

University of Wales Trinity Saint David

School of Engineering



# PLC Assignment

Ethan L J Byrne

BYR19852790

Electrical and Electronic Engineering HND

2023 / 2024

PLCs and Control – GOEE5002

## Table of Contents

|                                                          |    |
|----------------------------------------------------------|----|
| Introduction .....                                       | 3  |
| Main Body .....                                          | 4  |
| What is a PLC? .....                                     | 4  |
| PLC Tag Table .....                                      | 5  |
| PLC Networks .....                                       | 6  |
| HMI .....                                                | 11 |
| Operation of the HMI .....                               | 12 |
| Operation of the Conveyor .....                          | 14 |
| Comparison with Initial Requirements of the System ..... | 16 |
| Conclusion .....                                         | 17 |
| References .....                                         | 18 |

## Introduction

The task which was assigned is to sort items along a conveyor according to size with the use of a PLC (Programmable Logic Controller) to code the operations of the conveyor. The items are in the form of metal disks of varying diameter and height, ranging from small, medium, and large.

The following system operation is required: -

- The operator will press and hold a switch for 5 seconds in order to start the conveyor. During this period a green lamp will flash.
- When the conveyor is running the green lamp will be ON.
- The operator may stop the conveyor at any time by pressing the red switch. When the conveyor is stopped a red lamp will be ON.
- Items will be placed on the conveyor at the right-hand side and will progress along the conveyor towards the sorting bins. When reaching the correct bin, a cylinder will operate ejecting the item into the required bin.
- If no items are detected for a period of 30 seconds then the conveyor will automatically stop.

It is also required to provide a HMI (Human Machine Interface) based on the Siemens Touch semantic 7" basic HMI. The HMI should have controls to stop, and start the conveyor, and exercise the cylinders. It should be possible to adjust operating parameters within the program. The display element should have indicators for quantities of counters of each type.

This report will contain the following:

- What a PLC is and what they are used for
- A detailed specification for the operation of this system
- A design in a suitable form.
- The PLC code along with all necessary documentation
- Test the system
- Demonstration of the system operating
- Comparison of the final system with the initial requirements

## Main Body

### What is a PLC?

Programmable Logic Controllers (PLCs) are industrial computers with multiple inputs and outputs that control and monitor industrial equipment based on specific programming (inductive automation, 2020).

The sizes and form factors of PLCs vary widely. They can range from small to large in size. Certain PLCs have only basic inputs and outputs (I/O) and are therefore more modular in nature, but they can be tailored to match various industrial applications by adding more back planes and functional modules.

## PLC Tag Table

Below is a table which contains the tags for each component and operation of the PLC, simply known as a PLC tag table.

A PLC tag table is required for every PLC for them to work, as without tags, you cannot assign the operations to a component and therefore the code will not be able to run. An example of this would be assigning a switch as an input which connects to an LED as an output, so that when pressing that switch will turn on that LED. In the tag table, the switch would be assigned to I 0.0 for the first input and the LED would be Q 0.0 for the output. In order to set multiple inputs or outputs, you would go from 0.0 to 0.1 to 0.2 and so on.

Below is the tag table for the operation of this conveyor.

| Default tag table |                      |           |         |                          |                                     |                                     |                                     |
|-------------------|----------------------|-----------|---------|--------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
|                   | Name                 | Data type | Address | Retain                   | Access...                           | Write...                            | Visibl...                           |
| 1                 | Cylinder Small       | Bool      | %Q0.0   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 2                 | R1 (Cylinder 1)      | Bool      | %I0.0   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 3                 | R1 (Cylinder 2)      | Bool      | %I0.1   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 4                 | R1 (Cylinder 3)      | Bool      | %I0.2   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 5                 | Laser 1              | Bool      | %I0.3   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 6                 | Laser 2              | Bool      | %I0.4   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 7                 | Switch Red           | Bool      | %I0.5   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 8                 | Switch Green         | Bool      | %I0.6   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 9                 | Cylinder Medium      | Bool      | %Q0.1   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 10                | Cylinder Large       | Bool      | %Q0.2   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 11                | Conveyor Start       | Bool      | %Q0.3   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 12                | Conveyor Reverse     | Bool      | %Q0.4   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 13                | LED Red              | Bool      | %Q0.5   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 14                | LED Green            | Bool      | %Q0.6   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 15                | System_Byte          | Byte      | %MB1    | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 16                | FirstScan            | Bool      | %M1.0   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 17                | DiagStatusUpdate     | Bool      | %M1.1   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 18                | AlwaysTRUE           | Bool      | %M1.2   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 19                | AlwaysFALSE          | Bool      | %M1.3   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 20                | Clock_Byte           | Byte      | %MB0    | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 21                | Clock_10Hz           | Bool      | %MO.0   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 22                | Clock_5Hz            | Bool      | %MO.1   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 23                | Clock_2.5Hz          | Bool      | %MO.2   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 24                | Clock_2Hz            | Bool      | %MO.3   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 25                | Clock_1.25Hz         | Bool      | %MO.4   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 26                | Clock_1Hz            | Bool      | %MO.5   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 27                | Clock_0.625Hz        | Bool      | %MO.6   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 28                | Clock_0.5Hz          | Bool      | %MO.7   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 29                | Counter Reset Button | Bool      | %I2.0   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 30                | Conveyor Stop        | Bool      | %Q2.1   | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

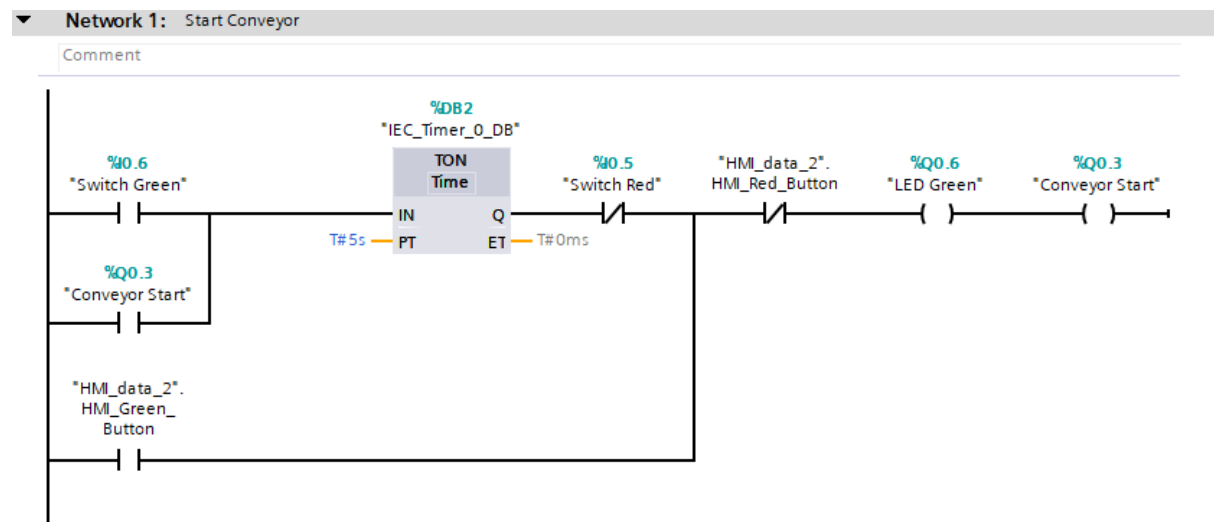
As mentioned before, I in a tag table stands for input and Q stands for output. Each switch is set to inputs, along with the lasers on the conveyor belt. The LEDs are set to outputs, as well as the pistons for pushing the cylinders into their individual containers. M in a tag table stands for memory, which is a dedicated variable for the system. The memories in this table are mostly made up of each of the clocks for the system network operations.

There is also a counter reset button as seen toward the bottom of the table, however the counter is an internal operation which cannot be observed until the HMI is used. This HMI will be discussed extensively later in the report.

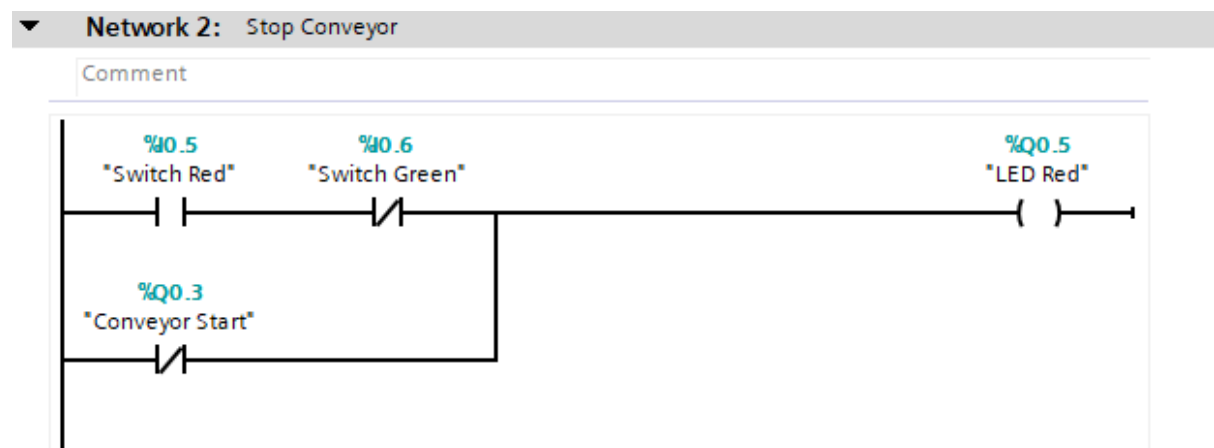
## PLC Networks

Below are the networks which contain the code for the PLC to operate:

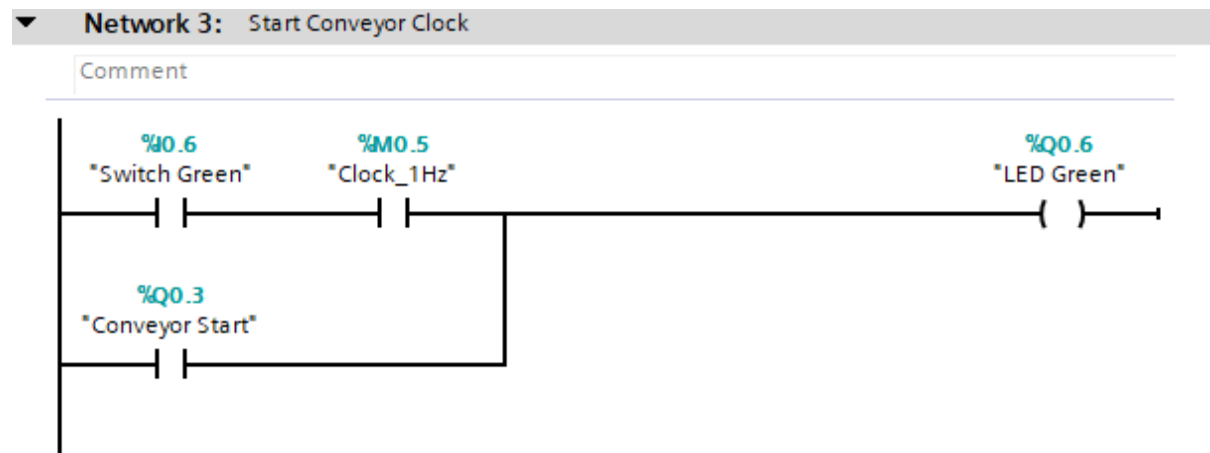
Network 1 shown below is the code for the conveyor to start moving. This is done by pressing and holding the green button for 5 seconds, and after it is released, the conveyor turns on and begins to move. To make sure that the conveyor starts only when the button is held for 5 seconds or more, a timer is added, which of course has its time set to 5s.



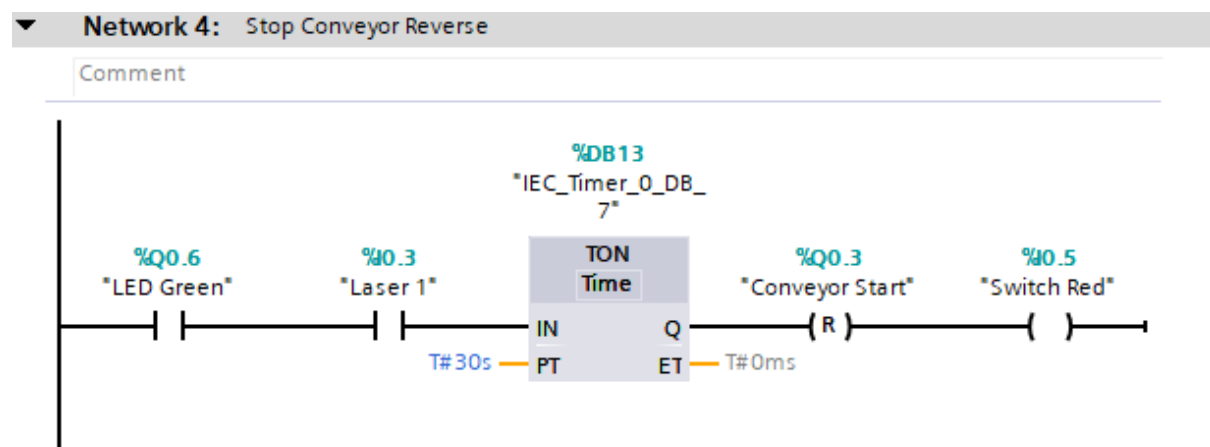
Network 2 is the code for stopping the conveyor. This is done by pressing the red button, which instantly turns the conveyor off. The conveyor will keep moving unless this button is pressed.



Network 3 contains the operation for the clock for the start operation of the conveyor shown in Network 1. This is what counts how long the start button has been held.

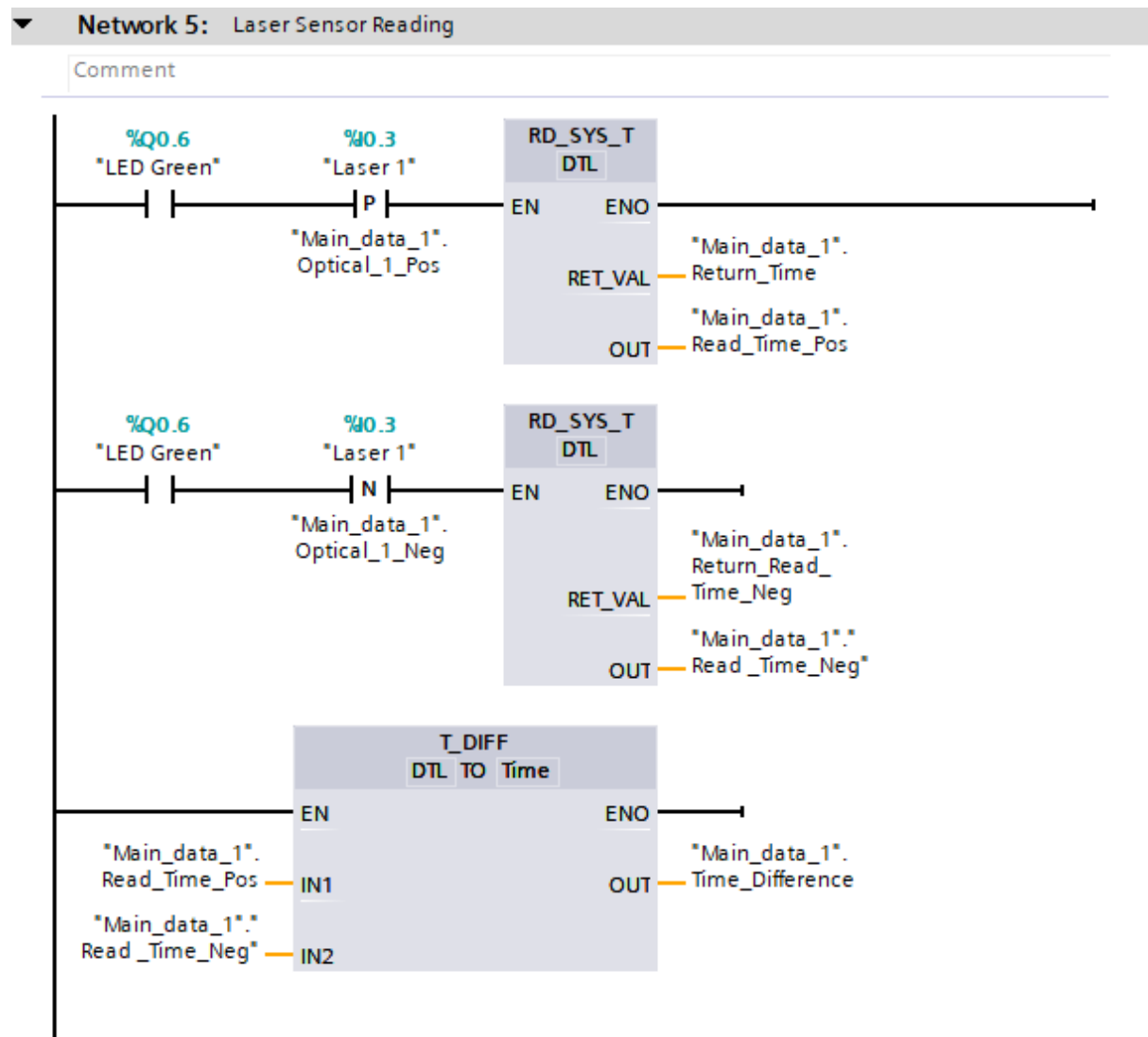


Network 4 has the operation to automatically stop the conveyor when the conveyor does not detect any items moving along for 30 seconds, as can be seen below with the timer which has been set to 30s.

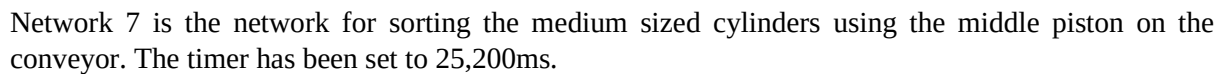


Network 5 is for the sensors to be able to operate as intended and read the cylinders when they break the lasers and pass through. As each cylinder varies in size and needs to be sorted as such, the lasers work to ensure this by timing how long each size of cylinder takes to pass through the lasers fully, meaning the time for the laser to recover after breaking.

It is important that the speed of the conveyor is consistent for the entirety of the operation as the time for the cylinders to pass through the cylinders is based upon the conveyor moving at the same speed, and if the conveyor were to be sped up or slowed down, the cylinders would not be pushed into their containers.



As this is the operation for the small cylinder, the timer has been set to 1,200ms, the quickest of the three cylinders. There is also a counter coded which counts the number of small cylinders have been passed along the conveyor which can be shown on the HMI.



Network 8: Large Cylinders

Comment

```

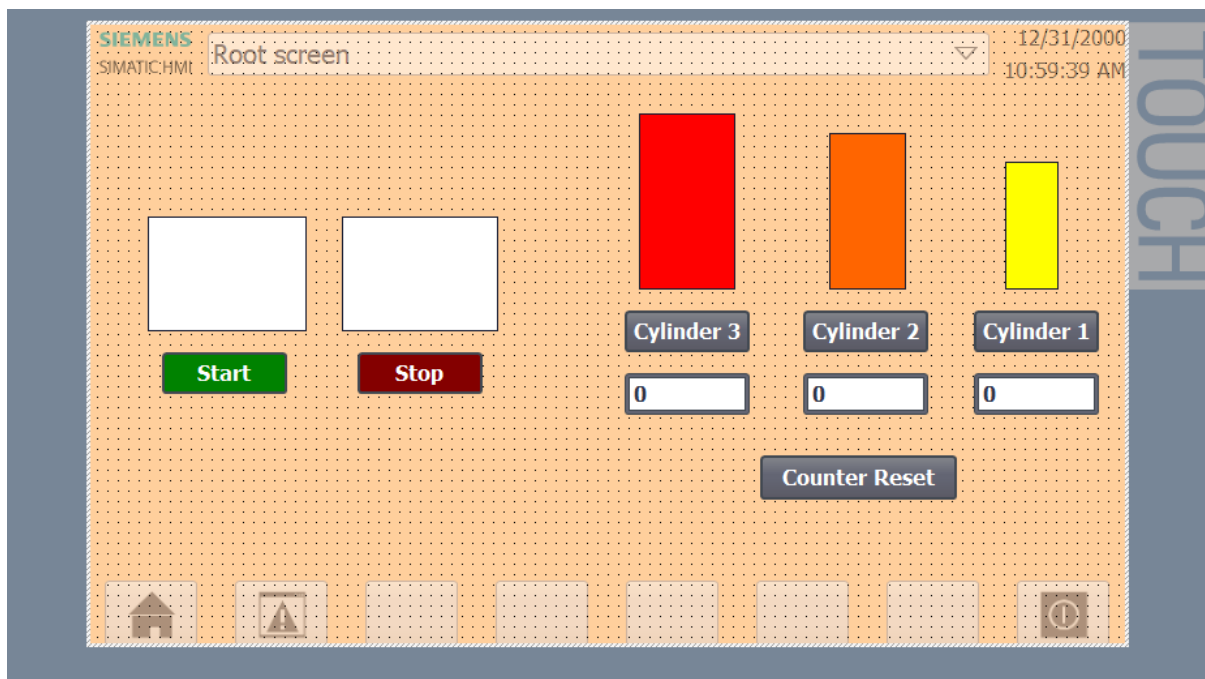
graph LR
    Laser2["'Laser 2'"] --> T_CONV1[TON]
    T_CONV1 --> TON[TON]
    TON --> TP[TP]
    TP --> T_CONV2[T_CONV]
    T_CONV2 --> CTU[CTU]
    CTU --> LargeFull["'Large Full'"]
    CTU --> LargeCounter["'Large Counter'"]
    CTU --> LargeLimit["'Large Limit'"]
    HMIReset["'HMI Counter Reset'"] --> CTU
    Cylinder3Man["'Cylinder 3 man'"] --> CTU
    
```

## HMI

The HMI (Human Machine Interface) is an interface that can be used by people to engage with machines to monitor and control that machine's operations (Aveva, 2024). The HMI that was used in order to interact with this conveyor system was in the form of a touchscreen, although they can also come with keyboards.

HMIs must be coded before they are able to be used with the machine that you want to use it for, and so first it had to be connected to the PLC so that it could access all of the PLC's tags and operations. Once connected, the HMI screen can be seen, and you can put anything that you want here, including images and shapes. You can also add operations to individual shapes to turn them into buttons that control different operations of the machine, which is what has been done in this conveyor system.

Below is the HMI that was created for the conveyor system.



On the display of the HMI are various buttons. Starting from the left, there is a button that says start, which starts the conveyor when held. The box above that then turns green when the system is operating. To the right of that is a stop button which stops the conveyor instantly when pressed, and similarly to the start button, the box above turns red when the conveyor is off.

On the right are three buttons which are labelled cylinder 1, 2, and 3, with counters below them which are currently 0. When the cylinder buttons are pressed, or if the conveyor is 0 and a cylinder of the corresponding button passes through the lasers, the counter below goes up. In order to show which counter displays which cylinders, rectangles of different sizes were added which vary in size just like the cylinders. The small rectangle represents the small cylinders, the middle-sized rectangle represents the medium cylinders, and of course the larger rectangle represents the large cylinders.

There is also a button labelled 'Counter Reset' below the counters. As implied, when this button is pressed, all of the counters reset back to zero.

## Operation of the HMI

After coding the conveyor and the HMI screen, the next step was to test the system and see if the PLC code worked as intended. In order to prove the operations as being successful, images have been provided below which shows each button of the HMI working along with images of the conveyor operating, with the cylinders moving along the conveyor.

Below are the images of the HMI working.

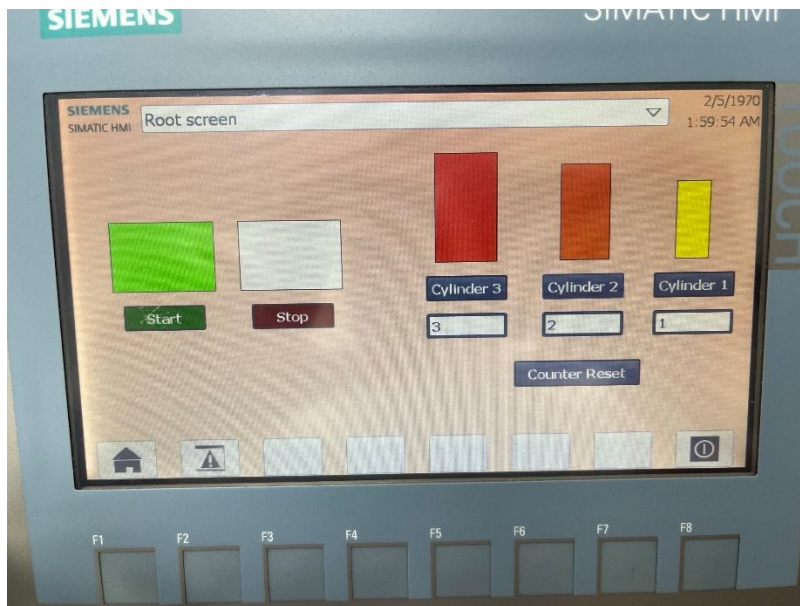
Here is the HMI when the conveyor is not running. The stop button is red to show the operator that the machine is not currently active, and the start button is white, meaning the start button has not yet been pressed.



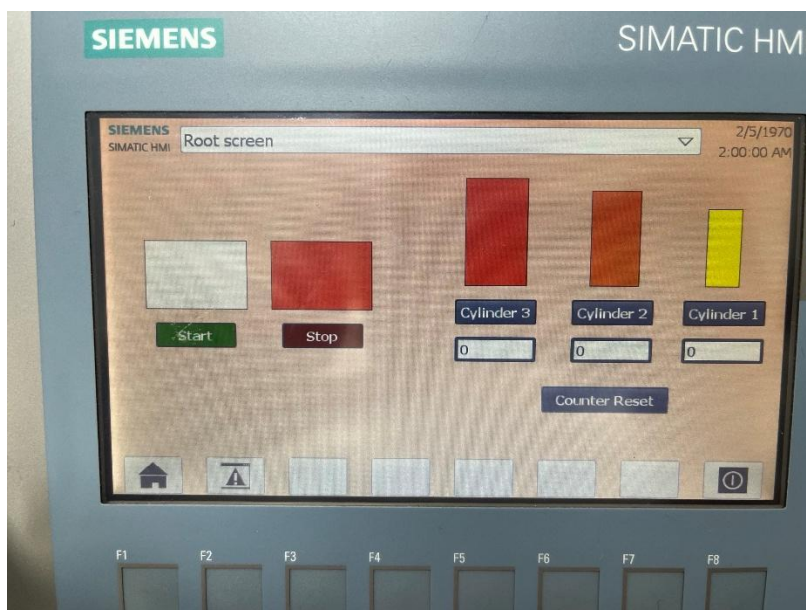
Here is the HMI when the conveyor is running. The stop button has now gone from red to white, showing that the machine is no longer stopped, and instead the start button has gone green.



The below image is supposed to show that counter works, as the numbers for the counters have gone up from 0. It shows that 3 large cylinders, 2 medium cylinders, and 1 small cylinder have travelled along the conveyor and have been sorted into their designated containers.



As seen below, the counters have gone back from 3, 2, and 1 to 0. The reason for this is because the counter reset button on the display has been pressed, which has reset the counter back to 0.



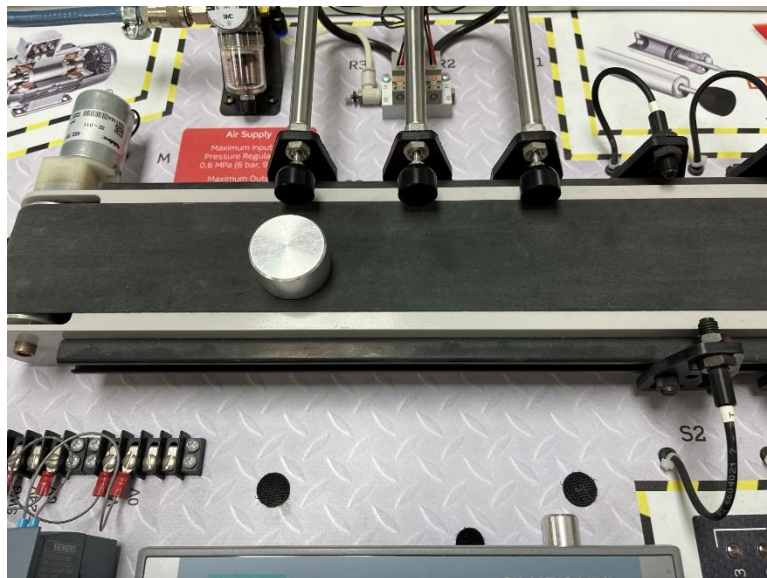
## Operation of the Conveyor

Below are the images that show the conveyor working.

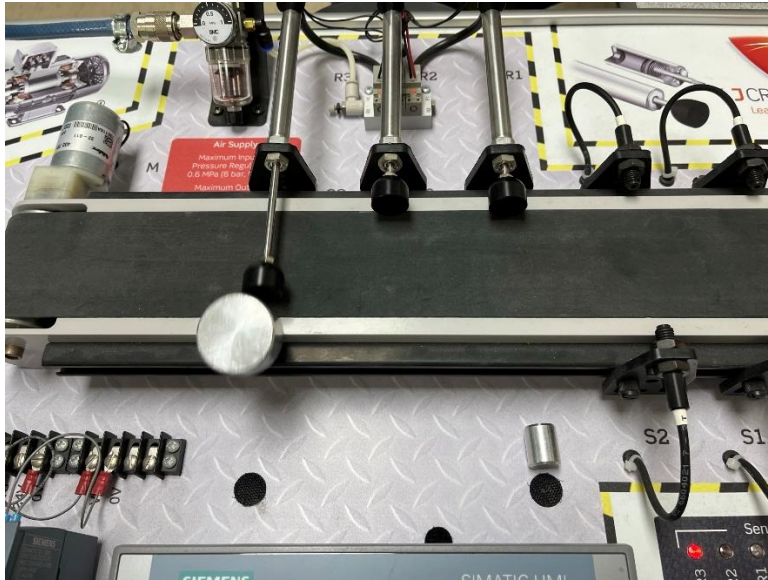
This image shows a large-sized cylinder moving along the conveyor. It has passed through the lasers and the small cylinder piston. This shows that the pistons have been coded correctly, and will let the large and medium cylinders pass.



This image shows the large cylinder moving to the piston for the large cylinders which should push it off the conveyor belt and into the container.



This image shows the same large cylinder being pushed off of the conveyor belt by the final piston, which is its designated cylinder. This shows that pistons have been coded correctly and will push the cylinders, with the calculations for the timers being correct. If the cylinder would have been pushed by another piston or had not been pushed and kept moving past all of the pistons, then it would be clear that the PLC code does not work correctly.



## Comparison with Initial Requirements of the System

The images shown previously show that the conveyor works in reference to the initial requirements of the system. As shown in network 1 of the PLC networks, when the green button is held for 5 seconds, the conveyor starts, and the green lamp also flashes when the button is held. This can also be seen in the image of the HMI start button turning green to indicate that the conveyor has started moving.

Network 2 and the stop button on the HMI shows that pressing the red switch will be turn off the conveyor and activate the red lamp when the conveyor is off.

Network 4 shows the operation of which the conveyor automatically stops when the lasers do not detect any items after 30 seconds.

Networks 6, 7, and 8 each show the code for sorting the different sized cylinders into their own containers when placed on the conveyor and the lasers detect them. The final image of the conveyor working also shows that this code is correct as only the final piston pushed the large cylinder off of the conveyor when it reached the piston.

## Conclusion

To conclude, the task of coding a PLC's operations in order to sort varying sizes of cylinders along the conveyor, and also coding the HMI to control said operations remotely using a touchscreen was successful. The list of initial requirements were met along with each of the operations.

All of the information that the task had asked for was provided in the report, and evidence of the system working has been provided also.

## References

Aveva, 2024. *Human Machine Interface (HMI) Software & Solutions*. [Online]

Available at:

<https://www.aveva.com/en/solutions/operations/hmi/#:~:text=What%20is%20Human%20Machine%20Interface%3F,Interfaces%20are%20keyboards%20and%20touchscreens.>

[Accessed 23 May 2024].

inductive automation, 2020. *PLC: Programmable Logic Controller*. [Online]

Available at: [https://inductiveautomation.com/resources/article/what-is-a-](https://inductiveautomation.com/resources/article/what-is-a-PLC#:~:text=Programmable%20Logic%20Controllers%20(PLCs)%20are,equipment%20based%20on%20custom%20programming.)

[PLC#:~:text=Programmable%20Logic%20Controllers%20\(PLCs\)%20are,equipment%20based%20on%20custom%20programming.](https://inductiveautomation.com/resources/article/what-is-a-PLC#:~:text=Programmable%20Logic%20Controllers%20(PLCs)%20are,equipment%20based%20on%20custom%20programming.)

[Accessed 23 May 2024].