مدرسة دير البلح الثانوية الصناعية

قسم الإلكترونيات الصناعية

التحكم المنطقي المبرمج
Programming Logic Controller
(PLC)

الجانب التشغيلي للبرنامج S7

SIMATIC S7
Working with STEP 7

إعداد

محمد علي أبو حجر
قسم الإلكترونيات الصناعية

م. كمال نبهان أبو معيلق
مشرف التعليم المهني والتقني

2002

فهرســـــت

الجانب التشغيلي للبرنامج S7

1- الفصل الأول : التحكم المنطقي المبرمج

مقدمة ٢

التحكم المنطقي المبرمج ٢

نبذة عن النظام المستخدم لدينا ٤

طرق تمثيل دوائر التحكم ٥

المخطط السلمي..... ٥

مخطط البوابات المنطقية... ٨

قائمة الإجراءات .....

١١أنواع البلوكات ١٤

2- الفصل الثاني : وحدة الدخل والخرج ١٦

3- الفصل الثالث : وحدة الحاسب الآلي

وحدة الحاسب الآلي ٢٣

خطوات إنشاء البرنامج ٢٣

الرموز ٢٨

البرمجة ٣٠

البرمجة باستخدام المخطط السلمي ٣١

البرمجة باستخدام قائمة الإجراءات ٣٢

البرمجة باستخدام البوابات المنطقية ٣٤

إنشاء البرنامج باستخدام FB,DB ٣٦

إعدادات جهاز PLC ٣٧

ملاحظات في البرمجة ٣٩

تحويل البرنامج و اختباره ٤٠

استعادة البرنامج لقرائته أو تعديله ٤٢

4- الفصل الرابع : وحدة الربط بين جهاز PLC وجهاز الحاسوب ... ٤٤

5- الفصل الخامس : عناصر التمثيل و البرمجة

البوابات المنطقية و المفاتيح ٤٦

مسجلات العلامات ٤٦

دائرة الإلغاء و الإبقاء R/S ٤٧

المزمنات ٥٠
Pulse Timer SP ٥١
Extended Pulse Timer SE ٥٢
On Delay Timer SD ٥٣
Retentive On Delay Timer SS ٥٤
Off Delay Timer SF ٥٥
العدادات ٥٩
عداد تصاعدي / تنازلي ٥٩
عداد تصاعدي ٦٠
عداد تنازلي ٦١
المقارنات ٦٣
مقارنة عددين صحيحين ١٦ بت ٦٣
مقارنة عددين صحيحين ٣٢ بت ٦٤
مقارنة عددين حقيقيين ٦٤

الفصل الأول

التحكم المنطقي المبرمج

مقدمة :-

إن ظهور الحاسب الآلي (الكمبيوتر) كان أبرز عنصر في التقدم التكنولوجي الذي يشهده العالم الآن ، ولم يكتفي بظهور الحاسب فقط ولكن ثم الاتجاه إلى الاستفادة من الحاسب في جميع مجالات الحياة ، وقد نجح هذا الاتجاه بشكل ممتاز حيت أصبح يستفاد من الحاسب في جميع مجالات الحياة وأصبح يستخدم الحاسب في المجالات الطبية والمجالات الزراعية والمجالات التجارية ..... .

ونحن نستعرض كيفية الاستفادة من استخدام الحاسب في المجالات الصناعية و خاصة في عمليات التحكم .

و عملية التحكم هي من العمليات الأساسية و ذات الأهمية الكبرى في الصناعة ، و كان التحكم في بداية الأمر يتم عن طريق أشخاص ثم مع تطوير الإلكترونيات أصبح تستخدم الحواكم الإلكترونية (مثل الحاكم التناسبي و الحاكم التفاضلي و الحاكم التكاملي) ثم بعد ذلك ظهر عملية التحكم بالحاسب ، ونحن نعرض كيفية استخدام الحاسب في عملية التحكم المنطقي المبرمج .

التحكم المنطقي المبرمج Programming Logic Controller (PLC)

والتحكم باستخدام الحاسب يمكن تلخيصه في أن الحاسب يستقبل إشارات الدخل للنظام المطلوب التحكم فيه وهذه الإشارات تكون بالنظام الثنائي (و هو النظام الذي يتعامل به الحاسب) ثم يقوم الحاسب بتنفيذ البرنامج الموجود بذاكرته عن طريق و حدة CPU .

و يكون نتيجة تنفيذ البرنامج إشارات خرج تخرج من الحاسب إلى خرج النظام المطلوب التحكم فيه .

و نجد أن الأجزاء الأساسية لنظام التحكم هي :-

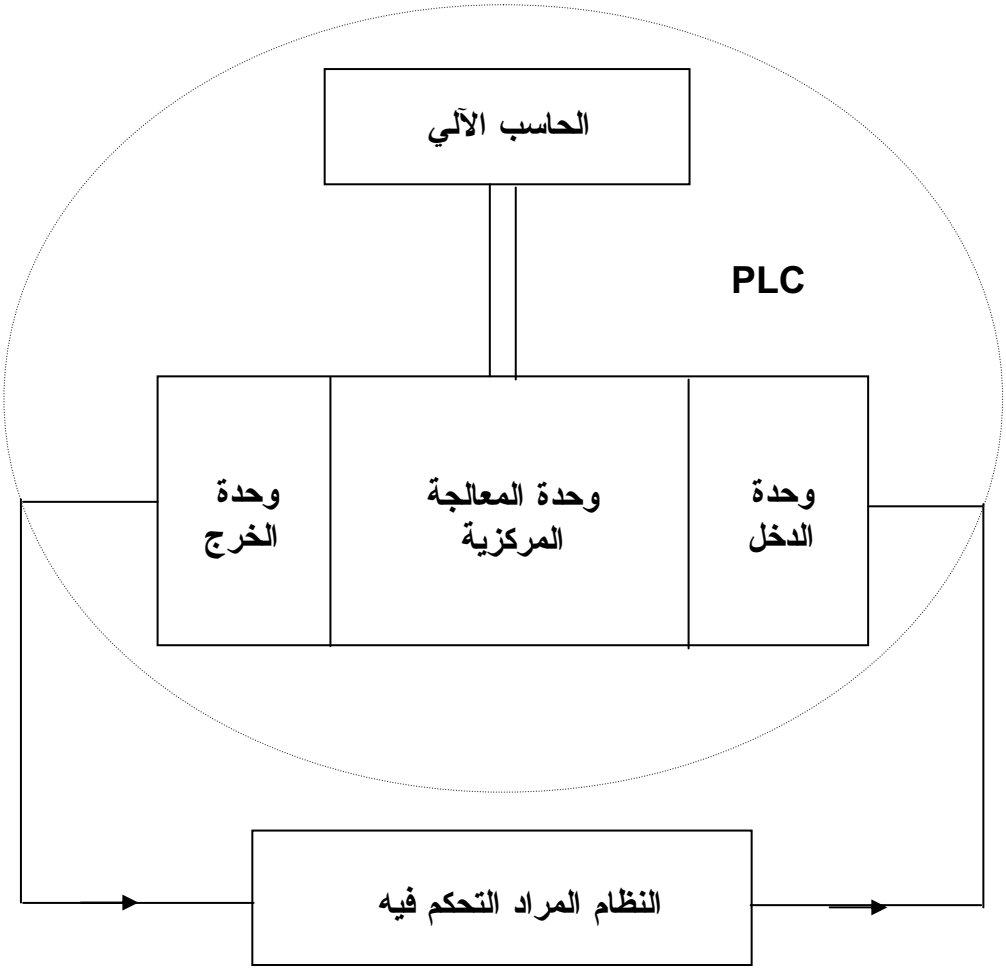
١- جزء الدوائر الإلكترونية و هو ما يعرف بـــ **Hardware** و أهم ما به هو وحده الـ CPU و هي عقل الجهاز و المسئولة عن تنفيذ البرنامج و يوجد كذلك و حدة الذاكرة التي يتم فيها تخزين برنامج التحكم المطلوب تنفيذه .

٢- وحدات الدخل و الخرج حيت إن وحدات الدخل هي التي تستقبل الإشارة من الوسط الخارجي المطلوب التحكم فيه، وتحويلها إلى إشارات ثنائية يمكن للحاسب إن يتعامل معها ، أما وحدات الخرج تقوم باستقبال إشارات الخرج من الحاسب بعد تنفيذ البرنامج و هي إشارات ثنائية و تحويلها إلى إشارات تغذي بها خرج النظام المطلوب التحكم فيه .

و نجد إننا نحتاج مع وحدات الدخل والخرج إلى المجسات Sensors وهي التي تعطى الإشارات إلى وحدة الدخل ، ونحتاج أيضا إلي المفاعلات التي تستقبل الإشارات من وحدات الخرج لتشغيل خرج نظام التحكم .

٣- برنامج التحكم وهو ما يعرف بالـــ (Software) و هي مجموعة من الأوامر المطلوب تنفيذها بالتتابع المكتوبة به لتنفيذ عملية التحكم و يخزن البرنامج علي وحدات التخزين مثل الأقراص المرنة و الصلبة .

وعند الرغبة في التعديل في برنامج التحكم يتم التعديل في البرنامج دون الحاجة إلى تغيير نظام التحكم .



نظام التحكم المنطقي المبرمج

نبذة عن النظام المستخدم لدينا :-

إن نظام التحكم المنطقي المبرمج الذي سوف نتعامل معه هو من شركة (**Siemens**) الألمانية وهذا النظام موجود بالفعل في الورش الصناعية الكبرى ، وهذا يعني أننا نتعامل مـــع نظـــام صناعي متواجد بالفعل .

وإمكانيات هذا النظام كثيرة ومتعددة و لكننا إن شاء الله سوف نســـتعرض أهـــم الخطـــوات اللازمة لاستخدام النظام في أداء وظائف التحكم المطلوب .

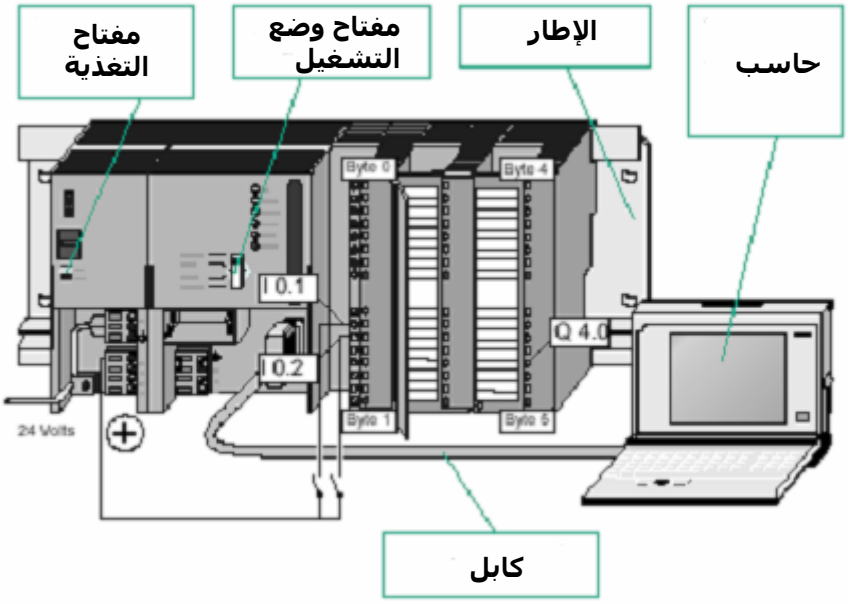
و نظام التحكم لدينا يتكون من ثلاث وحدات أساسية هي :-

١-و حدة الحاسب الآلي (جهاز الكمبيوتر) .

٢- وحدة الدخل و الخرج **I/O**.

٣-و حدة الربط بين الحاسب الآلي و جهاز التحكم (**Interface**) .

و الشكل الآتي يوضح الوحدات الثلاث :-



طرق تمثيل دوائر التحكم

نستعرض الآن كيف يتم تمثيل دوائر التحكم و الجدير بالذكر بان هناك ثلاث طرق رئيسة تستخدم لاعداد وتمثيل دوائر التحكم وهذه الطرق هي :-

١- المخطط السلمي Ladder Diagram Method (LAD)

٢- مخطط البوابات المنطقية Function Block Diagram (FBD)

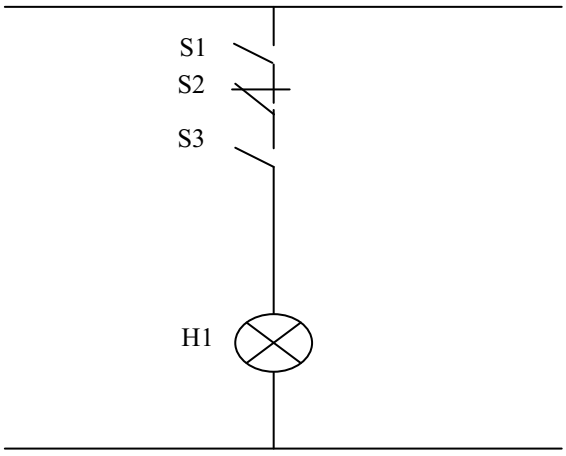
٣- قائمة الإجراءات Statement List Method(STL)

أولا :-

المخطط السلمي (LAD)

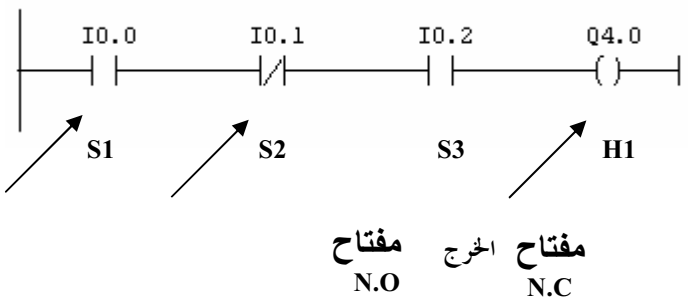
و هذه الطريقة هي أقرب ما يكون لمخطط مسار التيار (Power Rail) الذي يستخدم في الدوائر الكهربية ولكنها تكون في وضع أفقي في حين أن دائرة مسار التيار في وضع رأسي وهذه الطريقة هي أكثر الطرق المستخدمة في برمجة دوائر التمديدات الكهربية و كذلك دوائر التحكم في الآلات الكهربية بأنواعها .

وعلي سبيل المثال لو لدينا دائرة مسار التيار الآتية



دائرة مسار التيار

وعند تمثيل هذه الدائرة علي جهاز التحكم المنطقي المبرمج بطريقة (LAD) فتكون علي الصورة :-



ونجد أن دائرة مسار التيار هي نفسها دائرة الـ LAD و لكن دائرة الـ LAD في وضع أفقي .

ويرمز للنقاط المفتوحة وهو ما يعرف بـ N.O (Normally Open) ويرمز ⊣ ⊢

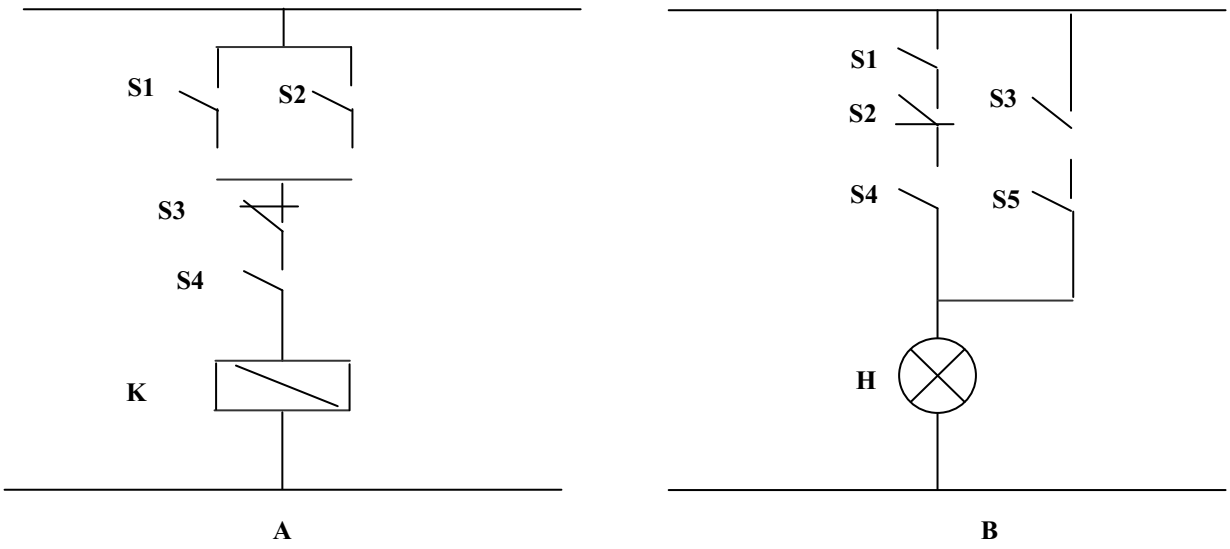
للنقاط المغلقة بالرمز وهو يعرف بـ N.C (Normally Close) . ⊣/⊢

أما الخرج فيرمز له بالرمز وهذا الخرج ثابت لأي خرج سواء كان مصباح أو محرك ⊣( )⊢

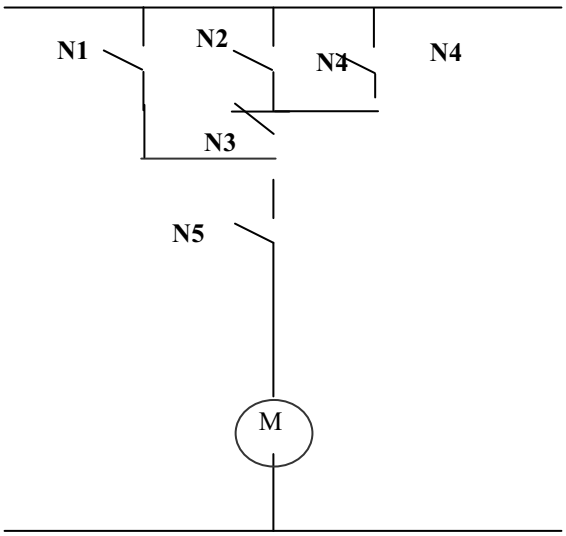
أو .....

ويراعى أنه يستخدم رموز الدخل I و رموز الخرج Q بدلا من S,H وسوف نشرح بالتفصيل استخدام هذه الرموز في وحدة الدخل والخرج ، وهنا في هذه الطريقة يتم رسم الدائرة علي الحاسب كما هي موضحة .

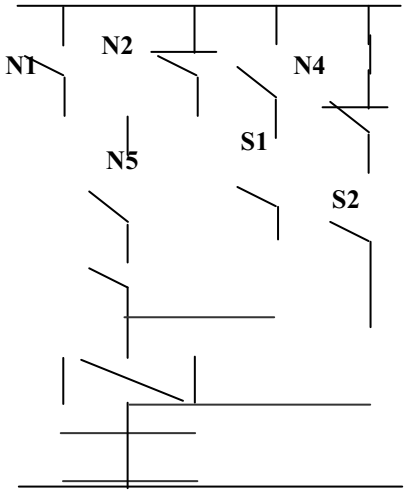
مثال: مثل دوائر مسار التيار الآتية باستخدام المخطط السلمي LAD :-



A B



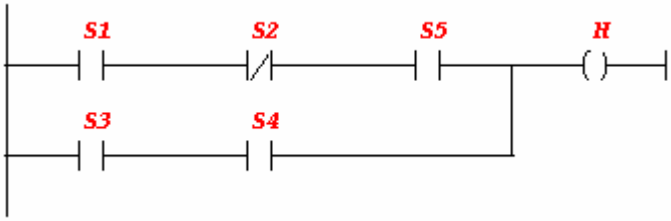
C



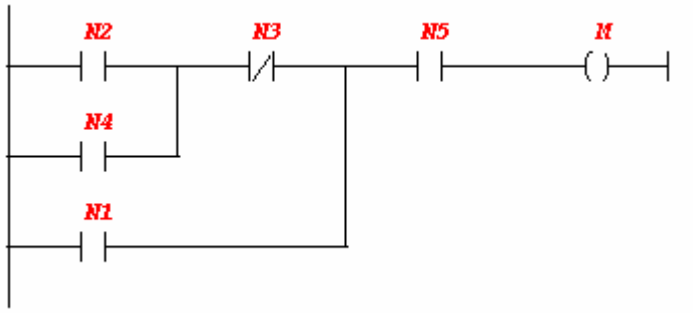
D

الحل :-

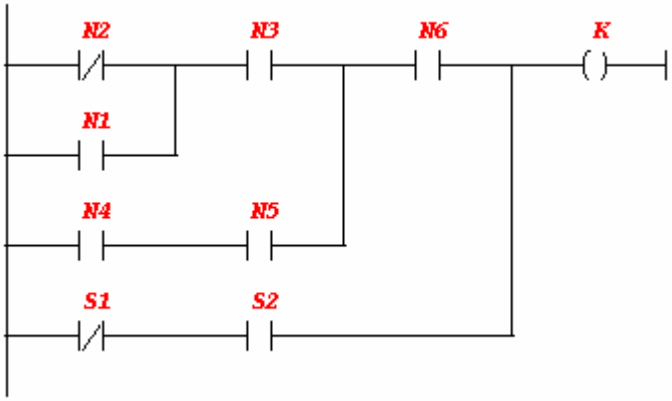
A



B



C

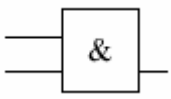


D

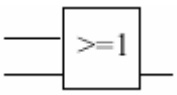
ثانيا :-

مخطط البوابات المنطقية(FBD)

وهذه الطريقة تستخدم فيها البوابات المنطقية وهي بوابة AND و بوابة OR و بوابة NOT وباقي البوابات المنطقية الاخرى ويرمز لها بالرموز الآتية



AND



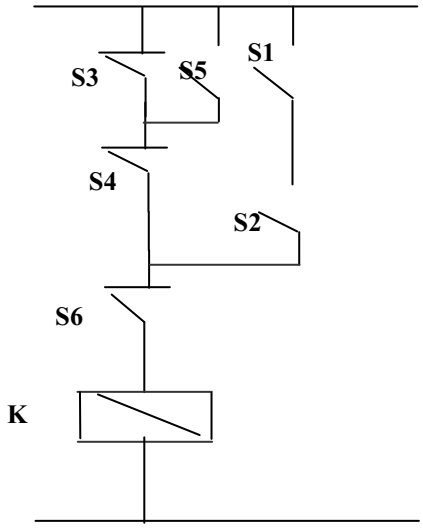
OR

NOT

ونجد أن التوصيل علي التوالي في دائرة مسار التيار يعادل بوابة AND ،التوصـــيل علـــي والتوازي يعادل بوابة OR و المفتاح المغلق N.C يعبر عنه بـــ NOT

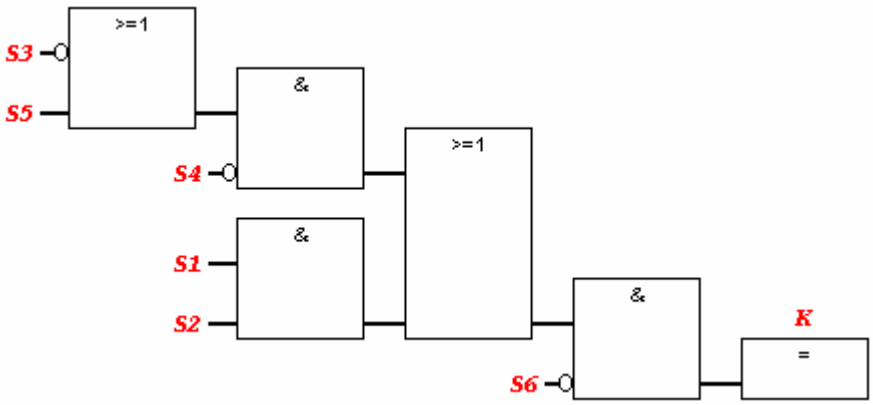
مثال

مثل الدائرة الآتية بالـــ FBD



دائرة مسار التيار

الحل :



دائرة الـــ FBD

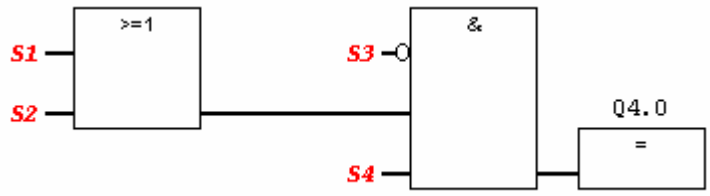
في هذه الدائرة نجد أن المفتاحين S3,S5 توازي لذلك وصلا ببوابة OR ثم يكونان توالي مع المفتاح S4 لذلك وصل خرج بوابة OR مع S4 علي بوابة AND .

و المفتاحين S1 , S2 توالي فوصلا علي بوابة AND و هي موصلان في فرع توازي مع مجموعة المفاتيح S3,S4,S5 ولذلك وصل خرج بوابة مع خرج بوابة علي بوابة OR و كل هذه المفاتيح توالي مع S6 لذلك وصل خرج مع S6 علي بوابة.

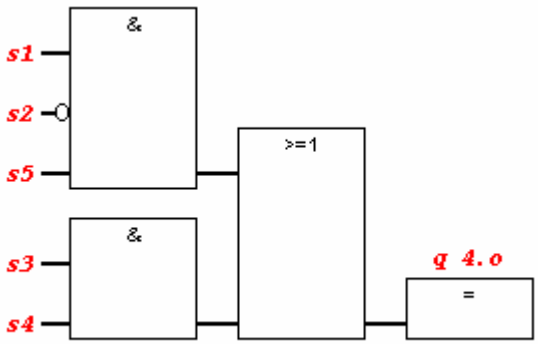
و نجد أن S4,S6, S3 هم مفاتيح N.C لذلك وضع لهم الرمز NOT علي مداخل البوابة .

مثال

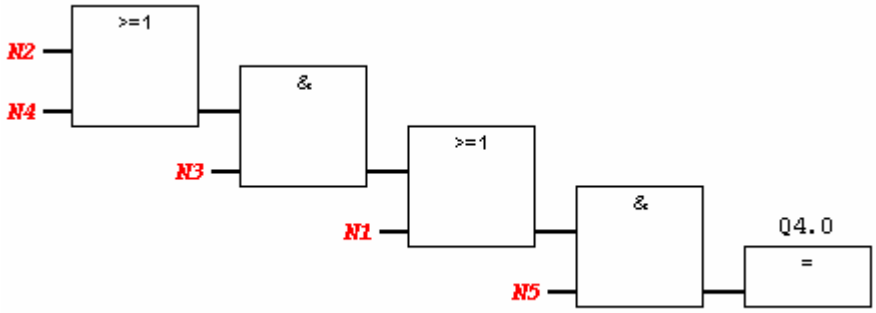
اعد حل المثال ص XX باستخدام FBD



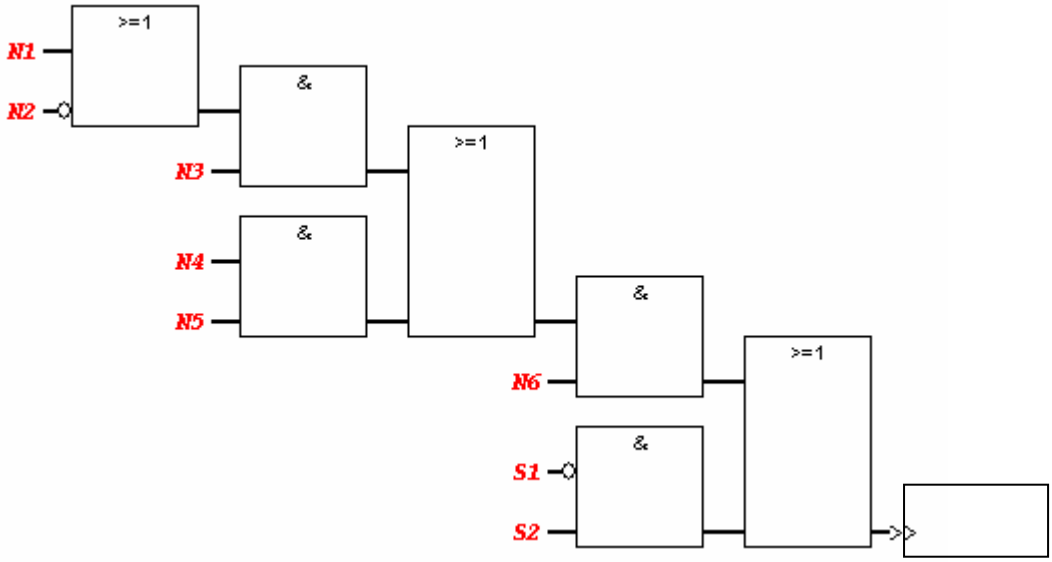
A



B



C



D

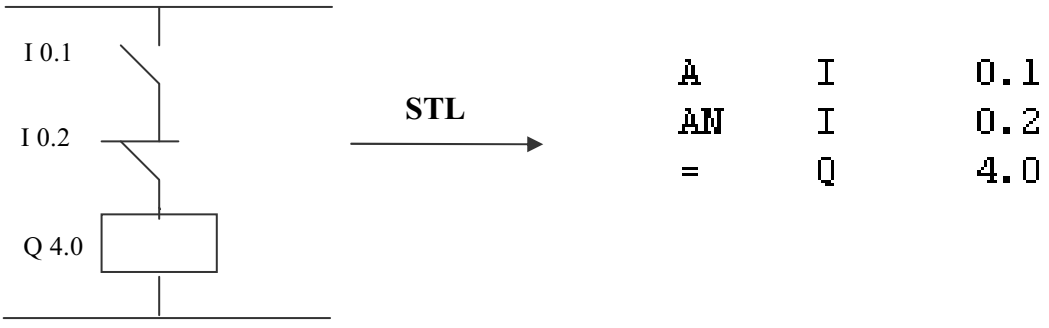
ثالثا :-

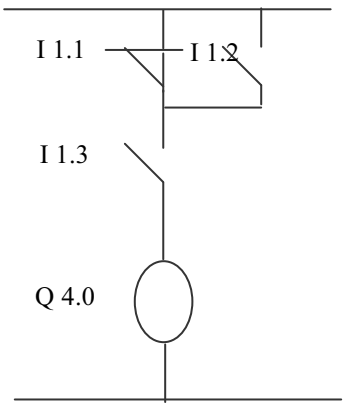
قائمة الإجراءات (STL)

هذه الطريقة نقوم فيها بوصف الدائرة في مجموعة أوامر، وهذه الطريقة قريبة من طريقة البرمجة بلغة التجميع وفيها نستخدم بعض الأوامر ويعبرعن هذه الأوامر ببعض الحروف عملية(التوالي ---> AND) ويرمز لها بالرمز A وعملية(التوازي ---> OR) ويرمز لها بالرمز O و المفتاح المغلق يضاف لاسمه N .

مثال

برمج مسار الدائرة الموجود أمامك بـــ STL





STL

```
A(
ON      I       1.1
O       I       1.2
)
A       I       1.3
=       Q       4.0
```

- في الدائرة الأولى نجد أن المفتاحين **I 0.1,I 0.2** علي التوالي **AND** لذلك ســبقهما **A**والمفتاح **IO.2** مغلق ولذلك وضع **N**بعد حرف **A**
- في الدائرة الثانية المفتاحين **I 1.1, I 1.2** توازي **OR** ولذلك سبقهما **O**و المفتاح **I 1.1** مغلق ولذلك وضع **N** بعد حرف **O** وهذا أن المفتاحان علي التوالي مع **I 1.3** ولذلك وضع الرمز **A(**ثم غلق القوس بعد ذلك ،(هذا يعني أن المفتاحين علي التوالي مع المفتاح الذي سوف وصف بعد ذلك و هو في هذه الحالة الدائرة المفتاح **I 1.3**

مثال

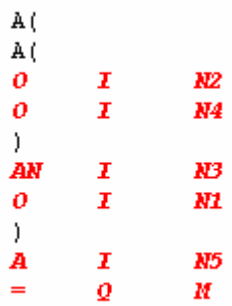
اعد حل المثال ص XX باستخدام STL

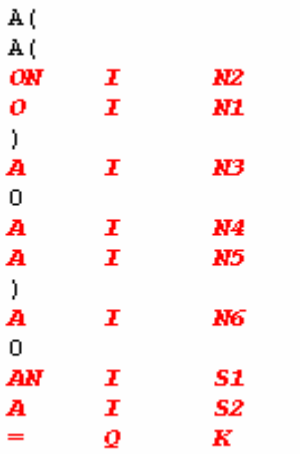
```
A(
O       I       S1
O       I       S2
)
AN      I       S3
A       I       S4
=       Q       K
```

A

```
A       I       S1
AN      I       S2
A       I       S4
O
A       I       S3
A       I       S5
=       Q       H
```

B





أنواع البلوكات

إننا نبدأ في مشاكل التحكم ومن ثم تحديد خطوات حل المشكلة ثم تحديد الدائرة المثلي ، للحصول علي الحل الأمثل بقدر الإمكان ، وهناك كثيرا من الوسائل الحديثة التي تساعد علي هذا وتسهل كثيرا ثم تمثيل الدائرة بإحدى الطرق الثلاثة السابقة الذكر .

و من هذه الوسائل هو استخدام الكمبيوتر و البرنامج المعد لذلك

- و الكمبيوتر المستخدم لدينا متوافق مع أجهزة IBM .
- و البرنامج المعد هو " STEP 7"

و سوف نبدأ بالتحدث عن بعض الأمور التي تهمنا و تسهل علينا أثناء استخدام البرنـــامج إن برنامج STEP 7 يسمح بترتيب المعلومات في BLOCKS بلوكات و هناك العديـــد مـــن أنواع الـــ BLOCKS تبعا لاستخدامها، و يمكن ربط هذه الأنواع من BLOCKS معا و يعرف BLOCKS عن طريق :

١- رقمـــــه

٢- نوعـــــه (FB, FC, OB ,.....) .

و نستعرض معا أنواع البلوكات التي يمكن استخدامها و هي :-

1- Organization Blocks (OB)

هذا البلوك هو الذي يقود البرنامج حيث عن طريقه يمكن الربط و النداء علي أنواع البلوكات الأخرى

و هو يقوم أيضا بعمل موائمة بين نظام التشغيل المستخدم و برنامج التحكم المعد (USER PROGRAM) و يندرج تحته أنواع متعددة كل نوع يكلف بتنفيذ مهمة محددة .

2- Functions (FCS)

هذا البلوك عبارة عن بلوك منطقي بدون ذاكرة مخرجة يحتوي علي الدالة المحسوبة بعد معالجتها ثم يأتي بعد ذلك مهمة المستخدم في كيفية استخدامها و حفظها .

3- FUNCTIONS BLOCKS (FBS)

هذا البلوك عبارة عن بلوك منطقي ذو ذاكرة بمعني انه يحتفظ بالبيانات و هو يحتوي علي مجموعة من الدوال المبرمجة التي نستخدمها بصورة مستمرة مثل PROGRAMMING CONTROLLER STRUCTURE

4- DATA BLOCKS (DB)

يحتوي علي المعلومات التي يحتاج إليها برنامج التحكم المصمم من قبل المستخدم .

5- SYSTEM BLOCKS

عبارة عن دوال مبرمجة موجودة في النظام نفسه يتم الاستفادة منها وتسمي دوال نظامية حيث ليس كل مرة نقوم ببرمجتها و لكن يتم استدعائها من النظام و يوجد منها نوعين

1-SYSTEM FUNCTION (SFCS)

تشبه FUNCTIONS

2-SYSTEM FUNCTION BLOCKS

تشبه FUNCTION BLOCK

6- SYSTEM DATA BLOCKS (SDB)

يحتوي علي معاملات وحدات الدخل و الخرج و العناوين .

وهذه أهم أنواع البلوكات المستخدمة و إن كان هناك بعض الأنواع ليس من الضروري ذكرها في هذه المرحلة و سوف يمكن التعرف و التعامل مع هذه البلوكات أثناء عملية البرمجة .

الفصل الثاني

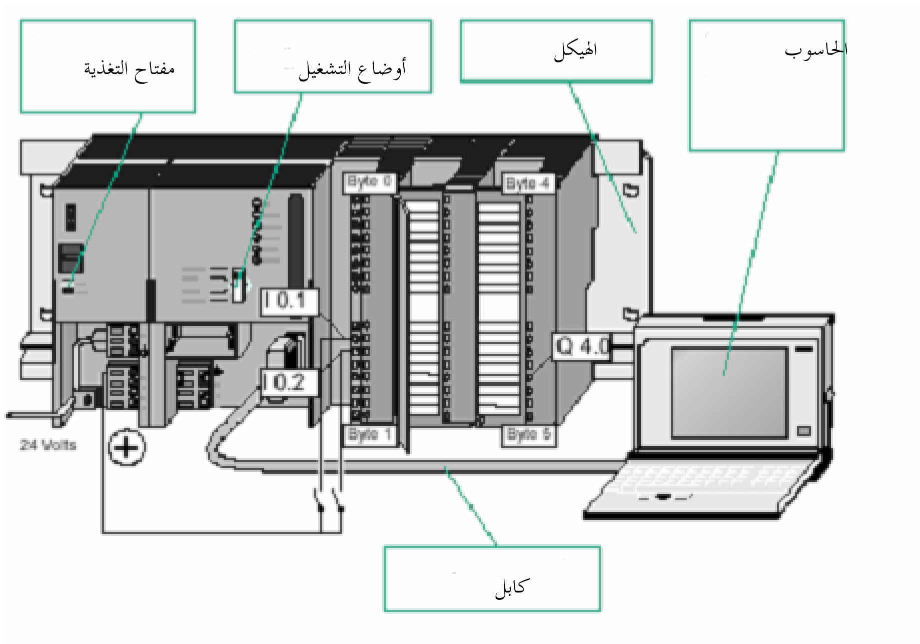
وحدة الدخل و الخرج

سنبدأ الآن في استعراض وحدات نظام التحكم المنطقي الموجود لدينا و تفاصيل استخدام كـــل وحدة علي حدة و كيفية التعامل مع كل وحدة و سنبدأ عرضهم بالترتيب الأتي

١- وحدة الدخل و الخرج

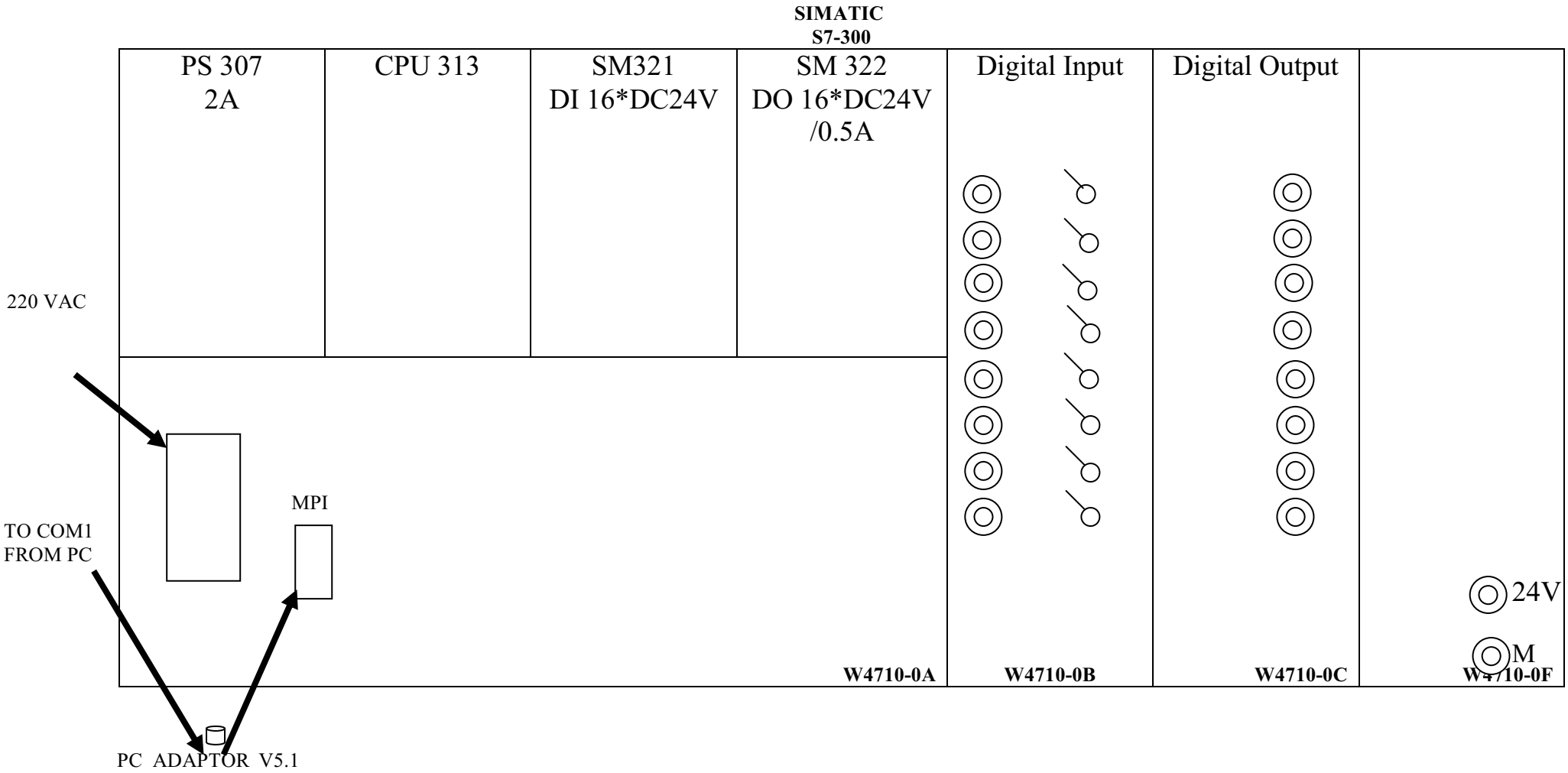
٢- وحدة الحاسب الآلي

٣- وحدة الربط بين الحاسب الآلي و جهاز التحكم PLC .



ندعوا الله أن يوفقنا وإياكم في الاستفادة من هذا النظام الاستفادة الكاملة

و اللـــــــــــــــــــــــــــه الموفـــــــــــــــــــــــــــــــــــق


SIMATIC
S7-300
PS 307
2A
CPU 313
SM321
DI 16*DC24V
SM 322
DO 16*DC24V
/0.5A
Digital Input
Digital Output
220 VAC
MPI
TO COM1
FROM PC
24V
M
W4710-0A
W4710-0B
W4710-0C
W4710-0F
PC ADAPTOR V5.1

وحدة الدخل و الخرج هي الوحدة الواسطة بين جهاز الحاسب الألي و الألة المراد التحكم فيها.

الصورة السابقة توضح شكل الوحدة الـــ plc و فيما يلي سنستعرض أهم أجزائها

١-وحدة CPU و الذاكرة :-

ان الجزء الموجود في اعلي الوحدة من اليسار يحتوي علـــي CPU (منفـــذ العمليـــات) أو الميكروبروسسور و هو المخ المسئول عن تنفيذ عمليات التحكم و يقوم بتنفيذها تبعا لبرنـــامج التحكم المكتوب و المخزن في الذاكرة وكما انه يوجد في هذا الجزء أيضـــا الـــذاكرة و هـــي تعرف بالـــ RAM و هذه الذاكرة يخزن فيها برنامج التحكم و تبعا لسعة الذاكرة يتم تحديد أقصى عدد من خطوات التحكم التي يمكن تخزينها في الذاكرة ويمكن معرفة مود يل CPU و كذلك سعة RAM من الكتالوج المرافق لجهاز التحكم .

٢-بطارية صغيرة :-

هذه البطارية تمد الذاكرة بالكهرباء اللازمة لها حيث أن الذاكرة التي مـــن النـــوع RAM و المستخدمة في النظام بمجرد ان تفصل عنها الكهرباء تفقد كل ما بداخلها من معلومات و لذلك يتم تغذيتها عن طريق هذه البطارية حتى إذا فصلنا الكهرباء عن جهاز PLC تستطيع الذاكرة الاحتفاظ بما في بداخلها من برامج

٣-لمضة مصباح وبيان :-

يقع اسفل الناحية اليسرى حيث يتم من خلاله توصيل جهاز PLC بالخط التغذية الرئيسي ويوجد بالأعلى منه لمضة بيان ومفتاح يسمح بتوصيل جهد مقـــداره ٢٤ V إلـــى دوائـــر الجهاز المختلفة .

٤-مفتاح وضع التشغيل Operating Mode S

يستخدم لتغيير وضع التشغيل وحيث يوجد أربعة أوضاع مقابل كـــل وضـــع يوجـــد مبـــين والأوضاع الأربعة هي كما يلي :-

- Memory Reset (MRST) يلغي كافة المعلومات الموجودة في الذاكرة .
- STOP عند هذا الوضع لا ينفذ برنامج التحكم من ناحية و من ناحية أخري و يكون علي وضع استعداد في استقبال برنامج جديد .
- RUN-P & RUN يستخدمان لتنفيذ البرنامج .

٥- وحدة الدخل رقم 0

و هي تحتوي علي ٨ فتحات يمكن عن طريقها توصيل ٨ نقاط دخل و تحتوى كذلك علي ٨ مفاتيح تستطيع استخدامها لإعطاء دخل برنامج التحكم إذا ما لم يكن هناك دخل موصل مـــن

الخارج و عناوين هذه الوحدة يتكون من ٨ خانات و يرمز للدخل بـــالرمز **I** و يمكـــن الآن معرفة عنوان كل من دخل من الدخول الثمانية ويكونوا عنوانيهم

I0.0	I0.1	I0.2	I0.3	I0.4	I0.5	I0.6	I0.7

وبصورة عامة يمكن القول بأن الحرف **I** يعني وحدة دخل ، والرقم الاول من اليسار هـــو رقم Byte و الرقم الاخر هو رقم Bit

٦- مبين وحدة دخل رقم 0

و هي تحتوي علي ٨ لمبات بيان حالة خاصة بوحدة الدخل رقم 0 و هي مرقمة مثل و حـــدة الدخل رقم 0 الموجودة بجانبها و تضيء اللمبة التي تناظر المفاتيح الموجودة علـــي و حـــدة الدخل إذا كان المفتاح أو الدخل في الوضع ON ولا تضيء إذا كان المفتاح في الوضع OFF

٧- وحدة دخل رقم 1

هي وحدة دخل أيضا ذات ٨ نقاط دخل و يكون عناوين هذه النقاط الثمانية هي

I1.0	I1`.1	I1.2	I1.3	I1.4	I1.5	I1.6	I1.7

٨- مبين وحدة دخل رقم 1

ويحتوي أيضا علي ٨ لمبات بيان تناظر ٨ نقاط الدخل الموجودة في الوحدة رقم 1

٩- وحدة الخرج رقم 4

و هي تحتوى علي ٨ نقاط خرج يمكن عن طريقهم تشغيل ٨ وحدات مختلفة وعنـــوان هـــذه الوحدة هو 4 و يرمز بالخرج الرمز **Q** وبذلك تكون لدينا ٨ نقاط خرج عنوانيهم كالأتي

Q4.0	Q4.1	Q4.2	Q4.3	Q4.4	Q4.5	Q4.6	Q4.7

١٠-مبينات لوحدة الخرج رقم ٤

و تحتوي علي ٨ لمبات تناظر ٨ نقاط الخرج الموجودة علي الوحدة رقم ٤ و تضئ اللمبة إذا كان نقطة الخرج يوجد عليها خرج

١١- وحدة الخرج رقم 5

و تحتوي أيضا علي ٨ نقاط خرج وعناوين هذه النقاط تكون

١٢- مبينات لوحدة الخرج رقم 5

و تحتوي علي ٨ لمبات بيان تناظر الـــــ ٨ نقاط الموجودة علي الوحدة رقم ٥

ونجد انه في جميع المبينات سواء كانت خاصة بوحدات الدخل أو خاصة بوحـــدات الخـــرج يوجـــد بجـــوار لمبـــات البيـــان شـــريحة بيضـــاء اللـــون ليـــتم عليهـــا كتابـــة المفـــاتيح و الخرج المناظر لكل نقطة ليسهل غلي المستخدم معرفتها

١٣- فتحة لتركيب كارد Flash Memory Card

EPROM هي نوع من أنواع الذاكرة و في هذا النظام يسمح لنا بكتابة البرنامج علي هذا النوع من الذاكرة و تركب هذه الذاكرة علي النظام المراد التحكم فيه و فيها يتم قيادة النظام

ملاحظة //

نجد أن وحدات الدخل و الخرج الموجودة علي النظام الذي لدينا هي رقمية DIGITAL INPUT, DIGITAL OUTPUT

ووحدات الدخل هي 24v*8 تعني إن كل نقطة من الثمانية نقط الموجودة تستطيع ان تتحمل ٢٤ فولت تيار مستمر

ووحدات الخرج هي 24v/0.5A*8لنقطة من نقاط الخرج تستطيع ان تغذي خرج يعمل بجهد ٢٤ فولت تيار مستمر ولا يزيد سحبه عن 0.5A

و قد تتساءل و ماذا عن المعدات و الآلات التي تعمل عند جهد اعلي من ٢٤ فولت وتسحب اكثر من ٠,٥ أمبير ونقول لك إن وحدات الدخل و الخرج يوجد منها وحدات تماثلية ANALOG MODULE و ايمكن شراؤها من الشركة الموردة و تركيبها بدلا من الوحدات الموجودة لدينا

وبشكل عام تستطيع ان نقول ان هناك نماذج عديدة منت وحدات الدخل و الخرج ذات قيم تيار و فولت مختلفة تحصل عليها حسب احتياجك وحسب المعلومات الكهربية الخاص بالنظام المراد التحكم فيه من حيث الفولت و الأمبير

و هناك حل أخر هو انك تستطيع ان تستخدم مرحل V٢٤ و من أطراف التوصيل لهذا المرحل يمكن أن تتعامل مع الآلة

كما انه يوجد دخل وخرج ٣ اوجه (3-PHASE) و هي فعلا التي تستخدم في المصانع الكبرى التي يكون معظم آلاتها ٣ اوجه .

و أخيرا نستطيع القول بان وحدة الدخل و الخرج هي الدائرة الوسيطة بين الكمبيوتر و الآلات و المحركات حيث إنها تتعامل مع الكمبيوتر و تعطيه و تستقبل منه الـــ ١،٠ الرقمي **و** نفس الوقت تغذي الآلة بالأمير و الفولت اللازم لتشغيلها و تستقبل من دخل الآلة جهدها الطبيعي و بالتالي تكون و حدة الدخل و الخرج نعم الوحدة الوسيطة التي سهلت كثير من المشاكل و أيضا يمكننا إضافة أعداد إضافية من وحدات الدخل و الخرج إلى جهاز PLC .

الفصل الثالث

وحدة الحاسب الآلي

وحدة الحاسب الآلي

هي جهاز حاسب آلي يمكن اختياره متوافق مع أجهزة (IBM) حيث يمكن استخدامه كحاسب آلى بعد الانتهاء من تنفيذ عملية التحكم .
و هناك بعض الشركات تنتج أجهزة الحاسب المستخدم في نظام PLC بنظام تشغيل خاص به يوجد علي هذا الحاسب برنامج التحكم المستخدم لأعداد برنامج التحكم المطلوب وهذا البرنامج (Software) يمكن أن يعمل مع الأجهزة المتوافقة مع IBM و لكن يحتاج معه إلى مشفر (Decoder) و هذا المشفر يطلب من الشركة المنتجة الـ Software حتى تستطيع الشركة حماية الـ (Software) الخاص بها و قد يبدو لك أن هذا النظام يحتاج إلى مبرمج متخصص ليقوم بإعداد برنامج التحكم المطلوب ولكن البرنامج المستخدم يتميز بسهولته التي تسمح بتصميم برنامج التحكم دون سابق معرفة بلغات البرمجة المتعارف عليها مثل , PASCAL} C++ ,.....{
ويعد هذا السبب من الأسباب التي شجعت كثيرا نظام الـ PLC .

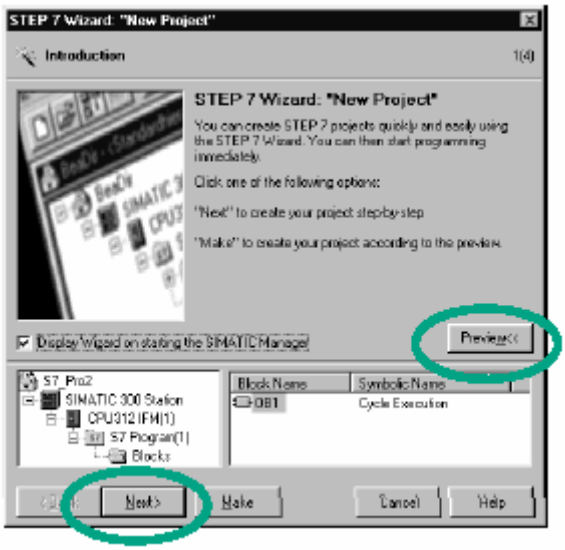
سنبدأ في هذا الفصل بالتعرض لبرنامج step 7 و طريقة استخدامه لتمثيل و تنفيذ برنامج التحكم
باستخدام برنامج step 7 تستطيع أن تنشأ برنامج التحكم المطلوب الذي عند نقله الي جهاز PLC يتحكم ويراقب الآلة من خلال البرنامج المعد .

وقبل أن نبدأ في شرح البرنامج نريد أن نوضح أن خطوات كتابة البرنامج وتصميمه هي
١- تحديد المشكلة
٢- طريقة حلها
٣- كتابة البرنامج
٤- مراجعة واختبار البرنامج
٥- تنفيذ البرنامج

و الآن سنبدأ معا خطوة بخطوة في شرح البرنامج
بداية يجب عليك أن تقوم بتنزيل Step 7 علي جهاز الحاسب من خلال تشغيل التطبيق الموجود في المسار <CD-ROM Drive>:\Step 7\Disk1\Setup.exe

بعد الانتهاء من عملية التنزيل و إعادة تشغيل الجهاز ، أيقونة رمز البرنامج سوف تظهر علي سطح المكتب (SIMATIC Manager)

لفتح البرنامج اضغط مرتين علي تلك الأيقونة بزر الماوس الأيسر ، ستظهر عندئذ واجهة البرنامج Step 7 ، حيث يظهر صندوق حواري (Wizard) يتكون من أربعة مراحل من خلاله سوف تحدد الإطار العام للبرنامج المطلوب إعداده
و الصورة التالية توضح ذلك

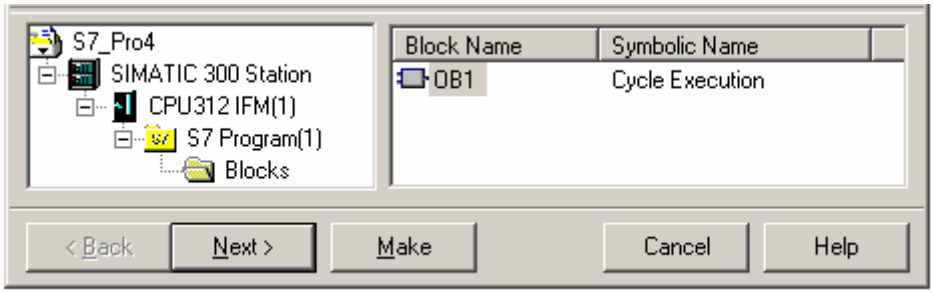


هذه الشاشة تقول الأتي

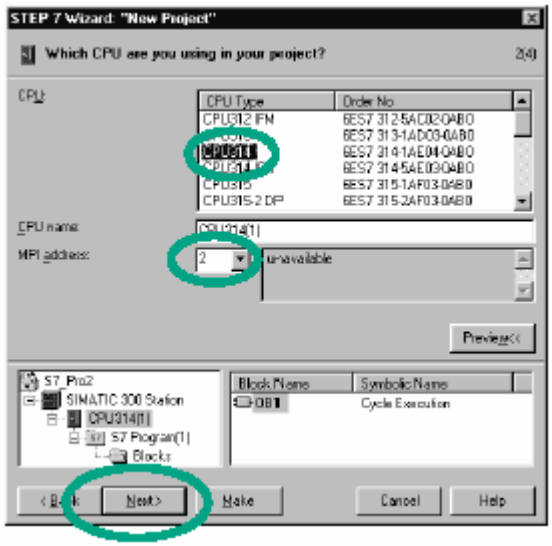
بواسطة الشاشة الحوارية لـ Step 7 تستطيع أن تنشأ المشروع بسرعة وبكل سهولة ، اضغط علي أحد الخيارات الآتية

١- Next حتى تنشأ المشروع خطوة وراء خطوة.

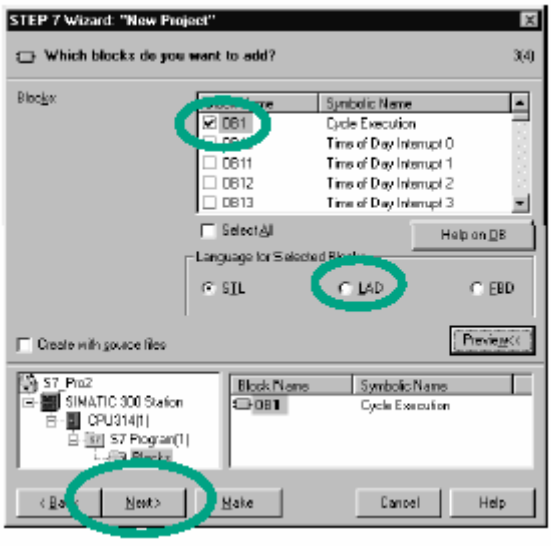
٢- Make حتى تنشأ المشروع حسب التشكيلة الموجودة في أسفل الصندوق الحواري والموضحة بالشكل الآتي



بالنقر علي Preview تظهر أو تخفي القائمة السفلي من الصندوق الحواري و الآن ننتقل إلى الخطوة الثانية بالضغط علي Next عندها سوف تظهر الشاشة الآتية



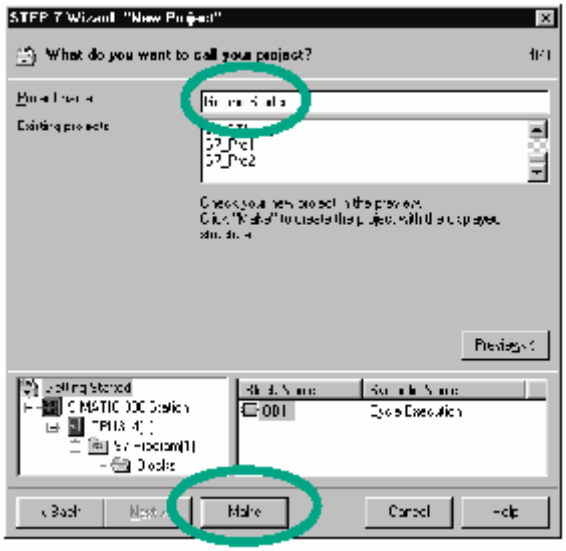
والتي من خلالها نختار نوع المعالج لأن كل معالج له عدة مميزات تميزه عن المعالجات الأخرى فعلى سبيل المثال بالنسبة للذاكرة و عنوانيها تختلف من معالج لأخر، وكذلك نختار عنوان (MPI) Multipoint Interface حيث يتصل جهاز الحاسب مع جهاز PLC من خلال هذا العنوان ،ثم ننتقل إلى الخطوة التالية بالضغط علي Next وسوف تظهر الشاشة



التالية

أختر طريقة تمثيل البرنامج التي ستكتب بها البرنامج (LAD, STL ,FBD) ثم اضغط Next لتنتقل إلى الشاشة الأخيرة

وفى هذه الخطوة نحدد اسما للبرنامج



عند الانتهاء اضغط على make حتى تطبق الإعدادات الجديدة وسوف تظهر واجهة جديدة بعنوان اسم البرنامج الذي سميته .

ملاحظة :-

يمكن تفعيل أو إزالة الصندوق الحواري من خلال وضع علامة ( ) أو إزالتها في المربع الذي يظهر في أول شاشة حوارية

إذا احتجت لمساعدة من قبل البرنامج توجد عندك ثلاث طرق

١- بواسطة الضغط علي F1

٢- افتح القائمة Help

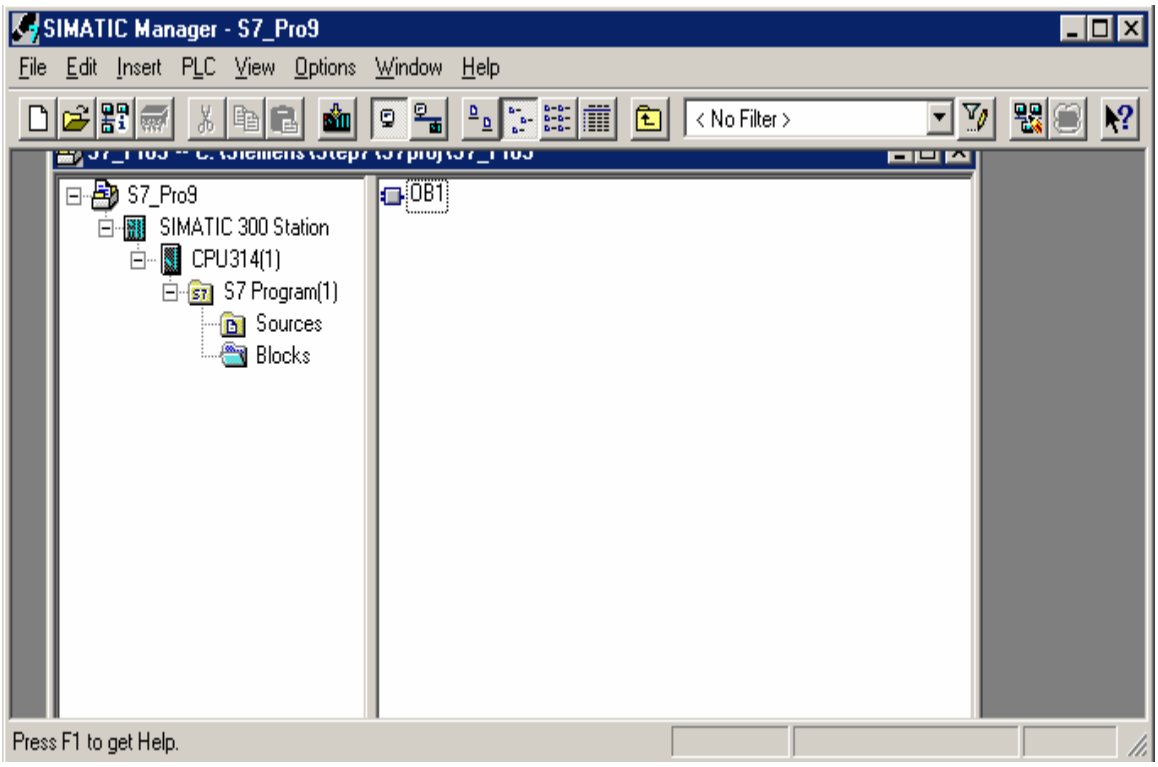
٣- اضغط علي الأيقونة، ثم اضغط علي الكائن المراد معرفة بعد الأشياء عنه

ملاحظة

اضغط علي الشارة + ، - افتح وإغلاق المجلد .

بعد الانتهاء تظهر الشاشة الآتية

مساعدة



الرموز Symbols

وهي تستخدم لإيضاح هل سنستخدم رموز الدخل و الخرج المتعارف عليها مثل I0.0, I0.1,Q 4.0 , أم سنستخدم المفاتيح بأسمائها مثل Key1, Key2, Out3 أو الرموز S1 ,S2, K3

لاستيضاح ذلك

اضغط مرتين علي S7 Program(1) لإظهار محتوياته ، تلاحظ ظهور ثلاث ملفات من بينها Symbols كما في الشكل الأتي

اضغط علي ملف Symbols لفتحه تجد الجدول يتكون من أربعة قوائم هـــي -Symbols Address- DataType – Comments

١- Symbols عبارة عن أسماء او الرموز التي ستفترضها لما يقابلها في وحدة الـــدخل و الخرج .

٢-Address العنوان الحقيقي للرمز .

٣- DataType نوع البيانات .

٤- Comments و مقصود أننا سوف نكتب التعليقات علي أوامر البرنامج الذي تكتبـــه، و كذلك علي كل جزء من البرنامج أيضا، ووضع عنوان علي كل جزء من البرنامج إذا كـــان البرنامج يتكون من مجموعة من الأجزاء .

	Symbol	Address	Data type	Comment
1	Cycle Execution	OB 1	OB 1	
2				

طريقة الكتابة

اختر خلية من قائمة Symbol واكتب فيها الاسم الذي ستفرضه و لـــيكن Switch 1 ثـــم انتقل إلى الـــ Address و أكتب العنوان وليكن I 1.5 تلاحظ أن نوع البيانات يضاف بطريقة أوتوماتيكية في حالتنا(BOOL) ، أما بالنسبة للتعليقات فلك الحرية في استخدامها أم لا ، في النهاية لا تنسي أن تقوم بحفظ عملك بالضغط علي الزر

	Symbol	Address	Data type	Comment
1	Cycle Execution	OB 1	OB 1	
2	switch 1	I 1.5	BOOL	start swich
3				

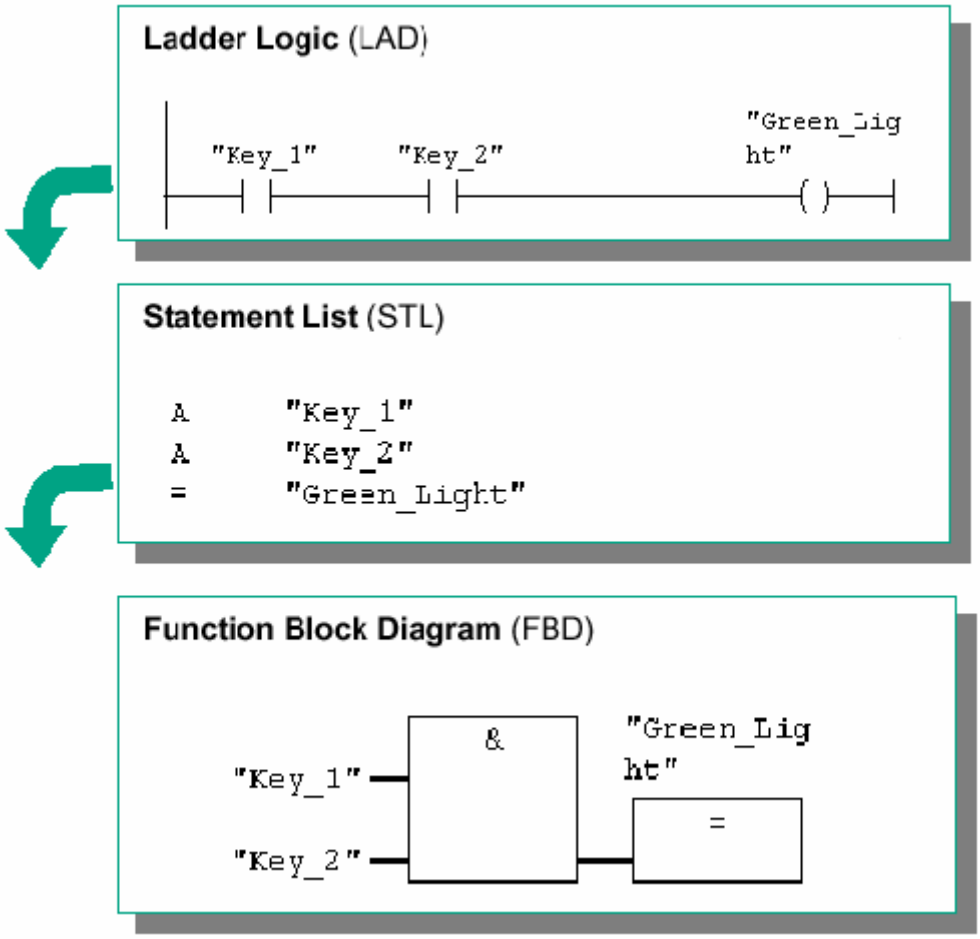
ملاحظة :-

بالنسبة لنوع البيانات التي أضيفت مسبقا بطريقة أوتوماتيكية، فهي تحدد نوع البيانات التـــي يتعامل معها CPU، و برنامج STEP 7 يستخدم عدة أنواع من البيانات ومنها

Bool (It Means Bit 0 Or Bit 1)- Byte- Word –Dword

توضيح بعض النقاط عن إنشاء البرنامج

عليك أن تقرر بأي طريقة سوف تقوم بتمثيل البرنامج أهي بطريقة المخطط السلمي أم هـــي بطريقة البوابات المنطقية أم بطريقة قائمة الإجراءات



ملاحظة

عملية اختيار الطريقة تتم خلال الشاشة الحوارية هذا من ناحية و تستطيع من ناحية أخري إن تغير الطريقة من إحدى قوائم البرنامج فيما بعد

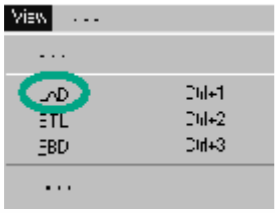
لنقوم بعملية البرمجة اضغط مرتين علي OB1 حيث يفتح برنامج جديد من خلالـــه ســـنقوم بكتابة البرنامج و الشكل الآتي يوضح ذلك

البرمجة باستخدام طريقة المخطط السلمي Progarming In Lader Diagram

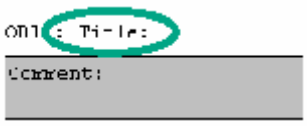
في الجزء الآتي سوف نقوم بعمل دائرة توالي ، ودائرة توازي ، و دائرة الإلغـــاء و الإبقـــاء بطريقة المخطط السلمي .

أولا دائرة توالي

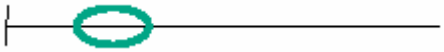
١. اذهب إلـــى قائمـــة View و اختر منها LAD



٢. اضغط علي Title Area وأكتب فيها اسم دائرة التحكم



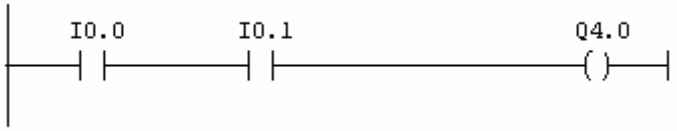
٣. حدد المسار



٤. اضغط مرتين علي المفتاح

٥. ثم اضغط

٦. قم بتســـمية المفـــاتيح و ذلـــك بتعليم المفتاح و كتابة عنوانه



ثانيا دائرة التوازي

١. اختر Network 1

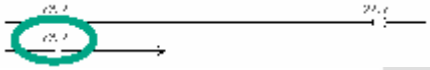
٢. ادرج فرع جديد

٣. ادرج مفتاح مفتوح و مخرج

٤. علم الخط العمودي القصير

٥. ادرج فرع توازي

٦. ادرج مفتاح طبيعي مفتوح



٧. اغلق الفرع

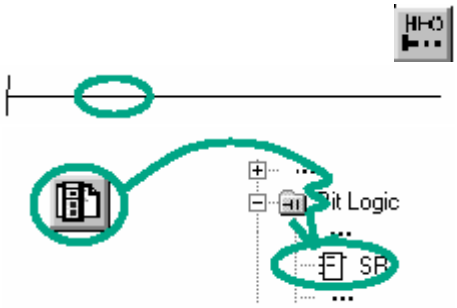
٨. قم بتسمية المفاتيح و ذلك بتعليم المفتاح و كتابة عنوانه

ثالثا دائرة الإلغاء و الإبقاء

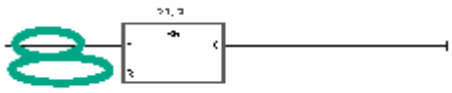
١. ادرج شبكة جديدة

٢. اختر المسار

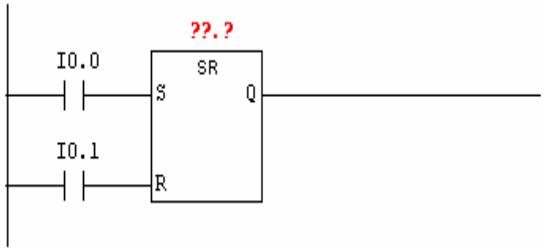
٣. اذهب إلى قائمة Insert ثم اختر منها Program Elements ثم اختر Bit Logic واختر SR بالضغط علية مرتين



٤. ادرج مفتاح لكل من المـــداخل S,R



٥. قم بتسمية المفاتيح و ذلك بتعليم المفتاح و كتابة عنوانه



ملاحظة :

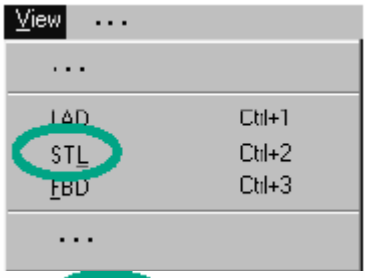
لا تنسي أن تقوم بحفظ عملك

البرمجة باستخدام قائمة الإجراءات Programming In Statement List

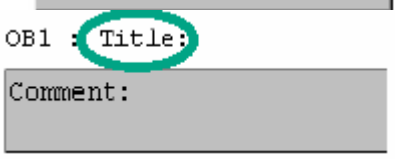
في هذا الجزء سوف نقوم بتوضيح كيف يتم كتابة الدوال المنطقية باستخدام قائمة الإجراءات

أولا تعليمةِAND

١. اذهب إلى قائمة View و اختر منها STL



٢. اضغط علي Title Area وأكتب فيها اسم دائرة التحكم



٣. حـــدد المنطقـــة الأســـفل التعليقcomment

٤. اكتب A واترك مسافة ثم أكتب العنوان وعند الانتهاء اضـــغط مفتاح التنفيذ ENTER

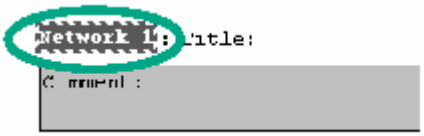
A I 0.0

٥. ثم اكمل بطريقة مماثلة

A I 0.1
= Q 4.0

ثانيا تعليمه OR

١. اختر Network 1



2. ادرج شـــبكة جديـــدة و حـــدد المنطقة الأسفل التعليق

٣. اكتب O واترك مسافة ثم أكتب العنوان وعند الانتهاء اضـــغط مفتاح التنفيذ ENTER

O I 0.0

٤. ثم اكمل بطريقة مماثلة

O I 0.1
= Q 4.0

ثالثا تعليمه الإبقاء و الإلغاء

١. ادرج شـــبكة جديـــدة و حـــدد المنطقة الأسفل التعليق

٢. اكتب A واترك مسافة ثم أكتب العنوان وعند الانتهاء اضغط مفتاح التنفيذ ENTER ، ثم اكتب S ثم اتبعه بالعنوان ثم اضغط ENTER ، ثم اكتب A ثم اتبعه بالعنوان ثم اضغط ENTER ، ثم اكتب R ثم اتبعه بالعنوان

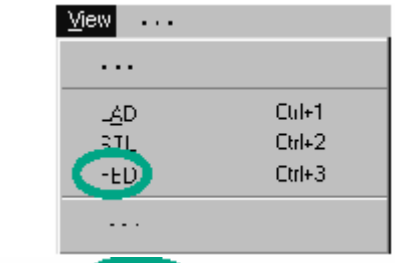
A I 0.0
S Q 4.0
A I 0.1
R Q 4.0

البرمجة بطريقة البوابات المنطقية Programing In Function Block Diagram

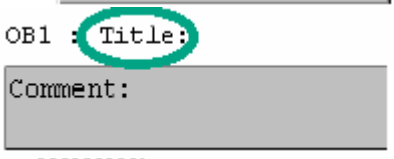
في الجزء التالي سوف نقوم بأذن الله ببرمجة كل من تعليم AND ، وتعليمه OR ، و تعليمه الإلغاء و الإبقاء

أولا تعليمه AND

١. اذهب إلى قائمة View و اختر منها FBD



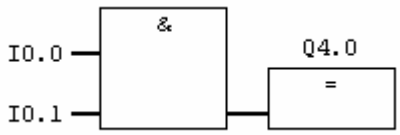
٢. اضغط علي Title Area وأكتب فيها اسم دائرة التحكم



٣. حدد المنطقة الأسفل التعليقcomment

٤. ادرج العنصرين الآتيين

٥. ليا خذ الشكل الآتي ، ثم الكتب العناوين .



٦. لزيادة افرع الدخل للبوابة نحدد البوابة ثم مـــن قائمـــة الادوات نختار الرمز الاتي

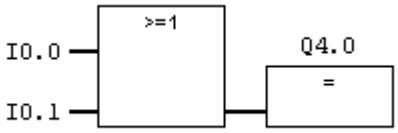
ثانيا تعليمه OR

١. ادرج شبكة جديدة

٢. حـــــدد المنطقـــــة الأســـــفل التعليقcomment

٣. ادرج العنصرين الآتيين

٤. ليا خذ الشكل الآتي ، ثم الكتب العناوين .



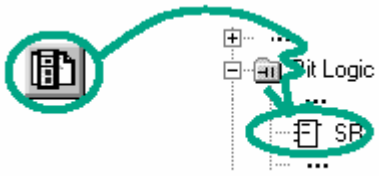
٥. احفظ عملك

ثالثا تعليمه الإبقاء و الإلغاء

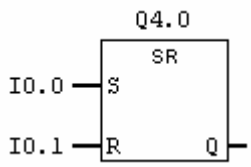
١. ادرج شبكة جديدة

٢. حـــــدد المنطقـــــة الأســـــفل التعليقcomment

٣. اذهب إلى قائمة Insert ثم اختر منها Program Elements ثم اختر Bit Logic واختر SR بالضغط علية مرتين



٤. ثم اكتب العناوين



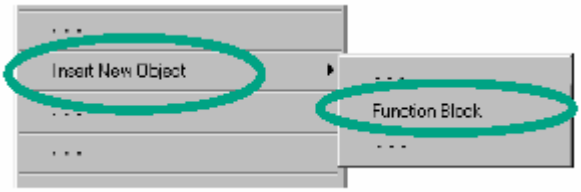
٥. احفظ عملك

بهذا بينا كيفية كتابة التعليمات الثلاثة بطرق البرمجة الثلاثة .

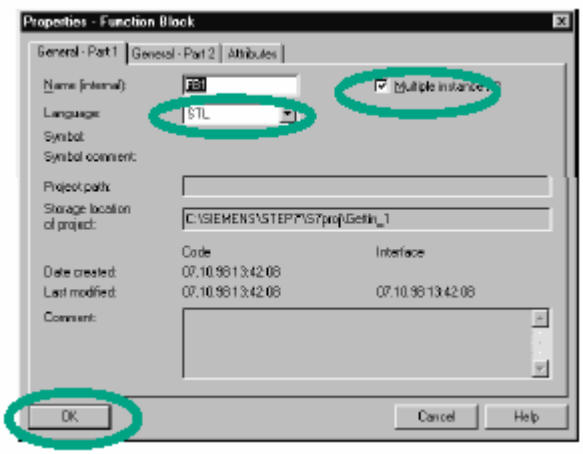
إنشاء البرنامج باستخدام Function block و Data blocks

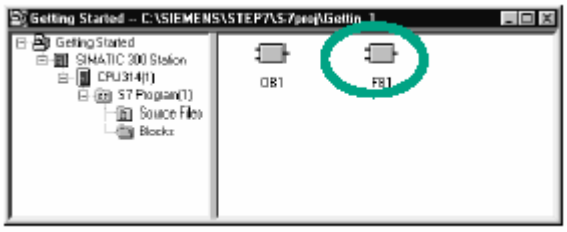
Function block يندرج تحت (organization blocks) من ناحية تركيبية و هو عبـــارة عن برنامج فرعي يتم استدعاءه مرة واحدة أو اكثر وذلك يتوقف علي طبيعة البرنامج المعد ، وكل المعاملات المتعلقة به تخزن في DATA BLOCKS

و لعمل FB استعرض المجلد BLOCKS وقم بفتحة و اضغط بزر الماوس الأيمـــن فـــي الجانب الأيمن من البرنـــامج و اختـــر منهـــا INSERT NEW OBJECT ثـــم اختـــر function blocks



اضغط مرتين علي fb لتفتح خصائص تلك الكائن ، من خلال الصندوق الحواري الذي ظهر اختر طريقة البرمجة التي ستكتب فيها البرنامج وضع علامة أمام multiple instance ، ثم اضغط ok لتنفذ الأعدادات الجديدة حيث تلاحظ ان FB قد ادرج



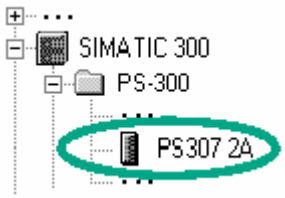


إعدادات جهاز Plc

قبل عملية تحويل برنامج التحكم إلى وحدة plc يجب علينا أن نعمل إعدادات لـ Hardware مثل وحدة التغذية و حدات الدخل و الخرج و منها نقول إن نقطة البداية هي أن نفتح Manager Simatic تلاحظ في الجانب الأيمن هيكلية المشروع ، افتح المجلد Simatic 300 Station تلاحظ ظهور الكائنين Hardware و CPU313

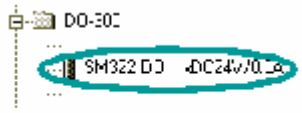
اضغط علي Hardware لفتحة لتتم من خلاله إعدادات وحدات جهاز التحكم المنطقي المبرمج ،نافذة HW Config تفتح و CPU الذي اخترته في بداية إنشاء المشروع

١. لتحديد وحدة التغذية ، استعرض القائمة في الجانب الأيمن حتى تصل إلى PS307 2A ثم اضغط عليه و اسحبه إلى فتحة التوسعة رقم ١

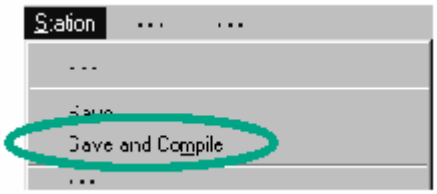


٢. لتحديد وحدة الدخل ، استعرض القائمة تجد مجلد اسمه SM-300 اختر منها DI-300 ثم اختر SM321 DI16*DC24V وضعه في فتحة التوسعة رقم ٤

٣. بطريقة مماثلة اذهب إلى DO_300 واختر منها SM322 DO16*DC24V/0.5A و اسحبه إلى فتحة التوسعة ٥



٤. في النهاية اختر من قائمة Save And Compile File



٥. **في النهاية تحصل علي الشكل الأتي**

Slot	Module	Order Number	MPI Address	I Add	Q	Comment
-	PS307 2A	6ES7 307-1BA00-0AA0				
2	CPU314(1)	6ES7 314-1AE04-0AB0	2			
3						
4	DI32xDC24V	6ES7 321-1BL00-0AA0		0...3		
5	DO32xDC24V/0.5A	6ES7 322-1BL00-0AA0			4...7	
6						

٦. ملاحظات :-

١. فتحة التوسعة رقم ٣ تبقى فارغة .
٢. DI اختصار Digital Input
٣. DO اختصار Digital Output
٤. SM321 رمز لوحدات الدخل
٥. SM322 رمز لوحدات الخرج

ملاحظات في البرمجة

١. اضافة Network جديدة في البرنامج نعلم عنوان Network السابقة لها ثم نضغط علي Insert ونختار منها الامر Network تظهر بعدها لنا دائرة جديدة فارغة .

٢. لاعادة نسخ Network كاملة نعلم عنوان الدائرة المطلوب نسخها ثم نضغط علي قائمة Edit زنختار منها الامر Copy ثم نعلم عنوان الدائرة التي نريد رسم الدائرة التي نريد رسم الدائرة الجديدة ونختار من قائمة Edit الامر Past .

٣. لمسح البرنامج الموجود في ذاكرة جهاز PLC نضع مفتاح التشغيل علي الوضع Stop ومن قائمة PLC علي الحاسوب نختار Clear/Reset .
كما انه يمكن مسح البرنامج الموجود في ذاكرة جهاز PLC يدويا بوضع مفتاح التشغيل على الوضع MRES .

تحويل البرنامج و اختباره
Downloading and Debugging the Program

سنقوم الآن بتوضيح عملية تحويل البرنامج إلى وحدة PLC وكيفية اختباره

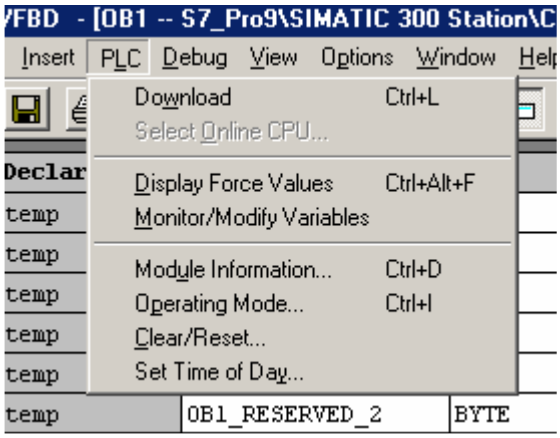
للقيام بعملية تنزيل البرنامج إلى الوحدة اتبع الخطوات الآتية :

١. صل جهاز Plc بواسطة الكابل الخاص الذي يوصل الوحدة بمفند التسلسلي (com1) فى الحاسب الالي .

٢. صل الوحدة بمصدر التغذية 220/110 فولت (يمكن تحديد جهد المصدر عن طريق مفتاح اختياروهو 220V) .

٣. شغل المفتاح الرئيسي والمفتـــاح الخـــاص بوحدة التغذية الداخلية .

٤. نختار الوضع stop ثم اذهب القائمـــة plc واختر منها Download .



٥. **لاختبار البرنامج** فيتم عن طريق الذهاب إلى قائمة Debug ثم نفعل منها الأمر Monitor

ملاحظة:-

هناك أربع أوضاع لتشغيل هي

١. MRES (Memory Reset) يلغي البرنامج المخزن و لعمل ذلك انتقل إلى الوضع MRES أبقه كذلك لمدة ٣ ثواني علي الأقل لغاية المبين الأحمر يومض ببطئ ، بعد ذلك أرخه ثم عاود من جديد لتحركه إلى الوضع MRES ، عندما تلاحظ انه يومض بسرعة عندها يكون البرنامج المخزن قد أزيل .

٢. STOP عند هذا الوضع يتم إرسال البرنامج ويتم إيقاف تنفيذ البرنامج أيضا.

٣. الوضع STOP,RUN-P عندهما يمكن تنزيل البرنامج ولكن الفرق بينهما في أن الوضع RUN-P ينفذ البرنامج المنزل تلقائيا .

٤. الوضع RUN يستخدم في حالة تنفيذ البرنامج .

استعادة البرنامج لقرائته أو تعديله

لاستخراج البرنامج المخزن في جهاز الـــــ PLC أو المخزن فى ذاكرة الألة المراد صيانتها لقرائته أو تعديله نتبع الخطوات التالية :-

١. ننشأ ملف جديد بنفس إعدادات خطوات كتابة برنامج جديد حيث تظهر عندنا شاشـــة مشابهة لشاشة كتابة البرنامج الجديد أو نعتمد علي الإعدادات الموجودة في الجهاز .
٢. نذهب إلى قائمة View ونختار منها الأمر Online حيث يظهر عنـــدنا البرنـــامج الموجود علي جهاز PLC و لكن بالاسم الذي اخترناه نحن في الإعدادات (أي بدون الاسم الأصلي للبرنامج) ثم نضغط علي OB1 لفتحه وعمل التعديل المطلوب .

تمرين ١

يوجد في أحد أنفاق القدس أربع مراوح لتهوية النفق و مطلوب تنفيذ برنامج التحكم بحيث أن يكون هناك ثلاث مصابيح بيان في غرفة التحكم الخاصة بالنفق و هذه المصابيح هي : مصباح اخضر ويضئ عندما تكون المراوح كلها تعمل و مصباح اصفر يضئ عندما تكون ثلاثة من المراوح من أربعة تعمل ومصباح أحمر يضئ عندما تكون مروحيتين أو اقل يعملان فقط نفد هذا التمرين علي جهاز Plc .

تمرين ٢

هناك مؤسسة يتخذ قرارها بناء علي نسبة اسهم الأعضاء و يتكون مجلس إدارة المؤسسة من ٤ أعضاء الأول ٤٠ % من اسهم المؤسسة و الثاني له ٢٠ % و الثالث له نسبة ٢٠ % و لرابع له ١٠ % و تتم الموافقة علي القرار إذا حقق نسبة ٦٠ % حيث يأخذ صوت كل عضو بنسبة أسهمه في المؤسسة مطلوب تنفيذ دائرة و برنامج التحكم اللازم بحيث يكون أمام كل عضو مفتاح يستخدمه لإعطاء راية في القرار و يكون المفتاح On في حالة الموافقة علي الفرار و يكون Off في حالة عدم الموافقة ة يكون هناك لوحتان متصدرة غرفة الاجتماع و لوحة خضراء تضئ في حالة موافقة الأعضاء علي القرار و لوحة حمراء تضئ في حالة رفض الأعضاء جميعهم .

الفصل الثالث
و حدة الربط بين جهاز PLC و جهاز الكمبيوتر

هذه الوحدة عبارة عن كابل مزود بمحول خارجي (لا يحتاج إلى فتحة توسعة (PC slot) مما يمكنه ان يعمل أيضا علي أجهزة الحاسب الالي المحمول) تستخدم لربط نظـــام S7 بجهـــاز الكمبيوتر ، حيث يتم ربط جهاز الحاسب عن طريق المنفذ التسلسلي (com 1) وربط الجهـــة الأخرى من الكابل بجهاز plc، وهذه الوحدة مزودة بمبينين إحداهما لإشـــارة علـــي وجـــود اتصال ، و الآخر يضئ لحظة وجود نشاط أي لحظة إرسال البرنامج ، و الشكل الأتي يوضح ذلك .

الفصل الرابع

عناصر التمثيل و البرمجة

المقصود بعناصر التشغيل هي العناصر المستخدمة في تمثيل دائرة التحكم المنطقي وهي

١. البوابات المنطقية – و المفاتيح NO , NC – ومسجلات العلامات .
٢. دائرة الإلغاء الإبقاء R/S .
٣. المزمنات .
٤. العدادات .

وسوف نبدأ باستعراض هذه العناصر و كيفية تمثيلها و كيفية الاستفادة منها في دوائر التحكم المنطقي المبرمج .

١- البوابات المنطقية والمفاتيح NO, NC

فالمفاتيح التي نتعامل معها هي NO , NC وهي تظهر عند استخدام طريقة LAD
أما البوابات فهي تظهر عند استخدام الـــ FBD و هي بوابة & وبوابة OR وبوابة NOT وهي معروفة لدينا وقد سبق دراستها في مراحل سابقة .

٢- مسجلات العلامات

إنها من العناصر المساعدة في عملية البرمجة والتشغيل في عمليات التحكم المبرمج وهي تقوم بالواسطة لنقل حالات الدخل إلى الخرج .
مسجلات العلامات هي أماكن موجودة في الذاكرة الخاصة بجهاز التحكم المنطقي المبرمج و يرمز لها بالرمز M .
ويتم عنونتها ابتداء من المسجل M 0 إلى المسجل M 255
أما المسجل M0 (وأي مسجل من مسجلات العلامات) يتكون من ٨ Bit
وتكون كما يلي

M0								
	.0	.1	.2	.3	.4	.5	.6	.7

لذلك فإننا لدينا ٨ علامات هي M0.0,M0.1,M0.2,M0.3,M0.4,M0.5,M0.6,M0.7
ويتكرر ذلك بالنسبة لباقي مسجلات العلامات كالأتي

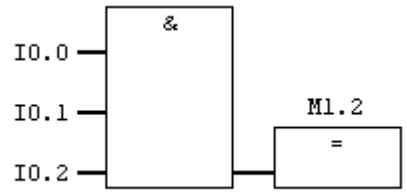
و علي سبيل المثال عناوين الخانات المشهرة هي

M0.2 , M1.0,M2.2 ,M2.7,M 3.3, M255.5.

	.0	.1	.2	.3	.4	.5	.6	.7
M0								
M1								
M2								
M3								

M2.55								

ولتحويل الاستفادة منها، فمثلا لو لدينا الدائرة الآتية



فهذا يعني عندما تكون كل دخول بوابة الـــ & تساوي (١) فـــان خانـــة العلامـــة M1.2 يسجل بها (١) وإذا كان خرج بوابة & (٠) فانه خانـــة العلامـــة M1.2 يســـجل بهـــا (٠)ويمكن القول بأننا يمكن استخدام مسجلات العلامات لتخزين فيها "١" أو "٠" تبعا لحالـــة استخدامها في دائرة التحكم .

مثال

اعد حل تمرين (١) و (٢) ص لاستخدام مسجلات العلامات

٢-دائرة الإلغاء الإبقاء R/S

هذه الدائرة تقوم علي دالتين هما دالة الإلغاء ودالة و الإبقاء

_ دالة الإبقاء (SET-S) هي تحافظ علي حالة توصل الخرج في حالة إعطائها إشـــارة [حتى لو كانت هذه الإشارة المعطاة لها لفترة زمنية صغيرة جدا مثل استخدام المفتاح الضاغط حيث انه عند الضغط علبه يعطي إشارة (i.e. 1) مما يجعـــل s = ١) و يـــتم توصـــيل الخرج و عند الرفع عنه (عدم الضغط) تلغي هذه الإشارة (i.e. s=0) ولكن دالة الإبقـــاء تحافظ علي توصيل هذا الخرج] .

_ أما دالة البقاء هي تلغي حالة التوصيل للخرج في حالة إعطاءها إشارة [حتى لـــو كانـــت الإشارة لفترة زمنية صغيرة مثل إشارة مفتاح الضاغط]

و تعرف دالة (دائرة) الإلغاء و الإبقاء بـــــ S/R LATCH

وهي لها دخلين هما ضبط (SET) واعادة ضبط (RSEET) .

وبناء علي حالة الدخلين يكون الخرج Q ويبقي هذا الخرج علي وضعة إلا أن يتغير وضـــع الدخل S أو R مرة أخرى وجدول الحالة لها هو

RS LATCH

R/S Latch with set Priority

S/R Latch with reset Priority

S	R	Q (binary output)
0	0	Old status retained
0	1	0
1	0	1
1	1	0 for S/R Latch
		1 for R/S Latch

ونجد من الحالات أن عندما يكون R=0 , S=0

فان الخرج Q يكون علي حالته القديمة التي كان عليها أي إذا كانت Q=1 قبل أن نوضـــع علي R,S=0 فان Q تبقي كما هي تساوي ١ بعد وضع R,S=0 .

S=0, R=1 > Q =0

وهي تعني إن الخرج Q عندما تكون S=0, R= 1 و لا يكون هناك خرج علي الدائرة (الإلغاء)

S=1, R=0 > Q =1

وهي تعني إبقاء الخرج موجود عندما تكون S=1, R=0

S=0, R=1

و هذه الحالة لها خرجين و هما

إما Q=0 أي نلغي الخرج و نحصل علي هذا عندما نستخدم دائرة S/R

أو Q=1 أي يبقي الخرج و نحصل علي هذا عندما نستخدم دائرة R/S

وستضح التفاصيل عند استخدامه اكثر في الدوائر .

تمرين ٣

يوجد لدينا ثلاث خزانات تملأ بثلاث سوائل مختلفة و يتم ملئ هذه الخزانات عن طريق صمام حيث يوجد لكل خزان الصمام الخاص به ويوجد داخل كل خزان مجسان مجس علوي ويعطى إشارة إذا ارتفع السائل داخل الخزان اعلي منه و مجس سفلي إذا انخفض السائل في الخـــزان إلى أقل منه و المطلوب تنفيذ دائرة التحكم بحيث يفتح كل صمام إذا وصل معدل السائل داخل الخزان الخاص به إلي اقل من المجس السفلي ويتم ملي الخزان ويغلق الصـــمام إذا وصـــل معدل السائل داخل الخزان به أعلي من المجس العلوي .

المزمنـــــــــــات Timers

إن من أهم عمليات التحكم هو التحكم التتابعي ويوجد في التحكم التابعي . تحكم تتتابعي يعتمد على الزمن. ويسمح لنما النظام باستخدام خمس أنواع منة المزمنات وسوف نستعرضهم جميعا لمعرفة الفرق بينهم .

و الزمنات لها مكان محجوز في الذاكرة ذاكرة CPU هذا المكان طوله 16 خلية لكل مزمن و عدد المزمنات تختلف من جهاز لأخر تبعا لنوعية المعالج المركزي ، ومع هذا في حالتنا يوجد 512 مزمن .

أما بالنسبة لزمن المزمن فيتم كتابته بإحدى الطريقتين الآتيتين ولك الحرية أخي في اختيار أي منها :

١- **الطريقة الأولي**

أن يكتب علي الهيئة S5T#aH_bbM_ccS_ddMS

حيث a عدد الساعات ، b الدقائق ، c الثواني ، d ميلي ثانية .

مثال

40 ثانية	S5T#40S
ساعة و 15 دقيقة	S5T#1H_15M
ساعتين و 46 دقيقة و 20 ثانية	S5T#2H_46M_20S

٢- **الطريقة الثانية**

أن يكتب علي الهيئة W#16#wxyz

حيث w المعامل ، xyz الزمن .

حيث يتم ضرب المعامل في الزمن للحصول علي الزمن الحقيقي للمزمن ، أما المعامل فهو موضح بالجدول الأتي .

Time Base	Binary Code for the Time Base
10 ms	00
100 ms	01
1 s	10
10 s	11

وكل مزمن له مفاتيح دخل ومفاتيح خرج .

أما مفاتيح الدخل فهي :

1-مفتاح بدء تشغيل المزمن S ليبدأ في الوظيفة الزمنية المطلوبة

2-مفتاح إنهاء تأدية تشغيل المزمن R

2-مفتاح القيمة الزمنية ????? = TV المطلوبة لتشغيل المزمن

أما مفاتيح الخرج فهي :

4-الخرج المراد تشغيله Q

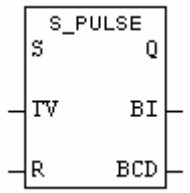
BI -5

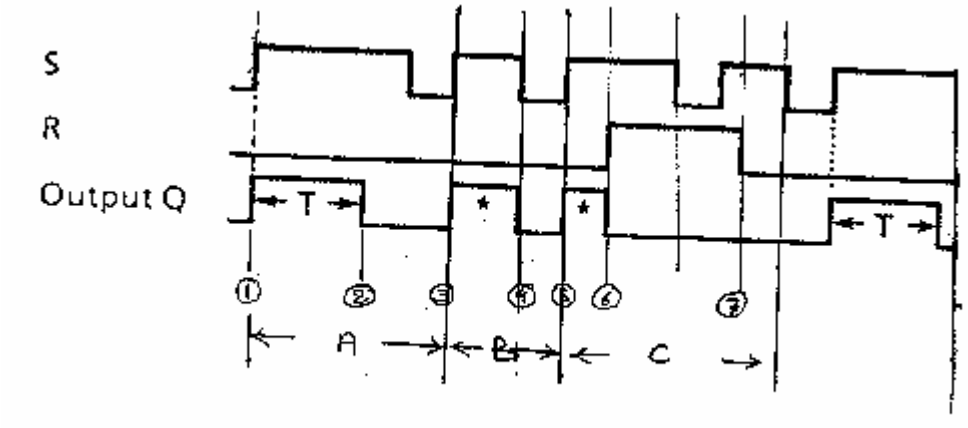
BCD -6

وسنبدأ الآن في استعراض كل نوع من المزمنات وشرح الوسيلة الأساسية له .

1- Pulse Timer SP

الرمز





في هذا المزمن نجد أن :

عند النقطة 1 وبعد تشغيل مفتاح S للمزمن يكون هناك خـــرج Q ، ويســـتمر الخـــرج Q بمقدار الفترة الزمنية T حيث أننا نحن اللذين نقوم بتحديد الفترة الزمنيـــة عـــن طريـــق K ويستمر الخرج Q موجود وينتهي عند النقطة 2 بالرغم من أن مفتاح S ما زال يعمل .

وعند النقطة 2 نجد إننا بدأنا في تشغيل مفتاح S وقبل انقضاء الفتـــرة الزمنيـــة T قمنـــا بإغلاق المفتاح S ومع إغلاق S فان الخرج Q أصبح = صفر أي لا يوجد خرج وذلك عن النقطة 4 .

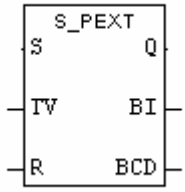
عند النقط ة 5 بدأنا في تشغيل المفتاح S وبدا المزمن يعمل ولكن عند النقطـــة 6 اشـــتغل المفتاح R فحول الخرج Q إلى صفر مباشرة بالرغم من أن الـــــ S فـــي الوضـــع ON ونستنتج أن المفتاح R إذا كان ON فان الخرج مباشرة يكونOFF

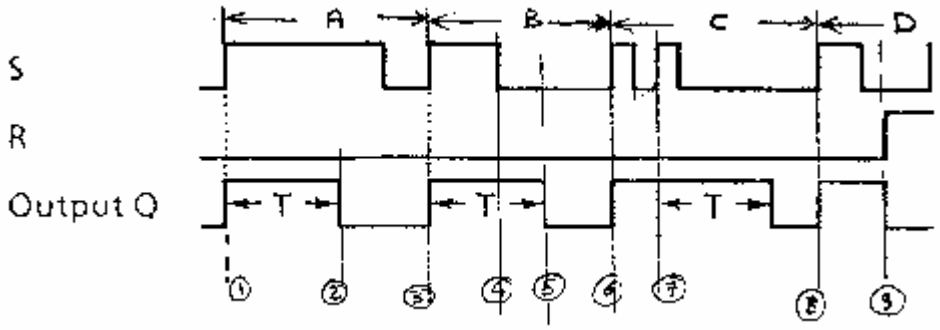
عند النقطة 7 نجد أن بعد أن تم وضع المفتاح R في الوضع OFF بالرغم من أن مفتاح S في الوضع ON فان الخرج ما زال = صفر .

وللحصول علي خرج لا بد من تغير مفتاح S من ON إلى OFF ثم ON مرة ثانية كي يبدأ المزمن في إخراج خرج .

2-Extended Pulse Timer SE

الرمز





في هذا المزمن نجد أن :

عند النقطة 1 عند وضع مفتاح S في الوضع ON يبدأ المزمن يعمل ويعطي الخرج Q ويستمر هذا الخرج لفترة زمنية T التي يتم تحديدها عن طريق KT وينتهي الخرج عــن النقطة 2 بعد انتهاء المزمن T بالرغم من استمرار S = ON.

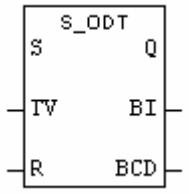
عند النقطة 2 يكون S في الوضع ON ويبدأ المزمن في إعطاء خرج ولكن عند النقطــة 4 نضع الــ S في الوضع OFF ولكن ما زال الخرج Q موجود ويستمر الفترة الزمنية T وينتهي عند النقطة 5 دون التأثر بان الــ S علي الوضع OFF .

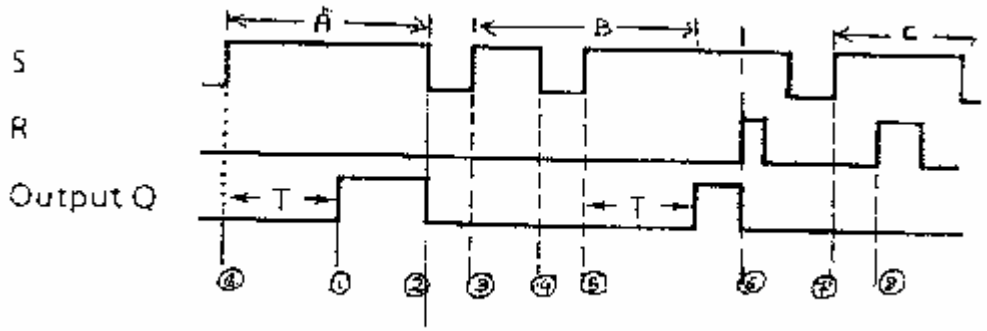
_ عند النقطة 6 نضع S في الوضع ON فبدا المزمن في العمل واعطاء خرج وتغيــر وضع الــ S إلى OFF ثم إلي IN مرة أخري فنجد أن المزمن يبدأ حساب الفترة الزمنية ابتداء من أخر مرة ثم وضع المفتاح S علي الوضع ON.

عند الوضع 8 نضع S في الوضع ON فيبدأ المزمن في العمل واخراج خرج إذا أن يتم وضع المفتاح R في الوضع ON فيصبح الخرج = صفر مباشرة وذلك عند النقطة 9 .

3- On Delay Timer SD

الرمز





في هذا المزمن نجد أن :

عند النقطة 0 نفع مفتاح S في الوضع ON و نجد ان الخرج لا يبدأ إلا عند النقطة 1 وبعد مرور الفترة الزمنية المحددة غلي الطرف TVيستمر الخرج يعمل إلى أن يتم تحويل S إلى OFF فينفصل الخرج وذلك عند النقطة 2

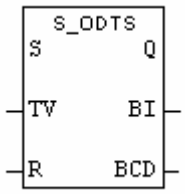
عند النقطة 2 تم وضع المفتاح S في الوضع ON ثم عند النقطة 4 ثم تغيير وضع المفتاح إلى الوضع إلى الوضع OFF ولكن لم تمر الفترة الزمنية المحددة و بالتالي لم يكن هناك خرج و عند النقطة 5 ثم اعادة المفتاح S إلى الوضع ON ويبدأ المزمن في العمل و يعطي خرج بعد الفترة الزمنية T

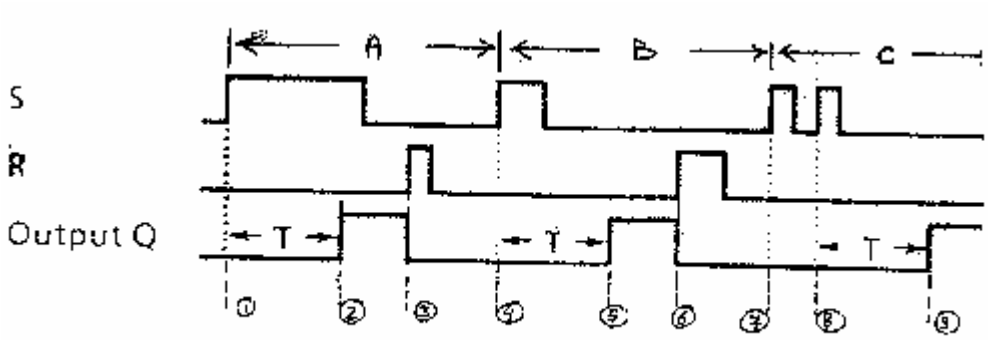
عند النقطة 6 نجد انه بالرغم من أن المفتاح S علي الوضع ON ويوجد خرج إلا انه بعد تشغيل المفتاح R فان الخرج بصل مباشرة إلى القيمة 0

عند النقطة 7 ثم تشغيل المفتاحS وقبل انقضاء الفترة الزمنية T ثم تشغيل مفتاح الـ R فبالتالي لن يخرج لان R يجعل الخرج يساوي 0

4-Retentive ON Delay Timer SS

الرمز





وفي هذا المزمن نجد أن :

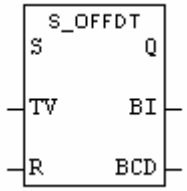
عند النقطة 1 نضع المفتاح s في الوضع ON ويبدأ المزمن يعمل حتى تنقضي الفترة الزمنية t المحددة علي الطرف kV وبعدها يعطي خرج وذلك عن النقطة 2 ونجد انه من الغم من ان المفتاح S تغير إلى الوضع OFF فان الخرج ما الموجود ولا يقف الخرج إلا عند التشغيل المفتاح r وذلك عند النقطة 2

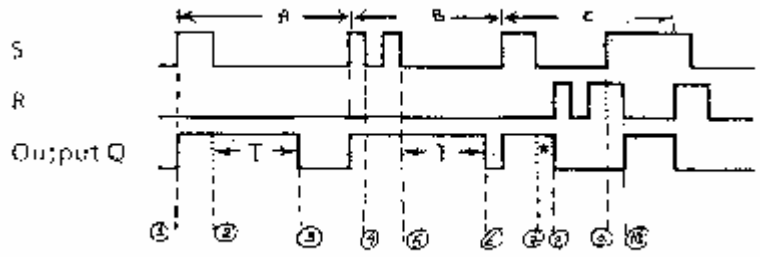
عند النقطة 4 نضع المفتاح s في الوضع ON ويبدأ المزمن في العمل و بالرغم من أننا غير المفتاح إلى OFF إلا ان المزمن يعمل و لا يقف إلا عند تشغيل مفتاح r وذلك عند النقطة 6

عند النقطة 7 نضع المفتاح S في الوضع ON ثم تتم تغيره إلى OFF و إعادته إلى ON مرة أخري عند النقطة 8 و نجد ان المزمن بدأ يعد من النقطة 7 ولكن يقف ويبدأ عد مرة أخري من النقطة 8 ويعطي خرج بعد انقضاء الفترة الزمنية T

5-Off Delay Timer

الرمز





و يعرف باسم مسجل تأخير إيقاف

وفي هذا المزمن نجد أن المفتاح S تم وضعه علي الوضع ON وتبع ذلك وجود خرج و لكن عند النقطة 2 نجد إن المفتاح S إلى الوضع OFF ولكن عند النقطة 2 نجد إن تغير المفتاح S إلى الوضع OFF ولكن الخرج مستمر موجود ويستمر الخرج فترة زمنية قدرها T وتحسب من لحظة وضع المفتاح S في الوضع OFF

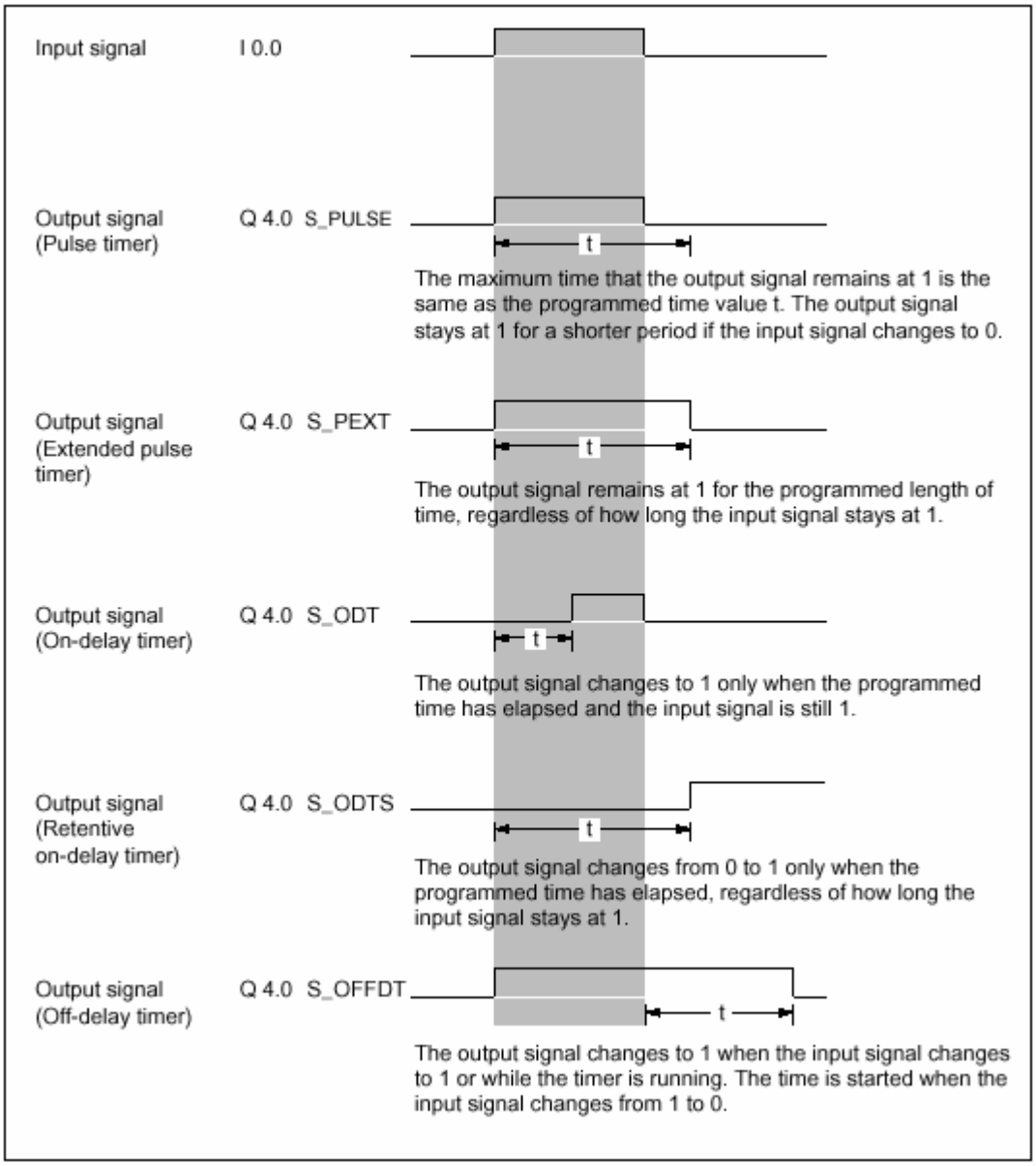
وبعد الفترة T يصبح الخرج مساوي لـ 0 وذلك عند النقطة 2

عند النقطة 4 نجد إننا ضعنا المفتاح S في الوضع OFF فيبدأ المزمن يعمل ولكننا غيرنا وضع المفتاح S إلى الوضع ON فوقف المزمن ويبدأ يعمل مرة أخري عندما نغير الـ S إلى الوضع OFF وذلك عند النقطة 5 ويبدأ يحسب t من عند هذه اللحظة

عند النقطة 7 تم وضع الـ S في الوضع OFF و اخرج ما زال موجود و المزمن بدأ يحسب الفترة الزمنية T و لكن فجا تم وضع المفتاح R في الوضع ONفحول الخرج إلى 0 عند النقطة 8

عند النقطة 9 تم وضع الـ S علي ON و لكن ما زال R علي ON فلذلك لا يوجد خرج و لكن بمجرد وضع الـ R علي OFF يظهر لنا خرج كما هو عند النقطة 10

ويمكن تلخيص الأنواع الخمسة السالفة الذكر بالشكل التي



تمرين ٤

محرك حثي ثلاثي الأوجه يعمل نجمة دلتا ويراد تشغيله بحيث يعمل أولا نجمة لتقليل تيـــار البدء و بعد فترة زمنية ٦ ثواني يتم تحويله إلى دلتا عن طريق مزمن

و مطلوب تنفيذ الدائرة المبرمجة اللازمة لتحقيق ذلك مستخدما مصهر لحماية الدائرة و قاطع حراري و ضاغط تشغيل و ضاغط إيقاف .

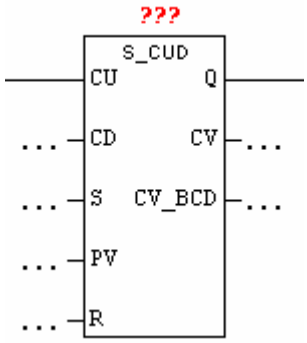
العدادات Counters

العدادات لها مكان محجوز في الذاكرة (ذاكرة cpu) وهذا المكان طواه ١٦ خلية لكل عـــداد وعدد العدادات يختلف من جهاز لآخر تبعا لنوعية المعالج المركزي ومع هذا في حالتنا يوجد ٢٥٦ عداد وقيمة العداد تتراوح ما بين 0–999 ، أما بالنسبة لقيمة العداد فيتم كتابته بالطريقة الآتية

C#xxx حيث xxx تتراوح ما بين 0–999 ، ومع هذا فان هذا النظام يدعم ثلاثة أنواع مـــن العدادات سنسردها بالتفصيل

١- عداد تصاعدي / تنازلي UP- Down counter

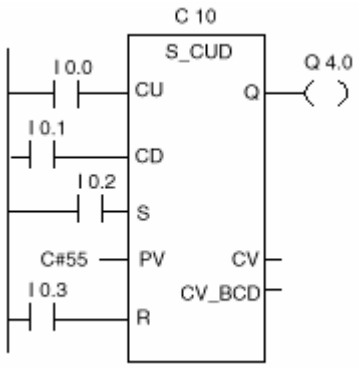
الرمز



عند وضع قيمة منطقية ١ علي المدخل s للعداد يبدأ العداد في العمل بتحميله القيمة علي الطرف pv أما عند وضع قيمة منطقية ١ علي الطرف R فان العداد يتوقف عن العمـــل و بذلك تكون القيمة pv تساوي الصفر

هذا بالنسبة لتشغيل العداد و إيقائه أما بالنسبة لتحديد طبيعة عملة هل يعمل تصاعدي ، هل يعمل تنازلي فهذا يعتمد علي القيمة المنطقية علي كن من الطرفين cu , cd ، فـــإذا أتـــت علي الطرف cu فان العداد يتزايد بمقدار واحد صحيح أما إذا أتت علي الطرف cd فـــان العداد يتناقص بمقدار واحد صحيح أما إذا أتت علي الطرفين معا قيمة منطقيـــة ١ فـــان العداد لا يعمل ، بالنسبة لخرج العداد Q طالما كان pv اكبر من الصفر تكون نتيجته on وعندما تساوي الصفر تكون نتيجته off

فلو أخذنا علي سبيل المثال الدائرة الآتية



حيث نجد عند تحول المفتاح I 0.2 يعمل العداد c١ 0 بتهيئته القيمة 55 وعندما يتحـــول المفتاح i0.0

من لوضع off إلى الوضع on يتزايد العداد بمقدار واحد صحيح ما عدا عنـــدما تكـــون القيمة pv تساوي 999 و عندما يتحول المفتاح .I 0 ١ من الوضع off إلـــى الوضـــع on يتناقص العداد بمقدار واحد صحيح ما عدا عندما تكون قيمة pv تساوي الصفر

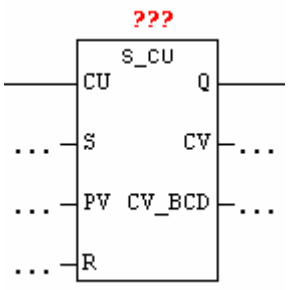
أما إذا تحول وضع المفتاح I 0.3 فان العداد يتوقف عن العمـــل ، والقيمـــة pv تســـاوي الصفر (تصفير العداد من جديد) أما بالنسبة لمخرج العداد Q 4.0 فانه يعطى on طالما كان العداد لا يساوي الصفر ويعطى off عندما يساوي الصقر .

م//

اسم العداد C10.

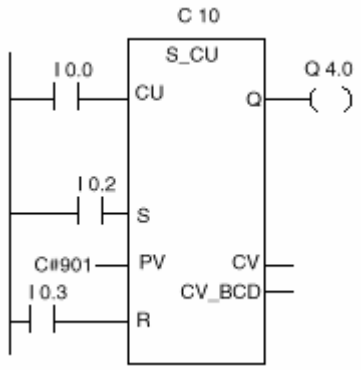
٢ – عداد تصاعدي up counter

الرمز



عند وضع قيمة منطقية علي الطرف s فان العداد يبدأ في العمل بإعطائه القيمة pv وبإعطائه قيمة منطقية علي الطرف R فان العداد يتوقف عن العمل (تصفير العداد) ، وعند وضع قيمة منطقية ١ علي الطرف cu فان العداد يتزايد عن العمل بمقدار واحد ، وتكون نتيجة المخرج Q on في حالة قيمة pv لا تساوي الصفر وتكون نتيجته off في حالة قيمة pv تساوي صفر .

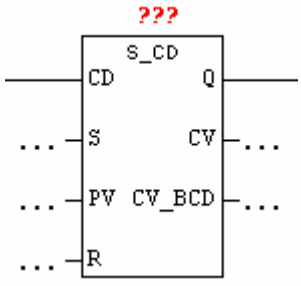
ولو أخذنا علي سبيل المثال الدائرة الآتية



ولتشغيل العداد نحول وضع المفتاح من الوضع off إلى الوضع on حيث نحمل العداد القيمة 901 و قيمة العداد تتزايد بمقدار واحد لغاية عندما يصل القيمة 999 ، لإيقاف العداد نحول وضع المفتاح I 0.3 من الوضع off إلى الوضع on ، وخرج العداد يساوي قيمة منطقية ذ 1 طالما لا يساوي pv الصفر .

٣-عداد تنازلي Down counter

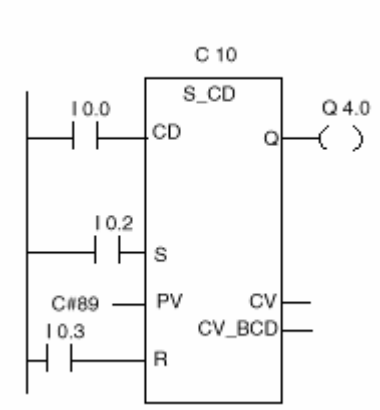
الرمز



عند وضع قيمة منطقية علي الطرف s فان العداد يبدأ في العمل بإعطائه القيمة pv وبإعطائه قيمة منطقية علي الطرف R فان العداد يتوقف عن العمل (تصفير العداد) ، وعند وضع قيمة منطقية ١ علي الطرف cd فان العداد يتناقص عن العمل بمقدار واحد ، وتكون نتيجة

المخرج Q on في حالة قيمة pv لا تساوي الصفر وتكون نتيجته off فـــي حالـــة قيمـــة pv تساوي صفر .

ولو أخذنا علي سبيل المثال الدائرة الآتية



ولتشغيل العداد نحول وضع المفتاح من الوضع off إلى الوضع on حيث نحمل العداد القيمـــة 89 و قيمة العداد تتناقص بمقدار واحد لغاية عندما يصل القيمة الصفر ، لإيقاف العداد نحول وضع المفتاح I 0.3 من الوضع off إلى الوضع on ، وخرج العداد يساوي قيمة منطقيـــة 1 طالما لا يساوي pv الصفر .

المقارنات Comparison instruction

النظام الذي لدينا يدعم ثلاثة أنواع من المقارنات وهي

compare double instruction
compare flouting point numbers
compare integer

وسوف نتناولها بالترتيب

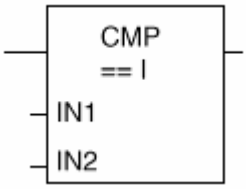
١- compare integer

تعليمه compare integer تعمل علي مقارنة بين عددين صحيحين طول كل عدد 16 خليـــة ، تلك التعليمية تجري عملية مقارنة بين مدخل IN1 و IN2 حسب طبيعة المقارن الذي اختارتـــه فإذا تحقق الدالة أو التعليمية فان نتيجة المقارن تكون القيمة المنطقية 1 ، و الجـــدول التـــالي يبين _____

Types of Comparisons for Integers

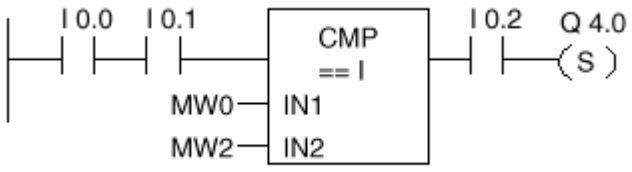
Type of Comparison	Symbols in Name at Top of Box
IN1 is equal to IN2.	==
IN1 is not equal to IN2.	<>
IN1 is greater than IN2.	>
IN1 is less than IN2.	<
IN1 is greater than or equal to IN2.	>=
IN1 is less than or equal to IN2.	<=

الرمز



IN1	القيمة الأولى
IN2	القيمة الثانية

ولتوضيح فكرة هذا المقارن دعنا نري هذا المثال



المخرج Q4.0 يكون في حالة on إذا تحققت الشروط الآتية :-

١- المفتاحين I0.0, I 0.1 في حالة ON

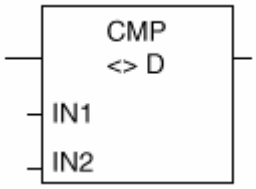
٢- MW0=Mw2 (قيمة المفتاحين في الذاكرة .

٣- المفتاح I 0.2 في حالة on .

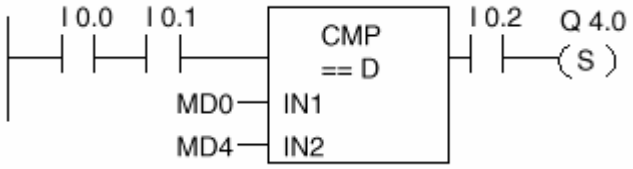
٢- Compare Double Integer

تعليمه compare double integer تعمل علي مقارنة بين عددين صحيحين طول كل عـــدد 32 خلية ، تلك التعليمية تجري عملية مقارنة بين مدخل IN1 و IN2 حسب طبيعة المقارن الـــذي اختارته فإذا تحقق الدالة أو التعليمية فان نتيجة المقارن تكون القيمة المنطقية 1 .

الرمز



و لمزيد من الإيضاح دعنا نري هذا المثال



المخرج Q4.0 يكون في حالة on إذا تحققت الشروط الآتية :-

٤- المفتاحين I0.0, I 0.1 في حالة ON

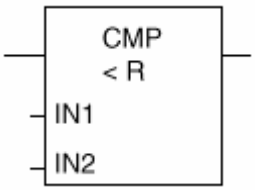
٥- MW0=Mw4 (قيمة المفتاحين في الذاكرة .

٦- المفتاح I 0.2 في حالة on .

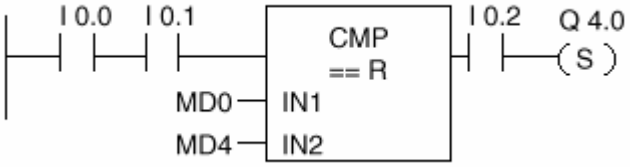
٣-Compare Flouting Point Numbers

تعليمه compare flouting point numbers تعمل علي مقارنة بين عددين حقيقيين ، تلـــك التعليميـــة تجري عملية مقارنة بين مدخل IN1 و IN2 حسب طبيعة المقارن الذي اختارته فـــإذا تحقـــق الدالة أو التعليمية فان نتيجة المقارن تكون القيمة المنطقية 1 .

الرمز



و لمزيد من الإيضاح دعنا نري هذا المثال



المخرج Q4.0 يكون في حالة on إذا تحققت الشروط الآتية :-

٧- المفتاحين I0.0, I 0.1 في حالة ON

٨- MW0=Mw4 (قيمة المفتاحين في الذاكرة .

٩- المفتاح I 0.2 في حالة on .