

การใช้งาน PLC CONTROLLER ร่วมกับระบบ IOT  
เพื่อควบคุมอุปกรณ์ที่ใช้ไฟฟ้า  
USING PLC CONTROLLER WITH IOT SYSTEMS  
TO CONTROL THE ELECTRONIC DEVICES



โดย  
นายธนกฤต ธนกุลพรรณ  
นายธรรณธร มิตรดี  
นายปรีกร พุ่มปรีดา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมโทรคมนาคม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การใช้งาน PLC CONTROLLER ร่วมกับระบบ IOT  
เพื่อควบคุมอุปกรณ์ที่ใช้ไฟฟ้า  
USING PLC CONTROLLER WITH IOT SYSTEMS  
TO CONTROL THE ELECTRONIC DEVICES

โดย

นายธนกฤต	ธนกุลพรรณ	63010395
นายธรรณธร	มิตรดี	63010446
นายปธิกร	พุ่มปรีดา	63010559

อาจารย์ที่ปรึกษา

ผศ.ดร.กฤษณ์ วรจิริระ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2566

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การใช้งาน PLC CONTROLLER ร่วมกับระบบ IOT เพื่อควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

USING PLC CONTROLLER WITH IOT SYSTEMS TO CONTROL THE ELECTRONIC  
DEVICES

ผู้จัดทำ

- |              |           |          |
|--------------|-----------|----------|
| 1. นายชนกฤต  | ชนกุลพรรณ | 63010395 |
| 2. นายธรรณธร | มิตรดี    | 63010446 |
| 3. นายปธิกร  | พุ่มปรีดา | 63010559 |



..... อาจารย์ที่ปรึกษา

(ผศ.ดร.กฤษณ์ วงจรูจีระ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

การดำเนินปริญญานิพนธ์เรื่อง “การใช้งาน PLC CONTROLLER ร่วมกับระบบ IOT เพื่อควบคุมอุปกรณ์ที่ใช้ไฟฟ้า” จะไม่สามารถสำเร็จลุล่วงไปได้ด้วยดี หากไม่ได้รับความช่วยเหลือและความอนุเคราะห์อย่างดียิ่งจากอาจารย์ที่ปรึกษาปริญญานิพนธ์ คือ ผศ.ดร.กฤษณ์ วงรุจิระ ที่กรุณาให้คำแนะนำ คำปรึกษาที่เป็นประโยชน์ต่อการศึกษาค้นคว้าวิจัยให้ปริญญานิพนธ์นี้สำเร็จสมบูรณ์ยิ่งขึ้น รวมถึงสนับสนุนสถานที่สำหรับการดำเนินปริญญานิพนธ์

ขอขอบคุณท่านอาจารย์และเจ้าหน้าที่ประจำภาควิชาวิศวกรรมโทรคมนาคม สาขาวิศวกรรมโทรคมนาคมและโครงข่าย คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ได้อบรมสั่งสอน มอบความรู้ และประสบการณ์ให้แก่ผู้จัดทำ

ขอขอบคุณบริษัท SCG และบริษัท ZIFISENSE ที่กรุณาให้คำแนะนำ คำปรึกษา และแนวทางการแก้ไขปัญหาต่างๆ รวมถึงสนับสนุนเครื่องมือ และอุปกรณ์ต่างๆ ที่ใช้ระหว่างการจัดทำปริญญานิพนธ์

นายธนกฤต	ธนกุลพรรณ
นายธรรมธร	มิตรดี
นายปธิกร	พุ่มปรีดา
	ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน PLC CONTROLLER ร่วมกับระบบ IOT เพื่อควบคุมอุปกรณ์ที่ใช้ไฟฟ้า  
 USING PLC CONTROLLER WITH IOT SYSTEMS  
 TO CONTROL THE ELECTRONIC DEVICES

โดย	นายธนกฤต	ธนกุลพรรณ	63010395
	นายธรรณธร	มิตรดี	63010446
	นายปธิกร	พุ่มปรีดา	63010559

อาจารย์ที่ปรึกษา ผศ.ดร.กฤษณ์ วงศ์รุจิระ

#### บทคัดย่อ

ปริญญานิพนธ์นี้นำเสนอวิธีการนำอุปกรณ์ IoT มาใช้งานร่วมกับ PLC Controller เพื่อทำการสร้างระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า โดยทำการใช้ Sensor ร่วมกับ RS-485 เพื่อส่งข้อมูลผ่าน AP Gateway ขึ้นสู่ระบบ Zeta Cloud จากนั้นจึงใช้ MQTT Protocol บน Router ที่ติดตั้ง Firmware OpenWrt ในการ Subscribe บิตข้อมูลและทำการแปลงบิตข้อมูลที่ได้เป็นตัวเลขฐานสิบมาบันทึกไว้บน Database ของ MySQL Server และนำข้อมูลต่างๆมาแสดงผลบน Website ที่ได้สร้างขึ้นรวมถึงนำข้อมูลต่างๆมาใช้ในการสร้างเงื่อนไขสำหรับส่งค่าข้อมูลให้กับ PLC Controller ที่เชื่อมต่อกับ Router ผ่านระบบ LAN โดยใช้ Protocol Modbus TCP/IP ในการสื่อสารระหว่าง Router กับ PLC Controller ในการควบคุมอุปกรณ์ Output ที่เชื่อมต่อกับ PLC Controller ในปริญญานิพนธ์นี้จะทำการศึกษาเกี่ยวกับการสื่อสารและการทำงานร่วมกันระหว่างอุปกรณ์ต่างๆเพื่อนำมาประยุกต์ใช้งานในการออกแบบระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ABSTRACT

This thesis presents a method for using IoT devices together with a PLC Controller to create an electrical equipment control system. Using sensors together with RS-485 to send data through the AP Gateway to the Zeta Cloud system, then using the MQTT Protocol on the router OpenWrt to Subscribe bits of data and convert the bits of data into decimal numbers and save them on the Database of MySQL Server and bring the various data to display on the website. Including sending data values to the PLC Controller connected to the Router via LAN using the Modbus TCP Protocol to communicate between the Router and the PLC Controller to control the Output devices connected to the PLC Controller. In this thesis, we will study the Communication and collaboration between various devices for application in designing electrical equipment control system.

## สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	XVI
<b>บทที่ 1</b>	
<b>บทนำ</b>	<b>1</b>
2.1 ความเป็นมาและความสำคัญของปัญหา	1
2.2 วัตถุประสงค์	1
2.3 ขอบเขตของปริญญาานิพนธ์	2
<b>บทที่ 2</b>	
<b>ทฤษฎีและหลักการที่เกี่ยวข้อง</b>	<b>3</b>
2.1 IoT (Internet of Things)	3
2.2 Zeta Network	5
2.3 JSON (JavaScript Object Notation)	6
2.4 JavaScript	7
2.5 MQTT (Message Queue Telemetry Transport)	8
2.6 Database and MySQL	11
2.7 HTML (Hypertext Markup Language)	12
2.8 CSS (Cascading Style Sheets)	13
2.9 PHP (PHP Hypertext Preprocessor)	14
2.10 Modbus Protocol	15
2.11 RS-485 (Recommended Standard No.485)	21
2.12 DNS (Domain Name Server)	24
2.13 PLC (Programmable Logic Controller)	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.14 ZETA 485 Transceiver	28
2.15 AP Gateway Zifisense	30
2.16 Temperature and Humidity Sensor Zifisense	32
2.17 Mote Zifisense	33
2.18 RS-485 Soil Temperature and Humidity	34
2.19 Light Temperature and Humidity Sensor	36
2.20 Raspberry Pi	37
2.21 PL2303TA USB To TTL RS-232	38
2.22 EEPROM Flash BIOS รุ่น CH431A	39
2.23 Xiaomi Wi-Fi Range Extender AC1200	39
<b>บทที่ 3 การออกแบบและการจัดทำปฏิญญานิพนธ์</b>	<b>41</b>
3.1 การออกแบบ	41
3.2 เครื่องมือที่ใช้ในการทดลอง	94
3.3 การจัดเก็บผลการทดลอง	103
<b>บทที่ 4 ผลการทดลอง</b>	<b>108</b>
4.1 การทดสอบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud	108
4.2 การทดสอบการรับข้อมูลของ Sensor ด้วย Raspberry Pi	110
4.3 การทดสอบการส่งข้อมูลจาก Raspberry Pi ไปยัง PLC Controller	111
4.4 การทดสอบการส่งข้อมูลจาก Raspberry Pi ไปยังฐานข้อมูล	115
4.5 การทดสอบการควบคุมอุปกรณ์ Output ด้วย PLC Controller	116
4.6 การทดสอบการทำงานเบื้องต้นของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า	117
4.7 การทดสอบการใช้งาน RS485 ร่วมกับ Sensor มากกว่า 1 ตัว	121
4.8 การทดสอบการรัน Python Script บน Raspberry Pi ที่ทำการลง Firmware OpenWrt	122

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
4.9 การทดสอบการรับข้อมูลของ Sensor ด้วย OpenWrt Router	122
4.10 การทดสอบการส่งข้อมูลจาก OpenWrt Router ไปยัง PLC Controller	124
4.11 การทดสอบการส่งข้อมูลจาก OpenWrt Router ไปยังฐานข้อมูล	126
4.12 การทดสอบการแสดงผลข้อมูลบนเว็บไซต์	127
4.13 การทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า	139
4.14 การทดสอบประสิทธิภาพความแม่นยำในการวัดข้อมูลของ Sensor	151
4.15 การทดสอบประสิทธิภาพการส่งข้อมูลของ AP Gateway	152
4.16 การทดสอบประสิทธิภาพการส่งข้อมูลจาก Sensor	153
<b>บทที่ 5</b>	<b>156</b>
<b>สรุปผล</b>	<b>156</b>
5.1 สรุปผล	156
5.2 ปัญหาและข้อเสนอแนะ	157
<b>บรรณานุกรม</b>	<b>158</b>
<b>ภาคผนวก ก</b>	<b>162</b>
โปรแกรมการทำงานของการทำงานของการรับข้อมูลจาก Sensors โดยใช้ MQTT Protocol และการสื่อสารระหว่าง OpenWrt กับ PLC Controller รวมถึงการเก็บค่าข้อมูลที่ต้องการลงในฐานข้อมูล (Projectfinal.py)	162
<b>ภาคผนวก ข</b>	<b>179</b>
โปรแกรมสำหรับหน้าต่าง Login Website (สำหรับ User และ Admin)	179
<b>ภาคผนวก ค</b>	<b>188</b>
โปรแกรมสำหรับหน้าต่าง Dashboard Website (User และ Admin)	188
<b>ภาคผนวก ง</b>	<b>210</b>
โปรแกรมสำหรับหน้าต่าง About Website (สำหรับ User และ Admin)	210
<b>ภาคผนวก จ</b>	<b>221</b>
โปรแกรมสำหรับหน้าต่าง Members Website (สำหรับ User และ Admin)	221
<b>ภาคผนวก ฉ</b>	<b>232</b>
โปรแกรมสำหรับหน้าต่าง History Website (สำหรับ Admin)	232
<b>ภาคผนวก ช</b>	<b>250</b>
โปรแกรมสำหรับหน้าต่าง Graph Website (สำหรับ Admin)	250
<b>ภาคผนวก ซ</b>	<b>267</b>
โปรแกรมสำหรับการตกแต่ง Style ของ Website	267

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่		หน้า
1.1	Block Diagram ภาพรวมของปริญญาานิพนธ์	2
2.1	ระบบ IoT	5
2.2	ระบบ Zeta Network	5
2.3	การเชื่อมต่อ MQTT	10
2.4	MySQL Database	11
2.5	การสื่อสารแบบอนุกรมด้วย RS-485 สำหรับ Modbus RTU	16
2.6	การสื่อสารแบบอีเทอร์เน็ตสำหรับ Modbus TCP	19
2.7	ส่วนประกอบชุดข้อมูลของ Modbus TCP เทียบกับ Modbus RTU	20
2.8	ตัวอย่างการเชื่อมต่อ RS-485 ระหว่างเครื่องมือวัดกับตัวแปลงสัญญาณ	21
2.9	ตัวอย่างการทำงานของ RS-485 แบบ Network	22
2.10	ขั้นตอนการทำงานของ DNS	25
2.11	โครงสร้างของ PLC	26
2.12	PLC LOGO! 230 RCE	27
2.13	Zeta 485 Transceiver	29
2.14	AP Gateway Zifisense	30
2.15	Temperature and Humidity Sensor Zifisense	32
2.16	Mote Zifisense	33
2.17	RS-485 Soil Temperature and Humidity Sensor	35
2.18	Light Temperature and Humidity Sensor	36
2.19	ส่วนประกอบของ Raspberry Pi	38
2.20	PL2303TA USB TO TTL RS-232	38
2.21	EEPROM FLASH BIOS รุ่น CH431A	39
2.22	Xiaomi Wi-Fi Range Extender AC1200	40
3.1	บล็อกไดอะแกรมแสดงภาพรวมการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า	41
3.2	ติดตั้ง AP Gateway	42

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.3 ติดตั้ง RS485 และ Temperature & Humidity Sensor	43
3.4 ติดตั้ง PLC Controller และเชื่อมต่อกับ Raspberry Pi	43
3.5 การเชื่อมต่ออุปกรณ์เข้าด้วยกันผ่าน Layer 2 Switch	45
3.6 การเชื่อมต่ออุปกรณ์ Output กับ PLC Controller	45
3.7 การออกแบบวงจรสำหรับเชื่อมต่อ Sensor	46
3.8 การออกแบบวงจรสำหรับเชื่อมต่อ Sensor	47
3.9 วงจรสำหรับเชื่อมต่อ Sensor	47
3.10 การเชื่อมต่อระหว่าง RS485 Sensor ทั้ง 2 ตัว และแหล่งจ่ายไฟ	47
3.11 การตั้งค่า Parameter ต่างๆให้กับ Sensor	49
3.12 รูปแบบข้อมูลที่รับได้จาก Sensor ทั้งสองตัว	49
3.13 ภาพด้านหน้าของตู้สำหรับปลุกต้นไม้	50
3.14 ภาพด้านหลังของตู้สำหรับปลุกต้นไม้	51
3.15 ภาพด้านล่างของตู้สำหรับปลุกต้นไม้	51
3.16 การตั้งค่าเวลาในการส่งข้อมูลของ RS485	53
3.17 การตั้งค่า Address Code และ Meter Name	53
3.18 การตั้งค่า Acquisition Parameter	54
3.19 การตั้งค่า Baud rate, Data bit, Check digit และ Stop bit	54
3.20 การตั้งค่า Register Function Code	55
3.21 แผนผังแสดงการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud	55
3.22 การตั้งค่า Raspberry Pi ให้อยู่ใน Root โหมด	57
3.23 การตั้งค่า IP ของ Raspberry Pi	57
3.24 แผนผังแสดงการรับและส่งข้อมูลของ Raspberry Pi	58
3.25 การติดตั้ง ภาษา Python 3	59
3.26 ตัวอย่างการลง Library	59
3.27 Library ทั้งหมดที่ใช้งาน	59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.28 การเชื่อมต่อคอมพิวเตอร์กับ OpenWrt Router ด้วยโปรแกรม WinSCP	60
3.29 ทำการโอนย้ายไฟล์ Python Script ไปยัง OpenWrt Router	60
3.30 ทำการ Run Python Script	61
3.31 การตรวจสอบสถานะการอนุญาตของไฟล์	61
3.32 แผนผังแสดงการรับและส่งข้อมูลของ OpenWrt Router	63
3.33 การเตรียมอุปกรณ์ EEPROM Flash BIOS	64
3.34 ขาที่ 1 คือขาที่อยู่ด้านล่างซ้ายสุดบริเวณลูกศรสีทองแดง	64
3.35 ทำการเชื่อมต่อ CH เข้ากับ SPI Flash Memory ของ Router	65
3.36 ทำการเชื่อมต่อ CH เข้ากับ SPI Flash Memory ของ Router	65
3.37 ทำการเชื่อมต่อ CH เข้ากับคอมพิวเตอร์ผ่านช่องเสียบ USB	66
3.38 Read SPI Firmware	66
3.39 Write SPI Firmware	67
3.40 เลือกโหมดในการทำงานของ Router	68
3.41 ใส่ข้อมูลต่างๆให้กับ Router	68
3.42 การตั้งค่าในโปรแกรม Tftpd64 ในส่วนของคอมพิวเตอร์	68
3.43 การอัปเดตเครื่องมือที่ใช้ในการติดตั้ง Package ต่างๆ	69
3.44 การติดตั้ง Luci	69
3.45 การตั้งค่า Routing Protocol หมายเลข IP และ Netmask ของ Router	70
3.46 Web Interface ของ OpenWrt Router	70
3.47 Package ที่จำเป็นต้องใช้งาน	71
3.48 Library ที่จำเป็นต้องใช้งาน	71
3.49 หน้าแรกของโปรแกรม LOGO! Soft Comfort V8.3	72
3.50 การเพิ่มอุปกรณ์ PLC Controller ในโปรแกรม	72
3.51 การกำหนด IP Address บนเครื่องคอมพิวเตอร์	73
3.52 การเชื่อมต่อของคอมพิวเตอร์และ PLC Controller	73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.53 โปรแกรมที่ออกแบบใน LOGO! Soft Comfort V8.3	74
3.54 โปรแกรมส่วนที่ใช้รับค่า Coil single bit 0 หรือ 1	75
3.55 โปรแกรมส่วนที่ใช้รับค่า Input Register 16-bit โดย VW2 เป็นตัวแปรสำหรับรับค่าอุณหภูมิ และ VW4 เป็นตัวแปรสำหรับรับค่าความชื้นสัมพัทธ์	75
3.56 Network input สำหรับรับค่า Coil และ Network analog input สำหรับรับค่า Input Register	76
3.57 Output 1-4 ของ PLC Controller เมื่อ logic เป็น 0 จะ open circuit เมื่อ logic เป็น 1 closed circuit	76
3.58 NOT เป็นการเปลี่ยน logic จาก 0 เป็น 1 หรือ จาก 1 เป็น 0	76
3.59 Mathematic instruction ใช้ในสำหรับการทำกระบวนการคณิตศาสตร์	76
3.60 Flag ตัวกำหนด status สีของหน้าจอ	77
3.61 Message texts หน้าจอแสดงข้อความ	77
3.62 Threshold trigger เป็นการกำหนดช่วงข้อมูลที่ได้รับให้มี logic เป็น 0 หรือ 1 สามารถกำหนดเพิ่มค่า gain และ จุดทศนิยม	77
3.63 การตั้งค่า PLC Controller ทั่วไป	78
3.64 สร้าง Modbus server connection	78
3.65 การสร้าง Data transfer information table	79
3.66 หน้าจอแสดงผลของ PLC Controller ขณะที่สถานะของ Output คือ ปิด	80
3.67 หน้าจอแสดงผลของ PLC Controller ขณะที่สถานะของ Output คือ เปิด	80
3.68 การสร้าง Data transfer information table	81
3.69 โปรแกรมส่วนที่ใช้รับค่า Coil single bit 0 หรือ 1	82
3.70 โปรแกรมส่วนที่ใช้รับค่า Input Register 16-bit ของ Soil Temperature And Humidity Sensor	83
3.71 โปรแกรมส่วนที่ใช้รับค่า Input Register 16-bit ของ Light Temperature and Humidity Sensor	83

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.72 การเชื่อมต่อเข้ากับฐานข้อมูล	84
3.73 การสร้าง Schema หรือ Database ในฐานข้อมูล	85
3.74 การสร้างตาราง หรือ Table สำหรับจัดเก็บข้อมูล	85
3.75 หน้า Dashboard	86
3.76 หน้า About Project	87
3.77 หน้า History	88
3.78 หน้า Members	89
3.79 หน้า Graphs	90
3.80 แผนผังแสดงการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าในขั้นตอนการทดลองการทำงานของระบบเบื้องต้น	92
3.81 แผนผังแสดงการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า	93
3.82 Router Zyxel	95
3.83 พัดลมระบายอากาศ	97
3.84 ป้อนน้ำ	97
3.85 ไฟ LED สำหรับปลูกต้นไม้	98
3.86 Fluke Model 51 II Handheld Portable Digital Probe Thermometer	99
3.87 โปรแกรม LOGO! Soft Comfort V8.3	100
3.88 โปรแกรม Visual Studio Code	100
3.89 โปรแกรม MySQL Workbench	101
3.90 โปรแกรม WinSCP	101
3.91 โปรแกรม Tftpd64	102
4.1 บิตข้อมูลเมื่ออุณหภูมิและความชื้นมีค่าต่ำ	109
4.2 บิตข้อมูลเมื่อทำการเพิ่มอุณหภูมิและความชื้น	109
4.3 บิตข้อมูลที่ Subscribe ได้เมื่ออุณหภูมิมียังมีค่าต่ำ	110
4.4 บิตข้อมูลที่ Subscribe ได้เมื่อทำการเพิ่มอุณหภูมิ	111

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.5 ผลการทดสอบการส่งข้อมูลประเภท Coil โดยมีค่าเท่ากับ 1	111
4.6 ค่าที่ฝั่งรับได้รับซึ่งมีค่าเท่ากับ 1	112
4.7 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบจับข้อมูล	112
4.8 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบจับข้อมูล	112
4.9 ผลการทดสอบการส่งข้อมูลประเภท Register โดยมีค่าเท่ากับ 800	113
4.10 ค่าที่ฝั่งรับได้รับซึ่งมีค่าเท่ากับ 800	113
4.11 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบจับข้อมูล	114
4.12 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบจับข้อมูล	114
4.13 ผลที่ได้จากการเก็บข้อมูลลงในฐานข้อมูล	115
4.14 Relay Switch ของ Output Q1 ที่เชื่อมต่ออยู่กับหลอดไฟอยู่ในสถานะ Open circuit	116
4.15 Relay Switch ของ Output Q1 ที่เชื่อมต่ออยู่กับหลอดไฟอยู่ในสถานะ Short circuit	116
4.16 การเชื่อมต่อ Temperature and Humidity Sensor เข้ากับ RS485	118
4.17 การเชื่อมต่อ PLC Controller เข้ากับ Raspberry Pi และหลอดไฟ	118
4.18 น้ำเย็นและน้ำร้อนสำหรับการทดสอบ	119
4.19 อุปกรณ์ทั้งหมดที่ใช้ในการทดลอง	119
4.20 ทำการทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้ากับน้ำร้อน	120
4.21 ทำการทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้ากับน้ำเย็น	120
4.22 ค่าที่วัดได้จาก Soil Temperature And Humidity Sensor Light Temperature And Humidity Sensor	121
4.23 ผลจากการรัน Python Script	122
4.24 ค่าที่รับได้ได้จาก Soil Temperature And Humidity	123
4.25 ค่าที่รับได้จาก Light Temperature And Humidity Sensor	123
4.26 การแสดงผลค่าที่ได้จาก Soil Temperature And Humidity	124

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.27 การแสดงผลค่าที่ได้จาก Light Temperature And Humidity Sensor	125
4.28 ค่าที่ OpenWrt Router ส่งมายังฐานข้อมูล	126
4.29 การดึงค่าข้อมูลต่างๆที่ได้ทำการเก็บไว้ในฐานข้อมูลมาแสดงผลบนเว็บไซต์	127
4.30 หน้า Dashboard เมื่อเข้าใช้งานแบบ Admin หลัก	128
4.31 หน้า About Project เมื่อเข้าใช้งานแบบ Admin หลัก	128
4.32 หน้า History เมื่อเข้าใช้งานแบบ Admin หลัก	129
4.33 หน้า Members เมื่อเข้าใช้งานแบบ Admin หลัก	129
4.34 หน้า Graphs เมื่อเข้าใช้งานแบบ Admin หลัก	130
4.35 หน้า Dashboard เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1	131
4.36 หน้า About Project เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1	131
4.37 หน้า History เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1	132
4.38 หน้า Members เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1	132
4.39 หน้า Graphs เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1	133
4.40 หน้า Dashboard เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2	134
4.41 หน้า About Project เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2	134
4.42 หน้า History เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2	135
4.43 หน้า Members เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2	135
4.44 หน้า Graphs เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2	136
4.45 หน้า Dashboard เมื่อเข้าใช้งานแบบ User ทั่วไป	137
4.46 หน้า About Project เมื่อเข้าใช้งานแบบ User ทั่วไป	138
4.47 หน้า Members เมื่อเข้าใช้งานแบบ User ทั่วไป	138
4.48 การติดตั้งอุปกรณ์ทั้งหมดเข้ากับตู้สำหรับปลูกต้นไม้ในที่ร่มอย่างง่าย	139
4.49 Soil Temperature and Humidity Sensor	140
4.50 Light Temperature and Humidity Sensor, ไฟ Led และ พัดลมระบายอากาศ	140

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า	
4.51	ปั้มน้ำสำหรับรดน้ำต้นไม้	141
4.52	PLC Controller, RS485 และ แผ่นวงจร PCB สำหรับเชื่อม Sensor เข้ากับ RS485	141
4.53	พัดลมระบายอากาศไม่ทำงาน	142
4.54	PLC Controller แสดงสถานะการทำงานของพัดลมว่าปิด	143
4.55	พัดลมระบายอากาศทำงาน	143
4.56	PLC Controller แสดงสถานะการทำงานของพัดลมระบายอากาศว่าเปิด	144
4.57	ทำการปิดสวิตช์	144
4.58	PLC Controller แสดงสถานะการทำงานของพัดลมระบายอากาศว่าถูกควบคุมให้เปิดแต่พัดลมระบายอากาศไม่ทำงาน	144
4.59	ไฟ LED ไม่ทำงาน	145
4.60	PLC Controller แสดงสถานะการทำงานของไฟ LED ว่าปิด	145
4.61	ไฟ LED ทำงาน	146
4.62	PLC Controller แสดงสถานะการทำงานของไฟ LED ว่าเปิด	146
4.63	ทำการปิดสวิตช์	147
4.64	PLC Controller แสดงสถานะการทำงานของไฟ LED ว่าถูกควบคุมให้เปิดแต่ไฟ LED ไม่ทำงาน	147
4.65	ปั้มน้ำไม่ทำงาน	148
4.66	PLC Controller แสดงสถานะการทำงานของปั้มน้ำว่าปิด	148
4.67	ปั้มน้ำทำงาน	149
4.68	PLC Controller แสดงสถานะการทำงานของปั้มน้ำว่าเปิด	149
4.69	อุปกรณ์ Output ทั้งหมดทำงาน	150
4.70	PLC Controller แสดงสถานะการทำงานของอุปกรณ์ Output ทั้งหมดว่าเปิด	150
4.71	การทดสอบประสิทธิภาพความแม่นยำในการวัดข้อมูลกับน้ำเย็น	151

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.72 การทดสอบประสิทธิภาพความแม่นยำในการวัดข้อมูลกับน้ำร้อน	152
4.73 เวลาที่ทำการรับข้อมูลจาก AP Gateway ครั้งที่ 1 เมื่อเวลาผ่านไป 3 นาที	152
4.74 เวลาที่ทำการรับข้อมูลจาก AP Gateway ครั้งที่ 20 เมื่อเวลาผ่านไป 60 นาที	153
4.75 ระยะทางที่ไกลที่สุดเมื่อ AP Gateway อยู่ในอาคาร และไม่ได้ทำการใช้ Mote ขยายสัญญาณ	153
4.76 ระยะทางที่ไกลที่สุดเมื่อ AP Gateway อยู่ด้านบนดาดฟ้าของอาคาร และไม่ได้ทำการใช้ Mote ขยายสัญญาณ	154
4.77 ระยะทางที่ไกลที่สุดเมื่อ AP Gateway อยู่บนดาดฟ้า และได้ทำการใช้ Mote ขยายสัญญาณบริเวณหน้าอาคารยิมเนเซียม 1	155

## สารบัญตาราง

ตารางที่	หน้า
2.1 ชุดข้อมูลสำหรับการสื่อสาร Modbus RTU	16
2.2 ตำแหน่งแอดเดรสใน Modbus RTU โดยแบ่งตามรูปแบบการทำงาน	17
2.3 รายละเอียดชุดข้อมูล Function Code	18
2.4 ชุดคำสั่งสำหรับการอ่าน (Read Command)	19
2.5 ชุดคำสั่งสำหรับการเขียน (Write Command)	19
2.6 รายละเอียดของแต่ละ Field ในหนึ่งเฟรมของ Modbus TCP	20
2.7 การเปรียบเทียบคุณสมบัติต่างๆของ RS-232, RS-423, RS-422, RS-485	23
2.8 ตารางแสดงคุณสมบัติของ PLC LOGO! 230 RCE	27
2.9 แสดงพารามิเตอร์ผลิตภัณฑ์	28
2.10 ตารางแสดงคุณสมบัติของอุปกรณ์ ZETA 485 Transceiver	28
2.11 AP Gateway Specification	31
2.12 Temperature And Humidity Sensor Specification	32
2.13 Mote Specification	34
2.14 RS-485 Soil Temperature and Humidity Sensor Specification	35
2.15 Light Temperature and Humidity Sensor Specification	36
2.16 Xiaomi Wi-Fi Range Extender AC1200 Specification	40
3.1 ตัวแปรที่ใช้ในการรับส่งข้อมูลและรายละเอียดเกี่ยวกับตัวแปร	79
3.2 ตัวแปรที่ใช้ในการรับส่งข้อมูลและรายละเอียดเกี่ยวกับตัวแปร	82

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบันโรงงานอุตสาหกรรมต่างๆมีการควบคุมระบบและอุปกรณ์ที่หลากหลายทำให้เกิดความซับซ้อนและมีความไม่ยืดหยุ่นในการควบคุมระบบทำให้มีโอกาสที่จะเกิดข้อผิดพลาดจากการควบคุมได้จึงมีการนำ PLC Controller มาใช้ซึ่ง PLC ถือเป็นตัวควบคุมที่เป็นที่นิยมในอุตสาหกรรมต่างๆ เนื่องจากเป็นอุปกรณ์ที่มีความยืดหยุ่นและเสถียรภาพสูงแถมยังเป็นตัวควบคุมที่ผู้ใช้สามารถโปรแกรมได้ จึงสามารถนำตัวควบคุม PLC นี้ไปใช้งานได้ในทุกอุตสาหกรรมไม่ว่าจะเป็น อุตสาหกรรมรถยนต์และชิ้นส่วนรถยนต์ อุตสาหกรรมอาหารและยา อุตสาหกรรมอิเล็กทรอนิกส์ สาธารณูปโภคต่างๆ รวมไปถึงภาคเกษตรกรรมอีกด้วย สามารถปรับให้เหมาะสมกับการใช้งานและขนาดระบบได้เป็นอย่างดี ตัวอย่างเช่นการใช้งานแบบเดี่ยว (Standalone) ซึ่งเหมาะสำหรับการทำงานในไลน์การผลิตที่ไม่ซับซ้อน หรือการเชื่อมต่อพีแอลซีหลายๆตัวเข้าด้วยกันเป็นรูปแบบ Network หรือเครือข่ายเพื่อควบคุมการทำงานของระบบให้มีประสิทธิภาพมากยิ่งขึ้นซึ่งเหมาะสำหรับกระบวนการที่มีความซับซ้อน

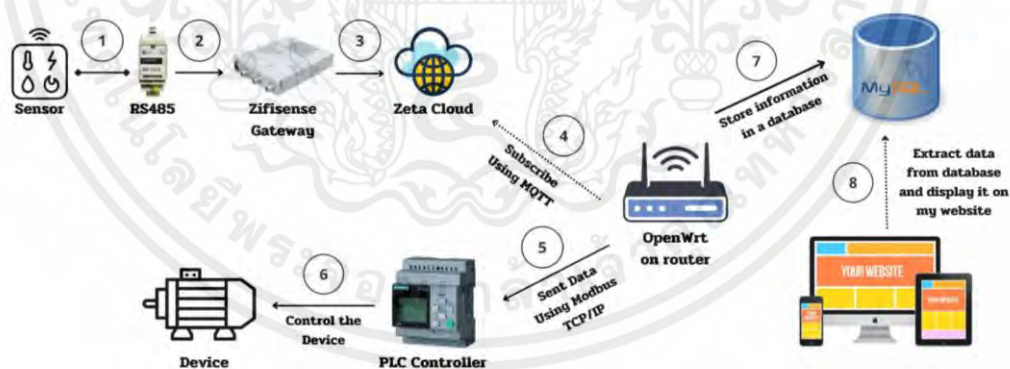
### 1.2 วัตถุประสงค์

- 1) เพื่อศึกษาหลักการทำงานของ PLC Controller
- 2) เพื่อศึกษาหลักการทำงานของ Modbus TCP/IP เพื่อทำให้ PLC Controller สามารถอ่านค่าและรับค่าจากอุปกรณ์อื่นมาเก็บไว้ได้
- 3) เพื่อศึกษาหลักการทำงานของ OpenWrt บนอุปกรณ์ Router
- 4) เพื่อศึกษาหลักการใช้งานของ LOGO! 8.3 Software
- 5) เพื่อศึกษาหลักการทำงานของ MQTT Protocol
- 6) เพื่อศึกษาหลักการทำงานของ RS-485 ร่วมกับ Sensor
- 7) เพื่อศึกษาหลักการทำงานของ PLC Controller ร่วมกับระบบ IoT ของ Zeta Network
- 8) เพื่อออกแบบ Website สำหรับแสดงผลข้อมูลและสถานะต่างๆ
- 9) เพื่อออกแบบระบบควบคุมอัตโนมัติจากการใช้งาน PLC Controller ร่วมกับระบบ IoT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1.3 ขอบเขตของปริญญาโท

ทำการออกแบบระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าจากการใช้งาน PLC Controller ร่วมกับระบบ IoT โดยมีการใช้งาน Sensor ร่วมกับ RS-485 เพื่อส่งข้อมูลผ่าน AP Gateway ขึ้นสู่ระบบ Zeta Cloud หลังจากนั้นจะใช้ MQTT Protocol บน Router ที่ติดตั้ง Firmware OpenWrt ในการ Subscribe ข้อมูลและทำการแปลงบิตข้อมูลที่ได้รับเป็นตัวเลขฐานสิบแล้วจึงนำมาทำการประมวลผลเพื่อสร้างเงื่อนไขในการควบคุมการเปิด-ปิดอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller ซึ่งจะมีการบันทึกค่าข้อมูลไว้บน Database ของ MySQL Server หลังจากนั้นจึงนำข้อมูลต่าง ๆ มาแสดงผลบน Website ที่ได้ทำการสร้างขึ้น โดยจะทำการเชื่อมต่อ PLC Controller กับ Router ผ่านระบบ LAN ซึ่ง PLC Controller จะใช้งาน Modbus TCP/IP ในการอ่านค่าหรือรับค่าจาก Router และนำข้อมูลที่ได้มาเก็บไว้ที่ตัว PLC Controller แล้วจึงทำการออกแบบโปรแกรมบน LOGO! 8.3 Software ในการสร้างการแสดงผลข้อมูลที่ได้และสถานะการทำงานของอุปกรณ์ Output บนหน้าจอ PLC Controller และ ควบคุมอุปกรณ์ Output ที่เชื่อมต่ออยู่ และจะทำการส่งข้อความแจ้งเตือนผ่านทาง LINE Notify เมื่ออุปกรณ์ Output ที่เชื่อมต่ออยู่ถูกสั่งให้ทำงาน ซึ่งจะทำการแสดงตัวอย่างการทำงานร่วมกันของระบบทั้งหมดโดยทำการจำลองเป็นตู้สำหรับปลุกต้นไม้ในที่ร่มอย่างง่าย



รูปที่ 1.1 Block Diagram ภาพรวมของปริญญาโท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

ปริญญาบัตรเรื่อง “การใช้งาน PLC Controller ร่วมกับระบบ IoT เพื่อสร้างระบบควบคุมอัตโนมัติ” มีทฤษฎีและหลักการสำคัญที่เกี่ยวข้องดังต่อไปนี้

#### 2.1 IoT (Internet of Things)

คำว่า IoT หรืออินเทอร์เน็ตในทุกสิ่ง (Internet of Things) หมายถึงเครือข่ายรวมของอุปกรณ์ที่เชื่อมต่อถึงกันและเทคโนโลยีที่อำนวยความสะดวกในการสื่อสารระหว่างอุปกรณ์กับระบบคลาวด์ ตลอดจนระหว่างอุปกรณ์ด้วยกันเอง จากการเกิดขึ้นของชิปคอมพิวเตอร์ราคาไม่แพงและการสื่อสารโทรคมนาคมที่มีแบนด์วิดท์สูง จึงทำให้มีอุปกรณ์หลายพันล้านเครื่องที่เชื่อมต่อกับอินเทอร์เน็ต ซึ่งหมายความว่าอุปกรณ์ต่างๆ ในชีวิตประจำวัน เช่น แปรงสีพื้น เครื่องดูดฝุ่น รถยนต์ และเครื่องจักรสามารถใช้เซ็นเซอร์เพื่อรวบรวมข้อมูลและตอบสนองต่อผู้ใช้ได้อย่างชาญฉลาด

##### 2.1.1 องค์ประกอบหลักของเทคโนโลยี IoT

องค์ประกอบของ IoT จะแบ่งออกได้เป็น 4 องค์ประกอบหลัก ได้แก่

2.1.1.1 อุปกรณ์ IoT คือ องค์ประกอบหลักของระบบเลย ซึ่งหากปราศจากตัวอุปกรณ์อิเล็กทรอนิกส์ระบบ IoT จะไม่สามารถเกิดขึ้นได้เลย ซึ่งตัวอุปกรณ์จะสามารถเป็นได้ทั้งอุปกรณ์สั่งการโดยจะมีคอมพิวเตอร์ขนาดเล็กไว้ประมวลผลข้อมูล หลังจากได้ผลลัพธ์ออกมา ก็จะมีระบบสั่งการที่จะสั่งการไปยังอุปกรณ์ IoT อีกตัวซึ่งมีระบบที่สามารถรับคำสั่งได้ก็จะทำงานตามที่ได้รับคำสั่งมา

2.1.1.2 อุปกรณ์เกตเวย์ คือ อุปกรณ์อีกประเภทที่จะเป็นตัวกลางในการส่งผ่านคำสั่งการจากอุปกรณ์ IoT ไปยังอุปกรณ์เป้าหมายที่ต้องรับคำสั่งเพื่อทำตาม

2.1.1.3 เครื่องบริการ Server หรือ Broker คือ เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นจุดศูนย์กลางในการประมวลผล เนื่องจากอุปกรณ์มักจะใช้หน่วยประมวลผลขนาดเล็กทำให้บางครั้งไม่สามารถคำนวณคำสั่งที่ซับซ้อนได้จึงจำเป็นต้องมีหน่วยประมวลผลกลางที่รับคำสั่งจากอุปกรณ์ IoT ประเภทหนึ่งหลังจากนั้นก็ใช้การประมวลผลเพื่อให้ได้คำสั่งอย่างง่ายส่งไปยังอุปกรณ์ที่เป็นผู้รับอีกทอด

2.1.1.4 อุปกรณ์ฝั่งผู้ใช้งาน (User Device) คือ ส่วนของอุปกรณ์ที่จะใช้แสดงผลสถานะจากอุปกรณ์ IoT ทั้งหมดที่อยู่ในเครือข่าย เนื่องจากโดยปกติระบบ IoT จะเป็นการเชื่อมต่อเพื่อสื่อสารกันระหว่างอุปกรณ์อิเล็กทรอนิกส์ ซึ่งมนุษย์จำเป็นต้องมีอุปกรณ์อีกหนึ่งชิ้นเพื่อใช้ในการเชื่อมต่อกับระบบเพื่อดูสถานะการทำงาน

## 2.1.2 ประโยชน์ของระบบ IoT

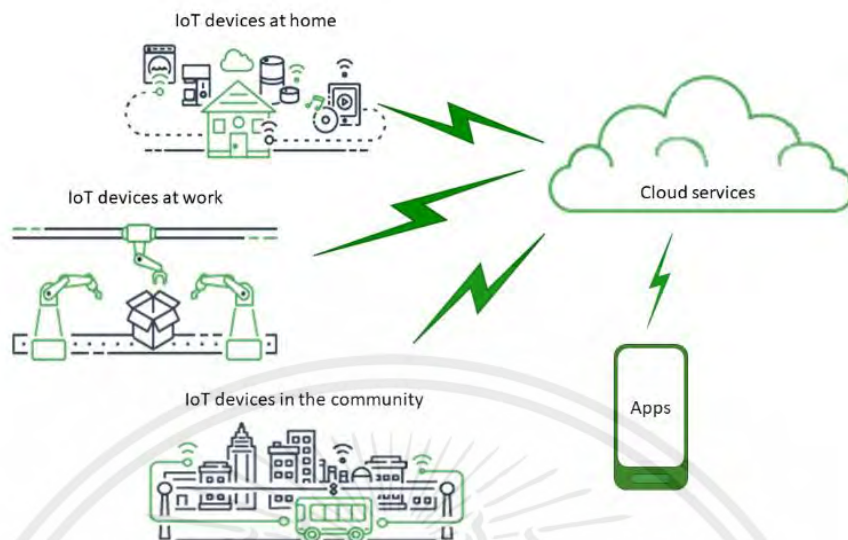
IoT ถือเป็นเทคโนโลยีที่สำคัญและมีประโยชน์ต่อแวดวงอุตสาหกรรมหลายอย่าง และเมื่อพูดถึงประโยชน์ของ IoT สำหรับคนทั่วไป ก็คงส่งผลดีต่อการใช้ชีวิตประจำวัน นอกเหนือจากภาพความทันสมัยและสะดวกสบายแล้ว การใช้เทคโนโลยี IoT ยังมีประโยชน์อีกหลายอย่างที่เรามองพิจารณาไว้ ได้แก่

2.1.2.1 ควบคุมการทำงานอุปกรณ์ต่าง ๆ ง่ายขึ้น โดยเทคโนโลยี IoT จะช่วยให้ควบคุมกระบวนการทำงานของอุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ ได้มีประสิทธิภาพยิ่งขึ้น โดย IoT จะเก็บข้อมูลว่ากิจกรรมในแต่ละวันเป็นอย่างไร ต้องทำกิจกรรมหรืองานอะไรบ้าง จากนั้นก็จะประมวลผล เพื่อนำไปเชื่อมต่อและสั่งการทำงานต่อไป

2.1.2.2 ประหยัดค่าใช้จ่าย ซึ่งสมาร์ทโฮมจะใช้พลังงานในการทำงานแต่อย่างได้คุ้มค่าและมีประสิทธิภาพมากกว่า เพราะเราสามารถกำหนดตั้งค่าการใช้งานอุปกรณ์อิเล็กทรอนิกส์ที่เชื่อมต่อกับระบบ IoT ซึ่งเอื้อต่อการประหยัดค่าใช้จ่ายโดยรวมได้ นอกจากนี้ เราสามารถตั้งค่าให้อุปกรณ์เหล่านั้นทำงานทุกอย่างได้เอง โดยระบบจะตัดดาว์นอุปกรณ์อื่นที่ไม่ได้ใช้งานขณะนั้น ส่งผลให้การใช้พลังงานที่ไม่จำเป็นลดลงตามไปด้วย

2.1.2.3 เป็นมิตรต่อสิ่งแวดล้อม ซึ่งนอกจากจะช่วยประหยัดพลังงานและค่าใช้จ่ายแล้วนั้น การประหยัดพลังงานยังส่งผลดีต่อสิ่งแวดล้อม ทำให้สภาพแวดล้อมถูกทำลายน้อยลง อีกทั้งยังส่งเสริมการใช้ชีวิตคุณภาพในสภาพแวดล้อมปลอดภัยยิ่งขึ้น

2.1.2.4 มีความปลอดภัยสูง โดยระบบรักษาความปลอดภัยของสมาร์ทโฮมจะช่วยปกป้องความเสียหายและการสูญหายของสินทรัพย์ภายในบ้านได้เป็นอย่างดี เพราะระบบ IoT จะมีชุดคำสั่งที่ช่วยควบคุมและดูแลความปลอดภัยสภาพแวดล้อมทั้งภายในและภายนอกบ้าน ไม่ว่าจะเป็น กล้องวงจรปิด ระบบปลดล็อกสมาร์ทโฟน หรือแม้กระทั่งระบบเซ็นเซอร์ตรวจจับการเคลื่อนไหวหรือควันอื่น ๆ อุปกรณ์เหล่านี้จะทำงานร่วมกันและส่งสัญญาณแจ้งเตือนในกรณีที่เกิดสิ่งผิดปกติ

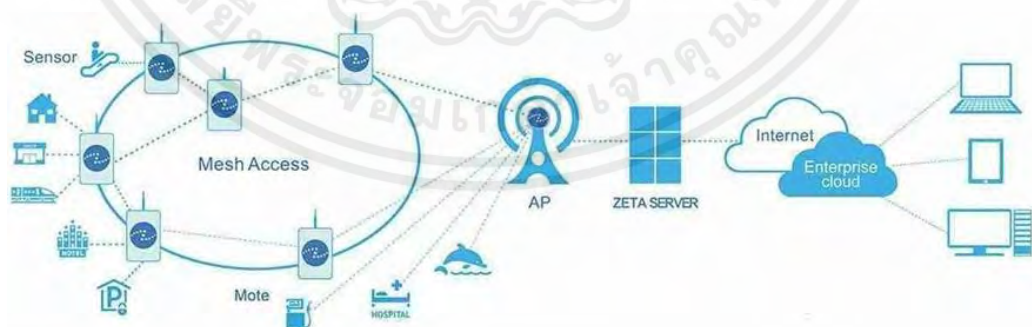


รูปที่ 2.1 ระบบ IoT

ที่มา : <https://d1.awsstatic.com/whatisimg/iot-universe.83de2944cc979eac98695797ecb5668f244bd7a5.png>

## 2.2 Zeta Network

การส่งข้อมูลจะเริ่มจาก sensor ส่งข้อมูลไปยัง AP Gateway จากนั้น AP Gateway จะทำการส่งค่าข้อมูลขึ้นไปเก็บไว้ยัง Zeta server



รูปที่ 2.2 ระบบ Zeta Network

ที่มา : <https://www.ecoxplore.com/wp-content/uploads/2018/10/zeta-structure.jpg>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 JSON (JavaScript Object Notation)

JSON (JavaScript Object Notation) คือ รูปแบบของข้อมูลที่ใช้สำหรับแลกเปลี่ยนข้อมูลที่มีขนาดเล็ก ซึ่งคนสามารถทำความเข้าใจได้ง่าย และสามารถถูกสร้างและอ่านโดยเครื่องได้ง่าย มันถูกกำหนดภายใต้ภาษา JavaScript (JavaScript Programming Language, Standard ECMA-262 3rd. Edition – December 1999.) JSON เป็นรูปแบบข้อมูลตัวอักษรที่มีความเป็นอิสระอย่างสมบูรณ์ แต่จะมีหลักการการเขียนที่คุ้นเคยกับนักเขียนโปรแกรมภาษาต่างๆ ได้ ไม่ว่าจะเป็น ภาษา C, C++, C#, Java, JavaScript, Perl, Python และอื่นๆ คุณสมบัติเหล่านี้ทำให้ JSON เป็นภาษาแลกเปลี่ยนข้อมูลที่มีความสมบูรณ์แบบ ปัจจุบัน JSON นิยมใช้ในเว็บแอปพลิเคชัน โดยเฉพาะ AJAX โดย JSON เป็นฟอร์แมตทางเลือกในการส่งข้อมูลนอกเหนือไปจาก XML ซึ่งนิยมใช้กันอยู่แต่เดิม สาเหตุที่ JSON เริ่มได้รับความนิยมเป็นเพราะกระชับและเข้าใจง่ายกว่า XML โดย JSON สามารถสร้างได้ 2 รูปแบบ คือ

1. การจัดเก็บในชุดข้อมูลที่มีชื่อข้อมูลและข้อมูลคู่กัน ในภาษาต่างๆ ข้อมูลจะจัดอยู่ในรูปแบบของ Object, record, struct, dictionary, hash table, keyed list หรือ associative array
2. ลำดับของค่าข้อมูล ในภาษาโปรแกรมส่วนใหญ่ จะจัดอยู่ในรูปแบบของ array, vector, list หรือ sequence

### 2.3.1 โครงสร้างของ JSON

JSON นั้นใช้ลักษณะภาษาของ JavaScript แต่ไม่ถูกมองว่าเป็นภาษาโปรแกรม กลับถูกมองว่าเป็นภาษาในการแลกเปลี่ยนข้อมูลมากกว่า ในปัจจุบันมีไลบรารีของภาษาโปรแกรมอื่นๆ ที่ใช้ประมวลผลข้อมูลในรูปแบบ JSON มากมาย ตัวอย่างของ JSON ในลักษณะของ Master – Detail

```
[
  {
    "firstname" : "name",
    "lastname": "name",
    "address" : [
      {
        "address1" : "adress",
        "province" : "bangkok"
      }
    ]
  }
]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 JavaScript

JavaScript เป็นภาษาโปรแกรมที่นักพัฒนาใช้ในการสร้างหน้าเว็บแบบอินเทอร์แอคทีฟ ตั้งแต่การรีเฟรชฟีดสื่อโซเชียลไปจนถึงการแสดงภาพเคลื่อนไหวและแผนที่แบบอินเทอร์แอคทีฟ ฟังก์ชันของ JavaScript สามารถปรับปรุงประสบการณ์ที่ผู้ใช้จะได้รับจากการใช้งานเว็บไซต์ และในฐานะที่เป็นภาษาในการเขียนสคริปต์ฝั่งไคลเอนต์ จึงเป็นหนึ่งในเทคโนโลยีหลักของ World Wide Web ยกตัวอย่างเช่น เมื่อท่องเว็บแล้วเห็นภาพสไลด์ เมนูหรือป๊อปอัพแบบคลิกให้แสดงผล หรือสื่อบรรยากาศที่เปลี่ยนแบบไดนามิกบนหน้าเว็บ นั่นคือเอฟเฟกต์ของ JavaScript

### 2.4.1 การทำงานของ JavaScript

ภาษาโปรแกรมทั้งหมดทำงานด้วยการแปลไวยากรณ์ที่คล้ายภาษาอังกฤษเป็นโค้ดสำหรับเครื่อง จากนั้นระบบปฏิบัติการจะเรียกใช้โค้ดนั้น JavaScript ได้รับการจัดประเภทอย่างกว้าง ๆ ว่าเป็นภาษาเขียนสคริปต์ หรือภาษาที่แปลผลแล้ว โค้ด JavaScript ได้รับการแปลผล นั่นคือ แปลโดยตรงเป็นโค้ดภาษาสำหรับเครื่อง ด้วยกลไก JavaScript ในขณะที่ในภาษาโปรแกรมอื่น ๆ คอมไพเลอร์จะคอมไพล์โค้ดทั้งหมดเป็นโค้ดสำหรับเครื่องในขั้นตอนที่แยกต่างหาก ดังนั้น ภาษาเขียนสคริปต์ทั้งหมดจึงเป็นภาษาโปรแกรม แต่ไม่ใช่ภาษาโปรแกรมทั้งหมดจะเป็นภาษาเขียนสคริปต์เสมอไป

### 2.4.2 กลไก JavaScript

กลไก JavaScript คือโปรแกรมคอมพิวเตอร์ที่เรียกใช้โค้ด JavaScript กลไก JavaScript เคยเป็นเพียงตัวแปลผล แต่กลไกสมัยใหม่ทั้งหมดใช้การคอมไพล์แบบ Just-in-time หรือ รันไทม์เพื่อปรับปรุงประสิทธิภาพ

### 2.4.3 JavaScript ฝั่งไคลเอนต์

JavaScript ฝั่งไคลเอนต์หมายถึงวิธีที่ JavaScript ทำงานในเบราว์เซอร์ของคุณ ในกรณีกลไก JavaScript จะอยู่ภายในโค้ดเบราว์เซอร์ เว็บเบราว์เซอร์เจ้าใหญ่ ๆ ทั้งหมดจะมาพร้อมกับกลไก JavaScript ในตัว นักพัฒนาแอปพลิเคชันเว็บจะเขียนโค้ด JavaScript ที่มีฟังก์ชันที่แตกต่างกันสัมพันธ์กับเหตุการณ์ต่าง ๆ เช่น การคลิกเมาส์ หรือการเลื่อนเมาส์ผ่าน ฟังก์ชันเหล่านี้จะเปลี่ยนแปลง HTML และ CSS

## 2.4.4 ภาพรวมการทำงาน JavaScript ฝั่งไคลเอนต์

2.4.4.1 เบราวเซอร์โหลดเว็บเพจเมื่อคุณเยี่ยมชมเว็บเพจ

2.4.4.2 ระหว่างการโหลด เบราวเซอร์แปลงหน้าและองค์ประกอบทั้งหมดของหน้า เช่น ปุ่ม ป้าย และกล่องครอบตัวอักษร เป็นโครงสร้างข้อมูลที่เราเรียกว่าโมเดลวัตถุเอกสาร (DOM)

2.4.4.3 กลไก JavaScript ของเบราว์เซอร์แปลงโค้ด JavaScript เป็นไบต์โค้ด โค้ดนี้เป็นตัวกลางระหว่างไวยากรณ์ JavaScript และเครื่อง

2.4.4.4 เหตุการณ์ต่าง ๆ เช่น การคลิกเมาส์บนปุ่ม จะกระตุ้นให้บล็อกโค้ด JavaScript ที่เกี่ยวข้องดำเนินการ จากนั้นกลไกจะแปลผลไบต์โค้ด และทำการเปลี่ยนแปลง DOM

2.4.4.5 เบราวเซอร์แสดงผล DOM ใหม่

## 2.4.5 JavaScript ฝั่งเซิร์ฟเวอร์

JavaScript ฝั่งเซิร์ฟเวอร์หมายถึงการใช้ภาษาเขียนโค้ดในลอจิกของเซิร์ฟเวอร์แบ็คเอนด์ ในกรณีนี้ กลไก JavaScript จะอยู่บนเซิร์ฟเวอร์โดยตรง ฟังก์ชัน JavaScript ฝั่งเซิร์ฟเวอร์สามารถเข้าถึงฐานข้อมูล ดำเนินการทางตรรกะแบบต่าง ๆ และตอบสนองต่อเหตุการณ์ต่าง ๆ ที่ถูกกระตุ้นจากระบบปฏิบัติการของเซิร์ฟเวอร์ ข้อได้เปรียบหลักของการเขียนสคริปต์ฝั่งเซิร์ฟเวอร์คือคุณสามารถปรับแต่งการตอบสนองของเว็บไซต์โดยอ้างอิงตามข้อกำหนดของคุณ สิทธิ์เข้าถึงของคุณ และคำขอข้อมูลจากระบบได้เป็นอย่างมาก

## 2.5 MQTT

### 2.5.1 โพรโตคอล MQTT

MQTT เป็นโพรโตคอลการส่งข้อความที่อิงตามมาตรฐาน หรือชุดของกฎที่ใช้สำหรับการสื่อสารระหว่างเครื่องต่อเครื่อง เช่น เซอร์วิซ อุปกรณ์สวมใส่ และอุปกรณ์ Internet of Things (IoT) อื่นๆ มักจะต้องส่งและรับข้อมูลผ่านเครือข่ายที่มีข้อจำกัดด้านทรัพยากร ซึ่งมีแบนด์วิดท์จำกัด อุปกรณ์ IoT เหล่านี้ใช้ MQTT ในการรับส่งข้อมูล เนื่องจากใช้งานง่ายและสามารถสื่อสารข้อมูล IoT ได้อย่างมีประสิทธิภาพ MQTT รองรับการส่งข้อความจากอุปกรณ์ไปยังคลาวด์และจากคลาวด์ไปยังอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5.2 หลักการของ MQTT

โปรโตคอล MQTT ทำงานบนหลักการแบบ เผยแพร่/สมัครรับ ในการสื่อสารผ่านเครือข่ายแบบดั้งเดิม โคลเอนต์และเซิร์ฟเวอร์จะสื่อสารกันโดยตรง โคลเอนต์จะร้องขอทรัพยากรหรือข้อมูลจากเซิร์ฟเวอร์ จากนั้นเซิร์ฟเวอร์จะประมวลผลและส่งข้อความตอบกลับมา อย่างไรก็ตาม MQTT ใช้รูปแบบ เผยแพร่/สมัครรับ เพื่อจะได้แยกผู้ส่งข้อความ (ผู้เผยแพร่) ออกจากผู้รับข้อความ (ผู้สมัครรับ) องค์ประกอบที่สามที่เรียกว่าโบรกเกอร์ข้อความจะเป็นตัวจัดการการสื่อสารระหว่างผู้เผยแพร่และผู้สมัครรับข้อมูลแทน งานของโบรกเกอร์คือการกรองข้อความขาเข้าทั้งหมดจากผู้เผยแพร่และกระจายไปยังผู้สมัครรับข้อความอย่างถูกต้องโบรกเกอร์จะแยกผู้เผยแพร่กับผู้สมัครรับข้อความออกจากกัน ดังนี้

2.5.2.1 การแยกเชิงพื้นที่ : ผู้เผยแพร่และผู้สมัครรับจะไม่ทราบตำแหน่งเครือข่ายของกันและกัน และไม่แลกเปลี่ยนข้อมูล เช่น ที่อยู่ IP หรือหมายเลขพอร์ต ซึ่งกันและกัน

2.5.2.2 การแยกเชิงเวลา : ผู้เผยแพร่และผู้สมัครรับไม่ต้องทำงานหรือมีการเชื่อมต่อเครือข่ายในเวลาเดียวกัน

2.5.2.3 การแยกการซิงโครไนซ์ : ทั้งผู้เผยแพร่และผู้สมัครรับสามารถส่งหรือรับข้อความได้โดยไม่รบกวนการทำงานกัน เช่น ผู้สมัครรับไม่ต้องรอให้ผู้เผยแพร่ส่งข้อความ

## 2.5.3 ส่วนประกอบของ MQTT

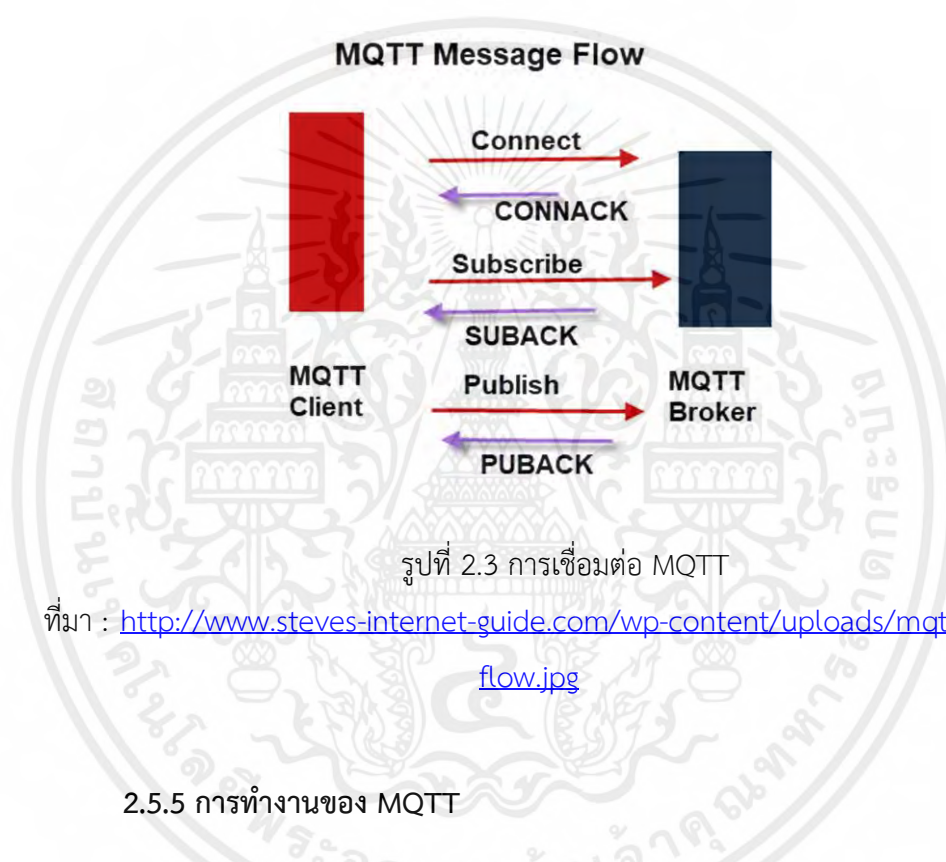
MQTT ใช้รูปแบบการเผยแพร่/สมัครรับข้อความ โดยกำหนดนิยามโคลเอนต์และโบรกเกอร์ ดังต่อไปนี้

2.5.3.1 โคลเอนต์ MQTT : คืออุปกรณ์ใดๆ ตั้งแต่เซิร์ฟเวอร์ไปจนถึงไมโครคอนโทรลเลอร์ที่เรียกใช้ไลบรารี MQTT หากโคลเอนต์ส่งข้อความ มันทำหน้าที่เป็นผู้เผยแพร่ และหากโคลเอนต์รับข้อความ มันทำหน้าที่เป็นผู้รับโดยทั่วไปแล้ว อุปกรณ์ใดๆ ที่สื่อสารโดยใช้ MQTT ผ่านเครือข่าย ก็สามารถเรียกว่าเป็นอุปกรณ์โคลเอนต์ MQTT

2.5.3.2 โบรกเกอร์ MQTT : เป็นระบบแบ็กเอนด์ที่ประสานงานการรับส่งข้อความระหว่างโคลเอนต์ต่างๆ หน้าที่ของโบรกเกอร์นั้นรวมถึงการรับและการกรองข้อความ การระบุโคลเอนต์ที่สมัครรับข้อความแต่ละข้อความ และการส่งข้อความไปที่โคลเอนต์เหล่านั้น นอกจากนี้ยังมีหน้าที่งานอื่น ๆ ด้วย เช่น การอนุญาตและการรับรองความถูกต้องให้โคลเอนต์ MQTT การส่งต่อข้อความไปยังระบบอื่นเพื่อการวิเคราะห์เพิ่มเติม การจัดการข้อความที่พลาดไป และเซสชันของโคลเอนต์

## 2.5.4 การเชื่อมต่อ MQTT

ไคลเอนต์และโบรกเกอร์เริ่มต้นการสื่อสารโดยใช้การเชื่อมต่อ MQTT ไคลเอนต์เริ่มต้นการเชื่อมต่อโดยส่งข้อความ CONNECT ถึงโบรกเกอร์ MQTT โบรกเกอร์ยืนยันว่าสร้างการเชื่อมต่อแล้ว โดยการตอบกลับด้วยข้อความ CONNACK ทั้งไคลเอนต์ MQTT และโบรกเกอร์จำเป็นต้องใช้สแต็ก TCP/IP ในการสื่อสาร ไคลเอนต์จะไม่เชื่อมต่อกันเอง แต่จะเชื่อมต่อเฉพาะกับโบรกเกอร์เท่านั้น



## 2.5.5 การทำงานของ MQTT

2.5.5.1 ไคลเอนต์ MQTT จะสร้างการเชื่อมต่อกับโบรกเกอร์ MQTT

2.5.5.2 เมื่อเชื่อมต่อแล้ว ลูกค้ายสามารถเผยแพร่ข้อความ สมัครรับข้อความที่เฉพาะเจาะจง หรือทำทั้งสองอย่างได้

2.5.5.3 เมื่อโบรกเกอร์ MQTT ได้รับข้อความแล้ว ก็ส่งต่อไปยังผู้สมัครรับข้อความที่ให้ความสนใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 Database and MySQL

Database ปัจจุบันมีการใช้งานหลักๆอยู่ 2 แบบ คือ

1. Relational database หรือ ระบบฐานข้อมูลเชิงสัมพันธ์ เป็นการเก็บข้อมูลในรูปแบบของตาราง (table) ในแต่ละตารางแบ่งออกเป็นแถวๆ และในแต่ละแถวจะแบ่งเป็นคอลัมน์ (Column) ซึ่งในการเชื่อมโยงกันระหว่างข้อมูลในตารางต่างๆ จะ เชื่อมโยงโดยใช้การอ้างอิงจากข้อมูลในคอลัมน์ที่กำหนดไว้ อาศัย RDBMS tools ในการควบคุม

2. LDAP database หรือ Lightweight Directory Access Protocol ทำงานโดยอาศัย Active Directory (AD) เป็นไต่แรกทอรีเซอร์วิสในระดับองค์กร ที่ถูกออกแบบบนมาตรฐานของ Internet Technology เอาไว้รองรับการค้นหาทรัพยากรต่าง ๆ บนเครือข่ายขนาดใหญ่ และยังช่วย Admin จัดการบริหารเครือข่ายที่ซับซ้อนจากศูนย์กลางได้อย่างสะดวก AD เป็นการทำงานร่วมกันระหว่าง DNS (Domain Naming System) และ LDAP (Lightweight Directory Access Protocol) ทำให้สามารถจะติดต่อเชื่อมโยง (interoperability) กับไต่แรกทอรีเซอร์วิสอื่นๆ ได้อีกด้วย

MySQL เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์โอเพนซอร์สบนพื้นฐานของ SQL ซึ่ง MySQL ได้รับการออกแบบและปรับให้เหมาะสมสำหรับเว็บแอปพลิเคชันและสามารถทำงานบนแพลตฟอร์มใดก็ได้ MySQL ทำงานเป็นดาต้าเบสเซิร์ฟเวอร์ และอนุญาตให้ผู้ใช้หลายคนจัดการและสร้างฐานข้อมูลจำนวนมาก



รูปที่ 2.4 MySQL Database

ที่มา : <https://d1.awsstatic.com/asset-repository/products/amazon-rds/1024px-MySQL.ff87215b43fd7292af172e2a5d9b844217262571.png>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.7 HTML (Hypertext Markup Language)

HTML ย่อมาจากคำว่า Hypertext Markup Language เป็นภาษาหลักที่ใช้ในการสร้างไฟล์เว็บเพจ โดยมีแนวคิดจากการสร้างเอกสารไฮเปอร์เท็กซ์ (Hypertext Document) ซึ่งพัฒนาขึ้นมาจากภาษา SGML (Standard Generalized Markup Language) โดย Tim Berners-Lee เป็นภาษามาตรฐานที่ใช้พัฒนาเอกสารในรูปแบบของเว็บเพจเผยแพร่บนระบบเครือข่ายอินเทอร์เน็ต มีโครงสร้างการเขียนที่อาศัยตัวกำกับ เรียกว่า แท็ก (Tag) ควบคุมการแสดงผลของข้อความ, รูปภาพ หรือวัตถุอื่นๆ เรียกใช้เอกสารเหล่านี้โดยการใช้โปรแกรมเว็บเบราว์เซอร์ (Web Browser) เช่น Mozilla Firefox, Opera , Netscape navigator, Internet Explorer ฯลฯ เป็นต้น ในปัจจุบัน HTML เป็นมาตรฐานหนึ่งของ ISO ซึ่งจัดการโดย World Wide Web Consortium (W3C) ในปัจจุบัน ทาง W3C ผลักดัน รูปแบบของ HTML แบบใหม่ ที่เรียกว่า XHTML ซึ่งเป็นลักษณะของโครงสร้าง XML แบบหนึ่งที่มีหลักเกณฑ์ในการกำหนดโครงสร้างของโปรแกรมที่มีรูปแบบที่มาตรฐานกว่า มาทดแทนใช้ HTML รุ่น 4.01 ที่ใช้กันอยู่ในปัจจุบันขณะที่ HTML รุ่น 5 ยังคงยังอยู่ในระหว่างการพิจารณาในการใช้งาน

### 2.7.1 โครงสร้างพื้นฐานของภาษา HTML

```
<html>
<head>
<title>หัวข้อเว็บเพจ</title>
</head >
<body>
ส่วนของเนื้อหา หรือข้อความที่จะใช้แสดงเนื้อหาในเว็บเพจ
</body>
</html>
```

2.7.1.1 <html>...</html> เป็นแท็กแรกที่ต้องมีในภาษา HTML ซึ่งบอกให้ทราบว่านี่คือ ภาษา HTML ใช้ในการกำหนดจุดเริ่มและจุดสิ้นสุดของเอกสาร HTML

2.7.1.2 <head>...</head> เป็นส่วนกำหนดรายละเอียดหัวเอกสาร HTML

2.7.1.3 <title>...</title> เป็นแท็กที่ใช้กำหนดชื่อเว็บเพจ ซึ่งภายแท็กจะเป็นชื่อเรื่องเว็บที่ต้องการ ซึ่งความยาวไม่เกิน 64 ตัวอักษร ข้อความนี้จะปรากฏที่ title bar ของเบราว์เซอร์ที่เราใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1.4 <body>...</body> ส่วนที่เป็นเนื้อหาเว็บเพจของเรา ไม่ว่าจะเป็นข้อความ รูปภาพ ตาราง และการเชื่อมโยงต่อไปยังเอกสารอื่น ๆ

## 2.8 CSS (Cascading Style Sheets)

CSS คือ ภาษาที่ใช้สำหรับตกแต่งเอกสาร HTML/XHTML ให้มีหน้าตา สี สัน ระยะห่าง พื้นหลัง เส้นขอบและอื่นๆ ตามที่ต้องการ CSS ย่อมาจาก Cascading Style Sheets มีลักษณะเป็นภาษาที่มีรูปแบบในการเขียน Syntax แบบเฉพาะและได้ถูกกำหนดมาตรฐานโดย W3C เป็นภาษาหนึ่งในการตกแต่งเว็บไซต์ ได้รับความนิยมน้อย่างแพร่หลาย CSS มีประโยชน์อย่างหลากหลาย ได้แก่

1. ช่วยให้เนื้อหาภายในเอกสาร HTML มีความเข้าใจได้ง่ายขึ้นและในการแก้ไขเอกสารก็สามารถทำได้ง่ายกว่าเดิม เพราะการใช้ CSS จะช่วยลดการใช้ภาษา HTML ลงได้ในระดับหนึ่ง และแยกแยะระหว่างเนื้อหาที่รูปแบบในการแสดงผลได้อย่างชัดเจน
2. ทำให้สามารถดาวน์โหลดไฟล์ได้เร็ว เนื่องจาก code ในเอกสาร HTML ลดลง จึงทำให้ไฟล์มีขนาดเล็กลง
3. สามารถกำหนดรูปแบบการแสดงผลจากคำสั่ง style sheet ชุดเดียวกัน ให้มีการแสดงผลในเอกสารแบบเดียวกันทั้งหน้าหรือในทุกๆ หน้าได้ ช่วยลดเวลาในการปรับปรุงและทำให้การสร้างเอกสารบนเว็บมีความรวดเร็วยิ่งขึ้น นอกจากนี้ยังสามารถควบคุมการแสดงผล ให้คล้ายหรือเหมือนกันได้ในหลาย Web Browser
4. ช่วยในการกำหนดการแสดงผลในรูปแบบที่มีความเหมาะสมกับสื่อต่างๆ ได้เป็นอย่างดี
5. ทำให้เว็บไซต์มีความเป็นมาตรฐานมากขึ้นและมีความทันสมัย สามารถรองรับการใช้งานในอนาคตได้ดี

### 2.8.1 ตัวอย่างโครงสร้างของภาษา CSS

```
<body>
  <p>ฉันจะเปลี่ยนสีแล้วนะ</p>
  <style>
p{
  color:yellow;
  font-size : 14px;
  background-color : black;
  border : 2px solid green;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

text-align : center;
font-family : sans-serif;
}
body{
background-color: black;
}
</style>
</body>

```

## 2.9 PHP (PHP Hypertext Preprocessor)

PHP ย่อมาจากคำว่า PHP Hypertext Preprocessor เป็นภาษาคอมพิวเตอร์ประเภท Server-Side Script ซึ่งจะทำการประมวลผลที่เซิร์ฟเวอร์ ใช้กับการพัฒนาเว็บไซต์ และสามารถแสดงผลและใช้คู่กับ HTML ได้ และ PHP สามารถใช้งานกับระบบปฏิบัติการ (Operating Systems) ที่หลากหลาย เช่น Linux (HP-UX, Solaris, และ OpenBSD), Microsoft, macOS และสามารถใช้งานได้กับเว็บเซิร์ฟเวอร์ เช่น Apache, Microsoft Internet Information Services (IIS) ได้นอกจากนั้นแล้ว PHP ยังสนับสนุนฐานข้อมูลรูปแบบต่างๆ ตั้งแต่ MySQL, PDO หรือ Open Database Connection

### 2.9.1 การใช้งาน PHP

คำสั่งภาษา PHP นี้จะต้องเขียนอยู่ระหว่างข้อความ <?php และ ?> ยกตัวอย่างด้านล่างคือโค้ดที่ให้แสดงคำว่า Hello World ในหน้า HTML เช่น

```

<?php
echo "Hello, World";
?>

```

### 2.9.2 ความสามารถของภาษา PHP

2.9.2.1 เป็นภาษาที่มีลักษณะเป็นแบบ Open source ผู้ใช้สามารถ Download และนำ Source code ของ PHP ไปใช้ได้โดยไม่เสียค่าใช้จ่าย

2.9.2.2 เป็นสคริปต์แบบ Server Side Script ดังนั้นจึงทำงานบนเว็บเซิร์ฟเวอร์ ไม่ส่งผลกับการทำงานของเครื่อง Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.2.3 PHP สามารถทำงานได้ในระบบปฏิบัติการที่ต่างชนิดกัน เช่น Unix, Windows, Mac OS หรือ Risc OS อย่างมีประสิทธิภาพ

2.9.2.4 PHP สามารถทำงานได้ในเว็บเซิร์ฟเวอร์หลายชนิด เช่น Personal Web Server(PWS), Apache, OmniHttpd และ Internet Information Service(IIS) เป็นต้น

2.9.2.5 ภาษา PHP สนับสนุนการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming)

2.9.2.6 PHP มีความสามารถในการทำงานร่วมกับระบบจัดการฐานข้อมูลที่หลากหลาย ซึ่งระบบจัดการฐานข้อมูลที่สนับสนุนการทำงานของ PHP เช่น Oracle, MySQL, FirePro, Solid, Front Base, MySQL และ MS SQL เป็นต้น

2.9.2.7 PHP อนุญาตให้ผู้ใช้สร้างเว็บไซต์ซึ่งทำงานผ่านโปรโตคอลชนิดต่างๆ ได้ เช่น LDAP, IMAP, SNMP, POP3 และ HTTP เป็นต้น

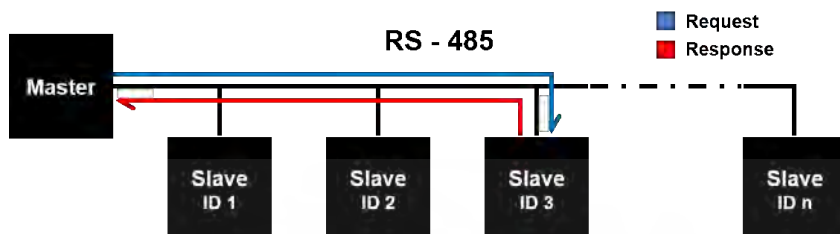
2.9.2.8 โค้ด PHP สามารถเขียน และอ่านในรูปแบบของ XML ได้

## 2.10 Modbus Protocol

การสื่อสารตามมาตรฐาน Modbus เป็นหนึ่งในมาตรฐานการสื่อสารแบบอนุกรม (Serial Communications protocol) ที่ใช้งานอย่างแพร่หลายในระบบอัตโนมัติอุตสาหกรรม (Industrial Automation Systems : IAS) เพื่อสร้างการเชื่อมโยงข้อมูลระหว่างอุปกรณ์ต่างๆ เช่น อุปกรณ์ควบคุมพีแอลซี (Programmable Logic Controllers : PLC) อุปกรณ์ตรวจวัด (Sensor) อุปกรณ์เครื่องกล อุปกรณ์ขับเคลื่อน (Actuator) หน่วยตรวจวัดระยะไกล (Remote Terminal Unit : RTU) รวมถึงระบบคอมพิวเตอร์ที่ใช้ในการควบคุมและแสดงสถานะของอุปกรณ์ต่างๆ (Supervisory control and Data acquisition : SCADA)

ในปัจจุบันนี้การสื่อสารสามารถแบ่งได้เป็น 2 ระบบคือ Modbus RTU และ Modbus TCP โดยความแตกต่างอยู่ที่โปรโตคอลการสื่อสารที่ใช้ ในระบบ Modbus RTU จะใช้โปรโตคอลการสื่อสารแบบอนุกรม (Serial-based Protocol) ในขณะที่ระบบ Modbus TCP จะใช้โปรโตคอลการสื่อสารแบบอีเทอร์เน็ต (Ethernet-based Protocol) ซึ่งทั้งสองแบบจะแตกต่างกันตรงที่ความเร็วและระยะทางในการรับส่งข้อมูล โดย Modbus RTU สามารถรับส่งได้ระยะทางสูงสุดถึง 1.2 กิโลเมตร (ที่ความเร็ว 57.6 kbps) ในขณะที่ Modbus TCP สามารถรับส่งได้ที่ความเร็ว สูงสุดถึง 100 Mbps (ที่ระยะทาง 100 เมตร)

### 2.10.1 Modbus RTU



รูปที่ 2.5 การสื่อสารแบบอนุกรมด้วย RS-485 สำหรับ Modbus RTU

ที่มา : <https://www.nectec.or.th/sectionImage/13787>

Modbus RTU คือ โพรโทคอลที่ใช้การสื่อสารแบบอนุกรม (Serial-based Protocol) ด้วยสถาปัตยกรรมการสื่อสารแบบ Master/Slave หรืออาจกล่าวได้ว่าอุปกรณ์ Slave จะไม่ส่งข้อมูล (Response) กลับมาจนกว่าจะมีการร้องขอ (Request) จากอุปกรณ์ Master ดังรูปที่ 2.5

โดยทั่วไปจะใช้การสื่อสารในระดับกายภาพ (Physical Layer) แบบ RS-232 หรือ RS-485 ข้อมูลในโพรโทคอล Modbus จะถูกเก็บ 4 รูปแบบ คือ

1. Output coils
2. Input contacts
3. Input registers
4. Holding registers

โดย Output coils และ Input contacts แต่ละแอดเดรสจะเก็บค่าเพียง 1 บิต หรือมีค่าได้แค่ “0” กับ “1” เปรียบเสมือนค่าการเปิดและปิดของอุปกรณ์รีเลย์และสวิตช์ที่พบได้ในระบบงานอัตโนมัติอุตสาหกรรม ในขณะที่ Input registers และ Holding registers สามารถเก็บค่าเป็นตัวเลขได้ถึง 16 บิต เปรียบเสมือนค่าที่มาจากอุปกรณ์ตรวจวัดที่ส่งข้อมูลแบบอนาล็อก (Analog)

การสื่อสารของข้อมูลในระบบ Modbus RTU จะรับส่งเป็นชุดข้อมูล โดยที่ใน 1 ชุดข้อมูลนั้นจะประกอบด้วยส่วน 6 ส่วน ดังแสดงในตารางที่ 2.1

ตารางที่ 2.1 ชุดข้อมูลสำหรับการสื่อสาร Modbus RTU

Field Name	Bit length	Function
Start	28	At least 3.5 Character times of silence (mark condition)
Address	8	Station address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function	8	Indicates function code e.g. read coils/holding registers
Data	n x 8	Data + length will be filled depending on message type
CRC	16	Cyclic Redundancy Check
End	28	At least 3.5 character times of silence between frames

1. เริ่มต้นด้วยชุดบิตเริ่มต้น (Start bits) อ้างอิงถึงการเริ่มต้นชุดข้อมูล
2. ค่าตำแหน่งแอดเดรส (Address) ของอุปกรณ์ที่ต้องการสื่อสารด้วย
3. ชุดสำหรับ Function Code
4. ข้อมูลที่ต้องการ (Data)
5. ชุดข้อมูลตรวจสอบความผิดพลาด (Cyclic Redundancy Check : CRC)
6. ชุดบิตปิดท้าย (End bits) อ้างอิงถึงการสิ้นสุดข้อมูล

#### 2.10.1.1 ตำแหน่ง Address ของ Modbus RTU

ตำแหน่งแอดเดรสใน Modbus RTU จะมีขนาด 16 บิต หรือ 65535 ตำแหน่ง ในแต่ละรูปแบบการทำงาน ดังตารางที่ 2.2 รุ่นเก่าอาจจะมีได้เพียง 9999 ตำแหน่งในช่วง

- Output coils : ตำแหน่งแอดเดรสจะเริ่มต้นที่ 000001
- Input contacts : ตำแหน่งแอดเดรสจะเริ่มต้นที่ 100001
- Input registers : ตำแหน่งแอดเดรสจะเริ่มต้นที่ 300001
- Holding registers : ตำแหน่งแอดเดรสจะเริ่มต้นที่ 400001

ตารางที่ 2.2 ตำแหน่งแอดเดรสใน Modbus RTU โดยแบ่งตามรูปแบบการทำงาน

Register Number (DEC)	Register Address (HEX)	Extended Register Number (DEC)	Extended Register Address (HEX)	Type	Object Type
00001-09999	0000 to 270E	000001-065535	0000 to FFFF	Read-Write	Output Coils
10001-19999	0000 to 270E	100001-165535	0000 to FFFF	Read-Only	Input Contacts
30001-39999	0000 to 270E	300001-365535	0000 to FFFF	Read-Only	Input Registers

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

40001-49999	0000 to 270E	40001-465535	0000 to FFFF	Read-Write	Holding Registers
-------------	--------------	--------------	--------------	------------	-------------------

### 2.10.1.2 ฟังก์ชันการทำงานสำหรับ Modbus RTU (Function code)

ชุดฟังก์ชันการทำงานสามารถแบ่งหน้าที่ต่างๆ ได้ตามรหัส หรือ Function code รายละเอียดแสดงดังตารางที่ 2.3 โดยหลักๆ แล้วจะมีฟังก์ชันการทำงานอยู่ 2 แบบ คือ การอ่าน (Read) และเขียน (Write) โดยสามารถเลือกที่จะอ่านหรือเขียนข้อมูลไปยัง Coils หรือ Contacts สำหรับข้อมูลแบบดิจิทัล (Digital) หรือ “0” กับ “1” และ Registers สำหรับอ่านหรือเขียนข้อมูลแบบอนาล็อก โดยมีขนาด 16 บิต หรือ ตั้งแต่ 0000 ถึง FFFF

ตารางที่ 2.3 รายละเอียดชุดข้อมูล Function Code

Function Code (DEC)	Action	Data Type	Object Type
01	Read	Single bit	Output Coils
05	Write Single	Single bit	Output Coils
15	Write Multiple	Single bit	Output Coils
02	Read	Single bit	Input Contacts
04	Read	Word (16bit)	Input Registers
03	Read	Word (16bit)	Holding Registers
06	Write Single	Word (16bit)	Holding Registers
16	Write Multiple	Word (16bit)	Holding Registers

### 2.10.1.3 ชุดข้อมูล (Data)

ในส่วนชุดข้อมูล Data Field นั้นจะถูกแบ่งเป็น 2 ชุด ได้แก่

1. ชุดคำสั่งสำหรับการอ่าน (Read Command) ตามตารางที่ 2.4
2. ชุดคำสั่งสำหรับการเขียน (Write Command) ตามตารางที่ 2.5

โดยชุดคำสั่งทั้ง 2 จะถูกส่งจากอุปกรณ์ที่ทำหน้าที่เป็น Master เท่านั้น เพื่อส่งไปยังอุปกรณ์ Slave ที่ต้องการสื่อสาร

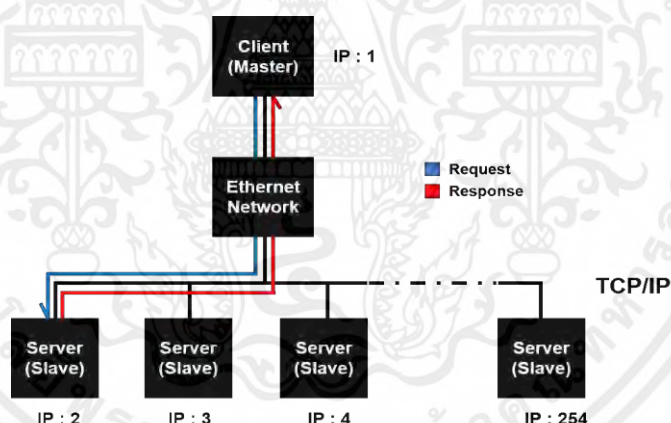
ตารางที่ 2.4 ชุดคำสั่งสำหรับการอ่าน (Read Command)

Read Command	
Request Message	Start register address (2bytes) + no. of registers (2bytes)
Response Message	Byte count (1byte) + data (no. of registers * 2bytes)

ตารางที่ 2.5 ชุดคำสั่งสำหรับการเขียน (Write Command)

Write Command	
Request Message	Start register address (2bytes) + no. of register (2bytes) + byte count (1 byte) + data (no. of registers * 2 bytes)
Response Message	Start register address (2 bytes) + no. of registers (2 bytes)

### 2.10.2 Modbus TCP



รูปที่ 2.6 การสื่อสารแบบอีเทอร์เน็ตสำหรับ Modbus TCP

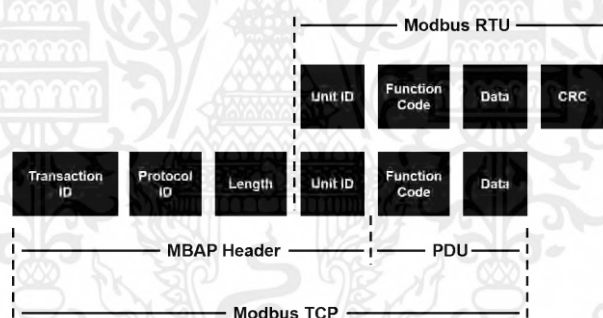
ที่มา : <https://www.nectec.or.th/sectionImage/13820>

Modbus TCP คือ โพรโทคอลที่ครอบ Modbus RTU เพื่อใช้การสื่อสารแบบอีเทอร์เน็ต (Ethernet-based protocol) ด้วย TCP/IP (Transmission control protocol) ที่พอร์ต (Port) 502 แทนการใช้การสื่อสารแบบอนุกรม ดังรูปที่ 2.6 ทำให้อุปกรณ์สามารถสร้างการสื่อสารผ่านเครือข่ายเฉพาะบริเวณ (Local area network : LAN) หรือ เครือข่ายอินเทอร์เน็ต (Internet network) รวมไปถึงการเชื่อมต่อแบบไร้สาย (Wireless) โดยมีอุปกรณ์กระจายสัญญาณ (Router หรือ Access point) เป็นตัวกลางในการเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดข้อความใน Modbus TCP (ดังรูปที่ 2.7 และ ตารางที่ 2.6) เริ่มต้นข้อมูลด้วย Modbus application protocol (MBAP) Header ซึ่งประกอบด้วย Transaction ID, Protocol ID, Length, Unit ID ซึ่งเพิ่มเติมขึ้นมาจาก Modbus RTU ส่วนชุดข้อมูล Function code และ Data จะยังคงเหมือนเดิม ยกเว้นชุดข้อมูล CRC สำหรับเช็คความผิดพลาดจะไม่มี แต่เปลี่ยนไปใช้ของ Ethernet ใน Data link layer แทน

1. Transaction Identifier (2 Bytes) : ใช้จับคู่การแลกเปลี่ยนข้อมูลเมื่อมี Message หลายชุด ถูกส่งออกมาด้วย TCP เดียวกัน ด้วย Client ตัวหนึ่ง โดยไม่ต้องรอลำดับการ Response
2. Protocol Identifier (2 bytes) : ในส่วนจะมีค่าเป็น 0 เสมอ
3. Length (2 bytes) : เป็นการระบุจำนวน Byte ที่รวมจำนวน Byte ของ unit identifier, function code, และ data fields
4. Unit Identifier (1 byte) : เป็นการระบุ ID ของ server ที่อยู่ในระบบสื่อสาร อาจตั้งเป็น 00 ถึง FF ก็ได้



รูปที่ 2.7 ส่วนประกอบชุดข้อมูลของ Modbus TCP เทียบกับ Modbus RTU

ที่มา : <https://www.nectec.or.th/sectionImage/13823>

ตารางที่ 2.6 รายละเอียดของแต่ละ Field ในหนึ่งเฟรมของ Modbus TCP

Area Name		Area Size	Description
MBAP header (MODBUS application header)	Transaction ID	2 bytes	Used by the master for matching of the response message from the slave.
	Protocol ID	2 bytes	Indicates the protocol of the PDU (protocol data unit) Stores 0 in the case of MODBUS TCP.
	Message	2 bytes	Stores the message size in byte unit.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	length		The message length after this field is stored. (See the above figure.)
	Module ID	1 byte	Used to specify the slave connected to the other line, e.g. MODBUS serial protocol.
PDU (Protocol data unit)	Function code	1 byte	The master specifies the processing to be performed for the slave.
	Data	1 to 252 bytes	[master sends request message to slave] Stores the requested processing. [slave sends response message to master] Stores the result of processing execution.

## 2.11 RS-485 (Recommended Standard no. 485)

RS-485 (ย่อมาจาก: Recommended Standard no. 485) คือมาตรฐานการสื่อสารข้อมูลดิจิทัลแบบอนุกรม (serial communication) ซึ่งถูกกำหนดขึ้นครั้งแรกในปี ค.ศ. 1998 โดยความร่วมมือของ TIA (Telecommunications Industry Association) และ EIA (Electronic Industries Association) มาตรฐาน RS-485 ถูกใช้อย่างแพร่หลายในโรงงานอุตสาหกรรม เนื่องจากสามารถส่งสัญญาณได้ไกลและยังสามารถส่งพร้อมๆกันได้หลายจุด

### 2.11.1 หลักการทำงานของ RS-485

มาตรฐาน RS-485 เป็นมาตรฐานที่รับ/ส่งข้อมูลในแบบที่เรียกว่า Half duplex คือสามารถรับและส่งข้อมูลได้ทีละอย่างเท่านั้นไม่สามารถทำทั้งสองอย่างได้ในเวลาเดียวกัน



รูปที่ 2.8 ตัวอย่างการเชื่อมต่อ RS-485 ระหว่างเครื่องมือวัดกับตัวแปลงสัญญาณ

ที่มา : [https://www.omi.co.th/image/upload/Image/article\\_056\\_RS485-wiring.jpg](https://www.omi.co.th/image/upload/Image/article_056_RS485-wiring.jpg)

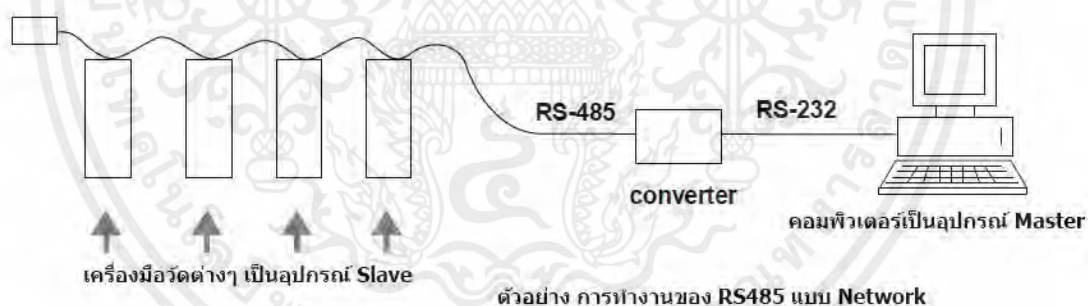
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการรับ/ส่งข้อมูลดิจิทัลแบบ RS-485 นั้น จะส่งข้อมูลโดยใช้สายไฟเพียงแค่ 2 เส้นคือ A และ B เป็นตัวบอกรหัสดิจิทัล(Digital code) โดยใช้ความแตกต่างของแรงดันไฟฟ้าระหว่างขั้ว A และ B เป็นตัวบอก ดังนี้

- เมื่อ  $V_a - V_b$  ได้แรงดันไฟฟ้าน้อยกว่า  $-200$  mV คือ สัญญาณดิจิทัลเป็น 1
- เมื่อ  $V_a - V_b$  ได้แรงดันไฟฟ้ามากกว่า  $+200$  mV คือ สัญญาณดิจิทัลเป็น 0

### 2.11.2 หลักการทำงานของ RS-485 แบบ NETWORK

มาตรฐาน RS-485 สามารถเชื่อมต่อการรับส่งข้อมูลแบบเครือข่าย (Network) โดยมีอุปกรณ์ในเครือข่ายได้สูงสุดถึง 32 ตัว ซึ่งในเครือข่าวนั้น จะต้องมีอุปกรณ์อยู่ 1 ตัว ทำหน้าที่คอยจัดคิวการสื่อสารในเครือข่าย ซึ่งเราจะเรียกอุปกรณ์ตัวนี้ว่า "Master" และอุปกรณ์ส่วนที่เหลือเราจะเรียกว่า "Slave" โดยที่ Slave แต่ละตัวจะมีหมายเลข Address ของตัวเอง และเมื่อตัว Master ต้องการสั่งการตัว Slave ตัว Master จะส่งชุดคำสั่งพร้อมหมายเลข Address ไปยังอุปกรณ์ Slave ทุกตัว เมื่ออุปกรณ์ Slave ได้รับคำสั่งและคำสั่งนั้นมีหมายเลข Address ตรงกับตัวเอง อุปกรณ์ Slave ถึงจะทำตามคำสั่งของ Master เป็นลำดับไป



รูปที่ 2.9 ตัวอย่างการทำงานของ RS-485 แบบ Network

ที่มา : [https://www.omi.co.th/image/upload/Image/article\\_057\\_RS485-wiring-network.jpg](https://www.omi.co.th/image/upload/Image/article_057_RS485-wiring-network.jpg)

### 2.11.3 ข้อดีของสัญญาณ RS-485

2.11.3.1 สามารถส่งสัญญาณได้ไกล โดย RS-485 สามารถส่งสัญญาณได้ไกลสูงสุดถึง 1,200 เมตร ซึ่งถือว่าเป็นระยะทางที่ไกลมาก เพียงพอต่อการใช้งานในโรงงานอุตสาหกรรมอย่างแน่นอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.3.2 สามารถเชื่อมต่อเป็นเครือข่ายได้ ซึ่งนอกจากจะส่งสัญญาณได้ไกลแล้ว RS-485 ยังสามารถเชื่อมต่อเป็นเครือข่าย (Network) แบบ Multipoint ได้ด้วย ซึ่งสามารถเชื่อมต่ออุปกรณ์ในระบบได้สูงสุดถึง 32 ตัว

2.11.3.3 ประหยัดงบประมาณในการเดินสาย เพราะ มาตรฐาน RS-485 เป็นมาตรฐานที่ใช้สายไฟเพียง 2 เส้นในการรับส่งข้อมูล ซึ่งราคาสายเคเบิลแบบ 2 แกน จะมีราคาถูก

#### 2.11.4 ข้อเสียของสัญญาณ RS-485

2.11.4.1 ต้องใช้ตัวแปลงสัญญาณในการเชื่อมต่อคอมพิวเตอร์ เนื่องจากปัจจุบันคอมพิวเตอร์ที่เราใช้กันอยู่นั้นไม่มี Port เชื่อมต่อสัญญาณ RS-485 โดยตรง จะมีก็แค่ USB หรือ RS-232 เท่านั้น เพราะอย่างนั้นหากเราจะเชื่อมต่ออุปกรณ์ที่ใช้ RS-485 กับคอมพิวเตอร์นั้น เราต้องเสียงบประมาณเพิ่มขึ้นในการซื้อตัวแปลงสัญญาณ (Converter) เพื่อแปลงสัญญาณจาก RS-485 เป็น USB หรือ RS-232 ในการเชื่อมต่อนั่นเอง

2.11.4.2 ความเร็วในการรับ-ส่งข้อมูล ถึงแม้ RS-485 จะถูกพัฒนาด้านความเร็วในการรับส่งข้อมูลขึ้นมากแล้วก็ตามเมื่อเทียบกับมาตรฐานเก่า แต่ก็ยังมีความล่าช้าอยู่เมื่อเชื่อมต่อในลักษณะเครือข่ายจำนวนมากๆ

ตารางที่ 2.7 การเปรียบเทียบคุณสมบัติต่างๆของ RS-232, RS-423, RS-422, RS-485

	RS-232	RS-423	RS-422	RS-485
Differential	No	No	Yes	Yes
Max number of drivers	1	1	1	32
Max number of receivers	1	10	10	32
Modes of operation	Half duplex Full duplex	Half duplex	Half duplex	Half duplex
Network topology	Point-to-Point	Multidrop	Multidrop	Multipoint
Max distance (acc. Standard)	15 m	1200 m	1200 m	1200 m
Max speed at 12 m	20 kbs	100 kbs	10 Mbs	35 Mbs
Max speed at 1200 m	(1 kbs)	1 kbs	100 kbs	100 kbs
Max slew rate	30 V/ $\mu$ s	adjustable	n/a	n/a

Receiver input resistance	3-7 k $\Omega$	$\geq 4$ k $\Omega$	$\geq 4$ k $\Omega$	$\geq 12$ k $\Omega$
Driver load impedance	3-7 k $\Omega$	$\geq 450$ $\Omega$	100 $\Omega$	54 $\Omega$
Receiver input sensitivity	$\pm 3$ V	$\pm 200$ mV	$\pm 200$ mV	$\pm 200$ mV
Receiver input range	$\pm 15$ V	$\pm 12$ V	$\pm 10$ V	- 7-12 V
Max driver output voltage	$\pm 25$ V	$\pm 6$ V	$\pm 6$ V	- 7-12 V
Min driver output voltage (with load)	$\pm 5$ V	$\pm 3.6$ V	$\pm 2.0$ V	$\pm 1.5$ V

## 2.12 DNS (Domain Name Server)

หน้าที่ของ DNS คือจะทำการแปลงชื่อ Domain เป็น IP Address เพื่อให้ง่ายต่อการจดจำ ซึ่ง DNS ในโลกนี้จะไม่ซ้ำกัน ดังนั้น DNS จึงมีความสำคัญในเชิงธุรกิจเพื่อการสร้างความจดจำเว็บไซต์ต่อลูกค้าด้วย นอกจากนี้ในทางเทคนิคแล้วการทำงานของ DNS ยังเกี่ยวข้องกับ IP address อีกด้วย มันจะสามารถบอกได้ว่าเว็บไซต์นั้นตั้งอยู่ที่ไหน

### 2.12.1 ส่วนประกอบ DNS คืออะไรและมีหน้าทียังไร

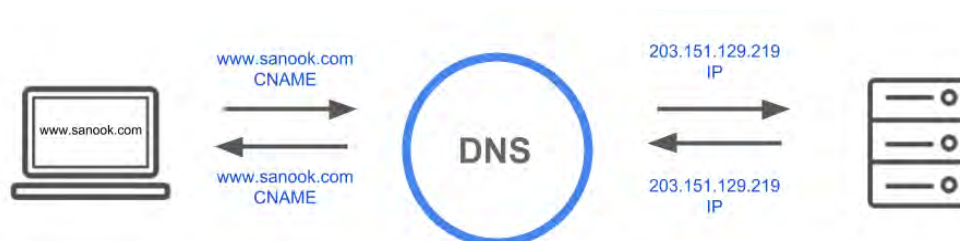
ถ้าต้องการหาที่อยู่ของเว็บไซต์ DNS คือ เครื่องมือที่จะใช้หาตำแหน่งปลายทางนั้น โดย DNS จะมีส่วนประกอบ 3 ส่วน คือ

2.12.1.1 Name Resolvers ของ DNS มีหน้าที่หลักในการแปลงชื่อคอมพิวเตอร์ ให้เป็นหมายเลข IP เครื่องลูกข่ายที่ต้องการทราบหมายเลข IP เราจะเรียกว่า resolver

2.12.1.2 Domain Name Space จะเป็นฐานข้อมูลระบบ DNS มีโครงสร้างเป็นต้นไม้ โดยแต่ละ Domain Name Space จะมีชื่อเรียกและสามารถมีโดเมนย่อยหรือที่เรียกว่า subdomain จะใช้จุดในการแบ่งระหว่างโดเมนหลักและโดเมนย่อย

2.12.1.3 Name Servers คือ เครื่องคอมพิวเตอร์แม่ข่ายที่มีหน้าที่จัดการฐานข้อมูลในระบบ DNS โดยจะใช้โปรแกรมตอบกลับการร้องขอที่ได้รับมาด้วยข้อมูลจากฐานข้อมูลตัวเอง หรือบางการทำงานก็สามารถเขียนโปรแกรมให้สามารถค้นหาข้อมูลจากฐานข้อมูลอื่นใน Name server ได้ ถ้าพบข้อมูลที่ได้รับการร้องขอก็ถือว่าเป็นเจ้าของโดเมนนั้นจะเรียกว่า Authoritative ถ้าไม่พบข้อมูลจะเรียกว่า Non-Authoritative

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 ขั้นตอนการทำงานของ DNS

ที่มา : <https://dnsblog.technologyland.co.th/wp-content/uploads/2018/11/Flow-Process-DNS.png>

## 2.13 PLC (Programmable Logic Controller)

เป็นอุปกรณ์ควบคุมการทำงานของเครื่องจักรหรือกระบวนการทำงานต่างๆ โดยภายในมี Microprocessor เป็นมันสมองสิ่งสำคัญ PLC จะมีส่วนที่เป็นอินพุตและเอาต์พุตที่สามารถต่อออกไปใช้งานได้ทันที ตัวตรวจวัดหรือสวิตช์ต่างๆ จะต่อเข้ากับอินพุต ส่วนเอาต์พุตจะใช้ต่อออกไปควบคุมการทำงานของอุปกรณ์หรือเครื่องจักรที่เป็นเป้าหมาย เราสามารถสร้างวงจรหรือแบบของการทำงานควบคุมได้โดยการป้อนเป็นโปรแกรมคำสั่งเข้าไปใน PLC นอกจากนี้ยังสามารถใช้งานร่วมกับอุปกรณ์อื่นเช่น เซนเซอร์ต่างๆ เครื่องพิมพ์ ซึ่งในปัจจุบันสามารถต่อ PLC หลายๆ ตัวเข้าด้วยกัน (Network) เพื่อควบคุมการทำงานของระบบให้มีประสิทธิภาพมากยิ่งขึ้น

### 2.13.1 โครงสร้างของ PLC ประกอบด้วย

2.13.1.1 ตัวประมวลผล (CPU) ทำหน้าที่คำนวณและควบคุม ซึ่งเปรียบเสมือนสมองของ PLC ภายในประกอบด้วยวงจรลอจิกหลายชนิดและมี Micro Processor Based ใช้แทนอุปกรณ์จำพวกรีเลย์ เคาน์เตอร์/ไทม์เมอร์ และซีควนเซอร์ เพื่อให้ผู้ใช้สามารถออกแบบวงจรโดยใช้ Relay Ladder Diagram ได้ CPU จะยอมรับข้อมูลจากอุปกรณ์อินพุตต่างๆ จากนั้นจะทำการประมวลผลและเก็บข้อมูลโดยใช้โปรแกรมจากหน่วยความจำ หลังจากนั้นจะส่งส่งข้อมูลที่เหมาะสมและถูกต้องออกไปยังอุปกรณ์เอาต์พุต

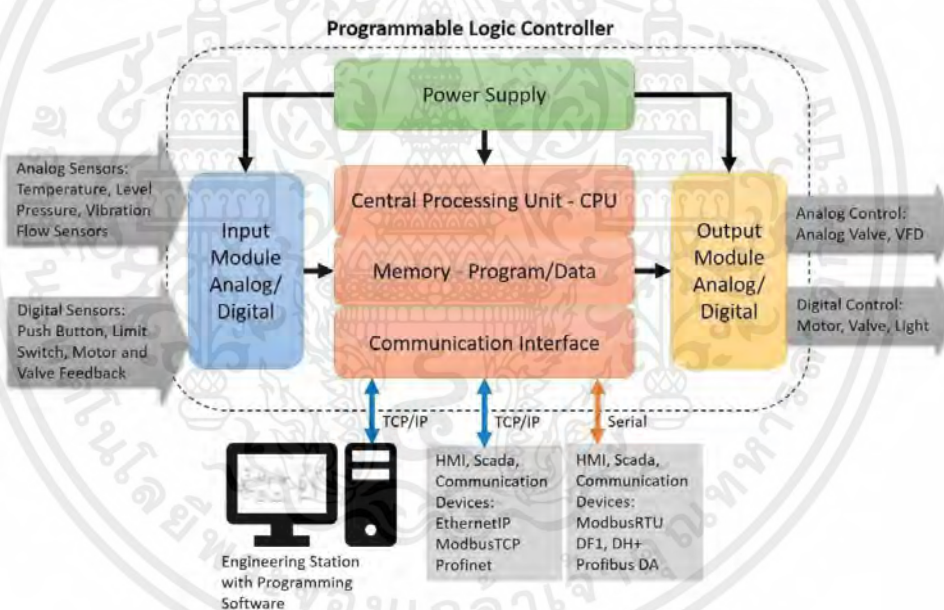
2.13.1.2 หน่วยความจำ (Memory Unit) ทำหน้าที่เก็บรักษาโปรแกรมและข้อมูลที่ใช้ในการทำงาน โดยขนาดของหน่วยความจำจะถูกแบ่งออกเป็นบิตข้อมูล (Data Bit) ภายในหน่วยความจำ 1 บิต ก็จะมีค่าสถานะทางลอจิก 0 หรือ 1 แตกต่างกันไปแล้วแต่คำสั่ง ซึ่ง PLC ประกอบด้วยหน่วยความจำสองชนิดคือ ROM และ RAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- RAM ทำหน้าที่เก็บโปรแกรมของผู้ใช้และข้อมูลที่ใช้ในการปฏิบัติงานของ PLC หน่วยความจำประเภทนี้จะมีแบตเตอรี่เล็กๆ ต่อไว้เพื่อใช้เป็นไฟเลี้ยงข้อมูลเมื่อเกิดไฟดับ

- ROM ทำหน้าที่เก็บโปรแกรมสำหรับการใช้ในการปฏิบัติงานของ PLC ตามโปรแกรมของผู้ใช้ หน่วยความจำแบบ ROM ยังสามารถแบ่งได้เป็น EPROM ซึ่งจะต้องใช้อุปกรณ์พิเศษในการเขียนและลบโปรแกรม เหมาะกับงานที่ไม่ต้องการเปลี่ยนแปลงโปรแกรม นอกจากนี้ยังมีแบบ EEPROM หน่วยความจำประเภทนี้ไม่ต้องใช้เครื่องมือพิเศษในการเขียนและลบโปรแกรม สามารถใช้งานได้เหมือนกับ RAM แต่ไม่ต้องใช้แบตเตอรี่สำรอง

2.13.1.3 หน่วยอินพุต-เอาต์พุต (Input-Output Unit) หน่วยอินพุต ทำหน้าที่รับสัญญาณจากอุปกรณ์ภายนอกแล้วแปลงสัญญาณให้เป็นสัญญาณที่เหมาะสมแล้วส่งให้หน่วยประมวลผลต่อไปเพื่อควบคุมอุปกรณ์ที่ต่อกับหน่วยของเอาต์พุต



รูปที่ 2.11 โครงสร้างของ PLC

ที่มา : <https://i0.wp.com/flutech.co.th/wp-content/uploads/2023/01/PLC-Architecture.jpg?resize=1245%2C800&ssl=1>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.13.2 LOGO! 230 RCE

PLC ขนาดเล็กที่สามารถโปรแกรมได้ รองรับพอร์ต Ethernet ซึ่งรองรับการสื่อสารผ่านระบบ LAN มีหน้าจอแสดงผลเปลี่ยนสีได้ 3 สี ขาว, ส้ม, แดง การโปรแกรมสามารถทำได้ด้วยโปรแกรม LOGO! Soft COMFORT และมี Web server ในตัวเครื่อง



รูปที่ 2.12 PLC LOGO! 230 RCE

ที่มา : <https://5.imimg.com/data5/SELLER/Default/2022/11/DW/FI/OS/163748252/siemens-plc-logo-230rce-di8-do4-6ed1052.jpg>

ตารางที่ 2.8 ตารางแสดงคุณสมบัติของ PLC LOGO! 230 RCE

Hardware Specification	
Supply voltage	115V/230V AC/DC
Line Frequency	Lower limit : 47 Hz Upper limit : 63 Hz
Number of digital inputs	8
Number of digital outputs	4 (relay)
Class of protection	IP20
Ambient temperature during operation	Min : -20 °C ; No condensation Max : 55 °C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.14 ZETA 485 Transceiver

เป็นโมดูลพลังงานต่ำที่ใช้อินเทอร์เฟซ RS-485 (Modbus RTU) เพื่อรวบรวมข้อมูลหรืออ่านค่าจาก Sensor และส่งข้อมูลขึ้น Zeta Cloud

ตารางที่ 2.9 แสดงพารามิเตอร์ผลิตภัณฑ์

Wireless Features	Transmission protocol	ZETA
	Frequency band	920-925 MHz
	Output power	20±3 mW
Electrical Features	Power supply	100~240 V AC
	Power frequency	50~60 Hz
	Stand-by current	≤ 10 μA
	Working current	≤ 70 mA
Physical Features	Size	36*104*59 mm
	Enclosure material	ABS
	Antenna	Monopole antenna
Sensor Characteristics	Interface	RS-485 RTU
Working Environment	Operating temperature	-20 °C~+75 °C
	Storage temperature	-30 °C~+85 °C

ตารางที่ 2.10 ตารางแสดงคุณสมบัติของอุปกรณ์ ZETA 485 Transceiver

Field Name	Bit length	Function
Start	28	At least 3.5 character times of silence (mark condition)
Address	8	Station address
Function	8	Indicates function code eg. read coils/holding registers
Data	n X 8	Data + length will be filled depending on message type
CRC	16	Cyclic Redundancy Check
End	28	At least 3.5 character times of silence between frames

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 ZETA 485 Transceiver

ที่มา : [https://makeshop-multi-images.akamaized.net/zetafactory/itemimages/000000000510\\_X2laeGh.png](https://makeshop-multi-images.akamaized.net/zetafactory/itemimages/000000000510_X2laeGh.png)

#### 2.14.1 ฟังก์ชันการสนับสนุน

2.14.1.1 หมายเลขเวอร์ชัน : รายงานหมายเลขเวอร์ชันซอฟต์แวร์ของเซนเซอร์เพียงครั้งเดียวหลังจากเปิดเครื่อง

2.14.1.2 รายงานสถานะ : รายงานข้อมูลสถานะเป็นระยะตามการตั้งค่า

2.14.1.3 ตั้งค่า/สอบถามช่วงการรายงานพารามิเตอร์: สามารถตั้งค่าหรือค้นหารอบการรายงานข้อมูล (ช่วง : 1~65535 นาที ค่าเริ่มต้น : 12\*60 นาที)

2.14.1.4 ตั้งค่า/รหัสที่อยู่แบบสอบถาม: การสื่อสาร 485 ต้องมีการจับคู่รหัสที่อยู่ (ช่วง: 1~247 ค่าเริ่มต้น: ไม่มี)

2.14.1.5 Set/Query register function code: สืบค้นหรือตั้งรหัสฟังก์ชันของ Modbus (ช่วง : 03 หรือ 04)

2.14.1.6 ตั้งค่า/พารามิเตอร์การรับข้อความค้นหา: เมื่อตั้งค่า คุณต้องป้อนที่อยู่ทีละทาบะเบียนไว้เพื่ออ่านและจำนวนไบต์ที่อ่าน คุณสามารถตั้งค่าพารามิเตอร์การได้มาหลายรายการพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14.1.7 ตั้งค่า/สอบถามพารามิเตอร์พอร์ตอนุกรม: พารามิเตอร์พอร์ตอนุกรมประกอบด้วย: อัตราบอด บิตข้อมูล บิตพาริตี บิตหยุด และพารามิเตอร์ต้องจับคู่กับอุปกรณ์เพื่ออ่านข้อมูลตามปกติ

2.14.1.8 ข้อมูลการสืบค้น: นอกเหนือจากการรอให้เซ็นเซอร์รายงานแล้วคุณยังสามารถสืบค้นข้อมูลปัจจุบันที่อุปกรณ์รวบรวมได้

2.14.1.9 ค้นหาหมายเลขเวอร์ชัน: คุณสามารถสอบถามหมายเลขเวอร์ชันซอฟต์แวร์ของเซ็นเซอร์ปัจจุบันได้

## 2.15 AP Gateway Zifisense

เป็นเกตเวย์ระหว่าง ZETA wireless network และ ZETA Server การโต้ตอบทั้งหมดระหว่าง ZETA wireless network และ ZETA Server จะทำผ่าน AP โดย AP สื่อสารกับอุปกรณ์เครือข่าย ZETA ที่เหลือผ่านโปรโตคอล ZETA และสื่อสารกับ ZETA Server ผ่านโปรโตคอล IP มาตรฐาน ZETA ซึ่ง AP รองรับการสื่อสารหลายช่องทาง การสื่อสารสองทาง ที่ไม่เพียงแต่รองรับการรับส่งสัญญาณ uplink ของอุปกรณ์ปลายทางเท่านั้นแต่ยังรองรับคำสั่ง downlink เช่น Broadcast, Multicast และ Unicast



รูปที่ 2.14 AP Gateway Zifisense

ที่มา : <https://en.zifisense.com/v4/images/network-devices/column1-2.png>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.11 AP Gateway specification

Wireless Features	Transmission protocol	ZETA
	Frequency band	Sub-GHz, adjustable according to local regulations
	Output power	20dBm max, adjustable according to local regulations
	Number of channels	1 (expandable to 3 channels)
Electrical Features	Power supply	POE or AC100-240V to DC 5V
	Maximum standby power	≤ 5 W
Physical Features	Weight	1.5 Kg
	Size	256*200*86 mm
	Enclosure material	Aluminum alloy
	Waterproof level	IP67
	Antenna	FRP Antenna
Working Environment	Operating temperature	-20°C~+70°C
	Storage temperature	-30°C~+85°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.16 Temperature and Humidity Sensor Zifisense

เป็น sensor ระบบ wireless ที่ใช้สัญญาณในย่านความถี่ 920-925 MHz สามารถวัดค่าอุณหภูมิและความชื้นโดยรอบได้



รูปที่ 2.15 Temperature and Humidity Sensor Zifisense

ที่มา : <http://zifisense.co.uk/network-devices/#5>

ตารางที่ 2.12 Temperature and Humidity Sensor Specification

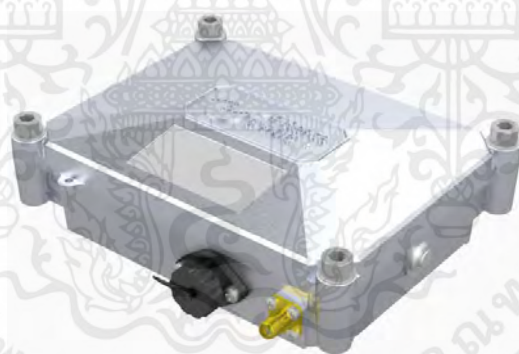
Wireless Features	Transmission protocol	ZETA
	Frequency band	920~925MHz
	Output power	20±3 mW
Electrical Features	Power supply	Battery, 2*ER14505
	Battery capacity	2*2700 mAh
	Stand-by current	≤ 5 μA
	Working current	≤ 70 mA
Physical Features	Size	66*55*36 mm
	Enclosure material	ABS
	Waterproof level	IP30
	Antenna	Monopole antenna
Sensor Characteristics	Temperature detection	-10°C~+85°C (±0.4°C)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	range & precision	
	Humidity detection range & precision	0~100%RH ±3%RH (0~80%RH) ±4.5%RH (80~100%RH)
Working Environment	Operating temperature	-20°C~+75°C
	Storage temperature	-30°C~+85°C

## 2.17 Mote Zifisense

เป็นตัวขยายสัญญาณซึ่งสามารถสร้างโครงข่ายประเภท tree ได้สูงสุด 4 ชั้น เพิ่มความครอบคลุมของ AP อย่างมีประสิทธิภาพ พร้อมการครอบคลุมพื้นที่อับสัญญาณ และป้องกันความแออัดของข้อมูล



รูปที่ 2.16 Mote Zifisense

ที่มา : <http://zifisense.co.uk/network-devices/#2>

ตารางที่ 2.13 Mote Specification

Wireless Features	Transmission protocol	ZETA
	Frequency band	Sub-GHz, adjustable according to local regulations
	Output power	20dBm max, adjustable according to local regulations
Electrical Features	Power supply	DC 5-12V, Battery, 2*ER34615
	Battery capacity	2*19000 mAh
	Stand-by current	≤ 10 μA
	Working current	≤ 70 mA
Physical Features	Weight	0.4 Kg
	Size	148.5*113.5*43.9 mm
	Enclosure material	ADC12 aluminum alloy
	Waterproof level	IP65
	Antenna	FRP Antenna
Working Environment	Operating temperature	-20°C~+70°C
	Storage temperature	-30°C~+85°C

## 2.18 RS-485 Soil Temperature and Humidity Sensor

เป็นเซ็นเซอร์ที่ใช้ในการวัดอุณหภูมิตั้งแต่ -40 ถึง 80 องศาเซลเซียส และวัดความชื้นตั้งแต่ 0 ถึง 100 เปอร์เซ็นต์ ซึ่งเป็นอุปกรณ์ที่ใช้วัดอุณหภูมิและความชื้นในดินโดยการนำอุปกรณ์ส่วนที่เป็นโพรบวัดค่า 3 ขา ปักลงไปในดิน โดยอุปกรณ์นี้สามารถเชื่อมต่อกับ Arduino esp8266 และ esp32 ได้โดยการส่งข้อมูลแบบ RS-485



รูปที่ 2.17 RS-485 Soil Temperature and Humidity Sensor

ที่มา : <https://inwfile.com/s-cq/nhfceq.jpg>

ตารางที่ 2.14 RS-485 Soil Temperature and Humidity Sensor Specification

Power supply	5~24 VDC
Power consumption	30mA*24 VDC
Humidity range	0~100 %RH
Humidity accuracy	±3 %RH
Temperature range	-40°C~+80°C
Temperature accuracy	±0.5°C
Response time	< 1s
Output signal	RS-485
Protection level	IP68
Dimensions	45*15*123 mm
Probe material	304 stainless steel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.19 Light Temperature and Humidity Sensor

เป็นเซ็นเซอร์ 3 in 1 ที่ใช้ในการวัดอุณหภูมิตั้งแต่ -20 ถึง 60 องศาเซลเซียส, วัดความชื้นตั้งแต่ 0 ถึง 80 เปอร์เซ็นต์ และวัดค่าความสว่างตั้งแต่ 0 ถึง 200,000 ลักซ์ ซึ่งเป็นอุปกรณ์ที่ใช้วัดค่าอุณหภูมิ ความชื้นและความสว่างในอากาศ



รูปที่ 2.18 Light Temperature and Humidity Sensor

ที่มา : [https://cm.lnwfile.com/\\_/cm/\\_raw/kk/li/lj.jpg](https://cm.lnwfile.com/_/cm/_raw/kk/li/lj.jpg)

ตารางที่ 2.15 Light Temperature and Humidity Sensor Specification

Power supply	5~24 VDC
Power consumption	24 VDC
Illumination measuring range	0~200,000 lux
Illumination accuracy	±7 %
Temperature range	-20°C~+60°C
Temperature accuracy	±0.5°C
Humidity range	0~80 %RH
Humidity accuracy	±3 %RH
Response time	< 1s
Output signal	RS-485
Probe material	PVC

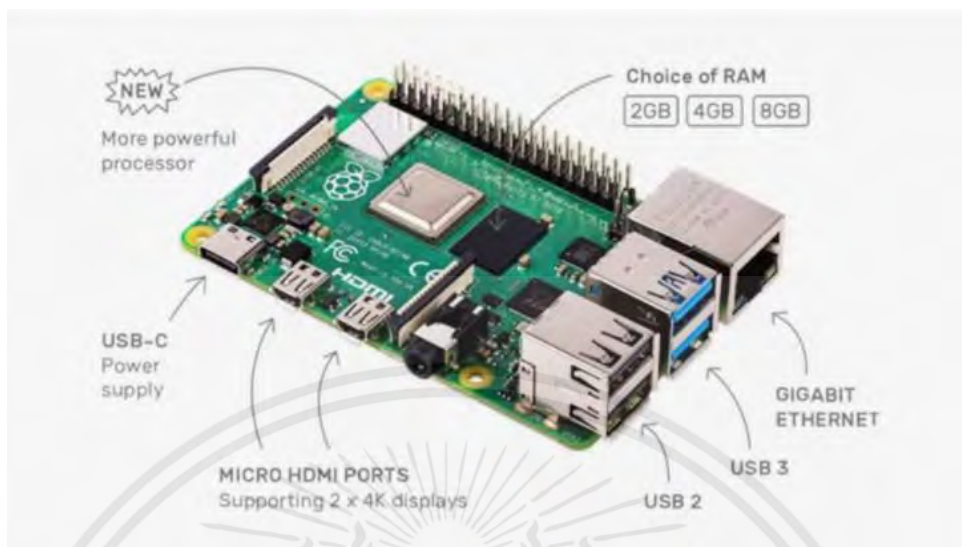
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.20 Raspberry Pi

เป็นเครื่องคอมพิวเตอร์ขนาดเล็ก ที่มีขนาดเพียงเท่ากับบัตรเครดิต ที่สำคัญคือ ราสเบอร์รี่พายนี้มีราคาที่ถูกมาก เมื่อเทียบกับคอมพิวเตอร์เดสก์ท็อปปกติ คือ มีราคาเพียงหนึ่งพันกว่าบาท แต่ในราคาเท่านี้ ทำงานได้เหมือนเครื่องคอมพิวเตอร์ทุกอย่างเลยนะครับ เราสามารถต่อ ราสเบอร์รี่พายนี้เข้ากับจอคอมพิวเตอร์หรือจอทีวีที่รองรับ HDMI หรือถ้าไม่มีพอร์ต HDMI ก็ไม่ต้องกังวล สามารถต่อผ่านสายสัญญาณวิดีโอปกติ (เส้นสีเหลือง) ได้เช่นกัน แต่ความละเอียดอาจจะต่ำกว่า นอกจากต่อจอแสดงผลแล้ว ก็ต้องต่ออุปกรณ์รับข้อมูล ราสเบอร์รี่พายนี้รองรับเมาส์และคีย์บอร์ดผ่าน USB port ปกติ เพราะฉะนั้นสามารถนำเมาส์และคีย์บอร์ดที่มีอยู่แล้วมาต่อได้เลย ระบบจ่ายไฟของราสเบอร์รี่พายก็ง่ายมากๆ เพียงเสียบสาย Mini USB ที่เราใช้ชาร์จมือถือและอุปกรณ์อื่นๆ เข้ากับคอมพิวเตอร์หรือเข้ากับหัวชาร์จไฟมือถือก็ได้เช่นกัน ซึ่งตัว Raspberry Pi ก็มีการรองรับการเขียนภาษาคอมพิวเตอร์ (Programming Language) อย่าง Scratch, HTML, CSS และภาษา Python ซึ่งเป็นภาษาคอมพิวเตอร์หลัก ๆ อีกด้วย

### 2.20.1 ส่วนประกอบภายใน Raspberry Pi

ส่วนประกอบของ Raspberry Pi หลัก ๆ ประกอบด้วย หน่วยประมวลผลกลาง (CPU), หน่วยความจำ RAM, ตัวรับสัญญาณไวไฟ (Wi-Fi Receiver) และ ตัวรับสัญญาณบลูทูธ (Bluetooth Receiver) ตามด้วยพอร์ตต่าง ๆ เช่น HDMI, Audio Output, USB หรือแม้แต่ LAN เป็นต้น ซึ่งจะเห็นได้ว่า ด้วยส่วนประกอบเหล่านี้ เราสามารถนำ Raspberry Pi มาใช้ทำคอมพิวเตอร์เดสก์ท็อปหนึ่งเครื่องได้เลย เพียงแค่คุณต้องหาซื้ออุปกรณ์มาเสริม อาทิ จอมอนิเตอร์, เมาส์, คีย์บอร์ด และสาย LAN มาเชื่อมต่อเอา และจุดเด่นของ Raspberry Pi ที่ทำให้มันกลายเป็นของเล่นยอดนิยมสำหรับนักประดิษฐ์ทั้งหลายก็คือ การที่มันมี พอร์ต เอนกประสงค์ หรือ GPIO (General Purpose Input Output) พอร์ตนี้สามารถนำมาประยุกต์ เพื่อเชื่อมต่อกับอุปกรณ์อิเล็กทรอนิกส์ หรือ อุปกรณ์เปิดปิดในบ้านได้ และควบคุมผ่าน Raspberry Pi เช่น สวิตช์ไฟ, ทีวี, พัดลม เป็นต้น



รูปที่ 2.19 ส่วนประกอบของ Raspberry Pi

ที่มา : <https://www.lifewire.com/what-is-raspberry-pi-4042337>

## 2.21 PL2303TA USB To TTL RS-232

เป็นโมดูลแปลงสัญญาณ USB เป็นพอร์ตอนุกรมระดับสัญญาณ TTL ใช้ไอซีเบอร์ PL2303TA ซึ่งแบ่งออกเป็น +5V, GND, TXD และ RXD



รูปที่ 2.20 PL2303TA USB To TTL RS-232

ที่มา : [https://th.sz-kuongshun.com/Content/File\\_img/S\\_Product/2016-08-](https://th.sz-kuongshun.com/Content/File_img/S_Product/2016-08-)

[25/201608251341376366588.jpg](https://th.sz-kuongshun.com/Content/File_img/S_Product/2016-08-25/201608251341376366588.jpg)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.22 EEPROM Flash BIOS รุ่น CH431A

เป็นอุปกรณ์ที่ใช้ในการอ่าน ดู หรือ แก้ไข Firmware จากชิป ROM โดยในปริณญา  
นิพนธ์นี้ได้ใช้ในการเข้าไปแก้ไขโค้ด Hex ในส่วนของการ interrupt ในหน้า console ของ OpenWrt



รูปที่ 2.21 EEPROM Flash BIOS รุ่น CH431A

ที่มา : <https://down->

[th.img.susercontent.com/file/100f44dc857ead1c226ea502042e9d8](https://down-th.img.susercontent.com/file/100f44dc857ead1c226ea502042e9d8)

## 2.23 Xiaomi Wi-Fi Range Extender AC1200

เป็นอุปกรณ์ที่ใช้ในการขยายสัญญาณ Wi-Fi ซึ่งในปริณญานิพนธ์นี้เราได้ทำการลง  
Firmware OpenWrt เพื่อให้มีความสามารถในการรันโปรแกรม Python

โดย OpenWrt คือ การกระจาย GNU / Linux ขึ้นอยู่กับ Debian GNU / Linux ที่  
ช่วยให้เราสามารถใช้ประโยชน์จากคุณสมบัติทางเทคนิคของ Router ของเราให้ได้มากที่สุด  
นอกเหนือจากการใช้งานที่เราสามารถมอบให้กับเฟิร์มแวร์ที่ผู้ผลิตติดตั้งไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 Xiaomi Wi-Fi Range Extender AC1200

ที่มา : <https://inwfile.com/s-fn/c173n5.jpg>

ตารางที่ 2.16 Xiaomi Wi-Fi Range Extender AC1200 Specification

Standard	IEEE 802.11ac/n/a 5 GHz, IEEE 802.11n/b/g 2.4 GHz
Speed Wi-Fi	5 GHz: 867 Mbps (802.11ac, 2*2 @80MHz)
	2.4 GHz: 300 Mbps (802.11n, 2*2 @40MHz)
Mode	Range Extender
	Access Point
Wireless security	WPA-PSK / WPA2-PSK
Signal strength	<20 dBm (2.4GHz), <23 dBm (5GHz)
RAM	64 MB
Port	1* Ethernet port 10/100 Mbps
Input	100~240V, 50/60Hz, 0.3A
Dimensions	84*100*82 mm

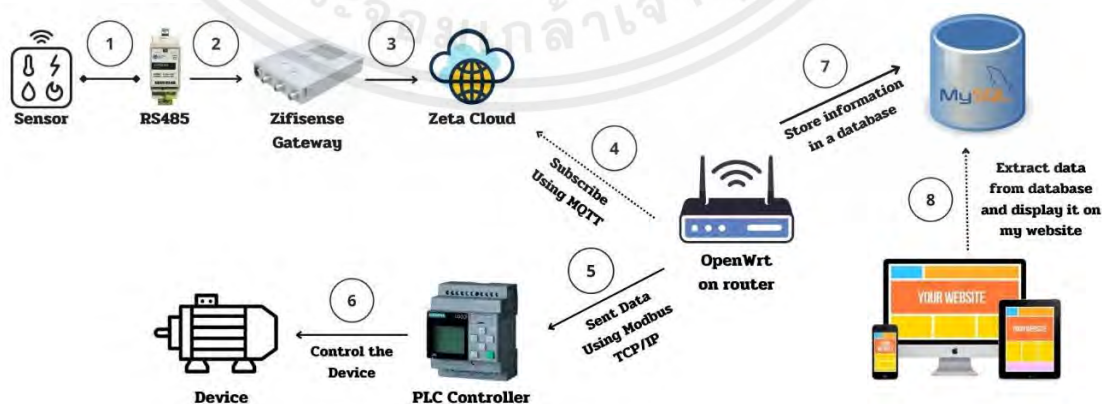
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบและการจัดทำปริญญานิพนธ์

#### 3.1 การออกแบบ

ทำการออกแบบระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าจากการใช้งาน PLC Controller ร่วมกับระบบ IoT โดยมีการใช้งาน Sensor ร่วมกับ RS-485 เพื่อส่งข้อมูลผ่าน AP Gateway ขึ้นสู่ระบบ Zeta Cloud หลังจากนั้นจะใช้ MQTT Protocol บน Router ที่ติดตั้ง Firmware OpenWrt ในการ Subscribe ข้อมูลและทำการแปลงบิตข้อมูลที่ได้รับเป็นตัวเลขฐานสิบแล้วจึงนำมาทำการประมวลผลเพื่อสร้างเงื่อนไขในการควบคุมการเปิด-ปิดอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller ซึ่งจะมีการบันทึกค่าข้อมูลไว้บน Database ของ MySQL Server หลังจากนั้นจึงนำข้อมูลต่างๆมาแสดงผลบน Website ที่ได้ทำการสร้างขึ้น โดยจะทำการเชื่อมต่อ PLC Controller กับ Router ผ่านระบบ LAN ซึ่ง PLC Controller จะใช้งาน Modbus TCP/IP ในการอ่านค่าหรือรับค่าจาก Router และนำข้อมูลที่ได้อ่านเก็บไว้ที่ตัว PLC Controller แล้วจึงทำการออกแบบโปรแกรมบน LOGO! 8.3 Software ในการสร้างการแสดงผลข้อมูลที่ได้และสถานะการทำงานของอุปกรณ์ Output บนหน้าจอ PLC Controller และ ควบคุมอุปกรณ์ Output ที่เชื่อมต่ออยู่ และจะทำการส่งข้อความแจ้งเตือนผ่านทาง LINE Notify เมื่ออุปกรณ์ Output ที่เชื่อมต่ออยู่ถูกสั่งให้ทำงาน ซึ่งจะทำให้การแสดงผลการทำงานร่วมกันของระบบทั้งหมดโดยทำการจำลองเป็นตู้สำหรับปลูกต้นไม้ที่เรียบง่าย



รูปที่ 3.1 บล็อกไดอะแกรมแสดงภาพรวมการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.1 การออกแบบการติดตั้งอุปกรณ์สำหรับการทดสอบเบื้องต้น

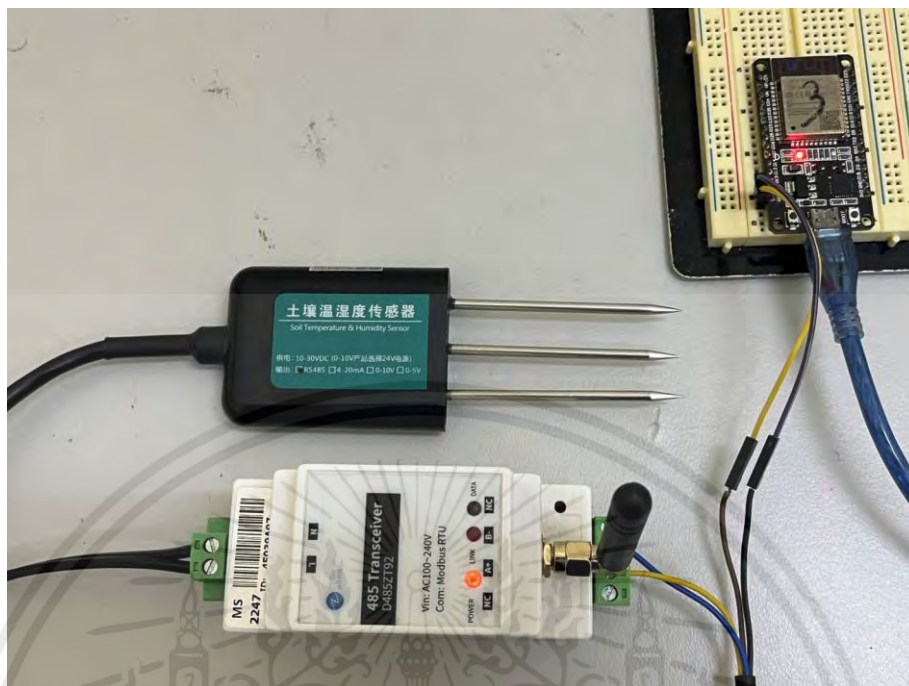
ในส่วนแรกจะเป็นการติดตั้ง AP Gateway โดยทำการนำ AP Gateway และ Sensor ไปจัดวางไว้ในพื้นที่ที่เหมาะสม คือ พื้นที่ที่โปร่งหรือที่ไม่ค่อยมีสิ่งกีดขวาง เพื่อให้ AP Gateway และ Sensor สามารถเชื่อมต่อกันแบบไร้สายได้ง่ายและมีประสิทธิภาพการเชื่อมต่อที่เสถียร จากนั้นทำการติดตั้ง AP Gateway โดยทำการ ใส่ SIM Card สำหรับใช้งาน internet จากนั้นต่อตัวรับสัญญาณต่างๆ ได้แก่ 1.GPIS สำหรับการเชื่อมต่อ internet 2.GPS สำหรับการระบุตำแหน่ง 3.RF antenna สำหรับการรับและส่งสัญญาณระหว่าง AP Gateway และ sensor จากนั้นทำการต่อเข้ากับ Universal Adaptor และต่อเข้ากับไฟบ้าน 220 V โดยมีการติดตั้ง AP Gateway แสดงดังรูปที่ 3.2

ในส่วนที่สองจะทำการติดตั้ง RS485 เข้ากับ Soil Temperature and Humidity Sensor ซึ่งใช้ในการวัดค่าอุณหภูมิและความชื้นในดิน และทำการติดตั้ง PLC Controller โดยอาศัยไฟบ้าน 220 V เป็นไฟเลี้ยง และทำการเชื่อมต่อ PLC Controller เข้ากับ Raspberry Pi ผ่านระบบ LAN โดยการออกแบบนี้ใช้สำหรับการทดสอบการทำงานเบื้องต้นของระบบก่อนที่จะใช้งาน OpenWrt Router แทน Raspberry Pi ใช้งาน Sensor ตัวอื่นๆเพิ่มเติม และนำอุปกรณ์ที่ใช้ไฟฟ้าที่ต้องการควบคุมมาเป็น Output ที่เชื่อมเข้ากับ PLC Controller



รูปที่ 3.2 ติดตั้ง AP Gateway

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ติดตั้ง RS485 และ Temperature & Humidity Sensor



รูปที่ 3.4 ติดตั้ง PLC Controller และเชื่อมต่อกับ Raspberry Pi

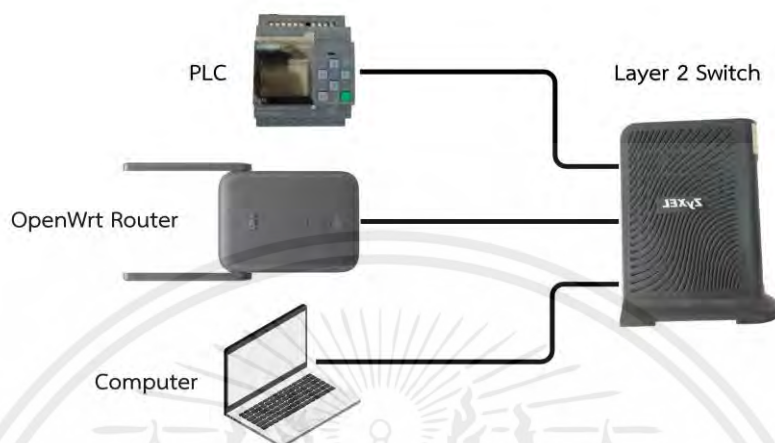
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.2 การออกแบบการติดตั้งอุปกรณ์ของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

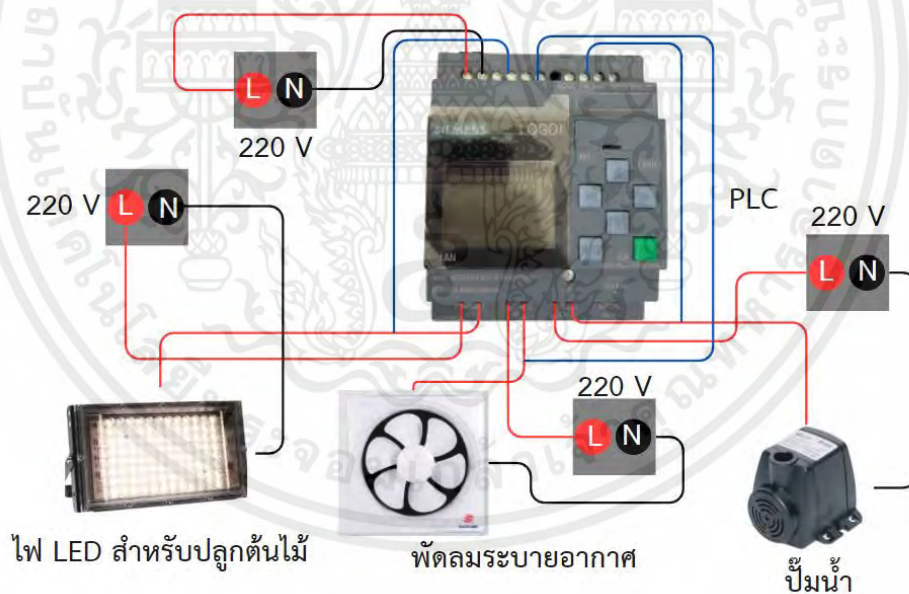
ในส่วนแรกจะเป็นการติดตั้ง AP Gateway ซึ่งจะเหมือนกับการติดตั้ง AP Gateway ในข้อที่ 3.1.1 โดยทำการนำ AP Gateway และ Sensor ไปจัดวางไว้ในพื้นที่ที่เหมาะสม คือ พื้นที่ที่โปร่งหรือที่ไม่ค่อยมีสิ่งกีดขวาง เพื่อให้ AP Gateway และ Sensor สามารถเชื่อมต่อกันแบบไร้สายได้ง่ายและมีประสิทธิภาพการเชื่อมต่อที่เสถียร จากนั้นทำการติดตั้ง AP Gateway โดยทำการใส่ SIM Card สำหรับใช้งาน internet จากนั้นต่อตัวรับสัญญาณต่างๆ ได้แก่ 1.GPIS สำหรับการเชื่อมต่อ internet 2.GPS สำหรับการระบุตำแหน่ง 3.RF antenna สำหรับการรับและส่งสัญญาณระหว่าง AP Gateway และ sensor จากนั้นทำการต่อเข้ากับ Universal Adaptor และต่อเข้ากับไฟบ้าน 220 V โดยมีการติดตั้ง AP Gateway แสดงดังรูปที่ 3.2

ในส่วนที่สองจะทำการติดตั้ง RS485 โดยอาศัยไฟบ้าน 220 V เป็นไฟเลี้ยง และทำการเชื่อมต่อเข้ากับ Sensor 2 ตัว คือ Soil Temperature and Humidity Sensor สำหรับวัดค่าอุณหภูมิและความชื้นในดิน และ Light Temperature and Humidity Sensor สำหรับวัดค่าความเข้มแสง อุณหภูมิในอากาศ และความชื้นในอากาศ โดย Sensor ทั้งสองจะอาศัยไฟเลี้ยง 5 V จากถ่าน

ในส่วนสุดท้ายคือการติดตั้ง PLC Controller โดยอาศัยไฟบ้าน 220 V เป็นไฟเลี้ยง จากนั้นทำการเชื่อมต่อ คอมพิวเตอร์ PLC Controller และ OpenWrt Router เข้ากับ Layer 2 Switch เพื่อให้อุปกรณ์ทั้งหมดสามารถเชื่อมต่อและสื่อสารกันได้ผ่านระบบ LAN มีลักษณะการเชื่อมต่อแสดงดังรูปที่ 3.5 และในส่วน Output ของ PLC Controller จะทำการเชื่อมต่ออยู่กับอุปกรณ์ที่ใช้ไฟฟ้าได้แก่ ไฟ LED สำหรับปลูกต้นไม้ ป้อนน้ำ และพัดลมระบายอากาศ ซึ่งอุปกรณ์ Output จะอาศัยไฟบ้าน 220 V เป็นไฟเลี้ยง ทำการเชื่อมต่อสายไฟในลักษณะที่มี PLC Controller เป็น Relay Switch คอยควบคุมการทำงานโดยทำการต่อสาย Line จากแหล่งจ่ายไฟเข้ากับช่อง Output ทางด้านซ้าย ของ PLC Controller และเชื่อมต่อสายไฟจากช่อง Output ทางด้านขวาซึ่งอยู่ด้านหลัง Relay Switch เข้ากับอุปกรณ์ Output ส่วนสาย Neutral จะเชื่อมต่อเข้ากับอุปกรณ์ Output โดยตรง และจะมีการเชื่อมต่อสาย Line ที่อยู่ด้านหลัง Relay Switch วนกลับเข้าไปที่ช่อง Input ของ PLC Controller เพื่อเป็นการตรวจสอบสถานะการทำงานของอุปกรณ์ Output ว่ามีการทำงานตรงกับที่ PLC Controller ได้ควบคุมหรือไม่ โดยมีลักษณะการเชื่อมต่อแสดงดังรูปที่ 3.6



รูปที่ 3.5 การเชื่อมต่ออุปกรณ์เข้าด้วยกันผ่าน Layer 2 Switch

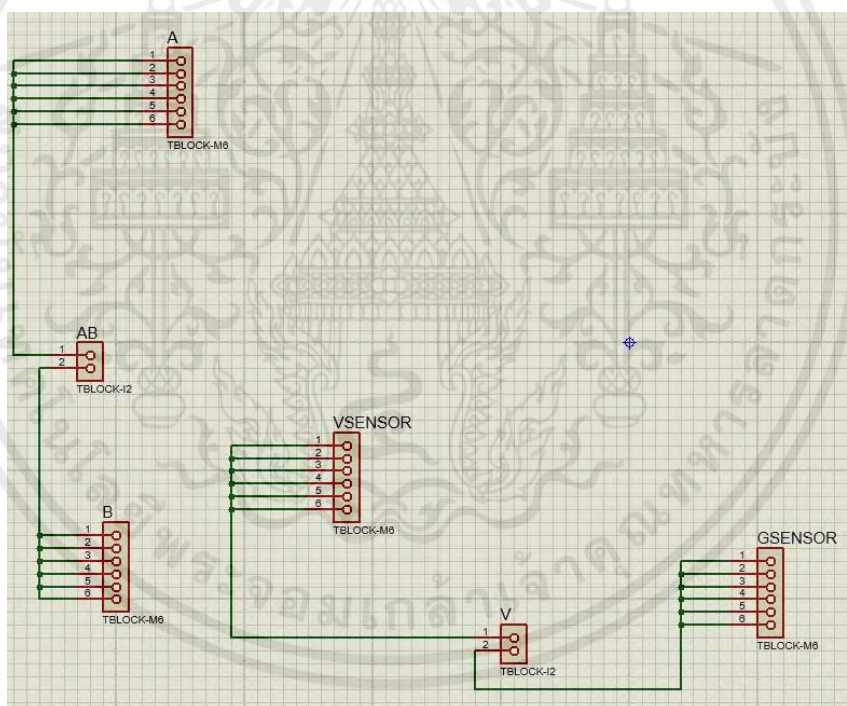


รูปที่ 3.6 การเชื่อมต่ออุปกรณ์ Output กับ PLC Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

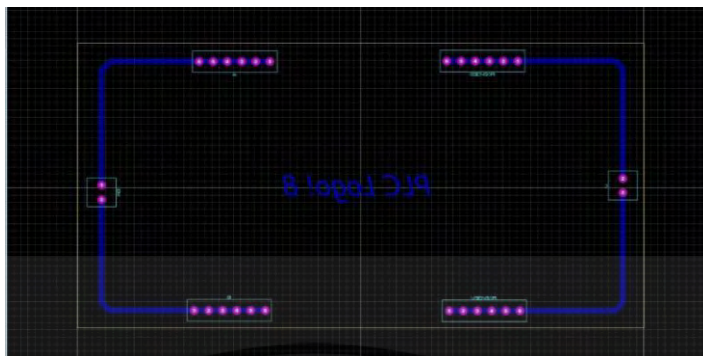
### 3.1.3 การออกแบบการเชื่อมต่อ RS485 เข้ากับ Sensor มากกว่า 1 ตัว

ในปฏิญานิพนธ์นี้จะใช้ Sensor 2 ตัวเชื่อมต่อกับ RS485 ได้แก่ Soil Temperature and Humidity Sensor และ Light Temperature and Humidity Sensor ซึ่งในการใช้งาน RS485 ของ Zifisense ร่วมกับ Sensor ที่มากกว่า 1 ตัวจะต้องทำการเชื่อมต่อแบบอนุกรมโดยต่อสาย A ของ RS485 เข้ากับสาย A ของ Sensor ทั้งหมดที่ต้องการใช้งาน และทำการเชื่อมต่อสาย B ของ RS485 เข้ากับ สาย B ของ Sensor ทั้งหมดที่ต้องการใช้งาน หลังจากนั้นต้องทำการตั้งค่า Parameter ต่างๆของ Sensor ให้ตรงกันและทำการตั้งเวลาในการสลับกันอ่านข้อมูลจาก Sensor แต่ละตัว ในส่วนของการออกแบบจะใช้โปรแกรม Proteus ในการออกแบบวงจรที่ใช้สำหรับจ่ายไฟและเชื่อมต่อ RS485 เข้ากับ Sensor และทำให้ Sensor ทั้งสองเชื่อมต่อกันแบบอนุกรม โดยจะมีลักษณะการเชื่อมต่อกันของอุปกรณ์แสดงดังรูปที่ 3.10



รูปที่ 3.7 การออกแบบวงจรสำหรับเชื่อมต่อ Sensor

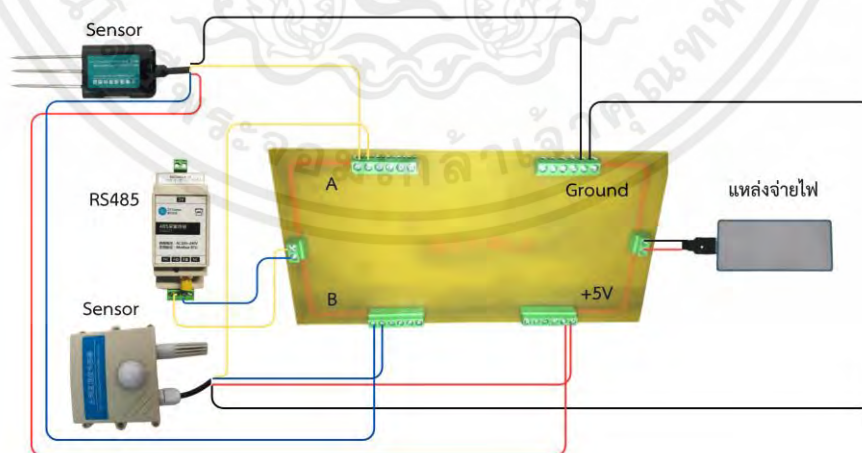
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 การออกแบบวงจรสำหรับเชื่อมต่อ Sensor



รูปที่ 3.9 วงจรสำหรับเชื่อมต่อ Sensor



รูปที่ 3.10 การเชื่อมต่อระหว่าง RS485 Sensor ทั้ง 2 ตัว และแหล่งจ่ายไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการเชื่อมต่อ Sensor ทั้งสองแบบอนุกรมเข้ากับ RS485 และแหล่งจ่ายไฟสำหรับ Sensor จากนั้น ต้องทำการตั้งค่า Parameter ต่างๆให้กับ Sensor ได้แก่ Baud Rate และ Slave ID ของ Sensor เพื่อให้ RS485 สามารถใช้ระบุได้ว่าต้องการข้อมูลจาก Sensor ตัวไหนผ่าน Slave ID หลังจากทำการเชื่อมต่อ RS485 to USB เข้ากับ Soil Temperature and Humidity Sensor และ Light Temperature and Humidity Sensor แล้ว ต้องทำการเชื่อมต่อโดยกำหนด Baud Rate และ Slave ID ของทั้ง 2 Sensor ให้ถูกต้อง ในที่นี้ทั้ง 2 Sensor ค่าเริ่มต้นจะมี Slave ID เป็น 1 และ Baud Rate เป็น 9600 โดยจะทำการแก้ไขให้ Light Temperature and Humidity Sensor มี Slave ID เป็น 2 และ Baud Rate 9600 เพื่อให้ Master สามารถเลือก Slave ID ที่ต้องการอ่านข้อมูลได้

ทำการกำหนด Parameter ต่างๆของ RS485 เพื่อให้สามารถอ่านข้อมูลจากทั้ง 2 Sensor ได้ โดยทำการตั้งค่า Baud rate เป็น 9600, Data bit เป็น 8 และ Stop bit เป็น 1 และการตั้งค่าการเพิ่ม Address Code เพื่อให้สามารถรับค่าจาก 2 Slave ID ได้ จากนั้นตั้งค่า Acquisition parameter สำหรับการอ่านค่าว่าต้องการอ่านค่าข้อมูลที่อยู่ใน Register Address ไตบ้าง การอ่านค่าจากทั้ง 2 Sensor RS485 ของ Zifisense ต้องทำการตั้งค่า Backoff เพื่อเป็นการกำหนดให้อ่านค่าข้อมูลจาก Address 1 ก่อน แล้วอีกเป็นระยะเวลาสั้นเท่าไรจึงจะอ่านค่าจาก Address 2 ซึ่งในที่นี้ตั้งค่าให้อ่านข้อมูลจาก Address ที่ 2 หลังจากผ่านไป 5 วินาที

หลังจากทำการเชื่อมต่อ RS485 เข้ากับ Soil Temperature and Humidity Sensor และ Light Temperature and Humidity Sensor แล้วทำการตั้งค่า Parameter ต่างๆให้ถูกต้อง RS485 ก็จะสามารถส่งข้อมูลที่วัดได้จาก Sensor ทั้งสองตัวไปยัง Zeta Cloud โดยมีรูปแบบแสดงดังรูปที่ 3.12 ซึ่งข้อมูลจะถูกสลับกันส่งตามเวลาที่ได้ตั้งไว้ ซึ่ง Sensor 1 คือ Soil Temperature and Humidity Sensor ซึ่งใช้สำหรับการวัดค่าอุณหภูมิและความชื้นในดิน ส่วน Sensor 2 คือ Light Temperature and Humidity Sensor ที่ใช้สำหรับวัดค่าความเข้มแสง และวัดอุณหภูมิและความชื้นในอากาศ

รูปที่ 3.11 การตั้งค่า Parameter ต่างๆให้กับ Sensor

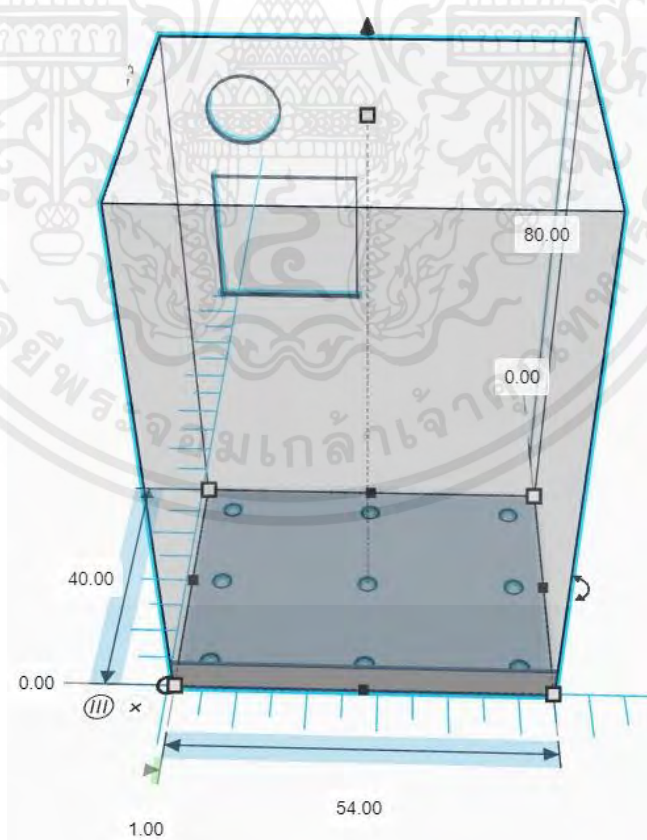
Raw data	Data content	Report time
010201c700eb002e	Air sensor:2 Humid:45.5%RH Temp:23.5°C Light:4.6lux	2023-12-08 17:06:43
010100000e40000	Soil sensor:1 Humid:0.0%RH Temp:22.8°C Light:0.0lux	2023-12-08 17:06:37
010201b200ea002b	Air sensor:2 Humid:43.4%RH Temp:23.4°C Light:4.3lux	2023-12-08 17:05:43
010100000e300e3	Soil sensor:1 Humid:0.0%RH Temp:22.7°C Light:22.7lux	2023-12-08 17:05:37

รูปที่ 3.12 รูปแบบข้อมูลที่รับได้จาก Sensor ทั้งสองตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.4 การออกแบบตู้สำหรับปลูกต้นไม้ในที่ร่มอย่างง่าย

ทำการออกแบบตู้สำหรับปลูกต้นไม้ในที่ร่มอย่างง่ายเพื่อนำมาเป็นตัวอย่งในการแสดงการทำงานร่วมกันของระบบ โดยเลือกวัสดุที่ใช้ในการทำตู้เป็นอะคริลิก ซึ่งจะมีการปลูกต้นไม้ไว้ในตู้บริเวณด้านล่าง มีอุปกรณ์ที่นำมาประกอบเข้ากับตู้ได้แก่ พัดลมสำหรับระบายอากาศ ไฟ LED สำหรับปลูกต้นไม้ ป้อนน้ำสำหรับรดน้ำต้นไม้ Sensor สำหรับวัดค่าอุณหภูมิ ความชื้น และความเข้มแสงในอากาศ Sensor สำหรับวัดค่าอุณหภูมิ และความชื้นในดิน และ PLC Controller มีการออกแบบตู้โดยแบ่งออกเป็น 6 แผ่นและนำมาประกอบกันภายหลังด้วยน้ำยาเชื่อมอะคริลิก แผ่นที่ 1 เป็นหลังคาด้านบนสุด มีขนาด 50 x 40 ซม. แผ่นที่สองเป็นฐานด้านล่างสุดมีขนาด 50 x 40 ซม. เช่นเดียวกันแต่จะมีการเจาะรูสำหรับระบายน้ำที่มีเส้นผ่านศูนย์กลางขนาด 3 ซม. ไว้จำนวน 9 รู แผ่นที่ 3 และแผ่นที่ 4 คือด้านซ้ายและด้านขวามีขนาดเท่ากันคือ 40 x 80 ซม. แผ่นที่ 5 คือด้านหน้ามีขนาด 54 x 80 ซม. และแผ่นที่ 6 คือด้านหลังมีขนาด 54 x 80 ซม. เช่นกันแต่จะมีการเจาะรูวงกลมที่มีเส้นผ่านศูนย์กลางขนาด 10 ซม. สำหรับโยงสายไฟ และสี่เหลี่ยมจัตุรัสขนาด 40 ซม. สำหรับติดตั้งพัดลมระบายอากาศ มีการออกแบบแสดงดังรูปที่ 3.13 3.14 และ 3.15



รูปที่ 3.13 ภาพด้านหน้าของตู้สำหรับปลูกต้นไม้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 ภาพด้านหลังของตู้สำหรับปลูกต้นไม้

รูปที่ 3.15 ภาพด้านข้างของตู้สำหรับปลูกต้นไม้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.5 การออกแบบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud

ทำการเชื่อมต่อ Soil Temperature and Liity Sensor เข้ากับ RS485 จากนั้นทำการตั้งค่า Parameter ต่างๆของ RS485 ได้แก่

1. Parameter reporting cycle(min) คือ เวลาในการส่งข้อมูลแต่ละรอบขึ้น Zeta Cloud โดยมีหน่วยเป็นนาที (minute) แสดงดังรูปที่ 3.16

2. Address Code คือ การกำหนด Address Code และ Meter name ซึ่ง Address Code จะต้องตั้งค่าให้ตรงกับ Address ของ Sensor ซึ่งในที่นี้มีค่าเท่ากับ 1 และสามารถกำหนดชื่อที่เราต้องการเพื่อให้เข้าใจว่า Address นี้คืออะไรในส่วนของ Meter name แสดงดังรูปที่ 3.17

3. Acquisition parameter คือ การกำหนดค่าที่เราต้องการจะอ่าน โดยกำหนดพารามิเตอร์ที่สำคัญ ดังนี้

- Parameter convert คือ ประเภทข้อมูลที่เรากำลังจะอ่านโดยแบ่งออกเป็น Byte และ Bit

- Register คือ ตำแหน่งเริ่มต้นของข้อมูลที่เรากำลังจะอ่านโดยจะต้องทำการตั้งค่าให้ตรงกับ Register address ของค่าข้อมูลใน Sensor

- Number of registers read คือ จำนวนของ Register ที่เรากำลังจะอ่าน

- Number of คือ จำนวนข้อมูลของ Register ที่เรากำลังจะอ่านซึ่งมีขนาดเป็น Byte

- Parameter คือ การกำหนดชื่อค่าข้อมูลของ Register นั้นๆ

- Data Type คือ ประเภทของข้อมูลที่เรากำลังจะแปลง

- Unit คือ หน่วยของค่าข้อมูลซึ่งจะมีการกำหนดค่าที่ต้องการมาคูณเพื่อให้ได้ค่าข้อมูลที่ถูกต้อง โดยแสดงดังรูปที่ 3.18

4. Serial port parameter คือ การกำหนด Protocol Modbus RTU ในการรับ-ส่งข้อมูลซึ่งจะต้องมีการกำหนดพารามิเตอร์ที่สำคัญ ดังนี้

- Baud Rate คือ อัตราเร็วของข้อมูลในการรับ-ส่ง

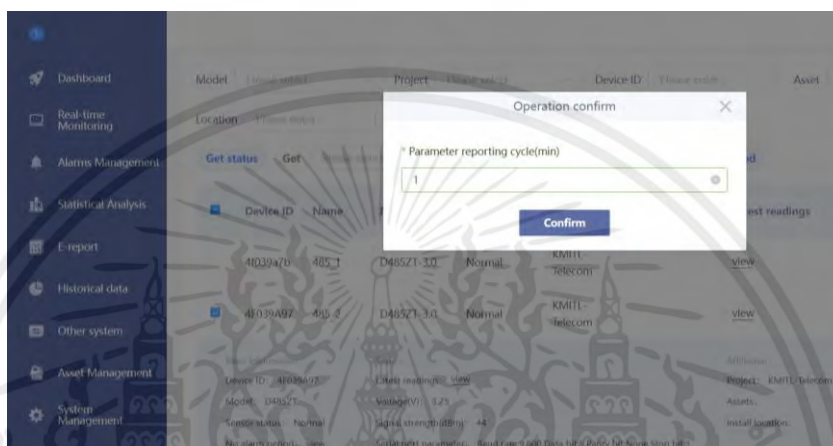
- Data Bit คือ ขนาดความกว้างของบิตข้อมูล

- Check Digit คือ การกำหนดว่าข้อมูลนั้นจะมีการตรวจสอบ Parity Bit หรือไม่

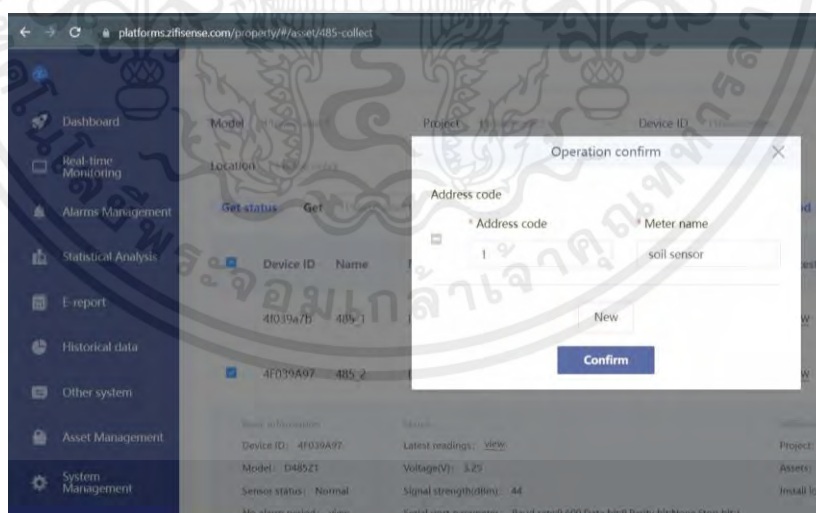
- Stop Bit คือ บิตสิ้นสุดของข้อมูล โดยแสดงดังรูปที่ 3.19

5. Register function code คือ การกำหนด Function Code ที่เรากำลังจะใช้ ซึ่งในที่นี้เราทำการกำหนดเป็น Function Code 03 ไว้สำหรับการอ่านข้อมูลเพียงอย่างเดียว แสดงดังรูปที่ 3.20

เมื่อ Sensor ทำการวัดค่าอุณหภูมิและความชื้นได้ RS485 จะทำการอ่านข้อมูลที่ Sensor วัดค่าได้แล้วทำการส่งข้อมูลไปยัง AP Gateway ซึ่ง AP Gateway จะทำการส่งข้อมูลไปสู่ Zeta Cloud โดยแผนผังขั้นตอนในการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud สามารถแสดงได้ดัง รูปที่ 3.21

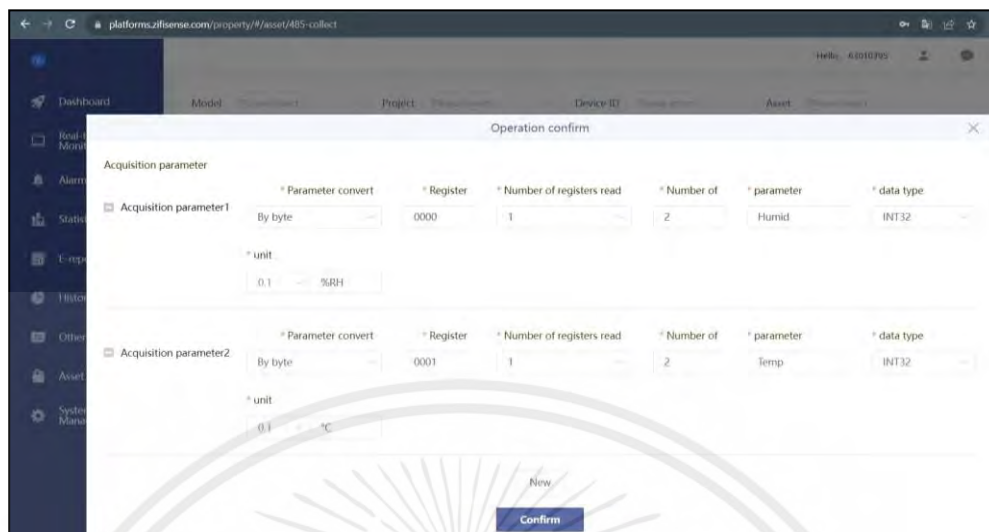


รูปที่ 3.16 การตั้งค่าเวลาในการส่งข้อมูลของ RS485

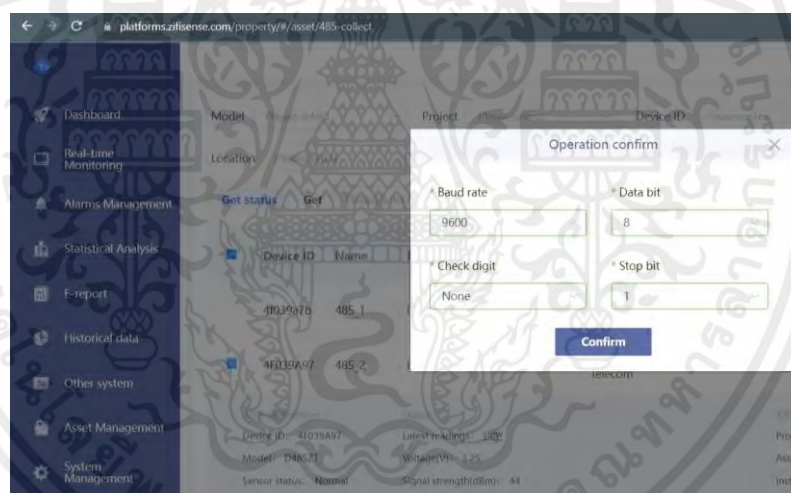


รูปที่ 3.17 การตั้งค่า Address Code และ Meter Name

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

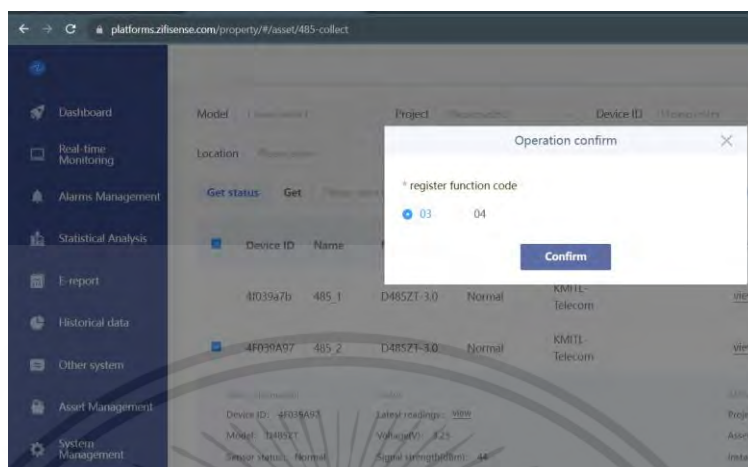


รูปที่ 3.18 การตั้งค่า Acquisition Parameter

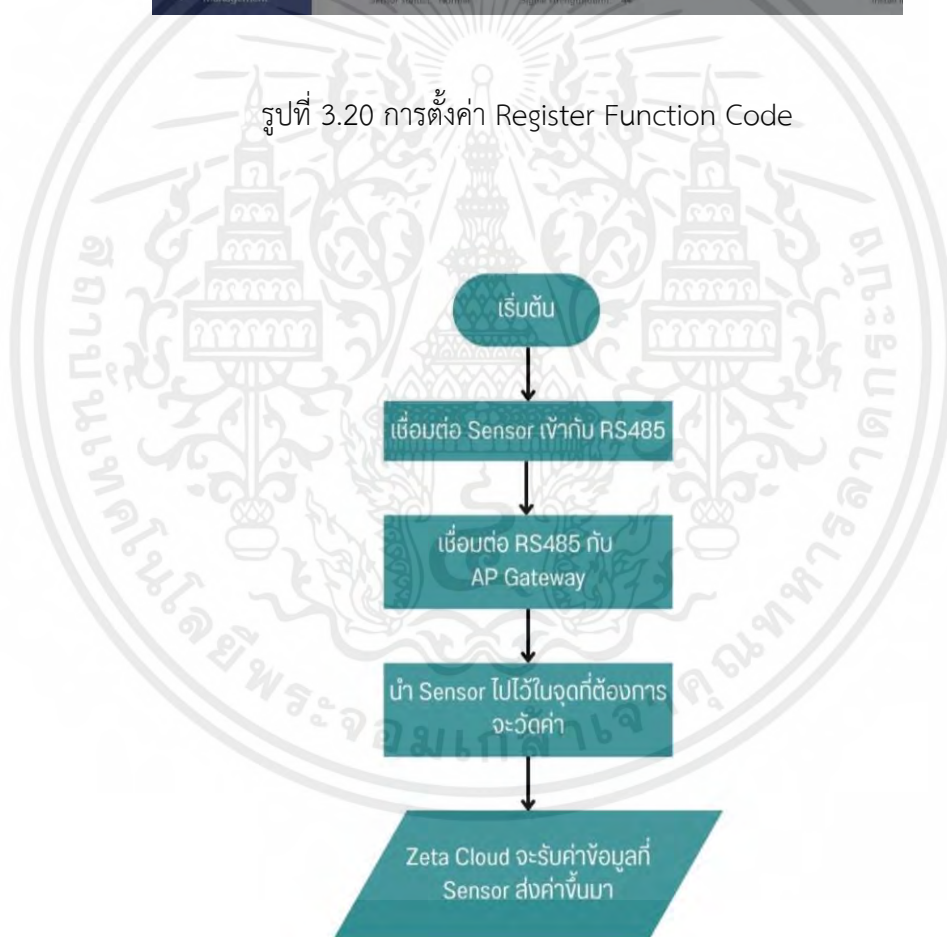


รูปที่ 3.19 การตั้งค่า Baud rate, Data bit, Check digit และ Stop bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 การตั้งค่า Register Function Code



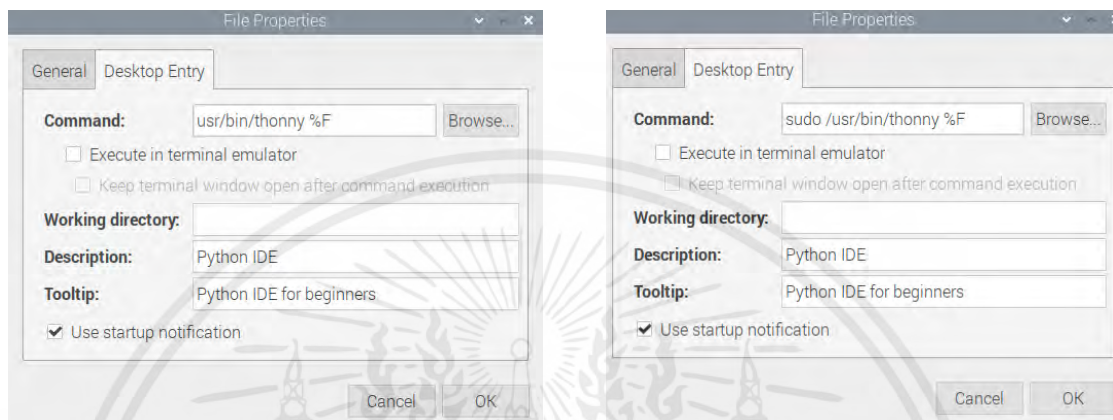
รูปที่ 3.21 แผนผังแสดงการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.6 การออกแบบการรับและส่งข้อมูลของ Raspberry Pi

ในการทดสอบการทำงานของระบบเบื้องต้นจะใช้งาน Raspberry Pi ในการรับและส่งข้อมูลก่อนที่จะทำการใช้งาน OpenWrt Router แทนในการใช้งานจริง ในลำดับแรกจะทำการตั้งค่าให้ Raspberry Pi อยู่ใน Root mode โดยพิมพ์คำสั่ง sudo เพิ่มที่ด้านหน้าของช่อง Command เพื่อให้สามารถใช้งาน Reserved ports 1 – 1023 ที่ต้องใช้สิทธิ์ Root ในการใช้งาน แสดงดังรูปที่ 3.22 หลังจากนั้นทำการตั้ง IP ของ Raspberry Pi ให้เป็น 192.168.100.105 ซึ่งอยู่ใน Subnet เดียวกันกับ PLC Controller แสดงดังรูปที่ 3.23 ในการทดสอบเบื้องต้นจะใช้ Soil Temperature and Humidity Sensor 1 ตัวเชื่อมต่อเข้ากับ RS485 โดย Raspberry Pi จะทำการรับข้อมูลจาก Sensor ผ่านการ Subscribe จาก Zeta Cloud ซึ่งการ Subscribe ข้อมูลจะใช้หลักการของ โพรโตคอล MQTT (Message Queuing Telemetry Transport) จากนั้นจะทำการแปลงข้อมูลให้อยู่ในรูปฐาน 10 แล้วทำการส่งบิตข้อมูลพร้อมกับส่งค่า Coil 0 หรือ 1 ตามเงื่อนไขที่ได้กำหนดไว้ให้กับ PLC Controller ผ่านโพรโตคอล Modbus TCP/IP ผ่านการใช้ Python Script ซึ่ง PLC Controller จะนำข้อมูลที่ได้ไปทำการแสดงผลบนหน้าจอและทำการควบคุมอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller

ในอีกส่วนหนึ่ง Raspberry Pi จะส่งข้อมูลที่อยู่ในรูปแบบ JSON String ไปเก็บไว้ที่ฐานข้อมูลที่สร้างโดย MySQL ซึ่งเป็น Software ที่ใช้ SQL เป็นภาษาข้อมูลพื้นฐานและจัดเก็บข้อมูลในตารางบนดิสก์ไดรฟ์ของเซิร์ฟเวอร์ โดยได้มีการออกแบบ Table ที่ใช้ในการเก็บข้อมูลไว้ใน Database ของข้อมูลที่ได้จากแต่ละ Sensor โดยจะมีการเก็บค่า CompanyCode , DeviceType , Data , Temp , Humid , Info , Datetime และ Status จะได้จากการ Subscribe ของโพรโตคอล MQTT และโดยค่า Data เป็นข้อมูลเลขฐาน 16 มาแปลงเป็นข้อมูลเลขฐาน 10 เพื่อจัดเก็บไว้ใน Info และส่วนของ Uptime คือ วันที่และเวลาที่ข้อมูลนั้นถูกจัดเก็บใน Database ซึ่งข้อมูลจะถูกนำมาแสดงบน Website และแสดงส่วนประกอบต่างๆของปริญญานิพนธ์ รวมถึงแสดงกราฟการเปลี่ยนแปลงของข้อมูลแต่ละ Sensor โดยทำการเขียนโปรแกรมภาษา PHP , HTML , CSS และ JavaScript ในการออกแบบหน้าต่าง Website เพื่อให้ง่ายต่อผู้คนที่จะมาใช้งาน โดยได้นำ Website อัพขึ้นไปเป็น Public และเชื่อมต่อกับ Domain Name ชื่อว่า www.apacontrol.site โดยขั้นตอนในการรับและส่งข้อมูลของ Raspberry Pi สามารถแสดงได้ดังรูปที่ 3.24



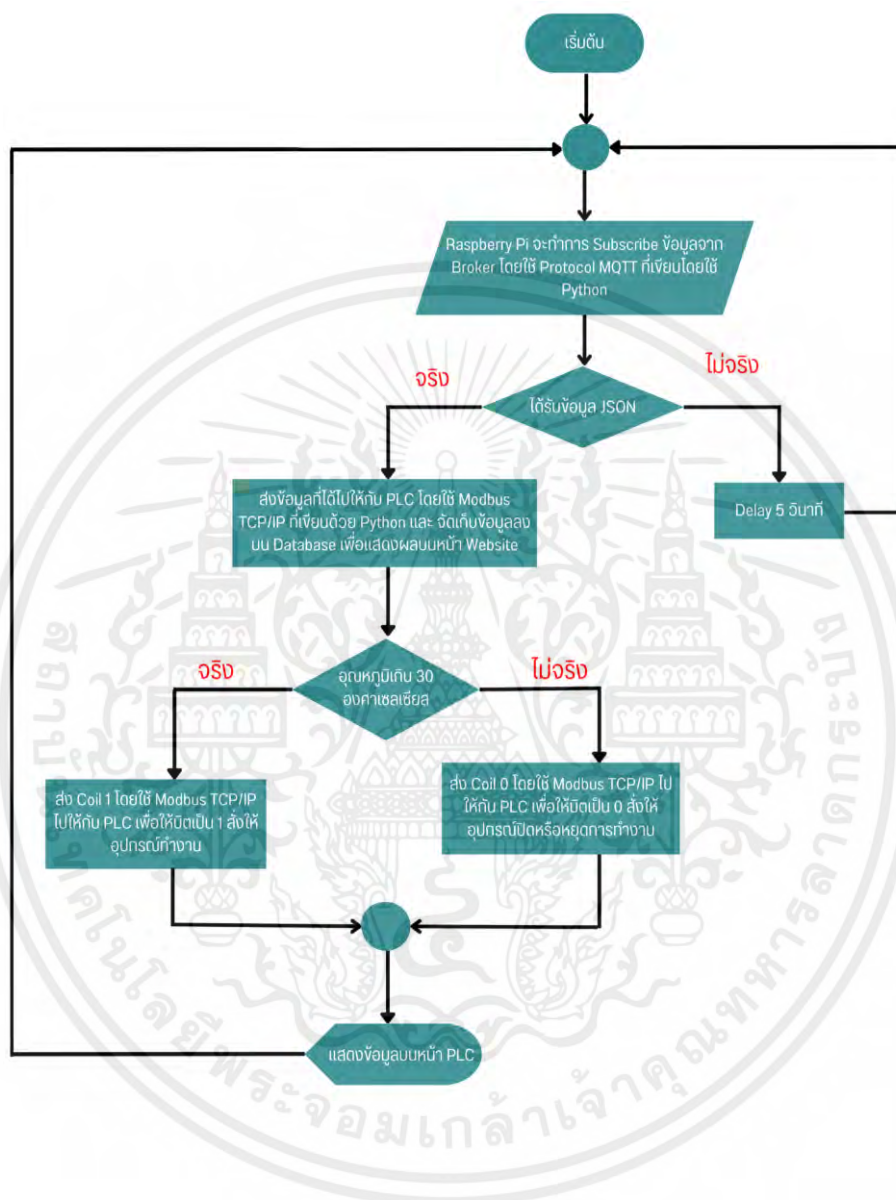
รูปที่ 3.22 การตั้งค่า Raspberry Pi ให้อยู่ใน Root โหมด

```

pi@raspberrypi:~ $ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.105 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::ff7d:6708:a18c:84ad prefixlen 64 scopeid 0x20<link>
    ether e4:5f:01:0c:46:d4 txqueuelen 1000 (Ethernet)
    RX packets 3467 bytes 208236 (203.3 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 360 bytes 32343 (31.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
  
```

รูปที่ 3.23 การตั้งค่า IP ของ Raspberry Pi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 แผนผังแสดงการรับและส่งข้อมูลของ Raspberry Pi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.7 การออกแบบการใช้งาน Firmware OpenWrt บน Raspberry Pi

ทำการติดตั้ง Firmware OpenWrt ลงบน Raspberry Pi เพื่อใช้ในการทดสอบการทำงาน ของ Firmware OpenWrt ก่อนจะนำไปใช้งานจริงบน Router จากนั้นทำการตั้งค่าต่างๆ เพื่อให้สามารถใช้งาน Internet ได้

1) เมื่อทำการติดตั้ง Firmware OpenWrt ลงบน Raspberry Pi เรียบร้อยแล้ว Raspberry Pi จะทำงานเสมือนเป็น OpenWrt Router หลังจากนั้นจะทำการลงภาษา Python 3 ด้วยคำสั่ง `opkg install python3` แสดงดังรูปที่ 3.25

```
root@OpenWrt:~# opkg install python3
Package python3 (3.9.16-1) installed in root is up to date.
```

รูปที่ 3.25 การติดตั้ง ภาษา Python 3

2) ทำการติดตั้ง Library ต่างๆ ได้แก่ 1. mysql connector python ซึ่งใช้ในการส่งข้อมูลไปยัง Database 2. paho mqtt ซึ่งจะใช้สำหรับการ Subscribe ข้อมูลต่างๆที่วัดได้จาก Sensor 3. pymodbus, pyModbusTCP และ uModbus ซึ่งใช้ในการส่งข้อมูลจาก OpenWrt Router ไปยัง PLC Controller

```
root@OpenWrt:~# pip install pymodbus
Requirement already satisfied: pymodbus in /usr/lib/python3.9/site-packages (2.5.1)
Requirement already satisfied: six>=1.15.0 in /usr/lib/python3.9/site-packages (from pymodbus) (1.16.0)
Requirement already satisfied: pyserial>=3.4 in /usr/lib/python3.9/site-packages (from pymodbus) (3.5)
```

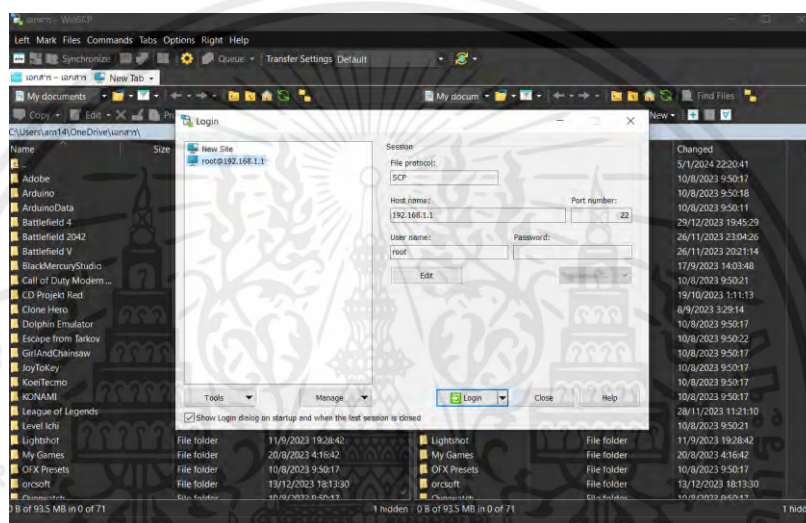
รูปที่ 3.26 ตัวอย่างการลง Library

Package	Version
mysql-connector-python	8.2.0
paho-mqtt	1.6.1
pip	23.3.2
protobuf	4.21.12
pymodbus	2.5.1
pyModbusTCP	0.2.1
pyserial	3.5
setuptools	58.1.0
six	1.16.0
sudo	1.0.0
uModbus	1.0.4

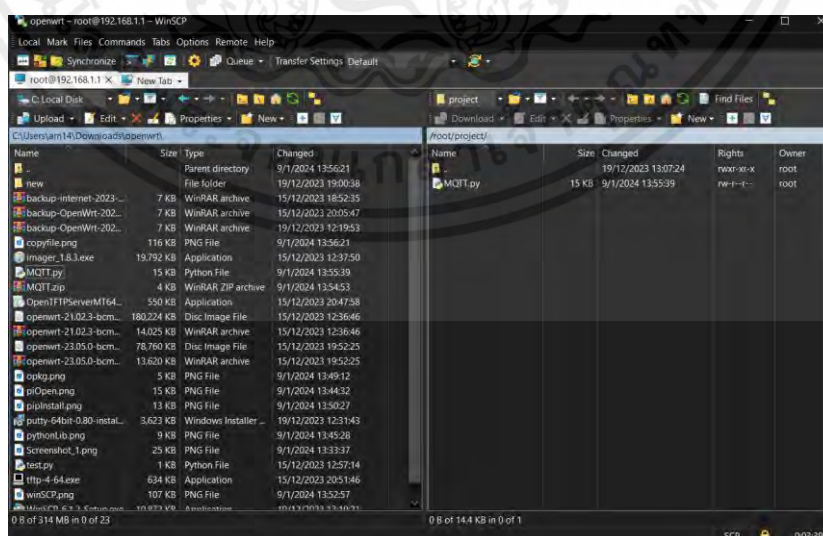
รูปที่ 3.27 Library ทั้งหมดที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ในส่วนของ การโอนย้ายไฟล์ Python Script จะใช้งานโปรแกรม WinSCP เพื่อใช้งานโปรโตคอล SCP ในการโอนย้ายไฟล์ Python Script ที่ได้ทำการเขียนไว้จากคอมพิวเตอร์ไปยัง OpenWrt Router โดยจะต้องทำการเลือกใช้งาน Port 22 เพื่อใช้งานโปรโตคอล SCP ทำการตั้งค่า IP ของ OpenWrt Router ในส่วนของ Host name และทำการใส่ Username และ Password ตามที่ได้ตั้งไว้ใน OpenWrt Router จากนั้นทำการคัดลอกไฟล์จากคอมพิวเตอร์ทางด้านซ้ายไปวางยัง OpenWrt Router ซึ่งอยู่ทางด้านขวา



รูปที่ 3.28 การเชื่อมต่อคอมพิวเตอร์กับ OpenWRT Router ด้วยโปรแกรม WinSCP



รูปที่ 3.29 ทำการโอนย้ายไฟล์ Python Script ไปยัง OpenWrt Router

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ในขั้นตอนนี้จะยังไม่สามารถรัน Python Script ได้เนื่องจากยังไม่ได้ทำการให้สิทธิในการรัน จะสามารถทำได้เพียงแค่การอ่านและเขียน Script แสดงดังรูปที่ 3.30 จึงต้องทำการใช้คำสั่ง `chmod +x` ตามด้วยชื่อไฟล์ เพื่อให้สิทธิในการรัน และสามารถใช้คำสั่ง `ls -l` ตามด้วยชื่อไฟล์ เพื่อตรวจสอบสถานะการอนุญาตของไฟล์แสดงดังรูปที่ 3.31 หลังจากนั้นจึงจะสามารถรัน Python Script ได้ตามปกติ

```
root@OpenWrt:~/project# ./MQTT.py
-ash: ./MQTT.py: Permission denied
```

รูปที่ 3.30 ทำการ Run Python Script

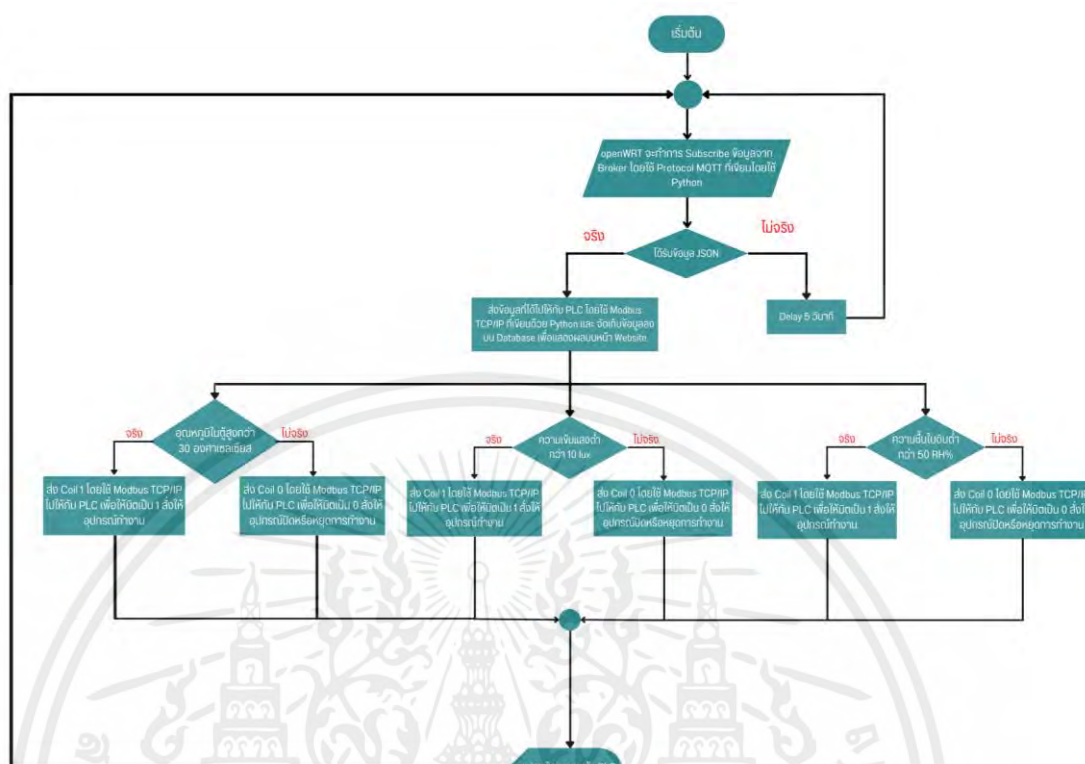
```
root@OpenWrt:~/project# ls -l MQTT.py
-rwxr-xr-x  1 root  root    14812 Jan  9  2024 MQTT.py
```

รูปที่ 3.31 การตรวจสอบสถานะการอนุญาตของไฟล์

### 3.1.8 การออกแบบการรับและส่งข้อมูลของ OpenWrt Router

การใช้งานจริงในระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าจะใช้งาน OpenWrt Router แทน Raspberry Pi เนื่องจากมีความเหมาะสมกับการใช้งานมากกว่า มีราคาต่ำกว่า เนื่องจากในปริณิญา นิพนธ์นี้ต้องการใช้อุปกรณ์ในการรัน Python Script สำหรับรับข้อมูลด้วยโปรโตคอล MQTT และทำการส่งข้อมูลด้วยโปรโตคอล Modbus TCP/IP ไปยัง PLC Controller ซึ่งสามารถใช้ OpenWrt Router ในการทำงานนี้ได้ โดยการออกแบบการรับและส่งข้อมูลของ OpenWrt Router จะเหมือนกับ Raspberry Pi โดย OpenWrt Router จะทำการรับข้อมูลจาก Sensor ผ่านการ Subscribe จาก Zeta Cloud ซึ่งการ Subscribe ข้อมูลจะใช้หลักการของ โปรโตคอล MQTT (Message Queuing Telemetry Transport) จากนั้นจะทำการแปลงข้อมูลให้อยู่ในรูปฐาน 10 แล้วทำการส่งบิตข้อมูลพร้อมกับส่งค่า Coil 0 หรือ 1 ตามเงื่อนไขที่ได้กำหนดไว้ให้กับ PLC Controller ผ่านโปรโตคอล Modbus TCP/IP ผ่านการใช้ Python Script ซึ่ง PLC Controller จะนำข้อมูลที่ได้ไปทำการแสดงผลบนหน้าจอและทำการควบคุมอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller

ในอีกส่วนหนึ่ง OpenWrt Router จะส่งข้อมูลที่อยู่ในรูปแบบ JSON String ไปเก็บไว้ในที่ ฐานข้อมูลที่สร้างโดย MySQL ซึ่งเป็น Software ที่ใช้ SQL เป็นภาษาข้อมูลพื้นฐานและจัดเก็บข้อมูล ในตารางบนดิสก์ไดรฟ์ของเซิร์ฟเวอร์ โดยได้มีการออกแบบ Table ที่ใช้ในการเก็บข้อมูลไว้ใน Database ของข้อมูลที่ได้จากแต่ละ Sensor โดยจะมีการเก็บค่า CompanyCode , DeviceType , Data , Temp , Humid , Info , Datetime และ Status จะได้จากการ Subscribe ของโปรโตคอล MQTT และโดยค่า Data เป็นข้อมูลเลขฐาน 16 มาแปลงเป็นข้อมูลเลขฐาน 10 เพื่อจัดเก็บไว้ใน Info และส่วนของ Uptime คือ วันที่และเวลาที่ข้อมูลนั้นถูกจัดเก็บใน Database ซึ่งข้อมูลจะถูกนำมา แสดงบน Website และแสดงส่วนประกอบต่างๆของปริณิญา นิพนธ์ รวมถึงแสดงกราฟการ เปลี่ยนแปลงของข้อมูลแต่ละ Sensor โดยทำการเขียนโปรแกรมภาษา PHP , HTML , CSS และ JavaScript ในการออกแบบหน้าต่าง Website เพื่อให้ง่ายต่อผู้คนที่จะมาใช้งาน โดยได้นำ Website อัปขึ้นไปเป็น Public และเชื่อมต่อกับ Domain Name ชื่อว่า ชื่อว่า www.apacontrol.site โดยขั้นตอนในการรับและส่งข้อมูลของ OpenWrt Router สามารถแสดงได้ดังรูปที่ 3.32



รูปที่ 3.32 แผนผังแสดงการรับและส่งข้อมูลของ OpenWrt Router

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบการใช้งาน OpenWrt Router ในการรับและส่งข้อมูลมีขั้นตอนดังนี้

- 1) ทำการเตรียมอุปกรณ์ EEPROM Flash BIOS รุ่น CH431A สำหรับใช้ในการ Flash SPI Flash Memory ของ Router โดยจะต้องทำการหนีบอุปกรณ์ EEPROM Flash BIOS เข้ากับ SPI Flash Memory ของ Router โดยสายสีแดงจะต้องต่อเข้ากับขาที่ 1 ของ SPI Flash Memory ของ Router ดังรูปที่ 3.35

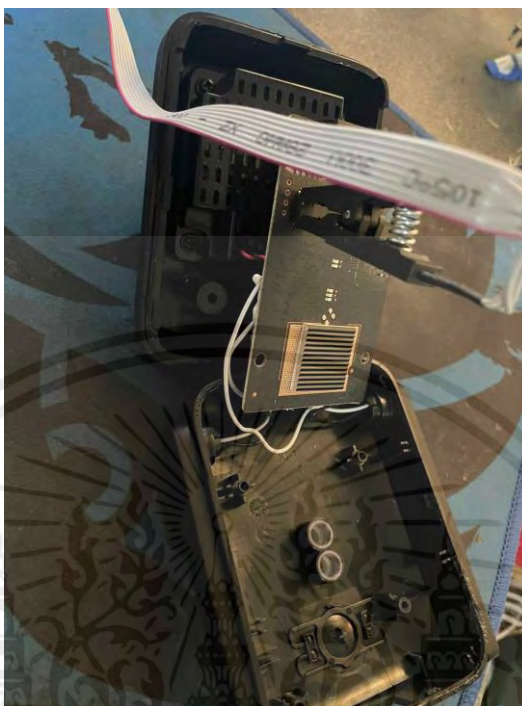


รูปที่ 3.33 การเตรียมอุปกรณ์ EEPROM Flash BIOS



รูปที่ 3.34 ขาที่ 1 คือขาที่อยู่ด้านล่างซ้ายสุดบริเวณลูกศรสีทองแดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

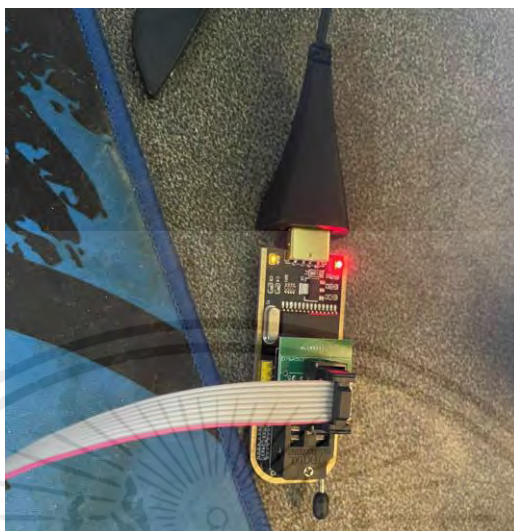


รูปที่ 3.35 ทำการเชื่อมต่อ CH เข้ากับ SPI Flash Memory ของ Router



รูปที่ 3.36 ทำการเชื่อมต่อ CH เข้ากับ SPI Flash Memory ของ Router

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



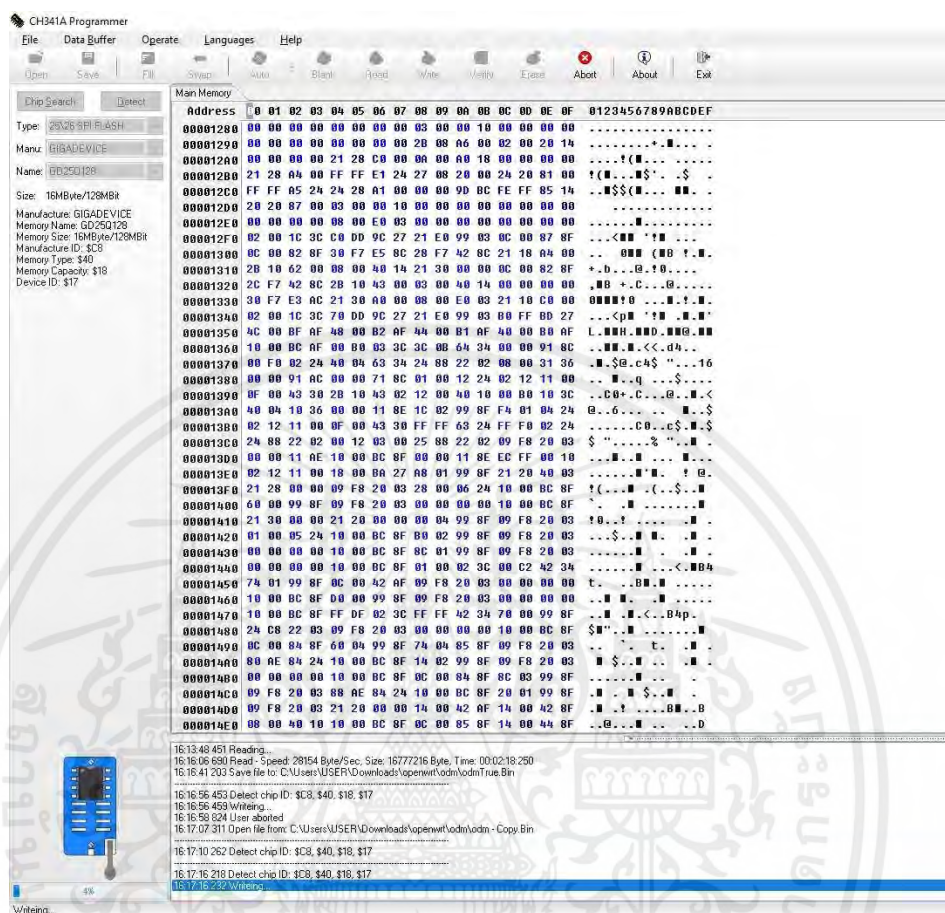
รูปที่ 3.37 ทำการเชื่อมต่อ CH เข้ากับคอมพิวเตอร์ผ่านช่องเสียบ USB

2) ทำการ Read SPI Firmware ที่อยู่ใน SPI Flash Memory ของ Router เพื่อนำมาแก้ไขและบันทึกสำรองไว้ และทำการ Write SPI เพื่อทำการแก้ไข Firmware โดยทำการแก้ไข Parameter Input\_wait จาก off เป็น on และ Flash ลงไปใน Memory เพื่อให้สามารถเลือกโหมดในการ Boot Router ได้

Address	0	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
00000000	FF	00	00	10	00	00	00	00	FD	00	00	10	00	00	00	00	.....
00000010	10	03	00	10	00	00	00	00	16	03	00	10	00	00	00	00	.....
00000020	14	03	00	10	00	00	00	00	12	03	00	10	00	00	00	00	.....
00000030	10	03	00	10	00	00	00	00	0E	03	00	10	00	00	00	00	.....
00000040	0C	03	00	10	00	00	00	00	0A	03	00	10	00	00	00	00	.....
00000050	08	03	00	10	00	00	00	00	06	03	00	10	00	00	00	00	.....
00000060	04	03	00	10	00	00	00	00	02	03	00	10	00	00	00	00	.....
00000070	00	03	00	10	00	00	00	00	FE	02	00	10	00	00	00	00	.....
00000080	FC	02	00	10	00	00	00	00	FA	02	00	10	00	00	00	00	.....
00000090	F8	02	00	10	00	00	00	00	F6	02	00	10	00	00	00	00	.....
000000A0	F4	02	00	10	00	00	00	00	F2	02	00	10	00	00	00	00	.....
000000B0	F0	02	00	10	00	00	00	00	EE	02	00	10	00	00	00	00	.....
000000C0	EC	02	00	10	00	00	00	00	EA	02	00	10	00	00	00	00	.....
000000D0	E8	02	00	10	00	00	00	00	E6	02	00	10	00	00	00	00	.....
000000E0	E4	02	00	10	00	00	00	00	E2	02	00	10	00	00	00	00	.....
000000F0	E0	02	00	10	00	00	00	00	DE	02	00	10	00	00	00	00	.....
00000100	DC	02	00	10	00	00	00	00	DA	02	00	10	00	00	00	00	.....
00000110	D8	02	00	10	00	00	00	00	D6	02	00	10	00	00	00	00	.....
00000120	D4	02	00	10	00	00	00	00	D2	02	00	10	00	00	00	00	.....
00000130	D0	02	00	10	00	00	00	00	CE	02	00	10	00	00	00	00	.....

รูปที่ 3.38 Read SPI Firmware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.39 Write SPI Firmware

3) ทำการเลือกโหมดการทำงานของ Router โดยเลือกเป็นโหมดที่ 2 ดังรูปที่ 3.40 ซึ่งในโหมดนี้ Router จะทำการโหลด Firmware ที่เราต้องการติดตั้งและเขียนทับลงไป ใน SPI Flash Memory โดยในที่นี้เราจะทำการติดตั้ง Firmware OpenWrt ลงบน Router โดยจะใช้งานโปรแกรม Tftpd64 ในการใช้งานโปรโตคอล TFTP ในการโอนย้ายไฟล์ Firmware OpenWrt จากคอมพิวเตอร์ซึ่งเป็น TFTP Sever ไปสู่ Router ซึ่งเป็น TFTP Client ในส่วนของ Router ต้องทำการใส่ข้อมูลต่างๆได้แก่ IP ของอุปกรณ์ TFTP Sever คือ 192.168.31.100 IP ของอุปกรณ์ TFTP Client คือ 192.168.31.1 และชื่อไฟล์ที่ต้องการโอนย้ายดังรูปที่ 3.41 ในส่วนของ คอมพิวเตอร์ต้องทำการตั้งค่าต่างๆใน โปรแกรมได้แก่ ตำแหน่งที่อยู่ของไฟล์ที่ต้องการโอนย้าย IP ของคอมพิวเตอร์ และชื่อของไฟล์ที่ต้องการโอนย้าย ดังรูปที่ 3.42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

COM3-PUTTY
Flash component: SPI Flash
Date:Mar 31 2023 Time:06:51:50

=====
icache: sets:512, ways:4, linesz:32 ,total:65536
dcache: sets:256, ways:4, linesz:32 ,total:32768

#### The CPU freq = 575 MHZ ####
estimate memory size =64 Mbytes
RESET MT7628 PHY!!!!!!
Please choose the operation:
 1: Load system code to SDRAM via TFTP.
 2: Load system code then write to Flash via TFTP.
 3: Boot system code via Flash (default).
 4: Entr boot command line interface.
 9: Load Boot Loader code then write to Flash via TFTP.

You choosed 2

2: System Load Linux Kernel then write to Flash via TFTP.
Warning!! Erase Linux in Flash then burn new one. Are you sure?(Y/N)
Please Input new ones /or Ctrl-C to discard
Input device IP () --:

```

รูปที่ 3.40 เลือกโหมดในการทำงานของ Router

```

COM3-PUTTY
icache: sets:512, ways:4, linesz:32 ,total:65536
dcache: sets:256, ways:4, linesz:32 ,total:32768

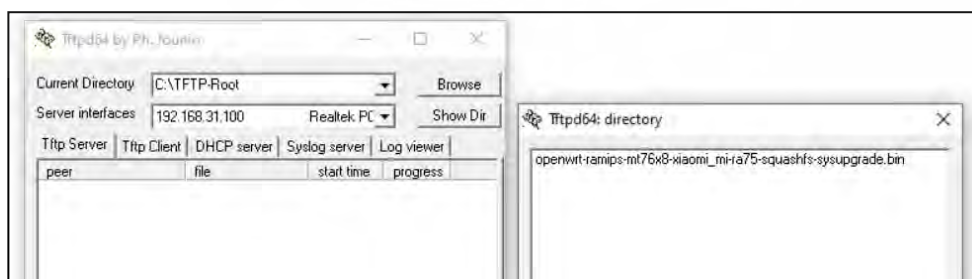
#### The CPU freq = 575 MHZ ####
estimate memory size =64 Mbytes
RESET MT7628 PHY!!!!!!
Please choose the operation:
 1: Load system code to SDRAM via TFTP.
 2: Load system code then write to Flash via TFTP.
 3: Boot system code via Flash (default).
 4: Entr boot command line interface.
 9: Load Boot Loader code then write to Flash via TFTP.

You choosed 2

2: System Load Linux Kernel then write to Flash via TFTP.
Warning!! Erase Linux in Flash then burn new one. Are you sure?(Y/N)
Please Input new ones /or Ctrl-C to discard
Input device IP () ==:192.168.31.1
Input server IP () ==:192.168.31.100
Input Linux Kernel filename () ==:openwrt-ramips-mt76x8-xiaomi_mi-ra75-squashfs-sysupgrade

```

รูปที่ 3.41 ใส่ข้อมูลต่างๆให้กับ Router



รูปที่ 3.42 การตั้งค่าในโปรแกรม Tftpd64 ในส่วนของคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ทำการพิมพ์คำสั่ง `opkg update` ใน Router เพื่ออัปเดตเครื่องมือในการติดตั้ง Package ต่างๆ ดังรูปที่ 3.43 จากนั้นทำการติดตั้ง `Luci` ซึ่งเป็น Web Interface ของ OpenWrt Router โดยใช้คำสั่ง `opkg install luci` ดังรูปที่ 3.44 จากนั้นทำการตั้งค่า Routing Protocol หมายเลข IP และ Netmask ของ Router โดยใช้คำสั่งดังรูปที่ 3.45 จากนั้นจะสามารถใช้งาน Web Interface ของ OpenWrt Router ได้ดังรูปที่ 3.46

```

root@OpenWrt:~# opkg update
Downloading https://downloads.openwrt.org/snapshots/targets/ramips/mt76x8/packages/Packages.gz
Updated list of available packages in /var/opkg-lists/openwrt_core
Downloading https://downloads.openwrt.org/snapshots/targets/ramips/mt76x8/packages/Packages.sig
Signature check passed.
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/base/Packages.gz
Updated list of available packages in /var/opkg-lists/openwrt_base
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/base/Packages.sig
Signature check passed.
Downloading https://downloads.openwrt.org/snapshots/targets/ramips/mt76x8/kmods/5.15.148-1-c669
Updated list of available packages in /var/opkg-lists/openwrt_kmods
Downloading https://downloads.openwrt.org/snapshots/targets/ramips/mt76x8/kmods/5.15.148-1-c669
Signature check passed.
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/Packages.gz
Updated list of available packages in /var/opkg-lists/openwrt_luci
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/Packages.sig
Signature check passed.
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/packages/Packages.gz
Updated list of available packages in /var/opkg-lists/openwrt_packages
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/packages/Packages.sig
Signature check passed.

```

รูปที่ 3.43 การอัปเดตเครื่องมือที่ใช้ในการติดตั้ง Package ต่างๆ

```

root@OpenWrt:~# opkg install luci
Installing luci (git-23.051.66410-a505bb1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/luci_git-23.051.66410-
Installing luci-proto-ipv6 (git-23.355.78793-4ad5f81) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/luci-proto-ipv6_git-23
Installing rpcd (2023-07-01-c07ab2f9-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/base/rpcd_2023-07-01-c07ab2
Installing rpcd-mod-file (2023-07-01-c07ab2f9-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/base/rpcd-mod-file_2023-07-
Installing rpcd-mod-luci (2023-0123-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/rpcd-mod-luci_2023-0123
Installing rpcd-mod-ucode (2023-07-01-c07ab2f9-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/base/rpcd-mod-ucode_2023-07
Installing cgi-io (2022-08-10-901b0f04-21) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/packages/cgi-io_2022-08-10-
Installing ucode-mod-math (2023-11-30-6e89b89e-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/base/ucode-mod-math_2023-11
Installing ucode-mod-html (1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/ucode-mod-html_1_mipse
Installing liblucihttp0 (2023-03-15-9b5b683f-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/liblucihttp0_2023-03-1
Installing liblucihttp-ucode (2023-03-15-9b5b683f-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/liblucihttp-ucode_2023
Installing luci-base (git-24.031.09542-8a0ee66) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/luci-base_git-24.031.0
Installing luci-app-firewall (git-23.364.10482-c74c861) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/luci-app-firewall_git-
Installing rpcd-mod-iwinfo (2023-07-01-c07ab2f9-1) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/base/rpcd-mod-iwinfo_2023-0
Installing luci-mod-status (git-23.351.84061-8b65199) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/luci-mod-status_git-23
Installing luci-mod-system (git-23.364.85099-cbfa9ca) to root...
Downloading https://downloads.openwrt.org/snapshots/packages/mipsel_24kc/luci/luci-mod-system_git-23
Installing luci-mod-network (git-24.031.09542-8a0ee66) to root...

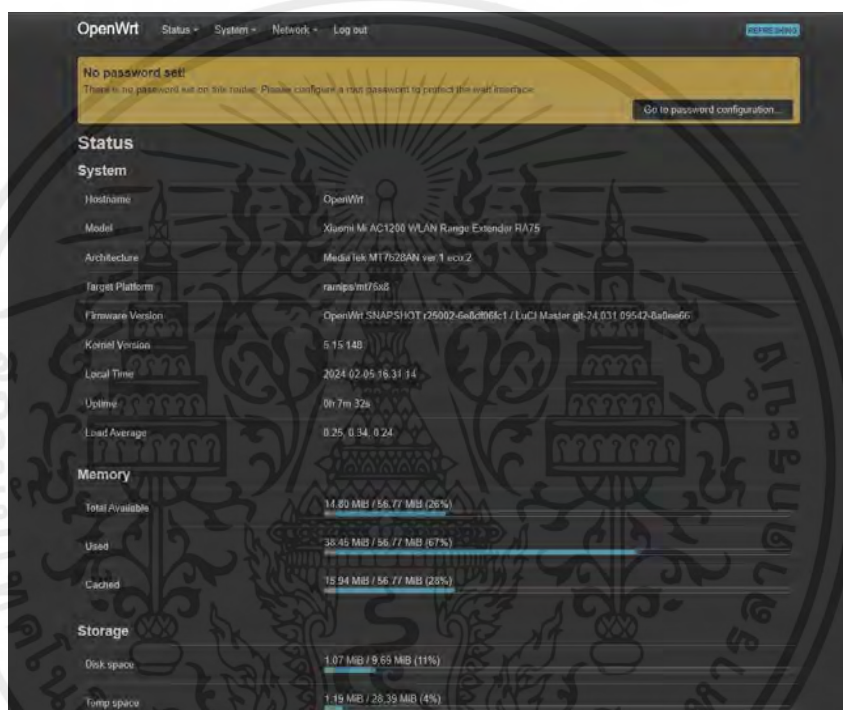
```

รูปที่ 3.44 การติดตั้ง Luci

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
uci set network.lan.proto="static"
uci set network.lan.ipaddr="192.168.1.1"
uci set network.lan.netmask="255.255.255.0"
uci commit network
```

รูปที่ 3.45 การตั้งค่า Routing Protocol หมายเลข IP และ Netmask ของ Router



รูปที่ 3.46 Web Interface ของ OpenWrt Router

5) ทำการติดตั้ง Package ต่างๆลงบน OpenWrt Router ผ่าน Web Interface โดยเลือกเฉพาะส่วนที่จำเป็นต้องใช้งานเนื่องจากหน่วยความจำของ Router มีจำนวนจำกัด โดยมี Package ต่างๆที่จำเป็นเป็นต้องใช้งานดังรูปที่ 3.47 จากนั้นทำการตัดออก Library จากคอมพิวเตอร์ไปยัง OpenWrt Router ด้วยโปรแกรม WinSCP โดยเลือก Path ปลายทางเป็น Python311\Lib\site-packages จะนั้นจะสามารถใช้งาน OpenWrt Router ในการ Run Python Script ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

python3-base	3.11.7-1	~216.76 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-codecs	3.11.7-1	~722.47 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-decimal	3.11.7-1	~71.88 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-email	3.11.7-1	~180.31 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-idna	3.4-1	~68.97 KiB	A library to support the Internationalised Domain Names in Applications...	Remove...
python3-light	3.11.7-1	~2.71 MiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-logging	3.11.7-1	~83.41 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-openssl	3.11.7-1	~65.43 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-urllib	3.11.7-1	~83.59 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
python3-uuid	3.11.7-1	~16.17 KiB	Python is an interpreted, interactive, object-oriented programming...	Remove...
libpython3-3.11	3.11.7-1	~1.13 MiB	Python is an interpreted, interactive, object-oriented programming...	Remove...

รูปที่ 3.47 Package ที่จำเป็นต้องใช้งาน

idna	9/2/2567 11:01	File folder
idna-3.4.dist-info	9/2/2567 11:01	File folder
paho	9/2/2567 11:01	File folder
paho_mqtt-1.6.1.dist-info	9/2/2567 11:01	File folder
pymysql	9/2/2567 11:01	File folder
PyMySQL-1.1.0.dist-info	9/2/2567 11:01	File folder
urllib3	9/2/2567 10:34	File folder
urllib3-2.0.7.dist-info	9/2/2567 10:34	File folder

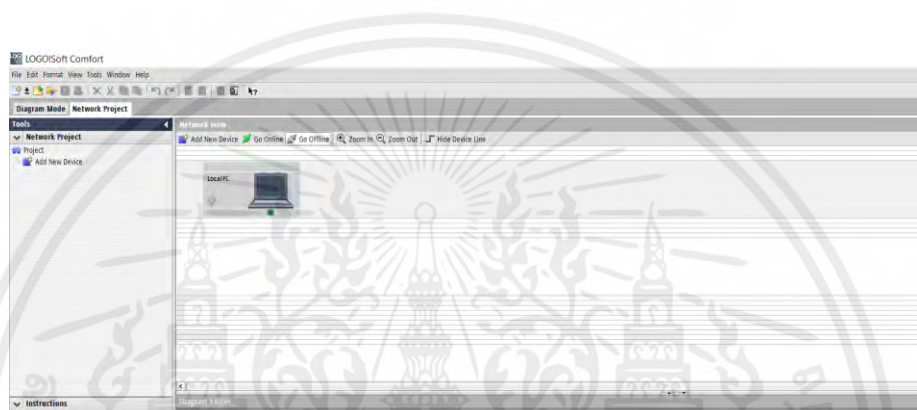
รูปที่ 3.48 Library ที่จำเป็นต้องใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.9 การออกแบบโปรแกรมการทำงานของ PLC Controller สำหรับใช้ในการทดสอบการทำงานเบื้องต้น

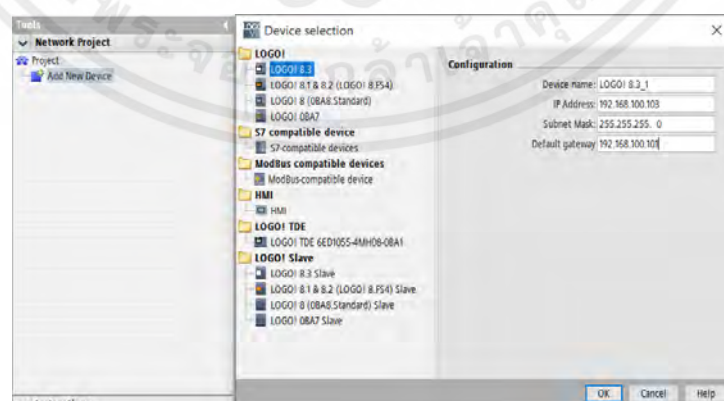
ทำการออกแบบโปรแกรมการทำงานให้กับ PLC Controller ผ่านโปรแกรม LOGO! Soft Comfort V8.3 โดยมีลำดับขั้นตอนในการออกแบบดังนี้

- 1) ทำการเปิดใช้งาน โปรแกรม LOGO! Soft Comfort V8.3 บนเครื่องคอมพิวเตอร์



รูปที่ 3.49 หน้าแรกของโปรแกรม LOGO! Soft Comfort V8.3

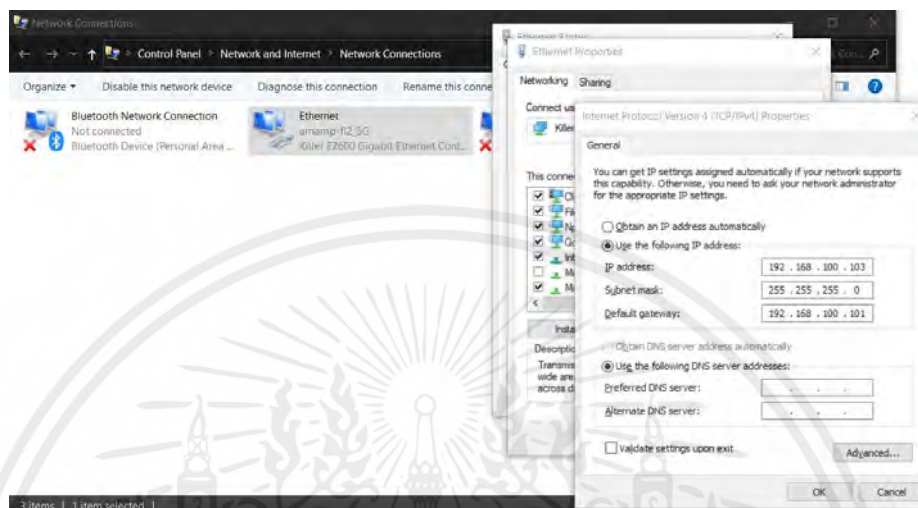
- 2) เปิดหน้า Network Project เลือก Add New Device เชื่อมต่อกับอุปกรณ์ LOGO! 8.3 และทำการตั้ง IP Adress, Subnet Mask, Default gateway ตามที่ตั้งค่าบน PLC Controller



รูปที่ 3.50 การเพิ่มอุปกรณ์ PLC Controller ในโปรแกรม

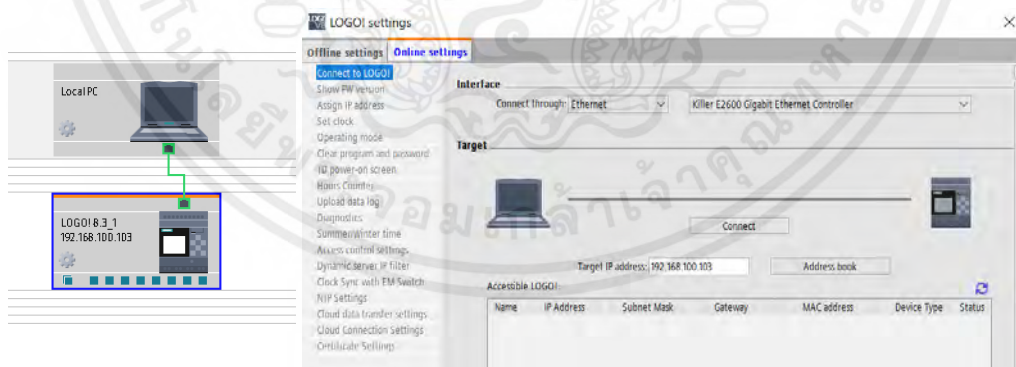
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ตั้งค่า IP Adress อุปกรณ์ที่เชื่อมต่อกับ PLC Controller ผ่าน Ethernet ให้อยู่ใน Subset เดียวกัน



รูปที่ 3.51 การกำหนด IP address บนเครื่องคอมพิวเตอร์

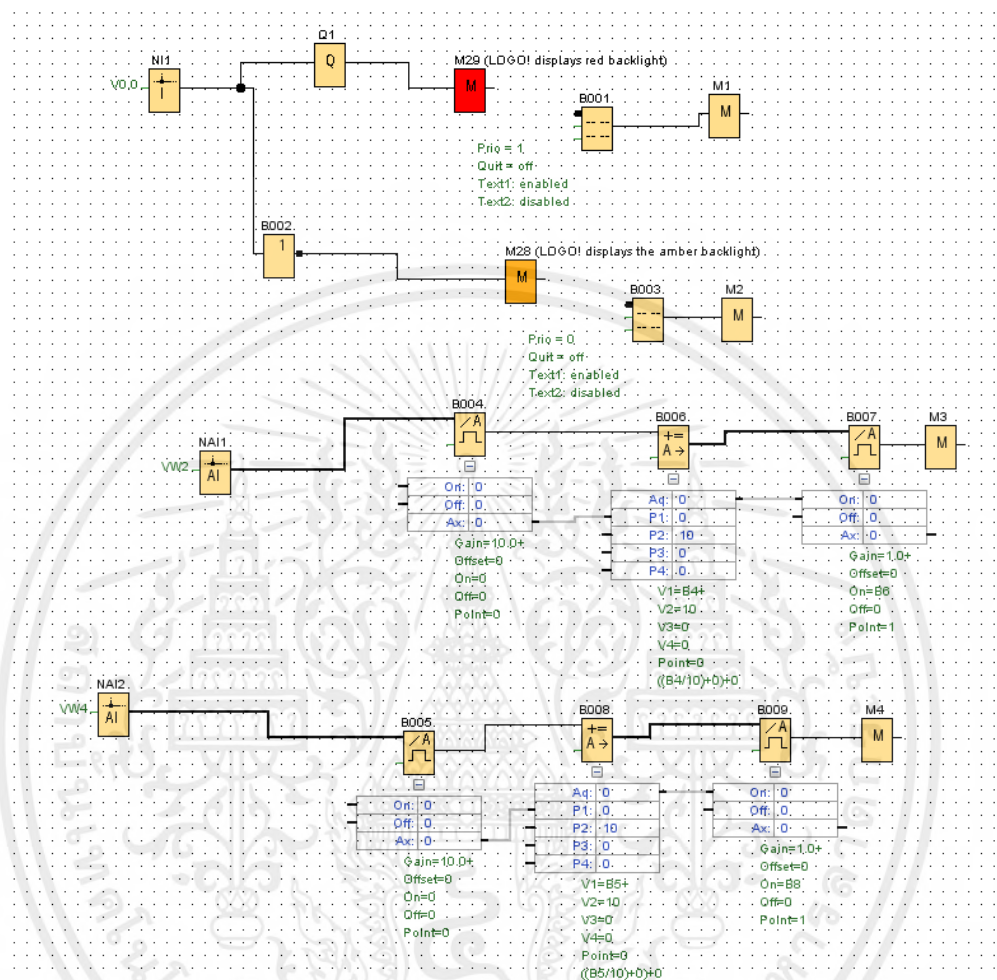
4) คลิกที่รูปฟันเฟืองของ PLC Controller ไปยังหน้า online setting กด connect เพื่อทดสอบการเชื่อมต่อ



รูปที่ 3.52 การเชื่อมต่อของคอมพิวเตอร์และ PLC Controller

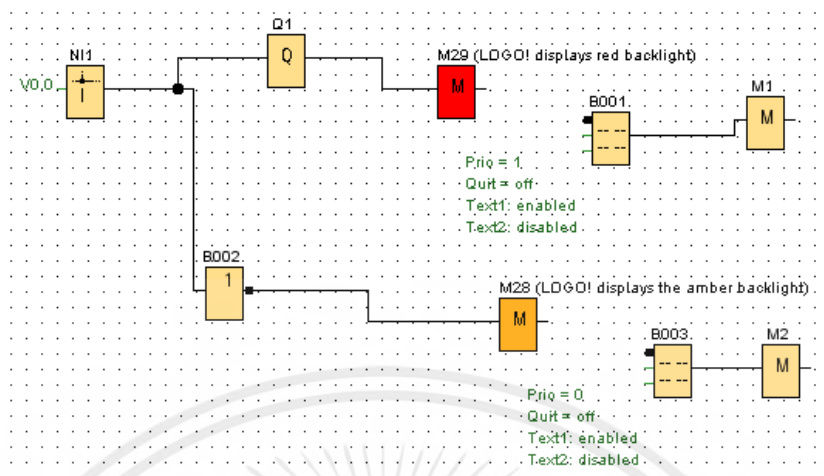
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) ออกแบบโปรแกรมในการทำงานควบคุม Output Q1 ที่เป็น Relay Switch

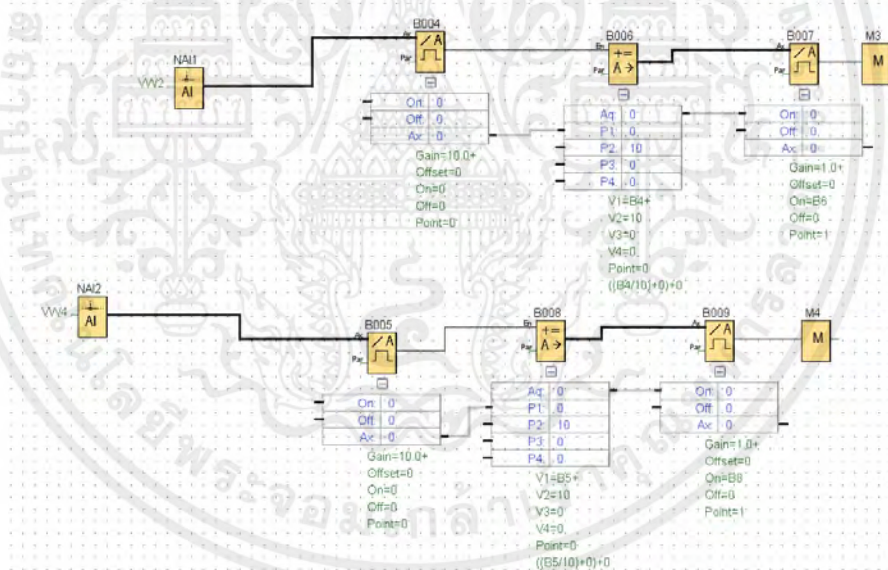


รูปที่ 3.53 โปรแกรมที่ออกแบบใน LOGO! Soft Comfort V8.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



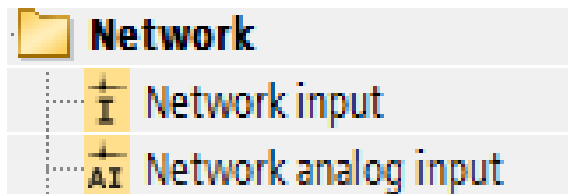
รูปที่ 3.54 โปรแกรมส่วนที่ใช้รับค่า Coil single bit 0 หรือ 1



รูปที่ 3.55 โปรแกรมส่วนที่ใช้รับค่า Input Register 16-bit โดย VW2 เป็นตัวแปรสำหรับรับค่าอุณหภูมิ และ VW4 เป็นตัวแปรสำหรับรับค่าความชื้นสัมพัทธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

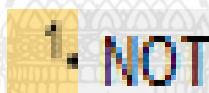
เครื่องมือที่ใช้งานในการออกแบบโปรแกรมการทำงาน



รูปที่ 3.56 Network input สำหรับรับค่า Coil และ Network analog input สำหรับรับค่า Input Register



รูปที่ 3.57 Output 1-4 ของ PLC Controller เมื่อ logic เป็น 0 จะ open circuit เมื่อ logic เป็น 1 closed circuit



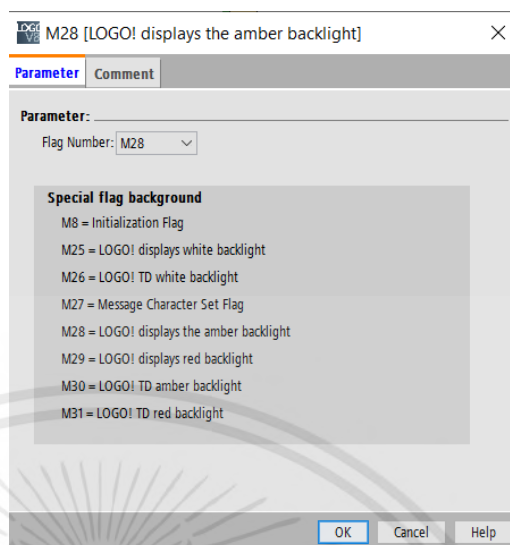
รูปที่ 3.58 NOT เป็นการเปลี่ยน logic จาก 0 เป็น 1 หรือ จาก 1 เป็น 0



รูปที่ 3.59 Mathematic instruction ใช้ในสำหรับการทำกระบวนการคณิตศาสตร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

M Flag



รูปที่ 3.60 Flag ตัวกำหนด status สีของหน้าจอ

Message texts

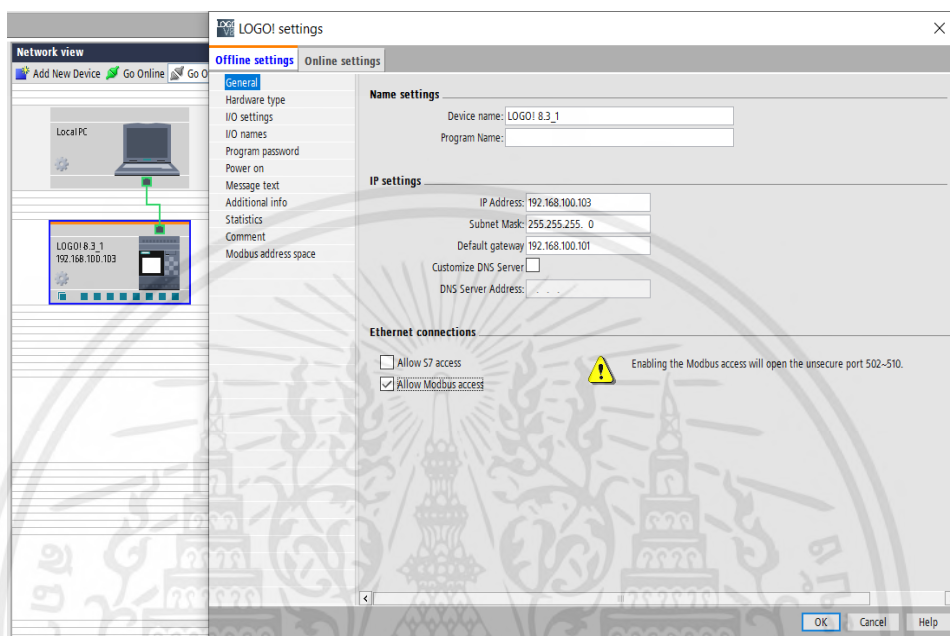
รูปที่ 3.61 Message texts หน้าจอแสดงข้อความ

Threshold trigger

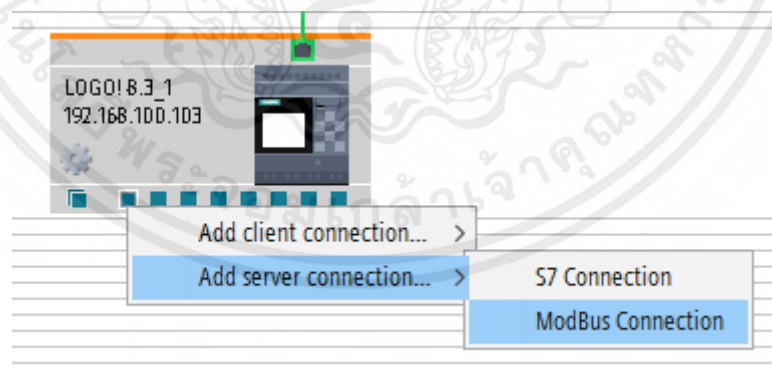
รูปที่ 3.62 Threshold trigger เป็นการกำหนดช่วงข้อมูลที่ได้รับให้มี logic เป็น 0 หรือ 1 สามารถกำหนดเพิ่มค่า gain และ จุดทศนิยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) ทำการตั้งค่าการเชื่อมต่อ Modbus TCP บน PLC Controller โดยกดที่รูปฟันเฟืองของ PLC Controller ที่แถบ General ดึงถูกที่ Allow Modbus access เพื่อเปิดการเชื่อมต่อ Modbus TCP



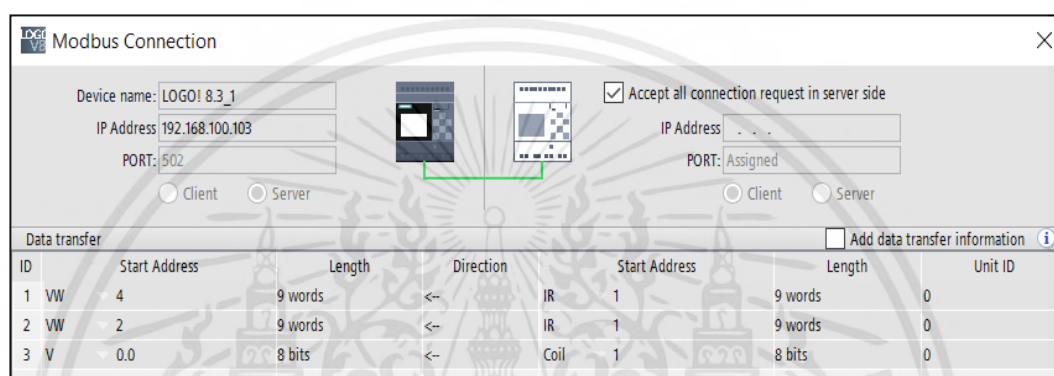
รูปที่ 3.63 การตั้งค่า PLC Controller ทั่วไป



รูปที่ 3.64 สร้าง Modbus server connection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) สร้าง Modbus Connection โดยให้ PLC Controller เป็น server แล้วสร้างตารางข้อมูลที่ได้รับมาจาก Client ที่เป็น Raspberry pi 3 ค่า โดยกำหนด ตัวแปร V start Adress 0.0 เป็นตัวแปรรับค่า Coil สำหรับการแสดงสีหน้าจอ 0 เป็น สีส้ม 1 เป็นสีแดง ตัวแปร VW start Adress 2 เป็นตัวแปรรับค่า Input Register รับค่าอุณหภูมิ ตัวแปร VW start Adress 4 เป็นตัวแปรรับค่า Input Register รับค่าความชื้นสัมพัทธ์ เมื่อเสร็จทุกขั้นตอนแล้ว upload โปรแกรมลง PLC Controller

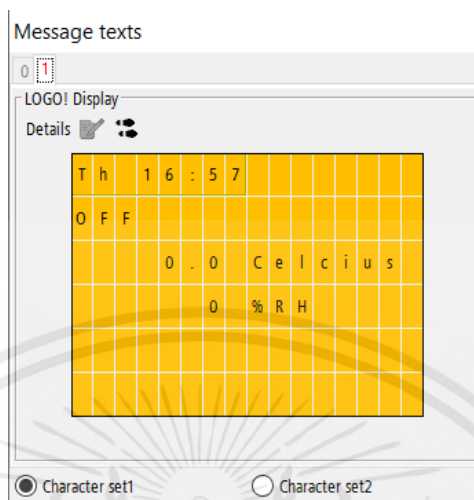


รูปที่ 3.65 การสร้าง Data transfer information table

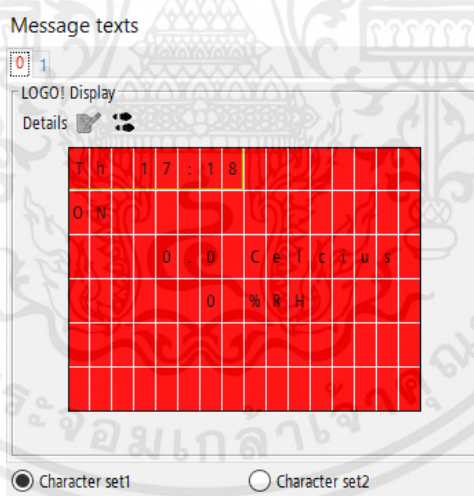
ตารางที่ 3.1 ตัวแปรที่ใช้ในการรับส่งข้อมูลและรายละเอียดเกี่ยวกับตัวแปร

ตัวแปร	รายละเอียดเกี่ยวกับตัวแปร
V0.0	รับข้อมูล coil 0 และ 1 เพื่อแสดงผลสถานะ ON-OFF
VW2	รับข้อมูล Holding Register เพื่อแสดงค่าอุณหภูมิในดิน
VW4	รับข้อมูล Holding Register เพื่อแสดงค่าความชื้นในดิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.66 หน้าจอแสดงผลของ PLC Controller ขณะที่สถานะของ Output คือ ปิด



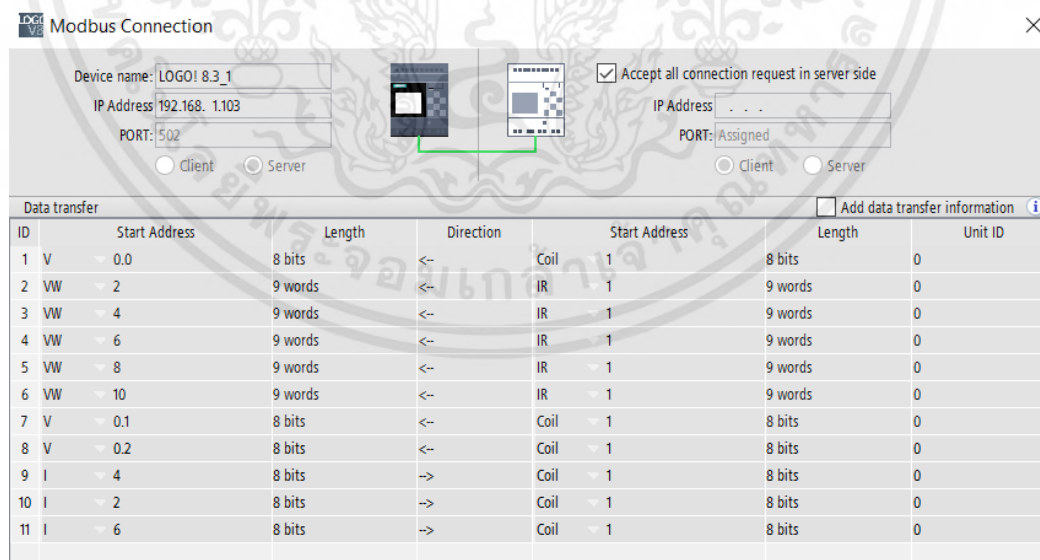
รูปที่ 3.67 หน้าจอแสดงผลของ PLC Controller ขณะที่สถานะของ Output คือ เปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.10 การออกแบบโปรแกรมการทำงานของ PLC Controller สำหรับการใช้งานจริงในระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

ทำการปรับปรุงและออกแบบโปรแกรมการควบคุมการทำงานของ PLC Controller เพิ่มเติม ผ่านโปรแกรม LOGO! 8.3 Software โดยทำการเพิ่ม Network Input สำหรับรับค่า Coil (0 หรือ 1) ให้กลายเป็น 3 ตัวแปรได้แก่ V0.0 V0.1 และ V0.2 เพื่อใช้ในการควบคุมอุปกรณ์ Output ได้แก่ ไฟ LED สำหรับปลูกต้นไม้ พัดลมระบายอากาศ และ ป้อนน้ำ ตามลำดับ ซึ่งในส่วนของป้อนน้ำจะทำการใช้เครื่องมือ On- Delay ในการกำหนดระยะเวลาการทำงานของป้อนน้ำให้ทำงานเพียงแค่ 5 วินาที เพื่อป้องกันไม่ให้น้ำที่รดมากเกินไป ทำการเพิ่ม Input ทั้ง 3 ตัวได้แก่ I4, I2 และ I6 เพื่อใช้สำหรับการตรวจสอบสถานะการทำงานของอุปกรณ์ Output ว่าเป็นไปตามที่ PLC Controller ควบคุมหรือไม่ และทำการออกแบบหน้าจอแสดงผลของ PLC Controller ให้ประกอบไปด้วย 4 หน้า ได้แก่ 1. หน้าสำหรับแสดงวันที่และเวลาแบบเรียลไทม์ 2. หน้าสำหรับแสดงสถานะการทำงานของอุปกรณ์ Output 3. หน้าสำหรับแสดงค่าที่วัดได้จาก Soil Temperature and Humidity Sensor 4. หน้าสำหรับแสดงผลค่าที่วัดได้จาก Light Temperature and Humidity Sensor

จากนั้นทำการตั้งค่าพารามิเตอร์ต่างๆเกี่ยวกับการรับ-ส่งข้อมูลด้วยโปรโตคอล Modbus TCP/IP แสดงดังรูปที่ 3.68



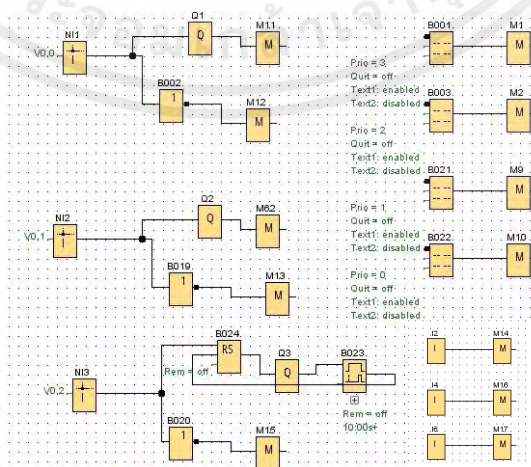
ID	Start Address	Length	Direction	Start Address	Length	Unit ID
1	V 0.0	8 bits	<-	Coil 1	8 bits	0
2	VW 2	9 words	<-	IR 1	9 words	0
3	VW 4	9 words	<-	IR 1	9 words	0
4	VW 6	9 words	<-	IR 1	9 words	0
5	VW 8	9 words	<-	IR 1	9 words	0
6	VW 10	9 words	<-	IR 1	9 words	0
7	V 0.1	8 bits	<-	Coil 1	8 bits	0
8	V 0.2	8 bits	<-	Coil 1	8 bits	0
9	I 4	8 bits	->	Coil 1	8 bits	0
10	I 2	8 bits	->	Coil 1	8 bits	0
11	I 6	8 bits	->	Coil 1	8 bits	0

รูปที่ 3.68 การสร้าง Data transfer information table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

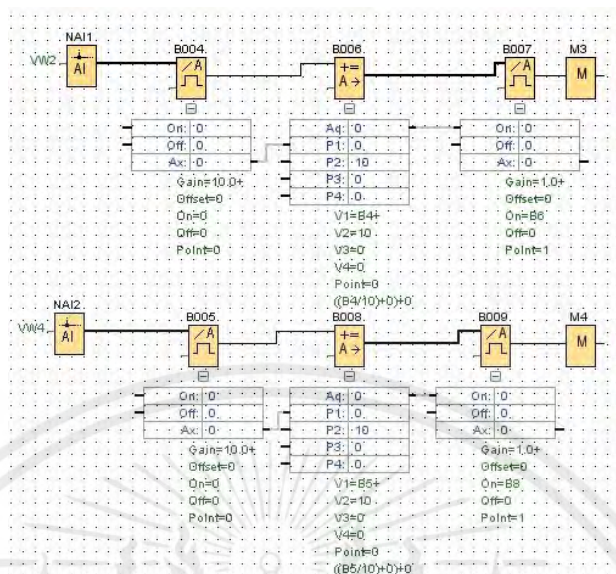
ตารางที่ 3.2 ตัวแปรที่ใช้ในการรับส่งข้อมูลและรายละเอียดเกี่ยวกับตัวแปร

ตัวแปร	รายละเอียดเกี่ยวกับตัวแปร
V0.0	รับข้อมูล coil 0 และ 1 เพื่อควบคุมการทำงานของไฟ LED
V0.1	รับข้อมูล coil 0 และ 1 เพื่อควบคุมการทำงานของพัดลมระบายอากาศ
V0.2	รับข้อมูล coil 0 และ 1 เพื่อควบคุมการทำงานของปั้มน้ำ
VW2	รับข้อมูล Holding Register เพื่อแสดงค่า อุณหภูมิในดิน
VW4	รับข้อมูล Holding Register เพื่อแสดงค่า ความชื้นในดิน
VW6	รับข้อมูล Holding Register เพื่อแสดงค่าอุณหภูมิในอากาศ
VW8	รับข้อมูล Holding Register เพื่อแสดงค่าความชื้นในอากาศ
VW10	รับข้อมูล Holding Register เพื่อแสดงค่าความสว่าง
I2	รับข้อมูล coil 0 และ 1 เพื่อตรวจสอบสถานะการทำงานของไฟ LED
I4	รับข้อมูล coil 0 และ 1 เพื่อตรวจสอบสถานะการทำงานของพัดลมระบายอากาศ
I6	รับข้อมูล coil 0 และ 1 เพื่อตรวจสอบสถานะการทำงานของปั้มน้ำ

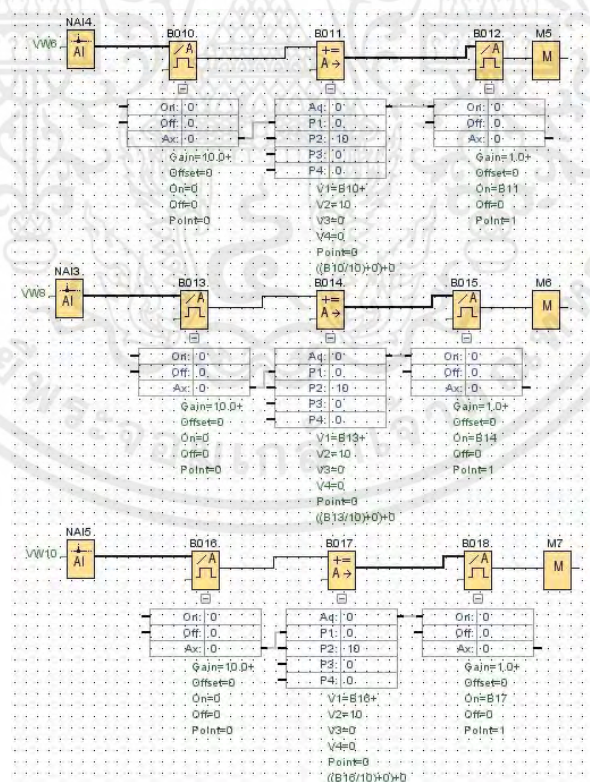


รูปที่ 3.69 โปรแกรมส่วนที่ใช้รับค่า Coil single bit 0 หรือ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.70 โปรแกรมส่วนที่ใช้รับค่า Input Register 16-bit ของ Soil Temperature and Humidity Sensor

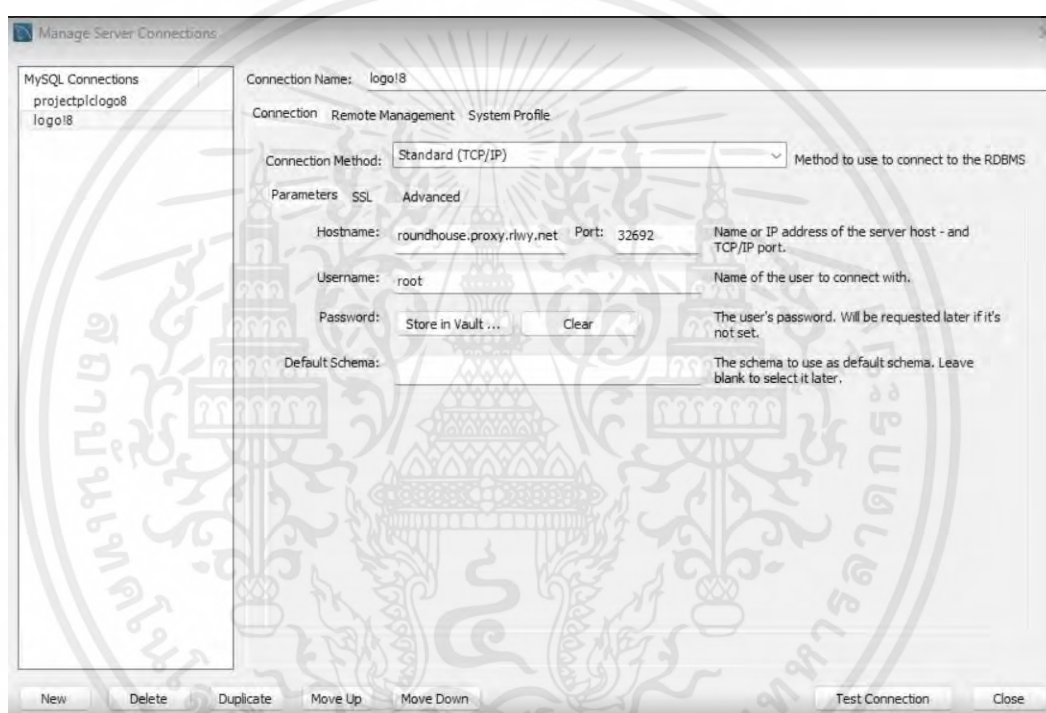


รูปที่ 3.71 โปรแกรมส่วนที่ใช้รับค่า Input Register 16-bit ของ Light Temperature and Humidity Sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

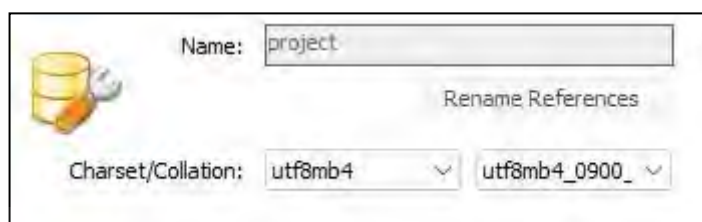
### 3.1.11 การออกแบบฐานข้อมูล

ทำการออกแบบฐานข้อมูลโดยเริ่มจากการเชื่อมต่อกับฐานข้อมูลของเราจากการเชื่อมต่อ Hostname, Port, Username และ Password หลังจากที่สามารถเชื่อมต่อกับฐานข้อมูลของเราได้ก็จะทำการสร้าง Schema สำหรับ Project ของเรา เพื่อทำการสร้างตาราง หรือ Table โดยตารางที่เราได้สร้างขึ้นจะใช้สำหรับเก็บข้อมูลต่างๆในแต่ละคอลัมน์ ที่ได้จากการ Subscribe โดยใช้ Protocol MQTT และ เก็บค่าสถานะการทำงานของอุปกรณ์ โดยแสดงดังรูปที่ 3.72, 3.73 และ 3.74 ตามลำดับ

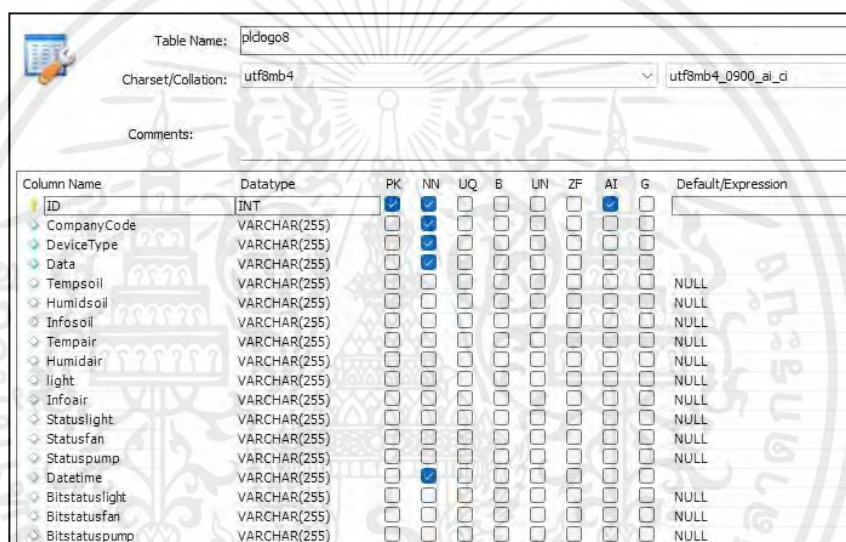


รูปที่ 3.72 การเชื่อมต่อเข้ากับฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.73 การสร้าง Schema หรือ Database ในฐานข้อมูล



รูปที่ 3.74 การสร้างตาราง หรือ Table สำหรับจัดเก็บข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.12 การออกแบบเว็บไซต์สำหรับแสดงผลข้อมูล

ทำการออกแบบเว็บไซต์สำหรับแสดงผลข้อมูลโดยประกอบไปด้วย 5 หน้าได้แก่

1. Dashboard เป็นหน้าสำหรับแสดงค่าที่ Sensor แต่ละตัววัดได้ สถานะการทำงานของ PLC Controller ควบคุมอุปกรณ์ และการตรวจสอบสถานะการทำงานจริงๆของอุปกรณ์ Output ว่า เป็นไปตามที่ PLC Controller ควบคุมหรือไม่



รูปที่ 3.75 หน้า Dashboard

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2. About Project เป็นหน้าสำหรับแสดงขอบเขตและบล็อกไดอแกรมของปริญญาโท

**About Project**

Using PLC Controller with IoT systems to control the electronic devices

ที่ประมวลผลระบบควบคุมอัตโนมัติที่ทำงานร่วมกับ PLC Controller ร่วมกับระบบ IoT โดยมีอุปกรณ์ Sensor ร่วมกับ RS-485 ที่ส่งข้อมูลผ่าน AP Gateway ที่เชื่อมกับ Zeta Cloud หรือจากนั้นจะส่ง MQTT Protocol ไป Router ที่ติดตั้ง Firmware OpenWrt ในระบบ Subscribe ข้อมูลและทำการแปลงข้อมูลที่ได้รับเป็นข้อมูลในรูปแบบที่เก็บใน Database ของ MySQL Server หรือจากนั้นจึงไปส่งข้อมูลมาแสดงบน Website ที่ใช้ทำหน้าสำหรับ และจะทำการเชื่อมต่อ PLC Controller กับ Router ที่เชื่อมกับ LAN ต่อ PLC Controller จะรับข้อมูล Modbus TCP/IP ในกรณีนี้หรือรับค่าจากอุปกรณ์ที่รับข้อมูลในกรณีนี้คือ PLC Controller แล้วจึงทำการส่งข้อมูลในระบบ LOGIX 8.3 Software ในกรณีนี้หรือส่งข้อมูลไปยัง PLC Controller และควบคุมอุปกรณ์ Output ที่เกี่ยวข้องโดยผ่านอุปกรณ์การประมวลผล และส่งข้อมูลไปยัง PLC Controller ที่เชื่อมต่อไปยัง PLC Controller ให้มีการทำงานได้ และจะทำการแสดงข้อมูลการทำงานบนหน้าจอสมมติของอุปกรณ์ที่ส่งข้อมูลไปยัง PLC Controller

**Network Architecture**

12:06:57 AM

ZifiSense  
纵行科技

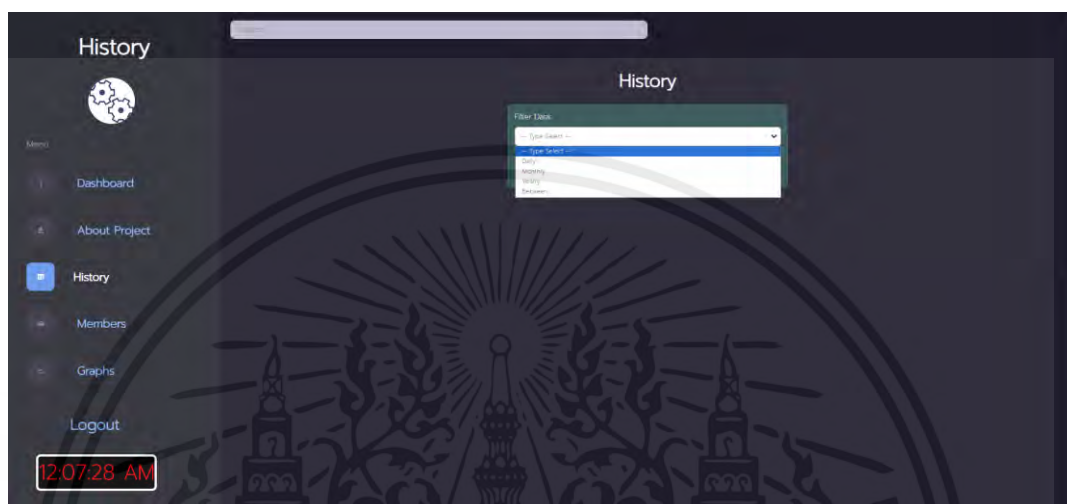
SCG

KMITL  
พระจอมเกล้าลาดกระบัง

รูปที่ 3.76 หน้า About Project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

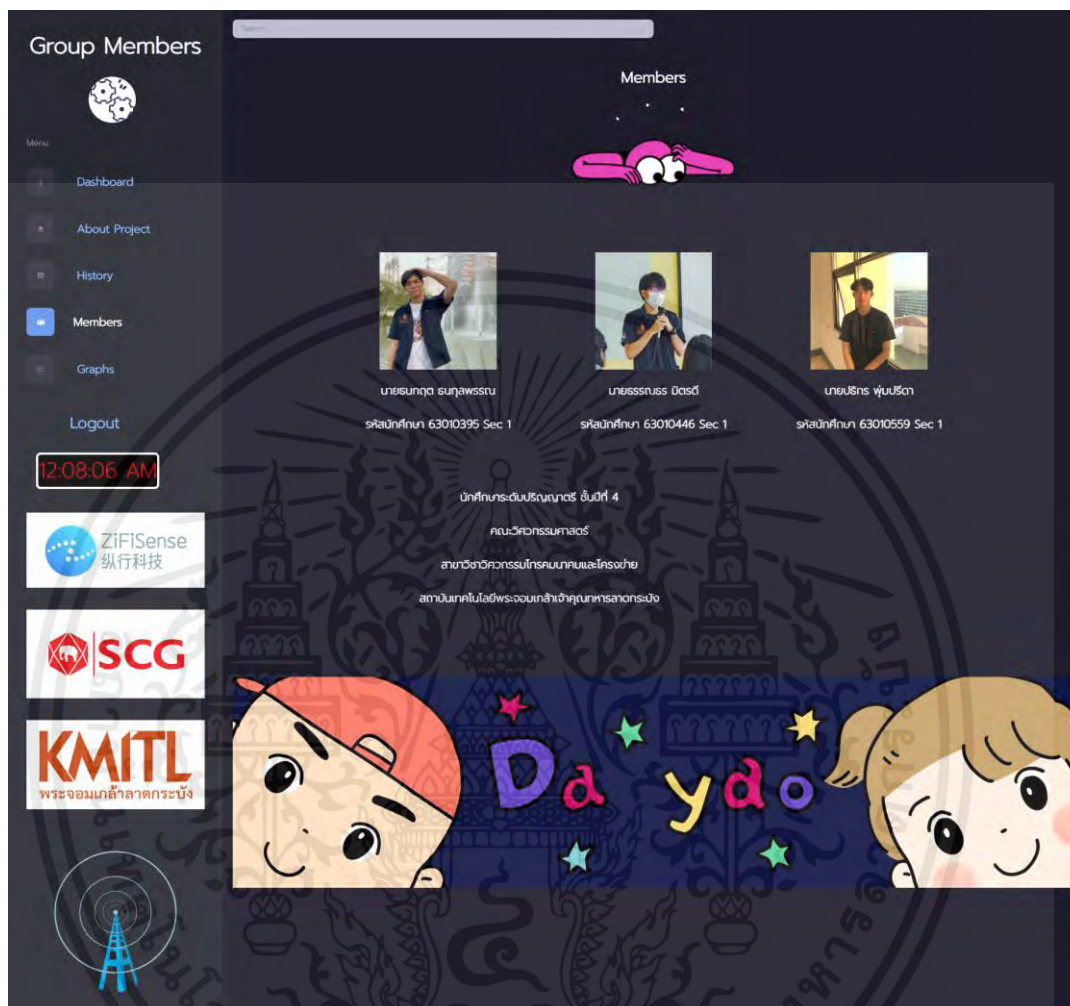
3. History เป็นหน้าสำหรับแสดงประวัติค่าที่วัดได้จาก Sensor ซึ่งสามารถเลือกการกรองข้อมูลได้ทั้งหมด 4 รูปแบบคือ รายวัน (Daily), รายเดือน (Monthly), รายปี (Yearly) และ เลือกขอบเขตที่ต้องการดูข้อมูล (Between)



รูปที่ 3.77 หน้า History

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

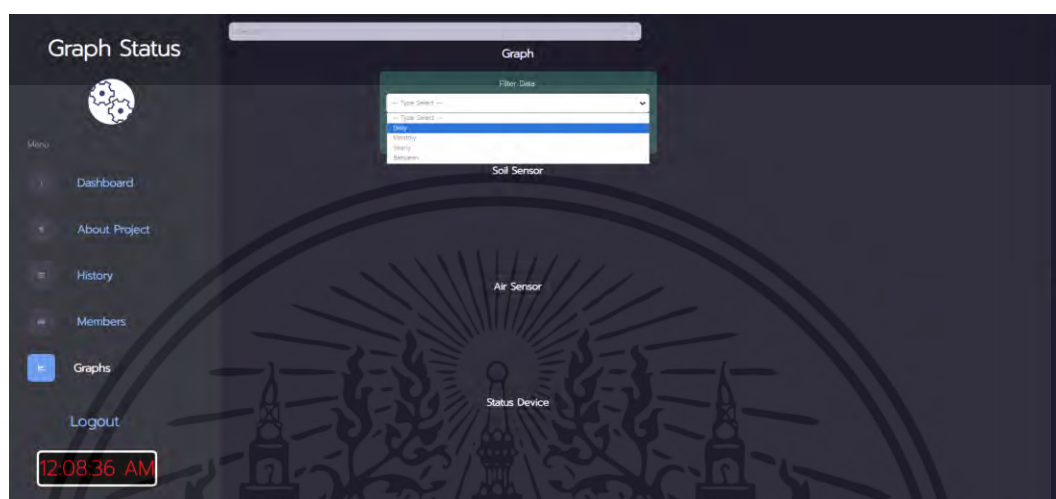
#### 4. Members เป็นหน้าสำหรับแสดงรายชื่อคณะผู้จัดทำปฏิญานิพนธ์



รูปที่ 3.78 หน้า Members

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Graphs เป็นหน้าสำหรับแสดงข้อมูลที่วัดได้จาก Sensor ในรูปแบบของกราฟ ซึ่งสามารถเลือกการกรองข้อมูลได้ทั้งหมด 4 รูปแบบคือ รายวัน (Daily), รายเดือน (Monthly), รายปี (Yearly) และเลือกขอบเขตที่ต้องการดูข้อมูล (Between)



รูปที่ 3.79 หน้า Graphs

โดยเว็บไซต์จะถูกออกแบบให้มีสิทธิในการใช้งานที่ต่างกันโดยแบ่งออกเป็น 2 ระดับ ได้แก่ User และ Admin ซึ่ง User จะสามารถใช้งานได้โดยการป้อนอีเมล ส่วน Admin จะสามารถใช้งานได้โดยการป้อน Username และ Password ในส่วนของ User จะสามารถมองเห็นแค่หน้า Dashboard, About Project และ Members ส่วน Admin จะสามารถเข้าถึงได้ทุกหน้า ซึ่ง Admin จะสามารถแบ่งได้เป็น 3 ระดับ ได้แก่

1. Admin หลัก ซึ่งสามารถเข้าถึงได้โดยการป้อน Username เป็น “admin” และ Password เป็น “Logol8” โดยจะสามารถเข้าถึงข้อมูลค่าที่วัดได้จากทั้ง 2 Sensor
2. Admin จากบริษัทที่ 1 ซึ่งสามารถเข้าถึงได้โดยการป้อน Username เป็น “company1” และ Password เป็น “Logol8” โดยจะสามารถเข้าถึงข้อมูลค่าที่วัดได้จาก Soil Temperature and Humidity Sensor เท่านั้น
3. Admin จากบริษัทที่ 2 ซึ่งสามารถเข้าถึงได้โดยการป้อน Username เป็น “company2” และ Password เป็น “Logol8” โดยจะสามารถเข้าถึงข้อมูลค่าที่วัดได้จาก Light Temperature and Humidity Sensor เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

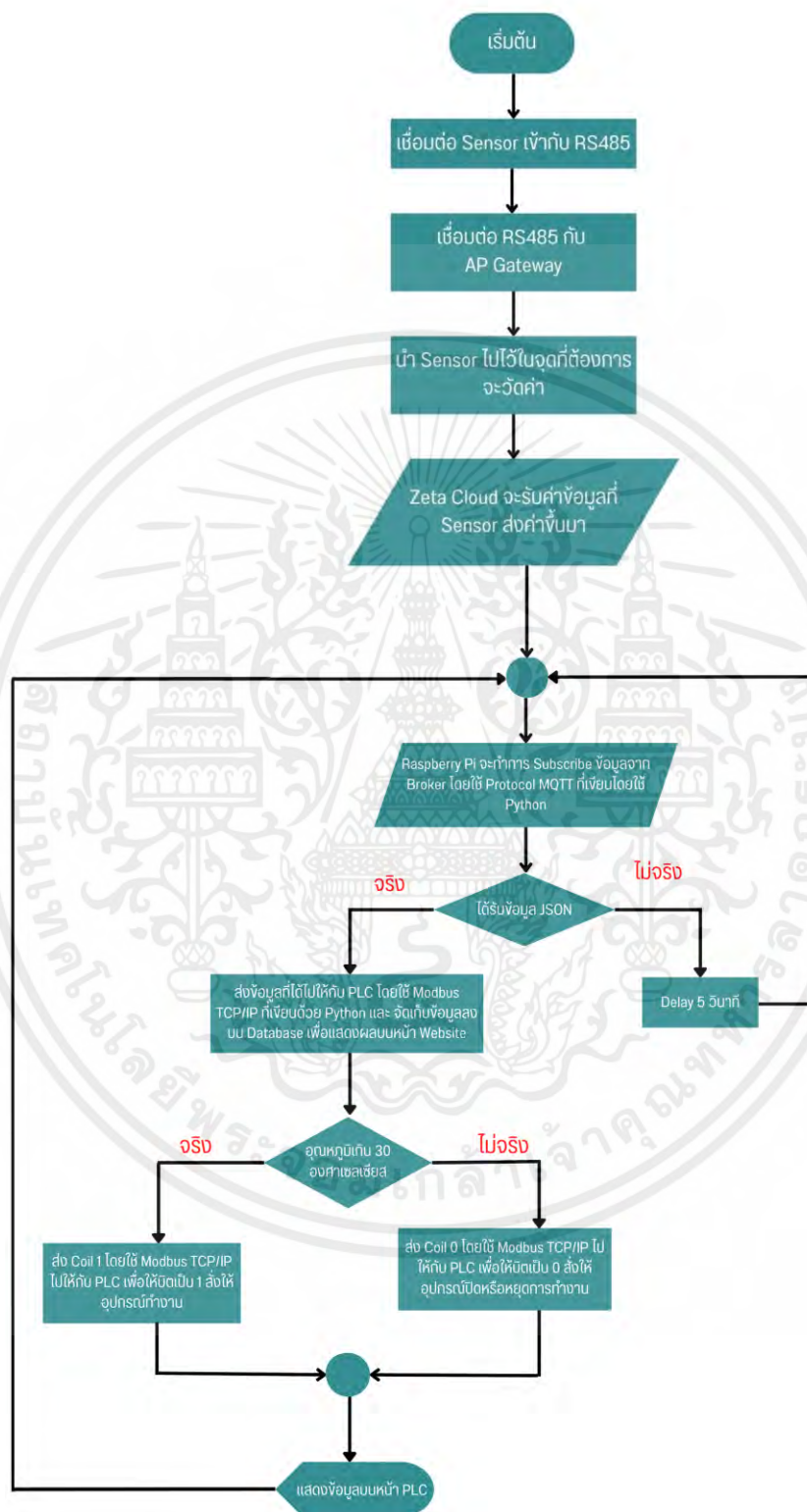
### 3.1.13 สรุปภาพรวมการทำงานของระบบ

ในขั้นตอนการทดสอบการทำงานของระบบเบื้องต้นของระบบผู้จัดทำได้ออกแบบขั้นตอนการทำงาน ของระบบ โดยมีรายละเอียดดังนี้ เมื่อ Sensor ทำทำการวัดค่าข้อมูลมาได้ RS485 จะทำการส่งข้อมูลไปยัง AP Gateway ขึ้นสู่ระบบ Zeta Cloud หลังจากนั้น Raspberry Pi จะทำการ Subscribe บิตข้อมูลและทำการส่งข้อมูลที่ได้จากการ Subscribe ไปยัง PLC Controller ที่เชื่อมต่อกันด้วยระบบ LAN จากนั้น PLC Controller จะนำข้อมูลที่ได้มาทำการประมวลผลและควบคุมการทำงานของอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller ในอีกส่วนหนึ่ง Raspberry Pi จะทำการแปลงบิตข้อมูลที่ได้เป็นตัวเลขฐานสิบมาบันทึกไว้บน Database ของ MySQL Server หลังจากนั้นจึงนำข้อมูลต่างๆมาแสดงผลบน Website ที่ได้สร้างขึ้น มีแผนผังภาพรวมการทำงานของระบบแสดงดังรูปที่ 3.81

ซึ่งในการออกแบบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าที่สมบูรณ์แล้วจะมีลักษณะและขั้นตอนในการทำงานที่เหมือนเดิมทุกประการแต่จะทำการปรับการใช้งานอุปกรณ์ให้เหมาะสมกับการทำงานมากขึ้น โดยจะใช้งาน OpenWrt Router แทน Raspberry Pi ซึ่งจะยังคงทำหน้าที่ต่างๆเช่นเดียวกับ Raspberry Pi ดังที่กล่าวไว้ข้างต้น และจะมีการแสดงการทำงานของระบบนี้ โดยทำการจำลองเป็นระบบปลุกต้นไม้อย่างง่ายซึ่งจะมีการเพิ่มการใช้งานอุปกรณ์ Output เป็น 3 อุปกรณ์ได้แก่ ไฟ LED สำหรับปลุกต้นไม้ พัดลมระบายอากาศ และ ป้อนน้ำ โดยมีเงื่อนไขในการควบคุมการทำงานของอุปกรณ์ Output ดังนี้

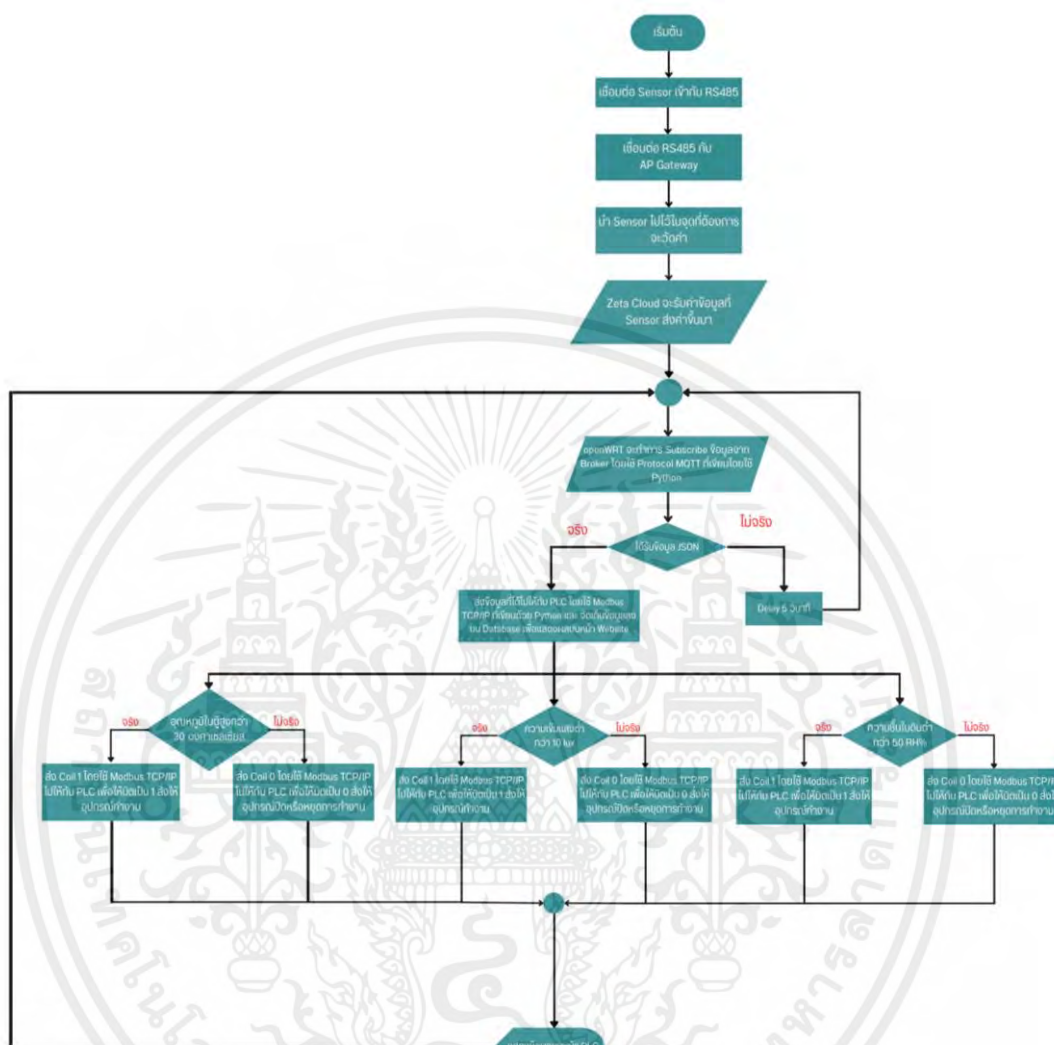
1. เมื่ออุณหภูมิที่วัดได้ภายในตู้มีค่ามากกว่า 30 องศาเซลเซียส พัดลมระบายอากาศจะถูกควบคุมให้ทำงาน
2. เมื่อความสว่างที่วัดได้มีค่าต่ำกว่า 10 ลักซ์ ไฟ LED สำหรับปลุกต้นไม้จะถูกควบคุมให้ทำงาน
3. เมื่อความชื้นในดินที่วัดได้มีค่าต่ำกว่า 50 RH% ป้อนน้ำจะถูกควบคุมให้ทำงาน

โดยแผนผังภาพรวมการทำงานของระบบสามารถแสดงได้ดังรูปที่ 3.81



รูปที่ 3.80 แผนผังแสดงการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าในขั้นตอนการทดลองการทำงาน  
 งานของระบบเบื้องต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.81 แผนผังแสดงการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.2 เครื่องมือที่ใช้ในการทดลอง

ในปฏิญานิพนธ์นี้ มีอุปกรณ์และเครื่องมือที่ใช้ในการทดลอง ดังนี้

### 3.2.1 PLC Controller

PLC Controller แสดงดังรูปที่ 2.12 เป็นอุปกรณ์ควบคุมการทำงานของเครื่องจักรหรือกระบวนการทำงานต่างๆ โดยภายในมี Microprocessor เป็นสมองสั่งการที่สำคัญ PLC จะมีส่วนที่เป็นอินพุตและเอาต์พุตที่สามารถต่อออกไปใช้งานได้ โดยปกติตัวตรวจวัดต่างๆจะต่อเข้ากับอินพุตส่วนเอาต์พุตจะใช้ต่อออกไปควบคุมการทำงานของอุปกรณ์ ในปฏิญานิพนธ์นี้ใช้งาน PLC Controller รุ่น LOGO! 230 RCE ซึ่งสามารถใช้งานไฟฟ้ากระแสสลับ 220 V เป็นไฟเลี้ยงให้กับ PLC Controller ได้

### 3.2.2 ZETA 485 Transceiver

ZETA 485 Transceiver หรือ RS485 แสดงดังรูปที่ 2.13 เป็นโมดูลที่ใช้พลังงานต่ำที่ใช้อินเทอร์เฟซ RS-485 (Modbus RTU) เพื่อรวบรวมข้อมูลหรืออ่านค่าจาก Sensor และส่งข้อมูลขึ้น Zeta Cloud โดยส่งข้อมูลแบบไร้สายที่ใช้งานความถี่ในย่าน 920-925 MHz ในการส่งข้อมูล

### 3.2.3 Raspberry Pi

ในปฏิญานิพนธ์นี้ใช้งาน Raspberry Pi 4 ดังรูปที่ 2.19 ซึ่งเป็นเครื่องคอมพิวเตอร์ขนาดเล็กที่มีขนาดประมาณบัตรเครดิต รองรับการเขียนภาษาคอมพิวเตอร์ (Programming Language) อย่าง Scratch, HTML, CSS และภาษา Python ซึ่งเป็นภาษาคอมพิวเตอร์หลักๆ สามารถเชื่อมต่อกับ หน้าจอทั่วไปเพื่อแสดงผลโดยใช้สาย HDMI และมี USB port สำหรับการเชื่อมต่อเมาส์และคีย์บอร์ดได้ โดยสามารถใช้ Mini USB เชื่อมต่อเข้ากับคอมพิวเตอร์หรือ Adapter เพื่อเป็นไฟเลี้ยงให้กับ Raspberry Pi

### 3.2.4 Xiaomi Wi-Fi Range Extender

ในปฏิญานิพนธ์นี้ใช้งาน Xiaomi Wi-Fi Range Extender AC1200 ดังรูปที่ 2.22 เป็น Router ที่จะนำมาติดตั้ง Firmware OpenWrt เพื่อให้ Router สามารถรัน Python Script ได้ และสามารถใช้งาน Modbus TCP/IP ในการส่งข้อมูลให้กับ PLC Controller ได้

### 3.2.5 Router Zyxel

ในปฏิญญาฉบับนี้ใช้งาน Router Zyxel P-660HN-T1A แสดงดังรูปที่ 3.82 เป็น Layer 2 Switch เพื่อใช้สำหรับการเชื่อมต่อคอมพิวเตอร์ OpenWrt Router และ PLC Controller เข้าด้วยกันผ่านระบบ LAN



รูปที่ 3.82 Router Zyxel

### 3.2.6 RS-485 Soil Temperature and Humidity Sensor

Soil Temperature and Humidity Sensor แสดงดังรูปที่ 2.17 เป็นเซ็นเซอร์ที่ใช้ในการวัดอุณหภูมิในดินตั้งแต่ -40 องศา ถึง 80 องศา และวัดความชื้นในดินตั้งแต่ 0 ถึง 100 เปอร์เซ็นต์ โดยการนำอุปกรณ์ส่วนที่เป็นโพรบวัดค่า 3 ขา ปักลงไปในดิน โดยอุปกรณ์นี้สามารถเชื่อมต่อกับอุปกรณ์อื่นๆได้โดยการส่งข้อมูลแบบ RS-485

### 3.2.7 Light Temperature and Humidity Sensor

Light Temperature and Humidity Sensor แสดงดังรูปที่ 2.18 เป็นเซ็นเซอร์ที่ใช้ในการวัดค่าความเข้มแสงตั้งแต่ 0 ถึง 200,000 ลักซ์ วัดอุณหภูมิในอากาศตั้งแต่ -20 ถึง 60 องศา เซลเซียส และวัดความชื้นในอากาศตั้งแต่ 0 ถึง 80 เปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.8 Temp Humid Sensor

Temp Humid Sensor แสดงดังรูปที่ 2.15 เป็น Sensor ระบบ wireless ที่ใช้สัญญาณในย่านความถี่ 920-925 MHz ในการส่งข้อมูล ใช้สำหรับวัดค่าอุณหภูมิและความชื้นของอากาศโดยรอบ

### 3.2.9 AP Gateway

AP Gateway แสดงดังรูปที่ 2.14 เป็นเกตเวย์ระหว่าง ZETA wireless network และ ZETA Server การโต้ตอบทั้งหมดระหว่าง ZETA wireless network และ ZETA Server จะทำผ่าน AP โดย AP สื่อสารกับอุปกรณ์เครือข่าย ZETA ที่เหลือผ่านโปรโตคอล ZETA และสื่อสารกับ ZETA Server ผ่านโปรโตคอล IP มาตรฐาน ZETA

### 3.2.10 Mote

Mote แสดงดังรูปที่ 2.16 เป็นอุปกรณ์สำหรับขยายสัญญาณซึ่งสามารถสร้างโครงข่ายประเภท tree ได้สูงสุด 4 ชั้น เพิ่มความครอบคลุมของ AP อย่างมีประสิทธิภาพ พร้อมการครอบคลุมพื้นที่อับสัญญาณ และป้องกันความแออัดของข้อมูล

### 3.2.11 หลอดไฟ

ในปริญญานิพนธ์นี้ใช้หลอดไฟชนิดหลอดไส้ 100 W ซึ่งสามารถใช้งานกับไฟฟ้ากระแสสลับ 220 V ได้ เป็นอุปกรณ์ Output สำหรับเชื่อมต่อกับ PLC Controller เพื่อใช้ในการทดสอบการทำงานเบื้องต้น

### 3.2.12 พัดลมระบายอากาศ

ในปฏิญญาฉบับนี้ใช้พัดลมระบายอากาศซึ่งใช้งานกับไฟฟ้ากระแสสลับ 220 V ดังรูปที่ 3.83 เป็นอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller



รูปที่ 3.83 พัดลมระบายอากาศ

### 3.2.13 ปั๊มน้ำ

ในปฏิญญาฉบับนี้ใช้ปั๊มน้ำซึ่งใช้งานกับไฟฟ้ากระแสสลับ 220 V ดังรูปที่ 3.84 เป็นอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller



รูปที่ 3.84 ปั๊มน้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.14 ไฟ LED สำหรับปลุกต้นไม้

ในปฏิญานิพนธ์นี้ใช้ไฟ LED สำหรับปลุกต้นไม้ ซึ่งใช้งานกับไฟฟ้ากระแสสลับ 220 V ดังรูปที่ 3.85 เป็นอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller



รูปที่ 3.85 ไฟ LED สำหรับปลุกต้นไม้

### 3.2.15 PL2303TA USB To TTL RS232

ในปฏิญานิพนธ์นี้ใช้งานอุปกรณ์ PL2303TA USB To TTL RS232 ดังรูปที่ 2.20 สำหรับเชื่อมต่อ Console Port ของ Router

### 3.2.16 EEPROM Flash BIOS

ในปฏิญานิพนธ์นี้ใช้งาน EEPROM Flash BIOS รุ่น CH431A ดังรูปที่ 2.21 สำหรับ Flash SPI Flash Memory ของ Router

### 3.2.17 FLUKE MODEL 51 II HANDHELD PORTABLE DIGITAL PROBE THERMOMETER

FLUKE MODEL 51 II HANDHELD PORTABLE DIGITAL PROBE THERMOMETER แสดงดังรูปที่ 3.86 เป็นอุปกรณ์สำหรับวัดค่าอุณหภูมิที่ผ่านการสอบเทียบมาแล้ว ใช้สำหรับการตรวจสอบความแม่นยำในการวัดค่าอุณหภูมิของ Sensor ที่ใช้ในปริญญานิพนธ์นี้



รูปที่ 3.86 Fluke Model 51 II Handheld Portable Digital Probe Thermometer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.18 โปรแกรม LOGO! Soft Comfort V8.3

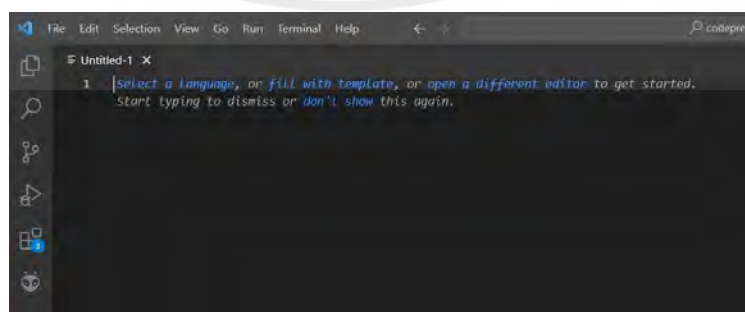
โปรแกรม LOGO! Soft Comfort V8.3 แสดงดังรูปที่ 3.38 เป็นโปรแกรมที่ใช้สำหรับการออกแบบโปรแกรมการทำงานให้กับ PLC Controller ใช้ในการตั้งค่าต่างๆให้กับ PLC Controller และใช้ในการทดสอบการทำงานของโปรแกรมเบื้องต้นก่อนที่จะทำการอัปโหลดไปยัง PLC Controller



รูปที่ 3.87 โปรแกรม LOGO! Soft Comfort V8.3

### 3.2.19 โปรแกรม Visual Studio Code

โปรแกรม Visual Studio Code แสดงดังรูปที่ 3.39 เป็นโปรแกรม Open Source ที่ออกแบบมาให้ใช้งานได้ทั้งบน Windows, Linux และ MacOS ใช้งานเป็น Text Editor หรือ IDE ที่สำหรับใช้ในการเขียนโปรแกรม โดยในปริญญานิพนธ์นี้จะใช้งานภาษา Python ภาษา JavaScript ภาษา HTML ภาษา CSS และภาษา PHP



รูปที่ 3.88 โปรแกรม Visual Studio Code

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.20 โปรแกรม MySQL Workbench

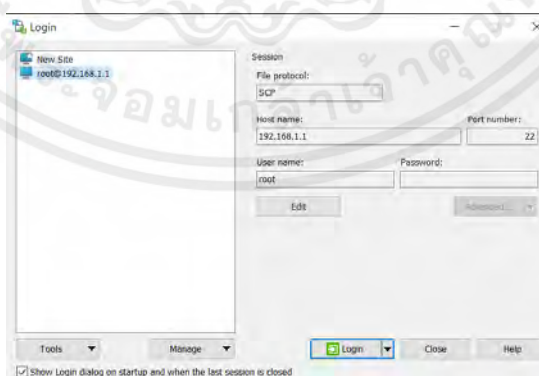
โปรแกรม MySQL Workbench แสดงดังรูปที่ 3.89 เป็นโปรแกรมที่ใช้ในการพัฒนา ส่วนของฐานข้อมูล ใช้สำหรับจัดการฐานข้อมูล MySQL Server ของค่าย MySQL ซึ่งประกอบด้วยไป การทำงานหลักๆด้วยกันทั้งหมด 3 อย่าง คือ SQL Development, Data Modeling และ Server Administration



รูปที่ 3.89 โปรแกรม MySQL Workbench

### 3.2.21 โปรแกรม WinSCP

โปรแกรม WinSCP แสดงดังรูปที่ 3.90 เป็นโปรแกรมที่ใช้ในการใช้งานโปรโตคอล SCP ในการโอนย้ายไฟล์ Python Script ที่ได้ทำการเขียนไว้ในคอมพิวเตอร์ไปยัง OpenWrt Router

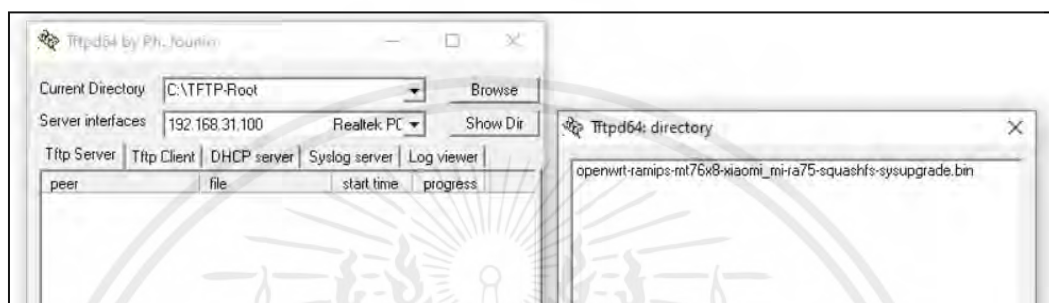


รูปที่ 3.90 โปรแกรม WinSCP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.22 โปรแกรม Tftpd64

โปรแกรม Tftpd64 แสดงดังรูปที่ 3.91 เป็นโปรแกรมที่ใช้สำหรับใช้งานโปรโตคอล TFTP ในการโอนย้ายไฟล์ Firmware OpenWrt จากคอมพิวเตอร์ซึ่งเป็น TFTP Sever ไปสู่ Router ซึ่งเป็น TFTP Client



รูปที่ 3.91 โปรแกรม Tftpd64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 การจัดเก็บผลการทดลอง

#### 3.3.1 การทดสอบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud

ทำการทดสอบการเชื่อมต่อ Soil Temperature and Humidity Sensor เข้ากับ RS485 และทำการทดสอบการส่งข้อมูลค่าอุณหภูมิและความชื้นไปยัง Zeta Cloud

- 1) ทดสอบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud ในสภาพแวดล้อมที่มีอุณหภูมิและความชื้นต่ำ
- 2) ทดสอบการทดสอบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud โดยทำการเพิ่มอุณหภูมิและความชื้น

#### 3.3.2 การทดสอบการรับข้อมูลของ Sensor ด้วย Raspberry Pi

ทำการทดสอบการใช้ Raspberry Pi ในการ Subscribe ข้อมูลอุณหภูมิและความชื้นที่วัดได้จาก Sensor จาก Broker โดยเขียนด้วยโปรแกรม Python

- 1) ทดสอบการรับข้อมูลของ Raspberry Pi ในสภาพแวดล้อมที่มีอุณหภูมิและความชื้นต่ำ
- 2) ทดสอบการรับข้อมูลของ Raspberry Pi โดยทำการเพิ่มอุณหภูมิและความชื้น

#### 3.3.3 การทดสอบการส่งข้อมูลจาก Raspberry Pi ไปยัง PLC Controller

ทำการทดสอบการใช้ Raspberry Pi ในการส่งข้อมูล ไปยัง PLC Controller ผ่านโปรโตคอล Modbus TCP/IP

- 1) ทดสอบการส่งข้อมูลประเภท Coil
- 2) ทดสอบการส่งข้อมูลประเภท Register

#### 3.3.4 การทดสอบการส่งข้อมูลจาก Raspberry Pi ไปยัง ฐานข้อมูล

ทำการทดสอบโดยการนำข้อมูลที่ได้จากการ Subscribe และนำบิต Data ที่ได้มาทำการแปลงเป็นฐาน 10 รวมถึงค่าต่างๆและสถานะของการทำงาน โดยส่งข้อมูลไปจัดเก็บไว้บนฐานข้อมูลที่ได้เขียนโดยใช้โปรแกรม Python โดยในที่นี้ใช้ MySQL Workbench เป็นฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.5 การทดสอบการควบคุมอุปกรณ์ Output ด้วย PLC Controller

ทำการทดสอบการใช้งาน PLC Controller ในการควบคุมอุปกรณ์ Output โดยในการทดสอบนี้จะใช้หลอดไฟเป็นอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller ซึ่งจะทำการทดสอบโดยให้ PLC Controller ควบคุมการ ปิด-เปิด หลอดไฟ

### 3.3.6 การทดสอบการทำงานเบื้องต้นของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

ทำการทดสอบการทำงานเบื้องต้นของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าโดยทำการเชื่อมต่ออุปกรณ์ต่างๆทั้งหมดตามที่ได้ออกแบบไว้สำหรับการทดสอบเบื้องต้นซึ่งจะมีการใช้งาน Sensor เพียงตัวเดียว มีอุปกรณ์ Output เพียงตัวเดียว และใช้ Raspberry Pi ในการรับและส่งข้อมูลซึ่งทำการทดสอบการทำงานร่วมกันโดยกำหนดเงื่อนไขให้อุปกรณ์ Output ทำงานเมื่ออุณหภูมิที่ Soil Temperature and Humidity Sensor วัดได้มีค่ามากกว่า 30 องศาเซลเซียส

- 1) ทำการทดสอบการทำงานของระบบโดยการจุ่ม Sensor ลงในน้ำร้อนซึ่งมีอุณหภูมิสูงกว่า 30 องศาเซลเซียส
- 2) ทำการทดสอบการทำงานของระบบโดยการจุ่ม Sensor ลงในน้ำเย็นซึ่งมีอุณหภูมิต่ำกว่า 30 องศาเซลเซียส

### 3.3.7 การทดสอบการใช้งาน RS485 ร่วมกับ Sensor มากกว่า 1 ตัว

ทำการทดสอบการใช้งาน RS485 ร่วมกับ Sensor 2 ตัว ได้แก่ RS-485 Soil Temperature and Humidity Sensor และ Light Temperature and Humidity Sensor

### 3.3.8 การทดสอบการรัน Python Script บน Raspberry Pi ที่ทำการลง Firmware OpenWrt

ทำการทดสอบการใช้งาน Firmware OpenWrt ในการรัน Python Script โดยทำการติดตั้งและทดสอบการทำงานบน Raspberry Pi ก่อนจะนำ Firmware นี้ไปใช้งานจริงกับ Router

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.9 การทดสอบการรับข้อมูลของ Sensor ด้วย OpenWrt Router

ทำการทดสอบการใช้ OpenWrt Router ในการ Subscribe ข้อมูลที่วัดได้จาก Sensor ทั้งสองตัว

- 1) ทดสอบการรับข้อมูลจาก RS-485 Soil Temperature and Humidity Sensor ด้วย OpenWrt Router
- 2) ทดสอบการรับข้อมูลจาก Light Temperature and Humidity Sensor ด้วย OpenWrt Router

### 3.3.10 การทดสอบการส่งข้อมูลจาก OpenWrt Router ไปยัง PLC Controller

ทำการทดสอบการใช้ OpenWrt Router ในการส่งข้อมูลที่ได้รับได้จาก Sensor ไปยัง PLC Controller ผ่านโปรโตคอล Modbus TCP/IP

- 1) ทดสอบการส่งข้อมูลที่ได้รับได้จาก RS-485 Soil Temperature and Humidity Sensor ไปยัง PLC Controller ด้วย OpenWrt Router
- 2) ทดสอบการส่งข้อมูลที่ได้รับได้จาก Light Temperature and Humidity Sensor ไปยัง PLC Controller ด้วย OpenWrt Router

### 3.3.11 การทดสอบการส่งข้อมูลจาก OpenWrt Router ไปยัง ฐานข้อมูล

ทำการทดสอบการนำข้อมูลที่ได้จากการ Subscribe และนำบิต Data ที่ได้มาทำการแปลงเป็นฐาน 10 รวมถึงค่าต่างๆและสถานะของการทำงาน โดยส่งข้อมูลไปจัดเก็บไว้บนฐานข้อมูลที่ได้เขียนโดยใช้โปรแกรม Python โดยในที่นี้ใช้ MySQL Workbench เป็นฐานข้อมูล

### 3.3.12 การทดสอบการแสดงผลข้อมูลบนเว็บไซต์

ทำการทดสอบการแสดงผลข้อมูลบนเว็บไซต์ตามเงื่อนไขการใช้งานที่ได้กำหนดไว้โดยแบ่งออกเป็น 4 กรณีตามการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) กรณีที่เข้าใช้งานแบบ Admin หลัก จะสามารถเห็นข้อมูลของทั้งสอง Sensor และสามารถเห็นข้อมูลของหน้าย่อยทั้ง 5 หน้า ในเว็บไซต์ได้ทั้งหมด

2) กรณีที่เข้าใช้งานแบบ Admin บริษัทที่ 1 จะสามารถเห็นข้อมูลของหน้าย่อยทั้ง 5 หน้า ในเว็บไซต์ได้ทั้งหมดเช่นเดียวกันแต่จะสามารถเห็นข้อมูลจาก RS-485 Soil Temperature and Humidity Sensor เท่านั้น

3) กรณีที่เข้าใช้งานแบบ Admin บริษัทที่ 2 จะสามารถเห็นข้อมูลของหน้าย่อยทั้ง 5 หน้า ในเว็บไซต์ได้ทั้งหมดเช่นเดียวกันแต่จะสามารถเห็นข้อมูลจาก จาก Light Temperature and Humidity Sensor เท่านั้น

4) กรณีที่เข้าใช้งานโดย User ทั่วไป จะสามารถเห็นข้อมูลของหน้าย่อยในเว็บไซต์ได้เพียง 3 หน้า ไม่สามารถใช้งานหน้า History และ Graphs ได้

### 3.3.13 การทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

ทำการทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าโดยทำการเชื่อมต่ออุปกรณ์ต่างๆทั้งหมดตามที่ได้ทำการออกแบบไว้ โดยจะมีการใช้งาน OpenWrt Router ในการรับและส่งข้อมูลแทน Raspberry Pi มีมีการใช้งาน RS485 ร่วมกับ Sensor 2 ตัว และมีอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller 3 อุปกรณ์ ซึ่งทำการทดสอบการทำงานร่วมกันของระบบตามเงื่อนไขที่ได้กำหนดไว้ให้อุปกรณ์ Output ทำงาน

- 1) กรณีที่อุณหภูมิที่วัดได้ภายในตู้มีค่าน้อยกว่าหรือเท่ากับ 30 องศาเซลเซียส
- 2) กรณีที่อุณหภูมิที่วัดได้ภายในตู้มีค่ามากกว่า 30 องศาเซลเซียส
- 3) กรณีที่ความสว่างที่วัดได้มีค่ามากกว่าหรือเท่ากับ 10 ลักซ์
- 4) กรณีที่ความสว่างที่วัดได้มีค่าน้อยกว่า 10 ลักซ์
- 5) เมื่อความชื้นในดินที่วัดได้มีค่ามากกว่าหรือเท่ากับ 50 RH%
- 6) เมื่อความชื้นในดินที่วัดได้มีค่าน้อยกว่า 50 RH%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.14 การทดสอบประสิทธิภาพความแม่นยำในการวัดข้อมูลของ Sensor

ทำการทดสอบความแม่นยำในการวัดค่าอุณหภูมิของ Sensor โดยเทียบอุณหภูมิที่ Sensor วัดได้กับอุปกรณ์วัดอุณหภูมิที่ผ่านการสอบเทียบมาแล้ว

### 3.3.15 การทดสอบประสิทธิภาพการส่งข้อมูลของ AP Gateway

เนื่องจาก AP Gateway มี heartbeat time อยู่ที่ 3 นาที (มีการส่งข้อมูลทุกๆ 3 นาที) จะทำการทดสอบโดยใช้ python code ทำการ subscribe MQTT ของ AP Gateway และทำการจับเวลาเป็นเวลา 60 นาที และดูว่าใน 60 นาที สามารถรับข้อมูลจาก AP Gateway ได้กี่ครั้งคิดเป็นประสิทธิภาพได้กี่เปอร์เซ็นต์ถ้าหากสามารถรับข้อมูลได้ 20 ครั้งใน 60 นาที แสดงว่าประสิทธิภาพการส่งข้อมูลของ AP Gateway อยู่ที่ 100% โดยในการทดลองนี้เริ่มจับเวลาและเริ่มทำการทดสอบที่เวลา 18:14 น. และจะหยุดการทดสอบที่เวลา 19:14 น.

### 3.3.16 การทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor

เนื่องจาก temp Humid sensor มี heartbeat time อยู่ที่ 3 นาที (มีการส่งข้อมูลทุกๆ 3 นาที) จะทำการทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor โดยใช้ python code ทำการ subscribe MQTT ของ Temp Humid sensor และทำการจับเวลาเป็นเวลา 30 นาที และดูว่าใน 30 นาที สามารถรับข้อมูลจาก Temp Humid sensor ได้กี่ครั้งคิดเป็นประสิทธิภาพได้กี่เปอร์เซ็นต์

- 1) ทำการทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor เมื่อ AP Gateway อยู่ในห้องภายในอาคาร (มีสิ่งกีดขวางมากกว่า) และไม่ได้ทำการใช้ Mote ขยายสัญญาณ
- 2) ทำการทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor เมื่อ AP Gateway อยู่ด้านบนคานฟ้าของอาคาร (มีสิ่งกีดขวางน้อยกว่า) และไม่ได้ทำการใช้ Mote ขยายสัญญาณ
- 3) ทำการทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor เมื่อ AP Gateway อยู่ด้านบนคานฟ้าของอาคาร (มีสิ่งกีดขวางน้อยกว่า) และทำการใช้ Mote ในการขยายสัญญาณ

## บทที่ 4

### ผลการทดลอง

ผู้จัดทำได้ทำการเก็บผลการทำงานของระบบโดยแบ่งการทดลองและจัดเก็บผลการทดลองเป็นส่วนๆ ดังต่อไปนี้

#### 4.1 การทดสอบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud

##### 4.1.1 ทดสอบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud ในสภาพแวดล้อมที่มีอุณหภูมิและความชื้นต่ำ

ทดสอบการวัดค่าอุณหภูมิและความชื้นและการส่งบิตข้อมูลไปยัง Zeta Cloud ด้วย Soil Temperature and Humidity Sensor ที่เชื่อมต่ออยู่กับ RS485 โดยทำการปัก Sensor ลงในดินทรายที่มีความชื้นเพียงเล็กน้อย เมื่อสภาพแวดล้อมมีอุณหภูมิและความชื้นต่ำจะได้ค่าบิตข้อมูลที่ Sensor ได้ทำการส่งมาผ่านเว็บไซต์ของทาง Zeta Cloud แสดงดังรูปที่ 4.1 พบว่าค่าอุณหภูมิและความชื้นที่ได้มีค่าเป็น 25.4 องศาเซลเซียส และ 17.5%RH ตามลำดับ ซึ่งมีค่าที่อยู่ในช่วงที่ใกล้เคียงกับความเป็นจริงของสภาพแวดล้อมในขณะนั้นซึ่งแสดงว่า Temperature and Humidity Sensor สามารถวัดค่าอุณหภูมิและความชื้นได้ และ RS485 สามารถส่งค่าข้อมูลที่ได้ไปยัง Zeta Cloud ได้

##### 4.1.2 ทดสอบการส่งข้อมูลจาก Sensor ไปยัง Zeta Cloud โดยทำการเพิ่มอุณหภูมิและความชื้น

ทดสอบการวัดค่าอุณหภูมิและความชื้นและการส่งบิตข้อมูลไปยัง Zeta Cloud ด้วย Soil Temperature and Humidity Sensor ที่เชื่อมต่ออยู่กับ RS485 โดยทำการจุ่ม Sensor ลงในน้ำร้อน เมื่อสภาพแวดล้อมมีอุณหภูมิและความชื้นที่เพิ่มขึ้นจากเดิมจะได้ค่าบิตข้อมูลที่ Sensor ได้ทำการส่งมาผ่านเว็บไซต์ของทาง Zeta Cloud แสดงดังรูปที่ 4.2 พบว่าค่าอุณหภูมิและความชื้นที่ได้มีค่าเพิ่มขึ้นเป็น 62.3 องศาเซลเซียส และ 63.6%RH ตามลำดับ ซึ่งมีค่าที่อยู่ในช่วงที่ใกล้เคียงกับความเป็นจริงของสภาพแวดล้อมในขณะนั้นซึ่งแสดงว่า Temperature and Humidity Sensor สามารถวัดค่าอุณหภูมิและความชื้นได้ และ RS485 สามารถส่งค่าข้อมูลที่ได้ไปยัง Zeta Cloud ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



010100af00fe	soil sensor:1 Humid:17.5%RH Temp:25.4°C	2023-11-11 17:10:15
--------------	---	---------------------

รูปที่ 4.1 บิตข้อมูลเมื่ออุณหภูมิและความชื้นมีค่าต่ำ



0101027c026f	soil sensor:1 Humid:63.6%RH Temp:62.3°C	2023-11-11 17:11:15
--------------	---	---------------------

รูปที่ 4.2 บิตข้อมูลเมื่อทำการเพิ่มอุณหภูมิและความชื้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 การทดสอบการรับข้อมูลของ Sensor ด้วย Raspberry Pi

### 4.2.1 ทดสอบการรับข้อมูลของ Raspberry Pi ในสภาพแวดล้อมที่มีอุณหภูมิและความชื้นต่ำ

ทดสอบการรับข้อมูลของ Raspberry Pi โดยทำการรัน Python code ที่ได้เขียนขึ้นสำหรับการใช้ MQTT Protocol ในการ Subscribe ข้อมูลอุณหภูมิและความชื้นที่ได้จาก Sensor โดยทำการ Subscribe Topic ของ RS485 พบว่าเมื่อสภาพแวดล้อมมีอุณหภูมิและความชื้นต่ำ Raspberry Pi สามารถ Subscribe ข้อมูลได้ แสดงดังรูปที่ 4.3 พบว่าบิตข้อมูลค่าอุณหภูมิและความชื้นที่ได้จากการ Subscribe มีค่าตรงกับค่าใน Zeta Cloud ที่ทำการดูผ่านเว็บไซต์

### 4.2.2 ทดสอบการรับข้อมูลของ Raspberry Pi โดยทำการเพิ่มอุณหภูมิและความชื้น

ทดสอบการรับข้อมูลของ Raspberry Pi โดยทำการรัน Python code ที่ได้เขียนขึ้นสำหรับการใช้ MQTT Protocol ในการ Subscribe ข้อมูลอุณหภูมิและความชื้นที่ได้จาก Sensor โดยทำการ Subscribe Topic ของ RS485 เมื่อสภาพแวดล้อมมีอุณหภูมิและความชื้นที่เพิ่มขึ้น Raspberry Pi สามารถ Subscribe ข้อมูลได้ แสดงดังรูปที่ 4.4 พบว่าบิตข้อมูลค่าอุณหภูมิและความชื้นที่ได้จากการ Subscribe มีค่าตรงกับค่าใน Zeta Cloud ที่ทำการดูผ่านเว็บไซต์

```

No any data is Queue!!
-----
----- processTask -----
json_get = {'companyCode': '980f70347bc145c8a0adb4a34a76b264', 'deviceType': '485 ...
CompanyCode : 980f70347bc145c8a0adb4a34a76b264
DeviceType : RS485_2
Data_Temp485 : 010100af00fe
Bit_Temp485 : 00fe
Temp485 : 25.40 Celsius
Bit Humid485 : 00af
Humid485 : 17.50 RH%
unix time : 1699697415
2023-11-11 17:10:15
Connected to PLC
Data sent successfully to address 1 : [254]
Data sent successfully to address 2 : [175]
send 0 to address 0
Transfer finished
----- processTask -----
No any data is Queue!!

```

รูปที่ 4.3 บิตข้อมูลที่ Subscribe ได้เมื่ออุณหภูมิมีค่าต่ำ

```

No any data is Queue!!
----- processTask -----
json_get = {'companyCode': '980f70347bc145c8a0adb4a34a76b264', '
CompanyCode : 980f70347bc145c8a0adb4a34a76b264
DeviceType : RS485_2
Data_Temp485 : 0101027c026f
Bit_Temp485 : 026f
Temp485 : 62.30 Celsius
Bit Humid485 : 027c
Humid485 : 63.60 RH%
unix time : 1699697475
2023-11-11 17:11:15
Connected to PLC
Data sent successfully to address 1 : [623]
Data sent successfully to address 2 : [636]
send 1 to address 0
Transfer finished
----- processTask -----
No any data is Queue!!

```

รูปที่ 4.4 บิตข้อมูลที่ Subscribe ได้เมื่อทำการเพิ่มอุณหภูมิ

## 4.3 การทดสอบการส่งข้อมูลจาก Raspberry Pi ไปยัง PLC Controller

### 4.3.1 ทดสอบการส่งข้อมูลประเภท Coil

ทดสอบการส่งข้อมูลจาก Raspberry Pi โดยทำการส่งข้อมูลประเภท Coil ไปยัง PLC Controller ผ่านโปรโตคอล Modbus TCP/IP โดยทำการทดลองส่งค่าเท่ากับ 1 ไปยัง PLC Controller ได้ผลการทดสอบแสดงดังรูปที่ 4.5 พบว่าค่าที่ฝั่งรับสามารถรับได้มีค่าเท่ากับ 1 แสดงดังรูปที่ 4.6 และเมื่อใช้โปรแกรม Wireshark ในการตรวจจับจะได้ Packet ข้อมูล แสดงดังรูปที่ 4.7 และ 4.8

```

Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 22:22:05) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\modbus\code\Coil.py =====
1
Transfer finished
>>>

```

รูปที่ 4.5 ผลการทดสอบการส่งข้อมูลประเภท Coil โดยมีค่าเท่ากับ 1

	Name	00000
0		1
1		0
2		0
3		0

รูปที่ 4.6 ค่าที่ฝั่งรับได้รับซึ่งมีค่าเท่ากับ 1

```

> Frame 4834: 58 bytes on wire (464 bits), 58 bytes captured (464 bits) on interface \Device\NPF_{Loopback}
> Null/Loopback
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
> Transmission Control Protocol, Src Port: 57180, Dst Port: 502, Seq: 1, Ack: 1, Len: 14
√ Modbus/TCP
  Transaction Identifier: 5837
  Protocol Identifier: 0
  Length: 8
  Unit Identifier: 1
√ Modbus
  .000 1111 = Function Code: Write Multiple Coils (15)
  Reference Number: 0
  Bit Count: 1
  Byte Count: 1
  Data: 01

```

รูปที่ 4.7 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบข้อมูล

```

0000  02 00 00 00 45 00 00 36 e9 42 40 00 80 06 00 00  ...E..6.B@...
0010  7f 00 00 01 7f 00 00 01 df 5c 01 f6 96 7b b7 3a  ...-...-...{.:
0020  0c 2e 86 22 50 18 27 f9 af 83 00 00 16 cd 00 00  ..."p.....
0030  00 08 01 0f 00 00 00 01 01 01  ...

```

รูปที่ 4.8 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบข้อมูล

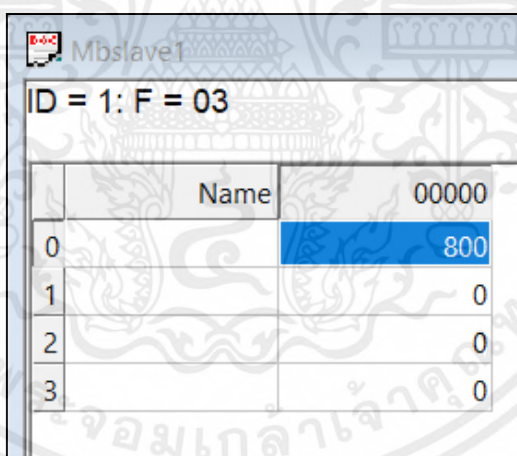
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3.2 ทดสอบการส่งข้อมูลประเภท Register

ทดสอบการส่งข้อมูลจาก Raspberry Pi โดยทำการส่งข้อมูลประเภท Register ไปยัง PLC Controller ผ่านโปรโตคอล Modbus TCP/IP โดยทำการทดลองส่งค่าเท่ากับ 800 ไปยัง PLC Controller ได้ผลการทดสอบแสดงดังรูปที่ 4.9 พบว่าค่าที่ฝั่งรับสามารถรับได้มีค่าเท่ากับ 1 แสดงดังรูปที่ 4.10 และเมื่อใช้โปรแกรม Wireshark ในการตรวจจับจะได้ Packet ข้อมูล แสดงดังรูปที่ 4.11 และ 4.12

```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 22:22:05) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\modbus\code\Register.py =====
Connected to PLC
Data sent successfully: [800]
>>>
```

รูปที่ 4.9 ผลการทดสอบการส่งข้อมูลประเภท Register โดยมีค่าเท่ากับ 800



Mbslave1	
ID = 1: F = 03	
	Name
	00000
0	800
1	0
2	0
3	0

รูปที่ 4.10 ค่าที่ฝั่งรับได้รับซึ่งมีค่าเท่ากับ 800

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

> Frame 5121: 59 bytes on wire (472 bits), 59 bytes captured (472 bits) on interface \Device\NPF_{Loopback}
> Null/Loopback
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
> Transmission Control Protocol, Src Port: 57205, Dst Port: 502, Seq: 1, Ack: 1, Len: 15
  Modbus/TCP
    Transaction Identifier: 1
    Protocol Identifier: 0
    Length: 9
    Unit Identifier: 0
  Modbus
    .001 0000 = Function Code: Write Multiple Registers (16)
    Reference Number: 0
    Word Count: 1
    Byte Count: 2
  > Register 0 (UINT16): 800

```

Modbus/TCP (mbtcp), 15 byte(s)

รูปที่ 4.11 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบจับข้อมูล

0000	02 00 00 00 45 00 00 37	ea 59 40 00 80 06 00 00	...E..7.Y@....
0010	7f 00 00 01 7f 00 00 01	df 75 01 f6 f7 b3 d3 7a	.....u.....z
0020	ef f6 bd 62 50 18 27 f9	0d b0 00 00 00 01 00 00	...bP'.....
0030	00 09 00 10 00 00 00 01	02 03 20	.....

รูปที่ 4.12 Packet ข้อมูลที่ได้จากการใช้ Wireshark ตรวจสอบจับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4 การทดสอบการส่งข้อมูลจาก Raspberry Pi ไปยังฐานข้อมูล

ทดสอบการส่งข้อมูลที่ได้จากการ Subscribe โดยใช้ MQTT Protocol โดยการนำค่าต่างๆที่ได้มาทำการจัดเก็บไว้ในฐานข้อมูล รวมถึงค่าบิตข้อมูลที่ได้รับจากเซ็นเซอร์มาทำการแปลงเป็นเลขฐาน 10 โดยการใช้ภาษา Python ในการทำงาน และมีการส่งสถานะการทำงานของอุปกรณ์ว่าเปิดหรือปิดเข้าไปในฐานข้อมูล โดยในการทดลองนี้จะทำเก็บค่า ID, CompanyCode, DeviceType, Data, Tempsoil, Humidsoil, Infosoil, Statuslight, Datetime และ Bitstatuslight ตามแต่ละคอลัมน์ที่ได้ทำการสร้างไว้ใน Table และจากการทดสอบจะพบว่าสามารถเก็บค่าข้อมูลต่างๆตามที่ต้องการลงในฐานข้อมูลได้อย่างครบถ้วน แสดงดังรูปที่ 4.13

ID	CompanyCode	DeviceType	Data	Tempsoil	Humidsoil	Infosoil
1	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100000105	26.10	0.00	Temp : 26.10 Celcius , Humid : 0.00 RH%
2	980f70347bc145c8a0adb4a34a76b264	RS485_2	010101b50149	32.90	43.70	Temp : 32.90 Celcius , Humid : 43.70 RH%
3	980f70347bc145c8a0adb4a34a76b264	RS485_2	01010000012c	30.00	0.00	Temp : 30.00 Celcius , Humid : 0.00 RH%
4	980f70347bc145c8a0adb4a34a76b264	RS485_2	010101ad0121	28.90	42.90	Temp : 28.90 Celcius , Humid : 42.90 RH%
5	980f70347bc145c8a0adb4a34a76b264	RS485_2	01010192014c	33.20	40.20	Temp : 33.20 Celcius , Humid : 40.20 RH%
6	980f70347bc145c8a0adb4a34a76b264	RS485_2	01010000012e	30.20	0.00	Temp : 30.20 Celcius , Humid : 0.00 RH%
7	980f70347bc145c8a0adb4a34a76b264	RS485_2	01010000011a	28.20	0.00	Temp : 28.20 Celcius , Humid : 0.00 RH%
8	980f70347bc145c8a0adb4a34a76b264	RS485_2	01010000010f	27.10	0.00	Temp : 27.10 Celcius , Humid : 0.00 RH%
9	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100000104	26.00	0.00	Temp : 26.00 Celcius , Humid : 0.00 RH%
10	980f70347bc145c8a0adb4a34a76b264	RS485_2	010101a30148	32.80	41.90	Temp : 32.80 Celcius , Humid : 41.90 RH%
11	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100960110	27.20	15.00	Temp : 27.20 Celcius , Humid : 15.00 RH%
12	980f70347bc145c8a0adb4a34a76b264	RS485_2	010101940154	34.00	40.40	Temp : 34.00 Celcius , Humid : 40.40 RH%
13	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100000134	30.80	0.00	Temp : 30.80 Celcius , Humid : 0.00 RH%
14	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100000120	28.80	0.00	Temp : 28.80 Celcius , Humid : 0.00 RH%
15	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100150115	27.70	2.10	Temp : 27.70 Celcius , Humid : 2.10 RH%
16	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100140110	27.20	2.00	Temp : 27.20 Celcius , Humid : 2.00 RH%
17	980f70347bc145c8a0adb4a34a76b264	RS485_2	0101000000f2	24.20	0.00	Temp : 24.20 Celcius , Humid : 0.00 RH%
18	980f70347bc145c8a0adb4a34a76b264	RS485_2	0101000000f2	24.20	0.00	Temp : 24.20 Celcius , Humid : 0.00 RH%
19	980f70347bc145c8a0adb4a34a76b264	RS485_2	0101000000f4	24.40	0.00	Temp : 24.40 Celcius , Humid : 0.00 RH%
20	980f70347bc145c8a0adb4a34a76b264	RS485_2	010101b70148	32.80	43.90	Temp : 32.80 Celcius , Humid : 43.90 RH%
21	980f70347bc145c8a0adb4a34a76b264	RS485_2	010100000125	29.30	0.00	Temp : 29.30 Celcius , Humid : 0.00 RH%
22	980f70347bc145c8a0adb4a34a76b264	RS485_2	010102730293	65.90	62.70	Temp : 65.90 Celcius , Humid : 62.70 RH%
23	980f70347bc145c8a0adb4a34a76b264	RS485_2	010102ae0085	13.30	68.60	Temp : 13.30 Celcius , Humid : 68.60 RH%
24	980f70347bc145c8a0adb4a34a76b264	RS485_2	010102bb0251	59.30	69.90	Temp : 59.30 Celcius , Humid : 69.90 RH%
25	980f70347bc145c8a0adb4a34a76b264	RS485_2	01010000023f	57.50	0.00	Temp : 57.50 Celcius , Humid : 0.00 RH%
26	980f70347bc145c8a0adb4a34a76b264	RS485_2	010101c800a9	16.90	45.60	Temp : 16.90 Celcius , Humid : 45.60 RH%

Tempair	Humidar	light	Infoair	Statuslight	Statusfan	Statuspump	Datetime	Bitstatuslight	Bitstatusfan	Bitstatuspump
0	0	0	0	OFF	0	0	2023-11-10 18:04:58	0	0	0
0	0	0	0	ON	0	0	2023-11-10 18:05:58	1	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:06:58	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:07:58	0	0	0
0	0	0	0	ON	0	0	2023-11-10 18:08:58	1	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:09:58	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:10:58	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:11:58	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:15:58	0	0	0
0	0	0	0	ON	0	0	2023-11-10 18:16:58	1	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:41:58	0	0	0
0	0	0	0	ON	0	0	2023-11-10 18:42:58	1	0	0
0	0	0	0	ON	0	0	2023-11-10 18:43:57	1	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:44:58	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:45:58	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 18:46:58	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 20:57:03	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 20:58:03	0	0	0
0	0	0	0	OFF	0	0	2023-11-10 20:59:03	0	0	0
0	0	0	0	ON	0	0	2023-11-10 21:00:03	1	0	0
0	0	0	0	OFF	0	0	2023-11-10 21:01:03	0	0	0
0	0	0	0	ON	0	0	2023-11-10 21:37:03	1	0	0
0	0	0	0	OFF	0	0	2023-11-10 21:38:03	0	0	0
0	0	0	0	ON	0	0	2023-11-10 21:52:47	1	0	0
0	0	0	0	ON	0	0	2023-11-10 21:53:48	1	0	0
0	0	0	0	OFF	0	0	2023-11-10 21:54:47	0	0	0

รูปที่ 4.13 ผลที่ได้จากการเก็บข้อมูลลงในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5 การทดสอบการควบคุมอุปกรณ์ Output ด้วย PLC Controller

ทดสอบการใช้งาน PLC Controller ควบคุมการทำงานของอุปกรณ์ Output ซึ่งเป็นอุปกรณ์ที่ใช้ไฟฟ้าไม่เกิน 220 V โดยในการทดลองนี้ใช้เป็นหลอดไฟชนิดหลอดไส้ต่อเข้ากับช่อง Output Q1 ของ PLC Controller โดยทำการส่งค่า Coil เท่ากับ 0 เป็นการสั่งสั่งให้ Relay Switch ของ Output Q1 ที่เชื่อมต่ออยู่กับหลอดไฟอยู่ในสถานะ Open circuit พบว่าหลอดไฟจะดับ แสดงดังรูปที่ 4.14 และเมื่อทำการส่งค่า Coil เท่ากับ 1 เป็นการสั่งสั่งให้ Relay Switch ของ Output Q1 ที่เชื่อมต่ออยู่กับหลอดไฟอยู่ในสถานะ Short circuit พบว่าหลอดไฟจะติด แสดงดังรูปที่ 4.15



รูปที่ 4.14 Relay Switch ของ Output Q1 ที่เชื่อมต่ออยู่กับหลอดไฟอยู่ในสถานะ Open circuit



รูปที่ 4.15 Relay Switch ของ Output Q1 ที่เชื่อมต่ออยู่กับหลอดไฟอยู่ในสถานะ Short circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.6 การทดสอบการทำงานเบื้องต้นของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

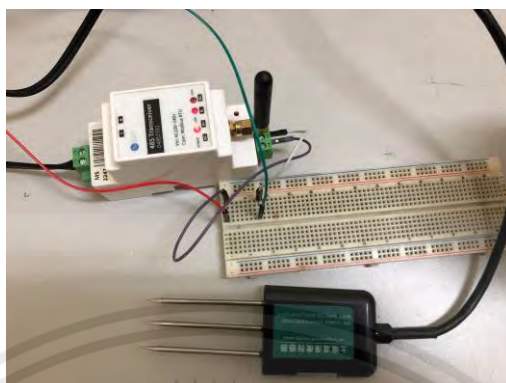
ทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าโดยทำการเชื่อมต่ออุปกรณ์ต่างๆ ทั้งหมดได้แก่ ทำการติดตั้ง AP Gateway และอุปกรณ์กระจายสัญญาณต่างๆ ทำการเชื่อมต่อ Soil Temperature and Humidity Sensor เข้ากับ RS485 สำหรับส่งข้อมูลค่าอุณหภูมิและความชื้นที่ Sensor วัดได้ ไปยัง AP Gateway เพื่อส่งข้อมูลต่อไปยัง Zeta Cloud หลังจากนั้นทำการเชื่อมต่อ PLC Controller เข้ากับ Raspberry Pi และ อุปกรณ์ Output ซึ่งก็คือหลอดไฟ การเชื่อมต่ออุปกรณ์ต่างๆแสดงดังรูปที่ 4.16 ถึง 4.19 และทำการทดสอบการทำงานร่วมกันโดยให้ Raspberry Pi Subscribe ข้อมูลค่าอุณหภูมิและความชื้นและส่งข้อมูลต่อไปยัง PLC Controller ด้วยโปรโตคอล Modbus TCP/IP และกำหนดโปรแกรมการทำงานบน PLC Controller โดยมีเงื่อนไขควบคุมให้อุปกรณ์ Output ทำงานเมื่ออุณหภูมิที่ Soil Temperature and Humidity Sensor วัดได้มีค่ามากกว่า 30 องศาเซลเซียส

### 4.6.1 ทดสอบการทำงานของระบบโดยการจุ่ม Sensor ลงในน้ำร้อนซึ่งมีอุณหภูมิสูงกว่า 30 องศาเซลเซียส

ทดสอบการทำงานในกรณีที่ค่าอุณหภูมิมีค่าสูงกว่า 30 องศาเซลเซียส พบว่า PLC Controller จะควบคุม ให้ Relay switch ของ Output Q1 อยู่ในสภาวะ Short circuit ทำให้หลอดไฟติดและหน้าจอแสดงผลของ PLC Controller จะเปลี่ยนเป็นสีแดงพร้อมกับแสดงค่าอุณหภูมิกับความชื้นและสถานะ ON แสดงดังรูปที่ 4.20 ซึ่งเป็นไปตามเงื่อนไขที่กำหนด

### 4.6.2 ทดสอบการทำงานของระบบโดยการจุ่ม Sensor ลงในน้ำเย็นซึ่งมีอุณหภูมิต่ำกว่า 30 องศาเซลเซียส

ทดสอบการทำงานในกรณีที่ค่าอุณหภูมิมิมีค่าต่ำกว่า 30 องศาเซลเซียส พบว่า PLC Controller จะควบคุม ให้ Relay switch ของ Output Q1 อยู่ในสภาวะ Open circuit ทำให้หลอดไฟดับ และหน้าจอแสดงผลของ PLC Controller จะเปลี่ยนเป็นสีส้มพร้อมกับแสดงค่าอุณหภูมิกับความชื้นและสถานะ OFF แสดงดังรูปที่ 4.21 ซึ่งเป็นไปตามเงื่อนไขที่กำหนด



รูปที่ 4.16 การเชื่อมต่อ Temperature and Humidity Sensor เข้ากับ RS485

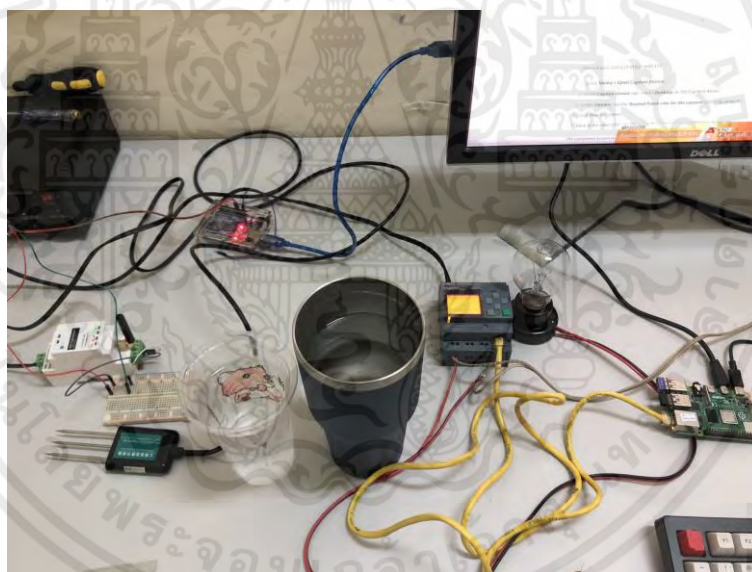


รูปที่ 4.17 การเชื่อมต่อ PLC Controller เข้ากับ Raspberry Pi และหลอดไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 น้ำเย็นและน้ำร้อนสำหรับการทดสอบ



รูปที่ 4.19 อุปกรณ์ทั้งหมดที่ใช้ในการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 ทำการทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้ากับน้ำร้อน



รูปที่ 4.21 ทำการทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้ากับน้ำเย็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.7 การทดสอบการใช้งาน RS485 ร่วมกับ Sensor มากกว่า 1 ตัว

ทำการติดตั้ง AP Gateway และอุปกรณ์กระจายสัญญาณต่างๆ และทำการเชื่อมต่อ Soil Temperature and Humidity Sensor สำหรับวัดค่าอุณหภูมิและความชื้นในดิน และ Light Temperature and Humidity Sensor สำหรับวัดค่าอุณหภูมิและความชื้นในอากาศ โดยทำการเชื่อมต่อ Sensor ทั้งสองแบบอนุกรมตามวงจรที่ได้ทำการออกแบบไว้เข้ากับ RS485 เพื่อส่งข้อมูลค่าที่ Sensor วัดได้ ไปยัง AP Gateway และส่งข้อมูลต่อไปยัง Zeta Cloud เมื่อทำการตรวจสอบค่าที่ได้บนเว็บไซต์ของ Zeta Cloud พบว่า RS485 สามารถทำงานร่วมกับ Sensor ทั้งสองตัวได้และสามารถส่งค่าของแต่ละ Sensor สลับกันตามเวลาที่ได้ตั้งค่าไว้ซึ่ง Sensor 1 คือ Light Temperature and Humidity Sensor ส่วน Sensor 2 คือ Soil Temperature and Humidity Sensor ได้ผลแสดงดังรูปที่ 4.22

010201cc01130068	Air sensor:2 Humid:46.0%RH Temp:27.5°C Light:10.4lux	2024-02-21 17:40:28
0101017400f10000	Soil sensor:1 Humid:37.2%RH Temp:24.1°C Light:0.0lux	2024-02-21 17:40:22

รูปที่ 4.22 ค่าที่วัดได้จาก Soil Temperature and Humidity Sensor Light Temperature and Humidity Sensor

#### 4.8 การทดสอบการรัน Python Script บน Raspberry Pi ที่ทำการลง Firmware OpenWrt

หลังจากที่ทำการติดตั้ง AP Gateway และอุปกรณ์กระจายสัญญาณต่างๆ และทำการเชื่อมต่อ Soil Temperature and Humidity Sensor และ Light Temperature and Humidity เข้ากับ RS485 เพื่อส่งข้อมูลค่าที่ Sensor วัดได้ ไปยัง AP Gateway และส่งข้อมูลต่อไปยัง Zeta Cloud จะทำการทดสอบการรัน Python Script บน Raspberry Pi ที่ทำการลง Firmware OpenWrt ไว้ เพื่อเป็นการทดสอบการทำงานก่อนจะนำ Firmware นี้ไปใช้งานจริงกับ Router หลังจากทำการลงภาษา Python 3 และทำการติดตั้ง Library ต่างๆที่จำเป็นต้องใช้งานเรียบร้อยแล้ว จากนั้นใช้โปรแกรม WinSCP ในการโอนย้ายไฟล์ Python Script ที่เขียนไว้บนคอมพิวเตอร์ไปยัง Raspberry Pi และกำหนดสิทธิในการรันให้กับไฟล์ พบว่าสามารถใช้งาน Firmware OpenWrt ในการรัน Python Script สำหรับใช้งาน MQTT Protocol ในการ Subscribe ข้อมูลจาก Sensor ได้ ได้ผลแสดงดังรูปที่ 4.23

```
root@OpenWrt:~/project# ./MQTT.py
----- procesTask -----
Client ID: python-mqtt-26
No any data is Queue!!
----- procesTask -----
No any data is Queue!!
```

รูปที่ 4.23 ผลจากการรัน Python Script

#### 4.9 การทดสอบการรับข้อมูลของ Sensor ด้วย OpenWrt Router

หลังจากที่ทำการทดสอบการรัน Python Script บน Raspberry Pi ที่ทำการลง Firmware OpenWrt เรียบร้อยแล้ว และทำการติดตั้ง Firmware OpenWrt ลงบน Router ที่จะใช้งานพร้อมกับติดตั้งภาษา Python และ Library ต่างๆที่จำเป็นต้องใช้ จะทำการทดสอบการใช้งาน OpenWrt Router ในการรัน Python Script สำหรับรับข้อมูลจาก Sensor ด้วยโปรโตคอล MQTT โดยแบ่งการทดลองออกเป็นสองส่วนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.9.1 ทดสอบการรับข้อมูลจาก RS-485 Soil Temperature and Humidity Sensor ด้วย OpenWrt Router

ทำการรัน Python Script เพื่อทดสอบการรับข้อมูลจาก RS-485 Soil Temperature and Humidity Sensor ด้วย OpenWrt Router

```

json_get = {'companyCode': '980f70347bc145c8a0adb4a34a76b264', 'deviceType': '485 Transceiver', 'addressCode': None, 'data': {'unit': '', 'rawValue': '0101017400f10000', 'name': 'de': None, 'retreatDuration': None, 'stopBit': None, 'dataBit': None, 'heartCycle': None, 'value': '01017400f10000', 'name': '采集参数', 'nameEn': '采集参数', 'value': '01017400f10000', 'name', 'registersNum': None, 'header': '01', 'addr': None, 'collectedResult': None}, 'dataDetail': {'name': '原始数据', 'nameEn': '原始数据', 'value': '0101017400f10000', 'nameJp': '原始数据'}, 'function': 'resp', 'completeData': '0101017400f10000', 'collectedParam': {'unit': '', 'rawValue': '01017400f10000', 'baudRate': null, 'parityBit': null, 'enableRetreat': null, 'registersNum': null, 'header': '01', 'addr': '4f039a97', 'upTime': 1708512022675, 'deviceaddr': None, 'accessKey': '', 'deviceModel': 'RS485 2', 'companyCode': '980f70347bc145c8a0adb4a34a76b264', 'deviceType': 'RS485 2', 'Data Temp485 : 0101017400f10000', 'Bit Temp485 : 00f1', 'Temp485 : 24.10 Celsius', 'Bit Humid485 : 0174', 'Humid485 : 37.20 RH%', 'unix time : 1708512022', '2024-02-21 17:40:22'

```

รูปที่ 4.24 ค่าที่รับได้ได้จาก Soil Temperature and Humidity

#### 4.9.2 ทดสอบการรับข้อมูลจาก Light Temperature and Humidity Sensor ด้วย OpenWrt Router

ทำการรัน Python Script เพื่อทดสอบการรับข้อมูลจาก Light Temperature and Humidity Sensor ด้วย OpenWrt Router

```

json_get = {'companyCode': '980f70347bc145c8a0adb4a34a76b264', 'deviceType': '485 Transceiver', 'addressCode': None, 'data': {'unit': '', 'rawValue': '010201cc01130068', 'name': 'de': None, 'retreatDuration': None, 'stopBit': None, 'dataBit': None, 'heartCycle': None, 'value': '0201cc01130068', 'name': '采集参数', 'nameEn': '采集参数', 'value': '0201cc01130068', 'name', 'registersNum': None, 'header': '01', 'addr': None, 'collectedResult': None}, 'dataDetail': {'name': '原始数据', 'nameEn': '原始数据', 'value': '010201cc01130068', 'nameJp': '原始数据'}, 'function': 'resp', 'completeData': '010201cc01130068', 'collectedParam': {'unit': '', 'rawValue': '0201cc01130068', 'baudRate': null, 'parityBit': null, 'enableRetreat': null, 'registersNum': null, 'header': '01', 'addr': '4f039a97', 'upTime': 1708512028573, 'deviceaddr': None, 'accessKey': '', 'deviceModel': 'Light Tempair : 0113', 'Tempair : 27.50 Celsius', 'Bit Humidair : 01cc', 'Humidair : 46.00 RH%', 'Bit Light : 0068', 'Light : 10.40 lux'

```

รูปที่ 4.25 ค่าที่รับได้จาก Light Temperature and Humidity Sensor

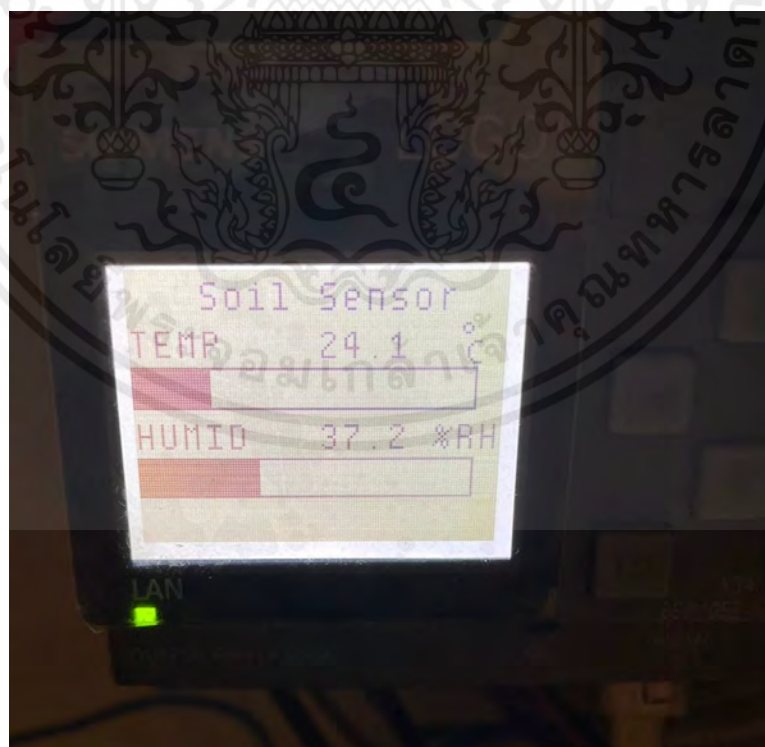
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการรัน Python Script พบว่า OpenWrt Router สามารถรัน Python Script สำหรับใช้งานโปรโตคอล MQTT ในการ Subscribe ข้อมูลจาก Sensor ได้ ได้ผลแสดงดังรูปที่ 4.24 และ 4.25 ซึ่งพบว่าค่าที่รับได้จากทั้งสอง Sensor มีค่าที่ถูกต้องตรงกับค่าที่ได้บนเว็บไซต์ของ Zeta Cloud

#### 4.10 การทดสอบการส่งข้อมูลจาก OpenWrt Router ไปยัง PLC Controller

หลังจากที่ OpenWrt Router สามารถ Subscribe ข้อมูลจาก RS-485 Soil Temperature and Humidity Sensor และ Light Temperature and Humidity Sensor ได้เรียบร้อยแล้ว จะทำการทดสอบการส่งข้อมูลที่ OpenWrt Router รับได้จาก Sensor ไปยัง PLC Controller โดยทำการรัน Python Script เพื่อใช้โปรโตคอล Modbus TCP/IP และทำการทดสอบการนำค่าที่ PLC Controller ได้รับมาแสดงผลบนหน้าจอของตามที่ได้ทำการออกแบบไว้ โดยแบ่งการทดลองออกเป็นสองส่วนดังนี้

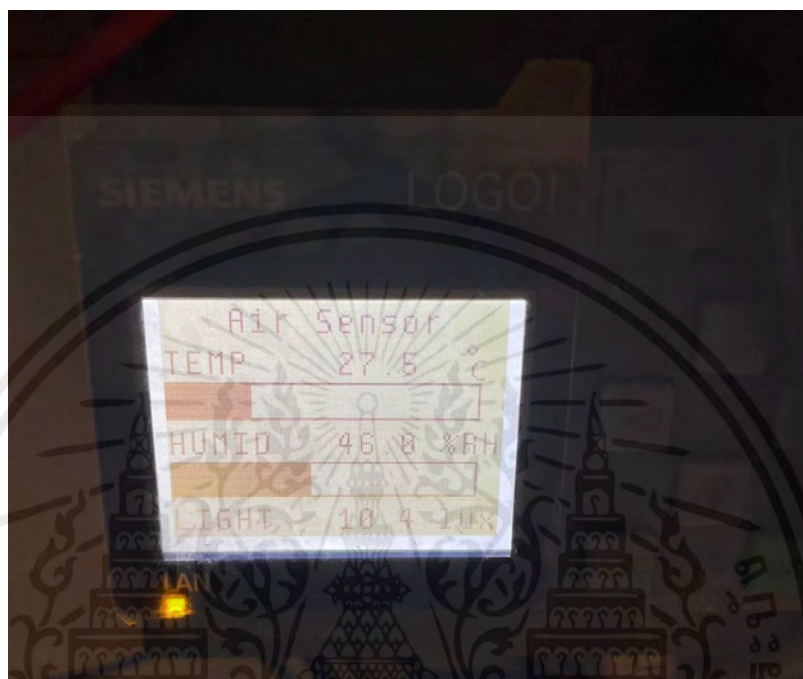
##### 4.10.1 ทดสอบการส่งข้อมูลที่รับได้จาก RS-485 Soil Temperature and Humidity Sensor ไปยัง PLC Controller ด้วย OpenWrt Router



รูปที่ 4.26 การแสดงผลค่าที่ได้จาก Soil Temperature and Humidity

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.10.2 ทดสอบการส่งข้อมูลที่รับได้จาก Light Temperature and Humidity Sensor ไปยัง PLC Controller ด้วย OpenWrt Router



รูปที่ 4.27 การแสดงผลค่าที่ได้จาก Light Temperature and Humidity Sensor

เมื่อทำการรัน Python Script พบว่า OpenWrt Router สามารถใช้งานโปรโตคอล Modbus TCP/IP ในการส่งค่าที่รับได้จาก Sensor ไปยัง PLC Controller ได้ ได้ผลแสดงดังรูปที่ 4.26 และ 4.27 ซึ่งพบว่าค่าที่ PLC Controller ได้รับได้จากทั้งสอง Sensor มีค่าที่ถูกต้องตรงกับค่าที่ได้บนเว็บไซต์ของ Zeta Cloud

#### 4.11 การทดสอบการส่งข้อมูลจาก OpenWrt Router ไปยังฐานข้อมูล

หลังจากที่ OpenWrt Router สามารถ Subscribe ข้อมูลจาก RS-485 Soil Temperature and Humidity Sensor และ Light Temperature and Humidity Sensor ได้เรียบร้อยแล้ว จะทำการทดสอบการส่งข้อมูลที่ OpenWrt Router รับผิดชอบได้จาก Sensor ไปจัดเก็บไว้ในฐานข้อมูล รวมถึงค่าบิตข้อมูลที่รับจากเซ็นเซอร์มาทำการแปลงเป็นเลขฐาน 10 โดยการใช้ภาษา Python ในการทำงาน และมีการส่งสถานะการทำงานของอุปกรณ์เข้าไปในฐานข้อมูล โดยในการทดลองนี้จะทำเก็บค่า ID, CompanyCode, DeviceType, Data, Tempsoil, Humidsoil, Infosoil, Tempair, Humidair, Light, Infoair, Statuslight, Statusfan, Statuspump, Datetime, Bitstatuslight, Bitstatusfan และ Bitstatuspump ตามแต่ละคอลัมน์ที่ได้ทำการสร้างไว้ใน Table จากการทดสอบจะพบว่าสามารถเก็บค่าข้อมูลต่างๆตามที่ต้องการลงในฐานข้อมูลได้อย่างครบถ้วน และมีค่าตรงกับค่าที่ได้บนเว็บไซต์ของ Zeta Cloud แสดงดังรูปที่ 4.28

CompanyCode	DeviceType	Data	Tempsoil	Humidsoil	Infosoil
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101011e01090000010201f7010400d0	26.50	28.60	Temp : 26.50 Celcius , Humid : 28.60 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101012a01080000010201f7010300d0	26.40	29.80	Temp : 26.40 Celcius , Humid : 29.80 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101014901080000010201f2010200d0	26.40	32.90	Temp : 26.40 Celcius , Humid : 32.90 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	010101950105000001020201010200d7	26.10	34.10	Temp : 26.10 Celcius , Humid : 34.10 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	010101600115000001020203010200db	27.70	35.20	Temp : 27.70 Celcius , Humid : 35.20 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101017801020000010201ff010200de	25.80	37.60	Temp : 25.80 Celcius , Humid : 37.60 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101016d01010000010201fa01020003	25.70	36.50	Temp : 25.70 Celcius , Humid : 36.50 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101017000f60000010201be010d00d7	25.30	36.80	Temp : 25.30 Celcius , Humid : 36.80 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101016e00f60000010201c001050000	25.20	36.60	Temp : 25.20 Celcius , Humid : 36.60 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101017000f60000010201b901090076	25.20	36.80	Temp : 25.20 Celcius , Humid : 36.80 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101017300fb0000010201c201040000	25.10	37.10	Temp : 25.10 Celcius , Humid : 37.10 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101017500fb0000010201b801020076	25.10	37.30	Temp : 25.10 Celcius , Humid : 37.30 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101015300f1000001020205010500d0	24.10	33.90	Temp : 24.10 Celcius , Humid : 33.90 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101016b00f10000010201ba01110068	24.10	36.30	Temp : 24.10 Celcius , Humid : 36.30 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101016f00f10000010201c101120058	24.10	36.70	Temp : 24.10 Celcius , Humid : 36.70 RH%
980f70347bc145c8a0adb4a34e76b264	RS485_2	0101017400f10000010201cc01130068	24.10	37.20	Temp : 24.10 Celcius , Humid : 37.20 RH%

Tempair	Humidair	Light	Infoair	Statuslight	Statusfan	Statuspump	Datetime	Bitstatuslight	Bitstatusfan	Bitstatuspump
26.00	50.30	20.80	Temp : 26.00 Celcius , Humid : 50.30 RH% , Light : 20.80 lux	OFF	ON	ON	2024-02-21 16:41:22	0	1	1
25.90	50.30	20.80	Temp : 25.90 Celcius , Humid : 50.30 RH% , Light : 20.80 lux	OFF	ON	ON	2024-02-21 16:42:22	0	1	1
25.80	49.80	20.80	Temp : 25.80 Celcius , Humid : 49.80 RH% , Light : 20.80 lux	OFF	ON	ON	2024-02-21 16:43:22	0	1	1
25.80	51.30	21.50	Temp : 25.80 Celcius , Humid : 51.30 RH% , Light : 21.50 lux	OFF	ON	ON	2024-02-21 16:44:22	0	1	1
25.80	51.50	21.90	Temp : 25.80 Celcius , Humid : 51.50 RH% , Light : 21.90 lux	OFF	ON	ON	2024-02-21 16:45:22	0	1	1
25.80	51.10	22.20	Temp : 25.80 Celcius , Humid : 51.10 RH% , Light : 22.20 lux	OFF	ON	ON	2024-02-21 16:46:22	0	1	1
25.80	50.60	0.30	Temp : 25.80 Celcius , Humid : 50.60 RH% , Light : 0.30 lux	ON	ON	ON	2024-02-21 16:54:22	1	1	1
26.90	44.60	21.50	Temp : 26.90 Celcius , Humid : 44.60 RH% , Light : 21.50 lux	OFF	ON	OFF	2024-02-21 17:08:22	0	1	0
26.10	44.80	0.00	Temp : 26.10 Celcius , Humid : 44.80 RH% , Light : 0.00 lux	ON	ON	ON	2024-02-21 17:11:22	1	1	1
26.10	44.10	11.80	Temp : 26.10 Celcius , Humid : 44.10 RH% , Light : 11.80 lux	OFF	ON	ON	2024-02-21 17:12:22	0	1	1
26.00	45.00	0.00	Temp : 26.00 Celcius , Humid : 45.00 RH% , Light : 0.00 lux	ON	ON	ON	2024-02-21 17:13:22	1	1	1
25.80	44.00	11.80	Temp : 25.80 Celcius , Humid : 44.00 RH% , Light : 11.80 lux	OFF	ON	ON	2024-02-21 17:14:22	0	1	1
26.10	51.70	20.80	Temp : 26.10 Celcius , Humid : 51.70 RH% , Light : 20.80 lux	OFF	ON	ON	2024-02-21 17:25:22	0	1	1
27.30	44.20	10.40	Temp : 27.30 Celcius , Humid : 44.20 RH% , Light : 10.40 lux	ON	ON	ON	2024-02-21 17:38:22	1	1	1
27.50	44.90	10.40	Temp : 27.50 Celcius , Humid : 44.90 RH% , Light : 10.40 lux	ON	ON	ON	2024-02-21 17:39:22	1	1	1
27.50	46.00	10.40	Temp : 27.50 Celcius , Humid : 46.00 RH% , Light : 10.40 lux	ON	ON	ON	2024-02-21 17:40:22	1	1	1

รูปที่ 4.28 ค่าที่ OpenWrt Router ส่งมายังฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.12 การทดสอบการแสดงผลข้อมูลบนเว็บไซต์

ทดสอบการนำข้อมูลที่วัดได้จาก RS-485 Soil Temperature and Humidity Sensor และ Light Temperature and Humidity Sensor มาแสดงผลบนเว็บไซต์ที่ได้เขียนขึ้น โดยทำการดึงค่าข้อมูลต่างๆที่ได้ทำการเก็บไว้ในฐานข้อมูลมาแสดงผล โดยสามารถนำข้อมูลมาแสดงผล ได้ผลดังรูปที่ 4.29 พบว่าข้อมูลที่แสดงผลบนเว็บไซต์ที่ได้ทำการเขียนขึ้นมีความถูกต้องและตรงกับค่าที่ได้บนเว็บไซต์ของ Zeta Cloud



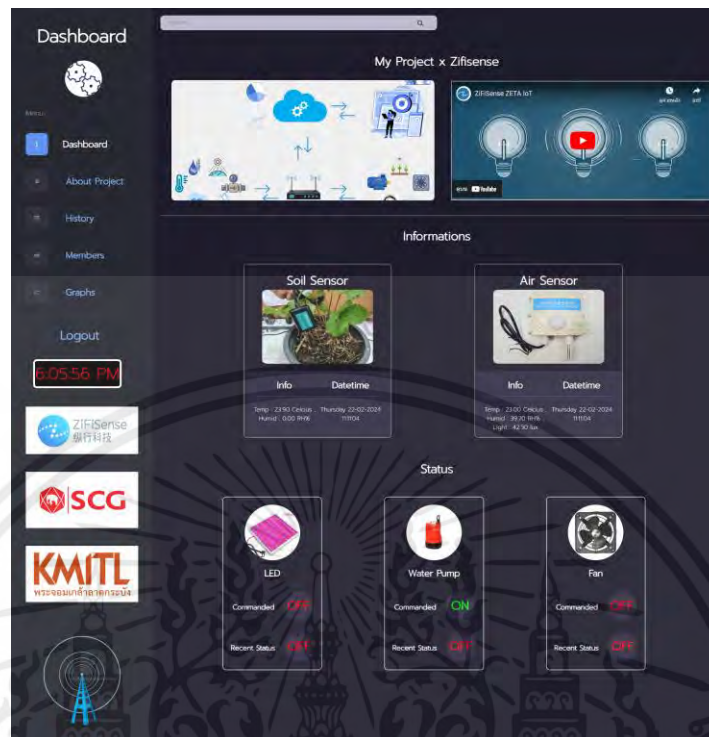
รูปที่ 4.29 การดึงค่าข้อมูลต่างๆที่ได้ทำการเก็บไว้ในฐานข้อมูลมาแสดงผลบนเว็บไซต์

นอกจากนี้ยังได้ทำการทดสอบการแสดงผลข้อมูลบนเว็บไซต์ตามเงื่อนไขการเข้าใช้งานที่ได้กำหนดไว้โดยแบ่งออกเป็น 4 กรณีตามการเข้าใช้งานดังนี้

### 4.12.1 กรณีที่เข้าใช้งานแบบ Admin หลัก

เมื่อเข้าใช้งานด้วย Username ที่เป็น Admin หลัก จะสามารถเห็นข้อมูลของทั้งสอง Sensor และสามารถเห็นข้อมูลของหน้าย่อยทั้ง 5 หน้า ในเว็บไซต์ได้ทั้งหมด ได้แก่ Dashboard, About Project, History, Members และ Graphs ได้ผลแสดงดังรูปที่ 4.30, 4.31, 4.32, 4.33 และ 4.34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.30 หน้า Dashboard เมื่อใช้งานแบบ Admin หลัก

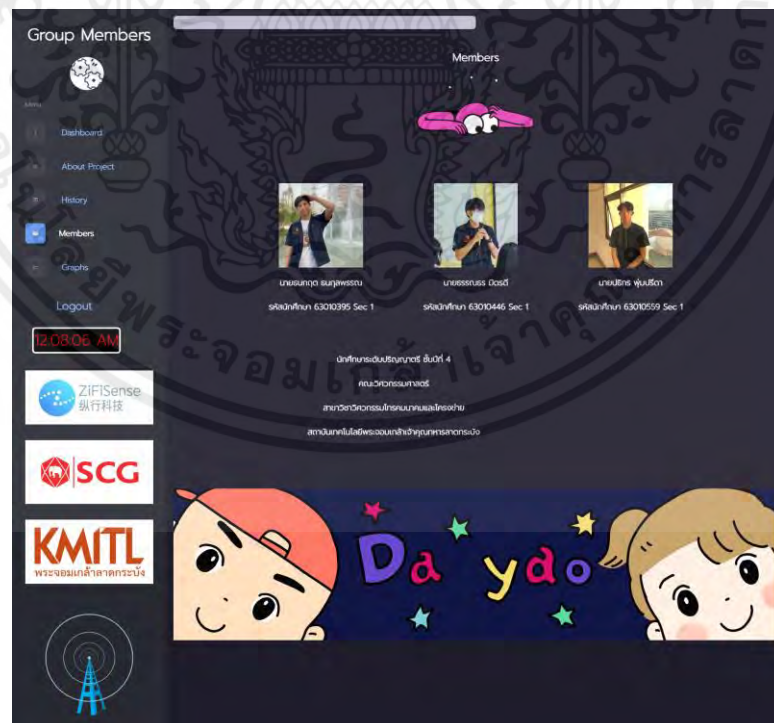


รูปที่ 4.31 หน้า About Project เมื่อใช้งานแบบ Admin หลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

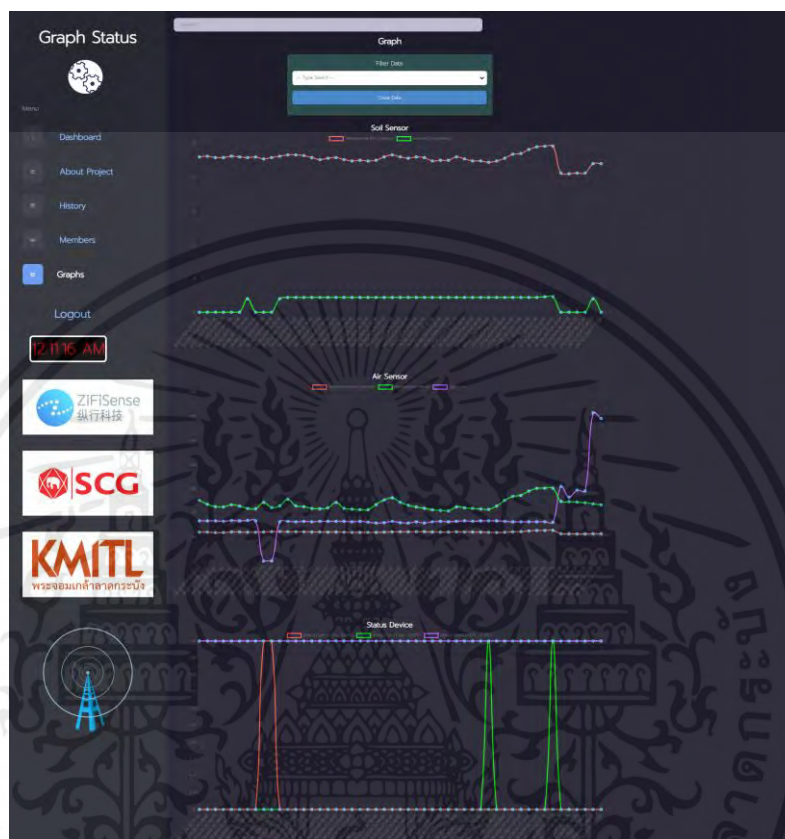


รูปที่ 4.32 หน้า History เมื่อเข้าใช้งานแบบ Admin หลัก



รูปที่ 4.33 หน้า Members เมื่อเข้าใช้งานแบบ Admin หลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

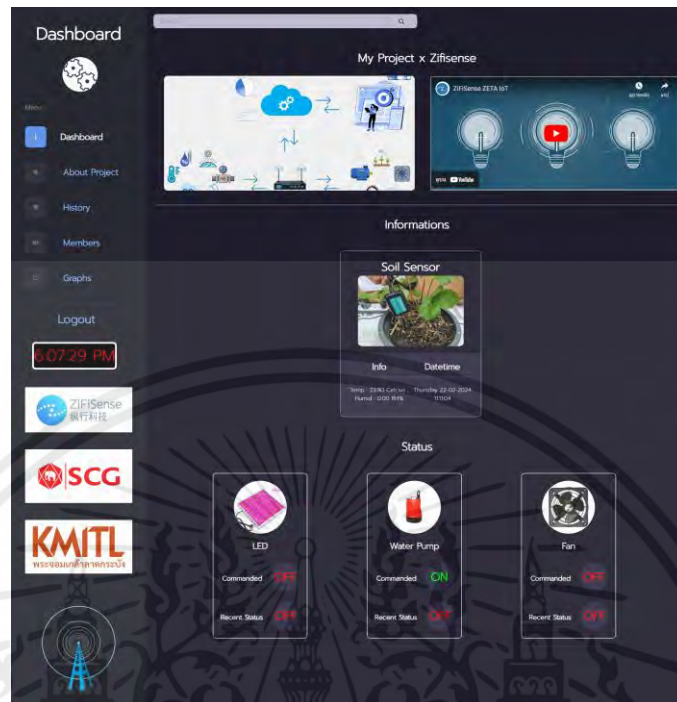


รูปที่ 4.34 หน้า Graphs เมื่อเข้าใช้งานแบบ Admin หลัก

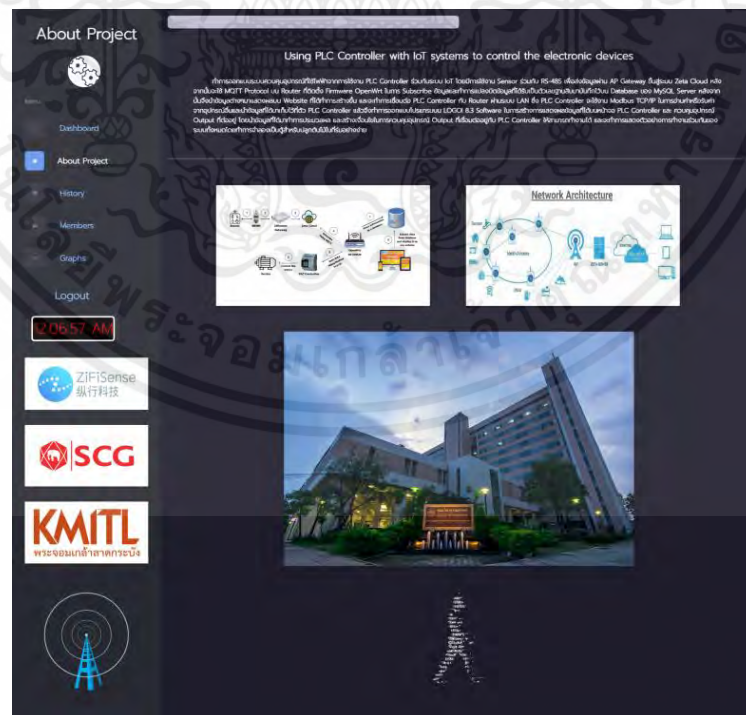
#### 4.12.2 กรณีที่เข้าใช้งานแบบ Admin บริษัทที่ 1

เมื่อเข้าใช้งานด้วย Username ที่เป็น Admin บริษัทที่ 1 จะสามารถเห็นข้อมูลของหน้าย่อยทั้ง 5 หน้า ในเว็บไซต์ได้ทั้งหมดเช่นเดียวกันแต่จะสามารถเห็นข้อมูลจาก RS-485 Soil Temperature and Humidity Sensor เท่านั้น ได้ผลแสดงดังรูปที่ 4.35, 4.36, 4.37, 4.38 และ 4.39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.35 หน้า Dashboard เมื่อใช้งานแบบ Admin บริษัทที่ 1



รูปที่ 4.36 หน้า About Project เมื่อใช้งานแบบ Admin บริษัทที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

History

Filter Date

ID	TEMP	HUMID	STATUS1	STATUS2	STATUS3	DATE
00000000000000000000000000000000	Temp: 23.30 Celsius	Humid: 0.00 RH%	OFF	OFF	ON	2024-02-06 22:36:42
00000000000000000000000000000000	Temp: 23.20 Celsius	Humid: 0.00 RH%	OFF	OFF	ON	2024-02-06 22:28:42
00000000000000000000000000000000	Temp: 23.00 Celsius	Humid: 0.00 RH%	OFF	OFF	ON	2024-02-06 22:23:42
00000000000000000000000000000000	Temp: 23.00 Celsius	Humid: 0.00 RH%	OFF	OFF	ON	2024-02-06 22:18:42
00000000000000000000000000000000	Temp: 23.20 Celsius	Humid: 0.00 RH%	OFF	OFF	ON	2024-02-06 22:14:42
00000000000000000000000000000000	Temp: 23.30 Celsius	Humid: 0.00 RH%	OFF	OFF	ON	2024-02-06 22:09:42
00000000000000000000000000000000	Temp: 23.30 Celsius	Humid: 0.00 RH%	OFF	OFF	ON	2024-02-06 22:04:42
00000000000000000000000000000000	Temp: 23.00 Celsius	Humid: 0.00 RH%	ON	OFF	ON	2024-02-06 22:02:42
00000000000000000000000000000000	Temp: 23.20 Celsius	Humid: 2.00 RH%	OFF	OFF	ON	2024-02-06 22:52:42
00000000000000000000000000000000	Temp: 23.20 Celsius	Humid: 2.00 RH%	OFF	OFF	ON	2024-02-06 22:53:42
00000000000000000000000000000000	Temp: 23.40 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:07:42
00000000000000000000000000000000	Temp: 23.40 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:12:42
00000000000000000000000000000000	Temp: 23.30 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:10:42
00000000000000000000000000000000	Temp: 23.00 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:18:42
00000000000000000000000000000000	Temp: 23.20 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:17:42
00000000000000000000000000000000	Temp: 23.30 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:16:42
00000000000000000000000000000000	Temp: 23.30 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:15:42
00000000000000000000000000000000	Temp: 23.30 Celsius	Humid: 2.20 RH%	OFF	OFF	ON	2024-02-06 23:14:42

รูปที่ 4.37 หน้า History เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1

Group Members

Members

เมธพงศ์ อรุณพรหม ID: 6310395 Sec 1

เมธสวัสต์ เดช ID: 6310446 Sec 1

เมธธิธ พูนรัตน์ ID: 6310559 Sec 1

ทั้งหมด 4 คน

ศูนย์นวัตกรรม

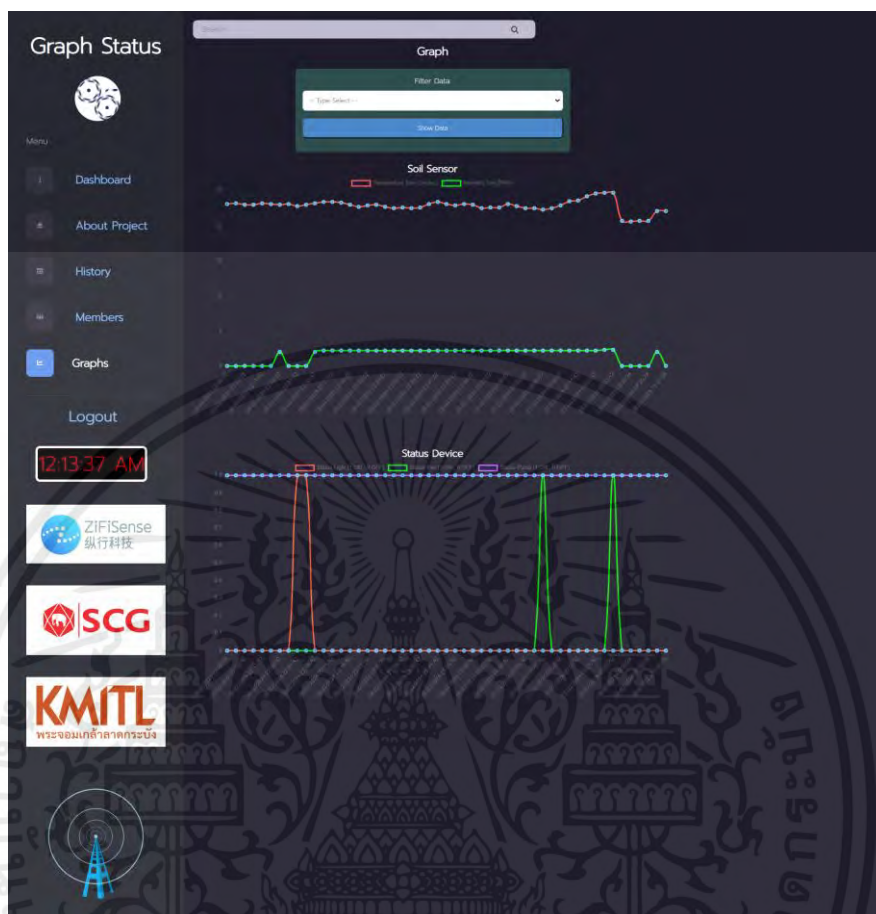
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Da y do

รูปที่ 4.38 หน้า Members เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

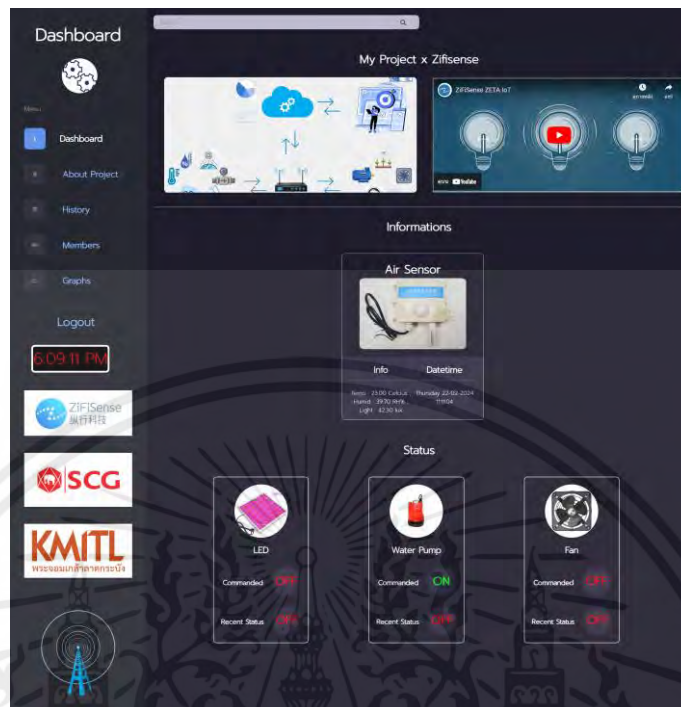


รูปที่ 4.39 หน้า Graphs เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 1

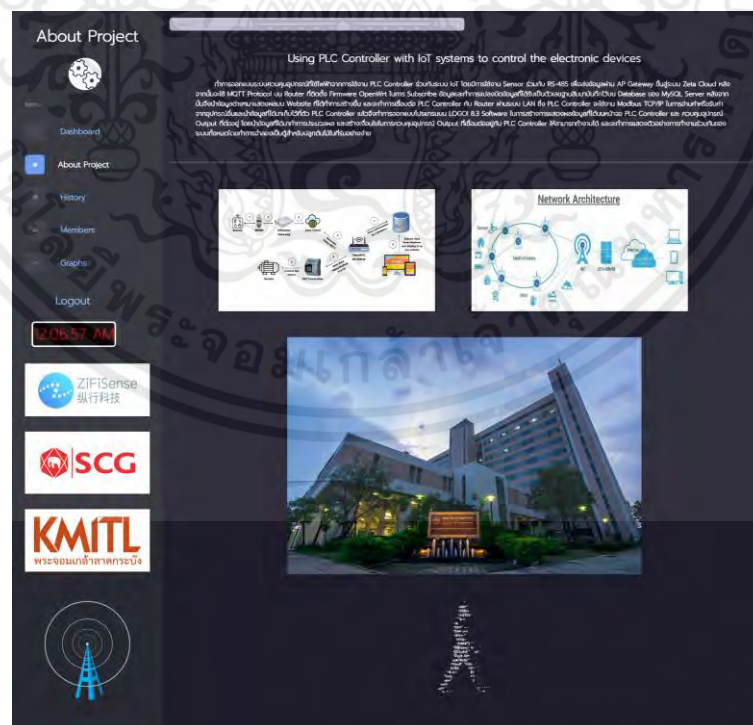
#### 4.12.3 กรณีที่เข้าใช้งานแบบ Admin บริษัทที่ 2

เมื่อเข้าใช้งานด้วย Username ที่เป็น Admin บริษัทที่ 2 จะสามารถเห็นข้อมูลของหน้าย่อยทั้ง 5 หน้า ในเว็บไซต์ได้ทั้งหมดเช่นเดียวกันแต่จะสามารถเห็นข้อมูลจาก จาก Light Temperature and Humidity Sensor เท่านั้น ได้ผลแสดงดังรูปที่ 4.40, 4.41, 4.42, 4.43 และ 4.44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



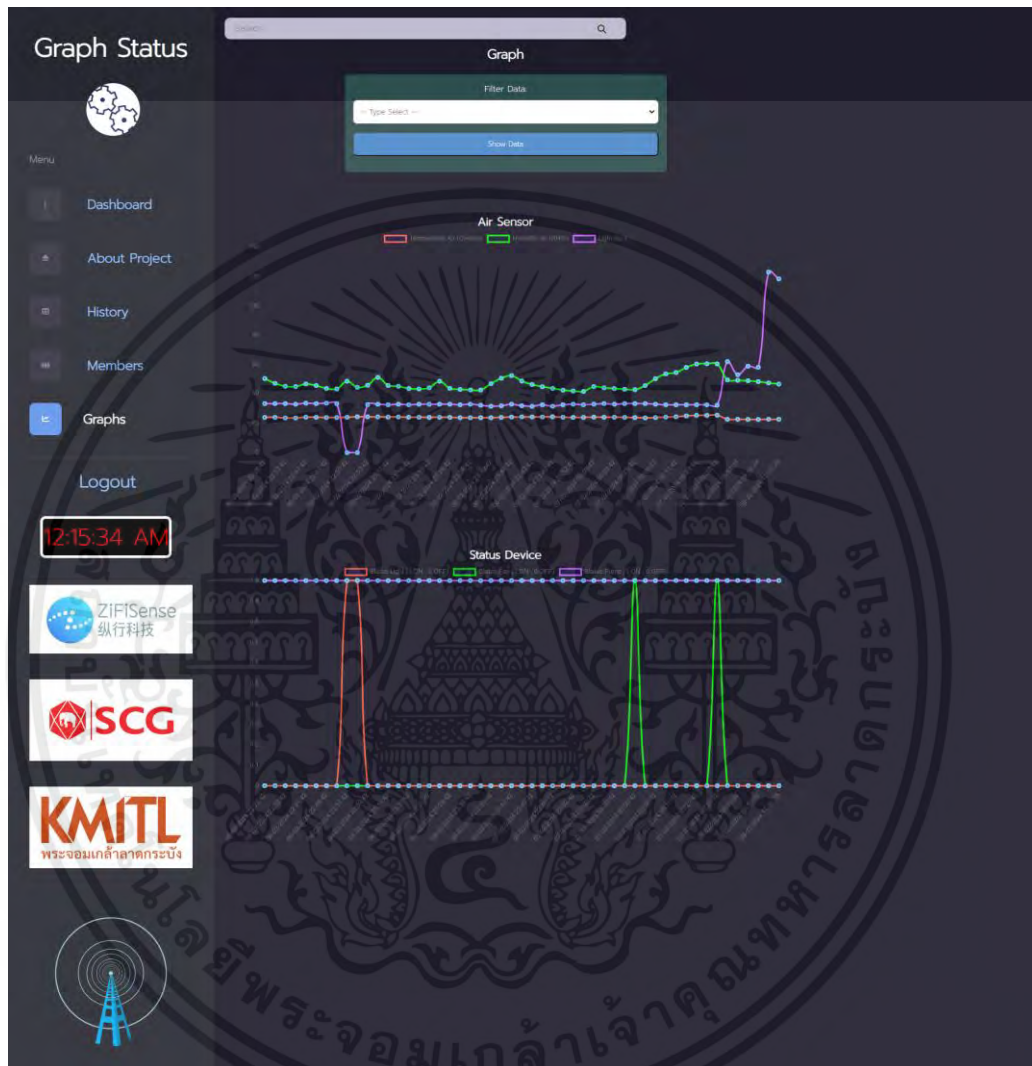
รูปที่ 4.40 หน้า Dashboard เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2



รูปที่ 4.41 หน้า About Project เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



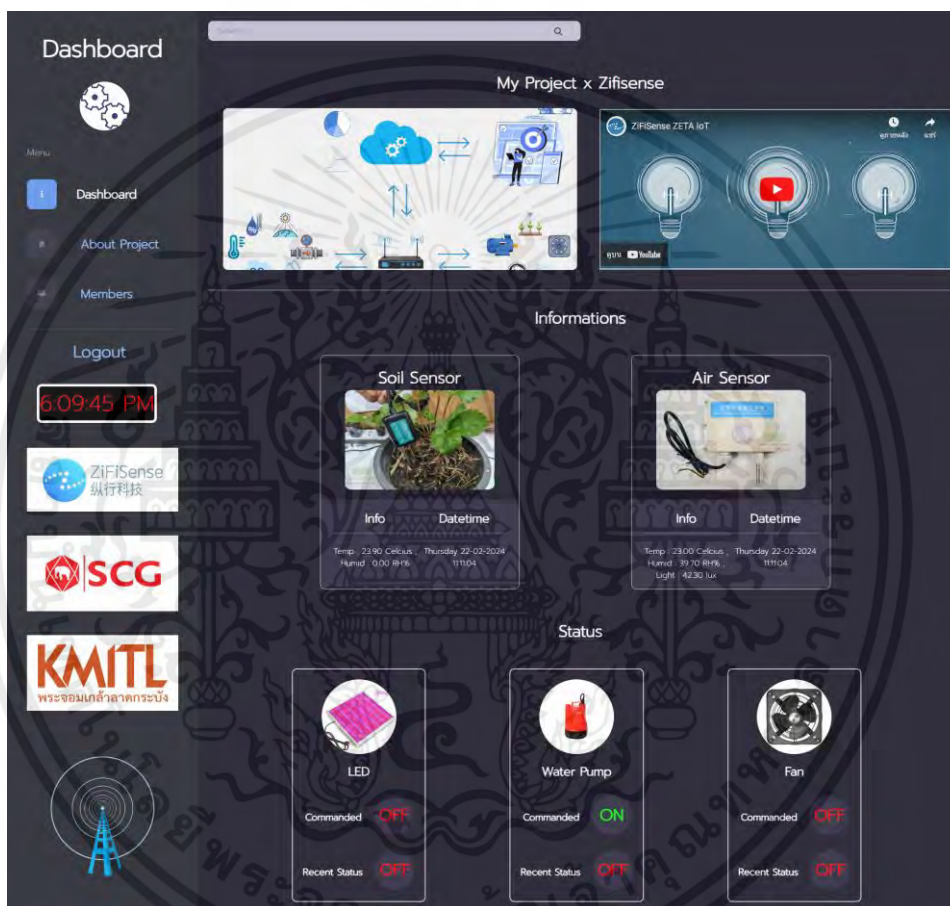


รูปที่ 4.44 หน้า Graphs เมื่อเข้าใช้งานแบบ Admin บริษัทที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.12.4 กรณีที่เข้าใช้งานโดย User ทั่วไป

เมื่อเข้าใช้งานด้วย Username ที่เป็น User ทั่วไป จะสามารถเห็นข้อมูลของหน้าย่อยใน เว็บไซต์ได้เพียง 3 หน้า ได้แก่ Dashboard, About Project และ Members ไม่สามารถใช้งานหน้า History และ Graphs ได้ ได้ผลแสดงดังรูปที่ 4.45, 4.46 และ 4.47

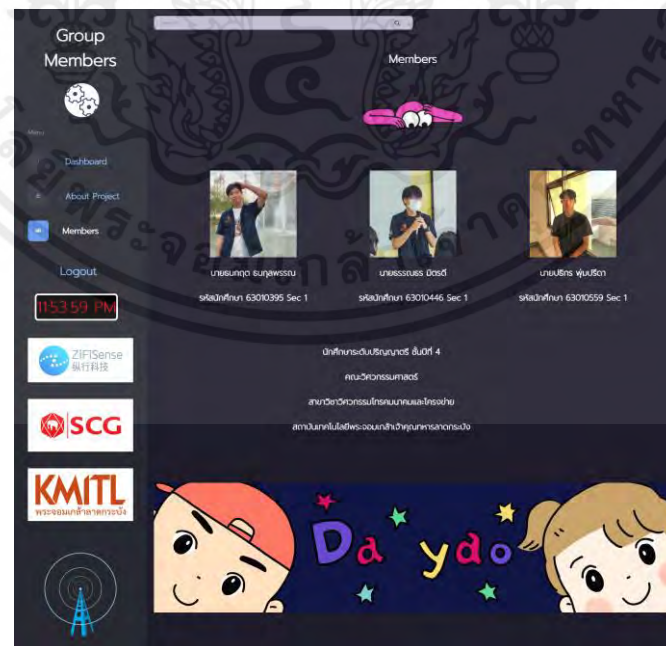


รูปที่ 4.45 หน้า Dashboard เมื่อเข้าใช้งานแบบ User ทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.46 หน้า About Project เมื่อใช้งานแบบ User ทั่วไป



รูปที่ 4.47 หน้า Members เมื่อใช้งานแบบ User ทั่วไป

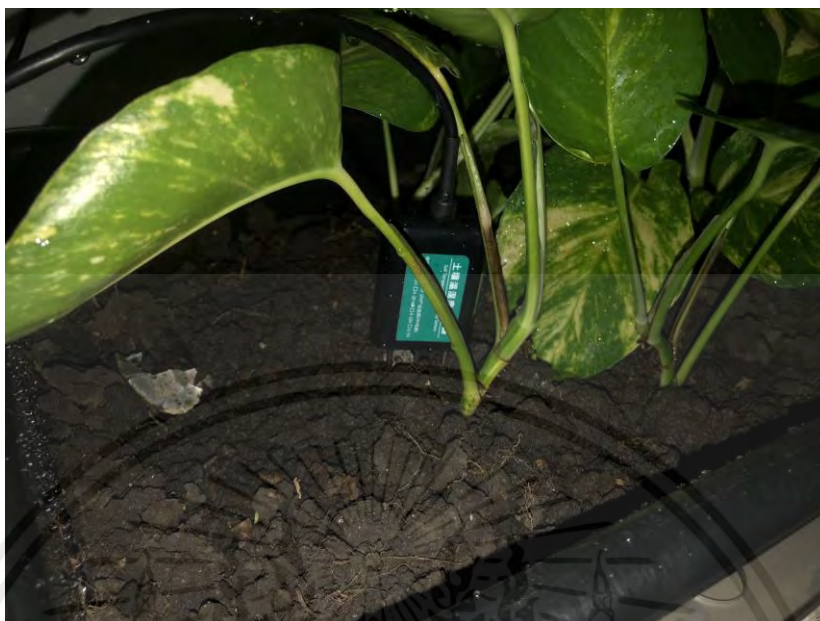
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.13 การทดสอบการทำงานของระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้า

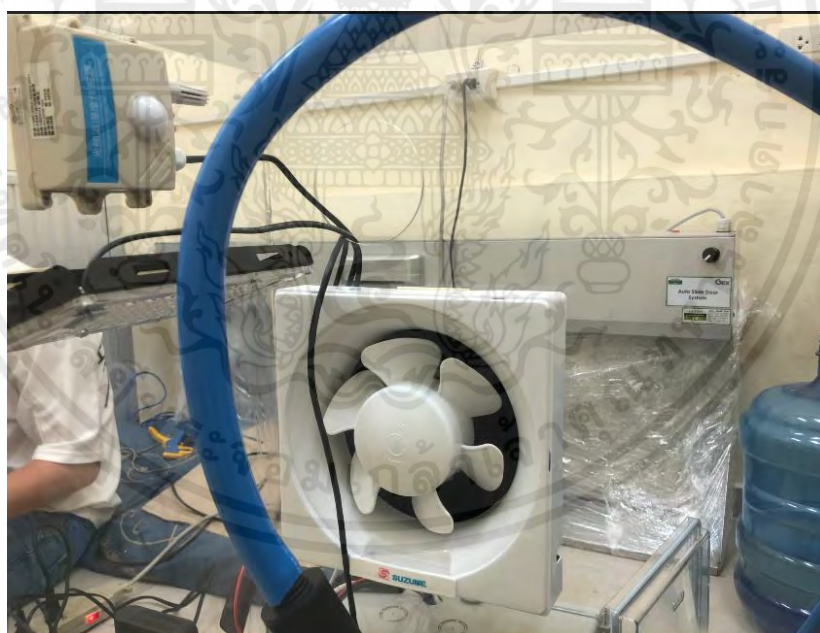
ทำการติดตั้ง AP Gateway และอุปกรณ์กระจายสัญญาณต่างๆ และทำการเชื่อมต่ออุปกรณ์ทั้งหมดตามที่ได้ทำการออกแบบไว้โดยเชื่อมต่อ OpenWrt Router เข้ากับ PLC Controller ผ่าน Layer 2 Switch ทำการเชื่อมต่ออุปกรณ์ Output ได้แก่ ไฟ LED สำหรับปลูกต้นไม้, พัดลมระบายอากาศ และ ปั๊มน้ำ เข้ากับ PLC Controller จากนั้นทำการเชื่อมต่อ Soil Temperature and Humidity Sensor สำหรับวัดค่าอุณหภูมิและความชื้นในดิน และ Light Temperature and Humidity Sensor สำหรับวัดค่าอุณหภูมิและความชื้นในอากาศ โดยทำการเชื่อมต่อ Sensor ทั้งสองแบบอนุกรมตามวงจรที่ได้ทำการออกแบบไว้เข้ากับ RS485 จากนั้นทำการติดตั้งอุปกรณ์ทั้งหมดเข้ากับตู้สำหรับปลูกต้นไม้ในที่ร่มอย่างง่ายแสดงดังรูปที่ 4.48



รูปที่ 4.48 การติดตั้งอุปกรณ์ทั้งหมดเข้ากับตู้สำหรับปลูกต้นไม้ในที่ร่มอย่างง่าย เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.49 Soil Temperature and Humidity Sensor

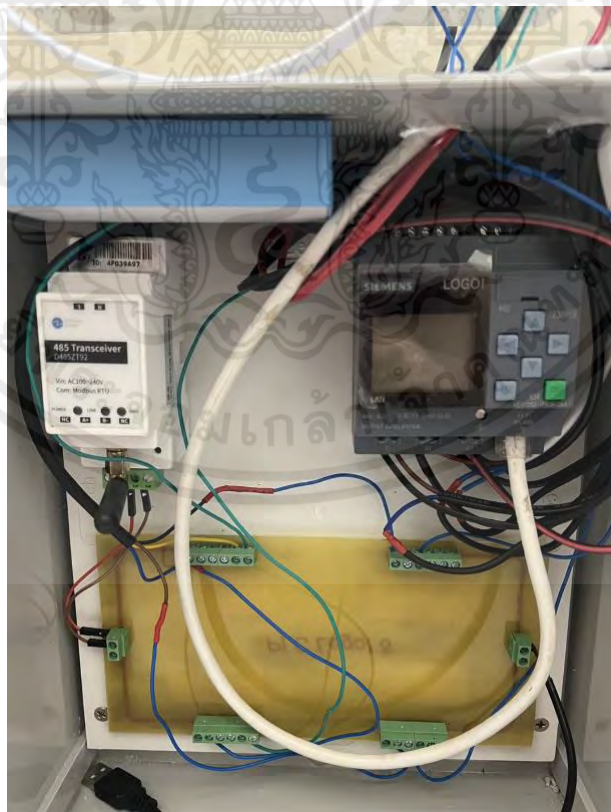


รูปที่ 4.50 Light Temperature and Humidity Sensor, ไฟ LED และ พัดลมระบายอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.51 ปั๊มน้ำสำหรับรดน้ำต้นไม้



รูปที่ 4.52 PLC Controller, RS485 และ แผ่นวงจร PCB สำหรับเชื่อม Sensor เข้ากับ RS485

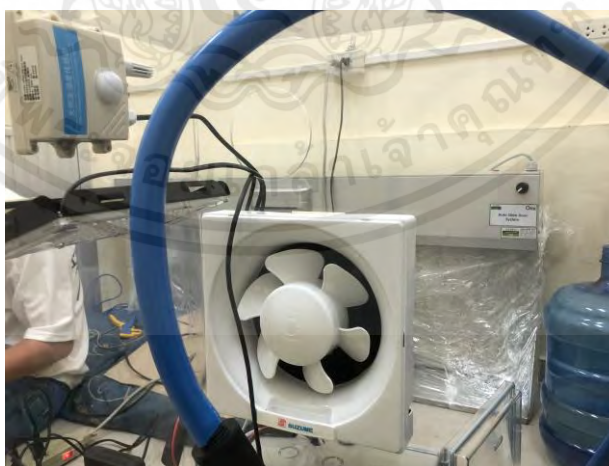
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มการทดสอบโดยการใช้ RS485 ส่งข้อมูลค่าที่ Sensor วัดได้ ไปยัง AP Gateway และส่งข้อมูลต่อไปยัง Zeta Cloud หลังจากนั้นจะใช้ MQTT Protocol บน Router ที่ติดตั้ง Firmware OpenWrt ในการ Subscribe ข้อมูลจากนั้นจะทำการแปลงข้อมูลให้อยู่ในรูปแบบ 10 แล้วทำการส่งบิตข้อมูลพร้อมกับส่งค่า Coil 0 หรือ 1 ตามเงื่อนไขที่ได้กำหนดไว้ให้กับ PLC Controller ผ่านโปรโตคอล Modbus TCP/IP ผ่านการใช้ Python Script ซึ่ง PLC Controller จะนำข้อมูลที่ได้ไปทำการแสดงผลบนหน้าจอและทำการควบคุมอุปกรณ์ Output ที่เชื่อมต่ออยู่กับ PLC Controller ซึ่งได้ทำการกำหนดเงื่อนไขในการควบคุมอุปกรณ์ Output ไว้ว่า เมื่ออุณหภูมิที่วัดได้ภายในตู้มีค่ามากกว่า 30 องศาเซลเซียส พัดลมระบายอากาศจะถูกควบคุมให้ทำงาน เมื่อความสว่างที่วัดได้มีค่าต่ำกว่า 10 ลักซ์ ไฟ LED สำหรับปลูกต้นไม้จะถูกควบคุมให้ทำงานและเมื่อความชื้นในดินที่วัดได้มีค่าต่ำกว่า 50 RH% ป้อนน้ำจะถูกควบคุมให้ทำงาน

ทำการแบ่งการทดลองออกเป็น 6 กรณีดังนี้

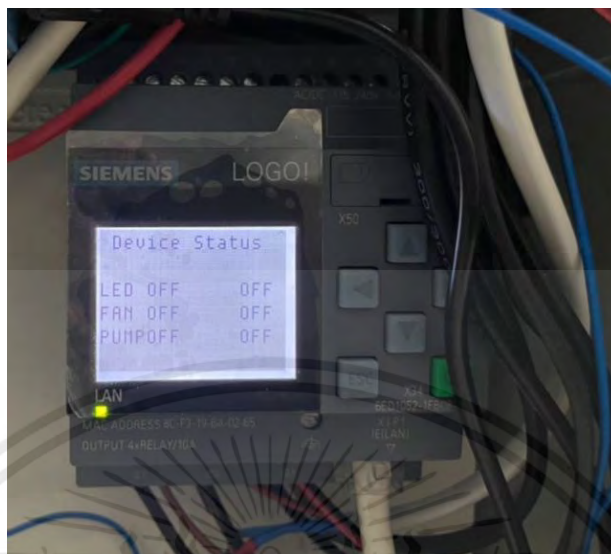
#### 4.13.1 กรณีที่อุณหภูมิที่วัดได้ภายในตู้มีค่าน้อยกว่าหรือเท่ากับ 30 องศาเซลเซียส

จากการทดลองพบว่าเมื่ออุณหภูมิที่วัดได้ภายในตู้มีค่าน้อยกว่าหรือเท่ากับ 30 องศาเซลเซียสพัดลมระบายอากาศจะไม่ทำงาน และ PLC Controller จะแสดงผลว่าพัดลมถูกสั่งให้ปิดอยู่ โดยจะแสดงผลเป็น OFF ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น OFF ทางด้านขวา



รูปที่ 4.53 พัดลมระบายอากาศไม่ทำงาน

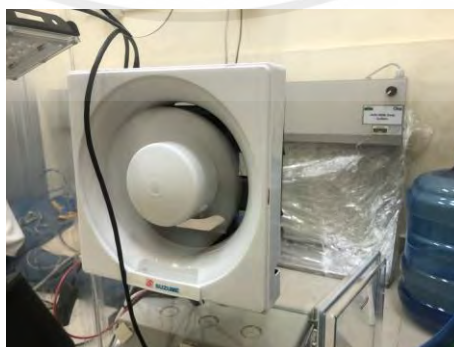
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.54 PLC Controller แสดงสถานะการทำงานของพัดลมว่าปิด

#### 4.13.2 กรณีที่อุณหภูมิที่วัดได้ภายในตู้มีค่ามากกว่า 30 องศาเซลเซียส

จากการทดลองพบว่าเมื่ออุณหภูมิที่วัดได้ภายในตู้มีค่ามากกว่า 30 องศาเซลเซียสพัดลมระบายอากาศจะทำงาน และ PLC Controller จะแสดงผลว่าพัดลมถูกสั่งให้เปิดอยู่ โดยจะแสดงผลเป็น ON ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น ON ทางด้านขวา แต่ถ้าหาก PLC Controller ควบคุมให้พัดลมระบายอากาศเริ่มทำงานแต่สวิตช์ของพัดลมระบายอากาศถูกปิดอยู่หรืออุปกรณ์ชำรุดแล้วไม่ทำงานจริง PLC Controller จะแสดงผลว่าพัดลมถูกสั่งให้เปิดอยู่ โดยจะแสดงผลเป็น ON ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น OFF ทางด้านขวา

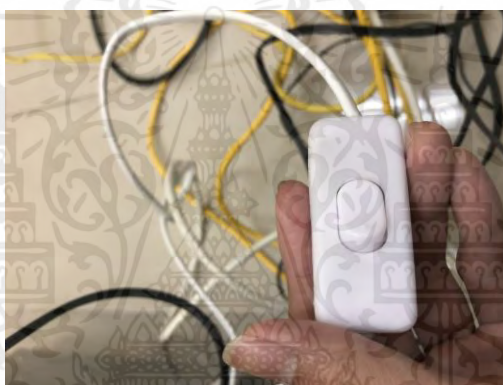


รูปที่ 4.55 พัดลมระบายอากาศทำงาน

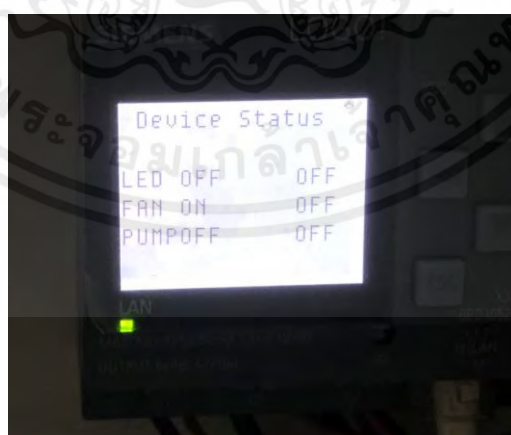
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.56 PLC Controller แสดงสถานะการทำงานของพัดลมระบายอากาศว่าเปิด



รูปที่ 4.57 ทำการปิดสวิตช์



รูปที่ 4.58 PLC Controller แสดงสถานะการทำงานของพัดลมระบายอากาศว่าถูกควบคุมให้เปิดแต่พัดลมระบายอากาศไม่ทำงาน

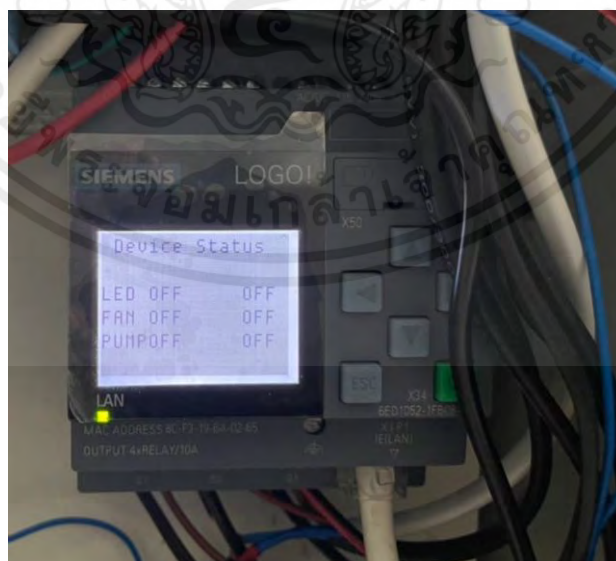
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.13.3 กรณีที่ความสว่างที่วัดได้มีค่ามากกว่าหรือเท่ากับ 10 ลักซ์

จากการทดลองพบว่าเมื่อความสว่างที่วัดได้มีค่ามากกว่าหรือเท่ากับ 10 ลักซ์ ไฟ LED สำหรับปลุกต้นไม้จะไม่ทำงาน และ PLC Controller จะแสดงผลว่าไฟ LED ถูกสั่งให้ปิดอยู่ โดยจะแสดงผลเป็น OFF ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น OFF ทางด้านขวา



รูปที่ 4.59 ไฟ LED ไม่ทำงาน

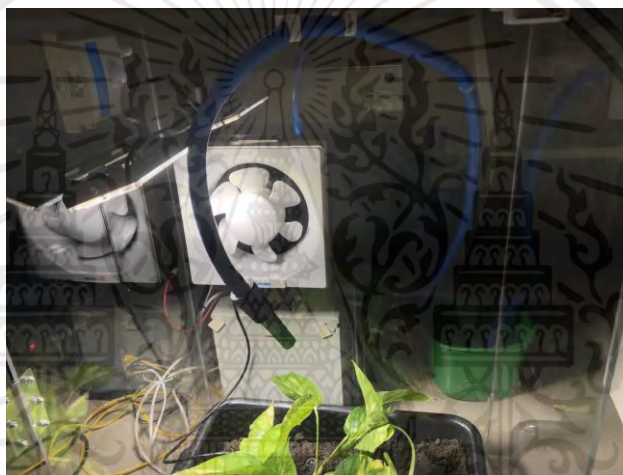


รูปที่ 4.60 PLC Controller แสดงสถานะการทำงานของไฟ LED ว่าปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.13.4 กรณีที่ความสว่างที่วัดได้มีค่าน้อยกว่า 10 ลักซ์

จากการทดลองพบว่าเมื่อความสว่างที่วัดได้มีค่าน้อยกว่า 10 ลักซ์ไฟ LED จะทำงาน และ PLC Controller จะแสดงผลว่าไฟ LED ถูกสั่งให้เปิดอยู่ โดยจะแสดงผลเป็น ON ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น ON ทางด้านขวา แต่ถ้าหาก PLC Controller ควบคุมให้ไฟ LED เริ่มทำงานแต่สวิทช์ของไฟ LED ถูกปิดอยู่หรืออุปกรณ์ชำรุดแล้วไม่ทำงานจริง PLC Controller จะแสดงผลว่าไฟ LED ถูกสั่งให้เปิดอยู่ โดยจะแสดงผลเป็น ON ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น OFF ทางด้านขวา

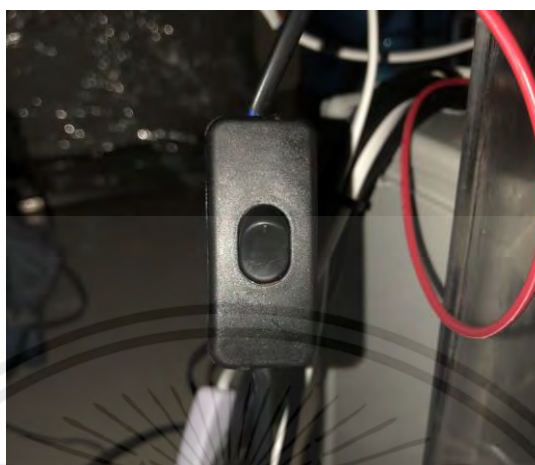


รูปที่ 4.61 ไฟ LED ทำงาน

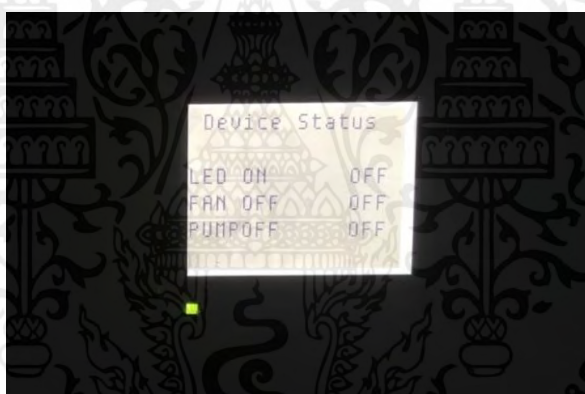


รูปที่ 4.62 PLC Controller แสดงสถานะการทำงานของไฟ LED ว่าเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.63 ทำการปิดสวิตช์



รูปที่ 4.64 PLC Controller แสดงสถานการณ์ทำงานของไฟ LED ว่าถูกควบคุมให้เปิดแต่ไฟ LED ไม่ทำงาน

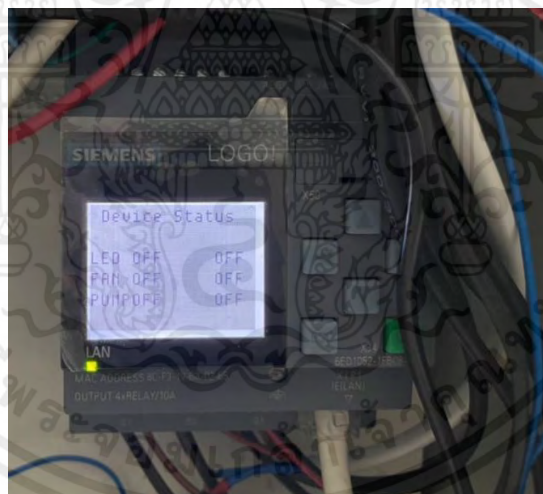
#### 4.13.5 กรณีที่ความชื้นในดินที่วัดได้มีค่ามากกว่าหรือเท่ากับ 50 RH%

จากการทดลองพบว่าเมื่อความชื้นในดินที่วัดได้มีค่ามากกว่าหรือเท่ากับ 50 RH% ป้อนน้ำจะไม่ทำงาน และ PLC Controller จะแสดงผลว่าป้อนน้ำถูกสั่งให้ปิดอยู่ โดยจะแสดงผลเป็น OFF ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น OFF ทางด้านขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.65 ปั๊มน้ำไม่ทำงาน



รูปที่ 4.66 PLC Controller แสดงสถานะการทำงานของปั๊มน้ำว่าปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.13.6 กรณีที่ความชื้นในดินที่วัดได้มีค่าน้อยกว่า 50 RH%

จากการทดลองพบว่าเมื่อความชื้นในดินที่วัดได้มีค่าน้อยกว่า 50 RH% ป้อนน้ำจะทำงาน และ PLC Controller จะแสดงผลว่าปั้มน้ำ ถูกสั่งให้เปิดอยู่ โดยจะแสดงผลเป็น ON ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น ON ทางด้านขวา แต่ถ้าหาก PLC Controller ควบคุมให้ปั้มน้ำเริ่มทำงานแต่อุปกรณ์ชำรุดแล้วไม่ทำงานจริง PLC Controller จะแสดงผลว่าปั้มน้ำ ถูกสั่งให้เปิดอยู่ โดยจะแสดงผลเป็น ON ทางด้านซ้าย และแสดงผลการตรวจสอบสถานะการทำงานจริงเป็น OFF ทางด้านขวา เช่นเดียวกับกับอุปกรณ์อื่นๆ แต่เนื่องจากในที่นี่ปั้มน้ำที่เลือกใช้ในปริยญา นิพนธ์ไม่มีสวิตซ์ จึงไม่ได้ทำการแสดงตัวอย่างการทดลอง



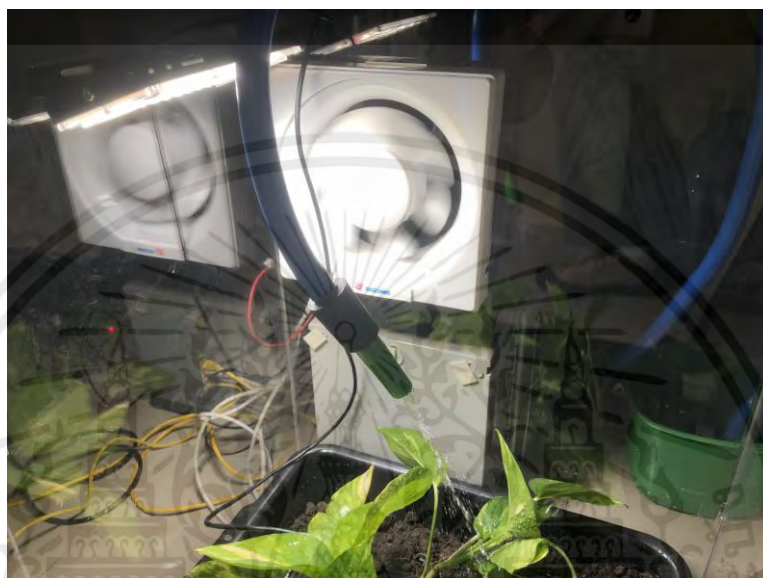
รูปที่ 4.67 ปั้มน้ำทำงาน



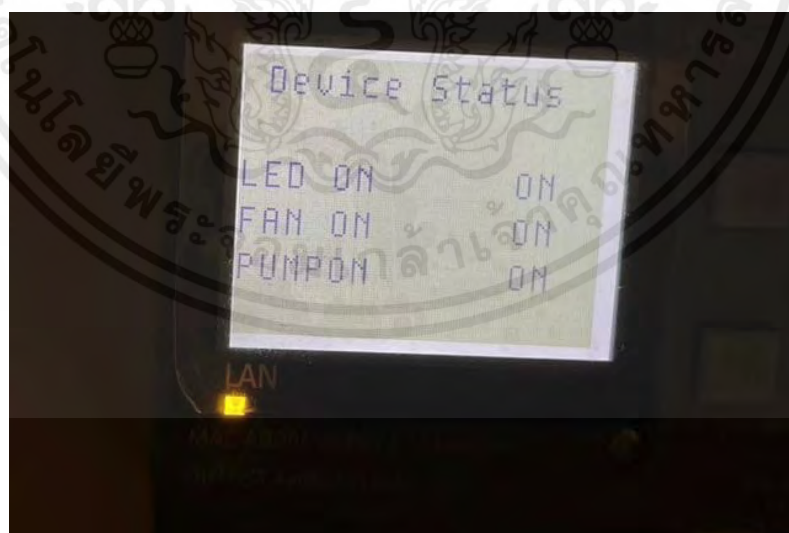
รูปที่ 4.68 PLC Controller แสดงสถานะการทำงานของปั้มน้ำว่าเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ PLC Controller ยังสามารถควบคุมอุปกรณ์ Output ให้ทำงานพร้อมกัน 2 อุปกรณ์ หรือ ควบคุมอุปกรณ์ Output ให้ทำงานพร้อมกันทั้ง 3 อุปกรณ์ได้ หากค่าที่ Sensor วัดได้ เป็นไปตามเงื่อนไขการควบคุมให้เริ่มทำงานพร้อมๆกัน



รูปที่ 4.69 อุปกรณ์ Output ทั้งหมดทำงาน



รูปที่ 4.70 PLC Controller แสดงสถานะการทำงานของอุปกรณ์ Output ทั้งหมดว่าเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

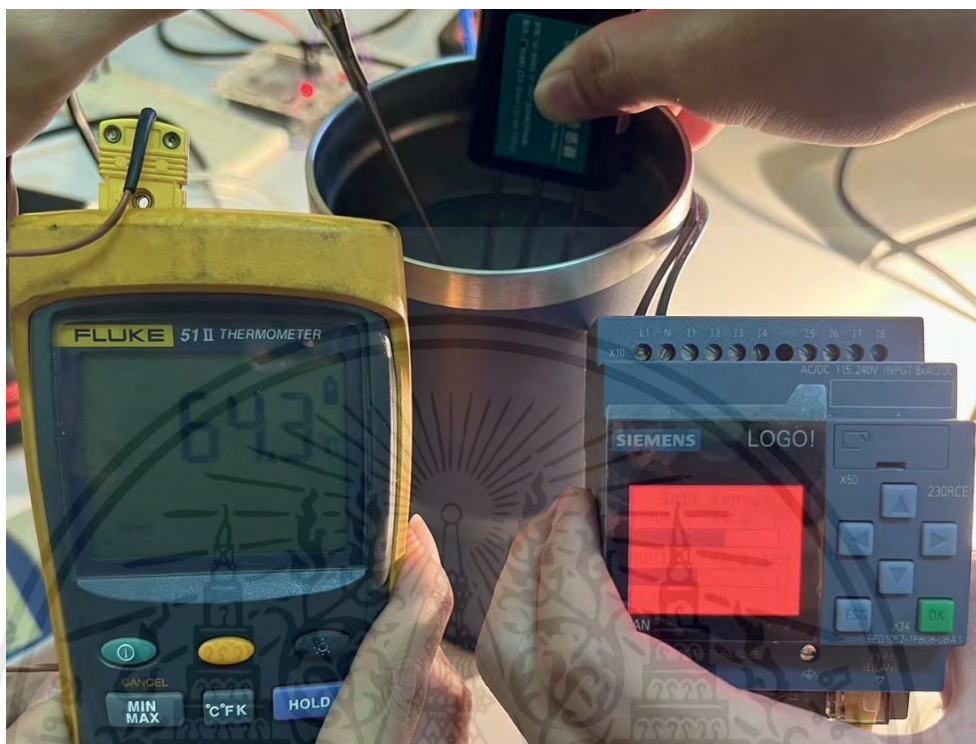
#### 4.14 การทดสอบประสิทธิภาพความแม่นยำในการวัดข้อมูลของ Sensor

ทดสอบประสิทธิภาพในการวัดค่าอุณหภูมิของ Soil Temperature and Humidity Sensor โดยเปรียบเทียบค่าอุณหภูมิที่วัดได้จาก Sensor กับค่าอุณหภูมิที่วัดได้จาก FLUKE MODEL 51 II HANDHELD PORTABLE DIGITAL PROBE THERMOMETER ซึ่งเป็นอุปกรณ์สำหรับวัดค่าอุณหภูมิที่ผ่านการสอบเทียบมาแล้ว โดยทำการทดสอบวัดค่าอุณหภูมิทั้งหมดสองรอบโดยใช้น้ำเย็นและน้ำร้อนในการทดสอบ พบว่าค่าที่ได้จากอุปกรณ์ที่ผ่านการสอบเทียบมาแล้วเป็น 14.2 และ 64.3 องศาเซลเซียส ตามลำดับ และค่าที่วัดได้จาก Soil Temperature and Humidity Sensor เป็น 14.6 และ 64.4 องศาเซลเซียสตามลำดับ แสดงดังรูปที่ 4.71 และ 4.72 ซึ่งสามารถนำมาหาค่าเฉลี่ยของความผิดพลาดที่ได้เท่ากับ 1.49 เปอร์เซ็นต์



รูปที่ 4.71 การทดสอบประสิทธิภาพความแม่นยำในการวัดข้อมูลกับน้ำเย็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.72 การทดสอบประสิทธิภาพความแม่นยำในการวัดข้อมูลกับน้ำร้อน

#### 4.15 การทดสอบประสิทธิภาพการส่งข้อมูลของ AP Gateway

ทดสอบประสิทธิภาพการส่งข้อมูลของ AP Gateway พบว่าใน 60 นาทีสามารถรับข้อมูลจาก AP Gateway ได้ 20 ครั้ง คิดเป็นประสิทธิภาพได้ 100%

```
Heartbeat_count = 1
Ap_Time(unix_time) = 1680520658
AP_Time = 2023-04-03 18:17:38
```

รูปที่ 4.73 เวลาที่ทำการรับข้อมูลจาก AP Gateway ครั้งที่ 1 เมื่อเวลาผ่านไป 3 นาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

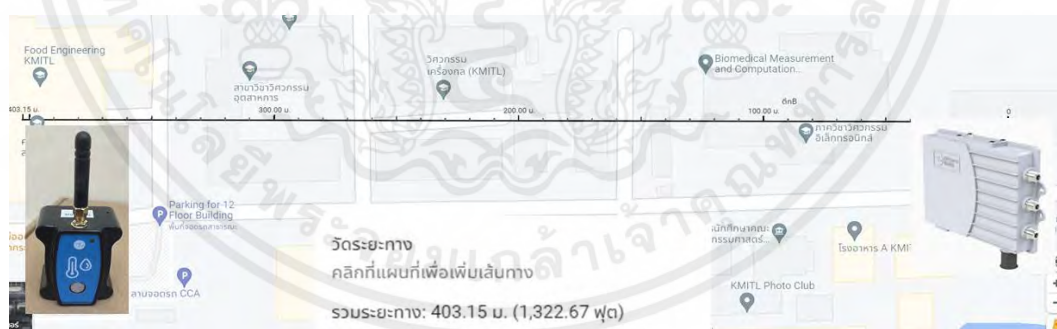
```
Heartbeat_count = 20
Ap_Time(unix_time) = 1680524079
AP_Time = 2023-04-03 19:14:39
```

รูปที่ 4.74 เวลาที่ทำการรับข้อมูลจาก AP Gateway ครั้งที่ 20 เมื่อเวลาผ่านไป 60 นาที

#### 4.16 การทดสอบประสิทธิภาพการส่งข้อมูลจาก Sensor

4.16.1 ทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor เมื่อ AP Gateway อยู่ในห้องภายในอาคาร (มีสิ่งกีดขวางมากกว่า) และไม่ได้ทำการใช้ Mote ขยายสัญญาณ

พบว่า sensor สามารถเชื่อมต่อกับ AP Gateway และส่งข้อมูลได้ที่ระยะทางไกลที่สุดประมาณ 403 เมตรและมีคุณภาพของสัญญาณอยู่ที่ -124 dBm และพบว่าใน 30 นาทีสามารถรับข้อมูลจาก Temp Humid sensor ได้ 10 ครั้ง คิดเป็นประสิทธิภาพได้ 100%



รูปที่ 4.75 ระยะทางที่ไกลที่สุดเมื่อ AP Gateway อยู่ในอาคาร และไม่ได้ทำการใช้ Mote ขยายสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.16.2 จากการทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor เมื่อ AP Gateway อยู่ด้านบนตาดฟ้าของอาคาร (มีสิ่งกีดขวางน้อยกว่า) และไม่ได้ทำการใช้ Mote ขยายสัญญาณ

พบว่า sensor สามารถเชื่อมต่อกับ AP Gateway และส่งข้อมูลได้ที่ระยะทางไกลที่สุดประมาณ 555 เมตร และมีคุณภาพของสัญญาณอยู่ที่  $-113$  dBm และพบว่าใน 30 นาทีสามารถรับข้อมูลจาก Temp Humid sensor ได้ 10 ครั้ง คิดเป็นประสิทธิภาพได้ 100%

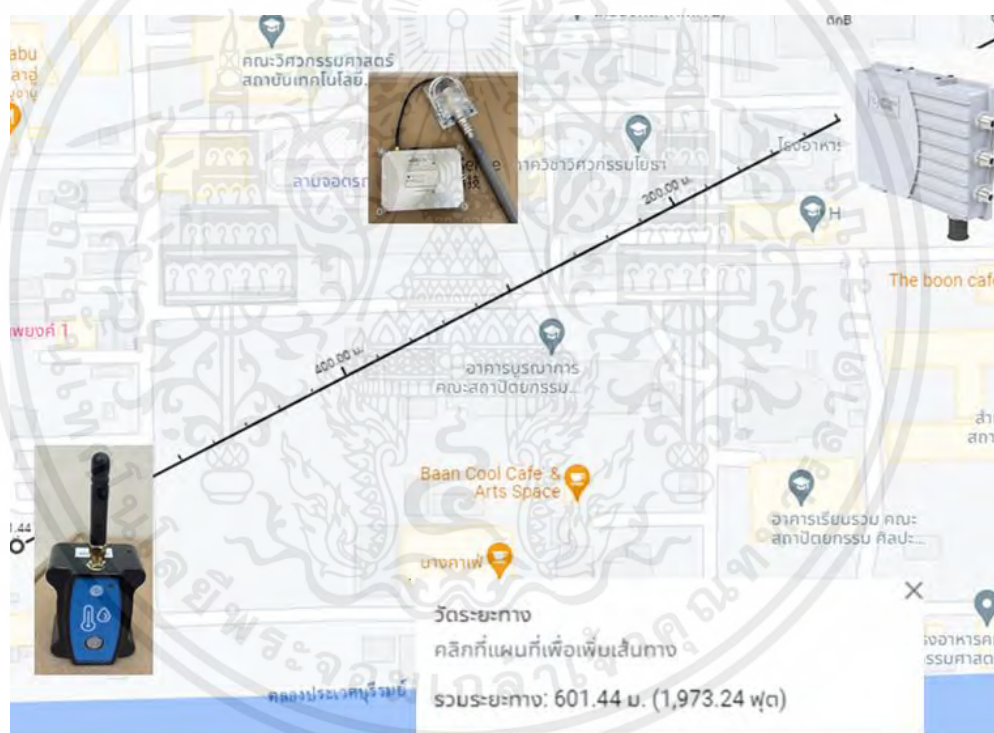


รูปที่ 4.76 ระยะทางที่ไกลที่สุดเมื่อ AP Gateway อยู่ด้านบนตาดฟ้าของอาคาร และไม่ได้ทำการใช้ Mote ขยายสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.16.3 จากการทดสอบประสิทธิภาพการส่งข้อมูลจาก sensor เมื่อ AP Gateway อยู่ด้านบนตาดฟ้าของอาคาร (มีสิ่งกีดขวางน้อยกว่า) และทำการใช้ Mote ขยายสัญญาณ อยู่ที่บริเวณหน้าอาคารยิมเนเซียม 1

พบว่า sensor สามารถเชื่อมต่อกับ Mote และส่งข้อมูลได้ที่ระยะทางไกลที่สุดประมาณ 600 เมตรและมีคุณภาพของสัญญาณอยู่ที่ -121 dBm และพบว่าใน 30 นาทีสามารถรับข้อมูลจาก Temp Humid sensor ได้ 10 ครั้ง คิดเป็นประสิทธิภาพได้ 100% ซึ่งจากการทดลองจะพบว่า ระยะทางจาก AP Gateway ถึง Mote จะห่างกันได้ที่ประมาณ 300 เมตร และ จาก Mote มายัง Sensor อีกประมาณ 350 เมตร



รูปที่ 4.77 ระยะทางที่ไกลที่สุดเมื่อ AP Gateway อยู่บนตาดฟ้า และได้ทำการใช้ Mote ขยายสัญญาณบริเวณหน้าอาคารยิมเนเซียม 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผล

#### 5.1 สรุปผล

ปริญญาบัตรนี้มีเป้าหมายเพื่อที่จะศึกษาเกี่ยวกับหลักการรับ-ส่งข้อมูลของ RS-485 และ Modbus TCP/IP โดยการใช้งาน RS-485 อ่านค่าจากเซ็นเซอร์ โดยใช้ MQTT Protocol ในการ Subscribe ข้อมูลที่ต้องการและนำบิตข้อมูลที่ต้องการมาทำการแปลงเป็นฐานสิบ จากนั้นสามารถส่งข้อมูลหรือค่าที่ได้รับจากเซ็นเซอร์ให้กับ PLC Controller โดยใช้กระบวนการ Modbus TCP/IP ที่ได้ทำการเขียนโดยใช้ภาษา Python ซึ่งได้มีการใช้อุปกรณ์ Router ที่ได้ทำการลง Firmware OpenWrt เข้าไปเพื่อให้ Router สามารถทำงานฟังก์ชันต่างๆได้มากขึ้นและสามารถที่จะรันไฟล์ Python ที่เราเขียนได้ รวมถึงยังสามารถออกแบบ Function Block ที่อยู่ในโปรแกรม LOGO! 8.3 Software ที่จะสามารถสร้างวงจรในการทำงานของ PLC Controller เพื่อควบคุมอุปกรณ์ที่เชื่อมต่อกับ Port Output ของ PLC Controller เช่น LED สำหรับปลูกต้นไม้, พัดลมระบายอากาศ และปั้มน้ำ ซึ่งยังสามารถออกแบบการแสดงผลผ่านทางหน้าจอของ PLC Controller ตามที่เราต้องการตั้งค่าได้ พร้อมทั้งยังมี Web Server ของ PLC Controller ที่สามารถแสดงพร้อมๆกับหน้าจอบนตัวเครื่องได้ และสามารถออกแบบระบบฐานข้อมูลสำหรับเก็บค่าข้อมูลต่างๆ ตามที่เราต้องการได้เพื่อนำค่าข้อมูลต่างๆ ที่เก็บเข้ามาไปแสดงผลผ่านทางหน้าเว็บไซต์ที่ได้ทำการสร้างขึ้น และยังสามารถจำลองการทำงานของ การสื่อสารระหว่าง Router กับ PLC Controller ที่ใช้ Modbus TCP/IP ด้วยการสร้างตู้สำหรับปลูก ต้นไม้ในที่ร่มอย่างง่าย

จากการทดสอบประสิทธิภาพการส่งของ AP Gateway ที่มี heartbeat time อยู่ที่ 3 นาที (มีการส่งข้อมูลทุกๆ 3 นาที) จะพบว่าใน 60 นาที สามารถจับการส่ง heartbeat ของตัว AP Gateway ได้ทั้งหมด 20 ครั้ง ซึ่งคิดเป็นเปอร์เซ็นต์ ได้ 100% และ ทดสอบประสิทธิภาพการส่งข้อมูล จาก Sensor โดยจะมี heartbeat time อยู่ที่ 3 นาที ตามที่เราได้ตั้งไว้ และทำการทดสอบจุดละ 30 นาที จะพบว่าเมื่อ AP Gateway อยู่ภายในอาคารที่มีสิ่งกีดขวางมาก Sensor จะสามารถส่งข้อมูลได้ ที่ระยะทางไกลสุด 403 เมตร แต่คุณภาพสัญญาณจะไม่ค่อยดีนักถ้าเทียบกับ Sensor ที่อยู่ใกล้กับ AP Gateway หรือเมื่อนำ AP Gateway มาติดตั้งไว้บริเวณบนดาดฟ้าของตึกภาควิชาโทรคมนาคม ก็จะได้คุณภาพสัญญาณที่ดีกว่าและได้ระยะทางที่ไกลมากขึ้นด้วยอยู่ที่ 555 เมตร ซึ่งจากการทดลองเมื่อ มาที่ระยะทางที่ไกลขึ้นจนทำให้ Sensor ไม่สามารถหา AP Gateway เพื่อเชื่อมต่อได้ Sensor จะขาด การเชื่อมต่อและพยายามเชื่อมต่ออีกครั้งถ้าเชื่อมต่อได้ Sensor จะส่งค่าที่เก็บไว้ทั้งหมดตามที่เดียว แต่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้ามีการเปิดใช้ Mote Sensor จะเชื่อมต่อกับ Mote แทน โดยระยะจาก AP Gateway ถึง Mote จะอยู่ที่ 300 เมตร และจาก Sensor ถึง Mote จะได้อยู่ที่ 350 เมตร ทำให้สามารถไปได้ไกลขึ้นได้

## 5.2 ปัญหาและข้อเสนอแนะ

1. เนื่องจากไม่มีความรู้และประสบการณ์ในการทำงานของอุปกรณ์ PLC Controller ทำให้การทำงานค่อนข้างใช้เวลามากในการศึกษาหลักการการทำงานหลายๆส่วน
2. การออกแบบการรับ-ส่งข้อมูลของ PLC Controller ต้องมีการตั้งค่าพารามิเตอร์ใน Data transfer information table ให้ถูกต้อง เพื่อที่จะสามารถรับข้อมูลที่ส่งมาจาก Router โดยใช้ Modbus TCP/IP
3. การตั้งค่า IP ของ PLC Controller และ Router ต้องอยู่ใน Network เดียวกันถึงจะสามารถเชื่อมต่อกันได้
4. การส่งค่าข้อมูลที่ได้จากเซ็นเซอร์ให้กับ PLC โดยใช้ Modbus TCP/IP ต้องส่งเป็นจำนวนเต็มเท่านั้นและถ้าต้องการแสดงผลบนหน้าจอ PLC Controller เราสามารถออกแบบให้แสดงเป็นทศนิยมได้จากโปรแกรม LOGO! 8.3 Software
5. การใช้ RS-485 ของทาง Zifisense เราต้องทำการสร้าง Address Code ให้ต่างกันระหว่าง 2 เซ็นเซอร์เพื่อให้สามารถที่จะอ่านค่าจาก Sensor ได้และเราก็ต้องทำการเข้าไปตั้งค่า Sensor ในการเปลี่ยน Slave ID ของ 2 Sensor โดยการเชื่อมต่อ Sensor กับ คอมพิวเตอร์และทำการเข้าไปแก้ไขผ่านโปรแกรม Modbus Poll ให้มีค่าต่างกันและตรงกับที่ตั้งไว้ในตัวอุปกรณ์ RS-485
6. การใช้อุปกรณ์ Router ของ Xiaomi Wi-Fi Range Extender AC1200 ที่จะนำมาทำการลง Firmware OpenWrt ครอบทับเข้าไปนั้นมีความยากลำบากเนื่องจากเมื่อเข้า Console Port ผ่าน Secure shell แล้วพอเครื่องบูตแล้วเราไม่สามารถ interrupt ได้เนื่องจากตัวอุปกรณ์ถูกตั้งมาไม่ให้อินเตอร์รัปต์ได้ทำให้เราต้องทำการใช้ EEPROM Flash BIOS รุ่น CH431A เพื่อเข้าไปแก้ไขโค้ด Hex ให้สามารถ interrupt ได้
7. เนื่องจาก Router ของ Xiaomi Wi-Fi Range Extender AC1200 ที่ทำการลง Firmware OpenWrt ครอบทับไปแล้วทำให้มีพื้นที่เหลือใช้ของตัวอุปกรณ์ค่อนข้างน้อยทำให้เราไม่สามารถที่จะลง Python ตัวเต็มได้ต้องเลือกเฉพาะแพ็คเกจที่จำเป็นต้องใช้ของ Python จริงๆเท่านั้นและเนื่องจากมีพื้นที่เหลือน้อยก็ส่งผลกับโปรแกรม Python ที่เราได้ทำการเขียนไว้ไม่สามารถใช้ได้เพราะเนื่องจาก Library บางตัวต้องใช้พื้นที่เยอะทำให้ไม่สามารถลงได้เลยต้องหาตัวที่เล็กกว่าในการทำงานแทนและบางตัวก็ต้องเขียน Python เองแทนการใช้ Library เพราะเนื่องจากความจำไม่พอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- [1] Aws.amazon. “IoT คืออะไร.” [ออนไลน์]. เข้าถึงได้จาก : <https://aws.amazon.com/th/what-is/iot/>. (วันที่ค้นข้อมูล : 9 กรกฎาคม 2566).
- [2] สถาบันนวัตกรรมและกรรมมาภิบาลข้อมูล. “เทคโนโลยี IoT คืออะไร.” [ออนไลน์]. เข้าถึงได้จาก : <https://digi.data.go.th/blog/what-is-iot-technology/>. (วันที่ค้นข้อมูล : 9 กรกฎาคม 2566).
- [3] DDproperty. “ทำไมต้องมี Internet of Things หรือ IoT.” [ออนไลน์]. เข้าถึงได้จาก : <https://www.ddproperty.com/คู่มือซื้อขาย/iot-คืออะไร-ทำไมต้องมีติดบ้าน-คอนโดในยุคดิจิทัล-33489>. (วันที่ค้นข้อมูล : 9 กรกฎาคม 2566).
- [4] Zifisense. “ZETA-LPWAN 2.0 technology.” [ออนไลน์]. เข้าถึงได้จาก : <http://zifisense.co.uk/index.html>. (วันที่ค้นข้อมูล : 10 กรกฎาคม 2566).
- [5] BoxSingle. “ทำความรู้จักกับ JSON คืออะไร.” [ออนไลน์]. เข้าถึงได้จาก : <https://www.boxsingle.com/13/ทำความรู้จักกับ-JSON-คืออะไร#>. (วันที่ค้นข้อมูล : 12 กรกฎาคม 2566).
- [6] Aws.amazon. “JavaScript คืออะไร.” [ออนไลน์]. เข้าถึงได้จาก : <https://aws.amazon.com/th/what-is/javascript/>. (วันที่ค้นข้อมูล : 12 กรกฎาคม 2566).
- [7] Aws.amazon. “MQTT คืออะไร.” [ออนไลน์]. เข้าถึงได้จาก : <https://aws.amazon.com/th/what-is/mqtt/>. (วันที่ค้นข้อมูล : 12 กรกฎาคม 2566).
- [8] Aws.amazon. “MySQL คืออะไร.” [ออนไลน์]. เข้าถึงได้จาก : <https://aws.amazon.com/th/rds/mysql/>. (วันที่ค้นข้อมูล : 14 กรกฎาคม 2566).
- [9] Saixiii. “MySQL คืออะไร? และ ไว้ทำอะไร?” [ออนไลน์]. เข้าถึงได้จาก : <https://saixiii.com/what-is-mysql/>. (วันที่ค้นข้อมูล : 14 กรกฎาคม 2566).
- [10] Zixzax Studio. “MySQL Database คืออะไร ?.” [ออนไลน์]. เข้าถึงได้จาก : <https://zixzax.net/database/mysql-database-คืออะไร/>. (วันที่ค้นข้อมูล : 14 กรกฎาคม 2566).
- [11] Piyadanai Wikein. “พื้นฐานภาษา HTML.” [ออนไลน์]. เข้าถึงได้จาก : <https://krupiyadanai.wordpress.com/บทเรียน-html/รู้จักภาษา-html/>. (วันที่ค้นข้อมูล : 10 สิงหาคม 2566).
- [12] Kipakaporn. “CSS คืออะไร มีประโยชน์อย่างไร.” [ออนไลน์]. เข้าถึงได้จาก : <https://blog.sogoodweb.com/Article/Detail/79237/CSS-คืออะไร-มีประโยชน์-อย่างไร>. (วันที่ค้นข้อมูล : 10 สิงหาคม 2566).

## บรรณานุกรม (ต่อ)


- [13] Codingonblog. “CSS คือ อะไร ? มือใหม่ อ่านจบครบทุกเรื่อง !.” [ออนไลน์]. เข้าถึงได้จาก : <https://codingonblog.com/basic-css-codingonblog/>. (วันที่ค้นข้อมูล : 10 สิงหาคม 2566).
- [14] Webdoodee. “PHP คืออะไร สามารถทำอะไรได้บ้าง และสิ่งที่ต้องการก่อนเริ่มใช้งาน.” [ออนไลน์]. เข้าถึงได้จาก : <https://www.webdoodee.com/what-is-php/>. (วันที่ค้นข้อมูล : 15 สิงหาคม 2566).
- [15] Mospichit. “เรื่อนำรู้ของภาษา PHP.” [ออนไลน์]. เข้าถึงได้จาก : <https://blog.sogoodweb.com/Article/Detail/52790/เรื่อนำรู้ของภาษา-PHP>. (วันที่ค้นข้อมูล : 15 สิงหาคม 2566).
- [16] NECTEC. “การสื่อสารในงานอุตสาหกรรมด้วยโพรโทคอล Modbus.” [ออนไลน์]. เข้าได้จาก : <https://www.nectec.or.th/news/news-public-document/modbus-protocol.html>. (วันที่ค้นข้อมูล : 25 สิงหาคม 2566).
- [17] OMEGA. “RS-485.” [ออนไลน์]. เข้าได้จาก : <https://www.omi.co.th/th/article/rs485>. (วันที่ค้นข้อมูล : 25 สิงหาคม 2566).
- [18] CloudHM. “DNS คืออะไร ? และมีความสำคัญอย่างไรต่อระบบของคุณ.” [ออนไลน์]. เข้าได้จาก : <https://blog.cloudhm.co.th/what-is-dns/>. (วันที่ค้นข้อมูล : 25 สิงหาคม 2566).
- [19] Quickserv. “DNS คืออะไร ทำไมต้องแปลงเป็น IP Address ด้วย.” [ออนไลน์]. เข้าได้จาก : <https://www.quickserv.co.th/knowledge-base/solutions/dns-คืออะไร-ทำไมต้องแปลงเป็น-IP-Address-ด้วย-/>. (วันที่ค้นข้อมูล : 25 สิงหาคม 2566).
- [20] FLUTECH. “PLC หรือ Programmable Logic Controller คืออะไร.” [ออนไลน์]. เข้าได้จาก : <https://flutech.co.th/what-is-a-plc-or-programmable-logic-controller/>. (วันที่ค้นข้อมูล : 5 กันยายน 2566).
- [21] SEIMENS. “Data sheet LOGO! 230 RCE.” [ออนไลน์]. เข้าได้จาก : <https://docs.rs-online.com/300a/0900766b8164240b.pdf>. (วันที่ค้นข้อมูล : 5 กันยายน 2566).
- [22] Manuals. “ZiFiSense D485ZT ZETA 485 เครื่องรับส่งสัญญาณ.” [ออนไลน์]. เข้าถึงได้จาก : <https://manuals.plus/th/zifisense/d485zt-zeta-485-transceiver-manual#axzz8IO87cC85>. (วันที่ค้นข้อมูล : 5 กันยายน 2566).

## บรรณานุกรม (ต่อ)

- [23] Zifisense “ZETA485Transceiver.” [ออนไลน์]. เข้าถึงได้จาก : <https://electric.garden/xiamen-zifisense-infotech-2ay22/manuals/D485ZT92-Users-Manual-Xiamen-ZiFiSense-InfoTech-2ay22-d485zt92-ex-1-8.pdf>. (วันที่ค้นข้อมูล : 5 กันยายน 2566).
- [24] CyberTice. “RS485 Soil Temperature and Humidity Sensor.” [ออนไลน์]. เข้าถึงได้จาก : <https://www.cybertice.com/product/4886/rs485-soil-temperature-and-humidity-sensor>. (วันที่ค้นข้อมูล : 8 กันยายน 2566).
- [25] Talil. “Raspberry Pi คืออะไร ? มีประโยชน์อย่างไร ทำไมถึงน่าใช้ ?.” [ออนไลน์]. เข้าได้จาก : <https://tips.thaiware.com/1813.html>. (วันที่ค้นข้อมูล : 8 กันยายน 2566).
- [26] Brian Yu. “การใช้งาน JSON ร่วมกับ Python ครบจบในบทความเดียว.” [ออนไลน์]. เข้าถึงได้จาก : <https://stackpython.co/tutorial/json-python>. (วันที่ค้นข้อมูล : 11 กันยายน 2566).
- [27] Supaluck Singjan. “PHP & MySQLi ดึงข้อมูลมาแสดงบนหน้าเว็บ [สำหรับมือใหม่มา ก ๆ].” [ออนไลน์]. เข้าถึงได้จาก : <https://dev.to/supaluckn/php-mysqli-4ke6>. (วันที่ค้นข้อมูล : 11 กันยายน 2566).
- [28] Miniature Solution. “เซนเซอร์อุณหภูมิ ความชื้น แสง 3 ใน 1 Light Transmitter/อุตสาหกรรม Lux Meter/RS485 ความสว่าง ตัววัดความเข้มแสง-65K.” [ออนไลน์]. เข้าถึงได้จาก : <https://www.miniature-solution.com/product>. (วันที่ค้นข้อมูล : 4 ธันวาคม 2566).
- [29] AllNewStep. “PL2303TA download cable USB to TTL RS232 module โมดูล USB TTL PL2303TA.” [ออนไลน์]. เข้าถึงได้จาก : <https://www.allnewstep.com/product/1624/pl2303ta-download-cable-usb-to-ttl-rs232-module-โมดูล-usb-ttl-pl2303ta>. (วันที่ค้นข้อมูล : 16 ธันวาคม 2566).
- [30] Kuongshun. “CH341A 24 25 ซีรีส์ EEPROM Flash BIOS โปรแกรม USB พร้อมซอฟต์แวร์และไดรเวอร์.” [ออนไลน์]. เข้าถึงได้จาก : <https://th.sz-kuongshun.com/uno/uno-board-shield/ch341a-24-25-series-eprom-flash-bios-usb-programm.html>. (วันที่ค้นข้อมูล : 18 ธันวาคม 2566).
- [31] Mi. “ตัวขยายระยะสัญญาณ Wi-Fi Mi AC1200.” [ออนไลน์]. เข้าถึงได้จาก : <https://www.mi.com/th/product/mi-wifi-range-extender-ac1200/specs/>. (วันที่ค้นข้อมูล : 12 มกราคม 2567).

## บรรณานุกรม (ต่อ)

- [32] rogerpueyo. “Xiaomi Mi Router 4A Gigabit Edition (R4AG/R4A Gigabit) – fully supported and flashable with OpenWRTInvasion.” [ออนไลน์]. เข้าถึงได้จาก : <https://forum.openwrt.org/t/xiaomi-mi-router-4a-gigabit-edition-r4ag-r4a-gigabit-fully-supported-and-flashable-with-openwrtinvasion/36685>. (วันที่ค้นข้อมูล : 12 มกราคม 2567).
- [33] openwrt. “Xiaomi Mi WiFi Range Extender AC1200 Model RA75.” [ออนไลน์]. เข้าถึงได้จาก : [https://openwrt.org/inbox/toh/xiaomi/mi\\_wifi\\_range\\_extender\\_ac1200\\_model\\_ra75#supported\\_versions](https://openwrt.org/inbox/toh/xiaomi/mi_wifi_range_extender_ac1200_model_ra75#supported_versions). (วันที่ค้นข้อมูล : 12 กุมภาพันธ์ 2567).
- [34] fccid. “RA75 Mi WiFi Range Extender AC1200 Teardown Internal Photos ATTACHMENT Xiaomi Communications.” [ออนไลน์]. เข้าถึงได้จาก : <https://fccid.io/2AFZZRA75/Internal-Photos/Internal-photos-5364844>. (วันที่ค้นข้อมูล : 14 กุมภาพันธ์ 2567).



ภาคผนวก ก

โปรแกรมการทำงานของารรับข้อมูลจาก Sensors โดยใช้ MQTT Protocol และ  
การสื่อสารระหว่าง OpenWrt กับ PLC Controller รวมถึงการเก็บค่าข้อมูลที่ต้องการ  
ลงในฐานข้อมูล (projectfinal.py)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#!/usr/bin/python
import paho.mqtt.client as mqtt
import pymysql
import random
import json
import queue
import time
import threading
import socket
import encodings.idna
import http.client

# Create Queue Handle
q = queue.Queue()

company_data = None
device = None
bit_data_Temphumid485 = None
Temp485soil = None
Humid485soil = None
Info_rs485soil = None
strdatetime = None
bit_air = None
TempAir = None
HumidAir = None
LuxAir = None
Info_Air = None
alldata = None

word_485 = "4f039a97"
broker = 'en-apis.zifisense.com'

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

port = 1883
topic = "980f70347bc145c8a0adb4a34a76b264/jll/property/ms/+/updata"
client_id = f'python-mqtt-{random.randint(0, 1000)}'
username = '980f70347bc145c8a0adb4a34a76b264'
password = 'a7ae992006b5486bbda172e1752303aa'

```

```

def connect_mqtt():
    def on_connect(client, userdata, flags, rc):
        if rc == 0:
            print("MQTT Connected")
        else:
            print("MQTT Not Connect")
    # Set Connecting Client ID
    client = mqtt.Client(client_id,protocol=mqtt.MQTTv31)
    print("Client ID: ",client_id)
    client.username_pw_set(username, password)
    client.on_connect = on_connect
    client.connect(broker, port)
    return client

```

```

def subscribe(client: mqtt):
    def on_message(client, userdata, msg):
        global q
        decoded_text = msg.payload.decode("utf-8")
        #print("decoded_text is ",decoded_text)
        try:
            json_res = json.loads(decoded_text)
            # print("json is ",json_res)
            q.put(json_res)
        except:
            print("The incoming data is not json!!")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

print(" ----- ")
return

client.subscribe(topic)
client.on_message = on_message

def
insert_data(CompanyCode,DeviceType,Data,Tempsoil,Humidsoil,Infofoil,Tempair,Humidair,Light,Infoair,Statuslight,Statusfan,Statuspump,Datetime,Bitstatuslight,Bitstatusfan,Bitstatuspump):

vibra_db = pymysql.connect(
    host="roundhouse.proxy.rlwy.net",
    port=32692,
    user="root",
    password="cbA5b511H2334bFDF-3a4ACEHA2GcBD2",
    database = "project"
)

db_cursor = vibra_db.cursor()

sql = "insert into
plclog8(CompanyCode,DeviceType,Data,Tempsoil,Humidsoil,Infofoil,Tempair,Humidair,Light,Infoair,Statuslight,Statusfan,Statuspump,Datetime,Bitstatuslight,Bitstatusfan,Bitstatuspump) values(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)"

val=(str(CompanyCode),str(DeviceType),str(Data),str(Tempsoil),str(Humidsoil),str(Infofoil),str(Tempair),str(Humidair),str(Light),str(Infoair),str(Statuslight),str(Statusfan),str(Statuspump),str(Datetime),str(Bitstatuslight),str(Bitstatusfan),str(Bitstatuspump))

db_cursor.execute(sql,val)
vibra_db.commit()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

vibra_db.close()

def insert_read(Inputlight,Inputfan,Inputpump):

    vibra_db = pymysql.connect(
        host="roundhouse.proxy.rlwy.net",
        port=32692,
        user="root",
        password="cbA5b511H2334bFDF-3a4ACEHA2GcBD2",
        database = "project"
    )

    db_cursor = vibra_db.cursor()

    sql = "insert into readinput(Inputlight,Inputfan,Inputpump) value(%s,%s,%s)"
    val=(str(Inputlight),str(Inputfan),str(Inputpump))
    db_cursor.execute(sql,val)

    vibra_db.commit()
    vibra_db.close()

def send_line_notify(token, message):
    conn = http.client.HTTPSConnection("notify-api.line.me")
    headers = {
        "Authorization": "Bearer " + token,
        "Content-Type": "application/x-www-form-urlencoded"
    }
    payload = "message=" + message
    conn.request("POST", "/api/notify", payload, headers)
    res = conn.getresponse()
    data = res.read()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

print(data.decode("utf-8"))

def processQueueTask(q):
    start = {}
    end = {}
    json_get = ""
    while(1):
        print("----- processTask -----")
        if (q.empty() == True):
            print("No any data is Queue!!")

            plc_ip = '192.168.1.103'
            plc_port = 502
            sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            sock.connect((plc_ip, plc_port))

            command1 = b'\x00\x01\x00\x00\x00\x06\x01\x02\x00\x01\x00\x01'
            #transactionid+protocolid+length+unitid+functioncode(decimal 2 = read input
            status)+address+quantity
            sock.send(command1)
            response1 = sock.recv(1024)
            input2 = response1[9]
            command2 = b'\x00\x01\x00\x00\x00\x06\x01\x02\x00\x03\x00\x01'
            #transactionid+protocolid+length+unitid+functioncode(decimal 2 = read input
            status)+address+quantity
            sock.send(command2)
            response2 = sock.recv(1024)
            input4 = response2[9]
            command3 = b'\x00\x01\x00\x00\x00\x06\x01\x02\x00\x05\x00\x01'
            #transactionid+protocolid+length+unitid+functioncode(decimal 2 = read input
            status)+address+quantity

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sock.send(command3)
response3 = sock.recv(1024)
input6 = response3[9]
sock.close()

if(input2 == 0):
    readinputlight = "OFF"
    print("Status Light : ",readinputlight)
if(input2 == 1):
    readinputlight = "ON"
    print("Status Light : ",readinputlight)
if(input4 == 0):
    readinputfan = "OFF"
    print("Status Fan : ",readinputfan)
if(input4 == 1):
    readinputfan = "ON"
    print("Status Fan : ",readinputfan)
if(input6 == 0):
    readinputpump = "OFF"
    print("Status Pump : ",readinputpump)
if(input6 == 1):
    readinputpump = "ON"
    print("Status Pump : ",readinputpump)
insert_read(readinputlight,readinputfan,readinputpump)

else:
    json_get = q.get()
    print("json_get = ",json_get)
    with open('data.text', 'w') as json_file:
        json.dump(json_get, json_file, indent=4)
    with open("data.text", "r") as file:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for line_number, line in enumerate(file, start=1):
    if word_485 in line:
        with open('data.text') as fs:
            data_filter = fs.read()
            data = json.loads(data_filter)
            bitdata = data['data']
            bitlave = str(bitdata[0:4])
            if bitlave == '0101':
                company_data = data['companyCode']
                print("CompanyCode : ",company_data)
                device = data['deviceAlias']
                print("DeviceType : ",device)
                bit_data_Temphumid485 = data['data']
                print('Data_Temp485 :',bit_data_Temphumid485)
                # Decode Bit Temp
                bit_Temp485 = (bit_data_Temphumid485[8:12])
                print('Bit_Temp485 :', bit_Temp485)
                decode_bit_temp485 = int(bit_Temp485,16)*0.1
                print('Temp485 : %.2f Celsius' %decode_bit_temp485)
                # Decode Bit Humid
                bit_humid485 = (bit_data_Temphumid485[4:8])
                print('Bit Humid485 : ',bit_humid485)
                decode_bit_humid485 = int(bit_humid485,16)*0.1
                print('Humid485 : {:.2f} RH%' .format(decode_bit_humid485))
                Temp485soil = "{:.2f}".format(decode_bit_temp485)
                Humid485soil = "{:.2f}".format(decode_bit_humid485)
                Info_rs485soil = 'Temp : {} Celcius , Humid : {}
                RH%' .format(Temp485soil,Humid485soil)
                unixtime = str(data['upTime'])
                unixtimereal = unixtime[0:10]
                print("unix time : ",unixtimereal)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datetime = time.localtime(int(unixtimereal))
strdatetime = (time.strftime("%Y-%m-%d %H:%M:%S",
datetime))

print(strdatetime)
if bitslave == '0102':
    bit_air = data['data']
    print('Data_Air :',bit_air)
    # Decode Bit Temp
    bit_Tempair = (bit_air[8:12])
    print('Bit Tempair :', bit_Tempair)
    decode_bit_tempair = int(bit_Tempair,16)*0.1
    print('Tempair : %.2f Celsius' %decode_bit_tempair)
    # Decode Bit Humid
    bit_humidair = (bit_air[4:8])
    print('Bit Humidair : ',bit_humidair)
    decode_bit_humidair = int(bit_humidair,16)*0.1
    print('Humidair : {:.2f} RH%' .format(decode_bit_humidair))
    bit_lux = bit_air[12:16]
    print('Bit Light : ', bit_lux)
    decode_bit_lux = int(bit_lux,16)*0.1
    print('Light : %.2f lux' %decode_bit_lux)
    TempAir = "{:.2f}".format(decode_bit_tempair)
    HumidAir = "{:.2f}".format(decode_bit_humidair)
    LuxAir = "{:.2f}" .format(decode_bit_lux)
    Info_Air = 'Temp : {} Celcius , Humid : {} RH% , Light : {}
lux'.format(TempAir,HumidAir,LuxAir)

alldata = str(bit_data_Temphumid485) + str(bit_air)
print("Total Data : ',alldata)

# PLC's IP address and Modbus TCP port
PLC_IP = '192.168.1.103'

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PLC_PORT = 502

# Modbus address (register number) to write data to
REGISTER_ADDRESS1 = 1
REGISTER_ADDRESS2 = 2
REGISTER_ADDRESS3 = 3
REGISTER_ADDRESS4 = 4
REGISTER_ADDRESS5 = 5

# Calculate data to send to the PLC (assuming 16-bit integers)
Temp485real = round(decode_bit_temp485, 1)
Tempuse = int(Temp485real * 10)
Humid485real = round(decode_bit_humid485, 1)
Humiduse = int(Humid485real * 10)
Tempairreal = round(decode_bit_tempair, 1)
Tempairuse = int(Tempairreal * 10)
Humidairreal = round(decode_bit_humidair, 1)
Humidairuse = int(Humidairreal * 10)
Lightreal = round(decode_bit_lux, 1)
Lightuse = int(Lightreal * 10)

# Data to send to the PLC
data_to_send1 = [Tempuse]
data_to_send2 = [Humiduse]
data_to_send3 = [Tempairuse]
data_to_send4 = [Humidairuse]
data_to_send5 = [Lightuse]

# Function to build Modbus TCP/IP message for writing
multiple registers

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

def
build_modbus_tcp_message_write_multiple_registers(slave_id, starting_address,
values):

    # Modbus TCP/IP Header
    transaction_id = b'\x00\x01'
    protocol_id = b'\x00\x00'
    unit_id = bytes([slave_id])

    # Modbus PDU (Protocol Data Unit)
    function_code = b'\x10' # Function code for writing
multiple registers (decimal 16)
    address_bytes = starting_address.to_bytes(2, byteorder='big')
    quantity_bytes = len(values).to_bytes(2, byteorder='big')
    byte_count = bytes([len(values) * 2]) # Each register is 2
bytes
    values_bytes = b''
    for value in values:
        value_bytes = value.to_bytes(2, byteorder='big')
        values_bytes += value_bytes

    # Build Modbus TCP/IP message
    modbus_pdu = function_code + address_bytes +
quantity_bytes + byte_count + values_bytes
    length = (len(unit_id) + len(modbus_pdu)).to_bytes(2,
byteorder='big')
    modbus_adu = transaction_id + protocol_id + length +
unit_id + modbus_pdu

    return modbus_adu

# Connect to the PLC and write data to registers

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

try:
    # Create a socket
    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as
sock:

        # Connect to the PLC
        sock.connect((PLC_IP, PLC_PORT))

        # Write data to the Modbus registers

sock.sendall(build_modbus_tcp_message_write_multiple_registers(1,
REGISTER_ADDRESS1, data_to_send1))
        print("Data sent successfully to address",
REGISTER_ADDRESS1, ":", data_to_send1)

sock.sendall(build_modbus_tcp_message_write_multiple_registers(1,
REGISTER_ADDRESS2, data_to_send2))
        print("Data sent successfully to address",
REGISTER_ADDRESS2, ":", data_to_send2)

sock.sendall(build_modbus_tcp_message_write_multiple_registers(1,
REGISTER_ADDRESS3, data_to_send3))
        print("Data sent successfully to address",
REGISTER_ADDRESS3, ":", data_to_send3)

        time.sleep(1)

sock.sendall(build_modbus_tcp_message_write_multiple_registers(1,
REGISTER_ADDRESS4, data_to_send4))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        print("Data sent successfully to address",
REGISTER_ADDRESS4, ":", data_to_send4)

sock.sendall(build_modbus_tcp_message_write_multiple_registers(1,
REGISTER_ADDRESS5, data_to_send5))

        print("Data sent successfully to address",
REGISTER_ADDRESS5, ":", data_to_send5)

    except Exception as e:
        print("Error:", e)

    ### Creating connection
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    sock.connect(('192.168.1.103', 502))

    def build_modbus_tcp_message_write_multiple_coils(slave_id,
starting_address, values):
        # Modbus TCP/IP Header
        transaction_id = b'\x00\x01'
        protocol_id = b'\x00\x00'
        unit_id = bytes([slave_id])

        # Modbus PDU (Protocol Data Unit)
        function_code = b'\x0F' # Function code for writing multiple
coil (decimal 15)

        address_bytes = starting_address.to_bytes(2, byteorder='big')
        quantity_bytes = len(values).to_bytes(2, byteorder='big')
        byte_count = bytes([len(values)])
        values_bytes = b"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for value in values:
    if value:
        values_bytes += b'\xFF'
    else:
        values_bytes += b'\x00'

# Build Modbus TCP/IP message
modbus_pdu = function_code + address_bytes +
quantity_bytes + byte_count + values_bytes
length = (len(unit_id) + len(modbus_pdu)).to_bytes(2,
byteorder='big')
modbus_adu = transaction_id + protocol_id + length +
unit_id + modbus_pdu
return modbus_adu

# Create a message to write multiple coils
message =
build_modbus_tcp_message_write_multiple_coils(slave_id=1, starting_address=0,
values=[1])
message2 =
build_modbus_tcp_message_write_multiple_coils(slave_id=1, starting_address=0,
values=[0])
message3 =
build_modbus_tcp_message_write_multiple_coils(slave_id=1, starting_address=1,
values=[1])
message4 =
build_modbus_tcp_message_write_multiple_coils(slave_id=1, starting_address=1,
values=[0])

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

message5 =
build_modbus_tcp_message_write_multiple_coils(slave_id=1, starting_address=2,
values=[1])

message6 =
build_modbus_tcp_message_write_multiple_coils(slave_id=1, starting_address=2,
values=[0])

# Send the message and get the response
if round(Lightreal)<10:
    sock.sendall(message)
    statuslight = "ON"
    bitstatuslight = "1"
    print("send {} to address 0 ".format(bitstatuslight))
    line_token =
"jdWiumVgKb56laPY5NqVLmPFSWLvkaemm8BwpaNJsLU"
    message_to_send = "Light : {} lux and Status LED : {}
".format(Lightreal,statuslight)
    send_line_notify(line_token, message_to_send)
if round(Lightreal)>=10:
    sock.sendall(message2)
    statuslight = "OFF"
    bitstatuslight = "0"
    print("send {} to address 0 ".format(bitstatuslight))
if round(Tempairreal)>30:
    sock.sendall(message3)
    statusfan = "ON"
    bitstatusfan = "1"
    print("send {} to address 1 ".format(bitstatusfan))
    line_token =
"jdWiumVgKb56laPY5NqVLmPFSWLvkaemm8BwpaNJsLU"
    message_to_send = "Temp(Air) : {} Celcius , Humid(Air) : {}
RH(Percent) and Status Fan : {}".format(Tempairreal,Humidairreal,statusfan)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        send_line_notify(line_token, message_to_send)
    if round(Tempairreal)<=30:
        sock.sendall(message4)
        statusfan = "OFF"
        bitstatusfan = "0"
        print("send {} to address 1 ".format(bitstatusfan))
    if round(Humid485real)<50:
        sock.sendall(message5)
        time.sleep(1)
        sock.sendall(message6)
        statuspump = "ON"
        bitstatuspump = "1"
        print("send {} to address 2 ".format(bitstatuspump))
    line_token =
"jdWiumVgKb56laPY5NqVLmPFSWLvkaemm8BwpaNjSlU"
    message_to_send = "Temp(Soil) : {} Celcius , Humid(Soil) : {}
RH(Percent) and Status Pump : {}".format(Temp485real,Humid485real,statuspump)
    send_line_notify(line_token, message_to_send)
    if round(Humid485real)>=50:
        sock.sendall(message6)
        statuspump = "OFF"
        bitstatuspump = "0"
        print("send {} to address 2 ".format(bitstatuspump))

# Close the socket connection
sock.close()
print("Transfer finished")

insert_data(company_data,device,alldata,Temp485soil,Humid485soil,Info_rs485soil,TempAir,HumidAir,LuxAir,Info_Air,statuslight,statusfan,statuspump,strdatetime,bitstatuslight,bitstatusfan,bitstatuspump)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
time.sleep(13)# delay for 13 sec.  
return  
  
worker = threading.Thread(target=processQueueTask, args=(q,), daemon=True) #  
create python thread (like xTaskCreate in platform IO)  
worker.start()  
  
client = connect_mqtt()  
subscribe(client)  
client.loop_forever()
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข

โปรแกรมสำหรับหน้าต่าง Login Website (สำหรับ User และ Admin)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมสำหรับหน้าต่าง Login Website (สำหรับ User)

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login</title>
</head>
<style>
  body {
    font-family: Arial, sans-serif;
    background-image : url('login2.jpg');
    background-repeat : no-repeat;
    margin: 0;
    padding: 0;
    display: flex;
    align-items: center;
    justify-content: center;
    height: 100vh;
  }

  .login-container {
    background-color: #f6948e;
    border-radius: 8px;
    padding: 20px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    width: 300px;
    text-align: center;
  }

  .login-container h2 {
    margin-bottom: 20px;
  }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

.login-form input {
  width: 100%;
  padding: 10px;
  margin-bottom: 15px;
  box-sizing: border-box;
}

```

```

.login-form button {
  width: 100%;
  padding: 10px;
  background-color: #4d89ca;
  color: #fff;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

```

```

.login-form button:hover {
  background-color: #444b8e;
}

```

```

.login-email {
  margin-top: 10px;
  margin-left : 50px;
}

```

```

.login-email a {
  text-decoration: none;
  color: #333;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        display: block;
        background-color: #f6948e;
        padding: 10px;
        border-radius: 4px;
        text-align: center;
    }

    .login-email a:hover {
        background-color: #4d89ca;
    }
</style>
</head>
<body>
<div class="login-container">
    <h1>Login</h1>
    <form class="login-form" action="Dashboarduserlogin.php" method="post"
onsubmit="return validateLogin()">
        <input type="email" name="email" id="email" placeholder="Enter your email"
required>
        <button type="submit">Login</button>
    </form>
</div>

<div class="login-email">
    <a href="Login.php"><font size = "6"><b>Login By Admin</b></font></a>
</div>

<script>
    function validateLogin() {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// Since you don't want to check the email, simply return true
return true;
}
</script>
</body>
</html>
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมสำหรับหน้าต่าง Login Website (สำหรับ Admin)

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login</title>
</head>
<body>
  <div class="login-container">
    <h2>Login</h2>
  </div>
</body>
</html>

```

```

body {
  font-family: Arial, sans-serif;
  background-image : url('login2.jpg');
  background-repeat : no-repeat;
  margin: 0;
  padding: 0;
  display: flex;
  align-items: center;
  justify-content: center;
  height: 100vh;
}

.login-container {
  background-color: #f6948e;
  border-radius: 8px;
  padding: 20px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  width: 300px;
  text-align: center;
}

.login-container h2 {
  margin-bottom: 20px;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

.login-form input {
  width: 100%;
  padding: 10px;
  margin-bottom: 15px;
  box-sizing: border-box;
}

.login-form button {
  width: 100%;
  padding: 10px;
  background-color: #4d89ca;
  color: #fff;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

.login-form button:hover {
  background-color: #444b8e;
}

.login-email {
  margin-top: 10px;
  margin-left : 50px;
}

.login-email a {
  text-decoration: none;
  color: #333;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

display: block;
background-color: #f6948e;
padding: 10px;
border-radius: 4px;
text-align: center;
}

.login-email a:hover {
background-color: #4d89ca;
}

</style>
</head>
<body>
<div class="login-container">
  <h1>Login</h1>
  <form class="login-form" action="" method="post" onsubmit="return
validateLogin()">
    <input type="text" name="username" id="username" placeholder="Username"
required>
    <input type="password" name="password" id="password"
placeholder="Password" required>
    <button type="submit">Login</button>
  </form>
</div>

<div class="login-email">
  <a href="index.php"><font size ="6"><b>Login By Email</b></font></a>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<script>
  function validateLogin() {
    var username = document.getElementById('username').value;
    var password = document.getElementById('password').value;
    var form = document.querySelector('.login-form');

    if (username === 'admin' && password === 'Logo!8') {
      form.action = 'Dashboard.php';
      return true; // อนุญาตให้ส่งฟอร์ม
    } else if (username === 'company1' && password === 'Logo!8'){
      form.action = 'Dashboardcompany1.php';
      return true;
    } else if (username === 'company2' && password === 'Logo!8'){
      form.action = 'Dashboardcompany2.php';
      return true;
    } else {
      alert('Invalid username or password');
      return false; // ป้องกันการส่งฟอร์ม
    }
  }
</script>
</body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ค

โปรแกรมสำหรับหน้าต่าง Dashboard Website (User และ Admin)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมสำหรับหน้าต่าง Dashboard Website (สำหรับ User)

```

<?php
// เชื่อมต่อกับ MySQL
$servername = "roundhouse.proxy.rlwy.net:32692";
$username = "root";
$password = "cbA5b511H2334bFDF-3a4ACEHA2GcBD2";
$dbname = "project";

// สร้างการเชื่อมต่อ
$conn = new mysqli($servername, $username, $password, $dbname);

// ตรวจสอบการเชื่อมต่อ
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// รับค่า email จากฟอร์ม
$email = $_POST['email'];

if (!empty($email)) {
    // เพิ่มข้อมูลลงในตาราง users
    $sql = "INSERT INTO user(EmailUser) VALUES ('$email')";

    $result = $conn->query($sql);

    // ส่วนที่เพิ่ม: ส่งข้อความไลน์แจ้งเตือน
    $lineNotifyToken = "Pt5DKEnMroHtQL8ltafx0ljXvg3CUltjBZijXLTUpI";
    $message = "มีการเข้าสู่ระบบด้วย Email: $email";

    $lineNotifyAPI = "https://notify-api.line.me/api/notify";
    $headers = array(

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

"Authorization: Bearer $lineNotifyToken",
"Content-Type: application/x-www-form-urlencoded"
);

$data = array('message' => $message);

$options = array(
    'http' => array(
        'header' => implode("\r\n", $headers),
        'method' => 'POST',
        'content' => http_build_query($data),
    ),
);

$context = stream_context_create($options);
$result = file_get_contents($lineNotifyAPI, false, $context);

// ปิดการเชื่อมต่อ MySQL
$conn->close();
}
?>

<?php
$con =
mysqli_connect("roundhouse.proxy.rlwy.net:32692","root","cbA5b511H2334bFDF-
3a4ACEHA2GcBD2","project");
$sql1 ="select* from plclogo8 order by Datetime desc limit 1";
$result1 = mysqli_query($con,$sql1);
while ($row = mysqli_fetch_array($result1)) {
    $datetime = date_format(date_create( $row['Datetime']),"l d-m-Y H:i:s") ;
    $infooil = $row['Infooil'];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    $infoair = $row['Infoair'];
    $statuslight = $row['Statuslight'];
    $statusfan = $row['Statusfan'];
    $statuspump = $row['Statuspump'];
}
$sql2 ="select * from readinput order by Id desc limit 1";
$result2 = mysqli_query($con,$sql2);
while ($row = mysqli_fetch_array($result2)) {
    $readlight = $row['Inputlight'];
    $readfan = $row['Inputfan'];
    $readpump = $row['Inputpump'];
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>APA-Control</title>

    <link rel="stylesheet" href="web.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.3.0/css/all.min.css">
</head>

<body>
    <script src="https://code.jquery.com/jquery-3.5.1.js"></script>
    <div class="page">
        <div class="sidebar">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<font size ="5"><center><h1>Dashboard</h1></font></center>
<div class="sidebar-img">
  
</div>

<div class="menu">
  <b class="cg">Menu</b>
  <div class="Item Item-active">
    <div class="sidebar-icon">
      <i class="fa-solid fa-info"></i>
    </div>
    <font size ="5"><b>Dashboard</b></font>
  </div>
  <div class="Item">
    <div class="sidebar-icon">
      <i class="fa-solid fa-eject"></i>
    </div>
    <a href="Aboutuser.php"><font size ="5"><b>About Project</b></font></a>
  </div>
  <div class="Item">
    <div class="sidebar-icon">
      <i class="fa-solid fa-people-group"></i>
    </div>
    <a href="Memberuser.php"><font size ="5"><b>Members</b></font></a>
  </div>
</div>
<div class="end"></div>
<div class="logout-button">
  <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div class="clock-box">
  <p id="clock"></p>
</div>
<script>
function updateTime() {
  var now = new Date();
  var hours = now.getHours();
  var minutes = now.getMinutes();
  var seconds = now.getSeconds();

  // Add leading zeros to minutes and seconds
  minutes = (minutes < 10 ? "0" : "") + minutes;
  seconds = (seconds < 10 ? "0" : "") + seconds;

  // Determine AM/PM notation
  var ampm = (hours < 12 ? "AM" : "PM");
  hours = (hours > 12 ? hours - 12 : hours);
  hours = (hours == 0 ? 12 : hours);

  // Format the time
  var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

  // Update the clock element
  document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">

```

```

<br><br><br>
<br><br><br>
<br><br><br><br><br>
<center></center>
</div>
</div>
<div class="content">
<div class="nav-bar">
<div class="search">
<a href="" target="_blank" hidden></a>
<input type="text" placeholder="Search">
<div class="autocom-box">
<!-- here list are inserted from javascript -->
</div>
<div class="icon"><i class="fas fa-search"></i></div>
</div>
</div>
<script src="js/script.js"></script>
<script src="js/suggestions.js"></script>
<div class="content-space">
<font size = "3"><center><h1>My Project x Zifisense</h1></center></font>
<div class="image">
<div class = "image-item image-one">
</div>
<div class="video">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<iframe width="100%" height="115%"
src="https://www.youtube.com/embed/2d2NnEnHtTQ" title="YouTube video player"
frameborder="10px" allow="accelerometer; autoplay; clipboard-write; encrypted-
media; gyroscope; picture-in-picture; web-share" allowfullscreen></iframe>
</div>
</div>
<div class="end2"></div>
<div
class="wordinformation"><center><h1>Informations</h1></center></div>
<div class="information">
<div class="frame">
<div class="container">
<div class="header">
<h1>Soil Sensor</h1>

</div>
<div class="contentinformation">
<div class="left-content">
<!-- เนื้อหาด้านซ้าย -->
<h2><center>Info</center></h2>
<div class="end4"></div><br>
<p><center><?php echo $info soil ?></center></p>
</div>
<div class="right-content">
<!-- เนื้อหาด้านขวา -->
<h2><center>Datetime</center></h2>
<div class="end4"></div><br>
<p><center><?php echo $datetime ?></center></p>
</div>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>
</div>
<div class="frame">
  <div class="container">
    <div class="header">
      <!-- รูปภาพด้านบน -->
      <h1>Air Sensor</h1>
      
    </div>
    <div class="contentinformation">
      <div class="left-content">
        <!-- เนื้อหาด้านซ้าย -->
        <h2><center>Info</center></h2>
        <div class="end4"></div><br>
        <p><center><?php echo $infoair ?></center></p>
      </div>
      <div class="right-content">
        <!-- เนื้อหาด้านขวา -->
        <h2><center>Datetime</center></h2>
        <div class="end4"></div><br>
        <p><center><?php echo $datetime ?></center></p>
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
<div class="wordstatus"><center><h1>Status</h1></center></div>
<div class="status">
  <div class="framestatus">
    <div class="statusdevice">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```


<h2>LED</h2>
<div class="commandstatus">
  <div class="wordcommand">
    <h3>Commanded</h3>
  </div>
  <div class="rounded-box">
    <?php echo ($statuslight == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
  </div>
</div>
<div class="readstatus">
  <div class="wordread">
    <h3>Recent Status</h3><br>
  </div>
  <div class="rounded-box">
    <?php echo ($readlight == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
  </div>
</div>
</div>
<div class="framestatus">
  <div class="statusdevice">
    

```
<h2>Water Pump</h2>
```

```
<div class="commandstatus">
```

```
<div class="wordcommand">
```

```
<h3>Commanded</h3>
```

```
</div>
```

```
<div class="rounded-box">
```

```
<?php echo ($statuspump == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
```

```
</div>
```

```
</div>
```

```
<div class="readstatus">
```

```
<div class="wordread">
```

```
<h3>Recent Status</h3><br>
```

```
</div>
```

```
<div class="rounded-box">
```

```
<?php echo ($readpump == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="framestatus">
```

```
<div class="statusdevice">
```

```

```

```
<h2>Fan</h2>
```

```
<div class="commandstatus">
```

```
<div class="wordcommand">
```

```
<h3>Commanded</h3>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## โปรแกรมสำหรับหน้าต่าง Dashboard Website (สำหรับ Admin)

```

<?php
$con =
mysqli_connect("roundhouse.proxy.rlwy.net:32692","root","cbA5b511H2334bFDF-
3a4ACEHA2GcBD2","project");
$sql1 ="select * from plclogo8 order by Datetime desc limit 1";

$result1 = mysqli_query($con,$sql1);
while ($row = mysqli_fetch_array($result1)) {
    $datetime = date_format(date_create( $row['Datetime']),"l d-m-Y H:i:s" );
    $infooil = $row['Infooil'];
    $infoair = $row['Infoair'];
    $statuslight = $row['Statuslight'];
    $statusfan = $row['Statusfan'];
    $statuspump = $row['Statuspump'];
}
$sql2 ="select * from readinput order by Id desc limit 1";
$result2 = mysqli_query($con,$sql2);
while ($row = mysqli_fetch_array($result2)) {
    $readlight = $row['Inputlight'];
    $readfan = $row['Inputfan'];
    $readpump = $row['Inputpump'];
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<title>APA-Control</title>

<link rel="stylesheet" href="web.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.3.0/css/all.min.css">
</head>

<body>
<script src="https://code.jquery.com/jquery-3.5.1.js"></script>
<div class="page">
<div class="sidebar">
<font size ="5"><center><h1>Dashboard</h1></font></center>
<div class="sidebar-img">

</div>
<div class="menu">
<b class="cg">Menu</b>
<div class="Item Item-active">
<div class="sidebar-icon">
<i class="fa-solid fa-info"></i>
</div>
<font size ="5"><b>Dashboard</b></font>
</div>
<div class="Item">
<div class="sidebar-icon">
<i class="fa-solid fa-eject"></i>
</div>
<a href="About.php"><font size ="5"><b>About Project</b></font></a>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-table"></i>
  </div>
  <a href="History.php"><font size ="5"><b>History</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-people-group"></i>
  </div>
  <a href="Member.php"><font size ="5"><b>Members</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-chart-line"></i>
  </div>
  <a href="Graph.php"><font size ="5"><b>Graphs</b></font></a>
</div>
</div>
<div class="end"></div>
<div class="logout-button">
  <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>
<div class="clock-box">
  <p id="clock"></p>
</div>
<script>
function updateTime() {
  var now = new Date();
  var hours = now.getHours();
  var minutes = now.getMinutes();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

var seconds = now.getSeconds();

// Add leading zeros to minutes and seconds
minutes = (minutes < 10 ? "0" : "") + minutes;
seconds = (seconds < 10 ? "0" : "") + seconds;

// Determine AM/PM notation
var ampm = (hours < 12 ? "AM" : "PM");
hours = (hours > 12 ? hours - 12 : hours);
hours = (hours == 0 ? 12 : hours);

// Format the time
var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

// Update the clock element
document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">
    <br><br><br>
    <br><br><br>
    <br><br><br><br><br>

```

```

<center></center>
</div>
</div>
<div class="content">
<div class="nav-bar">
<div class="search">
<a href="" target="_blank" hidden></a>
<input type="text" placeholder="Search">
<div class="autocom-box">
<!-- here list are inserted from javascript -->
</div>
<div class="icon"><i class="fas fa-search"></i></div>
</div>
</div>
<script src="js/script.js"></script>
<script src="js/suggestions.js"></script>
<div class="content-space">
<font size ="3"><center><h1>My Project x Zifisense</h1></center></font>
<div class="image">
<div class ="image-item image-one">
</div>
<div class="video">
<iframe width="100%" height="115%"
src="https://www.youtube.com/embed/2d2NnEnHtTQ" title="YouTube video player"
frameborder="10px" allow="accelerometer; autoplay; clipboard-write; encrypted-
media; gyroscope; picture-in-picture; web-share" allowfullscreen></iframe>
</div>
</div>
<div class="end2"></div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div
class="wordinformation"><center><h1>Informations</h1></center></div>
<div class="information">
<div class="frame">
<div class="container">
<div class="header">
<h1>Soil Sensor</h1>

</div>
<div class="contentinformation">
<div class="left-content">
<!-- เนื้อหาด้านซ้าย -->
<h2><center>Info</center></h2>
<div class="end4"></div><br>
<p><center><?php echo $info soil ?></center></p>
</div>
<div class="right-content">
<!-- เนื้อหาด้านขวา -->
<h2><center>Datetime</center></h2>
<div class="end4"></div><br>
<p><center><?php echo $datetime ?></center></p>
</div>
</div>
</div>
</div>
<div class="frame">
<div class="container">
<div class="header">
<!-- รูปภาพด้านบน -->
<h1>Air Sensor</h1>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```


</div>
<div class="contentinformation">
  <div class="left-content">
    <!-- เนื้อหาด้านซ้าย -->
    <h2><center>Info</center></h2>
    <div class="end4"></div><br>
    <p><center><?php echo $infoair ?></center></p>
  </div>
  <div class="right-content">
    <!-- เนื้อหาด้านขวา -->
    <h2><center>Datetime</center></h2>
    <div class="end4"></div><br>
    <p><center><?php echo $datetime ?></center></p>
  </div>
</div>
</div>
</div>
</div>
<div class="wordstatus"><center><h1>Status</h1></center></div>
<div class="status">
  <div class="framestatus">
    <div class="statusdevice">
      
      <h2>LED</h2>
      <div class="commandstatus">
        <div class="wordcommand">
          <h3>Commanded</h3>
        </div>
        <div class="rounded-box">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <?php echo ($statuslight == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
    </div>
</div>
<div class="readstatus">
    <div class="wordread">
        <h3>Recent Status</h3><br>
    </div>
    <div class="rounded-box">
        <?php echo ($readlight == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
    </div>
</div>
</div>
</div>
<div class="framestatus">
    <div class="statusdevice">
        
        <h2>Water Pump</h2>
        <div class="commandstatus">
            <div class="wordcommand">
                <h3>Commanded</h3>
            </div>
            <div class="rounded-box">

```

```

        <?php echo ($statuspump == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
    </div>
</div>
<div class="readstatus">
    <div class="wordread">
        <h3>Recent Status</h3><br>
    </div>
    <div class="rounded-box">
        <?php echo ($readpump == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
    </div>
</div>
</div>
</div>
<div class="framestatus">
    <div class="statusdevice">
        
        <h2>Fan</h2>
        <div class="commandstatus">
            <div class="wordcommand">
                <h3>Commanded</h3>
            </div>
            <div class="rounded-box">
                <?php echo ($statusfan == 'ON') ? '<h1 style="color:
#00FF00;">ON</h1>' : '<h1 style="color: #FF0000;">OFF</h1>'; ?>
            </div>
        </div>
    </div>
</div>
<div class="readstatus">
    <div class="wordread">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





ภาคผนวก ง  
โปรแกรมสำหรับหน้าต่าง About Website (สำหรับ User และ Admin)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมสำหรับหน้าต่าง About Website (สำหรับ User)

```

<!DOCTYPE html>

<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>APA-Control</title>

  <link rel="stylesheet" href="web.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.3.0/css/all.min.css"
</head>

<body>

  <div class="page">
    <div class="sidebar">
      <font size = "5"><center><h1>About Project</h1></font></center>
      <div class="sidebar-img">
        
      </div>

      <div class="menu">
        <b class="cg">Menu</b>
        <div class="Item">
          <div class="sidebar-icon">
            <i class="fa-solid fa-info"></i>
          </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <a href="Dashboarduser.php"><font size
="5"><b>Dashboard</b></font></a>
    </div>
    <div class="Item Item-active" >
        <div class="sidebar-icon">
            <i class="fa-solid fa-eject"></i>
        </div>
        <font size ="5"><b>About Project</b></font>
    </div>
    <div class="Item">
        <div class="sidebar-icon">
            <i class="fa-solid fa-people-group"></i>
        </div>
        <a href="Memberuser.php"><font size ="5"><b>Members</b></font></a>
    </div>
</div>
<div class="end"></div>
<div class="logout-button">
    <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>
<div class="clock-box">
    <p id="clock"></p>
</div>
<script>
function updateTime() {
    var now = new Date();
    var hours = now.getHours();
    var minutes = now.getMinutes();
    var seconds = now.getSeconds();

    // Add leading zeros to minutes and seconds

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

minutes = (minutes < 10 ? "0" : "") + minutes;
seconds = (seconds < 10 ? "0" : "") + seconds;

// Determine AM/PM notation
var ampm = (hours < 12 ? "AM" : "PM");
hours = (hours > 12 ? hours - 12 : hours);
hours = (hours == 0 ? 12 : hours);

// Format the time
var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

// Update the clock element
document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">
  <br><br><br>
  <br><br><br>
  <br><br><br><br><br>
  <center></center>
</div>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div class="content">
  <div class="nav-bar">
    <div class="search">
      <a href="" target="_blank" hidden></a>
      <input type="text" placeholder="Search">
      <div class="autocom-box">
        <!-- here list are inserted from javascript -->
      </div>
      <div class="icon"><i class="fas fa-search"></i></div>
    </div>
  </div>
  <script src="js/script.js"></script>
  <script src="js/suggestions.js"></script>
  <div class="content-space">
    <font size = "3"><center><h1>Using PLC Controller with IoT systems to control
the electronic devices</h1></center></font>
    <div class="info">
      <h2>&emsp;&emsp;&emsp;ทำการออกแบบระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าจากการใช้
งาน PLC Controller ร่วมกับระบบ IoT โดยมีการใช้งาน Sensor ร่วมกับ RS-485 เพื่อส่งข้อมูลผ่าน
AP Gateway ขึ้นสู่ระบบ Zeta Cloud หลังจากนั้นจะใช้ MQTT Protocol บน Router ที่ติดตั้ง
Firmware OpenWrt ในการ Subscribe ข้อมูลและทำการแปลงบิตข้อมูลที่ได้รับเป็นตัวเลขฐานสิบ
มาบันทึกไว้บน Database ของ MySQL Server หลังจากนั้นจึงนำข้อมูลต่างๆมาแสดงผลบน
Website ที่ได้ทำการสร้างขึ้น และจะทำการเชื่อมต่อ PLC Controller กับ Router ผ่านระบบ LAN
ซึ่ง PLC Controller จะใช้งาน Modbus TCP/IP ในการอ่านค่าหรือรับค่าจากอุปกรณ์อื่นและนำ
ข้อมูลที่ได้มาเก็บไว้ที่ตัว PLC Controller แล้วจึงทำการออกแบบโปรแกรมบน LOGO! 8.3
Software ในการสร้างการแสดงผลข้อมูลที่ได้บนหน้าจอ PLC Controller และ ควบคุมอุปกรณ์
Output ที่ต่ออยู่ โดยนำข้อมูลที่ได้มาทำการประมวลผล และสร้างเงื่อนไขในการควบคุมอุปกรณ์
Output ที่เชื่อมต่ออยู่กับ PLC Controller ให้สามารถทำงานได้ และจะทำการแสดงตัวอย่างการ
ทำงานร่วมกันของระบบทั้งหมดโดยทำการจำลองเป็นตู้สำหรับปลุกต้นไม้ในที่ร่มอย่างง่าย</h2>
    </div>
  <div class="end3"></div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## โปรแกรมสำหรับหน้าต่าง About Website (สำหรับ Admin)

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>APA-Control</title>

  <link rel="stylesheet" href="web.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.3.0/css/all.min.css"
</head>

<body>

  <div class="page">
    <div class="sidebar">
      <font size ="5"><center><h1>About Project</h1></font></center>
      <div class="sidebar-img">
        
      </div>

      <div class="menu">
        <b class="cg">Menu</b>
        <div class="Item">
          <div class="sidebar-icon">
            <i class="fa-solid fa-info"></i>
          </div>
          <a href="Dashboard.php"><font size ="5"><b>Dashboard</b></font></a>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>
<div class="Item Item-active" >
  <div class="sidebar-icon">
    <i class="fa-solid fa-eject"></i>
  </div>
  <font size ="5"><b>About Project</b></font>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-table"></i>
  </div>
  <a href="History.php"><font size ="5"><b>History</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-people-group"></i>
  </div>
  <a href="Member.php"><font size ="5"><b>Members</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-chart-line"></i>
  </div>
  <a href="Graph.php"><font size ="5"><b>Graphs</b></font></a>
</div>
</div>
<div class="end"></div>
<div class="logout-button">
  <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>
<div class="clock-box">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<p id="clock"></p>
</div>
<script>
function updateTime() {
    var now = new Date();
    var hours = now.getHours();
    var minutes = now.getMinutes();
    var seconds = now.getSeconds();

    // Add leading zeros to minutes and seconds
    minutes = (minutes < 10 ? "0" : "") + minutes;
    seconds = (seconds < 10 ? "0" : "") + seconds;

    // Determine AM/PM notation
    var ampm = (hours < 12 ? "AM" : "PM");
    hours = (hours > 12 ? hours - 12 : hours);
    hours = (hours == 0 ? 12 : hours);

    // Format the time
    var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

    // Update the clock element
    document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">

```

```

<br><br><br>
<br><br><br>
<br><br><br><br><br>
<center></center>
</div>
</div>
<div class="content">
<div class="nav-bar">
<div class="search">
<a href="" target="_blank" hidden></a>
<input type="text" placeholder="Search">
<div class="autocom-box">
<!-- here list are inserted from javascript -->
</div>
<div class="icon"><i class="fas fa-search"></i></div>
</div>
</div>
<script src="js/script.js"></script>
<script src="js/suggestions.js"></script>
<div class="content-space">
<font size ="3"><center><h1>Using PLC Controller with IoT systems to control
the electronic devices</h1></center></font>
<div class="info">
<h2>&emsp;&emsp;&emsp;ทำการออกแบบระบบควบคุมอุปกรณ์ที่ใช้ไฟฟ้าจากการใช้
งาน PLC Controller ร่วมกับระบบ IoT โดยมีการใช้งาน Sensor ร่วมกับ RS-485 เพื่อส่งข้อมูลผ่าน

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





ภาคผนวก จ

โปรแกรมสำหรับหน้าต่าง Members Website (สำหรับ User และ Admin)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมสำหรับหน้าต่าง Members Website (สำหรับ User)

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>APA-Control</title>

  <link rel="stylesheet" href="web.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.3.0/css/all.min.css">
</head>

<body>

  <div class="page">
    <div class="sidebar">
      <font size = "5"><center><h1>Group Members</h1></font></center>
      <div class="sidebar-img">
        
      </div>

      <div class="menu">
        <b class="cg">Menu</b>
        <div class="Item">
          <div class="sidebar-icon">
            <i class="fa-solid fa-info"></i>
          </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <a href="Dashboarduser.php"><font size
="5"><b>Dashboard</b></font></a>
    </div>
    <div class="Item">
        <div class="sidebar-icon">
            <i class="fa-solid fa-eject"></i>
        </div>
        <a href="Aboutuser.php"><font size ="5"><b>About Project</b></font></a>
    </div>
    <div class="Item Item-active">
        <div class="sidebar-icon">
            <i class="fa-solid fa-people-group"></i>
        </div>
        <font size ="5"><b>Members</b></font>
    </div>
</div>
<div class="end"></div>
<div class="logout-button">
    <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>
<div class="clock-box">
    <p id="clock"></p>
</div>
<script>
function updateTime() {
    var now = new Date();
    var hours = now.getHours();
    var minutes = now.getMinutes();
    var seconds = now.getSeconds();

    // Add leading zeros to minutes and seconds

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

minutes = (minutes < 10 ? "0" : "") + minutes;
seconds = (seconds < 10 ? "0" : "") + seconds;

// Determine AM/PM notation
var ampm = (hours < 12 ? "AM" : "PM");
hours = (hours > 12 ? hours - 12 : hours);
hours = (hours == 0 ? 12 : hours);

// Format the time
var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

// Update the clock element
document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">
  <br><br><br>
  <br><br><br>
  <br><br><br><br><br>
  <center></center>
</div>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div class="content">
  <div class="nav-bar">
    <div class="search">
      <a href="" target="_blank" hidden></a>
      <input type="text" placeholder="Search">
      <div class="autocom-box">
        <!-- here list are inserted from javascript -->
      </div>
      <div class="icon"><i class="fas fa-search"></i></div>
    </div>
  </div>
  <script src="js/script.js"></script>
  <script src="js/suggestions.js"></script>
  <div class="content-space">
    <font size = "3"><center><h1>Members</h1></center></font>
  </div>
  <div class="memgif">
    <center></center>
    </div>
    <center><div class="member">
      <div class="img-info1">
        
        <div class="nameaun"><h2>นายชนกฤต ชนกุลพรรณ<br><br>รหัสนักศึกษา
63010395 Sec 1</h2></div>
      </div>
      <div class="img-info2">
        
        <div class="namearn"><h2>นายชรธรณธร มิตรดี <br><br>รหัสนักศึกษา 63010446
Sec 1</h2></div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>
<div class="img-info3">
  
  <div class="namepee"><h2>นายปธิกร พุ่มปรีดา <br><br>รหัสนักศึกษา 63010559
Sec 1</h2></div>
</div>
</div></center>
<div class="img-info4">
  <h2><center>นักศึกษาระดับปริญญาตรี ชั้นปีที่ 4 <br><br>
  คณะวิศวกรรมศาสตร์ <br><br>
  สาขาวิชาวิศวกรรมโทรคมนาคมและโครงข่าย<br><br>
  สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง</h2></center>
</div>
<div class="endgif">
  <br><br><br><br><br><br>
</div>
</div>
</div>
</div>
</body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมสำหรับหน้าต่าง Members Website (สำหรับ Admin)

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>APA-Control</title>

  <link rel="stylesheet" href="web.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.3.0/css/all.min.css">
</head>
<body>
  <div class="page">
    <div class="sidebar">
      <font size ="5"><center><h1>Group Members</h1></font></center>
      <div class="sidebar-img">
        
      </div>
      <div class="menu">
        <b class="cg">Menu</b>
        <div class="Item">
          <div class="sidebar-icon">
            <i class="fa-solid fa-info"></i>
          </div>
          <a href="Dashboard.php"><font size ="5"><b>Dashboard</b></font></a>
        </div>
        <div class="Item">
          <div class="sidebar-icon">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <i class="fa-solid fa-eject"></i>
    </div>
    <a href="About.php"><font size ="5"><b>About Project</b></font></a>
</div>
<div class="Item">
    <div class="sidebar-icon">
        <i class="fa-solid fa-table"></i>
    </div>
    <a href="History.php"><font size ="5"><b>History</b></font></a>
</div>
<div class="Item Item-active">
    <div class="sidebar-icon">
        <i class="fa-solid fa-people-group"></i>
    </div>
    <font size ="5"><b>Members</b></font>
</div>
<div class="Item">
    <div class="sidebar-icon">
        <i class="fa-solid fa-chart-line"></i>
    </div>
    <a href="Graph.php"><font size ="5"><b>Graphs</b></font></a>
</div>
</div>
<div class="end"></div>
<div class="logout-button">
    <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>
<div class="clock-box">
    <p id="clock"></p>
</div>
<script>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

function updateTime() {
    var now = new Date();
    var hours = now.getHours();
    var minutes = now.getMinutes();
    var seconds = now.getSeconds();

    // Add leading zeros to minutes and seconds
    minutes = (minutes < 10 ? "0" : "") + minutes;
    seconds = (seconds < 10 ? "0" : "") + seconds;

    // Determine AM/PM notation
    var ampm = (hours < 12 ? "AM" : "PM");
    hours = (hours > 12 ? hours - 12 : hours);
    hours = (hours == 0 ? 12 : hours);

    // Format the time
    var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

    // Update the clock element
    document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">
    <br><br><br>
    <br><br><br>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<br><br><br><br><br>
<center></center>
</div>
</div>
<div class="content">
<div class="nav-bar">
<div class="search">
<a href="" target="_blank" hidden></a>
<input type="text" placeholder="Search">
<div class="autocom-box">
<!-- here list are inserted from javascript -->
</div>
<div class="icon"><i class="fas fa-search"></i></div>
</div>
</div>
<script src="js/script.js"></script>
<script src="js/suggestions.js"></script>
<div class="content-space">
<font size ="3"><center><h1>Members</h1></center></font>
</div>
<div class="memgif">
<center></center>
</div>
<center><div class="member">
<div class="img-info1">


```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>APA-Control</title>

  <link rel="stylesheet" href="web.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.3.0/css/all.min.css">

  <style>
    form {
      max-width: 600px;
      margin: 20px auto;
      padding: 15px;
      background-color: #2F4F4F;
      border-radius: 8px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }

    label {
      display: block;
      margin-bottom: 8px;
    }

    select, input {
      width: 100%;
      padding: 10px;
      margin-bottom: 15px;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    box-sizing: border-box;
    border-radius: 8px;
}

input[type="submit"] {
    background-color: #4d89ca;
    color: #fff;
    cursor: pointer;
}

input[type="submit"]:hover {
    background-color: #444b8e;
}

.result {
    margin-top: 20px;
    padding: 15px;
    background-color: #fff;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

.pagination {
    margin-top: 20px;
    text-align: center;
}

.pagination a {
    color: #fff;
    padding: 8px 16px;
    text-decoration: none;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

background-color: #4d89ca;
border-radius: 5px;
margin-right: 5px;
}

.pagination a:hover {
background-color: #444b8e;
}

.pagination .active {
background-color: #444b8e;
}

.current-page{
margin-left : 5px;
margin-right : 8px;
font-size : large;
}
</style>
<script>
function toggleDateFields() {
var filterType = document.getElementById("filter_type").value;
var dateFields = document.getElementById("date_fields");
var dateFromtoEnd = document.getElementById("date_fromtoend");

// ซ่อนทุกอย่างก่อน
dateFields.style.display = "none";
dateFromtoEnd.style.display = "none";

// แสดงเฉพาะที่เลือก
if (filterType === "daily" || filterType === "monthly" || filterType === "yearly") {
dateFields.style.display = "block";
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    if (filterType === "between") {
        dateFields.style.display = "block";
        dateFromtoEnd.style.display = "block";
    }
}

</script>
</head>

<body>
<script src="https://code.jquery.com/jquery-3.5.1.js"></script>
<div class="page">
<div class="sidebar">
<font size = "5"><center><h1>History</h1></font></center>
<div class="sidebar-img">

</div>

<div class="menu">
<b class="cg">Menu</b>
<div class="Item">
<div class="sidebar-icon">
<i class="fa-solid fa-info"></i>
</div>
<a href="Dashboard.php"><font size = "5"><b>Dashboard</b></font></a>
</div>
<div class="Item">
<div class="sidebar-icon">
<i class="fa-solid fa-eject"></i>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>
<a href="About.php"><font size ="5"><b>About Project</b></font></a>
</div>
<div class="Item Item-active">
  <div class="sidebar-icon">
    <i class="fa-solid fa-table"></i>
  </div>
  <font size ="5"><b>History</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-people-group"></i>
  </div>
  <a href="Member.php"><font size ="5"><b>Members</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-chart-line"></i>
  </div>
  <a href="Graph.php"><font size ="5"><b>Graphs</b></font></a>
</div>
</div>
<div class="end"></div>
<div class="logout-button">
  <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>
<div class="clock-box">
  <p id="clock"></p>
</div>
<script>
  function updateTime() {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

var now = new Date();
var hours = now.getHours();
var minutes = now.getMinutes();
var seconds = now.getSeconds();

// Add leading zeros to minutes and seconds
minutes = (minutes < 10 ? "0" : "") + minutes;
seconds = (seconds < 10 ? "0" : "") + seconds;

// Determine AM/PM notation
var ampm = (hours < 12 ? "AM" : "PM");
hours = (hours > 12 ? hours - 12 : hours);
hours = (hours == 0 ? 12 : hours);

// Format the time
var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

// Update the clock element
document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">
    <br><br><br>
    <br><br><br>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<br><br><br><br><br>
<center></center>
</div>
</div>
<div class="content">
<div class="nav-bar">
<div class="search">
<a href="" target="_blank" hidden></a>
<input type="text" placeholder="Search">
<div class="autocom-box">
<!-- here list are inserted from javascript -->
</div>
<div class="icon"><i class="fas fa-search"></i></div>
</div>
</div>
<script src="js/script.js"></script>
<script src="js/suggestions.js"></script>
<div class="content-space">
<div class="wordtable"><center><h1>History</h1></center></div>
<form method="post" action="History.php" oninput="toggleDateFields()">
<label for="filter_type">Filter Data:</label>
<select name="filter_type" id="filter_type">
<option value="">-- Type Select --</option>
<option value="daily">Daily</option>
<option value="monthly">Monthly</option>
<option value="yearly">Yearly</option>
<option value="between">Between</option>
</select>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<div id="date_fields" style="display: none;">
  <label for="from_day">From Day:</label>
  <select name="from_day">
    <option value="">-- Select --</option>
    <?php for ($day = 1; $day <= 31; $day++) : ?>
      <option value="<?php echo sprintf("%02d", $day); ?>"><?php
echo sprintf("%02d", $day); ?></option>
    <?php endfor; ?>
  </select>

  <label for="from_month">From Month:</label>
  <select name="from_month">
    <option value="">-- Select --</option>
    <?php for ($month = 1; $month <= 12; $month++) : ?>
      <option value="<?php echo sprintf("%02d", $month); ?>"><?php
echo sprintf("%02d", $month); ?></option>
    <?php endfor; ?>
  </select>

  <label for="from_year">From Year:</label>
  <select name="from_year">
    <option value="">-- Select --</option>
    <?php
    $currentYear = date('Y');
    for ($year = $currentYear; $year >= 1900; $year--) :
    ?>
      <option value="<?php echo $year; ?>"><?php echo $year;
?></option>
    <?php endfor; ?>
  </select>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>

<div id="date_fromtoend" style="display: none;">
  <label for="to_day">To Day:</label>
  <select name="to_day">
    <option value="">-- Select --</option>
    <?php for ($day = 1; $day <= 31; $day++) : ?>
      <option value="<?php echo sprintf("%02d", $day); ?>"><?php
echo sprintf("%02d", $day); ?></option>
    <?php endfor; ?>
  </select>
  <label for="to_month">To Month:</label>
  <select name="to_month">
    <option value="">-- Select --</option>
    <?php for ($month = 01; $month <= 12; $month++) : ?>
      <option value="<?php echo sprintf("%02d", $month); ?>"><?php
echo sprintf("%02d", $month); ?></option>
    <?php endfor; ?>
  </select>
  <label for="to_year">To Year:</label>
  <select name="to_year">
    <option value="">-- Select --</option>
    <?php
$currentYear = date('Y');
for ($year = $currentYear; $year >= 1900; $year--) :
  ?>
    <option value="<?php echo $year; ?>"><?php echo $year;
?></option>
  <?php endfor; ?>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        </select>
    </div>
    <input type="submit" value="Show Data">
</form>

<?php
    // ตรวจสอบว่ามีค่าส่งผ่าน URL parameters หรือไม่และดึงค่าเข้ามาใช้งาน
    if (isset($_GET["filter_type"]) && isset($_GET["from_day"]) &&
isset($_GET["from_month"]) && isset($_GET["from_year"]) && isset($_GET["to_day"]) &&
isset($_GET["to_month"]) && isset($_GET["to_year"])) {
        $filterType = $_GET["filter_type"];
        $fromDay = $_GET["from_day"];
        $fromMonth = $_GET["from_month"];
        $fromYear = $_GET["from_year"];
        $fromDateFilter = "$fromYear-$fromMonth-$fromDay";
        $toDay = $_GET["to_day"];
        $toMonth = $_GET["to_month"];
        $toYear = $_GET["to_year"];
        $toDateFilter = "$toYear-$toMonth-$toDay";

        // เพิ่มโค้ดการดึงข้อมูลจากฐานข้อมูลตามความเหมาะสม
        $servername = "roundhouse.proxy.rlwy.net:32692";
        $username = "root";
        $password = "cbA5b511H2334bFDF-3a4ACEHA2GcBD2";
        $dbname = "project";

        // สร้างการเชื่อมต่อ
        $conn = new mysqli($servername, $username, $password, $dbname);

        // ตรวจสอบการเชื่อมต่อ
        if ($conn->connect_error) {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        die("การเชื่อมต่อล้มเหลว: " . $conn->connect_error);
    }
    // กำหนดจำนวนข้อมูลต่อหน้า
    $per_page = 20;

    // หาหน้าปัจจุบัน
    if (isset($_GET['page'])) {
        $page = $_GET['page'];
    } else {
        $page = 1;
    }

    $start_from = ($page - 1) * $per_page;

    // สร้างคำสั่ง SQL เพื่อดึงข้อมูล
    $sql = "SELECT * FROM plclogo8 WHERE ";
    $sql2 = "SELECT COUNT(*) AS total FROM plclogo8 WHERE ";

    if ($filterType === "daily") {
        // ตัวอย่างเงื่อนไขรายวัน
        $sql .= "Date(Datetime) = '$fromDateFilter'";
        $sql2 .= "Date(Datetime) = '$fromDateFilter'";
    } elseif ($filterType === "monthly") {
        // ตัวอย่างเงื่อนไขรายเดือน
        $sql .= "MONTH(Datetime) = '$fromMonth' AND YEAR(Datetime) = '$fromYear'";
        $sql2 .= "MONTH(Datetime) = '$fromMonth' AND YEAR(Datetime) = '$fromYear'";
    } elseif ($filterType === "yearly") {
        // ตัวอย่างเงื่อนไขรายปี
        $sql .= "YEAR(Datetime) = '$fromYear'";
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    $sql2 .= "YEAR(Datetime) = '$fromYear'";
} elseif ($filterType === "between") {
    // ตัวอย่างเงื่อนไขระหว่างวันที่
    $sql .= "Datetime BETWEEN '$fromDateFilter' AND '$toDateFilter'";
    $sql2 .= "Datetime BETWEEN '$fromDateFilter' AND '$toDateFilter'";
}

$sql .= "LIMIT $start_from, $per_page";
// ดำเนินการต่อเพื่อดึงข้อมูลจากฐานข้อมูล
$result = $conn->query($sql);
// แสดงผล
if ($result->num_rows > 0) {
    echo
"<table><tr><th>DATA</th><th>INFOSOIL</th><th>INFOAIR</th><th>STATUSLIGHT</th><th>STATUSFAN</th><th>STATUSPUMP</th><th>DATETIME</th></tr>";
    while ($row = $result->fetch_assoc()) {
        echo "<tr><td>" . $row["Data"] . "</td><td>" . $row["Infooil"] .
"</td><td>" . $row["Infoair"] . "</td><td>" . $row["Statuslight"] . "</td><td>" .
$row["Statusfan"] . "</td><td>" . $row["Statuspump"] . "</td><td>" . $row["Datetime"] .
"</td></tr>";
    }
    echo "</table>";
}

// สร้างลิงก์เปลี่ยนหน้า
$result = $conn->query($sql2);
$row = $result->fetch_assoc();
$total_pages = ceil($row["total"] / $per_page);

echo "<br><br>";
echo "<div class='pagination'>";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if ($page > 1) {
    echo "<a href='?page=" . ($page - 1) .
"&filter_type=$filterType&from_day=$fromDay&from_month=$fromMonth&from_year
=$fromYear&to_day=$toDay&to_month=$toMonth&to_year=$toYear'>Previous</a>";
}

for ($i = 1; $i <= $total_pages; $i++) {
    // ตรวจสอบหน้าปัจจุบัน
    if ($i == $page) {
        echo "<span class='current-page'>$page</span>";
    }
}

if ($page < $total_pages) {
    echo "<a href='?page=" . ($page + 1) .
"&filter_type=$filterType&from_day=$fromDay&from_month=$fromMonth&from_year
=$fromYear&to_day=$toDay&to_month=$toMonth&to_year=$toYear'>Next</a>";
}

echo "</div>";
} elseif ($_SERVER["REQUEST_METHOD"] == "POST" &&
isset($_POST["filter_type"])) {
    $filterType = $_POST["filter_type"];
    $fromDay = $_POST["from_day"];
    $fromMonth = $_POST["from_month"];
    $fromYear = $_POST["from_year"];
    $fromDateFilter = "$fromYear-$fromMonth-$fromDay";
    $toDay = $_POST["to_day"];
    $toMonth = $_POST["to_month"];
    $toYear = $_POST["to_year"];
    $toDateFilter = "$toYear-$toMonth-$toDay";
    // เพิ่มโค้ดการดึงข้อมูลจากฐานข้อมูลตามความเหมาะสม

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

$servername = "roundhouse.proxy.rlwy.net:32692";
$username = "root";
$password = "cbA5b511H2334bFDF-3a4ACEHA2GcBD2";
$dbname = "project";

// สร้างการเชื่อมต่อ
$conn = new mysqli($servername, $username, $password, $dbname);

// ตรวจสอบการเชื่อมต่อ
if ($conn->connect_error) {
    die("การเชื่อมต่อล้มเหลว: " . $conn->connect_error);
}
// กำหนดจำนวนข้อมูลต่อหน้า
$per_page = 20;
// หาหน้าปัจจุบัน
if (isset($_GET['page'])) {
    $page = $_GET['page'];
} else {
    $page = 1;
}

$start_from = ($page - 1) * $per_page;

// สร้างคำสั่ง SQL เพื่อดึงข้อมูล
$sql = "SELECT * FROM plclogo8 WHERE ";
$sql2 = "SELECT COUNT(*) AS total FROM plclogo8 WHERE ";

if ($filterType === "daily") {
    // ตัวอย่างเงื่อนไขรายวัน
    $sql .= "Date(Datetime) = '$fromDateFilter'";
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    $sql2 .= "Date(Datetime) = '$fromDateFilter'";
} elseif ($filterType === "monthly") {
    // ตัวอย่างเงื่อนไขรายเดือน
    $sql .= "MONTH(Datetime) = '$fromMonth' AND YEAR(Datetime) =
'$fromYear'";
    $sql2 .= "MONTH(Datetime) = '$fromMonth' AND YEAR(Datetime) =
'$fromYear'";
} elseif ($filterType === "yearly") {
    // ตัวอย่างเงื่อนไขรายปี
    $sql .= "YEAR(Datetime) = '$fromYear'";
    $sql2 .= "YEAR(Datetime) = '$fromYear'";
} elseif ($filterType === "between") {
    // ตัวอย่างเงื่อนไขระหว่างวันที่
    $sql .= "Datetime BETWEEN '$fromDateFilter' AND '$toDateFilter'";
    $sql2 .= "Datetime BETWEEN '$fromDateFilter' AND '$toDateFilter'";
}

$sql .= "LIMIT $start_from, $per_page";
// ดำเนินการต่อเพื่อดึงข้อมูลจากฐานข้อมูล
$result = $conn->query($sql);
// แสดงผล
if ($result->num_rows > 0) {
    echo
" <table><tr><th>DATA</th><th>INFOSOIL</th><th>INFOAIR</th><th>STATUSLIGHT</
th><th>STATUSFAN</th><th>STATUSPUMP</th><th>DATETIME</th></tr>";
    while ($row = $result->fetch_assoc()) {
        echo "<tr><td>" . $row["Data"] . "</td><td>" . $row["Infooil"] .
"</td><td>" . $row["Infoair"] . "</td><td>" . $row["Statuslight"] . "</td><td>" .
$row["Statusfan"] . "</td><td>" . $row["Statuspump"] . "</td><td>" . $row["Datetime"] .
"</td></tr>";
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        echo "</table>";
    }

    // สร้างลิงก์เปลี่ยนหน้า
    $result = $conn->query($sql2);
    $row = $result->fetch_assoc();
    $total_pages = ceil($row["total"] / $per_page);

    echo "<br><br>";
    echo "<div class='pagination'>";
    if ($page > 1) {
        echo "<a href='?page=" . ($page - 1) .
"&filter_type=$filterType&from_day=$fromDay&from_month=$fromMonth&from_year
=$fromYear&to_day=$toDay&to_month=$toMonth&to_year=$toYear'>Previous</a>";
    }

    for ($i = 1; $i <= $total_pages; $i++) {
        // ตรวจสอบหน้าปัจจุบัน
        if ($i == $page) {
            echo "<span class='current-page'>$page</span>";
        }
    }

    if ($page < $total_pages) {
        echo "<a href='?page=" . ($page + 1) .
"&filter_type=$filterType&from_day=$fromDay&from_month=$fromMonth&from_year
=$fromYear&to_day=$toDay&to_month=$toMonth&to_year=$toYear'>Next</a>";
    }

    echo "</div>";
} else {
    // กรณีที่ไม่มีการส่งค่าผ่าน URL parameters หรือ POST parameters

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// ให้กำหนดค่าเริ่มต้น
$filterType = "";
$fromDay = "";
$fromMonth = "";
$fromYear = "";
$fromDateFilter = "$fromYear-$fromMonth-$fromDay";
$toDay = "";
$toMonth = "";
$toYear = "";
$toDateFilter = "$toYear-$toMonth-$toDay";
}
?>
</div>
</div>
</div>
</body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ช

โปรแกรมสำหรับหน้าต่าง Graph Website (สำหรับ Admin)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>APA-Control</title>

  <link rel="stylesheet" href="web.css">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.3.0/css/all.min.css">
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

<style>
  form {
    max-width: 600px;
    margin: 20px auto;
    padding: 15px;
    background-color: #2F4F4F;
    border-radius: 8px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  }

  label {
    display: block;
    margin-bottom: 8px;
  }

  select, input {
    width: 100%;
    padding: 10px;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

margin-bottom: 15px;
box-sizing: border-box;
border-radius: 8px;
}

input[type="submit"] {
background-color: #4d89ca;
color: #fff;
cursor: pointer;
}

input[type="submit"]:hover {
background-color:#444b8e;
}

.result {
margin-top: 20px;
padding: 15px;
background-color: #fff;
border-radius: 8px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
</style>
<script>
function toggleDateFields() {
var filterType = document.getElementById("filter_type").value;
var dateFields = document.getElementById("date_fields");
var dateFromtoEnd = document.getElementById("date_fromtoend");

// ซ่อนทุกอย่างก่อน
dateFields.style.display = "none";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

dateFromtoEnd.style.display = "none";

// แสดงเฉพาะที่เลือก
if (filterType === "daily" || filterType === "monthly" || filterType === "yearly") {
    dateFields.style.display = "block";
}
if (filterType === "between") {
    dateFields.style.display = "block";
    dateFromtoEnd.style.display = "block";
}
}
</script>
</head>
<body>
<div class="page">
<div class="sidebar">
<font size = "5"><center><h1>Graph Status</h1></font></center>
<div class="sidebar-img">

</div>

<div class="menu">
<b class="cg">Menu</b>
<div class="Item">
<div class="sidebar-icon">
<i class="fa-solid fa-info"></i>
</div>
<a href="Dashboard.php"><font size = "5"><b>Dashboard</b></font></a>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</div>
<div class="Item" >
  <div class="sidebar-icon">
    <i class="fa-solid fa-eject"></i>
  </div>
  <a href="About.php"><font size ="5"><b>About Project</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-table"></i>
  </div>
  <a href="History.php"><font size ="5"><b>History</b></font></a>
</div>
<div class="Item">
  <div class="sidebar-icon">
    <i class="fa-solid fa-people-group"></i>
  </div>
  <a href="Member.php"><font size ="5"><b>Members</b></font></a>
</div>
<div class="Item Item-active">
  <div class="sidebar-icon">
    <i class="fa-solid fa-chart-line"></i>
  </div>
  <font size ="5"><b>Graphs</b></font>
</div>
</div>
<div class="end"></div>
<div class="logout-button">
  <a href="Loginmail.php"><font size ="6"><b>Logout</b></font></a>
</div>
<div class="clock-box">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<p id="clock"></p>
</div>
<script>
function updateTime() {
    var now = new Date();
    var hours = now.getHours();
    var minutes = now.getMinutes();
    var seconds = now.getSeconds();

    // Add leading zeros to minutes and seconds
    minutes = (minutes < 10 ? "0" : "") + minutes;
    seconds = (seconds < 10 ? "0" : "") + seconds;

    // Determine AM/PM notation
    var ampm = (hours < 12 ? "AM" : "PM");
    hours = (hours > 12 ? hours - 12 : hours);
    hours = (hours == 0 ? 12 : hours);

    // Format the time
    var timeString = hours + ":" + minutes + ":" + seconds + " " + ampm;

    // Update the clock element
    document.getElementById("clock").innerHTML = timeString;
}

// Call the updateTime function every second
setInterval(updateTime, 1000);
</script>
<div class="other">

```

```

<br><br><br>
<br><br><br>
<br><br><br><br><br>
<center></center>
</div>
</div>
<div class="content">
<div class="nav-bar">
<div class="search">
<a href="" target="_blank" hidden></a>
<input type="text" placeholder="Search">
<div class="autocom-box">
<!-- here list are inserted from javascript -->
</div>
<div class="icon"><i class="fas fa-search"></i></div>
</div>
</div>
<script src="js/script.js"></script>
<script src="js/suggestions.js"></script>
<div style="width:70%;hieght:10%;text-align:center">
<div class="page-header"><h2>Graph</h2></div>
<form method="post" action="" oninput="toggleDateFields()">
<label for="filter_type">Filter Data:</label>
<select name="filter_type" id="filter_type">
<option value="">-- Type Select --</option>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<option value="daily">Daily</option>
<option value="monthly">Monthly</option>
<option value="yearly">Yearly</option>
<option value="between">Between</option>
</select>

```

```

<div id="date_fields" style="display: none;">
  <label for="from_day">From Day:</label>
  <select name="from_day">
    <option value="">-- Select --</option>
    <?php for ($day = 1; $day <= 31; $day++) : ?>
      <option value="<?php echo sprintf("%02d", $day); ?>"><?php echo
sprintf("%02d", $day); ?></option>
    <?php endfor; ?>
  </select>
  <label for="from_month">From Month:</label>
  <select name="from_month">
    <option value="">-- Select --</option>
    <?php for ($month = 1; $month <= 12; $month++) : ?>
      <option value="<?php echo sprintf("%02d", $month); ?>"><?php
echo sprintf("%02d", $month); ?></option>
    <?php endfor; ?>
  </select>

```

```

<label for="from_year">From Year:</label>
<select name="from_year">
  <option value="">-- Select --</option>
  <?php
$currentYear = date('Y');
for ($year = $currentYear; $year >= 1900; $year--) :

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

?>
    <option value="<?php echo $year; ?>"><?php echo $year;
?></option>
    <?php endforeach; ?>
</select>
</div>

<div id="date_fromtoend" style="display: none;">
    <label for="to_day">To Day:</label>
    <select name="to_day">
        <option value="">-- Select --</option>
        <?php for ($day = 1; $day <= 31; $day++) : ?>
            <option value="<?php echo sprintf("%02d", $day);?>"><?php echo
sprintf("%02d", $day); ?></option>
        <?php endforeach; ?>
    </select>

    <label for="to_month">To Month:</label>
    <select name="to_month">
        <option value="">-- Select --</option>
        <?php for ($month = 1; $month <= 12; $month++) : ?>
            <option value="<?php echo sprintf("%02d", $month); ?>"><?php
echo sprintf("%02d", $month); ?></option>
        <?php endforeach; ?>
    </select>

    <label for="to_year">To Year:</label>
    <select name="to_year">
        <option value="">-- Select --</option>
        <?php
        $currentYear = date('Y');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for ($year = $currentYear; $year >= 1900; $year--) :
?>
    <option value="<?php echo $year; ?>"><?php echo $year;
?></option>
    <?php endforeach; ?>
</select>
</div>

```

```

<input type="submit" value="Show Data">
</form>

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $fromDay = $_POST["from_day"];
    $fromMonth = $_POST["from_month"];
    $fromYear = $_POST["from_year"];
    $fromDateFilter = "$fromYear-$fromMonth-$fromDay";

    $toDay = $_POST["to_day"];
    $toMonth = $_POST["to_month"];
    $toYear = $_POST["to_year"];
    $toDateFilter = "$toYear-$toMonth-$toDay";

    $filterType = $_POST["filter_type"];

    // เพิ่มโค้ดการดึงข้อมูลจากฐานข้อมูลตามความเหมาะสม
    $servername = "roundhouse.proxy.rlwy.net:32692";
    $username = "root";
    $password = "cbA5b511H2334bFDF-3a4ACEHA2GcBD2";
    $dbname = "project";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// สร้างการเชื่อมต่อ
$conn = new mysqli($servername, $username, $password, $dbname);

// ตรวจสอบการเชื่อมต่อ
if ($conn->connect_error) {
    die("การเชื่อมต่อล้มเหลว: " . $conn->connect_error);
}
else {
    if ($filterType == 'daily') {
        $sql1 = "SELECT * FROM plclogo8 WHERE DATE(Datetime) =
'$fromDateFilter'";
    } elseif ($filterType == 'monthly') {
        $sql1 = "SELECT * FROM plclogo8 WHERE MONTH(Datetime) =
'$fromMonth' AND YEAR(Datetime) = '$fromYear'";
    } elseif ($filterType == 'yearly') {
        $sql1 = "SELECT * FROM plclogo8 WHERE YEAR(Datetime) =
'$fromYear'";
    } elseif ($filterType === "between") {
        // ตัวอย่างเงื่อนไขรายปี
        $sql1 = "SELECT * FROM plclogo8 WHERE Datetime BETWEEN
'$fromDateFilter' AND '$toDateFilter'";
    }
}

$result1 = mysqli_query($conn, $sql1);

while ($row = mysqli_fetch_array($result1)) {
    // Your existing code to fetch data
    $datetime[] = date_format(date_create($row['Datetime']), "d-m-Y
H:i:s");

    $tempsoil[] = $row['Tempsoil'];
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    $humidsoil[] = $row['Humidsoil'];
    $tempair[] = $row['Tempair'];
    $humidair[] = $row['Humidair'];
    $light[] = $row['Light'];
    $statuslight[] = $row['Bitstatuslight'];
    $statusfan[] = $row['Bitstatusfan'];
    $statuspump[] = $row['Bitstatuspump'];
}
// Close the database connection
$conn->close();
}
}
?>
<div class="word_device"><h3>Soil Sensor</h3></div>
<canvas id="chart1"></canvas>
</div>
<script src="//code.jquery.com/jquery-1.9.1.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/4.2.1/chart.min.js"></script>
<script type="text/javascript">
    var ctx1 = document.getElementById("chart1").getContext('2d');
    var Chart1 = new Chart(ctx1, {
        type: 'line',
        data: {
            labels:<?php echo json_encode($datetime); ?>,
            datasets: [{
                label:'Temperature Soil (Celcius)',
                backgroundColor:[
                    'RGB(0, 0, 255,0.1)',borderColor:["RGB(255, 99,
71)"],borderWidth:3,tension:0.4,
                pointBorderColor: ["#66CCFF"],

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

data:<?php echo json_encode($tempsoil); ?>,
},
{
label:'Humidity Soil (RH%)',
backgroundColor:[
'RGB(0, 0,
255,0.1)'],borderColor:["#00ff00"],borderWidth:3,tension:0.4,
pointBorderColor: ["#66CCFF"],
data:<?php echo json_encode($humidsoil); ?>,
}
],
},
options: {
legend: {
display: true,
position: 'top',
labels: {
fontColor: '#FFFFFF',
fontFamily: 'Circular Std Book',
fontSize: 14,
}
},
},
}
});
</script>
<div style="width:70%;hieght:10%;text-align:center">
<div class="word_device"><br><br><h3>Air Sensor</h3></div>
<canvas id="chart2"></canvas>
</div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<script src="//code.jquery.com/jquery-1.9.1.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/4.2.1/chart.min.js"></script>
<script type="text/javascript">
    var ctx2 = document.getElementById("chart2").getContext('2d');
    var Chart2 = new Chart(ctx2, {
        type: 'line',
        data: {
            labels:<?php echo json_encode($datetime); ?>,
            datasets: [{
                label:'Temperature Air (Celcius)',
                backgroundColor:[
                    'RGB(0, 0, 255,0.1)',borderColor:["RGB(255, 99,
71)"],borderWidth:3,tension:0.4,
                pointBorderColor: ["#66CCFF"],
                data:<?php echo json_encode($tempair); ?>,
            },
            {
                label:'Humidity Air (RH%)',
                backgroundColor:[
                    'RGB(0, 0,
255,0.1)',borderColor:["#00ff00"],borderWidth:3,tension:0.4,
                pointBorderColor: ["#66CCFF"],
                data:<?php echo json_encode($humidair); ?>,
            },
            {
                label:'Light (lux)',
                backgroundColor:[
                    'RGB(0, 0,
255,0.1)',borderColor:["#CC66FF"],borderWidth:3,tension:0.4,
                pointBorderColor: ["#66CCFF"],

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        data:<?php echo json_encode($light); ?>,
    }
]
},
options: {
    legend: {
        display: true,
        position: 'top',

        labels: {
            fontColor: '#FFFFFF',
            fontFamily: 'Circular Std Book',
            fontSize: 14,
        }
    },
});
</script>
<div style="width:70%;hieght:10%;text-align:center">
    <div class="word_device"><br><br><h3>Status Device</h3></div>
    <canvas id="chart3"></canvas>
</div>
<script src="//code.jquery.com/jquery-1.9.1.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/4.2.1/chart.min.js"></script>
<script type="text/javascript">
    var ctx3 = document.getElementById("chart3").getContext('2d');
    var Chart3 = new Chart(ctx3, {
        type: 'line',
        data: {
            labels:<?php echo json_encode($datetime); ?>,

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datasets: [{
  label:'Status Light (1:ON , 0:OFF)',
  backgroundColor:[
    'RGB(0, 0, 255,0.1)],borderColor:["RGB(255, 99,
71)"],borderWidth:3,tension:0.4,
  pointBorderColor: ["#66CCFF"],
  data:<?php echo json_encode($statuslight); ?>,
},
{
  label:'Status Fan (1:ON , 0:OFF)',
  backgroundColor:[
    'RGB(0, 0,
255,0.1)],borderColor:["#00ff00"],borderWidth:3,tension:0.4,
  pointBorderColor: ["#66CCFF"],
  data:<?php echo json_encode($statusfan); ?>,
},
{
  label:'Status Pump (1:ON , 0:OFF)',
  backgroundColor:[
    'RGB(0, 0,
255,0.1)],borderColor:["#CC66FF"],borderWidth:3,tension:0.4,
  pointBorderColor: ["#66CCFF"],
  data:<?php echo json_encode($statuspump); ?>,
}]
},
options: {
  legend: {
    display: true,
    position: 'top',

  labels: {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        fontColor: '#FFFFFF',
        fontFamily: 'Circular Std Book',
        fontSize: 14,
    }
},
}
});
</script>
</div>
</div>
</body>
</html>

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

@import
url('https://fonts.googleapis.com/css2?family=Prompt:wght@200&display=swap');
*{
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: 'Prompt', sans-serif;
}
.page {
width: auto;
height: auto;
background-color: #292931;
display: flex;
color: white;
}
.sidebar {
width: 20%;
font-size: smaller;
padding: 40px;
}
.sidebar-img{
width: 100px;
height: 98px;
border-radius: 50%;
overflow: hidden;
margin-top: 30px;
margin-left: 35%;
}
.sidebar-img img{
width: 100px;
height: 100px;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    object-fit: cover;
}
.cg{
    color:#808191;
    font-size:large;
}
.content{
    width: 80%;
    background-color: #1f1d2b;
    padding: 20px;
}
.menu{
    margin-top: 30px;
}
.item{
    margin: 40px 0;
    display: flex;
    align-items: center;
    color:#808191
}
.sidebar-icon{
    margin-right: 40px;
    width: 60px;
    height: 60px;
    background: #353340;
    color: #808191;
    display: flex;
    justify-content: center;
    align-items: center;
    border-radius: 10px;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.Item-active{
    color: white;
}
.Item-active .sidebar-icon{
    background-color:cornflowerblue;
    color: white;
}
.end{
    width: 100%;
    height: 1.5px;
    background-color: #393846;
}
.end2{
    width: 100%;
    height: 3px;
    margin-top: 8%;
    background-color: #4c4b5c;
}
.end3{
    width: 100%;
    height: 3px;
    margin-top: 4%;
    background-color: #4c4b5c;
}
.end4{
    width: 100%;
    height: 2px;
    margin-top: 10px;
    background-color: #4c4b5c;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.nav-bar{
  width: 100%;
}
.search{
  width: 50%;
  height: 40px;
  background-color: #c0bdd1;
  border-radius: 10px;
}
.search input{
  width: 100%;
  height: 100%;
  font-size: 16px;
  background-color: transparent;
  border: none;
  outline: none;
  color: rgb(0, 0, 0);
  padding: 20px;
}
.search .autocom-box{
  padding: 0;
  opacity: 0;
  pointer-events: none;
  max-height: 280px;
  overflow-y: auto;
}
.search.active .autocom-box{
  padding: 10px 8px;
  opacity: 1;
  pointer-events: auto;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.autocom-box li{
  list-style: none;
  display: none;
  width: 100%;
  cursor: default;
  border-radius: 2px;
}
.search.active .autocom-box li{
  display: block;
}
.autocom-box li:hover{
  background: #0078da;
}
.search .icon{
  position: absolute;
  right: 700px;
  top: 30px;
  height: 100px;
  width: 100px;
  color: #382900;
  cursor: pointer;
}
.content-space{
  margin-top: 60px;
}
.image{
  margin-top: 20px;
  width: 100%;
  height: 250px;
  display: flex;
  justify-content: space-between;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    align-items: center;
}
.image-item{
    height: 100%;
    border-radius: 10px;
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
}
.image-one{
    width: 100%;
    height: 130%;
    margin-left: 30px;
    margin-top: 87px;
    background-image:
url(https://static.wixstatic.com/media/788668_19bfa48877f7416ab5db2482b12fee55~
mv2.gif);
}
.video{
    width: 100%;
    height: 130%;
    padding-top: 45px;
    margin-left: 45px;
}
.wordinformation{
    margin-top: 30px;
}
.information{
    display: flex;
}
.container {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

margin-left: 225px;
margin-top: 50px;
width: 400px;
height: 470px;
border: 1px solid #ccc;
background-color: #302d42;
overflow: hidden;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
border-radius: 10px;
}
.header {
position: relative;
background-color: #1f1d2b;
color: #ffffff;
text-align: center;
padding: 20px;
}
.header img {
max-width: 100%;
height: 200px;
width: 300px;
display: block;
margin: 0 auto;
border-radius: 5%;
}
.contentinformation {
display: flex;
justify-content: space-between;
padding: 20px;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.left-content,
.right-content {
  flex: 1;
}
.wordstatus{
  margin-top: 60px;
}
.status{
  display: flex;
}
.framestatus{
  flex-direction: column;
  align-items: center;
  margin-top: 50px;
  display: flex;
  margin-left: 168px;
}
.statusdevice{
  text-align: center;
  padding: 20px;
  border: 2px solid #ccc;
  border-radius: 10px;
}
.commandstatus{
  display: flex;
  text-align: left;
  margin-top: 30px;
}
.readstatus{
  display: flex;
  text-align: left;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

margin-top: 30px;
}
.wordcommand{
margin-top: 30px;
margin-left: 5px;
}
.wordread{
margin-top: 30px;
}
.rounded-box {
width: 80px;
height: 80px;
background-color: #302d42; /* สีพื้นหลัง */
border-radius: 50%; /* ทำให้เป็นวงกลม */
display: flex;
align-items: center;
justify-content: center;
text-align: center;
margin-left: 25px;
}
/* เพิ่มขนาดและระยะห่างสำหรับรูปภาพ */
.statusdevice img {
max-width: 150px;
border-radius: 50%;
margin-bottom: 10px;
}
.content-table{
width: 100%;
padding-top: 20px;
}
.tableinfo{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

margin-top: 20px;
}
.wordtable{
margin-top: 30px;
font-size: larger;
}
table {
width: 90%;
border-collapse: collapse;
margin-top: 40px;
color: #fff; /* เปลี่ยนสีข้อความในตารางเป็นขาว */
margin-left : 75px;
border-radius: 10px;
overflow: hidden;
}
th, td {
padding: 12px 15px;
text-align: center;
border: 1px solid #565656;
}
th {
background-color: #5e93ca;
color: #fff;
}
tr:nth-child(even) {
background-color: #405f5f; /* สีพื้นหลังของแถวคู่ */
}
tr:hover {
background-color: #708090; /* สีพื้นหลังเมื่อโฮเวอร์เมาส์ */
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

a {
  padding : 8px;
  margin-right: 8px;
  text-decoration: none;
  color: #8fb9e6;
  border-radius: 3px;
}
a:hover {
  background-color: #dc3545;
  color: white;
}
.member{
  display: flex;
  justify-content: center;
}
.img-info1{
  margin-right: 150px;
}
.img-info3{
  margin-left: 150px;
}
.nameaun{
  margin-top: 20px;
}
.nameearn{
  margin-top: 20px;
}
.namepee{
  margin-top: 20px;
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.img-info4{
  margin-left: 400px;;
  padding-top: 120px;
  display: flex;
}
.info{
  padding-top: 40px;
  margin-left: 80px;
  margin-right: 80px;
  font-size:small;
}
.block-diagram{
  padding-top: 80px;
  margin-left: 150px;
  display: flex;
}
.kmitl{
  padding-top: 80px;
}
.page-header{
  padding-top: 10px;
  color:#ffffff;
}
.word_device{
  color: #ffffff;
  margin-top: 10px;
  font-size: large;
}
.memgif{
  padding-top: 30px;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.other{
    margin-top: 50px;
}
.clock-box {
    display: inline-block;
    margin-top: 40px;
    background-color: #000000;
    border: 5px solid #ffffff;
    border-radius: 10px;
    font-size: 45px;
    margin-left: 20px;
    color: red;
}
.logout-button {
    margin-top: 20px;
    margin-left : 85px
}
@media only screen and (min-width:360px) and (max-width:760px){
    .page{
        flex-direction: column;
        width: 230vh;
        height: auto;
    }
    .content{
        width: 100%;
        height: auto;
    }
    .sidebar-img{
        display: none;
    }
}
.end{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        width: 180vh;
    }
    .other{
        width: 100%;
        height: 20%;
    }
    .video{
        width: 100%;
        height: 130%;
        padding-top: 40px;
    }
    .member{
        margin-left: 40px;
    }
    .img-info4{
        padding-left: 30px;
        padding-top: 80px;
    }
    .search .icon{
        display: none;
    }
    .search{
        display: none;
    }
    .totaltemp{
        margin-left: 8%;
    }
    .lightvibra{
        margin-left: 8%;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

@media only screen and (min-width:768px) and (max-width:1280px){
  .page{
    flex-direction: column;
    width: 200vh;
    height: auto;
  }
  .content{
    width: 200vh;
  }
  .sidebar{
    width: auto;
  }
  .sidebar-img{
    display: none;
  }
  .menu{
    margin-left: 42%;
  }
  .other{
    width: 40%;
    height: 20%;
    margin-left: 30%;
  }
  .clock-box{
    margin-left: 43%;
  }
  .video{
    width: 100%;
    height: 100%;
    margin-top: 10px;
  }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
.image-one{
    width: 100%;
    height: 100%;
}
.content{
    width: auto;
    height: auto;
}
.totaltemp{
    margin-left: 13%;
}
.lightvibra{
    margin-left: 13%;
}
.img-info4{
    padding-top: 100px;
    padding-left: 17%;
}
.sidebar-img{
    margin-left: 710px;
}
.search .icon{
    display: none;
}
.search{
    display: none;
}
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้