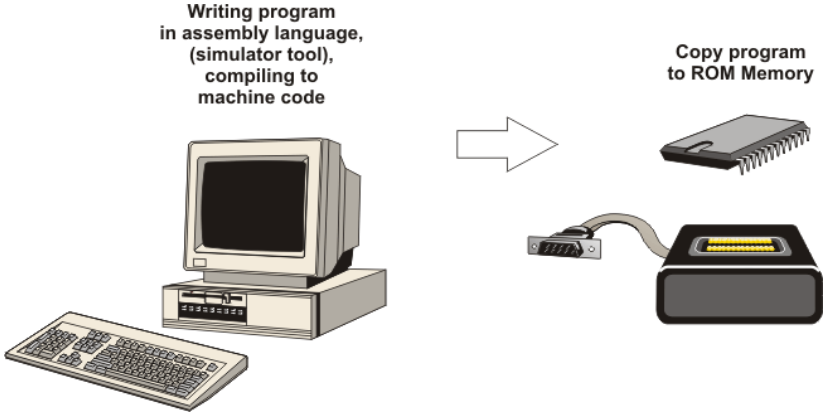

PIC



8051

میکروکنترلر

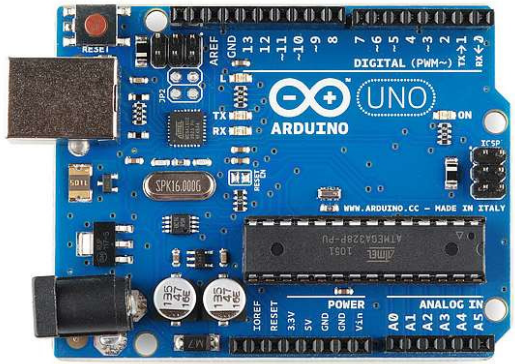
زبان برنامه نویسی اصلی میکروکنترلرها زبان اسمبلی است. هرچند با زبانهای دیگری مانند **Basic**، **C**، **Java** نیز قابل برنامه ریزی هستند.



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

میکروکنترلر

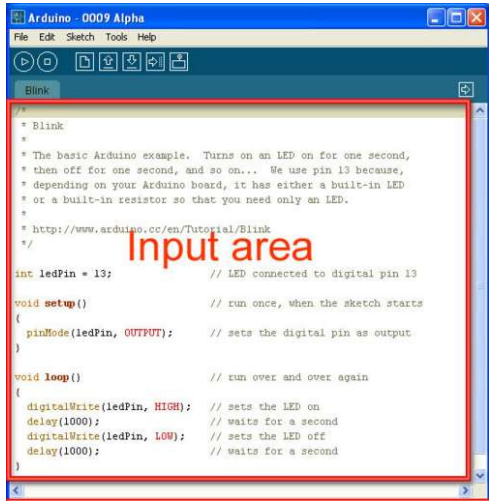
برد میکروکنترلر آردوینو:



learn.sparkfun.com

میکروکنترلر

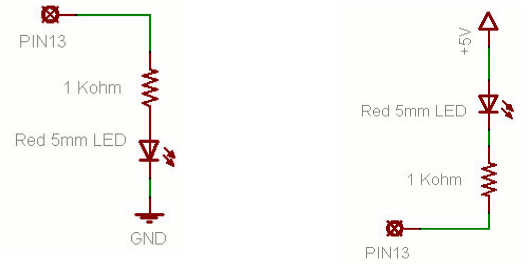
محیط برنامه نویسی:



www.ladyada.net/learn/arduino

میکروکنترلر

برنامه نویسی:

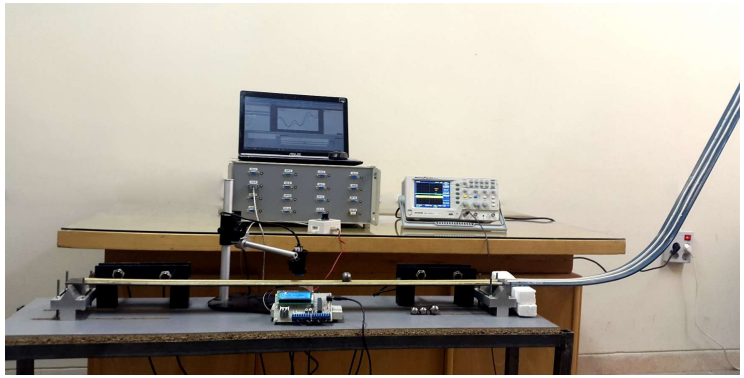



```
int ledPin = 13;                // LED connected to digital pin 13
void loop()                     // run over and over again
{
  digitalWrite(ledPin, HIGH);   // sets the LED on
  delay(500);                   // waits for a second
  digitalWrite(ledPin, LOW);    // sets the LED off
  delay(500);                   // waits for a second
}
```

www.ladyada.net/learn/arduino

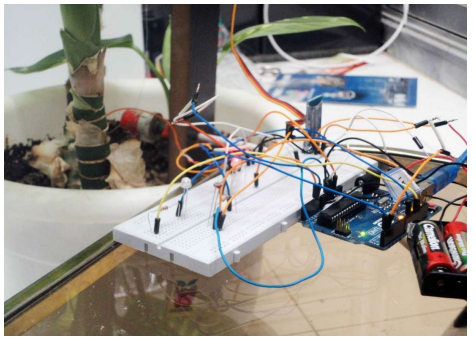
میکروکنترلر

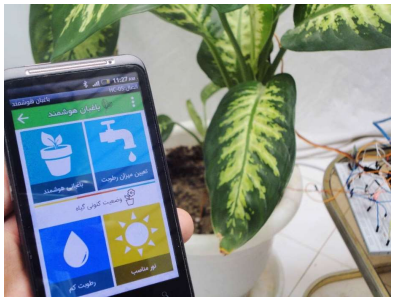
اختراع ثبت شده:



میکروکنترلر

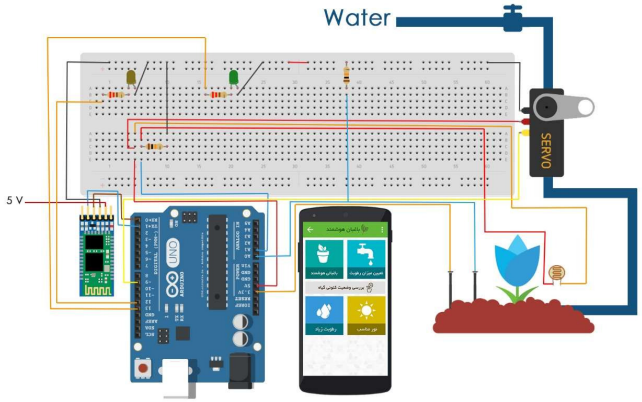
اختراع در حال ثبت:





میکروکنترلر

اختراع در حال ثبت: هوشمندسازی فرایند آبیاری در حوزه کشاورزی



میکروکنترلر

تمرین و تحقیق

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

کنترل کننده منطقی قابل برنامه ریزی (PLC)

پیش از اختراع **PLC**ها، 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.

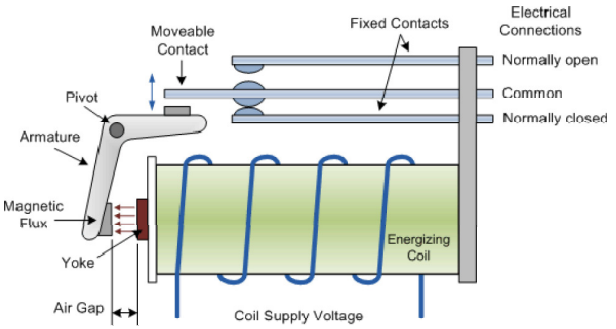
با استفاده از **PLC** تغییر در روند کنترل به راحتی و بدون نیاز به تغییر در سیم کشی و سیستم سخت افزاری صورت می پذیرد. با عوض کردن برنامه **PLC** این امر محقق می گردد.

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.

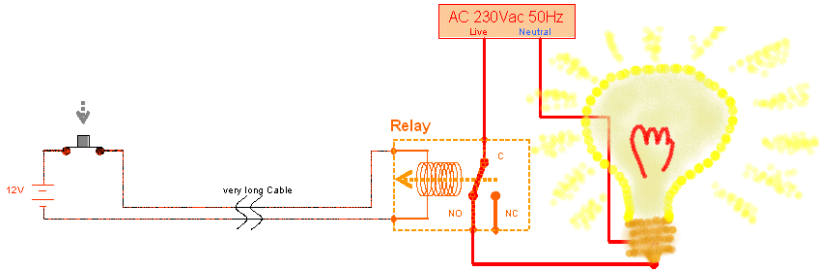
رله ها

رله مکانیکی



رله ها

رله مکانیکی

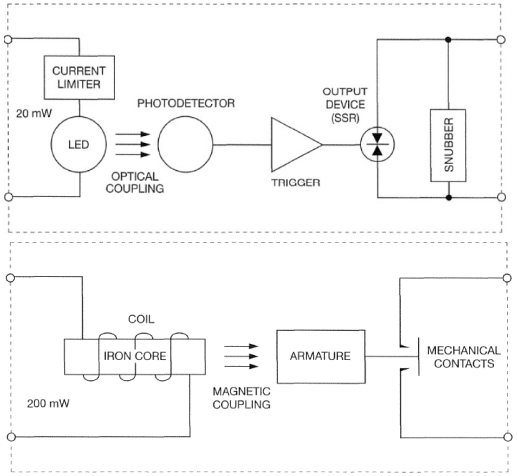


http://www.siongboon.com/projects/2006-06-19_switch/

رله ها

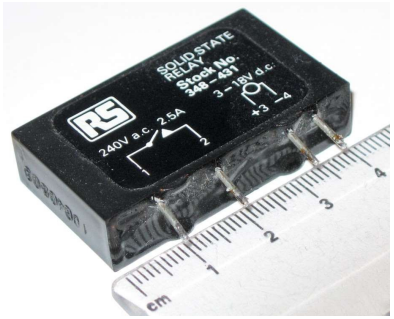
مقایسه رله نیمه هادی با رله الکترومکانیکی:

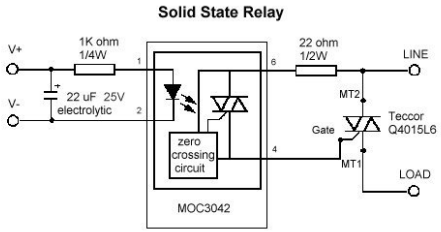
۱. توان مورد نیاز
۲. سرعت
۳. نویز
۴. فرکانس کاری
۵. اتلاف انرژی
۶. قیمت



رله ها

رله نیمه هادی

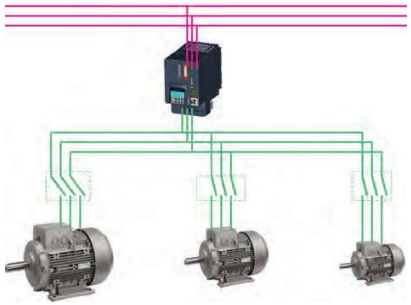




کنترل کننده منطقی قابل برنامه ریزی (PLC)

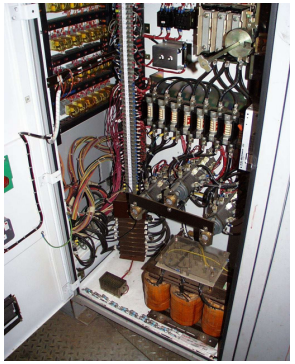
مقایسه با سیستم رله ای:

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



سیستم رله‌ای

خودکار کردن ماشینها بر اساس منطق رله‌ای:



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.

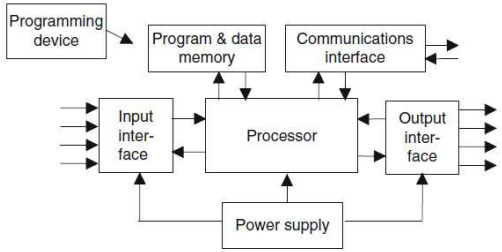
کنترل کننده منطقی قابل برنامه ریزی (PLC)

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.

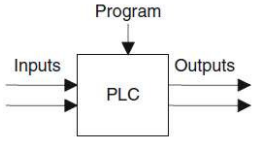
۲- PLC به گونه‌ای طراحی شده که با تمامی نیازهای کنترل یک سیستم انطباق دارد فقط لازم است هنگام طراحی شرایط را پیش‌بینی کرده و بر اساس نیاز ماژول‌ها را انتخاب و استفاده کنیم ولی در کامپیوتر جهت برقرای ارتباط با سیستم‌های مختلف صنعت لازم است روی آن یک سری ماژول اضافه شود که بررسی و خرید تجهیزات خاص جهت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

بخشهای مختلف یک PLC



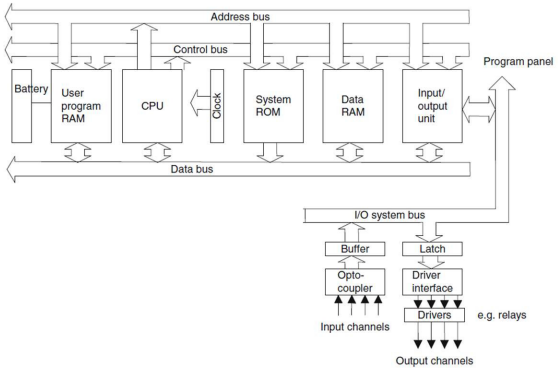
- ✓ پردازشگر (CPU)
- ✓ بخش تغذیه انرژی
- ✓ بخش ارتباط با وسیله برنامه نویسی
- ✓ واحد حافظه
- ✓ ورودی و خروجی (دیجیتال و آنالوگ)
- ✓ ارتباط با سایر PLC ها

کنترل کننده منطقی قابل برنامه ریزی (PLC)



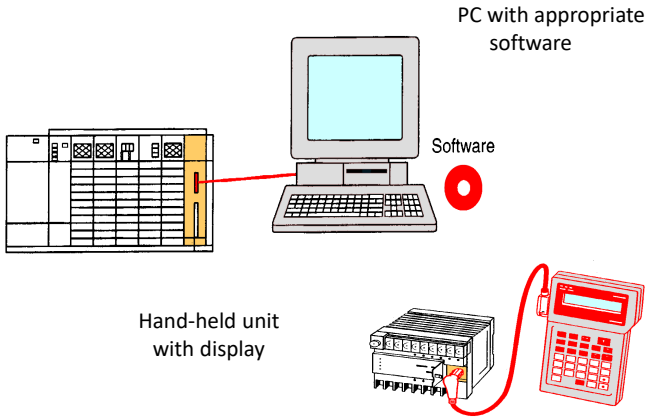
کنترل کننده منطقی قابل برنامه ریزی (PLC) نوع خاصی از کنترلرهای بر پایه میکروپروسسور است که از حافظه قابل برنامه ریزی برای ذخیره سازی دستورات و توابع اجرایی مانند منطقی، ترتیب دهی، زمانبندی، شمارش برای کنترل پروسه استفاده می کند.

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.



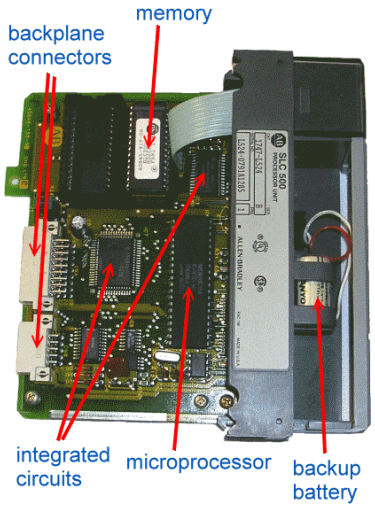
۲- تغذیه انرژی: این واحد وظیفه تبدیل ولتاژ **AC** ورودی به ولتاژ **DC** (۵ ولت) برای پردازنده و همچنین تامین انرژی سایر اجزاء و مدارهای موجود را برعهده دارد.

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.



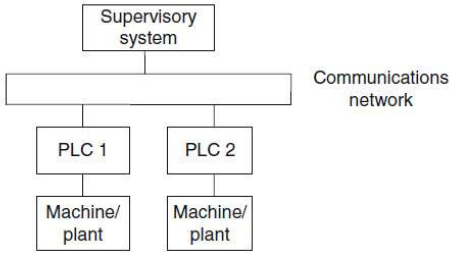
۴-واحد حافظه:

محل ذخیره برنامه کنترلی و داده های ورودی و خروجی



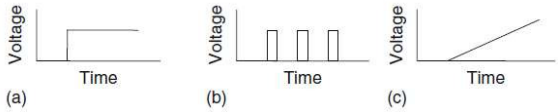
۵- ارتباط با سایر PLCها:

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



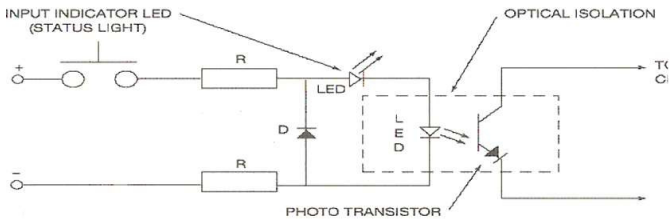
۵- ورودی و خروجی:

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.



۲- اتصال انواع ورودی به PLC:

ورودی اکثر PLCهای 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.



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

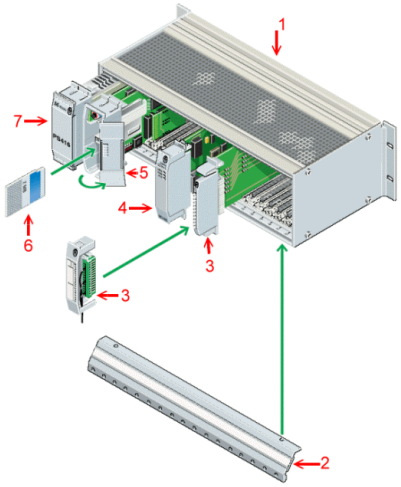
بهترین روش برای جلوگیری از آسیب رسیدن به PLC در اثر ولتاژ ورودی استفاده از چند فیوز است تا از عبور جریان بیش از حد جلوگیری کند.

کنترل کننده منطقی قابل برنامه ریزی (PLC)

اجزای **PLC** به دو صورت ممکن است ارائه شوند:

۱- ساختار یکپارچه **(Compact)**

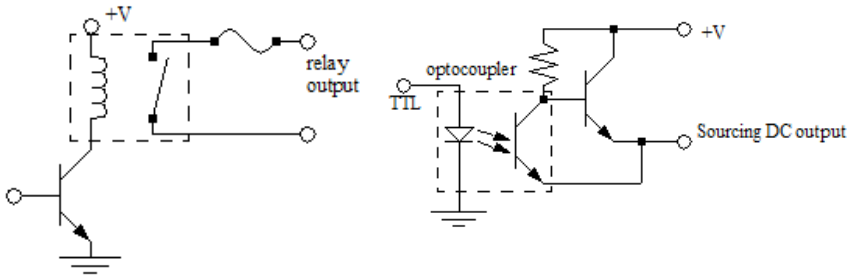
۲- ساختار مجزا و قابل توسعه **(Modular)**



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)

معرفی سری های مختلف **PLC** شرکت زیمنس:

الف) **PLC**های سری **Simatic S5**

ب) **PLC**های سری **S7**

ب۱) **Simatic S7 – 200**

ب۲) **Simatic S7 – 1200**

ب۳) **Simatic S7-300**

ب۴) **Simatic S7-400**

کنترل کننده منطقی قابل برنامه ریزی (PLC)

شرکت های سازنده **PLC**:

MITSUBISHI ELECTRIC

OMRON

AEG

Allen-Bradley

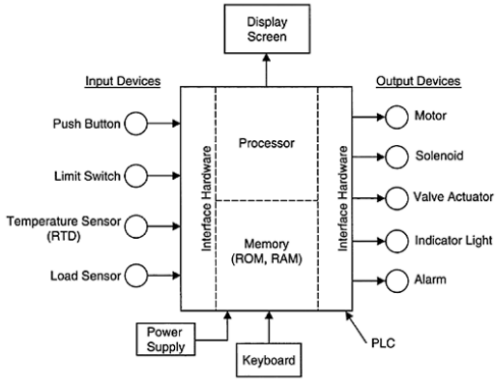
SIEMENS

DELTA

VIGOR
PROGRAMMABLE CONTROLLER

ساختار PLC:

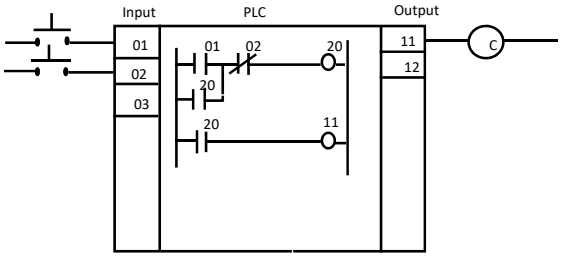
ساختار PLC:



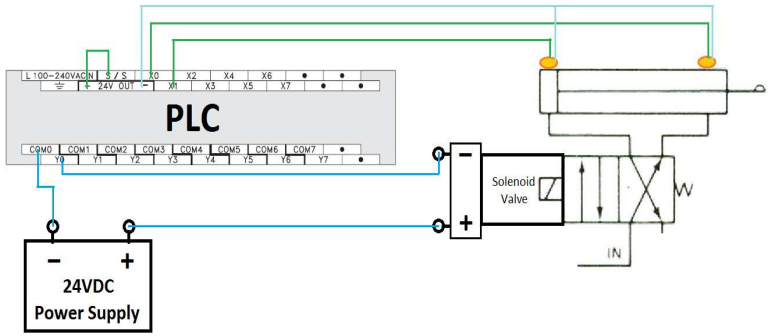
De Silva, C.W., "Mechatronics-An Integrated Approach", Taylor & Francis, CRC Press, 2005

سیم کشی در یک PLC:

سیم کشی در یک PLC:

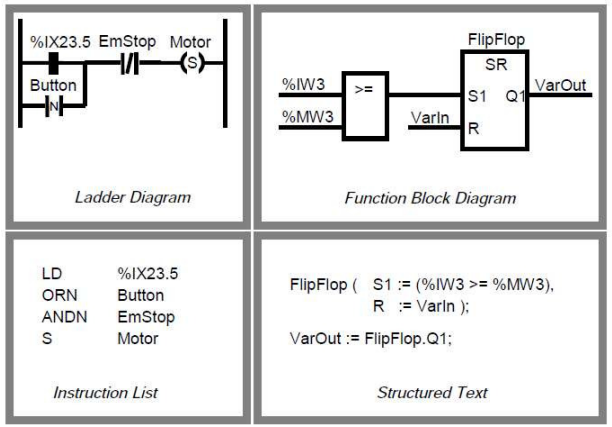


نحوه سیم کشی در PLC



برنامه نویسی PLC

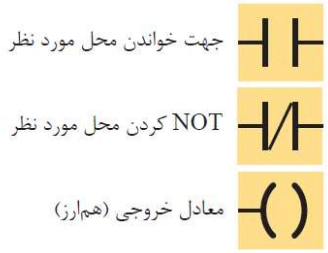
The IEC 61131 defined the standards for PLCs, with 61131-3 defining the programming languages: ladder diagrams (LAD), instruction list (IL), sequential function charts (SFC), structured text (ST), and function block diagrams (FBD).



دیاگرام نردبانی (Ladder Diagram):

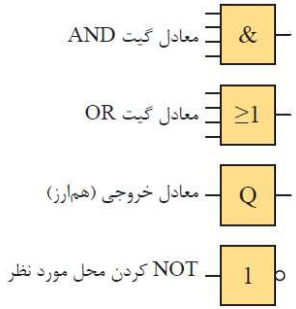
زبان برنامه نویسی گرافیکی است که از دیاگرام مدارهای کنترل رله ای نشات گرفته است.

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



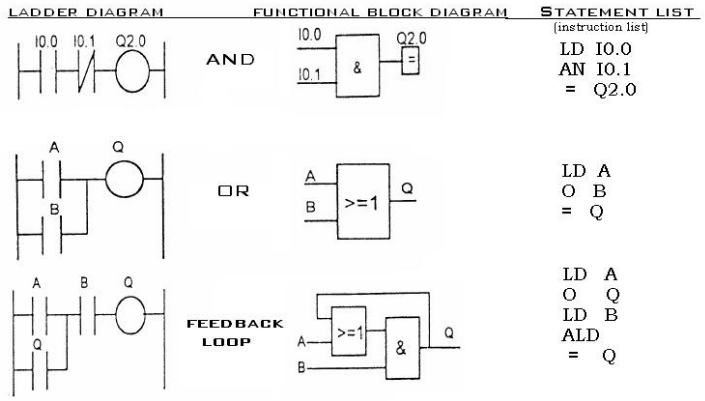
نمایش بلوکی (Function Block Diagram):

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):

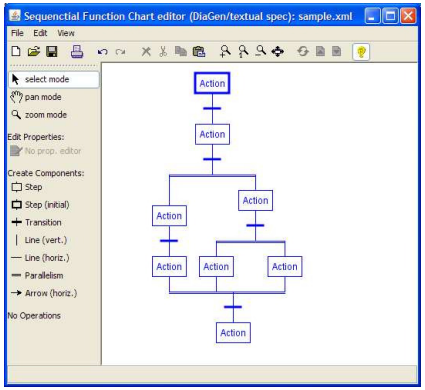
این زبان یک زبان سطح بالا بر اساس زبان **Pascal** است که شامل تعاریف و دستورالعمل ها می باشد.

```
// 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;
```

برنامه نویسی PLC

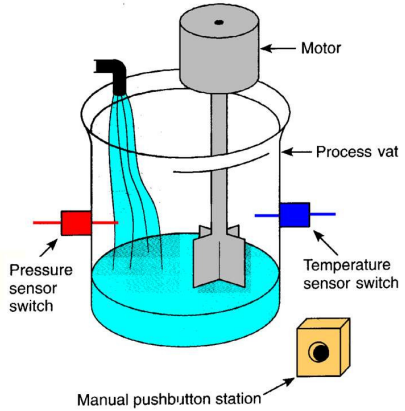
جدول توابع پیاپی (Sequential Function Chart):

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



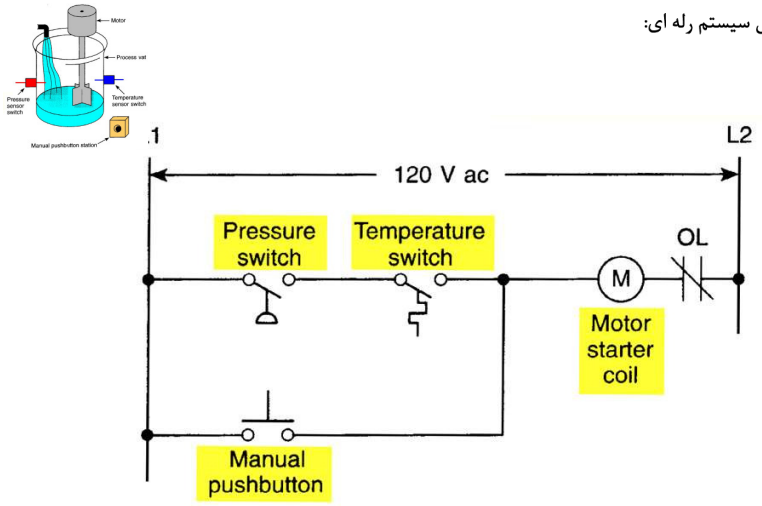
دیاگرام نردبانی (LD)

با یک مثال شروع می کنیم:



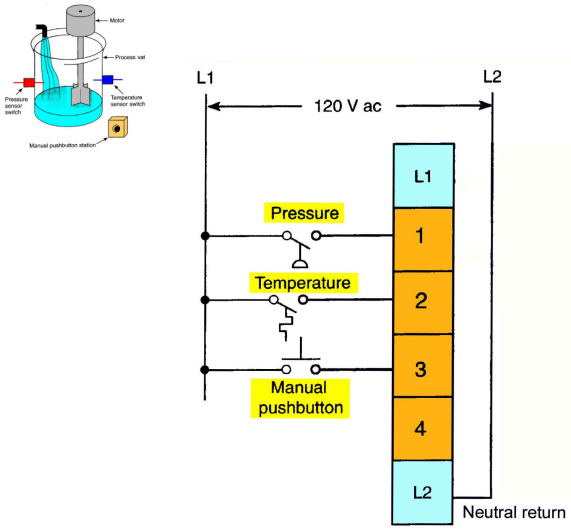
دیاگرام نردبانی (LD)

نمایش سیستم رله ای:



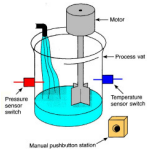
دیاگرام نردبانی (LD)

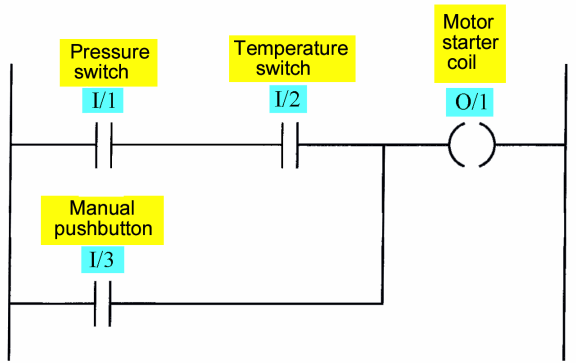
ورودی به PLC:



دیاگرام نردبانی (LD)

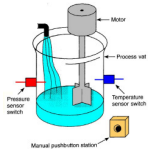
برنامه PLC:

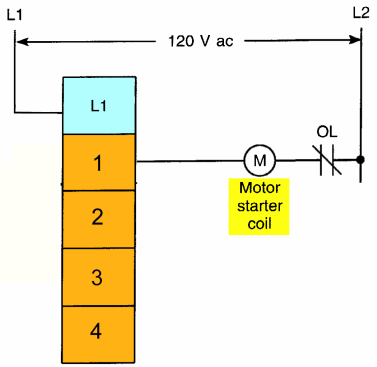




دیاگرام نردبانی (LD)

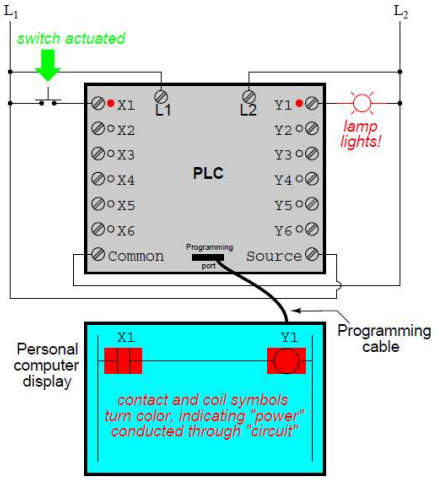
خروجی از PLC:





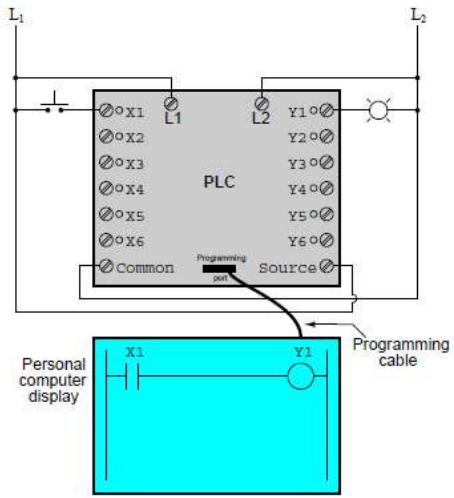
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



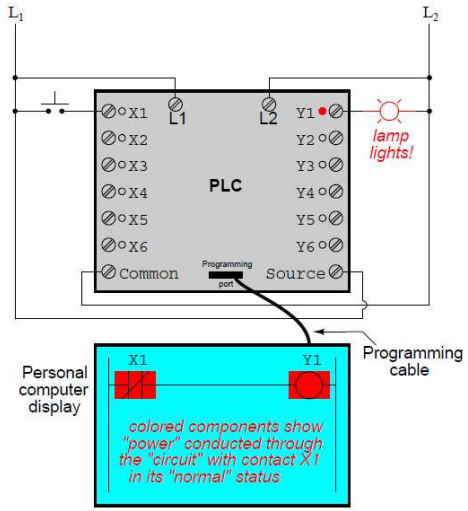
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



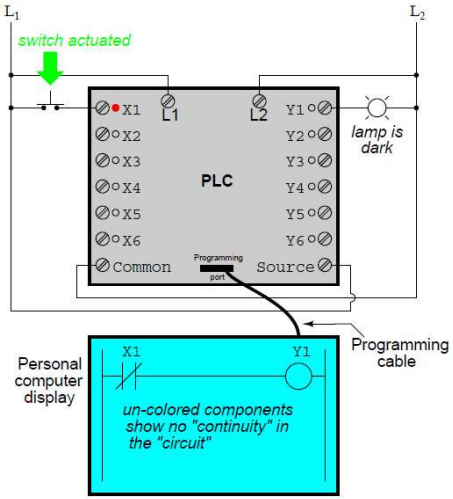
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



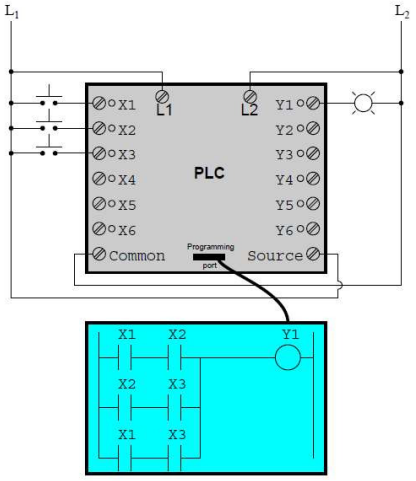
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



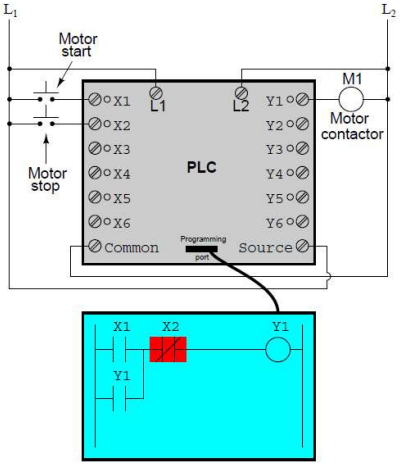
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

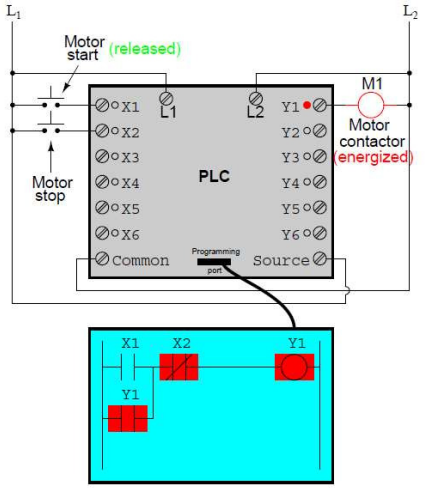
آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

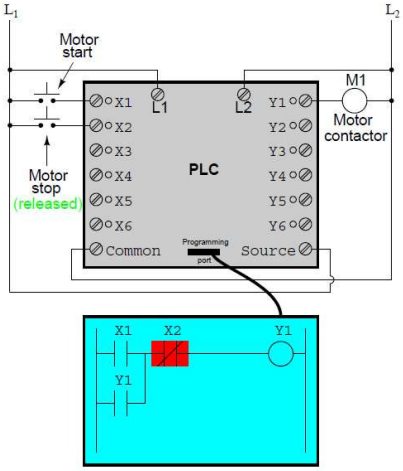
آشنایی با المانهای ساده:

(مدار حافظه، نگهدارنده یا **Latch**)



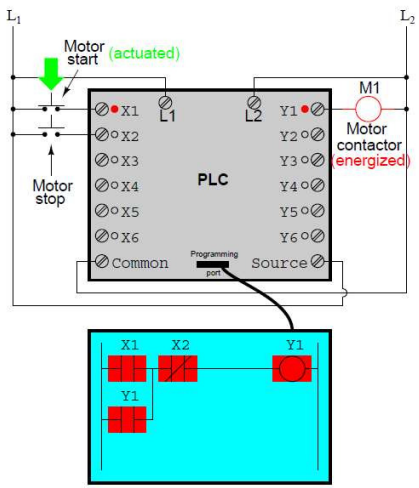
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



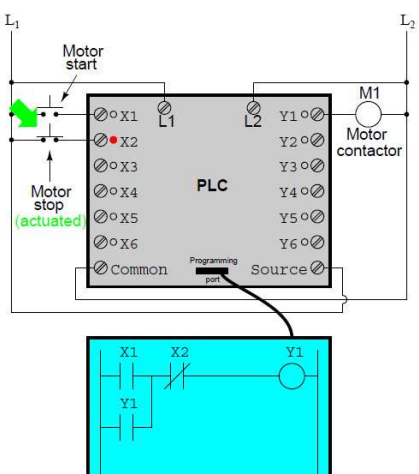
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



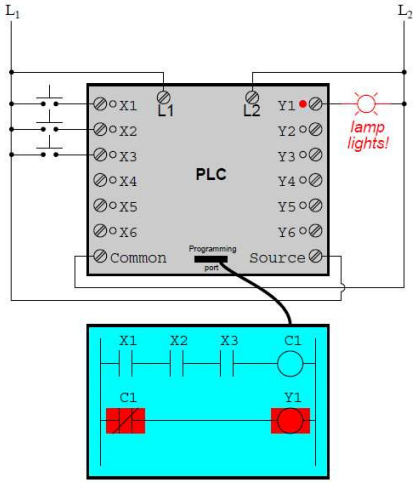
دیاگرام نردبانی (LD)

آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

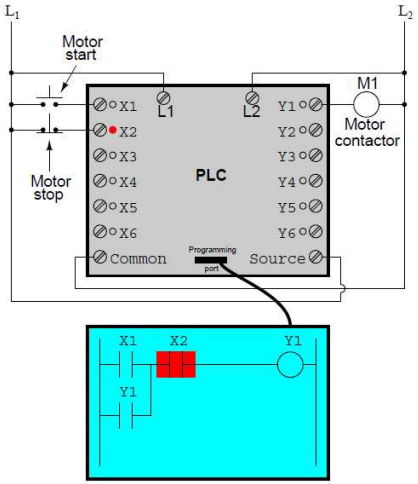
آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

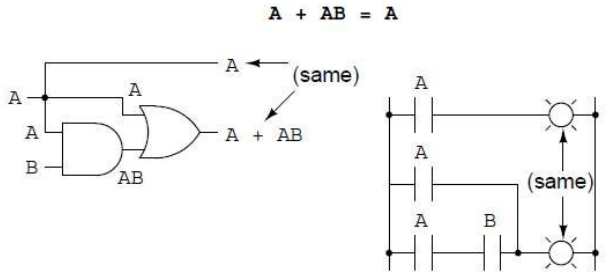
آشنایی با المانهای ساده:

- طراحی ایمن



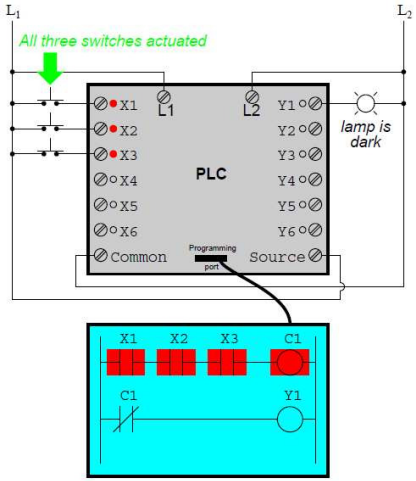
دیاگرام نردبانی (LD)

مدارهای منطقی و دیاگرام نردبانی:



دیاگرام نردبانی (LD)

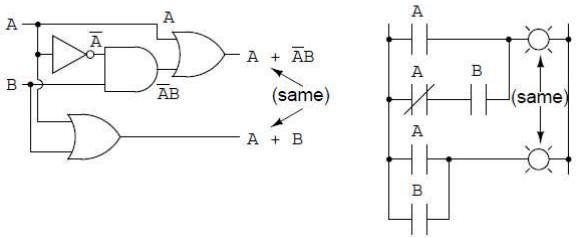
آشنایی با المانهای ساده:



دیاگرام نردبانی (LD)

مدارهای منطقی و دیاگرام نردبانی:

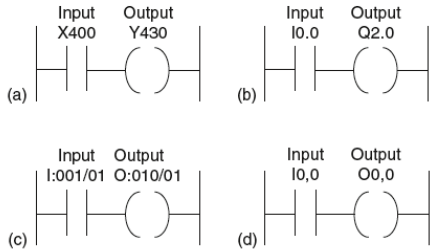
$$A + \overline{A}B = A + B$$



دیاگرام نردبانی (LD)

در ادامه به بررسی اصول کلی برنامه نویسی و جزئیات مشترک در **PLC**های صنعتی می پردازیم.

سازندگان **PLC** از محدوده اعداد متفاوتی برای آدرس دهی ورودیها و خروجیها استفاده می کنند.

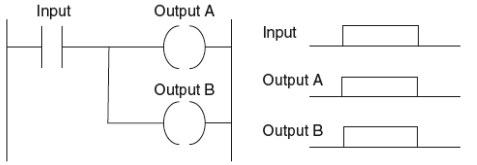


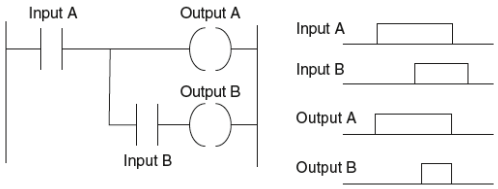
Notation: (a) Mitsubishi (b) Siemens (c) Allen-Bradley (d) Telemecanique

Ladder and Functional Block Programming - ElsevierDirect

دیاگرام نردبانی (LD)

- ✓ در دیاگرام نردبانی می توان بیش از یک خروجی را روشن نمود.

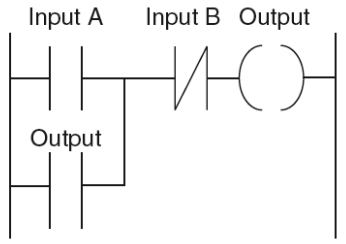




Ladder and Functional Block Programming - ElsevierDirect

دیاگرام نردبانی (LD)

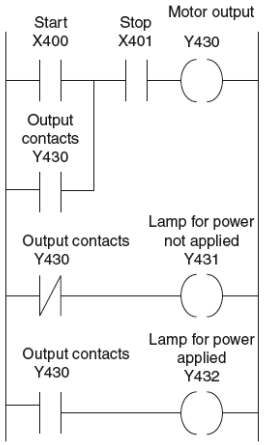
- ✓ برای آنکه خروجی روشن شده توسط یک کلید ورودی پس از خاموش شدن آن روشن باقی بماند از تکنیک حافظه نگهدارنده یا لچ (**Latch**) استفاده می شود.
- ✓ این مدار زمانی که برای روشن کردن یک خروجی از کلید فشاری استفاده شود، به کار گرفته می شود.



Ladder and Functional Block Programming - ElsevierDirect

دیاگرام نردبانی (LD)

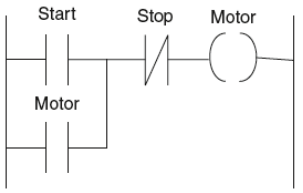
مثال:



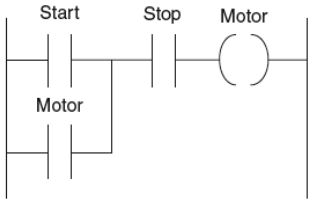
Motor on-off, with signal lamps, ladder diagram. Note that the stop contacts X401 are shown as being programmed as open. If the stop switch used is normally closed then X401 receives a start-up signal to close. This gives a safer operation than programming X401 as normally closed.

دیاگرام نردبانی (LD)

✓ موقعیت قرار دادن کلید توقف موتور:



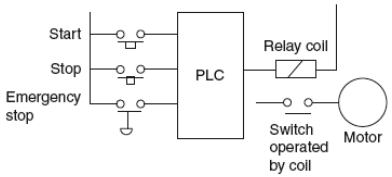
(a) An unsafe stop switch



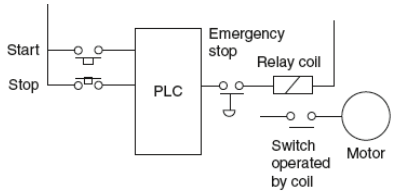
(b) A safe stop switch

دیاگرام نردبانی (LD)

✓ موقعیت قرار دادن کلید توقف اضطراری:



(a) Unsafe arrangement

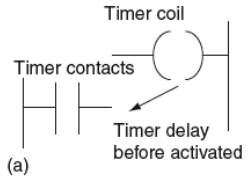


(b) Safe arrangement

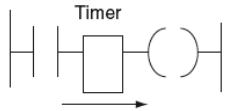
دیاگرام نردبانی (LD)

✓ تایمر (Timer)

زمان سنجها یا تایمرها در واقع شمارنده های ثانیه ها و یا بخشی از ثانیه توسط ساعت سخت افزاری و یا CPU می باشند. توسط تایمر می توان اعمال فرمان به یک خروجی را یا با تأخیر اعمال نمود و یا اعمال فرمان را تا یک زمان مشخص ادامه داد.



(a)



(b) Timer delay before input signal reaches output

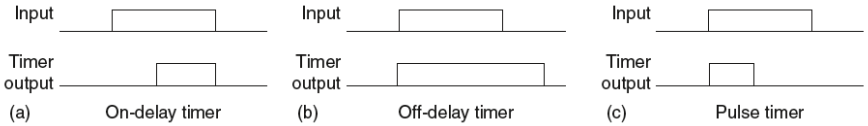
✓ تایمر (Timer)

سه نوع تایمر در PLCها وجود دارد. (در PLCهای کوچک تنها تایمر نوع اول وجود دارد.)

۱- تایمر روشن ساز با تاخیر

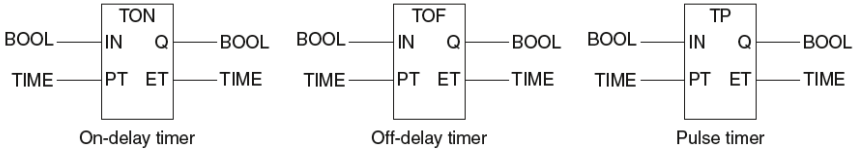
۲- تایمر خاموش شونده با تاخیر

۳- تایمر پالسی



✓ تایمر (Timer)

نمایش تایمرها طبق استاندارد IEC 1131-3 به صورت زیر است:

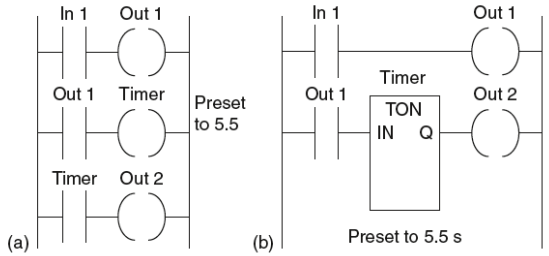


IEC 1131-1 standards. BOOL indicates a Boolean input/output, i.e., on/off. IN is the input. Q is the output. ET is the elapsed time output. PT is the input used to specify the time.

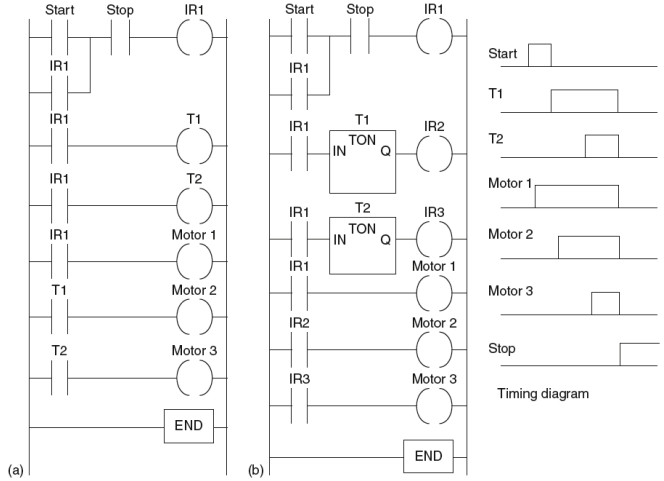
✓ تایمر روشن ساز با تاخیر

a) نمایش تایمر به صورت کویل

b) نمایش تایمر به صورت بلوک تاخیر



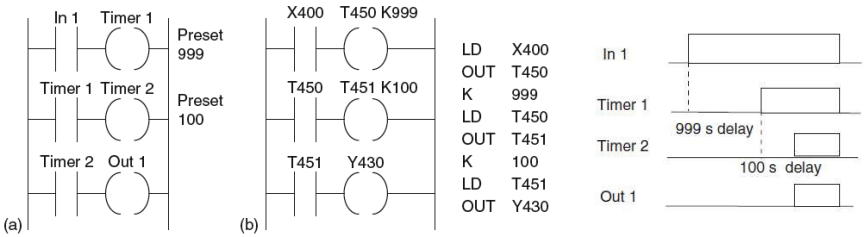
مثال:راه اندازی سه موتور



دیاگرام نردبانی (LD)

زمان تنظیم شده برای یک تایمر بین ۰،۱ تا ۹۹۹ ثانیه است.

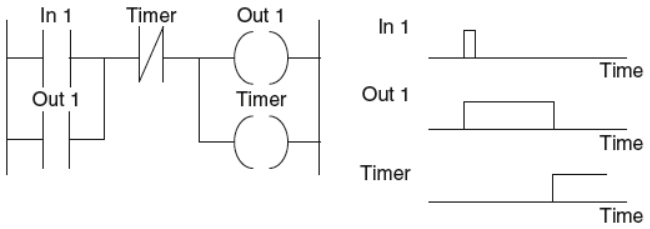
اگر زمان مورد نیاز ۱۰۹۹ ثانیه باشد، داریم:



دیاگرام نردبانی (LD)

✓ تایمر خاموش شونده با تاخیر

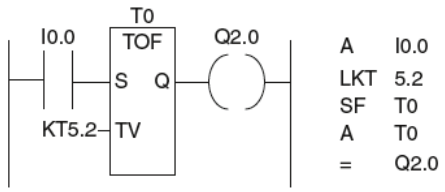
الف) نمایش تایمر با استفاده از تایمر روشن شونده با تاخیر



دیاگرام نردبانی (LD)

✓ تایمر خاموش شونده با تاخیر

ب) نمایش تایمر به صورت بلوک تاخیر

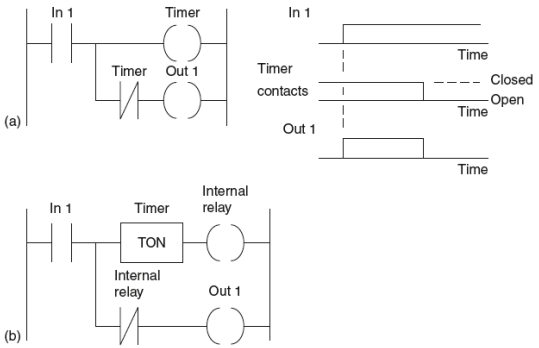


دیاگرام نردبانی (LD)

✓ تایمر پالسی:

a) نمایش تایمر به صورت کویل

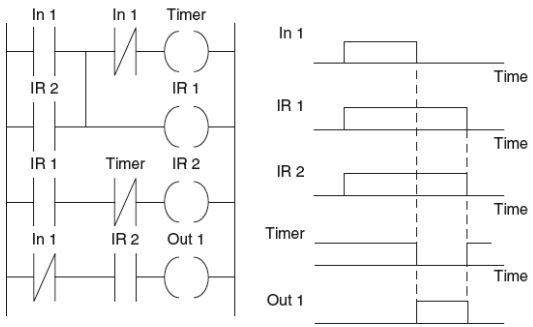
b) نمایش تایمر به صورت بلوکی



دیاگرام نردبانی (LD)

✓ تایمر پالسی:

تایمر پالسی با قابلیت شروع شدن در زمان رها شدن کلید

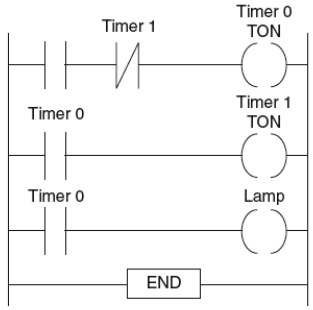


Ladder and Functional Block Programming - ElsevierDirect

دیاگرام نردبانی (LD)

✓ مثال:

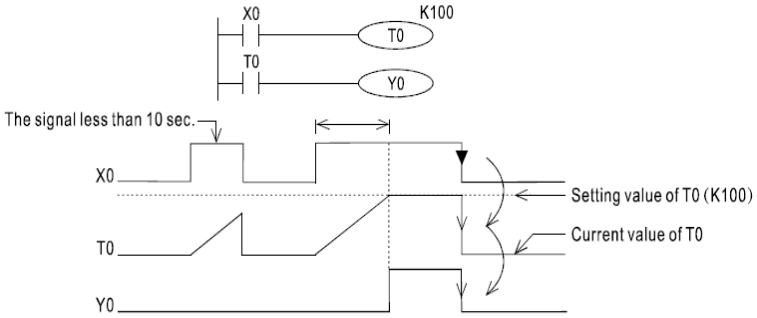
دیاگرام نردبانی برای کنترل چراغی که هر یک ثانیه خاموش و روشن می شود.



Ladder and Functional Block Programming - ElsevierDirect

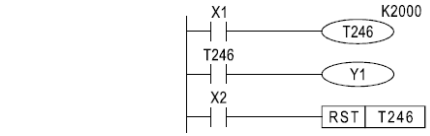
دیاگرام نردبانی (LD)

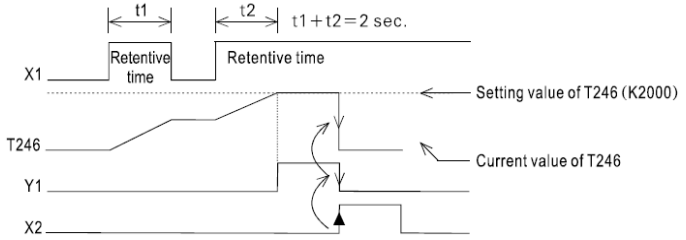
✓ توضیحی در مورد **Timer**:



دیاگرام نردبانی (LD)

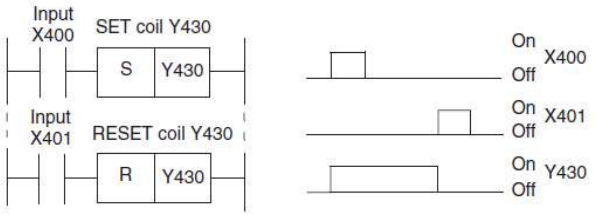
✓ توضیحی در مورد **Timer** نگهدارنده:





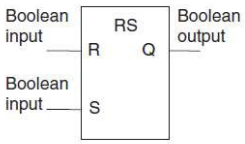
دستور SET و RESET:

درصورتی‌که بخواهیم با تحریک یک متغیر، خروجی فعال گردد و تا زمانی که بخواهیم مقدار یک منطقی داشته باشد؛ از SET و برای غیرفعال‌سازی از RST استفاده می‌کنیم. در واقع یک کلید ماندگار ایجاد کرده‌ایم.



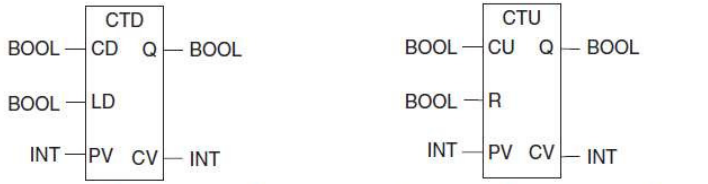
✓ بلوک تابع RS:

ورودی S خروجی Q را فعال می کند و ورودی R خروجی Q را غیرفعال می کند . یک رله نگهدارنده بصورت یک سلول حافظه دودویی می باشد . روشن بودن یک خروجی بستگی به وضعیت ورودی ها و وضعیت قبلی خروجی دارد . هرگاه ورودی های S و R هر دو باهم در وضعیت یک قرار بگیرند ، در آنصورت اولویت با پایه R می باشد .



S	R	Q
0	0	بدون تغییر
0	1	0
1	0	1
1	1	0

✓ شمارنده (Counter):

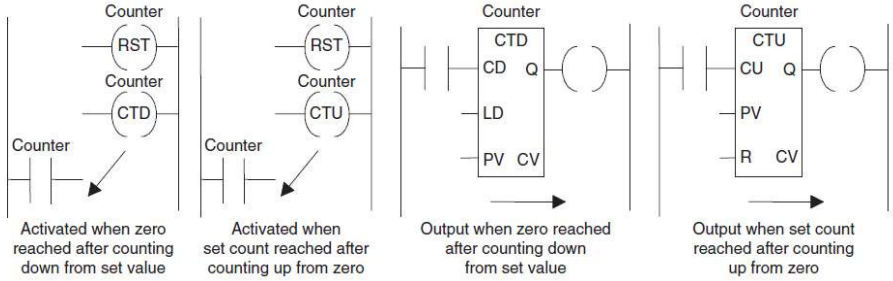


(a) The pulses at CD are counted. When the counter goes from the start PV value to 0, Q is set to 1 and the counting stops. An input to LD clears Q to 0.

(b) The pulses at CU are counted. When the counter reaches the PV value, Q is set to 1 and the counting stops. An input to R clears Q to 0.

نمادهای استاندارد IEC

✓ شمارنده (Counter):



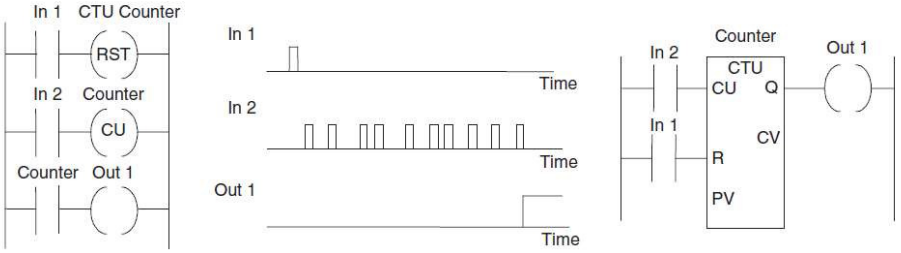
نمایش به صورت کویل

نمایش به صورت بلوکی

دیاگرام نردبانی (LD)

✓ شمارنده (Counter):

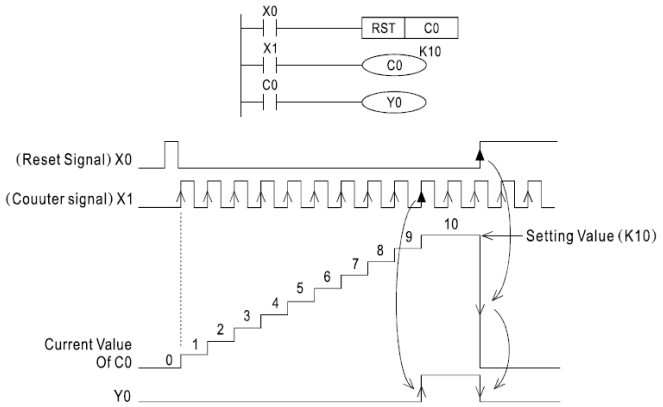
مثال:



Ladder and Functional Block Programming - ElsevierDirect

دیاگرام نردبانی (LD)

✓ شمارنده (Counter):

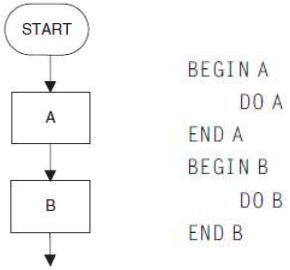


دیاگرام نردبانی (LD)

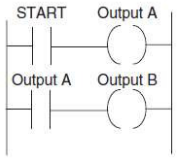
چگونه یک مسأله را به صورت دیاگرام نردبانی بنویسیم؟

۱- مسائل ترتیبی (Sequential)

در این مسائل رخداد **B** بعد از رخداد **A** انجام می شود:



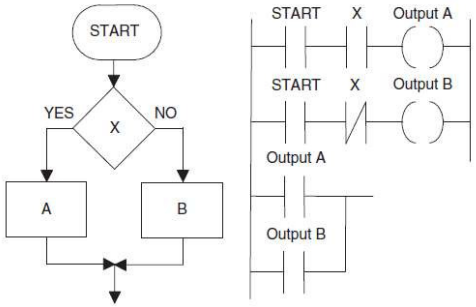

```
BEGIN A
    DO A
END A
BEGIN B
    DO B
END B
```



دیاگرام نردبانی (LD)

۲- مسائل شرطی (Conditional)

انجام **A** یا **B** بستگی به **yes/no** بودن **X** دارد.



سایر دستورات:

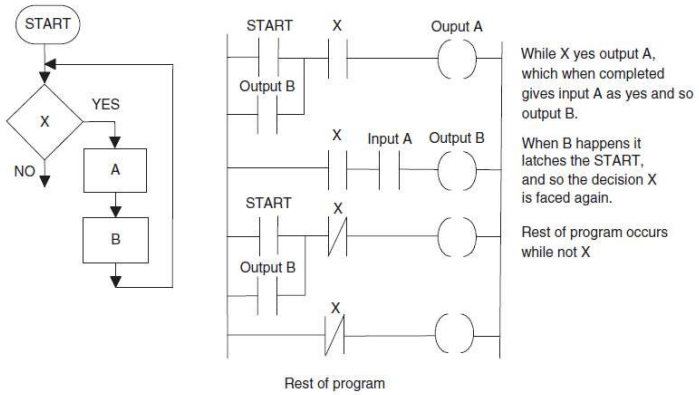
PLSY:

با استفاده از دستور PLSY می‌توان پالس‌هایی با فرکانس دلخواه ایجاد کرد. این پالس‌ها جهت ارائه سرعت مورد نظر در درایورها استفاده می گردد.

S1 : Pulse output frequency
S2 : Number of pulse outputs
D : Pulse output point

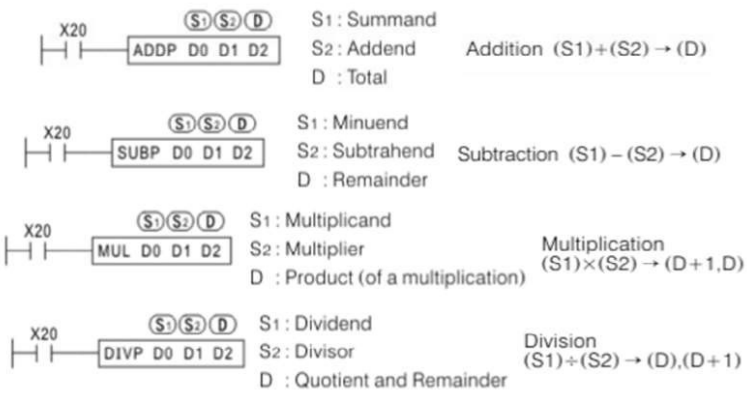
۳- مسائل در حلقه (Looping)

در این مسائل بخشی از برنامه تا وقوع شرایطی تکرار می گردد.



دستورات اعمال ریاضی

با استفاده از دستورات جمع، تفریق، ضرب و تقسیم می‌توان اعمال ریاضی نیز در PLC انجام داد. طریقه استفاده از آن‌ها در زیر آمده است.



دستور انتقال

با استفاده از دستور انتقال می توان مقادیر خاص را به متغیرها اختصاص داد به این صورت که در صورت استفاده از تابع می توان دستورات آن را با استفاده از متغیرهای مثل D0 و... ایجاد کرد و سپس جهت فراخوانی تابع با استفاده از متغیرهای آن از دستور MOV قبل از عبارت K100 D0 استفاده می‌کنیم با این کار مقدار ۱۰۰ به متغیر D0 انتقال می‌یابد.

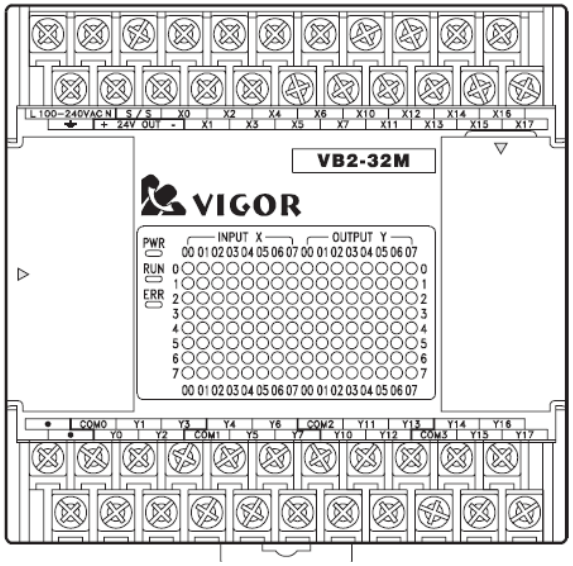
```
MOV K100
    D0
```

دیاگرام نردبانی (LD)

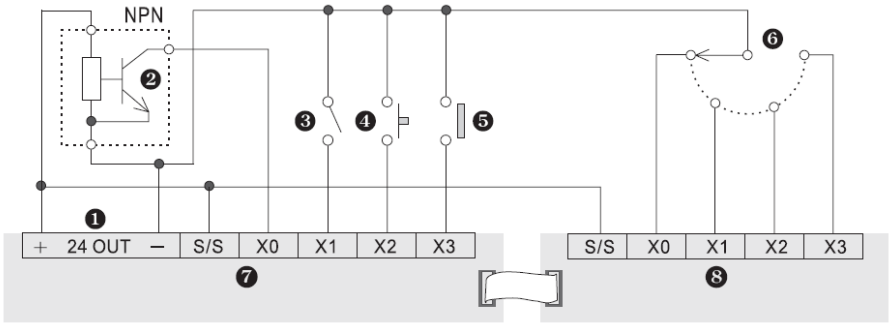
مقایسه کننده:

D= S1 S2	Initial comparison contact. Active when (S1)=(S2)
D> S1 S2	Initial comparison contact. Active when (S1)>(S2)
D< S1 S2	Initial comparison contact. Active when (S1)<(S2)
D<> S1 S2	Initial comparison contact. Active when (S1)≠(S2)
D<= S1 S2	Initial comparison contact. Active when (S1)≤(S2)
D>= S1 S2	Initial comparison contact. Active when (S1)≥(S2)

اتصالات سخت افزاری

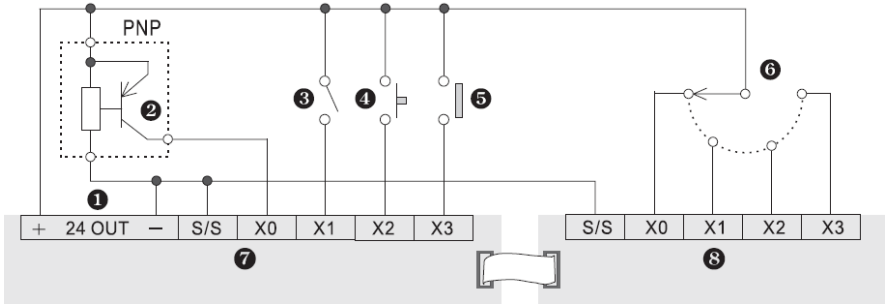


اتصالات سخت افزاری



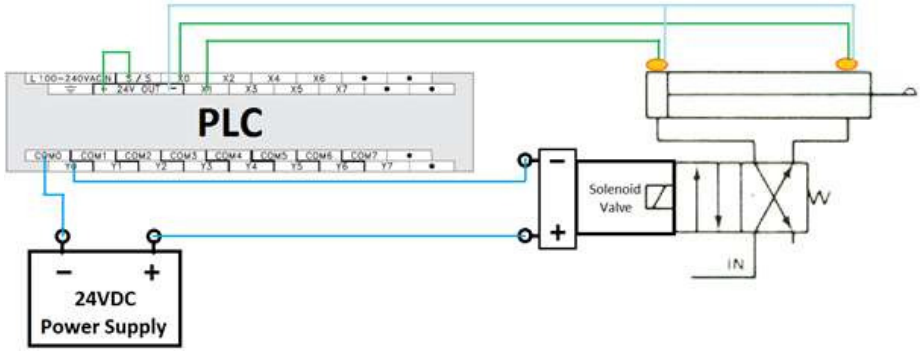
❶ DC 24V Output
❷ NPN/PNP Proximity (Photoelectric) Switch
❸ Change-over Switch
❹ Push-button Switch
❺ Limit (Micro) Switch
❻ Selector Switch
❼ PLC Main Unit
❽ Expansion Unit

اتصالات سخت افزاری



❶ DC 24V Output
❷ NPN/PNP Proximity (Photoelectric) Switch
❸ Change-over Switch
❹ Push-button Switch
❺ Limit (Micro) Switch
❻ Selector Switch
❼ PLC Main Unit
❽ Expansion Unit

اتصالات سخت افزاری



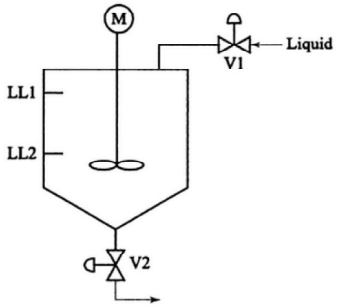
تمرین
در کلیه تمرینها از نمایش کویلی استفاده نمایید.

✓ **تمرین ۱:**

مخزن نشان داده شده در شکل را در نظر بگیرید. شیر V1 با فشار دادن دکمه فشاری PB1 باز شده و سیال وارد مخزن می‌شود. در همین زمان، موتور M شروع به کار می‌کند. زمانی که سیال از حد LL2 عبور کند و به LL1 برسد، شیر V1 بسته شده و موتور متوقف می‌شود. در صورتی‌که کلید PB1 دوباره فشرده شود، شیر V2 باز شده و سیال شروع به خارج شدن می‌کند. وقتی سیال به حدی کمتر از LL2 رسید، شیر V2 بسته می‌شود. این پروسه ۵ بار تکرار می‌شود. زنگ پایان پس از این ۵ بار تکرار به صدا در می‌آید و با زدن دکمه فشاری PB2 خاموش می‌شود. در این حالت سیستم آماده شروع مجدد می‌شود.

الف) برنامه PLC برای رسیدن به این سیستم را طراحی نمایید.

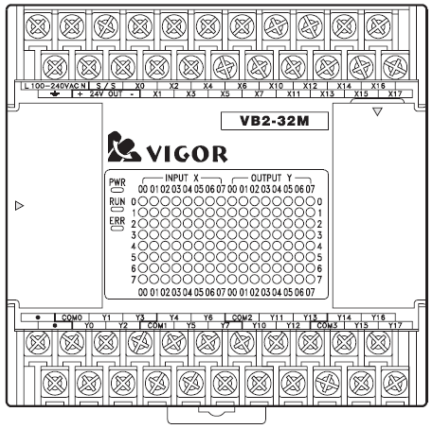
ب) برنامه نوشته شده را به کمک نرم افزار LOGO شبیه سازی نمایید.



تمرین

✓ **تمرین ۱:**

ب) نحوه اتصالات سیستم به PLC را بر روی شکل شماتیک نشان دهید.



تمرین

✓ **تمرین:**

۲. دیاگرام نردبانی برای انجام عملیات زیر را بنویسید و در نرم افزار **LOGO** شبیه سازی کنید.

- با فشردن کلید **A** چراغ برای ۵ ثانیه روشن و خاموش شود.
- روشن و خاموش شدن لامپ به صورت ۰/۲۵ ثانیه روشن و ۰/۷۵ ثانیه خاموش باشد.
- در صورتیکه کلیه **A** بیش از ۵ مرتبه فشرده شود، لامپ تا ریست شدن سیستم روشن نشود.
- ریست شدن سیستم با فشردن کلید **B** اتفاق می افتد.