1605160

by Steffan Davies

FILE	15323_STEFFAN_DAVIES_1605160_179196_2101448522.DOCX (9.3M)		
TIME SUBMITTED	24-MAY-2021 07:57PM (UTC+0100)	WORD COUNT	2319
SUBMISSION ID	154656442	CHARACTER COUNT	12231

Course: HNC Electrical and Electronic Engineering

Module: SGE4004 Programmable Logic Controllers

Assignment: 2.Level and flow control system

Name: Steffan Wyn Davies

University number: 1605160

Date submitted: 24/05/2021

Word count: 2349

Contents

1.Research 3
Ladder logic 3
Instruction list 3
Function Block Diagram 3
Structured Control Language 4
GRAPH 4
2.Hardware design 5
3.Programme design and explanation 6
4.System testing 8
5.Meeting customer needs and team working 9
6.Conclusions and recommendations 10
7.Citation List 11
8.Appendix 12

Figure 1-Drain valve flow control 6
Figure 2-Fill valve level control HMI screen 7
Figure 3-Testing table 8
Figure 4-LAD example 12
Figure 5-IL example 12
Figure 6-FBD example 13
Figure 7-SCL example 13
Figure 8-GRAPH example 14
Figure 9 14
Figure 10 15
Figure 11 15
Figure 12 16
Figure 13 16
Figure 14- digital IO drawing 31
Figure 15-analogue input module drawing 32
Figure 16-analogue output module drawing 33

1.Research

There are several IEC languages available to PLC programmers. TIA portal uses their own programming languages which correspond to IEC 61131-3 standards (Siemens, 2021), as can be seen in the comparison table below:

Correctly cited

STEP 7(TIA portal) language	IEC language 61131-3 correspondent
Ladder Logic (LAD)	LD
Instruction List (STL)	IL
Function Block Diagram (FBD)	FBD
Structured Control Language (SCL)	ST
S7 Graph	SFC

Ladder logic

Useful information

Ladder logic is one of the most popular programming language, it involves dragging and dropping elements to form networks. Jones states that "It is a graphical PLC programming language which expresses logic operations with symbolic notation. Ladder logic is made out of rungs of logic, forming what looks like a ladder (2017, p. 1). An example of ladder logic can be seen in figure 1.

Correctly cited

Instruction list

Instruction list(STL) is one of the less popular IEC programming language. Instruction list as the name suggests provides a programme with a sequence of instructions. One line will represent one operation, e.g. MUL will represent multiply, which will be followed by an operand. This programming language has the ability to execute much faster and use less memory compared to graphical programming languages such as LAD and FBD. However, IL has no code structure making it harder to fault find (VISAYA, 2019). An example of IL can be seen in figure 2.

Correctly cited

Function Block Diagram

Function block diagram is another graphical programming language and is often used to mimic a specific process. Romanov states that "function block diagrams in PLC programming allow users to create a system that takes multiple inputs, sends them through various instruction blocks and changes specified outputs" (2020, p. 1). In FBD, all functions are put into blocks. You can then connect the output of one function block to an input of another, resulting in a diagram of function blocks (Jones, 2017) An example of FBD can be seen in figure 3.

Correctly cited

Correctly cited

Structured Control Language

Structured Control Language(SCL) is a high level, text based language that corresponds to Structured text(ST) in the IEC 61131-3. This programming language uses operators such as NOT, AND, OR, equal to(=),greater/less than(<>) with a condition after the operator in order to from a list of instructions. According to (Anton, 2019), common uses for SCL include math, program flow and communication. Comments can be added to SCL networks in order to assist with fault finding activities. This can be seen in figure 4.

Improper Citation

GRAPH

GRAPH is a graphic programming languages that is used for creating sequential control. It corresponds to the IEC language Sequential Flow Chart(SFC). Siemens states that "Within a GRAPH function block you program your program in the form of sequence cascades; whereby sequential processes can be quickly program in a clearly organized manner" (2021, p. 4). This language is fairly simple to programme, and includes extending the cascades with elements such as step, transition, jump, open alternative branch (OR branch) and end sequence. Graph is very useful for sequential processes that have to be executed in a certain order. An example of graph can be seen in figure 5.

Correctly cited

2.Hardware design

The PLC used for this system is the S7-1214C. Additional analogue input (AI4X13BIT_1) and analogue output (AQ4 (AG, 2021)X14BIT_1) were added to increase number of IO. The IO list for this system can be seen below:

Good information

Input/Output list	
Inputs	**Address**
Start button	I0.0
Reset button	I0.1
Stop button	I0.2
Factory IO running DI	I0.3
Level meter AI real	ID30
Flow meter AI real	ID34
Setpoint potentiometer	ID38
Outputs	
Start light	Q0.0
Reset light	Q0.1
Stop light	Q0.2
Fill valve AQ real	QD30
Drain valve AQ real	QD34
Setpoint panel meter	QD38
Process variable panel meter	QD42

The start, stop and reset buttons are all digital inputs to the PLC. The level meter, flow meter and setpoint potentiometers are analogue inputs. By looking at the factory IO programme we can see that a capacitive level meter is used for level control and a differential pressure transmitter is used for flow control. They both produce a 0-10V feedback signal in order to tell the PLC what state the tank is in. For example, if the level meter is reading 10V then the level of the tank is at maximum. The flow meter, level meter and setpoint potentiometer are then manipulated in the network programme using functions such as MOVE,MUL,SUB etc. Using these functions will then determine the CV for flow and level, which will then send a signal to the drain valve/fill valve to open or close depending on the setpoint.

3.Programme design and explanation

This programme is made up of 3 programme blocks, the main network, flow control and level control. The HMI screens provide the engineer/operator with all necessary indications of what the program is doing. This programme allows the user to fill and empty a tank using manual and semi-automatic control. The semi-automatic control contains on/off control as well as PID control. For the on/off control method, the deadband can be chosen by the engineer. Meanwhile the engineer can also alter the tuning values for PID control method in order to ensure the process is running smoothly and the system doesn't overshoot/undershoot. It's important that the programme can reach the setpoint effectively and accurately, which is why this programme has the ability to tune the values.

The flow control and level control work together in this programme to maintain the setpoint. For example, if the tank was full and the setpoint was changed to 100, then level control would be stopped and the flow drain valve would be opened using semi-automatic flow control which would see the level drop to the desired setpoint. The trend graphs on the HMI screen can assist the engineer in choosing tuning values, making this programme suitable for the customer. The 2 HMI screens can be seen below:

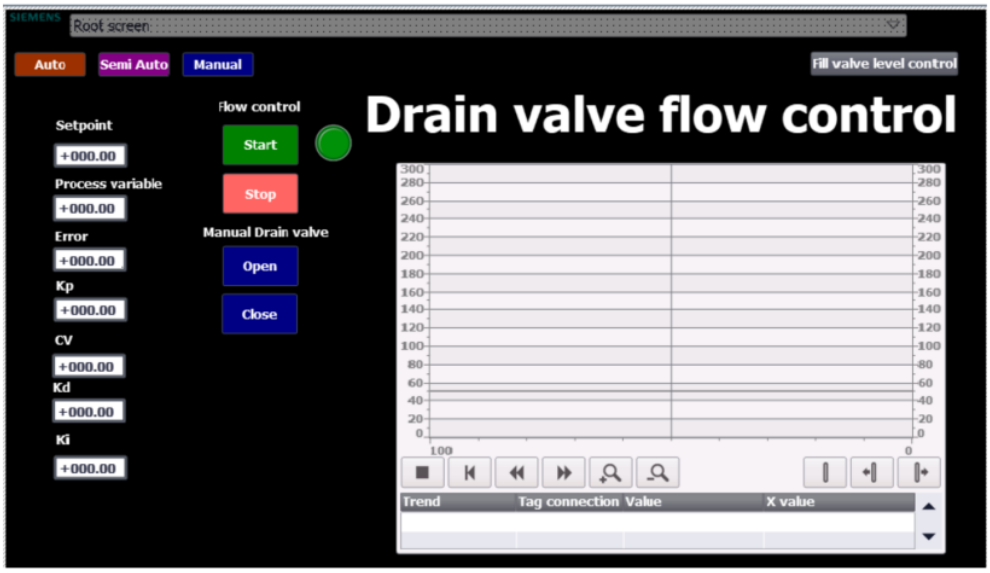


Figure 1-Drain valve flow control

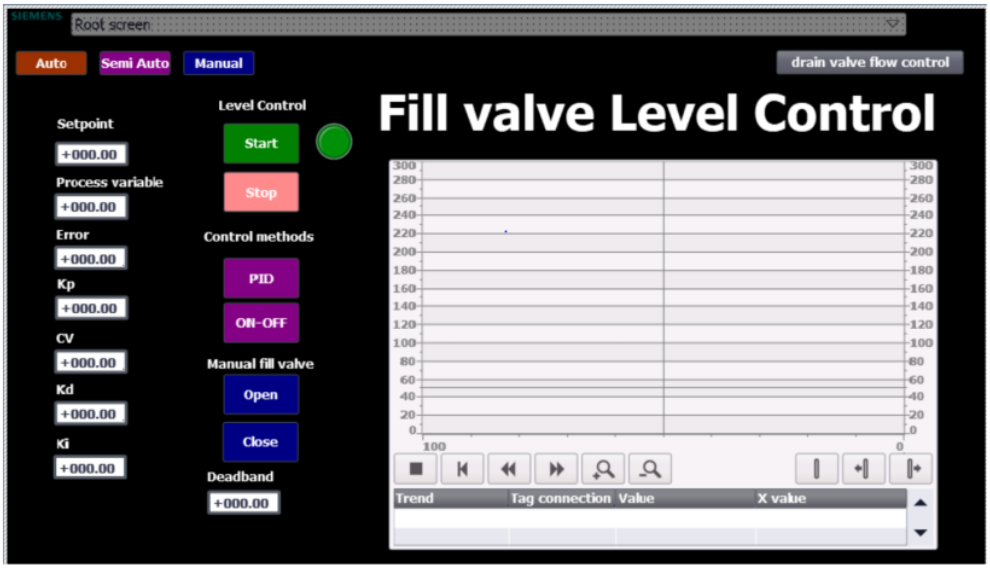


Figure 2-Fill valve level control HMI screen

4.System testing

Test description	Expected result	Actual result	Pass/fail	Appendix reference
Enter HMI fill valve level control screen and manually fill the tank to 100 and then close the valve	When in manual mode, the tank should start filling when "open" is pressed and then stop filling when "close" is pressed	Manually opening the fill valve successfully filled the tank to the desired level and then when "close" was pressed the fill valve closed	pass *the open and close buttons should have incremented and decremented the valve position*	Figure 9
Enter HMI drain valve level control screen and manually drain the valve until tank is empty	When in manual mode and "open" is pressed, the drain valve should begin to open and the tank level should drop	The drain valve was operated successfully and the tank was emptied successfully	pass	Figure 10
Enter HMI fill valve level control screen and select "semi-automatic mode" . Choose a setpoint of 200 and select PID mode	When in semi-automatic mode and in PID control method, the tank level should rise to 200. Tuning might be required to improve accuracy	The level of the tank successfully filled to 200- no tuning was required	pass	Figure 11
Enter HMI fill valve level control screen and select "ON-OFF" mode. Increase setpoint to 280.	When in semi-automatic mode and "ON-OFF" control method, the level of the tank should rise to the desired setpoint. The engineer should also be able to adjust the dead band	The level of the tank reached the desired setpoint in "ON-OFF" mode. The dead band was also adjustable on the HMI screen	pass	Figure 12
Enter HMI drain valve level control and select semi-automatic flow control. When the tank is full, change the setpoint to 100	If the tank is full, when semi-automatic flow control is selected and level control is off, the drain valve should open and the water level should drop to the setpoint of 100	Level control was switched off and semi-automatic flow control was selected. The water level dropped down to 100 as expected	pass	Figure 13

Figure 3-Testing table

What about full auto mode?

5.Meeting customer needs and team working

Team working was done to ensure that the customers' needs were achieved. This was achieved by taking it in turns to share each other's programme to firstly ensure it was working properly, and secondly to suggest any improvements that could be done to ensure the customers' needs are met. The team working group consisted of Steffan Davies(author), Jason Booth and Ethan Rigby. Monitor mode was used to discover a fault in the flow control network of Jason booths programme. Each person took it in turns to share their programme and HMI design. As a result of the team working discussions, it was recommended that the buttons pressed on the HMI should change colour when pressed. This was to ensure the user/customer have a clear indication of the current status of the system.

The maintenance and reliability was considered when designing the programme. In order to ensure the reliability of the programme, trend graphs were added to the HMI screen so that the operator can alter the tuning values for PID control. If the tank was feeding a critical process then it's important that the PID control is working as efficiently and reliable as possible. Maintenance of this system should be relatively easy in future because all networks have network comments added to them explaining what each element of the network does. This would very much help anyone who would need to carry out maintenance tasks or fault finding procedures on the programme.

what about auto mode?

6.Conclusions and recommendations

The aims and objectives of this assignment was to design a system to control the level and flow of a tank. This was done successfully by producing IO lists, drawings, and designing HMI screens as well as the programme design. By looking at the testing table labelled figure 3, we can see that the system has passed all the tests required. However, further work that could be done to improve the system could be to add an entirely automatic mode to the system. This would be done by using SCL to write a list of instructions so that when automatic mode is selected, the tank will fill up to a certain level and then flow control should initiate and begin to run to a setpoint rate. This would very much improve the efficiency of the system as the operator wouldn't have to press as many buttons on the HMI screens. However, the current system still successfully controls flow and level and therefore satisfy the assessment criteria.

not auto mode

good observation

When testing the programme, the bar graphs on the HMI screen showed that PID control ran smoother than on/off control which was overshooting the setpoint. As a result of this, it can be concluded that PID control is more effective in a critical system where stable control is required, while on/off control is suitable for simpler applications where accuracy isn't as vital. PID control ensures there is no error between the process value and the setpoint, therefore making it more effective than on/off control. (Elprocus, 2021) Correctly cited

When designing the HMI screens and the main programme, it's important to take into consideration the requirements of the customer. By working in groups, everyone could contribute ideas to make the system more efficient and more user friendly. One of the ideas that came from the group working was to change the colour of the buttons on the HMI when pressed. This helped in showing the customer the current status of all the devices.

Maintenance and reliability are also a very important factor when designing a programme. By adding network comments to every network, any future maintenance/fault finding activities would be much easier for the customer, therefore making this system more user friendly and desirable for the customer. In addition, the graphs used for the HMI screen allows the engineer to see how the system is behaving, and allows the user to alter the tuning values.

In order to improve this programme further, one idea would be to add an operator screen with auto start/stop along with mimic graphs for flow and level. Furthermore, password protecting the engineer screens could be a sensible idea in order to ensure that only the engineer can tune the PID control. If anyone can alter the tuning values then it could affect the process.

these are good conclusions
and recommendations

7.Citation List

A good range of source correctly listed

AG, S., 2021. *Siemens.* [Online]
Available at: https://support.industry.siemens.com/cs/mdm/91696622?c=61288500491&lc=en-TZ
[Accessed 23 May 2021].

Anton, K., 2019. *pccweb.com.* [Online]
Available at: https://www.pccweb.com/wp-content/uploads/2019/11/S17-SCL.pdf
[Accessed 08 03 2021].

Elprocus, 2021. *ELPROCUS.* [Online]
Available at: https://www.elprocus.com/the-working-of-a-pid-controller/
[Accessed 24 May 2021].

Jones, P., 2017. *PLC academy.* [Online]
Available at: https://www.plcacademy.com/author/peter/
[Accessed 08 March 2021].

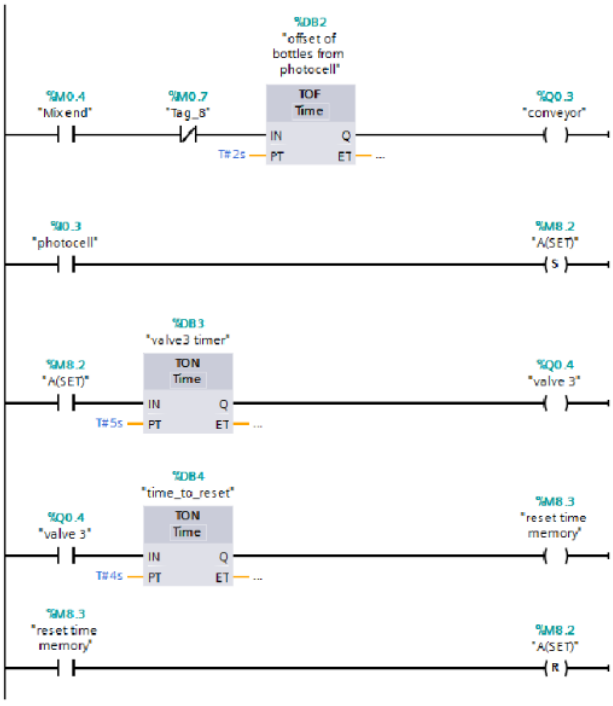
Romanov, V., 2020. *solisPLC.* [Online]
Available at: https://www.solisplc.com/tutorials/function-block-programming
[Accessed 10 March 2021].

Siemens, 2021. *Siemens.* [Online]
Available at: https://support.industry.siemens.com
[Accessed 08 03 2021].

SIEMNS, 2021. *SIEMENS.* [Online]
Available at: https://mall.industry.siemens.com/mall/en/us/Catalog/Products/10015491
[Accessed 10 March 2021].

VISAYA, 2019. *visaya.solutions.* [Online]
Available at: https://visaya.solutions/en/video/plc-programming-languages
[Accessed 10 March 2021].

8.Appendix



(AG, 2021)

Figure 4-LAD example

Network 30: EQ130 - Read current transducer A/D value

Comment

```
// Convert CT signal to a percentage
      CALL  "AD Norm (int)"
         AD_in      :="EQ130 CT":P
         AD_0       :=0          // A/D value at 4mA, aliminate base noise here
         Scale_0    :=0          // Percentage at 4mA
         AD_1       :=27648      // A/D value as measured under load
         Scale_1    :=100        // Percentage at given A/D value
         Scale_LL   :=0          // Output lower limit
         Scale_UL   :=150        // Output upper limit
         Scaled_Out :="HMI word read"."EQ130 Amps"
```

Figure 5-IL example

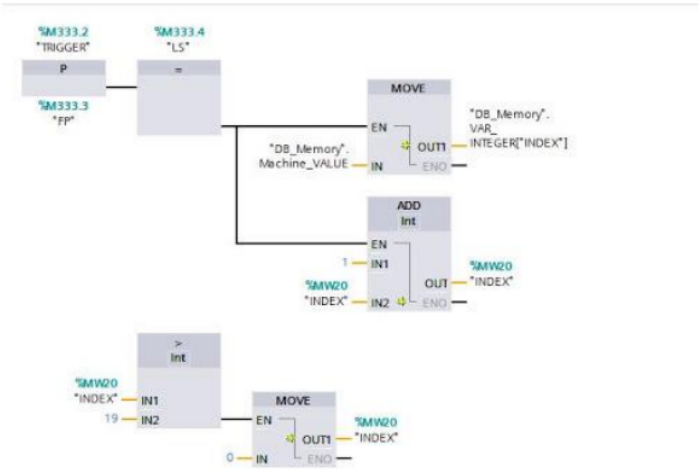


Figure 6-FBD example

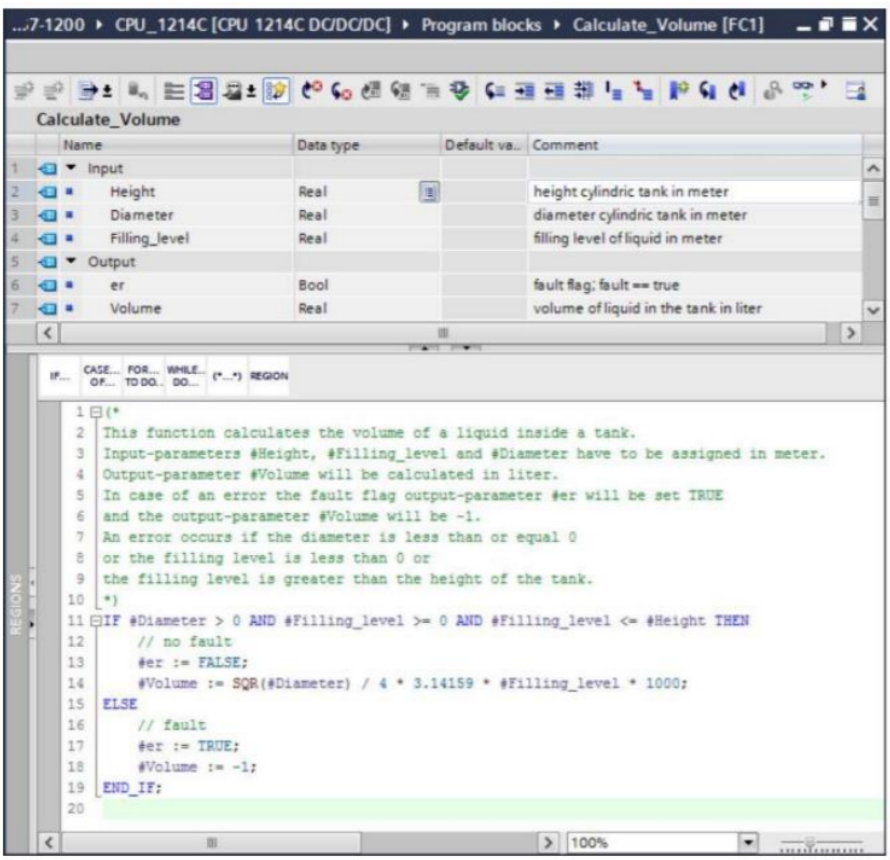
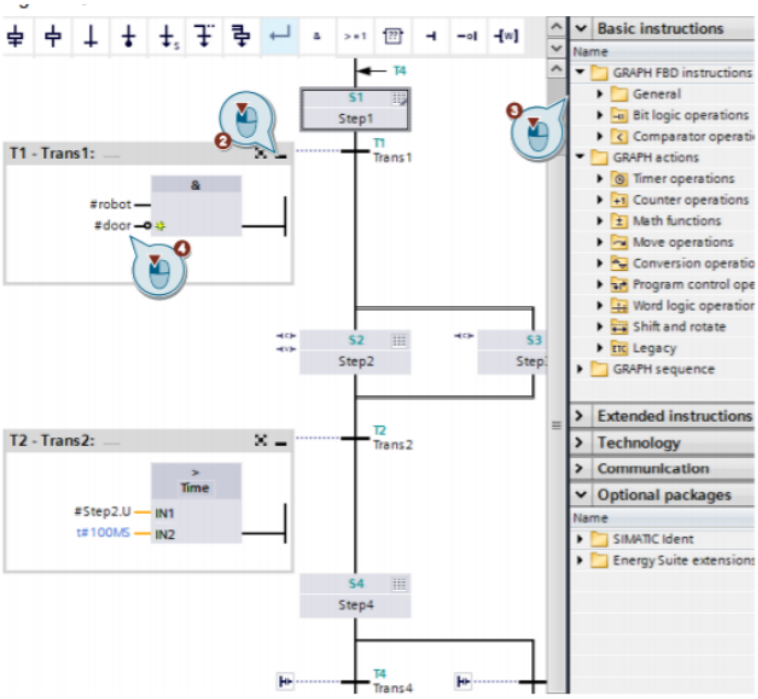


Figure 7-SCL example

(Anton, 2019)



(Siemens, 2021)

Figure 8-GRAPH example

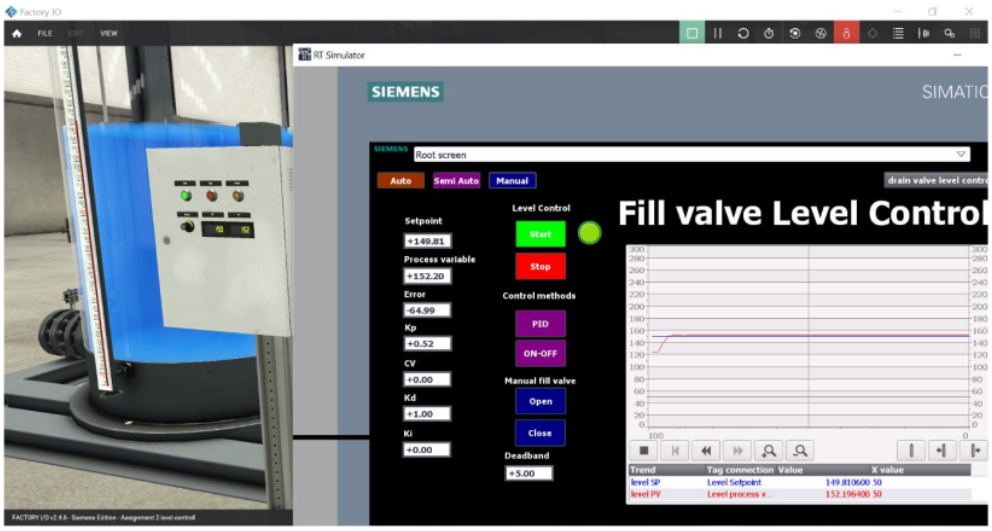


Figure 9

which mode are you in?

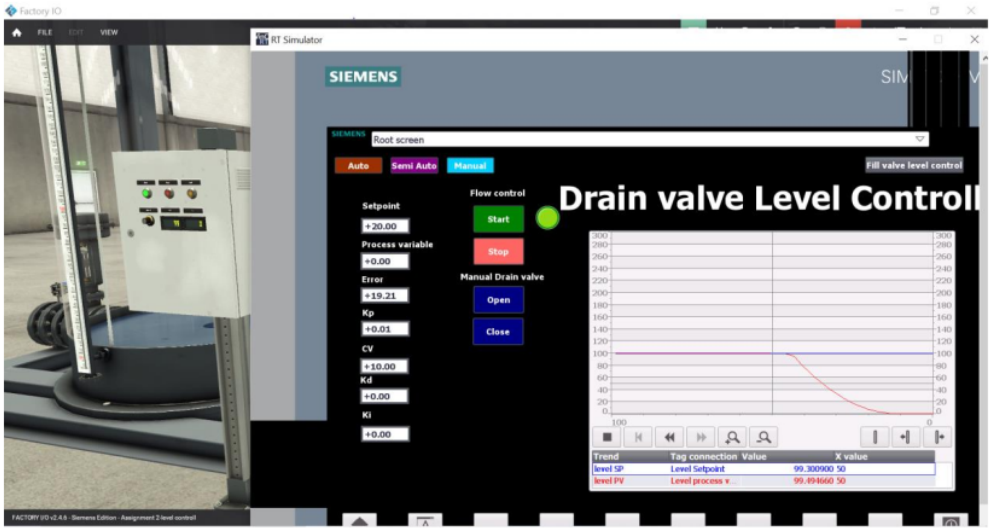


Figure 10

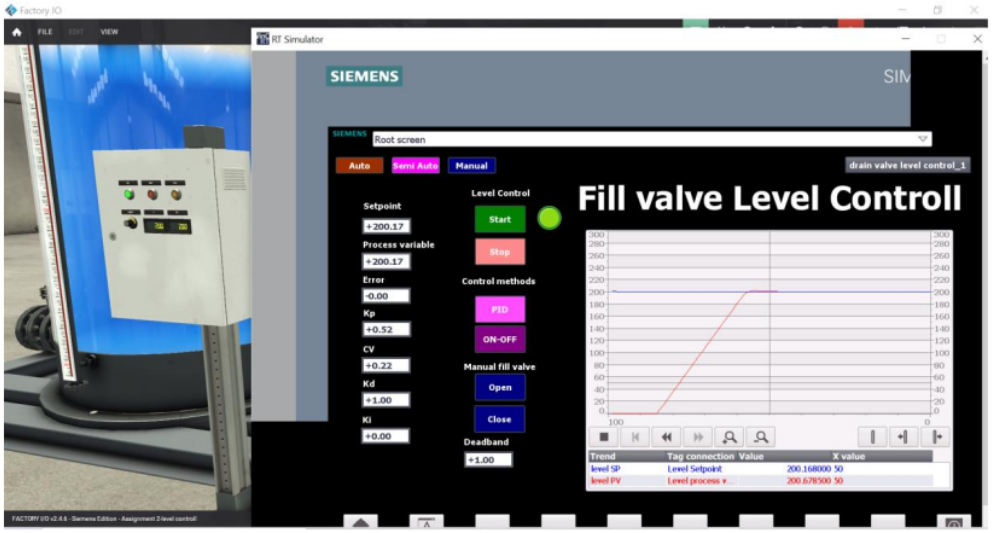


Figure 11

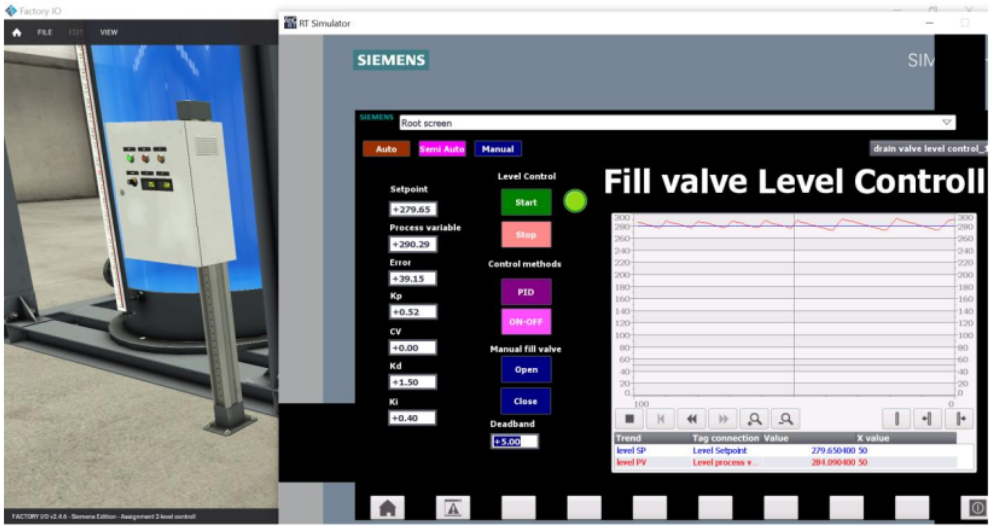


Figure 12

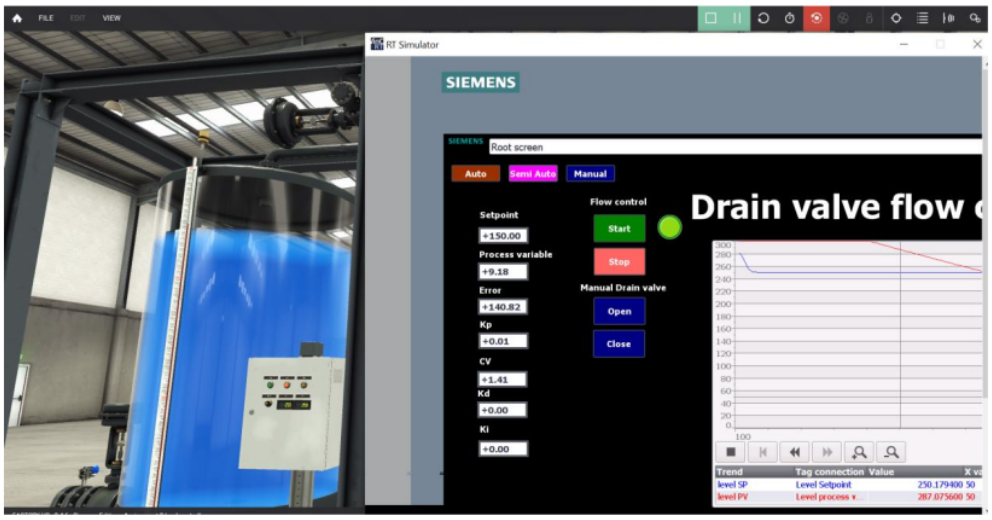
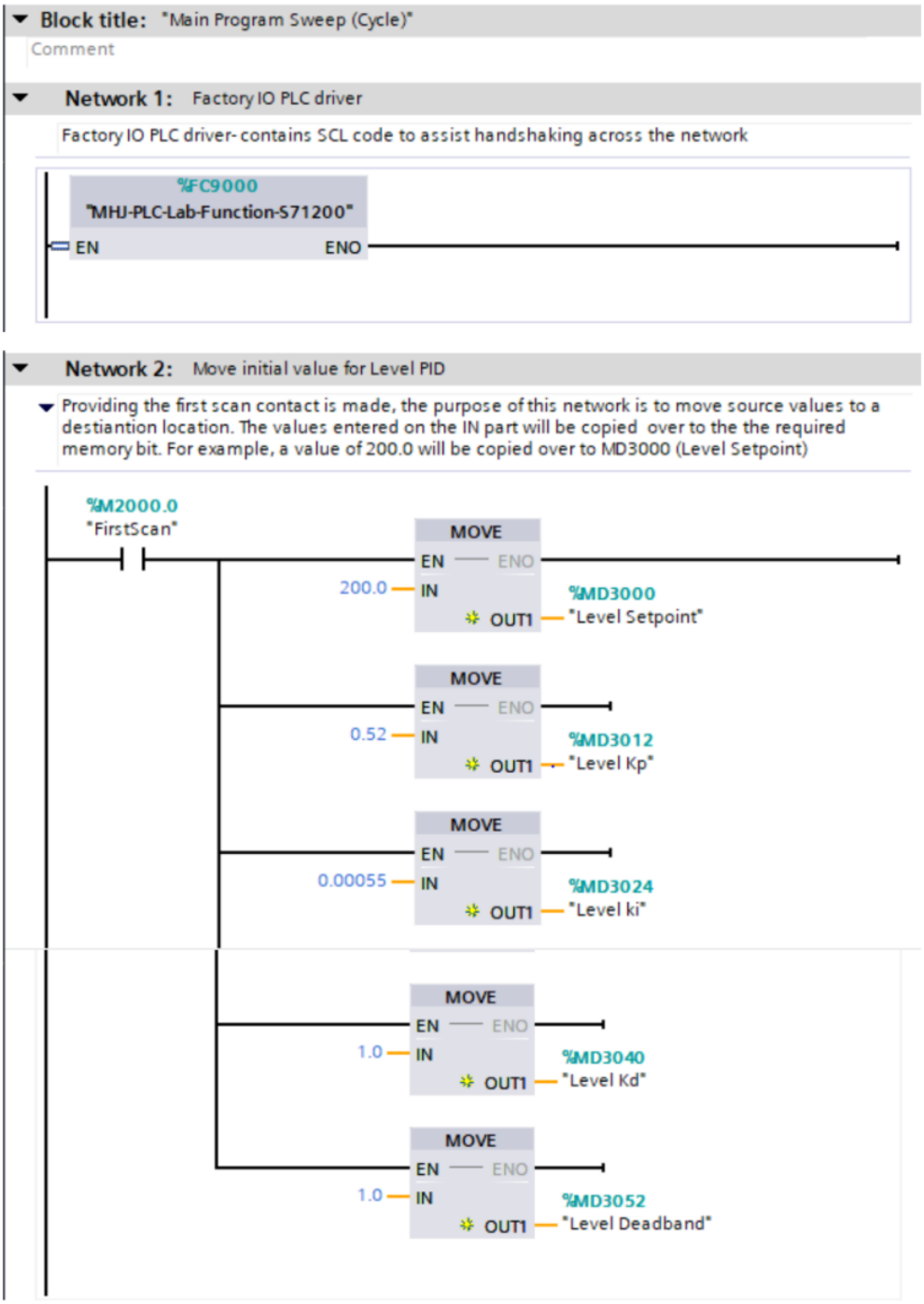


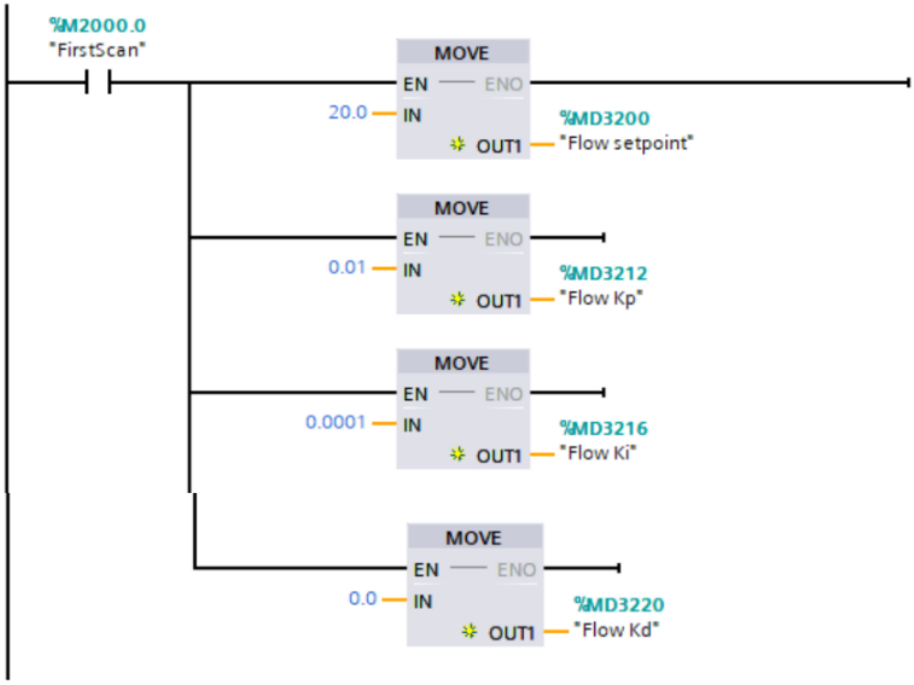
Figure 13

Programme main network



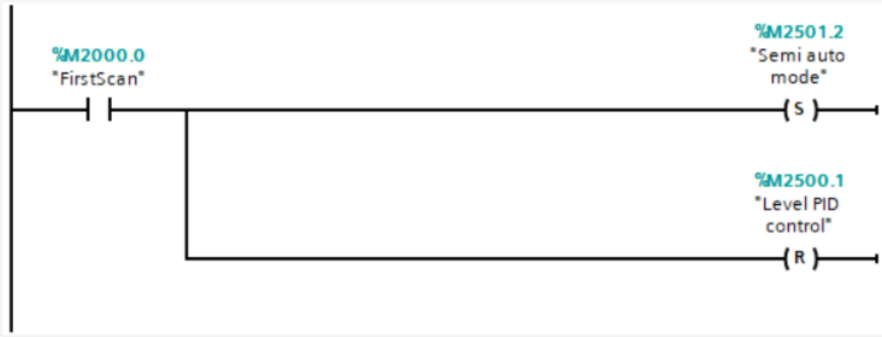
Network 3: Move initial value for flow PID

Providing the first scan contact is made, the purpose of this network is to move source values to a destiantion location. The values entered on the IN part will be copied over to the the required memory bit. For example, a value of 20.0 will be copied over to MD3200 (Flow Setpoint)



Network 4: Set semi automatic mode

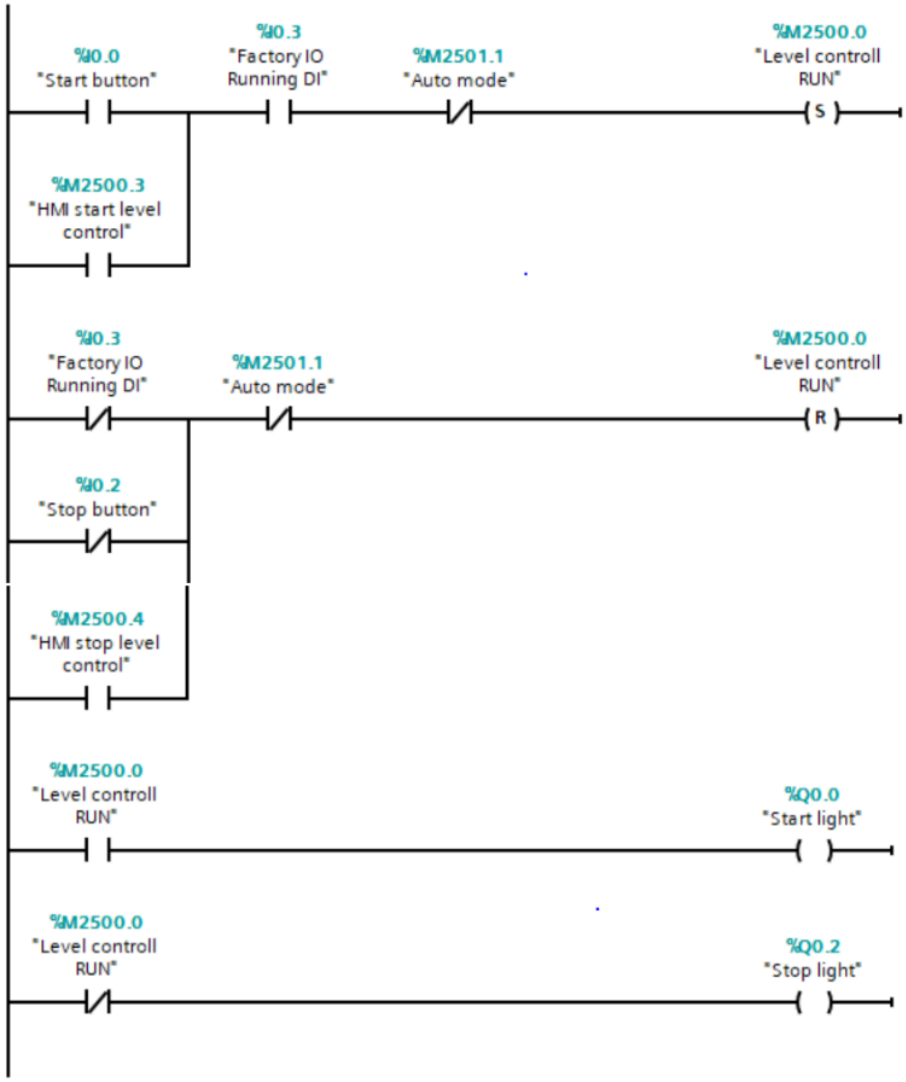
The purpose of this network is to set the automatic mode on the HMI screen. When semi auto is pressed on the HMI screen it will set the bit for semi auto mode. This network also resets the bit for level PID control if other control method (ON/OFF) is selected or if the engineer presses manual mo...



Level control network

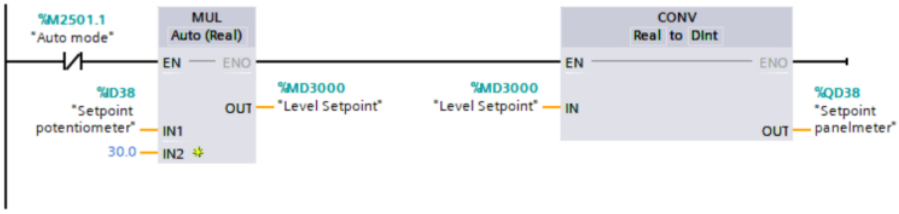
Network 1: Level ON-OFF control

The purpose of this network is to set the bit for level controll providing the start button is pressed, factory IO is running and auto mode is off. Alternatively to the start button, the HMI start level controll can be used.
Level control will be reset when stop button is pressed on the HMI
If level controll run is on, then start light Q0.0 will come on
If level controll run is off, then stop light Q0.1 will come on



Network 2: Scale level setpoint and display

This network works by scaling analogue input for the level setpoint from volts to millimeters by multiplying by 30 so that 10V=300cm.
The level setpoint will then be converted from real to Dint for the setpoint panelmeter

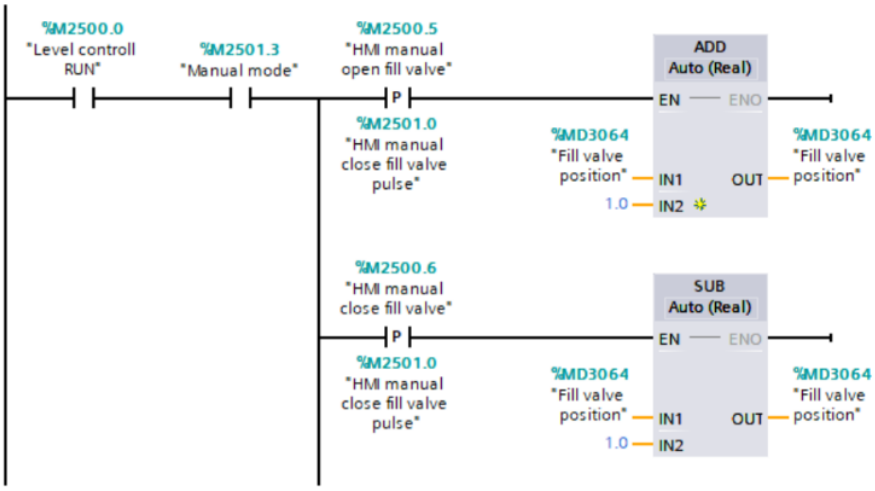


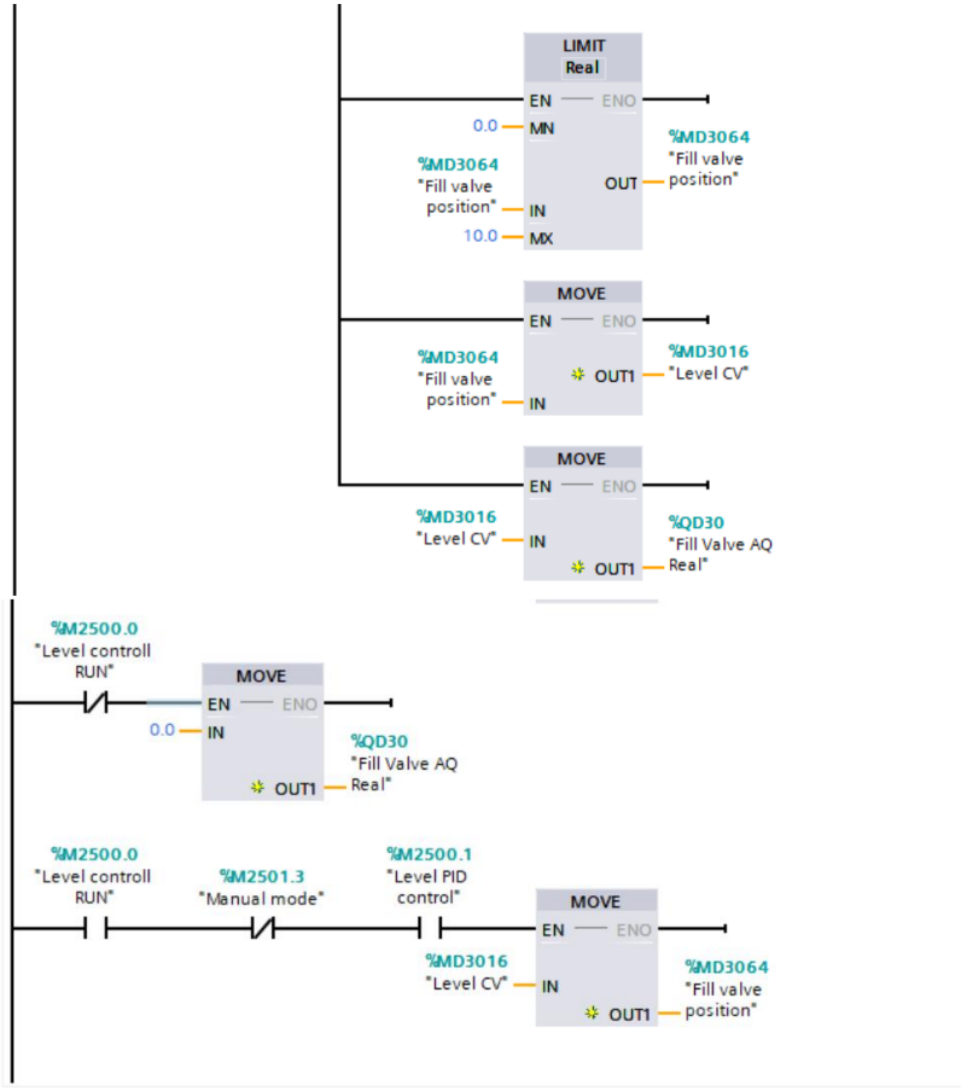
Network 3: Scale level process variable and display

Tells you how much water is in the tank (0-300mm) converting volts from analogue sensor. multiplying ID30 by 30 to get MD3004, converting MD3004 in to a double int to get output QD42

Network 4: Manual fill valve control

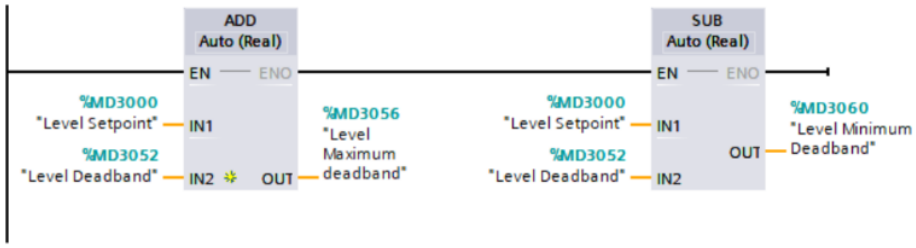
This network Uses a pulse from HMI manual open fill valve to give the fill valve position. That fill valve position is then copied straight over to MD3016(level CV)
The limit function is used to ensure that the fill valve position is greater than 0 and less than 10





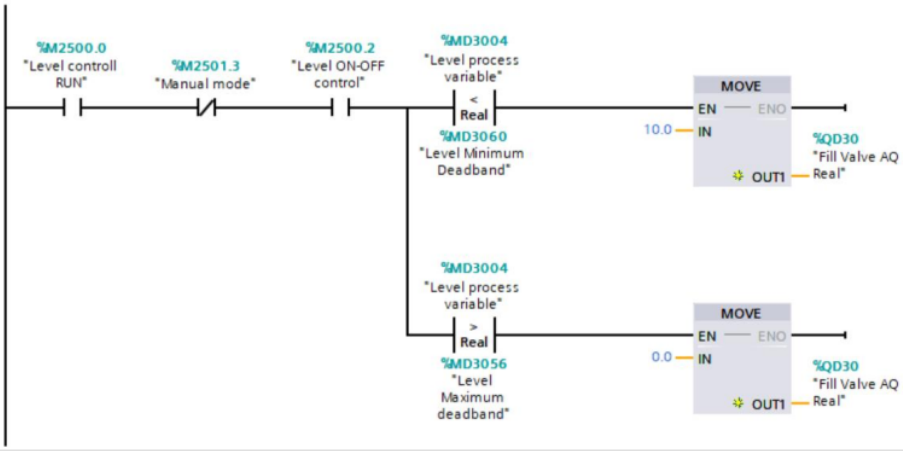
Network 5: ON-OFF control deadband

This network works by adding the level setpoint(MD3000) and level deadband(MD3052) to form level maximum deadband(MD3056) using the ADD function.
To from the level minimum deadband(MD3060), the level setpoint(MD3000) and level deadband(MD3052) are subtracted from eachother using the SUB function.



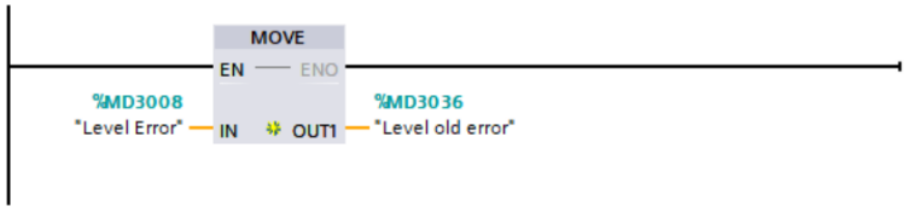
Network 6: Level ON-OFF control with deadband

If the level process variable(MD3004) is less than Level minimum deadband(MD3060) then a value of 10 will be copied over to Fill valve AQ real(QD30)
If the level process variable(MD3004) is greater than Level maximum deadband(MD3056) then a value of 0 will be copied over to Fill valve AQ real(QD30)



Network 7: Store the old error

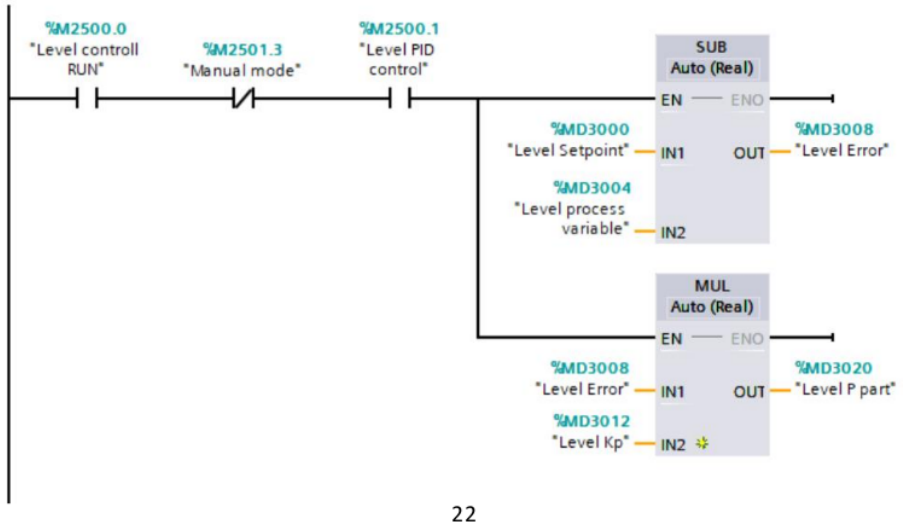
This network copies the value of level error (MD3008) over to level old error(MD3036)



Network 8: Level PID control- calculate error and P part

This network subtracts the level porcess variable(MD3004) from the level setpoint(MD3000) to determine the level error(MD3008). This will only work if level controll and PID control run are on and manual mode is off.

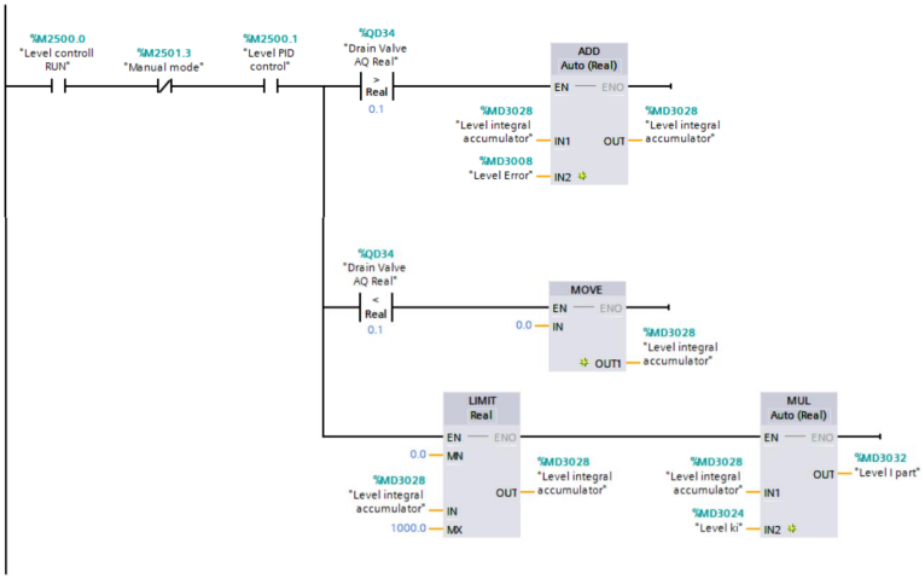
This network also multiply's level error(MD3008) by level Kp(MD3012) to creat the level P part (MD 3020)



Network 9: Level PID control-calculate the I part

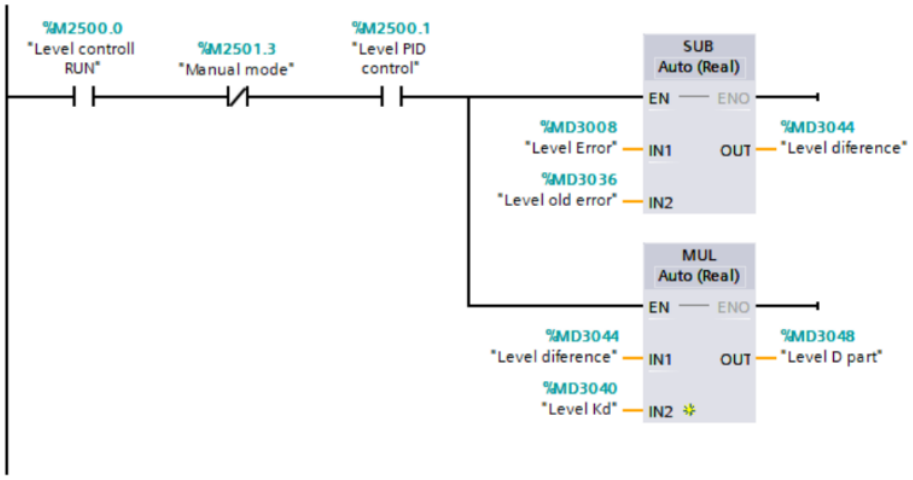
When level controll is running and level PID controll is on, if the drain valve AQ real is greater than 0.1, then it will ADD MD3008(level error) to MD3028(Level integral accumulator). However, if the drain valve is less than 0.1 (meaning there is no flow) then a value of zero will be copied to MD3028-therefore switching I off if theres no flow.

switches I off if this theres no flow
Providing the level integral accumulator is between 0-1000, then MD3028 will be multiplied by MD3024 to create Level I part (MD3032)



Network 10: Level PID control-calculate D part

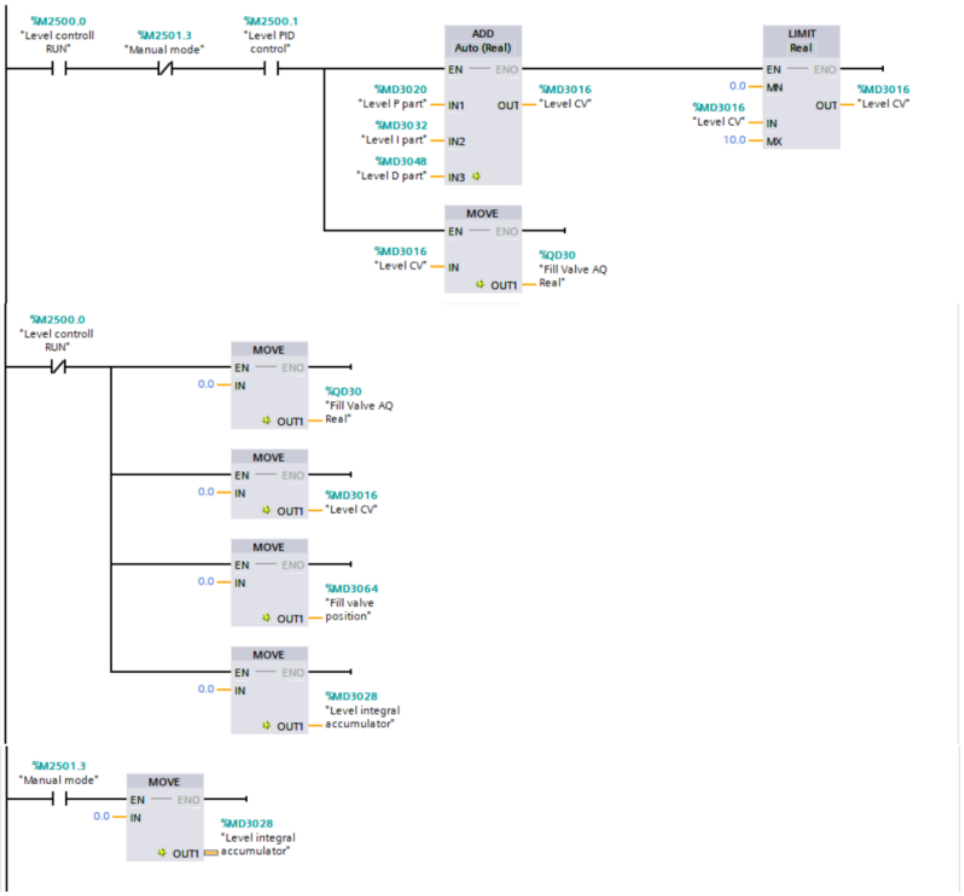
calculates the difference between the old error and the new error to produce MD3044(level difference). Multiplies level difference by level Kd to get MD3048(Level D part). D is added to imporve the performance of I



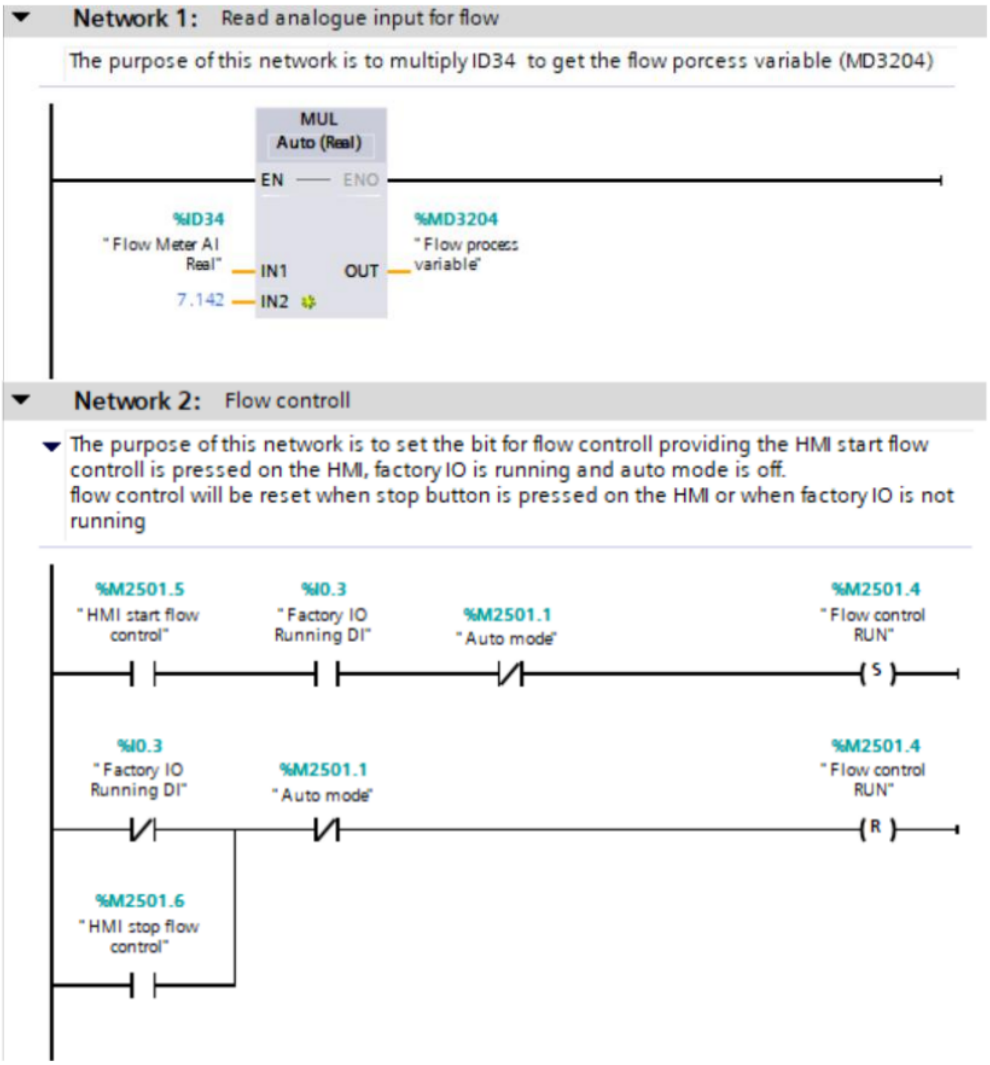
Network 11: Level PID controll- Calculate CV

This network works by adding MD3020,MD3032 and MD3048 together to creat MD3016(Level CV). The limit function is then used to ensure level CV value is gretaer than 0 and less than 10. The value of Level CV will then be copied over to QD30(Fill valve AQ Real) using the MOVE function. This will only work if the level controll and PID control are on, and Manual mode off.

If level controll is off then values from IN will be copied over to QD30, MD3016, MD3064 and MD3028 using the move function

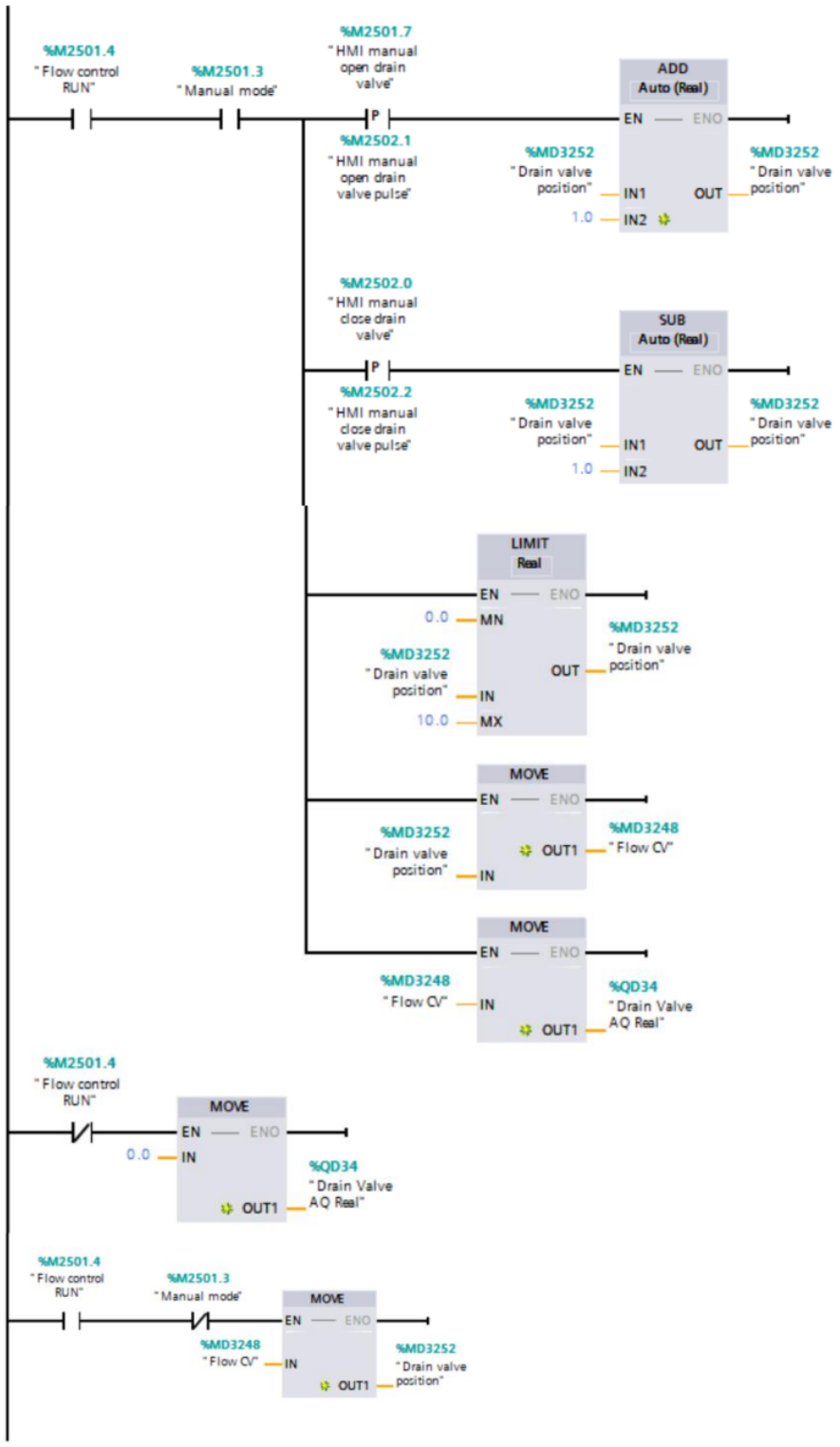


Flow control network



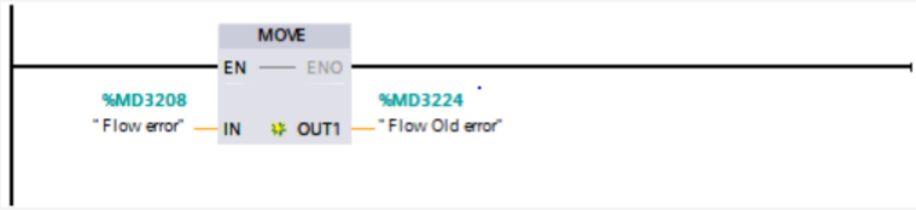
Network 3: Manual drain valve controll

This network Uses a pulse from HMI manual open drain valve to give the drain valve position(MD3252). That fill valve position is then copied straight over to MD3248(flow CV)
The limit function is used to ensure that the drain valve position is greater than 0 and less than 10



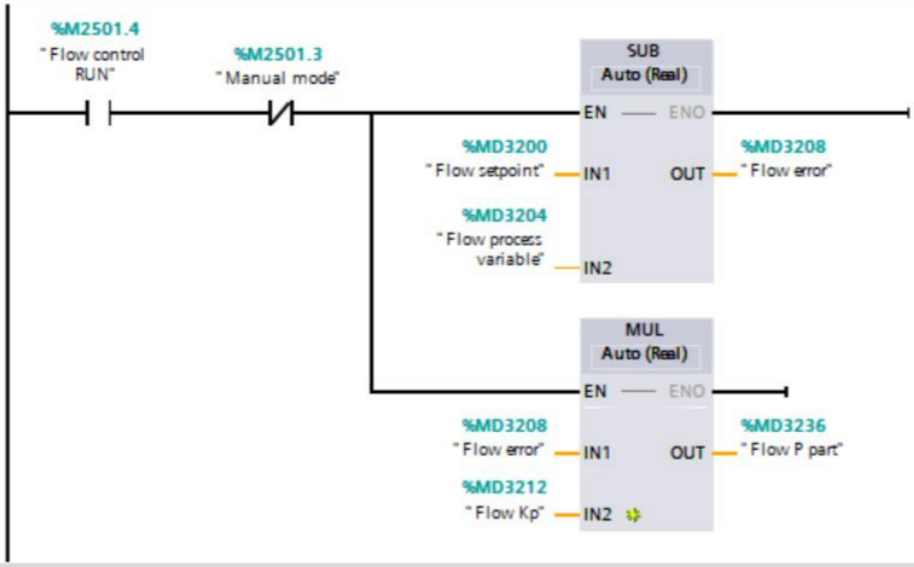
Network 4: Store the old error to calculate D for PID controll

This network moves the flow error(MD3208) to flow old error(MD3224)



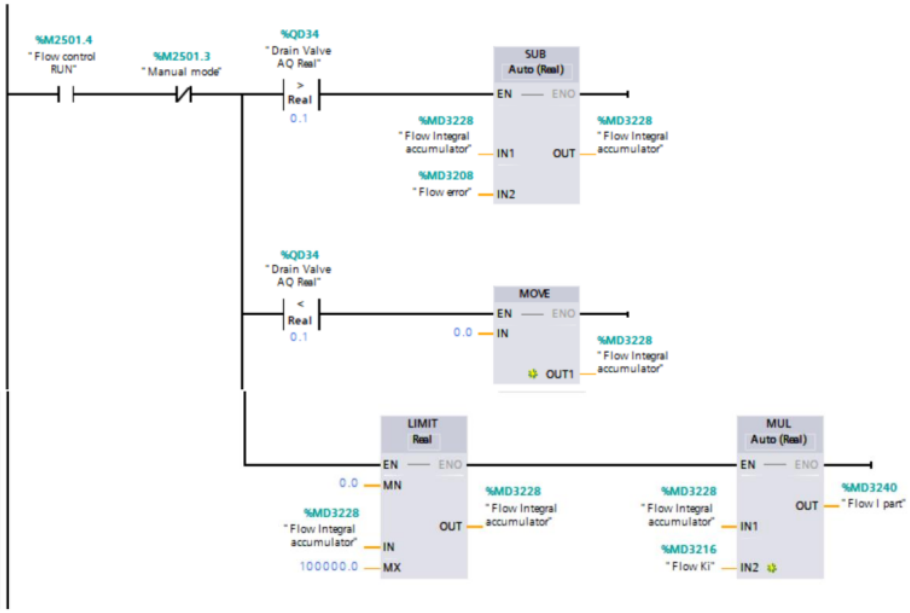
Network 5: Flow PID controll-calculate P part

Providing flow controll is on, the flow process variable(MD3204) will be subtracted from the flow setpoint(MD3200) to determine the flow error (MD3208). The flow error will hten be multiplied by flow Kp to dtermine the flow P part(MD3236)



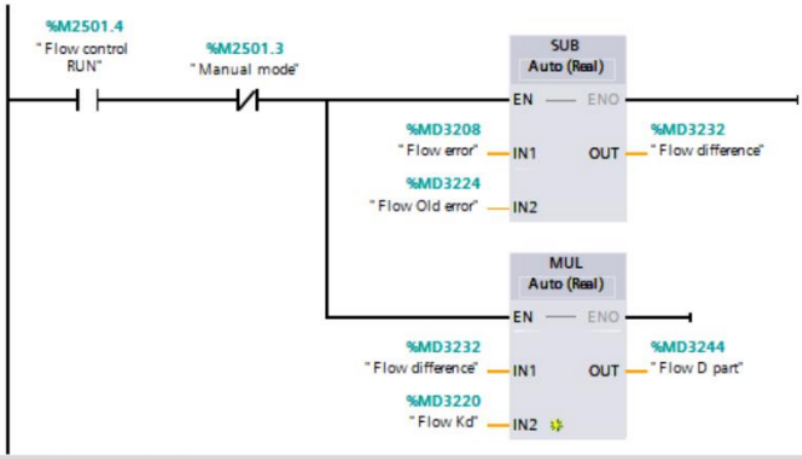
Network 6: Flow PID controll-calculate I part

this network calculates the I part of the PID controll. Fistly the flow error (MD3208) is subtracted from the flow integral accumulator providing you are not in manual mode and flow controll is running. The flow integral accumulator is multiplied by the flow Ki in order to determine the flow I part



Network 7: Flow PID controll-calculate D part

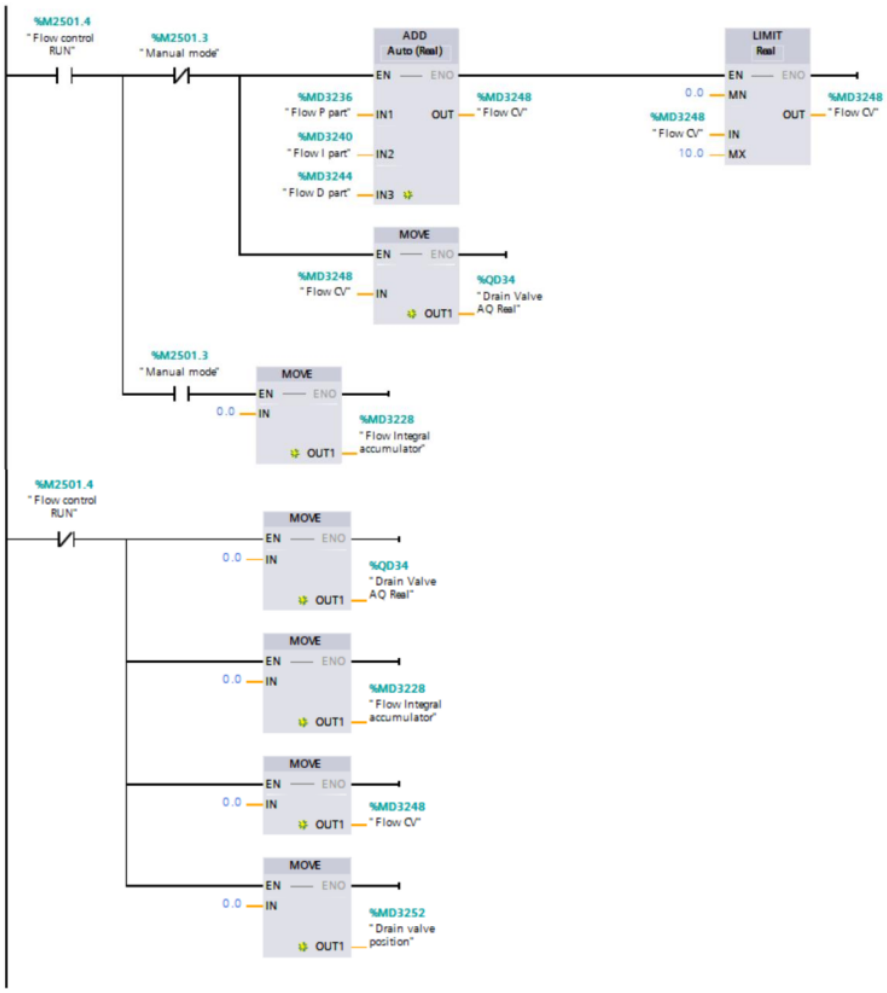
This network subtracts flow old error(MD3224) from flow error(MD3208) to dtermine flow difference(MD3232). The flow difference is then multiplied by flow Kd(MD3220) in order to calculate the D part(MD3244)



Network 8: Calculate CV

This network works by adding MD3236,MD3240 and MD3244 together to creat MD3248(flow CV). The limit function is then used to ensure flow CV value is gretaer than 0 and less than 10. The value of flow CV will then be copied over to QD34(drain valve AQ Real) using the MOVE function. This will only work if the flow controll is running, PID control is on, and Manual mode off.

If flow controll is off then values from IN will be copied over to QD34, MD3228, MD3248 and MD3258 using the move function



PLC tags

FactoryIO - Tank Level Control - assignment 2 ▸ PLC180-factoryIO - Tank Level Controller

PLC tags

	Name	Tag table	Data type	Address
1	Start button	Factory IO	Bool	%I0.0
2	Reset button	Factory IO	Bool	%I0.1
3	Stop button	Factory IO	Bool	%I0.2
4	Factory IO Running DI	Factory IO	Bool	%I0.3
5	Level Meter AI Real	Factory IO	Real	%ID30
6	Flow Meter AI Real	Factory IO	Real	%ID34
7	Setpoint potentiometer	Factory IO	Real	%ID38
8	Start light	Factory IO	Bool	%Q0.0
9	Reset light	Factory IO	Bool	%Q0.1
10	Stop light	Factory IO	Bool	%Q0.2
11	Fill Valve AQ Real	Factory IO	Real	%QD30
12	Drain Valve AQ Real	Factory IO	Real	%QD34
13	Setpoint panelmeter	Factory IO	DInt	%QD38
14	Process variable panel meter	Factory IO	DInt	%QD42
15	System_Byte	Standard-Variablen...	Byte	%MB2000
16	FirstScan	Standard-Variablen...	Bool	%M2000.0
17	DiagStatusUpdate	Standard-Variablen...	Bool	%M2000.1
18	AlwaysTRUE	Standard-Variablen...	Bool	%M2000.2
19	AlwaysFALSE	Standard-Variablen...	Bool	%M2000.3
20	Clock_Byte	Standard-Variablen...	Byte	%MB2001
21	Clock_10Hz	Standard-Variablen...	Bool	%M2001.0
22	Clock_5Hz	Standard-Variablen...	Bool	%M2001.1
23	Clock_2.5Hz	Standard-Variablen...	Bool	%M2001.2
24	Clock_2Hz	Standard-Variablen...	Bool	%M2001.3
25	Clock_1.25Hz	Standard-Variablen...	Bool	%M2001.4
26	Clock_1Hz	Standard-Variablen...	Bool	%M2001.5
27	Clock_0.625Hz	Standard-Variablen...	Bool	%M2001.6
28	Clock_0.5Hz	Standard-Variablen...	Bool	%M2001.7
29	Level controll RUN	My tags	Bool	%M2500.0
30	Level PID control	My tags	Bool	%M2500.1
31	Level ON-OFF control	My tags	Bool	%M2500.2
32	HMI start level control	My tags	Bool	%M2500.3
33	HMI stop level control	My tags	Bool	%M2500.4
34	HMI manual open fill valve	My tags	Bool	%M2500.5
35	HMI manual close fill valve	My tags	Bool	%M2500.6
36	HMI manual open fill valve pulse	My tags	Bool	%M2500.7
37	HMI manual close fill valve pulse	My tags	Bool	%M2501.0
38	Auto mode	My tags	Bool	%M2501.1
39	Semi auto mode	My tags	Bool	%M2501.2
40	Manual mode	My tags	Bool	%M2501.3
41	Flow control RUN	My tags	Bool	%M2501.4
42	HMI start flow control	My tags	Bool	%M2501.5
43	HMI stop flow control	My tags	Bool	%M2501.6
44	HMI manual open drain valve	My tags	Bool	%M2501.7

45	HMI manual close drain valve	My tags	Bool	%M2502.0
46	HMI manual open drain valve p..	My tags	Bool	%M2502.1
47	HMI manual close drain valve p..	My tags	Bool	%M2502.2
48	Level Setpoint	My tags	Real	%MD3000
49	Level process variable	My tags	Real	%MD3004
50	Level Error	My tags	Real	%MD3008
51	Level Kp	My tags	Real	%MD3012
52	Level CV	My tags	Real	%MD3016
53	Level P part	My tags	Real	%MD3020
54	Level ki	My tags	Real	%MD3024
55	Level integral accumulator	My tags	Real	%MD3028
56	Level I part	My tags	Real	%MD3032
57	Level old error	My tags	Real	%MD3036
58	Level Kd	My tags	Real	%MD3040
59	Level diference	My tags	Real	%MD3044
60	Level D part	My tags	Real	%MD3048
61	Level Deadband	My tags	Real	%MD3052
62	Level Maximum deadband	My tags	Real	%MD3056
63	Level Minimum Deadband	My tags	Real	%MD3060
64	Fill valve position	My tags	Real	%MD3064
65	Flow setpoint	My tags	Real	%MD3200
66	Flow process variable	My tags	Real	%MD3204
67	Flow error	My tags	Real	%MD3208
68	Flow Kp	My tags	Real	%MD3212
69	Flow Ki	My tags	Real	%MD3216
70	Flow Kd	My tags	Real	%MD3220
71	Flow Old error	My tags	Real	%MD3224
72	Flow Integral accumulator	My tags	Real	%MD3228
73	Flow difference	My tags	Real	%MD3232
74	Flow P part	My tags	Real	%MD3236
75	Flow I part	My tags	Real	%MD3240
76	Flow D part	My tags	Real	%MD3244
77	Flow CV	My tags	Real	%MD3248
78	Drain valve position	My tags	Real	%MD3252

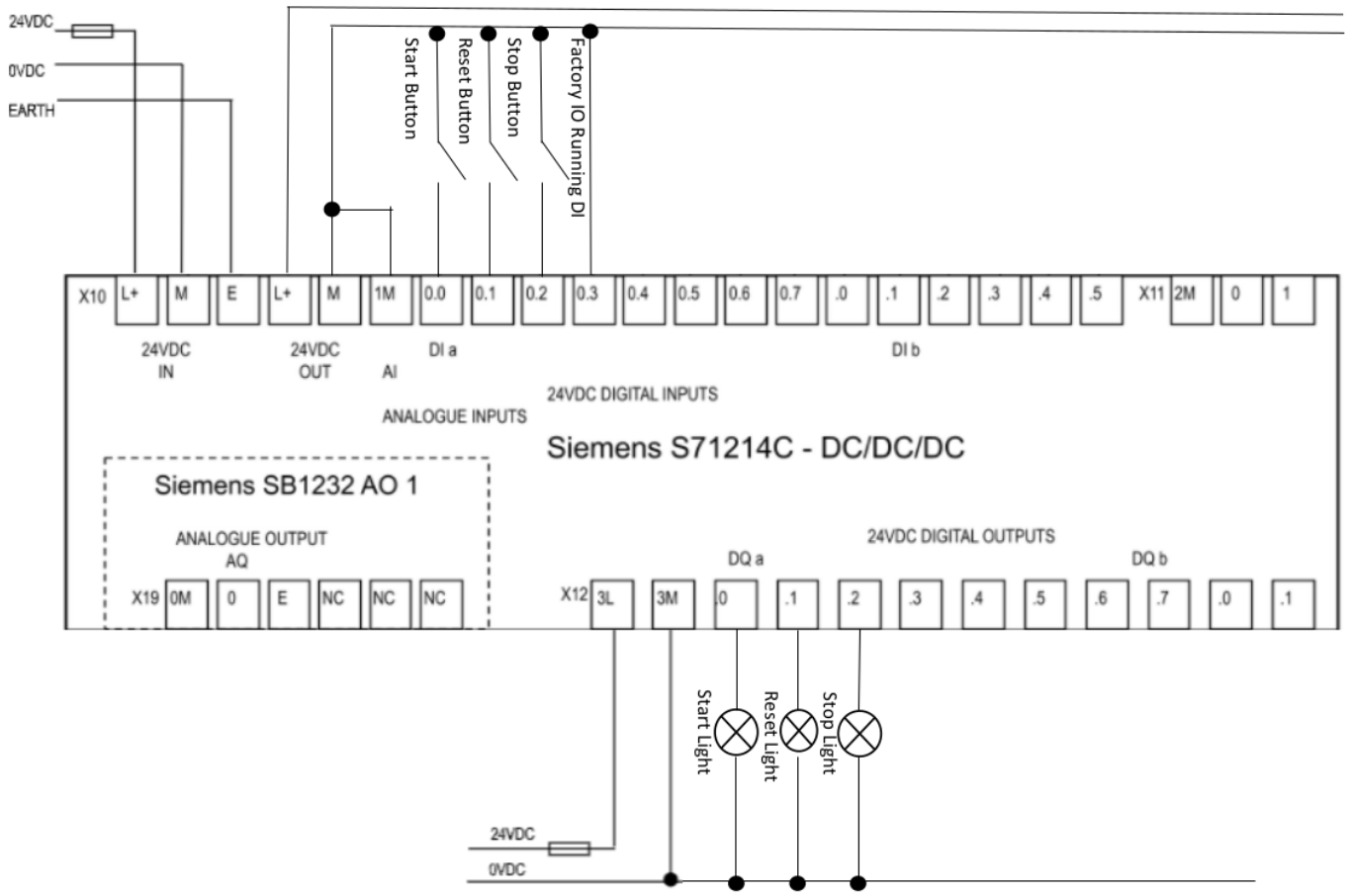


Figure 14- digital IO drawing

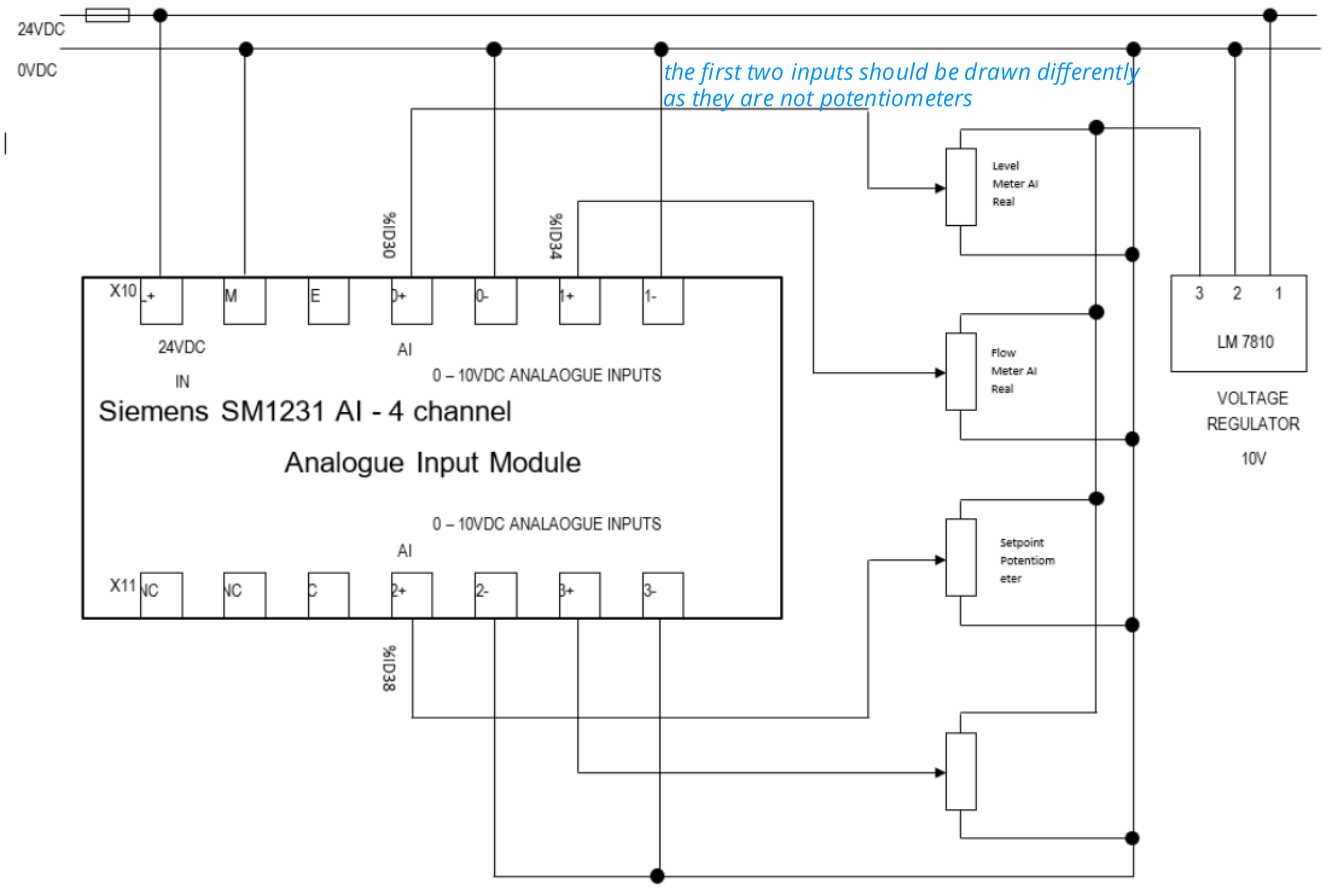
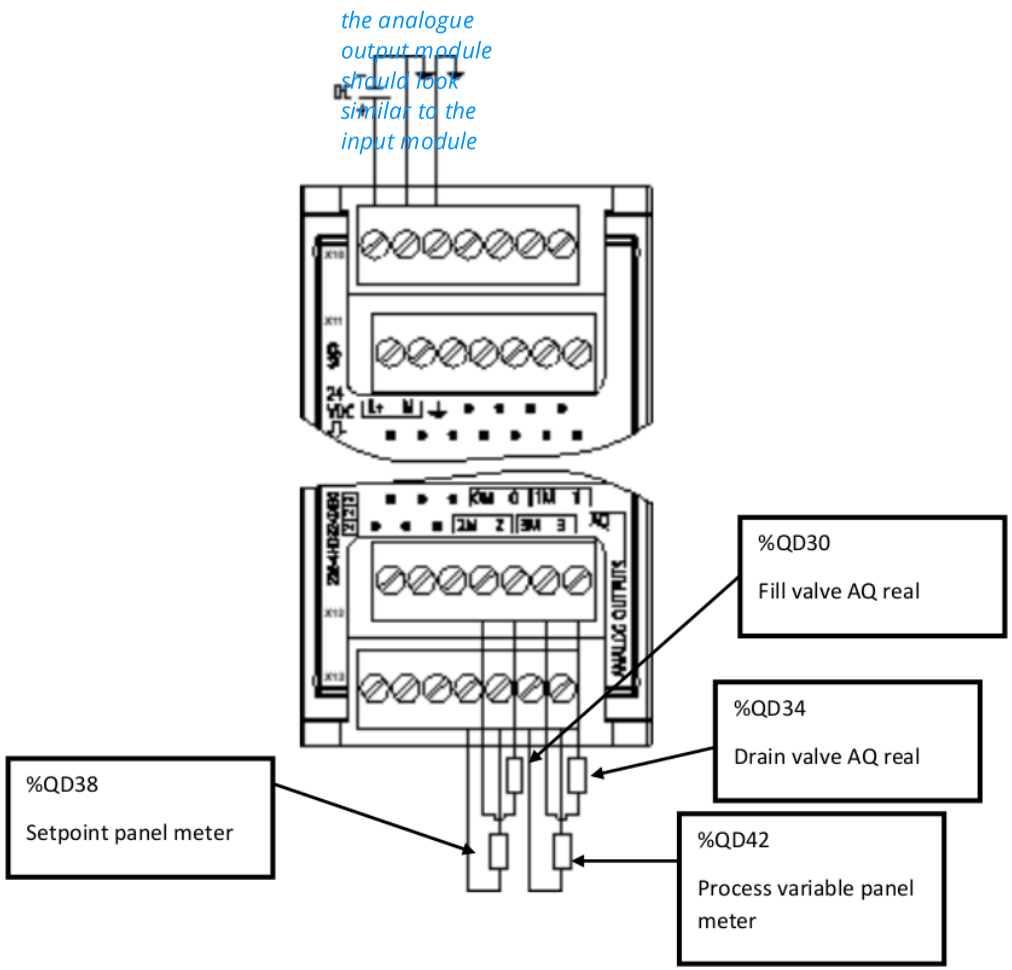


Figure 15-analogue input module drawing


the analogue output module should look similar to the input module
%QD30
Fill valve AQ real
%QD34
Drain valve AQ real
%QD38
Setpoint panel meter
%QD42
Process variable panel meter

1605160

GRADEMARK REPORT

FINAL GRADE

70/100

GENERAL COMMENTS

Instructor

PAGE 1

PAGE 2

PAGE 3

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

Text Comment. Useful information

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

Correctly cited

Citation method conforms to Harvard standard, well done.

PAGE 4

QM

Improper Citation

Improper citation:
Improperly cited material. Please use the link below to find links to information regarding specific citation styles: http://www.plagiarism.org/plag_article_citation_styles.html

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

PAGE 5

Text Comment. Good information

PAGE 6

PAGE 7

PAGE 8

Text Comment. the open and close buttons should have incremented and decremented the valve position

Text Comment. What about full auto mode?

PAGE 9

Text Comment. what about auto mode?

PAGE 10

Text Comment. not auto mode

Text Comment. good observation

QM

Correctly cited

Citation method conforms to Harvard standard, well done.

Text Comment. these are good conclusions and recommendations

PAGE 11

Text Comment. A good range of source correctly listed

PAGE 12

PAGE 13

PAGE 14

Text Comment. which mode are you in?

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

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

Text Comment. the first two inputs should be drawn differently

Text Comment. as they are not potentiometers

PAGE 33

Text Comment. the analogue output module should look similar to the input module

GRADING FORM: NEW CSG-RUBRIC

STEFFAN DAVIES

70

1. ENGAGEMENT WITH LITERATURE

Engagement with a wide range of literature, including sources retrieved independently. Selection of relevant and credible sources. Generally sound referencing, with no/very few inaccuracies or inconsistencies. (12/20)

12

2. KNOWLEDGE AND UNDERSTANDING

Excellent knowledge and understanding of the basic underlying concepts and principles of the subject. (15/20)

15

3. COGNITIVE AND INTELLECTUAL

Sound evaluation of the underlying concepts and principles. Can interpret qualitative and quantitative data accurately. Ability to devise arguments using evidence to make mostly appropriate and valid judgements. A good testing table but you could have done more tests. Very good assessment of user needs and conclusions and recommendations. (13/20)

13

4. PRACTICAL APPLICATION

An advanced application of a range of taught, basic methods, materials, tools and/or techniques. The context of the application is well considered, and insightful. Excellent evidence of different approaches to problem-solving in particular contexts and proposes appropriate solutions. A well designed program though no auto mode included. (15/20)

15

5. TRANSFERABLE SKILLS

Work is coherent, very fluent and is presented professionally. Can work effectively independently and/or as part of a team, with an excellent contribution to group activities. Demonstrates excellent skills for employment requiring the exercise of some personal responsibility with an appetite for further development. (15/20)

15