



แอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอที
EV-GOLF DATA ACQUISITION APPLICATION WITH IOT TECHNOLOGY

ธนพล จิตภักดี

THANAPON JITPAKDEE

วันเฉลิม เสียงเปรม

WANCHALOEM SIANGPRE

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมสารสนเทศ ภาควิชาวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

วิทยาเขตชุมพรเขตรอุดมศักดิ์ จังหวัดชุมพร

ปีการศึกษา 2565

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2565


สาขาวิชาวิศวกรรมสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง วิทยาเขตชุมพรเขตรอุดมศักดิ์ จังหวัดชุมพร

เรื่อง แอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอที
EV-Golf Data acquisition Application with IoT Technology

ผู้จัดทำ

1. นายชนพล จิตภักดิ์ รหัสนักศึกษา 63201265
2. วันเฉลิม เสียงเปรม รหัสนักศึกษา 63201267



.....อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.ศรัญญ์ ดวงสุวรรณ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2023

DEPARTMENT OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

PRINCE OF CHUMPHON CAMPUS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
วิทยาเขตชุมพรเขตรอุดมศักดิ์ จังหวัดชุมพร
ใบรับรองปริญญาานิพนธ์

ชื่อปริญญาานิพนธ์ แอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอที
Project Title EV-Golf Data acquisition Application with IoT Technology
ชื่อนักศึกษา นายธนพล จิตภักดิ์ รหัสนักศึกษา 63201265
นายวันเฉลิม เสียงเปรม รหัสนักศึกษา 63201267
ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมสารสนเทศ
อาจารย์ที่ปรึกษา ผศ.ดร.ศรัญ ดวงสุวรรณ

คณะกรรมการสอบปริญญาานิพนธ์			ลายมือชื่อ
ผศ.ดร.รัฐพงษ์	สุวลักษณ์	ประธานกรรมการ	รัฐพงษ์ สุวลักษณ์
รศ.ดร.บุญยวีร์	จามจรีกุลกาญจน์	กรรมการ	บุญยวีร์ จม
ผศ.ดร.ศรัญ	ดวงสุวรรณ	กรรมการ	ศรัญ
อ.อรรถศาสตร์	นาคเทวีญ	กรรมการ	อ.อรรถศาสตร์
อ.นภัสรพี	สิทธิวิจน์	กรรมการ	นภัสรพี สิทธิวิจน์

วัน/เดือน/ปี ที่สอบ 7 มิถุนายน พ.ศ. 2566 เวลา 09.00 น. - 17.00 น.
สถานที่สอบ ตึก E ห้อง E107

ภาควิชาวิศวกรรมศาสตร์ รับรองแล้ว

รัฐพงษ์ สุวลักษณ์

(ผู้ช่วยศาสตราจารย์ ดร.ณัฐพงศ์ รัตนเดช)

หัวหน้าภาควิชาวิศวกรรมศาสตร์

วันที่ 19 มิถุนายน พ.ศ. 2566

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อปริญญาบัตร	แอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอที	
นักศึกษา	นายธนพล จิตภักดี	รหัสนักศึกษา 63201265
	นายวันเฉลิม เสียงเปรม	รหัสนักศึกษา 63201267
อาจารย์ที่ปรึกษา	ผศ.ดร.ศรัญ ดวงสุวรรณ	
หลักสูตร	วิศวกรรมศาสตรบัณฑิต	
สาขาวิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2565	

บทคัดย่อ

รถกอล์ฟพลังงานไฟฟ้าเป็นยานพาหนะชนิดหนึ่งที่ใช้สำหรับขนส่งนักกีฬากอล์ฟไปยังตำแหน่งหลุมกอล์ฟต่างๆ ภายในสนาม เนื่องด้วยการใช้งานระบบไฟฟ้าจึงจำเป็นต้องทราบค่าระดับปริมาณพลังงานของแบตเตอรี่อย่างสม่ำเสมอ อย่างไรก็ตาม การใช้งานภายในสนามกอล์ฟจะมีพื้นที่กว้างมากส่งผลให้การส่งข้อมูลระดับพลังงานของแบตเตอรี่มีข้อจำกัด ทั้งนี้เทคโนโลยีไอโอทีที่สามารถตอบโจทย์การใช้งานในการสื่อสารข้อมูลได้ในระยะไกล คือ ลอรา (LoRa) ซึ่งเป็นทางเลือกที่เหมาะสมที่สุดสำหรับการนำมาใช้ติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าในสนามกอล์ฟ

ปริญญาบัตรฉบับนี้นำเสนอการศึกษาและพัฒนาอุปกรณ์ติดตามข้อมูลตำแหน่ง ระดับแบตเตอรี่ และอุณหภูมิของรถกอล์ฟพลังงานไฟฟ้า โดยสามารถรายงานข้อมูลผ่านแอปพลิเคชันได้ทันที อุปกรณ์ที่พัฒนาจะประกอบด้วย ไมโครคอนโทรลเลอร์ ESP8266 เซนเซอร์ DHT11 บอร์ดลอรา TTGO ร่วมกับบอร์ด ESP-NOW และเชื่อมต่อโมดูล GPS รุ่น NEO-6M ทั้งนี้การใช้งานเกตเวย์จะเปิดการใช้บริการเครือข่ายไอโอทีจากบริษัท โทรคมนาคมแห่งชาติ จำกัด หรือ NT เพื่อแสดงผลข้อมูลบนแอปพลิเคชัน Cayenne ทั้งนี้การทดลองได้ทดสอบ ณ สนามกอล์ฟแหลมฉบังอินเตอร์เนชั่นแนล อ.ศรีราชา จ.ชลบุรี จากผลการทดลองตามระยะทางทั้งหมดของสนามกอล์ฟสามารถติดตามข้อมูลพิกัดตำแหน่ง ระดับแบตเตอรี่ และอุณหภูมิของแบตเตอรี่ได้จริง ประโยชน์ที่ได้รับจากปริญญาบัตรฉบับนี้จะเป็แนวทางของการพัฒนาอุปกรณ์ไอโอทีสำหรับติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าได้ในระยะไกล

คำสำคัญ : การติดตามข้อมูล, เทคโนโลยีไอโอที, รถกอล์ฟพลังงานไฟฟ้า

Project Title	EV-Golf Data acquisition Application with IoT Technology
Students	Mr.Thanapon Jitpakdee Student ID 63201265 Mr.Wanchaloem Siangpream Student ID 63201267
Advisor	Asst. Prof. Dr. Sarun Duangsuwan
Degree	Bachelor of Engineering
Program in	Information Engineering
Academic Year	2022

ABSTRACT

Electric golf carts are a type of vehicle used to transport golf players to golf holes on the course. Due to the use of electrical systems, it is necessary to know the level of battery power regularly where the golf area is very wide, resulting that is limited of the transmission battery based on power level information. IoT technology that can use for long-distance data communication, that is LoRa, which suitable option for tracking electric vehicle (EV) golf on the course.

This thesis presents a study and development of location tracking devices on battery level, positioning, and temperature of EV golf. The data acquisition can be reported immediately through the application. The developed equipment will consist of ESP8266 microcontroller, DHT11 sensor, LaRa TTGO board connected with ESP-NOW board and link GPS module model NEO-6M. The cloud network was used by National Telecommunications Co., Ltd. or NT to display data on the Canyenne application. The experiment was tested at LaemChabang International Golf Club, Si Racha District, Chonburi Province. As the result, we can actually track the information such as battery level, positioning, and temperature immediately. The benefits of this work will guide the development of IoT devices for remote tracking of EV golf.

Keyword Data acquisition, IoT technology, EV golf energy

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้อย่างสมบูรณ์ นั้นเป็นเพราะผู้จัดทำได้รับความช่วยเหลือ ได้รับการสนับสนุน เอื้อเฟื้อเครื่องมือ และได้รับคำปรึกษาทั้งทางตรงและทางอ้อมจากบุคคลหลายท่าน ซึ่งผู้จัดทำรู้สึกซาบซึ้งใจอยากขอขอบพระคุณ ขอขอบคุณและขอขอบใจ จึงขอแสดงออกถึงสิ่งที่รู้สึกไว้ ณ ที่แห่งนี้

ขอขอบพระคุณครอบครัวและบุพการีผู้ซึ่งเป็นกำลังใจ ส่งความห่วงใย สนับสนุนค่าใช้จ่าย ด้วยดีเสมอมา บุคคลเหล่านี้เป็นกลุ่มบุคคลที่ผู้จัดทำอยากกล่าวถึงเป็นลำดับแรก มีคำที่อยากบอกมากมายแต่ขอกลับออกมาเพียงหนึ่งคำที่มีความหมายอัดแน่นนั่นคือ ขอขอบพระคุณครับ

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.ศรัญ ดวงสุวรรณ อาจารย์ที่ปรึกษา ผู้ซึ่งให้คำแนะนำต่างๆ รวมทั้งให้คำแนะนำแนวทางในการทำโครงการ และติดตามถามไถ่เกี่ยวกับงานโครงการอย่างสม่ำเสมอ ผู้เขียนรู้สึกซาบซึ้งในความเมตตาของท่านจึงขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบคุณเพื่อนๆ และน้องๆ ที่คอยช่วยเหลือกันในการทำปริญญานิพนธ์ชิ้นนี้จนสำเร็จได้อย่างสมบูรณ์ ขอขอบคุณที่สู้มาด้วยกัน อดทนมาด้วยกัน

และสุดท้ายขอขอบพระคุณอาจารย์ที่เคารพทุกท่านที่ให้ความเอาใจใส่แนะนำ คอยช่วยเหลือเสมอมา คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ ผู้จัดทำขอมอบแด่ผู้มีพระคุณทุกท่าน คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ ผู้จัดทำขอมอบแก่ผู้มีพระคุณทุกท่าน

ธนพล จิตภักดี

วันเฉลิม เสียงเปรม

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.5 แผนการดำเนินงาน.....	2
1.6 โครงสร้างของปริญญาานิพนธ์.....	3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	4
2.1 เทคโนโลยี LoRa.....	4
2.1.1 สถาปัตยกรรมเครือข่าย LoRa.....	4
2.1.2 พื้นฐานของเทคโนโลยี LoRa.....	5
2.1.3 คุณสมบัติของ LoRa Protocol.....	5
2.1.4 ความปลอดภัยของเครือข่าย LoRa.....	6
2.2 การวัดอุณหภูมิ.....	6
2.3 แบตเตอรี่ลิเธียมไอออน.....	8
2.3.1 ชนิดของแบตเตอรี่ลิเธียม.....	8
2.3.1.1 แบตเตอรี่ลิเธียมฟอสเฟต.....	9
2.3.1.2 แบตเตอรี่ลิเธียมโคบอลต์ออกไซด์.....	10
2.3.1.3 แบตเตอรี่ลิเธียมแมงกานีสออกไซด์.....	11
2.3.1.4 แบตเตอรี่ลิเธียมนิเคิลแมงกานีสโคบอลต์ออกไซด์.....	12
2.3.1.5 แบตเตอรี่ลิเธียมนิเคิลโคบอลต์อลูมิเนียมออกไซด์.....	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.3.1.6 แบตลิเธียมไททานต.....	14
2.4 การระบุตำแหน่ง GPS.....	15
2.5 Arduino IDE	16
2.5.1 แนวคิดการใช้งานโปรแกรม Arduino IDE	16
2.5.2 วิธีใช้งานโปรแกรม Arduino IDE	17
2.6 Cayenne	20
2.6.1 การใช้งาน Cayenne บนเว็บไซต์	21
2.6.2 การใช้งาน Cayenne บนแอปพลิเคชัน.....	25
2.7 TTGO t-beam esp32 lora	27
2.8 Pzem-017.....	28
2.8.1 คุณสมบัติ.....	29
2.8.2 ข้อมูลจำเพาะ.....	29
2.9 เซ็นเซอร์วัดอุณหภูมิ DHT11.....	30
2.9.1 คุณสมบัติ.....	30
2.9.2 การใช้งานและการพัฒนา.....	30
2.9.3 ตารางการเชื่อมต่อ	30
2.10 MAX485	31
2.11 MODBUS	32
2.12 GY-NEO6MV2 Ublox NEO-6M GPS Module	32
2.13 CAT LoRaIoT Platform	33
2.14 ESP-NOW	36
2.15 Received Signal Strength Indicator (RSSI).....	37
2.16 Signal-to-Noise Ratio (SNR).....	38
บทที่ 3 การออกแบบ.....	39
3.1 Block Diagram	39
3.1.1 Block Diagram แสดงการเชื่อมต่ออุปกรณ์	39
3.1.2 Block Diagram แสดงการทำงาน LoRa	39
3.2 Flow chart	40
3.3 การต่อวงจร	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.3.1 วงจร TTGO t-beam esp32 lora	41
3.3.2 วงจร Esp8266	41
3.4 การออกแบบกล่องเซนเซอร์	42
3.5 การติดตั้งอุปกรณ์จริง	43
3.6 หน้าเว็บไซต์ Cayenne	43
3.7 หน้าแอปพลิเคชัน Cayenne	44
บทที่ 4 ผลการทดลอง	46
4.1 สถานที่การทดลอง.....	46
4.2 เตรียมการทดลอง	47
4.2.1 อุปกรณ์ที่ใช้	47
4.2.2 พารามิเตอร์ที่เก็บผล	49
4.3 ผลการทดลอง.....	50
บทที่ 5 บทสรุป	55
5.1 สรุปผลการทดลอง.....	55
5.2 ปัญหาและอุปสรรค	55
5.3 วิธีการแก้ไขปัญหา.....	55
เอกสารอ้างอิง	56
ภาคผนวก ก ข้อมูลเฉพาะของอุปกรณ์ที่ใช้ (Datasheet).....	57
ภาคผนวก ข ซอร์สโค้ด (Source Code)	76
ภาคผนวก ค คู่มือการใช้งานระบบ (Manual).....	88
ประวัติผู้จัดทำ	107

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนและวิธีการดำเนินงานภาคเรียนที่ 1	2
1.2 ขั้นตอนและวิธีการดำเนินงานภาคเรียนที่ 2	2
2.1 คุณสมบัติของ LoRa Protocol	6
2.2 ตารางการเชื่อมต่อเซ็นเซอร์ DHT11.....	32



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 ความปลอดภัยของเครือข่าย LoRa	6
2.2 แบตเตอรี่ลิเธียมไอออน	8
2.3 แบตเตอรี่ลิเธียมฟอสเฟต.....	9
2.4 คุณสมบัติแบตเตอรี่ลิเธียมฟอสเฟต	9
2.5 แบตเตอรี่ลิเธียมโคบอลต์ออกไซด์	10
2.6 คุณสมบัติแบตเตอรี่ลิเธียมโคบอลต์ออกไซด์	11
2.7 คุณสมบัติแบตเตอรี่ลิเธียมแมงกานีสออกไซด์	12
2.8 คุณสมบัติแบตเตอรี่ลิเธียมนิเคิลแมงกานีสโคบอลต์ออกไซด์	13
2.9 คุณสมบัติแบตเตอรี่ลิเธียมนิเคิลโคบอลต์อลูมิเนียมออกไซด์	14
2.10 ภาพแสดงคุณสมบัติแบตเตอรี่ลิเธียมไททานต	15
2.11 การเปิดโปรแกรม Arduino IDE	17
2.12 การสร้างไฟล์ใหม่โปรแกรม Arduino IDE	17
2.13 การเขียน code ในโปรแกรม Arduino IDE	18
2.14 การบันทึกไฟล์โปรแกรม Arduino IDE	19
2.15 การเลือกรุ่นของบอร์ด	19
2.16 การเลือก Port ของบอร์ดที่เชื่อมต่อ	20
2.17 การอัปโหลด code ลงบอร์ด.....	20
2.18 หน้าเว็บไซต์ Cayenne.....	21
2.19 หน้าลงทะเบียนเว็บไซต์ Cayenne	21
2.20 หน้าเข้าสู่ระบบเว็บไซต์ Cayenne	22
2.21 หน้าเลือกอุปกรณ์เว็บไซต์ Cayenne.....	22
2.22 หน้าเชื่อมต่ออุปกรณ์เว็บไซต์ Cayenne	23
2.23 หน้าเชื่อมต่ออุปกรณ์เสร็จสมบูรณ์.....	23
2.24 หน้าเพิ่ม Widgets	24
2.25 หน้า Overview	24
2.26 หน้า Data	25
2.27 แอปพลิเคชัน Cayenne	25
2.28 การเข้าสู่ระบบแอปพลิเคชัน Cayenne	26
2.29 หน้าเลือกอุปกรณ์แอปพลิเคชัน Cayenne	26
2.30 หน้า Overview อุปกรณ์แอปพลิเคชัน Cayenne.....	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
2.31 TTGO t-beam esp32 lora	27
2.32 Pinout Diagram TTGO t-beam esp32 lora	28
2.33 Pzem-017.....	28
2.34 การต่อใช้งาน Pzem-017	29
2.35 เซ็นเซอร์ DHT11.....	31
2.36 การต่อใช้งานเซ็นเซอร์ DHT11.....	31
2.37 เซ็นเซอร์ MAX485.....	32
2.38 GY-NEO6MV2 Ublox NEO-6M GPS Module	33
2.39 การต่อใช้งาน GY-NEO6MV2 Ublox NEO-6M GPS Module	33
2.40 เข้าสู่ระบบเว็บไซต์ CAT LoRaIoT Platform	34
2.41 การลงทะเบียนก่อนเข้าใช้งานเว็บไซต์	34
2.42 การเพิ่มอุปกรณ์.....	35
2.43 การ Routing Profile โดยใช้ URL.....	35
2.44 หน้าแสดงข้อมูลของอุปกรณ์ที่ส่งข้อมูลมาให้เว็บไซต์	36
2.45 การเชื่อมต่อระหว่างบอร์ดโดยใช้ mac address	37
2.46 การส่งข้อมูลไปยังบอร์ดอีกตัว.....	37
3.1 Block Diagram แสดงการเชื่อมต่ออุปกรณ์	39
3.2 Block Diagram แสดงการทำงาน LoRa	39
3.3 Flow chart	40
3.4 การต่อวงจร TTGO t-beam esp32 lora	41
3.5 การต่อวงจร ESP 8266.....	41
3.6 โครงสร้างภายในกล่อง.....	42
3.7 โครงสร้างฝาปิดกล่อง	42
3.8 รถกอล์ฟที่ใช้ทดสอบจริง	43
3.9 หน้าแสดงข้อมูลภายในเว็บไซต์ Cayenne.....	43
3.10 หน้าแสดงข้อมูลภายในแอปพลิเคชัน Cayenne.....	44
3.11 แสดงข้อมูลที่ Device ส่งข้อมูลมายัง Platform	45
4.1 สนามกอล์ฟแหลมฉบังอินเตอร์เนชันแนล.....	46
4.2 ตำแหน่งสนามทดสอบจาก Google map	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.3 รถกอล์ฟไฟฟ้าที่ใช้พลังงานจากแบตเตอรี่แรงดัน 48 โวลต์.....	47
4.4 สมาร์ทโฟนใช้สำหรับติดต่อสื่อสาร.....	48
4.5 โน้ตบุ๊กใช้สำหรับรายงานข้อมูล.....	48
4.6 ตำแหน่งกล่องเซนเซอร์ที่ติดตั้ง.....	49
4.7 ติดตั้งอุปกรณ์เข้ากับรถกอล์ฟ.....	50
4.8 รถกอล์ฟที่พร้อมทดสอบ.....	50
4.9 ตำแหน่งหลุมกอล์ฟในสนาม.....	51
4.10 การระบุตำแหน่งของรถกอล์ฟในสนาม.....	52
4.11 ตารางผลการทดลอง.....	53
4.12 แผนที่ความครอบคลุมสัญญาณ LoRa.....	54

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

รถกอล์ฟคือรถที่มักใช้ในสนามกอล์ฟ เนื่องจากสนามมีขนาดใหญ่ จึงใช้เพื่อขนของหรือสัมภาระต่างๆ ที่มีน้ำหนักมากและพาโปรกอล์ฟไปยังจุดต่างๆ เพื่อสนุกสนานกับกีฬากอล์ฟ มีทั้งแบบเป็นรถกอล์ฟไฟฟ้าและน้ำมันเป็นพลังงาน แต่แน่นอนว่าทุกวันนี้ โลกของเราเปลี่ยนแปลงไปอย่างรวดเร็ว และได้รับผลกระทบจากมลพิษทางอากาศมากมาย ดังนั้นรถกอล์ฟที่แนะนำคงจะหนีไม่พ้นรถกอล์ฟไฟฟ้าที่ขับเคลื่อนด้วยระบบไฟฟ้า ซึ่งก็มีข้อเสียที่เกิดขึ้นในเรื่องของตำแหน่งของรถกอล์ฟที่ไม่สามารถทราบได้ทำให้บางครั้งโปรกอล์ฟอาจจะใช้เวลาในการรอรถนาน ปริมาณคงเหลือและความร้อนของแบตเตอรี่

สนามกอล์ฟส่วนใหญ่มีพื้นที่อยู่ประมาณ 500-700 ไร่ ซึ่งอินเทอร์เน็ตไม่สามารถเข้าถึงได้ทุกจุด และในส่วนของรถกอล์ฟที่ไม่สามารถทราบถึงตำแหน่ง อุณหภูมิและปริมาณของแบตเตอรี่ ทางผู้จัดทำได้เล็งเห็นปัญหา จึงได้นำเทคโนโลยีเครือข่ายที่สามารถส่งสัญญาณทางไกล หรือ “Long Range (LoRa)” โดยใช้ย่านความถี่ที่ 923 MHz ซึ่งรองรับในประเทศไทย ระยะในการส่งจะอยู่ที่ 5-15 กม. โดยการส่งข้อมูลนั้นจะส่งไปยังเกตเวย์ของ NT จากนั้นเกตเวย์จะส่งข้อมูลไปแสดงผลบน Application Server ที่ชื่อว่า Cayenne

1.2 วัตถุประสงค์

1. เพื่อศึกษาปัจจัยความต้องการของการติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้า
2. เพื่อพัฒนาชุดเซนเซอร์ติดตามข้อมูลแบตเตอรี่ ตำแหน่ง และอุณหภูมิรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอทีในระยะไกล
3. เพื่อสร้างต้นแบบแอปพลิเคชันติดตามข้อมูลแบตเตอรี่ ตำแหน่ง และอุณหภูมิรถกอล์ฟพลังงานไฟฟ้า

1.3 ขอบเขตของโครงการ

1. สามารถระบุข้อมูลตำแหน่งจีพีเอสผ่านอุปกรณ์ลอรา LoRa ได้
2. สามารถรายงานข้อมูลระดับพลังงานไฟฟ้าอุปกรณ์ลอรา LoRa ได้
3. ออกแบบส่วนการใช้งานของแอปพลิเคชันทั้ง User interface (UI) ได้
4. สามารถแสดงส่วนของข้อมูลตำแหน่ง ระดับพลังงานไฟฟ้าบนหน้าจอแอปพลิเคชันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้ในการสร้างแอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอที
2. ได้รับความรู้ในการพัฒนาระบบติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอที
3. ได้รับความรู้ในการออกแบบแอปพลิเคชันแสดงข้อมูลสถานะ ตำแหน่ง ระดับพลังงานที่ใช้ งานได้ของรถกอล์ฟพลังงานไฟฟ้า

1.5 แผนการดำเนินงาน

ขั้นตอนการดำเนินงานทั้งหมดที่ผู้จัดทำได้วางแผนเอาไว้ จะแบ่งออกเป็น 2 ช่วงใหญ่ๆ คือ แผนการดำเนินงานในภาคเรียนที่ 1 และ 2 ซึ่งได้แจกแจงรายละเอียดไว้ในตารางที่ 1.1 และ 1.2 ดังนี้

ตารางที่ 1.1 ขั้นตอนและวิธีการดำเนินงานภาคเรียนที่ 1

ขั้นตอนการดำเนินงาน	ระยะเวลาดำเนินงาน (สิงหาคม - ธันวาคม 2565)																					
	สิงหาคม				กันยายน				ตุลาคม				พฤศจิกายน				ธันวาคม					
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
1. พัฒนาส่วนของการระบุตำแหน่งรถกอล์ฟด้วยเทคโนโลยีไอโอที																						
2. พัฒนาส่วนของการรายงานข้อมูลระดับพลังงานไฟฟ้าของรถกอล์ฟด้วยเทคโนโลยีไอโอที																						

ตารางที่ 1.2 ขั้นตอนและวิธีการดำเนินงานภาคเรียนที่ 2

ขั้นตอนการดำเนินงาน	ระยะเวลาดำเนินงาน (มกราคม - พฤษภาคม 2566)																					
	มกราคม				กุมภาพันธ์				มีนาคม				เมษายน				พฤษภาคม					
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
1. ออกแบบส่วนการใช้งานของแอปพลิเคชันทั้ง User experience (UX) และ User interface (UI)																						
2. ทดสอบแสดงผลส่วนของข้อมูลตำแหน่ง ระดับพลังงานไฟฟ้าบนหน้าจอแอปพลิเคชัน																						
3. ประเมินผลการทดสอบและสรุปผลโครงการ																						

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 โครงสร้างปฏิญานิพนธ์

ปฏิญานิพนธ์ฉบับนี้แบ่งออกเป็น 8 ส่วนหลัก ๆ คือ บทที่ 1-5 และภาคผนวก ก-ค ซึ่งแต่ละส่วนจะอธิบายเนื้อหาที่เกี่ยวข้องกับโครงการแอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอทีที่มีระบบ สมัครสมาชิกและระบบล็อกอินข้อมูลที่ถูกกรอกก่อนเข้าสู่เว็บแอปพลิเคชัน Cayenne จะถูกเก็บไว้ในระบบฐานข้อมูลทั้งหมด ส่วนเว็บแอปพลิเคชันจะนำเสนอเนื้อหาเกี่ยวกับการวัดอุณหภูมิ แรงดัน และการระบุตำแหน่ง โดยจะใช้ Cayenne ในการนำเสนอเนื้อหาที่เหมาะสมกับผู้ใช้งาน

บทที่ 1 บทนำ

ในบทนี้จะกล่าวถึงความเป็นมาและความสำคัญของโครงการ จุดประสงค์ของโครงการ ขอบเขตของการทำโครงการ ประโยชน์ที่คาดว่าจะได้รับ ขั้นตอนการจัดทำโครงการ และโครงสร้างของปฏิญานิพนธ์ฉบับนี้

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องกับการออกแบบ โครงการแอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอที LoRa , การวัดอุณหภูมิ , แบตเตอรี่ลิเธียมไอออน , GPS , Arduino IDE , Cayenne , TTGO T-Beam ESP32 LoRa 923MHz V.1.1 , Pzem-017 , DHT11 , MAX485 , MODBUS , GY-NEO6MV2 ,NT LoRaloT Platform , ESP-NOW , Received Signal Strength Indicator (RSSI) , Signal-to-Noise Ratio (SNR)

บทที่ 3 การออกแบบ

ในบทนี้จะกล่าวถึงการออกแบบและหลักการบล็อกไดอะแกรมของการเชื่อมต่ออุปกรณ์ , บล็อกไดอะแกรมของการทำงาน LoRa , Flow chart , การต่อวงจร , การออกแบบ , การติดตั้งอุปกรณ์ , แสดงหน้าเว็บไซต์ แอปพลิเคชัน Cayenne และ เว็บไซต์ NT LoRaloT Platform

บทที่ 4 ผลการทดลอง

ในบทนี้จะกล่าวถึงการทดลองการทำงานของอุปกรณ์ตามที่ได้ออกแบบไว้

บทที่ 5 บทสรุป

ในบทนี้จะกล่าวถึง สรุปผลการทดลอง ปัญหาและอุปสรรคในการทำงาน

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 เทคโนโลยี LoRa

เทคโนโลยี LoRa คือเทคโนโลยีการสื่อสารไร้สายที่ใช้ความถี่วิทยุและการแพร่กระจายสัญญาณจากเครื่องส่งไปยังเครื่องรับในระยะไกล โดยมีการใช้ความถี่สาธารณะ (ISM) ซึ่งเป็นช่วงความถี่ที่ใช้ได้โดยได้รับการอนุญาตจาก กสทช. สามารถใช้งานได้ไกลกว่าสื่อสารแบบ Wi-Fi หรือแบบ Bluetooth วิธีการแพร่กระจายสัญญาณแบบช่องย่อย (Spread Spectrum) เพื่อเพิ่มประสิทธิภาพในการสื่อสาร และใช้เทคนิคการมอดูเลตแบบ CSS (Chirp Spread Spectrum) เพื่อลดการชนกันของสัญญาณระหว่างอุปกรณ์ ทำให้สามารถสื่อสารในระยะไกลได้ด้วยการใช้พลังงานไฟฟ้าที่ต่ำ มีการใช้งานกับอุปกรณ์ Internet of Things (IoT) โดยอุปกรณ์ LoRa สามารถติดต่อสื่อสารกับเกตเวย์ (Gateway) ที่เป็นตัวกลางในการเชื่อมต่ออุปกรณ์ LoRa กับอินเทอร์เน็ต เกตเวย์จะเป็นตัวแปลงสัญญาณจากโปรโตคอล เป็นโปรโตคอลอื่นๆ เช่น TCP/IP เพื่อให้ข้อมูลจากอุปกรณ์ สามารถส่งไปยังแอปพลิเคชันหรือระบบที่เกี่ยวข้องได้ [3]

2.1.1 สถาปัตยกรรมเครือข่าย LoRa

เครือข่าย LoRa ประกอบด้วยองค์ประกอบหลายอย่าง

1. จุดสิ้นสุดคือองค์ประกอบของเครือข่าย LoRa ที่มีการควบคุมหรือการตรวจจับปกติจะอยู่ห่างไกล
2. LoRa เกตเวย์รับโครงสร้างพื้นฐานจากปลายทาง LoRa แล้วโอนไปยังระบบ backhaul. ส่วนนี้ของเครือข่าย LoRa สามารถเป็นแบบเซลลูลาร์ได้ อีเธอร์เน็ตหรือลิงค์โทรคมนาคมอื่น ๆ แบบไร้สายหรือแบบมีสาย เกตเวย์เชื่อมต่อกับเซิร์ฟเวอร์เครือข่ายโดยใช้การเชื่อมต่อ IP ทั่วไป ด้วยวิธีนี้ข้อมูลจะใช้โปรโตคอลทั่วไป แต่สามารถเชื่อมต่อกับเครือข่ายโทรคมนาคมใดก็ได้ ไม่ว่าจะเป็นส่วนตัวหรือสาธารณะในแง่ของความคล้อยคลึงของเครือข่าย LoRa กับเครือข่ายเซลลูลาร์ เกตเวย์ LoRaWAN มักจะอยู่ร่วมกับสถานีฐานมือถือ ทางนี้สามารถใช้ความจุเพิ่มเติมบนเครือข่าย backhaul ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 พื้นฐานของเทคโนโลยี LoRa

มีองค์ประกอบสำคัญหลายประการของเทคโนโลยี LoRa คุณสมบัติที่สำคัญบางประการ ได้แก่

1. ระยะในการส่ง 15-20 กม.

2. การใช้พลังงานต่ำ

3. ความเสถียรภาพสูง: การใช้เทคโนโลยี Spread Spectrum ช่วยลดอิทธิพลของสัญญาณรบกวนจากแหล่งอื่นๆ ทำให้เครือข่าย LoRa มีความเสถียรภาพสูงและสามารถทำงานได้ในสภาพแวดล้อมที่มีการรบกวนจากสัญญาณอื่นๆ ได้

เทคโนโลยี LoRa มีการนำมาใช้งานในหลายสถานการณ์ เช่น ระบบเครือข่ายเมืองอัจฉริยะ (Smart City), การติดตามและการจัดการทรัพย์สิน (Asset Tracking), การวิเคราะห์และการควบคุมในการเกษตร (Agricultural Monitoring and Control), ระบบอัตโนมัติในโรงงาน (Industrial Automation) และอื่นๆ ที่ต้องการการสื่อสารไร้สายระยะไกลที่ประหยัดพลังงานได้

2.1.3 คุณสมบัติของ LoRa Protocol

ตารางต่อไปนี้แสดงคุณสมบัติหลักบางประการของโปรโตคอล LoRa เช่น การมอดูเลต ความจุและช่วง

ตารางที่ 2.1 คุณสมบัติของ LoRa Protocol

ข้อมูลจำเพาะ	พีเจอร์ LoRa
พิสัย	2-5 กม.เออร์เบิน (1.24-3.1 มิ)
	15Km ชานเมือง (9.3 มิ)
ความถี่	ISM 920-925 MHz
มาตรฐาน	IEEE 802.15.4g
การมอดูเลต	ประเภทการมอดูเลตสเปกตรัมตามพัลส์ FM ซึ่งแตกต่างกัน.
MKGW-มินิซีรีส์	หนึ่งเกตเวย์ LoRa ใช้โหนดหลายโหนด
แบตเตอรี่	อายุการใช้งานแบตเตอรี่ยาวนาน
LoRa ชั้น คุณภาพ	ความถี่, พลัง, การมอดูเลตและการส่งสัญญาณระหว่างโหนดและเกตเวย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

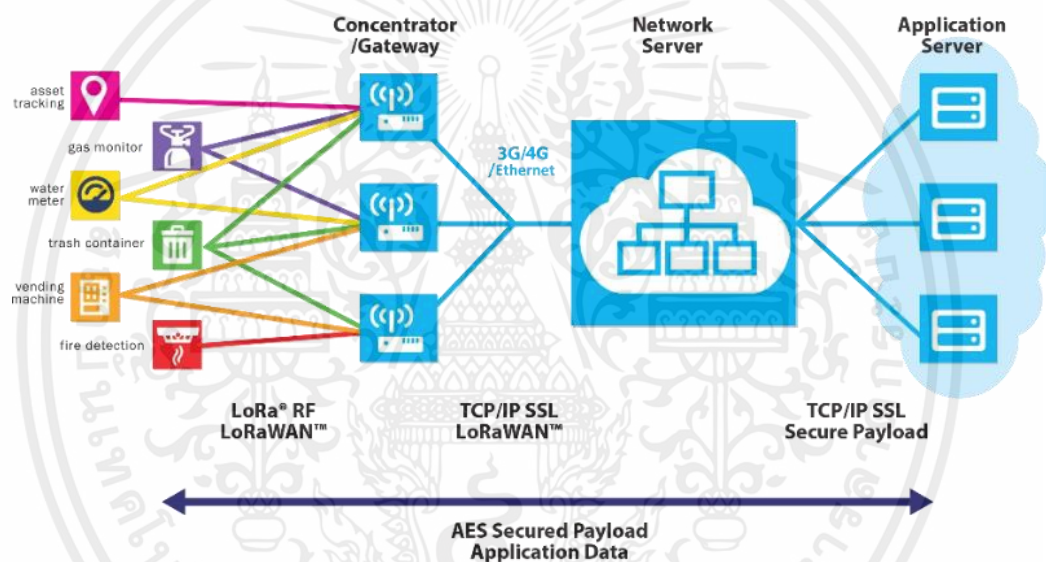
2.1.4 ความปลอดภัยของเครือข่าย LoRa

ปัญหาความปลอดภัยของเครือข่ายเริ่มมีความสำคัญขึ้นเรื่อย ๆ เนื่องจากเครือข่าย LoRa นั้นต้องการการรักษาความปลอดภัยระดับสูงเพื่อป้องกันปัญหาของระบบต่างๆ เพื่อให้ได้ระดับความปลอดภัยที่จำเป็นสำหรับเครือข่าย LoRa มีการใช้การเข้ารหัสหลายชั้น

ประเภทการเปิดใช้งานมี 2 แบบ คือ

1. OTAA ประกอบด้วย Device EUI (16-character hexadecimal), Application EUI (16-character hexadecimal), Application Key (32-character hexadecimal)

2. ABP ประกอบด้วย Device EUI (16-character hexadecimal), Device Address (8-character hexadecimal), Network Session Key (32-character hexadecimal), Application Session Key (32-character hexadecimal)



รูปที่ 2.1 ความปลอดภัยของเครือข่าย LoRa

2.2 การวัดอุณหภูมิ

การวัดอุณหภูมิ (temperature measurement) เป็นการวัดที่อาศัยหลักการแตกต่างจากหลักการวัดปริมาณทางฟิสิกส์ของตัวแปรอื่นๆ ตรงที่การวัดอุณหภูมิสนใจ "สเกล" (scale) ในขณะที่การวัดอื่นๆ สนใจ "หน่วย" (unit) เช่น การวัดความยาว หากให้นิยามว่าไม้ 1 อันมีความยาวเท่ากับ 1 เมตร ถ้านำไม้ 2 อันที่มีความยาวเท่ากับที่กำหนดนิยามไว้มาวางต่อกัน จะได้ความยาวเท่ากับ 2 เมตร โดยแนวคิดนี้ไม่สามารถนำมาใช้กับการวัดอุณหภูมิได้ เช่น แก้วน้ำ 2 ใบบรรจุน้ำด้วยปริมาณที่เท่ากัน อุณหภูมิเท่ากันเท่ากับ T เมื่อนำมาเทรวมกันในแก้วใบที่ 3 ผลที่ได้คือ น้ำในแก้วใบที่ 3 มีปริมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพิ่มขึ้นในขณะที่อุณหภูมิยังคงเท่าเดิมซึ่งมีค่าเท่ากับ T หลักการวัดอุณหภูมิแบ่งตามการเปลี่ยนแปลงสมบัติได้เป็น 4 กลุ่ม ดังนี้

1. หลักการวัดอุณหภูมิโดยอาศัยการเปลี่ยนแปลงสมบัติทางไฟฟ้า

ค่าสมบัติทางไฟฟ้าจะเปลี่ยนแปลงตามอุณหภูมิ เช่น เทอร์โมคัปเปิล (thermocouple) ทำงานโดยอาศัยหลักการเปลี่ยนแปลงค่าแรงดันไฟฟ้าอาร์ทีดี (RTD) และเทอร์มิสเตอร์ (thermister) ทำงานโดยอาศัยหลักการเปลี่ยนแปลงค่าความต้านทาน เป็นต้น

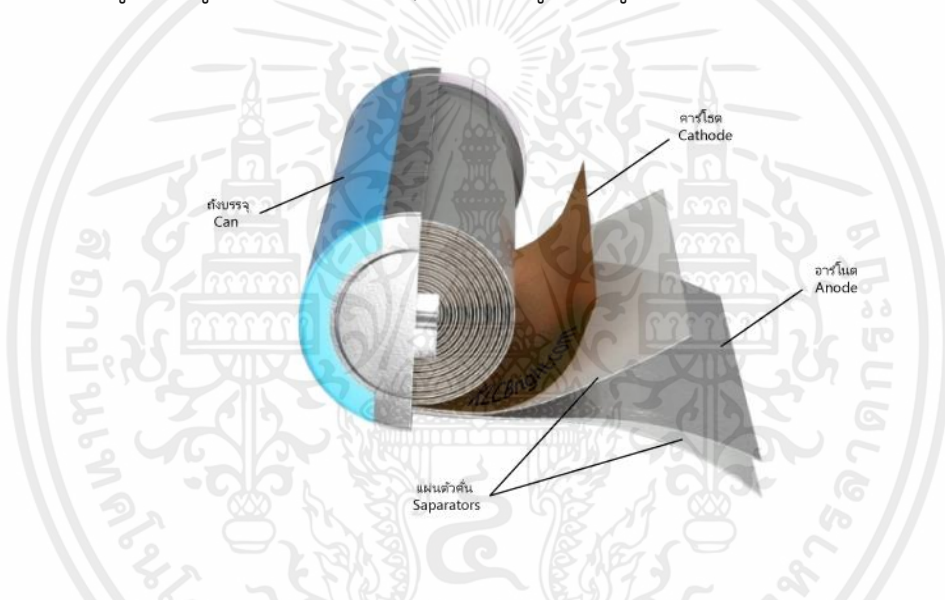
2. หลักการวัดอุณหภูมิโดยอาศัยการเปลี่ยนแปลงสมบัติทางแสงและการแผ่รังสีการวัดอุณหภูมิที่สูงมากๆ ไม่สามารถใช้เครื่องมือวัด (instrument) ดังกล่าวข้างต้นได้ เนื่องจากอาจทำให้เกิดความเสียหายต่อเครื่องมือวัดจึงจำเป็นต้องใช้เครื่องมือวัดชนิดพิเศษที่สามารถวัดอุณหภูมิได้โดยไม่ต้องสัมผัสกับวัตถุที่ต้องการวัด แต่อาศัยการวัดการแผ่รังสีความร้อนของวัตถุเพื่อบ่งบอกอุณหภูมิ โดยทั่วไปวัตถุในช่วงอุณหภูมิประมาณ 800 ถึง 1,800 องศาเซลเซียส แผ่รังสีออกมาในรูปของแสงในย่านที่ตามองเห็น ส่วนวัตถุในช่วงอุณหภูมิต่ำกว่า 800°C จนถึงอุณหภูมิห้อง วัตถุจะแผ่รังสีออกมาในย่านของรังสีอินฟราเรด (infrared radiation) ซึ่งอุณหภูมิแตกต่างกันความยาวคลื่นหรือความถี่ของรังสีที่แผ่ออกมาจากวัตถุจะแตกต่างกันด้วย เครื่องมือวัดอุณหภูมิที่อาศัยหลักการแผ่รังสีและสมบัติเชิงแสงนี้เรียกว่า ไพโรมิเตอร์ (pyrometer) สามารถแบ่งประเภทตามหลักการงานได้ 3 ประเภท คือ ไพโรมิเตอร์ชนิดเทียบความสว่างของไส้หลอด (optical pyrometer) ไพโรมิเตอร์ชนิดวัดการแผ่รังสี (radiation pyrometer) และไพโรมิเตอร์ชนิดอินฟราเรด (infrared pyrometer)

3. หลักการวัดอุณหภูมิโดยอาศัยการเปลี่ยนแปลงสมบัติทางกลหลักการวัดอุณหภูมิของเทอร์โมมิเตอร์ประเภทนี้มีหลายแบบขึ้นอยู่กับชนิดของเทอร์โมมิเตอร์ เช่น เทอร์โมมิเตอร์แบบของเหลวบรรจุในหลอดแก้วปิด (liquid filled in glass thermometer) ทำงานโดยอาศัยการเปลี่ยนแปลงการขยายตัวของของเหลวที่บรรจุอยู่ภายในหลอดแก้วเทอร์โมมิเตอร์แบบเปลี่ยนการขยายตัวเป็นความดัน (pressure thermometer) และเทอร์โมมิเตอร์แบบแถบโลหะคู่ (bi-metal thermometer) โดยเทอร์โมมิเตอร์ประเภทนี้มีช่วงของการวัดอุณหภูมิ (range) ค่อนข้างแคบขึ้นอยู่กับวัสดุที่ใช้และชนิดของเทอร์โมมิเตอร์

4. หลักการวัดอุณหภูมิโดยอาศัยการเปลี่ยนแปลงสมบัติทางเคมี เครื่องมือวัดอุณหภูมิประเภทนี้ทำงานโดยอาศัยอุณหภูมิเฉพาะที่สารเคมีละลายหรือเปลี่ยนสีเป็นจุดสังเกต เช่น อุปกรณ์วัดอุณหภูมิแบบดินสอ (crayon temperature indicator) อุปกรณ์วัดอุณหภูมิแบบแล็กเกอร์ (lacquer temperature indicator) อุปกรณ์วัดอุณหภูมิแบบเม็ดยา (pellet temperature indicator) และอุปกรณ์วัดอุณหภูมิแบบแผ่นฉลาก (label temperature indicator) เป็นต้น

2.3 แบตเตอรี่ลิเทียมไอออน

แบตเตอรี่ลิเทียม (Lithium Battery) หรือ เรียกแบบเต็มว่า แบตเตอรี่ลิเทียมไอออน (Lithium Ion Battery) นั้นเรียกชื่อตามการกักเก็บพลังงานในลิเทียมไอออน (Li ion) โดยการสร้างศักย์ไฟฟ้าให้เกิดขึ้นระหว่างขั้วบวกและขั้วลบของแบตเตอรี่และคั่นด้วยฉนวนที่เรียกว่า ตัวคั่น(Saparator) ตัวคั่นหรือ Saparator สามารถเป็นได้ทั้งฉนวนและนำไอออน เมื่อตอนขณะชาร์จไฟ ลิเทียมไอออนจะเคลื่อนที่จากขั้วบวกไปยังขั้วลบผ่านตัวคั่น และเมื่อตอนคายประจุ ลิเทียมไอออนจะเคลื่อนที่ในทิศตรงกันข้าม การเคลื่อนที่ของประจุลิเทียมไอออนนี้เองทำให้เกิดความต่างศักย์ไฟฟ้าหรือที่เรียกว่าโวลท์ (Voltage)เกิดขึ้น แต่เมื่อไม่มีการใช้งานแบตเตอรี่ตัวคั่นจะทำหน้าที่เป็นฉนวนกั้นไม่ให้ไอออนเคลื่อนที่เข้ามาไปมา และเมื่อมีการเอาแบตเตอรี่ไปต่อเข้ากับอุปกรณ์เพื่อใช้งาน ประจุลิเทียมไอออนหรืออิเล็กตรอนที่ถูกกั้นอยู่ด้วยตัวคั่นหรือ Saparator อยู่ นั้นจะถูกดันให้วิ่งผ่านตัวคั่นได้



รูปที่ 2.2 แบตเตอรี่ลิเทียมไอออน

2.3.1 ชนิดของแบตเตอรี่ลิเทียม (Types of Lithium Battery)

ประเภทหรือชนิดของแบตเตอรี่ลิเทียม นั้น ส่วนใหญ่เรียกชนิดตามองค์ประกอบของคาร์โธด แต่ก็ยังมีที่เรียกตามองค์ประกอบของอาร์โนดด้วยเช่นกัน ดูจากภายนอกเราไม่สามารถจำแนกได้ ดังนั้นเราจึงควรตรวจสอบให้แน่ชัดและศึกษารายละเอียดให้แน่ใจก่อนเลือกซื้อและนำไปใช้งาน ทั้งนี้เพื่อความปลอดภัย ประสิทธิภาพสูงสุดและมีคุณสมบัติตรงตามแต่ละประเภทการใช้งานของอุปกรณ์หรือเครื่องมือ นั้นๆ ชนิดของแบตเตอรี่ลิเทียมมี 6 ชนิดหลักๆ ได้แก่

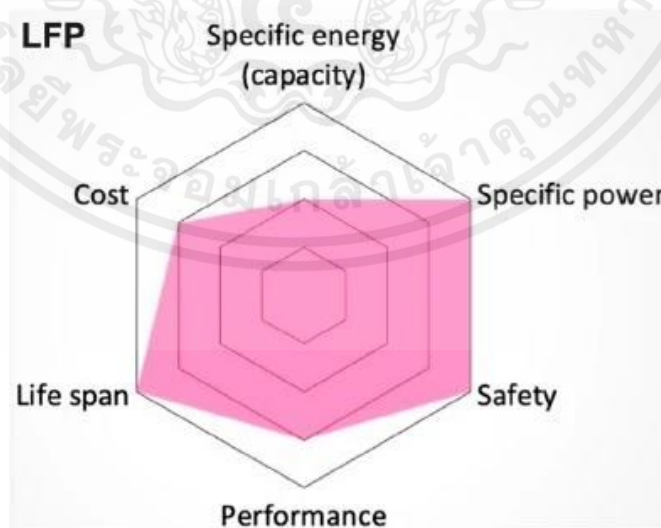
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1.1 แบตเตอรี่ลิเทียมฟอสเฟต



รูปที่ 2.3 แบตเตอรี่ลิเทียมฟอสเฟต

แบตเตอรี่ลิเทียมฟอสเฟต หรือ LiFePO_4 หรือ LFP หรือเราอาจจะรู้จักกันในชื่อ LiPO หรือที่บ้านเรานิยมเรียกกันว่า แบต LiPO เป็นแบตเตอรี่ลิเทียมที่มีใช้ทั่วไปและใช้อยู่มากที่สุด ใช้ฟอสเฟตเป็นคาร์โบไซด์ ใช้กราฟไฟต์เป็นอาร์โนด แบต LiPO มีรอบอายุการใช้งานที่นาน ไม่ค่อยมีปัญหาด้านความร้อน มีประสิทธิภาพด้านเคมีไฟฟ้าที่ดี แบตเตอรี่ LFP หรือ LiPO 1 cell มีความต่างศักย์หรือแรงดันไฟฟ้าประมาณ 3.2V ถ้าต้องการนำไปใช้งานเป็นระบบ 12V ต้องเอามาต่ออนุกรมกัน 4cell จะได้ 12.8V จึงเป็นเหตุผลว่าทำไมแบต LiPO จึงนำมาใช้อย่างแพร่หลายที่สุด รวมถึงไฟถนนโซล่าเซลล์ เพราะสามารถใช้ 4cell แล้วนำมาใช้แทนแบตเตอรี่ตะกั่วหรือแบบ deep cycle ได้



รูปที่ 2.4 คุณสมบัติแบตเตอรี่ลิเทียมฟอสเฟต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ข้อเด่นของแบตเตอรี่ลิเทียมฟอสเฟต

1.1 อายุการใช้งานนาน ปกติแล้วแบตเตอรี่ลิเทียมฟอสเฟตมีอายุการใช้งาน 2000 cycles หรือมากกว่า

1.2 deep of discharge สูง ปกติแล้วแบตเตอรี่ลิเทียมฟอสเฟตมี deep of discharge สูงประมาณ 80% สูงถึง 100% สำหรับผู้ผลิตบางรายโดยไม่ได้ทำให้แบตเตอรี่เสียหาย

1.3 ปลอดภัยและทนต่ออุณหภูมิได้สูง

2. ข้อด้อยของแบตเตอรี่ลิเทียมฟอสเฟต

2.1 แบตเตอรี่ลิเทียมฟอสเฟตมีค่าพลังงานจำเพาะไม่สูง

2.2 ประสิทธิภาพจะลดลงเมื่อใช้งานในที่ที่มีอุณหภูมิต่ำ

2.3 ไม่ค่อยเหมาะใช้งานที่มีการใช้งานแบบกระแสวิกฤตสูงๆ (high cranking applications) เป็นเพราะด้วยคุณสมบัติ ข้อด้อย 2 ข้างข้างบนจึงทำให้แบตเตอรี่ลิเทียมฟอสเฟตไม่ค่อยเหมาะใช้งานที่มีการใช้งานแบบกระแสวิกฤตสูงๆ เช่น การสตาร์ทเครื่องยนต์

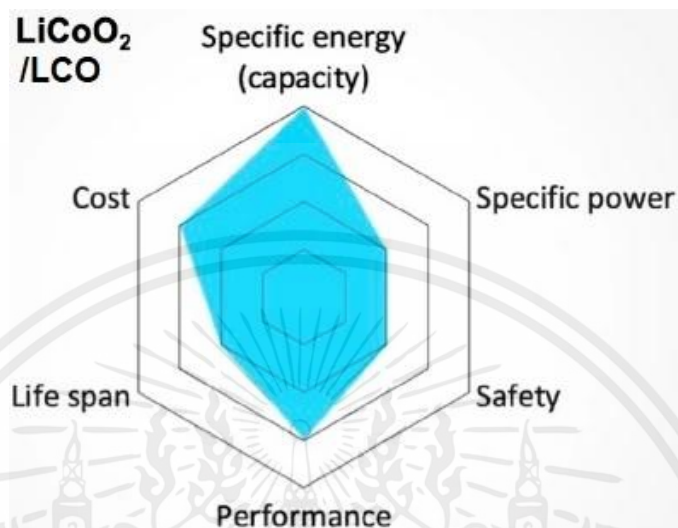
2.3.1.2 แบตเตอรี่ลิเทียมโคบอลต์ออกไซด์



รูปที่ 2.5 แบตเตอรี่ลิเทียมโคบอลต์ออกไซด์

แบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ หรือ LCO หรือที่บ้านเรานิยมเรียกว่า แบตลิเทียมไอออน (ซึ่งไม่ถูกต้องมากนัก เพราะว่าแบตลิเทียมไอออนเป็นคำที่กว้าง ที่กำลังกล่าวถึงทั้งหมดล้วนเป็นลิเทียมไอออน) มีค่าพลังงานจำเพาะ (specific power) สูงแต่มีกำลังงานจำเพาะ (specific power) ต่ำ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล่าวคือสามารถจ่ายไฟได้นานแต่จ่ายไฟให้กับอุปกรณ์ที่กินไฟสูงได้ไม่ดี จึงเหมาะและมีใช้ในงาน อุปกรณ์อิเล็กทรอนิกส์ เช่น โทรศัพท์มือถือ แท็บเล็ต โน้ตบุ๊ก กล้องถ่ายรูป ที่ต้องการกำลังน้อยๆแต่อยู่ได้นานๆ โดยที่ แบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ 1 cell มีแรงดันไฟฟ้าประมาณ 3.7V



รูปที่ 2.6 คุณสมบัติแบตเตอรี่ลิเทียมโคบอลต์ออกไซด์

1. ข้อเด่นของแบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ LCO

1.1 ข้อดีของแบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ คือมีค่าพลังงานจำเพาะสูง สามารถจ่ายไฟให้กับอุปกรณ์ที่กินไฟต่ำได้นาน

2. ข้อด้อยของแบตเตอรี่ลิเทียมโคบอลต์ออกไซด์ LCO

2.1 มีอายุการใช้งานสั้น ประมาณ 500-1000cycles

2.2 วัสดุขั้ว โคบอลต์ มีราคาค่อนข้างสูง

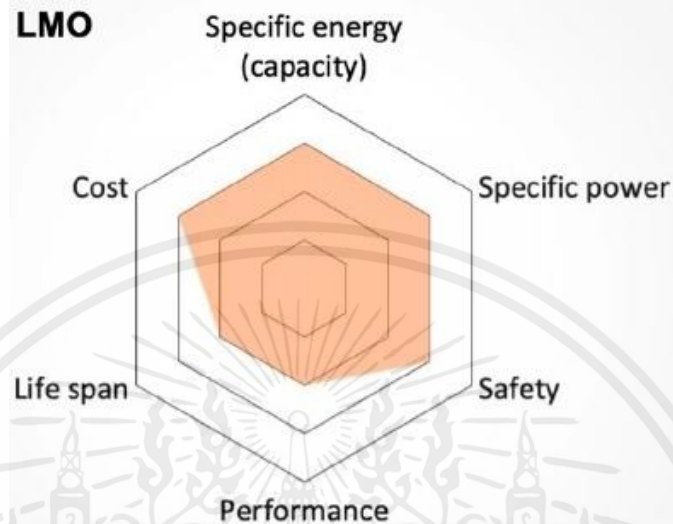
2.3 ทนความร้อนได้ต่ำจึงมีปัญหาเรื่องความปลอดภัยอยู่พอสมควร

2.4 ไม่เหมาะใช้งานกับอุปกรณ์ที่กินไฟสูง

2.3.1.3 แบตเตอรี่ลิเทียมแมงกานีสออกไซด์

แบตเตอรี่ลิเทียมแมงกานีสออกไซด์ หรือ LMO ใช้ส่วนผสมของธาตุลิเทียมแมงกานีสออกไซด์ เป็นคาร์โธด์ ซึ่งองค์ประกอบของธาตุสามชนิดนี้ เมื่อรวมกันจะสร้างให้เกิดโครงสร้างแบบ 3 มิติ ทำให้การไหลของอิออนดีขึ้น มีความต้านทานภายในต่ำลง เพิ่มกระแสได้มากขึ้นและมีค่าการทนต่อความร้อนได้มากขึ้นด้วย การค้นพบแบตเตอรี่ LMO (ประมาณ ค.ศ.1981) ถือเป็นความสำเร็จก้าวสำคัญในวงการแบตเตอรี่ ด้วยความที่มีความต้านทานภายในเซลล์ต่ำจึงสามารถชาร์จแบบเร็ว (fast charge) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจ่ายกระแสสูง (high discharge current) ได้ แบตเตอรี่ LMO สามารถที่จ่ายกระแสได้ถึง 20-30A โดยที่ค่าความร้อนไม่ขึ้นสูงมากได้ แต่แบตเตอรี่ LMO ก็มีค่าการเก็บพลังงานได้น้อย และอายุการใช้งานที่ไม่ได้ยาวนานนักแบตเตอรี่ชนิด LMO มีใช้งาน ในอุปกรณ์ประเภทเครื่องมือไร้สาย (power tools) อุปกรณ์ทางการแพทย์ รถไฟฟ้าและรถไฮบริด บางรุ่น



รูปที่ 2.7 คุณสมบัติแบตเตอรี่ลิเทียมแมงกานีสออกไซด์

1. ข้อเด่นของแบตเตอรี่ลิเทียมแมงกานีสออกไซด์

- 1.1 ชาร์จเร็ว
- 1.2 มีค่าพลังงานจำเพาะสูงกว่าแบตเตอรี่ LCO
- 1.3 ขนาดเล็กกว่า LCO
- 1.4 จ่ายกระแสได้สูง

2. ข้อด้อยของแบตเตอรี่ลิเทียมแมงกานีสออกไซด์

- 2.1 มีอายุการใช้งานได้สั้น เพียง 300-700cycles เท่านั้น และสั้นที่สุดในบรรดาแบตเตอรี่ลิเทียมทั้งหมด

2.3.1.4 แบตเตอรี่ลิเทียมนิเคิลแมงกานีสโคบอลต์ออกไซด์

แบตเตอรี่ลิเทียมนิเคิลแมงกานีสโคบอลต์ออกไซด์ (Lithium Nickel Manganese Cobalt Oxide: NMC) หรือ NMC ด้วยการเพิ่มนิเคิล Ni ซึ่งมีความจุพลังงานจำเพาะ (Specific Capacity; Ah/kg) สูง เข้าไปในส่วนประกอบของคาร์โบไซด์ ทำให้แบตเตอรี่ NMC มีค่าพลังงานจำเพาะสูงขึ้น ผลที่ได้คือได้เซลล์ที่มีความตึงเครียดหรือแรงดันไฟฟ้าสูงถึงประมาณ 4V/cell เดิมทีเดียว

แบตเตอรี่ชนิด NMC ไม่ได้มีส่วนผสมของแมงกานีส ต่อมานักวิจัยจึงได้เพิ่มแมงกานีสเข้าไปเพื่อเพิ่มเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเสถียร ผลก็คือแบตเตอรี่ NMC มีทั้งความเสถียรและใช้งานได้ในงานแรงดันสูง (high voltage applications) การปรับเปลี่ยนส่วนผสมระหว่าง นิกเกิล แมงกานีส และโคบอลต์ ทำให้เกิดชนิดย่อย และคุณสมบัติที่แตกต่างกันและยังคงเป็นจุดขายของผู้ผลิตแต่ละราย เช่น รุ่น NMC111 (ส่วนผสมเท่ากัน) รุ่น NMC442 รุ่น NMC622 หรือ รุ่น NM811 ในปัจจุบันแบตเตอรี่ NMC มีความต้องการสูงในอุตสาหกรรมรถยนต์ EV เพราะด้วยคุณสมบัติ มีค่าพลังงานสูง จ่ายกระแสได้สูง มีเสถียรภาพ และทนต่ออุณหภูมิได้ดีมาก มีใช้ในรถยนต์ไฟฟ้าหลายรุ่น เช่น Tesla Model S, Nissan Leaf, Chevrolet Volt, BMW i3



รูปที่ 2.8 คุณสมบัติแบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์

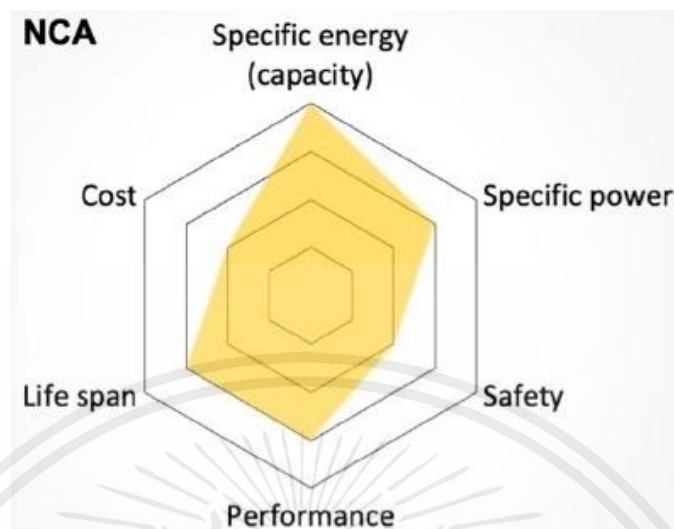
1. ข้อเด่นของแบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์ NMC
 - 1.1 มีค่าพลังงานจำเพาะสูง
 - 1.2 สามารถจ่ายกระแสได้มาก
 - 1.3 ชาร์จเร็ว
 - 1.4 มีอายุการใช้งานนาน
2. ข้อด้อยของแบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์ NMC
 - 2.1 ข้อด้อยของแบตเตอรี่ลิเทียมนิกเกิลแมงกานีสโคบอลต์ออกไซด์ คือมีแรงดันไฟฟ้าต่อเซลล์ต่ำกว่า LCO เล็กน้อย

2.2.1.5 แบตเตอรี่ลิเทียมนิกเกิลโคบอลต์อลูมิเนียมออกไซด์

(Lithium Nickel Cobalt Aluminum Oxide: NCA) หรือ NCA มีคุณสมบัติคล้ายกับ NMC

คือเก็บไฟได้มากและจ่ายไฟได้นาน แต่มีข้อเสียคือด้านความปลอดภัยยังไม่ค่อยสูงต้องมีระบบเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจเช็คความปลอดภัยอยู่ตลอดเวลาเมื่อนำมาใช้ในรถ EV จนถึงปัจจุบันมี Tesla เพียงเจ้าเดียวที่ใช้ แบตเตอรี่ NCA มีในรุ่น Model 3 และ Model S รุ่นแรก (2012)



รูปที่ 2.9 คุณสมบัติแบตเตอรี่ลิเทียมนิเคิลโคบอลต์อลูมิเนียมออกไซด์

1. ข้อดีแบตเตอรี่ลิเทียมนิเคิลโคบอลต์อลูมิเนียมออกไซด์ NCA

- 1.1 มีค่าพลังงานจำเพาะสูง
- 1.2 สามารถจ่ายกระแสได้มาก
- 1.3 ชาร์จเร็ว
- 1.4 มีอายุการใช้งานนาน

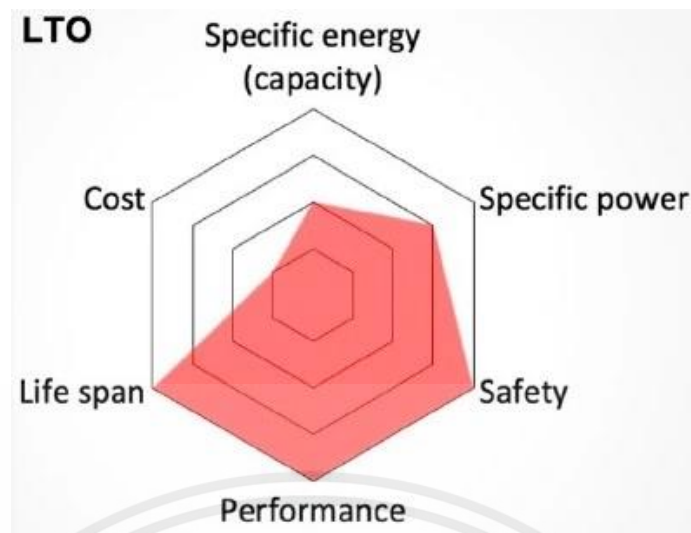
2. ข้อด้อยของแบตเตอรี่ลิเทียมนิเคิลโคบอลต์อลูมิเนียมออกไซด์ NCA

- 2.1 ความปลอดภัยต่ำกว่าชนิดอื่น
- 2.2 ราคาสูง

2.2.1.6 แบตลิเทียมไททานเตต (Lithium Titanate: LTO)

แบตเตอรี่ลิเทียมไททานเตต หรือ LTO ที่เปลี่ยนอาร์โนดจากกราไฟต์เป็นลิเทียมไททานเตต ส่วนคาร์โธดมีทั้งเป็น LMO หรือ LMC ผลคือได้แบตเตอรี่ที่ทั้ง ชาร์จเร็วกว่าแบตเตอรี่ชนิดอื่นๆ อายุการใช้งานนาน และมีความปลอดภัยขึ้น แบตลิเทียมไททานเตต (LTO) มีใช้ใน รถยนต์ EV เวอร์ชันที่มีใช้ภายในประเทศญี่ปุ่นเอง เช่น รถยนต์ไฟฟ้า Mitsubishi i-MiEV , Honda Fit EV , Tosa Electric Concept Bus นอกจากนี้ยังมีใช้ในเครื่องมือทางการทหาร โฟลนโนโซล่าเซลล์ รถ EV สถานีชาร์จรถ EV UPS เครื่องมือสื่อสาร รวมไปถึงอุปกรณ์ทางอวกาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 ภาพแสดงคุณสมบัติแบตเตอรี่ลิเทียมไททาเนต

1. ข้อดีแบตเตอรี่ลิเทียมไททาเนต LTO
 - 1.1 แบตเตอรี่ลิเทียมไททาเนต มีจุดเด่นหลายอย่าง
 - 1.2 ชาร์จเร็ว
 - 1.3 มีช่วงอุณหภูมิใช้งานที่กว้างมาก
 - 1.4 มีอายุการใช้งานนาน
 - 1.5 มีความปลอดภัยและความเสถียรสูงที่สุด
2. ข้อด้อยของแบตเตอรี่ลิเทียมนิเคิลโคบอลต์ออลูมิเนียมออกไซด์ NCA
 - 2.1 มีค่าพลังงานจำเพาะต่ำ แบตเตอรี่จึงมีขนาดใหญ่
 - 2.2 ราคาสูงมาก

2.4 การระบุตำแหน่ง GPS

GPS หรือ global positioning system ในภาษาไทยคือระบบระบุตำแหน่งบนพื้นโลกซึ่งเป็นระบบที่ใช้ดาวเทียมประมาณ 24 ดวงที่โคจรรอบโลกโดยแต่ละดวงมีระยะห่างเท่ากันจากระบบ GPS ทำให้คนบนโลกที่มีเครื่องรับสัญญาณสามารถที่จะทราบพิกัดตำแหน่งของตนเองได้โดยความแม่นยำในการระบุตำแหน่งอยู่ระหว่าง 10 ถึง 100 เมตรในอุปกรณ์รับสัญญาณส่วนใหญ่แต่อุปกรณ์รับสัญญาณ GPS ชนิดพิเศษในกองทัพสามารถรับสัญญาณได้แม่นยำถึง 1 เมตรซึ่งแต่ก่อนการใช้อุปกรณ์รับสัญญาณจะใช้สำหรับงานด้านวิทยาศาสตร์เป็นหลักแต่เนื่องด้วยในปัจจุบันต้นทุนการผลิตตัวรับสัญญาณ GPS มีราคาถูกจึงให้คนทุกกลุ่มสามารถซื้อมาใช้งานส่วนตัวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ GPS ทำงานโดยใช้ดาวเทียม GPS หลักทั้งหมด 21 ดวงดาวเทียมลงอีก 3 ดวงโคจรรอบโลกที่ความสูง 10,600 ไมล์จากผิวโลกดาวเทียมจะเว้นพื้นที่เท่ากันจากมุมโลกและจะมีดาวเทียม 4 ดวงที่เหนือเส้นขอบฟ้า

ดาวเทียมแต่ละดวงจะประกอบไปด้วยคอมพิวเตอร์ที่เวลาในระบบจะไม่หยุดเดินและส่งสัญญาณวิทยุได้ตลอดเวลาจนกว่าดาวเทียมจะหมดอายุดาวเทียมทุกดวงโคจรตามวงโคจรของตนเองและเวลาในดาวเทียมดวงนั้นๆจะถูกตั้งค่าไว้แล้วแต่เวลาก็ยังสามารถเปลี่ยนแปลงได้อย่างต่อเนื่องด้วยอุปกรณ์รับสัญญาณ GPS ที่อยู่บนภาคพื้นดินจะมีคอมพิวเตอร์ที่คำนวณได้แบบ 3 มุมโดยจะสามารถบอกพิกัดจากการรับข้อมูล 3 ใน 4 ดาวเทียมซึ่งได้ตำแหน่งทางภูมิศาสตร์และค่าละติจูดลองจิจูด

อุปกรณ์รับสัญญาณ GPS ที่มีความสามารถเพิ่มเติมสามารถแสดงถึงความสูงของตำแหน่งได้ถ้าเกิดการเคลื่อนที่อุปกรณ์รับสัญญาณ GPS สามารถคำนวณความเร็วทิศทางในการเคลื่อนที่ได้และหากเป็นระบบในอุปกรณ์รับสัญญาณที่มีความสามารถสูงจะสามารถระบุตำแหน่งเป้าหมายที่ต้องการไปและคำนวณเวลาการเดินทางได้สำหรับ GPS ใช้ในงานด้านวิทยาศาสตร์จะหาข้อมูลที่ไม่เคยทราบมาก่อนในระยะทางองศาการเคลื่อนที่ที่แม่นยำที่สุดโดยนักวิทยาศาสตร์จะใช้ GPS ในการวัดการเคลื่อนที่ของแผ่นน้ำแข็งขั้วโลกวัดการเคลื่อนที่ของแผ่นเปลือกโลกวัดความเคลื่อนที่ของภูเขาไฟต่างๆเป็นต้นแต่สำหรับเทคโนโลยีนั้นเป็นของที่เข้าถึงง่ายเพราะทุกคนใช้สมาร์ทโฟนได้มีการฝังตัว GPS ลงไปทำให้บุคคลทั่วไปสามารถซื้อมาครอบครองได้ซึ่งการติดตั้งตัวรับสัญญาณ GPS นั้นในสมาร์ทโฟนมีเป้าหมายเพื่ออำนวยความสะดวกในการนำทางให้กับบุคคลทั่วไปในชีวิตประจำวันโดยสมาร์ทโฟนที่ฝังตัวรับสัญญาณ GPS และมีคำสั่งในการนำทางอยู่แล้วจะเชื่อมต่อกับดาวเทียมเพื่อทราบตำแหน่งปัจจุบัน ก่อนจะนำทางไปยังเป้าหมายที่ถูกระบุไว้

นอกจากการใช้ GPS สำหรับการนำทางแล้วยังมีการใช้ GPS ในงานติดตามเป้าหมายหรือเรียกว่า GPS tracking เพื่อติดตามเป้าหมายหรือสิ่งที่เราต้องการจะติดตามเพื่อให้เราสามารถทราบตำแหน่งความเร็วในการเคลื่อนที่ระยะทาง

2.5 Arduino IDE

Arduino IDE คือโปรแกรมสำหรับใช้เขียนโปรแกรม คอมไพเลอร์ และอัปโหลดโปรแกรมลงบอร์ด Arduino หรือบอร์ดตัวอื่นๆ ที่คล้ายกัน เช่น Generic ESP8266 modules, NodeMCU หรือ WeMos D1 เป็นต้น

2.5.1 แนวคิดการใช้งานโปรแกรม Arduino IDE

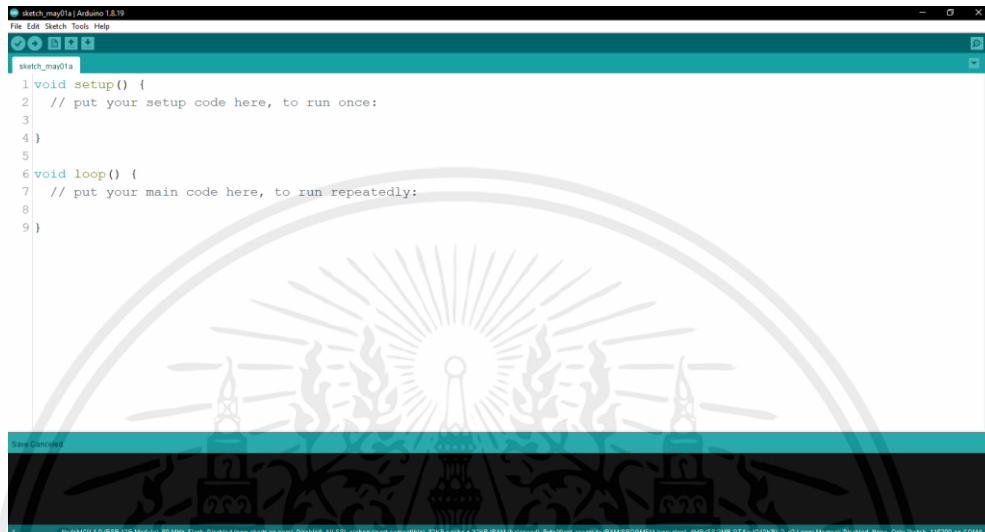
1. ใช้ภาษา C/C++ ในการเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. คอมไพล์ให้เป็นภาษาสำหรับบอร์ด Arduino บันทึกเป็น Intel Hex File
3. อัปโหลด Intel Hex File ลงบนบอร์ด Arduino

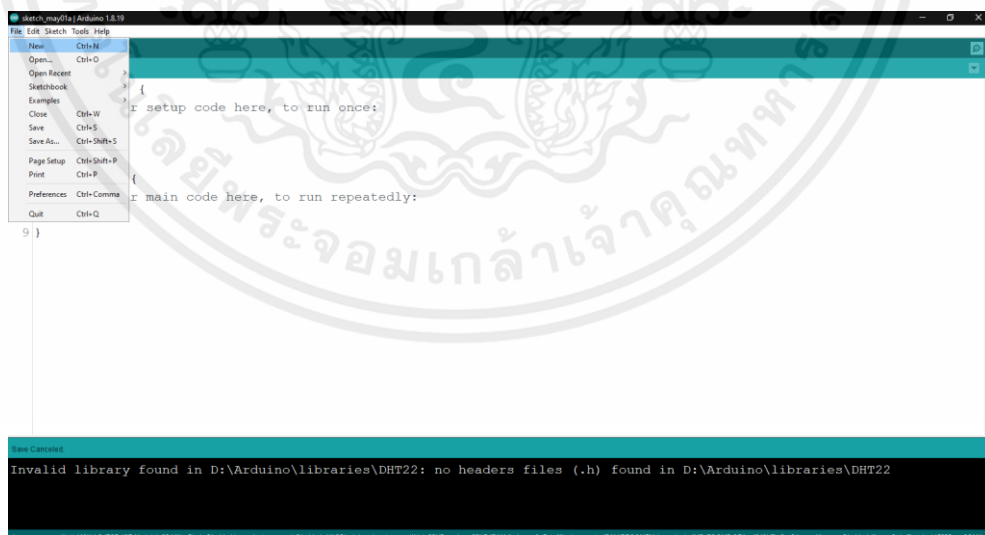
2.5.2 วิธีใช้งานโปรแกรม Arduino IDE

1. เปิดโปรแกรม Arduino IDE ขึ้นมา



รูปที่ 2.11 การเปิดโปรแกรม Arduino IDE

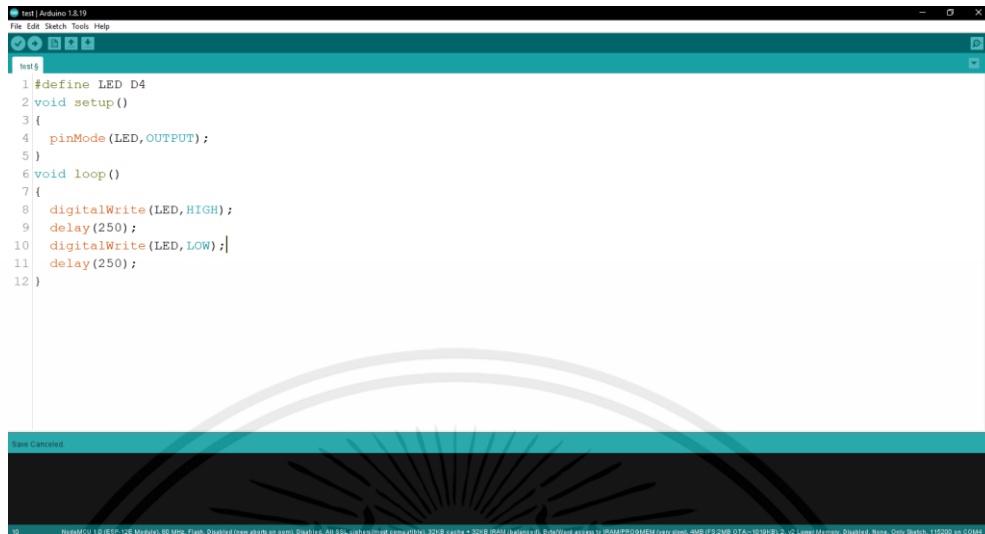
2. สร้าง Sketch ใหม่ โดยคลิกที่เมนู File > New



รูปที่ 2.12 การสร้างไฟล์ใหม่โปรแกรม Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เขียนโปรแกรมลงไปตรงพื้นที่สีขาว การเขียนโปรแกรมสำหรับ Arduino จะมีฟังก์ชัน setup และ ฟังก์ชัน loop



```

1 #define LED D4
2 void setup()
3 {
4   pinMode(LED, OUTPUT);
5 }
6 void loop()
7 {
8   digitalWrite(LED, HIGH);
9   delay(250);
10  digitalWrite(LED, LOW);
11  delay(250);
12 }

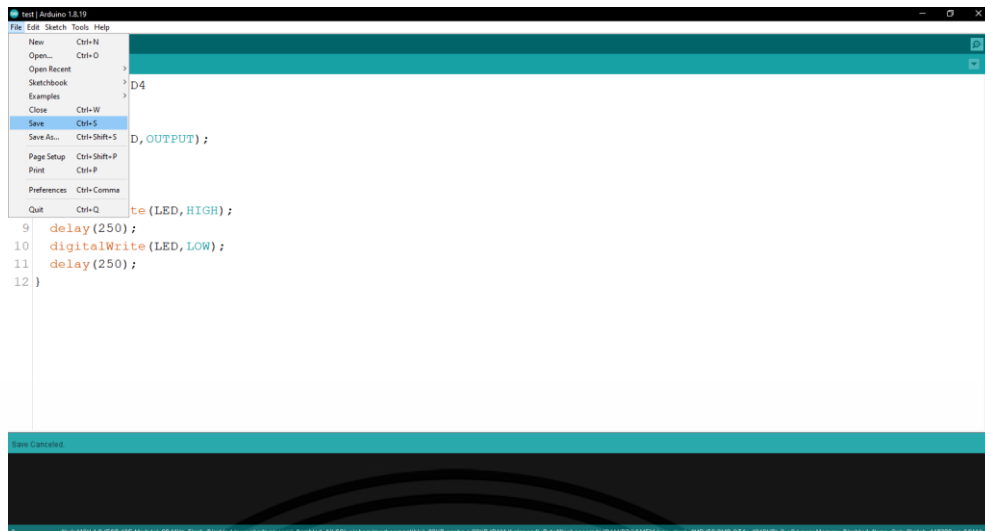
```

รูปที่ 2.13 การเขียน code ในโปรแกรม Arduino IDE

ฟังก์ชัน setup จะเริ่มทำงานเป็นอันดับแรกเมื่อ Arduino เริ่มทำงาน และคำสั่งที่ถูกเขียนลงไป
ไปในนี้ จะทำงานเพียงครั้งเดียวเท่านั้น

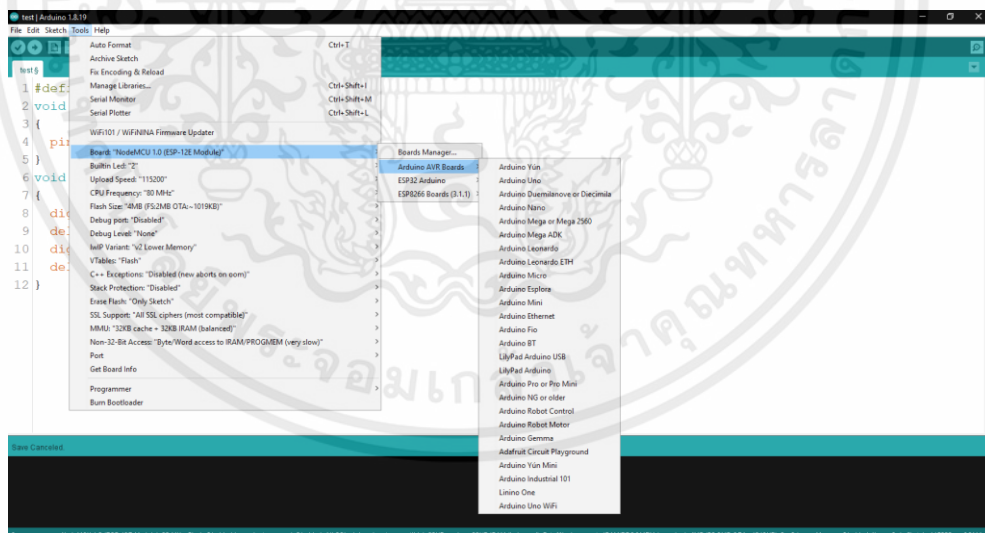
ฟังก์ชัน loop จะเริ่มทำงานทันทีเมื่อฟังก์ชัน setup ทำงานเสร็จ และคำสั่งที่ถูกเขียนลงไป
ในนี้ จะทำงานตั้งแต่คำสั่งแรก ไล่ไปจนถึงคำสั่งสุดท้าย แล้วกลับมาที่คำสั่งแรก ไล่ไปจนถึงคำสั่ง
สุดท้าย วนซ้ำแบบนี้ไปเรื่อย ๆ ตลอดการทำงานของ Arduino

4. เมื่อเขียนโปรแกรมเสร็จแล้วให้บันทึกเก็บไว้โดยคลิกที่เมนู File > Save จากนั้นจะมี
หน้าต่างขึ้นมาให้เราเลือกว่าจะบันทึกไว้ที่ไหน และจะตั้งชื่อ Sketch ว่ายังไง ตรงนี้ก็แล้วแต่เราจะตั้ง
ชื่อครับ



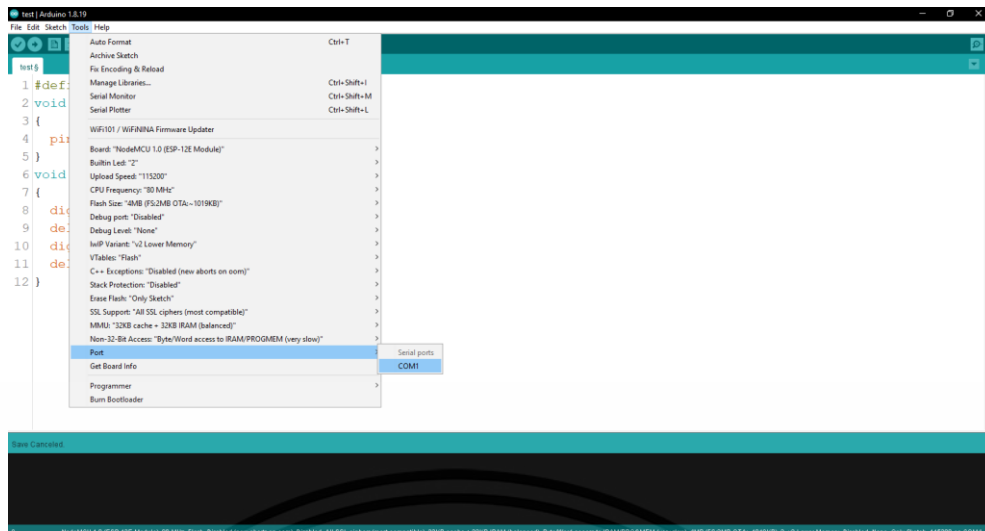
รูปที่ 2.14 การบันทึกไฟล์โปรแกรม Arduino IDE

5. ต่อบอร์ด Arduino เข้ากับคอมพิวเตอร์ผ่านสาย USB (แนะนำให้ต่อครั้งละ 1 บอร์ด ป้องกันการสับสน)
6. ตั้งค่าเพื่อบอก Arduino IDE ว่าเราจะอัปโหลดโปรแกรมให้กับบอร์ดรุ่นอะไร และอัปโหลดผ่าน Port ไหน



รูปที่ 2.15 การเลือกรุ่นของบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 การเลือก Port ของบอร์ดที่เชื่อมต่อ

7. คลิกปุ่มอัปโหลด หรือจะคลิกที่เมนู Sketch > Upload



รูปที่ 2.17 การอัปโหลด code ลงบอร์ด

2.6 Cayenne

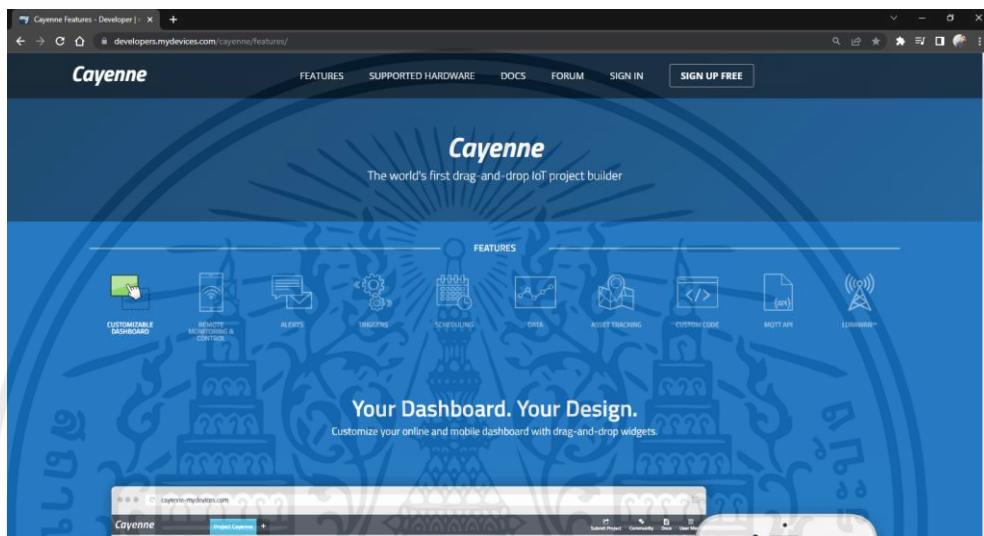
Cayenne เป็นแพลตฟอร์มที่เป็นแอปพลิเคชันและเว็บไซต์เพื่อควบคุมอุปกรณ์บนระบบอินเทอร์เน็ต ซึ่งเป็นแผงควบคุมระบบดิจิทัลที่ผู้ใช้สามารถสร้างส่วนต่อประสานกราฟิกสำหรับโครงการของผู้ใช้โดยการลากและวางเครื่องมือ (widgets) ที่มีให้เลือกอยู่หลากหลาย เป็นเรื่องที่ยากมากในการตั้งค่าทุกอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Cayenne ไม่ได้ผูกติดอยู่กับบอร์ดหรืออุปกรณ์เสริมบางตัว แต่จะสนับสนุนฮาร์ดแวร์ที่นักพัฒนาเลือกไม่ว่าจะเป็นอุปกรณ์ใดจะเชื่อมโยงกับอินเทอร์เน็ตผ่านสัญญาณอินเทอร์เน็ตไร้สาย (Wi-Fi) หรือแบบมีสาย จะช่วยให้อุปกรณ์ของนักพัฒนาออนไลน์และพร้อมสำหรับ Internet of Things (IoT)

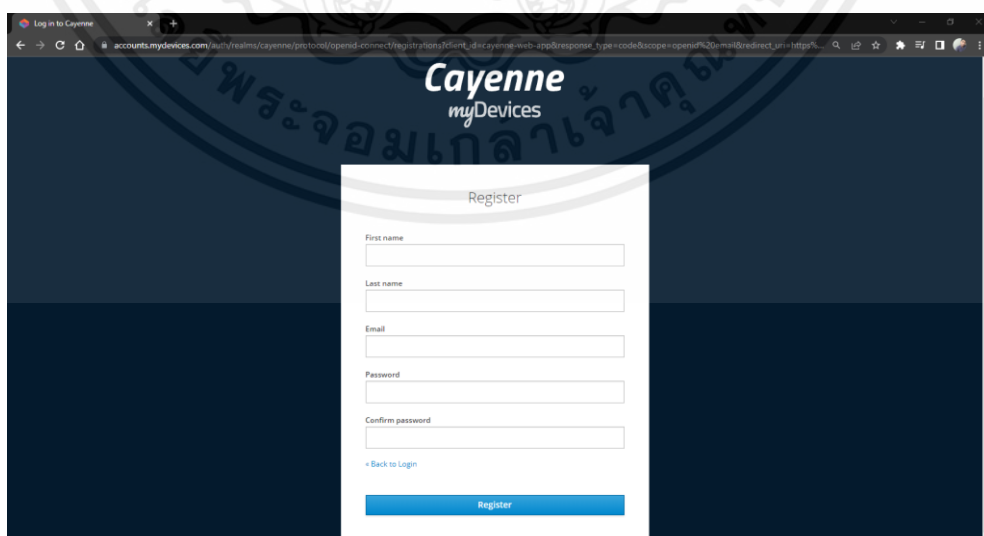
2.6.1 การใช้งาน Cayenne บนเว็บไซต์

1. เข้าสู่เว็บไซต์ Cayenne



รูปที่ 2.18 หน้าเว็บไซต์ Cayenne

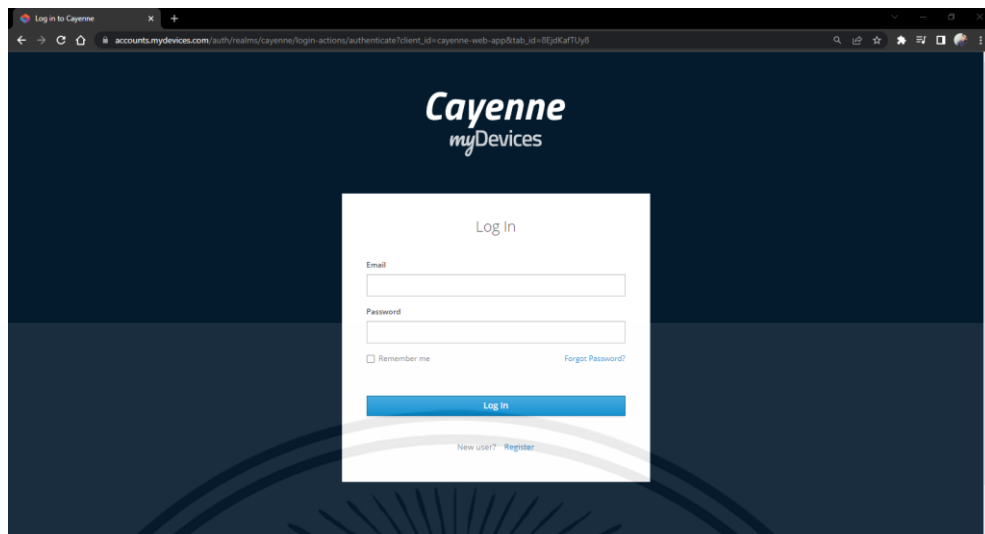
2. ลงทะเบียน



รูปที่ 2.19 หน้าลงทะเบียนเว็บไซต์ Cayenne

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ล็อกอินเข้าสู่ระบบ



รูปที่ 2.20 หน้าเข้าสู่ระบบเว็บไซต์ Cayenne

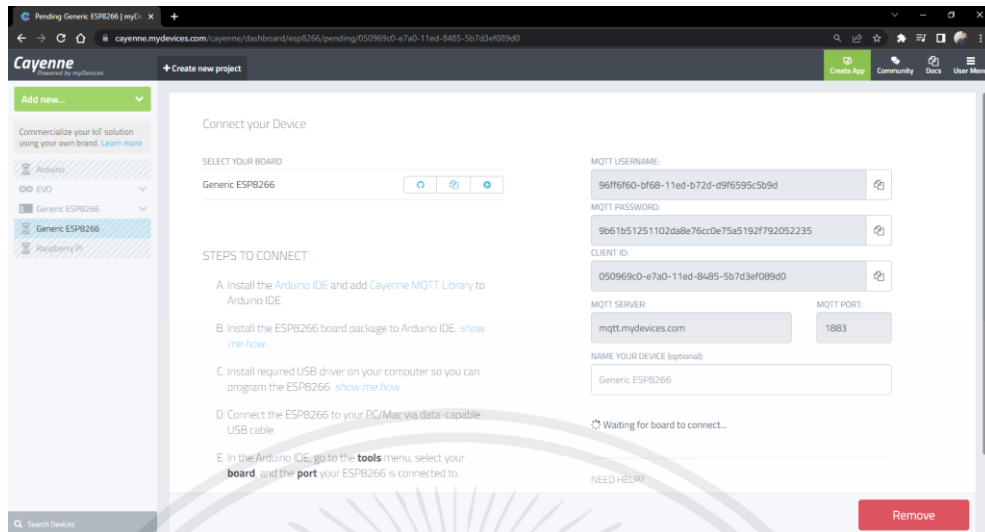
4. เลือกอุปกรณ์ที่ใช้



รูปที่ 2.21 ภาพแสดงหน้าเลือกอุปกรณ์เว็บไซต์ Cayenne

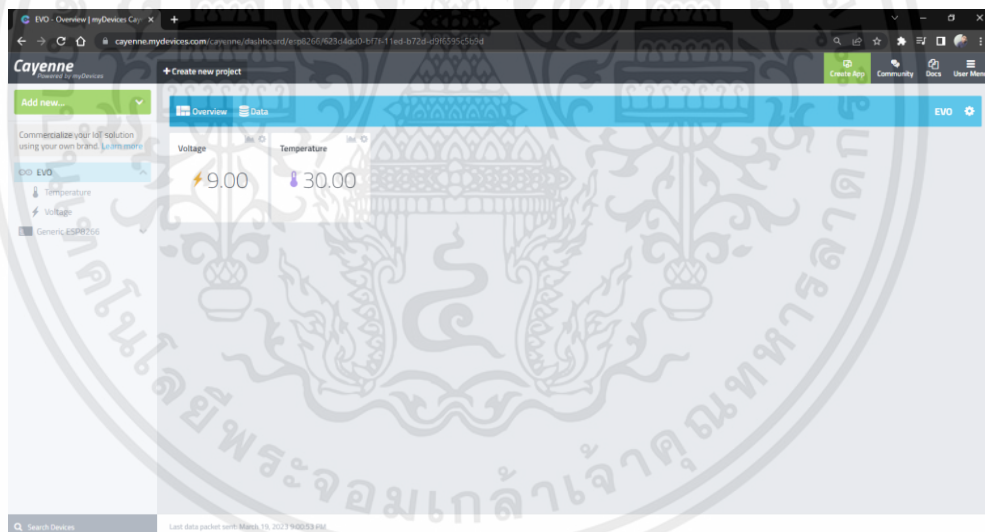
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เชื่อมต่ออุปกรณ์เข้ากับ Cayenne



รูปที่ 2.22 หน้าเชื่อมต่ออุปกรณ์เว็บไซต์ Cayenne

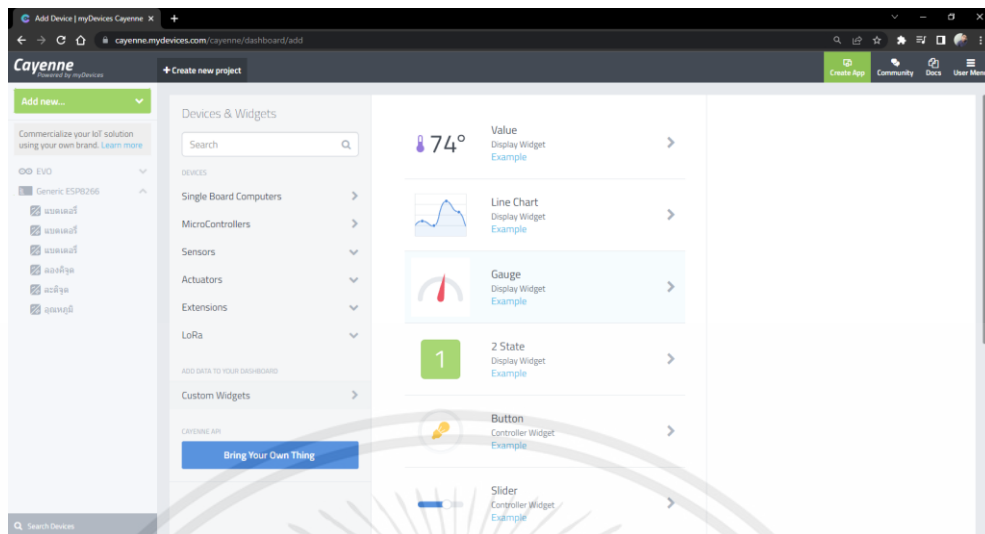
6. รอเชื่อมต่ออุปกรณ์เสร็จสมบูรณ์



รูปที่ 2.23 หน้าเชื่อมต่ออุปกรณ์เสร็จสมบูรณ์

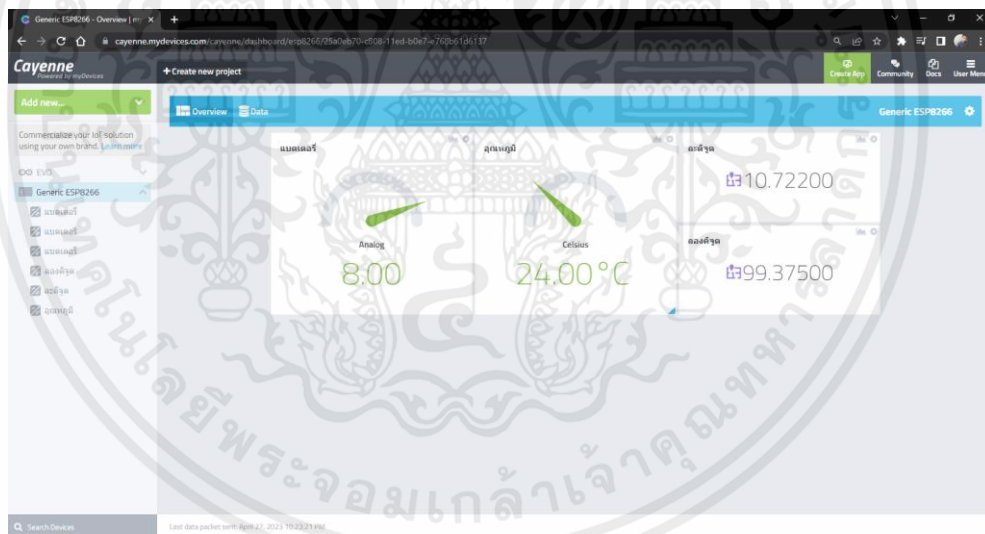
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เพิ่ม Widgets



รูปที่ 2.24 หน้าเพิ่ม Widgets

8. หน้า Overview แสดงข้อมูลแบบ Realtime



รูปที่ 2.25 หน้า Overview

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. หน้า Data แสดงข้อมูลแบบตาราง

Timestamp	Device No.	Channel	Sensor Name	Sensor ID	Data Type	Unit	Values
2023-04-27 10:23:21	Generic ESP8266	1	อุณหภูมิ	9fcd7800-cae4-11ed-b726-df...			8
2023-04-27 10:23:21	Generic ESP8266	2	แสงสีจิม	1af55980-cae4-11ed-b726-df...			10.72200012207
2023-04-27 10:23:21	Generic ESP8266	3	แสงสีจิม	4307d9d0-cae4-11ed-b726-df...			99.375
2023-04-27 10:23:21	Generic ESP8266	0	อุณหภูมิ	f42ae520-cae4-11ed-b726-df...	temp	c	24
2023-04-27 10:23:06	Generic ESP8266	1	อุณหภูมิ	9fcd7800-cae4-11ed-b726-df...			8
2023-04-27 10:23:06	Generic ESP8266	2	แสงสีจิม	1af55980-cae4-11ed-b726-df...			10.72200012207
2023-04-27 10:23:06	Generic ESP8266	3	แสงสีจิม	4307d9d0-cae4-11ed-b726-df...			99.375
2023-04-27 10:23:06	Generic ESP8266	0	อุณหภูมิ	f42ae520-cae4-11ed-b726-df...	temp	c	37
2023-04-27 10:22:51	Generic ESP8266	3	แสงสีจิม	4307d9d0-cae4-11ed-b726-df...			99.375
2023-04-27 10:22:51	Generic ESP8266	1	อุณหภูมิ	9fcd7800-cae4-11ed-b726-df...			5
2023-04-27 10:22:51	Generic ESP8266	2	แสงสีจิม	1af55980-cae4-11ed-b726-df...			10.72200012207
2023-04-27 10:22:50	Generic ESP8266	0	อุณหภูมิ	f42ae520-cae4-11ed-b726-df...	temp	c	33
2023-04-27 10:22:35	Generic ESP8266	3	แสงสีจิม	4307d9d0-cae4-11ed-b726-df...			99.375
2023-04-27 10:22:35	Generic ESP8266	2	แสงสีจิม	1af55980-cae4-11ed-b726-df...			10.72200012207
2023-04-27 10:22:35	Generic ESP8266	1	อุณหภูมิ	9fcd7800-cae4-11ed-b726-df...			8
2023-04-27 10:22:35	Generic ESP8266	0	อุณหภูมิ	f42ae520-cae4-11ed-b726-df...	temp	c	25
2023-04-27 10:22:30	Generic ESP8266	2	แสงสีจิม	1af55980-cae4-11ed-b726-df...			10.72200012207
2023-04-27 10:22:30	Generic ESP8266	3	แสงสีจิม	4307d9d0-cae4-11ed-b726-df...			99.375
2023-04-27 10:22:20	Generic ESP8266	1	อุณหภูมิ	9fcd7800-cae4-11ed-b726-df...			2
2023-04-27 10:22:20	Generic ESP8266	0	อุณหภูมิ	f42ae520-cae4-11ed-b726-df...	temp	c	47

รูปที่ 2.26 หน้า Data

2.6.2 การใช้งาน Cayenne บนแอปพลิเคชัน

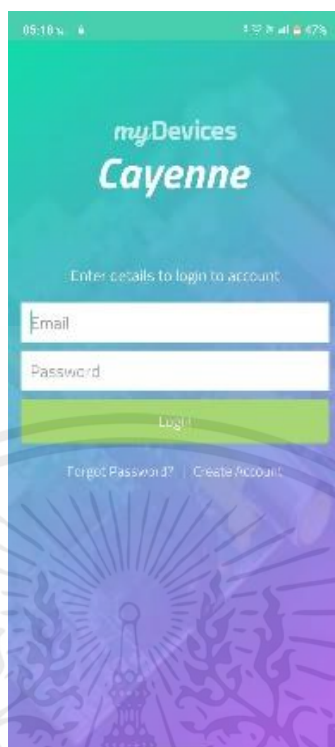
1. เข้าสู่แอปพลิเคชัน Cayenne



รูปที่ 2.27 แอปพลิเคชัน Cayenne

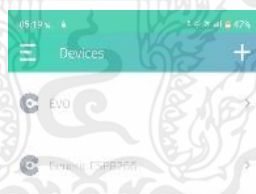
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เข้าสู่ระบบแอปพลิเคชัน Cayenne



รูปที่ 2.28 การเข้าสู่ระบบแอปพลิเคชัน Cayenne

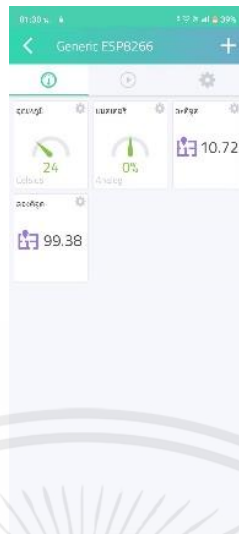
3. เลือกอุปกรณ์



รูปที่ 2.29 หน้าเลือกอุปกรณ์แอปพลิเคชัน Cayenne

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. หน้า Overview



รูปที่ 2.30 หน้า Overview อุปกรณ์แอปพลิเคชัน Cayenne

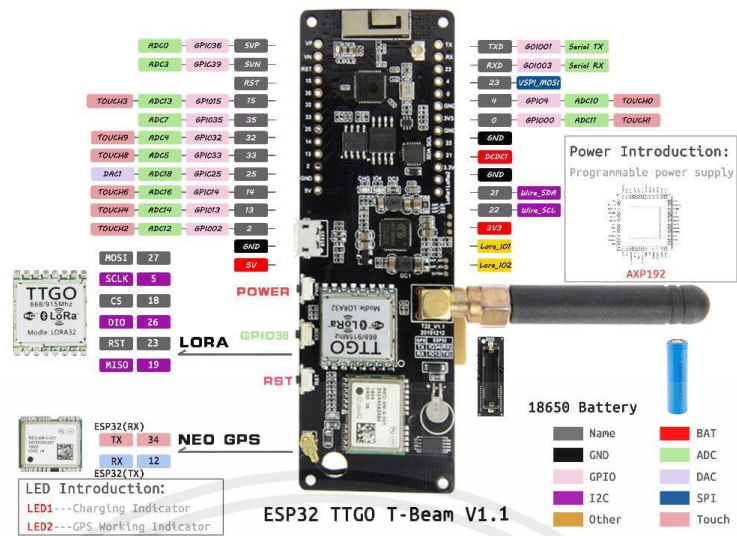
2.7 TTGO t-beam esp32 lora

เป็นชุดพัฒนาไมโครคอนโทรลเลอร์ในตระกูล ESP32 + LoRa SX1276 โดยใช้ Module ของ TTGO รุ่น WIFI LoRa32 V2 มีส่วนแสดงผล OLED(128x64) ขนาด 0.96 นิ้ว ออกแบบรองรับการพัฒนาเป็น Node Station มี WIFI 2.4G Bluetooth และ Lora รองรับย่านความถี่ 923 MHZ แรงดันไฟฟ้าที่ใช้งาน 3.3 V - 7 V



รูปที่ 2.31 TTGO t-beam esp32 loRa

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



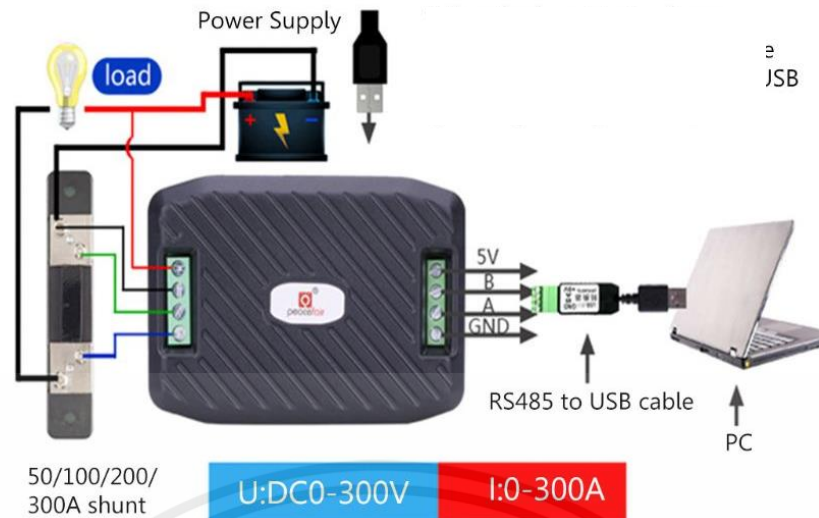
รูปที่ 2.32 Pinout Diagram TTGO t-beam esp32 loRa

2.8 Pzem-017



รูปที่ 2.33 Pzem-017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.34 การต่อใช้งาน Pzem-017

โมดูลนี้ใช้เป็นหลักในการวัดแรงดัน DC, กระแส, กำลังไฟฟ้าที่ใช้งาน, ความถี่และการใช้พลังงาน โมดูลนี้ไม่มีฟังก์ชันการแสดงผล ข้อมูลจะถูกอ่านผ่านอินเทอร์เฟซ RS485 [7]

2.8.1 คุณสมบัติ

ส่วนใหญ่มักจะใช้สำหรับการวัดแรงดัน DC กระแสกำลังไฟฟ้าที่ใช้งานอยู่ ความถี่และการใช้พลังงาน ช่วงการวัดปัจจุบันคือ 50A, 100A, 200A, 300A ขึ้นอยู่กับข้อกำหนดการใช้อินเทอร์เฟซการสื่อสาร UART ถึง RS485 อัตราการรับส่งข้อมูลคือ 9600, 8 บิตข้อมูล, 2 บิตแอปพลิเคชันใช้โปรโตคอล ModbusRTU ในการสื่อสาร รองรับเฉพาะรหัสฟังก์ชันเช่น 0x03, 0x04, 0x06, 0x41, 0x42 เป็นต้น สามารถตั้งค่าเกณฑ์แรงดันไฟฟ้า แบ่งเป็นแรงดันไฟฟ้าสูงและเกณฑ์แรงดันไฟฟ้าต่ำ เมื่อแรงดันไฟฟ้าที่วัดได้เกินเกณฑ์ สามารถแจ้งเตือนได้

2.8.2 ข้อมูลจำเพาะ

1. ยี่ห้อ: PEACEFAIR รุ่น: PZEM-017
2. แรงดันไฟฟ้า ช่วงการวัด 0.05-300 V ความละเอียด 0.01 V
3. กระแสไฟฟ้า ช่วงการวัด 0.02-300 A ความละเอียด 0.01 A
4. กำลังงาน ช่วงการวัด 0.2-90 kW ความละเอียด 0.1 W
5. การใช้พลังงาน ช่วงการวัด 0-9999 kWh ความละเอียด 1 Wh
6. รีเซ็ตพลังงาน ใช้ซอฟต์แวร์เพื่อรีเซ็ตสัญญาณเตือนแรงดันไฟเกิน 300 V
7. อุณหภูมิในการทำงาน -20~60 °C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ขนาดผลิตภัณฑ์ ประมาณ 9*6*2.3 ซม.

9. น้ำหนัก: ประมาณ. 77 กรัม ~ 287 กรัม

2.9 เซ็นเซอร์วัดอุณหภูมิ DHT11

DHT11 เป็นโมดูลวัดอุณหภูมิและความชื้นราคาถูก ใช้งานง่าย มีไลบรารีพร้อมใช้ เหมาะสำหรับการใช้ในงานวัดอุณหภูมิและความชื้นทั่วไป สามารถใช้ไฟได้ 3-5V อ่านค่าอุณหภูมิได้ตั้งแต่ 0 ถึง 50°C และความชื้น 20% ถึง 90%

2.9.1 คุณสมบัติ

1. วัดความชื้นแสดงผลเป็นดิจิตอล
2. สามารถป้องกันสัญญาณรบกวนได้อย่างมีประสิทธิภาพ
3. เมื่อตรวจวัดความชื้นแล้ว สามารถเทียบกับอุณหภูมิได้

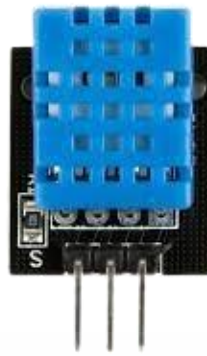
2.9.2 การใช้งานและการพัฒนา

1. เครื่องวัดความชื้นและอุณหภูมิ
2. นำไปติดตั้งในที่ที่ต้องการวัดอุณหภูมิและความชื้น

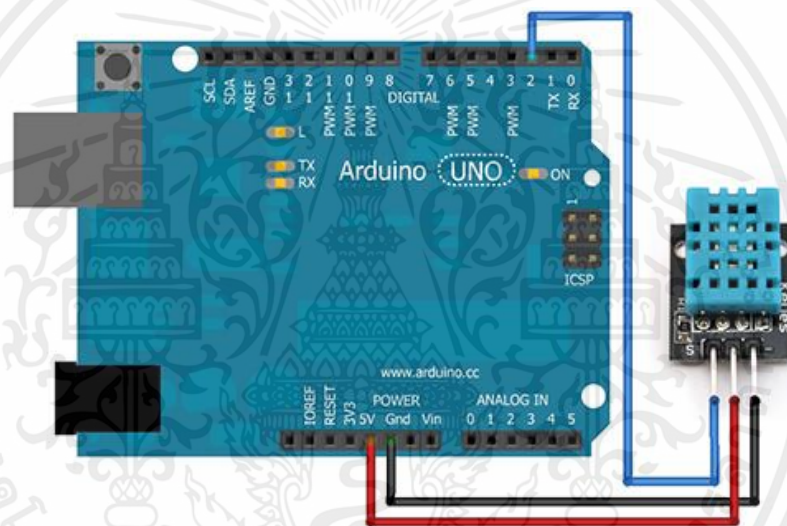
2.9.3 ตารางการเชื่อมต่อ

ตารางที่ 2.2 ตารางการเชื่อมต่อเซ็นเซอร์ DHT11

Pin	Name	Description
1	VDD	Power (3.5V-5.5V)
2	SDA	Serial data bidirectional
3	GND	Ground



รูปที่ 2.35 เซ็นเซอร์ DHT11

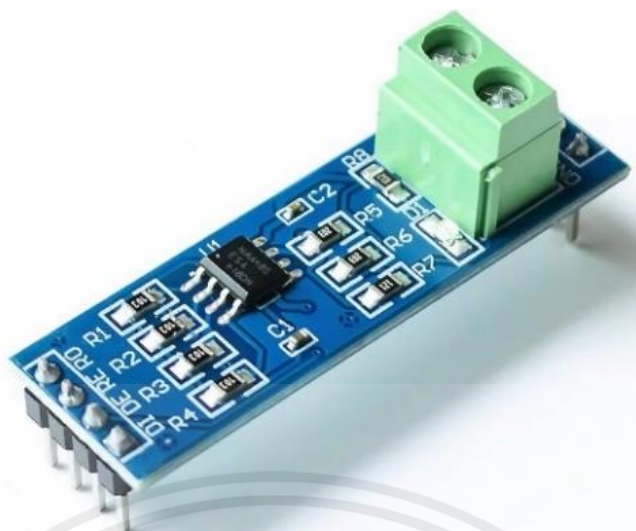


รูปที่ 2.36 การต่อใช้งานเซ็นเซอร์ DHT11

2.10 MAX485

MAX485 TTL to RS485 Converter Module โมดูลส่งข้อมูลระยะไกลแบบใช้สาย RS485 สามารถลากสายยาวได้ถึง 1.2 กิโลเมตร ใช้พลังงานต่ำมาก สามารถเซ็ตให้เป็น ตัวรับ (slave) หรือ ตัวส่ง (master) ก็ได้ การเชื่อมต่อสายง่ายเพียงแค่นำสายสัญญาณต่อเข้ากับ 2P Terminal ก็สามารถใช้งานได้ทันที รองรับไมโครคอนโทรลเลอร์ Arduino ทำงานที่แรงดัน DC 5 Volt

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.37 เซ็นเซอร์ MAX485

2.11 MODBUS

คือ โพรโทคอลหรือรูปแบบการสื่อสารข้อมูลดิจิทัลแบบอนุกรมรูปแบบหนึ่ง ซึ่งถูกเผยแพร่ครั้งแรกในปี ค.ศ. 1979 โดย Modicon ซึ่งปัจจุบันคือบริษัท Schneider Electric เพื่อใช้ร่วมกับ PLC (Programmable Logic Controllers) ซึ่งทางบริษัทได้เปิดให้ MODBUS เป็น Open Protocol หรือก็คือผู้สนใจสามารถนำโพรโทคอลนี้ไปใช้หรือพัฒนาได้โดยไม่มีค่าใช้จ่ายใดๆ ตั้งแต่นั้นเป็นต้นมา MODBUS จึงกลายเป็นโพรโทคอลที่ได้รับความนิยมและถูกใช้เป็นโพรโทคอลมาตรฐานในระบบอิเล็กทรอนิกส์อุตสาหกรรมจนถึงปัจจุบัน

ปัจจุบันหากพูดถึง MODBUS หลายคนคงนึกถึง RS485 เพราะเครื่องมือวัดอุตสาหกรรมส่วนใหญ่มักจะใช้ RS485 แบบ MODBUS protocol ในการส่งสัญญาณ ซึ่งความจริงแล้ว MODBUS เป็นแค่เพียงโพรโทคอลหรือภาษาที่ใช้สื่อสารในคอมพิวเตอร์เท่านั้น ส่วน RS485 จะเป็นวิธีการส่งข้อมูลซึ่งจะพูดถึงลักษณะของ Hardware ลักษณะการเดินสายไฟ รูปแบบการเชื่อมต่อเครือข่าย เป็นต้น และอีกหนึ่งความจริงที่หลายคนยังไม่ทราบคือ MODBUS สามารถประยุกต์ใช้กับการสื่อสารแบบอนุกรมลักษณะอื่นได้ ไม่ว่าจะเป็น RS232 RS422 หรือ RS423 เป็นต้น ไม่ได้ถูกจำกัดการใช้งานเฉพาะ RS485 อย่างเดียว แบบที่หลายๆคนคิด

2.12 GY-NEO6MV2 Ublox NEO-6M GPS Module

โมดูล GPS GY-NEO6MV2 เป็นโมดูล U-blox รุ่น NEO-6M ซึ่งเป็นที่นิยมกันอย่างมาก สามารถเชื่อมต่อได้กับไมโครคอนโทรลเลอร์หลายหลายประเภทไม่ว่าจะเป็น Arduino, NodeMCU,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Raspberry Pi, ESP32 ผ่านทาง Serial UART ความเร็วที่ 9600 (สามารถอัปเดต) และตำแหน่ง อัปเดตตลอดทุก 1 วินาทีสามารถตั้งค่าให้เร็วกว่านี้ได้ การทำงานเมื่อตัวโมดูลจับสัญญาณได้จะขึ้นไฟ สีเขียวกระพริบ ตัวโมดูลมีแบตเตอรี่เก็บตำแหน่งล่าสุดและคอนฟิกต่างๆ [6]



รูปที่ 2.38 GY-NEO6MV2 Ublox NEO-6M GPS Module



รูปที่ 2.39 การต่อใช้งาน GY-NEO6MV2 Ublox NEO-6M GPS Module

2.13 CAT LoRa IoT Platform

คือ ระบบบริหารจัดการการเชื่อมต่ออุปกรณ์ IoT โดยอาศัย LoRa Network เพื่อให้อุปกรณ์ สามารถเชื่อมต่อกับ Application ได้ผ่านโปรโตคอลที่ได้รับความนิยม เช่น HTTP และ MQTT ซึ่งจะ ช่วยให้ผู้ใช้สามารถพัฒนา Application เกี่ยวกับ IoT ได้ตามความต้องการของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. หน้าล็อกอินเข้าสู่ระบบ

[Resend Verify Email](#) [User Guide](#)

รูปที่ 2.40 เข้าสู่ระบบเว็บไซต์ CAT LoRaIoT Platform

2. หน้าลงทะเบียน

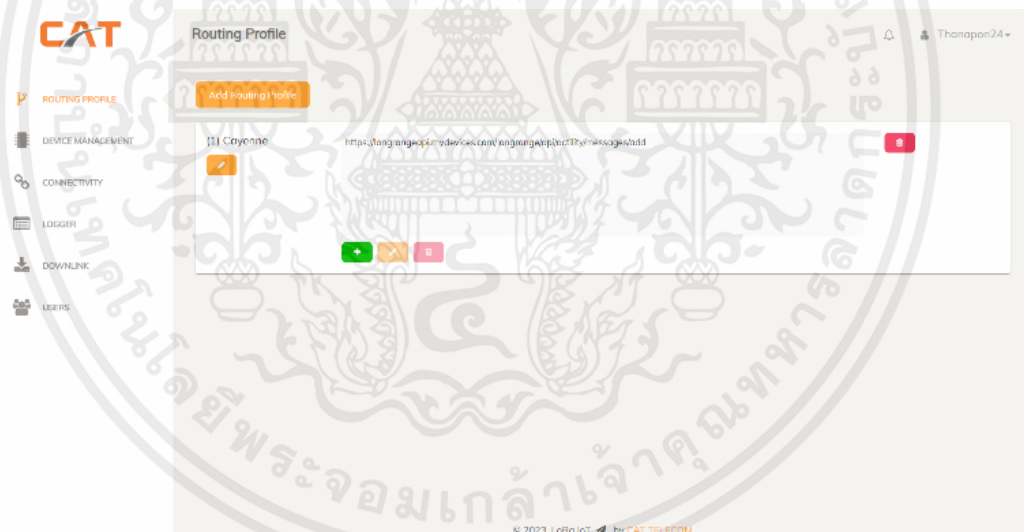
รูปที่ 2.41 การลงทะเบียนก่อนเข้าใช้งานเว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เพิ่มอุปกรณ์

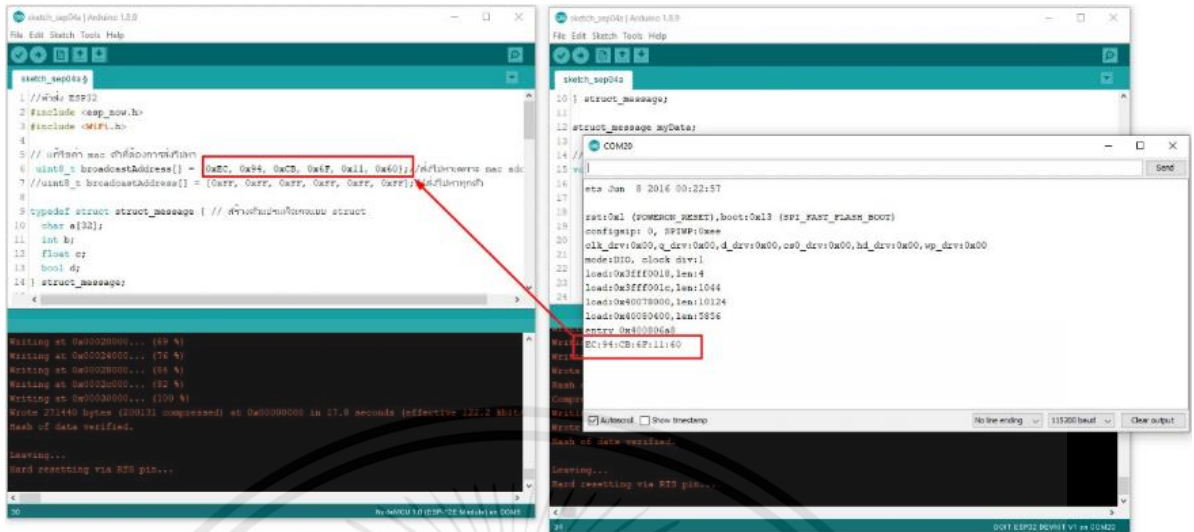
รูปที่ 2.42 การเพิ่มอุปกรณ์

4. หน้าการ Routing Profile

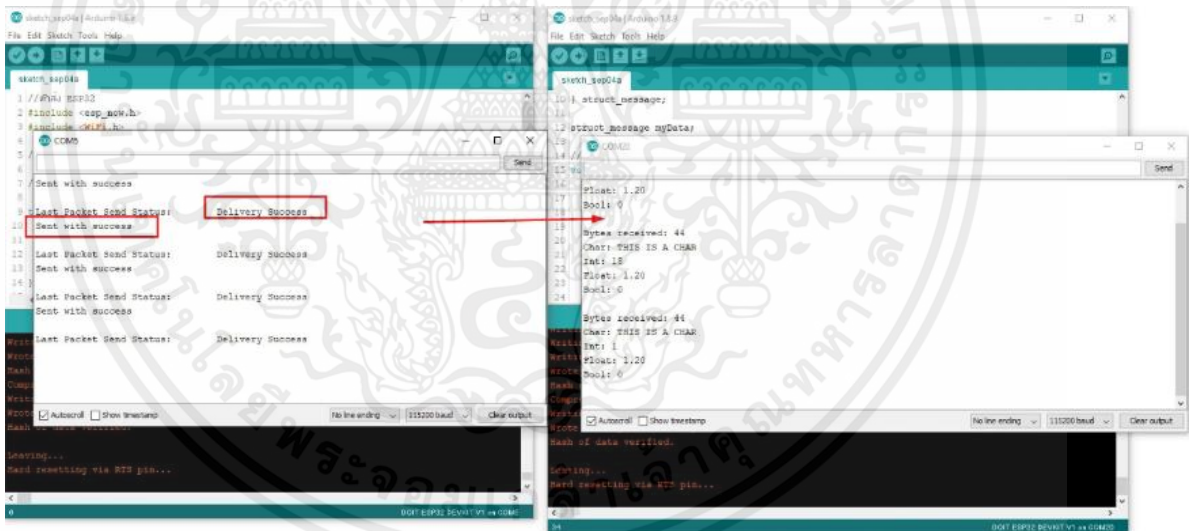


รูปที่ 2.43 การ Routing Profile โดยใช้ URL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.45 การเชื่อมต่อระหว่างบอร์ดโดยใช้ mac address



รูปที่ 2.46 การส่งข้อมูลไปยังบอร์ดอีกตัว

2.15 Received Signal Strength Indicator (RSSI)

RSSI เป็นคำย่อของคำว่า Received Signal Strength Indicator ซึ่งคือระดับบ่งชี้ของความแรงของคลื่นความถี่วิทยุที่วัดที่ ตัวรับ โดยที่ระดับความแรงของสัญญาณวิทยุจะมีหน่วยที่ใช้กันส่วนใหญ่คือ dBm หรือ decibel-milliwatts ที่อ้างอิงกับที่ ระดับ milliwatt 0 dBm จะมีค่าเท่ากับ 1 milliwatt ซึ่งคำนวณได้จากสูตรดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$x = 10 \log_{10} \frac{P}{1 \text{ mW}}, \quad (2.1)$$

$$P = 1 \text{ mW} \cdot 10^{\frac{x}{10}}$$

สมการหาค่าความแรงสัญญาณ

โดยที่ x คือ ระดับ RSSI มีหน่วยเป็น dBm และ P คือกำลังมีหน่วยเป็น watt

2.16 Signal-to-Noise Ratio (SNR)

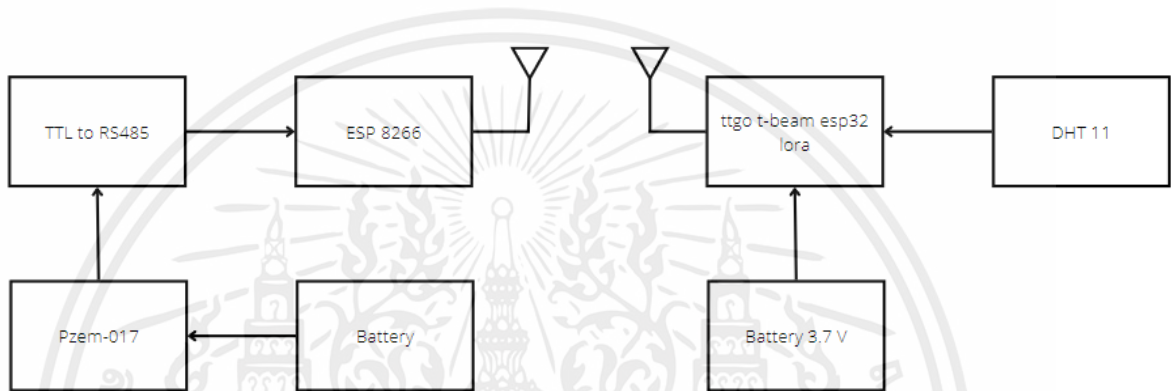
สัญญาณต่อเสียงรบกวนอัตราส่วน (SNR หรือ S/N) เป็นตัวชี้วัดที่ใช้ในการวิทยาศาสตร์และวิศวกรรมที่เปรียบเทียบระดับของที่ต้องการส่งสัญญาณไปถึงระดับของพื้นหลังเสียง SNR ถูกกำหนดเป็นอัตราส่วนของสัญญาณไฟฟ้าไปที่พลังเสียงมักจะแสดงออกในเดซิเบล อัตราส่วนที่สูงกว่า 1 : 1 (มากกว่า 0 dB) หมายถึง สัญญาณมากกว่าสัญญาณรบกวน

บทที่ 3

การออกแบบ

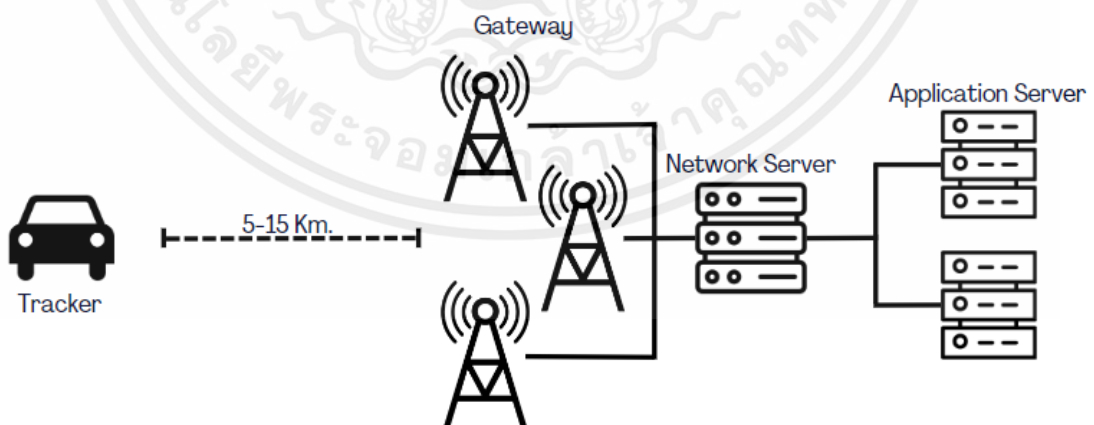
3.1 Block Diagram

3.1.1 Block Diagram แสดงการเชื่อมต่ออุปกรณ์



รูปที่ 3.1 Block Diagram แสดงการเชื่อมต่ออุปกรณ์

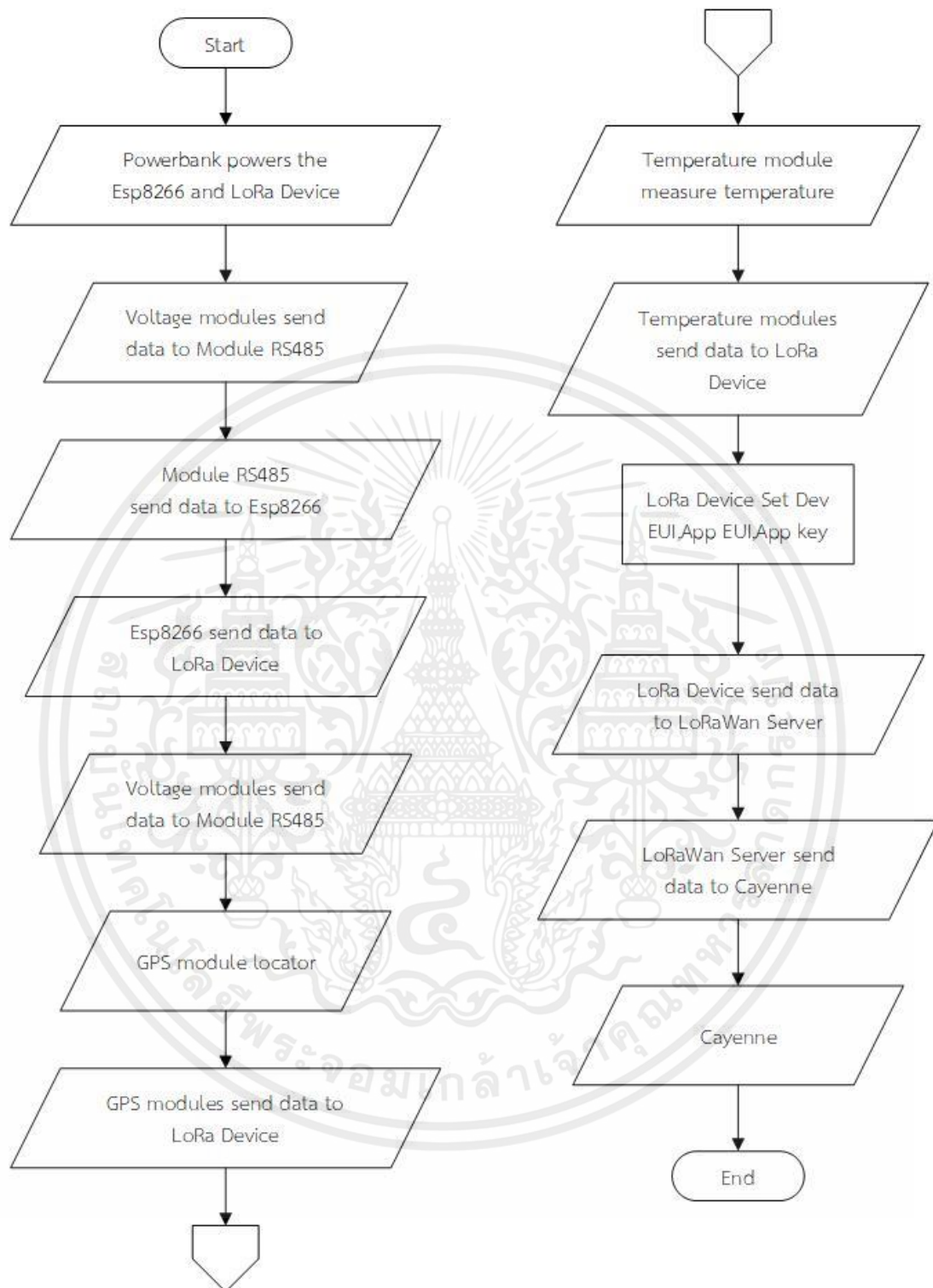
3.1.2 Block Diagram แสดงการทำงาน LoRa



รูปที่ 3.2 Block Diagram แสดงการทำงาน LoRa

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ผังการทำงานของระบบ

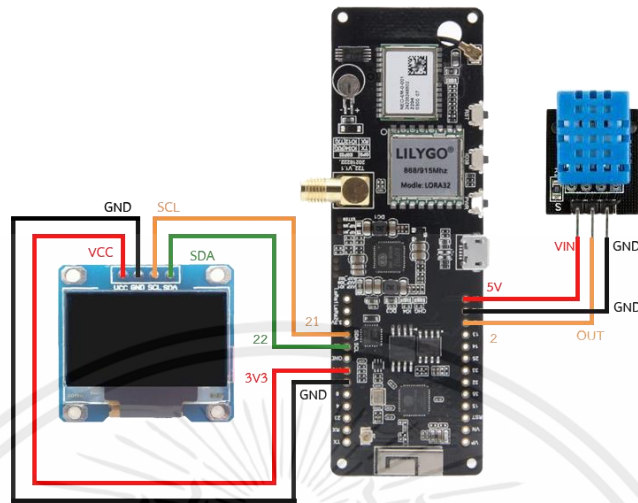


รูปที่ 3.3 Flow chart

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

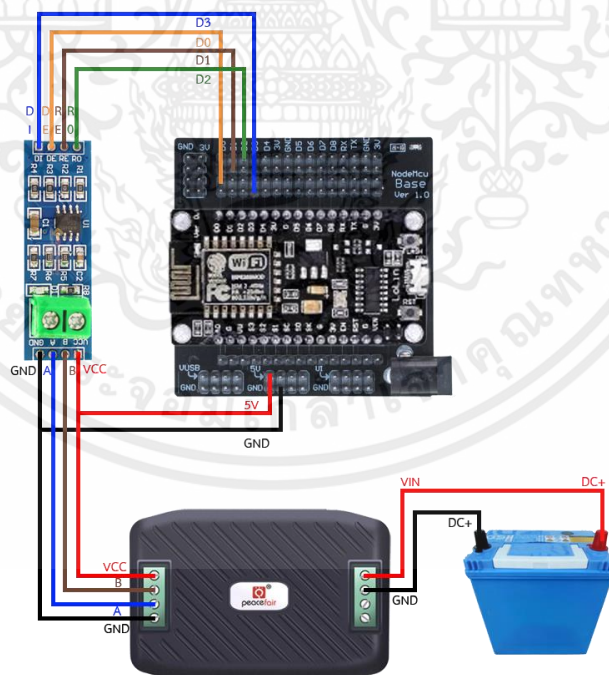
3.3 การต่อวงจร

3.3.1 วงจร TTGO t-beam esp32 lora



รูปที่ 3.4 การต่อวงจร TTGO t-beam esp32 lora

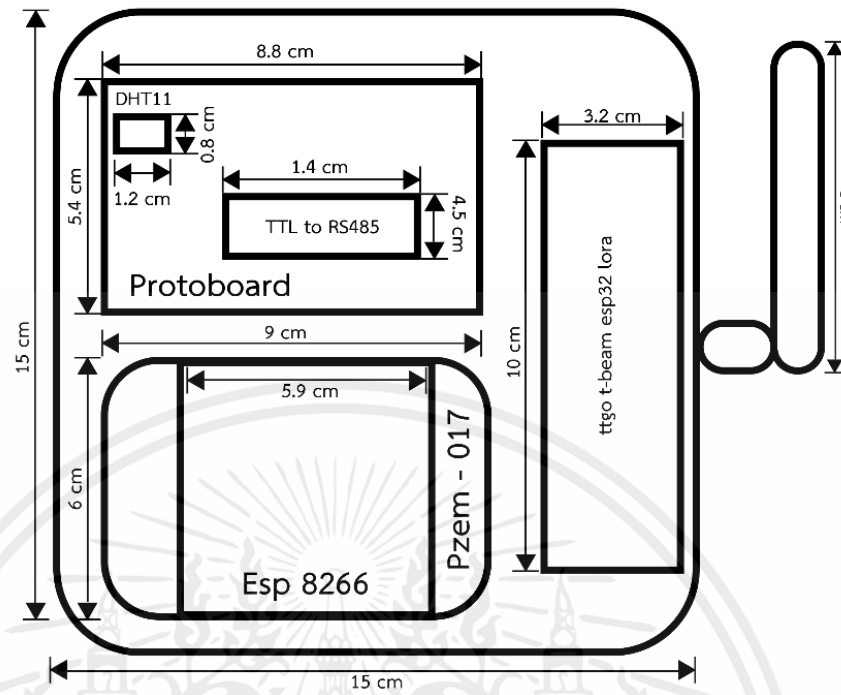
3.3.2 วงจร ESP 8266



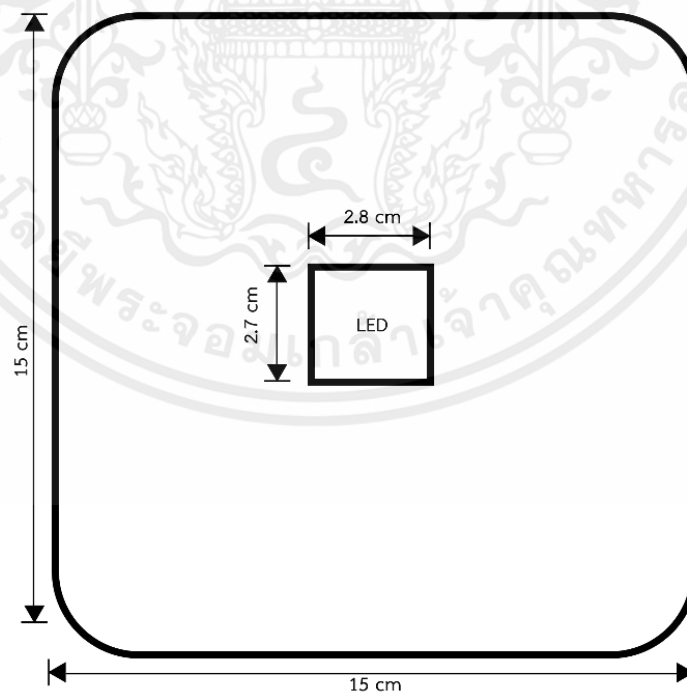
รูปที่ 3.5 การต่อวงจร ESP8266

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การออกแบบกล่องเซนเซอร์



รูปที่ 3.6 โครงสร้างภายในกล่อง



รูปที่ 3.7 โครงสร้างฝาปิดกล่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การติดตั้งอุปกรณ์จริง



รูปที่ 3.8 รถกอล์ฟที่ใช้ทดสอบจริง

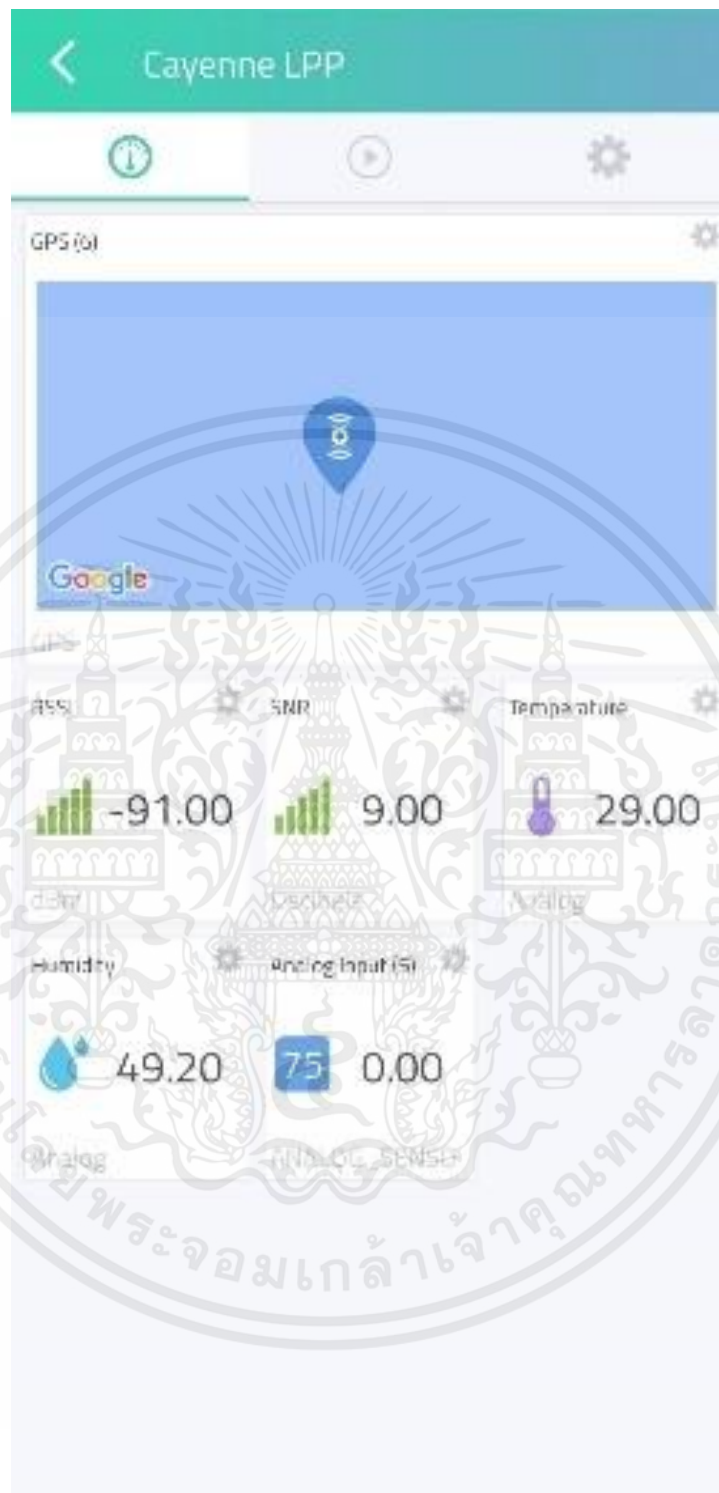
3.6 หน้าเว็บไซต์ Cayenne



รูปที่ 3.9 หน้าแสดงข้อมูลภายในเว็บไซต์ Cayenne

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 หน้าแอปพลิเคชัน Cayenne



รูปที่ 3.10 หน้าแสดงข้อมูลภายในแอปพลิเคชัน Cayenne

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Logger Data

Timestamp	Device Address	DevEUI	Payload	FPort	FCnt	FCnt	RSSI	SNR	SubBand	Channel	LRR ID
05/05/2023 05:00:36	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	38	25	-111.000000	-7.750000	G1	LC1	10000C3C
05/05/2023 05:00:19	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	37	24	-111.000000	-8.000000	G1	LC2	10000C3C
05/05/2023 04:59:20	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	35	22	-110.000000	-5.500000	G1	LC2	10000C3C
05/05/2023 04:58:18	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	33	21	-109.000000	-5.500000	G1	LC1	10000C3C
05/05/2023 04:57:15	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	31	20	-109.000000	-4.250000	G1	LC1	10000C3C
05/05/2023 04:56:18	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	29	18	-109.000000	-6.750000	G1	LC1	10000C3C
05/05/2023 04:54:45	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	26	17	-112.000000	-7.500000	G1	LC2	10000C3C
05/05/2023 04:54:29	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	25	16	-110.000000	-4.250000	G1	LC2	10000C3C

รูปที่ 3.11 แสดงข้อมูลที่ Device ส่งข้อมูลมายัง Platform



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

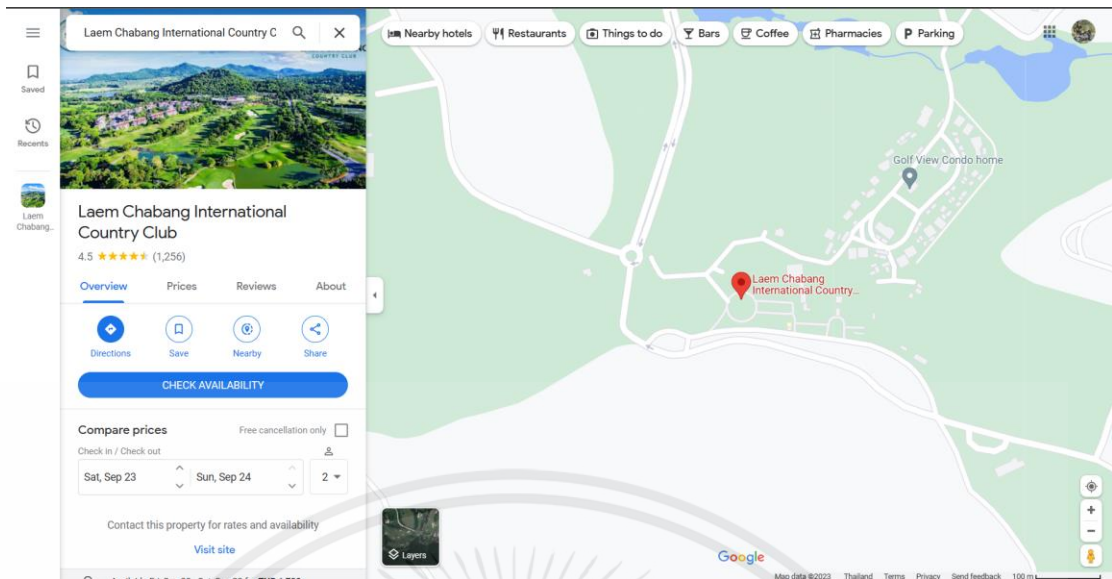
4.1 สถานที่การทดลอง

โครงการนี้ได้ทำการทดลอง ณ สนามกอล์ฟแหลมฉบังอินเตอร์เนลชั่นแนล อ.ศรีราชา จ.ชลบุรี เมื่อวันที่ 17 มิถุนายน พ.ศ. 2566 ในช่วงเวลาทดสอบตั้งแต่ 8.00 น – 14.00 น.



รูปที่ 4.1 สนามกอล์ฟแหลมฉบังอินเตอร์เนลชั่นแนล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 ตำแหน่งสนามทดสอบจาก Google map

4.2 เตรียมการทดลอง

4.2.1 อุปกรณ์ที่ใช้

1. รถกอล์ฟไฟฟ้า

ใช้ในการติดตั้งอุปกรณ์และทดสอบโดยการขับไปยังพื้นที่ต่างๆในสนาม



รูปที่ 4.3 รถกอล์ฟไฟฟ้าที่ใช้พลังงานจากแบตเตอรี่แรงดัน 48 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สมาร์ทโฟน



รูปที่ 4.4 สมาร์ทโฟนใช้สำหรับติดต่อสื่อสาร

3. แล็ปท็อป



รูปที่ 4.5 แล็ปท็อปใช้สำหรับรายงานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. อุปกรณ์ LoRa



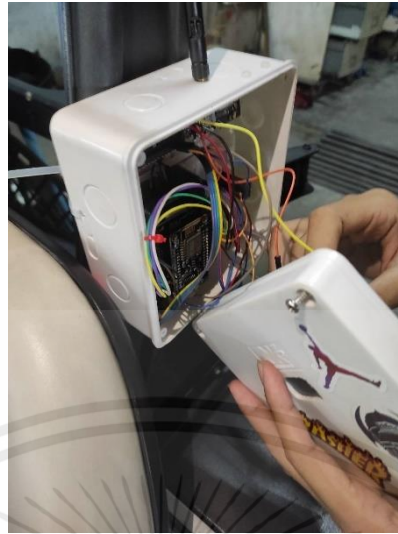
รูปที่ 4.6 ตำแหน่งกล่องเซนเซอร์ที่ติดตั้ง

4.2.2 พารามิเตอร์ที่เก็บผล

1. วันที่
2. เวลา
3. โซนพื้นที่
4. หลุมที่
5. พิกัดตำแหน่งละติจูดและลองจิจูด
6. ค่า SNR RSSI อุณหภูมิ และแรงดันไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลอง



รูปที่ 4.7 ติดตั้งอุปกรณ์เข้ากับรถกอล์ฟ

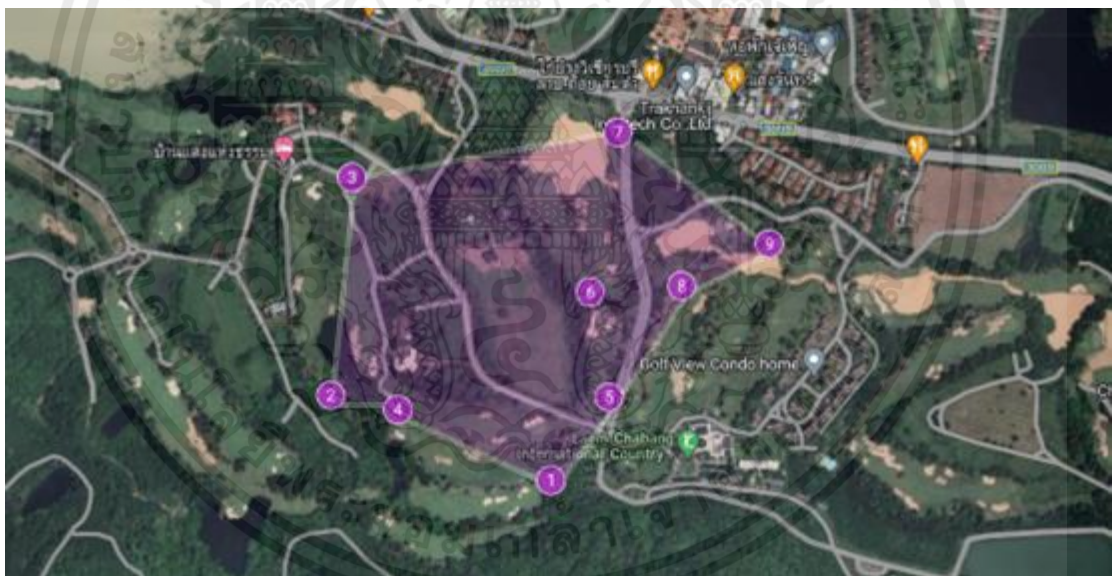


รูปที่ 4.8 รถกอล์ฟที่พร้อมทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)



(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ค)

รูปที่ 4.9 ตำแหน่งหลุมกอล์ฟในสนาม

- (ก) ตำแหน่งหลุมกอล์ฟในสนาม โซนพื้นที่ A
- (ข) ตำแหน่งหลุมกอล์ฟในสนาม โซนพื้นที่ B
- (ค) ตำแหน่งหลุมกอล์ฟในสนาม โซนพื้นที่ C



รูปที่ 4.10 การระบุตำแหน่งของรถกอล์ฟในสนาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

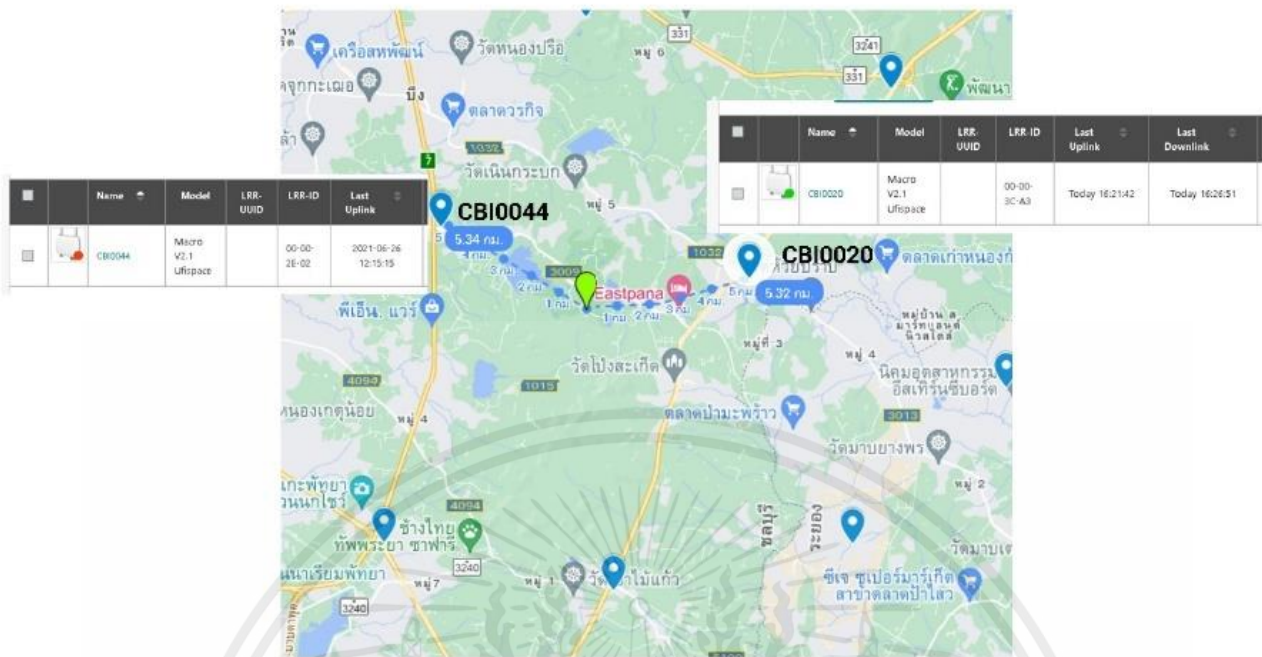
วันที่	เวลา	โซนพื้นที่	หลุมที่	พิกัดตำแหน่ง		ค่าจากอุปกรณ์					
				ละติจูด	ลองจิจูด	SNR	RSSI	อุณหภูมิ	ละติจูด	ลองจิจูด	แรงดันไฟฟ้า
17/6/2023	10:08:05	A	1	13.027923	101.037907	2.5	117	40	13.027400	101.037000	56.74
17/6/2023	10:10:46	A	2	13.027994	101.032897	2.25	117	38	13.028	101.0328	56.4300003
17/6/2023	10:12:00	A	3	13.028599	101.030937	Null	Null	Null	Null	Null	Null
17/6/2023	10:16:09	A	4	13.032542	101.028637	5.25	118	36.79	13.0329	101.0282	56.15
17/6/2023	10:19:28	A	5	13.035868	101.024012	7	119	38	13.0357	101.0243	55.31
17/6/2023	10:20:53	A	6	13.035313	101.028052	7.5	118	38	13.0353	101.0282	55.58
17/6/2023	10:23:00	A	7	13.034369	101.031314	Null	Null	Null	Null	Null	Null
17/6/2023	10:28:00	A	8	13.031977	101.029258	Null	Null	Null	Null	Null	Null
17/6/2023	10:34:01	A	9	13.028198	101.033451	8.75	119	39	13.0281	101.0335	55.65
17/6/2023	10:38:26	B	1	13.028367	101.037292	5	117	39	13.0284	101.0371	55.61
17/6/2023	10:41:00	B	2	13.029957	101.033066	Null	Null	Null	Null	Null	Null
17/6/2023	10:46:00	B	3	13.034018	101.033477	Null	Null	Null	Null	Null	Null
17/6/2023	10:50:36	B	4	13.029692	101.034349	5	109	39	13.0289	101.0371	54.2
17/6/2023	10:51:00	B	5	13.029927	101.038412	Null	Null	Null	Null	Null	Null
17/6/2023	10:53:00	B	6	13.031892	101.038050	Null	Null	Null	Null	Null	Null
17/6/2023	10:57:00	B	7	13.034855	101.038584	Null	Null	Null	Null	Null	Null
17/6/2023	10:59:00	B	8	13.031997	101.039797	Null	Null	Null	Null	Null	Null
17/6/2023	11:01:00	B	9	13.032773	101.041468	Null	Null	Null	Null	Null	Null
17/6/2023	11:04:00	C	1	13.029529	101.040666	Null	Null	Null	Null	Null	Null
17/6/2023	11:07:00	C	2	13.032502	101.043516	Null	Null	Null	Null	Null	Null
17/6/2023	11:10:00	C	3	13.031808	101.048428	Null	Null	Null	Null	Null	Null
17/6/2023	11:13:00	C	4	13.028175	101.050222	Null	Null	Null	Null	Null	Null
17/6/2023	11:15:00	C	5	13.030024	101.047233	Null	Null	Null	Null	Null	Null
17/6/2023	11:18:00	C	6	13.027780	101.047130	Null	Null	Null	Null	Null	Null
17/6/2023	11:21:00	C	7	13.029832	101.043171	Null	Null	Null	Null	Null	Null
17/6/2023	11:24:00	C	8	13.031085	101.044584	Null	Null	Null	Null	Null	Null
17/6/2023	11:26:00	C	9	13.032845	101.041999	Null	Null	Null	Null	Null	Null

รูปที่ 4.11 ตารางผลการทดลอง

หมายเหตุ Null คือ ค่าที่อุปกรณ์ไม่สามารถส่งข้อมูลไปยังเกตเวย์

จากตารางการทดลองในบางตำแหน่งของสนามกอล์ฟพบว่า เกตเวย์ไม่สามารถรับค่าจากอุปกรณ์ได้เนื่องจากเสาสัญญาณมีระยะห่างจากสนามอยู่ที่ 5.32 กิโลเมตร จึงเกิดปัญหาในบางพื้นที่ไม่สามารถส่งข้อมูลไปยังเกตเวย์ได้ ทำให้ข้อมูลในตารางบางส่วนขาดหายไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แผนที่ความครอบคลุมสัญญาณ LoRa

จากรูปที่ 4.12 จะเห็นได้ว่า อุปกรณ์ CBI0044 ซึ่งเป็นเกตเวย์ที่มีรัศมีครอบคลุมสนามกอล์ฟได้ปิดการใช้งานเนื่องจากไม่มีผู้ใช้ในพื้นที่รอบๆอุปกรณ์ จึงทำให้การส่งสัญญาณบางพื้นที่ไปยังอุปกรณ์ CBI0020 มีระยะทางที่ไกลกว่าอุปกรณ์ CBI0044 มีผลให้ไม่สามารถส่งข้อมูลในบางพื้นที่ของสนามได้

บทที่ 5

บทสรุป

5.1 สรุปผลการทดลอง

โครงการนี้นำเสนอแอปพลิเคชันติดตามข้อมูลรถกอล์ฟพลังงานไฟฟ้าด้วยเทคโนโลยีไอโอทีที่ใช้วิธีการส่งข้อมูลไปยังเว็บแอปพลิเคชันด้วยเทคโนโลยี LoRa สามารถช่วยให้ติดตามข้อมูลระดับพลังงานแบตเตอรี่รถกอล์ฟได้ รวมถึงสามารถระบุพิกัด GSP เพื่อติดตามตำแหน่งของรถกอล์ฟในสนามได้ในระยะทางไกล จากการทดลองพบว่า

1. ชุดเซ็นเซอร์สามารถส่งข้อมูลที่อ่านค่าอุณหภูมิแบตเตอรี่ ระดับพลังงานแบตเตอรี่ และพิกัดตำแหน่งไปยังเกตเวย์ได้
2. สามารถแสดงผลข้อมูลบนเว็บไซต์และแอปพลิเคชัน
3. การใช้งานภายในสนามกอล์ฟสามารถสื่อสารข้อมูลได้ในระยะทางไกลมากกว่า 10 กิโลเมตร

5.2 ปัญหาและอุปสรรค

1. สถานีส่งสัญญาณหรือเกตเวย์มีจำนวนจำกัดทำให้ไม่ครอบคลุมในพื้นที่สนามกอล์ฟทั้งหมด
2. ไม่สามารถใช้งานจีพีเอสได้ในอาคารได้

5.3 วิธีการแก้ไขปัญหา

1. จากปัญหาที่เสาสัญญาณมีระยะไกลทำให้สัญญาณไม่ครอบคลุมในพื้นที่วิธีแก้ปัญหาโดยให้ทางสนามกอล์ฟติดต่อกับทาง NT เพื่อทำการเพิ่มเกตเวย์ในบริเวณใกล้เคียง
2. จากปัญหาที่ไม่สามารถใช้งานจีพีเอสได้ในอาคารได้จะต้องแก้ไขด้วยวิธีการเปลี่ยนเสาตัวรับสัญญาณจีพีเอส ซึ่งจะช่วยให้ระบบสามารถรับสัญญาณจีพีเอสได้ทั้งในและนอกอาคารได้

เอกสารอ้างอิง

- [1] กนิษฐ์ สายวิจิตร, “วงจรกำเนิดสัญญาณไซน์แบบเลื่อนเฟสด้วย อาร์ซี ที่สามารถควบคุมขนาดโดยการกำหนดเงื่อนไขเริ่มต้น”, วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต, สาขาวิชาวิศวกรรมไฟฟ้า, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2537.
- [2] ชลวิทย์ วินิจฉัย และ ชัยวัฒน์ เทพศิลป์, “การควบคุมขดลวดเหนี่ยวนำอิมิตัวความถี่สูง”, วิทยานิพนธ์วิศวกรรมศาสตรบัณฑิต, สาขาวิชาวิศวกรรมไฟฟ้า, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2541
- [3] สอนการใช้งาน Lora เบื้องต้น, URL: <https://cs.bru.ac.th/lora/> เข้าถึงครั้งสุดท้าย วันที่ 25 มีนาคม 2566
- [4] สอนใช้งาน NodeMCU ESP8266 วัดอุณหภูมิและความชื้นด้วย Sensor DHT11, URL: <https://www.cybertice.com/article/161> เข้าถึงครั้งสุดท้าย วันที่ 25 มีนาคม 2566
- [5] ภาสกร พาเจริญ, “IoT Workshop ด้วย Arduino”, กรุงเทพมหานคร, สำนักพิมพ์ โปรวีชั่น จำกัด, 2563.
- [6] สอนใช้งาน GY-NEO6MV2 Ublox NEO-6M GPS Module กับ arduino, URL: <https://www.cybertice.com/article/45> เข้าถึงครั้งสุดท้าย วันที่ 25 มีนาคม 2566
- [7] PZEM-0 17 DC Energy Meter Online Monitoring with Blynk App”, URL: <https://solarduino.com/pzem-017-dc-energy-meter-online-monitoring-with-blynk-app/> เข้าถึงครั้งสุดท้าย วันที่ 25 มีนาคม 2566



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DHT11

1. การแนะนำ

DHT11 Temperature & Humidity Sensor มีเซ็นเซอร์วัดอุณหภูมิและความชื้นจับคู่กันด้วยเอาต์พุตสัญญาณดิจิทัลที่ปรับเทียบแล้ว โดยใช้การรับสัญญาณดิจิทัลพิเศษเทคนิคและเทคโนโลยีการตรวจจับอุณหภูมิและความชื้นทำให้มั่นใจได้ถึงความน่าเชื่อถือสูงและเสถียรภาพระยะยาวที่ดีเยี่ยม เซ็นเซอร์นี้มีการวัดความชื้นแบบต้านทานและส่วนประกอบวัดอุณหภูมิ NTC และเชื่อมต่อกับไมโครคอนโทรลเลอร์ประสิทธิภาพสูง 8 บิต ให้คุณภาพดีเยี่ยม ตอบสนองเร็ว ป้องกันการรบกวนความสามารถและความคุ้มทุน



รูปที่ ก.1 DHT11

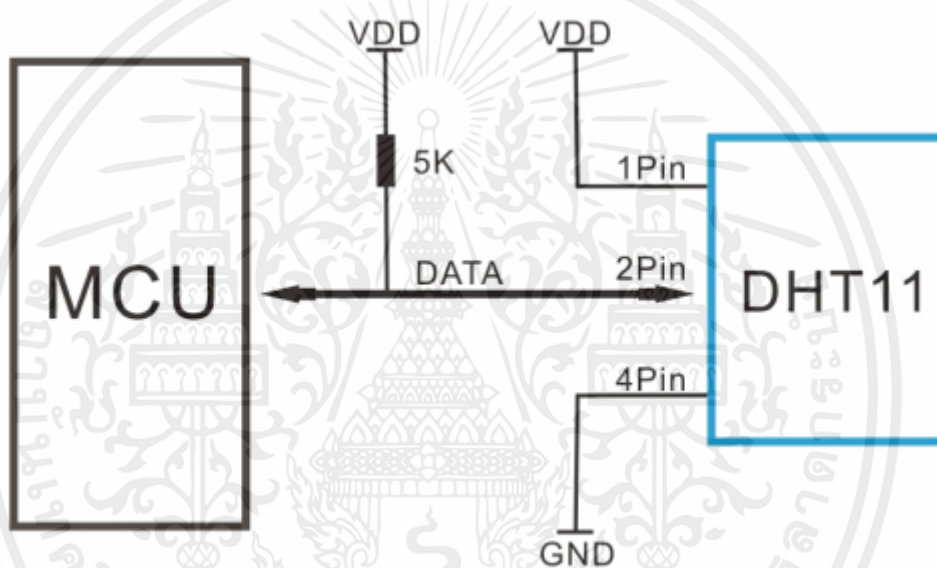
องค์ประกอบ DHT11 แต่ละชิ้นได้รับการสอบเทียบอย่างเข้มงวดในห้องปฏิบัติการซึ่งมีความแม่นยำสูงมากการสอบเทียบความชื้น ค่าสัมประสิทธิ์การสอบเทียบจะถูกจัดเก็บเป็นโปรแกรมในหน่วยความจำ OTPซึ่งใช้โดยกระบวนการตรวจจับสัญญาณภายในของเซ็นเซอร์ อินเทอร์เฟซแบบอนุกรมสายเดี่ยวทำให้การรวมระบบทำได้ง่ายและรวดเร็ว ขนาดที่เล็ก ใช้พลังงานต่ำ และสูงถึง 20 การส่งสัญญาณแบบเมตรทำให้เป็นตัวเลือกที่ดีที่สุดสำหรับการใช้งานที่หลากหลายรวมถึงเหล่านั้นสิ่งที่ต้องการมากที่สุด ส่วนประกอบคือแพ็คเกจจิ้งแพ็คเกจจิ้ง 4 พิน มันสะดวกที่จะสามารถเชื่อมต่อและแพ็คเกจจิ้งพิเศษตามคำขอของผู้ใช้

2. ข้อกำหนดทางเทคนิค

2.1 ภาพรวม

รายการ	ช่วงการวัด	ความแม่นยำ ของความชื้น	ความแม่นยำ ของอุณหภูมิ	ความ ละเอียด	รูปแบบ
DHT11	20-90% RH 0-50 °C	±5%RH	±2°C	1	3 ขา 1 แฉก

3. การต่อใช้งานแบบทั่วไป



รูปที่ ก.2 การต่อใช้งานทั่วไป

4. พลังงานและพิน

แหล่งจ่ายไฟของ DHT11 คือ 3-5.5V DC เมื่อจ่ายพลังงานให้กับเซ็นเซอร์ ห้ามส่งใดๆคำสั่งไปยังเซ็นเซอร์ภายในหนึ่งวินาทีเพื่อผ่านสถานะที่ไม่เสถียร หนึ่งสามารถเพิ่มตัวเก็บประจุมูลค่า 100nF ระหว่าง VDD และ GND สำหรับการกรองพลังงาน

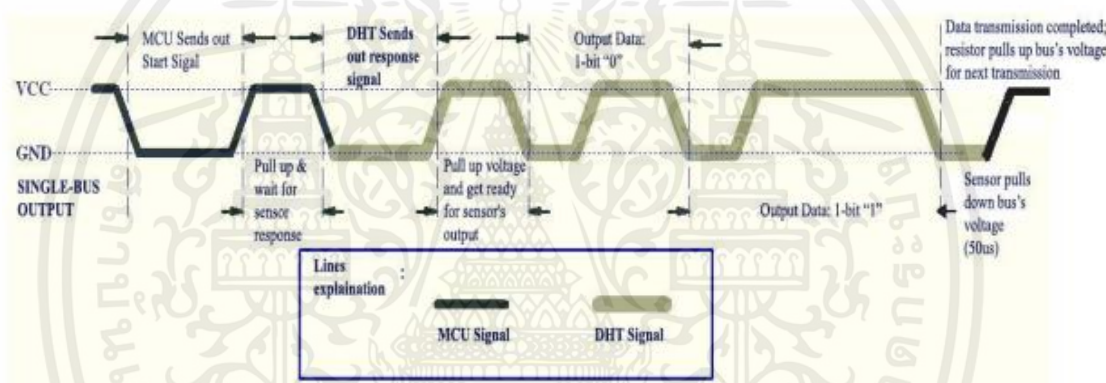
5. กระบวนการสื่อสาร: อินเทอร์เฟซแบบอนุกรม (สายเดี่ยวสองทาง)

รูปแบบข้อมูลบัสเดี่ยวใช้สำหรับการสื่อสารและการซิงโครไนซ์ระหว่าง MCU และเซ็นเซอร์ DHT11 หนึ่งกระบวนการสื่อสารประมาณ 4 มิลลิวินาทีที่ข้อมูลประกอบด้วยส่วนทศนิยมและส่วนรวม การส่งข้อมูลที่สมบูรณ์คือ 40 บิตและเซ็นเซอร์จะส่งข้อมูลบิตที่สูงกว่าก่อนรูปแบบข้อมูล: ข้อมูล RH เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินทิกรัล 8 บิต + ข้อมูล RH ทศนิยม 8 บิต + ข้อมูลอินทิกรัล 8 บิต + ทศนิยม 8 บิตข้อมูล + ผลรวมการตรวจสอบ 8 บิต หากการรับส่งข้อมูลถูกต้อง ผลรวมตรวจสอบควรเป็น 8 บิตสุดท้ายของ "ข้อมูล RH อินทิกรัล 8 บิต + ข้อมูล RH ทศนิยม 8 บิต + ข้อมูลอินทิกรัล 8 บิต + ข้อมูลทศนิยม 8 บิต"

5.1 กระบวนการสื่อสารโดยรวม

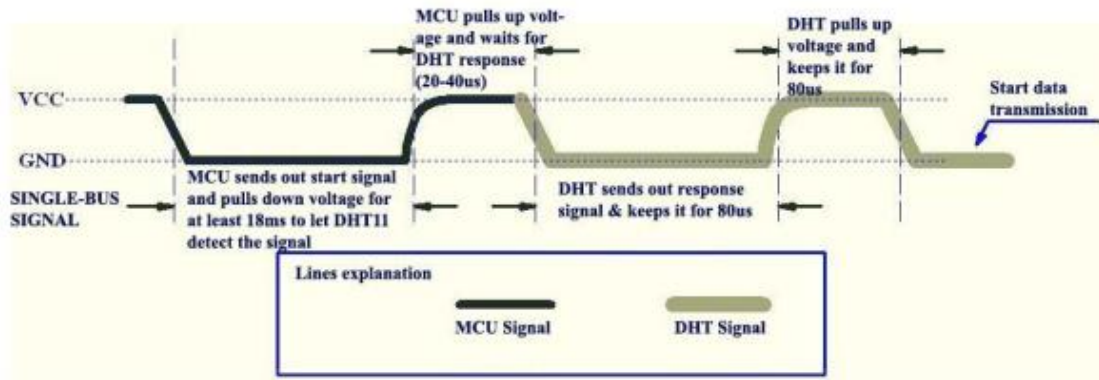
เมื่อ MCU ส่งสัญญาณเริ่มต้น DHT11 จะเปลี่ยนจากโหมดการไม่ใช้พลังงานต่ำเป็น โหมดโหมดการทำงาน รอให้ MCU เสร็จสิ้นสัญญาณเริ่มต้น เมื่อเสร็จสิ้น DHT11 จะส่งสัญญาณการตอบสนองของข้อมูล 40 บิตที่มีข้อมูลความชื้นสัมพัทธ์และอุณหภูมิด้วยมจร. ผู้ใช้สามารถเลือกที่จะรวบรวม (อ่าน) ข้อมูลบางส่วน โดยไม่มีสัญญาณเริ่มต้นจาก MCU, DHT11 จะไม่ให้สัญญาณตอบกลับไปยัง MCU เมื่อรวบรวมข้อมูลแล้ว DHT11 จะเปลี่ยนเข้าสู่โหมดการไม่ใช้พลังงานต่ำจนกว่าจะได้รับสัญญาณเริ่มต้นจาก MCU อีกครั้ง



รูปที่ ก.3 กระบวนการสื่อสารโดยรวม

5.2 MCU ส่งสัญญาณเริ่มต้นไปยัง DHT

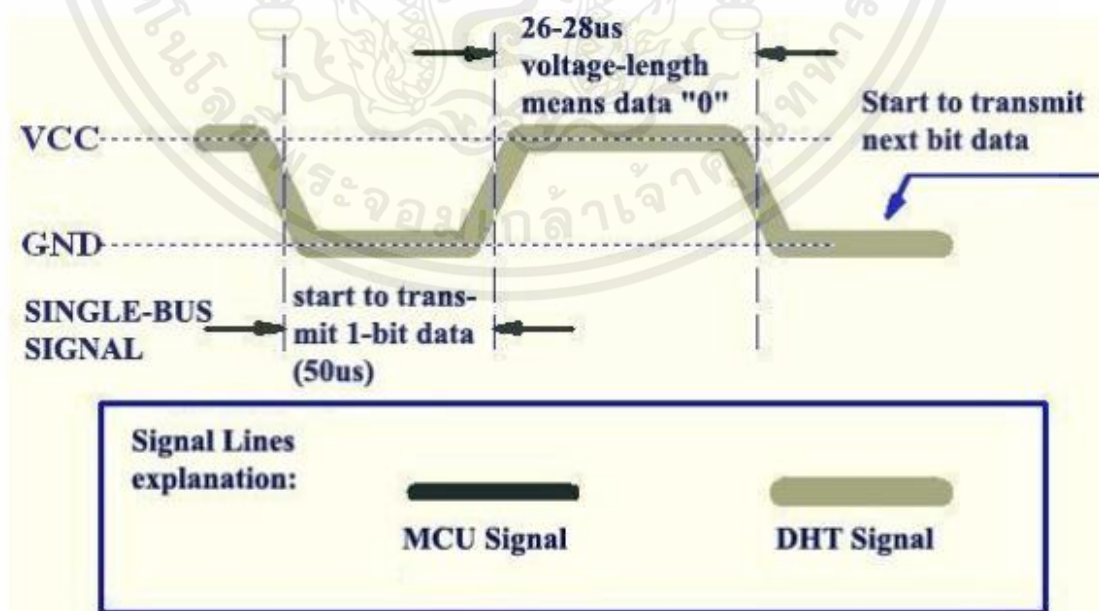
สถานะว่างบัสเดียวของข้อมูลอยู่ที่ระดับไฟฟ้าแรงสูง เมื่อการสื่อสารระหว่าง มจร. และ DHT11 เริ่มต้นขึ้น โปรแกรมของ MCU จะตั้งค่าระดับแรงดัน Data Single-bus จากสูงไปต่ำและกระบวนการนี้ต้องใช้เวลาอย่างน้อย 18 มิลลิวินาทีเพื่อให้แน่ใจว่า DHT จะตรวจจับสัญญาณของ MCU จากนั้น MCU จะดึงแรงดันไฟฟ้าขึ้นและรอ 20-40us สำหรับการตอบสนองของ DHT



รูปที่ ก.4 MCU ส่งสัญญาณเริ่มต้นไปยัง DHT

5.3 การตอบสนอง DHT ไปยัง MCU

เมื่อ DHT ตรวจพบสัญญาณเริ่มต้น มันจะส่งสัญญาณตอบสนองระดับแรงดันต่ำออกมา ซึ่งมีอายุการใช้งาน 80us. จากนั้นโปรแกรมของ DHT จะตั้งค่าระดับแรงดัน Data Single-bus จากต่ำไปสูงและเก็บไว้เป็นเวลา 80us สำหรับการเตรียม DHT สำหรับการส่งข้อมูลเมื่อ DATA Single-Bus อยู่ที่ระดับแรงดันไฟฟ้าต่ำ หมายความว่า DHT กำลังส่งการตอบสนองสัญญาณ. เมื่อ DHT ส่งสัญญาณตอบกลับ มันจะดึงแรงดันไฟฟ้าขึ้นและคงไว้ที่ 80us และเตรียมการรับส่งข้อมูลเมื่อ DHT ส่งข้อมูลไปยัง MCU ข้อมูลทุกบิตจะเริ่มต้นด้วยระดับแรงดันไฟฟ้าต่ำ 50us และความยาวของสัญญาณระดับแรงดันสูงต่อไปนี้กำหนดว่าบิตข้อมูลเป็น "0" หรือ "1"



รูปที่ ก.5 การตอบสนอง DHT ไปยัง MCU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากสัญญาณตอบรับจาก DHT อยู่ที่ระดับแรงดันสูงเสมอ แสดงว่า DHT ไม่ใช่ตอบสนองอย่างถูกต้องและโปรดตรวจสอบการเชื่อมต่อ เมื่อมีการส่งข้อมูลบิตสุดท้าย DHT11 ถึงระดับแรงดันไฟฟ้าลงและคงไว้ที่ 50us จากนั้นแรงดันบัสเดี่ยวจะเป็นตัวต้านทานดึงขึ้นเพื่อตั้งค่ากลับเป็นสถานะว่าง

6. ลักษณะทางไฟฟ้า

	Conditions	Minimum	Typical	Maximum
Power Supply	DC	3V	5V	5.5V
Current Supply	Measuring	0.5mA		2.5mv
	Average	0.2mA		1mA
	Standby	100uA		150uA
Sampling period	Second	1		

7. ความสนใจของการสมัคร

การใช้เซ็นเซอร์ DHT11 เกินกว่าช่วงการทำงานที่ระบุไว้ในเอกสารข้อมูลนี้อาจส่งผลให้ 3%RH สัญญาณเปลี่ยน/ความคลาดเคลื่อน เซ็นเซอร์ DHT11 สามารถกลับสู่สถานะสอบเทียบได้ที่ละน้อยเมื่อมันกลับสู่สภาพการทำงานปกติและทำงานภายในขอบเขต

ESP8266

1. การแนะนำ

DFRobot FireBeetle เป็นชุดของการใช้พลังงานต่ำการพัฒนาฮาร์ดแวร์ที่ออกแบบมาสำหรับ Internet of Things (IoT) Firebeetle ESP8266 เป็นบอร์ดพัฒนาที่รวมเข้ากับ IoT WiFi TCP/IP, MCU 32 บิต, ADC 10 บิต และหลายอินเทอร์เฟซ เช่น HSPI, UART, PWM, I2C และ I2S ใน DTIM10 การใช้พลังงานเต็มถึง รักษาการเชื่อมต่อ WiFi ถึง 1.2mW มาพร้อมกับ 16MB หน่วยความจำแฟลชภายนอก SPI, ESP8266 พร้อมใช้งานสำหรับโปรแกรมและการจัดเก็บเฟิร์มแวร์ เข้ากันได้กับการเขียนโปรแกรม Arduino ทำให้ Firebeetle ESP8266 ลดอุปสรรคในการเขียนโปรแกรม ผู้ประกอบการสามารถนำรหัสการเขียนโปรแกรม Arduino เข้าสู่ ESP8266 โดยตรงเพื่อลดความยากในการทำงานและเพิ่มความมั่นคงของบอร์ด



รูปที่ ก.6 บอร์ด ESP8266

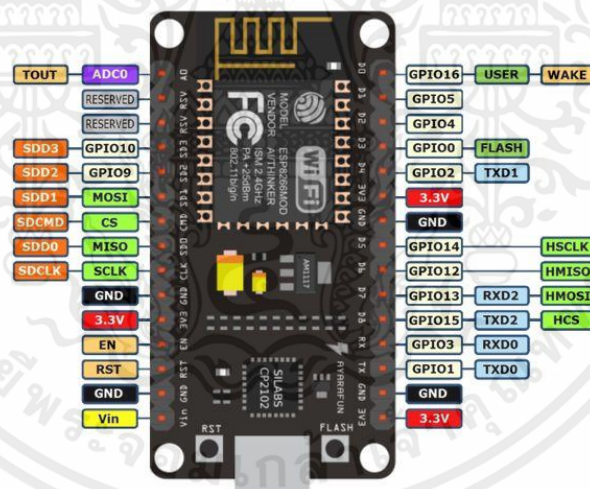
2. ข้อมูลจำเพาะ

- Operating Voltage: 3.3V
- Input Voltage (limits): 3.3~5V (Lithium Battery:3.7V & USB:5V)
- Microcontroller: Tensilica L106 (32-bit MCU)
- Clock Speed: 80MHz (Maximum: 160MHz)
- SRAM : 50KB
- External Flash Memory: 16MB
- (/wiki/index.php/File:Warning_yellow.png)
- DC Current in the Low-Power-Consumption: 46uA
- Average Operating Current: 80mA
- Maximum Discharging Current: 600mA (LDO-3.3 Output)
- Maximum Charging Current: 500mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Digital Pin x10
- Analog Pin x1
- SPI interface x1
- I2C interface x1
- IR interface x1
- I2S interface x1
- Interface: XH2.54mm Pin (No soldering default)
- In Combination of Wi-Fi MAC/ BB/RF/PA/LNA
- Wi-Fi: IEEE802.11 b/g/n (2.4 GHz~2.5 GHz), not support 5GHz Wi-Fi.
- Operating Temperature: $-10^{\circ}\text{C}\sim+55^{\circ}\text{C}$
- Dimension: 58 × 29(mm)
- Weight: 24g

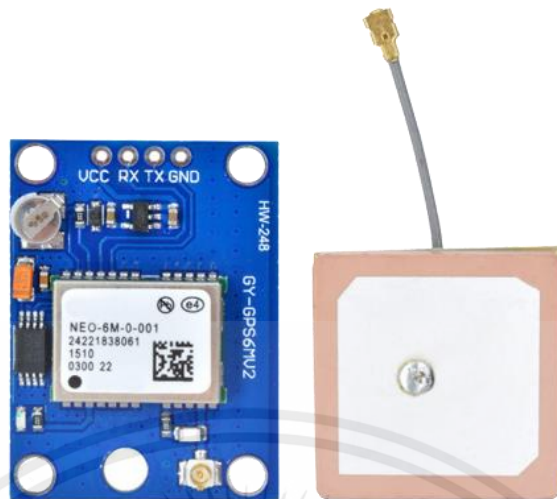
3. ภาพรวม



รูปที่ ก.7 ESP8266 Pinout

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Arduino GY-NEO6MV2 GPS Module c/w Antenna & Flight Control EEPROM



รูปที่ ก.8 GY-NEO6MV2

1. การแนะนำ

บอร์ด GY-NEO6MV2 มีโมดูล GPS u-blox NEO-6M พร้อมเสาอากาศและในตัว EEPROM เข้ากันได้กับบอร์ดควบคุมการบินต่างๆ ที่ออกแบบมาเพื่อทำงานร่วมกับโมดูลจีพีเอส

2. ข้อกำหนดทางเทคนิค

- ช่วงแหล่งจ่ายไฟ 3 V ถึง 5 V
- รุ่น: GY-GPS6MV2
- เสาอากาศเซรามิก
- EEPROM สำหรับบันทึกข้อมูลการกำหนดค่าเมื่อปิดเครื่อง
- แบตเตอรี่สำรอง
- ไฟสัญญาณ LED
- เส้นผ่านศูนย์กลางรูยึด 3 มม.
- อัตรารับส่งข้อมูลเริ่มต้น 9600 bps
- ขนาดโมดูล 23mm * 30mm
- เสาอากาศ ขนาด 12*12mm
- สาย 20 mm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

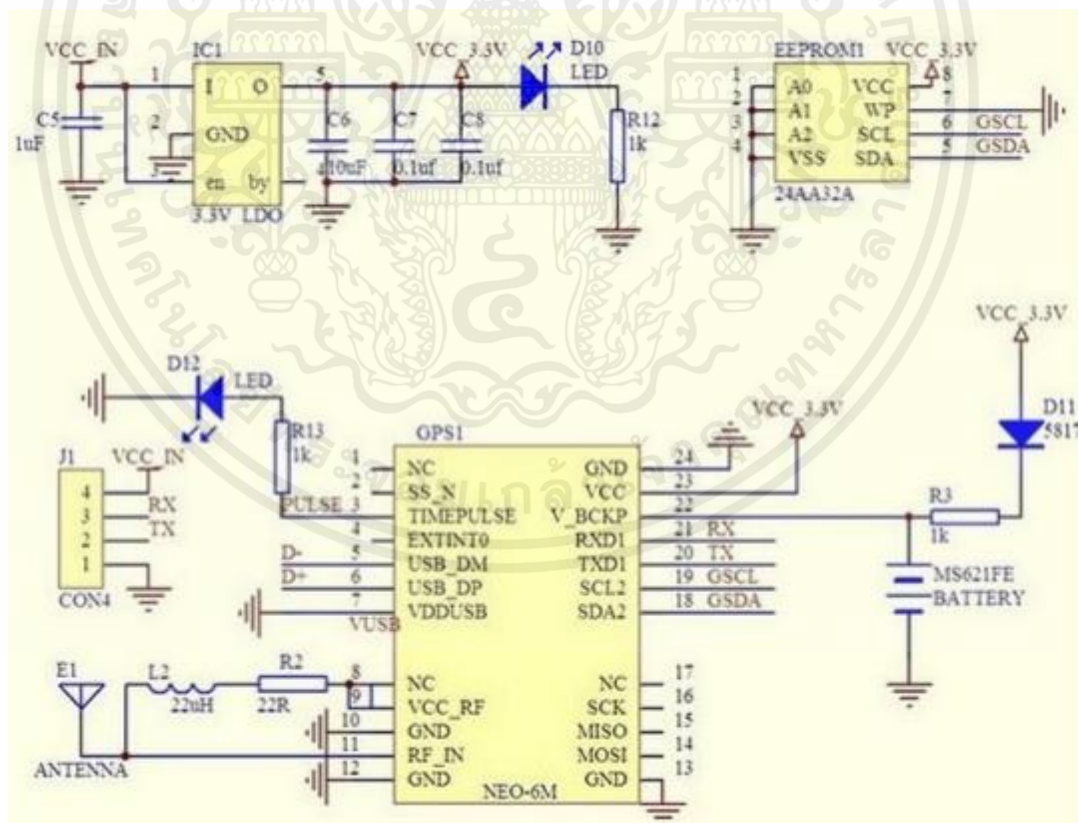
3. คุณสมบัติ

- ใช้โมดูล XM37-1612 แพลตฟอร์ม MTK พร้อมเสาอากาศรับสัญญาณสูง
- ระดับ TTL เข้ากันได้กับระบบ 3.3V/5V
- อัตราบอดเริ่มต้น 9600
- ด้วยแบตเตอรี่สำรองแบบชาร์จไฟได้ สามารถบันทึกข้อมูล ephemeris เมื่อปิดเครื่องและเริ่มต้นอย่างอบอุ่น
- เหมาะสำหรับ RC quad copter, navigator

4. Pin out

- VCC: Connect 3.3V/5V
- GND: Connect GND
- TXD: Connect MCU.RX
- RXD: Connect MCU.TX

5. Schematic



รูปที่ ก.9 Schematic GY-NEO6MV2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PZEM-003/017 DC communication module

ข้อมูลจำเพาะของโมดูลสื่อสาร DC PZEM-003/017 โมดูลส่วนใหญ่ใช้สำหรับการวัดแรงดัน DC, กระแส, กำลังไฟฟ้าที่ใช้งาน, ความถี่และพลังงานการบริโภค, โมดูลไม่มีฟังก์ชันการแสดงผล, ข้อมูลถูกอ่านผ่านอินเทอร์เฟซ RS485 PZEM-003: ช่วงการวัด 10A (ตัวปิดในตัว) PZEM-017: ช่วงการวัด 50A, 100A, 200A, 300A (ช่วงปัจจุบันขึ้นอยู่กับข้อกำหนดการแบ่งภายนอก)

1. คำอธิบายฟังก์ชัน

1.1 แรงดันไฟฟ้า

1. ช่วงการวัด: 0.05-300V. (เมื่อแรงดัน < 7V โพรตใช้โหมดการจ่ายไฟอิสระ)
2. ความละเอียด: 0.01V.
3. ความแม่นยำในการวัด: 1%

1.2 กระแส

1. ช่วงการวัด: 0.01-10A (PZEM-003); 0.02-300A (PZEM-017; สามารถจับคู่กับ 50, 100, 200, 300A shunt)
2. ความละเอียด: 0.01A
3. ความแม่นยำในการวัด: 1%

1.3 กำลังไฟ

1. ช่วงการวัด: 0.1-3kW (PZEM-003); 0.2-90kW (PZEM-017)
2. ความละเอียด: 0.1W
3. ความแม่นยำในการวัด: 1%

1.4 การใช้พลังงาน

1. ช่วงการวัด: 0-9999kWh
2. ความละเอียด: 1Wh
3. ความแม่นยำในการวัด: 1%
4. รีเซ็ตพลังงาน: ใช้ซอฟต์แวร์เพื่อรีเซ็ต

1.5 สัญญาณเตือนแรงดันไฟเกิน

สามารถตั้งค่าเกณฑ์แรงดันไฟฟ้า แบ่งเป็นแรงดันไฟฟ้าสูงและเกณฑ์แรงดันไฟฟ้าต่ำ เมื่อแรงดันไฟฟ้าที่วัดได้เกินเกณฑ์ก็สามารถแจ้งเตือนได้เกณฑ์แรงดันไฟฟ้าสูงเริ่มต้นคือ 300V เกณฑ์แรงดันไฟฟ้าต่ำเริ่มต้นคือ 7V

1.6 ส่วนติดต่อสื่อสารอินเทอร์เฟซ RS485

2. โพรโตคอลการสื่อสาร

2.1 โพรโตคอลชั้นกายภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พีสคัลเลเยอร์ใช้อินเตอร์เฟซการสื่อสาร UART ถึง RS485 อัตรารับส่งข้อมูลคือ 9600, 8 data bits, 2 stop bit, no parity

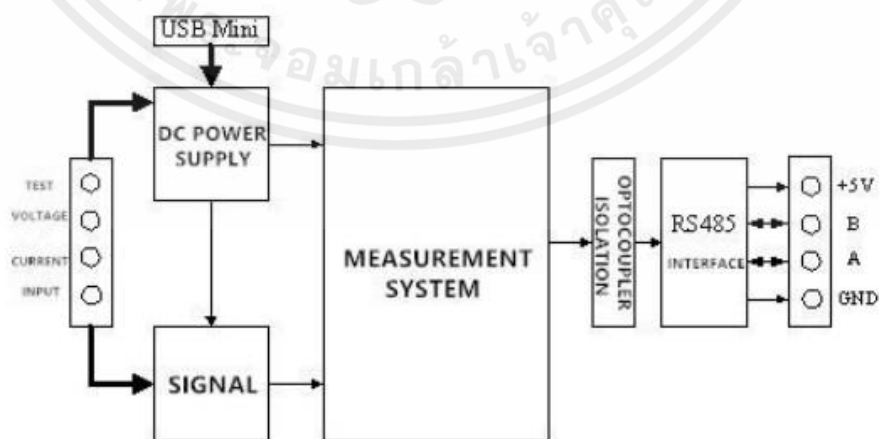
2.2 โพรโตคอลชั้นแอปพลิเคชัน

ชั้นแอปพลิเคชันใช้โปรโตคอล Modbus-RTU เพื่อสื่อสาร ปัจจุบันก็เท่านั้นรองรับรหัสฟังก์ชัน เช่น 0x03 (Read Holding Register), 0x04 (Read Input Register), 0x06 (เขียนทะเบียนเดียว), 0x41 (สอบเทียบ), 0x42 (รีเซ็ตพลังงาน) เป็นต้นรหัสฟังก์ชัน 0x41 ใช้สำหรับภายในเท่านั้น (ที่อยู่สามารถเป็น 0xF8 เท่านั้น) ใช้สำหรับโรงงานสอบเทียบและกลับไปโรงงานครั้งการบำรุงรักษา หลังจากรหัสฟังก์ชันเพื่อเพิ่ม 16 บิตรหัสผ่าน รหัสผ่านเริ่มต้นคือ 0x3721 ช่วงที่อยู่ของสเลฟคือ 0x01 ~ 0xF7 ใช้ที่อยู่ 0x00 เป็นการออกอากาศที่อยู่ทาสไม่จำเป็นต้องตอบเจ้านาย ที่อยู่ 0xF8 ใช้เป็นที่อยู่ทั่วไปที่อยู่ ที่อยู่นี้สามารถใช้ได้เฉพาะในสภาพแวดล้อมแบบ single-slave และสามารถใช้ในการสอบเทียบ ฯลฯ การดำเนินงาน

2.3 อ่านผลการวัด

รูปแบบคำสั่งของมาสเตอร์อ่านผลการวัดคือ (ทั้งหมด 8 ไบต์): ที่อยู่ทาส + 0x04 + ลงทะเบียนที่อยู่ไบต์สูง + ลงทะเบียนที่อยู่ไบต์ต่ำ + หมายเลขของการลงทะเบียนไบต์สูง + จำนวนการลงทะเบียนไบต์ต่ำ + ตรวจสอบ CRC ไบต์สูง + ตรวจสอบ CRC ไบต์ต่ำรูปแบบคำสั่งตอบกลับจากสเลฟแบ่งเป็น 2 แบบ คือคำตอบที่ถูกต้อง: ที่อยู่ทาส + 0x04 + จำนวนไบต์ + ลงทะเบียน 1 Data High Byte + ลงทะเบียน 1 ไบต์ข้อมูลต่ำ + ... + CRC ตรวจสอบไบต์สูง + CRC ตรวจสอบไบต์ต่ำตอบกลับข้อผิดพลาด: ที่อยู่ทาส + 0x84 + รหัสผิดปกติ + ตรวจสอบ CRC ไบต์สูง + ตรวจสอบ CRC ไบต์ต่ำวิเคราะห์โค้ดผิดปกติได้

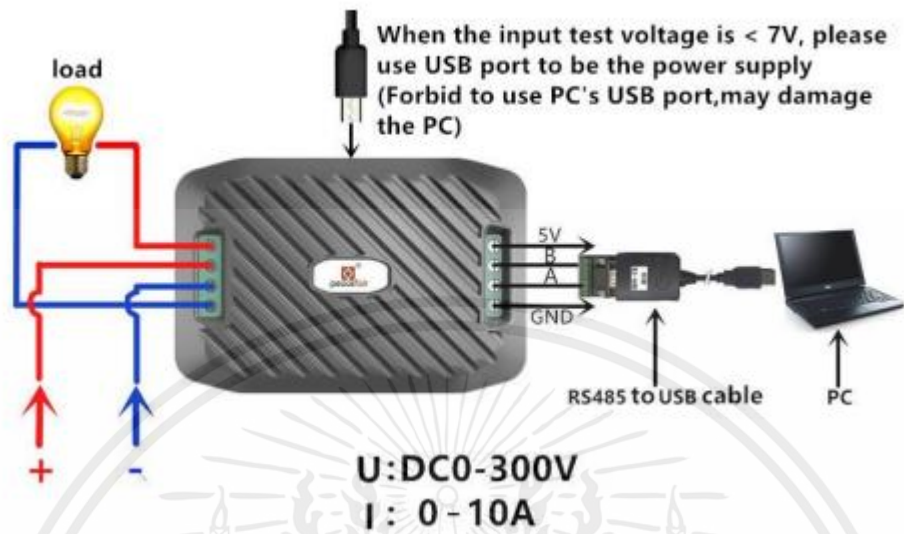
3. ไตอะแกรมบล็อกการทำงาน



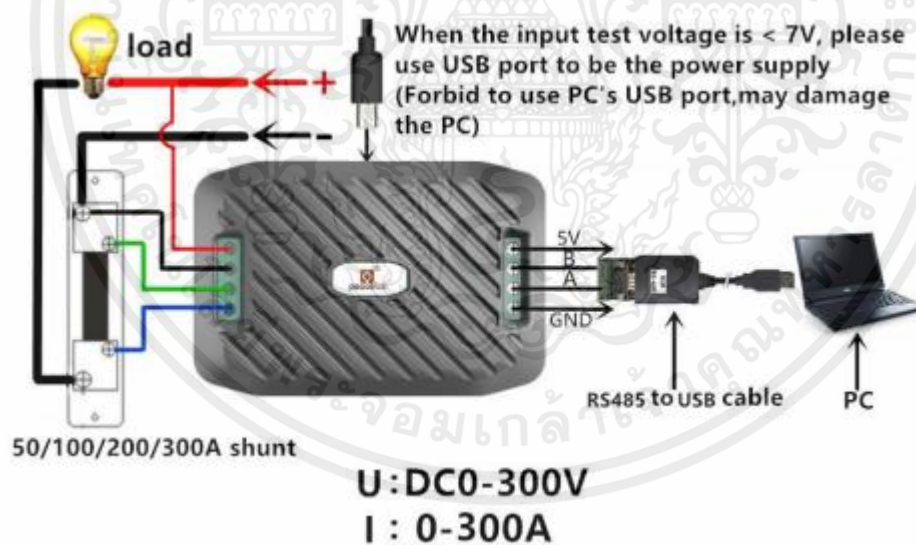
รูปที่ ก.10 ไตอะแกรมบล็อกการทำงาน PZEM-017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.แผนภาพการเดินสายไฟ



รูปที่ ก.11 แผนภาพการเดินสายไฟแบบไม่มี Shut



รูปที่ ก.12 แผนภาพการเดินสายไฟแบบมี Shut

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TTGO t-beam esp32 lora

1. คำอธิบาย

1.1 ภาพรวม

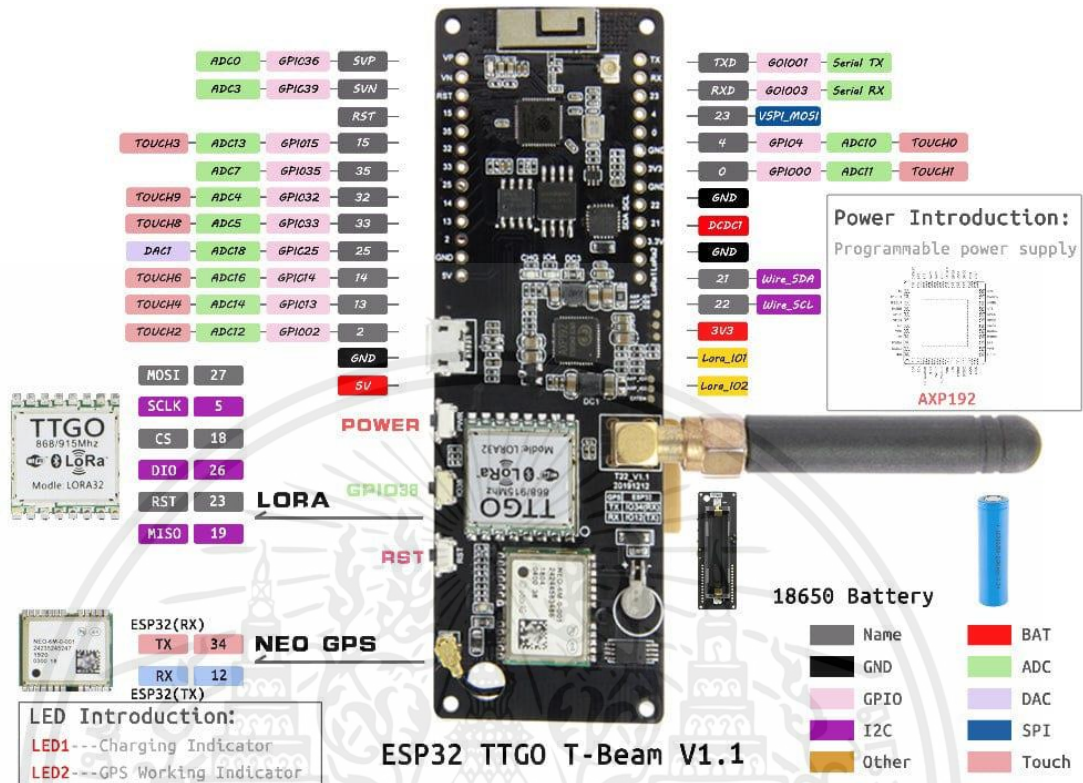
TTGO T-Beam เป็นบอร์ดพัฒนา ESP32 พร้อมแฟลช 4 MB ที่ใช้ชิป ESP32 โดยตรง มั่นรวม: SemTech SX1276 หรือ SX1278 สำหรับการสื่อสาร LoRaWANตัวรับสัญญาณ U-Blox Neo-6M GPS มีบอร์ดอย่างน้อยสามประเภท: rev0, rev1 และ V1.0 เวอร์ชัน rev0 และ rev1 มีความคล้ายคลึงกันมาก ข้อแตกต่างเพียงอย่างเดียวคือ rev1 มี LED เพิ่มเติมที่เชื่อมต่อกับ GPIO14 pinout ของ V1.0 มีการเปลี่ยนแปลงมากขึ้น ดังนั้นจึงจำเป็นต้องเพิ่มบรรทัดต่อไปนี้ใน makefile ของแอปพลิเคชันเพื่อใช้ในการกำหนดค่าตามสำหรับ TTGO T-Beam V1.0:

1.2 คุณลักษณะของผลิตภัณฑ์

TTGO T-Beam มีส่วนประกอบบอร์ดดังต่อไปนี้: SemTech SX1276 หรือ SX1278 สำหรับการสื่อสาร LoRaWANตัวรับสัญญาณ U-Blox Neo-6M GPSบอร์ดมีฮาร์ดแวร์สองเวอร์ชัน: SemTech SX1278 สำหรับการสื่อสาร LoRaWAN ในย่านความถี่ 433 MHz SemTech SX1276 สำหรับการสื่อสาร LoRaWAN ในย่านความถี่ 868/915 MHz เนื่องจาก GPIO จำนวนมากแยกย่อยออกไป จึงสามารถใช้เพื่อวัตถุประสงค์ที่แตกต่างกันในแอปพลิเคชันต่างๆ เพื่อความยืดหยุ่น GPIO บางตัวอาจแสดงอยู่ในการกำหนดค่าอุปกรณ์ต่อพ่วงต่างๆ ตัวอย่างเช่น GPIO0 ใช้ในข้อกำหนดของสัญญาณ ADC ADC_GPIOS และข้อกำหนดของสัญญาณ PWM PWM0_GPIOS สิ่งนี้เป็นไปได้เนื่องจาก GPIO ใช้สำหรับอินเทอร์เฟซอุปกรณ์ต่อพ่วงเฉพาะเมื่อมีการใช้โมดูลต่อพ่วงที่เกี่ยวข้อง เช่น โมดูล periph_i2c หรือเรียกฟังก์ชัน init ที่เกี่ยวข้อง เช่น adc_init, dac_init และ pwm_init หรืออินเทอร์เฟซอุปกรณ์ต่อพ่วงที่เกี่ยวข้องถูกใช้เป็นครั้งแรก เช่น spi_acquire นั่นคือวัตถุประสงค์สำหรับการใช้ GPIO นั้นขึ้นอยู่กับว่าโมดูลหรือฟังก์ชันใดถูกใช้งานก่อน

2. Pin Definition

2.1 Pin assignment



รูปที่ ก.13 TTGO t-beam esp32 lora pinout

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function	GPIOs	Remarks	Configuration
BUTTON0	GPIO39	low active	
LED0	GPIO14	high active	
ADC	GPIO0, GPIO2, GPIO4, GPIO13, GPIO25, GPIO32, GPIO33, GPIO34, GPIO35		ADC Channels
DAC	GPIO25		DAC Channels
PWM_DEV(0)	GPIO0, GPIO2, GPIO25		PWM Channels
I2C_DEV(0):SDA	GPIO21		I2C Interfaces
I2C_DEV(0):SCL	GPIO22	I2C_SPEED_FAST is used	I2C Interfaces
SPI_DEV(0):CLK	GPIO5	VSPI is used	SPI Interfaces
SPI_DEV(0):MISO	GPIO19	VSPI is used	SPI Interfaces
SPI_DEV(0):MOSI	GPIO27	VSPI is used	SPI Interfaces
SPI_DEV(0):CS0	GPIO18	VSPI is used	SPI Interfaces
UART_DEV(0):TxD	GPIO1	Console (configuration is fixed)	UART interfaces
UART_DEV(0):RxD	GPIO3	Console (configuration is fixed)	UART interfaces
UART_DEV(1):TxD	GPIO15	GPS (configuration is fixed)	UART interfaces
UART_DEV(1):RxD	GPIO12	GPS (configuration is fixed)	UART interfaces
TTGO- T-Beam V1.0			
Function	GPIOs	Remarks	Configuration
BUTTON0	GPIO38	low active	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

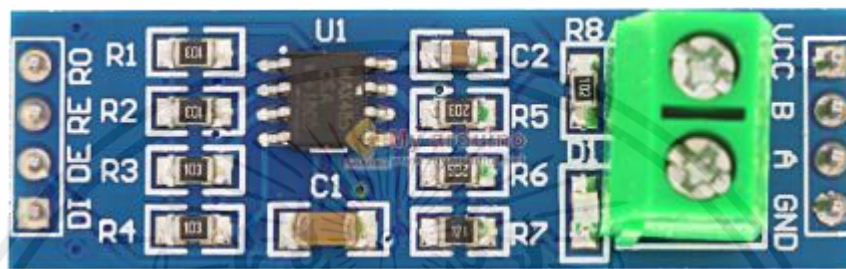
Function	GPIOs	Remarks	Configuration
ADC	GPIO0, GPIO2, GPIO4, GPIO13,		ADC Channels
	GPIO25, GPIO32, GPIO33, GPIO35		
DAC	GPIO25		DAC Channels
PWM_DEV(0)	GPIO0, GPIO2, GPIO25		PWM Channels
I2C_DEV(0):SDA	GPIO21		I2C Interfaces
I2C_DEV(0):SCL	GPIO22	I2C_SPEED_FAST is used	I2C Interfaces
SPI_DEV(0):CLK	GPIO5	VSPI is used	SPI Interfaces
SPI_DEV(0):MISO	GPIO19	VSPI is used	SPI Interfaces
SPI_DEV(0):MOSI	GPIO27	VSPI is used	SPI Interfaces
SPI_DEV(0):CS0	GPIO18	VSPI is used	SPI Interfaces
UART_DEV(0):TxD	GPIO1	Console (configuration is fixed)	UART interfaces
UART_DEV(0):RxD	GPIO3	Console (configuration is fixed)	UART interfaces
UART_DEV(1):TxD	GPIO34	GPS (configuration is fixed)	UART interfaces
UART_DEV(1):RxD	GPIO12	GPS (configuration is fixed)	UART interfaces

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAX485

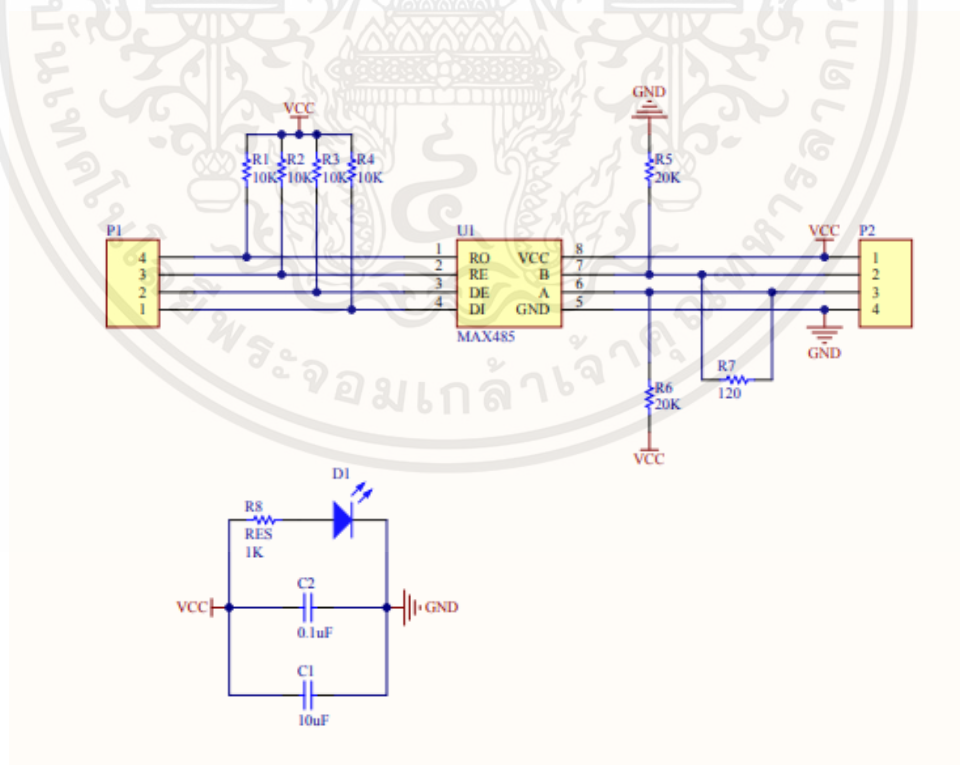
1. ข้อมูลทั่วไป

MAX485 RS485 module ใช้สำหรับให้ Arduino ส่งข้อมูลแบบระยะไกล แบบใช้สายสามารถลากสายยาวได้ถึง 1.2 กิโลเมตร ใช้พลังงานต่ำมาก สามารถเซ็ตให้เป็น ตัวรับหรือตัวส่งก็ได้ (Slave and Master) การเชื่อมต่อสายง่ายเพียงแค่นำสายสัญญาณเสียบเข้ากับ 2P Terminal ก็สามารถใช้งานได้ทันที รองรับไมโครคอนโทรลเลอร์ หลากหลายชนิด ทำงานที่แรงดัน DC 5 Volt



รูปที่ ก.14 MAX485

2. วงจรภายใน



รูปที่ ก.15 วงจรภายใน MAX485

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. รายละเอียด

3.1 ขนาดของบอร์ด 44*14 mm

3.2 ใช้ Chip MAX485 เพื่อเชื่อมต่อกับสาย RS 485 ใช้พลังงานต่ำ

3.3 ในบอร์ดมี 2P Terminal ทำให้ง่ายต่อการเชื่อมต่อวงจร

3.4 ใช้แรงดันไฟฟ้า DC 5Volt

3.5 รองรับ ไมโครคอนโทรลเลอร์ หลากหลายรุ่น

4. ข้อมูลจำเพาะ

Comment	Description	Designator	Footprint	LibRef	Quantity
10uF	Capacitor	C1	1206	CAP	1
0.1uF	Capacitor	C2	0805	CAP	1
LED	Typical INFRARED GaAs LED	D1	LED-0805	LED	1
Header 4	Header, 4-Pin	P1,P2	HDR1X4	Header 4	2
10K	Semiconductor Resistor	R1,R2,R3,R4	0805	RES	4
20K	Semiconductor Resistor	R5,R6	0805	RES	2
120	Semiconductor Resistor	R7	0805	RES	1
RES	Semiconductor Resistor	R8	0805	RES	1
MAX485		U1	NSO8_N	MAX485	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code esp8266

```

Project_Learn_21 (Arduino 1.8.9)
File Edit Search Tools Help
Project_Learn_21.h
1 #include <SoftwareSerial.h>
2 #include <Arduino.h>
3 #include <TinyGPS++.h>
4 SoftwareSerial F2EMSerial;
5 #include <Time.h>
6 #include <Wire.h>
7 #include <ModbusMaster.h>
8 #define MAX485_DE 16
9 #define MAX485_SE 5
10 static uint8_t ppsmSlaveAddr = 0x01;
11 static uint16_t NewshuntAddr = 0x0000;
12 ModbusMaster node;
13 float F2EMVoltage = 0;
14 float F2EMCurrent = 0;
15 float F2EMPower = 0;
16 float F2EMEnergy = 0;
17 unsigned long startMillisF2EM;
18 unsigned long currentMillisF2EM;
19 const unsigned long periodF2EM = 1000;
20 unsigned long startMillisReadData;
21 unsigned long currentMillisReadData;
22 const unsigned long periodReadData = 1000;
23 int ResetEnergy = 0;
24 int a = 1;
25 unsigned long startMillis;
26 int timesone = 7 * 3600;

```

(ก)

```

Project_Learn_21 (Arduino 1.8.9)
File Edit Search Tools Help
Project_Learn_21.h
27 int dnt = 0;
28 #include <ESP8266WiFi.h>
29 #include <espnow.h>
30 uint8_t BroadcastAddress[] = {0x78, 0x21, 0x94, 0x08, 0x76, 0xAC};
31 typedef struct struct_message {
32     float a,b,c,d;
33 } struct_message;
34 struct_message myData;
35 void OnDataRecv(const uint8_t * mac_addr, uint8_t * sendStatus) {
36     Serial.print("Last Packet Send Status: ");
37     if (sendStatus == 0) {
38         Serial.println("Delivery success");
39     }
40     else {
41         Serial.println("Delivery fail");
42     }
43 }
44 void setup()
45 {
46     Serial.begin(115200);
47     startMillis = millis();
48     Serial.begin(9600);
49     F2EMSerial.begin(9600, SWSERIAL_8N2, 4, 0);
50     Serial.println(" ");
51     startMillisF2EM = millis();
52     pinMode(MAX485_SE, OUTPUT);

```

(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Project_Lab04_21 Arduino 1.8.7
File Edit Sketch Tools Help

Project_Lab04_21
53 pinMode(MAX485_DE, OUTPUT);
54 digitalWrite(MAX485_DE, 0);
55 digitalWrite(MAX485_DE, 0);
56 node.preTransmission(preTransmission);
57 node.postTransmission(postTransmission);
58 node.begin(psemSlaveAddr, P2EMSerial);
59 delay(1000);
60 startMillisReadData = millis();
61 Serial.begin(115200);
62 WiFi.mode(WIFI_STA);
63 Serial.println(WIFI_macAddress());
64 if (esp_now_init() != 0) {
65     Serial.println("Error initializing ESP-NOW");
66     return;
67 }
68 esp_now_set_self_role(ESP_NOW_ROLE_CONTROLLER);
69 esp_now_register_send_cb(OnDataSent);
70 esp_now_md_peer(broadcastAddress, ESP_NOW_ROLE_SLAVE, 1, NULL, 0);
71 }
72 void loop()
73 {
74     if ((millis() - startMillis) >= 10000) && (a == 1)
75     {
76         setShunt(psemSlaveAddr);
77         changeAddress(0xFS, psemSlaveAddr);
78         a = 0;

```

(ค)

```

Project_Lab04_21 Arduino 1.8.7
File Edit Sketch Tools Help

Project_Lab04_21
79 }
80 currentMillisP2EM = millis();
81 if (currentMillisP2EM - startMillisP2EM >= periodP2EM)
82 {
83     uint8_t result;
84     result = node.readInputRegisters(0x0000, 6);
85     if (result == node.kStatusSuccess)
86     {
87         uint32_t tempdouble = 0x00000000;
88         P2EMVoltage = node.getResponseBuffer(0x0000) / 100.0;
89         P2EMCurrent = node.getResponseBuffer(0x0001) / 100.0;
90         tempdouble = (node.getResponseBuffer(0x0003) << 16) + node.getResponseBuffer(0x0002);
91         P2EMPower = tempdouble / 10.0;
92         tempdouble = (node.getResponseBuffer(0x0005) << 16) + node.getResponseBuffer(0x0004);
93         P2EMEnergy = tempdouble;
94         if (psemSlaveAddr == 2)
95         {
96             }
97         }
98         else
99         {
100             }
101         startMillisP2EM = currentMillisP2EM;
102     }
103     currentMillisReadData = millis();
104     if (currentMillisReadData - startMillisReadData >= periodReadData)

```

(ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

105 {
106     Serial.print("Vdc : "); Serial.print(PZEMVoltage); Serial.println(" V");
107     Serial.print("Idc : "); Serial.print(PZEMCurrent); Serial.println(" A");
108     Serial.print("Power : "); Serial.print(PZEMPower); Serial.println(" W");
109     Serial.print("Energy : "); Serial.print(PZEMEnergy); Serial.println(" Wh ");
110     startMillisReadData = millis();
111 }
112 myData.a = PZEMVoltage;
113 myData.b = PZEMCurrent;
114 myData.c = PZEMPower;
115 myData.d = PZEMEnergy;
116 esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
117 delay(2000);
118 }
119 void preTransmission()
120 {
121     if(millis() - startMillis > 5000)
122     {
123         digitalWrite(MAX485_RE, 1);
124         digitalWrite(MAX485_DE, 1);
125         delay(1);
126     }
127 }
128 void postTransmission()
129 {
130     if(millis() - startMillis > 5000)

```

(จ)

```

131 {
132     delay(1);
133     digitalWrite(MAX485_RE, 0);
134     digitalWrite(MAX485_DE, 0);
135 }
136 }
137 void setShunt(uint8_t slaveAddr)
138 {
139     static uint8_t SlaveParameter = 0x06;
140     static uint16_t registerAddress = 0x0003;
141     uint16_t ui16CRC = 0xFFFF;
142     ui16CRC = crc16_update(ui16CRC, slaveAddr);
143     ui16CRC = crc16_update(ui16CRC, SlaveParameter);
144     ui16CRC = crc16_update(ui16CRC, highByte(registerAddress));
145     ui16CRC = crc16_update(ui16CRC, lowByte(registerAddress));
146     ui16CRC = crc16_update(ui16CRC, highByte(NewshuntAddr));
147     ui16CRC = crc16_update(ui16CRC, lowByte(NewshuntAddr));
148     preTransmission();
149     PZEMSerial.write(slaveAddr);
150     PZEMSerial.write(SlaveParameter);
151     PZEMSerial.write(highByte(registerAddress));
152     PZEMSerial.write(lowByte(registerAddress));
153     PZEMSerial.write(highByte(NewshuntAddr));
154     PZEMSerial.write(lowByte(NewshuntAddr));
155     PZEMSerial.write(lowByte(ui16CRC));
156     PZEMSerial.write(highByte(ui16CRC));

```

(ฉ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

156   FEMSerial.write(highByte(ui6CRC));
157   delay(10);
158   postTransmission();
159   delay(100);
160 }
161 void changeAddress(uint8_t OldslaveAddr, uint8_t NewslaveAddr)
162 {
163     static uint8_t SlaveParameter = 0x06;
164     static uint16_t registerAddress = 0x0002;
165     uint16_t ui6CRC = 0xFFFF;
166     ui6CRC = crc16_update(ui6CRC, OldslaveAddr);
167     ui6CRC = crc16_update(ui6CRC, SlaveParameter);
168     ui6CRC = crc16_update(ui6CRC, highByte(registerAddress));
169     ui6CRC = crc16_update(ui6CRC, lowByte(registerAddress));
170     ui6CRC = crc16_update(ui6CRC, highByte(NewslaveAddr));
171     ui6CRC = crc16_update(ui6CRC, lowByte(NewslaveAddr));
172     preTransmission();
173     FEMSerial.write(OldslaveAddr);
174     FEMSerial.write(SlaveParameter);
175     FEMSerial.write(highByte(registerAddress));
176     FEMSerial.write(lowByte(registerAddress));
177     FEMSerial.write(highByte(NewslaveAddr));
178     FEMSerial.write(lowByte(NewslaveAddr));
179     FEMSerial.write(lowByte(ui6CRC));
180     FEMSerial.write(highByte(ui6CRC));
181     delay(10);

```

(๗)

```

159     delay(100);
160 }
161 void changeAddress(uint8_t OldslaveAddr, uint8_t NewslaveAddr)
162 {
163     static uint8_t SlaveParameter = 0x06;
164     static uint16_t registerAddress = 0x0002;
165     uint16_t ui6CRC = 0xFFFF;
166     ui6CRC = crc16_update(ui6CRC, OldslaveAddr);
167     ui6CRC = crc16_update(ui6CRC, SlaveParameter);
168     ui6CRC = crc16_update(ui6CRC, highByte(registerAddress));
169     ui6CRC = crc16_update(ui6CRC, lowByte(registerAddress));
170     ui6CRC = crc16_update(ui6CRC, highByte(NewslaveAddr));
171     ui6CRC = crc16_update(ui6CRC, lowByte(NewslaveAddr));
172     preTransmission();
173     FEMSerial.write(OldslaveAddr);
174     FEMSerial.write(SlaveParameter);
175     FEMSerial.write(highByte(registerAddress));
176     FEMSerial.write(lowByte(registerAddress));
177     FEMSerial.write(highByte(NewslaveAddr));
178     FEMSerial.write(lowByte(NewslaveAddr));
179     FEMSerial.write(lowByte(ui6CRC));
180     FEMSerial.write(highByte(ui6CRC));
181     delay(10);
182     postTransmission();
183     delay(100);
184 }

```

(๗)

รูปที่ ข.1 Source Code esp8266

ใน code ชุดนี้มีการใช้โมดูล 3 ชนิดคือ DHT11 Pzem-017 และ Max485 โดย code นี้มีการคำนวณค่าที่ได้จากเซ็นเซอร์การแปลงข้อมูลที่ได้จากโมดูล

ใน code ชุดนี้มีการใช้คำสั่ง Esp now โดยเป็นฝั่งส่งข้อมูลซึ่งเป็นการส่งข้อมูลผ่าน Wi-Fi ของตัวบอร์ด Esp ด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code TTGO t-beam esp32 lora

```

main
1 #include "configuration.h"
2 #include <rom/rtc.h>
3 #include <esp_now.h>
4 #include <WiFi.h>
5 #include <Wire.h>
6 #include <CayenneLPP.h>
7 #include <HardwareSerial.h>
8 #include "DHT.h"
9 DHT dht;
10 HardwareSerial ss(2);
11 float temperature;
12 typedef struct struct_message {
13     float a;
14 } struct_message;
15 struct_message myData;
16 void onDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
17     memcpy(myData, incomingData, sizeof(myData));
18     Serial.print("a: ");
19     Serial.println(myData.a);
20 CayenneLPP.lpp(51);
21 RTC_DATA_ATTR uint32_t count = 0;
22 void send() {
23     char buffer[40];
24     sprintf(buffer, sizeof(buffer), "Latitude: %10.6f\n", gpa_latitude());
25     screen_print(buffer);
26     sprintf(buffer, sizeof(buffer), "Longitude: %10.6f\n", gpa_longitude());

```

(ก)

```

main
27     screen_print(buffer);
28     sprintf(buffer, sizeof(buffer), "Error: %4.2f\n", gpa_hdop());
29     screen_print(buffer);
30     lpp.reset();
31     lpp.addGPS(1, gpa_latitude(), gpa_longitude(), gpa_altitude());
32     lpp.addTemperature(7, temperature);
33     lpp.addAnalogInput(4, myData.a);
34     if (LORAWAN_CONFIRMED_EVENT > 0)
35         bool_confirmed = (count % LORAWAN_CONFIRMED_EVENT == 0);
36     else
37         bool_confirmed = false;
38     #endif
39     ttn_cnt(count);
40     ttn_send(lpp.getBuffer(), lpp.getSize(), LORAWAN_PORT, confirmed);
41     count++;
42     void sleep() {
43     #if SEND_MESSAGES
44         char buffer[20];
45         sprintf(buffer, sizeof(buffer), "Sleeping in %1.1fs\n", (MESSAGE_TO_SLEEP_DELAY / 1000.0));
46         screen_print(buffer);
47         delay(MESSAGE_TO_SLEEP_DELAY);
48         screen_off();
49         sleep_interrupt(BUTTON_PIN, LOW);
50         uint32_t sleep_for = (millis() < SEND_INTERVAL) ? SEND_INTERVAL - millis() : SEND_INTERVAL;
51         sleep_millis(sleep_for);
52     #endif

```

(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

53 void callback(uint8_t message) {
54   if (EV_JOINING == message) screen_print("Joining TTN...\n");
55   if (EV_JOINED == message) screen_print("TTN joined\n");
56   if (EV_JOIN_FAILED == message) screen_print("TTN join failed\n");
57   if (EV_REJOIN_FAILED == message) screen_print("TTN rejoin failed\n");
58   if (EV_RESET == message) screen_print("Reset TTN connection\n");
59   if (EV_LINK_DEAD == message) screen_print("TTN link dead\n");
60   if (EV_ACK == message) screen_print("ACK received\n");
61   if (EV_FENDING == message) screen_print("Message discarded\n");
62   if (EV_QUEUED == message) screen_print("Message queued\n");
63   if (EV_TXCOMPLETS == message) {
64     screen_print("Message sent\n");
65     sleep(1);
66   }
67   if (EV_RESPONSE == message) {
68     screen_print("TTN Response: ");
69     uint8_t len = ttn_response_len();
70     uint8_t data[len];
71     ttn_response(data, len);
72     char buffer[64];
73     for (uint8_t i=0; i<len; i++) {
74       sprintf(buffer, sizeof(buffer), "%02X", data[i]);
75       screen_print(buffer);
76     }
77 }
78 uint32_t get_count() {

```

(ค)

```

79   return count;
80 }
81 void setup() {
82   WiFi.mode(WIFI_STA);
83   Serial.println(WiFi.macAddress());
84   if (esp_now_init() != ESP_OK) {
85     Serial.println("Error initializing ESP-NOW");
86     return;
87   }
88   esp_now_register_recv_cb(OnDataRecv);
89   #ifdef DEBUG_PORT
90     DEBUG_PORT.begin(SERIAL_BAUD);
91     pinMode(BUTTON_PIN, INPUT_PULLUP);
92     pinMode(LED_PIN, OUTPUT);
93     DEBUG_MSG(APP_NAME " * APP_VERSION " + APP_VERSION + "\n");
94   #endif
95   screen_setup();
96   Serial.begin(115200);
97   Serial.println();
98   dht_setup(2);
99   gpg_setup();
100   if (0 == count) {
101     screen_print(APP_NAME " * APP_VERSION, 0, 0);
102     screen_show_logo();
103     screen_update();
104     delay(LOGO_DELAY);
105 }

```

(ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

105 if (!ttn_setup()) {
106     screen_print("([ERR] radio module not found!\n");
107     delay(MESSAGE_TO_SLEEP_DELAY);
108     screen_off();
109     sleep_forever();
110 }
111 ttn_register(callback);
112 ttn_join();
113 ttn_sf(LORAWAN_SF);
114 ttn_addr(LORAWAN_ADDR);
115 }
116 void loop() {
117     //Serial.print(myData.a);
118     gps_loop();
119     ttn_loop();
120     screen_loop();
121     delay(dht.getMinimumSamplingPeriod());
122     float humidity = dht.getHumidity();
123     temperature = dht.getTemperature();
124     static uint32_t last = 0;
125     static bool first = true;
126     if (0 == last || millis() - last > SEND_INTERVAL) {
127         if (0 < gps_hdop() && gps_hdop() < 50 && gps_latitude() != 0 && gps_longitude() != 0) {
128             last = millis();
129             first = false;
130             send();
131         }

```

(จ)

```

116 void loop() {
117     //Serial.print(myData.a);
118     gps_loop();
119     ttn_loop();
120     screen_loop();
121     delay(dht.getMinimumSamplingPeriod());
122     float humidity = dht.getHumidity();
123     temperature = dht.getTemperature();
124     static uint32_t last = 0;
125     static bool first = true;
126     if (0 == last || millis() - last > SEND_INTERVAL) {
127         if (0 < gps_hdop() && gps_hdop() < 50 && gps_latitude() != 0 && gps_longitude() != 0) {
128             last = millis();
129             first = false;
130             send();
131         }
132         else {
133             if (first) {
134                 screen_print("Waiting GPS lock\n");
135                 first = false;
136                 if (millis() > GPS_WAIT_FOR_LOCK) {
137                     sleep();
138                 }
139             }
140         }
141     }

```

(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1 #include <Wire.h>
2 #include "SSD1306Wire.h"
3 #include "OLEDDisplay.h"
4 #include "images.h"
5 #include "fonts.h"
6 #define SCREEN_HEADER_HEIGHT 14
7 SSD1306Wire * display;
8 uint8_t screen_line = SCREEN_HEADER_HEIGHT - 1;
9 void screen_header() {
10     char buffer[20];
11     // Message count
12     sprintf(buffer, sizeof(buffer), "%03d", get_count() % 1000);
13     display->setTextAlignment(TEXT_ALIGN_LEFT);
14     display->drawString(0, 2, buffer);
15     // Datetime
16     gps_time(buffer, sizeof(buffer));
17     display->setTextAlignment(TEXT_ALIGN_CENTER);
18     display->drawString((display->getWidth())/2, 2, buffer);
19     // Satellite count
20     display->setTextAlignment(TEXT_ALIGN_RIGHT);
21     display->drawString((display->getWidth() - SATELLITE_IMAGE_WIDTH - 4, 2, stoa(gps_sats), buffer, 10));
22     display->drawXbm((display->getWidth() - SATELLITE_IMAGE_WIDTH, 0, SATELLITE_IMAGE_WIDTH, SATELLITE_IMAGE_HEIGHT, SATELLITE_IMAGE);
23 }
24 }
25 void screen_show_logo() {
26     uint8_t x = (display->getWidth() - TTN_IMAGE_WIDTH) / 2;

```

(๗)

```

27     uint8_t y = SCREEN_HEADER_HEIGHT + (display->getHeight() - SCREEN_HEADER_HEIGHT - TTN_IMAGE_HEIGHT) / 2 + 1;
28     display->drawXbm(x, y, TTN_IMAGE_WIDTH, TTN_IMAGE_HEIGHT, TTN_IMAGE);
29 }
30 void screen_off() {
31     display->displayOff();
32 }
33 void screen_on() {
34     display->displayOn();
35 }
36 void screen_clear() {
37     display->clear();
38 }
39 void screen_print(const char * text, uint8_t x, uint8_t y, uint8_t alignment) {
40     display->drawString(x, y, text);
41     display->setTextAlignment((OLEDDISPLAY_TEXT_ALIGNMENT) alignment);
42     display->drawString(x, y, text);
43 }
44 void screen_print(const char * text, uint8_t x, uint8_t y) {
45     screen_print(text, x, y, TEXT_ALIGN_LEFT);
46 }
47 }
48 void screen_print(const char * text) {
49     display->print(text);
50     if (screen_line + 6 > display->getHeight()) {
51         // scroll
52     }

```

(๗)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

50     if (_screen_line + 8 > display->getHeight()) {
51         // scroll
52     }
53     _screen_line += 8;
54     screen_loop();
55 }
56 void screen_update() {
57     display->display();
58 }
59 void screen_setup() {
60     // Display instance
61     display = new SSD1306Wire(OLED_ADDRESS, OLED_SDA, OLED_SCL);
62     display->init();
63     display->flipScreenVertically();
64     display->setFont(Custom_ArialMT_Plain_10);
65     // Scroll buffer
66     display->setLogBuffer(5, 30);
67 }
68 void screen_loop() {
69     display->clear();
70     screen_header();
71     display->drawLogBuffer(0, SCREEN_HEADER_HEIGHT);
72     display->display();
73 }

```

(ณ)

```

1 #include <esp_sleep.h>
2 void sleep_interrupt(uint8_t gpio, uint8_t mode) {
3     esp_sleep_enable_ext0_wakeup((gpio_num_t) gpio, mode);
4 }
5 void sleep_interrupt_mask(uint64_t mask, uint8_t mode) {
6     esp_sleep_enable_ext1_wakeup(mask, (esp_sleep_ext1_wakeup_mode_t) mode);
7 }
8 void sleep_millis(uint64_t ms) {
9     esp_sleep_enable_timer_wakeup(ms * 1000);
10    esp_deep_sleep_start();
11 }
12 void sleep_seconds(uint32_t seconds) {
13     esp_sleep_enable_timer_wakeup(seconds * 1000000);
14     esp_deep_sleep_start();
15 }
16 void sleep_forever() {
17     esp_deep_sleep_start();
18 }

```

(ญ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

1 #include <hal/hal.h>
2 #include <spi.h>
3 #include <vector>
4 #include "configuration.h"
5 #include "credentials.h"
6 #ifndef CFG_AS923
7 #error "This script is meant to connect to TTN AS network at 923Mhz"
8 #endif
9 const lmic_pinmap lmic_pins = {
10     .nss = NSS_GPIO,
11     .rxtx = LMIC_UNUSED_PIN,
12     .rst = RESET_GPIO,
13     .dio = (DIO0_GPIO, DIO1_GPIO, DIO2_GPIO),
14 };
15 #ifdef USE_APP
16 void on_getArtEui (ui_t* buf) { }
17 void on_getDevEui (ui_t* buf) { }
18 void on_getDevKey (ui_t* buf) { }
19 #endif
20 #ifdef USE_OTAA
21 void on_getArtEui (ui_t* buf) { memcpy_P(buf, APPEUI, 8); }
22 void on_getDevEui (ui_t* buf) { memcpy_P(buf, DEVEUI, 8); }
23 void on_getDevKey (ui_t* buf) { memcpy_P(buf, APPKEY, 16); }
24 #endif
25 std::vector<void(*) (uint8_t message)> _lmic_callbacks;
26 void ttn_callback(uint8_t message) {

```

(๓)

```

27     for (uint8_t i=0; i<_lmic_callbacks.size(); i++) {
28         _lmic_callbacks[i](message);
29     }
30 }
31 void onEvent(ev_t event) {
32     if (EV_TXCOMPLETE == event) {
33         if (LMIC.txrxFlags & TXRX_ACK) {
34             ttn_callback(EV_ACK);
35         }
36         if (LMIC.dataLen) {
37             ttn_callback(EV_RESPONSE);
38         }
39     }
40     ttn_callback(event);
41 }
42 void ttn_register(void (*callback)(uint8_t message)) {
43     _lmic_callbacks.push_back(callback);
44 }
45 size_t ttn_response_len() {
46     return LMIC.dataLen;
47 }
48 void ttn_response(uint8_t* buffer, size_t len) {
49     for (uint8_t i = 0; i < LMIC.dataLen; i++) {
50         buffer[i] = LMIC.frame[LMIC.databeg + i];
51     }
52 }

```

(๓)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

53 bool ttn_setup() {
54     SPI.begin(SCK_GPIO, MISO_GPIO, MOSI_GPIO, NSS_GPIO);
55     return ( 1 == os_init_ex( (const void *) &lmic_pins ) );
56 }
57
58 void ttn_join() {
59     LMIC_reset();
60
61     #ifdef USE_ABP
62         uint8_t appkey[sizeof( APPSKEY)];
63         uint8_t nwkkey[sizeof( NWSKEY)];
64         memcpy_P( appkey, APPSKEY, sizeof( APPSKEY));
65         memcpy_P( nwkkey, NWSKEY, sizeof( NWSKEY));
66         LMIC_setSession(0x01, DEVADDR, nwkkey, appkey);
67         LMIC_setupChannel(0, 923000000, DR_RANGE_MAP( DR_SF12, DR_SF7), BAND_CENTI); // g-band
68         LMIC_setupChannel(1, 923400000, DR_RANGE_MAP( DR_SF12, DR_SF7B), BAND_CENTI); // g-band
69         LMIC_setupChannel(2, 922000000, DR_RANGE_MAP( DR_SF12, DR_SF7), BAND_CENTI); // g-band
70         LMIC_setupChannel(3, 922200000, DR_RANGE_MAP( DR_SF12, DR_SF7), BAND_CENTI); // g-band
71         LMIC_setupChannel(4, 922400000, DR_RANGE_MAP( DR_SF12, DR_SF7), BAND_CENTI); // g-band
72         LMIC_setupChannel(5, 922600000, DR_RANGE_MAP( DR_SF12, DR_SF7), BAND_CENTI); // g-band
73         LMIC_setupChannel(6, 922800000, DR_RANGE_MAP( DR_SF12, DR_SF7), BAND_CENTI); // g-band
74         LMIC_setupChannel(7, 923000000, DR_RANGE_MAP( DR_SF12, DR_SF7), BAND_CENTI); // g-band
75         LMIC_setupChannel(8, 924800000, DR_RANGE_MAP( DR_FSK, DR_FSK), BAND_MILLI); // g2-band
76         LMIC_setLinkCheckMode(0);
77         LMIC_ds2Dx = DR_SF9;
78         LMIC_setDxTxpow( DR_SF7, 14);

```

(จ)

```

79         LMIC_ds2Dx = DR_SF9;
80         LMIC_setDxTxpow( DR_SF7, 14);
81         _ttn_callback( EV_JOINED);
82     #endif // USE_ABP
83 }
84
85 void ttn_sf( unsigned char sf ) {
86     LMIC_setDxTxpow( sf, 14);
87 }
88
89 void ttn_adf( bool enabled ) {
90     LMIC_setAdmMode( enabled);
91 }
92
93 void ttn_send( uint8_t * data, size_t len, uint8_t port, bool confirmed ) {
94     if ( LMIC_opmode & OP_TXRXEND ) {
95         _ttn_callback( EV_TXRXEND);
96         return;
97     }
98     LMIC_setTxData2( port, data, len, confirmed ? 1 : 0);
99     _ttn_callback( EV_QUEUE);
100 }
101
102 void ttn_loop() {
103     os_runloop_once();
104 }

```

(ข)

รูปที่ ข.2 Source Code TTGO t-beam esp32 lora

ใน code ชุดนี้มีการมีเชื่อมต่อกับ Esp now โดยเป็นฝั่งรับ เชื่อมต่อกับ LoRaWan เพื่อส่งข้อมูลไป LoRaWan และ Cayenne เพื่อแสดงผลในเว็บไซต์และแอปพลิเคชัน Cayenne

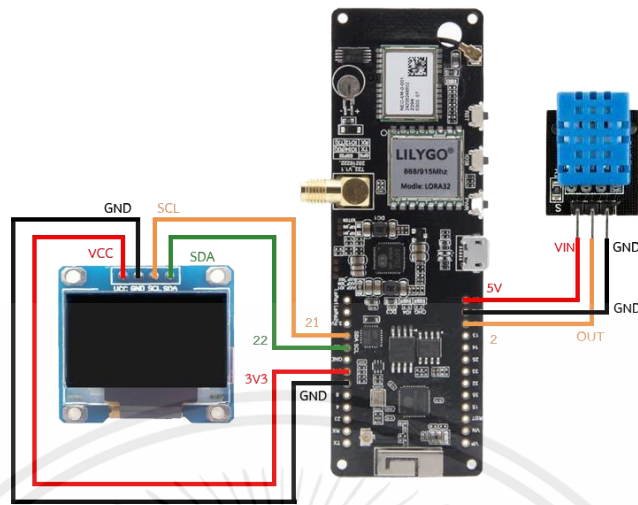
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



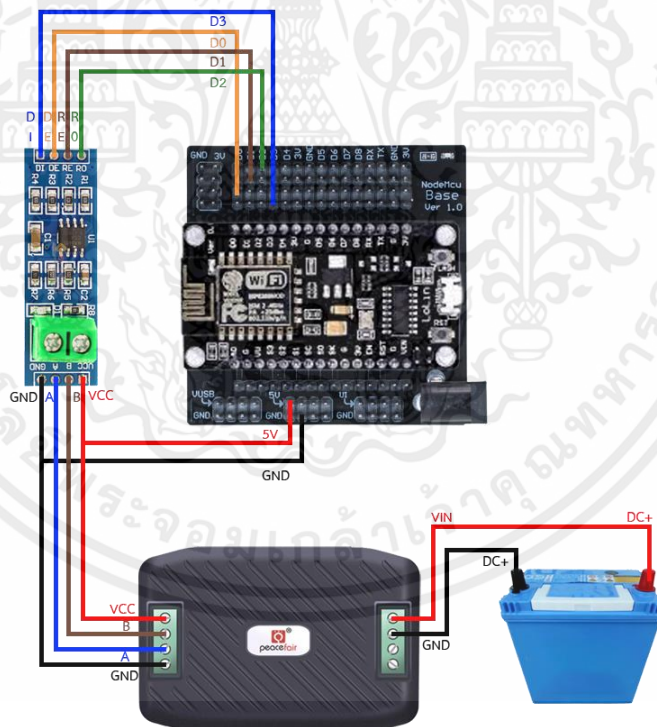
ภาคผนวก ค
คู่มือการใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.ต่อวงจรตามภาพต่อไปนี้



รูปที่ ค.1 การต่อวงจร TTGO t-beam esp32 lora



รูปที่ ค.2 การต่อวงจร ESP 8266

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. อัปเดต code ของ TTGO t-beam esp32 lora ลงบอร์ด

```

main()
{
  #include "configuration.h"
  #include <rom/rtc.h>
  #include <esp_now.h>
  #include <WiFi.h>
  #include <Wire.h>
  #include <CayenneLPP.h>
  #include <HardwareSerial.h>
  #include "DHT.h"
  #DHT dht;
  #HardwareSerial ss(2);
  #float temperature;
  #typedef struct struct_message {
  #  float a;
  #} struct_message;
  #struct_message myData;
  #void onDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
  #  memcpy(myData, incomingData, sizeof(myData));
  #  Serial.print("a: ");
  #  Serial.println(myData.a);
  #  CayenneLPP lpp(51);
  #  RTC_DATA_ATTR uint32_t count = 0;
  #  void send() {
  #    char buffer[40];
  #    sprintf(buffer, sizeof(buffer), "Latitude: %10.6f\n", gpa_latitude());
  #    screen_print(buffer);
  #    sprintf(buffer, sizeof(buffer), "Longitude: %10.6f\n", gpa_longitude());
  #  }
}

```

(ก)

```

screen_print(buffer);
sprintf(buffer, sizeof(buffer), "Error: %4.2f\n", gpa_hdop());
screen_print(buffer);
lpp.reset();
lpp.addGPS(1, gpa_latitude(), gpa_longitude(), gpa_altitude());
lpp.addTemperature(7, temperature);
lpp.addAnalogInput(4, myData.a);
#if LOZAMAN_CONFIRMED_EVENT > 0
  bool confirmed = (count % LOZAMAN_CONFIRMED_EVENT == 0);
#else
  bool confirmed = false;
#endif
ttn_cnt(count);
ttn_send(lpp, getBuffer(), lpp.getSize(), LOZAMAN_PORT, confirmed);
count++;
void sleep() {
  #if MESSAGE_BETWEEN_MESSAGES
  char buffer[20];
  sprintf(buffer, sizeof(buffer), "Sleeping in %1.1fs\n", (MESSAGE_TO_SLEEP_DELAY / 1000.0));
  screen_print(buffer);
  delay(MESSAGE_TO_SLEEP_DELAY);
  screen_off();
  sleep_interrupt(BUTTON_PIN, LOW);
  uint32_t sleep_for = (millis() < SEND_INTERVAL) ? SEND_INTERVAL - millis() : SEND_INTERVAL;
  sleep_millis(sleep_for);
  #endif
}

```

(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
53 void callback(uint8_t message) {
54   if (EV_JOINED == message) screen_print("Joining TTN...\n");
55   if (EV_JOINED == message) screen_print("TTN joined\n");
56   if (EV_JOIN_FAILED == message) screen_print("TTN join failed\n");
57   if (EV_REJOIN_FAILED == message) screen_print("TTN rejoin failed\n");
58   if (EV_RESET == message) screen_print("Reset TTN connection\n");
59   if (EV_LINK_DEAD == message) screen_print("TTN link dead\n");
60   if (EV_ACK == message) screen_print("ACK received\n");
61   if (EV_FENDING == message) screen_print("Message discarded\n");
62   if (EV_QUEUE == message) screen_print("Message queued\n");
63   if (EV_TXCOMPLETS == message) {
64     screen_print("Message sent\n");
65     sleep(1);
66   }
67   if (EV_RESPONSE == message) {
68     screen_print("TTN Response: ");
69     uint8_t len = ttn_response_len();
70     uint8_t data[len];
71     char buffer[64];
72     for (uint8_t i=0; i<len; i++) {
73       sprintf(buffer, "%02X", data[i]);
74       screen_print(buffer);
75     }
76   }
77 }
78 uint32_t get_count() {
```

(ค)

```
79   return count;
80 }
81 void setup() {
82   WiFi.mode(WIFI_STA);
83   Serial.println(WiFi.macAddress());
84   if (esp_now_init() != ESP_OK) {
85     Serial.println("Error initializing ESP-NOW");
86     return;
87   }
88   esp_now_register_recv_cb(OnDataRecv);
89   #ifdef DEBUG_PORT
90     DEBUG_PORT.begin(SERIAL_BAUD);
91     sendif(DEBUG_PORT.begin(SERIAL_BAUD));
92   #endif
93   pinMode(BUTTON_PIN, INPUT_PULLUP);
94   pinMode(LED_PIN, OUTPUT);
95   DEBUG_MSG(APP_NAME " * APP_VERSION " + APP_VERSION "\n");
96   screen_setup();
97   Serial.begin(115200);
98   Serial.println();
99   dht_setup(2);
100  gpa_setup();
101  if (0 == count) {
102    screen_print(APP_NAME " * APP_VERSION, 0, 0);
103    screen_show_logo();
104    screen_update();
105    delay(LOGO_DELAY);
106  }
107 }
```

(ง)

```
105  if (!ttn_setup()) {
106    screen_print("ERR! Radio module not found\n");
107    delay(MESSAGE_TO_SLEEP_DELAY);
108    screen_off();
109    sleep_forever();
110  }
111  ttn_register(callback);
112  ttn_join();
113  ttn_rf(LORAWAN_SF);
114  ttn_addr(LORAWAN_ADDR);
115 }
116 void loop() {
117   //Serial_print(myData.a);
118   gpa_loop();
119   ttn_loop();
120   screen_loop();
121   delay(dht.getMinimumSamplingPeriod());
122   float humidity = dht.getHumidity();
123   temperature = dht.getTemperature();
124   static uint32_t last = 0;
125   static bool first = true;
126   if (0 == last || millis() - last > SEND_INTERVAL) {
127     if (0 < gpa_hdop() && gpa_hdop() < 50 && gpa_latitude() != 0 && gpa_longitude() != 0) {
128       last = millis();
129       first = false;
130       send();
131     }
132   }
```

(จ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
116 void loop() {
117   //Serial.print(myData.a);
118   gpa_loop();
119   ttn_loop();
120   screen_loop();
121   delay(dht.getMinimumSamplingPeriod());
122   float humidity = dht.getHumidity();
123   temperature = dht.getTemperature();
124   static uint32_t last = 0;
125   static bool first = true;
126   if (0 == last || millis() - last > SEND_INTERVAL) {
127     if (0 < gpa_hdop() && gpa_hdop() < 50 && gpa_latitude() != 0 && gpa_longitude() != 0) {
128       last = millis();
129       first = false;
130       send();
131     } else {
132       if (first) {
133         screen_print("Waiting GPS lock\n");
134         first = false;
135       }
136       if (millis() > GPS_WAIT_FOR_LOCK) {
137         sleep();
138       }
139     }
140   }
141 }
```

(ฉ)

```
1 #include <Wire.h>
2 #include "SSD1306Wire.h"
3 #include "OLEDDisplay.h"
4 #include "images.h"
5 #include "fonts.h"
6 #define SCREEN_HEADER_HEIGHT 14
7 #define SCREEN_WIDTH * display;
8 uint8_t screen_line = SCREEN_HEADER_HEIGHT - 1;
9 void screen_header() {
10   char buffer[20];
11   // Message count
12   sprintf(buffer, sizeof(buffer), "%03d", get_count() % 1000);
13   display->setTextAlignment(TEXT_ALIGN_LEFT);
14   display->drawString(0, 2, buffer);
15   // Datetime
16   gps_time(buffer, sizeof(buffer));
17   display->setTextAlignment(TEXT_ALIGN_CENTER);
18   display->drawString(display->getWidth()/2, 2, buffer);
19   // Satellite count
20   display->setTextAlignment(TEXT_ALIGN_RIGHT);
21   display->drawString(display->getWidth() - SATELLITE_IMAGE_WIDTH - 4, 2, ltoa(gps_sats), buffer, 10);
22   display->drawImage(display->getWidth() - SATELLITE_IMAGE_WIDTH, 0, SATELLITE_IMAGE_WIDTH, SATELLITE_IMAGE_HEIGHT, SATELLITE_IMAGE);
23 }
24
25 void screen_show_logo() {
26   uint8_t x = (display->getWidth() - TTN_IMAGE_WIDTH) / 2;
```

(ช)

```
27   uint8_t y = SCREEN_HEADER_HEIGHT + (display->getHeight() - SCREEN_HEADER_HEIGHT - TTN_IMAGE_HEIGHT) / 2 + 1;
28   display->drawImage(x, y, TTN_IMAGE_WIDTH, TTN_IMAGE_HEIGHT, TTN_IMAGE);
29 }
30 void screen_off() {
31   display->displayOff();
32 }
33 void screen_on() {
34   display->displayOn();
35 }
36 void screen_clear() {
37   display->clear();
38 }
39 void screen_print(const char * text, uint8_t x, uint8_t y, uint8_t alignment) {
40   DEB_MSG(text);
41   display->setTextAlignment((OLEDDisplay_Text_Alignment) alignment);
42   display->drawString(x, y, text);
43 }
44 void screen_print(const char * text, uint8_t x, uint8_t y) {
45   screen_print(text, x, y, TEXT_ALIGN_LEFT);
46 }
47
48 void screen_print(const char * text) {
49   display->print(text);
50   if (_screen_line + 8 > display->getHeight()) {
51     // scroll
52   }
53 }
```

(ซ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

50     if (_screen_line + 8 > display->getHeight()) {
51         // scroll
52     }
53     _screen_line += 8;
54     screen_loop();
55 }
56 void screen_update() {
57     display->display();
58 }
59 void screen_setup() {
60     // Display instance
61     display = new SSD1306Wire(OLED_ADDRESS, OLED_SDA, OLED_SCL);
62     display->init();
63     display->flipScreenVertically();
64     display->setFont(Custom_ArialMT_Plain_10);
65     // Scroll buffer
66     display->setLogBuffer(5, 30);
67 }
68 void screen_loop() {
69     display->clear();
70     screen_header();
71     display->drawLogBuffer(0, SCREEN_HEADER_HEIGHT);
72     display->display();
73 }

```

(ณ)

```

1 #include <esp_sleep.h>
2 void sleep_interrupt(uint8_t gpio, uint8_t mode) {
3     esp_sleep_enable_ext0_wakeup(gpio_num_t) gpio, mode);
4 }
5 void sleep_interrupt_mask(uint64_t mask, uint8_t mode) {
6     esp_sleep_enable_ext1_wakeup(mask, (esp_sleep_ext1_wakeup_mode_t) mode);
7 }
8 void sleep_millis(uint64_t ms) {
9     esp_sleep_enable_timer_wakeup(ms * 1000);
10    esp_deep_sleep_start();
11 }
12 void sleep_seconds(uint32_t seconds) {
13    esp_sleep_enable_timer_wakeup(seconds * 1000000);
14    esp_deep_sleep_start();
15 }
16 void sleep_forever() {
17    esp_deep_sleep_start();
18 }

```

(ญ)

```

1 #include <hal/bsp.h>
2 #include <SPI.h>
3 #include <vector>
4 #include "configuration.h"
5 #include "credentials.h"
6 #ifndef CFG_A923
7     #error "This script is meant to connect to TTN AB network at 923MHz"
8 #endif
9 const Imic_pinsap imic_pins = {
10     .nss = NSS_GPIO,
11     .extx = IMIC_UNUSED_PIN,
12     .rst = RESET_GPIO,
13     .dio = {DIO0_GPIO, DIO1_GPIO, DIO2_GPIO},
14 };
15 #ifdef USE_ABP
16 void oa_getArtEui (ui_t* buf) { }
17 void oa_getDevEui (ui_t* buf) { }
18 void oa_getDevKey (ui_t* buf) { }
19 #endif
20 #ifdef USE_OTAA
21 void oa_getArtEui (ui_t* buf) { memcpy_F(buf, APPEUI, 8); }
22 void oa_getDevEui (ui_t* buf) { memcpy_F(buf, DEVEUI, 8); }
23 void oa_getDevKey (ui_t* buf) { memcpy_F(buf, APPKEY, 16); }
24 #endif
25 std::vector<void(*) (uint8_t message)> _imic_callbacks;
26 void _ttm_callback(uint8_t message) {

```

(ฉ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
27 for (uint8_t i=0; i<_lmic_callbacks.size(); i++) {
28     _lmic_callbacks[i](message);
29 }
30
31 void onEvent(ev_t event) {
32     if (KV_SUCCESSFUL == event) {
33         if (LMIC_TXRX_FLAGS & TXRX_ACK) {
34             _ttn_callback(EV_ACK);
35         }
36         if (LMIC_DATALEN) {
37             _ttn_callback(EV_RESPONSE);
38         }
39     }
40     _ttn_callback(event);
41 }
42 void ttn_register(void (*callback)(uint8_t message)) {
43     _lmic_callbacks.push_back(callback);
44 }
45 size_t ttn_response_len() {
46     return LMIC_DATALEN;
47 }
48 void ttn_response(uint8_t * buffer, size_t len) {
49     for (uint8_t i = 0; i < LMIC_DATALEN; i++) {
50         buffer[i] = LMIC.frame[LMIC_DATAREG + i];
51     }
52 }
```

(๓)

```
53 bool ttn_setup() {
54     #define LMIC_PINN_CFG MISO_GPIO, MISO_GPIO, MISO_GPIO, MISO_GPIO
55     return ( 1 == os_init_ex( (const void *) &lmic_pins ) );
56 }
57
58 void ttn_join() {
59     LMIC_reset();
60 }
61
62 #ifdef USE_ABP
63 uint8_t appkey[sizeof(APPKEY)];
64 memcpy_P(appkey, APPKEY, sizeof(APPKEY));
65 uint8_t nwkkey[sizeof(NWKKEY)];
66 memcpy_P(nwkkey, NWKKEY, sizeof(NWKKEY));
67 LMIC_setSession(0x1, CWIDREG, nwkkey, appkey);
68 LMIC_setupChannel(0, 923200000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENT); // g-band
69 LMIC_setupChannel(1, 923400000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND_CENT); // g-band
70 LMIC_setupChannel(2, 922000000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENT); // g-band
71 LMIC_setupChannel(3, 922200000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENT); // g-band
72 LMIC_setupChannel(4, 922400000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENT); // g-band
73 LMIC_setupChannel(5, 922600000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENT); // g-band
74 LMIC_setupChannel(6, 922800000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENT); // g-band
75 LMIC_setupChannel(7, 923000000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENT); // g-band
76 LMIC_setupChannel(8, 924800000, DR_RANGE_MAP(DR_SF8, DR_FS8), BAND_MILL); // g2-band
77 LMIC_setLinkCheckMode(0);
78 LMIC_dndr = DR_SF9;
79 LMIC_setTxPwr(DR_SF7, 14);
80 }
```

(๔)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

77 LMIC.dn2Dr = DR_SF9;
78 LMIC_setDtTpow(DR_SF7, 14);
79   ttn_callback(EV_JOINED);
80 #endif // USE_ABP
81
82
83 void ttn_sf(unsigned char sf) {
84   LMIC_setDtTpow(sf, 14);
85 }
86 void ttn_adr(bool enabled) {
87   LMIC_setAdrMode(enabled);
88 }
89 void ttn_cnt(unsigned long num) {
90   LMIC_setSeqnoOp(num);
91 }
92 void ttn_send(uint8_t * data, size_t len, uint8_t port, bool confirmed) {
93   if (LMIC.opmode & OP_TXRXPEND) {
94     ttn_callback(EV_TXRXPEND);
95     return;
96   }
97   LMIC_setTxData2(port, data, len, confirmed ? 1 : 0);
98   ttn_callback(EV_QUEUED);
99 }
100 void ttn_loop() {
101   os_runloop_once();
102 }

```

(๓)

รูปที่ ค.3 Source Code TTGO t-beam esp32 lora

3.อัปเดต code ลงบอร์ด Esp8266

```

1 #include <SoftwareSerial.h>
2 #include <Arduino.h>
3 #include <TinyGPS.h>
4 SoftwareSerial P2EMSerial;
5 #include <Time.h>
6 #include <Wire.h>
7 #include <ModbusMaster.h>
8 #define MAX485_DE 16
9 #define MAX485_RE 5
10 static uint8_t p2emSlaveAddr = 0x01;
11 static uint8_t p2emMasterAddr = 0x00;
12 ModbusMaster m;
13 float P2EMVoltage = 0;
14 float P2EMCurrent = 0;
15 float P2EMPower = 0;
16 float P2EMEnergy = 0;
17 unsigned long startMillisP2EM;
18 unsigned long currentMillisP2EM;
19 const unsigned long periodP2EM = 1000;
20 unsigned long startMillisReadData;
21 unsigned long currentMillisReadData;
22 const unsigned long periodReadData = 1000;
23 int ResultEnergy = 0;
24 int a = 1;
25 unsigned long startMillis1;
26 int timezone = 7 * 3600;

```

(ก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Project_Learn_21 Arduino 1.8.19
File Edit Search Tools Help

Project_Learn_21
27 int dst = 0;
28 #include <ESP8266WiFi.h>
29 #include <espnow.h>
30 uint8_t broadcastAddress[] = {0x78, 0x21, 0x84, 0x88, 0x76, 0xAC};
31 typedef struct struct_message {
32   float a,b,c,d;
33 } struct_message;
34 struct_message myData;
35 void onDataSent(uint8_t *mac_addr, uint8_t sendStatus) {
36   Serial.print("Last Packet Send Status: ");
37   if (sendStatus == 0)
38     Serial.println("Delivery success");
39   }
40   } else {
41     Serial.println("Delivery fail");
42   }
43 }
44 void setup()
45 {
46   Serial.begin(115200);
47   startMillis = millis();
48   Serial.begin(9600);
49   PZEMSerial.begin(SM600, SWSERIAL_SMD, 4, 0);
50   Serial.println("");
51   startMillisPZEM = millis();
52   pinMode(MAX485_RE, OUTPUT);
```

(ก)

```
Project_Learn_21 Arduino 1.8.19
File Edit Search Tools Help

Project_Learn_21
53 pinMode(MAX485_RE, OUTPUT);
54 digitalWrite(MAX485_RE, 0);
55 digitalWrite(MAX485_RE, 0);
56 node.preTransmission(preTransmission);
57 node.postTransmission(postTransmission);
58 node.begin(pzemsSlaveAddr, PZEMserial);
59 delay(1000);
60 startMillisReadData = millis();
61 Serial.begin(115200);
62 WiFi.mode(WIFI_STA);
63 Serial.println(WiFi.macAddress());
64 if (esp_now_init() != 0) {
65   Serial.println("Error initializing ESP-NOW");
66   return;
67 }
68 esp_now_set_self_role(ESP_NOW_ROLE_CONTROLLER);
69 esp_now_register_send_cb(onDataSent);
70 esp_now_add_peer(broadcastAddress, ESP_NOW_ROLE_SLAVE, 1, NULL, 0);
71
72 void loop()
73 {
74   if (millis() - startMillis >= 10000) { a = 1; }
75   }
76   setShunt(pzemsSlaveAddr);
77   changeAddress(0XFF, pzemsSlaveAddr);
78   a = 0;
79 }
```

(ค)

```
Project_Learn_21 Arduino 1.8.19
File Edit Search Tools Help

Project_Learn_21
79
80 currentMillisPZEM = millis();
81 if (currentMillisPZEM - startMillisPZEM >= periodPZEM)
82 {
83   uint8_t result;
84   result = node.readInputRegisters(0x0000, 6);
85   if (result == node.kuSMBSuccess)
86   {
87     uint32_t tempdouble = 0x00000000;
88     PZEMVoltage = node.getResponseBuffer(0x0000) / 100.0;
89     PZEMCurrent = node.getResponseBuffer(0x0001) / 100.0;
90     tempdouble = (node.getResponseBuffer(0x0003) << 16) + node.getResponseBuffer(0x0002);
91     PZEMPower = tempdouble / 10.0;
92     tempdouble = (node.getResponseBuffer(0x0005) << 16) + node.getResponseBuffer(0x0004);
93     PZEMEnergy = tempdouble;
94     if (pzemsSlaveAddr-->=2)
95     {
96       }
97     }
98   } else
99   {
100   }
101   startMillisPZEM = currentMillisPZEM ;
102 }
103 currentMillisReadData = millis();
104 if (currentMillisReadData - startMillisReadData >= periodReadData)
```

(ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

105
106 {
107     Serial.print("Vdc : "); Serial.print(PZEMVoltage); Serial.println(" V ");
108     Serial.print("Idc : "); Serial.print(PZEMCurrent); Serial.println(" A ");
109     Serial.print("Power : "); Serial.print(PZEMPower); Serial.println(" W ");
110     Serial.print("Energy : "); Serial.print(PZEMEnergy); Serial.println(" Wh ");
111     startMillisReadData = millis();
112 }
113 myData.a = PZEMVoltage;
114 myData.b = PZEMCurrent;
115 myData.c = PZEMPower;
116 myData.d = PZEMEnergy;
117 esp_now_send(broadcastAddress, (uint8_t *) myData, sizeof(myData));
118 delay(2000);
119
120 void preTransmission()
121 {
122     if(millis() - startMillis > 5000)
123     {
124         digitalWrite(DMAX485_RE, 1);
125         digitalWrite(DMAX485_DE, 1);
126         delay(1);
127     }
128 }
129 void postTransmission()
130 {
131     if(millis() - startMillis > 5000)

```

(จ)

```

131
132     delay(3);
133     digitalWrite(DMAX485_RE, 0);
134     digitalWrite(DMAX485_DE, 0);
135 }
136
137 void setShunt(uint8_t slaveAddr)
138 {
139     static uint8_t slaveParameter = 0x06;
140     static uint16_t registerAddress = 0x0003;
141     uint8_t u16CRC = 0xFFFF;
142     u16CRC = crc16_update(u16CRC, slaveAddr);
143     u16CRC = crc16_update(u16CRC, slaveParameter);
144     u16CRC = crc16_update(u16CRC, registerAddress);
145     u16CRC = crc16_update(u16CRC, lowByte(registerAddress));
146     u16CRC = crc16_update(u16CRC, highByte(registerAddress));
147     u16CRC = crc16_update(u16CRC, lowByte(NewshuntAddr));
148     u16CRC = crc16_update(u16CRC, highByte(NewshuntAddr));
149     preTransmission();
150     PZEMSerial.write(slaveAddr);
151     PZEMSerial.write(slaveParameter);
152     PZEMSerial.write(highByte(registerAddress));
153     PZEMSerial.write(lowByte(registerAddress));
154     PZEMSerial.write(lowByte(NewshuntAddr));
155     PZEMSerial.write(highByte(NewshuntAddr));
156     PZEMSerial.write(lowByte(u16CRC));
157     PZEMSerial.write(highByte(u16CRC));

```

(ข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

156     FEMSerial.write(highByte(ui6CRC));
157     delay(10);
158     postTransmission();
159     delay(100);
160 }
161 void changeAddress(uint8_t OldslaveAddr, uint8_t NewslaveAddr)
162 {
163     static uint8_t SlaveParameter = 0x06;
164     static uint16_t registerAddress = 0x0002;
165     uint16_t ui6CRC = 0xFFFF;
166     ui6CRC = crc16_update(ui6CRC, OldslaveAddr);
167     ui6CRC = crc16_update(ui6CRC, SlaveParameter);
168     ui6CRC = crc16_update(ui6CRC, highByte(registerAddress));
169     ui6CRC = crc16_update(ui6CRC, lowByte(registerAddress));
170     ui6CRC = crc16_update(ui6CRC, highByte(NewslaveAddr));
171     ui6CRC = crc16_update(ui6CRC, lowByte(NewslaveAddr));
172     preTransmission();
173     FEMSerial.write(OldslaveAddr);
174     FEMSerial.write(SlaveParameter);
175     FEMSerial.write(highByte(registerAddress));
176     FEMSerial.write(lowByte(registerAddress));
177     FEMSerial.write(highByte(NewslaveAddr));
178     FEMSerial.write(lowByte(NewslaveAddr));
179     FEMSerial.write(lowByte(ui6CRC));
180     FEMSerial.write(highByte(ui6CRC));
181     delay(10);

```

(๗)

```

159     delay(100);
160 }
161 void changeAddress(uint8_t OldslaveAddr, uint8_t NewslaveAddr)
162 {
163     static uint8_t SlaveParameter = 0x06;
164     static uint16_t registerAddress = 0x0002;
165     uint16_t ui6CRC = 0xFFFF;
166     ui6CRC = crc16_update(ui6CRC, OldslaveAddr);
167     ui6CRC = crc16_update(ui6CRC, SlaveParameter);
168     ui6CRC = crc16_update(ui6CRC, highByte(registerAddress));
169     ui6CRC = crc16_update(ui6CRC, lowByte(registerAddress));
170     ui6CRC = crc16_update(ui6CRC, highByte(NewslaveAddr));
171     ui6CRC = crc16_update(ui6CRC, lowByte(NewslaveAddr));
172     preTransmission();
173     FEMSerial.write(OldslaveAddr);
174     FEMSerial.write(SlaveParameter);
175     FEMSerial.write(highByte(registerAddress));
176     FEMSerial.write(lowByte(registerAddress));
177     FEMSerial.write(highByte(NewslaveAddr));
178     FEMSerial.write(lowByte(NewslaveAddr));
179     FEMSerial.write(lowByte(ui6CRC));
180     FEMSerial.write(highByte(ui6CRC));
181     delay(10);
182     postTransmission();
183     delay(100);
184 }

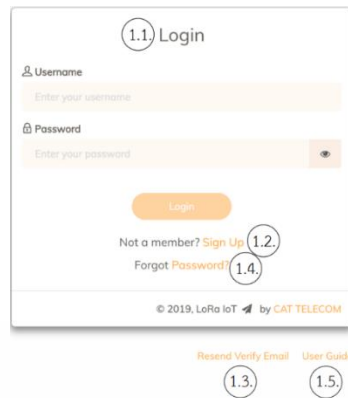
```

(๗)

รูปที่ ค.4 Source Code esp8266

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.การเข้าใช้งาน Web Portal CAT LoRaIoT Platform



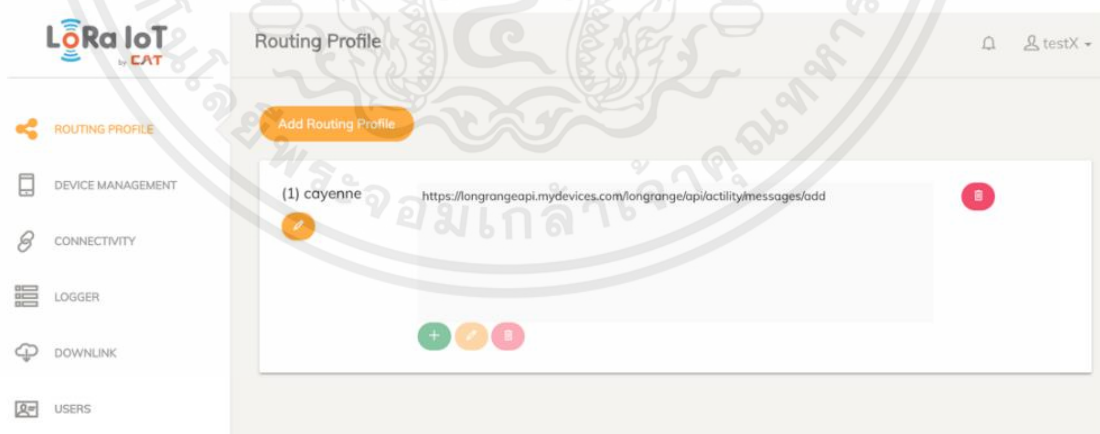
ประกอบด้วย

- 1.1. เข้าสู่ระบบ (Login)
- 1.2. การลงทะเบียนเข้าใช้งาน (Sign Up)
- 1.3. การส่งอีเมลยืนยันการสมัครเข้าใช้บริการอีกครั้ง (Resend Verify Email)
- 1.4. ลืมรหัสผ่าน (Forgot Password)
- 1.5. คู่มือการใช้งาน Web Portal (User Guide)



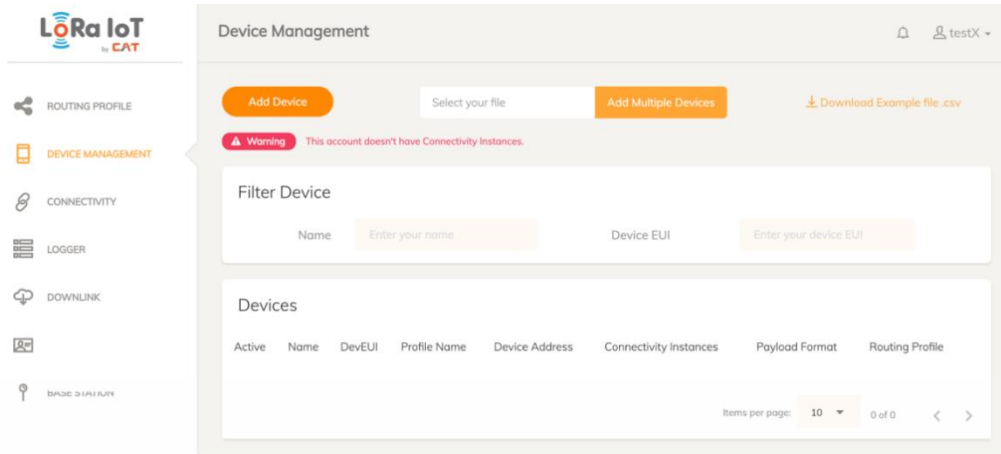
ประกอบด้วย

- 3.1. เมนูกำหนดโปรไฟล์การเชื่อมต่อ (Routing Profile)
- 3.2. เมนูจัดการอุปกรณ์ (Device Management)
- 3.3. เมนูจัดการอุปกรณ์เชื่อมต่อ (Connectivity)
- 3.4. เมนูการใช้งานข้อมูล (Logger)
- 3.5. เมนูการส่งข้อมูลจากโครงข่ายมายังอุปกรณ์ (Downlink)
- 3.6. เมนูจัดการสมาชิกผู้ใช้บริการ (Users)

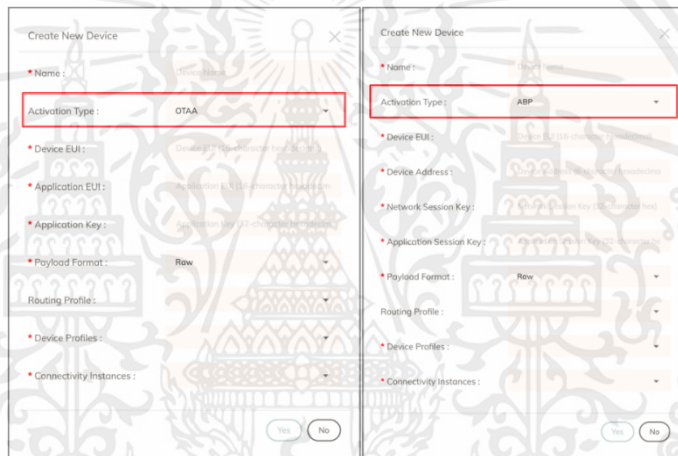


(ค)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ง)



(จ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

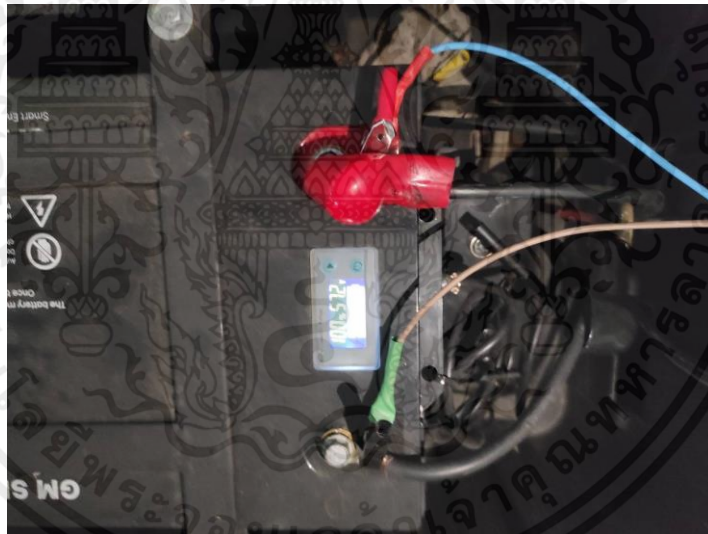
Logger Data

Timestamp	Device Address	DevEUI	Payload	FPort	FCnt _A	FCnt _D	RSSI	SNR	SubBand	Channel	LRR ID
05/05/2023 05:00:36	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	38	25	-111.000000	-7.750000	G1	LC1	10000C3C
05/05/2023 05:00:19	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	37	24	-111.000000	-8.000000	G1	LC2	10000C3C
05/05/2023 04:59:20	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	35	22	-110.000000	-5.500000	G1	LC2	10000C3C
05/05/2023 04:58:18	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	33	21	-109.000000	-5.500000	G1	LC1	10000C3C
05/05/2023 04:57:15	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	31	20	-109.000000	-4.250000	G1	LC1	10000C3C
05/05/2023 04:56:18	E01AEEC6	9984217800003C39	03020000040200000502000068800000000000000000	2	29	18	-109.000000	-6.750000	G1	LC1	10000C3C
05/05/2023 04:54:45	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	26	17	-112.000000	-7.500000	G1	LC2	10000C3C
05/05/2023 04:54:29	E01AEEC6	9984217800003C39	030213d804020c800502000068800000000000000000	2	25	16	-110.000000	-4.250000	G1	LC2	10000C3C

(ฉ)

รูปที่ ค.5 การใช้งาน Web Portal CAT LoRaloT Platform

5. ติดตั้งอุปกรณ์เข้ากับรถกอล์ฟ

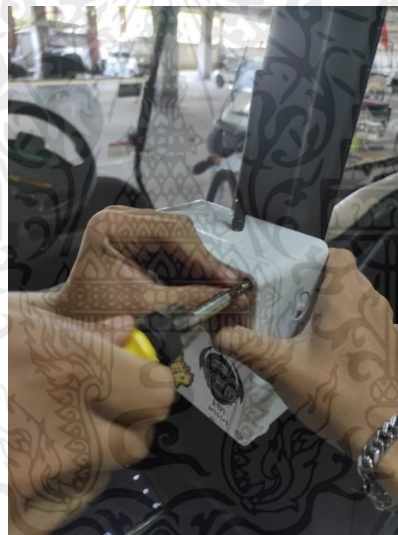


(ก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข)



(ค)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ง)

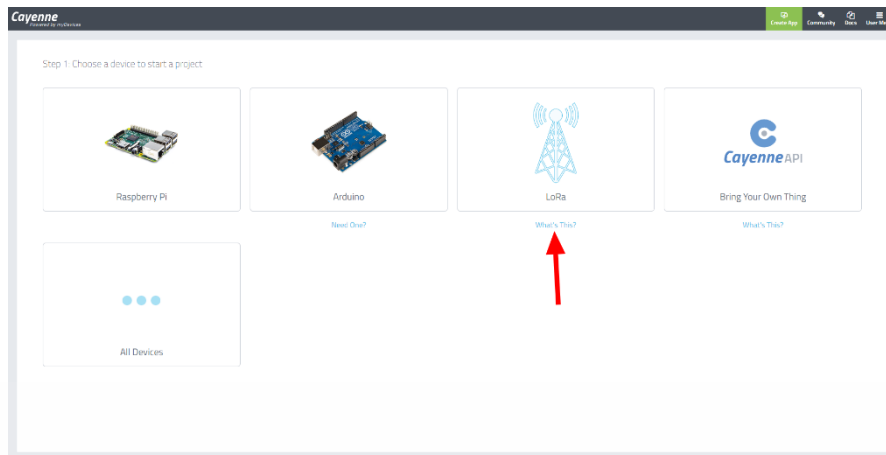
รูปที่ ค.6 การติดตั้งอุปกรณ์

6. การเข้าใช้งาน Web Application Cayenne

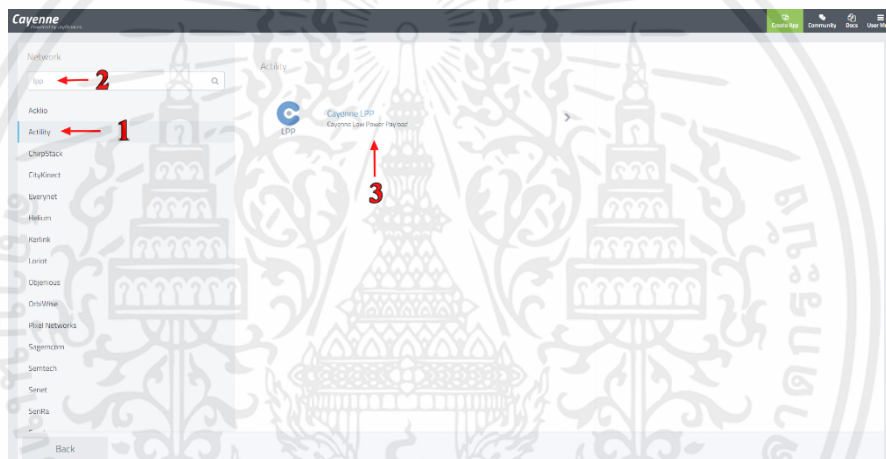


(ก)

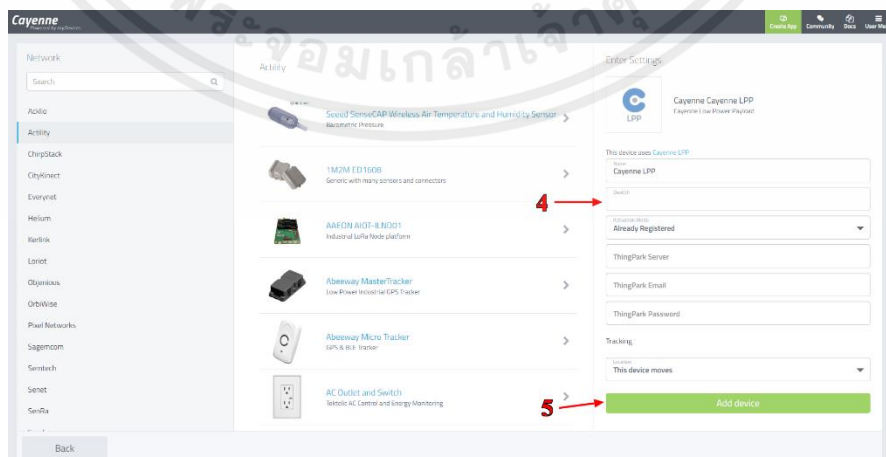
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข)

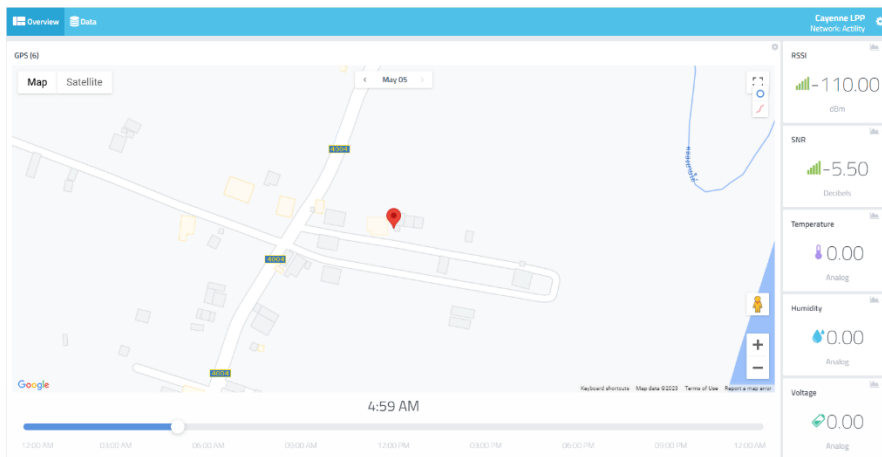


(ค)



(ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(จ)

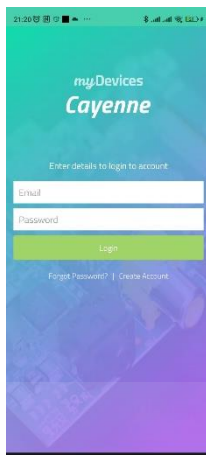
รูปที่ ค.7 การใช้งาน Web Application Cayenne

7. การเข้าใช้งาน Application Cayenne



(ก)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข)



(ค)

รูปที่ ค.8 การเข้าใช้งาน Application Cayenne

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้จัดทำ



ชื่อ-นามสกุล นายชนพล จิตภักดี
 วัน เดือน ปีเกิด วันที่ 24 กุมภาพันธ์ พ.ศ. 2540
 ที่อยู่ 78 ถนนหลวงวิวัฒนะ ตำบลตะกั่วป่า อำเภอตะกั่วป่า
 จังหวัดพังงา 82110 ประเทศไทย
 ประวัติการศึกษา พ.ศ. 2562 ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)
 แผนก เทคโนโลยีสารสนเทศ
 วิทยาลัยเทคนิคพังงา
 การติดต่อ โทรศัพท์ 061-2369729
 อีเมล 63201265@kmitl.ac.th



ชื่อ-นามสกุล นายวันเฉลิม เสียงเปรม
 วัน เดือน ปีเกิด วันที่ 5 ธันวาคม พ.ศ. 2540
 ที่อยู่ เลขที่ 200/1 หมู่ที่ 2 ตำบลตะโก อำเภอต่งตะโก
 จังหวัดชุมพร 86220 ประเทศไทย
 ประวัติการศึกษา พ.ศ. 2561 ประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.)
 แผนก คอมพิวเตอร์ธุรกิจ
 วิทยาลัยเกษตรและเทคโนโลยีชุมพร
 การติดต่อ โทรศัพท์ 098-7265453
 อีเมล 63201267@kmitl.ac.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้