

2024423

by Taylor Hoskins

FILE	16348_TAYLOR_HOSKINS_2024423_178848_276306472.PDF (2.93M)		
TIME SUBMITTED	08-FEB-2021 10:26AM (UTC+0000)	WORD COUNT	4565
SUBMISSION ID	144388153	CHARACTER COUNT	22145

Course: HNC Electrical and Electronic Engineering
Module: SGEE4004 Programmable Logic Controllers
Assignment: 1. Pick and Place Robot
Name: Taylor Hoskins
University number: 2024423
Date submitted: 08/02/2021
Word count: 3888 words

Contents

Good Structure

1. Research	3
2. Hardware Investigation, Circuit Diagram and I/O List	5
3. Program Design and Explanation.....	8
4. Simulation Testing	14
5. Practical Testing and Team Working	18
6. Discussion, Conclusions and Recommendations	20
7. Citation List	22
8. Annexe	23

1. Research

Your first sentence is an unnecessary use of words

There are multiple types of PLCs to choose from when looking for the right one for the specific application. Research was carried out of the Siemens Range of PLCs, specifically the S7 range closely, to identify which would be best suited to the Pick and Place Robot. The three main types of PLCs are Unitary, modular and rack mounted.

S7-300 – This PLC was one of the first with a unique modular design, which could be expanded through expansion modules. This type is usually used for small to medium scale applications. It is the upgraded version from the S7-200, with a larger number of I/O and a work memory of 192KB. It is wall mounted. This type is a robust PLC with almost endless expandability. It is versatile and can be used for complex machines. (Areej, 2018) Correctly cited



FIGURE 1

Put your citation before the full stop, otherwise it looks like the citation belongs to the next sentence

S7-400 – This PLC was designed in order to suit the process industry, fast paced. Compared to the S7-300, this type is used for a much larger system and carries more power. It has more advanced features and purpose is for large scale applications. If used for incorrect purpose, the PLC can seem expensive. (Areej, 2018) Correctly cited It is more type specific and does not tend to be a popular choice. Since the new PLC's have been created, the difference between the two have become a lot smaller due to much better conditions.

S7-1200 – This is a new version of the S7-200. It is a micro controller with a limited function and expandability. Its communication protocol is Modbus, and it is only used with TIA Portal software. This PLC type is ideal for automation tasks, with low to medium performance needs. It is simplistic, but precise. It is versatile in terms of automation. Expansion modules can be added if more I/O's are needed for the application. If using CPU 1212C, it only accepts two signal modules, whereas other CPU's can have up to eight.

The S7-1214C is a compact CPU with a speed of 0.08m Seconds and a smaller memory compared to the other types. The main use of this type is for automation, perfect for the application of a pick and place robot. (Siemens, n.d.) Correctly cited



Which one is a unitary PLC?

FIGURE 2

S71500 – This is the latest PLC, based upon its previous types. It is designed to carry out shorter product life cycles and fast market changes. This model has a display screen, meaning basic configuration can be carried out without the use of an external screen. It can be used for all types of applications from small to large scale industry, dependent on its CPU. (Siemens.AG, 2020) Correctly cited

A model of the S7-1500 PLC is the S7-1511C. This type is a high-performance PLC, with maximum user friendliness. It allows for shorter reaction times which increases speed and precision. (Reichelt, n.d.) The main advantage of this type over the earlier produced types, is its performance speed. (siemens, 2014) Correctly cited

Comparison Table

PLC	Type	Digital number of I/O	Memory size	Expandability	Processor Speed	Use	Cost (average)
S7-313C	Modular	24 inputs 16 outputs	4MB	Can be expanded.	0.07m Seconds	Small Scale industry	£1,500
S7-412-4	Rack Mounted	Almost Endless	32MB	Large expandability.	0.03m Seconds	Large scale industry	£3,000
S7-1214C	Modular	14 inputs 10 outputs	100KB	Can be expanded.	0.08m Seconds	Automation Control	£370
S7-1511C	Modular	16 inputs 16 outputs	4MB	Can be expanded.	0.06m Seconds	Automation Control	£600

FIGURE 3

Incorrectly cited, this is not the Harvard Method

Most results were taken from data sheets from Parmley Graham wholesalers' website, for ease of comparison.

After comparing the Siemens types of PLCs, the best options were the S7-1214C and the S7-1511C type. They are both used for Automation Control, with benefits of both. Considering the Pick and Place only needs a small-scale program to move the robot, the S7-1214C would be the best choice, it is cheaper and has enough memory and speed for the intended purpose.

Good conclusion

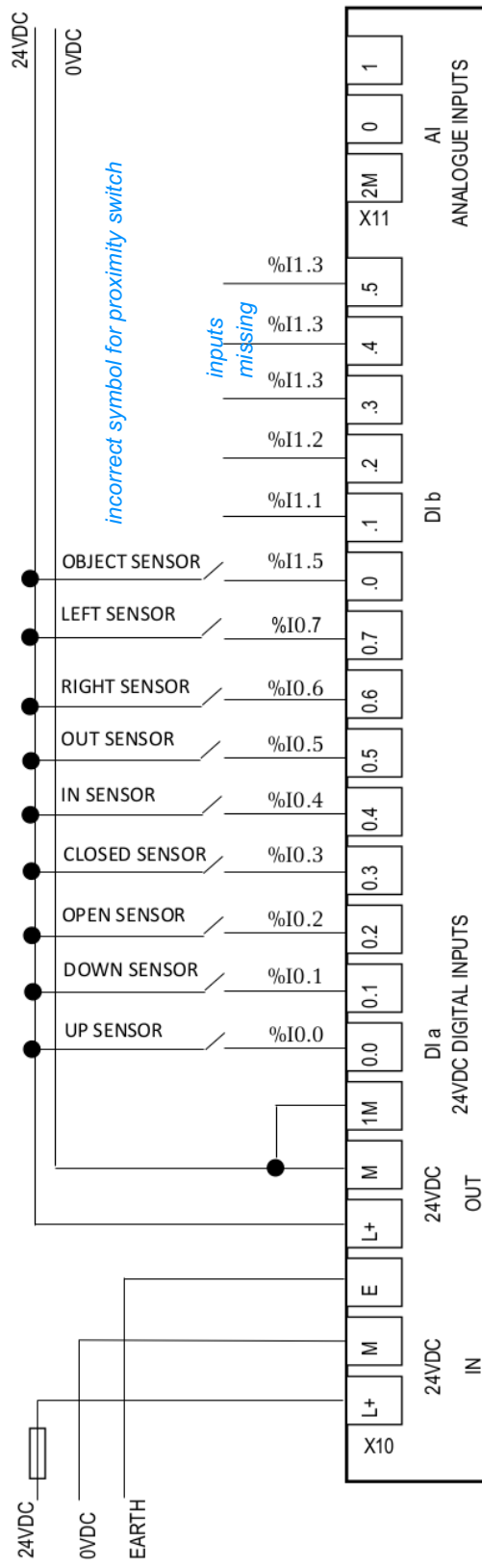
2. Hardware Investigation, Circuit Diagram and I/O List

The hardware used on the pick and place robot is as follows:

- Siemens 1214C PLC
- Robot Object Sensors and Solenoids
- 3 Push buttons
- 2 Rocker Switches
- Entry Chute
- Exit Chute
- Compressed air supplied to solenoid

The PLC is connected to the pick and place robot which will allow the robot to follow the commands sent by the PLC. The object sensor detects the object entering at the entry chute and entering station 1. The solenoid is used to move the robot arm to its desired location. Rocker switches and push buttons are found on the robot to manually control the system. *These switches should be in your drawing and the I/O list*

Good drawing but not all inputs are shown



incorrect symbol for proximity switch

inputs missing

Siemens S71214C - DC/DC/DC 24VDC DIGITAL OUTPUTS

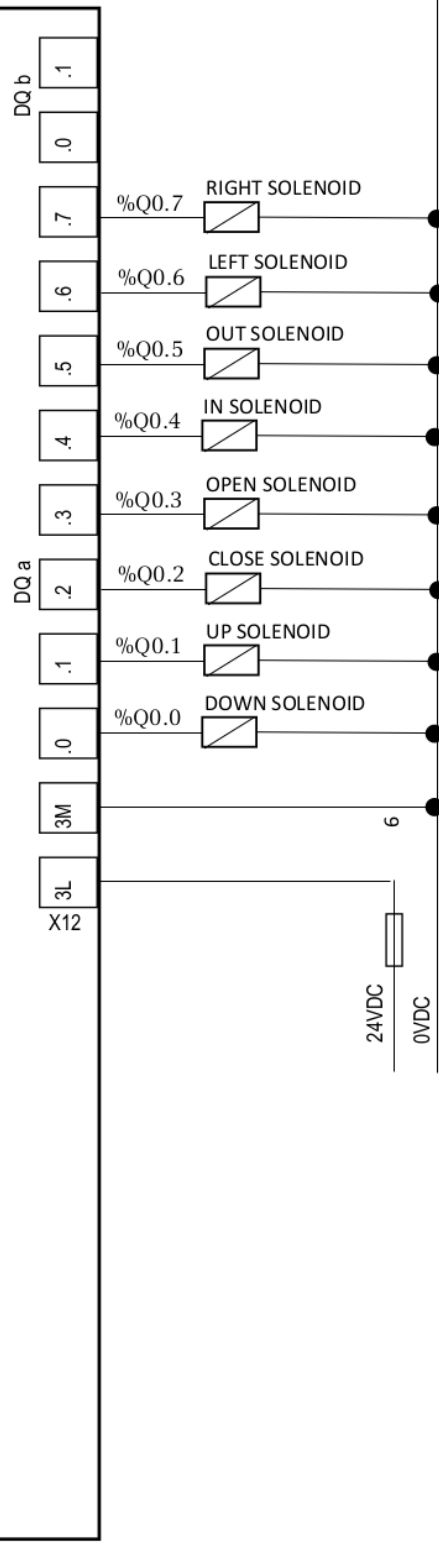


FIGURE 4

The I/O table should actual I addresses not the M addresses to which they are mapped. It should match up with the drawing

An Input/out list has been created as the basis of my tag table for my program in the form of a table.

Tag Name	I/O	Analogue/Digital	Address
UP Sensor	Input	All signals are Digital as they have a binary output. For example, their output is either 1 or 0, meaning that they could be in and on or off state.	%M0.0
Down Sensor	Input		%M0.1
Open Sensor	Input		%M0.2
Closed Sensor	Input		%M0.3
In Sensor	Input		%M0.4
Out Sensor	Input		%M0.5
Right Sensor	Input		%M0.6
Left Sensor	Input		%M0.7
Object Sensor	Input		%M1.5
Down Solenoid	Output		%Q0.0
Up Solenoid	Output		%Q0.1
Closed Solenoid	Output		%Q0.2
Open Solenoid	Output		%Q0.3
In Solenoid	Output		%Q0.4
Out Solenoid	Output		%Q0.5
Left Solenoid	Output		%Q0.6
Right Solenoid	Output		%Q0.7

FIGURE 5

3. Program Design and Explanation

*Don't waste word
giving a history of
how it was made*

The project was created in TIA portal, where the S7-1214C type PLC was chosen as the ideal PLC. A tag table was then able to be started from the original I/O list, listed in section 2, where more tags were added in the process of the program to suit its requirements. AUTO and Manual requests were added, as well as tags such as Start and Pause, for the buttons on the HMI screens to perform actions. The memory tag addresses started from %M2000.0. The full tag table can be found in the Annexe, figures 19-21.

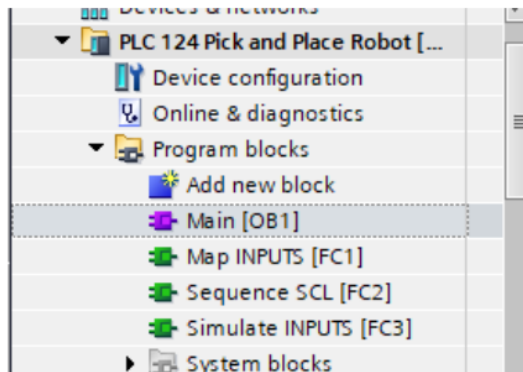


FIGURE 6

*Don't waste word giving a history of how it
was made*

A 'Main' program block was added. This is the block where all the commands were inputted, where the program could either work in manual or auto. It consists of twelve networks all named for ease of read. Each network contains comments about how each work, which can be shown through screenshots shown in the Annexe.

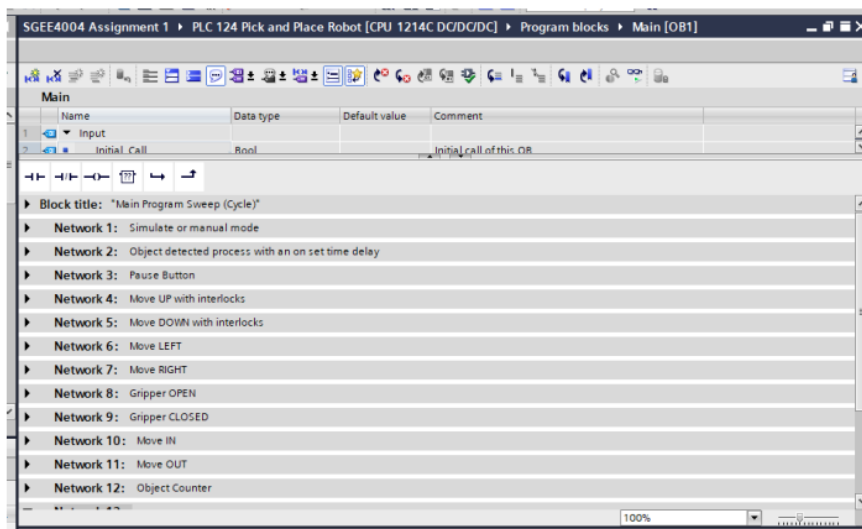


FIGURE 7

Networks 3-11 all contain a normally open 'Auto' contact in series with Auto Request contact, and then in parallel to this, a normally closed 'Auto' contact in series with an HMI request. This is so that when the AUTO switch is

on, the automation requests can work. The manual requests will not work if the system is in Auto mode, only in manual.

Within these networks, interlocks were used so that the robot did not break. When in the UP position, the robot can only move left, right. This meant it only move in free space, not hitting any objects on the conveyor, whereas it would if it were down. A normally open 'UP' contact was added to the left, right, in and out networks so that when the buttons on the engineer screen are pressed, they would only be able to work if the robot were in the UP position.

It could also only move down over stations, for the same reason. This interlock was added to the down network. The left and right sensor normally open contacts are in parallel together, as well as the in and out sensors, shown in the below figure. This means that the robot must be either be at station one, two, three or four in order to move down.

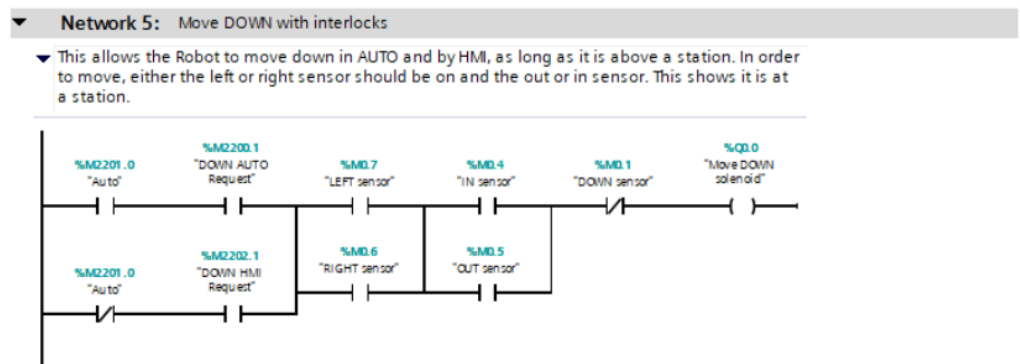


FIGURE 8

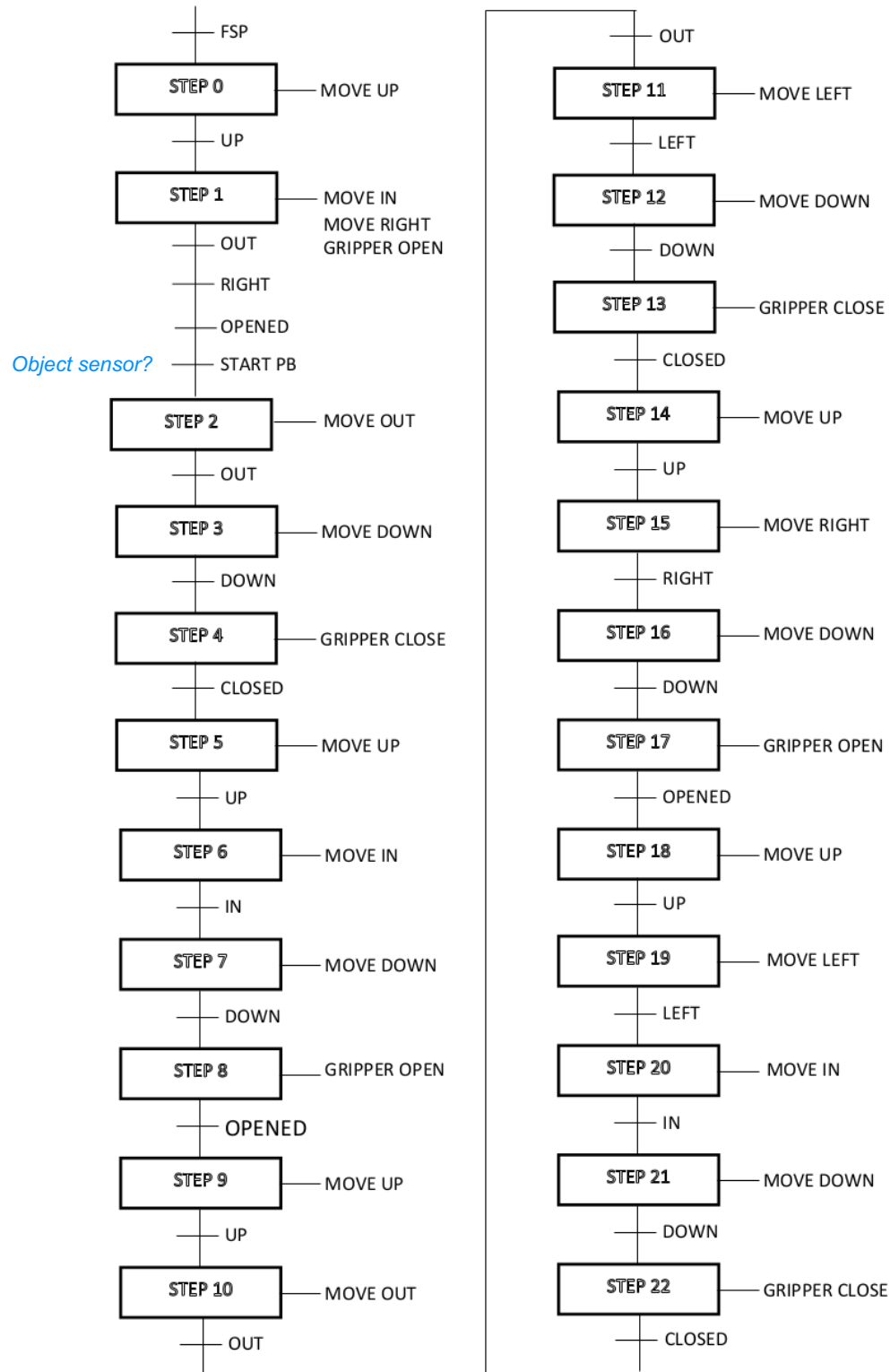
The networks can be found in the Annexe, showing the full networks with comments.

'Map inputs' and 'Simulate inputs' program blocks were already in the program so that the commands within the main block could work.

The 'Sequence SCL' is the Auto sequence block. A sequential Function chart was created in order to follow the Automation Control Requirements. The Auto Sequence allows the robot to:

- Automatically make its way to station 4 from any position when the auto sequence is switched on and wait at the station. This is names 'Step 0.'
- When a counter is added, a two second time delay should be present before it is added into the system. It should be moved to station 2. The counter number should e incremented by 1.
- If there are already 2 counters in the machine, the objects should be moved to the next station before the new one can be added to station 2. E.g., From station 3 to 4, and another counter from station 2 moved to station 3.
- When an object reaches station 4, the counter value will be decremented.

This is a repetition of the brief, unnecessary use of words which has increased your word count for which you have lost 5 marks



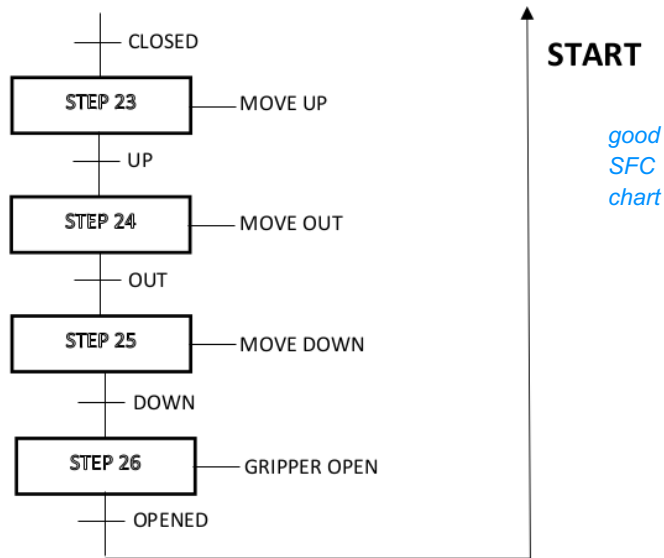


FIGURE 9

Again, you are wasting words telling your reader the story of how you made the program

From this sequential chart, the sequence was written into the program block 'Sequence SCL.' The first part of the SCL is shown in figure 10. This part allows the sequence to reset if the first scan pulse is equal to 1, the auto is equal to 0, or the reset sequence button is pressed.

```

1 IF ("FSP" =1) OR ("Auto"=0) OR ("Reset Sequence" =1) THEN
2   "Step" := 0;
3   "Next_Step" := 0;
4   "LEFT AUTO Request" := 0;
5   "RIGHT AUTO Request" := 0;
6   "IN AUTO Request" := 0;
7   "OUT AUTO Request" := 0;
8   "OPEN AUTO Request" := 0;
9   "CLOSE AUTO Request" := 0;
10 END_IF;
11

```

FIGURE 10

The rest of the sequence follows the SFC. From step 0-26. With an added object counter function, which will allow the sequence to start on a particular step, relating to a station, which is dependent on how many counters are in the system.

- When there is one counter in the system, it will start at step 19. Moving from station 1 to 2.
- Two counters, starts at step 10. Moving from station 2 to 3.
- Three counters, starts at step 2. Moving from station 3 to 4.

Comments are within the sequence, explaining when these steps are reached.

Wasted words telling a story

Two HMI screens were created, one for the purpose of the operator and the purpose of use for the engineer

The purpose of the operator screen was to allow the user to carry out basic commands. This is the root screen when the HMI is simulated. The screen allows the user to see the step the auto sequence is in, number of counters and an auto sequence light. The user can change the sequence to either Auto or manual, start and pause the sequence and reset the number of counters in the system. The user should not be able to do anything that could potentially break the robot. The screen is shown below.

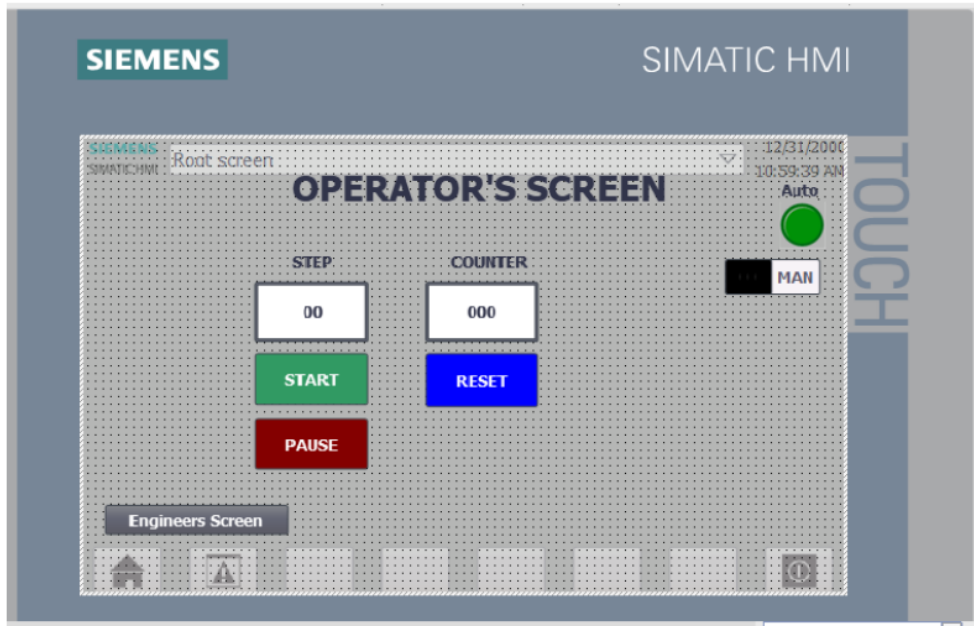


FIGURE 11

The purpose of the engineer's screen is to be able to move the robot safely in manual mode and test the automation sequence. The screen allows the user to do everything the Operator screen can, as well as performing tests within the specific modes. Light sensors were added to show the state of each sensor and solenoid. It allows the engineer to use simulation mode, with a simulation switch, to enter commands such as 'LEFT'. A reset step button was created on the engineer's screen, so the sequence can be reset if put into manual. If the program interlocks are not set up correctly, the engineer could damage the robot.

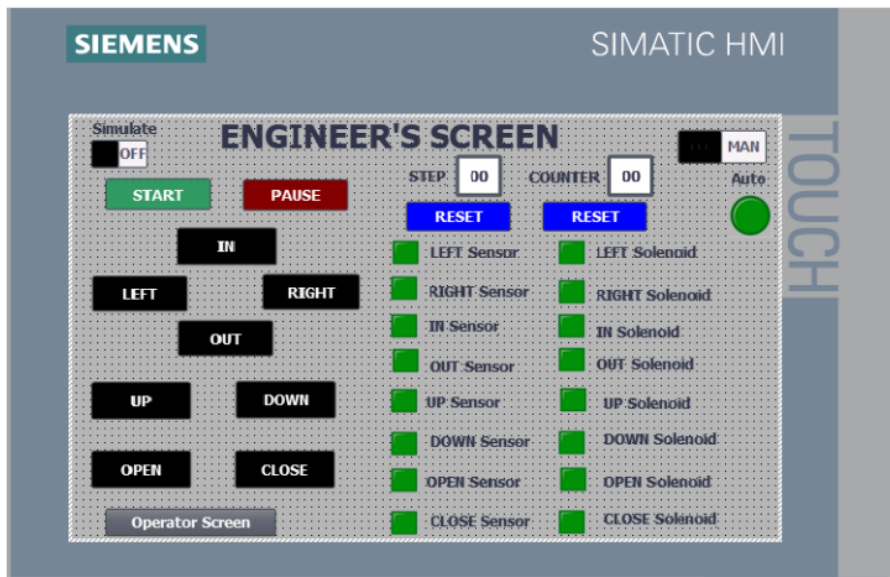


FIGURE 12

4. Simulation Testing

The PLC and HMI were loaded up into simulation mode. The program was tested to show that all requirements were met. To do this, the engineering screen was loaded up and a table was created to fully test every aspect of the program.

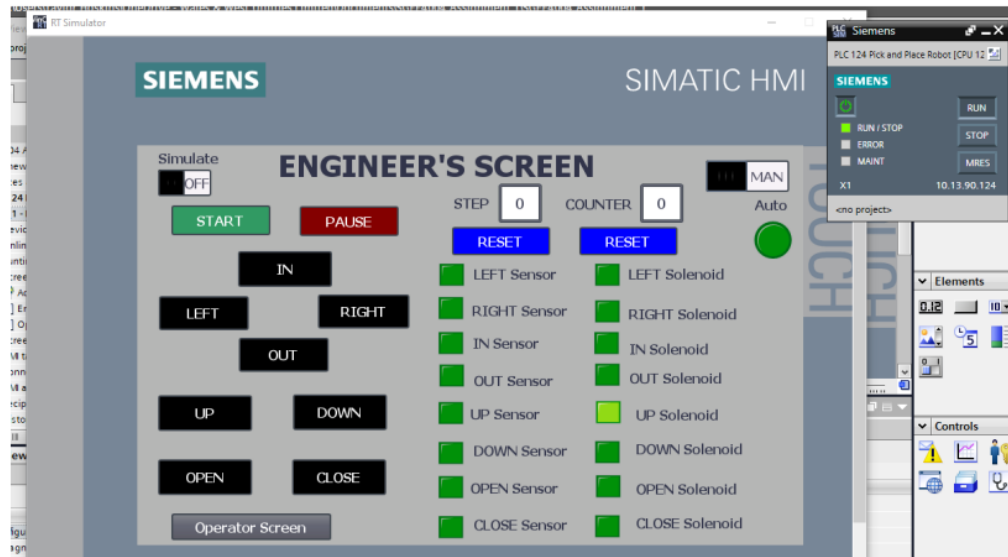


FIGURE 13

The first test was manual mode. In order to prevent damage to the robot, interlocks were put in place so that the robot could not move in the down position, previously described in task 3. Figure 14 shows a table of all the interlocks specified and whether they were met. [A good table](#)

Test no.	Description	Expected Result	Actual Result	PASS/FAIL
1	Move Left in UP position	Free Movement	Free Movement	PASS
2	Move Left in DOWN position	No Movement	No Movement	PASS
3	Move Right in UP position	Free Movement	Free Movement	PASS
4	Move Right in DOWN position	No Movement	No Movement	PASS
5	Move in in UP position	Free Movement	Free Movement	PASS
6	Move in in DOWN position	No Movement	No Movement	PASS
7	Move out in UP position	Free Movement	Free Movement	PASS
8	Move out in DOWN position	No Movement	No Movement	PASS

[you could have linked each test to the evidence image in the appendix](#)

9	Move Down at Station 1 (LEFT and IN)	Free Movement	Free Movement	PASS
10	Move Down at Station 2 (LEFT and OUT)	Free Movement	Free Movement	PASS
11	Move Down at Station 3 (RIGHT and OUT)	Free Movement	Free Movement	PASS
12	Move Down at Station 4 (RIGHT and IN)	Free Movement	Free Movement	PASS
13	Move Down when the robot is only left	No Movement	No Movement	PASS
14	Move Down when the robot is only right	No Movement	No Movement	PASS
15	Move Down when the robot is only in	No Movement	No Movement	PASS
16	Move Down when the robot is only out	No Movement	No Movement	PASS
17	Move down when the robot is only up (Central)	No Movement	No Movement	PASS

FIGURE 14

Monitor mode within the main block was opened in order to prove these interlocks. When trying to move down, not over a station, network 5 showed the Move down solenoid was not able to be lit unless it was left or right, and either in or out. Figure 15 shows when the down button is pressed, not over a station, it is unable to move. Screenshots of evidence in monitor mode for other interlocks are shown in the Annexe.

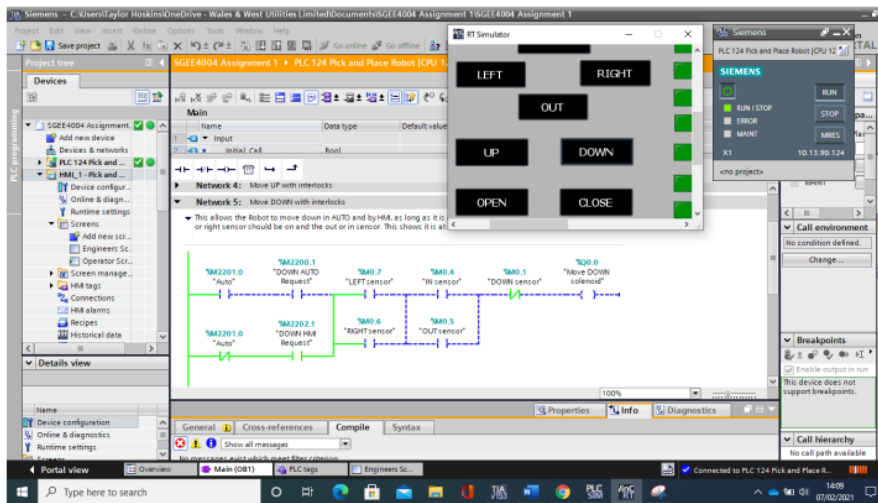


FIGURE 15

The Auto Sequence was also tested. Figure 16 shows the condition of each step relating to the Sequential Function chart that was created. The SCL Sequence was written following the SFC, which should allow the actual results in the below sequence to have the same result as the expected, showing on the HMI, when the program is in auto sequence mode and the start button is pressed.

Step no.	Expected Result	Actual Result	PASS/FAIL
0	MOVE UP	MOVE UP	PASS
1	MOVE IN MOVE RIGHT GRIPPER OPEN	MOVE IN MOVE RIGHT GRIPPER OPEN	PASS
2	MOVE OUT	MOVE OUT	PASS
3	MOVE DOWN	MOVE DOWN	PASS
4	GRIPPER CLOSE	GRIPPER CLOSE	PASS
5	MOVE UP	MOVE UP	PASS
6	MOVE IN	MOVE IN	PASS
7	MOVE DOWN	MOVE DOWN	PASS
8	GRIPPER OPEN	GRIPPER OPEN	PASS
9	MOVE UP	MOVE UP	PASS
10	MOVE OUT	MOVE OUT	PASS
11	MOVE LEFT	MOVE LEFT	PASS
12	MOVE DOWN	MOVE DOWN	PASS
13	GRIPPER CLOSE	GRIPPER CLOSE	PASS

14	MOVE UP	MOVE UP	PASS
15	MOVE RIGHT	MOVE RIGHT	PASS
16	MOVE DOWN	MOVE DOWN	PASS
17	GRIPPER OPEN	GRIPPER OPEN	PASS
18	MOVE UP	MOVE UP	PASS
19	MOVE LEFT	MOVE LEFT	PASS
20	MOVE IN	MOVE IN	PASS
21	MOVE DOWN	MOVE DOWN	PASS
22	GRIPPER CLOSE	GRIPPER CLOSE	PASS
23	MOVE UP	MOVE UP	PASS
24	MOVE OUT	MOVE OUT	PASS
25	MOVE DOWN	MOVE DOWN	PASS
26	GRIPPER OPEN	GRIPPER OPEN	PASS

FIGURE 16

When a new counter is put into the system, the sequence should jump to the corresponding step number. It was put in the program that there can be a maximum of 3 counters in the robot at any time. Once step 5 is reached, the number of counters should go down from 3 to 2, indicating that one counter has left the area, as it has made it to the end of the cycle. Evidence of this can be shown in the annexe.

Counter Number	Expected Initial Step	Actual Initial Step	PASS/FAIL
1	19	19	PASS
2	10	10	PASS
3	2	2	PASS

FIGURE 17

5. Practical Testing and Team Working

Use the third

Due to the Current pandemic, the practical work was unable to be carried out within the college. We were also unable to connect onto the VPN and test remotely due to Rob needing access into the college. Instead, separate groups of three were made where we each went through our programs to show the other group members and gave each other a different perspective on how we could improve our programme.

Use the third

Person

We took it in turns to first go through the tag list, then into the main block to show each rung. Within these rungs we went into detail on how we achieved the specification, creating interlocks and explaining how each one worked. We also showed the Auto Sequence block. Although the sequences were slightly different, the outputs of them all were the same, which did not matter if we proved they could work. For example, step 10 and 11 could have been switched. E.g., Out then left.

Use the third

The HMI screen was the loaded up. It was interesting to see how other students laid out their HMI screen. We all interpreted the brief differently, mainly the way we presented our HMI buttons and sensors.

Use the third

Person

We then had to prove that these worked, by loading up the PLC. We compared results tables and methods to create these, such as a table on the document or through excel. We decided to just make a quick list following the brief in order to prove the results and we took it in turns to load our HMI up to show this. The interlocks were tested first in simulation and manual mode, to ensure the robot could not be broken. I have included my table from section 4, to show the tests that were carried out. Then we each showed our Auto Sequence. Allowing the steps to count when a counter was added each time. The Auto Sequence testing can be shown in section 4, figure 16.

Test no.	Description	Expected Result	Actual Result	PASS/FAIL
1	Move Left in UP position	Free Movement	Free Movement	PASS
2	Move Left in DOWN position	No Movement	No Movement	PASS
3	Move Right in UP position	Free Movement	Free Movement	PASS
4	Move Right in DOWN position	No Movement	No Movement	PASS
5	Move in in UP position	Free Movement	Free Movement	PASS
6	Move in in DOWN position	No Movement	No Movement	PASS
7	Move out in UP position	Free Movement	Free Movement	PASS
8	Move out in DOWN position	No Movement	No Movement	PASS
9	Move Down at Station 1 (LEFT and IN)	Free Movement	Free Movement	PASS
10	Move Down at Station 2	Free Movement	Free Movement	PASS

	(LEFT and OUT)			
11	Move Down at Station 3 (RIGHT and OUT)	Free Movement	Free Movement	PASS
12	Move Down at Station 4 (RIGHT and IN)	Free Movement	Free Movement	PASS
13	Move Down when the robot is only left	No Movement	No Movement	PASS
14	Move Down when the robot is only right	No Movement	No Movement	PASS
15	Move Down when the robot is only in	No Movement	No Movement	PASS
16	Move Down when the robot is only out	No Movement	No Movement	PASS
17	Move down when the robot is only up (Central)	No Movement	No Movement	PASS

FIGURE 18

Individuals tasks could not be given out for this section any longer due to the current situation. Instead, we all participated in every aspect and explained extensively about our program to make sure that we each fully understood it and could ask questions if we were unsure. At the end of the testing, a group discussion was held where recommendations were made.

6. Discussion, Conclusions and Recommendations

Due to the current pandemic, the program was unable to be run and ^{Grocer's apostrophe} tested fully on the Pick and Place robot within the college. The aim was to download the best program to one of the PLC's at college, which could then connect to the system and the full manual and automation sequences could be tested.

Instead of the practical task, breakout rooms were held online where we were able to share our screens and show our programs in depth working in simulation and monitor mode on TIA portal.

In Relation to the specification, the program achieved everything that it needed to in terms of simulation meeting all the customer and user needs. All manual tests, such as the interlocks, program, and automation tests passed their requirements, shown in figures 14, 16 and 17, with evidence in the Annexe of the working HMI screen.

All tags were added and addressed appropriately relating to an input or output. Within each program block, comments were added to explain each network and how the interlocks work. The comments also explain what to expect from the Pick and place Robot at that step. For example, at step 19, the robot should be moving the object from station 1 over to station 2. Two HMI screens were also created, one operator and one engineer, where modes could be switched from auto and manual. All sensor and solenoid information were visible through LED lights on the engineer screen.

Although it was not practical tested, the simulation was confidently working with no errors found after extensive testing.

Use the third

At the end of the group task, we had a team discussion comparing each of our programmes. We came to an agreement that all our programmes had met the brief, although presented differently. Recommendations were made to improve each program. Some recommendations made were:

- Laying out buttons in different format to improve ease of use.
- Laying out sensor lights next to each button for ease of read.
- Add more comments to each rung so that users can understand exactly the purpose of each.
- Create testing table within the word document.
- Whether the step reset button fulfils its purpose.

Use the third

Person 1

We decided that had we have gone into the college to carry out the practical testing, that Ben's program would be used due to in ^{detail comments and} ease of use of the HMI screen.

Use the third

Recommendations I would give myself are to add more extensive comments to all the networks and auto sequence. Even though the comments are made, first time users could be more considered when writing them. Also, comments for the HMI screen should be considered. There could be explanations on the buttons and each of their uses, for example the purpose of the reset sequence counter could be made clearer.

Currently, the user can access all part of the HMI screens, the operator screen and the Engineers screen. Adding password protection to engineer screen would make the program a lot safer for the user. This would only allow

access for the intended purpose, as only the Engineer should know the screen password. Access would be denied to the those who did not have the authority.

The layout of the program is easy to read and navigate and the intended user should not have any problems using it.

7. Citation List

Correct method of

Good range of

Anon., n.d. *Parmley Graham*. [Online]

Available at: <https://www.parmley-graham.co.uk/automation/siemens/plc-s/s7-300-advanced-controllers/6ES7313-5BG04-0AB0>

[Accessed 12th December 2020].

Anon., n.d. *Parmley Graham*. [Online]

Available at: <https://www.parmley-graham.co.uk/automation/siemens/plc-s/s7-400-advanced-controllers/6ES7412-2XK07-0AB0>

[Accessed 12th December 2020].

Anon., n.d. *Parmley Graham*. [Online]

Available at: <https://www.parmley-graham.co.uk/automation/siemens/plc-s/s7-1500-advanced-controllers/6ES7511-1AK02-0AB0>

[Accessed 20th January 2021].

Areej, 2018. *Automation Forum*. [Online]

Available at: <https://automationforum.co/introduction-siemens-plc/>

[Accessed 20th December 2020].

Automation24 GmbH, n.d. *Automation24*. [Online]

Available at: <https://www.automation24.co.uk/siemens-cpu-1214c-6es7214-1ag40-0xb0>

[Accessed 15th December 2020].

Reichelt, n.d. *Reichelt*. [Online]

Available at: https://www.reichelt.com/gb/en/s7-1500-starter-kit-cpu-1511c-1-pn-s7-1500c-starter-p189322.html?CTYPE=0&MWSTFREE=0&PROVID=2788&wt_guka=54951063835_263550072692&PROVID=2788&gclid=EAlaIqobChMlo72k_snT3QIVUYGyCh3VGQIGEAkYDCABEgJSZ_D_BwE&r=1

[Accessed January 2021].

RS, n.d. *RS Online*. [Online]

Available at: <https://uk.rs-online.com/web/p/plcs-programmable-logic-controllers/7877989/>

Siemens AG, 2014. *PLC4Good*. [Online]

Available at:

http://plc4good.org.ua/files/02_materials/webinar/s7_1500_compare_table_int_memo_en_US.pdf

[Accessed 5th February 2021].

Siemens AG, 2020. *Siemens*. [Online]

Available at: <https://assets.new.siemens.com/siemens/assets/api/uuid:5e335011-0a8e-4377-8ed6-f9edb19844a2/simatic-controller-application-overview.pdf>

[Accessed 8th January 2021].

Siemens AG, 2020. *Siemens*. [Online]

Available at: <https://assets.new.siemens.com/siemens/assets/api/uuid:5e335011-0a8e-4377-8ed6-f9edb19844a2/simatic-controller-application-overview.pdf>

[Accessed 4th January 2021].

siemens, 2014. *Reichelet Elektronik*. [Online]

Available at: https://www.reichelt.com/gb/en/s7-1500-starter-kit-cpu-1511c-1-pn-s7-1500c-starter-p189322.html?CTYPE=0&MWSTFREE=0&PROVID=2788&wt_guka=54951063835_263550072692&PROVID=2788&gclid=EAlaIqobChMlo72k_snT3QIVUYGyCh3VGQIGEAkYDCABEgJSZ_D_BwE&r=1

Siemens, n.d. *Parmley Graham*. [Online]

Available at: <https://www.parmley-graham.co.uk/automation/siemens/plc-s/s7-1200-basic->

controllers/6ES7214-1HG40-0XB0
[Accessed 1st February 2021].

8. Annexe

SGEE4004 Assignment 1 ▶ PLC 124 Pick and Place Robot [CPU 1214C DC/DC/DC] ▶ PLC tags

Tags User constants System constants

PLC tags

	Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	UP sensor	Simulation Tags	Bool	%M0.0	☐	☒	☒	☒	
2	DOWN sensor	Simulation Tags	Bool	%M0.1	☐	☒	☒	☒	
3	OPEN sensor	Simulation Tags	Bool	%M0.2	☐	☒	☒	☒	
4	CLOSED sensor	Simulation Tags	Bool	%M0.3	☐	☒	☒	☒	
5	IN sensor	Simulation Tags	Bool	%M0.4	☐	☒	☒	☒	
6	OUT sensor	Simulation Tags	Bool	%M0.5	☐	☒	☒	☒	
7	RIGHT sensor	Simulation Tags	Bool	%M0.6	☐	☒	☒	☒	
8	LEFT sensor	Simulation Tags	Bool	%M0.7	☐	☒	☒	☒	
9	X10 Pushbutton	Simulation Tags	Bool	%M1.0	☐	☒	☒	☒	
10	X11 Pushbutton	Simulation Tags	Bool	%M1.1	☐	☒	☒	☒	
11	X12 Pushbutton	Simulation Tags	Bool	%M1.2	☐	☒	☒	☒	
12	X13 toggle switch	Simulation Tags	Bool	%M1.3	☐	☒	☒	☒	
13	X14 toggle switch	Simulation Tags	Bool	%M1.4	☐	☒	☒	☒	
14	OBJECT sensor	Simulation Tags	Bool	%M1.5	☐	☒	☒	☒	
15	Simulate	Assignment tag table	Bool	%M2000.0	☐	☒	☒	☒	
16	PLC Base Unit input byte 0	Assignment tag table	Byte	%IB0	☐	☒	☒	☒	
17	PLC Base Unit mapped input by..	Assignment tag table	Byte	%MB0	☐	☒	☒	☒	
18	PLC Base Unit input byte 1	Assignment tag table	Byte	%IB1	☐	☒	☒	☒	
19	PLC Base Unit mapped input by..	Assignment tag table	Byte	%MB1	☐	☒	☒	☒	
20	System_Byte	System Tags	Byte	%MB2100	☐	☒	☒	☒	
21	FSP	System Tags	Bool	%M2100.0	☐	☒	☒	☒	
22	DiagStatus:Update	System Tags	Bool	%M2100.1	☐	☒	☒	☒	
23	Always TRUE	System Tags	Bool	%M2100.2	☐	☒	☒	☒	

FIGURE 19

	Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
24	Always FALSE	System Tags	Bool	%M2100.3	☐	☒	☒	☒	
25	Clock_Byte	System Tags	Byte	%MB2101	☐	☒	☒	☒	
26	Clock_10Hz	System Tags	Bool	%M2101.0	☐	☒	☒	☒	
27	Clock_5Hz	System Tags	Bool	%M2101.1	☐	☒	☒	☒	
28	Clock_2.5Hz	System Tags	Bool	%M2101.2	☐	☒	☒	☒	
29	Clock_2Hz	System Tags	Bool	%M2101.3	☐	☒	☒	☒	
30	Clock_1.25Hz	System Tags	Bool	%M2101.4	☐	☒	☒	☒	
31	Clock_1Hz	System Tags	Bool	%M2101.5	☐	☒	☒	☒	
32	Clock_0.625Hz	System Tags	Bool	%M2101.6	☐	☒	☒	☒	
33	Clock_0.5Hz	System Tags	Bool	%M2101.7	☐	☒	☒	☒	
34	Move UP solenoid	Assignment tag table	Bool	%Q0.1	☐	☒	☒	☒	
35	Move DOWN solenoid	Assignment tag table	Bool	%Q0.0	☐	☒	☒	☒	
36	Move LEFT solenoid	Assignment tag table	Bool	%Q0.6	☐	☒	☒	☒	
37	Move RIGHT solenoid	Assignment tag table	Bool	%Q0.7	☐	☒	☒	☒	
38	Move IN solenoid	Assignment tag table	Bool	%Q0.5	☐	☒	☒	☒	
39	Move OUT solenoid	Assignment tag table	Bool	%Q0.4	☐	☒	☒	☒	
40	Gripper OPEN solenoid	Assignment tag table	Bool	%Q0.3	☐	☒	☒	☒	
41	Gripper CLOSE solenoid	Assignment tag table	Bool	%Q0.2	☐	☒	☒	☒	
42	Step	Assignment tag table	UInt	%MW2010	☐	☒	☒	☒	
43	Next_Step	Assignment tag table	UInt	%MW2012	☐	☒	☒	☒	
44	UP AUTO Request	Assignment tag table	Bool	%M2200.0	☐	☒	☒	☒	
45	DOWN AUTO Request	Assignment tag table	Bool	%M2200.1	☐	☒	☒	☒	
46	LEFT AUTO Request	Assignment tag table	Bool	%M2200.2	☐	☒	☒	☒	

FIGURE 20

46	LEFT AUTO Request	Assignment tag table Bool	%M2200.2	☐	☒	☒	☒
47	RIGHT AUTO Request	Assignment tag table Bool	%M2200.3	☐	☒	☒	☒
48	IN AUTO Request	Assignment tag table Bool	%M2200.4	☐	☒	☒	☒
49	OUT AUTO Request	Assignment tag table Bool	%M2200.5	☐	☒	☒	☒
50	OPEN AUTO Request	Assignment tag table Bool	%M2200.6	☐	☒	☒	☒
51	CLOSE AUTO Request	Assignment tag table Bool	%M2200.7	☐	☒	☒	☒
52	Auto	Assignment tag table Bool	%M2201.0	☐	☒	☒	☒
53	Start pb	Assignment tag table Bool	%M2201.1	☐	☒	☒	☒
54	UP HMI Request	Assignment tag table Bool	%M2202.0	☐	☒	☒	☒
55	DOWN HMI Request	Assignment tag table Bool	%M2202.1	☐	☒	☒	☒
56	LEFT HMI Request	Assignment tag table Bool	%M2202.2	☐	☒	☒	☒
57	RIGHT HMI Request	Assignment tag table Bool	%M2202.3	☐	☒	☒	☒
58	IN HMI Request	Assignment tag table Bool	%M2202.4	☐	☒	☒	☒
59	OUT HMI Request	Assignment tag table Bool	%M2202.5	☐	☒	☒	☒
60	OPEN HMI Request	Assignment tag table Bool	%M2202.6	☐	☒	☒	☒
61	CLOSE HMI Request	Assignment tag table Bool	%M2202.7	☐	☒	☒	☒
62	Pause pb	Assignment tag table Bool	%M2201.2	☐	☒	☒	☒
63	Reset Sequence	Assignment tag table Bool	%M2201.3	☐	☒	☒	☒
64	Object Detected	Assignment tag table Bool	%M2201.4	☐	☒	☒	☒
65	Object Ready	Assignment tag table Bool	%M2201.5	☐	☒	☒	☒
66	Start Sequence	Assignment tag table Bool	%M2201.6	☐	☒	☒	☒
67	Reset Object counter	Assignment tag table Bool	%M2201.7	☐	☒	☒	☒
68	<Add new>			☐	☒	☒	☒

FIGURE 21

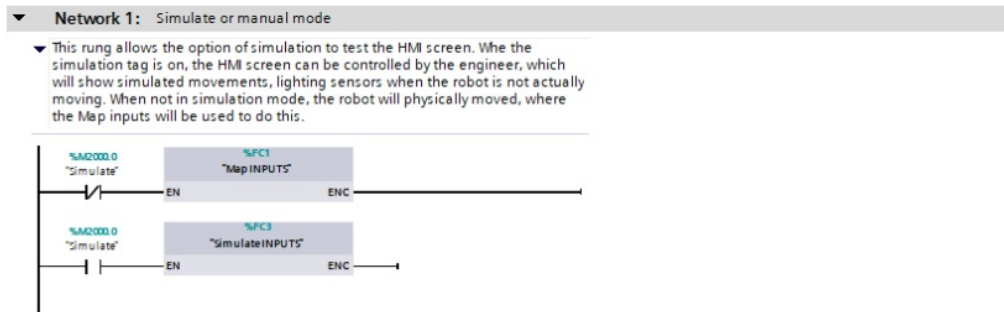


FIGURE 22

Network 2: Object detected process with an on set time delay

- The object will either be detected if the object sensor picks up an object, or the start pb is pressed, as long as the auto sequence step is less than 2. When the object is detected, an onset time delay of two seconds occurs, which will then start the sequence. A one second delay will occur after this, once the sequence has started.

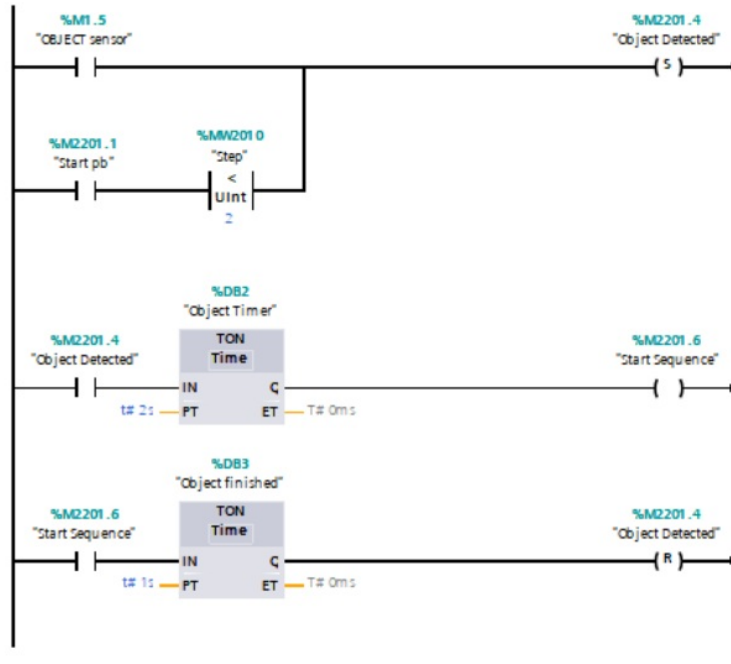


FIGURE 23

Network 3: Pause Button

- The pause pb is tagged to the pause button on the HMI screen. This is a normally closed contact. When the button is not pressed, the Auto sequence will continue. When the button is pressed, the auto sequence will 'stop' until the pause button is released.



FIGURE 24

▼ Network 4: Move UP with interlocks

- ▼ This rung allows the Robot to move up when either in AUTO mode or the HMI UP button is pressed. The Robot must be in the UP position in order for the Robot to move left or right, or in or out.

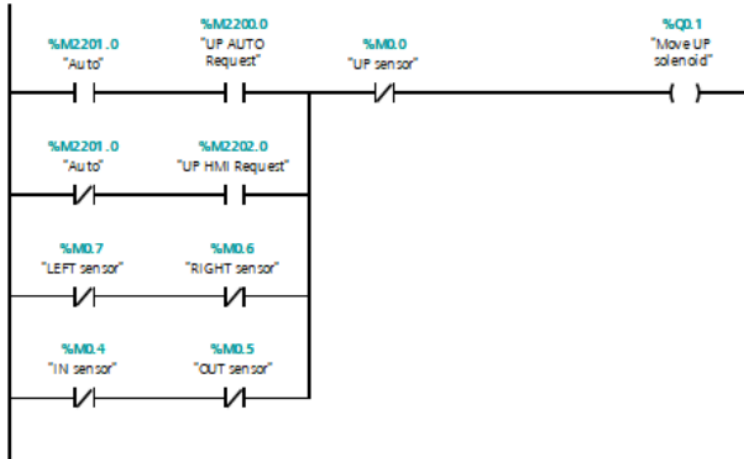


FIGURE 25

▼ Network 5: Move DOWN with interlocks

- ▼ This allows the Robot to move down in AUTO and by HMI, as long as it is above a station. In order to move, either the left or right sensor should be on and the out or in sensor. This shows it is at a station.

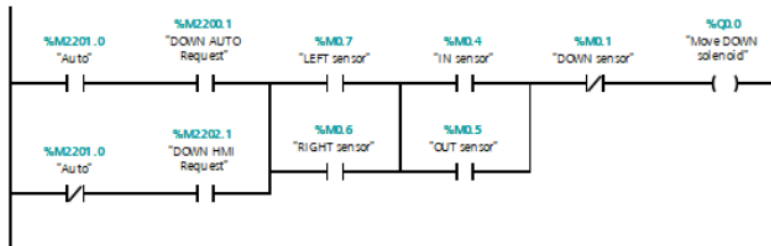
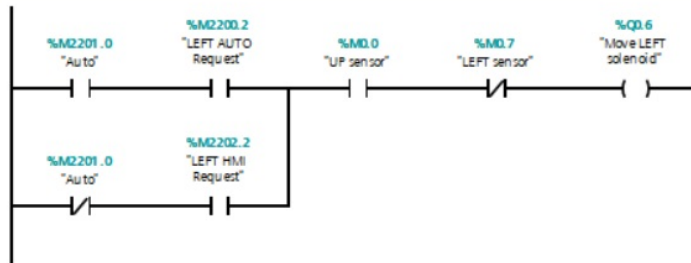


FIGURE 26

Network 6: Move LEFT

- ▼ This rung allows the Robot to Move left in the AUTO or HMI condition. The UP sensor is placed into the rung so that this condition can only take place when the robot is UP. The normally closed contact 'left sensor' makes sure the sensor has not already come in.



Network 7: Move RIGHT

- ▼ This rung allows the Robot to Move Right in the AUTO or HMI condition. The UP sensor is placed into the rung so that this condition can only take place when the robot is UP. The normally closed contact 'right sensor' makes sure the sensor has not already come in.

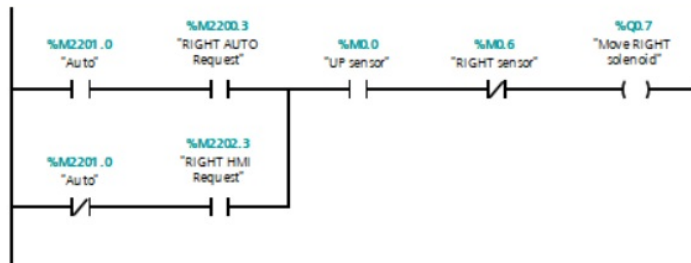


FIGURE 27

Network 8: Gripper OPEN

- ▼ This rung allows the Robot to Open the Gripper in the AUTO or HMI condition. The normally closed Open sensor contact is placed into the rung so that this condition can only take place if its not open already. The gripper can open at any stage.

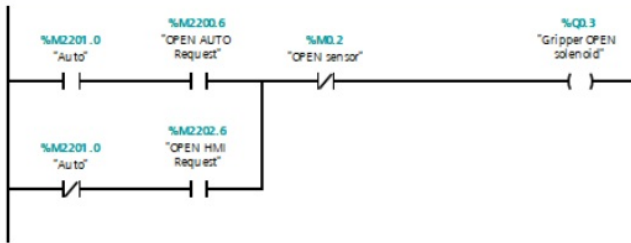


FIGURE 28

Network 9: Gripper CLOSED

- ▼ This rung allows the Robot to Close the Gripper in the AUTO or HMI condition. The normally closed 'Closed sensor' contact is placed into the rung so that this condition can only take place if its not already in this condition. The gripper can close at any stage in the system.

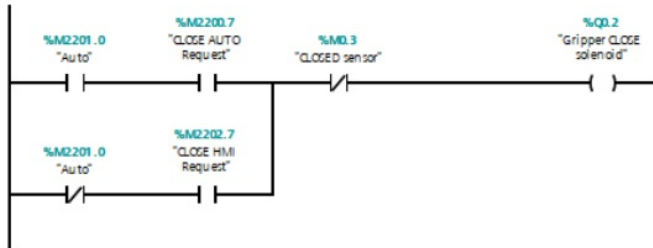


FIGURE 29

Network 10: Move IN

- ▼ In this condition, the Robot to Move IN in the AUTO or HMI condition. The UP sensor is placed into the rung so that this condition can only take place when the robot is UP. The normally closed IN sensor contact is used so this request only hapened if the robot is not in this position already

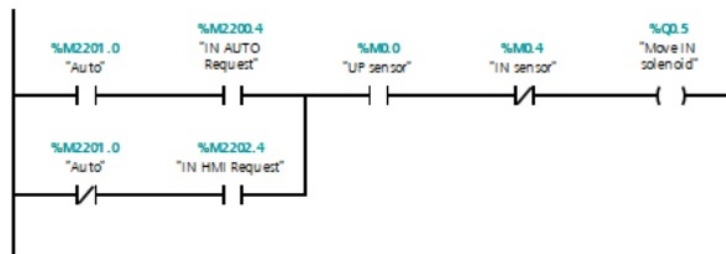


FIGURE 30

Network 11: Move OUT

- ▼ This rung allows the Robot to Move Out in the AUTO or HMI condition. The UP sensor is placed into the rung so that this condition can only take place when the robot is UP. The normally closed contact OUT sensor is used so the commmand is not performed again.

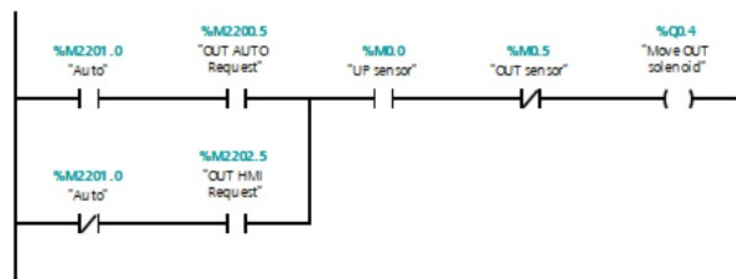


FIGURE 31

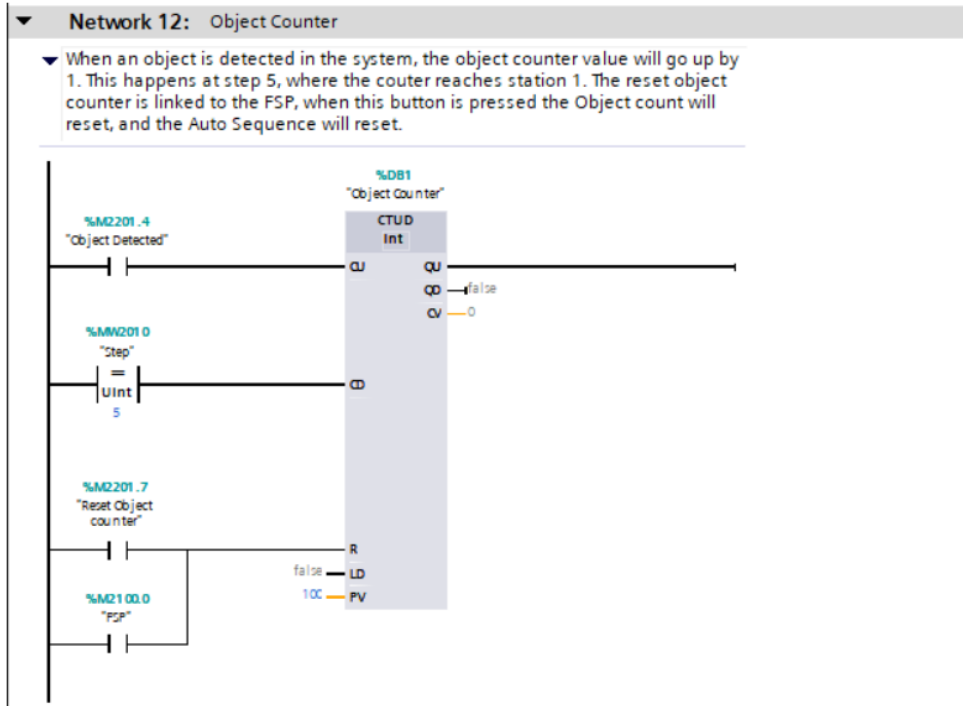


FIGURE 32

```

1 IF ("FSP" =1) OR ("Auto"=0) OR ("Reset Sequence" =1) THEN
2   "Step" := 0;
3   "Next_Step" := 0;
4   "LEFT AUTO Request" := 0;
5   "RIGHT AUTO Request" := 0;
6   "IN AUTO Request" := 0;
7   "OUT AUTO Request" := 0;
8   "OPEN AUTO Request" := 0;
9   "CLOSE AUTO Request" := 0;
10  END_IF;
11
12 CASE "Step" OF
13   0: //move up
14     //This is the start of the auto sequence
15     "UP AUTO Request" := 1;
16     IF "UP sensor" = 1 THEN
17       "UP AUTO Request" := 0;
18       "Next_Step" := 1;
19     END_IF;
20   1: //move home
21     //At this stage the robot will be above station 4 and wait until a counter is added.

```

FIGURE 33

```

20 1: //move home
21 //At this stage the robot will be above station 4 and wait until a counter is added.
22 "IN AUTO Request" := 1;
23 "RIGHT AUTO Request" := 1;
24 "OPEN AUTO Request" := 1;
25 IF ("IN sensor" = 1) AND
26 ("RIGHT sensor" = 1) AND
27 ("OPEN sensor" = 1) AND
28 ("Start Sequence" = 1) THEN
29 "IN AUTO Request" := 0;
30 "RIGHT AUTO Request" := 0;
31 "OPEN AUTO Request" := 0;
32 IF ("Object Counter".CV = 1) THEN
33 "Next_Step" := 19;
34 END_IF;
35 IF ("Object Counter".CV = 2) THEN
36 "Next_Step" := 10;
37 END_IF;
38 IF ("Object Counter".CV = 3) THEN
39 "Next_Step" := 2;
40 END IF;

```

FIGURE 34

```

42 2: //move object from station 3 to 4
43 //The sequence will start on this step if the object counter is equal to 3.
44 //It will move the first object from station 3 to station 4 so there is a max 2 counters in the system.
45 //move out
46 "OUT AUTO Request" := 1;
47 IF "OUT sensor" = 1 THEN
48 "OUT AUTO Request" := 0;
49 "Next_Step" := 3;
50 END_IF;
51 3: //move down
52 "DOWN AUTO Request" := 1;
53 IF "DOWN sensor" = 1 THEN
54 "DOWN AUTO Request" := 0;
55 "Next_Step" := 4;
56 END_IF;
57 4: //gripper close
58 "CLOSE AUTO Request" := 1;
59 IF "CLOSED sensor" = 1 THEN
60 "CLOSE AUTO Request" := 0;
61 "Next_Step" := 5;
62 END IF;

```

FIGURE 35

```

63 END_IF;
64 5: //move up
65 "UP AUTO Request" := 1;
66 IF "UP sensor" = 1 THEN
67 "UP AUTO Request" := 0;
68 "Next_Step" := 6;
69 END_IF;
70 6: //move in
71 "IN AUTO Request" := 1;
72 IF "IN sensor" = 1 THEN
73 "IN AUTO Request" := 0;
74 "Next_Step" := 7;
75 END_IF;
76 7: //move down
77 "DOWN AUTO Request" := 1;
78 IF "DOWN sensor" = 1 THEN
79 "DOWN AUTO Request" := 0;
80 "Next_Step" := 8;
81 END_IF;
82 8: //gripper open
83 "OPEN AUTO Request" := 1;

```

FIGURE 36

```

83 IF "OPEN sensor" = 1 THEN
84     "OPEN AUTO Request" := 0;
85     "Next_Step" := 9;
86 END_IF;
87 9: //move up
88     "UP AUTO Request" := 1;
89 IF "UP sensor" = 1 THEN
90     "UP AUTO Request" := 0;
91     "Next_Step" := 10;
92 END_IF;
93 10: //move from station 2 to 3
94     //The sequence will start on this step if the object counter is equal to 2.
95     //It will move the original counter from station 2 to 3 before the second counter can be added.
96     //move out
97     "OUT AUTO Request" := 1;
98 IF "OUT sensor" = 1 THEN
99     "OUT AUTO Request" := 0;
100     "Next_Step" := 11;
101 END_IF;
102 11: //move left
103     "LEFT AUTO Request" := 1;
104 IF "LEFT sensor" = 1 THEN
105     "LEFT AUTO Request" := 0;
106     "Next_Step" := 12;
107 END_IF;
108 12: //move down

```

FIGURE 37

```

108 12: //move down
109     "DOWN AUTO Request" := 1;
110 IF "DOWN sensor" = 1 THEN
111     "DOWN AUTO Request" := 0;
112     "Next_Step" := 13;
113 END_IF;
114 13: //gripper close
115     "CLOSE AUTO Request" := 1;
116 IF "CLOSED sensor" = 1 THEN
117     "CLOSE AUTO Request" := 0;
118     "Next_Step" := 14;
119 END_IF;
120 14: //move up
121     "UP AUTO Request" := 1;
122 IF "UP sensor" = 1 THEN
123     "UP AUTO Request" := 0;
124     "Next_Step" := 15;
125 END_IF;
126 15: //move right
127     "RIGHT AUTO Request" := 1;
128 IF "RIGHT sensor" = 1 THEN
129     "RIGHT AUTO Request" := 0;
130     "Next_Step" := 16;
131 END_IF;
132 16: //move down
133     "DOWN AUTO Request" := 1;

```

FIGURE 38

```

132 16: //move down
133    "DOWN AUTO Request" := 1;
134    IF "DOWN sensor" = 1 THEN
135        "DOWN AUTO Request" := 0;
136        "Next_Step" := 17;
137    END_IF;
138 17: //gripper open
139    "OPEN AUTO Request" := 1;
140    IF "OPEN sensor" = 1 THEN
141        "OPEN AUTO Request" := 0;
142        "Next_Step" := 18;
143    END_IF;
144 18: //move up
145    "UP AUTO Request" := 1;
146    IF "UP sensor" = 1 THEN
147        "UP AUTO Request" := 0;
148        "Next_Step" := 19;
149    END_IF;
150 19: //This moves the from station 1 to 2.
151    //The sequence will start on this step if the object counter is equal to 1.
152    //move left
153    "LEFT AUTO Request" := 1;
154    IF "LEFT sensor" = 1 THEN
155        "LEFT AUTO Request" := 0;
156        "Next_Step" := 20;
157    END_IF;

```

FIGURE 39

```

157    END_IF;
158 20: //move in
159    "IN AUTO Request" := 1;
160    IF "IN sensor" = 1 THEN
161        "IN AUTO Request" := 0;
162        "Next_Step" := 21;
163    END_IF;
164 21: //move down
165    "DOWN AUTO Request" := 1;
166    IF "DOWN sensor" = 1 THEN
167        "DOWN AUTO Request" := 0;
168        "Next_Step" := 22;
169    END_IF;
170 22: //gripper close
171    "CLOSE AUTO Request" := 1;
172    IF "CLOSED sensor" = 1 THEN
173        "CLOSE AUTO Request" := 0;
174        "Next_Step" := 23;
175    END_IF;
176 23: //move up
177    "UP AUTO Request" := 1;
178    IF "UP sensor" = 1 THEN
179        "UP AUTO Request" := 0;
180        "Next_Step" := 24;
181    END_IF;
182 24: //move out

```

FIGURE 40

```

182 24: //move out
183     "OUT AUTO Request" := 1;
184     IF "OUT sensor" = 1 THEN
185         "OUT AUTO Request" := 0;
186         "Next_Step" := 25;
187     END_IF;
188 25: //move down
189     "DOWN AUTO Request" := 1;
190     IF "DOWN sensor" = 1 THEN
191         "DOWN AUTO Request" := 0;
192         "Next_Step" := 26;
193     END_IF;
194 26: //gripper open
195     "OPEN AUTO Request" := 1;
196     IF "OPEN sensor" = 1 THEN
197         "OPEN AUTO Request" := 0;
198         "Next_Step" := 0;
199     END_IF;
200
201
202 END_CASE;
203 "Step" := "Next_Step";
204

```

FIGURE 41

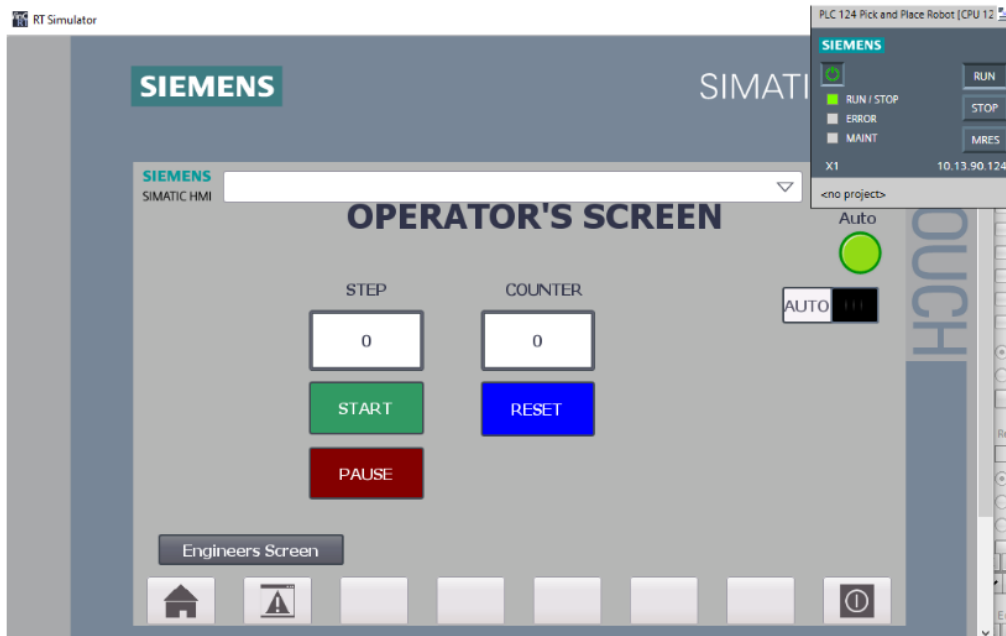


FIGURE 42

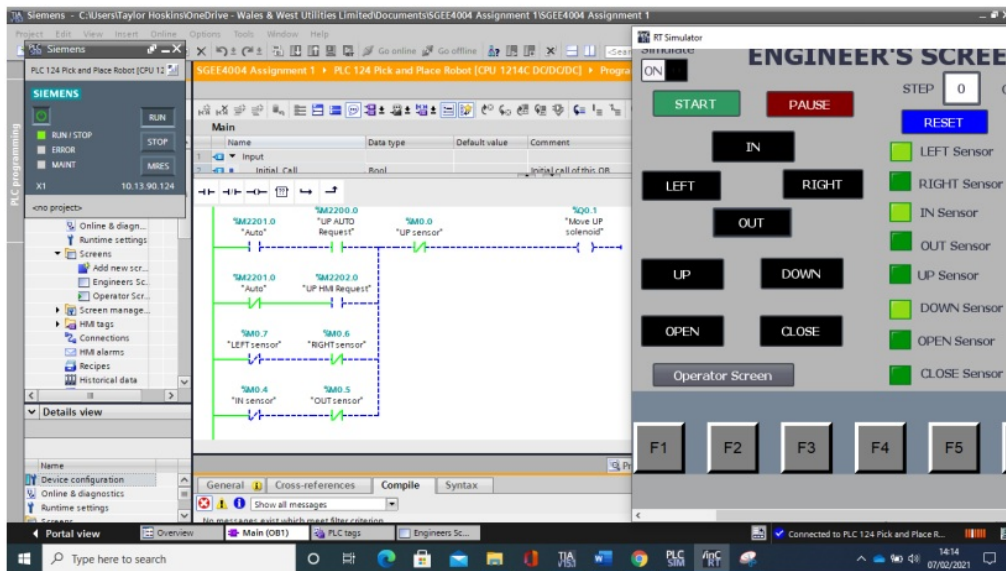


FIGURE 43

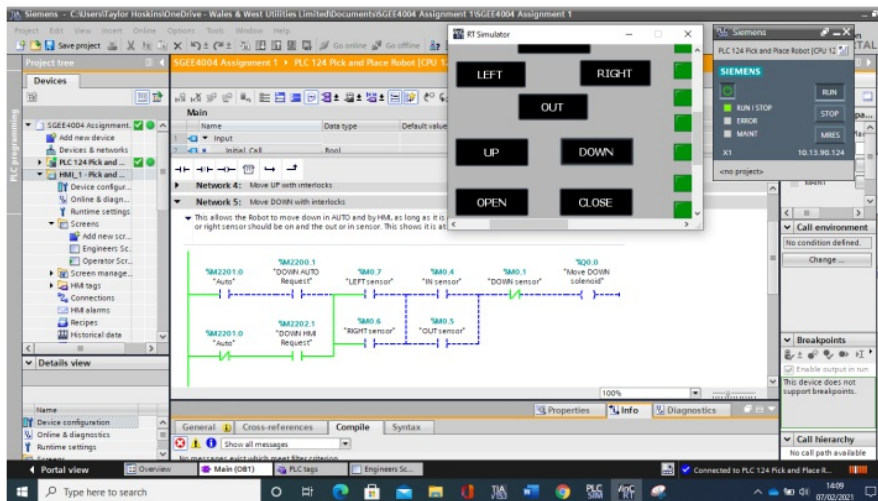


FIGURE 44

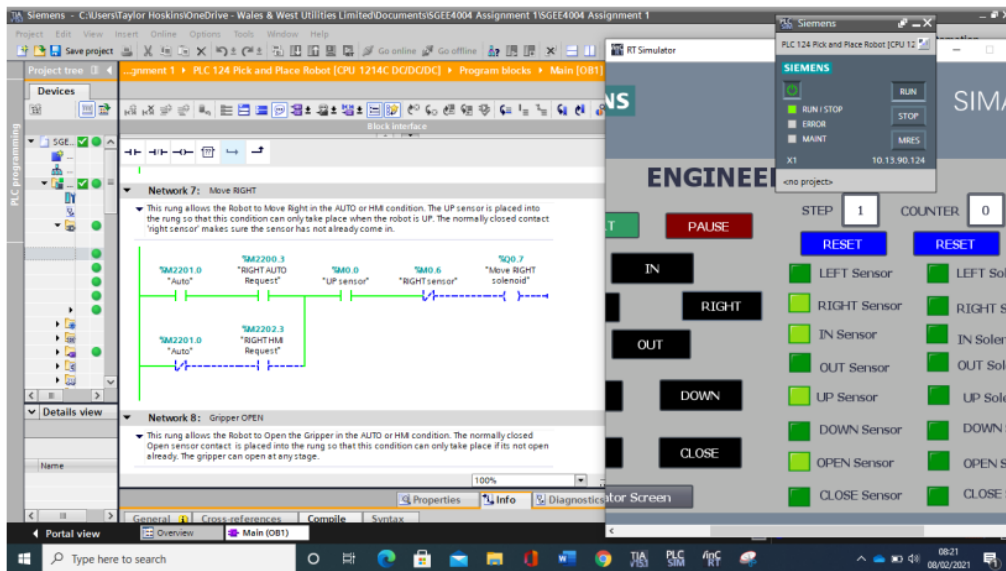


FIGURE 45

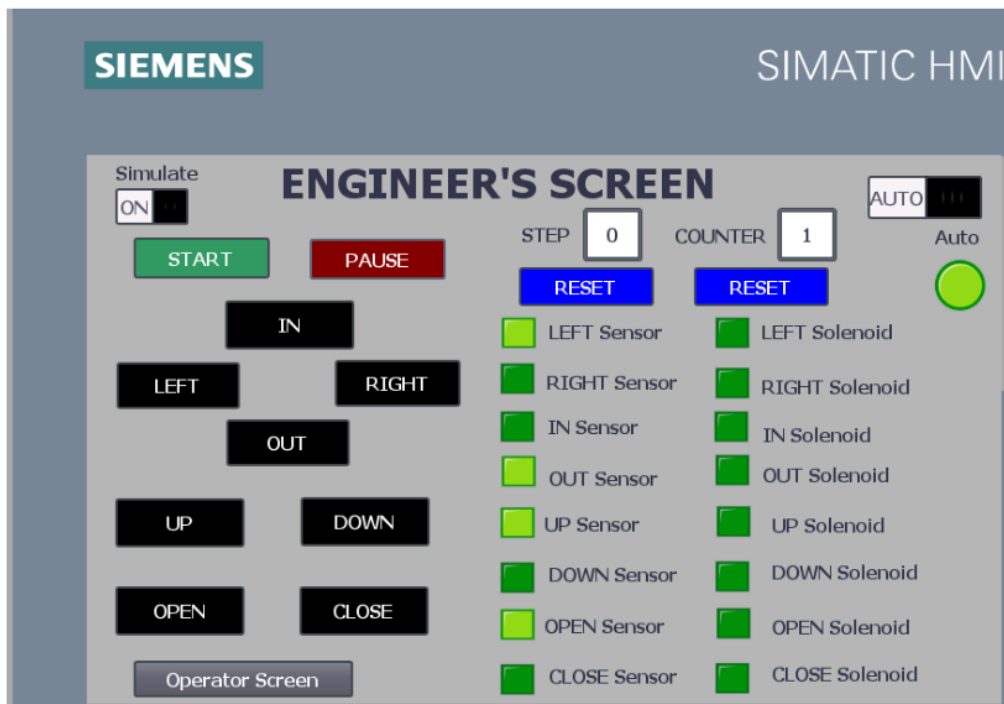


FIGURE 46

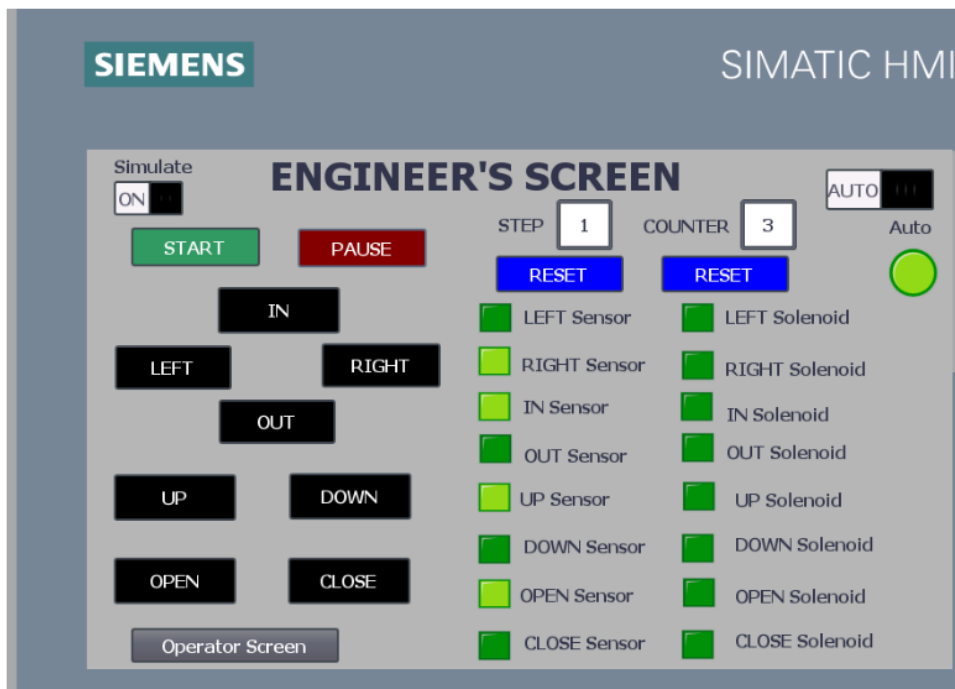


FIGURE 47

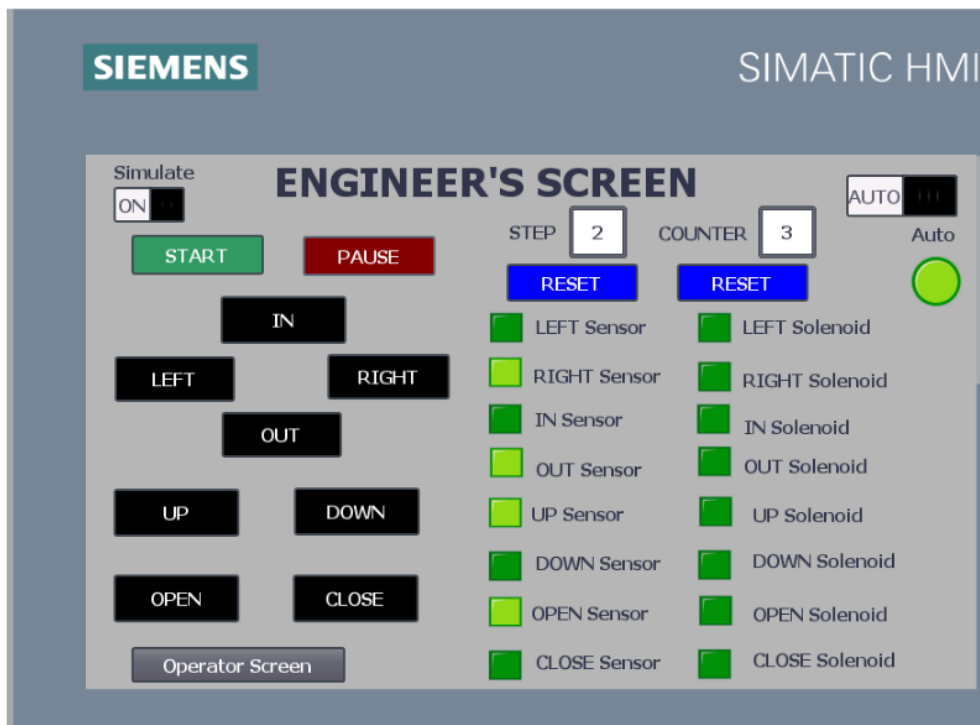


FIGURE 48

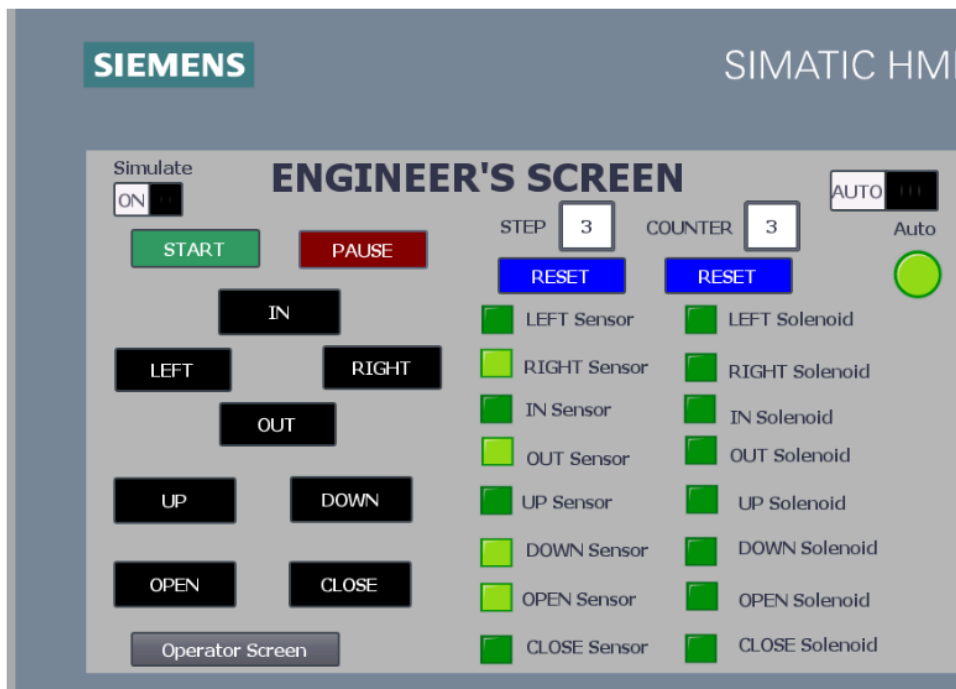


FIGURE 49

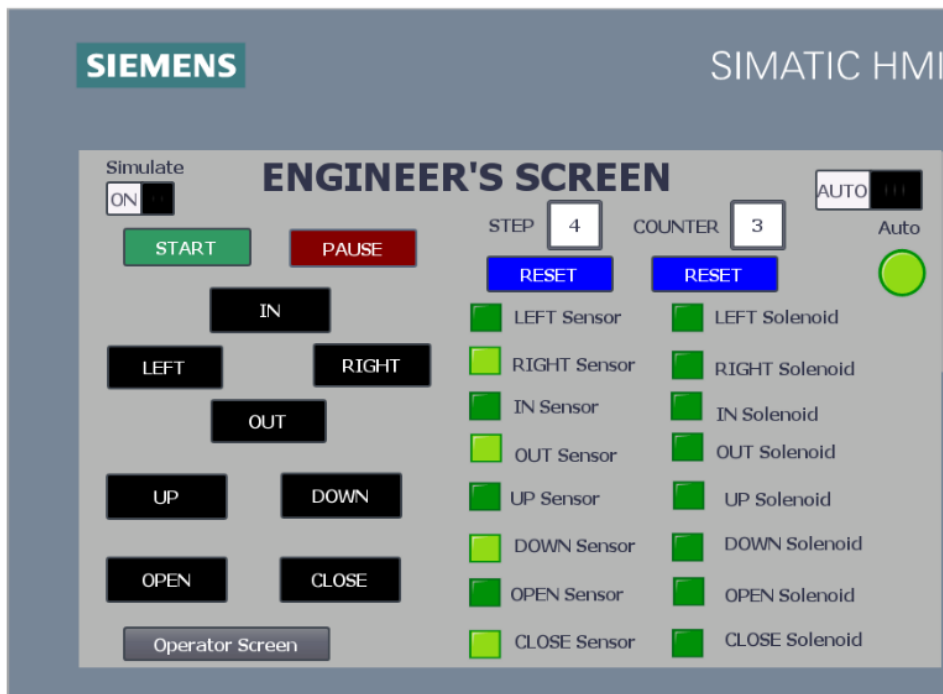


FIGURE 50

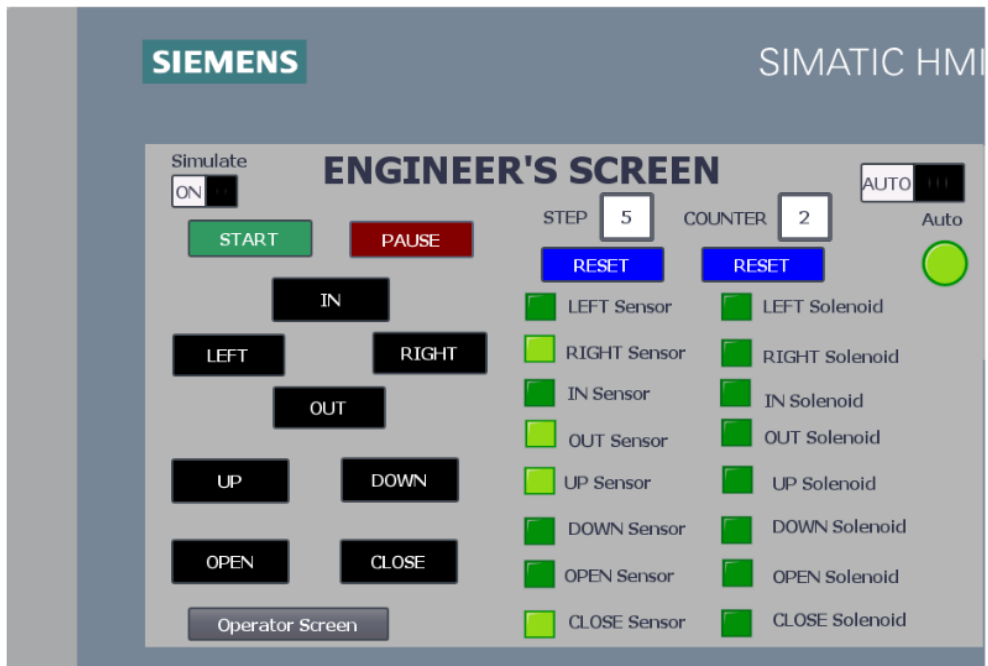


FIGURE 51

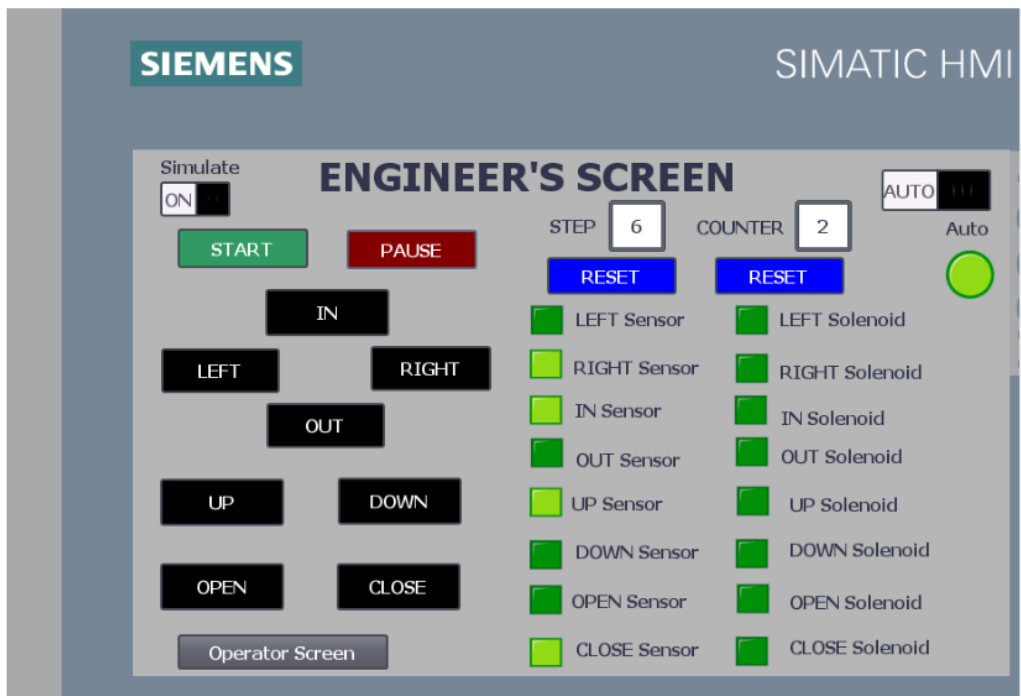


FIGURE 52

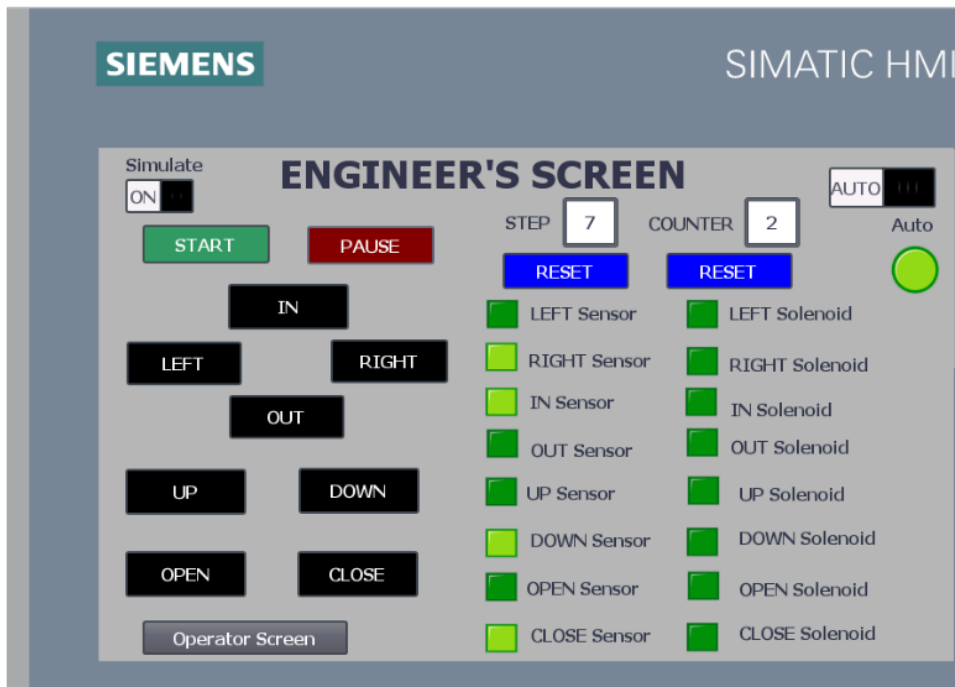


FIGURE 53

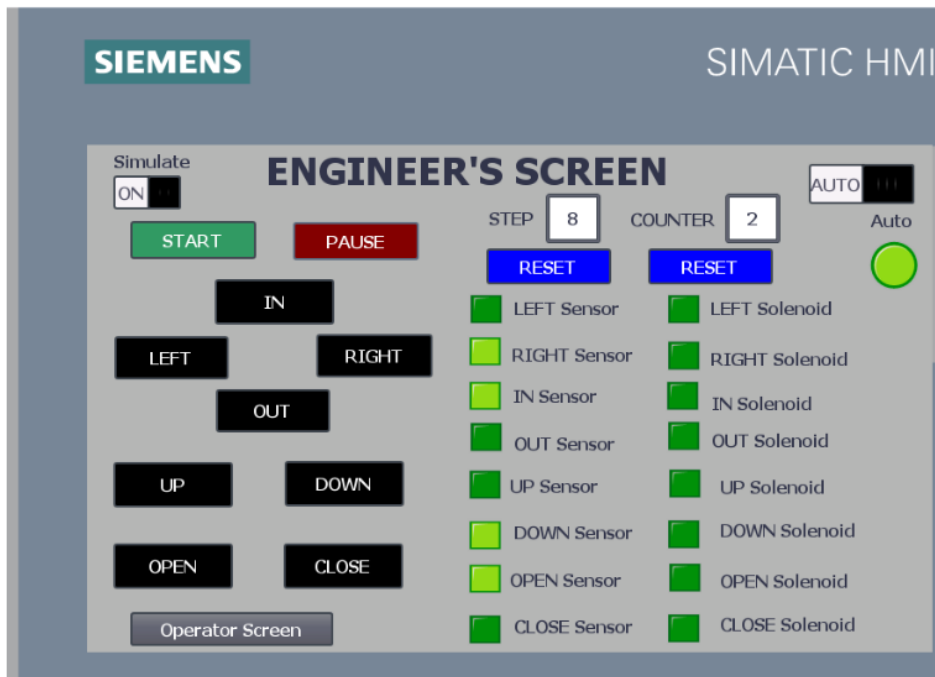


FIGURE 54

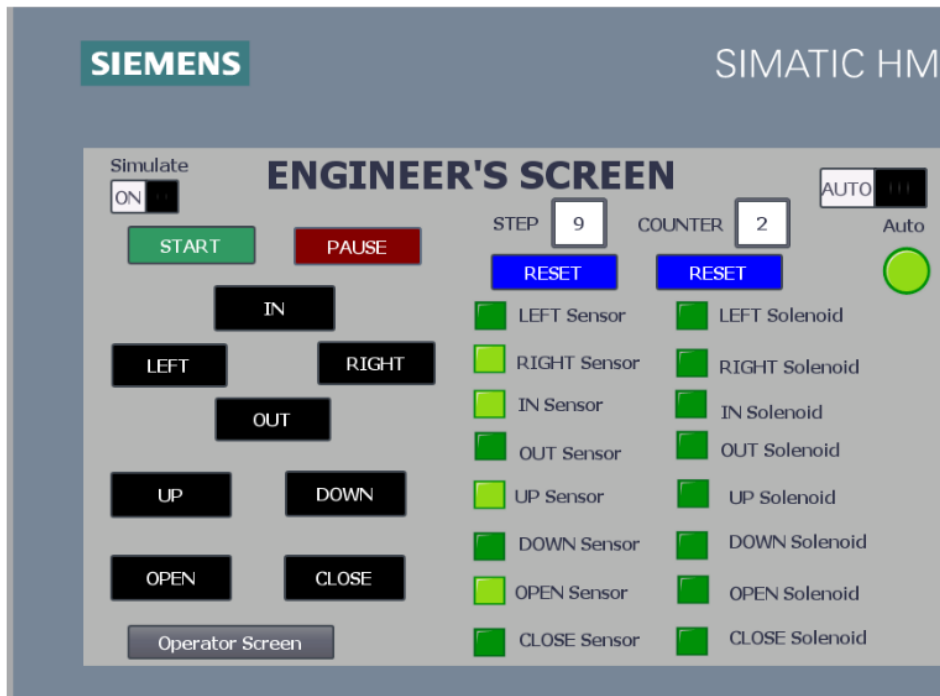


FIGURE 55

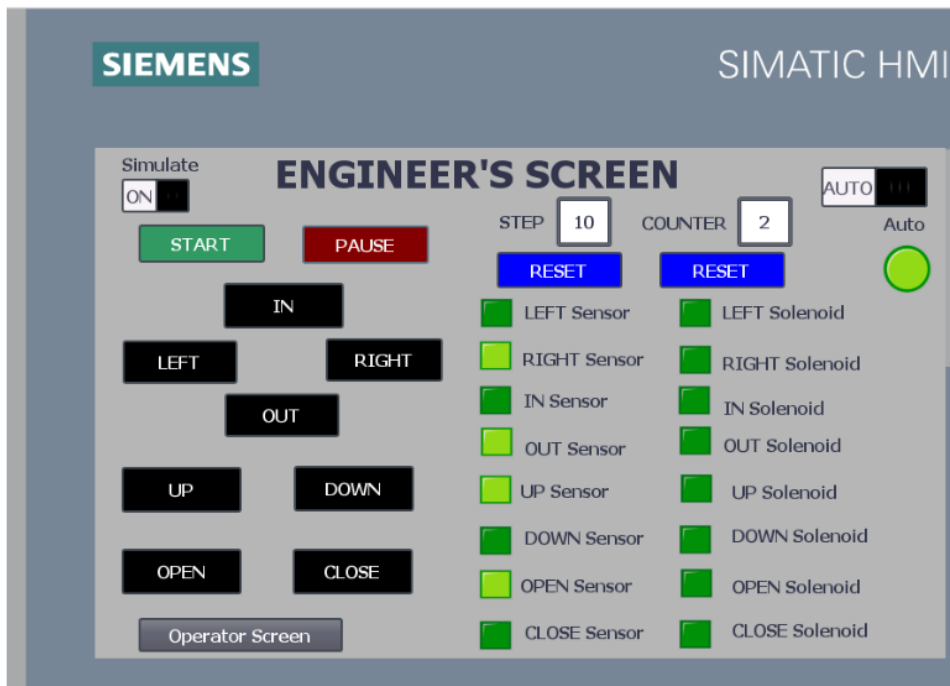


FIGURE 56

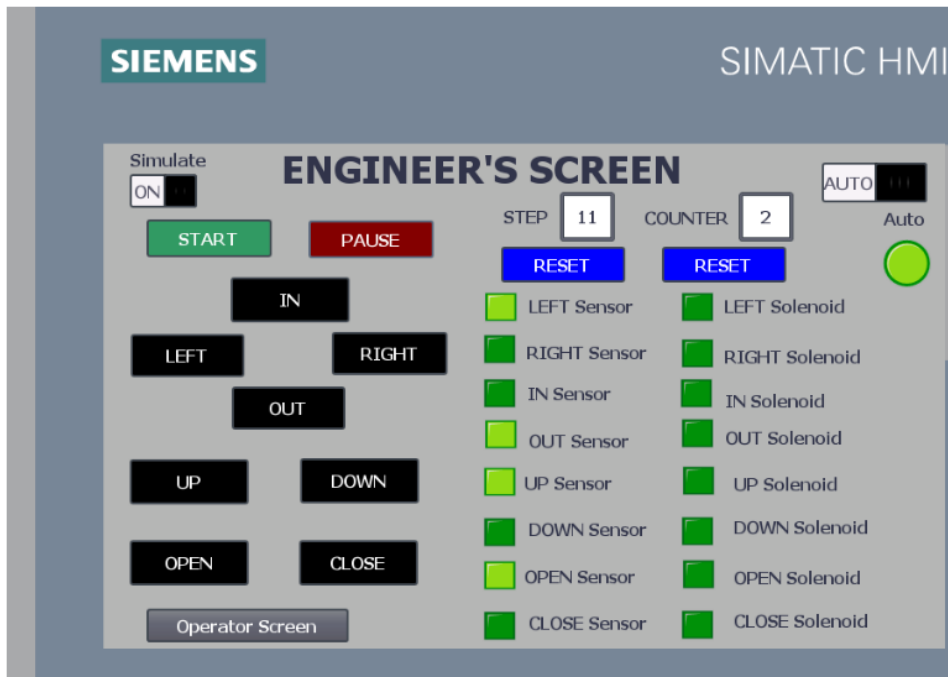


FIGURE 57

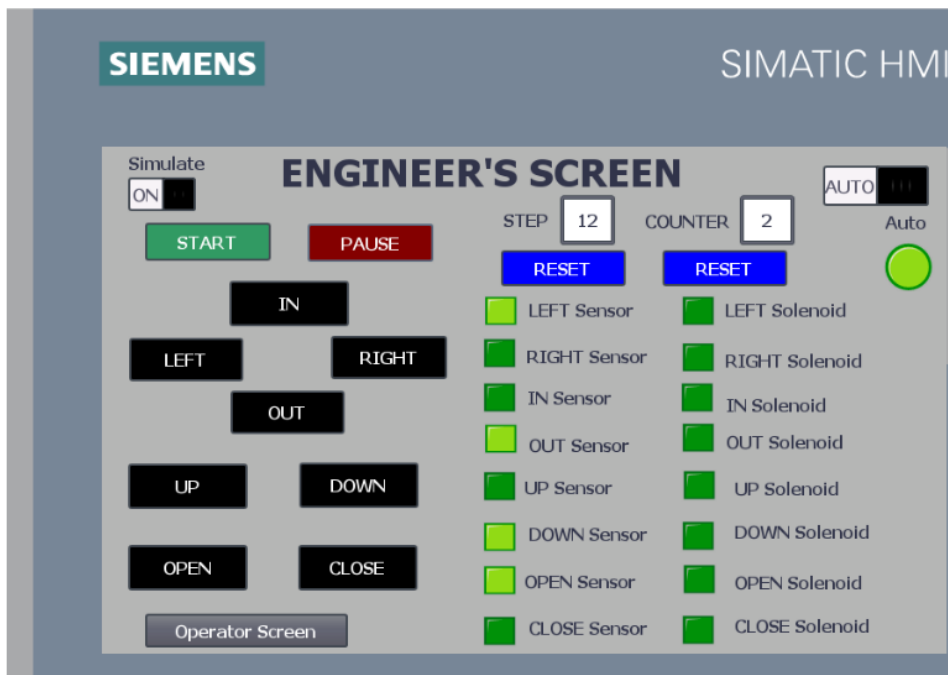


FIGURE 58

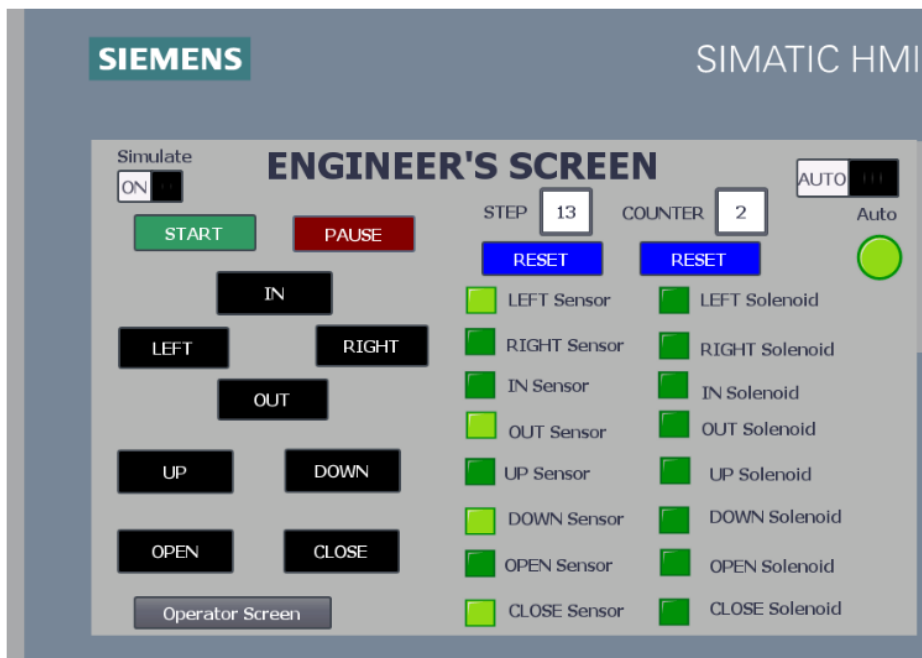


FIGURE 59

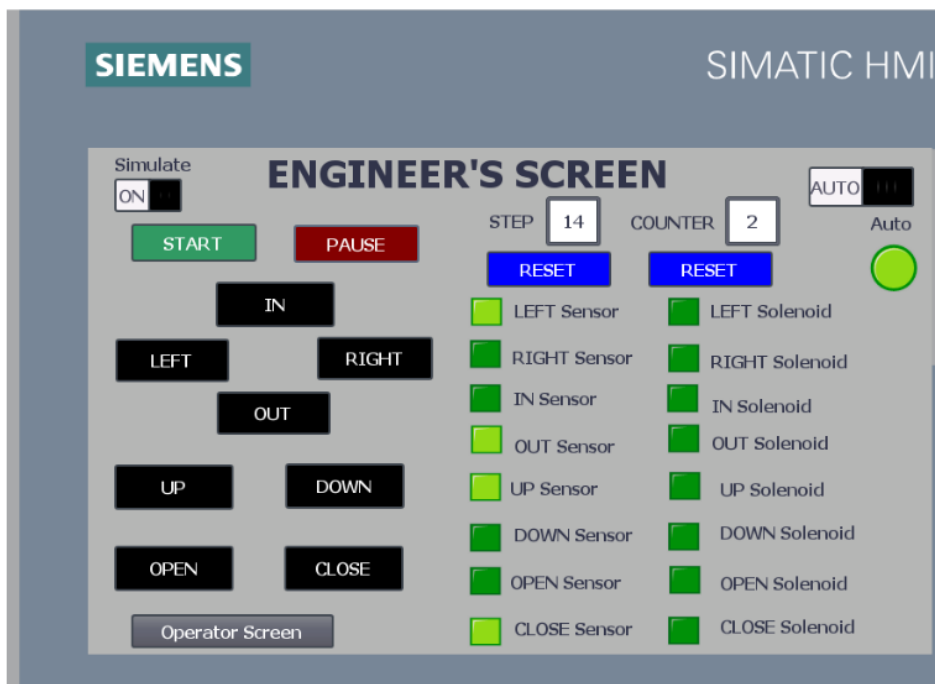


FIGURE 60

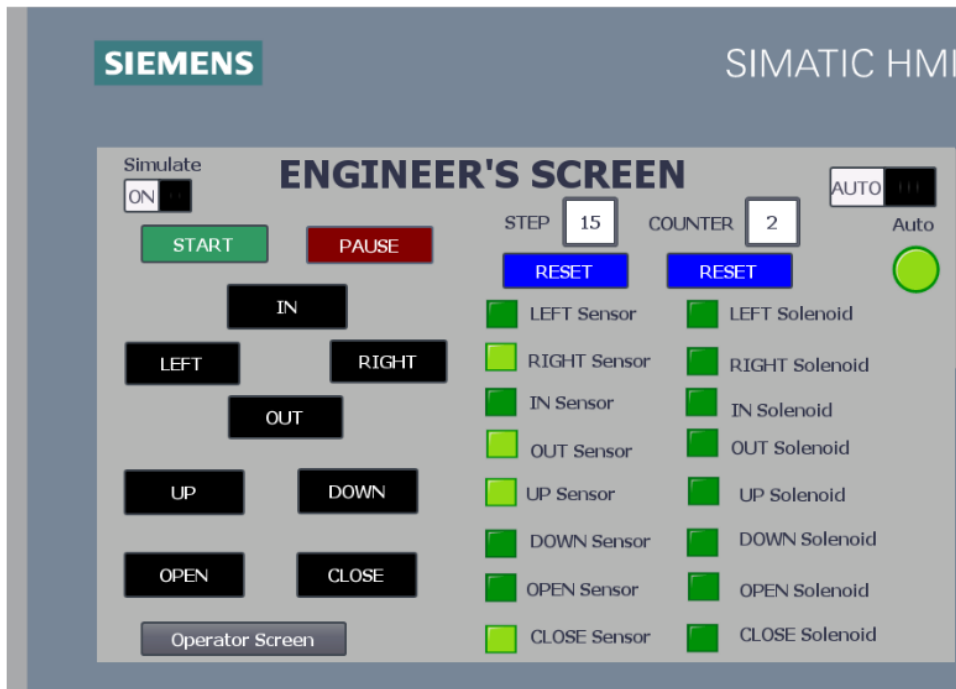


FIGURE 61

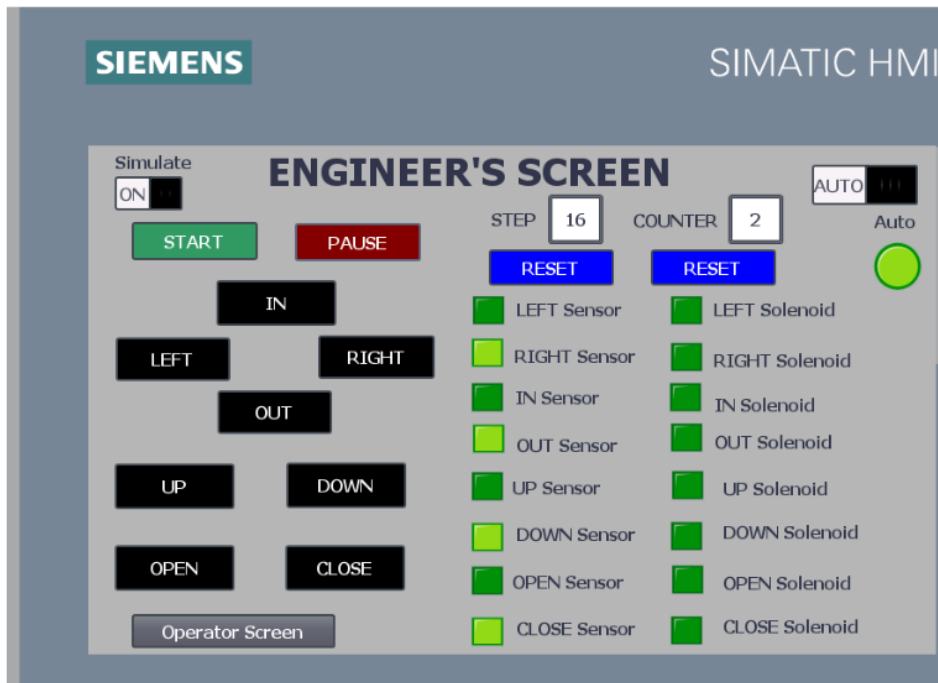


FIGURE 62

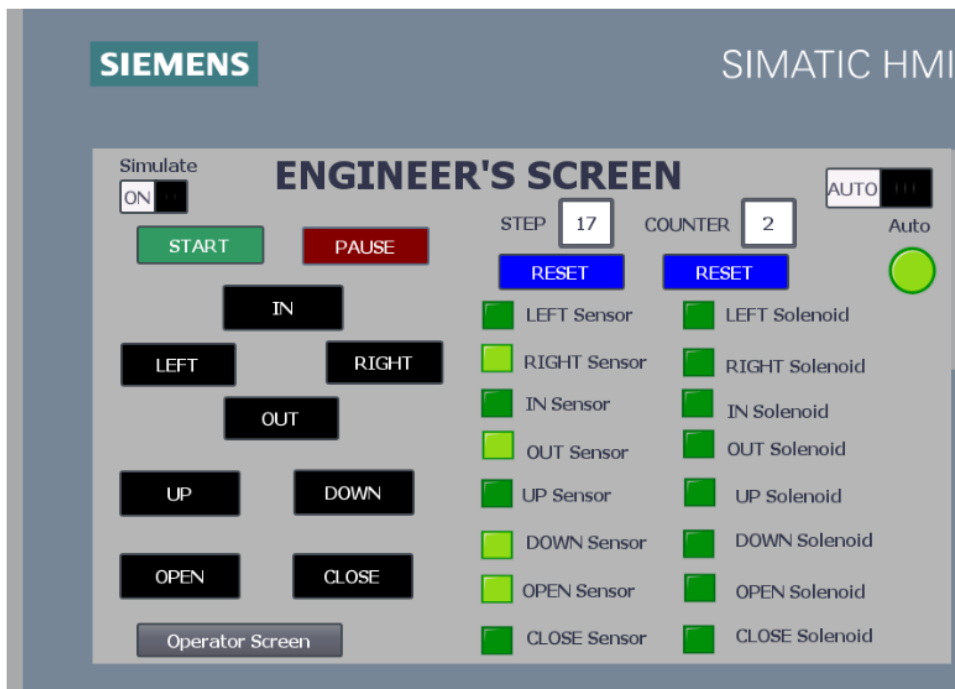


FIGURE 63

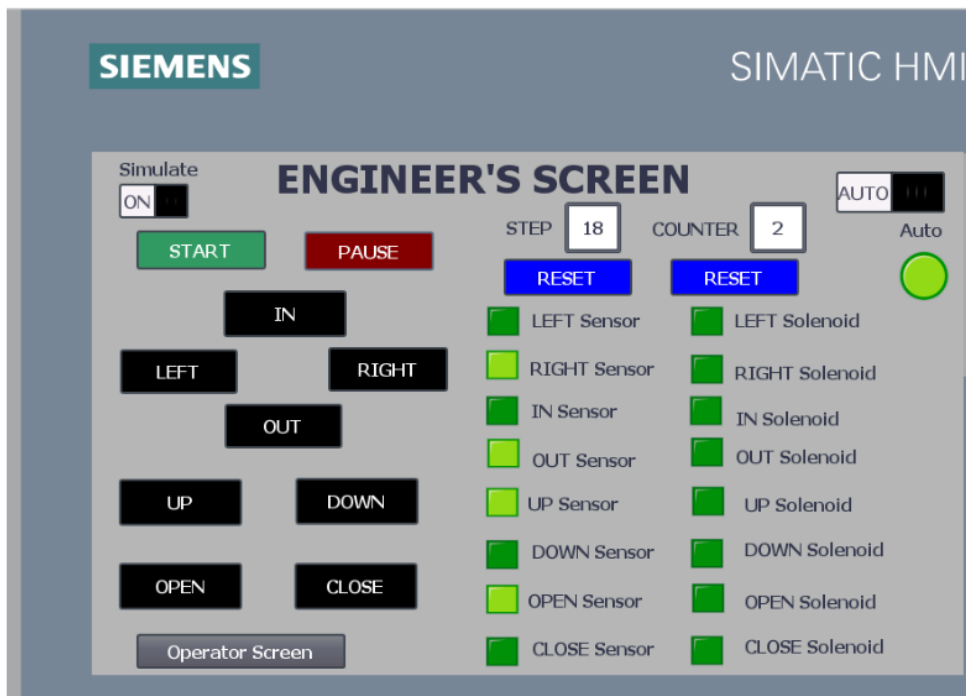


FIGURE 64

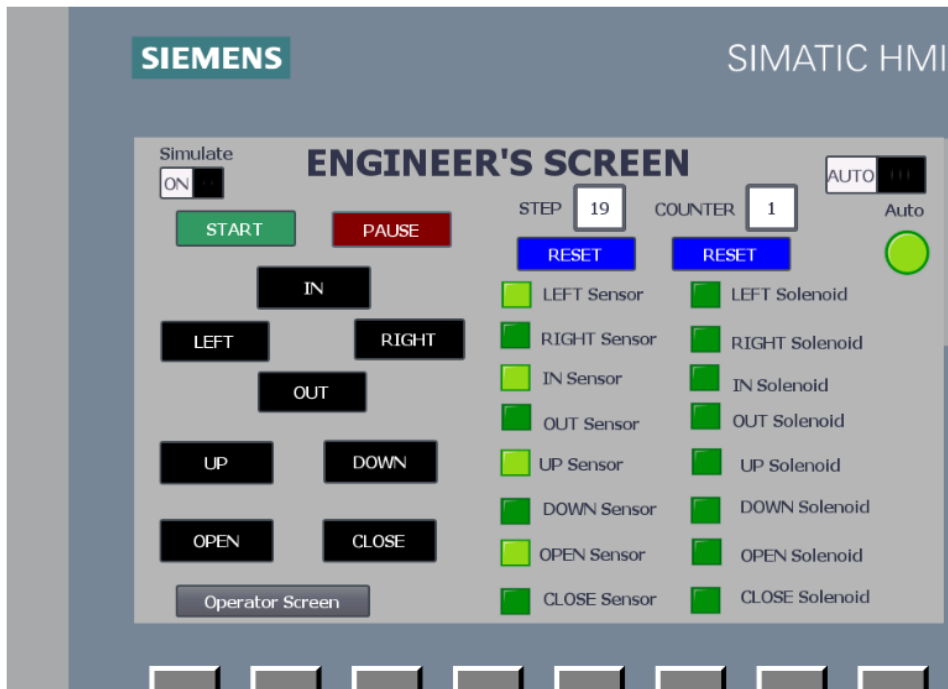


FIGURE 65

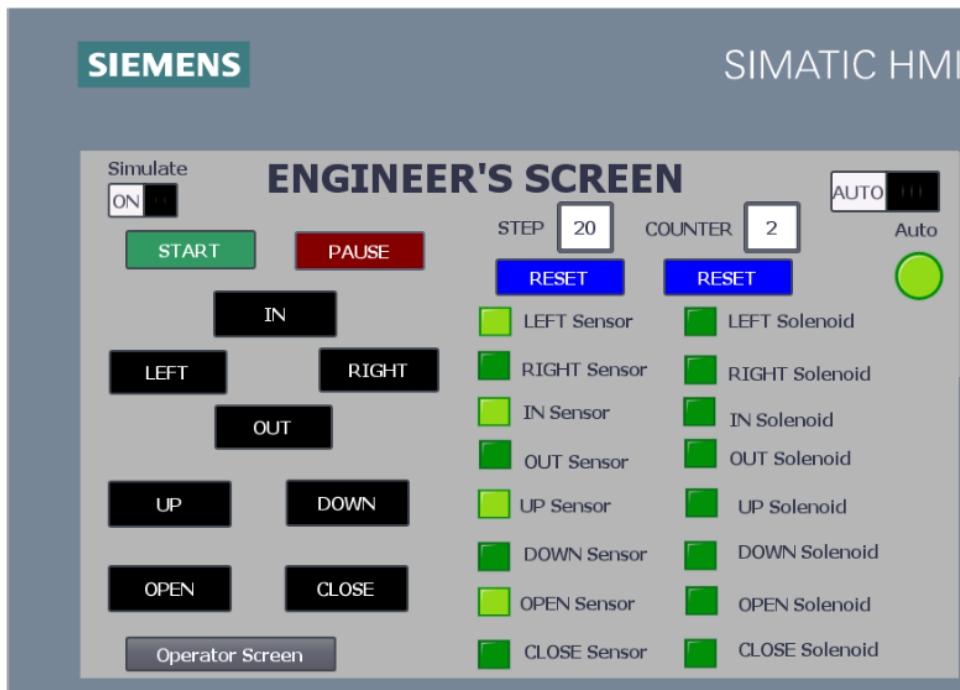


FIGURE 66

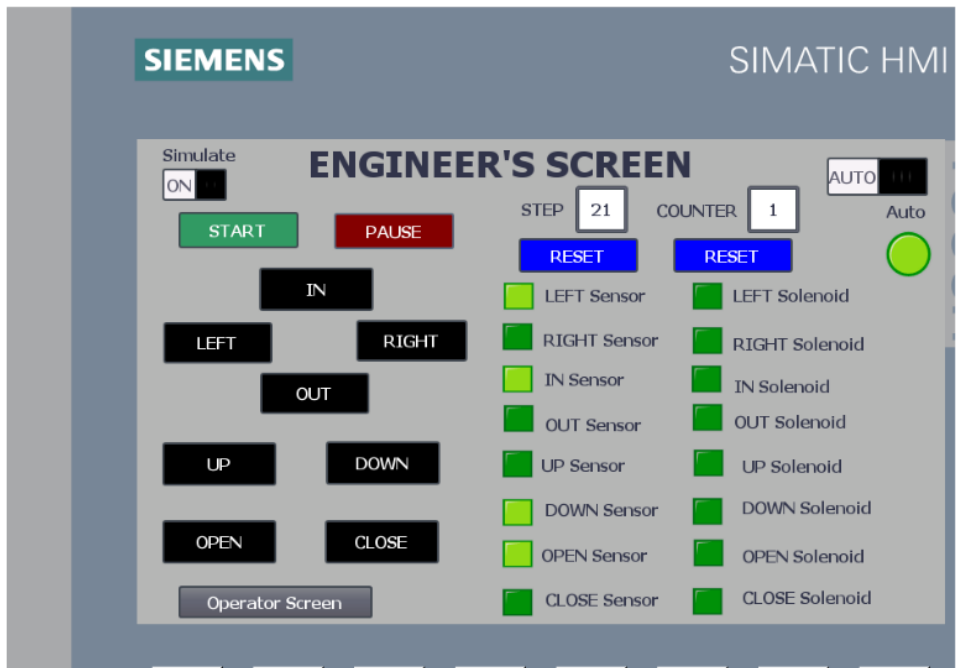


FIGURE 67

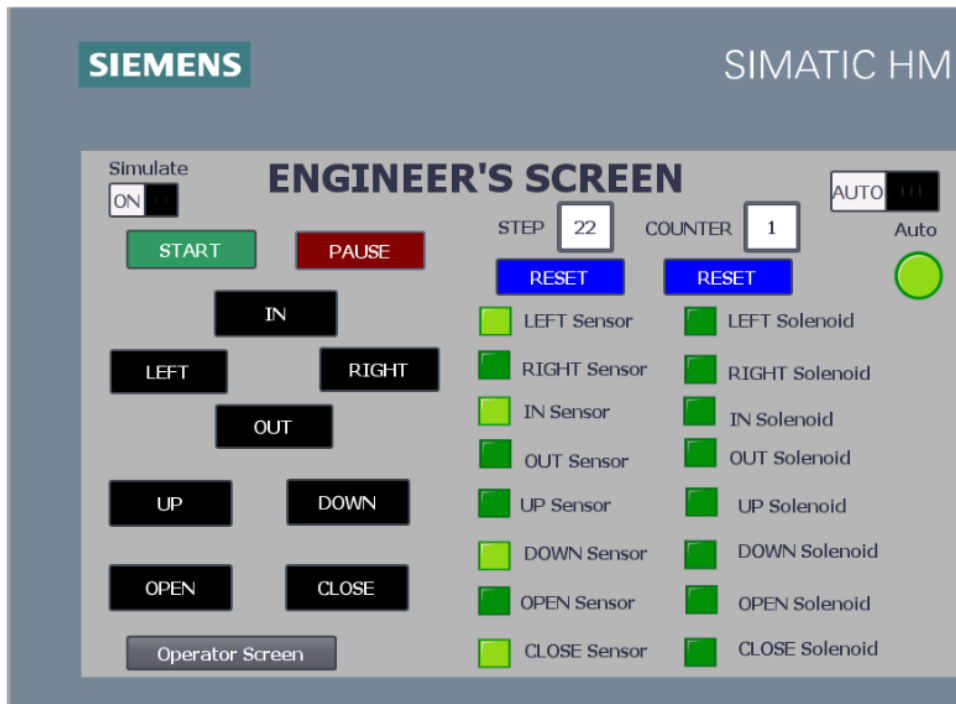


FIGURE 68

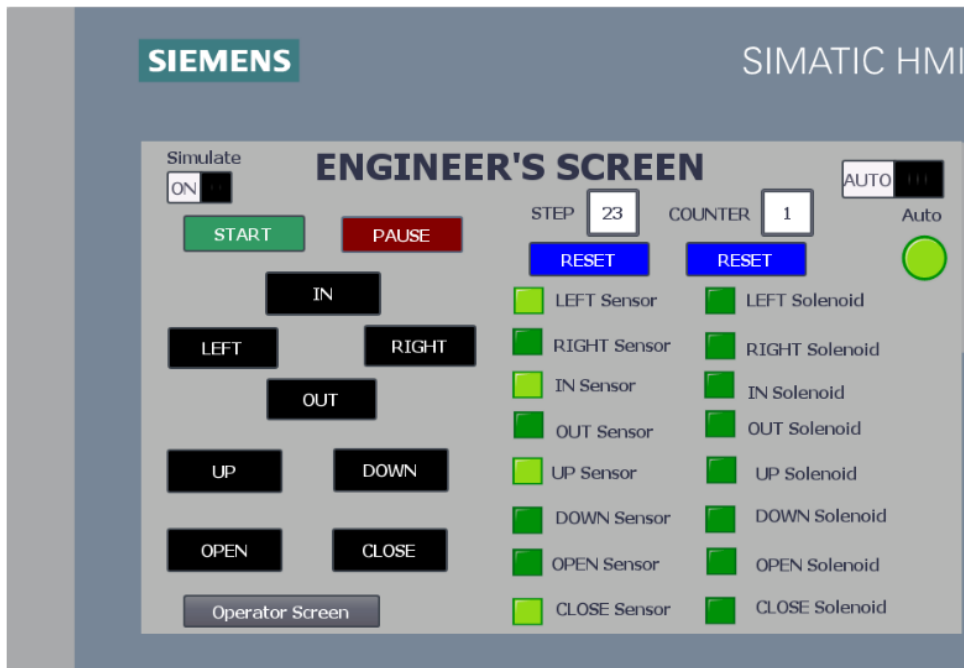


FIGURE 69

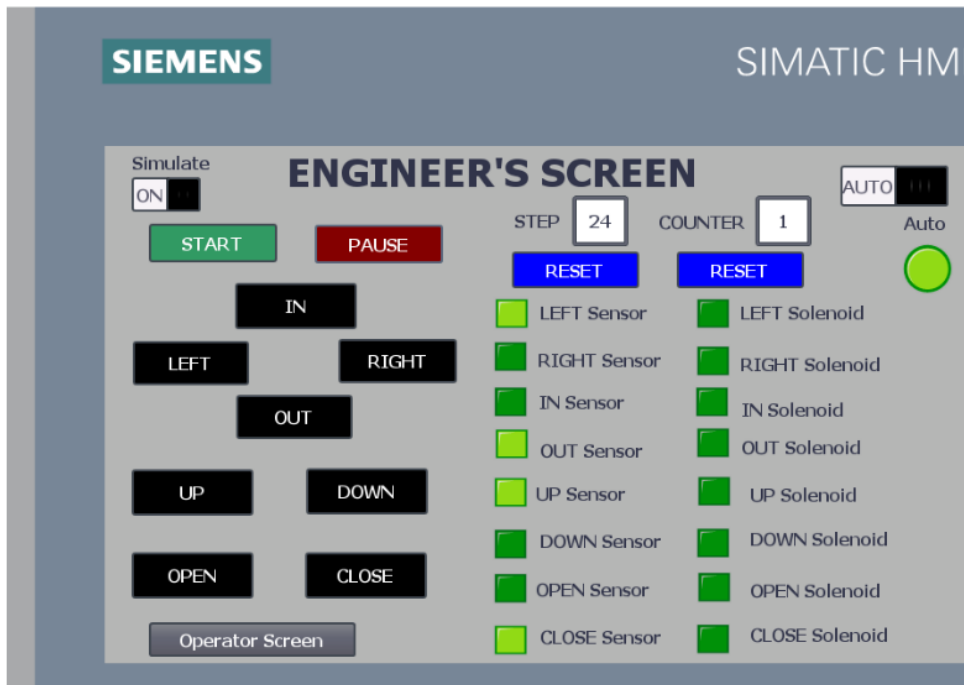


FIGURE 70

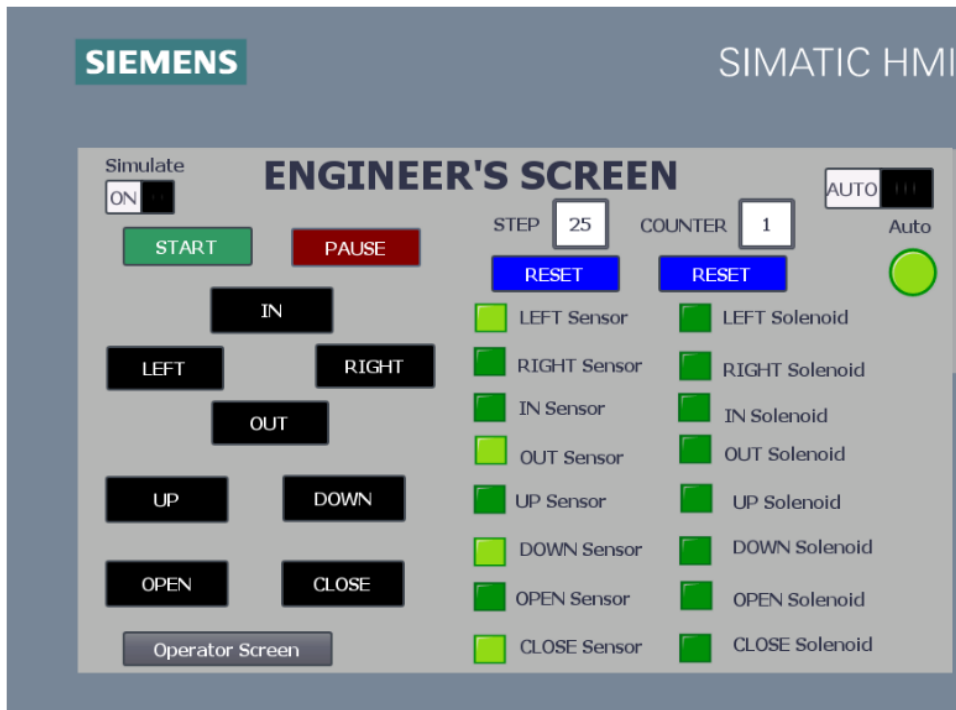


FIGURE 71

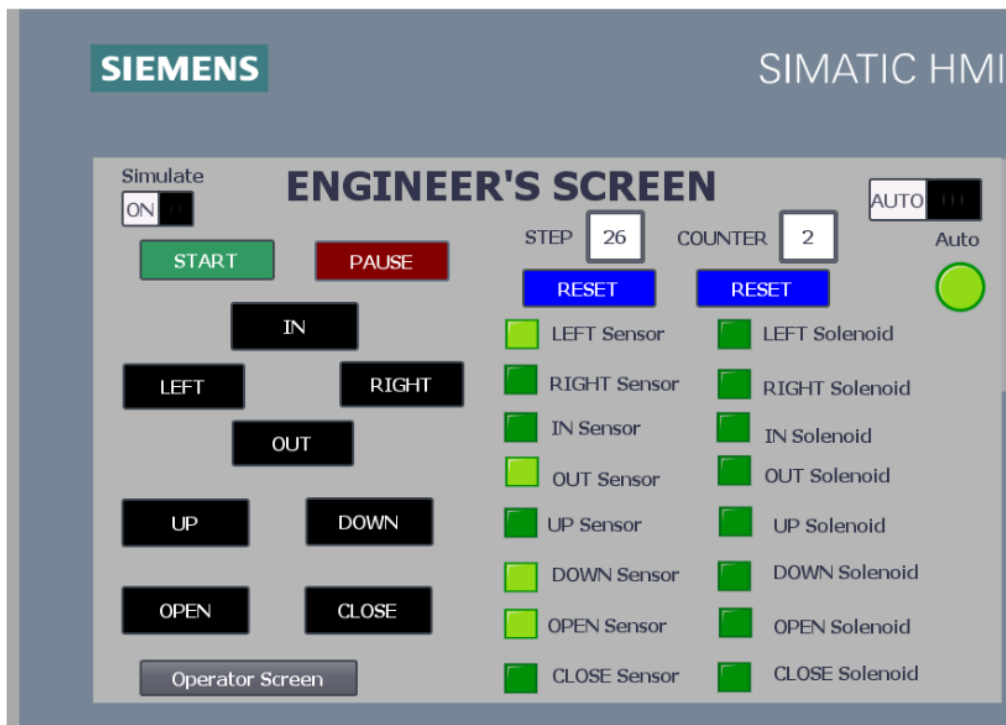


FIGURE 72

FINAL GRADE

GENERAL COMMENTS

Instructor

68

/100

PAGE 1

PAGE 2

QM

Good Structure

Report is structured exactly as stated in the assignment brief.

PAGE 3

Text Comment. Your first sentence is an unnecessary use of words

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

Text Comment. Put your citation before the full stop, otherwise it looks like the citation belongs to the next sentence.

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

Text Comment. You could have mentioned it is rack mounted

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

Text Comment. Which one is a unitary PLC?

PAGE 4

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

QM

Incorrectly cited, this is not the Harvard Method

Text Comment. Good conclusion

PAGE 5

Text Comment. These switches should be in your drawing and the I/O list

PAGE 6

Text Comment. Good drawing but not all inputs are shown

Text Comment. incorrect symbol for proximity switch

Text Comment. inputs missing

PAGE 7

Text Comment. The I/O table should actual I addresses not the M addresses to which they are mapped. It should match up with the drawing

PAGE 8

Text Comment. Don't waste word giving a history of how it was made

Text Comment. Don't waste word giving a history of how it was made

PAGE 9

Text Comment. Don't waste word giving a history of how it was made

Text Comment. This is a repetition of the brief, unnecessary use of words which has increased your word count for which you have lost 5 marks

PAGE 10

Text Comment. Object sensor?

PAGE 11

Text Comment. good SFC chart

Text Comment. Again, you are wasting words telling your reader the story of how you made the program

PAGE 12

Text Comment. Wasted words telling a story

PAGE 13

PAGE 14

Text Comment. A good table

Text Comment. you could have linked each test to the evidence image in the appendix

PAGE 15

PAGE 16

PAGE 17

PAGE 18

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

PAGE 19

PAGE 20

QM

Grocer's apostrophe

If you meant this as a plural word, don't use an apostrophe. This also applies to acronyms.

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

QM

Use the third Person

You are referring to first or second persons (I, we, you, us etc) whereas you should use the third person thus depersonalising it.

PAGE 21

PAGE 22

QM

Good range of sources

You have demonstrated reading from a good range of sources shown by your list below

QM

Correct method of listing.

You have listed your sources correctly according to the Harvard method.

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42

PAGE 43

PAGE 44

PAGE 45

PAGE 46

PAGE 47

PAGE 48

PAGE 49

TAYLOR HOSKINS**68****1. ENGAGEMENT WITH LITERATURE**

-  There is very good evidence of research and reading from a wide range of source correct cited and listed using the Harvard method. However, you have not identified a unitary PLC (15/20)

15**2. KNOWLEDGE AND UNDERSTANDING**

-  Very good evidence of knowledge and understand in your system and program explanations. (15/20)

15**3. COGNITIVE AND INTELLECTUAL**

-  Clear evidence of analytical skills through testing and discussion, conclusions and recommendations. (14/20)

14**4. PRACTICAL APPLICATION**

-  A well designed program, good comments within. Test results are well presented but could be linked to evidence in appendix. Drawing has some

14

components missing. (14/20)

5. TRANSFERABLE SKILLS



Well structured and presented. Generally well written, good evidence of team working. However, word count exceeded (-5 mark penalty applied) because you are wasting precious word telling your reader a story oh how you made the program which is unnecessary. (10/20)

10