

# جزوه آموزشی LOGO!

قابل استفاده برای دانشجویان برق و کنترل صنعتی

تالیف: مهندس احمد طهماسبی

By: A. Tahmasebi

[Tahmasebi\\_eng63@yahoo.com](mailto:Tahmasebi_eng63@yahoo.com)



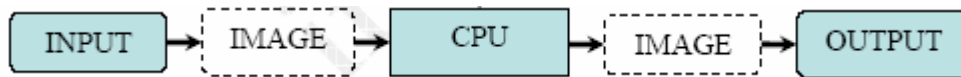
## فهرست مطالب

مقدمه.....	معرفی PLC
فصل اول.....	سخت افزار LOGO
فصل دوم.....	سیم بندی سخت افزار
فصل سوم.....	آشنائی با دستورات LOGO
فصل چهارم.....	برنامه نویسی بر روی سخت افزار
فصل پنجم.....	سنسورها
ضمیمه.....	اطلاعات فنی

## مقدمه:

**PLC** از عبارت **Programable Logic Controler** به معنای کنترل کننده منطقی برنامه پذیر گرفته شده است. اولین سیستمهای PLC با استفاده از رایانه های معمولی در اواخر دهه ۱۹۶۰ و اوایل دهه ۱۹۷۰ پدید آمدند.

در حالت کلی PLC از بخشهای زیر تشکیل یافته است:



PLC دارای تعدادی ورودی و تعدادی خروجی است، که ورودیهای آن می توانند سنسورها، شستی ها و سویچ های برقی باشند و خروجی های آن نیز می توانند لامپها، کنتاکتورها و موتورها و... باشند.

از شرکتهای سازنده PLC می توان SIEMENS، ALLEN BRADLEY، OMRON،

HITACHI، TELEMECANIQUE، MITSUBISHI، FANUC را نام برد. از این میان PLC های SIEMENS از

مقبولیت بیشتری برخوردار هستند.

**مزایای PLC نسبت به مدارهای فرمان رله ای:**

۱- استفاده از PLC موجب کاهش حجم تابلوی فرمان می شود.

۲- صرفه جوئی در هزینه

۳- عمر بیشتر

۴- انرژی کمتری مصرف می کنند.

۵- PLC ها بر خلاف مدارات رله کنتاکتوری، نویزهای الکتریکی و صوتی ایجاد نمی کنند.

۶- طراحی و اجرای مدارهای کنترل و فرمان با استفاده از PLC های بسیار سریع و آسان است.

۷- عیب یابی مدارات فرمان الکترومکانیکی بسیار سخت است، حال آنکه عیب یابی در مدارات فرمان کنترل شده توسط PLC بسیار به آسانی و با سرعت بیشتری صورت می گیرد.

۸- ضریب اطمینان بالا

۹- بالا بودن امنیت برنامه PLC از طریق PASSWORD دادن به آن

۱۰- به آسانی می توان در مدارات فرمان کنترل شده با PLC تغییرات ایجاد کرد، حال آنکه چنین موردی

در مدارات فرمان الکترومکانیکی بسیار پیچیده است.

۱۱- امکان شبیه سازی فرایند قبل از پیاده سازی آن

انواع PLC های زیمنس عبارتند از:

**LOGO!**(miniplc):ارزان قیمت-برای کاربرد های ساده- غیر ماژولار(compact)

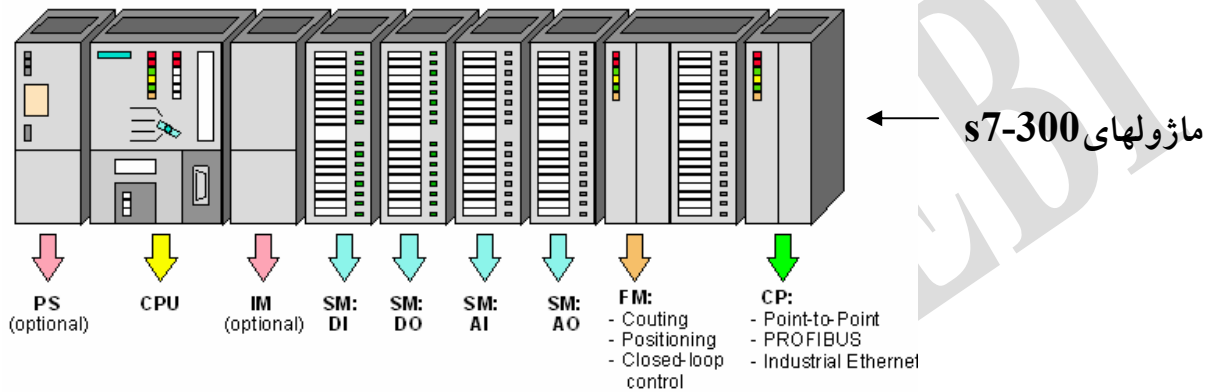
**step5**: {90u-95u}-{100u-115u}-{135u-155u}

کاربردهای وسیع کاربردهای متوسط کاربردهای محدود

**Step7**: simatic s7-200:غیر ماژولار(compact)

simatic s7-300 ماژولار- برای کاربردهای متوسط

simatic s7-400 ماژولار- برای کاربردهای پیچیده وبزرگ



## فصل اول

### Logo!

#### معرفی سخت افزار! Logo!



امروزه سیستمهای کنترلی که برای کنترل از لوگو استفاده می کنند با توجه به سادگی کنترل، قابلیت توسعه و هزینه نسبتاً پایتتر آنها نسبت به سیستمهایی که از PLC استفاده میکنند بسیار مورد توجه هستند. از جمله موارد کاربردی که از این سیستمها استفاده می شود، می توان به

سیستمهای کنترل روشنائی، سیستمهای آبیاری برای گیاهان گلخانه، دستگاههای صنعتی مثل خم کن وبالا بر، کلید های چند مرحلهای یا زمانی، تنظیم بار ثابت روی چند مصرف کننده، سیستمهای کنترل متمرکز، دستگاههای برش، کنترل شابلونها در صنایع نساجی، سیستم پر کردن سیلو، چراغ راهنمائی، روشنائی راه پله، سیستم تهویه هوا و... اشاره کرد.

لازم به ذکر است که logo در سال ۱۹۹۶ وارد بازار شده است.

#### اجزاء تشکیل دهنده logo

۱- منبع تغذیه: تغذیه سخت افزار از طریق دو ترمینال L1 و N انجام می شود.

۲- ورودیها

۳- خروجیها: این خروجیها بصورت رله ای یا ترانزیستوری می باشند.

۴- دريچه اتصال کارت حافظه (micro memory card; MMC) و کابل ارتباطی برای

کامپیوتر (RS232)

۵- صفحه کنترل

۶- صفحه نمایش

☺ دو مورد ۵ و ۶ در مدل های RCO وجود ندارند.

۷- نمایش وضعیت حالت های stop/Run

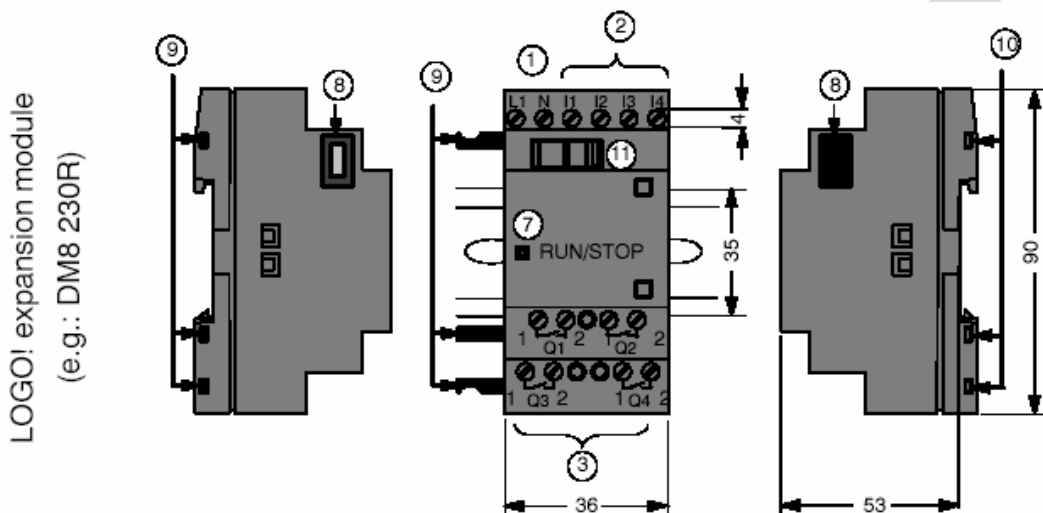
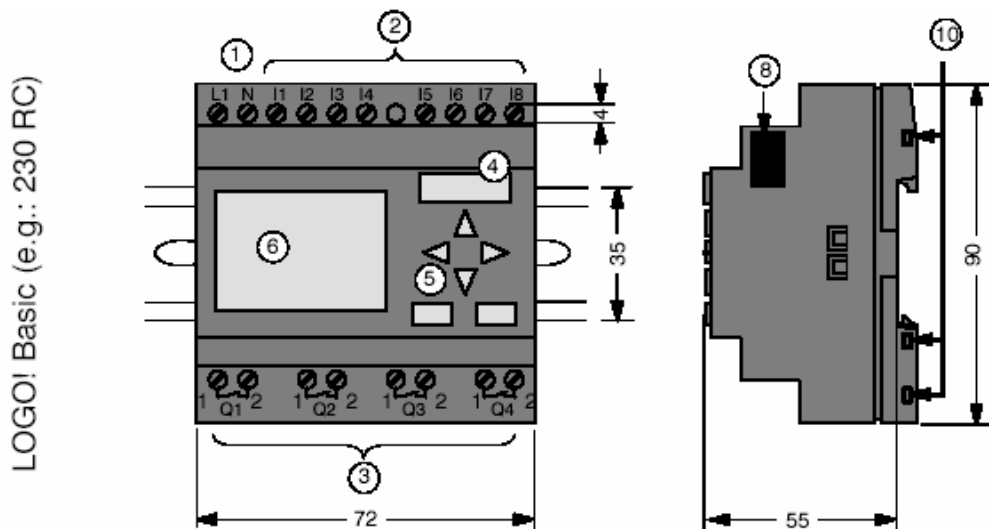
۸- رابط توسعه

۹- میله های قفل کننده مکانیکی

۱۰- حفره های قفل گذاری مکانیکی

۱۱- کشو

۱۲- ترمینال اتصال به زمین برای اتصال دادن دستگاه به زمین واستحفاظ اندازه گیری آنالوگ



نکته: نسخه هایی که دارای ۶ ورودی و در انواع قدیمی تر، نسخه های که دارای ۱۲ ورودی و ۸ خروجی هستند تقریباً منسوخ شده و دیگر استفاده نمی شوند.

چگونگی تشخیص مدل های LOGO از روی دستگاه:

برای تشخیص مدل های LOGO از روی دستگاه آشنایی با یکسری از اعداد و حروف که هر یک بیانگر

یک ویژگی از دستگاه می باشد لازم است، که عبارتند از:

عدد 12: نشان دهنده نوع 12VDC می باشد.

عدد 24: نشان دهنده نوع 24VDC می باشد.

عدد **230**: نشان دهنده نوع 230VAC...115 می باشد.

**R (relay)**: نشان دهنده خروجی های رله ای می باشد در صورتیکه حرف **R** درج نشده باشد مفهوم

آن این است که خروجیها بصورت ترانزیستوری هستند.

**C (clock)**: نشان دهنده دارا بودن سویچ تایمر وزمان بندی هفتگی است.

**O**: نشان دهنده نسخه بدون نمایشگر است.



**DM (Digital Module)**: بیانگر ماژول دیجیتال است.

**AM (Analog Module)**: بیانگر ماژول آنالوگ است.

**L (long)**: بیانگر ۱۲ ورودی و ۸ خروجی است (در حالت معمولی ۴ خروجی داریم).

Symbol	Designation	Outputs	Typs
	LOGO! 12/24RC *	4 x 230 V; 10A	Relay
	LOGO! 24 *	4 x 24 V; 0,3 A	Transistor
	LOGO! 24RC (AC)	4 x 230 V; 10A	Relay
	LOGO! 230RC	4 x 230 V; 10A	Relay
	LOGO! 12/24RCo *	4 x 230 V; 10A	Relay
	LOGO! 24RCo (AC)	4 x 230 V; 10A	Relay
	LOGO! 230RCo	4 x 230 V; 10A	Relay
	LOGO! 12RCL	8 x 230 V; 10A	Relay
	LOGO! 24L	8 x 24 V; 0.3 A	Transistor
	LOGO! 24RCL	8 x 230 V; 10A	Relay
	LOGO! 230RCL	8 x 230 V; 10A	Relay
	LOGO! 24RCLB11	8 x 230 V; 10A	Relay
	LOGO! 230RCLB11	8 x 230 V; 10A	Relay

\* با ورودیهای آنالوگ

Sym- bol	Designation	Supply voltage	Inputs	Outputs	Properties
	LOGO! 12/24 RC	12/24 V DC	8 digital (1)	4 relays (10 A)	
	LOGO! 24	24 V DC	8 digital (1)	4 solid state 24V / 0.3A	no clock
	LOGO! 24RC (3)	24 V AC / 24 V DC	8 digital	4 relays (10A)	
	LOGO! 230RC (2)	115...240 V AC/DC	8 digital	4 relays (10A)	
	LOGO! 12/24RCo	12/24 V DC	8 digital (1)	4 relays (10A)	no display unit no keyboard
	LOGO! 24o	24 V DC	8 digital (1)	4 solid state 24 V / 0.3A	no display unit no keyboard no clock
	LOGO! 24RCo (3)	24 V AC / 24 V DC	8 digital	4 relays (10A)	no display unit no keyboard
	LOGO! 230RCo (2)	115...240 V AC/DC	8 digital	4 relays (10A)	no display unit no keyboard

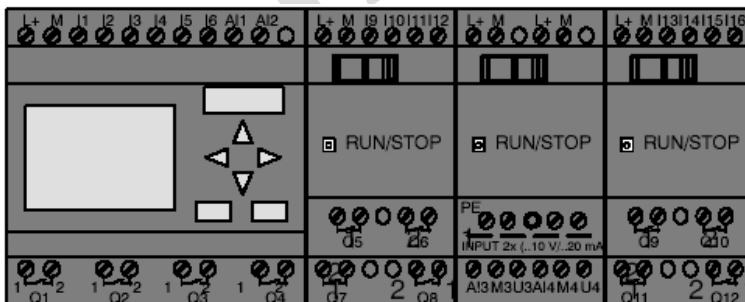
(1): Of those can be used alternatively: 2 analog inputs (0 ... 10V) and 2 fast inputs.

(2): 230 V AC versions: Two groups consisting of 4 inputs each. Each input within a group must be connected to the same phase. It is possible to interconnect groups with a different phase.

(3): The digital inputs can be operated with P or N action.

ماژولهای توسعه: هر گاه بخواهیم از ورودیها و خروجیهای اضافی برای یک دستگاه LOGO استفاده




کنیم، از ماژولهای توسعه استفاده می کنیم.



انواع ماژولهای توسعه در جدول زیر آمده است.

LOGO به همراه ماژولهای افزایشی



Symbol	Name	Power supply	Inputs	Outputs
	LOGO! DM 8 12/24R	12/24 V DC	4 digital	4 relays (5A)
	LOGO! DM 8 24	24 V DC	4 digital	4 solid state 24V / 0.3A
	LOGO! DM 8 24R (3)	24 V AC/DC	4 digital	4 relays (5A)
	LOGO! DM 8 230R	115...240 V AC/DC	4 digital <sup>(1)</sup>	4 relays (5A)
	LOGO! DM 16 24	24 V DC	8 digital	8 solid state 24V / 0.3A
	LOGO! DM 16 24R	24 V DC	8 digital	8 relays (5A)
	LOGO! DM 16 230R	115...240 V AC/DC	8 digital <sup>(4)</sup>	8 relays (5A)
	LOGO! AM 2	12/24 V DC	2 analog 0 ... 10V or 0 ... 20mA <sup>(2)</sup>	none
	LOGO! AM 2 PT100	12/24 V DC	2 Pt100 -50 °C to +200 °C	none
	LOGO! AM 2 AQ	24 V DC	none	2 analog 0 ... 10 V DC

- (1): Different phases are not allowed within the inputs.  
 (2): 0 ... 10 V, 0 ... 20 mA can be connected optionally.  
 (3): Digital inputs can be operated either with P or with N action.  
 (4): Two groups consisting of 4 inputs each. Each input within a group must be connected to the same phase. It is possible to interconnect groups with a different phase.

### ساختار مدولاتور LOGO

ساختار نهایی LOGO با ورودیهای آنالوگ (LOGO!24, LOGO!12/24RC/Rco) به صورت جدول زیر است. در مدل‌های LOGO!basic چهار ماژول دیجیتال و سه ماژول آنالوگ می‌تواند به این مدل از LOGO! اتصال یابد. در این مدل‌ها ورودیهای 7 و 18 به عنوان ورودیهای آنالوگ مورد استفاده قرار می‌گیرد.

☺ بهتر است اول ماژولهای دیجیتال اتصال یابند سپس آنالوگها، تا سرعت پردازش بهتر باشد.

نکته: جمعا ۲۴ ورودی و ۱۶ تا خروجی دیجیتال داریم و ۸ تا ورودی آنالوگ.

I1.....I6, I7, I8 AI1, AI2	I9...I12	I13...I16	I17...I20	I21...I24	AI3, AI4	AI5, AI6	AI7, AI8
LOGO! Basic	LOGO! DM 8	LOGO! DM 8	LOGO! DM 8	LOGO! DM 8	LOGO! AM 2	LOGO! AM 2	LOGO! AM 2
Q1...Q4	Q5...Q8	Q9...Q12	Q13...Q16				

نکته: هرگاه بخواهید ورودیهای ۱۷, ۱۸ را به عنوان ورودیهای آنالوگ AI1, AI2 استفاده کنید، در

آنصورت باید از به کار بردن آنها به عنوان ورودیهای دیجیتال خودداری کنید.

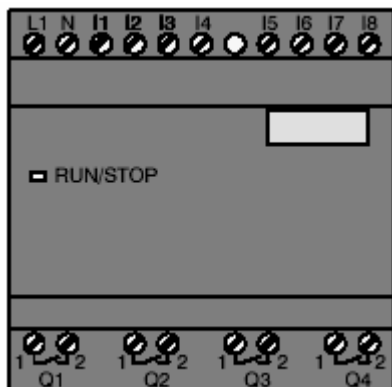
حداکثر ماژولهای آنالوگ و دیجیتال که می توان به مدل های ( LOGO!230 RC/Rco و LOGO! 24RC/Rco) وصل کرد، به صورت جدول زیر است. در مدل های LOGO!basic چهار ماژول دیجیتال و چهار ماژول آنالوگ می تواند به این مدل از LOGO! اتصال یابند.

I1 ..... I8	I9...I12	I13...I16	I17...I20	I21...I24	AI1 , AI2	AI3, AI4	AI5, AI6	AI7, AI8
LOGO! Basic	LOGO! DM 8	LOGO! DM 8	LOGO! DM 8	LOGO! DM 8	LOGO! AM 2	LOGO! AM 2	LOGO! AM 2	LOGO! AM 2
Q1...Q4	Q5...Q8	Q9...Q12	Q13... Q16					

نکته: ورودیهای ۱۵, ۱۶ در مدل های LOGO!24, LOGO!12/24RC/Rco، ورودیهای سریع نامیده می شوند. در مدل های L و ورودیهای ۱۱, ۱۲ به عنوان ورودیهای سریع بکار می روند.

**LOGO!** بدون نمایشگر و صفحه کنترل:

این نوع LOGO! در مواردی که برای عملیات نیاز به صفحه کلید و صفحه نمایش نباشد کاربرد دارد.



LOGO! بدون نمایشگر

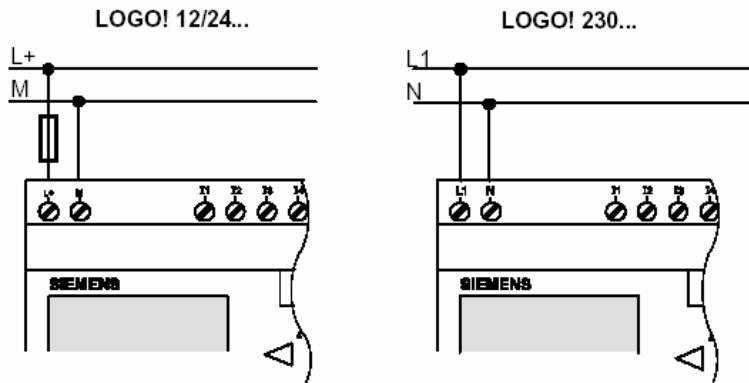
LOGO!230 RC/Rco, LOGO! 24RC/Rco مزایای استفاده از LOGO! بدون صفحه نمایش:

- ۱- استفاده از آنها بسیار آسان است.
- ۲- بدون کابل ارتباطی و نرم افزار نمی توان وارد برنامه آنها شد.
- ۳- اشخاص غیر مجاز نمی توانند از آنها استفاده کنند.

## فصل دوم

### سیم بندی LOGO

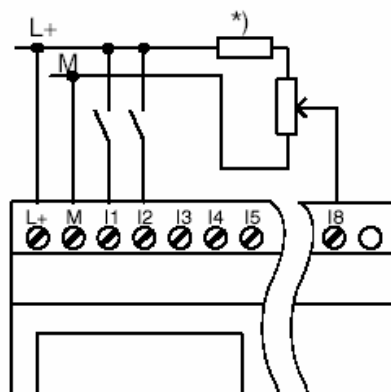
اتصال منبع تغذیه



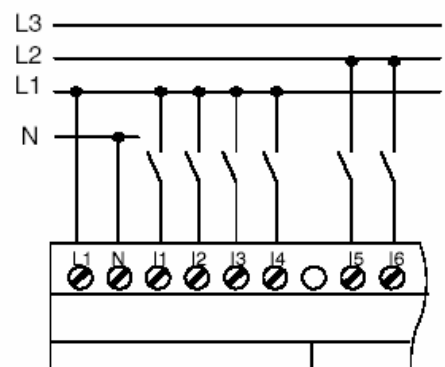
اتصال ورودیها

ورودیهای یک دستگاه LOGO می تواند انواع سنسورها، شستیها، سویچهای برقی و... باشد. در شکل زیر چگونگی اتصال سنسورها به مدل‌های مختلف LOGO را مشاهده می کنید. در جدول زیر نیز ویژگی های چند نوع سنسور برای مدل‌های مختلف LOGO بیان شده است.

LOGO! 12/24 ....

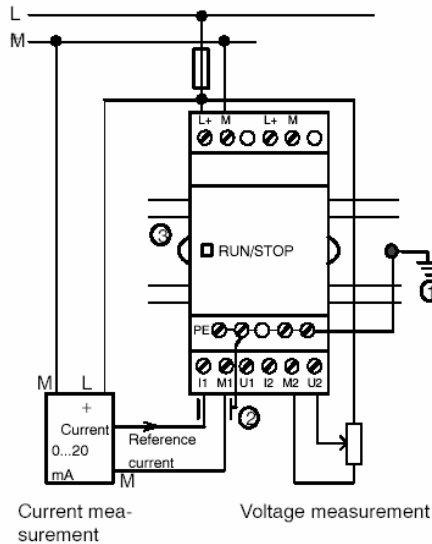


LOGO! 230 ....



در مدل 230RC/RCo ورودیهای دیجیتال به دو گروه که هر کدام دارای ۴ ورودی است تقسیم می شوند. باید توجه داشت که داخل یک گروه همه ورودیها توسط یک فاز یکسان عمل می کنند. بطور مثال ورودیهای I1 تا I4 توسط فاز L1 و ورودی I5 تا I8 بوسیله فاز L2 تغذیه می شوند.

LOGO! AM 2



اتصال ورودیهای آنالوگ

ماژول توسعه LOGO! AM2 برای افزودن ورودیهای آنالوگ

قابل دسترسی است.

اتصال سنسور دمای Pt100 به ماژول LOGO!AM2 Pt100

ابتدا توضیح مختصری در مورد RTD می دهیم. RTD اختصار کلمات

**RESISTANCE TEMPERATURE DETECTORS** به معنای حسگرهای مقاومتی دما است. این نوع

حسگرها از جنس فلز یا نیمه هادی می باشند. حسگرهای مقاومتی دمای ساخته شده از نیمه هادیها

اصطلاحاً ترمیستور نامیده می شوند. با افزایش دما مقاومت RTD نیز افزایش می یابد. زیاد شدن فاصله بین

محل نصب حسگرهای دما با ماژول ورودی آنالوگ باعث افزایش طول سیم می شود و بطوری که

مقاومت سیمهای رابط با مقاومت RTD قابل مقایسه می گردد و در نتیجه تغییرات مقاومت سیمهای رابط با

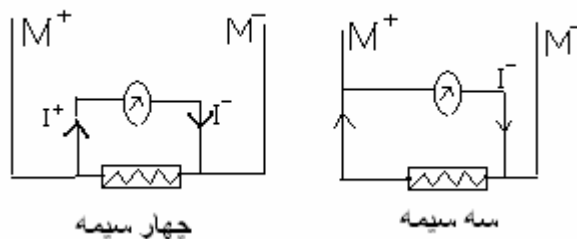
دما منجر به بروز خطاهای قابل توجه در اندازه گیری میشود.

RTDها با مقاومت حدود چند صد اهم ساخته میشوند در حالی که ترمیستورها با مقاومت حدود چندده

کیلو اهم ساخته میشوند. بنابراین اثر سیمهای رابط ترمیستورها منتفی می گردد، در حالی که در RTDها این

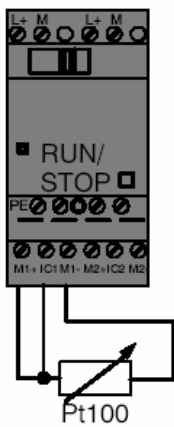
اثر مشکل ساز است، لذا برای حل این مشکل RTDها به صورت سه یا چهار سیمه (شکل زیر)

می سازند.



RTDهای چهار سیمه دارای بالاترین دقت می باشند، زیرا همان طور که در شکل مشاهده می کنید در این RTDها منبع جریان ثابتی به دو سر حسگر اعمال میشود و چون از این مسیر جریان نمی گذرد (چون مقاومت داخلی ولت متر بی نهایت است)، لذا مقاومت سیمهای رابط هیچ نقشی در اندازه گیری ولتاژ ندارند. در نوع سه سیمه چون در یک مسیر عبور جریان و اندازه گیری ولتاژ یکسان می باشد اندازه گیری با کمی خطا همراه خواهد بود اما از آنجا که تنظیم و کالیبراسیون چهار سیمه ها مشکل است معمولا از نوع سه سیمه استفاده می شود.

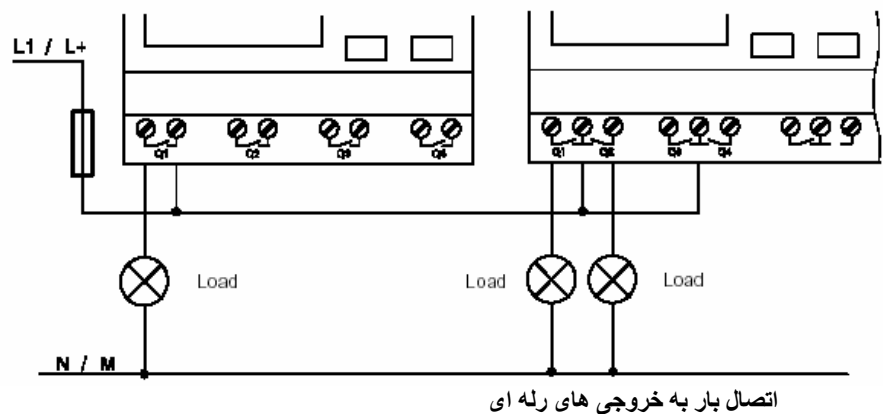
**Pt100** یک نوع RTD است که جنس مقاومت آن از نوع پلاتینیوم بوده و در دمای صفر درجه برابر ۱۰۰ اهم است.



شکل مربوط به سنسور دمای pt100  
به همراه ماژول AM2 PT100

### اتصال خروجیها

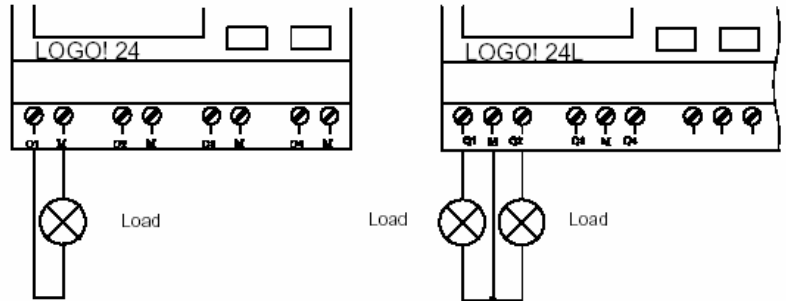
قبل از اتصال خروجیها باید نوع خروجی را تشخیص داد. حرف R نشان دهنده خروجی های رله ای است. عدم درج این حرف نشان دهنده خروجی های ترانزیستوری است. خروجی های رله ای همانند یک کنتاکت عمل می کنند یعنی هر ولتاژی که به هر یک از دو پایه خروجی وصل کنیم، همان ولتاژ را در طرف دیگر خواهیم داشت. خروجی های رله ای این قابلیت را دارند که می توان بارهای مختلف از جمله کنتاکتورها، موتورها و لامپها را مستقیما به آنها وصل کرد.



اتصال بار به خروجی های رله ای

### خروجی های ترانزیستوری

در خروجی های ترانزیستوری در ترمینال خروجی یک ولتاژ مرجع (M) خواهیم داشت. ولتاژ و جریان این خروجیها به ترتیب 24V و 0.3A است.

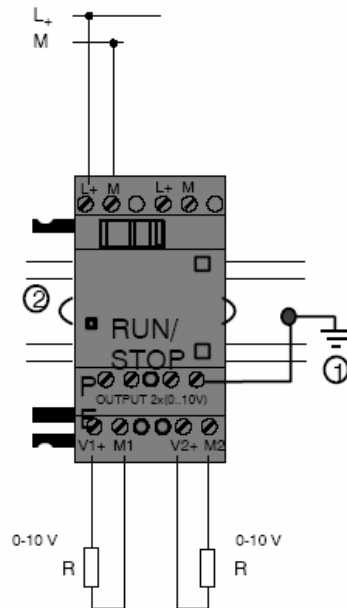


Load: 24 V DC, 0.3 A max.

اتصال بار به خروجی های ترانزیستوری

### خروجی های آنالوگ

LOGO! AM 2 AQ



## فصل سوم

### آشنایی با دستورات LOGO

دو نوع روش برای برنامه نویسی وجود دارد:

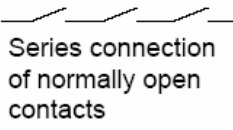
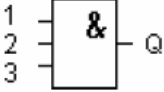
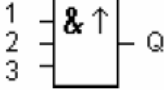
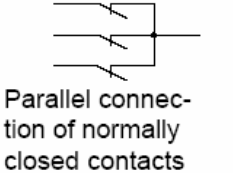
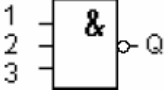
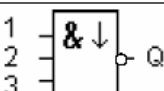
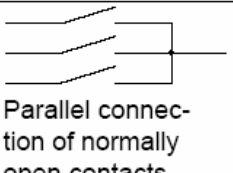
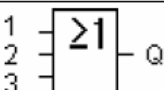
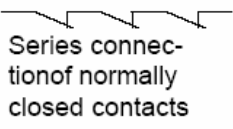
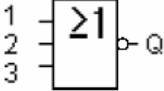
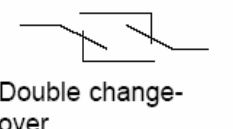
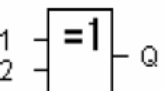
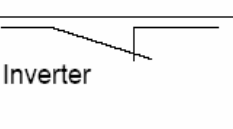
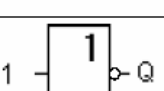
**LAD (ladder diagram)**: که بیشتر برای طراحی مدارات فرمان مورد استفاده قرار می گیرد.

**FBD (function Block diagram)**: برای درک بهتر مسئله مناسب است.

### توابع بیسیک (پایه)

توابع مبنا ( basic function )

منظور از توابع مبنا، توابع منطقی مانند AND, OR, XOR و ..... می باشد. که شرح آن به صورت زیر است.

Circuit diagram representation	Representation in LOGO!	Designation of the basic function
 <p>Series connection of normally open contacts</p>		<p>AND</p> <p>(see page 67)</p>
		<p>AND with RLO edge detection</p> <p>(see page 67)</p>
 <p>Parallel connection of normally closed contacts</p>		<p>NAND (AND not)</p> <p>(see page 68)</p>
		<p>NAND with RLO edge detection</p> <p>(see page 69)</p>
 <p>Parallel connection of normally open contacts</p>		<p>OR</p> <p>(see page 67)</p>
 <p>Series connection of normally closed contacts</p>		<p>NOR (OR not)</p> <p>(see page 70)</p>
 <p>Double change-over</p>		<p>XOR (exclusive OR)</p> <p>(see page 71)</p>
 <p>Inverter</p>		<p>NOT (negation, inverter)</p> <p>(see page 71)</p>

**: AND**

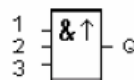
مشابه با اتصال سری در یک حلقه است که خروجی آن وقتی ۱ است که تمام ورودیها ۱ باشد.



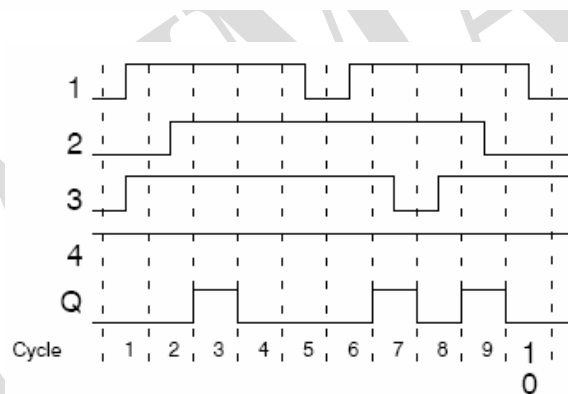
ورودی X (ورودی بی اهمیت و یا متصل نشده) در این بلاک ۱ در نظر گرفته میشود. جدول منطقی آن به شکل زیر است.

1	2	3	Q
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

AND تحریک شده با لبه بالا رونده :



خروجی فقط هنگامی ۱ است که همه ورودیها ۱ بوده و حداقل یکی از ورودیها در سیکل قبل ۰ بوده باشد. ورودی X (ورودی بی اهمیت و یا متصل نشده) در این بلاک ۱ در نظر گرفته میشود. نمایش سیکلهای زمانی به صورت زیر می باشد.



**: NAND**

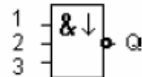
خروجی فقط وقتی ۰ است که همه ورودیها ۱ باشد. ورودی X (ورودی بی اهمیت و یا متصل نشده) در این بلاک ۱ در نظر گرفته میشود. جدول منطقی آن به شکل زیر است



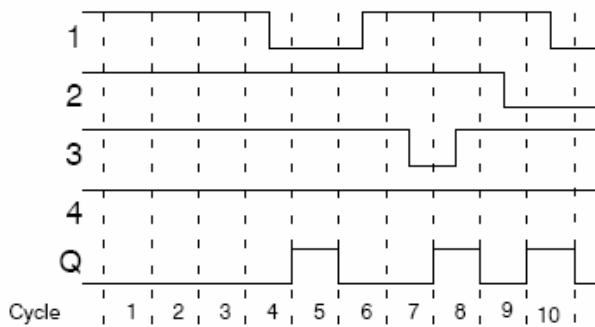


1	2	3	4	Q
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

NAND تحریک شده با لبه پایین رونده :



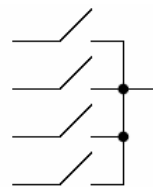
خروجی فقط هنگامی ۱ است که حداقل یکی از ورودیها ۰ بوده و همه ورودیها در سیکل قبل ۱ بوده باشد .  
 ورودی X ( ورودی بی اهمیت و یا متصل نشده ) در این بلاک ۱ در نظر گرفته میشود . نمایش سیکلهای زمانی  
 به صورت زیر می باشد .



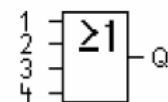
: OR

مشابه با اتصال موازی در یک حلقه است که خروجی آن وقتی ۰ است که تمام ورودیها ۰ باشد و خروجی آن  
 وقتی ۱ است که حداقل یکی از ورودیها ۱ باشد .

1	2	3	4	Q
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1



Symbol in LOGO!:



**: NOR**

خروجی فقط وقتی ۱ است که همه ورودیها ۰ باشد. ورودی X (ورودی بی اهمیت و یا متصل نشده) در این بلاک ۰ در نظر گرفته میشود. در تغییر ۰ به ۱ در یکی از ورودیها، خروجی ۰ میشود. جدول منطقی آن به شکل زیر است

1	2	3	Q
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

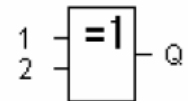


**: XOR**

خروجی فقط وقتی ۱ است که ورودیها غیر یکسان باشد. ورودی X (ورودی بی اهمیت و یا متصل نشده) در این بلاک ۰ در نظر گرفته میشود. جدول منطقی آن به شکل زیر است:

1	2	Q
0	0	0
0	1	1
1	0	1
1	1	0

Symbol in LOGO!:

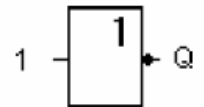


**NOT (Negation, Inverter)**

این تابع معکوس کننده ورودی است، یعنی ۰ را به ۱ و ۱ را ۰ تغییر می دهد جدول منطقی آن به شکل زیر است:

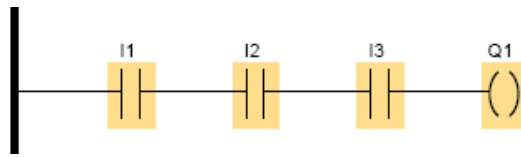
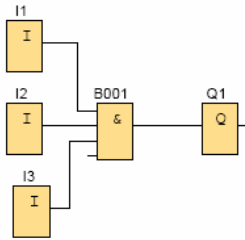
1	Q
0	1
1	0

Symbol in LOGO!:

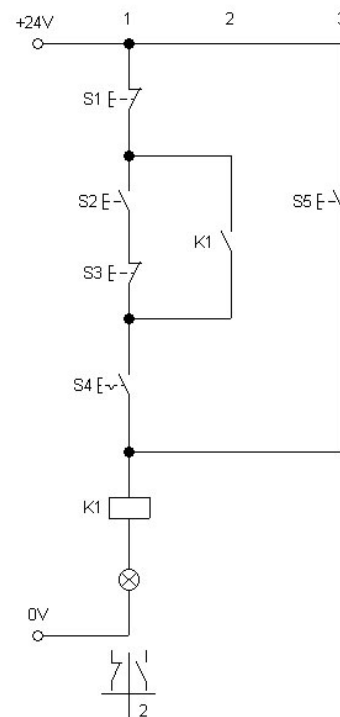
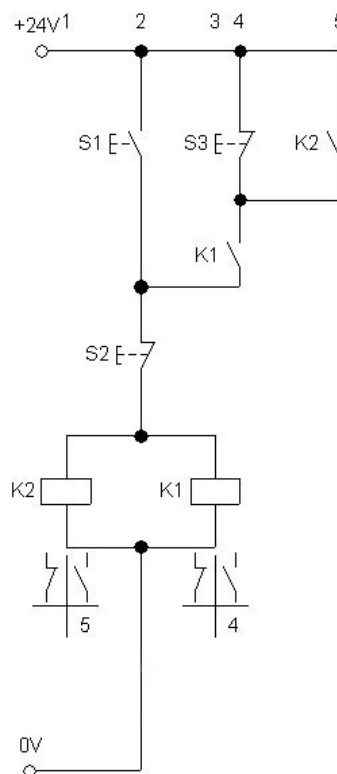
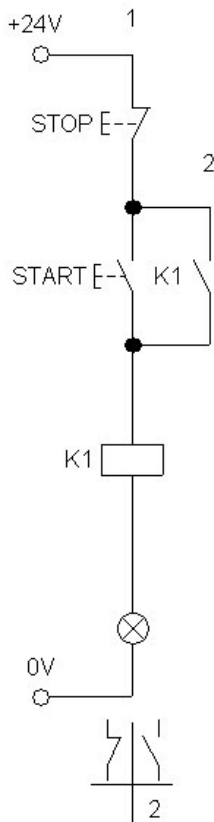
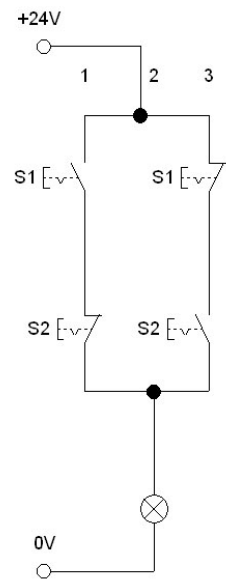
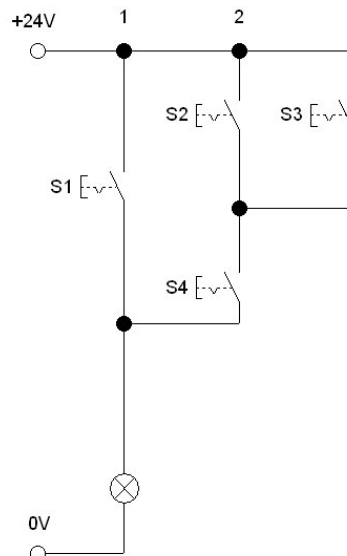
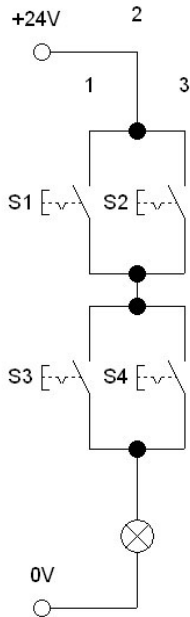
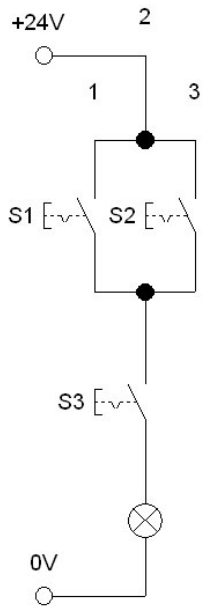


مثال: یک سیستم دارای سه ورودی است. دستگاه هنگامی کار می کند که هر سه ورودی

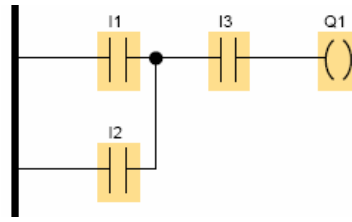
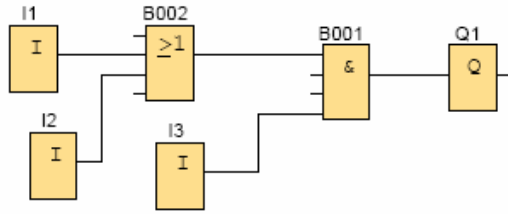
فعال باشند. برنامه مربوطه را به زبانهای lad و FBD بنویسید.



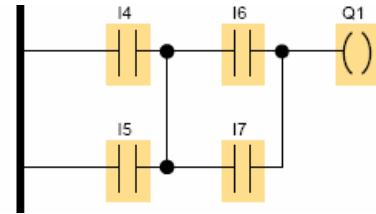
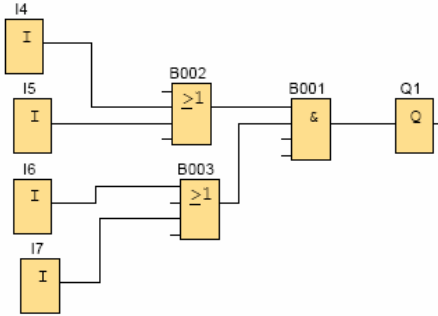
تمرین: برنامه مدارهای زیر را به هر دو زبان LAD, FBD بنویسید.



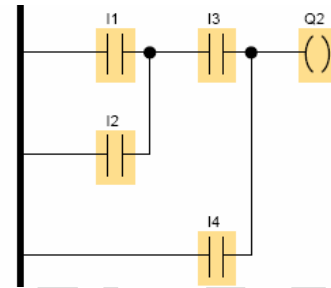
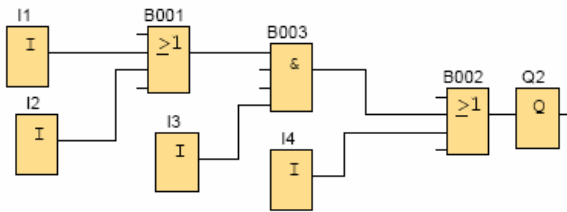
طعماسد



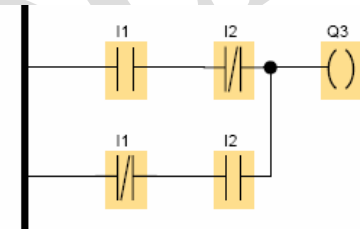
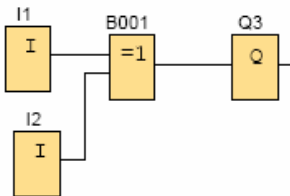
(۱) از طرف سمت چپ



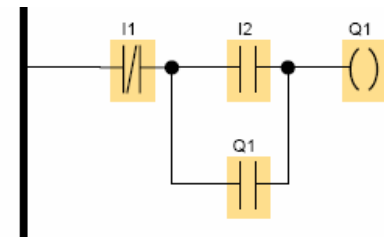
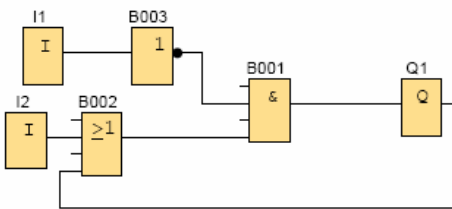
(۲)



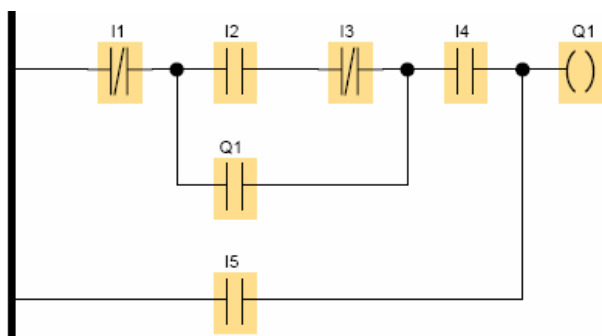
(۳)



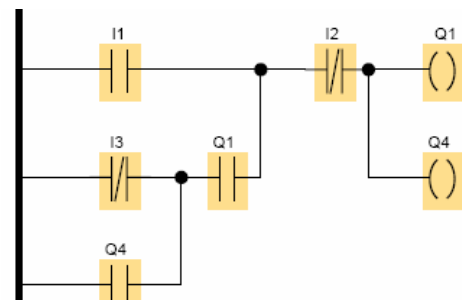
(۴)



(۵)

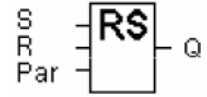
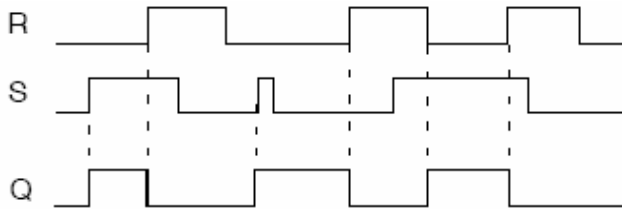


(۶)



(۷)

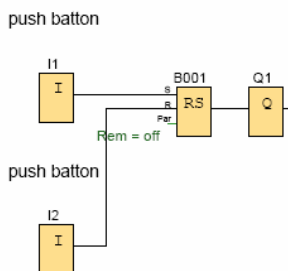
## رله نگهدارنده Latching Relay



خروجی با لبه بالا رونده S فعال می شود و با لبه پایین رونده S خاموش نمی شود مبادا اینکه ورودی R فعال شود. اگر گزینه Retentivity را فعال کنید در این صورت وقتی برق قطع شد با وصل مجدد برق، وضعیت رله در همان حالت قبلی باقی می ماند.

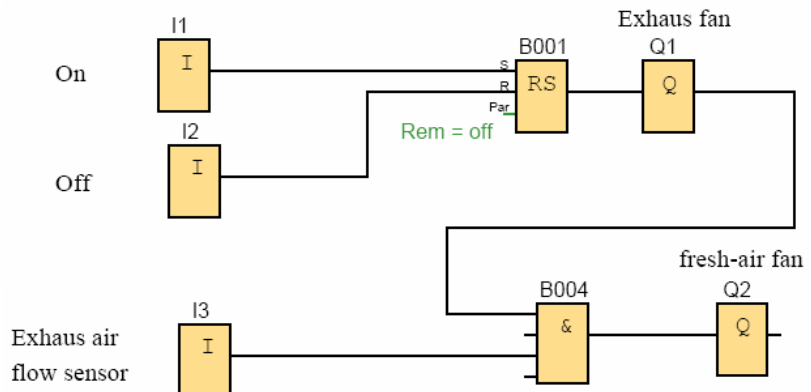
اگر هر دو ورودی با هم فعال شوند در این حالت R بر S ارجحیت خواهد داشت.

مثال: مدار شماره ۵ را با SR بنویسید.

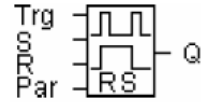
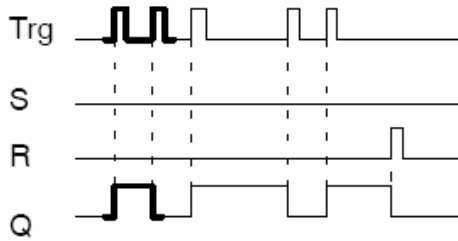


تمرین: سیستم تهویه هوای اتاق

اتاقی مجهز به یک فن تخلیه هوا EXHAUST و یک فن هوای تازه FRESH-AIR FAN می باشد. فن تخلیه توسط شستی ON روشن می شود و فن هوای تازه زمانی روشن می شود که قبل از آن فن تخلیه به همراه سنسور جریان تخلیه هوا فعال شده باشد. سیستم با زدن شستی OFF خاموش می شود.



### رله پالسی Pulse Relay



در این رله، خروجی با لبه بالا رونده پالس اول روشن و با لبه بالا رونده پالس دوم خاموش می شود. این رله همانند کلید start و stop عمل می کند. اگر پایه Trg فعال نباشد رله مثل فلیپ فلاپ RS (البته با ویژگی ارجحیت دادن) عمل می کند.

تمرین: برنامه مدارات فرمان زیر را بنویسید.

**لازم به یادآوری است که معمولاً برنامه مدارات فرمان به روش LAD نوشته می شوند.**

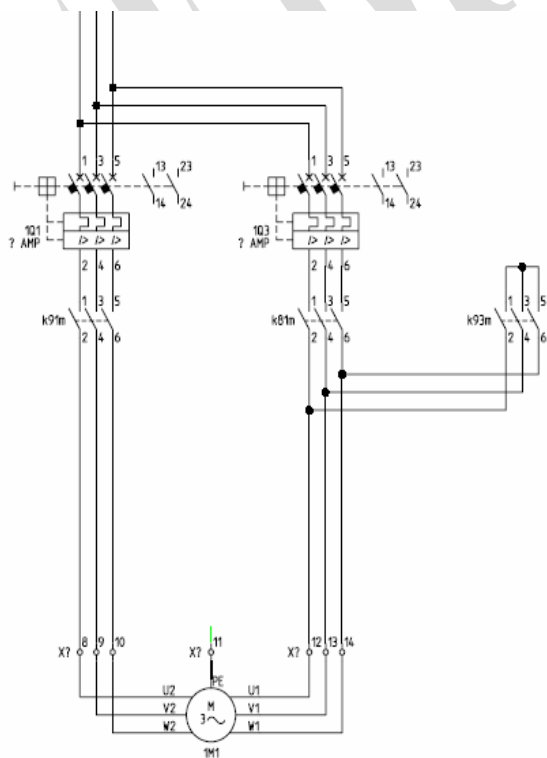
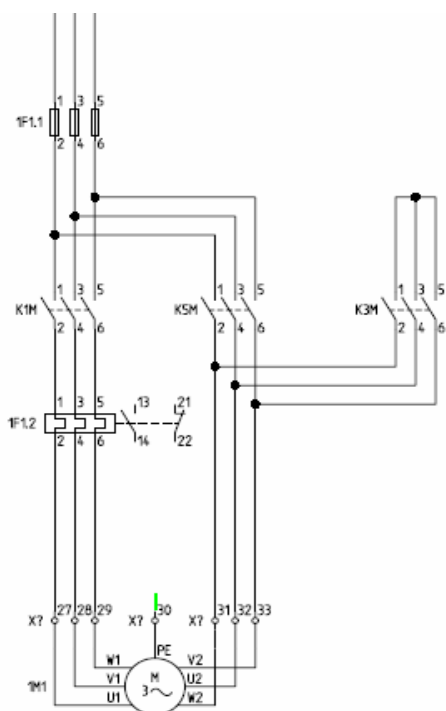
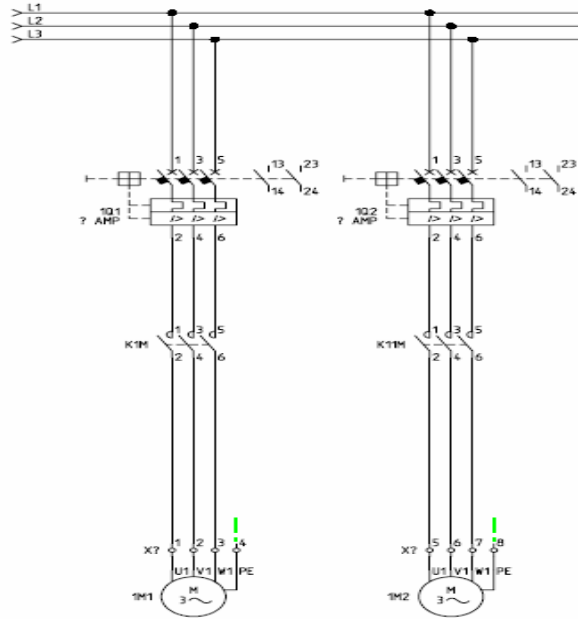
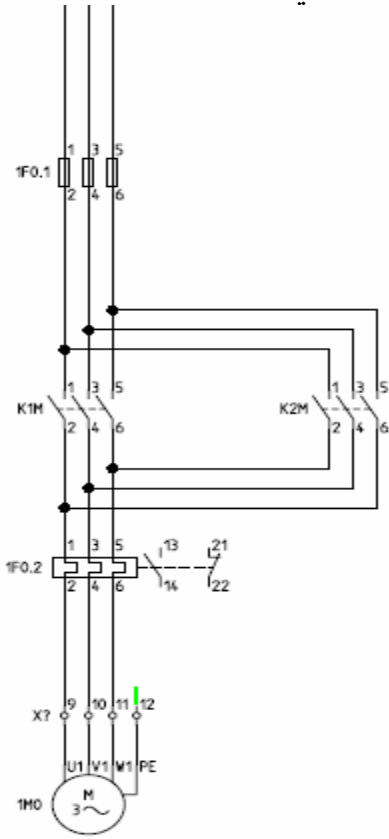
- ۱- راه اندازی الکترو موتور سه فاز روتور قفسه ای بصورت دائم کنترل از دو محل
- ۲- راه اندازی دو الکترو موتور سه فاز به صورت یکی به جای دیگری به صورت دستی
- ۳- راه اندازی دو الکترو موتور سه فاز به صورت یکی پس از دیگری صورت دستی
- ۴- راه اندازی الکترو موتور سه فاز روتور قفسه ای بصورت چپگرد و راستگرد دور کند
- ۵- راه اندازی الکترو موتور سه فاز روتور قفسه ای بصورت چپگرد و راستگرد دور تند
- ۶- راه اندازی الکترو موتور سه فاز روتور قفسه ای بصورت ستاره و مثلث
- ۷- راه اندازی الکترو موتور سه فاز به صورت دالاندر
- ۸- راه اندازی الکترو موتور سه فاز روتور قفسه ای بصورت لحظه ای و دائم کنترل از دو محل

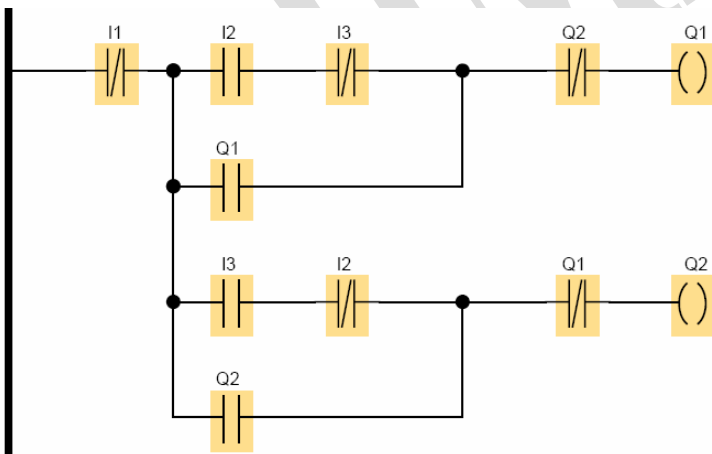
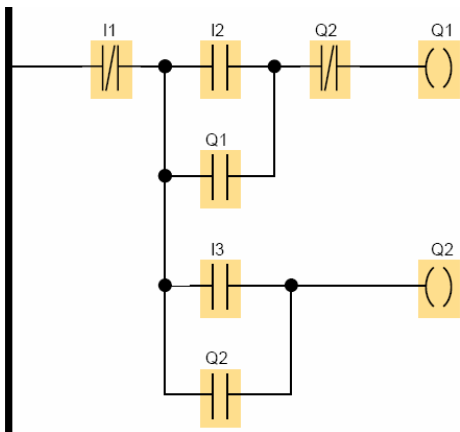
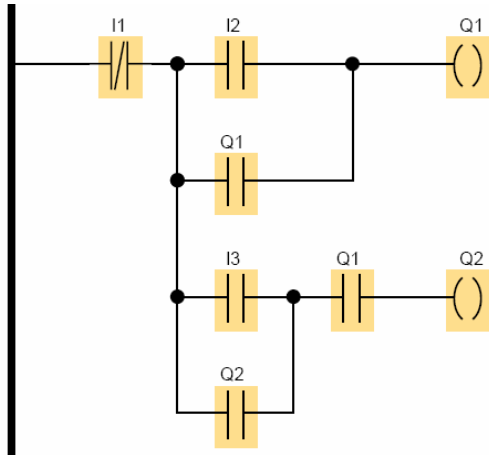
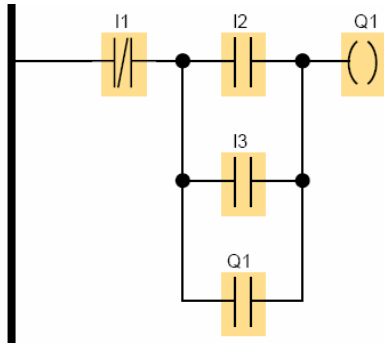
**با وجود plc، مدار فرمان حذف می شود، ولی مدار قدرت بر سر جای خود باقی است. یعنی فرمان کنترل را بوسیله plc انجام خواهیم داد.**

در مدارات فرمان رله کنتاکتوری، باز یا بسته بودن کنتاکتهای مدار قدرت را، مدار فرمان تعیین می کرد.

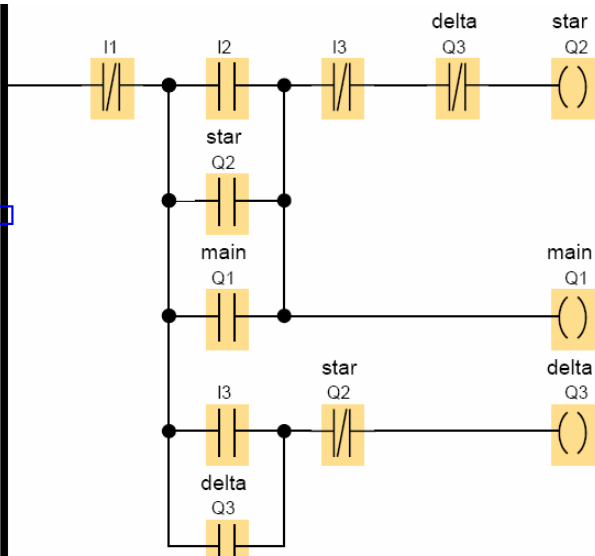
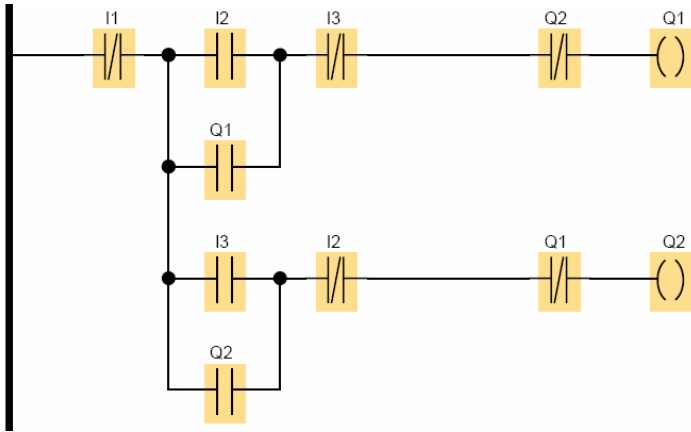
که مدار فرمان اینکار را با برق دار یا غیر برق دار کردن بوبین کنتاکتور ،انجام می داد. اکنون در حالت اتوماسیون اینکار را با وصل کردن خروجی های logo (البته مدل های رله ای) به بوبین کنتاکتورها ،انجام خواهیم داد. در فصل قبل نیز با نحوه سیم بندی این خروجیها آشنا شده اید.

مدارات  
قدرت

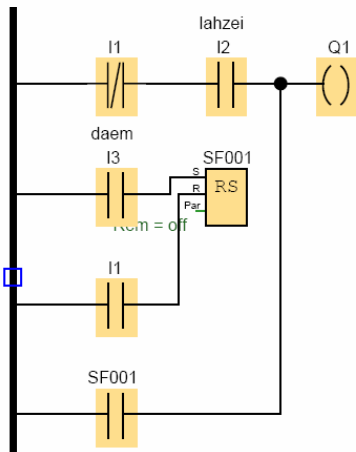
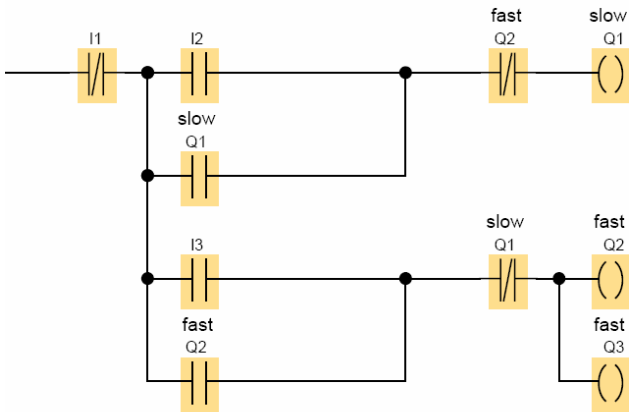








dahlander



## SF (SPECIAL FUNCTION): توابع ویژه

همانطور که پیش از این گفته شد، هنگام برنامه ریزی LOGO، علاوه بر توابع مبنا از توابع ویژه نیز می توان استفاده نمود. ابتدا اتصالات و یا پارامترهایی که به سایر ماژولها و یا ورودیها متصل میشود را تعریف کرده و سپس لیست این توابع آورده میشود و بدنبال آن مشروح هر یک نیز می آید.

- **S ( set )** : می تواند برای یک کردن خروجی استفاده شود.
  - **R ( reset )** : این ورودی نسبت به سایر ورودیها تقدم و ارجحیت داشته و برای صفر کردن خروجی استفاده میشود.
  - **TRG ( trigger )** : این ورودی برای تحریک و آغاز به کار سیکل تابع استفاده می شود
  - **CNT ( count )** : این ورودی برای شمارش پالسها استفاده میشود
  - **DIR ( direction )** : این ورودی برای تعیین جهت شمارش ( صعودی یا نزولی بودن ) استفاده میشود.
  - **EN ( enable )** : این ورودی بلاک را فعال می نماید. وقتی این ورودی  $\bar{\phantom{x}}$  باشد، بلوک تمام سیگنالهای دیگر را ندیده می انگارد.
  - **INV ( invert )** : هنگامی که این ورودی set باشد خروجی بلوک معکوس می گردد.
  - **RAL ( reset all )** : تمام مقادیر داخلی reset میشوند.
  - **PAR ( parameter )** : از این ورودی برای تنظیم پارامترها استفاده میشود.
  - **T ( time )** : از این ورودی برای تنظیم زمان استفاده میشود.
- لیست بلوک توابع ویژه بصورت زیر است :

View in LOGO!	Name of the special function	Rem
<b>Times</b>		
	On-delay (see page 135)	REM
	Off-delay (see page 139)	REM
	On-/Off-delay (see Page 141)	REM
	Retentive on-delay (see page 143)	REM
	Wiping relay (pulse output) (see page 145)	REM

	<p>Edge triggered wiping relay (see page 147)</p>	REM
	<p>Asynchronous pulse generator (see Page 150)</p>	REM
	<p>Random generator (see page 152)</p>	
	<p>Stairway lighting switch (see page 154)</p>	REM
	<p>Multiple function switch (see page 157)</p>	REM
	<p>Weekly timer (see Page 160)</p>	
	<p>Yearly timer (see Page 165)</p>	

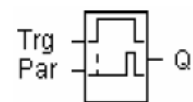
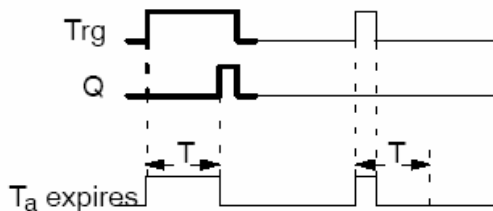
<b>Counter</b>		
	<p>Up/down counter (see Page 168)</p>	REM
	<p>Hours counter (see page 172)</p>	REM
	<p>Threshold trigger (see Page 177)</p>	

<b>Analog</b>		
	<p>Analog threshold trigger (see page 180)</p>	
	<p>Analog differential trigger (see page 183)</p>	
	<p>Analog comparator (see Page 187)</p>	
	<p>Analog value monitoring (see Page 192)</p>	
	<p>Analog amplifier (see Page 195)</p>	

	Analog multiplexer (see Page 216)	
	Analog ramp (see Page 220)	
	PI controller (see Page 225)	REM
<b>Miscellaneous</b>		
	Latching relay (see Page 198)	REM
	Pulse relay (see Page 199)	REM
	Message texts (see Page 202)	
	Softkey (see Page 209)	REM
	Shift register (see Page 213)	REM

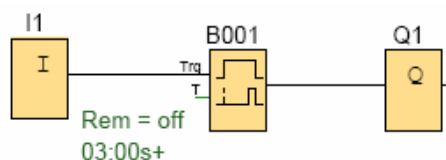
## تایمرها TIMERS

### تایمر تاخیر در وصل On-delay

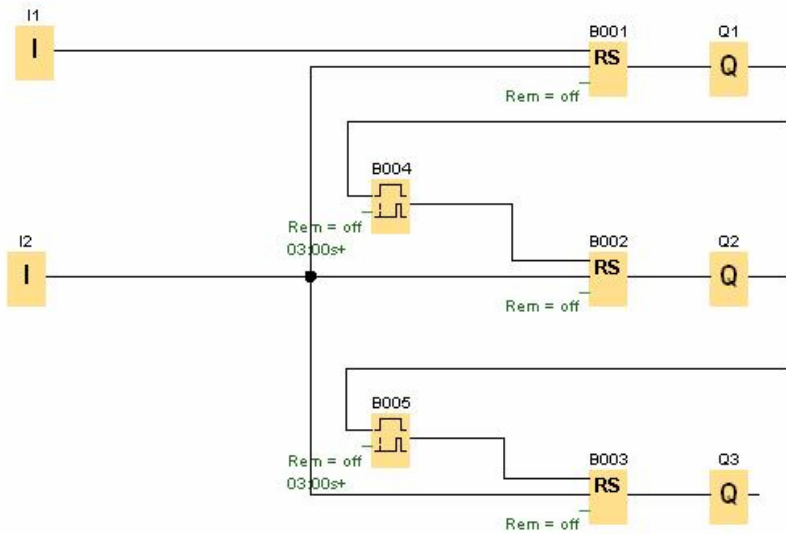


در این تابع خروجی زمانی فعال می شود که زمان تعریف شده  $T_a$  بعد از فعال شدن  $Trg$  سپری شده باشد. همچنین خروجی با لبه پایین رونده ورودی غیر فعال می شود.

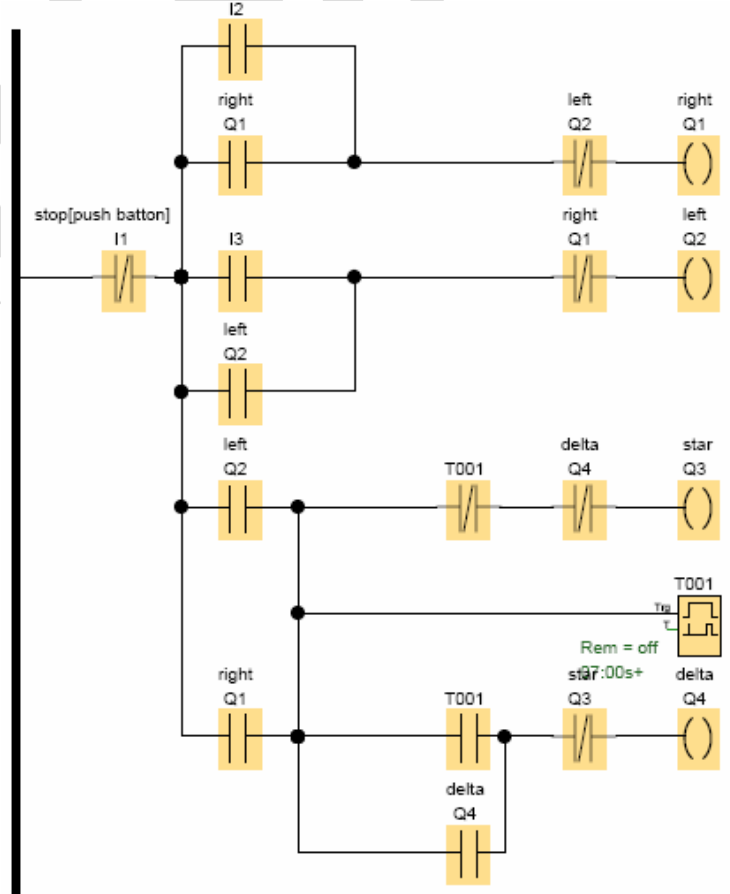
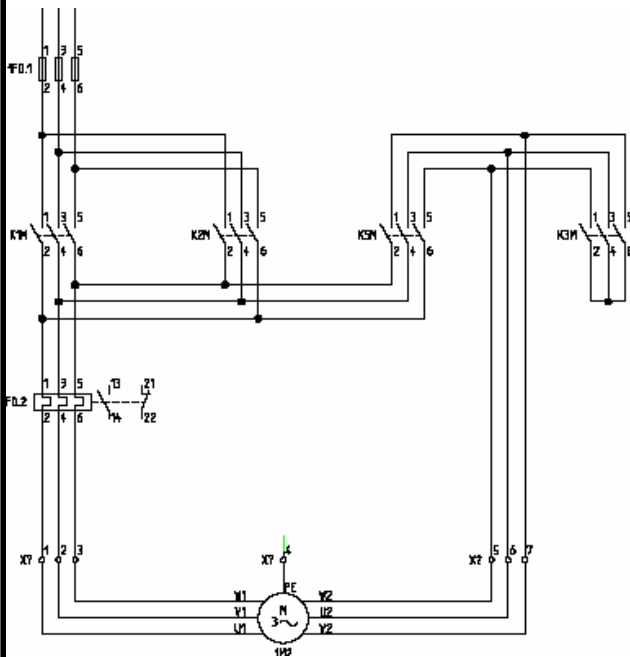
مثال: با برنامه ای بنویسید که  $Q_2$ ، ۳ ثانیه بعد از فعال شدن ورودی  $I_2$  روشن شود.



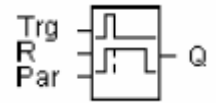
تمرین: با زدن Start ، ابتدا Q1 روشن و پس از 2S ، Q2 ، نیز روشن و پس از 2S ، Q3 نیز روشن می شود. هرگاه Stop زده شد همگی خاموش می شوند.



تمرین: برنامه راه اندازی موتور سه فاز بصورت ستاره و مثلث توام با راستگرد و چپگرد را بنویسید. مدت زمان تغییر از ستاره به مثلث را ۷ ثانیه در نظر بگیرید.

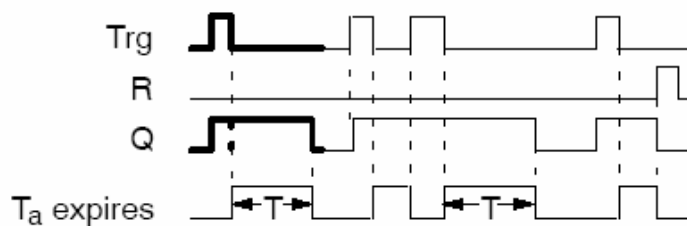


## تایمر تاخیر در خاموشی off-delay



در این تابع، خروجی با لبه بالا رونده ورودی  $Trg$  روشن می شود زمان  $T$  نیز با لبه پایین رونده  $Trg$  شروع به شمارش می کند. خروجی تا زمانیکه زمان تنظیم شده  $Ta$  سپری نشده روشن باقی می ماند. شکل زیر دیاگرام عملکرد این تابع را نشان می دهد.  
خروجی با لبه بالا رونده  $R$  خاموش می شود.

### Timing diagram



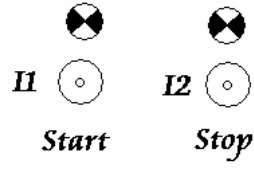
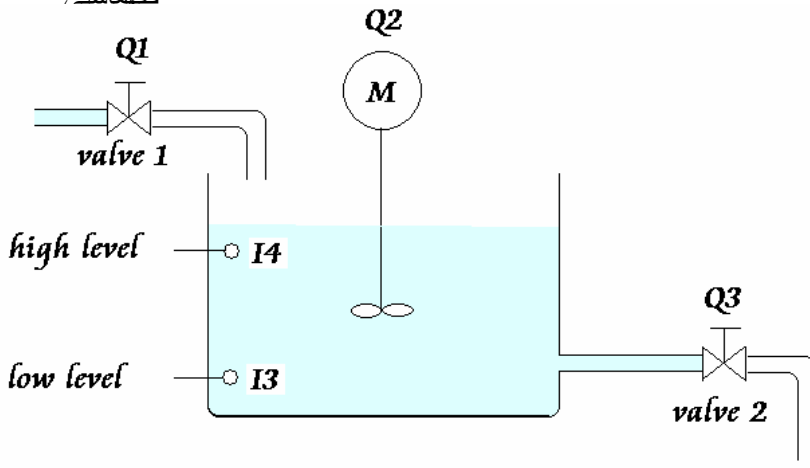
مثال: برنامه ای بنویسید که خروجی، ۳ ثانیه بعد از فعال شدن ورودی به مدت ۵ ثانیه روشن مانده سپس خاموش گردد.

### تمرین: mixer

شرح برنامه:

با زدن شستی استارت شیر ۱ باز و شیر ۲ بسته میشود تا مایع وارد مخزن شود. وقتی مخزن به اندازه high level پر شد شیر ۱ غیر فعال شده، موتور همزن روشن می شود و به مدت 10 s میکسر را می چرخاند. سپس شیر ۲ باز شده مخزن را خالی می کند. زمانیکه سنسور Low Level غیر فعال شد، شیر دو نیز بسته می شود. هر وقت هم Stop زده شد سیستم متوقف شود. یک لامپ هم وضعیت سیستم را از نظر Start یا Stop بودن نشان دهد.

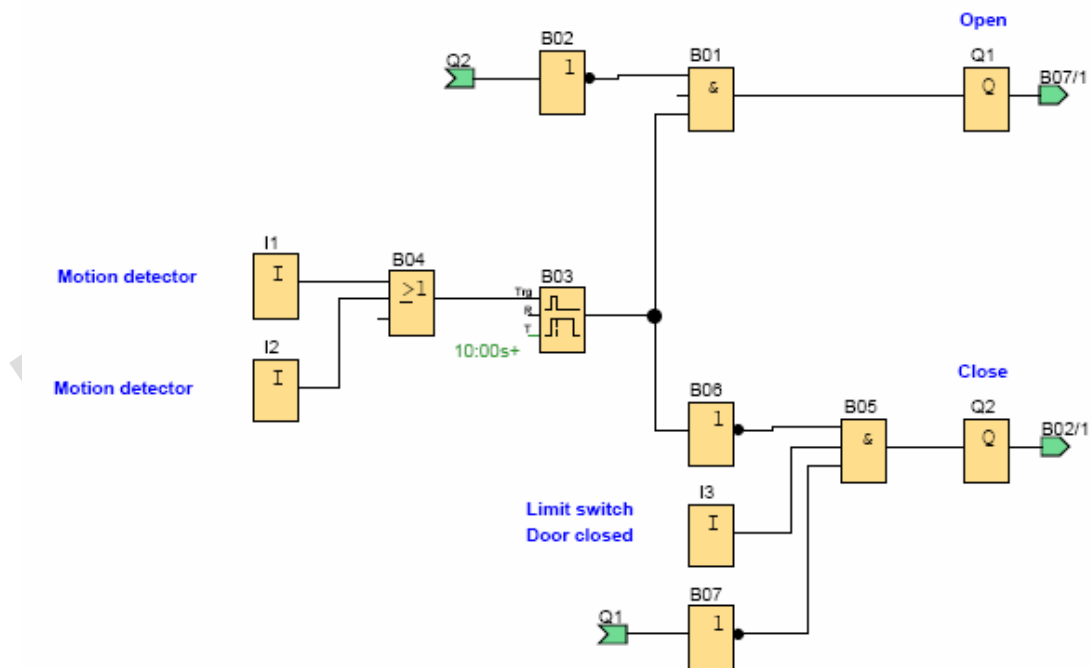
طعماسد



TAHMASEBI

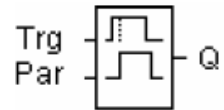
### تمرین: سیستم درهای اتوماتیک بانکها و فروشگاهها

دو آشکار ساز حرکتی I1 و I2 یکی در داخل و دیگری در خارج، نزدیک شدن اشخاص به در را، حس می کنند و یک پالس را به LOGO ارسال می کنند. در همین حال در به مدت 5 ثانیه بطور اتوماتیک باز می شود و اگر قبل از اتمام 5 ثانیه شخص دیگری وارد شود بایستی این زمان دوباره تمدید شود. اگر شخصی در لای در بماند لیمیت سویچ تعبیه شده حضور وی را به LOGO گزارش می دهد و تا شخص از در فاصله نگرفته در بسته نمی شود تا به وی آسیبی نرسد (لازم به یادآوری است که در حالت عادی درب بسته است). برنامه پروسه فوق را بنویسید.

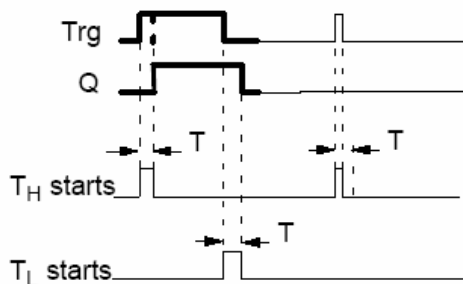




## تایمر تاخیر در وصل و قطع ON/OFF-DELAY



این تابع یک تایمر تاخیر در وصل و تاخیر در قطع می باشد. در این تابع خروجی بعد از سپری شدن اولین زمان  $T_H$  روشن می شود و بعد از سپری شدن دومین زمان تنظیم شده  $T_L$  خاموش می شود. زمان  $T_H$  با لبه بالا رونده و زمان  $T_L$  با لبه پائین رونده ورودی  $Trg$  شروع به شمارش می کند. شکل زیر دیاگرام عملکرد این تابع را نشان می دهد.

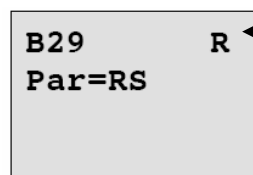


$T_H$  = زمان تاخیر در وصل

$T_L$  = زمان تاخیر در قطع

توضیحات پنجره تایمر:

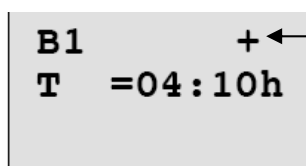
**RETENTIVE**: با انتخاب این گزینه داده های جاری بر روی بلوک تابع ویژه بعد از قطع برق تغییر نمی کند. در هنگام برنامه نویسی توسط سخت افزار نیز برای حفظ اطلاعات طبق شکل زیر عمل کنید.



حرف **R** نشاندهنده **RETENTIVE** بودن است.

☺ در ورژن های قدیمی، فقط کانتورها دارای این گزینه هستند.

**PROTECTION ACTIVE**: با انتخاب این گزینه به اپراتور این اجازه را نمی دهیم که در زمان اجرای برنامه وارد پارامتر برنامه شود و مقادیر بلوک زمانها و .. را تغییر بدهد. در سخت افزار این کار بوسیله علامت (-) صورت می گیرد. با انتخاب علامت (+) که پیش فرض دستگاه است کاربر می تواند پارامترها را بدون متوقف کردن برنامه از منوی SET PARM تغییر دهد.



**دقت زمان T**: تلرانس در تایمرها برابر  $+5$  - ثانیه در یک ۲۴ ساعت می باشد.

## باتری BACK UP

در یک LOGO بعد از قطع برق باید اطلاعات بدون هیچ تغییری ثابت باقی بمانند تا در زمان استارت مجدد، برنامه به درستی اجرا شود. مقدار زمانی که LOGO می تواند بعد از قطع برق بوسیله باتری های داخلی اطلاعات یک برنامه را بدون تغییر نگهدارد ۸۰ ساعت است. اگر قبل از اتمام این زمان برق logo وصل نشود، در آن صورت اطلاعات از روی حافظه پاک می شوند.

البته اصل برنامه بعد از گذشت این زمان از بین نمی رود، بلکه مقادیری که در زمان اجرا متغیر بوده (مانند مقادیر کانتورها و تایمرها و تاریخ سفت افزار) و از حافظه های پایدار Retentive استفاده کرده اند، رست می شوند.

تمرین: کنترل نوار نقاله جهت تخلیه بار

مدار بایستی طبق خواسته های زیر عمل کند:

۱- با زدن شستی I1 مدار به مدت ۵ ثانیه شروع به بارگیری نماید.

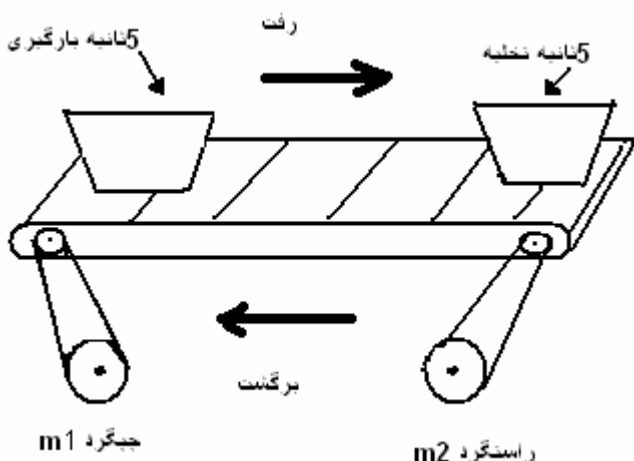
۲- بعد از اتمام ۵ ثانیه بارگیری، موتور M2 برای انتقال بار فعال می شود و همزمان با آن ۲۰ ثانیه رفت نیز آغاز می شود.

۳- بعد از سپری شدن ۲۰ ثانیه رفت، موتور M2 خاموش شده و همزمان با آن زمان ۵ ثانیه برای تخلیه بار شروع می شود. سپس بعد از اتمام ۵ ثانیه موتور M1 برای برگشت واگن شروع به کار می کند و بعد از سپری شدن ۲۰ ثانیه برگشت، موتور M1 خاموش شده و برای اجرای سیکل بعدی، زمان ۵ ثانیه برای بارگیری آغاز می شود.

۴- با زدن شستی I2 کل مدار غیر فعال شده و عملیات متوقف می شود.

۵- هرگاه دو شستی I1, I2 به هر علتی فعال شوند در آن صورت نباید مدار عملیات خود را آغاز کند

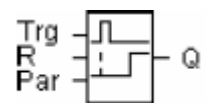
۶- یک چراغ چشمک زن به معنای فعال بودن سیستم روشن شود.



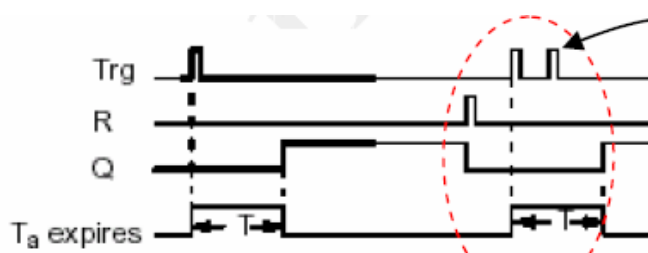
I1 → START  
I2 → STOP  
Q2 → M1  
Q1 → M2  
Q3 → START LIGHT

TAHMASEBI

### RETENTIVE ON DELAY

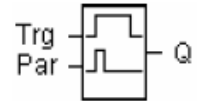


همانند تایمر ON DELAY است با این تفاوت که به لبه پایین رونده ورودی حساس نیست. همچنین پایه RESET نیز دارد. شکل زیر دیاگرام عملکرد این تابع را نشان می دهد.



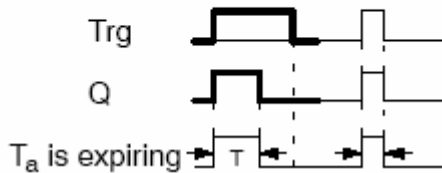
کلیدزنی مجدد که در فاصله زمانی کمتر از T رخ داده است

### خروجی پالسی Wiping relay(Pulse out put)



در این تابع، با فعال شدن ورودی  $Trg$ ، خروجی به اندازه زمان تعیین شده فعال می شود. اگر ورودی  $Trg$  قبل از سپری شدن زمان  $Ta$  صفر شود در این صورت خروجی فوراً غیرفعال می شود و اگر دوباره فعال شود زمان  $Ta$  نیز از صفر شروع به شمارش خواهد کرد.

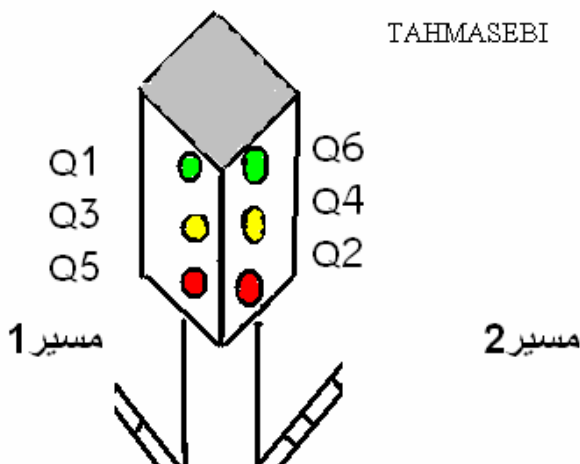
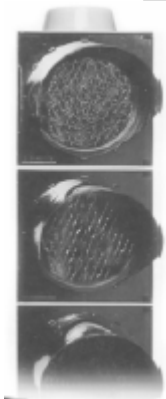
#### Timing diagram



#### تمرین: پروسه چراغ راهنمایی

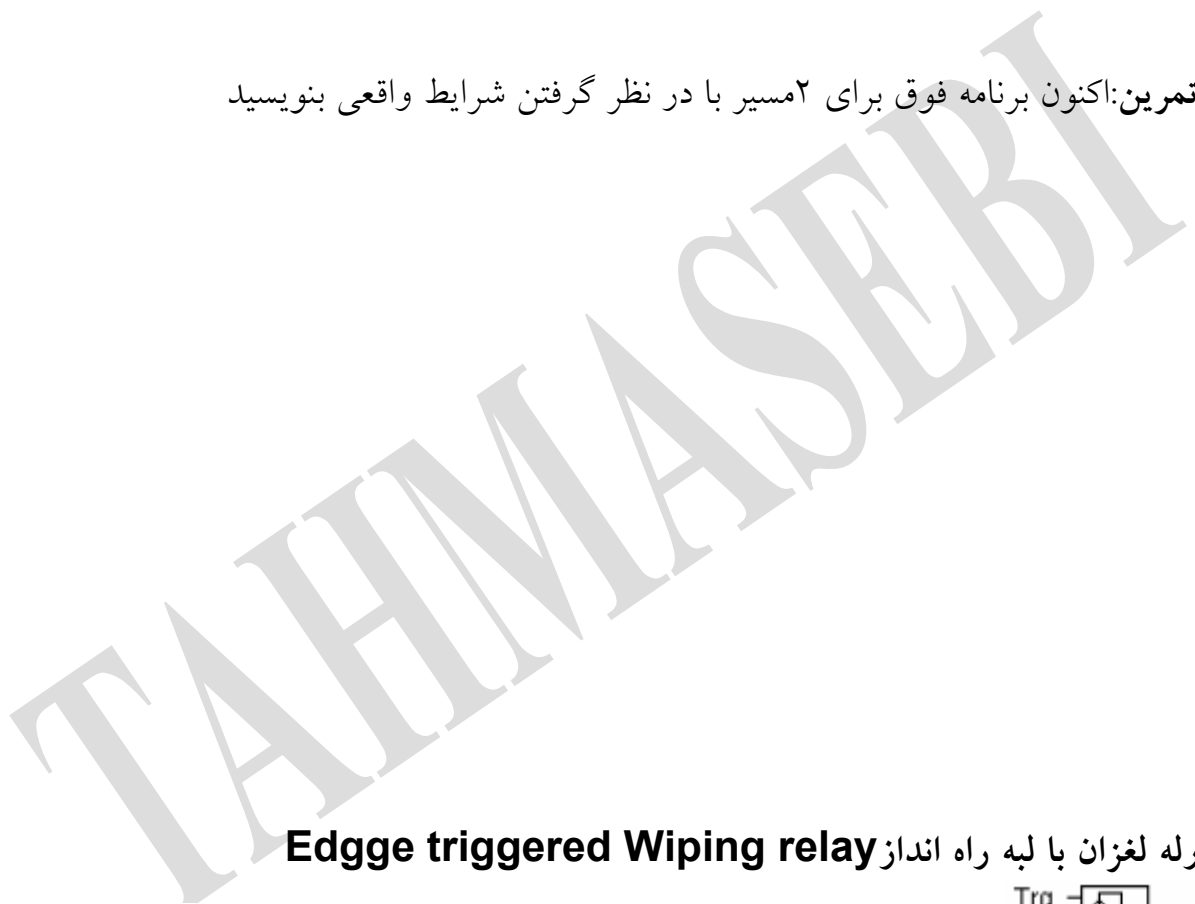
شرایط برنامه:

- ۱- هنگامی که کلید فعال می شود سیستم کنترل شروع به کار نماید.
- ۲- زمانیکه چراغ سبز برای مسیر ۱ روشن است چراغ قرمز برای مسیر ۲ روشن باشد.
- ۳- زمانیکه چراغ زرد برای مسیر ۱ روشن است چراغ زرد برای مسیر ۲ روشن باشد. (برای راحتی مسئله این طور فرض شده است)
- ۴- زمانیکه چراغ سبز برای مسیر ۲ روشن است چراغ قرمز برای مسیر ۱ روشن باشد
- ۵- مدت زمان روشن ماندن چراغهای سبز و قرمز ۳۰ ثانیه و زرد ۵ ثانیه باشد.

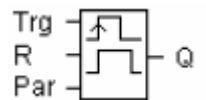
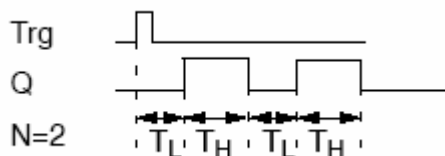


Q1	سبز مسیر ۱
Q2	قرمز مسیر ۲
Q3	زرد مسیر ۱
Q4	زرد مسیر ۲
Q5	قرمز مسیر ۱
Q6	سبز مسیر ۲

تمرین: اکنون برنامه فوق برای ۲ مسیر با در نظر گرفتن شرایط واقعی بنویسید



**رله لغزان با لبه راه انداز Edge triggered Wiping relay**



این تایمر مشابه تایمر on/off است با این تفاوت که :

اولاً، اینکار به اندازه N بار صورت می گیرد.

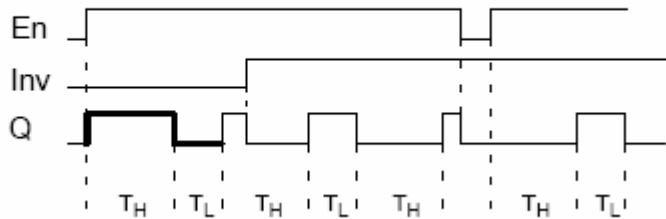
ثانیاً، فقط به لبه بالا رونده حساس بوده و روشن و خاموش شدن خروجی پشت سر هم انجام می شود.

وقتی که زمان TL به پایان رسید خروجی Q فعال می شود و همچنین بعد از سپری شدن زمان TH غیر فعال می شود.

\*در سخت افزارهای سری OBA2,OBA3 فقط پارامتر TH موجود می باشد. همچنین ورودی R نیز موجود نیست. اگر  $N=1$  باشد، از این تایمر می توان به عنوان تاخیر در خاموش مستقل از ورودی استفاده کرد.

### مولد پالس غیر همزمان یا آسنکرون Asynchronous pulse generator

Timing diagram

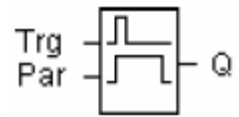


TH: عرض پالس

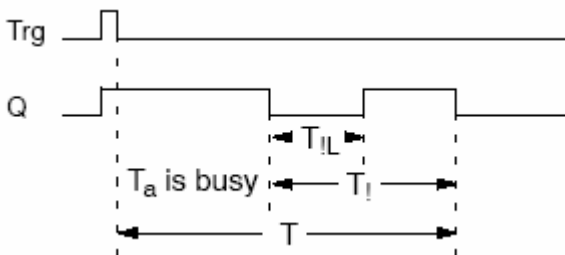
TL: عرض وقفه پالس

برای معکوس کردن عملیات (یعنی تعویض زمان TH با TL) می توانید ورودی INV را فعال کنید.

### کلید روشنایی راه پله Stairway lighting switch



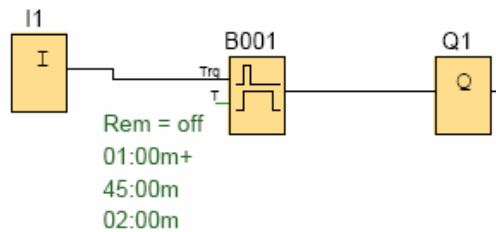
خروجی با لبه بالا رونده پالس ورودی Trg فعال می شود و بعد از سپری شدن زمان T خاموش می شود. لازم به توضیح است که زمان کل با لبه پایین رونده Trg فعال می شود و هر بار که ورودی را فعال کنیم، زمان از صفر شروع به شمارش می کند، یعنی مبنای زمان آخرین لبه پالس ورودی است.



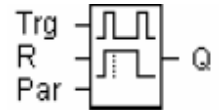
Pre-warning time (T!): زمانی است که قبل از اتمام زمان کل، خروجی هشدار می دهد.

Pre-warning period(TL!): زمان خاموشی بعد از هشدار

مثال: اگر کل زمان روشن ماندن خروجی، دقیقه باشد و زمان  $(T!) = 5$  ثانیه و  $(TL!)$  را برابر ۲ ثانیه فرض کنیم در آن صورت خروجی در ثانیه ۴۵ ام به مدت ۲ ثانیه خاموش شده سپس روشن می شود تا اینکه کل زمان دقیقه تمام شود.



### کلید با عملکرد چند گانه Multiple function switch



این تابع، یک تایمر چندکاره می باشد:

۱- به عنوان رله ضربه ای

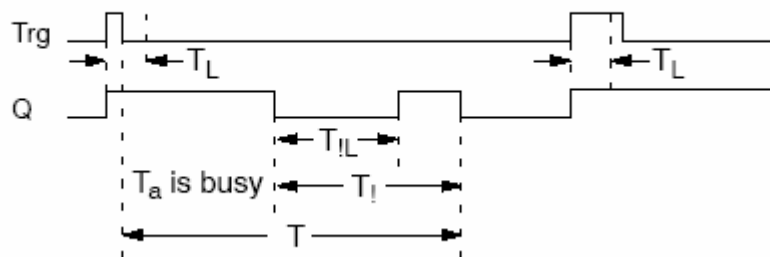
۲- تایمر راه پله

۳- به عنوان ست و رست

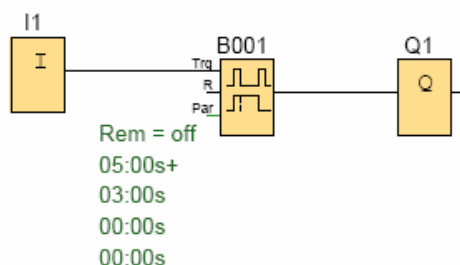
\* اگر زمان TL که با لبه رونده Trg فعال می شود قبل از غیر فعال شدن Trg سپری شود، در آن صورت خروجی به صورت دائم فعال می شود و با تمام شدن زمان T یا با فعال شدن ورودیهای Trg یا R غیر فعال می شود.

\* برای حالت دوم فرض کنید  $T_i, T_{iL}$  صفر باشند، عملکرد تابع به این صورت است که با فعال شدن Trg خروجی نیز فعال می شود. در این صورت اگر Trg قبل از سپری شدن زمان TL باز نشاندن شود، با لبه پایین رونده Trg زمان T فعال شده و بعد از سپری شدن T خروجی غیر فعال می شود.

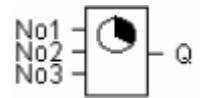
#### Timing diagram



تمرین: چراغ راه پله را با تابع double function به گونه ای طراحی کنید که اگر فرد ۳ ثانیه دست خود را بر روی کلید نگه دارد چراغ دائم روشن بماند.



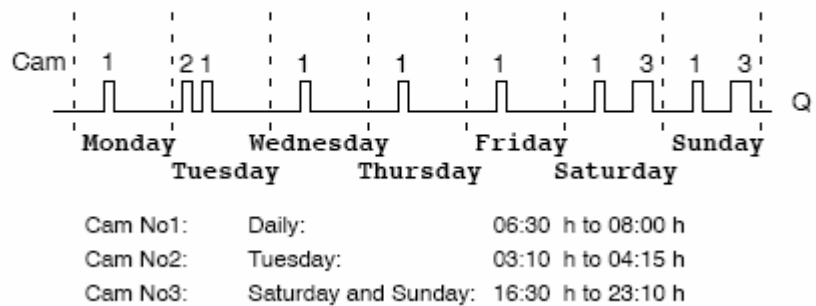
## Weekly timer تایمر هفتگی



این تابع یک تایمر هفتگی می باشد که در ساعات و روزهای قابل تنظیم، خروجی آن فعال خواهد شد.

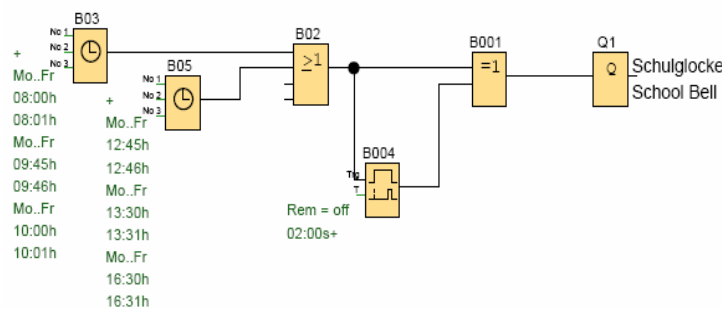
ذکر این نکته ضروری است که قطع شدن برق هیچ تاثیری بر روی عملکرد تابع زمان سنج ندارد.

Timing diagram (three examples)

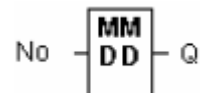


## تمرین: سیستم زنگ مدرسه

بوسیله تایمر هفتگی زنگ مدرسه را برای شروع و پایان مدرسه طراحی کنید، بطوریکه در روزهای دوشنبه تا جمعه در ساعت‌های ۸-۹:۴۵-۱۰-۱۳:۳۰-۱۶:۳۰ زده شود مدت زمان زنگ زدن را ۲ ثانیه در نظر بگیرید.



## Yearly timer تایمر سالانه



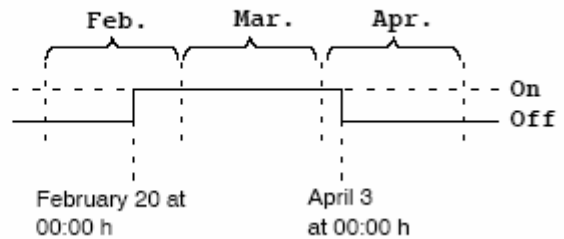
در این تابع خروجی توسط تاریخ روشن و خاموش می شود. یک تاریخ برای روشن شدن و یک تاریخ برای خاموش شدن، از ۱۲ ماه تعیین می شود.

خروجی از یک تاریخ تنظیم شده روشن می شود تا تاریخ بعدی، که نشان دهنده خاموش شدن خروجی می باشد.

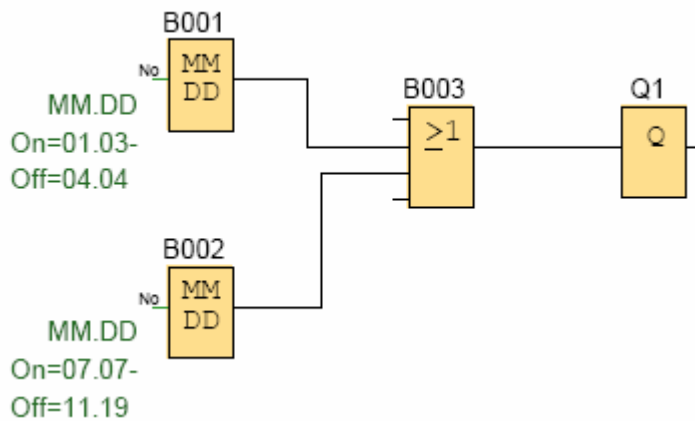


Timing diagram

MM.DD  
On=02.20  
Off=04.03

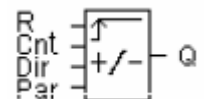


بطور مثال خروجی یک مدل LOGO از تاریخ ۱/۳ روشن می شود و در تاریخ ۴/۴ خاموش می گردد و همچنین خروجی این LOGO از تاریخ ۷/۷ روشن شده و در تاریخ ۱۹/۱۱ خاموش میشود. زمانیکه B1 یا B2 هر کدام باشند در آن صورت خروجی فعال می باشد و برای این منظور از یک OR استفاده شده است.



شمارنده Counter

شمارنده بالا/پایین شمار UP/DOWN COUNTER



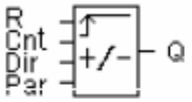
این تابع یک شمارنده می باشد که می تواند پالسهای که به ورودی آن توسط سنسورها و غیره وارد می شود را شمارش کند و در مقدار تنظیم شده، خروجی را فعال و غیر فعال کند.

\* خروجی این شمارنده زمانی فعال می شود که مقدار شمارش بیشتر یا برابر تنظیم شده شود.

\* ورودی cnt: این پایه مقدار شمارش را با توجه به هر پالس ورودی و با لبه بالا رونده تنظیم می کند.

\* جهت شمارش رانیز می توانید با ورودی Dir تنظیم کنید.

Dir=1 شمارش به طرف پایین

Symbol in LOGO!	Wiring	Description
	Input R	A signal at input R resets the internal count value to zero.
	Input Cnt	The function counts the 0 to 1 transitions at input Cnt. 1 to 0 transitions are not counted. Use <ul style="list-style-type: none"> <li>inputs I5/I8 for high-speed counting (only LOGO! 12/24 RC/RCo and LOGO! 24/24o): max. 2 kHz.</li> <li>any other input or circuit component for counting low frequency signals (typ. 4 Hz).</li> </ul>
	Input Dir	You set the direction of count at input Dir: Dir = 0: Up count Dir = 1: Down count
	Parameter	On: On threshold Range of values: 0...999999  Off: Off threshold Range of values: 0...999999  Retentivity for internal counter value Cnt: / = No retentivity R = The status is retentive.
	Output Q	Q is set or reset, depending on the current value at Cnt and the set thresholds.

Dir=0 شمارش به طرف بالا ۱..۲..۳...  
On threshold آستانه روشن شدن  
Off threshold آستانه خاموش شدن

☺ اگر بخواهید مقدار شمارش ، با قطع شدن برق تغییر نکند گزینه Retentivity را از پنجره

تنظیمات فعال کنید.

● اگر On threshold آستانه روشن شدن، بزرگتر از Off threshold آستانه خاموش شدن باشد:

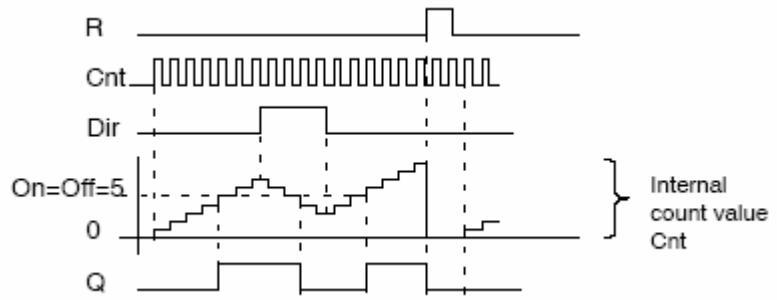
$$Q=1 \text{ اگر } cnt \geq on$$

$$Q=1 \text{ اگر } cnt < off$$

● اگر On threshold آستانه روشن شدن، کوچکتر از Off threshold آستانه خاموش شدن باشد:

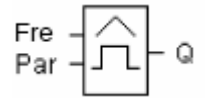
$$Q=1 \text{ اگر } on \leq cnt < off$$

با فعال شدن Reset تمامی مقادیر پاک می شوند.

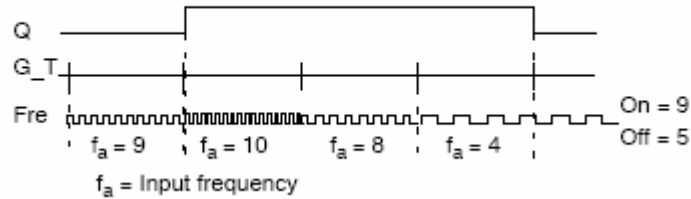


تمرین: برنامه ای بنویسید که یک شمارنده بسته های روی یک کانوایر را شمرده و زمانی که تعداد بسته ها به عدد ۵ رسید شمارش را به مدت ۲ ثانیه متوقف کرده و سپس چرخه را تکرار کند.

### Threshold trigger



### Timing diagram

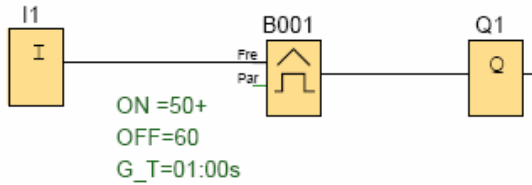


**G\_T**: فاصله زمانی برای هر پالس فرکانسی می باشد که می تواند یک پریود مابین ۰۰:۰۵ تا ۹۹:۹۹ ثانیه باشد.

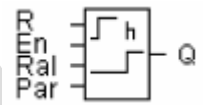
**Fa**: فرکانس ورودی است. تعداد تکرار پالسهای ایجاد شده توسط ورودی در یک ثانیه فرکانس ورودی نامیده می شود.

- If the On threshold  $\geq$  Off threshold, then:  
 $Q = 1$ , if  $f_a > On$   
 $Q = 0$ , if  $f_a \leq Off$ .
- If the On threshold  $<$  Off threshold, then  $Q = 1$  if  
 $On \leq f_a < Off$ .

در مورد دوم یعنی اگر تعداد پالسهای ورودی در مدت زمان  $G\_T$  بین مقادیر  $off$  و  $on$  باشد روشن می شود.



### شمارنده ساعتی Hours counter



این تابع، یک ساعت می باشد که برای ذخیره کردن مدت زمان کارکرد وسیله مورد نظر مورد استفاده قرار می گیرد. وقتی ورودی  $En$  فعال می شود پارامتر زمان تنظیم شده شروع به فعالیت می کند. خروجی زمانی فعال می شود که زمان باقی مانده  $MN$  به پایان برسد.

$MI$ : فاصله زمانی طی شده

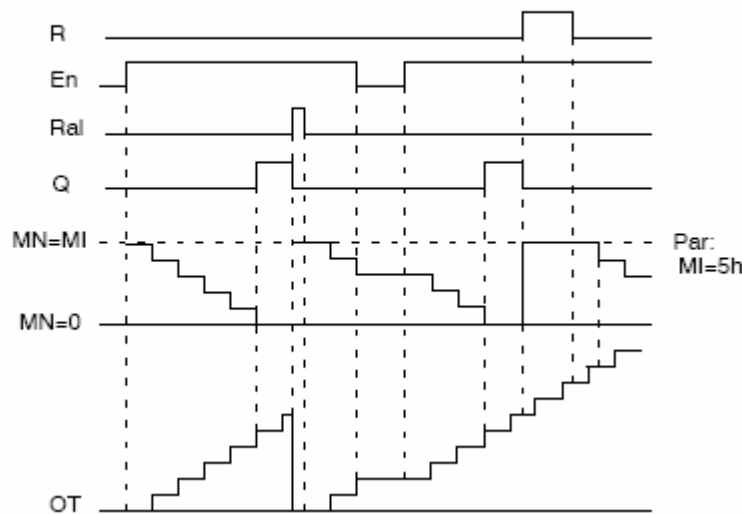
$MN$ : زمان باقی مانده

$OT$ : کل زمان سپری شده بعد از آخرین سیگنال بر حسب ورودی  $Ral$  می باشد.

در صورتیکه ورودی  $Ral$  (reset all) نشانده شود تمامی مقادیر صفر می شوند.

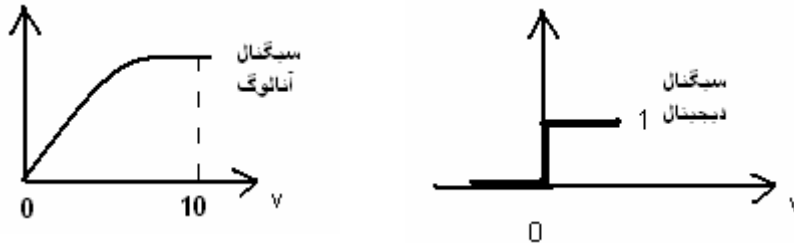
اگر ورودی  $R$  نشانده شود شمارش داخلی  $OT$  حالت اجرا را دنبال کرده و تغییر نمی کند، ولی خروجی خاموش می شود.

### Timing diagram



## آنالوگ Analog

آنالوگ به سیگنالی گفته می شود که پیوستگی در آن حفظ شود و متغیر نیز باشد مثل سیگنال سنسور دما، فشار و سطح مایع یک مخزن.



در اکثر توابع آنالوگ از دو پارامتر **gain** و **offset** استفاده می شود. سیگنالی که از سنسور به **logo** وارد می شود بین ۰-۱۰ ولت و ۰-۲۰ میلی آمپر می باشد که این سیگنال برای **logo** ارزش ۰ تا ۱۰۰۰ را دارد.

مثال: یک سنسور دما، دمای بین ۰ تا ۱۰۰ درجه سانتیگراد را اندازه گیری می کند. مقدار واقعی  $(Ax \cdot gain) + offset =$  تبدیل می کند.

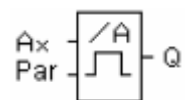
با فرض آنکه دمای کارخانه در جایی که سنسور دما قرار دارد برابر ۲۵ درجه باشد. برای آنکه **logo** کنترل مناسبی روی فرآیند انجام بدهد، مقادیر **A, B** را تعیین کنید.

$$A = 1200 / 1000 = 1.2 \quad \blacktriangleright \quad A = 1.2$$

پس بایستی دمای کارخانه را از دمای اندازه گیری کم کنیم تا همواره مقدار واقعی به **logo** تحویل داده شود.

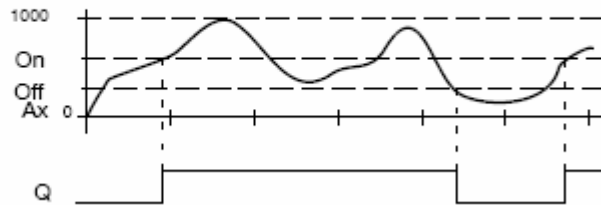
$$B = -25$$

### Analog threshold trigger



خروجی زمانی روشن می شود که مقدار آنالوگ از مقدار تنظیم شده برای آستانه روشن شدن تجاوز کند. هرگاه مقدار آنالوگ برابر یا کمتر از مقدار تنظیم شده برای آستانه خاموش شدن شود خروجی خاموش می شود.

### Timing diagram

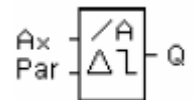


اعداد اعشاری فقط در زمان نمایش ON,OFF و مقدار AX در یک متن پیام قابل اجرا می باشند. همچنین مقدار ON,OFF را برای مقایسه نمی توان اجرا کرد، زیرا در حین مقایسه از علامت اعشاری چشم پوشی می شود.

### Calculation rule

- If the On threshold  $\geq$  Off threshold, then:  
 $Q = 1$ , if the actual value  $Ax > On$   
 $Q = 0$ , if the actual value  $Ax \leq Off$ .
- If the On threshold  $<$  Off threshold, then  $Q = 1$  if  $On \leq$  actual value  $Ax < Off$ .

### راه انداز تفاضلی آنالوگ Analog differential trigger

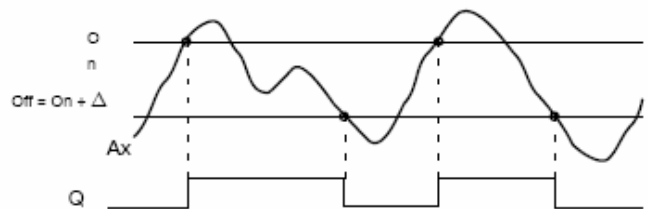


در این تابع خروجی متناسب با ارزش آستانه و یک مقدار تفاضلی فعال و غیر فعال می شود. پارامتر P یا تعداد ارقام اعشاری در یک پیام متنی قابل اجرا است.

تابع بطور خودکار پارامتر OFF را بصورت  $OFF = ON + DELTA$  محاسبه می کند که به موجب آن DELTA ممکن است منفی یا مثبت باشد.

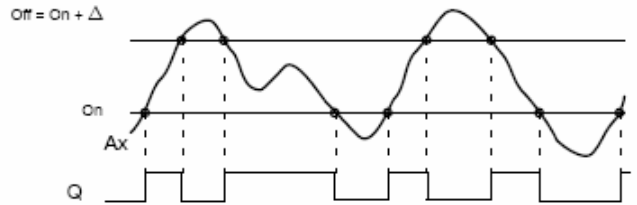
$$(AX.GAIN)+B=.act.AX$$

$$(AY.GAIN)+B=.act.AY$$



- When you set a negative differential value  $\Delta$ , the On threshold  $\geq$  Off threshold, and:  
 $Q = 1$ , if the actual value  $Ax > On$   
 $Q = 0$ , if the actual value  $Ax \leq Off$ .

- When you set a positive differential value  $\Delta$ , the On threshold  $<$  the Off threshold, and  $Q = 1$ , if:  
 $On \leq \text{actual value } Ax < Off$ .



DELTA=(-)

### مقایسه کننده آنالوگ Analog comparator



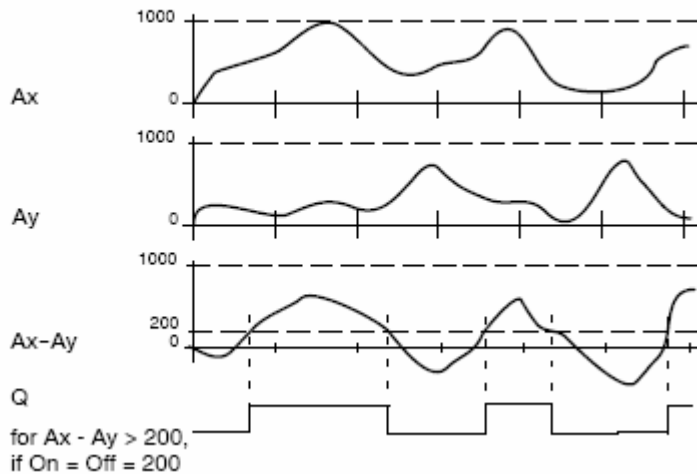
در این تابع، خروجی زمانی فعال است که تفاوت بین  $AX, AY$  از مقدار تعیین شده تجاوز کند.

$$AX - AY > ON$$

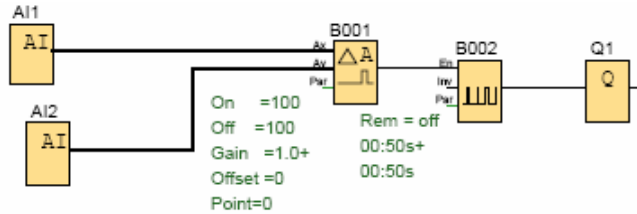
قاعده محاسبه:

- If the On threshold  $\geq$  Off threshold, then:  
 $Q = 1$ , if:  
 (actual value Ax - actual value Ay)  $>$  On  
 $Q = 0$ , if:  
 (actual value Ax - actual value Ay)  $\leq$  Off.
- If the On threshold  $<$  Off threshold, then  $Q = 1$ , if:  
 $On \leq$  (actual value Ax - actual value Ay)  $<$  Off.

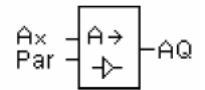
### Timing diagram



مثال: برای تولید یک آلیاژ مرغوب از دو کوره ذوب استفاده می کنیم. شرط تولید محصول با کیفیت آن است که دمای کوره دوم نباید بیش از ۱۰۰ درجه با دمای کوره اول اختلاف داشته باشد. برنامه ای بنویسید که در صورت بروز چنین مسئله ای چراغ آلارم چشمک بزند.



### تقویت کننده آنالوگ Analog amplifier



☺ این تابع فقط برای دستگاههای سری OBA4 قابل دسترسی است.

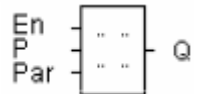
این بلوک یک مقدار ورودی آنالوگ را تقویت می کند و به یک خروجی آنالوگ باز گشت می دهد.

$$(Ax \cdot gain) + offset = actual\ value\ Ax.$$

مقدار واقعی AX همان مقدار واقعی AQ است.

خروجی آنالوگ تنها میتواند مقادیری از ۰ تا ۱۰۰۰ را پردازش کند. بنابراین برای آنکه دامنه خروجی تابع در محدوده ۰ تا ۱۰۰۰ باشد، باید از تقویت کننده استفاده کنیم.

### Message texts



یک متن پیامی تنظیم شده در مدت اجرا نمایش داده می شود.

طریقه عملکرد تابع به این صورت است که هرگاه وضعیت ورودی En از صفر به یک تغییر پیدا کند، متن

پیام تنظیم شده در حالت اجرا نمایش داده می شود. هرگاه چندین متن پیام داشته باشید، می توانید برای

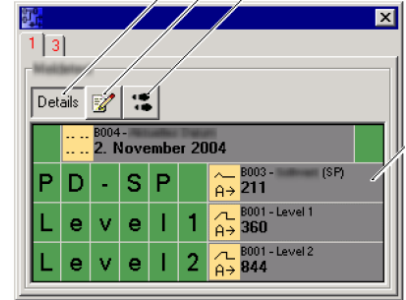
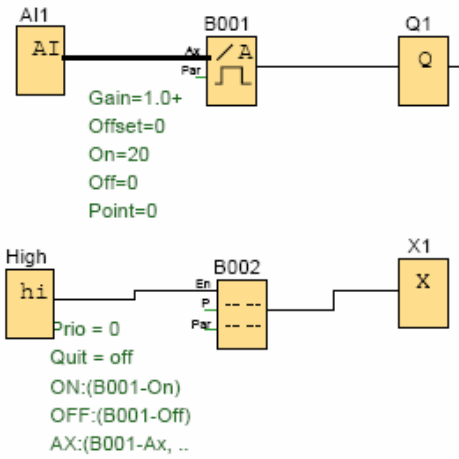
اولویت دادن بلوکی، شماره آنرا از جدول مشخصات در قسمت priority تعیین کنید.

شما می توانید در هر پیام حداکثر ۴ خط که هر خط دارای ۱۲ حرف می باشد، را استفاده کنید.

بطور مثال برنامه زیر را اجرا کنید و text های on, off, AX را همراه با پارامترهای مربوطه در

Message texts تایپ کنید.



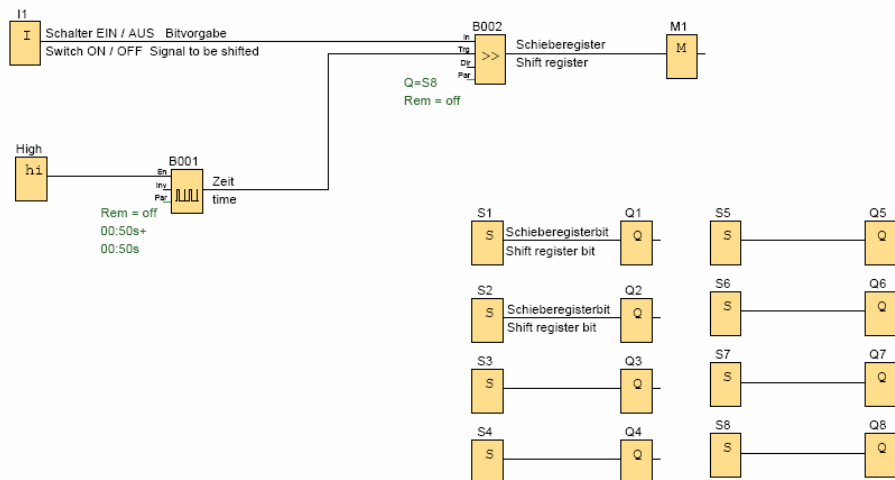


### Shift register



این تابع فقط برای دستگاههای سری OBA4 قابل دسترسی است. تابع شیفت رجیستر می تواند برای خواندن مقدار یک ورودی و بیتهای شیفت استفاده شده باشد. مقدار خروجی با بیت ایجاد شده شیفت رجیستر مطابقت دارد. تابع شیفت رجیستر فقط برای یکبار در داخل مدار برنامه استفاده می شود. ورودی IN: وقتی تابع مقدار این ورودی را می خواند که فعال باشد. ورودی Trg: تابع با یک لبه بالا رونده Trg تحریک می شود. بیت شیفت رجیستر تعیین کننده مقدار خروجی Q است. به عنوان مثال اگر برای آن انتخاب کنیم خروجی در بیت چهارم روشن می شود.

برای آنکه بطور کامل متوجه مسئله شوید مثال زیر را اجرا کنید.



## فصل چهارم

برنامه نویسی بر روی خود LOGO!

سوئیچینگ روی LOGO!

شما می توانید برای راه اندازی LOGO!basic از قواعد آسان زیر پیروی کنید:

۱- اگر در داخل LOGO یا کارت متصل شده به LOGO هیچ برنامه ای نباشد در آنصورت پیغام "No program" گزارش داده میشود، که در آن حالت دکمه ESC را برای خارج شدن از آن

حالت فشار دهید.

No Program

توجه: در نسخه های قدیمی به جای دکمه ESC بایستی سه دکمه OK and ► ◀ به طور همزمان فشار داده شود.

۲- اگر برنامه ای در داخل کارت متصل شده به logo وجود داشته باشد، در آن صورت، برنامه بعد از وصل برق بطور اتوماتیک بر روی LOGO کپی خواهد شد.

۳- اگر در داخل LOGO یا کارت متصل شده به LOGO برنامه ای باشد در آنصورت LOGO حالت عملیات را قبل از قطع برق بر عهده خواهد گرفت. (یعنی اگر قبل از قطع برق در حالت RUN بود بعد از وصل برق نیز به حالت RUN می رود).

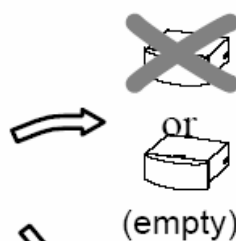
\*نسخه بدون صفحه نمایش LOGO...RC بطور اتوماتیک از حالت Stop به حالت RUN سوئیچ می شود و LED از حالت قرمز به سبز تغییر رنگ می دهد.

نکته: هرگاه برق بطور اتفاقی ضعیف شود و شما برنامه ای را روی LOGO وارد کنید، بعد از برگشت برق به حالت نرمال برنامه پاک خواهد شد. از این رو شما برنامه اصلی را قبل از تغییرات روی یک کارت یا روی برنامه ذخیره کنید.

Before power off

No Program

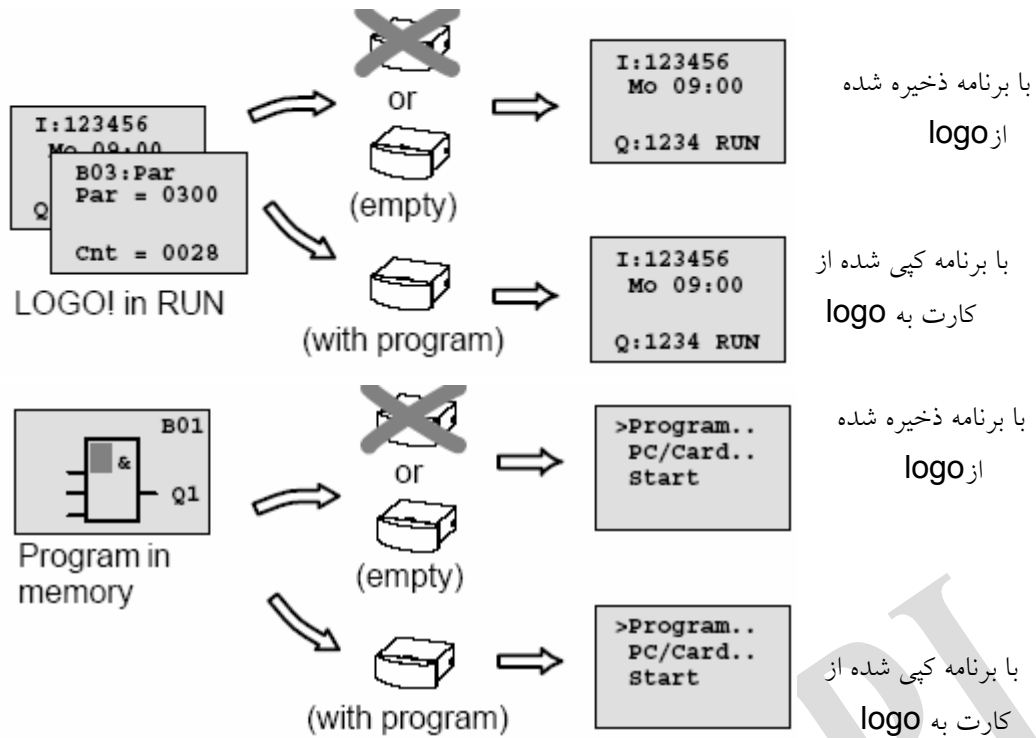
No program in memory



After power on

No Program

>Program..  
PC/Card..  
Start



### کپی کردن برنامه از logo به کارت

برای این کار لازم است مراحل زیر را طی کنید:  
 \* ابتدا کارت را در داخل logo قرار دهید.  
 \* logo را به حالت برنامه نویسی سوئیچ کنید.

```
>Program..
PC/Card..
Start
```

\* بوسیله کلید های جهتی PC/Card را انتخاب کنید.

\* کلید ok را فشار دهید.

```
>PC↔[+]
[+]→Card
Card→[+]
```

[+] = LOGO!

\*بوسیله کلیدهای جهتی گزینه **Card** را انتخاب کنید.  
\* **OK** را فشار دهید.

\*وقتی **logo** عمل کپی کردن را به پایان برساند، بطور خودکار به منوی اصلی بر می گردد.

```
>Program..  
PC/Card..  
Start
```

نکته: اگر برنامه همراه رمز عبور در داخل **logo** ذخیره شده باشد، باید برنامه کپی شده در داخل کارت به همراه همان رمز عبور باشد.

### کپی کردن از کارت به **logo**

اگر برنامه ای در داخل کارت حافظه داشته باشید در آنصورت می توانید به دو روش زیر آن را بر روی **logo** کپی کنید:

\***کپی شدن خودکار در هنگام راه اندازی **logo** یا **Power on****، هرگاه kartی که دارای برنامه است به یک **logo** وصل شود در آنصورت با روشن شدن **logo** برنامه کارت بطور خودکار به **logo** کپی می شود.

\***کپی شدن از طریق منوی **PC/Card** به **logo****

### برنامه نویسی بر روی خود **LOGO!**

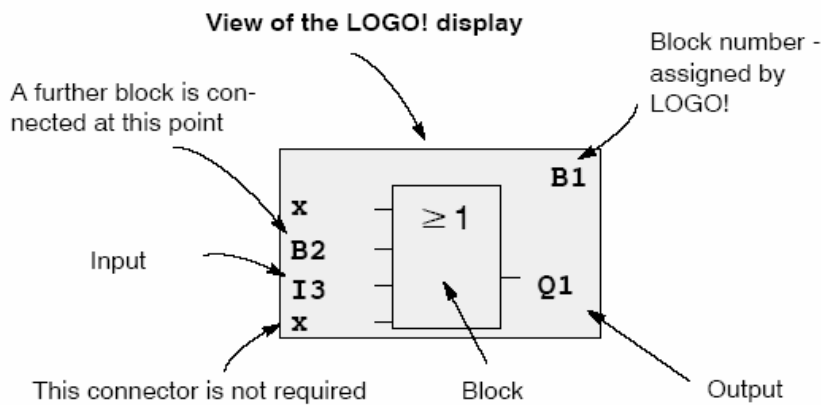
ابتدا عناصر مربوط به برنامه نویسی را توضیح می دهیم.  
بطور کلی هر برنامه شامل دو قسمت بصورت بلوکها **BLOCKS** و اتصال دهنده ها **connectors** می باشد که شامل مجموعه ای از ورودیها **I** و خروجیها **Q** و فلگها **M** است. در ورژن جدید ورودیها **I** و خروجیها **Q** و فلگها **M** که برای برنامه نویسی قابل دسترس هستند **I1** تا **I24**، **A1** تا **A18**، **Q1** تا **Q16** و **M1** تا **M24** می باشند. وضعیت های ورودی و خروجی می توانند صفر یا یک باشند. بطور پیش فرض وضعیت های ثابتی برای **hi** بصورت یک و برای **lo** بصورت صفر در نظر گرفته شده است.

Con-nectors	LOGO! Basic / Pure		DM	AM	AM2AQ
Inputs	LOGO! 230RC/RC <sub>o</sub> , LOGO! 24 RC/RC <sub>o</sub>	Two groups: I1... I4 and I5 ... I8	I9 ... I24	AI1... AI8	none
	LOGO! 12/24RC/ RC <sub>o</sub> , LOGO! 24/24 <sub>o</sub>	I1... I6, I7, I8  AI1, AI2	I9 ... I24	AI3... AI8	
Outputs	Q1...Q4		Q5 ... Q16	none	AQ1, AQ2
lo	Logical '0' signals (off)				
hi	Logical '1' signals (on)				
x	An existing connection that is not used				

DM: Digital module  
AM: Analog module

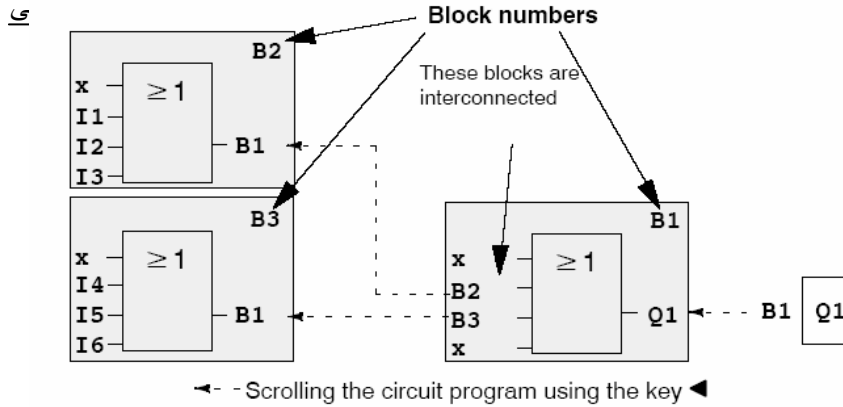
### بلوکها و شماره های بلوک

یک بلوک تابعی است که برای تبدیل اطلاعات ورودی به اطلاعات خروجی مورد استفاده قرار می گیرد. ورودی این بلوکها را از منوی CO و گاهی SF، GF و BN انتخاب می کنیم.



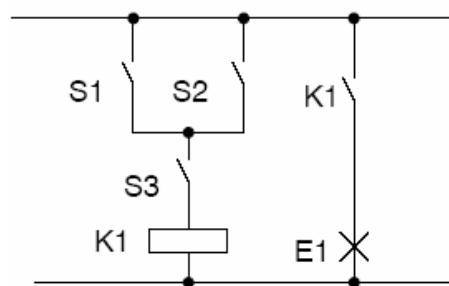
### تعیین شماره های بلوک

وقتی شما یک بلوک را در داخل نرم افزار یا LOGO درج می کنید در آن صورت یک شماره برای آن بلوک توسط نرم افزار یا LOGO تعیین می شود. شماره های بلوک به شما کمک می کند تا مسیر اتصالات را در داخل برنامه همانند شکل زیر به سادگی پیدا کنید. بهتر است برای برنامه نمودار بلوکی ایجاد کنید (مشابه روش FBD).



### نمونه ای از دیاگرام مدار LOGO

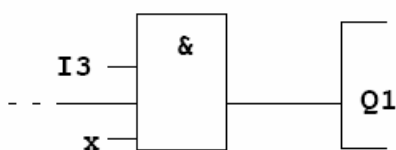
در مدار نشان داده شده در شکل زیر بار E1 بوسیله کلیدهای S1 یا S2 و S3 در وضعیت on و off قرار می گیرد. رله k1، زمانی که کلیدهای s1 یا s2 و s3 بسته می شوند عمل می کند. این مدار را با طی مراحل زیر برای logo قابل اجرا می کنیم.



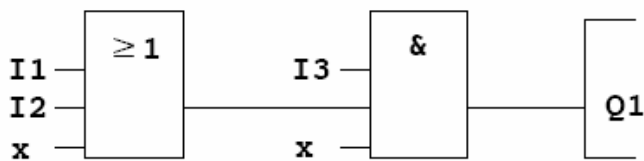
😊 برای تبدیل مدار به دیاگرام بلوکی از فروجی به سمت ورودی حرکت کنید.

برای اینکار مراحل زیر را طی کنید:

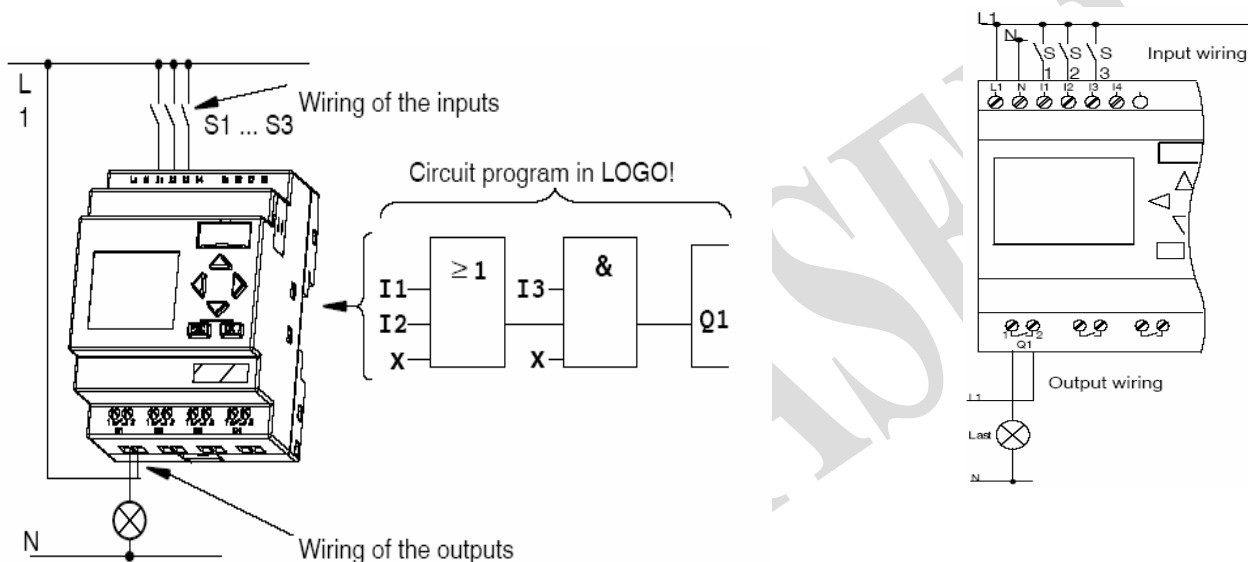
**گام اول:** در اینجا خروجی به کنتاکت نرمال باز S3 وصل شده و از طریق آن به اجزاء دیگر مدار متصل شده است. رابط سری S3 بصورت یک بلوک AND تعریف می شود. پس بجای کلید S3 که به صورت یک رابط سری بین خروجی و S1 و S2 است مشابه شکل زیر از یک بلوک AND استفاده می کنیم.



گام دوم: S1 و S2 بصورت موازی وصل شده اند. عملکرد مدار مشابه گیت OR است پس از یک بلوک OR استفاده می کنیم.



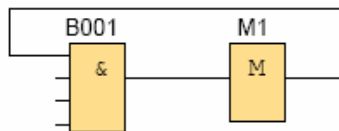
همانطور که مشاهده می کنید فقط دو تا از ورودیهای بلوک AND و OR استفاده شده در نتیجه ورودی سوم هر یک را باید بصورت غیر فعال "X" علامت گذاری کرد. سیم کشی این مدار را بوسیله LOGO در شکل زیر مشاهده می کنید.



### قانون های مهم برای برنامه نویسی با LOGO

- برای ویرایش مدار در حالت برنامه نویسی بعد از وصل کردن برق یا روشن کردن دستگاه اگر گزینه "no program, press Esc" نمایش داده شد. در آن صورت می توانید حالت برنامه نویسی را با انتخاب گزینه ESC ایجاد کنید. (در نسخه های جدید این کار بوسیله فشار دادن سه دکمه ◀, ▶ and OK صورت می گیرد.)

- می توانید زمان و مقادیر پارامتر یک برنامه موجود را در حالت SET PARAMETR و در حالت برنامه نویسی ویرایش کنید.
- می توانید حالت اجرای برنامه RUN را بوسیله گزینه START از منوی اصلی آغاز کنید.
- در حالت اجرا شما می توانید از طریق گزینه ESC (در انواع قدیمی تر با فشار دادن همزمان OK,ESC) به حالت پارامتر برگردید.
- همیشه طراحی برنامه یک مدار از طرف خروجی به طرف ورودی می باشد.
- شما می توانید یک خروجی را به چندین ورودی وصل کنید، اما نمی توانید یک ورودی را به چندین خروجی گسترش دهید.
- نمی توانید خروجی یک بلوک را به ورودی خودش وصل کنید. برای اینکار در صورت نیاز باید خروجی یک ورودی را به یک پرچم M یا یک خروجی وصل کنید سپس از طریق آنها به خود ورودی وصل کنید.



- برای تایید انتخاب ترمینال یا یک بلوک، OK را فشار دهید و برای خارج شدن از حالت برنامه نویسی مدار ESC را فشار دهید.
- قبل از اینکه برنامه مدار خود را وارد LOGO کنید، نقشه آنرا بر روی کاغذ پیاده کنید.
- LOGO فقط برنامه های کامل ذخیره می کند، اگر برنامه ناقص باشد در آن صورت LOGO نمی تواند از حالت برنامه نویسی خارج شود.

### معرفی منوهای LOGOV3.0

Main menu

```
>Program..
PC/Card..
Start
```

منوی اصلی

Programming menu

```
>Edit Prg
Clear Prg
Set Clock
ASi BUS..
```

منوی برنامه نویسی

PC/card menu

```
>PC<=>[F5]
[F5]>Card
Card->[F5]
```

منوی کارت

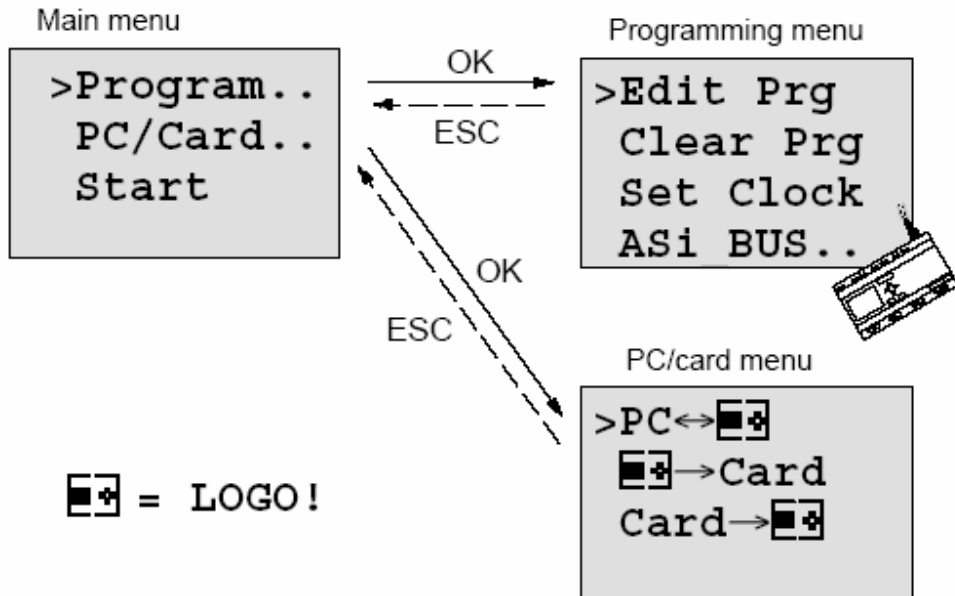
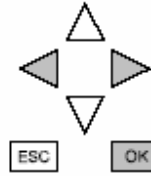
```
>Set Clock
Set Param
```

منوی پارامتر

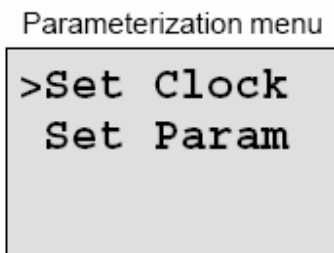
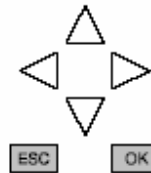


نحوه انتخاب منوها

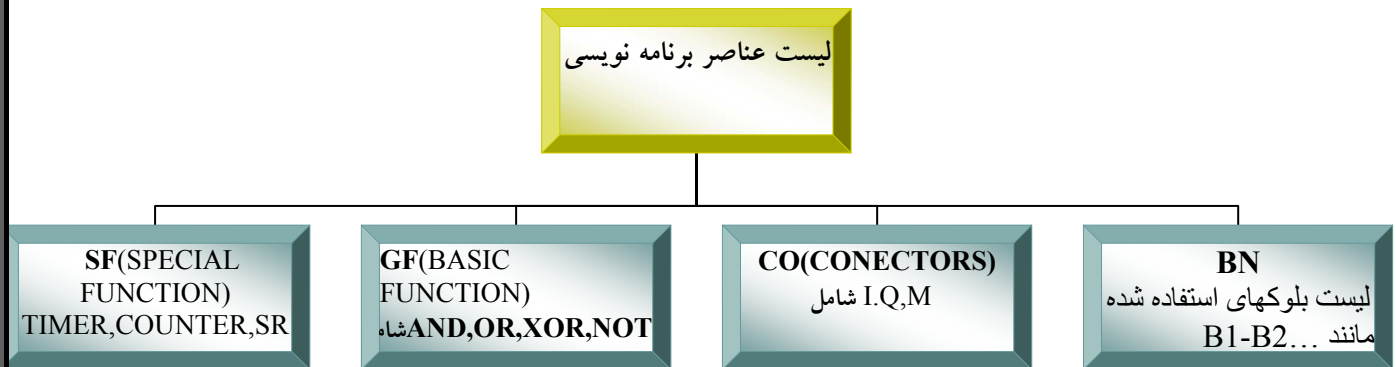
Programming mode



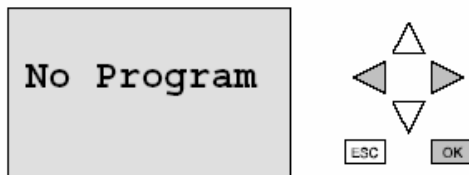
Parameterization mode



لیست عناصر برنامه نویسی

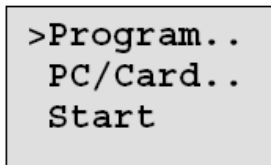


بعد از روشن کردن دستگاه زمانیکه پیغام NO PROGRAM ظاهر شد کلیدهای **OK** and **◀▶** را فشار دهید تا به منوی برنامه نویسی بروید.

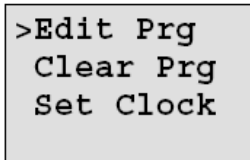


😊 **تذکره:** این کار در نسخه های جدید با استفاده از گزینه ESC صورت می گیرد.

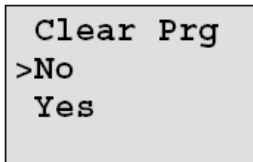
نکته: اگر قبلاً برنامه ای در داخل LOGO وجود داشته بعد از روشن کردن دستگاه، لوگو به منوی اصلی می رود.



اگر می خواهید برنامه قبلی را پاک کنید، در منوی اصلی گزینه **program** را انتخاب و **OK** را فشار دهید تا وارد منوی برنامه نویسی شوید.



با استفاده از مکان نما گزینه **Clear Prg** را انتخاب و **ok** را فشار دهید سپس در آنجا "yes" را انتخاب کنید. در آنصورت برنامه حذف خواهد شد و به منوی برنامه نویسی برمی گردد.

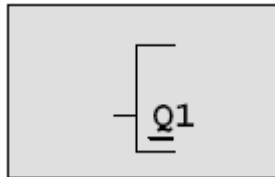


اگر برنامه قبلی را پاک کردید، در منوی اصلی بازگزینه **program** را انتخاب و **OK** را فشار دهید تا وارد منوی برنامه نویسی شوید.

```
>Edit Prg
Clear Prg
Set Clock
```

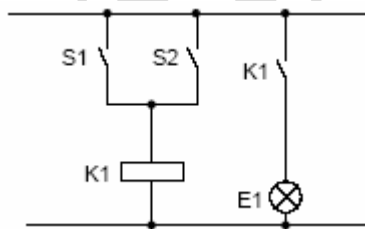
اما بار دیگر یاد آور می شویم که جهت انتحاب گزینه های مختلف از کلیدهای **▲** or **▼** استفاده کنید و برای تایید انتحاب کلید **OK** را فشار دهید.

گزینه **Edit Prg** را انتخاب کنید. سپس برای تایید **OK** را فشار دهید. در این حالت شکل زیر که نشان دهنده اولین خروجی است ظاهر می شود.

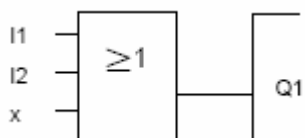


حالا شما در حالت برنامه نویسی هستید. با استفاده از کلید های نشانگر می توانید خروجی های دیگری را انتخاب کنید. از اینرو می توانید از همین نقطه، برنامه نویسی مدار را آغاز کنید. در غالب یک مثال نحوه وارد کردن برنامه از روی دیاگرام مداری را بیان می کنیم.  
برنامه اول:

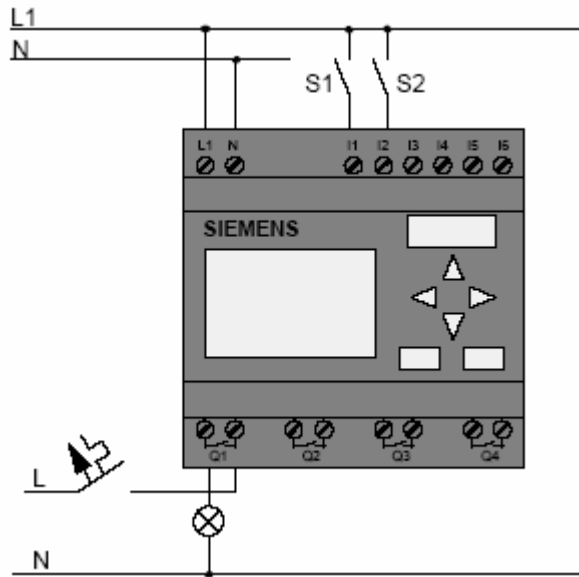
بار **E1** بوسیله کلیدهای **S1** یا **S2** و **S3** در وضعیت **on** و **off** قرار می گیرد. رله **K1** زمانی که کلید های **S1** یا **S2** و **S3** بسته شوند عمل می کند.



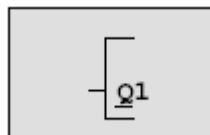
این برنامه بشکل زیر توسط **logo** قابل اجرا است.



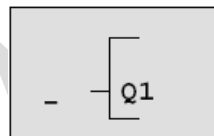
سیم بندی مدار نیز بصورت شکل زیر انجام می شود. همانطور که ملاحظه می کنید کلید s1 و s2 ورودی I1 و I2 را فعال می کنند. همچنین بار به رله خروجی Q1 متصل شده است.



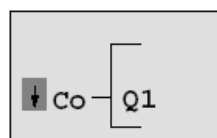
برای وارد کردن برنامه به این صورت عمل می کنیم که بعد از انتخاب گزینه Edit Prg ما اجازه ویرایش برنامه را در مسیر خروجی بسمت ورودی را خواهیم داشت. اولین مرحله تعیین خروجی و اتصال آن به عناصر دیگر این است، که به صورت زیر عمل می کنیم.



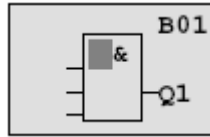
با استفاده از کلید ◀ مکان نمائی را که در زیر Q قرار دارد به طرف چپ حرکت دهید.



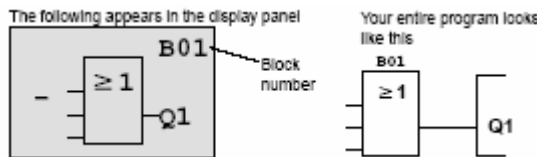
در اینجا باید یک بلوک OR را انتخاب کنید. برای اینکار کلید OK را فشار دهید. سپس علامت CO که نشان دهنده اتصال دهنده ها است مشابه شکل زیر ظاهر خواهد شد. در این حال با استفاده از گزینه های ▲ یا ▼ علامت مربوط به توابع بیسیک GF را انتخاب کنید.



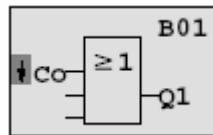
سپس OK را برای وارد شدن به داخل توابع بیسیک فشار دهید. در این حالت مطابق شکل زیر اولین بلوک این توابع که AND است، ظاهر خواهد شد.



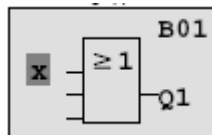
با کلید های ▲ یا ▼ بلوک OR را انتخاب کنید. سپس OK را فشار دهید.



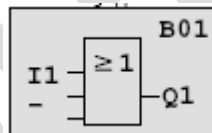
OK را فشار دهید.



OK را فشار دهید. می بینید که اولین گزینه در CO، گزینه "X" است

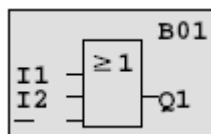


با استفاده از گزینه های ▲ یا ▼ I1 را انتخاب و برای تایید OK را فشار دهید.

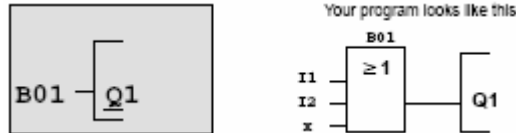


نکته: اگر بخواهید از ورودی تابعی استفاده نکنید ممتما بایستی از منوی CO برای آن ورودی گزینه "X" را انتخاب کنید، در غیر اینصورت خطا تلقی می شود.

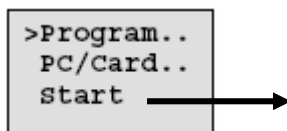
برای ورودی دوم یعنی I2 همین کار را انجام دهید.



برای ورودی سوم، چون بدون استفاده است، از **CO**، **X** را برای آن انتخاب کنید. سپس **OK** را فشار دهید. تا مطابق آنچه در شکل زیر مشاهده می کنید، ظاهر شود.

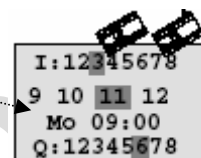


برای اجرای برنامه دوبار دکمه **ESC** را فشار دهید تا به منوی اصلی بروید. سپس در منوی اصلی گزینه **START** را انتخاب کنید و برای تایید **OK** را فشار دهید.



بعد از انتخاب و تایید

تاریخ و زمان جاری روز فقط در نسخه هائی که دارای ساعت هستند مثل **RC**



هرگاه ورودی و خروجی در وضعیت فعال قرار گیرند بوسیله علامت ■ مشخص می شوند.

وضعیت نمایش

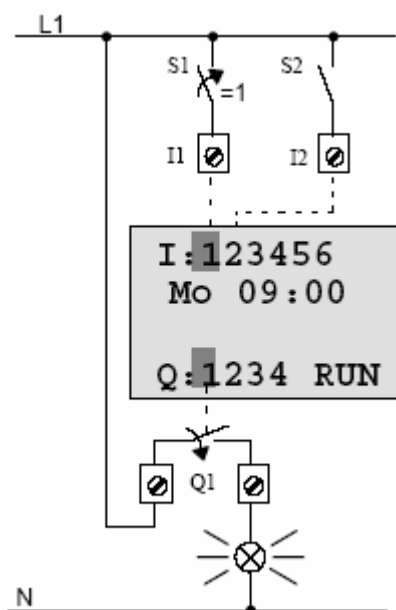
وقتی کلید **S1** بسته می شود، ولتاژ روی **I1** قرار می گیرد و ورودی را در وضعیت فعال قرار می هد.

**LOGO** برنامه را پردازش کرده و وضعیت خروجی را

مشخص می کند. اگر خروجی **Q1** در وضعیت فعال قرار

گیرد، **LOGO** رله **Q1** را فعال می کند و ولتاژ باری که بر روی

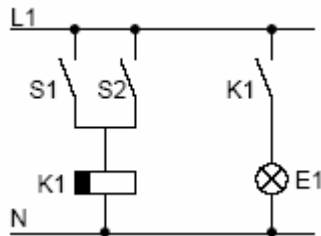
**Q1** است تامین می شود.



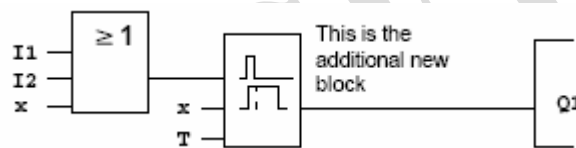
## برنامه دوم

### اصلاح مدار

برای اصلاح مدار، مثال قبل را با کمی تغییر بصورت مدار شکل زیر ارائه می کنیم.



همانطور که مشاهده می کنید هرگاه کلید S1 یا S2 بسته شود در آن صورت رله K1 عمل کرده و کنتاکت باز آن بسته می شود و با این عمل بار E1 دارای ولتاژ خواهد شد. سپس بعد از سپری شدن ۱۲ دقیقه تنظیم شده، کنتاکت K1 رله باز شده و ولتاژ خروجی قطع خواهد شد. طراحی این مدار برای LOGO مطابق شکل زیر خواهد بود.



\* تایمر جز تابع ویژه SF است.

### ویرایش برنامه

برای ویرایش برنامه بصورت زیر عمل کنید.

LOGO را به حالت برنامه نویسی سویچ کنید. اگر LOGO در حالت RUN قرار دارد سه کلید

◀▶, OK, را بطور همزمان فشار دهید.

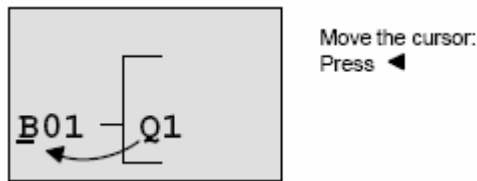
\*تذکره: در نسخه های جدید در حالت RUN کلید ESC را فشار دهید. در این حالت منوی پارامتر گشوده می شود. حالت STOP را انتخاب و بوسیله OK تایید کنید و سپس گزینه YES را انتخاب و بوسیله OK تایید کنید.

در منوی اصلی گزینه Program را انتخاب کنید.

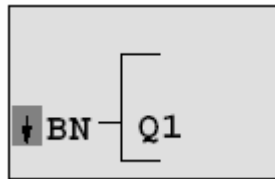
در منوی برنامه نویسی گزینه **Edit prg** را انتخاب کنید. در این حالت می توانید برنامه قبلی را اصلاح کنید.

```
>Edit Prg
Clear Prg
Set Clock
```

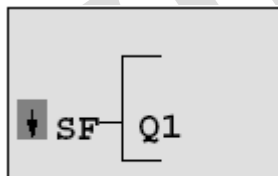
برای درج بلوک جدید به این صورت عمل می کنیم که در برنامه قبلی علامت مکان نما که در زیر Q قرار دارد را بوسیله کلید ◀ حرکت داده تا زیر حرف B از B01 که شماره بلوک OR بود قرار گیرد.



حالا می توان در این مکان یک بلوک جدید درج کرد. برای اینکار بلوک OK را فشار دهید.

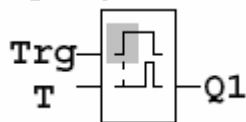


بوسیله کلید ▼ لیست SF را که شامل تایمرها است را انتخاب کنید.

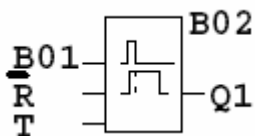


Press OK.

OK را فشار دهید تا اولین تابع ویژه نمایش داده شود.

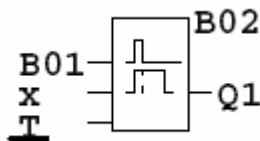


با استفاده از کلیدهای ▲ و ▼ بلوک مورد نظر را انتخاب کنید و انتخاب خود را با فشار دادن OK تایید کنید. در این قسمت بلوک تابع OFF Delay را انتخاب کنید.



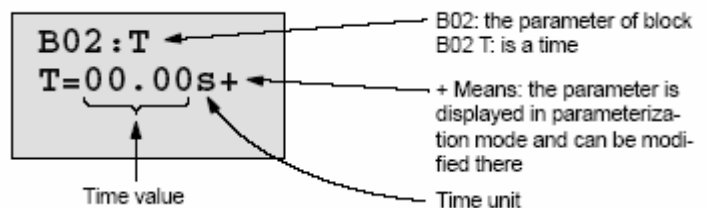


شماره بلوک درج شده B02 می باشد. بلوک شماره یک B01 بطور خودکار به ورودی بالائی بلوک شماره ۲ وصل شده است و این بلوک نیز به خروجی Q1 متصل می شود.  
 بلوک OFF Delay دارای سه ورودی است که ورودی بالائی، ورودی راه انداز Trg می باشد. در این مثال تابع OFF Delay بوسیله بلوک OR که همان بلوک شماره یک می باشد راه اندازی می شود.  
 در این مثال چون از ورودی R استفاده نشده لذا آن را بوسیله "X" محدود کرده ایم.  
 با انتخاب گزینه T می توانید زمان خاموش شدن خروجی تابع OFF Delay را تعیین کنید.



### روش تعیین پارامترهای بلوک

با استفاده از کلیدهای ◀ و ▶ می توانید به صورت طولی حرکت کنید و با استفاده از کلیدهای ▲ و ▼ می توانید مقدار زمان را تغییر دهید و در نهایت با فشردن کلید ok تایید کنید.

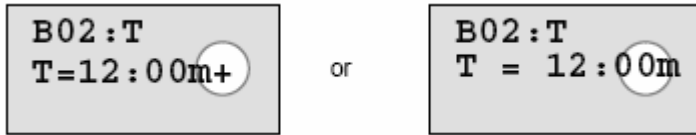


### تنظیمات زمان

زمان T را به روش زیر برابر ۱۲:۰۰ دقیقه قرار دهید.

1. Move the cursor to the first position:  
◀ or ▶
2. Select '1':  
▲ or ▼
3. Move the cursor to the second position:  
◀ or ▶
4. Select '2':  
▲ or ▼
5. Move the cursor to the unit:  
◀ or ▶
6. Select the unit m for minutes:  
▲ or ▼

\*نوع حفاظت+ : با انتخاب این پارامتر می توان زمان T رادر حالت جایگزینی پارامتر تغییر داد.



پیش فرض نیز بر روی + است.

نوع حفاظت - : با انتخاب این پارامتر

نمی توان زمان T رادر حالت جایگزینی پارامتر تغییر داد.

طریقه انتخاب حفاظت

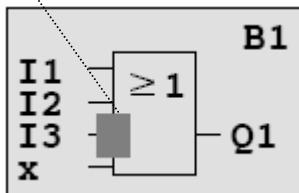
7. Move the cursor to the protection mode::  
◀ or ▶
8. Select protection mode '-':  
▲ or ▼

طریقه not کردن ورودی

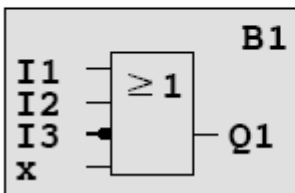
تذکره: در ورژن های جدید برای not کردن ورودی ، می توان بدون کمک گرفتن از بلوک not این کار را انجام داد.

جزئیات آن در شکل های زیر آمده است.

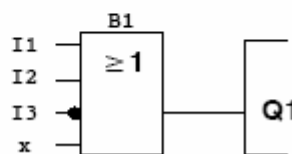
مکان نما را با استفاده از کلید ▶ به طرف راست حرکت دهید تا مطابق شکل روبرو علامت ■ ظاهر شود. سپس مطابق دستورات زیر حرکت کنید.



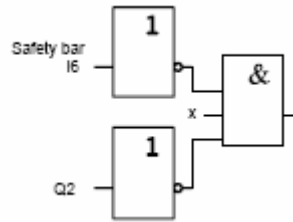
Confirm with OK.  
Now press ▼ or ▲ to invert this input: ●  
Next, press ESC.



Your circuit program layout



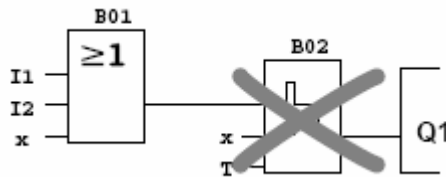
اما در ورژن های قدیمی برای not کردن ورودی باید متما، از بلوک not استفاده کنید.



### حذف کردن یک بلوک

فرض کنید می خواهیم بلوک شماره B02 را حذف و بلوک B01 را به جای آن به خروجی Q1 اتصال دهیم.

برای این کار به روش زیر عمل می کنیم.

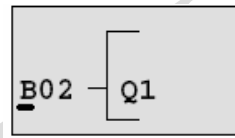


▪ logo را به حالت برنامه نویسی سویچ کنید.

▪ گزینه Edit prg را به وسیله کلیدهای جهت انتخاب کنید.

▪ با فشار دادن کلید ok انتخاب خود را تایید کنید.

▪ مکان نما ◀ را بر روی ورودی Q1 که همان بلوک B02 است قرار می دهید.



▪ OK را فشار دهید.

حالا بلوک B02 را با بلوک B01 مستقیماً روی ورودی Q1 با طی مراحل زیر جابجا می کنیم.

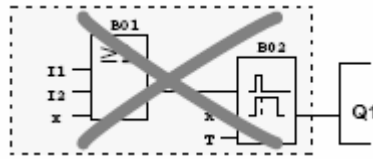
۱- لیست BN را بوسیله کلیدهای جهت انتخاب کرده و سپس OK را فشار دهید.

۲- بلوک B01 را بوسیله کلیدهای جهت انتخاب کرده و سپس OK را فشار دهید.

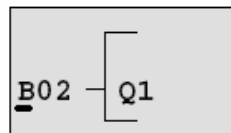
با طی مراحل بالا بلوک B02 حذف می شود. زیرا این بلوک دیگر در داخل مدار استفاده نشده

است و بلوک B01 مستقیماً به جای بلوک B02 به خروجی متصل شده است.

حذف کردن چند بلوک بهم پیوسته



- logo را به حالت برنامه نویسی سویچ کنید.
- گزینه Edit prg را به وسیله کلیدهای جهت انتخاب کنید.
- با فشار دادن کلید ok انتخاب خود را تایید کنید.
- مکان نما ◀ را بر روی ورودی Q1 که همان بلوک B02 است قرار می دهید.



- OK را فشار دهید.
- حالا به جای B02 اتصال دهنده X را از منوی CO انتخاب و برای تایید OK را فشار دهید.
- با طی مراحل بالا بلوک B02 و همه بلوک های متصل شده به B02 حذف می شوند.

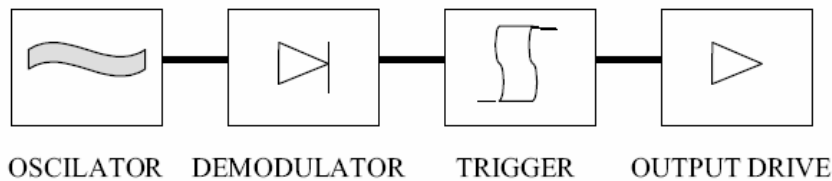
## فصل پنجم: سنسورها

انواع سنسورهای بدون تماس

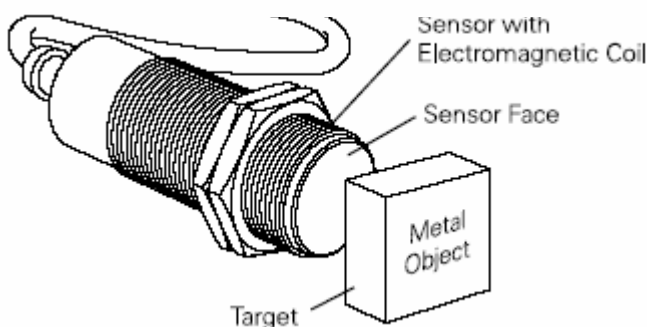
- ۱) سنسور القایی (حساس در مقابل فلزات)
- ۲) سنسور خازنی (حساس در مقابل همه چیز)
- ۳) سنسور نوری (حساس در مقابل همه چیز)
- ۴) سنسور مغناطیسی (حساس در مقابل آهنربا)
- ۵) سنسور کد رنگ (تشخیص نوار رنگی کاغذهای بسته بندی)

### ۱-۵ سوئیچهای القایی

سوئیچ القایی، سنسور بدون تماس می باشد که در مقابل فلزات عکس العمل نشان میدهد و میتواند فرمان مستقیم به رله ها، شیر برقی، سیستمهای اندازه-گیری و مدارات کنترل الکترونیکی مانند PLC ارسال نماید. این سوئیچها از چهار قسمت تشکیل می شوند.



قسمت اساسی این سوئیچها از یک اسیلاتور با فرکانس بالا تشکیل یافته که میتواند توسط قطعات فلزی تحت تاثیر قرار گیرد. این اسیلاتور باعث بوجود آمدن میدان الکترومغناطیسی در قسمت حساس سنسور میشود. نزدیک شدن یک قطعه فلزی باعث بوجود آمدن جریانهای گردابی در قطعه گردیده و این عمل سبب جذب انرژی میدان می شود و در نتیجه دامنه اسیلاتور کاهش می یابد. از آنجا که طبقه دمدولاتور، آشکار ساز دامنه اسیلاتور است، در نتیجه کاهش دامنه اسیلاتور توسط این قسمت به طبقه اشویت تریر منتقل میشود.



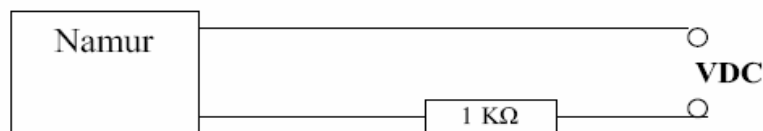
کاهش دامنه سیلاتور باعث فعال شدن خروجی اشمیت تریگر گردیده و این قسمت نیز بنوبه خود باعث تحریک طبقه خروجی می شود. سنسورهای القایی از نظر تعداد سیم، ولتاژ تغذیه و نوع خروجی به انواع زیر تقسیم می شوند:

- دو سیمه AC
- سه سیمه AC
- دو سیمه DC
- دو سیمه نامور (Namur)
- سه سیمه NPN
- سه سیمه PNP
- چهار سیمه NPN
- چهار سیمه PNP

برخی از سنسورهای القایی خاص عبارتند از:

#### - سنسور القایی نامور

سنسورهای دو سیمه هستند که مقاومت داخلی آنها بر حسب فاصله قطعه از سنسور تغییر می کند. معمولا اتصال این سنسورها به منبع تغذیه از طریق یک مقاومت  $1\text{ K}\Omega$  صورت می گیرد.



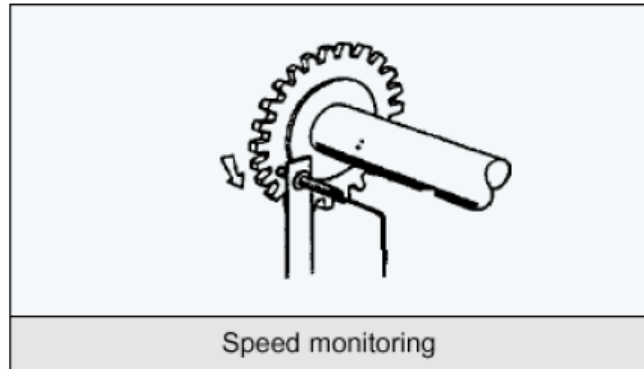
با توجه به سیم بندی مدار، جریان اتصال کوتاه در دو سر سنسور محدود می باشد، لذا بدلیل این محدودیت در محیطهای قابل انفجار میتوان از این سنسور استفاده کرد.

#### - سنسور القایی آنالوگ

سنسورهای القایی آنالوگ، سنسورهایی هستند که عکس العمل مقابل فلزات، بصورت ولتاژ و یا جریان خطی در خروجی آنها ظاهر می شود. کاربرد این سنسورها در اندازه گیری فاصله، جدا سازی قطعات با ابعاد مختلف، اندازه-گیری ضخامت قطعات فلزی و غیره می باشد.

### - سنسور القایی سرعت ( Speed Monitor )

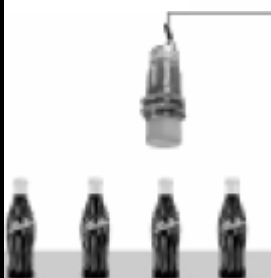
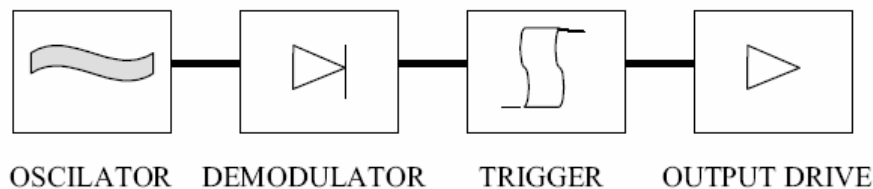
از این سنسور بمنظور اندازه گیری سرعت استفاده می شود.



در قسمت حساس این سنسورها میدان مغناطیسی وجود دارد و این میدان در اثر حرکت چرخ دنده ها تغییر میکند. از آنجا که خروجی این سنسورها سلفی میباشد، لذا این تغییرات میدان بصورت پالسهایی در خروجی ظاهر میشوند. برخی از این سنسورها احتیاج به تقویت کننده دارند. کاربرد این سنسورها در اندازه گیری سرعت موتور، لوکوموتیو، چرخ دنده، پمپ، توربین و غیره می باشد.

#### ۵-۲) سنسورهای خازنی

سنسورهای خازنی، سنسورهای بدون تماس و کنتاکت الکتریکی هستند که در مقابل فلزات و اغلب غیرفلزات عمل می نمایند. این سوئیچها برای کنترل سطوح مخازنی که از مواد پودری، مایع و یا دانه دانه پر شده اند، مناسب می باشند. همچنین از آنها میتوان بعنوان مولد پالس بمنظور کنترل وضعیت برنامه ماشین آلات، برای شمارنده ها و آشکارسازی تقریبا تمام مواد فلزی و غیر فلزی استفاده نمود. ساختمان اساسی این سنسورها از چهار قسمت تشکیل شده است.



در حالت کلی این سنسورها بدو نوع زیر تقسیم می شوند:

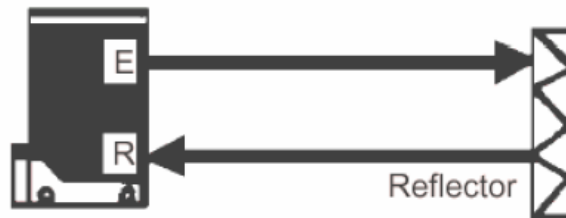
نرمال باز ( Light ON )

نرمال بسته ( Dark ON )

کاربرد این قطعه در آشکار سازی وجود اشیا، تشخیص پارگی ورق، کنترل انحراف ورق و ... می باشد.

### - سنسور نوری رفلکتوری ( Retro-Reflective )

این سنسور بر اساس ارسال امواج مادون قرمز مدوله شده توسط دیود مادون قرمز و انعکاس این امواج به وسیله رفلکتور و دریافت این امواج بوسیله فتو ترانزیستور عمل می کند.



در این سنسور امواج مادون قرمز بصورت پلاریزه شده در فضا پخش میشوند. یک منعکس کننده در جلوی سنسور در فاصله معینی قرار می گیرد و امواج ارسال شده پس از برخورد به این منعکس کننده با زاویه ۹۰ درجه نسبت به امواج پخش شده بطرف گیرنده برمی گردند. این سنسورها نیز به دو صورت Light ON و Dark ON ارائه می شوند.

کاربرد این سنسورها در شمارش تولید، تشخیص پارگی ورق، کنترل حرکت ورق و ... می باشد.

### - سنسور نوری دوطرفه ( Thru-Beam )

این سنسور بر اساس ارسال امواج مادون قرمز مدوله شده توسط دیود مادون قرمز در قسمت فرستنده و دریافت این امواج توسط فتوترانزیستور در طرف گیرنده که در مقابل فرستنده نصب می شود، عمل می نماید.



در این نوع سنسورها، فرستنده و گیرنده مجزا از هم میباشند. امواج مدوله شده مادون قرمز توسط فرستنده ارسال می شود و گیرنده در مقابل فرستنده نصب می شود. هرگاه مابین گیرنده و فرستنده مانعی وجود نداشته باشد، این امواج به گیرنده میرسند و در صورت وجود مانع این امواج دیگر به گیرنده نخواهند رسید. این سنسورها نیز به دو صورت Light ON و Dark ON ارائه میشوند.



۴-۵) سنسور مغناطیسی

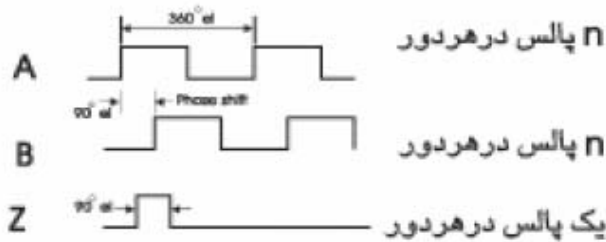
این سنسورها در مجاورت میدان مغناطیسی عمل می کنند. هرگاه یک قطعه آهنربا در مقابل این سنسور قرار گیرد، کنتاکت آن عمل خواهد کرد. کاربرد این نوع سنسور در تشخیص و کنترل سطح مایع ( با نصب یک آهنربا بر روی یک شناور )، تشخیص موقعیت پیستون درون سیلندر، اندازه گیری سرعت ( با نصب آهنربا بر روی چرخ دنده ها) و غیره می باشد.

۵-۵) سنسورهای تشخیص رنگ

در ماشین آلات صنایع غذایی، بسته بندی و غیره اغلب از سنسورهای استفاده میشود که علائم رنگی چاپ شده روی کاغذهای بسته بندی را تشخیص داده و فرمان لازم مانند قطع کاغذ را صادر کند. حساسیت به رنگ های مختلف توسط پتانسیومتر این سنسورها قابل تنظیم می باشد. این سنسورها به صورت چهار سیمه و با خروجی NPN یا PNP تولید می شوند.

### تعداد خروجی

شفت انکودر دارای سه خروجی بصورت زیر می باشد .



خروجی B نسبت به خروجی A ، 90 درجه الکتریکی اختلاف فاز دارد و در موارد زیر می تواند کاربرد داشته باشد :

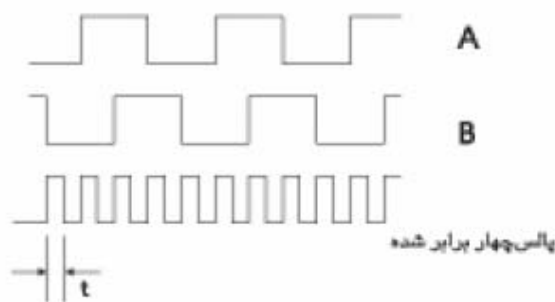
#### • تعیین جهت دوران

با استفاده از دو خروجی A و B می توان جهت دوران را به کمک یک فلیپ فلاپ تشخیص داد.



#### • چهار برابر کردن تعداد پالس در هر دور

این کار با استفاده از یک مدار الکترونیکی انجام می گیرد .



دقت کنید که پهنای پالس t باید طوری در نظر گرفته شود که در ماکزیمم سرعت دوران ، پالسها باهم تداخل ( Overlap ) نداشته باشند.

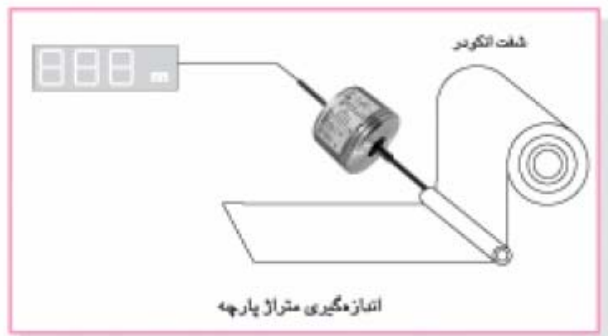
### شفت انکودر ( افزایشی ) Incremental Shaft-Encoder

شفت انکودر بمنظور تبدیل حرکت دورانی به پالسهای الکتریکی مورد استفاده قرار می گیرد.

پالسهای فرستاده شده از خروجی انکودر توسط مدارات مناسب الکترونیکی تجزیه و تحلیل می شوند.

#### کاربرد

برخی از کاربردهای شفت انکودر بشرح زیر است :



از شفت انکودر می توان در موارد متعددی استفاده کرد مانند تعیین موقعیت شفت (از نظر زاویه) ، تعیین جهت دوران و ...

#### اتصالات شفت انکودر

قرمز	+ 10-30 VDC
سیاه	( 0V )
سفید	A
آبی	B
قهوه ای	Z
زرد	

ضمیمه:

## اطلاعات فنی LOGO

## A.2 Technical data: LOGO! 230...

	LOGO! 230 RC LOGO! 230 RCo
<b>Power supply</b>	
Input voltage	115...240 V AC/DC
Permissible range	85 ... 265 V AC 100 ... 253 V DC
Permissible mains frequency	47 ... 63 Hz
Power consumption	
• 115 V AC	10 ... 40 mA
• 240 V AC	10 ... 25 mA
• 115 V DC	5 ... 25 mA
• 240 V DC	5 ... 15 mA
Voltage failure buffering	
• 115 V AC/DC	typ. 10 ms
• 240 V AC/DC	typ. 20 ms
Power loss at	
• 115 V AC	1.1 ... 4.6 W
• 240 V AC	2.4 ... 6.0 W
• 115 V DC	0.5 ... 2.9 W
• 240 V DC	1.2 ... 3.6 W
Backup of the real-time clock at 25 °C	typ. 80 h
Accuracy of the real-time clock	typ. ± 2 s / day
<b>Digital inputs</b>	
Number	8
Electrical isolation	No

	LOGO! 230 RC LOGO! 230 RC <sub>o</sub>
Input voltage L1	
<ul style="list-style-type: none"> <li>• Signal 0 &lt; 40 V AC</li> <li>• Signal 1 &gt; 79 V AC</li> <li>• Signal 0 &lt; 30 V DC</li> <li>• Signal 1 &gt; 79 V DC</li> </ul>	
Input current at	
<ul style="list-style-type: none"> <li>• Signal 0 &lt; 0.03 mA</li> <li>• Signal 1 &gt; 0.08 mA</li> </ul>	
Delay time at	
<ul style="list-style-type: none"> <li>• 0 to 1: 120 V AC typ. 50 ms : 240 V AC typ. 30 ms : 120 V DC typ. 25 ms : 240 V DC typ. 15 ms</li> <li>• 1 to 0: 120 V AC typ. 65 ms : 240 V AC typ. 105 ms : 120 V DC typ. 95 ms : 240 V DC typ. 125 ms</li> </ul>	
Line length (unshielded)	100 m
<b>Digital outputs</b>	
Number	4
Output type	Relay outputs
Electrical isolation	Yes
In groups of	1
Control of a digital input	Yes
Continuous current I <sub>th</sub>	max. 10 A per relay
Surge current	max. 30 A
Incandescent lamp load (25000 switching cycles) at	

	LOGO! 230 RC LOGO! 230 RCo
230/240 V AC 115/120 V AC	1000 W 500 W
Fluorescent tubes with ballast (25000 switching cycles)	10 x 58 W (at 230/240 V AC)
Fluorescent tubes, conven- tionally compensated (25000 switching cycles)	1 x 58 W (at 230/240 V AC)
Fluorescent tubes, uncom- pensated (25000 switching cycles)	10 x 58 W (at 230/240 V AC)
Short circuit-proof cos 1	Power protection B16 600A
Short circuit-proof cos 0.5 to 0.7	Power protection B16 900A
Derating	none; across the entire temperature range
Parallel output circuits for power increase	Not permitted
Protection of output relay (if desired)	max. 16 A, characteristic B16
<b>Switching rate</b>	
Mechanical	10 Hz
Ohmic load/lamp load	2 Hz
Inductive load	0.5 Hz

## A.4 Technical data: LOGO! 24...

	LOGO! 24 LOGO! 24o
<b>Power supply</b>	
Input voltage	24 V DC
Permissible range	20.4 ... 28.8 V DC
Reverse polarity protection	Yes
Permissible mains frequency	
Power consumption from 24 V DC	30 ... 55 mA 0.3 A per output
Voltage failure buffering	
Power loss at 24 V	0.7 ... 1.3 W
Backup of the real-time clock at 25 °C	
Accuracy of the real-time clock	
<b>Digital inputs</b>	
Number	8
Electrical isolation	No
Input voltage	L+
• Signal 0	< 5 V DC
• Signal 1	> 12 V DC
Input current at	
• Signal 0	< 0,85 mA (I1...I6) < 0.05 mA (I7, I8)
• Signal 1	> 2 mA (I1... I6) > 0.15 mA (I7, I8)

	LOGO! 24 LOGO! 24o
Delay time at	
<ul style="list-style-type: none"> <li>• 0 to 1</li> <li>• 1 to 0</li> </ul>	typ. 1.5 ms <1.0 ms (I5, I6) typ. 1.5 ms <1.0 ms (I5, I6)
Line length (unshielded)	100 m
<b>Analog inputs</b>	
Number	2 (I7 = AI1, I8 = AI2)
Range	0 ... 10 V DC input impedance 78 kΩ
Cycle time for analog value generation	300 ms
max. input voltage	28.8 V DC
Line length (shielded and twisted)	10 m
Error limit	+/- 1.5%
<b>Digital outputs</b>	
Number	4
Output type	Transistor, current-sourcing <sup>(1)</sup>
Electrical isolation	No
In groups of	
Control of a digital input	Yes
Output voltage	△ Supply voltage
Output current	max. 0.3 A
Continuous current I <sub>th</sub>	
Incandescent lamp load (25000 switching cycles) at	
Fluorescent tubes with ballast (25000 switching cycles)	

	LOGO! 24 LOGO! 24o
Fluorescent tubes, conventionally compensated (25000 switching cycles)	
Fluorescent tubes, uncompensated (25000 switching cycles)	
Short circuit-proof and overload-proof	Yes
Short circuit current limitation	Approx. 1 A
Derating	none; across the entire temperature range
Short circuit-proof cos 1	
Short circuit-proof cos 0.5 to 0.7	
Parallel output circuit for power increase	Not permitted
Protection of output relay (if desired)	
<b>Switching rate</b> <sup>(2)</sup>	
Mechanical	
Electrical	10 Hz
Ohmic load/lamp load	10 Hz
Inductive load	0.5 Hz



## A.4 Technical Specifications: LOGO! 24 Long

	LOGO! 24L	LOGO! 24RCL LOGO! 24RCLB11
<b>Power supply</b>		
Input voltage	24 V DC	24 V DC
Permissible range	20.4 to 28.8 V DC	20.4 to 28.8 V DC
Power consumption from 24 V DC	10 ... 30 mA + 0.3 A per output	15 ... 120 mA
Voltage failure bridging		Typically 5 ms
Power loss at 24 V DC	0.2 ... 0.8 W	0.3 ... 2.9 W
Clock buffering at 25 °C		Typically 80 h
Accuracy of the real-time clock		Max. ± 5 s a day
Electrical isolation	No	No
Reverse polarity protection	Yes	Yes
<b>Digital inputs</b>		
Number	12	12
Electrical isolation	No	No
Input voltage L+		
• Signal 0	< 5 V DC	< 5 V DC
• Signal 1	> 12 V DC	> 12 V DC
Input current at		
• Signal 0	< 1.5 mA	< 1.5 mA
• Signal 1	> 4.5 mA	> 4.5 mA
Delay time at		
• 0 after 1	Typically 1.5 ms	Typically 1.5 ms
• 1 after 0	Typically 1.5 ms	Typically 1.5 ms
Line length (unshielded)	100 m	100 m

	LOGO! 24L	LOGO! 24RCL LOGO! 24RCLB11
Digital outputs		
Number	8	8
Output type	Transistor, current-sourcing	Relay outputs
Electrical isolation	No	Yes
In groups of		2
Activation of digital input	Yes	Yes
Output voltage	$\triangle$ Supply voltage	
Output current	Max. 0.3 A	
Continuous current $I_{th}$ (per connector)		Max. 10 A
Incandescent lamp load (25,000 switching cycles) at		1000 W
Fluorescent tubes with electr. control gear (25,000 switching cycles)		10 x 58 W
Fluorescent tubes, conventionally compensated (25,000 switching cycles)		1 x 58 W
Fluorescent tubes, uncompensated (25,000 switching cycles)		10 x 58 W
Short circuit-proof and overload-proof	Yes	
Short-circuit current limitation	Approx. 1 A	
Derating	None throughout the entire temperature range	None throughout the entire temperature range
Short circuit-proof cos 1		Power protection B16 600A

## A.6 Technical data: LOGO! 24RC...

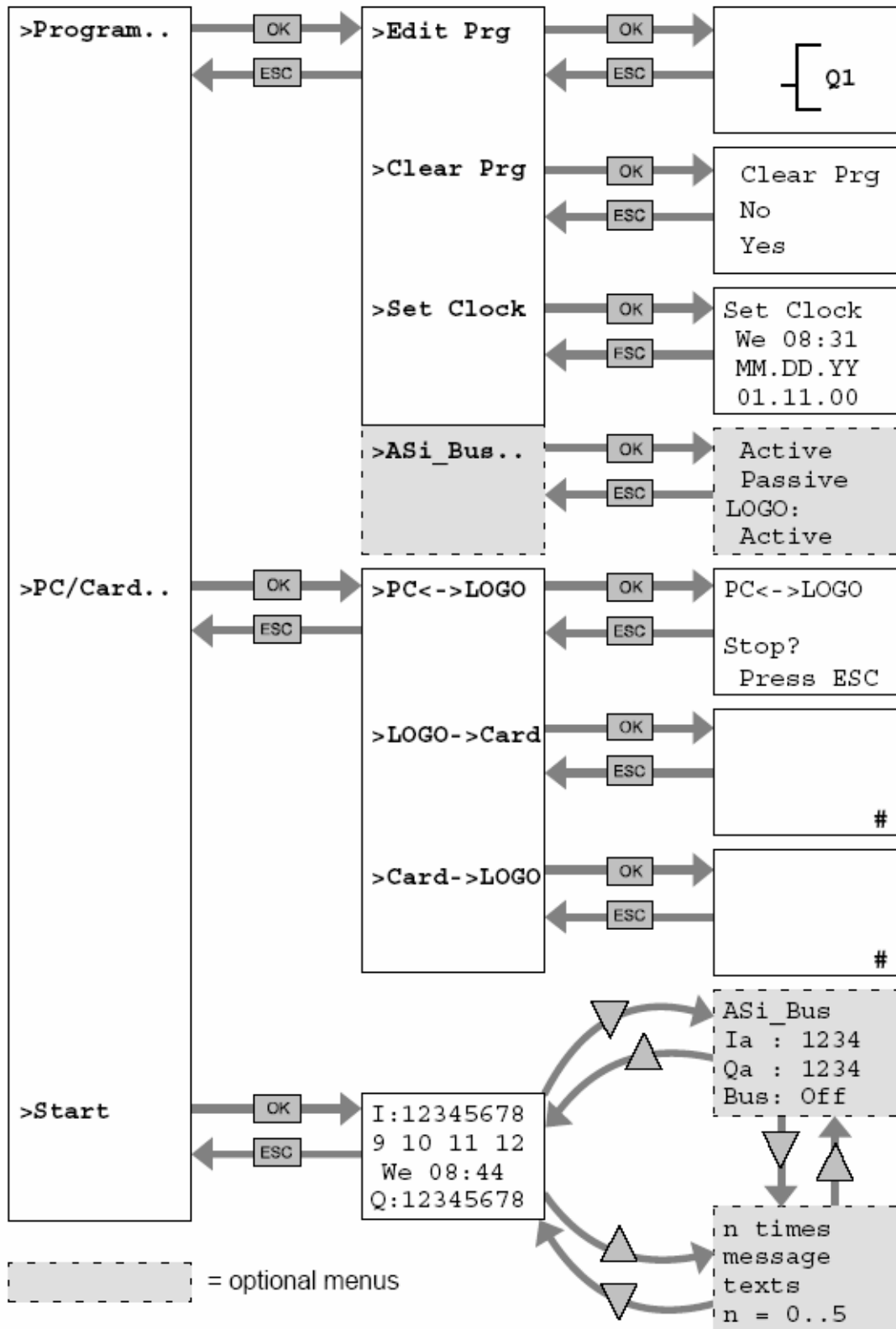
	LOGO! 24RC LOGO! 24RCo
<b>Power supply</b>	
Input voltage	24 V AC/DC
Permissible range	20.4 ... 26.4 V AC 20.4 ... 28.8 V DC
Reverse polarity protection	
Permissible mains frequency	47 ... 63 Hz
Power consumption	
• 24 V AC	40 ... 110 mA
• 24 V DC	20 ... 75 mA
Voltage failure buffering	typ. 5 ms
Power loss	
• 24 V AC	0.9 ... 2.7 W
• 24 V DC	0.4 ... 1.8 W
Backup of the real-time clock at 25 °C	typ. 80 h
Accuracy of the real-time clock	typ. $\pm 2$ s / day
<b>Digital inputs</b>	
Number	8, optional P action or N action
Electrical isolation	No
Input voltage	L
• Signal 0	< 5 V AC/DC
• Signal 1	> 12 V AC/DC

	LOGO! 24RC LOGO! 24RCo
Input current at	
• Signal 0	< 1.0 mA
• Signal 1	> 2.5 mA
Delay time at	
• 0 to 1	typ. 1.5 ms
• 1 to 0	typ. 15 ms
Line length (unshielded)	100 m
<b>Analog inputs</b>	
Number	
Range	
max. Input voltage	
<b>Digital outputs</b>	
Number	4
Output type	Relay outputs
Electrical isolation	Yes
In groups of	1
Control of a digital input	Yes
Output voltage	
Output current	
Continuous current $I_{th}$	max. 10 A per relay
Surge current	max. 30 A
Incandescent lamp load (25000 switching cycles) at	1000 W
Fluorescent tubes with ballast (25000 switching cycles)	10 x 58 W

## Technical data: LOGO!Power 24 V

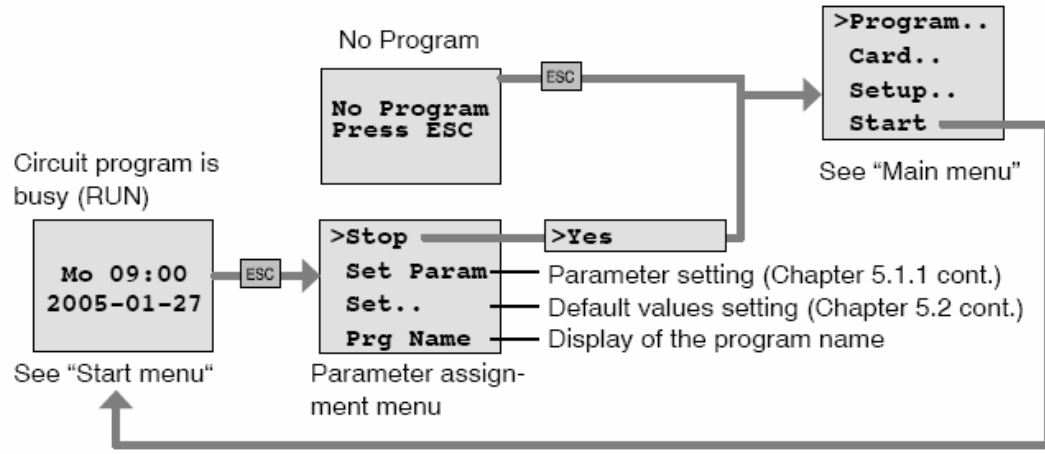
	LOGO! Power 24 V / 1.3 A	LOGO! Power 24 V / 2.5 A
<b>Input data</b>		
Input voltage	100 ... 240 V AC	
Permissible range	85 ... 264 V AC	
Permissible mains frequency	47 ... 63 Hz	
Voltage failure buffering	40 ms (at 187 V AC)	
Input current	0.70 ... 0.35 A	1.22 ... 0.66 A
Inrush current (25 °C)	< 15 A	< 30 A
Device protection	Internal	
Recommended circuit breaker (IEC 898) in mains line	≥ 16 A characteristic B ≥ 10 A characteristic C	
<b>Output data</b>		
Output voltage	24 V DC	
Overall tolerance	+/- 3 %	
Adjustment range	22.2 ... 26.4 V DC	
Residual ripple	< 200/300 mV <sub>pp</sub>	
Output current	1.3 A	2.5 A
Overcurrent limiting	typ. 2.0 A	typ. 3.4 A
Efficiency	> 82 %	> 87 %
Parallel circuit for power increase	Yes	
<b>Electromagnetic compatibility</b>		
Interference suppression	EN 50081-1, Class B to EN 55022	
Interference immunity	EN 61000-6-2, EN 61000-4-2/-3/-4/-5/-6/-11	

# LOGO! Menu Structure

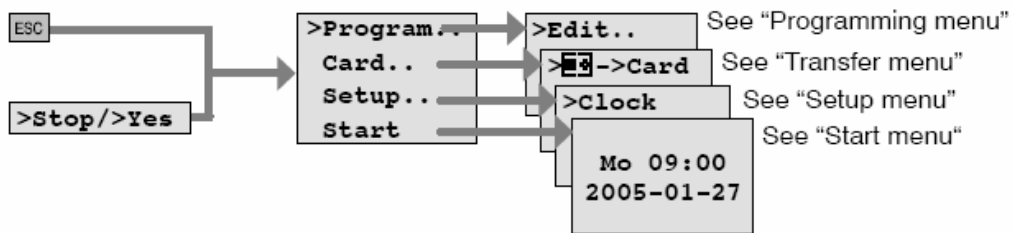


## D LOGO! menu structure

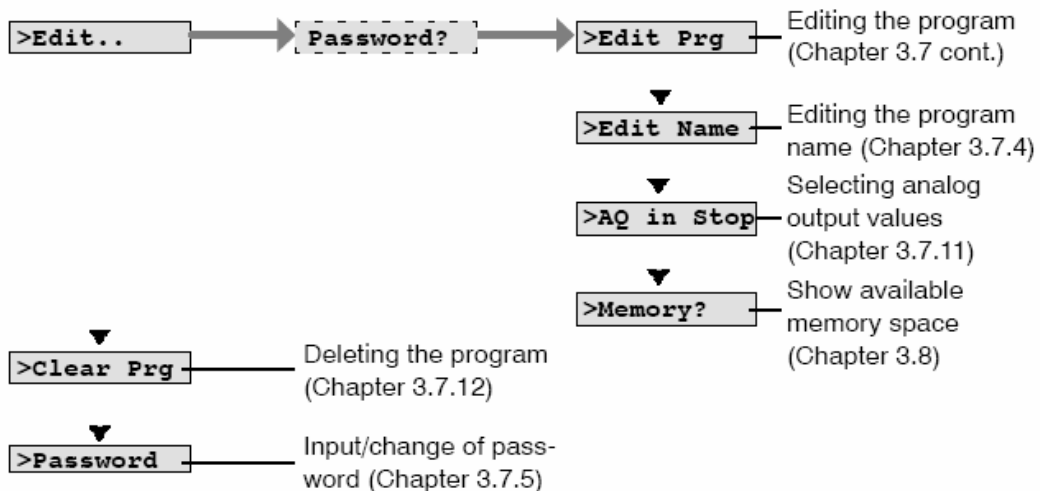
### Menu overview



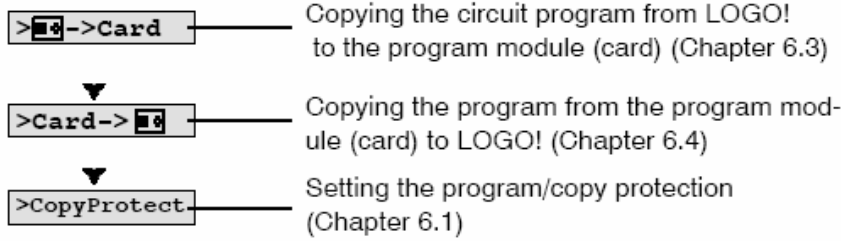
### Main menu (ESC / >Stop)



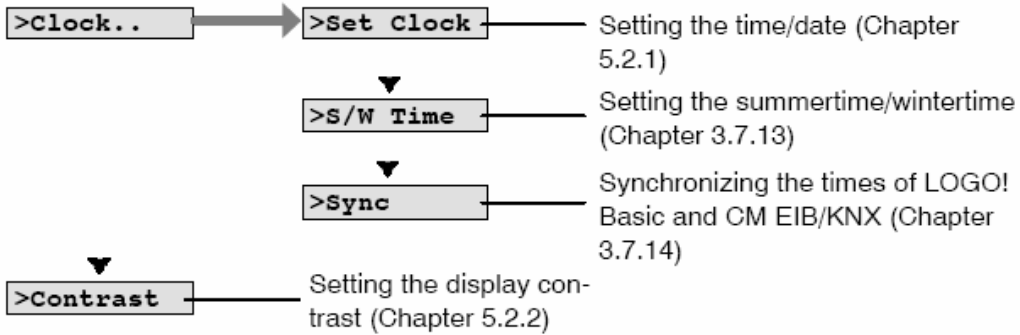
### Programming menu (ESC / >Stop → >Program)



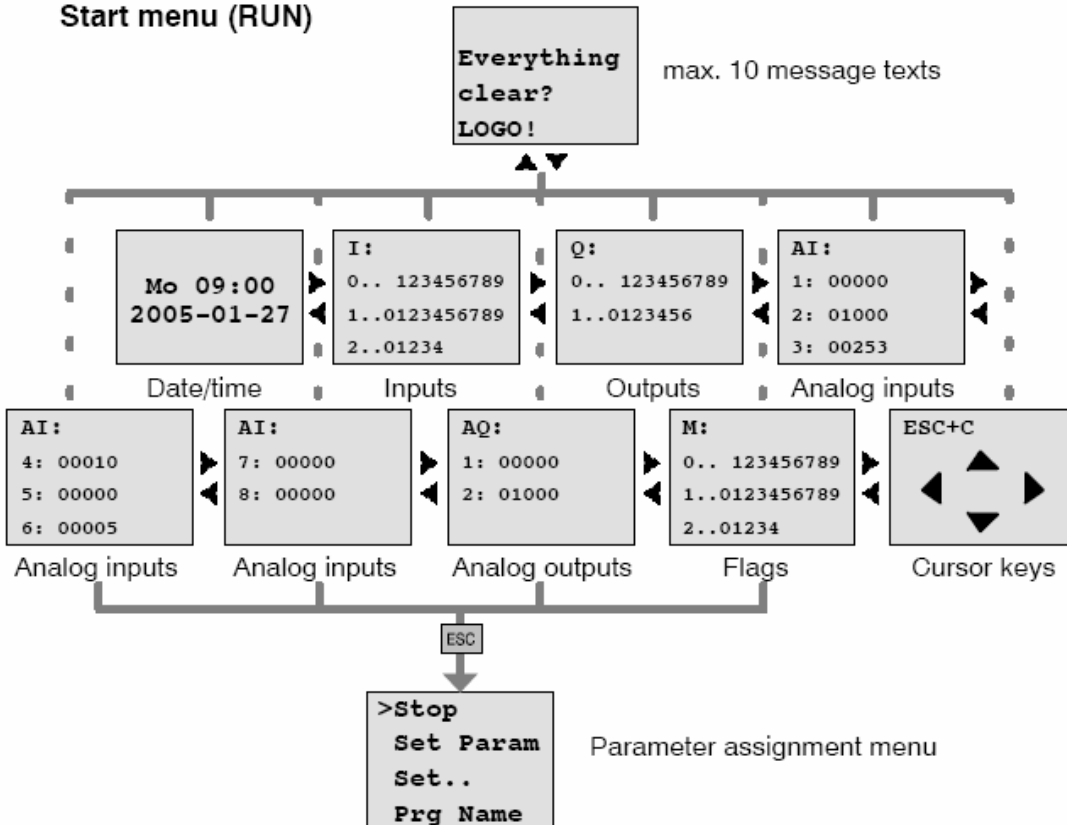
**Transfer menu (ESC / >Stop → >Card)**



**Setup menu (ESC / >Stop → >Setup)**



**Start menu (RUN)**





وضعیت حافظه برای ورژن ۳

Blocks	Par	RAM	Timer	REM	Markers
56	48	27	16	15	8

Function block	Par	RAM	Timer	REM
Latching relay*	0	(1)	0	(1)
Current impulse relay*	0	(1)	0	(1)
Interval time-delay relay	1	1	1	0
Edge-triggered interval time-delay relay	1	1	1	0
On delay	1	1	1	0
Off delay	2	1	1	0
On/off delay	2	1	1	0
Retentive on delay	2	1	1	0
Seven-day time switch	6	2	0	0
Twelve-month time switch	2	0	0	0
Up/down counter*	2	(2)	0	(2)
Operating hours counter	2	0	0	4
Symmetrical clock pulse generator	1	1	1	0
Asynchronous pulse generator	3	1	1	0
Random generator	2	1	1	0
Frequency trigger	3	3	1	0
Analog trigger	4	2	0	0
Analog comparator	3	4	0	0
Stairwell light switch	1	1	1	0
Dual-function switch	2	1	1	0
Message texts	1	0	0	0

وضعیت حافظه برای ورژن ۵

Bytes	Blocks	REM
2000	130	60

ادامه جدول

Function	Program memory	Rem memory*
<b>Basic functions</b>		
AND	12	-
AND with edge evaluation	12	-
NAND (not AND)	12	-
NAND with edge evaluation	12	-
OR	12	-
NOR (not OR)	12	-
XOR (exclusive OR)	8	-
NOT (Negation)	4	-
<b>Special functions</b>		
Times		
On-delay	8	3
Off-delay	12	3
On-/Off-delay	12	3
Retentive on-delay	12	3
Wiping relay (pulse output)	8	3
Edge triggered wiping relay	16	4
Asynchronous pulse generator	12	3
Random generator	12	-
Stairway lighting switch	12	3
Multiple function switch	16	3
Weekly timer	20	-
Yearly timer	8	-

Function	Program memory	Rem memory*
Counter		
Up/down counter	24	5
Hours counter	24	9
Threshold trigger	16	-
Analog		
Analog threshold trigger	16	-
Analog differential trigger	16	-
Analog comparator	24	-
Analog value monitoring	20	-
Analog amplifier	12	-
Analog multiplexer	20	-
Analog ramp	36	-
PI controller	40	2
Others		
Latching relay	8	1
Pulse relay	12	1
Message texts	8	-
Softkey	8	2
Shift register	12	1

\*: Bytes in the Rem memory area if retentivity is enabled.