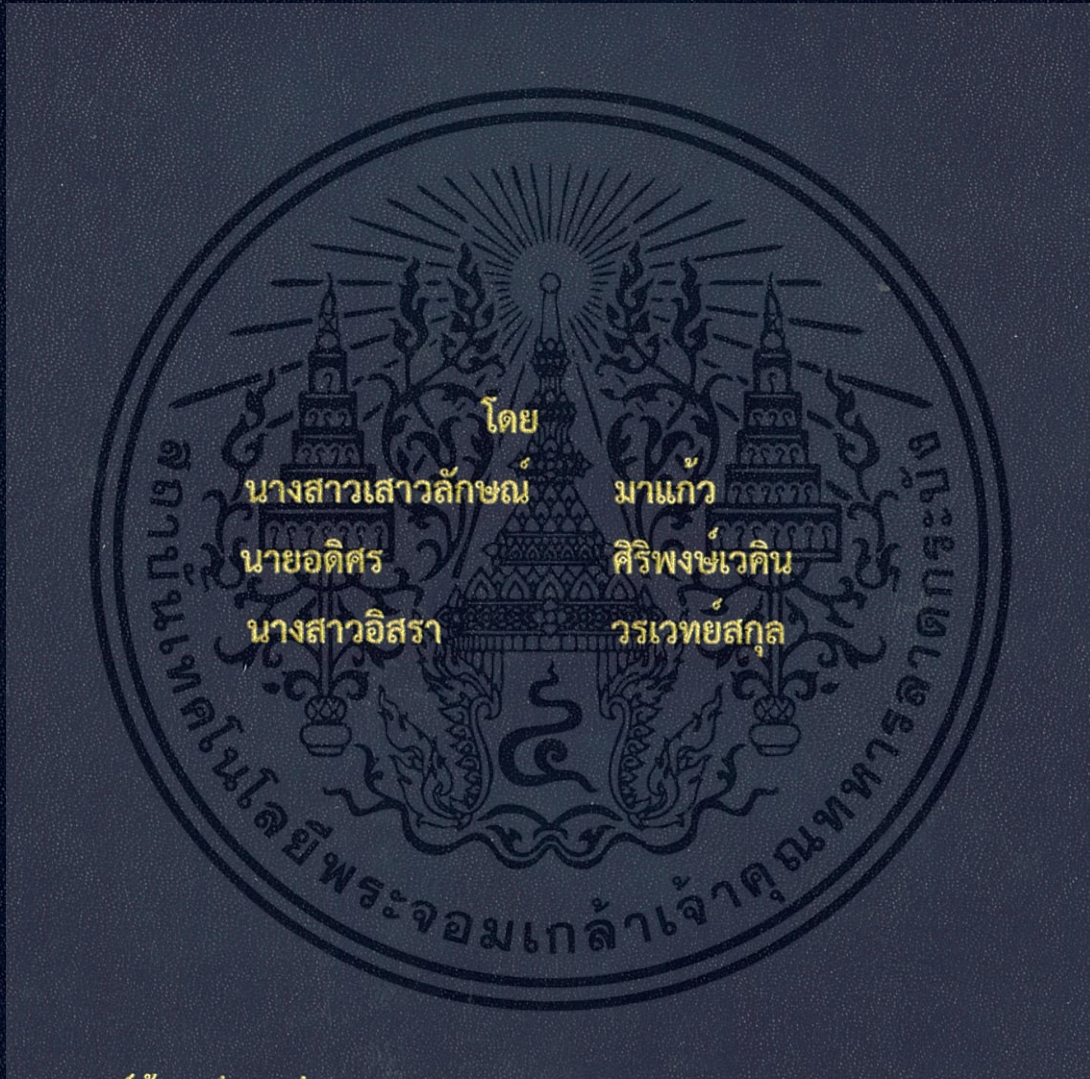


ระบบกำหนดเส้นทางการบินของอากาศยาน 4 ใบพัด
ระบุตำแหน่งผ่านระบบปฏิบัติการแอนดรอยด์
WAY POINTSYSTEM OF UNMANNED AERIAL VEHICLE
SPECIFICATION BY ANDROID APPLICATION



ปฏิญานีพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

ระบบกำหนดเส้นทางการบินของอากาศยาน 4 ใบพัด
ระบุตำแหน่งผ่านระบบปฏิบัติการแอนดรอยด์
WAY POINT SYSTEM OF UNMANNED AERIAL VEHICLE
SPECIFICATION BY ANDROID APPLICATION

โดย
นางสาวเสาวลักษณ์ มาแก้ว 54011445
นายอดิศร ศิริพงษ์เวคิน 54011459
นางสาวอิสรา วรเวทย์สกุล 54011568

อาจารย์ที่ปรึกษา
รศ.ดร. ไกรสิน ส่องวัฒนา
ผศ.ดร. พิเชฐ ม่วงนวล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมโทรคมนาคม
คณะวิศวกรรมศาสตร์

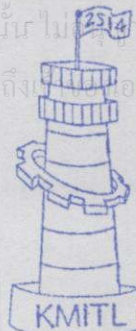
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้
ไม่ว่ากรณีมีลิขสิทธิ์ของงานที่มีให้คิดเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึง
ผ่านการตรวจรูปเล่มแล้ว
(.....)
อาจารย์ที่ปรึกษา
7, 5, 58

วิศวกรรมโทรคมนาคม
Telecommunications Engineering



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้
ไม่ว่ากรณีมีลิขสิทธิ์ของงานที่มีให้คิดเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึง
ผ่านการตรวจชิ้นงานแล้ว
(.....)
กรรมการผู้ตรวจชิ้นงาน
14, 5, 58

วิศวกรรมโทรคมนาคม
Telecommunications Engineering

ปริญญาานิพนธ์ปีการศึกษา 2557

ภาควิชาวิศวกรรมโทรคมนาคม


คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบกำหนดเส้นทางการบินของอากาศยาน 4 ใบพัด ระบุตำแหน่งผ่าน
ระบบปฏิบัติการแอนดรอยด์

WAY POINT SYSTEM OF UNMANNED AERIAL VEHICLE SPECIFICATION BY
ANDROID APPLICATION

ผู้จัดทำ

- | | | |
|---------------------|---------------|----------|
| 1. นางสาวเสาวลักษณ์ | มาแก้ว | 54011445 |
| 2. นายอดิศร | ศิริพงษ์เวคิน | 54011459 |
| 3. นางสาวอิสรา | วรวุฒย์สกุล | 54011568 |


.....
(รศ.ดร. ไกรสิน สังวanna)

อาจารย์ที่ปรึกษา

.....
(ผศ.ดร.พิเชฐ ม่วงนวล)

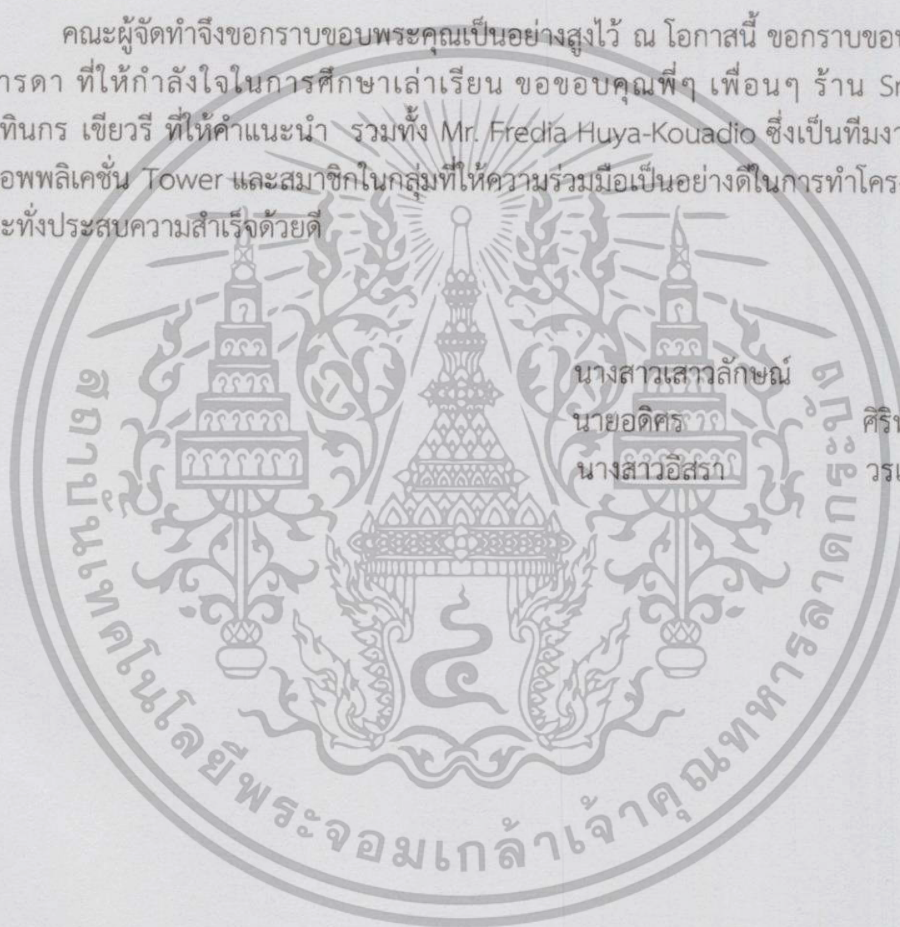
อาจารย์ที่ปรึกษาร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า-
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เรื่อง ระบบกำหนดเส้นทางการบินของอากาศยาน 4 ใบพัด ระบุตำแหน่งผ่านระบบปฏิบัติการแอนดรอยด์ สำเร็จลุล่วงได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา รศ.ดร. ไกรสิน ส่วงวัฒนา, ผศ.ดร. พิเชฐ ม่วงนวล และอาจารย์ในภาควิชาวิศวกรรมโทรคมนาคมที่ได้ให้คำปรึกษาในการศึกษาค้นคว้า แนะนำขั้นตอน และวิธีจัดทำโครงการจนสำเร็จลุล่วงด้วยดี

คณะผู้จัดทำจึงขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้ ขอกราบขอบพระคุณ บิดา มารดา ที่ให้กำลังใจในการศึกษาเล่าเรียน ขอขอบคุณพี่ๆ เพื่อนๆ ร้าน SrRCtech, และคุณทินกร เขียววี ที่ให้คำแนะนำ รวมทั้ง Mr. Fredia Huya-Kouadio ซึ่งเป็นทีมงานในการพัฒนาแอปพลิเคชัน Tower และสมาชิกในกลุ่มที่ให้ความร่วมมือเป็นอย่างดีในการทำโครงการครั้งนี้ จนกระทั่งประสบความสำเร็จด้วยดี



นางสาวเสาวลักษณ์
นายอดิศร
นางสาวอิสรา

มาแก้ว
ศิริพงษ์เวทิน
วรเวทย์สกุล
ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบกำหนดเส้นทางการบินของอากาศยาน 4 ใบพัด
ระบุตำแหน่งผ่านระบบปฏิบัติการแอนดรอยด์
WAY POINT SYSTEM OF UNMANNED AERIAL VEHICLE
SPECIFICATION BY ANDROID APPLICATION

โดย นางสาวเสาวลักษณ์ มาแก้ว 54011445
นายอดิสร ศิริพงษ์เวคิน 54011459
นางสาวอิสรา วรเวทย์สกุล 54011568

อาจารย์ที่ปรึกษา รศ.ดร. ไกรสิน ส่องวัฒนา
อาจารย์ที่ปรึกษารวม ผศ.ดร.พิเชฐ ม่วงนวล

บทคัดย่อ

ปฏิญานี้ฉบับนี้เป็นการพัฒนาระบบกำหนดเส้นทางการบินของอากาศยาน 4 ใบพัด โดยระบุตำแหน่งผ่านระบบปฏิบัติการแอนดรอยด์ เพื่อให้อากาศยาน 4 ใบพัดสามารถเคลื่อนที่ไปยังตำแหน่งปลายทางที่ระบุไว้ ซึ่งใช้อุปกรณ์ระบุตำแหน่ง (GPS) ในการอ้างอิงตำแหน่ง และใช้บอร์ดควบคุมการบิน APM ควบคุมการเคลื่อนที่ของอากาศยาน 4 ใบพัด ทั้งในส่วนของการเชื่อมต่ออิเล็กทรอนิกส์, ไจโรเซ็นเซอร์ รวมทั้งสปีดคอนโทรล โดยมีส่วนของแอปพลิเคชันที่ถูกพัฒนาใช้ในการช่วยระบุตำแหน่งปลายทางบนแผนที่จาก Google Map และมีการติดตั้งอุปกรณ์บันทึกภาพไว้บนอากาศยาน 4 ใบพัด ที่ทำการบันทึกภาพแบบต่อเนื่องจากมุมสูงเก็บไว้ในหน่วยความจำบนตัวกล้อง

ABSTRACT

This research is going to presents about the target point system of an unmanned aerial vehicle controlled by android application. This system can move to any place without a human pilot aboard with using GPS to navigate the destination. Furthermore, it uses multi-rotor controller board; Ardupilot APM 2.5, to control all of sensors on board, i.e. compass, gyros and speed control, which has Google Maps applications part to help identify the destination. Lastly, it has a camera on the multi-rotor that can be long-lasting record from sky to the ground station, stored in memory in the camera. Also, using for the traffic exploration.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
กิตติกรรมประกาศ.....	I
บทคัดย่อ	II
สารบัญ	III
สารบัญรูป	VIII
สารบัญตาราง.....	XI
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขตของโครงการ.....	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง	
2.1 เครื่องวัดอำนาจแม่เหล็ก.....	3
2.1.1 ประเภทของเครื่องวัดอำนาจแม่เหล็ก.....	3
2.1.2 เครื่องวัดอำนาจแม่เหล็กฟลักซ์เกต.....	3
2.1.3 การเชื่อมต่ออุปกรณ์ระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกแบบไอเอสแควซี.....	4
2.2 สปีดคอนโทรล.....	4
2.2.1 ชนิดของสปีดคอนโทรล.....	5
2.2.2 หลักการทำงานของสปีดคอนโทรล.....	5
2.3 แอคเซลโรมิเตอร์.....	5
2.3.1 การตรวจวัดการช็อก (shock) และการสั่นสะเทือน (Vibration).....	6
2.3.2 การตรวจวัดอัตราเร่งของวัตถุ.....	6
2.4 อากาศยานแบบ 4 ใบพัด	8
2.4.1 หลักการทำงานของอากาศยานแบบ 4 ใบพัด.....	9
2.4.2 รูปแบบการเคลื่อนที่ของอากาศยานแบบ 4 ใบพัด.....	9
2.5 บอร์ดควบคุมการบิน Arducopter APM 2.5.....	10
2.5.1 ไมโครคอนโทรลเลอร์ ATMEGA2560.....	10
2.5.2 เซ็นเซอร์ไจโรสโคปดิจิตอล และ เซ็นเซอร์วัดความเร่ง 3 MPU-6000.....	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.5.3 เซ็นเซอร์วัดความดันอากาศ MS5611.....	10
2.5.4 เซ็นเซอร์เข็มทิศอิเล็กทรอนิกส์ Honeywell's HMC5883L-TR chip.....	10
2.6 ไจโรสโคป.....	10
2.6.1 หลักการทำงานของไจโรสโคป.....	11
2.7 อุปกรณ์ระบุตำแหน่ง (GPS).....	12
2.7.1 องค์ประกอบของอุปกรณ์ระบุตำแหน่ง (GPS).....	12
2.7.2 การทำงานของอุปกรณ์ระบุตำแหน่ง (GPS).....	13
2.7.3 การประยุกต์ใช้งาน.....	13
2.7.4 การทำงานของระบบนำทางอุปกรณ์ระบุตำแหน่ง (GPS).....	13
2.7.5 อุปกรณ์ที่ช่วยในการนำทางด้วยอุปกรณ์ระบุตำแหน่ง (GPS).....	14
2.7.6 ปัจจัยที่มีผลต่อความถูกต้องของตำแหน่ง.....	14
2.7.7 ข้อมูลเอ็นเอ็มอีไอ.....	15
2.8 ระบบโทรมาตร (Telemetry).....	15
2.9 PID Control.....	16
2.9.1 ทฤษฎีคอนโทรลเลอร์แบบ PID.....	17
2.9.2 รวมสามเทอมเข้าด้วยกัน.....	19
2.9.3 การปรับแต่งค่า (Loop Tuning).....	20
2.9.4 ข้อจำกัดของคอนโทรลเลอร์แบบ PID.....	21
2.10 Android Studio.....	21
2.10.1 ข้อดีของ Android Studio.....	21
บทที่ 3 การออกแบบ และจัดทำปริญญานิพนธ์	
3.1 การออกแบบ.....	22
3.1.1 การทำงานของโปรแกรมรวม.....	23
3.2 การออกแบบโปรแกรมควบคุมการเคลื่อนที่ของอากาศยาน 4 ใบพัด.....	32
3.2.1 ส่วนการควบคุมการเคลื่อนที่ (แบบใช้รีโมทควบคุม).....	32
3.2.2 ส่วนการควบคุมการเคลื่อนที่ (แบบใช้โปรแกรมควบคุม).....	36
3.2.3 ส่วนแอพพลิเคชั่นในระบบปฏิบัติการแอนดรอยด์.....	39
3.2.4 โปรแกรมที่ใช้ในการศึกษา.....	43

สารบัญ (ต่อ)

	หน้า
3.3 การจัดเก็บผลการทดลอง	45
3.3.1 การตรวจความพร้อม (Calibrate) ของเข็มทิศอิเล็กทรอนิกส์.....	45
3.3.2 การทดสอบรับสัญญาณจากรีโมทคอนโทรล.....	45
3.3.3 การตรวจความพร้อม (Calibrate) ของรีโมทคอนโทรล และตัวรีซีฟเวอร์.....	45
3.3.4 การทดสอบการกำหนดโหมดการบิน	45
3.3.5 การทดสอบรับค่าความสูงจากบาร์โรมิเตอร์	45
3.3.6 การทดสอบการใช้งานอุปกรณ์ระบุพิกัด (GPS).....	45
3.3.7 การทดสอบการรับค่าพารามิเตอร์ Roll จากไจโรสโคป	45
3.3.8 การทดสอบการรับค่าพารามิเตอร์ Pitch จากไจโรสโคป.....	45
3.3.9 การทดสอบการรับค่าพารามิเตอร์ Yaw จากไจโรสโคป	45
3.3.10 การทดสอบวัดสัญญาณด้านเอาต์พุตที่ใช้ในการควบคุมความเร็วมอเตอร์ผ่านสปีด คอนโทรล.....	46
3.3.11 การทดสอบวัดความเร็วของมอเตอร์เทียบกับสัญญาณเอาต์พุตจากบอร์ดควบคุมการ บิน.....	46
3.3.12 การทดสอบหาความเร็วมอเตอร์ต่ำสุดที่ทำให้อากาศยาน 4 ใบพัด ยกตัว	46
3.3.13 การทดสอบวัดสัญญาณเอาต์พุตควบคุมความเร็วมอเตอร์จากบอร์ดควบคุมการบิน APM ในกรณีต่าง ๆ.....	46
3.3.14 การทดสอบการบินขึ้น (Take off) และลงจอด (Landing)	46
3.3.15 การทดสอบรักษาระดับความสูงการบินของอากาศยาน 4 ใบพัด.....	46
3.3.16 การทดสอบการเลือกใช้ระดับความสูงในการบินที่กฏของอากาศยาน 4 ใบพัด.....	46
3.3.17 การทดสอบเก็บข้อมูล (Data) พิกัดที่ใช้ในการบินของอากาศยาน 4 ใบพัด.....	46
3.3.18 การทดสอบหาค่าพารามิเตอร์ของ PID control ที่ใช้ในการแก้ไขความผิดพลาด ของอุปกรณ์ระบุตำแหน่ง (GPS).....	47
3.3.19 การทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 1 จุด โดยกำหนดค่าความสูงของอากาศยานคงที่.....	47
3.3.20 การทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 5 จุด โดยกำหนดค่าความสูงของอากาศยานคงที่.....	47
3.3.21 การทดสอบการใช้งานกำหนดเป้าหมายการบินของแอปพลิเคชัน ในระบบปฏิบัติการแอนดรอยด์.....	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาค้นคว้า ไม่นอนุญาติให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการทดลอง	
4.1 ผลการตรวจความพร้อม (Calibrate) ของเข็มทิศอิเล็กทรอนิกส์.....	48
4.2 ผลการทดสอบรับสัญญาณจากรีโมทคอนโทรล.....	49
4.3 ผลการตรวจความพร้อม (Calibrate) ของรีโมทคอนโทรล และตัวรีซีฟเวอร์.....	49
4.4 ผลการทดสอบการกำหนดโหมดการบิน.....	50
4.5 ผลการทดสอบรับค่าความสูงจากบาร์โอมิเตอร์.....	51
4.6 ผลการทดสอบการใช้งานอุปกรณ์ระบุพิกัด (GPS).....	52
4.7 ผลการทดสอบการรับค่าพารามิเตอร์ Roll จากไจโรสโคป.....	53
4.8 ผลการทดสอบการรับค่าพารามิเตอร์ Pitch จากไจโรสโคป.....	55
4.9 ผลการทดสอบการรับค่าพารามิเตอร์ Yaw จากไจโรสโคป.....	57
4.10 ผลการทดสอบวัดสัญญาณด้านเอาต์พุตที่ใช้ในการควบคุมความเร็วมอเตอร์ผ่านสปีดคอนโทรล.....	59
4.11 ผลการทดสอบวัดความเร็วของมอเตอร์เทียบกับสัญญาณเอาต์พุตจากบอร์ดควบคุมการบิน.....	60
4.12 ผลการทดสอบหาความเร็วมอเตอร์ต่ำสุดที่ทำให้อากาศยาน 4 ใบ พัดยกตัว.....	61
4.13 ผลการทดสอบวัดสัญญาณเอาต์พุตควบคุมความเร็วมอเตอร์จากบอร์ดควบคุมการบิน APM ในกรณีต่าง ๆ.....	62
4.14 ผลการทดสอบการบินขึ้น (Take off) และลงจอด (Landing).....	66
4.15 ผลการทดสอบรักษาระดับความสูงการบินของอากาศยาน 4 ใบพัด.....	67
4.16 ผลการทดสอบการเลือกใช้ระดับความสูงในการบันทึกภาพของอากาศยาน 4 ใบพัด.....	69
4.17 ผลการทดสอบเก็บข้อมูล (Data) พิกัดที่ใช้ในการบินของอากาศยาน 4 ใบพัด.....	71
4.18 ผลการทดสอบหาค่าพารามิเตอร์ของ PID control ที่ใช้ในการแก้ไขความผิดพลาดของอุปกรณ์ระบุตำแหน่ง (GPS).....	71
4.19 ผลการทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 1 จุด โดยกำหนดค่าความสูงของอากาศยานคงที่.....	73
4.20 การทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 5 จุด โดยกำหนดค่าความสูงของอากาศยานคงที่.....	80
4.21 ผลการทดสอบการใช้งานกำหนดเป้าหมายการบินของแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์.....	82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับนักเรียนเพื่อการศึกษาดูเท่านั้น มิอนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์ การค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผล และข้อเสนอแนะ	
5.1 สรุปผล	83
5.2 ข้อเสนอแนะ	83
5.3 แนวทางการพัฒนา	83
บรรณานุกรม	84
ภาคผนวก ก โค้ดโปรแกรมหลักที่ใช้ในการควบคุมการบิน	87
ภาคผนวก ข โค้ดโปรแกรมที่ใช้ในการรักษาเสถียรภาพการบิน	100
ภาคผนวก ค โค้ดโปรแกรมที่ใช้ในการคำนวณหาพิกัดเป้าหมาย	103

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	การหามุมฉากทั้งสามของ Azimuth	3
2.2	ลักษณะการเชื่อมต่ออุปกรณ์แบบ I ² C Bus.....	4
2.3	สปีดคอนโทรล.....	4
2.4	โครงสร้างของแอกเซลโรมิเตอร์	5
2.5	การตรวจวัดการช็อก (shock) และการสั่นสะเทือน (Vibration).....	6
2.6	หลักการพื้นฐานของมิเตอร์วัดอัตราเร่งแบบไซมิกแมส	6
2.7	โครงสร้างพื้นฐานของมิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริก.....	8
2.8	อากาศยานแบบ 4 ใบพัด	8
2.9	บอร์ดควบคุมการบิน Arducopter APM 2.5.....	10
2.10	หลักการทํางานของไจโรสโคป.....	11
2.11	ไจโรสโคปแบบมีแกน.....	12
2.12	433Mhz Radio Telemetry Kit.....	15
2.13	Block diagram ของคอนโทรลเลอร์แบบPID.....	17
2.14	การเปลี่ยนแปลงค่า PV เทียบกับ เวลา โดยเปลี่ยนค่า K_p (ค่า K_i และ K_d คงที่).....	17
2.15	การเปลี่ยนแปลงค่า PV เทียบกับเวลา โดยเปลี่ยนค่า K_i (ค่า K_p และ K_d คงที่).....	18
2.16	การเปลี่ยนแปลงค่า PV เทียบกับเวลา โดยเปลี่ยนค่า K_d (ค่า K_p และ K_i คงที่).....	19
2.17	ตารางผลของการเพิ่มค่าพารามิเตอร์ของคอนโทรลเลอร์ PID.....	20
2.18	ตารางแสดงค่า Gain ของวิธี Ziegler-Nichols.....	20
2.19	โปรแกรม Android Studio 1.1.0.....	21
3.1	โครงสร้างการทํางานของระบบ.....	22
3.2	ผังการทํางานของส่วนโปรแกรม (ใช้โหมดในการควบคุมการบิน).....	23
3.3	ผังการทํางานของส่วนโปรแกรม (ควบคุมแบบอัตโนมัติ) (setup).....	24
3.4	ผังการทํางานของส่วนโปรแกรม (ควบคุมแบบอัตโนมัติ) (loop).....	25
3.5	การคำนวณหาทิศไปยังพิกัดเป้าหมาย.....	26
3.6	การคำนวณหาระยะทางไปยังพิกัดเป้าหมาย.....	26
3.7	เฟรม 4 ใบพัดชนิดแกนสมดุล	27
3.8	บอร์ดควบคุมการบิน Arducopter APM 2.5.....	27
3.9	มอเตอร์ EMAX MT2213-935KV	28
3.10	ใบพัดขนาด 10x45 นิ้ว.....	28

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.11 สปีดคอนโทรล MYSTCRY 30A SimonK.....	28
3.12 รีโมทคอนโทรล FS-TH9X.....	29
3.13 เทเลเมทรีความถี่ 433MHz.....	29
3.14 อุปกรณ์ระบุตำแหน่ง (GPS) NEO-7N Ublox.....	30
3.15 การเชื่อมต่อกล่องสัญญาณด้าน input กับบอร์ดควบคุม APM.....	30
3.16 แสดงการเชื่อมต่อบอร์ดควบคุมการบิน APM กับมอเตอร์ รีซีฟเวอร์ และจีพีเอส.....	31
3.17 ตำแหน่งการวางมอเตอร์.....	31
3.18 วงจรนับรอบความเร็วของมอเตอร์.....	32
3.19 การทำงานขณะบิน ขึ้น - ลง ของทำอากาศยาน 4 ใบพัด.....	33
3.20 การทำงานขณะบิน เอียงซ้าย - ขวา ของทำอากาศยาน 4 ใบพัด.....	34
3.21 การทำงานขณะบิน เอียงหน้า - หลัง ของทำอากาศยาน 4 ใบพัด.....	35
3.22 การทำงานโปรแกรมควบคุมการบินขึ้น (Take off) และลงจอด (Landing).....	36
3.23 การทำงานโปรแกรมควบคุมการตัดสินใจ และระบุพิกัดเป้าหมาย.....	37
3.24 ลำดับการบินไปยังเป้าหมายถัดไปของโปรแกรม.....	38
3.25 การทำงานของแอปพลิเคชัน.....	39
3.26 ไอคอนของแอปพลิเคชัน.....	40
3.27 เมนูแสดงผลหน้าแอปพลิเคชัน.....	40
3.28 หน้าต่าง Flight.....	41
3.29 หน้าต่าง Editor.....	41
3.30 ระบุพิกัดในหน้า Editor.....	42
3.31 กำหนดความสูง และ การหน่วงเวลาของอากาศยาน.....	42
3.32 หน้าต่าง Flight History.....	46
3.33 แสดงโปรแกรม ArduPilot-Arduino 1.0.3 ที่ใช้ในการพัฒนา.....	46
3.34 หน้าต่างโปรแกรม Mission Planer 1.3.9.....	47
3.35 หน้าต่างโปรแกรม Android Studio 1.1.0.....	47
4.1 Calibrate Compass.....	48
4.2 การกำหนด Channel Calibrate Radio.....	49
4.3 แสดงรูปความกว้างพัลส์สัญญาณของรีโมทคอนโทรลที่ใช้ในการควบคุมอากาศยาน 4 ใบพัด.....	49
4.4 การทดสอบวัดค่าสูงสุด - ต่ำสุดในการ Calibrate Radio ผ่านโปรแกรม Mission Planer.....	50
4.5 การกำหนดโหมดของการบินของอากาศยานบิน 4 ใบพัด.....	50

สารบัญตาราง

ตารางที่	หน้า
4.1 ผลการทดสอบการรับค่าความสูงจากบารอมิเตอร์.....	52
4.2 ผลการทดลองการรับค่าพารามิเตอร์ roll จากไจโรสโคป.....	54
4.3 ผลการทดลองการรับค่าพารามิเตอร์ pitch จากไจโรสโคป.....	56
4.4 ผลการทดลองการรับค่า yaw จากไจโรสโคป.....	58
4.5 ผลการทดสอบการบิรักษาระดับความสูง.....	68
4.6 ตารางแสดงภาพเป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด เมื่อบินไปถึงเป้าหมาย.....	70
4.7 ความสัมพันธ์ระหว่าง ค่า K_p กับระยะความคลาดเคลื่อนของตำแหน่งปลายทาง เมื่อกำหนด ค่า K_i และ K_d เท่ากับ 0.....	72
4.8 ความสัมพันธ์ระหว่าง K_i กับระยะผิดพลาด เมื่อกำหนดให้ $K_p = 0.36 (0.45K_c)$ และ $K_d = 0$	72
4.9 ภาพเป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด เมื่อบินไปถึงเป้าหมาย.....	75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบัน เครื่องเล่นวิทยุบังคับมีความนิยมกันอย่างแพร่หลาย โดยมีรูปแบบการเล่นแตกต่างกัน อาทิเช่น รถวิทยุบังคับ เรือวิทยุบังคับ เครื่องบินวิทยุบังคับ รวมถึงเครื่องบิน 4 ใบพัด ที่มีชื่ออย่างเป็นทางการว่าอากาศยาน 4 ใบพัด ซึ่งเมื่อนำเครื่องเล่นเหล่านี้มาใช้งานร่วมกับเทคโนโลยีที่มีความทันสมัย จึงเป็นการประยุกต์ใช้เครื่องเล่นต่างๆ เหล่านี้ให้เกิดประโยชน์ในหลายด้าน ทั้งในด้านการจราจร ด้านการทหาร หรือด้านการสำรวจทางธรณีวิทยา เป็นต้น ซึ่งทำให้มีการประดิษฐ์ และพัฒนาเครื่องเล่นเหล่านี้กันมากขึ้น เพื่อนำมาใช้ประโยชน์ และอำนวยความสะดวกในด้านต่างๆ

คณะผู้จัดทำจึงพัฒนาระบบกำหนดเส้นทางการบินของอากาศยาน 4 ใบพัด โดยระบุตำแหน่งผ่านระบบปฏิบัติการแอนดรอยด์ เพื่อนำไปประยุกต์ใช้ประโยชน์ในด้านต่าง ๆ ตัวอย่างเช่น ด้านการจราจร โดยช่วยอำนวยความสะดวกในการทำงานของเจ้าหน้าที่ ทำให้เจ้าหน้าที่ทราบสถานการณ์บนท้องถนนเกี่ยวกับปัญหาการจราจร อาทิ การเกิดอุบัติเหตุบนท้องถนน หรือในช่วงเวลาเร่งด่วนที่ทำให้ปริมาณรถบนท้องถนนค่อนข้างหนาแน่น เป็นการช่วยต่อการควบคุมสถานการณ์ สามารถเข้าถึงตำแหน่งที่ต้องการตรวจสอบได้อย่างรวดเร็ว และสามารถเพิ่มประสิทธิภาพในการแก้ไขปัญหาได้อย่างทันท่วงที

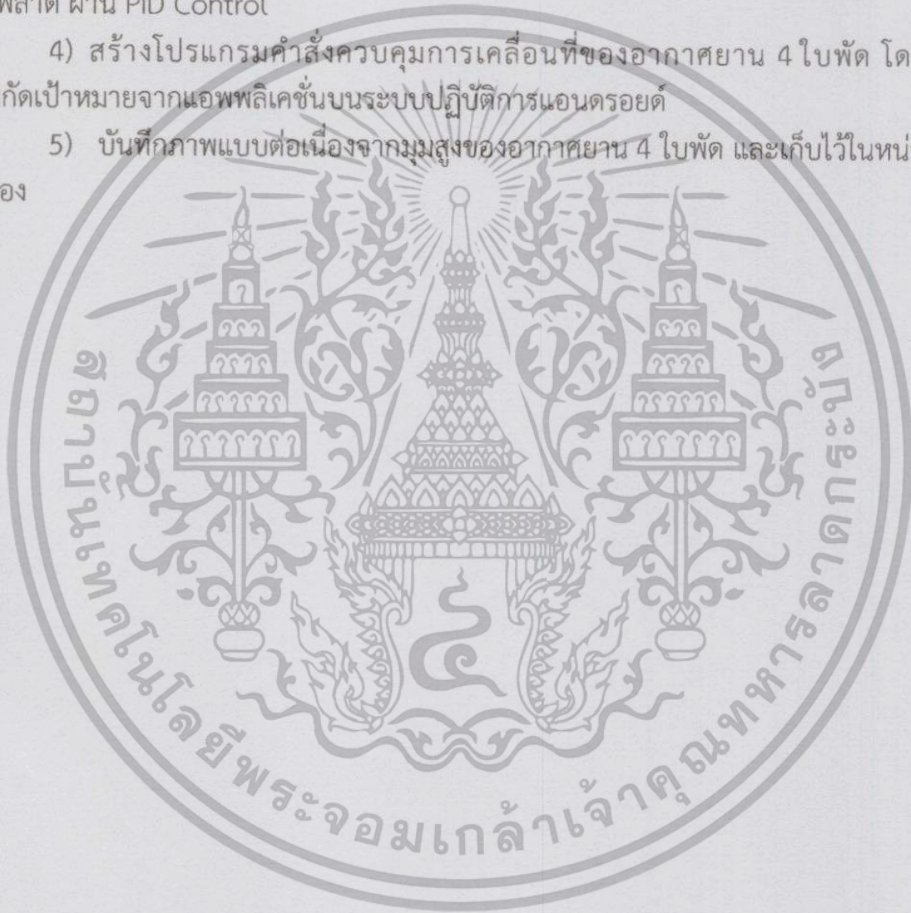
1.2 วัตถุประสงค์

- 1) เพื่อศึกษาการควบคุมอากาศยาน 4 ใบพัด
- 2) เพื่อศึกษาหลักการทางาน และวิธีประยุกต์ใช้งานอุปกรณ์ระบุตำแหน่ง (GPS)
- 3) เพื่อศึกษาหลักการควบคุมการเคลื่อนที่ของอากาศยาน 4 ใบพัด
- 4) เพื่อศึกษาการใช้งานเข็มทิศอิเล็กทรอนิกส์
- 5) เพื่อศึกษาการระบุตำแหน่งเป้าหมายของการเคลื่อนที่
- 6) เพื่อศึกษาระบบแก้ไขความผิดพลาด (PID Control)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงการ

- 1) บังคับการเคลื่อนที่ของอากาศยาน 4 ใบพัดได้โดยอาศัยการทำงานของมอเตอร์ 4 ตัว
- 2) ควบคุมการบินของอากาศยาน 4 ใบพัด โดยใช้โปรแกรมคำสั่งในการควบคุมการเคลื่อนที่
- 3) รักษาสมดุลการบินของอากาศยาน 4 ใบพัด โดยอาศัยโรเซ็นเซอร์ควบคุมกับการแก้ไขความผิดพลาด ผ่าน PID Control
- 4) สร้างโปรแกรมคำสั่งควบคุมการเคลื่อนที่ของอากาศยาน 4 ใบพัด โดยสามารถกำหนดพิกัดเป้าหมายจากแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์
- 5) บันทึกภาพแบบต่อเนื่องจากมุมสูงของอากาศยาน 4 ใบพัด และเก็บไว้ในหน่วยความจำบนตัวกล้อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 เครื่องวัดอำนาจแม่เหล็ก

เครื่องวัดอำนาจแม่เหล็กเป็นเครื่องมือวัดที่ใช้วัดความเข้ม และทิศทางของสนามแม่เหล็กมีทั้งแบบที่ผลิตในท้องปฏิบัติการหรือมีอยู่ในธรรมชาติ บางประเทศ เช่น สหรัฐอเมริกา แคนาดา และออสเตรเลีย ได้จัดให้เครื่องวัดอำนาจแม่เหล็กมีความสำคัญมาก โดยถือว่าเป็นเทคโนโลยีทางทหาร และต้องมีการควบคุมการใช้งาน ในระดับสากล หน่วยของการวัดความเข้มของสนามแม่เหล็กจะเป็นเทสลา (Tesla) ส่วนในทางวิศวกรรมจะวัดสนามแม่เหล็กในหน่วย เกาส์ (Gauss) โดยที่ 1 เกาส์ จะเท่ากับ 100,000 นาโนเทสลา (nT) เครื่องวัดอำนาจแม่เหล็กมีความหลากหลายในการใช้งาน และสามารถใช้งานได้ทั้งในยานอวกาศ เครื่องบิน เฮลิคอปเตอร์หรือบนพื้นดิน ขึ้นอยู่กับการใช้งาน

2.1.1 ประเภทของเครื่องวัดอำนาจแม่เหล็ก

เครื่องวัดอำนาจแม่เหล็กสามารถแบ่งออกได้เป็น 2 ประเภทพื้นฐาน ดังนี้

2.1.1.1 การวัดแบบสเกลลา เป็นการวัดความเข้มของสนามแม่เหล็ก แต่ไม่มีทิศทางของสนามแม่เหล็ก

2.1.1.2 การวัดแบบเวกเตอร์ มีความสามารถในการวัดองค์ประกอบของสนามแม่เหล็กโดยมีทิศทาง ซึ่งทิศทางนั้นจะเป็นสามมิติไม่เพียงแคทิศทางเหนือไปได้ แต่ยังรวมถึงมุมเอียงจากแนวนอนด้วย เวกเตอร์ของเครื่องวัดอำนาจแม่เหล็ก มาจากการใช้มุมฉากทั้งสามของ Azimuth ดังรูปที่ 2.1 โดยใช้รากที่สองของผลรวมของกำลังสองขององค์ประกอบที่ความเข้มของสนามแม่เหล็กรวม (ความเข้มสนามแม่เหล็กรวมเรียกว่า TMI) สามารถคำนวณได้โดยทฤษฎีบทของพีทาโกรัส



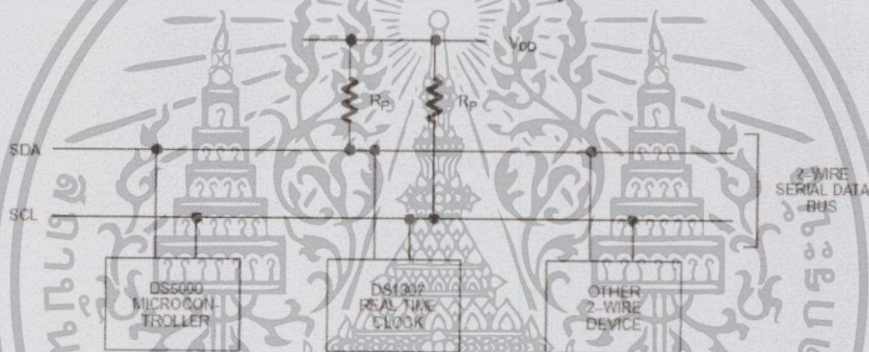
รูปที่ 2.1 การหามุมฉากทั้งสามของ Azimuth [4]

2.1.2 เครื่องวัดอำนาจแม่เหล็กฟลักซ์เกต

เอกสารนี้เป็นเอกสาร เครื่องวัดอำนาจแม่เหล็กฟลักซ์เกต ประกอบด้วยขดลวดขนาดเล็กลงสองขดที่ไวต่อสนามแม่เหล็ก กระแสไฟฟ้าสลับจะถูกส่งผ่านขดลวดขดหนึ่ง การเหนี่ยวนำไฟฟ้าทำให้

เกิดกระแสไฟฟ้าในอีกขดลวดหนึ่ง โดยปกติแล้วกระแสด้านอินพุทและเอาต์พุตต้องตรงกัน ส่วนใหญ่แล้ว แรงดันเอาต์พุตของขดลวดเป็นแรงดันอนาล็อกที่เป็นสัดส่วนกับสนามแม่เหล็ก เซ็นเซอร์ฟลักซ์เกตมีราคาไม่แพง และมีขนาดกะทัดรัด แข็งแรงทนทาน ใช้พลังงานต่ำ และสามารถนำไปใช้งานได้หลายด้าน

2.1.3 การเชื่อมต่ออุปกรณ์ระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกแบบ ไอ สแควซี (I²C Bus : Inter Integrate circuit Bus) เป็นการสื่อสารอนุกรมแบบซิงโครนัส (Synchronous) เพื่อใช้ในการติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์ (MCU) กับอุปกรณ์ภายนอก ซึ่งถูกพัฒนาขึ้นโดยบริษัท Philips Semiconductors โดยใช้สายสัญญาณเพียง 2 เส้น เท่านั้น คือ Serial Data (SDA) และสาย Serial Clock (SCL) โดยที่ขาสัญญาณทั้ง 2 ต้องต่อกับตัวต้านทานแบบ Pull Up 2-10 k Ω ซึ่งสามารถเชื่อมต่ออุปกรณ์จำนวนหลายตัวเข้าด้วยกันได้ จึงทำให้ไมโครคอนโทรลเลอร์ใช้พอร์ตเพียง 2 พอร์ตเท่านั้นดังรูปที่ 2.2



รูปที่ 2.2 ลักษณะการเชื่อมต่ออุปกรณ์แบบ I²C Bus [5]

2.2 สปีดคอนโทรล

สปีดคอนโทรล คือ อุปกรณ์อิเล็กทรอนิกส์ที่ใช้สำหรับควบคุมการจ่ายแรงดันไฟฟ้าให้กับมอเตอร์ไฟฟ้า ทำให้ควบคุมความเร็วของมอเตอร์ไฟฟ้าได้ โดยมีสายไฟออกมาทั้งสองด้าน ด้านหนึ่งใช้จ่ายกระแสไฟฟ้าให้แก่มอเตอร์ ส่วนอีกด้านหนึ่งมีอยู่ 3 สาย สองสายแรกสำหรับรับกระแสไฟจากแบตเตอรี่ และสายสุดท้ายใช้สำหรับรับคำสั่งจากตัวควบคุม ดังรูปที่ 2.3



รูปที่ 2.3 สปีดคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในเชิงการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ชนิดของสปีดคอนโทรล

สปีดคอนโทรล อาจแบ่งได้เป็น 2 ชนิด คือ

2.2.1.1 แบบกระแสตรง ใช้ควบคุมความเร็วมอเตอร์แบบมีแปรงถ่าน หรือ Brush Motor สปีดคอนโทรลชนิดนี้ ใช้ควบคุมการจ่ายกระแสไฟกระแสตรงจากแบตเตอรี่สู่มอเตอร์ ถ้าจ่ายแรงดันไฟมากมอเตอร์ก็หมุนเร็ว จ่ายแรงดันไฟน้อยมอเตอร์ก็หมุนช้า ตามแต่ที่ ต้องการ

2.2.1.2 แบบกระแสสลับ ใช้ควบคุมความเร็วมอเตอร์แบบไร้แปรงถ่าน หรือ Brushless Motor สปีดคอนโทรลชนิดนี้ ใช้การแปลงไฟจากกระแสตรงที่ได้รับจากแบตเตอรี่ เป็นไฟกระแสสลับเพื่อจ่ายแรงดันไฟให้แก่มอเตอร์แบบไร้แปรงถ่าน

2.2.2 หลักการทำงานของสปีดคอนโทรล

การควบคุมการทำงานของสปีดคอนโทรล มีลักษณะคล้ายกับของเซอร์โวมอเตอร์ โดยทำการป้อนสัญญาณความกว้างพัลส์ช่วง 1-2 ms ให้กับสปีดคอนโทรล ซึ่งความเร็วของมอเตอร์ จะแปรผันตรงกับสัญญาณความกว้างพัลส์ที่ป้อนให้กับสปีดคอนโทรล

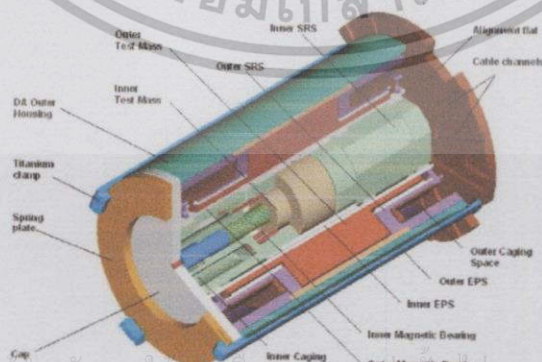
สัญญาณความกว้างพัลส์ขนาด 1.00 ms มอเตอร์ยังไม่ทำงาน

สัญญาณความกว้างพัลส์ขนาด 2.00 ms มอเตอร์มีความเร็วสูงสุด

เมื่อต้องการความเร็วของมอเตอร์ระดับอื่น ให้ทำการป้อนสัญญาณพัลส์อ้างอิงจาก สัญญาณทางด้านบน เช่น ต้องการความเร็วมอเตอร์ครึ่งหนึ่งของความเร็วสูงสุด ให้ป้อนสัญญาณ ความกว้างพัลส์ขนาด 1.5 ms เป็นต้น

2.3 แอคเชเลโรมิเตอร์

แอคเชเลโรมิเตอร์ คือ เครื่องตรวจวัดความเร่งของการเคลื่อนที่ของวัตถุ โครงสร้าง ของแอคเชเลโรมิเตอร์ จะประกอบด้วยสปริงและลูกตุ้มน้ำหนัก ดังรูปที่ 2.4 เมื่อมีการเคลื่อนที่ ด้วยความเร่ง ลูกตุ้มน้ำหนักจะถูกกดไปอีกฝั่งตรงข้ามกับการเคลื่อนที่ สปริงก็จะทำหน้าที่ดึงกลับเข้า ที่อีกครั้งเมื่อหยุดการเคลื่อนที่ การเคลื่อนที่ด้วยความเร็วคงที่ คือ ความเร่งเท่ากับศูนย์ ค่าที่วัดได้ จะไม่เปลี่ยนแปลง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 2.4 โครงสร้างของแอคเชเลโรมิเตอร์ [6]

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีให้คิดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนตัวเซนเซอร์ภายใน ที่จะใช้ในการตรวจวัดความเร่งของลูกตุ้มที่อยู่ในระบบนั้นมีหลายชนิด เช่น เพียโซอิเล็กทริก, สเตรนเกจ เป็นต้น โดยที่สามารถแบ่งลักษณะการตรวจวัดได้ 2 ลักษณะ คือ

2.3.1 การตรวจวัดการช็อก (shock) และการสั่นสะเทือน (Vibration)

ดังรูปที่ 2.5 ซึ่งการช็อก คือ อัตราเร่งขนาดมหาศาลที่เกิดขึ้นในช่วงเวลาสั้นๆ ในช่วงเวลาสั้นๆ ในขณะที่การสั่นสะเทือน คือ อัตราเร่งขนาดเล็กที่เกิดขึ้นซ้ำๆ กันไปเรื่อยๆ



รูปที่ 2.5 การตรวจวัดการช็อก (shock) และการสั่นสะเทือน (Vibration) [6]

2.3.2 การตรวจวัดอัตราเร่งของวัตถุ เพื่อนำข้อมูลไปใช้ในการระบุตำแหน่งความเร็วและระยะทางที่ได้จากการเคลื่อนที่

มิเตอร์วัดความเร่งนี้โดยหลักแล้ว จะแบ่งเป็น 2 ชนิด

2.3.2.1 มิเตอร์วัดอัตราเร่งแบบไซซมิกแมส (Seismic mass accelerometer) มิเตอร์ชนิดนี้อาศัยหลักการตรวจวัดระยะขจัดเชิงเส้นแล้วนำไปคำนวณหาอัตราเร่งที่เกิดขึ้นโดยเทคนิคดังกล่าวสามารถอธิบายได้ว่า วัตถุชิ้นหนึ่งจะมีความเร่งได้ ก็จะต้องมีแรงกระทำ ยังมีแรงกระทำมาก ก็จะยิ่งความเร่งมาก ในขณะที่เดียวกันแรงต้านทานการเคลื่อนที่ก็จะมากด้วย นอกจากนี้เมื่อมีแรงกระทำให้วัตถุเกิดการเคลื่อนที่ ก็จะมีระยะขจัด ซึ่งจะแปรผันตรงกับแรงที่มากกระทำที่วัตถุ ยิ่งแรงมาก ระยะขจัดยิ่งมาก จากความสัมพันธ์ดังกล่าวได้นำไปเป็นหลักการพื้นฐานของมิเตอร์วัดอัตราเร่งแบบไซซมิกแมสในการตรวจวัดอัตราเร่งของวัตถุในเทอมของระยะขจัดที่เกิดขึ้น ดังรูปที่ 2.6

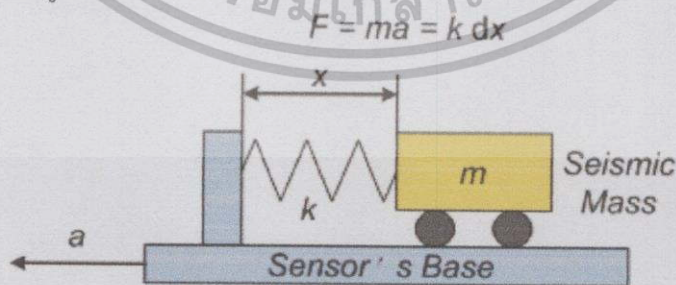


Fig-1 A spring-mass system

เอกสารนี้เป็นเอกสารที่รูปที่ 2.6 หลักการพื้นฐานของมิเตอร์วัดอัตราเร่งแบบไซซมิกแมส [6] ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างนี้มีมวล m ที่เรียกว่ามวลตรวจการสั่นไหว (Seismic mass) ยึดติดอยู่กับสปริงที่มีค่านิจของสปริงเท่ากับ k และมวลนี้สามารถเคลื่อนที่ในแนวระดับได้ ซึ่งหลักการทำงานคือ เมื่อตัวเซนเซอร์ตัวนี้ถูกทำให้มีอัตราเร่งเกิดขึ้นจะส่งผลให้มวล m เคลื่อนที่ ซึ่งระยะที่เคลื่อนที่ออกไปจะเป็นระยะขจัดเท่ากับ x และมีทิศทางตรงกันข้ามกับการเคลื่อนที่ของตัวมิเตอร์ โดยจะเห็นได้ว่า

- เมื่ออัตราเร่งของวัตถุมีค่าเพิ่มขึ้น ทำให้ระยะขจัดของมวล m มีค่าเพิ่มขึ้น

ตามไปด้วย

- เมื่ออัตราเร่งของวัตถุมีค่าลดลง ทำให้มวล m เคลื่อนที่ไปดันสปริง

- เมื่ออัตราเร่งของวัตถุหยุดลง ก็จะทำให้มวล m เคลื่อนที่กลับมาอยู่

ตำแหน่งเดิม (ตำแหน่งอ้างอิง)

ในทางปฏิบัติ สามารถวัดระยะขจัดของมวล m ได้โดยอาศัยมิเตอร์อีกชนิดหนึ่ง คือมิเตอร์วัดระยะขจัดเชิงเส้น (LVDT, potentiometer) ส่วนการวิเคราะห์หาอัตราเร่งที่เกิดขึ้นเราสามารถคำนวณได้โดยใช้คอมพิวเตอร์

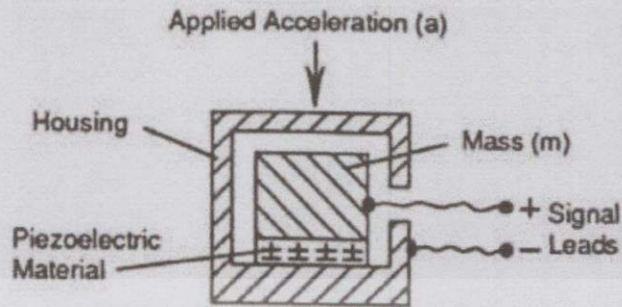
มิเตอร์วัดอัตราเร่งแบบโซนมิกแมสจะนิยมใช้ในการตรวจวัดลักษณะการสั่นไหว และลักษณะการสั่นสะเทือนที่มีความถี่ต่ำมากๆ เช่น ในเครื่องมือตรวจวัดแผ่นดินไหว หรือในเครื่องมือตรวจวัดการปะทุใต้ดินของภูเขาไฟ เป็นต้น

2.3.2.2 มิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริก (Piezoelectric accelerometer)

คุณสมบัติพื้นฐานทางไฟฟ้าของผลึกเพียโซอิเล็กทริก (Piezoelectric crystal) ถูกค้นพบโดย Pierre และ Jacques Curie ในราวปี ค.ศ.1880 ซึ่งผลึกเพียโซอิเล็กทริกมีคุณสมบัติพิเศษ คือ เมื่อถูกแรงทางกลมากระทำ จะสร้างประจุไฟฟ้าขึ้นมา โดยเป็นสัดส่วนกับแรงกระทำนั้น ซึ่งจากคุณสมบัติพิเศษนี้ได้ถูกดัดแปลงนำไปใช้สร้างอุปกรณ์ต่างๆ มากมาย เช่น ใช้เป็นแบตเตอรี่จ่ายพลังงานไฟฟ้าให้กับนาฬิกาข้อมือดิจิทัลที่เราใช้ทั่วไป และยังใช้สร้างมิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริกอีกด้วย

โครงสร้างของมิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริก จะประกอบด้วย Seismic mass ยึดติดกับผลึกเพียโซอิเล็กทริก และบรรจุอยู่ในตัวถังป้องกันดังรูปที่ 2.7 โดยผลึกเพียโซอิเล็กทริกที่นิยมนำมาใช้งาน ได้แก่ ผลึกควอตซ์ และ ผลึกโซเดียมโปตัสเซียมตาเตรต (Sodium potassium tetratrate) เพราะมีความทนทานต่อแรงกระทำ และราคาไม่สูงมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 โครงสร้างพื้นฐานของมิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริก [6]

การทำงานของมิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริก อธิบายได้ดังนี้ เมื่อ Seismic mass (m) ถูกทำให้เกิดอัตราเร่งขึ้น (ถูกกด) และจะส่งผ่านแรงกดไปกระทำกับผลึกเพียโซอิเล็กทริก ที่ถูกยึดติดอยู่ด้วยกัน ด้วยคุณสมบัติพิเศษของอุปกรณ์นี้จะทำให้ประจุไฟฟ้าถูกสร้างขึ้น และถูกส่งผ่านสัญญาณออกไปยังเอาต์พุตของวงจร โดยที่ด้านเอาต์พุตจะต้องมีวงจรขยายประจุไฟฟ้า (charge amplifier) เพื่อขยายค่าประจุไฟฟ้าที่ได้ให้เป็นแรงดันเอาต์พุตตามสัดส่วนของอัตราเร่งที่เกิด จะสามารถแสดงผลได้ด้วยโวลต์มิเตอร์

มิเตอร์วัดอัตราเร่งแบบเพียโซอิเล็กทริกตอบสนองต่อทางด้านความถี่สูงได้ดี แต่ในทางกลับกันจะมีผลตอบสนองทางด้านความถี่ต่ำที่ไม่ดีนัก มีขนาดค่อนข้างเล็ก น้ำหนักเบา และสามารถใช้งานที่มีอัตราเร่งได้สูงถึง $250,000 \text{ m/s}^2$

2.4 อากาศยานแบบ 4 ใบพัด

อากาศยานแบบ 4 ใบพัด คืออากาศยานที่มี 4 ใบพัด วางอยู่บนแกนที่ไขว้กัน ในลักษณะกากบาท ดังรูปที่ 2.8 ใช้แรงขับเคลื่อนจากใบพัดทั้ง 4 ตัว ซึ่งอากาศยานแบบ 4 ใบพัด จัดเป็นอากาศยานขึ้นลงทางดิ่ง โดยการควบคุมการเคลื่อนที่ของอากาศยานแบบ 4 ใบพัด จะใช้การปรับรอบความเร็วของใบพัดแต่ละตัว



รูปที่ 2.8 อากาศยานแบบ 4 ใบพัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 หลักการทำงานของอากาศยานแบบ 4 ใบพัด

การบินของอากาศยานแบบ 4 ใบพัด เป็นการเคลื่อนที่ 4 แกน ซึ่งมีลักษณะการเคลื่อนที่ทั้งหมด 8 รูปแบบคือ ขึ้น-ลง เดินหน้า-ถอยหลัง เอียงซ้าย-เอียงขวา และ หมุนซ้าย-หมุนขวา โดยทำการแบ่งใบพัดทั้งสี่เป็น 2 คู่ คือ คู่หน้า-หลัง มีลักษณะการหมุนเป็นแบบตามเข็มนาฬิกา และคู่ซ้าย-ขวามีลักษณะการหมุนเป็นแบบทวนเข็มนาฬิกา เพื่อให้ทั้ง 4 ใบพัดเกิดแรงบิด 2 ทิศทาง เพื่อใช้ในการควบคุมการหมุนของอากาศยาน 4 ใบพัด

2.4.2 รูปแบบการเคลื่อนที่ของอากาศยานแบบ 4 ใบพัด

รูปแบบการเคลื่อนที่และการควบคุมรอบความเร็วของใบพัด สามารถแบ่งออกได้เป็น

2.4.2.1 การเคลื่อนที่ขึ้นในแนวตั้ง

ควบคุมความเร็วของใบพัดทั้ง 4 ใบ โดยการเพิ่มความเร็วรอบของใบพัดทั้ง 4 ใบ ให้มีความเร็วรอบที่เท่ากัน เพื่อให้แรงบิดของใบพัดทุกใบหักล้างกัน เพื่อไม่ให้เกิดการหมุนของอากาศยานแบบ 4 ใบพัด

2.4.2.2 การเคลื่อนที่ลงในแนวตั้ง

ควบคุมความเร็วของใบพัดทั้ง 4 ใบ โดยการลดความเร็วรอบของใบพัดทั้ง 4 ใบ ให้มีความเร็วรอบที่เท่ากัน เพื่อให้แรงบิดของใบพัดทุกใบหักล้างกัน เพื่อไม่ให้เกิดการหมุนของอากาศยานแบบ 4 ใบพัด

2.4.2.3 การเคลื่อนที่ไปข้างหน้า

ควบคุมความเร็วของใบพัดคู่ซ้าย-ขวาให้เท่ากัน ในขณะที่ควบคุมความเร็วของใบพัดตัวหน้าให้มีความเร็วที่ต่ำกว่าใบพัดตัวหลัง อากาศยานแบบ 4 ใบพัด จะมีลักษณะเอียงไปด้านหน้าส่งผลให้เกิดการเคลื่อนที่ไปข้างหน้า

2.4.2.4 การเคลื่อนที่ไปข้างหลัง

ควบคุมความเร็วของใบพัดคู่ซ้าย-ขวาให้เท่ากัน ในขณะที่ควบคุมความเร็วของใบพัดตัวหลังให้มีความเร็วที่ต่ำกว่าใบพัดตัวหน้า อากาศยานแบบ 4 ใบพัด จะมีลักษณะเอียงไปด้านหลังส่งผลให้เกิดการเคลื่อนที่ไปข้างหลัง

2.4.2.5 การเคลื่อนที่ไปทางซ้าย

ควบคุมความเร็วของใบพัดคู่หน้า-หลังให้เท่ากัน ในขณะที่ควบคุมความเร็วของใบพัดตัวซ้ายให้มีความเร็วที่ต่ำกว่าใบพัดตัวขวา อากาศยานแบบ 4 ใบพัด จะมีลักษณะเอียงไปด้านซ้ายส่งผลให้เกิดการเคลื่อนที่ไปทางซ้าย

2.4.2.6 การเคลื่อนที่ไปทางขวา

ควบคุมความเร็วของใบพัดคู่หน้า-หลังให้เท่ากัน ในขณะที่ควบคุมความเร็วของใบพัดตัวขวาให้มีความเร็วที่ต่ำกว่าใบพัดตัวซ้าย อากาศยานแบบ 4 ใบพัด จะมีลักษณะเอียงไปด้านขวาส่งผลให้เกิดการเคลื่อนที่ไปทางขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2.7 การเคลื่อนที่แบบหมุนไปทางซ้าย

ควบคุมความเร็วของใบพัดคู่หน้า-หลัง ซึ่งมีทิศทางการหมุนของใบพัดเป็นแบบตามเข็มนาฬิกา ให้มีความเร็วที่สูงกว่าความเร็วของใบพัดคู่ซ้าย-ขวา ซึ่งมีทิศทางการหมุนของใบพัดเป็นแบบทวนเข็มนาฬิกา ส่งผลให้แรงบิดจากใบพัดคู่หน้า-หลังมีทิศทวนเข็มนาฬิกา มีค่ามากกว่าแรงบิดที่เกิดจากใบพัดคู่ซ้าย-ขวาซึ่งมีทิศตามเข็มนาฬิกา จึงทำให้อากาศยาน 4 ใบพัด มีลักษณะหมุนในทิศทวนเข็มนาฬิกาหรือหมุนไปทางซ้าย

2.4.2.8 การเคลื่อนที่แบบหมุนไปทางขวา

ควบคุมความเร็วของใบพัดคู่หน้า-หลัง ซึ่งมีทิศทางการหมุนของใบพัดเป็นแบบตามเข็มนาฬิกา ให้มีความเร็วที่ต่ำกว่าความเร็วของใบพัดคู่ซ้าย-ขวา ซึ่งมีทิศทางการหมุนของใบพัดเป็นแบบทวนเข็มนาฬิกา ส่งผลให้แรงบิดจากใบพัดคู่หน้า-หลังมีทิศทวนเข็มนาฬิกา มีค่าน้อยกว่าแรงบิดที่เกิดจากใบพัดคู่ซ้าย-ขวาซึ่งมีทิศตามเข็มนาฬิกา จึงทำให้อากาศยาน 4 ใบพัด มีลักษณะหมุนในทิศตามเข็มนาฬิกาหรือหมุนไปทางขวา

2.5 บอร์ดควบคุมการบิน ArduPilot Mega 2.5

ตัวบอร์ดควบคุมการบินประกอบไปด้วยไมโครคอนโทรลเลอร์ และเซนเซอร์ที่ช่วยควบคุมการ ดังนี้

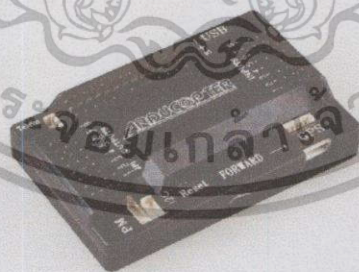
2.5.1 ไมโครคอนโทรลเลอร์ ATMEGA2560

2.5.2 เซ็นเซอร์ไจโรสโคปดิจิตอล และ เซ็นเซอร์วัดความเร็ว 3 MPU-6000

2.5.3 เซ็นเซอร์วัดความดันอากาศ MS5611

2.5.4 เซ็นเซอร์เข็มทิศอิเล็กทรอนิกส์ Honeywell's HMC5883L-TR chip

สำหรับโปรแกรมควบคุมการบินเป็นโปรแกรมแบบ Open Source โดยตัวบอร์ดควบคุมการบิน ArduPilot Mega 2.5 มีรูปร่าง ดังรูปที่ 2.10



รูปที่ 2.9 บอร์ดควบคุมการบิน ArduPilot Mega 2.5 [7]

2.6 ไจโรสโคป

ไจโรสโคป เป็นอุปกรณ์ที่อาศัยแรงเฉื่อยของล้อหมุน เพื่อช่วยรักษาระดับทิศทางของแกนหมุน ประกอบด้วยล้อหมุนเร็วบรรจุอยู่ในกรอบอีกทีหนึ่ง ทำให้เอียงในทิศทางต่างๆได้โดยอิสระ นั่นคือ หมุนในแกนใดๆก็ได้ โมเมนตัมเชิงมุมของล้อดังกล่าว ทำให้มันคงรักษาทิศทางของมันไว้ ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แม้กรอบล้อจะเอียง จากคุณสมบัติดังกล่าวทำให้สามารถทราบถึงสภาวะสมดุลของวัตถุที่ติดตั้งมันไว้ได้ ซึ่งสามารถนำไปประยุกต์ใช้งานในด้านต่างๆได้มากมาย เช่น เข็มทิศ ระบบการบินอัตโนมัติของเครื่องบิน เรือ กลไกบังคับหางเสือของตอร์ปิโด อุปกรณ์ป้องกันการกลิ้งบนเรือใหญ่ และระบบนำร่องเฉื่อย (Inertial Guidance) รวมถึงระบบในยานอวกาศ และสถานีอวกาศ

2.6.1 หลักการทำงานของไจโรสโคป

ทำการแทนที่ขอบด้วย A,B,C,D ทั้งแถบ จุดกลางสุดเป็นแกนคงที่แต่สามารถหมุนได้รอบทิศทาง เมื่อมีแรงกระทำ (Tilting Force) ที่ส่วนบนของแกนของไจโรสโคป จุด A มีการเคลื่อนที่ขึ้นตามแนวตั้ง จุด C เคลื่อนลงตามแนวนอนพร้อมกัน A และ B หมุนไป 90 องศา เช่นเดียวกับที่เกิดกับ C และ D โดยที่ A นั้นยังคงเคลื่อนขึ้นในตำแหน่ง 90 องศา และ C เคลื่อนลง ผลของการเคลื่อนของ A และ C ทำให้แกนของไจโรสโคปหมุนตามการกระทำของ Precession Plane เรียกการเกิดขึ้นของลักษณะนี้ว่า Precession แกนของไจโรสโคปจะหมุนไปทางมุมขวา ถ้าไจโรสโคปถูกทำให้หมุนทวนเข็มนาฬิกา มันก็จะไปทางมุมซ้าย หมายความว่าแรงกระทำตอนต้นเป็นการตั้ง

เมื่อไจโรสโคปหมุนไปอีก 90 องศา จุด C จะมาแทนจุด A ในจุดที่แรงกระทำไปแล้ว ครั้งแรกการเคลื่อนที่ลงของจุด C จะถูกต้านโดย Tilting Force ทำให้แกนของไจโรสโคปไม่เปลี่ยนแปลง ยังมี Tilting Force กระทำมากขึ้น แกนของไจโรสโคปจะมีแรงดึงกลับมามาก เมื่อขอบของแรงจุด A และ C เคลื่อนที่ขึ้นลงเพราะฉะนั้นเมื่อหมุนไจโรสโคปในทิศตรงกันข้ามกับข้างต้นจะเกิดแรงเคลื่อนที่ขึ้นลงมากขึ้น ดังรูปที่ 2.10



รูปที่ 2.10 หลักการทำงานของไจโรสโคป [8]

บางครั้งการเกิดขึ้นของ Precession เป็นสิ่งที่ไม่ต้องการ จึงมักมีการสร้างไจโรสโคปแบบมีแกนที่เรียกว่า Gimbaled Gyroscope ดังรูปที่ 2.12 ซึ่งเป็นไจโรสโคปพื้นฐาน ถูกติดตั้งไว้ในระนาบที่ตั้งฉากกับแนวแรง เมื่อหมุนขอบไปทางระนาบ Gimbal พลังงานจะถ่ายเทไปยังขอบโดย Tilting Force เพื่อหยุดการกระทำนั้น ขอบก็จะหมุนกลับไปในทิศทางระนาบของ Tilting Force โดยที่ไม่มีการเปลี่ยนความเร็วรอบการหมุนของขอบรอบ ๆ แกน

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

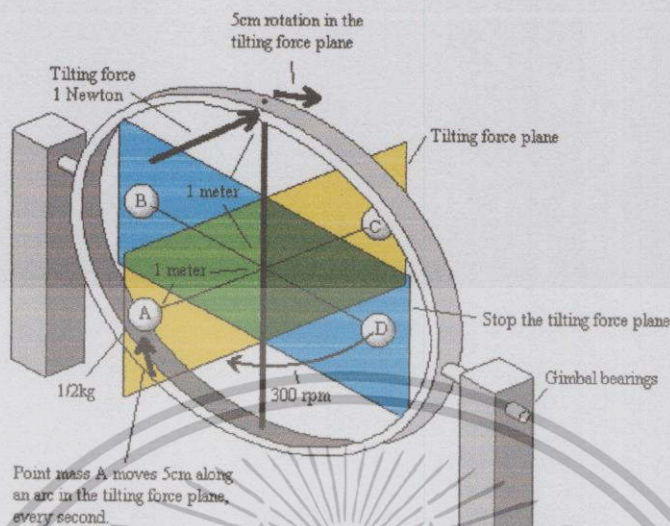


Figure 4

รูปที่ 2.11 ไจโรสโคปแบบมีแกน [8]

2.7 อุปกรณ์ระบุตำแหน่ง (GPS)

อุปกรณ์ระบุตำแหน่ง (GPS) คือ ระบบระบุตำแหน่งบนพื้นโลกย่อมาจากคำว่า Global Positioning System

2.7.1 องค์ประกอบของอุปกรณ์ระบุตำแหน่ง (GPS)

2.7.1.1 ส่วนอวกาศ ประกอบด้วยเครือข่ายดาวเทียม 3 ค่าย คือ

1) อเมริกา ชื่อ นาวาสตาร์ (NAVSTAR-Navigation Satellite Timing and Ranging GPS) มีดาวเทียม 28 ดวง ใช้งานจริง 24 ดวง อีก 4 ดวงเป็นตัวสำรอง มีรัศมีวงโคจรจากพื้นโลก 20,162.81 กม. หรือ 126,00 ไมล์ ดาวเทียมแต่ละดวงใช้เวลาในการโคจรรอบโลก 12 ชั่วโมง

2) ยุโรป ชื่อ Galileo มี 27 ดวง

3) รัสเซีย ชื่อ GLONASS หรือ Global Navigation Satellite

2.7.1.2 ส่วนควบคุม

ประกอบด้วย สถานีภาคพื้นดิน สถานีใหญ่อยู่ที่ Falcon Air Force Base ประเทศอเมริกา และศูนย์ควบคุมย่อยอีก 5 จุด กระจายไปยังภูมิภาคต่างๆ ทั่วโลก

2.7.1.3 ส่วนผู้ใช้งาน

ผู้ใช้งานต้องมีเครื่องรับสัญญาณที่สามารถรับคลื่น และแปลรหัสจากดาวเทียมเพื่อนำมาประมวลผลให้เหมาะสมกับการใช้งานในรูปแบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 การทำงานของจีพีเอส

ดาวเทียมจีพีเอส (นาฬิกาฟอสตาร์) ประกอบด้วยดาวเทียม 24 ดวง โดยแบ่งเป็น 6 รอบวงโคจร การโคจรทำมุมเอียง 55 องศากับเส้นศูนย์สูตร (Equator) ในลักษณะสานกันคล้ายลูกตะกร้อ แต่ละวงโคจรมีดาวเทียม 4 ดวง

จีพีเอส ทำงานโดยการรับสัญญาณจากดาวเทียมแต่ละดวง โดยสัญญาณดาวเทียมนี้ประกอบด้วย ข้อมูลที่ระบุตำแหน่งและเวลาขณะส่งสัญญาณ ตัวเครื่องรับสัญญาณ จีพีเอสจะต้องประมวลผลความแตกต่างของเวลาในการรับสัญญาณ เทียบกับเวลาจริง ณ ปัจจุบันเพื่อแปรเป็นระยะทางระหว่างเครื่องรับสัญญาณกับดาวเทียมแต่ละดวง ซึ่งได้ระบุตำแหน่งของมันมากับสัญญาณดังกล่าวข้างต้น

เพื่อให้เกิดความแม่นยำในการค้นหาตำแหน่งด้วยดาวเทียม ต้องมีดาวเทียมอย่างน้อย 4 ดวง เพื่อบอกตำแหน่งบนผิวโลก ซึ่งระยะห่างจากดาวเทียมทั้ง 3 กับเครื่องจีพีเอส (จุดตรงกลาง) สามารถระบุตำแหน่งบนผิวโลกได้หากพื้นโลกอยู่ในแนวระนาบ แต่ในความเป็นจริงพื้นโลกมีความโค้งเนื่องจากสัณฐานของโลกมีลักษณะกลม ดังนั้นดาวเทียมดวงที่ 4 จะทำให้สามารถคำนวณเรื่องความสูงเพื่อให้ได้ตำแหน่งที่ถูกต้องมากขึ้น

2.7.3 การประยุกต์ใช้งาน

ปัจจุบันนี้ได้มีการใช้งานจีพีเอสในรูปแบบต่างๆ ดังนี้

2.7.3.1 การกำหนดพิกัดของสถานีต่างๆ การทำแผนที่ งานสำรวจ โดยส่วนใหญ่นิยมใช้อุปกรณ์ที่สามารถพกพาได้ง่าย มีความทนทาน ถิ่นน้ำได้ สามารถใช้ได้กับถ่านไฟฉายขนาดมาตรฐานได้

2.7.3.2 การวางแผนการใช้ประโยชน์ที่ดิน เช่น โครงข่ายหมุนดาวเทียมจีพีเอสของกรมที่ดิน

2.7.3.3 การกำหนดจุดเพื่อบรรเทาสาธารณภัย เช่น เสื้อกั๊กชูชีพที่มีเครื่องส่งสัญญาณจีพีเอส

2.7.3.4 การนำไปใช้ประโยชน์ทางทหาร

2.7.4 การทำงานของระบบนำทางจีพีเอส

ก่อนอื่นผู้ใช้งานจะต้องมีเครื่องรับสัญญาณดาวเทียมหรือมีอุปกรณ์นำทาง เมื่อผู้ใช้นำเครื่องไปใช้งานจะมีการเปิดรับสัญญาณจีพีเอส แล้วตัวโปรแกรมจะแสดงตำแหน่งปัจจุบันบนแผนที่แผนที่สำหรับนำทางจะเป็นแผนที่พิเศษที่มีการกำหนดทิศทางจราจร เช่น การจราจรแบบชิดซ้ายหรือชิดขวา ข้อมูลการเดินทางเดียว จุดสำคัญต่างๆ ข้อมูลทางภูมิศาสตร์ต่างๆ ฝั่งไว้ในข้อมูลแผนที่ที่ได้ทำการสำรวจและตั้งค่าไว้แล้ว ในแต่ละทางแยกก็จะมีการกำหนดค่าเอาไว้ด้วยเช่นกัน เพื่อให้ตัวโปรแกรมทำการเลือกการเชื่อมต่อของเส้นทางจนถึงจุดหมายที่ได้เลือกไว้ เสียงนำทางก็จะทำงานสอดคล้องกับการเลือกเส้นทาง เช่น ถ้าโปรแกรมเลือกเส้นทางที่จะต้องไปทางขวาก็จะกำหนดให้มีการแสดงเสียงเตือนให้เลี้ยวขวา โดยแต่ละโปรแกรมก็จะมีการกำหนดเตือน

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไว้ล่วงหน้าว่าจะเตือนก่อนจุดเลี้ยว เท่าใด ส่วนการแสดงทิศทางก็จะมีการบอกไว้ล่วงหน้าเช่นกัน แล้วแต่ว่าจะกำหนดไว้ล่วงหน้ากี่จุด บางโปรแกรมก็กำหนดไว้จุดเดียว บางโปรแกรมกำหนดไว้สองจุด หรือบางโปรแกรมก็สามารถเลือกการแสดงได้ตามความต้องการของผู้ใช้ การคำนวณเส้นทางนี้จะถูกคำนวณให้ตั้งแต่แรก และตัวโปรแกรมจะแสดงผลทั้งภาพ และเสียงตามตำแหน่งจริงที่อยู่ ณ จุดนั้นๆ หากมีการเดินทางออกนอกเส้นทางที่ได้กำหนดไว้ เครื่องจะทำการเตือนให้ผู้ใช้ทราบ และจะคำนวณให้พยายามกลับสู่เส้นทางที่ได้วางแผนไว้ก่อน หากการออกนอกเส้นทางอยู่เกินกว่าค่าที่กำหนดไว้ก็จะมีการคำนวณเส้นทางใหม่เองอัตโนมัติ

2.7.5 อุปกรณ์ที่ใช้ร่วมกับการนำทางด้วยจีพีเอส

อุปกรณ์ที่ใช้ร่วมกับการนำทางด้วยจีพีเอส ประกอบด้วยตัวรับสัญญาณหน่วยประมวลผล โปรแกรม และข้อมูลแผนที่ ปัจจุบันนี้มีเครื่องจีพีเอสที่มีอุปกรณ์ครบทุกอย่างในตัวเอง ซึ่งจะมีความสะดวกในการใช้งานและมีความเสถียรสูงได้แก่ ระบบนำทางส่วนบุคคล (Personal / Portable Navigation Device) หรือแบบที่ใช้เครื่องรับจีพีเอสร่วมกับ PDA (Personal Digital Assistant), โน้ตบุ๊ก, สมาร์ทโฟน เป็นต้น นอกจากอุปกรณ์หลักแล้วยังมีอุปกรณ์เสริม เช่น เสารับสัญญาณภายนอกแบบติดตั้งเฉพาะเครื่องต่อเครื่อง หรือตัวกระจายคลื่น (GPS radiator) เพื่อให้สามารถใช้จีพีเอสได้ในที่อับสัญญาณ เช่น ในรถที่ติดฟิล์มที่มีสารโลหะอยู่ (หรือที่เรียกกันว่า ฉาบปรอท) หรือในอาคาร

2.7.6 ปัจจัยที่มีผลต่อความถูกต้องของตำแหน่ง

ความถูกต้องของตำแหน่งที่หาได้จากระบบที่กีดดาวเทียมนั้น มีปัจจัยที่เกี่ยวข้องจำนวนมาก เช่น

2.7.6.1 จำนวนดาวเทียม จำนวนยิ่งมากยิ่งมีโอกาสที่จะได้ความถูกต้องที่สูงขึ้นจากการวิเคราะห์ตำแหน่ง

2.7.6.2 ตำแหน่งและการเรียงตัวของดาวเทียม (satellite configuration) (ซึ่งสามารถสังเกตได้จากค่าการลดสัดส่วนของความแม่นยำ DOP (Dilution of Precision)

2.7.6.3 ชนิดของสัญญาณที่นำมาใช้วิเคราะห์ (code หรือ phase หรือทั้งสองอย่าง)

2.7.6.4 จำนวนสัญญาณคลื่นความถี่ (ความถี่เดียว หรือ ความถี่คู่ หรือมากกว่า)

2.7.6.5 วิธีการวิเคราะห์ (วิเคราะห์ตำแหน่งแบบเชิงเดี่ยว (single precise point positioning) หรือ ตำแหน่งสัมพัทธ์ (relative positioning)

2.7.6.6 เทคนิคการขจัดผลกระทบเนื่องจากชั้นไอโอโนสเฟียร์ (ionosphere เป็นชั้นอากาศเบาบาง ที่ประกอบด้วยแก๊สที่แตกตัวเป็นประจุไฟฟ้าบวกและลบ)

2.7.6.7 เทคนิคการประมาณผลกระทบจากโทรโปสเฟียร์ (troposphere เป็นชั้นอากาศที่เราอาศัยอยู่)

บริการใช้งานเพื่อการศึกษานั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.7 ข้อมูลเอ็นเอ็มอีเอ

NMEA ย่อมาจาก Nation Maritime Electronics Association ซึ่งเป็นสมาคมที่มุ่งเน้นศึกษาและพัฒนาอุปกรณ์อิเล็กทรอนิกส์ เพื่อการเชื่อมต่อและทำงานร่วมกันของอุปกรณ์โดยอุปกรณ์เหล่านี้ เมื่อเชื่อมต่อและทำงานร่วมกันต้องสามารถเข้าใจกันได้ หรือสื่อสารโดยใช้ภาษาเดียวกัน NMEA จึงพัฒนามาตรฐานในการสื่อสารข้อมูลระหว่างอุปกรณ์ดังกล่าว เรียกว่า NMEA Standard ซึ่งระบุข้อมูลการเชื่อมต่อทางไฟฟ้าและรูปแบบของข้อมูล

มาตรฐาน NMEA-0183 ในมาตรฐานนี้ ตัวอักษรที่ใช้คือ ASCII Text ซึ่งสามารถพิมพ์ได้ (รวมถึง Carriage Return and Line Feed) NMEA-0183 ส่งข้อมูลด้วยอัตรา 4800 baud ข้อมูลจะถูกส่งในรูปของประโยค (Sentences)

โปรโตคอลที่สำคัญของเครื่อง GPS จะอยู่ในชุด NMEA ซึ่งเป็นโปรโตคอลมาตรฐานของ GPS แต่ภายในอาจมีโปรโตคอลอื่นประกอบอีกมากมาย แต่โปรโตคอลที่ใช้งานหลักมีดังนี้

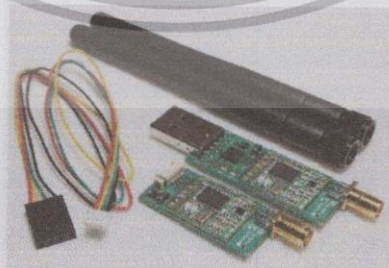
2.7.7.1 GGA รูปแบบที่แสดงว่าข้อมูลของ GPS เพียงพอที่จะแสดงพิกัดได้สามมิติ (3D) ซึ่งดาวเทียมที่รับได้ต้องมากถึง 4 ดวงขึ้นไป ภาษาอังกฤษเรียกว่า Fix data

2.7.7.2 GSA รูปแบบที่แสดงรายละเอียดของข้อมูล Fix จำนวนดาวเทียมที่ใช้งานได้ รวมถึงค่าความคลาดเคลื่อน DOP (dilution of precision) ซึ่งตัวเลขน้อยๆ จะเป็นค่าที่ดีมีความถูกต้องสูง

2.7.7.3 GSV รูปแบบที่แสดงรายละเอียดของ GPS แต่ละดวงเช่นระดับความสูง (Elevation) อะซิมัทและ SNR (Signal to Noise Ratio) ซึ่ง เทียบได้กับความแรงของสัญญาณ SNR มีค่าตั้งแต่ 0 ถึง 99 ซึ่งค่ามากเป็นค่าที่ดี ในบางขณะเครื่อง GPS อาจจะได้รับสัญญาณได้เต็มที่ทั้งหมด 12 ดวง การส่งข้อความจะมากไปถ้าต้องแสดงในบรรทัดเดียว สมาคม NMEA จึงออกแบบให้รูปแบบ GSV สามารถแสดงข้อมูลดาวเทียมได้เต็มที่ประโยคหรือบรรทัดละ 4 ดวงเท่านั้น ดังนั้นถ้ารับสัญญาณดาวเทียมได้ทั้ง 12 ดวงจะได้รับประโยคทั้งหมด 3 บรรทัด

2.7.7.4 RMC รูปแบบที่แสดงรายละเอียดของ GPS เรื่องความเร็ว (velocity) ค่าพิกัด เวลา ตลอดจนทิศทาง

2.8 ระบบโทรมาตร (Telemetry)



รูปที่ 2.12 433Mhz Radio Telemetry Kit [6]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Telemetry System เป็นเทคนิคที่ใช้ในการส่งและรับข้อมูลผ่านสื่อกลาง โดยข้อมูลนั้นสามารถวัดได้ เช่น โวลต์ ความเร็ว หรืออัตราการไหล ซึ่งข้อมูลเหล่านี้จะถูกส่งไปอีกสถานที่หนึ่ง โดยผ่านสื่อกลางต่าง ๆ เช่น เคเบิล สายโทรศัพท์ หรือคลื่นวิทยุ

คุณสมบัติ:

- มีขนาดเล็ก และน้ำหนักเบา
- ความไวในการรับสัญญาณ -121 dBm
- กำลังในการส่งสัญญาณ 20 dBm (100mW)
- รองรับ LBT และ AFA
- ปรับค่า duty cycle ได้
- มีการทำ Error correcting code
- ใช้ AT Command ในการปรับค่าสัญญาณวิทยุ
- ใช้ RT Command สำหรับปรับค่าสัญญาณวิทยุระยะไกล
- ใช้หลักการของ HM-TRP ไม่ตุล

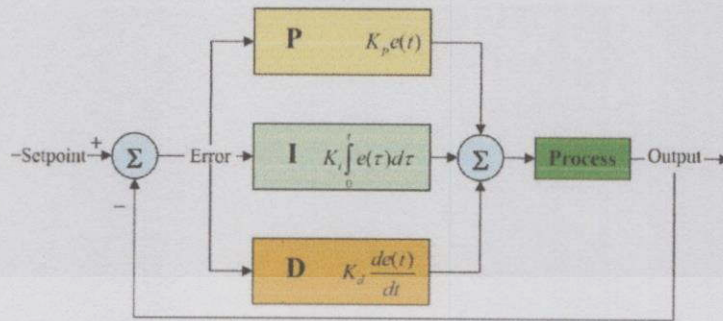
2.9 PID Control

คอนโทรลเลอร์แบบ PID (PID controller) จากชื่อจะเห็นได้ชัดว่ามีสองส่วนคือ คอนโทรลเลอร์ และ PID ส่วนคอนโทรลเลอร์ก็คืออุปกรณ์ในการควบคุมระบบตามที่ต้องการ โดยจะมีการตั้งค่า SP (Set Point: เซตพอยต์ หรือค่าที่ต้องการ) และนำมาเปรียบเทียบกับค่า PV (Process Variable: ตัวแปรกระบวนการ หรือค่าจริงที่เกิดจากผลการทำงานของระบบ) เพื่อให้ได้ค่า Error แล้วคอนโทรลเลอร์จะนำค่า Error นั้นมาทำการปรับปรุงแต่งค่าเอาต์พุต หรือ MV (Manipulated Variable: ตั้งแปรที่ถูกควบคุม บางทีอาจจะเรียกว่า CV (Control Variable: ตัวแปรในการคอนโทรล)) เพื่อลดค่า Error ให้ได้ต่ำที่สุด ส่วน PID ย่อมาจาก Proportional-Integral-Derivative แปลเป็นไทยได้ว่า อัตราส่วน-อินทิกรัล-อนุพันธ์ โดย

- 1) Proportional เป็นส่วนปฏิกิริยาต่อ Error ณ ขณะนั้น (Current Error)
- 2) Integral เป็นส่วนปฏิกิริยาที่เกิดจากผลรวมของค่า Error ที่ผ่านมาล่าสุด (Sum of recent errors)

3) Derivative เป็นส่วนปฏิกิริยาที่เกิดจากอัตราการเปลี่ยนแปลงของค่า Error (Rate at which the error has been changing) ผลรวมตามน้ำหนัก (Weighted sum) ของทั้งสามซึ่งปรับแต่งโดยค่าคงที่ K_p , K_i และ K_d เพื่อให้ได้การตอบสนองที่ต้องการ จะถูกนำไปใช้ในการควบคุมกระบวนการต่าง ๆ เช่น รอบเครื่อง หรือเพลลา, ตำแหน่งวาล์ว หรืออุปกรณ์ทำความร้อน โดยทั่วไปค่า Gain บางค่าอาจเป็นศูนย์ก็ได้ ซึ่งคอนโทรลเลอร์ PID นั้นจะกลายเป็นคอนโทรลเลอร์แบบ PI, PD, P หรือ I ขึ้นอยู่กับว่าเทอมไหนหายไป

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 Block diagram ของคอนโทรลเลอร์แบบPID [10]

2.9.1 ทฤษฎีคอนโทรลเลอร์แบบ PID

การควบคุมแบบ PID นั้นประกอบด้วยเทอมสำหรับปรับแต่ง 3 เทอมรวมกันเป็น MV

$$MV(t) = P_{out} + I_{out} + D_{out} \quad (2.1)$$

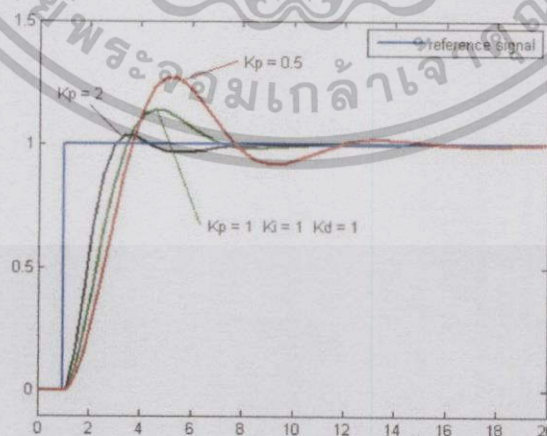
โดยที่ P_{out} , I_{out} และ D_{out} เป็นเอาต์พุตจากแต่ละเทอมตามลำดับ

2.9.1.1 เทอม Proportional

เทอม Proportional เปลี่ยนแปลงตามอัตราส่วนของค่า Error ปัจจุบัน ซึ่งค่า P_{out} สามารถคำนวณได้โดยการนำค่า Error มาคูณกับ ค่าคงที่ K_p

$$P_{out} = K_p e(t) \quad (2.2)$$

ค่า K_p ที่สูง จะเป็นผลให้ค่าเอาต์พุตมากขึ้นตาม หากค่า K_p มากเกินไป ระบบอาจจะไม่เสถียรได้ ในทางตรงข้ามหากค่า K_p น้อยเกินไปอาจทำให้ระบบตอบสนองช้าเกินไป ซึ่งในกรณีนี้เป็นไปได้ว่า การตอบสนองดังกล่าว อาจจะมีสิ่งรบกวน (Disturbance) ในระบบได้ทันการ เช่น ในการเร่งรอบเครื่องยนต์ เมื่อมีโหลดมากกระทำกับเครื่อง หากคอนโทรลเลอร์เร่งรอบเครื่องช้า ไม่ทันกับโหลดที่เพิ่มขึ้นมาทันที รอบเครื่องจะค่อย ๆ ตกลง และเครื่องก็จะดับในที่สุด



เอกสารนี้รูปที่ 2.14 การเปลี่ยนแปลงค่า PV เทียบกับ เวลา โดยเปลี่ยนค่า K_p (ค่า K_i และ K_d คงที่) [10] ด้านการถ้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่ไม่มี Disturbance การใช้เทอม Proportional อย่างเดียว จะทำให้ระบบเกิดการ Oscillate รอบ ๆ SP จะไม่อยู่ที่ SP พอดี

2.9.1.2 เทอม Integral

เทอม Integral เป็นอัตราส่วนของค่า Error สะสมในหนึ่งช่วงเวลา (ปัจจุบัน ย้อนไปในอดีต) ค่า I_{out} เกิดจากผลคูณของค่าคงที่ K_i กับผลรวมของ $e(t)$ ซึ่งเป็นค่าสะสมของ Error ที่ควรจะต้องถูกแก้ไขมาก่อนหน้านี้

$$I_{out} = K_i \int_0^t e(\tau) d\tau \quad (2.3)$$

เทอม Integral (เมื่อใช้ร่วมกับเทอม Proportional) จะช่วยเร่งให้ระบบ วิ่งเข้าหา SP เร็วขึ้น และช่วยลด Error ที่เกิดจากการใช้เทอม Proportional อย่างเดียว อย่างไรก็ตาม เนื่องจากว่า เทอม Integral เกิดจากการคำนวณโดยรวม Error ที่เกิดขึ้นในอดีตด้วย อาจจะทำให้เกิด Overshoot เกิน SP ในค่าปัจจุบันด้วย



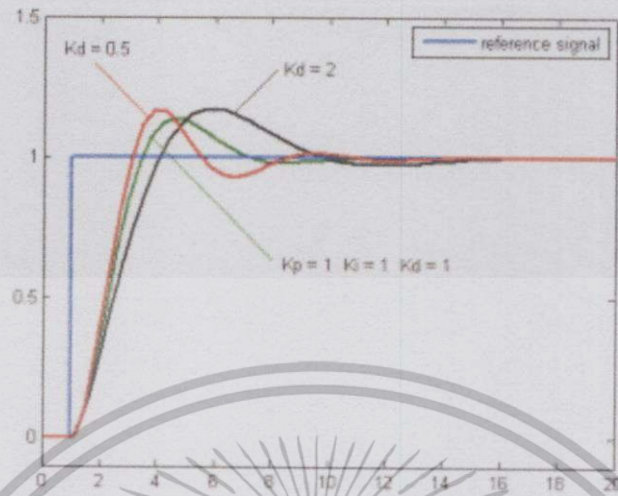
รูปที่ 2.15 การเปลี่ยนแปลงค่า PV เทียบกับเวลา โดยเปลี่ยนค่า K_i (ค่า K_p และ K_d คงที่) [10]

2.9.1.3 เทอม Derivative

อัตราการเปลี่ยนแปลงของค่า Error ในโปรเซสส์นั้นคำนวณได้โดยสโลป (Slope) ของกราฟ Error เทียบกับ เวลา ซึ่งก็คืออนุพันธ์แรก เทียบกับเวลา และคูณค่าสโลปนี้กับค่าคงที่ K_d ก็จะได้เทอม Derivative

$$D_{out} = K_d \frac{dc}{dt}(t) \quad (2.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 การเปลี่ยนแปลงค่า PV เทียบกับเวลา โดยเปลี่ยนค่า K_d (ค่า K_p และ K_i คงที่) [10]

เทอม Derivative ช่วยดึงเอาต์พุตจากคอนโทรลเลอร์ให้ช้าลง ซึ่งจะเห็นได้ชัดเมื่อโปรเซสเข้าใกล้ SP ฉะนั้นเทอม Derivative จึงช่วยลดอาการ Overshoot ซึ่งเกิดจากเทอม Integral และช่วยปรับปรุงเสถียรภาพของระบบโดยรวม อย่างไรก็ตาม ค่า Derivative ค่อนข้างไวต่อ Noise ซึ่งอาจทำให้ระบบไม่เสถียรได้ หาก Noise และค่า K_p สูงเกินไป

2.9.2 รวมสามเทอมเข้าด้วยกัน

เมื่อรวมเทอม Proportional, Integral และ Derivative เข้าด้วยกันจะได้เอาต์พุตจากคอนโทรลเลอร์ PID ดังนี้

$$MV(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de}{dt}(t) \quad (2.5)$$

โดยค่าพารามิเตอร์ต่าง ๆ คือ

2.9.2.1 ค่า Proportional Gain (K_p): ค่าที่มากขึ้น หมายถึงการตอบสนองที่เร็วขึ้น เพราะค่า Error ยิ่งมาก ค่าชดเชยจากเทอมนี้ก็จะมากขึ้นตามค่า Gain และค่า Gain ที่มากเกินไปจะนำไปสู่โปรเซสที่ไม่เสถียร และการแกว่ง (Oscillate)

2.9.2.2 ค่า Integral Gain (K_i): ค่าที่มากขึ้น หมายถึง ค่า Error แบบ Steady-State จะถูกกำจัดได้เร็วขึ้น ข้อเสียคือ Overshoot ค่า error ที่เป็นลบจะต้องถูกแก้ด้วย Error ที่เป็นบวกก่อน และระบบจะเข้าสู่ Steady-State

2.9.2.3 ค่า Derivative Gain (K_d): ค่าที่มากขึ้น หมายถึงขนาด Overshoot ที่ลดลง แต่ก็อาจทำให้การตอบสนองช้าลงบ้าง และอาจนำไปสู่ความไม่เสถียรของระบบเนื่องจาก Noise ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลของการเพิ่มค่าพารามิเตอร์ของคอนโทรลเลอร์ PID				
พารามิเตอร์	Rise time	Overshoot	Settling time	Error at equilibrium
K_p	ลดลง	เพิ่มขึ้น	เปลี่ยนเล็กน้อย	ลดลง
K_i	ลดลง	เพิ่มขึ้น	เพิ่มขึ้น	ถูกกำจัด
K_d	นานมาก (ลดลงหรือเพิ่มขึ้น น้อยมาก)	ลดลง	ลดลง	ไม่มี

รูปที่ 2.17 ตารางผลของการเพิ่มค่าพารามิเตอร์ของคอนโทรลเลอร์ PID [10]

2.9.3 การปรับแต่งค่า (Loop Tuning)

ในการปรับแต่งค่ามี 3 วิธีหลักดังนี้

2.9.3.1 การปรับแต่งด้วยมือ (Manual Tuning) จะเริ่มโดยการเซตค่า K_i และ K_d เป็นศูนย์ และค่อยๆ เพิ่มค่า K_p เรื่อยๆ จนกระทั่งระบบเริ่มเกิดการแกว่ง จากนั้นให้เซตค่า K_p เป็นครึ่งหนึ่งของค่านั้น จากนั้นให้เริ่มเพิ่มค่า K_i จนได้เวลาการตอบสนองของระบบที่ต้องการ จากนั้นหากจำเป็น เพิ่มค่า K_d จนกระทั่งการควบคุมเร็วพอที่ยอมรับได้ โดยเทียบกับระบบที่มี Disturbance อ้างอิง

2.9.3.2 การปรับแต่งด้วยวิธี Ziegler-Nichols จะเริ่มด้วยการเซตค่า K_i และ K_d เป็นศูนย์ และค่อยๆ เพิ่มค่า K_p ไปจนถึงค่า K_c (Critical Gain) ระบบจะเริ่มแกว่ง ให้วัดคาบของการแกว่ง P_c และให้ใช้ตารางข้างล่างเพื่อหาค่า Gain อื่นๆ

วิธี Ziegler-Nichols			
Control Type	K_p	K_i	K_d
P	$0.50 K_c$	—	—
PI	$0.45 K_c$	$1.2 K_p / P_c$	—
PID	$0.60 K_c$	$2 K_p / P_c$	$K_p P_c / 8$

รูปที่ 2.18 ตารางแสดงค่า Gain ของวิธี Ziegler-Nichols [10]

2.9.3.3 การปรับแต่งด้วยซอฟต์แวร์ ในอุตสาหกรรมส่วนใหญ่ในปัจจุบันจะใช้ซอฟต์แวร์ในการคำนวณ และปรับแต่งค่า Gain โดยซอฟต์แวร์จะทำการเก็บข้อมูล สร้างโมเดล และคำนวณค่า Gain ให้โดยอัตโนมัติ ซึ่งงานเพื่อการศึกษานี้ ไม่น่าจะเหมาะให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.4 ข้อจำกัดของคอนโทรลเลอร์แบบ PID

ในกรณีที่คอนโทรลเลอร์ทำงานได้ไม่ดี เกิดอาการฮันต์ อาจจะต้องใช้รวมคอนโทรลเลอร์แบบ Feed forward ด้วย โดยการบวกค่า Bias เพิ่มเข้าไปในสมการ $MV(t)$

เนื่องจากคอนโทรลเลอร์ PID เป็นแบบเชิงเส้น (Linear) ในกรณีที่ใช้กับระบบที่ไม่เป็นเชิงเส้น (Non-linear) เช่น ระบบ HVAC (Heating – Ventilating – Air – Condition หรือบางครั้งเรียกว่า Climate control) อาจจะต้องใช้การปรับค่า Gain ตามตารางเวลา หรือใช้คอนโทรลเลอร์แบบ Fuzzy logic แทน

2.10 Android Studio

Android Studio นี้ มีส่วนสำคัญในการพัฒนาแอป Android และรองรับกับ Android 5 Lollipop ซึ่งเป็นเวอร์ชันล่าสุดอย่างสมบูรณ์ด้วย ซึ่ง Google ได้แนะนำผู้ที่เคยใช้ Eclipse ในการพัฒนาแอป ให้ปรับเปลี่ยนมาใช้ Android Studio ในการพัฒนา ซึ่ง Android Studio เสถียรกว่า Eclipse ในหลายด้าน



รูปที่ 2.19 โปรแกรม Android Studio 1.1.0

2.10.1 ข้อดีของ Android Studio

2.10.1.1 สนับสนุนการติดตั้งหลากหลายระบบปฏิบัติการ (Platform) Windows, Mac OS X และ Linux

2.10.1.2 เครื่องมือช่วยพัฒนาการออกแบบส่วนติดต่อกับผู้ใช้ และสามารถพรีวิวให้ดูแบบหลายๆขนาดหน้าจอได้พร้อม เป็นแบบ Live Preview

2.10.1.3 ระบบการคาดการณ์การพิมพ์โค้ดที่ชาญฉลาด

2.10.1.4 ลดขั้นตอนที่เคยพัฒนาใน eclipse

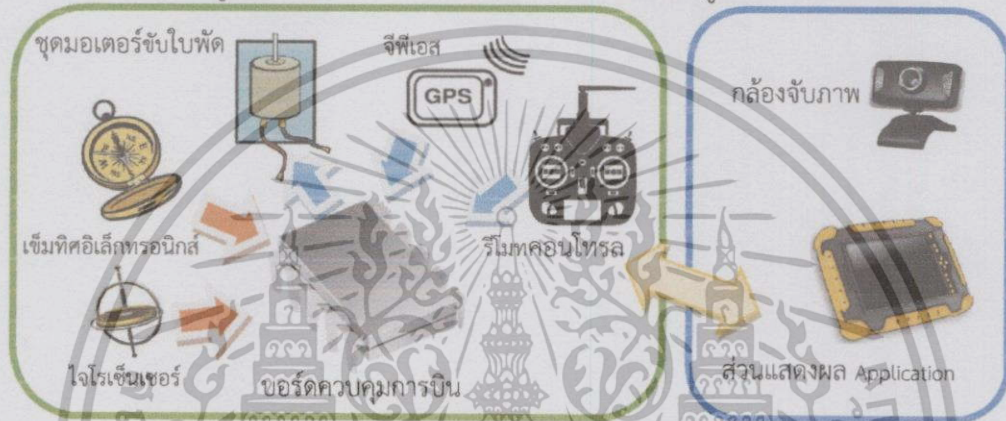
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบ และการจัดทำปริญญานิพนธ์

3.1 การออกแบบ

โครงสร้างนี้จะถูกติดอยู่กับตัวอากาศยาน 4 ใบพัดแบบไร้คนขับ จะมีส่วนประมวลผลทั้งหมด และจัดการข้อมูลต่าง ๆ โดยจะมีโครงสร้างโดยรวมแสดงดังรูปที่ 3.1

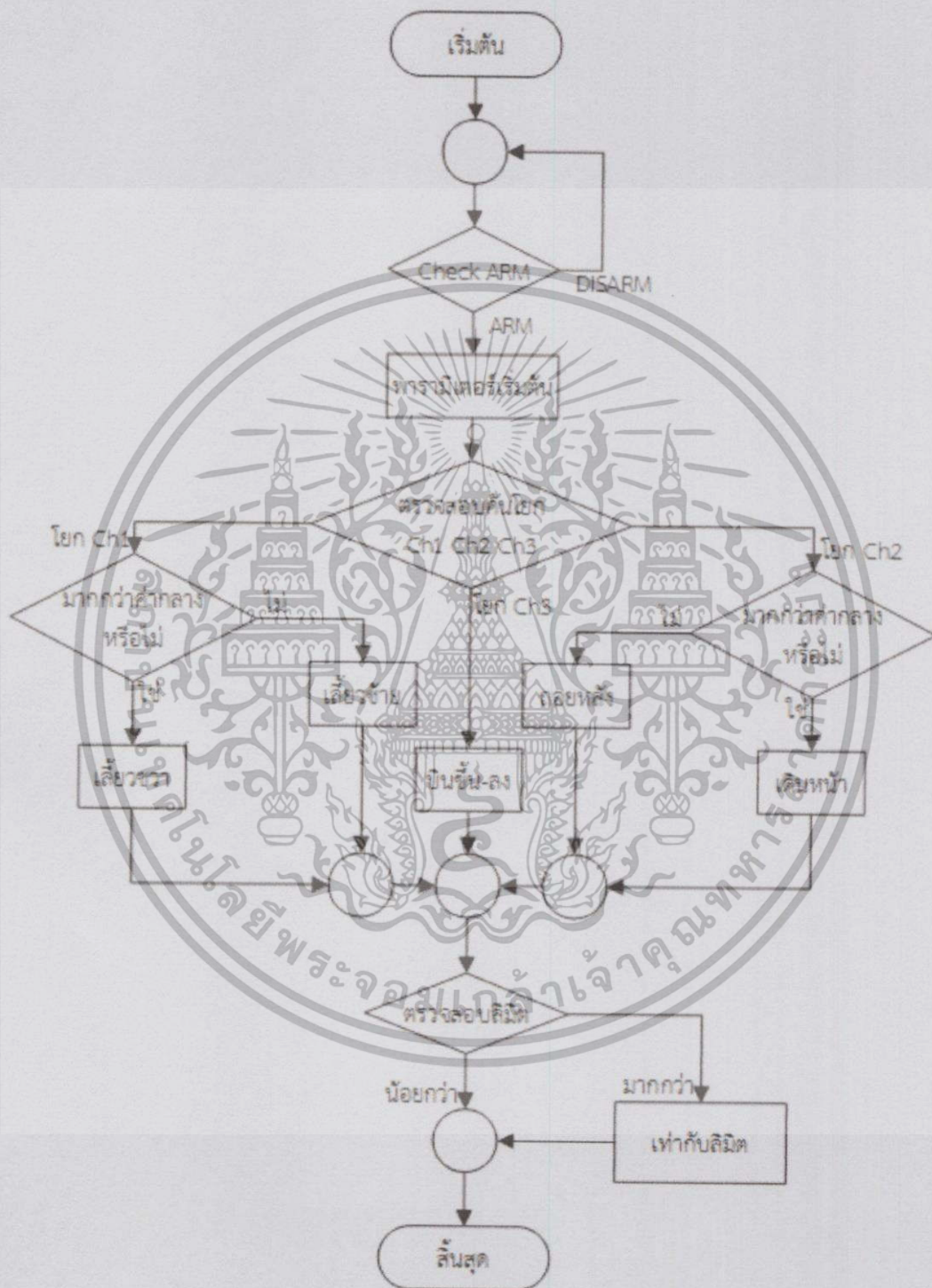


รูปที่ 3.1 โครงสร้างการทำงานของระบบ

หลักการทำงานของระบบควบคุมอากาศยาน 4 ใบพัด ซึ่งเป็นการนำเอาอุปกรณ์ระบุตำแหน่ง (GPS) มาประยุกต์ใช้งานร่วมกับบอร์ดควบคุมการบิน APM 2.5 ซึ่งบนตัวบอร์ดประกอบด้วยเซ็นเซอร์ต่าง ๆ มากมาย อาทิเช่น ไจโรเซ็นเซอร์ เข็มทิศอิเล็กทรอนิกส์ บาร์โรมิเตอร์ เป็นต้น โดยมีโปรแกรมควบคุมการใช้งานเซ็นเซอร์ต่าง ๆ ที่ต่อเข้ากับสปีดคอนโทรล (Speed Control) เพื่อใช้ควบคุมความเร็วมอเตอร์แต่ละตัวในการเคลื่อนที่ของอากาศยาน 4 ใบพัดในรูปแบบต่าง ๆ และทำการติดตั้งกล้องสำหรับถ่ายภาพบนอากาศยาน 4 ใบพัด เพื่อบันทึกภาพตลอดการบินในขณะที่อากาศยานเคลื่อนที่ผ่าน ซึ่งตัวอากาศยานจะเชื่อมต่อกับแอปพลิเคชันบนโทรศัพท์มือถือระบบปฏิบัติการแอนดรอยด์เพื่อกำหนดพิกัดปลายทางที่จะให้อากาศยาน 4 ใบพัดไปบันทึกภาพได้

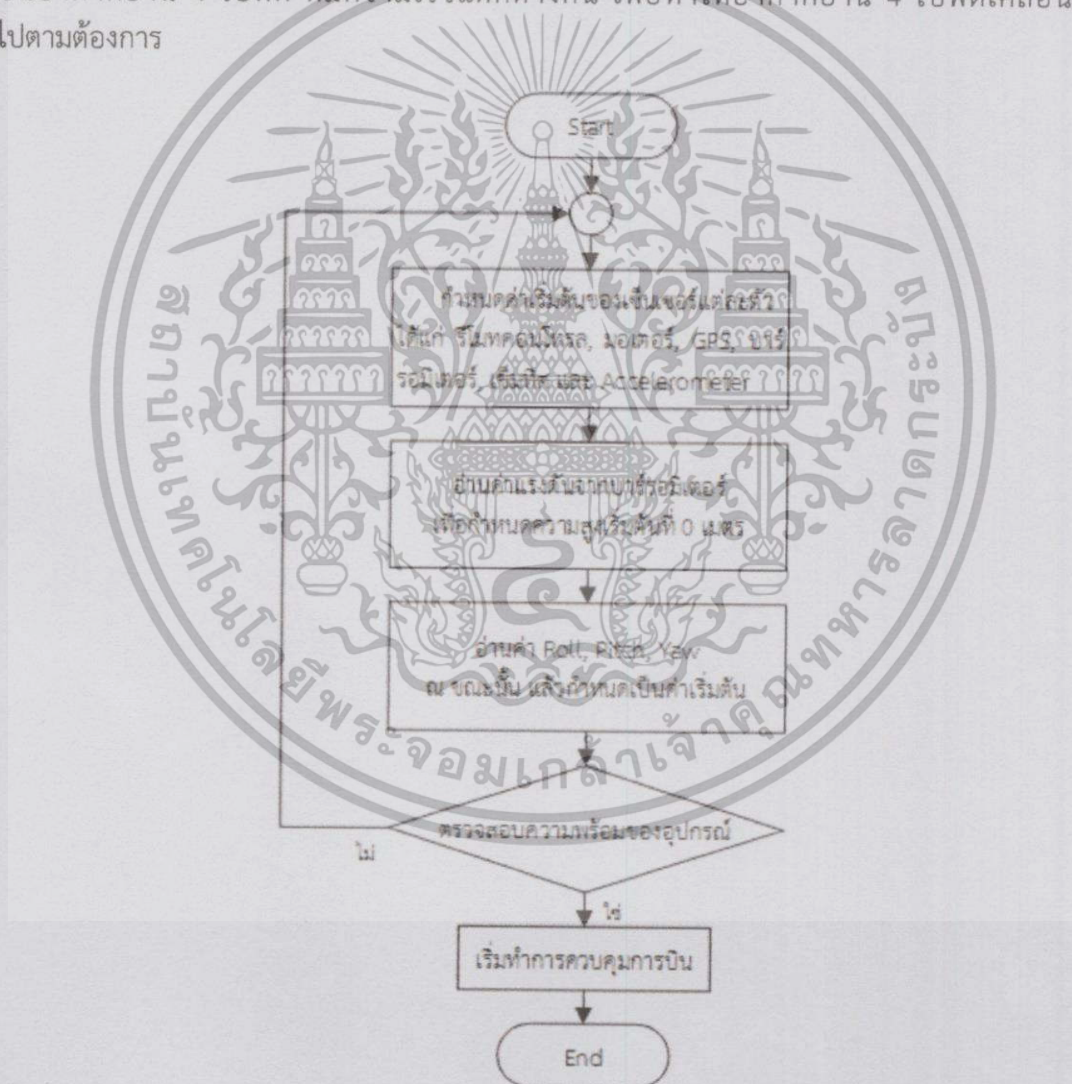
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 การทำงานของโปรแกรมรวม



รูปที่ 3.2 ผังการทำงานของส่วนโปรแกรม (ใช้รีโมทในการควบคุมการบิน) ประโยชน์ด้านการค้า เอกสารนี้เป็นเอกสารลับ ไม่ควรเผยแพร่ให้ผู้อื่นได้รับรู้ หากมีการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาต จะถือว่าผิดกฎหมาย

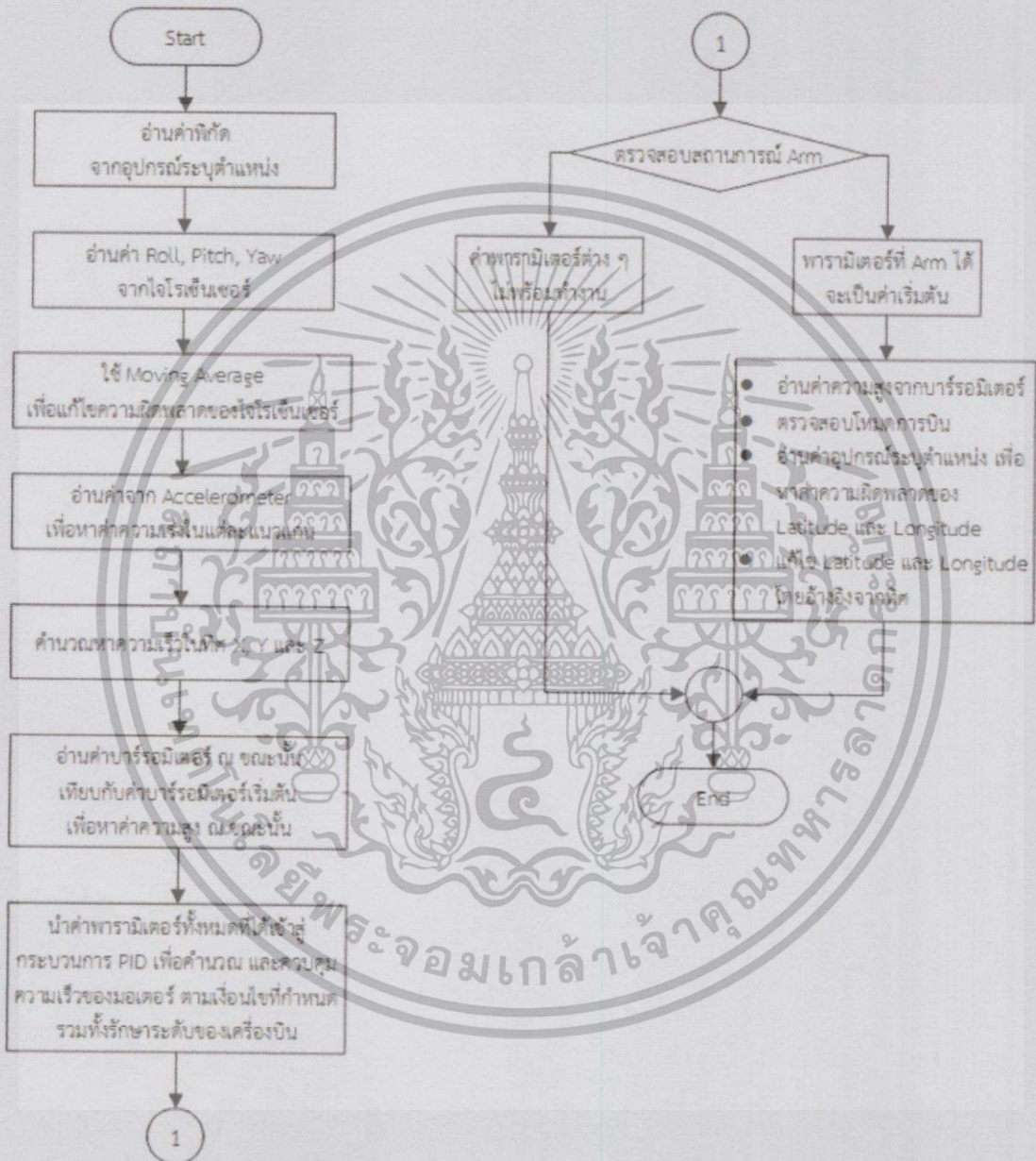
จากรูปที่ 3.2 เป็นภาพผังการทำงานของโปรแกรมควบคุมการบินแบบใช้รีโมทบังคับ ซึ่งเริ่มจากการรอรับสถานะพร้อมใช้งานของรีโมทคอนโทรลกับอากาศยาน 4 ใบพัด (สถานะ ARM) เมื่อพร้อมใช้งานบอร์ดควบคุมการบินของอากาศยาน 4 ใบพัด (APM 2.5) ทำการตั้งค่าพารามิเตอร์ เริ่มต้น เช่น ค่าความสูง ค่า Roll, Pitch, Yaw, พิกัดจีพีเอสเริ่มต้น และเข็มทิศอิเล็กทรอนิกส์ เป็นต้น แล้วทำการเรียกใช้งานโปรแกรมควบคุมการบินผ่านรีโมทคอนโทรล ซึ่งอ้างอิงจากช่องสัญญาณที่ใช้ในการควบคุม 3 ช่องสัญญาณคือ Channel 1 ควบคุมการเอียงซ้าย-ขวา, Channel 2 ควบคุมการเคลื่อนที่ไปด้านหน้า-ถอยหลัง และ Channel 3 ควบคุมการเคลื่อนที่ขึ้น-ลงในแนวตั้งของอากาศยาน 4 ใบพัด โดยอาศัยการควบคุมความเร็วมอเตอร์ทั้ง 4 ตัวบนอากาศยาน 4 ใบพัด ที่มีความเร็วแตกต่างกัน เพื่อให้อากาศยาน 4 ใบพัดเคลื่อนที่ไปตามต้องการ



รูปที่ 3.3 ผังการทำงานของส่วนโปรแกรม (ควบคุมแบบอัตโนมัติ) (setup)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.3 เป็นภาพแสดงผังการทำงานของส่วนโปรแกรมควบคุมการบิน (Set up) ไปยังเป้าหมายแบบอัตโนมัติ ซึ่งทำหน้าที่กำหนดค่าเริ่มต้นของเซ็นเซอร์แต่ละตัว ทั้งในส่วนของการรักษาสมดุล, ระดับความสูง และความพร้อมของอุปกรณ์ที่ใช้ในการบินของอากาศยาน 4 ใบพัด



รูปที่ 3.4 ผังการทำงานของส่วนโปรแกรม (ควบคุมแบบอัตโนมัติ) (loop)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.4 เป็นภาพแสดงผังการทำงานของส่วนโปรแกรมควบคุมการบิน (Loop) ซึ่งเป็นโปรแกรมในส่วนที่ใช้ควบคุมการบินไปยังเป้าหมายแบบอัตโนมัติ โดยจะทำงานวนซ้ำจนกระทั่งอากาศยานบินถึงเป้าหมายจึงจะจบการทำงาน มีการแบ่งเป็นฟังก์ชันย่อย ได้แก่ ฟังก์ชันการอ่านเซ็นเซอร์, การกำหนดค่าเริ่มต้นของเป้าหมาย, ควบคุม PID, Filter, สร้างสัญญาณควบคุมมอเตอร์, รักษาความสูง, การอ่าน และรับค่าพิกัดปัจจุบันจากอุปกรณ์ระบุตำแหน่ง (GPS), Take off, Landing และฟังก์ชันควบคุมตำแหน่งการหาพิกัดเป้าหมาย

รูปที่ 3.5 การคำนวณหาทิศไปยังพิกัดเป้าหมาย [11]

จากรูปที่ 3.5 แสดงการคำนวณหาทิศไปยังพิกัดเป้าหมาย โดยอ้างอิงจากทฤษฎีตรีโกณมิติ ซึ่งจะได้สมการทิศเทียบกับทิศเหนือ 0 องศา ดังสมการด้านล่าง

$$A_0 = \text{atan2} \left\{ \cos(\text{Lat}_1) \cdot \sin(\text{Lon}_1 - \text{Lon}_0), \sin(\text{Lat}_1) \cdot \cos(\text{Lat}_0) - \cos(\text{Lon}_1 - \text{Lon}_0) \cdot \cos(\text{Lat}_1) \cdot \sin(\text{Lat}_0) \right\} \quad (3.1)$$

รูปที่ 3.6 การคำนวณหาระยะทางไปยังพิกัดเป้าหมาย [12]

จากรูปที่ 3.6 แสดงการคำนวณหาระยะทางไปยังพิกัดเป้าหมาย โดยอ้างอิงจากทฤษฎีบทพีทาโกรัส, ทฤษฎีการหาระยะทางส่วนโค้งของทรงกลม ซึ่งจะได้ดังสมการด้านล่าง

$$d\text{lon} = \text{lon}_2 - \text{lon}_1 \quad (3.2)$$

$$d\text{lat} = \text{lat}_2 - \text{lat}_1 \quad (3.3)$$

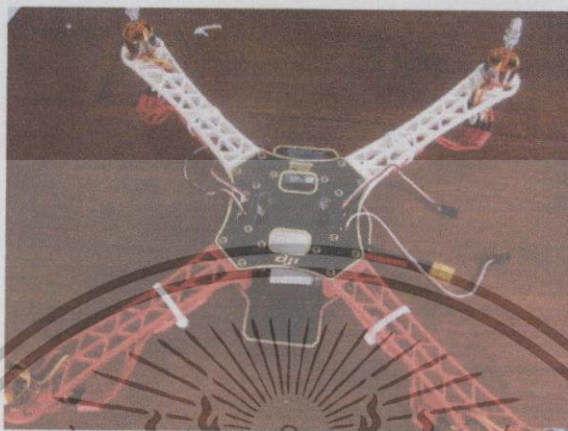
$$a = \left(\sin\left(\frac{d\text{lat}}{2}\right) \right)^2 + \cos(\text{lat}_1) \cdot \cos(\text{lat}_2) \cdot \left\{ \left(\sin\left(\frac{d\text{lon}}{2}\right) \right)^2 \right\} \quad (3.4)$$

$$\text{Distance} = (\text{Earth Radius}) \cdot c \quad (3.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้กับศูนย์วิจัยระบบเพื่อการศึกษาเท่านั้น ผิดอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.1 การจัดสร้างอากาศยาน 4 ใบพัด

1) การเลือกประกอบเฟรม



รูปที่ 3.7 เฟรม 4 ใบพัดชนิดแกนสมมูล [13]

โครงสร้างอากาศยาน 4 ใบพัด แบบเฟรมสมมูลมีลักษณะเป็นแกนไขว้กันในลักษณะกากบาทดังรูปที่ 3.7 ใช้แรงขับเคลื่อนจากใบพัดทั้ง 4 เป็นตัวควบคุมทิศทางโดยเฟรมที่ใช้มีขนาดกว้าง 330 มม. สูง 48 มม. น้ำหนัก 147 กรัม

2) การเลือกใช้บอร์ดควบคุมการบิน



รูปที่ 3.8 บอร์ดควบคุมการบิน Arducopter APM 2.5 [14]

บอร์ดควบคุมการบิน Arducopter APM 2.5 มีคุณสมบัติหลากหลายประการที่ประกอบด้วยเซ็นเซอร์ที่จำเป็นในการควบคุมอากาศยาน รวมทั้งมีขนาดเล็ก ราคาถูก และสามารถโปรแกรมข้อมูลได้ง่ายผ่านโปรแกรม Arduino ซึ่งเป็นโปรแกรมที่ใช้การเขียนในรูปแบบภาษา C++

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะในรูปแบบใดทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) การเลือกใช้มอเตอร์



รูปที่ 3.9 มอเตอร์ EMAX MT2213-935KV

มอเตอร์ EMAX MT2213-935KV มีความเร็วรอบเท่ากับ 935 RMP/V เส้นผ่านศูนย์กลาง 27.9 มม. ใช้คู่กับใบพัด 1045 และใช้แบตเตอรี่แบบ Lipo-3s

4) การเลือกใช้ใบพัด



รูปที่ 3.10 ใบพัดขนาด 10x45 นิ้ว [15]

ใบพัดขนาด 10x45 มม. เป็นใบพัดแบบอ่อน ไม่เหนียว ราคาถูก ถ้าใบตกกระทบพื้น ใบจะหักก่อนที่ shaft ของ motor จะงอ

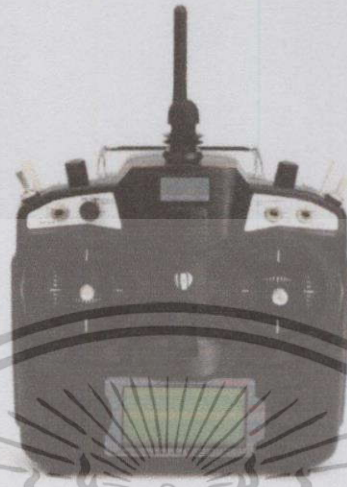
5) การเลือกใช้สปีดคอนโทรล



รูปที่ 3.11 สปีดคอนโทรล MYSTCRY 30A SimonK

ใช้ในการควบคุมความเร็วมอเตอร์ แบบ Brushless โดยรับคำสั่งจากเอกสารไมโครคอนโทรลเลอร์ เพื่อแปลงสัญญาณควบคุมความเร็วมอเตอร์ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

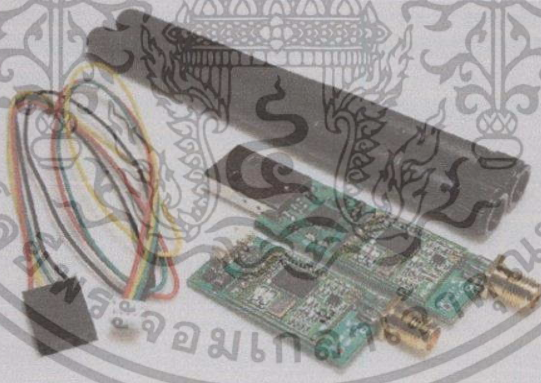
6) การเลือกใช้รีโมทคอนโทรล



รูปที่ 3.12 รีโมทคอนโทรล FS-TH9X [16]

รีโมทคอนโทรล FS-TH9X ขนาด 190x80x240 มม. น้ำหนัก 680 กรัม RF Power 20 dB มีการเข้ารหัสแบบ PPM/PCM ช่องสัญญาณควบคุม 8 ช่องสัญญาณ และการเตือนเมื่อแบตเตอรี่ต่ำ

7) การเลือกใช้เทเลเมทรี (Telemetry)

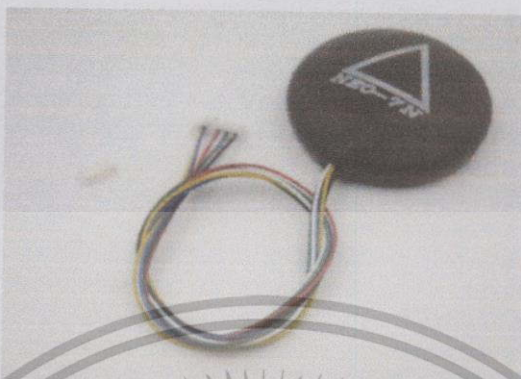


รูปที่ 3.13 เทเลเมทรีความถี่ 433MHz

เทเลเมทรีความถี่ 433MHz อุปกรณ์ที่ใช้ในการเชื่อมต่อไร้สายระหว่างบอร์ด APM ที่อยู่ภาคพื้นดิน ซึ่งมีราคาถูก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) การเลือกใช้อุปกรณ์ระบุตำแหน่ง (GPS)



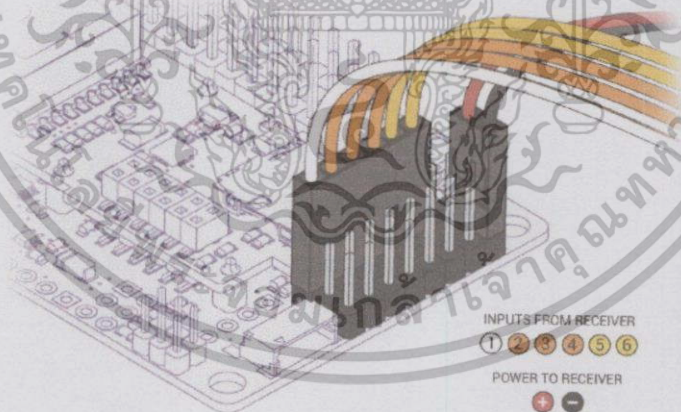
รูปที่ 3.14 อุปกรณ์ระบุตำแหน่ง (GPS) NEO-7M Ublox [17]

อุปกรณ์ระบุตำแหน่ง (GPS) NEO-7M Ublox เป็นอุปกรณ์ระบุตำแหน่ง (GPS) ที่ใช้คู่กับบอร์ด APM ซึ่งมีเข็มทิศยูนิตในตัว ซึ่งประกอบอยู่ภายในเคส โดยอุปกรณ์ระบุตำแหน่ง (GPS) มีขนาดเล็ก เส้นผ่านศูนย์กลาง 35x35 มม. เป็นตัวที่ได้รับการปรับปรุงให้มีประสิทธิภาพสูงขึ้น ค้นหาตำแหน่งได้เร็วขึ้น และตัดสินใจเลือกตำแหน่งได้อย่างรวดเร็ว

3.1.1.2 การออกแบบการทำงานการเชื่อมต่อกล่องรับสัญญาณ

1) การทดสอบกล่องสัญญาณด้าน input กับบอร์ดควบคุม

APM

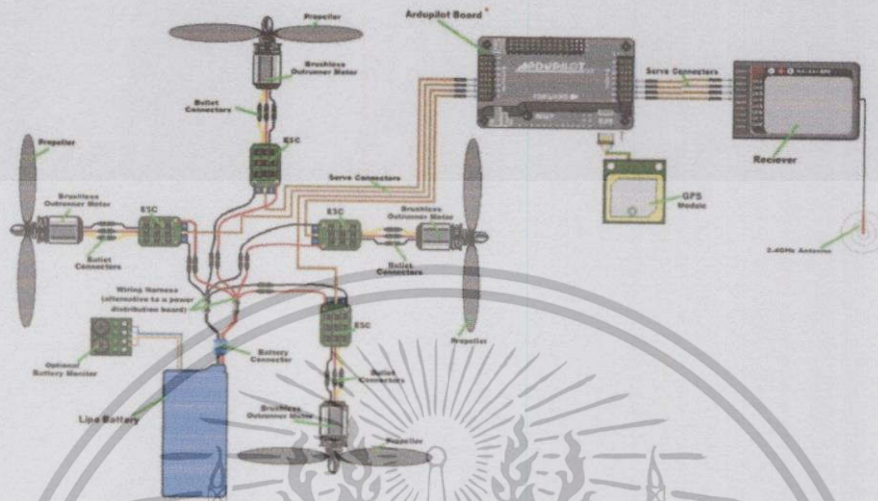


รูปที่ 3.15 การเชื่อมต่อกล่องสัญญาณด้าน input กับบอร์ดควบคุม APM [18]

การทดสอบการเชื่อมต่อกล่องสัญญาณด้าน input กับบอร์ดควบคุม APM ซึ่งเป็นการรับสัญญาณควบคุมจากรีโมทคอนโทรล โดยสัญญาณจะเป็นพัลส์สี่เหลี่ยม และใช้ความกว้างพัลส์ (Duty Cycle) เป็นตัวกำหนดความเร็วของการหมุนมอเตอร์

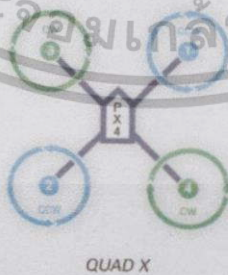
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) การเชื่อมต่ออุปกรณ์บนบอร์ดควบคุมการบิน APM



รูปที่ 3.16 การเชื่อมต่อบอร์ดควบคุมการบิน APM กับมอเตอร์ รีซีฟเวอร์ และจีพีเอส [18]

การเชื่อมต่อบอร์ดควบคุมการบิน APM กับมอเตอร์ รีซีฟเวอร์ และจีพีเอส จะทำการเชื่อมต่อกล่องรีซีฟเวอร์ สัญญาณที่ได้รับจากรีโมทคอนโทรลเข้าทางพอร์ตอินพุตของบอร์ดควบคุมการบิน APM และเชื่อมต่อกับชุดควบคุมมอเตอร์ (สปีดคอนโทรล และมอเตอร์) ออกทางด้านเอาต์พุตของบอร์ดควบคุมการบิน APM แสดงดังรูปที่ 3.16 โดยกำหนดให้อินพุต Channel 1 เป็นการควบคุมการเคลื่อนที่ไปด้านซ้าย-ขวา, อินพุต Channel 2 เป็นการควบคุมการเคลื่อนที่ไปด้านหน้า-หลัง, อินพุต Channel 3 เป็นการควบคุมการเคลื่อนที่ขึ้น-ลง ส่วนทางด้านเอาต์พุต Channel 1 ทำหน้าที่ควบคุมมอเตอร์ตัวที่ 1 (บนขวา), เอาต์พุต Channel 2 ทำหน้าที่ควบคุมมอเตอร์ตัวที่ 2 (ล่างซ้าย), เอาต์พุต Channel 3 ทำหน้าที่ควบคุมมอเตอร์ตัวที่ 3 (บนซ้าย), เอาต์พุต Channel 4 ทำหน้าที่ควบคุมมอเตอร์ตัวที่ 4 (ล่างขวา) ดังรูปที่ 3.17



CLOCKWISE ROTATION
USE PUSHER PROPELLER

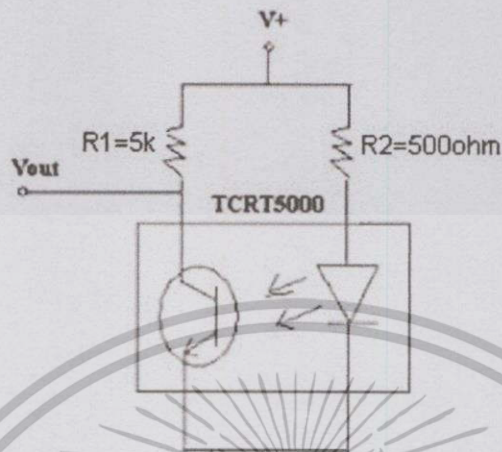


COUNTER-CLOCKWISE ROTATION
USE NORMAL PROPELLER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับเราใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.17 ตำแหน่งการวางมอเตอร์ [15]

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) วงจรนับรอบความเร็วของมอเตอร์



รูปที่ 3.18 วงจรนับรอบความเร็วของมอเตอร์

วงจรถับรอบความเร็วของมอเตอร์ โดยใช้ Optical Reflex sensor TCRT5000 ในการนับรอบความเร็วของมอเตอร์ โดยติดแทบสติกเกอร์สีเหลืองที่ตัวมอเตอร์ และนำวงจรไปตรวจจับความเร็วของแสงที่เปลี่ยนแปลงไป โดยใช้ Oscilloscope เป็นเครื่องมือวัดสัญญาณ ซึ่งจะได้รูปสัญญาณที่ต่างกันไป เมื่อมีการเปลี่ยนแปลงของลิ้มอเตอร์ จะได้เป็นสัญญาณที่มีคาบซ้ำ ๆ กัน สามารถนำมาคำนวณหาความถี่ และจำนวนรอบต่อนาที (RPM) ได้จากสมการที่ 3.6

$$\text{RPM(รอบต่อนาที)} = \frac{[\text{ความถี่(รอบต่อวินาที)} \times 60(\text{วินาที})]}{1(\text{วินาที})} \quad (3.6)$$

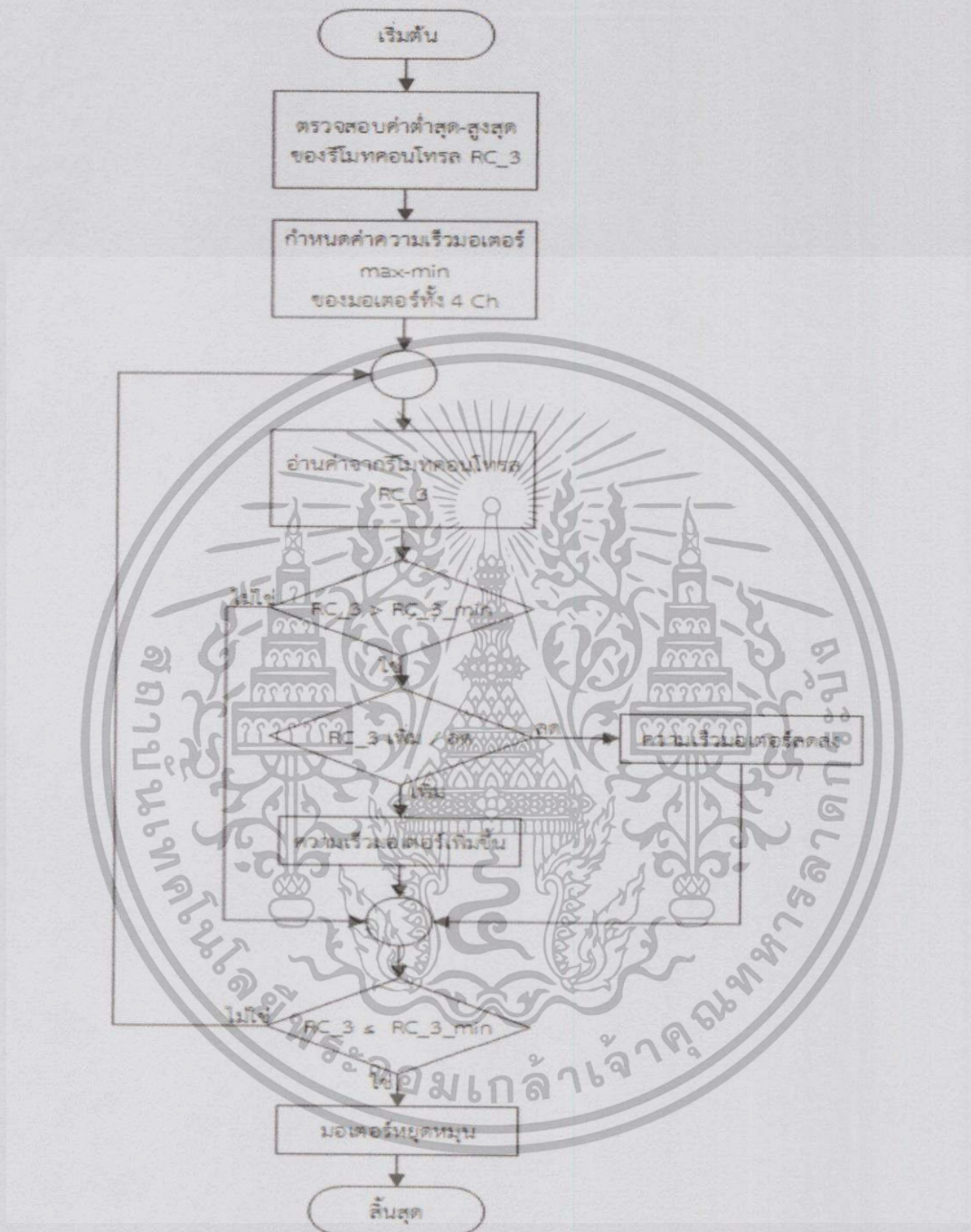
3.2 การออกแบบโปรแกรมควบคุมการเคลื่อนที่ของอากาศยาน 4 ใบพัด

3.2.1 ส่วนการควบคุมการเคลื่อนที่ (แบบใช้รีโมทควบคุม)

3.2.1.1 ควบคุมการเคลื่อนที่ขึ้น-ลง

โปรแกรมควบคุมการเคลื่อนที่ขึ้น-ลง โดยรับค่าจากรีโมทคอนโทรล จะใช้โปรแกรม ArduPilot Arduino 1.0.3 ใช้การเขียนโปรแกรมในรูปแบบของภาษา C++ เป็นภาษาที่ใช้ในการเขียนโปรแกรม โดยขั้นตอนการเขียนโปรแกรมสามารถอธิบายโครงสร้างด้วย Flow Chart ดังนี้

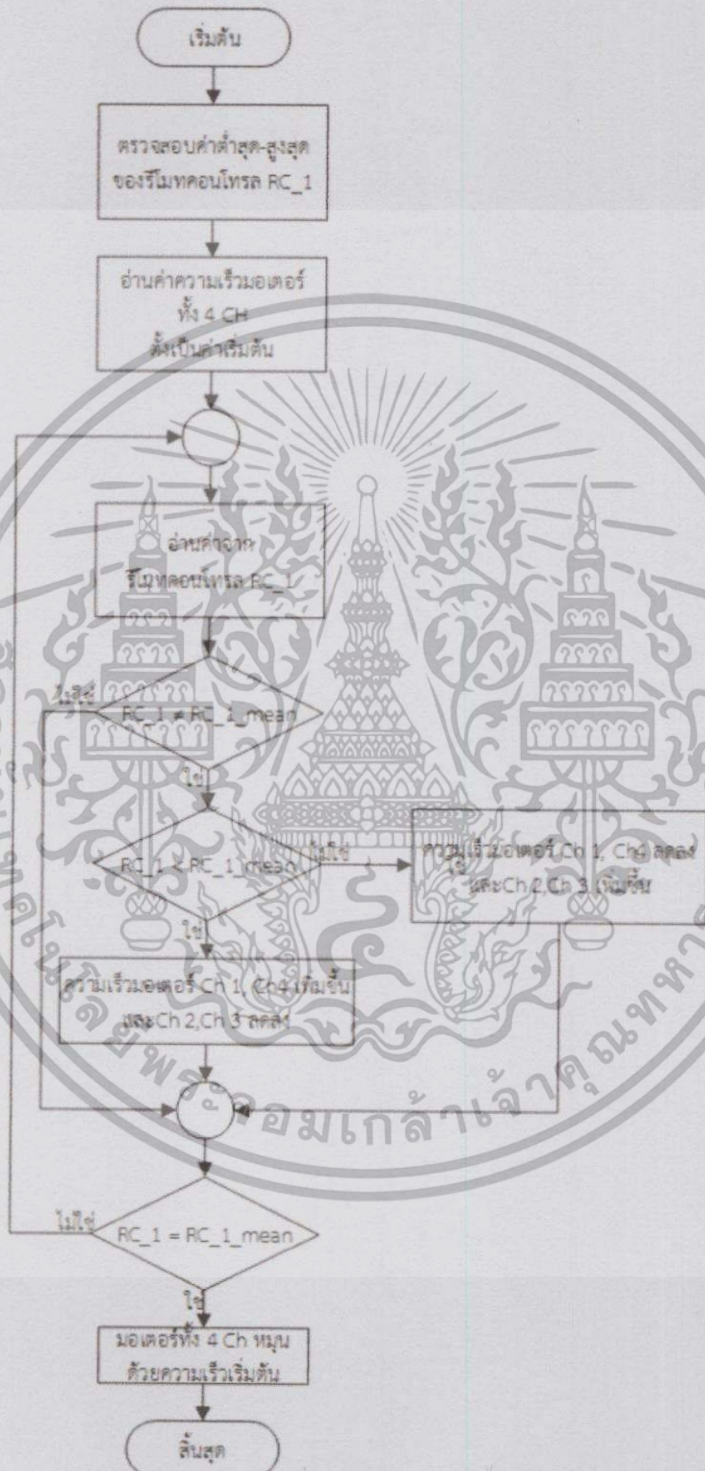
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 การทำงานขณะบินขึ้น - ลง ของอากาศยาน 4 ใบพัด

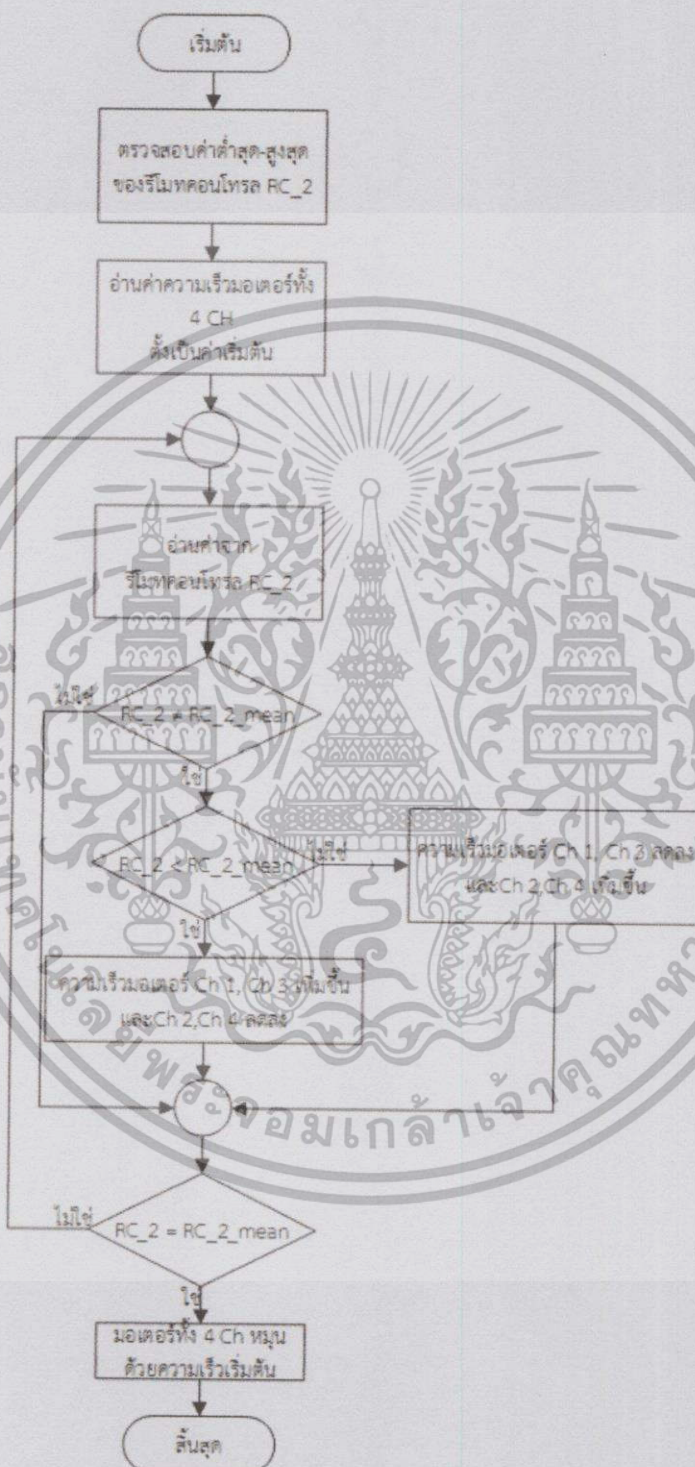
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.2 ควบคุมการเคลื่อนที่ซ้าย-ขวา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 3.20 การทำงานขณะบินเอียงซ้าย - ขวา ของอากาศยาน 4 ใบพัด
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

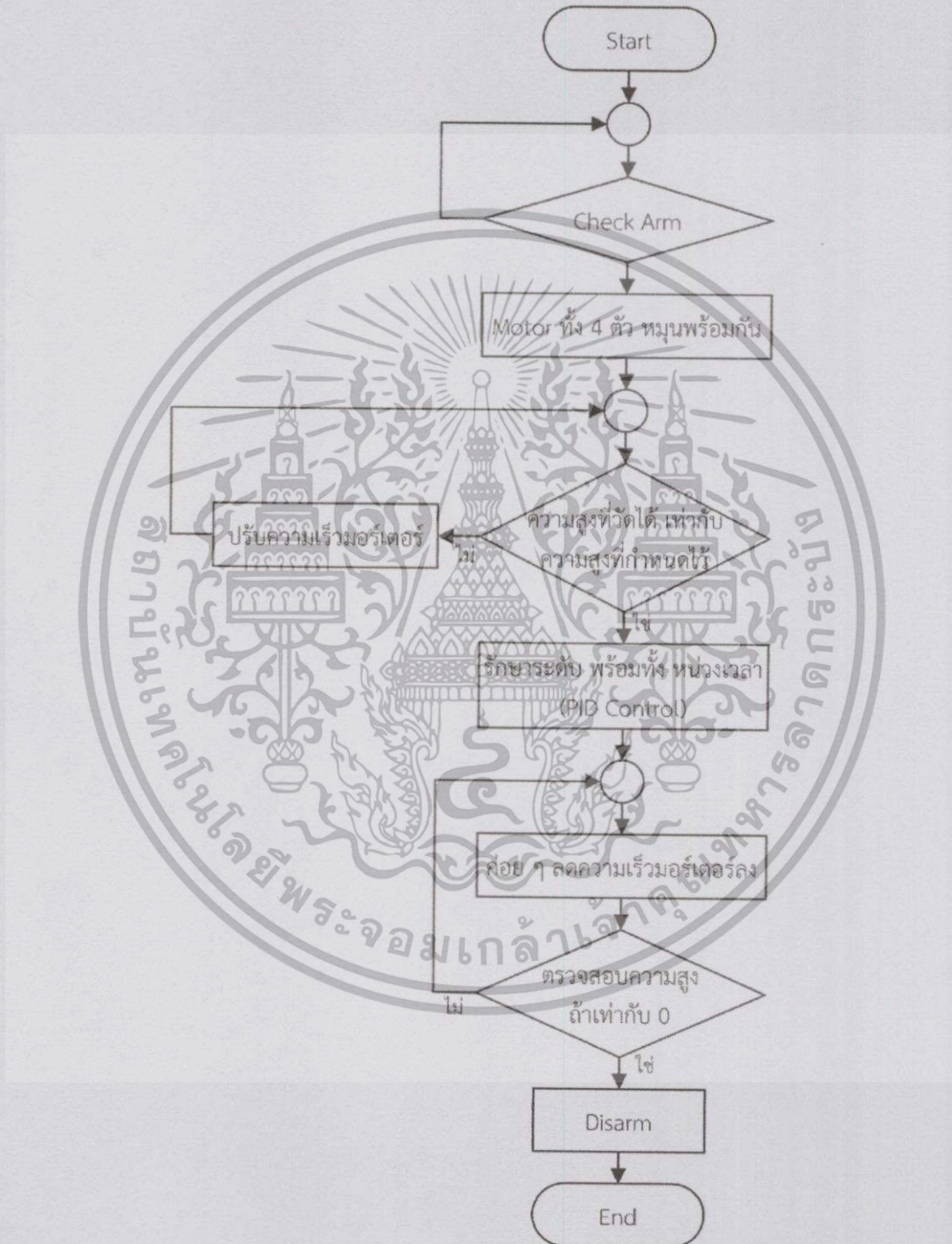
3.2.1.3 ควบคุมการเคลื่อนที่เอียงหน้า - หลัง



เอกสารนี้เป็นเอกสารรูปที่ 3.21 การทำงานขณะบินเอียงหน้า - หลัง ของอากาศยาน 4 ใบพัด ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ส่วนการควบคุมการเคลื่อนที่ (แบบใช้โปรแกรมควบคุม)

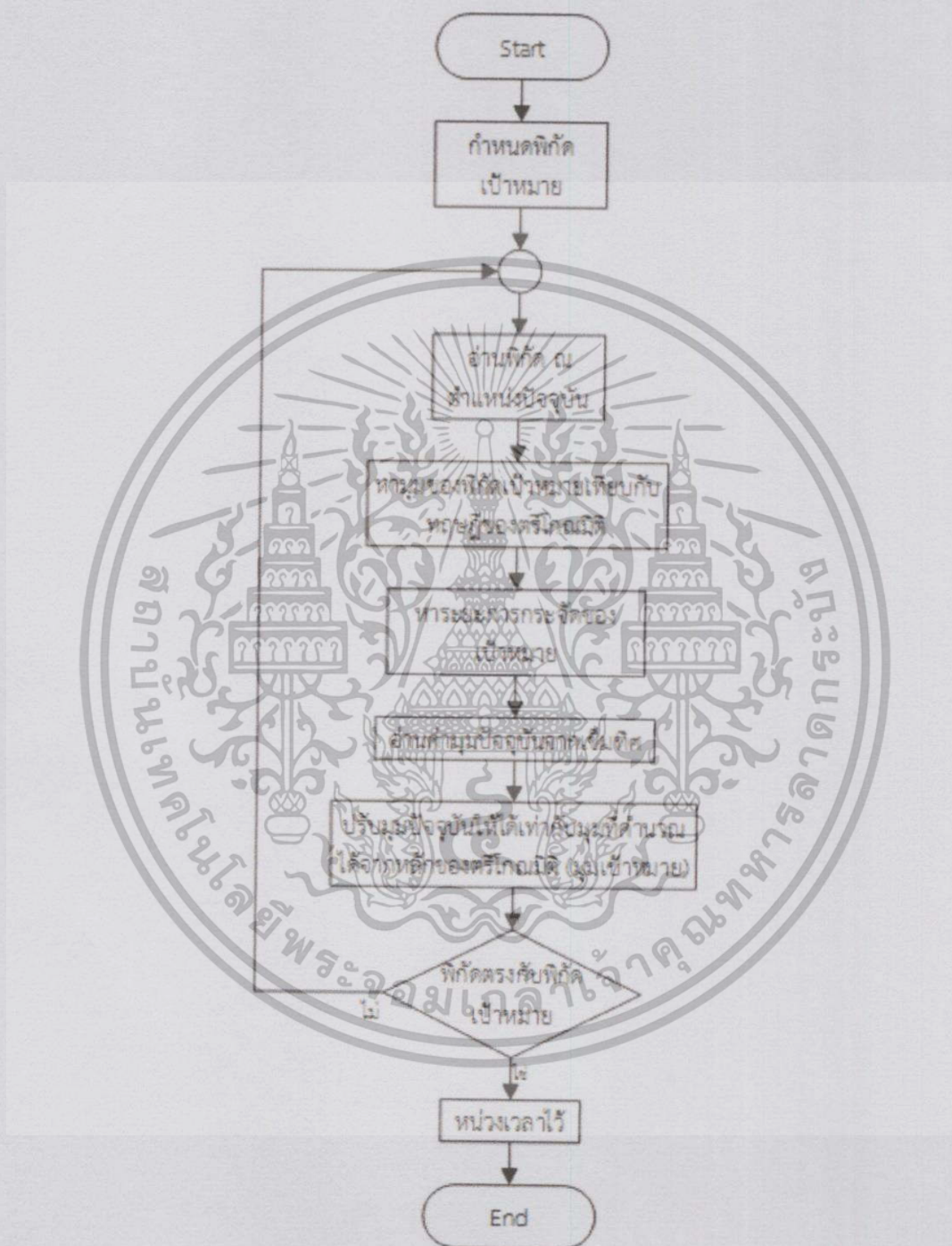
3.2.2.1 โปรแกรมควบคุมการบินขึ้น (Take off) และลงจอด (Landing)



เอกสารนี้เป็นรูปที่ 3.22 การทำงานโปรแกรมควบคุมการบินขึ้น (Take off) และลงจอด (Landing) ใช้นักเรียนด้านการศึกษา

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.2 โปรแกรมควบคุมการตัดสินใจเลือกพิกัด

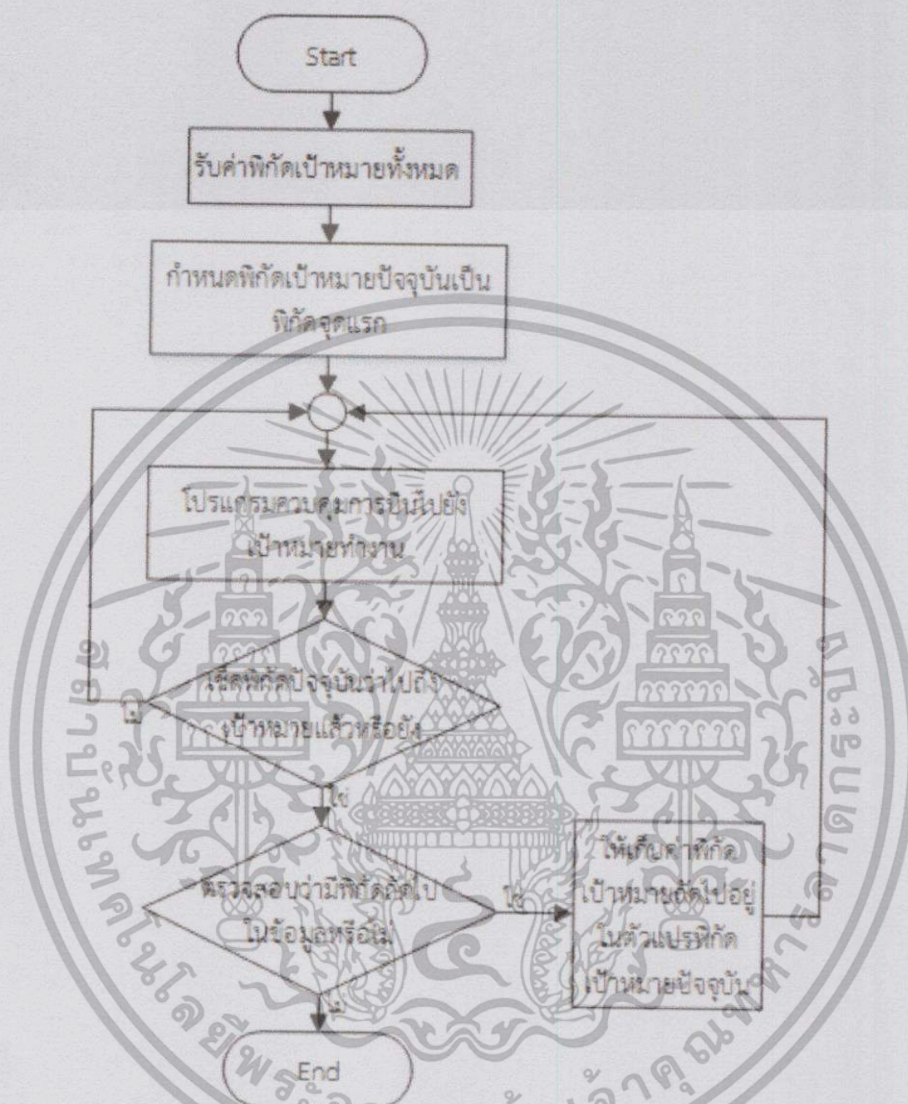


รูปที่ 3.23 การทำงานโปรแกรมควบคุมการตัดสินใจ และระบุพิกัดเป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.3 โปรแกรมควบคุมการบินไปยังเป้าหมายถัดไป

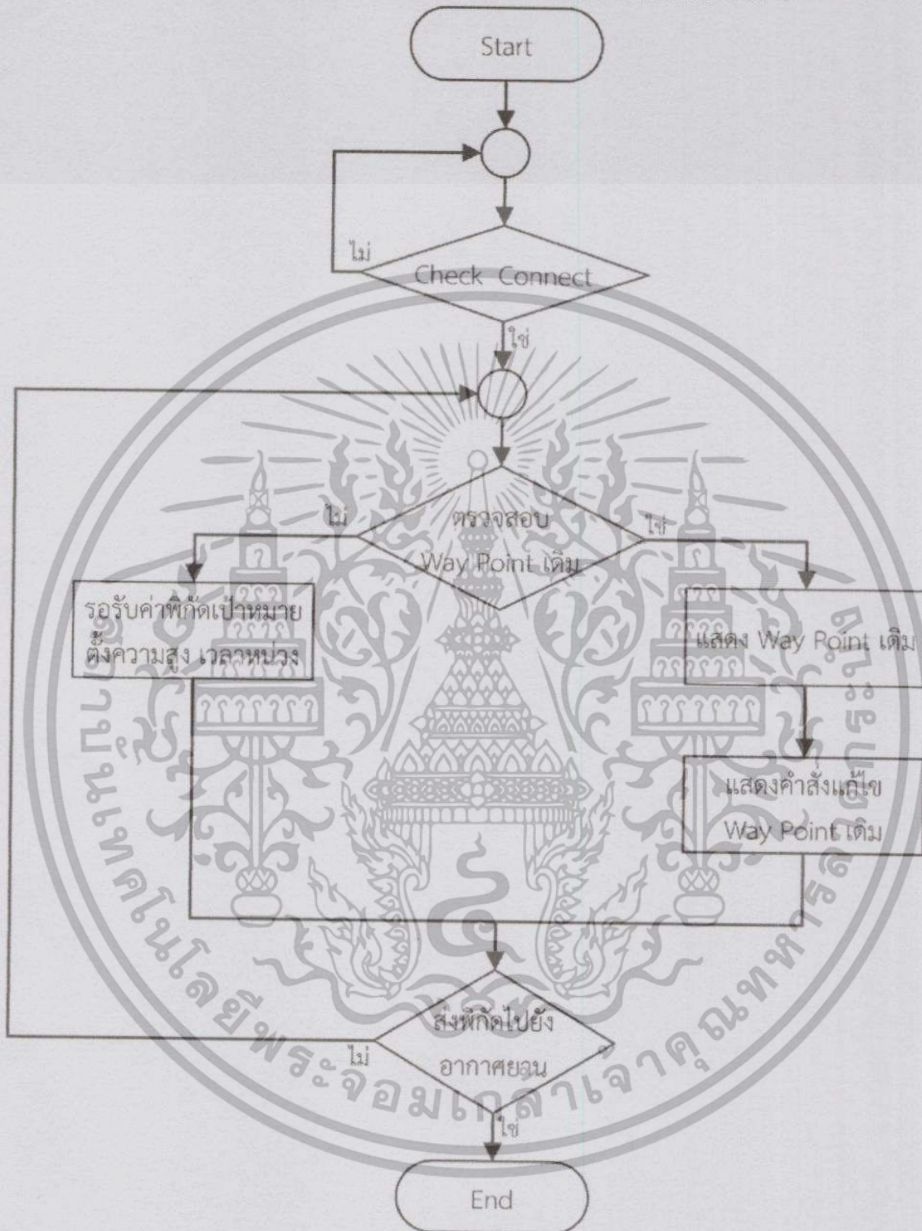


รูปที่ 3.24 ลำดับการบินไปยังเป้าหมายถัดไปของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะในรูปแบบใดทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 ส่วนแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์

3.2.3.1 การทำงานของแอปพลิเคชันกำหนดเป้าหมาย



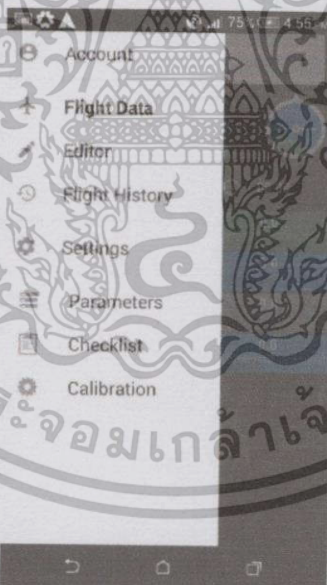
รูปที่ 3.25 การทำงานของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3.2 หน้า GUI ของแอปพลิเคชัน



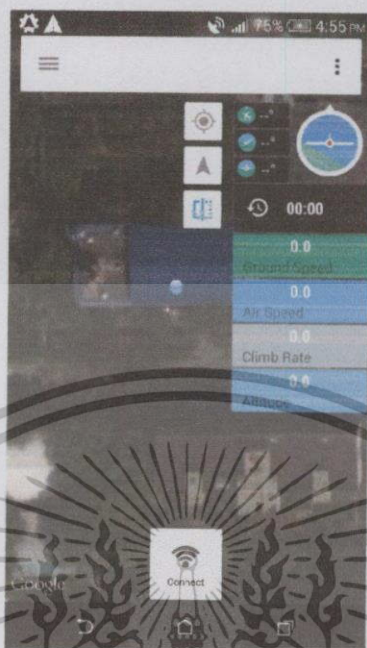
รูปที่ 3.26 ไอคอนของแอปพลิเคชัน



รูปที่ 3.27 เมนูแสดงผลหน้าแอปพลิเคชัน

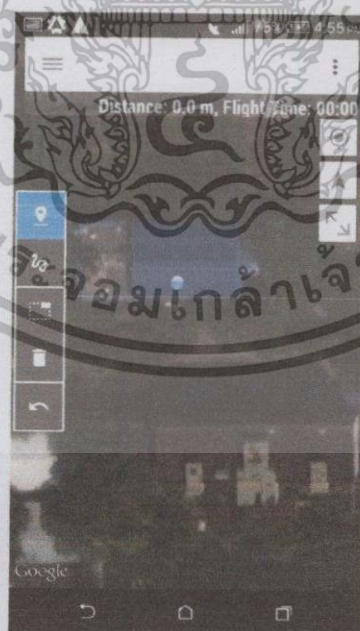
ในส่วนของแอนดรอยด์แอปพลิเคชัน จะใช้รายการ Flight data, Editor และ Flight History ดังแสดงในรูปที่ 3.28 และ 3.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.28 หน้าต่าง Flight

จากรูปที่ 3.28 แสดงหน้าต่างเมื่อเลือกรายการ Flight data ซึ่งเป็นหน้าที่แสดงข้อมูลการบินของอากาศยานไร้คนขับ โดยสามารถแสดงตำแหน่งปัจจุบันของอากาศยาน แสดงข้อมูลต่างๆ ของอากาศยาน และสามารถเชื่อมต่อระหว่างผู้ใช้กับอากาศยานโดยกดปุ่ม connect



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการรูปที่ 3.29 หน้าต่าง Editor ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.29 แสดงหน้าต่าง Editor เป็นหน้าต่างสำหรับจัดการการบินของอากาศยานไร้คนขับ โดยสามารถระบุพิกัดให้อากาศยานบินไปตามพิกัดที่ระบุไว้ได้ ดังแสดงในรูปที่ 3.30 และสามารถกำหนดระดับความสูงของอากาศยานรวมทั้งกำหนดการหน่วงเวลาในการบินไปพิกัดถัดไปได้ ดังรูปที่ 3.31 และในรูปที่ 3.32 จะแสดงหน้าต่าง Flight History ซึ่งแสดงประวัติการบินของอากาศยาน



รูปที่ 3.30 ระบุพิกัดในหน้า Editor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับก... มอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ข้อมูลนี้... ครั้งที่มีการนำไปใช้

รูปที่ 3.31 กำหนดความสูง และ การหน่วงเวลาของอากาศยาน



รูปที่ 3.32 หน้าต่าง Flight History

3.2.4 โปรแกรมที่ใช้ในการศึกษา

3.2.4.1 โปรแกรมที่ใช้ในการเขียนคำสั่งควบคุมการทำงาน

ในการพัฒนาโปรแกรมการทำงานของอากาศยาน 4 ใบพัด จะใช้โปรแกรมที่ใช้สำหรับพัฒนาโดยเฉพาะ คือ โปรแกรม ArduPilot-Arduino-1.0.3 แสดงดังรูปที่ 3.33 ซึ่งแตกต่างจาก Arduino ทั่วไป

```

R/C Input Test | Arduino IDE (Windows)
File Edit Sketch Tools Window Help

R/C Input Test

#include <AP_Common.h>
#include <AP_Math.h>
#include <AP_Param.h>
#include <StorageManager.h>
#include <AP_ProcAccel.h>

#include <AP_HAL.h>
#include <AP_HAL_AVR.h>

#if CONFIG_HAL_BOARD == HAL_BOARD_APM2
const AP_HAL::HAL& hal = AP_HAL_AVR_APM2;
#elif CONFIG_HAL_BOARD == HAL_BOARD_APM1
const AP_HAL::HAL& hal = AP_HAL_AVR_APM1;
#endif

void multiread(AP_HAL::RCInput* in) {
    // Multi-channel read method:
    uint16_t channels[8];
}

```

รูปที่ 3.33 โปรแกรม ArduPilot-Arduino-1.0.3 ที่ใช้ในการพัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4.2 โปรแกรมที่ใช้ในการแสดงผล และตั้งค่าเริ่มต้นของโปรแกรมที่พัฒนา

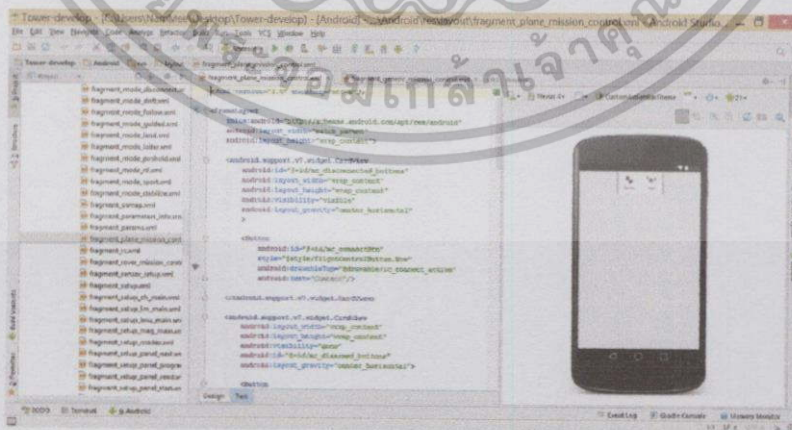
ในการทดสอบ และติดตั้งเฟิร์มแวร์ของโปรแกรมที่พัฒนาแล้วจำเป็นต้องใช้โปรแกรมที่ใช้ในการตรวจสอบ และตั้งค่าเริ่มต้นให้กับโปรแกรมที่ถูกพัฒนา รวมทั้งใช้ในการทดสอบการแสดงผลค่าที่รับได้จากเซ็นเซอร์ต่าง ๆ เช่น ค่า Roll, Pitch และ Yaw พิกัดอุปกรณ์ระบุตำแหน่ง (GPS) และเข็มทิศอิเล็กทรอนิกส์ เป็นต้น โดยใช้โปรแกรม Mission Planer 1.3.9 แสดงหน้าต่างโปรแกรมดังรูปที่ 3.34



รูปที่ 3.34 หน้าต่างโปรแกรม Mission Planer 1.3.9

3.2.4.2 โปรแกรมที่ใช้ในการออกแบบ และจัดสร้างแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์

ในการพัฒนา และออกแบบโครงสร้างของแอปพลิเคชันได้มีการใช้โปรแกรม Android Studio ในการออกแบบ และตรวจสอบความถูกต้องของแอปพลิเคชัน รวมทั้งการจำลองรูปแบบของแอปพลิเคชันในใช้ในการจัดทำ ดังแสดงในรูปที่ 3.35



เอกสารนี้เป็นเอกสารที่สงวน รูปที่ 3.35 หน้าต่างโปรแกรม Android Studio 1.1.0 ที่นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การจัดเก็บผลการทดลอง

3.3.1 การตรวจความพร้อม (Calibrate) ของเข็มทิศอิเล็กทรอนิกส์

ทำการทดสอบการตั้งค่าใช้งานเข็มทิศอิเล็กทรอนิกส์ โดยการหมุนเข็มทิศอิเล็กทรอนิกส์รอบแกน $\pm Roll$, $\pm Pitch$, $\pm Yaw$ เพื่อเก็บตัวอย่างองศาทิศทางในลักษณะการเคลื่อนที่รูปแบบต่าง ๆ

3.3.2 การทดสอบรับสัญญาณจากรีโมทคอนโทรล

ทำการทดสอบการรับสัญญาณจากรีโมทคอนโทรล โดยวัดสัญญาณจากกล่องรีซีฟเวอร์ด้านรับ ผ่าน Oscilloscope เพื่อสังเกตลักษณะของข้อมูลที่ใช้ในการควบคุม

3.3.3 การตรวจความพร้อม (Calibrate) ของรีโมทคอนโทรล และตัวรีซีฟเวอร์

ทำการเชื่อมต่อระหว่างเครื่องรีโมทคอนโทรล, รีซีฟเวอร์ และบอร์ดควบคุมการบิน APM จากนั้นทำการวัดค่าต่ำสุด-สูงสุด ของวงจรโยกคันโยกของรีโมทคอนโทรล เพื่อใช้ในการอ้างอิงในโปรแกรม

3.3.4 การทดสอบการกำหนดโหมดการบิน

ทำการทดสอบการกำหนดโหมดการบิน เพื่อกำหนดการใช้งานชุดคำสั่งโปรแกรมแต่ละโหมด

3.3.5 การทดสอบรับค่าความสูงจากบาร์รอมิเตอร์

ทำการทดสอบรับค่าความสูงจากบาร์รอมิเตอร์เทียบกับความสูงจริง เพื่อทดสอบการใช้งาน และความถูกต้องของบาร์รอมิเตอร์

3.3.6 การทดสอบการใช้งานอุปกรณ์ระบุพิกัด (GPS)

ทำการทดสอบการใช้งานอุปกรณ์ระบุพิกัด (GPS) เทียบกับพิกัดจริง เพื่อตรวจสอบความถูกต้อง และแม่นยำของค่าพิกัดที่วัดได้

3.3.7 การทดสอบการรับค่าพารามิเตอร์ Roll จากไจโรสโคป

ทำการทดสอบการรับค่าพารามิเตอร์ Roll จากไจโรสโคป โดยการหมุนรอบแกน Roll โดยหมุนจากซ้ายไปขวาเป็นมุมบวก และถ้าหมุนจากขวาไปซ้ายเป็นมุมลบ เทียบกับมุมที่หมุนจริง

3.3.8 การทดสอบการรับค่าพารามิเตอร์ Pitch จากไจโรสโคป

ทำการทดสอบการรับค่าพารามิเตอร์ Pitch จากไจโรสโคป โดยการหมุนรอบแกน Pitch โดยหมุนจากหน้าไปหลังเป็นมุมบวก และถ้าหมุนจากหลังไปหน้าเป็นมุมลบ เทียบกับมุมที่หมุนจริง

3.3.9 การทดสอบการรับค่าพารามิเตอร์ Yaw จากไจโรสโคป

ทำการทดสอบการรับค่าพารามิเตอร์ Yaw จากไจโรสโคป โดยการหมุนรอบแกน Yaw โดยอ้างอิงจากการหมุนของเข็มทิศ เทียบกับมุมที่หมุนจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.10 การทดสอบวัดสัญญาณด้านเอาต์พุตที่ใช้ในการควบคุมความเร็วมอเตอร์ผ่านสปีดคอนโทรล

ทำการทดสอบวัดสัญญาณด้านเอาต์พุตที่ใช้ในการควบคุมความเร็วมอเตอร์ผ่านสปีดคอนโทรลทั้ง 4 ตัว โดยกำหนดเงื่อนไขการเคลื่อนที่ที่แตกต่างกัน เพื่อตรวจสอบประสิทธิภาพโปรแกรมที่ถูกพัฒนา

3.3.11 การทดสอบวัดความเร็วของมอเตอร์เทียบกับสัญญาณเอาต์พุตจากบอร์ดควบคุมการบิน

ทำการทดสอบส่งสัญญาณควบคุมที่มี Duty Cycle ต่างกันไปยังสปีดคอนโทรล และดูค่าความเร็วในการหมุนของมอเตอร์ (RPM) เปรียบเทียบกัน

3.3.12 การทดสอบหาความเร็วมอเตอร์ต่ำสุดที่ทำให้อากาศยาน 4 ใบพัด ยกตัว

ทำการทดสอบวัดความเร็วรอบมอเตอร์ต่ำสุดที่สามารถยกตัวอากาศยาน 4 ใบพัด

3.3.13 การทดสอบวัดสัญญาณเอาต์พุตควบคุมความเร็วมอเตอร์จากบอร์ดควบคุมการบิน APM ในกรณีต่าง ๆ

ทำการทดสอบเปรียบเทียบสัญญาณที่ควบคุมความเร็วมอเตอร์แต่ละตัว โดยกำหนดเงื่อนไขการบินของอากาศยาน 4 ใบพัด จากช่องสัญญาณของรีโมทคอนโทรล กำหนด Ch.1 ควบคุมการเอียงซ้าย - ขวา ของอากาศยาน, Ch.2 ควบคุมการเดินหน้า - ถอยหลัง ของอากาศยาน และ Ch.3 ควบคุมการบินขึ้น - ลง ของอากาศยาน

3.3.14 การทดสอบการบินขึ้น (Take off) และลงจอด (Landing)

ทำการทดสอบคำสั่งในการควบคุมให้อากาศยาน 4 ใบพัดบินขึ้น (Take off) และลงจอด (Landing) โดยการกำหนดจุดเป้าหมาย รวมทั้งความสูงที่ใช้ในการบินแต่ละครั้ง

3.3.15 การทดสอบรักษาระดับความสูงการบินของอากาศยาน 4 ใบพัด

ทำการทดสอบความสูงของระดับการบินของอากาศยาน 4 ใบพัด ที่ใช้ในการบินจริง โดยอ้างอิงจากระดับความสูงที่วัดได้จากบาร์โรมิเตอร์ เปรียบเทียบกับความสูงจริงที่วัดได้ เพื่อเปรียบเทียบความถูกต้องของการบิน และใช้ในการควบคุมระดับการบินของอากาศยาน 4 ใบพัด

3.3.16 การทดสอบการเลือกใช้ระดับความสูงในการบันทึกภาพของอากาศยาน 4 ใบพัด

ทำการทดสอบบันทึกภาพที่ได้จากอากาศยาน 4 ใบพัด โดยกำหนดระดับความสูงที่ใช้อ้างอิง เทียบกับขนาดขอบเขตของพื้นที่ที่กล้องสามารถบันทึกได้ เพื่อกำหนดระยะความสูงของระดับการบิน

3.3.17 การทดสอบเก็บข้อมูล (Data) พิกัดที่ใช้ในการบินของอากาศยาน 4 ใบพัด

ทำการทดสอบเก็บค่าพิกัดในการใช้ทดสอบการบินของอากาศยาน 4 ใบพัด

เอกสาร โดยทำการเก็บข้อมูลในบริเวณสนามลี้กบีสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการวัดสนามลึกลับ และกำหนดจุดพิกัดให้ห่างกันระยะ 5 เมตร ทั้งในแนวของละติจูด และ ลองจิจูด เพื่อใช้ในการกำหนดขอบเขตการทดสอบการบินแบบอัตโนมัติ

3.3.18 การทดสอบหาค่าพารามิเตอร์ของ PID control ที่ใช้ในการแก้ไขความผิดพลาดของอุปกรณ์ระบุตำแหน่ง (GPS)

ทำการทดสอบหาค่าพารามิเตอร์ของ PID control ที่ใช้ควบคุมความผิดพลาด เพื่อให้อากาศยานสามารถบินไปยังพิกัดเป้าหมายได้ถูกต้อง และมีความคลาดเคลื่อนน้อยที่สุด โดยการเปรียบเทียบค่าพารามิเตอร์ที่ใช้กับผลการทดสอบหาระยะผิดพลาด เพื่อหาค่าพารามิเตอร์ที่ดีที่สุด

3.3.19 การทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 1 จุดโดยกำหนดค่าความสูงของอากาศยานคงที่

ทำการทดสอบการบินของอากาศยาน 4 ใบพัด โดยกำหนดพิกัดเป้าหมาย 1 จุด และสั่งงานให้ให้อากาศยาน 4 ใบพัด บินไปยังจุดเป้าหมายเพื่อตรวจสอบความถูกต้องของเงื่อนไขในการสร้างเส้นทางไปยังจุดเป้าหมาย

3.3.20 การทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 5 จุด โดยกำหนดค่าความสูงของอากาศยานคงที่

ทำการทดสอบการบินของอากาศยาน 4 ใบพัด มากกว่า 1 จุด โดยเลือกพิกัดจากที่ได้เก็บข้อมูลไว้ จำนวน 5 จุด รวมทั้งกำหนดความสูงที่ใช้ในการบินในแต่ละจุดเป้าหมายให้สูง 5 เมตร เพื่อทดสอบความถูกต้องของการบิน รวมทั้งการรักษาระดับการบินของอากาศยาน 4 ใบพัด

3.3.21 การทดสอบการใช้งานกำหนดเป้าหมายการบินของแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์

ทำการทดสอบการใช้งานกำหนดเป้าหมายการบิน โดยใช้แอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์ที่ได้จัดสร้างขึ้น เพื่อทดสอบประสิทธิภาพการใช้ของแอปพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

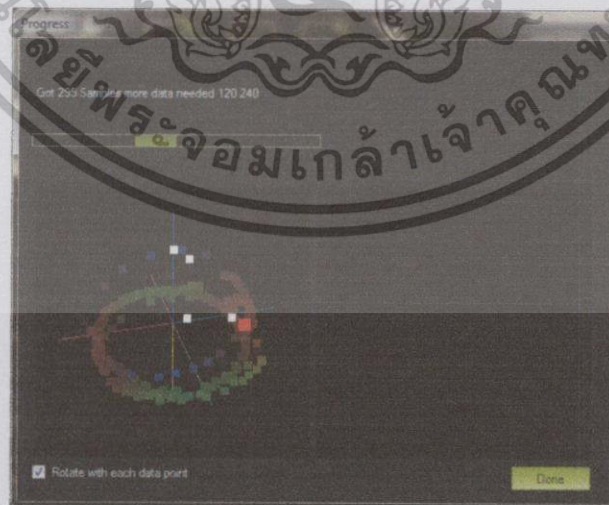
ผลการทดลอง

ในบทนี้จะกล่าวถึงการดำเนินงานโดยประมวลความรู้ที่ศึกษามาทั้งหมดมาประกอบกัน โดยผลการทดลองจะกล่าวถึงการทำงานส่วนต่างๆ ที่ได้พัฒนาขึ้นในโครงการนี้ ซึ่งแบ่งเป็น หัวข้อการทดลองหลักๆ คือ การตรวจความพร้อม (Calibrate) อุปกรณ์เซ็นเซอร์ต่าง ๆ ภายในบอร์ด ควบคุม APM ในส่วนของ เซ็มทีซีอิเล็กทรอนิกส์, ไจโรสโคป และบาร์โรมิเตอร์ รวมทั้งทำการตรวจความพร้อม (Calibrate) รีโมทคอนโทรล และทำการทดสอบการรับสัญญาณจากรีโมทคอนโทรล, การทดสอบการใช้งานอุปกรณ์ระบุพิกัด (GPS) นอกจากนี้ได้ทำการทดสอบการกำหนดโหมดการบิน ในโหมดต่างๆ, การทดสอบวัดความเร็วของมอเตอร์เทียบกับสัญญาณเอาต์พุตจากบอร์ดควบคุมการบิน APM และการทดสอบหาความเร็วมอเตอร์ต่ำสุดที่ทำให้อากาศยาน 4 ใบพัด ยกตัว ต่อมาได้ทำการทดสอบโปรแกรมที่ใช้ในการควบคุมอากาศยาน 4 ใบพัด โดยพัฒนาต่อยอดจากการควบคุมด้วย รีโมทคอนโทรล อาทิ การควบคุมการบินขึ้น (Take off) และลงจอด (Landing), การรักษาระดับ ความสูง, การทดสอบการบินของอากาศยาน 4 ใบพัดไปยังเป้าหมายทั้งแบบ 1 จุด และมากกว่า 1 จุด หลังจากนั้นทำการพัฒนาแอปพลิเคชันที่ใช้ในการกำหนดพิกัดเป้าหมายการเคลื่อนของอากาศยาน 4 ใบพัด

4.1 ผลการทดสอบการตรวจความพร้อม (Calibrate) ของเซ็มทีซีอิเล็กทรอนิกส์

วัตถุประสงค์ของการทดลอง

1. เพื่อตรวจความพร้อมของเซ็มทีซีอิเล็กทรอนิกส์
2. เพื่อเก็บตัวอย่างการหาทิศของอุปกรณ์เซ็มทีซีอิเล็กทรอนิกส์



รูปที่ 4.1 Calibrate Compass

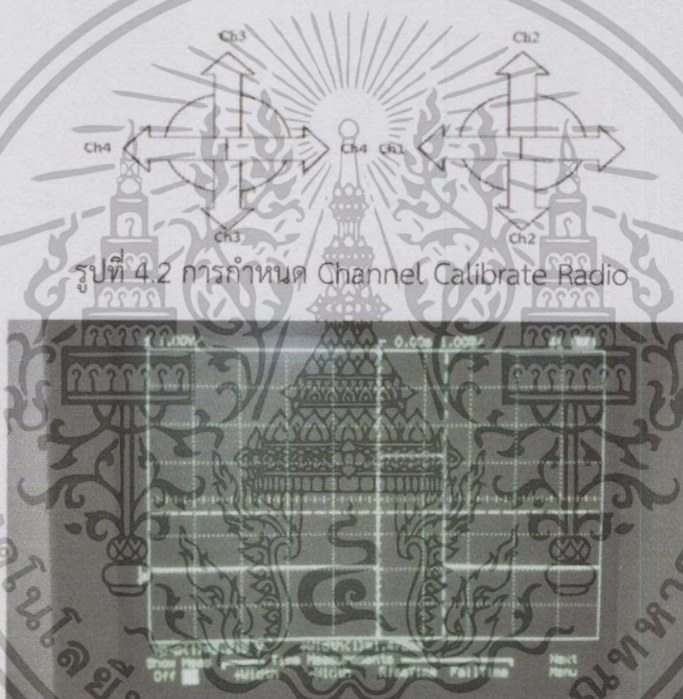
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.1 สามารถทำการ Calibrate เข็มทิศของอากาศยานเพื่อให้อากาศยานจดจำทิศที่ถูกต้อง โดยหมุนอากาศยาน 360 องศา ทั้งในทิศรอบแกน Roll, Pitch, Yaw ทั้งแบบหมุนตามเข็มนาฬิกาและหมุนทวนเข็มนาฬิกา โปรแกรมจะแสดงตำแหน่งของทิศของอากาศยานบนแกนในรูปแบบ 3 มิติ

4.2 ผลการทดสอบรับสัญญาณจากรีโมทคอนโทรล

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการรับสัญญาณจากรีโมทคอนโทรล
2. เพื่อสังเกตลักษณะของข้อมูลที่ใช้ในการควบคุม



รูปที่ 4.3 ความกว้างพัลส์สัญญาณของรีโมทคอนโทรลที่ใช้ในการควบคุมอากาศยาน 4 ใบพัด

สัญญาณควบคุมที่วัดได้จากรีโมทคอนโทรล (วัดจาก Receiver) โดยใช้ Oscilloscope พบว่า สัญญาณที่รับได้เมื่อเปลี่ยนแปลงตำแหน่งของคันบังคับจะใช้การเปลี่ยนแปลงความกว้างของพัลส์เป็นตัวควบคุมการบินของอากาศยาน 4 ใบพัดดังรูปที่ 4.3 โดยกำหนดคันบังคับของรีโมทคอนโทรล ตามรูปที่ 4.2

4.3 ผลการตรวจความพร้อม (Calibrate) ของรีโมทคอนโทรล และตัวรีซีฟเวอร์

วัตถุประสงค์ของการทดลอง

1. เพื่อตรวจความพร้อมของภาคส่งและภาครับของรีโมทคอนโทรล
2. เพื่อทราบค่าสูงสุด-ต่ำสุดของแต่ละช่องสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในเชิงพาณิชย์เท่านั้น ผิดกฎหมายหากมีการเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



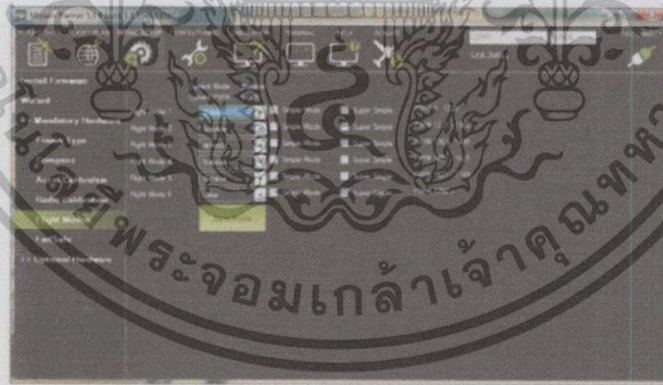
รูปที่ 4.4 การทดสอบวัดค่าสูงสุด - ต่ำสุดในการ Calibrate Radio ผ่านโปรแกรม Mission Planner

จากการทดสอบวัดค่าสูงสุด - ต่ำสุดของรีโมทคอนโทรล เมื่อโยกคันบังคับแต่ละ Channel จะพบว่าแต่ละ Channel จะให้ค่าสูงสุด - ต่ำสุดของความกว้างพัลส์ในหน่วย (um) เพื่อเอาไปใช้ในการควบคุมขั้นต่อไป ดังรูปที่ 4.4

4.4 ผลการทดสอบการกำหนดโหมดการบิน

วัตถุประสงค์ของการทดลอง

1. เพื่อกำหนดการใช้งานชุดคำสั่งโปรแกรมแต่ละโหมด
2. เพื่อทราบลักษณะการทำงานของแต่ละโหมดการบิน



รูปที่ 4.5 การกำหนดโหมดของการบินของอากาศยานบิน 4 ใบพัด

สามารถกำหนดโหมดของการบิน โดยแบ่งเป็น 3 Fight mode กำหนดให้ Fight mode 1 เป็น Stabilize mode, Fight mode 3 เป็น Alt Hold และ Fight mode 6 เป็น Loiter แสดงหน้าต่างการกำหนดโหมดดังรูปที่ 4.5 โดยที่กำหนดไว้ 3 Fight mode เนื่องจาก channel ที่ใช้ในการสั่งงานของรีโมทคอนโทรล จะให้ channel 6 ที่เป็นสวิตซ์ซึ่งสัญญาณสามารถปรับได้ 3 ระดับ คือ ต่ำ กลาง สูง จึงนำมาใช้ในการปรับโหมดการบินได้ โดย Stabilize mode คือ โหมด

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

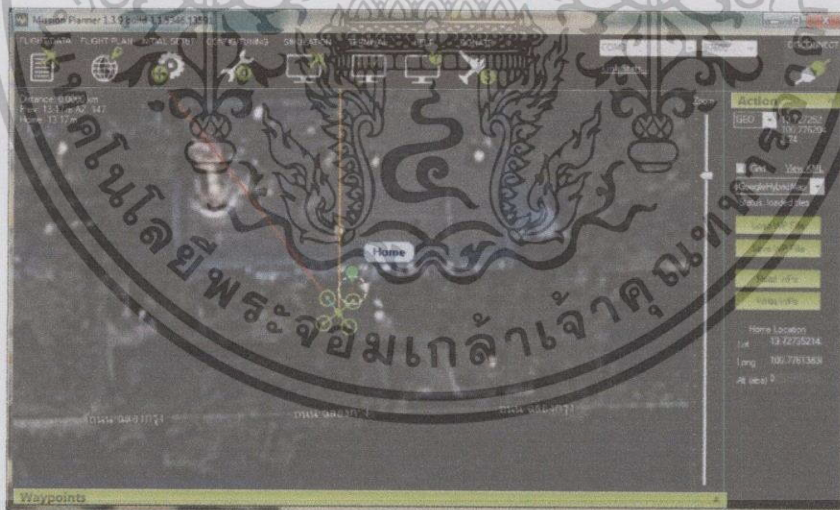
ตารางที่ 4.1 ผลการทดสอบการรับค่าความสูงจากบาร์อิมิตอร์

Distance (m)	ค่าที่วัดได้จาก บาร์อิมิตอร์ (m)
0	0.00
0.6	0.74
0.96	1.04
1.18	1.15
5.00	5.04
8.66	8.72
9.16	9.14
12.41	12.41
12.91	12.79
16.10	16.57

4.6 ผลการทดสอบการใช้งานอุปกรณ์ระบุพิกัด (GPS)

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการรับค่าตำแหน่งพิกัดที่รับได้เทียบกับพิกัดจริง
2. ตรวจสอบผลของข้อมูลที่ได้รับว่าถูกต้องตามความต้องการหรือไม่



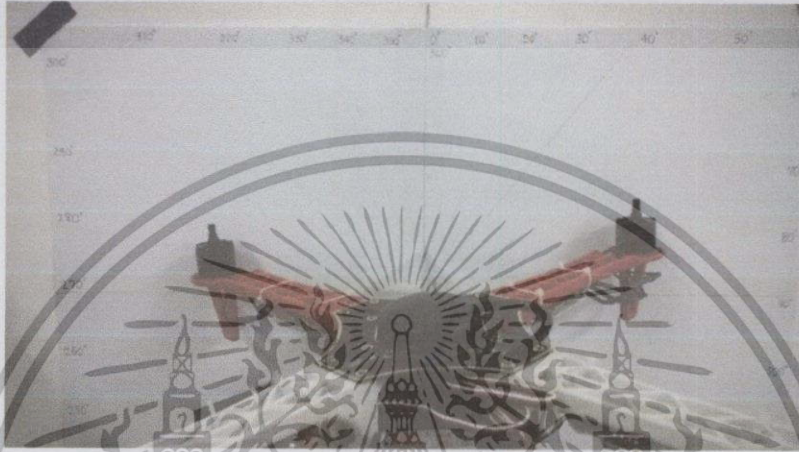
รูปที่ 4.7 การใช้งานอุปกรณ์ระบุพิกัด (GPS)

จากรูปที่ 4.7 แสดงการรับค่าพิกัดจากดาวเทียมโดยผ่านอุปกรณ์จีพีเอส โดยพิกัดที่วัดจริงคือตำแหน่ง home มีค่าละติจูด 13.727352 และลองจิจูด 100.776138 ซึ่งเปรียบเทียบกับพิกัดของอากาศยานที่วัดได้ มีค่าละติจูด 13.727252 และลองจิจูด 100.776204 ระยะที่คลาดเคลื่อนจะอยู่รอบๆ ตำแหน่ง home ในรัศมี 5 เมตร หลังจากทดลองวัดหลายๆ ครั้ง

4.7 ผลการทดสอบการรับค่าพารามิเตอร์ Roll จากไจโรสโคป

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการรับค่าพารามิเตอร์ Roll จากไจโรสโคป
2. เพื่อศึกษาการเปลี่ยนแปลงค่าพารามิเตอร์ Roll เทียบกับมุมที่หมุนจริง



รูปที่ 4.8 การรับค่าพารามิเตอร์ roll จากไจโรสโคปเทียบกับมุมที่หมุนจริงรอบแกน roll



รูปที่ 4.9 ค่าพารามิเตอร์ roll ที่อ่านได้จากโปรแกรม Mission Planner

จากรูปที่ 4.8 และ 4.9 จะแสดงการทดสอบการรับค่าพารามิเตอร์ roll จากไจโรสโคป เทียบกับการหมุนจริงเป็นองศา รอบแกน roll โดยอ้างอิงมุมจากแนวระดับ 0 องศา ให้หมุนตามเข็มนาฬิกา วัดค่ามุมสูงสุดเป็น 360 องศา มุมต่ำสุดเป็น 0 องศา จะแสดงผลที่ได้ดังตารางที่ 4.2 จะเห็นได้ว่า มุมของค่าพารามิเตอร์ roll จะเริ่มต้นที่ 0 องศาที่แนวระดับ หากหมุนตามเข็มนาฬิกา จะเป็นมุมค่าบวกไปจนถึงมุม +180 องศา และ หากหมุนทวนเข็มนาฬิกาจะเป็นมุมค่าลบไปจนถึงมุม -180 องศา เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ผลการทดลองการรับค่าพารามิเตอร์ roll จากโจโรสโคป

Angle (Degree)	Roll (Degree)
-180	-179.638
-170	-170.022
-160	-162.873
-150	-149.635
-140	-142.19
-130	-129.881
-120	-121.609
-110	-112.601
-100	-102.323
-90	-91.6265
-80	-79.094
-70	-70.2028
-60	-60.6568
-50	-50.381
-40	-41.6928
-30	-30.5213
-20	-20.2978
-10	-11.5769
0	0.4271
10	11.8318
20	20.4683
30	29.7243
40	39.6872
50	48.6192
60	60.4415
70	70.0561
80	82.7619
90	91.4881
100	100.8878
110	109.5934
120	119.9303

เอกสารนี้เป็นเอกสารที่สงวนไว้ 110 หรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่เอ 109.5934 ไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามทำซ้ำหรือดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าที่ 119.9303 ทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ผลการทดลองการรับค่าพารามิเตอร์ roll จากใจโรสโคป (ต่อ)

Angle (Degree)	Roll (Degree)
130	130.9599
140	141.7543
150	150.137
160	158.1894
170	169.2722
180	179.2916

4.8 ผลการทดสอบการรับค่าพารามิเตอร์ Pitch จากใจโรสโคป

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการรับค่าพารามิเตอร์ Pitch จากใจโรสโคป
2. เพื่อศึกษาหลักการหมุนของค่าพารามิเตอร์ Pitch เทียบกับมุมที่หมุนจริง



รูปที่ 4.10 การทดสอบการรับค่าพารามิเตอร์ pitch จากใจโรสโคป เทียบกับมุมที่หมุนจริงรอบแกน pitch



รูปที่ 4.11 ค่าพารามิเตอร์ pitch ที่อ่านได้จากโปรแกรม Mission Planner

เอกสารนี้เป็นเอกสารที่... ปรึกษาประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

ในรูปที่ 4.10 และ 4.11 แสดงการทดสอบการรับค่าพารามิเตอร์ pitch จากใจโรสโคป จะอ้างอิงมุมจากแนวระดับ 0 องศา ให้ทิศที่หมุนตามเข็มนาฬิกามีค่าเป็นบวก และทิศทวนเข็มนาฬิกามีค่าเป็นลบ และวัดค่ามุมสูงสุดเป็น 90 องศา มุมต่ำสุดเป็น -90 องศา จะแสดงผลที่ได้ดังตารางที่ 4.3

ตารางที่ 4.3 ผลการทดลองการรับค่าพารามิเตอร์ pitch จากใจโรสโคป

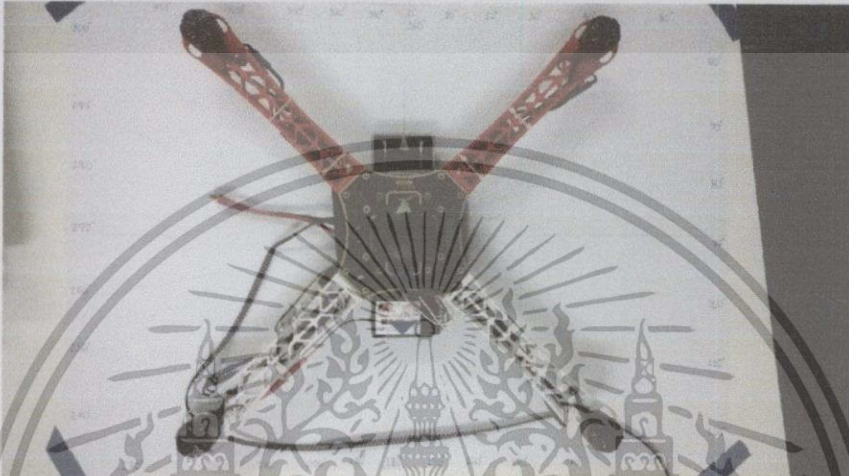
Angle (Degree)	Pitch (Degree)
-90	-89.0925
-80	-80.7182
-70	-69.0263
-60	-59.4194
-50	-50.3839
-40	-39.0434
-30	-30.1516
-20	-20.7143
-10	-10.7228
0	0.0647
10	10.8569
20	21.5081
30	31.9671
40	40.8904
50	50.7095
60	60.272
70	71.3499
80	81.5657
90	88.4347

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.9 ผลการทดสอบการรับค่าพารามิเตอร์ Yaw จากใจโรสโคป

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการรับค่าพารามิเตอร์ Yaw จากใจโรสโคป
2. เพื่อศึกษาหลักการหมุนของค่าพารามิเตอร์ Yaw เทียบกับมุมที่หมุนจริง



รูปที่ 4.12 การรับค่า yaw จากใจโรสโคป เทียบกับมุมที่หมุนจริงรอบแกน yaw



รูปที่ 4.13 ค่าพารามิเตอร์ yaw ที่อ่านได้จากโปรแกรม Mission Planer

ในรูปที่ 4.12 และ 4.13 จะแสดงการทดสอบการรับค่าพารามิเตอร์ yaw จากใจโรสโคป โดยใช้ทิศเหนือในการอ้างอิง (0 องศา) และหมุนตามเข็มนาฬิกา วัดค่ามุมสูงสุดเป็น 360 องศา มุมต่ำสุดเป็น 0 องศา จะแสดงผลที่ได้ดังตารางที่ 4.4 จะเห็นได้ว่ามุมที่หมุนจะเพิ่มในทิศทางทวนเข็มนาฬิกา การใช้งานจริงจึงต้องเอา 360 ลบด้วยมุมที่อ่านค่าได้จาก ค่าพารามิเตอร์ yaw เอกสาร จึงจะเอาไปใช้งานจริงได้ เช่นเดียวกับเข็มทิศที่การศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 ผลการทดลองการรับค่า yaw จากไจโรสโคป

Angle (Degree)	Yaw (Degree)
0	7.8582
10	358.6215
20	349.2006
30	339.5251
40	330.1513
50	320.3524
60	310.1923
70	299.5216
80	289.5392
90	279.7528
100	269.9234
110	260.2349
120	249.1962
130	239.4729
140	229.8716
150	219.5423
160	209.5913
170	200.3995
180	189.2503
190	179.0395
200	169.0203
210	159.1729
220	148.8428
230	138.2457
240	122.2368
250	116.5472
260	106.3951
270	94.8219
280	84.5492
290	75.4207
300	64.5923

เอกสารนี้เป็นเอกสารที่สงวนไว้ 290 รับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุ 75.4207 ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้าม 300 ที่ัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของ 64.5923 กรณีที่มีการนำไปใช้

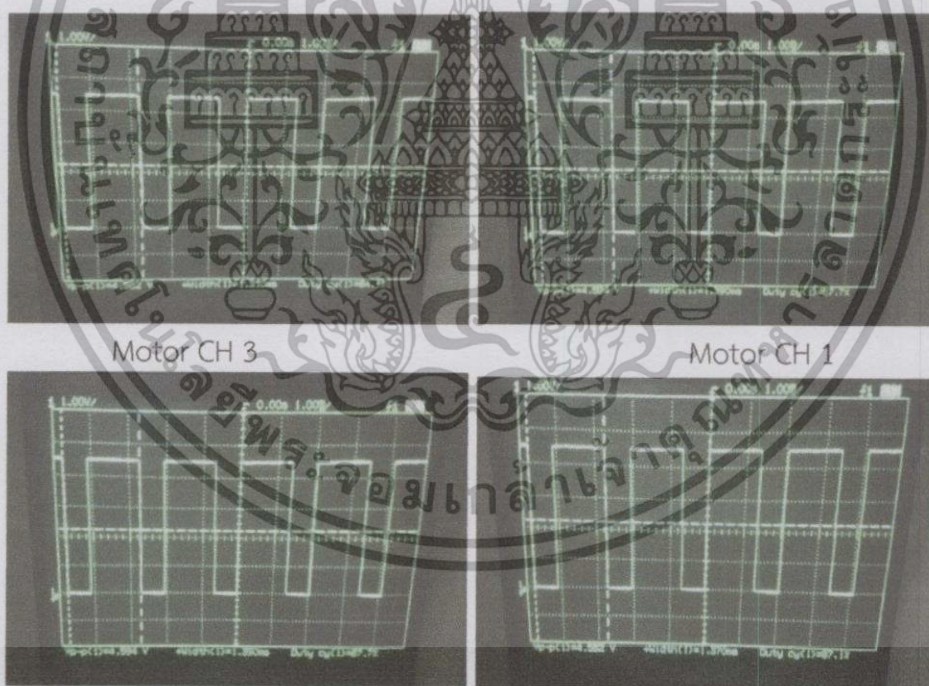
ตารางที่ 4.4 ผลการทดลองการรับค่า yaw จากไจโรสโคป (ต่อ)

Angle (Degree)	Yaw (Degree)
310	55.0247
320	45.0753
330	35.5627
340	26.7209
350	18.7925
360	7.2319

4.10 ผลการทดสอบวัดสัญญาณด้านเอาต์พุตที่ใช้ในการควบคุมความเร็วมอเตอร์ผ่านสปีดคอนโทรล

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบสัญญาณด้านเอาต์พุตที่ใช้ในการควบคุมความเร็วมอเตอร์
2. เพื่อตรวจสอบประสิทธิภาพการสั่งงานควบคุมผ่านรีโมทคอนโทรล



รูปที่ 4.14 สัญญาณควบคุมมอเตอร์ทั้ง 4 ตัวที่ออกจาก APM Board

สัญญาณที่ส่งออกไปควบคุมมอเตอร์ทั้ง 4 ตัว จะใช้ Duty Cycle เป็นตัวกำหนดเอกสารนี้เป็นเอกสารที่กล่าวไว้ว่าห้ามการใช้งานเพื่อวัตถุประสงค์อื่น ไม่นับแต่เห็นว่าเป็นประโยชน์ในการค้า ความเร็วมอเตอร์แต่ละตัว หาก Duty Cycle มีค่ามาก จำนวนรอบการหมุนของมอเตอร์แต่ละตัวก็ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

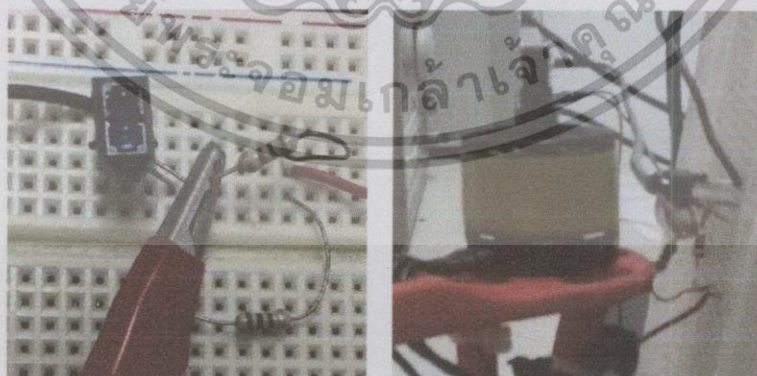
จะมากขึ้นตามอัตราการเพิ่มของ Duty Cycle โดยกำหนดค่า Duty Cycle ต่ำสุด 50% และค่า Duty Cycle สูงสุด 80% ซึ่งการเปลี่ยนแปลง Duty Cycle ของมอเตอร์แต่ละตัวจะถูกควบคุมจากความกว้างพัลส์ที่ได้จากรีโมทคอนโทรล และตัวโปรแกรมที่เขียนไว้ ซึ่งในรูปที่ 4.14 จะแสดงสัญญาณที่ใช้ควบคุมความเร็วมอเตอร์ทั้ง 4 ตัว ซึ่งตัวอย่างเมื่อบังคับให้อากาศยานสีใบพัดเอียงซ้ายซึ่งมอเตอร์ตัวที่ 1 และ 4 (ทางขวา) จะมีความเร็วมากกว่ามอเตอร์ตัวที่ 2 และ 3 (ทางซ้าย) ดังรูปที่ 4.14

4.11 ผลการทดสอบวัดความเร็วของมอเตอร์เทียบกับสัญญาณเอาต์พุตจากบอร์ดควบคุมการบิน

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบหาค่าความสัมพันธ์ระหว่างสัญญาณควบคุมกับความเร็วมอเตอร์
2. เพื่อตรวจสอบประสิทธิภาพการทำงานของมอเตอร์และสปีดคอนโทรล

การทดสอบวัดความเร็วของมอเตอร์เทียบกับสัญญาณเอาต์พุตจากบอร์ด โดยทำการป้อนสัญญาณอินพุตเข้าไปที่สปีดคอนโทรล ในลักษณะของขบวนพัลส์สี่เหลี่ยมที่มีค่า Duty Cycle ต่างกัน แล้วทำการนับรอบความเร็วมอเตอร์ โดยใช้วงจร Photo detector (รูปที่ 4.15) ในการนับซึ่งวัดจาก Oscilloscope ได้สัญญาณดังรูปที่ 4.16 โดยให้ความถี่ของสัญญาณ Channel 1 คือจำนวนรอบที่มอเตอร์หมุนได้ใน 1 วินาที หากต้องการเป็นค่า RPM จะหาจากนำความถี่ที่ได้ไปคูณด้วย 60 จะได้เป็นจำนวนรอบต่อนาที หรือ RPM ออกมา ส่วน Channel 2 เป็นค่าสัญญาณที่ป้อนเข้าไปในสปีดคอนโทรล โดย เมื่อทดลองสร้างกราฟความสัมพันธ์ระหว่างความเร็วมอเตอร์ (RPM) กับ Duty Cycle ของสัญญาณ (รูปที่ 4.17) พบว่า มอเตอร์จะเริ่มหมุนที่ Duty Cycle มีค่า = 57% และหมุนด้วยความเร็วสูงสุดที่ Duty Cycle \geq 80%

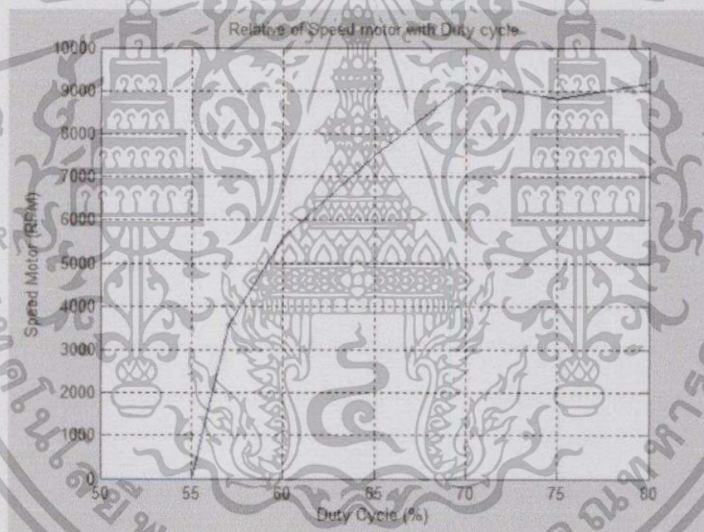


รูปที่ 4.15 การนับรอบความเร็วมอเตอร์ (RPM) โดยใช้ Photo detector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 เปรียบเทียบสัญญาณที่ใช้ควบคุม (ch2) กับความเร็วมอเตอร์ (RPM) (ch1) ที่วัดได้

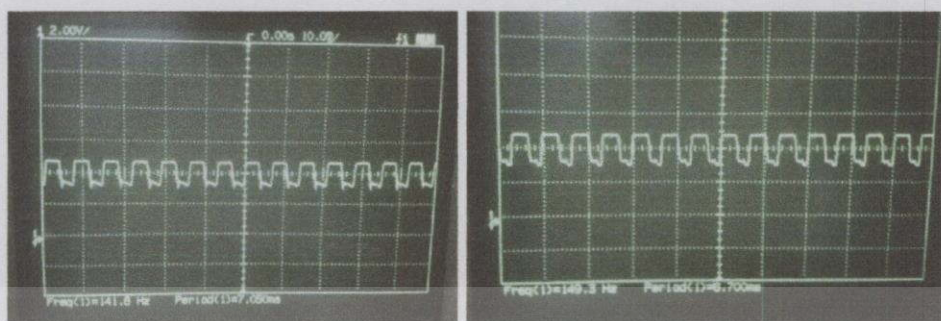


รูปที่ 4.17 กราฟความสัมพันธ์ระหว่างความเร็วมอเตอร์ (RPM) กับ Duty Cycle ของสัญญาณ

4.12 ผลการทดสอบหาความเร็วมอเตอร์ต่ำสุดที่ทำให้อากาศยาน 4 ใบ พัดยกตัว วัตถุประสงค์ของการทดลอง

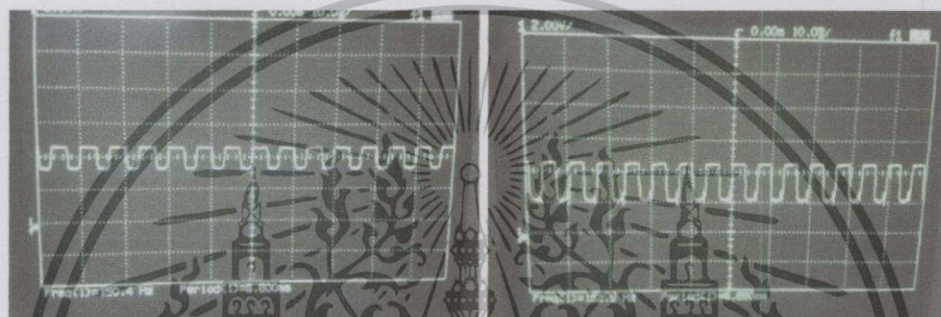
1. เพื่อทดสอบวัดความเร็วรอบมอเตอร์ต่ำสุดที่สามารถยกตัวอากาศยาน 4 ใบพัดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Motor CH3

Motor CH1



Motor CH2

Motor CH4

รูปที่ 4.18 ความเร็วมอเตอร์ต่ำสุดที่ทำให้อากาศยาน 4 ใบพัดเริ่มยกตัว

จากรูปที่ 4.18 แสดงความเร็วมอเตอร์เริ่มต้นที่ทำให้อากาศยาน 4 ใบพัดเริ่มยกตัว โดยจากสัญญาณพบว่าความเร็วมอเตอร์เฉลี่ยที่ทำให้อากาศยานเริ่มยกตัว อยู่ที่ประมาณ 9000 RPM ซึ่งเป็นความเร็วเริ่มต้นที่ทำให้อากาศยานยกตัว

4.13 ผลการทดสอบวัดสัญญาณเอาต์พุตควบคุมความเร็วมอเตอร์จากบอร์ดควบคุมการบิน APM ในกรณีต่างๆ

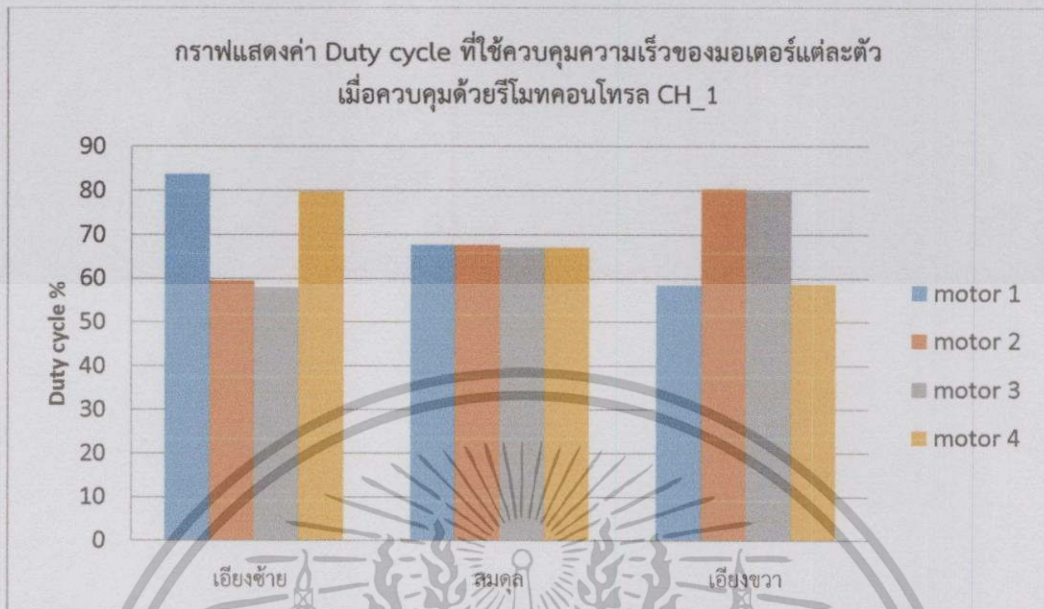
วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบส่งสัญญาณควบคุมที่มี Duty Cycle ต่างกันไปยัง สปีดคอนโทรลด้วยเงื่อนไขที่แตกต่างกัน

2. เพื่อทดสอบการทำงานของรีโมทคอนโทรลที่ใช้บังคับทิศทางการบิน

เมื่อทดสอบโยกคันบังคับของรีโมทคอนโทรล แต่ละ Channel ตามเงื่อนไขต่างๆ แล้ว วัด Duty Cycle ของสัญญาณเอาต์พุตของมอเตอร์แต่ละตัวโดยใช้ Oscilloscope มาเขียนเปรียบเทียบเป็นแผนภาพ จะได้แผนภาพของแต่ละ Channel ดังนี้

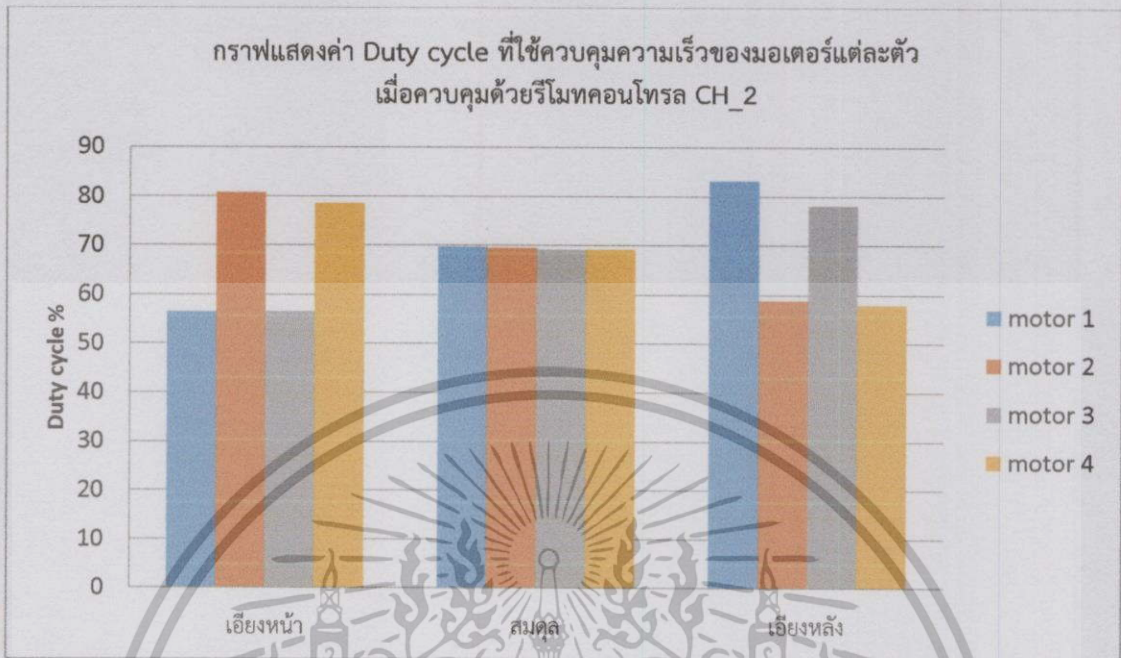
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 การเปลี่ยนแปลงความเร็วของมอเตอร์แต่ละตัว (Duty cycle) เมื่อบังคับโดยใช้รีโมทคอนโทรล CH_1

ในรูปที่ 4.19 เห็นได้ว่า CH 1 จะใช้ในการควบคุมการเอียงซ้าย เอียงขวาของอากาศยาน 4 ใบพัดเป็นหลัก โดยหากเอียงซ้าย มอเตอร์สองตัวด้านขวาต้องมีความเร็วมากกว่าทางด้านซ้าย ดังกราฟด้านบน แต่หากเอียงขวา มอเตอร์ตัวด้านขวาต้องมีความเร็วน้อยกว่าทางด้านซ้าย อากาศยานจึงจะเอียงเคลื่อนที่ถูกต้องทิศทาง กำหนดด้านหน้าของอากาศยานสี่ใบพัดคือ แขนสีแดง

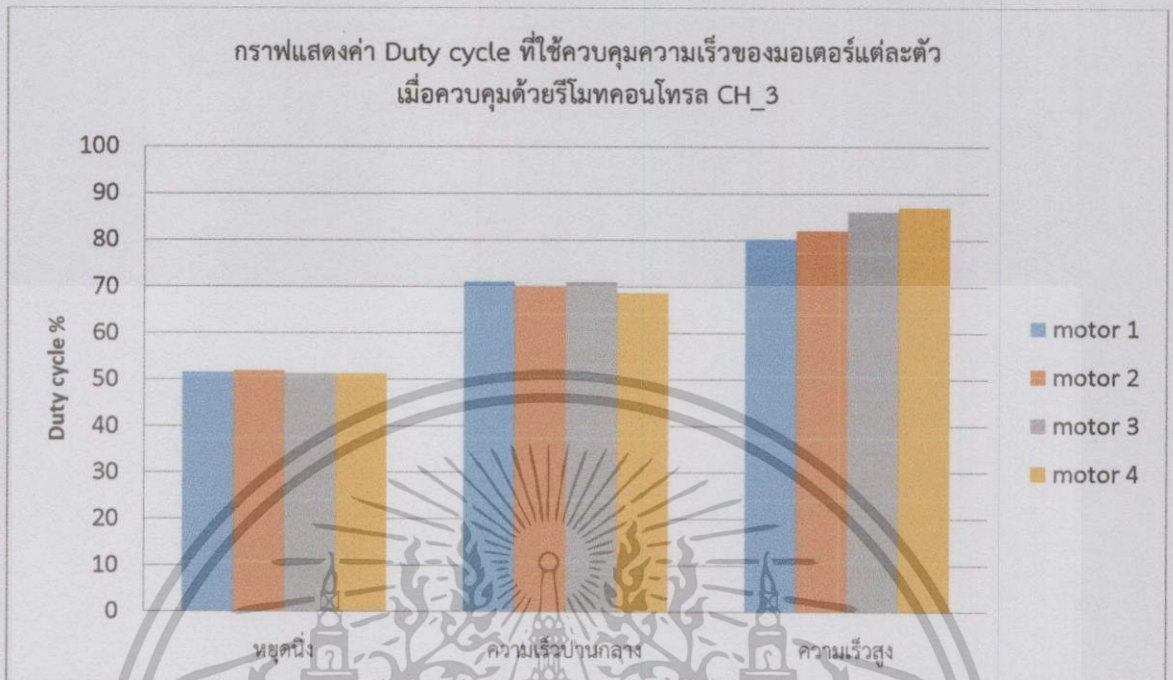
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.20 การเปลี่ยนแปลงความเร็วของมอเตอร์แต่ละตัว (Duty cycle) เมื่อบังคับโดยใช้รีโมทคอนโทรล CH_2

ในรูปที่ 4.20 เห็นได้ว่า CH 2 จะใช้ในการควบคุมการเอียงหน้า เอียงหลังของอากาศยาน 4 ใบพัดเป็นหลัก โดยหากเอียงหน้า มอเตอร์สองตัวด้านหลังต้องมีความเร็วมากกว่าทางด้านหน้า ดังกราฟด้านบน แต่หากเอียงหลัง มอเตอร์ตัวด้านหลังต้องมีความเร็วน้อยกว่าด้านหน้า อากาศยานจึงจะเอียงเคลื่อนที่ถูกทิศทาง กำหนดด้านหน้าของอากาศยานสี่ใบพัดคือ แขนสีแดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 การเปลี่ยนแปลงความเร็วของมอเตอร์แต่ละตัว (Duty cycle) เมื่อบังคับโดยใช้รีโมทคอนโทรล CH 3

ในรูปที่ 4.21 เห็นได้ว่า CH 3 จะใช้ในการควบคุมการบินขึ้น - บินลง ของอากาศยาน 4 ใบพัดเป็นหลัก โดยมอเตอร์ทั้ง 4 ตัวจะหมุนด้วยความเร็วเท่าๆกัน จากกราฟจะเห็นได้ว่า ค่า Duty cycle ที่จ่ายให้กับมอเตอร์แต่ละตัวนั้นมีค่าที่ใกล้เคียงกัน ทำให้อากาศยานสามารถบินขึ้น บินลงได้อย่างถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.14 ผลการทดสอบการบินขึ้น (Take off) และลงจอด (Landing)

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการบินขึ้น (Take off) และลงจอด (Landing) โดยใช้โปรแกรมคำสั่ง
2. เพื่อทดสอบประสิทธิภาพการควบคุมอากาศยาน 4 ใบพัดโดยใช้โปรแกรมคำสั่ง

การทดสอบการบินขึ้น (Take off) และลงจอด (Landing) โดยใช้โปรแกรมคำสั่งนั้นเป็นการทดสอบการใช้งานโปรแกรมสั่งงานมอเตอร์ของอากาศยานบิน 4 ใบพัด ว่าทำงานได้หรือไม่โดยตัวโปรแกรมคำสั่ง บินขึ้น (Take off) และลงจอด (Landing) จะอธิบายตามรูปที่ 3.22 ในบทที่ 3 ซึ่งผลการทดสอบจะแสดงดังรูปที่ 4.22 และ 4.23



รูปที่ 4.22 การทำงานในส่วนของโปรแกรมคำสั่งบินขึ้น (Take off)



รูปที่ 4.23 การทำงานในส่วนของโปรแกรมคำสั่งลงจอด (Landing)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.15 ผลการทดสอบรักษาระดับความสูงการบินของอากาศยาน 4 ใบพัด

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการรักษาระดับความสูงในการบินของอากาศยาน 4 ใบพัด
2. เพื่อทดสอบเปรียบเทียบค่าความสูงจริงกับความสูงของของอากาศยาน 4

ใบพัดเมื่อบินแบบรักษาระดับความสูง

เมื่อทำการทดสอบการบินรักษาระดับความสูง โดยการตั้งกำหนดความสูงไว้ที่ตัวโปรแกรม แล้วให้อากาศยานบินขึ้นเองโดยจะทำฉากบอกระดับความสูงไว้เปรียบเทียบซึ่งอากาศยานจะบินอยู่หน้าฉากบอกระดับ พร้อมทั้งตั้งกล้องให้หันเข้าหาฉากบอกระดับ เพื่อที่จะถ่ายภาพระดับความสูงที่อากาศยานบินได้ แล้วนำภาพถ่ายระดับความสูงที่บินได้เทียบกับระยะความสูงจริงดังรูปที่ 4.24 ซึ่งจะได้ผลดังตารางที่ 4.5





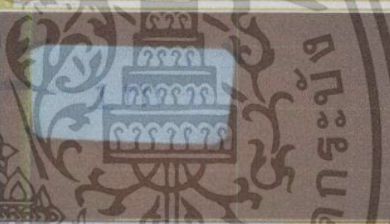

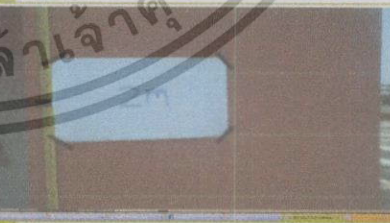
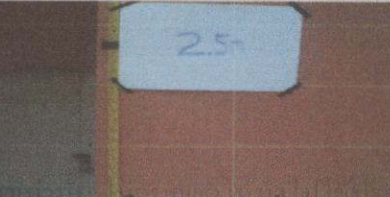
เอกสารนี้เป็น
ไม่ว่ากรณีใดๆ

รูปที่ 4.24 การทดสอบการบินรักษาระดับความสูงที่ความสูงต่างๆ แล้วถ่ายภาพเก็บไว้

นด้านการค้า

นำไปใช้

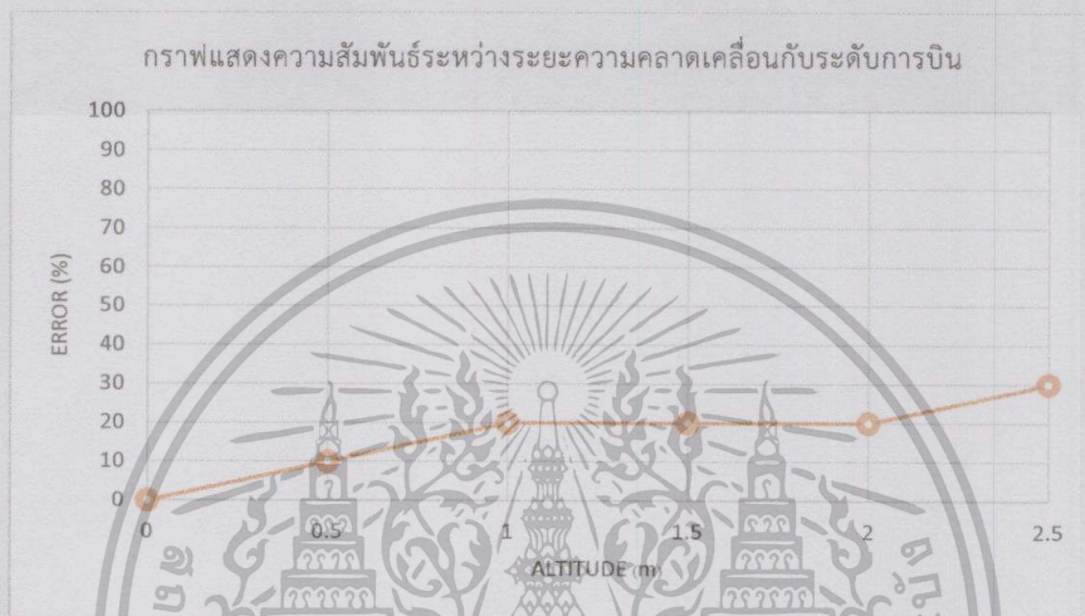
ตารางที่ 4.5 ผลการทดสอบการบิ่นร้าวระดับความสูง

ความสูงที่ตั้งไว้ (m)	ภาพถ่าย
0.00	
0.50	
1.00	
1.50	
2.00	
2.50	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่สามารถนำไปใช้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการทดสอบโปรแกรมรักษาระดับความสูง พบว่าค่าความสูงที่ถ่ายภาพได้นั้นอยู่ในระดับที่ยอมรับได้ อาจมีคลาดเคลื่อนไปบ้างจากลม หรือจากตัวเซ็นเซอร์แต่ถือว่ายอมรับได้เนื่องจากภาพจากระดับความสูงที่ถ่ายได้ยังอยู่ในระดับไม่หลุดจากขอบภาพมากเกินไป



รูปที่ 4.25 กราฟความสัมพันธ์ระหว่างระยะความคลาดเคลื่อนกับระดับการบิน

จากรูปที่ 25 เป็นกราฟความสัมพันธ์ระหว่างระยะความคลาดเคลื่อนกับระดับการบิน โดยได้ทำการทดลองบินอากาศยาน 4 ใบพัด ในระดับความสูงต่าง ๆ ซ้ำกัน 10 ครั้ง ต่อ 1 ระดับความสูง จากกราฟแสดงให้เห็นว่าที่ระดับความสูง 0.5 เมตร มีเปอร์เซ็นต์ความคลาดเคลื่อนเท่ากับ 10% แต่เมื่อระดับการบินสูงขึ้นจะทำให้ความคลาดเคลื่อนของระดับการบินแปรผันตามกัน





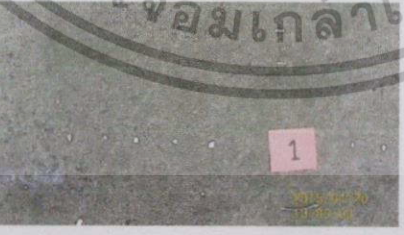
4.16 ผลการทดสอบการเลือกใช้ระดับความสูงในการบันทึกภาพของอากาศยาน 4 ใบพัด

วัตถุประสงค์ของการทดลอง

1. เพื่อใช้ในการเลือกระดับความสูงของอากาศยาน 4 ใบพัด
2. เพื่อกำหนดขอบเขตของภาพที่บันทึกได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 เปรียบเทียบขอบเขตของภาพที่บันทึกได้กับระดับความสูง

ระดับการบิน (m)	ภาพที่ได้รับ	ขนาดภาพ (กว้าง x ยาว) (m)
3		2.8 x 2.1
4		3.7 x 2.8
5		5.0 x 3.75
6		5.6 x 4.2
7		6.5 x 4.9

จากตารางที่ 4.6 จะเห็นได้ว่าที่ระยะความสูง 5 เมตร เป็นระยะที่สามารถบันทึกภาพได้ขนาด 5.0 x 3.75 เมตร ซึ่งเป็นขอบเขตของพิกัดที่ได้ทำการเก็บข้อมูลไว้ ดังนั้นจึงเลือกใช้ระดับความสูงของอากาศยาน 4 ใบพัด ที่ระดับ 5 เมตร เป็นระดับที่ใช้อ้างอิงในการบันทึกภาพ และจากเอกสารที่กล่าวถึงข้างต้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลองที่ 4.15 เมื่อความสูงเพิ่มขึ้นระยะคลาดเคลื่อนของความสูงก็จะแปรผันตาม ทำให้ที่ระดับความสูง 6-7 เมตร มีความคลาดเคลื่อนสูงกว่าจึงไม่เลือกใช้ที่ระดับความสูงดังกล่าว

4.17 ผลการทดสอบเก็บข้อมูล (Data) พิกัดที่ใช้ในการบินของอากาศยาน 4 ใบพัด

วัตถุประสงค์ของการทดลอง

1. เพื่อเก็บข้อมูล (Data) พิกัดขอบเขตของการบินของอากาศยาน 4 ใบพัด
2. เพื่อกำหนดขอบเขตการบินของโปรแกรมการบินแบบระบุพิกัดเป้าหมาย

โดยใช้พิกัดจากฐานข้อมูลที่เก็บได้เป็นข้อมูลอ้างอิง

ทำการเก็บข้อมูลพิกัดของพื้นที่ที่กำหนดไว้เป็นขอบเขตของการบินของอากาศยาน 4 ใบพัด เพื่อใช้เป็นจุดอ้างอิงในการกำหนดจุดบนแผนที่ โดยพื้นที่ที่ใช้ในการเก็บข้อมูลและทดสอบ จะใช้พื้นที่ของสนามกีฬาหลักบ๊อง ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เป็นพื้นที่ทดสอบโดยกำหนดให้แต่ละพิกัดห่างกัน 5 เมตร ในระนาบของทิศเหนือ-ใต้ และตะวันออก-ตะวันตก ซึ่งผลการเก็บข้อมูลพิกัดจะได้ดังรูปที่ 4.25 โดยระยะที่เปลี่ยนแปลงไปทุกๆ 5 เมตร จะมีผลทำให้ค่ามุมของพิกัดละติจูดและลองจิจูด เปลี่ยนไปทุกๆ 0.000045 องศา (1 องศา = 111 km) เทียบกับพิกัดบน Google Map



รูปที่ 4.26 พิกัดข้อมูลพื้นที่ทดสอบที่เก็บได้โดยใช้แผนที่ Google map

4.18 ผลการทดสอบหาค่าพารามิเตอร์ของ PID control ที่ใช้ในการแก้ไขความผิดพลาดของอุปกรณ์ระบุตำแหน่ง (GPS)

วัตถุประสงค์ของการทดลอง

1. เพื่อหาค่าพารามิเตอร์ของการแก้ไขความผิดพลาดของพิกัดโดยใช้ PID

Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการเรียนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
2. เพื่อทดสอบระยะเวลาความผิดพลาดของพิกัดที่ค่าพารามิเตอร์ต่างๆ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 ความสัมพันธ์ระหว่าง ค่า K_p กับระยะความคลาดเคลื่อนของตำแหน่ง
ปลายทาง เมื่อกำหนด ค่า K_i และ K_d เท่ากับ 0

ครั้งที่	ระยะที่ผิดพลาด(m)					
	$K_p = 0$	$K_p = 0.1$	$K_p = 0.2$	$K_p = 0.3$	$K_p = 0.4$	$K_p = 1.5$
1	1.30	2.43	154	122	125	134
2	1.25	1.31	87	140	177	93
3	5.10	3.42	43	133	187	178
4	2.90	2.42	159	167	55	80
5	3.77	1.76	76	57	89	99
6	2.47	2.87	134	185	78	125
7	1.93	1.63	46	134	64	106
8	4.68	1.35	355	158	58	145
9	1.27	1.98	120	167	135	84
10	3.12	2.77	233	78	168	146
เฉลี่ย	2.779	2.194	1.407	1.341	1.136	1.19

จากตารางที่ 4.7 ทำการทดสอบใช้ระบบ P-control เพื่อหาค่า P_c ตามเทคนิควิธี
การปรับแต่งค่า PID control ในบทที่ 2 โดยในระบบ P-control K_c หาได้จาก $\frac{K_p}{0.5}$ ซึ่งจากผล
การทดลองในตารางที่ 4.7 ค่า K_p ที่ดีที่สุดเท่ากับ 0.4 ดังนั้น ค่า K_c ที่ใช้จึงเท่ากับ 0.8 เมื่อได้ค่า K_c
จะต้องคำนวณหาค่า P_c โดยการใช้ระบบ PI-control ซึ่งจากเทคนิคการปรับแต่งค่า PID-control
ในบทที่ 2 จะกำหนด K_p ของระบบ PI-control เท่ากับ $0.45K_c$ และทำการทดสอบปรับหาค่า K_i
ที่ดีที่สุด แล้วนำมาคำนวณหาค่า P_c

ตารางที่ 4.8 ความสัมพันธ์ระหว่าง K_i กับระยะผิดพลาด เมื่อกำหนดให้ $K_p = 0.36$
($0.45K_c$) และ $K_d = 0$

ครั้งที่	ระยะที่ผิดพลาด (m)				
	$K_i = 0.1$	$K_i = 0.2$	$K_i = 0.3$	$K_i = 0.4$	$K_i = 0.5$
1	0.55	1.24	1.55	1.34	0.98
2	1.30	0.45	1.45	0.54	1.34
3	1.25	1.39	1.37	0.87	0.97
4	0.59	1.25	1.54	1.85	1.23
5	0.85	0.78	0.24	0.83	2.12
6	1.78	1.39	1.40	1.34	1.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 ความสัมพันธ์ระหว่าง K_i กับระยะผิดพลาด เมื่อกำหนดให้ $K_p = 0.36$ ($0.45K_c$) และ $K_d = 0$ (ต่อ)

ครั้งที่	ระยะที่ผิดพลาด (m)				
	$K_i = 0.1$	$K_i = 0.1$	$K_i = 0.1$	$K_i = 0.1$	$K_i = 0.1$
7	0.23	0.98	0.86	1.65	1.28
8	1.45	0.79	0.68	1.17	1.52
9	0.85	1.34	0.90	0.98	0.38
10	1.60	1.46	1.32	1.34	1.71
เฉลี่ย	1.045	1.107	1.131	1.191	1.278

จากตารางที่ 4.8 จะได้ว่า K_i ที่ดีที่สุดเมื่อใช้ระบบแก้ไขความผิดพลาดแบบ PI-control เท่ากับ 0.1 เมื่อนำมาคำนวณหาค่า P_c ตามเทคนิคการปรับแต่งค่า PID-control ในบทที่ 2 ซึ่งหาได้จาก $\frac{1.2K_p}{K_i}$ จะได้ P_c เท่ากับ 4.32

เมื่อได้ค่า K_p และ P_c ตามสมการการปรับแต่งค่า PID-control ในบทที่ 2 สามารถคำนวณหาค่า K_d มาใช้ในการชดเชยความเสถียรภาพของระบบ PID-control ซึ่งในระบบ PID-control ค่า K_p จะเท่ากับ $0.6K_c$, ค่า K_i เท่ากับ $\frac{2K_p}{P_c}$ และ K_d เท่ากับ $\frac{K_p P_c}{2}$ ในที่นี้ จะได้ค่า K_p เท่ากับ 0.48, K_i เท่ากับ 0.22 และ K_d เท่ากับ 0.26 ซึ่งเป็นค่าพารามิเตอร์ที่ใช้เป็น Gain ของ PID-control ของระบบการบินอากาศยาน 4 ใบพัด

4.19 ผลการทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 1 จุดโดยกำหนดค่าความสูงของอากาศยานคงที่

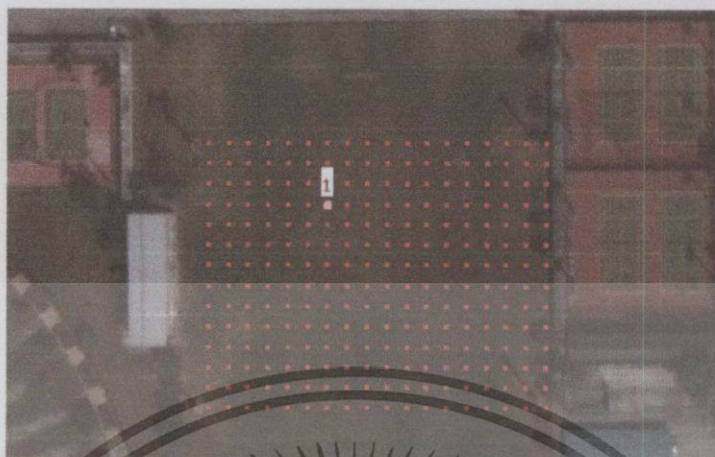
วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการทำงานของโปรแกรมควบคุมการบินแบบกำหนดเป้าหมาย

2. เพื่อทดสอบหาอัตราส่วนความผิดพลาดในการบินไปสู่เป้าหมายที่ถูกต้อง อ้างอิงจากภาพเป้าหมายที่ถ่ายได้จากกล้องที่ติดไว้บนอากาศยานบิน 4 ใบพัด

ทำการทดสอบการทำงานของโปรแกรมควบคุมการบินแบบกำหนดเป้าหมาย โดยกำหนดให้พิกัดเป้าหมายเป็นพิกัด ละติจูดที่ 13.731051 และลองจิจูดที่ 100.773606 ซึ่งอยู่ในขอบเขตของสนามล็กบี้ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ดังรูปที่ 4.26 โดยกำหนดความสูงที่ใช้บินคือ 5 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะในรูปแบบใดทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.27 พิกัดเป้าหมายที่ใช้ในการทดสอบการบินแบบ 1 เป้าหมาย

การที่อากาศยานจะบินไปยังเป้าหมาย ตัวอากาศยานจะทำการคำนวณหามุมที่จะไปยังเป้าหมาย จากทฤษฎีของตรีโกณมิติก่อน รวมถึงคำนวณระยะทางที่ห่างจากจุดเป้าหมาย แล้วจึงบินไปยังเป้าหมาย ซึ่งผลการทดสอบโปรแกรมควบคุมการบินไปยังเป้าหมาย 1 ตำแหน่งแสดงได้ดังรูปที่ 4.28 และ รูปที่ 4.29



รูปที่ 4.28 อากาศยานขณะบินไปยังเป้าหมายโดยใช้โปรแกรมควบคุมการบิน



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสถาบันวิจัยและพัฒนาเทคโนโลยีการบินและอวกาศแห่งชาติ ซึ่งถูกนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น รูปที่ 4.29 ภาพที่เป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด



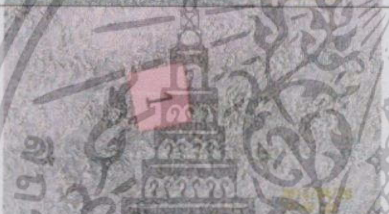



จากการทดสอบจะเห็นได้ว่าโปรแกรมควบคุมการบินสามารถทำงานได้อย่างมีประสิทธิภาพ ซึ่งในทดลองบินซ้ำเป็นจำนวน 30 ครั้ง แบบสุ่มจุดบินขึ้น พบว่าผลของภาพเป้าหมายที่กล้องบนอากาศยาน 4 ใบพัดถ่ายไว้ได้นั้น สามารถแสดงได้ดังตารางที่ 4.9 ซึ่งจะใช้ในการหาประสิทธิภาพของตัวโปรแกรมควบคุม

ตารางที่ 4.9 ภาพเป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด เมื่อบินไปถึงเป้าหมาย

ครั้งที่	ภาพที่บันทึกได้	ระยะที่ผิดพลาด (m)	True	False
1		0.51	✓	
2		1.13	✓	
3		1.95		✓
4		1.68	✓	
5		0.96	✓	






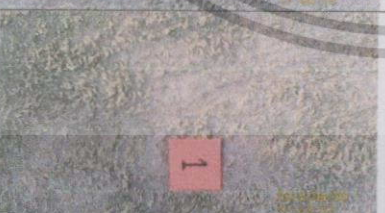
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

ตารางที่ 4.9 ภาพเป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด
เมื่อบินไปถึงเป้าหมาย (ต่อ)

ครั้งที่	ภาพที่บันทึกได้	ระยะที่ผิดพลาด (m)	True	False
6		1.08	✓	
7		0.72	✓	
8		0.43	✓	
9		0.85	✓	
10		0.86	✓	
11		0.83	✓	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.9 ภาพเป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด
เมื่อบินไปถึงเป้าหมาย (ต่อ)

ครั้งที่	ภาพที่บันทึกได้	ระยะที่ผิดพลาด (m)	True	False
12		0.90	✓	
15		1.90		✓
14		1.76		✓
15		1.07		
16		0.77	✓	
17		1.03	✓	


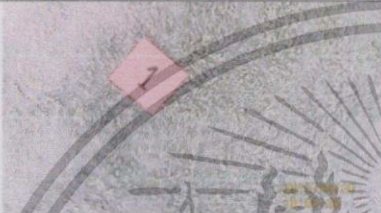
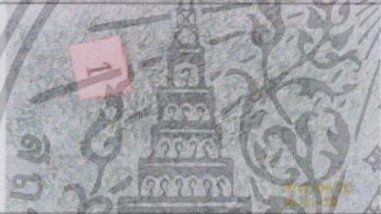


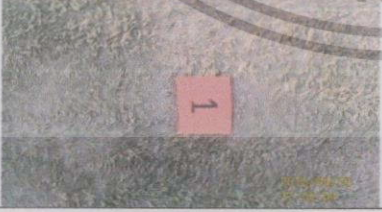
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.9 ภาพเป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด
เมื่อบินไปถึงเป้าหมาย (ต่อ)

ครั้งที่	ภาพที่บันทึกได้	ระยะที่ผิดพลาด (m)	True	False
18		1.91		✓
19		1.24	✓	
20		2.30		✓
21		0.71	✓	
22		0.42	✓	
23		1.03	✓	
24		0.46	✓	

เอกสารนี้เป็นเอกสารของกองทัพอากาศไทย การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำมาใช้

ตารางที่ 4.9 ภาพเป้าหมายที่บันทึกได้จากกล้องที่ติดอยู่บนอากาศยาน 4 ใบพัด
เมื่อบินไปถึงเป้าหมาย (ต่อ)

ครั้งที่	ภาพที่บันทึกได้	ระยะที่ผิดพลาด (m)	True	False
25		1.03	✓	
26		1.26	✓	
27		1.27	✓	
28		0.55	✓	
29		0.59	✓	
30		0.85	✓	

จากการทดสอบพบว่าอัตราส่วนของการบินไปยังเป้าหมายที่เกิดความคลาดเคลื่อนจากการ
ทดลองบิน 30 ครั้ง พบว่ามี 25 ครั้งที่พบภาพเป้าหมาย และ 5 ครั้งที่ไม่เห็นภาพเป้าหมาย ซึ่งคิด
เป็นเปอร์เซ็นต์ของความผิดพลาดได้ 16.67% โดยระยะผิดพลาดเฉลี่ยเท่ากับ 1.07 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น มิใช่ให้ผู้ใดนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.20 การทดสอบการบินของอากาศยาน 4 ใบพัด ไปยังเป้าหมาย 5 จุด โดยกำหนดค่าความสูงของอากาศยานคงที่

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบโปรแกรมควบคุมการบินแบบกำหนดเป้าหมายต่อเนื่อง 5 จุด

ทำการทดสอบการทำงานของโปรแกรมควบคุมการบินแบบกำหนดเป้าหมายต่อเนื่อง 5 จุด โดยกำหนดให้พิกัดเป้าหมายแต่ละจุดคือ (13.731051, 100.773606), (13.731051, 100.773831), (13.730781, 100.773831), (13.730781, 100.773606) และ (13.730916, 100.773471) ตามลำดับ ซึ่งอยู่ในขอบเขตของสนามลักบี้ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ดังรูปที่ 4.30 โดยกำหนดระดับการบินที่ความสูง 5 เมตรซึ่งจะได้ผลการทดสอบดังรูปที่ 4.31 และ 4.32

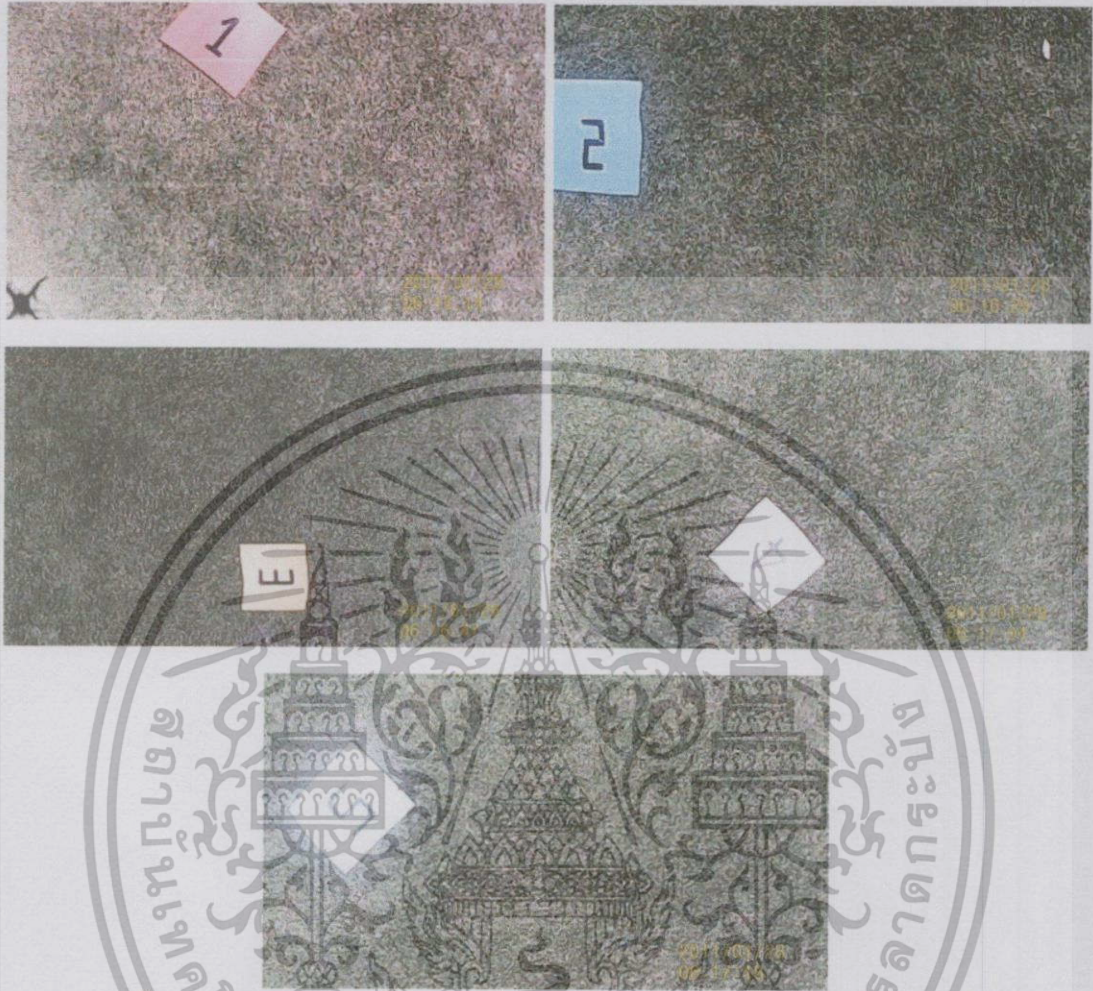


รูปที่ 4.30 พิกัดเป้าหมายที่ใช้ในการทดสอบการบินแบบ 5 เป้าหมาย ต่อเนื่อง



รูปที่ 4.31 ภาพขณะที่อากาศยานกำลังบินไปยังเป้าหมายทั้ง 5 จุด โดยใช้โปรแกรมควบคุมการบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีค่านำไปใช้



รูปที่ 4.32 ภาพเป้าหมายทั้ง 5 จุด ที่กล้องบนอากาศยานบันทึกภาพไว้ได้ขณะบิน

จากรูปที่ 4.32 เป็นภาพเป้าหมายทั้ง 5 จุด ที่บันทึกได้จากกล้องที่ติดตั้งไว้บนตัวอากาศยาน 4 ใบพัด ซึ่งเป็นการบันทึกภาพที่ระยะความสูง 5 เมตร และให้รูปที่มีขนาดของภาพครอบคลุมพื้นที่พิกัดที่เหมาะสมที่สุด ตามที่ได้ทำการทดลองในการทดลองที่ 3.3.16 ซึ่งแสดงให้เห็นว่าหากอากาศยาน 4 ใบพัดบินสูงขึ้นจะทำให้เกิดความคลาดเคลื่อนที่แปรผันตามกัน และถ้าหากภาพที่บันทึกได้นั้นมีวัตถุประสงค์เป้าหมายให้เห็นชัดเจนไม่แสดงว่าการบินในครั้งนั้นมีความแม่นยำ ซึ่งถือเป็นเกณฑ์มาตรฐานในการตัดสินความถูกต้อง รวมทั้งใช้รูปจากการทดลองที่ 3.3.16 ที่ระยะ 5 เมตร เป็นภาพอ้างอิงความถูกต้องรวมด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีค่านำไปใช้

4.21 ผลการทดสอบการใช้งานกำหนดเป้าหมายการบินของแอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์

วัตถุประสงค์ของการทดลอง

1. เพื่อทดสอบการใช้งานร่วมกับแอปพลิเคชัน
2. เพื่อทดสอบการกำหนดเป้าหมายผ่านแอปพลิเคชัน



รูปที่ 4.33 การกำหนดพิกัด 5 พิกัด ผ่านแอปพลิเคชันแอนดรอยด์

จากรูปที่ 4.33 เป็นหน้าต่างแสดงการกำหนดพิกัด 5 พิกัด โดยใช้แอปพลิเคชันแอนดรอยด์ เพื่อให้อากาศยานบินไปตามพิกัดที่กำหนดเอาไว้ ซึ่งบนหน้าจอแอปพลิเคชันสามารถระบุได้ทั้งตำแหน่งเป้าหมาย รวมถึงความสูง และระยะเวลาที่บินได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีคนนำไปใช้

บทที่ 5

สรุปผลและข้อเสนอแนะ

5.1 สรุปผล

โครงการนี้เป็นการออกแบบ สร้าง และทดสอบโปรแกรมควบคุมอากาศยาน 4 ใบพัด ว่าสามารถเคลื่อนที่ได้อย่างไร และต้องทำงานผ่านอุปกรณ์ตัวใดบ้าง โดยเริ่มจากการทดสอบความพร้อมของอุปกรณ์ และเซ็นเซอร์ต่างๆ เช่น เซ็นเซอร์อิเล็กทรอนิกส์ อุปกรณ์ระบุตำแหน่ง (GPS), ไจโรสโคป และรีโมทคอนโทรล เป็นต้น จากนั้นจึงทำการตรวจสอบค่าตัวแปร (Parameter) ผ่านโปรแกรม Mission Planner 1.3.9 เพื่อใช้ในการกำหนดค่าพื้นฐานต่างๆ ที่เป็นตัวกำหนดการบิน รวมทั้งการลงเฟิร์มแวร์ต่างๆ ซึ่งเป็นประโยชน์ต่อการเคลื่อนที่ของอากาศยาน 4 ใบพัด ต่อมาได้ทำการพัฒนาต่อโดยใช้โปรแกรมควบคุมอากาศยาน 4 ใบพัดเพื่อให้สามารถเคลื่อนที่ไปยังพิกัดเป้าหมายได้อัตโนมัติ โดยจะใช้แอปพลิเคชันในระบบปฏิบัติการแอนดรอยด์ในการกำหนดพิกัดเป้าหมาย รวมทั้งระบุความสูงที่ใช้ในการบินของอากาศยาน 4 ใบพัดในแต่ละครั้ง

5.2 ข้อเสนอแนะ

- 1) ควรทดสอบการบินจริงควรทดสอบในบริเวณโล่ง และมีสภาพอากาศที่เหมาะสม
- 2) ไม่ควรใส่ใบพัดในระหว่างการจัดสร้าง และกำหนดค่าเริ่มต้นต่างๆ
- 3) สำหรับมือสมัครเล่น ควรทดสอบการบินบริเวณสนามหญ้า และห่างจากแหล่งน้ำ
- 4) อุปกรณ์ที่ติดตั้งบนอากาศยานควรมีน้ำหนักเบา ไม่ต้านลม
- 5) ควรตรวจสอบความพร้อมของอุปกรณ์ก่อนทำการบินทุกครั้ง

5.3 แนวทางการพัฒนา

- 1) ประยุกต์ใช้เทคโนโลยีในด้านต่างๆ ให้เกิดความหลากหลาย และมีประสิทธิภาพ อาทิเช่น ประยุกต์ใช้ในการขนส่งวัตถุที่น้ำหนักเบา เป็นต้น
- 2) พัฒนาโปรแกรมในการรักษาเสถียรภาพ อาทิ การวัดแรงลม และทิศทางของลม เพื่อช่วยในการรักษาความสมดุลของอากาศยาน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- [1] ชงควัฒน์ ศรีจันตะ และคณะ. “อากาศยานปีกหมุน 4 ใบพัด.” วิทยานิพนธ์ปริญญาวิศวกรรมศาสตรบัณฑิต, สาขาวิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์, มหาวิทยาลัยขอนแก่น, 2556.
- [2] ปฏิพัทธ์ ญาติโสม และเมธาวุฒิ อุดรินทร์. “เฮลิคอปเตอร์ 4 ใบพัด.” วิทยานิพนธ์ปริญญาวิศวกรรมศาสตรบัณฑิต, สาขาวิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์, มหาวิทยาลัยขอนแก่น, 2555.
- [3] จิตพล ศรีสว่าง, สวรรค์ ไพฑูรย์วงศ์ และอัครพันธุ์ สังข์ช้าง. “เครื่องบินอัตโนมัติแบบสี่ใบพัดสำหรับการถ่ายภาพทางอากาศ.” วิทยานิพนธ์ปริญญาวิศวกรรมศาสตรบัณฑิต, สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์, มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี, 2555.
- [4] มุมทิศ [ออนไลน์]. เข้าถึงได้จาก: <http://th.wikipedia.org/wiki/มุมทิศ> (วันที่ค้นข้อมูล 22 พฤศจิกายน 2557).
- [5] การเชื่อมต่ออุปกรณ์แบบ I2C (I²C) [ออนไลน์]. เข้าถึงได้จาก: <http://www.thaimicrotron.com/CCS-628/Reference/I2CBUS.htm> (วันที่ค้นข้อมูล 13 พฤศจิกายน 2557).
- [6] Accelerometer [ออนไลน์]. เข้าถึงได้จาก: <http://bme231metrology.blogspot.com/2011/07/accelerometer-iphone-accelerometer.html> (วันที่ค้นข้อมูล 15 พฤศจิกายน 2557).
- [7] Plastic Case for APM 2.5.2 [ออนไลน์]. เข้าถึงได้จาก: <http://portal.weloveshopping.com/product/W25467210> (วันที่ค้นข้อมูล 15 พฤศจิกายน 2557).
- [8] หลักการทำงานของ Gyroscope [ออนไลน์]. เข้าถึงได้จาก: http://dk.coe.psu.ac.th/assign/gyro/intro_gyro1.htm (วันที่ค้นข้อมูล 20 พฤศจิกายน 2557).
- [9] 433 MHz Radio Telemetry Kit [ออนไลน์]. เข้าถึงได้จาก: <http://rctimer.com/product-834.html> (วันที่ค้นข้อมูล 20 พฤศจิกายน 2557).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกัรนำไปใช้

บรรณานุกรม (ต่อ)

- [10] ทบทวนความรู้: คอนโทรลเลอร์แบบ PID [ออนไลน์]. เข้าถึงได้จาก: <http://www.navy.mi.th/elecwww/magaz/magazine/no16/18.pdf> (วันที่ค้นข้อมูล 16 มีนาคม 2557).
- [11] The Terrestrial Navigational Triangle [ออนไลน์]. เข้าถึงได้จาก: <http://www.erikdeman.de/html/sail042e.htm> (วันที่ค้นข้อมูล 9 เมษายน 2558).
- [12] Geographic Distance and Azimuth Calculations [ออนไลน์]. เข้าถึงได้จาก: <http://www.codeguru.com/cpp/cpp/algorithms/article.php/e5115/Geographic-Distance-and-Azimuth-Calculations.htm> (วันที่ค้นข้อมูล 9 เมษายน 2558).
- [13] Upgrading My Quad copter from Kk2.0 to APM 2.5 [ออนไลน์]. เข้าถึงได้จาก: <http://quadcoptergarage.com/upgrading-my-quadcopter-from-kk2-to-apm-2-5/> (วันที่ค้นข้อมูล 22 พฤศจิกายน 2557).
- [14] APM 2.5 ArduPilot Mega 2.5 Multicopter Flight Control Board w/ MTK GPS Module [ออนไลน์]. เข้าถึงได้จาก: <http://www.goodluckbuy.com/apm-2-5-ardupilot-mega-2-5-multicopter-flight-control-board-w-mtk-gps-module.html> (วันที่ค้นข้อมูล 22 พฤศจิกายน 2557).
- [15] Setting correct motor orientation [ออนไลน์]. เข้าถึงได้จาก: <http://www.arducopter.co.uk/whats-new/arducopter-3dr-assembly-instructions> (วันที่ค้นข้อมูล 18 มกราคม 2558).
- [16] Why Should Choose the FlySky FS TH9X Transmitter Mode 2 [ออนไลน์]. เข้าถึงได้จาก: <http://adapter-converter.blogspot.com/2013/01/why-should-choose-flysky-fs-th9x.html> (วันที่ค้นข้อมูล 18 มกราคม 2558).
- [17] NEO-7N High Quality GPS Module with 3 Axis Compass for APM 2.6 and Pixhawk [ออนไลน์]. เข้าถึงได้จาก: http://www.queenhobby.com/PD984094-สินค้า-neo_7n_high_quality_gps_module_with_3_axis_compass_for_apm_2_6_and_pixhawk.html (วันที่ค้นข้อมูล 18 มกราคม 2558).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีค่านำไปใช้

บรรณานุกรม (ต่อ)

- [18] Connecting everything for Arducopter [ออนไลน์]. เข้าถึงได้จาก: <http://www.arducopter.co.uk/all-arducopter-guides/2connectingeverything-for-arducopter> (วันที่ค้นข้อมูล 18 มกราคม 2558).
- [19] การอบรมสัมมนาเชิงปฏิบัติการเรียนรู้และฝึกทักษะด้านระบบสมองกลฝังตัวของอากาศยาน (Quadrotor Controller) [ออนไลน์]. เข้าถึงได้จาก: <https://drive.google.com/file/d/0B4UGzNJ0SnKmMGIFTXp1em1jdEK/view> (วันที่ค้นข้อมูล 2 กุมภาพันธ์ 2558).
- [20] Controlling a QuadCopter with code [ออนไลน์]. เข้าถึงได้จาก: <http://diydrones.com/forum/topics/controlling-a-quadcopter-with-code> (วันที่ค้นข้อมูล 17 กุมภาพันธ์ 2558).
- [21] How to Android [ออนไลน์]. เข้าถึงได้จาก: <http://www.akexorcist.com/> (วันที่ค้นข้อมูล 7 มกราคม 2558).
- [22] Android tutorials [ออนไลน์]. เข้าถึงได้จาก: <http://www.thaicreate.com/mobile/android.html> (วันที่ค้นข้อมูล 13 มกราคม 2558).
- [23] เครื่องมือพัฒนา แอปแอนดรอยด์ (Android Studio) [ออนไลน์]. เข้าถึงได้จาก: <https://www.gotoknow.org/posts/548209> (วันที่ค้นข้อมูล 3 มีนาคม 2558).
- [24]] DroidPlanner/Tower [Android Studio] [ออนไลน์]. เข้าถึงได้จาก: <https://github.com/DroidPlanner/Tower> (วันที่ค้นข้อมูล 3 มีนาคม 2558).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

////////////////////////////////////Code main //////////////////////////////////////
#include <Arduino.h>
#include <Wire.h>
#include "MS561101BA.h"
#include "config.h"
#include "multi_rx2560.h"
#include "mpu6050.h"
#include "ahrs_tin.h"
#include "Control_PID.h"
#include "motorX4.h"
#include "GPS_multi.h"
#include "Ultrasonic.h"

float getAltitude(float pressure2, float temperature2)
{
  //return (1.0f - pow(press/101325.0f, 0.190295f)) * 4433000.0f;
  return log(sea_press/pressure2) * (temperature2+273.15) * 29.271267f; // in meter
  //return ((pow((sea_press/pressure),1/5.257)-1.0)*(temperature+273.15))/0.0065;
}
void pushAvg(float val)
{
  movavg_buff[movavg_i] = val;
  movavg_i = (movavg_i + 1) % MOVAVG_SIZE;
}
float getAvg(float * buff, int size)
{
  float sum=0.0;
  for(int i=0;i<size;i++)
  {
    sum += buff[i];
  }
  return sum/size;
}

}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
void setup()
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
{
  Serial.begin(57600); //38400

```

```

pinMode(13, OUTPUT);pinMode (30, OUTPUT);pinMode (31, OUTPUT);
//pinMode (30, OUTPUT);pinMode (31,
OUTPUT);//(13=A=M),(31=B=STABLEPIN),(30=C,GPS FIG LEDPIN)digitalWrite(13, HIGH);
//Serial1.begin(115200);//CRIUS Bluetooth Module pin code 0000
//Serial3.begin(38400);//3DR Radio Telemetry Kit 433Mhz
configureReceiver();//find multi_rx.h
motor_initialize();//find motor.h
ESC_calibration();//find motor.h
GPS_multilnt();
Wire.begin();
delay(1);
mpu6050_initialize();
delay(1);
MagHMC5883Int();
delay(1);
digitalWrite(13, HIGH);
baro.init(MS561101BA_ADDR_CSB_LOW);
UltrasonicInt();
TWBR = ((F_CPU / 400000L) - 16) / 2; // change the I2C clock rate to 400kHz
delay(1);
for(uint8_t i=0; i<50; i++)
{
mpu6050_Gyro_Values();
mpu6050_Accel_Values();
Mag5883Read();
UltrasonicRead();
temperature = baro.getTemperature(MS561101BA_OSR_4096);
presser = baro.getPressure(MS561101BA_OSR_4096);
pushAvg(presser);
delay(20);
}
//Altitude_Ground = Altitude_baro/10.0;
sea_press = presser + 0.11;//presser 1003.52
Serial.print("presser ");Serial.println(sea_press);
digitalWrite(13, LOW);
sensor_Calibrate();//sensor.h

```

เอกสารนี้เป็นลิขสิทธิ์งานวิจัยของศูนย์วิจัยและพัฒนาอากาศยานไทย ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ahrs_initialize();//ahrs.h
setupFourthOrder();//ahrs
RC_Calibrate();//"multi_rxPPM2560.h"
Serial.print("TK_Quadrotor_Run_Roop_100Hz");Serial.println("\t");
sensorPreviousTime = micros();
previousTime = micros();
}
void loop()
{
  while(Serial2.available())
  {
    //digitalWrite(30, LOW);//C
    char byteGPS1=Serial2.read();
    GPS_UBLOX_newFrame(byteGPS1);
    GPS_LAT1 = GPS_coord[LAT]/10000000.0;// 1e-7 degrees / position as degrees
    (*10E7)
    GPS_LON1 = GPS_coord[LON]/10000000.0;
    //if(GPS_FIX == 1 && GPS_Present == 1){
    //digitalWrite(30, HIGH);
    //}
    //}end gps
    Dt_sensor = micros() - sensorPreviousTime;
    ////////////Roop sensor//////////
    if(Dt_sensor <= 0)
    {
      Dt_sensor = 1001;
    }
    if(Dt_sensor >= 1000 && gyroSamples < 4)////Collect 3 samples = 2760 us &&
    gyroSamples < 5 && gyroSamples < 5
    {
      sensorPreviousTime = micros();
      mpu6050_readGyroSum();
      mpu6050_readAccelSum();
    }
    Dt_roop = micros() - previousTime;// 100 Hz task loop (10 ms) , 5000 = 0.02626 ms
    if(Dt_roop <= 0)

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
Dt_roop = 10001;
}
if (Dt_roop >= 10000)
{
previousTime = micros();
G_Dt = Dt_roop*0.000001;
frameCounter++;
mpu6050_Get_accel();
mpu6050_Get_gyro();
//////////Moving Average Filters//////////
GyroXf = (GyroX + GyroX2)/2.0;
//GyroXf = GyroXf + (GyroX - GyroXf)*15.6*G_Dt;
GyroYf = (GyroY + GyroY2)/2.0;
GyroZf = (GyroZ + GyroZ2)/2.0;
GyroX2 = GyroX;GyroY2 = GyroY;GyroZ2 = GyroZ;//gyro_Old1
//////////Low pass filter//////////
AccXf = AccX;
AccYf = AccY;
AccZf = AccZ;
//AccXf = AccXf + (AccX - AccXf)*15.6*G_Dt;//29.6 15.4 //Low pass filter
,smoothing factor  $\alpha := dt / (RG + dt)$ 
//AccYf = AccYf + (AccY - AccYf)*15.6*G_Dt;//15.4
//AccZf = AccZf + (AccZ - AccZf)*15.6*G_Dt;//15.4
//////////Filter FourthOrder //////////
Accel[XAXIS] = AccX;
Accel[YAXIS] = AccY;
Accel[ZAXIS] = AccZ;
for (int axis = XAXIS; axis <= ZAXIS; axis++) {
filteredAccel[axis] = computeFourthOrder(Accel[axis],
&fourthOrder[axis]);// "ahrs_tin.h"
}
//AccXf = filteredAccel[XAXIS];
//AccYf = filteredAccel[YAXIS];
//AccZf = filteredAccel[ZAXIS];
//////////

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//ahrs_updateMARG(GyroXf, GyroYf, GyroZf, AccXf, AccYf, AccZf, c_magnetom_x,
c_magnetom_y, c_magnetom_z, G_Dt); //quaternion ,direction cosine matrix ,Euler
angles
```

```
ahrs_updateMARG(GyroXf, GyroYf, GyroZf, filteredAccel[XAXIS],
filteredAccel[YAXIS], filteredAccel[ZAXIS], c_magnetom_x, c_magnetom_y,
c_magnetom_z, G_Dt);
```

```
//x_angle = x_angle + (GyroXf*RAD_TO_DEG*G_Dt);
```

```
//x_angle = kalmanCalculateX(ahrs_r*RAD_TO_DEG, GyroX*RAD_TO_DEG, G_Dt);
```

```
//y_angle = kalmanCalculateY(ahrs_p*RAD_TO_DEG, GyroY*RAD_TO_DEG, G_Dt);
```

```
///Observer velocity vx vy vz kalman//accelerometer and
```

```
GPS////////////////////////////////////
```

```
//GPS Speed Low Pass Filter
```

```
//actual_speedXf = actual_speedXf + (actual_speedX -
actual_speedXf)*10.2*G_Dt;//5.2 10.4 //cm/s +-400 cm/s
```

```
//actual_speedYf = actual_speedYf + (actual_speedY -
actual_speedYf)*10.2*G_Dt;//5.2 10.17
```

```
//GPS_LAT1Lf = GPS_LAT1Lf + (GPS_LAT1lead -
GPS_LAT1Lf)*8.2521*G_Dt;//12.5412
```

```
//GPS_LON1Lf = GPS_LON1Lf + (GPS_LON1lead - GPS_LON1Lf)*8.2521*G_Dt;
```

```
float temp_vx = accrX_Earth*100.0 + (_velocity_north - vx_hat2)*13.2;//2.15 1.5
6.5 vx_hat = vx_hat + temp_vx*G_Dt;
```

```
vx_hat2 = vx_hat2 + temp_vx*G_Dt;
```

```
vx_hat = constrain(vx_hat2, -400, 400); //+-400 cm/s
```

```
applyDeadband(vx_hat, 4.5); //10.5 cm/s
```

```
float temp_vy = accrY_Earth*100.0 + (_velocity_east - vy_hat2)*13.2;//2.15 1.5
6.5
```

```
vy_hat2 = vy_hat2 + temp_vy*G_Dt;
```

```
vy_hat = constrain(vy_hat2, -400, 400);
```

```
applyDeadband(vy_hat, 4.5); //10.5 cm/s
```

```
////////////////////////////////////
```

```
//Observer hz kalman , GPS_hz , GPS_vz
```

```
float temp_vz = accrZ_Earth + (baro_vz - vz_hat2)*0.45 + (Altitude_barof -
Altitude_hat)*3.72;//Altitude_barof Altitude_Baro_ult 4.72 ,,0.95
```

```
vz_hat2 = vz_hat2 + temp_vz*G_Dt;
```

```
vz_hat = constrain(vz_hat2, -4, 4); //+-4 m/s
```

```

applyDeadband(vz_hat, 0.04);//+-0.01 m/s
float temp_hz = vz_hat + (Altitude_barof - Altitude_hat)*6.72;//4.72 12.54 4.5
Altitude_hat = Altitude_hat + temp_hz*G_Dt;
//Altitude_hat = constrain(Altitude_hat, 0.0, 1.2);//1.5 m By Ultrasonic
//vz = vz + accrz_Earth*G_Dt;
    baro_vz = (Altitude_hat - baro_vz_old2)/0.02;//Diff Altitude
    baro_vz_old2 = baro_vz_old;
    baro_vz_old = Altitude_hat;
////////////////////////////////////

//PID Control////////////////////////////////
    Control_PIDRate();//Control_PID.h
/////Out motor////////////////////////////////
//armed = 1;
    motor_Mix();//"motor.h"
////////////////////////////////////
    motor_command();
/////end Out motor/////
if (frameCounter % TASK_50HZ == 0)// 50 Hz tak (20 ms)
{
    computeRC();//multi_rx.h
    //failsafeCnt++;
    //Fail_Safe();
    if (CH_THR < MINCHECK) //ARM and DISARM your Quadrotor////////////////////////////////
    {
        if (CH_RUD > MAXCHECK && armed == 0 && abs(ahrs_p) < 10 && abs(ahrs_r)
< 10)//+- 10 deg, ARM
        {
            armed = 1;
            digitalWrite(31, HIGH);//B
            setHeading = ahrs_y;// 0 degree ,ahrs_tin.h
        }
        if (CH_RUD < MINCHECK && armed == 1) //DISARM
        {
            armed = 0;
            digitalWrite(31, LOW);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การขงมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Altitude_Ground = Altitude_baro;
    Altitude_II = 0.17;
}
if (CH_RUD < MINCHECK && armed == 0 && CH_ELE > MAXCHECK)
//Mag_Calibrate
{
    Mag_Calibrate();//#include "mpu6050.h"
}
} //end ARM and DISARM your helicopter////////////////////
} //end roop 50 Hz
presser = baro.getPressure(MS561101BA_OSR_4096);//read 100 Hz
pushAvg(presser);//
if (frameCounter % TASK_20HZ == 0) // 20 Hz task (50 ms)
{
    UltrasonicRead();//"Ultrasonic.h"
    Mag5883Read();//"mpu6050.h"
    presserf = getAvg(movavg_buff, MOVAVG_SIZE);
    Altitude_baro = getAltitude(presserf,temperature);//Altitude_Ground
    Altitude_barof = Altitude_baro - Altitude_Ground;//+ Altitude_II
} //end roop 20 Hz
if (frameCounter % TASK_10HZ == 0) // 10 Hz task (100 ms)
{
    //presserf = getAvg(movavg_buff, MOVAVG_SIZE);
    //Altitude_baro = getAltitude(presserf,temperature);//Altitude_Ground
    //Altitude_barof = Altitude_baro - Altitude_Ground + Altitude_II;
    Chack_Command();//Control pid
    Altitude_sonaold = Altitude_sonaf;
    Automatictakeland();
} //end roop 10 Hz
if (frameCounter % TASK_5HZ == 0) //GPS_calc TASK_5HZ
{
    Cal_GPS();
    //GPS_distance_m_bearing(GPS_LAT1, GPS_LON1, GPS_LAT_HOME,
    GPS_LON_HOME, Altitude_hat);
    if (Mode == 2 && GPS_FIX == 1) //Position_Hold if (Mode == 2 || Mode == 3 &&
    GPS_FIX == 1)

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี วัตถุประสงค์เพื่อใช้ในการศึกษาวิจัยและการเรียนการสอนเท่านั้น ไม่สามารถนำออกจำหน่ายหรือทำซ้ำโดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆก็ตาม หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูง และต้องอภัยถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    GPS_calc_positionhold();//Control_PID.h
}
else
{
    GPS_I_LAT = 0.0;
    GPS_I_LON = 0.0;
    Control_XBf = 0.0;
    Control_YBf = 0.0;
    error_LAT = 0.0;
    error_LON = 0.0;
    target_LAT = GPS_LAT1f;//GPS_LAT_Hold
    target_LON = GPS_LON1f;//GPS_LON_Hold
}
} //end gps
if (frameCounter % TASK_20HZ == 0) //roop print TASK_5HZ TASK_10HZ
{
    //Serial.print("CH_THR ");Serial.print(CH_THR);Serial.print("\t");
    //Serial.print("CH_AIL ");Serial.print(CH_AIL);Serial.print("\t");
    //Serial.print("CH_ELE ");Serial.print(CH_ELE);Serial.print("\t");
    //Serial.print("CH_RUD ");Serial.print(CH_RUD);Serial.print("\t");
    //Serial.print("AUX_1 ");Serial.print(AUX_1);Serial.print("\t");
    //Serial.print("AUX_2 ");Serial.print(AUX_2);Serial.print("\t");
    //Serial.print("AUX_3 ");Serial.print(AUX_3);Serial.print("\t");
    //Serial.print("AUX_4 ");Serial.print(AUX_4);Serial.print("\t");
    //Serial.print("failsafeCnt ");Serial.print(failsafeCnt);Serial.print("\t");

    //Serial.print("setpoint_rate_roll
");Serial.print(setpoint_rate_roll);Serial.print("\t");
    //Serial.print("setpoint_rate_pitch
");Serial.print(setpoint_rate_pitch);Serial.print("\t");

    //Serial.print("MagXf ");Serial.print(MagXf);Serial.print("\t");
    //Serial.print("MagYf ");Serial.print(MagYf);Serial.print("\t");
    //Serial.print("MagZf ");Serial.print(MagZf);Serial.print("\t");
    //Serial.print("accelRaw[XAXIS] ");Serial.print(accelRaw[XAXIS]);Serial.print("\t");

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และสงวนสิทธิ์ในชื่อการค้าของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกสิ่งนี้ออกไป และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//Serial.print("accelRaw[YAXIS] ");Serial.print(accelRaw[YAXIS]);Serial.print("\t");
//Serial.print("accelRaw[ZAXIS] ");Serial.print(accelRaw[ZAXIS]);Serial.print("\t");
//Serial.print("GyroX*RAD_TO_DEG
");Serial.print(GyroX*RAD_TO_DEG);Serial.print("\t");
//Serial.print("GyroXf*RAD_TO_DEG
");Serial.print(GyroXf*RAD_TO_DEG);Serial.print("\t");
//Serial.print("GyroYf*RAD_TO_DEG
");Serial.print(GyroYf*RAD_TO_DEG);Serial.print("\t");
//Serial.print("GyroZf*RAD_TO_DEG
");Serial.print(GyroZf*RAD_TO_DEG);Serial.print("\t");

//Serial.print("c_acc_x ");Serial.print(c_acc_x);Serial.print("\t");
//Serial.print("c_acc_y ");Serial.print(c_acc_y);Serial.print("\t");
//Serial.print("c_acc_z ");Serial.print(c_acc_z);Serial.print("\t");
//Serial.print("c_magnetom_x ");Serial.print(c_magnetom_x);Serial.print("\t");
//Serial.print("c_magnetom_y ");Serial.print(c_magnetom_y);Serial.print("\t");
//Serial.print("c_magnetom_z ");Serial.print(c_magnetom_z);Serial.print("\t");

//Serial.print("GPS_FIX ");Serial.print(GPS_FIX);Serial.print("\t");
//Serial.print("GPS_LAT1,9 ");Serial.print(GPS_LAT1,9);Serial.print("\t");
//Serial.print("GPS_LAT1lead,9 ");Serial.print(GPS_LAT1lead,9);Serial.print("\t");
//Serial.print("GPS_LAT1f,9 ");Serial.print(GPS_LAT1f,9);Serial.print("\t");

//Serial.print("GPS_LON1,9 ");Serial.print(GPS_LON1,9);Serial.print("\t");
//Serial.print("GPS_LON1f,9 ");Serial.print(GPS_LON1f,9);Serial.print("\t");
//Serial.print("GPS_LON1f2,9 ");Serial.print(GPS_LON1f2,9);Serial.print("\t");
//Serial.print("GPS_LON1lead,9 ");Serial.print(GPS_LON1lead,9);Serial.print("\t");
//Serial.print("error_LAT ");Serial.print(error_LAT);Serial.print("\t");
//Serial.print("error_LON ");Serial.print(error_LON);Serial.print("\t");
//Serial.print("GPS_speed ");Serial.print(GPS_speed);Serial.print("\t");//cm/s
//Serial.print("GPS_ground_course
");Serial.print(GPS_ground_course);Serial.print("\t");//deg

//Serial.print(" velocity_north ");Serial.print( velocity_north);Serial.print("\t");
//Serial.print("actual_speedX ");Serial.print(actual_speedX);Serial.print("\t");
//Serial.print("actual_speedXf ");Serial.print(actual_speedXf);Serial.print("\t");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามนำไฟล์ต้นฉบับสงวนเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//Serial.print("vx_hat ");Serial.print(vx_hat);Serial.print("\t");
//Serial.print("_velocity_east ");Serial.print(_velocity_east);Serial.print("\t");
//Serial.print("actual_speedY ");Serial.print(actual_speedY);Serial.print("\t");
//Serial.print("actual_speedYf ");Serial.print(actual_speedYf);Serial.print("\t");
//Serial.print("vy_hat ");Serial.print(vy_hat);Serial.print("\t");
//Serial.print("GPS_Distance ");Serial3.print(GPS_Distance);Serial3.print("\t");
//Serial.print("GPS_ground_course
");Serial3.print(GPS_ground_course);Serial3.print("\t");

```

```

//Serial.print("temperature ");Serial.print(temperature);Serial.print("\t");
//Serial.print("presser*100 ");Serial.print(presser*100);Serial.print("\t");
//Serial.print("presserf*100 ");Serial.print(presserf*100);Serial.print("\t");
//Serial.print("Altitude_baro ");Serial.print(Altitude_baro);Serial.print("\t");
//Serial.print("Altitude_barof ");Serial.print(Altitude_barof);Serial.print("\t");
//Serial.print("Altitude_sona ");Serial.print(Altitude_sona);Serial.print("\t");
//Serial.print("Altitude_sonaf ");Serial.print(Altitude_sonaf);Serial.print("\t");
//Serial.print("Altitude_Baro_ult
");Serial.print(Altitude_Baro_ult);Serial.print("\t");
//Serial.print("Altitude_hat ");Serial.print(Altitude_hat);Serial.print("\t");
//Serial.print("Altitude_II ");Serial.print(Altitude_II);Serial.print("\t");

//Serial.print("baro_vz*10 ");Serial.print(baro_vz*10);Serial.print("\t");
//Serial.print("vz_sonaf*10 ");Serial.print(vz_sonaf*10);Serial.print("\t");
//Serial.print("vz_hat*10 ");Serial.print(vz_hat*10);Serial.print("\t");
//Serial.print("h_counter ");Serial.print(h_counter);Serial.print("\t");
//Serial.print("GPS_hz ");Serial.print(GPS_hz);Serial.print("\t");

```

```

//Serial.print("vz_hat ");Serial.print(vz_hat);Serial.print("\t");
//Serial.print("DCM10 ");Serial.print(DCM10);Serial.print("\t");
//Serial.print("DCM11 ");Serial.print(DCM11);Serial.print("\t");
//Serial.print("DCM12 ");Serial.print(DCM12);Serial.print("\t");
//Serial.print("AccX ");Serial.print(AccX);Serial.print("\t");
//Serial.print("AccXf ");Serial.print(AccXf);Serial.print("\t");
//Serial.print("AccXf2 ");Serial.print(AccXf2);Serial.print("\t");
//Serial.print("AccY ");Serial.print(AccY);Serial.print("\t");
//Serial.print("AccYf ");Serial.print(AccYf);Serial.print("\t");

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์การเขียนโปรแกรมเพื่อการศึกษาเท่านั้น มิใช่สัญญาให้ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังเป็นให้ตอบโต้เองเท่านั้น และต้องอ้างอิงถึงเจ้าที่ออกเอกสารทุกครั้งที่มีการนำไปใช้

```

//Serial.print("AccZ ");Serial.print(AccZ);Serial.print("\t");
//Serial.print("AccZf2,3 ");Serial.print(AccZf2,3);Serial.print("\t");
//Serial.print("AccZf3,3 ");Serial.print(AccZf3,3);Serial.print("\t");
//Serial.print("AccZf ");Serial.print(AccZf);Serial.print("\t");
//Serial.print("accrX_Earth ");Serial.print(accrX_Earth);Serial.print("\t");
//Serial.print("accrY_Earth ");Serial.print(accrY_Earth);Serial.print("\t");
//Serial.print("accrZ_Earth ");Serial.print(accrZ_Earth);Serial.print("\t");

//Serial.print("GyroX*RAD_TO_DEG
");Serial.print(GyroX*RAD_TO_DEG);Serial.print("\t");
//Serial.print("GyrofY ");Serial.print(GyrofY);Serial.print("\t");
//Serial.print("GyroZ ");Serial.print(GyroZ);Serial.print("\t");
//Serial.print("gyroRaw[XAXIS] ");Serial.print(gyroRaw[XAXIS]);Serial.print("\t");
//Serial.print("gyroRaw[YAXIS] ");Serial.print(gyroRaw[YAXIS]);Serial.print("\t");
//Serial.print("gyroRaw[ZAXIS] ");Serial.print(gyroRaw[ZAXIS]);Serial.print("\t");

//Serial.print("courseRads ");Serial.print(courseRads);Serial.print("\t");
//Serial.print("ahrs_r ");Serial.print(ahrs_r);Serial.print("\t");
//Serial.print("ahrs_p ");Serial.print(ahrs_p);Serial.print("\t");
//Serial.print("ahrs_y ");Serial.print(ahrs_y);Serial.print("\t");
//Serial.print("Heading ");Serial.print(Heading);Serial.print("\t");
//Serial.print("ahrs_y*RAD_TO_DEG
");Serial3.print(ahrs_y*RAD_TO_DEG);Serial3.print("\t");
//Serial.print("cos_rollcos_pitch
");Serial.print(cos_rollcos_pitch);Serial.print("\t");
//Serial.print("x_angle ");Serial.print(x_angle);Serial.print("\t");

//Serial.print("u_roll ");Serial.print(u_roll);Serial.print("\t");
//Serial.print("u_pitch ");Serial.print(u_pitch);Serial.print("\t");
//Serial.print("u_yaw ");Serial.print(u_yaw);Serial.print("\t");

//Serial3.print(Control_XEf);Serial3.print("\t");
//Serial3.print(Control_YEf);Serial3.print("\t");
//Serial3.print(Control_XBf);Serial3.print("\t");
//Serial3.print(Control_YBf);Serial3.print("\t");

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเว็บไซต์ใดๆ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//Serial.print("Control_XEf ");Serial.print(Control_XEf);Serial.print("\t");
//Serial.print("Control_YEf ");Serial.print(Control_YEf);Serial.print("\t");
//Serial.print("Control_XBf ");Serial.print(Control_XBf);Serial.print("\t");
//Serial.print("Control_YBf ");Serial.print(Control_YBf);Serial.print("\t");

//Serial.print("GPSI_LAT ");Serial.print(GPSI_LAT);Serial.print("\t");
//Serial.print("motor_FrontL ");Serial.print(motor_FrontL);Serial.print("\t");
//Serial.print("motor_FrontR ");Serial.print(motor_FrontR);Serial.print("\t");
//Serial.print("motor_BackL ");Serial.print(motor_BackL);Serial.print("\t");
//Serial.print("motor_BackR ");Serial.print(motor_BackR);Serial.print("\t");
//Serial.print("motor_Left ");Serial.print(motor_Left);Serial.print("\t");
//Serial.print("motor_Right ");Serial.print(motor_Right);Serial.print("\t");
//Serial.print("gyroSamples2 ");Serial.print(gyroSamples2);Serial.print("\t");
//Serial.print("Mode ");Serial.print(Mode);Serial.print("\t");
//Serial.print("G_Dt*1000 ");Serial.print(G_Dt*1000);Serial.print("\t");
//Serial.print("millis()/1000.0 ");Serial.print(millis()/1000.0);//millis() micros()
Serial.print("\n");
} //end loop 5 Hz
if (frameCounter >= TASK_1HZ) { // Reset frameCounter back to 0 after reaching
100 (1s)
    frameCounter = 0;
    time_auto++;
    temperature = baro.getTemperature(MS561101BA_OSB_4096);
    Remote_TrimACC();//motor.h
    if(Status_LED == LOW)
    {
        Status_LED = HIGH;
    }
    else
    {
        Status_LED = LOW;
    }
    digitalWrite(13, Status_LED);//A
} //end loop 1 Hz
} //end loop 100 HZ
}

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามแก้ไขตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//////////////////////////////////PID Control//////////////////////////////////
```

```
// ROLL CONTROL P-PID////////////////////////////////
```

```
float setpoint_roll = ((CH_AILf-CH_AIL_Cal)*0.085) + Control_YBf;/0.12 max +-45 deg
////+18 + Control_YBf
```

```
applyDeadband(setpoint_roll, 1.5);/1.2
```

```
setpoint_rate_roll = (0.065*setpoint_rate_roll/(0.065+G_Dt)) + ((setpoint_roll-
setpoint_rollold)/(0.065+G_Dt));/Diff remote
```

```
setpoint_rollold = setpoint_roll;
```

```
setpoint_rate_roll = constrain(setpoint_rate_roll, -80, 80);/+-80 deg/s
```

```
float error_roll = setpoint_roll - ahrs_r;/ahrs_r*ToDeg
```

```
float error_rate_roll = setpoint_rate_roll + error_roll*Kp_levelRoll -
GyroXf*RAD_TO_DEG;
```

```
roll_I_rate += error_rate_roll*Ki_rateRoll*G_Dt;
```

```
roll_I_rate = constrain(roll_I_rate, -50, 50);/+-150
```

```
roll_D_rate = (tar*roll_D_rate/(tar+G_Dt)) + ((error_rate_roll-
error_rate_rollold)/(tar+G_Dt));
```

```
error_rate_rollold = error_rate_roll;
```

```
u_roll = Kp_rateRoll*error_rate_roll + roll_I_rate + Kd_rateRoll*roll_D_rate;
```

```
u_roll = constrain(u_roll, -200, 200);/+-300 120
```

```
///// PITCH CONTROL P-PID////////////////////////////////
```

```
float setpoint_pitch = ((CH_ELEf-CH_ELE_Cal)*0.085) + Control_XBf;/max +-45 deg
////+18 - Control_XBf
```

```
applyDeadband(setpoint_pitch, 1.5);/1.2
```

```
setpoint_rate_pitch = (0.065*setpoint_rate_pitch/(0.065+G_Dt)) + ((setpoint_pitch-
setpoint_pitchold)/(0.065+G_Dt));/Diff remote
```

```
setpoint_pitchold = setpoint_pitch;
```

```
setpoint_rate_pitch = constrain(setpoint_rate_pitch, -80, 80);/+-80
```

```
float error_pitch = setpoint_pitch - ahrs_p;/ahrs_p*RAD_TO_DEG
```

```
float error_rate_pitch = setpoint_rate_pitch + error_pitch*Kp_levelPitch -
```

```
GyroYf*RAD_TO_DEG;
```

```

pitch_I_rate += error_rate_pitch*Ki_ratePitch*G_Dt;
pitch_I_rate = constrain(pitch_I_rate, -50, 50);//+-150
pitch_D_rate = (tar*pitch_D_rate/(tar+G_Dt)) + ((error_rate_pitch-
error_rate_pitchold)/(tar+G_Dt));
error_rate_pitchold = error_rate_pitch;
u_pitch = Kp_ratePitch*error_rate_pitch + pitch_I_rate + Kd_ratePitch*pitch_D_rate;
u_pitch = constrain(u_pitch, -200, 200);//+ 300 120

///// YAW CONTROL PID//////////

float setpoint_rate_yaw = (CH_RUDfCH_RUD_Cal)*0.55;//0.4 0.35
applyDeadband(setpoint_rate_yaw, 7.1);//6.5
if(abs(setpoint_rate_yaw)> 0.1){
  setHeading = ahrs_y;// 0 degree ,ahrs_tin.h
}
float error_yaw = 0.0 - Heading;
float error_rate_yaw = setpoint_rate_yaw + error_yaw*Kp_levelyaw-
GyroZf*RAD_TO_DEG;
yaw_I_rate += error_rate_yaw*Ki_rateYaw*G_Dt;
yaw_I_rate = constrain(yaw_I_rate, -50, 50);//+- 100
yaw_D_rate = (tar*yaw_D_rate/(tar+G_Dt)) + ((error_rate_yaw-
error_rate_yawold)/(tar+G_Dt));
error_rate_yawold = error_rate_yaw;
u_yaw = Kp_rateYaw*error_rate_yaw + yaw_I_rate + Kd_rateYaw*yaw_D_rate;
u_yaw = constrain(u_yaw, -120, 120);//+ 150

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

////////// คำวณ หาดำแหน่งเป้าหมาย //////////
void GPS_distance_m_bearing(float lat1, float lon1, float lat2, float lon2, float alt){
float a, R, c, d, dLat, dLon;
lon1=lon1/RAD_TO_DEG;
lat1=lat1/RAD_TO_DEG;
lon2=lon2/RAD_TO_DEG;
lat2=lat2/RAD_TO_DEG;
R=6371000.0; //m raio da terra 6371km
a=atan2(sin(lon2-lon1)*cos(lat2), cos(lat1)*sin(lat2)-sin(lat1)*cos(lat2)*cos(lon2-lon1));
GPS_ground_course2 = a*RAD_TO_DEG;
//if (yaw<0) yaw=360+yaw;
//calculo da distancia entre modelo e home
dLat = (lat2-lat1);
dLon = (lon2-lon1);
a = sin(dLat/2) * sin(dLat/2) + sin(dLon/2) * sin(dLon/2) * cos(lat1) * cos(lat2);
c = 2* asin(sqrt(a));
d = R * c;
//alt=alt-Alt Home;
//pitch=atan(alt/d);
//pitch=pitch*360/(2*PI);
GPS_Distance = sqrt(alt*alt+d*d);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้