

HOMEWORK5

REACTOR

จัดทำโดย

นายธนกฤต	จาดทอง	63070501208
นายลัทธิพล	ปิยะสุข	63070501216
นางสาวฉันท์สินี	เมืองหนู	63070501221

INC251

ภาคการศึกษา 2/2564

อ.สมชัย ตีรรัตนจารุ

Homework 5

Write the program for homework5. Then, achieve the project name as: HW5_group Axx and report. Please send to email: inc251.2a@gmail.com

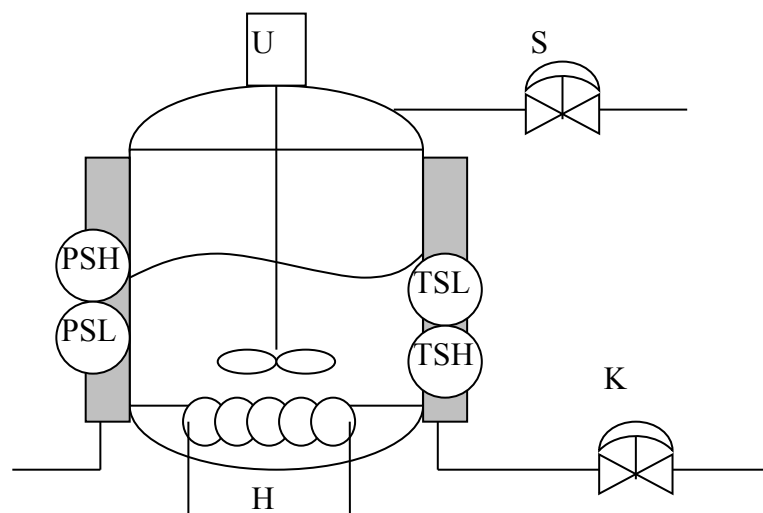
4.1 Reactor (S7 Graph/Ladder/SCL)

Write the IEC standard program according to the job descriptions as state below.

Write a stage diagram for Reactor. Then, write the Program (S7-Graph/Ladder/SCL) and HMI (WinCC) and OP73 for this application. (You must create “Start”, “Reset” buttons and “PSL”, “PSH”, “TSL”, “TSH” buttons on WINCC screen.)

Instructions

Problem



Chemical reaction will be controlled inside the reactor. The reactance will be filled by the others batching control system. Standalone PLC system will control timing, pressure and temperature for chemical reaction.

Reactor control sequence will be start by press start push button for 5 second, after that at this state the alarm signal will be turn on and off with 0.5 Hz for 30 second. Then the alarm signal will be holding on (**normal state**) and the circulating (U) will be turn on for 3 minutes.

During this 3 minutes, pressure and temperature will be regulated in the acceptable range. If there are too high conditions in pressure or temperature for more than 3 second, system will be in **warning state**. And cooling water will be feed to reactor by control valve (K) until the control system turning back to normal condition.

If there are too low conditions in pressure or temperature for more than 3 second, system will be in **warning state**. And heater (H) will be turn on until the control system turning back to normal condition.

Under the warning state, if control system does not turn back to normal condition within 10 second the **alarming state** will be activated. And after entering the **alarming state**, if control system does not turn back to normal condition within 20 seconds the emergency shut down will be start and set control system into the **safety state**.

If the pressure and temperature high condition are activated at the same time or the emergency button is pressed. System will be shut down and go to **safety state**.

If control system stay at **alarming state** and the reset push button is pressed, the control system will be set to **warning state**. And will go to **normal state** if the cause of irregularity is gone and have another press at reset push button.

After the setting time period is elapsed, if there are not in warning or alarming state, system will be set to **initial state**, waiting to working on the next batch.

Remark

Safety state: S is on, H is off and Alarm signal at 0.25 Hz (On 2s, Off 2s)

Alarming state: Alarm signal at 1 Hz (On 0.5s, Off 0.5s)

Warning state: Alarm signal at 0.5 Hz (On 1s, Off 1s)

Normal state: Alarm signal holding on

Initial state: Alarm signal holding off

Pressure too low condition: PSL is activated or analog pressure signal below 25% for more than 3 second.

Pressure too high condition: PSH is activated or analog pressure signal higher than 75% for more than 3 second.

Temperature too low condition: TSL is activated or analog temperature signal below 25% for more than 3 second.

Temperature too high condition: TSH is activated or analog temperature signal higher than 75% for more than 3 second.

Lab instructions

Write the PLC program using the analog signal and safety switch to generate alarming conditions for control system with the correct alarm procedure management.

Assignment submitted before final test.

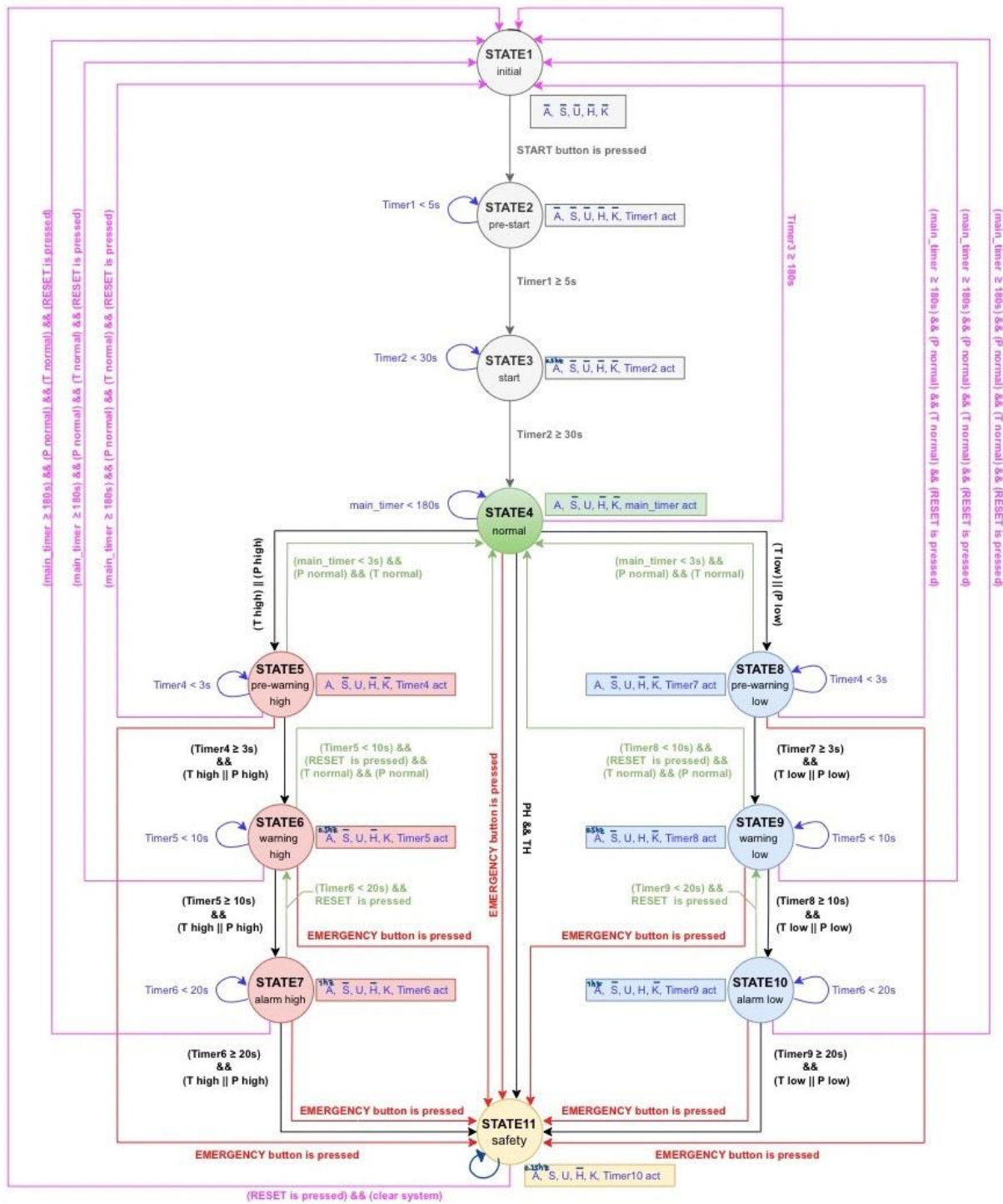
Complete PLC program according to control system described above.

➤ Symbol:

	Statu	Symbol /	Address	Data type	Comment
1		0.25 hz on	M 25.0	BOOL	
2		alarm (A)	Q 124.0	BOOL	
3		alarm 0.25hz	M 24.2	BOOL	
4		alarm 0.5hz	M 24.0	BOOL	
5		alarm 1hz	M 24.1	BOOL	
6		alarm normal	M 24.3	BOOL	
7		ALH alarm view	M 56.7	BOOL	
8		ALL alarm view	M 55.0	BOOL	
9		ciculating (U)	Q 124.1	BOOL	
1		clock 0.5hz	M 100.7	BOOL	
1		clock 1hz	M 100.5	BOOL	
1		COMPLETE RESTART	OB 100	OB 100	Complete Restart
1		cooling water valve (K)	Q 124.3	BOOL	
1		CYC_INT5	OB 35	OB 35	Cyclic Interrupt 5
1		dummy 0.25hz 1	M 25.1	BOOL	
1		dummy 0.25hz 2	M 25.2	BOOL	
1		Emer alarm view	M 56.4	BOOL	
1		EMERGENCY STOP button	I 124.2	BOOL	
1		end 3minutes	M 181.0	BOOL	
2		G7_STD_3	FC 72	FC 72	
2		go_initia	M 181.1	BOOL	
2		heater (H)	Q 124.2	BOOL	
2		main timer	MW 180	INT	
2		memEMER	M 30.2	BOOL	
2		memRESET	M 30.1	BOOL	
2		memSTART	M 30.0	BOOL	
2		once transfer	M 140.0	BOOL	
2		PH_analog	M 150.0	BOOL	
2		PH_hmi	M 80.3	BOOL	
3		PL_analog	M 150.1	BOOL	
3		PL_hmi	M 80.4	BOOL	
3		PRESSURE HIGH button	I 124.3	BOOL	
3		pressure high status	Q 124.5	BOOL	
3		PRESSURE LOW button	I 124.4	BOOL	
3		pressure low status	Q 124.6	BOOL	
3		pressure_down	MD 140	REAL	
3		pressure_limithi	MD 76	REAL	
3		pressure_limitlow	MD 36	REAL	
3		pressure_now	MD 136	REAL	
4		pressure_up	MD 132	REAL	
4		PressureHi	FC 12	FC 12	
4		PressureLow	FC 13	FC 13	
4		PSH alarm view	M 56.0	BOOL	
4		PSL alarm view	M 56.1	BOOL	
4		RESET button	I 124.1	BOOL	
4		reset timer	M 158.0	BOOL	
4		reset_hmi	M 80.1	BOOL	
4		safety valve (S)	Q 124.4	BOOL	
4		Safty alarm view	M 55.1	BOOL	
5		SCALE	FC 105	FC 105	Scaling Values
5		START UP button	I 124.0	BOOL	
5		start_hmi	M 80.0	BOOL	
5		start_timer	M 180.0	BOOL	
5		stop_hmi	M 80.2	BOOL	
5		TEMP HIGH button	I 124.5	BOOL	
5		temp high status	Q 124.7	BOOL	
5		TEMP LOW button	I 124.6	BOOL	
5		temp low status	Q 125.0	BOOL	
5		temp_down	MD 124	REAL	
6		temp_limithi	MD 72	REAL	
6		temp_limitlow	MD 32	REAL	
6		temp_now	MD 128	REAL	
6		temp_up	MD 120	REAL	
6		TempHi	FC 10	FC 10	
6		TempLow	FC 11	FC 11	
6		TH_analog	M 150.2	BOOL	
7		TL_hmi	M 80.6	BOOL	
7		TSH alarm view	M 56.2	BOOL	
7		TSL alarm view	M 56.3	BOOL	
7		VAT_1	VAT 1		
7		WNH alarm view	M 56.5	BOOL	
7		WNL alarm view	M 56.6	BOOL	

ตารางที่ 1: สัญลักษณ์ที่ใช้ในการเขียนโปรแกรมทั้งหมด

➤ State Diagram:

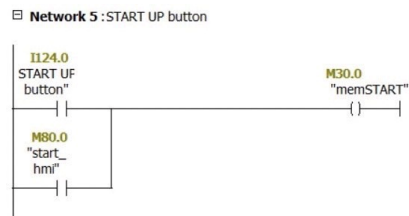


รูปที่ 1: state diagram for reactor

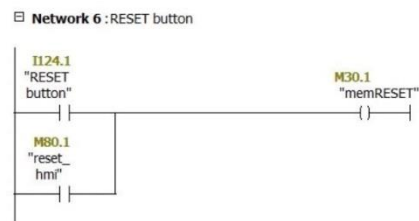
➤ Program:

การเขียนโปรแกรม reactor ทั้งหมด จะเขียนตาม state diagram ที่ได้ออกแบบไว้ นั่นคือ จะใช้การเขียน S7-Graph เป็นหลัก อีกทั้งมีการใช้การเขียน ladder และ SCL ร่วมด้วยในบางส่วน

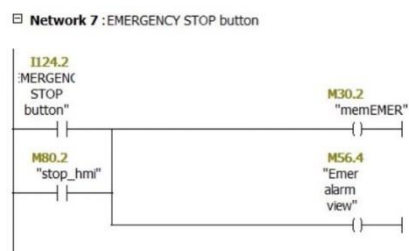
1. เริ่มเขียนโปรแกรมจากการสร้างปุ่มกดที่ต้องการใช้ทั้งหมดไว้ให้ครบโดยการเขียน ladder เพื่อที่จะ ทำให้สามารถเรียกไปใช้เขียนโปรแกรมต่อใน S7-graph ได้ โดยเขียน ladder ได้ดังนี้



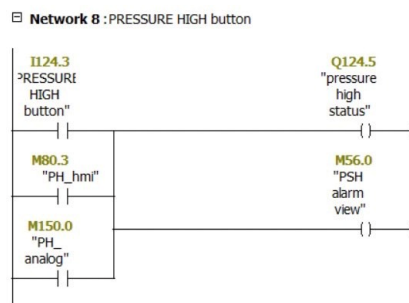
รูปที่ 2: สร้างปุ่ม START UP



รูปที่ 3: สร้างปุ่ม RESET

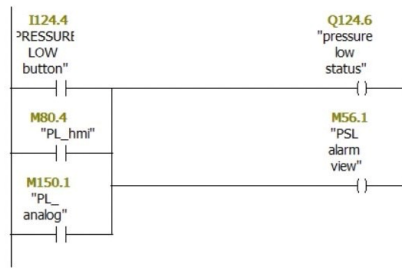


รูปที่ 4: สร้างปุ่ม EMERGENCY STOP



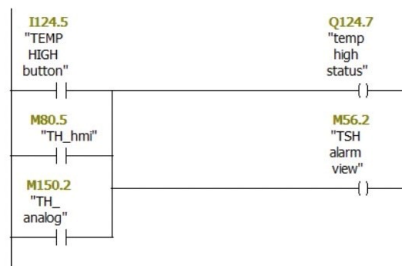
รูปที่ 5: สร้างปุ่ม PRESSURE HIGH (PSH)

Network 9 : PRESSURE LOW button



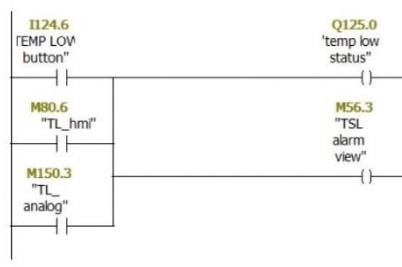
รูปที่ 6: สร้างปุ่ม PRESSURE LOW (PSL)

Network 10 : TEMP HIGH button



รูปที่ 7: สร้างปุ่ม TEMPERATURE HIGH (TSH)

Network 11 : TEMP LOW button



รูปที่ 8: สร้างปุ่ม TEMPERATURE LOW (TSL)

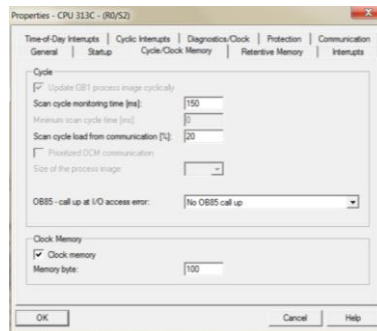
2. เนื่องจากโปรแกรม reactor มีการใช้สัญญาณไฟในการเตือนเมื่อเครื่องอยู่ในสถานะต่าง ๆ เช่น หากเครื่องอยู่ในสถานะ safety สัญญาณไฟเตือนจะกระพริบด้วยความถี่ 0.25 hz หรือถ้าหากอยู่ในสถานะ normal สัญญาณไฟจะติดตลอดเวลาแบบไม่กระพริบ เป็นต้น โดยโปรแกรมทั้งหมดที่ออกแบบจะมีการใช้สัญญาณไฟเตือนแบบกระพริบใน 3 ความถี่ คือ 0.5 hz, 0.25hz และ 1hz อีกทั้งยังคงมีการใช้สัญญาณไฟเตือนแบบปกติที่ไม่กระพริบด้วย ซึ่งเราจะเขียนโปรแกรมในส่วนของการควบคุมสัญญาณไฟโดยใช้ทั้งการเขียนโปรแกรมใน S7-Graph และ ladder ร่วมกัน

การเขียนโปรแกรม S7-Graph ในแต่ละ Step ของการทำงานของโปรแกรมก็จะใช้ไฟเตือนที่ต่างความถี่กัน เช่น จากรูปที่ 9 ชุดคำสั่งใน Step6 ของ S7-Graph เนื่องจากเราต้องการให้ไฟเตือนที่แสดงใน Step นี้ ติดด้วยความถี่ 0.5 hz เราจึงได้มีการสั่งให้ M24.0 (alarm 0.5 hz) ทำงาน ซึ่ง M24.0 นี้ ก็จะเอาไปใช้สั่งให้ไฟติดด้วยความถี่ 0.5 hz ได้ โดยการเขียน ladder ต่อไป

S6 Step6	warning high
S	"alarm 0.5hz"
R	"alarm 1hz"
R	"alarm 0.25hz"
R	"alarm normal"
S	"circulating (U)"
R	"heater (H)"
S	"cooling water valve (K)"
R	"safety valve (S)"
S	"start_timer"

รูปที่ 9: ชุดคำสั่งใน Step6 ของ S7-Graph

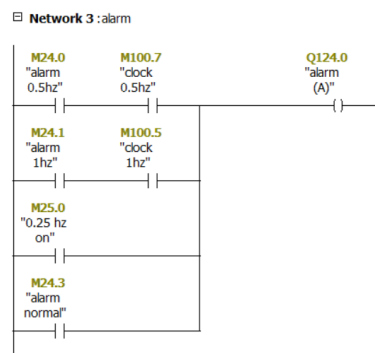
จากที่เขียนโปรแกรม S7-graph ในแต่ละ step ก็จะมีการสั่งให้ไฟเตือนทำงาน (M24.0-M24.3) ซึ่งสัญญาณไฟเตือนในแต่ละ step ก็จะมีเวลาที่ต่างกันตามเงื่อนไขโจทย แล้วเราจะนำ memory เหล่านี้ไปเขียนโปรแกรม ladder เพื่อสั่งให้ไฟติดตามความถี่ต่าง ๆ เริ่มแรกเราจะต้องไปตั้งค่า clock memory ที่เราต้องการใช้เพื่อสั่งให้ไฟกะพริบ โดยสามารถตั้งค่า clock memory ได้ด้วยการไป config. CPU ในที่นี้เราได้ตั้งค่า memory byte ของ clock memory ใน byte ที่ 100 ดังรูปที่ 10



รูปที่ 10: ตั้งค่า memory byte ใน clock memory
(config. Hardware: CPU → clock memory)

Period Duration								
A period duration/frequency is assigned to each bit of the clock memory byte:								
Bit	7	6	5	4	3	2	1	0
Period duration (s)	2	1.6	1	0.8	0.5	0.4	0.2	0.1
Frequency (Hz)	0.5	0.625	1	1.25	2	2.5	5	10

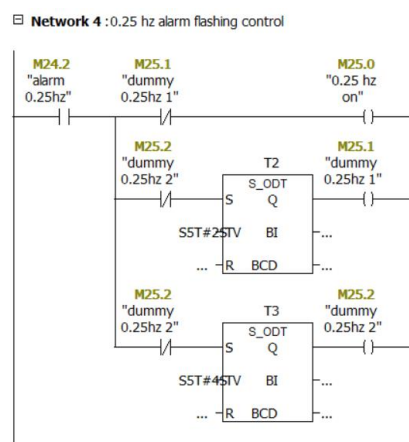
รูปที่ 11: ตั้งค่า memory byte ใน clock memory
(ที่มา: help on CPUs and Special FMs ในโปรแกรม S7-300)



รูปที่ 12: control alarm

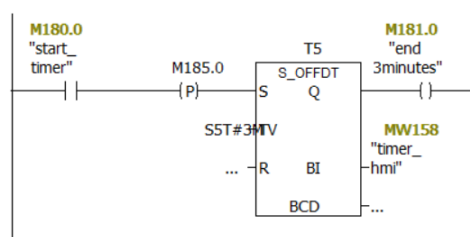
จากรูปที่ 12 เป็นการเขียนโปรแกรมควบคุมไฟเตือนทั้งหมด ซึ่งการสั่งให้ไฟกะพริบที่ความถี่ 0.5 hz และความถี่ 1 hz นั้น เราสามารถที่จะใช้ memory byte ที่เราได้ตั้งค่าไว้ให้เป็น clock memory ในการสั่งงานได้เลย โดยถ้าหากพิจารณาจากรูปที่ 11 จะเห็นได้ว่าถ้าหากต้องการสั่งให้ไฟกะพริบด้วยความถี่ 1 hz นั้น เราจะต้องใช้ bit ที่ 5 ของ memory byte ที่ตั้งค่าไว้ และถ้าหากต้องการสั่งให้ไฟกะพริบด้วยความถี่ 0.5 hz เราจะต้องใช้ bit ที่ 7 ของ memory byte ที่เราได้ตั้งค่าไว้ให้เป็น clock memory ในการสั่งงาน

แต่ถ้าหากต้องการสั่งให้ไฟกะพริบด้วยความถี่ 0.25 hz นั้น เราไม่สามารถที่จะใช้ memory byte ในการสั่งงานแบบก่อนหน้านี้ได้ เราจึงต้องเขียนโปรแกรม ladder สำหรับควบคุมไฟกะพริบด้วยความถี่ 0.25 hz โดยเฉพาะ เพื่อควบคุมการทำงานของไฟกะพริบที่ความถี่ 0.25 hz (ติด 2 วินาที ดับ 2 วินาที) ซึ่งเขียนได้ดังรูปที่ 13

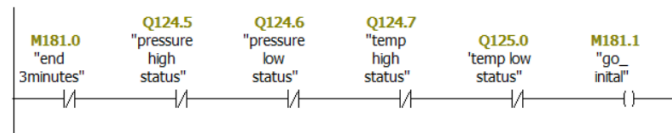


รูปที่ 13: 0.25 hz alarm flashing control

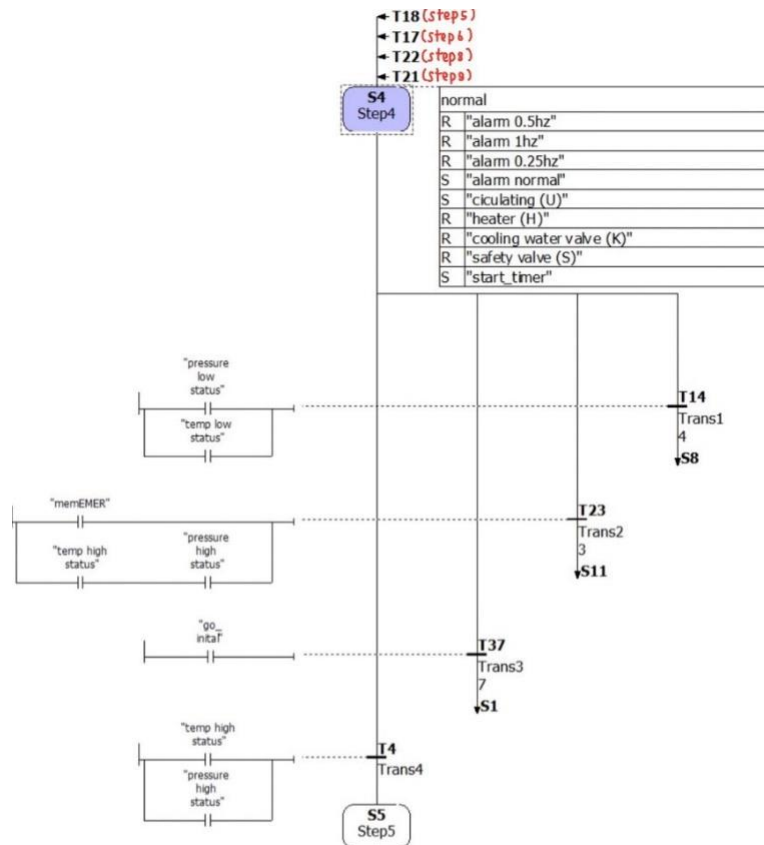
3. ในส่วนของการควบคุมเวลาของการทำงานทั้งหมด เราจะมี main timer ที่ทำหน้าที่จับเวลาเมื่อเครื่องทำงานครบ 3 นาที โดยเงื่อนไขที่เกี่ยวข้องกับ main timer มีดังนี้
 - ถ้าหากอยู่ใน Step 4 (normal) แล้วเครื่องก็ยังคงทำงานปกติจนครบ 3 นาที เครื่องจะกลับสู่ Step1 (initial) โดยอัตโนมัติ ซึ่งเราจะเขียนโปรแกรม ladder โดยสร้าง main timer ขึ้นมาสำหรับจับเวลา และมีการใช้ S7-Graph ในการสั่งให้ start timer ทำงาน ตั้งแต่เริ่มต้นที่เครื่องเข้าสู่ Step4 (normal) เมื่อ main timer นับเวลาจนครบ 3 นาทีแล้ว ถ้าหากเครื่องยังคงอยู่ในสถานะปกติ ก็จะกลับเข้าสู่ Step1 โดยเขียนโปรแกรมได้ดังรูปที่ 14, รูปที่ 15 และรูปที่ 16



รูปที่ 14: main timer



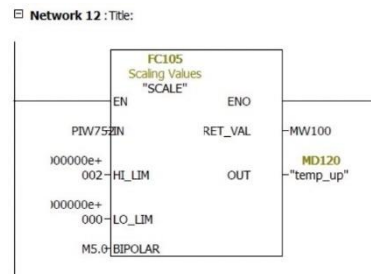
รูปที่ 15: ถ้าเครื่องทำงานครบ 3 นาที และยังคงอยู่ในสถานะปกติ -> go initial (M181.1) ทำงาน



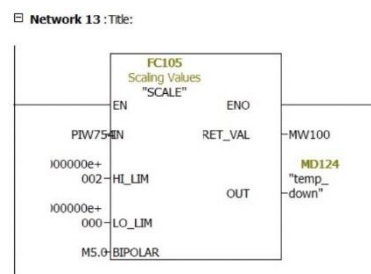
รูปที่ 16: ตัวอย่างการเขียนโปรแกรมใน S7-Graph สำหรับ Step4 (normal)

- ถ้าหากเครื่องยังคงอยู่ในโหมด pre-warning high/low, warning high/low และ alarm high/low กรณีที่เครื่องกลับมาอยู่ในสถานะปกติ และเครื่องทำงานครบ 3 นาทีแล้ว หากเรากดปุ่ม reset จะทำให้กลับสู่ step1 (initial) ได้ เราก็จะใช้การเขียนโปรแกรมแบบเดียวกับเงื่อนไขก่อนหน้านี้ ต่างกันเพียงเงื่อนไขนี้จะต้องการทำการกดปุ่ม reset ด้วย ถึงจะสามารถกลับสู่ Step1 (initial) ได้

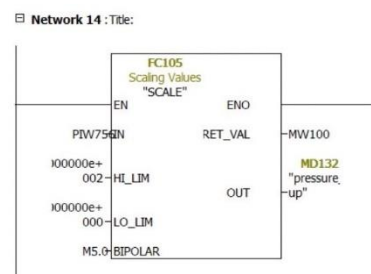
4. การแปลง scale ค่า digital ให้เป็น analog (0-100) โดยจากเงื่อนไขคือ ถ้าหากต่ำกว่า 25% เป็น low , ไม่เกิน 75% เป็น normal และมากกว่า 75% เป็น high ซึ่งการเขียนโปรแกรมเป็นไปดังรูปที่ 17-20



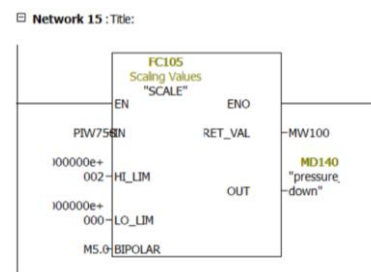
รูปที่ 17: FC1 network12



รูปที่ 18: FC1 network13

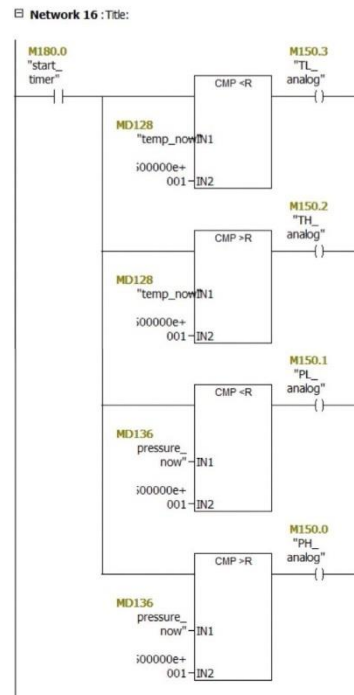


รูปที่ 19: FC1 network14



รูปที่ 20: FC1 network15

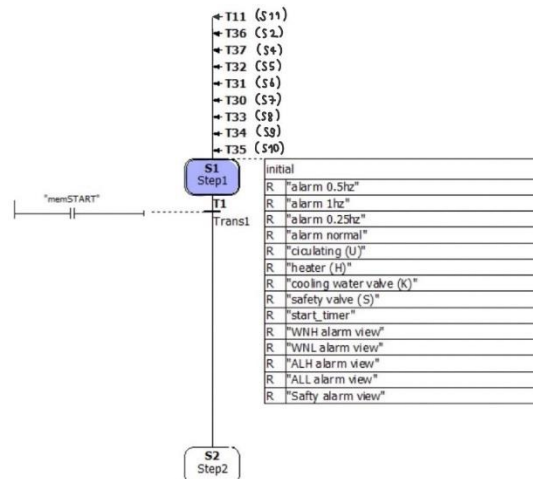
5. การ check เงื่อนไขว่าเครื่องอยู่ในสถานะที่ปกติหรือไม่ ซึ่งการ check เงื่อนไขนี้เกิดขึ้นตั้งแต่ที่เครื่องอยู่ใน Step4 (normal) ซึ่งเป็น Step ที่มีการสั่งให้ main timer เริ่มทำการนับเวลา ตัวอย่างเช่น เมื่อเราสั่งให้ timer ทำงาน มันก็จะทำการ check ว่าตอนนี้ค่าอุณหภูมิ/ความดันปัจจุบันน้อยกว่า 25% หรือไม่ สมมติถ้าหากอุณหภูมิต่ำกว่า 25% ก็ส่งผลให้ TL_analog (M150.3) ทำงาน โดย M150.3 นี้ก็จะไปทำให้ temp low status (Q125.0) ทำงาน



รูปที่ 21: FC1 network16

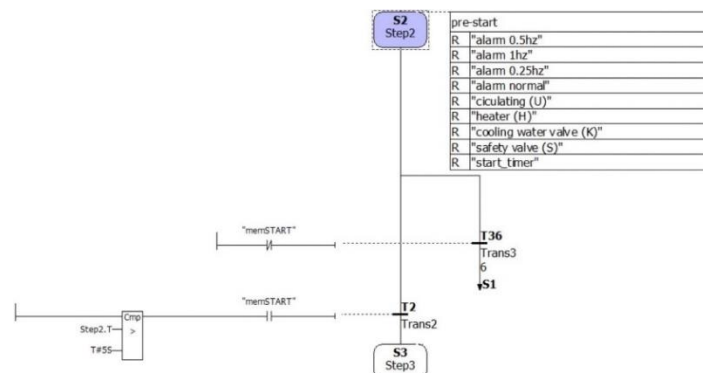
6. การเขียนโปรแกรมหลัก ๆ แล้ว จะเขียนเรียงขั้นตอนตาม state diagram ดังที่ได้ออกไว้ และเขียนไว้ใน S7-Graph โดยมีการใช้ ladder ร่วมกับในบางส่วน ซึ่งมีประเด็นหลัก ๆ ดังที่กล่าวไปข้างแล้ว ข้างต้น ส่วนการเขียนโปรแกรม S7-Graph ทั้งหมด หากแยกออกมาเป็นแต่ละ Step สามารถเขียนได้ดังนี้

- Step1: initial



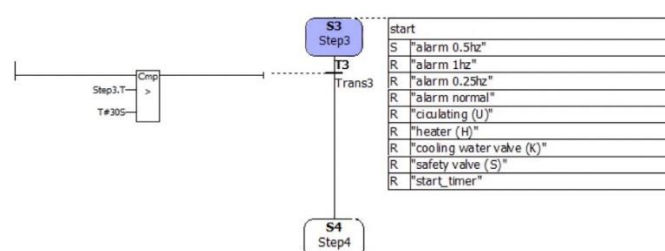
รูปที่ 22: S7-Graph >> Step1 (initial)

- Step2: pre-start



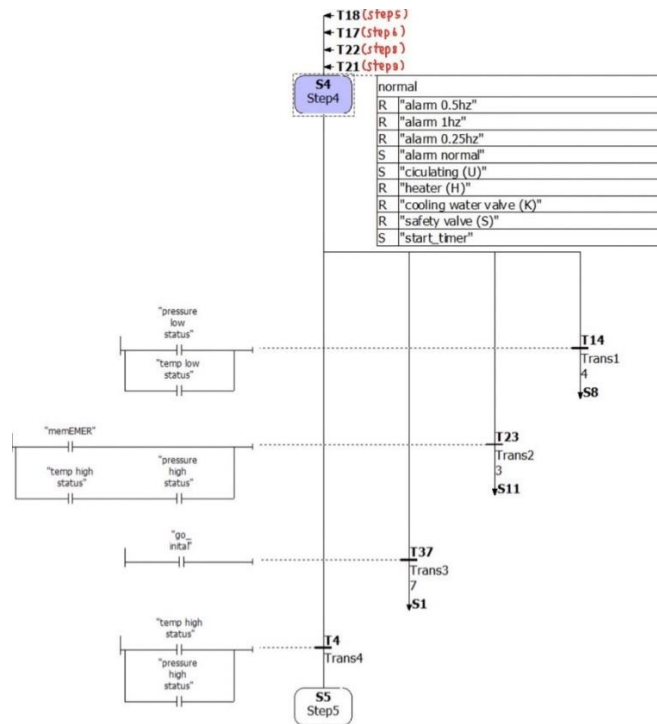
รูปที่ 23: S7-Graph >> Step2 (pre-start)

- Step3: start



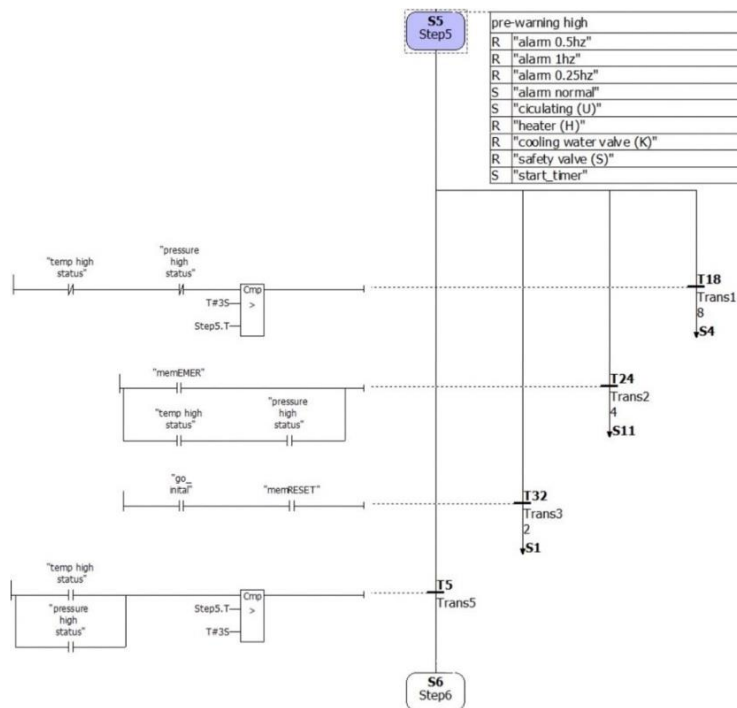
รูปที่ 24: S7-Graph >> Step3 (start)

- Step4: normal



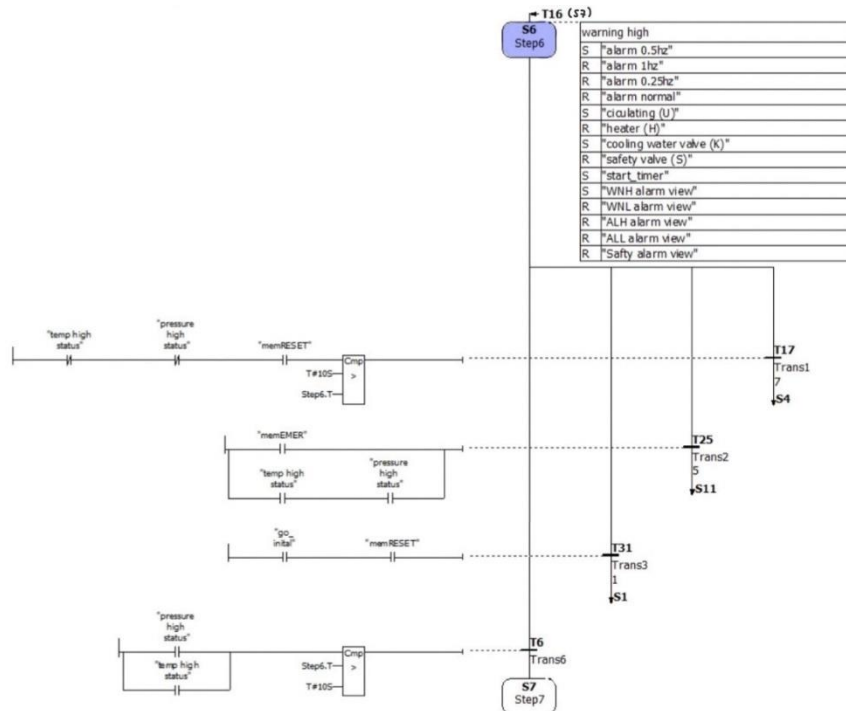
รูปที่ 25: S7-Graph >> Step4 (normal)

- Step5: pre-warning high



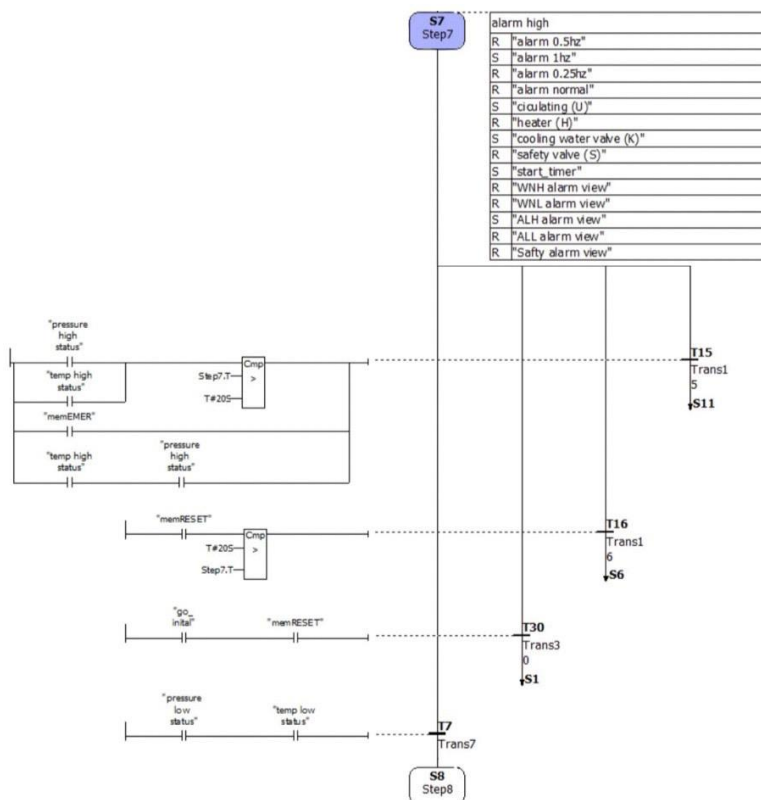
รูปที่ 26: S7-Graph >> Step5 (pre-warning high)

- Step6: warning high



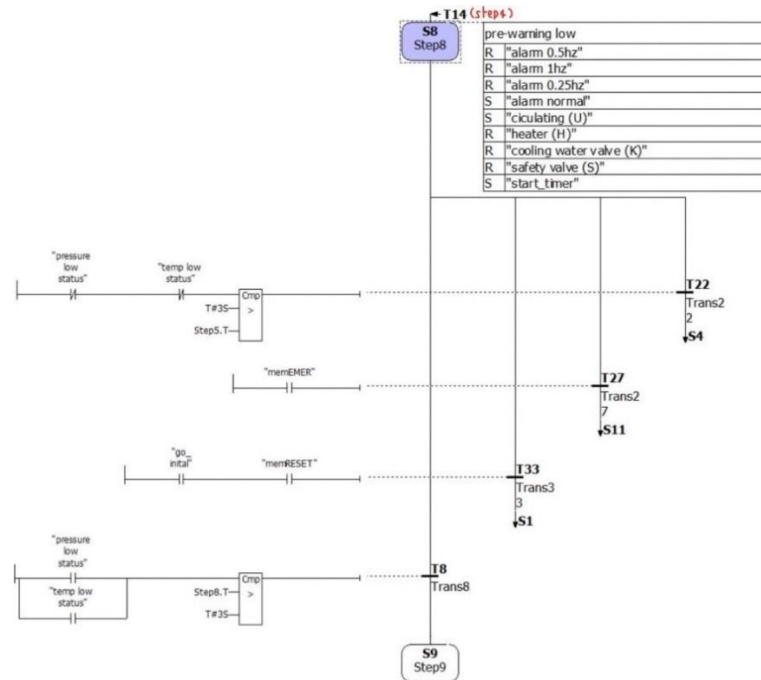
รูปที่ 27: S7-Graph >> Step6 (warning high)

- Step7: alarm high



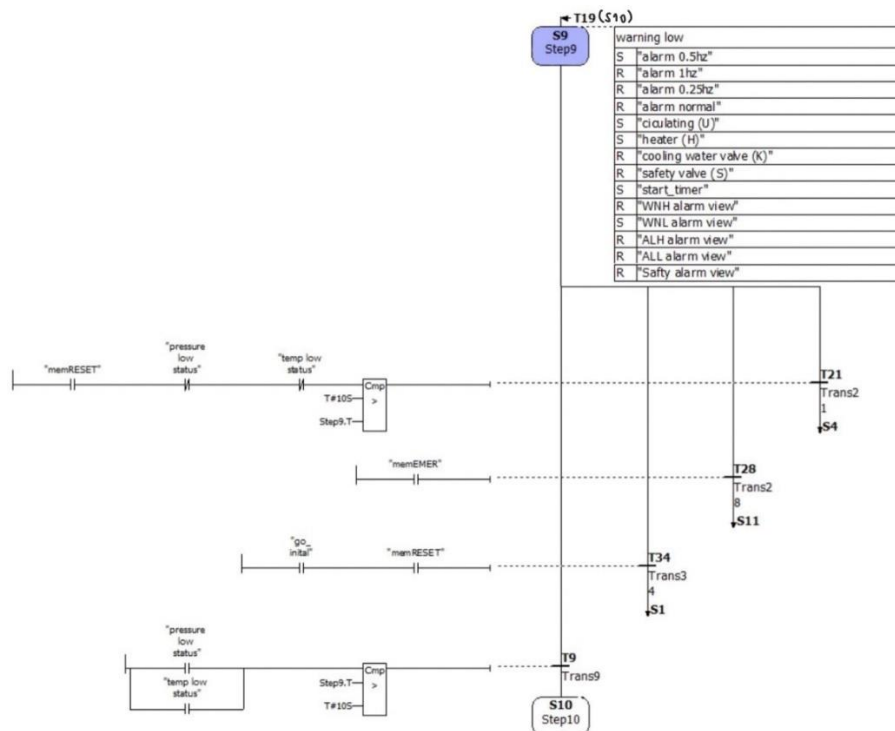
รูปที่ 28: S7-Graph >> Step7 (alarm high)

- Step8: pre-warning low



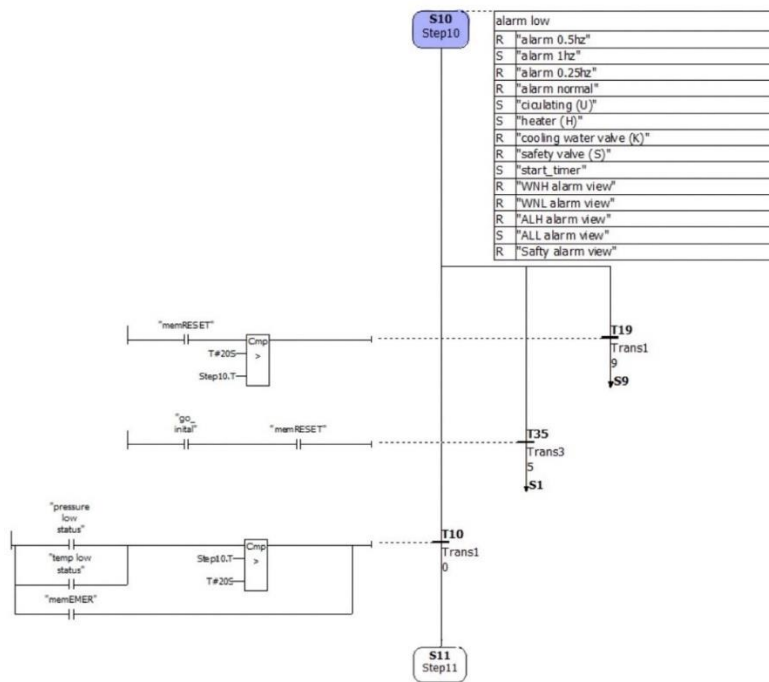
รูปที่ 29: S7-Graph >> Step8 (pre-warning low)

- Step9: warning low



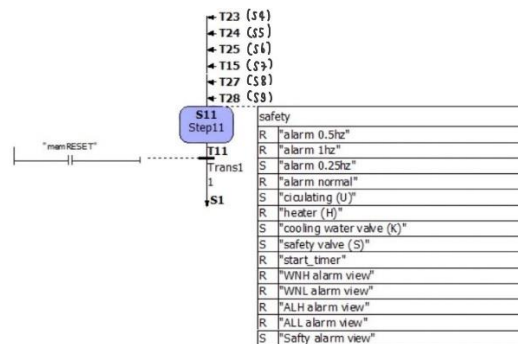
รูปที่ 30: S7-Graph >> Step9 (warning low)

- Step10: alarm low



รูปที่ 31: S7-Graph >> Step10 (alarm low)

- Step11: safety



รูปที่ 32: S7-Graph >> Step11 (safety)

7. SCL code

```
FUNCTION TempHi: REAL  
  
VAR_INPUT  
    temp_down: REAL;  
    temp_now: REAL;  
  
END_VAR  
  
BEGIN  
    TempHi := temp_now - temp_down;  
END_FUNCTION
```

รูปที่ 33: SCL code >> FUNCTION Temperature High

```
FUNCTION TempLow: REAL  
  
VAR_INPUT  
    temp_up: REAL;  
    temp_now: REAL;  
  
END_VAR  
  
BEGIN  
    TempLow := temp_now + temp_up;  
END_FUNCTION
```

รูปที่ 34: SCL code >> FUNCTION Temperature Low

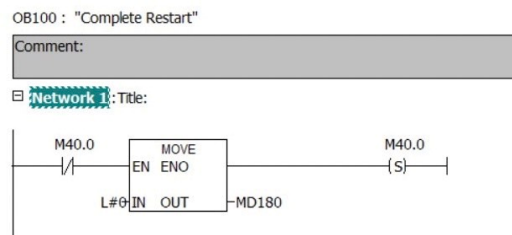
```
FUNCTION PressureHi: REAL  
  
VAR_INPUT  
    pressure_down: REAL;  
    pressure_now: REAL;  
  
END_VAR  
  
BEGIN  
    PressureHi := pressure_now - Pressure_down;  
END_FUNCTION
```

รูปที่ 35: SCL code >> FUNCTION Pressure High

```
FUNCTION PressureLow: REAL  
  
VAR_INPUT  
    pressure_up: REAL;  
    pressure_now: REAL;  
  
END_VAR  
  
BEGIN  
    PressureLow := pressure_now + Pressure_up;  
END_FUNCTION
```

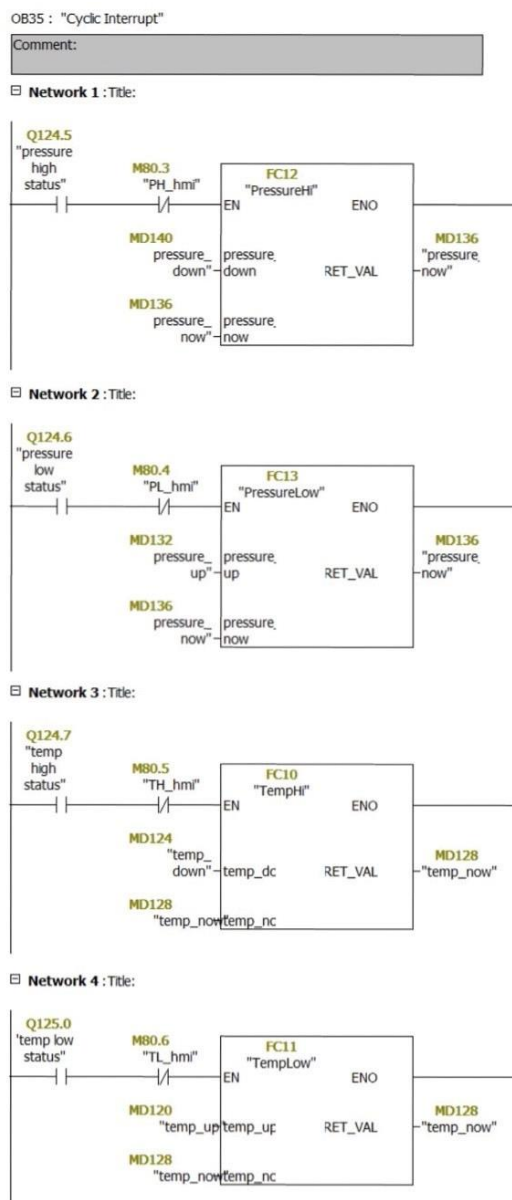
รูปที่ 36: SCL code >> FUNCTION Pressure Low

8. การเขียนโปรแกรมบน OB100 นี้ เขียนไว้ใช้สำหรับตอนรันและทดสอบโปรแกรมก่อนที่โปรแกรมจะเสร็จสมบูรณ์ เพื่อที่จะทำให้ PLC สามารถเริ่มทำงานใหม่ตั้งแต่ Step1 ได้ ถ้าหาก CPU ถูก stop ลง



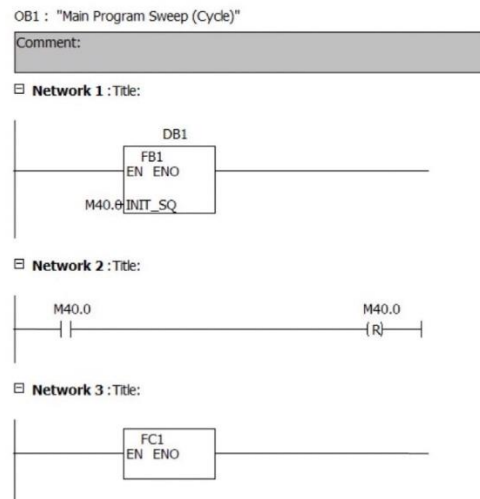
รูปที่ 36: OB100

9. การเขียนโปรแกรมบน OB35 (จาก SCL code)



รูปที่ 37: OB35

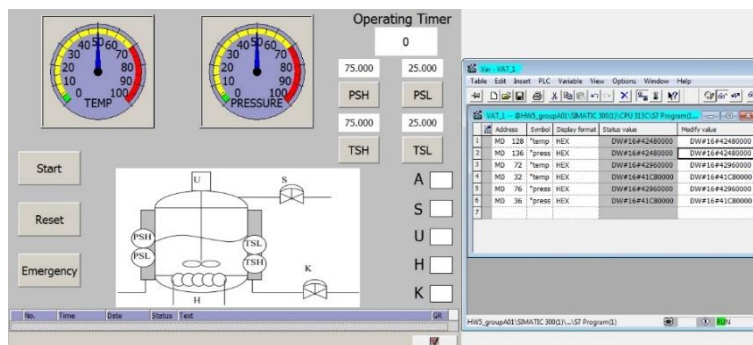
10. การทำงานทั้งหมดของโปรแกรมควบคุม reactor นี้ เราได้เขียนไว้ใน FC1, FB1 (มีการใช้งานควบคู่กับ DB1) โดยเราได้เรียกใช้งาน FC1 และ FB1 ที่สร้างขึ้นนี้ผ่าน OB1 ดังรูปที่ 38 และนอกจากนี้ก็ยังมีการเขียนโปรแกรมไว้ใน OB100 , SCL และ OB35 ดังที่กล่าวไปแล้วก่อนหน้านี้



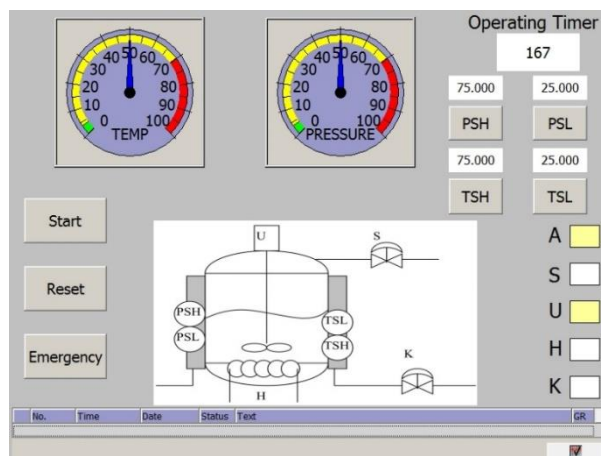
รูปที่ 38: OB1

➤ HMI Screen(WinCC & OP73)

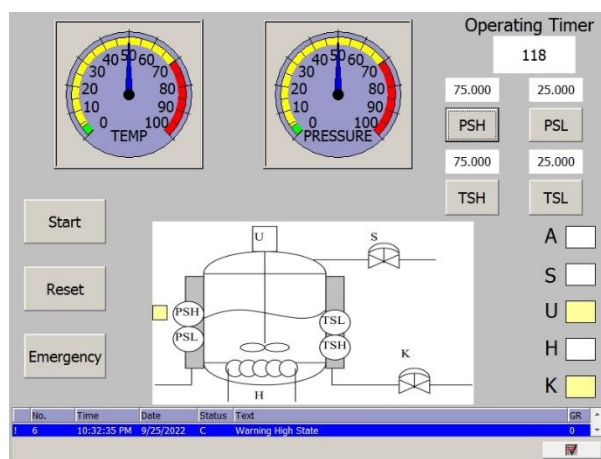
● Screen1:



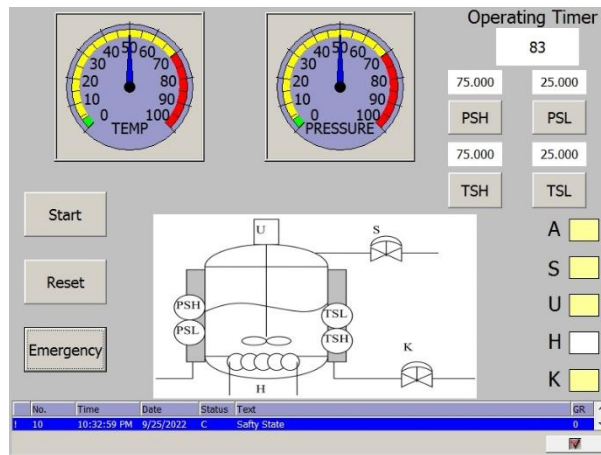
ใส่ค่าเริ่มต้นใน initial state



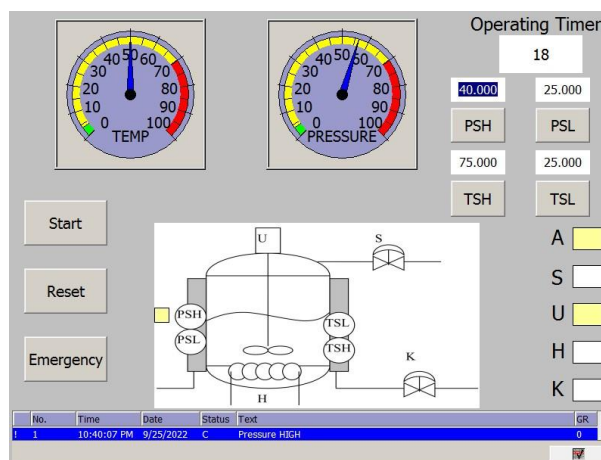
Normal state



เมื่อกดปุ่ม PSH และรอนจนเข้าสู่ Warning High State

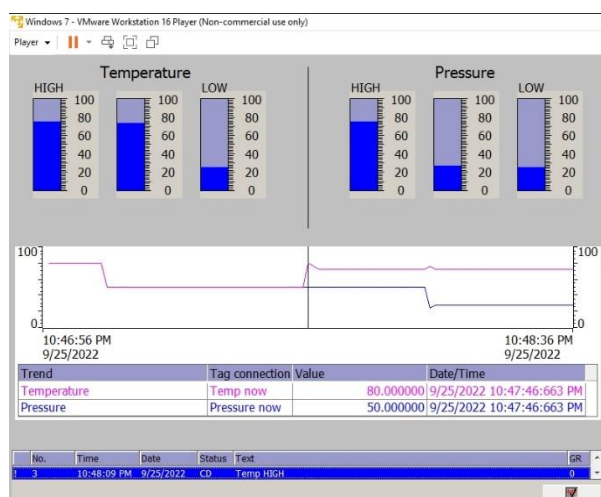


เมื่อกดปุ่ม Emergency ทำให้เข้าสู่ Safety state

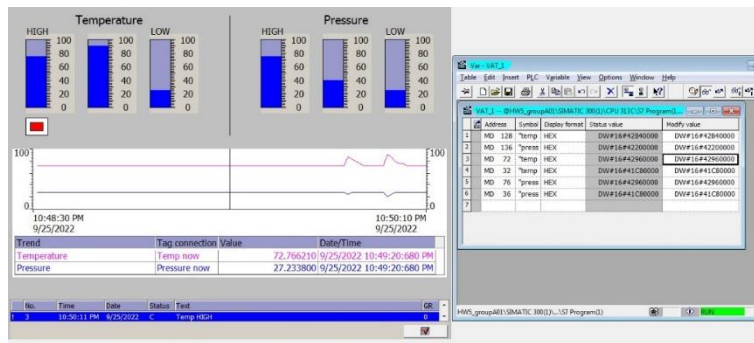


เมื่อปรับค่าขอบเขตสำหรับ Pressure ให้มีค่าน้อยกว่าค่า Pressure ปัจจุบัน

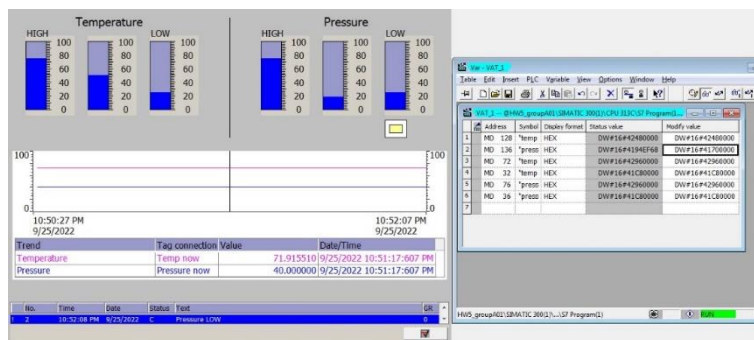
- Screen2:



การทำงานของ Trend view

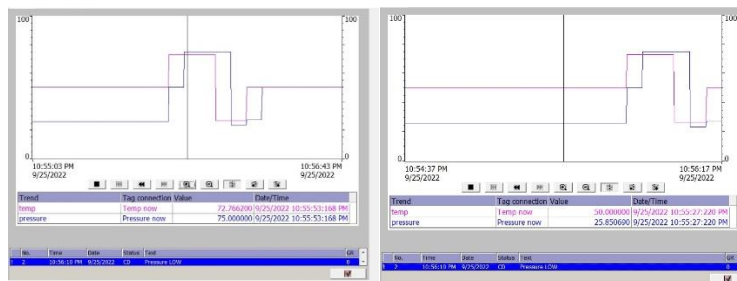


เมื่อกำหนดให้ Temperature มีค่าเกินกว่า TSH



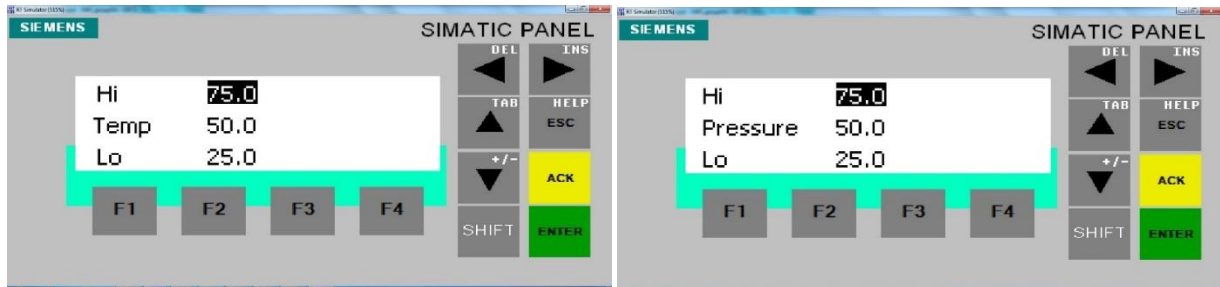
เมื่อกำหนดให้ Pressure มีค่าต่ำกว่า PSL

- Screen3:

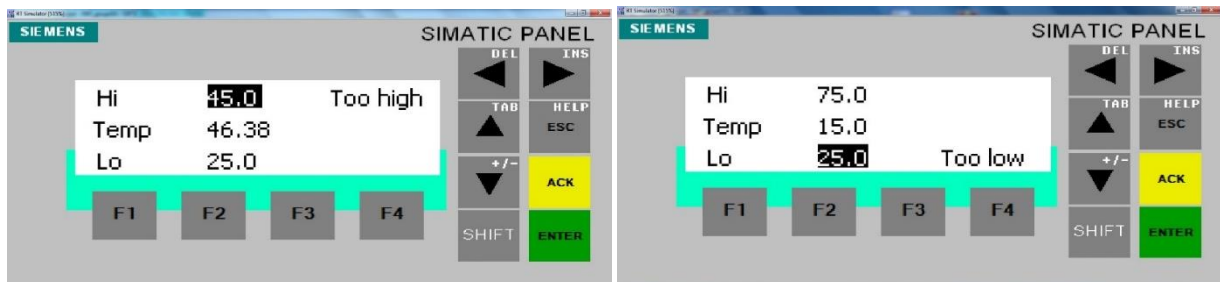


Historical Trend

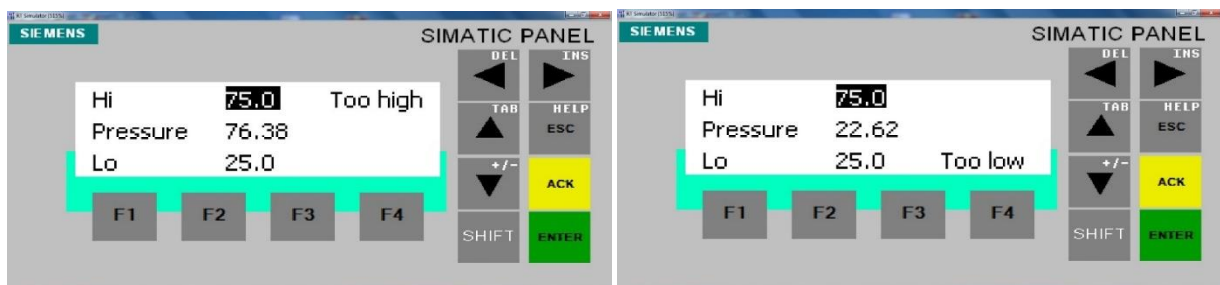
- OP73:



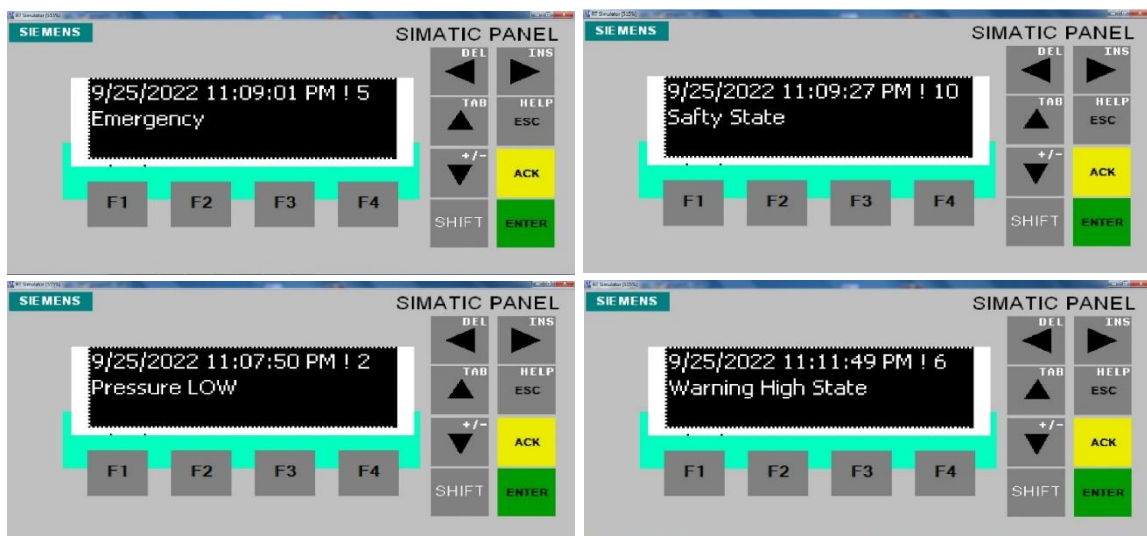
จอ OP73 เมื่ออยู่ใน normal state



เมื่อปรับค่า Temperature ให้สูงกว่า TSH และต่ำกว่า TSL ในหน้า Temperature Screen



เมื่อปรับค่า Pressure ให้สูงกว่า PSH และต่ำกว่า PSL ในหน้า Pressure Screen



จอ Alarm view ของ OP73 ที่แสดงสถานะต่างๆ

➤ Questions

1. What is the obvious additional feature for better control of this reactor?

ANS: ควรทำให้ระบบของเราสามารถดำเนินการแก้ไขด้วยตนเองได้โดยที่ Operator ไม่ต้องกดปุ่ม Reset ทุกครั้งในกรณีที่ไม่ใช่สถานะที่ฉุกเฉิน เพื่อให้ Process สามารถดำเนินงานต่อไปได้

2. If there are more reactor to control, what is the best programming method for this problem?

ANS: วางแผนลำดับการทำงานของโปรแกรมให้เหมาะสม โดยจะต้องเขียนโปรแกรมให้ตรวจสอบ Reactor แต่ละตัวให้มีลำดับการดำเนินงานในการแก้ไขปัญหาของแต่ละตัวโดยใช้เวลาน้อยที่สุดและ Operator สามารถเข้าใจและจัดการกับปัญหาได้ง่ายที่สุด

➤ Summary:

โปรแกรมควบคุม reactor ที่เราออกแบบนี้ จะมีการควบคุมทั้งเวลาในการทำงานของ reactor, ความดัน และอุณหภูมิ โดยมีเงื่อนไขต่าง ๆ โดยสรุป คือ เริ่มต้นเครื่องจะอยู่ใน Step1 (initial) ถ้าหากเรากดปุ่ม Start ตัวนับเวลาจะทำการนับเวลา 5 วินาที ถึงค่อยสั่งให้เครื่องเริ่มทำงานหรือเข้าสู่ Step3 (start) และเมื่ออยู่ใน Step 3 นี้ ก็จะมีสัญญาณไฟเตือนกระพริบ 0.5 hz เป็นเวลา 30 วินาที แล้วจึงเข้าสู่ Step4 (normal) ต่อไป โดยเมื่อเข้าสู่ Step4 จะทำให้ main timer เริ่มทำงาน ก็จะนับเวลาไปจนครบ 3 นาที ซึ่งถ้าหากนับเวลาไปจนครบ 3 นาทีแล้ว เครื่องยังคงมีสถานะปกติ เครื่องก็จะสามารถกลับเข้าสู่ Step1 ได้โดยอัตโนมัติ แต่เครื่องทั่วไปมันมักจะไม่นับปกติอยู่แล้วเนื่องจากภายในมีความร้อน มีความดัน มีอุณหภูมิที่เปลี่ยนแปลงได้ ก็อาจเกิดกรณีที่อุณหภูมิหรือความดันสูงเกินปกติ หรืออาจเกิดกรณีที่อุณหภูมิหรือความดันต่ำกว่าปกติ ตัวอย่างเช่น เดิมอยู่ใน Step4 (normal) ตัว main timer ก็จะนับเวลาการทำงานไปเรื่อย ๆ ตั้งแต่ตอนที่เข้าสู่ Step4 เป็นต้นไป แต่หากน้ำร้อนเกินปกติ จะส่งผลให้ตัวจับเวลาอีกตัวทำงาน โดยหากร้อนเกินไป 3 วินาที เครื่องก็จะเข้าสู่ Step6 (warning high) นั่นคือจะทำการเตือนและปล่อยน้ำเย็นออกมาเพื่อทำให้สถานะของความดันและอุณหภูมิกลับไปปกติ โดยหากอยู่ใน Step6 ไม่เกิน 10 วินาที แล้วความดันและอุณหภูมิกลับไปปกติอีกครั้ง ถ้าหากกดปุ่ม reset ร่วมด้วย ก็จะทำให้เครื่องกลับสู่ Step4 (normal) ได้ แต่ถ้าหากผ่านไป 10 วินาทีแล้วไม่สามารถที่จะทำให้สถานะกลับเป็นปกติได้ภายใน 20 วินาทีอีก ก็เข้าสู่ Step7 (alarm high) และถ้าหากยังไม่สามารถที่จะทำให้สถานะกลับเป็นปกติได้ภายใน 20 วินาทีอีก ก็เข้าสู่ Step11 (safety) ทันที ซึ่งจริง ๆ แล้ว เมื่อเครื่องมีอุณหภูมิที่ไม่ปกติ หรือแม้กระทั่งอยู่ในสถานะปกติใน Step4 (normal) เราก็สามารถที่จะกดปุ่ม Emergency stop เพื่อสั่งให้เครื่องเข้าสู่ Step safety ได้ทันที หรือหากมีอุณหภูมิและความดันสูงทั้งคู่ เครื่องก็จะสามารถเข้าสู่ Step safety ได้เองโดยอัตโนมัติเช่นเดียวกัน โดยระยะเวลาการทำงานทั้งหมดแม้ว่าเครื่องจะอยู่ในสถานะปกติหรือผิดปกตินั้น จะขึ้นอยู่กับการทำงานของ main timer ซึ่งจะมีรอบการทำงานทั้งหมด 3 นาที นั่นคือ ถ้าหากเครื่องยังคงอยู่ในโหมด pre-warning high/low, warning high/low และ alarm high/low กรณีที่เครื่องกลับมาอยู่ในสถานะปกติ และเครื่องทำงานครบ 3 นาทีแล้ว หากเรากดปุ่ม reset จะทำให้กลับสู่ step1 (initial) ได้