

## شروع برنامه نویسی با نرم افزار WINPROLADER

### حافظه داخلی FATEK PLC :

در این پی ال سی ها ، رجیسترها بصورت تک بیتی یا ۱۶بیتی قابل دسترسی می باشند (رجیسترهای ۳۲بیتی از استفاده دو رجیستر ۱۶بیتی بوجود می آیند)

رجیسترها دارای دو دسته هستند :

❖ رجیسترهای معمولی : این رجیسترها توسط برنامه نویس در برنامه برای نگهداری اطلاعات و اعداد قابل استفاده هستند و بر دو دسته می باشند

۱- Retentive : مقدار موجود در این رجیسترها با قطع و وصل برق صفر نمی شوند و ماندگار می مانند.

۲- Non retentive : مقدار موجود در این رجیسترها با قطع و وصل برق پاک (صفر) می شوند .

❖ رجیسترهای خاص : (Special Registers) این رجیسترها رابط بین اطلاعات CPU و برنامه کاربر هستند برای مثال CPU مقدار ساعت و تاریخ داخلی PLC در رجیسترهای R4128~R4133 قابل دسترسی می باشد.

رجیسترهای بیتی :

X	ورودی	X0 ~ X255
Y	خروجی	Y0 ~ Y255
M	رجیسترهای بیتی معمولی غیر ماندگار	M1400~M1911
	رجیسترهای بیتی معمولی قابل تنظیم برای ماندگار بودن یا غیر ماندگار بودن	M0~M1399
	رجیسترهای بیتی خاص	M1912 ~ M2001

تعدادی از رجیسترهای بیتی خاص :

(تمام رجیسترهای خاص از مسیر **Special Registers >> HELP** قابل دسترسی می باشند)

توضیحات	شماره رجیستر بیتی خاص
با یک شدن این بیت CPU به حالت استپ می رود و تمام خروجی ها خاموش می شوند . با خاموش و روشن کردن PLC دوباره RUN می شود	M1912
با یک شدن این بیت خروجی های سخت افزاری خاموش می شوند ولی وضعیت خروجی ها در برنامه در همان حالت باقی می ماند.	M1913
با یک شدن این بیت ، رجیسترهای بیتی غیر ماندگار صفر می شوند	M1914
با یک شدن این بیت ، رجیسترهای بیتی ماندگار صفر می شوند	M1915
با یک شدن این بیت ، رجیسترهای ۱۶بیتی غیر ماندگار صفر می شوند	M1916
با یک شدن این بیت ، رجیسترهای ۱۶بیتی ماندگار صفر می شوند	M1917
بوسیله این بیت ، CPU هر ۰.۰۱ ثانیه یک پالس تولید می کند.	M1920
بوسیله این بیت ، CPU هر ۰.۱ ثانیه یک پالس تولید می کند.	M1921
بوسیله این بیت ، CPU هر ۱ ثانیه یک پالس تولید می کند.	M1922
بوسیله این بیت ، CPU هر ۶۰ ثانیه یک پالس تولید می کند.	M1923
وقتی CPU از حالت STOP به RUN یا از حالت خاموش به روشن می رود ، فقط در اسکن اول برنامه این بیت ۱ می شود و تا آخر صفر می ماند.	M1924
برای تغییر ساعت و تاریخ داخلی PLC این بیت باید ابتدا ۱ شود و پس از تنظیم ساعت و تاریخ ، باید صفر شود .	M1952
با یک شدن این بیت ، وقتی تایمر به مقدار تنظیم شده رسید در این مقدار باقی می ماند.	M1957

رجیسترهای ۱۶ بیتی :

D	رجیسترهای ۱۶بیتی معمولی	D0~D4000
	رجیسترهای ۱۶بیتی خاص	D4001~D4095
R	رجیسترهای ۱۶بیتی معمولی قابل تنظیم برای ماندگار بودن یا غیر ماندگار بودن	R0~R3839
	رجیسترهای ۱۶بیتی معمولی ماندگار	R5000~R8071
	رجیسترهای ۱۶بیتی خاص	R3840~R4200

تعدادی از رجیسترهای ۱۶بیتی خاص :

(تمام رجیسترهای خاص از مسیر Special Registers >> HELP قابل دسترسی می باشند)

Register No.	توضیحات
R3840 ~ R3903	رجیسترهای ورودی آنالوگ
R3904 ~ R3967	رجیسترهای خروجی آنالوگ
R4050	تنظیم Baud Rate پورت ۰
R4146	مقدار مربوط به تنظیمات پارامترهای پورت ۱
R4158	مقدار مربوط به تنظیمات پارامترهای پورت ۲
R4161	مقدار مربوط به تنظیمات پارامترهای پورت ۲ High Speed
R4043	مقدار مربوط به تنظیمات پارامترهای پورت ۳
R4044	مقدار مربوط به تنظیمات پارامترهای پورت ۴
R4047	تنظیم پروتکل های ارتباطی پورتهای ۱-۴
R4055	۸ بیت کم ارزش شماره استیشن می باشد (مقدار ۸ بیتهای با ارزش را برابر با مقدار H۵۵ قرار می دهیم)
R4128	"پارامتر ثانیه" مربوط به ساعت داخلی CPU
R4129	"پارامتر دقیقه" مربوط به ساعت داخلی CPU
R4130	"پارامتر ساعت" مربوط به ساعت داخلی CPU
R4131	"پارامتر روز" مربوط به تاریخ داخلی CPU
R4132	"پارامتر ماه" مربوط به تاریخ داخلی CPU
R4133	"پارامتر سال" مربوط به تاریخ داخلی CPU
R4134	چندمین روز هفته
R4136	زمان اسکن برنامه
R4164	رجیستر V برای آدرس دهی غیر مستقیم
R4165	رجیستر Z برای آدرس دهی غیر مستقیم
D4080	رجیستر P0 برای آدرس دهی غیر مستقیم
D4081	رجیستر P1 برای آدرس دهی غیر مستقیم
D4082	رجیستر P2 برای آدرس دهی غیر مستقیم
D4083	رجیستر P3 برای آدرس دهی غیر مستقیم
D4084	رجیستر P4 برای آدرس دهی غیر مستقیم
D4085	رجیستر P5 برای آدرس دهی غیر مستقیم
D4086	رجیستر P6 برای آدرس دهی غیر مستقیم
D4087	رجیستر P7 برای آدرس دهی غیر مستقیم
D4088	رجیستر P8 برای آدرس دهی غیر مستقیم
D4089	رجیستر P9 برای آدرس دهی غیر مستقیم

برنامه نویسی PLC به زبان LADDER :

نحوه ی انجام عملیات در سیستم PLC به صورت زیر است :

PLC تمام ورودی ها را چک می کند (Scan Inputs)، ورودی هایی که وصل هستند از نظر PLC معادل “یک” و ورودی هایی که قطع هستند معادل “صفر” قرار داده می شوند .

CPU برنامه موجود در حافظه را خط به خط خوانده و اجرا می کند و پس از پایان اجرای برنامه ، وضعیت خروجی ها را به واحد خروجی می فرستد و این سیکل مجدداً از ابتدا آغاز می شود.

کل زمان انجام مراحل ۱ تا ۳ برابر است با  $\text{Scan Inputs} + \text{Scan Program} + \text{Scan Outputs}$  و آن را Scan Time می نامند.

چنانچه این زمان بیشتر از 0.25 ثانیه گردد، نشان دهنده ی این مطلب می باشد، که یکی از قسمت های PLC دچار اشکال شده بنابراین تایمر سگ نگهبان (Watch Dog Timer) عمل نموده و تمامی خروجی ها را غیرفعال می کند تا عملکرد اشتباه PLC منجر به حادثه نگردد. این زمان پیش فرض، از طریق تابع ۹۰ قابل تغییر است .

فرض کنید که در یک برنامه باید با وصل یک ورودی ، یک خروجی فعال گردد. حال اگر تصادفاً ورودی در لحظه ای وصل شود که PLC ، مرحله خواندن ورودی ها را به انجام رسانده باشد، در این صورت باید به اندازه ی یک اسکن کامل صبر کند تا وضعیت این ورودی به CPU انتقال یابد ، این تاخیر را تاخیر نرم افزاری PLC می نامند.

از طرف دیگر به دلیل نویزهای موجود در محیط های صنعتی ، ورودی ها عموماً دارای فیلتری می باشند که این نیز به نوبه ی خود تاخیری را در دریافت ورودی ایجاد می نماید (حدود 10ms )، همچنین اگر خروجی از نوع رله ای باشد مدت زمانی حدود 10ms نیز برای وصل رله ی خروجی خواهیم داشت ، مجموع این دو زمان را تاخیر سخت افزاری PLC می نامند.

بنابراین پاسخ زمانی PLC حاصل جمع تاخیر نرم افزاری و سخت افزاری موجود در آن می باشد.

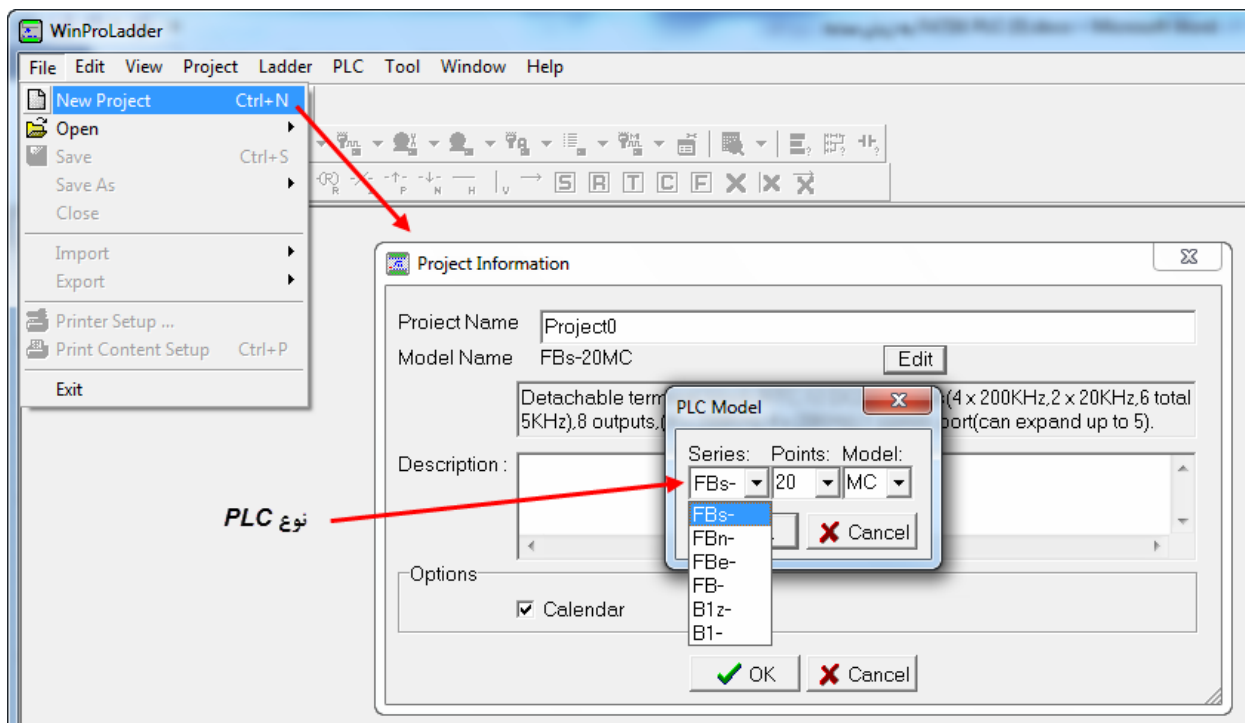
نرم افزار برنامه نویسی FATEK PLC “WinProladder” می باشد. کاربر بوسیله این نرم افزار می تواند مستقیماً برنامه موجود در

حافظه PLC را مشاهده و تغییر دهد و یا ابتدا برنامه را در داخل کامپیوتر شخصی بنویسد و سپس در موقع مناسب آن را به PLC منتقل نماید و قابلیت اجرای برنامه “RUN” یا “STOP” از محیط این نرم افزار انجام می شود .

برخی قابلیت های نرم افزار برنامه نویسی FATEK (Win Proladder) به شرح زیر می باشد:

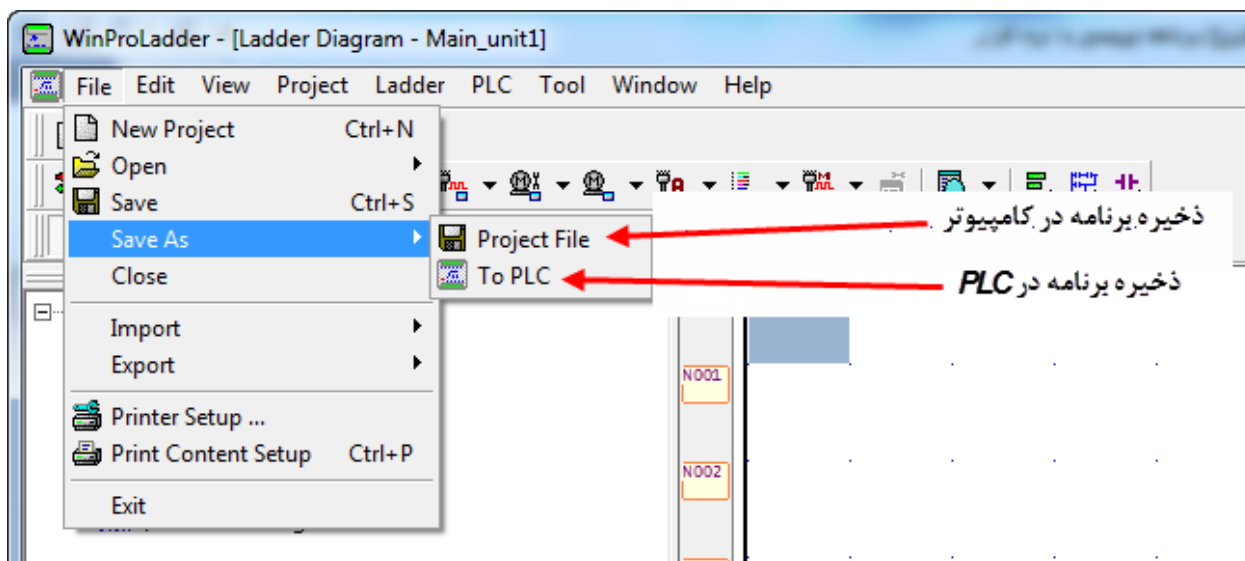
- ❖ امکان نوشتن برنامه به صورت **Off Line** و ذخیره آن به صورت یک فایل جهت دسترسی دوباره به برنامه فوق .
  - ❖ مشاهده ی اجرای یک برنامه روی PLC (**On Line Monitoring**).
  - ❖ مشاهده ی اجرای یک برنامه بدون استفاده از PLC (**Offline Simulation**)
  - ❖ قابلیت قطع و وصل هر ورودی یا فعال و غیرفعال کردن هر خروجی .
  - ❖ امکان تغییر برنامه در حالت (**RUN**) ، که از قابلیت های منحصر به فرد این نرم افزار می باشد.
  - ❖ امکان نظارت و تغییر حافظه ی داخلی PLC از طریق صفحه ی مانیتورینگ (**Status Page**)
  - ❖ امکان پیدا کردن سریع هر ورودی یا خروجی دلخواه (**search**) در برنامه و جایگزین نمودن آن ها.
  - ❖ امکان قرار دادن توضیحات اضافی در برنامه (**Comments**)
  - ❖ امکان قرار دادن رمز (**Password**) برای کل برنامه یا فقط زیر برنامه ها.
  - ❖ امکان اتصال PLC و PC با روش های متنوع (اتصال مستقیم از طریق **RS232** ، **USB** ، **Ethernet** و اتصال از راه دور به کمک مودم خط تلفن ).
- معرفی منوهای نرم افزار **Winprollader** :
- ایجاد پروژه جدید :
- از منوی **File** و انتخاب گزینه **New Project** و انتخاب نوع PLC می توان پروژه جدیدی را ایجاد کرد و در آن برنامه مورد نظر را نوشت .

www.eza-co.com

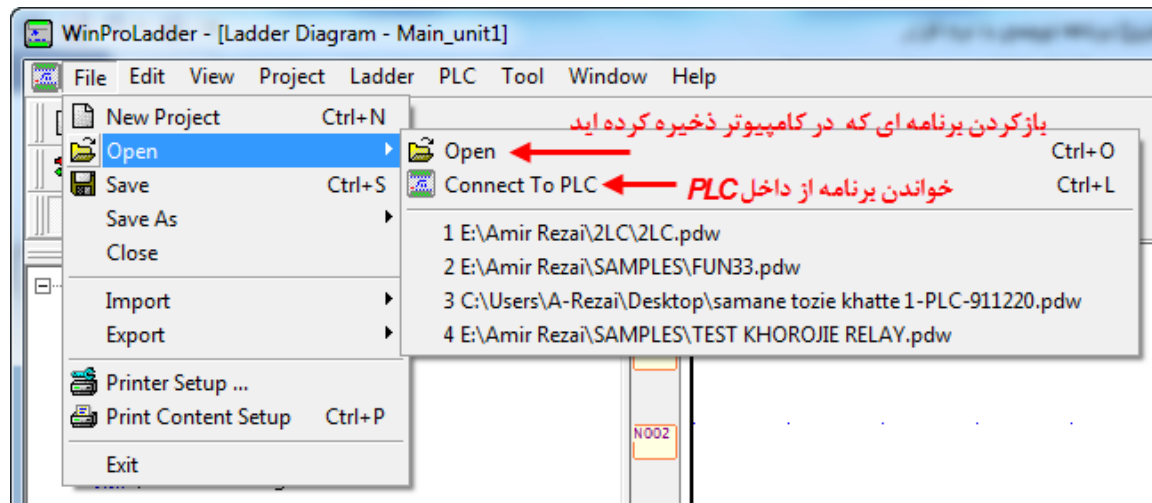


ذخیره برنامه :

در منوی File با استفاده از گزینه **Save As >> Project File** می توان برنامه را بعنوان یک فایل با فرمت **.pdw** در کامپیوتر ذخیره کرد یا با استفاده از گزینه **Save As >> To PLC** برنامه نوشته شده را در حافظه PLC ذخیره کرد.

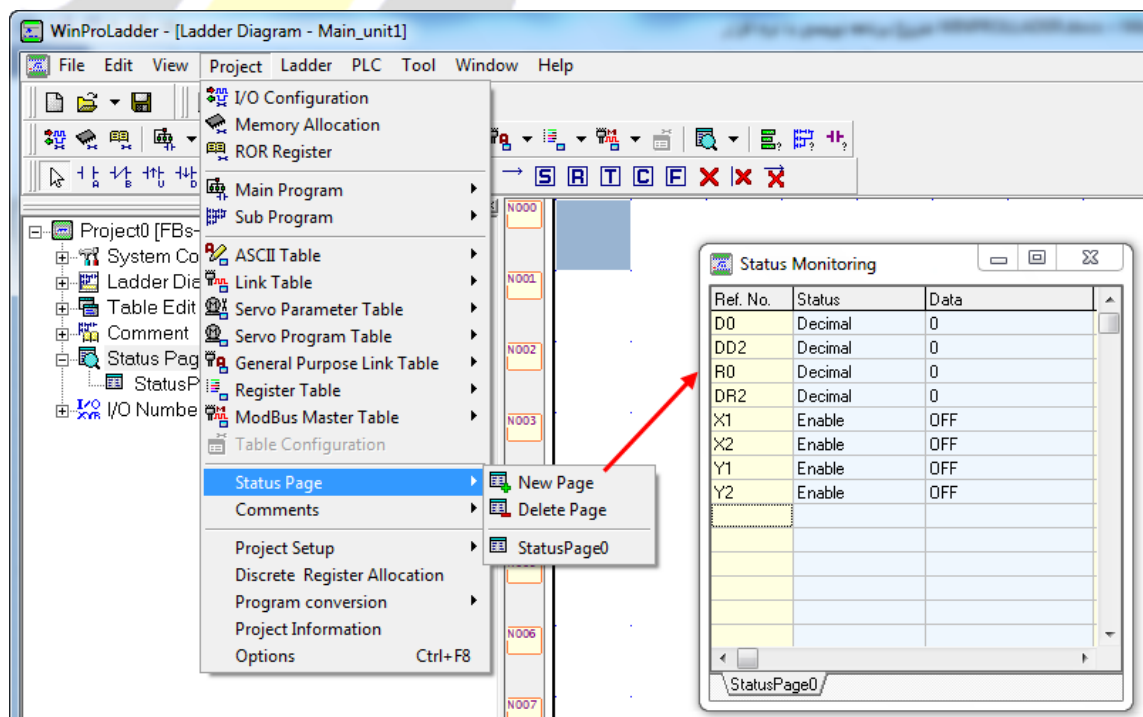


**باز کردن برنامه یا خواندن برنامه از PLC:**



### مانیتور کردن رجیسترها و ورودی ها و خروجی های PLC:

با استفاده از صفحات Status Page می‌توان نام رجیسترها یا ورودی و خروجی های مورد نیاز را در ستون Ref. No. نوشت و مقادیر جاری آنها را مشاهده یا تغییر داد.



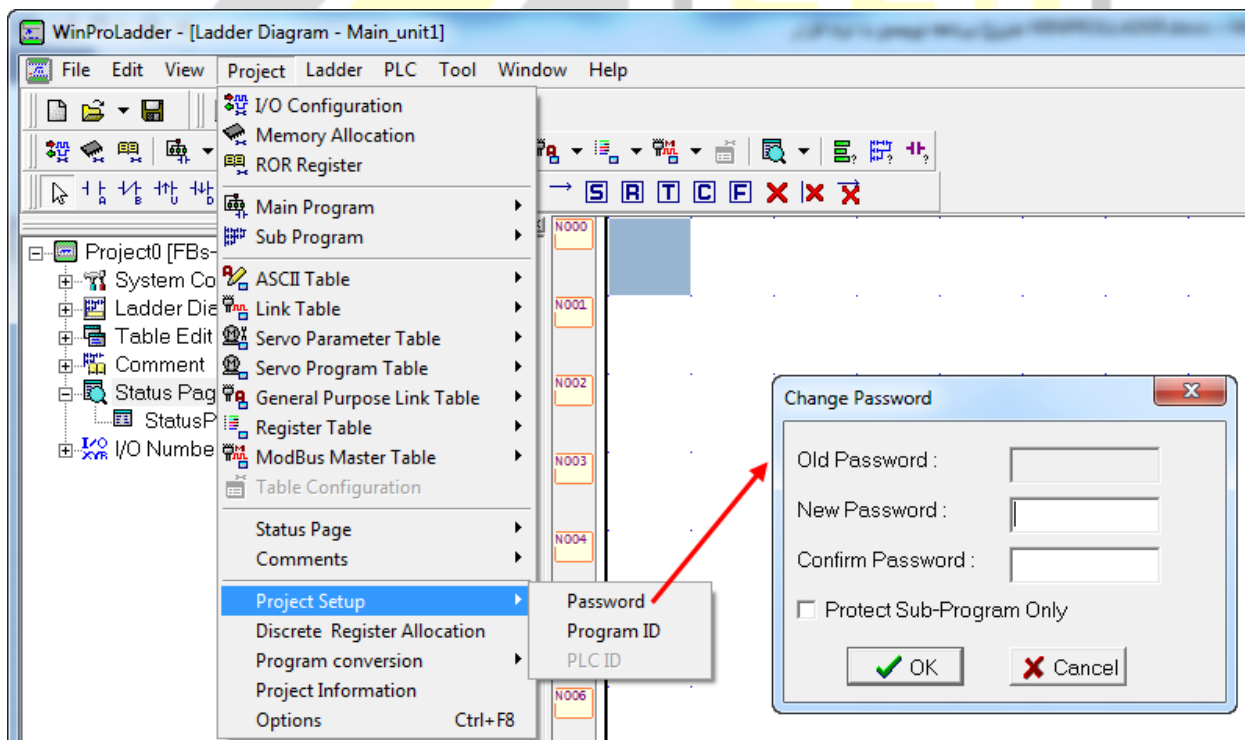
اختصاص کلمه عبور به پروژه :

دو نوع کلمه عبور را می توان به برنامه نوشته اختصاص داد :

- ۱- کلمه عبور به کل برنامه : با انتخاب نکردن عبارت **Protect Sub-program Only** ، کلمه عبور به کل برنامه های نوشته شده اختصاص می یابد ، در این حالت برای اینکه برنامه را باز کنیم ابتدا باید کلمه عبور را وارد کنیم .
- ۲- اختصاص دادن کلمه عبور به زیر برنامه ها : با تیک زدن عبارت **Protect Sub-program Only** کلمه عبور به زیر برنامه های نوشته شده اختصاص می یابد ، در این حالت برنامه باز می شود ولی فقط صفحه **Main** برنامه باز می شود و برای باز کردن صفحات زیر برنامه باید کلمه عبور را وارد کنیم .

PLC ID :

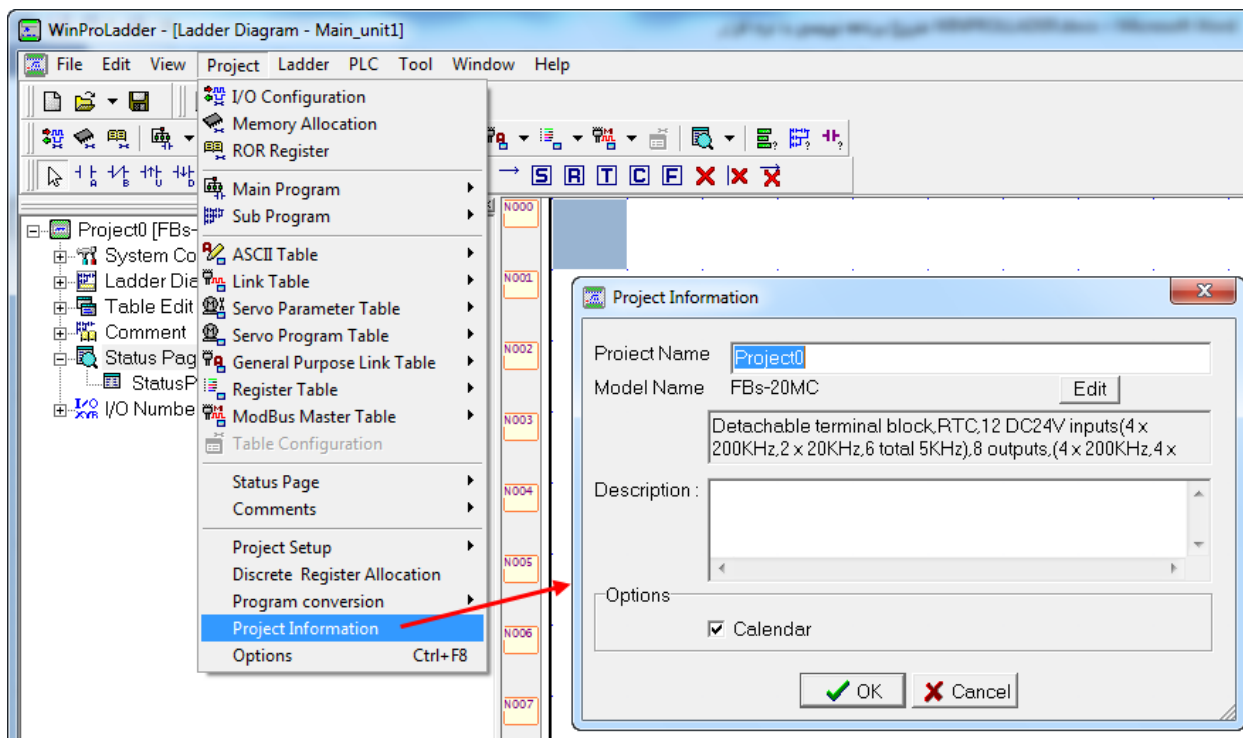
در PLC های FATEK می توان به برنامه نوشته شده یک مشخصه (ID) که می تواند شامل حروف و اعداد باشد اختصاص داده می شود و به PLC نیز یک مشخصه اختصاص داده می شود، اگر برنامه ای را بخواهیم در یک PLC اجرا کنیم باید **Program ID** و **PLC ID** با یکدیگر برابر باشند . گزینه **PLC ID** زمانی که به PLC آنلاین باشیم فعال می شود.





تغییر نوع PLC در برنامه :

در منوی Project و با انتخاب گزینه Project information می توان صفحه مشخصات PLC معرفی شده به برنامه را باز کرد و نوع PLC انتخاب شده را عوض کرد .

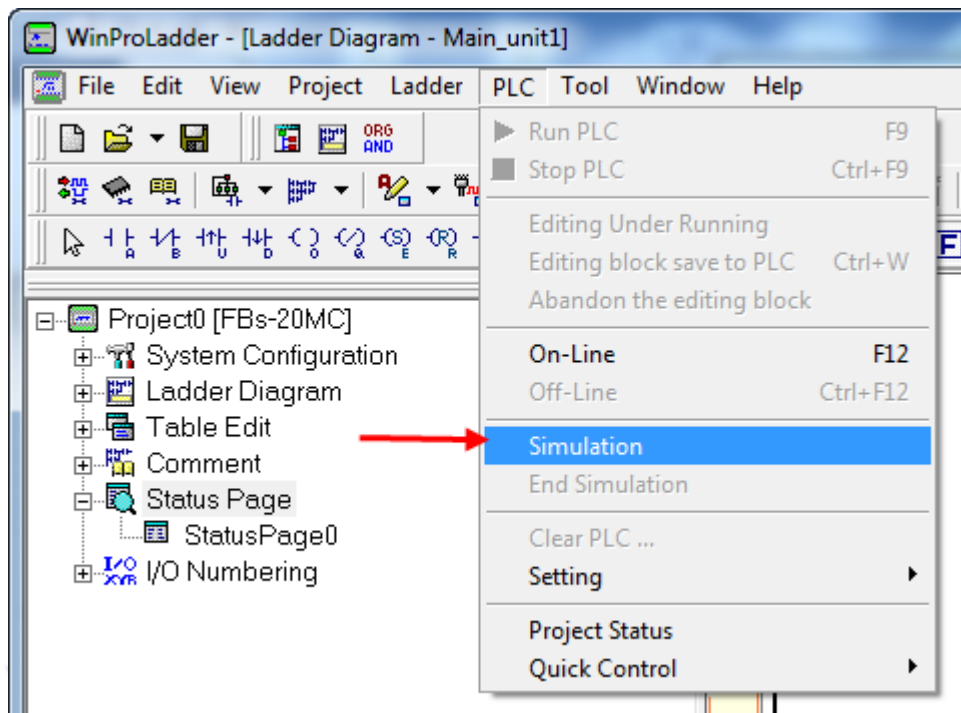


www.eza-co.com

شبیه سازی برنامه :

در مواردی که بدون PLC می خواهیم برنامه را تست کنیم می توانیم از مد **Simulation** استفاده کنیم .

با فعال کردن گزینه **Simulation** و سپس اجرای **Run** ، برنامه اجرا می شود .



www.eza-co.com

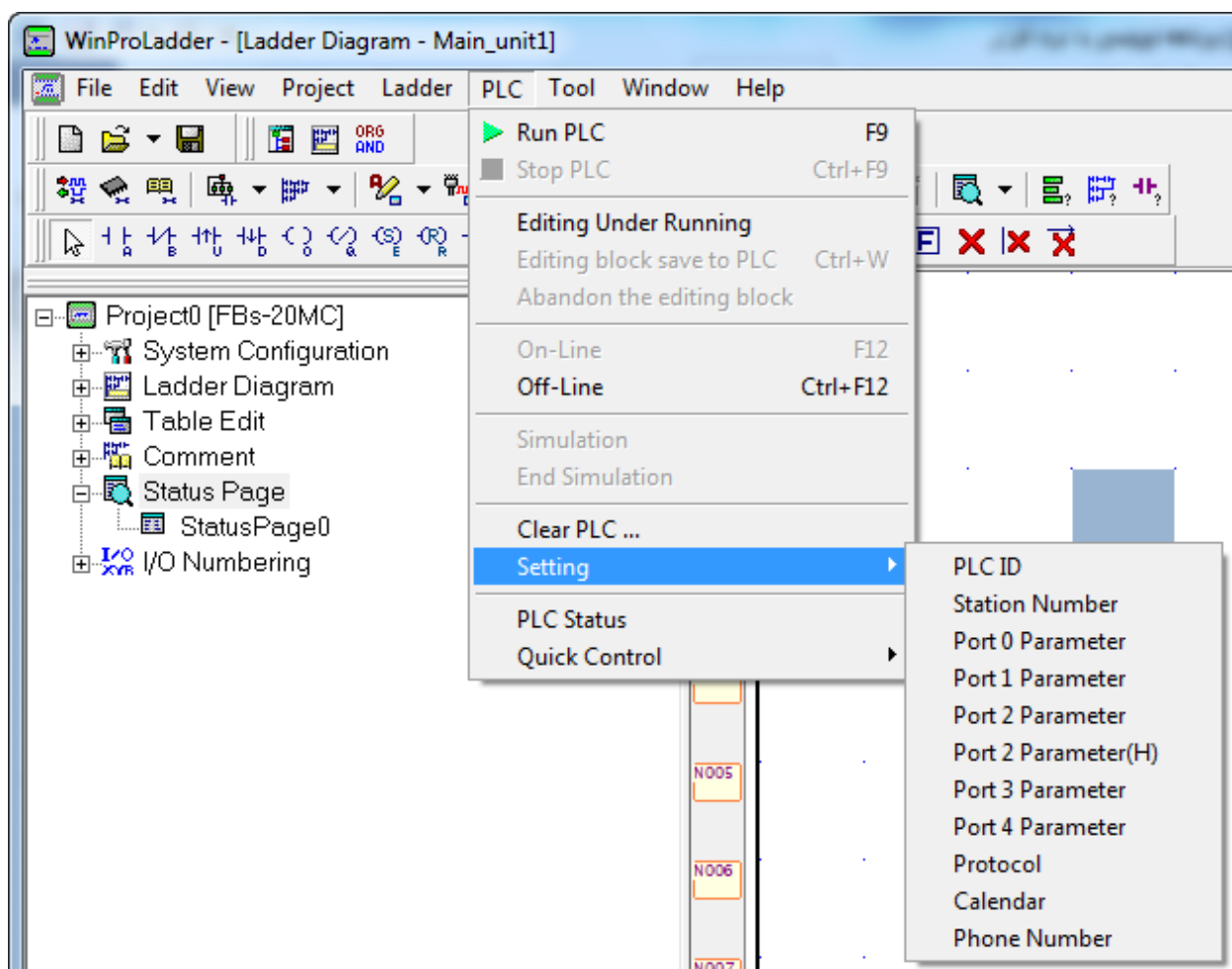
تنظیم پورتهای ارتباطی PLC :

با انتخاب منوی PLC و گزینه **Setting** در حالتی که برنامه به PLC آنلاین می باشد ، گزینه های تنظیمات پورتهای ارتباطی فعال می باشند، توسط این گزینه ها می توان **Baude rate , Parity , Data bits , Stop bit** و پروتکل ارتباطی که عبارتند از : **Fatek , Communication protocol , Modbus RTU (Slave) , Modbus Ascii (Slave)**

لازم به ذکر است که در پارامترهای پورت صفر فقط **Baude rate** قابل تغییر است و بقیه پارامترها برابر غیر قابل تغییر می باشند

پارامترهای پیش فرض پورتهای مطابق جدول زیر می باشند :

Baude Rate	۹۶۰۰	Reply delay time	۳
Parity	Even	Transmission delay	۰
Data bits	۷	Receive time out interval time	۵۰
Stop bit	۱	protocol	Fatek Communication protocol



Comm. Parameters Setting - Port1

Baud Rate: 9600  
Parity: Even parity  
Data Bit: 7 bits  
Stop Bit: 1 bit

☐ This port is used for current programming.

Reply delay time: 3 mS  
Transmission Delay: 0 x10mS  
Receive Time-out interval time: 50 x10mS

☐ Without checking of station number  
Protocol: Fatek Communication Protocol

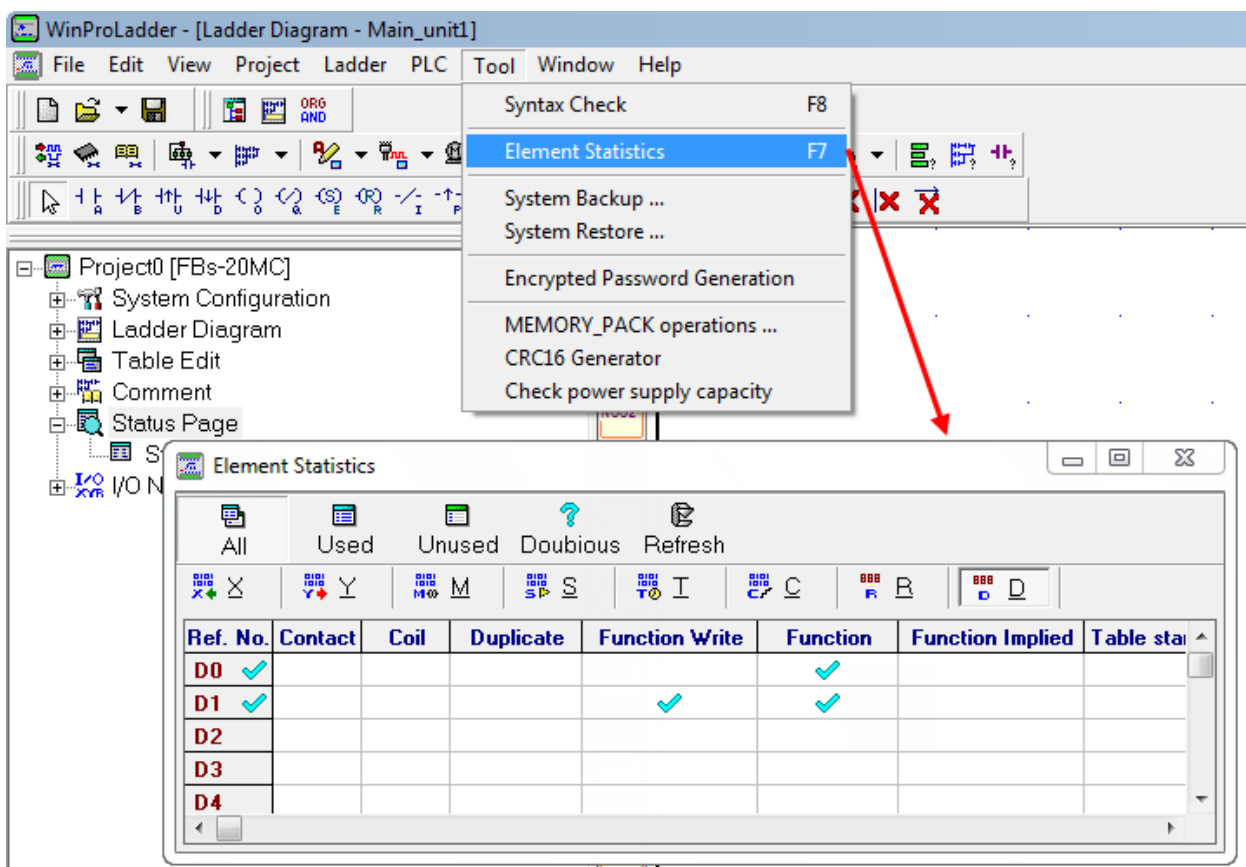
Port\_1 through Modem Interface Setting  
☒ Without above function  
☐ Remote CPU Link  
☐ Remote diagnosis

OK Cancel

www.eza-co.com

متغیرهای استفاده شده و استفاده نشده در برنامه :

با انتخاب از منوی **Tool** و گزینه **Element Statistics** می توان از شماره متغیرهای استفاده شده (**Used**) و یا استفاده نشده (**Unused**) آگاهی یافت .



## برنامه نویسی PLC

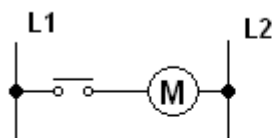
زبان ماشین مجموعه ای از کدهای باینری می باشد که تنها برای ریزپردازنده قابل درک است. از این رو برنامه نویسی با آن برای مهندسين دشوار می باشد. جهت سهولت در امر برنامه نویسی PLC- همانند کامپیوتر که ابتدا برنامه به زبان های سطح بالا نظیر C و Basic نوشته شده و سپس توسط کامپایلر به زبان ماشین تبدیل می شود- شرکت های سازنده ی PLC نیز هر کدام از زبان های سطح بالای خاص خود بهره می گیرند. در سال ۱۹۸۸ کمیته ی بین المللی الکتروتکنیکال (IEC) استاندارد IEC 1131-3 را به جهت شبیه ساختن زبان های برنامه نویسی در PLC منتشر ساخت. باوجود این هنوز به دلایل بسیاری، سازندگان PLC از زبان های مختص به خود استفاده می نمایند.

### ۴-۱) دیاگرام نردبانی

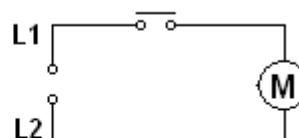
مدارهای فرمان عموماً به صورت دیاگرام نردبانی رسم می گردند. شکل ۴-۱ الف یک مدار الکتریکی و شکل ۴-۱ ب دیاگرام نردبانی معادل آن را در مدارهای فرمان نشان می دهد.

برای جایگزین ساختن یک سیستم کنترل مبتنی بر رله با یک PLC نیاز به تبدیل مدارهای فرمان با زبان برنامه نویسی PLC می باشد. استفاده از زبان LD (دیاگرام نردبانی) این تبدیل را بسیار ساده می نماید.

دیاگرام نردبانی از دو خط موازی تشکیل شده است که نشان دهنده خطوط تغذیه مدار می باشند و خطوط افقی که مانند پله های نردبانی می باشند خطوط برنامه هستند.



(ب)

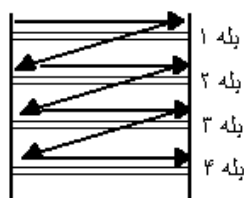


(الف)

شکل ۱-۴. ترسیم یک مدار الکتریکی به صورت : الف ( شماتیک ، ب ) دیاگرام نردبانی

هنگام نوشتن برنامه به زبان LD (دیاگرام نردبانی) موارد ذیل را به خاطر بسپارید:

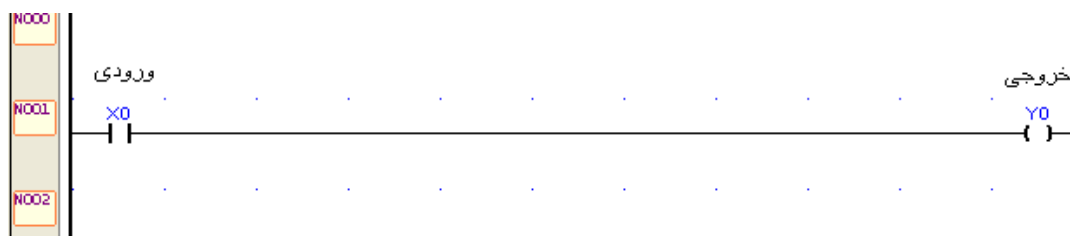
- ۱- هر خط از برنامه (هر پله نردبان) وظیفه ی خاصی را به عهده دارد.
- ۲- در PLC برنامه از سمت چپ به راست و از بالا به پایین اجرا می گردد و بعد از اجرای کامل برنامه ، اجرای آن دوباره از سرگرفته می شود توجه فرمایید که اگرچه شکل ظاهری دیاگرام نردبانی در مدارهای فرمان و برنامه های PLC یکسان است اما نحوه ی پردازش آن ها متفاوت می باشد.



شکل ۲-۴. چگونگی اجرای یک برنامه در PLC

- ۳- هر خط برنامه با تعدادی کنتاکت باز و یا بسته آغاز و با یک یا چند بوبین رله به انتها می رسد
  - ۴- کنتاکت ها در وضعیت عادی خود در برنامه نشان داده می شوند به عبارت دیگر کنتاکت های کمکی ، با فرض غیرفعال بودن رله ها نمایش داده می شوند
  - ۵- از کنتاکت های یک رله می توان در خطوط مختلف برنامه استفاده نمود.
  - ۶- هر کدام از کنتاکت های ورودی و رله های خروجی دارای آدرس منحصر به فرد می باشند. به عنوان مثال 40MA - FBs دارای ۲۴ ورودی و ۱۶ خروجی می باشد که آدرس آن ها به ترتیب ذیل است:
- خروجی ها: Y0 ~ Y15      ورودی ها: X0 ~ X23

به عنوان مثال در شکل ۳-۴ با وصل شدن کنتاکت ورودی، رله ی خروجی فعال می گردد و با باز شدن این کنتاکت خروجی نیز غیرفعال می شود.



شکل ۳-۴. یک خط برنامه در دیاگرام نردبانی

## ۴-۲) نوار المان ها

آیتم های این نوار برای ایجاد و ویرایش برنامه به کار گرفته می شوند.

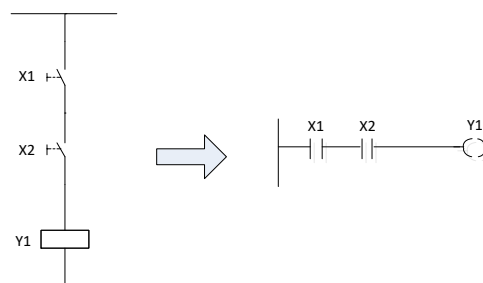
	"یک" بودن بیت مورد نظر را نشان می دهد
	"صفر" بودن بیت مورد نظر را نشان می دهد
	"لبه بالا رونده" بیت مورد نظر را نشان می دهد
	"لبه پایین رونده" بیت مورد نظر را نشان می دهد
	خروجی بیتی
	معکوس خروجی بیتی
	یک کردن بیت
	صفر کردن بیت
	معکوس کردن خط
	لبه بالا رونده خط
	لبه پایین رونده خط
	یک کردن رجیستر ( ۱بیتی ، ۱۶بیتی ، ۳۲بیتی )
	صفر کردن رجیستر ( ۱بیتی ، ۱۶بیتی ، ۳۲بیتی )
	تایمر
	شمارنده
	تمام دستورات نرم افزار



با ذکر چند مثال برنامه نویسی LADDER را توضیح می دهیم

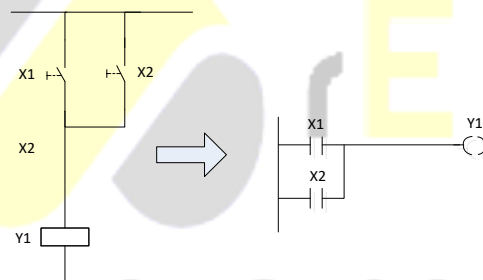
❖ مثال ۱ ) کاربری AND:

در اینجا خروجی Y0 تنها وقتی فعال می شود که هر دو ورودی X0 و X1 وصل شده باشند.



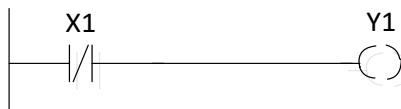
مثال ۲ ) کاربری OR

در این جا خروجی در صورتی فعال می شود که هر کدام از ورودی های A یا B یا هر دو وصل شوند.

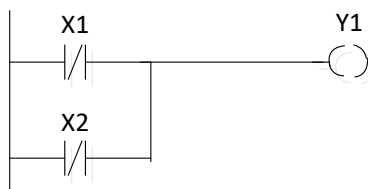


مثال ۳ ) کاربری NOT

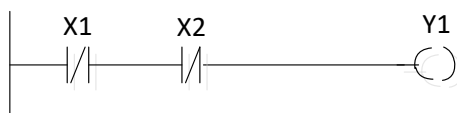
در این وضعیت بوبین خروجی Y0 و ورودی X0 عکس یکدیگر می باشند.



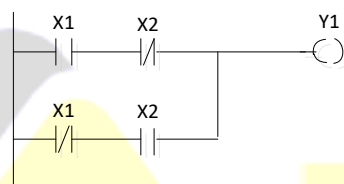
مثال ۴ ) کاربری NAND



مثال ۵ ( کاربرد NOR )



مثال ۶ ( کاربرد XOR )



مثال ۷ ( مدار خود نگهدار )

