



รายงานผลโครงการและงานวิจัย

เรื่อง

เมืองอัจฉริยะเอลิเซียม (Elysium)

นายพชร โสรส

นายวศิน เสริมสุข

โครงการนี้เป็นส่วนหนึ่งของวิชาโครงการ รหัส 30128-8501

หลักสูตร ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)

สาขาวิชาเทคโนโลยีคอมพิวเตอร์

ภาคเรียนที่ 2 ปีการศึกษา 2566

วิทยาลัยเทคนิคจันทบุรี สถาบันการอาชีวศึกษาภาคตะวันออก

ใบรับรองอนุปริณยานิพนธ์
สาขาเทคโนโลยีคอมพิวเตอร์ วิทยาลัยเทคนิคจันทบุรี

หัวข้อวิจัย : เมืองอัจฉริยะเอลิเซียม
Elysium
ผู้ดำเนินการวิจัย : นายพชร โสรส
นายวศิน เสริมสุข
ที่ปรึกษา : นายฉัตรพฤกษ์ สีม
นางสาวประภาพร บันเทา
สาขาวิชา : เทคโนโลยีคอมพิวเตอร์
สาขางาน : คอมพิวเตอร์ระบบเครือข่าย
ปี พ.ศ. : 2566

สาขาวิชาเทคโนโลยีคอมพิวเตอร์ ให้นับอนุปริณยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรประกาศนียบัตรวิชาชีพชั้นสูง สาขาวิชาเทคโนโลยีคอมพิวเตอร์

.....หัวหน้าสาขาวิชาเทคโนโลยีคอมพิวเตอร์
(นายศัตรารุธ ศรีชาติ)

คณะกรรมการสอบโครงการ

.....ประธานกรรมการ
(นายฉัตรพฤกษ์ สีม)

.....กรรมการ
(นายปิยพงษ์ เกตุวงา)

.....กรรมการ
(นางสาวประภาพร บันเทา)

หัวข้อวิจัย	ระบบเมืองอัจฉริยะเอลิเซียม
ผู้ดำเนินการวิจัย	นายพชร โสรส
	นายวศิน เสริมสุข
ที่ปรึกษา	นายฉัตรพฤกษ์ สีสิม
	นางสาวประภาพร บันเทา
สาขาวิชา	เทคโนโลยีคอมพิวเตอร์
สาขางาน	คอมพิวเตอร์ระบบเครือข่าย
ปีการศึกษา	2566

บทคัดย่อ

การทำโครงการระบบเมืองอัจฉริยะเอลิเซียมครั้งนี้มีวัตถุประสงค์เพื่อ 1. เพื่อสร้างระบบเมืองอัจฉริยะเอลิเซียม 2. เพื่อศึกษาหลักการของเซนเซอร์ตรวจจับแรงสั่นสะเทือนและเซนเซอร์ RFID 3. เพื่อประเมินประสิทธิภาพของเมืองอัจฉริยะเอลิเซียม

จากการดำเนินโครงการและทดสอบโครงการ โดยการทดลองการใช้งานและได้ทำการประเมินประสิทธิภาพของเมืองอัจฉริยะเอลิเซียม จากนั้นทำการบันทึกข้อมูลการทำงานของเมืองอัจฉริยะเอลิเซียมพบว่า ประสิทธิภาพของเมืองอัจฉริยะเอลิเซียมอยู่ในระดับ ดีมาก มีค่า เท่ากับ $\bar{x} = 92.5$
 $S.D = 0.27$

Research Title	Elysium smart city.	
Research	Mr. Photchara	Soros
	Mr. Wasin	Sermsuk
Research Consultants	Mr. Chattapluek	Seethim
	Ms. Prapaporn	Banthao
Organization	Technology computer departmeny	
	Chanthaburi technical collage	
Academic Year	2023	

Abstract

The objectives of this Elysium Smart City project are: 1. To create the Elysium Smart City system. 2. To study the principles of vibration sensors and RFID sensors. 3. To evaluate the efficiency of the Elysium Smart City.

From project implementation and project testing By testing the use and evaluating the efficiency of the Elysium Smart City. Then recorded the work data of the Elysium Smart City and found that The efficiency of the Elysium Smart City is at a very good level with a value equal to $\bar{x} = 92.5$ S.D = 0.27.

กิตติกรรมประกาศ

โครงการวิจัยฉบับนี้สำเร็จลุล่วงไปด้วยดี คณะผู้จัดทำได้รับความช่วยเหลือดูแลเอาใจใส่เป็นอย่างดีจากหลาย ๆ ฝ่าย โดยเฉพาะอาจารย์ที่ปรึกษา คือ นายฉัตรพฤษ์ สีทิม และนางสาวประภาพร บันทา ในการแนะนำ ตรวจสอบแก้ไข ให้ข้อเสนอแนะ ติดตามความก้าวหน้าในการดำเนินการวิจัย คณะผู้วิจัยรู้สึกซาบซึ้งในความกรุณาของอาจารย์เป็นอย่างยิ่ง และขอขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอขอบคุณอาจารย์ นายฉัตรพฤษ์ สีทิม นางสาวประภาพร บันทา และนายปิยพงษ์ เกตุภูงา กรรมการสอบโครงการวิจัยนี้ที่ได้กรุณาให้ข้อเสนอแนะ แก้ไข และให้แนวคิดต่าง ๆ ที่เป็นประโยชน์คณะผู้จัดทำ ขอขอบคุณเพื่อนๆ ที่ช่วยให้คำแนะนำดีๆ เกี่ยวกับการเลือกคำ และเกี่ยวกับโครงการชิ้นนี้ด้วย

สุดท้ายนี้หวังว่าโครงการฉบับนี้จะเป็นประโยชน์ต่อผู้อ่านและนักศึกษาไม่มากก็น้อย ประโยชน์และคุณค่าของโครงการฉบับนี้ ผู้จัดทำขอมอบให้แก่ บิดา มารดา และครูอาจารย์ วิทยาลัยเทคนิคจันทบุรี ที่ได้สั่งสอนอบรมมาจากอดีตจนปัจจุบันตลอดจนผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูปภาพ	ช
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการและการวิจัย	2
1.3 ขอบเขตของโครงการและงานวิจัย	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
1.5 สมมติฐานของโครงการและวิจัย	3
1.6 นิยามศัพท์	3
1.7 ระยะเวลาที่ใช้ในการจัดทำโครงการ	4
1.8 งบประมาณและทรัพยากร	5
บทที่ 2 แนวคิด ทฤษฎี เอกสารและงานวิจัยที่เกี่ยวข้อง	
2.1 ระบบ IOT	6
2.2 ESP32	9
2.3 ESP32 CAM	15
2.4 Relay 5V IOT	24
2.5 Ultrasonic Sensor	27
2.6 Terminal Block	30
2.7 Switch	31
2.8 LED	33
2.9 Switching	34
2.10 กระดาษชานอ้อย	35
2.11 NFC	35
2.12 Buzzer	42

สารบัญ(ต่อ)

	หน้า
2.13 จอ LCD	43
2.14 สมองกลฝังตัว	45
2.15 เอกสารงานวิจัยที่เกี่ยวข้อง	47
บทที่ 3 วิธีดำเนินงานโครงการ	
3.1 ศึกษาปัญหาและค้นคว้าข้อมูล	50
3.2 ออกแบบโมเดลเมือง โครงสร้างและวางแผนการเดินทางระบบไฟฟ้า	50
3.3 ดำเนินการสร้างโมเดลเมือง โครงสร้าง	52
3.4 เขียนโปรแกรม	54
3.5 การประเมินประสิทธิภาพ	55
3.6 การเก็บรวบรวมข้อมูลและการวิเคราะห์ข้อมูล	56
บทที่ 4 ผลการวิเคราะห์ข้อมูล	
4.1 ผลการทดลองและการแก้ไข	57
4.2 หาประสิทธิภาพของเมืองอัจฉริยะเอลิเซียม	58
บทที่ 5 สรุปผล ปัญหาอุปสรรค และข้อเสนอแนะ	
5.1 สรุปผล	59
5.2 ปัญหาและอุปสรรค	59
5.3 แนวทางแก้ไข	59
5.4 ข้อเสนอแนะ	59
บรรณานุกรม	60
ภาคผนวก	
ภาคผนวก ก. แบบขออนุมัติโครงการ	
ภาคผนวก ข. ผลการประเมินประสิทธิภาพ	
ภาคผนวก ค. คู่มือการใช้งาน	
ภาคผนวก ง. โค้ดที่ใช้ใน ระบบเมืองอัจฉริยะเอลิเซียม	
ประวัติผู้ทำโครงการ	

สารบัญตาราง

	หน้า
ตารางที่ 1.1 ระยะเวลาที่ใช้ในการจัดทำโครงการ	4
ตารางที่ 1.2 งบประมาณและทรัพยากร	5
ตารางที่ 4.1 หาประสิทธิภาพของเมืองอัจฉริยะเอลิเซีย	58

สารบัญรูปภาพ

	หน้า
รูปภาพที่ 2.1 แผนภาพองค์ประกอบของระบบ IOT	7
รูปภาพที่ 2.2 ESP32	9
รูปภาพที่ 2.3 ขาการใช้งานของ ESP32	9
รูปภาพที่ 2.4 หน้าเลือกเวอร์ชันของ ArduinoIDE	10
รูปภาพที่ 2.5 หน้าการ download ArduinoIDE	10
รูปภาพที่ 2.6 การเข้าโปรแกรม ArduinoIDE	11
รูปภาพที่ 2.7 หน้าการเข้าไปติดตั้งบอร์ด ESP32	11
รูปภาพที่ 2.8 การเพิ่ม link .json ลงในโปรแกรม	12
รูปภาพที่ 2.9 การเข้าไปติดตั้ง libraries ของบอร์ด ESP32	12
รูปภาพที่ 2.10 การค้นหาบอร์ด ESP32	12
รูปภาพที่ 2.11 การติดตั้ง libraries ของบอร์ด ESP32	13
รูปภาพที่ 2.12 หน้าการเลือก Port เชื่อมต่อ	13
รูปภาพที่ 2.13 หน้าการเลือกบอร์ดใช้งาน	13
รูปภาพที่ 2.14 Code ตัวอย่าง LED ติดกระพริบ	14
รูปภาพที่ 2.15 หน้าการ Upload Code	14
รูปภาพที่ 2.16 ESP32 CAM	15
รูปภาพที่ 2.17 การต่อวงจรเพื่ออัปโหลด Code ESP32 CAM	15
รูปภาพที่ 2.18 Jumper ของ usb ttl	15
รูปภาพที่ 2.19 การตั้งค่าบอร์ดเพื่ออัปโหลด Code	16
รูปภาพที่ 2.20 หน้า Serial Monitor แสดงหมายเลข IP	23
รูปภาพที่ 2.21 Relay 5V 4 Channel	24
รูปภาพที่ 2.22 การต่อใช้งาน Relay กับ ESP32	25
รูปภาพที่ 2.23 ผลลัพธ์ของโปรแกรม	26
รูปภาพที่ 2.24 Ultrasonic Sensor	27
รูปภาพที่ 2.25 ตัวอย่างการใช้งาน Ultrasonic Sensor	27
รูปภาพที่ 2.26 สูตรการเคลื่อนที่แนวราบ	28
รูปภาพที่ 2.27 ตำแหน่งของ Piezoelectric Ceramics	28

สารบัญรูปภาพ(ต่อ)

	หน้า
รูปภาพที่ 2.28 โครงสร้างภายใน Ultrasonic sensor	29
รูปภาพที่ 2.29 Terminal Block ทั้ง 3 แบบ	30
รูปภาพที่ 2.30 สัญลักษณ์ของสวิตช์	31
รูปภาพที่ 2.31 LED	33
รูปภาพที่ 2.32 switching power supply	34
รูปภาพที่ 2.33 หลักการทำงานของสวิตชิงเพาเวอร์ซัพพลาย	34
รูปภาพที่ 2.34 กระดาษขานอ้อย	35
รูปภาพที่ 2.35 การใช้ NFC	35
รูปภาพที่ 2.36 ประโยชน์ของ NFC	36
รูปภาพที่ 2.37 การต่อใช้งาน PN532 กับ ESP32	37
รูปภาพที่ 2.38 ผลลัพธ์ของโปรแกรม	42
รูปภาพที่ 2.39 Buzzer	42
รูปภาพที่ 2.40 การต่อใช้งาน Buzzer กับ ESP32	42
รูปภาพที่ 2.41 จอ LCD แบบ 16x2	43
รูปภาพที่ 2.42 การต่อจอ LCD กับ ESP32	44
รูปภาพที่ 2.43 ผลลัพธ์การต่อจอ LCD กับ ESP32	45
รูปภาพที่ 2.44 แผนภาพการทำงานของกระบวนการเชิงระบบ	45
รูปภาพที่ 2.45 แผนภาพการนำ Output มาเป็น Input	46
รูปภาพที่ 2.46 ตัวอย่างการใช้งานระบบสมองกลฝังตัว	46
รูปภาพที่ 3.1 แผนผังแสดงการดำเนินโครงการ	49
รูปภาพที่ 3.2 แบบโครงสร้างเมืองอัจฉริยะ	50
รูปภาพที่ 3.3 ออกแบบเมืองที่ 1	51
รูปภาพที่ 3.4 ออกแบบเมืองที่ 2	51
รูปภาพที่ 3.5 ออกแบบเมืองที่ 3	51
รูปภาพที่ 3.6 แบบการเดินสายไฟ	52
รูปภาพที่ 3.7 ตัดโมเดลเมืองที่ 1	52
รูปภาพที่ 3.8 ตัดโมเดลเมืองที่ 2 และเมืองที่ 3	53

สารบัญรูปภาพ(ต่อ)

	หน้า
รูปภาพที่ 3.9 ฐานไม้	53
รูปภาพที่ 3.10 ด้านหน้ากล่องวงจร	53
รูปภาพที่ 3.11 ภายในกล่องวงจร	54
รูปภาพที่ 3.12 โมเดลเมืองอัจฉริยะ	54
รูปภาพที่ 3.13 เขียนคำสั่งในโปรแกรม Arduino ลงบอร์ด ESP-32	54
รูปภาพที่ 3.14 ตัวอย่างโปรแกรมการทำงานของ เมืองอัจฉริยะเอลิเซียม	55

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

คำว่า IOT หรืออินเทอร์เน็ตในทุกสิ่ง (Internet of Things) หมายถึงเครือข่ายรวมของอุปกรณ์ที่เชื่อมต่อถึงกันและเทคโนโลยีที่อำนวยความสะดวกในการสื่อสารระหว่างอุปกรณ์กับระบบคลาวด์ ปัจจุบันระบบ IOT เข้ามามีความสำคัญอย่างมากในชีวิตของเราเพื่ออำนวยความสะดวกช่วยให้ชีวิตเราง่ายและมีประสิทธิภาพมากขึ้น จึงบ่อยครั้งที่มีการนำเทคโนโลยีต่างๆ มาใช้เพื่อช่วยอำนวยความสะดวก ซึ่งอุปกรณ์ IOT ในชีวิตประจำวันที่สามารถพบเห็นได้ง่ายคือ สมาร์ทโฟนโดยจะใช้เป็นอุปกรณ์สื่อสารที่ใช้สั่งการอุปกรณ์อื่น เช่น หลอดไฟรับคำสั่งเมื่อเจ้าของโทรศัพท์เข้าบ้าน การเข้า-ออกบ้านด้วยระบบความปลอดภัยที่สามารถตรวจสอบคนเข้าออกได้เพื่อเพิ่มความปลอดภัย การควบคุมอุณหภูมิด้วยเซนเซอร์ ความปลอดภัยบนท้องถนน หรือประตูอัตโนมัติจึงนับว่าเป็นเทคโนโลยีที่น่าสนใจและมีความสำคัญอย่างยิ่งในโลกยุคปัจจุบัน

เนื่องจากปัจจุบันชีวิตของเรามีความยากมากขึ้น จึงเป็นเหตุที่จำเป็นต้องพยายามพัฒนาให้เข้ากับยุค 4.0 โดยแก้ปัญหาต่างๆ ไม่ว่าจะเป็นปัญหาเศรษฐกิจตกต่ำ ปัญหาโรคติดต่อ ปัญหาในการค้า ปัญหาภาคอุตสาหกรรมการผลิต ปัญหาการขนส่ง และปัญหาด้านภาครัฐจึงต้องการเทคโนโลยีบางอย่างที่เข้ามาช่วยอำนวยความสะดวกในชีวิต นับว่าเป็นเทคโนโลยีที่ในยุคปัจจุบันนั้นค่อยๆทยอยเข้ามาปรับเปลี่ยนวิถีการใช้ชีวิตของเรา ซึ่งผู้ที่เห็นโอกาสดังนั้นก็ศึกษาหาความรู้เพื่อปรับตัวเพื่อที่จะเอาทรัพยากรต่าง ๆ นั้น ไปโฟกัสในการแก้ปัญหาต่างๆ เพื่อให้ลุล่วงปัญหานั้นไปได้

จากปัญหาข้างต้น ผู้จัดทำโครงการจึงคิดค้นและออกแบบเมืองอัจฉริยะเอลิเซียม เพื่อเป็นส่วนช่วยของการทำต้นแบบเมืองอัจฉริยะ ที่ช่วยอำนวยความสะดวกให้แก่ประชากรภายในเมือง เป็นการนำเอาเทคโนโลยีมาช่วยอำนวยความสะดวกและแก้ปัญหาที่เกิดขึ้นบ่อยครั้ง เช่นระบบช่วยจอดในที่จอดรถ เพื่อแก้ปัญหาการถอยชนในที่จอดรถ เพราะว่าไม่ใช่รถทุกคันที่จะมีระบบช่วยถอย ระบบควบคุมไฟเพื่อความสะดวกในการใช้งาน ระบบกล้องเพื่อที่ใช้ตรวจสอบเหตุการณ์ต่างๆในเมืองอัจฉริยะและมีระบบสแกนบัตร NFC เพื่อเข้าอาคารเพื่อแก้ปัญหาการโจรกรรม และสามารถช่วยคัดกรองเฉพาะบุคคลที่ได้รับอนุญาตเท่านั้น

1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อสร้างระบบเมืองอัจฉริยะเอลิเซีย
- 1.2.2 เพื่อสร้างระบบสแกนบัตร NFC
- 1.2.3 เพื่อสร้างระบบไฟฟ้าอัตโนมัติ
- 1.2.4 เพื่อหาประสิทธิภาพของเมืองเอลิเซีย

1.3 ขอบเขตของโครงการ

- 1.3.1 ประชากรและกลุ่มตัวอย่าง
 - 1.3.1.1 ประชากรคือ นักศึกษาในวิทยาลัยเทคนิคจันทบุรี
 - 1.3.1.2 กลุ่มผู้เชี่ยวชาญทางด้านไฟฟ้าและอิเล็กทรอนิกส์ที่มีประสบการณ์มากกว่า 3 ปี
- 1.3.2 ตัวแปรที่ศึกษา
 - 1.3.2.1 ตัวแปรต้น การใช้งานบัตร NFC เปิดประตู
 - 1.3.2.2 ตัวแปรตาม ประสิทธิภาพของระบบเมืองอัจฉริยะ
- 1.3.3 เครื่องมือที่ใช้ในโครงการ
 - 1.3.3.1 แบบทดสอบประสิทธิภาพของเมืองเอลิเซีย
- 1.3.4 วิธีการเก็บรวบรวมข้อมูล
 - 1.3.4.1 กำหนดกลุ่มประชากรและกลุ่มตัวอย่าง
 - 1.3.4.2 ศึกษาข้อมูลเบื้องต้น
 - 1.3.4.3 ออกแบบและสร้างระบบเมืองอัจฉริยะ
 - 1.3.4.4 นำไปใช้งานจริง
 - 1.3.4.5 ปรับปรุงและแก้ไขส่วนที่บกพร่อง
 - 1.3.4.6 สร้างแบบประเมินประสิทธิภาพการทำงาน
 - 1.3.4.7 เก็บรวบรวมแบบสอบถามทั้งหมดและนำมาประมวลให้สมบูรณ์ต่อไป
- 1.3.5 สถิติที่ใช้ในการวิเคราะห์ข้อมูล
 - 1.3.5.1 ค่าเฉลี่ย โดยใช้สูตรค่าเฉลี่ย

$$\bar{x} = \frac{\sum x}{n}$$

$\bar{x}$ = ค่าเฉลี่ยของคะแนน

$\sum x$ = ผลรวมของคะแนน

n = จำนวน

1.3.5.2 ส่วนเบี่ยงเบนมาตรฐานโดยใช้สูตร

$$S.D. = \frac{\sqrt{n \sum x^2 - (\sum x)^2}}{n(n-1)}$$

S.D. แทน ส่วนเบี่ยงเบนค่ามาตรฐาน

x แทน คะแนนแต่ละคน

n แทน จำนวนคน

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ได้ระบบเมืองอัจฉริยะเอลิเซียม

1.4.2 ได้ระบบสแกนบัตร NFC

1.4.3 ได้ระบบไฟฟ้าอัตโนมัติ

1.4.4 ได้เมืองเอลิเซียมที่มีประสิทธิภาพมากกว่า ร้อยละ 80 ขึ้นไป

1.5 สมมติฐานของโครงการและวิจัย

ได้ระบบเมืองอัจฉริยะเอลิเซียมที่มีประสิทธิภาพ ร้อยละ 80 ขึ้นไป

1.6 นิยามศัพท์

1.6.1 ระบบอัตโนมัติ คือ ระบบใดๆที่มีกลไกสามารถเริ่มทำงานได้ด้วยตัวเอง โดยทำงานตามโปรแกรมที่วางไว้

1.6.2 เซนเซอร์ คือ อุปกรณ์ที่ใช้ในการตรวจวัด ตรวจรู้ ค่าต่างๆ ในกระบวนการ เป็นอุปกรณ์ที่สัมผัสกับกระบวนการ หรืออยู่ในส่วนแรกของเครื่องมือวัดอุตสาหกรรมเลย ซึ่งเซนเซอร์ในงานเครื่องมือวัดนั้น อาจจะแสดงค่าการวัดได้ทันที หรือ จำเป็นต้องมีอุปกรณ์ในการแปลงค่าสัญญาณการวัด

1.6.3 อัจฉริยะ คือ วิเศษน่าอัศจรรย์ มีความรู้ความสามารถเกินกว่าระดับปรกติมาก

1.6.4 เทคโนโลยี คือ วิทยาการที่นำเอาความรู้ทางวิทยาศาสตร์มาประยุกต์ใช้ให้เกิดประโยชน์ ปฏิบัติและอุตสาหกรรม

1.6.5 อัลตราโซนิก คือ คลื่นเสียงที่มีความถี่สูงเกิน 20,000 Hz มากจนมนุษย์ไม่สามารถได้ยิน คลื่นเสียงที่มนุษย์สามารถได้ยินได้ดีนั้นจะอยู่ที่ประมาณ 20 Hz จนถึง 15 kHz โดยเฉลี่ยสำหรับบุคคลคนที่มีอายุเลยวัยเด็ก และประมาณ 20 Hz จนถึง 20 kHz แต่คลื่นอัลตราโซนิก จะมีความถี่อยู่ที่ 20 kHz ขึ้นไป

1.8 งบประมาณและทรัพยากร

ตารางที่ 1.2 งบประมาณที่ใช้ในการจัดทำโครงการ

ที่	รายการวัสดุ	จำนวน	หน่วย	ราคา/บาท
1.	บอร์ด Esp - 32	1	290	290
2.	เซนเซอร์ Ultrasonic	5	170	850
3.	Buzzer	1	100	100
4.	Relay 5 V	2	50	100
5.	Switching Power supply	1	150	150
6.	สวิตช์เปิด – ปิด	4	10	40
7.	หลอดไฟ - ฐาน	2	150	300
8.	ไม้อัดแบบแท่ง	20	50	1000
9.	สายไฟ Jumper	2	28	56
10	พลาสติก	1	390	390
11	บอร์ด ESP-32 CAM	1	300	300
รวม				3,576

บทที่ 2

แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้อง

ในการจัดทำเมืองอัจฉริยะเอลิเซียม มีวัตถุประสงค์เพื่อสร้างและหาประสิทธิภาพการทำงานของระบบเมืองอัจฉริยะเอลิเซียม ผู้จัดทำได้ทำการศึกษาทฤษฎีและเนื้อหาที่เกี่ยวข้องมีรายละเอียดดังต่อไปนี้

- 2.1 ระบบ IOT
- 2.2 ESP32
- 2.3 ESP32 CAM
- 2.4 RELAY 5V
- 2.5 Ultrasonic Sensor
- 2.6 Terminal Block
- 2.7 Switch
- 2.8 LED
- 2.9 Switching
- 2.10 กระดาษชานอ้อย
- 2.11 NFC
- 2.12 Buzzer
- 2.13 จอ LCD
- 2.14 สมองกลฝังตัว
- 2.15 เอกสารงานวิจัยที่เกี่ยวข้อง

2.1 ระบบ IOT (ที่มา : <https://digi.data.go.th/blog/what-is-iot-technology>)

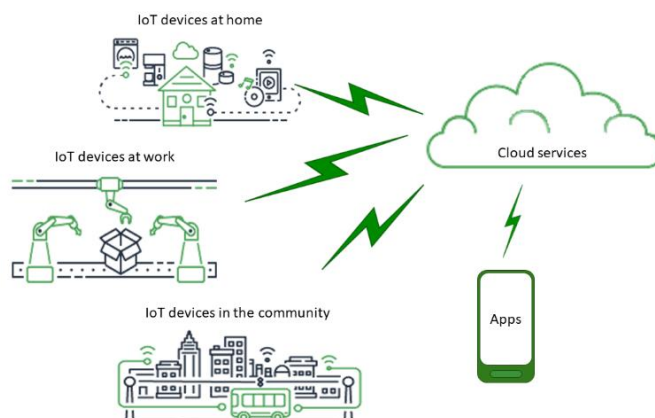
IoT ย่อมาจากคำว่า Internet of Things เกิดจากคำสองคำรวมกันได้แก่คำว่า Things ที่แปลว่าสิ่งของหลากหลายประเภท รวมกับคำว่า Internet ซึ่งก็คือเทคโนโลยีที่เชื่อมต่ออุปกรณ์อิเล็กทรอนิกส์เมื่อรวมคำเข้าด้วยกันก็เกิดเป็นความหมายว่าการที่อุปกรณ์ต่าง ๆ ที่อยู่รอบตัวเชื่อมต่อเข้าด้วยกันด้วยกันผ่านระบบ Internet องค์ประกอบของ IoT จะแบ่งออกได้เป็น 4 องค์ประกอบหลัก ได้แก่

2.1.1 อุปกรณ์ IoT เป็นองค์ประกอบหลักของระบบ ซึ่งหากปราศจากตัวอุปกรณ์อิเล็กทรอนิกส์ ระบบ IoT จะไม่สามารถเกิดขึ้นได้ ซึ่งตัวอุปกรณ์จะสามารถเป็นได้อุปกรณ์สั่งการโดยจะมีคอมพิวเตอร์ขนาดเล็กไว้ประมวลผลข้อมูล หลังจากได้ผลลัพธ์ออกมาจะมีระบบสั่งการซึ่งจะสั่งการไปยังอุปกรณ์ IoT อีกตัวซึ่งมีระบบที่สามารถรับคำสั่งได้ก็จะทำงานตามที่ได้รับคำสั่งมา

2.1.2 อุปกรณ์เกตเวย์ จะเป็นอุปกรณ์อีกประเภทที่จะเป็นตัวกลางในการส่งผ่านคำสั่งการจากอุปกรณ์ IoT ไปยังอุปกรณ์เป้าหมายที่ต้องรับคำสั่งเพื่อทำตาม

2.1.3 เครื่องบริการ Server หรือ Broker เป็นเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นจุดศูนย์กลางในการประมวลผล เนื่องจากอุปกรณ์มักจะใช้หน่วยประมวลผลขนาดเล็กทำให้บางครั้งไม่สามารถคำนวณคำสั่งที่ซับซ้อนได้จึงจำเป็นต้องมีหน่วยประมวลผลกลางที่รับคำสั่งจากอุปกรณ์ IoT ประเภทหนึ่งหลังจากนั้นก็จะใช้การประมวลผลเพื่อให้ได้คำสั่งอย่างง่ายส่งไปยังอุปกรณ์ IoT ที่เป็นผู้รับ

2.1.4 อุปกรณ์ฝั่งผู้ใช้งาน (User Device) เป็นส่วนของอุปกรณ์ที่จะใช้แสดงผลสถานะจากอุปกรณ์ IoT ทั้งหมดที่อยู่ในเครือข่าย เนื่องจากโดยปกติระบบ IoT จะเป็นการเชื่อมต่อเพื่อสื่อสารกันระหว่างอุปกรณ์อิเล็กทรอนิกส์ ซึ่งมนุษย์จำเป็นต้องมีอุปกรณ์อีกหนึ่งชิ้นเพื่อใช้ในการเชื่อมต่อกับระบบเพื่อดูสถานะการทำงาน



รูปภาพที่ 2.1 แผนภาพองค์ประกอบของระบบ IOT

2.1.5 ระบบ IoT ที่ใช้กันในปัจจุบันกัน ได้แก่ รถยนต์ที่เชื่อมต่อถึงกัน มีหลายวิธีที่ยานพาหนะต่างๆ เช่น รถยนต์ สามารถเชื่อมต่อกับอินเทอร์เน็ตได้ ซึ่งอาจเป็นการเชื่อมต่อผ่านกล่องติดรถยนต์อัจฉริยะ ระบบแสดงผลในตัวรถ หรือแม้แต่เกตเวย์ที่เชื่อมต่อถึงกันของยานพาหนะ โดยจะรวบรวมข้อมูลจากคันเร่ง เบรก มาตราวัดความเร็ว มาตราวัดระยะทาง ล้อ และถึงเชื้อเพลิง เพื่อเฝ้าติดตามประสิทธิภาพของผู้ขับขี่และความสมบูรณ์ของยานพาหนะ รถยนต์ที่เชื่อมต่อถึงกัน มีรูปแบบการใช้งานที่หลากหลาย ได้แก่

- 1) การเฝ้าติดตามกลุ่มรถยนต์ให้เข้าเพื่อเพิ่มประสิทธิภาพการใช้เชื้อเพลิงและลดต้นทุน
- 2) การช่วยให้ผู้ปกครองติดตามพฤติกรรมรถขับขี่ของบุตรหลาน

3) การแจ้งเตือนให้เพื่อนและครอบครัวทราบโดยอัตโนมัติในกรณีที่เกิดอุบัติเหตุรถชน

4) การคาดการณ์และการยับยั้งความจำเป็นในการบำรุงรักษายานพาหนะ

2.1.6 บ้านที่เชื่อมต่อถึงกัน อุปกรณ์สำหรับบ้านอัจฉริยะมุ่งเน้นไปที่การปรับปรุงประสิทธิภาพและความปลอดภัยของบ้านเป็นหลัก ตลอดจนถึงการปรับปรุงระบบเครือข่ายในบ้าน โดยอุปกรณ์ต่างๆ เช่น เตาไฟฟ้าอัจฉริยะจะตรวจสอบการใช้ไฟฟ้าและตัวควบคุมอุณหภูมิอัจฉริยะช่วยให้ควบคุมอุณหภูมิได้ดียิ่งขึ้น ระบบปลูกพืชด้วยน้ำสามารถใช้เซ็นเซอร์ IoT เพื่อจัดการสวน ในขณะที่เครื่องตรวจจับควันแบบ IoT สามารถตรวจจับควันบุหรี่ได้ นอกจากนี้ ระบบรักษาความปลอดภัยต่างๆ ภายในบ้าน เช่น กล้องประตู กล้องรักษาความปลอดภัย และเครื่องตรวจจับน้ำรั่ว สามารถตรวจจับและป้องกันภัยคุกคามได้ รวมถึงส่งการแจ้งเตือนให้แก่เจ้าของบ้านอีกด้วย โดยสามารถใช้อุปกรณ์ที่เชื่อมต่อถึงกันสำหรับบ้านเพื่อดำเนินการต่างๆ ได้แก่

- 1) การปิดอุปกรณ์ที่ไม่ได้ใช้งานโดยอัตโนมัติ
- 2) การจัดการและการบำรุงรักษาทรัพย์สินที่ให้เช่า
- 3) การค้นหาสิ่งของที่สูญหาย เช่น กุญแจหรือกระเป๋าตังค์
- 4) การปฏิบัติงานประจำวันให้เป็นระบบอัตโนมัติ เช่น การดูฝุ่น การชงกาแฟ ฯลฯ

2.1.7 เมืองอัจฉริยะ แอปพลิเคชัน IoT ทำให้การวางผังเมืองและการบำรุงรักษาโครงสร้างพื้นฐานมีประสิทธิภาพมากขึ้น ซึ่งรัฐบาลต่างๆ ใช้แอปพลิเคชัน IoT เพื่อจัดการกับปัญหาด้านโครงสร้างพื้นฐาน สุขภาพ และสิ่งแวดล้อม โดยคุณสามารถใช้แอปพลิเคชัน IoT เพื่อดำเนินการต่างๆ ได้แก่

- 1) การวัดคุณภาพอากาศและระดับการแผ่รังสี
- 2) การลดค่าไฟด้วยระบบไฟส่องสว่างอัจฉริยะ
- 3) การตรวจพบความต้องการในการบำรุงรักษาสำหรับโครงสร้างพื้นฐานต่างๆ ที่สำคัญ

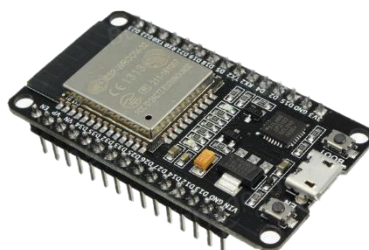
เช่น ถนนสะพาน และท่อส่งก๊าซ

- 4) การเพิ่มผลกำไรด้วยการจัดการที่จอดรถอย่างมีประสิทธิภาพ

2.1.8 อาคารอัจฉริยะอาคารต่างๆ เช่น วิทยาเขตของวิทยาลัยและอาคารพาณิชย์ใช้แอปพลิเคชัน IoT เพื่อเพิ่มประสิทธิภาพในการปฏิบัติงาน โดยคุณสามารถใช้อุปกรณ์ IoT ในอาคารอัจฉริยะเพื่อดำเนินการต่างๆ ได้แก่

- 1) การลดการใช้พลังงาน
- 2) การลดต้นทุนการบำรุงรักษาให้ต่ำลง
- 3) การใช้พื้นที่ทำงานให้มีประสิทธิภาพมากขึ้น

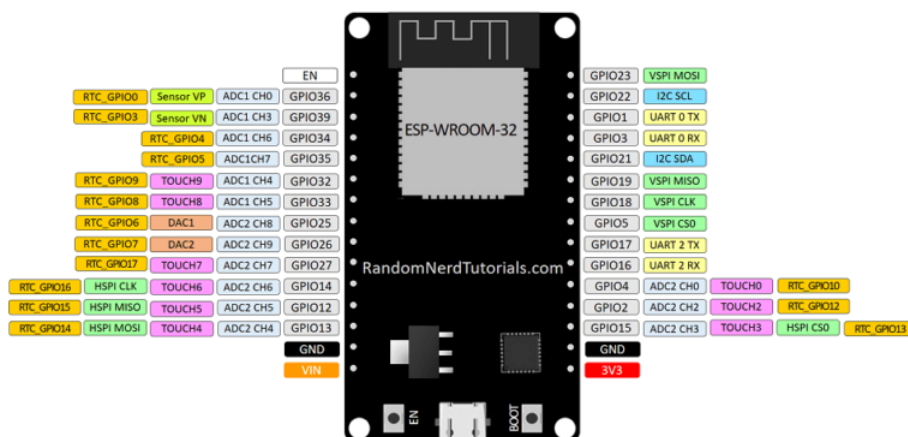
2.2 ESP32 (ที่มา : <https://tinyurl.com/bdfj6ykn>, <https://techtalk2apply.com/what-is-esp32/>)



รูปภาพที่ 2.2 ESP32

ESP32 คือ WIFI microcontroller ที่ถูกพัฒนาต่อจาก ESP8266 โดยเพิ่ม CPU เป็น 2 core, Wi-Fi ที่เร็วขึ้น, มีขา GPIO ให้ใช้งานมากขึ้น และรองรับ Bluetooth อีกด้วย นอกจากนี้ ESP32 ยังมาพร้อมกับ touch-sensitive pins ที่สามารถใช้ปลุก ESP32 จากโหมด deep sleep และยังมี hall effect sensor และ temperature sensor ในตัว (รุ่นล่าสุดของ ESP32 ไม่มี temperature sensor ในตัวอีกแล้ว) และแน่นอนว่าด้วยคุณสมบัติที่ดีกว่ามากจึงทำให้ ESP32 มีราคาแพงกว่า ESP8266 พอสมควร

ESP32 DEVKIT V1 - DOIT version with 30 GPIOs

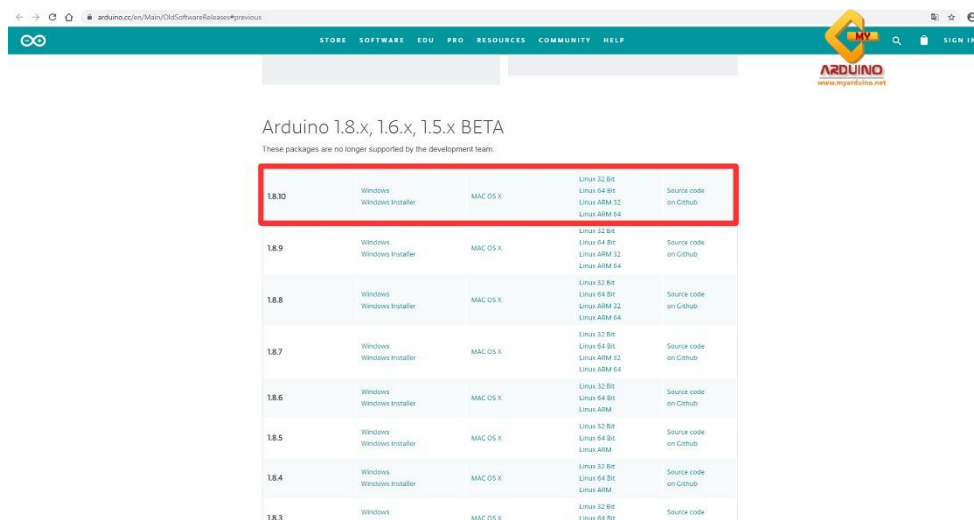


รูปภาพที่ 2.3 ขาการใช้งานของ ESP32

2.2.1 ขั้นตอนการติดตั้ง ArduinoIDE และ บอร์ด ESP32

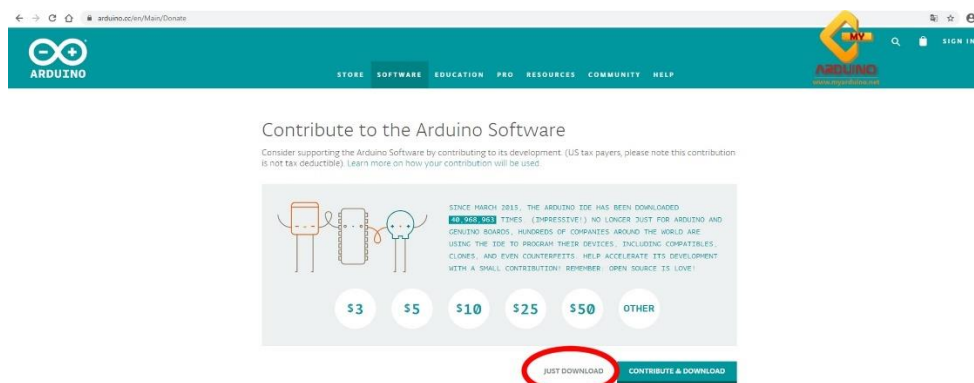
1) ดาวน์โหลด ArduinoIDE จาก <https://www.arduino.cc/>

2) เลือกระบบปฏิบัติการของเครื่องคอมพิวเตอร์ที่ท่านจะใช้ในการติดตั้ง



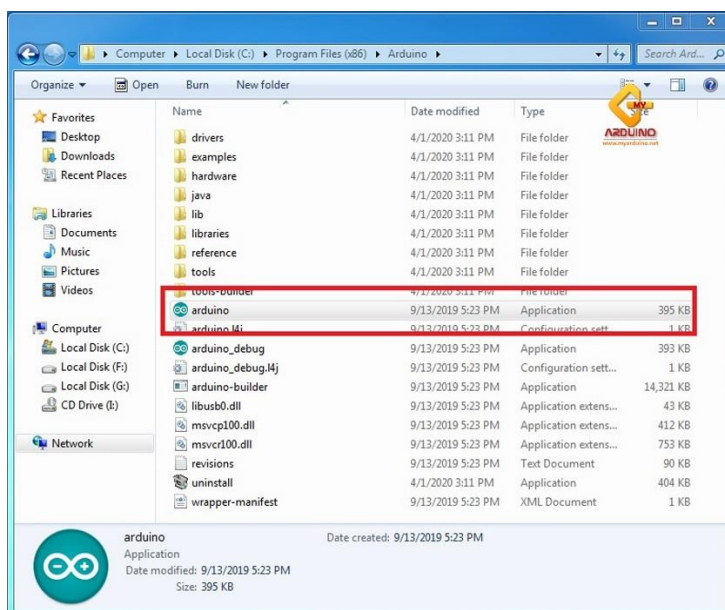
รูปภาพที่ 2.4 หน้าเลือกเวอร์ชันของ ArduinoIDE

3) กด JUST DOWNLOAD และทำการติดตั้งโปรแกรม (หากต้องการร่วมบริจาคช่วยการพัฒนา Arduino Software สามารถกด CONTRIBUTE & DOWNLOAD)



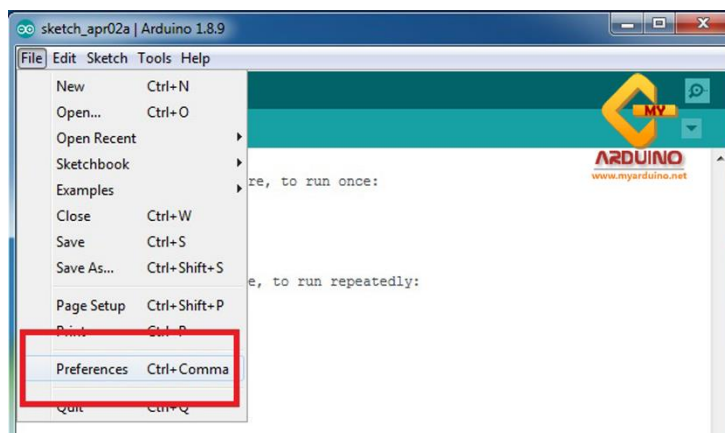
รูปภาพที่ 2.5 หน้าการ download ArduinoIDE

4) Double Click ที่ไฟล์ arduino.exe เพื่อเปิดโปรแกรม Arduino IDE



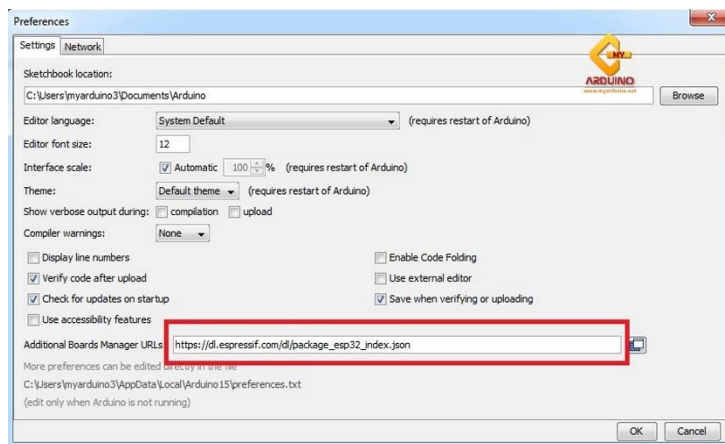
รูปภาพที่ 2.6 การเข้าโปรแกรม ArduinoIDE

5) เมื่อเปิดโปรแกรมขึ้นมาแล้วไปที่เมนู File -> Preferences เพื่อติดตั้งบอร์ด NodeMCU/ESP32 แบบออนไลน์



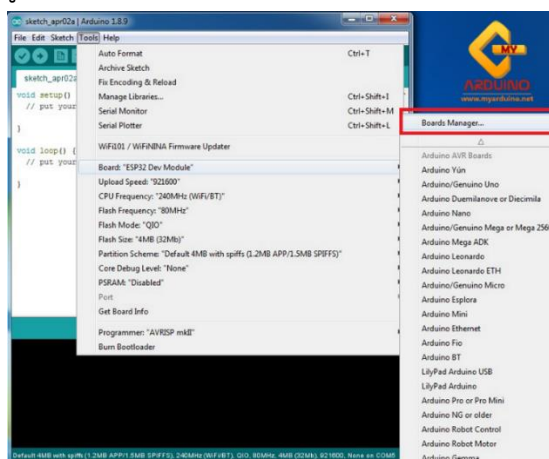
รูปภาพที่ 2.7 หน้าการเข้าไปติดตั้งบอร์ด ESP32

6) เพิ่ม https://dl.espressif.com/dl/package_esp32_index.json ลงในช่อง Additional Boards Manager URLs



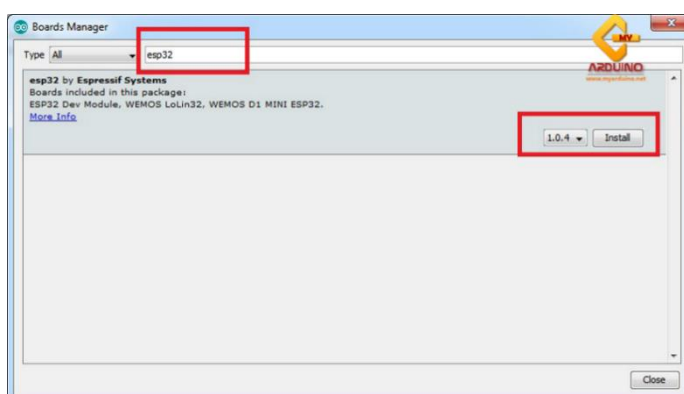
รูปภาพที่ 2.8 การเพิ่ม link .json ลงในโปรแกรม

7) คลิกไปที่เมนู Tools -> Board -> Board Manager



รูปภาพที่ 2.9 การเข้าไปติดตั้ง libraries ของบอร์ด ESP32

8) พิมพ์คำว่า ESP32 ลงในช่อง และเริ่มต้นติดตั้ง



รูปภาพที่ 2.10 การค้นหาคอร์ด ESP32

14) พิมพ์ Code ตัวอย่างที่หน้าต่างของโปรแกรม ArduinoIDE

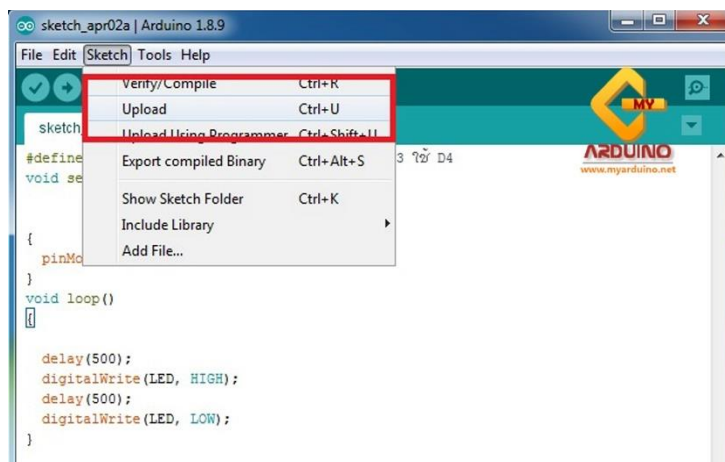
```

1  #define LED 2
2  void setup() {
3      // Set pin mode
4      pinMode(LED,OUTPUT);
5  }
6  void loop() {
7      delay(500);
8      digitalWrite(LED,HIGH);
9      delay(500);
10     digitalWrite(LED,LOW);
11 }

```

รูปภาพที่ 2.14 Code ตัวอย่าง LED ติดกระพริบ

15) จากนั้นทำการUpload Code ไปที่บอร์ด ESP32 ที่เลือกไว้



รูปภาพที่ 2.15 หน้าการ Upload Code

16) เมื่อ Upload ไปสักครู๋จะปรากฏข้อความ “Connecting.....”

17) ให้กดไปที่ปุ่ม “BOOT” ที่อยู่บนบอร์ด ESP32 ไว้ 2 วินาทีแล้วปล่อย

18) เมื่อกดปุ่ม BOOT แล้วให้รอสักครู๋ก็จะเสร็จสิ้นการ Upload Code และไฟสถานะที่ขา GPIO2 ก็จะติดกะพริบขึ้น

2.3 ESP32 CAM (ที่มา : <https://tinyurl.com/35s8xfcp>)

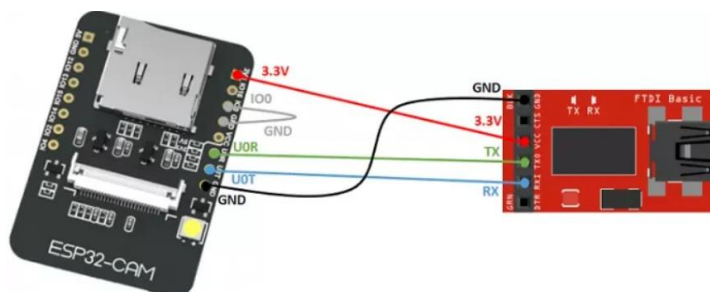


รูปภาพที่ 2.16 ESP32 CAM

ESP32-CAM เป็นโมดูลกล้องขนาดเล็กมาก ที่มาพร้อมกับชิป ESP32-S

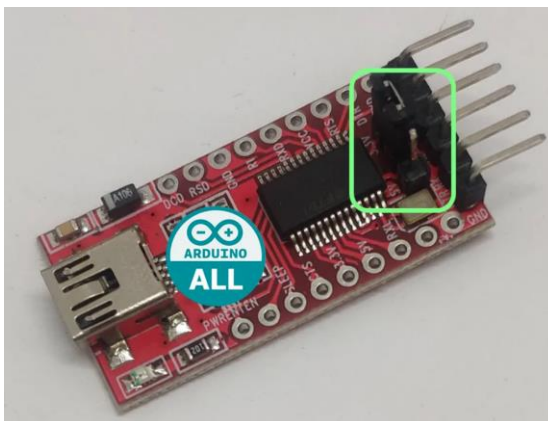
2.3.1 ทดลองใช้ Arduino ESP32-CAM

1) การต่อวงจรเพื่ออัปโหลด Code ESP32-CAM เนื่องจากบอร์ด ESP32-CAM นี้ไม่มีส่วน USB TTL สำหรับการอัปโหลดโปรแกรมลงบอร์ด ดังนั้นเราจึงต้องต่อสายไฟกับโมดูล USB TTL เพิ่มเพื่อให้อัปโหลดได้



รูปภาพที่ 2.17 การต่อวงจรเพื่ออัปโหลด Code ESP32 CAM

2) ตั้ง Jumper ของ usb ttl ไปที่ 3.3V

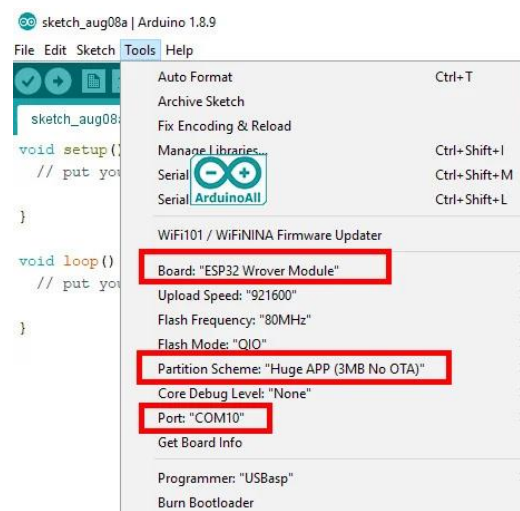


รูปภาพที่ 2.18 Jumper ของ usb ttl

3) ต่อ ESP32-CAM กับโมดูล USB TTL

- 3V3 - VCC/3.3V
- 5V - 5V // สำหรับจ่ายไฟเพิ่มให้บอร์ด เพราะไฟจาก usb ttl อาจไม่พอ
- GND - GND
- UOR - TX
- UOT - RX
- IO0 - GND // ขานี้ต้องต่อ เพื่อกำหนดเป็นโหมด flash สำหรับอัปโหลดโปรแกรม

4) ตั้งค่าบอร์ด ESP32 ดังรูปนี้



รูปภาพที่ 2.19 การตั้งค่าบอร์ดเพื่ออัปโหลด Code

Code ตัวอย่าง

```
#include "esp_camera.h"
#include <WiFi.h>
#include "esp_timer.h"
#include "img_converters.h"
#include "Arduino.h"
#include "fb_gfx.h"
#include "soc/soc.h" //disable brownout problems
#include "soc/rtc_cntl_reg.h" //disable brownout problems
#include "dl_lib.h"
#include "esp_http_server.h"

const char* ssid = "REPLACE_WITH_YOUR_SSID"; // แก้ชื่อ wifi
const char* password = "REPLACE_WITH_YOUR_PASSWORD";
```

```
#define PART_BOUNDARY "1234567890000000000000987654321"
M5STACK WITHOUT PSRAM

#define CAMERA_MODEL_AI_THINKER
#if defined(CAMERA_MODEL_WROVER_KIT)
#define PWDN_GPIO_NUM    -1
#define RESET_GPIO_NUM   -1
#define XCLK_GPIO_NUM     21
#define SIOD_GPIO_NUM     26
#define SIOC_GPIO_NUM     27
#define Y9_GPIO_NUM       35
#define Y8_GPIO_NUM       34
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
#define Y5_GPIO_NUM       19
#define Y4_GPIO_NUM       18
#define Y3_GPIO_NUM       5
#define Y2_GPIO_NUM       4
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM     22
#elif defined(CAMERA_MODEL_M5STACK_PSRAM)
#define PWDN_GPIO_NUM     -1
#define RESET_GPIO_NUM    15
#define XCLK_GPIO_NUM     27
#define SIOD_GPIO_NUM     25
#define SIOC_GPIO_NUM     23
#define Y9_GPIO_NUM       19
#define Y8_GPIO_NUM       36
#define Y7_GPIO_NUM       18
#define Y6_GPIO_NUM       39
#define Y5_GPIO_NUM       5
```

```
#define Y4_GPIO_NUM    34
#define Y3_GPIO_NUM    35
#define Y2_GPIO_NUM    32
#define VSYNC_GPIO_NUM 22
#define HREF_GPIO_NUM  26
#define PCLK_GPIO_NUM  21
#elif defined(CAMERA_MODEL_M5STACK_WITHOUT_PSRAM)
#define PWDN_GPIO_NUM  -1
#define RESET_GPIO_NUM 15
#define XCLK_GPIO_NUM  27
#define SIOD_GPIO_NUM   25
#define SIOC_GPIO_NUM   23
#define Y9_GPIO_NUM     19
#define Y8_GPIO_NUM     36
#define Y7_GPIO_NUM     18
#define Y6_GPIO_NUM     39
#define Y5_GPIO_NUM     5
#define Y4_GPIO_NUM     34
#define Y3_GPIO_NUM     35
#define Y2_GPIO_NUM     17
#define VSYNC_GPIO_NUM  22
#define HREF_GPIO_NUM   26
#define PCLK_GPIO_NUM   21
#elif defined(CAMERA_MODEL_AI_THINKER)
#define PWDN_GPIO_NUM   32
#define RESET_GPIO_NUM  -1
#define XCLK_GPIO_NUM    0
#define SIOD_GPIO_NUM    26
#define SIOC_GPIO_NUM    27
#define Y9_GPIO_NUM     35
#define Y8_GPIO_NUM     34
```

```

#define Y7_GPIO_NUM    39
#define Y6_GPIO_NUM    36
#define Y5_GPIO_NUM    21
#define Y4_GPIO_NUM    19
#define Y3_GPIO_NUM    18
#define Y2_GPIO_NUM    5
#define VSYNC_GPIO_NUM 25
#define HREF_GPIO_NUM  23
#define PCLK_GPIO_NUM  22
#else
#error "Camera model not selected"
#endif

static const char* _STREAM_CONTENT_TYPE = "multipart/x-mixed-
replace;boundary=" PART_BOUNDARY;
static const char* _STREAM_BOUNDARY = "\r\n--" PART_BOUNDARY "\r\n";
static const char* _STREAM_PART = "Content-Type: image/jpeg\r\nContent-
Length: %u\r\n\r\n";
httpd_handle_t stream_httpd = NULL;
static esp_err_t stream_handler(httpd_req_t *req){
    camera_fb_t * fb = NULL;
    esp_err_t res = ESP_OK;
    size_t _jpg_buf_len = 0;
    uint8_t * _jpg_buf = NULL;
    char * part_buf[64];
    res = httpd_resp_set_type(req, _STREAM_CONTENT_TYPE);
    if(res != ESP_OK){
        return res;
    }
    while(true){
        fb = esp_camera_fb_get();
        if (!fb) {

```

```

Serial.println("Camera capture failed");
res = ESP_FAIL;
} else {
  if(fb->width > 400){
    if(fb->format != PIXFORMAT_JPEG){
      bool jpeg_converted = frame2jpg(fb, 80, &_jpg_buf, &_jpg_buf_len);
      esp_camera_fb_return(fb);
      fb = NULL;
      if(!jpeg_converted){
        Serial.println("JPEG compression failed");
        res = ESP_FAIL;
      }
    } else {
      _jpg_buf_len = fb->len;
      _jpg_buf = fb->buf;
    }
  }
}

if(res == ESP_OK){
  size_t hlen = snprintf((char *)part_buf, 64, _STREAM_PART, _jpg_buf_len);
  res = httpd_resp_send_chunk(req, (const char *)part_buf, hlen);
}

if(res == ESP_OK){
  res = httpd_resp_send_chunk(req, (const char *)_jpg_buf, _jpg_buf_len);
}

if(res == ESP_OK){
  res = httpd_resp_send_chunk(req, _STREAM_BOUNDARY,
    strlen(_STREAM_BOUNDARY));
}

if(fb){
  esp_camera_fb_return(fb);
}

```

```

fb = NULL;
_jpg_buf = NULL;
} else if(_jpg_buf){
free(_jpg_buf);
_jpg_buf = NULL;
}
if(res != ESP_OK){
break;
}
}
return res;
}

void startCameraServer(){
httpd_config_t config = HTTPD_DEFAULT_CONFIG();
config.server_port = 80;
httpd_uri_t index_uri = {
.uri      = "/",
.method   = HTTP_GET,
.handler  = stream_handler,
.user_ctx = NULL
};
if (httpd_start(&stream_httpd, &config) == ESP_OK) {
httpd_register_uri_handler(stream_httpd, &index_uri);
}
}

void setup() {
WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0); //disable brownout
detector
Serial.begin(115200);
Serial.setDebugOutput(false);
camera_config_t config;

```

```

config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
if(psramFound()){
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;
    config.fb_count = 2;
} else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;
    config.fb_count = 1;
}

esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {

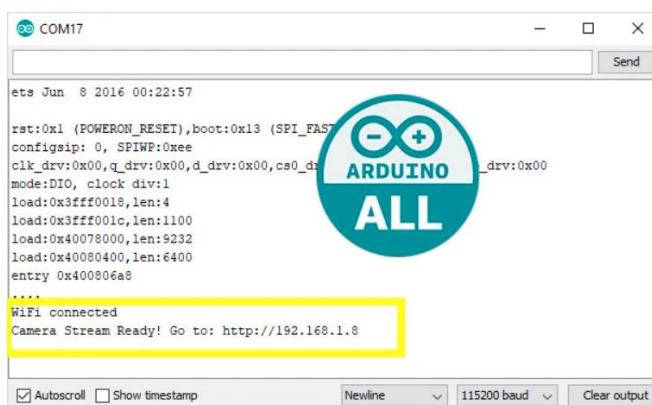
```

```

Serial.printf("Camera init failed with error 0x%x", err);
return;
}
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
Serial.print("Camera Stream Ready! Go to: http://");
Serial.print(WiFi.localIP());
startCameraServer();
}
void loop() {
  delay(1);
}

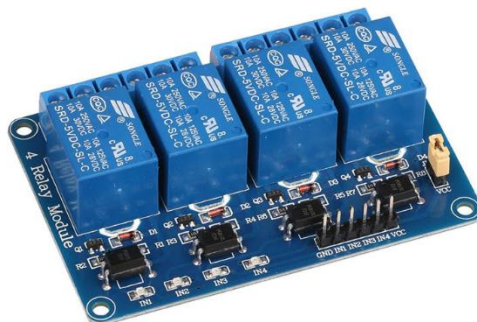
```

5) เมื่ออัปโหลดเสร็จแล้ว ถอดสายไฟที่ต่อกับขา IO0 และ GND ออก เพื่อเข้าโหมดทำงานปกติ เปิดดูที่ Serial Monitor กดปุ่ม reset ของบอร์ด ESP-32 CAM จะเห็นเลข IP ของกล้องวงจรปิดของเรา พิมพ์เลข IP ในหน้าเว็บ จะเห็นว่าการใช้งานได้



รูปภาพที่ 2.20 หน้า Serial Monitor แสดงหมายเลข IP

2.4 RELAY 5V IOT (ที่มา : <http://www.iot.codemobiles.com>)



รูปภาพที่ 2.21 Relay 5V 4 Channel

รีเลย์ (Relay) เป็นอุปกรณ์ที่เปลี่ยนพลังงานไฟฟ้าให้เป็นพลังงานแม่เหล็ก เพื่อใช้ในการ ดึงดูดหน้าสัมผัสของคอนแทคให้เปลี่ยนสถานะ โดยการป้อนกระแสไฟฟ้าให้กับขดลวด เพื่อทำการปิด หรือเปิดหน้าสัมผัสคล้ายกับสวิตช์อิเล็กทรอนิกส์ ซึ่งเราสามารถนำรีเลย์ไปประยุกต์ใช้ ในการควบคุม วงจรต่าง ๆ ในงานช่างอิเล็กทรอนิกส์มากมาย โมดูลรีเลย์ 2ช่อง 5V (2 Channel Relay Module) เป็นโมดูลที่ใช้ควบคุมโหลดได้ทั้งแรงดันไฟฟ้า DC และ AC ซึ่งโหลดสูงสุด (Maximum Load) คือ AC 250V/10A, DC 30V/10A โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณ logic TTL ทำงาน ด้วยสัญญาณแบบ Active Low, กระแสขับรีเลย์ (Drive Current) 15-20mA., มีการออกแบบให้เป็น Isolate ด้วย Optocoupler, มี LED แสดงสถานะ Relay สามารถนำไปประยุกต์ใช้งาน PLC Control, บ้านอัจฉริยะ, ใช้ในโรงงานอุตสาหกรรม หรืองานอื่นๆ ขึ้นอยู่กับการเขียนโปรแกรมและการ ต่อใช้งานภายนอก สามารถเชื่อมต่อใช้งานกับบอร์ด Raspberry Pi, Arduino, ARM, MCS-51, AVR, PIC, 8051, DSP, MSP430, TTL logic

2.4.1 คุณสมบัติของอุปกรณ์

1) ไฟเลี้ยงโมดูลรีเลย์ VCC = 5VDC.

ควบคุมโหลดได้ทั้งแรงดันไฟฟ้า AC ได้สูงสุด 250VAC 10A หรือ แรงดันไฟฟ้า DC ได้สูงสุด 30VDC 10A (Maximum Load)

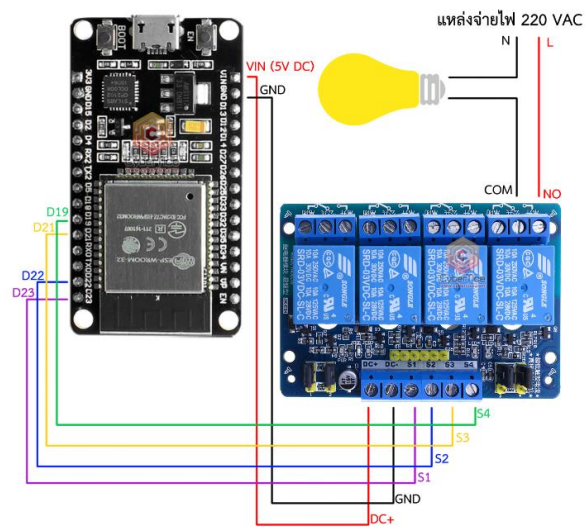
2) ระดับสัญญาณ Input ควบคุมแบบ TTL ทำงานด้วยสัญญาณแบบ Active Low

3) กระแสขับรีเลย์ (Drive Current) 15-20mA.

4) มีการออกแบบให้เป็น Isolate ด้วย Optocoupler

5) มี LED แสดงสถานะ Relay

6) โมดูลขนาด 3.85cm.(กว้าง) x 5.05cm.(ยาว) x 1.85cm.(สูง)



รูปภาพที่ 2.22 การต่อใช้งาน Relay กับ ESP32

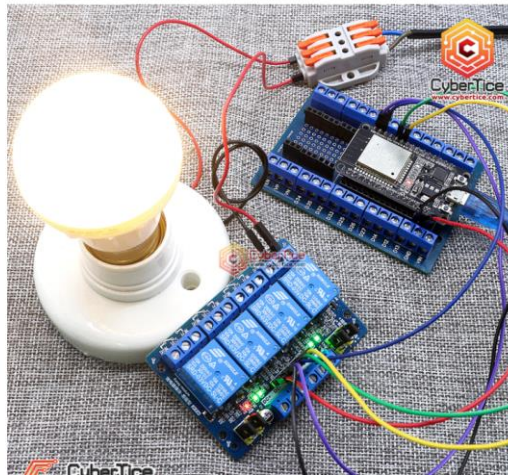
Code ตัวอย่าง

```
int S1 = 23;
int S2 = 22;
int S3 = 21;
int S4 = 19;

void setup() {
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(S4, OUTPUT);
}

void loop() {
  digitalWrite(S1, HIGH);
  delay(1000);
  digitalWrite(S2, HIGH);
  delay(1000);
  digitalWrite(S3, HIGH);
  delay(1000);
  digitalWrite(S4, HIGH);
```

```
delay(3000);  
digitalWrite(S4, LOW);  
delay(1000);  
digitalWrite(S3, LOW);  
delay(1000);  
digitalWrite(S2, LOW);  
delay(1000);  
digitalWrite(S1, LOW);  
delay(3000);  
}
```



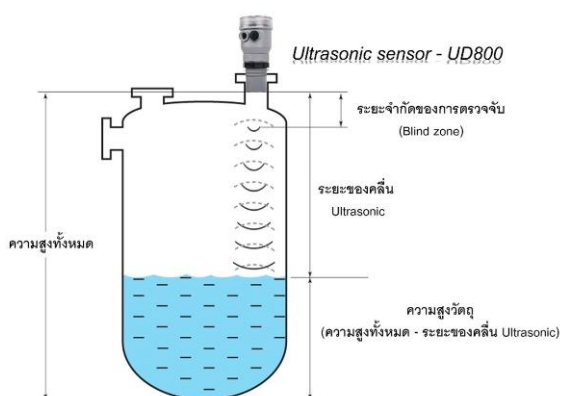
รูปภาพที่ 2.23 ผลลัพธ์ของโปรแกรม

2.5 Ultrasonic Sensor (ที่มา : <https://www.omi.co.th/th/article/ultrasonic-sensor>)



รูปภาพที่ 2.24 Ultrasonic Sensor

Ultrasonic sensor (อ่านว่า: อัลตราโซนิกเซ็นเซอร์) คือ อุปกรณ์สำหรับวัดระดับหรือระยะทางชนิดหนึ่งโดยใช้คลื่น Ultrasonic ซึ่งอาศัยหลักการสะท้อนของคลื่นความถี่สูง Ultrasonic โดยอุปกรณ์จะปล่อยคลื่น Ultrasonic ให้กระทบกับวัตถุ จากนั้นรอกคลื่น Ultrasonic สะท้อนกลับมาที่เซ็นเซอร์เพื่อคำนวณหาระยะทางที่วัดได้ นอกเหนือจาก Ultrasonic sensor แล้ว ยังมีเซ็นเซอร์ชนิดอื่นๆอีกที่ใช้ในการวัดระยะได้แก่ Radar sensor, Hydrostatic sensor เป็นต้น



รูปภาพที่ 2.25 ตัวอย่างการใช้งาน Ultrasonic Sensor

2.5.1 จุดประสงค์ของ Ultrasonic sensor นั้น ถูกแบ่งออกเป็น 2 ลักษณะการใช้งานหลัก ๆ คือ

- 1) ใช้เพื่องานตรวจสอบ (Monitoring) และดูผลลัพธ์ของข้อมูล เช่น ใช้แสดงผลของระดับน้ำในถัง, แสดงผลปริมาณข้าวเปลือกใน Silo หรือ ระดับน้ำแม่น้ำ เป็นต้น
- 2) ใช้เพื่อการควบคุม (Control) อะไรบางอย่าง ยกตัวอย่างเช่น การควบคุมระดับสารเคมีในถังให้อยู่ในระดับที่ต้องการตลอดเวลา เป็นต้น

2.5.2 เซ็นเซอร์วัดระดับ (Level sensor) ประเภท Ultrasonic หรือ Ultrasonic sensor เป็นเซ็นเซอร์ที่ต้องอาศัยหลักการของการสะท้อนคลื่นความถี่ Ultrasonic ในการตรวจจับวัตถุต่าง ๆ

Ultrasonic sensor นั้นจำเป็นต้องอาศัยตัวกลางในการเดินทาง เช่น อากาศ แก๊ส หรือของเหลว จึงทำให้ Ultrasonic sensor สามารถใช้งานตรวจจับวัตถุได้หลากหลายชนิด ซึ่งวัตถุที่มีสถานะของเหลว โดยที่เป็นสารเคมีหรือมีความหนืดก็สามารถใช้ Ultrasonic sensor ในการตรวจจับได้ และ Ultrasonic sensor มีความถี่ไปตั้งแต่ 20000Hz ขึ้นไปซึ่งเป็นความถี่ที่สูงเกินกว่ามนุษย์จะสามารถรับรู้ได้ โดยการคำนวณหาระยะของคลื่น Ultrasonic จะเป็นไปตามสูตรการเคลื่อนที่แนวราบดังนี้

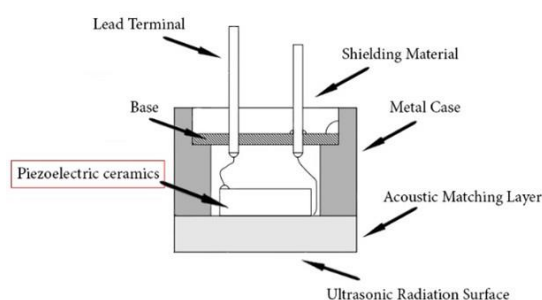
$$S = V_{\text{เสียง}} \cdot \left(\frac{t}{2}\right)$$

รูปภาพที่ 2.26 สูตรการเคลื่อนที่แนวราบ

โดยที่ S = ระยะทาง (m)
 $V_{\text{เสียง}}$ = ความเร็วเสียง (m/s)
 t = เวลาในการเดินทางของคลื่น Ultrasonic ทั้งขาไป-ขากลับ (s)

การทำงานภายใน Ultrasonic sensor

ในการทำงานของ Ultrasonic sensor จะมีส่วนประกอบสำคัญที่มีสัมพันธ์กัน ซึ่งแต่ละอุปกรณ์นั้นจะมีหน้าที่แตกต่างกัน ส่วนแรกคือวงจรกำเนิดสัญญาณความถี่ (Oscillator) จะเป็นวงจรอิเล็กทรอนิกส์ที่ผลิตสัญญาณออกมาซ้ำๆ กันคือความถี่สั่นพ้อง (resonance) และถูกส่งไปยัง Piezoelectric Ceramics ดังภาพ (A) เพื่อให้เกิดการสั่นพ้องกับความถี่ธรรมชาติของ Piezoelectric Ceramics ส่งผลทำให้เกิดค่าแอมพลิจูดสูงสุดเท่าที่จะสามารถเกิดขึ้นได้ และส่งคลื่น Ultrasonic ออกไป



รูปภาพที่ 2.27 ตำแหน่งของ Piezoelectric Ceramics

จากนั้นเมื่อวงจรกำเนิดสัญญาณความถี่ (Oscillator) สร้างสัญญาณขึ้นมา จะถูกส่งไปยัง Ultrasonic Transceiver ดังภาพ (B) จะทำหน้าที่แปลงสัญญาณที่ได้จาก Oscillator ในรูปแบบแรงดันไฟฟ้าไปเป็นคลื่น Ultrasonic และส่งออกไปยังวัตถุที่ต้องการตรวจจับหรือวัดระดับ และ Ultrasonic Transceiver ยังทำหน้าที่เป็นตัวรับคลื่น Ultrasonic ที่สะท้อนกลับมาจากวัตถุและแปลงเป็นสัญญาณแรงดันไฟฟ้าเพื่อนำไปใช้งานต่อไป ซึ่งหลังจาก Ultrasonic Transceiver ได้รับคลื่น Ultrasonic ที่สะท้อนมาจากวัตถุ จะแปลงเป็นแรงดันไฟฟ้าแต่มีขนาดเล็ก ดังนั้นแล้วจึงต้องขยายสัญญาณแรงดันไฟฟ้าด้วย Amplifier ดังภาพ (B) ก่อนที่จะนำไปใช้งาน และในกระบวนการสุดท้ายของ Ultrasonic sensor สัญญาณที่ถูกขยายจาก Amplifier จะถูกส่งเข้ามาที่วงจร output ดังภาพ (B) เพื่อแปลงเป็นสัญญาณมาตรฐานเช่น 4-20mA, 0-10V หรืออื่นๆ และนำสัญญาณเหล่านี้ไปเชื่อมต่อกับอุปกรณ์ภายนอกอีกที เช่น PLC, indicator, Controller, Relay หรือตัวควบคุมอื่น ๆ

2.5.3 ประเภทวงจร วงจร output ของ Ultrasonic sensor จะมีอยู่ด้วยกัน 3 ประเภท ดังนี้

1) วงจร output แบบไม่ต่อเนื่อง (Discrete Output)

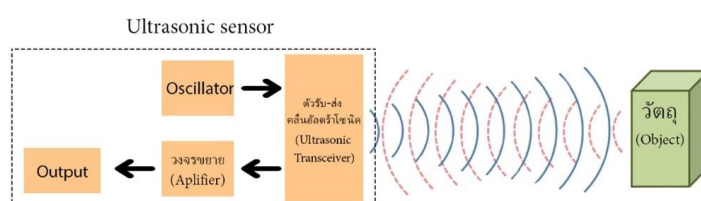
ทำงานแบบเปิดปิดเหมาะกับการตรวจจับวัตถุภายในถังเก็บเป็นจุด (ตรวจสอบภายในถังเก็บว่ามีปริมาตรมาก น้อย หรือไม่มี)

2) วงจร output แบบต่อเนื่อง (Continuous Output)

จะมี output เป็น Analog Output สัญญาณแรงดันไฟฟ้าหรือกระแสไฟฟ้ามาตรฐาน เช่น 4-20mA, 0-10V เป็นต้น เหมาะกับการตรวจจับที่อยากทราบตำแหน่ง ระยะทาง เช่น ระดับน้ำภายในถังเก็บ

3) วงจร output สำหรับส่งข้อมูล (Communication Data Output)

จะทำงานคล้ายวงจร output แบบต่อเนื่อง แต่แตกต่างกับที่สัญญาณที่ส่งออกมาเป็นสัญญาณดิจิทัล เช่น IO Link, CC Link, PROFIBUS ที่ใช้สื่อสารกับอุปกรณ์พิเศษ เช่น PLC



รูปภาพที่ 2.28 โครงสร้างภายใน Ultrasonic sensor

2.6 Terminal Block (ที่มา : <https://www.changfi.com>)

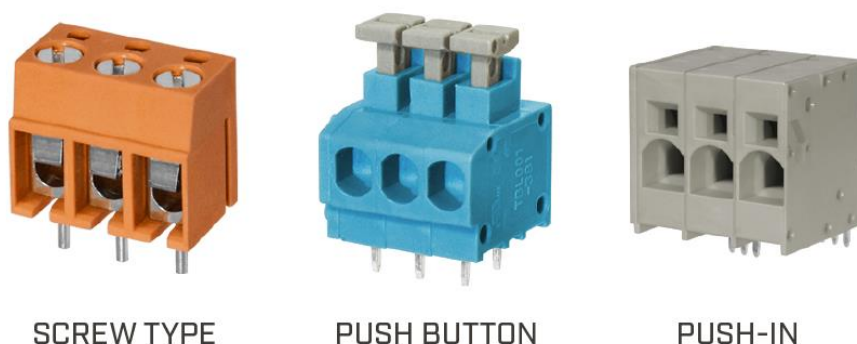
Terminal Block เป็นเครื่องมือที่ใช้เชื่อมต่อระหว่างสายไฟ หรือเป็นเครื่องมือที่ต่อสายไฟในตู้คอนโทรล ที่มีความปลอดภัย หรืออาจใช้เป็นจุดพักสายไฟ เพื่อง่ายต่อการซ่อมบำรุงต่าง ๆ นอกจากนี้ ยังสามารถใช้ตรวจสอบจุดที่มีปัญหาได้ง่าย และช่วยจัดเก็บสายไฟให้เป็นระเบียบเรียบร้อย ส่วนวัสดุเป็นฉนวนทำจากพลาสติกประเภทโพลีเอไมด์ มีคุณสมบัติ ทนต่อความร้อน กรด – ด่าง และไม่ติดไฟ อีกทั้งประหยัดพื้นที่ในการติดตั้ง ฉะนั้น เทอร์มินอลบล็อกจึงเหมาะกับการใช้งานภายในอาคารและงานอุตสาหกรรม

2.6.1 ประเภทของ Terminal Blocks มีรูปแบบที่แตกต่างกันออกไป ประเภทการเข้าสายของ Terminal Blocks มีทั้งหมด 3 ประเภท ได้แก่

1) การเข้าสายแบบสกรู (Screw Type) นิยมมาก เพราะเป็นการเข้าสายที่เหมาะสมกับงานทั่วไป ตั้งแต่สายขนาดเล็กไปจนถึงสายเบอร์ใหญ่ ส่วนวิธีการในการเข้าสายก็คือ ยึดสายด้วยการบีบอัดของตัวน็อต

2) การเข้าสายแบบสปริง (Spring Type) เป็นการเข้าสายที่ง่าย และประหยัดเวลา เหมาะสำหรับงานที่มีการสั่นสะเทือนของเครื่องจักร วิธีการเข้าสายก็คือใช้สปริงเป็นตัวล็อคสาย

3) การเข้าสายแบบพุชอิน (Push In Type) เป็นการเข้าสายที่ง่าย และไม่จำเป็นต้องใช้เครื่องมือ ไม่ต้องใช้ความชำนาญ อีกทั้งช่วยลดความผิดพลาดในการเข้าสาย เหมาะกับงานที่มีการสั่นสะเทือนเช่นเดียวกัน



รูปภาพที่ 2.29 Terminal Block ทั้ง 3 แบบ

2.7 Switch (ที่มา : <https://bgrimmtrading.com/switchs-and-socket-siemens/>)

สวิตช์ คือ อุปกรณ์ที่ทำหน้าที่ควบคุมการไหลของกระแสไฟฟ้าภายในวงจรหรือกล่าวง่ายๆ คือ อุปกรณ์เปิด ปิดกระแสไฟฟ้าภายในวงจรไฟฟ้า โดยใช้สัญลักษณ์ดังรูป



รูปภาพที่ 2.30 สัญลักษณ์ของสวิตช์

2.7.1 การทำงานของสวิตช์

ส่วนประกอบพื้นฐานของสวิตช์จะมีส่วนที่เรียกว่า หน้าสัมผัส อยู่ภายในซึ่งคล้ายกับสะพาน เชื่อมให้กระแสไฟฟ้าไหลในวงจรไฟฟ้าได้ สวิตช์ทำหน้าที่เปิด ปิด วงจรไฟฟ้า ทำให้วงจรไฟฟ้าเกิดการ ทำงานอยู่ 2 ลักษณะคือ วงจรเปิดและวงจรปิด วงจรเปิด คือลักษณะที่หน้าสัมผัสของสวิตช์ไม่ เชื่อมต่อกันทำให้กระแสไฟฟ้าไม่สามารถไหลไปในวงจรได้ และวงจรปิด คือ การที่หน้าสัมผัสของ สวิตช์เชื่อมต่อกันทำให้กระแสไฟฟ้าไหลในวงจรได้

2.7.2 ประเภทของสวิตช์

1) สวิตช์เลื่อน เป็นสวิตช์ชนิดหนึ่งที่ใช้เปิด ปิด การทำงานของอุปกรณ์ ใช้งานโดยการเลื่อน การควบคุมตัดต่อสวิตช์ ทำได้โดยผลักเลื่อนสวิตช์ขึ้นบนหรือลงล่าง การเลื่อนสวิตช์ขึ้นบนเป็นการ ต่อ (ON) การเลื่อนสวิตช์ลงล่างเป็นการตัด (OFF) นิยมใช้เป็นอุปกรณ์เปิด ปิด สิ่งของประเภทของ เล่นเด็ก และเครื่องใช้ต่างๆ เช่น นาฬิกาปลุก ไฟฉาย

2) สวิตช์กระดก เป็นสวิตช์ที่ใช้งานโดยการกด เมื่อต้องการเปิดสวิตช์ก็ให้กดด้านที่ระบุว่าเป็น การเปิดสวิตช์ลง ส่วนอีกด้านที่เหลือกก็จะกระดกขึ้น โดยส่วนใหญ่แล้วจะมีตัวอักษรระบุการ ทำงานบนตัวสวิตช์ เช่น เปิด ปิด On-OFF เราจะพบเห็นการใช้สวิตช์กระดกนี้กับหลอดไฟ ปลั๊กราง หรือเครื่องใช้ไฟฟ้าต่างๆ

3) สวิตช์กด ใช้งานโดยการกดเปิด ปิด ในปุ่มเดียวกัน คือ กดปุ่มที่อยู่ส่วนกลางสวิตช์ กด ปุ่มสวิตช์หนึ่งครั้งสวิตช์ต่อ (ON) และเมื่อกดปุ่มสวิตช์อีกครั้งสวิตช์ตัด (OFF) การทำงานเป็น เช่นนี้ตลอดเวลา แต่สวิตช์แบบกดบางแบบอาจเป็นชนิดกดติดปล่อยดับ (Momentary) คือขณะกด ปุ่มสวิตช์เป็นการต่อ (ON) เมื่อปล่อยมือออกจากปุ่มสวิตช์เป็นการตัด (OFF) ทันที เช่น ปุ่มปิด เปิด โทรทัศน์ รีโมท คอมพิวเตอร์

4) สวิตช์แบบก้านยาว (Toggle Switch) เป็นสวิตช์ที่เวลาใช้งานต้องโยกก้านสวิตช์ไปมา โดยมีก้านสวิตช์โยกยื่นยาวออกมาจากตัวสวิตช์ การควบคุมตัดต่อสวิตช์ ทำได้โดยโยกก้านสวิตช์ให้

ขึ้นบนหรือลงล่าง ในการโยกก้านสวิตช์ขึ้นมักจะเป็นการต่อ (ON) และโยกก้านสวิตช์ลงมักจะเป็นการตัด (OFF)

5) สวิตช์แบบหมุน (Rotary Switch) หรือเรียกว่าสวิตช์แบบเลือกค่า (Selector Switch) เป็นสวิตช์ที่ต้องหมุนก้านสวิตช์ไปโดยรอบเป็นวงกลม สามารถเลือกตำแหน่งการติดต่อได้หลายตำแหน่ง มีหน้าสัมผัสสวิตช์ให้เลือกต่อมากหลายตำแหน่ง เช่น 2, 3, 4 หรือ 5 ตำแหน่ง เป็นต้น

6) สวิตช์แบบไมโคร (Microswitch) คือสวิตช์แบบกดชนิดกดติดปล่อยดับนั่นเอง แต่เป็นสวิตช์ที่สามารถใช้แรงจำนวนน้อยๆ กดปุ่มสวิตช์ได้ ก้านสวิตช์แบบไมโครสวิตช์มีด้วยกันหลายแบบ อาจเป็นปุ่มกดเฉยๆ หรืออาจมีก้านแบบโยกได้มากกดปุ่มสวิตช์อีกทีหนึ่ง การควบคุมติดต่อสวิตช์ ทำได้โดยกดปุ่มสวิตช์หรือกดก้านคันโยกเป็นการต่อ (ON) และเมื่อปล่อยมือออกจากปุ่มหรือก้านคันโยกเป็นการตัด (OFF)

7) สวิตช์แบบดิฟ (DIP Switch) คำว่าดิฟ (DIP) มาจากคำเต็มว่าดูอัลอินไลน์แพ็คเกจ (Dual Inline Package) เป็นสวิตช์ขนาดเล็กใช้งานร่วมกับวงจรอิเล็กทรอนิกส์ที่สร้างขึ้นในรูปชิพ (Chip) ที่มีขนาดเล็กๆ หรือใช้งานกับไอซี (IC = Integrated Circuit) ลักษณะสวิตช์สามารถตัดหรือต่อวงจรได้ การควบคุมติดต่อสวิตช์แบบดิฟจะต้องใช้ปลายมปากกาหรือปลายดินสอในการปรับเลื่อนสวิตช์ สวิตช์แบบดิฟมักถูกติดตั้งบนแผ่นวงจรพิมพ์ (Printed Circuit Board) ใช้กับกระแสไม่เกิน 30mA ที่แรงดัน 30VD

2.8 LED (ที่มา : <https://www.ju-led.com/16574233/led>)



รูปภาพที่ 2.31 LED

LED คือไดโอดเปล่งแสง ย่อมาจากคำว่า (Light-Emitting Diode) ซึ่งสามารถเปล่งแสงออกมาได้แสงที่เปล่งออกมาประกอบด้วยคลื่นความถี่เดียวและเฟสต่อเนื่องกัน ซึ่งต่างกับแสงธรรมดาที่ตาคนมองเห็น โดย หลอด LED สามารถเปล่งแสงได้เมื่อจ่ายกระแสไฟฟ้าเข้าเพียงเล็กน้อยเท่านั้น และประสิทธิภาพในการให้แสงสว่างก็ยิ่งดีกว่าหลอดไฟขนาดเล็กทั่วไป

2.8.1 ข้อดีของแอลอีดี

- 1) ประสิทธิภาพในการให้แสงสว่างดีกว่าหลอดไฟธรรมดาทั่วไป
- 2) ตัวหลอด LED เองเมื่อทำให้เกิดแสงขึ้นจะกินกระแสน้อยมากประมาณ 1-20mA
- 3) มีอายุการใช้งานที่ยาวนาน ประมาณ 50,000 – 100,000 ชั่วโมง ขึ้นอยู่กับคุณภาพของแอลอีดี วงจรขับเคลื่อน สภาพภูมิอากาศ ความชื้น และอุณหภูมิ ซึ่งก็มีอายุการใช้งานที่ยาวนานกว่าหลอดที่ให้แสงสว่างชนิดอื่นๆมาก
- 4) ไม่มีรังสีอินฟราเรด รังสีอัลตราไวโอเลต ซึ่งเป็นอันตรายต่อผิวหนังและดวงตา
- 5) ทนทานต่อสภาวะอากาศ
- 6) ทนทานต่อการสั่นสะเทือน
- 7) มีหลากหลายสีให้เลือกใช้
- 8) เป็นมิตรกับสิ่งแวดล้อม เช่น หลอดสามารถถอดเปลี่ยน ซ่อมได้เฉพาะจุด ที่เสียไม่จำเป็นต้องทิ้งทั้งชุดเหมือนหลอดไฟทั่วไป ไม่เพิ่มปริมาณขยะ
- 9) หลอด LED สามารถเปิดใช้งานได้อย่างรวดเร็ว เปิดแล้วหลอดติดทันที ไม่ต้องรอกระพริบหรือวอร์มหลอดก่อน
- 10) สีสั่นจัดจ้าน หลอด LED ให้สีสั่นจัดจ้านชัดเจนมากกว่าหลอดไส้ ถ่ายรูปออกมาสีสดใส
- 11) มีความทนทานสูง เพราะหลอด LED เป็นอุปกรณ์ Solid State ซึ่งไม่มีชิ้นส่วนใดที่เคลื่อนไหว ไม่มีชิ้นใดที่เป็นกระจกไม่มีไส้หลอดซึ่งอาจจะขาดได้ง่าย

2.9 Switching (ที่มา : <http://www.siam-automation.com/>)

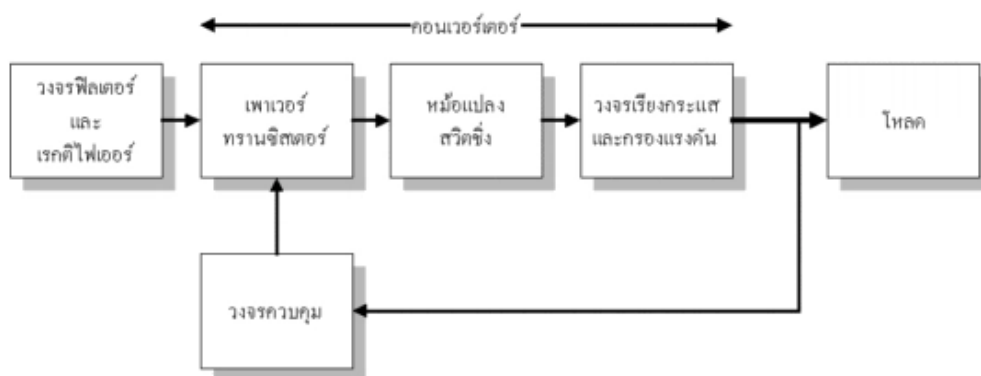


รูปภาพที่ 2.32 switching power supply

switching คือ สวิตชิงเพาเวอร์ซัพพลาย (Switching Power Supply) เป็นแหล่งจ่ายไฟตรงคงค่าแรงดันแบบหนึ่ง และสามารถเปลี่ยนไฟสลับแรงดันสูงให้เป็นไฟตรงแรงดันต่ำ เพื่อใช้ในงานอิเล็กทรอนิกส์ได้เช่นเดียวกันแหล่งจ่ายไฟเชิงเส้น (Linear Power Supply) ถึงแม้เพาเวอร์ซัพพลายทั้งสองแบบจะต้องมีการใช้หม้อแปลงในการลดทอนแรงดันสูงให้เป็นแรงดันต่ำเช่นเดียวกัน

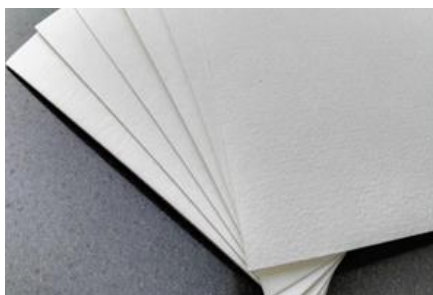
2.9.1 หลักการทำงานของสวิตชิงเพาเวอร์ซัพพลาย

สวิตชิงเพาเวอร์ซัพพลาย โดยทั่วไปมีองค์ประกอบพื้นฐานที่คล้ายคลึงกัน และไม่ซับซ้อนมากนัก หัวใจสำคัญของสวิตชิงเพาเวอร์ซัพพลาย จะอยู่ที่คอนเวอร์เตอร์ เนื่องจากทำหน้าที่ทั้งลดทอนแรงดันและคงค่าแรงดันเอาต์พุตด้วย องค์ประกอบต่างๆ ทำงานตามลำดับ ดังนี้



รูปภาพที่ 2.33 หลักการทำงานของสวิตชิงเพาเวอร์ซัพพลาย

2.10 กระดาษชานอ้อย (ที่มา : <https://iloftprint.com/architectural-models-materials/>)



รูปภาพที่ 2.34 กระดาษชานอ้อย

กระดาษชานอ้อย คือ กระดาษที่นำเยื่อกระดาษจากชานอ้อยที่เหลือทิ้งจากอุตสาหกรรมผลิตน้ำตาล คัดแยกเยื่อที่สะอาดนำไปผ่านกระบวนการผลิตและขึ้นรูปเป็นแผ่น นิยมนำไปทำผลิตภัณฑ์ประเภท กระดาษรองแก้ว จาน ชาม ถ้วยบรรจุอาหาร และเป็นที่นิยมนำมาทำเป็น วัสดุโมเดล สถาปัตยกรรม ตัดง่ายด้วยคัตเตอร์ และ เครื่องเลเซอร์ Co2 วัสดุความทนทานต่อความร้อน และเย็นจัด สามารถนำไปเข้าเตาอบ และแช่เย็นได้ ย่อยสลายได้เร็ว เป็นมิตรต่อสิ่งแวดล้อม

2.11 NFC (ที่มา : <https://www.cybertice.com/>)



รูปภาพที่ 2.35 การใช้ NFC

Near Field Communication (NFC) คือเทคโนโลยีสื่อสารไร้สายระยะสั้น ถูกพัฒนาขึ้นมาโดย Sony และ NXP โดยใช้คลื่นวิทยุความถี่ 13.56 MHz เพื่อให้อุปกรณ์จำพวกโทรศัพท์มือถือหรืออุปกรณ์พกพาประเภทอื่นๆสามารถสื่อสารระหว่างกันได้ง่ายขึ้น เพียงแต่นำอุปกรณ์ทั้งสองเครื่องมาวางชิดกันหรือแตะกันเท่านั้น เพื่อที่ทำการส่งข้อมูลหากัน



รูปภาพที่ 2.36 ประโยชน์ของ NFC

การส่งข้อมูลผ่านเทคโนโลยี Near Field Communication ต้องใช้อุปกรณ์ที่เปิดใช้งานเอ็นเอฟซี สองเครื่อง อุปกรณ์ส่งสัญญาณและอุปกรณ์รับสัญญาณ ตัวอย่างเช่นแท็กเอ็นเอฟซี/สติ๊กเกอร์และสมาร์ทโฟนในกรณีของแท็กเอ็นเอฟซีและสมาร์ทโฟนแท็กเอ็นเอฟซีเป็นอุปกรณ์แบบพาสซีฟซึ่งสามารถส่งข้อมูลได้ แต่ไม่สามารถรับได้ในขณะที่สมาร์ทโฟนเป็นอุปกรณ์เอ็นเอฟซีที่ใช้งานอยู่ซึ่งสามารถรับสัญญาณจากแท็กเอ็นเอฟซีและส่งสัญญาณข้อมูลไปยังสมาร์ทโฟนที่เปิดใช้งานเอ็นเอฟซีอื่น ในขณะที่อุปกรณ์บลูทูธทำงานเหมือนกันสิ่งที่ทำให้เอ็นเอฟซีมีเอกลักษณ์ คือความจริงที่ว่ามันไม่จำเป็นต้องมีการจับคู่อุปกรณ์และการอนุญาตให้ส่งข้อมูล ผู้ใช้ทุกคนต้องทำเพื่อรับข้อมูลหรือชำระเงินผ่านเทคโนโลยีเอ็นเอฟซี คือการแตะอุปกรณ์ส่งสัญญาณด้วยสมาร์ทโฟนที่รองรับเอ็นเอฟซี นี่คือ 2 วิธีที่การสื่อสารกับเทคโนโลยี Near Field Communication ทำงาน

2.11.1 การสื่อสารทางเดียว ต้องใช้อุปกรณ์ เอ็นเอฟซี แฝงและอุปกรณ์ เอ็นเอฟซี ที่ใช้งานอยู่ การสื่อสารระหว่างแท็กเอ็นเอฟซีและสมาร์ทโฟนที่เปิดใช้งานเอ็นเอฟซีบัตรชำระเงินเอ็นเอฟซีและเครื่องอ่านการ์ดเป็นตัวอย่างของการสื่อสารทางเดียว

2.11.2 การสื่อสารสองทาง ต้องใช้อุปกรณ์เอ็นเอฟซีที่ใช้งานสองเครื่อง การสื่อสารระหว่างสมาร์ทโฟนที่ใช้เอ็นเอฟซีสองเครื่องสำหรับการถ่ายโอนไฟล์เป็นตัวอย่างของการสื่อสารสองทาง วิธีการต่ออุปกรณ์ สอนใช้งาน ESP32 RFID / NFC Module Kit (PN532)

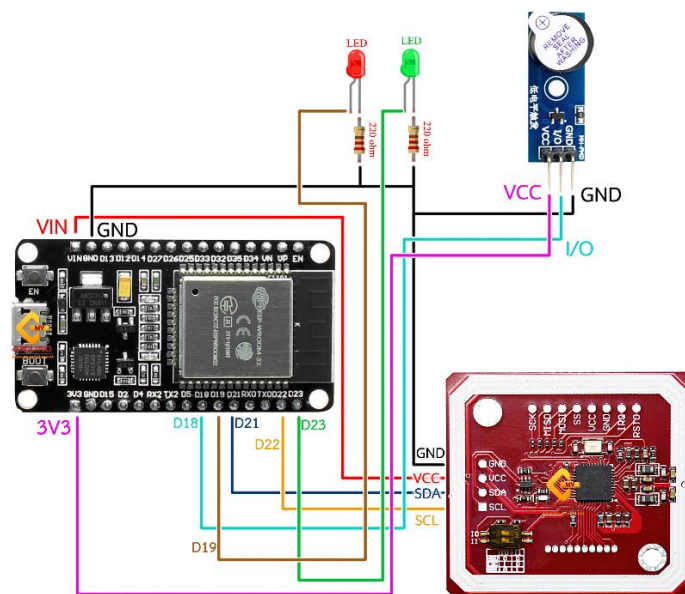
2.11.3 การเชื่อมต่ออุปกรณ์

ESP32 -> RFID / NFC Module Kit (PN532)

VIN -> VCC

GND -> GND

D22 -> SCL
 D21 -> SDA
 ESP32 -> LED
 D23 -> LED
 GND -> ตัวต้านทาน 220 ohm -> LED
 D19 -> LED
 GND -> ตัวต้านทาน 220 ohm -> LED
 ESP32 -> Active Buzzer Module 3.3 - 5V
 3V3 -> VCC
 GND -> GND
 D18 -> I/O



รูปภาพที่ 2.37 การต่อใช้งาน PN532 กับ ESP32

Code ตัวอย่าง

```

int LED_R = 19;
int LED_G = 23;
int buzzer = 18;
int val = 0;
# if 0

```

```

#include <SPI.h>
#include <PN532_SPI.h>
#include "PN532.h"
PN532_SPI pn532spi(SPI, 10);
PN532 nfc(pn532spi);
#elif 0
#include <PN532_HSU.h>
#include <PN532.h>
PN532_HSU pn532hsu(Serial1);
PN532 nfc(pn532hsu);
#else
#include <Wire.h>
#include <PN532_I2C.h>
#include <PN532.h>
#include <NfcAdapter.h>
PN532_I2C pn532i2c(Wire);
PN532 nfc(pn532i2c);
#endif

void setup(void) {
  pinMode(LED_R, OUTPUT);
  pinMode(LED_G, OUTPUT);
  pinMode(buzzer, OUTPUT);
  digitalWrite(buzzer, HIGH);
  Serial.begin(115200);
  Serial.println("Hello!");
  nfc.begin();
  uint32_t versiondata = nfc.getFirmwareVersion();
  if (! versiondata) {
    Serial.print("Didn't find PN53x board");
    while (1); // halt
  }
}

```

```

Serial.print("Found chip PN5");
Serial.println((versiondata >> 24) & 0xFF, HEX);
Serial.print("Firmware ver. ");
Serial.print((versiondata >> 16) & 0xFF, DEC);
Serial.print('.'); Serial.println((versiondata >> 8) & 0xFF, DEC);
nfc.setPassiveActivationRetries(0xFF);
nfc.SAMConfig();
Serial.println("Waiting for an ISO14443A card");
}

void loop(void) {
  boolean success;
  uint8_t uid[] = { 0, 0, 0, 0, 0, 0, 0, 0 };
  uint8_t uidLength;
  success = nfc.readPassiveTargetID(PN532_MIFARE_ISO14443A, &uid[0],
  &uidLength);
  if (success) {
    Serial.println("Found a card!");
    Serial.print("UID Length: ");
    Serial.print(uidLength, DEC);
    Serial.println(" bytes");
    Serial.print("UID Value: ");
    for (uint8_t i = 0; i < uidLength; i++)
    {
      Serial.print(" "); Serial.print(uid[i], DEC);
    }
    Serial.println("");
    delay(1000);
  }
  else
  {
    Serial.println("Timed out waiting for a card");
  }
}

```

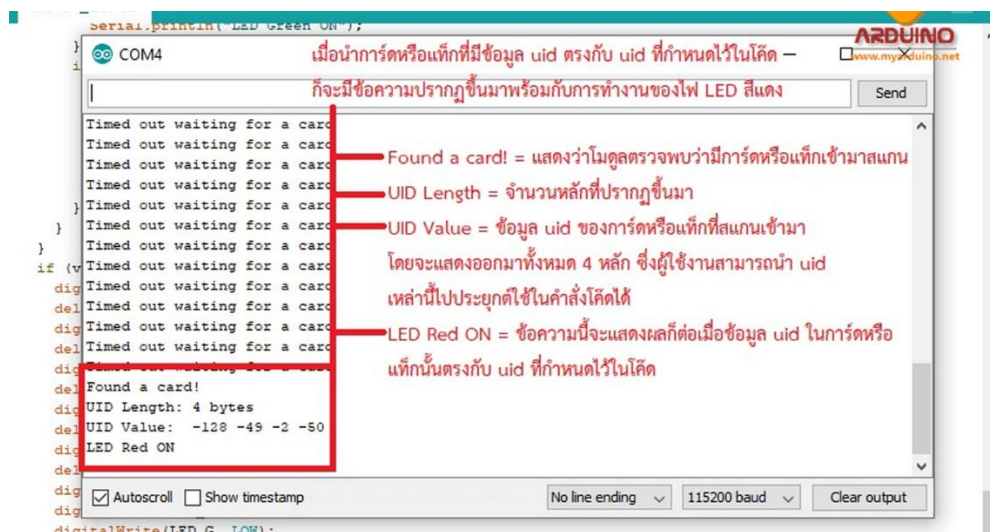
```

}
if (uid[0] == 128 || uid[1] == 49 || uid[2] == 2 || uid[3] == 50 )
{
  val = val + 1; {
  if (val == 1) {
    digitalWrite(buzzer, LOW);
    delay(100);
    digitalWrite(buzzer, HIGH);
    digitalWrite(LED_R, HIGH); /
    Serial.println("LED Red ON");
  }
  if (val == 2) {
    digitalWrite(buzzer, LOW);
    delay(100);
    digitalWrite(buzzer, HIGH);
    digitalWrite(LED_R, LOW); // สั้งปิดหลอดไฟ LED (LOW = ปิด)
    Serial.println("LED Red OFF");
    val = 0;
  }
}
}

if (uid[0] == 105 || uid[1] == 89 || uid[2] == 25 || uid[3] == 201 ) {
  val = val + 3; {
  if (val == 3) {
    digitalWrite(buzzer, LOW);
    delay(100);
    digitalWrite(buzzer, HIGH);
    digitalWrite(LED_G, HIGH);
    Serial.println("LED Green ON");
  }
  if (val == 6) {

```

```
digitalWrite(buzzer, LOW);
delay(100);
digitalWrite(buzzer, HIGH);
digitalWrite(LED_G, LOW);
Serial.println("LED Green OFF");
val = 0;
}
}
}
if (val >= 7) {
digitalWrite(buzzer, LOW);
delay(100);
digitalWrite(buzzer, HIGH);
delay(100);
digitalWrite(buzzer, LOW);
delay(100);
digitalWrite(buzzer, HIGH);
delay(100);
digitalWrite(buzzer, LOW);
delay(100);
digitalWrite(buzzer, HIGH);
digitalWrite(LED_R, LOW);
digitalWrite(LED_G, LOW);
Serial.println("All LED OFF");
val = 0;
}
}
```



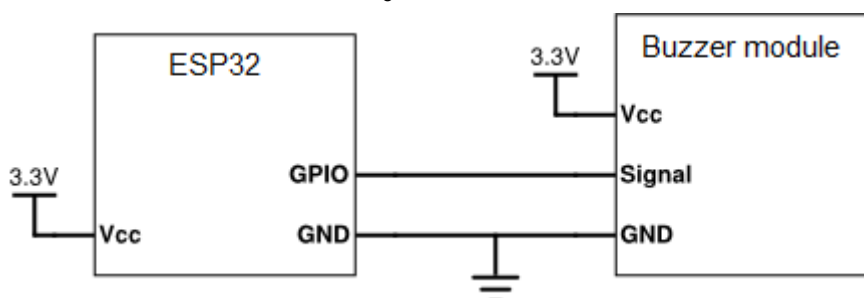
รูปภาพที่ 2.38 ผลลัพธ์ของโปรแกรม

2.12 Buzzer (ที่มา : <https://www.mindphp.com/>)



รูปภาพที่ 2.39 Buzzer

Buzzer คือ ลำโพงแบบแม่เหล็กหรือ แบบเพียโซที่มีวงจรกำเนิดความถี่ (oscillator) อยู่ในตัว ใช้ไฟเลี้ยง 3.3 - 5V สามารถสร้างเสียงเตือนหรือส่งสัญญาณที่เป็นรูปแบบต่างๆ เราอาจจะเคยได้ยินเสียงบัสเซอร์อยู่บ่อยๆ เช่น เสียง ป๊อปที่อยู่ในคอมพิวเตอร์ก็ใช้บัสเซอร์ในการส่งสัญญาณให้ทราบสถานะของคอมพิวเตอร์ให้ทราบว่ามีปัญหาอะไร



รูปภาพที่ 2.40 การต่อใช้งาน Buzzer กับ ESP32

Code ตัวอย่าง

```
const int buzzer = 9;

void setup() {
  pinMode (buzzer, OUTPUT);}

void loop() {
  tone (buzzer, 50);
  delay (1000);
  noTone(buzzer);
  delay (1000);}
```

2.13 จอ LCD (ที่มา : <https://www.mindphp.com/>)



รูปภาพที่ 2.41 จอ LCD แบบ 16x2

2.13.1 วิธีการต่ออุปกรณ์ สอนใช้งาน Arduino 1602 LCD โมดูลจอแสดงผล LCD พร้อม I2C Interface

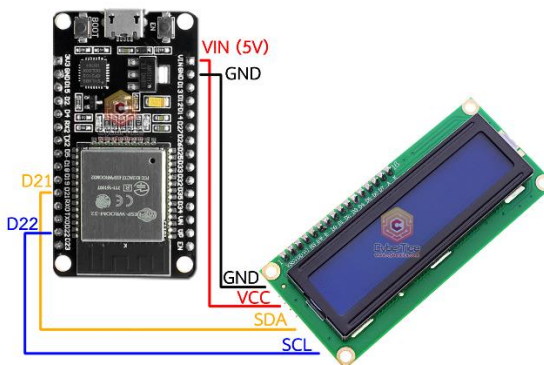
1) เชื่อมต่ออุปกรณ์ESP32ต่อกับจอแสดงผล LCD

VIN (5V) -> VCC

GND -> GND

D21 -> SDA

D22 -> SCL



รูปภาพที่ 2.42 การต่อจอ LCD กับ ESP32

2) ดาวน์โหลด Library LiquidCrystal_I2C

3) ทำการแยกไฟล์และ Copy โฟลเดอร์ไปไว้ในโฟลเดอร์ libraries ของโปรแกรม

Arduino IDE

4) เปิดโปรแกรม Arduino IDE

5) พิมพ์ Code ตัวอย่าง ดังนี้

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
void setup()
{
  lcd.begin();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Welcome To");
  lcd.setCursor(7, 1);
  lcd.print("WORLD");
}
void loop() {
}
```

6) อัปโหลดโปรแกรม



รูปภาพที่ 2.43 ผลลัพธ์การต่อจอ LCD กับ ESP32

2.14 สมองกลฝังตัว (ที่มา : <https://microbrain.io/>)

ระบบฝังตัว หรือ สมองกลฝังตัว (embedded system) คือระบบประมวลผล ที่ใช้ชิปหรือไมโคร โปรเซสเซอร์ที่ออกแบบมาโดยเฉพาะ เป็นระบบคอมพิวเตอร์ขนาดเล็กที่ฝังไว้ในอุปกรณ์เครื่องใช้ไฟฟ้า และเครื่องเล่นอิเล็กทรอนิกส์ต่างๆ เพื่อเพิ่มความฉลาด ความสามารถให้กับอุปกรณ์เหล่านั้นผ่าน ซอฟต์แวร์ซึ่งต่างจากระบบประมวลผลที่เครื่องคอมพิวเตอร์ทั่วไป ระบบฝังตัวถูกนำมาใช้กันอย่างแพร่หลายในยานพาหนะ เครื่องใช้ไฟฟ้าในบ้านและสำนักงาน อุปกรณ์อิเล็กทรอนิกส์ เทคโนโลยี ซอฟต์แวร์ เทคโนโลยีฮาร์ดแวร์ เทคโนโลยีเครือข่ายเน็ตเวิร์ค เทคโนโลยีด้านการสื่อสาร เทคโนโลยี เครื่องกลและของเล่นต่างๆ คำว่าระบบฝังตัวเกิดจากการที่ระบบนี้เป็นระบบประมวลผลเช่นเดียวกับ ระบบคอมพิวเตอร์ แต่ว่าระบบนี้จะฝังตัวลงในอุปกรณ์อื่น ๆ ที่ไม่ใช่เครื่องคอมพิวเตอร์ ในปัจจุบัน ระบบสมองกลฝังตัวได้มีการพัฒนามากขึ้น โดยในระบบสมองกลฝังตัวอาจจะประกอบไปด้วย ไมโครคอนโทรลเลอร์ หรือ ไมโครโปรเซสเซอร์ อุปกรณ์ที่ใช้ระบบสมองกลฝังตัวที่เห็นได้ชัดเช่น โทรศัพท์มือถือ และในระบบสมองกลฝังตัวยังมีการใส่ระบบปฏิบัติการต่างๆ แตกต่างกันไปอีกด้วย ดังนั้น ระบบสมองกลฝังตัวอาจจะทำงานได้ตั้งแต่ควบคุมหลอดไฟจนไปถึงใช้ในยานอวกาศ

2.14.1 กระบวนการเชิงระบบ (Input – Process – Output)

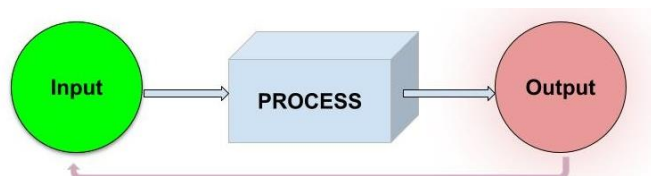


รูปภาพที่ 2.44 แผนภาพการทำงานของกระบวนการเชิงระบบ

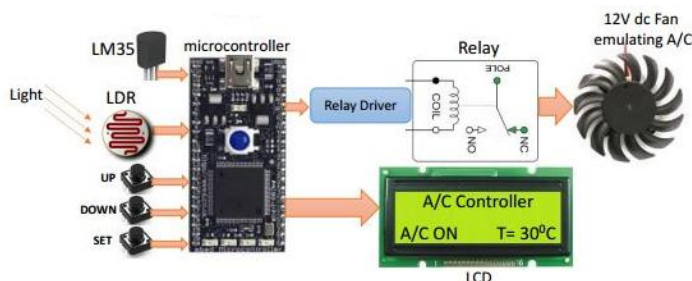
1) สิ่งที่ป้อนเข้าไป (Input) หมายถึง สิ่งต่างๆ ที่จำเป็นต้องใช้ในกระบวนการหรือโครงการต่างๆ เช่น ในระบบการเรียนการสอนในชั้นเรียน อาจได้แก่ ครู นักเรียน ชั้นเรียน หลักสูตร ตารางสอน วิธีการสอน เป็นต้น ถ้าในเรื่องระบบหายใจ อาจได้แก่ จมูก ปอด กระบังลม อากาศ เป็นต้น

2) กระบวนการหรือการดำเนินงาน (Process) หมายถึง การนำเอาสิ่งที่ป้อนเข้าไป มาจัดกระทำให้เกิดผลบรรลุตามวัตถุประสงค์ที่ต้องการ เช่น การสอนของครู หรือการให้นักเรียนทำกิจกรรม เป็นต้น

3) ผลผลิต หรือการประเมินผล (Output) หมายถึง ผลที่ได้จากการกระทำในขั้นที่สอง ได้แก่ ผลสัมฤทธิ์ทางการเรียนของนักเรียน หรือผลงานของนักเรียน เป็นต้น บางระบบ เรานำ Output มาปรับปรุง แล้วย้อนกลับประกอบเป็น Input เพื่อให้ได้ Output ใหม่ ที่ใกล้เคียงความต้องการมากขึ้น



รูปภาพที่ 2.45 แผนภาพการนำ Output มาเป็น Input



รูปภาพที่ 2.46 ตัวอย่างการใช้งานระบบสมองกลฝังตัว

2.15 เอกสารงานวิจัยที่เกี่ยวข้อง

พรรณเชษฐ์ ณ ลำพูน (2560 : บทคัดย่อ) ในธุรกิจการบริการลูกค้าเป็นกุญแจสำคัญที่ส่งผลต่อกำไรในการทำธุรกิจ ซึ่งธุรกิจแต่ละประเภทย่อมมีข้อจำกัดที่แตกต่างกันไม่ว่าจะเป็นเรื่องพื้นที่ที่จำกัดของทางร้าน ทุนในการทำธุรกิจ หรือแม้แต่จำนวนพนักงานที่ให้บริการลูกค้า เป็นต้น ด้วยข้อจำกัดเหล่านี้ทำให้ธุรกิจต่างๆ พยายามหาแนวทางในการพัฒนาแก้ไขปัญหา เพื่อก้าวข้ามขีดจำกัดของตัวเอง และสามารถแข่งขันกับธุรกิจอื่นได้ อีกทั้งเทคโนโลยีมีความหลากหลายทั้งในเรื่องรูปแบบและราคา ซึ่งราคาในการนำเทคโนโลยีมาใช้นั้นถือว่าเป็นปัจจัยที่สำคัญอีกปัจจัยหนึ่งเช่นกัน เทคโนโลยีการสื่อสารระยะใกล้ หรือ NFC จึงเป็นอีกแนวทางหนึ่งที่สามารถนำมาประยุกต์ใช้เพื่อเพิ่มประสิทธิภาพและประสิทธิผลในการบริการให้กับองค์กรธุรกิจและลูกค้า อีกทั้งตัวเทคโนโลยี NFC นั้นสามารถนำมาประยุกต์ใช้ได้หลากหลายรูปแบบ และมีราคาต่ำ ทำให้องค์กรธุรกิจสามารถประหยัดค่าใช้จ่ายในการนำเทคโนโลยีมาใช้

กมลมาลย์ เสวตวงษ์(2557 : บทคัดย่อ)การประยุกต์เทคโนโลยีสนามสื่อสารระยะใกล้ (Near Field Communication :NFC) และการพิมพ์รหัสคิวอาร์ (Quick Response Code : QR Code)ในการจัดทำสโมล์โปสเตอร์สำหรับแนะนำข้อมูลด้านบริการ ทรัพยากรห้องสมุด จุดบริการต่าง ๆ และเวลาบริการของห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น โดยพัฒนาแอปพลิเคชัน บนอุปกรณ์สมาร์ตโฟนที่ประยุกต์ใช้เทคโนโลยีเอ็นเอฟซี ข้อมูลของห้องสมุดที่ต้องการแสดงให้ผู้ใช้ได้ ทราบจะถูกทำให้เป็นลิงค์ (Link) และบรรจุลงในเอ็นเอฟซีแท็กก่อนที่จะฉีกแท็กนั้นลงบนแผ่น โปสเตอร์ และเมื่อผู้ใช้บริการต้องการทราบข้อมูลที่ต้องการก็สามารถนำสมาร์ตโฟนที่ได้รับการติดตั้ง ซอฟต์แวร์ที่สามารถอ่านและแปลงข้อมูลที่เก็บไว้ในเอ็นเอฟซีแท็ก (NFC Tag Reader) มาวางใกล้ๆ กับบริเวณที่ได้ฉีกเอ็นเอฟซีแท็กไว้ (Touch Point Indicator) รายละเอียดก็จะปรากฏบนหน้าจอ ของโทรศัพท์ซึ่งผู้ใช้จะได้รับทราบรายละเอียดของข้อมูลโดยทันที และเพื่อให้สมาร์ตโฟนสามารถรองรับสมาร์ตโฟนได้ทุกระบบ จึงมีการพิมพ์รหัสคิวอาร์ (QR Code) ลงบนแผ่นโปสเตอร์ ควบคู่กับเอ็นเอฟซีแท็ก ซึ่งผู้ใช้สามารถดาวน์โหลดโปรแกรมเพื่อสแกนอ่านข้อมูลได้เช่นเดียวกับ ระบบเอ็นเอฟซี

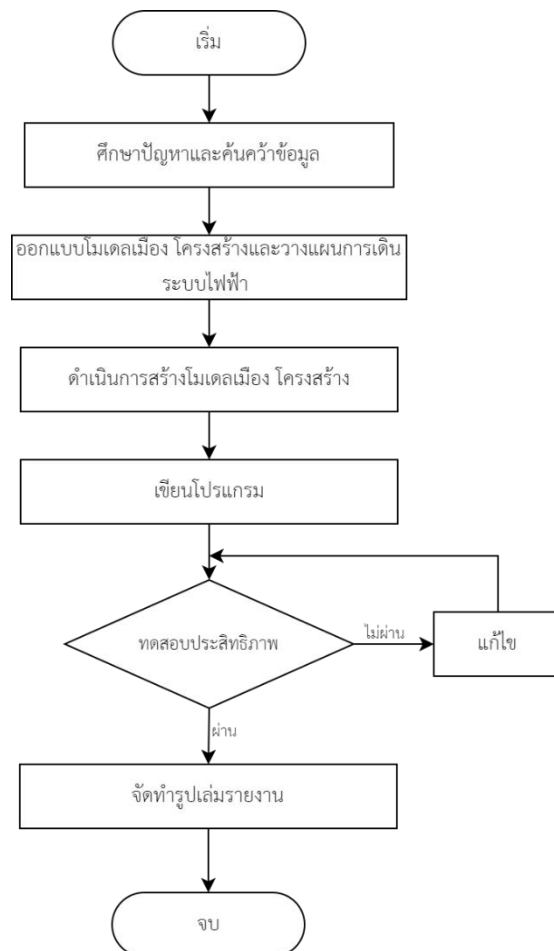
นิรันดร์ ทองอรุณ(2554 : บทคัดย่อ)จากอดีตสู่ปัจจุบัน แนวคิดในการออกแบบสถาปัตยกรรม ไม่ว่าจะเป็นเรื่องของการเกิดรูปทรงหรือที่ว่างทางสถาปัตยกรรม ล้วนมาจากปัจจัยด้านต่างๆ ที่เกี่ยวข้องกับสถาปัตยกรรมโดยตรง เช่น สถานที่ เวลา และบริบท ขณะเดียวกันกระแสการเปลี่ยนแปลง สู่ระบบบริโภคนิยมและกระแสโลกาภิวัตน์ รวมไปถึงบริบทแวดล้อมอื่นๆ ในปัจจุบัน ทำให้การดำรงอยู่ของสถาปัตยกรรมมิได้ มีหน้าที่เพียงเพื่อการใช้สอยหรืออาศัยอยู่ไปวันๆ นัก

ออกแบบได้พยายามสร้างบทบาทของสถาปัตยกรรมให้มีปฏิสัมพันธ์กับ ภูมิทัศน์ พื้นที่ว่าง รวมถึง กิจกรรม บริบทแวดล้อม และพฤติกรรมมนุษย์ให้มีมิติมากขึ้น บทบาทระหว่างสถาปัตยกรรมกับพื้นที่ ว่าง นำมาซึ่งแนวคิดในการออกแบบพื้นที่ว่าง และภูมิทัศน์ เช่น สวนสาธารณะ พื้นที่สีเขียวและพื้นที่ สำหรับกิจกรรมสันทนาการ นอกจากนั้น ที่ว่างอันรายรอบสถาปัตยกรรมหนึ่งๆ เริ่มมีความหมาย มากกว่าจะเป็นเพียงพื้นที่ว่าง ดังนั้นการออกแบบและคุณภาพในการปรับเปลี่ยนที่ว่าง กิจกรรม และ บริบทแวดล้อม จึงมีความเชื่อมโยงกับสถาปัตยกรรมในการมีปฏิสัมพันธ์กับพื้นที่ว่างและภูมิทัศน์ ไม่ เพียงแต่ในเชิงรูปธรรม(กายภาพ) แต่ยังรวมถึงความหมายในเชิงนามธรรมด้วย ซึ่งสะท้อนให้เห็นถึง ความสัมพันธ์ขององค์ประกอบต่างๆ ที่มีบทบาทส่งเสริมศักยภาพซึ่งกันและกัน ดังที่กล่าวมาแล้ว ข้างต้น เพราะสถาปัตยกรรมและที่ว่างมิใช่เพียงการแทนความหมายในเชิงสัญลักษณ์ แต่ยังรวมถึง ความหมาย ในเชิงข้อมูลข่าวสาร และสะท้อนถึงวิวัฒนาการของการปรับเปลี่ยนที่ว่างในการใช้งาน ตลอดจนกิจกรรมและพฤติกรรมต่างๆ ของมนุษย์ บทความนี้เป็นการรวบรวมและวิเคราะห์ประเด็น ต่างๆ ที่เกี่ยวกับแนวคิดและแนวทางการออกแบบสถาปัตยกรรม ภูมิทัศน์ และ พื้นที่ว่าง ให้ สอดคล้องสัมพันธ์กัน โดยเฉพาะคุณภาพของที่ว่างที่อยู่ระหว่างกลาง บริเวณรอยต่อระหว่าง สถาปัตยกรรมกับบริบท ที่สัมพันธ์กับกิจกรรม และพฤติกรรมมนุษย์ ซึ่งนำไปสู่แนวทางการพัฒนา รูปแบบและแนวคิดในการออกแบบสถาปัตยกรรม อย่างมีมิติต่อไป

บทที่ 3

วิธีการดำเนินโครงการ

ในการจัดทำโครงการ เมืองอัจฉริยะเอลิเซียม เป็นสิ่งประดิษฐ์ด้านนวัตกรรมซอฟต์แวร์และระบบสมองกลฝังตัว เพื่ออำนวยความสะดวกในการศึกษาไมโครคอนโทรลเลอร์ โดยคณะผู้จัดทำได้ดำเนินงานขึ้นตามขั้นตอนดังนี้

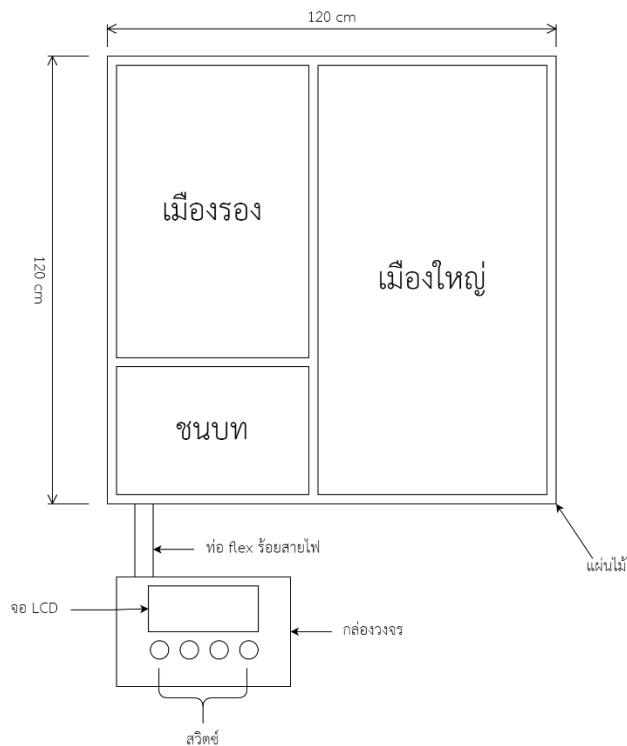


รูปภาพที่ 3.1 แผนผังแสดงการดำเนินโครงการ

3.1 ศึกษาปัญหาและค้นคว้าข้อมูล

ดำเนินการการศึกษาปัญหา และสิ่งที่จะแก้ไขโดยปัญหาที่พบเห็นในชีวิตประจำวัน หรือศึกษาจากปัญหาที่เกิดขึ้นอยู่ที่สามารถพบเห็น ณ ปัจจุบันประชาชนต้องการสิ่งอำนวยความสะดวกมากขึ้น ระบบ IOT จึงเข้ามาช่วยแก้ปัญหาให้แก่ประชาชนได้และเพื่อเป็นต้นแบบของการออกแบบเมืองอัจฉริยะทางผู้จัดทำจึงได้ทำการค้นคว้าหาข้อมูลต่าง ๆ ที่เกี่ยวข้องกับการทำโครงการในครั้งนี้ เพื่อนำมาใช้ในการจัดทำ เมืองอัจฉริยะ (Elysium)

3.2 ออกแบบโมเดลเมือง โครงสร้างและวางแผนการเดินระบบไฟฟ้า



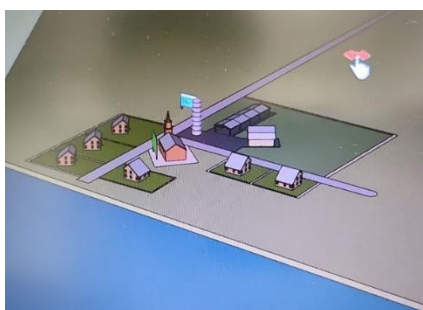
รูปภาพที่ 3.2 แบบโครงสร้างเมืองอัจฉริยะ



รูปภาพที่ 3.3 ออกแบบเมืองที่ 1



รูปภาพที่ 3.4 ออกแบบเมืองที่ 2



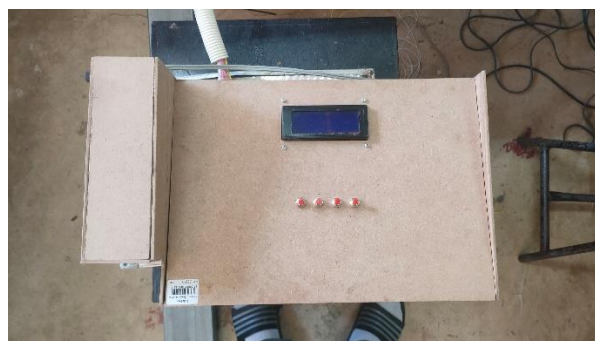
รูปภาพที่ 3.5 ออกแบบเมืองที่ 3



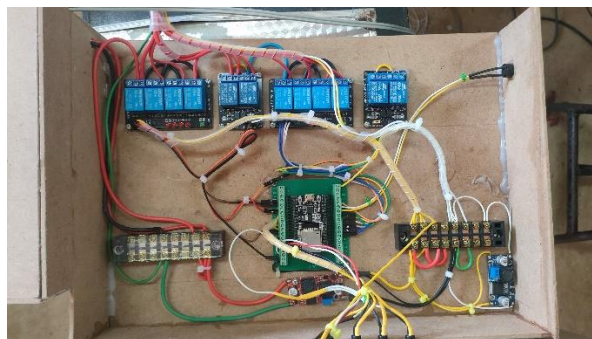
รูปภาพที่ 3.8 ตัดโมเดลเมืองที่ 2 และเมืองที่ 3



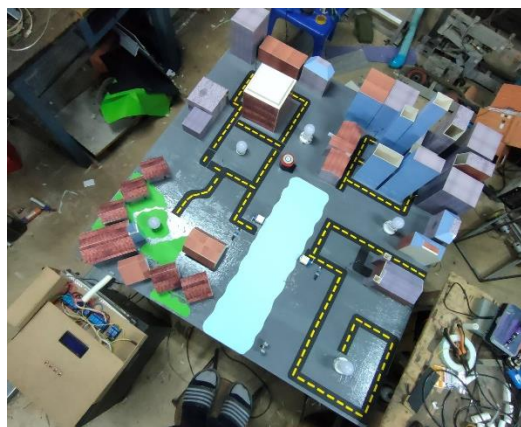
รูปภาพที่ 3.9 ฐานไม้



รูปภาพที่ 3.10 ด้านหน้ากล่องวงจร



รูปภาพที่ 3.11 ภายในกล่องวงจร



รูปภาพที่ 3.12 โมเดลเมืองอัจฉริยะ

3.4 เขียนโปรแกรม



รูปภาพที่ 3.13 เขียนคำสั่งในโปรแกรม Arduino ลงบอร์ด ESP-32

```

1 // btn 18 5 17 16 left to right
2 #include "WiFi.h" //เรียกใช้งานไลบรารี wifi
3 #include <WiFiUDP.h> //เรียกใช้งานไลบรารี UDP
4 #include <Wire.h>
5 #include <LiquidCrystal_I2C.h>
6 const char *ssid = "Freewifi"; //ชื่อ wifi ที่ต้องการเชื่อมต่อ
7 const char *password = "123456789"; //password ที่ต้องการเชื่อมต่อ
8 unsigned int UDPPort = 2390; // เปิดใช้งานพอร์ต 2390
9 int Status_LED = 2;
10 int packetSize;
11
12 String BufferString;
13 int len;
14 String data;
15 WiFiUDP Udp; //เปิดใช้ฟังก์ชัน UDP
16
17 void automode();
18 void twotown();
19 void smalltown();
20 void bigtown();
21 int wallmode = 0;
22 int kj=0;
23 int i=1 ;//loop pointer
24 int j=1;//loop pointer
25 int k1=0; //loop pointer
26 int k2=0;
27 int k3=0;
28 int k4=0;
29 int point=0; //pointer at home screen
30 int screen=0;
31 LiquidCrystal_I2C lcd(0x27, 20, 4); //address lcd
32 int relaylist[] = {23,19,4,13,2,15,32,33,25,26,27,14};
33 int buzzer = 12;
34 void setup()
35 {
36   Serial.print("connecting to "); //แสดงข้อความ "connecting to"

```

รูปภาพที่ 3.14 ตัวอย่างโปรแกรมการทำงานของ เมืองอัจฉริยะเอลิเซียม

3.5 การประเมินประสิทธิภาพ

เมืองอัจฉริยะเอลิเซียมมีขั้นตอนการประเมินประสิทธิภาพเมืองอัจฉริยะเอลิเซียม ดังนี้
 แบบการประเมินประสิทธิภาพจากการทดลอง เมืองอัจฉริยะเอลิเซียมมีการทดลองดังนี้
 สำหรับการให้ความหมายของค่าที่ทดลองได้ ผู้วิจัยได้กำหนดเกณฑ์ที่ใช้ในการให้ความหมาย
 โดยมีผลการประเมินดังนี้

5 คือ ดีมาก	หมายถึง	บรรลุผลร้อยละ 90 ขึ้นไป
4 คือ ดี	หมายถึง	บรรลุผลร้อยละ 80 - 89
3 คือ ค่อนข้างดี	หมายถึง	บรรลุผลร้อยละ 70 - 79
2 คือ พอใช้	หมายถึง	บรรลุผลร้อยละ 60 - 69
1 คือ ปรับปรุง	หมายถึง	บรรลุผลร้อยละ 50 - 59

ผลที่ได้จากการประเมินของผู้ทดลองใช้นั้น จะนำหลักการทางสถิติเข้ามาช่วยในการสรุป
 การประเมินอุปกรณ์ที่ได้สร้างขึ้น โดยคำนวณหาค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐานของระดับความคิดใน
 แต่ละด้าน

3.6 การเก็บรวบรวมข้อมูลและการวิเคราะห์ข้อมูล

สูตรหาค่าเฉลี่ย

3.6.1 ค่าเฉลี่ย โดยใช้สูตรค่าเฉลี่ย

$$\bar{x} = \frac{\Sigma x}{n}$$

เมื่อ $\bar{x}$	แทน	ค่าคะแนนเฉลี่ย
Σx	แทน	ผลรวมของคะแนนทั้งหมด
n	แทน	จำนวนครั้งที่ทดสอบ

3.6.2 ส่วนเบี่ยงเบนค่ามาตรฐานโดยใช้สูตร

$$S.D. = \sqrt{\frac{n\Sigma x^2 - (\Sigma x)^2}{n(n-1)}}$$

เมื่อ $S.D.$	แทน	ส่วนเบี่ยงเบนค่ามาตรฐาน
x	แทน	คะแนนของแต่ละครั้ง
n	แทน	จำนวนครั้งที่ทดสอบ

บทที่ 4

ผลการวิเคราะห์ข้อมูล

ในการจัดทำโครงการและงานวิจัยนี้ คณะผู้จัดทำได้ทำการพัฒนาเมืองอัจฉริยะเอลิเซียม โดยมีการเก็บรวบรวมข้อมูลผลการทดลองดังนี้

4.1 ผลการทดลองและการแก้ไข

4.2 การหาประสิทธิภาพของเมืองอัจฉริยะเอลิเซียม

4.1 ผลการทดลองและการแก้ไข

ผลการเปิดใช้งานครั้งที่ 1

อาการ ไม่สามารถรับ-ส่งข้อมูลระหว่างบอร์ดได้

การดำเนินการแก้ไข ตั้งค่าหมายเลข IP Address แบบคงที่ให้ทั้ง 2 บอร์ด

ผลการเปิดใช้งานครั้งที่ 2

อาการ ค่าของเซนเซอร์ที่รับมาไม่ตรงกับความเป็นจริง

การดำเนินการแก้ไข เปลี่ยนวิธีการต่อสายสัญญาณให้มีความหนาแน่นมากขึ้น

ผลการเปิดใช้งานครั้งที่ 3

ผลลัพธ์สามารถทำงานได้ปกติ

ผลการเปิดใช้งานครั้งที่ 4

ผลลัพธ์สามารถทำงานได้ปกติ

ผลการเปิดใช้งานครั้งที่ 5

อาการ หลอดไฟติดเร็วเกินไป

การดำเนินการแก้ไข เพิ่มค่า analog ในเงื่อนไขของการเปิดไฟให้สูงขึ้น

ผลการเปิดใช้งานครั้งที่ 6

อาการ บอร์ด esp32 reboot ต่อเนื่อง

การดำเนินการแก้ไข จากการตรวจสอบพบว่า การตั้งค่า IP Address ของบอร์ด esp32 ภายในโปรแกรม เขียนไม่ถูกต้องเป็นเหตุให้โปรแกรมทำงานต่อไม่ได้จึงทำการเขียนไว้ด้านนอก void setup ของโปรแกรม

ผลการเปิดใช้งานครั้งที่ 7 ผลลัพธ์สามารถทำงานได้ปกติ

ผลการเปิดใช้งานครั้งที่ 8 ผลลัพธ์สามารถทำงานได้ปกติ

ผลการเปิดใช้งานครั้งที่ 9 ผลลัพธ์สามารถทำงานได้ปกติ
 ผลการเปิดใช้งานครั้งที่ 10 ผลลัพธ์สามารถทำงานได้ปกติ

4.2 หาประสิทธิภาพของเมืองอัจฉริยะเอลิเซีย

จากการศึกษาทดลองประสิทธิภาพของเมืองอัจฉริยะเอลิเซียจึงมีการบันทึกผลการทดลอง
 ในตารางหาประสิทธิภาพดังตารางที่ 4.1 ดังนี้

ตารางที่ 4.1 หาประสิทธิภาพของเมืองอัจฉริยะเอลิเซีย

แบบประเมินประสิทธิภาพ	การทดสอบครั้งที่										ร้อยละ
	1	2	3	4	5	6	7	8	9	10	
1. ด้านการทำงานระบบอัตโนมัติ											
1.1 การเปิด-ปิดไฟตามความสว่างของแสงอาทิตย์	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
1.2 การแจ้งเตือนเมื่อเกิดแรงสั่นสะเทือน	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	80
1.3 การเปิด-ปิดของservoเมื่อมีสิ่งของผ่าน	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	90
1.4 การเปิด-ปิดของservoเมื่อมีการสแกนบัตร	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	90
2. ด้านการทำงานระบบแมนนวล											
2.1 สามารถเปิดไฟเมืองใหญ่ได้	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
2.2 สามารถเปิดไฟเมืองกลางได้	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
2.3 สามารถเปิดไฟเมืองเล็กได้	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100
2.4 สามารถดูค่าเซนเซอร์ที่ส่งมาได้	✓	✓	✓	✗	✓	✓	✗	✓	✓	✓	80
รวมทั้งหมด (ร้อยละ)											92.5

จากตารางที่ 4.1 เป็นการประเมินประสิทธิภาพของระบบเมืองอัจฉริยะเอลิเซียมีค่าเฉลี่ย
 ดังนี้ ด้านที่ 1 ด้านการทำงานระบบอัตโนมัติสูงสุดอยู่ที่ $\bar{X} = 90\%$ S.D. = 0.30382 อยู่ในเกณฑ์
 ระดับที่ 5 คือ ดีมาก และด้านที่ 2 ค่าเฉลี่ยด้านการทำงานระบบแบบแมนนวลสูงสุดอยู่ที่ $\bar{X} = 95\%$
 S.D. = 0.22072 ซึ่งอยู่ในเกณฑ์ระดับที่ 5 คือ ดีมาก ประสิทธิภาพของระบบเมืองอัจฉริยะเอลิเซีย
 โดยรวมมีค่าเฉลี่ยอยู่ที่ร้อยละ $\bar{X} = 92.5$ S.D. = 0.26505 ซึ่งอยู่ในเกณฑ์ระดับที่ 5 คือ ดีมาก

บทที่ 5

สรุปผล ปัญหาอุปสรรค และข้อเสนอแนะ

5.1 สรุปผล

โครงการนี้มีวัตถุประสงค์เพื่อสร้างระบบสแกนบัตร NFC เพื่อสร้างระบบไฟฟ้าอัตโนมัติ และเพื่อหาประสิทธิภาพของเมืองอัจฉริยะเอลิเซียม

จากการประเมินประสิทธิภาพ จำนวน 8 รายการ รายการละ 10 ครั้ง รวมทั้งหมด 80 ครั้ง มีคุณภาพอยู่ในระดับดีมาก สามารถทำงานได้ตามขอบเขตที่กำหนดไว้

5.2 ปัญหาและอุปสรรค

5.2.1 ตัวอุปกรณ์ต้องการกระแสไฟเลี้ยงเป็นจำนวนมาก

5.2.2 ชิ้นงานมีขนาดใหญ่ ไม่สามารถเคลื่อนย้ายได้

5.3 แนวทางแก้ไข

5.3.1 ใช้ power supply ที่จ่ายไฟได้มากขึ้น

5.3.2 ออกแบบกล่องวงจรให้สามารถถอดเพื่อเคลื่อนย้ายได้

5.4 ข้อเสนอแนะ

5.4.1 เพิ่มจำนวน sensor ให้มากขึ้น

5.4.2 ควรทำฐานวางให้กล่องวงจร

บรรณานุกรม

cybertice. (2563). **สอนใช้งาน Arduino ESP32-CAM**. สืบค้น 8 ตุลาคม 2566 :

จาก www.cybertice.com/article

cybertice. (2562). **สอนใช้งาน ESP32 เริ่มใช้งานเบื้องต้น**. สืบค้น 15 ตุลาคม 2566 :

จาก www.cybertice.com/article/226

Medium. (2562). **เรียนรู้ Interrupts โดยใช้ ESP32**. สืบค้น 20 ตุลาคม 2566 :

จาก <https://medium.com/@lil.eng.cmu/>

สถาบันนวัตกรรมและยุทธมาภิบาลข้อมูล. (2564). **IOTคืออะไร**. สืบค้น 20 ตุลาคม 2566 :

จาก <https://digi.data.go.th/blog/what-is-iot-technology>

Tech Talk 2 Apply. (2562). **Esp32 ดีกว่า Esp8266**. สืบค้น 25 ตุลาคม 2566 :

จาก <https://techtalk2apply.com/what-is-esp32/>

OMEGA. (2564). **Ultrasonic Sensor**. สืบค้น 12 พฤศจิกายน 2566 :

จาก <https://www.omi.co.th/th/article/ultrasonic-sensor>

B.GRIMM. (2564). **สวิตช์ไฟ เต้ารับ**. สืบค้น 12 พฤศจิกายน 2566 :

จาก <https://bgrimmtrading.com/switchs-and-socket-siemens/>

J.U.intertrade. (2562). **Led คืออะไร**. สืบค้น 10 ธันวาคม 2566 :

จาก <https://www.ju-led.com/16574233/led>

Siam automation. (2563). **Power Supply Switching**. สืบค้น 10 ธันวาคม 2566 :

จาก <http://www.siam-automation.com/>

iLoftPrint. (2563). **วัสดุโมเดลสถาปัตยกรรม?**. สืบค้น 1 มกราคม 2567 :

จาก <https://iloftprint.com/architectural-models-materials/>

ภาคผนวก

ภาคผนวก ก.
แบบขออนุมัติโครงการ

ภาคผนวก ข.

ผลการประเมินประสิทธิภาพของระบบเมืองอัจฉริยะเอลิเซียม

ผลการทำและหาคุณภาพเมืองอัจฉริยะเอลิเซียม

หัวข้อหาคุณภาพของเมืองอัจฉริยะเอลิเซียม	$\bar{x}$	S.D.	ระดับประสิทธิภาพ
1. ด้านการทำงานระบบอัตโนมัติ			
1.1 การเปิด-ปิดไฟตามความสว่างของแสงอาทิตย์	100	0	ดีมาก
1.2 การแจ้งเตือนเมื่อเกิดแรงสั่นสะเทือน	80	0.42	ดี
1.3 การเปิด-ปิดของservoเมื่อมีสิ่งของผ่าน	90	0.32	ดีมาก
1.4 การเปิด-ปิดของservoเมื่อมีการสแกนบัตร	90	0.32	ดีมาก
2. ด้านการทำงานระบบแมนนวล			
2.1 สามารถเปิดไฟเมืองใหญ่ได้	100	0	ดีมาก
2.2 สามารถเปิดไฟเมืองกลางได้	100	0	ดีมาก
2.3 สามารถเปิดไฟเมืองเล็กได้	100	0	ดีมาก
2.4 สามารถดูค่าเซนเซอร์ที่ส่งมาได้	80	0.42	ดี
รวมทั้งหมด	92.5	0.27	ดีมาก

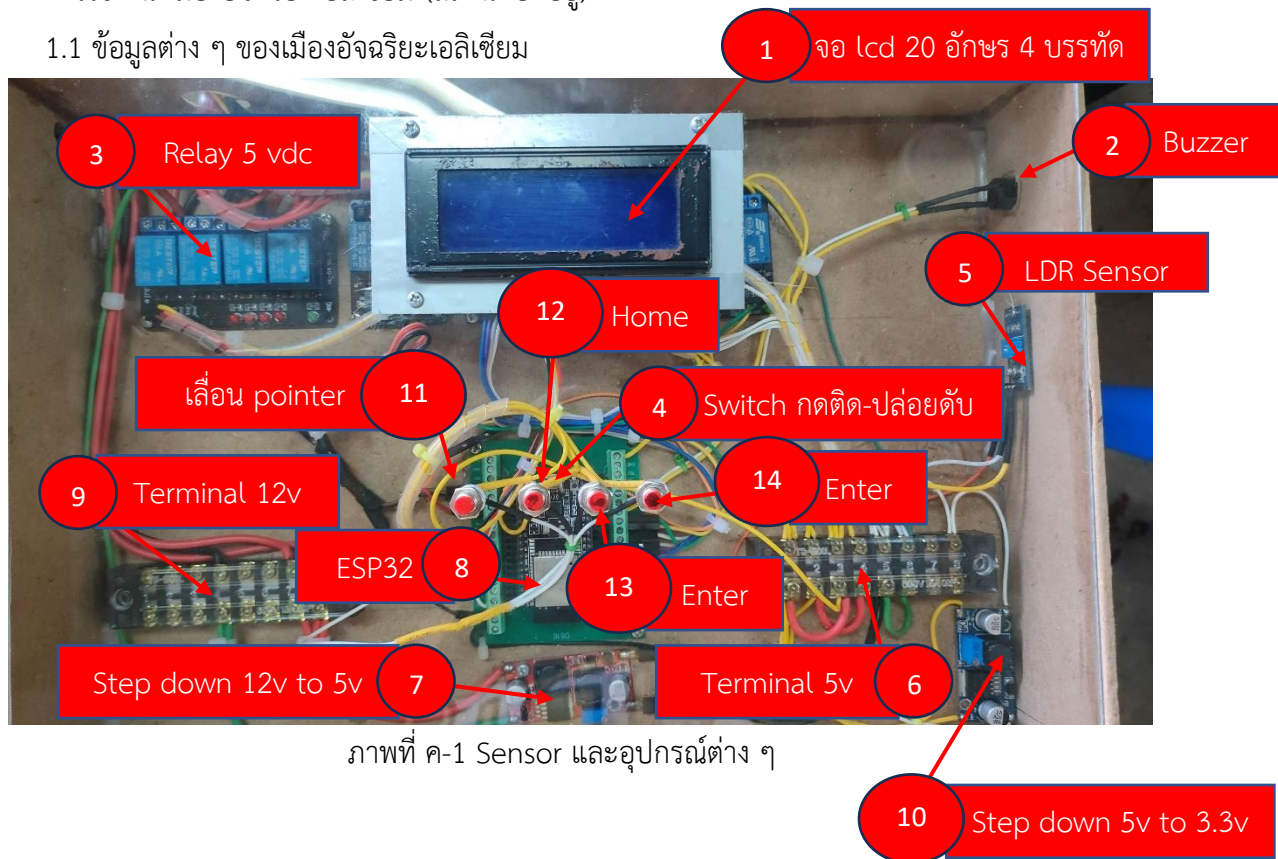
ภาคผนวก ค.

คู่มือการใช้งาน ระบบเมืองอัจฉริยะเอลิเซียม

คู่มือการใช้งาน เมืองอัจฉริยะเอลิเซียม

1. การใช้งาน เมืองอัจฉริยะเอลิเซียม (สถานะปิดอยู่)

1.1 ข้อมูลต่าง ๆ ของเมืองอัจฉริยะเอลิเซียม



ภาพที่ ค-1 Sensor และอุปกรณ์ต่าง ๆ

1.2 เสียบปลั๊ก และเปิด switch ไปที่ ON เพื่อเปิดใช้งาน เมืองอัจฉริยะเอลิเซียม



ภาพที่ ค-2 เปิดใช้งานเมืองอัจฉริยะเอลิเซียม

1.3 เมื่อบอร์ด Esp32 เชื่อมต่อ wifi สำเร็จ



ภาพที่ ค-3 หน้าเริ่มต้นการทำงาน

1.4 หน้า menu หลัก



ภาพที่ ค-4 หน้า menu หลัก

1.5 หน้า menu ใน Auto Mode

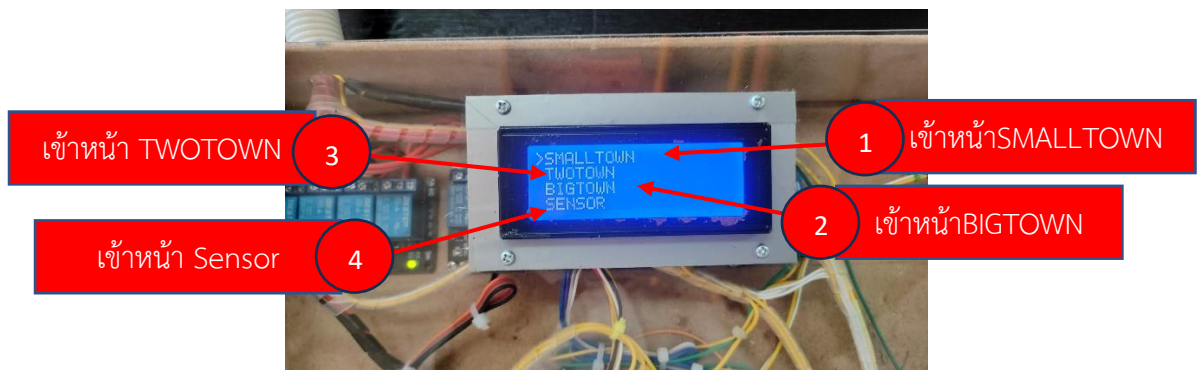


ภาพที่ ค-5 หน้า menu ใน Auto Mode Day



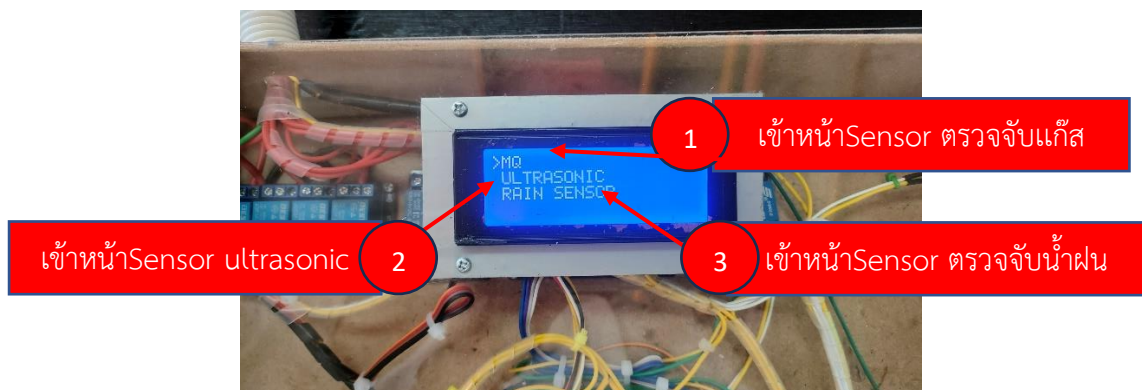
ภาพที่ ค-6 หน้า menu ใน Auto Mode Night

1.6 หน้า menu ใน Manual Mode



ภาพที่ ค-7 หน้า menu ใน Manual Mode

1.7 หน้า menu ใน Sensor



ภาพที่ ค-8 หน้า menu ใน Sensor

1.8 หน้า menu ใน Sensor ตรวจจับแก๊ส



ภาพที่ ค-9 หน้า menu ใน Sensor ตรวจจับแก๊ส

1.9 หน้า menu ใน Sensor ultrasonic



ภาพที่ ค-10 หน้า menu ใน Sensor ultrasonic

1.10 หน้า menu ใน Sensor ตรวจจับน้ำฝน



ภาพที่ ค-11 หน้า menu ใน Sensor ตรวจจับน้ำฝน

2. คุณสมบัติของ เมืองอัจฉริยะเอลิเซียม

2.1 มีระบบสแกนบัตร NFC

2.2 มีระบบไฟฟ้าอัตโนมัติ

2.3 มีการแสดงค่าแก๊สในอากาศ

2.4 มีการแจ้งเตือนแผ่นดินไหว

3. การติดตั้ง

3.1 เสียบอะแดปเตอร์เข้ากับไฟบ้านเพื่อใช้งาน

4. ข้อควรระวัง

4.1 ถ้าจำเป็นต้องเคลื่อนย้าย ควรย้ายอย่างระมัดระวัง

4.2 ห้ามทำความสะอาดบ่อยเพื่อป้องกันฝุ่น

ภาคผนวก ง.

โค้ดที่ใช้ใน ระบบเมืองอัจฉริยะเอลิเซีย

โค้ดที่ใช้ในระบบเมืองอัจฉริยะเอลิเซียม

Board 1

```
// btn 18 5 17 16 left to right
#include "WiFi.h"    //เรียกใช้งานไลบรารี wifi
#include <WiFiUDP.h> //เรียกใช้งานไลบรารี UDP
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
const char *ssid = "Freewifi";    //ชื่อ wifi ที่ต้องการเชื่อมต่อ
const char *password = "123456789"; //password ที่ต้องการเชื่อมต่อ
unsigned int UDPPort = 2390;      // เปิดใช้งานพอร์ต 2390
int packetSize;
String BufferString;
int len;
String data;
WiFiUDP Udp;                      //เปิดใช้ฟังก์ชัน UDP
IPAddress local_ip = { 192, 168, 1, 2 }; //ตั้งค่า IP
IPAddress gateway = { 192, 168, 1, 1 }; //ตั้งค่า IP Gateway
IPAddress subnet = { 255, 255, 255, 0 }; //ตั้งค่า Subnet
void automode();
void automode();
void twotown();
void smalltown();
void bigtown();
int wallmode = 0;
int kj = 0;
int rain = 0;
int i = 1; //loop pointer
int j = 1; //loop pointer14
int k1 = 0; //loop pointer
int k2 = 0;
```

```

int k3 = 0;
int k4 = 0;
int point = 0; //pointer at home screen
int screen = 0;
LiquidCrystal_I2C lcd(0x27, 20, 4); //address lcd
int relaylist[] = { 23, 19, 4, 2, 15, 32, 33, 25, 26, 27 };
int buzzer = 12;
void setup() {
  Serial.begin(115200);
  Serial.print("Connecting to ");          //แสดงข้อความ "Connecting to"
  Serial.println(ssid);                    //แสดงข้อความ ชื่อ wifi
  WiFi.begin(ssid, password);              // เชื่อมต่อ wifi
  while (WiFi.status() != WL_CONNECTED) { //รอจนกว่าจะเชื่อมต่อสำเร็จ
    delay(500);
    Serial.print(".");
  }
  Serial.print("WiFi UDP Mode : ");
  WiFi.config(local_ip, gateway, subnet); //set ค่า IP
  Serial.println(WiFi.localIP());          //แสดงหมายเลข IP ของ ESP32
  Udp.begin(UDPPort);                     //เริ่มใช้งานฟังก์ชัน UDP
  Serial.println("");
  Serial.println("WiFi connected"); //แสดงข้อความเชื่อมต่อสำเร็จ
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  Serial.begin(115200);
  for (int a = 0; a < sizeof(relaylist); a++) { //loop for pinmode relay
    pinMode(relaylist[a], OUTPUT);
  }
  for (int a = 0; a < sizeof(relaylist); a++) {
    digitalWrite(relaylist[a], 1);
  }
}

```

```

pinMode(34, INPUT);
pinMode(14, OUTPUT);
pinMode(16, INPUT_PULLUP); //red btn
pinMode(17, INPUT_PULLUP); //green btn
pinMode(5, INPUT_PULLUP); //blue btn
pinMode(18, INPUT_PULLUP); //yellow btn
pinMode(buzzer, OUTPUT); // เสียงกดปุ่ม
lcd.begin();
lcd.backlight();
digitalWrite(14, 1);
lcd.setCursor(0, 0);
lcd.print("*****");
lcd.setCursor(0, 1); // ไปที่ตัวอักษรที่ 0 แถวที่ 1
lcd.print("*Welcome To      *");
lcd.setCursor(0, 2); // ไปที่ตัวอักษรที่ 7 แถวที่ 2
lcd.print("*      ELYSIUM!! *");
lcd.setCursor(0, 3);
lcd.print("*****");
delay(2000);
lcd.clear();
}

void loop() {
  if (digitalRead(5) == 0) {
    lcd.clear();
    i = 1;
    screen = 0;
    //loop pointer
    j = 1; //loop pointer14
    k1 = 0; //loop pointer
    k2 = 0;
    k3 = 0;

```

```

digitalWrite(buzzer, 1);
delay(50);
digitalWrite(buzzer, 0);
k4 = 0;
}
if (i == 1 and screen == 0) {
    lcd.setCursor(2, 0);
    lcd.print("SELECT CITY MODE");
    lcd.setCursor(3, 1);
    lcd.print("SELECTED AUTO ");
    lcd.setCursor(0, 2);
    lcd.print("MODE : >AUTO< ");
    lcd.setCursor(7, 3);
    lcd.print(" MANUAL ");
    point = 1;
    wallmode = 1;
    for (int a = 0; a < sizeof(relaylist); a++) {
        digitalWrite(relaylist[a], 1);
    }
    digitalWrite(14, 1);
}
if (i == 2 and screen == 0) {
    lcd.setCursor(2, 0);
    lcd.print("SELECT CITY MODE");
    lcd.setCursor(3, 1);
    lcd.print("SELECTED MANUAL");
    lcd.setCursor(0, 2);
    lcd.print("MODE : AUTO ");
    lcd.setCursor(7, 3);
    lcd.print(">MANUAL<");
    point = 2;

```

```

wallmode = 1;
for (int a = 0; a < sizeof(relaylist); a++) {
    digitalWrite(relaylist[a], 1);
}
digitalWrite(14, 1);
}
if (digitalRead(18) == 0 and screen == 0) { //pointer เลื่อนได้
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    i = i + 1;
    delay(250);
}
if (i > 2) { //pointer วน
    i = 1;
}
if (digitalRead(17) == 0 and screen == 0) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    screen = 1;
    i = 0;
    delay(250);
    lcd.clear();
}
if (screen == 1 and point == 1) {
    int daylight = analogRead(34);
    digitalWrite(14, 0);
    if (daylight < 2000) {
        automodeoff();
        lcd.setCursor(5, 0);
    }
}

```

```
    lcd.print("AUTO MODE");
    lcd.setCursor(6, 1);
    lcd.print("  DAY  ");
    lcd.setCursor(7, 2);
    lcd.print("RUNNING");
    delay(250);
} else {
    automode();
    lcd.setCursor(5, 0);
    lcd.print("AUTO MODE");
    lcd.setCursor(8, 1);
    lcd.print("NIGHT");
    lcd.setCursor(7, 2);
    lcd.print("RUNNING");
    delay(250);
}
wallmode = 1;
}
```

```
if (digitalRead(16) == 0) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    screen = screen - 1;
    delay(250);
    i = 1;
    lcd.clear();
}
if (screen < 0) {
    screen = 0;
}
```

```
if (screen > 3) {  
    screen = 3;  
}  
  
if (screen == 1 and point == 2 and j == 1) {  
    lcd.setCursor(0, 0);  
    lcd.print(">SMALLTOWN");  
    lcd.setCursor(0, 1);  
    lcd.print(" TWOTOWN");  
    lcd.setCursor(0, 2);  
    lcd.print(" BIGTOWN");  
    lcd.setCursor(0, 3);  
    lcd.print(" SENSOR");  
    wallmode = 2;  
}  
  
if (screen == 1 and point == 2 and j == 2) {  
    lcd.setCursor(0, 0);  
    lcd.print(" SMALLTOWN");  
    lcd.setCursor(0, 1);  
    lcd.print(">TWOTOWN");  
    lcd.setCursor(0, 2);  
    lcd.print(" BIGTOWN");  
    lcd.setCursor(0, 3);  
    lcd.print(" SENSOR");  
    wallmode = 2;  
}  
  
if (screen == 1 and point == 2 and j == 3) {  
    lcd.setCursor(0, 0);  
    lcd.print(" SMALLTOWN");  
    lcd.setCursor(0, 1);  
    lcd.print(" TWOTOWN");  
    lcd.setCursor(0, 2);
```

```

    lcd.print(">BIGTOWN");
    lcd.setCursor(0, 3);
    lcd.print(" SENSOR");
    wallmode = 2;
}

if (screen == 1 and point == 2 and j == 4) {
    lcd.setCursor(0, 0);
    lcd.print(" SMALLTOWN");
    lcd.setCursor(0, 1);
    lcd.print(" TWOTOWN");
    lcd.setCursor(0, 2);
    lcd.print(" BIGTOWN");
    lcd.setCursor(0, 3);
    lcd.print(">SENSOR");
    wallmode = 2;
}

if (digitalRead(18) == 0 and screen == 1) { //pointer เลื่อนได้
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    j = j + 1;
    delay(250);
}

if (j > 4) {
    j = 1;
}

if (digitalRead(17) == 0 and screen == 1 and wallmode == 2) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    screen = 2;
}

```

```
    delay(250);
    lcd.clear();
}
if (screen == 2 and j == 1) {
    if (k1 == 0) {
        lcd.setCursor(0, 0);
        lcd.print(">TURN ON THE LIGHT");
        lcd.setCursor(0, 1);
        lcd.print(" TURN OFF THE LIGHT");
    }
    if (digitalRead(18) == 0) {
        digitalWrite(buzzer, 1);
        delay(50);
        digitalWrite(buzzer, 0);
        k1 = k1 + 1;
        delay(250);
    }
    if (k1 == 1) {
        lcd.setCursor(0, 0);
        lcd.print(" TURN ON THE LIGHT");
        lcd.setCursor(0, 1);
        lcd.print(">TURN OFF THE LIGHT");
    }
    if (k1 > 1) {
        k1 = 0;
    }
    if (digitalRead(17) == 0 and k1 == 0) {
        digitalWrite(buzzer, 1);
        delay(50);
        digitalWrite(buzzer, 0);
        smalltown();
    }
}
```

```
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(" LIGHT SMALLTOWN ON ");
    delay(1500);
    lcd.clear();
}
if (digitalRead(17) == 0 and k1 == 1) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    smalltownoff();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("LIGHT SMALLTOWN OFF");
    delay(1500);
    lcd.clear();
}
}
if (screen == 2 and j == 2) {
    if (k2 == 0) {
        lcd.setCursor(0, 0);
        lcd.print(">ON Light");
        lcd.setCursor(0, 1);
        lcd.print(" OFF Light");
    }
    if (digitalRead(18) == 0) {
        digitalWrite(buzzer, 1);
        delay(50);
        digitalWrite(buzzer, 0);
        k2 = k2 + 1;
        delay(250);
    }
}
```

```
}  
if (k2 == 1) {  
    lcd.setCursor(0, 0);  
    lcd.print(" ON Light");  
    lcd.setCursor(0, 1);  
    lcd.print(">OFF Light");  
}  
if (k2 > 1) {  
    k2 = 0;  
}  
if (digitalRead(17) == 0 and k2 == 0) {  
    digitalWrite(buzzer, 1);  
    delay(50);  
    digitalWrite(buzzer, 0);  
    twotown();  
    lcd.clear();  
    lcd.setCursor(0, 0);  
    lcd.print("LIGHT TWOTOWN ON ");  
    delay(1500);  
    lcd.clear();  
}  
if (digitalRead(17) == 0 and k2 == 1) {  
    digitalWrite(buzzer, 1);  
    delay(50);  
    digitalWrite(buzzer, 0);  
    twotownoff();  
    lcd.clear();  
    lcd.setCursor(0, 0);  
    lcd.print("LIGHT TWOTOWN OFF");  
    delay(1500);  
    lcd.clear();  
}
```

```

    }
}
if (screen == 2 and j == 3) {
    if (k3 == 0) {
        lcd.setCursor(0, 0);
        lcd.print(">ON Light");
        lcd.setCursor(0, 1);
        lcd.print(" OFF Light");
    }
    if (digitalRead(18) == 0) {
        digitalWrite(buzzer, 1);
        delay(50);
        digitalWrite(buzzer, 0);
        k3 = k3 + 1;
        delay(250);
    }
    if (k3 == 1) {
        lcd.setCursor(0, 0);
        lcd.print(" ON Light");
        lcd.setCursor(0, 1);
        lcd.print(">OFF Light");
    }
    if (k3 > 1) {
        k3 = 0;
    }
    if (digitalRead(17) == 0 and k3 == 0) {
        digitalWrite(buzzer, 1);
        delay(50);
        digitalWrite(buzzer, 0);
        bigtown();
        lcd.clear();
    }
}

```

```

    lcd.setCursor(0, 0);
    lcd.print("LIGHT BIGTOWN ON ");
    delay(1500);
    lcd.clear();
}

if (digitalRead(17) == 0 and k3 == 1) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    bigtownoff();
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(" LIGHT BIGTOWN OFF");
    delay(1500);
    lcd.clear();
}
}

if (screen == 2 and j == 4) {
    if (k4 == 0 and screen == 2) {
        lcd.setCursor(0, 0);
        lcd.print(">MQ ");
        lcd.setCursor(0, 1);
        lcd.print(" ULTRASONIC ");
        lcd.setCursor(0, 2);
        lcd.print(" RAIN SENSOR ");
    }

    if (k4 == 1 and screen == 2) {
        lcd.setCursor(0, 0);
        lcd.print(" MQ ");
        lcd.setCursor(0, 1);
        lcd.print(">ULTRASONIC ");
    }
}

```

```

    lcd.setCursor(0, 2);
    lcd.print(" RAIN SENSOR ");
}

if (k4 == 2 and screen == 2) {
    lcd.setCursor(0, 0);
    lcd.print(" MQ ");
    lcd.setCursor(0, 1);
    lcd.print(" ULTRASONIC ");
    lcd.setCursor(0, 2);
    lcd.print(">RAIN SENSOR ");
}

if (digitalRead(18) == 0) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    k4 = k4 + 1;
    delay(250);
}

}

if (k4 > 2) {
    k4 = 0;
}

if (digitalRead(17) == 0 and k4 == 0) {
    screen = 3;
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    lcd.clear();
}

if (digitalRead(17) == 0 and k4 == 1) {

```

```

screen = 3;
digitalWrite(buzzer, 1);
delay(50);
digitalWrite(buzzer, 0);
lcd.clear();
}
if (digitalRead(17) == 0 and k4 == 2) {
    screen = 3;
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    lcd.clear();
}
if (screen == 3 and k4 == 0) {
    kj = 2;
}
if (kj == 2) {
    for (;;) {
        packetSize = Udp.parsePacket(); // Check for incoming packets
        if (packetSize) {
            char packetBuffer[packetSize + 1];
            len = Udp.read(packetBuffer, packetSize);
            packetBuffer[len] = '\0';
            // Extract data from the received packet
            String receivedData = String(packetBuffer);
            // Find the position of the comma
            int firstCommaIndex = receivedData.indexOf(',');
            int secondCommaIndex = receivedData.indexOf(',', firstCommaIndex + 1);
            int thirdCommaIndex = receivedData.indexOf(',', secondCommaIndex + 1);
            // Split the received data into two parts based on the position of "Sensor2,"

```

```

    if (firstCommaIndex != -1 && secondCommaIndex != -1) {
        // Extract data for Sensor1
        String data1 = receivedData.substring(0, firstCommaIndex);
        // Extract data for Sensor2
        String data2 = receivedData.substring(firstCommaIndex + 1,
secondCommaIndex);
        // Extract data for Sensor3
        String data3 = receivedData.substring(secondCommaIndex + 1,
thirdCommaIndex);
        // Display the data on the LCD
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("MQ1: " + data2);
        lcd.setCursor(0, 1);
        lcd.print("MQ2: " + data3);
    } else {
        Serial.println("Invalid received data format");
    }
}

if (digitalRead(16) == 0) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    screen = screen - 1;
    delay(250);
    i = 1;
    kj = 0;
    lcd.clear();
    break;
}
delay(500);

```

```

    lcd.setCursor(8, 0);
    lcd.print("    ");
    lcd.setCursor(8, 1);
    lcd.print("    ");
    data = "";
}
}
if (screen == 3 and k4 == 1) {
    kj = 1;
}
if (kj == 1) {
    for (;;) {
        packetSize = Udp.parsePacket(); // Check for incoming packets
        if (packetSize) {
            char packetBuffer[packetSize + 1];
            len = Udp.read(packetBuffer, packetSize);
            packetBuffer[len] = '\0';
            // Extract data from the received packet
            String receivedData = String(packetBuffer);
            // Find the position of the comma
            int commaIndex = receivedData.indexOf(',');
            // Split the received data into two parts based on the position of "Sensor2,"
            if (commaIndex != -1) {
                // Extract data for Sensor1
                String data1 = receivedData.substring(0, commaIndex);
                // Extract data for Sensor2
                String data2 = receivedData.substring(commaIndex + 1);
                // Display the data on the LCD
                lcd.clear();
                lcd.setCursor(0, 0);
                lcd.print("Distance : " + data1 + " cm");
            }
        }
    }
}

```

```

        //lcd.setCursor(0, 1);
        //lcd.print("Sensor2: " + data2);
    } else {
        Serial.println("Invalid received data format");
    }
}

if (digitalRead(16) == 0) {
    digitalWrite(buzzer, 1);
    delay(50);
    digitalWrite(buzzer, 0);
    screen = screen - 1;
    delay(250);
    i = 1;
    kj = 0;
    lcd.clear();
    break;
}

delay(500);
lcd.setCursor(8, 0);
lcd.print("    ");
lcd.setCursor(8, 1);
lcd.print("    ");
data = "";
}
}

if (screen == 3 and k4 == 2) {
    kj = 2;
}

if (kj == 2) {
    for (;;) {
        packetSize = Udp.parsePacket(); // Check for incoming packets
    }
}

```

```

if (packetSize) {
    char packetBuffer[packetSize + 1];
    len = Udp.read(packetBuffer, packetSize);
    packetBuffer[len] = '\0';
    // Extract data from the received packet
    String receivedData = String(packetBuffer);
    // Find the position of the comma
    int firstCommaIndex = receivedData.indexOf(',');
    int secondCommaIndex = receivedData.indexOf(',', firstCommaIndex + 1);
    int thirdCommaIndex = receivedData.indexOf(',', secondCommaIndex + 1);
    // Split the received data into two parts based on the position of "Sensor2,"
    if (firstCommaIndex != -1) {
        String data1 = receivedData.substring(0, firstCommaIndex);
        // Extract data for Sensor2
        String data2 = receivedData.substring(firstCommaIndex+1, secondCommaIndex);
        // Extract data for Sensor3
        String data3 = receivedData.substring(secondCommaIndex+1, thirdCommaIndex);
        // Extract data for Sensor4
        String data4 = receivedData.substring(thirdCommaIndex + 1);
        int rain = data4.toInt();
        if (rain == 0) {
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("It's rainy");
        } else if (rain == 1) {
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("No rain ");
        } else {
            Serial.println("Invalid received data format");
        }
    }
}

```

```

    }
    if (digitalRead(16) == 0) {
        digitalWrite(buzzer, 1);
        delay(50);
        digitalWrite(buzzer, 0);
        screen = screen - 1;
        delay(250);
        i = 1;
        kj = 0;
        lcd.clear();
        break;
    }
    delay(500);
    lcd.setCursor(8, 0);
    lcd.print("    ");
    lcd.setCursor(8, 1);
    lcd.print("    ");
    data = "";
}
}
}
}

void automode() {
    for (int a = 0; a < sizeof(relaylist); a++) {
        digitalWrite(relaylist[a], 0);
    }
}

void automodeoff() {
    for (int a = 0; a < sizeof(relaylist); a++) {
        digitalWrite(relaylist[a], 1);
    }
}

```

```
}  
  
void twotown() {  
    digitalWrite(23, 0);  
    digitalWrite(19, 0);  
    digitalWrite(4, 0);  
    //digitalWrite(13,0);  
}  
  
void twotownoff() {  
    digitalWrite(23, 1);  
    digitalWrite(19, 1);  
    digitalWrite(4, 1);  
    //digitalWrite(13,1);  
}  
  
void smalltown() {  
    digitalWrite(2, 0);  
    digitalWrite(15, 0);  
}  
  
void smalltownoff() {  
    digitalWrite(2, 1);  
    digitalWrite(15, 1);  
}  
  
void bigtown() {  
    digitalWrite(32, 0);  
    digitalWrite(33, 0);  
    digitalWrite(25, 0);  
    digitalWrite(26, 0);  
}  
  
void bigtownoff() {  
    digitalWrite(32, 1);  
    digitalWrite(33, 1);  
    digitalWrite(25, 1);
```

```
digitalWrite(26, 1);  
}
```

Board 2

```
#include "WiFi.h"    //เรียกใช้งานไลบรารี wifi  
#include <WiFiUDP.h> //เรียกใช้งานไลบรารี UDP  
#include <Wire.h>    //เรียกใช้งานไลบรารีการรับ-ส่งข้อมูล  
  
const int pingPin = 13;  
int inPin = 12;  
  
const char *ssid = "Freewifi";    //ชื่อ wifi ที่ต้องการเชื่อมต่อ  
const char *password = "123456789"; //password ที่ต้องการเชื่อมต่อ  
  
unsigned int UDPPort = 2390;    // เปิดใช้งานพอร์ต 2390  
const char *host = "192.168.1.2"; //ใส่ IP อุปกรณ์แม่ข่าย(Server)ที่จะส่งข้อมูล  
  
int Status_LED = 2;  
int packetSize;  
int Gas_analog = 32;  
int Gas_analog2 = 33;  
int rain = 26;  
  
String BufferString;  
int len;  
  
String data1, data2, data3, data4;  
  
WiFiUDP Udp;    //เปิดใช้ฟังก์ชัน UDP  
IPAddress local_ip = { 192, 168, 1, 3 }; //ตั้งค่า IP  
IPAddress gateway = { 192, 168, 1, 1 }; //ตั้งค่า IP Gateway  
IPAddress subnet = { 255, 255, 255, 0 }; //ตั้งค่า Subnet  
  
void setup() {  
    pinMode(rain, INPUT);  
    Serial.begin(9600);  
    pinMode(Status_LED, OUTPUT);  
    Serial.print("Connecting to ");    //แสดงข้อความ "Connecting to"  
    Serial.println(ssid);    //แสดงข้อความ ชื่อ wifi  
    WiFi.begin(ssid, password);    // เชื่อมต่อ wifi
```

```

while (WiFi.status() != WL_CONNECTED) { //รอจนกว่าจะเชื่อมต่อสำเร็จ
    delay(500);
    Serial.print(".");
}
Serial.print("WiFi UDP Mode : ");
Serial.println(WiFi.localIP()); //แสดงหมายเลข IP ของ ESP32
Udp.begin(UDPPort); //เริ่มใช้งานฟังก์ชัน UDP
Serial.print("Connecting to "); //แสดงข้อความ "Connecting to"
Serial.println(ssid); //แสดงข้อความ ชื่อ wifi
WiFi.begin(ssid, password); // เชื่อมต่อ wifi
while (WiFi.status() != WL_CONNECTED) { //รอจนกว่าจะเชื่อมต่อสำเร็จ
    delay(500);
    Serial.print(".");
}
WiFi.config(local_ip, gateway, subnet); //set ค่า IP
Serial.println("");
Serial.println("WiFi connected"); //แสดงข้อความเชื่อมต่อสำเร็จ
Serial.println("IP address: ");
Serial.println(WiFi.localIP()); //แสดงหมายเลข IP ของ ESP32
}

void loop() {
    long duration, cm;
    pinMode(pingPin, OUTPUT);
    digitalWrite(pingPin, LOW);
    delayMicroseconds(2);
    digitalWrite(pingPin, HIGH);
    delayMicroseconds(5);
    digitalWrite(pingPin, LOW);
    pinMode(inPin, INPUT);
    duration = pulseIn(inPin, HIGH);
    cm = microsecondsToCentimeters(duration);

```

```

data1 = cm; //ข้อมูลที่ต้องการส่ง
data2 = analogRead(Gas_analog);
data3 = analogRead(Gas_analog2);
data4 = digitalRead(rain);

String dataToSend = String(data1) + "," + String(data2) + "," + String(data3) + "," +
String(data4);

Serial.println(dataToSend);

Udp.beginPacket(host, UDPPort);
Udp.print(dataToSend);           //คำสั่งส่งข้อมูล
Udp.endPacket();                 //ปิดการส่งข้อมูล
digitalWrite(Status_LED, !digitalRead(Status_LED)); //แสดงสถานะการส่งด้วย LED
delay(500);                     //หน่วงเวลาการส่งข้อมูล
//Serial.println(data1);
}

long microsecondsToCentimeters(long microseconds) {
    // The speed of sound is 340 m/s or 29 microseconds per centimeter.
    // The ping travels out and back, so to find the distance of the
    // object we take half of the distance travelled.
    return microseconds / 29 / 2;
}

```

Board 3

```

#include <ESP32Servo.h>

Servo myservo, myservo2; //ประกาศตัวแปรแทน Servo
int val, val2;
int vibration = 25;
int smallbuzzer = 23;
int bigbuzzer = 14;
int ir1 = 26;
int ir2 = 27;
int sun = 0;
int r = 19;

```

```

int y = 5;
int g = 18;
long delayTime = 0;
unsigned long prevTime_T1 = millis();
unsigned long prevTime_T2 = millis();
int state = 0;
void setup() {
    pinMode(r, OUTPUT);
    pinMode(y, OUTPUT);
    pinMode(g, OUTPUT);

    digitalWrite(r, 0);
    digitalWrite(g, 0);
    digitalWrite(y, 0);
    Serial.begin(9600);
    pinMode(ir1, INPUT);
    pinMode(ir2, INPUT);
    pinMode(vibration, INPUT);
    pinMode(smallbuzzer, OUTPUT);
    pinMode(bigbuzzer, OUTPUT);
    myservo.attach(16); // D22 (กำหนดขาควบคุม Servo)
    myservo2.attach(17); // D22 (กำหนดขาควบคุม Servo)
}
void loop() {
    unsigned long currentMillis = millis();
    if (currentMillis - prevTime_T1 > 1500) {
        if (state == 0) {
            digitalWrite(y, 0);
            digitalWrite(r, 1);
            digitalWrite(g, 0);
            state = 1;

```

```

} else if (state == 1) {
    digitalWrite(y, 0);
    digitalWrite(g, 1);
    digitalWrite(r, 0);
    state = 2;
} else if (state == 2) {
    digitalWrite(g, 0);
    digitalWrite(y, 1);
    digitalWrite(r, 0);
    state = 0;
}
prevTime_T1 = currentMillis;
// Update the timestamp
}

```

```

val = digitalRead(ir1); //อ่านค่าสัญญาณ digital ขา8 ที่ต่อกับ เซ็นเซอร์ตรวจจับวัตถุ IR
Infrared
val2 = digitalRead(ir2); //อ่
sun = analogRead(vibration);
if (val == 0) {
    myservo.write(90); // สั่งให้ Servo หมุนไปองศาที่ 0
    delay(3000);      // หน่วงเวลา 1000m
    myservo.write(0); // สั่งให้ Servo หมุนไปองศาที่ 180
    //delay(5000); // หน่วงเวลา 2000ms
}
if (val2 == 0) {
    myservo2.write(90); // สั่งให้ Servo หมุนไปองศาที่ 0
    delay(3000);      // หน่วงเวลา 1000m
}

```

```
myservo2.write(0); // สั่งให้ Servo หมุนไปองศาที่ 180
//delay(5000); // หน่วงเวลา 2000ms
}
```

```
if (sun == 4095) {
    digitalWrite(bigbuzzer, 1);
    delay(2000);
} else {
    digitalWrite(bigbuzzer, 0);
}
}
```

NFC

```
#include <Servo.h>
Servo myservo; //ประกาศตัวแปรแทน Servo
#include <SPI.h>
#include <RFID.h>
#define SS_PIN 10
#define RST_PIN 9
RFID rfid(SS_PIN, RST_PIN);
int buzzPin = 3;
int led1 = 2;
// Setup variables:
int serNum0;
int serNum1;
int serNum2;
int serNum3;
int serNum4;
void setup() {
    myservo.attach(5);
    Serial.begin(9600);
    SPI.begin();
}
```

```

rfid.init();
pinMode(led1, OUTPUT);
digitalWrite(led1, LOW);
myservo.write(0);
}

void loop() {
  if (rfid.isCard()) {
    if (rfid.readCardSerial()) {
      /* With a new cardnumber, show it. */
      Serial.println(" ");
      Serial.println("Card found");
      serNum0 = rfid.serNum[0];
      serNum1 = rfid.serNum[1];
      serNum2 = rfid.serNum[2];
      serNum3 = rfid.serNum[3];
      serNum4 = rfid.serNum[4];

      //Serial.println(" ");
      Serial.println("Cardnumber:");
      Serial.print("Dec: ");
      Serial.print(rfid.serNum[0], DEC);
      Serial.print(", ");
      Serial.print(rfid.serNum[1], DEC);
      Serial.print(", ");
      Serial.print(rfid.serNum[2], DEC);
      Serial.print(", ");
      Serial.print(rfid.serNum[3], DEC);
      Serial.print(", ");
      Serial.print(rfid.serNum[4], DEC);
      Serial.println(" ");
      Serial.print("Hex: ");

```

```

Serial.print(rfid.serNum[0], HEX);
Serial.print(" ");
Serial.print(rfid.serNum[1], HEX);
Serial.print(" ");
Serial.print(rfid.serNum[2], HEX);
Serial.print(" ");
Serial.print(rfid.serNum[3], HEX);
Serial.print(" ");
Serial.print(rfid.serNum[4], HEX);
Serial.println(" ");
//buzzer
analogWrite(3, 20);
delay(500);
analogWrite(3, 0);
}

if (rfid.serNum[0] == 172 && rfid.serNum[1] == 129 && rfid.serNum[2] == 26 &&
rfid.serNum[3] == 214 && rfid.serNum[4] == 225) {
    myservo.write(90); // สั่งให้ Servo หมุนไปองศาที่ 0
    delay(3000);      // หน่วงเวลา 1000ms
    myservo.write(0); // สั่งให้ Servo หมุนไปองศาที่ 90
    delay(3000);      // หน่วงเวลา 1000ms
    Serial.println("servojod");
} else if (rfid.serNum[0] == 220 && rfid.serNum[1] == 238 && rfid.serNum[2] == 28
&& rfid.serNum[3] == 214 && rfid.serNum[4] == 248) {
    myservo.write(90); // สั่งให้ Servo หมุนไปองศาที่ 0
    delay(3000);      // หน่วงเวลา 1000ms
    myservo.write(0); // สั่งให้ Servo หมุนไปองศาที่ 90
    delay(3000);      // หน่วงเวลา 1000ms
    Serial.println("servoguide");
}
}

```

```
    rfid.halt();  
}
```

ESP32 CAM

```
#include "esp_camera.h"  
#include <WiFi.h>  
#define CAMERA_MODEL_AI_THINKER  
#include "camera_pins.h"  
const char* ssid = "Freewifi";  
const char* password = "123456789";  
void startCameraServer();  
void setup() {  
    Serial.begin(115200);  
    Serial.setDebugOutput(true);  
    Serial.println();  
    camera_config_t config;  
    config.ledc_channel = LEDC_CHANNEL_0;  
    config.ledc_timer = LEDC_TIMER_0;  
    config.pin_d0 = Y2_GPIO_NUM;  
    config.pin_d1 = Y3_GPIO_NUM;  
    config.pin_d2 = Y4_GPIO_NUM;  
    config.pin_d3 = Y5_GPIO_NUM;  
    config.pin_d4 = Y6_GPIO_NUM;  
    config.pin_d5 = Y7_GPIO_NUM;  
    config.pin_d6 = Y8_GPIO_NUM;  
    config.pin_d7 = Y9_GPIO_NUM;  
    config.pin_xclk = XCLK_GPIO_NUM;  
    config.pin_pclk = PCLK_GPIO_NUM;  
    config.pin_vsync = VSYNC_GPIO_NUM;  
    config.pin_href = HREF_GPIO_NUM;  
    config.pin_sscb_sda = SIOD_GPIO_NUM;  
    config.pin_sscb_scl = SIOC_GPIO_NUM;
```

```

config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
//init with high specs to pre-allocate larger buffers
if (psramFound()) {
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;
    config.fb_count = 2;
} else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;
    config.fb_count = 1;
}

#ifdef CAMERA_MODEL_ESP_EYE
    pinMode(13, INPUT_PULLUP);
    pinMode(14, INPUT_PULLUP);
#endif

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    return;
}

sensor_t* s = esp_camera_sensor_get();
//initial sensors are flipped vertically and colors are a bit saturated
if (s->id.PID == OV3660_PID) {
    s->set_vflip(s, 1);    //flip it back
    s->set_brightness(s, 1);  //up the blightness just a bit
    s->set_saturation(s, -2); //lower the saturation
}

```

```
//drop down frame size for higher initial frame rate
s->set_framesize(s, FRAMESIZE_QVGA);
#if defined(CAMERA_MODEL_M5STACK_WIDE)
    s->set_vflip(s, 1);
    s->set_hmirror(s, 1);
#endif

WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
startCameraServer();
Serial.print("Camera Ready! Use 'http://");
Serial.print(WiFi.localIP());
Serial.println("' to connect");
}

void loop() {
    delay(10000);
}
```

ประวัติผู้จัดทำ



ประวัติผู้จัดทำโครงการ

ชื่อ นายพชร โสรส

เกิดเมื่อวันที่ 15 เดือน มีนาคม พ.ศ. 2547

ที่อยู่ 57 หมู่ 16 ต.ขุนซ่อง อ.แก่งหางแมว จ.จันทบุรี

หมายเลขโทรศัพท์ 097-036-1706

E-Mail : photcharasoros1@gmail.com

การศึกษา

ชั้นประถมศึกษา จากโรงเรียนลัมแบร์ตพิชญาลัย

ชั้นมัธยมศึกษาตอนต้น จากโรงเรียนลัมแบร์ตพิชญาลัย

ชั้นประกาศนียบัตรวิชาชีพ จากวิทยาลัยเทคนิคจันทบุรี

ขณะนี้กำลังศึกษาอยู่ในชั้นประกาศนียบัตรวิชาชีพชั้นสูงที่วิทยาลัยเทคนิคจันทบุรี



ประวัติผู้จัดทำโครงการ

ชื่อ นายวศิน เสริมสุข

เกิดเมื่อวันที่ 24 เดือน พฤษภาคม พ.ศ 2546

อยู่ที่ 71/12 ต.สองพี่น้อง อ.ท่าใหม่ จ.จันทบุรี

หมายเลขโทรศัพท์ 094-676-2480

E-Mail : guidesermnobikak@gmail.com

การศึกษา

ชั้นประถมศึกษา จากโรงเรียนประดิษฐ์ศึกษา

ชั้นมัธยมศึกษาตอนต้น จากโรงเรียนประดิษฐ์ศึกษา

ชั้นประกาศนียบัตรวิชาชีพ จากวิทยาลัยเทคนิคจันทบุรี

ขณะนี้กำลังศึกษาอยู่ในชั้นประกาศนียบัตรวิชาชีพชั้นสูงที่วิทยาลัยเทคนิคจันทบุรี