



รายงานสหกิจศึกษาฉบับสมบูรณ์

การใช้งาน LOGO!8 ในระบบควบคุมบ้านอัจฉริยะ
SIMATIC LOGO!8 for Smart Home

นายศรายุทธ์ ทองอูน

ภาควิชาวิศวกรรมการวัดคุมและควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560



รายงานสหกิจศึกษาฉบับสมบูรณ์

การใช้งาน LOGO!8 ในระบบควบคุมบ้านอัจฉริยะ
SIMATIC LOGO!8 for Smart Home

นายศรายุทธ์ ทองอู่่น

ภาควิชาวิศวกรรมการวัดคุมและควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ชื่อโครงการสหกิจศึกษา การใช้งาน LOGO!8 ในระบบควบคุมบ้านอัจฉริยะ

ชื่อ-สกุล นักศึกษา นายศรยุทธ์ ทองอ่อน

คณะ วิศวกรรมศาสตร์ ภาควิชา วิศวกรรมการวัดและควบคุม

ชื่อ-สกุล อาจารย์นิเทศ รศ.ดร.สุพรรณ กุลพาณิชย์

ดร.นภศุล วงษ์วานิช

ชื่อ-สกุล ผู้นิเทศงาน คุณฉันทิพย์ บุญอาจ

ชื่อสถานประกอบการ บริษัท ซีเมนส์ (ประเทศไทย) จำกัด

บทคัดย่อ

โครงการนี้ได้รับการสนับสนุนจาก บริษัท ซีเมนส์ (ประเทศไทย) จำกัด แผนก Digital Factory / Process Industries & Drives ให้ศึกษาเกี่ยวกับ “การใช้งาน SIMATIC LOGO!8 ในระบบควบคุมบ้านอัจฉริยะ” โดยนำตัวควบคุม SIMATIC LOGO!8 มาประยุกต์ใช้กับระบบควบคุมอัตโนมัติ ในการกำหนดการทำงานหรือเวลาการเปิดหรือปิดของอุปกรณ์ไฟฟ้า ซึ่งจะมีโหมดการทำงานหลักๆ 2 โหมด คือ โหมดอัตโนมัติและโหมดแมนนวล เพื่อให้ประหยัดการใช้พลังงาน อีกทั้งยังอำนวยความสะดวกสบาย ในการควบคุมอุปกรณ์ไฟฟ้าทั้งภายในบ้านและนอกบ้าน โดยใช้โปรแกรม LOGO! Soft Comfort ในการเขียนโปรแกรมควบคุม ให้เปิดหรือปิดอุปกรณ์ไฟฟ้าตามเวลาที่ได้โปรแกรมและต่อรวมกับ HMI เพื่อใช้ในการเปลี่ยนแปลงแก้ไขข้อมูลและตรวจสอบสถานะการทำงานของ LOGO!8 อีกทั้งยังสามารถแสดงสถานะการทำงานของเครื่องใช้ไฟฟ้าต่าง ๆ พร้อมทั้งใช้สั่งการทำงานของ LOGO!8 ที่ควบคุมอุปกรณ์ไฟฟ้า

Research Title: SIMATIC LOGO!8 for Smart Home

Student Intern Name: Mr. Sarayut Thong-un

Faculty: Engineer

Department: Instrumentation and Control Engineering

Advisor Name: Assoc. Prof. Dr. Suphan Gulpanich

Dr. Nopasool Wongvanich

Mentor Name: Mr. Chantid Boun-aht

Company: Siemens Limited Thailand

ABSTRACT

This cooperative educational report is supported by SIEMENS Limited Thailand, in the division of Digital Factory/Process Industries & Drives, which assigned to study about SIMATIC LOGO!8 for Smart Home by using SIMATIC LOGO!8 for automatic control system in scheduling electrical equipment. There are two main modes of operation: automatic mode and manual mode. To save energy It also facilitates the control of electrical equipment both inside and outside the home by using LOGO! Soft Comfort to programming control to turn on or turn off electrical equipment and connect it to the HMI for modification and monitoring of SIMATIC LOGO!8 statuses.

กิตติกรรมประกาศ

รายงานสหกิจศึกษาฉบับสมบูรณ์นี้ สำเร็จลุล่วงด้วยดีนั้น เนื่องจากการสนับสนุนจากบุคคลหลายฝ่าย ที่ให้คำปรึกษาแนะแนวทาง ทำให้ปริญญาานิพนธ์ฉบับนี้บรรลุเป้าหมายตามวัตถุประสงค์ ซึ่งทำให้เกิดการเรียนรู้และสร้างประสบการณ์จากการปฏิบัติงานจริง

ขอขอบพระคุณ รศ.ดร.สุพรรณ กุลพาณิชย์ และ ดร.นภศูล วงษ์วานิช อาจารย์นิเทศที่ให้คำแนะนำและข้อคิดเห็น รวมถึงช่วยตรวจทาน แก้ไขข้อบกพร่องต่าง ๆ อันเป็นประโยชน์อย่างยิ่งตลอดมา ทำให้รายงานสหกิจศึกษาฉบับนี้เสร็จสมบูรณ์ ขอขอบพระคุณ บริษัท ซีเมนส์ (ประเทศไทย) จำกัด ที่มอบโอกาสในการเข้ามาทำสหกิจศึกษา ประจำปีการศึกษา 2560 นี้ รวมถึง คุณฉันทิพย์ บุญอาจ ซึ่งเป็นผู้ดูแลและควบคุมการทำสหกิจศึกษา อีกทั้งยังคอยเอื้อเฟื้ออุปกรณ์ และเครื่องมือต่าง ๆ ตลอดการทำโครงการ และขอขอบคุณพี่ ๆ ในแผนกที่คอยให้คำปรึกษาในการทำงานจนทำให้ปริญญาานิพนธ์ฉบับนี้เสร็จสมบูรณ์

นายศรายุทธ์ ทองอุ่น

สารบัญ

| | หน้า |
|--|------|
| บทคัดย่อภาษาไทย | I |
| บทคัดย่อภาษาอังกฤษ | II |
| กิตติกรรมประกาศ | III |
| สารบัญ | IV |
| สารบัญรูป | VI |
| สารบัญตาราง | X |
| | |
| บทที่ 1 บทนำ | |
| 1.1 ความเป็นมาและความสำคัญ | 1 |
| 1.2 วัตถุประสงค์ของโครงการ | 2 |
| 1.3 ขอบเขตของโครงการ | 2 |
| 1.4 ขั้นตอนการศึกษา | 3 |
| 1.5 ประโยชน์ที่คาดว่าจะได้รับ | 3 |
| บทที่ 2 การเขียนโปรแกรมพื้นฐานกับเอกสารที่เกี่ยวข้อง | |
| 2.1 SIMATIC LOGO!8 Programming Concepts | 4 |
| 2.1.1 SIMATIC LOGO!8 Concepts | 4 |
| 2.1.2 Program Structure | 5 |
| 2.1.3 Basic Programming | 9 |
| 2.2 SIMATIC HMI Programming Concepts | 13 |
| 2.2.1 SIMATIC HMI Concepts | 13 |
| 2.2.2 Basic Programming | 13 |
| 2.3 Network and Communication | 21 |
| 2.3.1 Ethernet Protocol: OSI/ISO | 21 |
| 2.3.2 Ethernet Network: IEEE802.3 & IEEE802.11 | 28 |
| 2.4 มาตรฐานที่เกี่ยวข้อง | 33 |
| 2.4.1 EN 61000-6-1 | 33 |
| 2.4.2 EN 61000-6-2 | 34 |

สารบัญ (ต่อ)

| | หน้า |
|---|------|
| บทที่ 3 การออกแบบและการจำลอง | |
| 3.1 การออกแบบแปลนบ้าน | 35 |
| 3.1.1 การกำหนดสัญลักษณ์ | 35 |
| 3.1.2 วาดแปลนบ้านและจำลองการวางอุปกรณ์ภายในบ้านและนอกบ้าน | 36 |
| 3.2 ออกแบบ Functional ของ Smart Home | 37 |
| 3.3 การเขียนโปรแกรม LOGO!8 | 38 |
| 3.4 การออกแบบจอแสดงผล HMI | 57 |
| 3.4.1 การสร้าง Connections | 57 |
| 3.4.2 การสร้าง tag บนอุปกรณ์ HMI | 58 |
| 3.4.3 การสร้างกราฟฟิคแสดงผลและควบคุม | 59 |
| บทที่ 4 ผลการดำเนินงาน | |
| 4.1 การทดสอบ LOGO!8 | 66 |
| 4.2 การทดสอบหน้าจอ HMI | 67 |
| 4.3 การทดสอบการเชื่อมต่อระหว่าง LOGO!8 และ HMI | 73 |
| บทที่ 5 สรุปผล ปัญหาและข้อเสนอแนะ | |
| 5.1 สรุปผลการดำเนินงาน | 75 |
| 5.2 ปัญหาและแนวทางแก้ไข | 75 |
| 5.3 ข้อเสนอแนะ | 76 |
| เอกสารอ้างอิง | 77 |
| ภาคผนวก | 78 |
| ประวัติผู้เขียน | 86 |

สารบัญรูป

| รูปที่ | หน้า |
|--|------|
| 2.1 LOGO! แบบมีหน้าจอแสดงผล | 4 |
| 2.2 การต่อขยายโมดูล LOGO! ได้มากที่สุด | 5 |
| 2.3 การเขียนภาษา FBD | 5 |
| 2.4 การเขียนภาษา LAD | 6 |
| 2.5 การเขียนภาษา UDF | 6 |
| 2.6 หน้าต่างโปรแกรมใน Diagram Mode | 7 |
| 2.7 หน้าต่างโปรแกรมใน Network Mode | 8 |
| 2.8 การเพิ่มอุปกรณ์ LOGO!8 | 9 |
| 2.9 การตั้งค่าของ LOGO!8 | 10 |
| 2.10 การใช้งานแบบ Drag & Drop | 11 |
| 2.11 การปรับเปลี่ยนพารามิเตอร์ของฟังก์ชัน | 11 |
| 2.12 การ Simulate ของฟังก์ชัน | 12 |
| 2.13 การเลือกพอร์ท LAN ที่เราเชื่อมต่อเข้ากับคอมพิวเตอร์ | 12 |
| 2.14 การ Download โปรแกรมลง LOGO!8 | 13 |
| 2.15 การเพิ่ม HMI โดยใช้ Wizard | 14 |
| 2.16 การเลือก PLC ที่เชื่อมต่อกับ HMI | 15 |
| 2.17 การกำหนดสีพื้นหลังและแสดงวันที่และเวลา | 15 |
| 2.18 การเลือกชนิดของ Alarm ที่ต้องการแสดง | 16 |
| 2.19 การเพิ่มหน้าจอของ HMI | 17 |
| 2.20 การเลือกฟังก์ชันเพิ่มเติมของจอ | 18 |
| 2.21 การเลือกปุ่ม System | 19 |
| 2.22 หน้าต่างแสดงปุ่มกดลิงค์ไปยังหน้าจออื่นๆ | 19 |
| 2.23 การสร้างปุ่มกด | 20 |
| 2.24 การตั้งค่าปุ่มกดและการเลือก tag | 20 |
| 2.25 OSI Model 7 Layer | 22 |
| 2.26 OSI Model Upper/Lower Layer | 22 |
| 2.27 การส่งข้อมูลบน Physical Layer | 23 |
| 2.28 Frame การส่งข้อมูล | 23 |

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|---|------|
| 2.29 การส่งข้อมูลข้ามเครือข่าย | 25 |
| 2.30 การเชื่อมต่อกับ Upper Layer | 25 |
| 2.31 การใช้งานผ่าน Services Telnet | 26 |
| 2.32 กลไก Three-Way Handshake | 26 |
| 2.33 การติดต่อระหว่างต้นทางกับปลายทาง | 27 |
| 2.34 โครงสร้างของเฟรม 802.3 | 30 |
| 3.1 แพลนบ้านชั้น 1 | 36 |
| 3.2 แพลนบ้านชั้น 2 | 36 |
| 3.3 Flowchart ฟังก์ชันของบ้านชั้นที่ 1 | 37 |
| 3.4 Flowchart ฟังก์ชันของบ้านชั้นที่ 2 | 38 |
| 3.5 การเพิ่มอุปกรณ์ | 39 |
| 3.6 แสดงหลังจากการเพิ่ม LOGO!8 2 ตัว | 39 |
| 3.7 ตั้งค่าการเปลี่ยนชื่อและ IP | 40 |
| 3.8 การตั้งค่า I/O ของอุปกรณ์ | 40 |
| 3.9 การกำหนดชื่อ I/O ของอุปกรณ์ | 41 |
| 3.10 โปรแกรมของบ้านชั้นที่ 1 ส่วนที่ 1 | 42 |
| 3.11 โปรแกรมของบ้านชั้นที่ 1 ส่วนที่ 2 | 43 |
| 3.12 โปรแกรมของบ้านชั้นที่ 2 | 44 |
| 3.13 ฟังก์ชันที่เหมือนกันของโปรแกรม | 45 |
| 3.14 การเพิ่ม UDF diagram | 46 |
| 3.15 เขียนโปรแกรมที่เหมือนกันลง UDF | 46 |
| 3.16 การตั้งชื่อ I/O และตั้ง password | 47 |
| 3.17 การเพิ่มพารามิเตอร์ที่ต้องการ | 47 |
| 3.18 วิธีนำ UDF มาใช้งาน | 48 |
| 3.19 การนำฟังก์ชัน UDF มาแก้ไขโปรแกรมเดิม | 49 |
| 3.20 โปรแกรมใหม่ที่ใช้ฟังก์ชัน UDF ของบ้านชั้นที่ 1 ส่วนที่ 1 | 50 |
| 3.21 โปรแกรมใหม่ที่ใช้ฟังก์ชัน UDF ของบ้านชั้นที่ 1 ส่วนที่ 2 | 51 |
| 3.22 โปรแกรมของบ้านชั้นที่ 2 | 53 |

สารบัญรูป (ต่อ)

| รูปที่ | | หน้า |
|--------|--|------|
| 3.23 | ขั้นตอนการจับคู่พารามิเตอร์กับ VM | 55 |
| 3.24 | ขั้นตอนการจับคู่พารามิเตอร์กับ VM (ต่อ) | 55 |
| 3.25 | การสร้างการเชื่อมต่อ HMI และ LOGO!8 | 58 |
| 3.26 | การสร้าง Tag บนอุปกรณ์ HMI | 59 |
| 3.27 | หน้าจอหลัก | 59 |
| 3.28 | การออกแบบหน้าต่างของชั้นที่ 1 | 60 |
| 3.29 | หน้าต่างตั้งค่าของชั้นที่ 1 | 61 |
| 3.30 | หน้าต่างการตั้งค่าไฟรอบริ้วบ้าน | 61 |
| 3.31 | หน้าต่างการตั้งค่าวันและเวลาสปริงเกิล | 62 |
| 3.32 | หน้าต่างการตั้งค่าที่จอดรถ | 62 |
| 3.33 | หน้าต่างการตั้งค่าการเปิดปิดของเครื่องปรับอากาศ | 63 |
| 3.34 | หน้าต่างแสดงสถานะอุปกรณ์ไฟฟ้าของบ้านชั้นที่ 1 | 63 |
| 3.35 | หน้าต่างการตั้งค่าของบ้านชั้นที่ 2 | 64 |
| 3.36 | หน้าต่างการตั้งค่าการเปิดปิดไฟระเบียงบ้าน | 64 |
| 3.37 | หน้าต่างการตั้งค่าเครื่องปรับอากาศของบ้านชั้นที่ 2 | 65 |
| 4.1 | การ Simulator ของ LOGO! Soft Comfort | 66 |
| 4.2 | การ Simulator 2 ตัวพร้อมกัน | 67 |
| 4.3 | หน้าต่างการตั้งค่าที่จอดรถ | 67 |
| 4.4 | พารามิเตอร์ในโปรแกรม LOGO!8 | 68 |
| 4.5 | หน้าต่างแสดงสถานะของอุปกรณ์ไฟฟ้าของชั้นที่ 1 | 68 |
| 4.6 | หน้าต่างการตั้งค่าต่าง ๆ ของบ้านชั้นที่ 1 | 69 |
| 4.7 | หน้าต่างการตั้งค่าไฟรอบริ้วบ้าน | 69 |
| 4.8 | หน้าต่างการตั้งค่าไฟที่จอดรถ | 70 |
| 4.9 | หน้าต่างการตั้งค่าสปริงเกิล | 70 |
| 4.10 | หน้าต่างการตั้งค่าเครื่องปรับอากาศของบ้านชั้นที่ 1 | 71 |
| 4.11 | หน้าต่างแสดงสถานะอุปกรณ์ไฟฟ้าของบ้านชั้นที่ 2 | 71 |
| 4.12 | หน้าต่างการตั้งค่าต่าง ๆ ของบ้านชั้นที่ 2 | 72 |
| 4.13 | หน้าต่างการตั้งค่าเครื่องปรับอากาศของบ้านชั้นที่ 2 | 72 |

สารบัญรูป (ต่อ)

| รูปที่ | | หน้า |
|--------|--|------|
| 4.14 | หน้าต่างการตั้งค่าไฟระเบียงบ้าน | 73 |
| 4.15 | การเชื่อมต่อระหว่าง LOGO!8 ทั้ง 2 ตัวและ HMI | 73 |
| 4.16 | การจำลองการทำงานของอินพุตและเอาต์พุต | 74 |
| 4.17 | การใช้โทรศัพท์มือถือควบคุม HMI | 74 |

สารบัญตาราง

| ตารางที่ | หน้า |
|---|------|
| 1. การกำหนดสัญลักษณ์ในแปลนบ้าน | 35 |
| 2. ตารางแสดงหมายเลข Block ฟังก์ชันโปรแกรมของบ้านชั้นที่ 1 | 52 |
| 3. ตารางแสดงหมายเลข Block ฟังก์ชันโปรแกรมของบ้านชั้นที่ 2 | 54 |
| 4. Parameter VM Mapping ของบ้านชั้นที่ 1 | 56 |
| 5. Parameter VM Mapping ของบ้านชั้นที่ 2 | 57 |

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ซีเมนส์ เอจี (เบอร์ลิน, มิวนิค) เป็นบริษัทชั้นนำระดับโลกที่ยืนหยัดเพื่อสรรสร้างความเป็นเลิศในงานวิศวกรรม นวัตกรรม ด้วยความน่าเชื่อถือในระดับสากลมากกว่า 165 ปี บริษัทฯ มีสาขาครอบคลุมกว่า 200 ประเทศทั่วโลก ซึ่งมุ่งพัฒนาในการให้บริการทั้งงานด้านวิศวกรรมไฟฟ้า ระบบขับเคลื่อน และ ดิจิตอลไลเซชัน มีพนักงานประมาณ 357,000 คนทั่วโลก ที่กำลังพัฒนา ผลิตภัณฑ์ ออกแบบ และติดตั้งระบบงานซึ่งมีความซับซ้อน รวมถึงค้นหาโซลูชันที่ดีที่สุด ให้เหมาะสมกับความต้องการของลูกค้าที่มีความแตกต่างกัน

ซีเมนส์ (ประเทศไทย) ในประเทศไทย ซีเมนส์ได้มีส่วนร่วมในการเชื่อมโยงเศรษฐกิจและงานด้านเทคนิควิศวกรรมในประเทศไทยเป็นเวลากว่าหนึ่งศตวรรษ โดยเริ่มจากการเซ็นสัญญาแต่งตั้งตัวแทนกับบริษัท ปิกิริม แอนด์ โก เมื่อปี พ.ศ. 2443 บริษัท ซีเมนส์ จำกัด ก่อตั้งขึ้นในประเทศไทยในเดือนตุลาคม พ.ศ. 2538 เพื่อเป็นผู้ประกอบการอย่างเป็นทางการในลิขสิทธิ์ของซีเมนส์เพื่อนำเสนอสินค้า โซลูชัน และบริการในทุกกลุ่มธุรกิจหลักของซีเมนส์ โดยปัจจุบันมีพนักงานประมาณ 1,200 คน กิจกรรมที่สำคัญของบริษัท คือ การสร้างคุณค่าให้แก่ลูกค้าด้วยการปรับปรุงโรงงานต่างๆ และสถานประกอบการรวมถึงธุรกิจของลูกค้าให้เกิดการใช้พลังงานอย่างมีประสิทธิภาพ ช่วยเพิ่มผลผลิตและความยืดหยุ่นตลอดจนช่วยยกระดับคุณภาพชีวิตของผู้คน ตั้งแต่วันที่ 1 ตุลาคม 2557 เป็นต้นไป องค์กรของเราจะถูกปรับเปลี่ยนโดยลดทอนระดับ Sector และแบ่งธุรกิจออกเป็น 9 กลุ่มหลัก ดังนี้

- 1) Power and Gas
- 2) Wind Power and Renewables
- 3) Power Generation Services
- 4) Energy Management
- 5) Building Technologies
- 6) Mobility
- 7) Digital Factory
- 8) Process Industries & Drives
- 9) Healthcare

ในปัจจุบันประเทศไทยได้เริ่มมีการนำระบบ Automation มาใช้งานกับบ้านหรืออาคารมากขึ้น โดยเป็นการนำเทคโนโลยีมาใช้ควบคุมอุปกรณ์ต่างๆ เพื่อช่วยอำนวยความสะดวกและลดค่าใช้จ่ายด้าน

พลังงาน ทั้งนี้ราคาก็เป็นอีกปัจจัยหนึ่งที่ผู้ใช้งานตัดสินใจในการทำระบบ Automation ภายในบ้านหรืออาคาร นักศึกษาจึงได้นำ SIMATIC LOGO!8 ซึ่งมีราคาไม่แพงมากมาประยุกต์ใช้ร่วมกับ SIMATIC HMI จำลองระบบบ้านอัตโนมัติขึ้นมา เพื่อเป็นแนวทางการนำ SIMATIC LOGO!8 มาใช้งานกับระบบบ้านอัตโนมัติ (Smart Home) ให้กับผู้ที่ต้องการใช้งานหรือสนใจ ได้ศึกษาโปรแกรมของ SIMATIC LOGO!8 ที่นักศึกษาได้จัดทำระบบ Smart Home ขึ้นมา ซึ่ง Smart Home นั้นจะต้องมีคุณสมบัติ 3 ประการคือ

1. มี Smart Home Network คือระบบพื้นฐานของ Smart Home อาจเป็นการเดินสายหรือไร้สายก็ได้
2. มี Intelligent Control System คือ ระบบการควบคุมระบบอัจฉริยะที่มีความชาญฉลาด
3. มี Home Automation Device คือ อุปกรณ์เครื่องใช้ไฟฟ้าภายในบ้านที่สามารถเชื่อมโยงกันได้

สรุปคือความหมายของ Smart Home ได้ต้องมีโครงสร้าง 3 ส่วนคือเริ่มจากอุปกรณ์ที่เป็น Smart Device และอุปกรณ์เหล่านั้นต้องสามารถเชื่อมโยงเป็นเครือข่ายได้ที่เราเรียกว่า Smart Home Network และสุดท้ายคือต้องมีส่วนควบคุมหลักที่เป็นมันสมองของบ้านซึ่งเราสามารถเขียนโปรแกรมให้บ้านของเรามีความฉลาดแบบใดก็ได้ตามที่เรต้องการที่เราเรียกว่า Intelligent Control System เหล่านี้จึงจะประกอบรวมเรียกว่าเป็น Smart Home

นอกจากนี้นักศึกษายังได้จัดทำคู่มือการสอนพื้นฐาน LOGO!8 (Basic training LOGO!8) ที่ใช้ในการสอนผู้ใช้งานที่สนใจ และจัดทำ Quick Guide เป็นเอกสารที่ช่วยให้ใช้งาน Features ต่างๆ ได้ง่ายขึ้น เช่น การเชื่อมต่อกับอุปกรณ์อื่นๆ ที่มี Protocol เป็น Modbus TCP/IP ในการอ่านหรือเขียนข้อมูล การใช้ LOGO! Access Tool ดูค่าพารามิเตอร์ต่างๆ และทำ Data Logging เป็นต้น ในส่วนของการศึกษาเพิ่มเติม นักศึกษาได้ศึกษาการและเรียนรู้การใช้งาน HMI พื้นฐาน ที่นำมาเชื่อมต่อกับ LOGO!8 เพื่อใช้ตรวจสอบข้อมูล ค่าพารามิเตอร์ต่างๆ หรือเขียนข้อมูลในพารามิเตอร์ของ LOGO!8 ตามที่เราต้องการ

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาการใช้งาน SIMATIC LOGO!8 และประยุกต์ใช้กับอุปกรณ์อื่น
2. เพื่อศึกษาการใช้งาน SIMATIC HMI ในการเชื่อมต่อกับ LOGO!8
3. เป็นตัวอย่างระบบบ้านอัตโนมัติ ที่ประยุกต์ใช้ฟังก์ชันต่างๆ ของ LOGO!8 ให้กับผู้ที่สนใจนำไปประยุกต์ใช้งานได้ตามความต้องการ
4. จัดทำเอกสารคู่มือการสอนพื้นฐานและ Quick Guide Features ที่น่าสนใจให้กับผู้ใช้งาน

1.3 ขอบเขตของโครงการ

1. ศึกษาการใช้งานและการใช้ฟังก์ชันต่างๆ ของ SIMATIC LOGO!8

2. ศึกษาการใช้งาน SIMATIC HMI ในส่วนของการเขียนกราฟิกและโปรแกรมส่งข้อมูลระหว่าง LOGO!8 กับ HMI
3. จำลองระบบการทำงานระบบบ้านอัตโนมัติ

1.4 ขั้นตอนการศึกษา

1. ศึกษาการใช้งานฟังก์ชันของ SIMATIC LOGO!8 และการเชื่อมต่อกับ SIMATIC HMI
2. ศึกษาการใช้งาน SIMATIC HMI และการเชื่อมต่อกับ SIMATIC LOGO!8
3. ออกแบบแผนบ้านจำลอง
4. ออกแบบและกำหนดเงื่อนไขการทำงานของระบบไฟฟ้าทั้งภายในและภายนอกบ้าน
5. ออกแบบหน้าจอ SIMATIC HMI ที่ใช้สั่งระบบบ้านอัตโนมัติรวมทั้งการกำหนดพารามิเตอร์เวลาต่างๆ และการจำลองสถานการณ์
6. เขียนโปรแกรม SIMATIC LOGO!8 ควบคุมระบบไฟฟ้า และเขียนโปรแกรมหน้าจอ SIMATIC HMI ใช้สั่งระบบบ้านอัตโนมัติ
7. ดำเนินผลการทดลองและบันทึกผลการทดลอง
8. จัดทำเอกสารการใช้งานฟังก์ชัน SIMATIC LOGO!8 และ SIMATIC HMI
9. ส่งมอบให้กับทางบริษัท เพื่อนำไปใช้ประโยชน์ต่อไป

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ออกแบบระบบบ้านอัตโนมัติ
2. ได้ความรู้ความเข้าใจในการใช้งานฟังก์ชันต่างๆ ของ SIMATIC LOGO!8
3. ได้ใช้ฟังก์ชันพื้นฐานในการเขียนโปรแกรม SIMATIC HMI
4. ได้ออกแบบการจำลองระบบบ้านอัตโนมัติ
5. สามารถใช้ SIMATIC LOGO!8 สื่อสารกับจอ SIMATIC HMI และนำไปประยุกต์ใช้กับระบบบ้านอัตโนมัติ
6. สามารถใช้ SIMATIC HMI เปลี่ยนค่าพารามิเตอร์ของ SIMATIC LOGO!8 ได้ โดยไม่จำเป็นต้องเข้าไปแก้ไขในตัวโปรแกรมของ SIMATIC LOGO!8
7. ฝึกทำเอกสารคู่มือสื่อการสอน

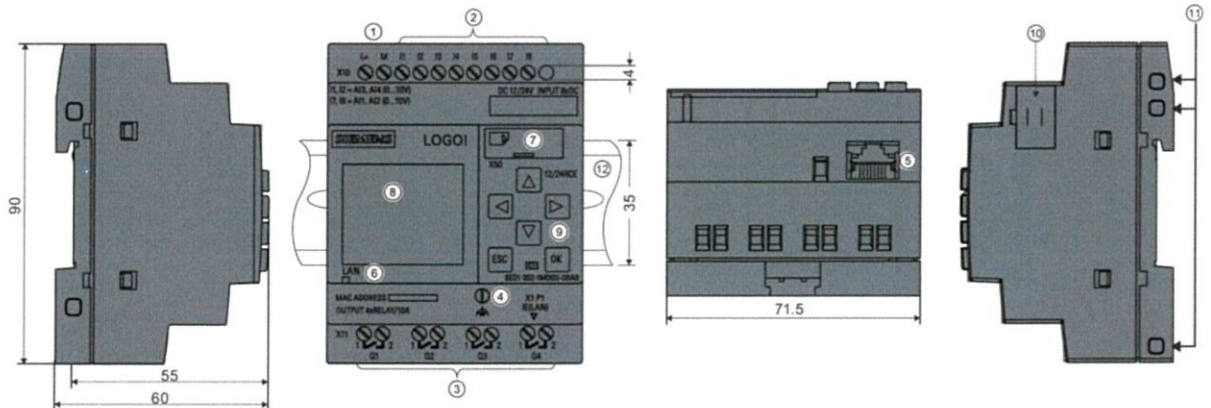
บทที่ 2

การเขียนโปรแกรมพื้นฐานและเอกสารที่เกี่ยวข้อง

2.1 SIMATIC LOGO!8 Programming Concepts

2.1.1 SIMATIC LOGO!8 Concept

SIMATIC LOGO!8 คือ PLC ขนาดเล็กที่สุดของ SIEMENS ซึ่ง LOGO! จะรวมฟังก์ชันที่ใช้ควบคุม , หน้าจอแสดงผล, ปุ่มกดการทำงาน, Timer Switch, Input/output, และฟังก์ชันพื้นฐานที่พร้อมใช้งานไว้ด้วยกัน ตัว LOGO! เองจะมี 2 แบบด้วยกันคือ แบบมีหน้าจอและไม่มีหน้าจอแสดงผล โดยทั้ง 2 แบบ จะมี Input และ Output ติดตั้งมาพร้อมกับโมดูล 8DI และ 4DO รูปที่ 2.1.1 เป็น LOGO! แบบมีหน้าจอแสดงผล



รูปที่ 2.1 LOGO! แบบมีหน้าจอแสดงผล

ส่วนประกอบของ LOGO!8 หลักๆ จะมีดังต่อไปนี้

1. Power Supply
2. Inputs
3. Output
4. สายดิน
5. พอร์ทที่ใช้ในการต่อ Ethernet
6. LED แสดงสถานะการเชื่อมต่อ Ethernet
7. ช่องใส่ Micro SD card
8. LCD
9. แผงปุ่มกดควบคุม
10. พอร์ทการต่อขยายของโมดูล

11. Sockets เสียบรางดิน

LOGO! สามารถต่อขยายโมดูลเพิ่มได้มากที่สุดคือ 24DI (Digital Input), 20DO (Digital Output), 8AI (Analog Input), และ 8AO (Analog Output)

| | | | | | | | | | | | | |
|-------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|--------------|--------------|--------------|--------------|
| I1 I8 | I9..I12 | I13..I16 | I17..I20 | I21..I24 | AI1, AI2 | AI3, AI4 | AI5, AI6 | AI7, AI8 | | | | |
| LOGO! Base Module | LOGO! DM8 | LOGO! DM8 | LOGO! DM8 | LOGO! DM8 | LOGO! AM2 | LOGO! AM2 | LOGO! AM2 | LOGO! AM2 | LOGO! AM2 AQ | LOGO! AM2 AQ | LOGO! AM2 AQ | LOGO! AM2 AQ |
| Q1..Q4 | Q5..Q8 | Q9..Q12 | Q13..Q16 | Q17..Q20 | | | | | AQ1, AQ2 | AQ3, AQ4 | AQ5, AQ6 | AQ7, AQ8 |

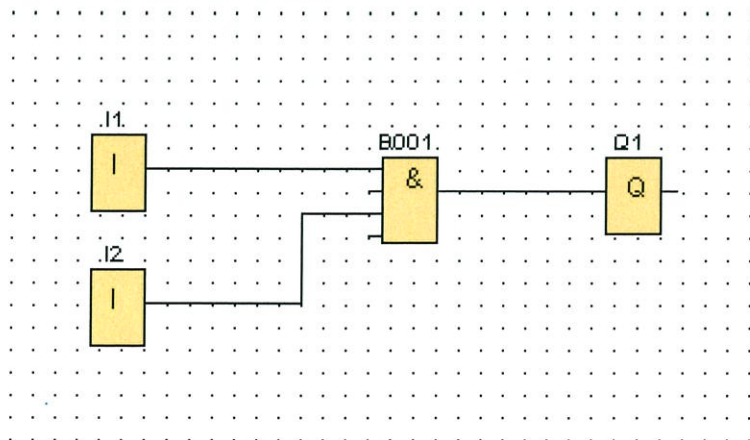
รูปที่ 2.2 การต่อขยายโมดูล LOGO! ได้มากที่สุด

การทำงานของ LOGO! มี 2 โหมดด้วยกันคือ 1) STOP Mode 2) RUN Mode ซึ่งมีหลักการทำงานดังนี้

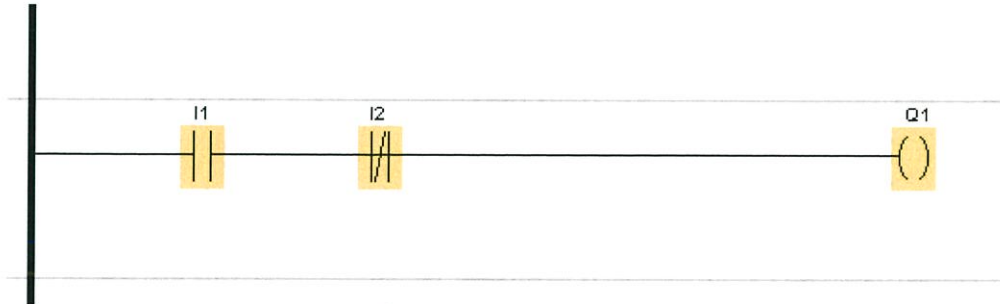
1. STOP Mode คือจะไม่ทำการประมวลผลโปรแกรม Input จะไม่ถูกอ่าน หน้าสัมผัสของรีเลย์จะเปิด หรือทรานซิสเตอร์จะถูกปิด โหมดนี้จะสามารถดาวน์โหลดโปรแกรมลง LOGO! ได้
2. RUN Mode คือ LOGO! จะทำการอ่านค่า Input และทำการประมวล Output ตามโปรแกรมที่เราที่เขียนไว้

2.1.2 Program Structure

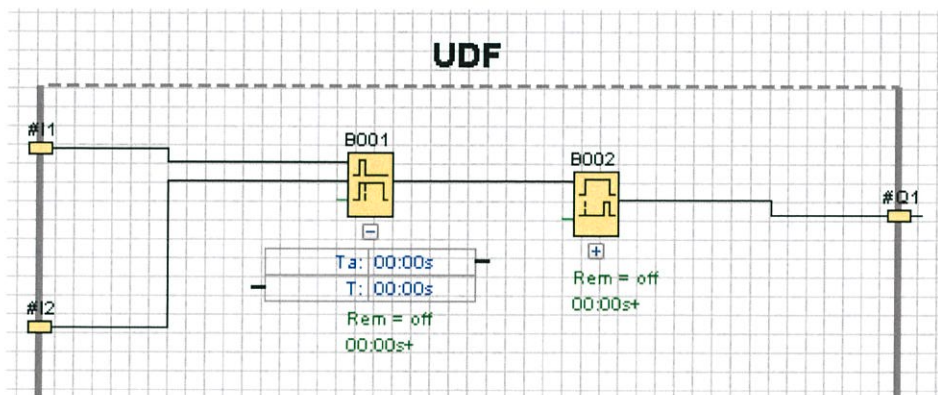
LOGO! ถูกพัฒนาให้เป็น Logic Module ที่สามารถเขียนโปรแกรมได้สะดวกต่อผู้ใช้ ซึ่งจะมีภาษาที่ใช้เขียนโปรแกรมอยู่ 3 ภาษาด้วยกัน 1) FBD (Function Block Diagram) ,2) LAD (Ladder Diagram) และ 3) UDF (User Define Function)



รูปที่ 2.3 การเขียนภาษา FBD



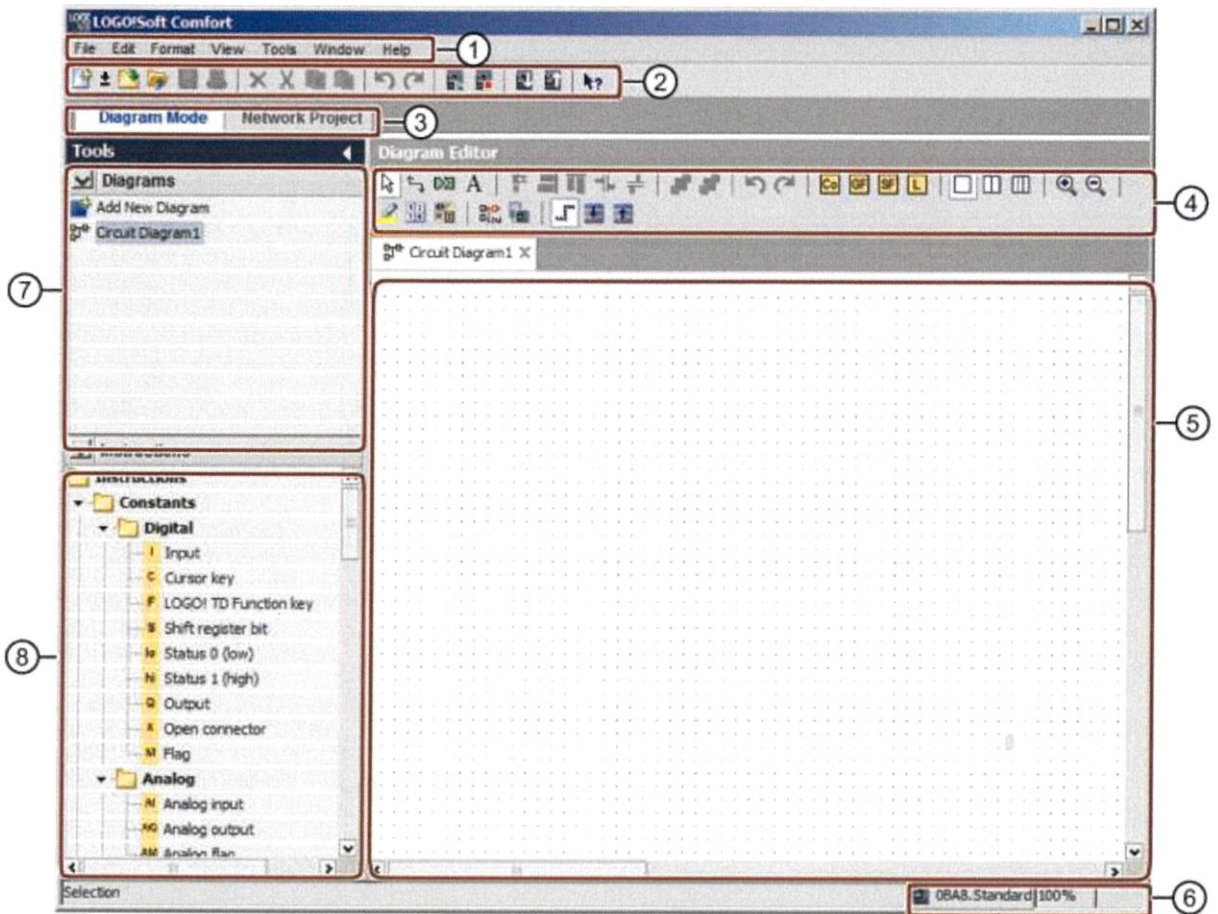
รูปที่ 2.4 การเขียนภาษา LAD



รูปที่ 2.5 การเขียนภาษา UDF

Software ที่ใช้ในการเขียนโปรแกรมของ LOGO!8 คือ LOGO! Soft Comfort ซึ่ง Software ถูกออกแบบมาให้ดูง่ายขึ้นและตั้งค่าได้รวดเร็ว ผู้ใช้สามารถพัฒนาโปรแกรมได้โดยการเลือกฟังก์ชัน และเชื่อมต่อแต่ละฟังก์ชันโดยการลากแล้ววาง ซึ่งจะมี 2 โหมดด้วยกันคือ

1. Diagram Mode เป็นโหมดที่ใช้เขียนโปรแกรม LOGO! เพียงตัวเดียวเท่านั้น โดยที่จะไม่สนใจ LOGO! ตัวอื่นถึงแม้จะต่อเป็นเน็ตเวิร์คอยู่ที่ตาม โหมดนี้สามารถทำเน็ตเวิร์คกับอุปกรณ์อื่นได้ แต่จะไม่สามารถทำการจับคู่เน็ตเวิร์คได้ หน้าต่าง Diagram Mode จะเป็นตามรูปที่ 2.6

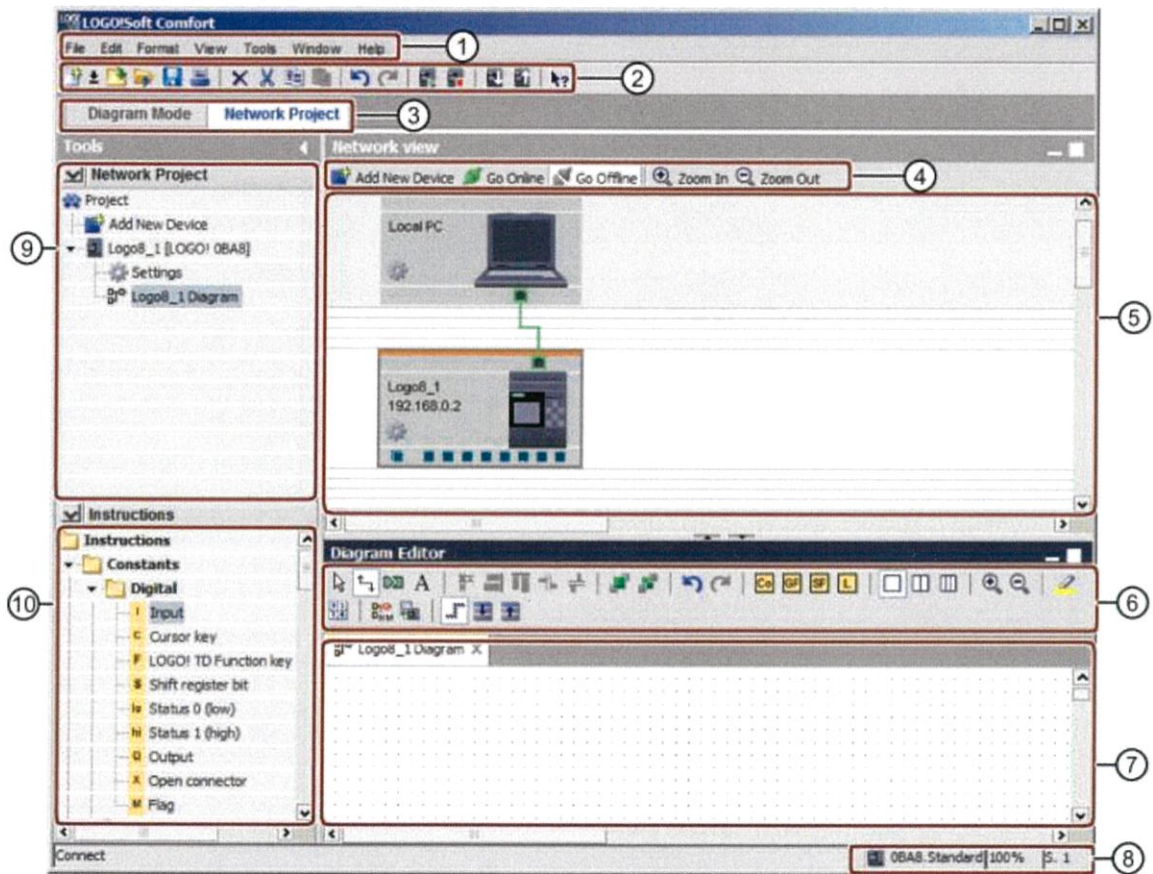


รูปที่ 2.6 หน้าต่างโปรแกรมใน Diagram Mode

ซึ่งจะมีส่วนประกอบของหน้าต่าง Diagram Mode จะมีดังนี้

1. แถบเมนูใช้เพื่อกำหนดการตั้งค่าต่างๆ ของ LOGO! และดาวน์โหลดหรืออัปโหลดโปรแกรมจาก LOGO!
2. แถบเครื่องมือมาตรฐาน เป็นคำสั่งที่ใช้งานบ่อยของโปรแกรม
3. แถบเปลี่ยนโหมด
4. แถบเครื่องมือที่ใช้ในการเขียนโปรแกรม
5. หน้าต่างในการเขียนโปรแกรม
6. แถบแสดงสถานะของคอนโทรลเลอร์
7. หน้าต่างไดอะแกรม
8. หน้าต่างคำสั่งที่ใช้เขียนโปรแกรม

- Network Mode เป็นโหมดที่ใช้เขียนโปรแกรม LOGO! หลายตัวรวมถึงใช้จับคู่เน็ตเวิร์คร่วมอุปกรณ์อื่นๆ เช่น S7 CPU, HMI, LOGO!8 ได้ หน้าต่างของ Network Mode จะมีหน้าต่างคล้ายคลึงกับ Diagram Mode เพียงแค่นี้ Network view เพิ่มเข้ามาให้เราสามารถจับคู่เน็ตเวิร์กกับอุปกรณ์อื่นๆ ได้ง่าย ตามรูปที่ 2.7



รูปที่ 2.7 หน้าต่างโปรแกรมใน Network Mode

ส่วนประกอบของหน้าต่างโปรแกรมใน Network Mode มีดังนี้

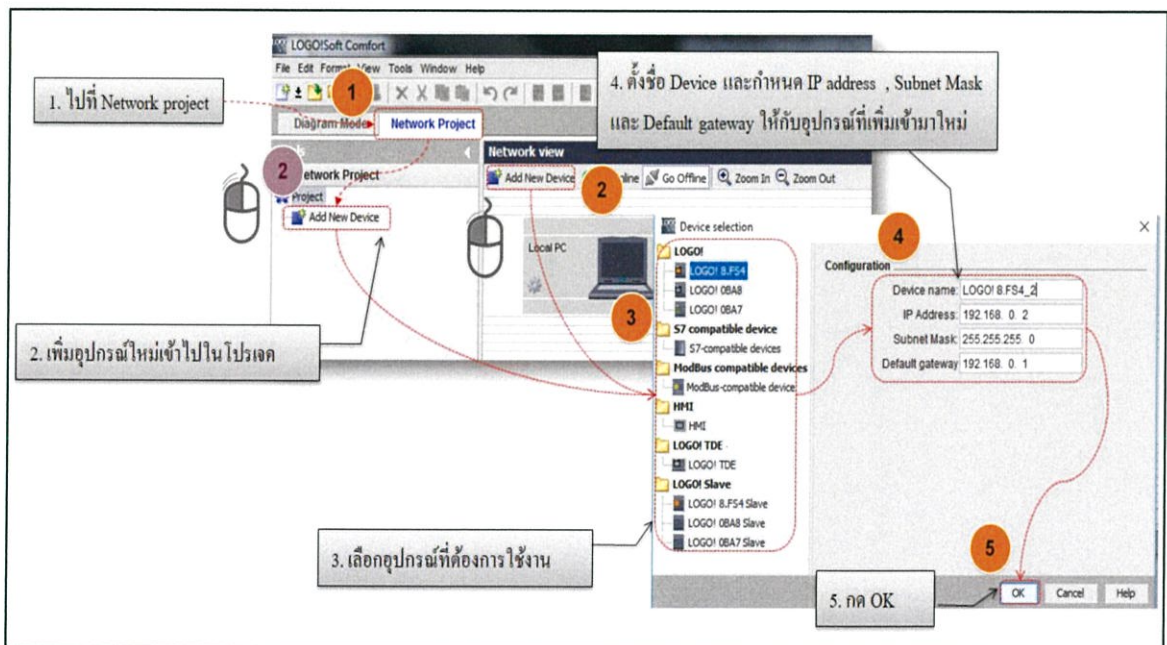
- แถบเมนูใช้เพื่อกำหนดการตั้งค่าต่างๆ ของ LOGO! และดาวน์โหลดหรืออัปโหลดโปรแกรมจาก LOGO!
- แถบเครื่องมือมาตรฐาน เป็นคำสั่งที่ใช้งานบ่อยของโปรแกรม
- แถบเปลี่ยนโหมด
- แถบเครื่องมือการทำงานของเน็ตเวิร์ค
- หน้าต่างแสดงการเชื่อมต่อเน็ตเวิร์ค

6. แถบเครื่องมือที่ใช้ในการเขียนโปรแกรม
7. หน้าต่างในการเขียนโปรแกรม
8. แถบแสดงสถานะของคอนโทรลเลอร์
9. หน้าต่างอุปกรณ์ที่มีในเน็ตเวิร์ค
10. หน้าต่างคำสั่งที่ใช้เขียนโปรแกรม

2.1.3 Basic Programming

ในการใช้งานของ Software LOGO! Soft Comfort โดยทั่วไปจะใช้ภาษา Function Block Diagram ซึ่งสามารถใช้งานได้ง่ายและสะดวกรวดเร็วเพียงแค่ Drag & Drop ฟังก์ชันที่อยู่หน้าต่างคำสั่งลงหน้าต่างเขียนโปรแกรมตามทีผู้ใช้งานต้องการ จากนั้นทำการเชื่อมต่อแต่ละฟังก์ชันเข้าด้วยกัน โดยมีขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 ทำการ Add New Device ขึ้นมาและตั้งชื่ออุปกรณ์และ IP Address ให้เรียบร้อย ซึ่งจะมีขั้นตอนตามรูปที่ 2.8



รูปที่ 2.8 การเพิ่มอุปกรณ์ LOGO!8

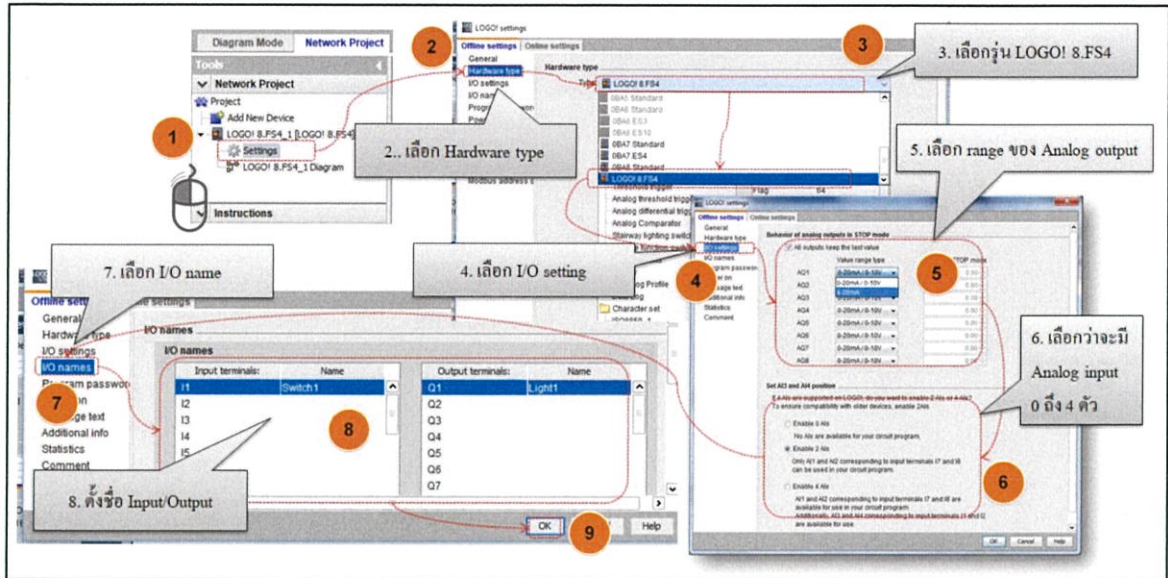
ขั้นตอนการเพิ่มอุปกรณ์มีขั้นตอนดังต่อไปนี้

1. ไปที่ Network Project
2. เพิ่มอุปกรณ์เข้าไปในโปรเจก
3. เลือก LOGO! ที่ต้องการใช้งาน

4. ตั้งชื่ออุปกรณ์และ IP Address ของอุปกรณ์

5. กด OK

ขั้นตอนที่ 2 เป็นขั้นตอนการตั้งค่าต่างๆ รวมทั้ง Input และ Output ของอุปกรณ์ตามรูปที่ 2.1.9

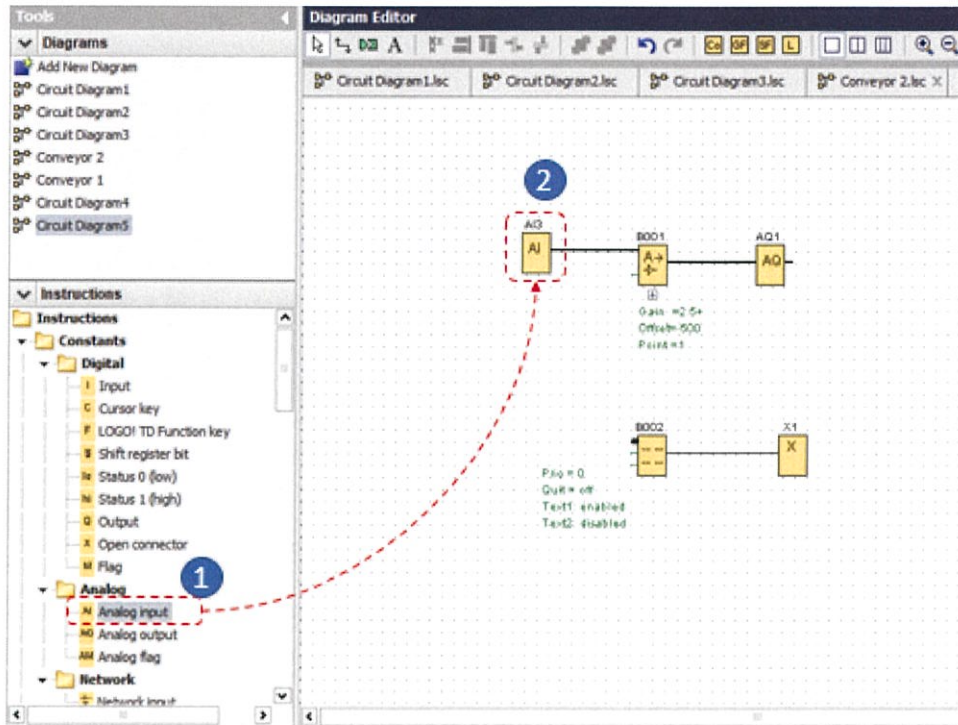


รูปที่ 2.9 การตั้งค่าของ LOGO!8

ขั้นตอนการตั้งค่าของ LOGO!8 มีขั้นตอนดังต่อไปนี้

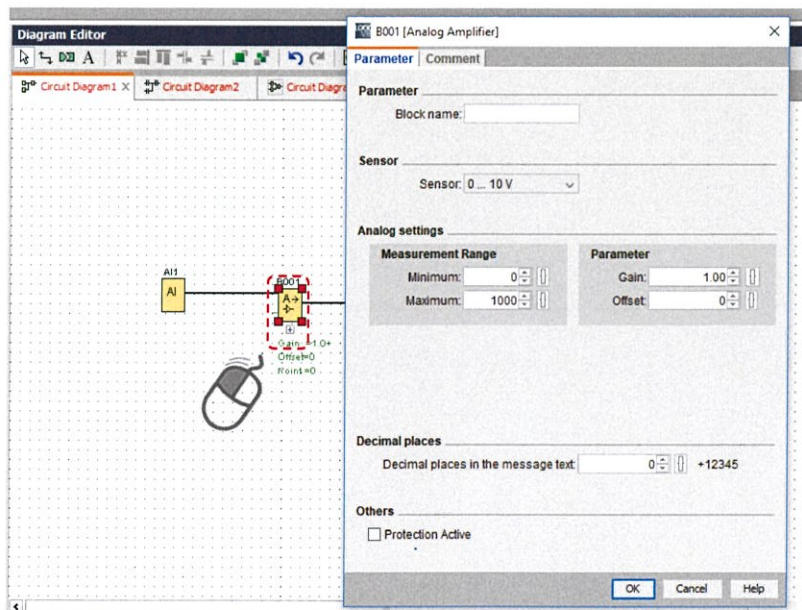
1. ไปที่ Setting ใน Network Project
2. เลือก Hardware Type
3. เลือกรุ่นของ LOGO! ให้ตรงกับที่ใช้งาน
4. ไปที่ I/O Setting
5. เลือก Range ของ Analog Output ให้เป็น 4-20 mA หรือ 0-20 mA/0-10 V
6. เลือกตั้งค่าให้มี Analog Input 0 – 4 ตัวที่โมดูล LOGO!
7. เลือก I/O Name
8. ตั้งชื่อให้กับ Input และ Output
9. กด OK เมื่อเสร็จการตั้งค่า

ขั้นตอนที่ 3 เป็นขั้นตอนในการเขียนโปรแกรมโดยใช้ภาษา Function Block Diagram ซึ่งเราสามารถลาก Function Block มาวางที่ หน้าต่าง Diagram ได้เลยตามรูปที่ 2.10



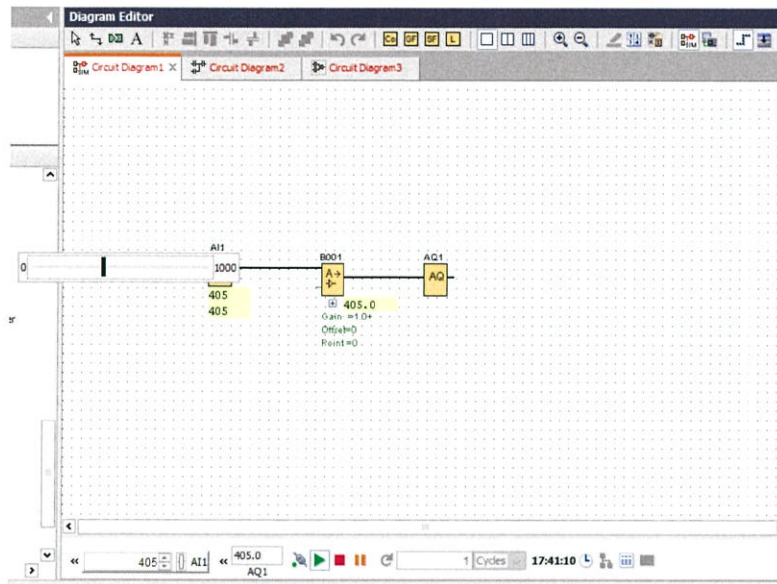
รูปที่ 2.10 การใช้งานแบบ Drag & Drop

ขั้นตอนที่ 4 การตั้งค่าพารามิเตอร์ ผู้ใช้สามารถตั้งค่าพารามิเตอร์ได้สะดวกและรวดเร็วเพียงแค่ดับเบิลคลิกที่ Function Block แล้วสามารถตั้งค่าพารามิเตอร์ได้เลยตามรูปที่ 2.11



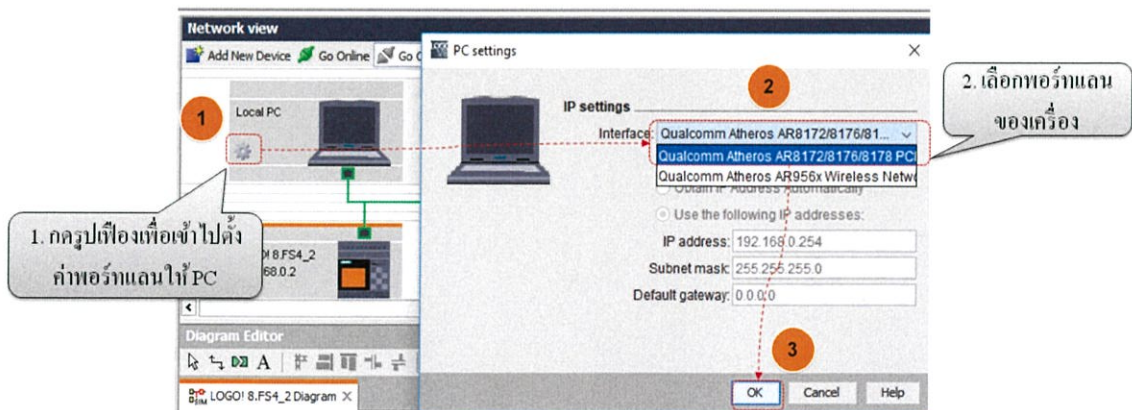
รูปที่ 2.11 การปรับเปลี่ยนพารามิเตอร์ของฟังก์ชัน

ขั้นตอนที่ 5 โปรแกรม LOGO! Soft Comfort สามารถจำลองและทดสอบโปรแกรมได้เลย เพื่อตรวจเช็คโปรแกรมก่อนนำไปดาวน์โหลด LOGO! จริง โดยกด Simulate (F3)



รูปที่ 2.12 การ Simulate ของฟังก์ชัน

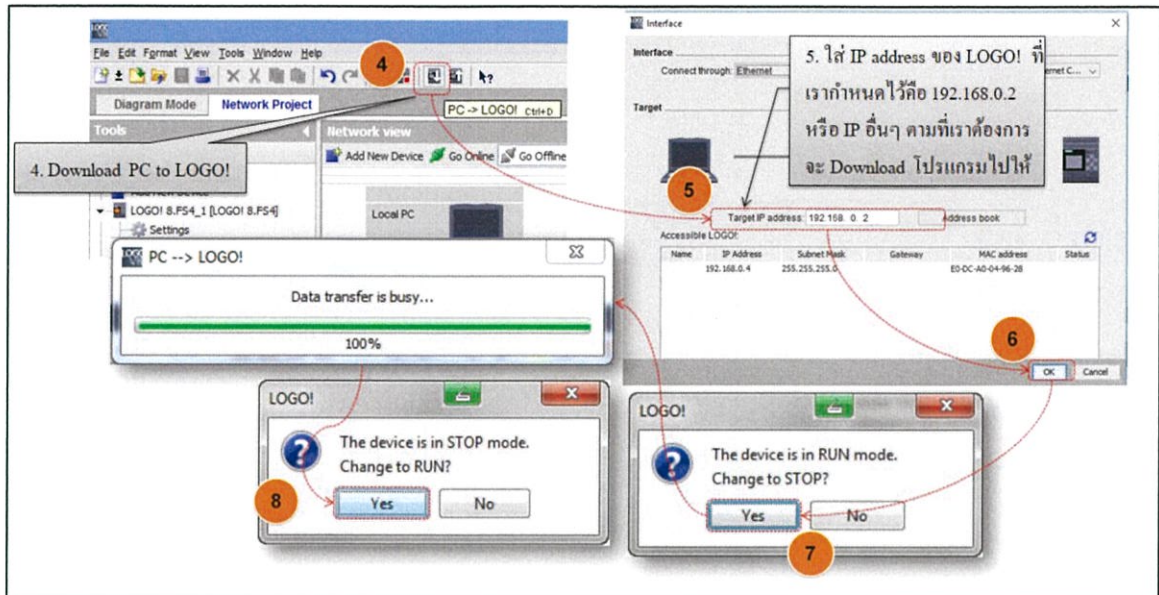
ขั้นตอนสุดท้ายหลังจากผู้ใช้งานทดสอบและตรวจเช็คโปรแกรมเรียบร้อยแล้ว ให้ทำการ Download ลง LOGO! ได้เลย ตามรูปที่ 2.13 และรูปที่ 2.14



รูปที่ 2.13 การเลือกพอร์ต LAN ที่เราเชื่อมต่อเข้ากับคอมพิวเตอร์

ขั้นตอนการ Download มีขั้นตอนดังต่อไปนี้

1. กดรูปเพื่อเข้าไป Setting
2. เลือก Port LAN ของเครื่องคอมพิวเตอร์
3. กด OK



รูปที่ 2.14 การ Download โปรแกรมลง LOGO!8

4. กด Download PC to LOGO!
5. ใส่ IP Address ของ LOGO!
6. กด OK
7. กด Yes เพื่อให้ LOGO! หยุดการทำงาน
8. กด Yes เพื่อให้ LOGO! เริ่มการทำงาน

2.2 SIMATIC HMI Programming Concepts

2.2.1 SIMATIC HMI Concepts

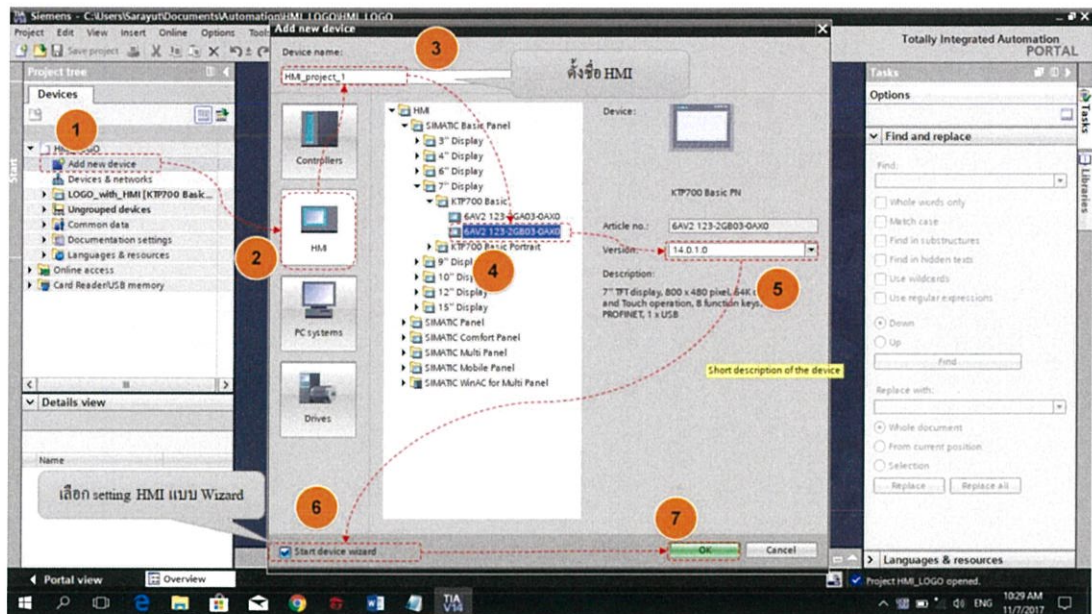
HMI (Human Machine Interface) คือ หน่วยแสดงผลและเป็นตัวกลางในการสื่อสารระหว่างเครื่องจักรกับมนุษย์ HMI เป็นผลิตภัณฑ์ที่มีการใช้งานอย่างแพร่หลายในทุก ๆ อุตสาหกรรม HMI มีหลายรุ่นตามลักษณะการใช้งาน ตั้งแต่การใช้งานขั้นพื้นฐานที่นิยมเรียกกันว่า หน้าจอ HMI (HMI Panel) ไปจนถึงการใช้งานขั้นสูงที่ทำงานบนระบบปฏิบัติการวินโดวส์ หรือที่นิยมเรียกกันว่า SCADA ในการเขียนโปรแกรมหน้าจอ จะใช้ซอฟต์แวร์ที่เรียกว่า WinCC ซึ่งรวมอยู่ใน TIA Portal

2.2.2 Basic Programming

ในการเขียนโปรแกรมหน้าจอเราจะใช้โปรแกรม TIA Portal เขียนโปรแกรม ซึ่ง TIA Portal จะรวมโปรแกรมที่ใช้ในการเขียนโปรแกรมที่เกี่ยวกับ Automation เข้าไว้ด้วยกันได้แก่ Step7, WinCC, Start Drive ซึ่งทำให้ผู้ใช้งานทำงานได้สะดวกและรวดเร็วมากยิ่งขึ้น สำหรับการเขียนโปรแกรมหน้าจอ HMI ผู้ใช้งานสามารถทำได้ง่าย โดยทำตามขั้นตอนการตั้งค่าที่หน้าต่าง Wizard หลังจากเราเพิ่มหน้าจอ HMI เข้ามาในโปรเจค

2.2.2.1 HMI Wizard

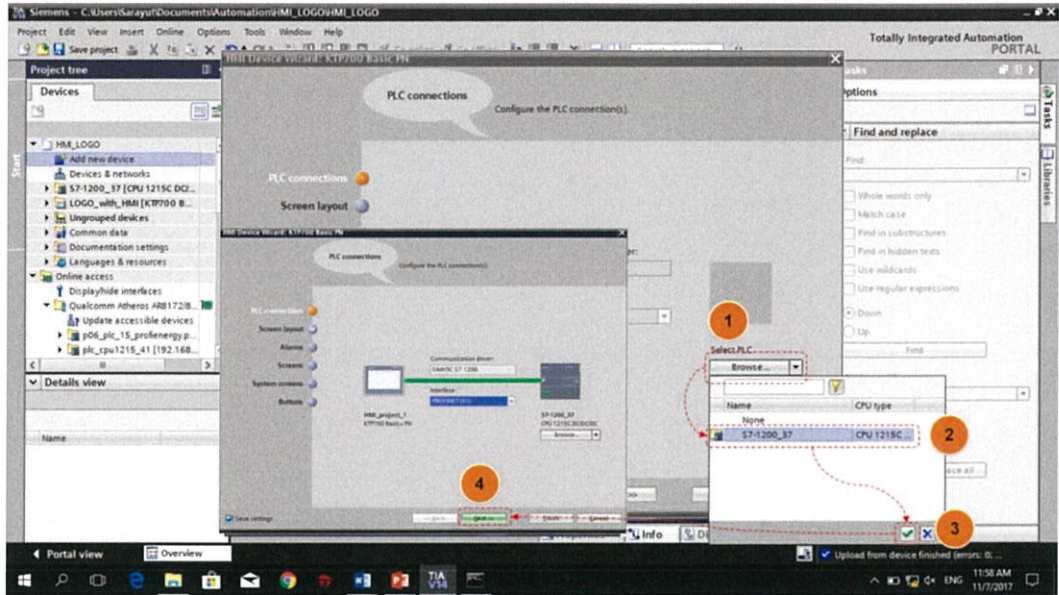
ในขั้นตอนการเพิ่ม HMI โดยวิธี Wizard มีดังนี้



รูปที่ 2.15 การเพิ่ม HMI โดยใช้ Wizard

1. ไปที่ Add Device
2. เลือก HMI
3. ตั้งชื่อให้อุปกรณ์
4. เลือกขนาดและรุ่นของจอ HMI
5. ระบุรุ่นของเฟิร์มแวร์
6. กดติดเครื่องหมายถูกหน้า Start Device Wizard
7. กด OK

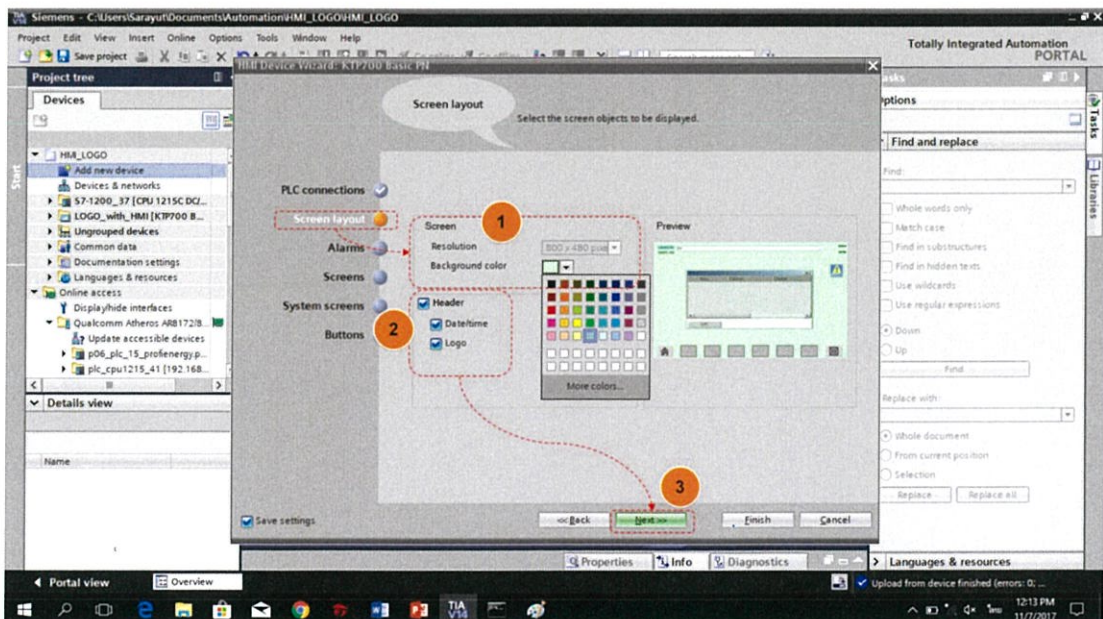
จากนั้นจะปรากฏหน้าต่างดังรูปที่ 2.16 ซึ่งในขั้นตอนแรกก็คือ การเชื่อมต่อกับ PLC โดยมีขั้นตอนดังนี้



รูปที่ 2.16 การเลือก PLC ที่เชื่อมต่อกับ HMI

1. กดปุ่มตรง Browse เพื่อค้นหา PLC ที่จะเลือก
2. เลือก PLC ที่จะนำมาใช้ แล้วกดที่เครื่องหมายถูก
3. เมื่อเลือกเสร็จเส้นเชื่อมต่อจะเป็นสีเขียว
4. กดปุ่ม Next เพื่อไปทำขั้นต่อไป

หลังจากทำการเลือกการเชื่อมต่อกับ PLC เสร็จแล้วต่อไปเป็นขั้นตอนที่ 2 ขั้นตอนนี้เป็นกรตั้งค่าของสีพื้นหลังและส่วนบนของหน้าจอ คือ

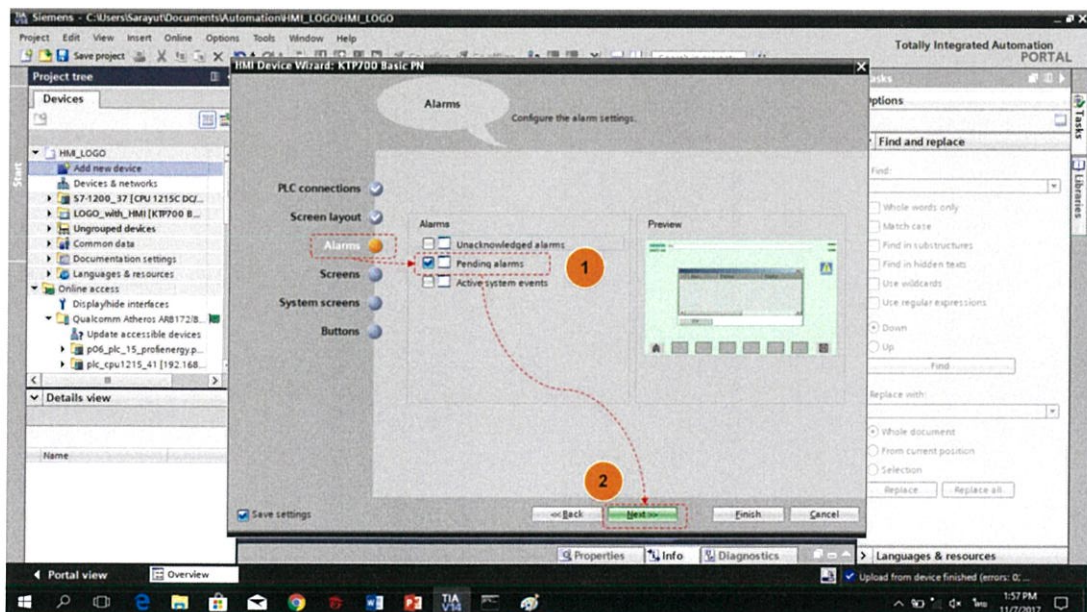


รูปที่ 2.17 การกำหนดสีพื้นหลังและแสดงวันที่และเวลา

1. เปลี่ยนสีของหน้าจอ HMI โดยการกดที่ปุ่มลูกศรซึ่งตรง Background Color แล้วกดเลือกสีได้เลย ดังรูปที่ 2.2.3
2. การเลือกโลโก้ที่จะแสดงและเลือกวันที่และเวลา ซึ่งการเลือกจะทำได้โดยการกดในช่องให้มีเครื่องหมายถูก
3. กด Next

หลังจากกด Next จะเป็นขั้นตอนที่ 3 ของการตั้งค่า ซึ่งเป็นการตั้งค่าฟังก์ชันและรูปแบบการเกิด Alarm โดยจะเลือกให้ทำการกดที่ช่องให้มีเครื่องหมายถูกดังรูปที่ 2.18 โดยรูปแบบการเกิด Alarm จะมีด้วยกัน 3 แบบ ดังนี้

1. Unacknowledged Alarm คือ เมื่อเกิด Alarm จะมีหน้าต่างแจ้งเตือนขึ้นมาและมีสัญลักษณ์ Alarm ทางด้านล่าง ซึ่งจะปิดหน้าต่างนี้ได้ก็ต่อเมื่อ คำนับออกจากขอบเขตการเกิด Alarm
2. Pending Alarm คือ จะมีลักษณะเหมือนกับ Unacknowledged Alarm แต่สัญลักษณ์ Alarm จะอยู่ด้านบน และจะต้องกดปุ่มรับรู้ถึงการเกิด Alarm เพื่อให้การแจ้งเตือนหายไป นอกเหนือจากนี้จะไม่สามารถปิดหน้าต่างที่แจ้งเตือนได้
3. Active System Events คือ หน้าต่างบอกเหตุการณ์ของระบบที่ใช้งาน เช่น บอกวันที่และเวลาของการเกิด Alarm เป็นต้น



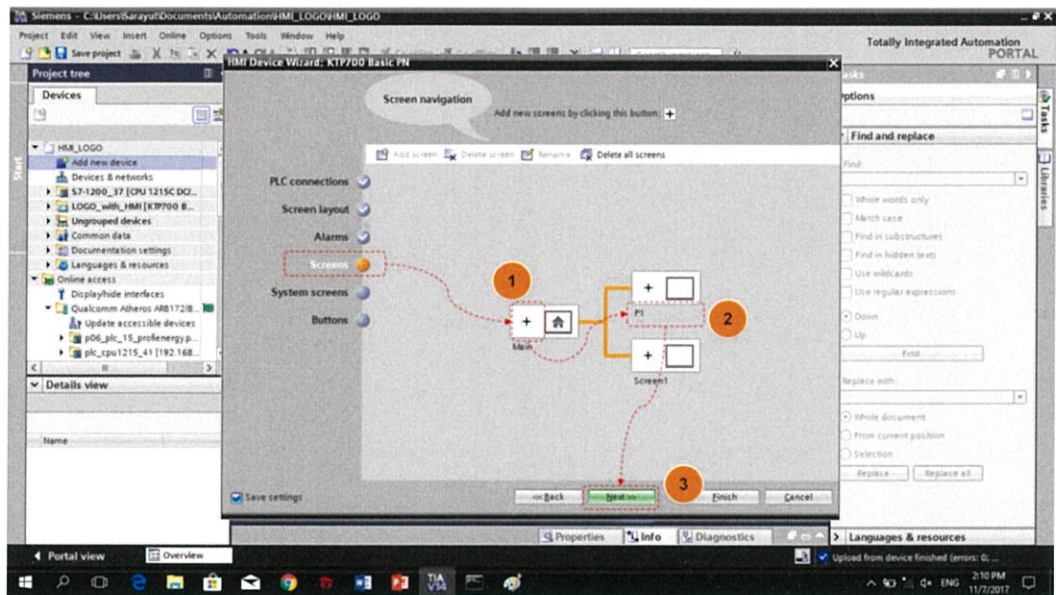
รูปที่ 2.18 การเลือกชนิดของ Alarm ที่ต้องการแสดง

โดยการเลือก Alarm มีขั้นตอนดังนี้

1. เลือกแค่ Pending Alarm

2. กด Next

ขั้นตอนที่ 4 เป็นขั้นตอนของการการสร้างหน้าจอ โดยการเพิ่มหน้าจอโดยวิธี Wizard นั้นจะสร้างหน้าจอพร้อมชื่อที่ตั้ง และมากไปกว่านั้นยังสร้างปุ่มลิงค์ไปหน้าจออื่นๆให้อัตโนมัติ โดยมีขั้นตอนการสร้างจอดังนี้



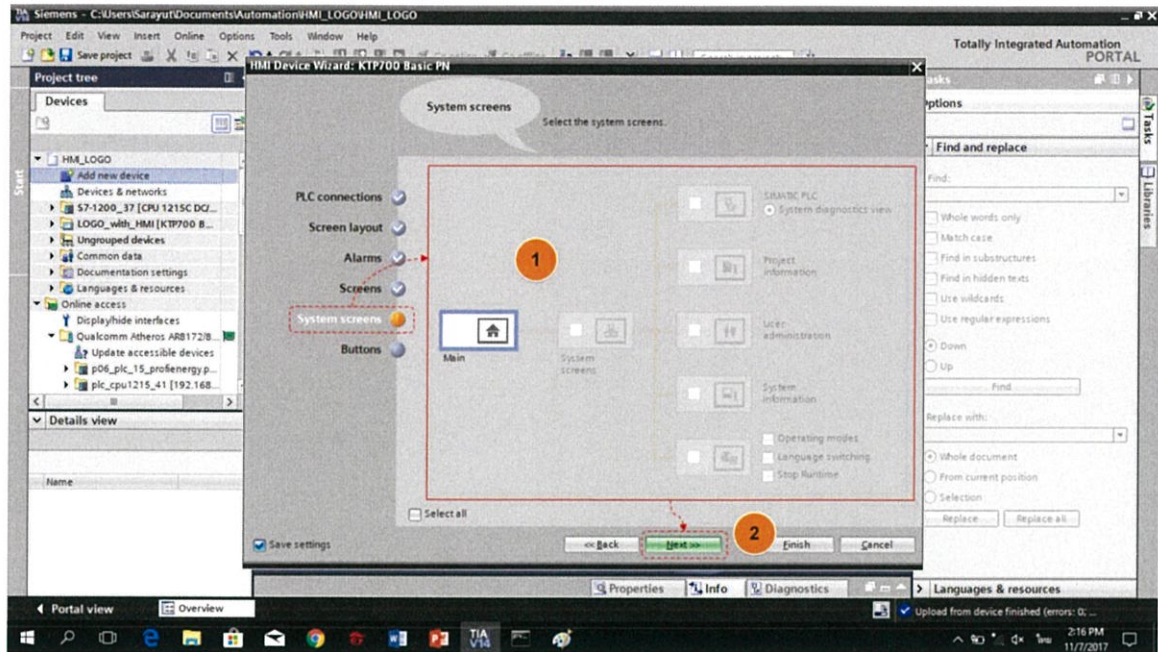
รูปที่ 2.19 การเพิ่มหน้าจอของ HMI

1. โดยเริ่มแรกจะมีเพียงหน้าจอเดียว การเพิ่มทำได้โดยการกดสัญลักษณ์บวก และถ้าจะลบให้กดปุ่ม Delete ได้เลย
2. ถ้าจะทำการตั้งชื่อให้คลิกที่ชื่อ แล้วตั้งชื่อได้ตามที่ต้องการ
3. เสร็จสิ้นที่สร้าง กด Next

ขั้นตอนที่ 5 เป็นการเลือกฟังก์ชันเพิ่มเติมของจอ เลือกฟังก์ชันต่างๆทำได้โดยตรงช่องว่างให้มีเครื่องหมายถูก ดังรูปที่ 2.20 มีฟังก์ชันที่ให้เลือกมีดังนี้

1. SIMATIC PLC คือ สามารถดูข้อมูลเมื่อ PLC ทำงานผิดปกติได้
2. Project Information คือ ข้อมูลของ Project
3. User Administration คือ หน้าต่างเข้าล็อกอิน
4. System Information คือ ข้อมูลและรายละเอียดของระบบ
5. Different Jobs จะมี 3 แบบให้เลือกคือ โหมดการทำงาน การเปลี่ยนภาษา และการหยุดการทำงาน

โดยมีขั้นตอนการเลือกฟังก์ชันเพิ่มเติมดังนี้

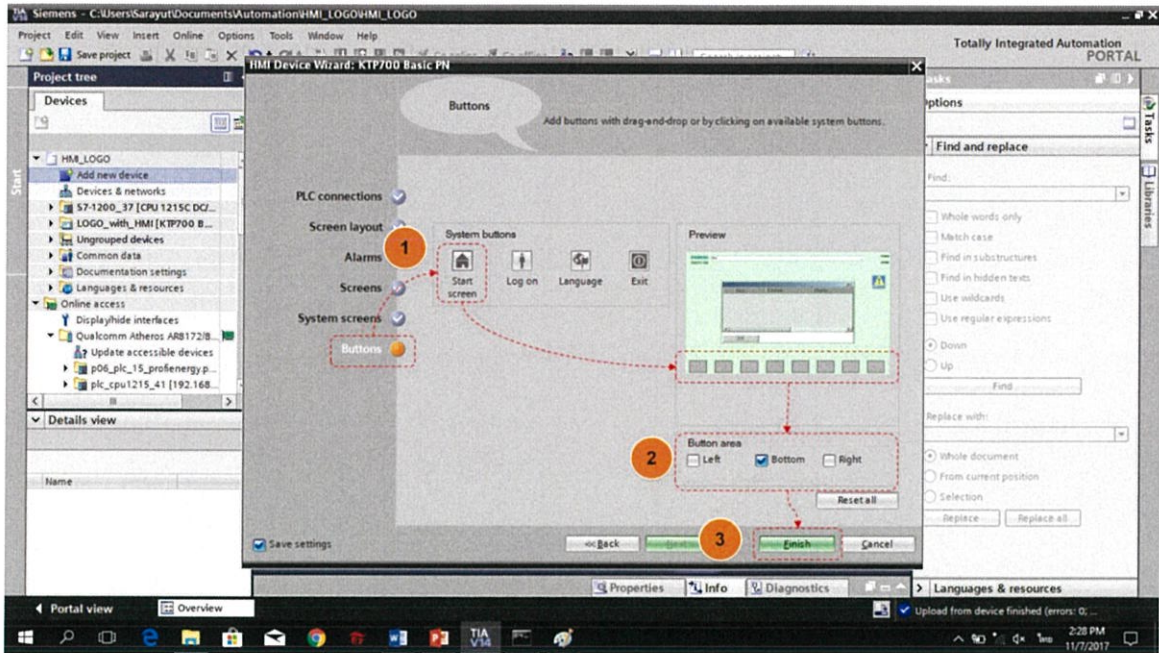


รูปที่ 2.20 การเลือกฟังก์ชันเพิ่มเติมของจอ

1. เลือกฟังก์ชันระบบตามที่ต้องการ
2. กด Next

และขั้นตอนสุดท้ายของการตั้งค่าแบบ Wizard คือ การเลือกปุ่ม โดยวิธีการเลือกจะมีด้วยกัน 2 แบบ คือ ดับเบิลคลิกตรงปุ่มที่ต้อง หรือลากปุ่มที่ต้องการลงในจอเล็กๆทางด้านขวา ดังรูปที่ 2.21 ซึ่งปุ่มที่จะมีให้เลือกดังนี้

- Start Screen คือ หน้าเริ่มต้นของหน้าจอ
- Log On คือ เข้ารหัสล็อกอิน (สำหรับปุ่มที่มีการตั้งค่าระดับการเข้าถึงไว้)
- Language คือ เปลี่ยนภาษา
- Exit คือ ออกจากหน้าจอ

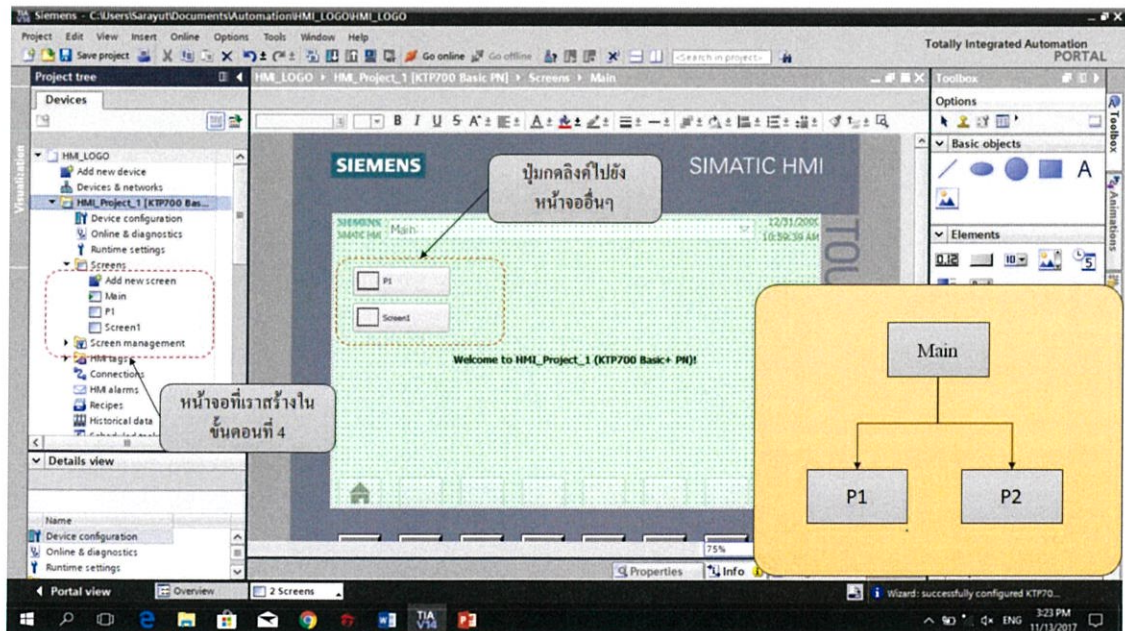


รูปที่ 2.21 การเลือกปุ่ม System

เราสามารถเลือก System Buttons ได้ตามขั้นตอนดังต่อไปนี้

1. เลือก System Buttons ที่ต้องการแล้วคลิกลากไปวางที่ช่อง Preview
2. เลือกตำแหน่งของปุ่มกด
3. กด Finish เพื่อสิ้นสุดการตั้งค่าโดย Wizard

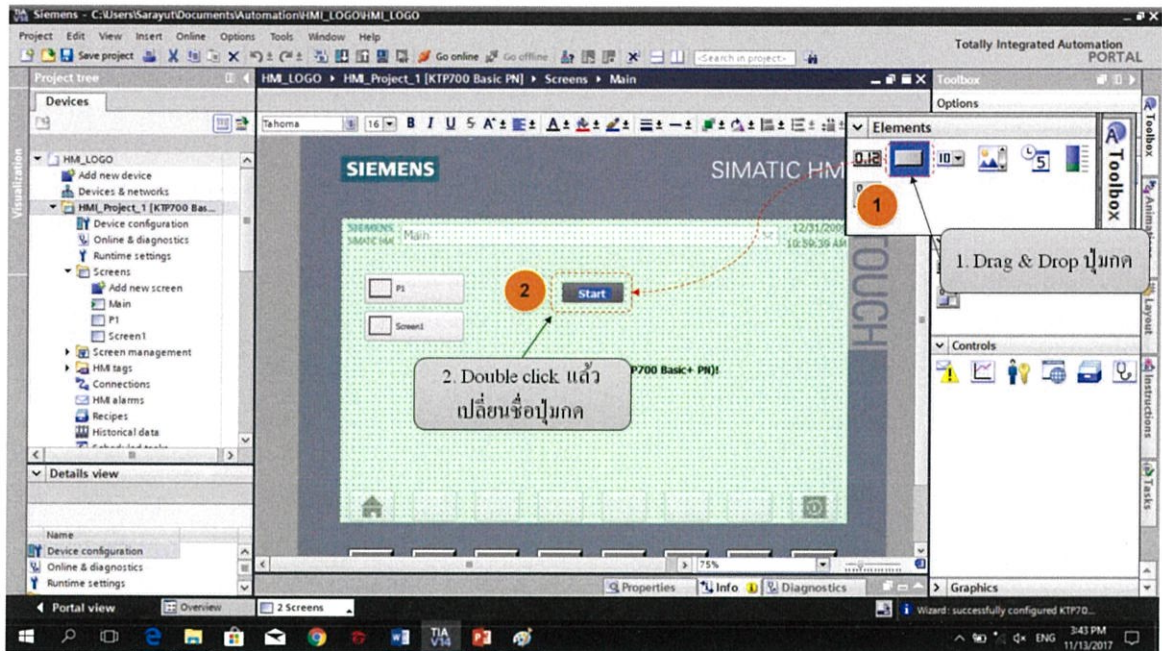
เมื่อกดปุ่ม Finish จะได้หน้าต่างดังรูปที่ 2.22 โดยด้านซ้ายจะเป็นหน้าจอต่างๆ ที่เรากำหนดในการตั้งค่าแบบ Wizard ในขั้นตอนที่ 4



รูปที่ 2.22 หน้าต่างแสดงปุ่มกดคลิกไปยังหน้าจออื่นๆ

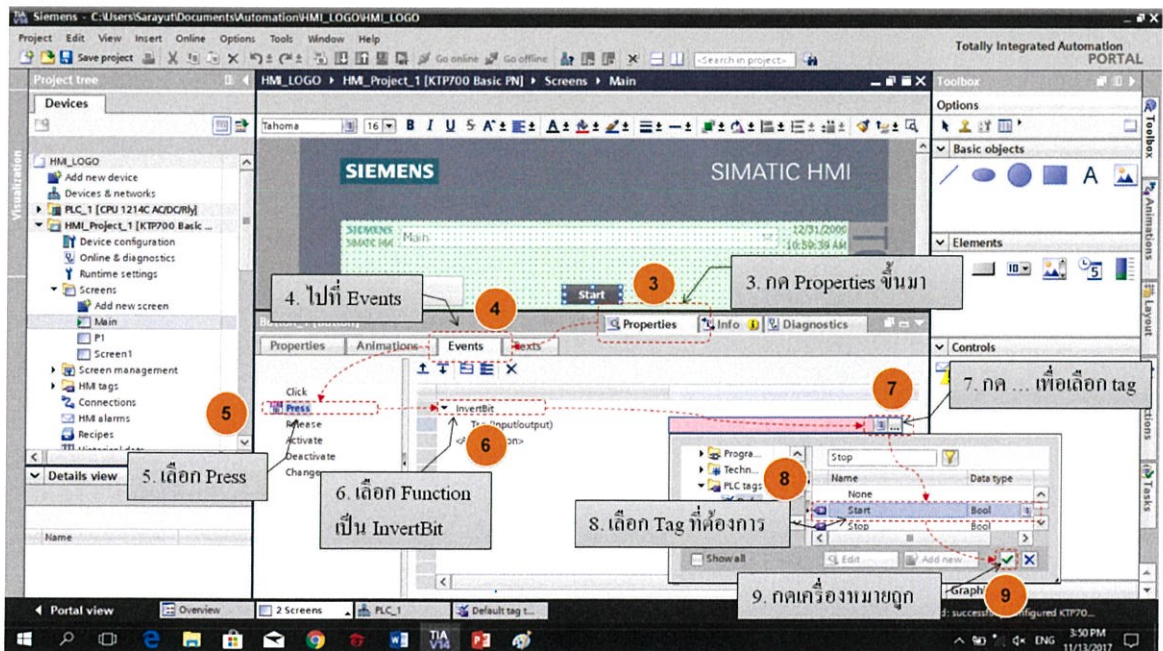
2.2.2.2 Create Elements

อุปกรณ์ใน Element จะเป็นพื้นฐานของอุปกรณ์ในการ Control มีวิธีการในการโปรแกรมที่ค่อนข้างยุ่งยากซับซ้อน ในที่นี้จะยกตัวอย่างการสร้างปุ่มกด โดยมีขั้นตอนดังต่อไปนี้



รูปที่ 2.23 การสร้างปุ่มกด

1. ลากปุ่มกดจาก Elements แล้วนำไปวางบน Screen
2. Double click แล้วเปลี่ยนชื่อปุ่มกด



รูปที่ 2.24 การตั้งค่าปุ่มกดและการเลือก tag

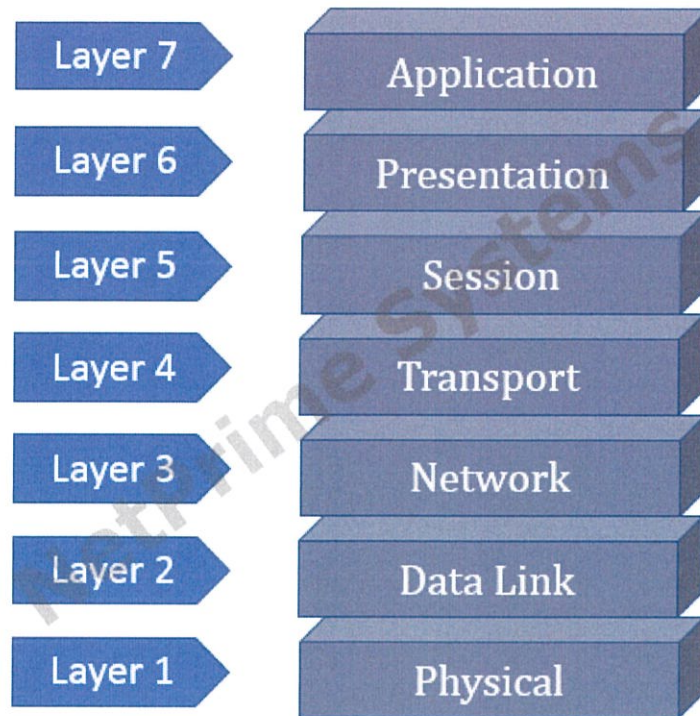
3. กด Properties ขึ้นมา
4. ไปที่ Events
5. เลือก Press
6. เลือก Function เป็น InvertBit เพื่อให้ปุ่มกดเป็นปุ่มกดติดกดดับ
7. กด จุดไข่ปลา เพื่อเลือก Tag
8. เลือก Tag ที่ต้องการ
9. กดเครื่องหมายถูกเป็นอันเสร็จ

2.3 Network and Communication

2.3.1. Ethernet Protocol : OSI/ISO

องค์การมาตรฐานระหว่างประเทศ (The International Organization for Standardization) หรือที่เรียกย่อ ๆ ว่า ISO (ไอโซ) ได้พัฒนา Model ของการทำงานบนระบบเครือข่ายขึ้นมาเป็นมาตรฐานกลาง ในปัจจุบัน ใช้เพื่ออ้างอิงการสื่อสารและเปรียบเทียบการทำงานบนเครือข่ายผู้ผลิตหลาย ๆ บริษัทที่ผลิตอุปกรณ์หรือ Software ต่าง ๆ ขึ้นมาก็จะต้องผลิตตามมาตรฐานที่กำหนดของ **องค์กร** ที่ดูแลในแต่ละส่วน ซึ่ง Model นี้ก็ถูกนำมาใช้มาเปรียบเทียบเพื่อให้เข้าใจและอธิบายการทำงานของแต่ละส่วนได้ง่ายขึ้น และ Model ที่กำลังกล่าวถึงนี้เรียกว่า Open System Interconnection (OSI) OSI Model เป็นเพียง Model ที่ใช้อ้างอิงในการสื่อสารเท่านั้น ปัจจุบันโลกเครือข่ายได้ใช้งานบน TCP/IP เป็นหลัก แต่ที่ยังพูดถึง OSI Model เพราะว่าการแบ่งเป็น Layer นั้นสามารถมองเป็นภาพ กว้างๆ ของการทำงานบนเครือข่ายนั่นเอง

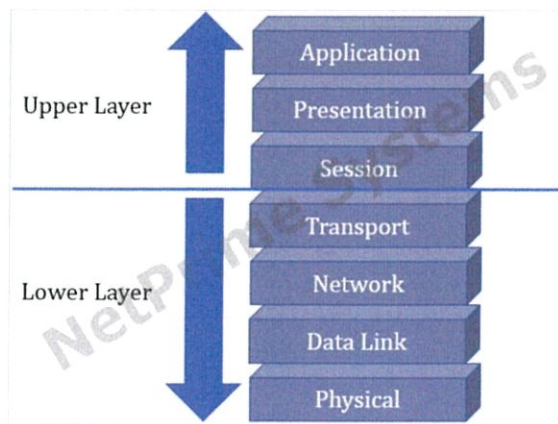
OSI Model ใช้อ้างอิงการสื่อสาร (Reference Model) แบ่งออกเป็นชั้น (Layer) โดยมีตั้งแต่ชั้นที่ 1 ถึงชั้นที่ 7 (Layer 1 – 7) โดย Layer 1 จะอยู่ด้านล่างสุด และเรียงขึ้นไปจนถึง Layer 7 แต่ละ Layer ก็มีชื่อเรียกตามรูปแบบการสื่อสารและการทำงาน ดังรูปที่ 2.3.1



รูปที่ 2.25 OSI Model 7 Layer

บน OSI Model ก็จะถูกแบ่งเป็น 2 กลุ่มใหญ่ คือ

- ตั้งแต่ Layer 1 – 4 เรียกว่า Lower Layer
- ตั้งแต่ Layer 5 – 7 เรียกว่า Upper Layer



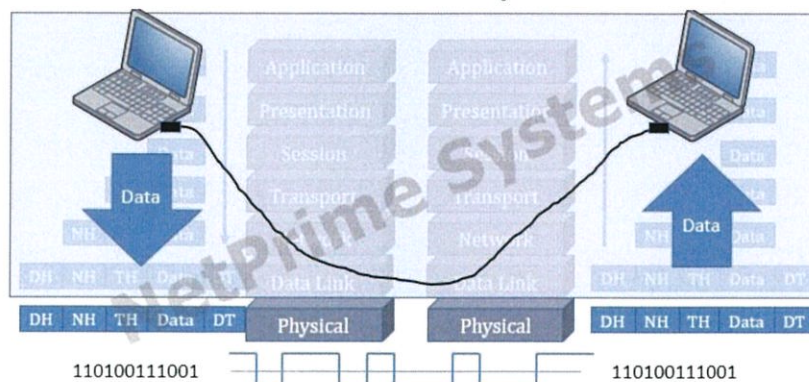
รูปที่ 2.26 OSI Model Upper/Lower Layer

2.3.1.1 Layer 1 (Physical Layer)

เป็นชั้นล่างสุดจะมีการกำหนดคุณสมบัติทางกายภาพของฮาร์ดแวร์ที่ใช้เชื่อมต่อระหว่างคอมพิวเตอร์ทั้งสองระบบ เช่น สายที่ใช้รับส่งข้อมูลจะเป็นแบบไหน, ข้อต่อที่ใช้ในการรับส่งข้อมูลมี

มาตรฐานอย่างไร, ความเร็วในการรับส่งข้อมูลเท่าใด, สัญญาณที่ใช้ในการรับส่งข้อมูลมีรูปร่างอย่างไร, ใช้แรงดันไฟฟ้าเท่าไร

ข้อมูลใน Layer ที่ 1 นี้จะมองเห็นเป็นการรับส่งข้อมูลที่ละบิตเรียงต่อกันไป



รูปที่ 2.27 การส่งข้อมูลบน Physical Layer

จากรูปที่ 2.27 แสดงถึงการส่งข้อมูลบน Physical Layer แสดงให้เห็นว่า ข้อมูลจะมาเป็นอย่างไรมาก็ตาม ก็จะถูกแปลงเป็นสัญญาณเพื่อส่งไปยังปลายทาง แล้วฝั่งปลายทางก็จะนำสัญญาณที่รับมาแปลงกลับเป็นข้อมูลเพื่อส่งให้เครื่อง Client ต่อไป อุปกรณ์ต่าง ๆ ที่มีความสามารถในการนำพาสัญญาณไป เช่น Card LAN (NIC), สาย UTP, สาย Fiber หรือ เต้าเสียบ หัวต่อต่าง ๆ RJ45, RJ11, RS323 ก็จัดอยู่ใน Physical Layer

2.3.1.2 Layer 2 (Data-Link Layer)

เป็นชั้นที่ทำหน้าที่กำหนดรูปแบบของการส่งข้อมูลข้าม Physical Network โดยใช้ Physical Address อ้างอิงที่อยู่ต้นทางและปลายทาง ซึ่งก็คือ MAC Address รวมถึงทำการตรวจสอบและจัดการกับ Error ในการรับส่งข้อมูล ข้อมูลที่ถูกส่งบน Layer 2 เราจะเรียกว่า Frame ซึ่งบน Layer 2 ก็จะแบ่งเป็น LAN และ WAN

ปัจจุบันบน Layer 2 LAN เรานิยมใช้เทคโนโลยีแบบ Ethernet มากที่สุด ส่วน WAN ก็จะมีหลายแบบแตกต่างกันไป เช่น Lease Line (HDLC , PPP) , MPLS , 3G และอื่น ๆ

| 8 Bytes | 6 Bytes | 6 Bytes | 2 Bytes | 46-1500 Bytes | 4 Bytes |
|----------|---------------------|----------------|---------|---------------|---------|
| Preamble | Destination Address | Source Address | Type | Data | FCS |

รูปที่ 2.28 Frame การส่งข้อมูล

LAN ยังมีการแบ่งย่อยออกเป็น 2 Sublayers คือ

- Logical Link Control (LLC)

IEEE 802.2 ซึ่งจะให้บริการกับ Layer ด้านบนในการเข้าใช้สัญญาณใน การรับ-ส่งข้อมูล ตามมาตรฐาน IEEE802 แล้วจะอนุญาตให้สถาปัตยกรรมของ LAN ที่ต่างกันสามารถทำงานร่วมกันได้ หมายความว่า Layer ด้านบนไม่จำเป็นต้องทราบว่า Physical Layer ใช้สายสัญญาณประเภทใดในการรับ-ส่งข้อมูล เพราะ LLC จะรับผิดชอบในการปรับ Frame ข้อมูลให้สามารถส่งไปได้ในสายสัญญาณประเภทนั้นได้ และไม่จำเป็นต้องสนใจว่าข้อมูลจะส่งผ่านเครือข่ายแบบไหน เช่น Ethernet, Token Ring และไม่จำเป็นต้องรู้ว่าการส่งผ่านข้อมูลใน Physical Layer จะใช้การรับส่งข้อมูลแบบใด LLC จะเป็นผู้จัดการเรื่องเหล่านี้ได้ทั้งหมด

- Media Access Control (MAC)

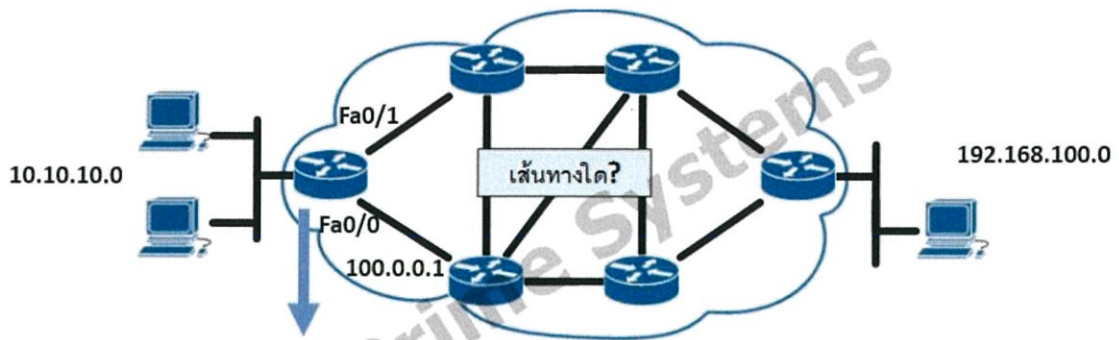
IEEE 802.3 ใช้ควบคุมการติดต่อสื่อสารกับ Layer 1 และรับผิดชอบในการรับ-ส่งข้อมูลให้สำเร็จ และถูกต้อง โดยมีการระบุ MAC Address ของอุปกรณ์เครือข่าย ซึ่งใช้อ้างอิงในการส่งข้อมูลจากต้นทางไปยังปลายทาง เช่น จาก ต้นทางส่งมาจาก MAC Address หมายเลข AAAA:AAAA:AAAA ส่งไปหาปลายทางหมายเลข BBBB:BBBB:BBBB เมื่อปลายทางได้รับข้อมูลก็จะรู้ว่าใครส่งมา เพื่อจะได้ตอบกลับไปถูกต้อง

บน Ethernet (IEEE802.3) เมื่อมีหน้าที่ในการรับผิดชอบการรับ-ส่งข้อมูลให้สำเร็จและถูกต้อง จึงมีการตรวจสอบข้อผิดพลาดในการส่งข้อมูลด้วย ที่เราเรียกว่า Frame Check Sequence (FCS) และ ยังตรวจสอบกับ Physical ด้วยว่าช่องสัญญาณพร้อมสำหรับส่งข้อมูลไหม ถ้าว่างก็ส่งได้ ถ้าไม่ว่างก็ต้องรอ กลไกนี้เรารู้จักกันในชื่อ CSMA/CD

CSMA/CD คือ กลไกการตรวจสอบการชนกันของข้อมูล บน Ethernet ถ้าเกิดมีการชนกันเกิดขึ้น (Jam Signal) มันก็จะส่งสัญญาณออกไปเพื่อให้ทุกคนหยุดส่งข้อมูล แล้วสุ่มรอเวลา (Back Off) เพื่อส่งใหม่อีกครั้ง

2.3.1.3 Layer 3 (Network Layer)

ทำหน้าที่ส่งข้อมูลข้ามเครือข่ายหรือข้าม Network โดยส่งข้อมูลผ่าน Internet Protocol (IP) โดยมีการสร้างที่อยู่ขึ้นมา (Logical Address) เพื่อใช้อ้างอิงเวลาส่งข้อมูล เรียกว่า IP Address ข้อมูลที่ถูกส่งมาจากต้นทาง เพื่อไปยังปลายทาง ที่ไม่ได้อยู่บนเครือข่ายเดียวกัน จำเป็นจะต้องพึ่งพาอุปกรณ์ที่ทำงานบน Layer 3 นั่นก็คือ Router หรือ Switch Layer 3 โดยใช้ Routing Protocol (OSPF, EIGRP) เพื่อหาเส้นทางและส่งข้อมูลนั้น (IP) ข้ามเครือข่ายไป



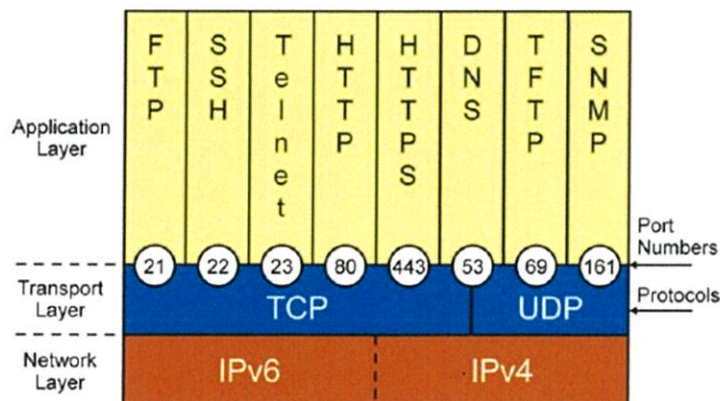
| Network Protocol | Destination | Exit Interface | Next Hop |
|------------------|---------------|----------------|-----------|
| Connected | 10.10.10.0 | Fa0/1 | |
| Learned | 192.168.100.0 | Fa0/0 | 100.0.0.1 |

รูปที่ 2.29 การส่งข้อมูลข้ามเครือข่าย

โดยการทำงานของ Internet Protocol (IP) เป็นการทำงานแบบ Connection-less หมายความว่า IP ไม่มีการตรวจสอบข้อมูลว่าส่งไปถึงปลายทางไหม แต่มันจะพยายามส่งข้อมูลออกไปด้วยความพยายามที่ดีที่สุด (Best-Effort) เพราะฉะนั้น ข้อมูลที่ส่งออกไปแล้วไม่ถึงปลายทาง ต้นทางก็จะไม่รู้เลย ถ้าส่งไปแล้วข้อมูลไม่ถึงปลายทาง ฝั่งต้นทางจะต้องทำการส่งไปใหม่ บน Layer 3 จึงมี Protocol อีกตัวเพื่อใช้ตรวจสอบว่าปลายทางยังมีชีวิตอยู่ไหมก่อนที่จะส่งข้อมูล นั่นคือ ICMP แต่ผู้ใช้งานจะต้องเป็นคนเรียกใช้ Protocol ตัวนี้เอง

2.3.1.4 Layer 4 (Transport Layer)

ทำหน้าที่เป็นตัวเชื่อมต่อกับ Upper Layer ในการใช้งาน Network Services ต่าง ๆ หรือ Application ต่าง ๆ จากต้นทางไปยังปลายทาง (End-to-End Connection) ในแต่ละ Services ได้ โดยใช้ Port Number ในการส่งข้อมูลของ Layer 4 จะใช้งานผ่าน Protocol 2 ตัว คือ TCP และ UDP



รูปที่ 2.30 การเชื่อมต่อกับ Upper Layer

เมื่อข้อมูลถูกส่งมาใช้งานผ่าน Services Telnet ไปยังปลายทางถูกส่งลงมาที่ Layer 4 ก็จะทำให้การแยกว่า Telnet คือ Port Number 23 เป็น Port Number ที่ใช้ติดต่อไปหาปลายทาง แล้วฝั่งต้นทางก็จะ Random Port Number ขึ้นมา เพื่อให้ปลายทางสามารถตอบกลับมาได้เช่นเดียวกัน

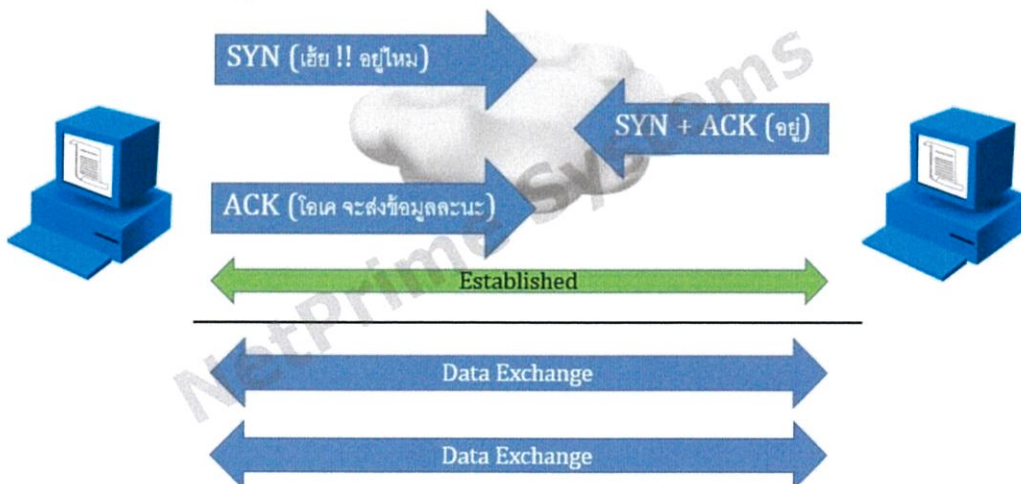


รูปที่ 2.31 การใช้งานผ่าน Services Telnet

Transmission Control Protocol (TCP) มีคุณลักษณะที่สำคัญ ดังนี้

- จัดแบ่งข้อมูลจากระดับ Application ให้มีขนาดพอเหมาะที่จะส่งไปบนเครือข่าย (Segment)
- มีการสร้าง Connection กันก่อนที่จะมีการรับส่งข้อมูลกัน (Connection-oriented)
- มีการใช้ Sequence Number เพื่อจัดลำดับการส่งข้อมูล
- มีการตรวจสอบว่าข้อมูลที่ส่งไปถึงปลายทางหรือไม่ (Recovery)

บน TCP ก่อนจะส่งข้อมูลนั้นจะต้องทำการตรวจสอบก่อนว่า ปลายทางสามารถติดต่อได้ โดยจะทำการสร้างการเชื่อมต่อระหว่างผู้ส่งและผู้รับก่อน โดยใช้กลไก Three-Way Handshake เพื่อให้แน่ใจว่าข้อมูลที่ส่งจะสามารถส่งถึงผู้รับแน่นอน



รูปที่ 2.32 กลไก Three-Way Handshake

นอกจาก Three-Way Handshake แล้ว TCP ยังมีกลไก Flow Control เพื่อควบคุมการส่งข้อมูลเมื่อเกิดปัญหาบนเครือข่ายระหว่างที่ส่งข้อมูลอยู่ หรือกลไก Error Recovery ในกรณีที่มีข้อมูลบางส่วนหายไปขณะส่ง ก็ให้ทำการส่งมาใหม่ (Retransmission)

ยังสามารถทำการจัดสรรหรือแบ่งส่วนของข้อมูลออกเป็นส่วนๆ (Segmentation) ก่อนที่จะส่งลงไป Layer 3 และข้อมูลที่ถูกแบ่งออกจะใส่ลำดับหมายเลขเข้าไป (Sequence Number) เพื่อให้ปลายทางนำข้อมูลไปประกอบกันได้ถูกต้อง

User Datagram Protocol (UDP) มีคุณลักษณะที่สำคัญ ดังนี้

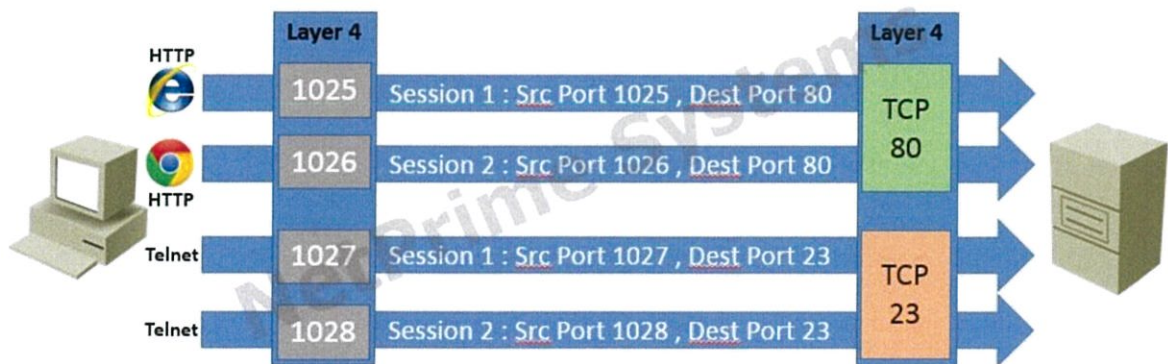
- ไม่มีการสร้าง Connection กันก่อนที่จะมีการรับส่งข้อมูลกัน (Connectionless)
- ส่งข้อมูลด้วยความพยายามที่ดีที่สุด (Best-Effort)
- ไม่มีการตรวจสอบว่าข้อมูลที่ส่งไปถึงปลายทางหรือไม่ (No Recovery)

บน UDP จะตรงข้ามกับ TCP เพราะไม่มีการสร้างการเชื่อมต่อกันก่อน หมายความว่าถ้า Services ใด ๆ ใช้งานผ่าน UDP ก็จะถูกส่งออกไปทันทีด้วยความพยายามที่ดีที่สุด (Best-Effort) และไม่มีการส่งใหม่เมื่อข้อมูลสูญหาย (No Recovery) หรือส่งไม่ถึงปลายทางอีกด้วย

ข้อดี คือ มีความรวดเร็วในการส่งข้อมูล เพราะฉะนั้น Services ที่ใช้งานผ่าน UDP ก็มีมากมาย เช่น TFTP , DHCP , VoIP และอื่น ๆ เป็นต้น

2.3.1.5 Layer 5 (Session Layer)

ทำหน้าที่ควบคุมการเชื่อมต่อ Session เพื่อติดต่อจากต้นทาง กับ ปลายทาง



รูปที่ 2.33 การติดต่อระหว่างต้นทางกับปลายทาง

เมื่อฝั่งต้นทางต้องการติดต่อไปยังปลายทางด้วย Port 80 (เปิด Internet Explorer) ฝั่งต้นทางก็จะทำการติดต่อไปยังปลายทาง โดยการสร้าง Session ขึ้นมา เป็น Session ที่ 1 ส่งผ่าน Layer 4 โดย

Random Port ต้นทางขึ้นมาเป็น 1025 ส่งไปหาปลายทางด้วย port 80 ระหว่างที่ Session ที่ 1 ใช้งานอยู่ เราติดต่อไปยังปลายทางอีกครั้งด้วย Port 80 (เปิด Google Chrome) ฝั่งต้นทางก็จะทำการสร้าง Session ที่ 2 ขึ้นมา ส่งผ่าน Layer 4 โดย Random Port ต้นทางขึ้นมาเป็น 1026 ส่งไปหาปลายทางด้วย Port 80 แล้วแต่ละ Session ฝั่งปลายทาง ก็จะตอบกลับมาด้วย Port ที่ฝั่งต้นทางส่งมา ทำให้สามารถแยก Session ออกได้ เมื่อเราส่งข้อมูลบนเครือข่าย

2.3.1.6 Layer 6 (Presentation Layer)

ทำหน้าที่ในการแปล หรือนำเสนอ Structure, Format, Coding ต่าง ๆ ของข้อมูลบน Application ที่จะส่งจากต้นทางไปยังปลายทาง ให้อยู่ในรูปแบบที่ฝั่งต้นทางและปลายทาง สามารถเข้าใจได้ทั้ง 2 ฝั่ง Computer ส่วนมากใช้ ASCII Format ถ้าจะติดต่อกับ Computer Mainframe จะใช้พวก EBCDIC Format ซึ่ง Layer 6 ก็จะทำให้ทำการ Reformat ข้อมูล ซึ่งสามารถทำให้ทั้ง 2 เครื่องสามารถเข้าใจ Format ข้อมูลที่จะสื่อสารกันได้

2.3.1.7 Layer 7 (Application Layer)

ทำหน้าที่ติดต่อระหว่างผู้ใช้ (User) กับ Application ที่ใช้งานบนเครือข่าย เช่น Web Browser (HTTP), FTP, Telnet เป็นต้น สรุปก็คือ Application ที่ใช้งานผ่าน Network

2.3.2. Ethernet Network: IEEE802.3 & IEEE802.11

2.3.2.1. IEEE 802.3

สำหรับมาตรฐาน 802.3 จะอธิบายถึง LAN ทั้งหมดที่ใช้หลักการของ CSMA/CD (Carrier Sense Multiple Access with Collision Detection) ที่มีอัตราการส่งข้อมูลตั้งแต่ 1 Mbps ถึง 100Mbps และใช้สายส่งชนิดต่างๆ นอกจากนี้มาตรฐาน IEEE 802.3 และอีเทอร์เน็ตยังมีบางส่วนของส่วนหัวของข้อมูล (Header) แตกต่างกันไปบ้าง (ฟิลด์ความยาวของ IEEE 802.3 ถูกใช้บ่งบอกชนิดของ Packet ในมาตรฐานอีเทอร์เน็ต)

ดังนั้นจะเห็นได้ว่ามาตรฐาน IEEE 802.3 จะอธิบายถึง LAN ที่ใช้วิธีส่งข้อมูลแบบ CSMA/CD ส่วนอีเทอร์เน็ตนั้นจะหมายถึงผลิตภัณฑ์ชนิดหนึ่งของแลนแบบ IEEE 802.3

LAN แบบนี้ส่งข้อมูลโดยใช้หลักการคล้ายๆกับการสนทนาระหว่างบุคคลหลายคน หากใครต้องการพูดก็สามารถพูดออกมาได้ในจังหวะที่ไม่มีคนอื่นพูด (เงียบ) แต่ก็อาจเป็นไปได้ที่บุคคล 2 คนจะพูดออกมาพร้อมๆกัน ทำให้เกิดการชนกันของเสียงพูด เมื่อเป็นเช่นนั้นทั้งสองคนจะต้องหยุดพูดทันที แล้วรอจังหวะที่จะพูดใหม่อีกครั้ง ซึ่งหากใครพูดก่อนก็จะสามารถพูดได้ และบุคคลอื่นๆจะต้องฟังอย่างเดียว

วิธีการรับส่งข้อมูลของแลน IEEE 802.3 ซึ่งเป็นแบบ CSMA/CD ก็ทำงานในลักษณะเดียวกัน คือ โหนดใดที่ต้องการส่งข้อมูลลงในสื่อกลางการส่งข้อมูล จะตรวจสอบดูสัญญาณในสื่อกลาง ถ้าหากสื่อกลางในการส่งข้อมูลว่างก็จะทำการส่งข้อมูลได้ทันที แต่หากโหนดตั้งแต่ 2 โหนดขึ้นไปส่งข้อมูลลงในสื่อกลางพร้อมๆกัน สัญญาณข้อมูลจะเกิดการชนกันขึ้น ทุกๆสถานีจะต้องหยุดการส่งข้อมูลแล้วรอเวลา ซึ่งช่วงเวลาของการรอแต่ละครั้งจะทำการสุ่มขึ้นมา (Random Time) หลังจากหมดเวลารอแล้วก็จะทำการตรวจสอบสัญญาณในสื่อกลางเพื่อส่งข้อมูลลงไปใหม่อีก

เมื่อเกิดการชนกันของสัญญาณข้อมูลแล้ว เวลาจะถูกแบ่งออกเป็นช่องๆ (Slots) แต่ละช่องมีช่วงเวลา 51.2 ไมโครวินาที (นั่นคือเวลาสถานีที่ส่งข้อมูลรู้ว่าเกิดการชนกันของข้อมูลหรือไม่ สำหรับความยาวของแลน 2,500 เมตร อัตราการส่งข้อมูล 10 Mbps) หลังจากการชนกันครั้งแรก แต่ละสถานีจะสร้างตัวเลขสุ่ม (Random) ที่มีค่า 0 หรือ 1 (เลขสุ่ม 2^1 ค่า)

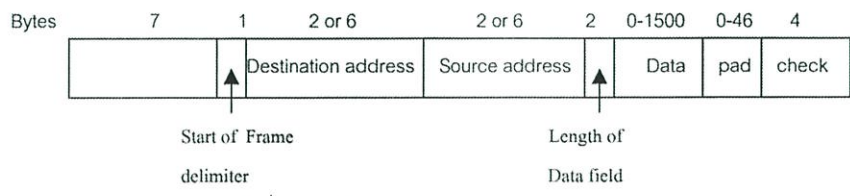
สถานีที่ได้ค่า 0 จะส่งข้อมูลออกไปในช่องเวลา 0 และสถานีที่ได้ค่า 1 จะส่งข้อมูลในช่องเวลาที่ 1 หากสองสถานีได้ค่าเลขสุ่มเดียวกันและส่งข้อมูลภายในช่องเวลาเดียวกัน จะเกิดการชนกันอีกครั้ง

หลังจากการชนกันครั้งที่ 2 แต่ละสถานีจะสร้างตัวเลขสุ่มที่มีค่า 0,1,2, หรือ 3 (นั่นคือเลขสุ่ม 2^2 ค่า) แล้วส่งข้อมูลภายในช่องเวลาของตนเอง หากชนกันอีกก็จะสร้างเลขสุ่มจำนวน 2^3 ค่า กล่าวคือ หลังจากการชนกัน i ครั้ง แต่ละสถานีก็จะมีการสร้างเลขสุ่มตั้งแต่ค่า 0 ถึง 2^{i-1} ค่า และสถานีก็จะส่งข้อมูลภายในช่องเวลาของตนเอง กระบวนการในการแก้ไขการชนกันของข้อมูลแบบนี้เรียกว่า Binary Exponential Back off ซึ่งจะเห็นได้ว่ากระบวนการนี้ทำให้โอกาสในการที่จะเกิดการชนกันของข้อมูลมีน้อยลง เมื่อจำนวนครั้งของการชนกันของข้อมูลมากขึ้น

IEEE 802.3 FRAME FORMAT

โครงสร้างของ Frame ข้อมูลตามมาตรฐาน IEEE 802.3 คือจะเริ่มต้นด้วยส่วนที่เรียกว่า preamble ซึ่งมีความยาว 7 ไบต์แต่ละไบต์จะมีข้อมูลเหมือนกันหมดคือ “10101010” เมื่อผ่านการเข้ารหัสแบบแมนเชสเตอร์จะทำให้ได้สัญญาณแบบสี่เหลี่ยมที่มีความถี่ 10 MHz เป็นเวลา 5.6 μ S เพื่อให้ปลายทางสามารถซิงโครไนซ์สัญญาณนาฬิกากับผู้ส่งได้และต่อไปจะเป็นไบต์เริ่มของเฟรม (Start of Frame) และมีขอมูลเป็น “10101011” เพื่อเป็นการบอกว่าเป็นการเริ่มต้นของเฟรม

แต่ละเฟรมจะประกอบไปด้วยแอดเดรสสองส่วนคือแอดเดรสของผู้ส่งและของผู้รับซึ่งสองแอดเดรสมีขนาดเป็น 2 หรือ 6 ไบต์ก็ได้ที่บิตสูงสุดของแอดเดรสปลายทางเป็น “0” สำหรับปลายทางธรรมดาและเป็น “1” สำหรับปลายทางที่เป็นกลุ่ม (หมายความว่าข้อมูลที่ส่งไปจะถูกรับโดยทุกสถานีภายในกลุ่ม การส่งลักษณะนี้เรียกว่า มัลติแคสต์)



รูปที่ 2.34 โครงสร้างของเฟรม 802.3

มาตรฐานสื่อการส่งข้อมูลของ IEEE 802.3

ตามมาตรฐานการส่งข้อมูล IEEE 802.3 สื่อข้อมูลจะส่งข้อมูลด้วยความเร็ว 10 เมกะบิตต่อ วินาที ยกเว้นมาตรฐาน 1BASE5 ที่มีการส่งข้อมูลด้วยความเร็วเพียง 1 เมกะบิตต่อวินาทีแต่มีระยะการส่งได้ไกลกว่า มาตรฐานที่นิยมได้แก่มาตรฐาน

10BASE5: เป็นการส่งข้อมูลโดยใช้สายส่งข้อมูลแบบ RG – 11 หรือสายโคแอกเชียลแบบหนา ความยาวสูงสุดของเซกเมนต์ 500 เมตร (1640 ฟุต) สามารถต่อตัว Transceiver ในแต่ละเซกเมนต์ได้สูงสุด 100 ตัว จำนวนสูงสุดของ DTE ในระบบเครือข่ายคือ 1024 ตัว ระยะห่างระหว่าง DTE ต้องไม่ต่ำกว่า 2.5 เมตร การส่งข้อมูลใช้วิธีแบบสแบนด

10BASE2: เป็นการส่งข้อมูลโดยใช้สายส่งข้อมูลแบบ RG – 58A/U 11 หรือสายโคแอกเชียล แบบบางนั่นเอง ความยาวสูงสุดของเซกเมนต์ต้องไม่เกิน 200 เมตร (600 ฟุต) สามารถต่อตัว Transceiver ในแต่ละเซกเมนต์ได้สูงสุด 30 ตัวและระยะห่างระหว่างตัว Transceiver กับ เซกเมนต์ต้องไม่ต่ำกว่า 0.5 เมตร การส่งข้อมูลใช้วิธีแบบสแบนด

10BASE-T: ใช้สายส่งข้อมูลแบบ 20 ถึง 24 AWG UTP หรือสายส่งข้อมูลแบบคูบิดเกลียวและใช้ตัวเชื่อมต่อแบบ RJ-45 ระยะห่างระหว่างตัว Transceiver กับฮับต้องยาวไม่เกิน 100 เมตร (328 ฟุต) ฮับ 1 ตัวมีสถานีผู้ใข้ต่ออยู่ไม่เกิน 1023 สถานี

10BROAD36: ใช้สายส่งข้อมูลแบบสายโคแอกเชียล (RG-59/U CATV Type) ความยาวสูงสุดของเซกเมนต์ 3600 เมตร ใช้การส่งข้อมูลแบบ Broadband

10BASE-F: ใช้สายส่งข้อมูลแบบใยแก้ว (Fiber Optic)

1BASE5: ใช้สายส่งข้อมูลแบบคูบิดเกลียวที่มีความยาวของเซกเมนต์ได้สูงสุด 500 เมตร ส่งข้อมูลด้วยความเร็ว 1 เมกะบิตต่อวินาที

2.3.2.2. IEEE 800.11

IEEE 802.11 คือมาตรฐานการทำงานของระบบเครือข่ายไร้สายกำหนดขึ้นโดย Institute of Electrical and Electronics Engineers (IEEE) เป็นมาตรฐานกลางที่ได้นำมาปฏิบัติใช้ เพื่อที่จะทำการเชื่อมโยงอุปกรณ์เครือข่ายไร้สายเข้าด้วยกันบนระบบ ในทางปกติแล้วการเชื่อมต่อระบบเครือข่ายไร้สายจำเป็นต้องใช้อุปกรณ์สองชิ้น นั่นคือ

- 1) Access Point คือ ตัวกลางที่ช่วยในการติดต่อระหว่าง ตัวรับ-ส่งสัญญาณ Wireless ของผู้ใช้กับ Router ผ่านทางสายนำสัญญาณที่ทำจากทองแดงที่ได้รับการเชื่อมต่อกับระบบเครือข่าย เช่น สายแลน หรือสายโทรศัพท์ ADSL หรือผ่านทางสายใยแก้วนำแสง
- 2) ตัวรับ-ส่งสัญญาณ Wireless ทำหน้าที่รับ-ส่ง สัญญาณระหว่างตัวรับส่งแต่ละตัวด้วยกัน หรือระหว่างตัวลูกข่ายกับ Access Point

มาตรฐาน 802.11 ใช้การส่งสัญญาณแบบคลื่นวิทยุที่ความถี่ 2.4 GHz ซึ่งเป็นความถี่ ISM (Industrial, Scientific and Medical) Band สามารถส่งข้อมูลได้ด้วยอัตราความเร็วค่อนข้างต่ำ คือ 1 และ 2 Mbps เท่านั้น โดยใช้เทคนิคการส่งสัญญาณหลักอยู่ 2 รูปแบบ คือ DSSS (Direct Sequent Spread Spectrum) และ FHSS (Frequency Hopping Spread Spectrum) ซึ่งถูกคิดค้นมาจากหน่วยงานทหาร การส่งสัญญาณทั้ง 2 รูปแบบจะใช้ความกว้างของช่องสัญญาณ (Bandwidth) ที่มากกว่าการส่งสัญญาณแบบ Narrow Band แต่ทำให้สัญญาณมีความแรงมากกว่าซึ่งง่ายต่อการตรวจจับมากกว่าแบบ Narrow Band

หน่วยงานทหารใช้วิธีการเหล่านี้ในการปิดกั้นการใช้งานจากอุปกรณ์อื่นๆ ที่จะมาทำให้ระบบเกิดปัญหา โดยการส่งสัญญาณแบบ FHSS สัญญาณจะกระโดดจากความถี่หนึ่งไปยังอีกความถี่หนึ่งในอัตราที่ได้กำหนดไว้แล้ว ซึ่งจะรู้จักเฉพาะตัวรับกับตัวส่งเท่านั้น ส่วนการส่งสัญญาณแบบ DSSS จะมีการส่ง Chipping Code ไปกับสัญญาณแต่ละครั้งด้วย ซึ่งจะมีเฉพาะตัวรับกับตัวส่งเท่านั้นที่จะรู้ลำดับของ Chip

การใช้งานระบบเครือข่ายแบบไร้สายทุกวันนี้ DSSS มีคุณสมบัติที่โดดเด่นและให้ Throughput ที่มากกว่า และจากที่ได้มีการพัฒนาจนได้อัตราการส่งข้อมูล 11 Mbps ผ่านการส่งแบบ DSSS และเป็นมาตรฐานที่โดดเด่นของ WLAN ผลิตภัณฑ์ซึ่งรองรับมาตรฐาน 802.11b (อัตราส่งถ่ายข้อมูลสูง 11 Mbps) นี้สามารถทำงานร่วมกับผลิตภัณฑ์ซึ่งทำงานกับมาตรฐาน DSSS แบบเก่า 802.11 (อัตราส่งถ่ายข้อมูล 1 และ 2 Mbps) ได้ แต่ระบบ FHSS จะถูกใช้กับอุปกรณ์ที่มีกำลังส่งต่ำ หรือเป็น Application ที่ใช้งานในย่านต่ำๆ เช่น โทรศัพท์ไร้สายความถี่ 2.4 GHz แต่จะไม่สามารถใช้งานร่วมกับผลิตภัณฑ์ DSSS ได้

หลังจากที่เทคโนโลยีเครือข่ายไร้สายนี้ได้เกิดขึ้น ก็ได้เกิดมาตรฐานตามมามากมาย การจะเลือกซื้อหรือเลือกใช้อุปกรณ์เครือข่ายไร้สายเหล่านั้น จำเป็นจะต้องคำนึงถึงเทคโนโลยีที่ใช้ในผลิตภัณฑ์นั้นๆ รวมถึงความเข้ากันได้ของเทคโนโลยีที่ต่างๆด้วย

IEEE 802.11 นั้นจะแบ่งระดับชั้นของเทคโนโลยีออกเป็น 4 ระดับ นั่นคือ

1. PHY (Physical Layer หรือ ชั้นกายภาพ)
2. MAC (Media Access Controller หรือตัวควบคุมการเข้าถึงสื่อ)
3. OS (ระบบปฏิบัติการ)
4. Application (แอปพลิเคชัน)

โดยในระดับชั้น PHY หรือชั้นกายภาพนั้น ก็คือส่วนของฮาร์ดแวร์ที่แบ่งมาตรฐานออกเป็น a, b และ g โดยหากเลือกต่างชนิดกันก็ไม่สามารถสื่อสารกันได้ว่าเรื่องเพราะเป็นความถี่ที่ต่างกันจะติดต่อบรรลุส่งข้อมูลกันไม่ได้ โดยปัจจุบันในส่วนของ PHY นี้มีมาตรฐานออกมาหลายอย่าง แต่ที่ได้รับความนิยมทั้งในอดีตและปัจจุบันนั้น แบ่งออกเป็น 7 มาตรฐานด้วยกัน ได้แก่

- 1) **มาตรฐาน IEEE 802.11a หรือ Class a** จะใช้คลื่นความถี่ 5 GHz ในการรับส่งสัญญาณข้อมูลไร้สาย ทำความเร็วสูงสุดที่ 54 Mbps ใช้เทคโนโลยีที่เรียกว่า OFDM (Orthogonal Frequency Division Multiplexing) รองรับอัตราการเร็วของการส่งข้อมูล เท่ากับ 6 , 9 , 12 , 18 , 24 , 36 , 48 และ 54 Mbps อัตราความเร็วในการรับส่งข้อมูลสามารถปรับระดับให้ช้าลงเพื่อเพิ่มระยะทางการเชื่อมต่อให้มากขึ้นได้ แต่ข้อเสียคือ ที่ความถี่ 5 GHz ในหลายประเทศไม่อนุญาตให้ใช้ รวมทั้งประเทศไทย และอุปกรณ์ไร้สายที่รองรับเทคโนโลยี IEEE 802.11a ไม่สามารถเข้ากันได้กับอุปกรณ์ที่รองรับมาตรฐาน IEEE 802.11b และ IEEE 802.11g ได้ รวมทั้งอุปกรณ์ของ IEEE 802.11a ยังมีราคาสูงกว่า IEEE 802.11b ด้วย
- 2) **มาตรฐาน IEEE 802.11b หรือ Class b** จะใช้คลื่นความถี่ 2.4 GHz ในการรับส่งสัญญาณข้อมูลไร้สาย ทำความเร็วสูงสุดที่ 11 Mbps ใช้เทคโนโลยีที่เรียกว่า CCK (Complimentary Code Keying) ผสมกับ DSSS (Direct Sequence Spread Spectrum) เพื่อปรับปรุงความสามารถของอุปกรณ์ให้รับส่งข้อมูลได้ด้วยความเร็วสูงสุด ได้รับการตั้งชื่อใหม่ว่า Wi-Fi โดยได้รับการรับรองมาตรฐานและกำหนดรายละเอียดโดยกลุ่ม WECA หรือ Wireless Ethernet Compatibility Alliance ที่ประกอบด้วยสมาชิกจากผู้ผลิตในอุตสาหกรรมคอมพิวเตอร์ 3com, Cisco Systems, Intersil, Agere Systems, Nokia และ Symbol Technologies ซึ่งปัจจุบันก็ยังมีสมาชิกจากบริษัทต่างๆ อีกกว่า 110 บริษัทเข้าร่วมอยู่ในมาตรฐานนี้
- 3) **มาตรฐาน IEEE 802.11g หรือ Class g** จะใช้คลื่นความถี่ 2.4 GHz ในการรับส่งสัญญาณข้อมูลไร้สาย ทำความเร็วสูงสุดที่ 54 Mbps ได้รับการพัฒนาจากการนำเอาเทคโนโลยี OFDM ของ 802.11a มาพัฒนาจนทำให้ได้ความเร็วที่สูงกว่ามาตรฐาน 802.11b ซึ่ง 802.11g สามารถปรับระดับความเร็วในการสื่อสารลงเหลือ 2 Mbps ได้ตามสภาพแวดล้อมของเครือข่ายที่ใช้งาน จุดเด่นที่สำคัญของ 802.11g ก็คือสามารถใช้งานร่วมกับ 802.11b ที่มีอยู่แล้วได้ มาตรฐานนี้เป็นที่ยอมรับจากผู้ใช้เป็นจำนวนมากและกำลังจะเข้ามาแทนที่ 802.11b ในอนาคตอันใกล้
- 4) **มาตรฐาน IEEE 802.11n หรือ Class n** จะใช้คลื่นความถี่ คือ 2.4 GHz และ 5 GHz ในการรับส่งสัญญาณข้อมูลไร้สาย ทำความเร็วสูงสุดที่ 150 Mbps และ 300 Mbps มีความสามารถในการส่งคลื่นสัญญาณได้ระยะประมาณ 70 เมตรในโครงสร้างปิด และ 250 เมตรในที่โล่งแจ้ง เพิ่มความสามารถในการกันสัญญาณรบกวนจากอุปกรณ์อื่นๆ ที่ใช้ความถี่ 2.4GHz เหมือนกัน และ

สามารถรองรับอุปกรณ์มาตรฐาน IEEE 802.11b และ IEEE 802.11g ได้ ซึ่งช่วงระยะหลังได้มีการพัฒนาการส่งสัญญาณแบบ "Dual-Band" หรือการใช้คลื่นความถี่ในย่าน 2.4 GHz และ 5 GHz ในการรับส่งสัญญาณ (จะใช้เสามากกว่า 1 ต้นขึ้นไป) ทำให้สามารถทำความเร็วได้สูงถึง 600 Mbps หรือเรียกสั้นๆ ว่า N600

- 5) มาตรฐาน IEEE 802.11-2012 ในปี 2007 กลุ่มงาน TGmb ได้รับการอนุมัติให้รวบรวมการแก้ไขทั้งหมดให้เป็นเวอร์ชันที่เรียกว่า REVmb หรือ 802.11 mb ที่ประกอบด้วย 802.11k, r, y, n, w, p, z, v, u, s ตีพิมพ์วันที่ 29 มีนาคม 201
- 6) มาตรฐาน IEEE 802.11ac หรือ Class ac จะใช้คลื่นความถี่ 5 GHz ในการรับส่งข้อมูลไร้สาย เป็นมาตรฐานที่ให้ Throughput กับ Wireless LAN แบบหลายสถานี สูงกว่าที่อย่างน้อย 1 Gbps และสำหรับลิงก์เดี่ยวที่อย่างน้อย 500 Mbps โดยการใช้ RF แบนด์วิธที่กว้างกว่า (80 หรือ 160 MHz) สตรีมมากกว่า (สูงถึง 8 สตรีม) และ Modulation ที่ความจุสูงกว่า (สูงถึง 256 QAM) โดย Class ac เป็นมาตรฐานใหม่ล่าสุดที่ได้รับการรับรองจาก IEEE ให้เป็นมาตรฐานใหม่เมื่อ ปี ค.ศ. 2013 ซึ่งมาตรฐานในการรับส่งข้อมูลแบบไร้สายของ Class ac สามารถทำความเร็วได้สูงถึง 6,930 Mbps หรือประมาณ 6.93 Gbps
- 7) มาตรฐาน IEEE 802.11ad หรือ Class ad หรือ "WiGig" เกิดจากการผลักดันจากผู้ผลิตฮาร์ดแวร์ ในวันที่ 24 กรกฎาคม 2012 Marvell และ Wilocity ได้ประกาศการเป็นคู่ค้าใหม่เพื่อนำ Wi-Fi Solution แบบ Tri-Band ใหม่ออกสู่ตลาด โดยการใช้ความถี่ที่ 60 GHz Throughput ทางทฤษฎีสูงสุดถึง 7 Gbps มาตรฐานนี้ได้ออกสู่ตลาดในช่วงเมื่อต้นปี 2014

2.4 มาตรฐานที่เกี่ยวข้อง

LOGO!8 นั้นได้รับใบรับรองมาตรฐาน ผ่านการทดสอบการใช้งานของ CE (European Conformity) ซึ่งสามารถใช้งานได้กับที่พักอาศัย อุตสาหกรรม และอาคารพาณิชย์ โดยเป็นไปตามมาตรฐานดังต่อไปนี้

2.4.1 EN 61000-6-1 ได้นำมาใช้ได้สำหรับอุปกรณ์ไฟฟ้าที่ถูกใช้หรือมีจุดประสงค์ใช้ในที่พักอาศัย, เกี่ยวกับการค้าและอุตสาหกรรมเบา ที่เกี่ยวข้องกับเครื่องมือทางไฟฟ้าและอุปกรณ์อิเล็กทรอนิกส์ ถ้าหากสินค้าไม่มีมาตรฐานสินค้าโดยเฉพาะหรือสินค้านั้นไม่มีมาตรฐานใดๆ จะสามารถใช้มาตรฐาน EN 61000-6-1:2007 ได้ โดยมาตรฐานนานาชาติ EN 61000-6-1:2007 นำมาใช้ได้สำหรับอุปกรณ์ที่สามารถเชื่อมต่อกับเครือข่ายแรงดันไฟฟ้าต่ำสาธารณะหรือเครือข่าย DC สาธารณะที่ซึ่งถูกใช้ร่วมกันโดยอุปกรณ์ไฟฟ้าอื่นๆ ได้ มาตรฐานนี้ยังใช้กับอุปกรณ์ที่ใช้พลังงานแบตเตอรี่ได้อีกด้วย ที่พักอาศัย การค้า และอุตสาหกรรมเบาทั้งในที่ร่มและกลางแจ้งสามารถเป็นได้ดังนี้

- Flats, Apartments, Studios, บ้าน
- ซูเปอร์มาร์เก็ต, ร้านค้า, ร้านอาหาร
- ธนาคาร, บริษัท, พื้นที่ธุรกิจอื่นๆ
- โรงภาพยนตร์, สโมสร, บาร์
- ศูนย์รวมกีฬา, ปั๊มน้ำมัน, พื้นที่จอดรถ
- ห้องปฏิบัติการแลป, ห้องทำงาน

มาตรฐานนานาชาติ EN 61000-6-1:2007 ไม่มีการจัดแบ่งประเภทสำหรับอุปกรณ์ที่ครอบคลุมตัวอย่างเช่น แหล่งจ่ายไฟฟ้าทั่วไป, เครื่องทำความร้อนไฟฟ้า, สัญลักษณ์ในที่จอดรถ, อุปกรณ์, เครื่องช่วยหายใจ, อุปกรณ์ช่าง, โซฟาและเก้าอี้แนวไฟฟ้า, วิทยู เป็นต้น

2.4.2 EN 61000-6-2 นำมาใช้สำหรับอุปกรณ์ไฟฟ้าซึ่งถูกใช้หรือมีจุดประสงค์ใช้ในสภาพแวดล้อมทางอุตสาหกรรมที่ใช้โดยหม้อแปลงแรงดันไฟฟ้าสูงหรือปานกลาง หรือสำหรับอุปกรณ์ที่มีจุดประสงค์เพื่อให้ทำงานใกล้เคียงกับสภาพแวดล้อมทางอุตสาหกรรม ถ้าสินค้าไม่มีมาตรฐานสินค้าโดยเฉพาะหรือมาตรฐานสินค้าในเครือ EN 61000-6-2:2005 สามารถนำมาใช้ได้มาตรฐานนี้ยังสามารถใช้กับอุปกรณ์พลังงานแบตเตอรี่ได้อีกด้วยถ้าอุปกรณ์มีจุดประสงค์ไว้ใช้ในสภาพแวดล้อมทางอุตสาหกรรมหรือบริเวณใกล้เคียงกัน

มาตรฐานนานาชาติ EN 61000-6-2:2005 ไม่มีการจัดแบ่งประเภทสำหรับอุปกรณ์ที่ครอบคลุมตัวอย่างเช่น แหล่งจ่ายไฟ, เครื่องช่าง, เครื่องช่วยหายใจ, เครื่องทำความร้อน, ระบบปั๊มน้ำ, สายพาน, อุปกรณ์วิเคราะห์การทำเหมือง, สว่านไฟฟ้า เป็นต้น

บทที่ 3

การออกแบบและการจำลอง




ในการทำระบบบ้านอัตโนมัติหรือ Smart Home นั้นจะแบ่งการดำเนินงานหลักๆ เป็น 4 ส่วนด้วยกัน คือ ส่วนของการออกแบบแปลนบ้าน, การออกแบบฟังก์ชันของบ้าน, การเขียนโปรแกรมของ LOGO!8 และการเขียนโปรแกรมของ HMI ซึ่งสาเหตุที่ต้องแบ่งการดำเนินงานเป็น 4 ส่วนนั้นเพราะว่าในการทำ Smart Home เราต้องมีแบบจำลองของตัวบ้าน เพื่อใช้ในการจำลองการทำงานของระบบ การสั่งการทำงานหรือดูค่าพารามิเตอร์ต่างๆ ของ LOGO!8 นั้นจะดูค่าและปรับได้ยาก จึงต้องใช้ HMI ในการดูค่าแสดงผลต่างๆ และปรับค่าพารามิเตอร์ๆ ของ LOGO!8 ได้สะดวกและง่ายต่อการสั่งการ ซึ่งจะช่วยให้ผู้ใช้งานเข้าใจและปรับค่าได้ง่ายขึ้น

3.1 ออกแบบแปลนบ้าน

ในหัวข้อนี้จะกล่าวถึงการออกแบบแปลนบ้าน ซึ่งจะถูกจัดสรรพื้นที่ต่างๆ ที่ของบ้านตามต้องการ รวมไปถึงสัญลักษณ์ต่างๆ ของเครื่องใช้ไฟฟ้าที่อยู่ภายในและภายนอกบ้านที่ใช้ LOGO!8 ในการควบคุมการทำงาน เพื่อนำไปใช้ในการจำลองการทำงานของระบบ Smart Home

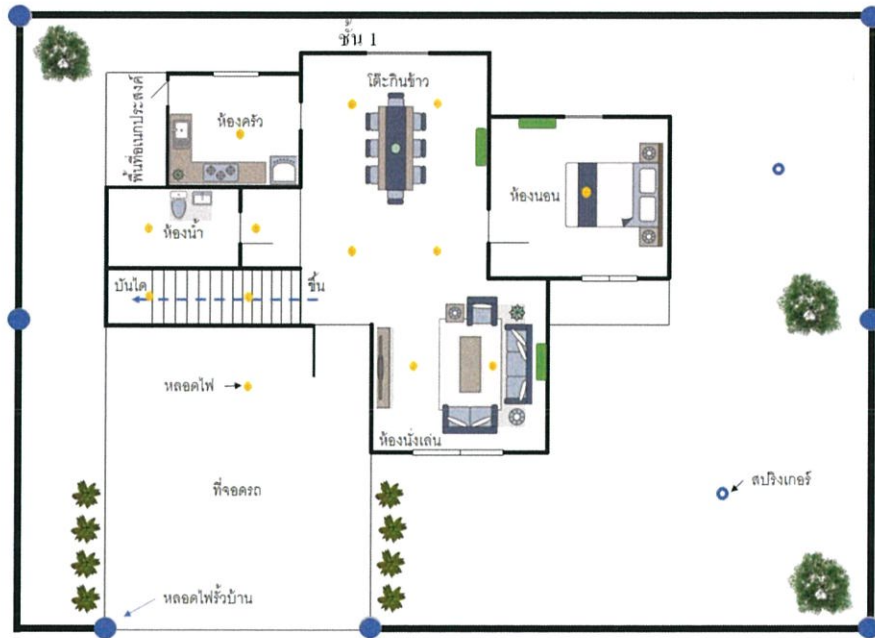
3.1.1 การกำหนดสัญลักษณ์

ตารางที่ 1 การกำหนดสัญลักษณ์ในแปลนบ้าน

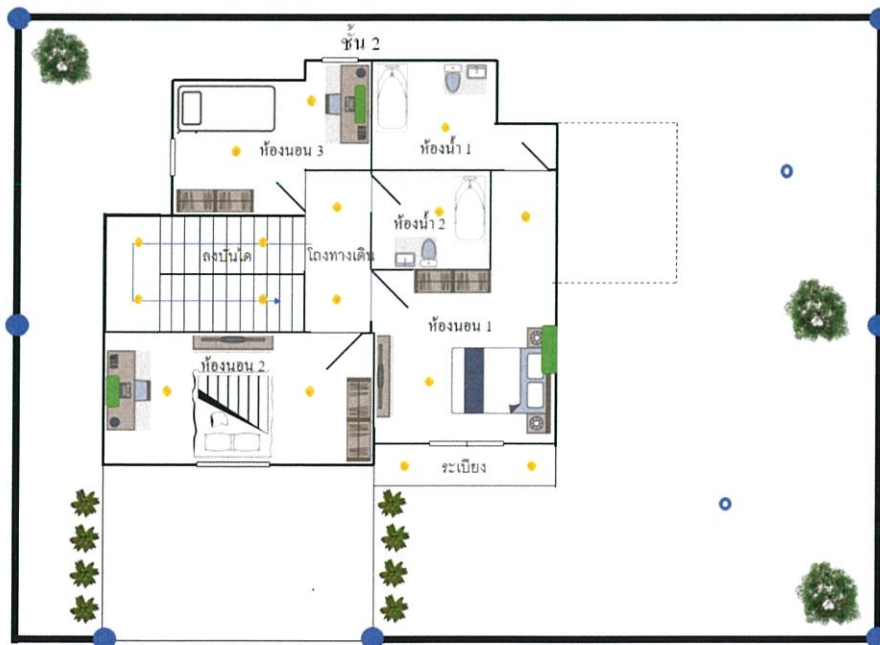
| ลำดับ | สัญลักษณ์ | คำอธิบาย |
|-------|---|-------------------|
| 1 |  | หลอดไฟ |
| 2 |  | หลอดไฟรอบรั้วบ้าน |
| 3 |  | สปริงเกอร์ |
| 4 |  | เครื่องปรับอากาศ |

3.1.2 วาดแปลนบ้านและจำลองการวางอุปกรณ์ภายในบ้านและนอกบ้าน

การวาดแปลนบ้านจะออกแบบให้เป็นบ้าน 2 ชั้น มี 4 ห้องนอน 3 ห้องน้ำ ซึ่งแต่ละห้องของบ้าน จะมีอุปกรณ์ไฟฟ้าต่างๆ อยู่ ที่ให้ LOGO18 สามารถควบคุมได้



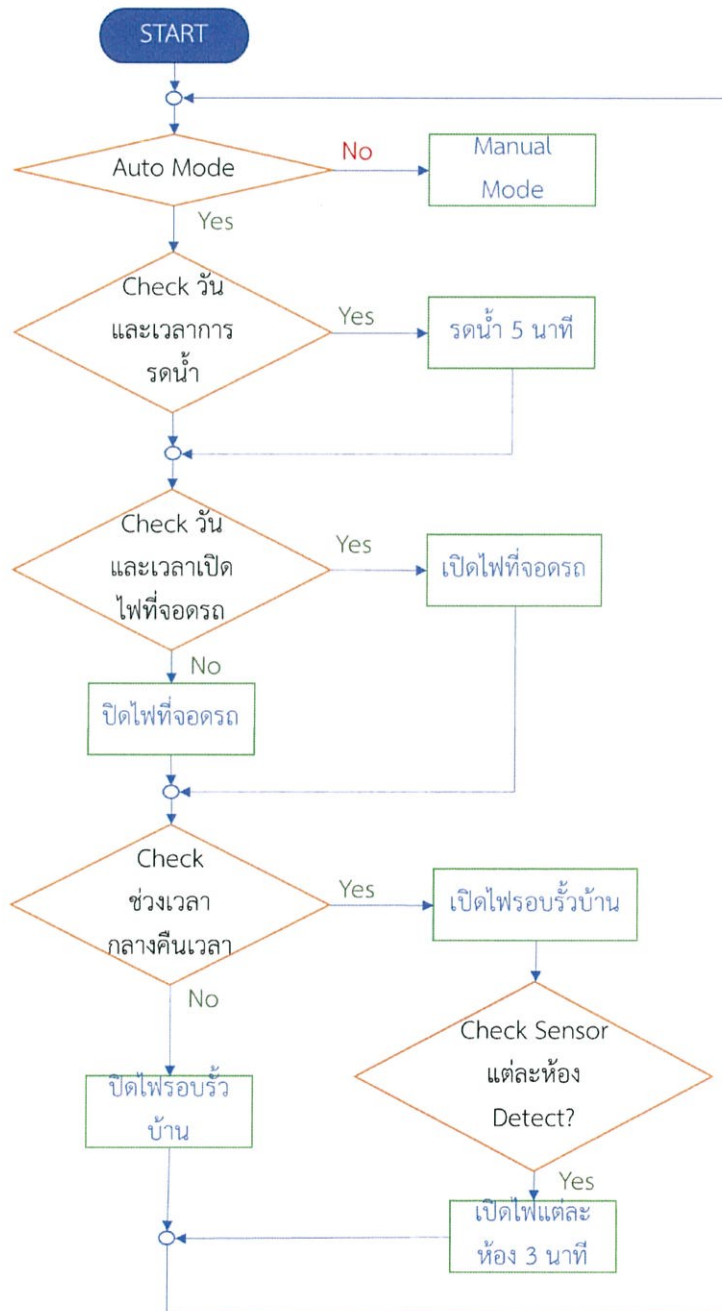
รูปที่ 3.1 แปลนบ้านชั้น 1



รูปที่ 3.2 แปลนบ้านชั้น 2

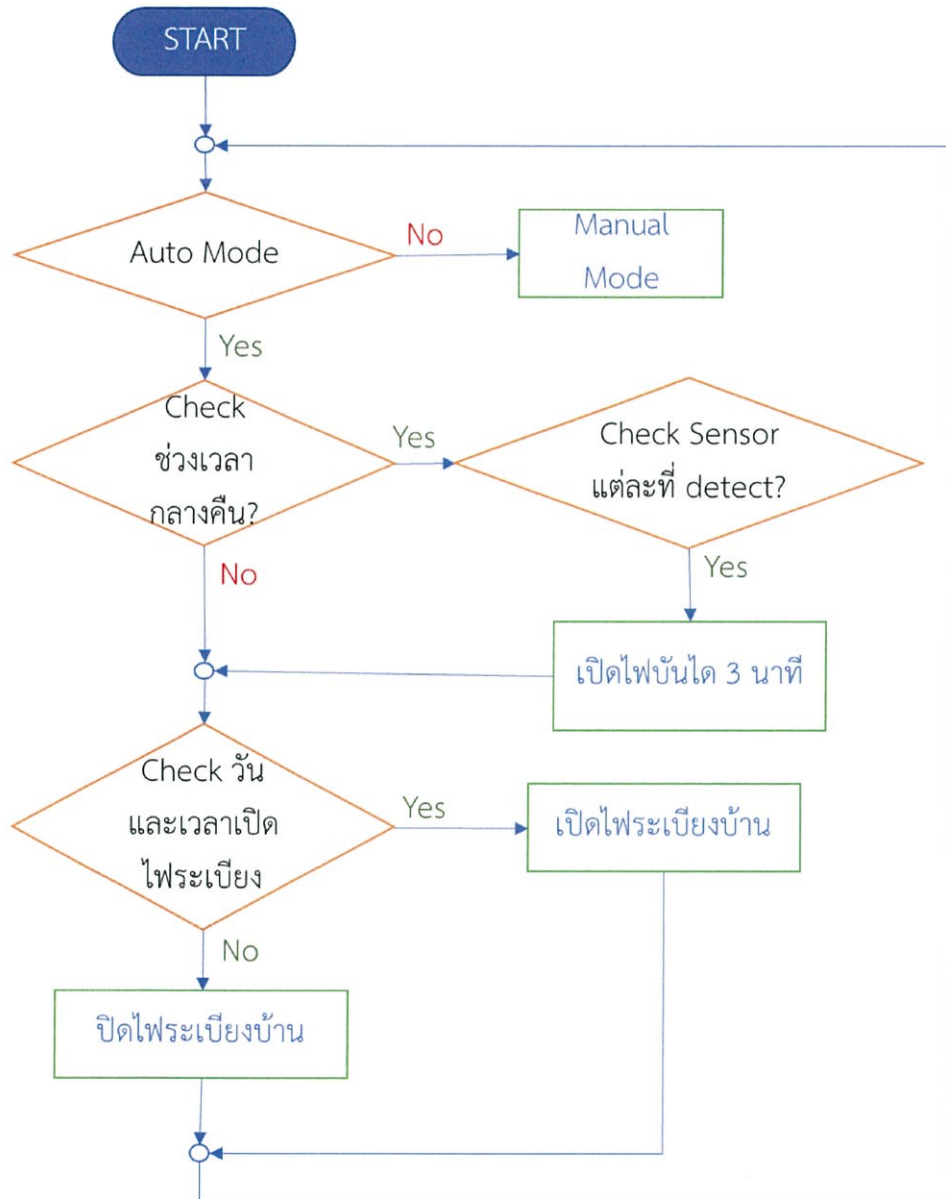
3.2 ออกแบบ Functional ของ Smart Home

ในหัวข้อนี้จะเป็นข้อกำหนดการทำงานและเงื่อนไขการทำงานของระบบไฟฟ้าภายในบ้านและนอกบ้าน ซึ่งจะใช้ LOGO!8 2 ตัวในการควบคุมระบบไฟฟ้า โดยแบ่งเป็น 2 ชั้น แต่ละชั้นก็จะมี LOGO!8 1 ตัวและมี HMI เป็นอุปกรณ์ที่ใช้สั่งงานและตั้งค่าการทำงานของ LOGO!8 โดยจะมีอยู่ 2 โหมดด้วยกัน คือ โหมด Manual และ โหมดอัตโนมัติ ซึ่งชั้นที่ 1 จะมี Flowchart ดังนี้



รูปที่ 3.3 Flowchart ฟังก์ชันของบ้านชั้นที่ 1

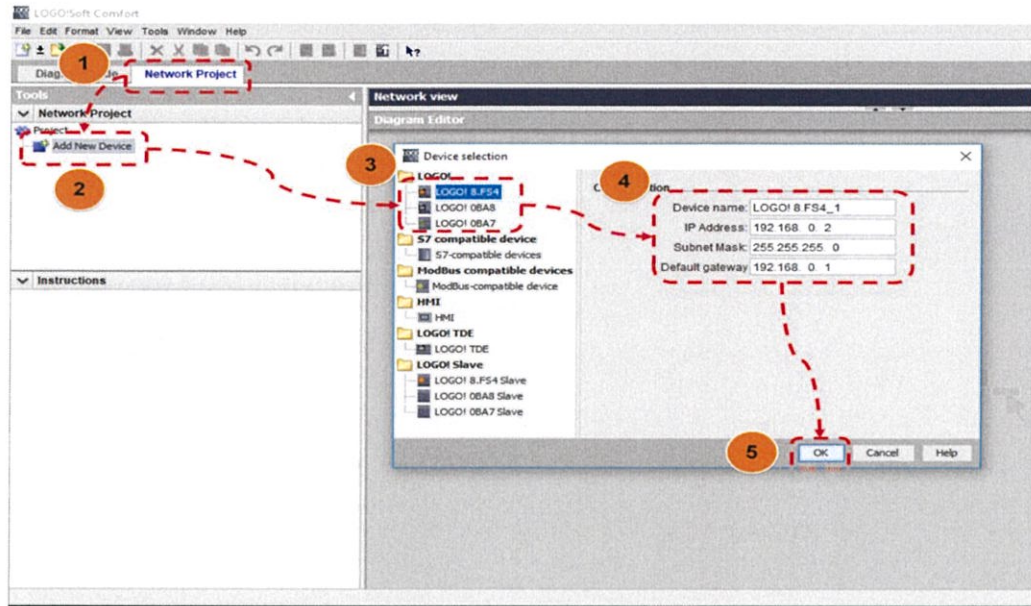
Flowchart ชั้น 2 มีดังนี้



รูปที่ 3.4 Flowchart ฟังก์ชันของบ้านชั้นที่ 2

3.3 การเขียนโปรแกรม LOGO!8

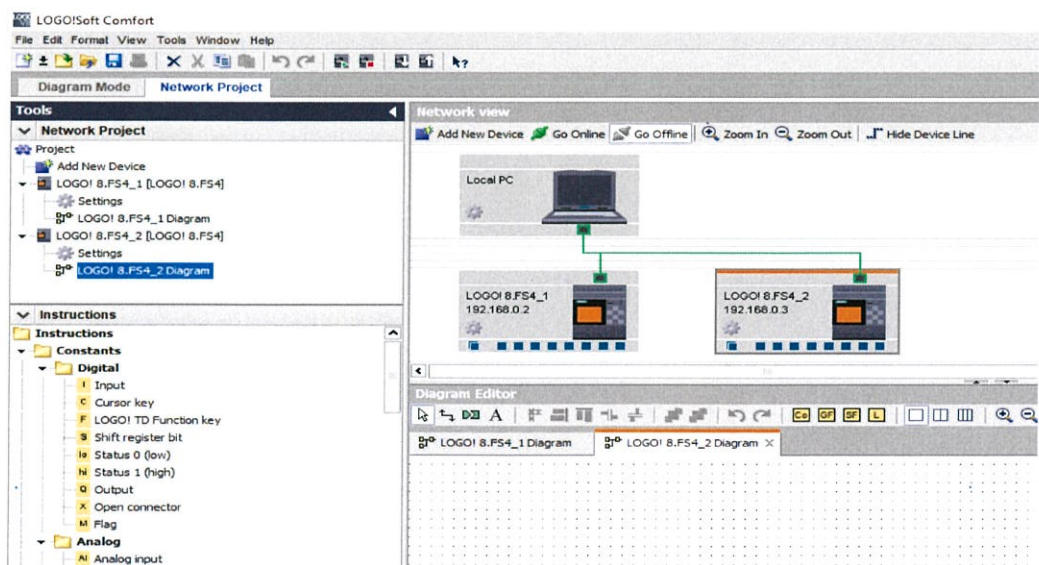
ในหัวข้อนี้จะอธิบายถึงการเขียนโปรแกรมของ LOGO!8 โดยใช้โปรแกรม LOGO! Soft comfort ซึ่งจะเริ่มต้นตั้งแต่การเพิ่ม Device เข้ามาในโปรเจคไปจนถึงการ Commissioning อันดับแรกให้ทำการเพิ่ม LOGO!8 ชั้นมา 2 ตัว ซึ่งจะเพิ่มตามขั้นตอนดังรูปที่ 3.3



รูปที่ 3.5 การเพิ่มอุปกรณ์

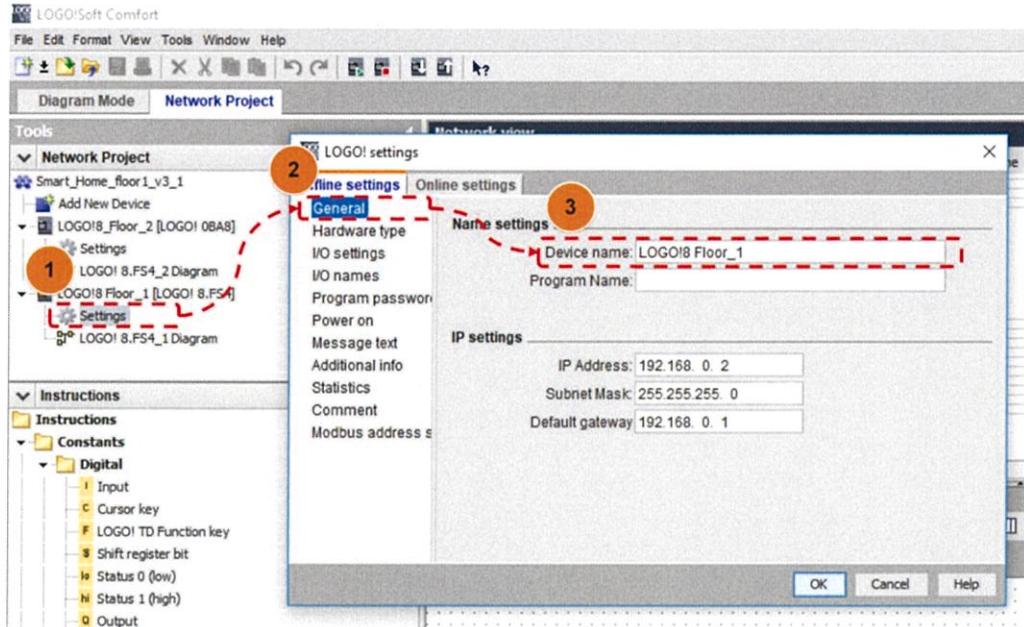
ขั้นตอนการเพิ่มอุปกรณ์ มีดังนี้

- 1)ไปที่ Network Project
- 2)ทำการคลิกเพิ่มอุปกรณ์
- 3)เลือกอุปกรณ์ที่ต้องการ ในที่นี่จะใช้เป็น LOGO! 8.FS4
- 4)ทำการกำหนดชื่อและ IP Address ของอุปกรณ์
- 5)กด OK เป็นการเสร็จสิ้นการเพิ่มอุปกรณ์

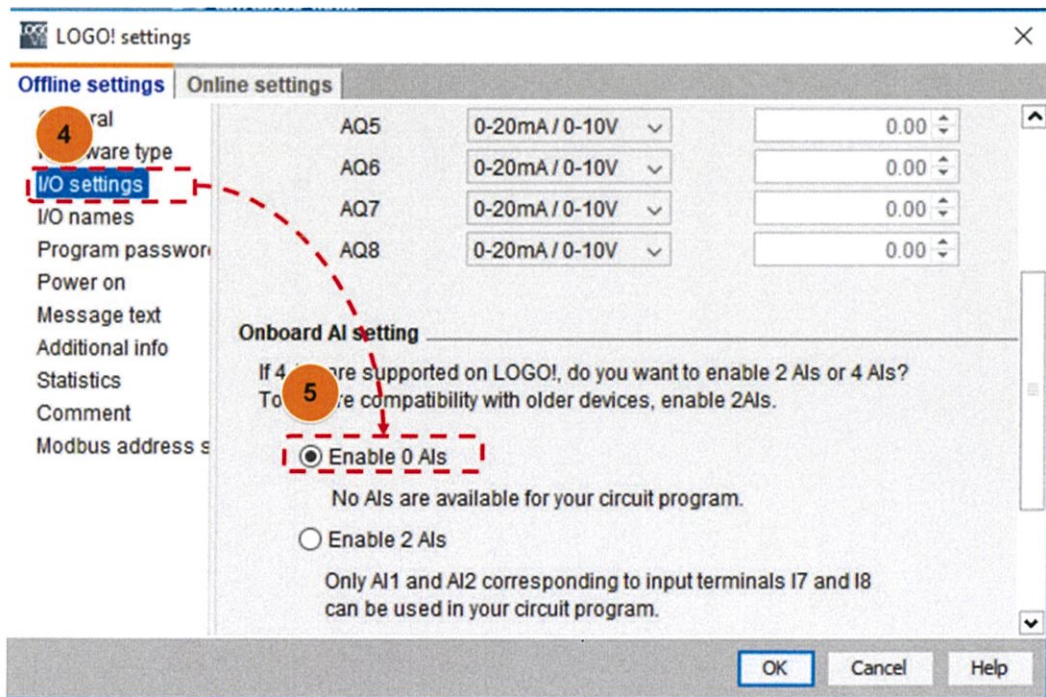


รูปที่ 3.6 แสดงหลังจากการเพิ่ม LOGO!8 2 ตัว

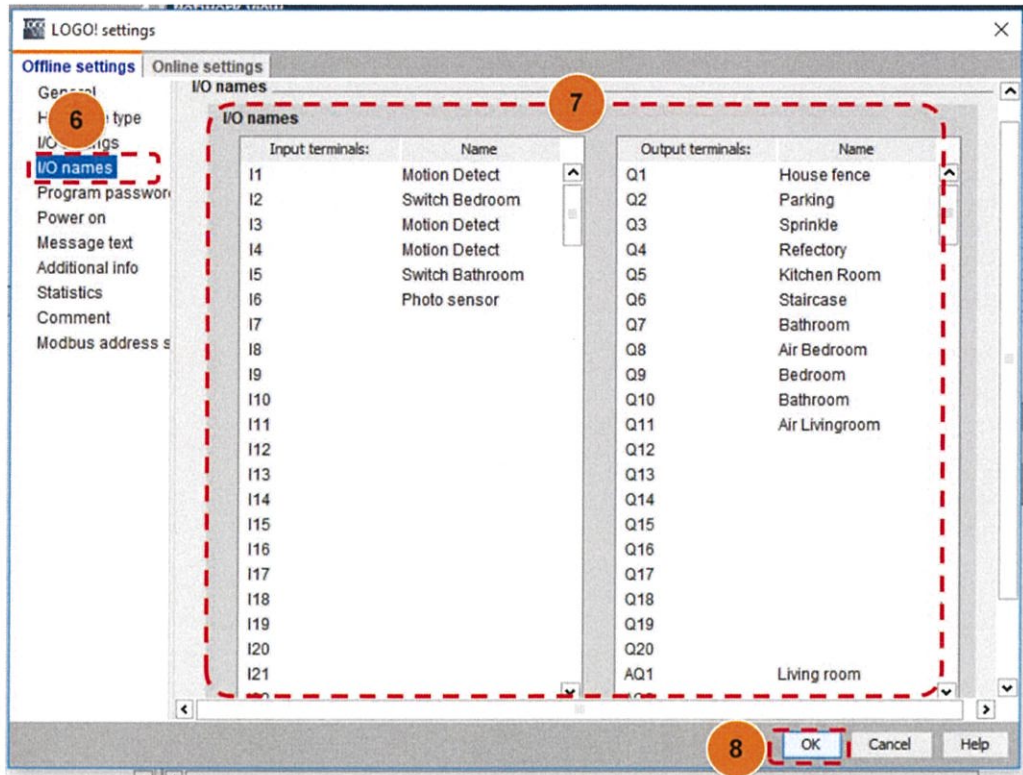
หลังจากทำการเพิ่มอุปกรณ์เข้ามาในโปรเจกต์แล้ว ต่อไปจะเป็นการตั้งค่าต่างๆ ของ LOGO!8 ทั้ง 2 ตัว รวมไปถึงการกำหนดชื่อของอุปกรณ์และ I/O ของอุปกรณ์ ซึ่งจะเป็นไปตามขั้นตอนตามรูปที่ 3.7 ถึงรูปที่ 3.9



รูปที่ 3.7 ตั้งค่าการเปลี่ยนชื่อและ IP



รูปที่ 3.8 การตั้งค่า I/O ของอุปกรณ์

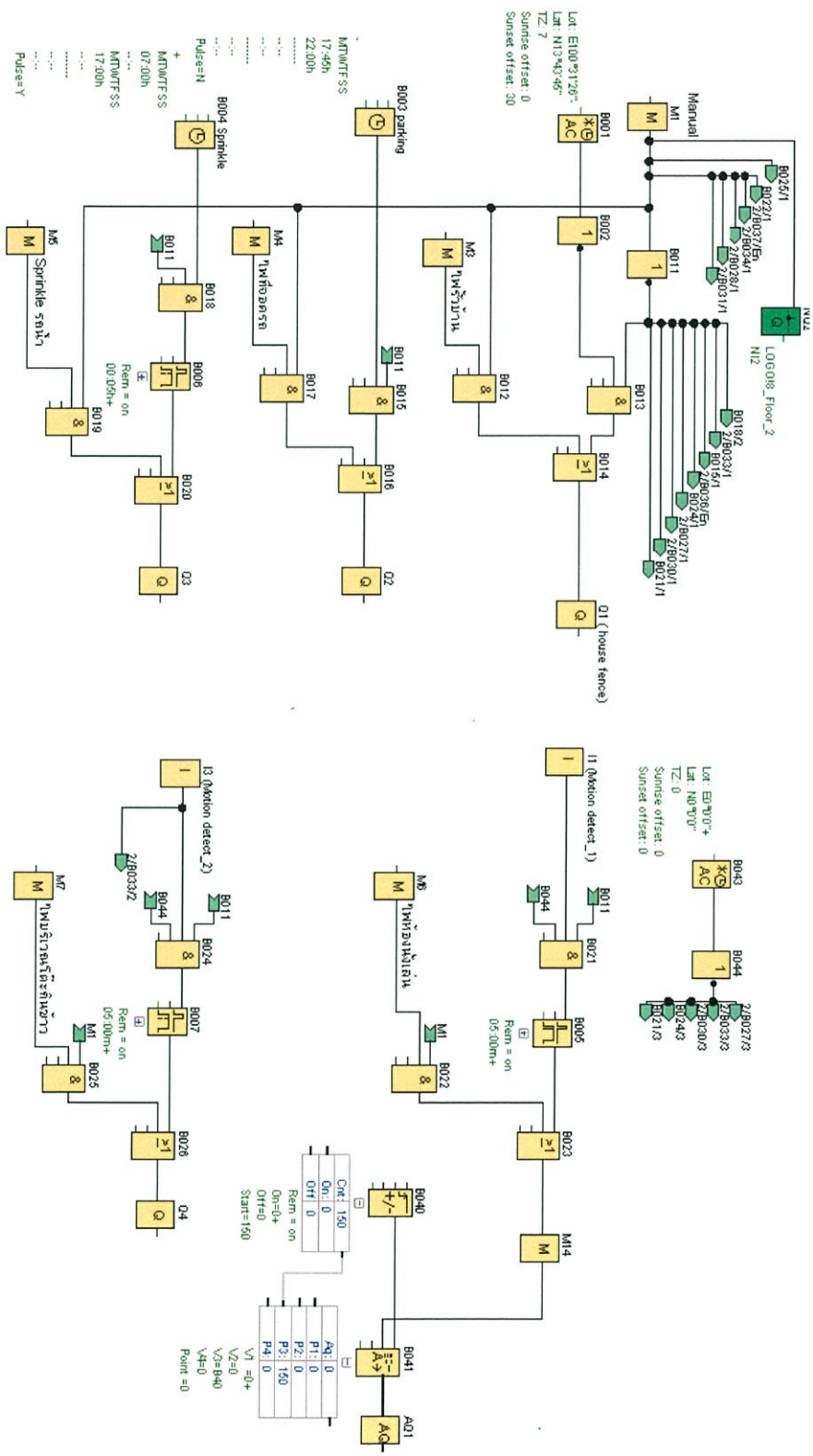


รูปที่ 3.9 การกำหนดชื่อ I/O ของอุปกรณ์

ขั้นตอนการตั้งค่าอุปกรณ์ มีดังนี้

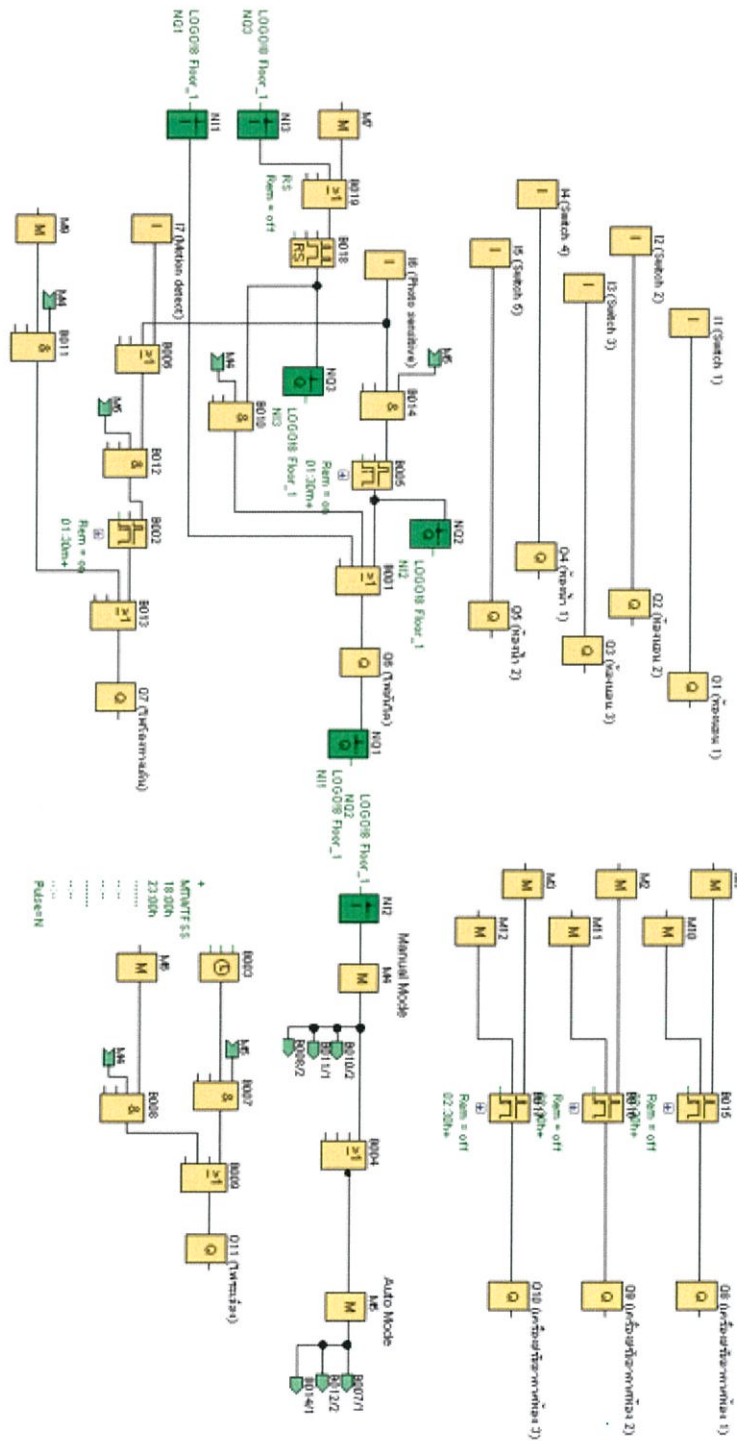
- 1) ไปที่ Setting
- 2) จะมีหน้าต่างการตั้งค่าขึ้นมาแล้วกด General
- 3) เปลี่ยนชื่อของอุปกรณ์
- 4) ไปที่ I/O Setting เพื่อกำหนด I/O ของอุปกรณ์
- 5) จากนั้นไปเลือก Enable 0 Als
- 6) ไปที่ I/O name
- 7) กำหนดชื่อให้กับ I/O ของอุปกรณ์ เพื่อให้ง่ายต่อการเขียนโปรแกรม
- 8) กด OK เพื่อสิ้นสุดการตั้งค่าของอุปกรณ์

หลังจากเราทำการตั้งค่าของอุปกรณ์เสร็จแล้ว ต่อไปจะเป็นการเขียนโปรแกรมตาม Flowchart ที่เราได้กำหนดขึ้นมา ในการเขียนโปรแกรมจะใช้ภาษา Function block ในการเขียน โดยเราสามารถเขียนได้ง่ายๆ โดยการลากแล้ววางตาม Basic programming ในบทที่ 2 ซึ่ง LOGO!8 ของบ้านชั้นที่ 1 จะมีโปรแกรมดังนี้



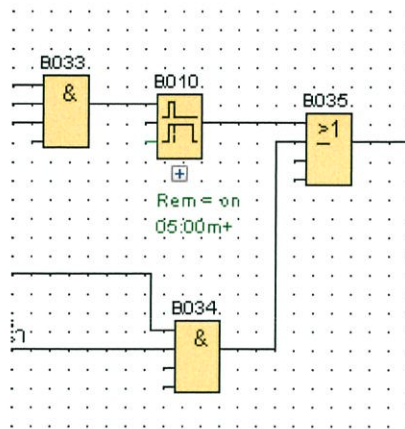
รูปที่ 3.10 โปรแกรมของบ้านชั้นที่ 1 ส่วนที่ 1

และชั้นที่ 2 ของบ้าน จะมีโปรแกรมดังนี้



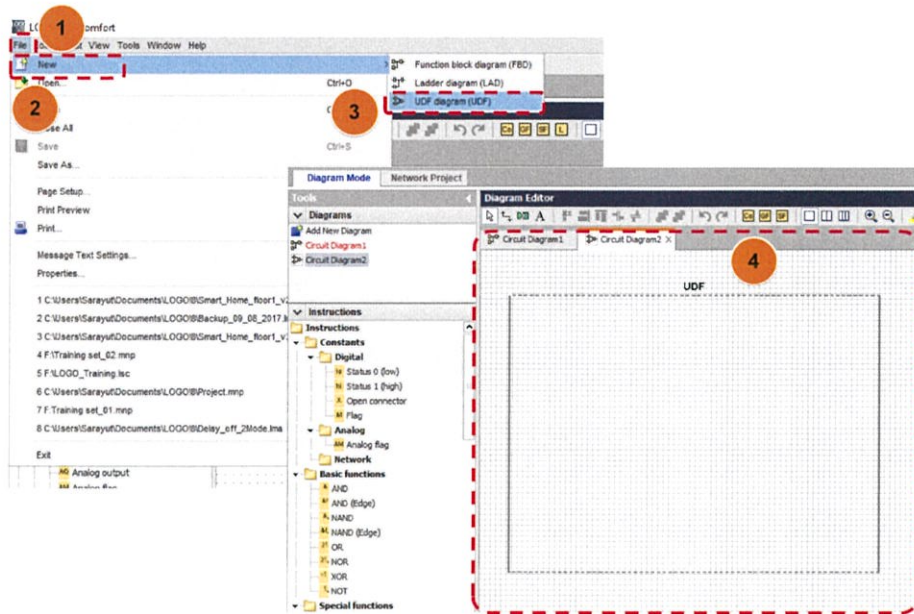
รูปที่ 3.12 โปรแกรมของบ้านชั้นที่ 2

จากโปรแกรมข้างต้นจะเห็นว่าโปรแกรมที่เขียนจะค่อนข้างเยอะพอสสมควร ถ้าหากโปรแกรมที่เราเขียนมีปัญหาหรือแก้ไขเพิ่มเติมจะทำให้แก้ไขได้ลำบาก จากโปรแกรมข้างต้นจะเห็นว่าเงื่อนไขในการทำงานของโปรแกรมนั้นมีส่วนที่เหมือนกันนั่นคือ ไฟห้องนั่งเล่น, ไฟห้องครัว, ไฟห้องโถะทานข้าว, สปริงเกอร์รดน้ำ และไฟหน้าห้องน้ำ ซึ่งโปรแกรมในส่วนเหล่านี้จะเขียนโปรแกรมควบคุมเหมือนกัน โดยจะมีฟังก์ชันดังรูปที่ 3.8



รูปที่ 3.13 ฟังก์ชันที่เหมือนกันของโปรแกรม

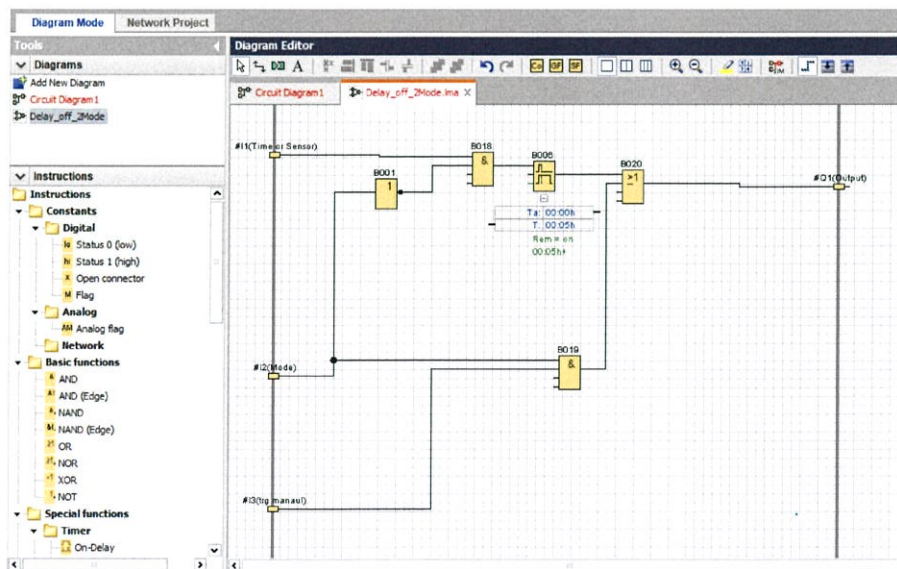
ฟังก์ชันที่เหมือนกันเหล่านี้ เราสามารถทำให้ฟังก์ชันนี้ดูง่ายขึ้นโดยการใช้ UDF (User Defined Function) โดยที่ UDF จะให้ผู้ใช้งานทำการเขียนโปรแกรมให้เป็นโปรแกรมที่ต้องการแล้วสามารถบันทึกโปรแกรมนั้นให้เป็น Function block ได้ อีกทั้งยังกำหนดพารามิเตอร์ที่ต้องการให้ปรับเปลี่ยนหรือไม่ต้องการก็สามารถทำได้ ถ้าหากเราไม่ต้องการให้ผู้ใช้งานคนอื่นมาแก้ไข UDF ของเรา เราสามารถตั้ง Password เพื่อป้องกันการดูหรือแก้ไขฟังก์ชันของผู้ใช้งานอีกด้วย การเขียนโปรแกรม UDF จะมีขั้นตอนดังรูปที่ 3.14



รูปที่ 3.14 การเพิ่ม UDF diagram

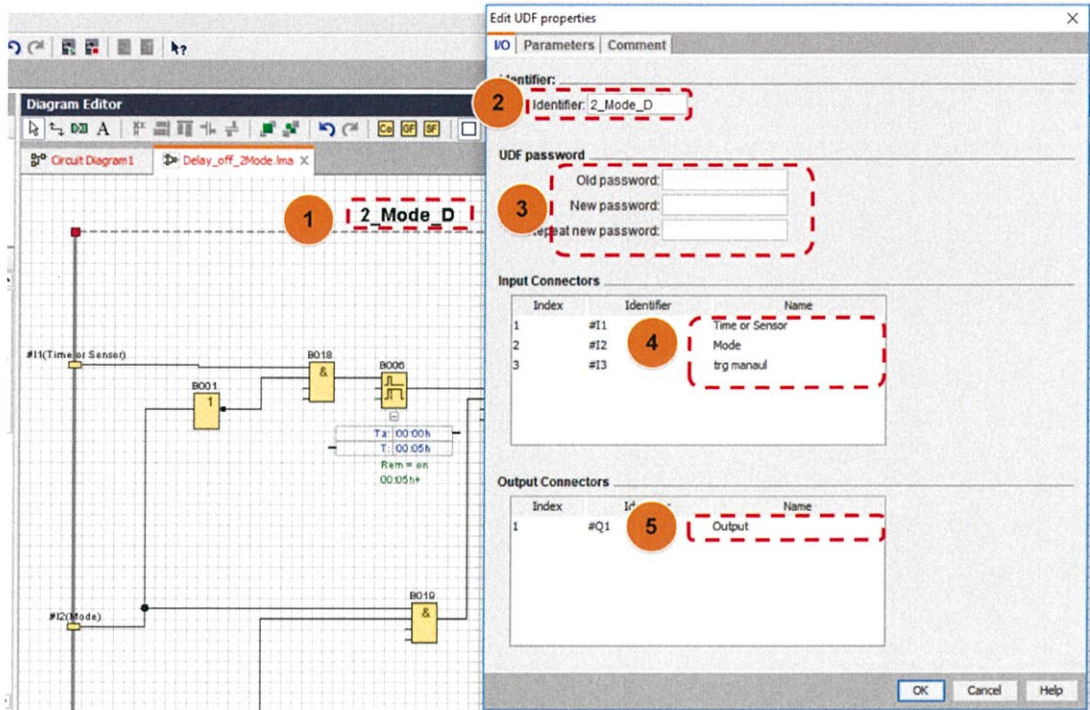
ขั้นตอนการเพิ่ม UDF diagram

- 1) ไปที่ File
- 2) จากนั้นให้ทำการเลือก New
- 3) เลือก UDF diagram
- 4) หลังจากเลือกจะมีหน้าต่าง UDF diagram ขึ้นมา

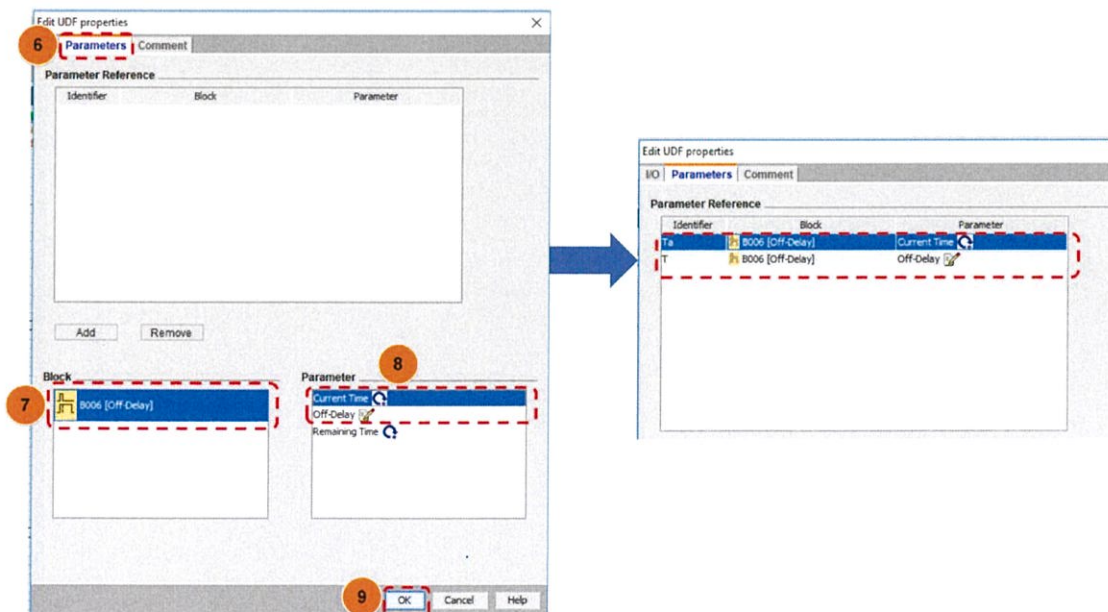


รูปที่ 3.15 เขียนโปรแกรมที่เหมือนกันลง UDF

หลังจากเพิ่ม UDF diagram ขึ้นมาแล้วให้ทำการเขียนโปรแกรมฟังก์ชันที่เหมือนกันลงไป
 ใน UDF diagram ตามรูปที่ 3.15 จากนั้นจะเป็นการตั้งค่าของ UDF ที่สร้างขึ้นมา โดยมีขั้นตอนดังรูปที่ 3.16
 และ 3.17



รูปที่ 3.16 การตั้งชื่อ I/O และตั้ง password

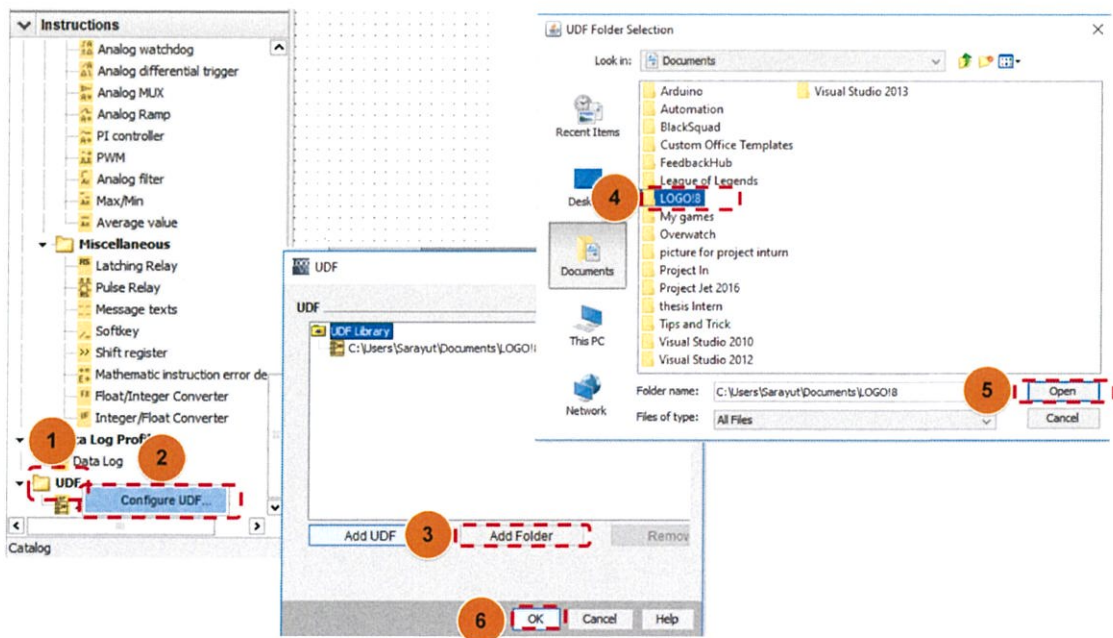


รูปที่ 3.17 การเพิ่มพารามิเตอร์ที่ต้องการ

ขั้นตอนการตั้งค่า UDF diagram

- 1) ดับเบิลคลิกที่ชื่อของ UDF diagram
- 2) ตั้งชื่อของฟังก์ชันนี้
- 3) กำหนด password
- 4) ตั้งชื่อของ Input ของฟังก์ชัน
- 5) ตั้งชื่อของ Output ของฟังก์ชัน
- 6) ไปที่พารามิเตอร์
- 7) คลิกเลือก function block ที่ต้องการ
- 8) ทำการเพิ่มพารามิเตอร์ที่ต้องการ
- 9) หลังจากเพิ่มเสร็จให้กด OK เพื่อเสร็จสิ้น

หลังจากทำการกำหนดการตั้งค่าต่างๆ ของ UDF เรียบร้อยแล้ว ให้ทำการบันทึก UDF diagram เมื่อทำการบันทึกเรียบร้อยแล้ว วิธีที่จะสามารถนำฟังก์ชัน UDF ที่เราสร้างขึ้นมาแสดงเพื่อใช้งานจาก Instructions สามารถทำได้โดยขั้นตอนตามรูปที่ 3.18

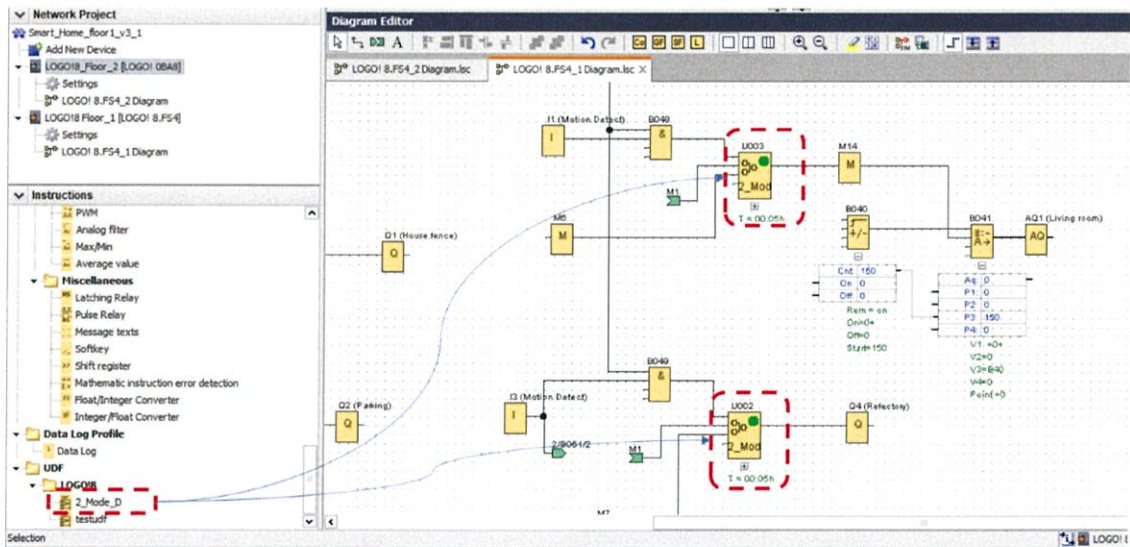


รูปที่ 3.18 วิธีนำ UDF มาใช้งาน

ขั้นตอนการนำ UDF มาใช้งาน

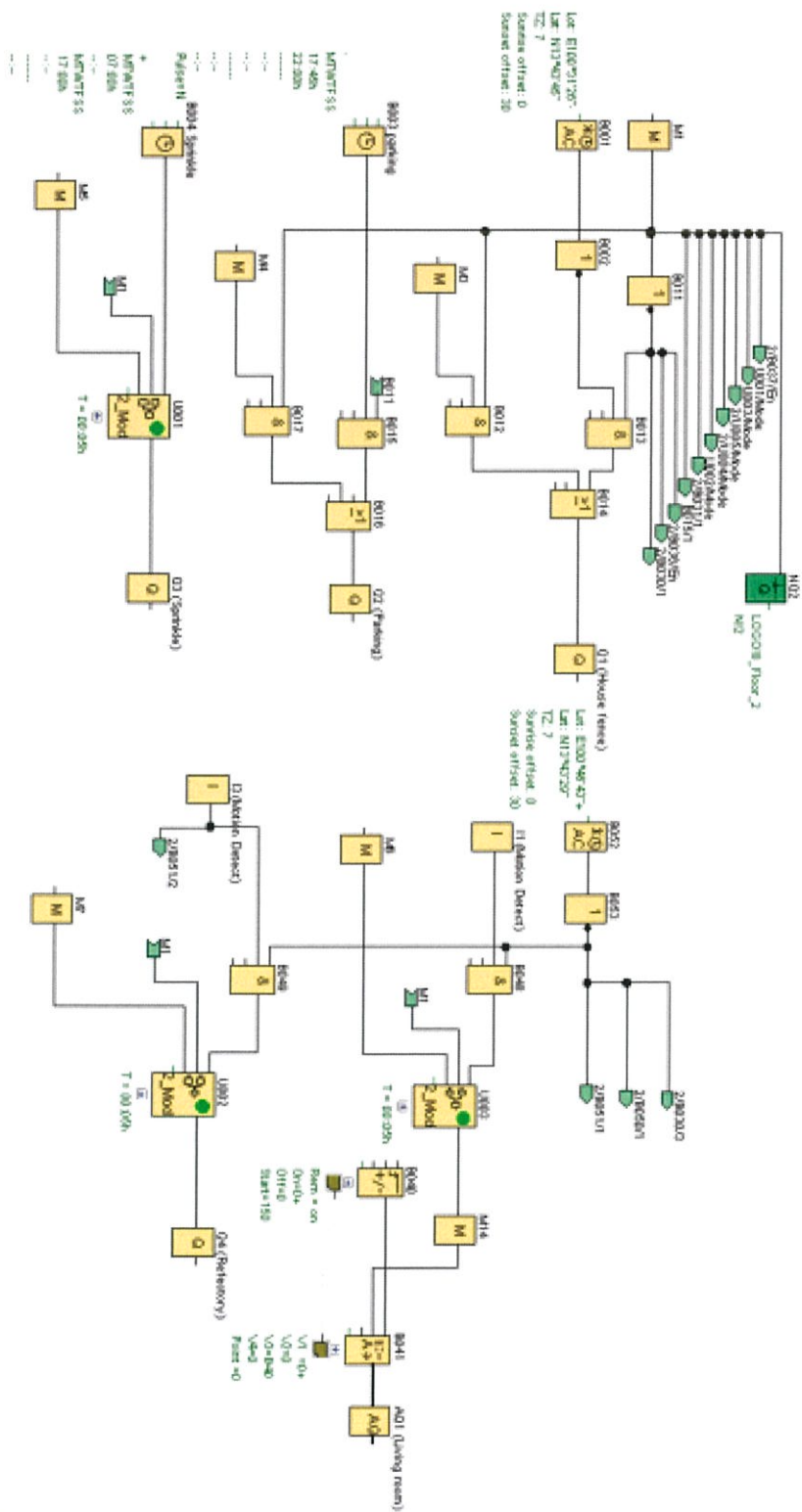
- 1) คลิกขวาที่ UDF
- 2) เลือก Configure UDF
- 3) เลือก Add Folder
- 4) ให้เราเลือก Folder ที่เราบันทึก UDF diagram เอาไว้
- 5) คลิก Open
- 6) คลิก OK

เมื่อเราทำการเรียกใช้งานฟังก์ชัน UDF แล้ว ฟังก์ชัน UDF ที่เราสร้างขึ้นมาจะอยู่ในส่วนของ Instructions ผู้ใช้งานสามารถลากฟังก์ชัน UDF แล้วนำไปวางแก้ไขในโปรแกรมได้เลยทันทีดังในรูปที่ 3.19

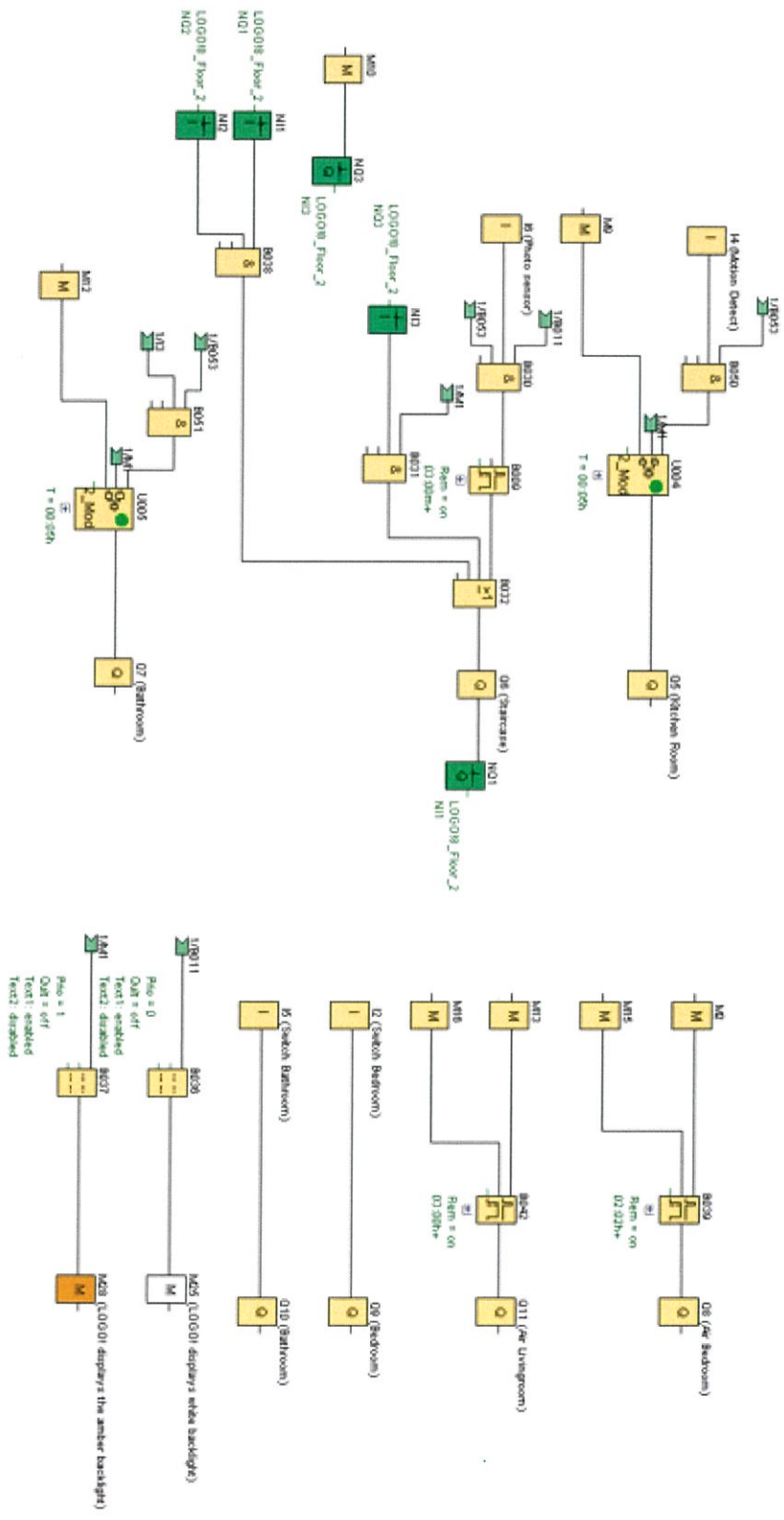


รูปที่ 3.19 การนำฟังก์ชัน UDF มาแก้ไขโปรแกรม

หลังจากเราทำการแก้ไขโปรแกรมทั้ง 2 ชั้นของบ้านแล้วจะได้โปรแกรมใหม่ที่ดูง่ายขึ้น โดยบ้านชั้นที่ 1 จะมีโปรแกรมดังนี้



รูปที่ 3.20 โปรแกรมใหม่ที่ใช้ฟังก์ชัน UDF ของบ้านชั้นที่ 1 ส่วนที่ 1



รูปที่ 3.21 โปรแกรมใหม่ที่ใช้ฟังก์ชัน UDF ของบ้านชั้นที่ 1 ส่วนที่ 2

ซึ่งในชั้นที่ 1 จะมีรายชื่อของฟังก์ชันที่ใช้ในการเขียนโปรแกรมตามตารางที่ 2

ตารางที่ 2 ตารางแสดงหมายเลข Block ฟังก์ชันโปรแกรมของบ้านชั้นที่ 1

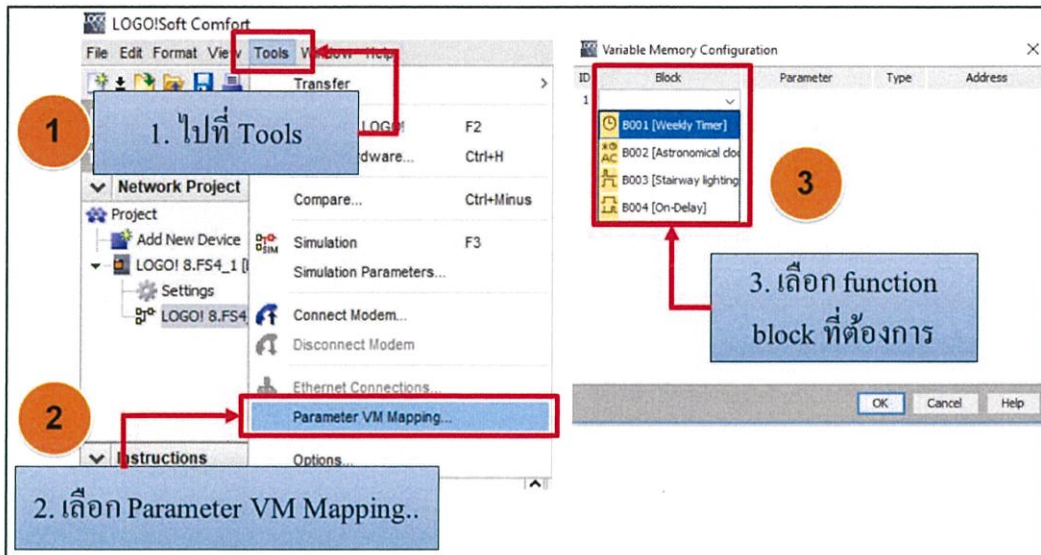
| ลำดับที่ | หมายเลข Block | ฟังก์ชัน |
|----------|---------------|-----------------------------|
| 1 | B001 | Astronomical clock |
| 2 | B003 | Weekly timer |
| 3 | B004 | Weekly timer |
| 4 | B009 | Off-Delay |
| 5 | B036 | Message texts |
| 6 | B037 | Message texts |
| 7 | B039 | Off-Delay |
| 8 | B040 | Up/Down counter |
| 9 | B041 | Analog MUX |
| 10 | B042 | Off-Delay |
| 11 | B052 | Astronomical clock |
| 12 | U001 | User-Defined Function Block |
| 13 | U002 | User-Defined Function Block |
| 14 | U003 | User-Defined Function Block |
| 15 | U004 | User-Defined Function Block |
| 16 | U005 | User-Defined Function Block |

ซึ่งในชั้นที่ 2 จะมีรายชื่อของฟังก์ชันที่ใช้ในการเขียนโปรแกรมตามตารางที่ 3

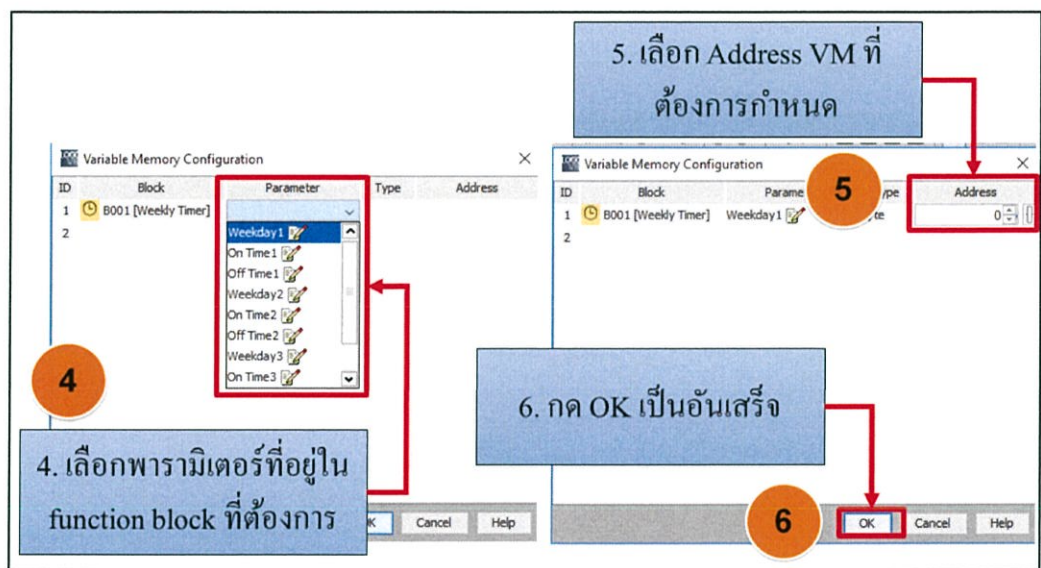
ตารางที่ 3 ตารางแสดงหมายเลข Block ฟังก์ชันโปรแกรมของบ้านชั้นที่ 2

| ลำดับที่ | หมายเลข Block | ฟังก์ชัน |
|----------|---------------|--------------------|
| 1 | B002 | Off-Delay |
| 2 | B003 | Weekly Timer |
| 3 | B005 | Off-Delay |
| 4 | B015 | Off-Delay |
| 5 | B016 | Off-Delay |
| 6 | B017 | Off-Delay |
| 7 | B018 | Pulse Relay |
| 8 | B020 | Astronomical clock |

หลังจากเขียนโปรแกรมเรียบร้อยแล้ว ทำการจับคู่พารามิเตอร์ที่อยู่ในฟังก์ชันที่เราต้องการให้ไปเก็บค่าไว้ใน VM (Variable Memory) เนื่องจากค่าพารามิเตอร์ที่อยู่ในฟังก์ชันนั้นยังไม่มี Address ของหน่วยความจำที่ไว้ใช้อ้างอิงในการดึงค่าออกไปใช้งาน เมื่อทำการจับคู่ของพารามิเตอร์กับ VM เราจะได้ Address ของพารามิเตอร์ที่เราต้องการ เพื่อที่จะดึงค่าพารามิเตอร์ในต้องการไปแสดงผลและเปลี่ยนแปลงค่าที่หน้าจอ HMI ซึ่งสามารถทำได้สะดวกโดยตามขั้นตอนต่อไปนี้



รูปที่ 3.23 ขั้นตอนการจับคู่พารามิเตอร์กับ VM



รูปที่ 3.24 ขั้นตอนการจับคู่พารามิเตอร์กับ VM (ต่อ)

ขั้นตอนการจับคู่พารามิเตอร์ที่อยู่ในฟังก์ชันให้ไปเก็บไว้ใน VM

1. ไปที่ tool
2. เลือก parameter VM Mapping
3. เลือกฟังก์ชันที่เราต้องการ
4. เลือกพารามิเตอร์ที่อยู่ในฟังก์ชันที่ต้องการ
5. กำหนด Address VM ที่เราต้องการจับคู่

6. หลังจากจับคู่เสร็จแล้วให้ทำการกด OK

ซึ่งในโปรแกรมของบ้านชั้นที่ 1 จะมีตาราง Parameter VM Mapping ดังนี้

ตารางที่ 4 Parameter VM Mapping ของบ้านชั้นที่ 1

| ลำดับ | Block | พารามิเตอร์ | ชนิดหน่วยความจำ | Address |
|-------|-------|----------------|-----------------|---------|
| 1 | B001 | Longitude | DWord | 0 |
| 2 | B001 | Latitude | DWord | 4 |
| 3 | B001 | Time Zone | Word | 8 |
| 4 | B003 | Weekday1 | Byte | 10 |
| 5 | B003 | On Time1 | Word | 11 |
| 6 | B003 | Off Time1 | Word | 13 |
| 7 | B004 | Weekday1 | Byte | 15 |
| 8 | B004 | Weekday2 | Byte | 15 |
| 9 | B004 | On Time1 | Word | 16 |
| 10 | B004 | On Time2 | Word | 18 |
| 11 | B009 | Off-Delay | Word | 24 |
| 12 | B040 | Counter | DWord | 30 |
| 13 | B001 | Sunrise offset | Word | 34 |
| 14 | B001 | Sunset offset | Word | 36 |
| 15 | B039 | Off-Delay | Word | 38 |
| 16 | B042 | Off-Delay | Word | 40 |

ซึ่งในโปรแกรมของบ้านชั้นที่ 2 จะมีตาราง Parameter VM Mapping ดังนี้

ตารางที่ 5 Parameter VM Mapping ของบ้านชั้นที่ 2

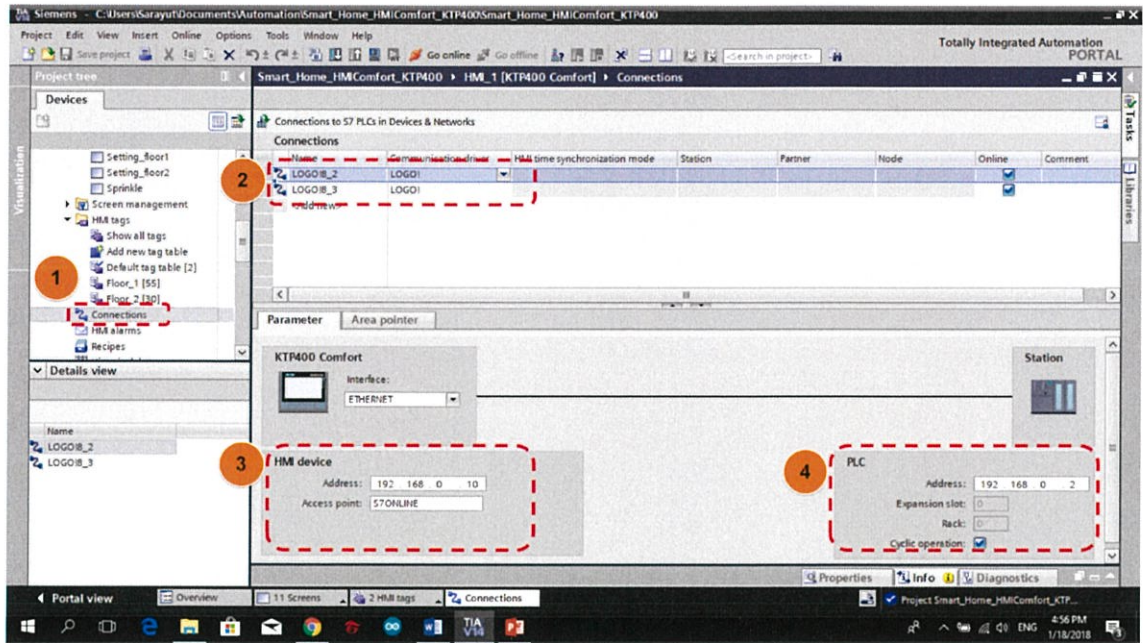
| ลำดับ | Block | พารามิเตอร์ | ชนิดหน่วยความจำ | Address |
|-------|-------|-------------|-----------------|---------|
| 1 | B015 | Off-Delay | Word | 0 |
| 2 | B016 | Off-Delay | Word | 2 |
| 3 | B017 | Off-Delay | Word | 4 |
| 4 | B005 | Off-Delay | Word | 6 |
| 5 | B002 | Off-Delay | Word | 6 |
| 6 | B003 | Weekday1 | Byte | 8 |
| 7 | B003 | On Time1 | Word | 9 |
| 8 | B003 | Off Time1 | Word | 11 |

3.4 การออกแบบจอแสดงผล HMI

ในหัวข้อนี้จะกล่าวถึงลำดับการสร้างจอแสดงผล HMI โดยใช้โปรแกรม TIA ซึ่งจะใช้จอ HMI Comfort KTP400 ในการสร้างกราฟฟิค เพื่อใช้ในการแสดงข้อมูลค่าสถานะต่าง ๆ และควบคุมการทำงานของอุปกรณ์ไฟฟ้าภายในบ้านและนอกบ้าน

3.4.1 การสร้าง Connections

ในการสร้างการเชื่อมต่อระหว่าง HMI กับ LOGO!8 นั้นเราไม่สามารถสร้างโดยการใช้นำต่าง Wizard ได้ เนื่องจากซอฟต์แวร์ของ TIA ไม่สามารถเชื่อมต่อกับ LOGO!8 เองโดยอัตโนมัติ เราจึงต้องสร้างการเชื่อมต่อของ LOGO!8 เอง โดยจะมีขั้นตอนดังต่อไปนี้ตามรูปที่ 3.17

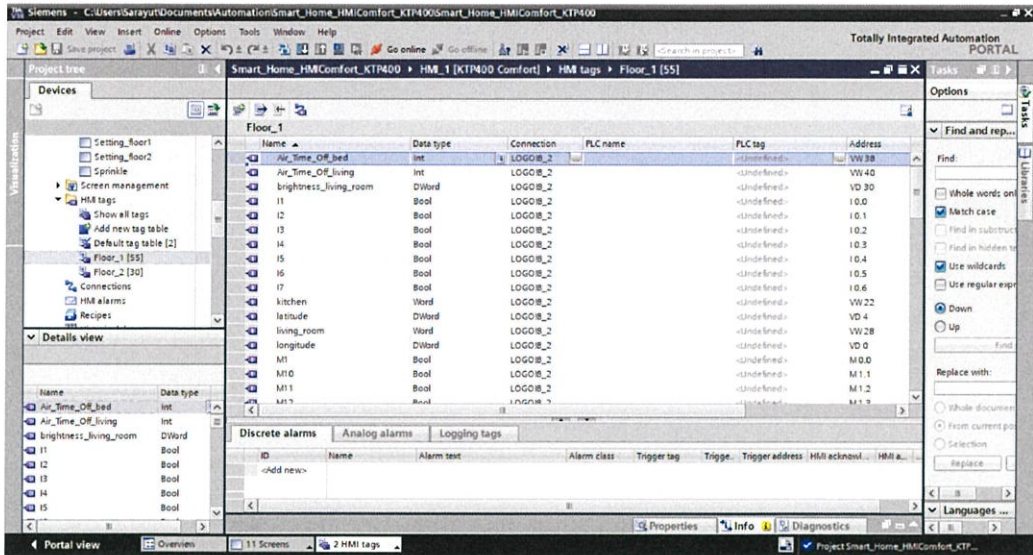


รูปที่ 3.25 การสร้างการเชื่อมต่อ HMI และ LOGO!8

1. ไปที่ Connections
2. ทำการเพิ่มการเชื่อมต่อขึ้นมาตั้งชื่อให้เรียบร้อยและเลือกการเชื่อมต่อเป็น LOGO!
3. ตั้งค่า IP Address ของจอ HMI
4. ตั้งค่า IP Address ของ LOGO!8 ให้ถูกต้อง

3.4.2 การสร้าง Tag บนอุปกรณ์ HMI

ในการสร้าง Tag ให้กับการสื่อสารระหว่าง LOGO!8 กับ HMI เราจำเป็นต้องสร้าง Tag เองทั้งหมด เนื่องจากซอฟต์แวร์ TIA จะไม่ทำการสร้าง Tag ให้อัตโนมัติเหมือนกับการเชื่อมต่อกับ PLC โดยหนึ่ง Tag เราจำเป็นต้องกำหนดชื่อ(Name) ชนิดข้อมูล(Data type) การเชื่อมต่อ(Connection) และ Address ของ LOGO!8 ดังรูปที่ 3.18 และมีข้อควรระวังในการกำหนด Address คือ Address ในอยู่ในซอฟต์แวร์ LOGO! Soft Comfort จะเริ่ม Address จาก 1 แต่ Address ในอยู่ในซอฟต์แวร์ TIA นั้นจะเริ่มจาก 0



รูปที่ 3.26 การสร้าง Tag บนอุปกรณ์ HMI

3.4.3 การสร้างกราฟฟิกแสดงผลและควบคุม

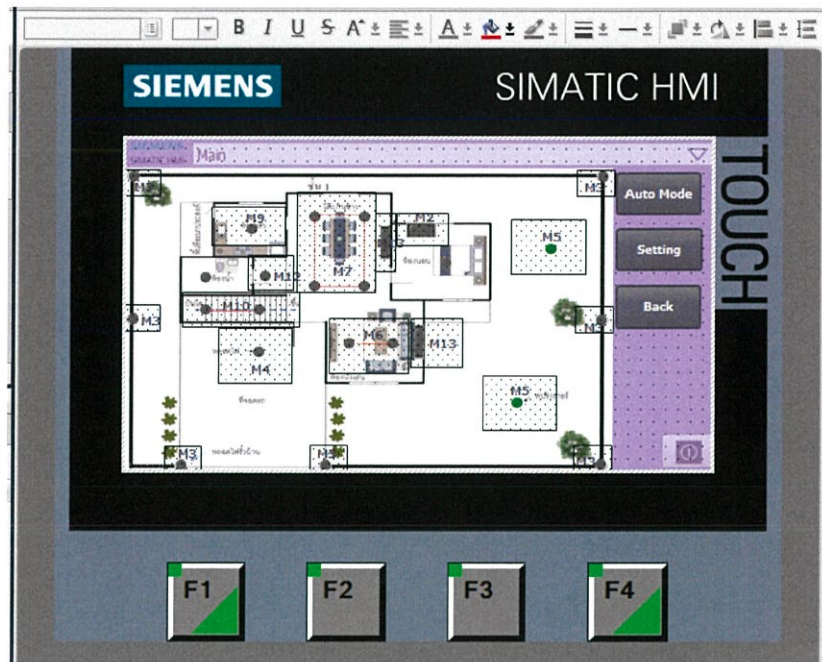
หน้าจอกราฟฟิกที่ทำการสร้างขึ้นมาจะแบ่งหน้าหลัก ๆ เป็น 2 ส่วนด้วยกันคือ หน้า Over View แสดงสถานะของอุปกรณ์ไฟฟ้าในแต่ละชั้นของบ้าน และ หน้าในส่วนของการปรับเปลี่ยนค่าพารามิเตอร์ต่าง ๆ ของ LOGO!8 ในแต่ละชั้นของบ้าน โดยจะเริ่มจากหน้าจอหลักตามรูปที่ 3.19



รูปที่ 3.27 หน้าจอหลัก

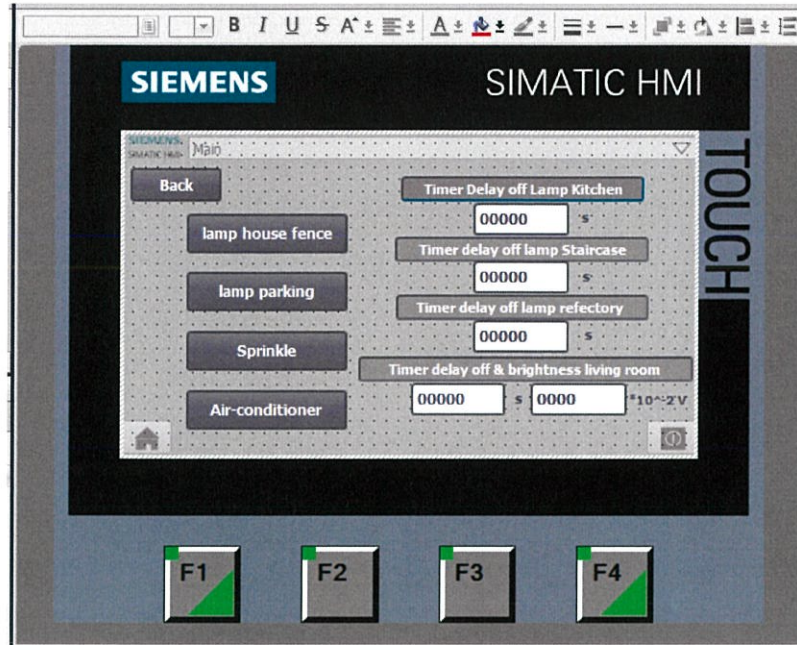
หน้าจอหลักนี้จะมีปุ่มกดทั้งหมด 3 ปุ่ม ซึ่งจะมีปุ่มปรับเปลี่ยนโหมดการทำงาน โดยโหมดอัตโนมัติ จะทำงานตามค่าพารามิเตอร์ที่ผู้ใช้งานตั้งค่าไว้ และอีกสองปุ่มจะเป็นปุ่มกดไปหน้าต่างแสดงสถานะของ

อุปกรณ์ไฟฟ้าภายในบ้านและนอกบ้านของแต่ละชั้น ต่อมาจะเป็นหน้าต่างแสดงสถานะของอุปกรณ์ไฟฟ้าของชั้นที่ 1 ได้ออกแบบไว้ตามรูปที่ 3.28



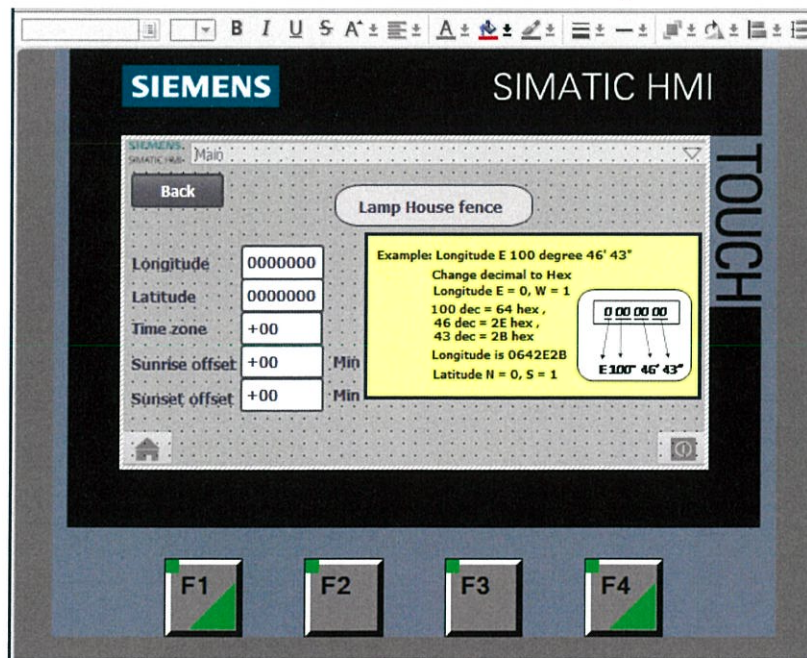
รูปที่ 3.28 การออกแบบหน้าต่างของชั้นที่ 1

จากรูปจะใช้แปลนบ้านมาใช้ในการแสดงตำแหน่งของอุปกรณ์ไฟฟ้า เพื่อให้ง่ายต่อการแสดงสถานะของอุปกรณ์และสั่งการทำงาน โดยช่องสี่เหลี่ยมที่มีจุดอยู่ข้างในจะเป็นปุ่มกดในการเปิดและปิดการทำงานของอุปกรณ์ไฟฟ้าในโหมดแมนนวล ซึ่งปุ่มเหล่านี้จะเป็นปุ่มที่ไม่แสดงให้เห็นเมื่อนำไปใช้งานจริง แต่เราสามารถกดสั่งการทำงานได้ในบริเวณของปุ่มสี่เหลี่ยมนี้ โดยมีหลอดไฟ แสดงสถานะการทำงานของอุปกรณ์ไฟฟ้าเมื่อมีการกดปุ่มหรือเข้าเงื่อนไขของโปรแกรมที่เราได้เขียนเอาไว้ และมีปุ่มตั้งค่าต่าง ๆ ของชั้นที่ 1 อยู่ด้านข้างซ้ายมือ เมื่อเราทำการกดปุ่มตั้งค่าก็จะมีหน้าต่างตั้งค่าขึ้นมาแสดงค่าพารามิเตอร์ต่าง ๆ ของ LOGO!8 ดังรูปที่ 3.29



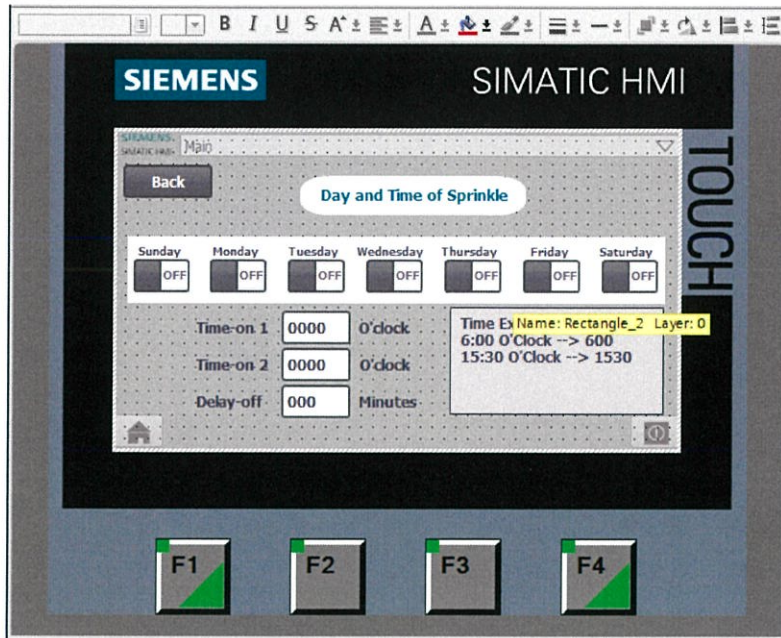
รูปที่ 3.29 หน้าต่างตั้งค่าของชั้นที่ 1

โดยหน้าต่างตั้งค่านี้จะมีการตั้งค่าระยะเวลาในการเปิดปิดของอุปกรณ์แต่ละส่วนนั้นแยกกัน โดยจะมีหน้าต่างการตั้งค่าแยกย่อยออกมาอีก 4 หน้าต่าง ซึ่งจะมีหน้าต่างตามรูปต่อไปนี้



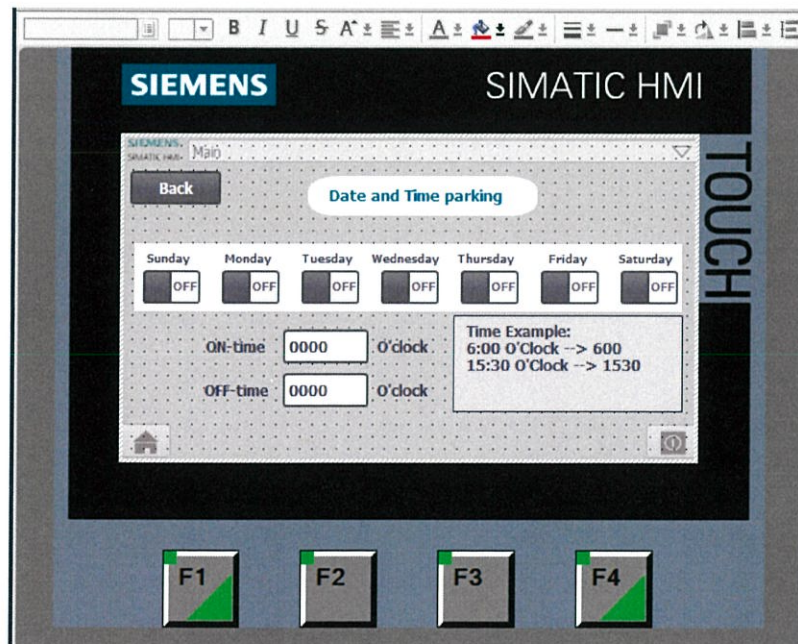
รูปที่ 3.30 หน้าต่างการตั้งค่าไฟรอบรั้วบ้าน

หน้าต่างการตั้งค่าไฟรอบรั้วบ้านนั้นเราออกแบบเพื่อให้ผู้ใช้งานที่อยู่ต่างประเทศสามารถปรับเปลี่ยนค่าละติจูดลองจิจูดให้เป็นที่อยู่ตามประเทศของตัวเอง เนื่องจากฟังก์ชันนี้ของ LOGO!8 จะเป็นการกำหนดเวลากลางวันกลางคืนในการเปิดหรือปิดไฟรอบรั้วบ้าน

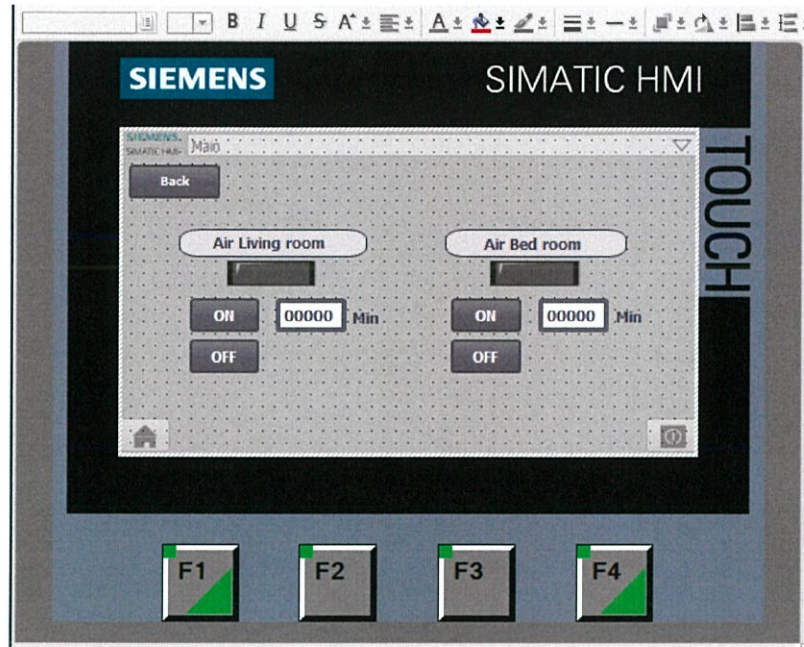


รูปที่ 3.31 หน้าต่างการตั้งค่าวันและเวลาสปริงเกิล

หน้าต่างการตั้งค่าวันและเวลาสปริงเกิลเราจะออกแบบให้สามารถกำหนดวันที่ต้องการกำหนดเวลาในการรดน้ำ และระยะเวลาในการรดน้ำได้ตามที่ผู้ใช้งานต้องการ

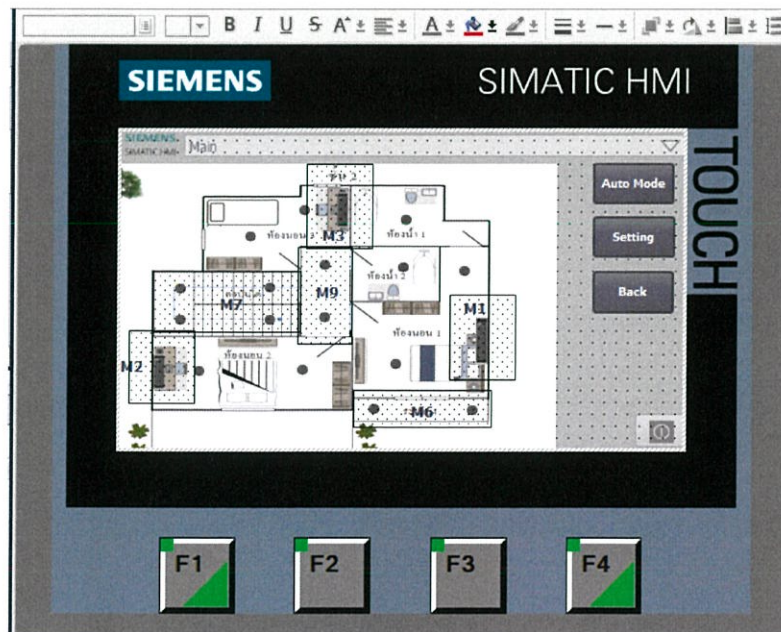


รูปที่ 3.32 หน้าต่างการตั้งค่าที่จอดรถ

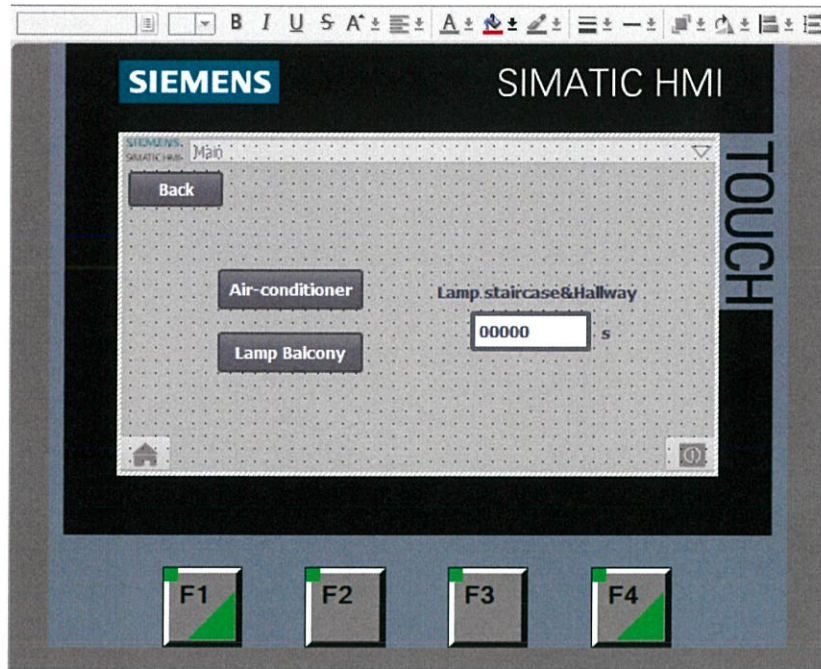


รูปที่ 3.33 หน้าต่างการตั้งค่าการเปิดปิดของเครื่องปรับอากาศ

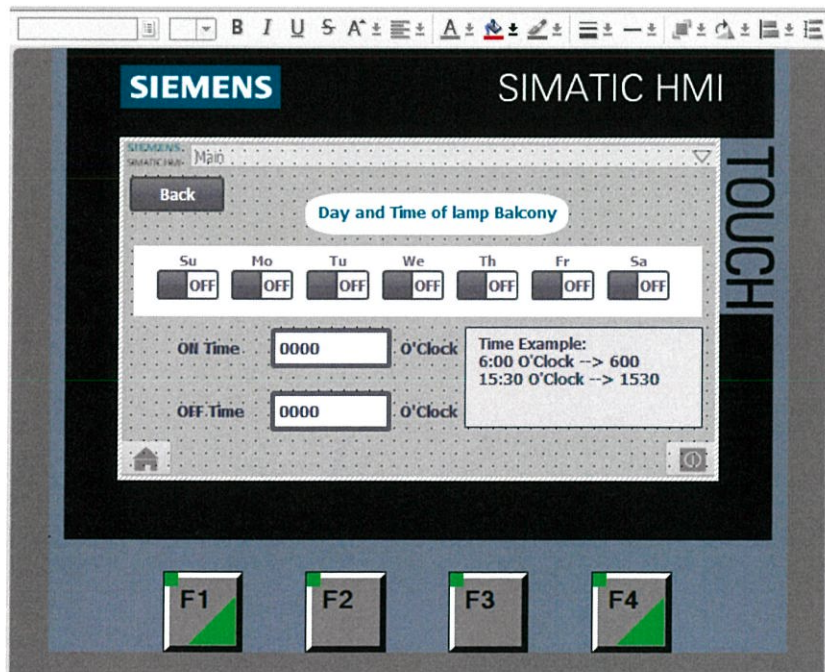
ในส่วนหน้าต่างแสดงสถานะของอุปกรณ์ไฟฟ้าของบ้านชั้นที่ 2 จะออกแบบคล้ายกับหน้าต่างแสดงสถานะของบ้านชั้นที่ 1 ดังรูปที่ 3.34 และมีหน้าต่างการตั้งค่าพารามิเตอร์ของ LOGO!8 ดังรูปที่ 3.27 ในหน้าต่างตั้งค่าก็จะมีหน้าต่างตั้งค่าโถงทางเดิน บันได เครื่องปรับอากาศ และไฟระเบียงดังรูปต่อไปนี



รูปที่ 3.34 หน้าต่างแสดงสถานะอุปกรณ์ไฟฟ้าของบ้านชั้นที่ 1

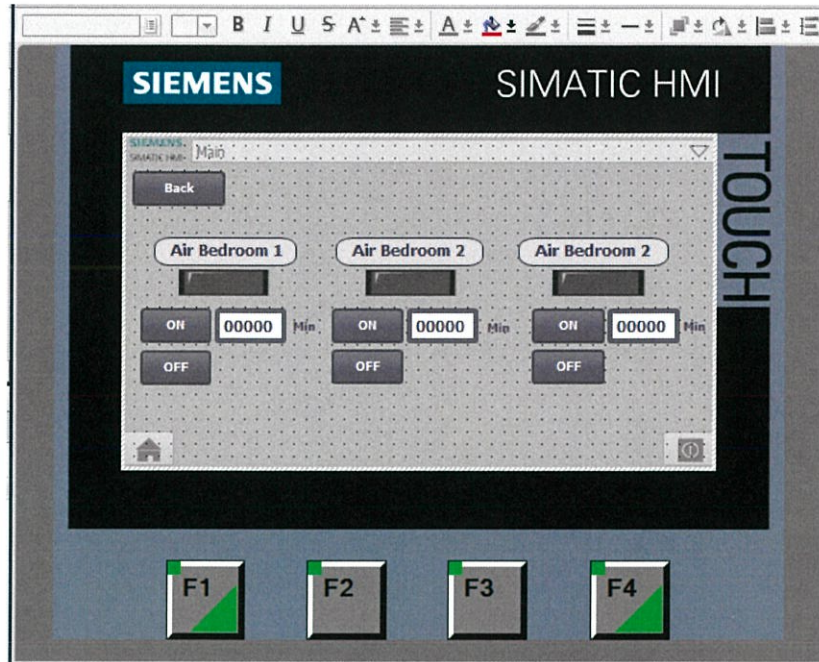


รูปที่ 3.35 หน้าต่างการตั้งค่าของบ้านชั้นที่ 2



รูปที่ 3.36 หน้าต่างการตั้งค่าการเปิดปิดไฟระเบียงบ้าน

หน้าต่าการตั้งค่าการเปิดปิดไฟระเบียงบ้าน เราจะสามารถกำหนดวันที่ต้องการเปิด และสามารถกำหนดเวลาในการเปิดและปิดได้อีกด้วย



รูปที่ 3.37 หน้าต่างการตั้งค่าเครื่องปรับอากาศของบ้านชั้นที่ 2

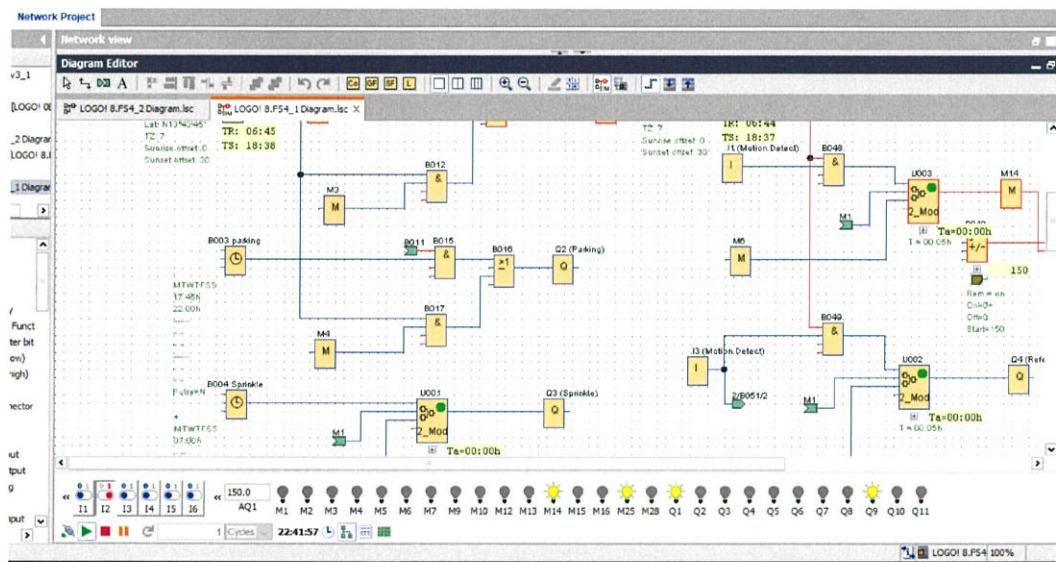
บทที่ 4

ผลการดำเนินงาน

4. ผลการทดสอบโปรแกรมและจอ HMI

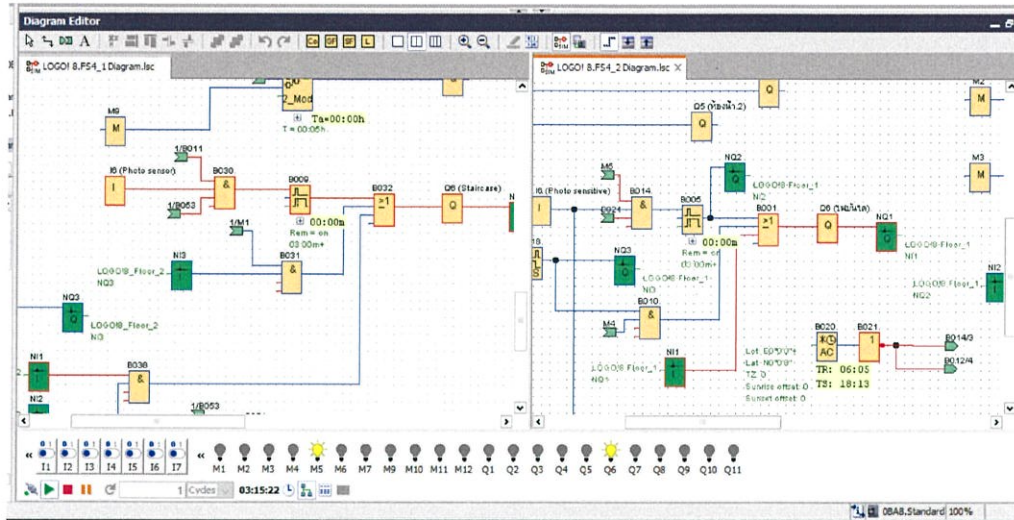
4.1 การทดสอบ LOGO!8

เราจะทดสอบการทำงานของเอาต์พุตของ LOGO!8 ตามที่เราออกแบบโปรแกรมมา โดยการจำลองสัญญาณอินพุตและตรวจสอบสัญญาณเอาต์พุตจาก Simulator ของโปรแกรม LOGO! Soft Comfort ให้เป็นไปตามที่เรากำหนด ดังรูปที่ 4.1



รูปที่ 4.1 การ Simulator ของ LOGO! Soft Comfort

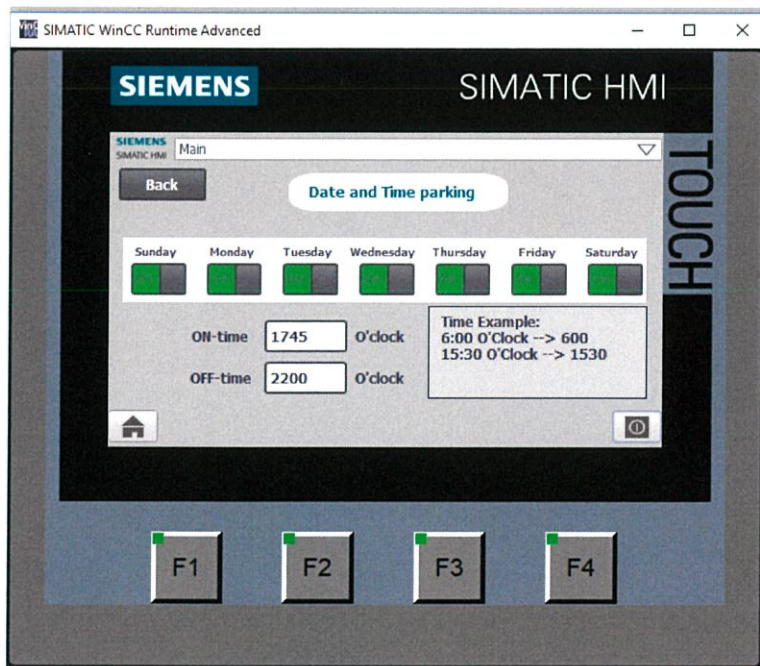
จากรูปข้างต้นเราจะทดสอบการทำงานของโปรแกรม 2 ช่วงเวลาในการ Simulator คือ ช่วงเวลากลางวันและช่วงเวลากลางคืน ซึ่งช่วงเวลาทั้งสองนี้จะมีผลต่อเอาต์พุตของโปรแกรมที่เราได้ออกแบบไว้ เมื่อทำการทดลองให้สัญญาณอินพุตกับโปรแกรมจนครบทุกอินพุตและปรับเปลี่ยนเวลาเป็นช่วงเวลาทั้งสอง จะพบว่าเอาต์พุตของโปรแกรมนั้นสามารถแสดงผลเป็นไปตามที่เราออกแบบไว้ ซึ่งเราจะทำการทดสอบแบบนี้กับ LOGO!8 อีกตัวหนึ่งเช่นกัน และทำการทดสอบทั้งสองตัวพร้อมกันดังรูปที่ 4.2



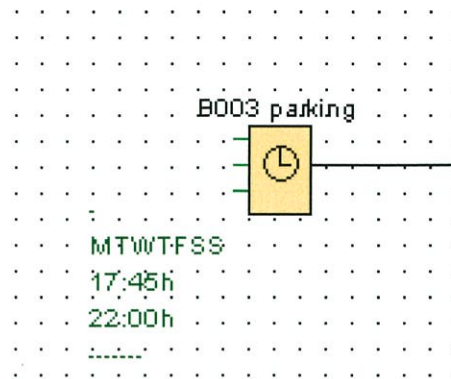
รูปที่ 4.2 การ Simulator 2 ตัวพร้อมกัน

4.2 การทดสอบหน้าจอ HMI

เราจะทำการทดสอบโดยใช้ Simulator ของโปรแกรม TIA โดยเราทำการเชื่อมต่อระหว่าง LOGO!8 กับคอมพิวเตอร์เข้าด้วยกัน แล้วทำการ Simulator หน้าจอ HMI ขึ้นมาเพื่อดูว่าหน้าจอ HMI ที่เราได้โปรแกรมไว้นั้น สามารถสื่อสารและสั่งการกับ LOGO!8 ได้หรือไม่ ซึ่งจะพบว่าในการ Simulator นั้น สามารถสื่อสารกันได้ระหว่าง HMI กับ LOGO!5 ดังรูปที่ 4.3 และ รูปที่ 4.4

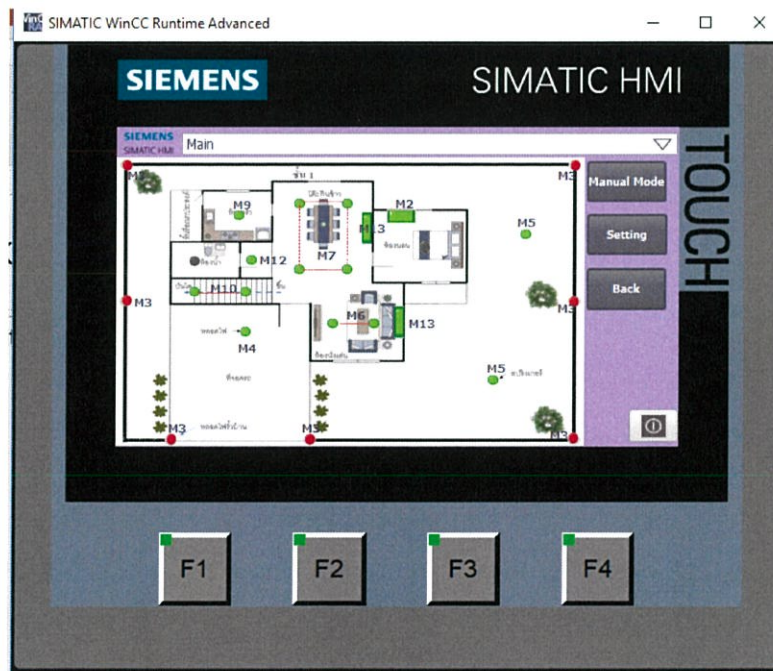


รูปที่ 4.3 หน้าต่างการตั้งค่าที่จอรถ



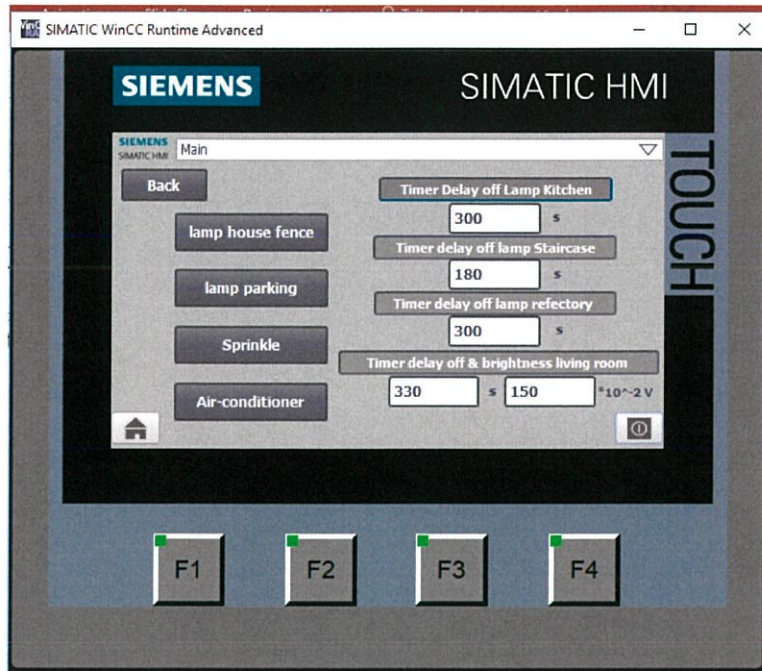
รูปที่ 4.4 พารามิเตอร์ในโปรแกรม LOGO!8

จากสองรูปข้างต้นจะพบว่า ค่าพารามิเตอร์ของ LOGO!8 นั้นมีค่าตรงกับในส่วนของ หน้าต่างการตั้งค่าพารามิเตอร์ของ HMI นั้นแสดงว่าระหว่าง LOGO!8 กับ HMI สามารถสื่อสารกันได้ ตามที่เราต้องการ ซึ่งจะมีหน้าต่าง HMI ที่ทดสอบทั้งหมดดังรูปต่อไปนี้

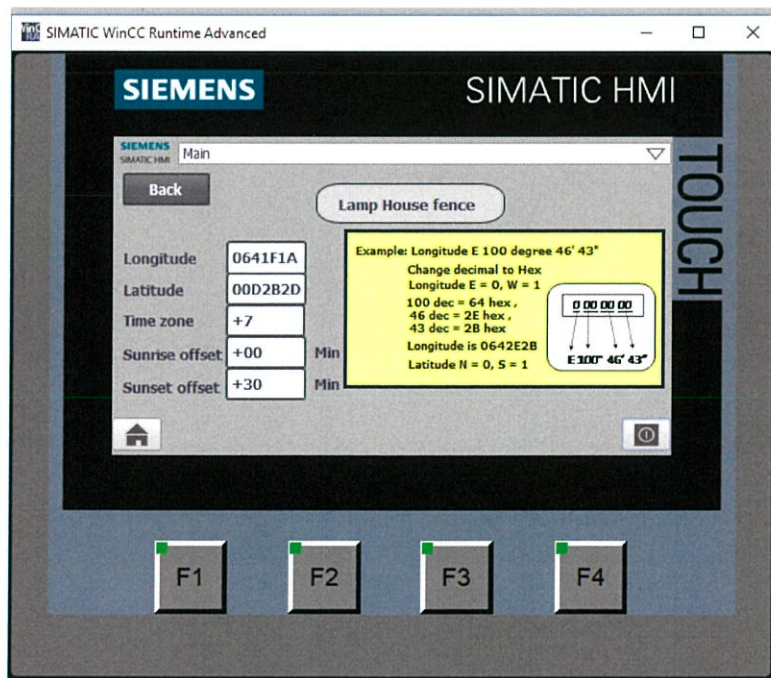


รูปที่ 4.5 หน้าต่างแสดงสถานะของอุปกรณ์ไฟฟ้าของชั้นที่ 1

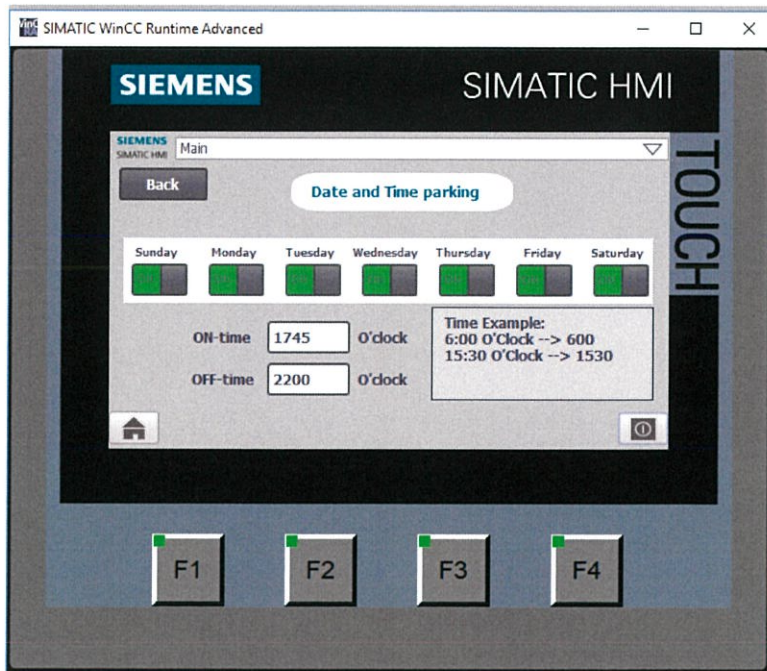
ในรูปที่ 4.5 จะเห็นว่าขณะนี้อยู่ในโหมดแมนนวล ซึ่งเราสามารถเปิดหรือปิดอุปกรณ์ไฟฟ้าได้โดยกดบริเวณตำแหน่งที่อุปกรณ์นั้น ๆ หากอุปกรณ์ไฟฟ้านั้นเป็นสีเขียวแสดงว่าอุปกรณ์นั้นกำลังทำงานอยู่ หากอยู่ในโหมดอัตโนมัติอุปกรณ์ไฟฟ้าจะทำงานตามค่าที่เรากำหนดเอาไว้



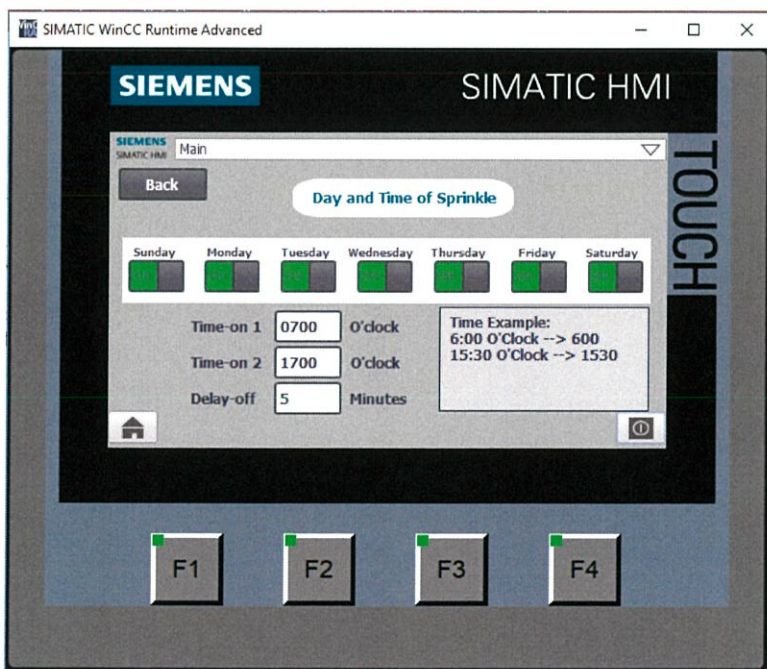
รูปที่ 4.6 หน้าต่างการตั้งค่าต่าง ๆ ของบ้านชั้นที่ 1



รูปที่ 4.7 หน้าต่างการตั้งค่าไฟรอบรั้วบ้าน



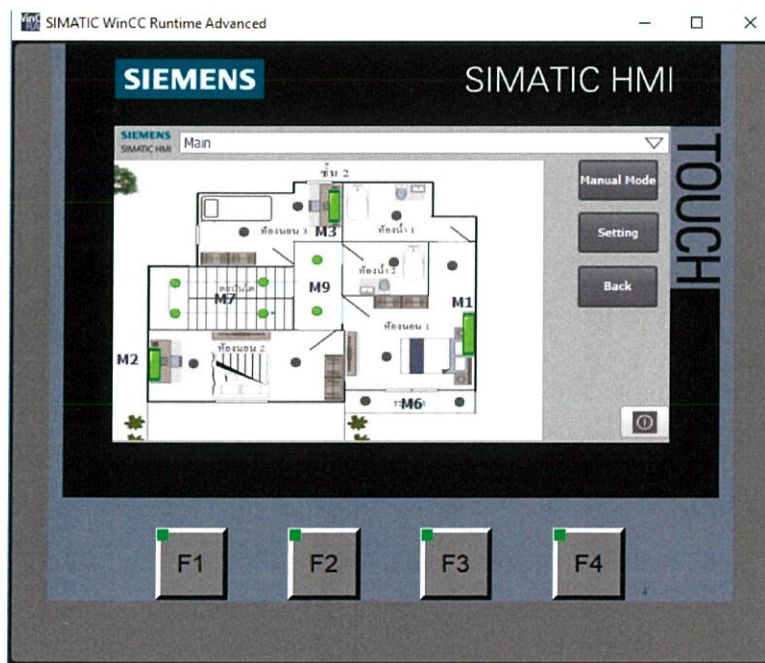
รูปที่ 4.8 หน้าต่างการตั้งค่าไฟที่จอดรถ



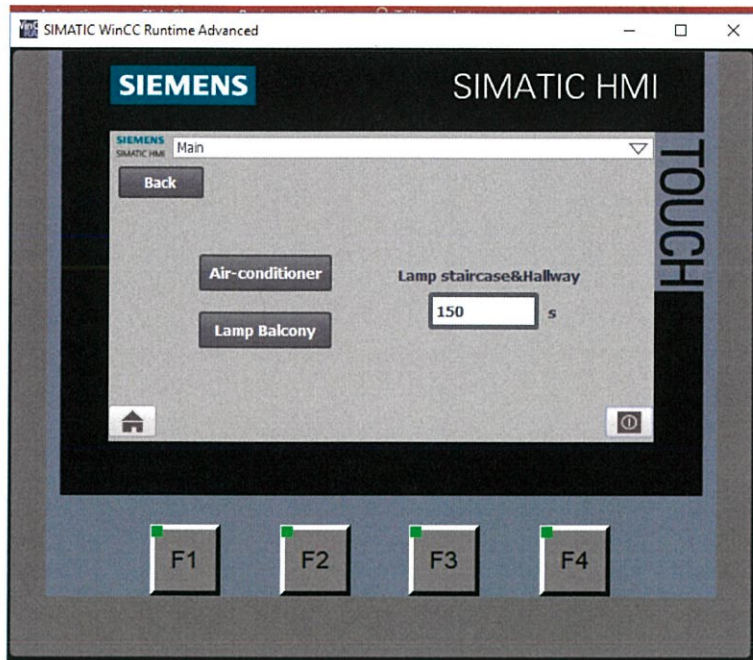
รูปที่ 4.9 หน้าต่างการตั้งค่าสปริงเกล



รูปที่ 4.10 หน้าต่างการตั้งค่าเครื่องปรับอากาศของบ้านชั้นที่ 1



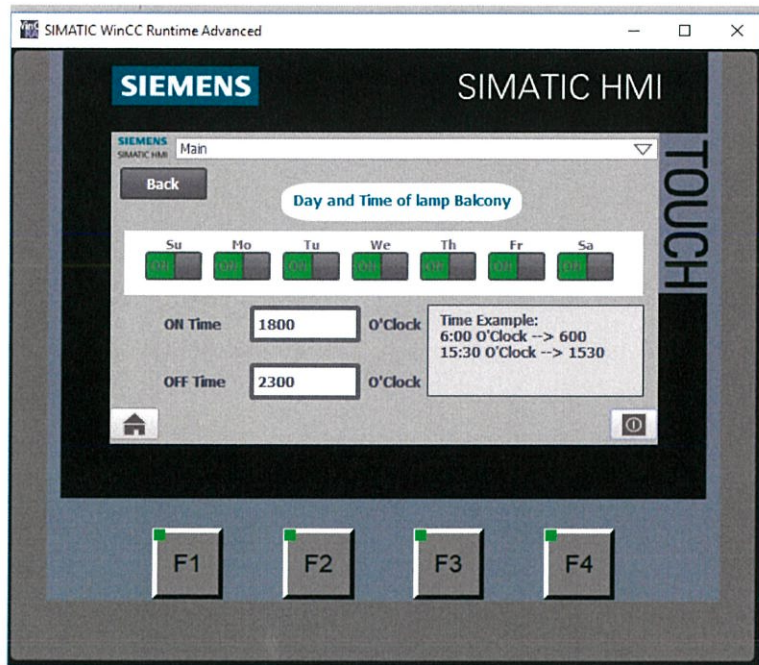
รูปที่ 4.11 หน้าต่างแสดงสถานะอุปกรณ์ไฟฟ้าของบ้านชั้นที่ 2



รูปที่ 4.12 หน้าต่างการตั้งค่าต่าง ๆ ของบ้านชั้นที่ 2



รูปที่ 4.13 หน้าต่างการตั้งค่าเครื่องปรับอากาศของบ้านชั้นที่ 2



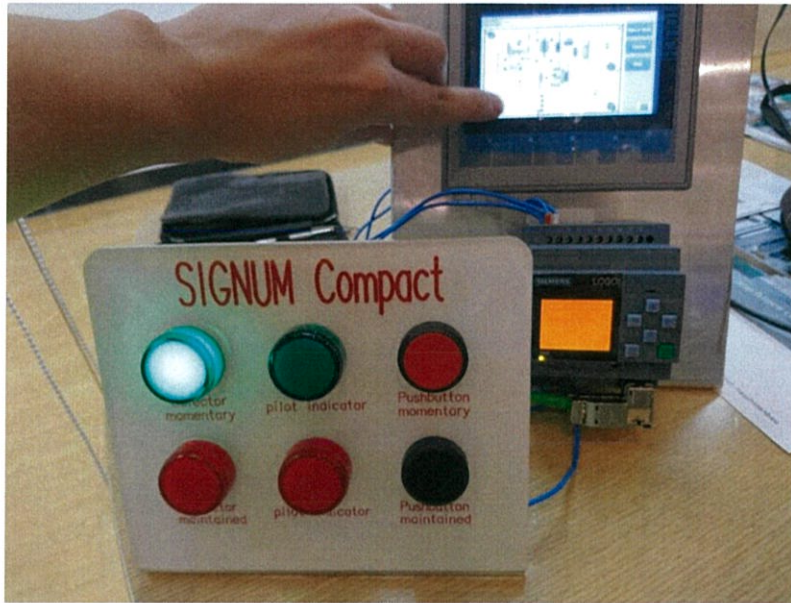
รูปที่ 4.14 หน้าต่างการตั้งค่าไฟระเบียงบ้าน

4.3 การทดสอบการเชื่อมต่อระหว่าง LOGO!8 และ HMI

ในผลการทดสอบนี้เราจะทำการเชื่อมต่อระหว่าง LOGO!8 ทั้ง 2 ตัวและ HMI เข้าด้วยกัน ดังรูปที่ 4.15 และเราจะจำลองการทำงานเซนเซอร์ซึ่งเป็นอินพุตของ LOGO!8 เป็นสวิตช์แทน และเอาท์พุตเป็นหลอดไฟ ซึ่งจะไวรั้งเข้ากับ LOGO!8 ดังรูปที่ 4.16

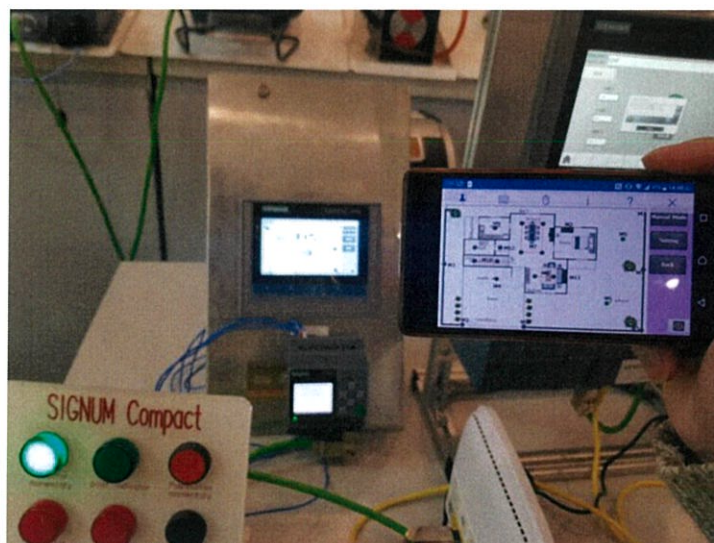


รูปที่ 4.15 การเชื่อมต่อระหว่าง LOGO!8 ทั้ง 2 ตัวและ HMI



รูปที่ 4.16 การจำลองการทำงานของอินพุตและเอาต์พุต

จากรูปที่ 4.16 จะเห็นว่าเราสั่งการทำงานที่หน้าจอ HMI แล้วเอาต์พุตของ LOGO!8 สามารถทำงานได้จริงตามที่เราเขียนโปรแกรมไว้ และในโหมดอัตโนมัติเมื่อเราทำการกดสวิทช์ เอาต์พุตของ LOGO!8 จะทำงานตามที่เราได้โปรแกรมไว้ด้วย นอกจากนี้หากเราทำการเชื่อมต่อ HMI เข้าไปในเราเตอร์ที่สามารถปล่อยสัญญาณไวไฟ (Wi-Fi) เราสามารถใช้โทรศัพท์มือถือเข้าไปควบคุม HMI ได้ โดยการใช้ผ่านโปรแกรม VNC Viewer ซึ่งสามารถหาดาวน์โหลดได้ฟรี ดังรูปที่ 4.17



รูปที่ 4.17 การใช้โทรศัพท์มือถือควบคุม HMI

บทที่ 5

สรุปผล ปัญหาและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการศึกษาการใช้งาน SIMATIC LOGO!8 ในระบบควบคุมบ้านอัจฉริยะ โดยประยุกต์ใช้ร่วมกับหน้าจอแสดงผลบนอุปกรณ์ SIMATIC HMI ทำให้ผู้ใช้งานสามารถเห็นสถานะและสั่งการทำงานของอุปกรณ์ไฟฟ้าต่าง ๆ ภายในบ้านและนอกบ้าน และสามารถปรับเปลี่ยนแก้ไขค่าพารามิเตอร์ต่าง ๆ ของ LOGO!8 ที่นำไปใช้ควบคุมเวลาการเปิดหรือปิดอุปกรณ์ไฟฟ้าผ่านหน้าจอ HMI ซึ่งการควบคุมระบบสามารถควบคุมระบบของอุปกรณ์ไฟฟ้าได้ตามที่ออกแบบไว้ สำหรับแอปพลิเคชันตัวอย่างนี้ สามารถเป็นตัวอย่างการใช้งานร่วมกันระหว่างอุปกรณ์ 2 ตัว ระหว่าง LOGO!8 และ HMI ให้กับผู้ใช้งานที่สนใจหรือต้องการใช้งานได้ศึกษา และนอกจากนี้นักศึกษายังจัดทำคู่มือการใช้งาน (Quick Guide) เป็นสื่อการสอนลูกค้าในคอร์ส Micro Automation กับทางบริษัทซีเมนส์ (ประเทศไทย) จำกัด ซึ่งคู่มือดังกล่าวได้ช่วยแนะแนวทางการเขียนโปรแกรมใช้งาน LOGO!8 เบื้องต้น สำหรับผู้ใช้งานที่ต้องการนำมาประยุกต์ใช้ตามต้องการ

5.2 ปัญหาและแนวทางแก้ไข

1. ผลลัพธ์และอุปกรณ์ของทางบริษัทซีเมนส์เป็นสิ่งที่นักศึกษาไม่เคยเรียนรู้มาก่อน จึงต้องเรียนรู้ศึกษาหาข้อมูลเป็นเวลานานกว่าจะเกิดความคุ้นเคย
2. เนื่องจากบริษัทซีเมนส์มีพนักงานหลายเชื้อชาติต้องสื่อสารเป็นภาษาอังกฤษ จึงจำเป็นต้องฝึกทักษะทางภาษาในการฟังและพูด
3. ในการทำงานร่วมกับเพื่อนร่วมงาน จำเป็นต้องวางแผน และพูดคุยแบ่งงานให้เหมาะสมในแต่ละคน
4. เอกสารคู่มือการใช้งานไม่ละเอียดพอ จึงจำเป็นต้องค้นหาในอินเทอร์เน็ตและส่ง E-mail สอบถามกับผู้เชี่ยวชาญ

5.3 ข้อเสนอแนะ

เนื่องจากนักศึกษาได้รับมอบหมายจากทางแผนกให้ศึกษา LOGO!8 เป็นหลัก โปรแกรมที่ใช้เขียน LOGO!8 นั้นแตกต่างกับโปรแกรมที่ใช้ออกแบบหน้าจอ HMI นักศึกษาจึงจำเป็นต้องเรียนรู้ในการเขียนโปรแกรม HMI เป็นระยะเวลาพอสมควร ผู้ที่ต้องการศึกษารายงานฉบับนี้จำเป็นต้องมีพื้นฐานในการเขียนโปรแกรมหน้าจอ HMI ในระดับหนึ่ง เพื่อต่อการออกแบบหน้าจอและไม่ต้องระยยะเวลานานในการศึกษาการเขียนโปรแกรม HMI

เอกสารอ้างอิง

- [1] www.support.industry.siemens.com
- [2] https://siemens.asia/e-pass_v2/TH/main
- [3] <https://www.slideshare.net/KathiravanB1/ethernet-frames-25196930>

ภาคผนวก

ภาคผนวก ก

SIMATIC LOGO!8 Manual

SIEMENS

LOGO!Soft Comfort V8.1 **1**

User Interface **2**

Tutorial **3**

Sample applications **4**

Reference material **5**

Tips and tricks **6**

LOGO!Soft Comfort Online Help

Operating Instructions

07/2016

SIEMENS

LOGO!

Manual

Preface

| | |
|--------------------------------------|----|
| Getting started with LOGO! | 1 |
| LOGO! Installation and wiring | 2 |
| Programming LOGO! | 3 |
| LOGO! functions | 4 |
| Web server | 5 |
| UDF (User-Defined Function) | 6 |
| Data log | 7 |
| Configuring LOGO! | 8 |
| Using memory cards | 9 |
| Security | 10 |
| LOGO! software | 11 |
| Applications | 12 |
| Technical data | A |
| Determining the cycle time | B |
| LOGO! without display ("LOGO! Pure") | C |
| LOGO! menu structure | D |
| Order numbers | E |
| Abbreviations | F |

This manual applies to LOGO! 8.FS4 device series only.

07/2016
AE33039676 - AC

ภาคผนวก ข

Quick Engineering Guide



The image shows a control room with several workstations. Overlaid on the scene are various digital data visualizations, including bar charts, line graphs, and text boxes. One prominent text box displays '321.089 kWh'. The Siemens logo and slogan 'Ingenuity for life' are in the top right corner. A blue banner at the bottom left contains the text 'Basic training' and 'LOGO! Soft Comfort for LOGO!8'. At the bottom, there is a white bar with 'Unrestricted © Siemens AG 2017' on the left and 'www.siemens.com/LOGO!' on the right.

SIEMENS
Ingenuity for life

TIA Portal Operator

321.089 kWh

Basic training
LOGO! Soft Comfort for LOGO!8

Unrestricted © Siemens AG 2017 www.siemens.com/LOGO!



This image is identical to the one above, showing a control room with HMI overlays. The Siemens logo and slogan 'Ingenuity for life' are in the top right corner. A blue banner at the bottom left contains the text 'Connect to HMI' and 'LOGO! Soft Comfort for LOGO!8'. At the bottom, there is a white bar with 'Unrestricted © Siemens AG 2017' on the left and 'www.siemens.com/LOGO!' on the right.

SIEMENS
Ingenuity for life

TIA Portal Operator

321.089 kWh

Connect to HMI
LOGO! Soft Comfort for LOGO!8

Unrestricted © Siemens AG 2017 www.siemens.com/LOGO!



SIEMENS
Ingenuity for Life

Modbus TCP/IP Client
LOGO! Soft Comfort for LOGO!8

Unrestricted © Siemens AG 2017 www.siemens.com/LOGO!

The advertisement features a background image of a modern industrial control room with multiple computer workstations and large data displays. A semi-transparent blue box in the lower-left corner contains the product title and subtitle. The Siemens logo is in the top right corner. The background displays various data visualizations, including a line graph and a numerical value '321,089 kWh'.



SIEMENS
Ingenuity for Life

Modbus TCP/IP Server
LOGO! Soft Comfort for LOGO!8

Unrestricted © Siemens AG 2017 www.siemens.com/LOGO!

This advertisement is identical in layout and background to the one above, but the central text in the blue box reads 'Modbus TCP/IP Server' instead of 'Client'. The Siemens logo and footer information remain the same.



ประวัติผู้เขียน

ชื่อ-นามสกุล : นายศรายุทธ์ ทองอุ่น
วัน เดือน ปีเกิด : 31 ธันวาคม 2538
ที่อยู่ : 4 หมู่ 3 ตำบลหนองสระปลา อำเภอหนองหาน จังหวัดอุดรธานี 41320
Email : sarayut.tb38@gmail.com
โทรศัพท์ : 08 2850 8101

ประวัติการศึกษา

- ระดับมัธยมศึกษาตอนปลาย โรงเรียนอุดรพิชัยรักษ์พิทยา
- ปัจจุบัน วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประสบการณ์

- นักศึกษาฝึกงาน แผนก Digital factory/Process Industries & drives บริษัท Siemens Limited Thailand
- นักศึกษาโครงการสหกิจศึกษา แผนก Digital factory/Process Industries & drives บริษัท Siemens Limited Thailand