



University of Wales trinity Saint David

School of engineering

PLC Assignment 2

Scott Tavner

2220039

HND in electrical and electronic engineering

2024

PLC's and Control

GOEE5002

Contents

Introduction.....	3
Design of the system.....	4
Coding the Siemens PLC	5
Setting up the PLC and HMI	5
Setting PLC tag table and data block	7
Programming networks.....	9
Network One	10
Network Two	11
Network Three	12
Network Four	13
Network Five	13
Network Six	14
HMI.....	15
Setting Tag Tables	15
Designing the Screen	16
Adding functions to the screen.....	17
Results.....	22
Concussion.....	24
References.....	25

Introduction

With the increasing demand of products from manufacturing many companies have turned to an automation system within factories. These automated systems prove to be far more efficient due to the production line being able to run 24/7. However, with all this new investment in automation comes innovation in the way the systems are controlled, one of these innovations was the PLC. (plantaautomation-technology, n.d.)

The PLC or Programmable Logic Controller are computing devices that are used to manage electromechanical processes within the industrial sector. The term "industrial PC" refers to a PLC's primary purpose as a specialised industrial computing device and is sometimes used to refer to PLCs. PLCs decide what should happen to an output device, such turning on or off a light, based on signals from an input device, such as a light switch. PLCs can also be used to move data from offsite locations or factory devices to centralised applications, which are typically PC-based. PLCs are frequently used to control device events, monitor and report on devices, and diagnose hardware issues in tools and industrial machinery. (paessler, n.d.)

This report will aim to cover the design and operation of a feedback control system. This system will utilize the Siemens S7-1200 series PLC. The PLC will be used to control a convey system that will sort different diameter cylinders into corresponding sections.

The specific requirement the system must meet are:

- A start button will need to be pressed for 3 contiguous seconds to begin operation of the conveyor.
- A green light is illuminated when conveyor is in operation.
- An emergency stop button must be present that can be pressed at any time to shut off the conveyor.
- When the conveyor is not in operation a red light will be illuminated.
- Items placed on the right side of the conveyor will travel along and be sorted into 3 different bins for small, medium and large objects.
- An automatic shutdown will occur when no objects are detected for 30 seconds.
- A Siemens Touch sematic 7" basic HMI will be present that can be used to control the system.

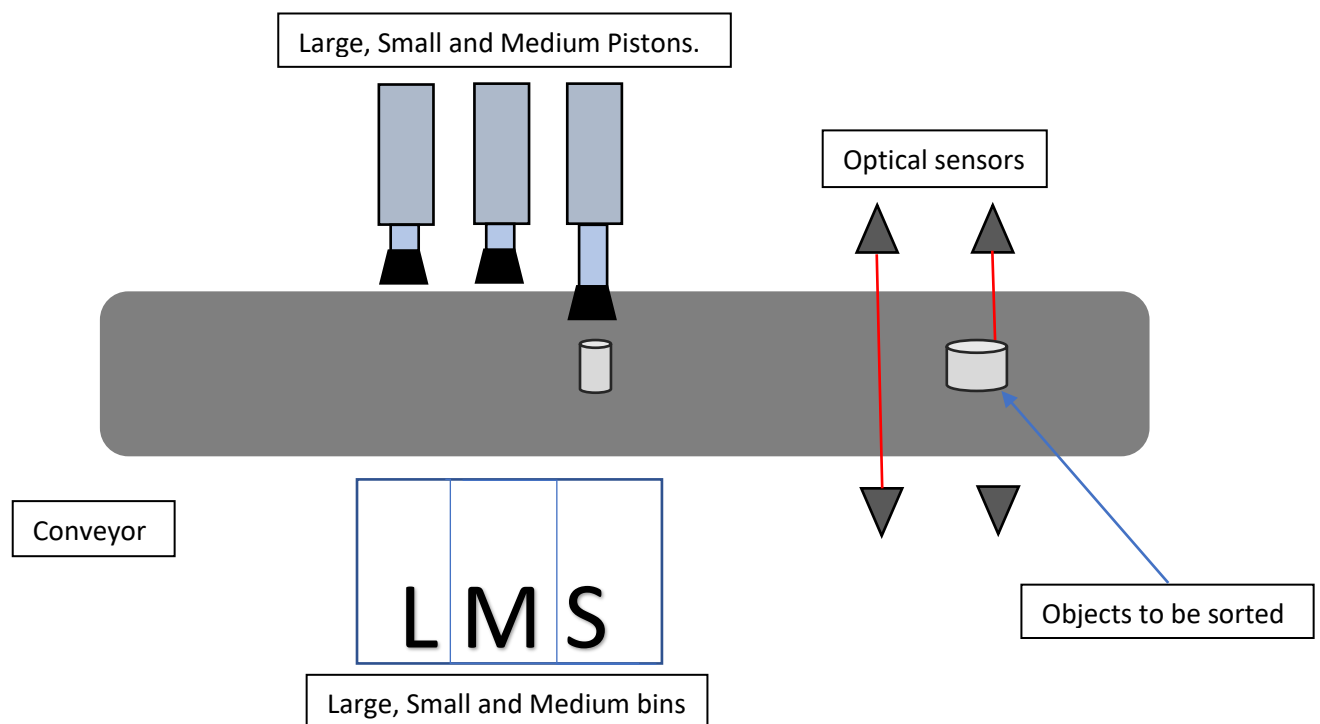
Design of the system

The system will use a conveyor belt system that will transport the objects at a constant speed through two optical sensors. These optical sensors will allow the PLC to identify the size of the cylinder from the time in which the beam of the optical sensor is broken. The cylinders will then be transported alongside three pistons that will protrude, pushing the cylinders into their corresponding bins.

In order to control the operation of the conveyor, there will be a control board that will contain the start and stop buttons as well as the accompanying green and red lights. There will be a switch to change the system to a manual mode. The manual mode will have switches for the three pistons, the red light, the green light, and the motor controlling the conveyor.

Alongside the control board, an HMI screen will be used to view and control parts of the system. HMI stands for Human Machine Interface and is a robust touch screen that can connect to the PLC system. This HMI will have the means to control the operation of the start stop functions, the operation of the sorting pistons, and a view of the count of objects that have been sorted.

See (Figure 1.) for a diagram of the system.

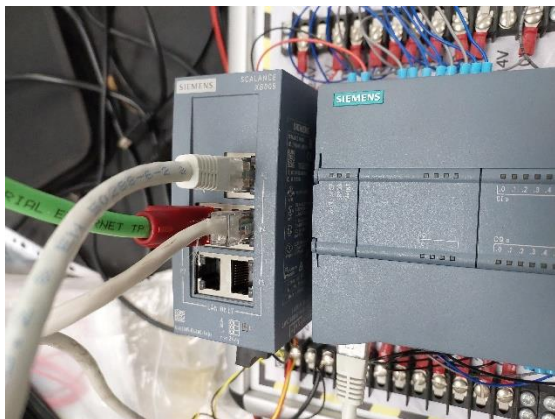


(figure 1.)

Coding the Siemens PLC

Setting up the PLC and HMI

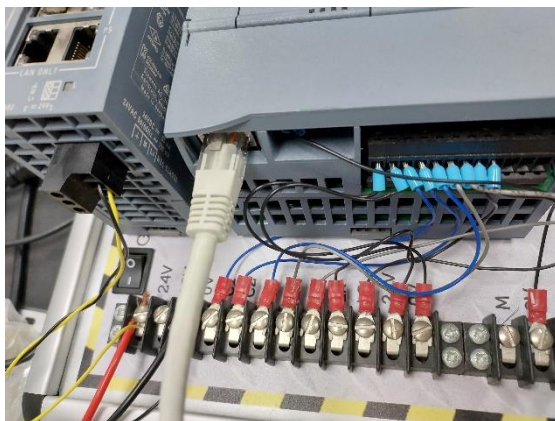
Before starting, the PLC, HMI and computer all need to be physical connected together. To do this each device needs to be connected together by ethernet cable through the junction box. (figure 2) shows the junction block. (figure 5) shows the PC connection, (figure 3) shows the HMI connection and finally the PLC connection is shown in (figure 4).



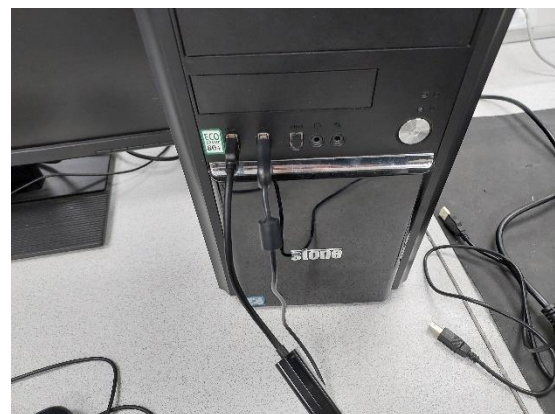
(figure 2)



(figure 3)



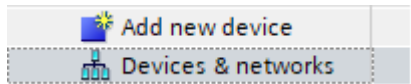
(figure 4)



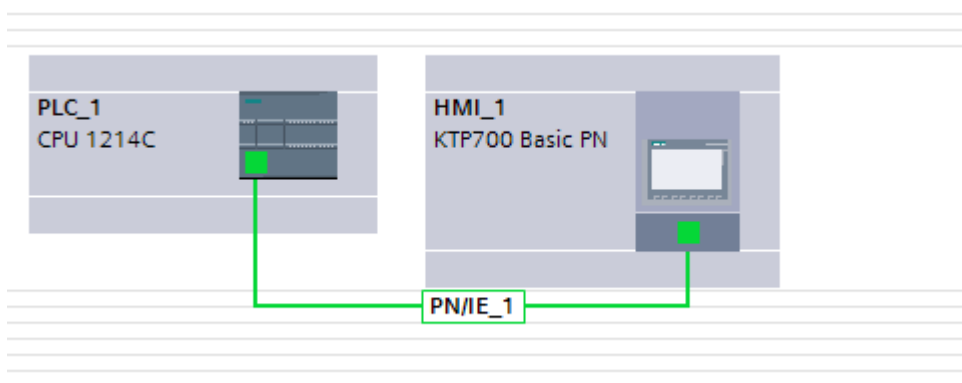
(figure 5)

Now that both the PLC and HMI have been set up physically they must be connected on the software to let the computer know that they are connected.

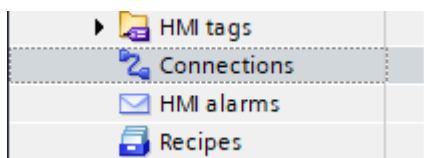
In order to do this, go to the “Devices and &networks” tab on the left side near the top.



Next there should be a representation of the PLC and HMI with green boxes. Drag these green boxes together in order to connect the two devices.



Now to check the connection go to the “Connections” tab under the HMI section.



From here the connection can be seen and the two devices addresses need to be checked that they match the devices.

The image shows the 'Connections' window in the software. At the top is a table with the following data:

Name	Communication driver	HMI time synchronization mode	Station	Partner	Node	Online	Comment
HMI_Connection_1	SIMATIC 57-1200	None	57-1200 station_1	PLC_1	CPU 1214C DC/DC/...	☒	
<Add new>							

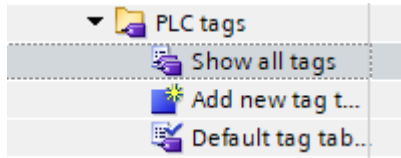
Below the table are two tabs: 'Parameter' and 'Area pointer'. The 'Parameter' tab is selected. It shows the configuration for the connection:

- KTP700 Basic PN** (HMI device):
 - Interface: PROFINET (X1)
 - Address: 192.168.0.2
 - Access point: 57ONLINE
- PLC** (Partner):
 - Address: 192.168.0.1
 - Access password: (empty field)

A line connects the HMI device to the PLC device.

Setting PLC tag table and data block

The next thing to do is set up the tag table for the PLC. A tag table is where the tags that will be used are defined and connected to the physical connection. In order to find the PLC tag table “Show all tags” tab under the PLC section.



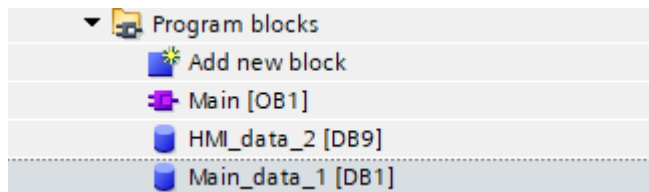
There are three types of tags that can be used I, Q and M. I is to tell the software that this tag is an input and then the following digits will give the address of the connection. Then for an output Q is used and M is for setting up an internal tag (memory tag). Unlike the output and input tags an internal tag will not be for a physical connection, instead it will represent a saved state that can be set to high or low. (Liam, 2022)

The data type that will be used is bool as the tag states will only need to be high or low.

Here are the set up tags:

PLC tags									
	Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	StopBT	Default tag table	Bool	%I0.5	☐	☒	☒	☒	
2	StartBT	Default tag table	Bool	%I0.6	☐	☒	☒	☒	
3	ConveyorShutDown	Default tag table	Bool	%Q2.1	☐	☒	☒	☒	
4	Sensor1	Default tag table	Bool	%I0.3	☐	☒	☒	☒	
5	Sensor2	Default tag table	Bool	%I0.4	☐	☒	☒	☒	
6	PistonL	Default tag table	Bool	%Q0.2	☐	☒	☒	☒	
7	PistonM	Default tag table	Bool	%Q0.1	☐	☒	☒	☒	
8	PistonS	Default tag table	Bool	%Q0.0	☐	☒	☒	☒	
9	ConveyorStart	Default tag table	Bool	%Q0.3	☐	☒	☒	☒	
10	ConveyorReverse	Default tag table	Bool	%Q0.4	☐	☒	☒	☒	
11	ResetBT	Default tag table	Bool	%I2.0	☐	☒	☒	☒	
12	System_Byte	Default tag table	Byte	%MB1	☐	☒	☒	☒	
13	FirstScan	Default tag table	Bool	%M1.0	☐	☒	☒	☒	
14	DiagStatusUpdate	Default tag table	Bool	%M1.1	☐	☒	☒	☒	
15	AlwaysTRUE	Default tag table	Bool	%M1.2	☐	☒	☒	☒	
16	AlwaysFALSE	Default tag table	Bool	%M1.3	☐	☒	☒	☒	
17	Clock_Byte	Default tag table	Byte	%MB0	☐	☒	☒	☒	
18	Clock_10Hz	Default tag table	Bool	%M0.0	☐	☒	☒	☒	
19	Clock_5Hz	Default tag table	Bool	%M0.1	☐	☒	☒	☒	
20	Clock_2.5Hz	Default tag table	Bool	%M0.2	☐	☒	☒	☒	
21	Clock_2Hz	Default tag table	Bool	%M0.3	☐	☒	☒	☒	
22	Clock_1.25Hz	Default tag table	Bool	%M0.4	☐	☒	☒	☒	
23	Clock_1Hz	Default tag table	Bool	%M0.5	☐	☒	☒	☒	
24	Clock_0.625Hz	Default tag table	Bool	%M0.6	☐	☒	☒	☒	
25	Clock_0.5Hz	Default tag table	Bool	%M0.7	☐	☒	☒	☒	
26	StopLT	Default tag table	Bool	%Q0.5	☐	☒	☒	☒	
27	StartLT	Default tag table	Bool	%Q0.6	☐	☒	☒	☒	
28	<Add new>				☐	☒	☒	☒	

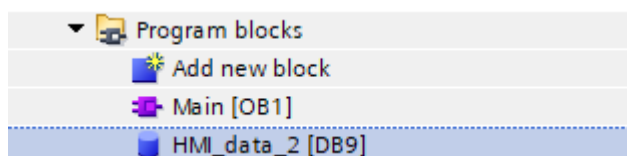
Now the data blocks need to be set up. These data block will be used to save values that is needed in the CPU of the PLC so that they can be called later when needed. In order to do this, select the “Add new Block” under “Program blocks”. A menu will pop up where a global DB block will be chosen.



Once that is set up the tags can be filled in:

Main_data_1									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static			☐	☐	☐	☐	☐	
2	Counter_PistonL	Int	0	☐	☒	☒	☒	☐	
3	Counter_PistonM	Int	0	☐	☒	☒	☒	☐	
4	Counter_PistonS	Int	0	☐	☒	☒	☒	☐	
5	Time_Start	DTL	DTL# 2021-10-...	☐	☒	☒	☒	☐	
6	Time_Return	Int	0	☐	☒	☒	☒	☐	
7	Pos_Sensor1	Bool	false	☐	☒	☒	☒	☐	
8	Pos_ReadTime	DTL	DTL# 1970-01-01-4	☐	☒	☒	☒	☐	
9	Pos_ReturnReadTime	Int	0	☐	☒	☒	☒	☐	
10	Neg_Read Time	DTL	DTL# 1970-01-01-4	☐	☒	☒	☒	☐	
11	Neg_ReturnReadTime	Int	0	☐	☒	☒	☒	☐	
12	Neg_Sensor1	Bool	false	☐	☒	☒	☒	☐	
13	Time_Difference	Time	T# 0ms	☐	☒	☒	☒	☐	
14	Time_Int	Int	0	☐	☒	☒	☒	☐	
15	Counter_Small	Int	0	☐	☒	☒	☒	☐	
16	Counter_Medium	Int	0	☐	☒	☒	☒	☐	
17	Counter_Large	Int	0	☐	☒	☒	☒	☐	
18	Pos_Sensor2	Bool	false	☐	☒	☒	☒	☐	
19	MAX_Small	Bool	false	☐	☒	☒	☒	☐	
20	MAX_Medium	Bool	false	☐	☒	☒	☒	☐	
21	MAX_Large	Bool	false	☐	☒	☒	☒	☐	
22	Static_1	DInt	0	☐	☒	☒	☒	☐	
23	smallone	Int	0	☐	☒	☒	☒	☐	
24	medone	Int	0	☐	☒	☒	☒	☐	
25	largeone	Int	0	☐	☒	☒	☒	☐	
26	Man_PistonS	Bool	false	☐	☒	☒	☒	☐	
27	Man_PistonM	Bool	false	☐	☒	☒	☒	☐	
28	Man_PistonL	Bool	false	☐	☒	☒	☒	☐	
29	SMALL_LIMIT	Int	5	☐	☒	☒	☒	☐	
30	MED_LIMIT	Int	5	☐	☒	☒	☒	☐	
31	LARGE_LIMIT	Int	5	☐	☒	☒	☒	☐	
32	ConveyorBt	Bool	false	☐	☒	☒	☒	☐	

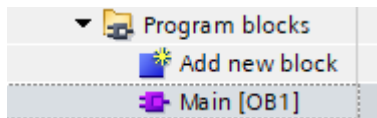
For this system two data blocks will be used:



HMI_data_2									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	Static			☐	☐	☐	☐	☐	
2	HMI_GreenBT	Bool	false	☐	☒	☒	☒	☐	
3	HMI_RedBT	Bool	false	☐	☒	☒	☒	☐	
4	HMI_ResetBT	Bool	false	☐	☒	☒	☒	☐	

Programming networks

The PLC has now been set up and is ready to be programmed. To do this, networks need to be set up that will determine how different tags will affect each other when they're states are changed. To find this go to the "Main" tab under the "Program blocks" section.



There are three main symbols that will be used in the networks.

The first is to tell the PLC to read a TAG and if it is HIGH then to set signal through.



The next is the very similar however it will let signal through if the state is LOW.



And the final that will be used is to tell the PLC to set this tag HIGH.

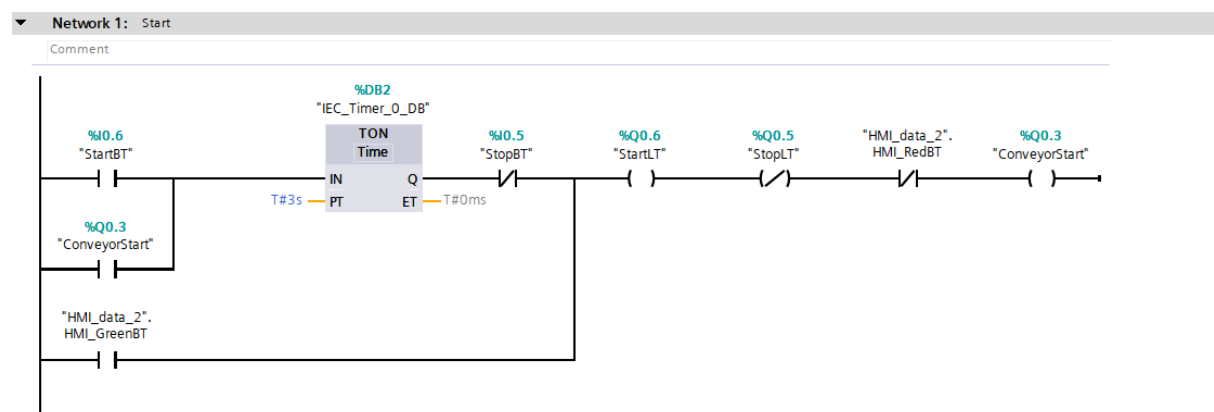


Network One

The first network that will be set up is to start the conveyor. It will do this by if the start button is pressed a TON timer will receive high and will allow it to go through after 3 seconds of continuous signal. If the stop button is undressed the signal will go through setting the conveyor and start light HIGH whilst also setting the stop light low. There is also a link to the HMI stop button that will need to be undressed.

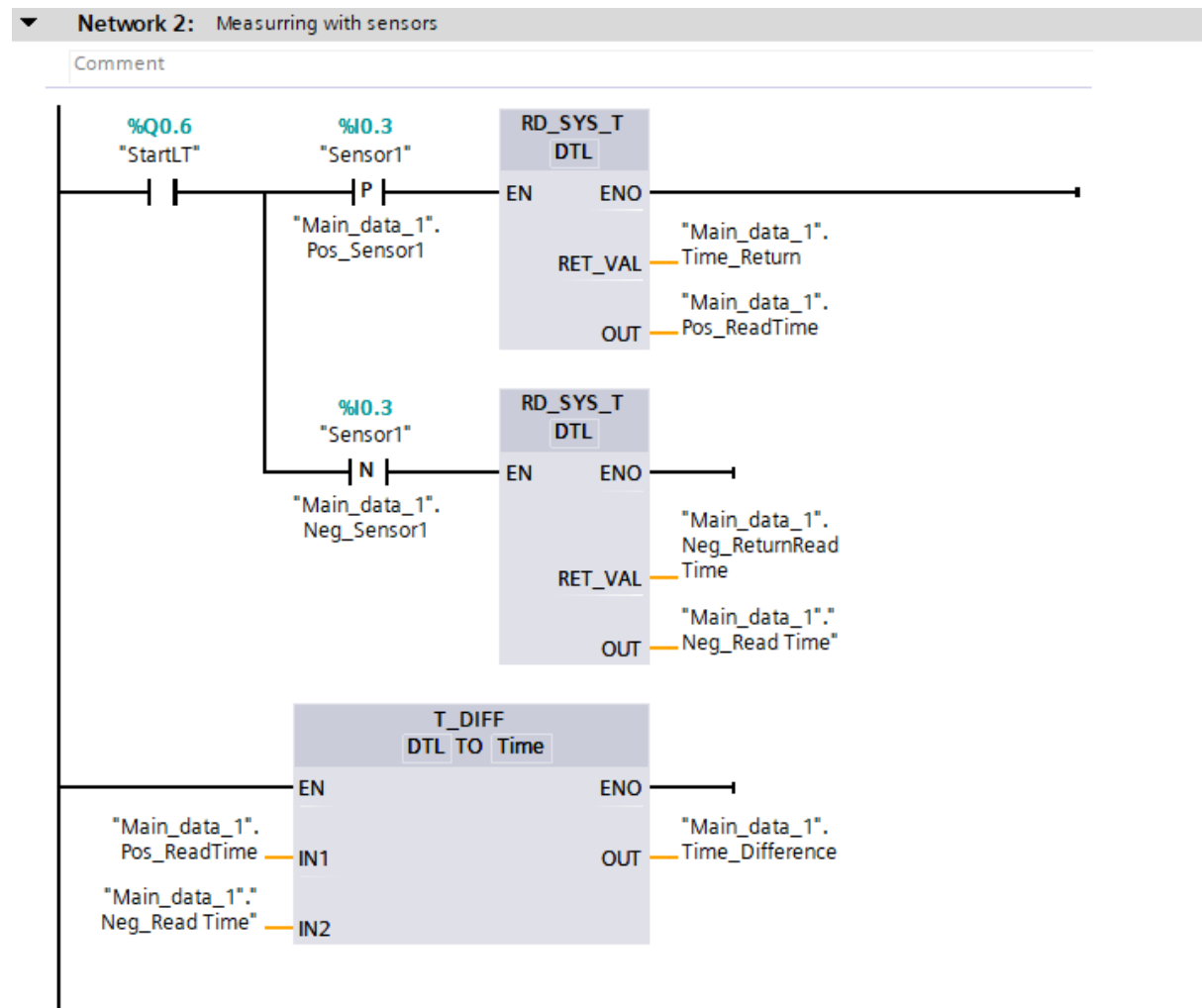
There are two other branches on the network that bypass different sections. The first is that if the conveyor is HIGH then it will bypass the start button needing to be pressed keeping the system on until the stop button break the signal.

The second is a link to the HMI start button that will also bypass the stop button and will only stop if the HMI stop button is pressed.



Network Two

For the second network the sensors will be set up. To do this the system has to be running in order for the start light to be high. This will allow a value to be obtained from the positive and negative edge of the Optical sensor when its beam is broken. The difference in the values will be found and converted to time through the T_DIFF block. This will let the system know the size of the object based on the final time value in “Time_Difference”.



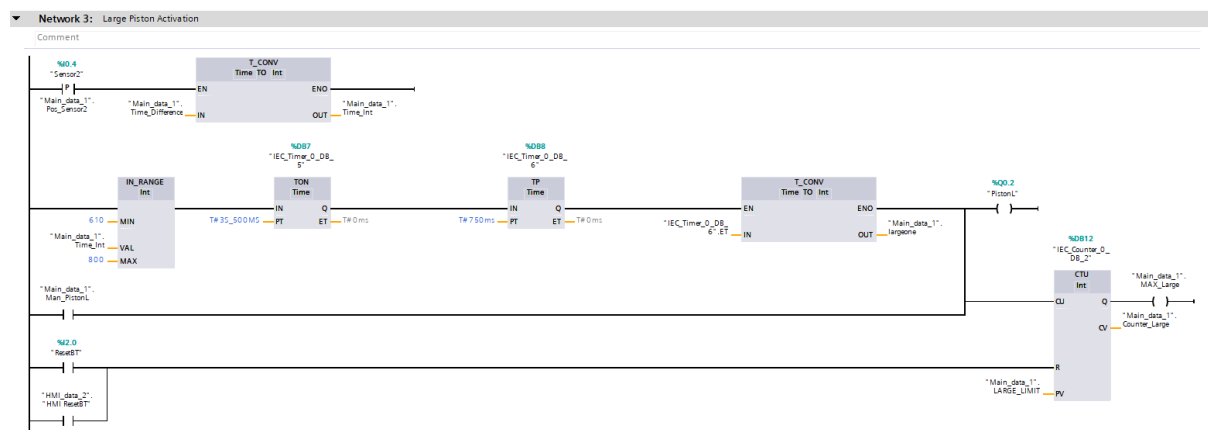
Network Three

Now that the PLC is reading the size of the objects placed through the sensor, the pistons can be setup in order to sort them. The sorting system will become HIGH based on the second optical sensor being broken. Once this has happened the value of the object will be transfoerd from time to an interger value so it can be used.

A range block is used and depending on the value of the interger will set a signal brnch to go high. For network 3 the large piston will go active if the value of the object falls between 610 and 800. This will then allow for timer to start that will wait to set the piston HIGH until the object is in front of it. This can be done as the conveyor moves at a constant speed.

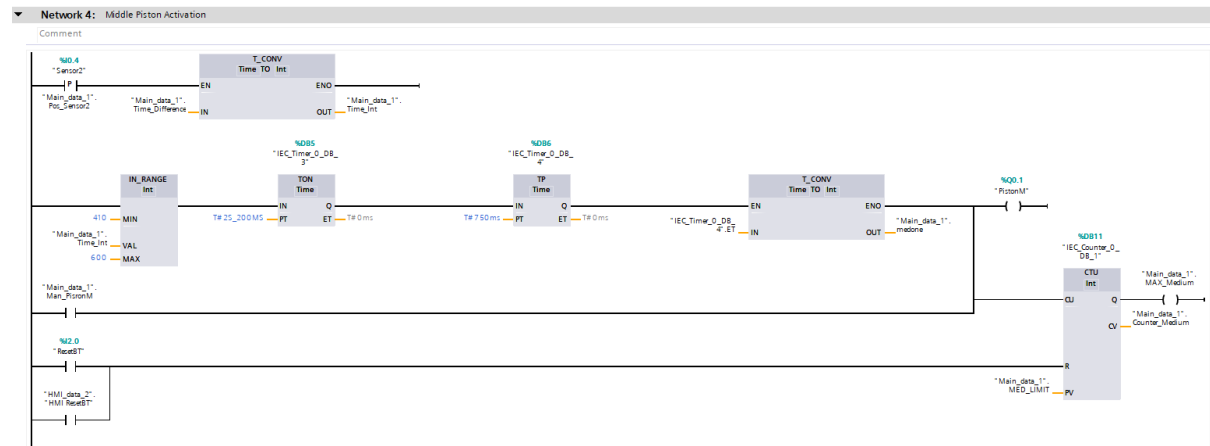
Once the piston has gone high a value of one will be added to the pistons counter.

There are also branches for tags that will be connected to the HMI screen in order to bypass the value reading and manually set the piston HIGH.



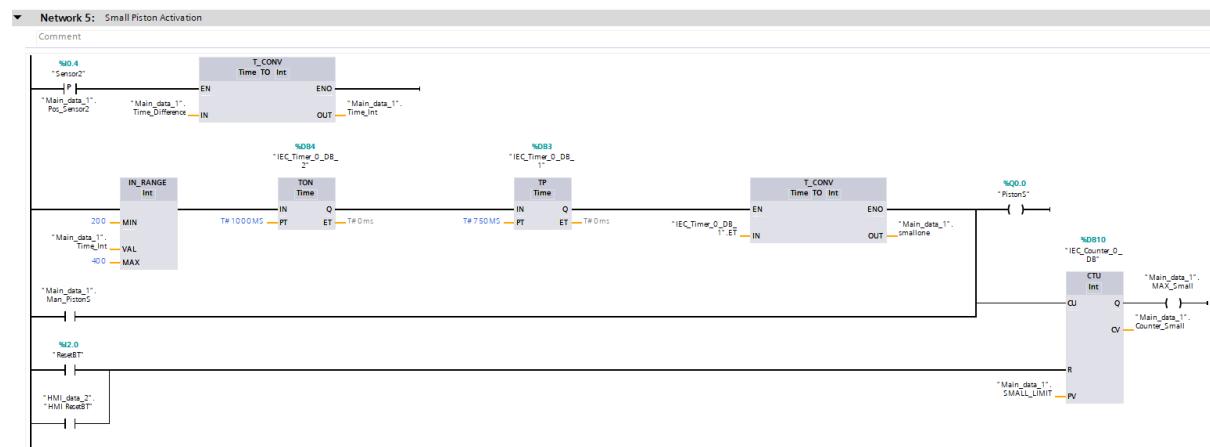
Network Four

Network 4 is the same process however with a range of 410 to 600 and will sort medium sized objects with the middle piston.



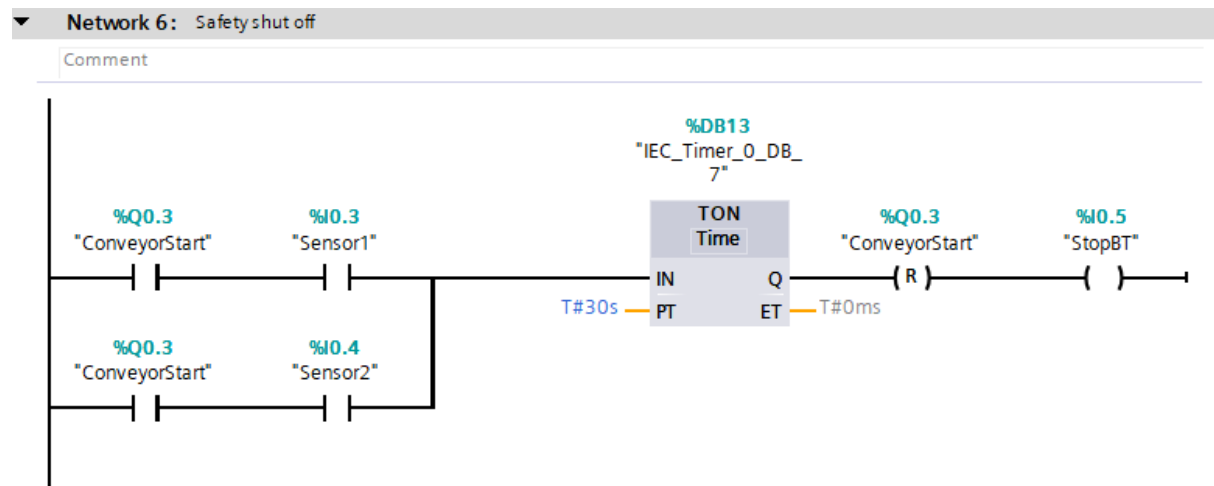
Network Five

Network 5 is the same again but with setting the small piston HIGH when an object falls between the values 200 and 400.



Network Six

The final network on the PLC will be a safety shut off that will become active if no objects are detected for 30 seconds. To do this if either sensor is unbroken a timer block will start and if the timer reaches 30 seconds it will let signal through setting the conveyor to restart and the stop button HIGH.

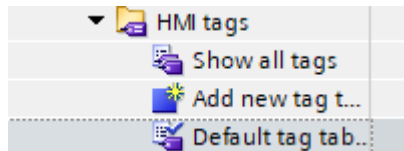


HMI

Setting Tag Tables

The Main PLC is now programmed and set up. The next step is to program the HMI screen.

First the Tag table for the HIM will be setup. The HMI tag table can be found in the “Default tag table” tab under “HMI tags”.



The setup of HMI tags differs to the PLC as the tags must be set to either be internal tags or connected to existing tags within the PLC. This is the reason why the data block was setup.

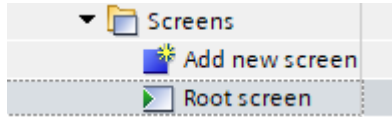
To link them simply select the tag name from under the PLC tag column.

Default tag table								
Name ▲	Data type	Connection	PLC name	PLC tag	Address	Access mode	Acquisition cycle	Source comment
Green Light	Bool	HMI_Connection...	PLC_1	StartLT		<symbolic access>	1 s	
HMI Reset	Bool	HMI_Connection...	PLC_1	HMI_data_2."HMI ResetBT"		<symbolic access>	1 s	
HMI_GreenBT	Bool	HMI_Connection...	PLC_1	HMI_data_2.HMI_GreenBT		<symbolic access>	1 s	
HMI_RedBT	Bool	HMI_Connection...	PLC_1	HMI_data_2.HMI_RedBT		<symbolic access>	1 s	
PistonS	Bool	HMI_Connection...	PLC_1	Main_data_1.Man_PistonS		<symbolic access>	1 s	
PistonM	Bool	HMI_Connection...	PLC_1	Main_data_1.Man_PistonM		<symbolic access>	1 s	
PistonL	Bool	HMI_Connection...	PLC_1	Main_data_1.Man_PistonL		<symbolic access>	1 s	
Large_Counter	Int	HMI_Connection...	PLC_1	Main_data_1.Counter_Le...		<symbolic access>	1 s	
Medium_Counter	Int	HMI_Connection...	PLC_1	Main_data_1.Counter_M...		<symbolic access>	1 s	
Small_Counter	Int	HMI_Connection...	PLC_1	Main_data_1.Counter_S...		<symbolic access>	1 s	
Red Light	Bool	HMI_Conne...	PLC_1	StopLT		<symbolic access>	1 s	
Tag_ScreenNumber	UInt	<internal tag>		<Undefined>			1 s	
<Add new>								

Designing the Screen

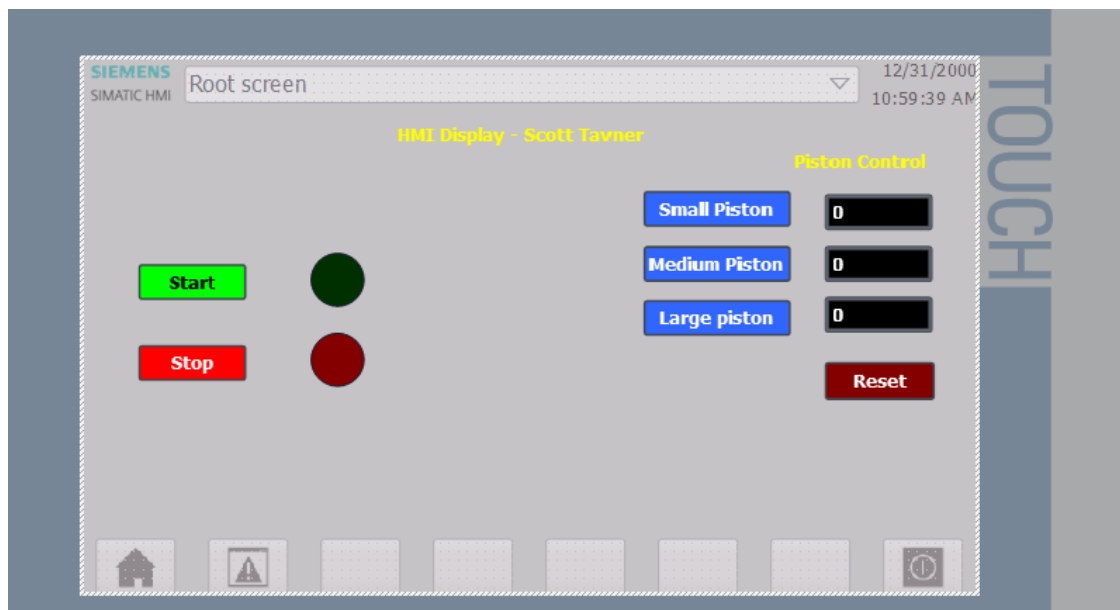
After setting up the tags the screen that will be displayed on the HMI can be designed. For this system the root screen will be modified however new screen scan be added if necessary.

To find the root screen go to “Screens” and then select “Root screen”

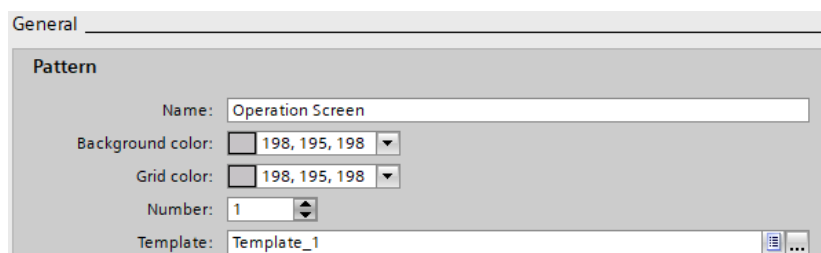


This is the design for the root screen. It features:

- buttons to start and stop the system
- buttons to activate each piston
- a display of the pistons count so the amount of objects sorted is known
- a reset button for the counters
- representation for the start and stop lights



The root screen is set to have a grey background so that things are easily visible but not too bright as to cause any eye pain. To do this simply select the background and grid colour under general tab in properties.

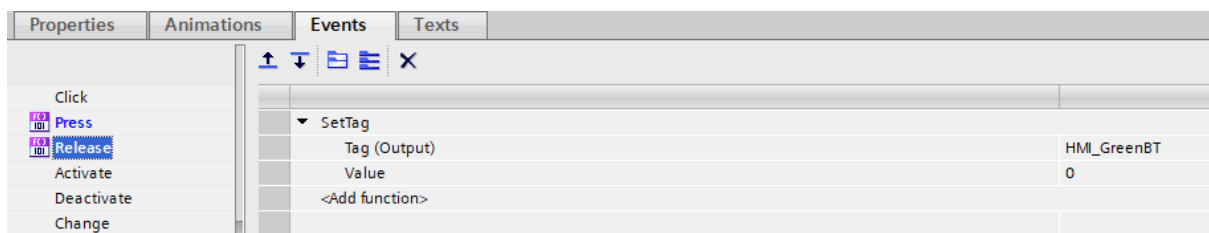
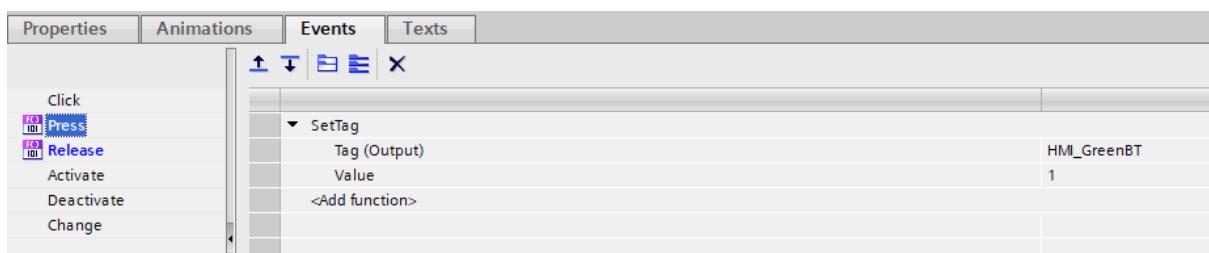


Adding functions to the screen

Now that the screen has been designed the icons displayed need to be set up.

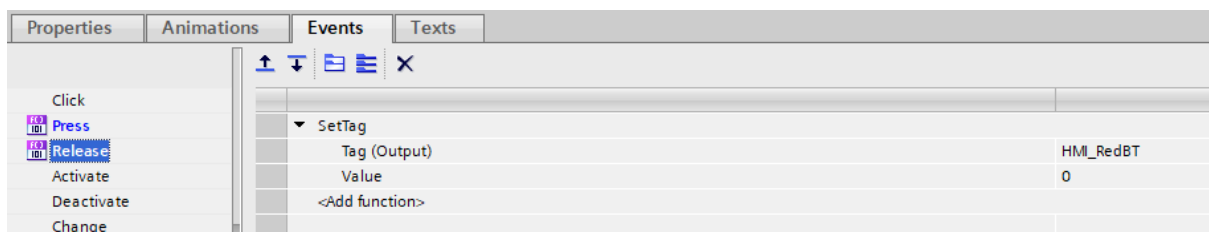
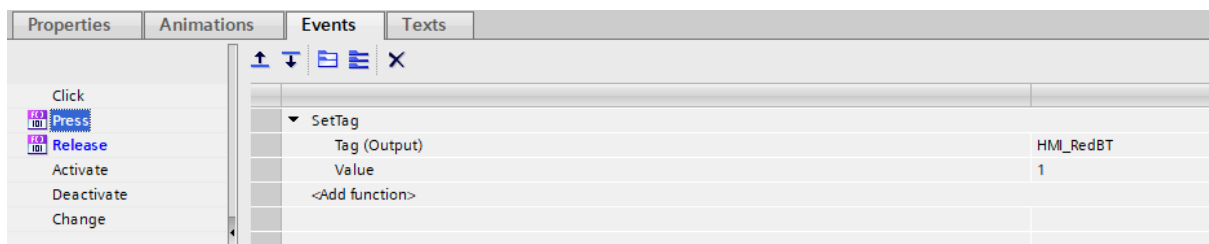
The first is the start button. To do this click on the button and go to the events tab. Here the button can be set up so that if pressed the tag for the Green button will change its state from low to high and once the button is released the value will be set back to 0.

Start button

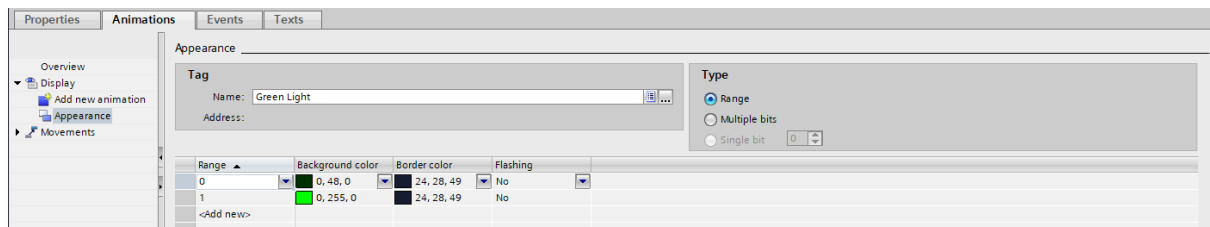


The same process is used for the stop button which is linked to the red button HMI tag.

Stop button

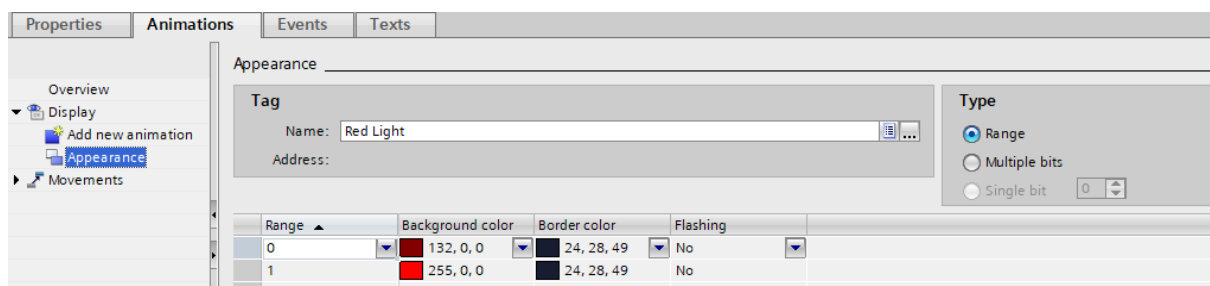


For the display the icon to represent the Green light needs to change colour to indicate when it is physically active. To do this a new animation is given to the icon under the animations tab, this will set the icon to be a dark green colour when the green light tag is a value of 0 and a bright green colour when the value is 1.

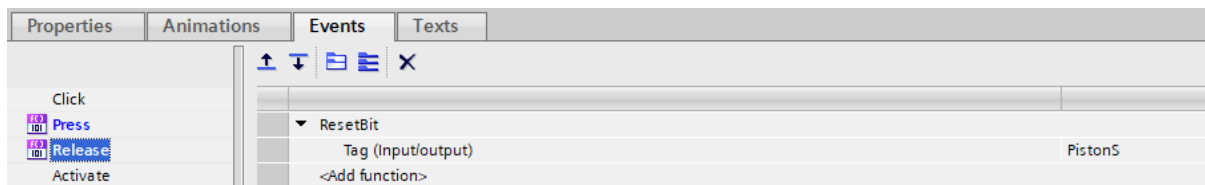


This process is used again for the red-light icon which is linked to the red-light tag.

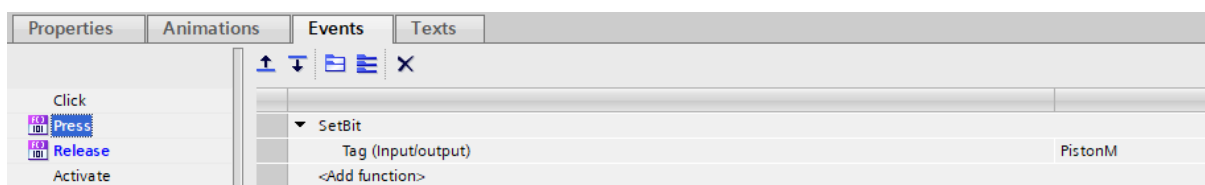
Red light



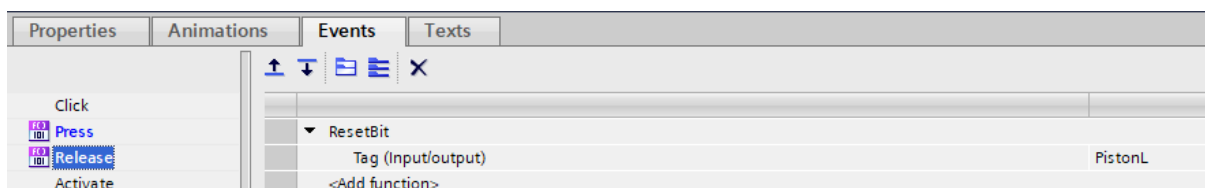
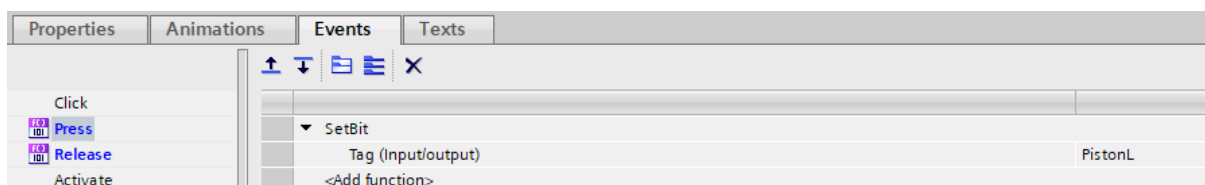
In order to manually control the pistons a tag will be linked to the button through the events tab similarly to the start and stop buttons. The small piston is linked to the PistonS tag.



The medium piston is the same and will be linked to the tag PistonM



And finally, the large piston will be linked to the tag PistonL.



Next the icons to show the values of each piston will be setup. To do this a new animation will be added through tag connections. This animation will allow the counter to display the value of a given tag.

The Small piston is linked to the tag Small_Counter.

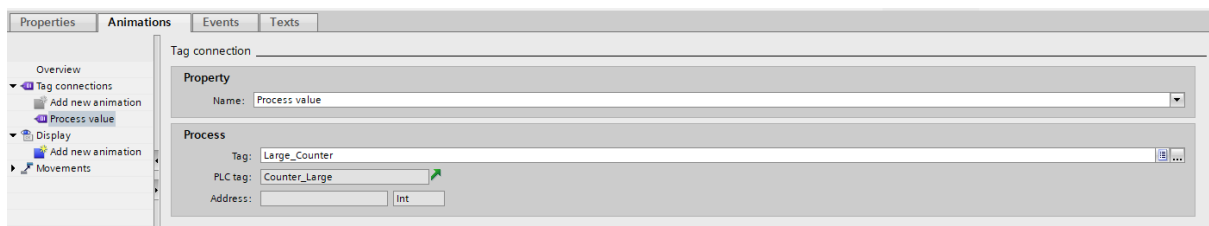
Small piston counter



The Medium piston is linked to the tag Meduim_Counter.

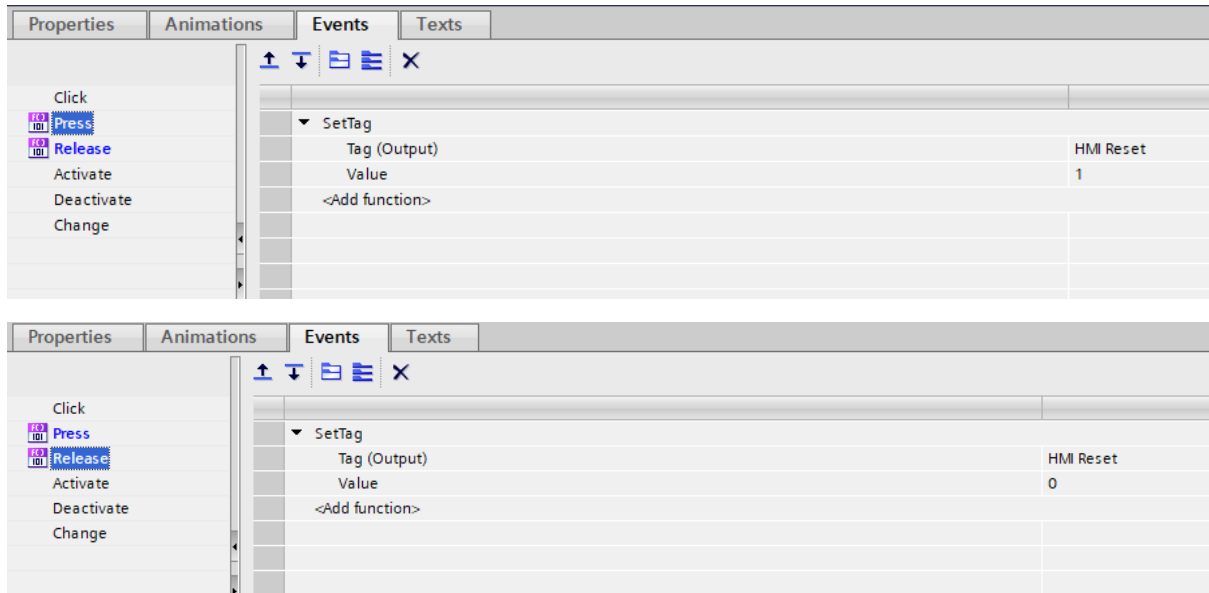


And the Large piston was linked to the Large_Counter tag.



The last thing to set up is the reset button that follows the same process as the other buttons working off a press and release system and is linked to the HMI reset tag.

Reset button

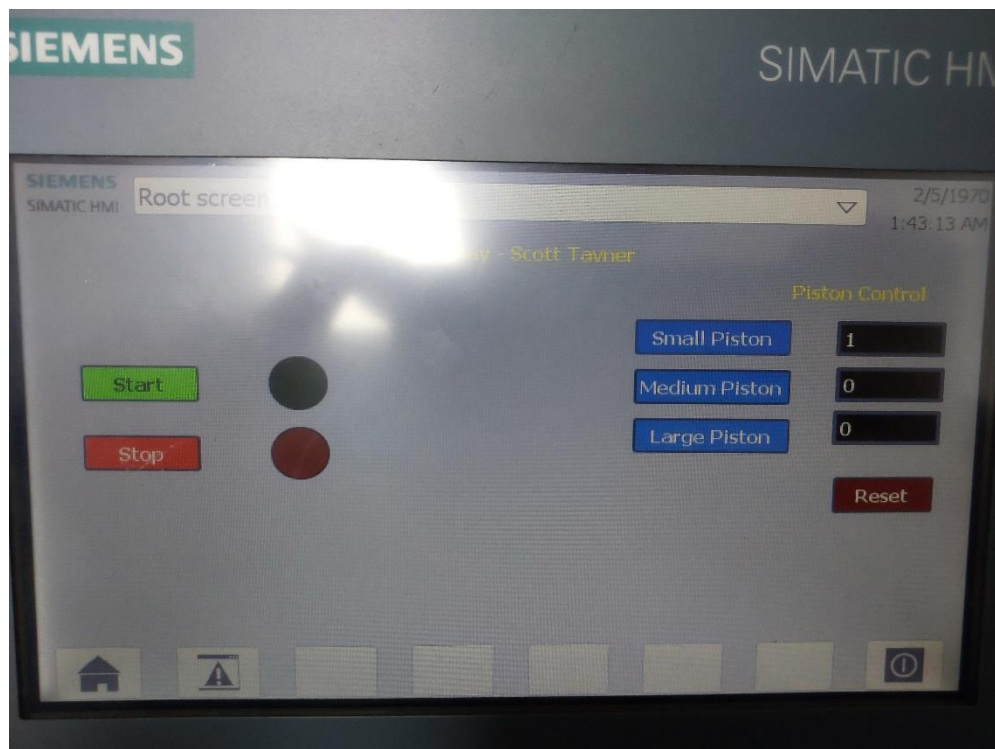


Results

After the code is downloaded to the PLC the system runs as expected with the different sized Pistons sorting as they should. At the time of testing there were no bins in which to sort into however each piston corresponds as desired.



As well as the autonoms part working the HMI screen functioned correctly allowing for manual control of the system.



The function of the start button can be seen in (figure 6 & 7). With the physical green light illuminating, conveyor starting and digital green light illuminating when the start button is pressed.



(figure 6)



(figure 7)

Concussion

Overall the system was a success and was able to achieve all the necessary requirements. Due to the simplicity of this system there are a number of ways in which it could be improved and expand upon. For example, adding more sensors to sort objects based on their vertical height as well or having added conveyor belts to transport sorted objects. There are also many ways that the HMI could be expanded on, adding more information about what is happening within the system or display the measured value of the objects passing through. Whatever the improvement may be, this report has demonstrated a baseline for how to code a simple system which is open in its ability to be expanded upon.

References

Liam, 2022. *Copying is not allowed..* [Online]

Available at: <https://liambee.me/siemens/difference-between-tags-variables-in-siemens-tia-portal/>

[Accessed 26 April 2024].

paessler, n.d.. *What is a PLC?*. [Online]

Available at: <https://www.paessler.com/it-explained/plc>

[Accessed 20 April 2024].

plantautomation-technology, n.d.. *The Impact of PLC Automation Services on Manufacturing.*

[Online]

Available at: <https://www.plantautomation-technology.com/articles/the-impact-of-plc-automation-services-on-manufacturing>

[Accessed 20 April 2024].