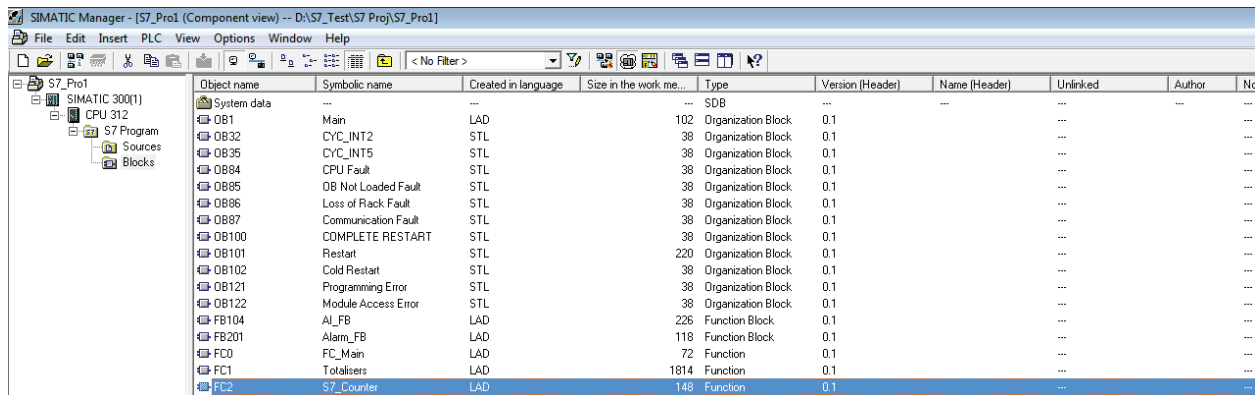


Following is the UP-Counter logic, step by step explained with significance of each counter input.

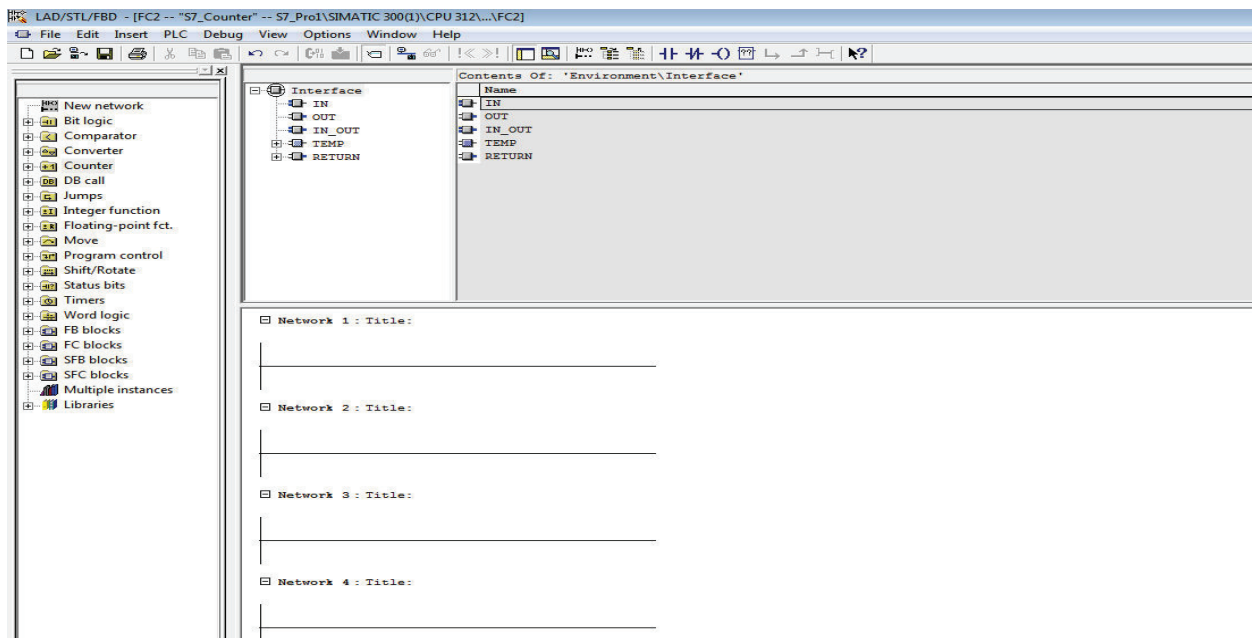
A) How to Insert Counter, explained in point 1,2.

1) How to drag counter in program?

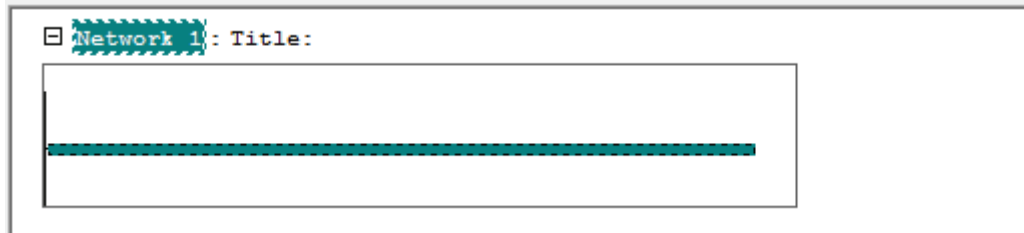
First open Simatic Manager software as shown below



Then double click on FC2 (Function Block FC2) and then following file appears



then double click on the line of network1 as shown below



After this goto left side of FC2 where it shows various features or instruction folder such as Bit Logic, Comparator etc. Exactly click on folder counter and it will show number of counter types as shown below and then double click on S_CU (UP Counter), this counter will appear with ??? over it, on network 1 as shown below.

The screenshot shows the SIMATIC Manager interface for a function block (FC2) named "S7_Counter". The left-hand pane displays a tree view of instruction folders, with the "Counter" folder expanded to show sub-folders: S_CUD, S_CU, S_CD, --(SC), --(CU), and --(CD). The "S_CU" folder is highlighted with a red box, and a red "???" is placed above it, indicating it has been selected for insertion into the network.

The right-hand pane shows the "Contents Of: 'Environment\Interface'" table, which lists the following variables:

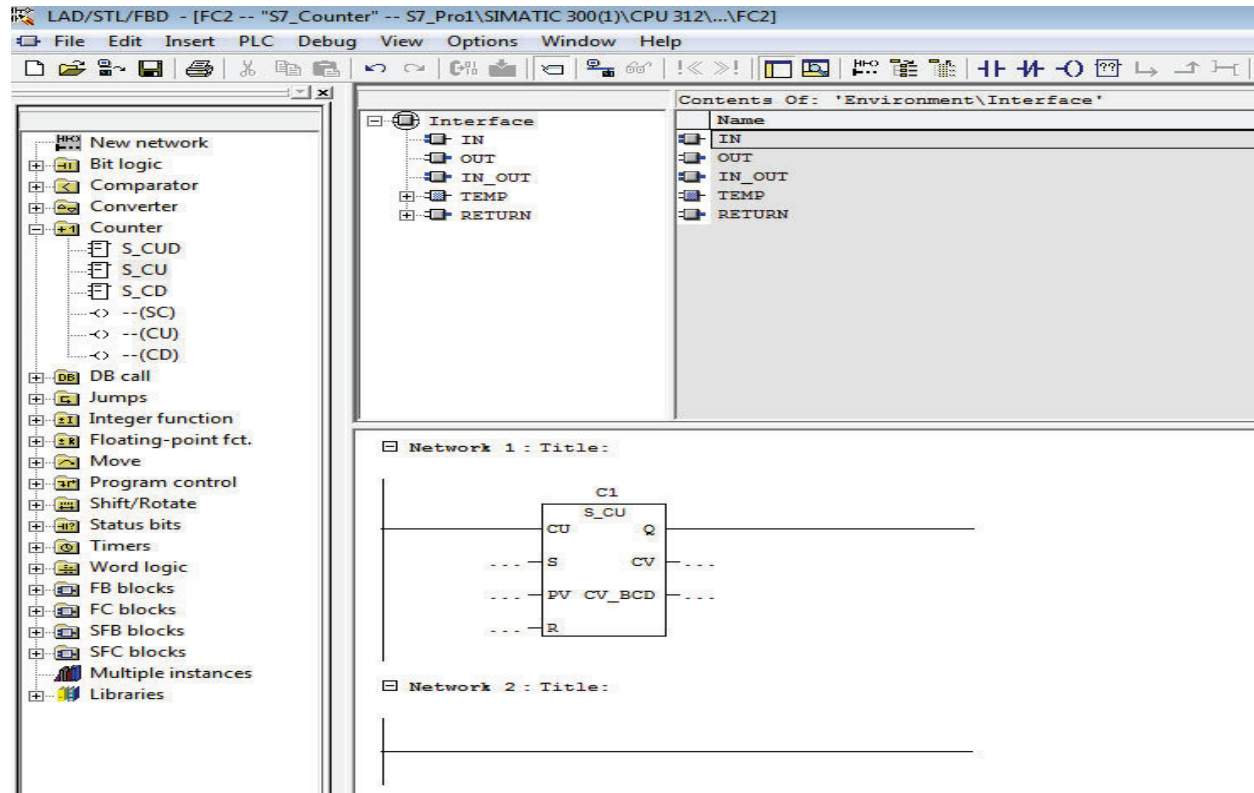
Name
IN
OUT
IN_OUT
TEMP
RETURN

The main workspace shows the ladder logic for "Network 1: Title:". The network contains a single coil (S_CU) with a red "???" above it. The coil has four terminals: CU (top), Q (right), S (left), and CV (bottom). The CV terminal is connected to a variable named "CV_BCD". The network rail is shown as a horizontal line on the left.

Below Network 1, "Network 2: Title:" is shown as an empty network with a horizontal line on the left.

2) How to assign number to Counter?

Then click on ??? type C1 and then this counter is assigned with name or tag C1 as shown below.



B) Load PV from HMI.

MW10, MW12, MW14 are internal memory words used to store PV's of Counter that is entered, as shown below it appears in logic.

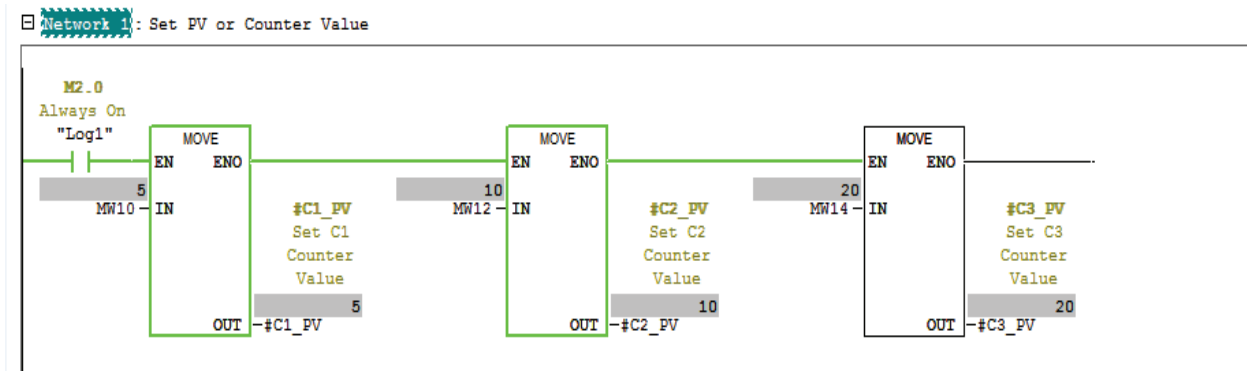
These memory words are counter values entered from HMI, the above memory words are configured in HMI to satisfy required operation in PLC. HMI means Human Machine Interface where user can see and edit values. Here PV is in word format. PLC has Boolean (i.e. 0 and 1), byte (0 to 7), Integer/Word (0 to 15), Real (0 to 31), Double Integer/Double Word (0 to 31) etc.

Note: You cannot use same memory words or used ones for other inputs or to enter values or store values because it will overlap.

1) Value of HMI internal memory word MW10 is copied to C1_PV (Local Memory LW0) from HMI, currently 5.

2) Value of HMI internal memory word MW12 is copied to C2_PV (Local Memory LW2) from HMI, currently 10.

3) Value of HMI internal memory word MW14 is copied to C3_PV (Local Memory LW4) from HMI, currently 20.

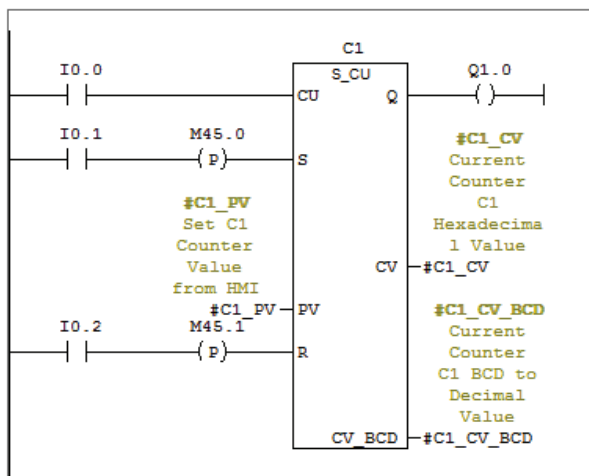


C) Counter IO's and its functions:

1) **Input CU:** When I0.0 as shown gets a rising edge i.e. turns from 0-->1 the counter C1 is incremented by one as it is UP counter, if the signal state (I0.0) at input CU changes from "0" to "1" and the value of the counter is less than "999".

If the counter is set (means incremented on rising edge) and if RLO = 1 (if I0.0 is high after 0 to 1 transition) at the inputs CU, the counter will count once in the next scan cycle (i.e. if there is 0 to 1 transition again, otherwise not) even if there was no change from a positive to a negative edge or vice versa. (RLO is Result of Logic Operation. RLO is 1 when conditions [(in above e.g. real world input I0.0 is On) are satisfied and RLO is 0 (in above e.g. real world input I0.0 is Low) when not satisfied, explained in point#8].

Here the counter has inbuilt function to detect rising edge [(P) it's function is explained in point#9] and used for S (Preset) and R (Reset).

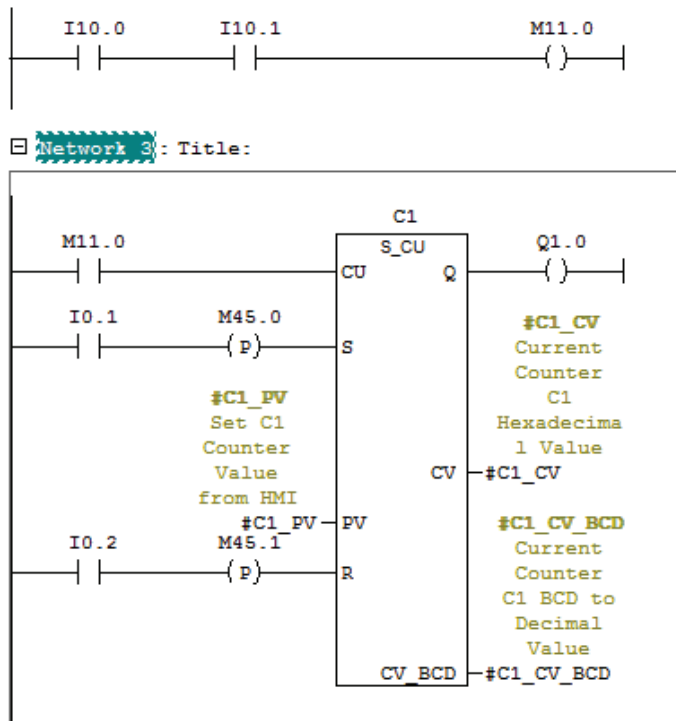


SIEMENS

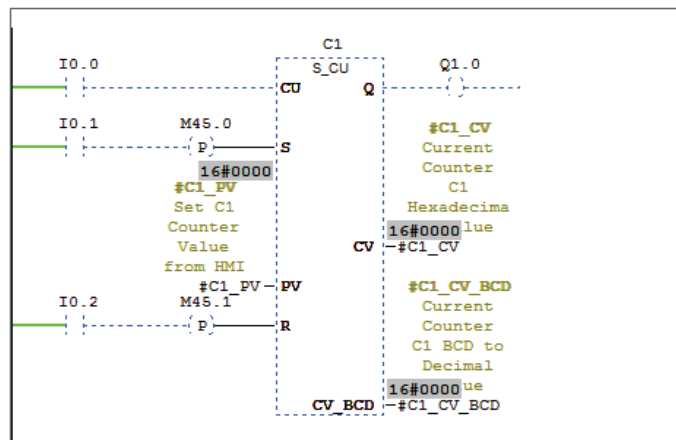
Instead of real world input I0.0 you can also assign internal memory bit (M11.0) as shown below, which will be ON/OFF when multiple conditions are satisfied for count operation.

In below example, internal memory bit is ON when real world inputs I10.0 and I10.1 both are ON and then counter will increment accordingly otherwise not, this is explained for your clear understanding.

These (I) real world inputs are hardwired digital inputs of system or PLC used to explain behavior here for counter function understanding.

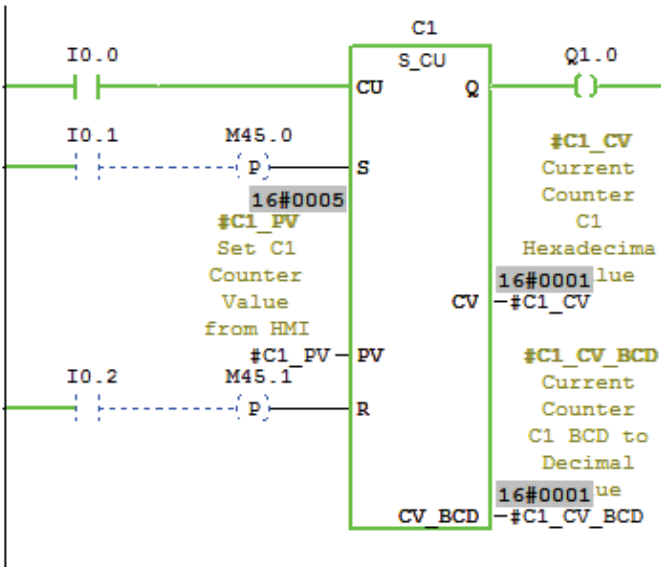


In following diagram, initially counter is zero,



SIEMENS

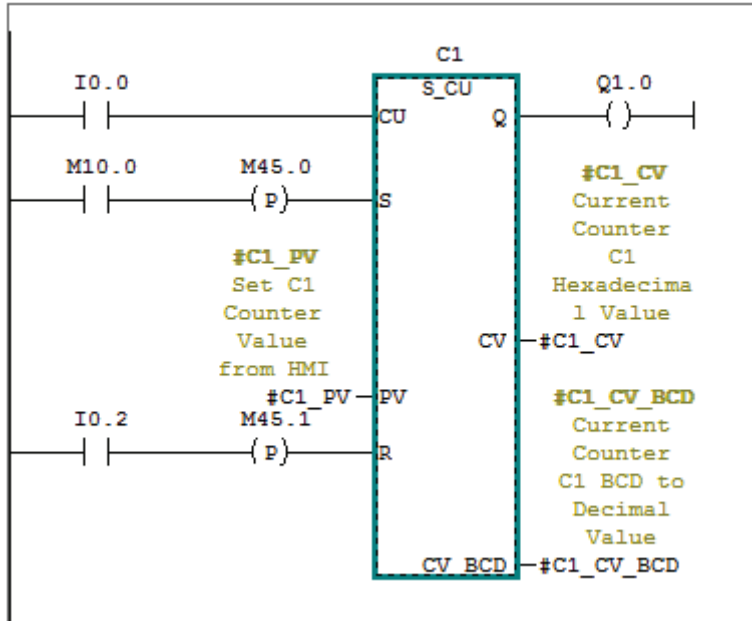
Now when I0.0 is ON count turns to 1 as shown in below diagram at CV (C1_CV), CV_BCD (C1_CV_BCD) and increments when I0.0 turns ON every time. Here input CU serves the function of rising edge and hence no rising edge (P) is required to be used after I0.0.



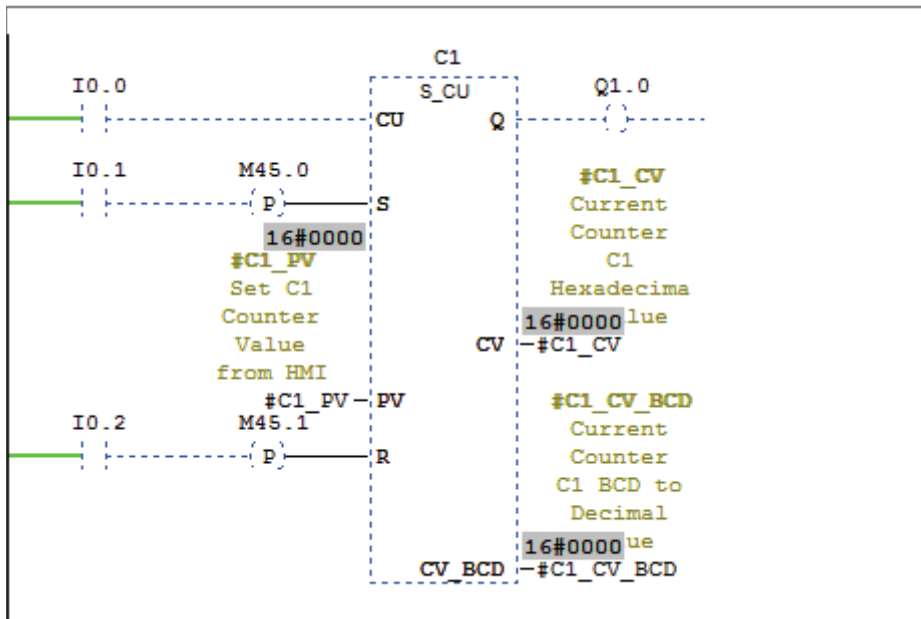
2) Input S: Counter is preset with the value at input PV if there is a positive edge at input S i.e. when there is rising i.e. I0.1 (as shown above) turns from 0-->1. It is not necessary every time to use. Here the internal memory bit M45.0 is used to detect rising edge (it's function is explained in point#9).

Instead of real world Input I0.1 you can assign internal memory bit (M10.0) as shown below to preset counter from HMI, this internal memory bit has to be configured from HMI then only it will serve the required action. This facility from HMI is given to user/operator whenever he enters new counter value from HMI.

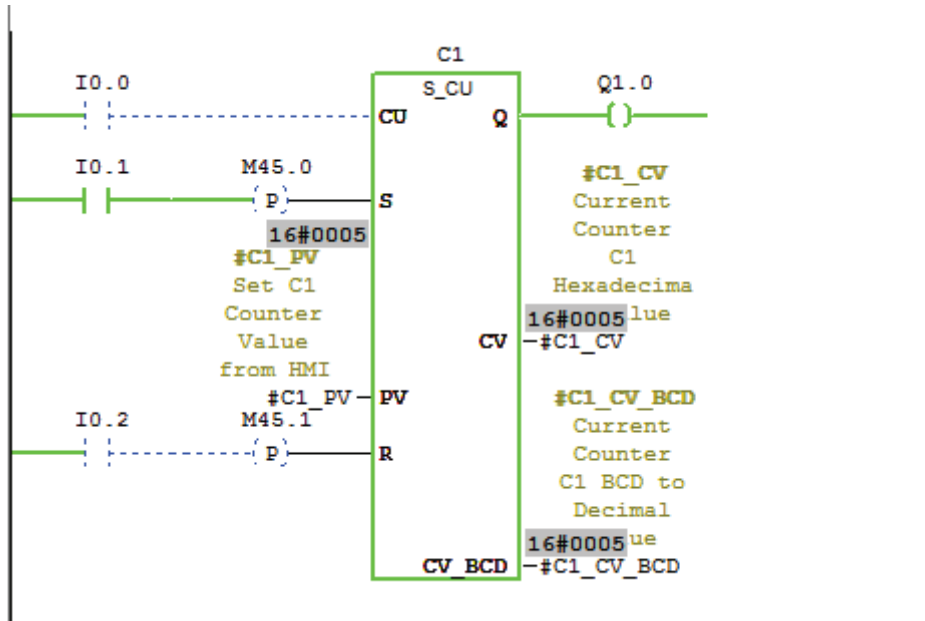
SIEMENS



Following diagram shows before I0.1 is ON counter looks in blue dotted lines before preset i.e. before I0.1 is turned ON as shown below (rung conditions not met, rung false, RLO=0).



Now when I0.1 turns ON (rung conditions are met, rung true, RLO=1). Counter is preset showing value 5 or 16#5 on PV (or C1_PV gets assigned with 5) and appears green as shown below:

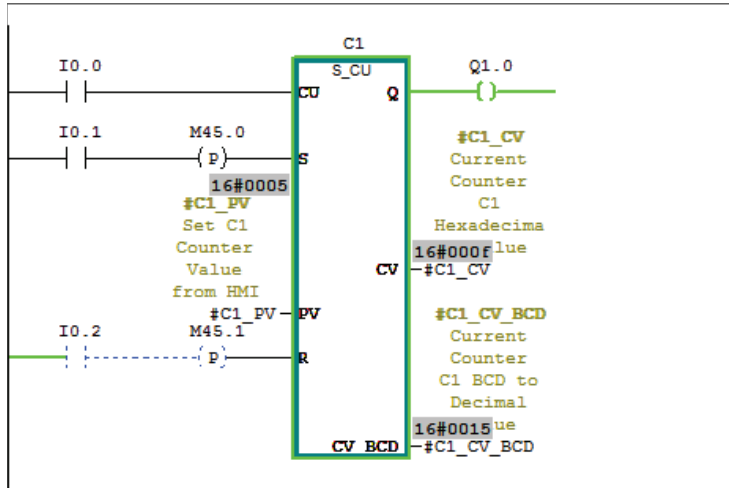


3) Input R: Whenever there is a positive edge at input R (i.e. when I0.2 turns from 0-->1) counter is reset at any condition, i.e. whether the count is done (equals PV) or not.

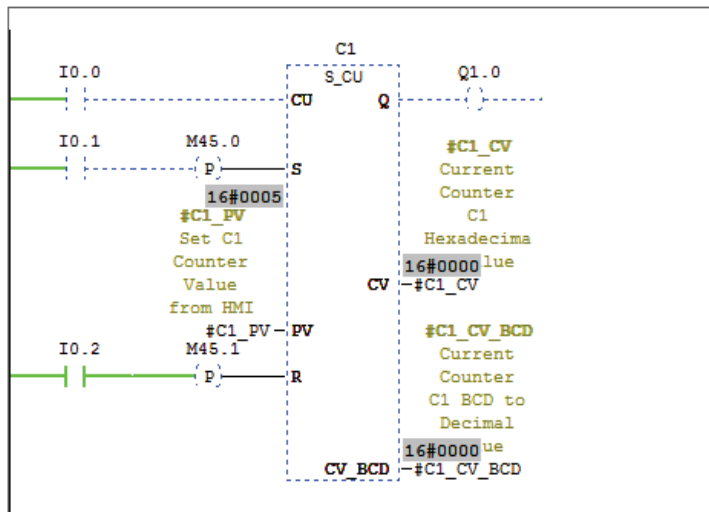
You can assign digital input or internal memory bit from HMI to reset counter, like internal memory M11.0 is used for S_CU and it should be used either in logic or configured from HMI depending upon the condition of operation. Here input I0.2 is used to understand the function. Also, here the internal memory bit M45.1 is used to detect rising edge (it's function is explained in point#9).

Following diagrams will clear your understanding.

3.1) In this diagram counter value is 15 (or 16#000f in hexadecimal and 16#0015 in BCD) before reset.

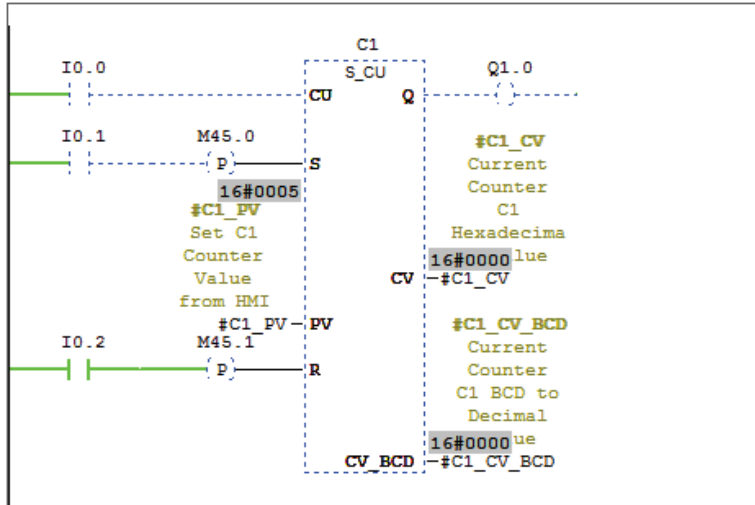


3.2) See after turning on real world input I0.2 counter is reset and appears in blue dotted line i.e. it's value is zero at and CV_BCD as shown in diagram:

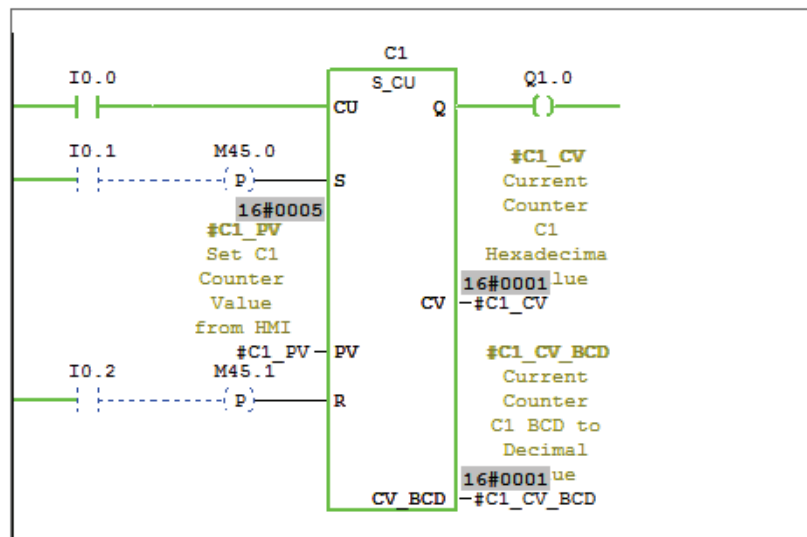


4) **Output Q:** Q1.0 represents Status of the counter. The signal state at output (Q) Q1.0 is "1" if the count is greater than zero and "0" if the count is equal to zero.

Q1.0 is 0 i.e. it's coil appears in dotted blue lines as counter has zero count or reset as shown below.



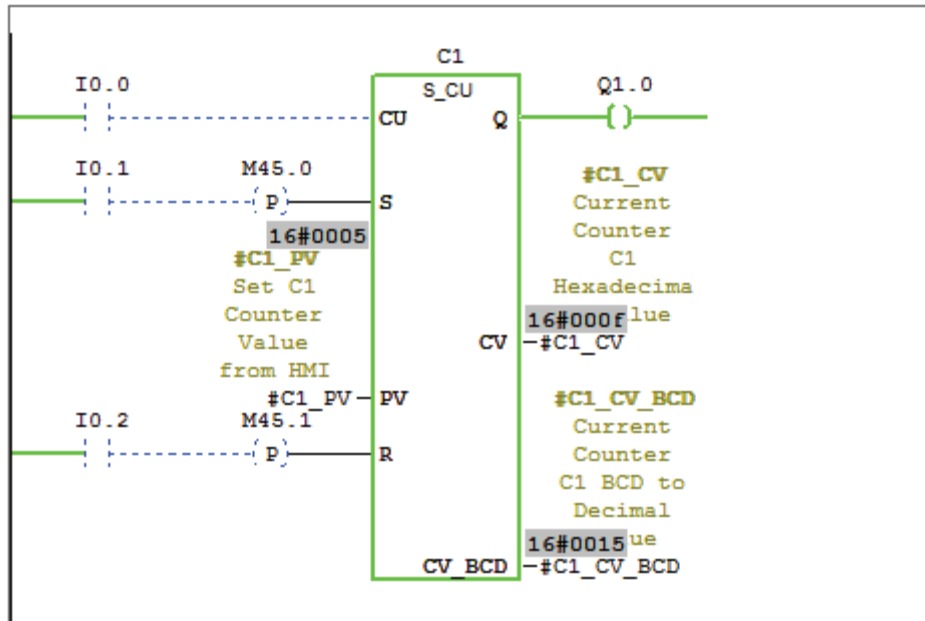
When counter is having value other than zero its coil appears in green bold line as shown below, here counter has value 1 i.e. is Q1.0 is 1.



5) Function: If there is rising edge i.e. I0.1 changes from "0" to "1", the counter is preset with the value of C1_PV local memory. If there is rising edge i.e. I0.0 changes from "0" to "1", the value of counter C1 will be incremented by one - unless the value of C1 is equal to "999". Q1.0 is "1" if C1 is not equal to zero.

6) Counter Hexadecimal Value: Current counter value read by local memory "C1_CV", hexadecimal number. The hexadecimal number system is as follows,

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A (10), B (11), C (12), D (13), E (14), F (15). It can be understood from below diagram:

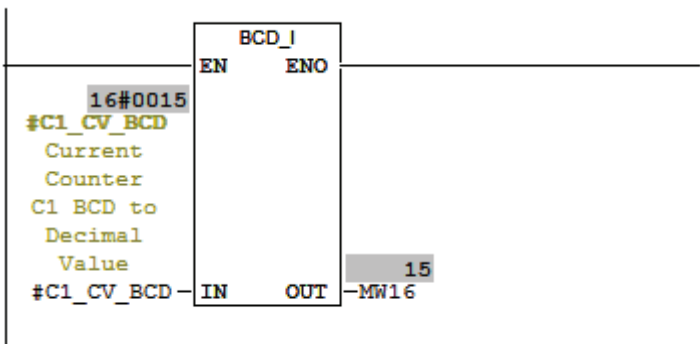


If you see at CV read by local memory C1_CV it is showing 16#000f i.e. count is 15 and at point#7 diagram count is showing same 15 in decimal (BDC To Decimal converted value). Hexadecimal number system lies from 0 to f (explained above), like decimal number system lies from 0 to 9.

7) **Counter BCD Value:** Current counter value read by local memory C1_CV_BCD, is BCD coded. The BCD number system is as follows, 16#0, 16#1, 16#2 likewise. It is converted into decimal using BCD To Integer (I) instruction as shown below:

Network 3 : CV_BCD to Decimal

MW16 has the decimal value and can be tagged into HMI to display counter value.



This value is shown on HMI (Human Machine Interface) for a user to understand count value, read in HMI memory word MW16. By this you can count parts (like bottles, capsules etc.), scrap, faults, etc. Which can be shown on HMI. The internal memory word MW16 must be configured in HMI so that user can view value in HMI.

SIEMENS

8) RLO Result of Logic Operation (Status Word, Bit 1)

The RLO bit stores the result of a logic operation string or comparison instruction.

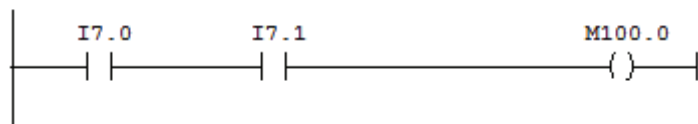
The RLO bit is status word bit 1.

Status word

BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC	
Writes:-	-	-	-	-	-	0	X	X	1

RLO can be understood from following example,

▣ Network 1: RLO Operation



In above logic when real inputs I7.0 and I7.1 both are ON then it means RLO is 1 and both or if either I7.0 or I7.1 is off then RLO is 0.

9) Rising Edge:

---(P)--- (Positive RLO Edge Detection) detects a signal change in the address from 0 to 1.

Falling Edge:

---(N)--- (Negative RLO Edge Detection) detects a signal change in the address from 1 to 0.