

PSDE4005

by Phillip Rowe

FILE	7557_PHILLIP_ROWES_PSDE4005_156961_661378007.DOCX		
TIME SUBMITTED	11-MAR-2018 10:22PM (UTC+0000)	WORD COUNT	2761
SUBMISSION ID	84474248	CHARACTER COUNT	13919

Phillip Rowe
Electrical and electronic engineering
PSDE 4005

Programmable logic controllers



Prifysgol Cymru

Y Drindod Dewi Sant

University of Wales

Trinity Saint David

Table of Contents

1. Introduction	3
2. Hardware Investigation, System Design and I/O List	4
Hardware	4
3. HMI's	8
4. Sequencer	14
5. Testing	23
6. Conclusion and evaluation	27

1. Introduction

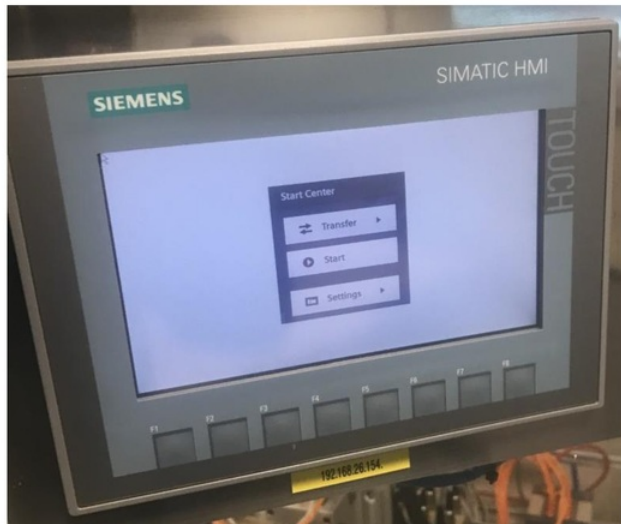
avoid personal references, write in the third person
The main purpose of this assignment is to further the understanding of PLC system designs around industrial process which in this assignment involves the pick and place robot where the key factor is sequencing. During this assignment knowledge will be demonstrated about the IEC programming languages available.

2. Hardware Investigation, System Design and I/O List

Hardware

In this section all the components that make up the pick and place robot will be explained with photographs to aid the description.

As shown below in figure 1 is the HMI(human machine interface) which is being used in order to manually test the pick and place robot. This screen allows an operator to force actions without running an auto sequence by using interactive buttons and ladder rungs in order to perform actions.



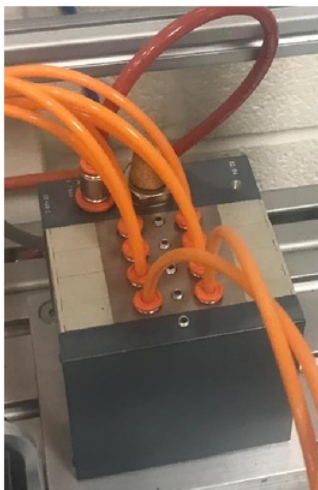
(Figure 1)

The second bit of hardware which can be seen in figure 2 is the reed switch. These switches are used in order to detect which position the claw is in and also which location the claw is in. The reed switch works by using a magnet to pull normally open switches closed which completes the circuit indicating the actuator is at that position.



(Figure 2)

The third item of hardware is the memory valve stand which can be seen in figure 3. This is responsible for memorising the last action sent to the valve and it will hold it until it is told otherwise. This is sort of a safety interlock which can prevent injury and also damage to the machine.



(Figure 3)

Phillip Rowe PSDE 4005

*Incorrect, figure 4 actually
shows the reed switch
junction box*

Following below in figure 4 is the valve stand where the electrical connections are connected to the valves in order to supply the signal to the valve to tell it to either open or close. This is done by using outputs from the PLC. Also in figure 5 is an inductive sensor which role is to sense is there is a plastic block in front of it, when it senses something it will close the contact sending it back to the PLC.



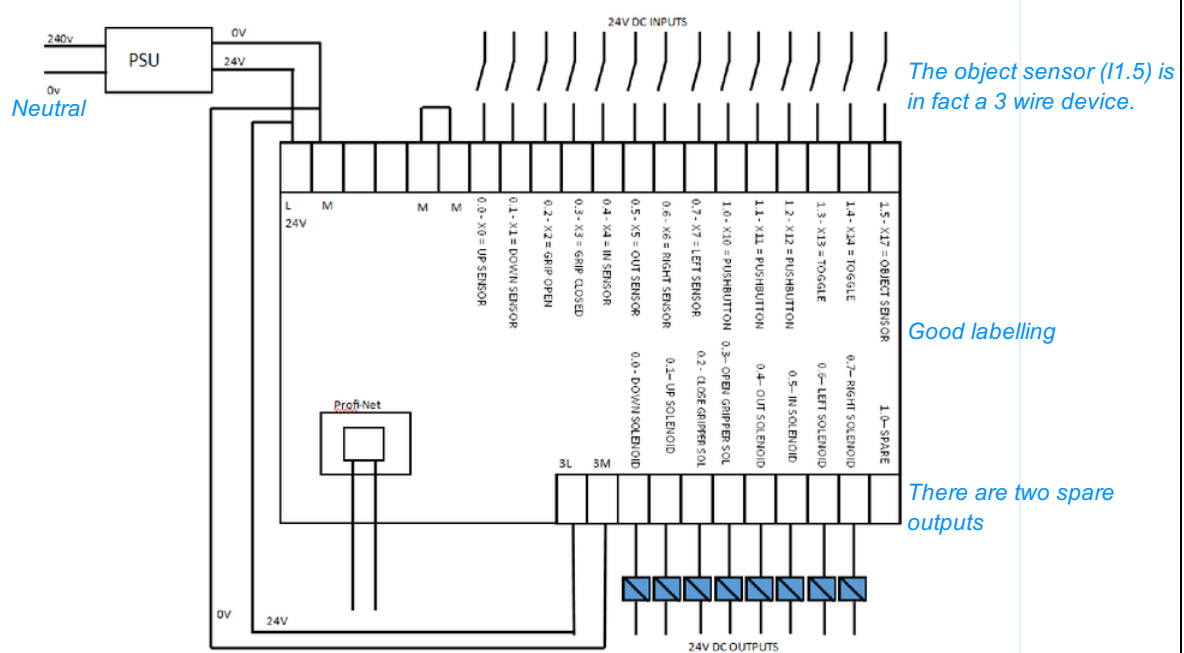
(Figure 4)



(Figure 5)

*Incor
rect,
it
woul
d
have
to be
a
capa
citive
sens
or to
dete
ct
plast
ic*

Following below in figure 6 is a drawing which has been created showing all the connections and their addresses to the PLC.



(Figure 6)

The other side of each coil must be grounded otherwise they will not energize

As shown below in figure 7 is the I/O table use for the physical inputs and outputs.

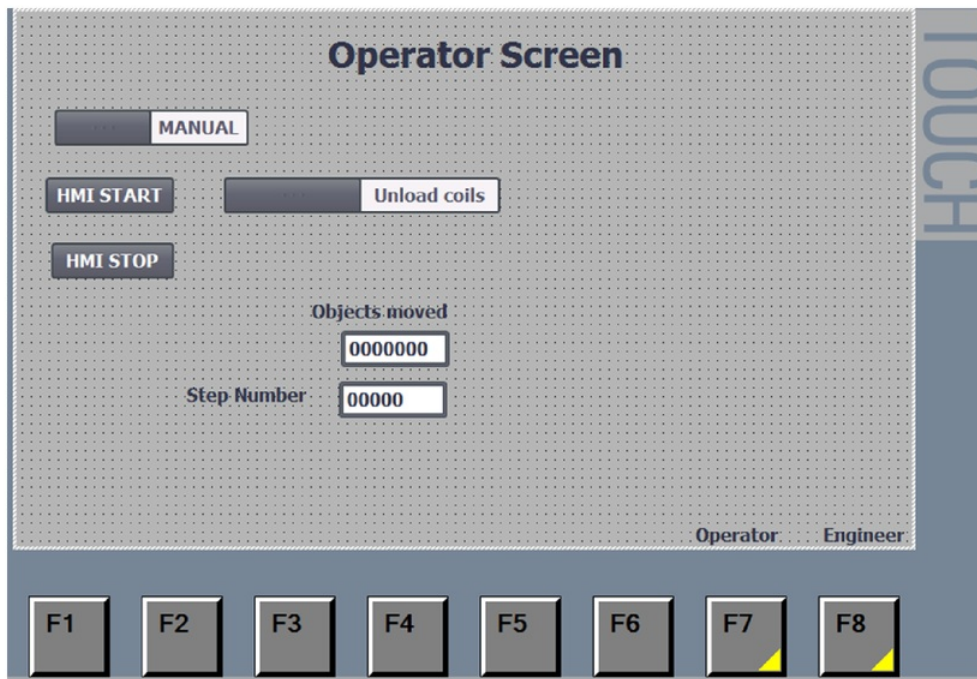
*A
useful
table*

PLC Address	Type of input/output
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 X10
I1.1	Pushbutton X11
I1.2	Pushbutton X12
I1.3	Toggle switch X13
I1.4	Toggle switch X14
I1.5	Object sensor
Q0.0	Move down solenoid
Q0.1	Move up solenoid
Q0.2	Close gripper solenoid
Q0.3	Open gripper solenoid
Q0.4	Move out solenoid
Q0.5	Move in solenoid
Q0.6	Move left solenoid
Q0.7	Move right solenoid

(Figure 7)

3. HMI's

For this assignment two HMI screens were created. One of these screens was for the operator and the other was an engineering screen. As seen below in figure 8 is the operators screen which was created.

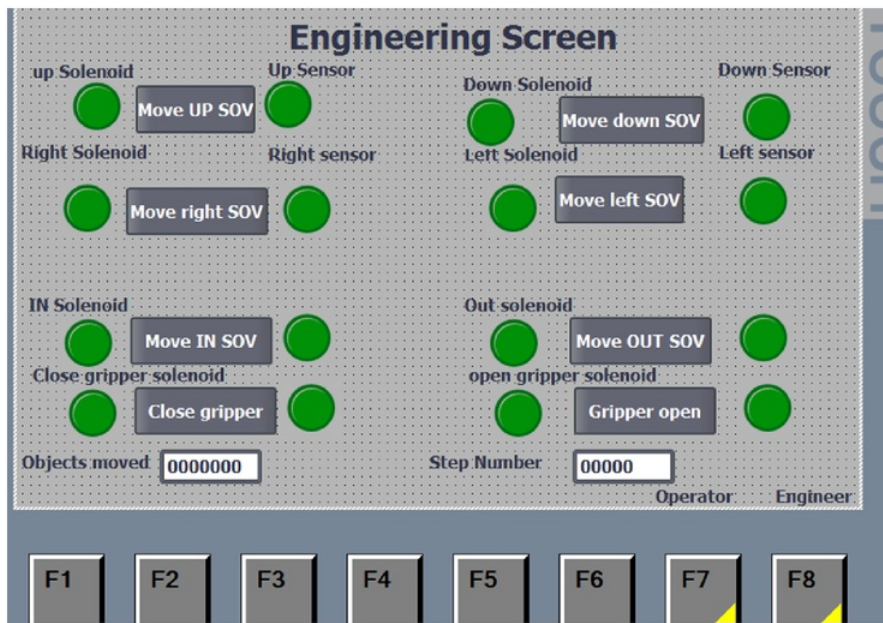


(Figure 8)

As seen above in figure 8 the screen has a start and a stop function, an option for loading and unloading coils and also a manual or auto selector switch. There are also counters in order to display the step number and also the amount of objects moved.

Following below in figure 9 is the engineering screen.

*The system is not
handling coils.*



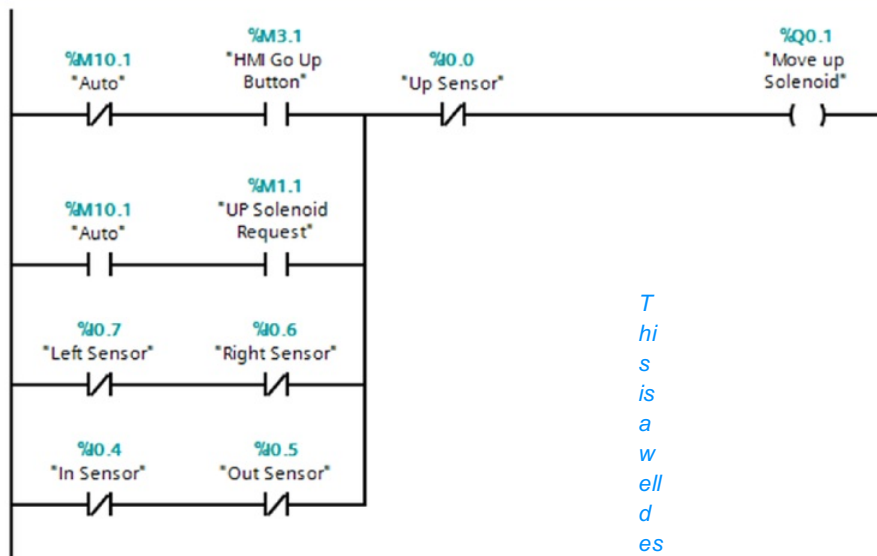
(Figure 9)

As seen above in figure 9 is the engineering screen. This screen allows engineers to operate solenoids manually (forcing outputs). Also there are two sets of indication for each solenoid which displays when the sensor is made and also when the solenoid is energised. The step number and objects moved are also on this screen.

Following below is the manual control for the engineering screen. This can be seen in figure 10 below.

Network 1: Interlocks to move arm UP

Comment



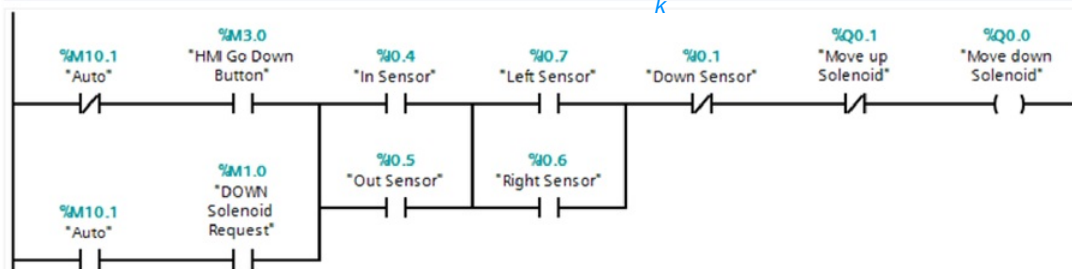
(Figure 10)

As shown above in figure 10 is the manual sequence for moving the solenoid arm up. There is also the auto sequence way for the solenoid arm to move up.

As shown below in figure 11 this is the move down solenoid. The arm can either move down in auto or manual. The arm has a safety interlock which means that if the claw is trying to move up it cannot move down at the same time. The gripper can also only move down if either the in or out sensor is made and also the left or right sensor is made. This stops the claw moving if it is already trying to perform another function.

Network 2: Interlocks to move arm Down

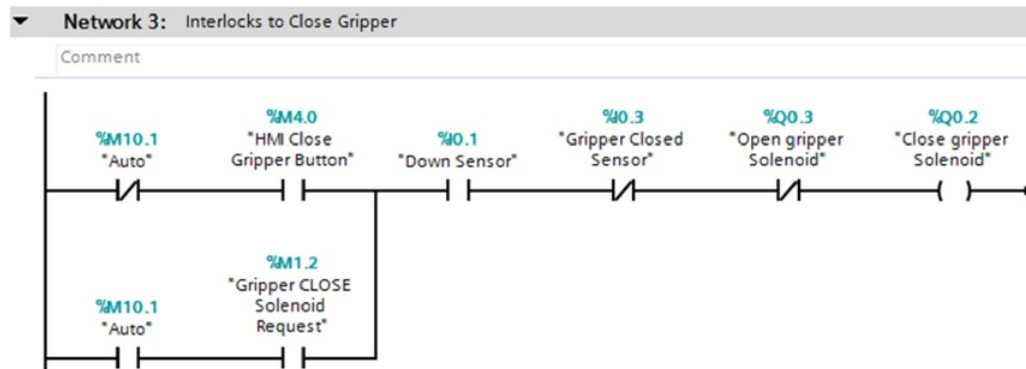
Comment



(Figure 11)

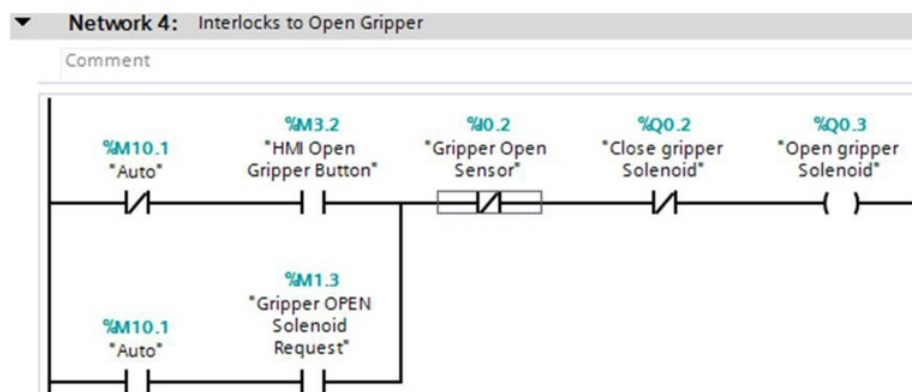
Good design
feature

Below in figure 12 is the way in which the gripper closes. It can either operate manually or automatically. There are safety interlocks such as the gripper cannot try to close if it is already trying to open, the gripper must also be open and also the claw must be down in order to prevent dropping of the coil in mid-air.



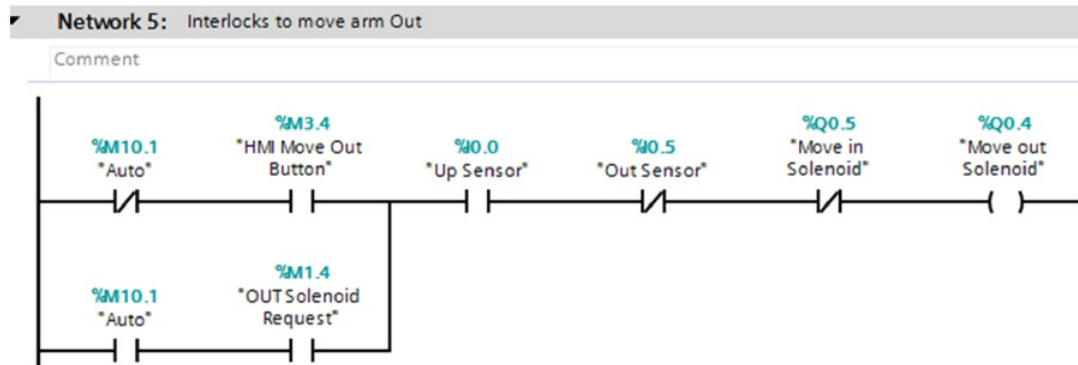
(Figure 12)

Below in figure 13 is the open gripper network. It can either operate manually or automatically. The gripper must be closed and also the gripper must not be trying to close before the gripper can try to open.



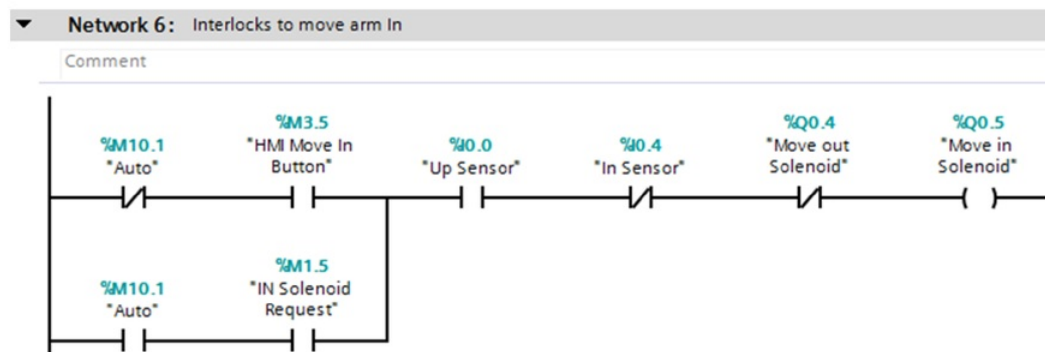
(Figure 13)

Below in figure 14 is how the claw arm moves out. It will either move manually or automatically. The arm must be up in order for the claw to move out which prevents the arm from smashing into stuff, the arm must be in and the in solenoid must not be energized at the same time trying to move out.



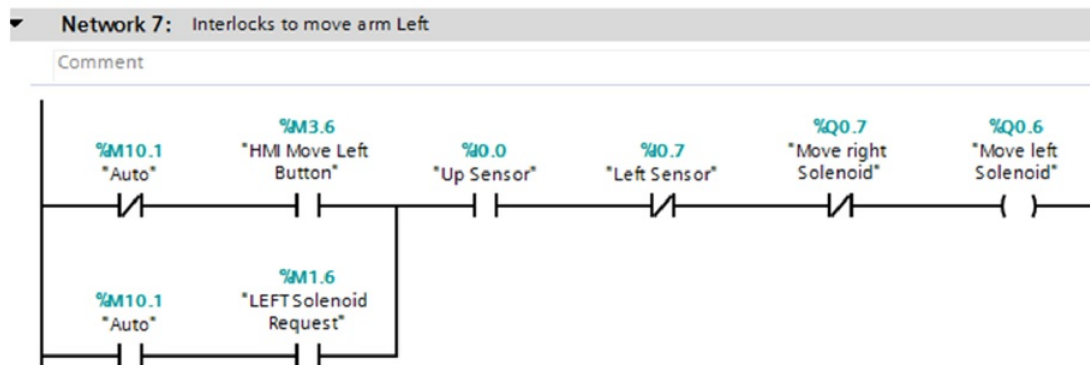
(Figure 14)

Below in figure 15 is the move in arm network. It will either work in automatic or manual. The up sensor must be in to prevent damage to the arm and the arm must be out and the out solenoid off in order for the arm to move in.



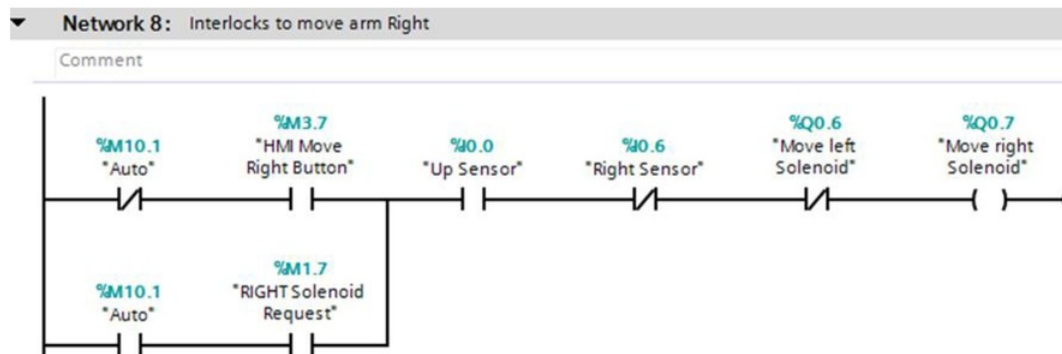
(Figure 15)

Below in figure 16 is the move left arm network which can run either automatically or manually. The up sensor must be made in order to prevent damage to the claw, the arm must be to the right and not trying to go to the right in order for the arm to move to the left.



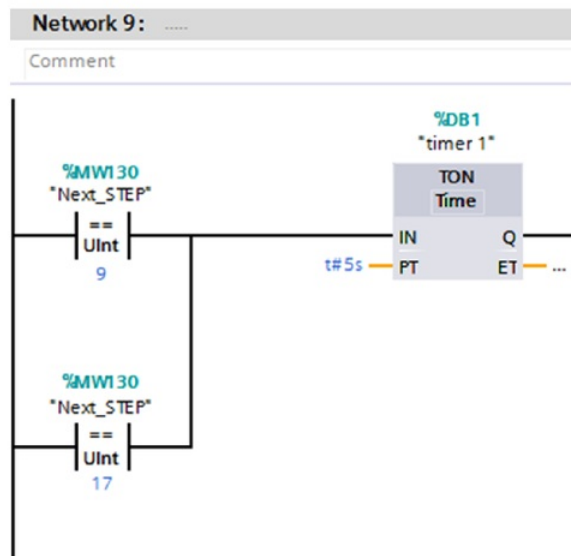
(Figure 16)

Below in figure 17 is the move arm right network. This works either manually or automatically. The arm must be in the up position, to the left and not trying to move left in order for the solenoid to energise and the arm to move to the right.



(Figure 17)

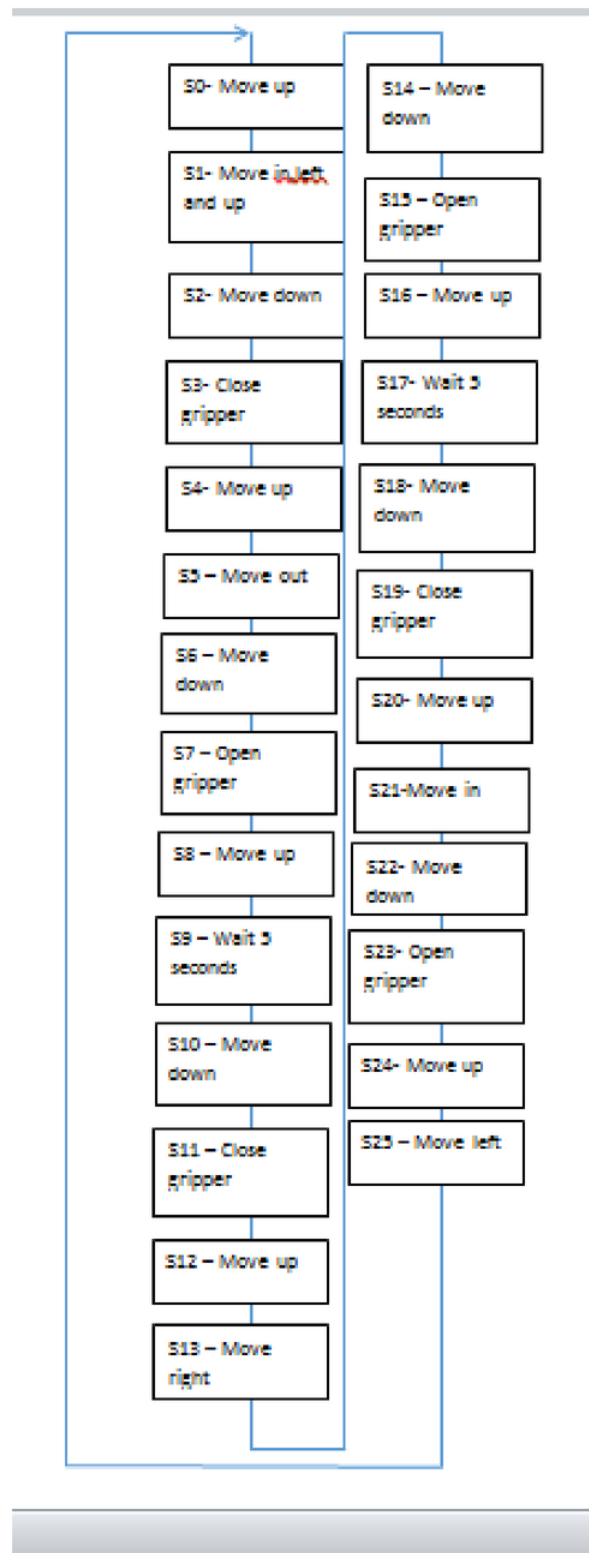
Below in figure 18 is the timer used for the auto sequence, this just delays the step for 5 seconds before moving onto the next step.



(Figure 18)

4. Sequencer

Below is the sequence flow chart used in order to write the automatic SCL programme.



Following below in figure 19 is the first line of code from the SCL programme created.

As seen below the first scan pulse sets step 0 which sets all of the requests to 0 to make sure nothing is trying to move. Then providing the "auto" button is equal to 1 then the solenoid will request to go up, go up and then reset the request and set the next step. The next step requests the solenoid to move up, resets the request and then sets the next step. The next step moves the arm left, in and opens the gripper and resets the request. This sets the next step and waits for a choice of either unloading coils or loading coils, if loading/unload is set to 1 then this will set step 2 of the auto sequence, if it is set to 0 then the unloading sequence will be set and it will skip to step 26. Step 2 sets the arm to move down, resets the request to move down and also sets step 3.

```

1 IF "FSP"=1 OR "Auto" = 0 THEN
2   "STEP" := 0;
3   "Next_STEP" := 0;
4   "UP Solenoid Request" := 0;
5   "DOWN Solenoid Request" := 0;
6   "LEFT Solenoid Request" := 0;
7   "RIGHT Solenoid Request" := 0;
8   "IN Solenoid Request" := 0;
9   "OUT Solenoid Request" := 0;
10  "Gripper OPEN Solenoid Request" := 0;
11  "Gripper CLOSE Solenoid Request" := 0;
12
13 END_IF;
14
15 IF "Auto" = 1 THEN
16   CASE "STEP" OF
17     0:
18       "UP Solenoid Request" := 1;
19       IF ("Up Sensor" = 1) THEN
20         "UP Solenoid Request" := 0;
21         "Next_STEP" := 1;
22       END_IF;
23     1:
24       "LEFT Solenoid Request" := 1;
25       "IN Solenoid Request" := 1;
26       "Gripper OPEN Solenoid Request" := 1;
27       IF (("Gripper Open Sensor" = 1) AND ("Left Sensor" = 1) AND ("In Sensor" = 1)) THEN
28         "LEFT Solenoid Request" := 0;
29         "IN Solenoid Request" := 0;
30         "Gripper OPEN Solenoid Request" := 0;
31       END_IF;
32       IF (("load/Unload" = 1) AND ("HMI START" = 1) OR ("Object sensor" = 1)) THEN
33         "Next_STEP" := 2;
34       END_IF;
35       IF (("load/Unload" = 0) AND ("HMI START" = 1)) THEN
36         "Next_STEP" := 26;
37       END_IF;
38     END_IF;
39
40     2:
41       "DOWN Solenoid Request" := 1;
42       IF ("Down Sensor" = 1) THEN
43         "DOWN Solenoid Request" := 0;
44         "Next_STEP" := 3;
45       END_IF;

```

(Figure 19)

Below in figure 20 Shows steps 3-10. All of the steps apart from step 9 requests the actuator(solenoid valve) to energize in a motion so steps 3-10 excluding 9 move the gripper arm close gripper, up ,out ,down ,open gripper, up and back down. Step 9 is a delay function where it waits for a delay of 5 seconds which is in the network design(shown in figure 18) and then once then timer has timed out it allows the sequence to carry on. After the valve has been energized and the respective sensor for that action has been made then that request will be set back to 0 and the next step will be set.

```

46 3:
47   "Gripper CLOSE Solenoid Request" := 1;
48   IF ("Gripper Closed Sensor") THEN
49     "Gripper CLOSE Solenoid Request" := 0;
50     "Next_STEP" := 4;
51   END_IF;
52 4:
53   "UP Solenoid Request" := 1;
54   IF ("Up Sensor" = 1) THEN
55     "UP Solenoid Request" := 0;
56     "Next_STEP" := 5;
57   END_IF;
58 5:
59   "OUT Solenoid Request" := 1;
60   IF ("Out Sensor" = 1) THEN
61     "OUT Solenoid Request" := 0;
62     "Next_STEP" := 6;
63   END_IF;
64 6:
65   "DOWN Solenoid Request" := 1;
66   IF ("Down Sensor" = 1) THEN
67     "DOWN Solenoid Request" := 0;
68     "Next_STEP" := 7;
69   END_IF;
70 7:
71   "Gripper OPEN Solenoid Request" := 1;
72   IF ("Gripper Open Sensor" = 1) THEN
73     "Gripper OPEN Solenoid Request" := 0;
74     "Next_STEP" := 8;
75   END_IF;
76 8:
77   "UP Solenoid Request" := 1;
78   IF ("Up Sensor" = 1) THEN
79     "UP Solenoid Request" := 0;
80     "Next_STEP" := 9;
81   END_IF;
82 9:
83   IF ("timer 1".Q = 1) THEN
84     "Next_STEP" := 10;
85   END_IF;
86 10:
87   "DOWN Solenoid Request" := 1;
88   IF ("Down Sensor" = 1) THEN
89     "DOWN Solenoid Request" := 0;
90     "Next_STEP" := 11;
91   END_IF;
92

```

(Figure 20)

Below in figure 21 is the code of the loading auto sequence for steps 11-18. Steps 11 – 18 excluding step 17 all perform a move in motion. These are close gripper, move up, move right, move down, open gripper, move up and then move down. Step 17 is a time delay function which can be seen in figure 18. Once this timer has timed out then it will allow the next step to be carried out. All of the motions requested will be rest upon the respective sensor being made.

```

93      11:
94          "Gripper CLOSE Solenoid Request" := 1;
95      IF ("Gripper Closed Sensor") THEN
96          "Gripper CLOSE Solenoid Request" := 0;
97          "Next_STEP" := 12;
98      END_IF;
99
100     12:
101         "UP Solenoid Request" := 1;
102     IF ("Up Sensor" = 1) THEN
103         "UP Solenoid Request" := 0;
104         "Next_STEP" := 13;
105     END_IF;
106
107     13:
108         "RIGHT Solenoid Request" := 1;
109     IF ("Right Sensor" = 1) THEN
110         "RIGHT Solenoid Request" := 0;
111         "Next_STEP" := 14;
112     END_IF;
113
114     14:
115         "DOWN Solenoid Request" := 1;
116     IF ("Down Sensor" = 1) THEN
117         "DOWN Solenoid Request" := 0;
118         "Next_STEP" := 15;
119     END_IF;
120
121     15:
122         "Gripper OPEN Solenoid Request" := 1;
123     IF ("Gripper Open Sensor" = 1) THEN
124         "Gripper OPEN Solenoid Request" := 0;
125         "Next_STEP" := 16;
126     END_IF;
127
128     16:
129         "UP Solenoid Request" := 1;
130     IF ("Up Sensor" = 1) THEN
131         "UP Solenoid Request" := 0;
132         "Next_STEP" := 17;
133     END_IF;
134
135     17:
136         IF ("timer 1".Q = 1) THEN
137             "Next_STEP" := 18;
138         END_IF;
139
140     18:
141         "DOWN Solenoid Request" := 1;
142     IF ("Down Sensor" = 1) THEN
143         "DOWN Solenoid Request" := 0;
144         "Next_STEP" := 19;
145     END_IF;

```

(Figure 21)

Below in figure 22 is the code for steps 19- 25. These are the motions for gripper close, move up, move in, move down, open gripper, move up and move left . These requested movements are all reset by their sensors being made. Once step 25 has been reached then it will set step 0 again and work for a human input in order to start the loading sequence again.

```

139      19:      -
140              "Gripper CLOSE Solenoid Request" := 1;
141      IF ("Gripper Closed Sensor") THEN
142              "Gripper CLOSE Solenoid Request" := 0;
143              "Next_STEP" := 20;
144      END_IF;
145      20:
146              "UP Solenoid Request" := 1;
147      IF ("Up Sensor" = 1) THEN
148              "UP Solenoid Request" := 0;
149              "Next_STEP" := 21;
150      END_IF;
151      21:
152              "IN Solenoid Request" := 1;
153      IF ("In Sensor" = 1) THEN
154              "IN Solenoid Request" := 0;
155              "Next_STEP" := 22;
156      END_IF;
157      22:
158              "DOWN Solenoid Request" := 1;
159      IF ("Down Sensor" = 1) THEN
160              "DOWN Solenoid Request" := 0;
161              "Next_STEP" := 23;
162      END_IF;
163      23:
164              "Gripper OPEN Solenoid Request" := 1;
165      IF ("Gripper Open Sensor" = 1) THEN
166              "Gripper OPEN Solenoid Request" := 0;
167              "Next_STEP" := 24;
168      END_IF;
169      24:
170              "UP Solenoid Request" := 1;
171      IF ("Up Sensor" = 1) THEN
172              "UP Solenoid Request" := 0;
173              "Next_STEP" := 25;
174      END_IF;
175      25:
176              "LEFT Solenoid Request" := 1;
177      IF ("Left Sensor" = 1) THEN
178              "LEFT Solenoid Request" := 0;
179              "Next_STEP" := 0;
180      END_IF;

```

(Figure 22)

Below in figure 23 is the sequence for the unloading coils sequence. This sequence will unload every coil at their set place and then return to the ready position. Steps 26-33 performs actions move down, close gripper, move up, move right, move down, open gripper, move up and move left and out. These steps are reset upon the making of their sensors.

```

181      26:
182      "DOWN Solenoid Request" := 1;
183      IF ("Down Sensor" = 1) THEN
184      "DOWN Solenoid Request" := 0;
185      "Next_STEP" := 27;
186      END_IF;
187
188      27:
189      "Gripper CLOSE Solenoid Request" := 1;
190      IF ("Gripper Closed Sensor") THEN
191      "Gripper CLOSE Solenoid Request" := 0;
192      "Next_STEP" := 28;
193      END_IF;
194
195      28:
196      "UP Solenoid Request" := 1;
197      IF ("Up Sensor" = 1) THEN
198      "UP Solenoid Request" := 0;
199      "Next_STEP" := 29;
200      END_IF;
201
202      29:
203      "RIGHT Solenoid Request" := 1;
204      IF ("Right Sensor" = 1) THEN
205      "RIGHT Solenoid Request" := 0;
206      "Next_STEP" := 30;
207      END_IF;
208
209      30:
210      "DOWN Solenoid Request" := 1;
211      IF ("Down Sensor" = 1) THEN
212      "DOWN Solenoid Request" := 0;
213      "Next_STEP" := 31;
214      END_IF;
215
216      31:
217      "Gripper OPEN Solenoid Request" := 1;
218      IF ("Gripper Open Sensor" = 1) THEN
219      "Gripper OPEN Solenoid Request" := 0;
220      "Next_STEP" := 32;
221      END_IF;
222
223      32:
224      "UP Solenoid Request" := 1;
225      IF ("Up Sensor" = 1) THEN
226      "UP Solenoid Request" := 0;
227      "Next_STEP" := 33;
228      END_IF;
229
230      33:
231      "LEFT Solenoid Request" := 1;
232      "OUT Solenoid Request" := 1;
233      IF (("Left Sensor" = 1) AND ("Out Sensor")) THEN
234      "LEFT Solenoid Request" := 0;
235      "OUT Solenoid Request" := 0;

```

(Figure 23)

Below in figure 24 are steps 34-40. The steps for these are move down, close gripper, move up, move right and move in, move down, open gripper, move up. These movements are all reset by their respective sensors so once the solenoids have fully extended.

```

228      "OUT Solenoid Request" := 0;
229      "Next_STEP" := 34;
230  END_IF;
231
232  34:
233      "DOWN Solenoid Request" := 1;
234      IF ("Down Sensor" = 1) THEN
235          "DOWN Solenoid Request" := 0;
236          "Next_STEP" := 35;
237      END_IF;
238
239  35:
240      "Gripper CLOSE Solenoid Request" := 1;
241      IF ("Gripper Closed Sensor" = 1) THEN
242          "Gripper CLOSE Solenoid Request" := 0;
243          "Next_STEP" := 36;
244      END_IF;
245
246  36:
247      "UP Solenoid Request" := 1;
248      IF ("Up Sensor" = 1) THEN
249          "UP Solenoid Request" := 0;
250          "Next_STEP" := 37;
251      END_IF;
252
253  37:
254      "RIGHT Solenoid Request" := 1;
255      "IN Solenoid Request" := 1;
256      IF (("Right Sensor" = 1) AND ("In Sensor")) THEN
257          "RIGHT Solenoid Request" := 0;
258          "IN Solenoid Request" := 0;
259          "Next_STEP" := 38;
260      END_IF;
261
262  38:
263      "DOWN Solenoid Request" := 1;
264      IF ("Down Sensor" = 1) THEN
265          "DOWN Solenoid Request" := 0;
266          "Next_STEP" := 39;
267      END_IF;
268
269  39:
270      "Gripper OPEN Solenoid Request" := 1;
271      IF ("Gripper Open Sensor" = 1) THEN
272          "Gripper OPEN Solenoid Request" := 0;
273          "Next_STEP" := 40;
274      END_IF;
275
276  40:
277      "UP Solenoid Request" := 1;
278      IF ("Up Sensor" = 1) THEN
279          "UP Solenoid Request" := 0;
280          "Next_STEP" := 41;
281      END_IF;

```

(Figure 24)

Below in figure 25 is the unloading sequence steps 41-47. The movements for these steps are move out, move down, close gripper, move up, move in, move down and then open gripper. After each step is completed the request is reset by the perspective sensor. After step 47 has been complete it will set back to the ready state which will bring the claw back to where it needs to be by running the code in figure 19 to make sure the claw is in the correct place to start next time.

```

275      41:
276          "OUT Solenoid Request" := 1;
277      IF ("Out Sensor" = 1) THEN
278          "OUT Solenoid Request" := 0;
279          "Next_STEP" := 42;
280      END_IF;
281      42:
282          "DOWN Solenoid Request" := 1;
283      IF ("Down Sensor" = 1) THEN
284          "DOWN Solenoid Request" := 0;
285          "Next_STEP" := 43;
286      END_IF;
287      43:
288          "Gripper CLOSE Solenoid Request" := 1;
289      IF ("Gripper Closed Sensor") THEN
290          "Gripper CLOSE Solenoid Request" := 0;
291          "Next_STEP" := 44;
292      END_IF;
293      44:
294          "UP Solenoid Request" := 1;
295      IF ("Up Sensor" = 1) THEN
296          "UP Solenoid Request" := 0;
297          "Next_STEP" := 45;
298      END_IF;
299      45:
300          "IN Solenoid Request" := 1;
301      IF ("In Sensor" = 1) THEN
302          "IN Solenoid Request" := 0;
303          "Next_STEP" := 46;
304      END_IF;
305      46:
306          "DOWN Solenoid Request" := 1;
307      IF ("Down Sensor" = 1) THEN
308          "DOWN Solenoid Request" := 0;
309          "Next_STEP" := 47;
310      END_IF;
311      47:
312          "Gripper OPEN Solenoid Request" := 1;
313      IF ("Gripper Open Sensor" = 1) THEN
314          "Gripper OPEN Solenoid Request" := 0;
315          "Next_STEP" := 0;
316      END_IF;
317
318      END_CASE;
319
320      "STEP" := "Next_STEP";
321  END_IF;

```

(Figure 25)

5. Testing

As shown below in figure 26 is the manual testing table for all of the solenoids and sensors.

Movement	Expected outcome	Actual outcome
Gripper open	Gripper open solenoid should energise and once open the gripper open sensor should show as made on the HMI. Gripper should physically open.	Gripper opened as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.
Gripper Closed	Gripper close solenoid should energise and once open the gripper close sensor should show as made on the HMI. Gripper should physically close.	Gripper closed as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.
Move up	Move up solenoid should energise and once fully up the up sensor should be made and display on the HMI. The arm should be in the up position.	Arm moved up as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.
Move down	Move down solenoid should energise and once fully down the down sensor should be made and display on the HMI. The arm should be in the down position.	Arm moved down as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.
Move left	Move left solenoid should energise and once fully left the left sensor should be made and display on the HMI. The arm should be to the left.	Arm moved left as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.
Move right	Move right solenoid should energise and once fully right the right sensor should be made and display on the HMI. The arm should be to the right.	Arm moved right as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.
Move in	Move in solenoid should energise and once fully in the in sensor should be made and display on the HMI. The arm should be to the in.	Arm moved in as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.
Move out	Move out solenoid should energise and once fully out the out sensor should be made and display on the HMI. The arm should be to the out.	Arm moved out as expected, when sensor was made it was displayed on the HMI. Solenoid energising display on HMI illuminated.

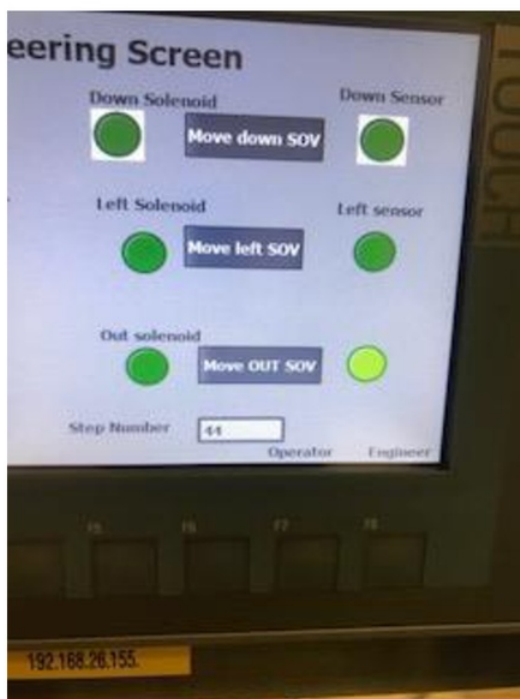
(Figure 26)

As shown below in figure 27 is the automatic sequence testing.

Movement	Expected outcome	Actual outcome
Automatic loading of the coils.	When powered up robot should return to home position. Sequence should start when all the correct parameters are met. The sequence should move the coil from start to finish with 2 5 second wait intervals, the sequence should stop when using the freeze button.	Robot moved to home position .Sequence ran perfectly from initial step to final step, coil was moved from start to finish and when stop was pressed the sequence stopped and resumed when started on the HMI again.
Automatic unloading of the coils	When powered up robot should return to home position. Sequence should start unloading of coils from every platform and place them in the outbound slope. Sequence should freeze if stop button is pressed.	Robot moved to home position. Sequence unloaded coils that were placed on platforms. Sequence froze when the stop button was pressed.

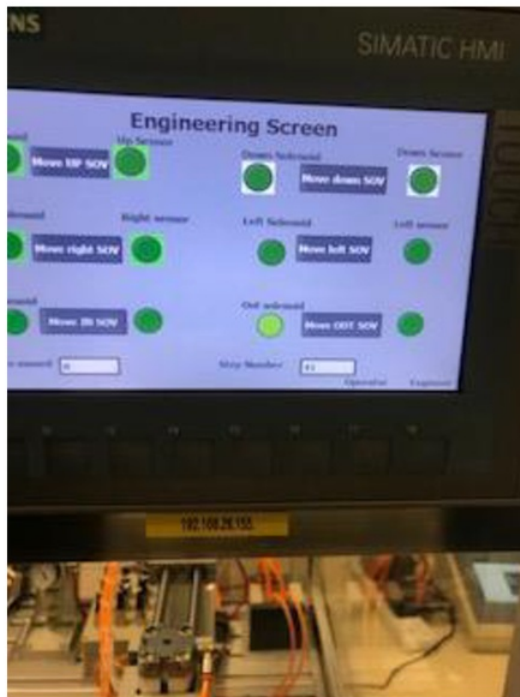
(Figure 27)

As evidence of testing the sequences the following figures illustrate commands being performed. As shown below in figure 28 shows the out solenoid sensor being made showing that the arm is out. It also shows step number 44 of the unloading sequence proving that the step number counter works.



(Figure 28)

Below in figure 29 shows the HMI indication screen which shows the up sensor being made, the right sensor being made and also the out solenoid being energised.



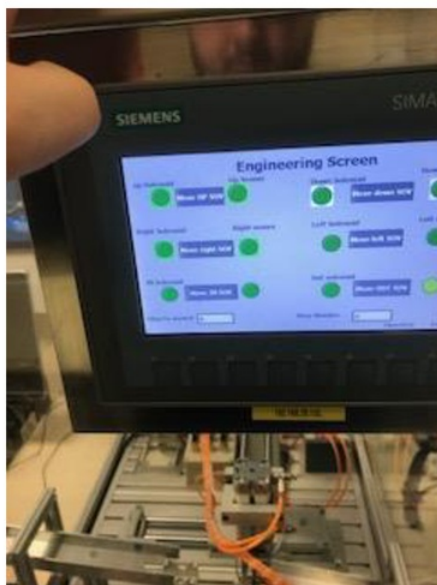
(Figure 29)

Below in figure 30 show the operator screen which shows that auto has been selected, unload coils has been selected and it displays the step number. Also in figure 31 shows the coil being transported from the last platform to the exit slope, however hmi stop was pressed and the sequence was paused until start was pressed again.



(Figure 30+31)

Below in figure 32 is the engineering screen which displays the right sensor being made and also the up solenoid being energised.



(Figure 33)

6. Conclusion and evaluation

Concluding on this assignment the aims and objectives of this assignment have been met, a manual testing screen and also two different auto sequences were made with safety interlocks carefully chosen and also freeze buttons were introduced as an extra feature. Individual knowledge of the SCL language within the Siemens software was vastly improved as it was a first time coming across this particular language. Also testing and commissioning knowledge has been improved as due to the amount of commands in this programme, it took a lot of thinking and development to produce a suitable testing table which ensured all of the possible tests were being done. Evaluating this assignment further personal work could have been done with the evidence for testing. During the testing phase it was hard to get photos as the claw was moving so for future improvements a suggestion would be to monitor the program while it was operating and take screenshots of the online version of the program in order to provide more effective evidence for the testing.

The purpose of the freeze button was to capture the evidence.

FINAL GRADE

GENERAL COMMENTS

Instructor

61 / 100

PAGE 1

PAGE 2

PAGE 3

Text Comment. avoid personal references, write in the third person

PAGE 4

PAGE 5

PAGE 6

Text Comment. Incorrect, figure 4 actually shows the reed switch junction box

Text Comment. Incorrect, it would have to be a capacitive sensor to detect plastic

PAGE 7

Text Comment. The object sensor (I1.5) is in fact a 3 wire device.

Text Comment. Neutral

Text Comment. Good labelling

Text Comment. There are two spare outputs

Text Comment. The other side of each coil must be grounded otherwise they will not energize

PAGE 8

Text Comment. A useful table

PAGE 9

Text Comment. The system is not handling coils.

PAGE 10

PAGE 11

Text Comment. This is a well designed ladder network

Text Comment. Good design feature

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

Text Comment. The purpose of the freeze button was to capture the evidence.

GRADING FORM: CSG-RUBIC

RESEARCH-INFORMED LITERATURE

 n/a

KNOWLEDGE AND UNDERSTANDING

 Section 1: Hardware Investigation - A good section that explains the hardware used with a helpful wiring diagram and some good photos of the system device. However, there are some errors. (7/10)
 Section 1: System Design and I/O list - A good I/O list with well design ladder networks. However, the SFC chart is blurred and only covers the loading sequence. (6/10) Section 2: Program Explanation – Ladder networks have been explained in some detail together with sections of the SCL code for the auto sequence. (8/10)

21

ANALYSIS

 Section 3: System Testing – A good testing table for manual control but the auto sequences need to be more detailed and broken down into each step. Further testing is required. There are some images that provide photographic evidence but there could be a lot more. (6/10) Section 3: Maintenance and Reliability - These issues have been considered in the design and testing of the system but a separate subsection could have been devoted to it, explaining

16

how they have been dealt with. (6/10) Section 4: Conclusions and Recommendations - Some conclusions and suggestions offered. (4/10)

PRACTICAL APPLICATION AND DEP.

 Section 2: Program Network Comments - Most ladder networks have titles but no comments and no comments have been included in the SCL code. (4/10) Section 2: HMI Screen Designs – Well designed screens for the operator and the engineer. However, you could have included password protection for the engineer's screen. (9/10) Section 3: Presentation of Test Results - Good but more evidence is needed as photos. (5/10)

18

SKILLS FOR PROFESSIONAL PRAC.

 All sections - Generally well structured and presented (but the SFC chart is blurred). (6/10)

6