# PLC Lab 2: Timer Applications, Function(FC)

## Objectives

* Create, test, and understand Timers in TIA Portal
* Interpret and Analyse Timing Diagrams
* Program Applications involving Timers

## Tasks

* Task 1: Using ON Delay Timer TON in TIA Portal
* Task 2: Using OFF Delay Timer TOF, Pulse Timer TOP, Retentive or Accumulate Timer TONR
* Task 3: Program Dual Timer Applications (SDL)
* Task 4: Use of Function (FC) in TIA Portal
* Task 5: Application: Conveyor Loading Unloading
* Task 6: The Blinking Light challenge (SDL)
* Task 7: The Blinking Light challenge Part 2 (SDL)
* Task 8: The Traffic Light Challenge Part 3 (SDL)

## Synopsis

Timers are fundamental in any programming language. Each language and programming tool will clearly define how these are used. In this lab we will examine how the different types of timers available in TIA Portal and learn how to use them effectively to create applications. Function(FC) are sub programs which can be called upon by other programs. We will also learn to create Function(FC) and pass values into them for processing.

## Equipment Required

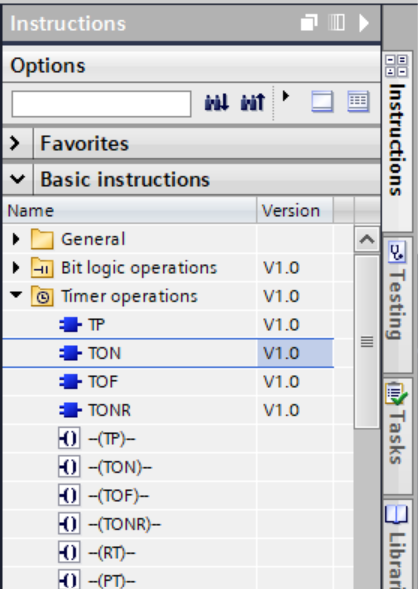
* PLC training kit with control panel.
* Laptop/PC with internet access
* Laptop/PC installed with TIA Portal V15.1 and PLCSIM

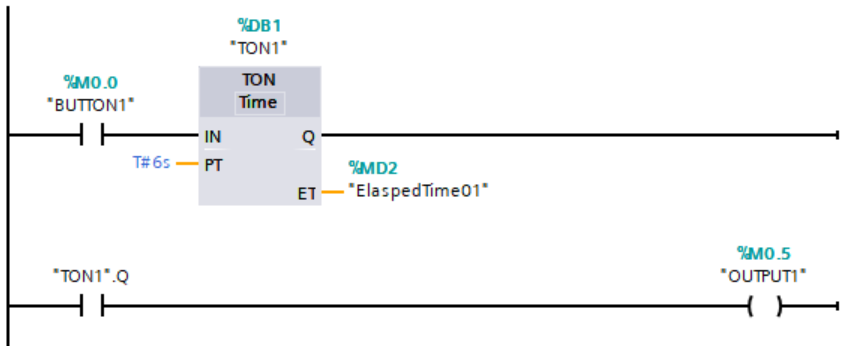
## Reference and Self Study materials

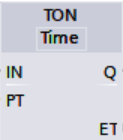
* Lab 02 playlist (Same as corner QR code)  
  <https://www.youtube.com/playlist?list=PLo5IISMe0m5OpwX4hqcW41DfEKSJ3yaIA>
* TIA Portal help files (Access from TIA Portal)
* 0XX-600 Complete SCE Training Curriculum for S7-1500  
  <https://www.automation.siemens.com/sce-static/learning-training-documents/tia-portal/summary-sce-training-curriculum-s7-1500-en.pdf>

## Task 1: Using ON Delay Timer TON in TIA Portal

1. ON Delay Timer TON. When the Timer Input(IN) is on, the Elasped Time(ET) increases. When the Elasped Time reaches Preset Time(PT), Output(Q) turns on.
2. Program the following. Obtain TON instruction from the right side panel, Instructions>BasicInstructions>TimerOperations.

(You may watch this video to learn to use TON <https://youtu.be/gom-14T3AQk>)

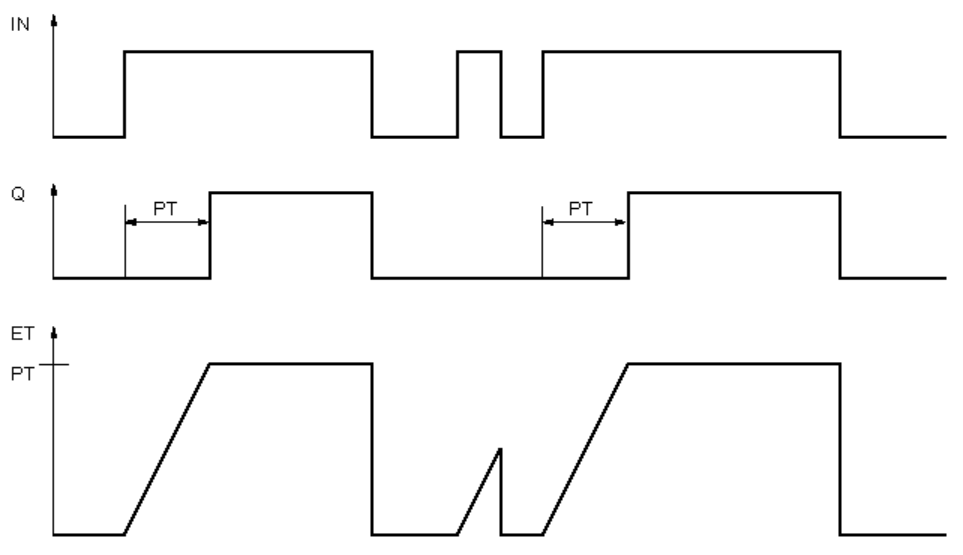


1. Get to know the TON parameters

|  |  |  |
| --- | --- | --- |
| Label | Description | Data Type |
|  | Timer Input | Bool |
|  | Preset Time | Time |
|  | Timer Output | Bool |
|  | Elasped Time | Time |

* The name of the timer above is \_\_\_\_\_\_\_\_\_.You can choose any other, eg. Timer04
* The output TON1.Q is the Timer Output bit. (format is Name.Q)
* T#1h5m8s is the format of the preset time to follow. (T#8s means 8sec)

1. Study the Timing Diagram of the ON Delay Timer TON.



Test your program. With PT set to T#6s, activate BUTTON1 for 15sec.

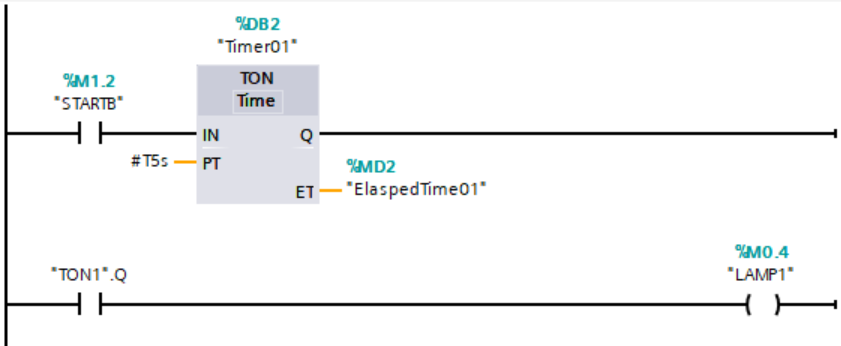
|  |
| --- |
| *Observation:* |
|  |

1. Activate BUTTON1 for 3sec. Explain why.

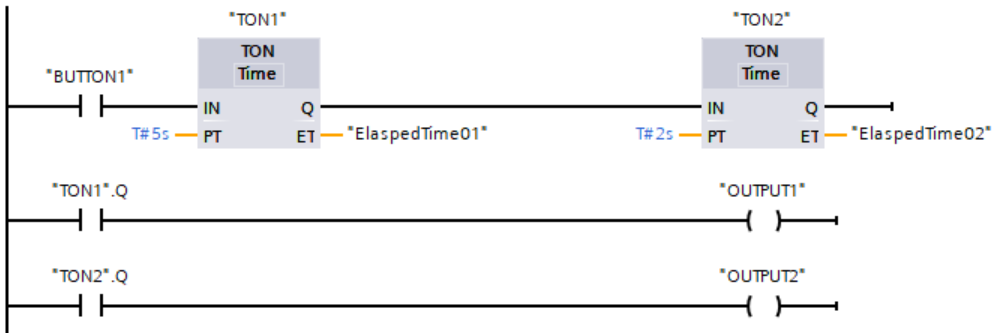
|  |
| --- |
| *Observation:* |
|  |

1. There are 2 errors in the ladder below that prevents it from working properly. Correct them.

|  |
| --- |
| *Error1:* |
| *Error2:* |

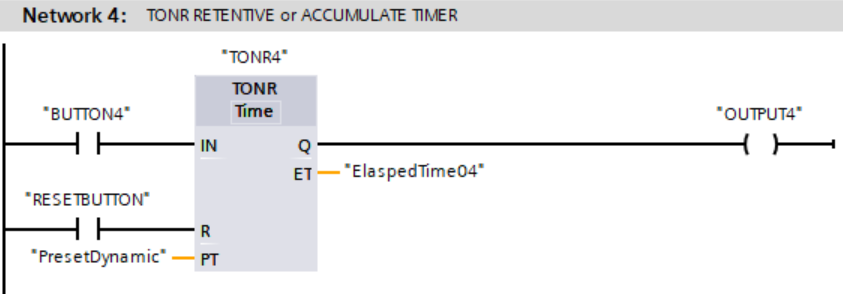
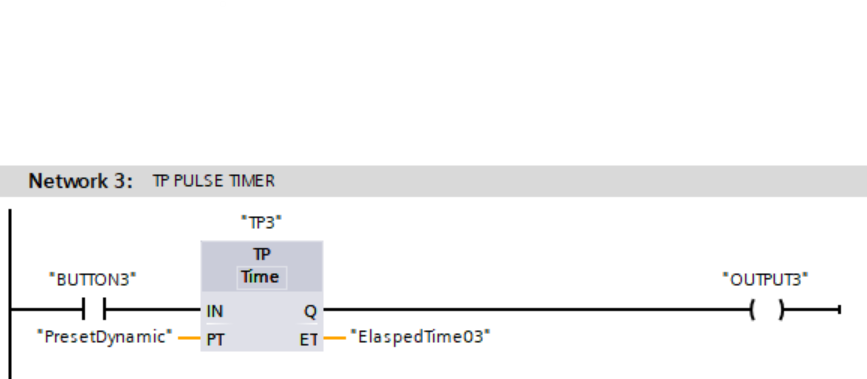
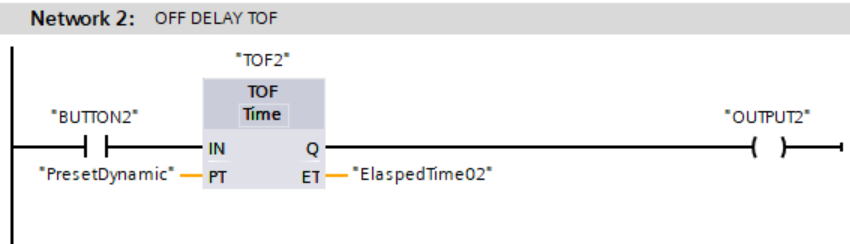


1. Key the following program and sketch its timing diagram. BUTTON1 is activated for 10 seconds.





## Task 2: Using OFF Delay Timer TOF, Pulse Timer TP, Retentive or Accumulate Timer TONR

1. Watch this video to learn how use TOF, TP, and TONR Timers in TIA Portal. <https://youtu.be/HMnRGJDG5mE>
2. Your task is to program and test out the following. Define “PresetDynamic” as datatype TIME, memory as GLOBAL\_MEMORY. While in online mode, you may right click on “Preset Dynamic” to modify the time.  
   
3. Study the Timing Diagrams. You need to understand this to work with timers.

|  |  |
| --- | --- |
| Off Delay Timer  TOF |  |
| Pulse Timer  TP |  |
| Retentive Timer or  Accumulate Timer  TONR |  |

## Task 3: Program Dual Timer Applications (SDL)

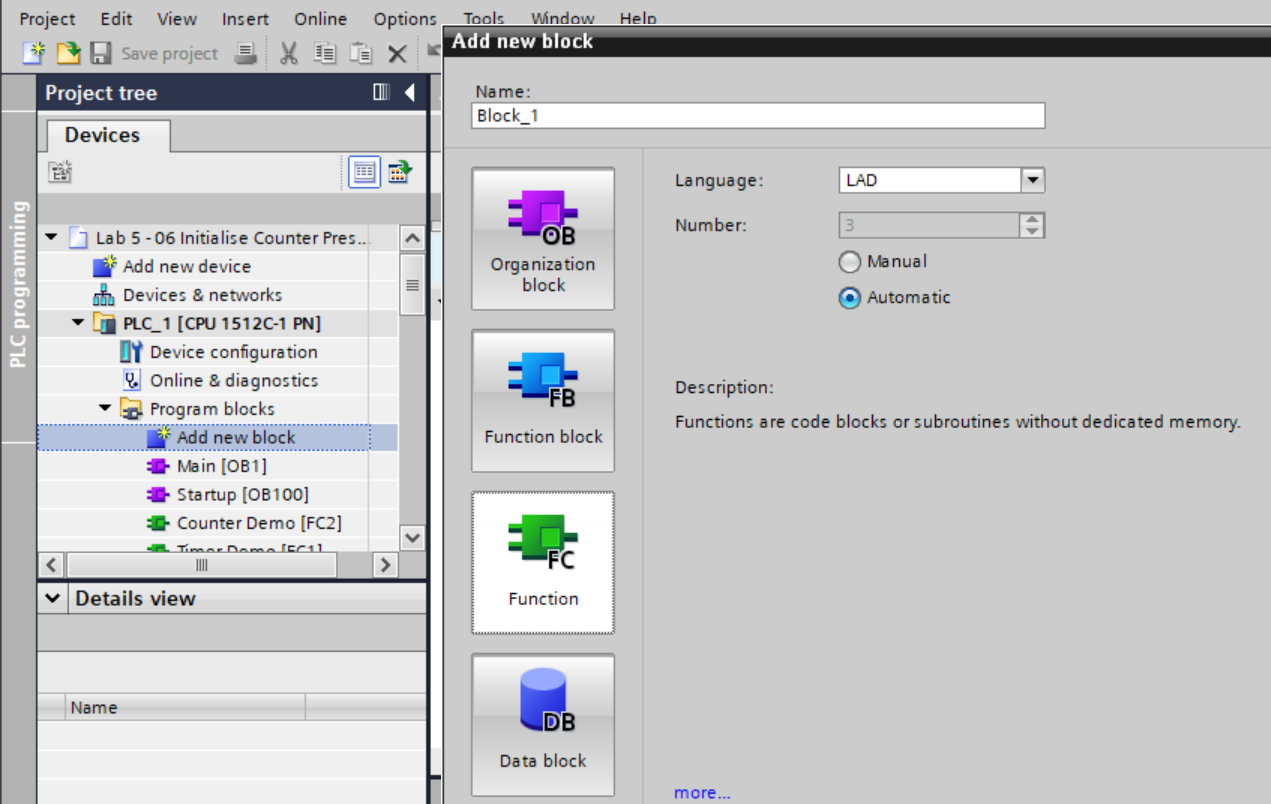
1. Using two different types of Timers(TON, TOF, TP, TONR), create a program that will fulfil the following timing diagram. Test it out using the Conveyor if you are in the lab. Use the appropriate address by referring to the Hardware Address List.



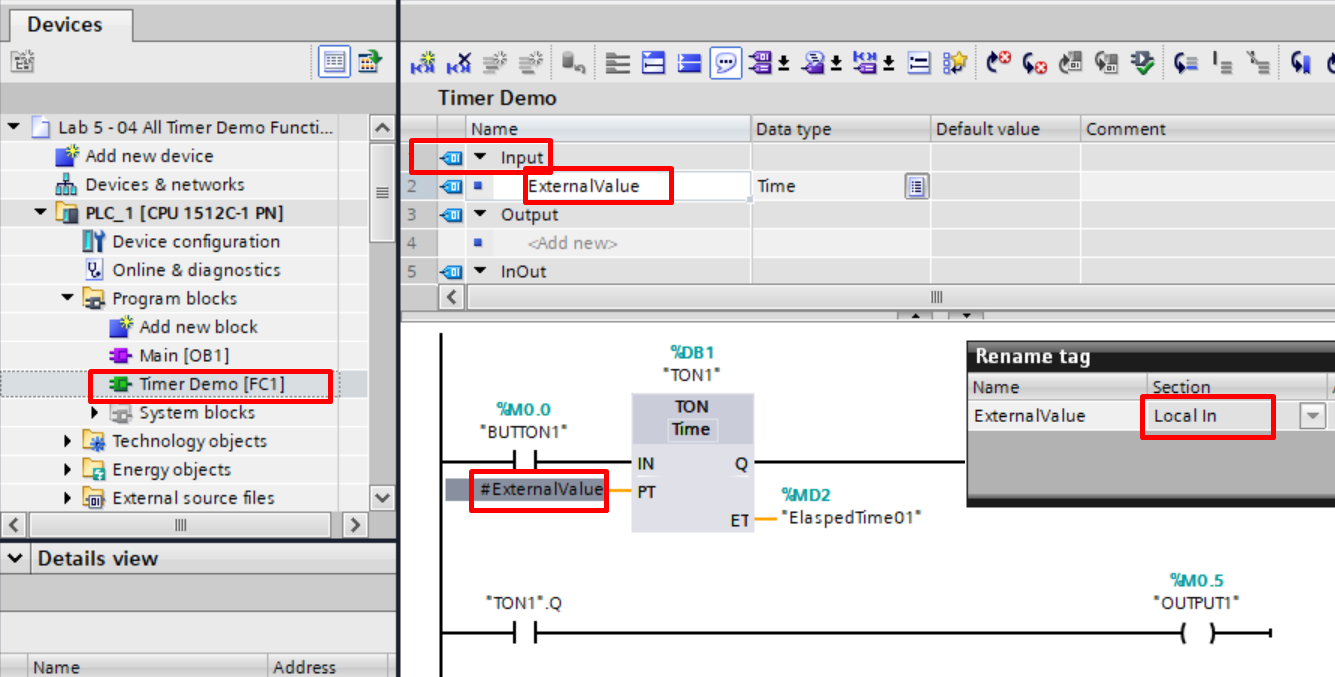
|  |
| --- |
| Insert your ladder diagram here |

## Task 4: Use of Function (FC) in TIA Portal

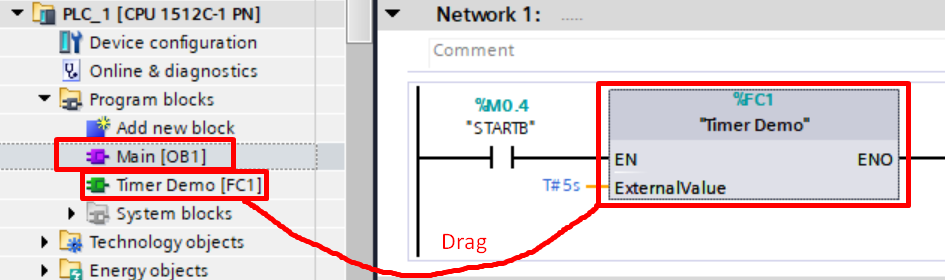
1. Watch the video to learn about the use of Function (FC) in TIA Portal  
   <https://youtu.be/DYdDoRr3MN0>
2. We may write all our program in OB1 (this is called linear programming), at some point in time we may find that it would be useful to create sub-programs that can be called from OB1 (This would be called structured programming).
   * TIA allows us two ways to do this. FC and FB
   * FC are called “function. They are code blocks without memory. Example of FC is maths functions such as add or subtract
   * FB are called “function block”. These are code block with memory. Example is the TON timer block
   * We can build our own FC and FB
3. Click on Add new block and select FC. Language remains as LAD which means ladder. You can name your block as “TIMER DEMO” to make it meaningful



1. Create the following program in FC1. Define “ExternalValue” as **memory type “Local\_In”** (we are using this for the first time). Local Memory is only recognised within this subprogram, as opposed to Global Memory which is recognise throughout the project. Local “In” means this is an input for the function.



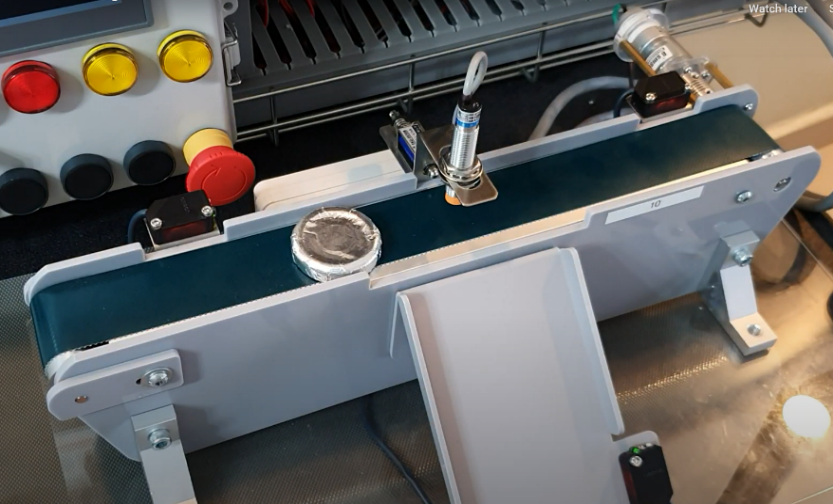
1. To call FC1, simply **drag** it from the left panel and **drop** it into your OB1 ladder diagram.



1. Remember. FC are **DORMANT**, until you call them in OB1, then they will be **ACTIVE**. They are just like the functions “ADD” or “SUBTRACT”. We can choose to call them when we need them.

## Task 5: Application: Conveyor Loading Unloading

1. Create a program such that:
   * When “CART” is detected at point “A” and StartButton is pressed, GreenLED turns ON to indicate operation has began.
   * To allow time for loading goods into the CART, there is a delay of 5 secs before the conveyor starts moving.
   * When the CART reaches point “B”, it stops immediately.
   * To allow time for unloading from the CART, there is a delay of 5 secs before GreenLED turns OFF to indicate the end of operation.
   * If you managed to do all the above successfully, congrats. Now you can level up your program by moving the CART between point A and B continously, making sure to stop 5 secs at each location for and loading and unloading.



“A”

Delay

5s

“B”

Delay

5s

**SDL**

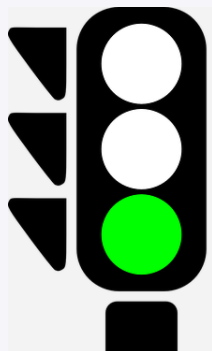
## Task 6: The Blinking Light challenge (SDL)

1. Create a program to blink the blue LED at 0.5Hz. (i.e. 1sec on, 1sec off)   
   Hint: one of the way is to use two TON timers. Or you can think of any possible ways.

## Task 7: The Blinking Light challenge Part 2 (SDL)

1. Start blinking BlueLED only when GreenButton is pressed. Automatically stop blinking after 20 seconds.

## Task 8: The Traffic Light Challenge (SDL)

1. Program the following sequence of a typical traffic light
   * Start with RedLED permanently ON.
   * When GreenButton is pressed,   
     GreenLED turns ON for 4 seconds,   
     followed by blinking at 2Hz for 6 seconds.
   * Followed by YellowLED turn ON for 3 seconds
   * Followed by RedLED turns ON permanently.

## Revision

1. **Match** the following description to the correct timer.

|  |  |
| --- | --- |
| When the input turns ON, this timer always turns ON for a fixed duration specified by PT | ● ● TON |
| When input is turned ON, a timer delays for a few seconds before turning ON | ● ● TOFF |
| When input is turned ON and OFF repeatedly, the timer adds up the ON timing until the total equals PT(preset time), then it turns ON. It can only be off by a RESET | ● ● TP |
| When input transits from ON to OFF, this timer’s output will maintain as ON for a while (PT) before turning OFF | ● ● TONR |

The End

**SOLUTIONS**

Task 5: Application: Conveyor Loading Unloading

Task 6: The Blinking Light challenge (SDL)

Task 7: The Blinking Light challenge Part 2 (SDL)

Task 8: The Traffic Light Challenge (SDL)