# 2221114

*by* Kai White

**Submission date:** 30-Jan-2023 06:42PM (UTC+0000)
**Submission ID:** 196448215
**File name:** 17313_Kai_White_2221114_180905_271796462.pdf (5.94M)
**Word count:** 3071
**Character count:** 25659

| | |
|---|---|
| Course: | HNC Electrical and Electronic Engineering |
| Module: | SGEE4004 Programmable Logic Controllers |
| Assignment: | 1. Pick and Place Robot |
| Name: | Kai White |
| University number: | 2221114 |
| Date submitted: | 30/01/2023 |
| Word count: | 2500 |

# Contents

Good Structure


Contents 2

Figure List 3

1. Work Plan for Lifelong Learning 6

2. Research 7

3. Hardware Investigation and I/O List 9

4. Program Design and Explanation 10

5. Simulation Testing 23

6. Practical Testing 25

7. Discussions, Conclusions and Recommendations 28

8. Team Working 29

9. Citation List 30

10. Appendix 31

## Figure List

Figure 1 ..... 31
Figure 2 ..... 31
Figure 3 ..... 32
Figure 4 ..... 32
Figure 5 ..... 33
Figure 6 ..... 33
Figure 7 ..... 34
Figure 8 ..... 34
Figure 9 ..... 35
Figure 10 ..... 35
Figure 11 ..... 36
Figure 12 ..... 36
Figure 13 ..... 37
Figure 14 ..... 37
Figure 15 ..... 38
Figure 16 ..... 38
Figure 17 ..... 39
Figure 18 ..... 39
Figure 19 ..... 40
Figure 20 ..... 40
Figure 21 ..... 41
Figure 22 ..... 41
Figure 23 ..... 42
Figure 24 ..... 42
Figure 25 ..... 43
Figure 26 ..... 43
Figure 27 ..... 44
Figure 28 ..... 44
Figure 29 ..... 45
Figure 30 ..... 45
Figure 31 ..... 46
Figure 32 ..... 46
Figure 33 ..... 47
Figure 34 ..... 47
Figure 35 ..... 48

Figure 36 ..... 48
Figure 37 ..... 49
Figure 38 ..... 49
Figure 39 ..... 50
Figure 40 ..... 50
Figure 41 ..... 51
Figure 42 ..... 51
Figure 43 ..... 52
Figure 44 ..... 52
Figure 45 ..... 53
Figure 46 ..... 53
Figure 47 ..... 54
Figure 48 ..... 54
Figure 49 ..... 55
Figure 50 ..... 55
Figure 51 ..... 56
Figure 52 ..... 56
Figure 53 ..... 57
Figure 54 ..... 57
Figure 55 ..... 58
Figure 56 ..... 58
Figure 57 ..... 59
Figure 58 ..... 59
Figure 59 ..... 60
Figure 60 ..... 60
Figure 61 ..... 61
Figure 62 ..... 61
Figure 63 ..... 62
Figure 64 ..... 62
Figure 65 ..... 63
Figure 66 ..... 63
Figure 67 ..... 64
Figure 68 ..... 64
Figure 69 ..... 65
Figure 70 ..... 65
Figure 71 ..... 66
Figure 72 ..... 66

Figure 73 ..... 67
Figure 74 ..... 67
Figure 75 ..... 68
Figure 76 ..... 68
Figure 77 ..... 69
Figure 78 ..... 69
Figure 79 ..... 70
Figure 80 ..... 70
Figure 81 ..... 71
Figure 82 ..... 71
Figure 83 ..... 72
Figure 84 ..... 72
Figure 85 ..... 73
Figure 86 ..... 73
Figure 87 ..... 74
Figure 88 ..... 74
Figure 89 ..... 75
Figure 90 ..... 75
Figure 91 ..... 76
Figure 92 ..... 76

## 1. Work Plan for Lifelong Learning

- Research three different PLCs from a single manufacturer. Draw a comparison table and look at technical and cost implications. Also, research the pros and cons of each of the PLCs and choose which one would be suitable in a robot control panel as an alternative.
- Make an I/O list with all that's needed in words for a plan.
- Then move onto creating the program, make sure the IP addresses are correct on both HMI and PLC. Create a tag list. Structure the function calls. Make sure there's a manual and automatic mode.
- Start the program design and automatic control task by making a sequential function call and follow the brief to complete the sequence.
- PLC must contain a documentation with comments and network titles.
- After completing the PLC, simulate it to start testing. Once simulation testing has been successful, you can download the program to the robot and start practically testing it.
- Write a discussion. Conclusion and recommendations.
- Discuss the benefits of working in a team and how it helps produce a good outcome.
- Make sure that citation list is up to date.
- Produce an appendix by using your program printout with testing evidence.

*sounds like a good plan*

## 2. Research

Programmable Logic Controllers have three components which are the power supply, processor and IO (varies on the PLC) (Group, 2023) Correctly cited

Modular PLC

Is a PLC which is on a rack and you [illegible] PSU, processor, IOs and expansion units Correctly cited the same rack or other racks. (Tools, 2023) Also, multiple IO modules can be added to the rack increasing the number of IOs the PLC can use.

Compact PLC

The compact PLC has a build-in CPU, PSU IO modules and is all one unit and is normally used for small programs. The compact PLC has a fixed number of IOs and cannot be increased unlike the modular. (Volt, 2023) Correctly cited

Comparison PLC

In the table below is a comparison of three different Schneider PLC[illegible]s the Write in the third person best alternative for the robot. After all, three PLCS have been compared I will make a valid decision on which would be the best alternative for the Siemens siemens PLC being used.

| PLC | M100 Schneider PLC (TM100C24R) | M200 Schneider PLC (TM200C24R) | M222 Schneider PLC (TM221C24R) | Website |
|---|---|---|---|---|
| Inputs | 14 | 14 | 14 | (Schneider, 2023) |
| Outputs | 10 | 10 | 10 | (Schneider, 2023) |
| Input current | 7 mA for regular input<br>7 mA for fast input<br>9 mA for high-speed input | 7 mA for input | N/A | (Schneider, 2023) |
| Output voltage | 24 V DC<br>220 V AC | 24 V DC<br>220 V AC | 125 V DC<br>250 V AC | (Schneider, 2023) |
| Cost | 169.80 | £401.18 | £832.83 | (O&Kmarts, 2023) |
| Pros | • Ideal logic controller for your small automation systems requiring flexibility and simple motion<br>• Fast and easy programming with a simplified tool<br>• S-curve profiling provides smoother motion control by reducing the jerk (in silicon application) | • Ideal logic controller for your small automation systems requiring flexibility and simple motion<br>• Fast and easy programming with a simplified tool<br>• S-curve profiling provides smoother motion control by reducing the jerk (in semicon application) | • SD card and USB port<br>• 2 analogue inputs and a broad choice of I/Os<br>• Run/stop switch and cartridge extension<br>• Ethernet and serial line ports<br>• M221 controllers can connect to Modicon TM3 | (Schneider, 2023) |

| | | | | |
|---|---|---|---|---|
| | • Fast control of applications, high-speed response in packaging and conveyor applications<br>• Designed for simple machines<br>• Environmental performance of the product<br>• Cost for a small pick and place robot<br>• Up to 40IO's | • Fast control of applications, high-speed response in packaging and conveyor applications<br>• Designed for simple machines<br>• Environmental performance of the product<br>• Up to 168 IO's | expansion modules for local, remote or distributed I/O configurations<br>• Environmental performance of the product | |

*From this table, it can be seen that* Write in the third person

From this table I have decided that the best alternative to the "Siemens S7-1214 PLC" out of these three Schneider PLCs would be "M100 Schneider PLC (TM100C24R)". Mainly because it's most suitable for the pick and place robot because the others would be overboard on cost and only 22 IO's are needed so both the other options would just be wasting space and money.

Good conclusions

## 3. Hardware Investigation and I/O List

*Where is the rest of your system explanation? Have you read the brief?*

| Inputs | |
|---|---|
| **Address** | **Description** |
| %I0.0 | UP SENSOR |
| %I0.1 | DOWN SENSOR |
| %I0.2 | GRIPPER OPEN SENSOR |
| %I0.3 | GRIPPER CLOSED SENSOR |
| %I0.4 | IN SENSOR |
| %I0.5 | OUT SENSOR |
| %I0.6 | RIGHT SENSOR |
| %I0.7 | LEFT SENSOR |
| %I1.0 | PUSHBUTTON X1O |
| %I1.1 | PUSHBUTTON X11 |
| %I1.2 | PUSHBUTTON X12 |
| %I1.3 | TOGGLE SWITCH X13 |
| %I1.4 | TOGGLE SWITCH X14 |
| %I1.5 | OBJECT SENSOR |

| Outputs | |
|---|---|
| **Address** | **Description** |
| %Q0.0 | MOVE DOWN SOLENOID |
| %Q0.1 | MOVE UP SOLENOID |
| %Q0.2 | CLOSE GRIPPER SOLENOID |
| %Q0.3 | OPEN GRIPPER SOLENOID |
| %Q0.4 | MOVE IN SOLENOID |
| %Q0.5 | MOVE OUT SOLENOID |
| %Q0.6 | MOVE LEFT SOLENOID |
| %Q0.7 | MOVE RIGHT SOLENOID |

## 4. Program Design and Explanation

**Tag list**

*Your explanations seem to be limited to program comments*

IO tags

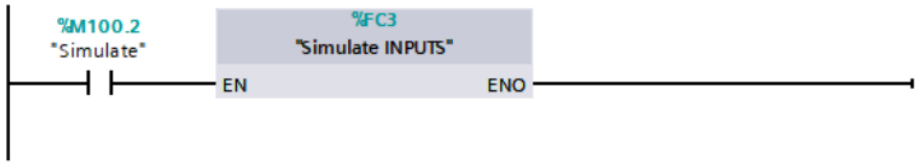
| | Name | Data type | Address | Retain | Acces... | Writa... | Visibl... |
|---|---|---|---|---|---|---|---|
| 1 | UP sensor | Bool | %I0.0 | | ☑ | ☑ | ☑ |
| 2 | DOWN sensor | Bool | %I0.1 | | ☑ | ☑ | ☑ |
| 3 | OPEN sensor | Bool | %I0.2 | | ☑ | ☑ | ☑ |
| 4 | CLOSED sensor | Bool | %I0.3 | | ☑ | ☑ | ☑ |
| 5 | IN sensor | Bool | %I0.4 | | ☑ | ☑ | ☑ |
| 6 | OUT sensor | Bool | %I0.5 | | ☑ | ☑ | ☑ |
| 7 | RIGHT sensor | Bool | %I0.6 | | ☑ | ☑ | ☑ |
| 8 | LEFT sensor | Bool | %I0.7 | | ☑ | ☑ | ☑ |
| 9 | X10 Pushbutton | Bool | %I1.0 | | ☑ | ☑ | ☑ |
| 10 | X11 Pushbutton | Bool | %I1.1 | | ☑ | ☑ | ☑ |
| 11 | X12 Pushbutton | Bool | %I1.2 | | ☑ | ☑ | ☑ |
| 12 | X13 toggle switch | Bool | %I1.3 | | ☑ | ☑ | ☑ |
| 13 | X14 toggle switch | Bool | %I1.4 | | ☑ | ☑ | ☑ |
| 14 | OBJECT sensor | Bool | %I1.5 | | ☑ | ☑ | ☑ |
| 15 | DOWN solenoid | Bool | %Q0.0 | | ☑ | ☑ | ☑ |
| 16 | UP solenoid | Bool | %Q0.1 | | ☑ | ☑ | ☑ |
| 17 | CLOSE solenoid | Bool | %Q0.2 | | ☑ | ☑ | ☑ |
| 18 | OPEN solenoid | Bool | %Q0.3 | | ☑ | ☑ | ☑ |
| 19 | OUT solenoid | Bool | %Q0.4 | | ☑ | ☑ | ☑ |
| 20 | IN solenoid | Bool | %Q0.5 | | ☑ | ☑ | ☑ |
| 21 | LEFT solenoid | Bool | %Q0.6 | | ☑ | ☑ | ☑ |
| 22 | RIGHT solenoid | Bool | %Q0.7 | | ☑ | ☑ | ☑ |
| 23 | <Add new> | | | | ☑ | ☑ | ☑ |

My Tags

| | Name | Data type | Address | Retain | Acces... | Writa... | Visibl... |
|---|---|---|---|---|---|---|---|
| 1 | UP HMI request | Bool | %M100.0 | | ☑ | ☑ | ☑ |
| 2 | DOWN HMI request | Bool | %M100.1 | | ☑ | ☑ | ☑ |
| 3 | Simulate | Bool | %M100.2 | | ☑ | ☑ | ☑ |
| 4 | IN HMI request | Bool | %M100.3 | | ☑ | ☑ | ☑ |
| 5 | OUT HMI request | Bool | %M100.4 | | ☑ | ☑ | ☑ |
| 6 | LEFT HMI request | Bool | %M100.5 | | ☑ | ☑ | ☑ |
| 7 | RIGHT HMI request | Bool | %M100.6 | | ☑ | ☑ | ☑ |
| 8 | OPEN HMI request | Bool | %M100.7 | | ☑ | ☑ | ☑ |
| 9 | CLOSE HMI request | Bool | %M101.0 | | ☑ | ☑ | ☑ |
| 10 | Auto mode | Bool | %M101.1 | | ☑ | ☑ | ☑ |
| 11 | UP AUTO request | Bool | %M101.2 | | ☑ | ☑ | ☑ |
| 12 | DOWN AUTO request | Bool | %M101.3 | | ☑ | ☑ | ☑ |
| 13 | IN AUTO request | Bool | %M101.4 | | ☑ | ☑ | ☑ |
| 14 | OUT AUTO request | Bool | %M101.5 | | ☑ | ☑ | ☑ |
| 15 | LEFT AUTO request | Bool | %M101.6 | | ☑ | ☑ | ☑ |
| 16 | RIGHT AUTO request | Bool | %M101.7 | | ☑ | ☑ | ☑ |
| 17 | OPEN AUTO request | Bool | %M102.0 | | ☑ | ☑ | ☑ |
| 18 | CLOSE AUTO request | Bool | %M102.1 | | ☑ | ☑ | ☑ |
| 19 | START SEQUENCE | Bool | %M110.0 | | ☑ | ☑ | ☑ |
| 20 | STEP | UInt | %MW200 | | ☑ | ☑ | ☑ |
| 21 | next_STEP | UInt | %MW202 | | ☑ | ☑ | ☑ |
| 22 | START MEMORY | Bool | %M111.0 | | ☑ | ☑ | ☑ |
| 23 | HMI Start request | Bool | %M111.1 | | ☑ | ☑ | ☑ |
| 24 | Pause | Bool | %M111.2 | | ☑ | ☑ | ☑ |

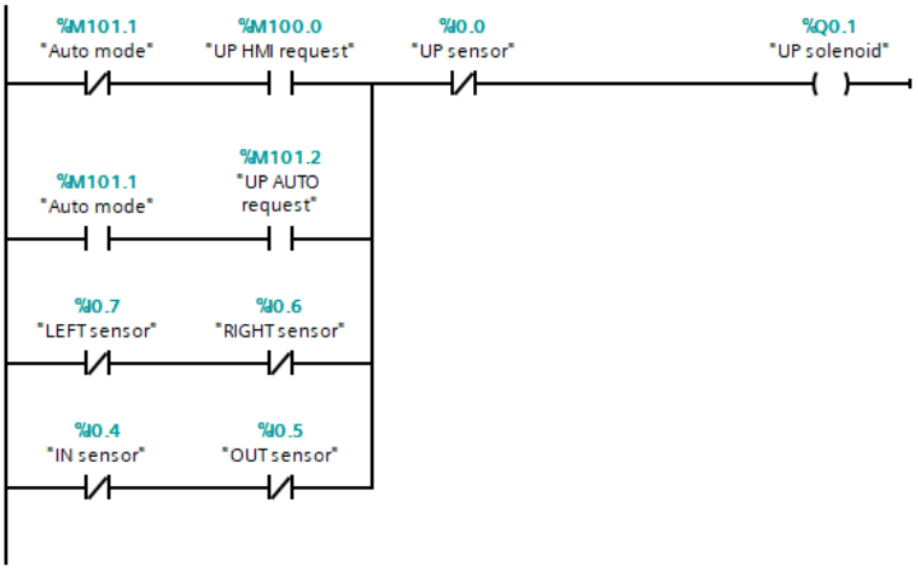
**Network 1:** Switch between simulation and actual inputs

Using the simulate bit, when it is off, the IO are read into memory bits. When the simulate bit is on, the inputs are simulated by the supplied function call.



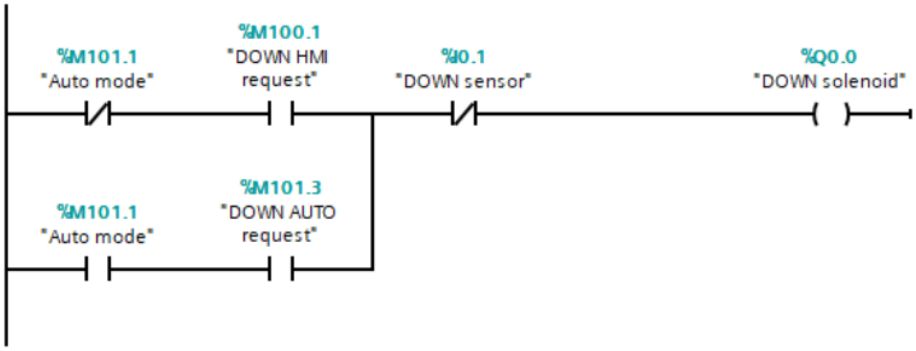

**Network 2:** Move up with SAFETY interlocks

When the robot isn't on any of these sesnors it will move up.



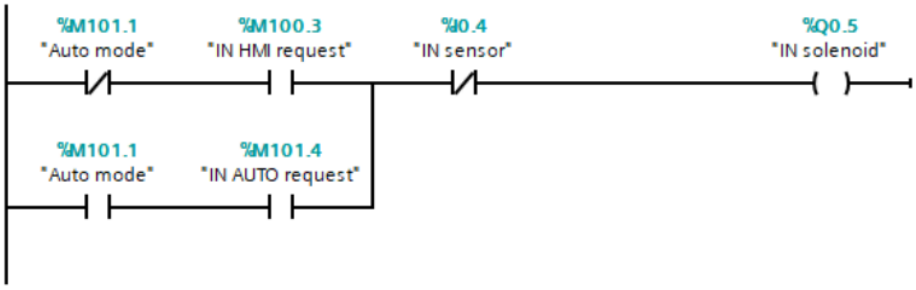
**Network 3:** Move down

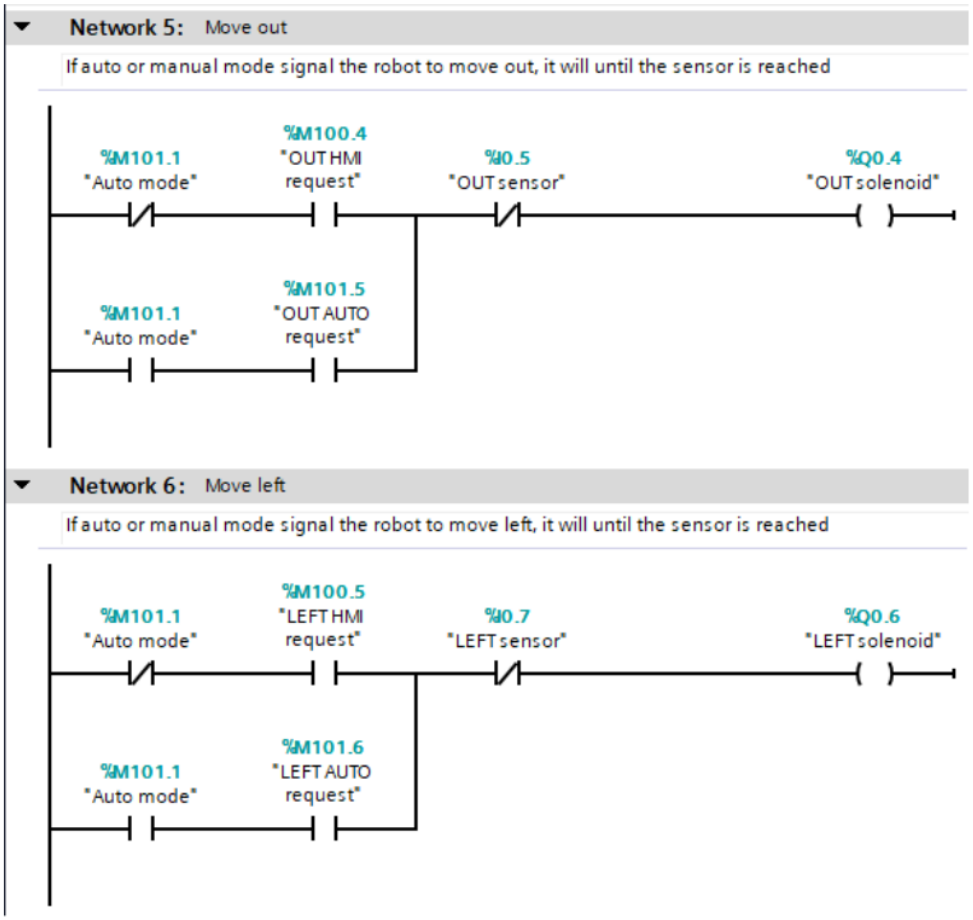
If auto or manual mode signal the robot to move down, it will until the sensor is reached




**Network 4:** Move in

If auto or manual mode signal the robot to move in, it will until the sensor is reached

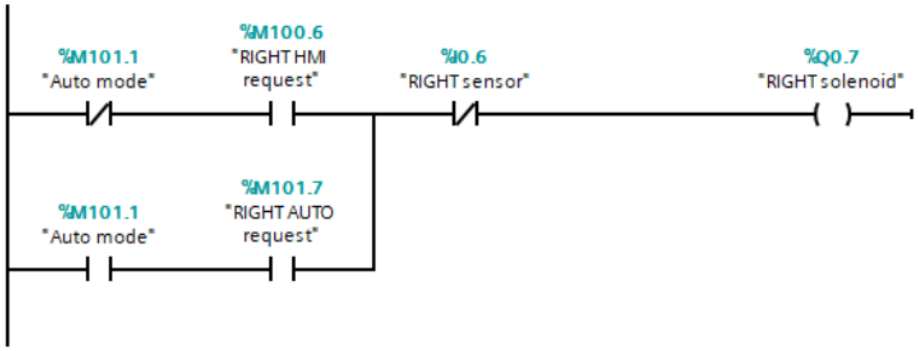





*Good program comments but you should have included an explanation*

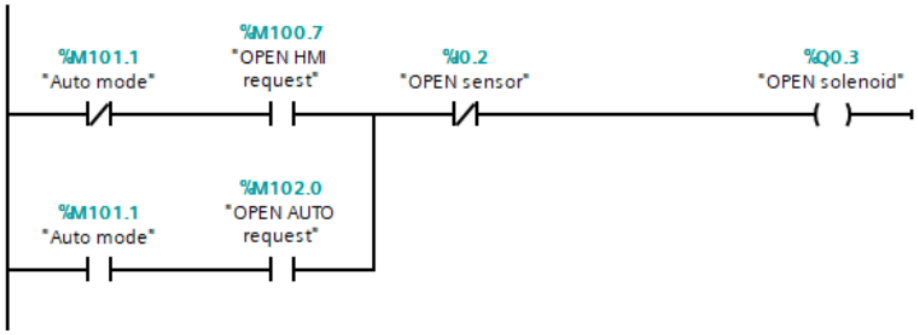
**Network 7:** Move right

If auto or manual mode signal the robot to move right, it will until the sensor is reached



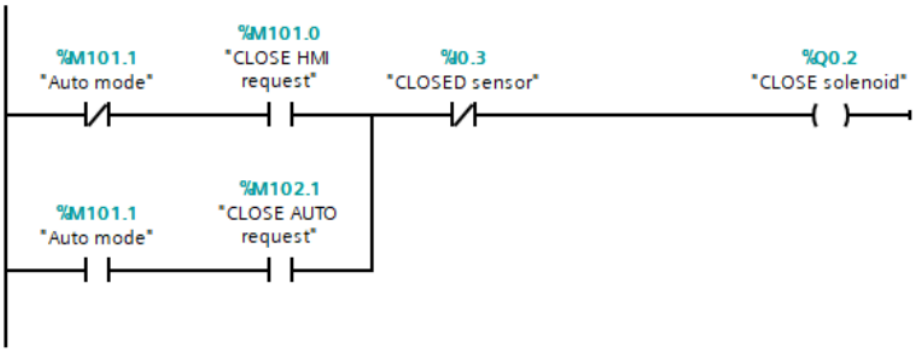

**Network 8:** Open gripper

If auto or manual mode signal the robot to open the gripper, it will until the sensor is reached



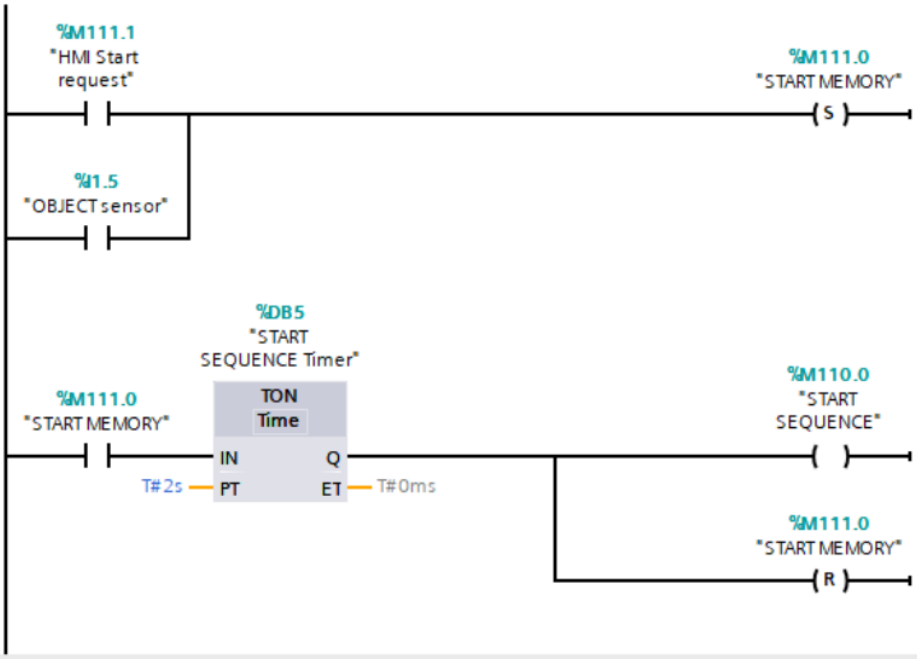
**Network 9:** Close gripper

If auto or manual mode signal the robot to close the gripper, it will until the sensor is reached



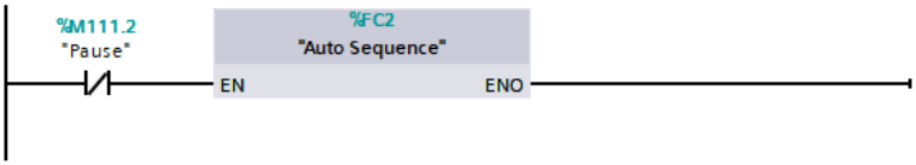

**Network 10:** Start sequence after 2 seconds

When start has been actiavted or an object has fallen in, the timer will activate and change every 2 seconds. Moving through every step with a two second delay.



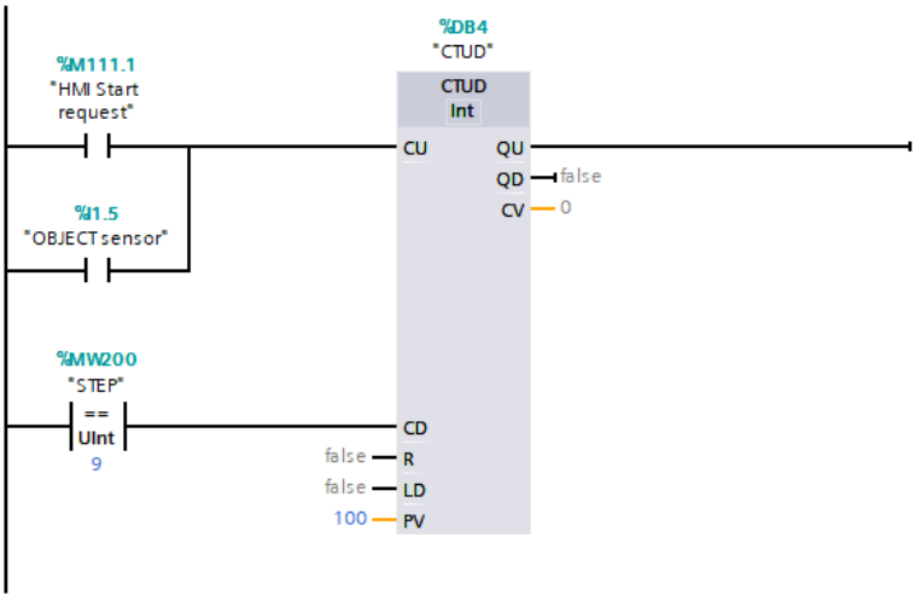
**Network 11:** Pause auto sequence

When going through the steps, it can be paused at any time or continued.




**Network 12:** Counter for number of items in machine

Counter for number of objects in the machine. Counts down at step 9 because thats when a item is removed from the machine.



**Auto sequence flow chart below.**



The flow chart is a detailed plan before code is written so that it reduces the amount or errors and mistakes in the code making it easier when testing the code. If no flow chart or plan was made before coding it would take a lot more time and effort to get everything correct with no faults, and it will most probably lead into doing a lot of testing to see what is wrong with your code. SO, planning is key.

Auto sequence code

```
IF ("Auto mode" = 0) THEN
    "STEP" := 0;
    "next_STEP" := 0;
    "RIGHT AUTO request" := 0;
    "LEFT AUTO request" := 0;                                                    // Resets all
    "OUT AUTO request" := 0;
    "IN AUTO request" := 0;
    "OPEN AUTO request" := 0;
    "CLOSE AUTO request" := 0;
    "UP AUTO request" := 0;
    "DOWN AUTO request" := 0;
END_IF;

IF ("Auto mode" = 1) THEN                                                        //When auto mode is selected
    CASE "STEP" OF
        0:
            "UP AUTO request" := 1;                                              //Makes sure the robot is up before being able to press any other movements
            IF "UP sensor" = 1 THEN
                "next_STEP" := 1;
                "UP AUTO request" := 0;
            END_IF;
        1:
            "LEFT AUTO request" := 1;
            "OUT AUTO request" := 1;
            "OPEN AUTO request" := 1;
            IF ("LEFT sensor" = 1) AND ("OUT sensor" = 1) AND ("OPEN sensor" = 1) AND ("START SEQUENCE" = 1) THEN
                IF ("CTUD".CV = 1) THEN                                          // if counter is = 1 skip to STEP 18
                    "next_STEP" := 18;
                END_IF;
                IF ("CTUD".CV = 2) THEN                                          // if counter is = 2 skip to STEP 9
                    "next_STEP" := 9;
                END_IF;
                IF ("CTUD".CV = 3) THEN                                          // if counter is = 3 skip to STEP 2
                    "next_STEP" := 2;
                END_IF;
                "LEFT AUTO request" := 0;
                "OUT AUTO request" := 0;
                "OPEN AUTO request" := 0;
            END_IF;
```

```
        2:                                                                  //Move right - Move object from station 3 to 4
            "RIGHT AUTO request" := 1;
            IF ("RIGHT sensor" = 1) THEN
                "next_STEP" := 3;
                "RIGHT AUTO request" := 0;
            END_IF;
        3:                                                                  //Move down
            "DOWN AUTO request" := 1;
            IF ("DOWN sensor" = 1) THEN
                "next_STEP" := 4;
                "DOWN AUTO request" := 0;
            END_IF;
        4:                                                                  //Close gripper
            "CLOSE AUTO request" := 1;
            IF ("CLOSED sensor" = 1) THEN
                "next_STEP" := 5;
                "CLOSE AUTO request" := 0;
            END_IF;
        5:                                                                  //Move up
            "UP AUTO request" := 1;
            IF ("UP sensor" = 1) THEN
                "next_STEP" := 6;
                "UP AUTO request" := 0;
            END_IF;
        6:                                                                  //Move in
            "IN AUTO request" := 1;
            IF ("IN sensor" = 1) THEN
                "next_STEP" := 7;
                "IN AUTO request" := 0;
            END_IF;
        7:                                                                  //Move down
            "DOWN AUTO request" := 1;
            IF ("DOWN sensor" = 1) THEN
                "next_STEP" := 8;
                "DOWN AUTO request" := 0;
            END_IF;
        8:                                                                   //Open gripper
            "OPEN AUTO request" := 1;
            IF ("OPEN sensor" = 1) THEN
                "next_STEP" := 9;
                "OPEN AUTO request" := 0;
            END_IF;
        9:                                                                  // Move up
            "UP AUTO request" := 1;
            IF ("UP sensor" = 1) THEN
                "next_STEP" := 10;
                "UP AUTO request" := 0;
            END_IF;
        10:                                                                 // Move left - Move object from station 2 to 3
            "LEFT AUTO request" := 1;
            IF ("LEFT sensor" = 1) THEN
                "next_STEP" := 11;
                "LEFT AUTO request" := 0;
            END_IF;
        11:                                                                 //Move out
            "OUT AUTO request" := 1;
            IF ("OUT sensor" = 1) THEN
                "next_STEP" := 12;
                "OUT AUTO request" := 0;
            END_IF;
        12:                                                                //Move down
            "DOWN AUTO request" := 1;
            IF ("DOWN sensor" = 1) THEN
                "next_STEP" := 13;
                "DOWN AUTO request" := 0;
            END_IF;
        13:                                                                //Close gripper
            "CLOSE AUTO request" := 1;
            IF ("CLOSED sensor" = 1) THEN
                "next_STEP" := 14;
                "CLOSE AUTO request" := 0;
            END_IF;
```

```
        14:                                                                     //Move up
            "UP AUTO request" := 1;
            IF ("UP sensor" = 1) THEN
                "next_STEP" := 15;
                "UP AUTO request" := 0;
            END_IF;
        15:                                                                     //Move right
            "RIGHT AUTO request" := 1;
            IF ("RIGHT sensor" = 1) THEN
                "next_STEP" := 16;
                "RIGHT AUTO request" := 0;
            END_IF;
        16:                                                                     //Move down
            "DOWN AUTO request" := 1;
            IF ("DOWN sensor" = 1) THEN
                "next_STEP" := 17;
                "DOWN AUTO request" := 0;
            END_IF;
        17:                                                                     //Open gripper
            "OPEN AUTO request" := 1;
            IF ("OPEN sensor" = 1) THEN
                "next_STEP" := 18;
                "OPEN AUTO request" := 0;
            END_IF;
        18:                                                                     //Move up
            "UP AUTO request" := 1;
            IF ("UP sensor" = 1) THEN
                "next_STEP" := 19;
                "UP AUTO request" := 0;
            END_IF;
        19:                                                                     //Move left - Move object from station 1 to 2
            "LEFT AUTO request" := 1;
            IF ("LEFT sensor" = 1) THEN
                "next_STEP" := 20;
                "LEFT AUTO request" := 0;
            END_IF;
         20:                                                                                    //Move in
             "IN AUTO request" := 1;
             IF ("IN sensor" = 1) THEN
                 "next_STEP" := 21;
                 "IN AUTO request" := 0;
             END_IF;
         21:                                                                                    //Move down
             "DOWN AUTO request" := 1;
             IF ("DOWN sensor" = 1) THEN
                 "next_STEP" := 22;
                 "DOWN AUTO request" := 0;
             END_IF;
         22:                                                                                    //Close gripper
             "CLOSE AUTO request" := 1;
             IF ("CLOSED sensor" = 1) THEN
                 "next_STEP" := 23;
                 "CLOSE AUTO request" := 0;
             END_IF;
         23:                                                                                    //Move up
             "UP AUTO request" := 1;
             IF ("UP sensor" = 1) THEN
                 "next_STEP" := 24;
                 "UP AUTO request" := 0;
             END_IF;
         24:                                                                                    //Move out
             "OUT AUTO request" := 1;
             IF ("OUT sensor" = 1) THEN
                 "next_STEP" := 25;
                 "OUT AUTO request" := 0;
             END_IF;
         25:                                                                                    //Move down
             "DOWN AUTO request" := 1;
             IF ("DOWN sensor" = 1) THEN
                 "next_STEP" := 26;
                 "DOWN AUTO request" := 1;
             END_IF;
```

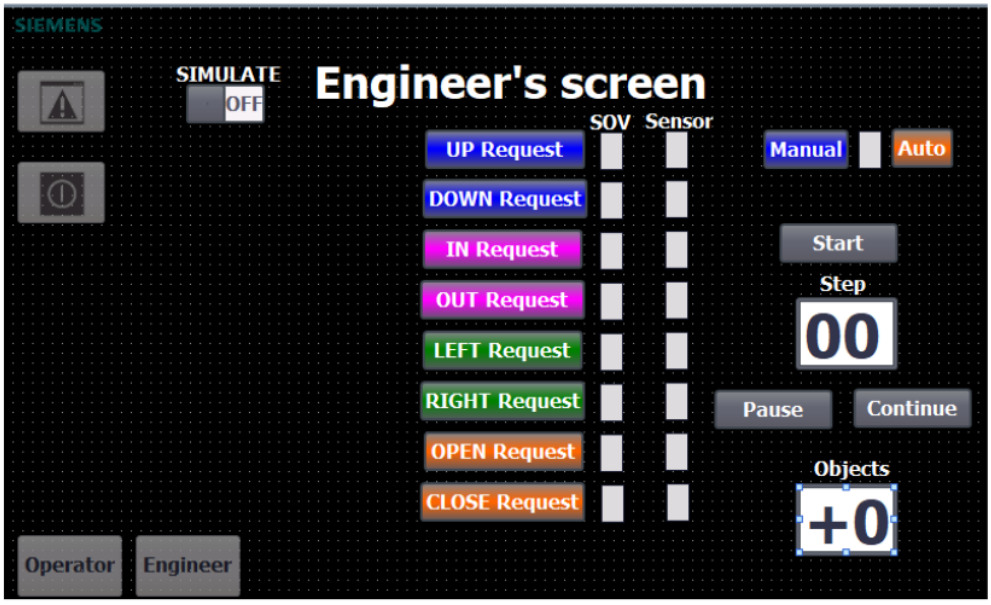
```
184         26:                                                                                    //Open gripper
185             "OPEN AUTO request" := 1;
186             IF ("OPEN sensor" = 1) THEN
187                 "next_STEP" := 0;                                                              //Restart the cycle by moving to STEP 0
188                 "OPEN AUTO request" := 0;
189             END_IF;
190     END_CASE;
191     "STEP" := "next_STEP";
192 END_IF;
```

Always comment on code so if someone else works on the code they will understand what each part of code does and they will be able to interpret it.

*Your code contains comments but where is your explanation?*

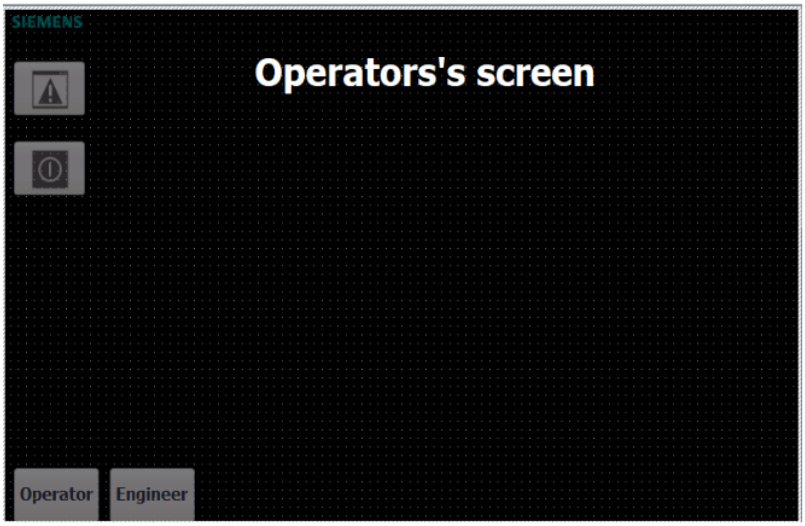
## HMI SCREENS

### Engineering Screen




Only use simulate when simulating.

### Operating Screen



## 5. Simulation Testing

The table below shows all the tests carried out on the TIA portal using simulation mode on my program to look for errors in the code or the ladder. So, that when testing the program with the machine, it will not get damaged because of the safety interlocks in place and testing in simulation mode has been completed. Evidence must be available to proof that the test has been completed and was successful.

| **Test description** | **Expected Results** | **Actual Results** | **Decision** | **Evidence** |
|---|---|---|---|---|
| **Auto Mode** | | | | |
| Start button | When pressed on step 1 should progress through steps | Pressed the start button and progressed through all the steps | Passed | Figure 1 |
| Step 0 | Should always move up | Moved up | Passed | Figure 2 |
| Step 1, press start with auto mode on<br><br>If counter = 1<br><br>If counter = 2<br><br><br>If counter = 3 | Should progress through all the steps<br>Move to STEP 20<br><br>Move to STEP 12<br><br><br>Move to STEP 2 | Progressed through all the steps<br>Moved to STEP 20<br><br>Moved to STEP 12<br><br><br>Moved to STEP 2 | Passed | Figure 3<br>Figure 4<br>Figure 5 |
| Step 2 | Move right | Moved right | Passed | Figure 6 |
| Step 3 | Move down | Moved down | Passed | Figure 7 |
| Step 4 | Close gripper | Closed gripper | Passed | Figure 8 |
| Step 5 | Move up | Moved up | Passed | Figure 9 |
| Step 6 | Move in | Moved in | Passed | Figure 10 |
| Step 7 | Move down | Moved down | Passed | Figure 11 |
| Step 8 | Open gripper | Opened gripper | Passed | Figure 12 |
| Step 9 | Move up | Moved up | Passed | Figure 13 |
| Step 10 | Move left | Moved left | Passed | Figure 14 |
| Step 11 | Move out | Moved out | Passed | Figure 15 |
| Step 12 | Move down | Moved down | Passed | Figure 16 |
| Step 13 | Close gripper | Closed gripper | Passed | Figure 17 |
| Step 14 | Move up | Moved up | Passed | Figure 18 |
| Step 15 | Move right | Moved right | Passed | Figure 19 |
| Step 16 | Move down | Moved down | Passed | Figure 20 |
| Step 17 | Open gripper | Opened gripper | Passed | Figure 21 |

| | | | | |
|---|---|---|---|---|
| Step 18 | Move up | Moved up | Passed | Figure 22 |
| Step 19 | Move left | Moved left | Passed | Figure 23 |
| Step 20 | Move in | Moved in | Passed | Figure 24 |
| Step 21 | Move down | Moved down | Passed | Figure 25 |
| Step 22 | Close gripper | Closed gripper | Passed | Figure 26 |
| Step 23 | Move up | Moved up | Passed | Figure 27 |
| Step 24 | Move out | Moved out | Passed | Figure 28 |
| Step 25 | Move down | Moved down | Passed | Figure 29 |
| Step 26 | Open gripper Then | Opened gripper | Passed | Figure 30 |
| Counter | Counter should progress through the steps | Counter progressed through the steps | Passed | Figure 31 |
| Pause Button | Pause button should pause the step it's on when pressed | Paused the current step | Passed | Figure 32 |
| Continue button | Should resume the step progression | Resumed the step progression | Passed | Figure 33 |
| **Manual Mode** | | | | |
| Interlocks | If the arm isn't on any sensors it will move up | Arm moves up when it isn't on a sensor | Passed | Figure 34 |
| Move right | Should move right | Moved right | Passed | Figure 35 |
| Move left | Should move left | Moved left | Passed | Figure 36 |
| Move up | Should move up | Moved up | Passed | Figure 37 |
| Move down | Should move down | Moved down | Passed | Figure 38 |
| Move in | Should move in | Moved in | Passed | Figure 39 |
| Move out | Should move out | Moved out | Passed | Figure 40 |
| Open gripper | Should open gripper | Opened gripper | Passed | Figure 41 |
| Close gripper | Should close gripper | Closed gripper | Passed | Figure 42 |
| Switch between Engineer's and Operator's screen | Should be able to switch between the two | Switched between both the screens | Passed | Figure 43<br>Figure 44 |
| Solenoid and Sensor lights | Solenoid and sensor lights should appear when activated | Light came on when solenoids were being used and sensors were activated | Passed | Figure 46 |

## 6. Practical Testing

The table below shows all the tests carried out on the actual robot to look for errors in the code or the ladder. Evidence must be available to proof that the test has been completed and was successful.

| Test description | Expected Results | Actual Results | Decision | Evidence |
|---|---|---|---|---|
| **Auto Mode** | | | | |
| Start button | When pressed on step 1 should progress through steps | Pressed the start button and progressed through all the steps | Passed | Figure 46 |
| Step 0 | Should always move up | Moved up | Passed | Figure 47 |
| Step 1, press start with auto mode on<br>If counter = 1<br><br><br>If counter = 2<br><br><br>If counter = 3 | Should progress through all the steps<br>Move to STEP 20<br>Move to STEP 12<br><br>Move to STEP 2 | Progressed through all the steps<br>Moved to STEP 20<br><br>Moved to STEP 12<br><br>Moved to STEP 2 | Passed | Figure 48<br>Figure 49<br>Figure 50 |
| Step 2 | Move right | Moved right | Passed | Figure 51 |
| Step 3 | Move down | Moved down | Passed | Figure 52 |
| Step 4 | Close gripper | Closed gripper | Passed | Figure 53 |
| Step 5 | Move up | Moved up | Passed | Figure 54 |
| Step 6 | Move in | Moved in | Passed | Figure 55 |
| Step 7 | Move down | Moved down | Passed | Figure 56 |
| Step 8 | Open gripper | Opened gripper | Passed | Figure 57 |
| Step 9 | Move up | Moved up | Passed | Figure 58 |
| Step 10 | Move left | Moved left | Passed | Figure 59 |
| Step 11 | Move out | Moved out | Passed | Figure 60 |

| | | | | |
|---|---|---|---|---|
| Step 12 | Move down | Moved down | Passed | Figure 61 |
| Step 13 | Close gripper | Closed gripper | Passed | Figure 62 |
| Step 14 | Move up | Moved up | Passed | Figure 63 |
| Step 15 | Move right | Moved right | Passed | Figure 64 |
| Step 16 | Move down | Moved down | Passed | Figure 65 |
| Step 17 | Open gripper | Opened gripper | Passed | Figure 66 |
| Step 18 | Move up | Moved up | Passed | Figure 67 |
| Step 19 | Move left | Moved left | Passed | Figure 68 |
| Step 20 | Move in | Moved in | Passed | Figure 69 |
| Step 21 | Move down | Moved down | Passed | Figure 70 |
| Step 22 | Close gripper | Closed gripper | Passed | Figure 71 |
| Step 23 | Move up | Moved up | Passed | Figure 72 |
| Step 24 | Move out | Moved out | Passed | Figure 73 |
| Step 25 | Move down | Moved down | Passed | Figure 74 |
| Step 26 | Open gripper Then | Opened gripper | Passed | Figure 75 |
| Counter | Counter should progress through the steps | Counter progressed through the steps | Passed | Figure 76 |
| Pause Button | Pause button should pause the step it's on when pressed | Paused the current step | Passed | Figure 77 |
| Continue button | Should resume the step progression | Resumed the step progression | Passed | Figure 78 |
| Manual Mode | | | | |
| Interlocks | If the arm isn't on any sensors it will move up | Arm moves up when it isn't on a sensor | Passed | Figure 79<br>Figure 80<br>Figure 81 |
| Move right | Should move right | Moved right | Passed | Figure 82 |
| Move left | Should move left | Moved left | Passed | Figure 83 |
| Move up | Should move up | Moved up | Passed | Figure 84 |
| Move down | Should move down | Moved down | Passed | Figure 85 |

| | | | | |
|---|---|---|---|---|
| Move in | Should move in | Moved in | Passed | Figure 86 |
| Move out | Should move out | Moved out | Passed | Figure 87 |
| Open gripper | Should open gripper | Opened gripper | Passed | Figure 88 |
| Close gripper | Should close gripper | Closed gripper | Passed | Figure 89 |
| Switch between Engineer's and Operator's screen | Should be able to switch between the two | Switched between both the screens | Passed | Figure 90<br>Figure 91 |
| Solenoid and Sensor lights | Solenoid and sensor lights should appear when activated | Light came on when solenoids were being used and sensors were activated | Passed | Figure 92 |

## 7. Discussions, Conclusions and Recommendations

*The results .....* Write in the third person *a testing table was used* *the*

My results were all successful and to prove that I used the testing table to ensure that my end product did what I wanted it to with no errors. If there would be any errors I would of found them through my simulation testing table when testing.

*three different PLCs were researched*

To conclude I researched 3 different PLCS found out which one would be the most suitable alternative to the one we are using in the robot. Then planning was done by making IO tables and flow charts. Next, I started to build my program with screenshots so show how I done it and explanations on what the screenshots show building up a step-to-step guide on how to build my program. In my code for the "auto sequence" I left loads of comments showing what each bit of code does, so any one can understand it. Progressing, testing had to be done so I made a testing table for both simulation and practical testing and evidence from when I carried it out. Write in the third person

When I was coding, I found a bug where the robot would not go back to "STEP 0" ending on "STEP 6" this took a while to solve but at the end I did by bracketing all the steps in a if statement as shown below.

```
14 IF ("Auto mode" = 1) THEN
15     CASE "STEP" OF
```

*this is a good bug fix*

If I was to do this again, I would make a more detailed plan of my "auto sequence" code so it would be easier for me to write it up and cause less mistakes saving a lot of time simulating and going through my program to fix the problem. Next time, to improve my skills I would add a 3x3 pick and place robot instead of a 2x2, which will complicate things and progress my understanding of programmable logic controllers.

*You have learnt a lot from this assignment*

## 8. Team Working

I worked in a team of three, to start off we had three separate programs and we compared all of them to see which one was ready for testing tested, we did this by using simulation mode and seeing if the program worked successfully. So, then we all took photos for evidence in our testing tables. As a team I feel we are contributed fairly and all had valuable inputs to the task.

I felt that working in a team benefited me as we could brainstorm ideas if we came across any faults to resolve it at a quicker pace. Also, the others might have had different testing procedures to me so we could do them on the program as well, to push the program to its limits and make sure it doesn't break.

## 9. Citation List

*a range of source correctly listed*


Group, T., 2023. *Technique Group*. [Online] Available at: https://learntechnique.com/programmable-logic-controllers-plc-programming-plc-components/#:~:text=Programmable%20Logic%20Controllers%20have%20three,(I%2FO)%20section.
[Accessed 30 01 2023].

O&Kmarts, 2023. *O&KMarts*. [Online] Available at: https://okmarts.com/schneider-programmable-controllers-plc-tm200c24r.html
[Accessed 27 01 2023].

Schneider, 2023. *Schneider*. [Online] Available at: https://www.se.com/uk/en/
[Accessed 27 01 2023].

Tools, I., 2023. *Inst Tools*. [Online] Available at: https://instrumentationtools.com/difference-between-compact-plc-and-modular-plc/#:~:text=Modular%20Type%20PLC%20is%20a,number%20of%20inputs%2C%20and%20outputs.
[Accessed 2023 01 2023].

Volt, E., 2023. *Electrical Volt*. [Online] Available at: https://www.electricalvolt.com/2022/08/difference-between-compact-plc-and-modular-plc/
[Accessed 30 01 2023].

## 10. Appendix

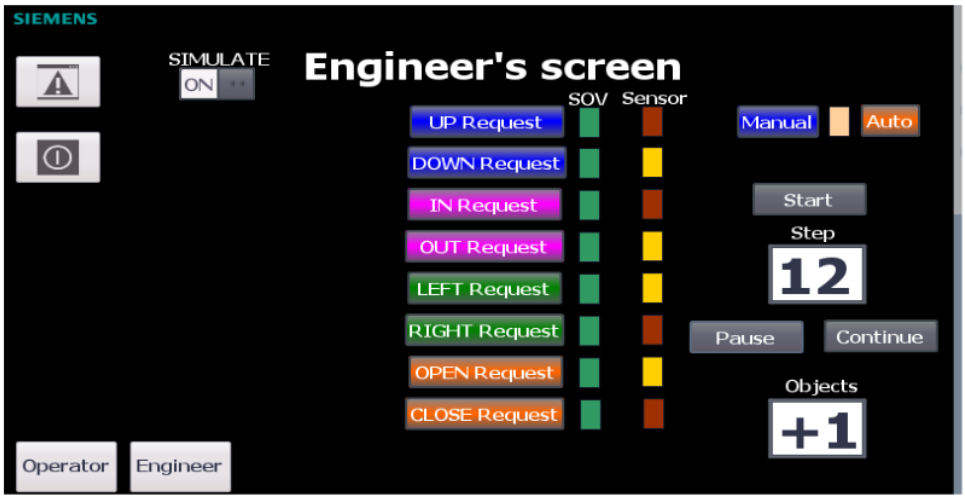



*Figure 1*

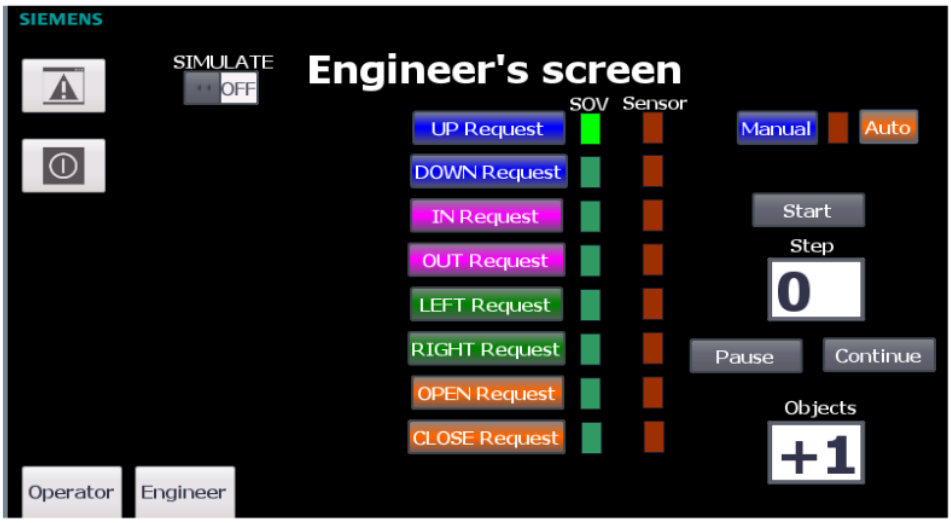



*Figure 2*

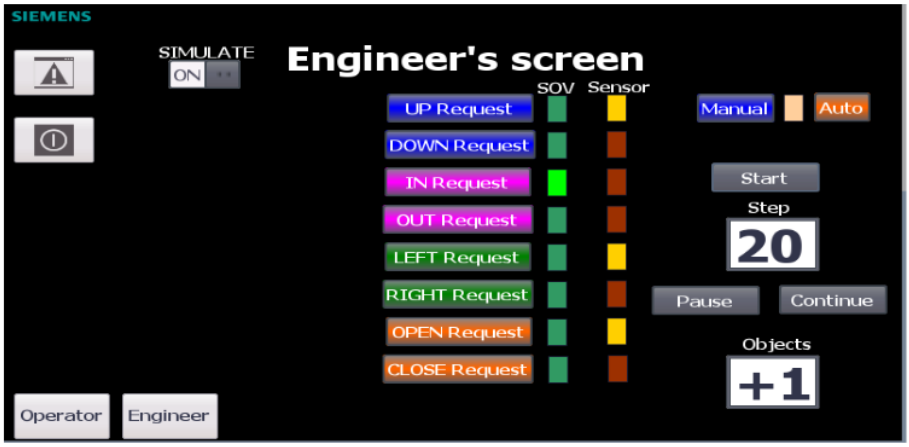



*Figure 3*

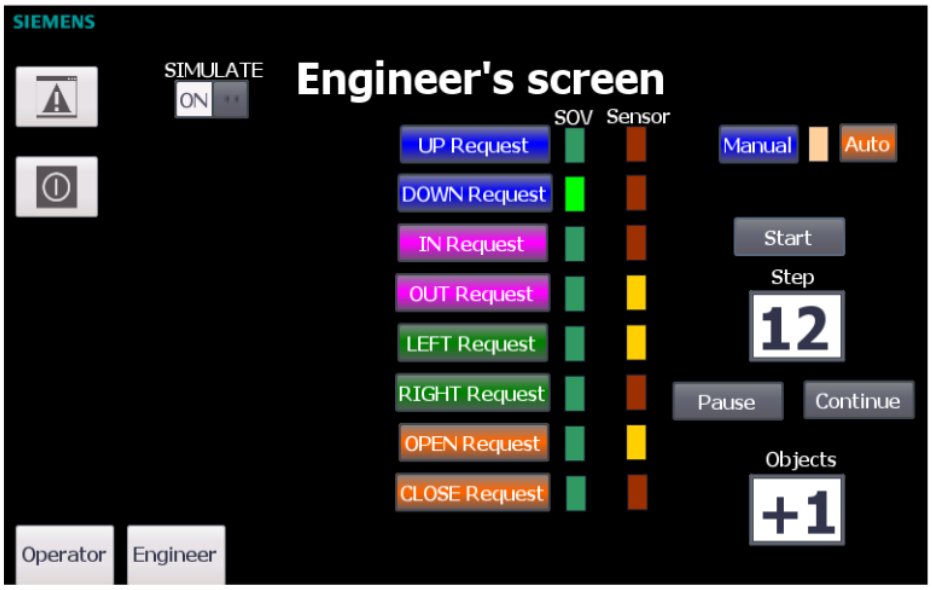



*Figure 4*

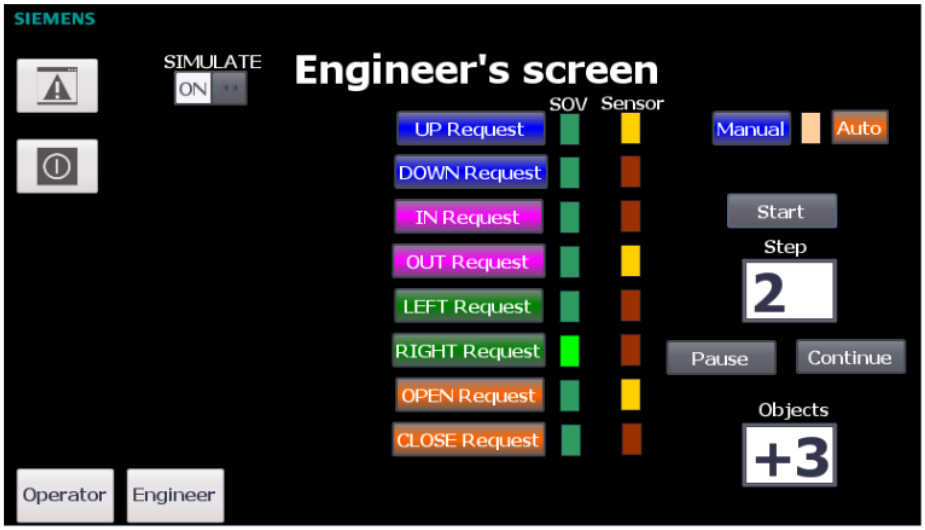



*Figure 5*

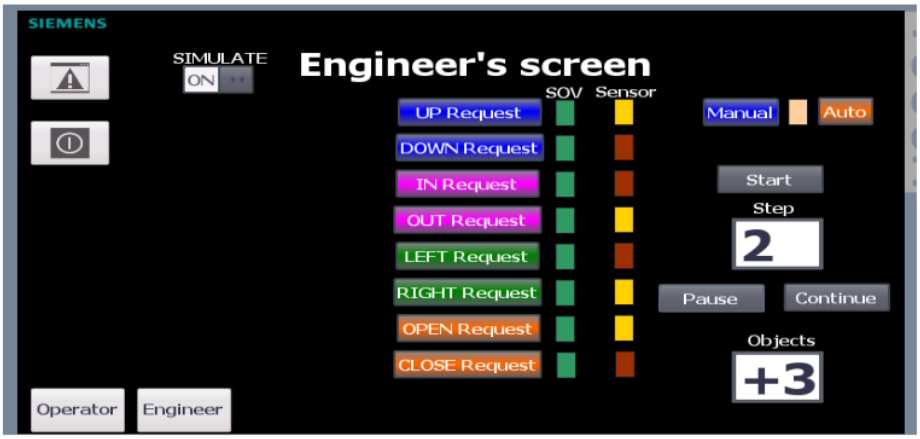



*Figure 6*

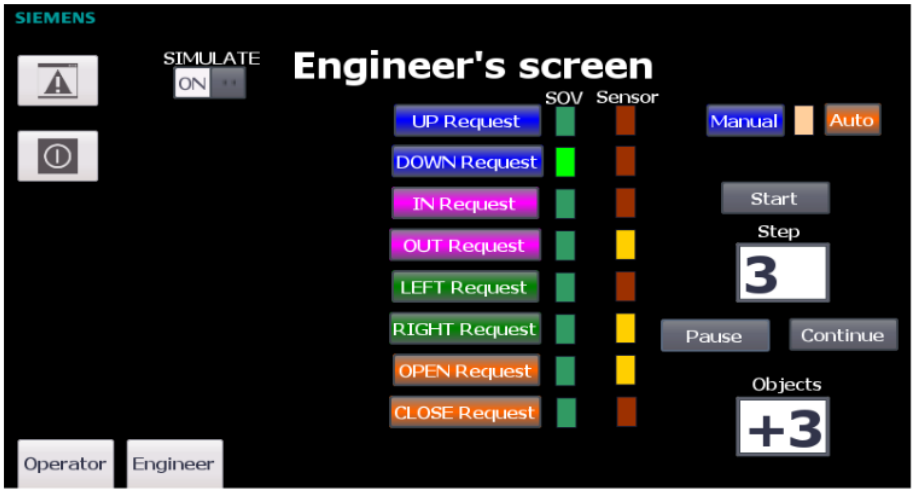



*Figure 7*

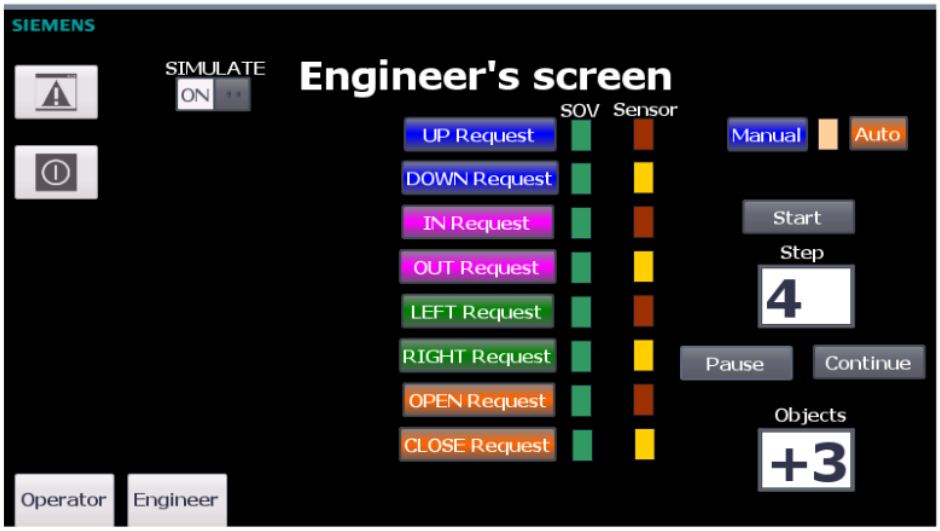



*Figure 8*

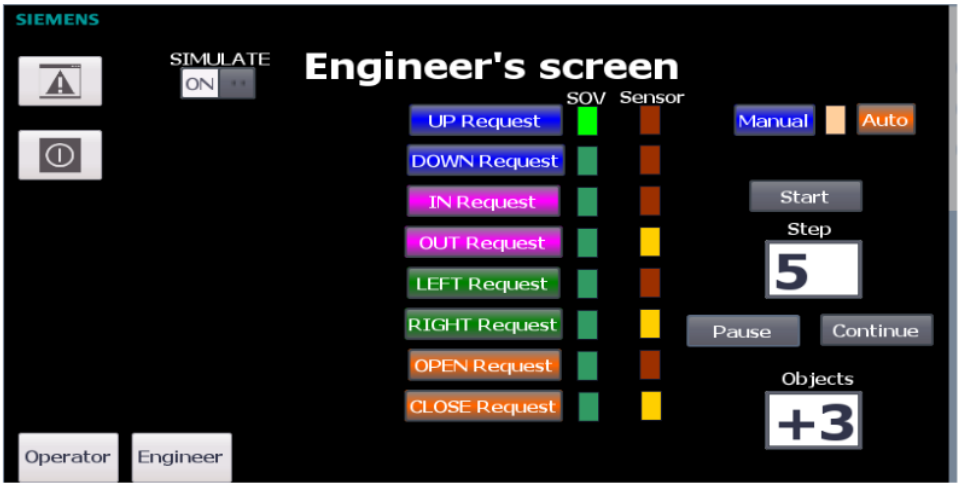



*Figure 9*

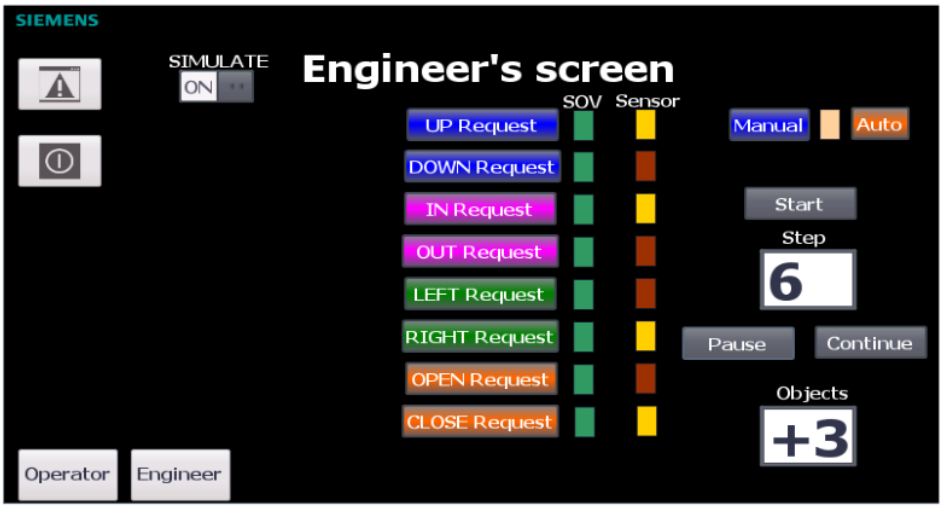



*Figure 10*

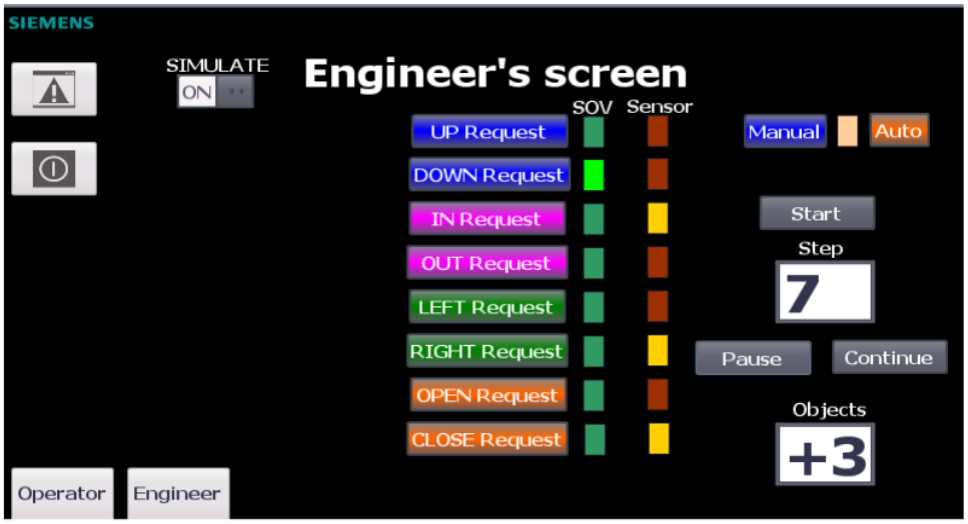



*Figure 11*

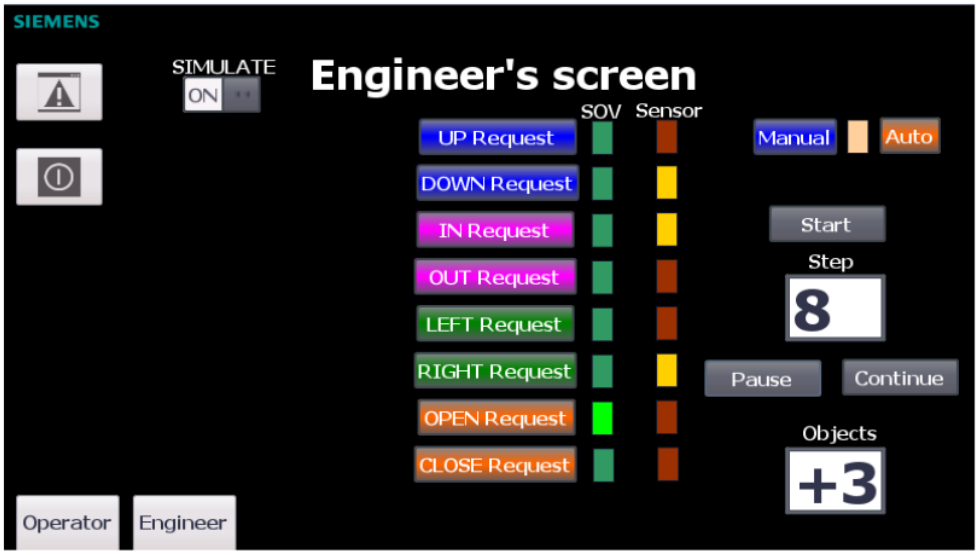



*Figure 12*

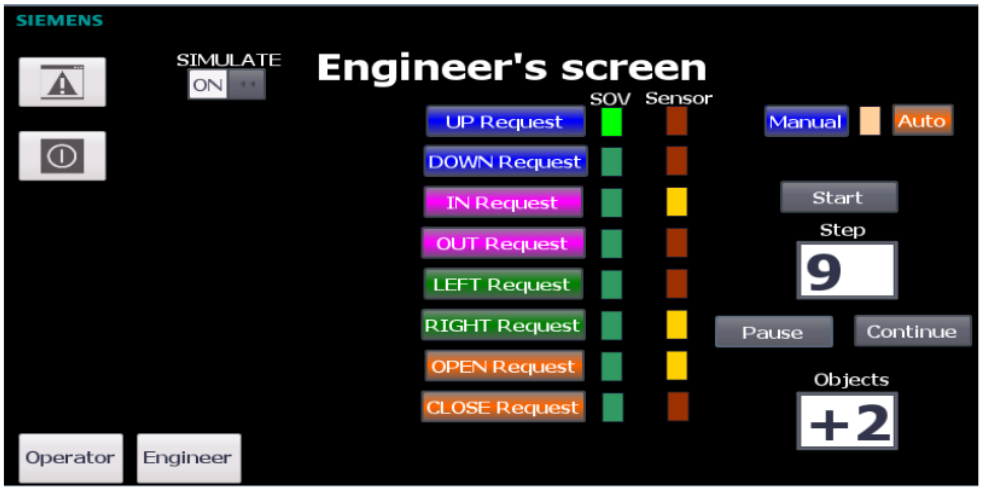



*Figure 13*

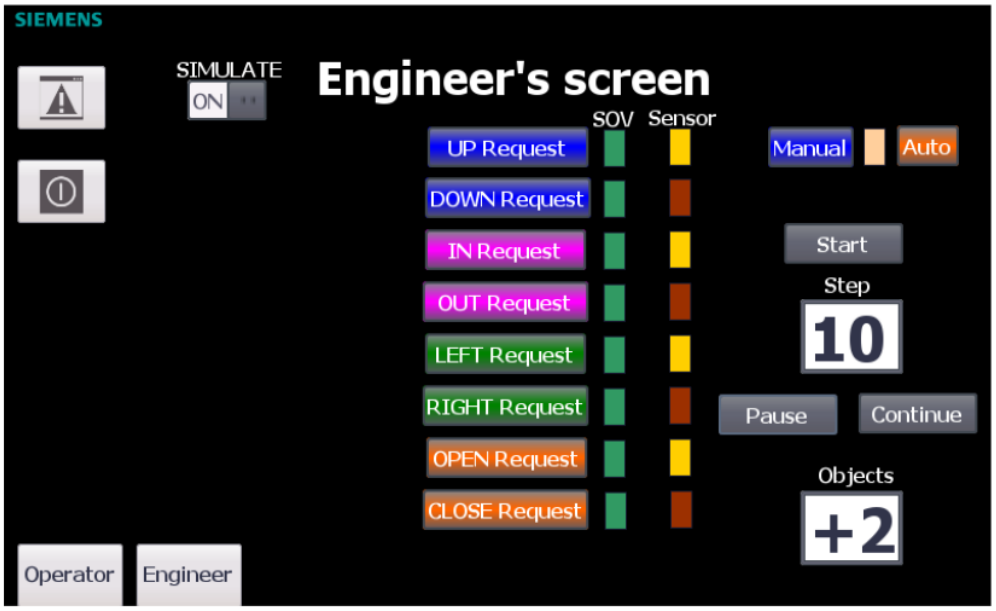



*Figure 14*

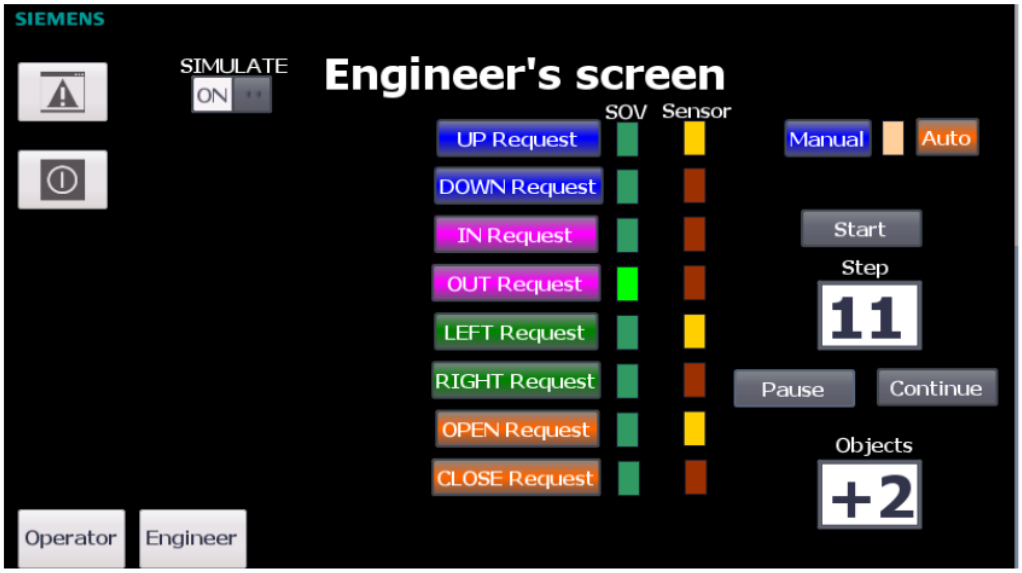



*Figure 15*

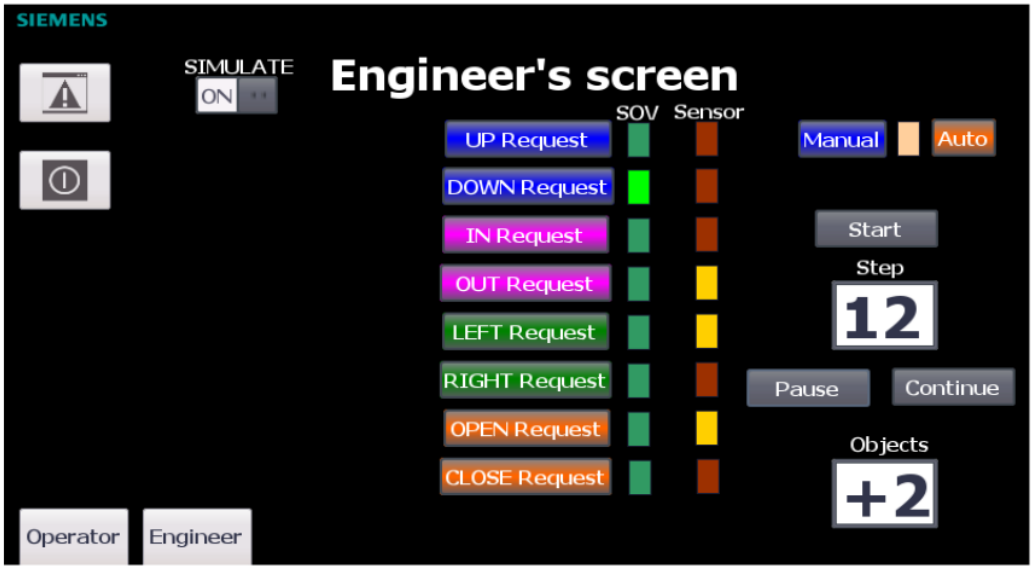



*Figure 16*

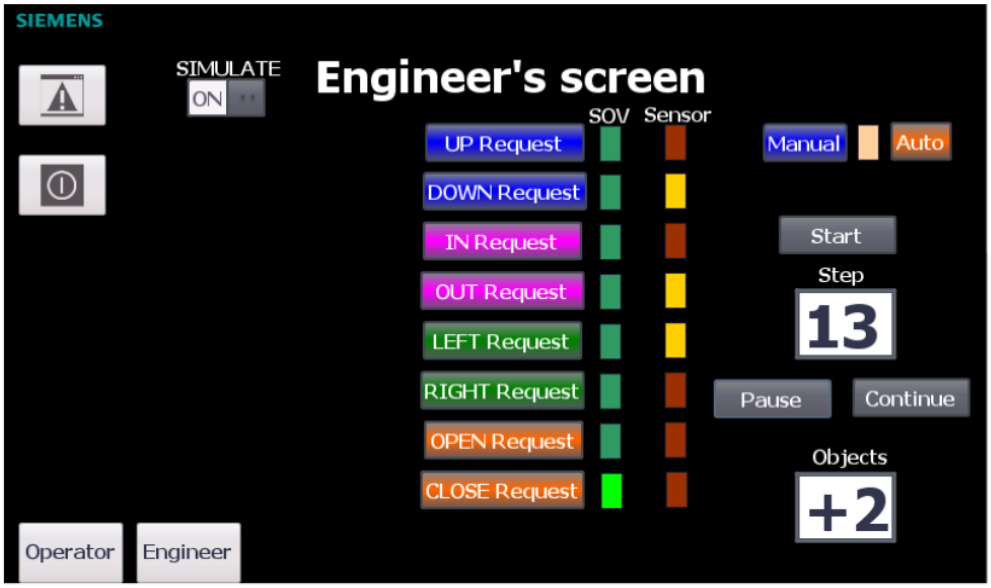



*Figure 17*

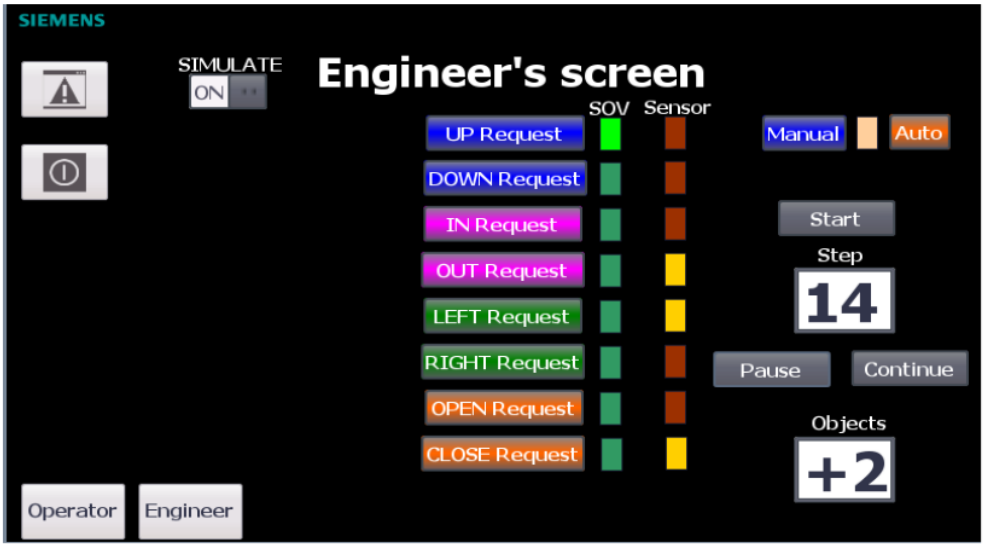



*Figure 18*

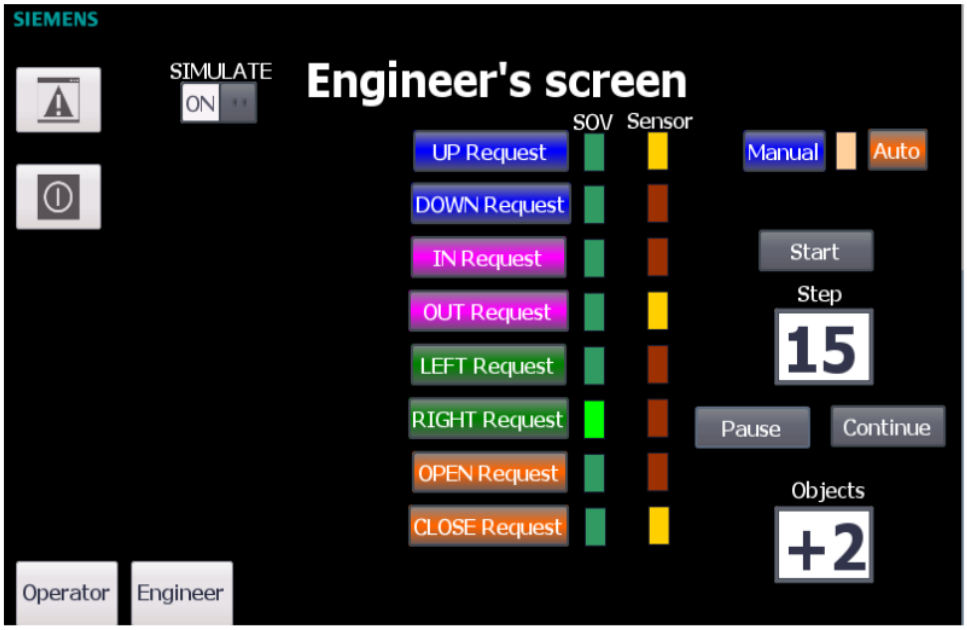



*Figure 19*

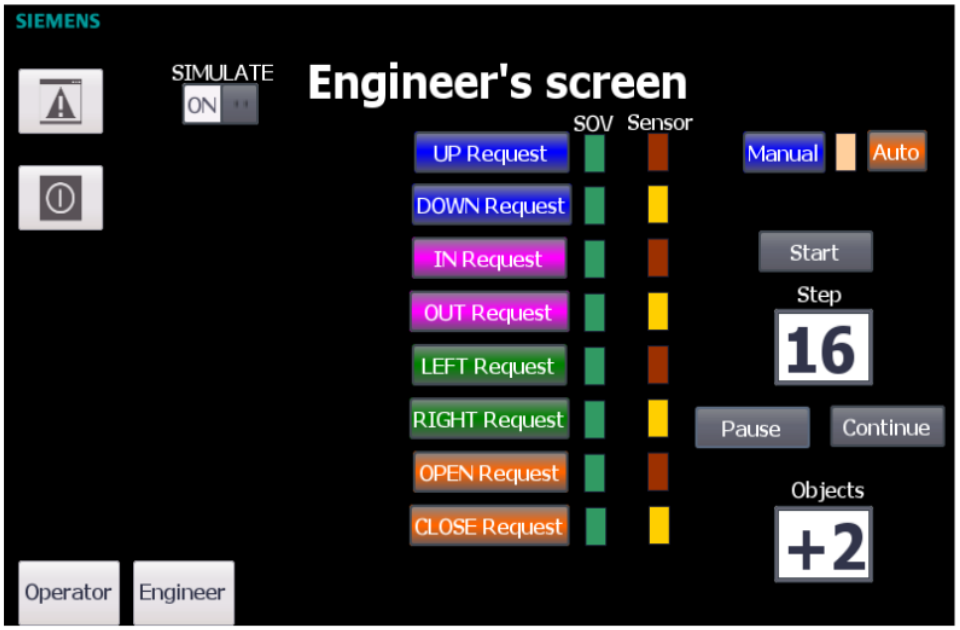



*Figure 20*

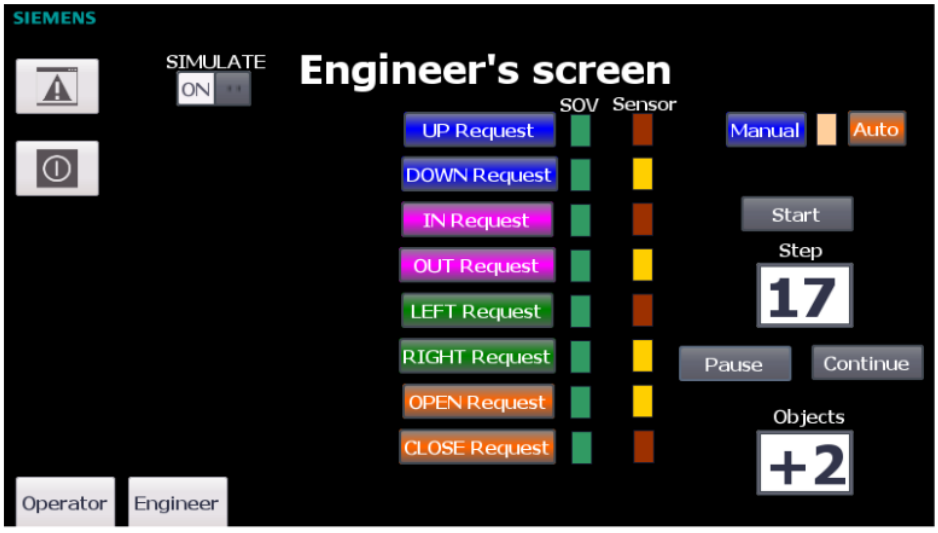



*Figure 21*

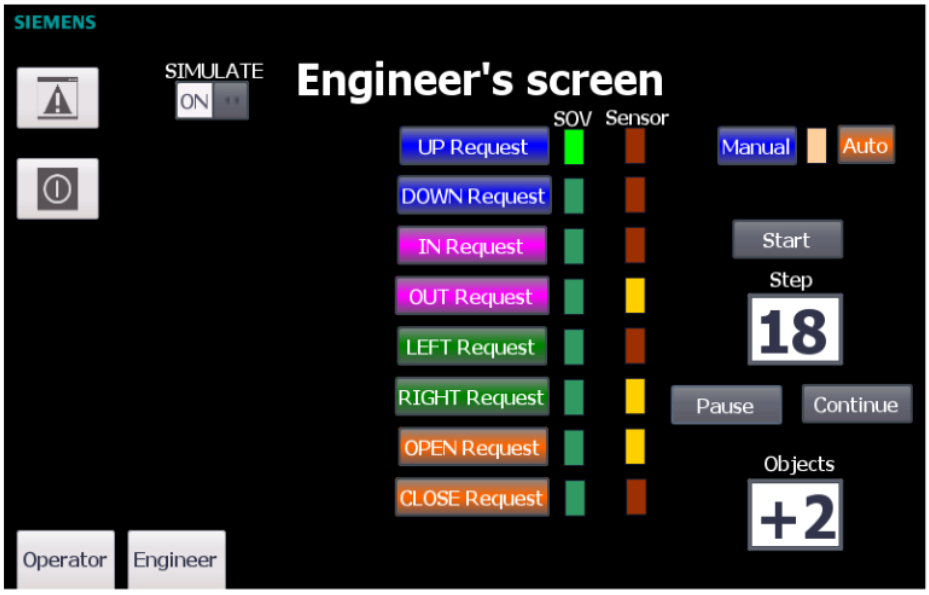



*Figure 22*

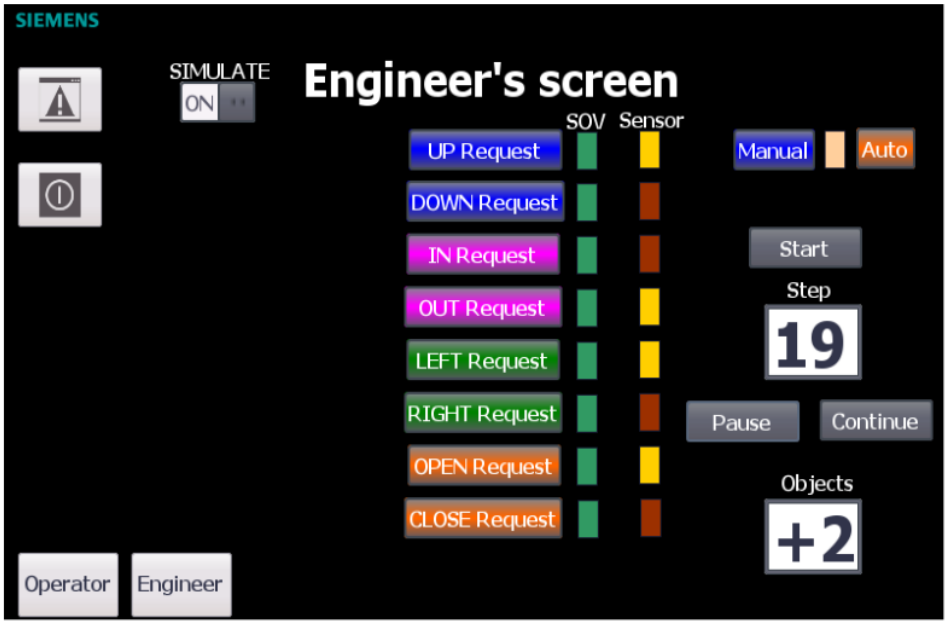



*Figure 23*

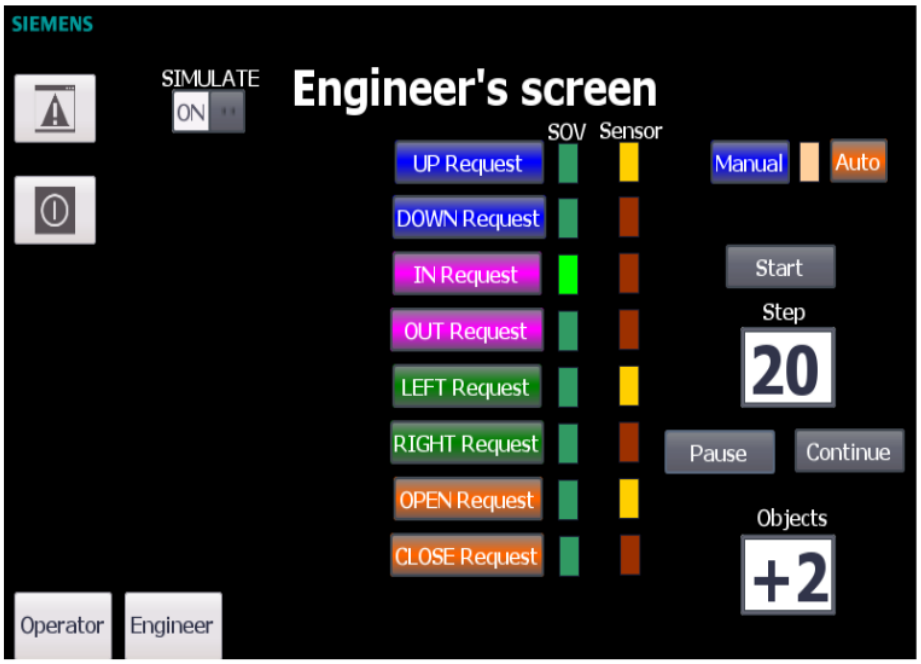



*Figure 24*

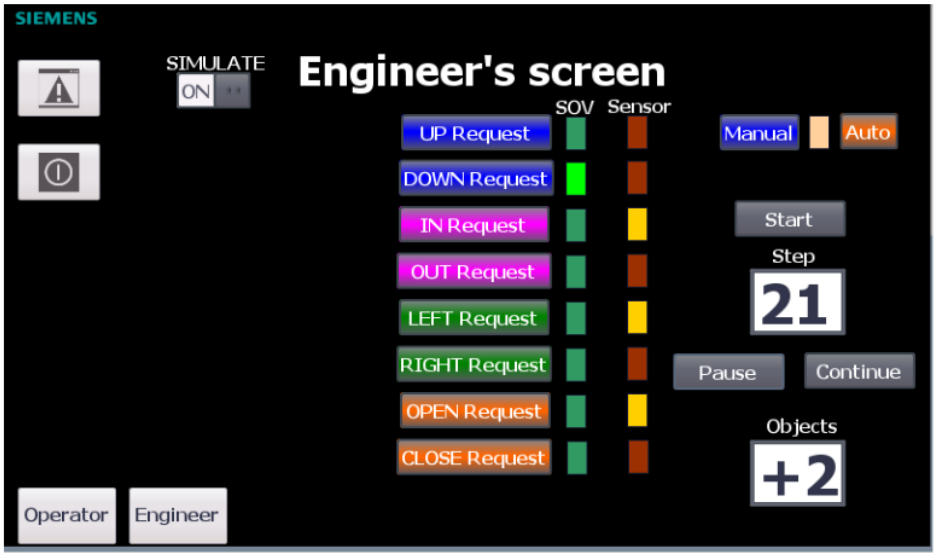



*Figure 25*

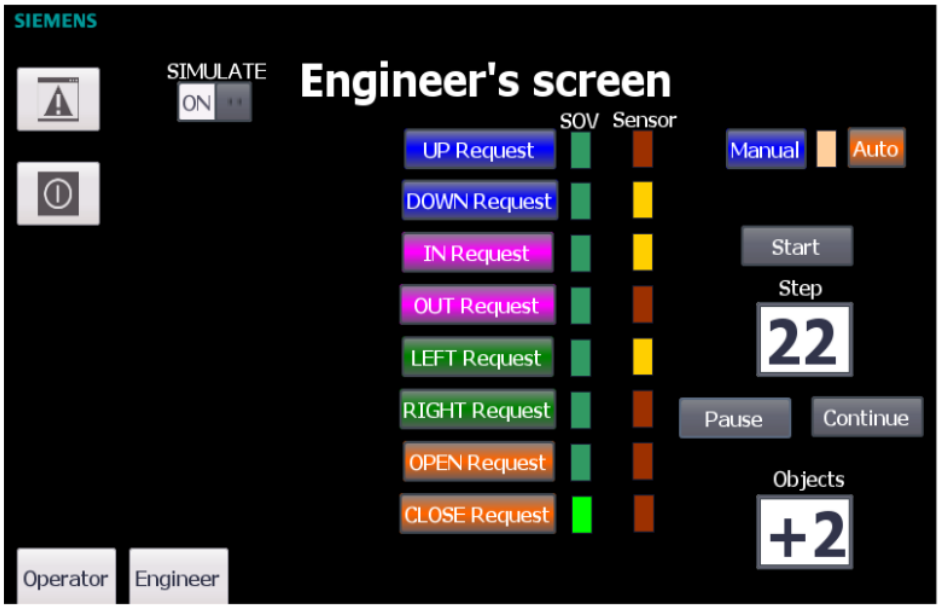



*Figure 26*

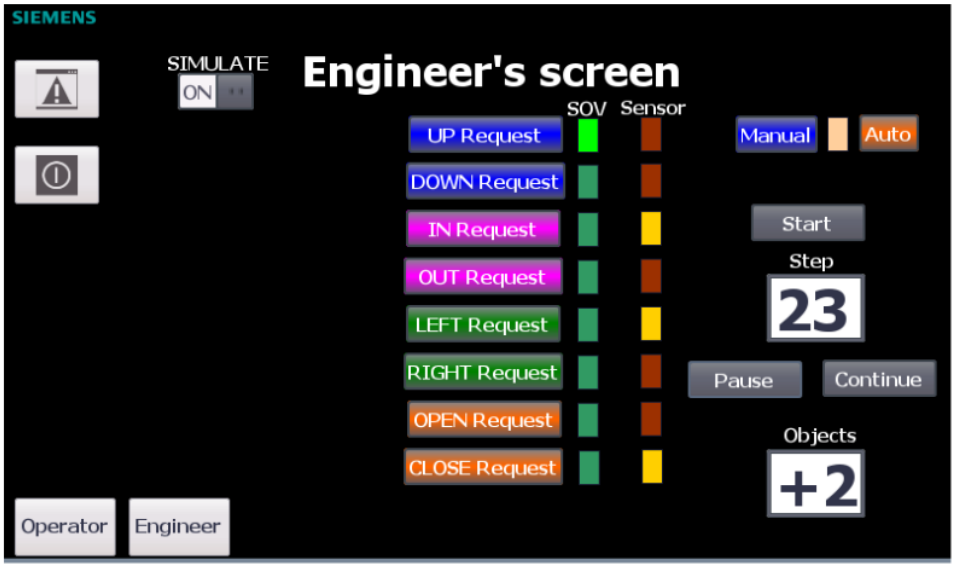



*Figure 27*

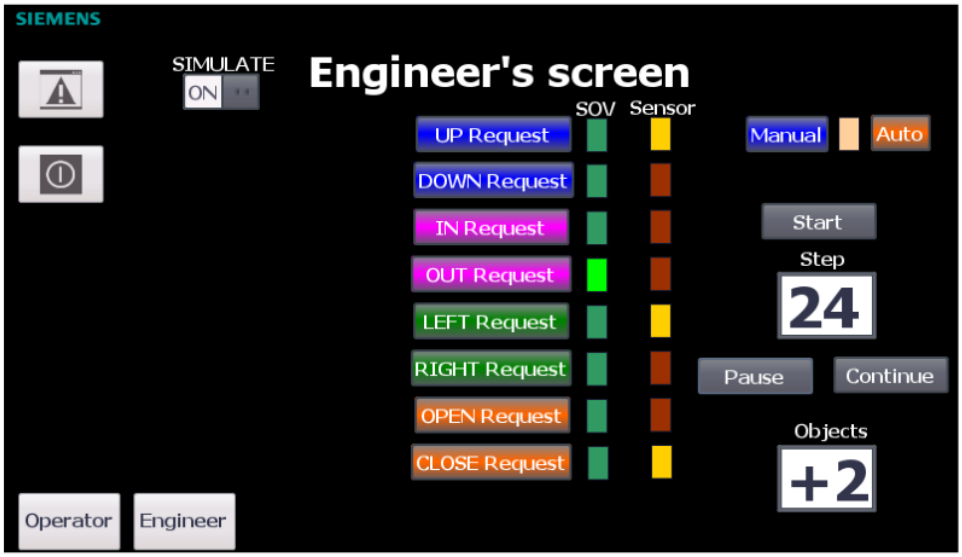



*Figure 28*

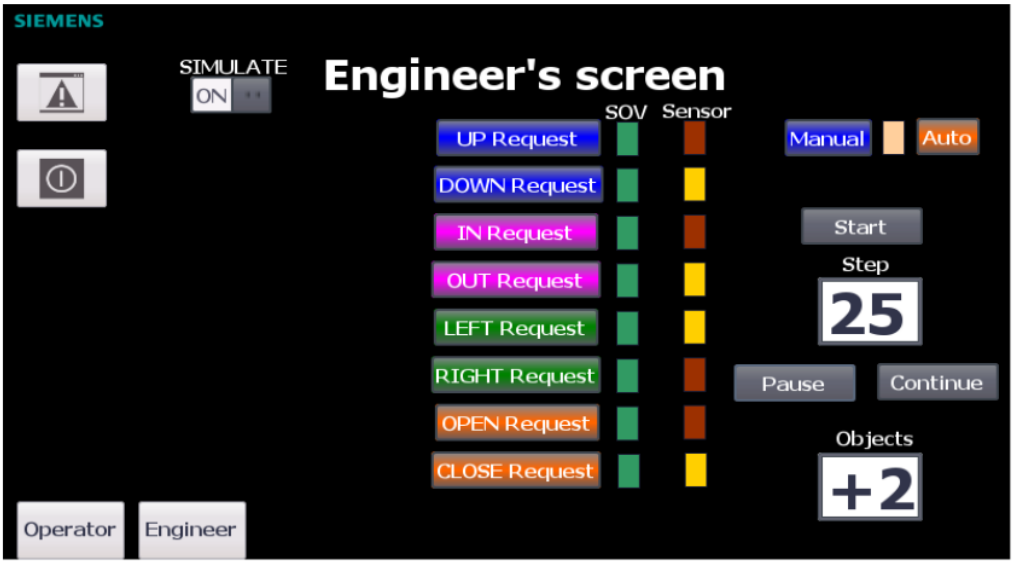



*Figure 29*

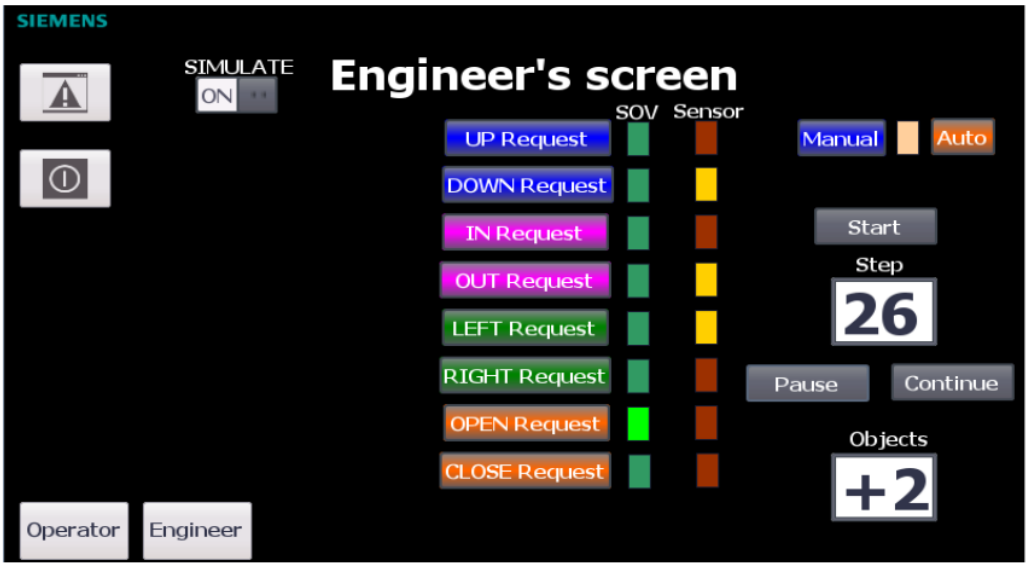



*Figure 30*

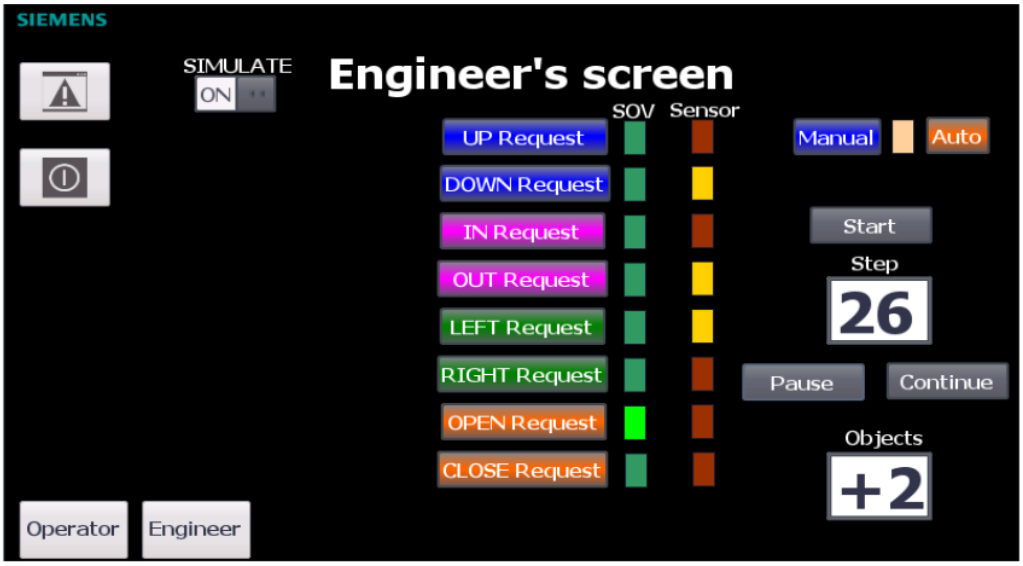



*Figure 31*

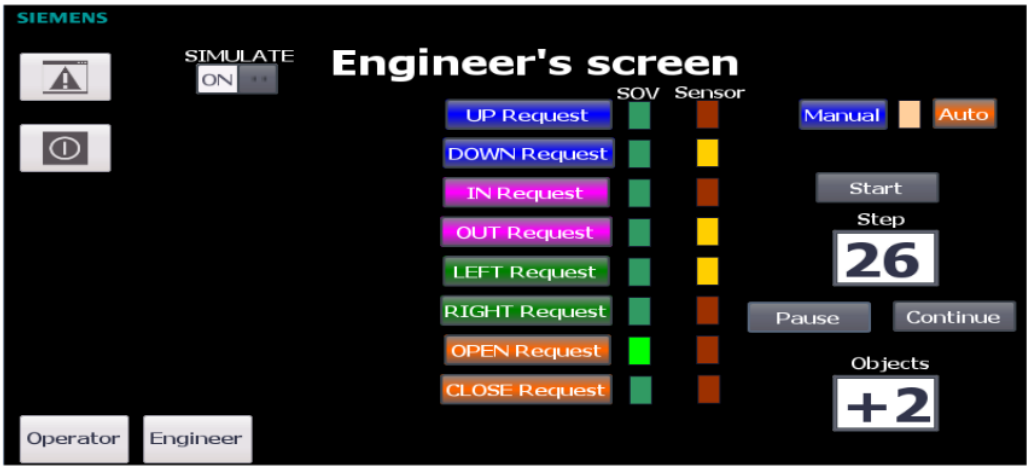



*Figure 32*

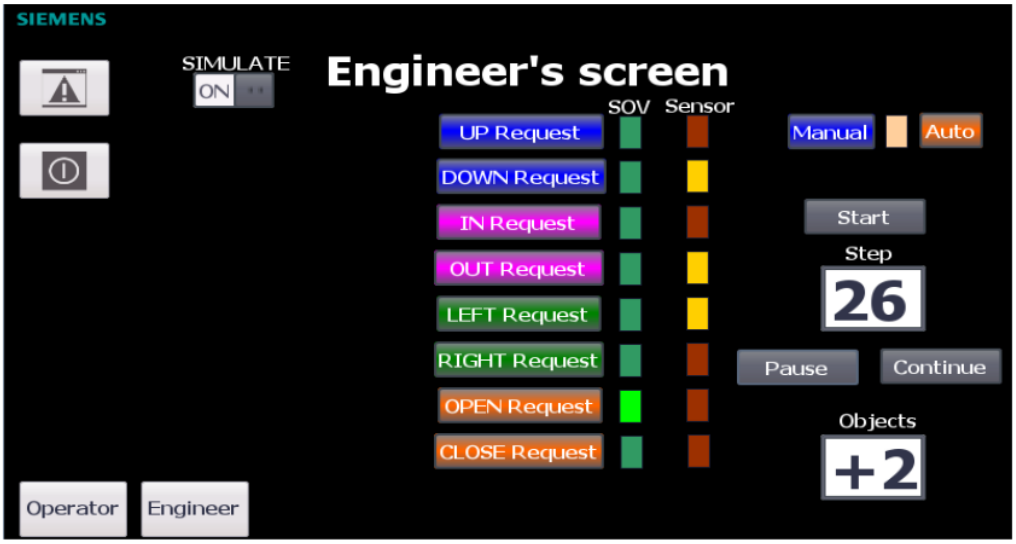



*Figure 33*

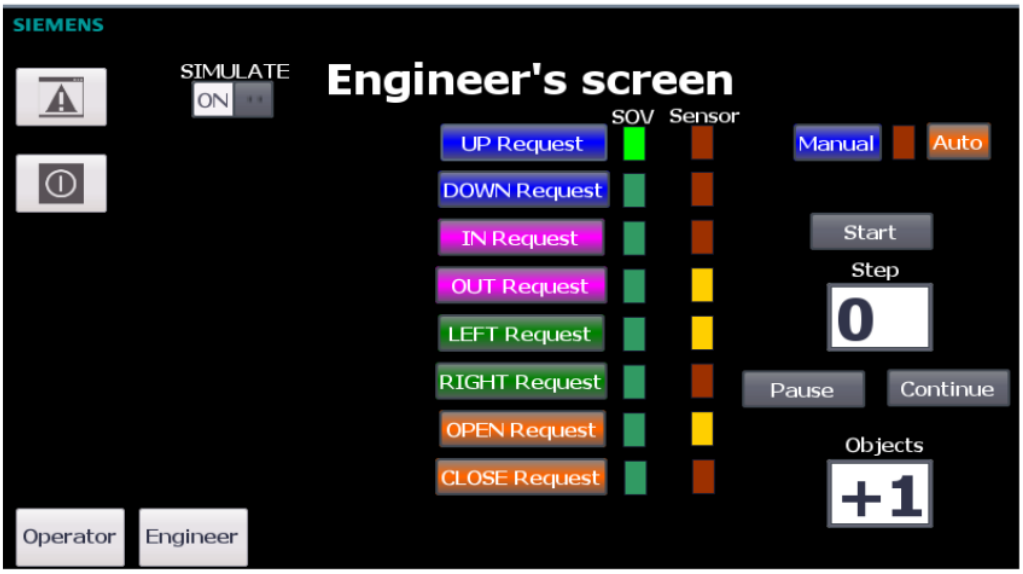



*Figure 34*

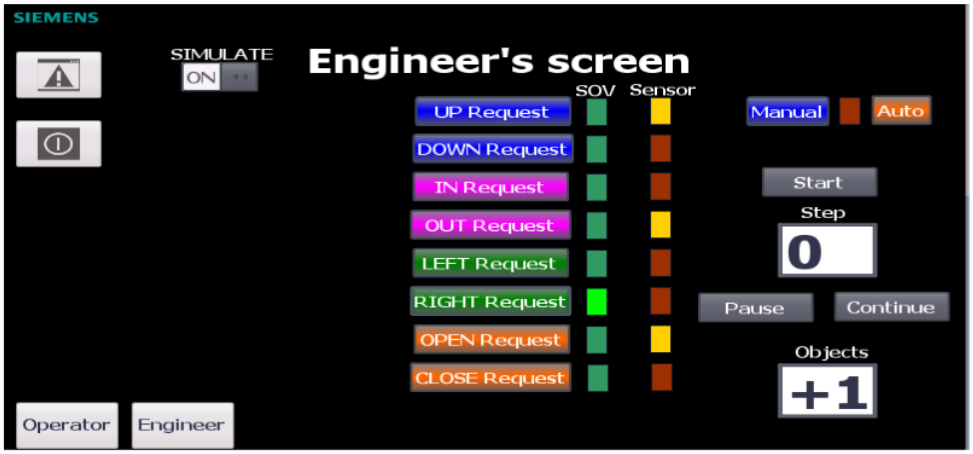



*Figure 35*

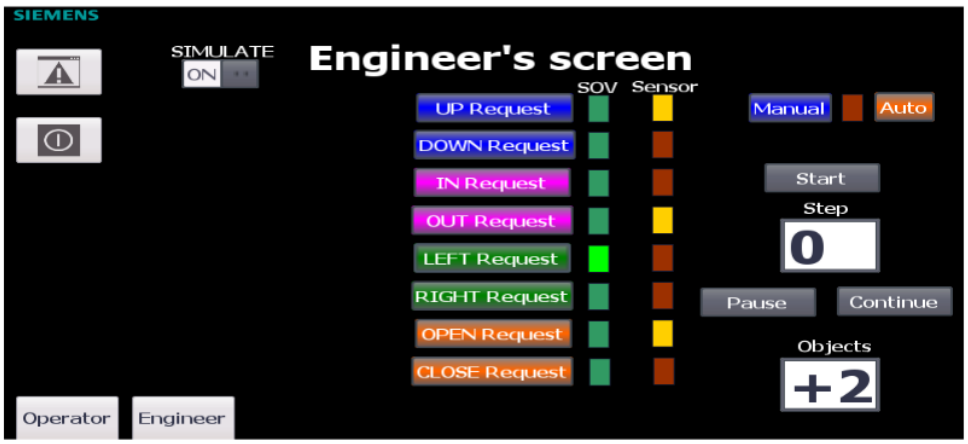



*Figure 36*

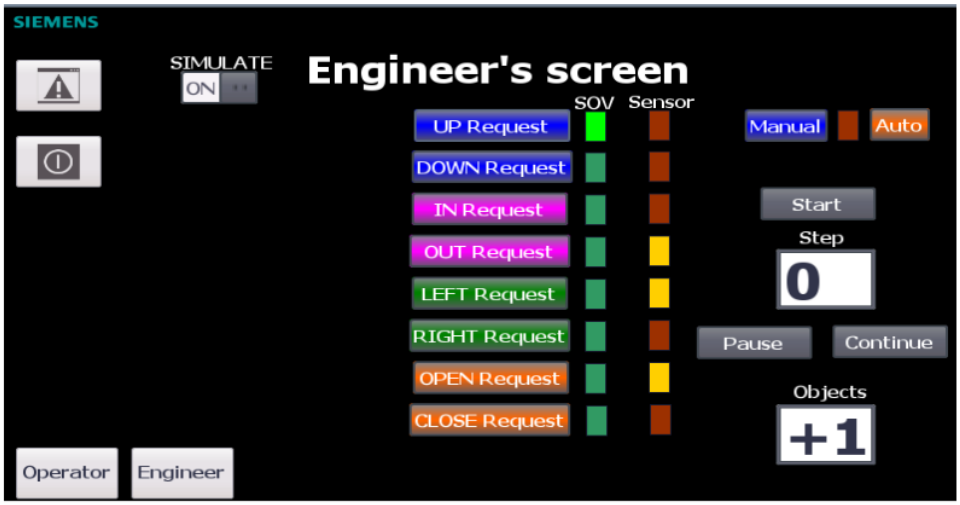



*Figure 37*

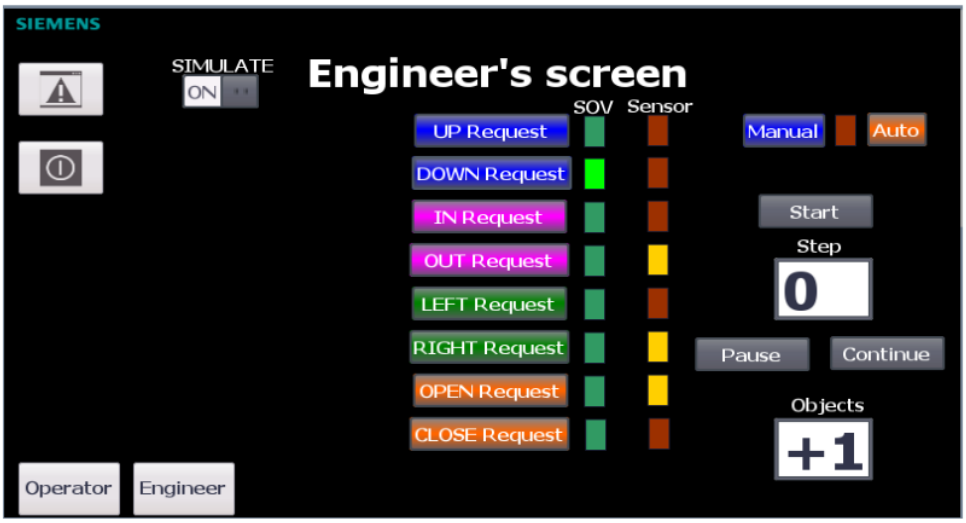



*Figure 38*

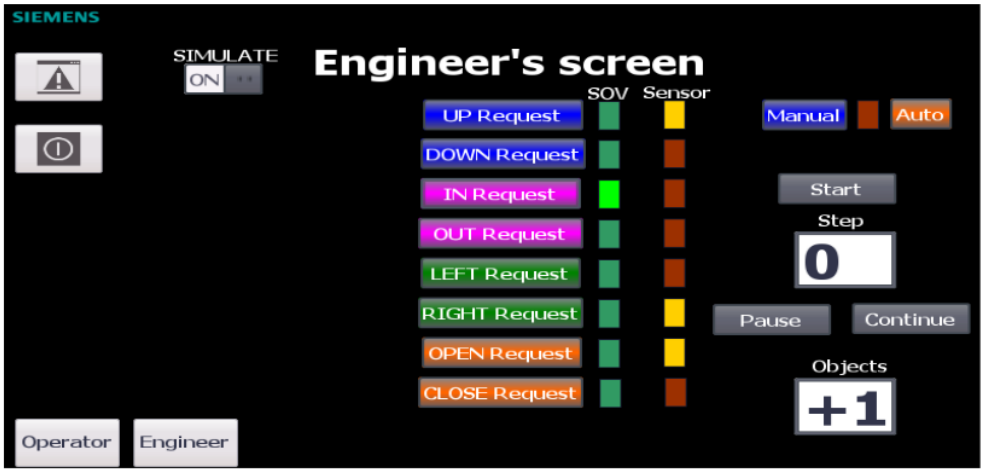



*Figure 39*

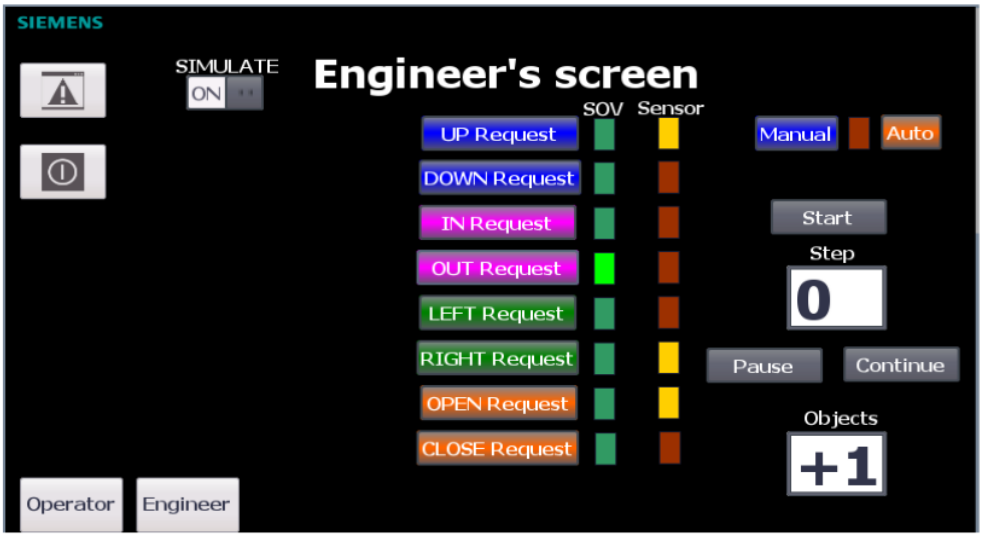



*Figure 40*

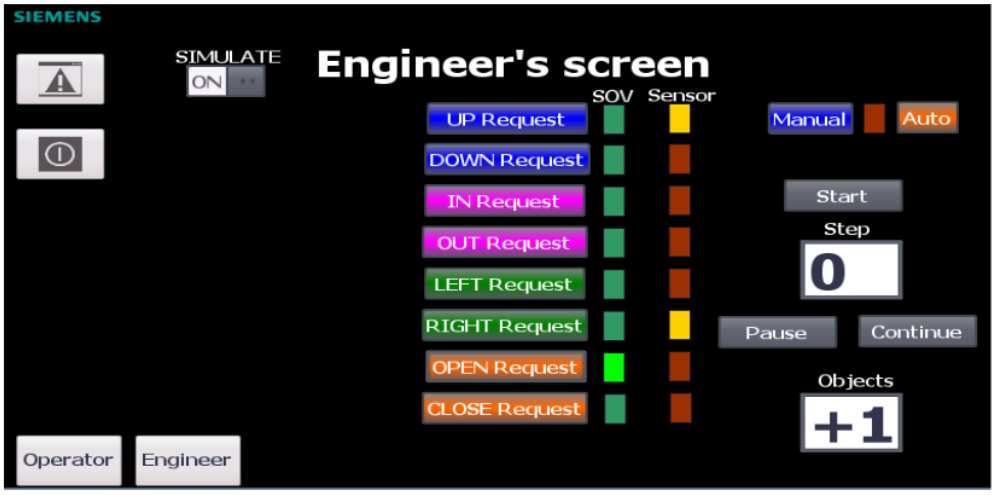



*Figure 41*

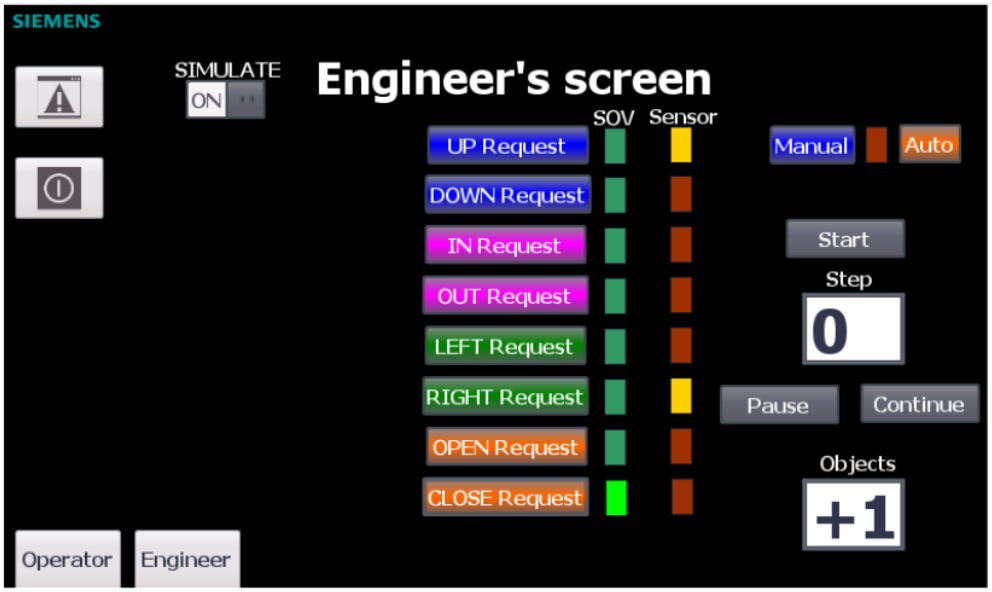



*Figure 42*

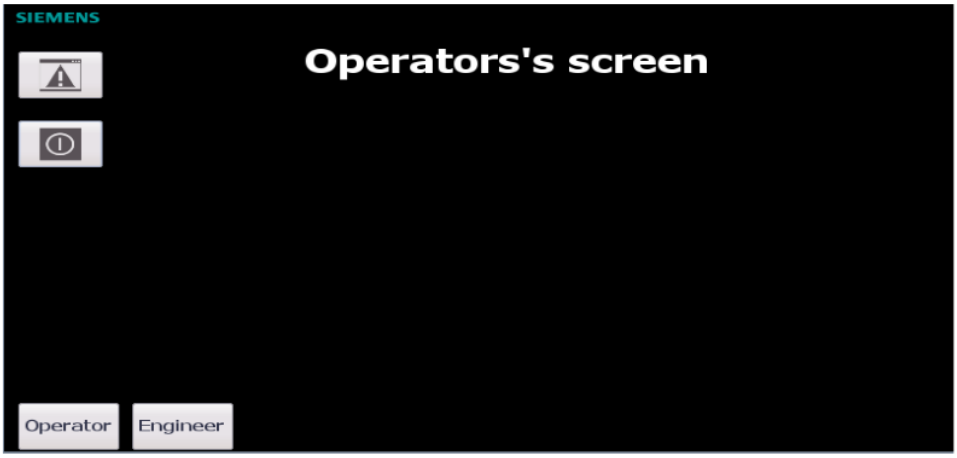



*Figure 43*

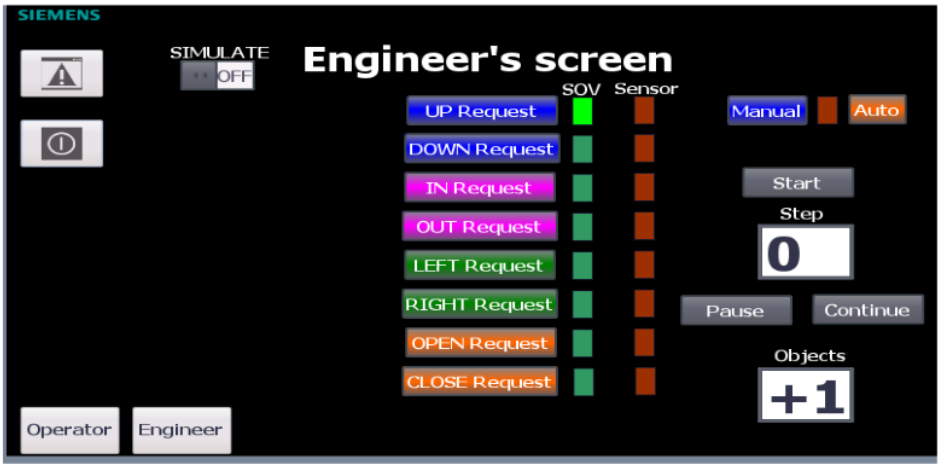



*Figure 44*

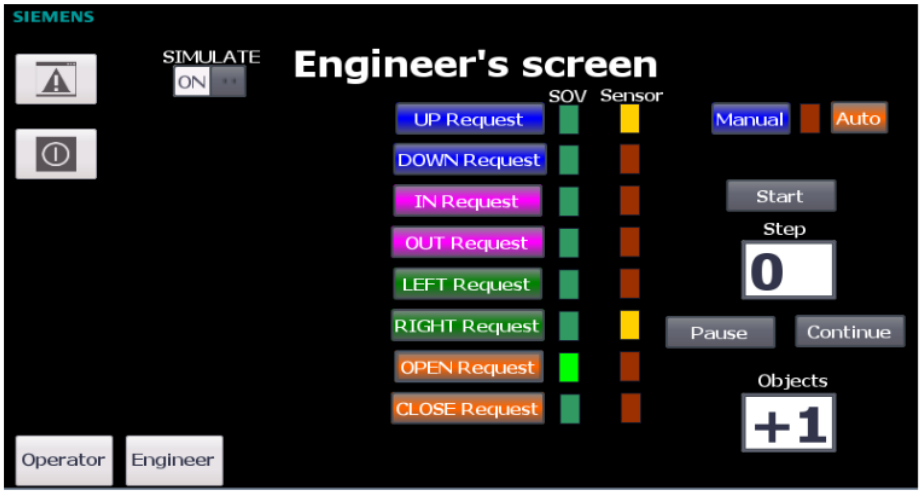



*Figure 45*

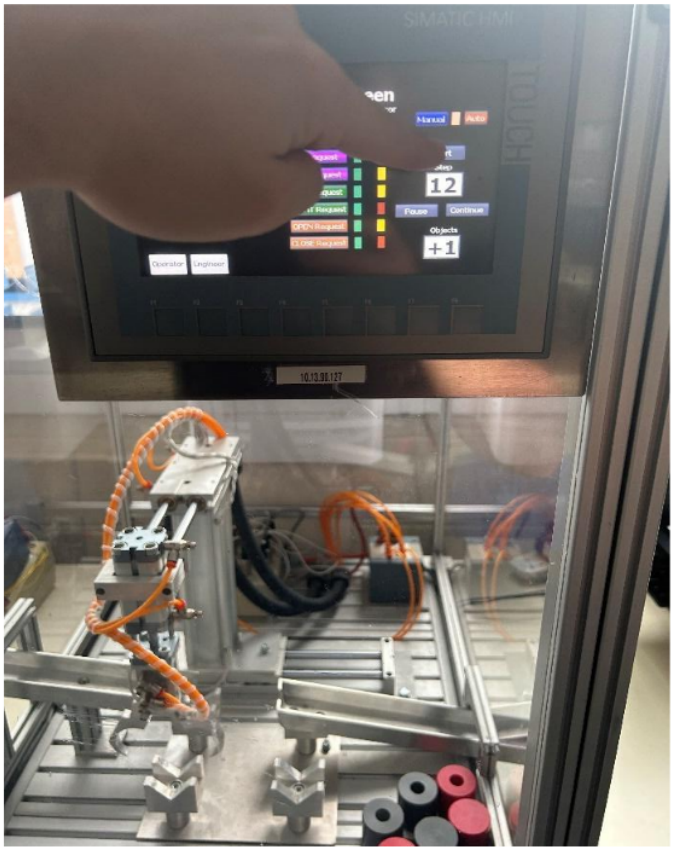


*Figure 46*




*Figure 47*




*Figure 48*




*Figure 49*




*Figure 50*




*Figure 51*

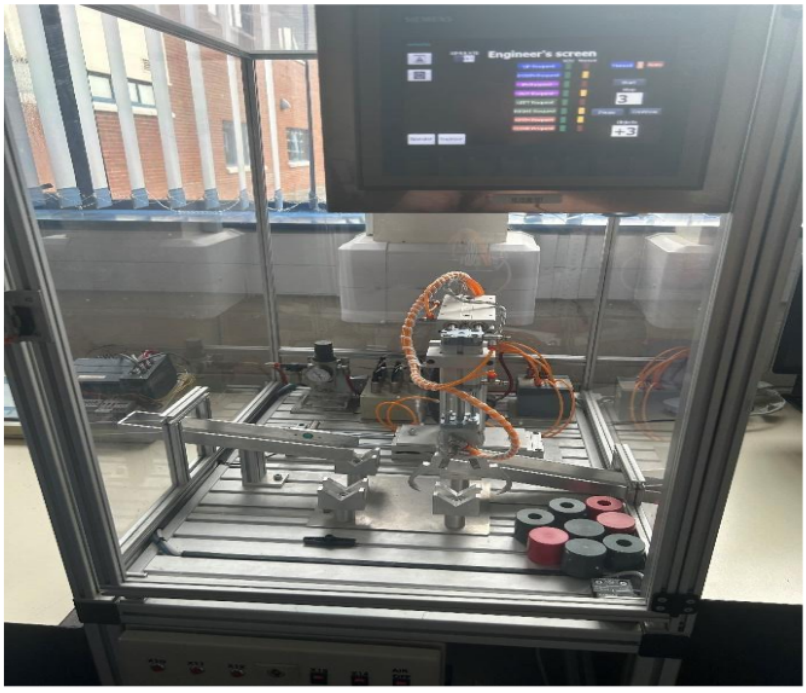



*Figure 52*

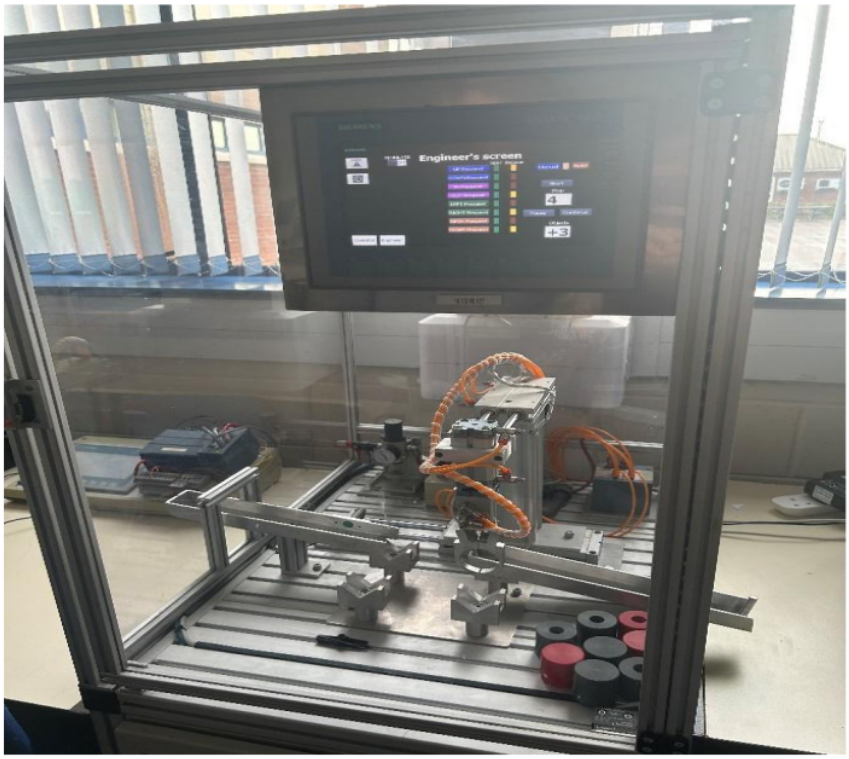



*Figure 53*




*Figure 54*




*Figure 55*




*Figure 56*




*Figure 57*




*Figure 58*




*Figure 59*




*Figure 60*




*Figure 61*




*Figure 62*




*Figure 63*




*Figure 64*




*Figure 65*

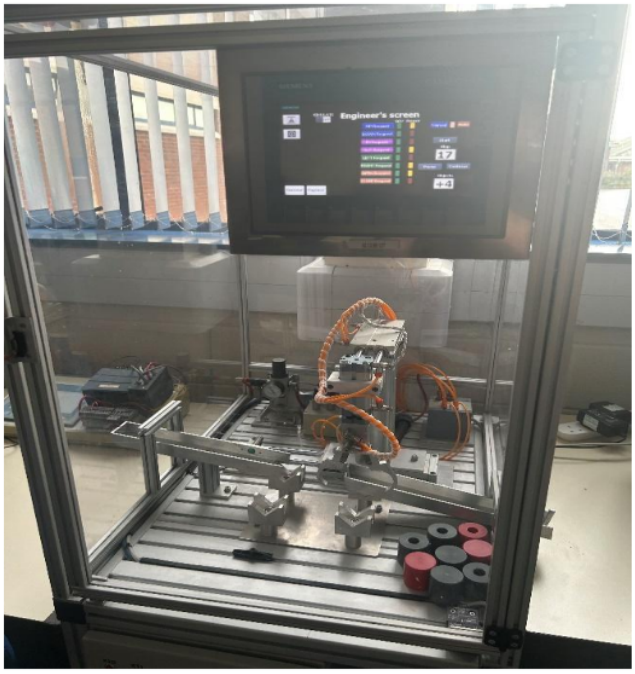



*Figure 66*




*Figure 67*




*Figure 68*




*Figure 69*




*Figure 70*




*Figure 71*




*Figure 72*




*Figure 73*




*Figure 74*




*Figure 75*




*Figure 76*




*Figure 77*




*Figure 78*



*Figure 79*




*Figure 80*



*Figure 81*

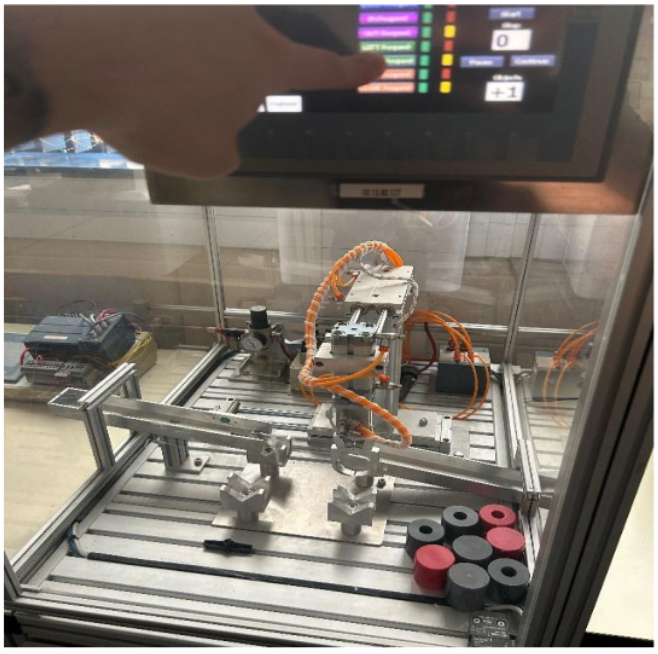


*Figure 82*

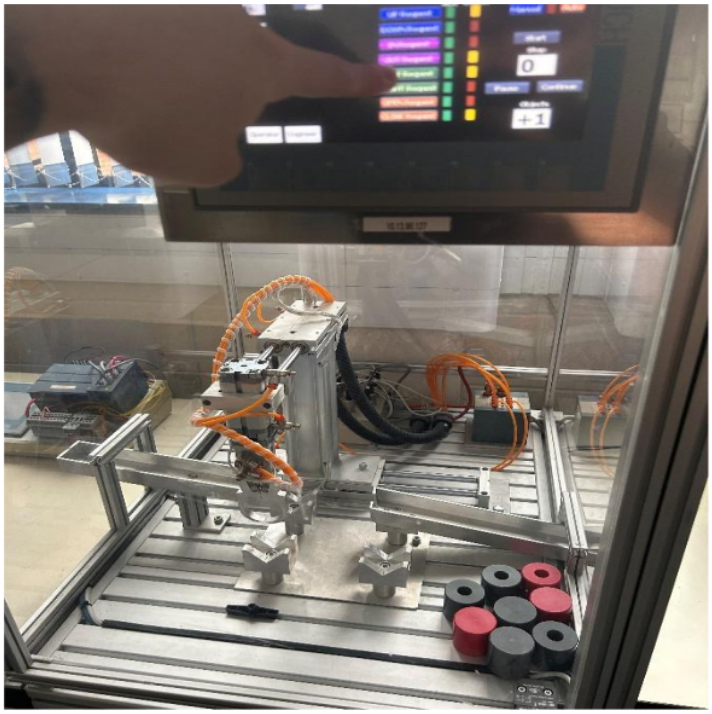


*Figure 83*



*Figure 84*




*Figure 85*

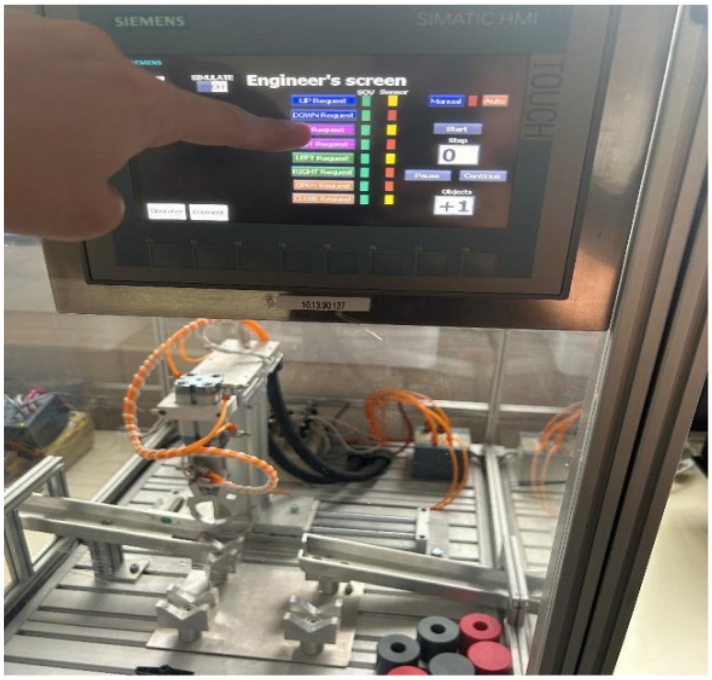



*Figure 86*

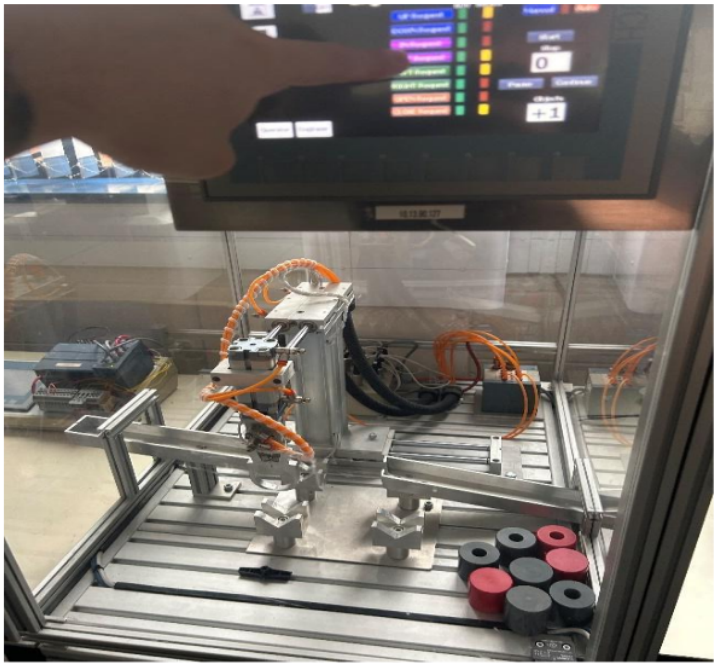


*Figure 87*




*Figure 88*

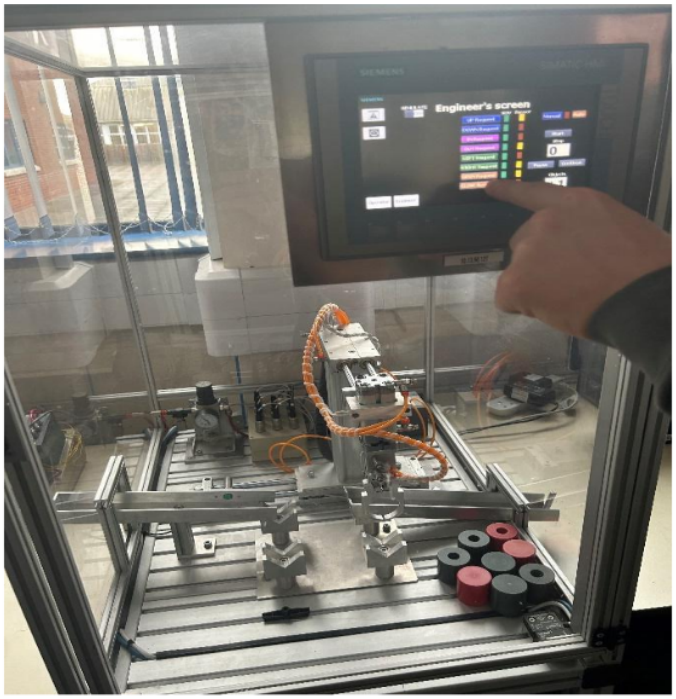



*Figure 89*




*Figure 90*




*Figure 91*

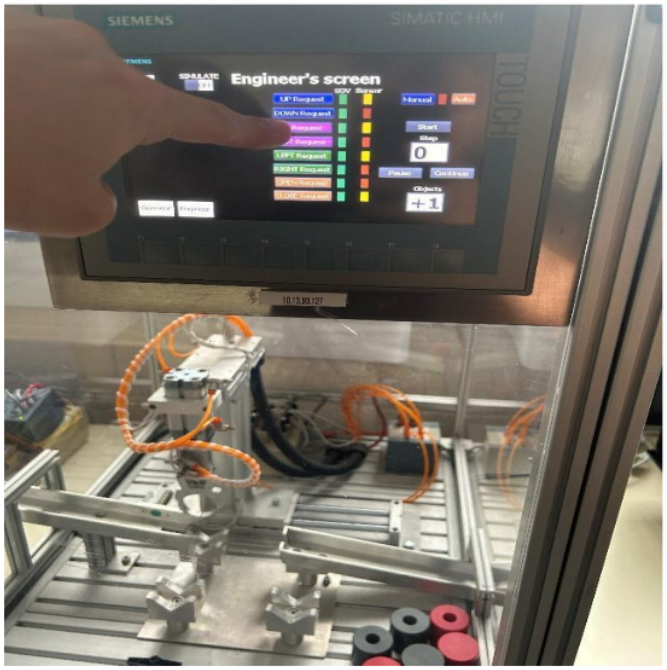



*Figure 92*

# 2221114

GRADEMARK REPORT

FINAL GRADE

60 /100

GENERAL COMMENTS

**Instructor**

PAGE 1

PAGE 2

QM

## Good Structure

Report is structured exactly as stated in the assignment brief.

PAGE 3

PAGE 4

PAGE 5

PAGE 6

**Text Comment.** sounds like a good plan

PAGE 7

QM

## Correctly cited

Citation method conforms to Harvard standard, well done.

QM

## Correctly cited

Citation method conforms to Harvard standard, well done.

QM

## Correctly cited

Citation method conforms to Harvard standard, well done.

QM

## Write in the third person

Don't refer to your self as I, me, my or others as you etc. Avoid personal references. If absolutely necessary, refer to self as "the author".

**Text Comment.** Siemens

PAGE 8

QM

## Write in the third person

Don't refer to your self as I, me, my or others as you etc. Avoid personal references. If absolutely necessary, refer to self as "the author".

**Text Comment.** From this table, it can be seen that

QM

## Good conclusions

You have shown understanding here from these observations and deductions

PAGE 9

**Text Comment.** Where is the rest of your system explanation?

**Text Comment.** Have you read the brief?

PAGE 10

**Text Comment.** Your explanations seem to be limited to program comments

PAGE 11

PAGE 12

PAGE 13

**Text Comment.** Good program comments but you should have included an explanation

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

**Text Comment.** Your code contains comments but where is your explanation?

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

QM

## Write in the third person

Don't refer to your self as I, me, my or others as you etc. Avoid personal references. If absolutely necessary, refer to self as "the author".

**Text Comment.** The results .....

**Text Comment.** a testing table was used

**Text Comment.** the

**Text Comment.** three different PLCs were researched

QM

## Write in the third person

Don't refer to your self as I, me, my or others as you etc. Avoid personal references. If absolutely necessary, refer to self as "the author".

**Text Comment.** this is a good bug fix

**Text Comment.** You have learnt a lot from this assignment

PAGE 29

PAGE 30

**Text Comment.** a range of source correctly listed

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

PAGE 50

PAGE 51

PAGE 52

PAGE 53

PAGE 54

PAGE 55

PAGE 56

PAGE 57

PAGE 58

PAGE 59

PAGE 60

PAGE 61

PAGE 62

PAGE 63

PAGE 64

PAGE 65

PAGE 66

PAGE 67

PAGE 68

PAGE 69

PAGE 70

PAGE 71

PAGE 72

PAGE 73

PAGE 74

PAGE 75

PAGE 76

GRADING FORM: NEW CSG-RUBRIC

KAI WHITE

**60**

**1. ENGAGEMENT WITH LITERATURE**

A good research section with evidence of reading from a range of sources cited and listed correctly using the Harvard method. 12/20

**12**

**2. KNOWLEDGE AND UNDERSTANDING**

Good knowledge and understanding demonstrated in your research findings and IO list. However, your hardware investigation is missing and your program design explanations are limited to comments inside the program. 11/20

**11**

**3. COGNITIVE AND INTELLECTUAL**

Good simulation and practical test results analysis, discussion and conclusions/recommendations with bug fix explained. 13/20

**13**

**4. PRACTICAL APPLICATION**

Very good program design. Good simulation and practical testing methods and presentation of results. 14/20

**14**

**5. TRANSFERABLE SKILLS**



A good work plan and explanation of team working. Report is well structured and presented with clear diagrams. Word count not exceeded. However, you must learn to write in the third person. The report is not meant to be a personal reflection. Avoid use of I, me, my, you, your, we, our etc 10/20

**10**