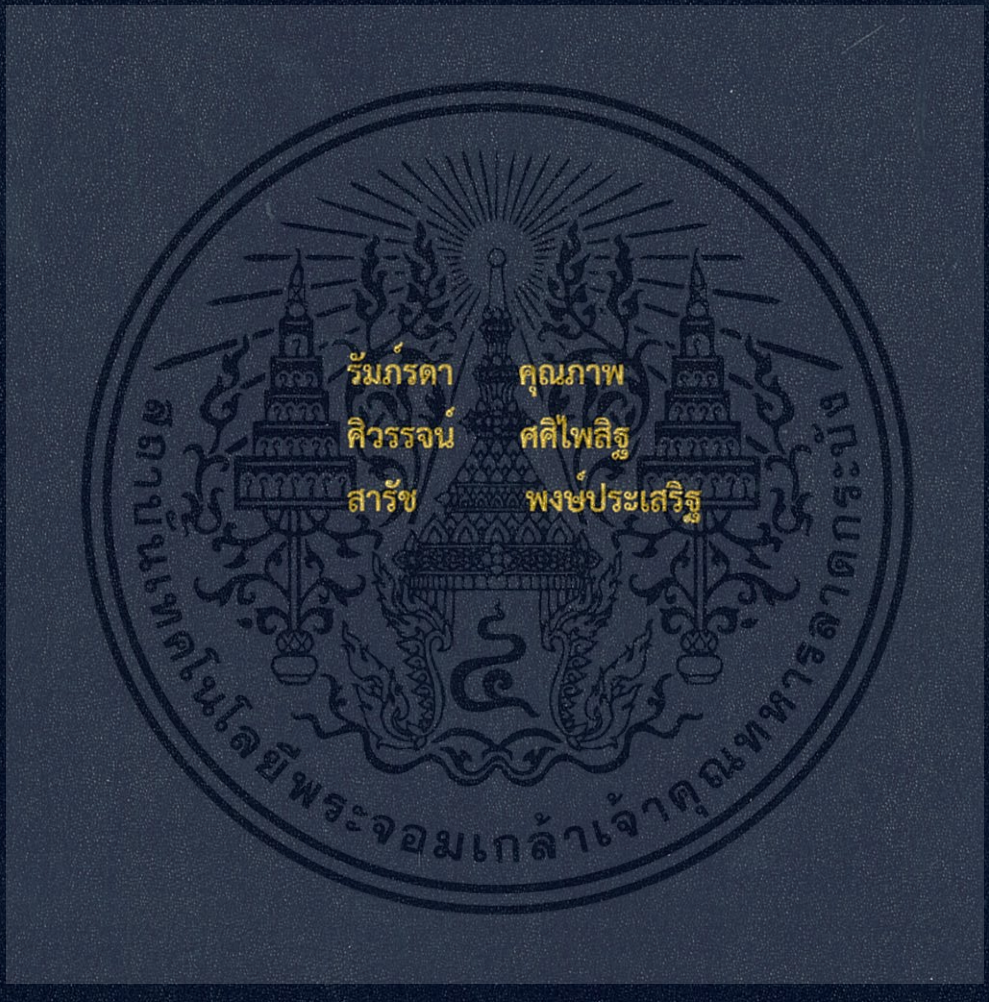
ระบบควบคุมระดับน้ำแบบไร้สาย : ส่วนซอฟต์แวร์
WIRELESS LEVEL CONTROL SYSTEM : SOFTWARE PART



รัมภ์รดา คุณภาพ

ศิวรรจน์ ศศิไพสิฐ

สารัช พงษ์ประเสริฐ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมระดับน้ำแบบไร้สาย : ส่วนซอฟต์แวร์

WIRELESS LEVEL CONTROL SYSTEM : SOFTWARE PART

T144415

รัมภ์รดา คุณภาพ

ศิวรรจน์ ศศิไพสิฐ

สารัช พงษ์ประเสริฐ

สจหมู
เลขทะเบียน 144415
วัน.เดือน.ปี 24 พ.ย. 2559

b.12818501
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WIRELESS LEVEL CONTROL SYSTEM : SOFTWARE PART

RAMRADA KHUNNAPHAP

SIWAN SASIPAISITH

SARAT PONGPRASERT

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN CONTROL ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2015

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2558

ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมระดับน้ำแบบไร้สาย : ส่วนซอฟต์แวร์
WIRELESS LEVEL CONTROL SYSTEM : SOFTWARE PART

ผู้จัดทำ นางสาวรัมภ์รดา คุณภาพ 55011046
นางสาวศิวรรจน์ ศศิไพสิฐ 55011216
นายสารัช พงษ์ประเสริฐ 55011294

..... อาจารย์ที่ปรึกษา
(ศาสตราจารย์ ดร.วันชัย ริ้วรุจา)

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์เทพจิตร์ เชยโภคา)

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.วรรณดี เพชรมณีล้ำค่า)

..... อาจารย์ที่ปรึกษา
(ดร.สิริชัย ธรรมารักษ์วัฒนะ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมระดับน้ำแบบไร้สาย : ส่วนซอฟต์แวร์

โดย

นางสาวรัมภ์รดา	คุณภาพ	55011046
นางสาวศิวรรจน์	ศศิไพสิฐ	55011216
นายสารัช	พงษ์ประเสริฐ	55011294

อาจารย์ที่ปรึกษา

ศาสตราจารย์ ดร.วันชัย ริ้วรุจา

ผู้ช่วยศาสตราจารย์เทพจิตร์ เชยโภคา

ผู้ช่วยศาสตราจารย์ ดร.วรรณดี เพชรมณีล้ำค่า

ดร.สิริชัย ธรรมารักษ์วัฒนะ

ปีการศึกษา 2558

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการนำเสนอในส่วนโปรแกรมของระบบควบคุมระดับน้ำแบบไร้สาย โดยการใช้ PLC Siemens รุ่น S7-1200 ซึ่งแบ่งเป็น 2 ส่วนคือ การเขียนโปรแกรมเพื่อควบคุมระบบ และการสร้างหน้าแสดงผลสำหรับผู้ใช้งาน

ในส่วนของการเขียนโปรแกรม จะใช้โปรแกรม STEP7 TIA Portal V13 ในการเขียนคำสั่งแลดเดอร์เพื่อไปควบคุมพีแอลซี โดยเครื่องพีแอลซีจะต้องรับค่าจากเซนเซอร์เพื่อนำมาประมวลผล และส่งข้อมูลไปยังมอเตอร์แบบไม่ใช้สายต่อระหว่างอุปกรณ์ โดยใช้ตัวควบคุมแบบพีไอดี ในการควบคุมระบบ โดยบล็อคพีไอดีจะนำค่าผิดพลาดระหว่างค่าเป้าหมาย และค่ากระบวนการ มาคำนวณแล้วพยายามลดค่าผิดพลาดลง โดยใช้พารามิเตอร์ต่างๆ ที่เกี่ยวข้อง

จากนั้นจึงสร้างหน้าแสดงผลสำหรับผู้ใช้งาน โดยใช้โปรแกรม WinCC Runtime Advanced V13 ที่มากับโปรแกรม TIA Portal V13 ซึ่งใช้สำหรับสร้างหน้ากราฟิกโดยจะเชื่อมกับตัวแปรต่างๆ ที่ถูกสร้างขึ้นในโปรแกรม TIA Portal V13 ทำให้ผู้ใช้สามารถดูข้อมูลต่างๆ ของระบบที่วัดได้จากเซนเซอร์, เปลี่ยนค่าเป้าหมายของระบบ รวมถึงการสั่งงานมอเตอร์ได้ง่ายขึ้นผ่านหน้าแสดงผลกราฟิกที่สร้างขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WIRELESS LEVEL CONTROL SYSTEM : SOFTWARE PART

By

Miss Ramrada Khunnaphap 55011046

Miss Siwan Sasipaisith 55011216

Mr. Sarat Pongprasert 55011294

Advisors

Prof.Dr.Vanchai Riewruja

Asst.Prof.Thepjit Cheypoca

Asst.Prof.Dr.Wandee Petchmaneelumka

Dr. Sirichai Tammaruckwattana

Academic Year 2015

ABSTRACT

This thesis presents wireless level control system in the part of software for control the system using PLC S7-1200 device from Siemens Company. There are two sessions in the thesis : system control and human interface. STEP7 TIA Portal V13 is used to design a Ladder logic to control the PLC as the process controller. The PLC controller processes the data received from sensors and transfers the result to the motor with wireless communication between the devices. PID Controller is selected as the plant control method. The PID controller continuously calculates an error value as the difference between a desired setpoint and a measured process variable and then attempts to minimize the error over time by adjustment of a control variable.

Later, the human machine interface is created with WinCC Runtime V13. The program is used to arrange the graphic windows and connect any object in the program with parameters created in TIA Portal V13. Users can monitor any values received from sensors, change the setpoint of the system and control the motor directly through the self-optimized graphic interface.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์ฉบับนี้ ได้รับการสนับสนุน และความช่วยเหลือเป็นอย่างดีจากผู้ช่วยศาสตราจารย์ ดร.วรรณดี เพชรมณีล้ำค่า และผู้ช่วยศาสตราจารย์เทพจิตร์ เชยโภคา ที่ได้ให้คำปรึกษา พร้อมทั้งคำแนะนำที่เป็นประโยชน์ต่อโครงงานมาโดยตลอด ทำให้งานสามารถสำเร็จลุล่วงไปได้ด้วยดี ทางผู้จัดทำรู้สึกซาบซึ้ง และขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณศาสตราจารย์ ดร.วันชัย ริ้วรุจา และดร.สิริชัย ธรรมารักษ์วัฒนะ ที่คอยให้คำปรึกษา และสอบถามถึงความคืบหน้าของงานมาโดยตลอด

ขอบคุณเพื่อนๆ ทุกคนที่เป็นกำลังใจ และช่วยเหลือหาทางแก้ไขปัญหาต่างๆ รวมทั้งคอยถามไถ่ความคืบหน้าของโครงงานอยู่เสมอ

สุดท้ายนี้ทางผู้จัดทำขอกราบขอบพระคุณบิดา มารดา และครอบครัวที่คอยให้กำลังใจ และเป็นแรงบันดาลใจที่ดีที่สุดที่ทำให้โครงงานนี้สำเร็จลุล่วงไปได้

ผู้จัดทำ

นางสาวรัมภ์รดา คุณภาพ

นางสาวศิวรรจน์ ศศิไพสิฐ

นายสารัช พงษ์ประเสริฐ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	XI
บทที่ 1 บทนำ	1
1.1 ที่มา และความสำคัญ	1
1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์	1
1.3 ขอบเขตของปริญญานิพนธ์	1
1.4 ขั้นตอนการศึกษา และจัดทำปริญญานิพนธ์	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
1.6 รายละเอียดของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 Programmable Logic Control (PLC)	3
2.1.1 โครงสร้างทั่วไป และหลักการทำงานพื้นฐานของพีแอลซี	3
2.1.2 ข้อดีของพีแอลซี	4
2.1.3 ความแตกต่างระหว่างพีแอลซีกับคอมพิวเตอร์	5
2.2 PLC Siemens รุ่น S7-1200	6
2.2.1 อุปกรณ์ต่อขยาย (Expansion Capability Module)	6
2.3 โปรแกรม STEP7 TIA Portal V13	7
2.3.1 ภาษาที่ใช้ในการเขียนโปรแกรม TIA Portal V13	8
2.3.2 บล็อกคำสั่งพื้นฐานที่เกี่ยวข้อง (Basic Instruction)	10
2.3.3 คำสั่งเพิ่มเติมที่เกี่ยวข้อง (Extended Instruction)	10
2.3.4 บล็อกการเชื่อมต่อ (Communications)	11
2.4 โปรแกรม WinCC Runtime Advanced V13	12
2.5 ประเภทการควบคุม	13
2.5.1 ระบบควบคุมแบบเปิด (Open-Loop Control System)	13
2.5.2 การควบคุมแบบป้อนกลับ (Feedback Control)	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.5.3 การควบคุมแบบแคสเคด (Cascade Control)	14
2.6 ระบบควบคุมแบบสัดส่วน-ปริพันธ์-อนุพันธ์	15
2.6.1 ตัวควบคุมแบบสัดส่วน (Proportional Controller)	16
2.6.2 ตัวควบคุมแบบปริพันธ์ (Integral Controller)	17
2.6.3 ตัวควบคุมแบบอนุพันธ์ (Derivative Controller)	18
2.7 เทคโนโลยี WiFi	20
2.7.1 การรบกวน	21
2.7.2 มาตรฐานของเทคโนโลยี WiFi	22
2.8 โปรโตคอล TCP/IP	23
บทที่ 3 ขั้นตอนการดำเนินงาน	24
3.1 โครงสร้างของระบบโดยรวม	24
3.2 โปรแกรม TIA Portal V13	25
3.2.1 สร้างโปรเจค และตั้งค่าอุปกรณ์	25
3.2.2 เขียนโปรแกรมพีแอลซีเพื่อควบคุมระบบโดยใช้ตัวควบคุมแบบพีไอดี	27
3.3 โปรแกรม WinCC RT Advanced	47
3.3.1 เพิ่ม และตั้งค่าอุปกรณ์	47
3.3.2 การสร้าง Symbolic I/O field แสดงสถานะ	49
3.3.3 การสร้างกราฟแสดงแนวโน้มบล็อกพีไอดี	51
บทที่ 4 การทดลองและผลการทดลอง	54
4.1 ขั้นตอนการทดลอง	54
4.1.1 ทดลองการรับค่าที่ขาแอนะล็อกของพีแอลซี โดยบล็อก NORM_X และ SCALE_X	54
4.1.2 ผลการทดลองเปลี่ยนค่าเป้าหมายของระบบเพื่อหา %Overshoot และ $\%e_{ss}$	57
บทที่ 5 สรุปผลการดำเนินงาน ปัญหา และแนวทางการปรับปรุงโครงงาน	71
5.1 สรุปผลการดำเนินงาน	71
5.2 ปัญหา และการแก้ไข	71
5.3 ข้อเสนอแนะ	72
เอกสารอ้างอิง	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
ภาคผนวก	74
ภาคผนวก ก เอกสารข้อมูลประกอบการเขียนแลดเดอร์บล็อกต่างๆ	75
ก.1 บล็อก NORM_X, SCALE_X	75
ก.2 บล็อก TSEND_C	76
ก.3 บล็อก VAL_STRG	78
ก.4 บล็อก Strg_TO_Chars	80
ก.5 บล็อก PID_Compact	81
ภาคผนวก ข ข้อมูลคุณสมบัติของ PLC Siemens ตระกูล S7-1200	83
ข.1 คุณสมบัติของ PLC Siemens ตระกูล S7-1200	83
ข.2 PLC Siemens Model S7-1214C AC/DC/Relay (6ES7 214-1BG31-0XB0)	84
ข.3 CPU 1214C Wiring Diagrams	86
ข.4 Analog Inputs	87
ภาคผนวก ค โปสเตอร์	88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างการทำงานพื้นฐานของพีแอลซี	4
2.2 PLC Siemens รุ่น S7-1200	6
2.3 การต่ออุปกรณ์ต่อขยาย	7
2.4 หน้าต่างแสดงโปรแกรม TIA Portal V13	8
2.5 รูปแบบโปรแกรม Ladder Logic (LAD)	8
2.6 รูปแบบการเขียนโปรแกรมแบบ Function Block Diagram (FBD)	9
2.7 รูปแบบการเขียนโปรแกรมแบบ Structure Control Language (SCL)	9
2.8 หน้าเชื่อมต่อของโปรแกรมภาษา SCL ที่สามารถกำหนดตัวแปรต่างๆ ได้	9
2.9 ฟังก์ชันบล็อก NORM_X	10
2.10 ฟังก์ชันบล็อก SCALE_X	10
2.11 ฟังก์ชันบล็อก VAL_STRG	11
2.12 ฟังก์ชันบล็อก Strg_TO_Char	11
2.13 ฟังก์ชันบล็อก TSEND_C	11
2.14 หน้าต่างโปรแกรม WinCC Runtime Advanced V13	12
2.15 หน้าต่าง WinCC Screen พร้อมกับหน้า Main Block สำหรับเขียนโปรแกรม	13
2.16 บล็อกไดอะแกรมของระบบควบคุมแบบเปิด	13
2.17 บล็อกไดอะแกรมของระบบควบคุมแบบป้อนกลับ	14
2.18 บล็อกไดอะแกรมของการควบคุมแบบแคสเคด	14
2.19 ตัวอย่างการควบคุมระดับน้ำแบบแคสเคด	15
2.20 ตัวควบคุมแบบสัดส่วน	16
2.21 ตัวควบคุมแบบปริพันธ์	17
2.22 ตัวควบคุมแบบอนุพันธ์	18
2.23 ผลตอบสนองต่ออินพุตแบบขั้นบันไดในการใช้งานระบบควบคุมพีไอดีในลักษณะต่างๆ กัน	18
2.24 ค่าพารามิเตอร์ของระบบควบคุมแบบพีไอดี	20
2.25 แบบจำลองการเชื่อมต่อ WiFi แบบโหมดแอคเซสพอยต์	21
2.26 พื้นที่การกระจายสัญญาณ WiFi	21
3.1 จำลองโครงสร้างระบบด้วยโปรแกรม SolidWorks	25
3.2 หน้าแสดงผล Portal view ในส่วนของการสร้างโปรเจค	25
3.3 หน้าแสดงผล Portal view ในส่วนของการเพิ่มอุปกรณ์	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.4 หน้าแสดงผล Portal view ในส่วนของการเลือกอุปกรณ์พีแอลซีคอนโทรลเลอร์	26
3.5 การตั้งค่า IP Address ของหน่วยประมวลผลกลางของพีแอลซี (1)	27
3.6 การตั้งค่า IP Address ของหน่วยประมวลผลกลางของพีแอลซี (2)	27
3.7 การสร้างพื้นที่สำหรับเขียนโปรแกรมควบคุม Main [OB1]	28
3.8 การสร้าง Cyclic interrupt block	29
3.9 การเพิ่มบล็อก PID_Compact	29
3.10 การสร้างหน้า Data block สำหรับตั้งค่าพารามิเตอร์พีไอดี	30
3.11 ตัวแปรสำหรับบล็อกพีไอดีในหน้า Data block	30
3.12 การใส่พารามิเตอร์ลงบล็อก PID_Compact	31
3.13 การตั้งค่า Cyclic time	31
3.14 สร้างพารามิเตอร์สำหรับบล็อกพีไอดีสำหรับการควบคุมแบบแคสเคด	32
3.15 การใส่พารามิเตอร์ลงบล็อก PID_Compact สำหรับการควบคุมแบบแคสเคด	32
3.16 การเพิ่มบล็อก TSEND_C สำหรับการเชื่อมต่ออุปกรณ์ไร้สายผ่านโปรโตคอล TCP/IP	32
3.17 การตั้งค่าการเชื่อมต่อของบล็อก TSEND_C	33
3.18 ค่าการตั้งค่าการเชื่อมต่อบล็อก TSEND_C กับอุปกรณ์	34
3.19 การใส่พารามิเตอร์ลงบล็อก TSEND_C	34
3.20 สร้างพารามิเตอร์สำหรับบล็อก NORM_X และ SCALE_X	35
3.21 การเพิ่มบล็อก NORM_X และ SCALE_X	35
3.22 การใส่พารามิเตอร์ลงบล็อก NORM_X และ SCALE_X สำหรับ Level	36
3.23 การใส่พารามิเตอร์ลงบล็อก NORM_X และ SCALE_X สำหรับ Flow	36
3.24 การโหลดโปรแกรมลงเครื่องพีแอลซี	36
3.25 หน้าต่าง Download to device	37
3.26 หน้าต่าง Load preview	37
3.27 หน้าต่าง Load result	38
3.28 การ Monitor ค่า และการทำงานของโปรแกรมแลดเดอร์	38
3.29 ค่าที่เข้าขาแอนะล็อก IW64 และ IW66	39
3.30 โปรแกรมสำหรับบล็อก TSEND_C ส่งข้อมูลเป็นเวลา 50 ms และส่งทุกๆ 50 ms	39
3.31 โปรแกรมสำหรับบล็อก TSEND_C ส่งข้อมูลเป็นเวลา 50 ms และส่งทุกๆ 50 ms	40
3.32 การเพิ่มบล็อก VAL_STRG และ Strg_TO_Chars	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.33 การเปิดหน้า configuration window	41
3.34 หน้าต่าง configuration window	42
3.35 การตั้งค่า Input/Output Parameter	42
3.36 การตั้งค่า Process value scaling	43
3.37 การตั้งค่า Process value limits	43
3.38 การปรับเปลี่ยนค่าเป้าหมาย	44
3.39 การเปิดหน้า commissioning window	44
3.40 การปรับค่าพีไอดี (1)	45
3.41 การปรับค่าพีไอดี (2)	45
3.42 การอัพโหลดพารามิเตอร์ของบล็อก PID_Compact	45
3.43 เงื่อนไขการสั่งงานมอเตอร์แบบอัตโนมัติ	46
3.44 แลดเดอร์สำหรับการสั่งงานมอเตอร์แบบไม่อัตโนมัติ (Manual)	46
3.45 หน้าแสดงผล Portal view	47
3.46 หน้าแสดงผล Project view	47
3.47 หน้า Device & networks แสดงการเชื่อมต่อของอุปกรณ์ต่างๆ	48
3.48 หน้า Runtime setting สำหรับตั้งค่าหน้าจอ HMI	48
3.49 หน้า Screen สำหรับสร้าง และปรับแต่งหน้า HMI	49
3.50 ตำแหน่งตัวบอกสถานะควบคุม	49
3.51 หน้า Text and graphic lists	50
3.52 การสร้าง Symbolic I/O field และ Tag text list	50
3.53 วิธีการเพิ่ม Event สำหรับ Button switch element	51
3.54 การสร้างหน้าแสดงผลแนวโน้มกราฟ	51
3.55 การสร้างปุ่มสำหรับเรียกใช้หน้าจอแสดงแนวโน้มกราฟของบล็อกพีไอดี	52
3.56 ส่วนประกอบหน้าเชื่อมต่อของระบบควบคุมระดับน้ำแบบไร้สาย	52
3.57 รูปแบบหน้าต่างแสดงกราฟแนวโน้มของบล็อกพีไอดี	53
4.1 บล็อก NORM_X & SCALE_X พร้อมใช้งาน และทดสอบ	54
4.2 ทดลองรับค่าที่ 1 โวลต์	55
4.3 ทดลองรับค่าที่ 2 โวลต์	55
4.4 ทดลองรับค่าที่ 3 โวลต์	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.5 ทดลองรับค่าที่ 4 โวลต์	56
4.6 ทดลองรับค่าที่ 5 โวลต์	57
4.7 หน้าจอ HMI เมื่อปรับค่าเป้าหมายจาก 0 เป็น 10	58
4.8 ค่าพารามิเตอร์พีไอดีของคอนโทรลเลอร์ตัวหลัก	59
4.9 ค่าพารามิเตอร์พีไอดีของคอนโทรลเลอร์ตัวรอง	59
4.10 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 0 เป็น 10	60
4.11 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 10 เป็น 20	60
4.12 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 20 เป็น 30	61
4.13 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 30 เป็น 40	61
4.14 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 40 เป็น 50	62
4.15 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 50 เป็น 60	62
4.16 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 60 เป็น 70	63
4.17 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 70 เป็น 80	63
4.18 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 80 เป็น 90	64
4.19 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 90 เป็น 100	64
4.20 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 100 เป็น 90	65
4.21 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 90 เป็น 80	65
4.22 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 80 เป็น 70	66
4.23 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 70 เป็น 60	66
4.24 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 60 เป็น 50	67
4.25 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 50 เป็น 40	67
4.26 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 40 เป็น 30	68
4.27 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 30 เป็น 20	68
4.28 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 20 เป็น 10	69
4.29 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 10 เป็น 0	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 เปรียบเทียบระหว่างระบบลำดับ (SEQUENCE) กับระบบพีแอลซี	5
2.2 ผลของการเพิ่มค่าพารามิเตอร์ของพีไอดีคอนโทรลเลอร์	19
4.1 ตารางผลการทดลองการรับค่าโวลต์ที่ขาแอนะล็อกพีแอลซี	57
4.2 ผลการทดลองเปลี่ยนค่าเป้าหมายของระบบเพื่อหา %OVERSHOOT และ $\%E_{SS}$	70
5.1 ปัญหาในการดำเนินงาน และวิธีแก้ไข	71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มา และความสำคัญ

ปัจจุบันพีแอลซี (Programmable Logic Control, PLC) ถือเป็นหัวใจหลักในการควบคุมการทำงานของระบบต่างๆ ในอุตสาหกรรม ซึ่งนำมาใช้แทนวงจรรีเลย์ ทำให้จำนวนสายไฟภายในระบบน้อยลง ซึ่งช่วยลดค่าใช้จ่ายได้ และการใช้งานที่ง่ายของพีแอลซีช่วยลดเวลา การเปลี่ยนระบบก็สามารถทำได้ง่าย แต่ระบบยังคงต้องใช้สายในการเชื่อมต่อสื่อสารกับอุปกรณ์อื่นๆ อยู่ นอกจากนี้ปัจจุบันระบบเครือข่ายไร้สายเริ่มเป็นที่สนใจมากขึ้นในระบบอุตสาหกรรม เนื่องจากมีความสะดวกรวดเร็วในการส่งผ่านข้อมูลได้โดยไม่ต้องเคลื่อนย้ายอุปกรณ์ ทำให้สามารถควบคุมระบบได้ในระยะไกลตราบเท่าที่ยังอยู่ในระยะการส่งข้อมูล แต่การใช้ Client Module ในการควบคุมแบบไร้สายนั้นมีราคาต้นทุนที่สูง ทางผู้จัดทำจึงนำพีแอลซีมาประยุกต์ใช้กับระบบเครือข่ายไร้สาย จำลองระบบการควบคุมระดับน้ำ โดยการส่งข้อมูลระหว่างพีแอลซีกับอุปกรณ์ต่างๆ ในระบบ เช่น เซนเซอร์ และมอเตอร์ ผ่านโมดูล WiFi เพื่อควบคุมความเร็วมอเตอร์ให้รักษาระดับน้ำตามที่เราต้องการ โดยใช้ WiFi Module ESP8266 Model 01 เนื่องจากราคาที่ถูก และมีรูปแบบการรับส่งข้อมูลตามมาตรฐาน IEEE 802.11 b/g/n ที่สามารถใช้งานได้ทั่วไป

1.2 วัตถุประสงค์ในการทำปริญญานิพนธ์

1. เพื่อออกแบบ และจำลองระบบควบคุมระดับน้ำอัตโนมัติแบบไร้สาย
2. เพื่อศึกษาการใช้งานโปรแกรม STEP7 TIA Portal V13 และการเขียนคำสั่งในการควบคุมระบบ
3. เพื่อศึกษาการทำงานของตัวควบคุมแบบพีไอดี
4. เพื่อสร้างหน้าจอแสดงผลสำหรับติดตามค่า และควบคุมระบบสำหรับผู้ใช้งาน (Human Interface : HMI) ผ่านโปรแกรม WinCC Runtime Advanced V13

1.3 ขอบเขตของปริญญานิพนธ์

1. สร้างแบบจำลองระบบควบคุมระดับน้ำอัตโนมัติแบบไร้สาย ที่สามารถควบคุมระบบผ่านทางหน้าจอแสดงผลในส่วนของผู้ใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เขียนโปรแกรมในการสั่งการพีแอลซี Siemens รุ่น S7-1200 เพื่อควบคุมระดับน้ำในระบบควบคุมระดับน้ำอัตโนมัติแบบไร้สาย

1.4 ขั้นตอนการศึกษา และจัดทำปริญญานิพนธ์

1. ศึกษาข้อมูล และทฤษฎีทั้งหมดที่เกี่ยวข้องกับปริญญานิพนธ์ ได้แก่ พีแอลซี Siemens S7-1200, โปรแกรม TIA Portal V13, โปรแกรม WinCC Runtime Advanced V13, ตัวควบคุมแบบพีไอดี, เทคโนโลยี WiFi และโปรโตคอล TCP/IP
2. วางแผนการทำงาน
3. เขียนโปรแกรมในการสั่งการพีแอลซีเพื่อควบคุมระดับน้ำในระบบ
4. สร้างหน้าจอแสดงผลสำหรับติดตามค่า และควบคุมระบบสำหรับผู้ใช้งาน
5. ทำการทดลอง และบันทึกข้อมูลการทำงาน
6. สรุปผลการทำงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถสร้างระบบควบคุมระดับอัตโนมัติแบบไร้สายได้
2. สามารถเขียนโปรแกรมในการสั่งการพีแอลซีเพื่อควบคุมระดับน้ำในระบบได้
3. สามารถนำระบบควบคุมระดับน้ำอัตโนมัติแบบไร้สายนี้ไปประยุกต์ใช้กับงานอุตสาหกรรมในอนาคตได้

1.6 รายละเอียดของปริญญานิพนธ์

เนื้อหาที่จะกล่าวในปริญญานิพนธ์ฉบับนี้ประกอบด้วย 5 บท กับอีก 3 ภาคผนวก ดังนี้

บทที่ 1 บทนำ เป็นการกล่าวถึงที่มาและความสำคัญของปริญญานิพนธ์ วัตถุประสงค์ของการทำโครงงาน ขอบเขตของโครงงาน ขั้นตอนการศึกษา และจัดทำโครงงาน ประโยชน์ที่คาดว่าจะได้รับ และรายละเอียดของปริญญานิพนธ์

บทที่ 2 ทฤษฎี หลักการ และความรู้ที่เกี่ยวข้องในส่วนของซอฟต์แวร์

บทที่ 3 ขั้นตอนการดำเนินงาน

บทที่ 4 ผลการทดลอง

บทที่ 5 สรุปผลการดำเนินงาน ปัญหา และแนวทางการปรับปรุงโครงงาน

ภาคผนวก ก เอกสารข้อมูลประกอบการเขียนแลดเดอร์บล็อกต่างๆ

ภาคผนวก ข ข้อมูลคุณสมบัติของพีแอลซี Siemens รุ่น S7-1200

ภาคผนวก ค โปสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 Programmable Logic Control (PLC)

อุตสาหกรรมในยุคแรกๆ การควบคุมการทำงานในโรงงานอุตสาหกรรมทั่วไปอาศัยแรงงานคนในการตรวจวัด คำนวณ และการปรับแต่งเป็นส่วนใหญ่ ซึ่งการควบคุมลักษณะนี้ให้ผลการควบคุมที่ไม่คงที่ และมีความแม่นยำค่อนข้างน้อย เนื่องจากถูกจำกัดด้วยขีดจำกัดความสามารถของมนุษย์ จึงได้มีการนำเครื่องจักรมาใช้งานแทนแรงงานคนมากขึ้น ซึ่งช่วยให้ลดต้นทุนการผลิต และเพิ่มมาตรฐานทั้งในด้านสินค้า และการดำเนินงานของโรงงาน จึงได้มีการคิดวงจรรีเลย์ขึ้นเพื่อให้สามารถควบคุมการทำงานเครื่องจักรให้ทำงานได้อย่างอัตโนมัติ จากนั้นจึงได้พัฒนามาเป็นพีแอลซีซึ่งทำให้สามารถลดความยุ่งยากทั้งในการติดตั้ง โปรแกรม และการบำรุงรักษา เป็นต้น

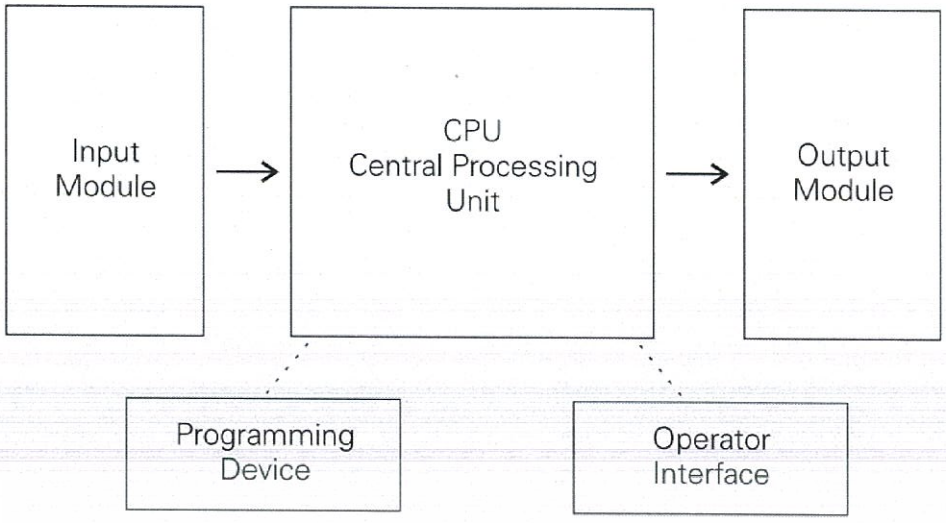
2.1.1 โครงสร้างทั่วไป และหลักการทำงานพื้นฐานของพีแอลซี

พีแอลซี (Programmable Logic Controller, PLC) เป็นอุปกรณ์ควบคุมการทำงานของเครื่องจักร หรือกระบวนการต่างๆ โดยมีโครงสร้างแบ่งออกเป็น 5 ส่วน ดังนี้

- หน่วยประมวลผลกลาง (Central Processing Unit, CPU)
- หน่วยความจำ (Memory Unit)
- หน่วยอินพุต/เอาต์พุต (Input/Output Unit)
- แหล่งจ่ายพลังงาน (Power Supply)
- อุปกรณ์ติดต่อภายนอก (Peripheral Device)

โดยตัวอินพุตจะรับข้อมูลทั้งสัญญาณแบบดิจิตอล และแอนะล็อกจากอุปกรณ์ต่างๆ เช่น สวิตช์เปิด – ปิด, เซนเซอร์ตรวจวัดอุณหภูมิ, เซนเซอร์ตรวจวัดระดับของเหลว หรือ เซนเซอร์ตรวจวัดอัตราการไหลของของเหลว เป็นต้น จากนั้นหน่วยประมวลผลกลางจะแปลงสัญญาณนั้นๆ ให้เป็นสัญญาณลอจิก (Logic Signal) และประมวลผลตามโปรแกรมที่ผู้ใช้เขียนไว้ในหน่วยความจำ แล้วตัวเอาต์พุตโมดูลจะแปลงคำสั่งควบคุมจากหน่วยประมวลผลกลางกลับเป็นสัญญาณดิจิตอลหรือแอนะล็อกเพื่อนำไปสั่งควบคุมอุปกรณ์ เช่น มอเตอร์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 โครงสร้างการทำงานพื้นฐานของพีแอลซี

2.1.2 ข้อดีของพีแอลซี

พีแอลซีถูกสร้างขึ้นมาเพื่อใช้ในการควบคุมการทำงานของเครื่องจักรหรือระบบต่างๆ แทนวงจรแบบเก่าหรือวงจรรีเลย์ ซึ่งวงจรรีเลย์มีข้อเสียคือ การเปลี่ยนแปลงเงื่อนไขในการควบคุมทำได้ยาก เมื่อใช้ไปนานๆ หน้าสัมผัสของรีเลย์จะเสื่อม และเมื่อมีความจำเป็นต้องเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่ จะต้องเดินสายใหม่ ซึ่งโดยทั่วไปตู้ควบคุมของวงจรรีเลย์ จะมีการต่อสายเป็นจำนวนมาก ทำให้การแก้ไขการควบคุมนี้เป็นไปได้ยาก แต่พีแอลซีสามารถทำได้โดยการเขียนโปรแกรมใหม่เท่านั้น นอกจากนี้พีแอลซียังใช้โซลิด-สเตท ซึ่งมีความน่าเชื่อถือกว่าระบบเดิม กินกระแสไปน้อยกว่า และสะดวกกว่าเมื่อต้องการปรับเปลี่ยนขั้นตอนการทำงานของเครื่องจักร

นอกจากนี้พีแอลซียังสามารถใช้งานร่วมกับอุปกรณ์อื่นๆ เช่น เครื่องอ่านบาร์โค้ด (Barcode Reader), เครื่องพิมพ์ (Printer) เป็นต้น และในปัจจุบันสามารถใช้งานเครื่องพีแอลซีแบบเดี่ยว (Standalone) หรือต่อพีแอลซีหลายๆ ตัวเข้าด้วยกัน (Network) เพื่อควบคุมการทำงานของระบบให้มีประสิทธิภาพมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 เปรียบเทียบระหว่างระบบลำดับ (Sequence) กับระบบพีแอลซี

	ระบบลำดับ หรือ การเดินสายไฟ	ระบบพีแอลซี
การปรับเปลี่ยนการควบคุมระบบ	ทำได้ยาก	ทำได้ง่าย
การซ่อมหรือการแก้ไข	ทำได้ยาก	ทำได้ง่าย
การติดต่อกับอุปกรณ์ภายนอก	ทำได้ยาก	ทำได้ง่าย
อายุการใช้งาน	น้อยกว่า เพราะมีส่วนเคลื่อนที่มาก	มากกว่า เพราะมีส่วนเคลื่อนที่น้อย
ติดต่อกับอุปกรณ์ไกลๆ	ทำได้ยุ่งยาก เพราะต้องเดินสายไฟยาวขึ้น	ทำได้ง่าย เพราะการเดินสายไฟน้อย
ความเร็วในการทำงาน	ช้า	เร็ว
ขนาด	ใหญ่	เล็ก
สัญญาณรบกวน	มาก	น้อย
การติดตั้ง	ใช้เวลามาก	ใช้เวลาน้อย
การทำงานที่ระบบซับซ้อน	ยาก ต้องใช้รีเลย์จำนวนมาก	ง่าย สะดวก

2.1.3 ความแตกต่างระหว่างพีแอลซีกับคอมพิวเตอร์

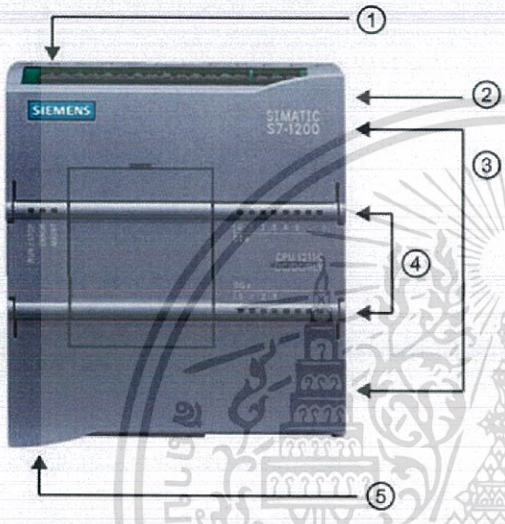
พีแอลซีเป็นคอมพิวเตอร์เฉพาะประเภทหนึ่ง โครงสร้างจึงเหมือนคอมพิวเตอร์แต่มีข้อแตกต่างดังต่อไปนี้คือ

- พีแอลซีถูกออกแบบให้มีความทนทานต่อสภาพแวดล้อมของโรงงานอุตสาหกรรม เช่น ความร้อน ความเย็น ระบบไฟฟ้ารบกวน การสั่นสะเทือน การกระแทก
- การใช้โปรแกรมของพีแอลซีจะไม่ยุ่งยากเหมือนของคอมพิวเตอร์ เนื่องจากพีแอลซีจะมีระบบตรวจสอบตัวเอง ทำให้สามารถใช้งาน และบำรุงรักษาได้ง่าย
- พีแอลซีทำงานได้ตามที่โปรแกรมเอาไว้เพียงโปรแกรมเดียว ทำให้ไม่ยุ่งยาก ส่วนคอมพิวเตอร์จะทำงานได้หลายๆ โปรแกรมพร้อมกัน จึงมีความยุ่งยากกว่า
- พีแอลซีใช้ควบคุมกระบวนการผลิตทุกชนิด ทั้งแบบแอนะล็อก (Analog) และแบบลอจิก (Logic Function)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 PLC Siemens รุ่น S7-1200

PLC จากบริษัท Siemens รุ่น S7-1200 เป็นพีแอลซีประเภทคอมแพค (Compact Type PLCs) ภายในกล่องประกอบด้วยตัวประมวลผล หน่วยความจำ ภาคอินพุต/เอาต์พุต และแหล่งจ่ายไฟ โดย PLC รุ่น S7-1200 นี้จะมีพอร์ตเฉพาะสำหรับการติดต่อสื่อสารผ่านเครือข่ายมาตรฐาน PROFINET ภายในตัว PLC คอมแพค แต่หากต้องการติดต่อสื่อสารผ่านมาตรฐาน RS232, RS485, GPRS, IEC, PROFIBUS, DNP3 และ WDC สามารถทำได้โดยการต่อ Communication Port Module เพื่อขยายเพิ่มได้



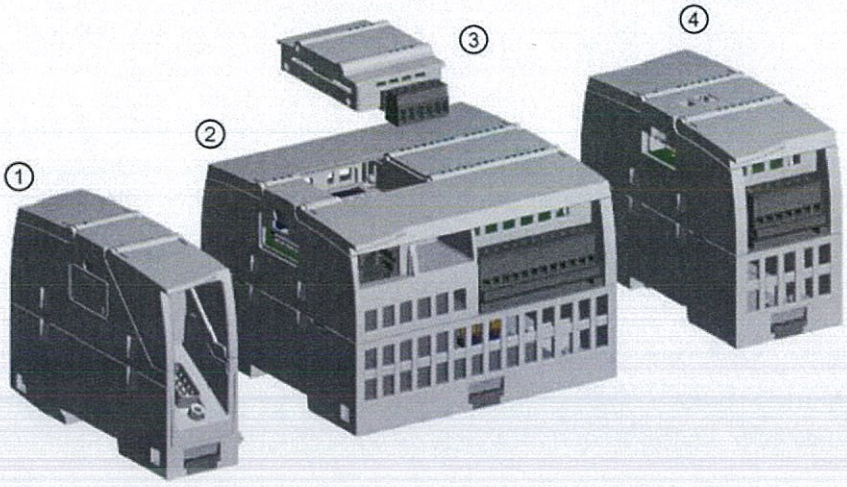
① Power connector
② Memory card slot under top door
③ Removable user wiring connectors (behind the doors)
④ Status LEDs for the on-board I/O
⑤ PROFINET connector (on the bottom of the CPU)

รูปที่ 2.2 PLC Siemens รุ่น S7-1200

2.2.1 อุปกรณ์ต่อขยาย (Expansion Capability Module)

หากมีความจำเป็นที่จะต้องขยับขยายระบบหรือต้องการเพิ่มความสามารถให้กับเครื่องจักรที่ใช้งาน สามารถเพิ่มอุปกรณ์ต่อขยายได้ เช่น Communication Port, Digital I/O, Analog I/O, RTD และ Thermocouple เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



① Communication module (CM) or communication processor (CP)
② CPU (CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C, CPU 1217C)
③ Signal board (SB) (digital SB, analog SB), communication board (CB), or Battery Board (BB) CPU (CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C, CPU 1217C)
④ Signal module (SM) (digital SM, analog SM, thermocouple SM, RTD SM, technology SM)

รูปที่ 2.3 การต่ออุปกรณ์ต่อขยาย

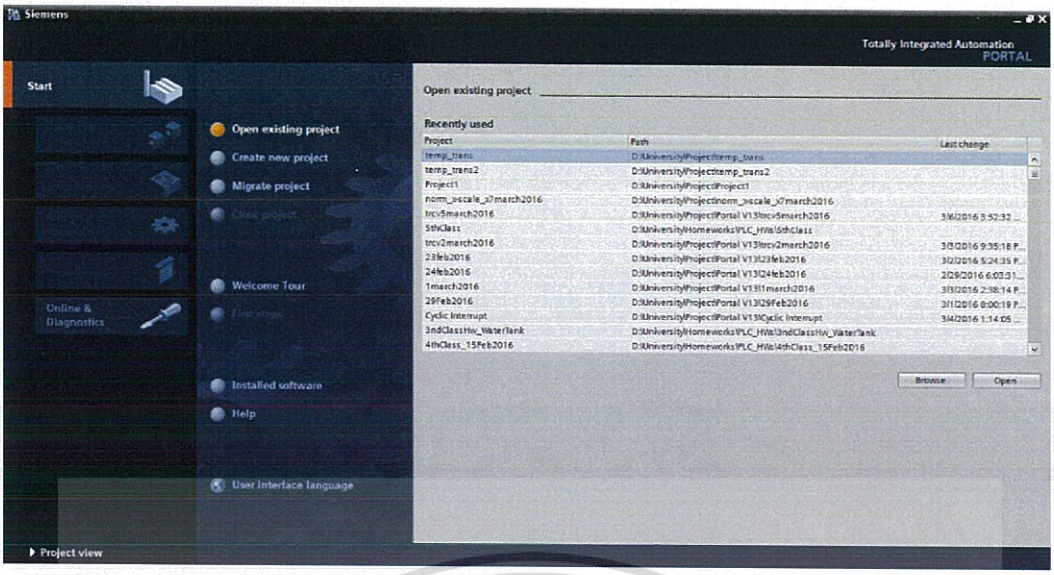
2.3 โปรแกรม STEP7 TIA Portal V13

โปรแกรม TIA Portal เป็นโปรแกรมสำหรับเขียนโปรแกรมสำหรับควบคุมพีแอลซีของบริษัท Siemens โดยเฉพาะ ซึ่งมีหลักการทำงานแบบอัตโนมัติเพื่อช่วยเหลือผู้ออกแบบระบบระหว่างการทำงาน ทำให้ผู้ออกแบบสามารถสร้างโปรเจคได้อย่างสะดวก และรวดเร็ว โดยในเวอร์ชัน 13 นั้น มี Function Block ให้เลือกใช้มากมายตามความต้องการของผู้ออกแบบระบบ และยังสามารถจำลองเครื่องพีแอลซีรวมถึงในส่วนของการเขียนโปรแกรม โปรแกรม TIA Portal รองรับการเขียนแบบ Ladder (LAD), Function Block Diagram (FBD), Structured Text (SCL), Statement List (STL) ซึ่งจะถูกเขียนในส่วน Program Blocks

Program Blocks แบ่งเป็น 4 ประเภท

1. Organization Block (OB) – คือ ส่วนหลักที่ใช้ในการรันโปรแกรม PLC
2. Function Block (FB) – เป็นบล็อกที่จะเก็บค่าจาก Data Block โดยถาวรมาเขียนเป็นโปรแกรม ซึ่งสามารถนำมาเรียกใช้ใน OB ได้
3. Function (FC) – ทำงานเหมือนกับ FB แต่ไม่เก็บค่าไว้ใน Data Block
4. Data Block (DB) – เป็นบล็อกสำหรับเก็บข้อมูลต่างๆ ไว้สำหรับการเขียนโปรแกรม เช่น ตัวแปร รวมถึงค่าเริ่มต้นของตัวแปรนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 หน้าต่างแสดงโปรแกรม TIA Portal V13

2.3.1 ภาษาที่ใช้ในการเขียนโปรแกรม TIA Portal V13

ภาษามาตรฐานในซอฟแวร์ TIA Portal V13 ที่ใช้ในการเขียนโปรแกรมมี 3 ภาษาต่อไปนี้

2.3.1.1 Ladder Logic (LAD)

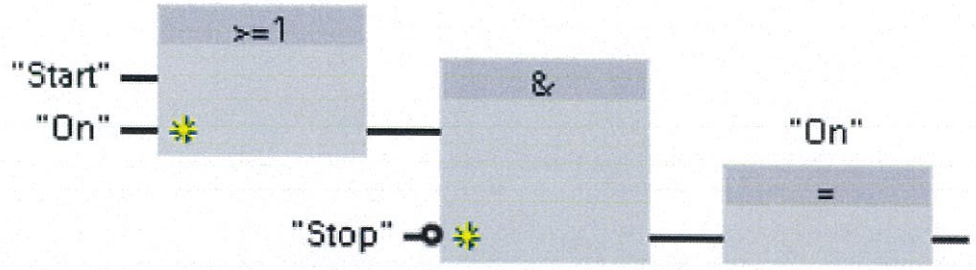
เป็นการเขียนโปรแกรมเชิงกราฟิก โดยจะแสดงผลออกมาคล้ายกับแผนผังการทำงานของวงจรรีเลย์ โดย LAD นี้จะมีฟังก์ชันบล็อกอื่นๆ ให้ เช่น ฟังก์ชันการคำนวณทางคณิตศาสตร์ (Math), ตัวจับเวลา (Timer), ตัวนับ (Counter) เป็นต้น

รูปที่ 2.5 รูปแบบโปรแกรม Ladder Logic (LAD)

2.3.1.2 Function Block Diagram (FBD)

รูปแบบของโปรแกรมเป็นการเขียนโปรแกรมเชิงกราฟิกเช่นเดียวกับการเขียนโปรแกรมแบบ Ladder Logic แต่การแสดงลอจิกต่างๆ จะใช้ในรูปแบบสัญลักษณ์ Boolean

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 รูปแบบการเขียนโปรแกรมแบบ Function Block Diagram (FBD)

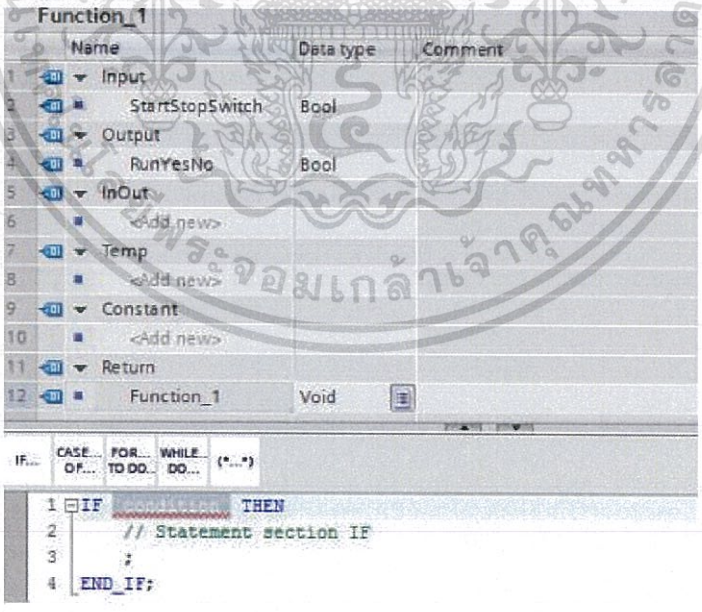
2.3.1.3 Structure Control Language (SCL)

เป็นรูปแบบการเขียนโปรแกรมขั้นสูง โดยใช้พื้นฐานความรู้ภาษาปาสคาล (Pascal Language)ในการเขียน

```
"C" := #A+#B;                          Assigns two local variables to a tag
"Data_block_1".Tag := #A;              Assignment to a data block tag
IF #A > #B THEN "C" := #A;             Condition for the IF-THEN statement
"C" := SQRT (SQR (#A) + SQR (#B));     Parameters for the SQRT instruction
```

รูปที่ 2.7 รูปแบบการเขียนโปรแกรมแบบ Structure Control Language (SCL)

นอกจากนี้ ซอฟแวร์ TIA Portal V13 ยังเปิดให้ผู้ใช้สามารถออกแบบรูปแบบของบล็อกโดยใช้ภาษาแบบ Structure Control Language (SCL) นี้ในการออกแบบอีกด้วย



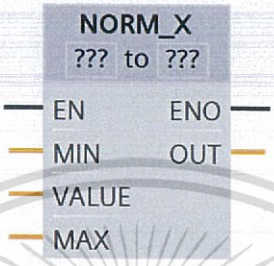
รูปที่ 2.8 หน้าเชื่อมต่อของโปรแกรมภาษา SCL ที่สามารถกำหนดตัวแปรต่างๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 บล็อกคำสั่งพื้นฐานที่เกี่ยวข้อง (Basic Instruction)

2.3.2.1 NORM_X

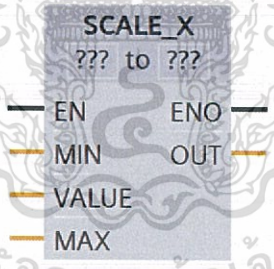
ใช้สำหรับสเกลค่าโวลต์ที่ได้จากแอนะล็อกขาเข้าที่ได้รับมาจากตัวเซนเซอร์ 0–5 โวลต์ หรือ จากค่าสเกลขาแอนะล็อกขาเข้า 0–13824 ออกเป็นค่าลิเนียร์ 0-1 โดยบล็อกสามารถรับข้อมูลประเภท Int, Dint, SInt, USInt, UInt, UDInt, Real และ LReal ส่วนขาออกสามารถเลือกให้ออกได้เป็นข้อมูลประเภท Real และ LReal



รูปที่ 2.9 ฟังก์ชันบล็อก NORM_X

2.3.2.2 SCALE_X

ใช้สำหรับสเกลค่าลิเนียร์ 0-1 ที่ได้จากขาออกของ NORM_X ให้เป็นช่วงที่ต้องการใช้งาน โดยรูปแบบของข้อมูลขาเข้าสามารถรับข้อมูลประเภท Real และ LReal ส่วนขาออกสามารถเลือกให้ออกได้เป็นข้อมูลประเภท Int, Dint, SInt, USInt, UInt, UDInt, Real และ LReal



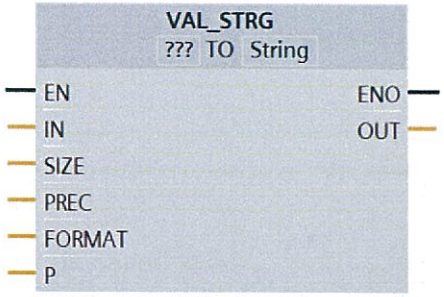
รูปที่ 2.10 ฟังก์ชันบล็อก SCALE_X

2.3.3 คำสั่งเพิ่มเติมที่เกี่ยวข้อง (Extended Instruction)

2.3.3.1 VAL_ STRG

ใช้สำหรับแปลงรูปแบบข้อมูล (Data Type) จาก USInt, SInt, UInt, Int, UDInt, DInt, Real หรือ LReal เป็น String

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 ฟังก์ชันบล็อก VAL_STRG

2.3.3.2 Strg_TO_Char

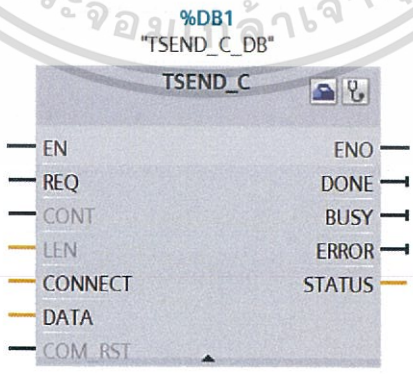
ใช้สำหรับแปลงรูปแบบข้อมูลจาก String เป็น Array of Chars

รูปที่ 2.12 ฟังก์ชันบล็อก Strg_TO_Char

2.3.4 บล็อกการเชื่อมต่อ (Communications)

2.3.4.1 TSEND_C

ใช้สำหรับส่งข้อมูลไปยังเครื่องที่ติดต่อผ่านโปรโตคอล TCP/IP ภายในบล็อกจะประกอบไปด้วย 3 บล็อก คือ TCON สำหรับการสร้างการเชื่อมต่อกับเครื่อง Partner, TSEND สำหรับการส่งข้อมูลไปยังเครื่อง Partner และ TDISCON สำหรับยกเลิกการเชื่อมต่อกับเครื่อง Partner

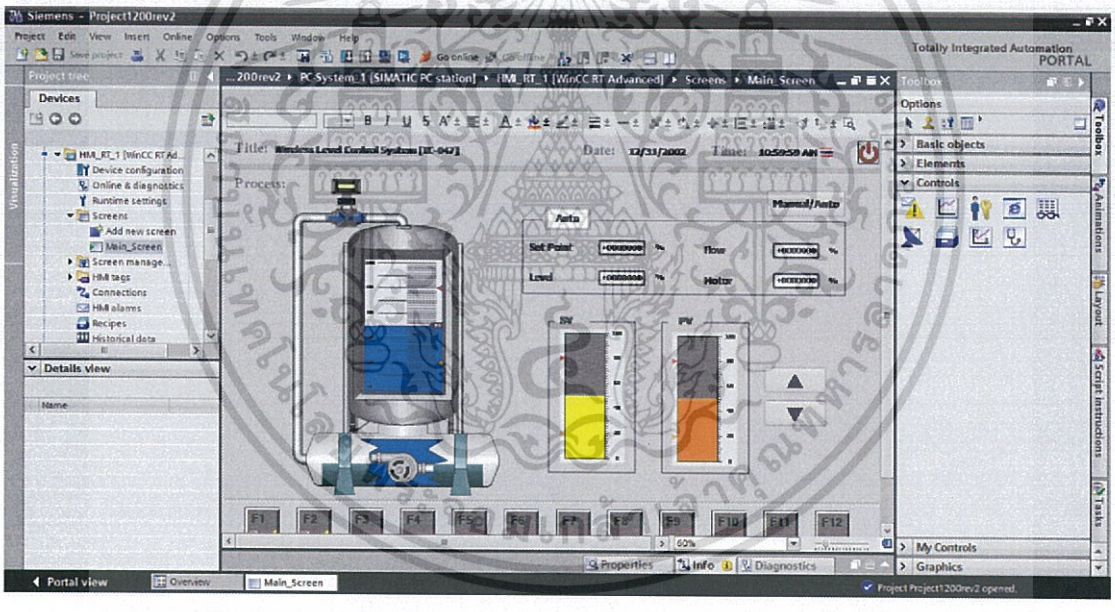


รูปที่ 2.13 ฟังก์ชันบล็อก TSEND_C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

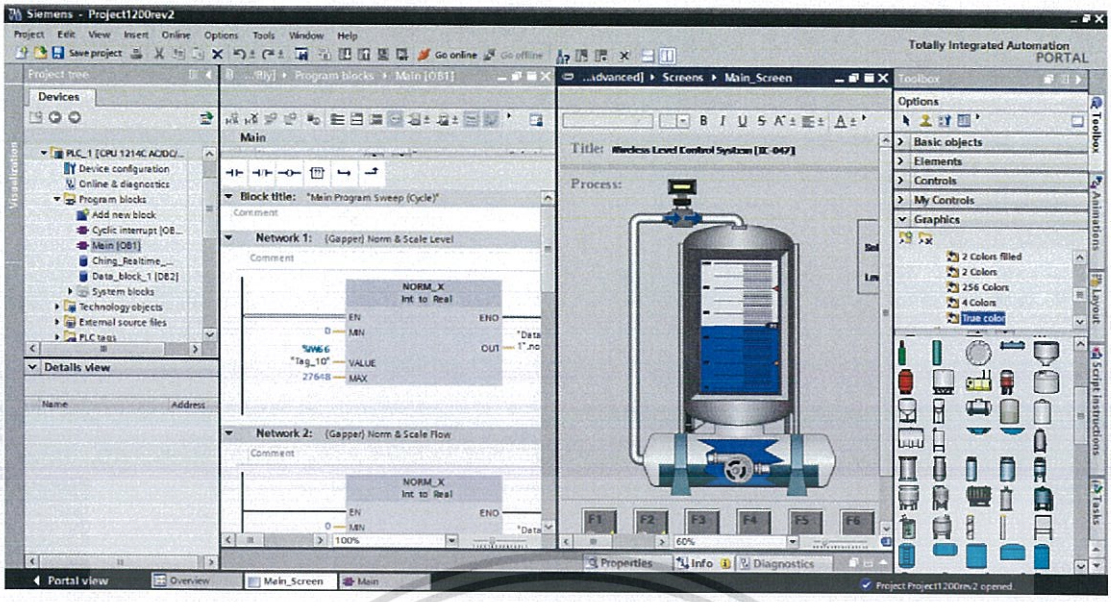
2.4 โปรแกรม WinCC Runtime Advanced V13

โปรแกรม WinCC Runtime Advanced V13 เป็นโปรแกรมของบริษัท Siemens ที่พัฒนาขึ้นมาภายใต้คอนเซ็ป "One Engineering Environment" – One Software Project For All Automation Tasks คือ หนึ่งโปรแกรมสำหรับงานอัตโนมัติทั้งหมด โดย WinCC Runtime Advanced V13 เป็นโปรแกรมที่เสริมมากับ STEP7 TIA Portal V13 ใช้สำหรับสร้างหน้าจอแสดงผลสำหรับติดตามค่า และควบคุมระบบสำหรับผู้ใช้งาน (Human Interface, HMI) เนื่องจากเป็นโปรแกรมเดียวกันทำให้ใช้งานง่าย และสะดวก เพราะสามารถเขียนโปรแกรมแลดเดอร์สำหรับควบคุมและสร้างหน้าเชื่อมต่อได้ภายในโปรแกรมเดียวพร้อมๆ กัน นอกจากนี้ภายในโปรแกรมยังมีรูปภาพกราฟิกที่ใช้ในกระบวนการต่างๆ ในโรงงานอุตสาหกรรมเป็นจำนวนมากที่สามารถเลือกใช้ได้ทันที หรือหากผู้ใช้ต้องการที่จะสร้างกราฟิกของตัวเองเพิ่มก็สามารถทำได้เช่นกัน โดยสามารถสร้างกราฟิกได้จากอุปกรณ์วาดพื้นฐานต่างๆ ภายในตัวโปรแกรม หรือสามารถสร้างจากโปรแกรมอื่นๆ เช่น Adobe Illustrator, Adobe Photoshop ฯลฯ แล้วอิมพอร์ตเข้ามาในโฟลเดอร์ My Graphics Folder ได้



รูปที่ 2.14 หน้าต่างโปรแกรม WinCC Runtime Advanced V13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

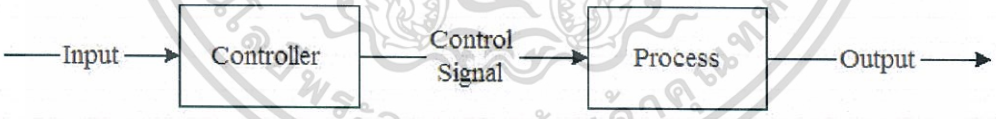


รูปที่ 2.15 หน้าต่าง WinCC Screen พร้อมกับหน้า Main Block สำหรับเขียนโปรแกรม

2.5 ประเภทการควบคุม

2.5.1 ระบบควบคุมแบบเปิด (Open-Loop Control System)

เป็นระบบควบคุมที่เอาต์พุตของระบบจะไม่มีผลต่อการควบคุมเลย กล่าวคือ เอาต์พุตของระบบจะไม่ถูกทำการวัดหรือป้อนกลับเพื่อที่จะนำมาเปรียบเทียบกับอินพุต ในทางปฏิบัติสามารถใช้การควบคุมแบบเปิดนี้ได้ ถ้าทราบถึงความสัมพันธ์ระหว่างอินพุตกับเอาต์พุตของระบบ เช่น การควบคุมการเปิด-ปิดไฟ สัญญาณไฟจราจร การควบคุมสายพานลำเลียง เป็นต้น

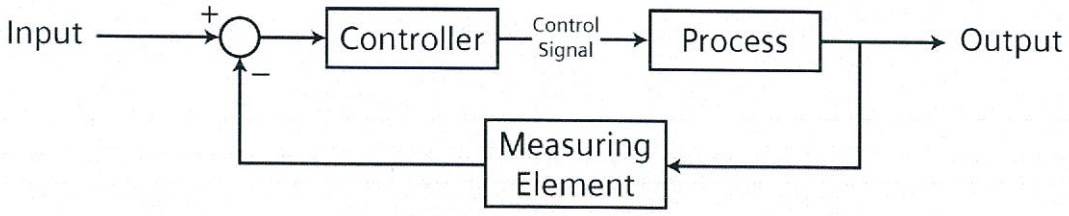


รูปที่ 2.16 บล็อกไดอะแกรมของระบบควบคุมแบบเปิด

2.5.2 การควบคุมแบบป้อนกลับ (Feedback Control)

เป็นระบบควบคุมที่เอาต์พุตของระบบมีผลโดยตรงกับสิ่งที่ต้องการควบคุม กล่าวคือ ระบบจะนำสัญญาณที่ได้จากเอาต์พุตมาป้อนกลับ เพื่อนำมาเปรียบเทียบกับอินพุตซ้ำๆ ในการควบคุมหรือเรียกว่า การควบคุมแบบป้อนกลับ โดยสัญญาณป้อนกลับนี้อาจจะเป็นสัญญาณเอาต์พุตโดยตรงหรือสัญญาณที่เป็นฟังก์ชันของสัญญาณเอาต์พุตหรือค่าอนุพันธ์ของสัญญาณเอาต์พุตก็ได้ ตัวอย่างการควบคุมแบบปิด ได้แก่ การควบคุมอุณหภูมิ การควบคุมความดัน การควบคุมอัตราการไหล ระบบควบคุมกระบวนการใช้ตู้เย็นที่ใช้ตามบ้านเรือน เป็นต้น

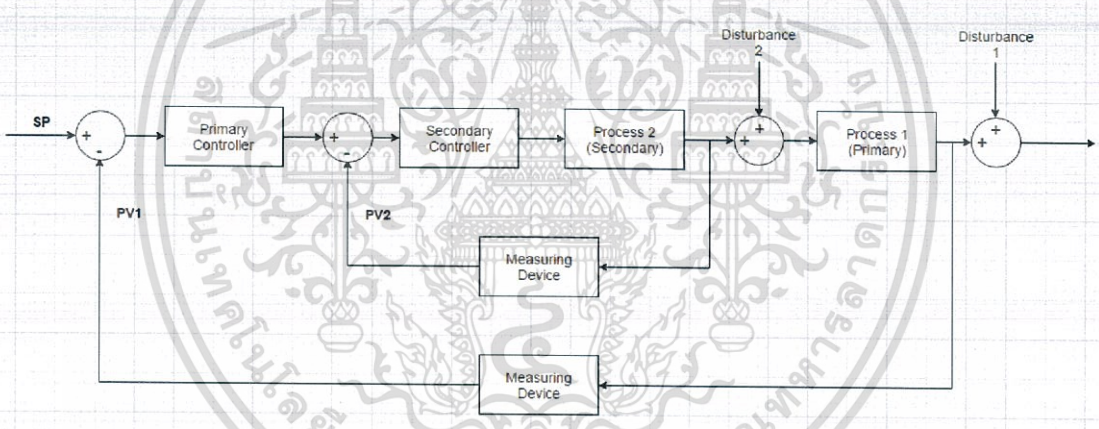
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 บล็อกไดอะแกรมของระบบควบคุมแบบป้อนกลับ

2.5.3 การควบคุมแบบแคสเคด (Cascade Control)

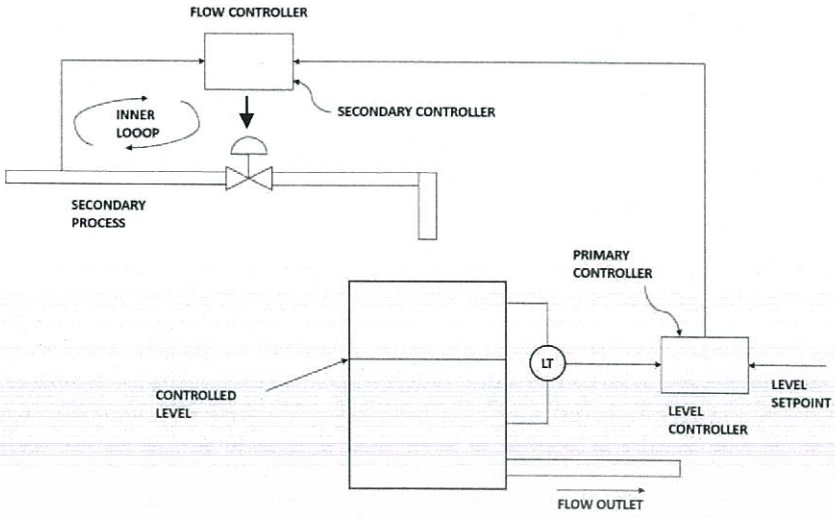
การควบคุมแบบแคสเคดโดยทั่วไปจะประกอบด้วย คอนโทรลเลอร์ 2 ตัวเซนเซอร์ 2 ตัว และตัวขับเร้า 1 ตัว นำระบบมาต่ออนุกรมกัน โดยให้ตัวที่ต้องการควบคุมเป็นหลัก โดยใช้คอนโทรลเลอร์ในการควบคุมทั้ง 2 ระบบ คอนโทรลเลอร์ที่ควบคุมระบบหลักจะเป็น Primary Controller (Master) ส่วนคอนโทรลเลอร์ที่ควบคุมระบบรองลงมาจะเป็น Secondary Controller (Slave) ซึ่งถึงแม้จะไม่มีก็สามารถควบคุมระบบโดยใช้ Master Controller เพียงตัวเดียวได้ โดยให้เอาต์พุตของ Master Controller เป็นค่าเป้าหมายของ Slave Controller



รูปที่ 2.18 บล็อกไดอะแกรมของการควบคุมแบบแคสเคด

การตัดระบบออกเป็นสองส่วน (Primary และ Secondary Process) จะทำให้ Process Lag ในกระบวนการรวมถูกตัดตอนออกเป็นสองส่วนไปด้วย ซึ่งจะเป็นผลให้การควบคุมทำได้ง่ายขึ้น เสมือนหนึ่งเป็นการลดขนาดของ Time Lag นั่นเอง โดย Time Lag แต่ละตัวจะถูกประมวลผลไปพร้อมๆ กัน ทำให้ Time Lag มีผลกระทบต่อการควบคุมลดลงไปด้วย การควบคุมแบบแคสเคดจึงเหมาะกับการควบคุมระบบที่มีการเปลี่ยนแปลงค่อนข้างช้าหรือระบบที่มี Time Constant มากๆ เช่น การควบคุมอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 ตัวอย่างการควบคุมระดับน้ำแบบแคสเคด

จากรูปที่ 2.19 เป็นการควบคุมระดับน้ำในถังน้ำขนาดใหญ่ โดยการควบคุมปริมาณน้ำที่ไหลผ่านวาล์วควบคุมมี Level Controller ทำหน้าที่เป็น Primary Controller วัดระดับน้ำเพื่อเปรียบเทียบกับค่าเป้าหมาย และส่งเอาต์พุตไปยัง Flow Controller ซึ่งทำหน้าที่เป็น Secondary Controller

จุดเด่นของการควบคุมในลักษณะนี้คือ เมื่อเกิดจากการเปลี่ยนแปลงของ Flow Rate ไปจากค่าเป้าหมายนั้น ตัว Secondary Controller จะทำการแก้ไขทันทีโดยที่ Level Error ยังไม่ทันจะเกิดขึ้น ทำให้การควบคุมระดับทำได้แม่นยำขึ้น ทั้งนี้เพราะว่าถังน้ำที่มีขนาดความจุสูงย่อมจะมี Process Lag ที่ยาวนาน และหากปล่อยให้ Level เปลี่ยนแปลงไป การควบคุมจะทำได้ยาก และใช้เวลานานขึ้น

จากที่กล่าวมาข้างต้นลูปในจะต้องมีความไวกว่าลูปนอก โดยระบบรองจะต้องตอบสนองต่อ Secondary Controller ไวกว่าระบบหลักตอบสนองต่อระบบหลักอย่างน้อย 4 เท่า เพื่อให้ลูปในมีเวลามากพอในการชดเชยผลจากการรบกวนก่อนที่จะส่งผลไปยังระบบหลัก

2.6 ระบบควบคุมแบบสัดส่วน-ปริพันธ์-อนุพันธ์ (Proportional-Integral-Derivative Controller, PID Controller)

การควบคุมแบบพีไอดีเป็นการควบคุมสัญญาณเอาต์พุตของกระบวนการ โดยใช้แบบสัดส่วนรวมกับแบบปริพันธ์ และแบบอนุพันธ์ (PID Controller) หรือที่เรียก “การควบคุมแบบ 3 เทอม” คือ Proportional (P) หมายถึงการปรับสัดส่วนสัญญาณ, Integral (I) หมายถึง การอินทิเกรตสัญญาณ และ Derivative (D) หมายถึง การอนุพันธ์สัญญาณ แสดงในรูปของสมการได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{Output} = K_P e + K_I \int_0^t e\,dt + K_D \frac{de}{dt} \qquad (2.1)$$

เมื่อ K_P = ค่าคงที่แบบสัดส่วน

K_I = ค่าคงที่ของการอินทิกรัล

K_D = ค่าคงที่ของการอนุพันธ์

$\int_0^t e\,dt$ = การอินทิกรัลค่าความคลาดเคลื่อนในช่วงคาบเวลา

$\frac{de}{dt}$ = อัตราการเปลี่ยนแปลงของความคลาดเคลื่อนเทียบเวลา

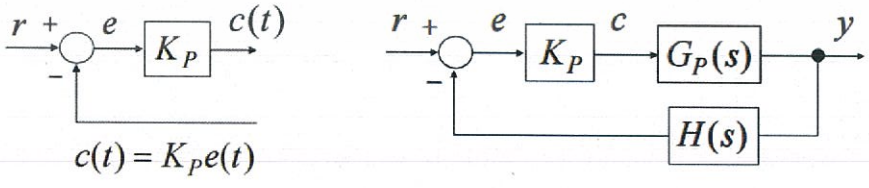
e = ค่าความคลาดเคลื่อนของกระบวนการ โดย e = SV – PV

เมื่อ SV คือ ค่าเป้าหมาย และ PV คือ ค่ากระบวนการ

ในสมการพีไอดีโปรแกรมจะนำค่าความผิดพลาด (Error) ของกระบวนการที่ได้จากผลต่างของค่า SV กับค่า PV ที่มาจากอุปกรณ์ตรวจวัดมาเป็นตัวแปรในสมการ และทำการส่งสัญญาณแรงดันเอาต์พุต ให้เป็นไปตามสมการพีไอดี และค่าตัวแปร K_P, K_I, K_D ที่ป้อนค่าไว้โปรแกรมจะปรับค่าแรงดันเอาต์พุตจนกว่าค่า SV จะเท่ากับค่า PV โปรแกรมจะรักษาแรงดันเอาต์พุตไว้เพื่อให้ได้ค่าของกระบวนการตามที่ต้องการ

2.6.1 ตัวควบคุมแบบสัดส่วน (Proportional Controller)

ลักษณะการทำงานเป็นการส่งสัญญาณเอาต์พุตออกมาเป็นสัดส่วนโดยตรงกับสัญญาณค่าความผิดพลาด บล็อกไดอะแกรม และฟังก์ชันการทำงานเป็นดังรูปที่ 2.20(ก) ตัวควบคุมแบบนี้จะนำเอาค่าความผิดพลาดระหว่างสัญญาณอ้างอิงกับสัญญาณเอาต์พุตของตัวควบคุม แล้วตัวควบคุมจะทำการสร้างสัญญาณเอาต์พุตด้วยการขยายสัญญาณความผิดพลาดดังกล่าวด้วยค่าอัตราขยายของตัวควบคุม บล็อกไดอะแกรม และลักษณะของการประมวลผลสัญญาณเป็นดังรูปที่ 2.20(ข)



(ก) (ข)

รูปที่ 2.20 ตัวควบคุมแบบสัดส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

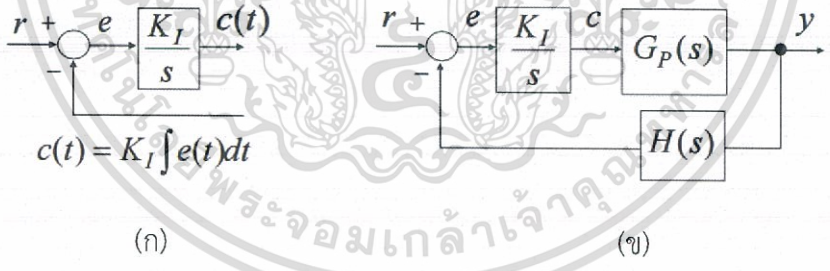
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ถ้านำการควบคุมแบบสัดส่วนไปใช้กับระบบชนิด 0 (System Type 0) ตัวควบคุมแบบนี้จะไม่สามารถขจัดค่าความผิดพลาดในสภาวะคงตัวได้ แต่ก็สามารถทำให้ค่าผิดพลาดดังกล่าวมีค่าน้อยลงด้วยการปรับค่า K_P ให้สูง ค่า K_P ที่สูง จะเป็นผลให้ค่าเอาต์พุตมากขึ้นตาม (ระบบมีผลตอบสนองเร็วขึ้น) แต่หากค่า K_P มากเกินไประบบอาจจะไม่เสถียรได้ อาจทำให้ระบบมีค่าพุ่งเกินสูง ซึ่งอาจเป็นอันตรายต่อระบบได้ ในทางตรงกันข้ามหากค่า K_P น้อยเกินไป อาจทำให้ระบบตอบสนองช้าเกินไป หรืออาจกล่าวได้ว่า ระบบไม่สามารถสู้กับสิ่งรบกวนที่เกิดได้ ตัวอย่างสถานการณ์เช่น การเร่งรอบเครื่องยนต์ เมื่อมีโหลดมากระทำกับเครื่องยนต์มากขึ้น ตัวคอนโทรลเลอร์จะต้องเร่งรอบเครื่องเพิ่ม เพื่อให้สามารถรองรับการกระทำของโหลดที่เพิ่มมาได้ โดยหากคอนโทรลเลอร์เร่งรอบเครื่องช้าอาจส่งผลให้ความเร็วรอบของเครื่องตกลง และส่งผลให้เครื่องดับหรือระบบหยุดทำงานในที่สุด

2.6.2 ตัวควบคุมแบบปริพันธ์ (Integral Controller)

ลักษณะการทำงานเป็นการส่งสัญญาณเอาต์พุตออกมาจากการอินทิเกรตสัญญาณค่าความผิดพลาด บล็อกไดอะแกรม และฟังก์ชันการทำงานเป็นดังรูปที่ 2.21(ก)

ตัวควบคุมแบบปริพันธ์จะนำเอาสัญญาณความผิดพลาดระหว่างสัญญาณอ้างอิงกับสัญญาณเอาต์พุตของตัวควบคุม แล้วตัวควบคุมจะทำการสร้างสัญญาณเอาต์พุตด้วยการอินทิเกรตสัญญาณความผิดพลาดดังกล่าวแล้วคูณด้วยค่าอัตราขยายของตัวควบคุม บล็อกไดอะแกรม และลักษณะของการประมวลผลสัญญาณเป็นดังรูปที่ 2.21(ข)



(ก) (ข)

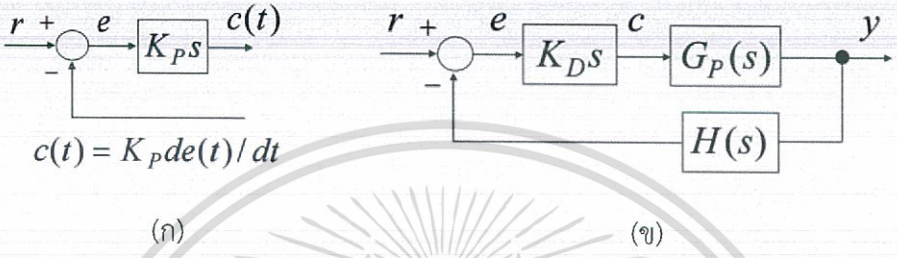
รูปที่ 2.21 ตัวควบคุมแบบปริพันธ์

จุดเด่นของตัวควบคุมแบบนี้คือ ถ้านำไปใช้กับระบบชนิด 0 (System Type 0) ตัวควบคุมแบบนี้จะสามารถขจัดค่าความผิดพลาดในสภาวะคงตัวได้ ข้อด้อยที่อาจเกิดขึ้นในการนำไปใช้งานคือตัวควบคุมแบบนี้ไม่สามารถลดผลของการพุ่งเกินของผลตอบสนองได้ และการปรับอัตราขยายให้มีค่าสูง อาจทำให้ผลตอบสนองที่ไม่พึงประสงค์ เช่น การปรับค่าเกนให้สูงขึ้น อาจมีผลทำให้ผลตอบสนองของระบบเกิดการแกว่งตัวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
144415

2.6.3 ตัวควบคุมแบบอนุพันธ์ (Derivative Controller)

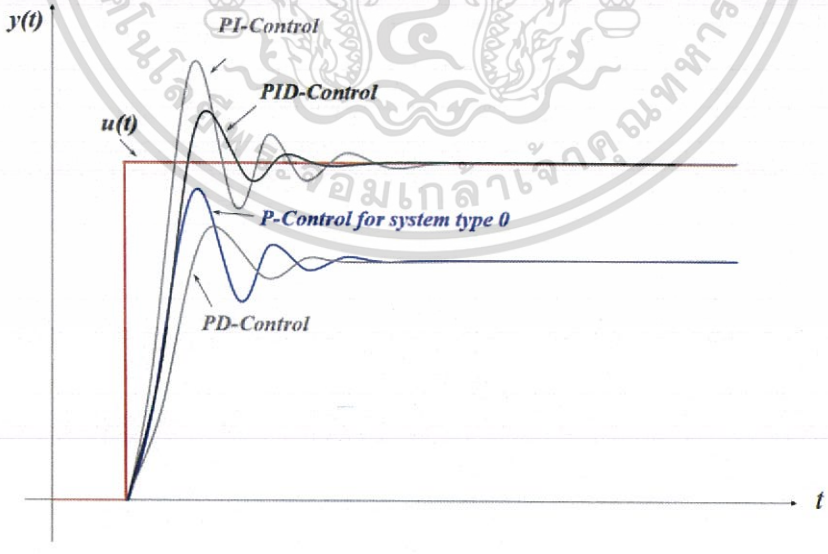
ลักษณะการทำงานเป็นการส่งสัญญาณเอาต์พุตมาจากการอนุพันธ์ค่าความผิดพลาด บล็อกไดอะแกรม และฟังก์ชันการทำงานเป็นดังรูปที่ 2.22(ก) ตัวควบคุมแบบนี้จะนำเอาค่าสัญญาณความผิดพลาดระหว่างสัญญาณอ้างอิงกับสัญญาณเอาต์พุตมาเป็นอินพุตของตัวควบคุม แล้วตัวควบคุมจะทำการสร้างสัญญาณเอาต์พุตด้วยการอนุพันธ์สัญญาณความผิดพลาดดังกล่าว แล้วคูณด้วยค่าอัตราขยายของตัวควบคุม บล็อกไดอะแกรม และลักษณะของการประมวลผลสัญญาณเป็นดังรูปที่ 2.22(ข)



(ก) (ข)

รูปที่ 2.22 ตัวควบคุมแบบอนุพันธ์

จุดเด่นของการควบคุมแบบนี้คือ ตัวควบคุมแบบนี้ใช้สำหรับลดผลของค่าพุ่งเกินของผลตอบสนองได้ ลดผลตอบสนองที่มีการเปลี่ยนแปลงไปมาได้ แต่ต้องปรับค่าอัตราขยายให้เหมาะสมด้วย ไม่เช่นนั้นอาจจะทำให้ได้ผลตอบสนองที่ไม่เป็นที่พึงประสงค์ ปัญหาที่อาจเกิดขึ้นในการนำตัวควบคุมแบบนี้ไปใช้คือ ตัวควบคุมแบบนี้จะไม่สามารถขจัดค่าความผิดพลาดในสภาวะคงตัวได้ และการใช้ตัวควบคุมนี้อาจจะทำให้ได้ผลตอบสนองที่ช้าลงได้



รูปที่ 2.23 ผลตอบสนองต่ออินพุตแบบขั้นบันไดในการใช้งานระบบควบคุมพีไอดีในลักษณะต่างๆ กัน

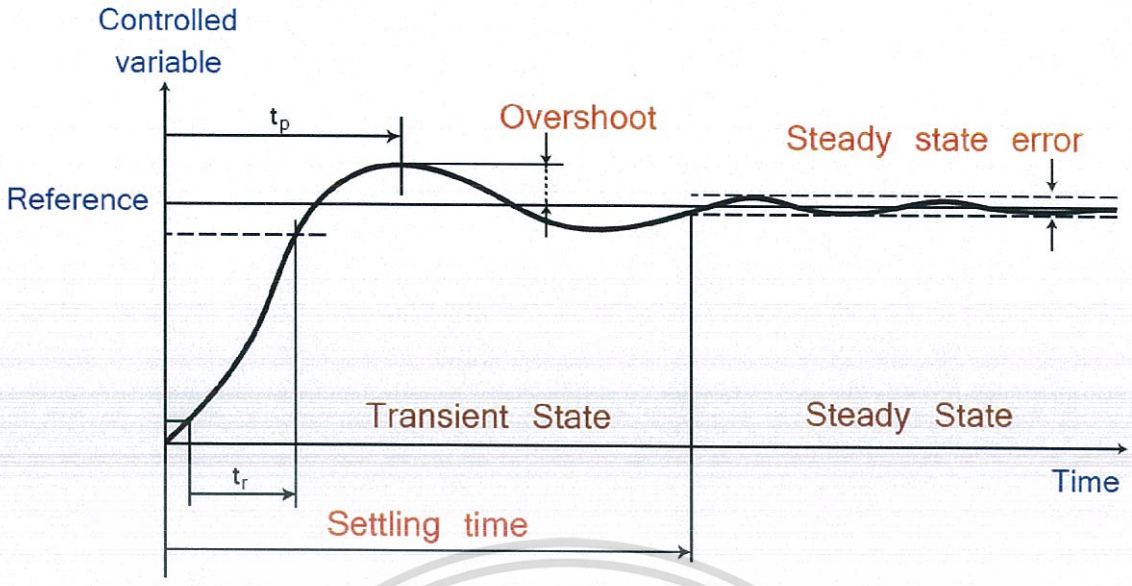
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.23 จะเห็นว่าถ้าหากนำตัวควบคุมแบบสัดส่วนไปใช้กับระบบอันดับหนึ่ง ผลตอบสนองที่ได้จะมีค่าความผิดพลาดในสภาวะคงตัว ซึ่งสามารถลดผลกระทบได้โดยการเพิ่มค่าอัตราขยายของตัวควบคุมให้สูงขึ้น สำหรับการนำไปใช้กับระบบที่มีอันดับสูงกว่า และเป็นระบบชนิด 0 ค่าความผิดพลาดในสภาวะคงตัวก็จะยังมีอยู่ และการลดผลกระทบด้วยการปรับค่าอัตราขยายของตัวควบคุมแบบสัดส่วน อาจจะส่งผลให้การพุ่งเกินมีค่าสูงขึ้นได้ ถ้าหากใช้ตัวควบคุมร่วมกันระหว่างตัวควบคุมแบบสัดส่วนกับแบบปริพันธ์ (PI Controller) ตัวควบคุมแบบปริพันธ์จะช่วยขจัดค่าความผิดพลาดในสภาวะคงตัว แต่ผลตอบสนองที่ได้จะยังมีค่าพุ่งเกินเหมือนเดิม ถ้าหากใช้ตัวควบคุมร่วมกันระหว่างแบบสัดส่วนกับแบบอนุพันธ์ (PD Controller) การพุ่งเกินของผลตอบสนองจะลดลง แต่ค่าความผิดพลาดในสภาวะคงตัวจะยังคงอยู่ ดังนั้นหากใช้ตัวควบคุมแบบสัดส่วน ปริพันธ์ และอนุพันธ์ร่วมกัน (PID Controller) ด้วยการปรับค่าอัตราขยายให้เหมาะสมกับระบบนั้นๆ ก็จะได้ผลตอบสนองแบบหน่วงต่ำกว่าวิกฤตที่มีค่าพุ่งเกินเหมาะกับระบบนั้นๆ

ตารางที่ 2.2 ผลของการเพิ่มค่าพารามิเตอร์ของพีไอดีคอนโทรลเลอร์

ผลของการเพิ่มค่าพารามิเตอร์ของพีไอดีคอนโทรลเลอร์				
Parameter	Rise time	Overshoot	Settling time	Steady-state Error
K_P	ลดลง	เพิ่มขึ้น	เปลี่ยนเล็กน้อย	ลดลง
K_I	ลดลง	เพิ่มขึ้น	เพิ่มขึ้น	ถูกกำจัด
K_D	นานมาก (ลดลงหรือเพิ่มขึ้นน้อยมาก)	ลดลง	ลดลง	ไม่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

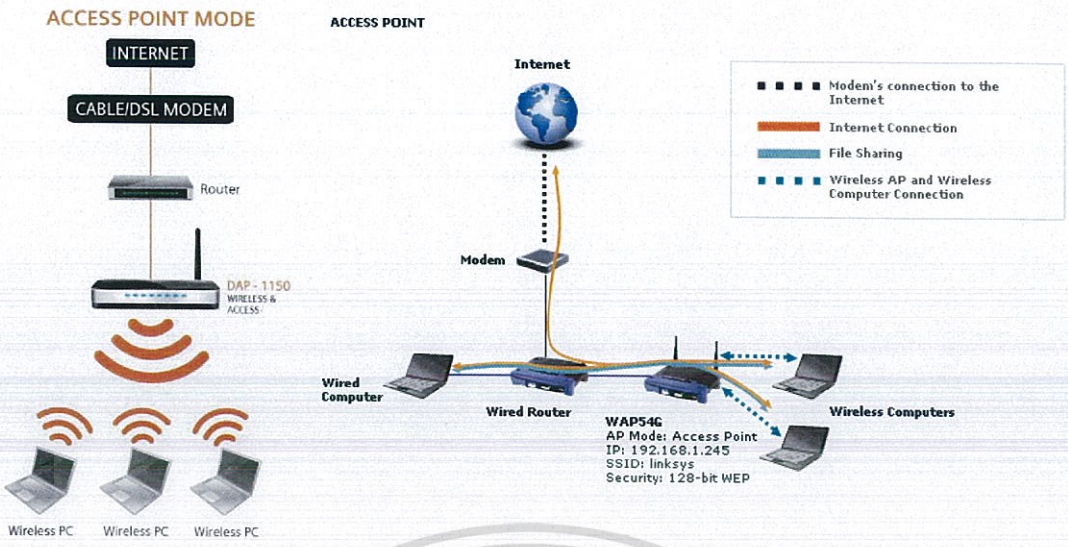


รูปที่ 2.24 ค่าพารามิเตอร์ของระบบควบคุมแบบพีไอดี

2.7 เทคโนโลยี WiFi

Wireless-Fidelity หรือ WiFi เป็นชุดผลิตภัณฑ์ใดๆ ที่สามารถทำงานได้ตามมาตรฐานเครือข่ายคอมพิวเตอร์แบบไร้สาย (แลนไร้สาย) ซึ่งอยู่บนมาตรฐาน IEEE 802.11 ซึ่งเป็นมาตรฐานที่ถูกอนุมัติให้ใช้จาก IEEE (The Institute of Electrical and Electronics Engineers) เพื่อให้อุปกรณ์คอมพิวเตอร์สามารถติดต่อสื่อสารกันได้บนมาตรฐานการทำงานแบบเดียวกัน กล่าวคือ เป็นเทคโนโลยีที่ช่วยให้อุปกรณ์อิเล็กทรอนิกส์สามารถแลกเปลี่ยนข้อมูลหรือการเชื่อมต่ออินเทอร์เน็ตแบบไร้สายโดยใช้คลื่นวิทยุ และคลื่นความถี่อินฟาเรด (Infrared) ในการรับส่งข้อมูล โดยคลื่นความถี่วิทยุสามารถทะลุกำแพงหรือสิ่งกีดขวางได้ ทำให้การใช้งานบนเครือข่ายไร้สายมีความคล่องตัว และสะดวกสบายมากขึ้น สามารถเชื่อมต่อเข้าสู่เครือข่ายได้ทุกที่ที่มีคลื่นสัญญาณ ซึ่งข้อมูลจะถูกรับส่งผ่านคลื่นวิทยุความถี่ 2.4 GHz ด้วยความเร็ว 11 Mbps ระยะห่างประมาณ 300 ฟุต เดิมที WiFi ออกแบบมาใช้สำหรับอุปกรณ์พกพาต่างๆ และใช้เครือข่าย LAN เท่านั้น แต่ปัจจุบันนิยมใช้ WiFi เพื่อต่อกับอินเทอร์เน็ต โดยอุปกรณ์พกพาต่างๆ สามารถเชื่อมต่อกับอินเทอร์เน็ตได้ผ่านอุปกรณ์ที่เรียกว่า แอคเซสพอยต์ (Access Point, AP) หรือ ฮอตสปอต (Hotspot) และบริเวณที่ระยะทำการของแอคเซสพอยต์ครอบคลุมอยู่ที่ประมาณ 20 ม. ในอาคาร แต่ระยะนี้จะไกลกว่าถ้าเป็นที่โล่งแจ้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 แบบจำลองการเชื่อมต่อ WiFi แบบโหมดแอคเซสพอยต์

รูปที่ 2.26 พื้นที่การกระจายสัญญาณ WiFi

[ที่มา : http://www.netronics-networks.com/wifi_coverage.html]

2.7.1 การรบกวน

การเชื่อมต่อ WiFi สามารถจะหยุดชะงักหรืออินเทอร์เน็ตมีความเร็วลดลงอันเนื่องมาจากอุปกรณ์อื่นๆ ในพื้นที่เดียวกัน หลายๆ แอคเซสพอยต์ที่ใช้มาตรฐาน 802.11b และ 802.11g ที่ 2.4 GHz มีค่าเริ่มต้นที่เป็นแชนแนลเดียวกัน นำไปสู่ความแออัดในบางแชนแนล WiFi ขยะหรือจำนวนแอคเซสพอยต์ที่มากเกินไปในพื้นที่ โดยเฉพาะอย่างยิ่งในแชนแนลข้างเคียง สามารถกีดขวางการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เข้าถึง และแทรกแซงการใช้แอคเซสพอยต์ของอุปกรณ์อื่นๆ สาเหตุจากการซ้อนทับกันของแชนแนลในแถบความถี่ของ 802.11g/b รวมทั้งมีการลดลงของอัตราส่วนสัญญาณต่อคลื่นรบกวน SNR ระหว่างแอคเซสพอยต์ด้วยกัน สิ่งนี้จะกลายเป็นปัญหาในพื้นที่ที่มีความหนาแน่นสูง เช่น อพาร์ตเมนต์ คอมเพล็กซ์ หรืออาคารสำนักงานขนาดใหญ่ที่มีหลาย WiFi แอคเซสพอยต์

นอกจากนี้อุปกรณ์อื่นๆ ที่ใช้แถบความถี่ 2.4 GHz เช่น เตาอบไมโครเวฟ อุปกรณ์ ISM กล้องรักษาความปลอดภัย อุปกรณ์ ZigBee อุปกรณ์บลูทูธ ผู้ส่ง วิดีโอ โทรศัพท์ไร้สาย เครื่องมอนิเตอร์ทารก และ (ในบางประเทศ) วิทยุสมัครเล่น ทั้งหมดที่สามารถก่อให้เกิดการรบกวนเพิ่มเติมอย่างมีนัยสำคัญ นอกจากนี้ยังเป็นปัญหาเมื่อหลายๆ เทศบาลหรือหลายๆ องค์กรขนาดใหญ่อื่นๆ (เช่น มหาวิทยาลัย) พยายามที่จะให้ครอบคลุมพื้นที่ขนาดใหญ่ และเกิดการทับซ้อนกัน [ที่มา : https://th.wikipedia.org/ wiki/ไวไฟ]

2.7.2 มาตรฐานของเทคโนโลยี WiFi

2.7.2.1 IEEE 802.11 b

ถูกเผยแพร่เมื่อปี พ.ศ. 2542 ซึ่งเป็นที่รู้จักกันดี และนิยมใช้กันอย่างแพร่หลาย โดยใช้เทคโนโลยีที่เรียกว่า CCK (Complimentary Code Keying) ผนวกกับ DSSS (Direct Sequence Spread Spectrum) สามารถรับส่งข้อมูลด้วยความเร็วสูงสุดที่ 11 Mbps ผ่านคลื่นวิทยุความถี่ 2.4 GHz ซึ่งอุปกรณ์ต่างๆ ที่ใช้คลื่นความถี่นี้ เช่น โทรศัพท์ไร้สาย, Bluetooth เป็นต้น โดยมีเครื่องหมายการค้าที่รู้จักกันดีในนาม WiFi ซึ่งสามารถใช้ร่วมกับอุปกรณ์ต่างๆ ที่มีเครื่องหมาย WiFi ได้

2.7.2.2 IEEE 802.11 a

ถูกเผยแพร่เมื่อปี พ.ศ. 2542 ใช้เทคโนโลยีที่เรียกว่า OFDM (Orthogonal Frequency Division Multiplexing) สามารถรับส่งข้อมูลได้ด้วยความเร็วสูงสุดที่ 54 Mbps แต่จะใช้คลื่นวิทยุที่ความถี่ 5 GHz ซึ่งเป็นย่านความถี่สาธารณะสำหรับใช้งานในประเทศสหรัฐอเมริกา ซึ่งมีข้อเสีย คือ ในบางประเทศย่านความถี่ดังกล่าวไม่สามารถนำมาใช้งานได้อย่างสาธารณะ โดยในประเทศไทยไม่อนุญาตให้มีการใช้งานเนื่องจากความถี่ย่าน 5 GHz ได้ถูกใช้สำหรับกิจการอื่นแล้ว และรัศมีของสัญญาณมีขนาดค่อนข้างสั้น คือ ประมาณ 30 เมตร จึงไม่เป็นที่นิยม

2.7.2.3 IEEE 802.11 g

ถูกเผยแพร่เมื่อกลางปี พ.ศ. 2546 ซึ่งเป็นมาตรฐานที่จะเข้ามาทดแทนมาตรฐาน IEEE 802.11 b โดยมีการนำเทคโนโลยี OFDM มาประยุกต์ใช้ในช่องสัญญาณวิทยุความถี่ 2.4 GHz ซึ่งอุปกรณ์ IEEE 802.11 g WLAN มีความสามารถในการรับส่งข้อมูลด้วยความเร็วสูงสุดที่ 54 Mbps และมีรัศมีสัญญาณของอุปกรณ์อยู่ระหว่างรัศมีสัญญาณของอุปกรณ์ IEEE 802.11 a และ IEEE 802.11 b เนื่องจากความถี่ 2.4 GHz เป็นย่านความถี่สาธารณะสากล และสามารถที่จะใช้งานร่วมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับอุปกรณ์ที่ใช้มาตรฐาน IEEE 802.11 b ได้ จึงทำให้ไม่จำเป็นต้องเปลี่ยนแปลงโครงสร้างพื้นฐานของระบบไร้สายที่ใช้กันอยู่ เพียงแค่อัพเกรดเฟิร์มแวร์หรือใช้งานกับแอคเซสพอยต์เลยก็ทำได้เช่นเดียวกัน ดังนั้นจึงมีแนวโน้มสูงว่าอุปกรณ์ IEEE 802.11 g WLAN จะได้รับความนิยมอย่างแพร่หลายหากมีราคาไม่แพงจนเกินไป

2.8 โปรโตคอล TCP/IP

โปรโตคอล TCP/IP (Transmission Control Protocol/Internet Protocol) เป็นโปรโตคอลที่ใช้ในการติดต่อสื่อสารในระบบอินเทอร์เน็ต โดยมีการระบุผู้รับผู้ส่งในเครือข่าย และจัดการแบ่งข้อมูลเป็นชิ้นเล็กๆ ที่เรียกว่าแพ็กเกจ (Package) ส่งผ่านไปในอินเทอร์เน็ต โดยข้อมูลที่ส่งไปจะได้รับอย่างถูกต้อง และครบถ้วน ในกรณีที่ข้อมูลเกิดข้อผิดพลาดระหว่างทางจะมีการร้องขอเพื่อให้ส่งข้อมูลใหม่

โครงสร้างแบบ TCP/IP (TCP/IP Model) เป็นมาตรฐานที่ทำให้คอมพิวเตอร์ภายในระบบเครือข่ายอินเทอร์เน็ตสามารถเชื่อมต่อ และติดต่อสื่อสารแลกเปลี่ยนข้อมูลกันได้ เป็นมาตรฐานที่ว่าด้วยการกำหนดวิธีการติดต่อสื่อสารระหว่างคอมพิวเตอร์ โดยใช้แนวคิดของการแบ่งลำดับชั้นโปรโตคอล

โปรโตคอล TCP/IP ทำงานโดยแบ่งชั้นเทียบกับ OSI Model โดยกลไกในการทำงานของโปรโตคอล TCP/IP มี 4 ชั้น ซึ่งในชั้นแรก คือ Process Layer จะทำหน้าที่ติดต่อกับแอพพลิเคชันและโปรโตคอลที่แอพพลิเคชันนั้นๆ ใช้งาน แล้วส่งต่อมาให้ชั้น Host-to-Host Layer เพื่อติดต่อสื่อสารกันระหว่างเครื่องเซิร์ฟเวอร์ให้บริการกับเครื่องผู้ขอใช้บริการ ในชั้นนี้จะมีการสร้าง Session หรือการเชื่อมต่อระหว่างระบบขั้นตามแต่ละโปรโตคอลที่ต้องการ ต่อมาเป็นการผนึกข้อมูลไปเป็น IP Datagram ที่ชั้น Internetwork Layer โดยอาศัยโปรโตคอล IP ในการติดต่อส่งข้อมูลข้ามเครือข่ายไปยังเครือข่าย และเครื่องที่ถูกต้องได้ ชั้นสุดท้าย คือ การส่งข้อมูลออกสู่โลกภายนอก ซึ่งจะต้องอาศัยกลไกในชั้น Network Interface Layer เพื่อแปลงข้อมูลใหม่ โดยจะเพิ่มข้อมูลที่จำเป็นในการอ้างอิงตำแหน่ง และแปลงข้อมูลเป็นสัญญาณไฟฟ้าส่งออกไปยังเครือข่าย และอาจจะออกไปยังเกตเวย์หรือเราต์เตอร์เพื่อข้ามเครือข่ายออกไปยังเส้นทางที่กำหนดไว้ในอินเทอร์เน็ตต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ขั้นตอนการดำเนินงาน

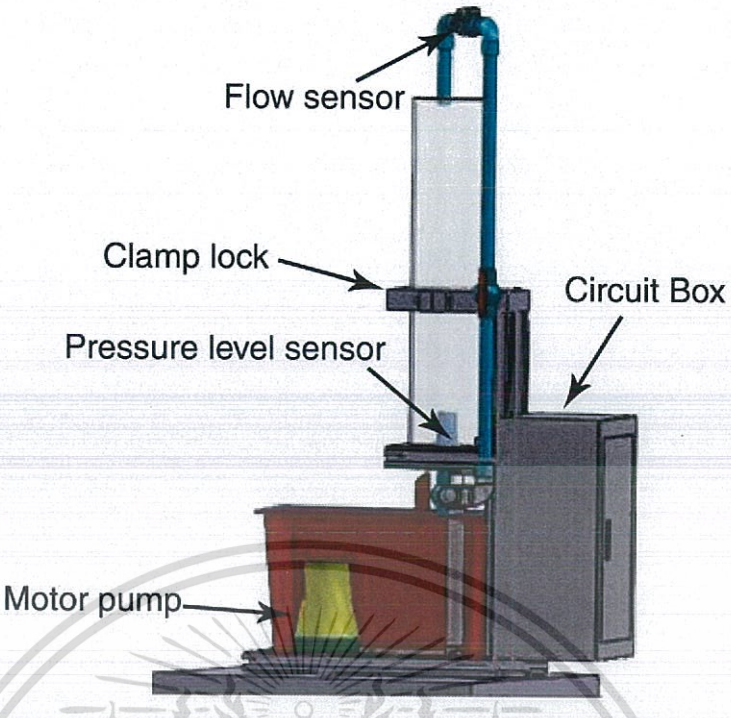
3.1 โครงสร้างของระบบโดยรวม

ระบบควบคุมระดับน้ำอัตโนมัติแบบไร้สายแบ่งออกเป็น 5 ส่วน ดังนี้

1. โมเดลจำลองระบบ
2. Actuator มอเตอร์ปั๊มน้ำสำหรับปั๊มน้ำเข้าถังโดยสามารถปรับความเร็วของมอเตอร์ปั๊มน้ำได้
3. Level Sensor สำหรับวัดระดับน้ำภายในถัง
4. Flow Sensor สำหรับวัดอัตราการไหลของน้ำภายในท่อพีวีซี
5. โปรแกรมพีแอลซีสำหรับควบคุมระบบ

ในส่วนของโครงสร้างโมเดลจำลองระบบออกแบบโดยโปรแกรม SolidWorks ส่วนของตัว Clamp Lock จะขึ้นรูปโดยการพิมพ์ 3 มิติ จากโปรแกรม MakerWare ใช้สำหรับยึดตัวถังกับส่วนโครง Aluminum profile ให้ถังไม่ขยับหรือเลื่อน ในส่วนของโปรแกรมพีแอลซีสำหรับควบคุมระบบที่ผู้จัดทำรับผิดชอบได้ใช้โปรแกรม STEP7 TIA Portal V13 ในการเขียนแลดเดอร์ไดอะแกรม โดยใช้บล็อกลูปพีไอดีในการควบคุมระดับน้ำตามค่าเป้าหมายที่เราต้องการโดยการควบคุมความเร็วมอเตอร์ปั๊มน้ำ ซึ่งสั่งการโดยบล็อก TSEND_C ตามค่าเอาต์พุตที่ได้จากการประมวลผลแบบป้อนกลับของลูปพีไอดี นอกจากนี้ผู้จัดทำได้ใช้โปรแกรม WinCC Runtime Advanced V13 ในการสร้างหน้าต่างสำหรับการติดตาม และควบคุมระบบ เพื่อให้สามารถติดตาม และควบคุมระบบได้ผ่านหน้าจอคอมพิวเตอร์ นอกจากนี้ยังได้สร้างหน้าต่างสำหรับการแสดงกราฟผลตอบสนองที่ได้จากลูปพีไอดีอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

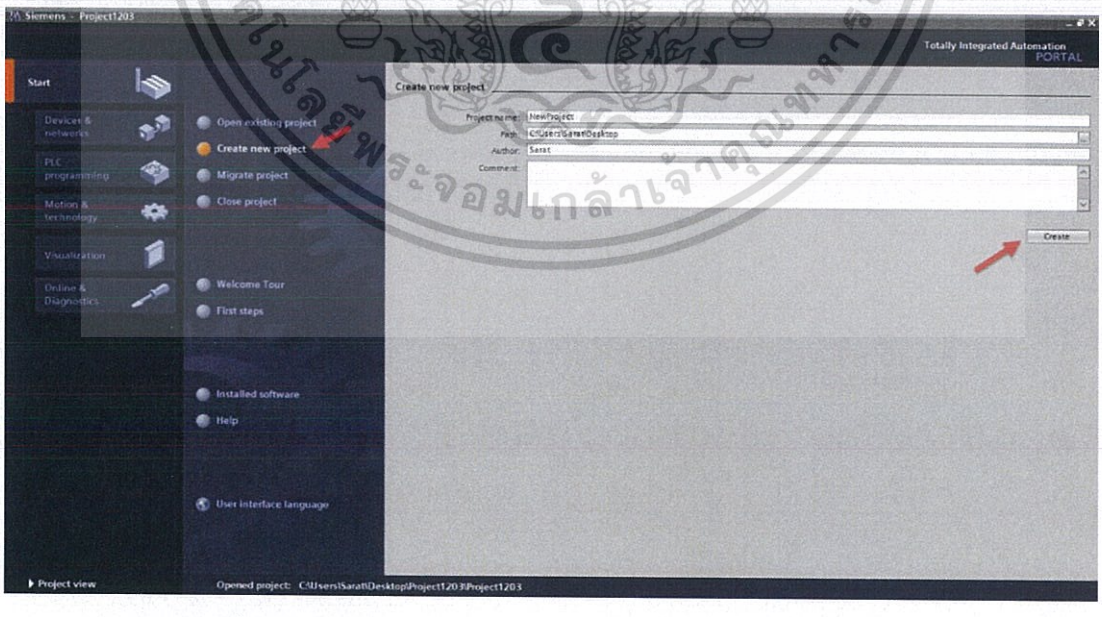


รูปที่ 3.1 จำลองโครงสร้างระบบด้วยโปรแกรม SolidWorks

3.2 โปรแกรม TIA Portal V13

3.2.1 สร้างโปรเจค และตั้งค่าอุปกรณ์

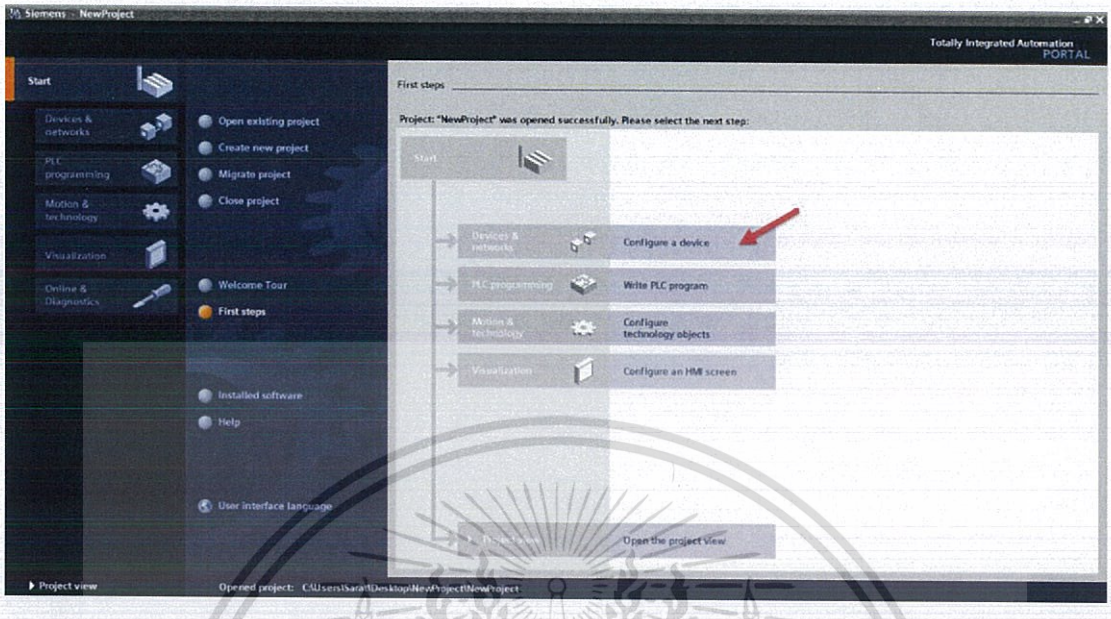
1. กด Create New Project > ตั้งชื่อโปรเจค > กด Create



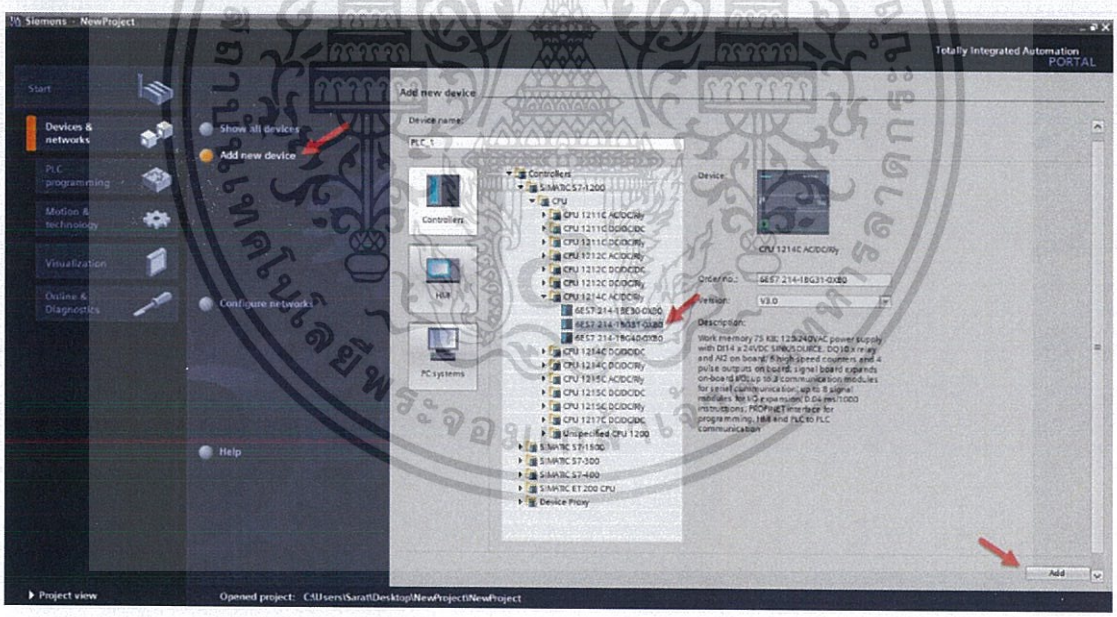
รูปที่ 3.2 หน้าแสดงผล Portal view ในส่วนของการสร้างโปรเจค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. กด Configure a device > Add new device > Controller > รุ่นของพีแอลซีที่ใช้ทำในโปรเจค ชื่อรุ่น 6ES7 214-1BG31-0XB0 > กด Add



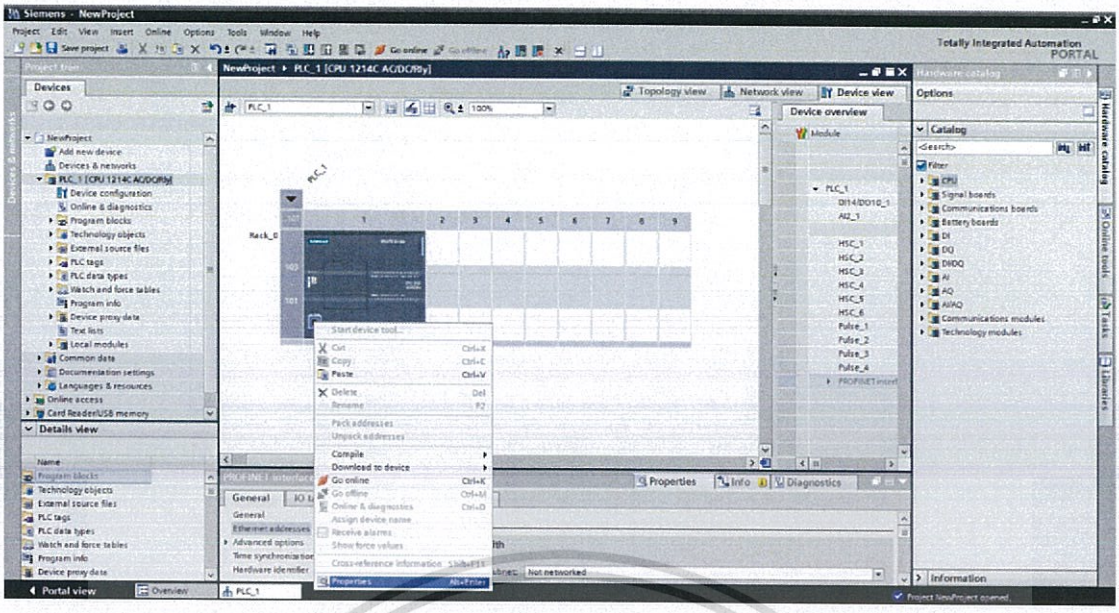
รูปที่ 3.3 หน้าแสดงผล Portal view ในส่วนของการเพิ่มอุปกรณ์



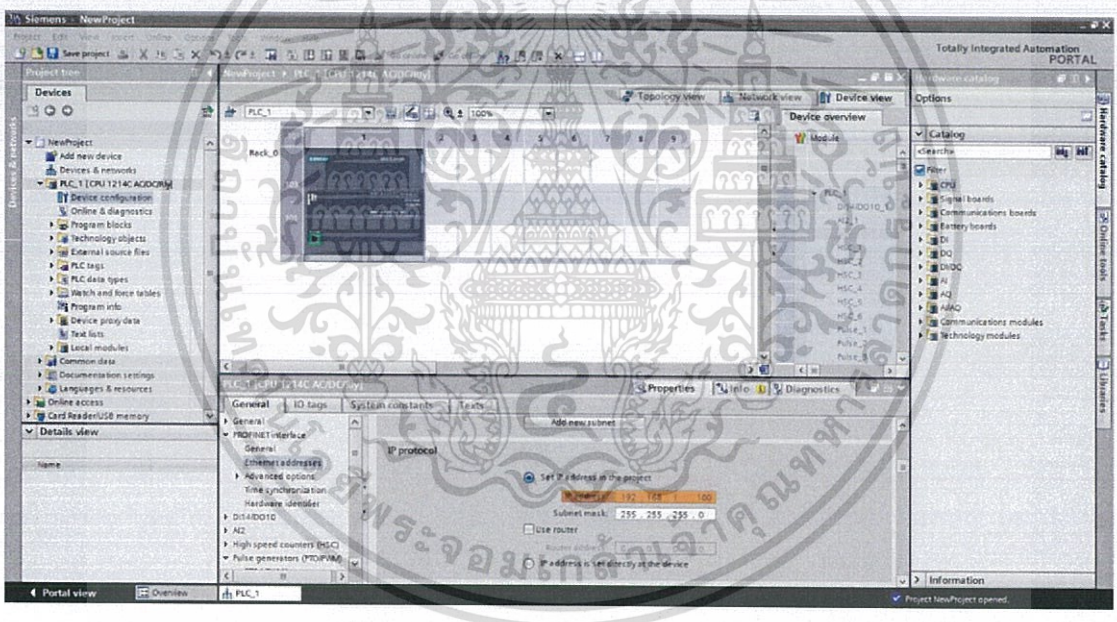
รูปที่ 3.4 หน้าแสดงผล Portal view ในส่วนของการเลือกอุปกรณ์พีแอลซีคอนโทรลเลอร์

3. เมื่อสร้างโปรเจคแล้วจะต้องตั้ง IP Address ของพีแอลซีในโปรแกรมให้ตรงกับเครื่องจริง โดย คลิกขวาที่รูปพีแอลซี > Properties > เลือก PROFINET Interface > Ethernet addresses > IP Protocol > ปรับ IP Address เป็น 192.168.1.100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 การตั้งค่า IP Address ของหน่วยประมวลผลกลางของพีแอลซี (1)

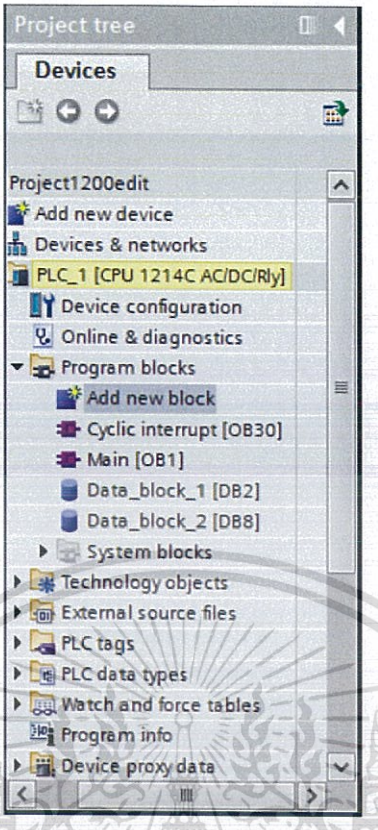


รูปที่ 3.6 การตั้งค่า IP Address ของหน่วยประมวลผลกลางของพีแอลซี (2)

3.2.2 เขียนโปรแกรมพีแอลซีเพื่อควบคุมระบบโดยใช้ตัวควบคุมแบบพีไอดี

1. ที่ Project Tree เลือก PLC_1 > Program Blocks > Add new block

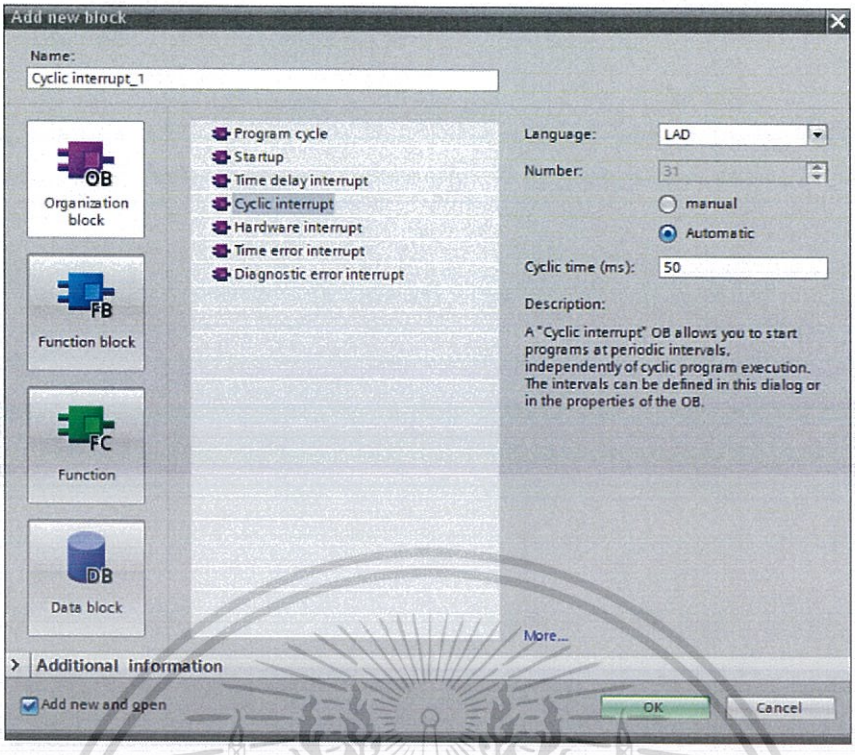
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 การสร้างพื้นที่สำหรับเขียนโปรแกรมควบคุม Main [OB1]

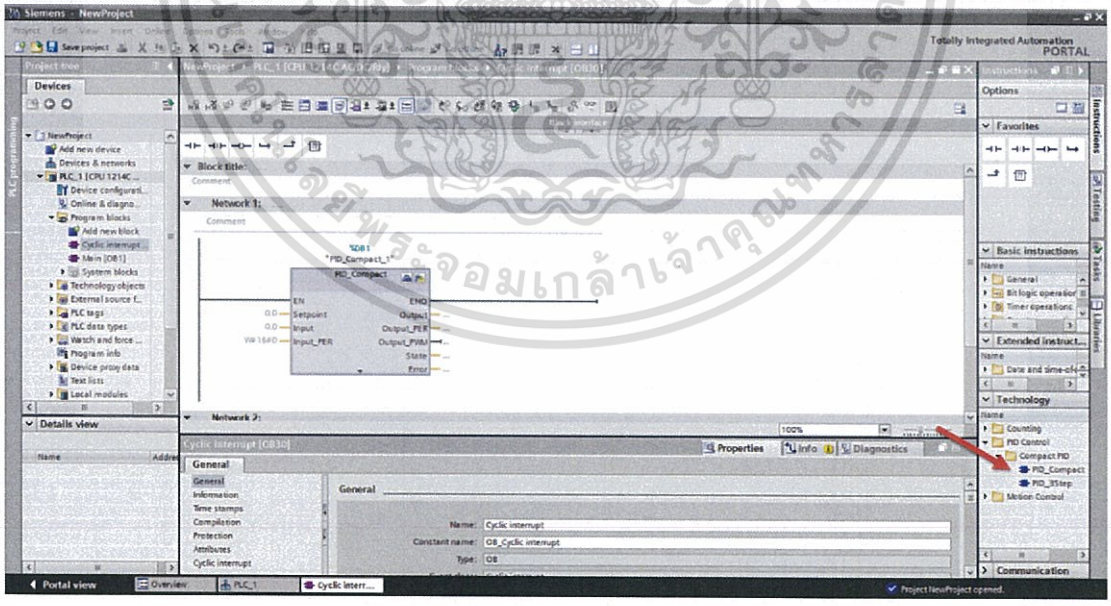
2. เลือก Organization Block > Cyclic interrupt โดย OB แบบ Cyclic interrupt จะสั่งให้โปรแกรมที่อยู่ภายใน OB ทำงานในทุกๆ Cyclic time ที่ตั้งไว้ ดังนั้นในการสร้างคอนโทรลเลอร์แบบพีไอดี จึงจำเป็นต้องสร้างใน OB นี้ เพื่อให้วัดค่าใหม่ตลอดเวลา ซึ่งในโปรเจคนี้จะตั้ง Cyclic time ไว้ที่ 50 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 การสร้าง Cyclic interrupt block

3. เลือก Technology > PID Control > Compact PID > PID_Compact > PID_Compact ไปวางบน Racks



รูปที่ 3.9 การเพิ่มบล็อก PID_Compact

โดย Setpoint – ค่าที่ต้องการให้ระบบลู่เข้าหา

Input – ค่าปัจจุบันของระบบที่ระต้องการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Input_PER – ค่าปัจจุบันของระบบที่ระต้องการควบคุมในหน่วย Analog

Output – ค่าที่ต้องสั่งเพื่อควบคุมระบบ

Output_PER – ค่าที่ต้องสั่งเพื่อควบคุมระบบในหน่วย Analog

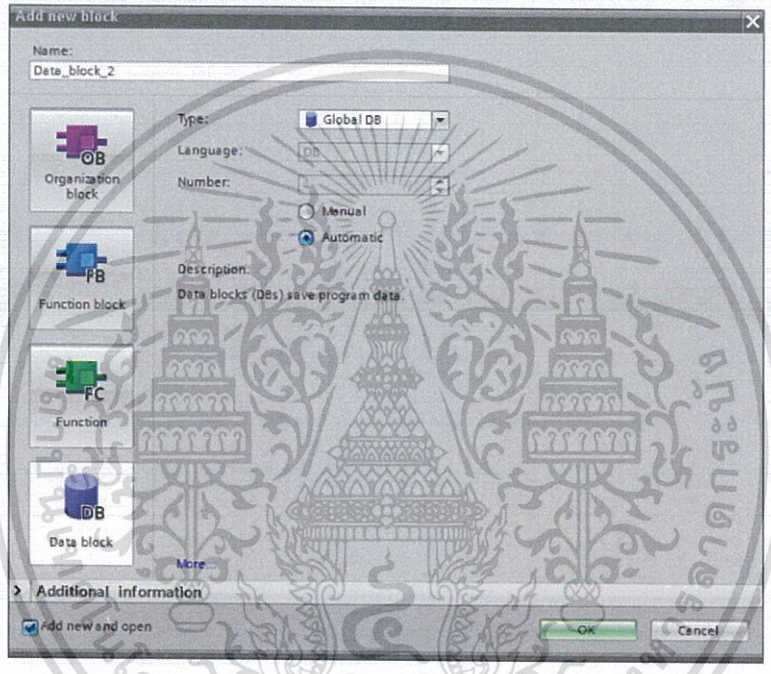
Output_PWM – ค่าที่ต้องสั่งเพื่อควบคุมระบบโดยควบคุมแบบ Boolean

State – สถานะของบล็อก PID_Compact

Error – สถานะ Error ของบล็อก PID_Compact

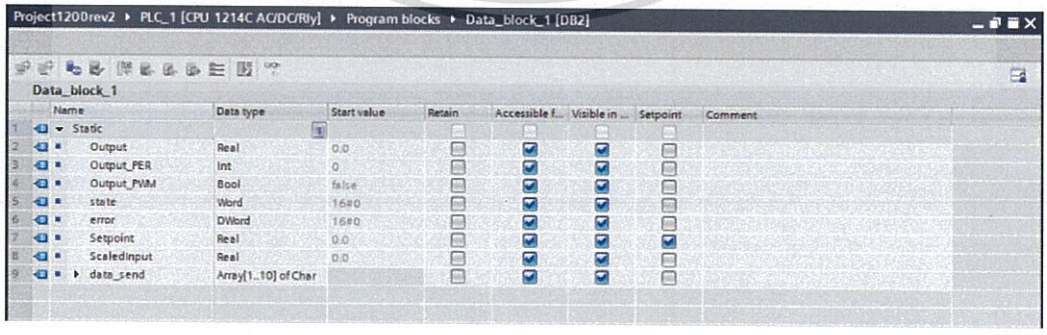
(โดยค่าของ State และ Error สามารถดูได้จาก Manual)

4. กด Add new Block > Data Block > OK



รูปที่ 3.10 การสร้างหน้า Data block สำหรับตั้งค่าพารามิเตอร์พีไอดี

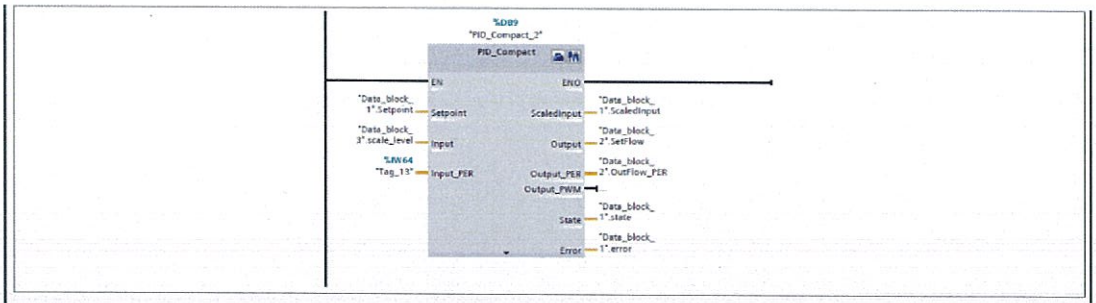
ตั้งพารามิเตอร์ต่างๆ สำหรับ Primary PLC Controller ดังนี้



รูปที่ 3.11 ตัวแปรสำหรับบล็อกพีไอดีในหน้า Data block

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. นำพารามิเตอร์มาใส่ในบล็อก PID_Compact ด้วยวิธี Drag and Drop ดังนี้

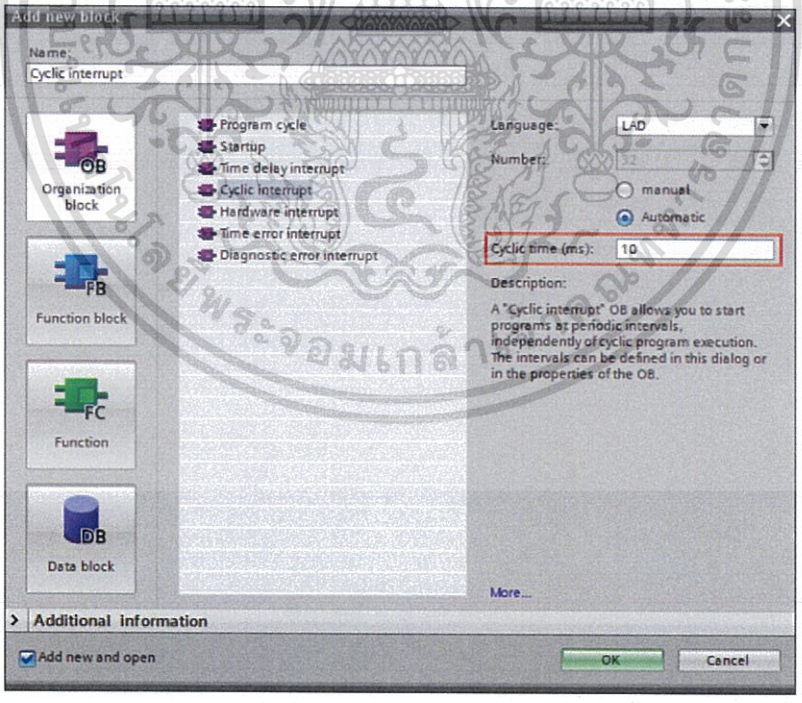


Symbol	Address	Type	Comment
"Data_block_1".error		DWord	
"Data_block_1".ScaledInput		Real	
"Data_block_1".Setpoint		Real	
"Data_block_1".state		Word	
"Data_block_2".OutFlow_PER		Int	
"Data_block_2".SetFlow		Real	
"Data_block_3".scale_level		Real	
"Tag_13"	%IW64	Int	

รูปที่ 3.12 การใส่พารามิเตอร์ลงบล็อก PID_Compact

โดย IW64 คือ Address ขา Analog In ของพีแอลซีที่รับค่าจาก Level Sensor

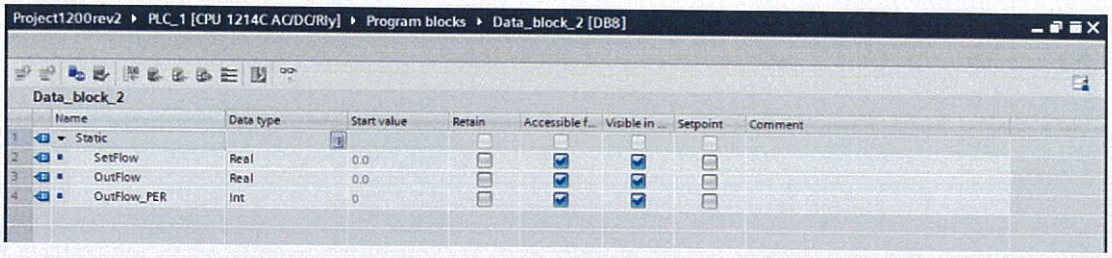
6. สร้าง Cyclic Interrupt Block สำหรับตัวควบคุมแบบพีไอดีอีกตัว เพื่อทำการควบคุมแบบแคสเคด โดยให้ Cyclic time มีค่า 10 ms เนื่องจากเวลาในการตอบสนองของ Secondary Controller จะต้องมีความไวกว่าของ Primary Controller เพื่อให้ได้ระบบที่มีความแม่นยำ



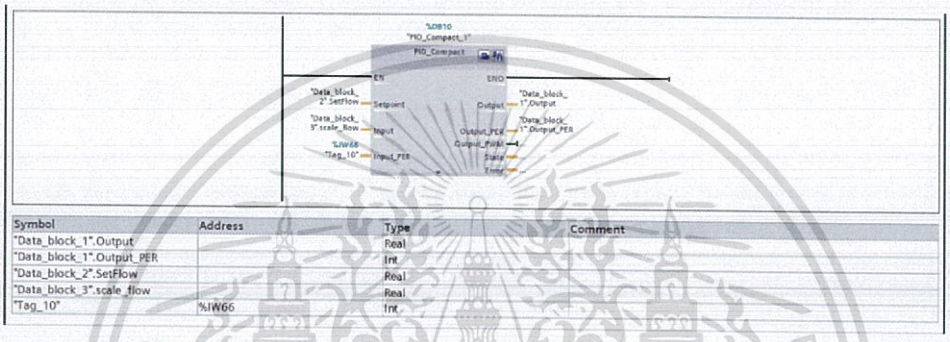
รูปที่ 3.13 การตั้งค่า Cyclic time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. สร้าง Data block เพื่อตั้งพารามิเตอร์สำหรับ Secondary PLC Controller และนำพารามิเตอร์ไปใส่ในบล็อก ดังนี้



รูปที่ 3.14 สร้างพารามิเตอร์สำหรับบล็อกพีไอดีสำหรับการควบคุมแบบแคสเคด

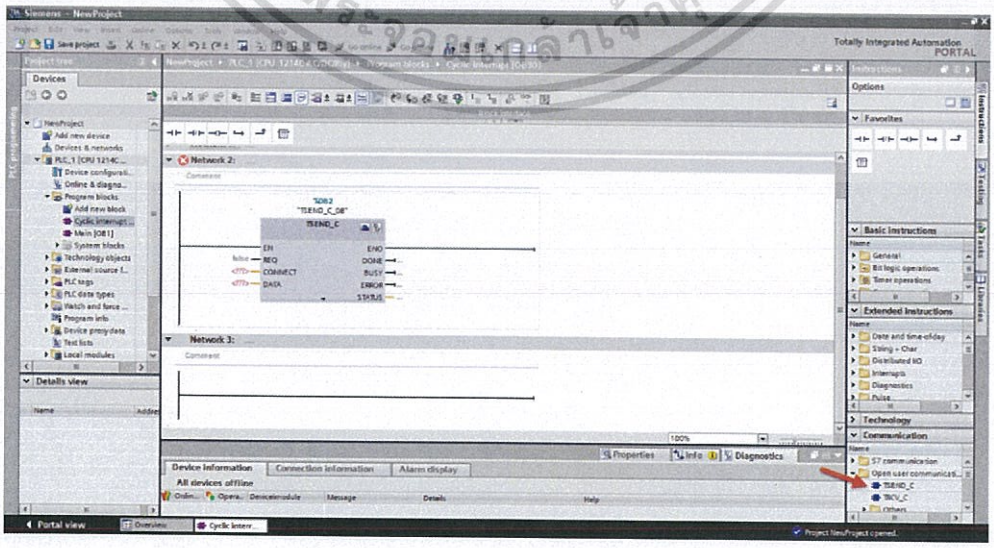


รูปที่ 3.15 การใส่พารามิเตอร์ลงบล็อก PID_Compact สำหรับการควบคุมแบบแคสเคด

โดย IW66 คือ Address ขา Analog In ของพีแอลซีที่รับค่าจาก Flow Sensor

8. ที่ Project Tree เลือก PLC_1 > Program Blocks > Main [OB1]

9. เลือก Communication > Open user communication > TSEND_C ไปวางบน Racks



รูปที่ 3.16 การเพิ่มบล็อก TSEND_C สำหรับการเชื่อมต่ออุปกรณ์ไร้สายผ่านโปรโตคอล TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย REQ - ส่งคำขอในการส่งข้อมูล ซึ่งสถานะของขานี้ต้องเป็น TRUE จึงจะทำการส่งข้อมูล

CONT – เป็นตัวอนุญาตการเชื่อมต่อระหว่างพีแอลซีกับพีแอลซี หรือพีซีเครื่องอื่น ซึ่งสถานะของขานี้ต้องเป็น TRUE จึงจะสามารถทำการเชื่อมต่อได้

LEN - ขนาดของข้อมูลที่จะส่งมีหน่วยเป็นบิต

CONNECT - เป็นขาที่เชื่อมกับบล็อกสำเร็จรูปที่จะสร้างขึ้นหลังทำการตั้งค่าบล็อก TSEND_C

DATA - ข้อมูลที่ต้องการส่ง

DONE - สถานะของบล็อก จะแสดงเป็น TRUE เมื่อบล็อกส่งข้อมูลเสร็จสิ้น

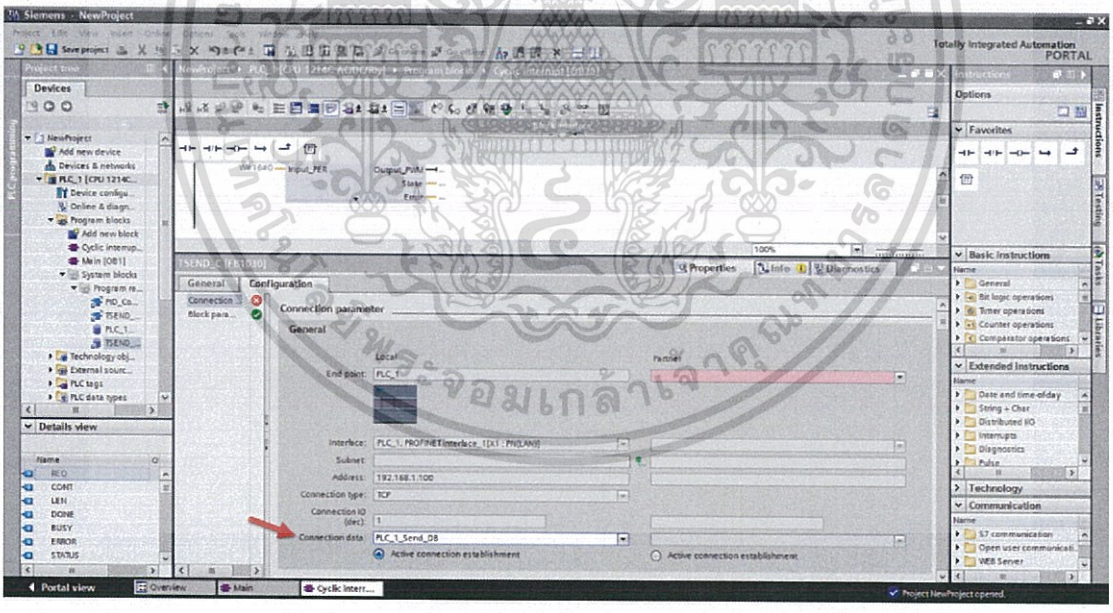
BUSY - สถานะของบล็อก จะแสดงเป็น TRUE เมื่อบล็อกกำลังทำงาน

ERROR - สถานะของบล็อก จะแสดงเป็น TRUE เมื่อบล็อกเกิดข้อผิดพลาด

STATUS - สถานะของบล็อก TSEND_C

(สามารถดูว่าค่าที่แสดงมีความหมายว่าอย่างไรจากคู่มือการใช้)

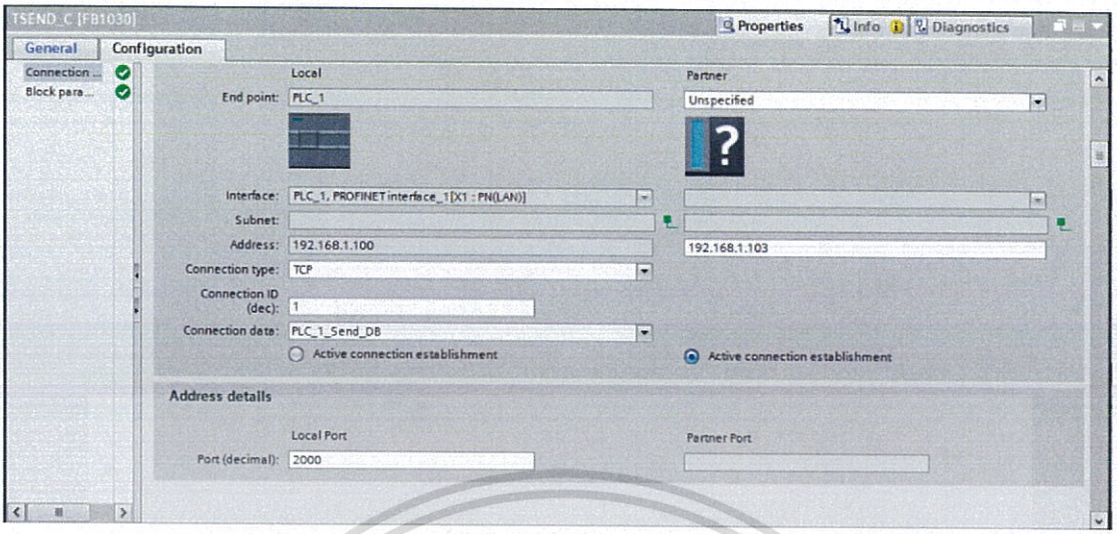
10. เลือกบล็อก TSEND_C > Properties > Configuration > Connection Parameter > Connection Data > new โปรแกรมจะสร้างบล็อก PLC1_Send_DB เพื่อเก็บพารามิเตอร์ต่างๆ และเชื่อมกับขา Connect ของบล็อกโดยอัตโนมัติ



รูปที่ 3.17 การตั้งค่าการเชื่อมต่อของบล็อก TSEND_C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

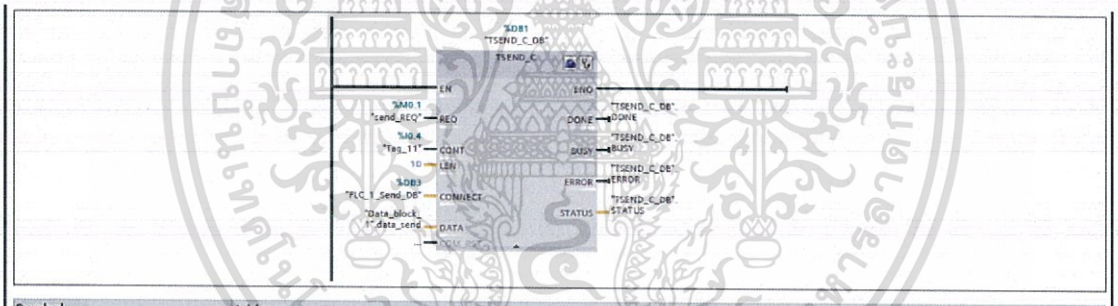
11. ใส่ค่าต่างๆ ดังนี้



รูปที่ 3.18 ค่าการตั้งค่าการเชื่อมต่อบล็อก TSEND_C กับอุปกรณ์

โดย 192.168.1.103 คือ IP ของ WiFi Module ที่ต่อกับมอเตอร์

12. นำพารามิเตอร์มาใส่ในบล็อก TSEND_C ด้วยวิธี Drag and Drop ดังนี้

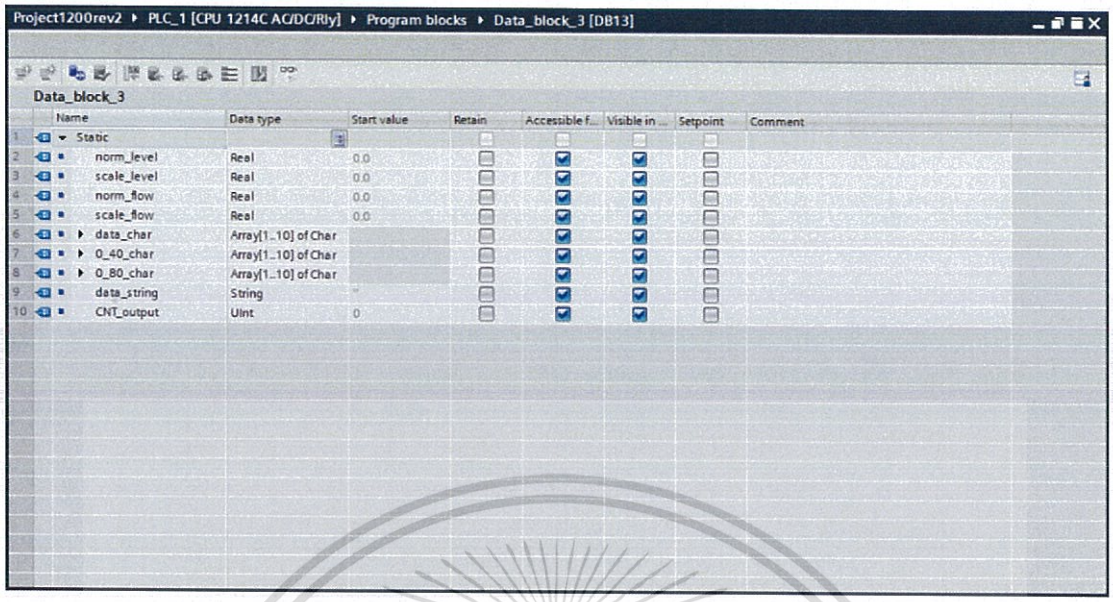


Symbol	Address	Type	Comment
"Data_block_1".data_send		Array	
"send_REQ"	%M0.1	Bool	
"Tag_11"	%I0.4	Bool	
"TSEND_C_DB".BUSY		Bool	Status parameter: BUSY = 1: Job is not yet completed. BUSY = 0: Job is complete
"TSEND_C_DB".DONE		Bool	DONE status parameter: 0: Job not yet started or still running. 1: Job is succ
"TSEND_C_DB".ERROR		Bool	Status parameter: ERROR=1: An error occurred in job processing. STATUS returns
"TSEND_C_DB".STATUS		Word	Status parameter: Error information

รูปที่ 3.19 การใส่พารามิเตอร์ลงบล็อก TSEND_C

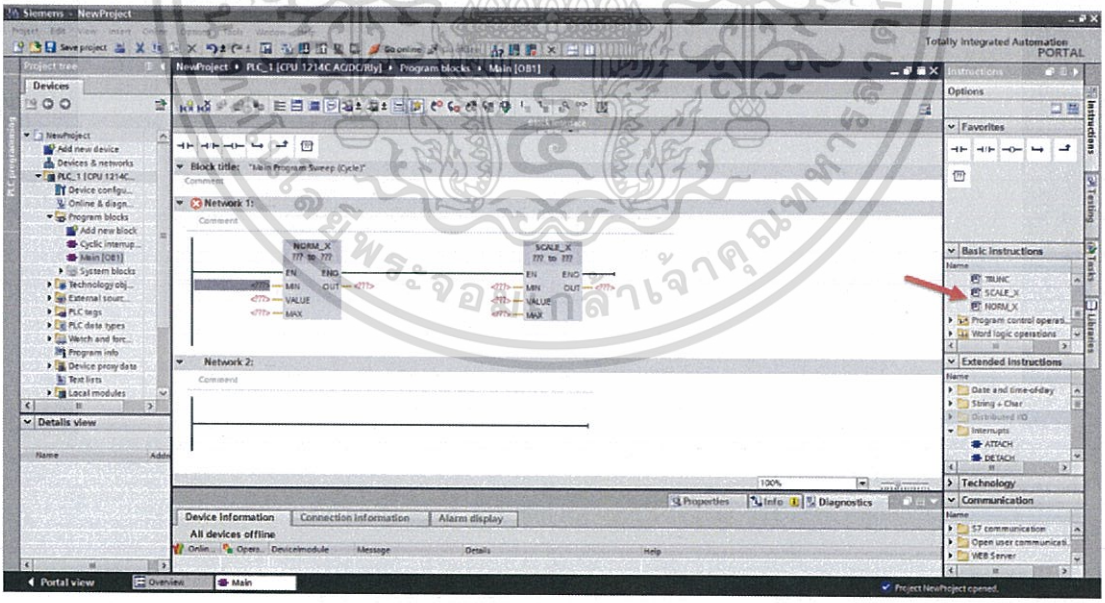
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. สร้าง Data Block เพื่อเก็บตัวแปรของแลดเดอร์สำหรับการแสดงผล และลูปต่างๆ



รูปที่ 3.20 สร้างพารามิเตอร์สำหรับบล็อก NORM_X และ SCALE_X

14. เขียนโปรแกรมเพื่อรับค่าแรงดันไฟฟ้าที่ได้จากขา Analog In ของพีแอลซีโดยเลือก Basic Instruction > Conversion Operations > NORM_X และ SCALE_X นำไปวางไว้ดังรูปที่ 3.21

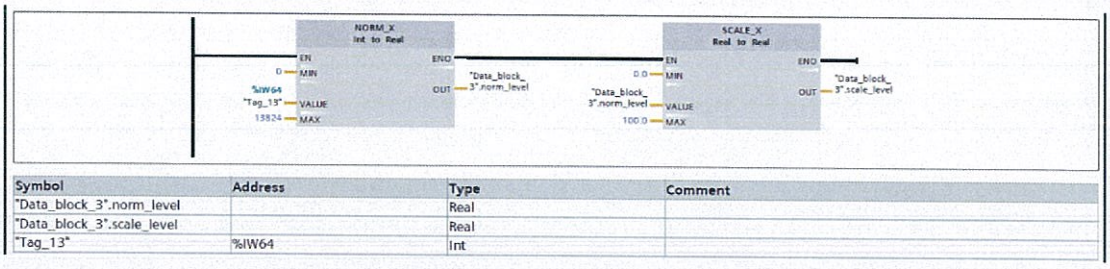


รูปที่ 3.21 การเพิ่มบล็อก NORM_X และ SCALE_X

บล็อก NORM_X จะแปลงค่า VALUE เป็นเปอร์เซ็นต์โดยที่ 100% มีค่าเท่ากับ 1 เมื่อเทียบด้วยค่า MAX และ MIN ส่วนบล็อก SCALE_X จะแปลงค่าเปอร์เซ็นต์ที่ขา VALUE เป็นค่าจริงเมื่อเทียบด้วยค่า MAX และ MIN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

15. ใส่ค่าพารามิเตอร์ดังนี้

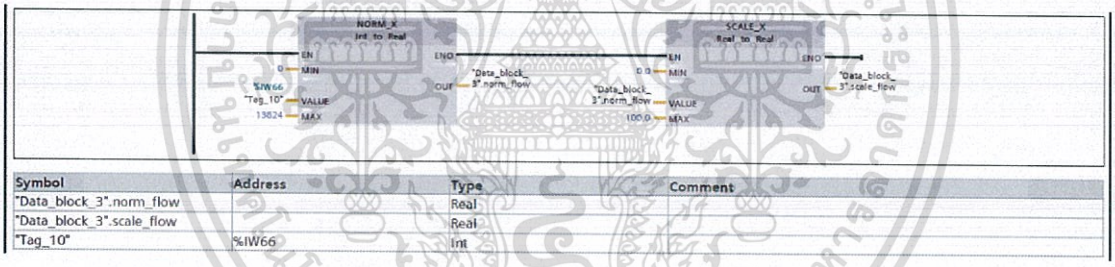


Symbol	Address	Type	Comment
"Data_block_3".norm_level		Real	
"Data_block_3".scale_level		Real	
"Tag_13"	%IW64	Int	

รูปที่ 3.22 การใส่พารามิเตอร์ลงบล็อก NORM_X และ SCALE_X สำหรับ Level

โดย IW64 คือค่าแรงดันไฟฟ้าที่ PLC รับได้จากขา Analog In ซึ่งต่ออยู่กับ Pressure Sensor แบบไร้สาย และมีค่าสูงสุดที่ 27648 จากนั้นจึงทำการแปลงค่าเปอร์เซ็นต์ที่ได้จาก NORM_X เป็นค่าแรงดันไฟฟ้าจริงที่วัดได้จากเครื่องพีแอลซีคือ 0 ถึง 10 โวลต์ แต่ค่าแรงดันไฟฟ้าสูงสุดที่ส่งมาจาก Arduino มีแค่ 5 โวลต์ จึงต้องใส่ค่า MAX เป็น 13824

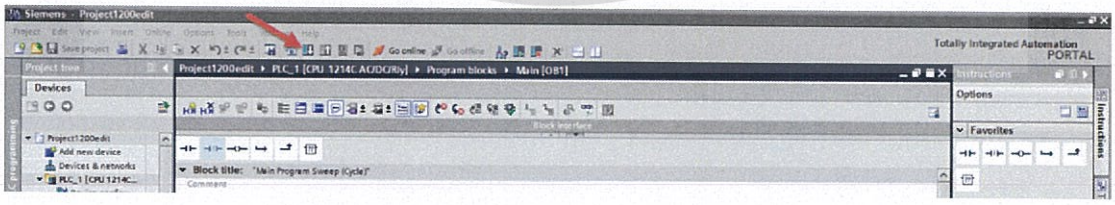
16. ทำซ้ำอีกครั้งโดยใช้ Address IW66 แทน ซึ่ง IW66 คือค่าแรงดันไฟฟ้าที่พีแอลซีรับได้จากขา Analog In ซึ่งต่ออยู่กับ Flow Sensor แบบไร้สาย และใช้บล็อก SCALE_X แปลงค่าที่อ่านได้เป็น 0 ถึง 100%



Symbol	Address	Type	Comment
"Data_block_3".norm_flow		Real	
"Data_block_3".scale_flow		Real	
"Tag_10"	%IW66	Int	

รูปที่ 3.23 การใส่พารามิเตอร์ลงบล็อก NORM_X และ SCALE_X สำหรับ Flow

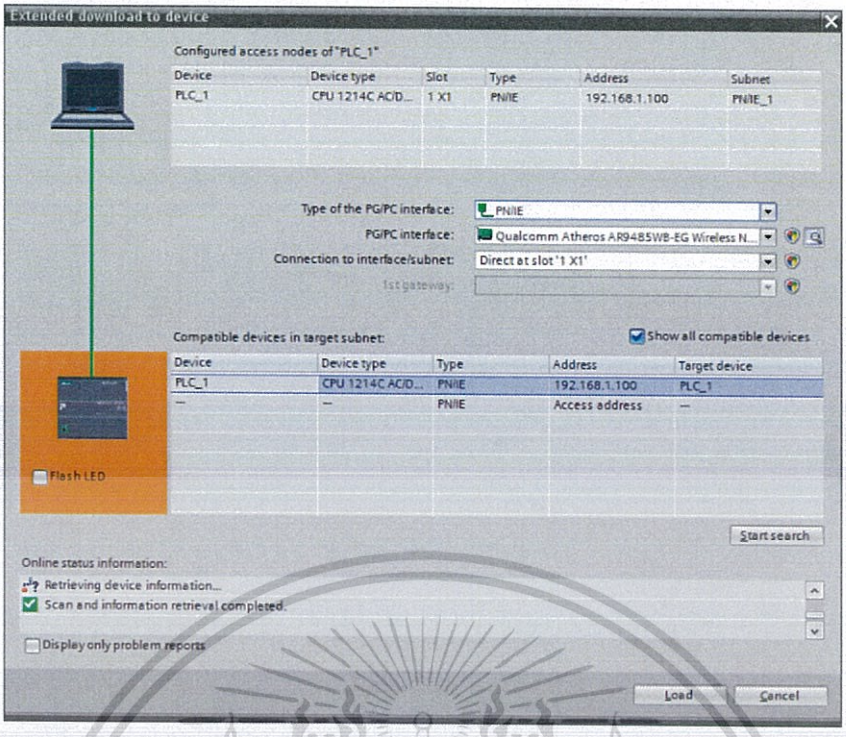
17. ดาวน์โหลดโปรแกรมเข้าเครื่องพีแอลซี



รูปที่ 3.24 การโหลดโปรแกรมลงเครื่องพีแอลซี

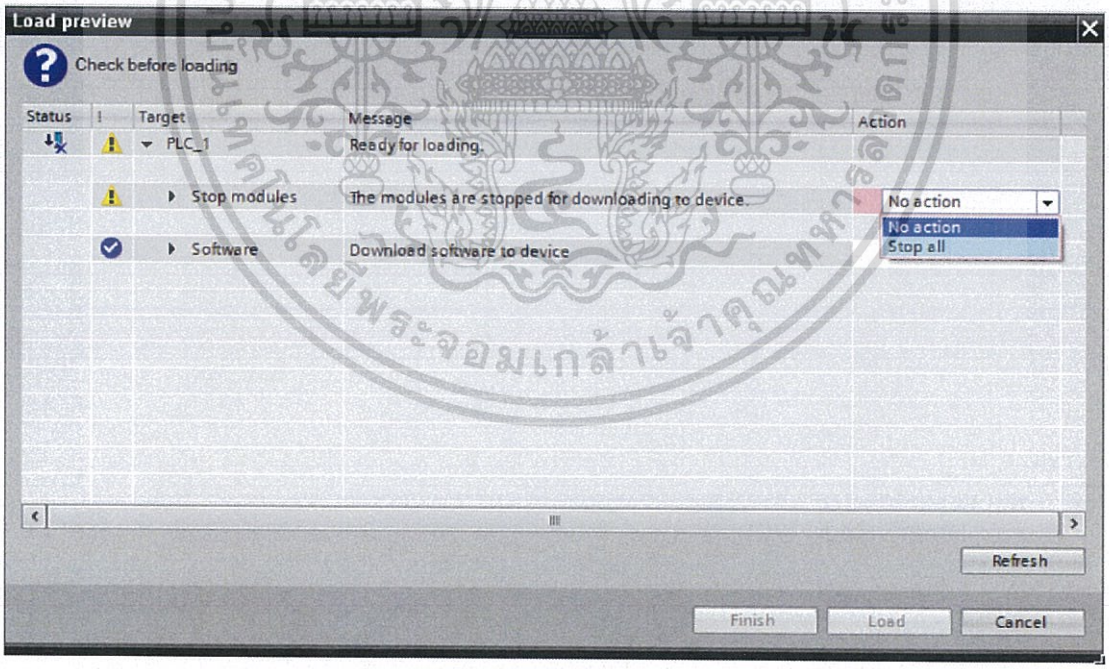
เลือก Wireless Network Module ของคอมพิวเตอร์ ที่ช่อง PG/PC Interfaces กด Start Search แล้วกด Load

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



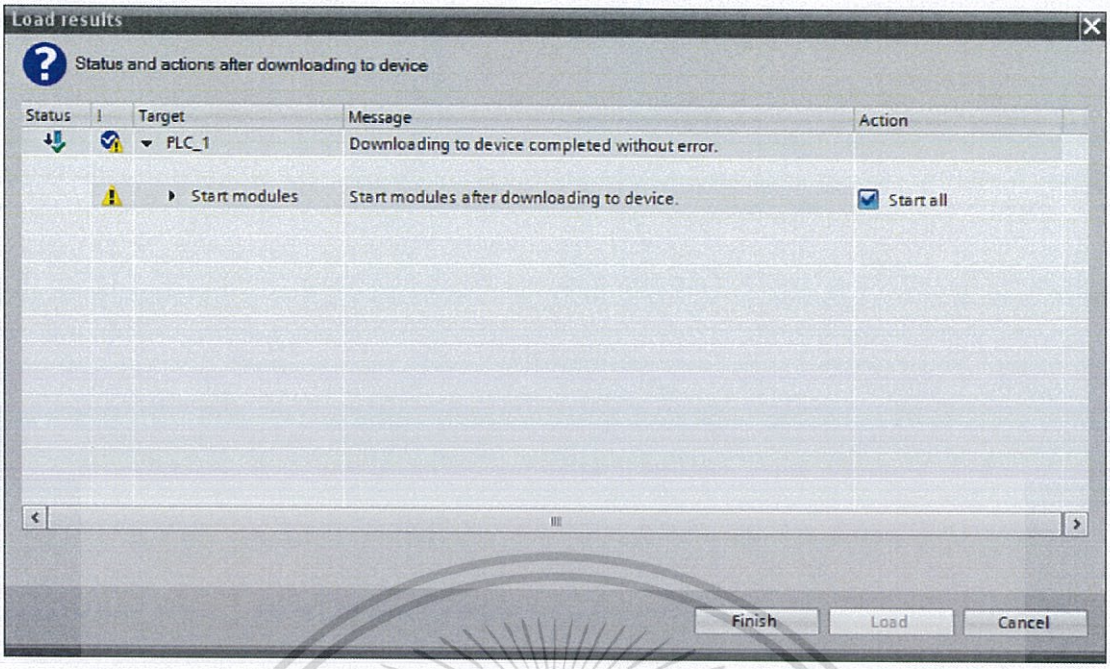
รูปที่ 3.25 หน้าต่าง Download to device

18. กด Stop all > Load > Finish



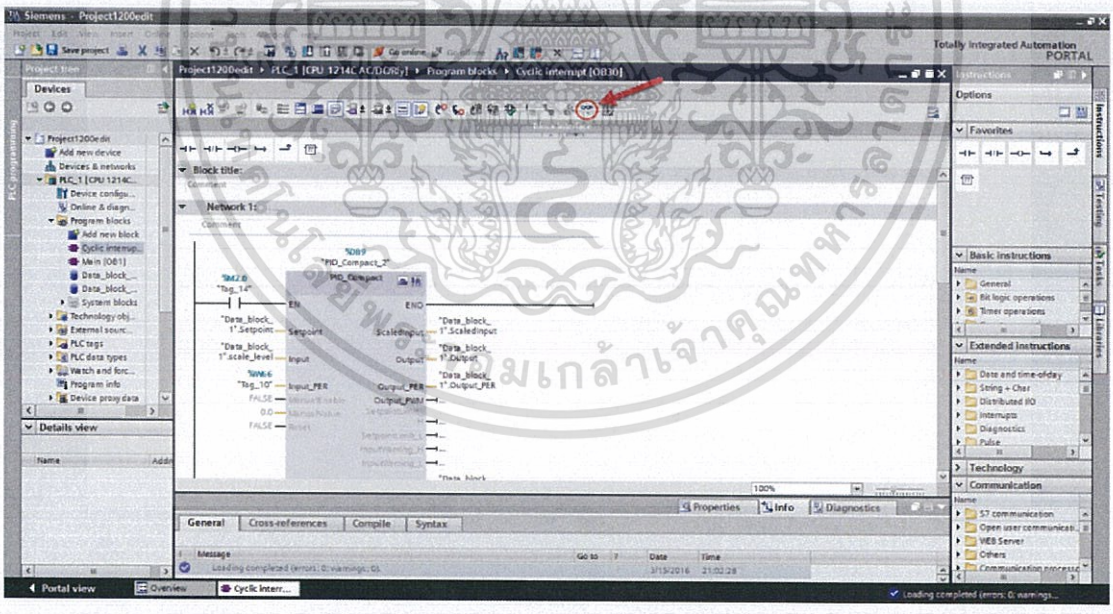
รูปที่ 3.26 หน้าต่าง Load preview

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 หน้าต่าง Load result

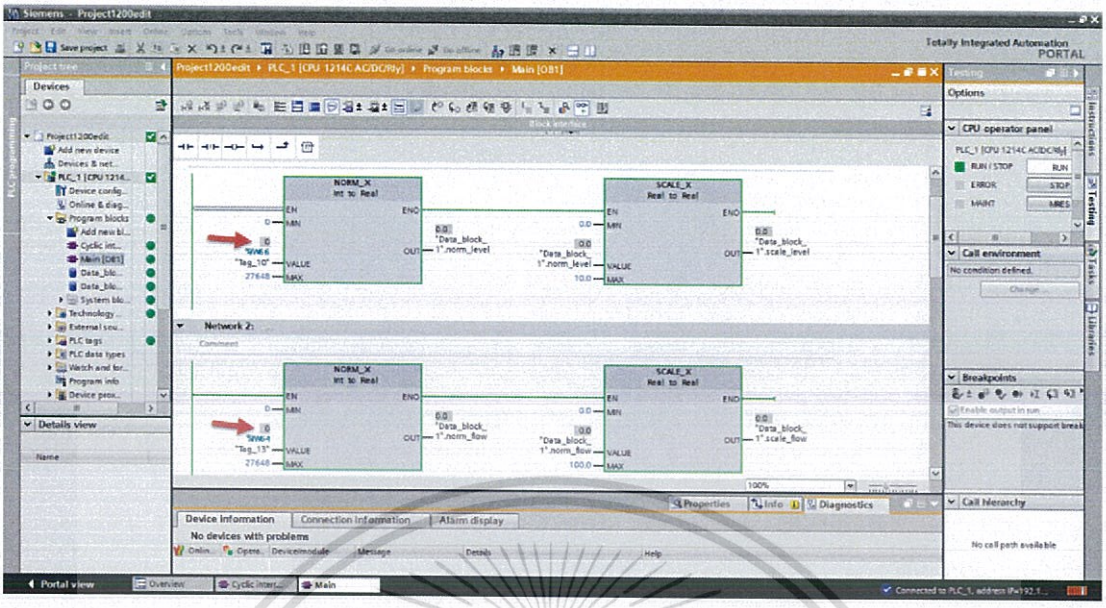
19. กดปุ่ม Monitor all เพื่อตรวจสอบว่าโปรแกรมสามารถรับค่าจากช่อง Analog In ได้หรือไม่ รวมถึงเป็นการตรวจสอบการเชื่อมต่อระหว่าง WiFi Module ด้วย



รูปที่ 3.28 การ Monitor ค่า และการทำงานของโปรแกรมแลดเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

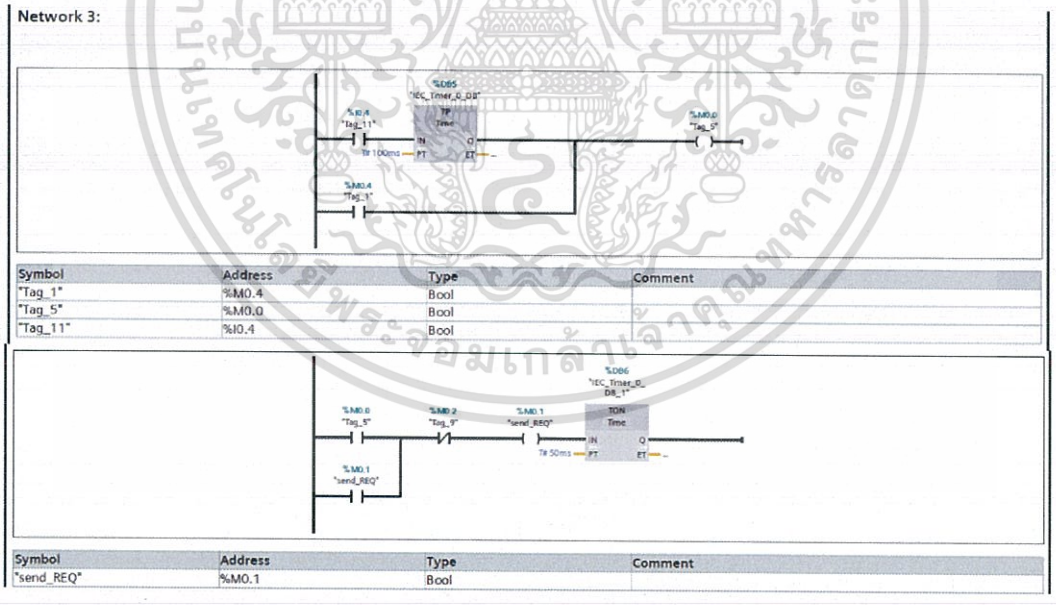
20. ตรวจสอบว่าเลขที่แสดงมีค่าตรงตามที่วัดจากเครื่องพีแอลซีหรือไม่



รูปที่ 3.29 ค่าที่เข้าขาแอนะล็อก IW64 และ IW66

จากนั้นกด Monitor all อีกครั้งเพื่อหยุด

21. เขียนโปรแกรมแลดเดอร์ให้บล็อก TSEND_C ส่งข้อมูลเป็นเวลา 50 ms และส่งทุกๆ 50 ms

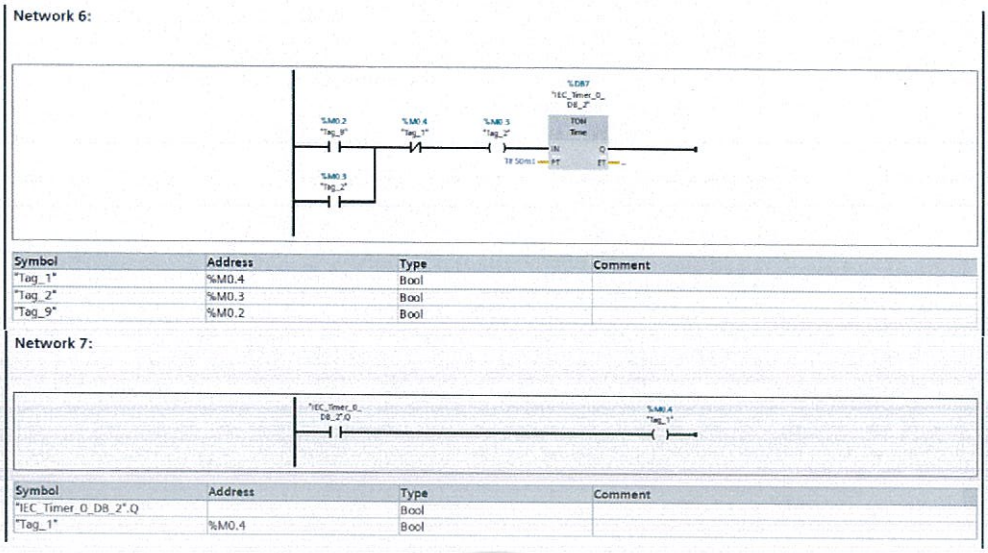


Symbol	Address	Type	Comment
"Tag_1"	%M0.4	Bool	
"Tag_5"	%M0.0	Bool	
"Tag_11"	%I0.4	Bool	

Symbol	Address	Type	Comment
"send_REQ"	%M0.1	Bool	

รูปที่ 3.30 โปรแกรมสำหรับบล็อก TSEND_C ส่งข้อมูลเป็นเวลา 50 ms และส่งทุกๆ 50 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Network 6:

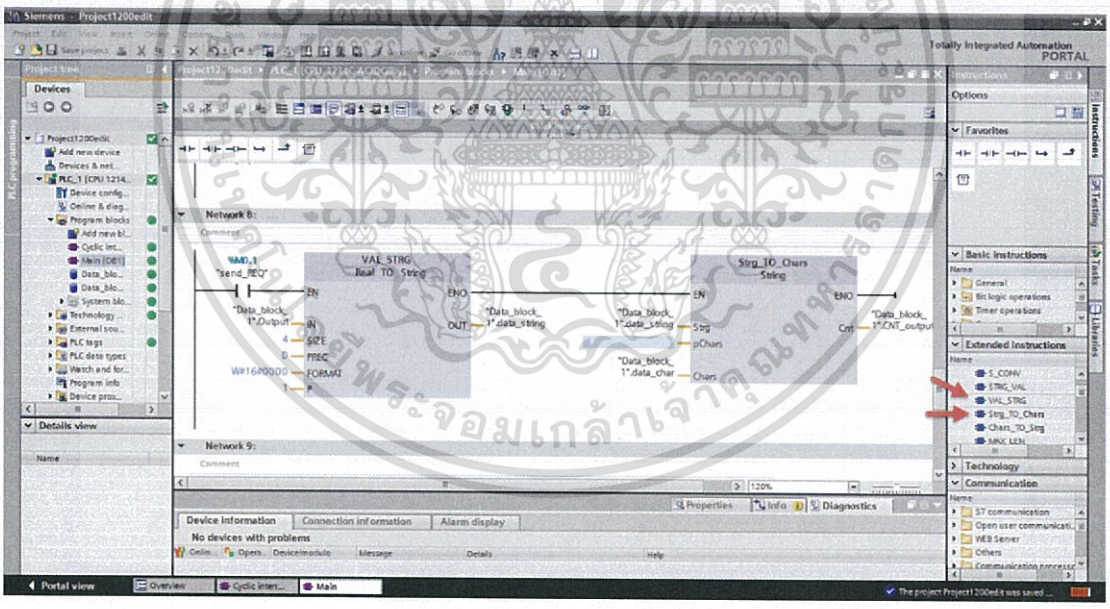
Symbol	Address	Type	Comment
"Tag_1"	%M0.4	Bool	
"Tag_2"	%M0.3	Bool	
"Tag_9"	%M0.2	Bool	

Network 7:

Symbol	Address	Type	Comment
"IEC_Timer_0_DB_2".Q		Bool	
"Tag_1"	%M0.4	Bool	

รูปที่ 3.31 โปรแกรมสำหรับบล็อก TSEND_C ส่งข้อมูลเป็นเวลา 50 ms และส่งทุกๆ 50 ms

22. เลือก Extended instructions > String + Char > VAL_STRG เพื่อแปลงค่าเอาต์พุตของบล็อก PID_Compact จาก Real เป็น String และ Strg_TO_Chars เพื่อแปลง String ให้เป็น Array of Char เพื่อส่งผ่านบล็อก TSEND_C ไปสั่งมอเตอร์



รูปที่ 3.32 การเพิ่มบล็อก VAL_STRG และ Strg_To_Chars

โดย IN – ข้อมูลที่ต้องการแปลงเป็น String

SIZE – ขนาดของ String

PREC – จำนวนเลขหลังจุดทศนิยม

FORMAT – รูปแบบของ String โดยสามารถดูค่าได้จากคู่มือการใช้งาน

P – บิตแรกของ String ที่จะใช้เก็บข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUT – ตำแหน่งที่จะเก็บ String

Strg – String ที่ต้องการแปลงเป็น Char

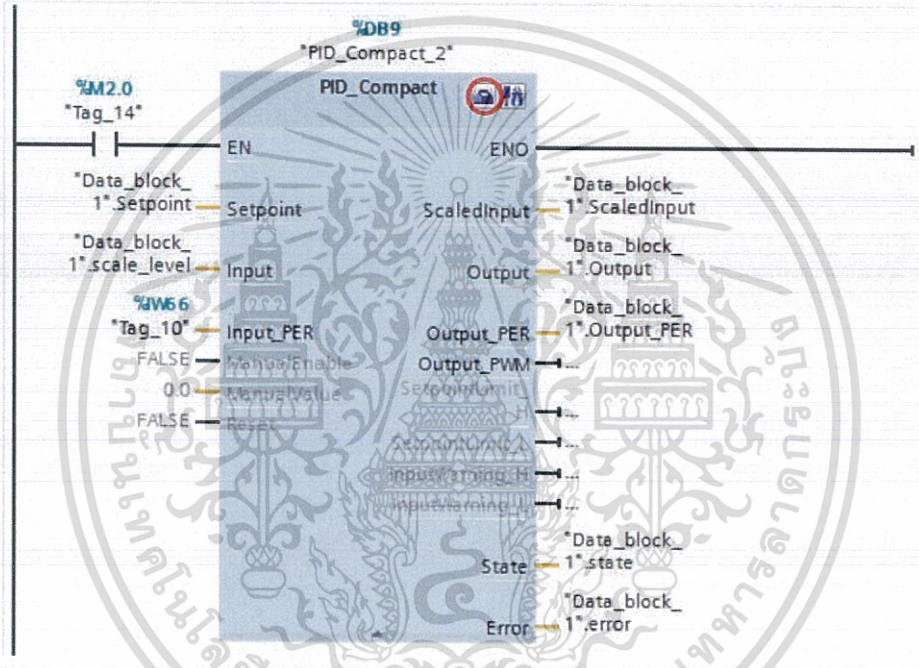
pChars – ตำแหน่งเริ่มต้นใน Array ของ Char ตัวแรก

Chars – ตำแหน่งที่จะเก็บ Chars

Cnt – จำนวนของตัวอักษรที่ซ้ำ

23. ปรับค่าพารามิเตอร์ต่างๆ ของบล็อก PID_Compact โดยต้องปรับที่ Secondary Controller ก่อนแล้วจึงค่อยปรับ Primary Controller

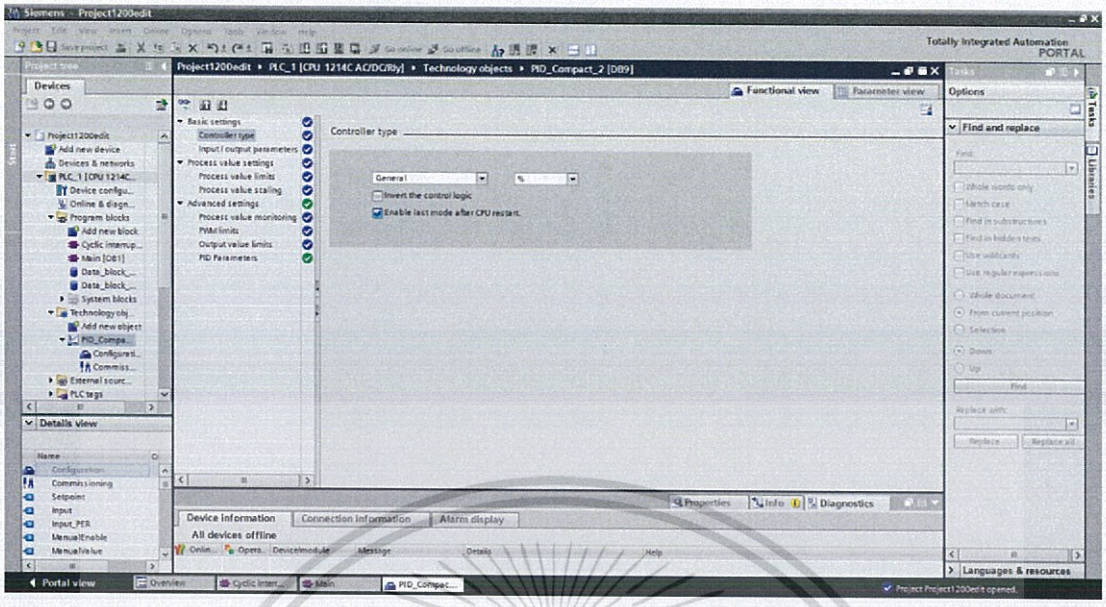
24. เลือกบล็อก PID_Compact > Opens the configuration window



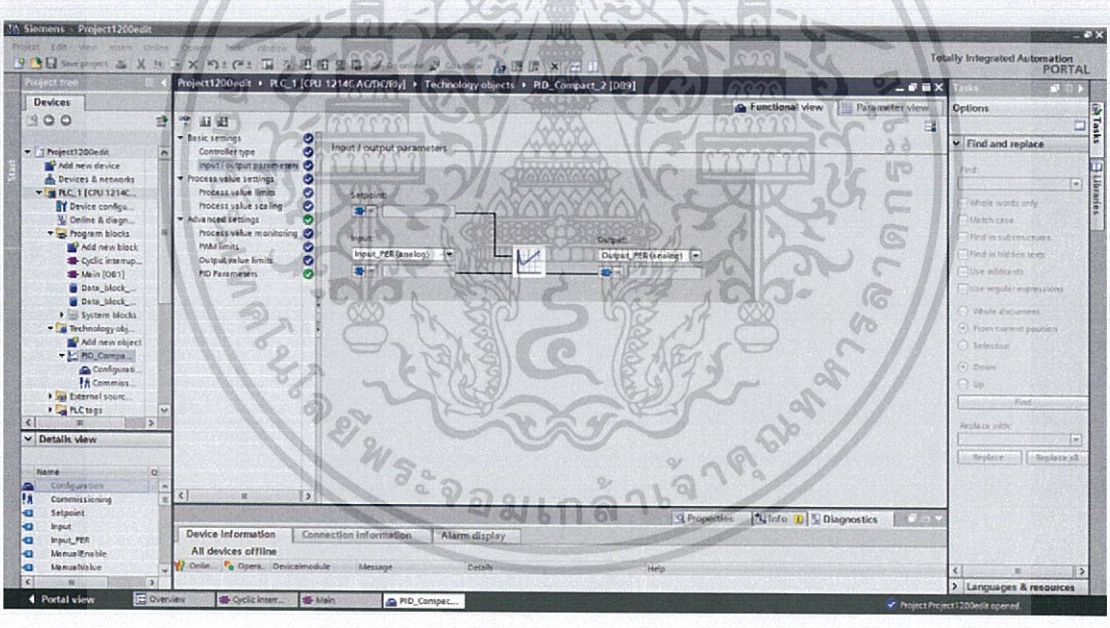
รูปที่ 3.33 การเปิดหน้า configuration window

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

25. ตั้งค่าต่างๆ ดังต่อไปนี้

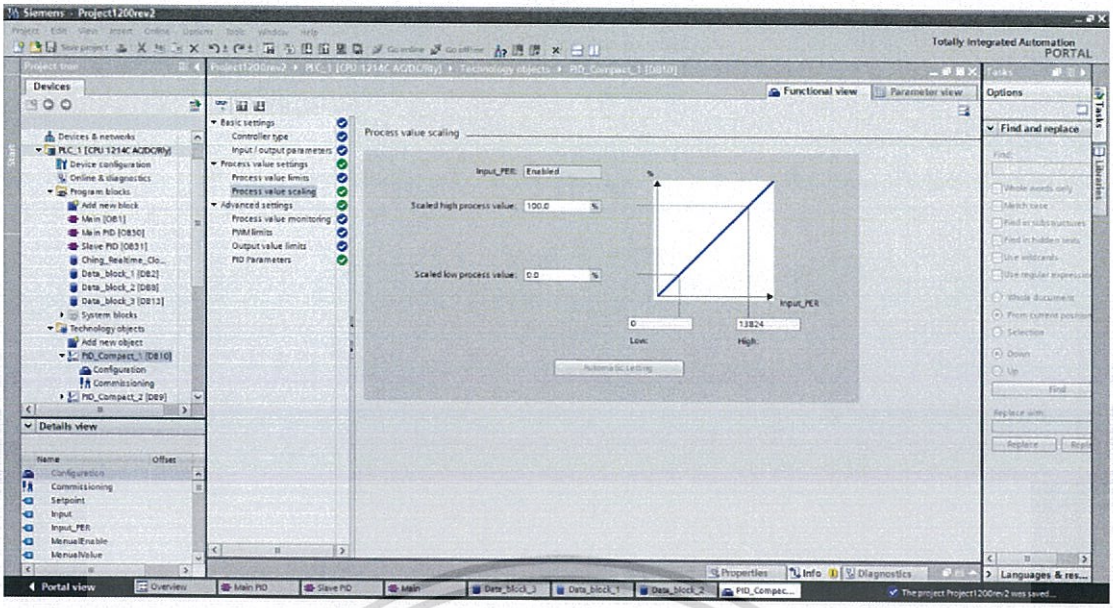


รูปที่ 3.34 หน้าต่าง configuration window

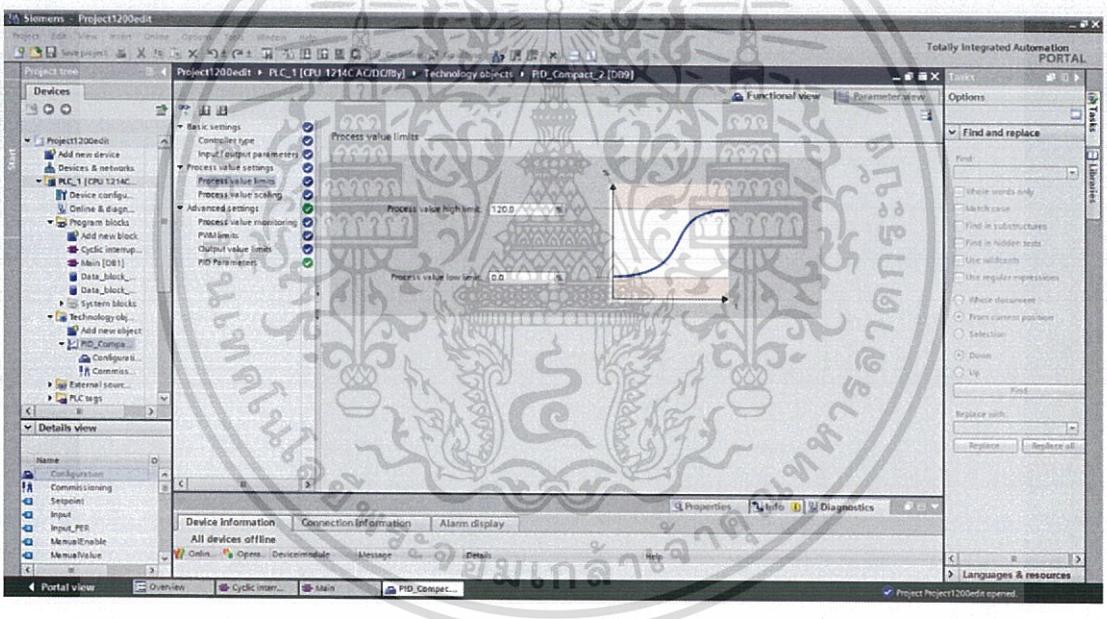


รูปที่ 3.35 การตั้งค่า Input/Output Parameter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



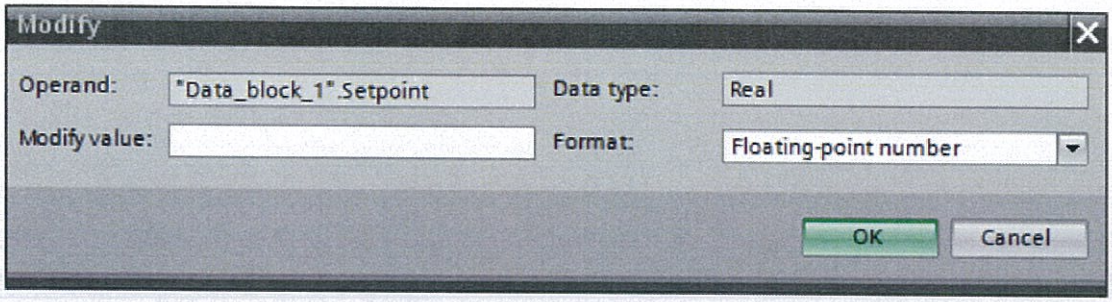
รูปที่ 3.36 การตั้งค่า Process value scaling



รูปที่ 3.37 การตั้งค่า Process value limits

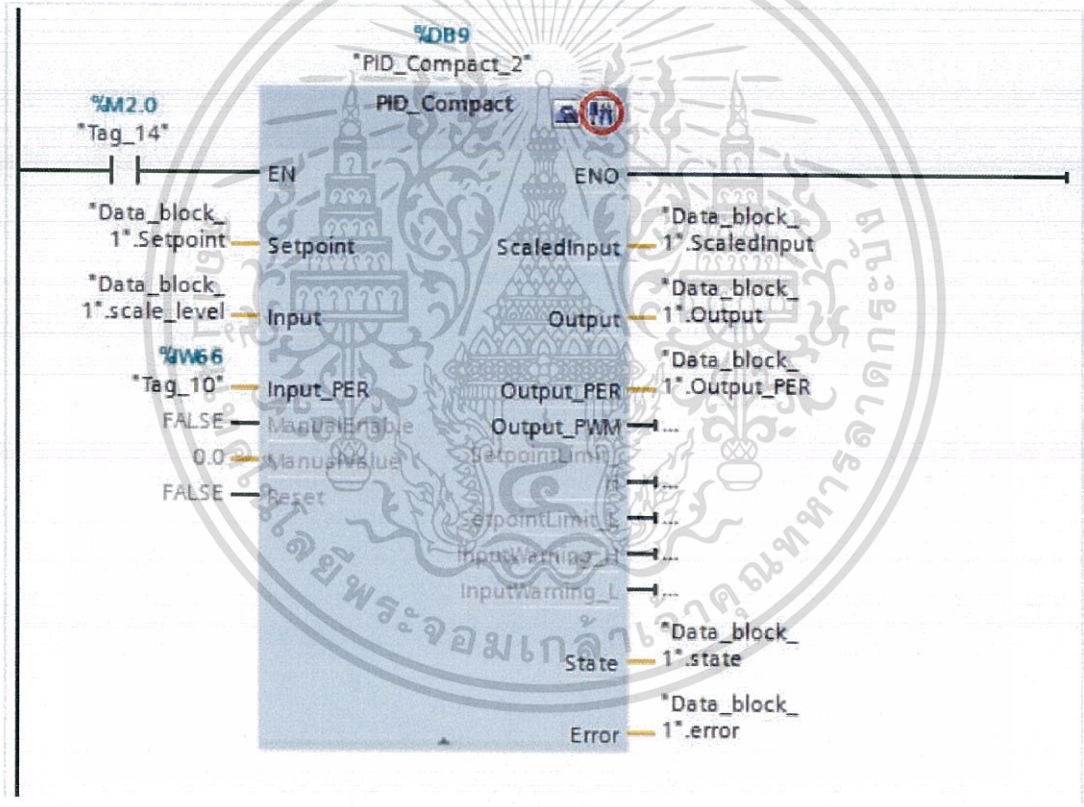
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้วนการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

26. ดาวน์โหลดโปรแกรมเข้าพีแอลซี และ Monitor > คลิกขวาที่ Setpoint ของบล็อก PID_Compact > Operands > Modify Operands > ปรับเป็นค่าเป้าหมายที่ต้องการ



รูปที่ 3.38 การปรับเปลี่ยนค่าเป้าหมาย

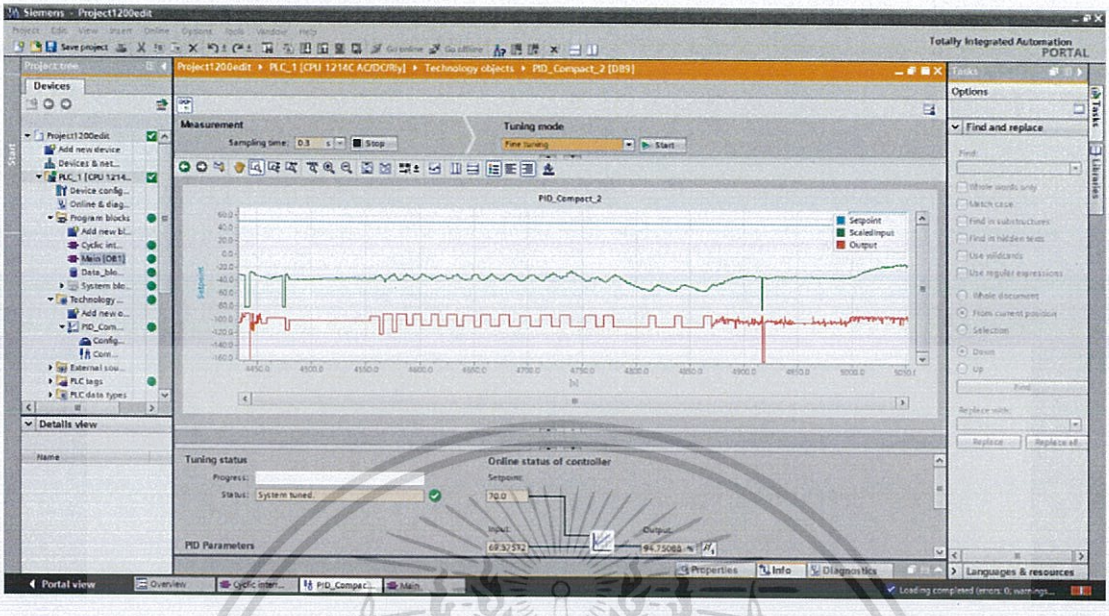
27. เลือกบล็อก PID_Compact > Open the commissioning window



รูปที่ 3.39 การเปิดหน้า commissioning window

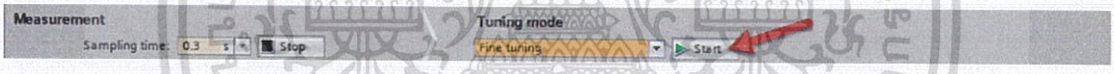
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

28. Measurement > Start เพื่อดูกราฟของระบบ และ Tuning mode > Fine Tuning > Start เพื่อปรับค่าพีไอดีให้เข้ากับระบบ



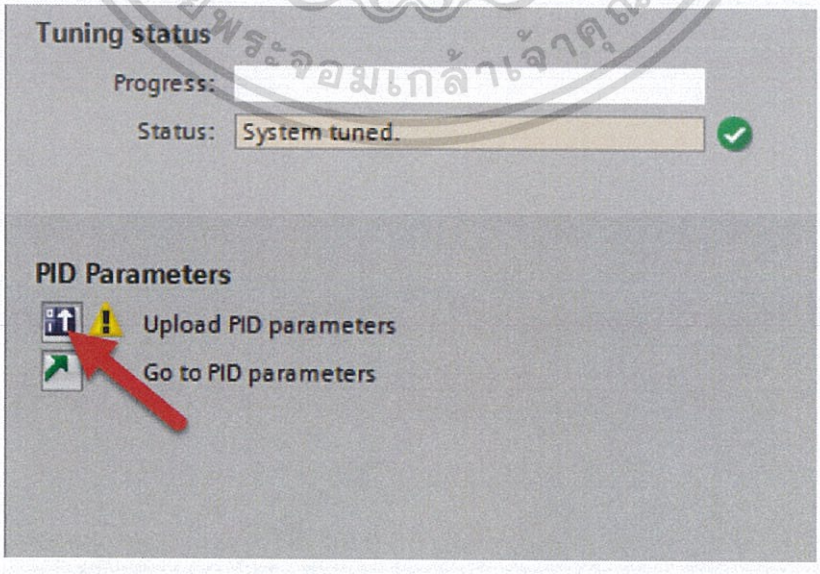
รูปที่ 3.40 การปรับค่าพีไอดี (1)

29. คลิก Start เพื่อเริ่มการปรับค่าพารามิเตอร์อัตโนมัติ



รูปที่ 3.41 การปรับค่าพีไอดี (2)

30. เมื่อปรับค่าเสร็จ คลิก Upload PID Patameters > ดาวน์โหลด > Moniter all

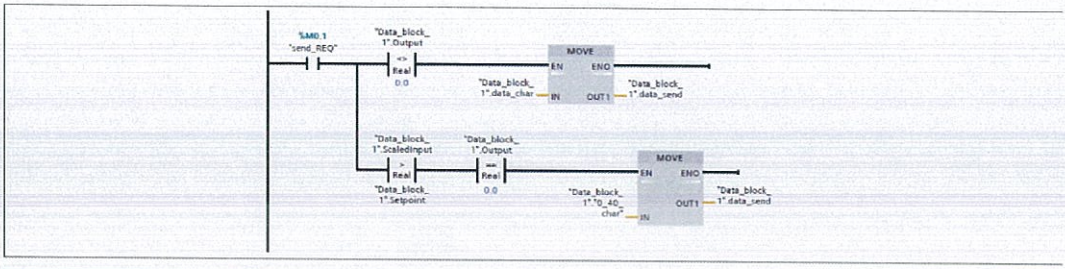


รูปที่ 3.42 การอัพโหลดพารามิเตอร์ของบล็อก PID_Compact

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

31. ทำเช่นเดียวกันกับบล็อก PID_Compact ของ Primary Controller

32. สร้างเงื่อนไขให้ส่งค่า 40 แทน 0 เมื่อเอาต์พุตของ PID_Compact เป็น 0 เนื่องจากมอเตอร์จะสั่งค่าเก่าเมื่อได้รับค่า 0 จึงใช้สั่งค่า 40 ซึ่งทำให้มอเตอร์ทำงาน แต่มีแรงไม่มากพอจะปั๊มน้ำกลับขึ้นไปได้

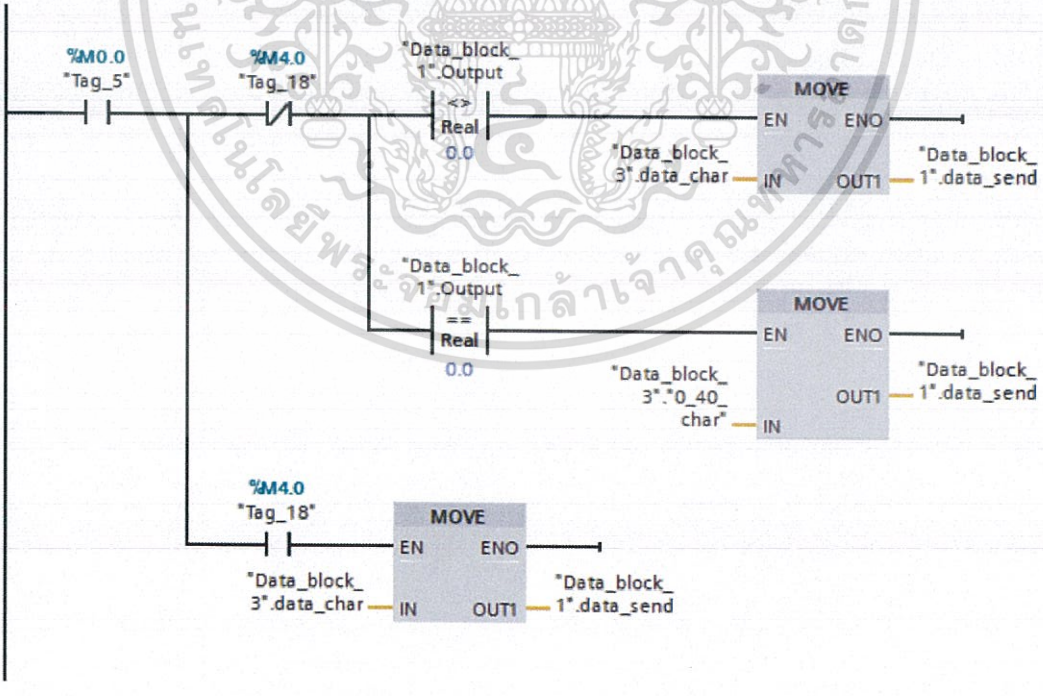


Symbol	Address	Type	Comment
"Data_block_1"."0_40_char"		Array	
"Data_block_1".data_char		Array	
"Data_block_1".data_send		Array	
"Data_block_1".Output		Real	
"Data_block_1".ScaledInput		Real	
"Data_block_1".Setpoint		Real	
"send_REQ"	%M0.1	Bool	

รูปที่ 3.43 เงื่อนไขการสั่งงานมอเตอร์แบบอัตโนมัติ

33. ดาวน์โหลด > Moniter all เพื่อทดสอบ

34. เขียนโปรแกรมเพื่อสร้างเงื่อนไขสำหรับการสั่งมอเตอร์ด้วยตนเอง



รูปที่ 3.44 แลดเดอร์สำหรับการสั่งงานมอเตอร์แบบไม่อัตโนมัติ (Manual)

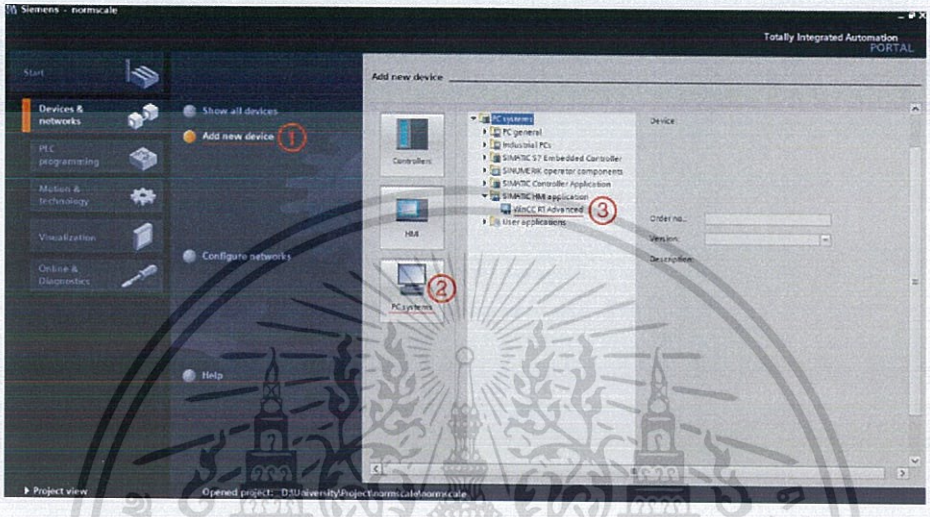
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 โปรแกรม WinCC RT Advanced

WinCC Runtime Advanced V13 เป็นโปรแกรมที่ใช้สำหรับสร้างหน้ากราฟิกสำหรับติดตาม และควบคุมระบบผ่านหน้าจอคอมพิวเตอร์ หรือที่เรียกว่า Human Interface (HMI)

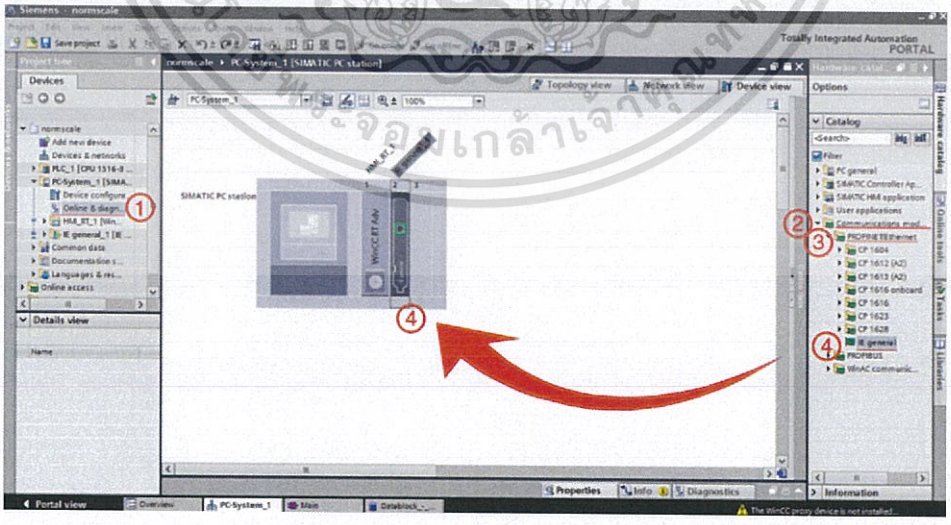
3.3.1 เพิ่ม และตั้งค่าอุปกรณ์

1. กด Add new device > PC System > WinCC RT Advanced



รูปที่ 3.45 หน้าแสดงผล Portal view

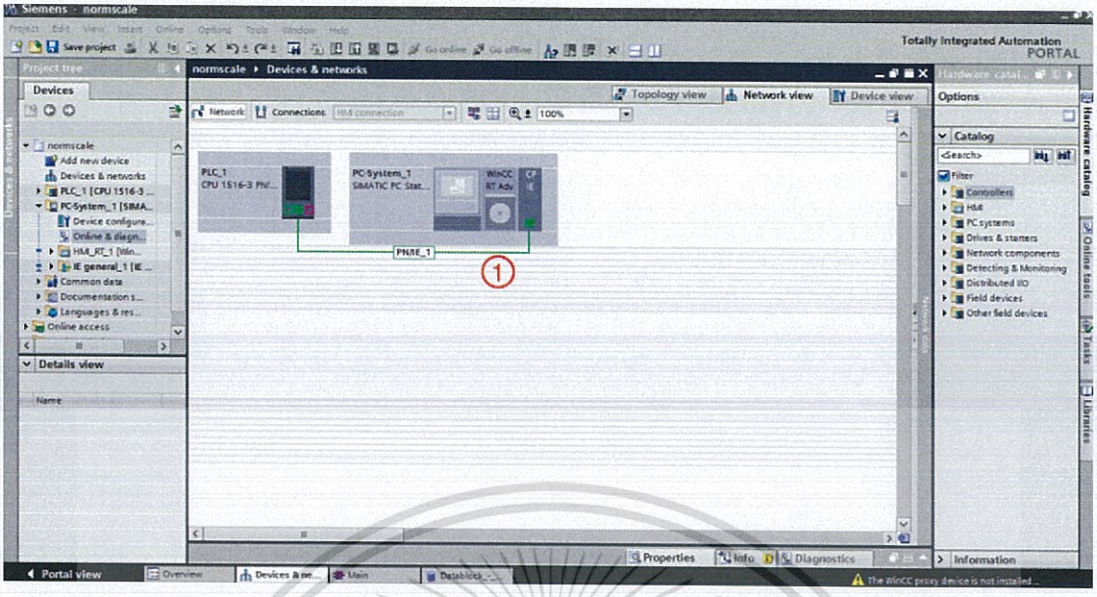
2. กด Online & diagnostics > Communications modules > PROFINET/Ethernet > ลาก IE general ไปไว้ในช่องที่ 2 แล้วกด ที่มุมซ้ายบนของหน้า Device view



รูปที่ 3.46 หน้าแสดงผล Project view

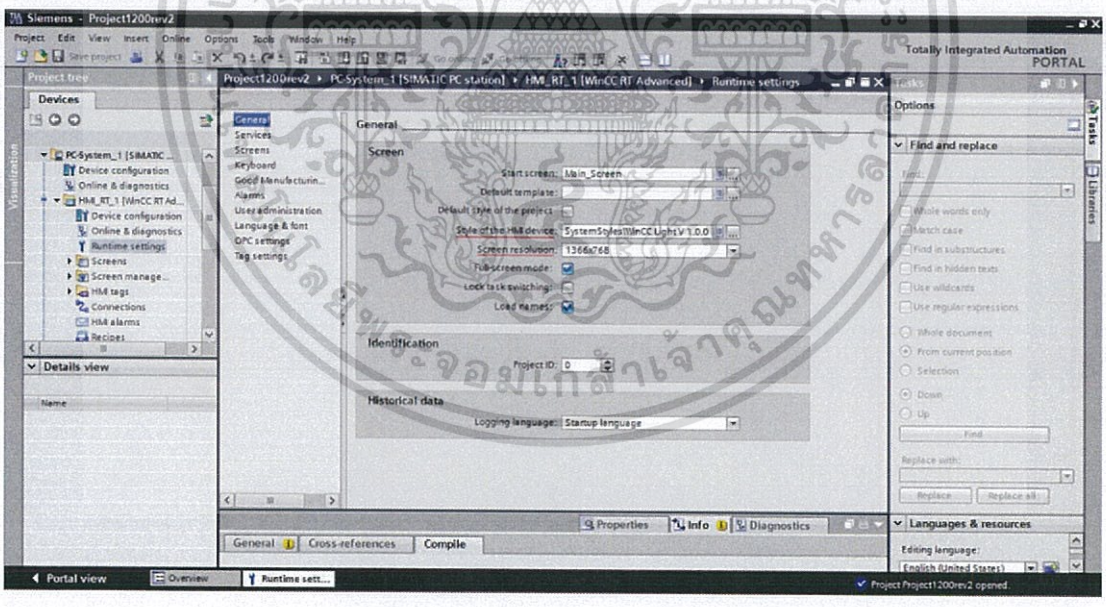
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ลากสายเชื่อมระหว่างพีแอลซีกับ Communications modules



รูปที่ 3.47 หน้า Device & networks แสดงการเชื่อมต่อของอุปกรณ์ต่างๆ

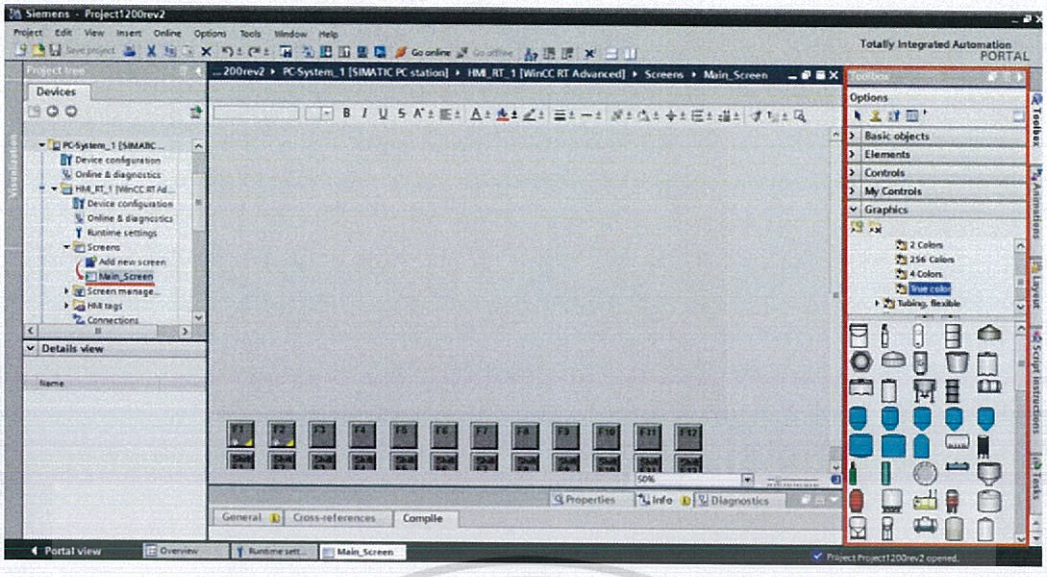
4. ตั้งค่ารูปแบบหน้าจอ (Style of the HMI device) และขนาดหน้าจอ (Screen resolution) ตามต้องการ



รูปที่ 3.48 หน้า Runtime setting สำหรับตั้งค่าหน้าจอ HMI

5. กด Screen > Add new screen จากนั้นสร้างหน้า HMI ตามที่ใช้งาน โดยสามารถใช้กราฟิกที่โปรแกรมมีให้ที่แถบ Toolbox ขวามือ

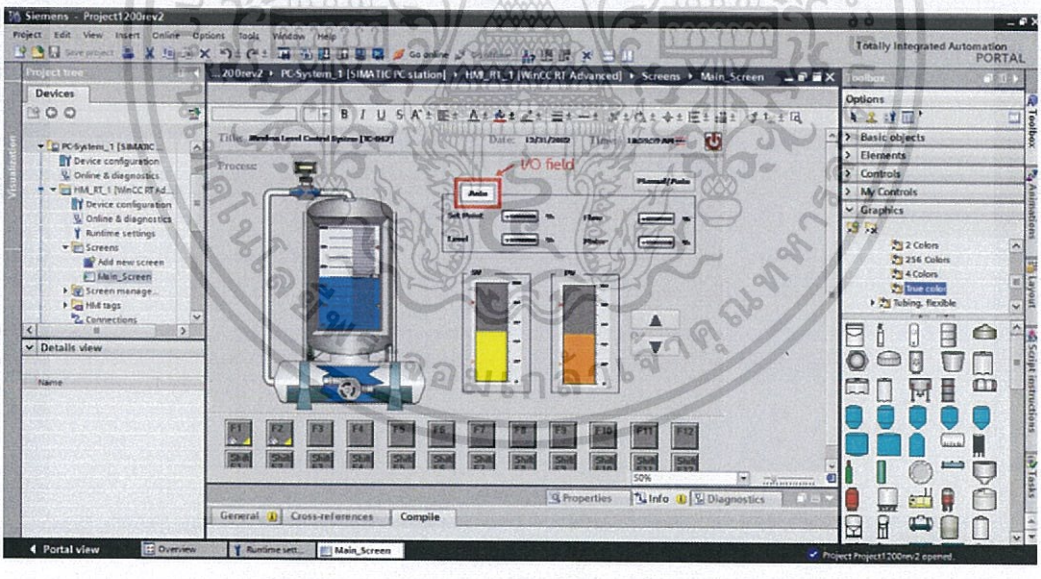
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.49 หน้า Screen สำหรับสร้าง และปรับแต่งหน้า HMI

3.3.2 การสร้าง Symbolic I/O field แสดงสถานะ

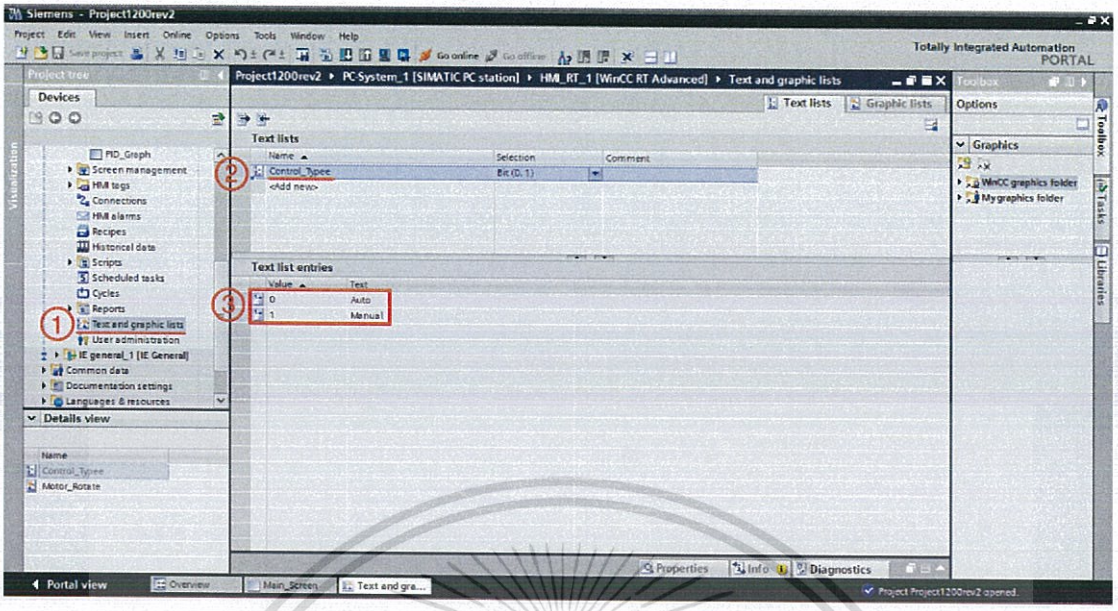
Symbolic I/O field คือ Dynamic text box ซึ่งสามารถนำมาใช้ในการแสดงสถานะได้ตามที่ต้องการ โดยค่าเริ่มต้นของตัว Symbolic I/O field คือ 0 กับ 1 ทั้งนี้สามารถตั้งค่า Text list ให้ Symbolic I/O field ได้ตามที่ต้องการ



รูปที่ 3.50 ตำแหน่งตัวบอกสถานะควบคุม

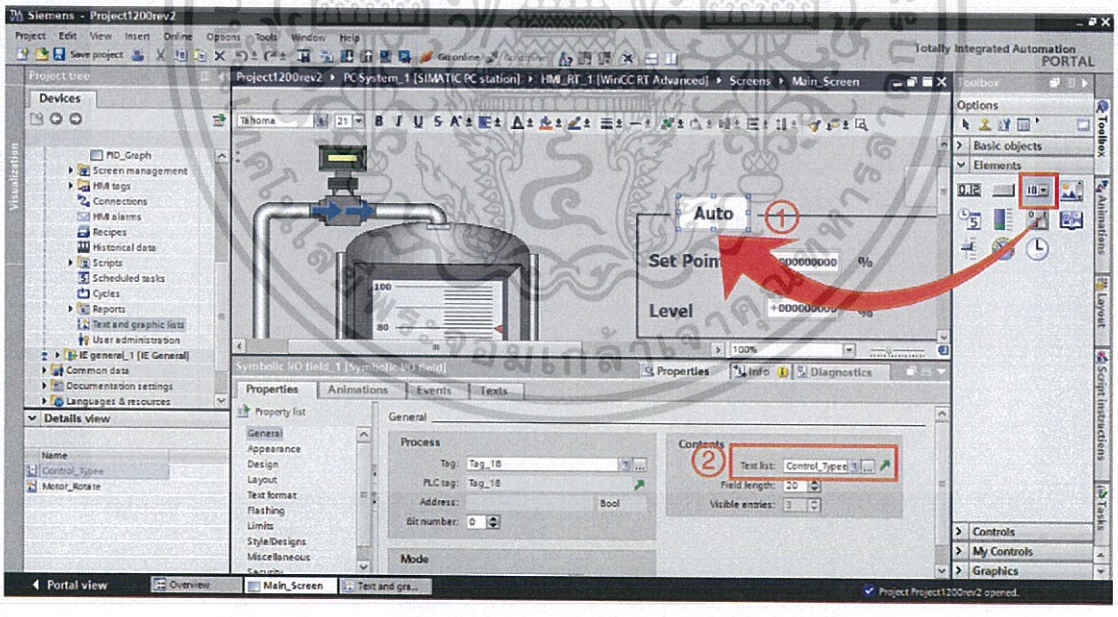
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. สร้าง Text list สถานะที่ต้องการจะใช้ในการแสดงผล



รูปที่ 3.51 หน้า Text and graphic lists

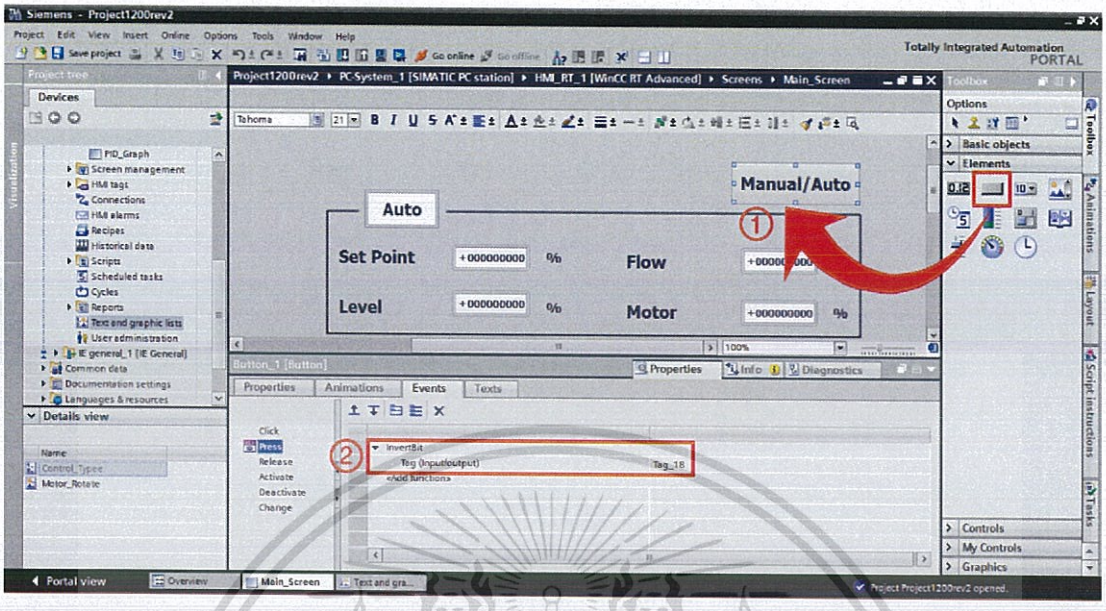
2. ลาก Symbolic I/O field ไปที่พื้นที่หน้าจอสำหรับแสดงผล แล้ว Tag text list : Control_Typee ที่ทำไว้ในข้อ 1



รูปที่ 3.52 การสร้าง Symbolic I/O field และ Tag text list

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

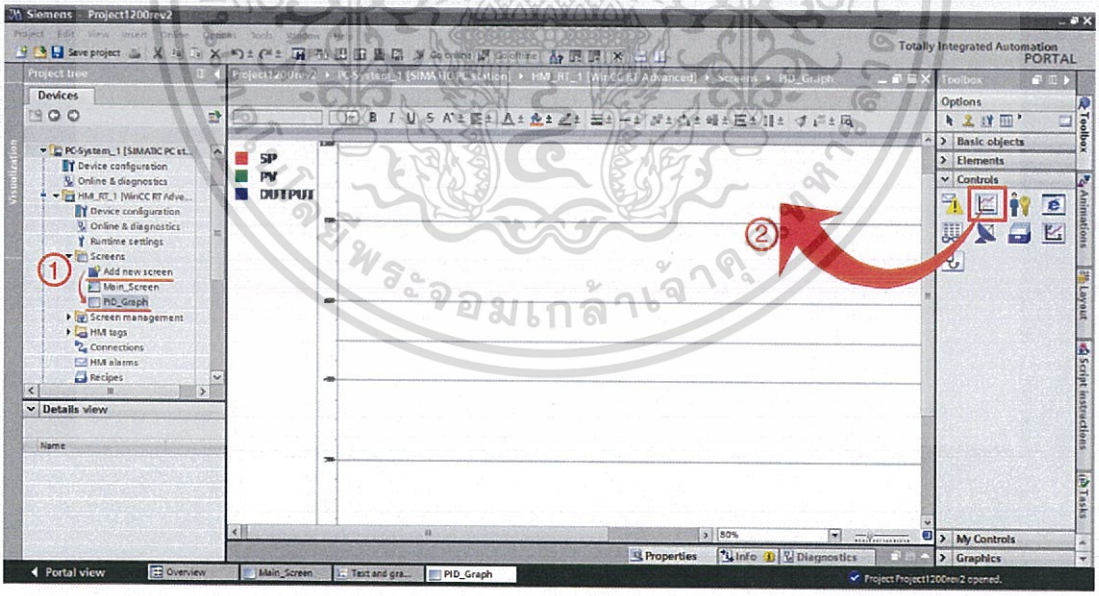
3. สร้าง Button switch สำหรับการเปลี่ยนรูปแบบการควบคุม โดยการลาก Button ไปที่พื้นที่หน้าจอสำหรับแสดงผล แล้ว Tag Event ขณะกดปุ่ม (Press) เป็น InvertBit



รูปที่ 3.53 วิธีการเพิ่ม Event สำหรับ Button switch element

3.3.3 การสร้างกราฟแสดงแนวโน้มบล็อกพีไอดี

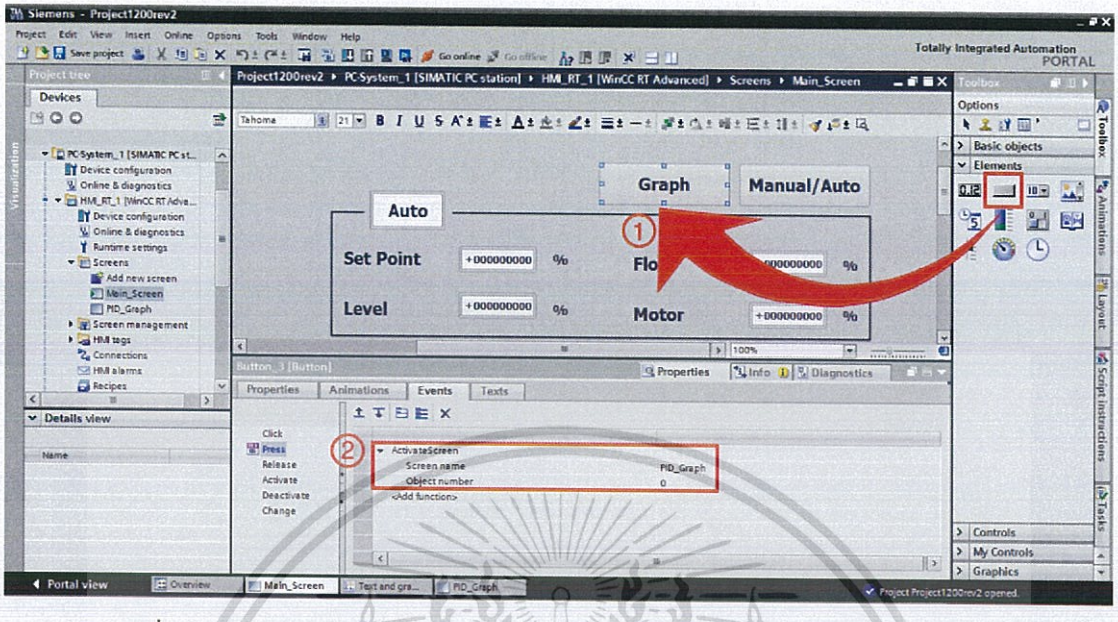
1. เลือก Add new screen > เลือกอุปกรณ์ TrendView



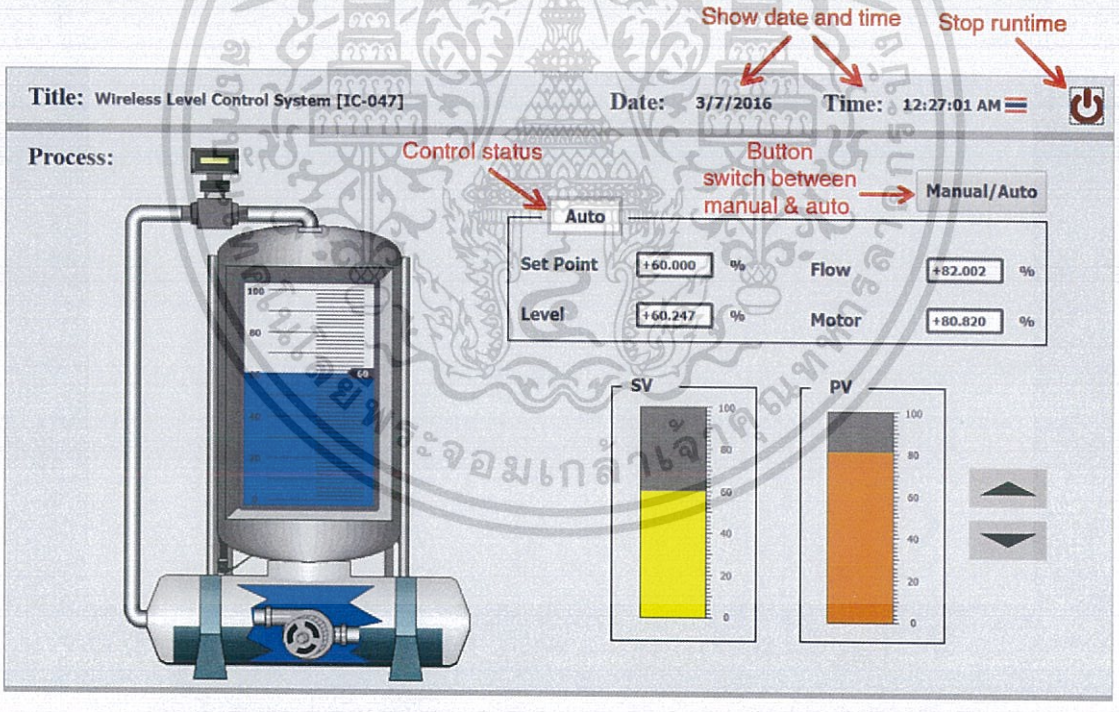
รูปที่ 3.54 การสร้างหน้าแสดงผลแนวโน้มกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สร้างปุ่มสำหรับเรียกใช้งานหน้ากราฟจากหน้า Main_Screen โดยการสร้าง Button แล้วเพิ่ม Event ขณะกดปุ่ม (Press) เป็น ActivateScreen

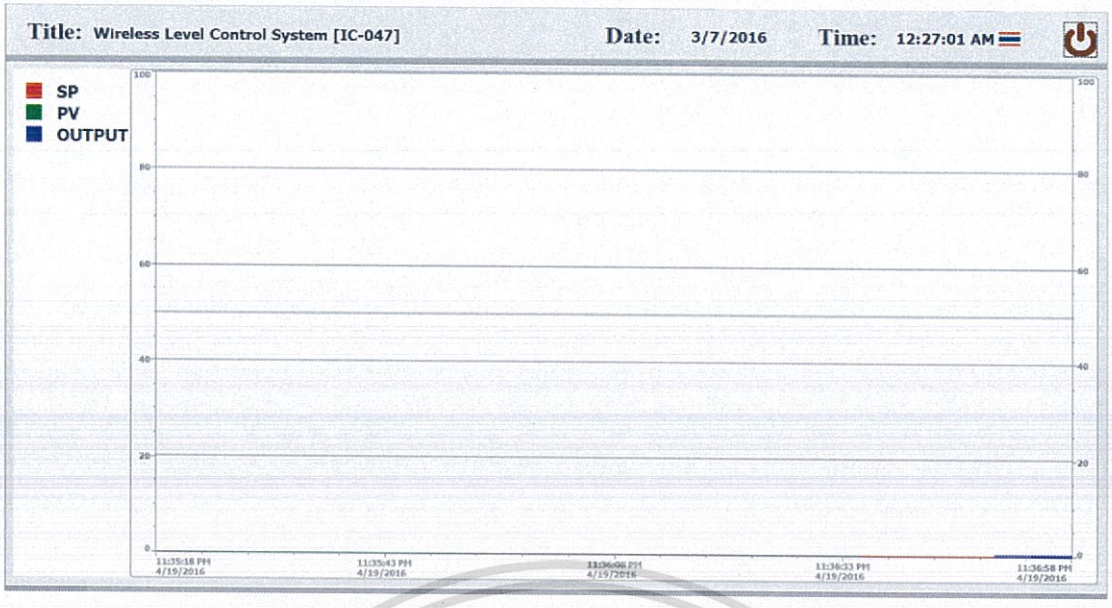


รูปที่ 3.55 การสร้างปุ่มสำหรับเรียกใช้หน้าจอแสดงแนวโน้มกราฟของบล็อกพีไอดี



รูปที่ 3.56 ส่วนประกอบหน้าเชื่อมต่อของระบบควบคุมระดับน้ำแบบไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.57 รูปแบบหน้าต่างแสดงกราฟแนวโน้มของบล็อกพีไอดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

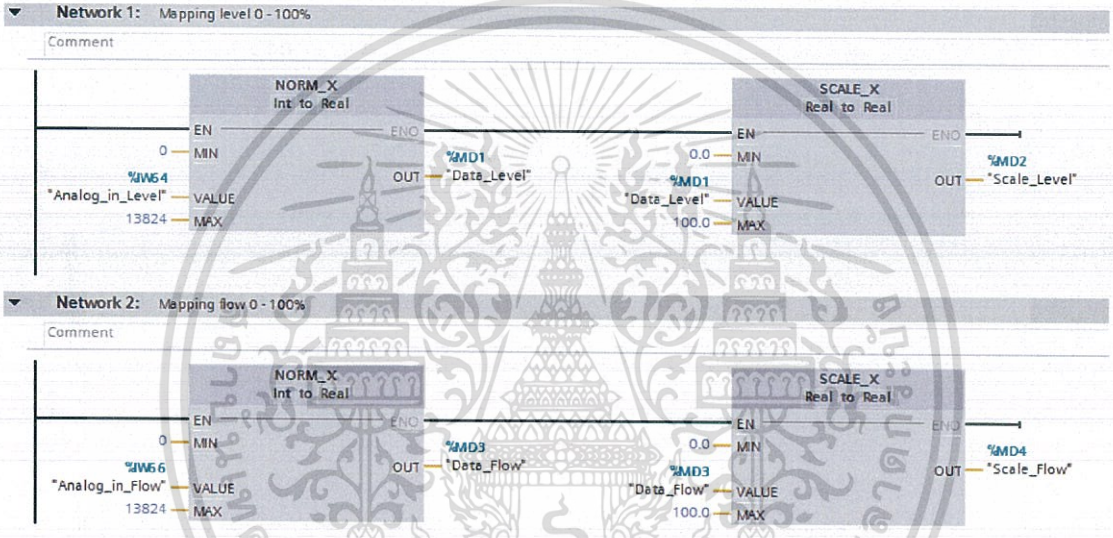
บทที่ 4

การทดลองและผลการทดลอง

4.1 ขั้นตอนการทดลอง

4.1.1 ทดลองการรับค่าที่ขาแอนะล็อกของพีแอลซี โดยบล็อก NORM_X และ SCALE_X

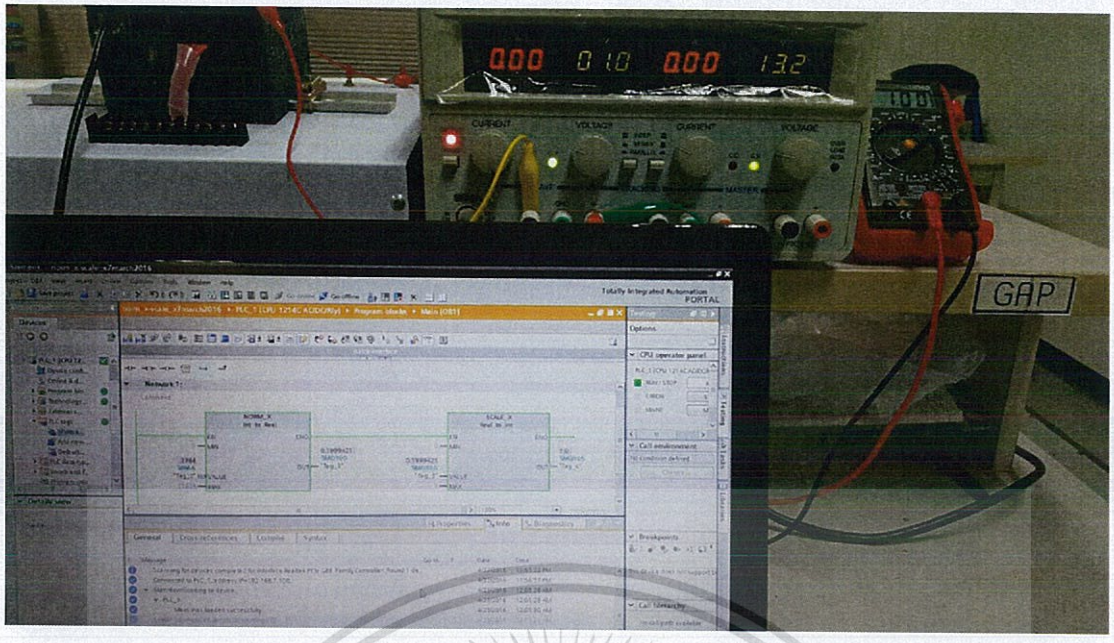
1. เขียนบล็อก NORM_X และ SCALE_X เพื่อทดลองการรับ และอ่านค่าจากขาแอนะล็อกขาเข้า IW64 และ IW66 โดยค่าโวลต์ 0–5 V จะมีค่าอยู่ในช่วง 0–13824 จากนั้นคลิกสัญลักษณ์ Download to device ที่แถบ Toolbar เพื่ออัพโหลดโปรแกรมลงพีแอลซี



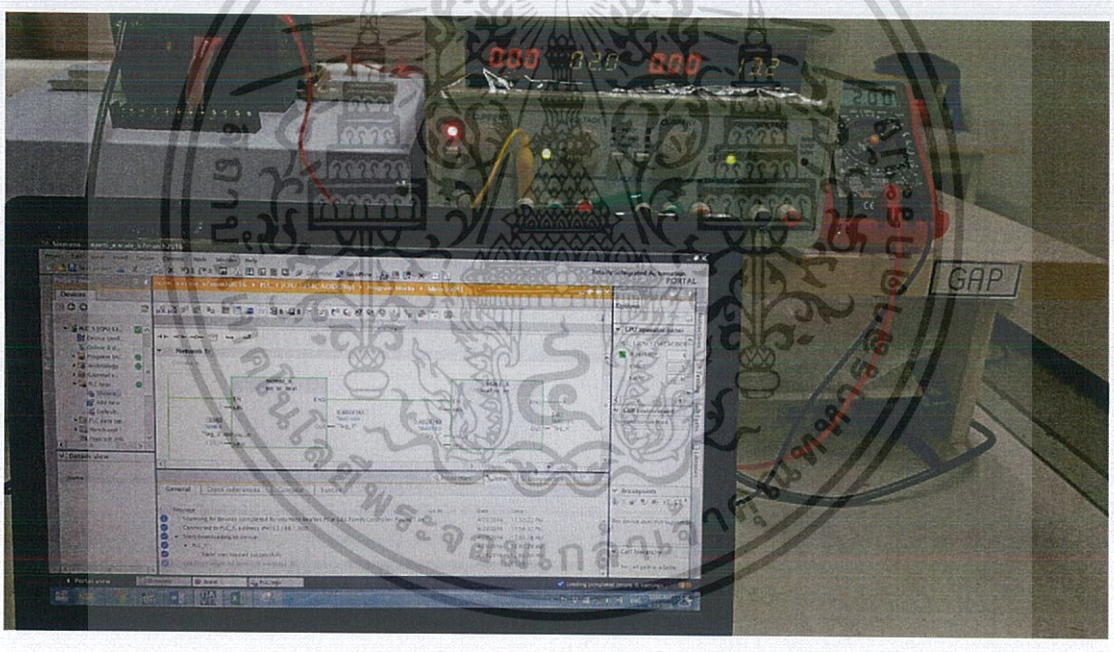
รูปที่ 4.1 บล็อก NORM_X & SCALE_X พร้อมใช้งาน และทดสอบ

2. กดสัญลักษณ์ Monitoring on/off เพื่อทดลองมอนิเตอร์ค่าโวลต์ตามที่ป้อนเข้าขาอินพุตโดยใช้ฟังก์ชันเจนเนอร์เรเตอร์ (Function Generator) โดยทดลองค่าตั้งแต่ 0–5 โวลต์หรือ 0–13824

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

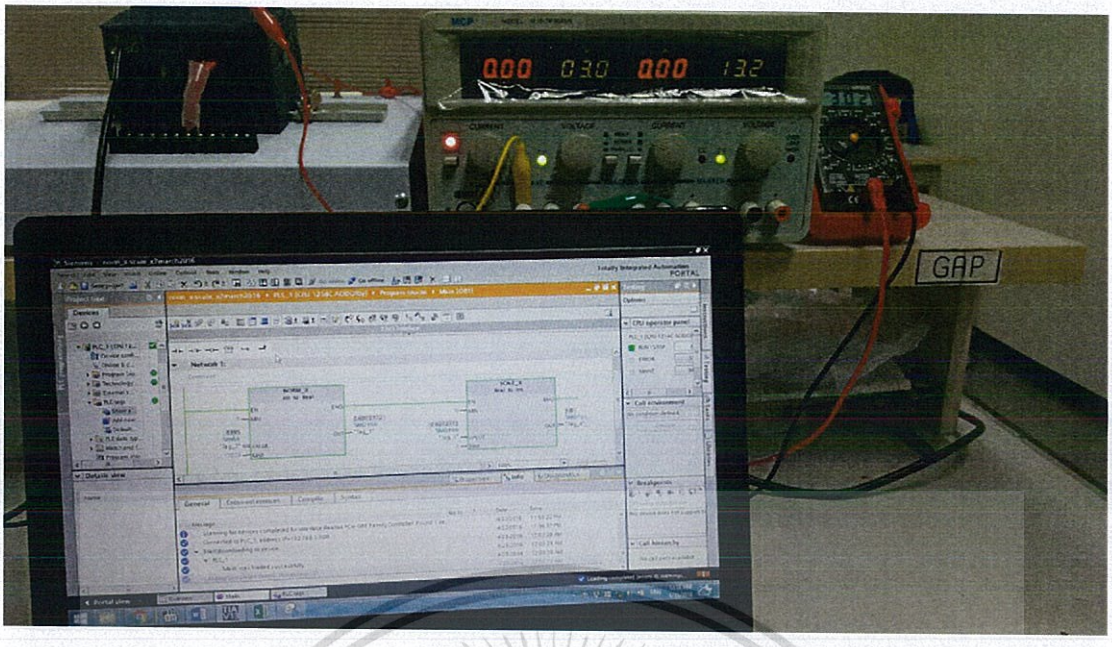


รูปที่ 4.2 ทดลองรับค่าที่ 1 โวลต์

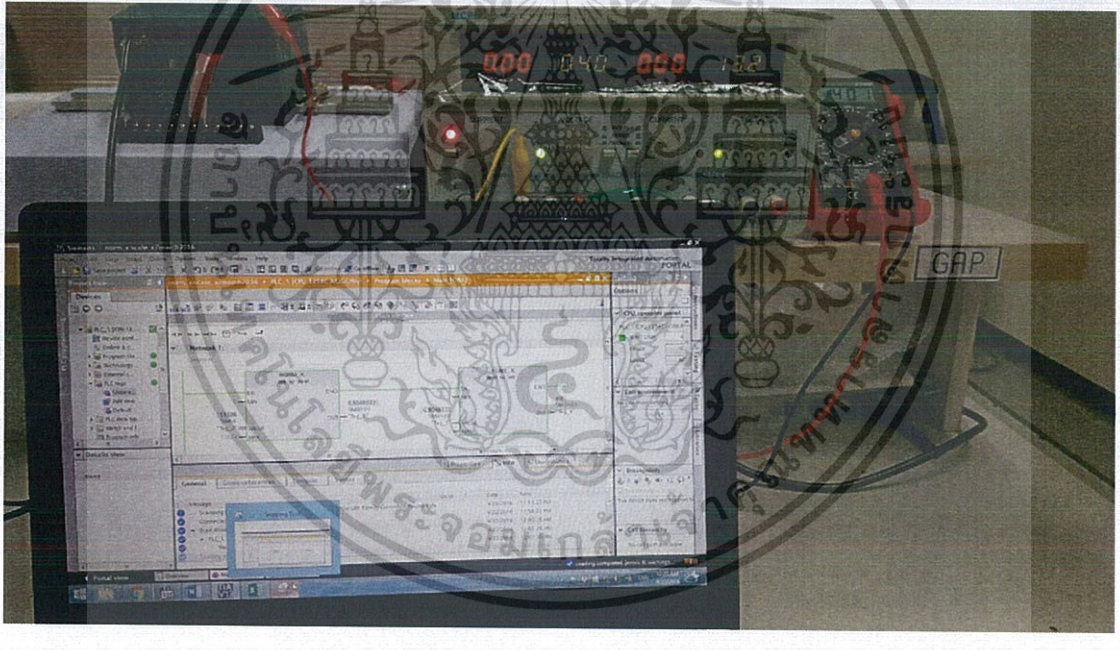


รูปที่ 4.3 ทดลองรับค่าที่ 2 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

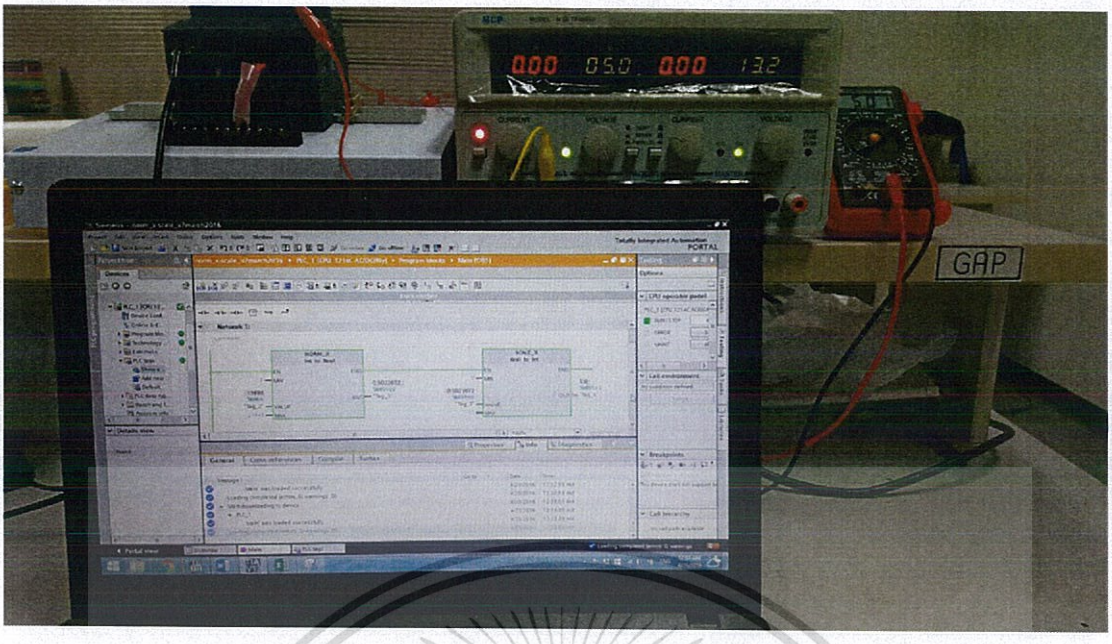


รูปที่ 4.4 ทดลองรับค่าที่ 3 โวลต์



รูปที่ 4.5 ทดลองรับค่าที่ 4 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 ทดลองรับค่าที่ 5 โวลต์

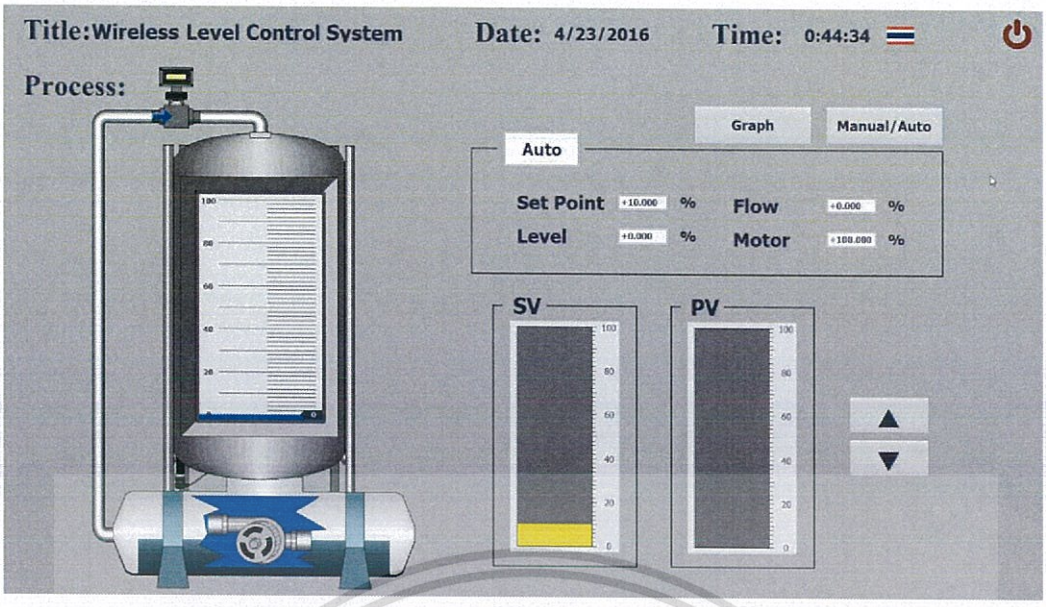
ตารางที่ 4.1 ตารางผลการทดลองการรับค่าโวลต์ที่ขาแอนะล็อกพีแอลซี

Exact value		Trial value					
Function generator (V)	PLC rate range	Voltmeter (V)	Error (%)	TIA Portal V13	Error (%)	PLC rate at pin IW66	Error (%)
1.0	2765	1.00	0	1.0	0	2764	0.04
2.0	5530	2.00	0	2.0	0	5563	0.59
3.0	8295	3.02	0.67	3.0	0	8395	1.20
4.0	11060	4.01	0.25	4.0	0	11126	0.59
5.0	13824	5.01	0.2	5.0	0	13890	0.48

4.1.2 ผลการทดลองเปลี่ยนค่าเป้าหมายของระบบเพื่อหา %Overshoot และ $\%e_{ss}$

เปิดการทำงาน และควบคุมระบบผ่านหน้าจอ HMI ปรับค่าเป้าหมายตามต้องการ โดยจะเปลี่ยนค่าเป้าหมายทีละ 10 เริ่มจาก 0 ถึง 100 และ จาก 100 ถึง 0 จากนั้นสังเกตกราฟผลตอบสนองที่ได้ของระบบเพื่อหาค่าเปอร์เซ็นต์พุ่งเกิน (%Overshoot) และเปอร์เซ็นต์ความคลาดเคลื่อนที่สภาวะคงที่ (Steady State Error, $\%e_{ss}$) ที่ค่าเป้าหมายซึ่งตัวเลขที่ต้องการของทั้งสองค่าจะต้องมีค่าใกล้เคียงกับ 0 ที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 หน้าจอ HMI เมื่อปรับค่าเป้าหมายจาก 0 เป็น 10

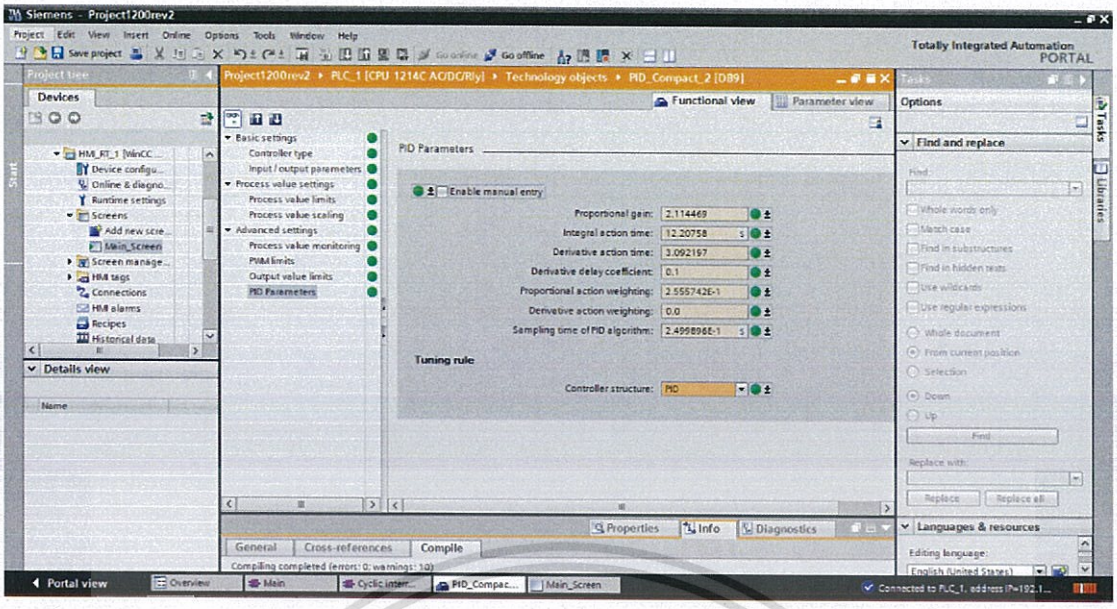
โดยใช้ค่าพารามิเตอร์พีไอดีของคอนโทรลเลอร์ตัวหลัก ดังนี้

Proportional gain = 2.114469
Integral action time = 12.20758 s
Derivative action time = 3.092197 s
Derivative delay coefficient = 0.1 s
Proportional action weighting = 2.555742E-1
Derivative action weighting = 0.0
Sampling time of PID algorithm = 2.499896E-1 s

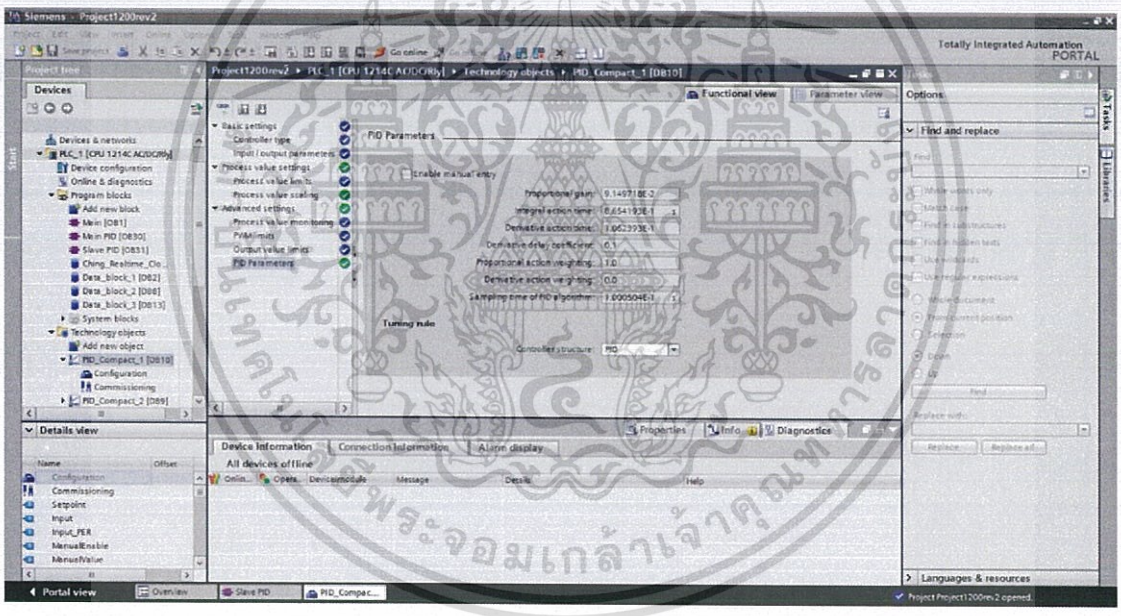
โดยใช้ค่าพารามิเตอร์พีไอดีของคอนโทรลเลอร์ตัวรอง ดังนี้

Proportional gain = 9.149718E-2
Integral action time = 8.654193E-1 s
Derivative action time = 1.062393E-1 s
Derivative delay coefficient = 0.1 s
Proportional action weighting = 1.0
Derivative action weighting = 0.0
Sampling time of PID algorithm = 1.000504E-1 s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

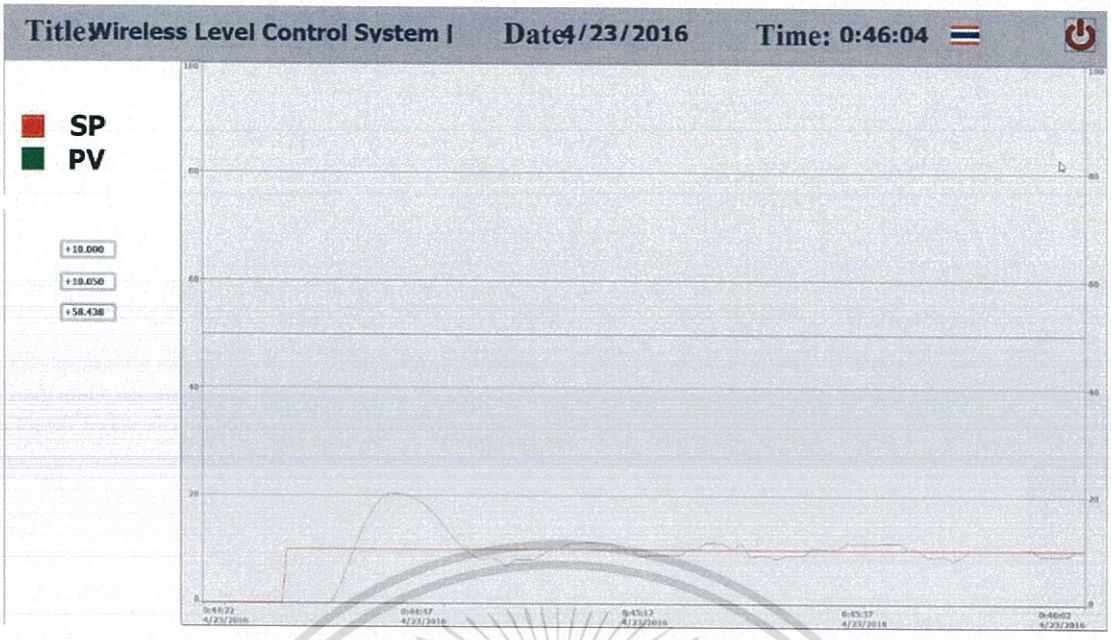


รูปที่ 4.8 ค่าพารามิเตอร์พีไอดีของคอนโทรลเลอร์ตัวหลัก

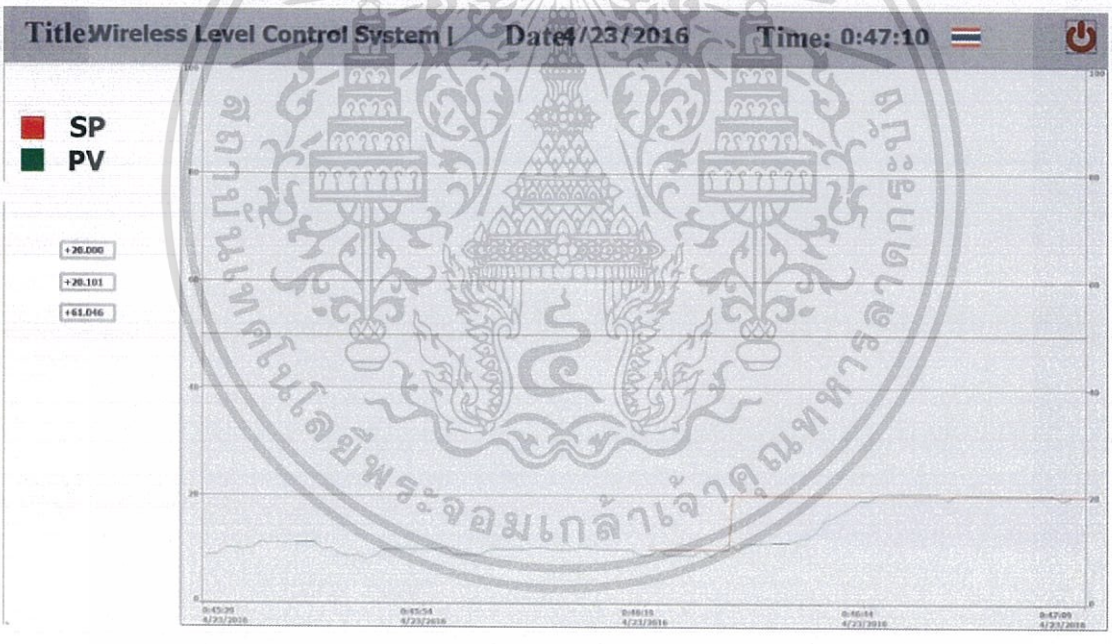


รูปที่ 4.9 ค่าพารามิเตอร์พีไอดีของคอนโทรลเลอร์ตัวรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

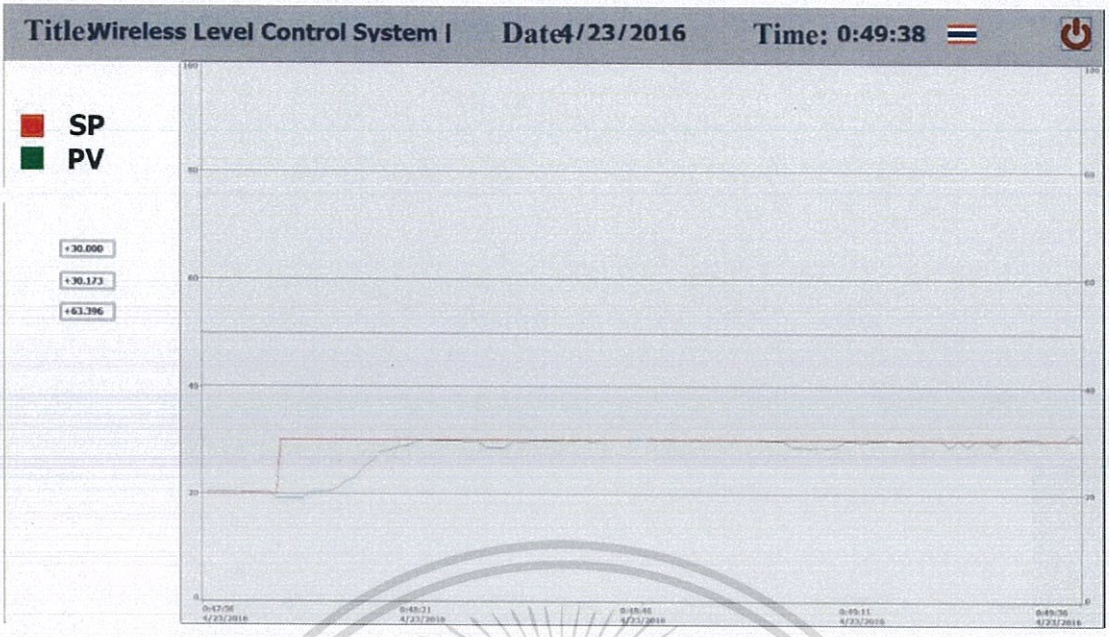


รูปที่ 4.10 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 0 เป็น 10

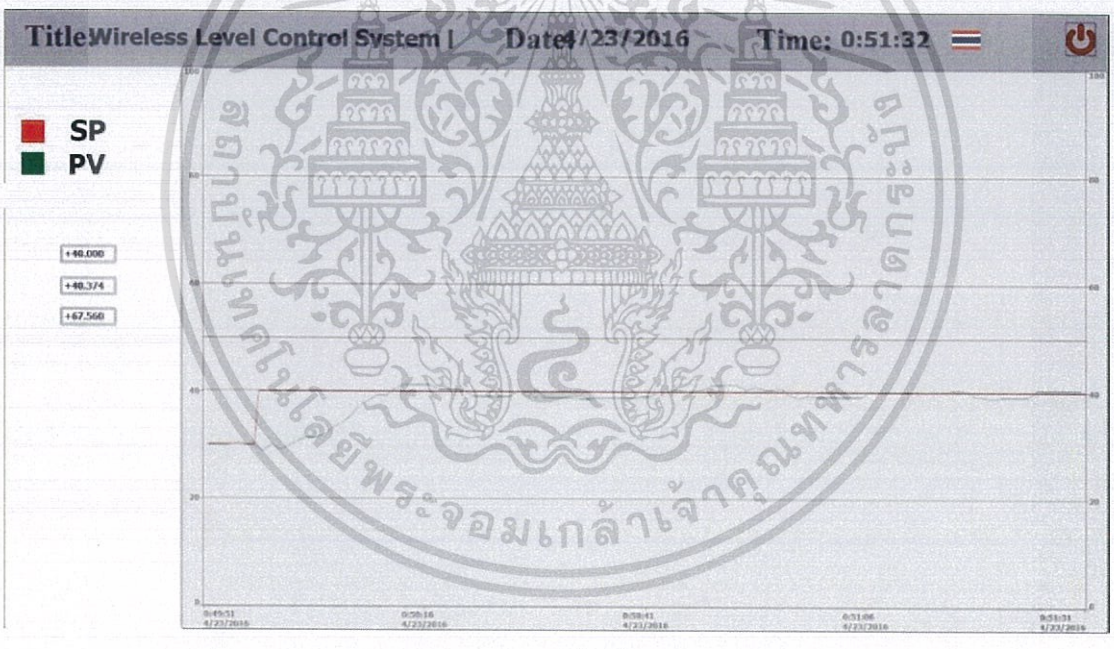


รูปที่ 4.11 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 10 เป็น 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

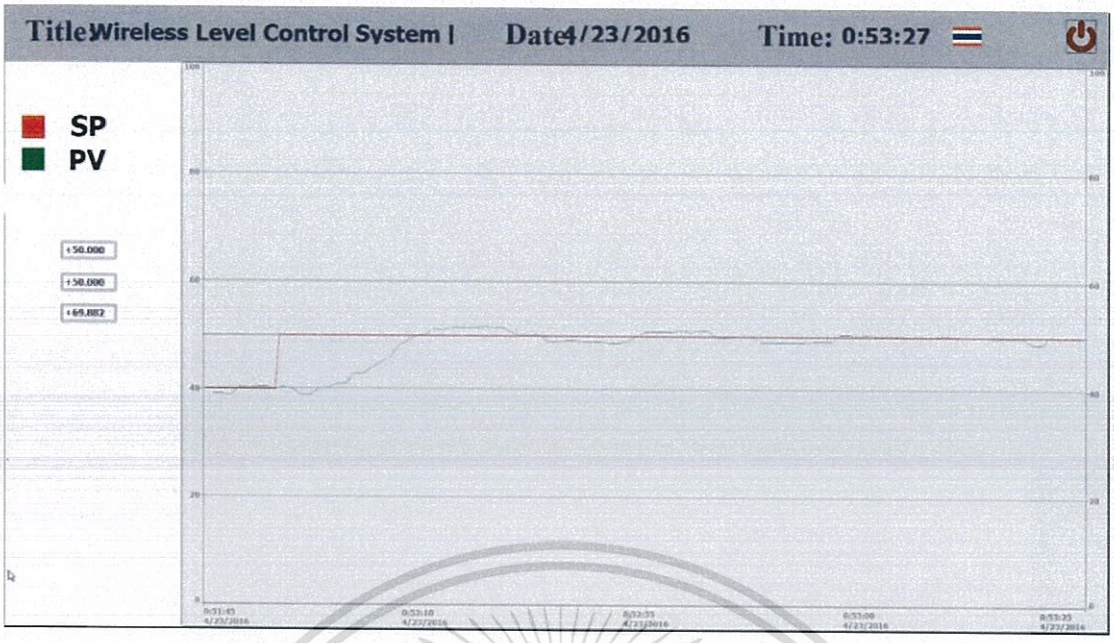


รูปที่ 4.12 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 20 เป็น 30



รูปที่ 4.13 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 30 เป็น 40

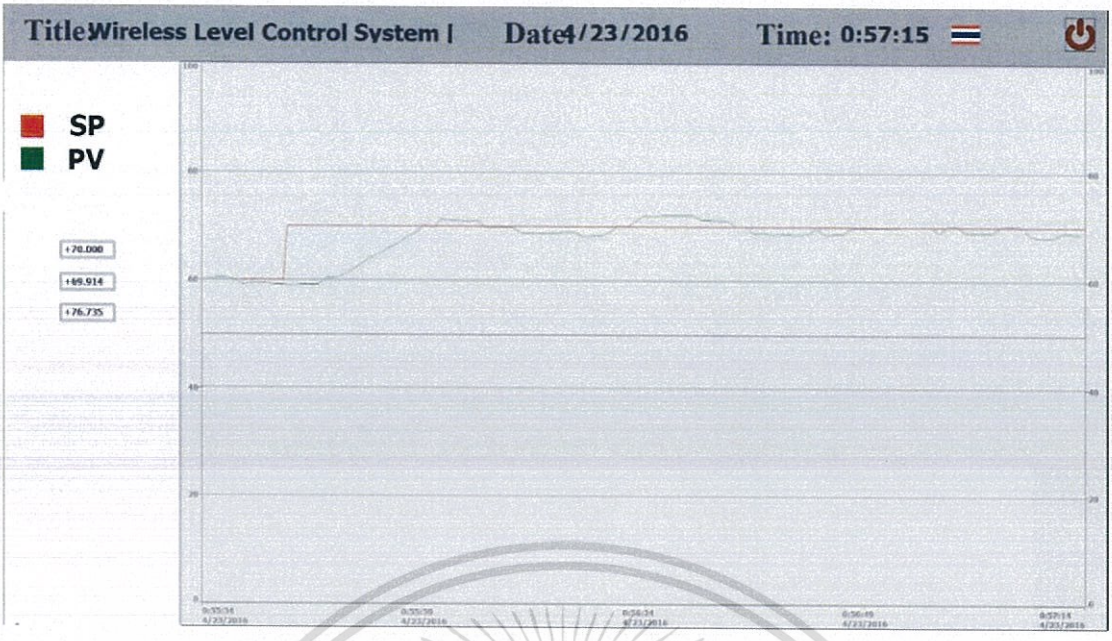
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



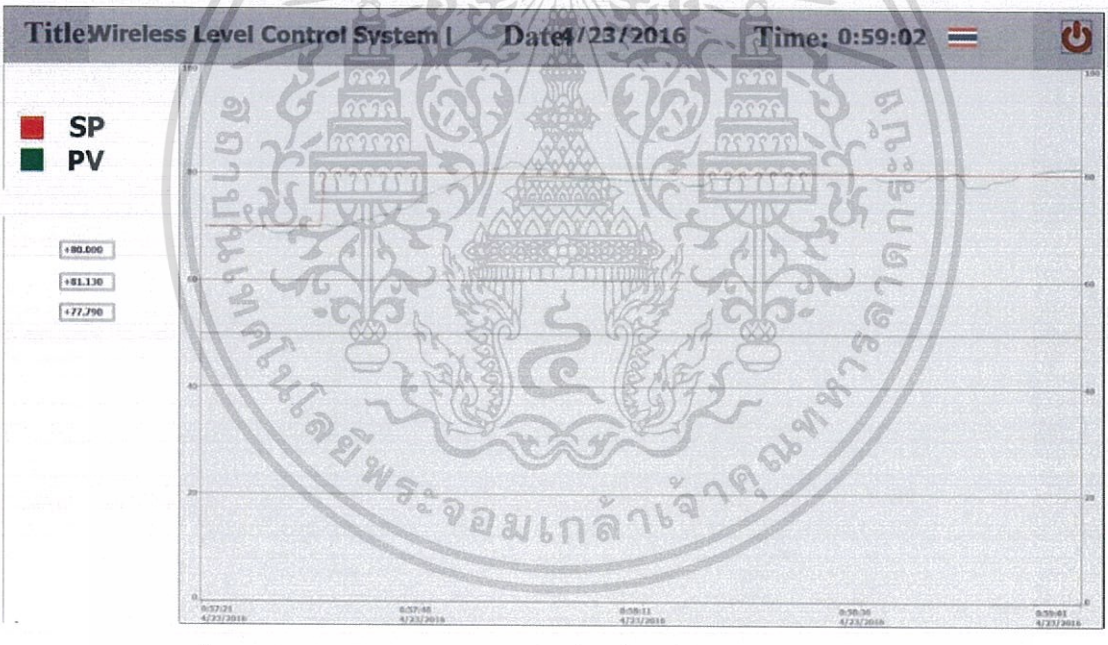
รูปที่ 4.14 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 40 เป็น 50

รูปที่ 4.15 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 50 เป็น 60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

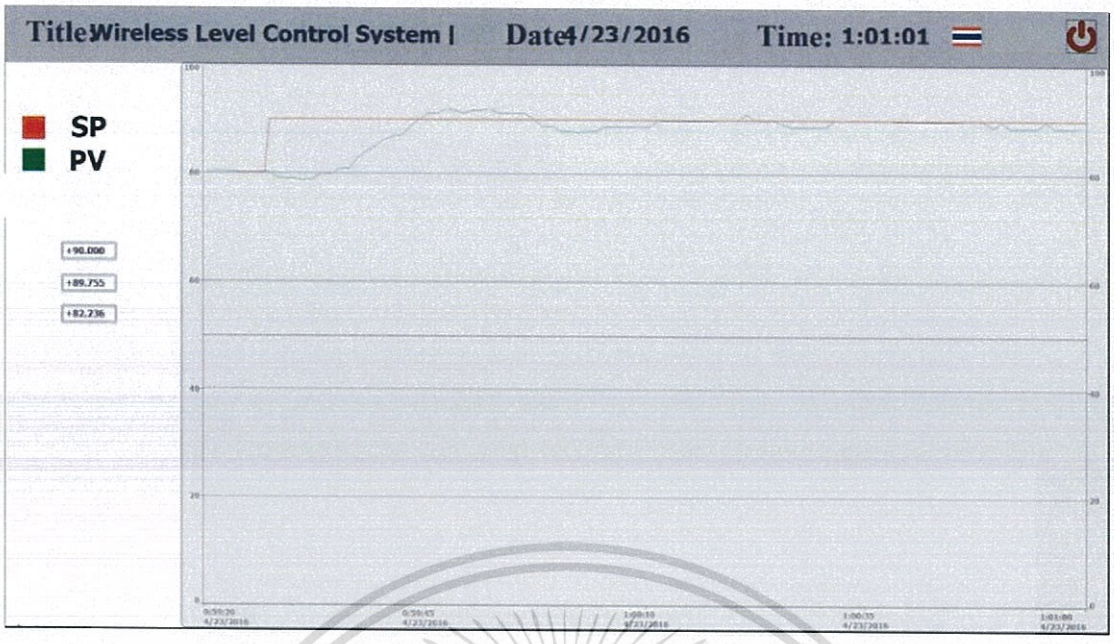


รูปที่ 4.16 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 60 เป็น 70

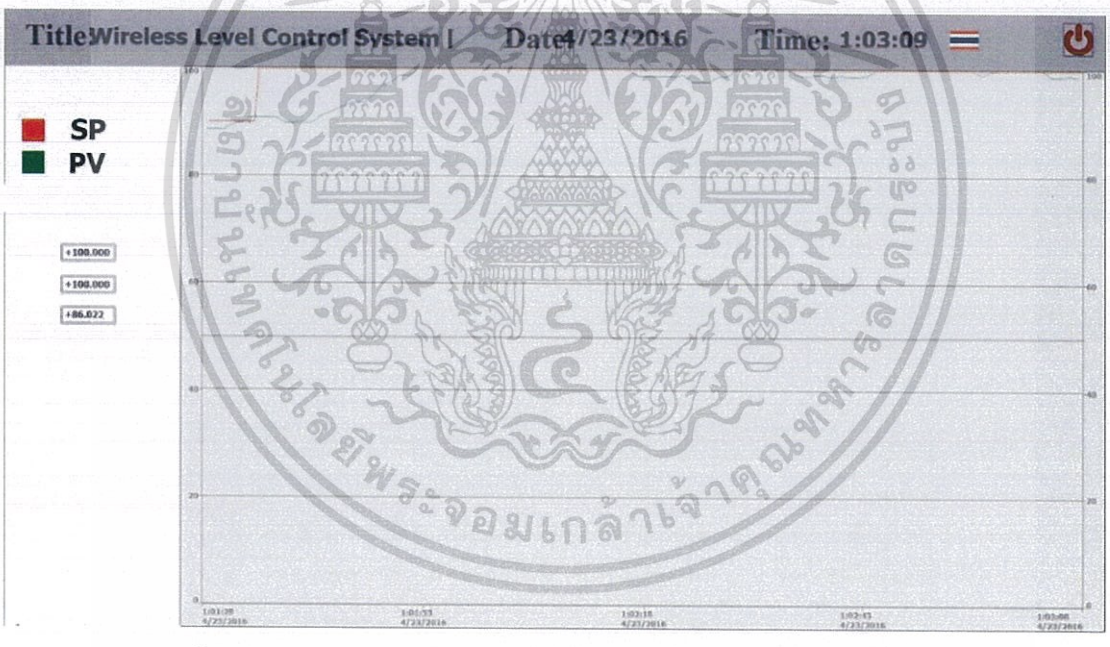


รูปที่ 4.17 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 70 เป็น 80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

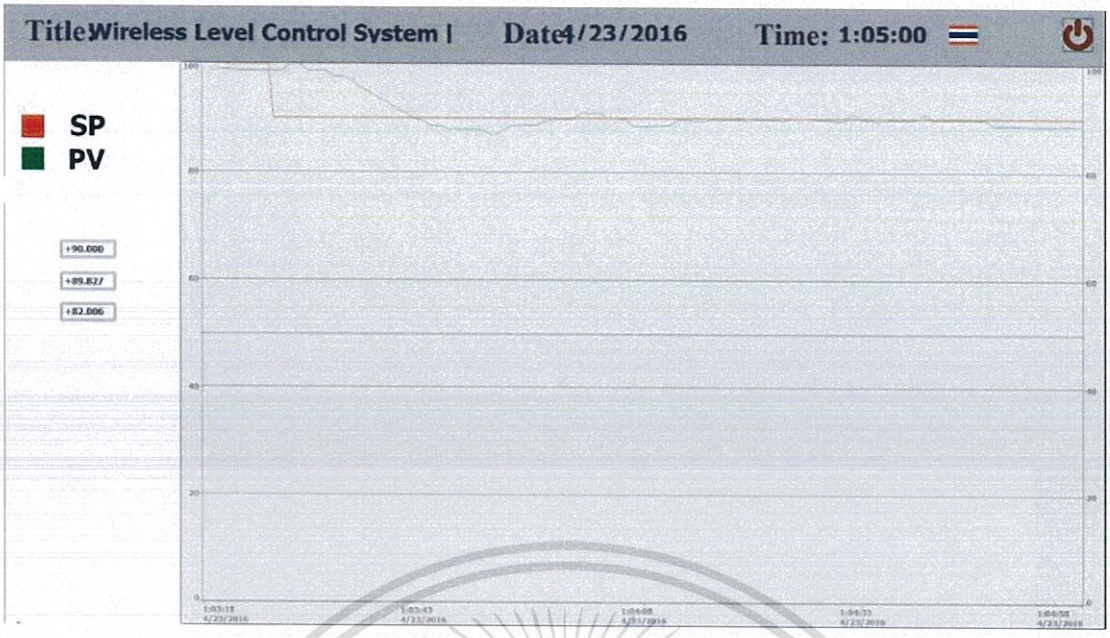


รูปที่ 4.18 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 80 เป็น 90

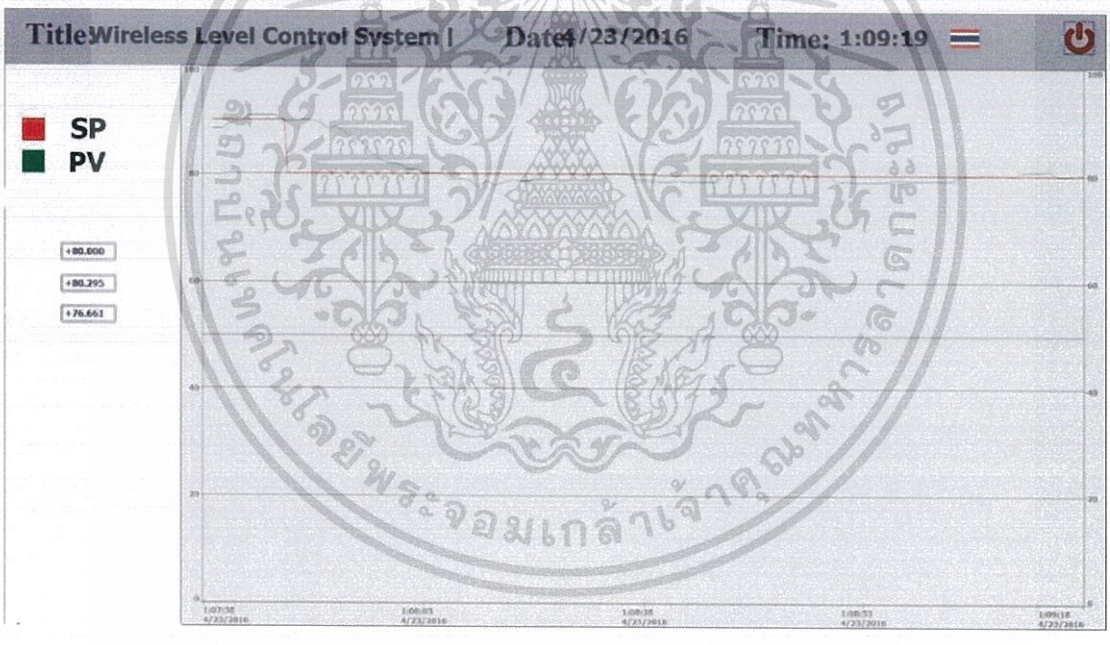


รูปที่ 4.19 กราฟแสดงผลตอบสนองเมื่อเพิ่มค่าเป้าหมายจาก 90 เป็น 100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

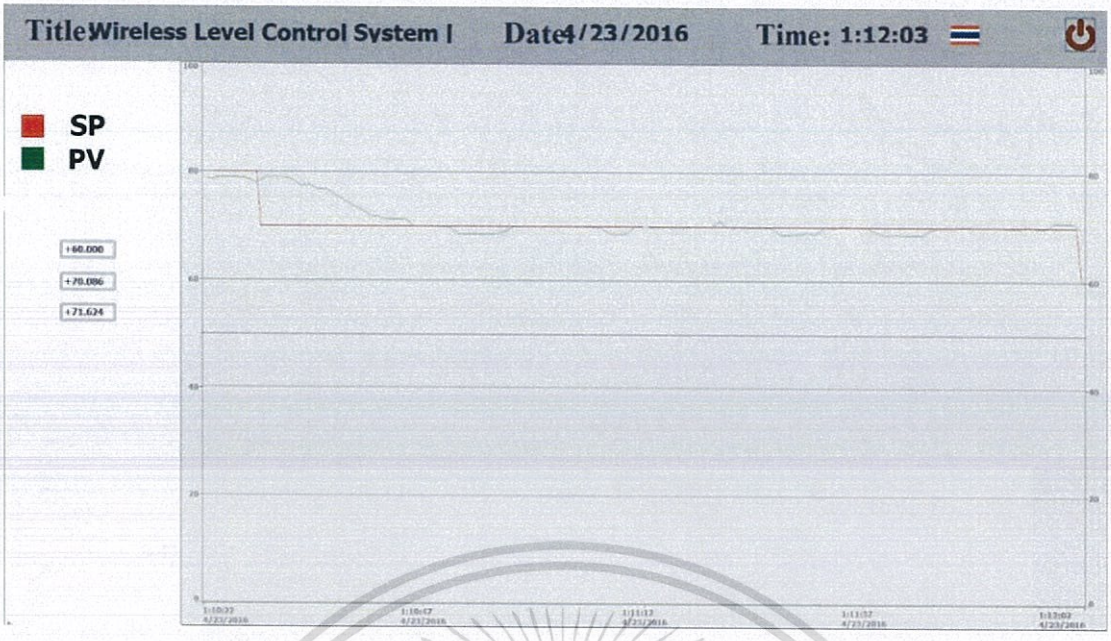


รูปที่ 4.20 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 100 เป็น 90

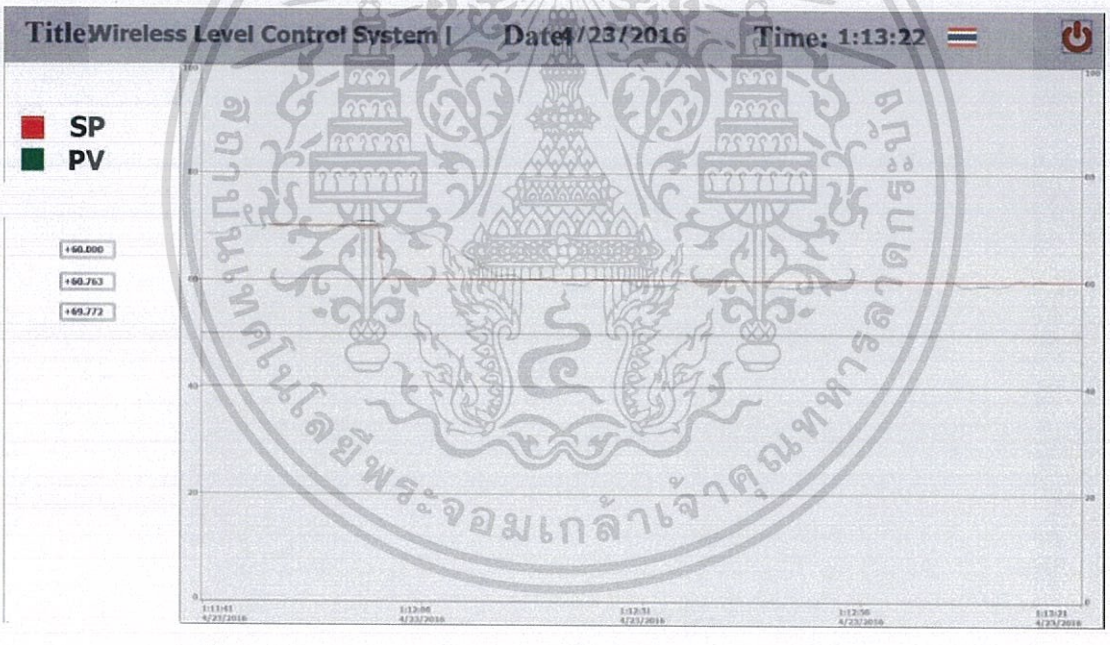


รูปที่ 4.21 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 90 เป็น 80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

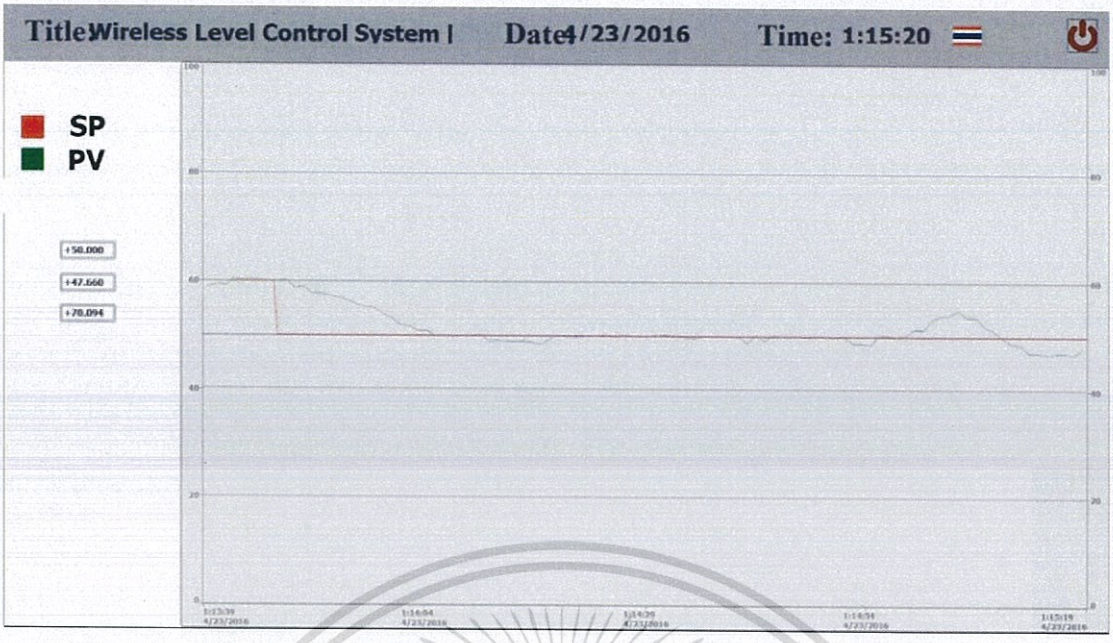


รูปที่ 4.22 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 80 เป็น 70

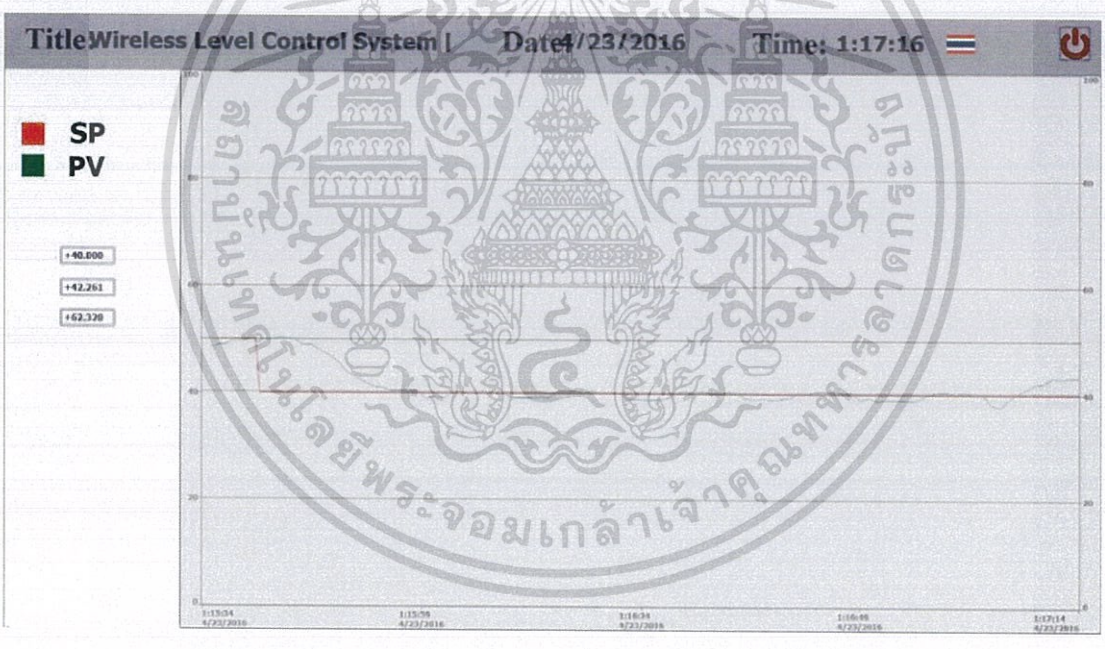


รูปที่ 4.23 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 70 เป็น 60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

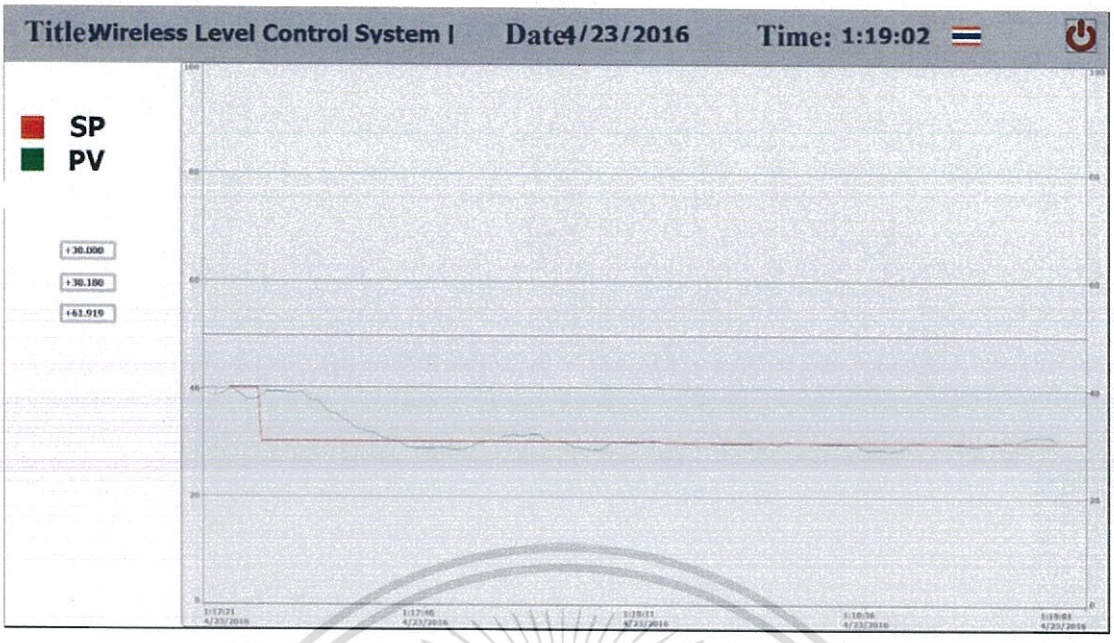


รูปที่ 4.24 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 60 เป็น 50

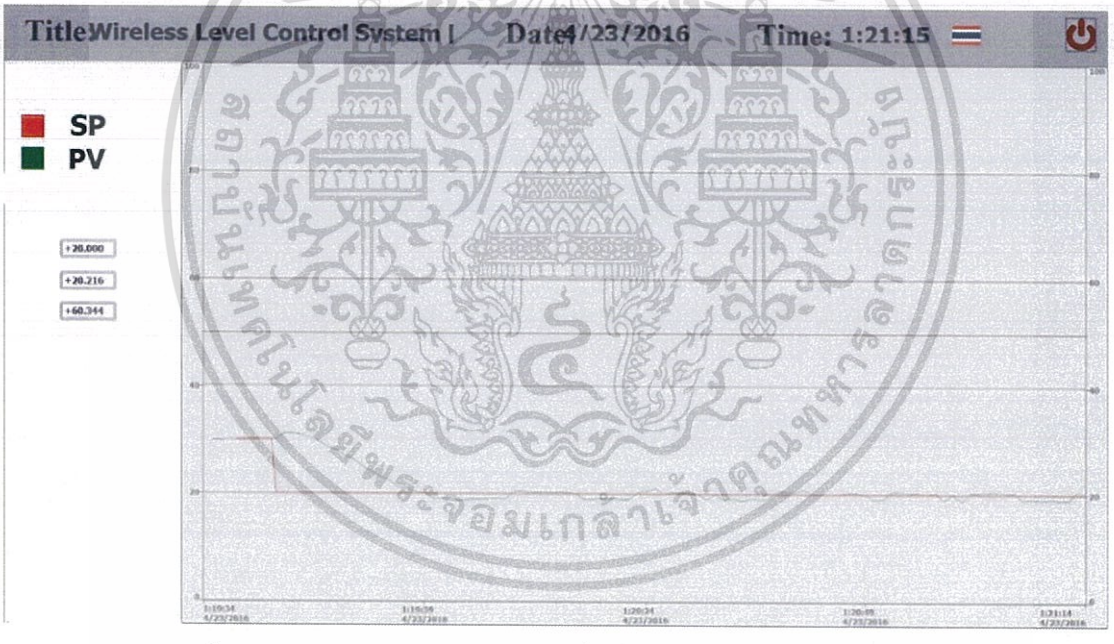


รูปที่ 4.25 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 50 เป็น 40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

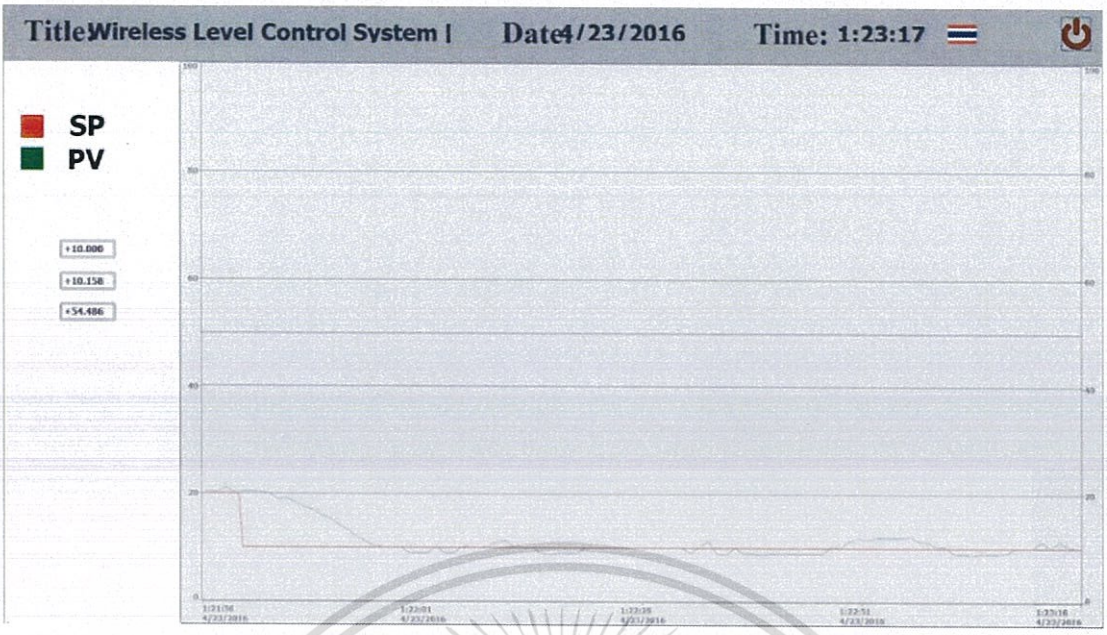


รูปที่ 4.26 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 40 เป็น 30

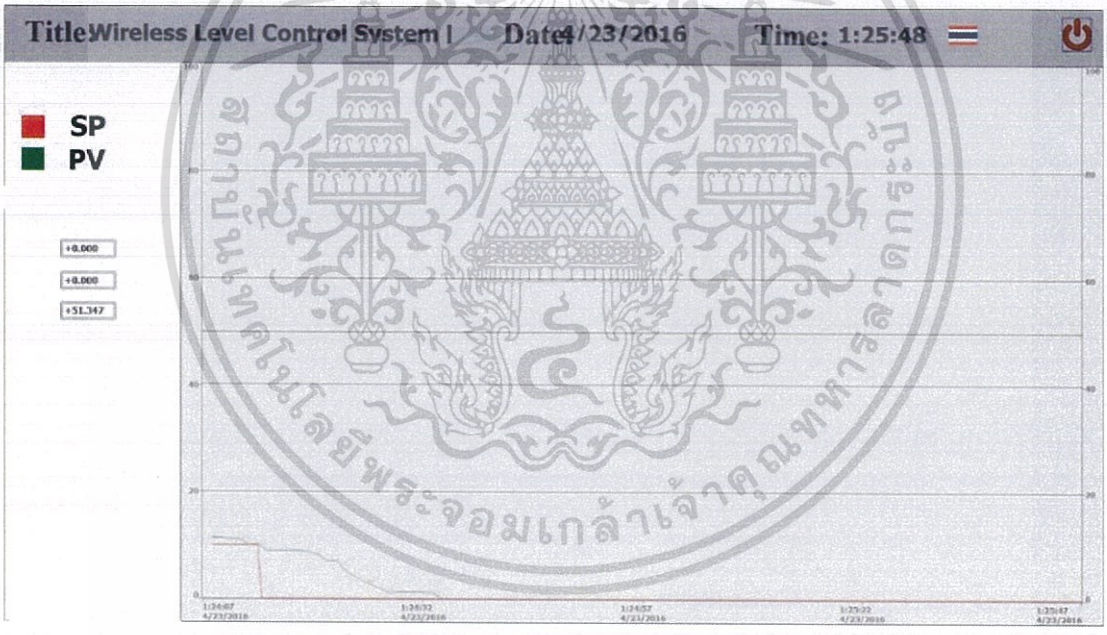


รูปที่ 4.27 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 30 เป็น 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.28 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 20 เป็น 10



รูปที่ 4.29 กราฟแสดงผลตอบสนองเมื่อลดค่าเป้าหมายจาก 10 เป็น 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ผลการทดลองเปลี่ยนค่าเป้าหมายของระบบเพื่อหา %Overshoot และ $\%e_{ss}$

ค่าเป้าหมายก่อนหน้า	ค่าเป้าหมายที่ต้องการ	Overshoot	ค่า PV ที่ Steady State	%Overshoot	$\%e_{ss}$
0	10	20.331	10	103.31	0
10	20	20.511	20	2.555	0
20	30	-	30	0	0
30	40	41.367	40	3.417	0
40	50	51.591	50	3.182	0
50	60	60.079	60	0.132	0
60	70	71.116	70	1.594	0
70	80	82.073	80	2.591	0
80	90	91.901	90	2.112	0
90	100	-	100	0	0
100	90	86.595	90	3.783	0
90	80	77.768	80	2.79	0
80	70	68.524	70	2.108	0
70	60	58.596	60	2.34	0
60	50	48.546	50	2.908	0
50	40	38.855	40	2.908	0
40	30	28.733	30	4.223	0
30	20	18.826	20	5.87	0
20	10	8.927	10	10.73	0
10	0	-	0	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการดำเนินงาน ปัญหา และแนวทางการปรับปรุงโครงงาน

5.1 สรุปผลการดำเนินงาน

จากผลการทดลองสรุปได้ว่าพีแอลซีสามารถรับส่งข้อมูลผ่าน WiFi จากเซนเซอร์พร้อมกัน 2 ตัว ได้โดยการรับค่าแรงดันไฟฟ้าจากตัว Module WiFi ที่ต่อเข้าขาแอนะล็อกพีแอลซีได้โดยมีความสูญเสียแรงน้อยมาก นอกจากนี้สรุปว่าค่าพารามิเตอร์ของตัวควบคุมหลักที่ P = 2.114469, T_i = 12.20758 s และ T_d = 3.092197 s และค่าพารามิเตอร์ตัวควบคุมรองที่ P = 9.149718E-2, T_i = 8.654193E-1 s และ T_d = 1.062393E-1 s เป็นค่าที่เหมาะสมสำหรับระบบควบคุมระดับน้ำแบบไร้สายจำลองระบบนี้ เนื่องจากเป็นค่าที่ทำให้ค่าเปอร์เซ็นต์ความผิดพลาดที่สถานะคงตัว หรือ $\%e_{ss}$ กับค่า %Overshoot มีค่าน้อยใกล้เคียง 0 มากที่สุด

5.2 ปัญหา และการแก้ไข

ทางผู้จัดทำได้สรุปปัญหาที่พบในการดำเนินงาน และแนวทางการแก้ไขเป็นลำดับดังตาราง

ตารางที่ 5.1 ปัญหาในการดำเนินงาน และวิธีแก้ไข

ปัญหาในการดำเนินงาน	วิธีการแก้ไขปัญหา
1. โปรแกรม SIMATIC PCS7 ไม่รองรับ PLC Siemens S7-1200	1. เปลี่ยนจากโปรแกรม SIMATIC PCS7 มาใช้โปรแกรม STEP7 TIA Portal V13 แทน
2. โปรแกรม STEP7 TIA Portal V13 สามารถเชื่อมต่อกับ Module WiFi ได้ทีละตัว	2. เพิ่ม Module WiFi เพื่อรับค่าจากตัว Module WiFi ที่ฝั่งระบบจำลอง แล้วต่อเข้ากับขาแอนะล็อกของพีแอลซีโดยตรง
3. ในบางครั้งการเชื่อมต่อมอเตอร์ผ่าน WiFi ไม่ประสบผลสำเร็จ เนื่องจากหมายเลข IP Address ของ Module WiFi มีการเปลี่ยนแปลง	3. ตรวจเช็ค IP Address ของ Module WiFi ให้ตรงกันทุกครั้งก่อนการเชื่อมต่อ
4. ใช้เวลาในการศึกษาค้นคว้า และปฏิบัติงานค่อนข้างมาก เนื่องจากผู้จัดทำยังขาดประสบการณ์ และความชำนาญ	4. ศึกษาค้นคว้าหาข้อมูลเพิ่มเติมให้มากขึ้น และสอบถามอาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ข้อเสนอแนะ

เนื่องจากทางผู้จัดทำใช้ PLC Siemens S7-1200 ที่เหมาะกับงานขนาดเล็ก และอาจมีเสถียรภาพไม่สูงนัก หากต้องการนำไปใช้งานในระบบที่ใหญ่ และต้องการเสถียรภาพสูง ควรเลือกใช้พีแอลซีรุ่นที่เหมาะสมในการใช้งานต่อระบบนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] Hans Berger. (2554). **Automating in STEP 7 Basic with SIMATIC S7-1200**. United State. Wiley

[2] ธีรศิลป์ ทุมวิภาต. (2547). **เรียนรู้ PLC ขั้นกลางด้วยตนเอง**. กรุงเทพมหานคร. ซีเอ็ดยูเคชั่น

[3] **Siemens S7-1200 System Manual**. เข้าถึงได้จาก: http://www.generationrobots.com/media/manuel-plc-siemens-s7-en.pdf (วันที่สืบค้นข้อมูล : 21 กันยายน 2558)

[4] **Technical specification SIMATIC S7-1200**. เข้าถึงได้จาก: http://www.paratrasnet.ro/pdf/automatizari-industriale/S7-1200.pdf (วันที่สืบค้นข้อมูล : 19 ตุลาคม 2558)

[5] **คุณสมบัติของ PLC Siemens model S7-1214C AC/DC/Relay (6ES7 214-1BG310X B0)** เข้าถึงได้จาก: http://docs-europe.electrocomponents.com/webdocs/1129/0900766b81129785.pdf (วันที่สืบค้นข้อมูล : 21 ตุลาคม 2558)

[6] **User Communication**. เข้าถึงได้จาก: https://cache.industry.siemens.com/dl/files/808/67196808/att_108115/v1/net_s7-1200_isoontcp_en.pdf (วันที่สืบค้นข้อมูล : 9 ธันวาคม 2558)

[7] **PID Control with PID_Compact**. เข้าถึงได้จาก: http://www.infoplc.net/files/descargas/siemens/infoPLC_net_100746401_S7-1200_PID_Compact_DOKU_v1d0_ en.pdf (วันที่สืบค้นข้อมูล : 10 มกราคม 2559)

[8] **Analog input specification SIMATIC S7-1200**. เข้าถึงได้จาก: http://www.sahkonumerot.fi/2702074/doc/technicalinfodoc/ (วันที่สืบค้นข้อมูล : 24 มกราคม 2559)

[9] **Cascade PID Control**. เข้าถึงได้จาก: http://thaicontrol.wordpress.com/2012/02/13/cascade-pid-control/ (วันที่สืบค้นข้อมูล : 13 มีนาคม 2559)

[10] **Process control**. เข้าถึงได้จาก: http://www.thailandindustry.com/guru/view.php?id=15688 (วันที่สืบค้นข้อมูล : 13 มีนาคม 2559)

[11] **Fundamentals of cascade control**. เข้าถึงได้จาก: http://www.controleng.com/single-article/fundamentals-of-cascade-control (วันที่สืบค้นข้อมูล : 13 มีนาคม 2559)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

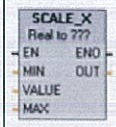
ภาคผนวก ก

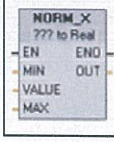
เอกสารข้อมูลประกอบการเขียนแลดเดอร์บล็อกต่างๆ

ก.1 บล็อก NORM_X, SCALE_X

Table 7- 88 SCALE_X and NORM_X instructions

LAD / FBD	SCL	Description

	`out :=SCALE_X(min:=_in_,`
	`value:=_in_,`
	`max:=_in_);`
		Scales the normalized real parameter VALUE where (0.0 <= VALUE <= 1.0) in the data type and value range specified by the MIN and MAX parameters:
		OUT = VALUE (MAX - MIN) + MIN

	`out :=NORM_X(min:=_in_,`
	`value:=_in_,`
	`max:=_in_);`
		Normalizes the parameter VALUE inside the value range specified by the MIN and MAX parameters:
		OUT = (VALUE - MIN) / (MAX - MIN), where (0.0 <= OUT <= 1.0)

[1] For LAD and FBD: Click the "???" and select a data type from the drop-down menu.

Table 7- 89 Data types for the parameters

Parameter	Data type[1]	Description
MIN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Input minimum value for range
VALUE	SCALE_X: Real, LReal NORM_X: SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Input value to scale or normalize
MAX	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Input maximum value for range
OUT	SCALE_X: SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal NORM_X: Real, LReal	Scaled or normalized output value

[1] For SCALE_X: Parameters MIN, MAX, and OUTmust be the same data type.
For NORM_X: Parameters MIN, VALUE, and MAXmust be the same data type.

Note

SCALE_X parameter VALUE should be restricted to (0.0 <= VALUE <= 1.0)

If parameter VALUE is less than 0.0 or greater than 1.0:

- The linear scaling operation can produce OUT values that are less than the parameter MIN value or above the parameter MAX value for OUT values that fit within the value range of the OUT data type. SCALE_X execution sets ENO = TRUE for these cases.
- It is possible to generate scaled numbers that are not within the range of the OUT data type. For these cases, the parameter OUT value is set to an intermediate value equal to the least-significant portion of the scaled real number prior to final conversion to the OUT data type. SCALE_X execution sets ENO = FALSE in this case.

NORM_X parameter VALUE should be restricted to (MIN <= VALUE <= MAX)

If parameter VALUE is less than MIN or greater than MAX, the linear scaling operation can produce normalized OUT values that are less than 0.0 or greater than 1.0. NORM_X execution sets ENO = TRUE in this case.

Table 7- 90 ENO status

ENO	Condition	Result OUT
1	No error	Valid result
0	Result exceeds valid range for the OUT data type	Intermediate result: The least-significant portion of a real number prior to final conversion to the OUT data type.
0	Parameters MAX <= MIN	SCALE_X: The least-significant portion of the Real number VALUE to fill up the OUT size. NORM_X: VALUE in VALUE data type extended to fill a double word size.
0	Parameter VALUE = +/- INF or +/- NaN	VALUE is written to OUT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

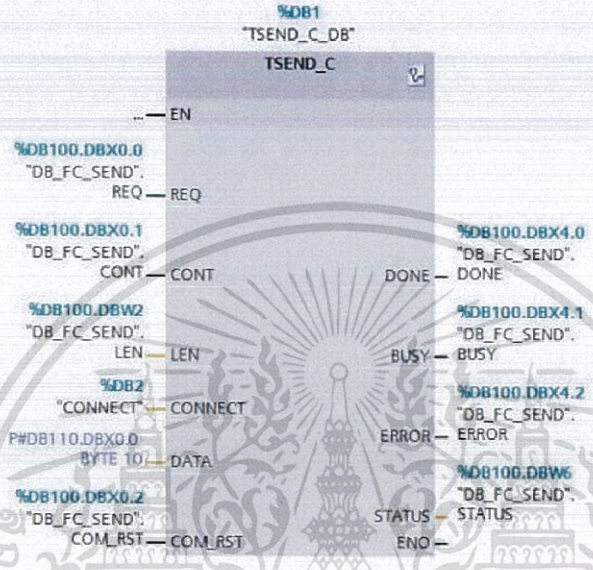
ก.2 บล็อก TSEND_C

2.3 FC100 "FC_SEND"

The FC100 "FC_SEND" function calls the TSEND_C instruction internally to establish a communication connection over ISO-on-TCP or UDP and send data through the communication connection.

The following figure shows the call of the TSEND_C instruction.

Figure 2-3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 Inputs and Outputs of the TSEND_C Instruction

Inputs

The table below gives an overview of

- the inputs of the TSEND_C instruction and
- the variables assigned to the inputs.

Table 2-1

Input	Data type	Variable	Description
REQ	BOOL	DB100.DBX0.0 "DB_FC_SEND".REQ	Starts the send job at a rising edge.
CONT	BOOL	DB100.DBX0.1 "DB_FC_SEND".CONT	Controls the communication connection: 0: Disconnects the communication connection. 1: Establishes and maintains the communication connection. If CONT=1, the TSEND_C instruction configures and establishes a communication connection. Once the connection has been configured and established, it is maintained and monitored automatically by the S7-1200 CPU.
LEN	UINT	DB100.DBW2 "DB_FC_SEND".LEN	Maximum number of bytes that can be sent with the job. **Note** If you use purely symbolic values at the DATA parameter, the LEN parameter must have the value 0.
CONNECT	TCON_Param	DB2 "CONNECT"	Pointer to the connection description used to establish the communication connection. Detailed information about the configuration of the connection description at the CONNECT parameter is available in section 2.5.
DATA	Variant	DB110.DBX0.0 BYTE 10	Pointer to the send area that contains the address and length of the data to be sent.
COM_RST	BOOL	DB100.DBX0.2 "DB_FC_SEND".COM_RST	Restarts the instruction: 0: Irrelevant. 1: Complete restart of the instruction which clears down the existing communication connection and establishes a new communication connection.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Outputs

The table below gives an overview of

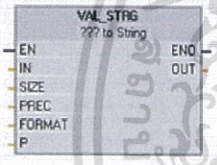
- the outputs of the TSEND_C instruction and
- the variables assigned to the outputs.

Table 2-2

Output	Data type	Variable	Description
DONE	BOOL	DB100.DBX4.0 "DB_FC_SEND".DONE	Status parameter with the values below: 0: Job not yet started or still being executed. 1: Job executed error-free.
BUSY	BOOL	DB100.DBX4.1 "DB_FC_SEND".BUSY	Status parameter with the values below: 0: Job not yet started or already terminated. 1: Job has not yet terminated. A new job cannot be started.
ERROR	BOOL	DB100.DBX4.2 "DB_FC_SEND".ERROR	Status parameter with the values below: 0: No error 1: Error occurred.
STATUS	WORD	DB100.DBW6 "DB_FC_SEND".STATUS	Status of the instruction

ก.3 บล็อก VAL_STRG

Table 8- 28 Value-to-string operation

LAD / FBD	SCL	Description
	"VAL_STRG"(in:=_variant_in, size:=_usint_in, prec:=_usint_in, format:=_word_in, p:=uint_in, out=>_string_out);	Converts an integer, unsigned integer, or floating point value to the corresponding character string representation.

1 For LAD / FBD: Click the "???" and select the data type from the drop-down list.

Table 8- 29 Data types for the VAL_STRG instruction

Parameter and type		Data type	Description
IN	IN	SInt, Int, DInt, USInt, UInt, UDInt, Real, LReal	Value to convert
SIZE	IN	USInt	Number of characters to be written to the OUT string
PREC	IN	USInt	The precision or size of the fractional portion. This does not include the decimal point.
FORMAT	IN	Word	Output format options
P	IN	UInt, Byte, USInt	IN: Index to the first OUT string character to be replaced (first character = 1)
OUT	OUT	String	The converted string

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The value represented by parameter IN is converted to a string referenced by parameter OUT. The parameter OUT must be a valid string before the conversion is executed.

The converted string will replace characters in the OUT string starting at character offset count P to the number of characters specified by parameter SIZE. The number of characters in SIZE must fit within the OUT string length, counting from character position P. This instruction is useful for embedding number characters into a text string. For example, you can put the numbers "120" into the string "Pump pressure = 120 psi".

Parameter PREC specifies the precision or number of digits for the fractional part of the string. If the parameter IN value is an integer, then PREC specifies the location of the decimal point. For example, if the data value is 123 and PREC = 1, then the result is "12.3". The maximum supported precision for the Real data type is 7 digits.

If parameter P is greater than the current size of the OUT string, then spaces are added, up to position P, and the result is appended to the end of the string. The conversion ends if the maximum OUT string length is reached.

The FORMAT parameter for the VAL_STRG instruction is defined below. The unused bit positions must be set to zero.

Table 8- 30 Format of the VAL_STRG instruction

Bit 16							Bit 8	Bit 7							Bit 0
0	0	0	0	0	0	0	0	0	0	0	0	0	s	f	r

s = Number sign character
1= use sign character "+" and "-"
0 = use sign character "-" only

f = Notation format
1= Exponential notation
0 = Fixed point notation

r = Decimal point format
1 = "," (comma character)
0 = "." (period character)

Table 8- 31 Values of the FORMAT parameter

FORMAT (WORD)	Number sign character	Notation format	Decimal point representation
W#16#0000	"-" only	Fixed point	"."
W#16#0001			","
W#16#0002		Exponential	"."
W#16#0003			","
W#16#0004	"+" and "-"	Fixed Point	"."
W#16#0005			","
W#16#0006		Exponential	"."
W#16#0007			","
W#16#0008 to W#16#FFFF	Illegal values		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.4 บล็อก Strg_TO_Chars

8.2.3.2 String-to-characters and characters-to-string conversions

Chars_TO_Strg copies an array of ASCII character bytes into a character string.

Strg_TO_Chars copies an ASCII character string into an array of character bytes.

Note

Only the zero based array types (Array [0..n] of Char) or (Array [0..n] of Byte) are allowed as the input parameter Chars for the Chars_TO_Strg instruction, or as the IN_OUT parameter Chars for the Strg_TO_Chars instruction.

Table 8- 37 Chars_TO_Strg instruction

LAD / FBD	SCL	Description

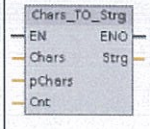
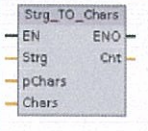
	`Chars_TO_Strg(`
	`  Chars:=_variant_in_,`
	`  pChars:=_dint_in_,`
	`  Cnt:=_uint_in_,`
	`  Strg=>_string_out_);`
		All or part of an array of characters is copied to a string.
		The output string must be declared before Chars_TO_Strg is executed. The string is then overwritten by the Chars_TO_Strg operation.
		Strings of all supported maximum lengths (1..254) may be used.
		The string maximum length value is not changed by Chars_TO_Strg operation. Copying from array to string stops when the maximum string length is reached.
		A nul character '$00' or 16#00 value in the character array works as a delimiter and ends copying of characters into the string.

Table 8- 38 Data types for the parameters (Chars_TO_Strg)

Parameter and type		Data type	Description
Chars	IN	Variant	The Chars parameter is a pointer to zero based array [0..n] of characters to be converted into a string. The array can be declared in a DB or as local variables in the block interface. Example: "DB1".MyArray points to MyArray [0..10] of Char element values in DB1.
pChars	IN	Dint	Element number for the first character in the array to copy. Array element [0] is the default value.
Cnt	IN	UInt	Count of characters to copy: 0 means all
Strg	OUT	String	Target string

Table 8- 39 Strg_TO_Chars instruction

LAD / FBD	SCL	Description

	`Strg_TO_Chars(`
	`  Strg:=_string_in_,`
	`  pChars:=_dint_in_,`
	`  Cnt=>_uint_out_,`
	`  Chars:=_variant_inout_);`
		The complete input string Strg is copied to an array of characters at IN_OUT parameter Chars.
		The operation overwrites bytes starting at array element number specified by the pChars parameter.
		Strings of all supported max lengths (1..254) may be used.
		An end delimiter is not written; this is your responsibility. To set an end delimiter just after the last written array character, use the next array element number [pChars+Cnt].

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 8- 40 Data types for the parameters (Strg_TO_Chars)

Parameter and type		Data type	Description
Strg	IN	String	Source string
pChars	IN	DInt	Array element number for the first string character written to the target array
Chars	IN_OUT	Variant	The Chars parameter is a pointer to zero based array [0..n] of characters copied from the input string. The array can be declared in a DB or as local variables in the block interface. Example: "DB1".MyArray points to MyArray [0..10] of Char element values in DB1.
Cnt	OUT	UInt	Count of characters copied

Table 8- 41 ENO status

ENO	Description
1	No error
0	Chars_TO_Strg: Attempt to copy more character bytes to the output string than allowed by the maximum length byte in the string declaration

ENO	Description
0	Chars_TO_Strg: The nul character (16#00) value was found in the input character byte array.
0	Strg_TO_Chars: Attempt to copy more character bytes to the output array than are allowed by the element number limit

ก.5 บล็อก PID_Compact

3.3.2 PID_Compact [FB1130]

STEP 7 V13 provides the "PID_Compact" version 2.2 technology object with the installation.
This function block was especially developed for controlling actuators acting proportionally.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 3-5

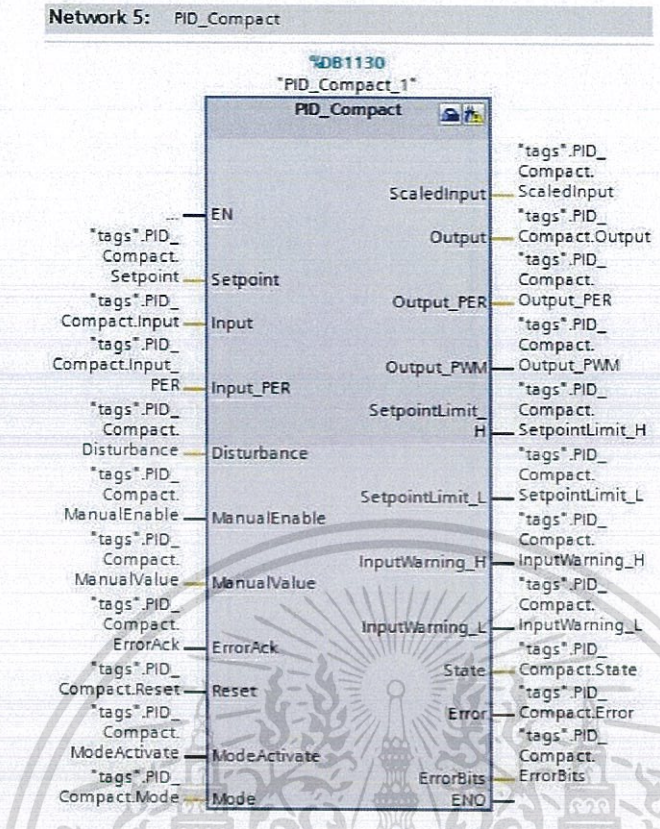


Table 3-2

	Name	Data type	Description
Input	Setpoint	Real	Setpoint input
	Input	Real	Current actual value in REAL
	Input_PER	Int	Current actual value from I/O
	Disturbance	Real	Disturbance connection
	ManualEnable	Bool	Manual value is activated for overwriting the manipulated value output
	ManualValue	Real	Manual value
	ErrorAck	Bool	Clearing the error message
	Reset	Bool	Reset, controller restart
	ModeActivate	Bool	Enable mode
Output	ScaledInput	Real	Scaled I/O actual value
	Output	Real	Output value in REAL
	Output_PER	Int	Output value to I/O
	Output_PWM	Bool	Pulse-width modulated output value

	Name	Data type	Description
	SetpointLimit_H	Bool	Setpoint is restricted to the upper limit
	SetpointLimit_L	Bool	Setpoint is restricted to the lower limit
	InputWarning_H	Bool	Actual value exceeded the upper warning limit
	InputWarning_L	Bool	Actual value fell short of the lower warning limit
	State	Int	Display of controller state (0=inactive, 1=SUT, 2=TIR, 3=automatic, 4=manual)
	Error	Bool	Error flag
	ErrorBits	DWord	Error message
InOut	Mode	Int	Mode selection

The "PID_Compact" controller is called in the "Cyclic interrupt" (OB200).
You will find the instance data block DB1130 for "PID_Compact" in the "Technology objects" folder:
It can be opened via right-click -> "Open DB editor".
Apart from the inputs and outputs, the application also accesses the static tags of "PID_Compact_1".

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

ข้อมูลคุณสมบัติของ PLC Siemens ตระกูล S7-1200

ข.1 คุณสมบัติของ PLC Siemens ตระกูล S7-1200

ตารางที่ ข.1 เปรียบเทียบคุณสมบัติของแต่ละรุ่นในตระกูล S7-1200

Feature		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C	CPU 1217C
Physical size (mm)		90 x 100 x 75		110 x 100 x 75	130 x 100 x 75	150 x 100 x 75
User memory	Work	50 Kbytes	75 Kbytes	100 Kbytes	125 Kbytes	150 Kbytes
	Load	1 Mbyte		4 Mbytes		
	Retentive	10 Kbytes				
Local on-board I/O	Digital	6 inputs/ 4 outputs	8 inputs/ 6 outputs	14 inputs/10 outputs		
	Analog	2 inputs			2 inputs/2outputs	
Process image size	Input(I)	1024 bytes				
	Output(Q)	1024 bytes				
Bit memory (M)		4096 bytes			8192 bytes	
Signal module (SM) expansion		None		2	8	
Signal board (SB), Battery board (BB), or communication board (CB)		1				
Communication module (CM) (left-side expansion)		3				
PROFINET Ethernet communication port		1			2	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.2 PLC Siemens Model S7-1214C AC/DC/Relay (6ES7 214-1BG31-0XB0)

ตารางที่ ข.2 คุณสมบัติของ PLC Siemens model S7-1214C AC/DC/Relay (6ES7 214-1BG31-0XB0)

General		
Technical data		**CPU 1214C AC/DC/Relay**
Order number		6ES7 214-1BG31-0XB0
Dimensions W x H x D (mm)		110 x 100 x 75
Shipping weight		475 grams
Power dissipation		14 W
Current available (SM and CM bus)		1600 mA max. (5 VDC)
Current available (24 VDC)		400 mA max. (sensor power)
Digital input current consumption (24 VDC)		4 mA/input used
CPU features		
Technical data		**Description**
User memory[1]	Work	75 Kbytes
	Load	4 Mbytes internal, expandable up to SD card size
	Retentive	10 Kbytes
On-board digital I/O		14 inputs/10 outputs
On-board analog I/O		2 inputs
Bit memory (M)		8192 bytes
Temporary (local) memory		• 16 Kbytes for startup and program cycle (including associated FBs and FCs) • 4 Kbytes for standard interrupt events including FBs and FCs • 4 Kbytes for error interrupt events including FBs and FCs
Pulse outputs[2]		4
Pulse catch inputs		14
Time delay / cyclic interrupts		4 total with 1 ms resolution

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข.2 (ต่อ) คุณสมบัติของ PLC Siemens model S7-1214C AC/DC/Relay (6ES7 214-1BG31-0XB0)

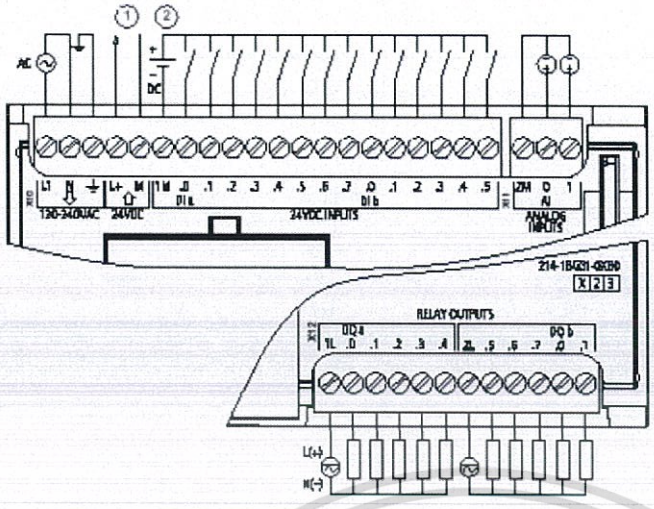
Communication functions	
Functional	
S7 communication	• As server • As client
Open IE communication	• TCP/IP • ISO-on-TCP (RFC1006) • UDP
Web server	User-defined websites
Configuration	
Engineering with	
Programming package	STEP 7 V11 SP2 or higher
Programming	
Programming Language	• LAD • FBD • SCL

[1] The size of user program, data, and configuration is limited by the available load memory and work memory in the CPU. There is no specific limit to the number of OB, FC, FB and DB blocks supported or to the size of particular block; the only limit is due to overall memory size.

[2] For CPU models with relay output, you must install a digital signal board (SB) to use the pulse outputs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.3 CPU 1214C Wiring Diagrams



① 24 VDC Sensor Power Out

For additional noise immunity, connect "M" to chassis ground even if not using sensor supply.

② For sinking inputs, connect "-" to "M" (shown).

For sourcing inputs, connect "+" to "M".

Note: X11 connectors must be gold. See Appendix C, Spare Parts for order number.

รูปที่ ข.1 แผนผังในการเดินสายไฟของ PLC SIEMENS รุ่น S7-1214C

ตารางที่ ข.3 ตารางขาพินของ PLC S7-1214C

Pin	X10	X11 (gold)	X12
1	L1 / 120-240 VAC	2M	1L
2	N / 120-240 VAC	AI 0	DQ a.0
3	Functional Earth	AI 1	DQ a.1
4	L+ / 24 VDC Sensor Out	--	DQ a.2
5	M / 24 VDC Sensor Out	--	DQ a.3
6	1M	--	DQ a.4
7	DI a.0	--	2L
8	DI a.1	--	DQ a.5
9	DI a.2	--	DQ a.6
10	DI a.3	--	DQ a.7
11	DI a.4	--	DQ b.0
12	DI a.5	--	DQ b.1
13	DI a.6	--	--
14	DI a.7	--	--
15 - 20	DI b.0 – DI b.5	--	--

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข.4 Analog Inputs

ตารางที่ ข.4 ข้อมูลขาแอนะล็อกของ PLC Siemens S7-1214C

Technical data	Description
Number of inputs	2
Type	Voltage (single-ended)
Full-scale range	0 to 10 V
Full-scale range (data word)	0 to 27,648
Overshoot range	10.001 to 11.759 V
Overshoot range (data word)	27,649 to 32,511
Overflow range	11.760 to 11.852 V
Overflow range (data word)	32,512 to 32,767
Resolution	10 bits
Maximum withstand voltage	35 VDC
Accuracy (25°C / -20 to 60°C)	3.0% / 3.5% of full-scale
Cable length (meters)	100m, shielded twisted pair

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

โปสเตอร์

IC-047

Wireless Level Control System

Ms.Waratchaya Boonchuoi, Ms.Pataraporn Singhaphet, Ms.Thanisa Musikasopon, Ms.Narisara Leatjantuk, Ms.Chanida Sriprab, Ms.Jitraphorn Chusaeng, Mr.Chanathip Chokkanapitak, Mr.Worrathun Thamsuntree, Ms.Parnravee Pornthisarn, Ms.Phattrawadee Sritas, Ms.Pantip Panyasit, Ms.Ramrada Khunnaphap, Ms.Siwan Sasipaisith and Mr.Sarat Pongprasert

Advisor: Prof. Dr. Vanchai Riewruja, Asst.Prof. Thepjit Cheypoca, Asst.Prof. Dr. Wandee Petchmaneelumka and Dr. Sirichai Thammarakwattana

Department of Instrumentation and Control Engineering, Faculty of Engineering

E-mail: 55011089@kmitl.ac.th, 55011046@kmitl.ac.th

Abstract

This project presents a design and implementation of a small plant for study and simulation of closed-loop control systems in the industrial work. Level control system is used for study in this project. Moreover, controlling with wireless is employed in the system. The project divides into two parts; wireless plant networks and controlling method.

First, the plant is built as a water level controlling plant. Transducers used in the model plant consist of two sensors and one motor functioned as flow and level transmitters and actuator. Arduino, a widely used microcontroller in engineering works, is employed to provide establish a communication between transducers and controller over Wi-Fi. As mention earlier, control system based on wireless connectivity can reduce signaling cable requirement, which is helpful in a big plant using wired system.

Second, PLC is used as a controller to control the level of water in the tank to the point where users need. There are various type of controller, however PID controller is selected as the plant control method. A PID controller is a control loop feedback mechanism commonly used in industrial control systems. A PID controller continuously calculates an error value as the difference between a desired setpoint and a measured process variable then attempts to minimize the error over time by adjustment of a control variable. TIA Portal V13 is used to program a control logic and PID controller to make the water reach to the setpoint of the wireless level control system spontaneously.

Introduction

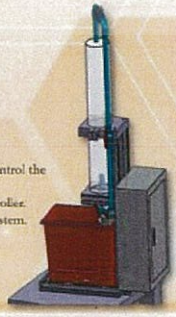
At present, most industrial factory prefer using PLC to control plant from remote location reducing the number of wiring used for connecting relays. Nevertheless, it still need some wiring to connect between some elements, such as sensors and actuators. SIEMENS has developed a client module to eliminate the problem by connect via wireless network instead of regular wiring. Due to the fact that client module has high price, so we use Wi-Fi module ESP8266-01 to control water level in the wireless level control system as it has lower cost. Wi-Fi module is used as a medium of a communication between PLC and components. The model plant is designed using SolidWorks and controlled with PLC SIEMENS model S7-1200 via Wi-Fi module ESP8266-01.

Objectives

1. Study of the principle of microcontroller and the communication via Wi-Fi module.
2. Study of the usage of Arduino and TIA Portal v13.
3. Design a wireless control system.

Methodology

1. Build a plant for study.
2. Study about sensor and actuator.
3. Design a circuit board for sensor and motor.
4. Create a library for Wi-Fi ESP8266-01.
5. Program the Arduino boards to send/ receive data and control the motor.
6. Make ladder program to control the plant using PID controller.
7. Make an interface for users for monitor and control the system.
8. Assemble all component of the plant together.
9. Tune PID controller and control the system via Wi-Fi.

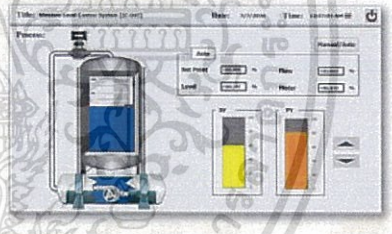


Results

The experimental result shows that the devices in the wireless system can communicate with each other over Wi-Fi by sending the data from pressure sensor to PLC then using TIA Portal to tune PID parameters while the setpoint of the system is 60%. The PID parameters are set as follow.

Proportional gain	=	3.506227E-1
Integral action time	=	14.78867 s
Derivative action time	=	3.986802 s
Derivative delay coefficient	=	0.1
Proportional action weighting	=	5.480434E-1
Derivative action weighting	=	0.0
Sampling time of PID algorithm	=	3.999916E-1 s

Afterward, PLC will send a data to control motor speed and pump the water and control it to the setpoint level. When the setpoint is changed to a difference values, the system still able to keep the stability at the request values.



Conclusion

We simulated an inexpensive water level control system to replace the commonly used in the industrial one and be able to control it to reach the setpoint. By using TIA Portal to program a PLC and using Arduino to connect PLC with the devices over Wi-Fi module, the system is stable and able to be controlled as user desire. Which can be applied to a bigger plant in a control field of work.

References

[1] Deitel, Harvey, M. and Deitel, Paul. (1997). C++ How to Program. 2nd Edition. Hall.

[2] MaysPrentice arm43. "Protocol." [Online]. Available : https://sites.google.com/site/maysarm43/phl-ngan-khxng-chan

[3] Electrocomponents. "คุณสมบัติของ PLC Siemens model S7-1214C AC/DC/Relay (6ES7 214-1BG31- 0XB0)." [Online]. Available : http://docseurope.electrocomponents.com/webdocs/1129/0900766b81129785.pdf

[4] Paratrasnet. "คุณสมบัติของ PLC Siemens model S7-1214C AC/DC/Relay (6ES7 214-1BG31- 0XB0)." [Online]. Available : http://www.paratrasnet.ro/pdf/automatizari-industriale/S7-1200.pdf

[5] Sahkonumerot. "ค่าของ analog datasheet." [Online]. Available : https://www.sahkonumerot.fi/2702074/doc/technicalinfodoc/

[6] Arduino.cc "ตัวอย่าง code และการต่อวงจร Flow sensor." [Online]. Available : http://forum.arduino.cc/index.php?topic=8548.0.html

[7] Learninginventions. "Hall Effect Sensor." [Online]. Available : http://mcu8e.learninginventions.org/?page_id=758

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้