



## รายงานสหกิจศึกษาฉบับสมบูรณ์

การปรับปรุงกระบวนการอบแห้งแบบพ่นฝอยผ่านระบบควบคุมอัตโนมัติ  
The Enhancement of Spray Drying Process  
via Automated Control System

นางสาว มณฑกานติ แพงวิเศษ

ภาควิชาวิศวกรรมการวัดและควบคุม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2560



## รายงานสหกิจศึกษาฉบับสมบูรณ์

การปรับปรุงกระบวนการอบแห้งแบบพ่นฝอยผ่านระบบควบคุมอัตโนมัติ  
The Enhancement of Spray Drying Process  
via Automated Control System

นางสาว มณฑกานติ แพงวิเศษ

ภาควิชาวิศวกรรมการผลิตและควบคุม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2560

ชื่อโครงการสหกิจศึกษา การปรับปรุงกระบวนการอบแห้งแบบพ่นฝอยผ่านระบบควบคุมอัตโนมัติ

ชื่อ-สกุลนักศึกษา นางสาวนทกานติ แพงวิเศษ

คณะ วิศวกรรมศาสตร์

ภาควิชา วิศวกรรมการวัดและควบคุม

ชื่อ-สกุล อาจารย์นิเทศ รศ.ทรงชัย วีระทวีมาศ

ชื่อ-สกุล ผู้นิเทศงาน คุณชาติรี อุดมมณีรัตน์กิจ

ชื่อสถานประกอบการ บริษัท อินส์ไปร์เทค จำกัด

## บทคัดย่อ

รายงานสหกิจศึกษานี้นำเสนอการปรับปรุงกระบวนการอบแห้งแบบพ่นฝอยของอุตสาหกรรม การผลิตผงสี เพื่อเพิ่มประสิทธิภาพกระบวนการผลิตและเพิ่มคุณภาพของผลิตภัณฑ์ ให้สามารถลด ปริมาณของเสียที่เกิดจากการผลิตที่ไม่มีคุณภาพ และลดการใช้พลังงานสำหรับการกำจัดของเสีย โดย ศึกษาในส่วนของ การออกแบบระบบควบคุมกระบวนการด้วยตัวควบคุมเชิงตรรกะที่สามารถโปรแกรมได้ (PLC : Programmable Logic Controller) การออกแบบกราฟิกบนอุปกรณ์ HMI (Human Machine Interface) ให้ผู้ใช้งานสามารถควบคุมการทำงานของอุปกรณ์ต่างๆเช่น วาล์ว มอเตอร์ เป็นต้น รวมถึง สามารถปรับตั้งค่าพารามิเตอร์การควบคุมด้วยตัวควบคุมพีไอดี (PID) ได้ผ่านระบบสกาตา (SCADA: Supervisory Control And Data Acquisition)

คำสำคัญ : PLC, HMI, SCADA, PID

**Co-operative Research Title:** The Enhancement of Spray Drying Process via Automated Control System

**Student Intern Name:** Ms. Monthakanti Phangwiset

**Faculty:** Engineering

**Department:** Instrumentation and Control Engineering

**Advisor Name:** Assoc. Songchai Weerathaweemas

**Mentor Name:** Mr. Chatree Udommaneeratkiij

**Company:** Inspire Tech Co.,Ltd.

## ABSTRACT

This research in cooperative education presents the modification of spray drying process in pigments manufacturing industry, to increase of efficient dryer operation and save the energy required to reduce elimination of waste in the quality production. The research consists of the design of process control via PLC (Programmable Logic Controller), graphic design of HMI device for operator use to control devices such as valve and motor, including parameter setting for PID control via SCADA system (Supervisory Control And Data Acquisition).

**keywords:** PLC, HMI, SCADA, PID

## กิตติกรรมประกาศ

รายงานสหกิจศึกษาฉบับนี้เกิดขึ้นได้เนื่องจากได้รับความอนุเคราะห์จาก บริษัท อินส์ปอร์ต จำกัด ให้ไปปฏิบัติโครงการสหกิจศึกษาตลอดระยะเวลาหนึ่งภาคการศึกษา ซึ่งทำให้เกิดการเรียนรู้และสร้างประสบการณ์จากการปฏิบัติงานจริง มีบุคคลมากมายที่ต้องขอขอบคุณโดยเฉพาะคุณชาตรี อุดมมณีรัตน์กิจ ผู้นิเทศงาน รวมถึงบุคลากรทุกท่านที่มอบความรู้ แนวคิดและประสบการณ์ที่ล้ำค่าอีกทั้งยังให้การดูแลอย่างดีตลอดมา

ทั้งนี้ขอขอบคุณ รศ.ทรงชัย วีระหวิมาศ อาจารย์นิเทศ ที่ให้การดูแลขณะปฏิบัติสหกิจ และให้คำแนะนำในการจัดทำรายงานสหกิจศึกษาฉบับนี้จนสมบูรณ์ รวมถึงขอขอบคุณคณาจารย์ภาควิชาวิศวกรรม การวัดคุมทุกท่าน ที่มอบความรู้และการอบรมที่เป็นประโยชน์แก่ข้าพเจ้า

นางสาวมนทกานติ แพ่งวิเศษ

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VIII
สารบัญภาพ.....	IX
บทที่ 1 บทนำ.....	1
1.1 ที่มาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบเขตการศึกษา.....	2
1.4 ขั้นตอนการศึกษา.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 เครื่องอบแห้งแบบพ่นฝอย.....	3
2.1.1 กระบวนการอบแห้งแบบพ่นฝอย.....	3
2.1.2 ขั้นตอนการอบแห้งแบบพ่นฝอยวงรอบเปิด.....	4
2.1.2.1 การทำละออง (Atomization).....	4
2.1.2.2 การอบแห้งที่ห้องอบแห้ง (Drying Chamber).....	6
2.1.2.3 การระเหยเป็นไอ (Evaporation).....	6
2.1.2.4 การแยกอนุภาคผงออกจากอากาศ.....	6
2.1.3 ระบบควบคุมในกระบวนการอบแห้ง.....	8
2.1.4 การอบแห้งสำหรับอุตสาหกรรมเม็ดสี.....	8
2.2 ความรู้พื้นฐานเกี่ยวกับ PLC.....	9
2.2.1 โครงสร้างของ PLC.....	10
2.2.2 ส่วนประกอบของ PLC.....	11
2.2.2.1 หน่วยประมวลผลกลาง.....	11
2.2.2.2 อินพุต/เอาต์พุต.....	12
2.2.2.3 อุปกรณ์การโปรแกรม.....	12
2.2.3 ความสามารถของ PLC.....	13
2.2.3.1 งานที่ทำตามลำดับก่อนหลัง.....	13

## สารบัญ (ต่อ)

	หน้า
2.2.3.2 งานควบคุมสมัยใหม่.....	13
2.2.3.3 การควบคุมเกี่ยวกับงานอำนวยการ.....	13
2.2.4 การติดตั้ง PLC.....	14
2.2.4.1 ข้อควรพิจารณาก่อนติดตั้ง PLC.....	14
2.2.4.2 สภาพแวดล้อมหรือสถานที่ที่ไม่ควรติดตั้ง PLC.....	14
2.2.5 ตู้ควบคุมสำหรับ PLC.....	14
2.2.6 ภาษาที่ใช้ในการเขียนโปรแกรมให้กับ PLC.....	14
2.2.7 คำสั่งพื้นฐานของแลตเตอร์ไดอะแกรม.....	17
2.2.7.1 หน้าสัมผัสปกติเปิด.....	17
2.2.7.2 หน้าสัมผัสปกติปิด.....	17
2.2.7.3 ขดลวดเอาต์พุต.....	18
2.2.7.4 ย้าย.....	18
2.3 ความรู้พื้นฐานเกี่ยวกับระบบสกาตา.....	19
2.3.1 โครงสร้างของสกาตา.....	19
2.3.1.1 โครงสร้างฮาร์ดแวร์.....	19
2.3.1.2 โครงสร้างด้านซอฟต์แวร์.....	20
2.3.1.3 โครงสร้างด้านการสื่อสาร.....	21
2.3.1.4 โครงสร้างการติดต่อ.....	21
2.3.1.5 โครงสร้างความสามารถในการขยายระบบ.....	22
2.3.1.6 โครงสร้างการสำรองระบบ.....	22
2.3.2 หน้าที่การทำงาน.....	22
2.3.2.1 ระบบแสดงผลแบบ MMI.....	22
2.3.2.2 ระบบแสดงกราฟสัญญาณแบบต่อเนื่อง.....	23
2.3.2.3 ระบบแจ้งเตือน.....	23
2.3.2.4 การทำงานแบบอัตโนมัติ.....	23
2.3.3 การสร้างและพัฒนา.....	24
2.3.3.1 การกำหนดคอนฟิกูเรชัน.....	24
2.3.1.2 เครื่องมือในการพัฒนา.....	24
2.4 ระบบเครือข่ายอีเทอร์เน็ต .....	24
2.5 การสื่อสารระหว่างอุปกรณ์ HMI และ SIMATIC S7.....	25
2.6 ซอฟต์แวร์ที่เกี่ยวข้อง.....	26
2.6.1 SIMATIC STEP 7.....	26

## สารบัญ (ต่อ)

	หน้า
2.6.2 WinCC flexible 2008.....	26
<b>บทที่ 3 การออกแบบโปรแกรมควบคุม.....</b>	<b>29</b>
3.1 ออกแบบโปรแกรมควบคุมด้วยพีแอลซี.....	29
3.1.1 ศึกษากระบวนการ.....	29
3.1.2 ออกแบบโปรแกรมควบคุม.....	32
3.1.2.1 สร้างตัวแปรในซอฟต์แวร์ SIMATIC S7.....	32
3.1.2.2 เขียนโปรแกรมควบคุม.....	33
3.1.2.3 การใช้ฟังก์ชันบล็อก 41.....	38
3.2 ออกแบบการแสดงผล.....	39
3.2.1 การสร้างแท็ก (Tag) บนอุปกรณ์ HMI.....	39
3.2.2 การสร้างวัตถุ (Object) .....	40
3.2.3 การตั้งค่าการเข้าถึงวัตถุ.....	43
3.2.4 การสร้าง Faceplate.....	44
3.2.5 สัญลักษณ์.....	47
3.3 การแสดงข้อความเตือน.....	48
3.3.1 การสร้างแท็กของข้อความเตือน.....	48
3.3.2 การเขียนโปรแกรมให้เกิดสัญญาณเตือน.....	49
<b>บทที่ 4 ผลการดำเนินงาน.....</b>	<b>50</b>
4.1 ทดสอบการทำงานและการแสดงผลของอุปกรณ์ต่างๆ.....	50
4.1.1 ทดสอบการแสดงผลสถานะของปุ่มกด.....	50
4.1.2 ทดสอบการแสดงผลสถานะของวาล์วเปิดปิด.....	51
4.1.3 ทดสอบการแสดงผลสถานะของวาล์วควบคุม.....	51
4.1.4 ทดสอบการแสดงผลสถานะของมอเตอร์ควบคุม.....	51
4.1.5 ทดสอบการแสดงผลสถานะของเซนเซอร์.....	52
4.1.6 ทดสอบการแสดงผลสถานะของทรานสมิตเตอร์.....	52
4.2 ทดสอบการแจ้งเตือนเมื่อเกิดข้อผิดพลาดกับอุปกรณ์.....	52
4.3 ทดสอบฟังก์ชันการกดปุ่มเรียกหน้าต่าง Faceplate.....	54
4.4 ทดสอบการควบคุมด้วยตัวควบคุมพีไอดี.....	54
4.5 การแสดงผลหน้าจอ.....	56

## สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลและข้อเสนอแนะ.....	58
5.1 สรุปผลการดำเนินงาน.....	58
5.2 ปัญหาและแนวทางแก้ไข.....	58
5.2.1 ปัญหาที่พบ.....	58
5.2.2 แนวทางแก้ไข.....	58
5.3 ข้อเสนอแนะ.....	58
เอกสารอ้างอิง.....	59
ภาคผนวก.....	60
ภาคผนวก ก.....	60
ภาคผนวก ข.....	61
ประวัติผู้เขียน.....	63

## สารบัญตาราง

ตารางที่	หน้า
2.1 ข้อแตกต่างของการอบแห้งด้วยละออง การอบแห้งของเหลว ..... และการอบแห้งแบบแฟลช	3
2.2 โครงข่ายการสื่อสารระหว่างอุปกรณ์ HMI และ SIMATIC S7.....	25
3.1 ตัวอย่างรายการอนาล็อกอินพุต.....	31

## สารบัญภาพ

ภาพที่	หน้า
2.1 แสดงกระบวนการของการอบแห้งแบบพ่นฝอยวงรอบเปิด.....	5
2.2 อุปกรณ์ที่เกี่ยวข้องในกระบวนการอบแห้งแบบพ่นฝอยวงรอบเปิด.....	7
2.3 ส่วนประกอบของ PLC.....	10
2.4 I/O Scan และ Program Scan.....	11
2.5 Ladder Diagram.....	15
2.6 Function Block Diagram.....	15
2.7 Instruction list.....	16
2.8 Structure text.....	16
2.9 Sequential Function.....	16
2.10 หน้าสัมผัสสปกติเปิด.....	17
2.11 หน้าสัมผัสสปกติปิด.....	17
2.12 ขดลวดเอาท์พุท.....	18
2.13 คำสั่งย้าย.....	18
2.14 โครงสร้างแบบฮาร์ดแวร์ของระบบสกาตา.....	20
2.15 โครงสร้างด้านซอฟต์แวร์ของสกาตา.....	21
2.16 ซอฟต์แวร์ SIMATIC STEP7 .....	26
2.17 ซอฟต์แวร์ WinCC flexible 2008 .....	27
2.18 องค์ประกอบของ WinCC flexible .....	27
3.1 แผนภาพ P&ID ของกระบวนการอบแห้งแบบพ่นฝอย.....	29
3.2 ฝั่งงาน Auto Mode.....	30
3.3 ตารางสัญญาณเตือน (Alarm) และการเกี่ยวข้องกันของอุปกรณ์ (Interlock).....	32
3.4 Data block.....	32
3.5 ฝั่งงานคำสั่งการเปิดอุปกรณ์ .....	33
3.6 คำสั่งสร้าง Flag ไปใช้งาน .....	33
3.7 คำสั่งการเปิดปิดวาล์วควบคุม.....	34
3.8 คำสั่งให้มอเตอร์ทำงาน.....	35
3.9 คำสั่งใช้งาน PID ในโหมด Auto .....	35
3.10 การรับค่าอนาล็อกและเงื่อนไข.....	36
3.11 การสเกลค่าผ่าน FC106 .....	37
3.12 คำสั่งให้เกิดการแจ้งเตือน (Alarm) .....	38
3.13 FB 41 และ Instance Data Block .....	39
3.14 การสร้างแท็กสำหรับกราฟิก.....	40

## สารบัญภาพ (ต่อ)

ภาพที่	หน้า
3.15 การสร้างปุ่มกด.....	41
3.16 การตั้งค่าการเคลื่อนไหวให้กับวัตถุ .....	41
3.17 การตั้งค่าการปรากฏให้กับวัตถุ.....	42
3.18 การตั้งค่าเหตุการณ์ให้กับวัตถุ.....	42
3.19 การกำหนดกลุ่มการเข้าถึง.....	43
3.20 การตั้งค่าผู้ใช้และรหัสผ่าน.....	44
3.21 หน้า Log on.....	44
3.22 การสร้างหน้าต่าง Faceplate.....	45
3.23 การเรียกใช้ Faceplate.....	46
3.24 หน้าต่าง Faceplate.....	47
3.25 สีของสารและสัญลักษณ์ท่อ.....	47
3.26 ชื่อและสัญลักษณ์อุปกรณ์ที่ใช้บนกราฟิก.....	48
3.27 การสร้างแท็กของข้อความแจ้งเตือน.....	48
3.28 แผนภาพแลตเตอร์การเกิดสัญญาณเตือน.....	49
3.29 ข้อความแจ้งเตือนบนหน้ากราฟิก.....	49
4.1 การจำลองสัญญาณขาเข้าบน PLC simulator.....	50
4.2 การแสดงฟังก์ชันการทำงานของปุ่มกด.....	51
4.3 การแสดงสถานะของวาล์วเปิดปิด.....	51
4.4 การแสดงสถานะของวาล์วควบคุม.....	51
4.5 การแสดงผลมอเตอร์ควบคุม.....	52
4.6 ข้อความแจ้งเตือนเมื่อใส่ค่าที่เกินขอบเขต .....	52
4.7 การแสดงสถานะเซนเซอร์.....	52
4.8 การแสดงผลของทรานสมิตเตอร์.....	52
4.9 หน้าต่างแจ้งเตือนเมื่อเกิดข้อผิดพลาดกับอุปกรณ์.....	53
4.10 ประวัติการแจ้งเตือน.....	53
4.11 หน้าต่าง Faceplate ที่ถูกเรียกขึ้นมาจากปุ่มกด.....	54
4.12 การทำงานในโหมดควบคุมด้วยมือของการควบคุมแบบ Cascade.....	54
4.13 การทำงานในโหมดอัตโนมัติของการควบคุมแบบ Cascade.....	55
4.14 การควบคุมในโหมด Cascade.....	56
4.15 กราฟิกหน้ากระบวนกร.....	57

# บทที่ 1

## บทนำ

### 1.1 ที่มาและความสำคัญ

บริษัทอินสปิรเทค จำกัด เป็นผู้ให้บริการครบวงจรด้านระบบควบคุมอัตโนมัติเพื่อการผลิตและอุตสาหกรรม ในการการออกแบบระบบวัดและควบคุม ครอบคลุมผลิตภัณฑ์ทั้งหมดของวิศวกรรมไฟฟ้า จัดหาอุปกรณ์ ทดสอบระบบควบคุมอัตโนมัติ บริการหลังการขาย การซ่อมบำรุง หลักสูตรอบรมการใช้ งาน PLC สำหรับลูกค้า ปรับเปลี่ยนและแก้ไขระบบควบคุมตามความต้องการของลูกค้า ประสบความสำเร็จในด้านระบบควบคุมอัตโนมัติสำหรับการผลิตและอุตสาหกรรมทั้งในและต่างประเทศ โดยมีบุคลากรผู้เชี่ยวชาญทางด้าน วิศวกรรมควบคุมและเครื่องมือวัด (Control & Instrument Engineers) วิศวกรรมไฟฟ้า (Electrical Engineers) วิศวกรรมคอมพิวเตอร์ (Computer Engineers) ช่างเทคนิค (Technicians) และวิศวกรขาย (Sales Engineers)

เพื่อให้ผลิตภัณฑ์มีคุณภาพ กระบวนการผลิตในอุตสาหกรรมจำเป็นต้องมีการควบคุมปริมาณทาง ฟิสิกส์ทั้ง อุณหภูมิ ความดัน อัตราการไหล และระดับ ให้มีค่าเท่ากับหรือใกล้เคียงกับค่าที่ต้องการ ทั้งใน สภาพแวดล้อมปกติหรือในสภาพแวดล้อมที่มีการเปลี่ยนแปลงตลอดเวลา <sup>[3]</sup> จึงนิยมใช้ระบบควบคุมแบบ อัตโนมัติ ให้ทำหน้าที่คำนวณหาปริมาณการควบคุมที่เหมาะสมตามที่ผู้ใช้งานกำหนด ซึ่งจะทำให้การประมวลผล โดยใช้ตัวควบคุมเชิงตรรกะที่สามารถโปรแกรมได้ (PLC : Programmable Logic Controller)

สำหรับการปฏิบัติงานสหกิจศึกษาที่บริษัทอินสปิรเทค ทางบริษัทได้มอบหมายให้รับผิดชอบการ แก้ไขระบบควบคุมแบบอัตโนมัติของกระบวนการอบแห้งแบบพ่นฝอยสำหรับอุตสาหกรรมผลิตผงหมึกสี ของบริษัทแห่งหนึ่งที่ประกอบธุรกิจด้านผลิตภัณฑ์เคมีภัณฑ์ สีย้อมและเคมีสิ่งทอ ที่ต้องการปรับปรุงแก้ไข กระบวนการผลิตที่มีการควบคุมการทำงานด้วย PLC เพื่อเพิ่มประสิทธิภาพให้กระบวนการผลิตและเพิ่ม คุณภาพของผลิตภัณฑ์ผงสี เนื่องจากผลการทดสอบคุณภาพของผลิตภัณฑ์ที่ได้จากกระบวนการผลิตเดิม ไม่ตรงตามความต้องการ ผลิตภัณฑ์ที่ได้จากการผลิตถูกรรจและจัดเก็บทันทีโดยไม่มีการทำงานให้ผลิตภัณฑ์ เย็นลงก่อน เมื่อเก็บไว้นานความสามารถการติดสีจากการย้อมจะลดลง ต้องปรับปรุงกระบวนการผลิตโดย เพิ่มเครื่องมือวัดและอุปกรณ์ทางกล เพิ่มเติมคำสั่งการทำงานให้กับอุปกรณ์โดยใช้ PLC และสามารถ ควบคุมและสื่อสารกับผู้ใช้งานผ่านหน้าจอ HMI

กระบวนการอบแห้งแบบพ่นฝอยจะมีเครื่องจักรทำหน้าที่เป่าลมร้อนเข้าไปในระบบ เพื่อให้ลม ร้อนไปสัมผัสกับตัวน้ำหมึก ทำให้น้ำที่เป็นส่วนประกอบของน้ำหมึกระเหยออกไป เหลือแต่เพียงผงหมึกตก ลงสู่ด้านล่าง เพื่อบรรจุและจัดเก็บผงหมึกต่อไป โดยเครื่องเป่าลมร้อนจะมีกระบวนการควบคุมอุณหภูมิ ของลมร้อน เพื่อให้ลมร้อนมีอุณหภูมิที่พอเหมาะและเพียงพอที่จะทำให้น้ำหมึกระเหยได้หมด นอกจากนี้ ยังมีการควบคุมปริมาณน้ำหมึกที่ฉีดเข้าไปในระบบให้มีปริมาณที่เหมาะสม การควบคุมปริมาณต่างๆทั้ง อุณหภูมิของลมร้อนและปริมาณของน้ำหมึกที่ฉีดเข้าไปในระบบจะถูกควบคุมด้วยตัวควบคุม PID ซึ่ง สามารถตั้งค่าพารามิเตอร์ได้ผ่านหน้าจอ HMI (Human Machine Interface)

## 1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาและออกแบบโปรแกรมควบคุมโดยใช้ PLC ของกระบวนการอบแห้งแบบพ่นฝอย
2. เพื่อศึกษาและออกแบบกราฟิกแสดงผลการทำงาน ควบคุมการทำงาน และติดต่อสื่อสารกับผู้ใช้งาน
3. เพื่อเพิ่มประสิทธิภาพให้กระบวนการผลิตผงหมึกสีและเพิ่มคุณภาพของผลิตภัณฑ์

## 1.3 ขอบเขตการศึกษา

1. ศึกษาการออกแบบ PLC เพื่อนำไปควบคุมการทำงานของเครื่องจักร โดยใช้ซอฟต์แวร์ SIMATIC STEP7
2. ศึกษาการเขียนกราฟิก ของหน้าจอ HMI เพื่อแสดงผลการทำงานของโปรแกรมควบคุม โดยใช้ซอฟต์แวร์ WinCC Flexible 2008
3. ศึกษาทฤษฎีที่เกี่ยวข้องและหลักการทำงานของ PLC ยี่ห้อ Siemens

## 1.4 ขั้นตอนการศึกษา

1. ศึกษาการเขียน PLC ยี่ห้อ Siemens ที่ใช้งานในระบบ
2. ศึกษาการเขียนกราฟิก โดยซอฟต์แวร์ WinCC flexible 2008
3. ศึกษาโปรแกรมควบคุม PLC ของระบบเก่า
4. ศึกษาขั้นตอนการทำงานของระบบที่ต้องการเพิ่มในกระบวนการ
5. ออกแบบโปรแกรมควบคุม PLC ของระบบที่เพิ่มใหม่ในกระบวนการ
6. ออกแบบหน้าจอกราฟิกของระบบที่เพิ่มใหม่ในกระบวนการ โดยอ้างอิงตามแบบ P&ID
7. เขียนโปรแกรมส่วน Interlock, Alarm, Control Shutdown และ Emergency Shutdown
8. ทดสอบการทำงานร่วมกันระหว่างหน้าจอHMI และโปรแกรม PLC
9. นำระบบไปติดตั้ง ทดสอบการทำงานและแก้ไขปัญหา

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถออกแบบโปรแกรมควบคุมกระบวนการอบแห้งแบบพ่นฝอยให้ทำงานได้อย่างต่อเนื่อง
2. สามารถออกแบบกราฟิกให้ผู้ใช้งานสื่อสารกับระบบควบคุม โดยสังเกตและสั่งการระบบได้
3. เพิ่มคุณภาพผลิตภัณฑ์ที่ได้จากกระบวนการผลิต
4. เข้าใจหลักการเขียน PLC เพื่อนำไปควบคุมระบบให้ทำงานได้ตามต้องการ

## บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

### 2.1 เครื่องอบแห้งแบบพ่นฝอย (Spray dryer)

#### 2.1.1 กระบวนการอบแห้งแบบพ่นฝอย

กระบวนการอบแห้งแบบพ่นฝอย (Spray Drying) อยู่ในตระกูลของระบบกระบวนการผลิตอนุภาคแขวนลอย Suspended Particle Processing (SPP) ร่วมกับการอบแห้งของเหลว (Fluid Bed Drying) การอบแห้งแบบแฟลช (Flash Drying) การทำอนุภาคเป็นเม็ดแบบพ่นฝอย (Spray Granulation) การทำอนุภาคเป็นกลุ่มก้อนแบบพ่นฝอย (Spray Agglomeration) การทำปฏิกิริยาแบบพ่นฝอย (Spray Reaction) การให้ความเย็นแบบพ่นฝอย (Spray Cooling) การดูดซึมแบบพ่นฝอย (Spray Absorption) เนื่องจากการอบแห้งกระทำขณะอนุภาคตกค้างอยู่ในอากาศ

ตารางที่ 2.1 ข้อแตกต่างของการอบแห้งแบบพ่นฝอย การอบแห้งของเหลว และการอบแห้งแบบแฟลช

	การอบแห้งแบบพ่นฝอย (Spray Drying)	การอบแห้งของเหลว (Fluid Bed Drying)	การอบแห้งแบบแฟลช (Flash Drying)
สารละลาย	ของเหลว	ของแข็ง	ของแข็ง
เวลาที่ใช้ทำให้แห้ง(วินาที)	5-100	1-300	1-5
ขนาดอนุภาค (ไมโครเมตร)	10-500	10-3000	10-300

รายงานฉบับนี้ใช้การอบแห้งแบบพ่นฝอย (Spray Drying) เป็นการแปรสภาพสารจากสถานะของเหลวหรือสารละลายเป็นอนุภาคแห้ง โดยส่งผ่านให้เป็นละอองในสภาพอากาศร้อน เป็นวิธีที่ใช้ในการอบแห้งสารที่ไวต่อความร้อนหลายชนิด สามารถอบแห้งได้ทั้งสารละลาย สารแขวนลอย อิมัลชัน (Emulsion) หรือสารที่มีลักษณะเหนียว เพื่อให้ได้การกระจายตัวของอนุภาคอย่างสม่ำเสมอ บางผลิตภัณฑ์ในอุตสาหกรรมต้องใช้ตัวเร่งปฏิกิริยาการอบแห้ง โดยการใช้ลมร้อนเพื่ออบแห้งเป็นกระบวนการแบบวงรอบเปิด (Open loop) สามารถปล่อยอากาศออกสู่บรรยากาศได้ อย่างไรก็ตามถ้าของเหลวเป็นตัวทำละลายที่ไวไฟ เช่น เอทานอลหรือผลิตภัณฑ์ที่เป็นออกซิเจน จะใช้ในโตรเจนแทนอากาศร้อน ซึ่งตัวทำละลายที่ไวไฟมีความเสี่ยงต่อการติดไฟและการระเบิดมากกว่าเป็นกระบวนการแบบวงรอบปิด (Close loop) นอกจากนี้ยังมีกระบวนการอบแห้งแบบกึ่งปิด โดยรายงานฉบับนี้จะกล่าวถึงเฉพาะกระบวนการแบบวงรอบเปิด

การอบแห้งแบบพ่นฝอยเป็นกระบวนการอบแห้งที่เป็นเอกลักษณ์ เนื่องจากมีการเกี่ยวข้องทั้งการสร้างอนุภาคและการอบแห้ง สามารถควบคุมคุณลักษณะเฉพาะของผง (Powder) ได้ และสามารถรักษาคุณสมบัติของผงให้คงที่อย่างต่อเนื่องตลอดการทำงาน การออกแบบเครื่องอบแห้งแบบพ่นฝอยสามารถเลือกรูปแบบให้กับผลิตภัณฑ์ได้ทั้งอนุภาคผงละเอียด ผงหยาบ เป็นกลุ่มก้อน หรือผงเป็นเม็ด ขึ้นกับการออกแบบเครื่องอบแห้ง การกำหนดโดยผู้ใช้งาน และคุณสมบัติทางกายภาพและทางเคมีของอัตราการไหลของสารขาเข้า (Feed)

การอบแห้งแบบพ่นฝอยในหลายอุตสาหกรรมใช้เป็นขั้นตอนการทำให้ผลิตภัณฑ์แห้ง ซึ่งเครื่องอบแห้งแบบพ่นฝอยมีคุณสมบัติตรงตามข้อกำหนดของผลิตภัณฑ์แห้งที่ต้องการมากที่สุด โดยลำดับกระบวนการอบแห้งมีการวิจัยอย่างเข้มข้นและพัฒนาในช่วงไม่กี่ทศวรรษที่ผ่านมา ทำให้มีผลิตภัณฑ์ที่มีความหลากหลาย ขอบเขตของผลิตภัณฑ์ยังสามารถพัฒนาต่อไปได้อีก ทำให้ปัจจุบันการอบแห้งแบบพ่นฝอยเกี่ยวข้องกับผลิตภัณฑ์หลายสิ่งในชีวิตประจำวัน ตัวอย่างในผลิตภัณฑ์อาหาร เช่น นมผง ครีมเทียม กาแฟสำเร็จรูป โปรตีนสกัด และผลิตภัณฑ์ยา เช่น ยาผงสมุนไพร เป็นต้น

## 2.1.2 ขั้นตอนการอบแห้งแบบพ่นฝอยวงรอบเปิด

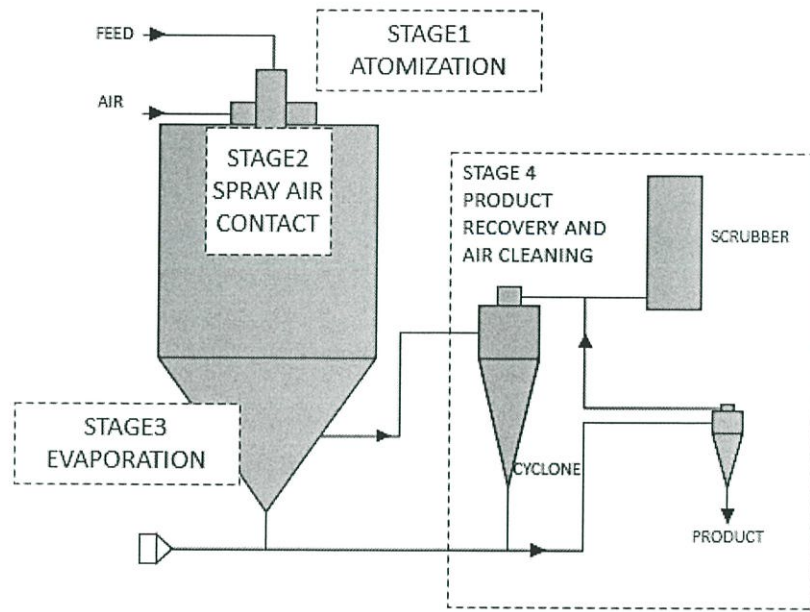
การอบแห้งแบบพ่นฝอยเกี่ยวข้องกับการทำให้เป็นละออง (Atomization) ของสสารกระจายเป็นล้านหยดในรูปแบบของละออง ซึ่งพลังงานจำเป็นสำหรับกระบวนการเพื่อสร้างแรงเหวี่ยง ความดัน การเคลื่อนไหว หรือเสียงสะท้อน ในขณะที่ทำให้เป็นละออง เมื่อละอองสัมผัสกับอากาศร้อน ความชื้นที่พื้นผิวของหยดละอองจะถูกทำให้กลายเป็นไอ การระเหยเป็นไอเกิดขึ้นอย่างรวดเร็วเนื่องจากหยดละอองมีพื้นที่ผิวมาก เช่น ละออง  $2 \times 10^{12}$  หยด มีเส้นผ่านศูนย์กลาง 100 ไมโครเมตร จะมีพื้นที่ผิวรวมมากกว่า 60,000 ตารางเมตร ปัจจัยสำคัญเพื่อให้ได้อนุภาคแห้งที่มีประสิทธิภาพคือ การออกแบบระบบการอบแห้งแบบพ่นฝอยอย่างถูกต้อง

รูปแบบต่างๆของเครื่องอบแห้ง มีวัตถุประสงค์เพื่อกำจัดความชื้นโดยการใช้ประโยชน์จากความร้อนสู่ผลิตภัณฑ์และควบคุมความชื้น ความร้อนถูกใช้สำหรับทำความร้อนในบรรยากาศและการระเหย ความชื้นถูกควบคุมโดยการไหลของอากาศและการควบคุมอุณหภูมิของอากาศ กระบวนการอบแห้งแบบพ่นฝอยวงรอบเปิดประกอบด้วย 4 ขั้นตอน ดังภาพที่ 2.1 ซึ่งแต่ละขั้นตอนต้องดำเนินการตามการออกแบบเครื่องอบแห้งและผู้ใช้รวมถึงคุณสมบัติทางกายภาพและทางเคมีของสสารเข้าเพื่อกำหนดคุณลักษณะของผลิตภัณฑ์ที่ได้

### 2.1.2.1 การทำละออง (Atomization)

มีเครื่องทำละออง (Atomizer) เป็นตัวควบคุมอัตราการไหลไปยังห้องอบแห้ง (Drying Chamber) และกำหนดขนาดของอนุภาค ตลอดจนคุณภาพต่างๆ ของผงผลิตภัณฑ์ มีหน้าที่ทำให้สสารแตกตัวเป็นละอองฝอย และกระจายของเหลวหรือสารละลายให้มีขนาดเล็กพอที่จะควบคุมได้ เพื่อเพิ่มพื้นที่สัมผัสกับความร้อนให้มากขึ้น

เครื่องทำละอองที่ใช้ในเครื่องอบแห้งแบบพ่นฝอยมีหลายระบบ ซึ่งแต่ละระบบมีผลต่อ ลักษณะและคุณภาพของผงผลิตภัณฑ์ที่ได้ ซึ่งการเลือกชนิดเครื่องทำละอองขึ้นอยู่กับสมบัติของสสารเริ่มต้น เช่น ความหนืด (Viscosity) นอกจากนี้ยังขึ้นอยู่กับและคุณสมบัติของผงผลิตภัณฑ์ที่ต้องการ เช่น ขนาดของอนุภาค ความสามารถในการละลาย (Solubility) ความหนาแน่น (Density) ความสามารถในการเปียก (Wettability) เป็นต้น เครื่องทำละอองที่นิยมใช้ ได้แก่



ภาพที่ 2.1 แสดงกระบวนการของการอบแห้งแบบพ่นฝอยวงรอบเปิด

1) หัวฉีดอัดแรงดัน (Pressure Nozzles หรือ Nozzle Atomizer) เครื่องทำละอองประเภทนี้จะใช้แรงดันสูงดันให้สสารไหลผ่านรูเปิดขนาดเล็ก (Orifice) ซึ่งขนาดของอนุภาคจะแปรผันโดยตรงกับอัตราการไหลและความหนืดของสสาร หากเพิ่มแรงดันขนาดของอนุภาคจะเล็กลง ซึ่งเป็นวิธีที่ใช้ในรายงานฉบับนี้

2) เครื่องทำละอองแบบของไหล 2 ชนิด (Two-fluid atomizer หรือ Pneumatic Atomizer) สสารจะไหลกระทบกับก๊าซหรืออากาศที่เคลื่อนที่ด้วยความเร็วสูง ทำให้สสารแตกเป็นละอองฝอยได้ขนาดอนุภาคที่มีขนาดเล็กมากสามารถใช้ได้ดีกับสสารที่มีความหนืดสูง (Viscosity)

3) เครื่องทำละอองแบบจานเหวี่ยงหรือล้อหมุน (Centrifugal Atomizer หรือ Rotary Atomizer) มีลักษณะเป็นจาน (Disk, Wheel) โดยสสารจะไหลลงบริเวณใกล้กับจุดศูนย์กลางของจาน ที่หมุนด้วยความเร็วรอบสูงมาก(ประมาณ 2,000-20,000 รอบต่อนาที) ถูกเหวี่ยงด้วยแรงหนีศูนย์กลาง (Centrifugal Force) ให้สสารกระจายออกด้านข้างของจานเป็นละอองเล็ก ๆ ซึ่งขนาดของอนุภาคแปรผันโดยตรงกับอัตราการไหลและความหนืดของสสารและแปรผกผันกับอัตราหมุนและเส้นผ่านศูนย์กลางของจานหมุน ใช้กับผลิตภัณฑ์ที่มีความหนืดสูง และไม่ใช่น้ำอย่างเดียว ซึ่งอาจมีปัญหาการอุดตันหัวฉีด

ในบางกรณีของอุตสาหกรรมซึ่งต้องการอนุภาคของผลิตภัณฑ์ที่มีขนาดใกล้เคียงกันมาก เพื่อให้มั่นใจได้ว่าการดูดซับภายในกายภาพหรือพฤติกรรมปลดปล่อย (Release Behavior) ของอนุภาคทุกอนุภาคเป็นไปอย่างสม่ำเสมอและใกล้เคียงกัน สามารถใช้เครื่องทำละอองที่ทำงานโดยการกระตุ้นของคลื่นเสียง (Ultrasonic Atomizer) หรือไฟฟ้าสถิต (Electrostatic Atomizer) ได้

### 2.1.2.2 การอบแห้งที่ห้องอบแห้ง (Drying Chamber)

เป็นบริเวณที่เกิดการทำแห้งของสสาร โดยอากาศร้อนแห้ง ซึ่งเป็นอากาศที่ถูกดูดผ่านระบบกรองอากาศและถูกทำให้ร้อน จะสัมผัสกับสสารที่ถูกทำให้เป็นละอองบริเวณนี้ เมื่อความชื้นในละอองถูกทำให้กลายเป็นไอเมื่อสัมผัสกับความชื้น จะได้อนุภาคผงของผลิตภัณฑ์ตกลงสู่ด้านล่างของห้องอบแห้ง ผงบางส่วนที่หลุดออกมากับอากาศจะถูกแยกออกจากอากาศโดยใช้ไซโคลนหรือถุงกรอง ซึ่งจะรวมเข้าเป็นผลิตภัณฑ์ในที่สุด

### 2.1.2.3 การระเหยเป็นไอ (Evaporation)

เกิดการระเหยของน้ำจากละอองของสสาร ซึ่งการสัมผัสระหว่างอากาศร้อนและละอองอาจเป็นได้หลายทิศทาง

1) อากาศและละอองไหลทางเดียวกัน (Co-current flow) ละอองและอากาศร้อนป้อนเข้าในทิศทางเดียวกัน อุณหภูมิอากาศร้อนจะลดลงระหว่างการทำให้แห้ง เหมาะกับสารละลายที่ไวต่อความร้อน

2) อากาศและละอองไหลสวนทางกัน (Counter-current flow)

3) การไหลแบบผสม (Mixed flow) การไหล 2 ลักษณะเกิดขึ้นพร้อมกัน อากาศร้อนจะถูกป้อนเข้าทั้งด้านบนและด้านล่าง เพื่อเพิ่มประสิทธิภาพของการทำให้แห้ง

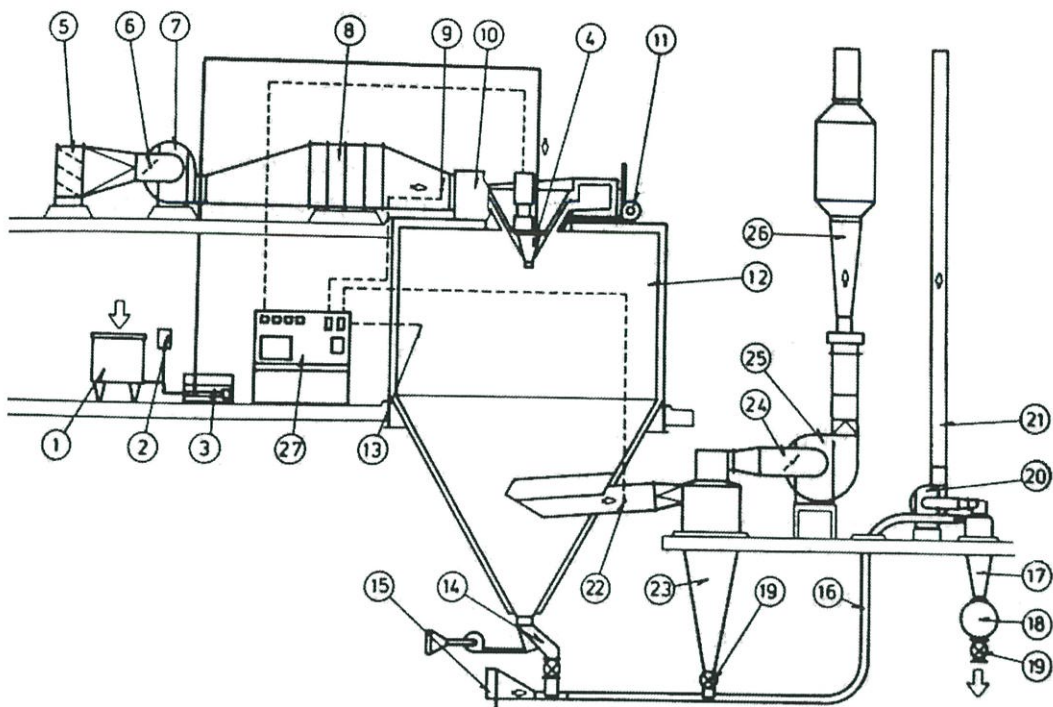
### 2.1.2.4 การแยกอนุภาคผงออกจากอากาศ

อุปกรณ์แยกอนุภาคผงออกจากอากาศเช่น ไซโคลน (cyclone) และถุงกรอง (bag filter) ปัจจัยที่มีผลต่อประสิทธิภาพและคุณภาพของผลิตภัณฑ์ที่ทำแห้งด้วยเครื่องอบแห้งแบบพ่นฝอยได้แก่ อัตราการไหลของสสารขาเข้า (Feed) ความหนืดของสสาร และอุณหภูมิของอากาศร้อนขาเข้า ไซโคลนโดยทั่วไปมีลักษณะเป็นถังกระบอกแนวตั้ง เมื่ออากาศหมุนเป็นแบบเกลียวรอบทรงกระบอก เกิดแรงหมุนเหวี่ยงหนีศูนย์กลาง (Centrifugal force) ซึ่งช่วยเร่งอัตราการตกของอนุภาคผง

ในกรณีที่อนุภาคซึ่งผ่านการอบแห้งแบบพ่นฝอยมีขนาดเล็ก และไม่สามารถที่จะคืนรูป (Reconstitute) ได้ดี อาจนำอนุภาคดังกล่าวไปทำการมวบรวม (Agglomerate) ให้เป็นอนุภาคผงที่มีขนาดใหญ่ขึ้นเพื่อให้คืนรูปได้ดีขึ้นในเครื่องอบแห้งแบบฟลูอิดไธซ์เบด (Fluidize bed) โดยอาจติดตั้งเครื่องอบแห้งแบบฟลูอิดไธซ์เบด (Fluidize bed) ไว้เป็นส่วนหนึ่งของเครื่องอบแห้งแบบพ่นฝอย (Spray dryer) โดยติดตั้งไว้ที่ฐาน ซึ่งเป็นการลดความเสี่ยงจากการปนเปื้อนของผลิตภัณฑ์ที่เกิดจากการย้ายอนุภาคผงจากเครื่องอบแห้งเครื่องหนึ่งไปยังเครื่องอบแห้งอีกเครื่องหนึ่ง และทำให้การดักจับฝุ่นเป็นไปได้โดยง่าย

อุปกรณ์ที่เกี่ยวข้องในเครื่องอบแห้งแบบพ่นฝอยวงรอบเปิด ซึ่งจำแนกอุปกรณ์ตามระบบในกระบวนการอบแห้งดังภาพที่ 2.2 ประกอบด้วย

- 1) ถังบรรจุผลิตภัณฑ์สสารขาเข้า
- 2) ถังบรรจุน้ำขาเข้าสำหรับใช้ขณะเครื่องอบแห้งเริ่มทำงาน และเลิกใช้งาน
- 3) เครื่องสูบลำสสารขาเข้า (Feed pump)



ภาพที่ 2.2 อุปกรณ์ที่เกี่ยวข้องในกระบวนการอบแห้งแบบพ่นฝอยวงรอบปิด

ระบบเครื่องทำละออง

4) เครื่องทำละออง (Atomizer)

ระบบอากาศ

5) ตัวกรองอากาศขาเข้า

6) อุปกรณ์ปรับอัตราไหลอากาศ (Damper)

7) พัดลมระบายอากาศ

8) เครื่องเป่าลมร้อน

9) อุปกรณ์วัดอุณหภูมิอากาศขาเข้า

10) ตัวกระจายอากาศ

11) พัดลมกระจายอากาศเย็น

12) ฉนวนห้องอบแห้ง

13) อุปกรณ์ตรวจจับความดันภายในห้องอบแห้ง

ระบบแยกผลิตภัณฑ์

14) หน่วยระบายอนุภาคผงผลิตภัณฑ์ที่ฐานของห้องอบแห้ง

15) ตัวกรองนำพาอากาศเข้า

16) ท่อนำพาอากาศ

17) ไชโคลนลำเลียง

18) ถังอนุภาคผง (Powder hopper)

- 19) โรตารีวาล์ว (Rotary valve)
- 20) พัดลมลำเลียงอากาศ
- 21) ท่อระบายอากาศ
- 22) อุปกรณ์วัดอุณหภูมิอากาศขาออก
- 23) ไส้โคลน ถังแยกหลัก
- 24) ท่อและอุปกรณ์ปรับอัตราการระบายอากาศจากไส้โคลน
- 25) พัดลมระบายอากาศ
- 26) เครื่องบำบัดอากาศแบบเวนทูรี (Venturi scrubber)
- 27) แผงควบคุมเครื่องอบแห้งแบบพ่นฝอย

การระบายอากาศสู่บรรยากาศผ่านพัดลมระบายอากาศ (25) และเครื่องบำบัดอากาศแบบเวนทูรี (26) อนุภาคผงที่ถูกทำให้แห้งในอากาศมีระบบลำเลียงคือสะสมที่ฐานของไส้โคลน ลำเลียง (17) มีเครื่องมือวัดประกอบด้วยการวัดอุณหภูมิและควบคุม (9,22) และการวัดความดันภายในห้องอบแห้ง (13) ปรับอัตราการไหลของอากาศโดยแดมเปอร์ (6) นำพาอากาศออกสู่บรรยากาศได้โดยท่อระบายอากาศ (21) หรือส่งกลับไปที่ไส้โคลนหลัก(23) หรือเครื่องบำบัดอากาศแบบเวนทูรี (26)

### 2.1.3 ระบบควบคุมในกระบวนการอบแห้ง

เป้าหมายของระบบควบคุมในกระบวนการอบแห้งแบบพ่นฝอย คือ ผลิตภัณฑ์ที่ถูกทำให้แห้งอย่างมีประสิทธิภาพ ถึงแม้ว่าจะมีการรบกวนเกิดขึ้นภายในการอบแห้งและการผันแปรของแหล่งจ่ายขาเข้า และสิ่งที่มีผลกระทบต่อผลิตภัณฑ์เพื่อการควบคุมคือความชื้น การใช้งานเครื่องอบแห้งแบบพ่นฝอยในปัจจุบันส่วนใหญ่จะใช้อุณหภูมิขาออก เป็นพารามิเตอร์ที่ถูกควบคุม ซึ่งอุณหภูมิแสดงถึงคุณภาพของผลิตภัณฑ์ เช่น ความหนาแน่น สี รส

เครื่องอบแห้งแบบพ่นฝอยสามารถใช้งานได้ทั้งการควบคุมด้วยมือ (manually) และการควบคุมแบบอัตโนมัติ (Automatically) การควบคุมด้วยมือเหมาะสำหรับระบบเล็กๆอย่างในห้องทดลองหรืออุตสาหกรรมขนาดเล็ก ที่ใช้งานแบบไม่ต่อเนื่องและเป็นผลิตภัณฑ์ที่ถูกทำให้แห้งได้ง่าย โดยการควบคุมด้วยมือสามารถใช้ในอุตสาหกรรมขนาดใหญ่ได้เช่นกัน แต่ต้องมีบุคลากรสำหรับปฏิบัติงานอย่างต่อเนื่อง แต่การควบคุมคุณภาพของผลิตภัณฑ์ให้คงที่สำหรับการผลิตระยะยาวต้องใช้ในการควบคุมแบบอัตโนมัติ หรือกึ่งอัตโนมัติ โดยการควบคุมแบบอัตโนมัติสามารถรักษาอุณหภูมิขาออกที่มีการเบี่ยงเบนเพียงเล็กน้อยจากค่าตั้งต้นการควบคุม ( $\pm 0.5$  ถึง 1 องศาเซลเซียส) ในช่วงเวลาที่มากกว่าการควบคุมด้วยมือ

### 2.1.4 การอบแห้งสำหรับอุตสาหกรรมเม็ดสีย้อม

อนุภาคของเม็ดสีย้อมมีลักษณะกลวง เปราะ และกระจายตัวได้ง่าย การไล่ความชื้นจึงทำได้โดยใช้เครื่องอบแห้งแบบพ่นฝอย สีย้อมจะถูกทำให้แห้งโดยอุณหภูมิขาเข้า ตั้งแต่ 120 ถึง 450 องศาเซลเซียส ขึ้นกับปริมาณสีย้อมขาเข้า คุณสมบัติและขนาดผงสีที่ต้องการ สำหรับเครื่องทำละอองแบบจานเหวี่ยงหรือล้อหมุน จะได้เม็ดสีขนาด 50 ถึง 120 ไมโครเมตร และเครื่องทำละอองแบบหัวฉีดแรงดัน จะได้

เม็ดสีขนาด 120 ถึง 250 ไมโครเมตร หากผงสีที่ได้มีความชื้นต่ำและอนุภาคน้ำหลังจากผ่านห้องอบแห้งแล้ว จะใช้ไซโคลนหรือถุงกรอง (Bag filter) แยกผงสีกับอากาศ เพื่อการจัดเก็บ และนำอากาศกรองผ่านเครื่องบำบัดอากาศแบบเวนทูลีเพื่อปล่อยออกสู่บรรยากาศสำหรับกระบวนการแบบวงรอบเปิด

เครื่องอบแห้งในอุตสาหกรรมเม็ดสีย่อมง่ายต่อการทำความสะอาด และระบบการทำความสะอาดแบบไม่ถอดชิ้นส่วน (Cleaning In Place) เขียนย่อว่า CIP เป็นระบบการทำความสะอาดสำหรับกระบวนการที่อุปกรณ์ต่อกันเป็นสายการผลิต หรือเครื่องจักรแต่ละตัวแบบอัตโนมัติซึ่งสามารถทำความสะอาดโดยไม่ต้องถอดชิ้นส่วนต่าง ๆ ของอุปกรณ์ออกจากกัน เป็นการทำความสะอาดแบบเปียก (Wet Cleaning) ด้วยสารละลาย CIP (CIP solution) หลายชนิด เช่น น้ำ สารซักฟอก (Detergent) สารทำความสะอาด (Cleaning Agent) หรือ น้ำยาฆ่าเชื้อ (Sanitizer) ใช้ทำความสะอาดกระบวนการหลังการผลิตเพื่อเตรียมการผลิตในสัปดาห์ถัดไป

กระบวนการทำความสะอาดแบบไม่ถอดชิ้นส่วน (CIP) ถูกพิจารณาให้อยู่ในขั้นตอนการออกแบบขั้นพื้นฐานของระบบเครื่องอบแห้งแบบพ่นฝอย เนื่องจากผงเม็ดสีสามารถระเบิดได้จากการผสมกับอากาศและเสี่ยงต่อการติดไฟ ถึงแม้ว่าเครื่องอบแห้งจะมีมาตรการความปลอดภัยถึงอันตรายที่เกี่ยวข้อง ซึ่งข้อกำหนดส่วนมากของเครื่องอบแห้งเม็ดสีจะเกี่ยวกับวัตถุไวไฟ ต้องมีระบบดับเพลิงและระบายความดัน โดยช่องระบายความดันจะอยู่ด้านบนของเครื่องอบแห้งในบริเวณที่ไม่มีผงสี แต่หากมีพื้นที่ไม่เพียงพอสามารถไว้ที่ด้านข้างห้องอบแห้งได้ และหากอยู่ในสถานะเสี่ยงต้องจำกัดอุณหภูมิเข้าได้ โดยปกติห้องอบแห้งที่มีความดันสูงจะมีพื้นที่สำหรับใช้ระบายความดันออก ผลิตภัณฑ์เม็ดสีที่ได้ถูกใช้ในอุตสาหกรรมยาง สีย้อม หมึก และพลาสติก<sup>[1]</sup>

## 2.2 ความรู้พื้นฐานเกี่ยวกับ PLC

เครื่องควบคุมเชิงตรรกะที่สามารถโปรแกรมได้ PLC (Programmable Logic Controller) เป็นอุปกรณ์ชนิดโซลิด-สเตท (Solid state) ที่ทำงานแบบลอจิก (Logic function) การออกแบบการทำงานของ PLC จะคล้ายกับหลักการการทำงานของคอมพิวเตอร์ จากหลักการพื้นฐานแล้ว PLC จะประกอบด้วยอุปกรณ์ที่เรียกว่า Solid state digital logic element เพื่อให้ทำงานและตัดสินใจแบบลอจิก PLC ใช้สำหรับควบคุมกระบวนการทำงานของเครื่องจักรและอุปกรณ์ในโรงงานอุตสาหกรรม

การใช้ PLC สำหรับควบคุมเครื่องจักรหรืออุปกรณ์ต่างๆ ในโรงงานอุตสาหกรรมจะมีข้อได้เปรียบกว่าการใช้ระบบของรีเลย์ (Relay) ซึ่งจำเป็นจะต้องเดินสายไฟฟ้า หรือที่เรียกว่า Hard-wire ฉะนั้นเมื่อมีความจำเป็นต้องเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่ก็ต้องเดินสายไฟฟ้าใหม่ ซึ่งเสียเวลาและค่าใช้จ่ายสูง แต่เมื่อเปลี่ยนมาใช้ PLC แล้ว การเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่นั้นทำได้โดยการเปลี่ยนโปรแกรมใหม่เท่านั้น นอกจากนี้แล้ว PLC ยังใช้ระบบโซลิด-สเตท ซึ่งน่าเชื่อถือกว่าระบบเดิม กินกระแสไฟน้อยกว่า และสะดวกกว่าเมื่อต้องการขยายขั้นตอนการทำงานของเครื่องจักร นอกจากนี้ PLC ยังถูกออกแบบมาเพื่อใช้ในงานอุตสาหกรรมโดยเฉพาะ ต้องมีความน่าเชื่อถือสูง ทนต่อสภาพแวดล้อมของโรงงานอุตสาหกรรม เช่น ความร้อน ความเย็น การสั่นสะเทือน สัญญาณรบกวน เป็นต้น มีอายุการใช้งานยาว เป็นตัวควบคุมมีฟังก์ชันการทำงานทั้งดิจิทัลและอนาล็อก

## 2.2.1 โครงสร้างของ PLC

PLC ประกอบด้วย หน่วยประมวลผลกลาง หน่วยความจำ หน่วยรับข้อมูล หน่วยส่งข้อมูล และหน่วยป้อนโปรแกรมดังภาพที่ 2.3 สำหรับ PLC ขนาดเล็ก ส่วนประกอบทั้งหมดของ PLC จะรวมกันเป็นเครื่องเดียว แต่หากเป็นขนาดใหญ่สามารถแยกออกเป็นส่วนประกอบย่อยๆได้

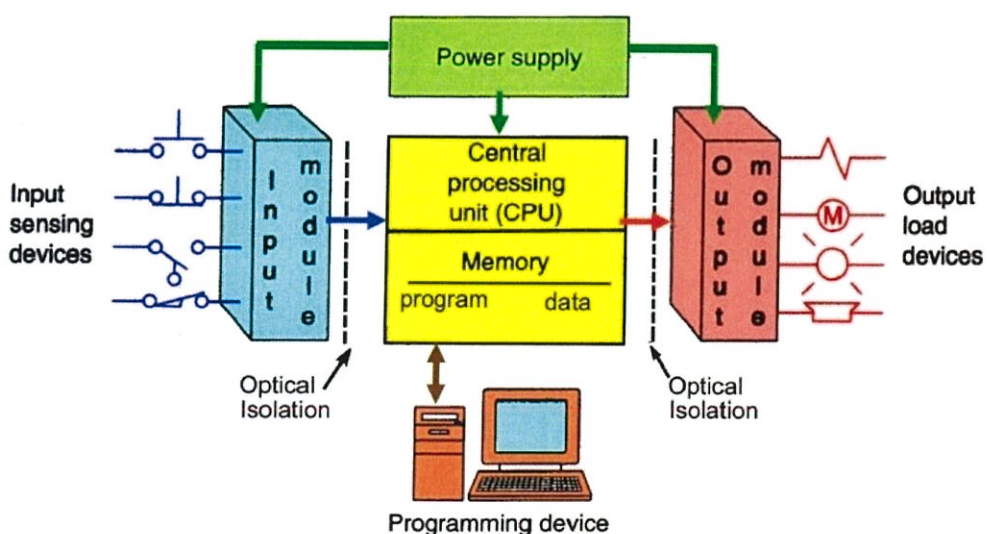
หน่วยความจำของ PLC ประกอบด้วย หน่วยความจำชนิด RAM และ ROM หน่วยความจำชนิด RAM ทำหน้าที่เก็บโปรแกรมของผู้ใช้และข้อมูลสำหรับการปฏิบัติงานของ PLC ส่วนหน่วยความจำชนิด ROM ทำหน้าที่เก็บโปรแกรมสำหรับการปฏิบัติงานของ PLC ตามโปรแกรมของผู้ใช้

ROM ย่อมาจาก Read Only Memory สามารถโปรแกรมได้แต่ลบไม่ได้ หากเกิดการชำรุดไม่สามารถซ่อมได้

RAM (Random Access Memory) หน่วยความจำประเภทนี้จะมีแบตเตอรี่เล็กๆต่อไว้เพื่อใช้เลี้ยงข้อมูลเมื่อเกิดไฟดับ การอ่านและเขียนโปรแกรมลงใน RAM ทำได้โดยง่าย จึงเหมาะกับการใช้งานในระยะทดลองเครื่อง ที่มีการเปลี่ยนแปลงแก้ไขโปรแกรมบ่อยๆ

EPROM (Erasable Programmable Read Only Memory) หน่วยความจำชนิดนี้ต้องใช้เครื่องมือชนิดพิเศษในการเขียนโปรแกรม การลบโปรแกรมทำได้โดยใช้แสงอัลตราไวโอเล็ตหรือตากแดดร้อนๆนานๆ ข้อดีคือโปรแกรมจะไม่สูญหายหากเกิดไฟดับ จึงเหมาะกับการใช้งานที่ไม่ต้องมีการเปลี่ยนโปรแกรม

EEPROM (Electrical Erasable Programmable Read Only Memory) หน่วยความจำชนิดนี้ ไม่ต้องใช้เครื่องมือพิเศษในการเขียนและลบโปรแกรม แต่ใช้วิธีการทางไฟฟ้า เหมือนกับ RAM นอกจากนี้ไม่จำเป็นต้องมีแบตเตอรี่สำรองไฟเมื่อไฟดับ ราคาสูง โดยประกอบด้วยคุณสมบัติที่ดีของทั้ง RAM และ EPROM



ภาพที่ 2.3 ส่วนประกอบของ PLC

## 2.2.2 ส่วนประกอบของ PLC

PLC แบ่งออกเป็น 3 ส่วน ดังภาพที่ 2.3 คือ

### 2.2.2.1 หน่วยประมวลผลกลาง (Central Processing Unit : CPU)

CPU เป็นส่วนมันสมองของระบบ ภายใน CPU จะประกอบด้วยวงจร Logic Gate ชนิดต่างๆหลายชนิด และมี Microprocessor-based ใช้สำหรับแทนอุปกรณ์จำพวกรีเลย์ (Relay) ตัวนับจำนวน (Counter) ตัวนับเวลา (Timer) และการทำงานเป็นลำดับ (Sequence) เพื่อให้ผู้ใช้ได้ ออกแบบใช้วงจรรีเลย์ แลตเตอร์ ลอจิก (Relay Ladder Logic) เข้าไปได้

CPU จะอ่านค่า (Read) ข้อมูลอินพุต (Input Data) จากอุปกรณ์ให้สัญญาณ (Sensing Device) ต่าง ๆ จากนั้นจะปฏิบัติการและเก็บรักษาข้อมูลโดยใช้โปรแกรมจากหน่วยความจำ และส่งข้อมูลที่เหมาะสมถูกต้องไปยังอุปกรณ์ควบคุม (Control Device) แหล่งของไฟฟ้ากระแสตรง (DC current) สำหรับใช้สร้าง ความต่างศักย์ต่ำ (Low Level Voltage) ซึ่งใช้โดยโปรเซสเซอร์ (Processor) และอินพุตเอาต์พุตโมดูล (I/O module) และแหล่งจ่ายไฟนี้จะเก็บไว้ที่ CPU หรือแยกออกไปติดตั้งที่จุดอื่นได้ขึ้นกับผู้ผลิตแต่ละราย

ภายในหน่วยประมวลผลกลาง CPU มีส่วนที่สำคัญในชุดของคือ Processor-Memory Module ซึ่งเป็นส่วนสำคัญชิ้นใหญ่ในตัวเรือนของ CPU เป็นมันสมองที่ควบคุมโปรแกรม ภายในประกอบด้วย Microprocessor Memory chips ที่ทำหน้าที่เก็บและเรียกข้อมูลจากหน่วยความจำ และติดต่อกับวงจรที่ต้องการ



ภาพที่ 2.4 I/O Scan และ Program Scan

การประมวลผลของ CPU จากโปรแกรมทำได้โดยรับข้อมูลมาจากหน่วยอินพุต และเอาต์พุต และส่งข้อมูลสุดท้ายที่ได้จากการประมวลผลไปยังหน่วยเอาต์พุต เรียกว่าการสแกน (Scan) ดังภาพที่ 2.4 ซึ่งใช้เวลาจำนวนหนึ่งเรียกว่า เวลาสแกน (Scan Time) เวลาในการสแกนแต่ละรอบใช้เวลาประมาณ 1 ถึง 100 มิลลิวินาที (10 มิลลิวินาที คือ 100 ครั้งต่อวินาที) ทั้งนี้ขึ้นอยู่กับข้อมูลและความยาวของโปรแกรมหรือจำนวนอินพุตเอาต์พุตหรือจำนวนอุปกรณ์ที่ต่อจาก PLC เช่นเครื่องพิมพ์ จอภาพ

เป็นต้น อุปกรณ์เหล่านี้จะทำให้เวลาในการสแกนยาวนานขึ้น การเริ่มต้นการสแกนเริ่มจากการรับค่าของสถานะอุปกรณ์จากหน่วยอินพุตมาเก็บไว้ในหน่วยความจำ (Memory) เสร็จแล้วจะทำการปฏิบัติการตามโปรแกรมที่เขียนไว้ทีละคำสั่งจากหน่วยความจำนั้นจนสิ้นสุด แล้วส่งไปที่หน่วยเอาต์พุต

เมื่อมีการเปลี่ยนแปลงระหว่างเวลาที่กำลังทำการสแกน PLC จะไม่สามารถเปลี่ยนแปลงค่านั้นได้ แต่จะรับไว้ในหน่วยความจำ หลังจากนั้นเมื่อทำการสแกนตามโปรแกรมเรียบร้อยแล้ว จึงนำผลการเปลี่ยนแปลงครั้งหลังสุดส่งออกไปที่หน่วยเอาต์พุต

#### 2.2.2.2 อินพุต/เอาต์พุต (Input Output : I/O)

ส่วนของอินพุตและเอาต์พุต (I/O Unit) จะต่อร่วมกับชุดควบคุมเพื่อรับสถานะและสัญญาณต่างๆ เช่น หน่วยอินพุตรับสัญญาณหรือสถานะแล้วส่งไปยัง CPU เพื่อประมวลผล เมื่อ CPU ประมวลผลแล้วจะส่งให้ส่วนของเอาต์พุต เพื่อให้อุปกรณ์ทำงานตามที่โปรแกรมเอาไว้ สัญญาณอินพุตจากภายนอกที่เป็นสวิตช์และตัวตรวจจับชนิดต่างๆ จะถูกแปลงให้เป็นสัญญาณที่เหมาะสมถูกต้อง ไม่ว่าจะ เป็น AC หรือ DC เพื่อส่งให้ CPU ดังนั้น สัญญาณเหล่านี้จึงต้องมีความถูกต้องเพื่อป้องกัน CPU เสียหาย

สัญญาณอินพุตที่ดีจะต้องมีคุณสมบัติและหน้าที่ดังนี้

1) ทำให้สัญญาณเข้า ได้ระดับที่เหมาะสมกับ PLC

2) การส่งสัญญาณระหว่างอินพุตกับ CPU แยกสัญญาณ (Isolate) ทางไฟฟ้า ออกจากกัน ผ่านโฟโตทรานซิสเตอร์ (Phototransistor) ซึ่งเป็นอุปกรณ์ส่งสัญญาณผ่านลำแสงเพื่อป้องกัน CPU เสียหายเมื่ออินพุตเกิดลัดวงจร

3) หน้าสัมผัสจะต้องไม่สั่นสะเทือน (Contact Chattering)

อุปกรณ์ที่ใช้เป็นสัญญาณอินพุตได้แก่ ลิ้มิตสวิตช์ (Limit Switch) ตัวนับเวลา (Timer) พรอกซิมิตีส์วิตช์ (Proximity Switch) โฟโตอิเล็กทริกสวิตช์ (Photoelectric Switch) เอนโค้ดเดอร์ (Encoder) ตัวนับจำนวน (Counter) เป็นต้น

สัญญาณเอาต์พุต จะทำหน้าที่รับค่าสถานะที่ได้จากการประมวลผลของ CPU แล้วนำค่าเหล่านี้ไปควบคุมอุปกรณ์ทำงาน เช่น รีเลย์ โซลินอยด์ หรือหลอดไฟ เป็นต้น นอกจากนั้นแล้วยังทำหน้าที่แยกสัญญาณของหน่วยประมวลผลกลาง (CPU) ออกจากอุปกรณ์เอาต์พุต โดยปกติเอาต์พุตนี้ จะมีความสามารถขับโหลดด้วยกระแสไฟฟ้าประมาณ 1-2 แอมแปร์ แต่ถ้าโหลดต้องการกระแสไฟฟ้ามากกว่านี้ จะต้องต่อเข้ากับอุปกรณ์ขับอื่นเพื่อขยายให้รับกระแสไฟฟ้ามากขึ้น เช่น รีเลย์หรือคอนแทคเตอร์ เป็นต้น

อุปกรณ์ที่ใช้เป็นสัญญาณเอาต์พุต ได้แก่ มอเตอร์ไฟฟ้า (Electric Motor) รีเลย์ (Relay) โซลินอยด์ (Solenoid) ขดลวดความร้อน (Heat Coil) หลอดไฟ (Lamp) เป็นต้น

#### 2.2.2.3 อุปกรณ์การโปรแกรม (Programming Device)

การสั่งให้ PLC ทำงาน จะต้องป้อนโปรแกรมให้กับ PLC ก่อน ซึ่งอุปกรณ์ที่ใช้ในการป้อนโปรแกรมให้กับ PLC นั้นสามารถแบ่งได้เป็น 2 ประเภท

1) เครื่องป้อนโปรแกรมแบบมือถือ (Hand Held) ทำหน้าที่ ควบคุมโปรแกรมของผู้ใช้ลงในหน่วยความจำของ PLC นอกจากนี้ยังทำหน้าที่ติดต่อระหว่างผู้ใช้กับ PLC เพื่อให้ผู้ใช้สามารถตรวจการปฏิบัติงานของ PLC ผลการควบคุมเครื่องจักรและกระบวนการตามโปรแกรมควบคุมที่ผู้ใช้เขียนขึ้นได้ โดยเครื่องป้อนโปรแกรม (Hand Held) แต่ละยี่ห้อจะไม่เหมือนกันแต่มีจุดประสงค์ในการใช้งานที่เหมือนกัน

2) คอมพิวเตอร์ (Computer) สามารถใช้ในการเขียนโปรแกรมให้กับ PLC ได้ โดยใช้งานร่วมกับซอฟต์แวร์เฉพาะของ PLC ยี่ห้อนั้น

### 2.2.3 ความสามารถของ PLC

PLC สามารถควบคุมงานได้ 3 ลักษณะคือ

#### 2.2.3.1 งานที่ทำตามลำดับก่อนหลัง (Sequence Control) ตัวอย่างเช่น

- 1) การทำงานของระบบรีเลย์
- 2) การทำงานของ Timer Counter
- 3) การทำงานของ P.C.B. Card
- 4) การทำงานในระบบกึ่งอัตโนมัติ ระบบอัตโนมัติ หรืองานที่เป็นกระบวนการทำงานของเครื่องจักรกลต่างๆ

#### 2.2.3.2 งานควบคุมสมัยใหม่ (Sophisticated Control) ตัวอย่างเช่น

- 1) การทำงานทางคณิตศาสตร์ เช่น บวก ลบ คูณ หาร
- 2) การควบคุมแบบอนาล็อก (Analog Control) เช่น การควบคุมอุณหภูมิ (Temperature) และความดัน (Pressure)
- 3) การควบคุม P.I.D. (Proportional-Integral-Derivation)
- 4) การควบคุมเซอร์โวมอเตอร์ (Servo-motor Control)
- 5) การควบคุม Stepper-motor
- 6) Information Handling

#### 2.2.3.3 การควบคุมเกี่ยวกับงานอำนวยการ (Supervisory Control) ตัวอย่างเช่น

- 1) งานสัญญาณเตือน (Alarm) และ Process Monitoring
- 2) Fault Diagnostic and Monitoring
- 3) งานต่อร่วมกับคอมพิวเตอร์ (RS-232C/RS422)
- 4) Printer/ASCII Interfacing
- 5) งานควบคุมอัตโนมัติในโรงงานอุตสาหกรรม (Factory Automation Networking)
- 6) LAN (Local Area Network)

- 7) WAN (Wide Area Network)
- 8) FA , FMS., CIM. เป็นต้น

## 2.2.4 การติดตั้ง PLC

### 2.2.4.1 ข้อควรพิจารณาก่อนติดตั้ง PLC

- 1) พื้นที่ในการติดตั้งมีเพียงพอหรือไม่
- 2) จะต้องเผื่อไว้ขยายในอนาคตหรือไม่
- 3) การซ่อมบำรุงต้องทำได้ง่าย
- 4) อุณหภูมิที่เกิดขึ้นจากเครื่องจักรมีผลกระทบต่อ PLC หรือไม่
- 5) วิธีการป้องกัน PLC จากสภาพแวดล้อมที่ไม่ปลอดภัย

### 2.2.4.2 สภาพแวดล้อมหรือสถานที่ที่ไม่ควรติดตั้ง PLC

- 1) มีแสงแดดส่องโดยตรง
- 2) มีอุณหภูมิต่ำกว่า 0 °C หรือสูงกว่า 55 °C
- 3) มีฝุ่น หรือไอเกลือ
- 4) มีความชื้นมาก
- 5) มีก๊าซที่มีคุณสมบัติกัดกร่อน หรือไวไฟ
- 6) สั่นสะเทือนมาก

## 2.2.5 คุ้มครองสำหรับ PLC

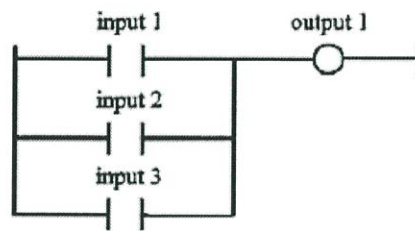
- 1) ต้องป้องกันไม่ให้ PLC เสียหายจากการใช้งานหรือจากส่วนอื่นๆ เช่น จากสิ่งแวดล้อมหรือสิ่งปนเปื้อนในอากาศ เช่น ความชื้น น้ำมัน ฝุ่นผง ก๊าซที่มีฤทธิ์การกัดกร่อน
- 2) มีขนาดใหญ่เพียงพอ สะดวกในการเดินสายไฟต่างๆ
- 3) ควรติดตั้งตู้ PLC ห่างจากแผงควบคุมไฟฟ้าแรงสูงอย่างน้อย 8 นิ้ว
- 4) มีสายดิน
- 5) ควรแยกการติดตั้งอุปกรณ์ไฟฟ้าแรงสูง
- 6) ควรแยกการติดตั้งกับอุปกรณ์ที่มีความร้อนสูง เช่น หม้อแปลง ตัวต้านทานขนาดใหญ่
- 7) ไม่ควรให้ PLC ติดตั้งอยู่บนเพดาน หรืออยู่กับพื้น
- 8) ถ้ามีอุณหภูมิสูงกว่า 60 °C ควรติดพัดลมเป่าระบายความร้อน
- 9) ควรต่อสายดินแยกออกจากอุปกรณ์ไฟฟ้าตัวอื่น คือ สายดินควรมีขนาด 2 ตารางมิลลิเมตร หรือใหญ่กว่า และค่าความต้านทานของสายดินไม่ควรเกิน 100 โอห์ม <sup>[2]</sup>

## 2.2.6 ภาษาที่ใช้ในการเขียนโปรแกรมให้กับ PLC

โปรแกรม ตามมาตรฐาน IEC61131-3 มี 5 ภาษา คือ LD (Ladder diagram), FBD (Function block diagram), IL (Instruction List) , ST (Structure text) , SFC

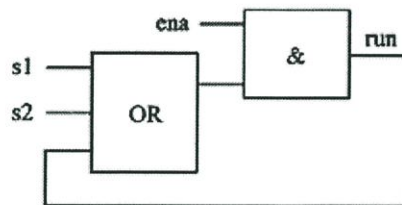
(sequential Function chart) ถึงแม้ว่าลักษณะโครงสร้างของแต่ละภาษาจะมีความแตกต่างกัน แต่ส่วนประกอบต่างๆ ในโปรแกรมมีลักษณะเดียวกัน เช่น ลักษณะการประกาศตัวแปร ฟังก์ชันและฟังก์ชันบล็อก เป็นต้นแต่ อย่างไรก็ตาม สามารถเขียนโปรแกรมโดยใช้หลายภาษารวมกันได้

1) LD (Ladder Diagram) จะเป็นที่เขียนอยู่ในรูปกราฟิก ซึ่งมีพื้นฐานมาจากวงจรควบคุมแบบรีเลย์และวงจรไฟฟ้า ซึ่งแลตเตอร์ไดอะแกรมจะประกอบด้วย ราง (Rail) ทั้งซ้ายขวาของไดอะแกรมเพื่อใช้สำหรับเชื่อมต่ออุปกรณ์ที่เป็นสวิตช์หน้าสัมผัส เพื่อเป็นทางผ่านของกระแสและมีขดลวด (Coil) เป็นเอาต์พุต



ภาพที่ 2.5 Ladder Diagram

2) FBD (Function Block Diagram) เป็นภาษาที่ฟังก์ชันการทำงานในรูปแบบของกราฟิกเช่นเดียวกับภาษาแลตเตอร์และเชื่อมกันเป็นโครงข่าย การเขียนโปรแกรมในรูปแบบของฟังก์ชันบล็อกไดอะแกรมจะมีพื้นฐานมาจากลอจิกไดอะแกรม



ภาพที่ 2.6 Function Block Diagram

3) IL (Instruction List) จะเป็นภาษาที่เขียนในรูปของข้อความ และมีลักษณะคล้ายกับภาษาแอสเซมบลี (Assembly) และภาษาเครื่อง (Machine code) และส่วนที่ถูกดำเนินการ (Operand) จะเห็นว่าในภาษาปัจจุบัน LD, FBD และ IL เป็นภาษาที่บริษัทผู้ผลิต PLC/PC ในปัจจุบัน กำหนดให้ใช้ในการเขียนโปรแกรม ซึ่งในแต่ละบริษัทจะมีการพัฒนารูปแบบของฟังก์ชัน และฟังก์ชันบล็อกมีความแตกต่างกัน

```

Label    LD    a1    (* result :=a1 *)
         ADD( a2    (* delayed ADD, result :=a2 *)
         MUL( a3    (* delayed MUL, result :=a3 *)
         SUB    a4    (* result :=a3-a4 *)
         )        (* execute delayed MUL, *)
         (* result :=a1+(a2*(a3-a4) *a5 *)
         ADD    a6    (* a1+(a2*(a3-a4)*a5)+a6 *)
         ST    res    (* store current result in res *)

```

ภาพที่ 2.7 Instruction list

4) ST (Structure text) ST จะเป็นภาษาในระดับสูง โดยพื้นฐานมาจากภาษา Pascal ซึ่งประกอบด้วย นิพจน์และคำสั่ง โดยคำสั่งทั่วไปจะอยู่ในรูปของคำสั่งเกี่ยวกับการเลือกการทำงาน เช่น IF.....THEN.....ELSE และ คำสั่งเกี่ยวข้องกับการทำงานซ้ำ เช่น FOR , WHILE เป็นต้น

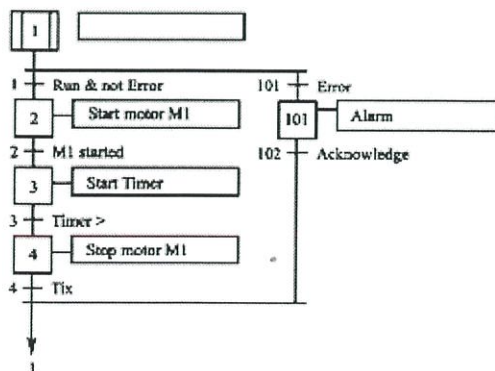
```

D := B*B -4*A*C;
IF D <0.0 THEN Nroots :=0;
ELSIF D= 0.0 THEN
    Nroot:=1;
    X1 := -B/(2.0*A);
ELSE Nroots :=2;
    X1 := (-B+sqrt(D))/(2.0*A);
    X2 := (-B-sqrt(D))/(2.0*A);
END_IF

```

ภาพที่ 2.8 Structure text

5) SFC (Sequential Function) SFC จะเป็นภาษาที่รองรับการเขียนโปรแกรมที่มีโครงสร้างการทำงานเป็นแบบลำดับ (Sequence) ซึ่งส่วนประกอบของ SFC จะประกอบด้วย การปฏิบัติการย่อย (Step) และเงื่อนไขที่กำหนดให้ปฏิบัติงานตามคำสั่งย่อย (Transition) นอกจากนี้ยังสามารถยังกำหนดลักษณะการทำงานเป็นแบบ Liner, Alternative และ Parallel Step Sequence เป็นต้น [5]



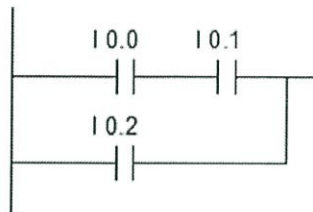
ภาพที่ 2.9 Sequential Function

## 2.2.7 คำสั่งพื้นฐานของแลตเตอร์ไดอะแกรม

### 2.2.7.1 หน้าสัมผัสปกติเปิด (Normally open contact) --- | |

หน้าสัมผัสปกติเปิด และจะปิดเมื่อค่าบิตที่ถูกเก็บที่ address มีค่าเท่ากับ “1” เมื่อหน้าสัมผัสถูกปิด กระแสจะเคลื่อนผ่านราง (Rail) และผลของการกระทำลอจิก (RLO: Result Logic Operation) เท่ากับ “1” ในทางกลับกันถ้าค่าที่ถูกเก็บที่ address มีค่าเท่ากับ “0” หน้าสัมผัสจะเปิด เมื่อหน้าสัมผัสเปิด กระแสไม่สามารถเคลื่อนผ่านราง (Rail) ได้ และผลของการกระทำลอจิก (RLO) เท่ากับ “0”

เมื่อเชื่อมต่อกันแบบอนุกรมหน้าสัมผัสปกติเปิดจะถูกเชื่อมโยงกับบิต RLO โดยลอจิกแอนด์ (AND) หากถูกเชื่อมต่อกันแบบขนาน หน้าสัมผัสปกติเปิดจะถูกเชื่อมโยงกับบิต RLO โดยลอจิกออร์ (OR)

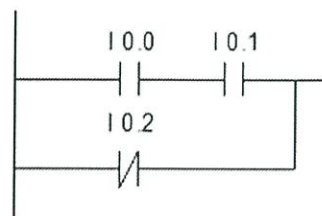


ภาพที่ 2.10 หน้าสัมผัสปกติเปิด

จากภาพที่ 2.10 กระแสเคลื่อนผ่านราง (Rail) ได้เมื่อค่าบิตที่ถูกเก็บที่อินพุต I0.0 และ I0.1 มีค่าเป็น “1” หรือค่าบิตในอินพุต I0.2 มีค่าเป็น “1”

### 2.2.7.2 หน้าสัมผัสปกติปิด (Normally closed contact) ---|/|

หน้าสัมผัสปกติปิด โดยจะปิดเมื่อค่าบิตที่ถูกเก็บที่ address มีค่าเท่ากับ “0” เมื่อหน้าสัมผัสปิด กระแสจะเคลื่อนผ่านราง (Rail) และผลของการกระทำลอจิก (RLO) เท่ากับ “1” ในทางกลับกัน ถ้าค่าบิตที่ถูกเก็บใน address มีค่าเท่ากับ “1” หน้าสัมผัสจะเปิด เมื่อหน้าสัมผัสเปิด กระแสไม่สามารถเคลื่อนผ่านราง (Rail) ได้ และผลของการกระทำลอจิก (RLO) เท่ากับ “0”

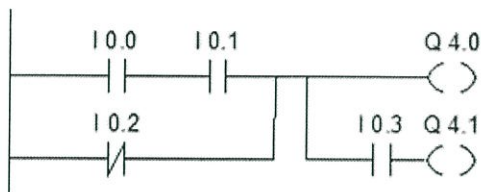


ภาพที่ 2.11 หน้าสัมผัสปกติปิด

จากภาพที่ 2.11 กระแสเคลื่อนผ่านราง (Rail) ได้เมื่อค่าบิตที่ถูกเก็บที่อินพุต I0.0 และ I0.1 มีค่าเป็น “1” หรือค่าบิตในอินพุต I0.2 มีค่าเป็น “0”

### 2.2.7.3 ขดลวดเอาต์พุต Output coil --( )

ขดลวดเอาต์พุต ถ้ามีกระแสให้กับขดลวด (RLO = 1) บิตที่ตำแหน่ง address จะมีค่าเป็น “1” ถ้าไม่มีกระแสให้กับขดลวด (RLO=0) บิตที่ตำแหน่ง จะมีค่าเป็น “0” ตำแหน่งของขดลวดเอาต์พุตอยู่ที่ขวาสุดของบรรทัดแลดเดอร์ (ladder rung) เท่านั้น และสามารถมีได้สูงสุด 16 เอาต์พุต

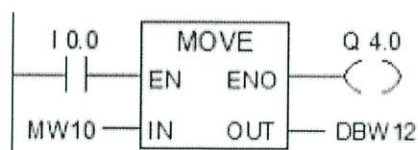


ภาพที่ 2.12 ขดลวดเอาต์พุต

จากภาพที่ 2.12 สัญญาณสถานะของเอาต์พุต Q4.0 เป็น “1” เมื่อค่าบิตในอินพุต I0.0 และ I0.1 มีค่าเป็น “1” หรือ ค่าบิตในอินพุต I0.2 มีค่าเป็น “0” และสัญญาณสถานะของเอาต์พุต Q4.1 เป็น “1” เมื่อค่าบิตในอินพุต I0.0 I0.1 และ I0.3 มีค่าเป็น “1” หรือ ค่าบิตในอินพุต I0.2 และอินพุต I0.3 มีค่าเป็น “1”

### 2.2.7.4 ย้าย (MOVE assign a value)

คำสั่งย้าย (MOVE) สามารถคัดลอกข้อมูลได้เฉพาะ BYTE, WORD หรือ DWORD หากชนิดข้อมูลเหมือนอาร์เรย์ (Array) หรือโครงสร้าง (Structure) ต้องย้ายโดยใช้ system function “BLKMOVE” SFC20



ภาพที่ 2.13 คำสั่งย้าย

จากภาพที่ 2.13 คำสั่งย้ายค่า ถูกกระตุ้น โดยอินพุต EN ค่าที่ระบุที่อินพุต IN ถูกคัดลอกไปที่ address ที่เอาต์พุต(OUT) และขา ENO มีสถานะลอจิกเหมือน EN คำสั่งย้ายจะใช้งานได้ ถ้าค่าบิตในอินพุต I0.0 มีค่าเป็น “1” ข้อมูลจาก MW10 จะถูกคัดลอกไปยัง DBW12 และค่าบิตใน Q4.0 มีค่าเป็น “1” [7]

## 2.3 ความรู้พื้นฐานเกี่ยวกับระบบสกาตา

SCADA ย่อมาจากคำว่า Supervisory Control And Data Acquisition เป็นระบบตรวจสอบและวิเคราะห์ข้อมูลแบบทันที (Real-time) ใช้ในการตรวจสอบสถานะตลอดจนควบคุมการทำงานของระบบควบคุมในอุตสาหกรรมและงานวิศวกรรมต่าง ๆ เช่น งานด้านโทรคมนาคมสื่อสาร การประปา การบำบัดน้ำเสีย โทรมาตร การจัดการด้านพลังงาน อุตสาหกรรมการกลั่นน้ำมันและก๊าซ อุตสาหกรรมเคมี อุตสาหกรรมประกอบรถยนต์ การขนส่ง กระบวนการนิวเคลียร์ในโรงไฟฟ้า เป็นต้น ตัวอย่างการใช้งาน เช่น การใช้สกาตาตรวจสอบข้อมูลการรั่วไหลของของเหลวที่เกิดขึ้นในท่อขนส่งจากอุปกรณ์ตรวจจับแล้วส่งสัญญาณแจ้งเตือนให้ผู้ใช้งานทราบ โดยส่งข้อมูลสู่ส่วนกลางของระบบสกาตา เป็นต้น

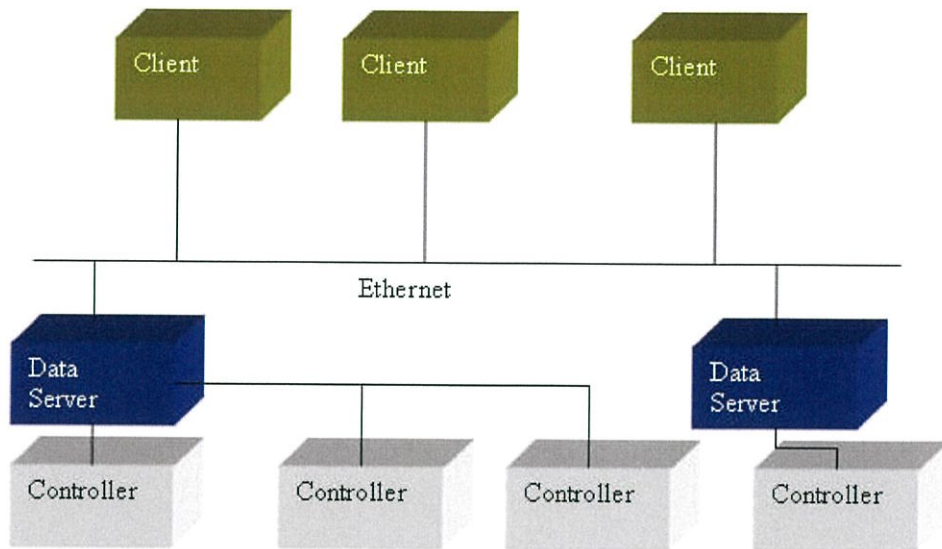
นอกจากนั้นสกาตาอาจทำหน้าที่คำนวณและประมวลผลข้อมูลที่ได้จากฮาร์ดแวร์ต่าง ๆ เช่น PLC, Controller, DCS, RTU แล้วแสดงข้อมูลทางหน้าจอ หรือส่งสัญญาณควบคุมฮาร์ดแวร์ดังกล่าว เช่น หากอุณหภูมิของอุปกรณ์สูงเกินพิกัดให้ทำการปิดอุปกรณ์นั้น โดยสั่งงานผ่าน PLC หรือ Controller ที่ติดต่อยู่ ทั้งนี้สกาตาสามารถเก็บรวบรวมข้อมูลที่ได้จากระบบควบคุมทั้งหมดไว้ในฐานข้อมูลเพื่อให้ผู้ใช้หรือโปรแกรมอื่นๆ สามารถนำไปใช้งานได้ สกาตานั้นเข้าไปมีส่วนในงานควบคุมทั้งเล็กและใหญ่ที่ต้องการแสดงผล แลกเปลี่ยนข้อมูล หรือควบคุมระบบต่าง ๆ จากส่วนกลาง เพื่อการทำงานของระบบรวมที่สัมพันธ์กัน มองเห็นภาพรวมได้อย่างชัดเจนและมีความรวดเร็วต่อเหตุการณ์ต่าง ๆ ที่เกิดขึ้น

ระบบสกาตาในปัจจุบันมีความสามารถในการสื่อสาร ควบคุม และประมวลผลข้อมูลจาก I/O ของอุปกรณ์เช่น PLC, DCS, RTU ได้มากถึงหนึ่งแสน I/O และได้รับการพัฒนาให้มีความสามารถรองรับความต้องการใหม่ ๆ ของผู้ใช้งาน

### 2.3.1 โครงสร้างของสกาตา (SCADA Architecture)

#### 2.3.1.1 โครงสร้างด้านฮาร์ดแวร์ (Hardware Architecture)

สกาตาแบ่งตามโครงสร้างฮาร์ดแวร์ได้สองระดับคือ Client และ Data Server โดยที่ Client คือคอมพิวเตอร์ที่รับและส่งข้อมูลไปยัง Data Server โดยฝั่ง Client นี้จะแสดงผลการทำงานของระบบควบคุมเช่น แสดงเป็นกราฟิก กราฟแบบต่อเนื่อง หรือระบบแจ้งเตือนเมื่อเกิดเหตุการณ์ฉุกเฉินหรือต้องการแจ้งเตือน เป็นต้น ฝั่ง Client สามารถสั่งงานควบคุมไปยัง Data Server เพื่อส่งสัญญาณไปยัง PLC, DCS หรือ Controller อีกทอดหนึ่ง โดย Controller จะติดต่อกับอุปกรณ์ Field Instrument ต่าง ๆ เช่น เซนเซอร์ รีเลย์ เป็นต้นเพื่อนำสัญญาณมาให้กับ Data Server โดย Data Server จะทำหน้าที่ติดต่อกับ PLC, DCS, Controller หรือ RTU ต่าง ๆ เพื่อรับสัญญาณและส่งสัญญาณไปยัง Client และรับการร้องขอจาก Client เพื่อควบคุมอุปกรณ์ PLC และ Controller ต่าง ๆ การเชื่อมต่อระหว่าง Client และ Data Server ส่วนใหญ่ติดต่อกันผ่านระบบเครือข่าย Ethernet ดังภาพที่ 2.14



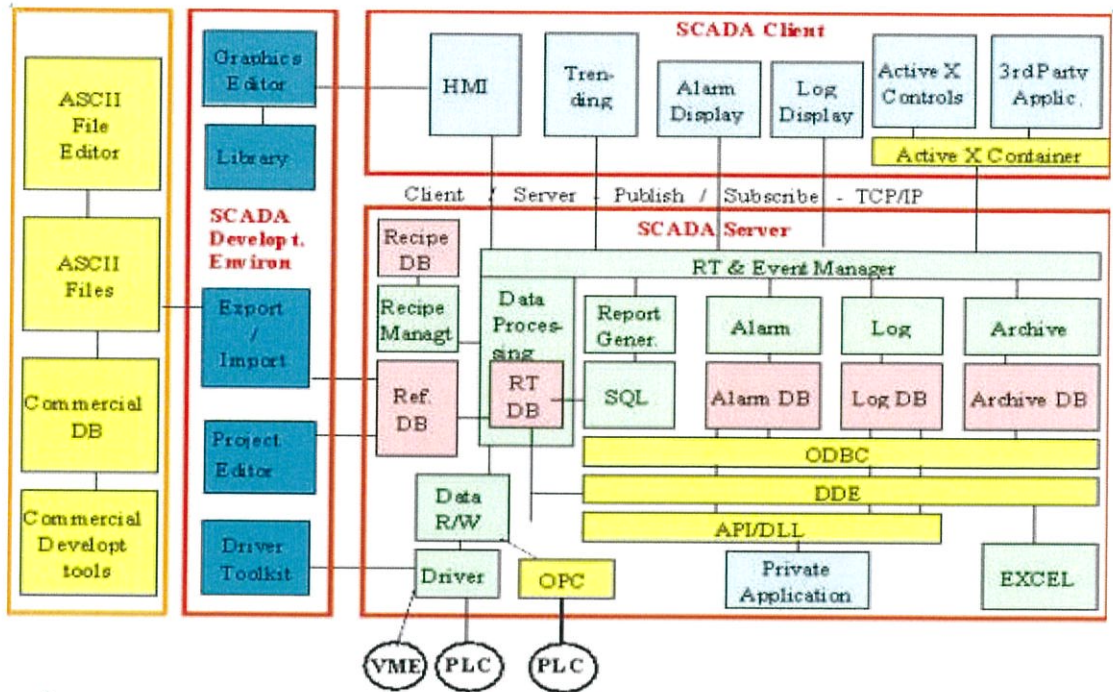
ภาพที่ 2.14 โครงสร้างแบบฮาร์ดแวร์ของระบบสกาดา

### 2.3.1.2 โครงสร้างด้านซอฟต์แวร์ (Software Architecture)

โครงสร้างด้านซอฟต์แวร์ของระบบสกาดานั้นมีข้อที่ต้องทราบ คือ สกาดาใช้เทคโนโลยีในการสื่อสารกับฮาร์ดแวร์ (เช่น PLC, DCS) ต่างกันไปตามผู้ผลิต เช่น การใช้ Driver เฉพาะของผู้ผลิตสกาดาเพื่อสื่อสารกับ PLC, DCS เป็นต้น ซึ่งในปัจจุบันมีการกำหนดมาตรฐานกลางคือ OPC (OLE for Process Control) ขึ้นมาเพื่อยุติปัญหาการใช้เทคโนโลยีเฉพาะด้านในการสื่อสาร นอกจากนั้นยังมีความสามารถในการบริการข้อมูลให้กับ Client ที่รวดเร็วและมีเสถียรภาพ จะพบว่าในส่วนของ SCADA Server การติดต่อกับ PLC หรือ Controller นั้น ทำได้ทั้งผ่าน Driver หรือ OPC โดยที่ OPC และ Driver สามารถรับคำสั่งแบบอ่านหรือเขียน (Read/Write) เพื่ออ่านข้อมูลจาก PLC หรือ เขียนข้อมูลเพื่อสั่งงานไปยัง PLC ได้

SCADA Server จะทำหน้าที่จัดการข้อมูลแบบทันที RTDB (Real Time Data Base) ที่ได้จาก PLC แล้วส่งให้กับ SCADA Client โดยที่ SCADA Server บางประเภทจะติดต่อกับ SCADA Client ผ่าน DDE Server ซึ่งทำให้สามารถนำข้อมูลจาก PLC เข้าสู่โปรแกรมเช่น MS Excel หรือ โปรแกรม Client อื่นๆ ที่ติดต่อกับ DDE Server ได้ สกาดา บางตัวจะออกแบบให้ SCADA Server ทำหน้าที่ตรวจจับ Alarm และเก็บไว้ใน Alarm DB หรือเก็บข้อมูลที่เป็น Historian ไว้ใน Log DB เป็นต้น เพื่อส่งให้ Alarm Display และ Log Display ทางฝั่ง SCADA Client ต่อไป

สำหรับส่วน Development Environment นั้นจะขึ้นอยู่กับกรออกแบบของสกาดา ซอฟต์แวร์นั้น ๆ ซึ่งโดยทั่วไปก็จะมีเครื่องมือในการสร้างและจัดการกราฟิก (Graphic Editor) เครื่องมือในการจัดการโปรเจกต์ที่สร้างขึ้น (Project Editor) มีเครื่องมือในการนำเข้าและส่งออกไฟล์ข้อความ (Text file) ที่เก็บค่าคอนฟิกูเรชันของการติดต่อกับ Driver หรือ OPC Serverไว้ ดังภาพที่ 2.15



ภาพที่ 2.15 โครงสร้างด้านซอฟต์แวร์ของสกาตา

### 2.3.1.3 โครงสร้างด้านการสื่อสาร (Communications)

การสื่อสารระหว่าง Client-Server จะสื่อสารผ่านโปรโตคอลโดยทั่วไป เช่น TCP/IP โดย Client จะติดต่อกับพารามิเตอร์หรือ Tag ภายใน Server ที่บริการข้อมูลด้วยรูปแบบที่แตกต่างกันไปตามผู้ผลิต เช่นมีการส่งค่าจาก Server เมื่อค่าของอินพุตเอาต์พุต PLC มีการเปลี่ยนแปลง เป็นต้น

การสื่อสารกับอุปกรณ์นั้น Server จะทำการตรวจสอบค่าจากอุปกรณ์ตามช่วงเวลาที่ใช้กำหนดไว้ (Defined polling rate) โดยอาจจะต่างกันไปตามพารามิเตอร์ประเภทต่าง ๆ โดยตัวควบคุม (Controller) จะส่งค่าพารามิเตอร์ตามที่ถูกร้องขอให้กับ Data Server พร้อมค่าเวลาขณะนั้น (Time Stamp) การสื่อสารกับอุปกรณ์ของ Data Server นั้นอาจเป็นการสื่อสารแบบ Modbus, Profibus, CAN bus เป็นต้น ขึ้นอยู่กับมาตรฐานการสื่อสารของอุปกรณ์นั้น ๆ ว่าเป็นแบบใด ในปัจจุบันมีการสร้าง OPC Server ที่สนับสนุนการติดต่อด้วยมาตรฐานต่างๆเพิ่มขึ้นมากมายจนครอบคลุมอุปกรณ์ทุกประเภท และมีการพัฒนาให้ทั่วถึงไปยังอุปกรณ์ใหม่ ๆ อย่างต่อเนื่อง

### 2.3.1.4 โครงสร้างการติดต่อ (Interface)

การติดต่อระหว่าง Data Server กับอุปกรณ์หรือระหว่าง Data Server และ Data Server และกับ Client นั้น มีการผลิตเป็น Driver ออกมามากมายตามเทคนิคเฉพาะของแต่ละผู้ผลิต ต่อมาจึงมีการกำหนดมาตรฐานของติดต่อขึ้นมาเป็น OPC (OLE for Process Control) ซึ่งมีความรวดเร็วในการสื่อสารและบริการข้อมูลโดยมีการจัดตั้ง OPC Foundation ขึ้นเป็นองค์กรหลักในการกำหนดมาตรฐานและถ่ายทอดเทคโนโลยีให้แก่สมาชิก OPC จึงเป็นมาตรฐานกลางที่เปิดกว้างมากที่สุด

การติดต่อกับฐานข้อมูลภายนอกของซอฟต์แวร์สกาดานั้น มีการสร้างให้สามารถติดต่อได้ผ่าน ODBC (Open Data Base Connectivity), OLEDB (Linking and Embedding Data Base), DDE (Dynamic Data Exchange) เป็นต้น เพื่อให้สามารถแลกเปลี่ยนข้อมูลหรือทำการเก็บข้อมูลไว้ในฐานข้อมูลรูปแบบต่าง ๆ ในปัจจุบันมีการพัฒนาให้สามารถติดต่อกับโปรแกรม ERP ต่าง ๆ เช่น SAP เป็นต้นได้ด้วย

### 2.3.1.5 โครงสร้างความสามารถในการขยายระบบ (Scalability)

Scalability คือความสามารถในการรองรับและต่อขยายระบบสกาดากับส่วนต่าง ๆ เช่น I/O ของอุปกรณ์ Controller และจำนวนเครื่อง SCADA Client ที่เพิ่มขึ้น หรือการต่อพ่วงกับระบบสกาดา ของยี่ห้ออื่น ๆ เป็นต้น ถ้าหาก Data Server เป็นแบบ Driver ที่สร้างด้วยเทคโนโลยีเฉพาะในการติดต่อกับอุปกรณ์ ก็เป็นเรื่องลำบากในการต่อขยาย เพราะ Driver บางประเภทสามารถติดต่อได้เฉพาะซอฟต์แวร์ของสกาดาบางยี่ห้อเท่านั้น ปัญหานี้เป็นที่วิพากษ์วิจารณ์กันอย่างกว้างขวาง ซึ่งปัจจุบันได้หันมาใช้มาตรฐานกลางคือ OPC เพื่อแก้ไขปัญหานี้

### 2.3.1.6 โครงสร้างการสำรองระบบ (Redundancy)

ซอฟต์แวร์สกาดาส่วนใหญ่มีความสามารถในการทำสำรองระบบของ Data Server โดยที่เมื่อ Data Server เกิดความขัดข้องก็จะสั่งงานให้ Data Server อีกตัวหนึ่งทำงานแทนที่ โดยจะมีการกำหนดคอนฟิกเรชั่นไว้ที่ Clientว่าจะให้เลือกติดต่อกับ Data Server ตัวใดเมื่อเกิดความขัดข้องเกิดขึ้น

ในบางครั้งโมดูลที่ทำหน้าที่จัดการด้านการสำรองระบบ (Redundancy) นี้อาจทำหน้าที่อีกประการหนึ่งคือเป็นจุดพักข้อมูลที่ได้รับมาจาก Data Server เพื่อนำไปส่งให้กับ Client ต่าง ๆ เพราะในกรณีที่มี Client จำนวนมากติดต่อกับ Data Server ตัวเดียวนั้นอาจมีความล่าช้าในการบริการข้อมูลของ Data Server เพราะต้องให้บริการข้อมูล Client ให้ครบจำนวนก่อนที่จะไปรับข้อมูลใหม่จากอุปกรณ์มาได้ ดังนั้นโมดูลที่ทำหน้าที่สำรองระบบ (Redundant) จึงทำหน้าที่เป็นจุดรับข้อมูลแล้วช่วยส่งต่อให้ Client ต่าง ๆ อีกทอดหนึ่ง Data Server จะได้ทำหน้าที่บริการข้อมูลให้แก่โหนดเพียงจุดเดียว จึงมีความรวดเร็วในการบริการข้อมูล

## 2.3.2 หน้าที่การทำงาน (Functionality)

การเข้าถึงพารามิเตอร์ของอุปกรณ์ คือ ความสามารถในการเข้าถึงกลุ่มของพารามิเตอร์ในอุปกรณ์เช่น I/O ของ PLC เป็นต้น ความสามารถของ Data Server ในการกำหนดว่าพารามิเตอร์ใด อ่านได้อย่างเดียว เขียนได้อย่างเดียว หรือทั้งอ่านทั้งเขียน เป็นต้น

### 2.3.2.1 ระบบแสดงผลแบบ MMI (Man Machine Interface)

คือความสามารถในการแสดงผลการทำงานของอุปกรณ์ในรูปแบบ กราฟิก ข้อความ สัญลักษณ์ แผนภาพ เป็นต้น โดยสามารถเชื่อมโยงลักษณะการเปลี่ยนแปลงของกราฟิกเหล่านี้

กับพารามิเตอร์จาก Data Server ได้ ความสามารถในการสั่งงานผ่านระบบกราฟิกเช่น การปิด/เปิด สวิตช์ บนจอมอนิเตอร์ส่งผลไปยัง I/O ของ PLC เป็นต้น

ความสามารถในการจัดการกราฟิกเช่น การย่อ ขยาย การกำหนดการเคลื่อนไหวแบบต่าง ๆ เช่น การหมุน การเคลื่อนที่แบบซิกแซกตามสัญญาณของ Data Server การแสดงผลสัญญาณในรูปแบบมิเตอร์และเกจวัดแบบต่างๆ การนำเข้ากราฟิกประเภทต่างๆ เป็นต้น เหล่านี้เป็นข้อเปรียบเทียบความสามารถของซอฟต์แวร์สกาตาทั้งสิ้น

#### 2.3.2.2 ระบบแสดงกราฟสัญญาณแบบต่อเนื่อง (Trending)

กราฟสัญญาณแบบต่อเนื่อง เป็นความสามารถในการพล็อตกราฟต่อเนื่องกันไปบนจอภาพเพื่อแสดงค่าสัญญาณจาก Data Server โดยอาจจะสามารถพล็อตสัญญาณได้หลายสัญญาณ เช่น 8 – 24 สัญญาณ พร้อมกันในหน้าต่างเดียว เพื่อให้สามารถเปรียบเทียบสัญญาณที่พล็อตได้ และไม่จำกัดว่าจะสร้างหน้าต่างพล็อตจำนวนเท่าใด

กราฟสัญญาณแบบต่อเนื่องอาจมีความสามารถในการ ซุ่มสัญญาณที่พล็อต และหยุดการพล็อตเพื่อเลื่อนดูค่าที่พล็อตในแต่ละช่วงเวลาได้ด้วยตัวของผู้ใช้งานเอง นอกจากนี้การพล็อตอาจสามารถเลือกได้ว่าจะให้เป็นการพล็อตแบบใดเช่น Time plot, Logarithmic plot, Strip Chart, Bar Chart, Circular, X-Y plot เป็นต้น นอกจากนี้บางผู้ผลิตยังสามารถนำค่า Historian หรือข้อมูลสัญญาณที่เก็บไว้ในฐานข้อมูลออกมาพล็อต ได้อีกด้วย โดย Trending Module นี้อาจเป็นแบบ ActiveX Control คือสามารถนำไปใช้งานในแอปพลิเคชันอื่นที่สนับสนุนการนำเข้า ActiveX ได้

#### 2.3.2.3 ระบบแจ้งเตือน (Alarm)

ซอฟต์แวร์สกาตาส่วนใหญ่มีระบบแจ้งเตือนโดย Alarm Display จะรับสัญญาณมาจาก Alarm DB ในฝั่ง SCADA Server โดย Alarm DB สามารถที่จะทำการกำหนดคอนฟิกูเรชันว่าจะนำสัญญาณตัวใดมาเป็นตัวพารามิเตอร์ในการแจ้งเตือนบ้าง และมีการแบ่งระดับของ Priority, Limit อย่างไร เป็นต้น

ระบบแจ้งเตือนยังสามารถที่จะเก็บข้อมูลการแจ้งเตือนไว้ในฐานข้อมูลประเภทต่างๆ ได้เช่น MS SQL Server, MS Access, Oracle, MS Excel เป็นต้น และบางยี่ห้อสามารถแสดงออกมาเป็นรายงานในรูปแบบตารางหรือ แผนภูมิได้อีกด้วย

#### 2.3.2.4 การทำงานแบบอัตโนมัติ

เป็นความสามารถที่สกาตาทำหน้าที่ต่าง ๆ ตามที่กำหนด เช่น ส่งอีเมลล์ แสดงข้อความแบบ Instance Message บนหน้าจอ เปิดไปยังหน้าจออื่น ๆ เก็บข้อมูลลงฐานข้อมูล เปิดโปรแกรม หรือรันคำสั่งสคริปต์ เป็นต้น ตามสัญญาณที่ได้รับจาก Data Server และข้อกำหนดที่สร้างขึ้น

### 2.3.3 การสร้างและพัฒนา (Application Development)

#### 2.3.3.1 การกำหนดคอนฟิกูเรชัน

การกำหนดคอนฟิกูเรชัน ขั้นแรกต้องมีการกำหนดว่าจะติดต่อกับพารามิเตอร์ หรือ Tag ใดบ้างจาก Data Server ดังนั้นจะต้องทำการ Define หรือสร้าง Tag ที่ Data Server ก่อนว่า Tag แต่ละตัวหมายถึง Address ที่เท่าใดของอุปกรณ์ (PLC, DCS, RTU, Controller ต่างๆ) โดยทั่วไปสามารถทำการนำเข้าคอนฟิกูเรชันไฟล์ที่สร้างไว้ก่อนเข้ามาได้ และสามารถ Export ไปยัง Data Server อื่น ๆ ได้ จากนั้นโปรแกรมย่อยอื่น ๆ ของ SCADA Software ฝั่ง client จึงทำคอนฟิกูเรชันตามหน้าที่การทำงานของตนเอง เช่น โมดูลที่มีหน้าที่แสดงผลกราฟก็ต้องกำหนดว่ากราฟนั้น ๆ จะเชื่อมโยงกับ Tag ใดจาก Data Server ส่วนโมดูลที่มีหน้าที่แจ้งเตือนก็ต้องทำคอนฟิกูเรชันว่าจะนำ Tag ใด มาเป็นสัญญาณแจ้งเตือน และกำหนดระดับสัญญาณ Limit เป็นต้น

#### 2.3.1.2 เครื่องมือในการพัฒนา (Development Tool)

เครื่องมือในการสร้างและพัฒนาระบบสกาดาโดยทั่วไปจะประกอบด้วย

- 1) เครื่องมือในการสร้างระบบกราฟิกที่ประกอบด้วยเครื่องมือวาดภาพ เครื่องมือกำหนดเอฟเฟ็คพิเศษต่าง ๆ โกลบารีของกราฟิกสำเร็จรูปในอุตสาหกรรมด้านต่าง ๆ
- 2) เครื่องมือในการสร้างกราฟสัญญาณแบบต่อเนื่อง (Trending)
- 3) เครื่องมือในการสร้างระบบแจ้งเตือน (Alarm)
- 4) เครื่องมือในการกำหนดการติดต่อกับฐานข้อมูลเพื่อทำการเก็บรวบรวมข้อมูลของ Trending และ Alarm ลงไว้ในฐานข้อมูล
- 5) เครื่องมือในการช่วยสร้าง Script เช่น Java script, VB Script
- 6) เครื่องมือจัดการด้านความปลอดภัย การแบ่งระดับ User และขอบเขตการใช้งาน
- 7) เครื่องมือในการสร้าง Web application เพื่อให้สามารถควบคุมและตรวจสอบระบบควบคุมผ่าน Web browser ได้<sup>[4]</sup>

### 2.4 ระบบเครือข่ายอีเทอร์เน็ต

อีเทอร์เน็ต (Ethernet) เป็นเครือข่ายคอมพิวเตอร์ที่พัฒนามาจากโครงสร้างการเชื่อมต่อแบบสายสัญญาณร่วมที่เรียกว่า บัส (Bus) โดยใช้สายสัญญาณแบบแกนร่วม คือ สายโคแอกเซียล (Coaxial Cable) เป็นตัวเชื่อม สำหรับระบบบัส เป็นระบบเทคโนโลยีที่คอมพิวเตอร์ทุกเครื่องเชื่อมโยงเข้ากับสายสัญญาณเส้นเดียวกัน คือ เมื่อมีผู้ต้องการส่งข้อมูล ก็ส่งข้อมูลได้เลย แต่เนื่องจากไม่มีวิธีการค้นหาเส้นทางที่ส่งว่างหรือเปล่า จึงไม่ทราบว่ามีอุปกรณ์ใดหรือคอมพิวเตอร์ เครื่องใดที่ส่งข้อมูลมาในช่วงเวลาเดียวกัน จะทำให้เกิดการชนกันขึ้นและเกิดการสูญหายของข้อมูล ผู้ส่งต้องส่งข้อมูล ไปยังปลายทางอีกครั้งหนึ่ง ทำให้เสียเวลามาก จึงมีการพัฒนาระบบการรับส่งข้อมูลผ่านอุปกรณ์กลางที่เรียกว่า ฮับ (Hub) และเรียกระบบใหม่นี้ว่า เทนเบสที (10 base t) โดยใช้สายสัญญาณที่มีขนาดเล็กลงและราคาถูกซึ่งเรียกว่า

สายคู่บิดเกลียวชนิดไม่หุ้มฉนวน (Unshielded twisted pair : UTP) ทำให้การเชื่อมต่อมีลักษณะแบบดาว

วิธีการเชื่อมแบบนี้จะมีจุดศูนย์กลางอยู่ที่ฮับ ใช้สายสัญญาณไปยังอุปกรณ์หรือคอมพิวเตอร์อื่น ๆ จุดเด่นของดาวตัวนี้ จะอยู่ที่ เมื่อมีการส่งข้อมูล จะมีการตรวจสอบความผิดพลาดว่า อุปกรณ์ใดจะส่งข้อมูลมาบ้างและจะมีการสับวิตช์ให้ส่ง ได้หรือไม่ แต่เมื่อมีฮับเป็นตัวแบกภาระทั้งหมด ก็มีจุดอ่อนได้คือ ถ้าฮับเกิดเป็นอะไรขึ้นมา อุปกรณ์ต่อพ่วงอื่น ๆ หรือคอมพิวเตอร์ก็ไม่สามารถเชื่อมต่อกันได้อีก

ภายในฮับมีลักษณะเป็นบัสที่เชื่อมสายทุกเส้นเข้าด้วยกัน ดังนั้นการใช้ฮับและบัสจะมีระบบการส่งข้อมูลแบบ เดียวกัน และมีการพัฒนาเป็นมาตรฐาน กำหนดชื่อมาตรฐานนี้ว่า 802.3 ความเร็วในการส่ง กำหนดไว้ที่ 10 ล้านบิตต่อ วินาที และกำลังมีมาตรฐานใหม่ให้สามารถรับส่งสัญญาณได้ถึง 100 ล้านบิตต่อ วินาที [6]

## 2.5 การสื่อสารระหว่างอุปกรณ์ HMI และ SIMATIC S7

เมื่อใช้งาน SIMATIC S7 อุปกรณ์ HMI สามารถเชื่อมต่อได้หลากหลายโครงข่าย ขึ้นกับโมดูลที่ใช้ งาน ซึ่งในรายงานฉบับนี้ใช้ PLC SIMATIC S7-300

ตารางที่ 2.2 โครงข่ายการสื่อสารระหว่างอุปกรณ์ HMI และ SIMATIC S7

PLC	Modules	Network	Profile
SIMATIC S7-200	CPU	PPI	PPI
		MPI	MPI
	CPU with PROFIBUS DP interface or PROFIBUS CP	PROFIBUS	PROFIBUS DP standard, Universal
	CP 234-1	Ethernet	TCP/IP
SIMATIC S7-300/400	CPU FM with communication function	MPI	MPI
	CPU with PROFIBUS DP interface or PROFIBUS CP	PROFIBUS	PROFIBUS DP standard, Universal
SIMATIC S7-300	CP 343-1	Ethernet	TCP/IP
SIMATIC S7-400	CP 443-1	Ethernet	TCP/IP

## 2.6 ซอฟต์แวร์ที่เกี่ยวข้อง

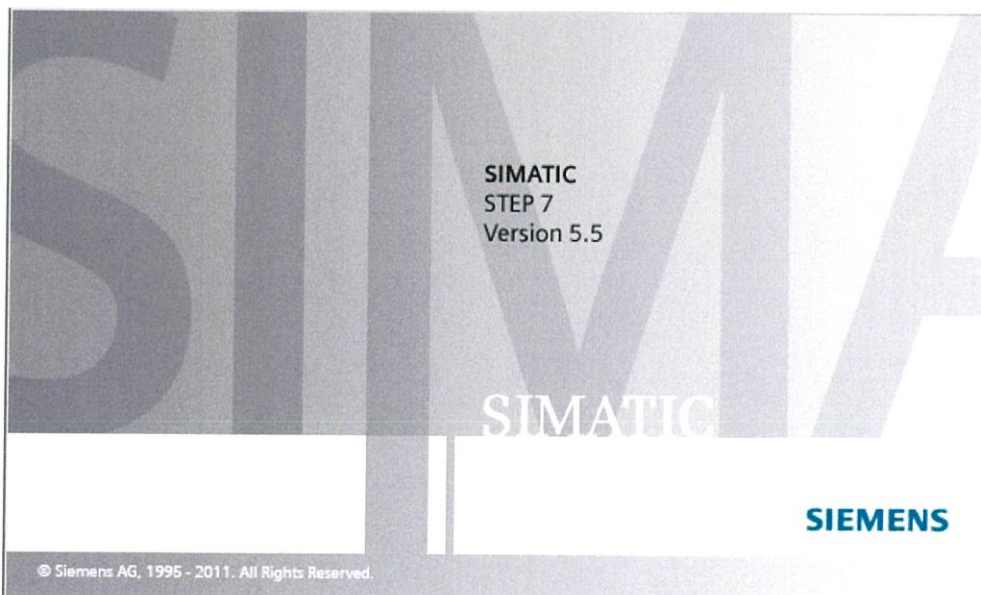
### 2.6.1 SIMATIC STEP 7

STEP 7 คือ ชุดซอฟต์แวร์มาตรฐานที่ใช้ในการกำหนดค่าและการเขียนโปรแกรมควบคุมตรรกะ(Programmable Logic Controllers) เป็นส่วนหนึ่งของซอฟต์แวร์อุตสาหกรรม โดยมีรุ่นดังต่อไปนี้

STEP 7 Micro/DOS และ STEP 7 Micro/Win สำหรับใช้งานได้ง่ายแบบสแตนด์ออลโปรแกรมประยุกต์บน SIMATIC S7-200

STEP 7 สำหรับการใช้งานบน SIMATIC S7-300/S7-400, SIMATIC M7-300/M7-400, และ SIMATIC C7 ที่มีฟังก์ชันที่กว้างขึ้น ดังนี้

- 1) สามารถขยายเป็นตัวเลือกโดยผลิตภัณฑ์ซอฟต์แวร์ใน SIMATIC
- 2) ซอฟต์แวร์อุตสาหกรรม
- 3) โอกาสในการกำหนดพารามิเตอร์ให้กับโมดูลฟังก์ชันและโปรเซสเซอร์สื่อสาร
- 4) การสื่อสารข้อมูลทั่วโลก
- 5) การถ่ายโอนข้อมูลที่เป็นเหตุการณ์โดยใช้บล็อกฟังก์ชันการติดต่อสื่อสาร
- 6) การกำหนดค่าการเชื่อมต่อ



ภาพที่ 2.16 ซอฟต์แวร์ SIMATIC STEP 7

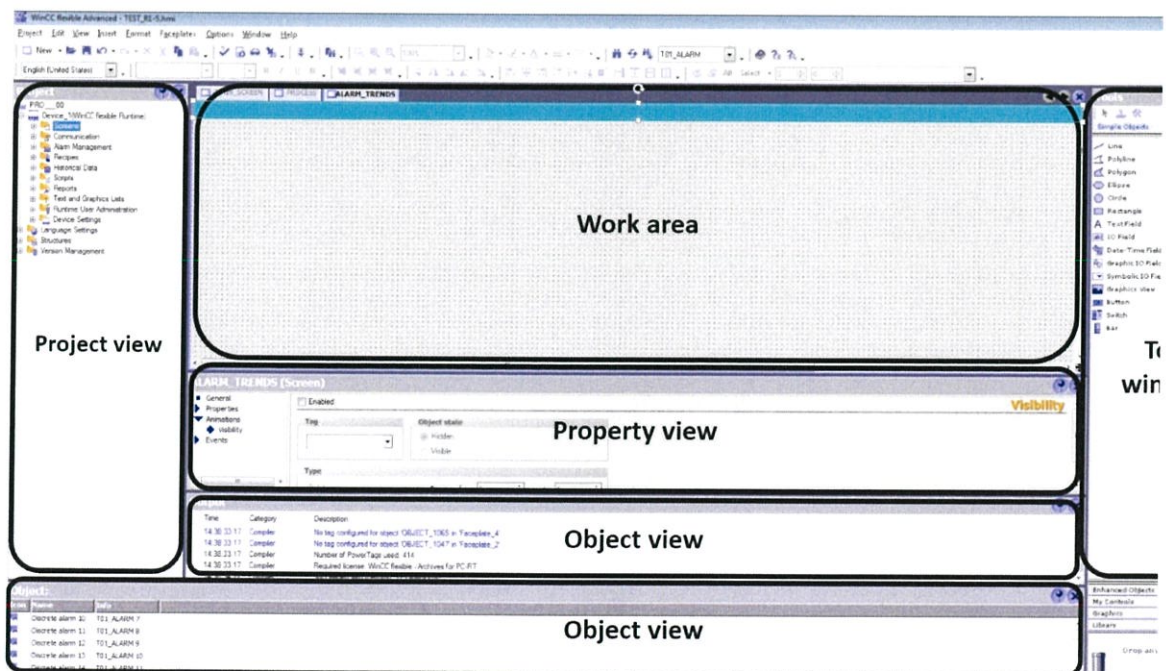
### 2.6.2 WinCC flexible 2008

WinCC flexible เป็นซอฟต์แวร์ที่มุ่งเน้นการทำงานแบบอัตโนมัติของเครื่องจักรด้วยการใช้งานที่สะดวกและมีประสิทธิภาพทางด้านวิศวกรรมสูง ใช้สำหรับกำหนดค่าการติดต่อกับผู้ใช้งานสำหรับการควบคุมเครื่องจักรหรือระบบ โดย ในโปรแกรมสามารถใช้งานกับอุปกรณ์ HMI ได้มากที่สุด 5 โปรแกรมใน WinCC flexible ประกอบด้วยข้อมูลกำหนดค่าทุกค่าสำหรับโรงงานหรือ อุปกรณ์ HMI ประกอบด้วย

- 1) หน้าจอกระบวนการ (Process screen) สำหรับแสดงกระบวนการ
- 2) แท็ก (Tag) สำหรับส่งข้อมูลระหว่าง PLC และอุปกรณ์ HMI
- 3) สัญญาณเตือน (Alarm) สำหรับแสดงสถานะการดำเนินการ
- 4) ล็อก (Log) สำหรับเก็บค่ากระบวนการและสัญญาณเตือน



ภาพที่ 2.17 ซอฟต์แวร์ WinCC flexible 2008



ภาพที่ 2.18 องค์ประกอบของ WinCC flexible

ซอฟต์แวร์ WinCC flexible มีองค์ประกอบดังภาพที่ 2.18

พื้นที่ทำงาน (Work area) วัตถุของโปรเจกต์ถูกแก้ไขในพื้นที่ทำงาน องค์ประกอบทั้งหมดของ WinCC Flexible ถูกจัดเรียงอยู่ในพื้นที่ทำงาน ยกเว้นพื้นที่ทำงานสามารถจัดระเบียบและกำหนดค่าตัวอย่างเช่น ย้ายหรือซ่อน องค์ประกอบใดก็ได้เพื่อให้ตรงกับความต้องการ

มุมมองโปรเจกต์ (Project view) องค์ประกอบทุกส่วนในโปรเจกต์จะปรากฏในโครงสร้างแบบต้นไม้ในมุมมองโปรเจกต์ สามารถสร้างโฟลเดอร์ให้เป็นองค์ประกอบย่อยของแต่ละตัว ทำให้สามารถจัดเก็บวัตถุได้อย่างเป็นระบบ นอกจากนี้การเข้าถึงโดยตรงไปยังวัตถุที่มีการกำหนดค่า ที่มีอยู่สำหรับ Screen, Recipes, Script และ Protocol ในหน้าต่างโปรเจกต์สามารถเข้าถึงการตั้งค่าอุปกรณ์ HMI, ตั้งค่าภาษา และการจัดการเวอร์ชัน

มุมมองคุณสมบัติ (Property view) มุมมองคุณสมบัติถูกใช้เพื่อแก้ไขคุณสมบัติของวัตถุ เช่นสีของวัตถุหน้าจอ โดยมุมมองคุณสมบัติใช้ได้เฉพาะการแก้ไขที่เฉพาะเจาะจง

หน้าต่างเครื่องมือ (Tool window) หน้าต่างเครื่องมือประกอบด้วยตัวเลือกวัตถุที่สามารถเพิ่มได้ลงบนหน้าจอ เช่น วัตถุ รูปภาพ หรือองค์ประกอบการควบคุม นอกจากนี้หน้าต่างเครื่องมือยังมีไลบรารีที่ประกอบไปด้วยวัตถุแม่แบบและรูปแบบของ Faceplate ที่สร้างขึ้น ไลบรารีเป็นฐานข้อมูลกลางสำหรับจัดเก็บวัตถุที่ใช้บ่อย เช่นหน้าจอวัตถุและแท็ก

มุมมองผลลัพธ์ (Output view) มุมมองผลลัพธ์แสดงระบบสัญญาณเตือนที่เกิดขึ้น ตัวอย่าง เช่นในการทดสอบการทำงานของโปรเจกต์

มุมมองวัตถุ (Object view) มุมมองวัตถุแสดงให้เห็นถึงองค์ประกอบทั้งหมดของพื้นที่ที่ถูกเลือกจากมุมมองโปรเจกต์ [8]

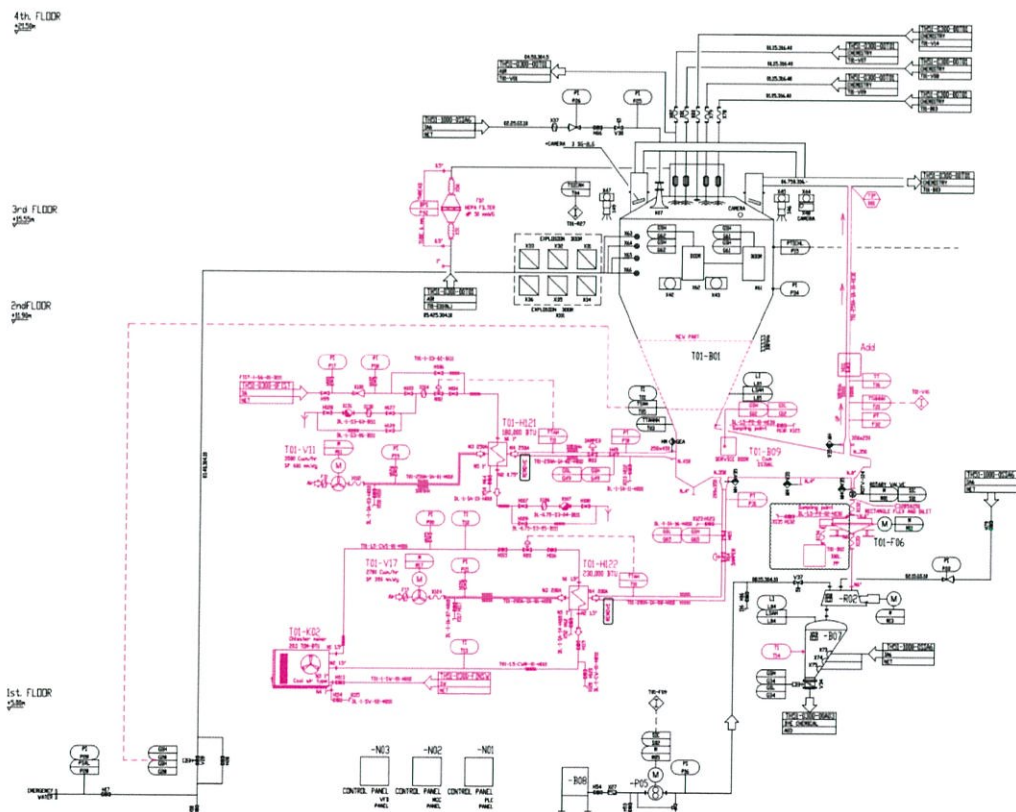
## บทที่ 3 การออกแบบโปรแกรมควบคุม

### 3.1 ออกแบบโปรแกรมควบคุมด้วยพีแอลซี

การปรับปรุงกระบวนการอบแห้งแบบพ่นฝอย (Spray drying) โดยเพิ่มส่วนของโปรแกรมควบคุมด้วย PLC ตามผังงาน (Flowchart) และเพิ่มเติมกราฟิกโดยอ้างอิงตามแผนภาพ P&ID

#### 3.1.1 ศึกษากระบวนการ

ศึกษาลำดับการทำงานของเครื่องอบแห้งแบบพ่นฝอยจากผังงาน (Flowchart) ประกอบกับแผนภาพ P&ID (Piping And Instrumentation Diagram) ที่แสดงอุปกรณ์และเครื่องมือวัดที่ใช้ในระบบเดิมและระบบที่เพิ่มใหม่ ตามรายการแอดเดรสของอินพุตและเอาต์พุต (I/O list) ที่เกี่ยวข้อง การแจ้งเตือน (Alarm) รวมถึงข้อมูลจำเพาะของอุปกรณ์และเครื่องมือวัดแต่ละตัว (Specification)

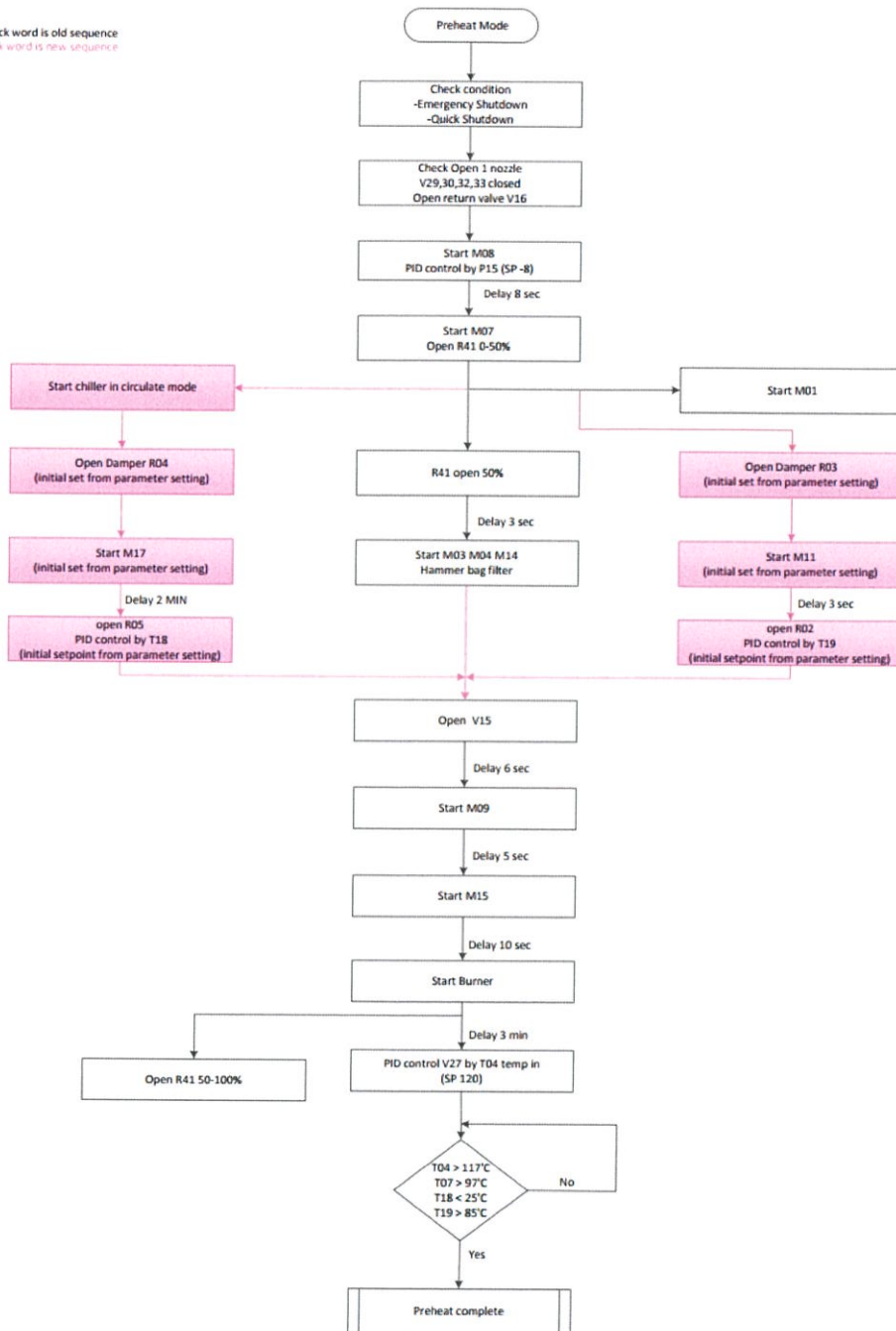


ภาพที่ 3.1 แผนภาพ P&ID ของกระบวนการอบแห้งแบบพ่นฝอย

จากภาพที่ 3.1 เป็นแผนภาพที่แสดงระบบท่อ อุปกรณ์ต่าง ๆ และตำแหน่งการติดตั้ง โดยสีดำหมายถึงอุปกรณ์ที่มีอยู่เดิมและสีชมพูหมายถึงอุปกรณ์ที่เพิ่มใหม่ ใช้อ้างอิงสำหรับการวาดกราฟิก

**Auto Mode:**

Black word is old sequence  
Pink words is new sequence



ภาพที่ 3.2 ฟังงาน Auto Mode

จากภาพที่ 3.2 คือผังงานแสดงลำดับการทำงาน สีดำหมายถึงระบบที่มีการใช้งานอยู่เดิม และสีชมพูหมายถึงส่วนที่ต้องเพิ่มการทำงานให้กับระบบ โดย V คือวาล์ว M คือมอเตอร์ R คือวาล์วควบคุม T คืออุณหภูมิ P คือความดัน และ SP คือค่าเป้าหมาย

ศึกษารายการอินพุตเอาต์พุตของแอดเดรส (Address) บน PLC ที่มีการยกเลิก (Cancel) เพื่อย้ายแอดเดรส (Relocate Address) แอดเดรสเดิมที่ใช้งานอยู่ (Existing) และอินพุตเอาต์พุตใหม่ (New Input/Output) ทั้งส่วนของดิจิทัลและอนาล็อก

ตารางที่ 3.1 ตัวอย่างรายการอนาล็อกอินพุต

Type	Tag	IO list	Type	Remark
AI 4-20 mA	S01 (0-1440)	PIW 256	WORD	Cancel
AI 4-20 mA	S04 (0-50)	PIW 260	WORD	
AI 4-20 mA	P04 (1-260)	PIW 262	WORD	
AI 4-20 mA	S01 (0-1440)	PIW 256	WORD	
RTD PT100 STD	P03 (0-70)	PIW 268	WORD	Existing
4-20 mA (2 Wire)	P15 (-200 - 200)	PIW 270	WORD	
RTD PT100 STD	T01	PIW 298	WORD	
RTD PT100 STD	T02	PIW 302	WORD	
4-20 mA (4 Wire)	S01 (0-1440)	PIW 346	WORD	Relocate Address
4-20 mA (4 Wire)		PIW 348	WORD	
4-20 mA (2 Wire)	S04 (0-50)	PIW 350	WORD	
4-20 mA (2 Wire)	P04 (1-260)	PIW 352	WORD	
4-20 mA (2 Wire)	T01T10	PIW 362	WORD	New I/O
4-20 mA (2 Wire)	T01T11	PIW 364	WORD	
4-20 mA (4 Wire)	T01T15	PIW 366	WORD	
4-20 mA (4 Wire)	T01T16	PIW 368	WORD	

ศึกษาสัญญาณเตือนสำหรับกระบวนการ โดยแบ่งออกเป็น 3 ระดับคือ

1) เตือน (Warning) เป็นการแสดงข้อความแจ้งเตือนที่หน้ากราฟิกเมื่อเกิดความผิดปกติที่เกิดขึ้นกับอุปกรณ์ โดยจะไม่หยุดการทำงานของอุปกรณ์ และข้อความจะหายไปเมื่อสถานะของบิตที่ทำให้เกิดสัญญาณเตือนกลับสู่สภาวะปกติ

2) Control shutdown เป็นการหยุดการทำงานของอุปกรณ์โดยการหยุดที่ละอุปกรณ์อย่างเป็นขั้นตอนเมื่อเกิดความผิดปกติกับอุปกรณ์ที่ส่งผลกระทบต่อการทำงานของกระบวนการ และจะแสดงข้อความเตือนที่หน้ากราฟิกด้วย

3) Emergency shutdown เป็นการหยุดการทำงานของทุกอุปกรณ์ทันทีเมื่อเกิดความผิดปกติกับอุปกรณ์ที่จะส่งผลกระทบร้ายแรงต่อกระบวนการ และจะแสดงข้อความที่ระบุถึงสาเหตุความผิดปกติบนหน้ากราฟิกด้วย

SEQUENTIAL AND INTERLOCK TABLE							Warning (W)	Control Shutdown	Emergency shutdown
No	Device Tag Function	Device Tag No.	Alarm	Alarm Type	Cause / Description	Operation mode	Remarks		
1	GEA_05	T15	X	H	T15 TEMPERATURE FBD ALARM HIGH +130 C	ALL MODE	GEA TEMP 03 HI ALARM		
2	GEA_03	T15	X	HH	GEA TEMPERATURE ALARM HIGH HIGH +130 C	ALL MODE	GEA TEMP HI HI ALARM		
3	GEA_02	T15	X	HHH	GEA Fire water on	ALL MODE	GEA TEMP HI HI HI ALARM		
4	GEA_01	GEA	X	Fault	GEA Major Fault	ALL MODE	GEA Major Fault		
5	GEA_04	GEA	X	Fault	GEA Minor Fault	ALL MODE	GEA Minor Fault		Interlock Prohibit
6	DPT	P30	X	H	P30 DPT Pressure Alarm High	ALL MODE	10 sec		A&A Alarm SP
7	DPT	P31	X	H	P31 DPT Pressure Alarm High	ALL MODE	10 sec		A&A Alarm SP
8	TTAH	T14	X	H	T14 TEMPERATURE PRODUCT ALARM HIGH + 40	ALL MODE	10 sec		A&A Alarm SP
9	TTAH	T18	X	H	T18 TEMPERATURE COOL AIR ALARM HIGH	ALL MODE	60 sec After AHU Running		A&A Alarm SP
10	TTAH	T19	X	H	T19 TEMPERATURE HOT AIR ALARM HIGH (80DegC)	ALL MODE			A&A Alarm SP
11	TRIP	M11	X	TR	Heating line blower (M11) trip	ALL MODE			
12	TRIP	M17	X	TR	Cooling line blower (M17) trip	ALL MODE			
13	GDH	H49	X	Fault	H49 Valve Fault	ALL MODE			
14	GDH	H65	X	Fault	H65 Valve Fault	ALL MODE			
15	On-off valve	V22	X	Fault	V22 Valve Fault	ALL MODE			
16	On-off valve	V23	X	Fault	V23 Valve Fault	ALL MODE			
17	TRIP	M23	X	TR	M23 CP Pump Trip	ALL MODE			
18	FSAL	F18	X	L	F18 Flow Switch Alarm Low	ALL MODE	10 After M23 Running		
19	On-off valve	V45	X	Fault	V45 Valve Fault	ALL MODE			
20	On-off valve	V46	X	Fault	V46 Valve Fault	ALL MODE			
21	On-off valve	V20	X	Fault	V20 Valve Fault	ALL MODE			
22	On-off valve	V21	X	Fault	V21 Valve Fault	ALL MODE			
23	On-off valve	V27	X	Fault	V27 Valve Fault	ALL MODE			
24	TTAH	T16	X	H	T16 TEMPERATURE FBD ALARM HIGH (see DegC)	ALL MODE			A&A Alarm SP
25	On-off valve	V47	X	Fault	V47 Valve Fault	ALL MODE			
26	On-off valve	V48	X	Fault	V48 Valve Fault	ALL MODE			

ภาพที่ 3.3 ตารางสัญญาณเตือน (Alarm) และการเกี่ยวข้งกันของอุปกรณ์ (Interlock)

### 3.1.2 ออกแบบโปรแกรมควบคุม

#### 3.1.2.1 สร้างตัวแปรในซอฟต์แวร์ SIMATIC S7

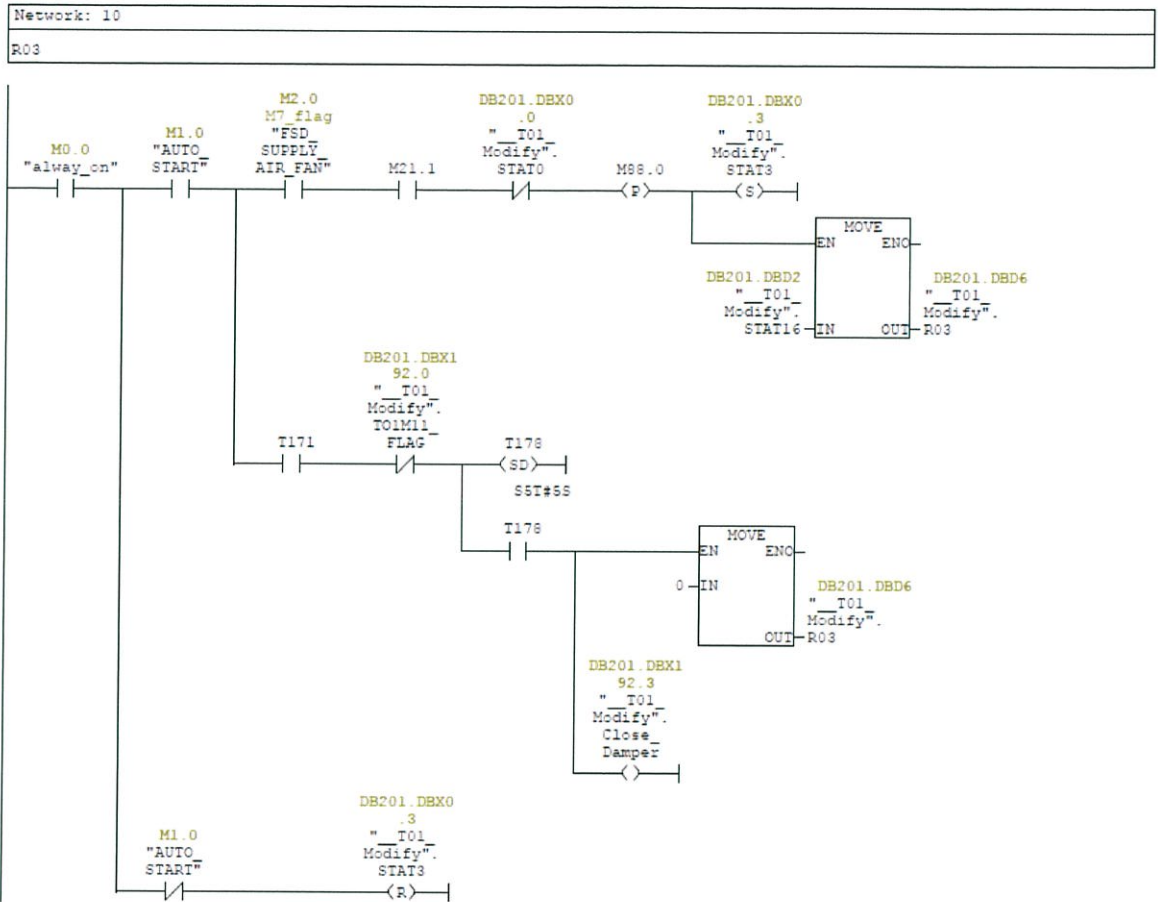
กำหนดชื่อและแอดเดรสตามรายการอินพุตเอาต์พุต (I/O list) กำหนดชนิดข้อมูล เช่น Boolean, Integer, WORD, TIMER เป็นต้น และใส่รายละเอียด (Comment) เกี่ยวกับตัวแปรนั้นๆ และหากอินพุตเอาต์พุตมีมากกว่าขนาดที่ PLC รองรับได้ สามารถสร้างแอดเดรสใช้งานได้บน Data block

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	STAT0	BOOL	FALSE	
+0.1	STAT1	BOOL	FALSE	
+0.2	STAT2	BOOL	FALSE	
+0.3	STAT3	BOOL	FALSE	
+0.4	STAT4	BOOL	FALSE	
+0.5	STAT5	BOOL	FALSE	
+0.6	STAT6	BOOL	FALSE	
+0.7	STAT7	BOOL	FALSE	
+1.0	STAT8	BOOL	FALSE	
+1.1	STAT9	BOOL	FALSE	
+1.2	STAT10	BOOL	FALSE	
+1.3	STAT11	BOOL	FALSE	
+1.4	STAT12	BOOL	FALSE	
+1.5	WIP_PLAS	BOOL	FALSE	
+1.6	STAT14	BOOL	FALSE	
+1.7	STAT15	BOOL	FALSE	
+2.0	STAT16	REAL	0.000000e+000	
+6.0	R03	REAL	0.000000e+000	
+10.0	STAT18	REAL	0.000000e+000	
+14.0	STAT19	REAL	0.000000e+000	
+18.0	STAT20	REAL	0.000000e+000	
+22.0	R04	REAL	0.000000e+000	
+26.0	STAT22	REAL	0.000000e+000	
+30.0	STAT23	WORD	#16#0	
+32.0	STAT24	WORD	#16#0	
+34.0	STAT25	ARRAY[0..30]	0	
+2.0		INT		
+56.0	STAT26	ARRAY[0..30]	0	
+2.0		INT		
+158.0	STAT27	REAL	0.000000e+000	
+162.0	STAT28	REAL	0.000000e+000	
+166.0	STAT29	BOOL	FALSE	
+166.1	STAT30	BOOL	FALSE	
+168.0	STAT31	DINT	L#0	
+172.0	STAT32	REAL	0.000000e+000	
+176.0	FOI319	REAL	0.000000e+000	

ภาพที่ 3.4 Data block



M1.7 เป็นบิตความจำมีสถานะเป็น 1 เมื่อมอเตอร์ M07 ทำงาน  
M5.5 เป็นบิตความจำมีสถานะเป็น 1 เมื่อเกิดการ Control Shutdown



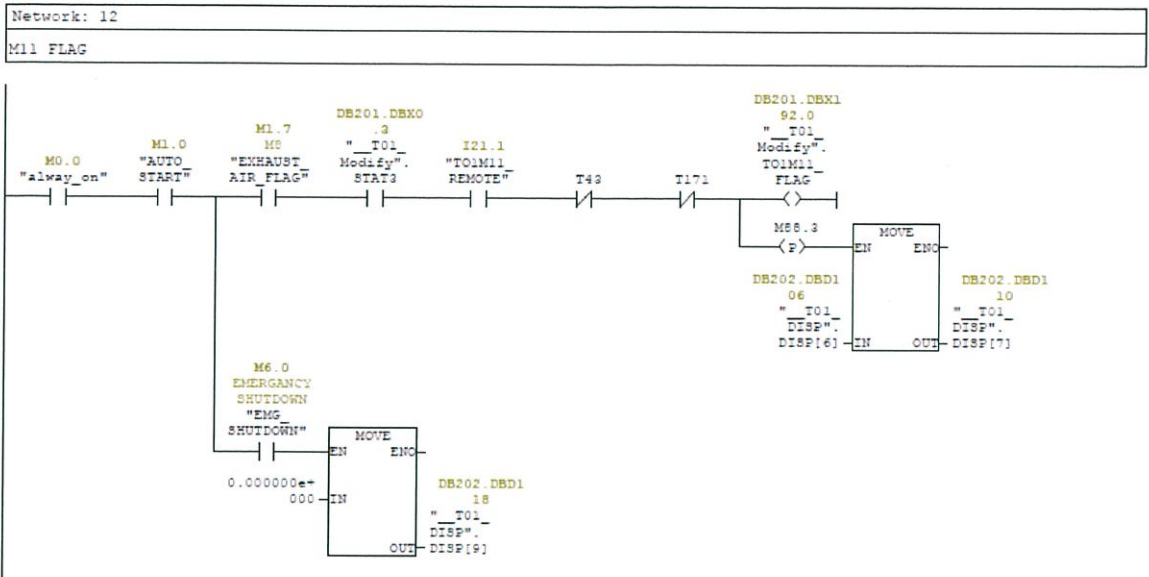
ภาพที่ 3.7 คำสั่งการเปิดปิดวาล์วควบคุม

จากภาพที่ 3.7 เขียนคำสั่งควบคุมการเปิดปิดวาล์วโดยนำ M21.1 ซึ่งเป็นบิตความจำ ที่มีสถานะเป็น 1 เมื่อมีการเปิดวาล์ว R42 ได้ 50% มาใช้เป็นเงื่อนไขการทำงานขั้นถัดไป

1) เมื่อ M0.0 M1.0 M2.0 M21.1 มีสถานะเป็น 1 และ DB201.DBX0.1 มีสถานะเป็น 0 จะมีสัญญาณขอบขาขึ้น (Positive transition) ที่ M88.0 ไปเซตให้ DB201.DBX0.3 มีสถานะเป็น 1 และเกิดการ MOVE ค่าจาก DB201.DBX2 ซึ่งเป็นแอดเดรสที่มีการรับค่า (Input) เปอร์เซ็นต์การเปิดวาล์วจากกราฟที่อุปกรณ์ HMI ไปที่ DB201.DBD6 ซึ่งเป็นแอดเดรสที่นำไปแสดงผลบนกราฟิก (Output)

2) เมื่อ M0.0 M1.0 T171 มีสถานะเป็น 1 และ DB201.DBX192.0 มีสถานะเป็น 0 ทำให้ตัวนับเวลา (Timer) T178 นับเวลา 5 วินาที หลังจากนั้นจะ MOVE ค่า 0 ไปที่ DB201.DBD6 ซึ่งเป็นคำสั่งการปิดวาล์วจากการเกิด Control shutdown

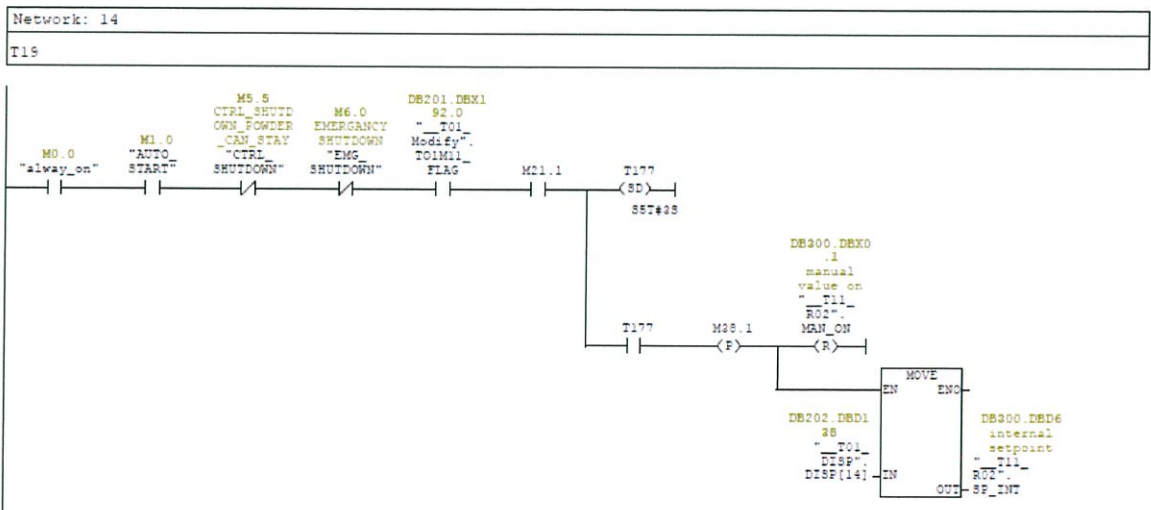
3) เมื่อ M0.0 มีสถานะเป็น 1 และ M1.0 มีสถานะเป็น 0 คือจะรีเซต DB201.DBX0.3 ให้มีสถานะเป็น 0 เมื่อไม่มีการทำงานในโหมด Auto



ภาพที่ 3.8 คำสั่งให้มอเตอร์ทำงาน

จากภาพที่ 3.8 เขียนคำสั่งให้มอเตอร์ทำงานโดยนำ DB201.DBX0.3 จากภาพที่ 3.7 มาเป็นบิตเงื่อนไขสั่งงานมอเตอร์ M11

- 1) เมื่อ M0.0 M1.0 M1.7 DB201.DBX0.3 I21.1 มีสถานะเป็น 1 และ T49 T171 มีสถานะเป็น 0 จะทำให้ DB201.DBX192.0 มีสถานะเป็น 1 ซึ่งเป็น Flag ของมอเตอร์ M11 นำไปสั่งให้มอเตอร์ทำงาน
- 2) มีการ MOVE ค่าจาก DB202.DBD106 ซึ่งเป็นค่าเปอร์เซ็นต์การเปิดมอเตอร์ที่รับจากกราฟิก ส่งไปยัง DB202.DBD110 เพื่อนำไปสเกลคำสั่งการเปิดมอเตอร์

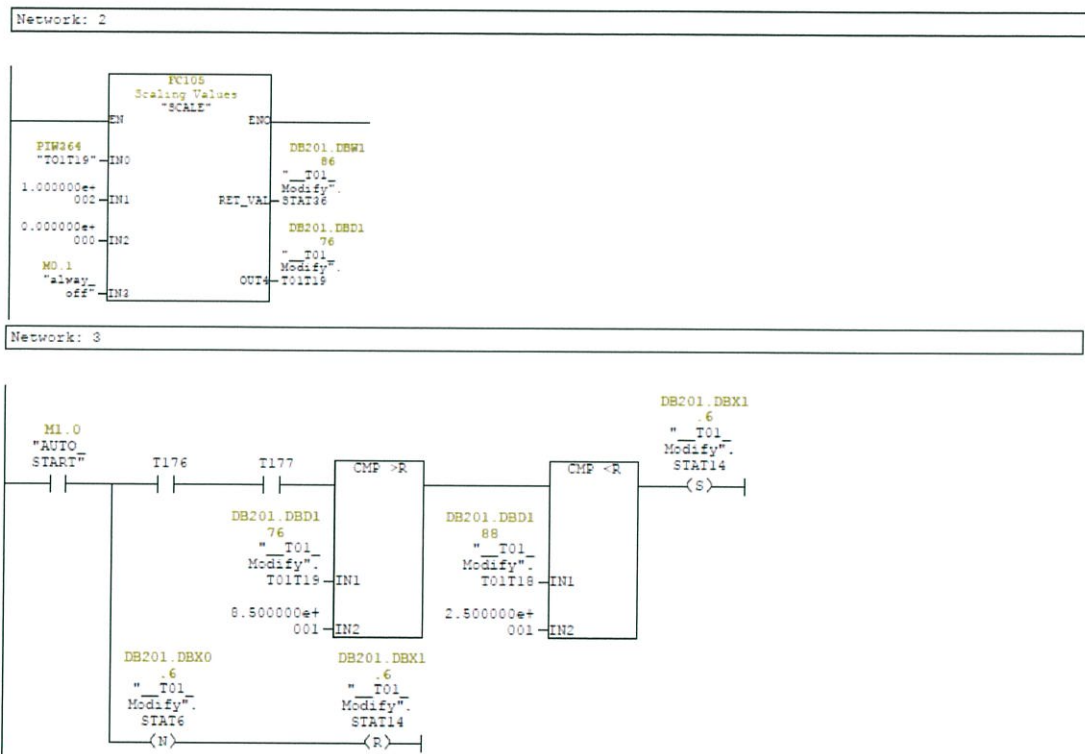


ภาพที่ 3.9 คำสั่งใช้งาน PID ในโหมด Auto

จากภาพที่ 3.9 เขียนคำสั่งให้ตัวควบคุม PID ทำงาน โดยหาก DB300.DBX0.1 มีค่าเป็น 0 จะทำงานในโหมดอัตโนมัติ (Auto) และหากมีค่าเป็น 1 จะทำงานในโหมด Manual

1) M0.0 M1.0 DB201.DBX192.0 M21.1 มีสถานะเป็น 1 และ M5.5 M6.0 มีสถานะเป็น 0 จะทำให้ T177 นับเวลา 3 วินาที เมื่อนับครบ T177 จะมีสถานะเป็น 1 แล้วเกิดสัญญาณขอขาขึ้นไปรีเซ็ต DB300.DBX0.1 ซึ่งเป็นบิตเลือกโหมด Auto หรือ Manual ในฟังก์ชันบล็อก PID หากบิตนี้มีค่าเป็น 1 จะทำงานในโหมด Manual และหากบิตนี้มีค่าเป็น 0 จะทำงานในโหมด Auto

2) ขอขาขึ้นจะทำให้เกิดการ MOVE ค่าจาก DB202.DBD138 ที่เป็นค่า Set point ของ PID ที่รับจากกราฟิก ย้ายไปที่ DB301.DBD6 ซึ่งเป็นพารามิเตอร์ Set point ใน FB41



ภาพที่ 3.10 การรับค่าอนาล็อกและเงื่อนไข

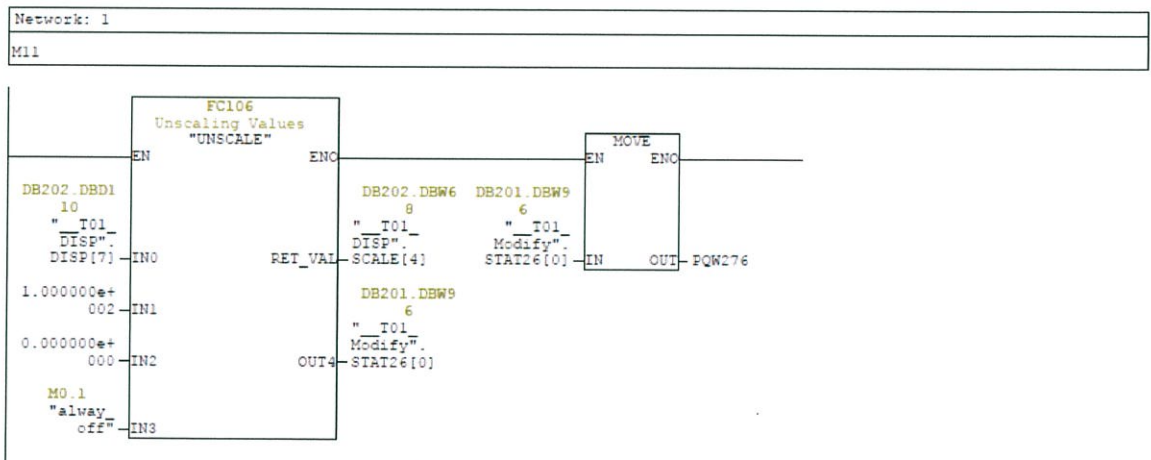
จากภาพที่ 3.10 Network 2 คือการรับค่าวัดจากเครื่องมือวัดอุณหภูมิ T19 เข้ามาเป็นสัญญาณอนาล็อก ค่า 0 ถึง 27648 (4-20 mA) ผ่านบล็อกสเกล FC105 (Scaling Value) กำหนดค่าสูงสุดต่ำสุดเป็น 0 และ 100 แล้วส่งไปเก็บไว้ที่ DB201.DBD176 เพื่อนำไปใช้เปรียบเทียบกับค่าอุณหภูมิให้มากกว่า 85 องศาเซลเซียสใน Network 3 แล้วเซตบิต DB201.DBX1.6 เพื่อทำงานในขั้นถัดไป โดยค่าเอาต์พุตจาก FC105 จะเปลี่ยนแปลงตามสมการที่ 1.1

$$OUT = [((IN0 - 0) / (27648 - 0)) * (IN1 - IN2)] + IN2 \quad (1.1)$$

เมื่อ IN0 คือ ค่าอินพุต

IN1 คือ ค่าสูงสุด  
 IN2 คือ ค่าต่ำสุด  
 OUT คือ ค่าเอาต์พุต มีค่าตั้งแต่ IN2 ถึง IN1 ในที่นี้คือ 0-100 โดยค่าที่  
 รับจากทรานสมิตเตอร์ที่มีสัญญาณเอาต์พุตเป็น 4-20 mA จะสเกลได้ค่าดังนี้

- 4 mA = 0
- 8 mA = 6912
- 12 mA = 13824
- 16 mA = 20736
- 20 mA = 27648

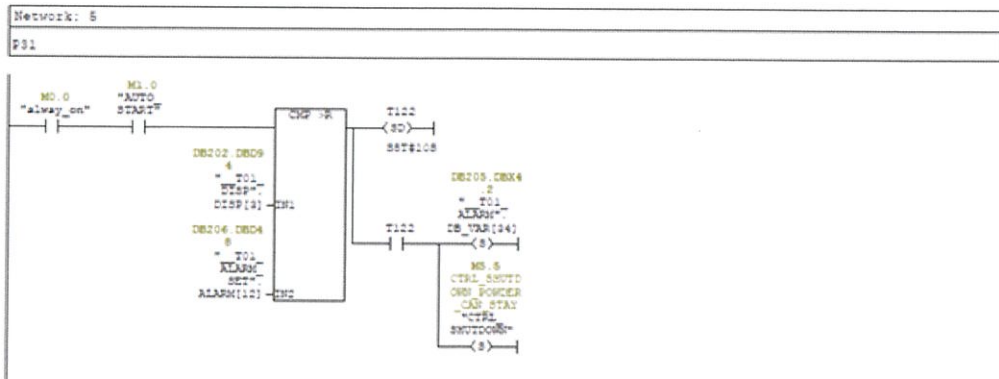


ภาพที่ 3.11 การสเกลค่าผ่าน FC106

จากภาพที่ 3.11 เป็นการสเกลค่าผ่าน FC106 โดยการทำงานของ FC106 จะคล้ายกับ FC105 เพียงแต่สลับฝั่งอินพุตและเอาต์พุต เมื่อรับคำสั่งค่าการขับเคลื่อนมอเตอร์จากกราฟิกเก็บไว้ที่ DB202.DBD110 นำมาผ่านบล็อก FC106 สเกลค่า 0-100 ได้เป็นค่าเอาต์พุตเป็นข้อมูลประเภท Integer มีค่า 0-27648 (4-20mA) ออกไปที่อนาล็อกเอาต์พุตแอดเดรส PQW276 เพื่อนำไปขับเคลื่อนมอเตอร์ M11 โดยค่าเอาต์พุตจาก FC106 (Unscaling Value) จะเปลี่ยนแปลงตามสมการที่ 1.2<sup>[7]</sup>

$$OUT = [((IN0 - IN2)/(IN1 - IN2)) * (27648-0)] + 0 \quad (1.2)$$

- เมื่อ
- IN0 คือ ค่าอินพุต
  - IN1 คือ ค่าสูงสุด
  - IN2 คือ ค่าต่ำสุด
  - OUT คือ ค่าเอาต์พุต มีค่าตั้งแต่ 0 - 27648

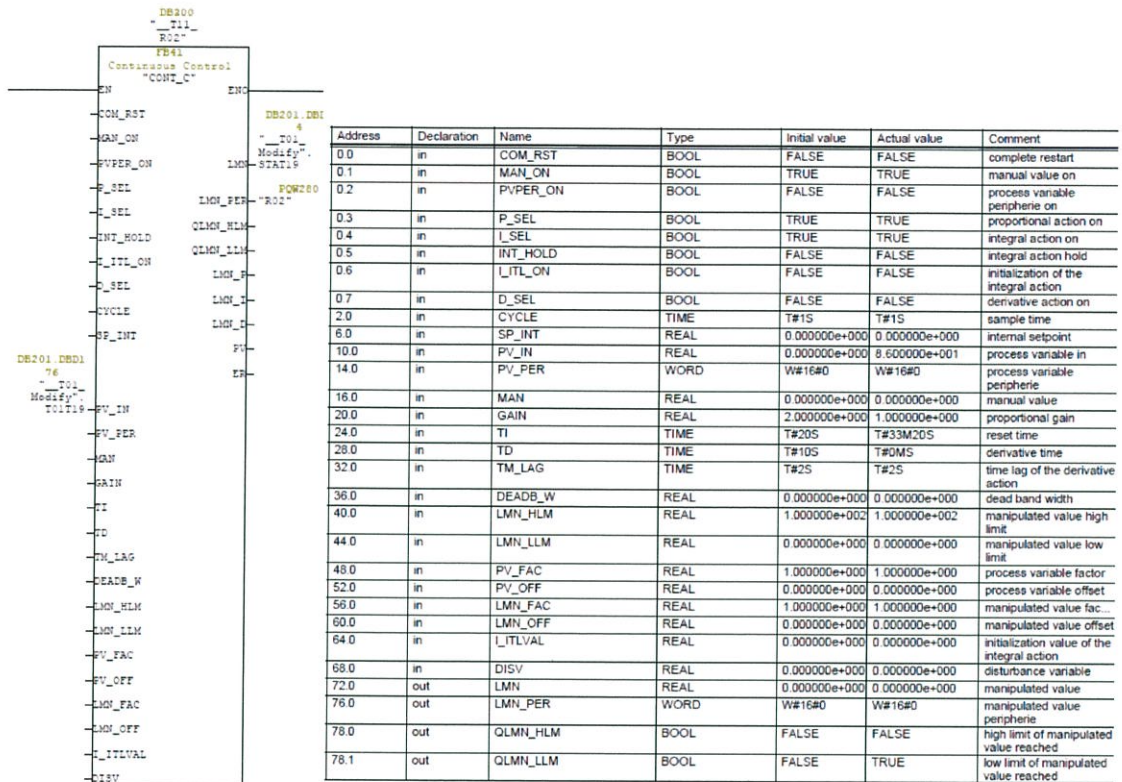


ภาพที่ 3.12 คำสั่งให้เกิดการแจ้งเตือน (Alarm)

จากภาพที่ 3.12 เป็นสัญญาณเตือนที่เกิดจากการทำงานในโหมด Auto และค่าวัดจากทรานสมิตเตอร์วัดความดันของ P31 คือ DB202.DBD94 มีค่ามากกว่า DB 206.DBD46 ซึ่งเป็นค่า Setting ที่รับค่าจากกราฟิก จะทำให้ T122 นับเวลา 10 วินาที แล้วเกิด Control shutdown พร้อมกับข้อความแจ้งเตือนบนหน้ากราฟิก

### 3.1.2.3 การใช้ฟังก์ชันบล็อก 41

ในซอฟต์แวร์ SIMATIC STEP7 จะมีฟังก์ชันบล็อกสำเร็จรูปให้เลือกใช้งาน โดยการควบคุมด้วย PID จะใช้ฟังก์ชันบล็อก FB 41 "CONT\_C" (continuous controller) ใช้กับ SIMATIC STEP7 PLC เพื่อควบคุมกระบวนการทางเทคนิคด้วยตัวแปรอินพุตและเอาต์พุตอย่างต่อเนื่อง ในการกำหนดพารามิเตอร์ โดยเงื่อนไขของการใช้ฟังก์ชันบล็อกนี้คือต้องมี Instance Data Block ขึ้นมารองรับพารามิเตอร์ของ FB41 ในที่นี้ใช้ DB300 เมื่อกำหนดแล้ว ซอฟต์แวร์จะสร้าง DB300 ขึ้นมา ภายในประกอบด้วยแอดเดรสแต่ละพารามิเตอร์ของ FB41 โดยการทำงานจะคำนวณค่าในบล็อกควบคุมเท่านั้น หากต้องการค่าที่แม่นยำ บล็อกต้องถูกเรียกใช้เป็นประจำ ด้วยเหตุนี้ ควรเรียกบล็อกควบคุมใน cyclic interrupt OB (OB30 ถึง OB38) หรือบ่อยเวลาในการสุ่มตัวอย่างในพารามิเตอร์ CYCLE



ภาพที่ 3.13 FB 41 และ Instance Data Block

## 3.2 ออกแบบการแสดงผล

### 3.2.1 การสร้างแท็ก (Tag) บนอุปกรณ์ HMI

สามารถสร้างแท็กใช้งานบน HMI โดยมีขั้นตอนดังภาพที่ 3.14

- 1) เลือก communication แล้วเลือก Tag สร้างโพลเดอร์สำหรับโปรเจกต์นี้คือ T01
- 2) จะปรากฏหน้าต่างขึ้นมาโดยมีรายละเอียดคือ

Name : ชื่อแท็ก ควรตั้งให้สอดคล้องกับที่จะนำไปใช้

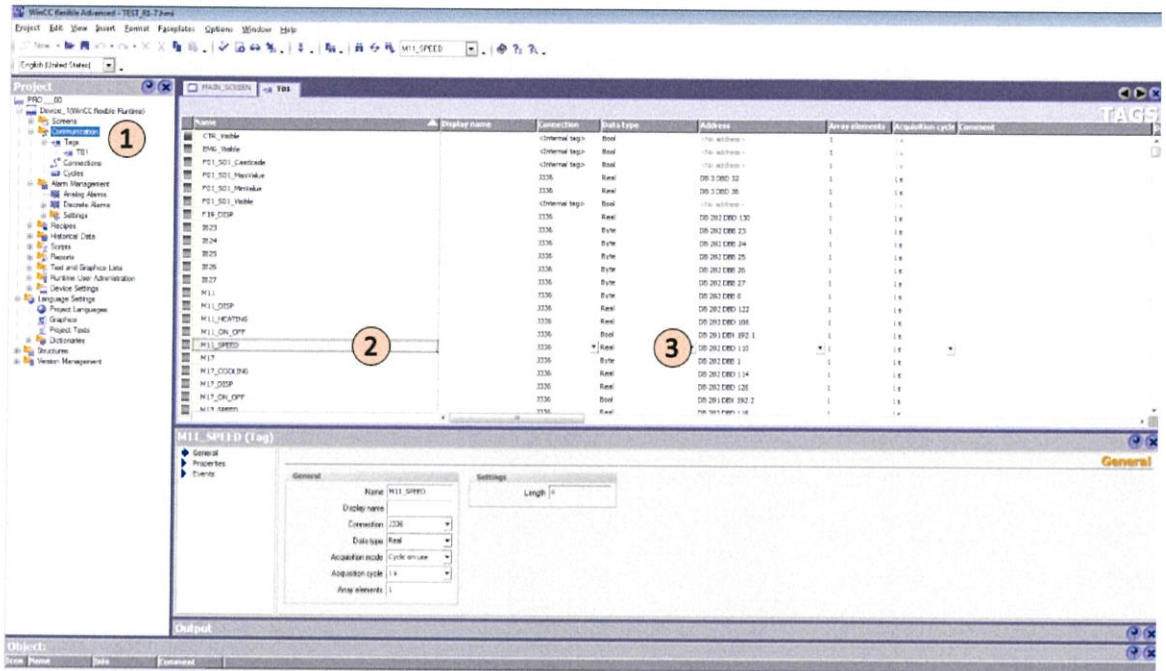
Connection : การเชื่อมต่อ หากไม่มีแอดเดรสใน PLC สามารถใช้แอดเดรสภายในโดยเลือก Internal Tag หรือใช้แอดเดรสที่เชื่อมต่อกับ PLC โดยเลือกชื่อที่ทำการเชื่อมต่อกับ PLC อยู่

Data type : ชนิดข้อมูล กำหนดชนิดข้อมูลของแท็กนั้น ๆ เช่น Bool Real Word

Address : แอดเดรสของแท็ก สำหรับ Internal tag ไม่ต้องกำหนดแอดเดรส

- 3) กำหนดชนิดข้อมูล (Data type) ให้กับแท็กนั้น ๆ

สามารถกำหนดขีดจำกัด (Limit) ของแท็กนั้น ๆ ได้ โดยเลือก Properties --> limit จะปรากฏขีดจำกัดให้เลือกรวมถึงค่าคงที่ (Constant) แท็ก (Tag) และหากไม่ต้องการกำหนดขีดจำกัดเลือก Disable



ภาพที่ 3.14 การสร้างแท็กสำหรับกราฟิก

### 3.2.2 การสร้างวัตถุ (Object)

การสร้างปุ่มกดโดยใช้สวิตช์ดังภาพที่ 3.15

1) เลือกวัตถุจากหน้าต่างเครื่องมือ (Tool windows) ในที่นี้เลือก Switch จากนั้นลากลงบนพื้นที่ทำงาน (Work area)

2) คลิกขวาบนวัตถุที่ต้องการแก้ไข เลือกคุณสมบัติ (Property) จะปรากฏเมนูที่มุมมองคุณสมบัติ (Property view) ที่ด้านล่าง

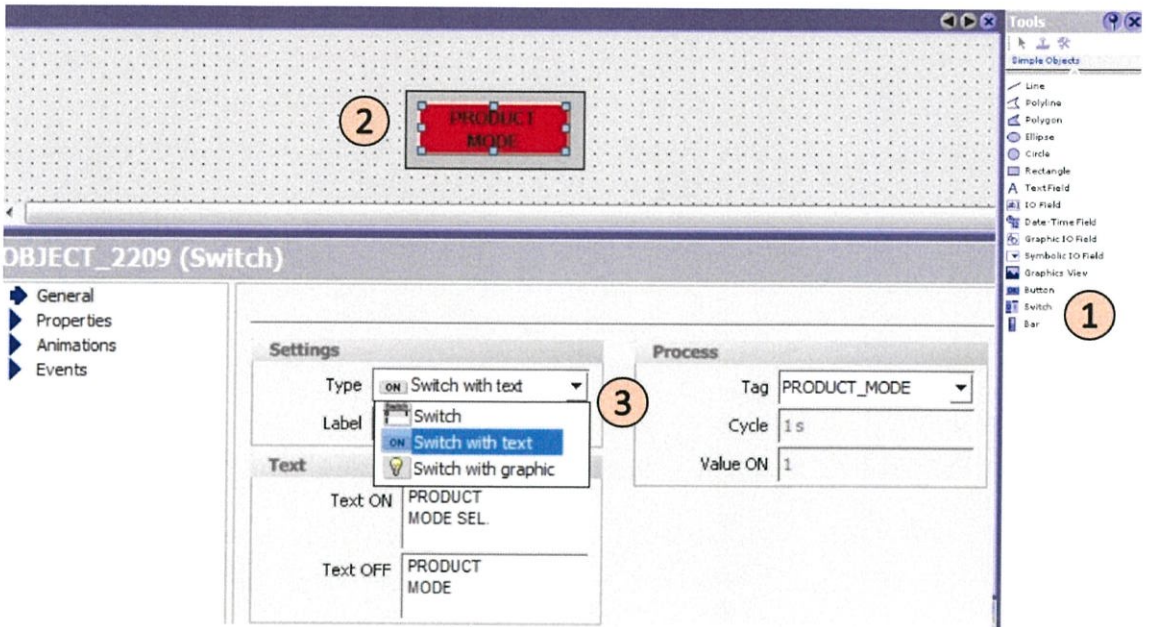
3) เลือก General เพื่อใส่แท็ก และแก้ไขชนิดของสวิตช์

Switch เมื่อให้เป็นสวิตช์เท่านั้น

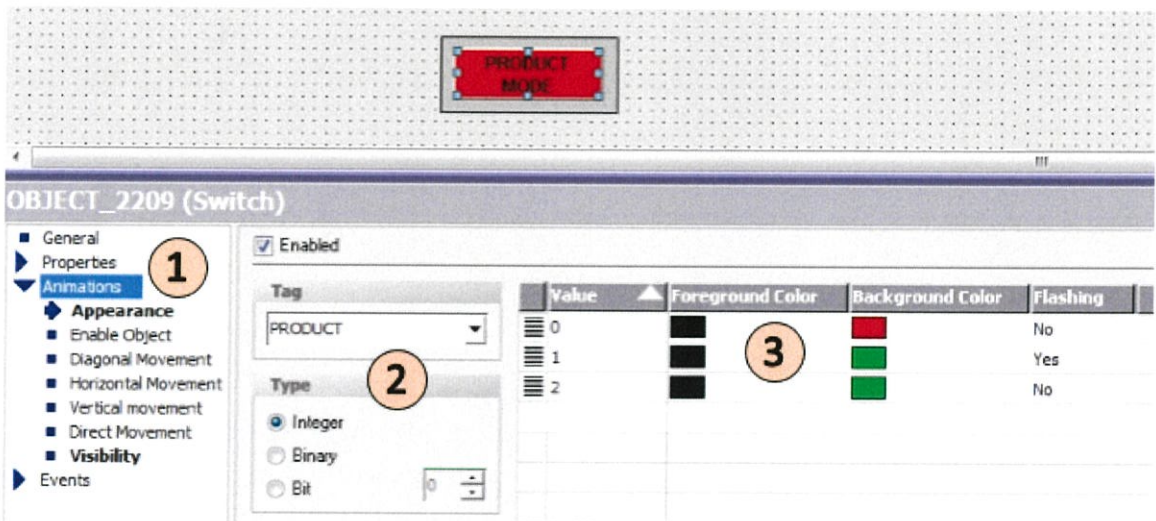
Switch with text เป็นตัวอักษรบนสวิตช์

Switch with graphic เป็นกราฟิกบนสวิตช์

หากเลือกเป็นตัวอักษรบนสวิตช์จะปรากฏช่องสำหรับใส่ข้อความบนสวิตช์เมื่อสวิตช์เปิดและปิด



ภาพที่ 3.15 การสร้างปุ่มกด



ภาพที่ 3.16 การตั้งค่าการเคลื่อนไหวให้กับวัตถุ

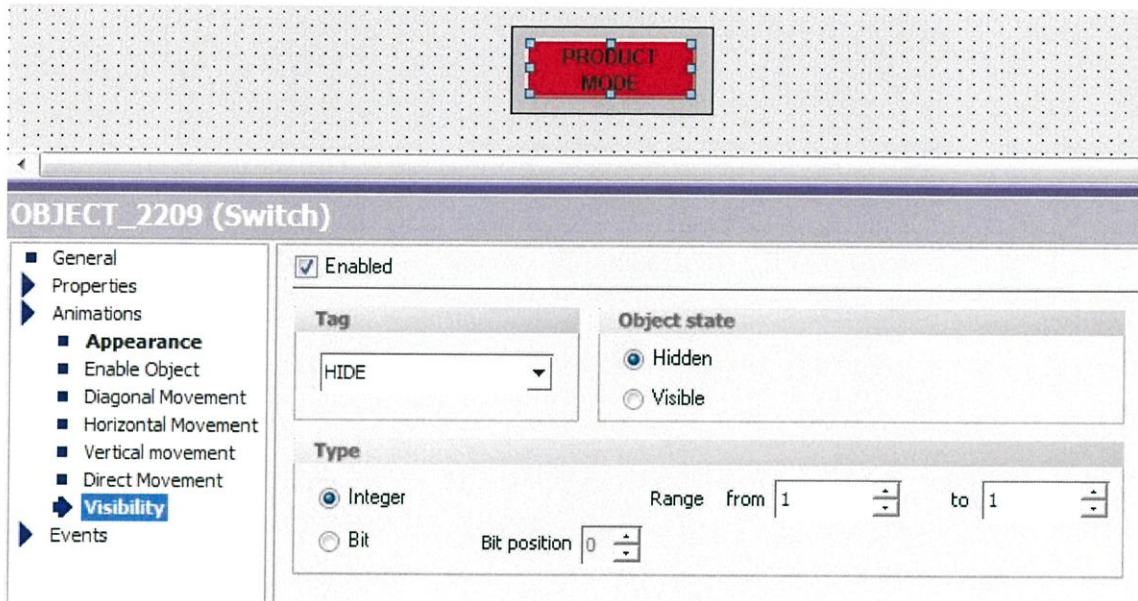
ดังภาพที่ 3.16 เลือก Properties เพื่อกำหนดรูปแบบของวัตถุ เช่น สีข้อความ ขนาด ตั้งค่าการมองเห็น ตั้งค่าความปลอดภัยโดยการใส่รหัสให้กับวัตถุ และตั้งค่าการเคลื่อนไหวให้กับวัตถุโดย

- 1) เลือก Animation -> Appearance ทำเครื่องหมายถูกหน้า Enable เพื่อใช้งานโหมดนี้
- 2) เชื่อมโยงแท็กให้กับวัตถุ และกำหนดประเภทชนิดข้อมูล
- 3) กำหนดค่าการแสดงผล ตัวอย่างเช่น

เมื่อข้อมูลในแท็กมีค่าเป็น 0 ให้แสดงผลเป็นพื้นหลังสีแดงตัวหนังสือสีดำ ไม่กระพริบ

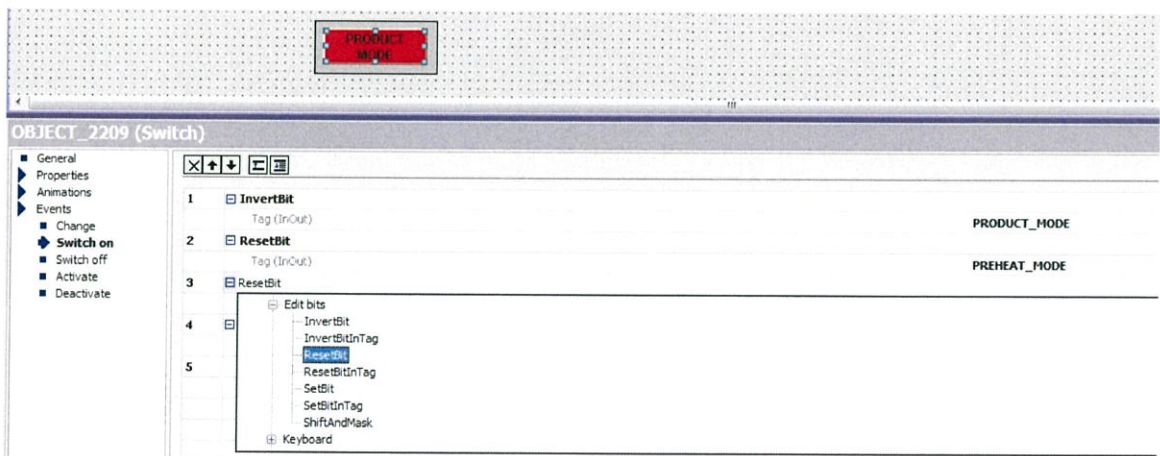
เมื่อข้อมูลในแท็กมีค่าเป็น 1 ให้แสดงผลเป็นพื้นหลังเขียวตัวหนังสือสีดำ กระพริบ

เมื่อข้อมูลในแท็กมีค่าเป็น 2 ให้แสดงผลพื้นหลังสีเขียวตัวหนังสือสีดำ ไม่กระพริบ



ภาพที่ 3.17 การตั้งค่าการปรากฏให้กับวัตถุ

ตั้งค่าการปรากฏให้กับวัตถุดังภาพที่ 3.17 โดยเลือก Animation -> Visibility ทำเครื่องหมายถูกหน้า Enable โดยให้ซ่อนการมองเห็น (Hidden) หากบิตในแท็กมีค่าเป็น 1 หรือ Visible หากบิตในแท็กไม่มีค่าเป็น 1 ให้มองเห็นได้



ภาพที่ 3.18 การตั้งค่าเหตุการณ์ให้กับวัตถุ

ตั้งค่าเหตุการณ์ (Event) ให้กับวัตถุเป็นเมื่อเกิดการเปลี่ยนแปลงได้ดังภาพที่ 3.18

1) เลือก Events สำหรับฟังก์ชันเหตุการณ์ โดยจะมีฟังก์ชันของสวิตช์ เช่น

Change : เมื่อสวิตช์เปลี่ยนแปลง

Switch on : เมื่อสวิตช์เปิด

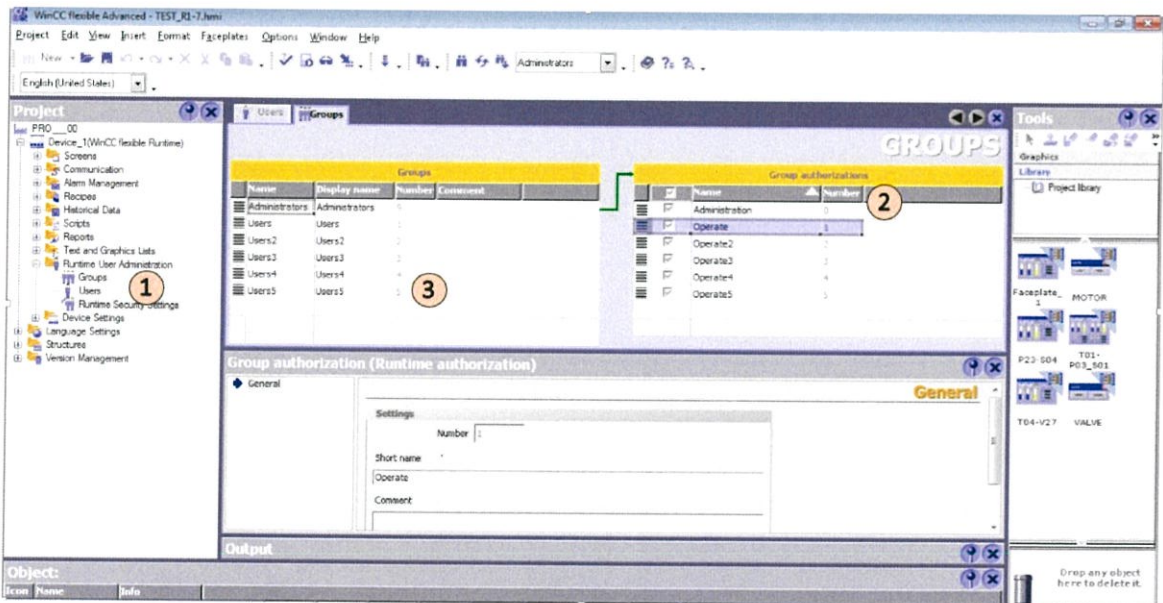
Switch off : เมื่อสวิตช์ปิด

- 2) เลือกเหตุการณ์ให้กับวัตถุ จากนั้นใส่แท็กให้กับเหตุการณ์ โดยมีเหตุการณ์ให้เลือกเช่น  
 Setbit : เมื่อกดสวิทช์จะเซตบิตในแท็ก เป็น 1  
 Resetbit : เมื่อกดสวิทช์จะรีเซตบิตในแท็ก เป็น 0  
 Invertbit : เมื่อกดสวิทช์จะเซตและรีเซตบิตในแท็กเป็น 1 และ 0 สลับกัน

### 3.2.3 การตั้งค่าการเข้าถึงวัตถุ

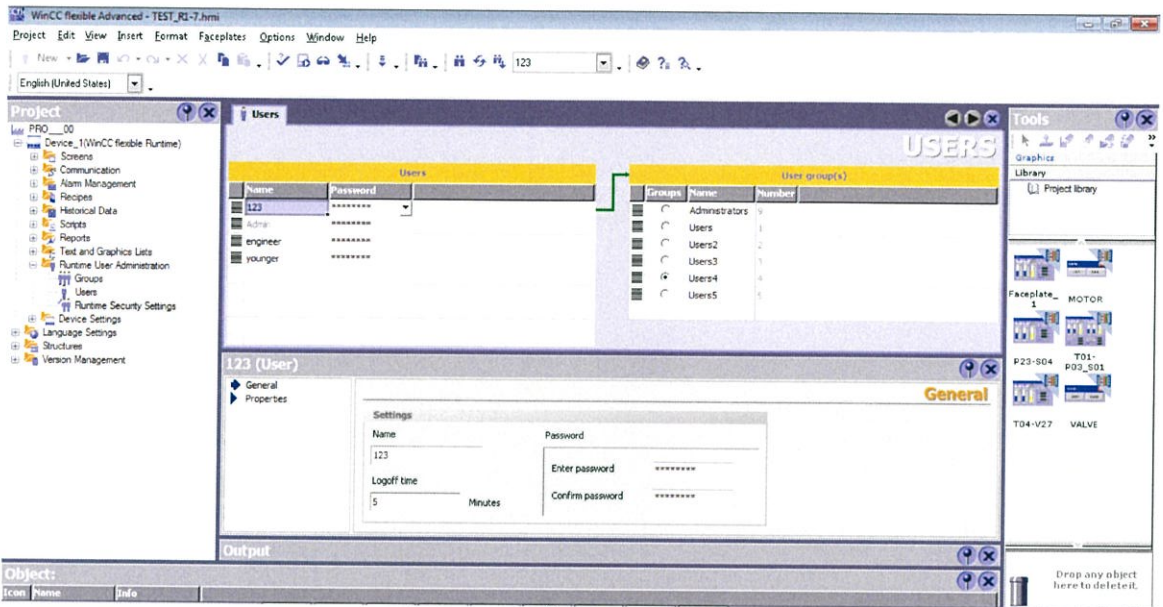
กำหนดกลุ่มการเข้าถึงวัตถุได้ดังภาพที่ 3.19

- 1) เลือกโฟลเดอร์ Runtime and Administration แล้วเลือก Group
- 2) สร้างแท็กการเข้าถึงโดยกำหนด Name ที่ด้าน Group Authorizations
- 3) สร้างกลุ่มการเข้าถึงโดยกำหนด Name และ Display name จากนั้นทำเครื่องหมายถูกหน้าแท็กที่ต้องการให้เข้าถึงได้ในฝั่ง Group Authorizations โดย Administrators สามารถเข้าถึงวัตถุได้ทั้งหมด

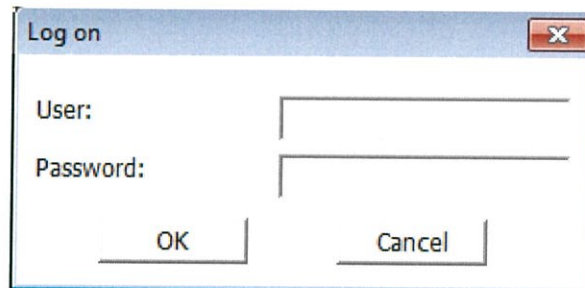


ภาพที่ 3.19 การกำหนดกลุ่มการเข้าถึง

สามารถตั้งค่าผู้ใช้และรหัสผ่านโดยเลือกโฟลเดอร์ Runtime and Administration แล้วเลือก Users เพื่อสร้าง ชื่อผู้ใช้และรหัสผ่าน แล้วเลือกกลุ่มการเข้าถึงของแต่ละผู้ใช้ โดยเมื่อมีการ Log in แต่ละชื่อผู้ใช้อาจเข้าถึงวัตถุได้ไม่เหมือนกัน ขึ้นกับการตั้งค่านี้



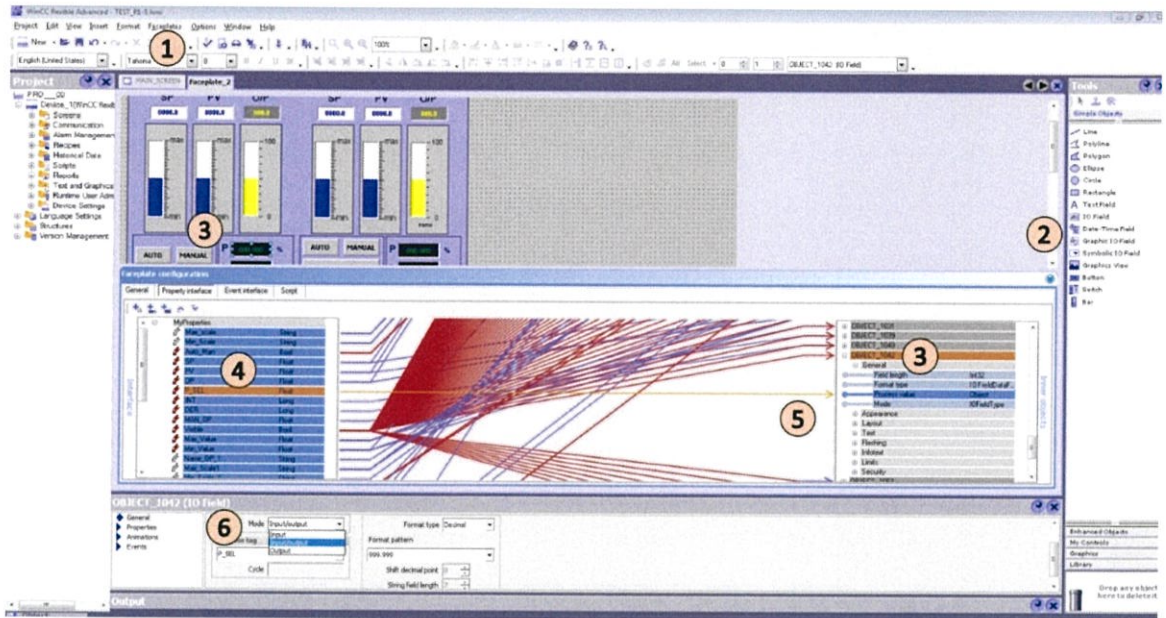
ภาพที่ 3.20 การตั้งค่าผู้ใช้และรหัสผ่าน



ภาพที่ 3.21 หน้า Log on

### 3.2.4 การสร้าง Faceplate

Faceplate ประกอบด้วยวัตถุและการแสดงผล ซึ่งสามารถจัดการและแก้ไขได้จากต้นแบบที่ถูกเก็บไว้ในไลบรารี (Library) สามารถเรียกและใช้งานหน้า Faceplate ได้โดยง่าย การใช้งานเหมาะสำหรับอุปกรณ์ที่มีฟังก์ชันการทำงานเหมือนกัน ในที่นี้จะนำมาใช้กับวาล์วควบคุมและมอเตอร์ ที่ถูกควบคุมการทำงานด้วย PID



ภาพที่ 3.22 การสร้างหน้าต่าง Faceplate

1) เลือก Faceplate -> Create faceplate จะปรากฏหน้าต่างที่มีพื้นที่ทำงาน  
 2) เพิ่มวัตถุที่ต้องการ โดยเลือกจากหน้าต่างเครื่องมือ (Tool windows) วางลงบนพื้นที่ทำงาน (Work area)

3) เมื่อวางเสร็จ จะปรากฏ Tag Object ของ Object แต่ละตัว

4) สร้างชื่อ Tag เพื่อรองรับ Tag จริง

5) เชื่อมโยงระหว่าง Tag ที่สร้างและวัตถุที่สร้าง

6) คลิกขวาที่วัตถุ เลือกคุณสมบัติ (Property)

General สำหรับตั้งค่า Tag รูปแบบการแสดงผล และ Mode ให้กับวัตถุ

Input : รับค่าเท่านั้น

Input/Output : รับค่าและแสดงผล

Output : แสดงผลเท่านั้น

Properties สำหรับกำหนดสี ขนาด ข้อความ การมองเห็น และความปลอดภัย

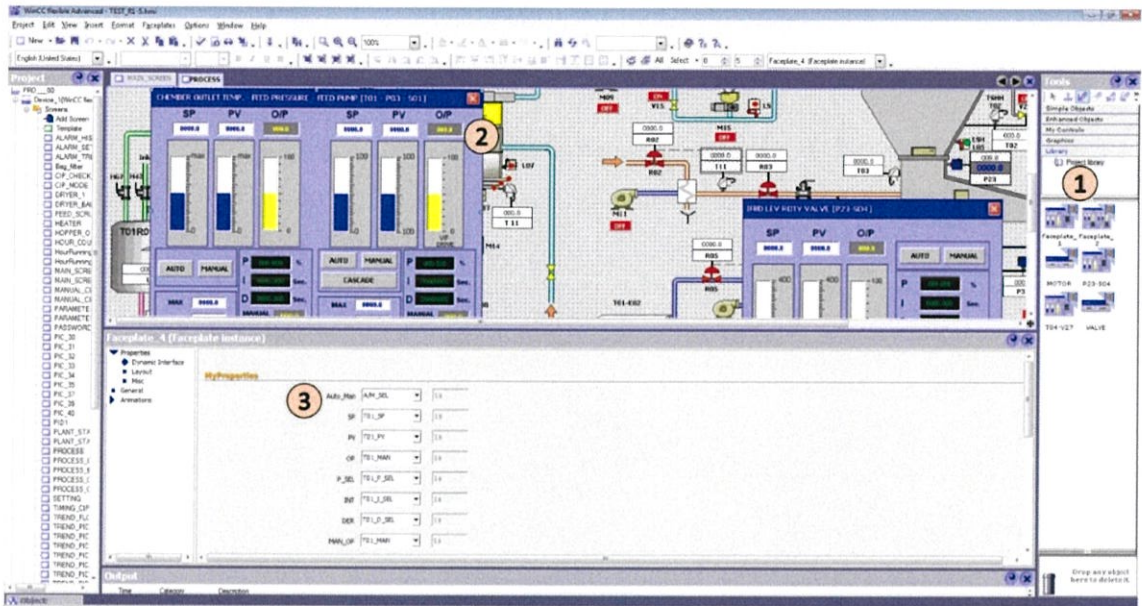
Animations สำหรับตั้งค่าการแสดงผล

Events สำหรับฟังก์ชันเหตุการณ์ โดยจะมีฟังก์ชันของ object เช่น

Setbit : เมื่อกด Object จะเซตบิตที่ tag อยู่กับ Object เป็น 1

Resetbit : เมื่อกด Object จะรีเซตบิตที่ Tag อยู่กับ Object เป็น 0

Invertbit : เมื่อกด Object จะเซตและรีเซตบิตเป็น 1 และ 0 สลับกัน



ภาพที่ 3.23 การเรียกใช้ Faceplate

1) เมื่อสร้าง Faceplate เสร็จจะปรากฏ Faceplate ที่ไลบรารีของหน้าต่างเครื่องมือ (Tool window)

2) ลาก Faceplate จากไลบรารีมาวางบนพื้นที่ทำงาน (Work area)

3) เลือก Properties ที่มุมมองคุณสมบัติ (Property view) จะปรากฏชื่อแท็กที่สร้างไว้ใน Faceplate สามารถใช้แท็กได้ที่ช่องว่างด้านข้าง

General สำหรับกำหนดขนาด และใส่ชื่อให้กับตัวแปรต่างๆ

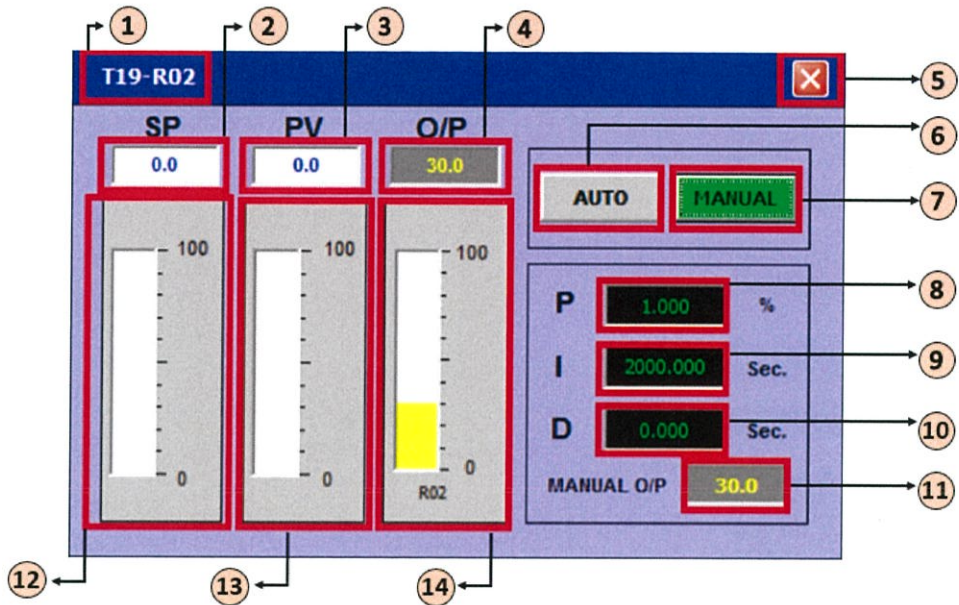
Animation สำหรับตั้งค่าการมองเห็นให้ Faceplate

หน้าต่าง Faceplate ประกอบด้วยประกอบด้วยส่วนรับและแสดงผลการควบคุมด้วยตัวควบคุม PID โดยรายละเอียดบนหน้า Faceplate ดังภาพที่ 3.24 ประกอบด้วย

- 1) ชื่อหน้าต่าง Faceplate
- 2) ส่วนรับและแสดงค่า Set point
- 3) ส่วนแสดงค่า Process value
- 4) ส่วนแสดงค่า Output
- 5) ปุ่มกดปิดหน้าต่าง Faceplate
- 6) ปุ่มกดทำงานในโหมด Auto
- 7) ปุ่มกดทำงานในโหมด Manual
- 8) รับและแสดงค่า P
- 9) รับและแสดงค่า I
- 10) รับและแสดงค่า D
- 11) รับและแสดงค่า Manual output
- 12) แสดงค่า Set point แบบระดับ

13) แสดงค่า Process value แบบระดับ

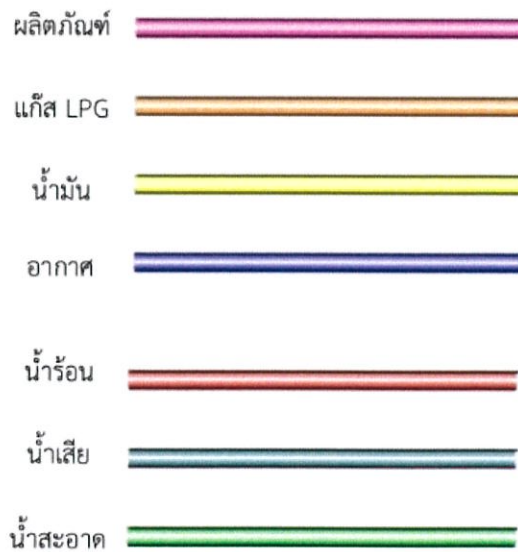
14) แสดงค่า Output แบบระดับ



ภาพที่ 3.24 หน้าต่าง Faceplate

### 3.2.5 สัญลักษณ์

ในหน้ากราฟิกใช้สัญลักษณ์ท่อและสีของสารในท่อดังภาพที่ 3.25 และใช้สัญลักษณ์ของเครื่องมือวัด และอุปกรณ์ต่าง ๆ ดังภาพที่ 3.26



ภาพที่ 3.25 สีของสารและสัญลักษณ์ท่อ

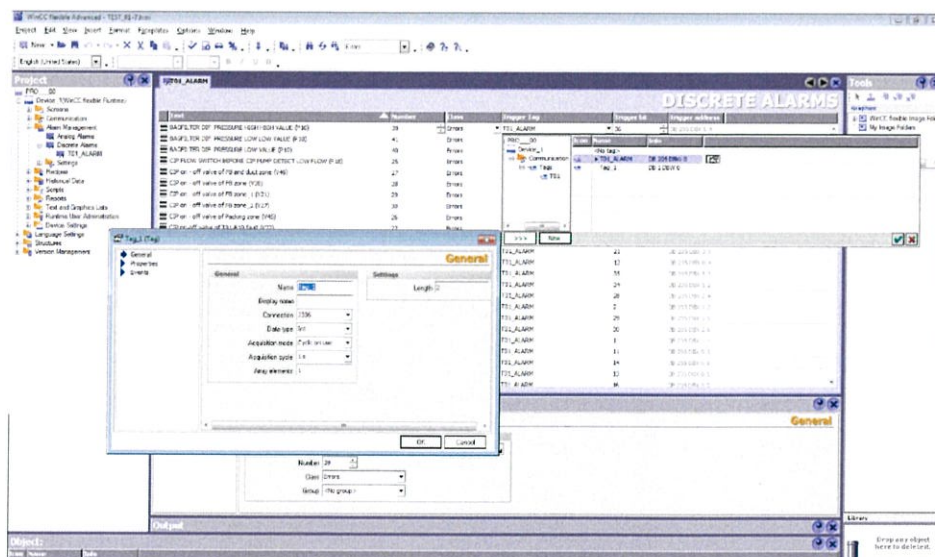


ภาพที่ 3.26 ชื่อและสัญลักษณ์อุปกรณ์ที่ใช้บนกราฟิก

### 3.3 การแสดงข้อความเตือน

#### 3.3.1 การสร้างแท็กของข้อความเตือน

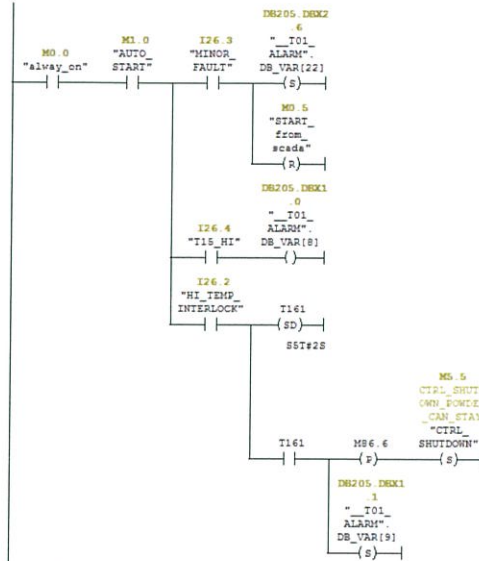
โดยกำหนดข้อความ ประเภท และ Address ที่เชื่อมต่อกับ PLC เพื่อรองรับการเกิดสัญญาณเตือน



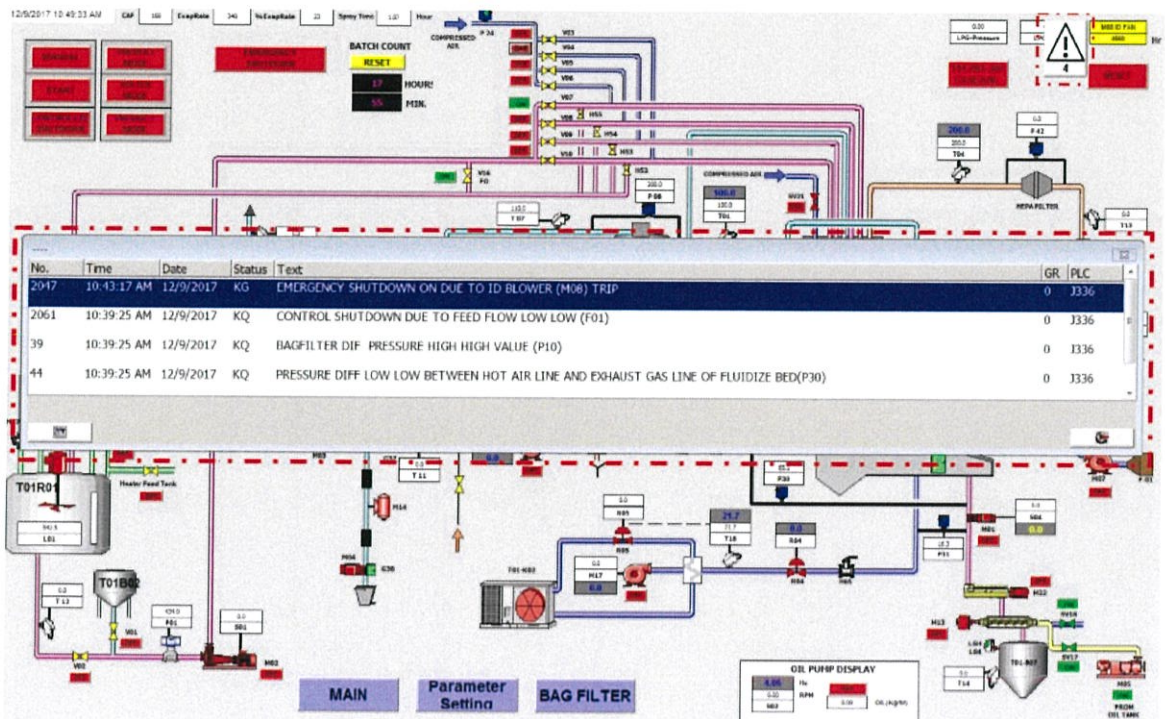
ภาพที่ 3.27 การสร้างแท็กของข้อความแจ้งเตือน

### 3.3.2 การเขียนโปรแกรมให้เกิดสัญญาณเตือน

นำแอดเดรสที่ได้จากการสร้างข้อความเตือนจาก WinCC flexible มาใช้เป็นบิตเอาต์พุตเมื่อเกิดเงื่อนไขที่ทำให้เกิดสัญญาณเตือนจะไปเซตบิตเตือนนั้นเป็น 1 และจะแสดงข้อความบนหน้ากราฟิก



ภาพที่ 3.28 แผนภาพแลตเตอร์การเกิดสัญญาณเตือน



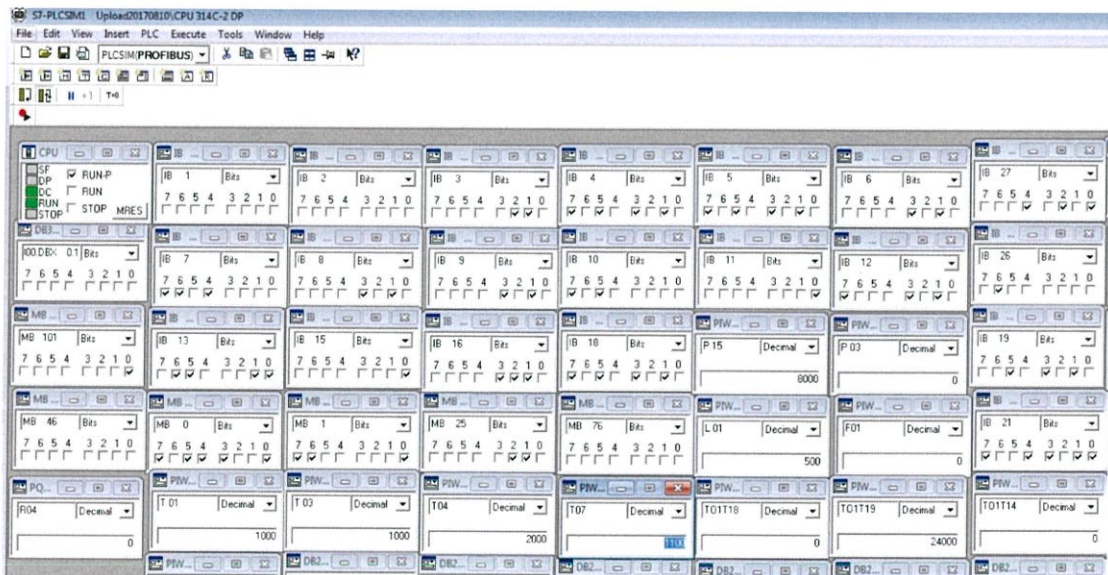
ภาพที่ 3.29 ข้อความแจ้งเตือนบนหน้ากราฟิก

## บทที่ 4 ผลการดำเนินงาน

การปรับปรุงกระบวนการอบแห้งแบบพ่นฝอยเป็นการดำเนินการภายใต้ความต้องการของลูกค้า เมื่อเสร็จสิ้นการดำเนินการจะต้องมีการทดสอบกับลูกค้า เพื่อตรวจสอบความถูกต้องก่อนเรียกว่าการ FAT (Factory Acceptance Test) โดยหากมีปัญหาเกิดขึ้น หรือมีความเข้าใจที่ไม่ตรงกัน จะสามารถทำการแก้ไขได้ก่อนการ SAT (Site Acceptance Test) คือ เมื่อติดตั้งที่ไซต์งานแล้วต้องมีการทดสอบอีกครั้ง เพื่อยืนยันว่ากระบวนการสามารถทำงานได้อย่างถูกต้อง

### 4.1 ทดสอบการทำงานและการแสดงผลของอุปกรณ์ต่างๆ

ทดสอบการทำงานและการแสดงผลการควบคุมด้วย PLC ตามที่ออกแบบโปรแกรมไว้ โดยการ FAT จะจำลองสัญญาณอินพุตและสังเกตสัญญาณเอาต์พุตจาก PLC Simulator ส่วนการ SAT จะใช้สัญญาณจากอุปกรณ์ที่ต่อเข้ากับ PLC โดยไม่อาศัยการจำลอง



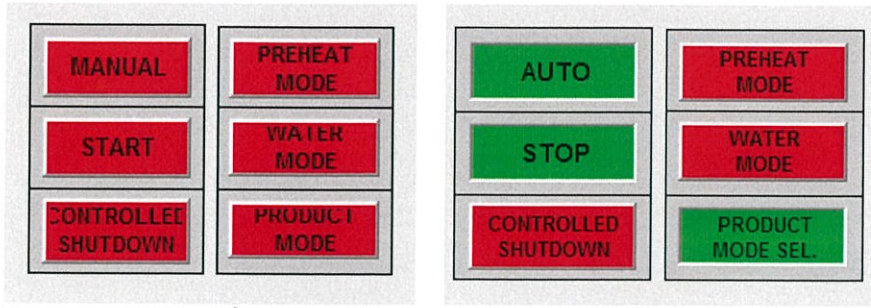
ภาพที่ 4.1 การจำลองสัญญาณขาเข้าบน PLC simulator

#### 4.1.1 ทดสอบการทำแสดงสถานะของปั๊มกด

ทดสอบปั๊มกดโดยทำการกดปุ่มเพื่อดูการเปลี่ยนแปลง เช่น ปั๊มเลือกให้ทำงานในกระบวนการ Product Mode เมื่อกดปุ่ม ข้อความและสีจะเปลี่ยนไปตามที่ตั้งค่าไว้

หากปั๊มเป็นสีแดงและแสดงคำว่า PRODUCT MODE หมายถึงการไม่กดปั๊ม

หากปั๊มเป็นสีเขียวและแสดงคำว่า PRODUCT MODE SEL หมายถึงการกดปั๊ม และระบบกำลังทำงานในกระบวนการ Product Mode

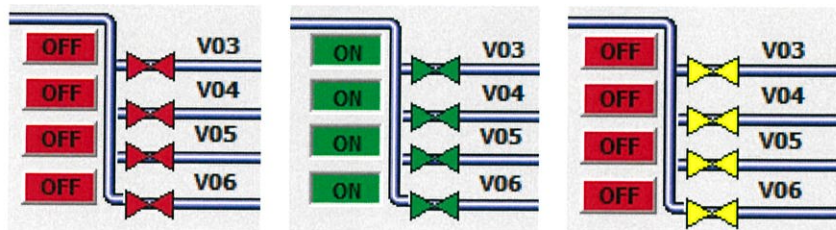


ภาพที่ 4.2 การแสดงฟังก์ชันการทำงานของปั๊มกด

#### 4.1.2 ทดสอบการแสดงผลสถานะของวาล์วเปิดปิด

การแสดงผลของวาล์วเปิดปิดประกอบด้วยชื่อแท็ก สัญลักษณ์วาล์วที่แสดงสถานะสีปั๊มควบคุมการเปิดปิดเมื่อทำงานในโหมด Manual

- หากวาล์วปิด วาล์วและปั๊มกดจะแสดงสีแดง
- หากวาล์วเปิด ปั๊มกดและวาล์วจะแสดงสีเขียว
- หากไม่มีการเชื่อมต่อกับ PLC จะแสดงสีเริ่มต้นคือสีเหลือง

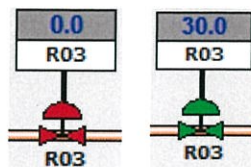


ภาพที่ 4.3 การแสดงผลสถานะของวาล์วเปิดปิด

#### 4.1.3 ทดสอบการแสดงผลสถานะของวาล์วควบคุม

การแสดงผลวาล์วควบคุม ประกอบด้วยชื่อแท็ก สัญลักษณ์วาล์วที่แสดงสถานะสี และส่วนรับค่าการสั่งการเปิดวาล์วเป็นเปอร์เซ็นต์ 0-100% โดยหาก

- หากวาล์วเปิดน้อยกว่า 2% จะแสดงสีแดง แสดงถึงวาล์วปิด
- หากวาล์วเปิดมากกว่า 2% จะแสดงสีเขียว แสดงถึงวาล์วเปิด

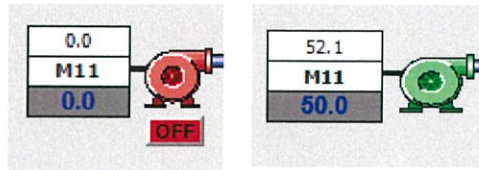


ภาพที่ 4.4 การแสดงผลสถานะของวาล์วควบคุม

#### 4.1.4 ทดสอบการแสดงผลสถานะของมอเตอร์ควบคุม

การแสดงผลมอเตอร์ควบคุมประกอบด้วย ชื่อแท็ก สัญลักษณ์มอเตอร์ ปั๊มควบคุมการเปิดปิดเมื่อทำงานในโหมด Manual มีส่วนรับค่าการสั่งการเปิดมอเตอร์เป็นเปอร์เซ็นต์ 0-100% และส่วนแสดงผลความเร็วของมอเตอร์เป็นเปอร์เซ็นต์เช่นกัน โดยเมื่อทำงานในโหมดอัตโนมัติ ปั๊มควบคุมการเปิดปิดมอเตอร์จะหายไป สามารถสั่งการเปิดมอเตอร์ได้เป็นเปอร์เซ็นต์ 0-100% หากเกิน

ขอบเขตจะแสดงข้อความเตือน และสามารถแสดงผลค่าความเร็วรอบของมอเตอร์ได้ตามที่สเกลไว้ในโปรแกรม PLC



ภาพที่ 4.5 การแสดงผลมอเตอร์ควบคุม



ภาพที่ 4.6 ข้อความแจ้งเตือนเมื่อใส่ค่าที่เกินขอบเขต

#### 4.1.4 ทดสอบการแสดงผลสถานะของเซนเซอร์

การแสดงผลของเซนเซอร์ ประกอบด้วยชื่อแท็ก สัญลักณ์เซนเซอร์ และส่วนแสดงผลการตรวจจับ เมื่ออุปกรณ์วัดตรวจจับได้จะแสดงสีเขียว และเมื่อตรวจจับไม่ได้จะแสดงสีแดงกะพริบ



ภาพที่ 4.7 การแสดงผลสถานะเซนเซอร์

#### 4.1.5 ทดสอบการแสดงผลสถานะของทรานสมิตเตอร์

การแสดงผลของทรานสมิตเตอร์ประกอบด้วย ชื่อแท็ก สัญลักณ์ทรานสมิตเตอร์ ส่วนแสดงผลค่าวัด ส่วนรับและแสดงผลค่าเป้าหมาย (Set point) สำหรับเครื่องมื่อวัดที่มีการควบคุมด้วยตัวควบคุม PID

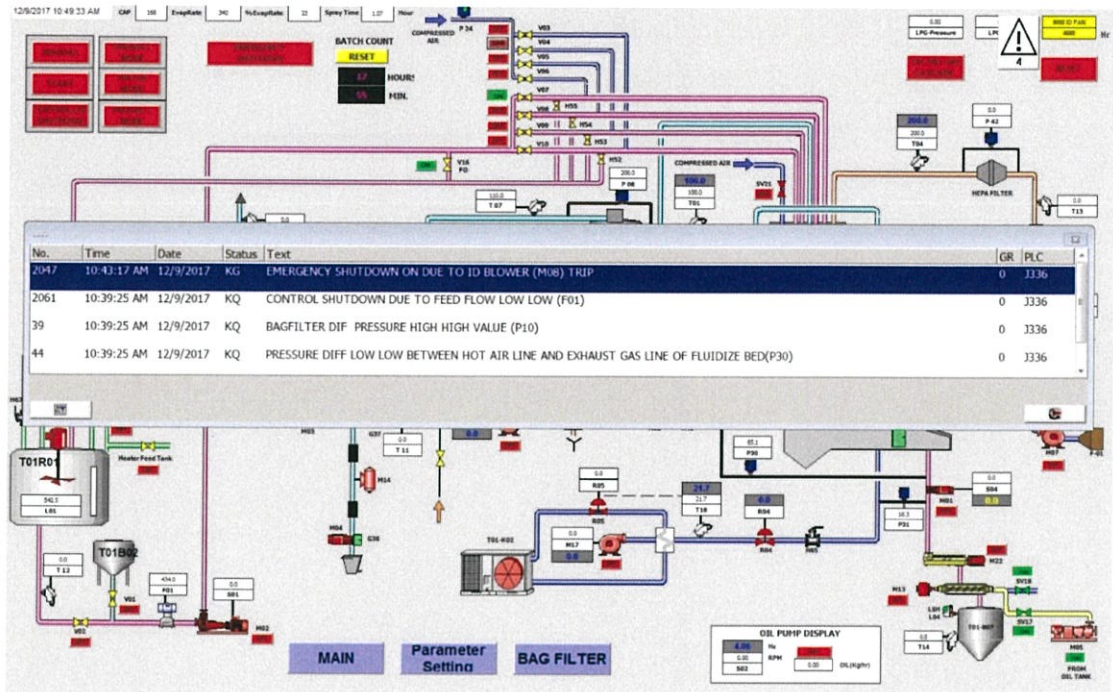


ภาพที่ 4.8 การแสดงผลของทรานสมิตเตอร์

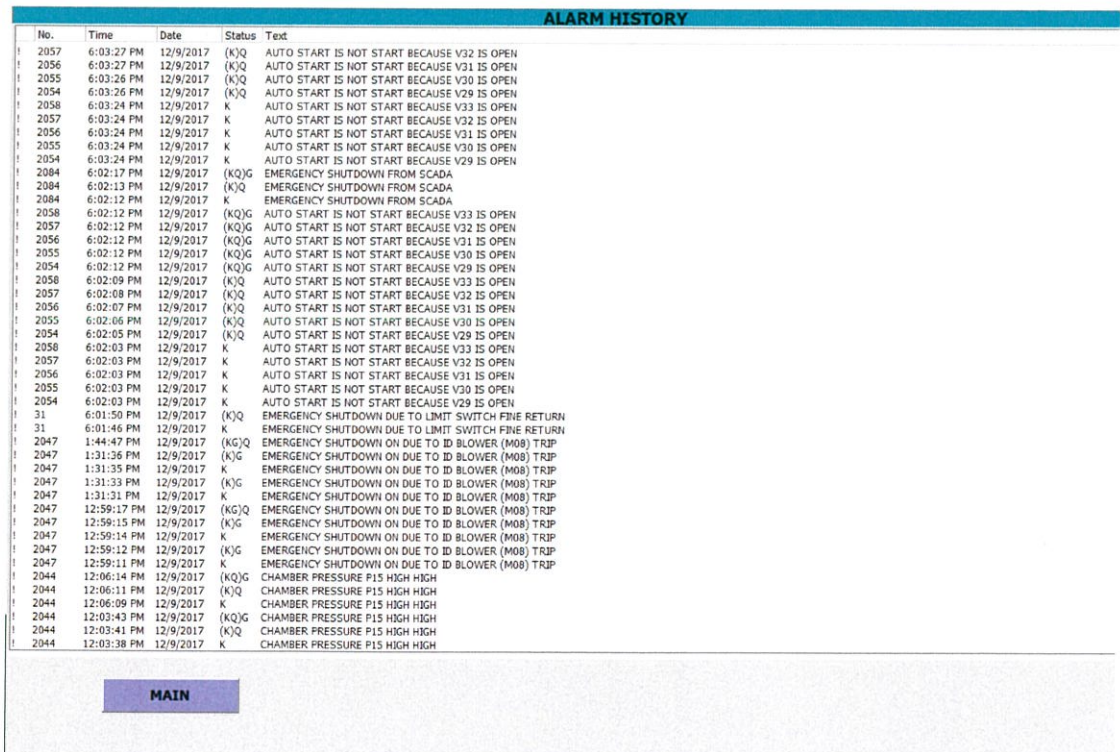
## 4.2 ทดสอบการแจ้งเตือนเมื่อเกิดข้อผิดพลาดกับอุปกรณ์

เมื่อเกิดข้อผิดพลาดขึ้นกับอุปกรณ์ จะปรากฏหน้าต่างแจ้งเตือนดังภาพที่ 4.9 ที่แสดงข้อความรายละเอียดความผิดพลาดที่เกิดขึ้นกับอุปกรณ์ให้ใช้งานทราบ และอุปกรณ์ตัวที่เกิดความผิดพลาดจะเปลี่ยนสีเป็นการกะพริบเพื่อให้ผู้ใช้งานเห็นได้ชัดเจน เพื่อให้ผู้ใช้งานสามารถแก้ไขความผิดพลาด

ได้ในเวลาอันสั้น และสามารถย้อนหลังดูการประวัติแจ้งเตือนทั้งหมดที่เกิดขึ้นได้ในหน้า Alarm History ดังภาพที่ 4.10



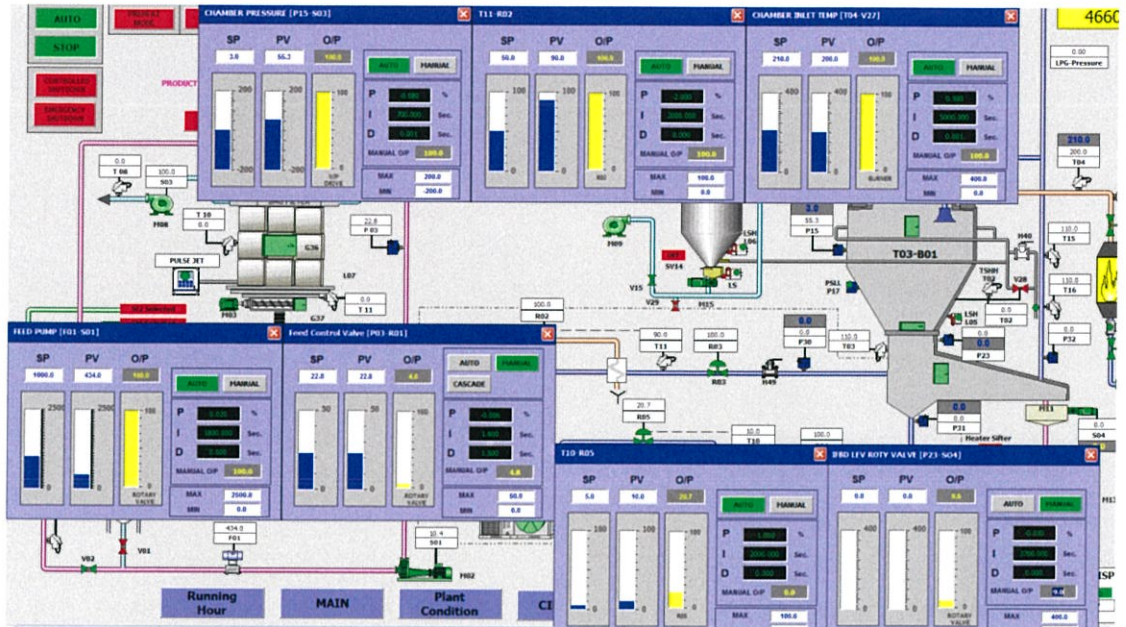
ภาพที่ 4.9 หน้าต่างแจ้งเตือนเมื่อเกิดข้อผิดพลาดกับอุปกรณ์



ภาพที่ 4.10 ประวัติการแจ้งเตือน

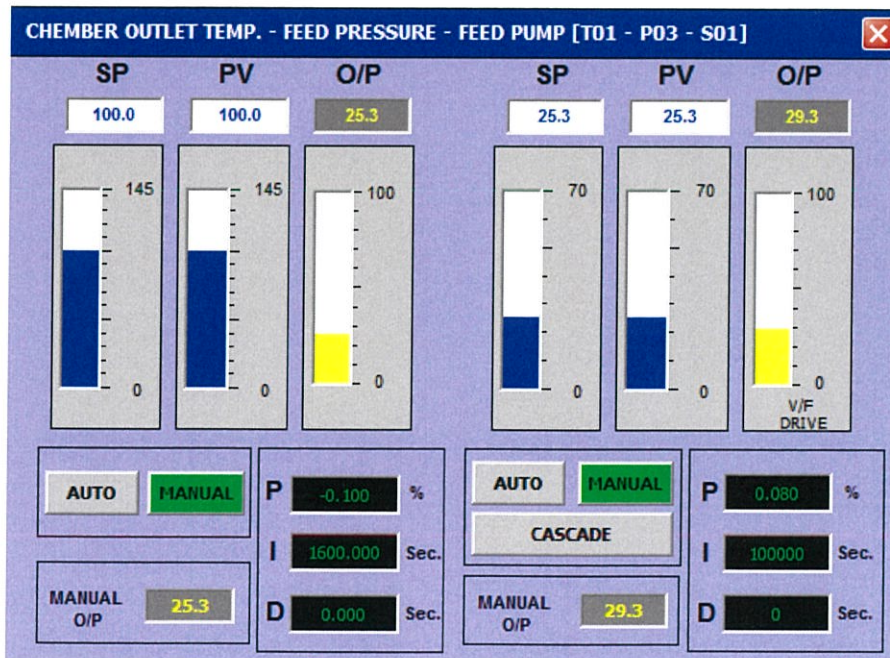
### 4.3 ทดสอบฟังก์ชันการกดปุ่มเรียกหน้าต่าง Faceplate

ทดสอบปุ่มกดเรียกหน้าต่าง Faceplate โดยคลิกที่อุปกรณ์ที่ได้มีการเชื่อมปุ่มกดไว้ หากวงเมาส์บริเวณที่เชื่อมปุ่มกดจะปรากฏเป็นรูปสายฟ้า แสดงถึงการเชื่อมโยง เมื่อกดแต่ละปุ่มเรียกจะปรากฏหน้าต่าง Faceplate ตรงตามที่เชื่อมโยงกับปุ่มไว้ดังภาพที่ 4.11



ภาพที่ 4.11 หน้าต่าง Faceplate ที่ถูกเรียกขึ้นมาจากปุ่มกด

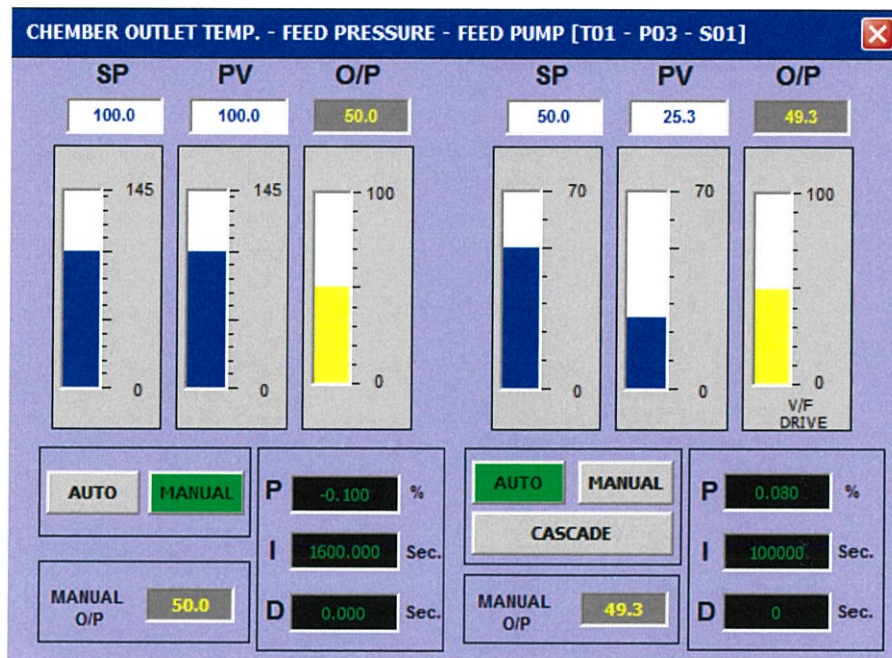
### 4.4 ทดสอบการควบคุมด้วยตัวควบคุมพีเอ็ด



ภาพที่ 4.12 การทำงานในโหมด Manual ของการควบคุมแบบ Cascade

ทดสอบการทำงานด้วยตัวควบคุม PID ของสองลูบที่สัมพันธ์กันมีการควบคุมแบบ Cascade เมื่อทำงานในโหมด Manual ดังภาพที่ 4.12 ฝั่งซ้ายคือลูบบน ฝั่งขวาคือลูบล่าง

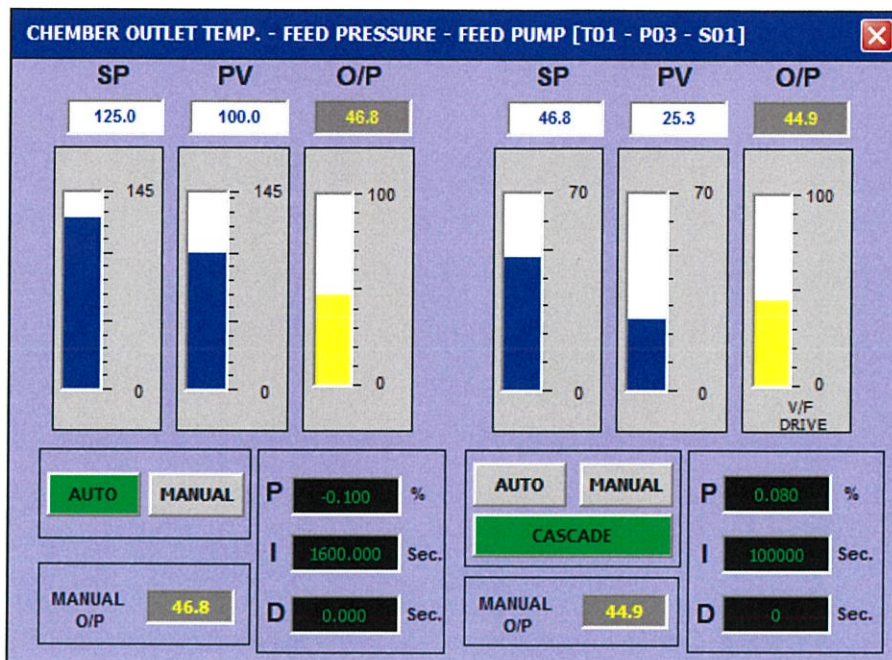
- 1) ค่า Output ต้องมีค่าเท่ากับค่า Manual output โดยสามารถกำหนดค่า Output ได้ตามต้องการ แต่ไม่สามารถเปลี่ยนได้หากทำงานในโหมด Auto
- 2) ค่าเป้าหมาย (SP) ต้องมีค่าเท่ากับค่ากระบวนการ (PV) ป้องกันการเปลี่ยนแปลงอย่างกะทันหันหากเปลี่ยนโหมดการควบคุมจาก Manual เป็น Auto
- 3) ค่า Output ของลูบบน ต้องมีค่าเท่ากับค่าเป้าหมาย (SP) ของลูบล่าง ป้องกันการเปลี่ยนแปลงอย่างกะทันหันหากเปลี่ยนโหมดการควบคุมเป็น Cascade
- 4) การแสดงผลในรูปแบบขีดระดับและการแสดงผลเป็นตัวเลขเป็นไปในทางเดียวกัน



ภาพที่ 4.13 การทำงานในโหมดอัตโนมัติของการควบคุมแบบ Cascade

ทดสอบการทำงานในโหมดอัตโนมัติ ดังภาพที่ 4.13

- 1) ลูบล่างมีค่า P มากกว่า 0 เมื่อค่าเป้าหมาย (SP) มากกว่าค่ากระบวนการ (PV) จะทำให้ค่า Output สูงขึ้น และเมื่อค่าเป้าหมาย (SP) น้อยกว่าค่ากระบวนการ (PV) จะทำให้ค่า Output น้อยลง
- 2) ค่า Output ของลูบบน ต้องมีค่าเท่ากับค่าเป้าหมาย (SP) ของลูบล่าง
- 3) สามารถเปลี่ยนแปลงค่าเป้าหมาย (SP) ได้หากทำงานในโหมด Auto และตัวควบคุม PID จะทำให้ค่ากระบวนการ (PV) มีค่าเข้าใกล้ค่าเป้าหมาย (SP)



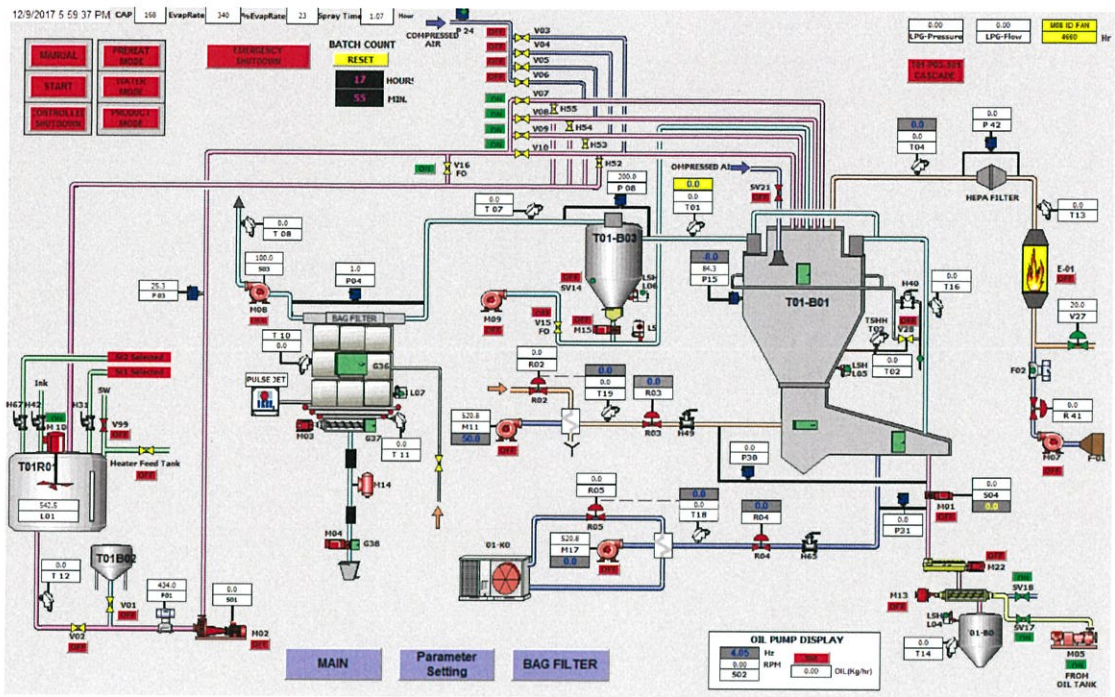
ภาพที่ 4.14 การควบคุมในโหมด Cascade

ทดสอบการทำงานในโหมดควบคุมแบบ Cascade ดังภาพที่ 4.14 ทั้งนี้ค่าพารามิเตอร์ต่างๆ ของตัวควบคุม PID สามารถกำหนดได้โดยผู้ใช้งาน

- 1) เมื่อมีการควบคุมแบบ Cascade ลูปล่างและลูบบนต้องทำงานในโหมดอัตโนมัติ
- 2) ค่า Output ของลูบบนจะเป็นค่าเป้าหมาย (SP) ของลูปล่าง
- 3) ลูบบนมีค่า P น้อยกว่า 0 เมื่อค่าเป้าหมาย (SP) มากกว่าค่ากระบวนการ (PV) จะทำให้ค่า Output น้อยลง และเมื่อค่าเป้าหมาย (SP) น้อยกว่าค่ากระบวนการ (PV) จะทำให้ค่า Output สูงขึ้น

#### 4.5 การแสดงผลหน้าจอ (Screen)

กราฟิกหน้ากระบวนการเป็นหน้ากระบวนการทำงานทั้งหมดของระบบ แสดงท่อและอุปกรณ์ ทั้งหมดตามแบบ P&ID ผู้ใช้งานจะใช้งานหน้านี้เป็นหลัก และมีปุ่มกดเชื่อมโยงไปยังหน้าจอย่อยต่างๆ ของระบบ โดยหน้ากระบวนการประกอบด้วยปุ่มกดสั่งเปิดปิดการทำงานของอุปกรณ์ ปุ่มกดเลือกโหมดการทำงาน การแจ้งเตือนเมื่อเกิดความผิดปกติขึ้นกับอุปกรณ์ การตั้งค่าการควบคุมด้วย PID โดยการเรียกหน้าต่าง Faceplate แสดงค่าวัดจากเครื่องมือวัด และสถานการณ์ทำงานของอุปกรณ์ทุกตัว



ภาพที่ 4.15 กราฟิกหน้ากระบวนกร

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 5.1 สรุปผลการดำเนินงาน

จากการดำเนินการปรับปรุงกระบวนการอบแห้งแบบพ่นฝอย โดยการออกแบบระบบควบคุมด้วยพีแอลซีเชื่อมโยงกับการแสดงผลบนอุปกรณ์ HMI ผ่านระบบสกาตา ทำให้ผู้ใช้งานสามารถสังเกตการทำงาน of กระบวนการผ่านหน้าจอที่แสดงสถานะการทำงานของอุปกรณ์ต่างๆ สามารถออกแบบการผลิตได้ผ่านการตั้งค่าพารามิเตอร์ที่นำไปใช้ควบคุมกระบวนการได้ผ่านส่วนรับและแสดงผลข้อมูล อีกทั้งทราบสาเหตุเมื่อเกิดความผิดปกติขึ้นกับกระบวนการจากข้อความแจ้งเตือน ผลลัพธ์การดำเนินการคือ กระบวนการอบแห้งแบบพ่นฝอยมีการทำงานเป็นลำดับ สามารถทำงานได้อย่างต่อเนื่อง และได้ผลิตภัณฑ์ที่มีคุณภาพตรงตามที่คุณภาพตรงตามที่คุณใช้งานต้องการ

#### 5.2 ปัญหาและแนวทางแก้ไข

##### 5.2.1 ปัญหาที่พบ

1. เนื่องจากเป็นการปรับปรุงกระบวนการที่มีการควบคุมด้วยพีแอลซีอยู่เดิม จึงต้องเริ่มจากการศึกษาโปรแกรมควบคุมเดิมก่อนเริ่มการแก้ไข ซึ่งโปรแกรมเดิมมีความซับซ้อนทำให้ใช้เวลาทำงานนาน
2. เป็นการปรับปรุงกระบวนการตามความต้องการของลูกค้า แต่ลูกค้ามีความต้องการที่ไม่แน่นอน ทำให้ต้องแก้งานบ่อยครั้ง เสี่ยงต่อความผิดพลาดที่อาจเกิดขึ้นได้
3. ขณะทดสอบกับกระบวนการผลิตจริงที่ไซต์งาน จะมีปัญหาหน้างานเกิดขึ้นจากหลายๆ ส่วน เช่น อุปกรณ์เสีย ทำให้ใช้เวลาทดสอบนานขึ้นเนื่องจากรอซ่อม หรือลูกค้าขอแก้ไขลำดับการทำงานเนื่องจากผลิตภัณฑ์ที่ได้ไม่เป็นตามที่คาดหวัง เป็นต้น

##### 5.2.2 แนวทางแก้ไข

1. เริ่มทำการศึกษาโปรแกรมควบคุมทันทีที่ได้รับมอบหมาย สอบถามและขอคำแนะนำจากผู้เฒ่าเมื่อมีข้อสงสัย
2. ติดต่อกับลูกค้าอยู่เสมอเพื่อยืนยันความถูกต้อง และสำรองข้อมูลทุกครั้งเมื่อมีการแก้ไข
3. เมื่อมีปัญหาเกิดขึ้นต้องแก้ไขได้ภายในระยะเวลาการ SAT (Site Acceptance Test) เพื่อให้มั่นใจได้ว่ากระบวนการสามารถทำงานได้อย่างถูกต้อง

#### 5.3 ข้อเสนอแนะ

เมื่อมีการทดสอบกระบวนการที่หน้างาน จะทำงานภายใต้ความกดดันโดยต้องทำให้กระบวนการถูกต้องตามที่ลูกค้าต้องการและหากมีปัญหาเกิดขึ้นต้องแก้ไขได้อย่างถูกต้องในเวลาอันสั้น ดังนั้นต้องรอบคอบและคำนึงถึงความปลอดภัย โดยก่อนสั่งการทำงานของอุปกรณ์จากห้องควบคุมต้องบอกกับผู้ที่อยู่ใกล้กับอุปกรณ์นั้นๆก่อน และการปฏิบัติงานสหกิจศึกษาเป็นเวลาหนึ่งภาคการศึกษา ถือเป็นระยะเวลาที่ไม่ยาวนานนักควรเก็บเกี่ยวประสบการณ์ให้ได้มากที่สุด และมีความใฝ่รู้อยู่เสมอ

## เอกสารอ้างอิง

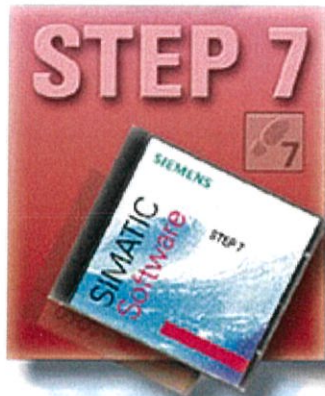
- [1] Masters, K.(Keith) (1991). *Spray drying handbook*. Great Britain: The bath press,Avon.
- [2] ณรงค์ ตันชีวะวงศ์. (2542). *ระบบ PLC*. กรุงเทพมหานคร: สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น)
- [3] สราวุฒิ ศิริวงษ์. (2548). *คล่องแคล่ว PLC*. กรุงเทพมหานคร: จุฬาลงกรณ์มหาวิทยาลัย
- [4] สุทธิพงษ์ เสนาธิ. (2549). *เรียนรู้การสร้างระบบ SCADA ด้วย GENESIS32*. กรุงเทพมหานคร: บริษัท อีดีเอ อินสทรูเมนต์ แอนด์ ซิสเต็ม จำกัด
- [5] ภาษาที่ใช้ในการเขียนโปรแกรม. สืบค้นวันที่ 20 พฤศจิกายน,2560,จากเว็บไซต์:  
<http://autocentrated.com/index.php/79-2013-05-13-08-21-38/2013-05-27-07-22-55/93-iec1131-3>
- [6] อีเทอร์เน็ต. สืบค้นวันที่ 24 พฤศจิกายน,2560,จากเว็บไซต์:  
<https://sites.google.com/site/kruarthit021159/xi-thex-r-net-ethernet>
- [7] ภาคผนวก ก
- [8] ภาคผนวก ข

# ภาคผนวก

ภาคผนวก ก

SIMATIC STEP3 Manual

SIMATIC



## Introduction to STEP 7

[What's New in STEP 7?](#)  
[Overview of STEP 7](#)  
[Objects and Object Hierarchy](#)  
[Operating Philosophy](#)

## Configuring and Programming

[Introduction to Configuring Hardware](#)  
[Configuring the Distributed I/O \(DP\)](#)  
[Networking Stations within a Project](#)  
[Basic Procedure for Creating Logic Blocks](#)  
Language Description: [STL](#), [LAD](#), [FBD](#)  
Help on [OBs](#), [SFBs/SFCs](#)  
[More than One User Editing Projects](#)

## Testing and Debugging

[Introduction to Testing with Variable Tables](#)  
[Testing Using Program Status](#)  
[Diagnosing Hardware and Troubleshooting](#)

## SIMATIC on the Internet

[SIMATIC Support/Service](#)  
[SIMATIC on the Internet](#)

ภาคผนวก ข

WinCC flexible 2008 manual

# SIEMENS

SIMATIC HMI

WinCC flexible 2008  
Compact / Standard / Advanced

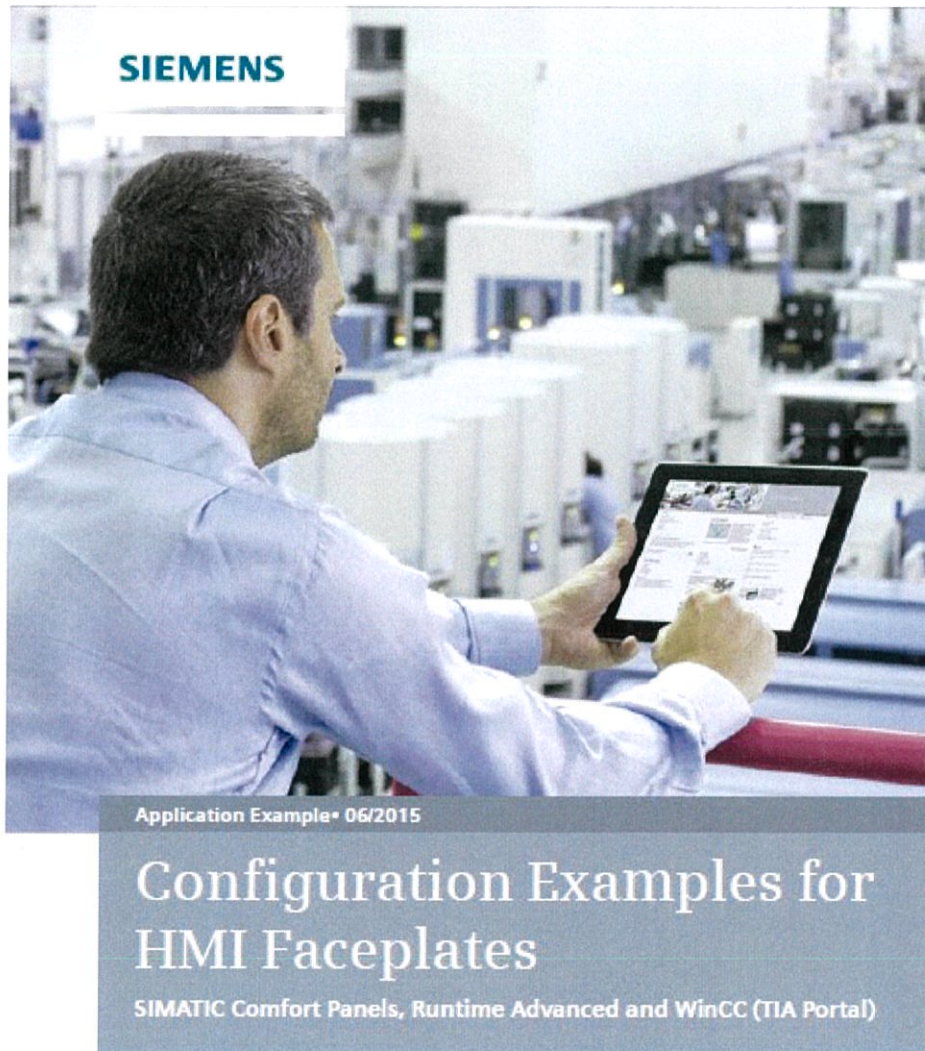
User's Manual

This manual is part of the documentation package  
with the order number 6AV6691-1AB01-3AB0.

07/2008  
ASE01024750-02

Preface	
Introduction to WinCC flexible	1
WinCC flexible Engineering System	2
Working with projects	3
Working with Tags	4
Creating Screens	5
Creating an Alarm System	6
Working with a connection	7
Structure of a recipe management system	8
Logging and displaying tags	9
Working with reports	10
User administration	11
System functions and runtime scripting	12
Structure of Multilingual Projects	13
Project documentation	14
Mobile Wireless	15
Planning jobs	16
Managing project versions	17
Logging changes	18
Transfer	19
Integration of WinCC flexible in STEP 7	20
Appendix	21

## Configuration Examples for HMI Faceplates



<https://support.industry.siemens.com/cs/ww/en/68014632>

## ประวัติผู้เขียน

ชื่อ-นามสกุล : นางสาวนันทกานติ แพงวิเศษ  
วัน เดือน ปีเกิด : 19 มกราคม พ.ศ.2539  
ภูมิลำเนา : 684/21 ถนนหลวงแพ่ง แขวงทับยาว เขตลาดกระบัง กรุงเทพมหานคร 10520

อีเมลล์ : pwmonthakanti@gmail.com

ประวัติการศึกษา : ระดับมัธยมศึกษาตอนต้น  
โรงเรียนพรตพิทยพยัต จังหวัด กรุงเทพมหานคร  
: ระดับมัธยมศึกษาตอนปลาย  
โรงเรียนพรตพิทยพยัต จังหวัด กรุงเทพมหานคร  
: ระดับปริญญาตรี  
ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประวัติการทำงาน : มิถุนายน – กรกฎาคม พ.ศ. 2560  
นักศึกษาฝึกงาน แผนก Electrical & Instrument  
บริษัท ไทยนิปปอน สตีล แอนด์ ซูมิคิน เอ็นจิเนียริง แอนด์ คอนสตรัคชั่น  
คอร์ปอเรชั่น จำกัด  
: สิงหาคม – พฤศจิกายน พ.ศ. 2560  
นักศึกษาสหกิจศึกษา แผนก Instrument & Control Engineering  
บริษัท อินส์ไพร์เทค จำกัด