

การควบคุมโดรนเพื่อการส่งอุปกรณ์ทางการแพทย์
DRONE CONTROL FOR MEDICAL EQUIPMENT DELIVERY



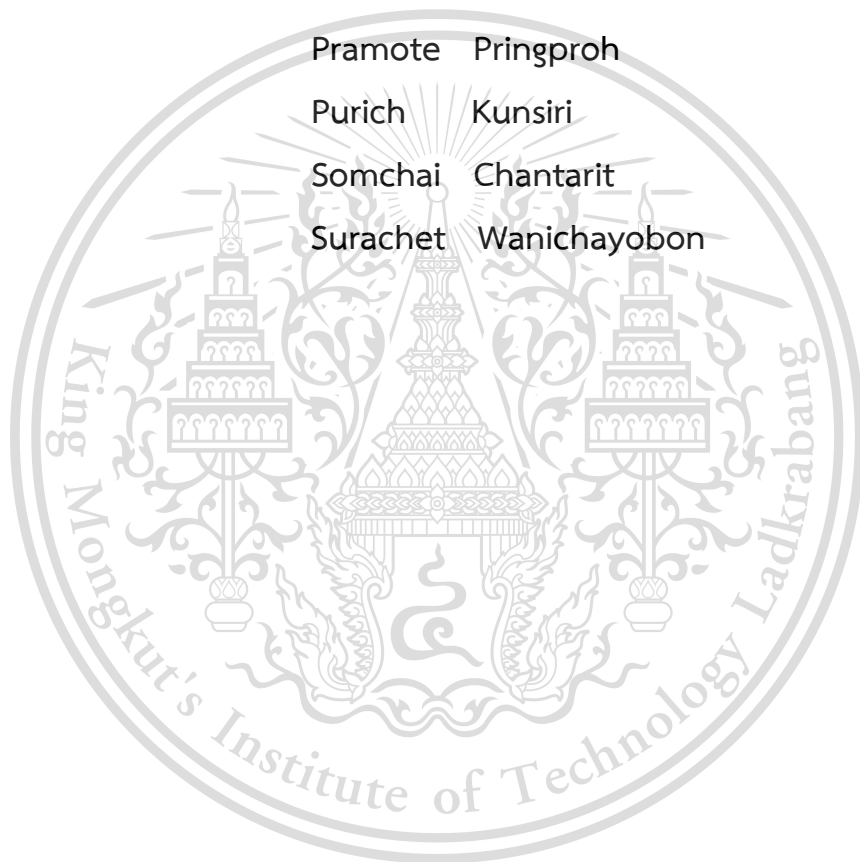
ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบควบคุม
คณะวิศวกรรมศาสตร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น มิใช่เพื่อเผยแพร่สู่สาธารณะโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา หรือทำซ้ำโดยไม่ได้รับอนุญาต
ปีการศึกษา 2563 ถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

DRONE CONTROL FOR MEDICAL EQUIPMENT DELIVERY



THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN CONTROL ENGINEERING
SCHOOL OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ACADEMIC YEAR 2020

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ปริญญาานิพนธ์ปีการศึกษา 2563

ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การควบคุมโดรนเพื่อการส่งอุปกรณ์ทางการแพทย์
DRONE CONTROL FOR MEDICAL EQUIPMENT DELIVERY

| | | | |
|----------|-------------|-----------|----------|
| ผู้จัดทำ | นายปราโมทย์ | พริ้งเพระ | 60010586 |
| | นายภูริช | กุลศิริ | 60010805 |
| | นายสมชัย | จันทฤทธิ | 60011013 |
| | นายสุรเชษฐ | วณิชโยบล | 60011105 |



.....อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.พิชชา ประสิทธิ์มีบุญ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

การควบคุมโดรนเพื่อการส่งอุปกรณ์ทางการแพทย์

โดย

นายปรามโทย์ พริ้งเพราะ 60010586

นายภูริช กุลศิริ 60010805

นายสมชัย จันทฤทธิ 60011013

นายสุรเชษฐ วณิชโยบล 60011105

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ ดร.พิชชา ประสิทธิ์มีบุญ

ปีการศึกษา 2563

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอการพัฒนาตัวต้นแบบของโดรนหรืออากาศยานไร้คนขับ เพื่อนำไปพัฒนาไปใช้ในการส่งอุปกรณ์ในทางการแพทย์ เนื่องจากปัจจุบันการจราจรในกรุงเทพมหานครค่อนข้างติดขัด หากเกิดสถานการณ์เร่งด่วน เช่น หากมีผู้ป่วยหมดสติต้องการการปฐมพยาบาลอย่างเร่งด่วน ซึ่งการรอการช่วยเหลือและเครื่องมือจากโรงพยาบาลอาจจะไม่ทันเวลา โดรนส่งอุปกรณ์ทางการแพทย์จะสามารถตอบสนองการแก้ปัญหานี้ได้ โดยการส่งอุปกรณ์ปฐมพยาบาลเบื้องต้น เช่น เครื่องกระตุกหัวใจ (AED) และช่วยชีวิตผู้ป่วยได้อย่างทันเวลา โดยโครงการนี้ครอบคลุมการออกแบบฟังก์ชันและอุปกรณ์ในการควบคุมการทำงานของโดรน ซึ่งประกอบด้วยควบคุมแบบ PID การใช้บอร์ดควบคุมและเซ็นเซอร์ต่างๆ การเขียนโปรแกรมด้วย Arduino IDE เพื่อควบคุมการทำงานของโดรน การสร้างแอปพลิเคชันด้วย Flutter เพื่อการระบุตำแหน่งปลายทางที่ต้องการส่งโดรน การทดสอบระบบการทำงานทั้งหมด และการปรับปรุงแก้ไขให้โดรนทำงานตามที่ต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

DRONE CONTROL FOR MEDICAL EQUIPMENT DELIVERY

By

Mr. Pramote Pringproh 60010586

Mr. Purich Kunsiri 60010805

Mr. Somchai Chantarit 60011013

Mr. Surachet Wanichayobon 60011105

Advisor

Asst.Prof.Dr.Pitcha Prasitmeeboon

Academic Year 2020

ABSTRACT

This thesis presents development of delivery drone which can be used for medical equipment delivery drones and rescue drones. Since Bangkok traffic is usually congested, it is hard for emergency patients to get help in time. Medical equipment delivery drone is a solution to the problem as drones can move without traffic problems. First aid equipment such AED can be delivered much faster than it is by an ambulance. This work includes studies and designs of the functions and devices to control the drone's operations using PID control, program design on Arduino IDE to control the drone's operations, development of an application with Flutter to locate drone's destination, and verification the entire system.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้ประสบความสำเร็จไปได้ด้วยดี ทั้งนี้ขอขอบพระคุณผู้ช่วยศาสตราจารย์ ดร.พิชชา ประสิทธิ์มีบุญ และอาจารย์ท่านอื่นๆ ในคณะวิศวกรรมศาสตร์ที่ได้อบรมสั่งสอนทั้งด้านวิชาการ และตักเตือนแนะนำแนวทางในการปฏิบัติงาน อีกทั้งยังช่วยแก้ไขปัญหาต่างๆ ตลอดการปฏิบัติงาน

ขอขอบพระคุณ Mr. Supun Dissanayaka ที่ช่วยให้ความรู้ต่างๆ เกี่ยวกับเครื่องบิน 4 ใบพัด และคอยแนะนำการแก้ไขปัญหาต่างๆ

ขอขอบพระคุณ บริษัท โซลูชั่น เซ็นเตอร์ จำกัด และคุณธราธิป เสนีวงศ์ ณ อยุธยา ที่ให้โอกาสมอบทุนในการดำเนินการโครงการครั้งนี้ อีกทั้งยังคอยรับฟังปัญหาต่างๆ และแนะนำแนวทางแก้ไข

ขอขอบคุณเพื่อนๆ ทุกคนที่ช่วยเหลือและให้กำลังใจในการทำโครงการนี้

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้ และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า

คณะผู้จัดทำ

ปราโมทย์ พริ้งเพราะ

ภูริช กุลศิริ

สมชัย จันทฤทธิ

สุรเชษฐ วณิชโยบล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and ^{III} cite the document when use.

สารบัญ

| | หน้า |
|--|------|
| บทคัดย่อภาษาไทย | VI |
| บทคัดย่อภาษาอังกฤษ | VI |
| กิตติกรรมประกาศ | III |
| สารบัญ | IV |
| สารบัญรูป | VIII |
| สารบัญตาราง | XII |
| บทที่ 1 บทนำ | |
| 1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์ | 1 |
| 1.2 วัตถุประสงค์ของปริญญานิพนธ์ | 1 |
| 1.3 ขอบเขตของโครงการ | 1 |
| 1.4 ขั้นตอนการดำเนินงาน | 2 |
| 1.5 ประโยชน์ที่คาดว่าจะได้รับ | 2 |
| บทที่ 2 ทฤษฎีที่เกี่ยวข้อง | |
| 2.1 โดรนสี่ใบพัด (Quadcopter) | 3 |
| 2.2 หลักการทำงานของโดรน | 3 |
| 2.2.1 การลอยตัวอยู่กับที่ (Hovering) | 4 |
| 2.2.2 การบินขึ้นลงของโดรน (Throttle) | 4 |
| 2.2.3 การเคลื่อนที่ไปด้านข้างของโดรน (Roll) | 4 |
| 2.2.4 การเคลื่อนที่ไปข้างหน้า-ถอยหลังของโดรน (Pitch) | 5 |
| 2.2.5 การหมุนตัวของโดรน (Yaw) | 5 |
| 2.3 มอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่าน (Brushless DC Motor) | 6 |
| 2.4 สัญญาณ PWM | 7 |
| 2.5 การสื่อสารข้อมูลแบบอนุกรม (Serial Communication) | 8 |
| 2.5.1 การสื่อสารข้อมูลแบบซิงโครนัส (Synchronous) | 8 |

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกห้ามมิทำซ้ำหรือดัดแปลงในเชิงพาณิชย์ของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

สารบัญ(ต่อ)

| | หน้า |
|--|------|
| 2.5.2 การสื่อสารข้อมูลแบบอะซิงโครนัส (Asynchronous) | 9 |
| 2.6 I ² C (Inter Integrate Circuit) | 10 |
| 2.6.1 การรับ-ส่งข้อมูลกับอุปกรณ์แบบ I ² C | 11 |
| 2.6.2 การกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I ² C | 11 |
| 2.6.3 รหัสควบคุมของ I ² C (Control Byte) | 12 |
| 2.7 แบตเตอรี่ลิเธียมโพลิเมอร์ | 12 |
| 2.7.1 แบตเตอรี่ลิเธียมโพลิเมอร์ | 13 |
| 2.7.2 ความจุแบตเตอรี่ | 13 |
| 2.7.3 ความสามารถในการจ่ายกระแส | 13 |
| 2.7.4 การสไตร์แบตเตอรี่ (Storage) | 13 |
| 2.8 ระบบระบุตำแหน่งบนพื้นโลก | 14 |
| 2.8.1 หลักการทำงานของ GPS | 14 |
| 2.8.2 GPS Protocol | 15 |
| 2.9 PID Controller | 17 |
| 2.9.1 ตัวควบคุมแบบสัดส่วน (Proportional: P) | 18 |
| 2.9.2 ตัวควบคุมปริพันธ์ (Integral: I) | 19 |
| 2.9.3 ตัวควบคุมอนุพันธ์ (Derivative: D) | 20 |
| บทที่ 3 วิธีการดำเนินงาน | |
| 3.1 ขั้นตอนการดำเนินงาน | 22 |
| 3.2 อุปกรณ์ที่เกี่ยวข้อง | 22 |
| 3.2.1 บอร์ดพัฒนา STM32F103C8T6 | 22 |
| 3.2.2 มอเตอร์ Dcw C2826 1290 KV | 23 |
| 3.2.3 แบตเตอรี่ลิเธียมโพลิเมอร์ 3S 11.1V | 23 |
| 3.2.4 Frame F450 450mm | 23 |

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งผู้ผลิตและผู้เผยแพร่เอกสารนี้ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

สารบัญ(ต่อ)

| | หน้า |
|--|------|
| 3.2.5 Electronic Speed Controller (ESC) | 24 |
| 3.2.6 บอร์ด PCB (Print Circuit Board) | 24 |
| 3.2.7 MC6C 6 channel remote control RC Transmitter and Receiver | 25 |
| 3.2.8 Ublox NEO-M8N GPS Module | 25 |
| 3.2.9 MPU6050 Module | 26 |
| 3.2.10 MS5611 Module | 26 |
| 3.2.11 HMC5883L Module | 26 |
| 3.2.12 FTDI FT232RL USB to TTL Serial Adapter | 27 |
| 3.3 ซอฟต์แวร์ที่เกี่ยวข้อง | 27 |
| 3.3.1 Arduino IDE | 27 |
| 3.3.2 Flutter | 27 |
| 3.3.3 Android Studio | 28 |
| 3.3.4 Firebase | 28 |
| 3.4 การออกแบบและการวางแผนการทำงาน | 28 |
| 3.4.1 การออกแบบและการวางแผนทางด้านฮาร์ดแวร์ | 28 |
| 3.4.2 การออกแบบและการวางแผนทางด้านซอฟต์แวร์ | 29 |
| 3.5 วิธีการดำเนินงาน | 29 |
| 3.5.1 การทดสอบเซ็นเซอร์ | 29 |
| 3.5.2 การต่อเซ็นเซอร์และอุปกรณ์ต่างๆ กับบอร์ดไมโครคอนโทรลเลอร์ | 33 |
| 3.5.3 การเขียนโปรแกรม Arduino | 34 |
| 3.5.4 การสร้างแอปพลิเคชันภาษา Flutter ด้วยโปรแกรม Android Studio | 42 |
| 3.5.5 ตั้งค่า Cloud Firestore บน Firebase | 45 |

บทที่ 4 ผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น 4.1 แผนผังการทำงานของโครงส่งอุปกรณ์ทางการแพทย์ 50

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use. VI

สารบัญ(ต่อ)

| | หน้า |
|---|------|
| 4.2 ผลการออกแบบหน้าจอดีแสดงผลผ่านแอปพลิเคชัน | 50 |
| 4.3 ผลการประดิษฐ์โครงส่งอุปกรณ์ทางการแพทย์ | 51 |
| 4.4 การทดสอบการส่งค่าตำแหน่งปัจจุบันระหว่างแอปพลิเคชันและอุปกรณ์รับ | 52 |
| บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ | |
| 5.1 สรุปผลการดำเนินงาน | 53 |
| 5.2 ปัญหาและอุปสรรคในการดำเนินงาน | 53 |
| 5.3 แนวทางการแก้ไขปัญหา | 54 |
| เอกสารอ้างอิง | 55 |
| ภาคผนวก | 60 |
| ภาคผนวก ก บอร์ดพัฒนา STM32F103C8T6 | 61 |
| ภาคผนวก ข ไมโคร MPU-6050 | 68 |
| ภาคผนวก ค ไมโคร MS5611 | 69 |
| ภาคผนวก ง ไมโคร HMC5883L | 70 |
| ภาคผนวก จ ไมโคร NEO-Ublox M8N | 71 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and ^{VII} cite the document when use.

สารบัญรูป

| รูปที่ | หน้า |
|---|------|
| 2.1 DJI Mini 2 Quadcopter | 3 |
| 2.2 การลอยตัวอยู่กับที่ของโดรน | 4 |
| 2.3 การบินขึ้นของโดรน | 4 |
| 2.4 การเคลื่อนที่ไปทางขวาของโดรน | 5 |
| 2.5 การเคลื่อนที่ถอยหลังของโดรน | 5 |
| 2.6 การหมุนตัวของโดรนในทิศทวนเข็มนาฬิกา | 6 |
| 2.7 ลักษณะของมอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่าน | 6 |
| 2.8 การผสมกันระหว่างสัญญาณรูปสามเหลี่ยมกับสัญญาณไซน์ | 7 |
| 2.9 ตัวอย่างสัญญาณ PWM ที่มี duty cycle แตกต่างกัน | 7 |
| 2.10 การส่งข้อมูลแบบซิงโครนัส | 9 |
| 2.11 การสื่อสารแบบอะซิงโครนัสที่ไม่ได้ใช้พาริตีบิต | 9 |
| 2.12 การสื่อสารแบบอะซิงโครนัสที่ใช้พาริตีบิต | 9 |
| 2.13 ลักษณะการเชื่อมต่อแบบ I ² C | 10 |
| 2.14 รูปแบบการเขียน/อ่านข้อมูลแบบ I ² C | 11 |
| 2.15 การกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I ² C บัส | 12 |
| 2.16 I ² C บัส Control Byte | 12 |
| 2.17 Discharge Voltage Curve ของแบตเตอรี่ลิเธียมโพลีเมอร์ | 14 |
| 2.18 Block Diagram ของ plant ที่ใช้ตัวควบคุมร่วมด้วย | 17 |
| 2.19 ผลตอบสนองเมื่อมีการปรับค่า K _p | 18 |
| 2.20 ผลตอบสนองเมื่อมีการปรับค่า K _i | 19 |
| 2.21 ผลตอบสนองเมื่อมีการปรับค่า K _d | 20 |
| 2.22 ลักษณะการทำงานของ PID Controller | 21 |

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|---|------|
| 3.2 Brushless Motor Dxm C2826 | 23 |
| 3.3 แบตเตอรี่ลิเทียมโพลิเมอร์ | 23 |
| 3.4 Frame F450 450mm | 24 |
| 3.5 Electronic Speed Controller | 24 |
| 3.6 บอร์ด PCB | 24 |
| 3.7 MC6C 2.4g 6 channel remote control RC Transmitter and Receiver | 25 |
| 3.8 Ublox NEO-M8N | 25 |
| 3.9 MPU6050 Module | 26 |
| 3.10 MS5611 Module | 26 |
| 3.11 HMC5883L Module | 26 |
| 3.12 FTDI FT232RL USB to TTL Serial Adapter | 27 |
| 3.13 Arduino IDE | 27 |
| 3.14 Flutter | 27 |
| 3.15 Android Studio | 28 |
| 3.16 Firebase | 28 |
| 3.17 การต่อโมดูล MPU6050 กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 | 29 |
| 3.18 ค่ามุมที่ได้จากโมดูล MPU6050 ในแนวระนาบกับพื้น | 30 |
| 3.19 ค่ามุมที่ได้จากโมดูล MPU6050 เมื่อขยับไปในทิศทางต่างๆ | 30 |
| 3.20 การต่อโมดูล MS5611 กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 | 31 |
| 3.21 ค่าความสูงที่ได้จากโมดูล MS5611 ที่ระดับพื้นดิน | 31 |
| 3.22 ค่าความสูงที่ได้จากโมดูล MS5611 เมื่อตำแหน่งของโมดูลมีการเปลี่ยนแปลง | 31 |
| 3.23 การต่อโมดูล HMC5883L กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 | 32 |

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ที่มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|--|------|
| 3.25 การต่อโมดูล Ublox Neo กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 | 33 |
| 3.26 ค่า NMEA GPS ที่โมดูล GPS อ่านได้ | 33 |
| 3.27 การต่อเซ็นเซอร์ และอุปกรณ์ต่างๆ กับบอร์ดไมโครคอนโทรลเลอร์ | 34 |
| 3.28 หน้าต่างโปรแกรม Arduino (1) | 34 |
| 3.29 หน้าต่างโปรแกรม Arduino (2) | 35 |
| 3.30 หน้าต่างโปรแกรม Arduino (3) | 35 |
| 3.31 หน้าต่างโปรแกรม Arduino (4) | 36 |
| 3.32 หน้าต่างโปรแกรม Arduino (5) | 36 |
| 3.33 หน้าต่างโปรแกรม Arduino (6) | 37 |
| 3.34 หน้าต่างโปรแกรม Arduino (7) | 37 |
| 3.35 หน้าต่างโปรแกรม Arduino (8) | 38 |
| 3.36 หน้าต่างโปรแกรม Arduino (9) | 38 |
| 3.37 หน้าต่างโปรแกรม Arduino (10) | 39 |
| 3.38 หน้าต่างโปรแกรม Arduino (11) | 39 |
| 3.39 หน้าต่างโปรแกรม Arduino (12) | 40 |
| 3.40 หน้าต่างโปรแกรม Arduino (13) | 40 |
| 3.41 หน้าต่างโปรแกรม Arduino (14) | 41 |
| 3.42 หน้าต่างโปรแกรม Arduino (15) | 41 |
| 3.43 หน้าต่างของโปรแกรม Android Studio ที่แสดงผลในหน้าแรก | 42 |
| 3.44 หน้าแรกของแอปพลิเคชัน | 42 |
| 3.45 หน้าต่างของโปรแกรม Android Studio ที่แสดงผลในหน้าสอง | 43 |
| 3.46 หน้าสองของแอปพลิเคชัน | 43 |

เอกสารนี้เป็นเอกสารต้นฉบับที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น ไม่สามารถนำเอกสารนี้ไปใช้ประโยชน์อื่นใดได้โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use. X

สารบัญรูป (ต่อ)

| รูปที่ | หน้า |
|---|------|
| 3.49 เว็บไซต์ Firebase ที่เป็นเซิร์ฟเวอร์ | 45 |
| 3.50 หน้าแรกของเว็บไซต์ Firebase | 45 |
| 3.51 หน้าตั้งชื่อ Project | 46 |
| 3.52 หน้าอนุญาตการใช้งานฟังก์ชันต่างๆ | 46 |
| 3.53 หน้าสุดท้ายของการสร้าง Project | 47 |
| 3.54 หน้าต่างเลือกระบบแอนดรอยด์การใช้งาน | 47 |
| 3.55 หน้าต่างหน้าเชื่อมกับแอปพลิเคชัน | 48 |
| 3.56 หน้าต่างเลือกฟังก์ชันการใช้งาน | 48 |
| 3.57 หน้าต่างแสดงค่าที่ถูกส่งขึ้นมาจากแอปพลิเคชัน | 49 |
| 4.1 แผนผังการทำงานของโดรนส่งอุปกรณ์ทางการแพทย์ | 50 |
| 4.2 ผลการออกแบบหน้าจอแสดงผลผ่านแอปพลิเคชัน | 51 |
| 4.3 โดรนที่ติดตั้งเซ็นเซอร์สำเร็จ | 51 |
| 4.4 เซิร์ฟเวอร์ที่ค่าตำแหน่งพิกัดส่งไป | 52 |
| 4.5 หน้าจอแสดงผลการส่งพิกัดสำเร็จ | 52 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and ^{XI} cite the document when use.

สารบัญตาราง

| ตารางที่ | หน้า |
|---|------|
| 2.1 ตัวอย่าง GGA | 15 |
| 2.2 ตัวอย่าง GLL | 16 |
| 2.3 ผลการปรับค่าตัวแปรพารามิเตอร์ทั้ง 3 ตัว | 21 |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and ^{XII} cite the document when use.

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปริญญานิพนธ์

ในปัจจุบันปัญหาการจราจรติดขัดบนท้องถนนในกรุงเทพฯหรือเขตปริมณฑลทำให้ทุกคนได้รับผลกระทบ โดยเฉพาะการเสียเวลาในการเดินทาง แต่การคมนาคมในกรุงเทพฯหรือเขตปริมณฑลเป็นสิ่งที่ไม่สามารถหลีกเลี่ยงได้ หน่วยกู้ภัยฉุกเฉินเป็นหนึ่งในหน่วยงานที่ได้รับผลกระทบอย่างมาก หากหน่วยกู้ภัยฉุกเฉินเดินทางเข้าไปถึงสถานที่ที่ผู้ป่วยอยู่ช้าเกินไปอาจทำให้ปฐมพยาบาลล่าช้าเกินไป ทำให้อาการของผู้ป่วยแย่ลงและไม่สามารถช่วยเหลือผู้ป่วยได้ทันเวลา การส่งโดรนที่ติดตั้งอุปกรณ์ทางการแพทย์เช่น เครื่องกระตุกหัวใจไฟฟ้า (AED) เข้าไปหาผู้ป่วยเพื่อทำการปฐมพยาบาลเบื้องต้นก่อนที่รถฉุกเฉินจะไปถึงอาจเป็นสิ่งจำเป็นเนื่องจากเส้นทางของรถฉุกเฉินไปหาผู้ป่วยขึ้นอยู่กับสถานที่ที่ผู้ป่วยอยู่ และหน่วยกู้ภัยที่อยู่ใกล้ที่สุด บางเส้นทางอาจใช้เวลานานขึ้นอยู่กับปัญหาการจราจรที่พบ การส่งโดรนหรืออากาศยานไร้คนขับเข้าไปหาผู้ป่วยสามารถทำได้อย่างรวดเร็ว ไม่พบปัญหาการจราจรติดขัดเหมือนอย่างรถฉุกเฉิน ทำให้คณะผู้จัดทำมีความสนใจในการศึกษา และพัฒนาโดรนตัวต้นแบบที่สามารถบินไปยังตำแหน่งที่ต้องการได้ โดยตำแหน่งที่ต้องการจะมาจากแอปพลิเคชันบนสมาร์ตโฟนในการระบุตำแหน่ง

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อช่วยเหลือการทำงานของหน่วยกู้ภัย
2. เพื่อสร้างโดรนตัวต้นแบบในการประยุกต์ในการใช้งานต่างๆ
3. เพื่อศึกษาการทำงานของโดรน
4. เพื่อศึกษาเกี่ยวกับการสร้างระบบ IoT (Internet of Thing)
5. เพื่อศึกษาการทำงานของตัวควบคุม PID
6. เพื่อศึกษาการทำงานของไมโครคอนโทรลเลอร์

1.3 ขอบเขตของโครงการ

1. ศึกษาการทำงานของโดรน วิธีการใช้งานเซ็นเซอร์ และอุปกรณ์อิเล็กทรอนิกส์ที่เกี่ยวข้อง
2. ศึกษาโปรแกรมที่นำมาใช้ในการควบคุมอุปกรณ์
3. ออกแบบ และพัฒนาแอปพลิเคชันให้สามารถใช้งานบนสมาร์ตโฟนได้

เอกสารนี้เป็นเอกสารที่สามารถนำส่วนหรือฉบับเต็มไปใช้ได้เฉพาะเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาค้นคว้าทฤษฎี และเอกสารที่เกี่ยวข้อง
2. ศึกษาหลักการของเซ็นเซอร์ที่จะนำมาใช้
3. ออกแบบโครงสร้าง และการทำงานของโครง
4. สั่งซื้อเซ็นเซอร์ และอุปกรณ์ต่างๆ
5. ทดสอบเซ็นเซอร์แต่ละตัวที่นำมาใช้
6. ประกอบโครงเข้ากับเซ็นเซอร์ทั้งหมด
7. เขียนโปรแกรมควบคุมการเคลื่อนที่ของโครง
8. ออกแบบ และสร้างแอปพลิเคชันสำหรับส่งตำแหน่งของผู้ป่วย
9. ทดสอบการทำงานของโครง และแก้ไขข้อผิดพลาด
10. สรุปผล และจัดทำเอกสารรายงานเป็นรูปเล่ม

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้เรียนรู้การสร้างโครง การเลือก และใช้งานอุปกรณ์ เช่น STM32F103C8T6
2. ได้รับความรู้เกี่ยวกับการใช้ API Google Map ระบุตำแหน่งของผู้ใช้เรียกโครงผ่านแอปพลิเคชันในโทรศัพท์
3. เพิ่มความสะดวกในการขนส่งของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

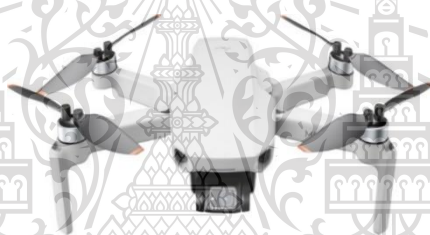
Forbidden to modify the content, and cite the document when use.

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 โดรนสี่ใบพัด (Quadcopter)

โดรน หรือ UAVs (Unmanned Aerial Vehicles) คือ อากาศยานไร้คนขับที่สามารถควบคุมจากภาคพื้นดินได้ด้วยรีโมท โดรนสี่ใบพัด (quadcopter) เป็นหนึ่งในประเภทของโดรน ใช้ใบพัดขับเคลื่อนทั้งหมดสี่ใบพัด โดยใบพัดจะอยู่ด้านบนของโดรนทั้งสี่มุมดังรูปที่ 2.1 โดรนประเภทนี้นิยมใช้ในการแข่งขันรวมถึงการบินเพื่อถ่ายภาพ เนื่องจากสามารถเคลื่อนที่ได้อย่างรวดเร็ว และคล่องแคล่วกว่าโดรนชนิดอื่น การเคลื่อนที่ของโดรนในทิศทางต่างๆ ได้แก่ การบินขึ้น-ลง เคลื่อนที่ไปข้างหน้า-ถอยหลัง เคลื่อนที่ไปด้านข้าง หมุนทวนเข็มนาฬิกา-ตามเข็มนาฬิกา สามารถทำได้โดยการควบคุมการเปลี่ยนแปลงความเร็วของใบพัดทั้งสี่ใบพัดที่ถูกจำกัดทิศทางการหมุนไว้ให้หมุนตามเข็มนาฬิกา และหมุนทวนเข็มนาฬิกา



รูปที่ 2.1 DJI Mini 2 Quadcopter [1]

ส่วนประกอบสำคัญที่ใช้สำหรับสร้างโดรนสี่ใบพัด ได้แก่ โครงโดรน, ใบพัดหมุนตามเข็มนาฬิกา, ใบพัดหมุนทวนเข็มนาฬิกา, มอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่าน, Electronic Speed Controller หรือ ESC, ไมโครคอนโทรลเลอร์ และแบตเตอรี่ โดยขนาดหรือความสามารถของแต่ละอุปกรณ์ที่ใช้ขึ้นอยู่กับลักษณะการใช้งานของโดรน

2.2 หลักการทำงานของโดรน

ใบพัดทั้งสี่จะถูกวางที่มุมทั้งสี่ของโดรนในระยะเท่าๆกัน โดยใบพัดคู่ตรงข้ามกันจะหมุนในทิศทางเดียวกัน หากใบพัดคู่หนึ่งหมุนในทิศตามเข็มนาฬิกาใบพัดคู่ที่เหลือจะหมุนในทิศทวนเข็มนาฬิกา ใบพัดทั้งสี่จะไม่สามารถหมุนในทิศเดียวกันทั้งหมดทุกใบพร้อมกันได้ หากใบพัดทั้งสี่หมุนในทิศเดียวกันโดรนจะหมุนเนื่องจากแรงบิด (torque) ที่เกิดจากใบพัดทั้งสี่ การใช้ใบพัดหมุนในทิศทางเดียวกันเป็นคู่แบบนี้จะช่วยยกเลิกแรงบิดที่เกิดจากใบพัดอีกคู่ ทำให้สามารถจัดสมดุลแรงบิดเพื่อ

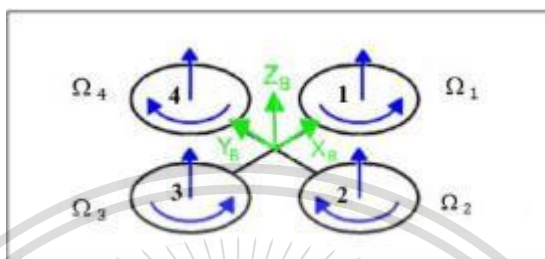
เอกสารนี้เป็นเอกสารที่เผยแพร่โดยมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี (KMITA) เพื่อใช้ในการศึกษาเท่านั้น ไม่สามารถนำเนื้อหาไปใช้เพื่อการค้า หรือเผยแพร่ในที่สาธารณะได้ [26]

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2.2.1 การลอยตัวอยู่กับที่ (Hovering)

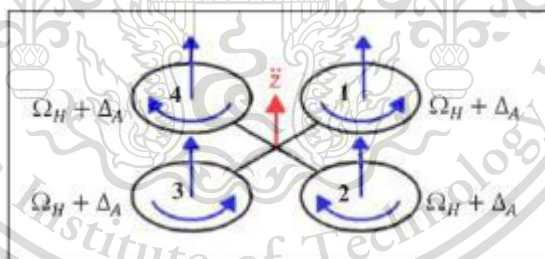
การลอยตัวอยู่กับที่ของโดรนสามารถทำได้โดยควบคุมความเร็วใบพัดทั้งสี่ให้มีความเร็วรอบที่เท่ากันเพื่อสร้างแรงยกในแนวตั้งที่ขนาดเท่ากับน้ำหนักของโดรนทำให้โดรนบินอยู่กับที่ โดยใบพัดคู่ตรงข้ามจะหมุนในทิศทางเดียวกันเพื่อหักล้างแรงบิดที่เกิดขึ้นจากใบพัดอีกคู่ที่หมุนในทิศทางตรงข้าม ทำให้โดรนไม่หมุนตัว [2]



รูปที่ 2.2 การลอยตัวอยู่กับที่ของโดรน [2]

2.2.2 การบินขึ้นลงของโดรน (Throttle)

การบินขึ้นของโดรนจะเกิดขึ้นเมื่อขนาดของแรงยกในแนวตั้งที่เกิดจากใบพัดทั้งสี่มากกว่าน้ำหนักของโดรน และการบินลงของโดรนเกิดจากขนาดแรงยกที่เกิดจากใบพัดทั้งสี่น้อยกว่าน้ำหนักของโดรน โดยที่ความเร็วรอบของใบพัดทั้งสี่จะต้องเท่ากันเหมือนเดิม [2]



รูปที่ 2.3 การบินขึ้นของโดรน [2]

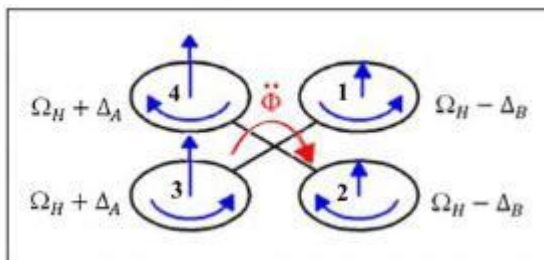
2.2.3 การเคลื่อนที่ไปด้านข้างของโดรน (Roll)

การเคลื่อนที่ไปทางซ้ายเกิดจากความเร็วนอกของคู่ใบพัดทางฝั่งด้านขวา (right) เพิ่มขึ้น และความเร็วนอกของคู่ใบพัดทางฝั่งด้านซ้าย (left) ลดลงอย่างเท่าๆกัน ส่วนการเคลื่อนที่ไปทางด้านขวาเกิดจากความเร็วนอกของคู่ใบพัดฝั่งด้านซ้าย (left) เพิ่มขึ้น และความเร็วนอกของคู่ใบพัดทางฝั่งด้านขวา (right) ลดลงอย่างเท่าๆกัน [2]

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

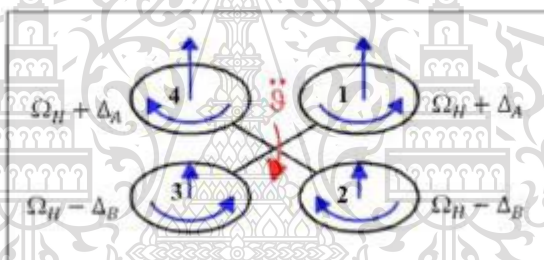
Forbidden to modify the content, and cite the document when use.



รูปที่ 2.4 การเคลื่อนที่ไปทางขวาของโดรน [2]

2.2.4 การเคลื่อนที่ไปข้างหน้า-ถอยหลังของโดรน (Pitch)

การเคลื่อนที่ไปข้างหน้าของโดรนเกิดจากความเร็รรอบของคูใบพัดทางฝั่งด้านหลัง (rear) เพิ่มขึ้น และความเร็รรอบของคูใบพัดทางฝั่งด้านหน้า (front) ลดลงอย่างเท่าๆกัน ส่วนการถอยหลังของโดรนเกิดจากความเร็รรอบของคูใบพัดทางฝั่งด้านหน้า (front) เพิ่มขึ้น และความเร็รรอบของคูใบพัดทางฝั่งด้านหลัง (rear) ลดลงอย่างเท่าๆกัน [2]



รูปที่ 2.5 การเคลื่อนที่ถอยหลังของโดรน [2]

ในขณะที่โดรนเคลื่อนที่ไปข้างหน้า-ถอยหลัง (pitch) หรือเคลื่อนที่ไปด้านข้าง (roll) แรงยกที่ได้จากใบพัดจะไม่อยู่ในแนวตั้งอีกต่อไป แต่จะอยู่ในแนวที่ทำมุมเอียงเล็กน้อยกับแนวตั้ง ทำให้แรงยกที่ได้จากใบพัดแตกออกเป็นสองแนว คือ แรงในแนวตั้ง และแรงในแนวระดับ ซึ่งแรงในแนวระดับจะทำให้โดรนเคลื่อนที่ไปข้างหน้า-ถอยหลัง (pitch) หรือเคลื่อนที่ไปด้านข้าง (roll) ในขณะที่แรงในแนวตั้งจะทำหน้าที่สมดุลกับน้ำหนักเพื่อให้โดรนลอยตัวในระดับความสูงที่คงที่ [2]

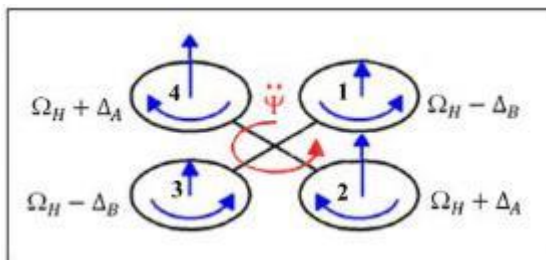
2.2.5 การหมุนตัวของโดรน (Yaw)

การหมุนตัวของโดรนเกิดจากความเร็รรอบของใบพัดคู่ตรงข้ามมีความเร็วมากกว่าใบพัดอีกคู่ หากต้องการหมุนโดรนทวนเข็มนาฬิกาจะต้องเพิ่มความเร็วของใบพัดด้านหน้าฝั่งซ้าย (left-front) กับใบพัดด้านหลังฝั่งขวา (right-rear) ซึ่งเป็นคู่ตรงข้ามกัน และลดความเร็วของใบพัดที่เหลือ หากต้องการหมุนโดรนตามเข็มนาฬิกาให้ทำวิธีตรงกันข้าม [2]

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์หรือการดำเนินงานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่อนำไปเผยแพร่หรือใช้เพื่อการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

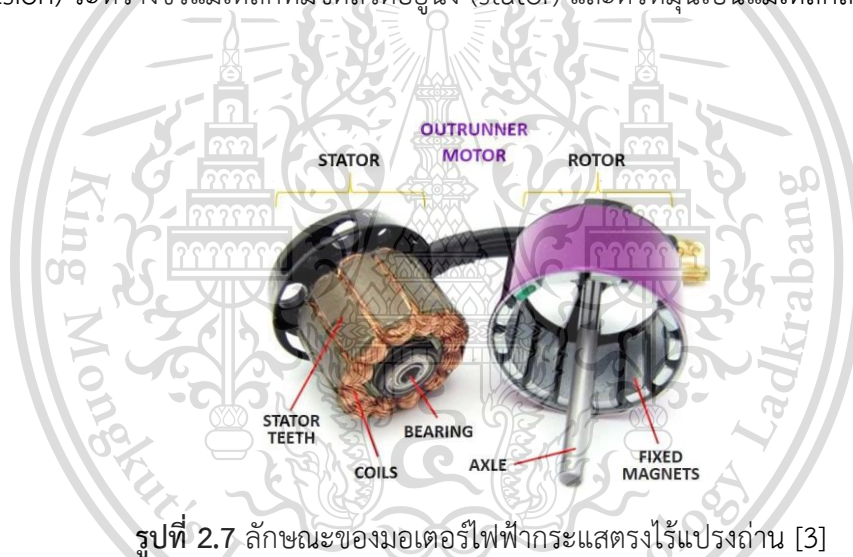
Forbidden to modify the content, and cite the document when use.



รูปที่ 2.6 การหมุนตัวของโรตอร์ในทิศทวนเข็มนาฬิกา [2]

2.3 มอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่าน (Brushless DC Motor)

มอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่านอาศัยหลักการเดียวกับมอเตอร์ไฟฟ้ากระแสสลับแบบซิงโครนัส (synchronous) สามเฟส ทำงานโดยการรับแรงดันไฟฟ้ากระแสตรงมาสร้างเป็นแรงดันไฟฟ้ากระแสสลับสามเฟสจ่ายให้กับมอเตอร์เพื่อให้เกิดแรงดูด (attraction) และแรงผลัก (repulsion) ระหว่างขั้วแม่เหล็กที่มีขดลวดอยู่นิ่ง (stator) และตัวหมุนเป็นแม่เหล็กถาวร (rotor)



รูปที่ 2.7 ลักษณะของมอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่าน [3]

มอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่านจะเริ่มหมุนเมื่อมีกระแสไหลผ่านขดลวดอยู่หนึ่งขดแรกจากทั้งหมดสามขด และสร้างขั้วแม่เหล็กนั้นเป็นแรงดูดให้สนามแม่เหล็ก จากนั้นเมื่อให้กระแสไหลผ่านขดลวดชุดต่อไปแม่เหล็กถาวรจะเคลื่อนที่ไปยังขดขดลวดนั้น การเปลี่ยนแปลงการจ่ายกระแสไฟฟ้าไปขดขดลวดแต่ละขดทำให้แม่เหล็กถาวรเกิดการเคลื่อนที่ไปเรื่อยๆ การเคลื่อนที่ของมอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่านจะใช้การทำงานครึ่งละสองเฟส ผลลัพธ์คือจะทำให้เกิดแรงบิดมากขึ้น เนื่องจากขดขดลวดชุดแรกจะสร้างแรงดูดให้แม่เหล็กถาวร และขดลวดอีกชุดจะสร้างแรงผลักให้แม่เหล็กถาวร และจ่ายกระแสให้ขดขดลวดคู่ที่อยู่ติดกันไป ทำให้มอเตอร์หมุนได้อย่างต่อเนื่อง โดย

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของ King Mongkut's Institute of Technology Ladkrabang. การสลับการจ่ายกระแสให้ขดขดลวดสามารถทำได้โดยการใช้ MOSFET ร่วมด้วย [29] ประโยชน์ด้านการค้า

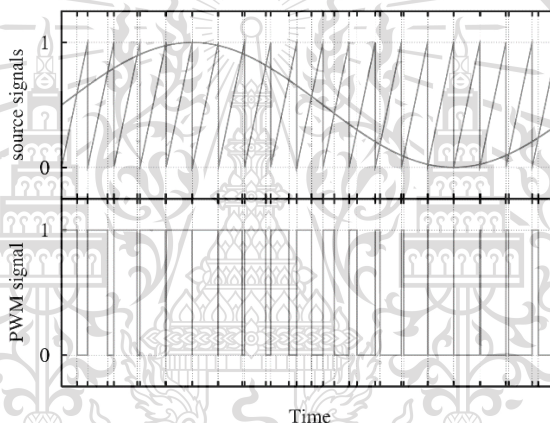
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

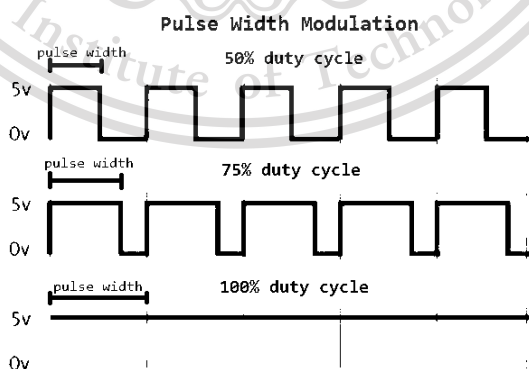
Forbidden to modify the content, and cite the document when use.

2.4 สัญญาณ PWM

สัญญาณ PWM (pulse width modulation) คือ สัญญาณที่เกิดจากการผสมกันระหว่างสัญญาณรูปสามเหลี่ยม (Triangle wave) ที่มีความเป็นเชิงเส้น (linearity) สูง กับระดับแรงเคลื่อนสัญญาณไฟกระแสตรงหรือสัญญาณอื่นๆ ที่มีระดับไฟฟ้ากระแสตรงเป็นตัวรองรับ การผสมสัญญาณดังกล่าวทำให้เกิดสัญญาณพัลส์ที่มี 2 สถานะ คือ เปิด (on) กับปิด (off) เมื่อนำสัญญาณพัลส์ที่ได้ไปขับอุปกรณ์กำลัง เช่น หลอดไฟ เมื่อสัญญาณพัลส์มีสถานะเปิดหลอดไฟจะติด หากสัญญาณพัลส์มีสถานะปิดหลอดไฟจะดับ โดยสัญญาณจะมีความต่อเนื่องเช่นนี้ตลอดทำให้หลอดไฟติดดับอย่างต่อเนื่อง และหากการติดดับมีความถี่สูงมากๆ จนกระทั่งไม่สามารถสังเกตเห็นการติดดับด้วยสายตาได้ ทำให้เมื่อช่วงเวลาเปิดน้อยลง แสงที่เห็นจากหลอดไฟจะหรี่ลง ในทางตรงกันข้ามหากช่วงเวลาเปิดมากขึ้น ทำให้หลอดไฟสว่างขึ้น จนกระทั่งสว่างเต็มที่เมื่อเปิดตลอด [27][28][30]



รูปที่ 2.8 การผสมกันระหว่างสัญญาณรูปสามเหลี่ยมกับสัญญาณไซน์ [4]



รูปที่ 2.9 ตัวอย่างสัญญาณ PWM ที่มี duty cycle ต่างกัน [4]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

รายละเอียดของสัญญาณ PWM ประกอบด้วยส่วนต่างๆ ดังนี้ คือ

1. ความถี่ คือ จำนวนรอบคลื่นต่อวินาที (cycle/second) ในการใช้งาน ความถี่จะต้องคงที่ไม่เปลี่ยนแปลง ความถี่ที่เหมาะสมในการใช้งานควบคุมอุปกรณ์กำลังต่างๆ คือตั้งแต่ 400 Hz ถึง 10 kHz ส่วนความถี่ที่ใช้เป็นพาหะของสัญญาณเสียงควรจะสูงกว่า 50 kHz ถึง 1 MHz
2. ความกว้างของสัญญาณ (pulse width) คือ ค่าคาบเวลาที่สัญญาณพัลส์เป็น 1 หรือช่วงเวลาที่อยู่ปรณทำงานใช้เพื่อบอกค่าของข้อมูลอนาล็อก
3. รอบทำงาน (Duty Cycle) คือ ค่าคาบเวลาช่วงเปิดของพัลส์เทียบกับช่วงเวลาที่ทั้งหมดของพัลส์ทั้งหมด ดังสมการที่ 2.1 [27][28][30]

$$\%Duty\ cycle = \frac{Time\ of\ Positive\ Pulse}{Time\ Period} \times 100\% \quad (2.1)$$

ค่ารอบทำงานมีผลทำให้ค่าเฉลี่ยแรงเคลื่อนเอาต์พุตที่ไปขับมอเตอร์เปลี่ยนแปลง ทำให้ความเร็วของมอเตอร์เปลี่ยนแปลงตามไปด้วย สามารถคำนวณได้ดังสมการที่ 2.2 [27][28][30]

$$V_{output} = \%Duty\ cycle \times V_{supply} \quad (2.2)$$

2.5 การสื่อสารข้อมูลแบบอนุกรม (Serial Communication)

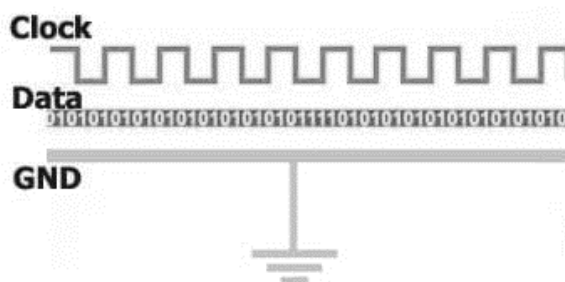
การสื่อสารข้อมูลแบบอนุกรม คือ การสื่อสารข้อมูลโดยการใช้สายสัญญาณเส้นเดียวในการรับ-ส่งข้อมูลระหว่างอุปกรณ์ตัวรับ และตัวส่งครั้งละ 1 บิตเป็นลำดับต่อเนื่องตั้งแต่บิตเริ่มต้นจนถึงบิตสุดท้าย โดยอาศัยเทคนิคต่างๆ เช่น การใช้สัญญาณทริกเพื่อรับข้อมูลเข้า การใช้บิตเริ่มต้นกำหนดการรับข้อมูล โดยอาจจะใช้ และไม่ใช้เวลาในการกำหนดการรับสัญญาณ ทั้งนี้การสื่อสารแบบอนุกรมสามารถแบ่งได้เป็น 2 รูปแบบ ได้แก่ การสื่อสารข้อมูลแบบซิงโครนัส และการสื่อสารข้อมูลแบบอะซิงโครนัส

2.5.1 การสื่อสารข้อมูลแบบซิงโครนัส (Synchronous)

การสื่อสารข้อมูลแบบอะซิงโครนัสจะใช้สัญญาณนาฬิกา ร่วมในการรับส่งข้อมูล ข้อดีของการสื่อสารแบบซิงโครนัส คือ การรับส่งข้อมูลมีความผิดพลาดน้อยหรือไม่มีความผิดพลาดเลย แต่ต้องใช้สายสัญญาณอย่างน้อย 3 เส้นในการสื่อสาร คือ ข้อมูล, สัญญาณนาฬิกา และกราวด์ โปรโตคอลที่ไม่ว่ากรณีใด ทั้งสิ้น อีพินซิงโครนัส ได้แก่ I²C, I²S และ SPI และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

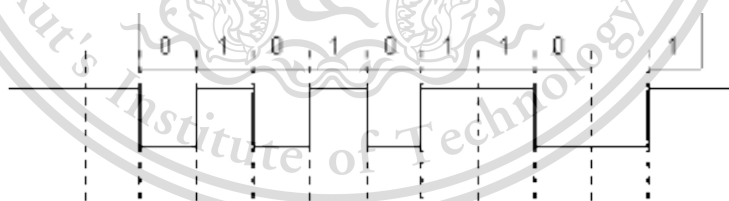
Forbidden to modify the content, and cite the document when use.



รูปที่ 2.10 การส่งข้อมูลแบบซิงโครนัส [5]

2.5.2 การสื่อสารข้อมูลแบบอะซิงโครนัส (Asynchronous)

การสื่อสารข้อมูลแบบอะซิงโครนัสเป็นการสื่อสารที่ไม่ต้องใช้สัญญาณนาฬิกาในการควบคุมจังหวะ ใช้สายสัญญาณข้อมูลเพียงเส้นเดียวในการทำงาน โดยอาศัยสัญญาณจากบิตเริ่มต้น และบิตสิ้นสุดในการบอกจังหวะการรับส่งข้อมูล การสื่อสารแบบนี้จำเป็นต้องกำหนดอัตราเร็วในการรับ และส่งข้อมูลให้เท่ากัน เรียกว่าอัตราเร็วนี้ว่า อัตราบอด (baud rate) มีหน่วยเป็น จำนวนบิตต่อวินาที (bit / second) ซึ่งหากกำหนดค่าไม่ถูกต้องจะทำให้อ่านข้อมูลที่ส่งมาได้ผิดพลาด การสื่อสารแบบอะซิงโครนัสมีข้อดี คือ ใช้สายสัญญาณเพียงเส้นเดียวในการรับ-ส่งข้อมูล แต่จะเกิดความผิดพลาดในการสื่อสารได้ง่าย โพรโตคอลที่ทำงานแบบอะซิงโครนัส คือ UART ลักษณะของสัญญาณที่ใช้ในการติดต่อสื่อสารประกอบด้วย บิตเริ่มต้น (start bit), บิตของข้อมูลที่สื่อสาร (transmission data) จำนวน 8 บิต, บิตตรวจสอบ (parity bit) และบิตสิ้นสุด (stop bit) โดยที่จะใช้บิตตรวจสอบหรือไม่ใช้ก็ได้ [5]



รูปที่ 2.11 การสื่อสารแบบอะซิงโครนัสที่ไม่ได้ใช้พาริตีบิต [5]



รูปที่ 2.12 การสื่อสารแบบอะซิงโครนัสที่ใช้พาริตีบิต [5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาระดับชั้นมัธยมศึกษาตอนต้นไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

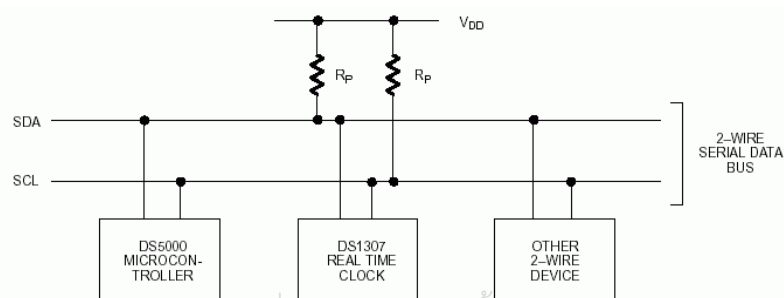
Forbidden to modify the content, and cite the document when use.

ขณะที่ไม่มีข้อมูลส่งออกมาจะมีระดับของสัญญาณเป็น 1 ตลอดเวลา เพื่อสื่อสารว่าอุปกรณ์ปลายทางยังคงติดต่อกับอุปกรณ์ต้นทางอยู่ เมื่อเริ่มส่งข้อมูลสัญญาณของอะชิงโคนัสจะเป็น 0 หนึ่งช่วงสัญญาณนาฬิกา เรียกบิตนี้ว่าบิตเริ่มต้น หลังจากบิตเริ่มต้นจะเป็นบิตข้อมูลที่ต้องการส่ง หลังจากบิตข้อมูลจะเป็นบิตตรวจสอบ แล้วจบด้วยบิตสิ้นสุด โดยบิตตรวจสอบจะใช้หรือไม่ใช้ก็ได้ หลังจากนั้นหากไม่มีข้อมูลส่งออกมาสัญญาณจะกลับไปเป็น 1 เพื่อรอการส่งข้อมูลต่อไป การสื่อสารแบบอะชิงโคนัสที่สัญญาณมีบางส่วนเป็นบิตเริ่มต้น บิตสิ้นสุด และบิตตรวจสอบจะทำให้ความเร็วในการส่งข้อมูลช้าลง เนื่องจากสูญเสียช่องทางการสื่อสารให้กับ บิตเริ่มต้น บิตสิ้นสุด และบิตตรวจสอบ (หากใช้) การสื่อสารแบบอะชิงโคนัสนี้มักใช้ในการติดต่อระหว่างคอมพิวเตอร์กับอุปกรณ์รอบข้าง [5]

2.6 I²C (Inter Integrate Circuit)

I²C ย่อมาจาก Inter Integrated Circuit เป็นการสื่อสารแบบอนุกรมรูปแบบหนึ่ง ใช้ติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก ถูกพัฒนาขึ้นโดยบริษัท Philips Semiconductors มีจุดประสงค์เพื่อใช้รับส่งข้อมูลความเร็วต่ำระหว่างอุปกรณ์ การเชื่อมต่อแบบบัสสามารถเชื่อมต่ออุปกรณ์ได้เป็นจำนวนมากด้วยสายสัญญาณเพียง 2 เส้น ทำให้ลดปริมาณการใช้สายสัญญาณ โดยสายสัญญาณเพียง 2 เส้นนั้น ได้แก่

1. serial data (SDA) มีหน้าที่รับส่งข้อมูลที่ต้องการจะสื่อสาร และเนื่องจากการรับส่งข้อมูลของสายสัญญาณทั้งสองเป็นแบบ Open-Drain จึงจำเป็นต้องต่อ Pull-Up Resistor ที่สายสัญญาณทั้งสอง และต้องใช้ไฟเลี้ยงด้วย
2. serial clock (SCL) มีหน้าที่ส่งสัญญาณนาฬิกาเพื่อใช้สำหรับควบคุมการรับส่งข้อมูล ซึ่งมีความเร็วตามมาตรฐาน คือ 100 kHz, Fast Mode มีความเร็วสูงสุด 400 kHz, Hi-Speed Mode มีความเร็วสูงสุด 3.4 MHz และ Ultra-Fast Mode มีความเร็วสูงสุดที่ 5 MHz โดยไมโครคอนโทรลเลอร์ แต่ละตัวก็จะมีความเร็วของสัญญาณนาฬิกาต่างกัน เช่น บอร์ด STM32 รองรับความเร็ว Standard และ Fast Mode เท่านั้น [32][33][34]



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 2.13 ลักษณะการเชื่อมต่อแบบ I²C [6]

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

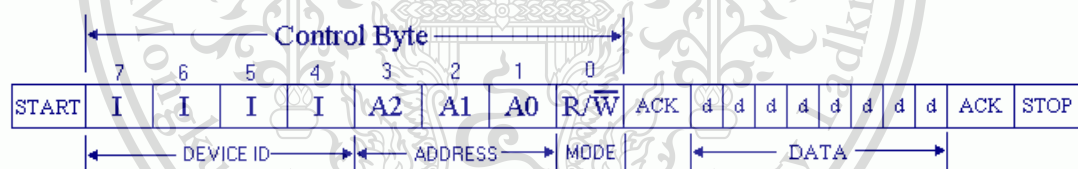
This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

การสื่อสารแบบ I²C ประกอบไปด้วย 2 ส่วน คือ อุปกรณ์หลัก (Master Device) และ อุปกรณ์ย่อย (Slave Device) ในระบบการสื่อสารแบบอนุกรม I²C สามารถต่ออุปกรณ์ I²C Device ได้หลายอุปกรณ์ การสื่อสารกันในระบบ Master Device จะอ้างอิง Address ของ Slave Device เพื่อระบุว่าต้องการสื่อสารกับ Slave Device ตัวใด สายสัญญาณทั้ง 2 เส้นต้องต่อตัวต้านทานแบบ Pull-up หากบอร์ดที่ใช้งานไม่สามารถรับอินพุตแบบ Open Drain หรือ Open Collector ของ อุปกรณ์ I²C Device ได้โดยตรง จะต้องเปลี่ยนสัญญาณทางไฟฟ้าให้กลายเป็นแบบพอร์ตที่ตัวบอร์ดสามารถรับได้ แต่โดยส่วนมากอุปกรณ์ I²C Device จะมีตัวต้านทานแบบ Pull-up มาจากโรงงานเรียบร้อยแล้ว [32][33][34]

2.6.1 การรับ-ส่งข้อมูลกับอุปกรณ์แบบ I²C

การรับ-ส่งข้อมูลแบบ I²C เริ่มต้นการส่งข้อมูลด้วยการส่งสถานะเริ่มต้น (start conditions) เพื่อแสดงการขอใช้บัส ตามด้วยรหัสควบคุม (control byte) ซึ่งประกอบด้วยรหัสประจำตัวอุปกรณ์ (Device ID), Device Address และ Mode ในการเขียนหรืออ่านข้อมูล เมื่ออุปกรณ์ที่ต้องการติดต่อทราบว่าไมโครคอนโทรลเลอร์ต้องการติดต่อด้วยจะส่งสถานะรับรู้ (acknowledge) หรือแจ้งให้ไมโครคอนโทรลเลอร์รับรู้ว่าคุณสมบัติที่ส่งมามีความถูกต้อง เมื่อสิ้นสุดการส่งข้อมูลไมโครคอนโทรลเลอร์จะต้องส่งสถานะสิ้นสุด (stop conditions) เพื่อสื่อสารกับอุปกรณ์ว่าสิ้นสุดการใช้บัส หากบัสไม่ได้ถูกใช้งานทั้ง SCL และ SDA จะมีสัญญาณเป็น 1 ทั้งคู่หรือสถานะบัสว่าง [32][33][34]



รูปที่ 2.14 รูปแบบการเขียน/อ่านข้อมูลแบบ I²C [6]

2.6.2 การกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I²C

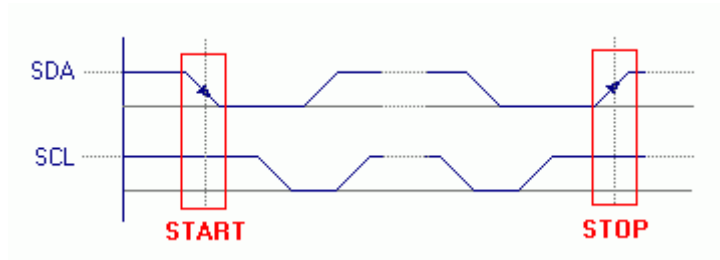
ลักษณะการกำหนดสถานะเริ่มต้น และสถานะสิ้นสุดของบัสเมื่อต้องการส่งข้อมูล ไมโครคอนโทรลเลอร์จะต้องส่งสถานะเริ่มต้น (start conditions) โดยการเปลี่ยน SDA จาก 1 เป็น 0 ในขณะที่ SCL มีค่าเป็น 1 เมื่อสิ้นสุดการใช้บัสไมโครคอนโทรลเลอร์จะต้องส่งสถานะสิ้นสุด (stop conditions) โดยการเปลี่ยน SDA จาก 0 เป็น 1 ในขณะที่ SCL มีค่าเป็น 1 เหมือนเดิม หากบัสไม่ได้ถูกใช้งานทั้ง SCL และ SDA จะเป็น 1 ทั้งคู่ หรือ สถานะบัสว่าง [32][33][34]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



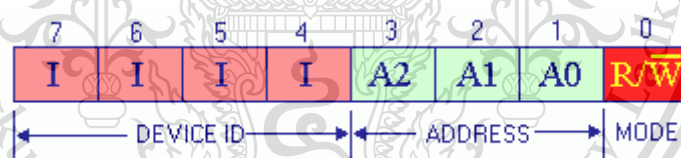
รูปที่ 2.15 การกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I²C บัส [6]

2.6.3 รหัสควบคุมของ I²C (Control Byte)

รหัสควบคุมของบัสประกอบด้วย รหัสประจำตัวของอุปกรณ์ (Device ID) และบิตควบคุมการเขียนอ่าน

1. รหัสประจำตัวของอุปกรณ์ ประกอบด้วยรหัสประจำตัวจากผู้ผลิต Product ID 4 บิต (บิต 4-7) ที่เปลี่ยนแปลงแก้ไขไม่ได้ และ Device Address 3 บิต (บิต 1-3) ซึ่งผู้ใช้สามารถกำหนดเองได้ รวมแล้วเป็นรหัส 7 บิต ใช้ระบุตัวอุปกรณ์ ที่ต่ออยู่บนบัสจะมีค่าซ้ำกันไม่ได้

2. บิตควบคุมการเขียนอ่าน (Mode) บิต 0 เมื่อไมโครคอนโทรลเลอร์ต้องการเขียนข้อมูลไปยังอุปกรณ์ก็กำหนดให้บิตนี้เป็น 0 และเมื่อต้องการอ่านข้อมูลจากอุปกรณ์จะกำหนดให้บิตนี้เป็น 1 [32][33][34]



รูปที่ 2.16 I²C บัส Control Byte [6]

2.7 แบตเตอรี่ลิเทียมโพลิเมอร์

แบตเตอรี่ลิเทียมโพลิเมอร์ในการออกแบบครั้งแรกได้รวมอิเล็กโทรไลต์-โพลิเมอร์ในรูปแบบแข็ง และแห้งคล้ายกับฟิล์มพลาสติกทำให้ผลออกมามีรูปร่างจะคล้ายๆกับบัตรเครดิต (มีลักษณะบาง) แต่ยังคงมีความสามารถในการคงอายุการใช้งานที่ดี นอกจากนี้แบตเตอรี่ประเภทนี้มีความเบา และถูกปรับปรุงให้มีความปลอดภัยสูงขึ้น แบตเตอรี่ประเภทนี้มีราคาสูงกว่าแบตเตอรี่ลิเทียมไอออนแต่มีความหนาแน่นของแบตเตอรี่น้อยกว่าแบตเตอรี่ลิเทียมไอออน ปัจจุบันแบตเตอรี่ลิเทียมโพลิเมอร์เป็น

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อใช้ในการศึกษา การวิจัย และการพัฒนา การค้า ไม่ว่ากรณีใด ๆ ที่มีการนำเนื้อหา หรือข้อมูลใด ๆ ไปใช้โดยไม่ได้รับอนุญาตจากสถาบันฯ

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2.7.1 แบตเตอรี่ลิเทียมโพลิเมอร์

การต่อแบตเตอรี่มี 2 ลักษณะ คือ แบบอนุกรม และแบบขนาน

1. แบบอนุกรม (Series) การต่อแบบอนุกรมเป็นการเพิ่มแรงดันไฟฟ้าของแบตเตอรี่ลิเทียมโพลิเมอร์ แบตเตอรี่ลิเทียมโพลิเมอร์ 1 เซลล์ มีแรงดันไฟฟ้า 3.7 โวลต์ ถ้านำมาต่ออนุกรมกัน 2 ก้อน จะได้แบตเตอรี่ลิเทียมโพลิเมอร์ 2 เซลล์ แรงดันไฟฟ้า 7.4 โวลต์ ถ้านำแบตเตอรี่ต่อกันมากจำนวน แรงดันไฟฟ้าจะมากขึ้น

2. แบบขนาน (Parallel) การต่อแบบขนานเป็นการเพิ่มค่า C ของแบตเตอรี่ทั้งระบบ เมื่อนำแบตเตอรี่ขนาด 500 มิลลิแอมป์ชั่วโมง 10C จำนวน 2 ก้อน มาต่อขนานจะได้แรงดันไฟฟ้าเท่ากับ 1 ก้อน (3.7 โวลต์) เท่าเดิม แต่จะได้กระแสเพิ่มเป็น 1,000 มิลลิแอมป์ชั่วโมง และ C เพิ่มขึ้นเป็น 20C ความจุของแบตเตอรี่ [35]

2.7.2 ความจุแบตเตอรี่

ความจุมีหน่วยเป็นมิลลิแอมป์ชั่วโมง (mAh) คือปริมาณกระแสไฟฟ้าที่จ่ายได้ภายในเวลา 1 ชั่วโมง ค่าความจุแบตเตอรี่ยิ่งมากจะทำให้สามารถใช้งานได้เป็นเวลานาน แต่ขนาดของแบตเตอรี่จะมีขนาดใหญ่ขึ้นด้วย จึงต้องคำนวณกระแสเทียบกับเวลาที่ต้องการใช้งานก็จะได้ปริมาณที่เหมาะสม

2.7.3 ความสามารถในการจ่ายกระแส

แบตเตอรี่จะแสดงค่าตัวเลข 40C 50C 60C เป็นค่าที่แสดงความสามารถในการจ่ายกระแสเป็นจำนวนเท่าของความจุ (C) หมายถึงปริมาณกระแสที่แบตเตอรี่จ่ายได้สูงสุดในช่วงเวลาหนึ่ง เช่น แบตเตอรี่ลิเทียมโพลิเมอร์ 1000 มิลลิแอมป์ชั่วโมง มีค่า 20C จะสามารถจ่ายกระแสได้ชั่วขณะถึง 20,000 มิลลิแอมป์ หรือ 20 แอมป์ [35]

2.7.4 การเก็บรักษาแบตเตอรี่ (Storage)

การเก็บรักษาแบตเตอรี่ เป็นการลดประจุของแบตเตอรี่ให้อยู่ในระดับที่เหมาะสมสำหรับการเก็บรักษาหรือในกรณีที่ไม่ได้ใช้แบตเตอรี่เป็นเวลานาน วิธีการเก็บรักษาแบตเตอรี่มี 2 วิธี ดังนี้

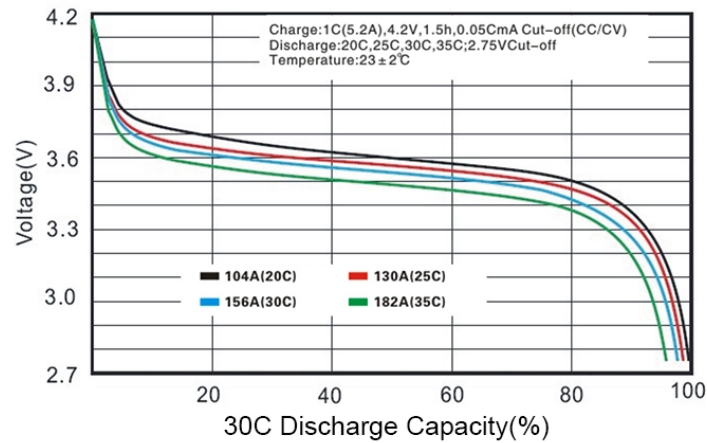
1. นำเข้าเครื่องชาร์จไปที่โหมด storage charge เมื่อการเก็บรักษาแบตเตอรี่เสร็จสิ้นแต่ละเซลล์จะมีแรงดันไฟฟ้าประมาณ 3.85 V

2. ใช้วิธีการจ่ายกระแสจนกระทั่งแบตเตอรี่เหลือประมาณ 3.85 V แล้วจึงนำแบตเตอรี่ไป

เอกสารนี้เป็นเอกสารที่เผยแพร่ในที่นี้เพื่อประโยชน์ใช้สอยเท่านั้น สถานการณ์ที่เก็บรักษาแบตเตอรี่ควรจะเป็นที่ร่มและเย็น การคำนวณค่าไม่ว่ากรณีใดไม่ควรนำไปเก็บไว้ในรถเด็ดขาด [35] ข้อควรระวัง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



รูปที่ 2.17 Discharge Voltage Curve ของแบตเตอรี่ลิเทียมโพลีเมอร์ [7]

2.8 ระบบระบุตำแหน่งบนพื้นโลก

ระบบระบุตำแหน่งบนโลก GPS คือ ระบบดาวเทียมนำร่องโลก GNSS (Global Navigation Satellite System) เพื่อระบุข้อมูลของตำแหน่ง และเวลา โดยระบุจากการคำนวณความถี่ของสัญญาณนาฬิกาที่ส่งมาจากดาวเทียมที่โคจรรอบโลกทำให้สามารถระบุตำแหน่ง ณ จุดที่สามารถรับสัญญาณได้ และสามารถคำนวณความเร็ว และทิศทางเพื่อนำมาใช้ในการนำทางได้ [41][42][43]

2.8.1 หลักการทำงานของ GPS

ในการระบุตำแหน่งของ GPS ตัวรับสัญญาณจะรับสัญญาณจากดาวเทียมที่โคจรรอบโลกอย่างน้อย 3 ตัวขึ้นไป เพื่อคำนวณหาระยะทางระหว่างตัวรับสัญญาณกับดาวเทียม และระบุตำแหน่งของดาวเทียม เมื่อสามารถระบุตำแหน่งของดาวเทียมตัวรับสัญญาณจะคำนวณหาจุดตัดของรัศมีดาวเทียมเพื่อระบุตำแหน่งของตัวรับสัญญาณความแม่นยำของการระบุตำแหน่งขึ้นอยู่กับหลายปัจจัย

1. ระยะห่างของดาวเทียมแต่ละดวง หากดาวเทียมอยู่ห่างกันจะมีความแม่นยำในการระบุตำแหน่งมากกว่า
2. จำนวนดาวเทียมหากมีจำนวนที่รับสัญญาณได้มากก็จะส่งผลให้มีความแม่นยำมากขึ้น
3. ความแปรปรวนของชั้นบรรยากาศ เนื่องจากชั้นบรรยากาศมีประจุไฟฟ้าความชื้นอุณหภูมิ และความหนาแน่นที่แปรปรวนตลอดเวลา ทำให้สัญญาณที่รับส่งได้อ่อนลง
4. สิ่งแวดล้อมในบริเวณรับสัญญาณ เช่น มีการบดบังจากกระจก ละอองน้ำ ไบโม่ จะมีผลต่อค่าความถูกต้องของความแม่นยำ เนื่องจากถ้าสัญญาณจากดาวเทียมมีการหักเหก็จะทำให้ ค่าที่คำนวณได้จากเครื่องรับสัญญาณผิดได้ [41][42][43]

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษายเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2.8.2 GPS Protocol

NMEA (Nation Maritime Electronics Association) เป็นสมาคมที่ศึกษา และพัฒนา อุปกรณ์อิเล็กทรอนิกส์ ได้พัฒนามาตรฐานในการสื่อสารข้อมูลระหว่างอุปกรณ์เรียกว่า NMEA Standard ซึ่งระบุข้อมูลการเชื่อมต่อทางไฟฟ้า และรูปแบบของข้อมูล โพรโตคอลที่ใช้งานส่วนใหญ่ [47] เช่น

1. GGA

ตัวอย่าง: \$GPGGA,002153.000,3342.6618,N,11751.3858,W,1,10,1.2,27.0,M,-34.2,M,,0000*5E [44]

ตารางที่ 2.1 ตัวอย่าง GGA

| ชื่อ | ตัวอย่างข้อมูล | รายละเอียด |
|--|------------------|---|
| Sentence Identifier | \$GPGGA | Global Positioning System Fix Data |
| Time | 002153.000 | ข้อมูล Fix เมื่อเวลา 0:21:53 UTC |
| Latitude | 3342.6618,N | ค่าพิกัดLatitudeด้านเหนือเส้นศูนย์สูตร 33 deg 42.6618 N |
| Longitude | 11751.3858, W | ค่าพิกัดLongitudeด้านตะวันออก 117 deg 51.3858 W |
| Fix Quality: 0 = Invalid 1 = GPS fix 2 = DGPS fix | 1 | โหมด GPS fix |
| Number of Satellites | 10 | จำนวนดาวเทียม GPS ที่รับสัญญาณได้ |
| Horizontal Dilution of Precision (HDOP) | 1.2 | ค่าความคลาดเคลื่อนการระบุตำแหน่งแนวราบ |
| Altitude | 27.0,M | ค่าความสูงเหนือระดับน้ำทะเลปานกลาง |
| Height of geoid above WGS84 ellipsoid | -34.2,M | ความสูงของจีออยด์เหนือทรงรี WGS84 |
| Time since last DGPS update | ว่าง | ไม่มีค่าอัปเดต |
| DGPS reference station id | 0000 | แสดงหมายเลขสถานีของ DGPS |
| Checksum | *5E | ค่า checksum นำหน้าด้วย * |

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเท่านั้น ไม่อนุญาตให้เผยแพร่ข้อมูลใดๆ การค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2. GLL

ตัวอย่าง: \$GPGLL,3723.2475,N,12158.3416,W,161229.487,A,A*41 [45]

ตารางที่ 2.2 ตัวอย่าง GLL

| ชื่อ | ตัวอย่างข้อมูล | รายละเอียด |
|---|----------------|--|
| Sentence Identifier | \$GPGLL | Geographic Position, Latitude / Longitude and time. |
| Latitude | 3723.2475,N | ค่าค่าพิกัดLatitudeด้านเหนือเส้นศูนย์สูตร 37 deg 23.2475 N |
| Longitude | 12158.3416,W | ค่าพิกัดLongitudeด้านตะวันออก 121 deg 58.3416 W |
| Time | 161229.487 | ข้อมูล Fix เมื่อเวลา 16:12:29 UTC |
| Status A – Active V – void | 3 | สถานะ Active |
| Mode A – Autonomous mode D – Differential mode E – Estimated mode M – Manual input mode S – Simulator mode N – Data not valid | A | Autonomous mode |
| Checksum | *41 | ค่า checksum นำหน้าด้วย * |

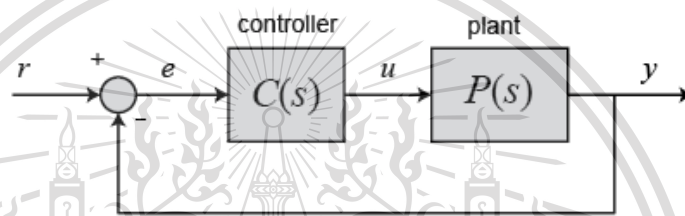
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2.9 PID Controller

การควบคุมตามสัดส่วน ปริพันธ์ อนุพันธ์ (Proportional Integral Derivative) เป็นระบบควบคุมที่ใช้กันมากที่สุดในอุตสาหกรรม และได้รับการยอมรับในระดับสากล เนื่องจาก ประสิทธิภาพที่สูง และความเรียบง่ายในการใช้งาน ทำให้ผู้ใช้สามารถประยุกต์ใช้งานได้หลากหลาย PID Controller ประกอบด้วยพารามิเตอร์สามตัว คือ ตัวควบคุมแบบสัดส่วน (Proportional) ตัวควบคุมปริพันธ์ (Integral) และตัวควบคุมอนุพันธ์ (Derivative) เป็นระบบควบคุมแบบป้อนกลับ โดยอาศัยหลักการจากการนำเอาต์พุตไปป้อนกลับเพื่อหาค่าความผิดพลาดระหว่างเอาต์พุตกับค่าที่ต้องการ และใช้การปรับพารามิเตอร์ เพื่อเปลี่ยนค่าเอาต์พุตให้มีค่าความผิดพลาดให้น้อยที่สุด เป็นไปตามระบบป้อนกลับ ดังรูปที่ 2.18 [37][38][39][40]



รูปที่ 2.18 Block Diagram ของ plant ที่ใช้ตัวควบคุมร่วมด้วย [8]

โดยเอาต์พุตจากตัวควบคุมจะเป็นไปตามสมการที่ 2.3

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_p \frac{de}{dt} \quad (2.3)$$

ตัวแปร e แสดงถึงค่าความผิดพลาดจากความแตกต่างระหว่างเอาต์พุตที่ต้องการและเอาต์พุตจริง ค่าความผิดพลาดนี้ถูกป้อนไปยังตัวควบคุม PID ตัวควบคุมจะคำนวณทั้งสัดส่วน อนุพันธ์ และอินทิกรัลของค่าความผิดพลาดนี้ตามเวลา t สัญญาณควบคุมที่ส่งไปยังแพลนตจะเท่ากับ เกนของสัดส่วนคูณด้วยขนาดของค่าความผิดพลาดบวกกับเกนของอินทิกรัลคูณอินทิกรัลของค่าความผิดพลาดบวกกับเกนของอนุพันธ์คูณด้วยอนุพันธ์ของค่าความผิดพลาดสัญญาณควบคุมที่ถูกป้อนไปยังแพลนต และจะได้รับเอาต์พุตใหม่ จากนั้นเอาต์พุตใหม่จะถูกป้อนกลับ และเปรียบเทียบกับค่าที่ต้องการ (set point) เพื่อหาค่าความผิดพลาดใหม่ กระบวนการนี้ดำเนินต่อไปเรื่อยๆ เป็นลูป เพื่อให้ได้ระบบที่มีประสิทธิภาพตามที่ผู้ใช้ต้องการมากที่สุด ค่าความผิดพลาด ความเร็วของการตอบสนอง และความเสถียรของระบบสามารถเปลี่ยนจากการปรับค่าพารามิเตอร์ P I และ D [37][38][39][40]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2.9.1 ตัวควบคุมแบบสัดส่วน (Proportional: P)

ตัวควบคุมแบบสัดส่วน หรือ P-controller ให้เอาต์พุตที่เป็นสัดส่วนกับค่าความผิดพลาดปัจจุบัน $e(t)$ เปรียบเทียบจุดที่ต้องการกับค่าจริง ค่าความผิดพลาดที่เกิดขึ้นจะคูณด้วยค่าคงที่ตามสัดส่วนเพื่อให้ได้ผลลัพธ์ไปควบคุมระบบ ดังสมการที่ 2.4 [37][38][39][40]

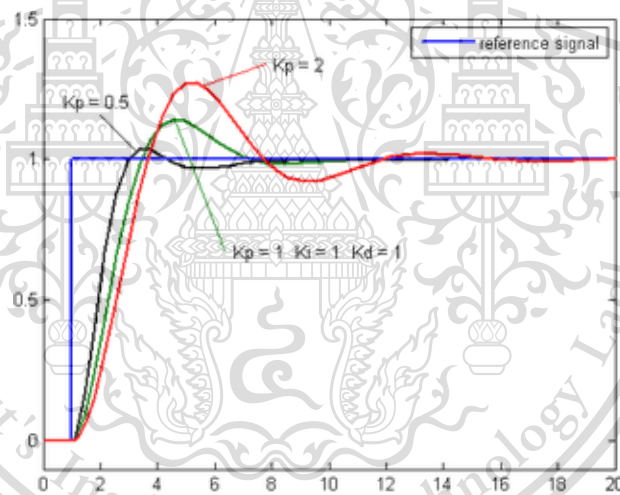
$$P_{out} = K_p e(t) \quad (2.4)$$

เมื่อ P_{out} = เอาต์พุต

K_p = อัตราขยายสัดส่วน

e = ค่าความผิดพลาด

t = เวลา



รูปที่ 2.19 ผลตอบสนองเมื่อมีการปรับค่า K_p [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2.9.2 ตัวควบคุมปริพันธ์ (Integral: I)

ตัวควบคุมปริพันธ์ หรือ I-controller มี offset แบบต่อเนื่อง จึงมีการเรียกใช้ I ปริพันธ์ (Integral) ที่เป็นสัดส่วนของขนาดความผิดพลาด และระยะเวลาของความผิดพลาด ผลรวมของความผิดพลาดในทุกช่วงเวลา (ปริพันธ์ของความผิดพลาด) จะให้ค่า offset สะสมที่ควรจะเป็นในก่อนหน้า เป็นค่าความผิดพลาดใหม่ ความผิดพลาดสะสมจะถูกคูณโดยอัตราขยายปริพันธ์ ดังสมการที่ 2.5 [37][38][39][40]

$$I_{out} = K_i \int_0^t e(\tau) \quad (2.5)$$

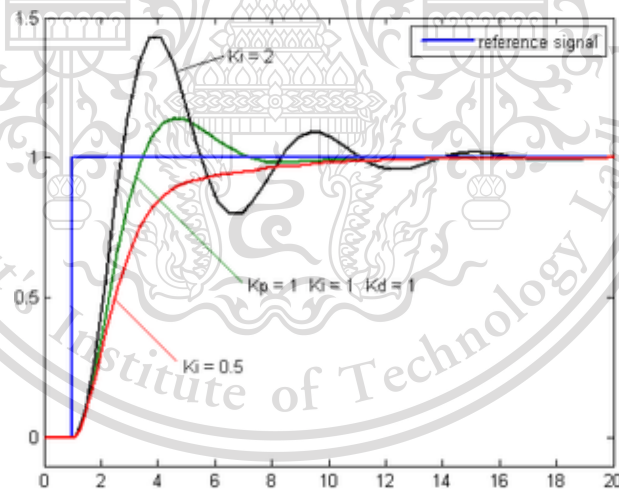
เมื่อ I_{out} = เอาต์พุต

K_i = อัตราขยายปริพันธ์

e = ค่าความผิดพลาด

t = เวลา

τ = ตัวแปรปริพันธ์



รูปที่ 2.20 ผลตอบสนองเมื่อมีการปรับค่า K_i [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

2.9.3 ตัวควบคุมอนุพันธ์ (Derivative: D)

ตัวควบคุมอนุพันธ์ หรือ D-controller มีหน้าที่ชะลอการเปลี่ยนแปลงของเอาต์พุต หรือลดโอเวอร์ชูตของระบบโดยจะคำนวณความชันของค่าความผิดพลาดทุกช่วงเวลา และคูณด้วยอัตราขยายอนุพันธ์ ดังสมการที่ 2.6 [37][38][39][40]

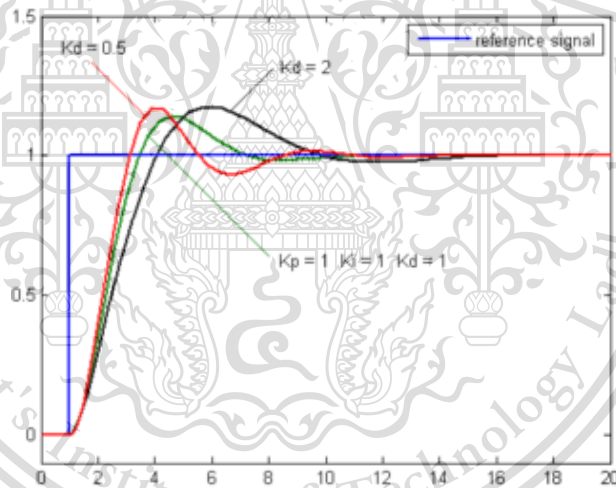
$$D_{out} = K_d \frac{d}{dt} e(t) \quad (2.6)$$

เมื่อ D_{out} = เอาต์พุต

K_d = อัตราขยายอนุพันธ์

e = ค่าความผิดพลาด

t = เวลา



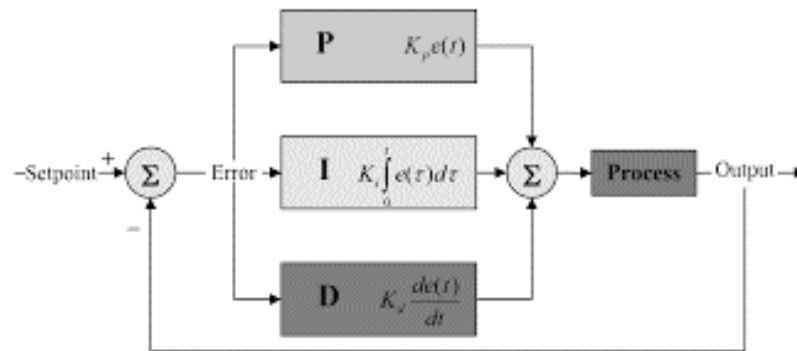
รูปที่ 2.21 ผลตอบสนองเมื่อมีการปรับค่า K_d [9]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

เมื่อนำพารามิเตอร์ทั้ง 3 ตัวมารวมกันจะได้ดังรูปที่ 2.22



รูปที่ 2.22 ลักษณะการทำงานของ PID Controller [10]

โดยผลจากการปรับเพิ่มค่าตัวแปรพารามิเตอร์ทั้ง 3 ตัว สามารถสรุปได้ดังตารางที่ 2.3

ตารางที่ 2.3 ผลการปรับค่าตัวแปรพารามิเตอร์ทั้ง 3 ตัว [37][38][39][40]

| ตัวแปร | Rise time | Overshoot | Settling time | Steady-state error | เสถียรภาพ |
|--------|-----------|-----------|---------------------|--------------------|-----------------------|
| K_p | ลดลง | เพิ่มขึ้น | เปลี่ยนแปลงเล็กน้อย | ลด | ลด |
| K_i | ลดลง | เพิ่มขึ้น | เพิ่มขึ้น | ลด | ลด |
| K_d | ลดลง | ลดลง | ลดลง | ในทางทฤษฎีไม่ส่งผล | ดีขึ้นหากไม่มากเกินไป |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

บทที่ 3

วิธีการดำเนินงาน

3.1 ขั้นตอนการดำเนินงาน

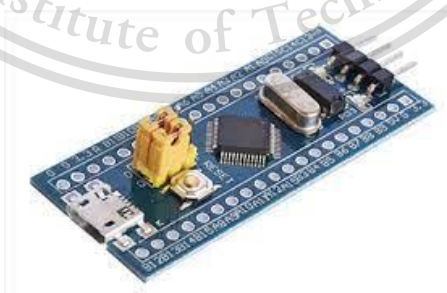
ขั้นตอนของการดำเนินงานได้วางแผนไว้ดังนี้

1. ศึกษาค้นคว้าทฤษฎี และเอกสารที่เกี่ยวข้อง
2. ศึกษาหลักการของเซ็นเซอร์ที่จะนำมาใช้
3. ออกแบบโครงสร้าง และการทำงานของโครง
4. จัดซื้อเซ็นเซอร์ และอุปกรณ์ต่างๆ
5. ทดสอบเซ็นเซอร์แต่ละตัวที่นำมาใช้
6. ประกอบโครงเข้ากับเซ็นเซอร์ทั้งหมด
7. เขียนโปรแกรมควบคุมการเคลื่อนที่ของโครง
8. ออกแบบ และสร้างแอปพลิเคชันสำหรับส่งตำแหน่งของผู้ป่วย
9. ทดสอบการทำงานของโครง และแก้ไขข้อผิดพลาด
10. สรุปผล และจัดทำเอกสารรายงานเป็นรูปเล่ม

3.2 อุปกรณ์ที่เกี่ยวข้อง

3.2.1 บอร์ดพัฒนา STM32F103C8T6

บอร์ดพัฒนา STM32F103C8T6 เป็นตัวประมวลผล เป็นตัวควบคุม ESC ให้ไปสั่งการมอเตอร์และเป็นตัวกลางหลักที่รับหรือส่งข้อมูลของอุปกรณ์ ดังรูปที่ 3.1



รูปที่ 3.1 บอร์ดพัฒนา STM32F103C8T6 [11]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.2.2 มอเตอร์ Dxx C2826 1290 KV

มอเตอร์ Dxx C2826 เป็นมอเตอร์ Brushless Motor โดยมอเตอร์จะรับคำสั่งมาจาก ESC ให้หมุนที่ความเร็วที่กำหนด เพื่อให้ใบพัดหมุนด้วยอัตราเร็วพอที่จะยกตัวอุปกรณ์ขึ้นลงหรือเลี้ยวซ้ายขวา สาเหตุที่ใช้ 1290 Kv เพราะมีกำลังที่จะยกน้ำหนักของโดรนที่ออกแบบไว้ขึ้นได้ ดังรูปที่ 3.2



รูปที่ 3.2 Brushless Motor Dxx C2826 [12]

3.2.3 แบตเตอรี่ลิเทียมโพลิเมอร์ 3S 11.1V

แบตเตอรี่ทำหน้าที่จ่ายไฟให้กับอุปกรณ์ในตัวโดรนทั้งหมด สาเหตุที่ใช้แบตเตอรี่ลิเทียมโพลิเมอร์เพราะเสื่อมสภาพช้า ใช้ได้นานสามารถชาร์จใช้ต่อได้ทันที มีกำลังจ่ายที่เสถียร และที่สำคัญมีน้ำหนักที่เบาทำให้ประหยัดไฟจากแบตเตอรี่ที่จ่ายให้มอเตอร์ได้ลดลงเหมาะสำหรับโดรน ดังรูปที่ 3.3



รูปที่ 3.3 แบตเตอรี่ลิเทียมโพลิเมอร์ [13]

3.2.4 Frame F450 450mm

โครงโดรนรุ่น F450 ใช้สำหรับติดอุปกรณ์ต่างๆบนตัวโดรน เช่น STM32F103C8T6, แบตเตอรี่ ESC และ มอเตอร์ เป็นต้น ดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



รูปที่ 3.4 Frame F450 450mm [14]

3.2.5 Electronic Speed Controller (ESC)

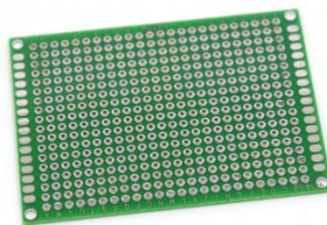
ESC ทำหน้าที่ควบคุมการทำงานของมอเตอร์โดยใช้สัญญาณ PWM และกำหนดการจ่ายไฟให้มอเตอร์ให้มีความเร็วตามกำหนด ดังรูปที่ 3.5



รูปที่ 3.5 Electronic Speed Controller [15]

3.2.6 บอร์ด PCB (Print Circuit Board)

ใช้เป็นที่สำหรับยึดชิ้นส่วน และเป็นทางเดินสัญญาณไฟฟ้าของอุปกรณ์อิเล็กทรอนิกส์ต่างๆ ที่อยู่บนวงจร และจะทำให้อุปกรณ์ต่างๆ เชื่อมต่อกัน และสามารถทำงานได้อย่างถูกต้องตามที่ได้ออกแบบไว้



รูปที่ 3.6 บอร์ด PCB [16]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.2.7 MC6C 2.4g 6 channel remote control RC Transmitter and Receiver

รีโมทวิทยุเครื่องบินควบคุมระยะไกล 800-1500 เมตร สามารถส่งสัญญาณออกมาควบคุมได้ 6 ช่อง เป็นสัญญาณ PWM ซึ่งใช้ในการควบคุมการเคลื่อนที่ และการเปลี่ยนโหมดของโดรน



รูปที่ 3.7 MC6C 2.4g 6 channel remote control RC Transmitter and Receiver [17]

3.2.8 Ublox NEO-M8N GPS Module

Ublox NEO-M8N เป็น GPS Module สามารถใช้งานระบบนำทาง 2 ระบบได้ในเวลาเดียวกันคือ GPS และ GNSS ใช้เพื่อระบุตำแหน่งที่โดรนอยู่ และใช้ในการคำนวณการเคลื่อนที่ในระบบอัตโนมัติ



รูปที่ 3.8 Ublox NEO-M8N [18]

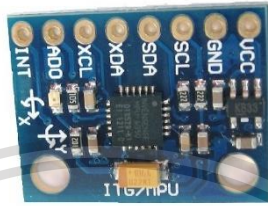
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.2.9 MPU6050 Module

ประกอบไปด้วย Accelerometer/Gyro Module เป็นโมดูลวัดความเร่งและความเร็วเชิงมุม วัดได้ 3 แกน โดย accelerometer จะวัดความเร่ง และ gyroscope จะวัดความเร็วเชิงมุม เมื่อทำการเอียงโมดูลค่าที่วัดได้จาก accelerometer และ gyroscope จะเปลี่ยนไป นำมาประยุกต์ใช้ในการควบคุมโดรนให้สามารถบินได้อย่างมีเสถียรภาพ โมดูลนี้ใช้ interface แบบ I²C ในการสื่อสาร



รูปที่ 3.9 MPU6050 Module [19]

3.2.10 MS5611 Module

เซ็นเซอร์วัดความดันบรรยากาศ ใช้วัดความสูงโดยใช้ค่าความดันบรรยากาศ และอุณหภูมิในการคำนวณหาความสูง ณ ตำแหน่งนั้นๆ โมดูลนี้ใช้ interface แบบ I²C ในการสื่อสาร



รูปที่ 3.10 MS5611 Module [20]

3.2.11 HMC5883L Module

เซ็นเซอร์เข็มทิศ สามารถตรวจจับทิศทางการกำลังหันหน้า รวมไปถึงความเอียง เซ็นเซอร์ตัวนี้สามารถจับทิศทางได้ทั้งหมด 3 แกน ใช้ interface แบบ I²C ในการสื่อสาร



รูปที่ 3.11 HMC5883L Module [21]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.2.12 FTDI FT232RL USB to TTL Serial Adapter

เป็นอุปกรณ์ที่ใช้ในการแปลงสัญญาณ USB ไปเป็น UART (RX,TX) โดยใช้ Chip ของ FTDI ใช้ในการเชื่อมต่อโน้ตบุ๊กกับบอร์ด STM32F103C8T6



รูปที่ 3.12 FTDI FT232RL USB to TTL Serial Adapter [22]

3.3 ซอฟต์แวร์ที่เกี่ยวข้อง

3.3.1 Arduino IDE

ใช้สำหรับเขียนโปรแกรมการทำงานต่างๆ ให้กับบอร์ดพัฒนา STM32F103C8T6
 ดังรูปที่ 3.13



รูปที่ 3.13 Arduino IDE

3.3.2 Flutter

เป็นภาษาใช้สำหรับเขียนโปรแกรม Application สำหรับการส่งค่าตำแหน่งไปยังอุปกรณ์รับ
 ดังรูปที่ 3.14



รูปที่ 3.14 Flutter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.3.3 Android Studio

เป็นโปรแกรมใช้สำหรับเขียนภาษาคอมพิวเตอร์ต่างๆ รองรับหลายฟังก์ชันในการเขียนของแต่ละภาษา มีแสดงผลในตัวเองนี้คือความสำคัญของแอปพลิเคชันนี้



รูปที่ 3.15 Android Studio

3.3.4 Firebase

เป็นเซิร์ฟเวอร์ฟรีบนเว็บไซต์ ใช้ในการเก็บข้อมูลหรือส่งข้อมูลได้



รูปที่ 3.16 Firebase

3.4 การออกแบบและการวางแผนการทำงาน

3.4.1 การออกแบบและการวางแผนทางด้านฮาร์ดแวร์

1. ออกแบบระบบการทำงานของโดรน
2. ศึกษาและเลือกไมโครคอนโทรลเลอร์สำหรับควบคุมการทำงาน
3. ศึกษาและเลือกเซ็นเซอร์ที่จะนำมาใช้
4. ทดสอบการทำงานของเซ็นเซอร์ทุกตัวที่นำมาใช้
5. ประกอบอุปกรณ์ และเซ็นเซอร์ต่างๆ ตามที่ออกแบบไว้
6. ทดสอบการทำงานของโดรน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

3.4.2 การออกแบบและการวางแผนทางด้านซอฟต์แวร์

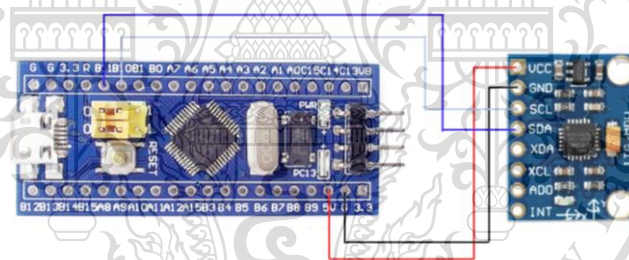
1. ศึกษาโปรแกรมที่จะนำมาเขียนให้กับบอร์ดไมโครคอนโทรลเลอร์ และการสร้างแอปพลิเคชัน
2. เขียนโปรแกรมบน Arduino IDE เพื่อรับค่าจากเซ็นเซอร์ และนำไปใช้คำนวณในระบบเคลื่อนที่ของโดรน
3. สร้างแอปพลิเคชันด้วยภาษา Flutter เพื่อแสดงผลบนโทรศัพท์มือถือ
4. ทดสอบโปรแกรม และแอปพลิเคชันกับชุดอุปกรณ์ที่ประกอบไว้

3.5 วิธีการดำเนินงาน

3.5.1 การทดสอบเซ็นเซอร์

การทดสอบเซ็นเซอร์โมดูลวัดความเร่ง 3 แกน (MPU6050)

ทดสอบเพื่อดูการเปลี่ยนแปลงของค่ามุมในแกน x, y และ z เมื่อโดรนเคลื่อนที่ค่ามุมจะเปลี่ยนแปลงไปจากแนวระดับ โดยลักษณะการต่อโมดูลเซ็นเซอร์เข้ากับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 จะแสดงดังรูปที่ 3.17



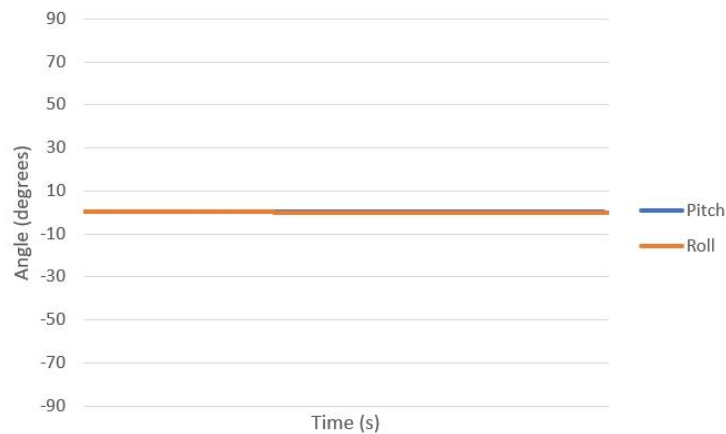
รูปที่ 3.17 การต่อโมดูล MPU6050 กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6

ในการทดสอบจะวางเซ็นเซอร์ไว้ในแนวระนาบกับพื้น ในสภาวะปกติที่ไม่มีการขยับเซ็นเซอร์ จะได้ค่ามุมทั้ง 3 แกน ดังรูปที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



รูปที่ 3.18 ค่ามุมที่ได้จากโมดูล MPU6050 ในแนวระนาบกับพื้น

เมื่อทดลองให้โมดูลขยับ โดยเมื่อขยับโมดูลในทิศของแกน x, y และ z ตามลำดับ ตำแหน่งของโมดูลที่เปลี่ยนไปทำให้ค่ามุมที่ได้เปลี่ยนไปตามทิศทางของแกน และเมื่อนำโมดูลวางไว้ในแนวระนาบกับพื้น จะเห็นว่าค่ามุมจะลดลงจนกระทั่งเป็นศูนย์หรือใกล้เคียง ดังรูปที่ 3.19



รูปที่ 3.19 ค่ามุมที่ได้จากโมดูล MPU6050 เมื่อขยับไปในทิศทางต่างๆ

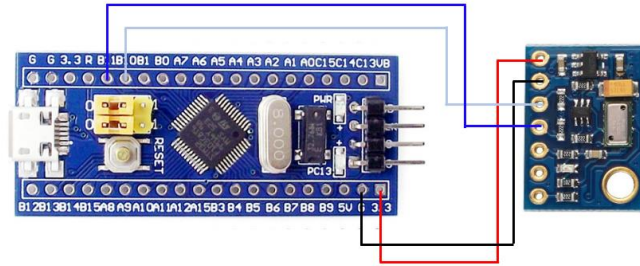
การทดสอบโมดูลเซ็นเซอร์ความสูงบรรยากาศความละเอียดสูง (MS5611)

ทดสอบเพื่อดูการเปลี่ยนแปลงของความสูงเมื่อความดันอากาศเปลี่ยนในพื้นที่ปิดเพื่อป้องกันแสงแดด และลมทำให้ค่าที่ได้แม่นยำมากขึ้น เมื่อตำแหน่งความสูงของโมดูลเปลี่ยนแปลง ความดันอากาศจะเปลี่ยนแปลง และโมดูลจะเก็บค่าเพื่อนำไปคำนวณต่อไป โดยลักษณะการต่อโมดูลเซ็นเซอร์เข้ากับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 จะแสดงดังรูปที่ 3.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

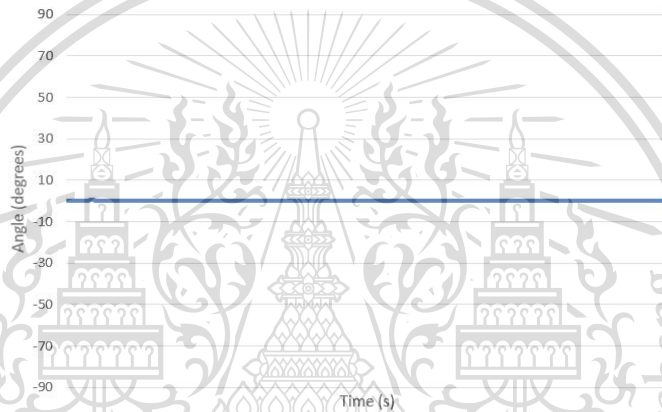
This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



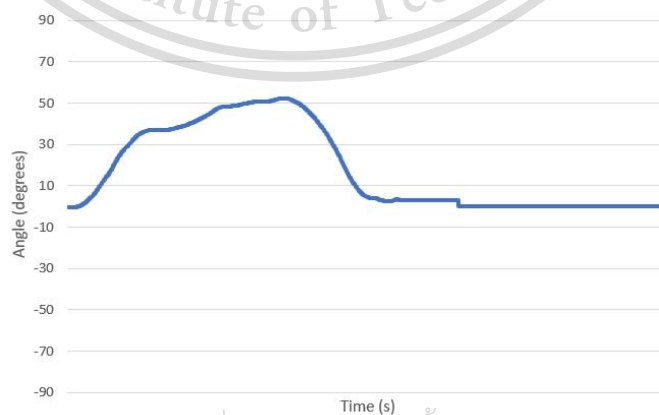
รูปที่ 3.20 การต่อโมดูล MS5611 กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6

เมื่อวางโมดูลไว้ในพื้นที่ปิดสูงระดับเดียวกันพื้นดิน ค่าความสูงที่ได้จากการคำนวณจากค่าที่โมดูลเก็บไว้แสดงดังรูปที่ 3.21



รูปที่ 3.21 ค่าความสูงที่ได้จากโมดูล MS5611 ที่ระดับพื้นดิน

เมื่อทดลองเปลี่ยนตำแหน่งให้เซ็นเซอร์อยู่สูงจากพื้นในแนวระนาบมากขึ้นเรื่อยๆ ค่าความสูงที่ได้จากการคำนวณจากค่าตัวแปรที่โมดูลเก็บไว้จะเปลี่ยนแปลง ดังรูปที่ 3.22



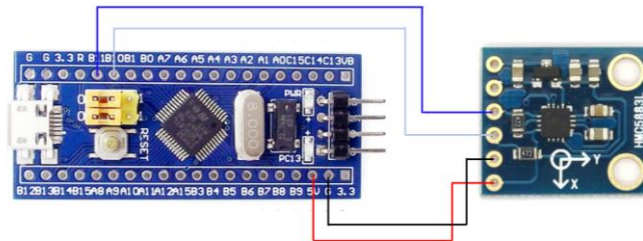
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.22 ค่าความสูงที่ได้จากโมดูล MS5611 เมื่อตำแหน่งของโมดูลมีการเปลี่ยนแปลง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ต่อผู้อื่น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

การทดสอบโมดูลเซ็นเซอร์เข็มทิศ (HMC5883L)

ทดสอบเพื่อดูการตรวจจับเส้นแรงแม่เหล็กของเซ็นเซอร์ เมื่อมีการเคลื่อนที่ของโมดูลค่าที่ได้จากจะเปลี่ยนไป โดยลักษณะการต่อโมดูลเซ็นเซอร์เข้ากับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 จะแสดงดังรูปที่ 3.23



รูปที่ 3.23 การต่อโมดูล HMC5883L กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6

ในการทดสอบจะวางโมดูลนี้อยู่กับที่ จากนั้นทำการขยับโมดูลเพื่อตรวจหาเส้นแรงแม่เหล็กที่โมดูลจับได้ ดังรูปที่ 3.24



รูปที่ 3.24 ค่าเส้นแรงแม่เหล็กที่โมดูล HMC5883L

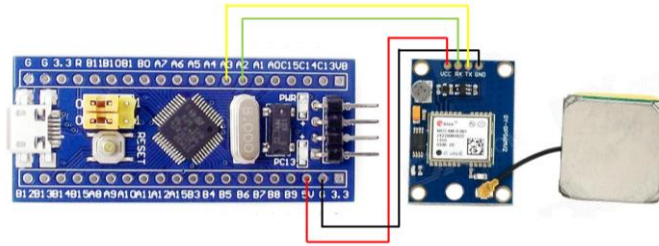
การทดสอบโมดูล Ublox Neo

ทดสอบเพื่อดูการตรวจจับสัญญาณ GPS ของเซ็นเซอร์ เมื่อมีการเคลื่อนที่ของโมดูลค่าที่ได้จากโมดูลจะเปลี่ยนไป โดยลักษณะการต่อโมดูลเซ็นเซอร์เข้ากับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6 จะแสดงดังรูปที่ 3.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



รูปที่ 3.25 การต่อโมดูล Ublox Neo กับบอร์ดไมโครคอนโทรลเลอร์ STM32F103C8T6

ในการทดสอบจะวางโมดูลนี้อยู่กับที่ในที่โล่งแจ้งเพื่อให้สามารถรับสัญญาณ GPS ได้ จากนั้นทำการขยับโมดูลเพื่อรับค่าสัญญาณ GPS ที่เปลี่ยนไป ดังรูปที่ 3.26

```

$GPGGA,3.2,12.74,44.271,26.75,22.333,34.82,09.082,20.85,56.057,30.60
$GPRMC,3.2,A,12.84,S,202.31,E,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
$GNRMC,1.34550.00,A,5231.90586,N,00601.81907,E,0.073,,080718,,,A*6D
$GNTG,1.34550.00,A,5231.90586,N,00601.81907,E,0.073,,080718,,,A*6D
$GNGGA,3.2,12.74,44.271,26.75,22.333,34.82,09.082,20.85,56.057,30.60
$GPGSV,4,1,14,07,16,310,36,08,34,290,35,10,43,147,32,13,07,023,32,70
$GPGSV,4,2,14,15,11,056,39,16,53,192,39,20,53,105,27,21,42,067,36,72
$GPGSV,4,3,14,26,27,175,36,27,73,294,38,30,08,337,36,32,27,206,37,75
$GPGSV,4,4,14,36,25,149,32,49,30,181,36,70
$GPGSV,3,1,12,66,00,003,67,32,043,36,68,35,110,69,06,154,64
$GPGSV,3,2,12,74,44,271,26,75,22,333,34,82,09,082,21,85,56,057,30,61
$GPGSV,3,3,12,84,53,292,31,85,03,272,35,35,34,6F
$GNGGA,3.2,12.74,44.271,26.75,22.333,34.82,09.082,21.85,56.057,30.61
$GNRMC,1.34551.00,A,5231.90589,N,00601.81907,E,0.082,,080718,,,A*60
$GNTG,1.34551.00,A,5231.90589,N,00601.81907,E,0.082,,080718,,,A*60
$GNGGA,3.2,12.74,44.271,26.75,22.333,34.82,09.082,21.85,56.057,30.61
$GPGSV,4,1,14,07,16,310,36,08,34,290,35,10,43,147,32,13,07,023,33,70
$GPGSV,4,2,14,15,11,056,39,16,53,192,39,20,53,105,27,21,42,067,36,72
$GPGSV,4,3,14,26,27,175,36,27,73,294,38,30,08,337,36,32,27,206,37,75

```

รูปที่ 3.26 ค่า NMEA GPS ที่โมดูล GPS อ่านได้

3.5.2 การต่อเซ็นเซอร์และอุปกรณ์ต่างๆ กับบอร์ดไมโครคอนโทรลเลอร์

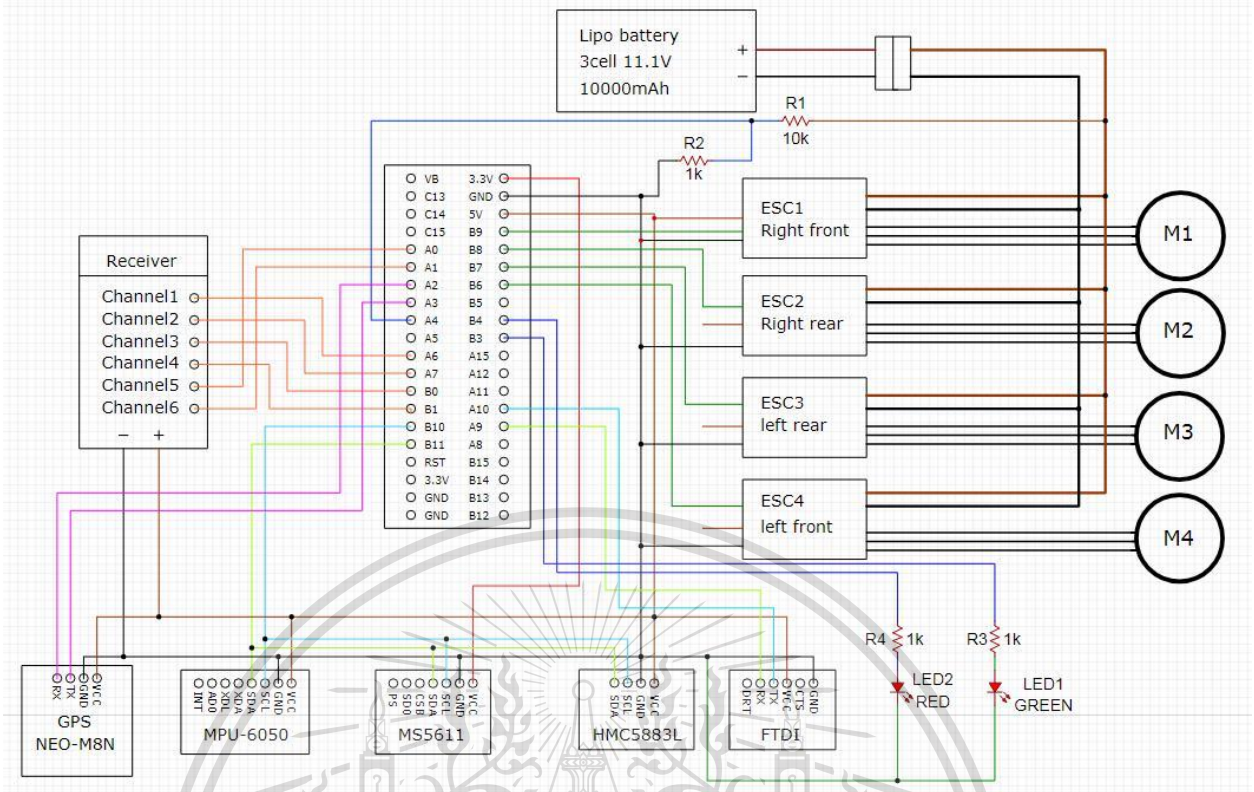
วงจรจะมีแหล่งจ่ายไฟ จากแบตเตอรี่ลิเทียมโพลีเมอร์ 11.1 โวลต์ จ่ายเข้าตัวควบคุมความเร็วมอเตอร์ (ESC) ทั้ง 4 ตัว และ ต่อกับวงจรแบ่งแรงดันเข้ากับบอร์ด STM32F103C8T6 ขา A4 และ GND ตัวควบคุมความเร็วมอเตอร์ตัวที่ 1 (ESC) จะทำหน้าที่แปลงสัญญาณไฟให้เป็น 5V และ จ่ายเข้าบอร์ด STM32F103C8T6 ต่อกับตัวควบคุมความเร็วมอเตอร์ (ESC) ทุกตัวเข้ากับ GND และ ขา B6 B7 B8 B9 ของบอร์ด STM32F103C8T6 และต่อเข้ากับมอเตอร์แต่ละตัวต่อไฟแสดงผล (LED) สีเขียวเข้ากับ B3 สีแดงเข้ากับ B4 ต่อกับโมดูล GPS NEO-M8N MPU-6050 HMC5883 FTDI เข้ากับไฟ 5V และ GND ส่วน MS5611 ต่อกับไฟ 3.3V และ GND โมดูล MPU-6050 HMC5883 MS5611 ใช้การสื่อสารแบบ I²C ใช้ขา SCL ต่อกับขา B11 และ SDA ต่อกับขา B10 ต่อกับขา RX TX ของ GPS NEO-M8N กับขา A2 A3 ตามลำดับ และ ต่อกับขา RX TX ของ FTDI กับขา A9 A10 ตามลำดับ ต่อกับตัวรับสัญญาณจากรีโมทควบคุม (Receiver) โดย Channel 1 ต่อกับขา A6 Channel 2 ต่อกับขา A7 Channel 3 ต่อกับขา B0 Channel 4 ต่อกับขา B1 Channel 5 ต่อกับขา A0 Channel 6 ต่อกับขา A1 ดังรูป 3.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



รูปที่ 3.27 การต่อเซ็นเซอร์ และอุปกรณ์ต่างๆ กับบอร์ดไมโครคอนโทรลเลอร์

3.5.3 การเขียนโปรแกรม Arduino

ขั้นตอนนี้เป็นส่วนของการเขียนโปรแกรมเพื่อนำจากอุปกรณ์ต่างๆ มาประมวลผลและควบคุมให้ได้ผลตามต้องการ

```

YMF32-auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMF32-auto_Flight_Controller_ver-1b $ StateMeter LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass

#include <EEPROM.h>
#include <Wire.h>
TwoWire HWire (2, I2C_FAST_MODE);

//////////////////////////////////
//PID gain and limit settings
//////////////////////////////////
float pid_p_gain_roll = 1.4; //default = 0.2 -> 1.4
float pid_i_gain_roll = 0.0175;
float pid_d_gain_roll = 14.97; //default = 18.7 -> 14.97
int pid_max_roll = 400;

float pid_p_gain_pitch = pid_p_gain_roll;
float pid_i_gain_pitch = pid_i_gain_roll;
float pid_d_gain_pitch = pid_d_gain_roll;
int pid_max_pitch = pid_max_roll;

float pid_p_gain_yaw = 3.0; //default = 3.0
float pid_i_gain_yaw = 0.01; //default = 0.01
float pid_d_gain_yaw = 0.0;
int pid_max_yaw = 400;

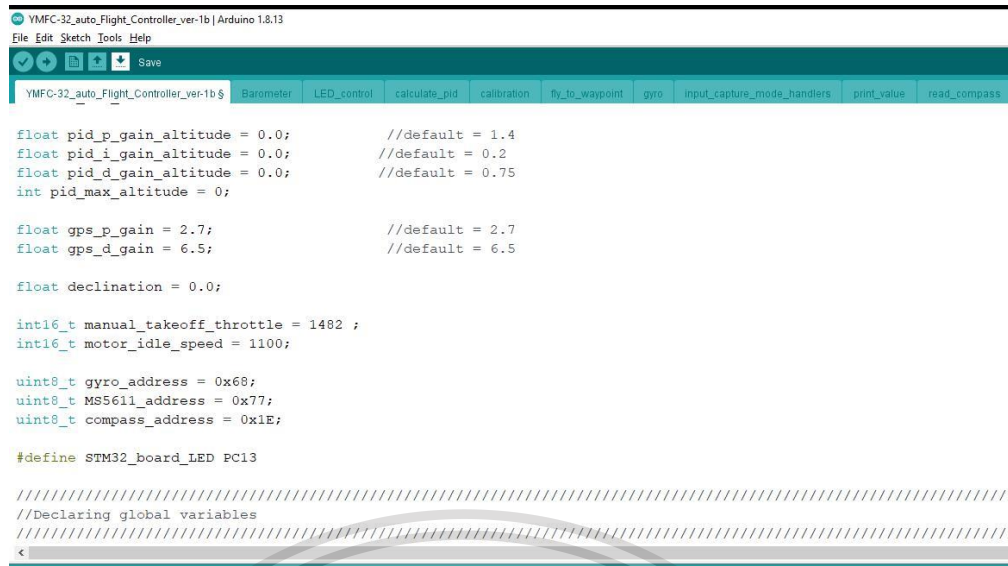
float pid_p_gain_altitude = 0.0; //default = 1.4

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้งานเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.28 หน้าต่างโปรแกรม Arduino (1) [46]



```

float pid_p_gain_altitude = 0.0;           //default = 1.4
float pid_i_gain_altitude = 0.0;         //default = 0.2
float pid_d_gain_altitude = 0.0;         //default = 0.75
int pid_max_altitude = 0;

float gps_p_gain = 2.7;                  //default = 2.7
float gps_d_gain = 6.5;                  //default = 6.5

float declination = 0.0;

int16_t manual_takeoff_throttle = 1402 ;
int16_t motor_idle_speed = 1100;

uint8_t gyro_address = 0x68;
uint8_t MS5611_address = 0x77;
uint8_t compass_address = 0x1E;

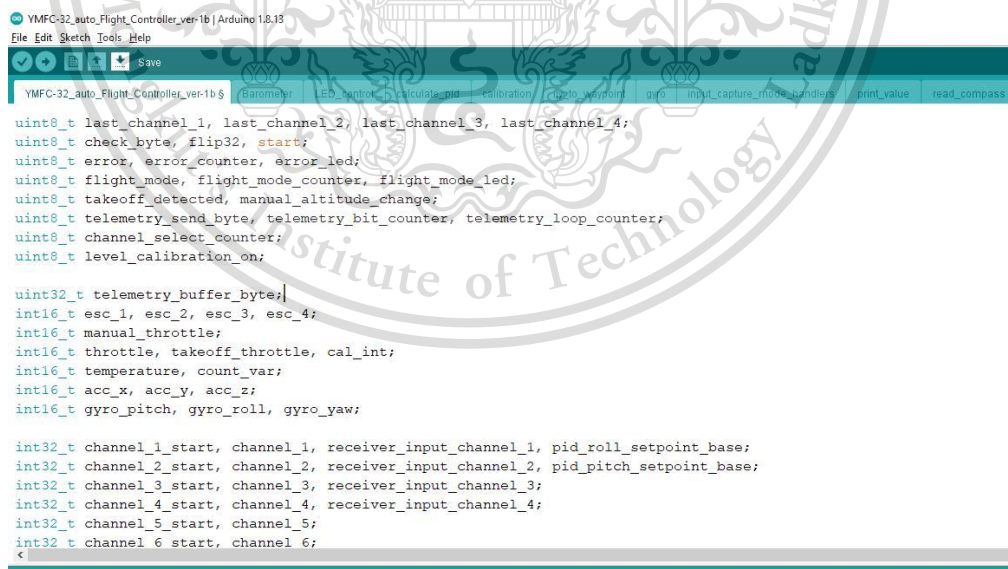
#define STM32_board_LED PC13

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//Declaring global variables
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
<

```

รูปที่ 3.29 หน้าต่างโปรแกรม Arduino (2) [46]

จากรูปที่ 3.28 และ 3.29 จะประกอบด้วย 3 ส่วนคือ ส่วนการเรียกใช้ไลบรารีที่จำเป็นในการสื่อสารระหว่างอุปกรณ์ และกำหนดค่าที่ใช้สื่อสารการเก็บค่าต่างๆกับเซ็นเซอร์ ส่วนที่เป็นการประกาศค่าตัวแปร P controller I controller D controller ที่ใช้ในควบคุมเสถียรภาพในการเคลื่อนที่ในทิศทางต่างๆของโดรน ใช้ควบคุมความสูง และใช้ควบคุมการติดตามพิกัดที่กำหนด และส่วนที่ ประกาศค่าที่ใช้ควบคุมความเร็วมอเตอร์ และประกาศ address ของเซ็นเซอร์



```

uint8_t last_channel_1, last_channel_2, last_channel_3, last_channel_4;
uint8_t check_byte, flip32, start;
uint8_t error, error_counter, error_led;
uint8_t flight_mode, flight_mode_counter, flight_mode_led;
uint8_t takeoff_detected, manual_altitude_change;
uint8_t telemetry_send_byte, telemetry_bit_counter, telemetry_loop_counter;
uint8_t channel_select_counter;
uint8_t level_calibration_on;

uint32_t telemetry_buffer_byte;
int16_t esc_1, esc_2, esc_3, esc_4;
int16_t manual_throttle;
int16_t throttle, takeoff_throttle, cal_int;
int16_t temperature, count_var;
int16_t acc_x, acc_y, acc_z;
int16_t gyro_pitch, gyro_roll, gyro_yaw;

int32_t channel_1_start, channel_1, receiver_input_channel_1, pid_roll_setpoint_base;
int32_t channel_2_start, channel_2, receiver_input_channel_2, pid_pitch_setpoint_base;
int32_t channel_3_start, channel_3, receiver_input_channel_3;
int32_t channel_4_start, channel_4, receiver_input_channel_4;
int32_t channel_5_start, channel_5;
int32_t channel_6_start, channel_6;
<

```

รูปที่ 3.30 หน้าต่างโปรแกรม Arduino (3) [46]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```

YMF32_auto_Flight_Controller_ver1b | Arduino 1.8.13
File Edit Sketch Tools Help
Save
YMF32_auto_Flight_Controller_ver1b | Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass
int32_t channel_6_start, channel_6;
int32_t measured_time, measured_time_start;
int32_t acc_total_vector, acc_total_vector_at_start;
int32_t gyro_roll_cal, gyro_pitch_cal, gyro_yaw_cal;
int16_t acc_pitch_cal_value;
int16_t acc_roll_cal_value;

int32_t acc_z_average_short_total, acc_z_average_long_total, acc_z_average_total ;
int16_t acc_z_average_short[26], acc_z_average_long[51];

uint8_t acc_z_average_short_rotating_mem_location, acc_z_average_long_rotating_mem_location;

int32_t acc_alt_integrated;

uint32_t loop_timer, error_timer, flight_mode_timer;

float roll_level_adjust, pitch_level_adjust;
float pid_error_temp;
float pid_i_mem_roll, pid_roll_setpoint, gyro_roll_input, pid_output_roll, pid_last_roll_d_error;
float pid_i_mem_pitch, pid_pitch_setpoint, gyro_pitch_input, pid_output_pitch, pid_last_pitch_d_error;
float pid_i_mem_yaw, pid_yaw_setpoint, gyro_yaw_input, pid_output_yaw, pid_last_yaw_d_error;
float angle_roll_acc, angle_pitch_acc, angle_roll, angle_yaw;
float battery_voltage, dummy_float;

```

รูปที่ 3.31 หน้าต่างโปรแกรม Arduino (4) [46]

```

YMF32_auto_Flight_Controller_ver1b | Arduino 1.8.13
File Edit Sketch Tools Help
Save
YMF32_auto_Flight_Controller_ver1b | Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass

//Compass variables
uint8_t compass_calibration_on, heading_lock;
int16_t compass_x, compass_y, compass_z;
int16_t compass_cal_values[6];
float compass_x_horizontal, compass_y_horizontal, actual_compass_heading;
float compass_scale_y, compass_scale_z;
int16_t compass_offset_x, compass_offset_y, compass_offset_z;
float course_a, course_b, course_c, base_course_mirrored, actual_course_mirrored;
float course_lock_heading, heading_lock_course_deviation;

//Pressure variables.
float pid_error_gain_altitude, pid_throttle_gain_altitude;
uint16_t C[7];
uint8_t barometer_counter, temperature_counter, average_temperature_mem_location;
int64_t OFF, OFF_C2, SENS, SENS_C1, P;
uint32_t raw_pressure, raw_temperature, temp, raw_temperature_rotating_memory[6], raw_average_temperature_total;
float actual_pressure, actual_pressure_slow, actual_pressure_fast, actual_pressure_diff;
float ground_pressure, altitude_hold_pressure;
int32_t dT, dT_C5;
//Altitude PID variables
float pid_i_mem_altitude, pid_altitude_setpoint, pid_altitude_input, pid_output_altitude, pid_last_altitude_d_error;
uint8_t parachute_rotating_mem_location;

```

รูปที่ 3.32 หน้าต่างโปรแกรม Arduino (5) [46]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```

YMFC-32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMFC-32_auto_Flight_Controller_ver-1b $ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass rea
//Altitude PID variables
float pid_i_mem_altitude, pid_altitude_setpoint, pid_altitude_input, pid_output_altitude, pid_last_altitude_d_error;
uint8_t parachute_rotating_mem_location;
int32_t parachute_buffer[35], parachute_throttle;
float pressure_parachute_previous;
int32_t pressure_rotating_mem[50], pressure_total_avarage;
uint8_t pressure_rotating_mem_location;
float pressure_rotating_mem_actual;
//GPS variables
uint8_t read_serial_byte, incomming_message[100], number_used_sats, fix_type;
uint8_t waypoint_set, latitude_north, longiude_east ;
uint16_t message_counter;
int16_t gps_add_counter;
int32_t l_lat_gps, l_lon_gps, lat_gps_previous, lon_gps_previous;
int32_t lat_gps_actual, lon_gps_actual, l_lat_waypoint, l_lon_waypoint;
float gps_pitch_adjust_north, gps_pitch_adjust, gps_roll_adjust_north, gps_roll_adjust;
float lat_gps_loop_add, lon_gps_loop_add, lat_gps_add, lon_gps_add;
uint8_t new_line_found, new_gps_data_available, new_gps_data_counter;
uint8_t gps_rotating_mem_location;
int32_t gps_lat_total_avarage, gps_lon_total_avarage;
int32_t gps_lat_rotating_mem[40], gps_lon_rotating_mem[40];
int32_t gps_lat_error, gps_lon_error;
int32_t gps_lat_error_previous, gps_lon_error_previous;
uint32_t gps_watchdog_timer;
<

```

รูปที่ 3.33 หน้าต่างโปรแกรม Arduino (6) [46]

จากรูปที่ 3.30 3.31 3.32 และ 3.33 เป็นส่วนที่ประกาศตัวแปรอื่นๆที่ใช้ในการเขียนโปรแกรมควบคุมการทำงาน โดยจะมีส่วนที่เป็นตัวแปรทั่วไป ส่วนที่เป็นตัวแปรรีโมท ส่วนที่เป็นตัวแปรเข็มทิศ ส่วนที่เป็นตัวแปรความดัน ส่วนที่เป็นตัวแปรความสูง และส่วนที่เป็นตัวแปรระบบนำทาง

```

YMFC-32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMFC-32_auto_Flight_Controller_ver-1b $ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass rea
void setup() {
  pinMode(4, INPUT_ANALOG);
  afio_cfg_debug_ports(AFIO_DEBUG_SW_ONLY);
  pinMode(PB3, OUTPUT);
  pinMode(PB4, OUTPUT);
  pinMode(STM32_board_LED, OUTPUT);
  digitalWrite(STM32_board_LED, HIGH);
  green_led(LOW);
  red_led(HIGH);
  //EEPROM emulation setup
  EEPROM.PageBase0 = 0x801F000;
  EEPROM.PageBase1 = 0x801F800;
  EEPROM.PageSize = 0x400;
  // Serial.begin(57600); //for debugging
  // delay(250);
  timer_setup();
  delay(50);
  gps_setup();
  //Check if the MPU-6050 is responding.
  HWire.begin();
  HWire.beginTransmission(gyro_address);
  error = HWire.endTransmission();
  while (error != 0) {
    error = 1;
  }
}
<

```

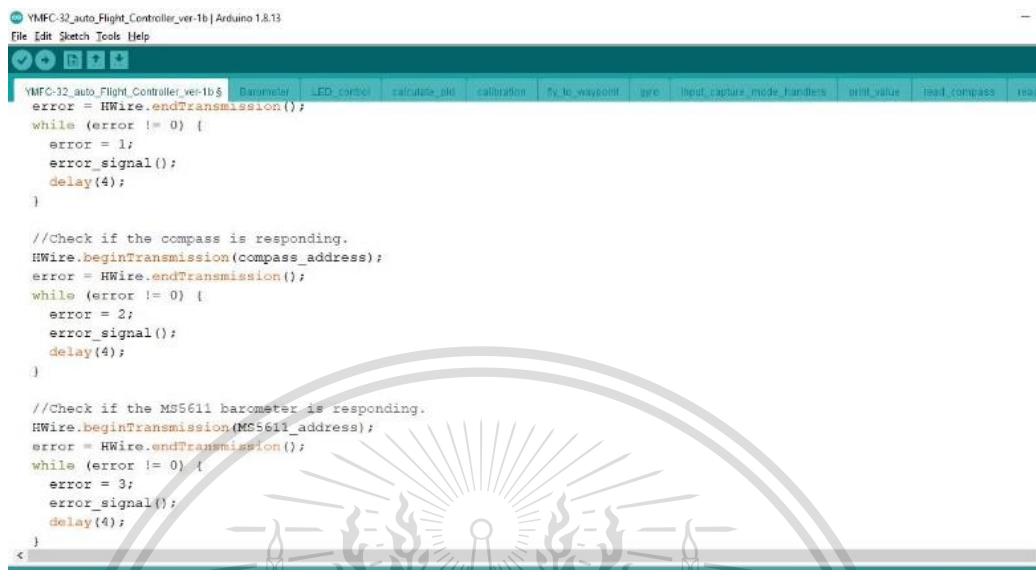
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.34 หน้าต่างโปรแกรม Arduino (7) [46] หน้าไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

จากรูปที่ 3.34 เป็นการกำหนดขาอินพุต ขาเอาต์พุต ที่เชื่อมต่ออยู่กับอุปกรณ์เซ็นเซอร์ และอุปกรณ์แสดงผลต่างๆ และเรียกใช้ฟังก์ชันนับเวลา และเริ่มการใช้งานอุปกรณ์ระบบนำทาง



```

YMF32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMF32_auto_Flight_Controller_ver-1b$ Barometer LED_control calculate_pid calibration fly_by_wire_control gyro thrust_capture_mode_handlers print_value read_compass read

while (error != 0) {
  error = 1;
  error_signal();
  delay(4);
}

//Check if the compass is responding.
HWire.beginTransmission(compass_address);
error = HWire.endTransmission();
while (error != 0) {
  error = 2;
  error_signal();
  delay(4);
}

//Check if the MS5611 barometer is responding.
HWire.beginTransmission(MS5611_address);
error = HWire.endTransmission();
while (error != 0) {
  error = 3;
  error_signal();
  delay(4);
}

```

รูปที่ 3.35 หน้าต่างโปรแกรม Arduino (8) [46]

จากรูปที่ 3.35 เป็นส่วนที่ตรวจสอบการทำงานของเซ็นเซอร์ต่างๆของโดรนว่าทำงานอยู่หรือไม่



```

YMF32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMF32_auto_Flight_Controller_ver-1b$ Barometer LED_control calculate_pid calibration fly_by_wire_control gyro input_capture_mode_handlers print_value read_compass read

gyro_setup();
setup_compass();
read_compass();
angle_yaw = actual_compass_heading;

//Create a 5 second delay before calibration.
for (count_var = 0; count_var < 1250; count_var++) {
  if (count_var % 125 == 0) {
    digitalWrite(PB4, !digitalRead(PB4));
  }
  delay(4);
}
count_var = 0;
calibrate_gyro();

//Wait until the receiver is active.
while (channel_1 < 990 || channel_2 < 990 || channel_3 < 990 || channel_4 < 990) {
  error = 4;
  error_signal();
  delay(4);
}
error = 0;

```

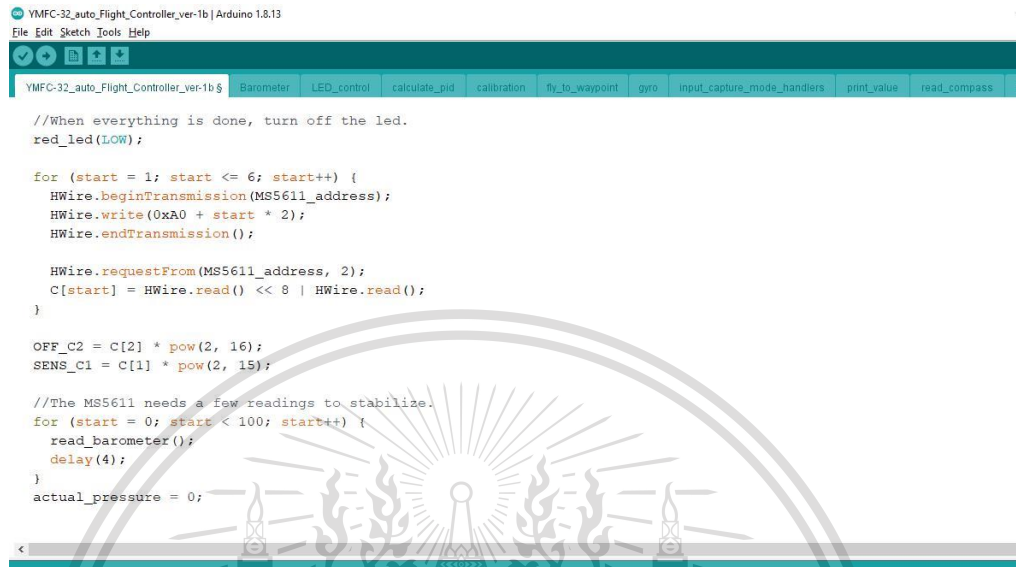
รูปที่ 3.36 หน้าต่างโปรแกรม Arduino (9) [46]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ทางวิชาการเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

จากรูปที่ 3.36 เป็นการปรับเทียบเซ็นเซอร์ก่อนการใช้งานในการควบคุมเสถียรภาพของโดรน โดยปรับเทียบ 5 วินาที และแสดงผลผ่าน LED หลังการปรับเทียบ



```

YMF32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMF32_auto_Flight_Controller_ver-1b $ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass

//When everything is done, turn off the led.
red_led(LOW);

for (start = 1; start <= 6; start++) {
  HWire.beginTransaction(MS5611_address);
  HWire.write(0xA0 + start * 2);
  HWire.endTransmission();

  HWire.requestFrom(MS5611_address, 2);
  C[start] = HWire.read() << 8 | HWire.read();
}

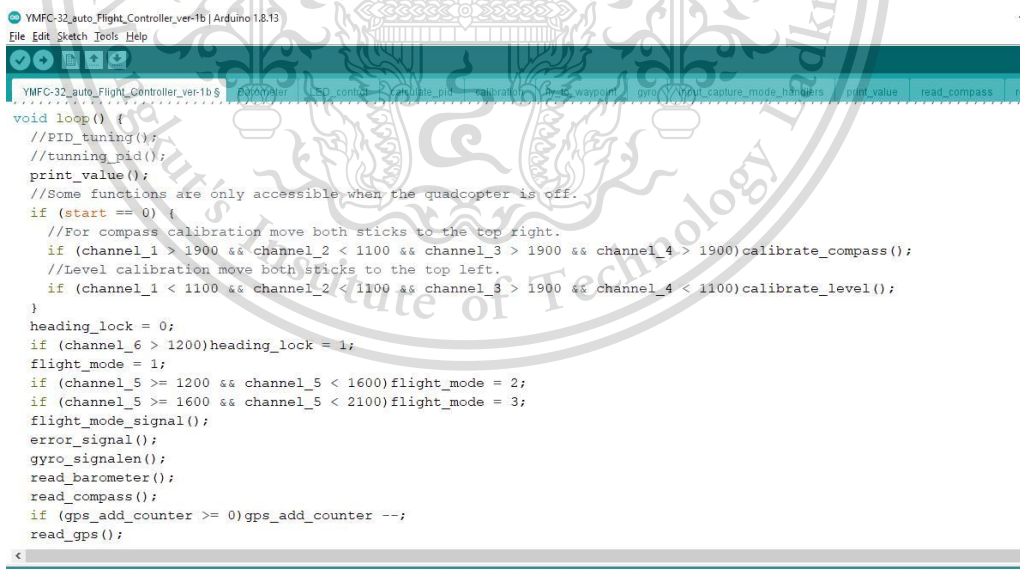
OFF_C2 = C[2] * pow(2, 16);
SENS_C1 = C[1] * pow(2, 15);

//The MS5611 needs a few readings to stabilize.
for (start = 0; start < 100; start++) {
  read_barometer();
  delay(4);
}
actual_pressure = 0;

```

รูปที่ 3.37 หน้าต่างโปรแกรม Arduino (10) [46]

จากรูปที่ 3.37 เริ่มเก็บค่าความดันจากเซ็นเซอร์ และนำมาคำนวณ



```

YMF32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMF32_auto_Flight_Controller_ver-1b $ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass

void loop() {
  //PID tuning();
  //tunning_pid();
  print_value();
  //Some functions are only accessible when the quadcopter is off.
  if (start == 0) {
    //For compass calibration move both sticks to the top right.
    if (channel_1 > 1900 && channel_2 < 1100 && channel_3 > 1900 && channel_4 > 1900)calibrate_compass();
    //Level calibration move both sticks to the top left.
    if (channel_1 < 1100 && channel_2 < 1100 && channel_3 > 1900 && channel_4 < 1100)calibrate_level();
  }
  heading_lock = 0;
  if (channel_6 > 1200)heading_lock = 1;
  flight_mode = 1;
  if (channel_5 >= 1200 && channel_5 < 1600)flight_mode = 2;
  if (channel_5 >= 1600 && channel_5 < 2100)flight_mode = 3;
  flight_mode_signal();
  error_signal();
  gyro_signalen();
  read_barometer();
  read_compass();
  if (gps_add_counter >= 0)gps_add_counter --;
  read_gps();
}

```

รูปที่ 3.38 หน้าต่างโปรแกรม Arduino (11) [46]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาระดับสูงเท่านั้น ไม่อนุญาตให้ทำซ้ำโดยไม่ได้รับอนุญาต
จากรูปที่ 3.38 เป็นส่วนการเรียกใช้ฟังก์ชันในการประมวลผลการทำงานของโดรน
ไม่ว่ากรณีใดก็ตาม ห้ามเผยแพร่ข้อมูลโปรแกรมนี้บนอินเทอร์เน็ตหรือสื่ออื่นใดโดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
ทั้งการเปลี่ยนโหมดบินและ เสถียรภาพในการบิน

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```

YMFC-32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMFC-32_auto_Flight_Controller_ver-1b$ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass

//65.5 = 1 deg/sec (check the datasheet of the MPU-6050 for more information).
gyro_roll_input = (gyro_roll_input * 0.7) + (((float)gyro_roll / 65.5) * 0.3);
gyro_pitch_input = (gyro_pitch_input * 0.7) + (((float)gyro_pitch / 65.5) * 0.3);
gyro_yaw_input = (gyro_yaw_input * 0.7) + (((float)gyro_yaw / 65.5) * 0.3);

//Gyro angle calculations
//0.0000611 = 1 / (250Hz / 65.5)
angle_pitch += (float)gyro_pitch * 0.0000611;
angle_roll += (float)gyro_roll * 0.0000611;
angle_yaw += (float)gyro_yaw * 0.0000611;
if (angle_yaw < 0) angle_yaw += 360;
else if (angle_yaw >= 360) angle_yaw -= 360;

//0.000001066 = 0.0000611 * (3.142(PI) / 180degr) The Arduino sin function is in radians and not degrees.
angle_pitch -= angle_roll * sin((float)gyro_yaw * 0.000001066);
angle_roll += angle_pitch * sin((float)gyro_yaw * 0.000001066);

angle_yaw -= course_deviation(angle_yaw, actual_compass_heading) / 1200.0;
if (angle_yaw < 0) angle_yaw += 360;
else if (angle_yaw >= 360) angle_yaw -= 360;

```

รูปที่ 3.39 หน้าต่างโปรแกรม Arduino (12) [46]

```

YMFC-32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMFC-32_auto_Flight_Controller_ver-1b$ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass

//Accelerometer angle calculations
acc_total_vector = sqrt((acc_x * acc_x) + (acc_y * acc_y) + (acc_z * acc_z));
if (abs(acc_y) < acc_total_vector) {
  angle_pitch_acc = asin((float)acc_y / acc_total_vector) * 57.296;
}
if (abs(acc_x) < acc_total_vector) {
  angle_roll_acc = asin((float)acc_x / acc_total_vector) * 57.296;
}

angle_pitch = angle_pitch * 0.9996 + angle_pitch_acc * 0.0004 + 4.0;
angle_roll = angle_roll * 0.9996 + angle_roll_acc * 0.0004;

pitch_level_adjust = angle_pitch * 15;
roll_level_adjust = angle_roll * 15;

vertical_acceleration_calculations();

pid_roll_setpoint_base = channel_1;
pid_pitch_setpoint_base = channel_2;

if (heading_lock == 1) {
  heading_lock_course_deviation = course_deviation(angle_yaw, course_lock_heading);
}

```

รูปที่ 3.40 หน้าต่างโปรแกรม Arduino (13) [46]

จากรูปที่ 3.39 และ 3.40 เป็นส่วนในการคำนวณค่ามุมที่เปลี่ยนไปของเซ็นเซอร์ซึ่งนำมาใช้ในการประมวลผลควบคุมเสถียรภาพของโดรนด้วย PID controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```

YMF32_auto_Flight_Controller_ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMF32_auto_Flight_Controller_ver-1b $ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass rs

if (heading_lock == 1) {
  heading_lock_course_deviation = course_deviation(angle_yaw, course_lock_heading);
  pid_roll_setpoint_base = 1500 + ((float)(channel_1 - 1500) * cos(heading_lock_course_deviation * 0.017453)) +
  ((float)(channel_2 - 1500) * cos((heading_lock_course_deviation - 90) * 0.017453));
  pid_pitch_setpoint_base = 1500 + ((float)(channel_2 - 1500) * cos(heading_lock_course_deviation * 0.017453)) +
  ((float)(channel_1 - 1500) * cos((heading_lock_course_deviation + 90) * 0.017453));
}

if (flight_mode >= 3 && waypoint_set == 1) {
  pid_roll_setpoint_base += gps_roll_adjust;
  pid_pitch_setpoint_base += gps_pitch_adjust;
}

//Because we added the GPS adjust values we need to make sure that the control limits are not exceeded.
if (pid_roll_setpoint_base > 2000)pid_roll_setpoint_base = 2000;
if (pid_roll_setpoint_base < 1000)pid_roll_setpoint_base = 1000;
if (pid_pitch_setpoint_base > 2000)pid_pitch_setpoint_base = 2000;
if (pid_pitch_setpoint_base < 1000)pid_pitch_setpoint_base = 1000;

calculate_pid();

start_stop_takeoff();

```

รูปที่ 3.41 หน้าต่างโปรแกรม Arduino (14) [46]

จากรูปที่ 3.41 เป็นส่วนที่ประมวลผลโหมดการเคลื่อนที่ของโดรนให้โดรนเคลื่อนที่ตามทิศทางที่กำหนดไว้ กำหนดเงื่อนไขในการเปลี่ยนโหมด และเปลี่ยนค่าที่ใช้ในการควบคุมโดรน และกำหนดค่าสูงสุดของพารามิเตอร์ PID

```

YMF32 auto Flight Controller ver-1b | Arduino 1.8.13
File Edit Sketch Tools Help

YMF32 auto Flight Controller ver-1b $ Barometer LED_control calculate_pid calibration fly_to_waypoint gyro input_capture_mode_handlers print_value read_compass rs

//The variable base throttle is calculated in the following part. It forms the base throttle for every motor.
if (takeoff_detected == 1 && start == 2) {
  throttle = channel_3 + takeoff_throttle;
  if (flight_mode >= 2) {
    throttle = 1500 + takeoff_throttle / pid_output_altitude * manual_throttle;
  }
}

if (start == 2) {
  if (throttle > 1900) throttle = 1900;
  esc_1 = throttle - pid_output_pitch + pid_output_roll + pid_output_yaw;
  esc_2 = throttle + pid_output_pitch + pid_output_roll + pid_output_yaw;
  esc_3 = throttle + pid_output_pitch - pid_output_roll - pid_output_yaw;
  esc_4 = throttle - pid_output_pitch - pid_output_roll + pid_output_yaw;

  if (esc_1 < motor_idle_speed) esc_1 = motor_idle_speed;
  if (esc_2 < motor_idle_speed) esc_2 = motor_idle_speed;
  if (esc_3 < motor_idle_speed) esc_3 = motor_idle_speed;
  if (esc_4 < motor_idle_speed) esc_4 = motor_idle_speed;

  if (esc_1 > 2000)esc_1 = 2000;
  if (esc_2 > 2000)esc_2 = 2000;
  if (esc_3 > 2000)esc_3 = 2000;
}

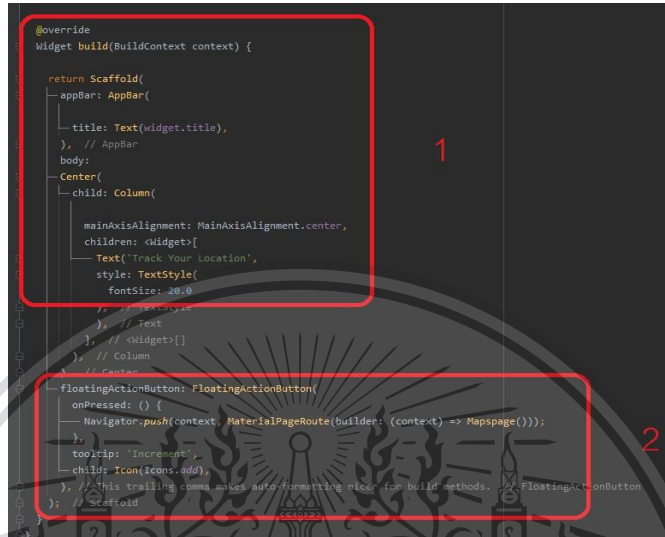
```

รูปที่ 3.42 หน้าต่างโปรแกรม Arduino (15) [46]

จากรูปที่ 3.42 เป็นการเปลี่ยนโหมดในการบิน และเปลี่ยนสมการการรับค่าที่ไปขับมอเตอร์ เอกสารนี้เป็นและเป็นส่วนที่คำนวณค่าจากสมการที่กำหนดและนำค่าที่ได้ไปขับมอเตอร์ให้นำไปใช้ประโยชน์ด้านการคำนวณว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.4 การสร้างแอปพลิเคชันภาษา Flutter ด้วยโปรแกรม Android Studio

ขั้นตอนนี้เป็น การสร้างแอปพลิเคชันสำหรับแสดงผลบนสมาร์ตโฟนโดยใช้โปรแกรม Android Studio ซึ่งในการสร้างแอปพลิเคชันแบ่งเป็น 2 ส่วน คือ ระบบการระบุตำแหน่ง กับ ระบบที่ส่งข้อมูลตำแหน่งพิกัดไปยังเซิร์ฟเวอร์ [23][24]



```

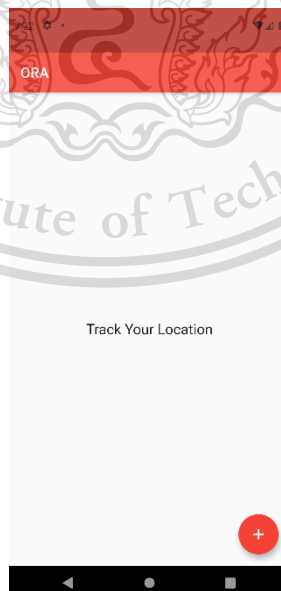
@override
Widget build(BuildContext context) {

  return Scaffold(
    appBar: AppBar(
      title: Text(widget.title),
    ), // AppBar
    body:
    Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          Text('Track Your Location',
            style: TextStyle(
              fontSize: 20.0
            ), // TextStyle
          ), // Text
        ], // <Widget>[]
      ), // Column
    ), // Center
    floatingActionButton: FloatingActionButton(
      onPressed: () {
        Navigator.push(context, MaterialPageRoute(builder: (context) => Mapspage()));
      },
      tooltip: 'Increment',
      child: Icon(Icons.add),
    ), // This trailing comma makes auto-formatting nicer for build methods. // FloatingActionButton
  ); // Scaffold
}

```

รูปที่ 3.43 หน้าต่างของโปรแกรม Android Studio ที่แสดงผลในหน้าแรก

จากรูปที่ 3.43 แสดงการเขียนโปรแกรมในส่วนแสดงผลหน้าแรกของแอปพลิเคชัน หมายเลข 1 แสดงข้อความที่จะบอกให้ควรทำอะไรเมื่อเข้ามาในหน้าแอปพลิเคชัน หมายเลข 2 มีการสร้างปุ่มกดทางด้านขวาล่าง ตั้งค่าขวามุมขวาให้ถ้ามีการกดจะไปแสดงผลหน้าถัดไป ดังรูป 3.44



รูปที่ 3.44 หน้าแรกของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในเชิงพาณิชย์เท่านั้น มิใช่ผู้ผูกขาดให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

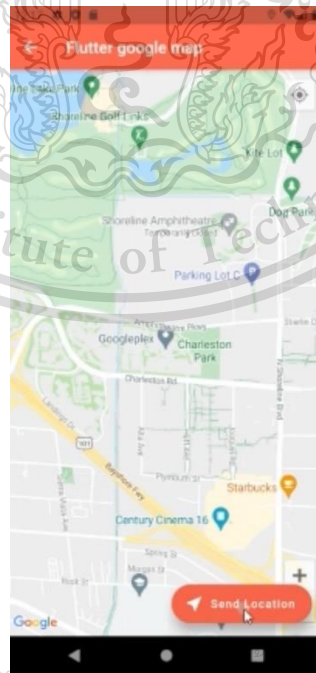
```

final Future<FirebaseApp> firebase = Firebase.initializeApp();
CollectionReference _droneCollection = FirebaseFirestore.instance.collection('drone');
@override
Widget build(BuildContext context) {
  return FutureBuilder(
    future: firebase,
    // ignore: missing_return
    builder: (context, snapshot) {
      if (snapshot.connectionState == ConnectionState.done) {
        return Scaffold(
          appBar: AppBar(
            title: Text('Dra2 map '),
          ), // AppBar
          body: FutureBuilder(future: _getLocation(),
            builder: (BuildContext context, AsyncSnapshot snapshot) {
              if (snapshot.hasData) {
                return GoogleMap(
                  mapType: MapType.normal,
                  onMapCreated: _onMapCreated,
                  myLocationEnabled: true,
                  initialCameraPosition: CameraPosition(
                    target:
                      LatLng(userLocation.latitude, userLocation.longitude),
                    zoom: 15), // CameraPosition
                ); // GoogleMap
              }
              else {
                return Center(
                  child: Column(
                    mainAxisAlignment: MainAxisAlignment.center,
                    children: <Widget>[
                      CircularProgressIndicator(),
                    ] // <Widget>[]
                  ), // Column
                );
              }
            }
          );
        }
      }
    );
  }
}

```

รูปที่ 3.45 หน้าต่างของโปรแกรม Android Studio ที่แสดงผลในหน้าจอ

จากรูปที่ 3.45 จะแสดงในส่วนแอปพลิเคชันหน้าต่างที่ 2 หมายเลข 3 เป็นการเชื่อมต่อแอปพลิเคชันกับตัวเซิร์ฟเวอร์บน Firebase หมายเลข 4 เป็นการดึงแผนที่บน Google Map มาแสดงบนแอปพลิเคชันและแสดงตำแหน่งของแอปพลิเคชันในรูปแบบละติจูดกับลองจิจูดดังรูป 3.46



รูปที่ 3.46 หน้าจอของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

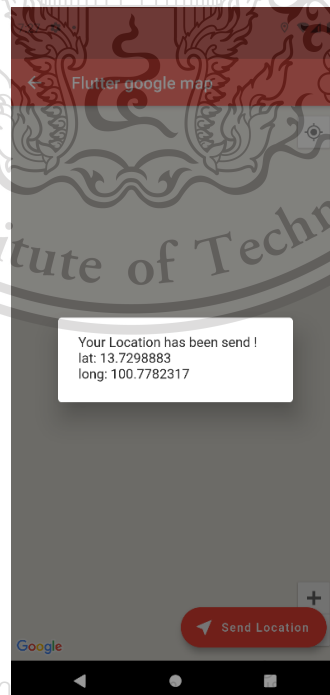
```

    },
    // FutureBuilder
    floatingActionButton:
    FloatingActionButton.extended(
      onPressed: () async{
        mapController.animateCamera(CameraUpdate.newLatLngZoom(
          LatLng(userLocation.latitude, userLocation.longitude),
          18));
        print('${userLocation.latitude}, ${userLocation.longitude}');
        await_droneCollection.add({
          'latitude': '${userLocation.latitude}',
          'longitude': '${userLocation.longitude}'
        });
        showDialog(context: context, builder: (context) {
          return AlertDialog(
            content: Text(
              'Your Location has been send ! \nlat: ${userLocation
                .latitude} \nlong: ${userLocation.longitude}'), // Text
            ); // AlertDialog
          );
        });
      }, label: Text("Send Location"),
      icon: Icon(Icons.near_me), // FloatingActionButton.extended
    ),
  ),
)
}

```

รูปที่ 3.47 หน้าต่างของโปรแกรม Android Studio ด้านล่างที่แสดงผลในหน้าจอ

จากรูป 3.47 เป็นการตั้งค่าปุ่มกดทางด้านขวาล่างให้ถ้ามีการกดจะมีการส่งตำแหน่งละติจูดและลองจิจูดไปที่เซิร์ฟเวอร์ทันที โดยสร้างข้อความที่เขียนว่า Send Location เพื่อง่ายต่อการเข้าใจ และเมื่อกดปุ่มขวาล่าง หน้าจอของแอปพลิเคชันจะแสดงข้อความว่าส่งค่าสำเร็จ และบอกค่าละติจูดกับลองจิจูด

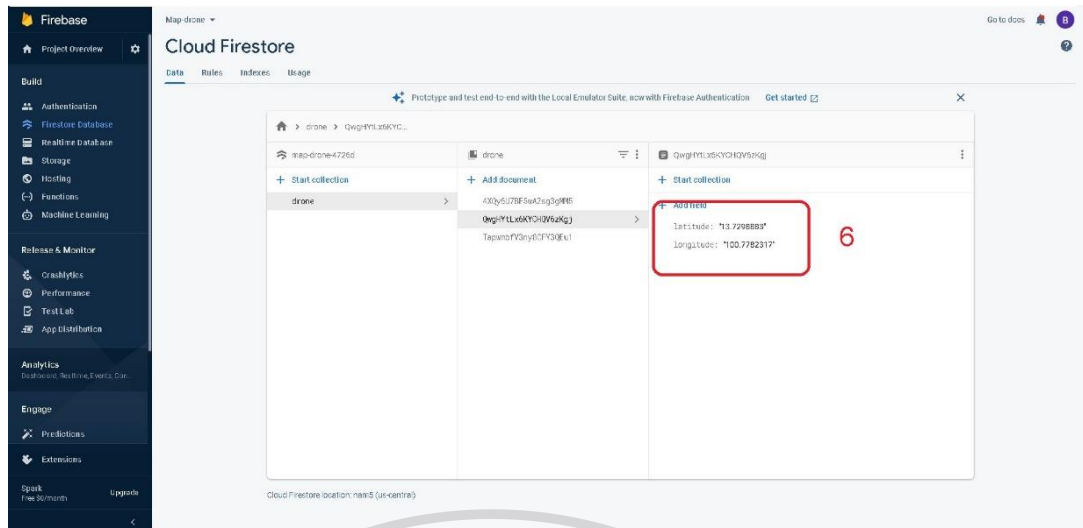


รูปที่ 3.48 หน้าของการกดปุ่มขวาล่าง Send Location แอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานส่วนบุคคลเท่านั้น ไม่ควรเผยแพร่หรือใช้เพื่อวัตถุประสงค์ทางการค้า

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

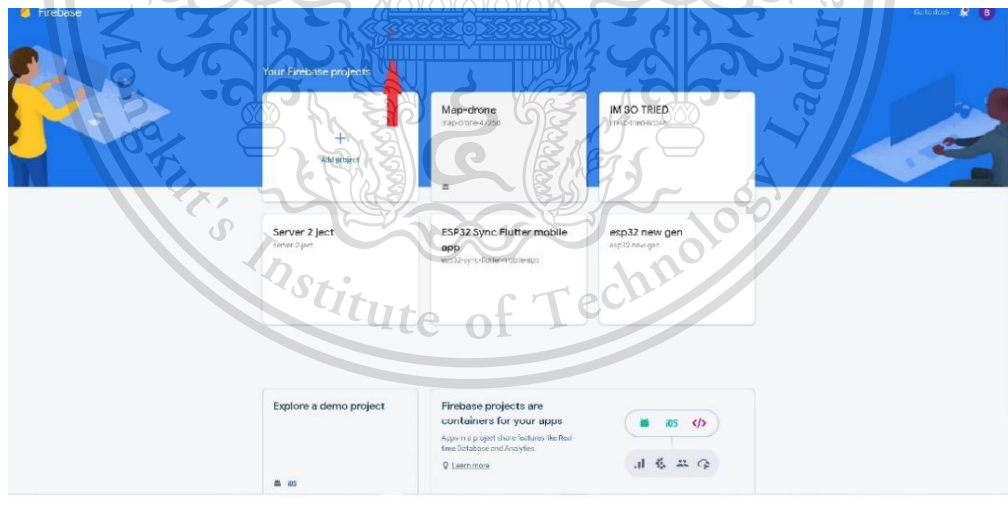


รูปที่ 3.49 เว็บไซต์ Firebase ที่เป็นเซิร์ฟเวอร์

จากรูป 3.49 ด้านบน หมายเลข 6 เป็นค่าละติจูดกับลองจิจูดที่ถูกส่งมาจากแอปพลิเคชันจากการกดปุ่มกลางขวาในแอปพลิเคชันในรูป 3.48

3.5.5 ตั้งค่า Cloud Firestore บน Firebase

เข้าไปที่เว็บไซต์ Firebase กดที่หมายเลข 1 เพื่อสร้างโปรเจกต์ [25]



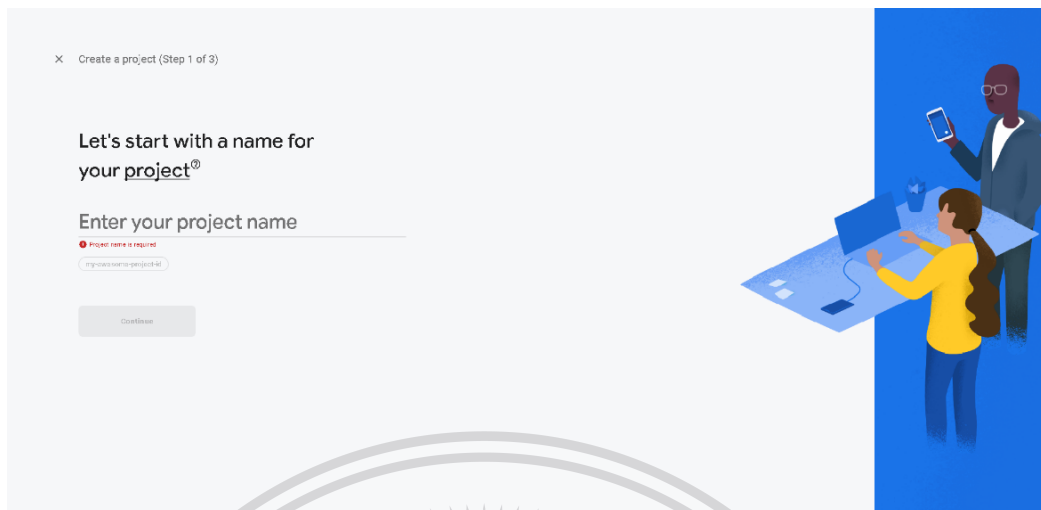
รูปที่ 3.50 หน้าแรกของเว็บไซต์ Firebase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

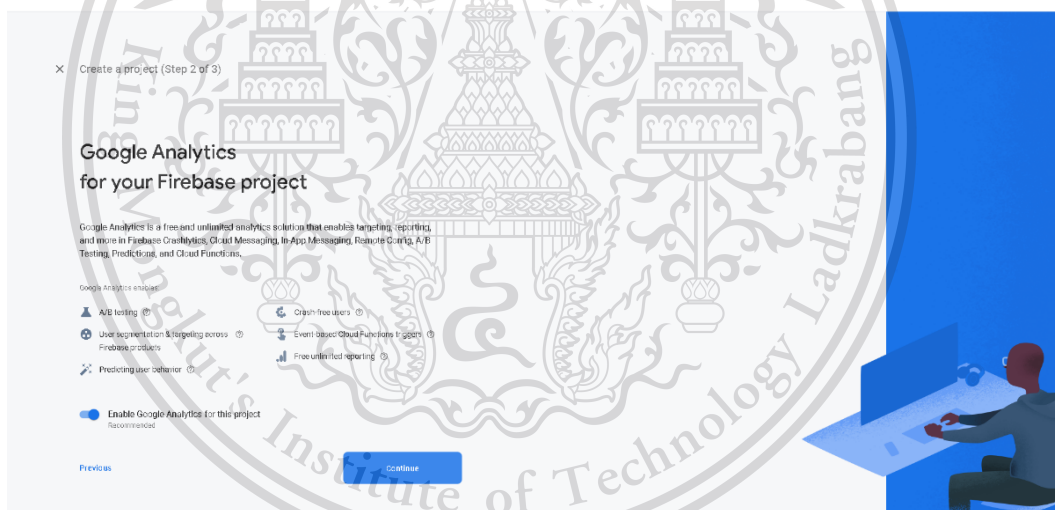
Forbidden to modify the content, and cite the document when use.

ตั้งชื่อโครงการ



รูปที่ 3.51 หน้าตั้งชื่อ Project

ยอมรับการเข้าถึงของเว็บไซต์ และกด Continue



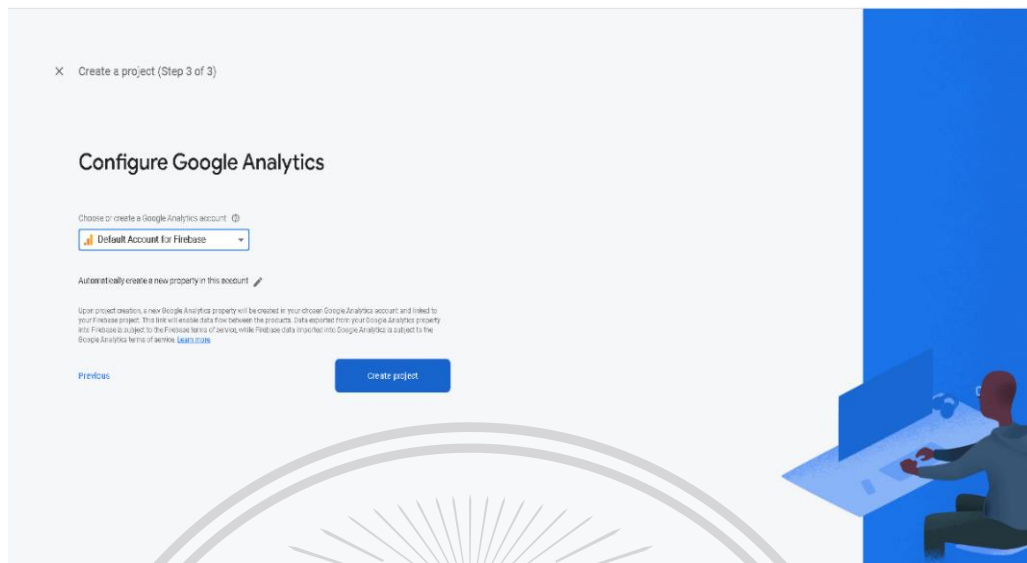
รูปที่ 3.52 หน้าอนุญาตการใช้งานฟังก์ชันต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

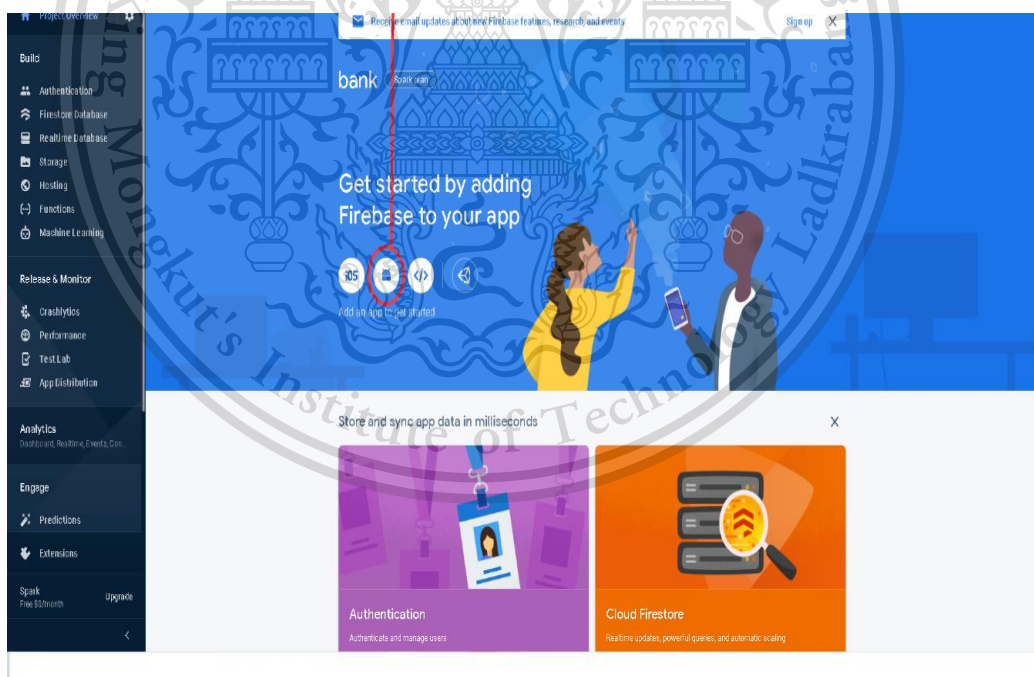
Forbidden to modify the content, and cite the document when use.

กต Create Project เพื่อสร้างโปรเจก



รูปที่ 3.53 หน้าสุดท้ายของการสร้าง Project

หมายเลข 7 เลือกสร้างเซิร์ฟเวอร์ที่เชื่อมกับระบบแอนดรอยด์



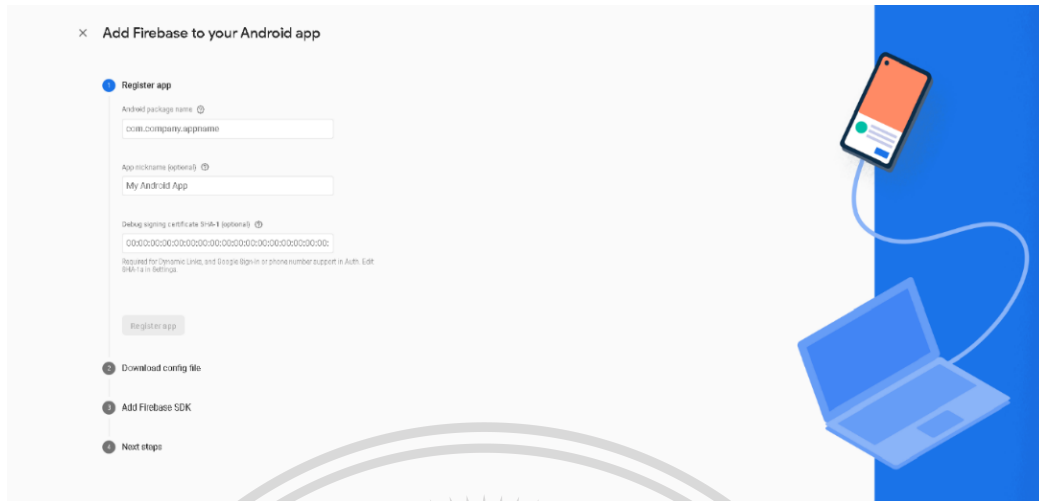
รูปที่ 3.54 หน้าต่างเลือกระบบแอนดรอยด์การใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

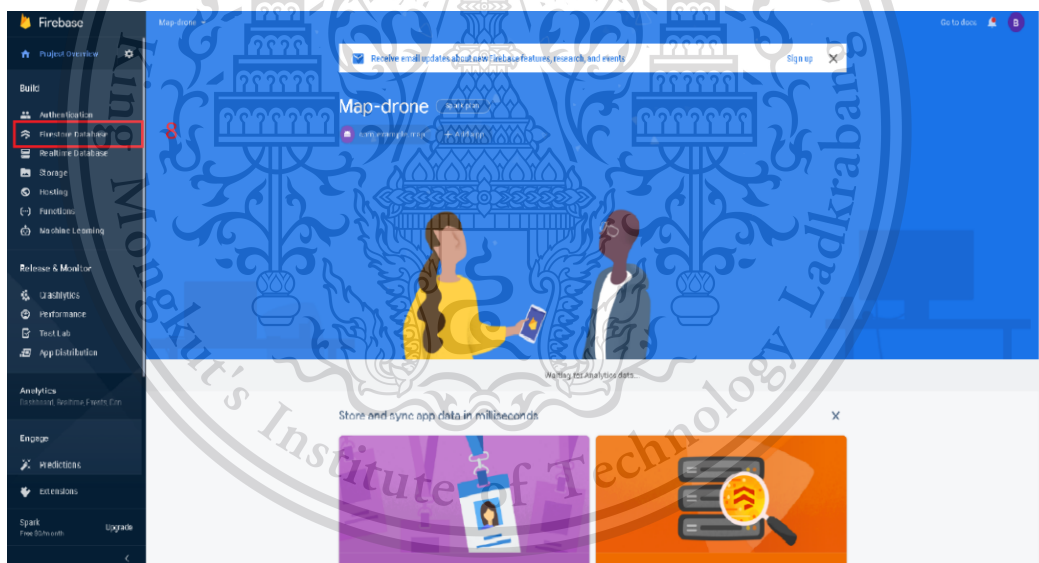
Forbidden to modify the content, and cite the document when use.

ทำตามขั้นตอนต่างๆ ที่เว็บไซต์แนะนำจนกระทั่งสามารถเชื่อมกับแอปพลิเคชันได้



รูปที่ 3.55 หน้าต่างหน้าเชื่อมกับแอปพลิเคชัน

เลือกหมายเลข 3 ใช้ฟังก์ชัน Firestore Database เพื่อเก็บข้อมูลที่ถูกส่งขึ้นมา



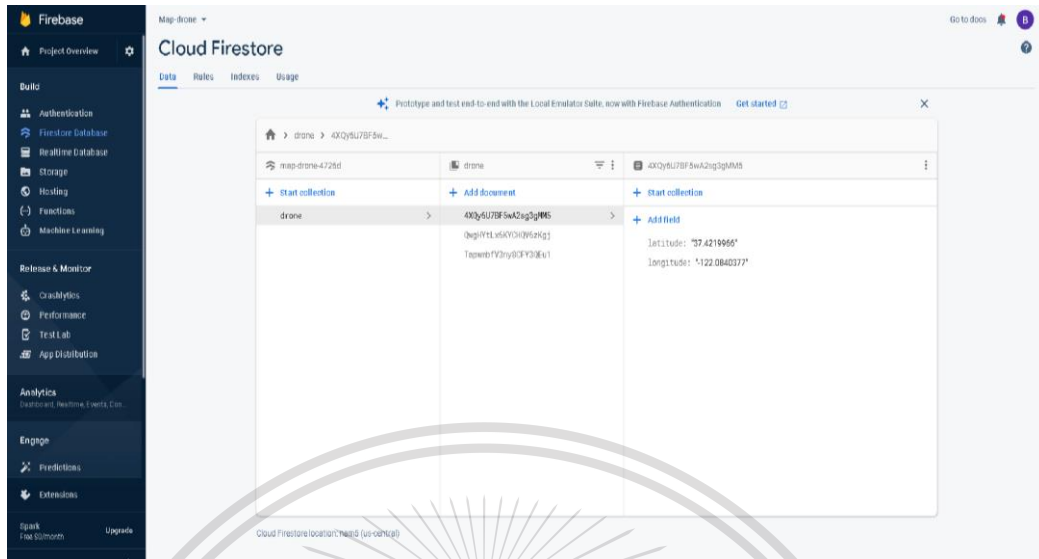
รูปที่ 3.56 หน้าต่างเลือกฟังก์ชันการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

หน้าต่างแสดงข้อมูลที่ถูกส่งขึ้นมาจากแอปพลิเคชัน



รูปที่ 3.57 หน้าต่างแสดงค่าที่ถูกส่งขึ้นมาจากแอปพลิเคชัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

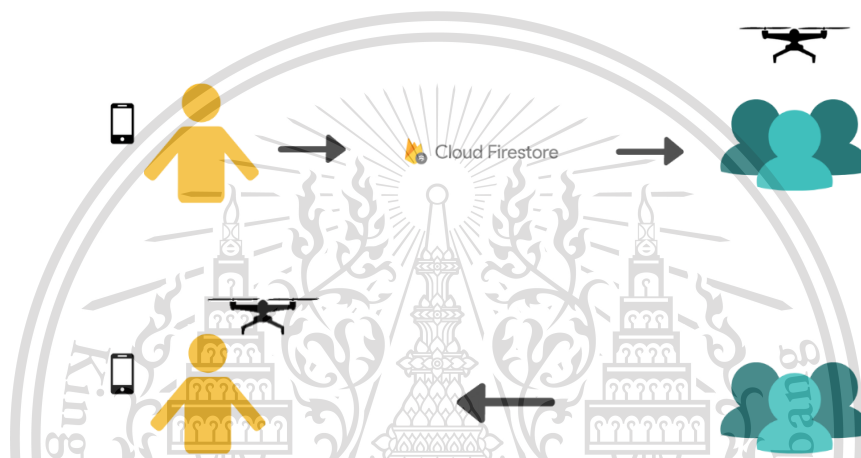
Forbidden to modify the content, and cite the document when use.

บทที่ 4

ผลการทดลอง

4.1 แผนผังการทำงานของโดรนส่งอุปกรณ์ทางการแพทย์

ผู้ที่ต้องการเรียกโดรนจะใช้แอปพลิเคชันส่งพิกัดไปยังผู้ควบคุมโดรน พิกัดที่ผู้เรียกจะถูกส่งไปในเซิร์ฟเวอร์ จากนั้นผู้ควบคุมโดรนจะนำพิกัดบ่อนลงบอร์ด STM32F103C8T6 ที่ควบคุมโดรนอยู่ จากนั้นเปลี่ยนโหมดเป็นโหมดติดตาม GPS และโดรนจะบินไปหาผู้ที่เรียกโดรน



รูปที่ 4.1 แผนผังการทำงานของโดรนส่งอุปกรณ์ทางการแพทย์

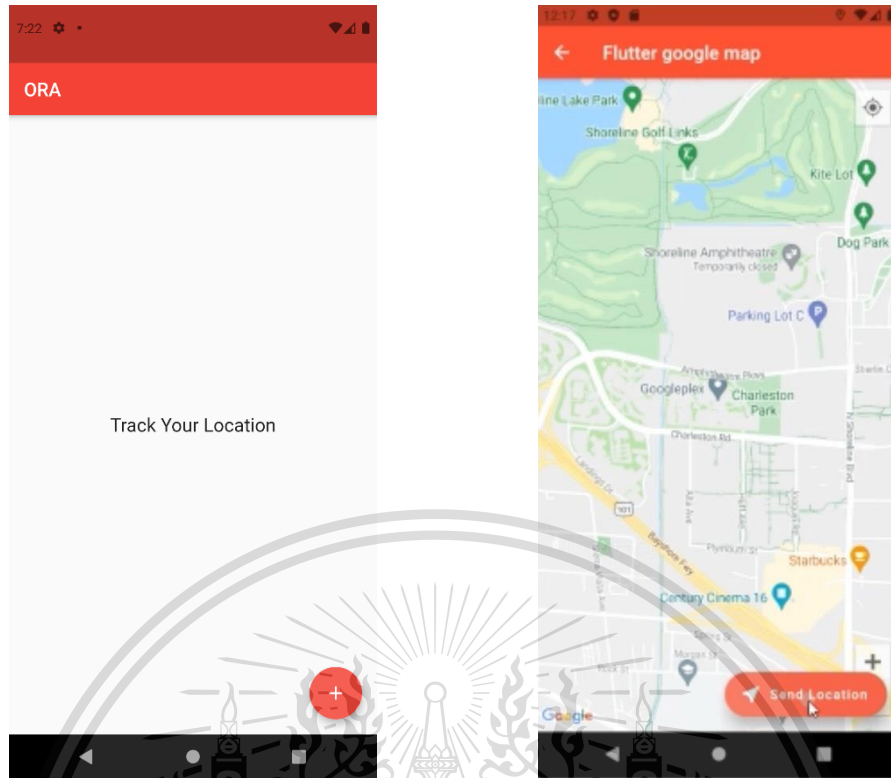
4.2 ผลการออกแบบหน้าจอแสดงผลผ่านแอปพลิเคชัน

เมื่อเปิดแอปพลิเคชันขึ้นมาจะเป็นดังรูป ก. จะมีการบอกแนะนำการใช้งานโดยถ้ากดปุ่มด้านล่างขวาจะหน้าที่ระบุพิกัดตำแหน่งดังรูป ข. จะมีการแสดงแผนที่ Google Map ขึ้นมาและมีปุ่มขวาล่างเพื่อกดส่งตำแหน่งพิกัดไปยังผู้ควบคุมโดรน ดังรูปที่ รูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



รูป ก

รูป ข

รูปที่ 4.2 ผลการออกแบบหน้าจอแสดงผลผ่านแอปพลิเคชัน

4.3 ผลการประดิษฐ์โดรนส่งอุปกรณ์ทางการแพทย์

ทำการติดตั้งเซ็นเซอร์เข้ากับไมโครคอนโทรลเลอร์ ภายในตัวโดรนมีเซ็นเซอร์ต่างๆที่ใช้ในการควบคุม ติดตั้งมอเตอร์ ใบพัด ตัวรับสัญญาณ และแบตเตอรี่ลิเธียมโพลีเมอร์ ดังแสดงในรูปที่ 4.3



รูปที่ 4.3 โดรนที่ติดตั้งเซ็นเซอร์สำเร็จ

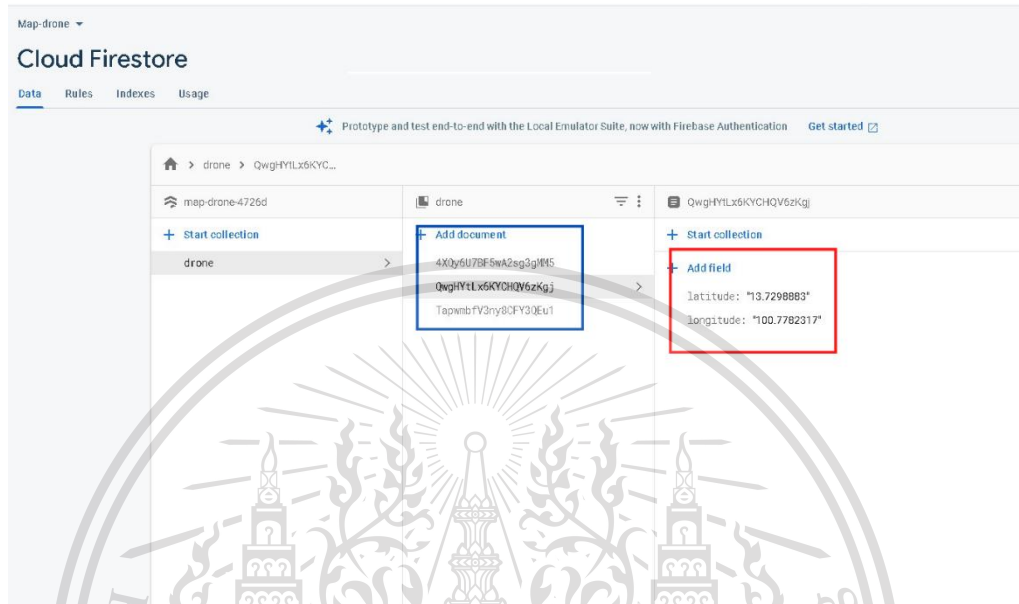
เอกสารนี้เป็นเอกสารที่สงวนไว้เพื่อใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และข้อมูลของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

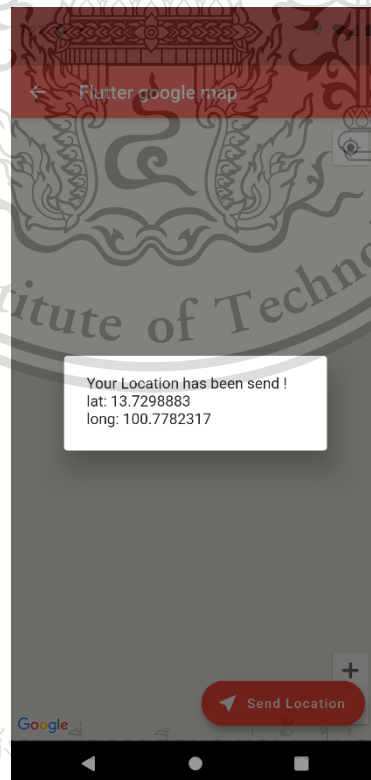
Forbidden to modify the content, and cite the document when use.

4.4 การทดสอบการส่งค่าตำแหน่งปัจจุบันระหว่างแอปพลิเคชันและอุปกรณ์รับ

เมื่อกดปุ่ม Send Location ค่าตำแหน่งพิกัดจะถูกส่งไปยังเซิร์ฟเวอร์ ดังรูป 4.4 และจะปรากฏบนหน้าจอแสดงผลว่าการส่งสำเร็จ ดังรูป 4.5



รูปที่ 4.4 เซิร์ฟเวอร์ที่ค่าตำแหน่งพิกัดส่งไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้... อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดรูปที่ 4.5 หน้าจอแสดงผลการส่งพิกัดสำเร็จเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

ผลการดำเนินงานตามขั้นตอนที่ได้กล่าวมาข้างต้นสามารถแบ่งออกเป็น 3 ส่วน คือ

1. ส่วนฮาร์ดแวร์ คณะผู้จัดทำได้ศึกษาหลักการการทำงานของโดรน อุปกรณ์อิเล็กทรอนิกส์ต่างๆที่นำมาสร้างโดรน รวมถึงการออกแบบชิ้นงานให้สอดคล้องกับวัตถุประสงค์ให้มากที่สุด
2. ส่วนซอฟต์แวร์ คณะผู้จัดทำได้เขียนโปรแกรมเพื่อให้อุปกรณ์ต่างๆทำงานร่วมกันได้โดยมี STM32 เป็นคอนโทรลเลอร์ในการควบคุมอุปกรณ์ต่างๆ ด้วยโปรแกรม Arduino IDE และทำการสร้างแอปพลิเคชันบนสมาร์ตโฟน เพื่อใช้ในการติดต่อสื่อสารระหว่างผู้ใช้กับโดรนเพื่อให้โดรนสามารถรู้จุดหมายปลายทางในการบินได้
3. ส่วนแอปพลิเคชัน เมื่อผู้ใช้ส่งตำแหน่งผ่านแอปพลิเคชันไปยังอุปกรณ์รับพบว่าสามารถส่งค่าตำแหน่งได้จริง และสามารถนำค่าที่ได้รับไปป้อนให้โดรนเพื่อบินไปยังตำแหน่งที่ต้องการได้

5.2 ปัญหาและอุปสรรคในการดำเนินงาน

1. โครงการนี้มีความรู้ และประสบการณ์ใหม่ๆ ทางคณะผู้จัดทำไม่ได้ศึกษาในชั้นเรียนมาก่อน จึงจำเป็นต้องใช้เวลาในการศึกษาหาความรู้เพิ่มเติมเพื่อที่จะนำมาใช้ในโครงการนี้
2. ในการประกอบอุปกรณ์โดรน จำเป็นต้องใช้ทักษะในการบัดกรี และเชื่อมโยงสาย การวางแผน และปฏิบัติจริงจึงเกิดความล่าช้า และใช้เวลามากกว่าที่ได้วางแผนไว้
3. คณะผู้จัดทำยังขาดความชำนาญในการเขียนโปรแกรมภาษา Flutter และ C++ ซึ่งจำเป็นต้องใช้เวลาในการศึกษารูปแบบการเขียนโปรแกรมเพิ่มเติมในช่วงแรก และยังมีข้อบกพร่องในความละเอียด อาจมีการตกหล่นของตัวอักษรในระหว่างการเขียนโปรแกรม ทำให้การประมวลผลในโปรแกรมไม่ผ่าน และใช้เวลานานในการแก้ไขจุดบกพร่อง
4. โครงการนี้มีการใช้อุปกรณ์จากต่างประเทศหลายชิ้น ซึ่งใช้เวลาขนส่งเป็นเวลานาน และมีการเกิดโรคระบาดโควิด-19 ทำให้มีการขนส่งล่าช้ามากกว่าเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

5. เนื่องจากเกิดโรคระบาดโควิด-19 ในประเทศไทย ส่งผลให้มีมาตรการการป้องกันโรคระบาด ของสถาบันฯ ทำให้ไม่สามารถเดินทางเข้าสถาบันฯ ได้ชั่วคราวจนเกิดความล่าช้าของการดำเนินการ

5.3 แนวทางการแก้ไขปัญหา

1. ศึกษาและสอบถามแนวทางเพิ่มเติมจากสื่อการสอนต่างๆ รวมถึงอาจารย์ที่ปรึกษา
2. วางแผนโครงสร้างของด้านฮาร์ดแวร์ทั้งหมดก่อนทำการประกอบ
3. ศึกษาการเขียนโปรแกรมและสอบถามจากผู้ที่มีความชำนาญในการเขียนโปรแกรม
4. จัดเตรียมแผนสำรองเมื่อเกิดเหตุการณ์ไม่คาดคิด เช่น การเกิดโรคระบาด เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

บรรณานุกรม

- [1] DJI Store. “DJI Mini 2 Quadcopter”. [Online]. เข้าถึงได้จาก : <https://www.amazon.com/DJI-Mini-UltralightQuadcopter-Transmission/dp/B08JGYF5W1>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [2] นางสาว มินตรา จิตตคุตตานนท์ นาย ศุภกร หลายชูไทย และผศ.ดร.ฐิติพงษ์ สติรเมธิกุล. “การลอยตัวอยู่กับที่ของโดรน”. [Online]. เข้าถึงได้จาก : https://www.eng.kps.ku.ac.th/dblibv2/fileupload/project_IdDoc187_IdPro573.pdf. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [3] ผศ.ดร.วีระชัย มาलयวเซ. 2558. “ลักษณะของมอเตอร์ไฟฟ้ากระแสตรงไร้แปรงถ่าน”. [Online]. เข้าถึงได้จาก : <http://www.mut.ac.th/research-detail-59>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [4] ptantiku. 2553. “การผสมกันระหว่างสัญญาณรูปสามเหลี่ยมกับสัญญาณไซน์”. [Online]. เข้าถึงได้จาก : <https://blog.anidear.com/2010/12/plasma-speaker.html>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [5] รศ. ณรงค์ บวบทอง. “การสื่อสารแบบอนุกรม”. [Online]. เข้าถึงได้จาก : http://narong.ece.engr.tu.ac.th/micro/document/avr/10-serial_57.pdf. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [6] นิรนาม. “การเชื่อมต่ออุปกรณ์แบบ I²C”. [Online]. เข้าถึงได้จาก : <http://www.thaimicrotron.com/CCS-628/Referrence/I2CBUS.htm>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [7] นิรนาม. “Discharge Voltage Curve ของแบตเตอรี่ลิเธียมโพลิเมอร์”. [Online]. เข้าถึงได้จาก : https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.alibaba.com%2Fproduct-detail%2F25C-35C-45C-60C-Lipo-Battery_60716540255.html&psig=AOvVaw1bXZa26WDgtoavsCsEuQIA&ust=1620322586624000&source=images&cd=vfe&ved=0CAIQjRxqFwoTCMCE6JOKs_ACFQAAAAAdAAAAABAr. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [8] นิรนาม. “Block Diagram ของ plant ที่ใช้ตัวควบคุมร่วมด้วย”. [Online]. เข้าถึงได้จาก : <https://ctms.engin.umich.edu/CTMS/index.php?example=Introduction§ion=ControlPID>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [9] นิรนาม. “ผลตอบสนองเมื่อมีการปรับค่า K_p”. [Online]. เข้าถึงได้จาก : http://www.research-system.siam.edu/images/EE/projectEE/wipavan1/07_ch2.pdf. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้แก้ไขได้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

บรรณานุกรม (ต่อ)

- [10] Swagatam Das. 2552. “ลักษณะการทำงานของ PID Controller”. [Online]. เข้าถึงได้จาก : https://www.researchgate.net/figure/A-generic-closed-loop-process-control-system-with-PID-controller_fig1_242727578. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [11] นิรนาม. “บอร์ดพัฒนา STM32F103C8”. [Online]. เข้าถึงได้จาก : <https://www.robot-siam.com/product/237/stm32f103c8t6-development-board-minimum-stm32>. 2564. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [12] นิรนาม. “Brushless Motor Dcx C2826”. [Online]. เข้าถึงได้จาก : <https://shopee.co.th/Bestprice%E0%B8%A1%E0%B8%AD%E0%B9%80%E0%B8%95%E0%B8%AD%E0%B8%A3%E0%B>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [13] นิรนาม. “แบตเตอรี่ลิเธียมโพลี”. [Online]. เข้าถึงได้จาก: <https://th.aliexpress.com/w/wholesale-lipo-battery-11.1v-5000mah.html>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [14] นิรนาม. “Frame F450 450mm”. [Online]. เข้าถึงได้จาก : <https://shopee.co.th/F-450-%E0%B9%82%E0%B8%94%E0%B8%A3%E0%B8%99%E0%B8%9A%E0%B8%B1%E0%B8%87%E0%B8%84%E0%B8%B1%E0%B8%9A-450-Frame-%E0%B8%AA%E0%B8%B3%E0%B8%AB%E0%B8%A3%E0%B8%B1%E0%B8%9A-RC-MK-MWC-4-Axis-RC-i.86940021.1682624911>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [15] นิรนาม. “Electronic Speed Controller”. [Online]. เข้าถึงได้จาก : <https://sea.banggood.com/th/2-or-4pcs-RC-Brushless-ESC-80A-UBEC-2S-6S-Electronic-Speed-Controller->. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [16] นิรนาม. “บอร์ด PCB”. [Online]. เข้าถึงได้จาก : <https://www.lazada.co.th/products/prototype-pcb-board-5x7-cm-i499994931.html>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [17] นิรนาม. “MC6C 2.4g 6 channel remote control RC Transmitter and Receiver”. [Online]. เข้าถึงได้จาก : <https://th.aliexpress.com/item/32919840132.html>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [18] นิรนาม. “Ublox NEO-M8N”. [Online]. เข้าถึงได้จาก : <https://www.arduitronics.com/product/747/ublox-neo-m8n-gps-module-free-antenna-uart-interface-2>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

บรรณานุกรม (ต่อ)

- [19] นิรนาม. “MPU6050 Module”. [Online]. เข้าถึงได้จาก : <https://www.arduitronics.com/product/481/3-axis-accelerometer-gyro-module-mpu6050-or-gy-521-2>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [20] นิรนาม. “MS5611 Module”. [Online]. เข้าถึงได้จาก : <https://www.techonicsltd.com/product/ms5611-high-resolution-atmospheric-pressure-module/>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [21] นิรนาม. “HMC5883L Module”. [Online]. เข้าถึงได้จาก : <https://www.arduitronics.com/product/50/3-axis-digital-compass-ic-hmc5883l-gy-271>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [22] นิรนาม. “FTDI FT232RL USB to TTL Serial Adapter”. [Online]. เข้าถึงได้จาก : <https://www.arduitronics.com/product/223/ftdi-basic-ft232rl-mini-pro-downloader-usb-to-ttl-rs232-5v-3-3v>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [23] ddos. “Flutter”. [Online]. เข้าถึงได้จาก : <https://meterpreter.org/flutter-release/>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [24] Android Studio. 2561. “Android Studio”. [Online]. เข้าถึงได้จาก : <https://www.behance.net/gallery/72465217/Android-Studio>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [25] codebee. 2560. “Firebase”. [Online]. เข้าถึงได้จาก : <https://www.codebee.co.th/labs/firebase%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3%E0%B9%80%E0%B8%A3%E0%B8%B4%E0%B9%88%E0%B8%A1%E0%B8%95%E0%B9%89%E0%B8%99%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99/>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [26] กุลธิดา เต็มวิทยานันท์. 2559. ““Drone” เทคโนโลยีอากาศยานไร้คนขับ”. [Online]. เข้าถึงได้จาก : <https://www.pwc.com/th/en/pwc-thailand-blogs/blog-20161031.html>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [27] Timothy Hirzel. 2561. “PWM”. [Online]. เข้าถึงได้จาก : <https://www.arduino.cc/en/Tutorial/Foundations/PWM>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [28] JordanDee. “Pulse Width Modulation”. [Online]. เข้าถึงได้จาก : <https://learn.sparkfun.com/tutorials/pulse-width-modulation>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ทำซ้ำโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น ผู้ใช้ที่มีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

บรรณานุกรม (ต่อ)

- [29] เฉลิมพล เรื่องพัฒนาวิวัฒน์ และยุทธนา กันทะพะเยา. 2559. “การควบคุมมอเตอร์ไฟฟ้า กระแสตรงไร้แปรงถ่านด้วยไมโครคอนโทรลเลอร์สำหรับจักรยานสามล้อไฟฟ้า”. [Online]. เข้าถึงได้จาก : <http://engineering.rmutl.ac.th/journal/files/RMUTLEngJ-1-03.pdf>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [30] มนตรีศิริปรัชญานันท และปราโมทย์वादเขียน. 2544. “การกำเนิดสัญญาณผลอนุพันธ์ PWM ที่มีความถี่คงที่โดยใช้วงจรขมิตริกเกอร์แบบใหม่”. [Online]. เข้าถึงได้จาก : http://www.te.kmutnb.ac.th/~msn/PWM_EECON24.pdf. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [31] Tarun Agarwal. “What is Electronic Speed Control (ESC) & Its Working”. [Online]. เข้าถึงได้จาก : <https://www.elprocus.com/electronic-speed-control-esc-working-applications/>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [32] Thaimicrotron. “การเชื่อมต่ออุปกรณ์แบบ I²C”. [Online]. เข้าถึงได้จาก : <http://www.thaimicrotron.com/CCS-628/Reference/I2CBUS.htm>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [33] นิรนาม. 2563. “การสื่อสารแบบ I²C”. [Online]. เข้าถึงได้จาก : <http://fitrox.lnwshop.com/article/81/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B8%AA%E0%B8%B7%E0%B9%88%E0%B8%AD%E0%B8%AA%E0%B8%B2%E0%B8%A3%E0%B9%81%E0%B8%9A%E0%B8%9A-i2c>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [34] Support ThaiEasyElec. 2563. “บทความ ESPino32 ตอนที่ 8 การสื่อสารอนุกรมแบบ I²C”. [Online]. เข้าถึงได้จาก : <https://blog.thaieasyelec.com/espino32-ch8-how-to-use-i2c/>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [35] นิรนาม. 2563. “การใช้งาน/ความรู้แบตเตอรี่ลิเทียม-โพลีเมอร์ (Lithium-Polymer : Li-Po)”. [Online]. เข้าถึงได้จาก : <http://www.sirensshop89.com/webboard/viewtopic/48>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [36] นิรนาม. “Ublox NEO-M8N GPS Module + Free Antenna(UART Interface)”. [Online]. เข้าถึงได้จาก : arduitronics.com/product/747/ublox-neo-m8n-gps-module-free-antenna-uart-interface-2. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [37] keerati rounghirun. 2562. “[Engineering] PID Controller PART 1”. [Online]. เข้าถึงได้จาก : <https://medium.com/maestro19/engineering-pid-controller-part-1-33a8f6d27fa8>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษาเท่านั้น ไม่เอามาทำเป็นงานที่ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น สืบค้นเมื่อวันที่ 20 เมษายน พ.ศ. 2564. และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

บรรณานุกรม (ต่อ)

- [38] วิกีพีเดีย สารานุกรมเสรี. “ระบบควบคุมพีไอดี”. [Online]. เข้าถึงได้จาก : <https://th.wikipedia.org/wiki/%E0%B8%A3%E0%B8%B0%E0%B8%9A%E0%B8%9A%E0%B8%84%E0%B8%A7%E0%B8%9A%E0%B8%84%E0%B8%B8%E0%B8%A1%E0%B8%9E%E0%B8%B5%E0%B9%84%E0%B8%AD%E0%B8%94%E0%B8%B5>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [39] วิกีพีเดีย สารานุกรมเสรี. “PID controller”. [Online]. เข้าถึงได้จาก : https://en.wikipedia.org/wiki/PID_controller. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [40] Dotx Control Solutions. “The PID Controller”. [Online]. เข้าถึงได้จาก : <https://www.pid-tuner.com/pid-control/>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [41] นิรนาม. 2560. “ระบบกำหนดตำแหน่งบนโลก GPS”. [Online]. เข้าถึงได้จาก : http://119.46.166.126/self_all/selfaccess8/m2/186/lesson1/web10/web10.php. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [42] นิรนาม. 2559. “GPS vs GNSS”. [Online]. เข้าถึงได้จาก : <http://www.gnsthailand.com/gps-vs-gnss/>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [43] global5thailand. “ความรู้ทั่วไปเกี่ยวกับ GPS”. [Online]. เข้าถึงได้จาก : <https://www.global5thailand.thai/gps.htm>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [44] นิรนาม. “NMEA GLL Message”. [Online]. เข้าถึงได้จาก : http://manuals.spectra.com.com/VSP/Content/VSP/NMEA_GLLmess.htm. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [45] นิรนาม. “NMEA-0183 message: GGA”. [Online]. เข้าถึงได้จาก : https://www.trimble.com/OEM_ReceiverHelp/V4.44/en/NMEA-0183messages_GGA.html. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [46] นิรนาม. “Project YMFC-32 - The STM32 quadcopter”. [Online]. เข้าถึงได้จาก : http://www.brokking.net/ymfc-32_main.html. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.
- [47] Wichet Darakai. 2554. “GPS : โปรโตคอล NMEA ของ GPS”. [Online]. เข้าถึงได้จาก : <https://darakai.blogspot.com/2011/11/nmea-gps.html>. สืบค้นเมื่อ 20 เมษายน พ.ศ. 2564.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

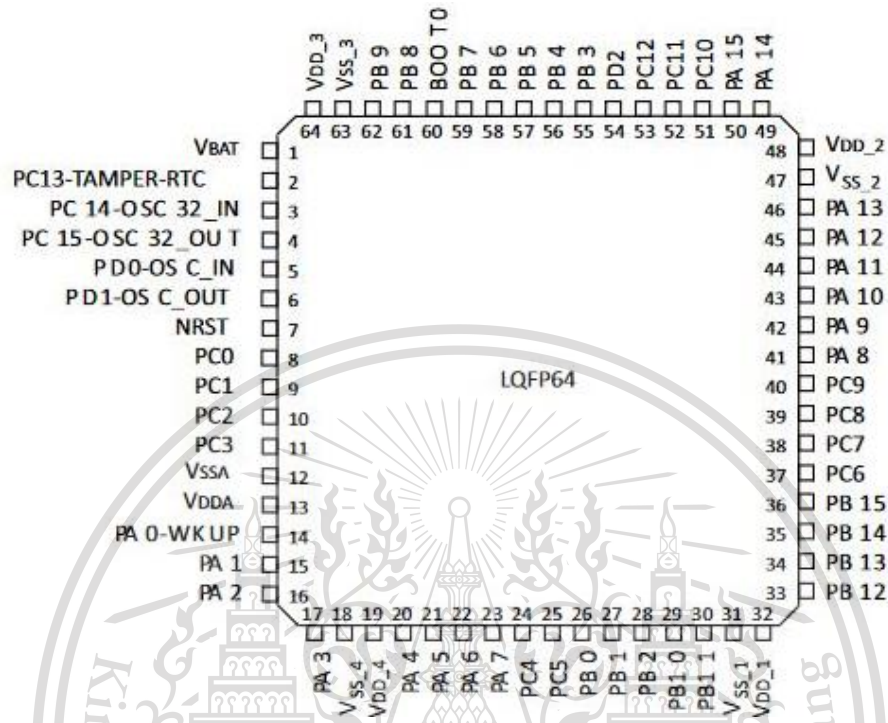


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ภาคผนวก ก
บอร์ด STM32F103C8T6



รูปที่ ก.1 คุณสมบัติทั่วไปของ บอร์ด STM32F103C8T6 (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

| Pins | | | | | | | Pin name | Type ⁽¹⁾ I/O Level ⁽²⁾ | Main function ⁽³⁾ (after reset) | Alternate functions ⁽⁴⁾ | |
|----------|---------|----------------|-------------------|--------|---------|---------|--------------------------------|---|---|------------------------------------|--------------------|
| LFBG4100 | UFBG100 | LQFP48/UQFPN48 | TFBGA64 | LQFP64 | LQFP100 | VQFPN36 | | | | Default | Remap |
| A3 | B2 | - | - | - | 1 | - | PE2 | I/O FT | PE2 | TRACECK | - |
| B3 | A1 | - | - | - | 2 | - | PE3 | I/O FT | PE3 | TRACED0 | - |
| C3 | B1 | - | - | - | 3 | - | PE4 | I/O FT | PE4 | TRACED1 | - |
| D3 | C2 | - | - | - | 4 | - | PE5 | I/O FT | PE5 | TRACED2 | - |
| E3 | D2 | - | - | - | 5 | - | PE6 | I/O FT | PE6 | TRACED3 | - |
| B2 | E2 | 1 | B2 | 1 | 6 | - | V _{BAT} | S - | V _{BAT} | - | - |
| A2 | C1 | 2 | A2 | 2 | 7 | - | PC13-TAMPER-RTC ⁽⁵⁾ | I/O - | PC13 ⁽⁶⁾ | TAMPER-RTC | - |
| A1 | D1 | 3 | A1 | 3 | 8 | - | PC14-OSC32_IN ⁽⁵⁾ | I/O - | PC14 ⁽⁶⁾ | OSC32_IN | - |
| B1 | E1 | 4 | B1 | 4 | 9 | - | PC15-OSC32_OUT ⁽⁵⁾ | I/O - | PC15 ⁽⁶⁾ | OSC32_OUT | - |
| C2 | F2 | - | - | - | 10 | - | V _{SS_S} | S - | V _{SS_S} | - | - |
| D2 | G2 | - | - | - | 11 | - | V _{DD_S} | S - | V _{DD_S} | - | - |
| C1 | F1 | 5 | C1 | 5 | 12 | 2 | OSC_IN | I - | OSC_IN | - | PD0 ⁽⁷⁾ |
| D1 | G1 | 6 | D1 | 6 | 13 | 3 | OSC_OUT | O - | OSC_OUT | - | PD1 ⁽⁷⁾ |
| E1 | H2 | 7 | E1 | 7 | 14 | 4 | NRST | I/O - | NRST | - | - |
| F1 | H1 | - | E3 | 8 | 15 | - | PC0 | I/O - | PC0 | ADC12_IN10 | - |
| F2 | J2 | - | E2 | 9 | 16 | - | PC1 | I/O - | PC1 | ADC12_IN11 | - |
| E2 | J3 | - | F2 | 10 | 17 | - | PC2 | I/O - | PC2 | ADC12_IN12 | - |
| F3 | K2 | - | - ⁽⁸⁾ | 11 | 18 | - | PC3 | I/O - | PC3 | ADC12_IN13 | - |
| G1 | J1 | 8 | F1 | 12 | 19 | 5 | V _{SSA} | S - | V _{SSA} | - | - |
| H1 | K1 | - | - | - | 20 | - | V _{REF} | S - | V _{REF} | - | - |
| J1 | L1 | - | G1 ⁽⁸⁾ | - | 21 | - | V _{REF-} | S - | V _{REF-} | - | - |
| K1 | M1 | 9 | H1 | 13 | 22 | 6 | V _{DDA} | S - | V _{DDA} | - | - |

รูปที่ ก.2 คุณสมบัติทั่วไปของ บอร์ด STM32F103C8T6 (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

| Pins | | | | | | | Pin name | Type ⁽¹⁾ | I/O Level ⁽²⁾ | Main function ⁽³⁾ (after reset) | Alternate functions ⁽⁴⁾ | |
|----------|---------|----------------|---------|--------|---------|---------|-------------------|---------------------|--------------------------|---|---|-----------|
| LFPGA100 | UFBG100 | LQFP48/UQFPN48 | TFBGA64 | LQFP64 | LQFP100 | VQFPN36 | | | | | Default | Remap |
| G2 | L2 | 10 | G2 | 14 | 23 | 7 | PA0-WKUP | I/O | - | PA0 | WKUP/ USART2_CTS ⁽⁵⁾ / ADC12_IN0/ TIM2_CH1- ETR ⁽⁵⁾ | - |
| H2 | M2 | 11 | H2 | 15 | 24 | 8 | PA1 | I/O | - | PA1 | USART2_RTS ⁽⁵⁾ / ADC12_IN1/ TIM2_CH2 ⁽⁵⁾ | - |
| J2 | K3 | 12 | F3 | 16 | 25 | 9 | PA2 | I/O | - | PA2 | USART2_TX ⁽⁵⁾ / ADC12_IN2/ TIM2_CH3 ⁽⁵⁾ | - |
| K2 | L3 | 13 | G3 | 17 | 26 | 10 | PA3 | I/O | - | PA3 | USART2_RX ⁽⁵⁾ / ADC12_IN3/ TIM2_CH4 ⁽⁵⁾ | - |
| E4 | E3 | - | C2 | 18 | 27 | - | V _{SS_4} | S | - | V _{SS_4} | - | - |
| F4 | H3 | - | D2 | 19 | 28 | - | V _{DD_4} | S | - | V _{DD_4} | - | - |
| G3 | M3 | 14 | H3 | 20 | 29 | 11 | PA4 | I/O | - | PA4 | SPI1_NSS ⁽⁵⁾ / USART2_CK ⁽⁵⁾ / ADC12_IN4 | - |
| H3 | K4 | 15 | F4 | 21 | 30 | 12 | PA5 | I/O | - | PA5 | SPI1_SCK ⁽⁵⁾ / ADC12_IN5 | - |
| J3 | L4 | 16 | G4 | 22 | 31 | 13 | PA6 | I/O | - | PA6 | SPI1_MISO ⁽⁵⁾ / ADC12_IN6/ TIM3_CH1 ⁽⁵⁾ | TIM1_BKIN |
| K3 | M4 | 17 | H4 | 23 | 32 | 14 | PA7 | I/O | - | PA7 | SPI1_MOSI ⁽⁵⁾ / ADC12_IN7/ TIM3_CH2 ⁽⁵⁾ | TIM1_CH1N |
| G4 | K5 | - | H5 | 24 | 33 | - | PC4 | I/O | - | PC4 | ADC12_IN14 | - |
| H4 | L5 | - | H6 | 25 | 34 | - | PC5 | I/O | - | PC5 | ADC12_IN15 | - |
| J4 | M5 | 18 | F5 | 26 | 35 | 15 | PB0 | I/O | - | PB0 | ADC12_IN8/ TIM3_CH3 ⁽⁵⁾ | TIM1_CH2N |
| K4 | M6 | 19 | G5 | 27 | 36 | 16 | PB1 | I/O | - | PB1 | ADC12_IN9/ TIM3_CH4 ⁽⁵⁾ | TIM1_CH3N |

รูปที่ ก.3 คุณสมบัติทั่วไปของ บอร์ด STM32F103C8T6 (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

| Pins | | | | | | | Pin name | Type ⁽¹⁾ | I/O Level ⁽²⁾ | Main function ⁽³⁾ (after reset) | Alternate functions ⁽⁴⁾ | |
|----------|--------|----------------|---------|--------|---------|---------|----------|---------------------|--------------------------|--|------------------------------------|-------|
| LFPGA100 | UFG100 | LQFP48/UQFPN48 | TFPGA64 | LQFP64 | LQFP100 | VQFPN35 | | | | | Default | Remap |
| G5 | L6 | 20 | G6 | 28 | 37 | 17 | PB2 | I/O FT | PB2/BOOT1 | - | - | |
| H5 | M7 | - | - | - | 38 | - | PE7 | I/O FT | PE7 | - | TIM1_ETR | |
| J5 | L7 | - | - | - | 39 | - | PE8 | I/O FT | PE8 | - | TIM1_CH1N | |
| K5 | M8 | - | - | - | 40 | - | PE9 | I/O FT | PE9 | - | TIM1_CH1 | |
| G6 | L8 | - | - | - | 41 | - | PE10 | I/O FT | PE10 | - | TIM1_CH2N | |
| H6 | M9 | - | - | - | 42 | - | PE11 | I/O FT | PE11 | - | TIM1_CH2 | |
| J6 | L9 | - | - | - | 43 | - | PE12 | I/O FT | PE12 | - | TIM1_CH3N | |
| K6 | M10 | - | - | - | 44 | - | PE13 | I/O FT | PE13 | - | TIM1_CH3 | |
| G7 | M11 | - | - | - | 45 | - | PE14 | I/O FT | PE14 | - | TIM1_CH4 | |
| H7 | M12 | - | - | - | 46 | - | PE15 | I/O FT | PE15 | - | TIM1_BKIN | |
| J7 | L10 | 21 | G7 | 29 | 47 | - | PB10 | I/O FT | PB10 | I2C2_SCL/ USART3_TX ⁽⁵⁾ | TIM2_CH3 | |
| K7 | L11 | 22 | H7 | 30 | 48 | - | PB11 | I/O FT | PB11 | I2C2_SDA/ USART3_RX ⁽⁵⁾ | TIM2_CH4 | |
| E7 | F12 | 23 | D6 | 31 | 49 | 18 | Vss_1 | S | Vss_1 | - | - | |
| F7 | G12 | 24 | E6 | 32 | 50 | 19 | Vdd_1 | S | Vdd_1 | - | - | |
| K8 | L12 | 25 | H8 | 33 | 51 | - | PB12 | I/O FT | PB12 | SPI2_NSS/ I2C2_SMBAL/ USART3_CK ⁽⁵⁾ TIM1_BKIN ⁽⁵⁾ | - | |
| J8 | K12 | 26 | G8 | 34 | 52 | - | PB13 | I/O FT | PB13 | SPI2_SCK/ USART3_CTS ⁽⁵⁾ TIM1_CH1N ⁽⁵⁾ | - | |
| H8 | K11 | 27 | F8 | 35 | 53 | - | PB14 | I/O FT | PB14 | SPI2_MISO/ USART3_RTS ⁽⁵⁾ TIM1_CH2N ⁽⁵⁾ | - | |
| G8 | K10 | 28 | F7 | 36 | 54 | - | PB15 | I/O FT | PB15 | SPI2_MOSI/ TIM1_CH3N ⁽⁵⁾ | - | |
| K9 | K9 | - | - | - | 55 | - | PD8 | I/O FT | PD8 | - | USART3_TX | |
| J9 | K8 | - | - | - | 56 | - | PD9 | I/O FT | PD9 | - | USART3_RX | |

รูปที่ ก.4 คุณสมบัติทั่วไปของ บอร์ด STM32F103C8T6 (4)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

| Pins | | | | | | | Pin name | Type ⁽¹⁾ I/O Level ⁽²⁾ | Main function ⁽³⁾ (after reset) | Alternate functions ⁽⁴⁾ | |
|----------|---------|----------------|---------|--------|---------|---------|----------|---|--|--|--------------------------|
| LFPGA100 | UFBG100 | LQFP48/UQFPN48 | TFBGA64 | LQFP64 | LQFP100 | VQFPN36 | | | | Default | Remap |
| H9 | J12 | - | - | - | 57 | - | PD10 | I/O FT | PD10 | - | USART3_CK |
| G9 | J11 | - | - | - | 58 | - | PD11 | I/O FT | PD11 | - | USART3_CTS |
| K10 | J10 | - | - | - | 59 | - | PD12 | I/O FT | PD12 | - | TIM4_CH1 / USART3_RTS |
| J10 | H12 | - | - | - | 60 | - | PD13 | I/O FT | PD13 | - | TIM4_CH2 |
| H10 | H11 | - | - | - | 61 | - | PD14 | I/O FT | PD14 | - | TIM4_CH3 |
| G10 | H10 | - | - | - | 62 | - | PD15 | I/O FT | PD15 | - | TIM4_CH4 |
| F10 | E12 | - | F6 | 37 | 63 | - | PC5 | I/O FT | PC6 | - | TIM3_CH1 |
| E10 | E11 | - | E7 | 38 | 64 | - | PC7 | I/O FT | PC7 | - | TIM3_CH2 |
| F9 | E10 | - | E8 | 39 | 65 | - | PC8 | I/O FT | PC8 | - | TIM3_CH3 |
| E9 | D12 | - | D8 | 40 | 66 | - | PC9 | I/O FT | PC9 | - | TIM3_CH4 |
| D9 | D11 | 29 | D7 | 41 | 67 | 20 | PA8 | I/O FT | PA8 | USART1_CK/ TIM1_CH1 ⁽⁵⁾ / MCO | - |
| C9 | D10 | 30 | C7 | 42 | 68 | 21 | PA9 | I/O FT | PA9 | USART1_TX ⁽⁵⁾ / TIM1_CH2 ⁽⁵⁾ | - |
| D10 | C12 | 31 | C6 | 43 | 69 | 22 | PA10 | I/O FT | PA10 | USART1_RX ⁽⁵⁾ / TIM1_CH3 ⁽⁵⁾ | - |
| C10 | B12 | 32 | C8 | 44 | 70 | 23 | PA11 | I/O FT | PA11 | USART1_CTS ⁽⁵⁾ / CANRX ⁽⁵⁾ / USBDM/ TIM1_CH4 ⁽⁵⁾ | - |
| B10 | A12 | 33 | B8 | 45 | 71 | 24 | PA12 | I/O FT | PA12 | USART1_RTS/ CANTX ⁽⁵⁾ / USBDM/ TIM1_ETR ⁽⁵⁾ | - |
| A10 | A11 | 34 | A8 | 46 | 72 | 25 | PA13 | I/O FT | JTMS/SWDIO | - | PA13 |
| F8 | C11 | - | - | - | 73 | - | | | Not connected | | - |
| E6 | F11 | 35 | D5 | 47 | 74 | 26 | Vss_2 | S | Vss_2 | - | - |
| F6 | G11 | 36 | E5 | 48 | 75 | 27 | Vcc_2 | S | Vcc_2 | - | - |

รูปที่ ก.5 คุณสมบัติทั่วไปของ บอร์ด STM32F103C8T6 (5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

| Pins | | | | | | | Pin name | Type ⁽¹⁾ I/O Level ⁽²⁾ | Main function ⁽³⁾ (after reset) | Alternate functions ⁽⁴⁾ | |
|----------|---------|-----------------|---------|--------|---------|----------|----------|---|---|--|--------------------------------|
| LFPGA100 | UFBG100 | LQFP48/UFQFPN48 | TFBGA64 | LQFP64 | LQFP100 | VQFPFN35 | | | | Default | Remap |
| A9 | A10 | 37 | A7 | 49 | 76 | 28 | PA14 | I/O FT | JTCK/SWCLK | - | PA14 |
| A8 | A9 | 38 | A6 | 50 | 77 | 29 | PA15 | I/O FT | JTDI | - | TIM2_CH1_ETR/PA15/SPI1_NSS |
| B9 | B11 | - | B7 | 51 | 78 | | PC10 | I/O FT | PC10 | - | USART3_TX |
| B8 | C10 | - | B6 | 52 | 79 | | PC11 | I/O FT | PC11 | - | USART3_RX |
| C8 | B10 | - | C5 | 53 | 80 | | PC12 | I/O FT | PC12 | - | USART3_CK |
| - | C9 | - | C1 | - | 81 | 2 | PD0 | I/O FT | PD0 | - | CANRX |
| - | B9 | - | D1 | - | 82 | 3 | PD1 | I/O FT | PD1 | - | CANTX |
| B7 | C8 | - | B5 | 54 | 83 | - | PD2 | I/O FT | PD2 | TIM3_ETR | - |
| C7 | B8 | - | - | - | 84 | - | PD3 | I/O FT | PD3 | - | USART2_CTS |
| D7 | B7 | - | - | - | 85 | - | PD4 | I/O FT | PD4 | - | USART2_RTS |
| B6 | A6 | - | - | - | 86 | - | PD5 | I/O FT | PD5 | - | USART2_TX |
| C6 | B6 | - | - | - | 87 | - | PD6 | I/O FT | PD6 | - | USART2_RX |
| D6 | A5 | - | - | - | 88 | - | PD7 | I/O FT | PD7 | - | USART2_CK |
| A7 | A8 | 39 | A5 | 55 | 89 | 30 | PB3 | I/O FT | JTDO | - | TIM2_CH2/PB3/TRACESWO/SPI1_SCK |
| A6 | A7 | 40 | A4 | 56 | 90 | 31 | PB4 | I/O FT | JNTRST | - | TIM3_CH1/PB4/SPI1_MISO |
| C5 | C6 | 41 | C4 | 57 | 91 | 32 | PB5 | I/O | PB5 | I2C1_SMBAL | TIM3_CH2/SPI1_MOSI |
| B5 | B5 | 42 | D3 | 58 | 92 | 33 | PB6 | I/O FT | PB6 | I2C1_SCL ⁽⁵⁾ /TIM4_CH1 ⁽⁵⁾ | USART1_TX |
| A5 | B4 | 43 | C3 | 59 | 93 | 34 | PB7 | I/O FT | PB7 | I2C1_SDA ⁽⁵⁾ /TIM4_CH2 ⁽⁵⁾ | USART1_RX |
| D5 | A4 | 44 | B4 | 60 | 94 | 35 | BOOT0 | I/O | BOOT0 | - | - |

รูปที่ ก.6 คุณสมบัติทั่วไปของ บอร์ด STM32F103C8T6 (6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

| Pins | | | | | | | Pin name | Type ⁽¹⁾ | I/O Level ⁽²⁾ | Main function ⁽³⁾ (after reset) | Alternate functions ⁽⁴⁾ | |
|----------|---------|----------------|---------|--------|---------|---------|----------|---------------------|--------------------------|---|------------------------------------|-------|
| LFBGA100 | UFBG100 | LQFP48/UQFPN48 | TFBGA64 | LQFP64 | LQFP100 | VQFPN36 | | | | | Default | Remap |
| B4 | A3 | 45 | B3 | 61 | 95 | - | PB8 | I/O FT | PB8 | TIM4_CH3 ⁽⁵⁾ | I2C1_SCL / CANRX | |
| A4 | B3 | 46 | A3 | 62 | 96 | - | PB9 | I/O FT | PB9 | TIM4_CH4 ⁽⁵⁾ | I2C1_SDA / CANTX | |
| D4 | C3 | - | - | - | 97 | - | PE0 | I/O FT | PE0 | TIM4_ETR | - | |
| C4 | A2 | - | - | - | 98 | - | PE1 | I/O FT | PE1 | - | - | |
| E5 | D3 | 47 | D4 | 63 | 99 | 36 | Vss_3 | S | - | Vss_3 | - | |
| F5 | C4 | 48 | E4 | 64 | 100 | 1 | Vcc_3 | S | - | Vcc_3 | - | |

รูปที่ ก.7 คุณสมบัติทั่วไปของ บอร์ด STM32F103C8T6 (7)



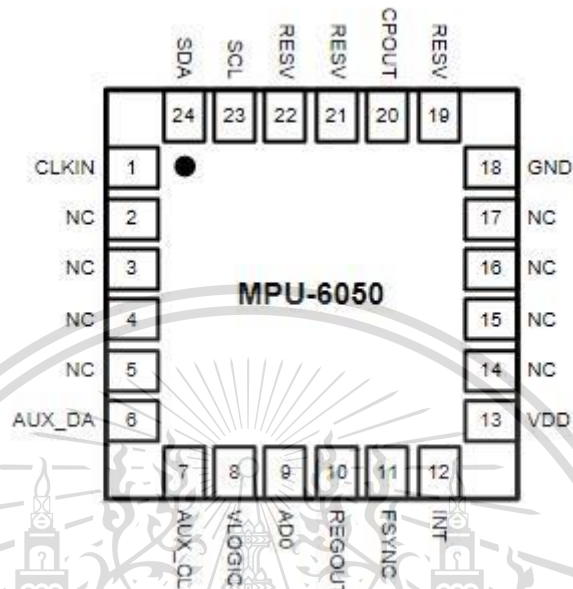
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ภาคผนวก ข

โมดูล MPU-6050



รูปที่ ข.1 คุณสมบัติทั่วไปของ โมดูล MPU-6050 (1)

| Pin Number | MPU-6000 | MPU-6050 | Pin Name | Pin Description |
|----------------------------|----------|----------|------------|--|
| 1 | Y | Y | CLKIN | Optional external reference clock input. Connect to GND if unused. |
| 6 | Y | Y | AUX_DA | I ² C master serial data, for connecting to external sensors |
| 7 | Y | Y | AUX_CL | I ² C Master serial clock, for connecting to external sensors |
| 8 | Y | Y | /CS | SPI chip select (0=SPI mode) |
| 8 | Y | Y | VLOGIC | Digital I/O supply voltage |
| 9 | Y | Y | AD0 / SDO | I ² C Slave Address LSB (AD0); SPI serial data output (SDO) |
| 9 | | Y | AD0 | I ² C Slave Address LSB (AD0) |
| 10 | Y | Y | REGOUT | Regulator filter capacitor connection |
| 11 | Y | Y | FSYNC | Frame synchronization digital input. Connect to GND if unused. |
| 12 | Y | Y | INT | Interrupt digital output (totem pole or open-drain) |
| 13 | Y | Y | VDD | Power supply voltage and Digital I/O supply voltage |
| 18 | Y | Y | GND | Power supply ground |
| 19, 21 | Y | Y | RESV | Reserved. Do not connect. |
| 20 | Y | Y | CPOUT | Charge pump capacitor connection |
| 22 | Y | Y | RESV | Reserved. Do not connect. |
| 23 | Y | Y | SCL / SCLK | I ² C serial clock (SCL); SPI serial clock (SCLK) |
| 23 | | Y | SCL | I ² C serial clock (SCL) |
| 24 | Y | Y | SDA / SDI | I ² C serial data (SDA); SPI serial data input (SDI) |
| 24 | | Y | SDA | I ² C serial data (SDA) |
| 2, 3, 4, 5, 14, 15, 16, 17 | Y | Y | NC | Not internally connected. May be used for PCB trace routing. |

รูปที่ ข.2 คุณสมบัติทั่วไปของ โมดูล MPU-6050 (2)

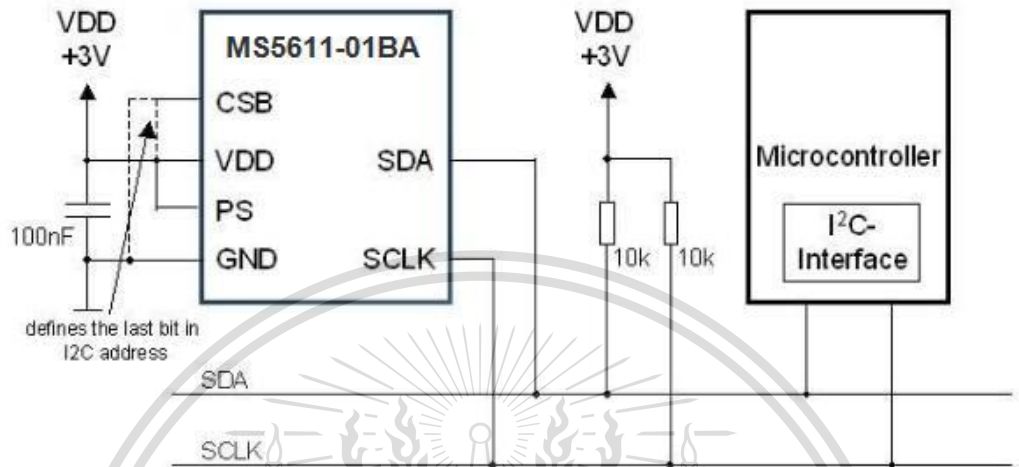
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ภาคผนวก ค

โมดูล MS5611



รูปที่ ค.1 คุณสมบัติทั่วไปของ โมดูล MS5611 (1)

| Pin | Name | Type | Function |
|-----|--------------|--------|---|
| 1 | VDD | P | Positive supply voltage |
| 2 | PS | I | Protocol select PS high (VDD) → I ² C PS low (GND) → SPI |
| 3 | GND | G | Ground |
| 4 | CSB | I | Chip select (active low), internal connection |
| 5 | SDO | O | Serial data output |
| 7 | SDI / SDA | I / IO | Serial data input / I ² C data IO |
| 8 | SCLK | I | Serial data clock |

รูปที่ ค.2 คุณสมบัติทั่วไปของ โมดูล MS5611 (2)

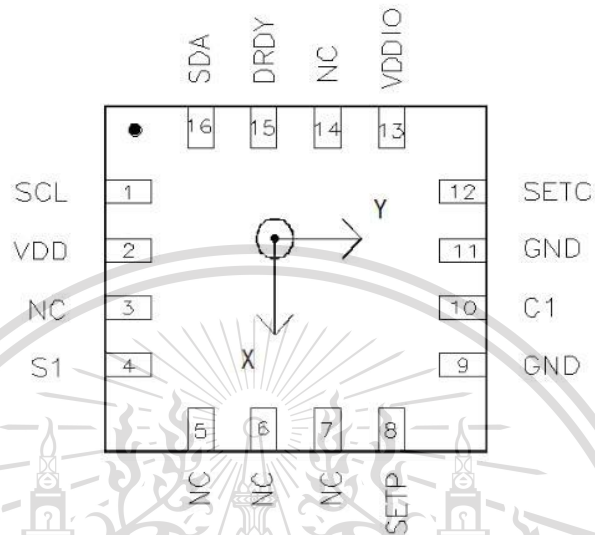
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ภาคผนวก ง

โมดูล HMC5883L



รูปที่ ง.1 คุณสมบัติทั่วไปของ โมดูล HMC5883L (1)

| Pin | Name | Description |
|-----|-------|---|
| 1 | SCL | Serial Clock – I ² C Master/Slave Clock |
| 2 | VDD | Power Supply (2.16V to 3.6V) |
| 3 | NC | Not to be Connected |
| 4 | S1 | Tie to VDDIO |
| 5 | NC | Not to be Connected |
| 6 | NC | Not to be Connected |
| 7 | NC | Not to be Connected |
| 8 | SETP | Set/Reset Strap Positive – S/R Capacitor (C2) Connection |
| 9 | GND | Supply Ground |
| 10 | C1 | Reservoir Capacitor (C1) Connection |
| 11 | GND | Supply Ground |
| 12 | SETC | S/R Capacitor (C2) Connection – Driver Side |
| 13 | VDDIO | IO Power Supply (1.71V to VDD) |
| 14 | NC | Not to be Connected |
| 15 | DRDY | Data Ready, Interrupt Pin. Internally pulled high. Optional connection. Low for 250 μ sec when data is placed in the data output registers. |
| 16 | SDA | Serial Data – I ² C Master/Slave Data |

รูปที่ ง.2 คุณสมบัติทั่วไปของ โมดูล HMC5883L (2)

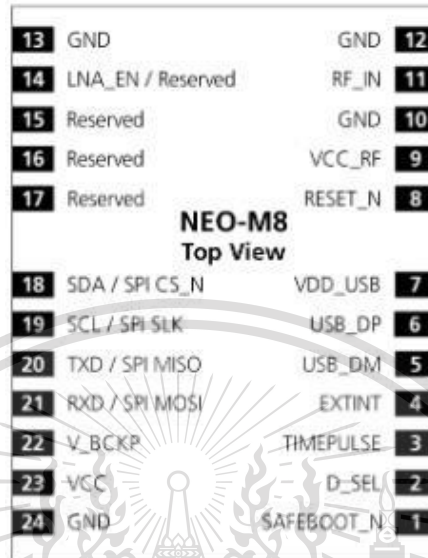
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ภาคผนวก จ

โมดูล NEO-Ublox M8N



รูปที่ จ.1 คุณสมบัติทั่วไปของ โมดูล NEO-Ublox M8N (1)

| No. | Name | PIO no. | I/O | Description |
|-----|----------------------|---------|-----|---|
| 1 | SAFEBOOT_N | - | I | SAFEBOOT_N (for future service, updates and reconfiguration, leave OPEN) |
| 2 | D_SEL | - | I | Interface select |
| 3 | TIMEPULSE | 11 | O | Time pulse (1PPS) |
| 4 | EXTINT | 13 | I | External interrupt pin |
| 5 | USB_DM | - | I/O | USB data |
| 6 | USB_DP | - | I/O | USB data |
| 7 | VDD_USB | - | I | USB supply |
| 8 | RESET_N | - | I | RESET_N |
| 9 | VCC_RF | - | O | Output voltage RF section |
| 10 | GND | - | I | Ground |
| 11 | RF_IN | - | I | GNSS signal input |
| 12 | GND | - | I | Ground |
| 13 | GND | - | I | Ground |
| 14 | LNA_EN / Reserved | 16 | O | LNA_EN (NEO-M8N/Q/J): Antenna/LNA control Reserved (NEO-M8M): Reserved |
| 15 | Reserved | - | - | Reserved |
| 16 | Reserved | - | - | Reserved |
| 17 | Reserved | - | - | Reserved |
| 18 | SDA / SPI CS_N | 9 | I/O | DDC data if D_SEL = 1 (or open) SPI chip select if D_SEL = 0 |
| 19 | SCL / SPI CLK | 8 | I/O | DDC clock if D_SEL = 1 (or open) SPI clock if D_SEL = 0 |
| 20 | TXD / SPI MISO | 6 | O | Serial port if D_SEL = 1 (or open) SPI MISO if D_SEL = 0 |
| 21 | RXD / SPI MOSI | 7 | I | Serial port if D_SEL = 1 (or open) SPI MOSI if D_SEL = 0 |
| 22 | V_BCKP | - | I | Backup voltage supply |
| 23 | VCC | - | I | Supply voltage |
| 24 | GND | - | I | Ground |

เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ จ.2 คุณสมบัติทั่วไปของ โมดูล NEO-Ublox M8N (2) ซึ่งประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.