

جلسه هشتم

هدف: بررسی

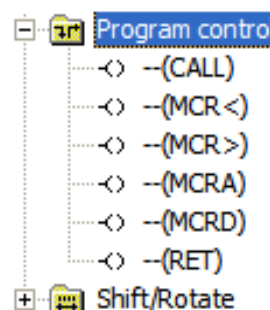
- استفاده از المان‌های کنترل برنامه^۱ شامل Call، Return و Master Control Relay
- استفاده از المان‌های مربوط به پرش در برنامه شامل Jump، Not Jump و Label
- نحوه ایجاد جدول متغیر^۲ و استفاده از آن برای اصلاح^۳ و پایش متغیرهای یک برنامه
- نحوه Force متغیرهای ورودی خروجی PLC
- بررسی نحوه استفاده از کلاک ژنراتور داخلی CPU

۸-۱) بررسی المان‌های کنترل برنامه

مقدمه

در جدول زیر مجموعه‌ای از المان‌های مربوط به کنترل برنامه نشان داده شده، که توضیح و کاربرد هر یک از این المان‌ها در ادامه این بخش آمده است.

برای فراخوانی FC و SFC های بدون ورودی خروجی تعریف شده
Master Control Relay On
Master Control Relay Off
Master Control Relay Activate
Master Control Relay Deactivate
برای برگشت اجرای برنامه از بلوک فراخوانی شده



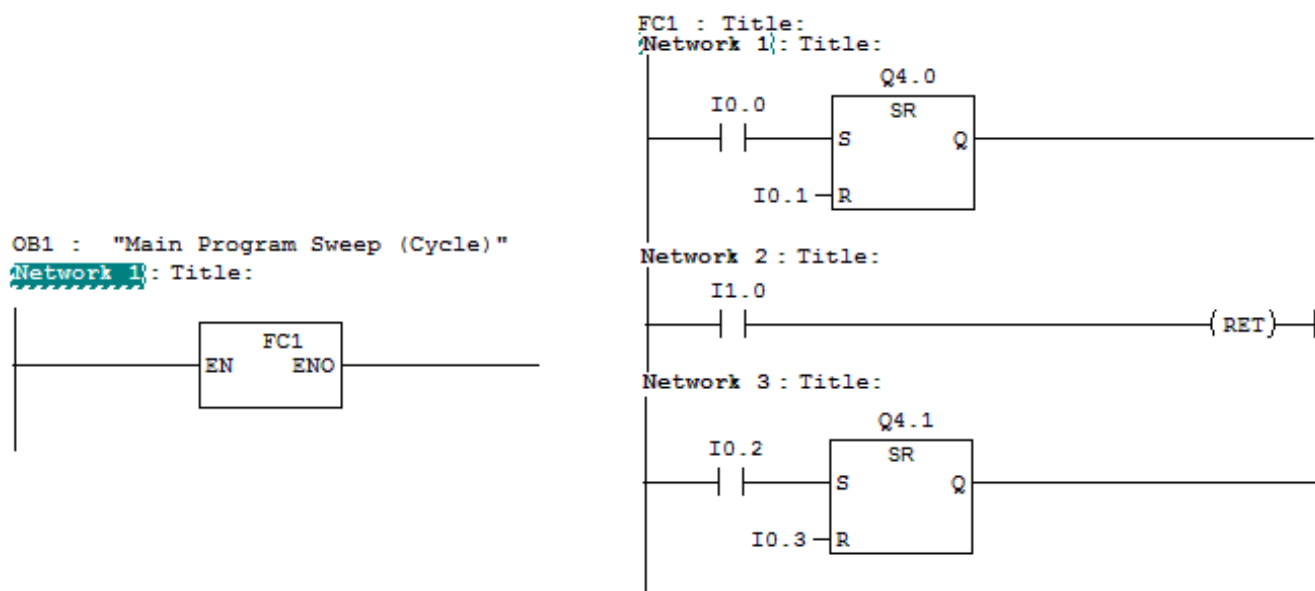
۸-۱-۱) **بررسی المان (RET) Return:** در صورت فعال شدن این المان در یکی از شبکه‌های یک بلوک، ادامه اجرای برنامه در آن بلوک پایان یافته و اجرای برنامه به محل بلوکی که بلوک جاری از آنجا فراخوانی شده برمی‌گردد. اگر این المان در بلوک OB1 فعال شود ادامه اجرای برنامه، به اولین شبکه این بلوک برمی‌گردد.

مثال برای کاربرد RET: برنامه شکل (۸-۱) دارای یک FC است. هرگاه در این FC کوئیل RET شبکه ۲ فعال شود، ادامه برنامه این بلوک (شبکه ۳) اجرا نشده و اجرای برنامه از آن بلوک خارج و به بلوک OB1 برمی‌گردد.

^۱ Program Control

^۲ Variable Table

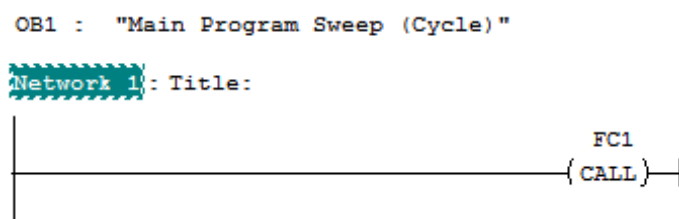
^۳ Modify



شکل (۸-۱)

۸-۱-۲) **بررسی المان CALL:** با این المان می‌توان بلوک های FC و SFC که دارای ورودی و خروجی تعریف شده نیستند را فراخوانی کرد. این فراخوانی می‌تواند بدون شرط یا با شرط انجام شود.

مثال برای کاربرد CALL: در طرح مثال قبل در بلوک OB1 می‌توان بلوک FC1 را که دارای ورودی و خروجی تعریف شده نیست بصورت شکل (۸-۲) فراخوانی کرد.



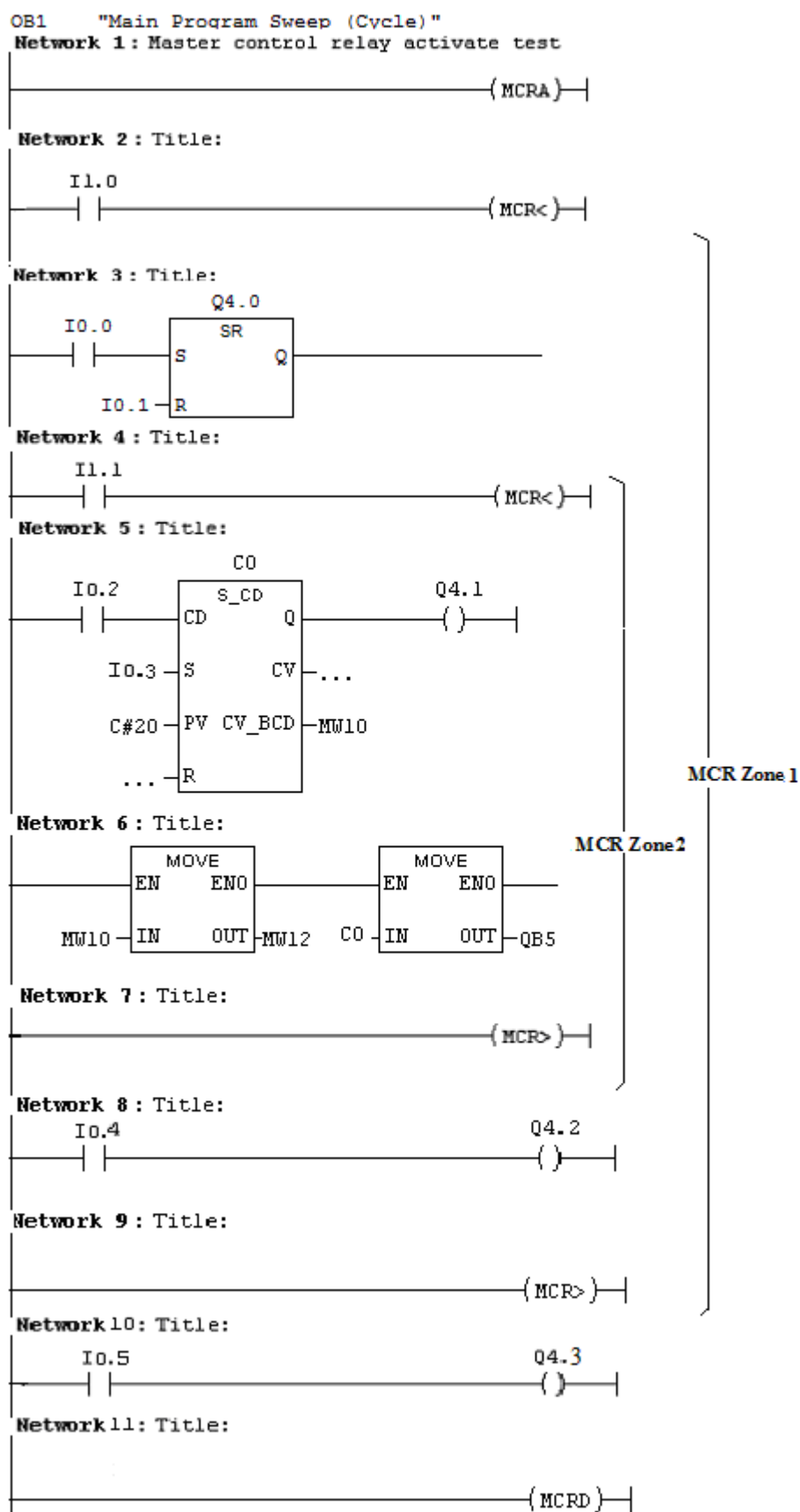
شکل (۸-۲)

۸-۱-۳) بررسی المانهای Master Control Relay

با جفت المان **MCRA** و **MCRD** می‌توان، انتقال نتایج اجرای بخشی از برنامه را با کمک جفت المانهای **MCR<** و **MCR>** ممکن و یا ناممکن ساخت. یعنی در صورت فعال بودن کوئیل **MCR<** انتقال ممکن و در صورت غیر فعال بودن آن انتقال غیر ممکن شده، و بجای نتایج اجرای برنامه به آدرس‌های مقصد صفر منتقل می‌شود. همچنین در این وضعیت تغییرات ورودی‌های R و S روی فلیپ فلاپ‌های مربوطه بی‌تاثیر خواهند شد.

توجه شود در برنامه المان **MCR<** بصورت شرطی اجرا می‌شود و المانهای **MCR>**، **MCRA** و **MCRD** برای اجرا، شرط نمی‌پذیرند.

مثال برای کاربرد المانهای MCR: در برنامه شکل (۸-۳) مثالی برای بکار گیری المانهای **MCR** آمده است. در این برنامه اگر ورودی I1.0 صفر شود در ناحیه یک انتقال داده‌ها صفر و تغییرات ورودی‌های فلیپ فلاپ بر روی آن بی‌تاثیر خواهند بود و همچنین اگر ورودی I1.0 یک و ورودی I1.1 صفر شود در ناحیه دو بجای انتقال داده‌ها صفر منتقل می‌شوند. اگر هر دو ورودی I1.0 و I1.1 یک شوند نتایج برنامه بصورت معمول اجرا خواهد شد.



شكل (٨-٣)

۲-۸) پرش در برنامه (JUMP):

مقدمه

المان‌های نشان داده شد در سمت راست شکل (۴-۸) برای ایجاد پرش در برنامه است که شرح عملکرد هریک از آنها در سمت چپ شکل آمده است..

(JMP): از این المان می‌توان به صورت شرطی و یا بدون شرط استفاده کرد.

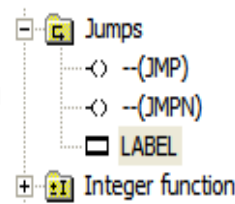
اگر بصورت غیر شرطی استفاده شود، برنامه، همواره تا محل LABEL اشاره شده به آن، اجرا نخواهد شد.

اگر بصورت شرطی استفاده شود در صورت برقرار شدن شرط (یعنی $RLO = 1$ شود) تا محل LABEL اشاره شده به آن، اجرا نخواهد شد.

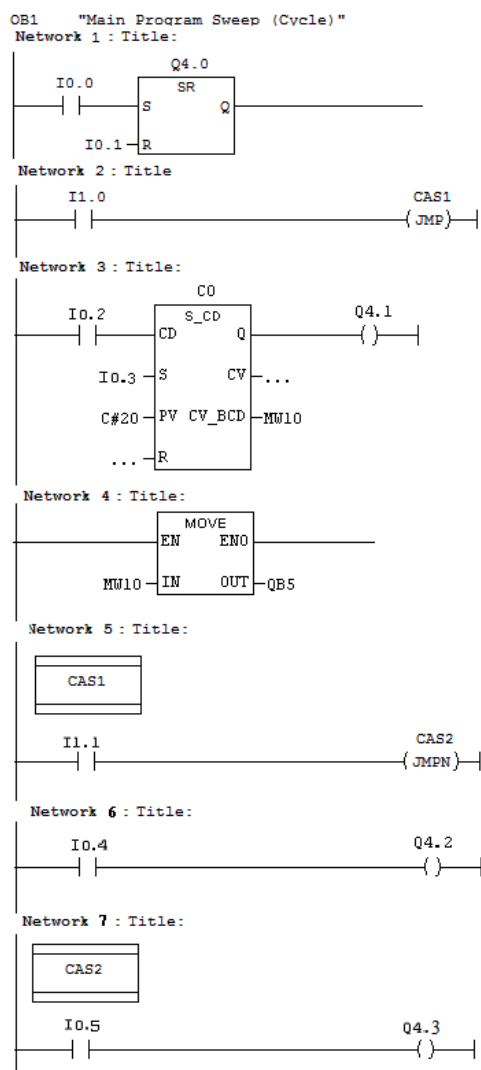
(JMPN): از این المان باید همیشه به صورت شرطی استفاده کرد و در صورت برقرار نشدن شرط

(یعنی $RLO = 0$ شود) تا محل LABEL اشاره شده به آن، اجرا نخواهد شد.

شکل (۴-۸) LABEL: محل پرش را نشان می‌دهد.



مثال برای کار برد المان‌های Jumps: در برنامه شکل (۵-۸) اگر ورودی I1.0 یک شود شبکه ۳ و ۴ اجرا نخواهند شد و اگر ورودی I1.1 صفر باشد شبکه شش اجرا نخواهد شد. **توجه شود**، در این نرم‌افزار عمل پرش فقط از بالا به پائین قابل اجرا است.



شکل (۵-۸)

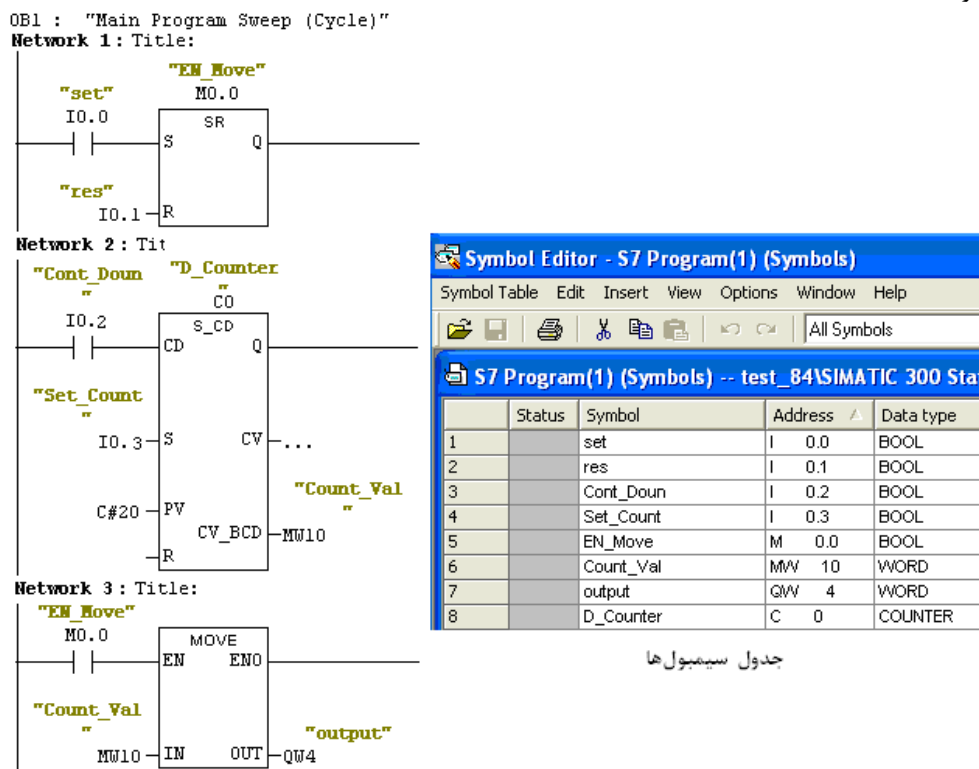
۸-۳) نحوه ایجاد بلوک جدول متغیر (Variable Table) و استفاده از آنها.

مقدمه

در برنامه‌های بزرگ، در بعضی از موارد، نیاز است، از مجموعه متغیرهای یک برنامه، تعدادی از آنها همزمان پایش شده و در صورت نیاز بعضی از آنها اصلاح شوند.

برای بررسی این موضوع ابتدا برنامه شکل (۸-۶) به همراه جدول سیمبول‌های آن آماده شده، سپس عملکرد آن بررسی

شود.



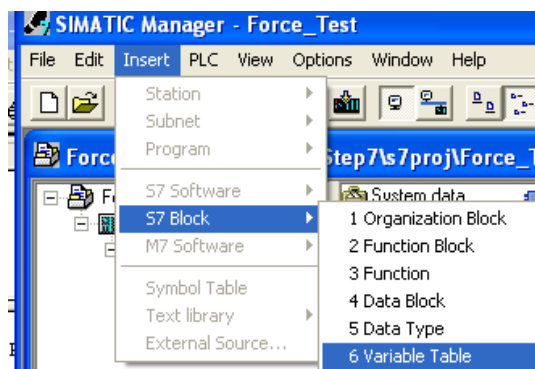
جدول سیمبول‌ها

شکل (۸-۶)

پس از آماده و بررسی کردن کار برنامه فوق یک بلوک جدول متغیرها برطبق مراحل آمده در زیر ایجاد و متغیرها مورد نیاز برای پایش و دستکاری را در آن وارد کنید آنگاه برای پایش و دستکاری متغیرها، جدول آماده کرده را بکار بگیرید.

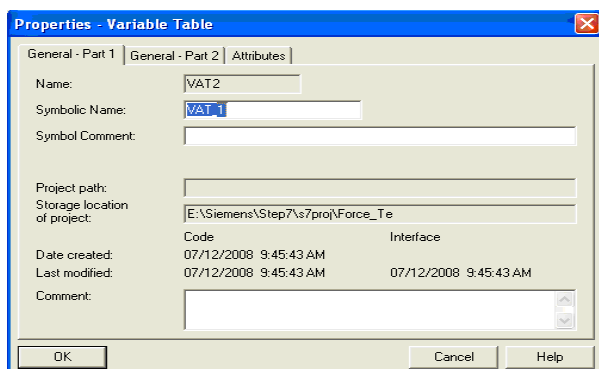
نحوه ایجاد بلوک جدول متغیرها: برای ایجاد بلوک جدول متغیرها به ترتیب زیر عمل می‌شود:

الف) در صفحه Simatic Manager، پس از انتخاب پوشه بلوک در پنجره جستجوگر، از مسیر نشان داده شده در شکل (۸-۷) گزینه Variable Table انتخاب می‌شود.

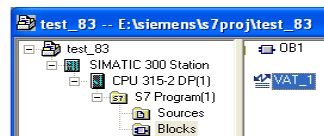


شکل (۸-۷)

ب) با این انتخاب، صفحه Properties Variable Table نشان داده شده در سمت چپ شکل (۸-۸) باز می‌شود،



شکل (۸-۸)



با تأیید کردن محتوای این صفحه توسط دکمه OK، یک بلوک جدول متغیر ایجاد می‌شود که آیکون آن در پنجره بلوک‌ها در صفحه Simatic Manager آشکار می‌شود. این آیکون در سمت راست شکل (۸-۸) در پنجره بلوک‌ها مشاهده می‌شود.

نحوه استفاده از بلوک جدول متغیرها: برای کار کردن با بلوک جدول متغیرها، ابتدا با دو بار راست کلیک کردن آیکون جدول متغیرها در پنجره بلوک‌ها آنرا باز و متغیرهای مورد نظر مانند مثال شکل (۸-۹) در آن وارد می‌شوند.

	Address	Symbol	Display format	Status value	Modify value
1	I 0.3	"Set_Count"	BOOL	false	true
2	M 0.0	"EN_Move"	BOOL	true	true
3	MW 10	"Count_Val"	HEX	W#16#0000	
4	QW 4	"output"	HEX	W#16#0000	W#16#0099

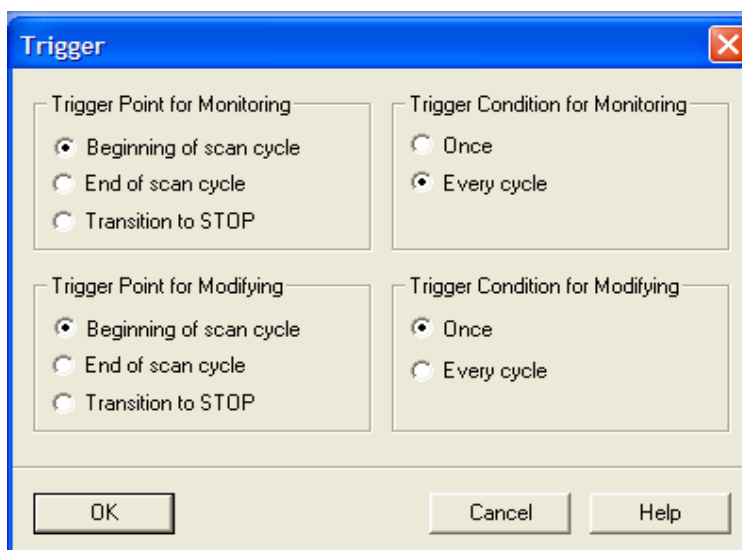
شکل (۸-۹)

در روی میله ابزار جدول متغیرها ابزارهای خاصی وجود دارند که در جدول زیر شرح عملکرد هر یک از این ابزارهای آمده است.

<p>برای غیرفعال کردن یک سطر استفاده می‌شود. (مشاهده تغییر وضعیت و اصلاح متغیر این سطر غیر فعال می‌شود).</p> <p>برای ارتباط جدول متغیرها با PLC در یک شبکه استفاده می‌شود.</p> <p>برای ارتباط جدول متغیرها با یک PLC مستقل استفاده می‌شود.</p> <p>برای تنظیم پارامترهای پایش^۱ و دستکاری (اصلاح)^۲ متغیرهای استفاده می‌شود.</p> <p>برای پایش متغیرهای جدول بصورت ممتد استفاده می‌شود. (وابسته است به تنظیم پارامتر نمایش)</p> <p>برای دستکاری (اصلاح) متغیرهای جدول بصورت ممتد استفاده می‌شود. (وابسته است به تنظیم پارامتر دستکاری)</p> <p>برای پایش مقادیر متغیرهای موجود در جدول برای یک سیکل (Scan Time) استفاده می‌شود.</p> <p>برای دستکاری (اصلاح) مقادیر متغیرهای موجود در جدول برای یک سیکل (Scan Time) استفاده می‌شود.</p> <p>برای فعال و غیر فعال کردن متغیرهای نیاز به دستکاری استفاده می‌شود. (تغییر وضعیت متغیر این سطر قابل مشاهده است).</p>	
---	--

¹ Monitoring
² Modifying

با انتخاب ابزار مربوط به تنظیم پارامترهای پایش و دستکاری متغیرها صفحه شکل (۸-۸) باز می‌شود. در این صفحه می‌توان عمل پایش و اصلاح در چه لحظه‌ای از اجرای برنامه صورت پذیرد تنظیم می‌شود. همچنین در این صفحه، از اینکه پایش و یا اصلاح در هر سیکل و یا (با هر بار انتخاب ابزار مربوطه) یک بار انجام شود تنظیم می‌شود.



شکل (۸-۱۰)

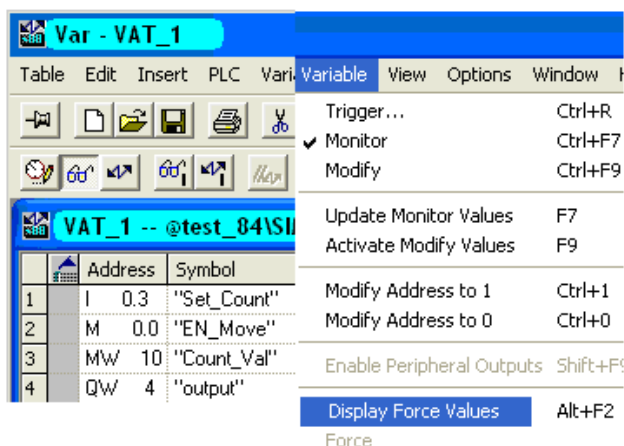
۸-۴ Force کردن متغیرها ورودی و خروجی

مقدمه

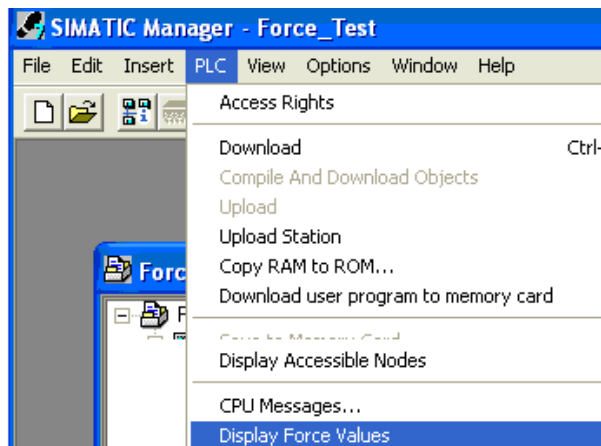
در بعضی از موارد برای اجرای یک برنامه PLC (بطور مثال در زمان تعمیرات) لازم می‌شود بعضی از متغیرها: مانند ورودی و خروجی برنامه بصورت اجباری به یک مقدار مشخص تثبیت شوند. برای این کار بصورت زیر عمل می‌شود.

نحوه باز کردن جدول Force : برای باز کردن جدول Force، در صفحه بلوک جدول متغیرها از منوی Variable (یا در صفحه Simatic Manager از منوی PLC) بطوری که در شکل (۸-۱۱) نشان داده شده است گزینه Display Force Value انتخاب می‌شود.

توجه کنید برای این انتخاب نباید جدول متغیرها به PLC ارتباط تبادل اطلاعات داشته باشد. (برای قطع این ارتباط، از منوی PLC گزینه Disconnect انتخاب، و برای اتصال مجدد آن از همین منو گزینه Connect to انتخاب می‌شود).



از بلوک جدول متغیرها



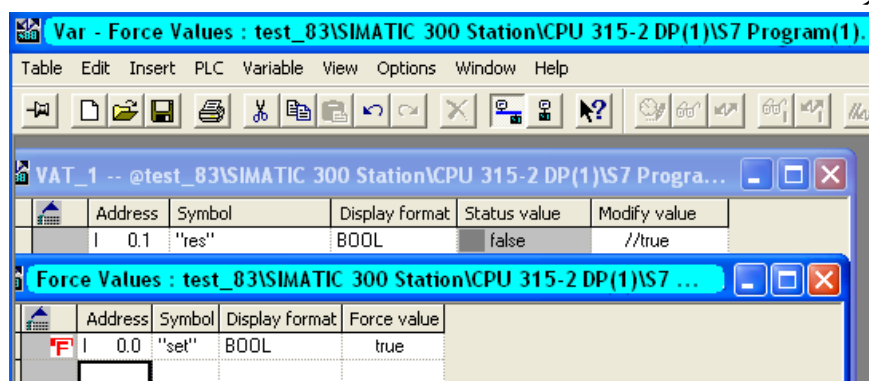
از صفحه Simatic Manager

شکل (۸-۱۱)

- با این انتخاب، صفحه Display Force Value که مانند جدول نشان داده شده در شکل (۸-۱۲) است باز می‌شود. در این جدول متغیرهایی که می‌بایستی اجباراً به یک مقدار مشخص تثبیت شوند، وارد می‌شوند این متغیرها برای PLC سری 300 فقط شامل ورودی، خروجی (بصورت بیت، بایت، Word و Double Word) هستند. **لازم است توجه شود**، در این نوع PLC ها مقادیر حافظه را نمی‌توان Force کرد.

نحوه Force کردن: پس از وارد کردن متغیرها در جدول و تعیین مقادیر مورد لزوم برای آنها، از منوی Variable این جدول گزینه Force انتخاب می‌شود. با این انتخاب متغیرهای وارد شده در جدول به مقادیر تعیین شده، در ورودی خروجی PLC تثبیت می‌شوند. در این شرایط چراغ Force روی ماژول PLC به رنگ زرد روشن می‌شود.

نحوه حذف Force: برای حذف متغیرهای تثبیت شده در PLC، در صفحه جدول متغیرها از منوی Variable گزینه Stop Forcing انتخاب می‌شود.



شکل (۸-۱۲)

۸-۵) نحوه استفاده از پالس ژنراتورهای داخلی CPU:

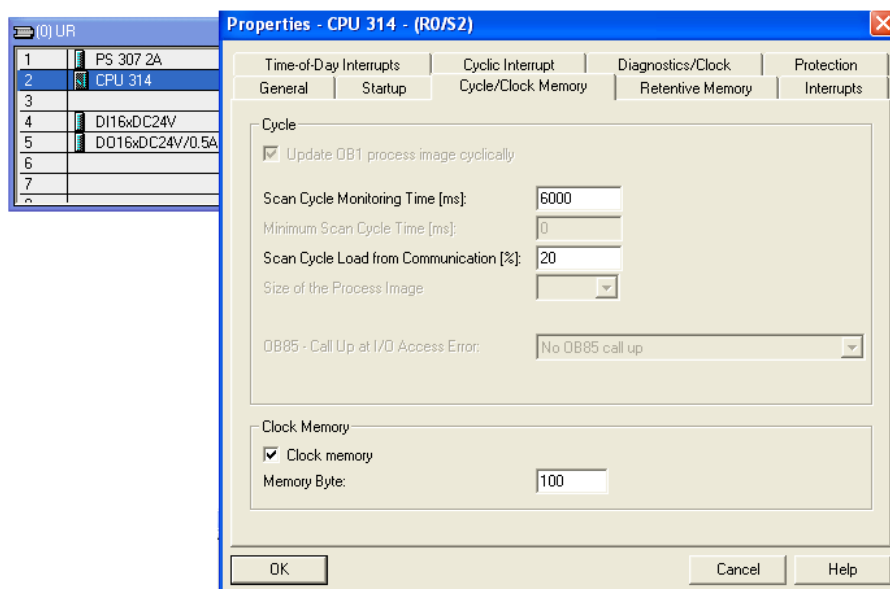
مقدمه

در ماژول CPU هشت بیت پالس ژنراتور وجود دارد که زمان پریود و فرکانس هر بیت آن در جدول زیر نشان داده شده است. برای استفاده از این پالس ژنراتورها لازم است برای هشت بیت آن یک بایت آدرس اختصاص داده شود. با اختصاص یافتن این یک بایت آدرس می‌توان در برنامه از هر بیت آن بعنوان یک پالس ژنراتور با فرکانس آمده در جدول زیر استفاده کرد.

بیت	0	1	2	3	4	5	6	7
زمان پریود	0.1	0.2	0.4	0.5	0.8	1	1.6	2
فرکانس	10	5	2.5	2	1.25	1	0.625	0.5

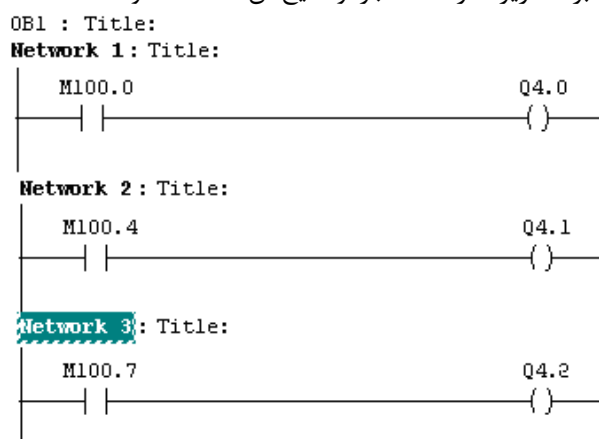
نحوه تنظیم آدرس اختصاصی برای پالس ژنراتورها

برای تنظیم یک بایت آدرس اختصاصی برای هشت پالس ژنراتور داخلی CPU، ابتدا در برنامه پیکربندی سخت افزار، با راست کلیک کردن بر روی ردیف CPU، از منویی که باز خواهد شد گزینه Object properties انتخاب می‌شود (یا اینکه ابتدا در برنامه پیکربندی سخت افزار ردیف مربوط به CPU انتخاب شده و سپس از منوی Edit گزینه Object properties انتخاب می‌شود)، با این انتخاب مجموعه صفحه‌های شکل (۸-۱۳) باز می‌شوند. در قسمت پائین صفحه با سر برگ Cycle/Clock Memory، گزینه Clock Memory انتخاب و سپس در جلوی عنوان Memory Byte شماره آدرس یک بایت اختصاصی برای پالس ژنراتورها (مثلاً ۱۰۰) وارد می‌شود. با تأیید این تنظیم‌ها توسط دکمه Ok و Down load کردن برنامه پیکربندی سخت افزار در CPU، می‌توان از این آدرس‌ها در برنامه‌ی PLC برای تولید Clock استفاده کرد.



شکل (۸-۱۳)

مثال برای استفاده از پالس ژنراتور داخلی CPU: برای بررسی سه بیت از یک بایت آدرس تنظیم شده در برنامه پیکر بندی سخت افزار برای پالس ژنراتور داخلی PLC، برنامه زیر در PLC اجرا و نتایج آن مشاهده شود.



توضیح: در این مثال می توان از المان Move برای مشاهده همه هشت بیت پالس ژنراتور بطور همزمان در خروجی QB5 استفاده کرد.