# PLC Lab 3: Counter Applications

## Objectives

* Create, test, and understand Counters in TIA Portal
* Program Applications involving Counters
* Program Applications involving Counters and Timers
* Learn to use SFC as a Flowchart to guide programming in ladder

## Tasks

* Task 1: Enabling System and Clock Memory
* Task 2: Using the UP Counter CTU
* Task 3: Resetting Counters at Startup and Initialising Preset Value
* Task 4: CTD Applications: Vending Machine (SDL)
* Task 5: CTUD Applications: Carpark (SDL)
* Task 6: Conveyor drink cans packaging, SFC
* Task 7: T/C Applications: Departmental Store (SDL)

## Synopsis

Counters are fundamental in automation where objects such as parts, vehicles or even persons can be counted using sensors, and PLC react accordingly based on the counts recorded. A conveyor may move/stop. A valve may open/close. A barrier may be raise/lowered based on this. Counters and Timers are very useful elements in PLC programming.

## Equipment Required

* PLC training kit with control panel.
* Laptop/PC with internet access
* Laptop/PC installed with TIA Portal V15.1 and PLCSIM

## Reference and Self Study materials

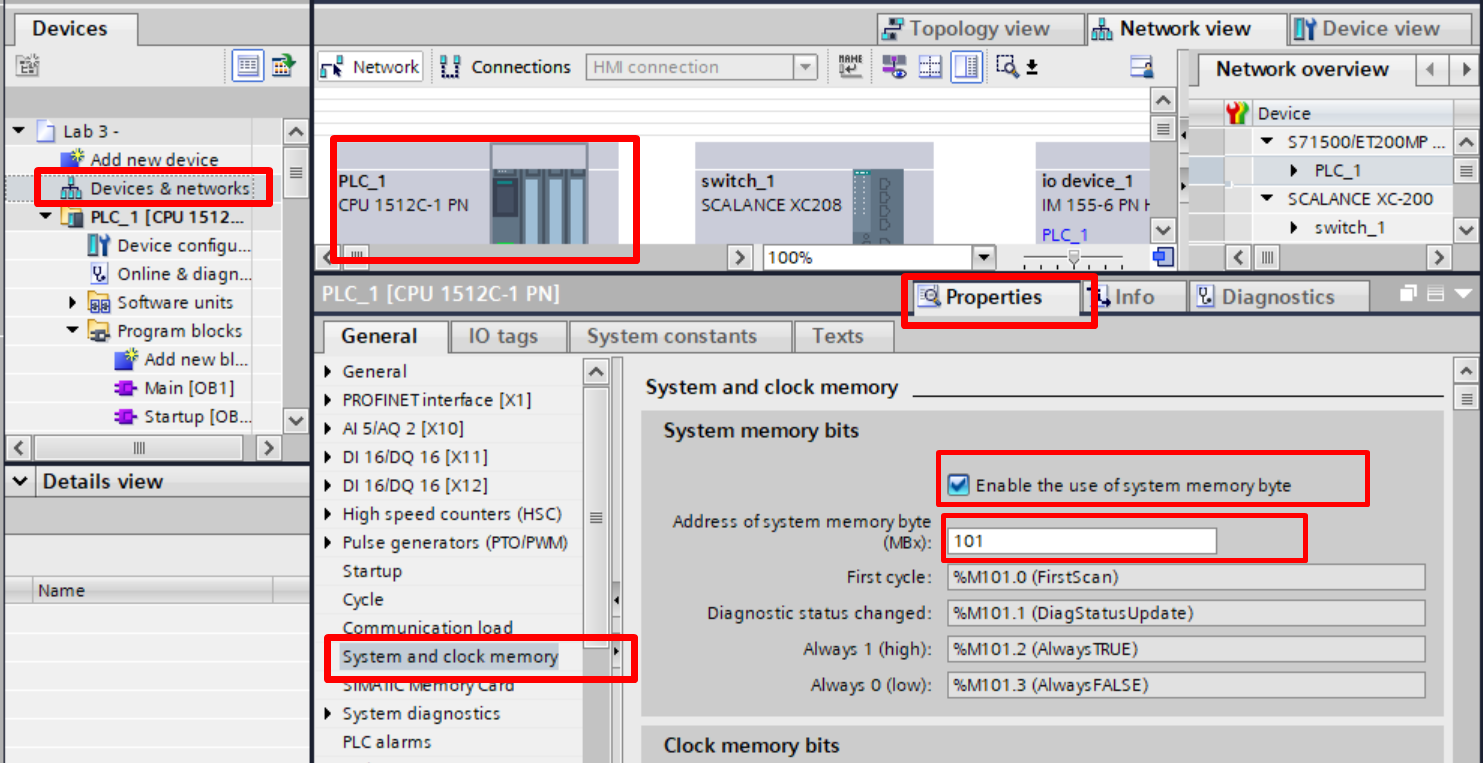
* Lab 03 playlist (Same as corner QR code)  
  <https://www.youtube.com/playlist?list=PLo5IISMe0m5N4FnnU8FQNwLeEQo6FmdUy>
* TIA Portal help files (Access from TIA Portal)
* 0XX-600 Complete SCE Training Curriculum for S7-1500  
  <https://www.automation.siemens.com/sce-static/learning-training-documents/tia-portal/summary-sce-training-curriculum-s7-1500-en.pdf>

## NOTE: In some examples, Buttons/LEDs uses %M. Feel free to change them to %I and %Q if you are using actual lab equipment.

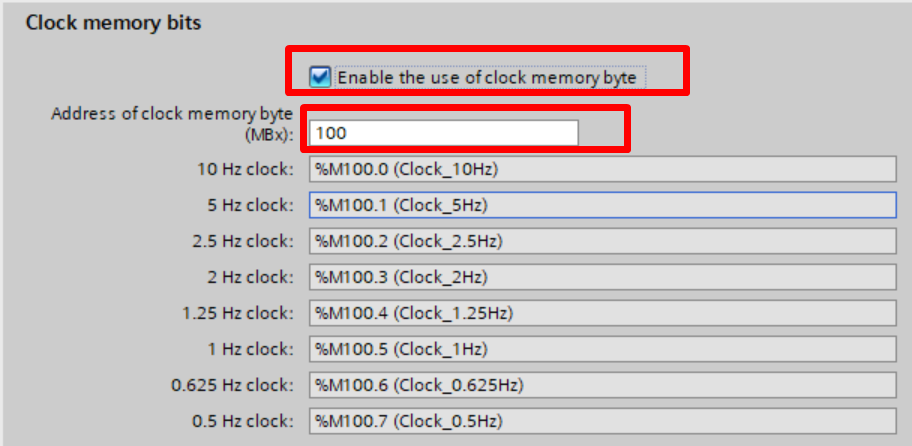
## HBL TIP: At home, change physical GreenButton %**I**126.7 to %**M**126.7. Modifying %M is easier than forcing %I. Use Ctrl+F2, Ctrl+F3 to toggle between ON/OFF for maximum efficiency.

## Task 1: Enabling System and Clock Memory

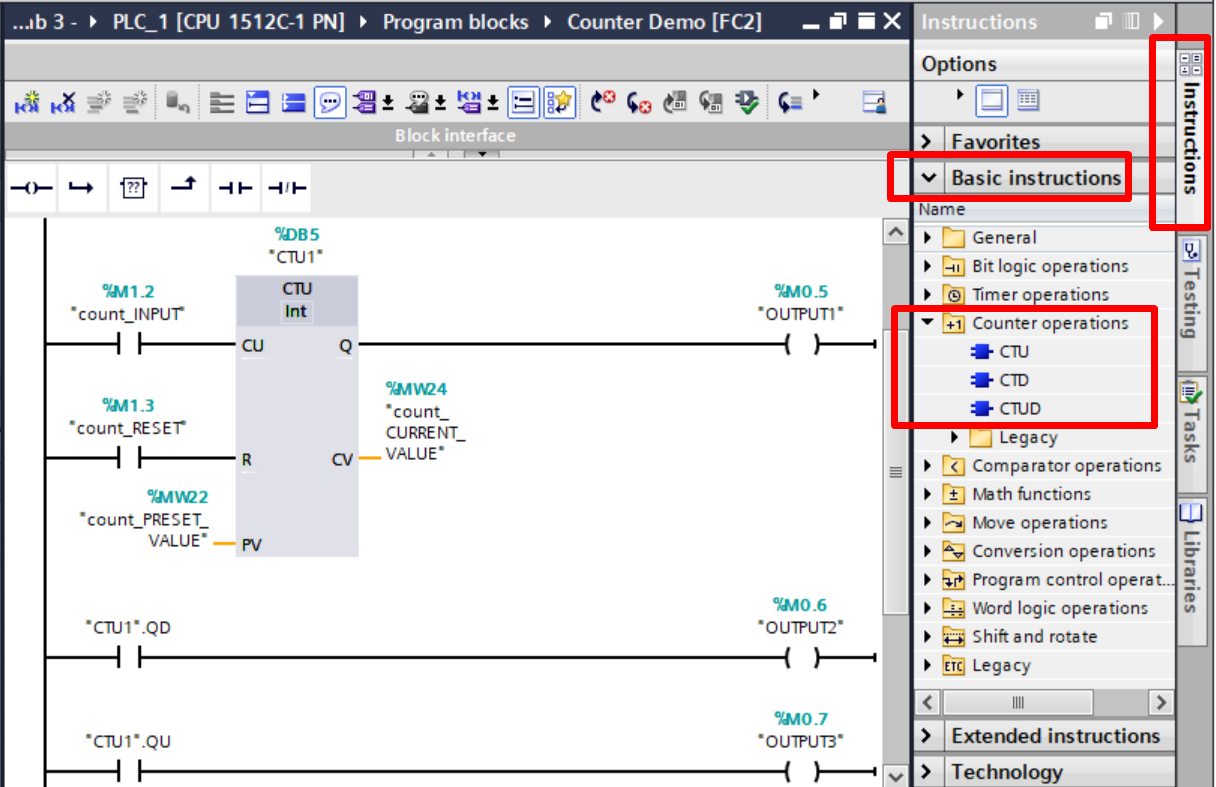
1. Follow screenshot below: Devices&Networks>PLC>Properties>System&ClockMemory>  
   Important: The default memory mapping are %MB1 and %MB0. They can be remapped. Try changing them to %MB101 and %MB100. Tick the checkboxes.



Tick the boxes & change the address



## Task 2: Using the UP Counter CTU

1. Watch the video to learn about using the UP counter CTU  
   <https://youtu.be/VxRlkyVRfmk>
2. Construct the following. Set “count\_Preset\_Value” as 5. (Right-click>ModifyOperand). Observe what happens to CV and Q as you toggle count\_INPUT 6 times.
3. Fill in the table for the CTU block

|  |  |  |
| --- | --- | --- |
| Label | Data Type | Description |
|  | Bool | Counter input |
|  | Bool | Counter reset |
|  |  | Preset Value |
| Q |  | Counter output |
|  |  | Current Value |

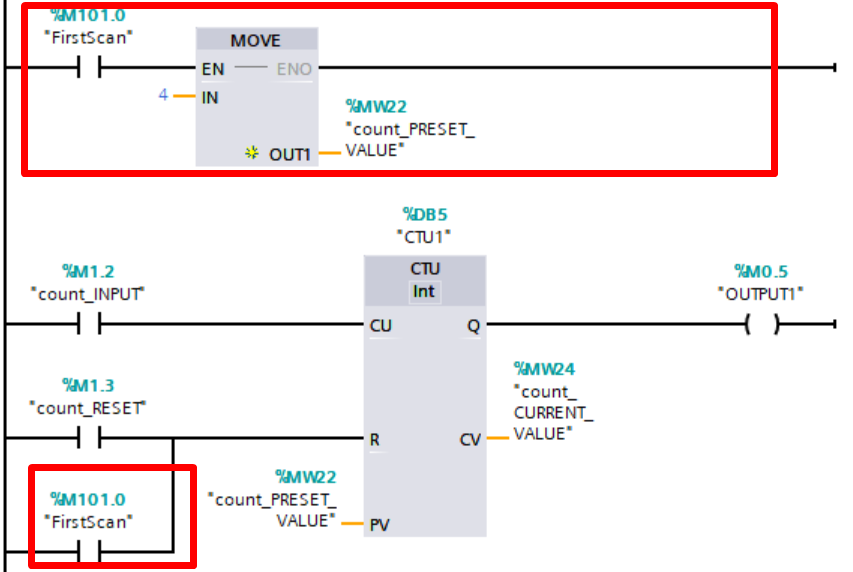
1. What is the difference between CTU1.QD and CTU1.QU?

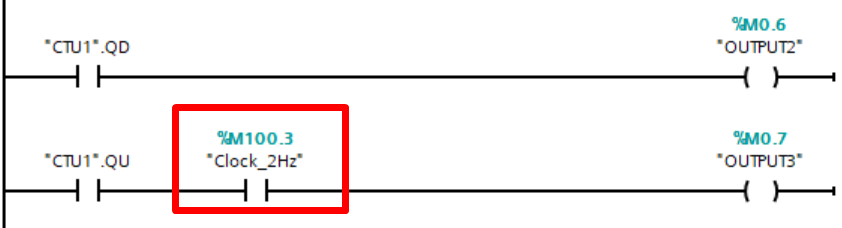
|  |
| --- |
|  |

## Task 3: Resetting Counters at Startup and Initialising Preset Value

1. Run the counter program. Keep your eyes on PV and CV values. Stop the CPU and Run the CPU again. What do you observe changed?

|  |
| --- |
| *CV:* |
| *PV:* |

1. Every time we reboot the CPU, non-retentive memory is lost. Counters may lose its PV if the memory storing it is non-retentive. It is therefore necessary to initialise it during startup. The CV of counters is retentive by default. There, if we want to clear it, we have to create a means to do so during startup. Using “FirstScan” flag is one method.
2. Modify the program as follows:  
   



1. Download and test the program. Trigger count\_INPUT a few times. Stop the CPU and Run the CPU again. What changes do you observe?

|  |
| --- |
| *CV:* |
| *PV:* |

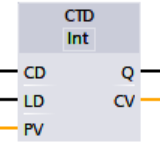
1. What is the role of M100.3 in the program?

|  |
| --- |
|  |

1. You have observe the initialisation of a counter. There are other ways to do it in TIA portal, but this basic method will probably work for all brands of PLC software. Remember that as a programmer, you will decide what to do with PV and CV. Do not leave it to chance.
2. **Optional: An alternate means**. OB100 is an organization block that only run once when the PLC start up, or goes from STOP to RUN mode. We can also use this to initialise the counter. <https://youtu.be/eCTsiXbs-18>. Watch from 2:37

## Task 4: CTD Applications: Vending Machine (SDL)

Down Counters (CTU) and UP/DOWN counters (CTUD) are commonly used counters. Explore and understand their operations through creating simple programs

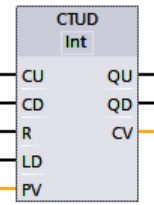
1. Create a CTD application. A vending machine starts of with 10 cans of drinks. Once a purchase is made using the PURCHASE\_BUTTON, the can of drinks reduces by 1 count. An EMPTY\_LED will light up when there are no more drinks. A TOPUP\_BUTTON will be press whenever the worker tops up the vending machine.

|  |  |  |
| --- | --- | --- |
| **PLC Tag (Symbol)** | **Type** | Data **Type** |
| PURCHASE\_BUTTON | Input | Bool |
| TOPUP \_BUTTON | Input | Bool |
| EMPTY\_LED | Output | Bool |

1. What is the function of the LD input in CTD?

|  |
| --- |
|  |
|  |

## Task 5: CTUD Applications: Carpark (SDL)

1. Create a CTUD application. There are 20 parking lots in a Carpark. Car goes in through the ENTRANCE and leaves through the EXIT. When the Carpark is full, a FULL\_LED lights up.

|  |  |  |
| --- | --- | --- |
| **PLC Tag (Symbol)** | **Type** | Data **Type** |
| ENTRANCE\_SENSOR | Input | Bool |
| EXIT\_SENSOR | Input | Bool |
| FULL\_LED | Output | Bool |

1. Present you ladder program below

|  |
| --- |
| Insert your ladder diagram here |

1. Imagine that you have just finished programming and installed the PLC parking system. Your CV is 0 because your system is brand new. However, there are already 12 cars in the carpark. What can you do?

|  |
| --- |
| Method1: Manually trigger the Entrance Sensor until CV is 12. |
| *Method2:* |
|  |

## Task 6: Conveyor drink cans packaging, SFC

1. A SENSOR detects can drinks on a CONVEYOR. For every 6 can drinks counted, the CONVEYOR stops to allow the worker to pack the can drinks into a box. When the worker is done, he will press an ACKNOWLEDGE button to resume the CONVEYOR operation.
2. Write your program using the Tags and SFC

|  |  |
| --- | --- |
| *Type* | *Symbol* |
| *Bool Input* | *SENSOR* |
| *Boolean Output* | *CONVEYOR* |
| *Boolean Input* | *ACKNOWLEDGE* |
| *Counter* | *CTU1* |

Present your ladder below

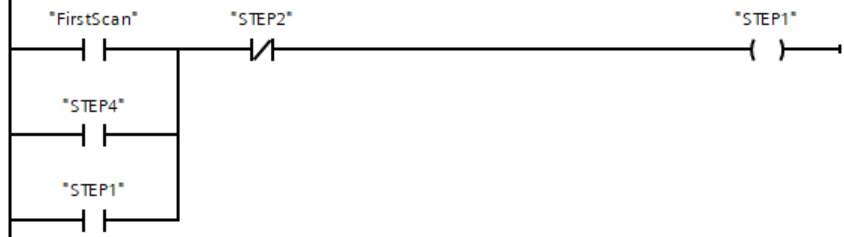
1. Spoiler. This contain the answer. Skip if you want to try on your own first.  
   How to do it in a structured way?   
   How to do it fast?   
   How to get into Step1?

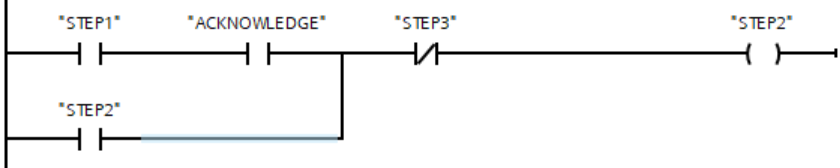
How to loop back to Step1 at the end of Step4?

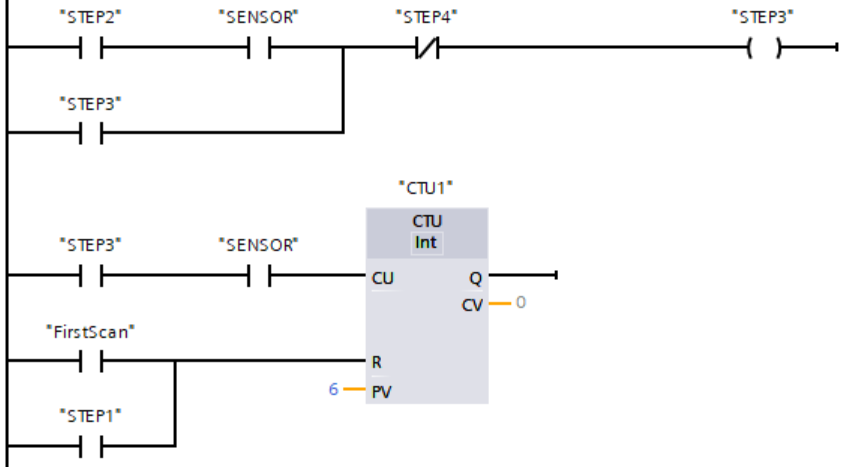
How to initialize and reset your counter?

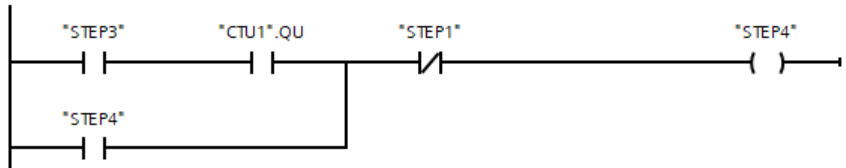
How to verify your program, clearly, easily?

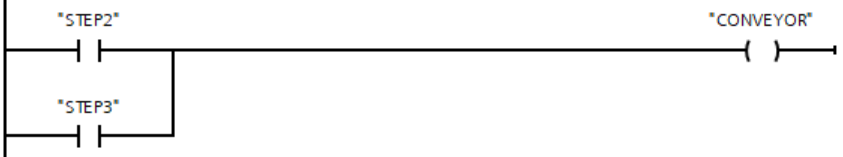
If you try and can’t do it or understand it, view this. <https://youtu.be/uwuDephOiOs>











## Task 7: T/C Applications: Departmental Store (SDL)

In many instances, timers and counters work together.

1. To help to liven up a departmental store, for every 5 customers who pass through the ENTRANCE, a welcome MELODY will play for 10 seconds. Implement this in PLC ladder.

|  |  |  |
| --- | --- | --- |
| **PLC Tag (Symbol)** | **Type** | Data **Type** |
| ENTRANCE\_SENSOR | Input | Bool |
| MELODY\_SPEAKER | Output | Bool |

1. Test your ladder. Does the MELODY\_SPEAKER plays when the 5th customer enters?
2. Does your MELODY\_SPEAKER plays when the 10th and 15th customer enters?
3. Create the program and insert your program below

|  |
| --- |
| Insert your ladder diagram here |

## Take Away

1. **What is FirstScan Flag?**  
   During startup, the FirstScan flag simply gives a short pulse (duration of exactly one scan cycle). We can use this flag to perform initialisation for counter values, etc.   
     
   **What is a clock memory bit?**  
   Example is a 2Hz clock pulse flag. We might conveniently use it to blink our LEDs without writing additional program using timers.   
     
   **How do we enable these?**PLC>DeviceView>Properties>System&ClockMemory>  
   Check the box: “Enable the use of system memory byte”&   
   Check the box: “Enable the use of clock memory byte”   
   Beware the default bytes are %MB1 and %MB0. (If they clash with your existing PLC tags, you need to take care of it)

**Counter have PV, R, LD. Initialize them.**Preset values may be fixed constant, or variable. Variables may be lost upon power cycle. They need to be initialize.   
CTU needs to be Reset(R) so CV=0.   
CTD needs to be Loaded(LD), so that the CV=PV for counting down purpose.  
These will need the FirstScan flag.

**Sequential Function Chart SFC**SFC can be used as flowchart to guide programming.  
SFC can be converted to ladder in a straightforward manner.

**There are certain tips to do it easily, refer to task6:item20**