



รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบช่วยลงจอดแบบแม่นยำสำหรับโดรนชาร์จอัตโนมัติ

PRECISION LANDING SYSTEM FOR AUTO CHARGING DRONE



พิชญ์ หุตะเจริญ

ภาควิชาวิศวกรรมการวัดและควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561



## รายงานสหกิจศึกษาฉบับสมบูรณ์

ระบบช่วยลงจอดแบบแม่นยำสำหรับโดรนชาร์จอัตโนมัติ

PRECISION LANDING SYSTEM FOR AUTO CHARGING DRONE

พิชญ์ หุตะเจริญ

ภาควิชาวิศวกรรมการวัดและควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโครงการ	ระบบช่วยลงจอดแบบแม่นยำสำหรับโดรนชาร์จอัตโนมัติ
นักศึกษา	นายพิชญ์ หุตะเจริญ
ภาควิชา	วิศวกรรมการวัดและควบคุม
อาจารย์นิเทศ	ผู้ช่วยศาสตราจารย์ ดร.นพดล มณีรัตน์
ผู้นิเทศงาน	นายอภิวัฒน์ เรืองประดิษฐ์
สถานประกอบการ	บริษัท เอสซีจี แพคเกจจิ้ง จำกัด (มหาชน)

### บทคัดย่อ

โครงการนี้เป็นส่วนหนึ่งของการวิจัย เพื่อพัฒนาระบบอากาศยานไร้คนขับ ซึ่งสามารถปฏิบัติการกิจอย่างเป็นกิจวัตรได้ โดยอัตโนมัติทุกขั้นตอนและไม่จำเป็นต้องมีผู้ดูแลขณะปฏิบัติการกิจ โครงการนี้นำเสนอการวิจัย, ทดลอง และพัฒนา ระบบช่วยเพิ่มความแม่นยำในการลงจอดของอากาศยานไร้คนขับขนาดเล็กรุ่น DJI Matrice 100 โดยการติดตั้งระบบเสริมบนตัวอากาศยานทดแทนการใช้ GPS ประกอบด้วยคอมพิวเตอร์และกล้องขนาดเล็ก, อุปกรณ์วัดความสูงจากพื้น และอุปกรณ์รักษา ระดับ เพื่อให้อากาศยานสามารถลงจอดบนพื้นที่ ที่มีขนาดจำกัดได้ และนำไปใช้งานร่วมกับระบบชาร์จแบตเตอรี่อัตโนมัติต่อไป ผลการทดลองแสดงให้เห็นว่า ระบบเสริมนี้ ช่วยให้อากาศยานลงจอดได้แม่นยำมากขึ้น สามารถนำไปใช้ร่วมกับระบบชาร์จอัตโนมัติได้ แต่ยังคงจำเป็นต้องพัฒนาเพิ่มเติมต่อไป

คำสำคัญ : Precision Landing, LIDAR, Gimbal, Image Processing, ระบบควบคุมวงปิด

**Project Title:** Precision Landing System for Auto Charging Drone

**Student:** Mr. Pitt Hutacharearn

**Department:** Instrumentation and Control

**Advisor:** Assistant Professor Dr. Noppadol Maneerat

**Mentor:** Mr. Natthaphong Wongwattanakij

**Company:** SCG Packaging Public Company Limited

## ABSTRACT

This project is a part of unmanned aerial vehicle development for fully-autonomous operation which does not require real-time monitoring personnel. This project presents the research, experimentation and development of land-coordinate precision enhancing system for DJI brand Matrice 100 quadcopter using additional onboard computer, camera, range finder and gimbal in order to allow the aircraft to land on areas with the limited position error and integrate into an auto-charging drone system. The result proved that this system helped the drone to land very precisely and capable of use for auto-charging drone system but more improvements still have to be made.

**keywords :** Precision Landing, LIDAR, Gimbal, Image Processing, Closed-loop Control

## กติกกรมประกาศ

ในการจัดทำโครงการในครั้งนี้ สามารถทำให้สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากได้รับความช่วยเหลือจากผู้ช่วยศาสตราจารย์ ดร.นพดล มณีรัตน์ ที่มอบโอกาสในการเข้าร่วมโครงการสหกิจศึกษา คอยให้คำปรึกษา การสนับสนุน และความช่วยเหลือเป็นอย่างดี ตลอดจนการตรวจสอบความถูกต้อง ทำให้โครงการนี้เสร็จสมบูรณ์ คณะผู้จัดทำรู้สึกซาบซึ้งและขอกราบขอบพระคุณอย่างสูง

ขอขอบคุณทางบริษัท เอสซีซี แพคเกจจิ้ง จำกัด (มหาชน) รวมทั้งพี่ๆ วิศวกรแผนก Design Engineering และแผนกอื่นๆ ที่คอยช่วยเหลือสนับสนุนต่างๆ พร้อมให้คำปรึกษาและให้ข้อมูลในการแก้ไขปัญหาต่างๆ ที่เกิดขึ้นตลอดช่วงเวลาโครงการสหกิจศึกษาให้สำเร็จลุล่วงด้วยดีได้

สุดท้ายนี้ขอกราบขอบพระคุณบิดา มารดา และขอบคุณเพื่อนร่วมงานทุกท่าน ที่คอยเป็นกำลังใจตลอดมาจนจบโครงการไว้ ณ ที่นี้ หากมีข้อผิดพลาดประการใดให้ถือเป็นความบกพร่องของทางคณะผู้จัดทำ และขออภัยมา ณ ที่นี้ด้วย

พิชญ์ หุตะเจริญ

# สารบัญ

หน้า

บทคัดย่อ .....	I
บทคัดย่อภาษาอังกฤษ.....	II
กติกกรมประกาศ .....	III
สารบัญ .....	IV
สารบัญรูป.....	VI
สารบัญตาราง .....	VIII
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาและความสำคัญ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	2
1.4 ขั้นตอนการศึกษา.....	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....	3
2.1 ทฤษฎีเกี่ยวกับ Quadcopter.....	3
2.2 ทฤษฎีการสื่อสารแบบอนุกรม [2].....	8
2.3 ทฤษฎีระบบควบคุมวงปิด.....	12
2.4 ทฤษฎีการทำงานของ LIDAR.....	13

## สารบัญ (ต่อ)

	หน้า
บทที่ 3 วิธีการดำเนินงาน.....	14
3.1 การวิเคราะห์ปัญหา.....	14
3.2 การกำหนดวิธีการแก้ไข .....	14
3.3 การดำเนินการแก้ไขปัญหา.....	22
บทที่ 4 ผลการดำเนินงาน .....	32
บทที่ 5 สรุปผลและข้อเสนอแนะ.....	34
5.1 สรุปผล.....	34
5.2 ปัญหาและวิธีการแก้ไขปัญหา.....	34
5.3 ข้อเสนอแนะ.....	35
เอกสารอ้างอิง.....	36
ประวัติผู้เขียน.....	37

# สารบัญรูป

รูปที่	หน้า
2.1 ตัวอย่าง Quadcopter ยี่ห้อ DJI รุ่น Matrice 100 .....	4
2.2 การ Pitch หรือ Roll ของ Quadcopter .....	5
2.3 ทิศทางการหมุนของใบพัด .....	5
2.4 การตั้งโปรแกรมเส้นทางการบินล่วงหน้า .....	7
2.5 การบินตามวัตถุโดยอัตโนมัติ โดยใช้ปัญญาประดิษฐ์ .....	7
2.6 การเชื่อมต่อระบบ UART .....	9
2.7 8-bit UART Data Format .....	9
2.8 Serial Peripheral Interface .....	10
2.9 การเชื่อมต่อระบบ I <sup>2</sup> C .....	11
2.10 ระบบควบคุมป้อนกลับแบบลบ .....	12
2.11 การตรวจจับแสงที่สะท้อนกลับของ LIDAR .....	13
3.1 การเชื่อมต่อ Onboard Computer กับ Flight Controller .....	15
3.2 STM32F429 Discovery .....	16
3.3 Raspberry Pi 3 .....	17
3.4 ระบบ RTK [4] .....	17
3.5 กล้อง Pixy .....	18
3.6 ข้อมูลที่ Pixy ประมวลผลแล้ว .....	19
3.7 Tarot T2-2D Gimbal .....	20
3.8 ความคลาดเคลื่อนของตำแหน่งที่เกิดจากมุมเอียง .....	20
3.9 ความคลาดเคลื่อนของตำแหน่งที่เกิดจากความสูง .....	21
3.10 Garmin LIDAR Lite V3 .....	21
3.11 การพัฒนาโครงสำหรับติดตั้ง Sensor โดยเครื่องพิมพ์ 3 มิติ .....	22

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.12 โครงสำหรับติดตั้ง Sensor ที่ติดตั้งต็มถ่วงน้ำหนักได้.....	22
3.13 Pixy และ LIDAR ที่ติดตั้งบน Gimbal พร้อมน้ำหนักถ่วง (ขวา).....	23
3.14 แผนผัง Onboard System .....	24
3.15 Flowchart ของ pixy_node .....	26
3.16 Flowchart ของ x_node.....	27
3.17 Flowchart ของ bms_node.....	28
3.18 วงจรควบคุมการชาร์จ .....	29
3.19 วงจรทั้งหมดของ Onboard System.....	30
3.20 แทนลงจอดสำหรับทดลอง .....	31
4.1 การทดลองลงจอดแบบแม่นยำ.....	32
4.2 ตำแหน่งสัมผัสของขั้วในการลงจอดแต่ละครั้ง .....	33

# สารบัญตาราง

ตารางที่	หน้า
2.1 ทิศทางการควบคุม Quadrotor แบบ Manual ด้วยคั่นบังคับ Mode 2 .....	6
3.1 ค่าซีตจำกัดของ DJI Matrice 100 .....	15



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญ

ความเจริญก้าวหน้าของเทคโนโลยีอากาศยานไร้คนขับหรือ “โดรน” ทำให้ราคาถูกลงมาก และสามารถใช้งานได้ง่าย จึงเกิดการนำเทคโนโลยีดังกล่าวมาใช้อย่างแพร่หลาย รวมไปถึงการนำไปใช้งานในอุตสาหกรรมต่างๆ เพื่อทดแทนการใช้แรงงานมนุษย์, ลดต้นทุนค่าแรงงาน และลดความสูญเสียจากการเกิดอุบัติเหตุในการทำงานบนที่สูง

บริษัท เอสซีจี แพคเกจจิ้ง จำกัด (มหาชน) มีความสนใจที่จะนำเทคโนโลยีอากาศยานไร้คนขับ มาใช้ในการตรวจวัดปริมาณของวัตถุติดที่คงเหลืออยู่ ซึ่งปกติใช้แรงงานมนุษย์ในการนับจำนวนและประมาณปริมาณวัตถุติด แต่วิธีดังกล่าวมีโอกาสเกิดความคลาดเคลื่อนได้มาก และสิ้นเปลืองต้นทุนแรงงาน ในขณะที่เดียวกันการใช้อากาศยานไร้คนขับแบบปัจจุบัน จำเป็นต้องมีผู้บังคับคอยควบคุมอากาศยาน และนำกลับเพื่อชาร์จแบตเตอรี่ใหม่ทุกครั้ง ซึ่งยังคงสิ้นเปลืองเวลาและต้นทุนค่าแรงงานอยู่

บริษัทฯ เล็งเห็นว่า งานดังกล่าวเป็นงานที่ซ้ำซาก ไม่เหมาะสมกับการใช้แรงงานมนุษย์ จึงเริ่มทำการวิจัยและพัฒนาาระบบอากาศยานไร้คนขับ ที่สามารถปฏิบัติการกิจทั้งหมดได้โดยอัตโนมัติ ซึ่งรวมถึงการเริ่มต้นทำงานโดยอัตโนมัติเป็นกิตวอร์ และขั้นตอนการชาร์จแบตเตอรี่ แต่เทคโนโลยีในตลาดปัจจุบันยังไม่สามารถทำได้ จึงต้องมีการวิจัยและพัฒนาขึ้นเอง

### 1.2 วัตถุประสงค์ของโครงการ

1. ศึกษาองค์ประกอบที่สำคัญของการสร้างระบบอากาศยานไร้คนขับอัตโนมัติ
2. ศึกษาข้อจำกัดของระบบอากาศยานไร้คนขับอัตโนมัติ
3. ลดต้นทุนแรงงานสำหรับงานบางประเภท โดยการทดแทนด้วยระบบอัตโนมัติ

### 1.3 ขอบเขตของปริญญาโท

1. ระบบต้นแบบ พัฒนาระบบโดรน รุ่น DJI Matrice 100
2. เป็นการวิจัยที่ส่วนระบบช่วยในการลงจอด และบางส่วนของระบบชาร์จอัตโนมัติ
3. อากาศยานสามารถลงจอดและชาร์จแบตเตอรี่จนเต็มได้โดยอัตโนมัติ
4. ต้นทุนของระบบจะต่อน้อยกว่าระบบ RTK สำเร็จรูป

### 1.4 ขั้นตอนการศึกษา

1. ศึกษาระบบการทำงาน และข้อจำกัดของโดรน รุ่น DJI Matrice 100
2. วิเคราะห์ปัญหาและความต้องการเพื่อเลือกวิธีการที่จะนำมาใช้ได้อย่างเหมาะสม
3. ศึกษาวิธีการเขียนโปรแกรมในภาษา C++
4. เลือกซื้ออุปกรณ์
5. ทดลองบนระบบต้นแบบ, การจำลองเหตุการณ์ และการจำลองด้วย Software
6. ติดตั้งระบบจริงบนอากาศยาน
7. ทดสอบระบบ แก้ไขปัญหาที่เกิดขึ้น และปรับแต่งการทำงานให้เสถียรมากที่สุด
8. ทดสอบระบบที่สมบูรณ์
9. นำเสนอผลการดำเนินงาน

### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับประสบการณ์การทำงานจริง
2. ได้เข้าใจความสนใจและความสามารถของตนเองมากขึ้น
3. ได้เรียนรู้การทำงานของอากาศยานไร้คนขับอย่างลึกซึ้ง
4. ช่วยลดต้นทุนแรงงานให้กับบริษัท
5. เป็นพื้นฐานสำหรับการพัฒนาต่อยอดระบบอากาศยานไร้คนขับอัตโนมัติ

## บทที่ 2

# แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

### 2.1 ทฤษฎีเกี่ยวกับ Quadcopter

อากาศยานไร้คนขับขึ้นลงแนวดิ่งแบบหลายเครื่องยนต์ นิยมเรียกว่า Multirotor หรือโดรน สามารถจำแนกได้ตามจำนวนและการจัดเรียงของใบพัด เช่น

1. 3 ใบพัด 3 แฉก เรียกว่า Tricopter Y3
2. 4 ใบพัด 4 แฉก เรียกว่า Quadcopter X4 หรือ Quadrotor X4
3. 6 ใบพัด 6 แฉก เรียกว่า Hexacopter V6
4. 6 ใบพัด 3 แฉก เรียกว่า Coaxial Hexacopter Y6
5. 8 ใบพัด 8 แฉก เรียกว่า Octocopter V8
6. 8 ใบพัด 4 แฉก เรียกว่า Coaxial Octocopter X8

#### 2.1.1 ส่วนประกอบของ Quadcopter

โครงสร้างพื้นฐานของ Quadcopter ประกอบด้วย

1. ใบพัด 4 ชุด
2. มอเตอร์ 4 ตัวในการขับเคลื่อนใบพัดทั้ง 4 ชุดเพื่อสร้างแรงยกทำให้ใบพัดแต่ละชุดทำงานอิสระจากกัน
3. วงจรควบคุมมอเตอร์ เรียกว่า Electronic Speed Controller (ESC) 4 ชุด ใช้ในการขับเคลื่อนมอเตอร์ทั้ง 4 ชุด
4. ตัวโครงมักทำจากวัสดุน้ำหนักเบารูปทรง 4 แฉก ซึ่งใบพัดจะถูกติดตั้งที่ปลายแต่ละแฉก
5. วงจรควบคุมการบิน เรียกว่า Flight Controller (FC) ทำหน้าที่รักษาสถิตของอากาศยานขณะบินและประมวลผลคำสั่งจากผู้ใช้เพื่อควบคุมมอเตอร์ทั้ง 4 ชุด ผ่าน ESC

6. Sensor ใช้ในการรับรู้สถานะและตำแหน่งของอากาศยาน ประกอบไปด้วย
  1. Gyroscope วัดความเร็วเชิงมุม 3 แกน
  2. Accelerometer วัดความเร่งเชิงมุม 3 แกน
  3. Barometer วัดความสูงจากด้วยความกดอากาศ
  4. Compass ตรวจสอบทิศทางเทียบกับสนามแม่เหล็กโลก
  5. Global Positioning System (GPS) ตรวจสอบตำแหน่งบนพื้นผิวโลก
7. ระบบรับ-ส่งสัญญาณวิทยุจากผู้ควบคุมระยะไกล
8. แบตเตอรี่ เป็นแหล่งพลังงานของอากาศยาน

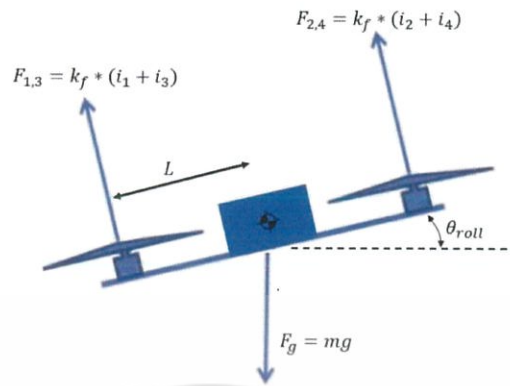


รูปที่ 2.1 ตัวอย่าง Quadcopter ยี่ห้อ DJI รุ่น Matrice 100

### 2.1.2 การเคลื่อนไหวของ Quadcopter [1]

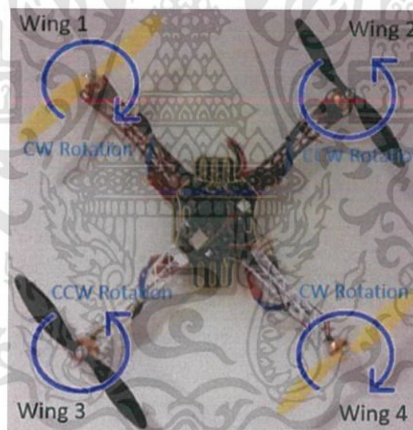
Quadcopter สามารถบินไปในทิศทางต่างๆ ได้โดยการปรับความเร็วของใบพัดทั้ง 4 ชุด การเคลื่อนไหวของอากาศยาน ถูกนิยามไว้เป็น 4 ทิศทาง ได้แก่

1. Pitch : การก้ม-เงยของอากาศยาน เกิดการเคลื่อนที่เดินหน้า-ถอยหลัง กระทำโดยการสร้างความต่างของแรงยกระหว่างใบพัดหน้าและหลัง
2. Roll : การเอียงตัวของอากาศยาน เกิดการเคลื่อนที่ซ้าย-ขวา กระทำโดยการสร้างความต่างของแรงยกระหว่างใบพัดซ้ายและขวา ดังรูปที่ 2.2



รูปที่ 2.2 การ Pitch หรือ Roll ของ Quadcopter

3. Yaw : การหมุนตัวของอากาศยานในแกน Z กระทำโดยการสร้างความต่างของความเร็วระหว่างใบพัด ชุดหมุนตามเข็มนาฬิกา และใบพัดชุดหมุนทวนเข็มนาฬิกา ดังรูปที่ 2.3



รูปที่ 2.3 ทิศทางการหมุนของใบพัด

4. Throttle : การเพิ่ม-ลดระดับความสูงกระทำโดยการเปลี่ยนแปลงแรงยกโดยรวมของใบพัดทั้งหมด


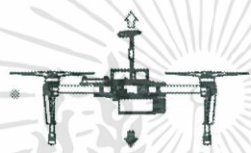





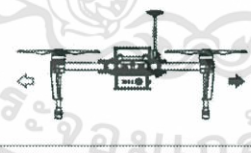
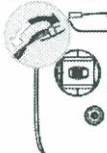
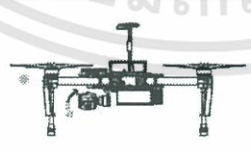
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.3 การควบคุม Quadcopter

การควบคุมการบินของ Quadcopter สามารถกระทำได้ 3 รูปแบบ ได้แก่

1. การควบคุมผ่านวิทยุ (Manual Control) ใช้การมองเห็นของผู้บังคับในการควบคุมตำแหน่งของอากาศยาน การควบคุมที่นิยมมากที่สุดคือ การควบคุมแบบ Mode 2 ดังตารางที่ 2.1

ตารางที่ 2.1 ทิศทางการควบคุม Quadrotor แบบ Manual ด้วยคันบังคับ Mode 2

Remote Controller (Mode 2)	Aircraft (* Indicates nose direction)	Function
		Moving the Left Stick up/down changes the aircraft's elevation. Push it up to ascend and down to descend. Use this stick to take off when the motors are spinning at idle speed. The aircraft will hover in place if the Left Stick is released.
		Moving the Left Stick left/right changes the heading of the aircraft. Push it left to rotate the aircraft counter clock-wise, and right to rotate the aircraft clockwise.
		Moving the Right Stick up/down changes the aircraft's forward and backward pitch. Push it up to fly forwards and down to fly backwards. Push the Right Stick further for a larger pitch angle (max 35°) and faster flight.
		Moving the Right Stick left/right changes the aircraft's left and right pitch. Push it left to fly left and right to fly right. Push the Right Stick further for a larger pitch angle (max 35°) and faster flight.
		Turn the Gimbal Dial to the right to point the camera upwards, and to the left to point it downwards.

2. การตั้งโปรแกรมเส้นทางการบินล่วงหน้า (Waypoint) เพื่อให้อากาศยานเดินทางไปยังตำแหน่งต่างๆ โดยอัตโนมัติ โดยอาศัยการระบุตำแหน่งด้วย GPS ดังรูปที่ 2.4



รูปที่ 2.4 การตั้งโปรแกรมเส้นทางการบินล่วงหน้า

3. การบินแบบอัตโนมัติโดยใช้ปัญญาประดิษฐ์ (AI) เส้นทางการบินถูกกำหนดขึ้นเอง โดยคอมพิวเตอร์ ดังรูปที่ 2.5



รูปที่ 2.5 การบินตามวัตถุโดยอัตโนมัติ โดยใช้ปัญญาประดิษฐ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 ทฤษฎีการสื่อสารแบบอนุกรม [2]

การสื่อสารแบบอนุกรม (Serial Communication) คือ การใช้สายสัญญาณ 1 เส้นในการรับส่งข้อมูลแบบต่อเนื่อง ทั้งนี้การสื่อสารแบบอนุกรมสามารถแบ่งได้เป็น 2 รูปแบบคือ

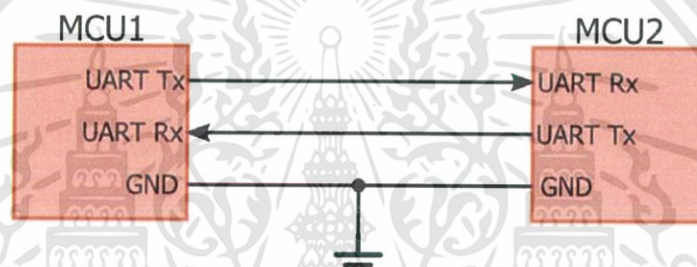
1. แบบอะซิงโครนัส (Asynchronous) เป็นการสื่อสารที่ใช้สายสัญญาณข้อมูลเพียงเส้นเดียวในการทำงาน โดยอาศัยสัญญาณจากส่วนต้นของข้อมูล และความเข้าใจกันระหว่างผู้รับและผู้ส่งข้อมูล การสื่อสารแบบนี้จำเป็นต้องอาศัยเวลาเป็นตัวกำหนดจังหวะการรับสัญญาณ ซึ่งหากมีการตั้งค่าที่ไม่ตรงกันระหว่างผู้รับและผู้ส่ง จะทำให้อ่านข้อมูลที่ส่งมาได้ผิดพลาด ข้อดีของการสื่อสารแบบ Asynchronous คือ ใช้สายสัญญาณเพียงเส้นเดียวในการรับ-ส่งข้อมูล แต่ข้อเสียคือ เกิดความผิดพลาดของข้อมูลได้ง่าย โดยโปรโตคอลที่ทำงานแบบ Asynchronous คือ Universal Asynchronous Receiver Transmitter (UART)

2. แบบซิงโครนัส (Synchronous) เป็นการสื่อสารที่ใช้สายสัญญาณข้อมูลอย่างน้อย 1 เส้น และมีสายอีก 1 เส้นกำหนดจังหวะการรับข้อมูล นิยมเรียกว่า Clock ข้อดีของการสื่อสารแบบ Synchronous คือ การรับส่งข้อมูลมีความผิดพลาดน้อยหรือไม่มีความผิดพลาดเลย แต่ข้อเสียคือต้องใช้สายสัญญาณเพิ่มขึ้น 1 เส้นในการสื่อสาร โดยโปรโตคอลที่ทำงานแบบ Synchronous ได้แก่ I-squared-C (I<sup>2</sup>C) และ Serial Peripheral Interface (SPI)

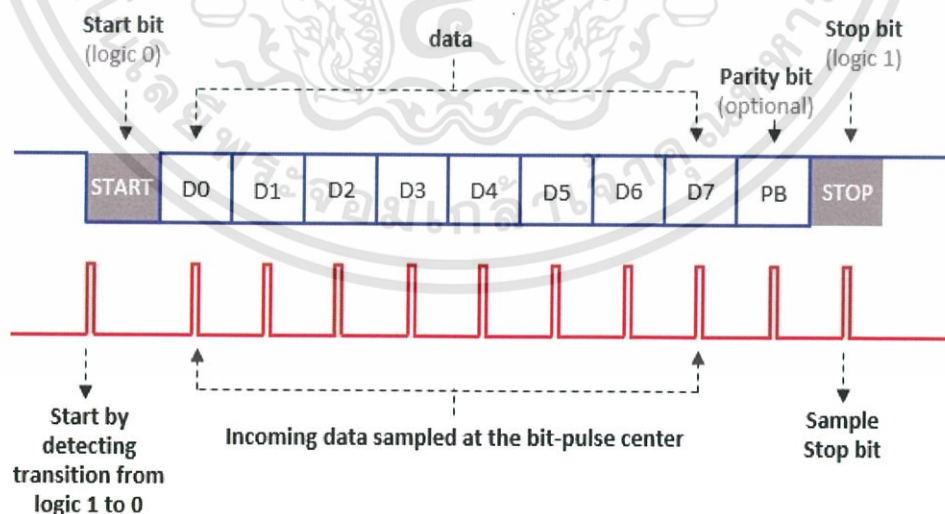
การสื่อสารแบบอนุกรมจะช้ากว่าการสื่อสารแบบขนานมาก แต่มีข้อดีคือ บางโปรโตคอลสามารถต่อพ่วงอุปกรณ์ได้หลายตัวในขณะที่ใช้จำนวนสายสัญญาณเท่าเดิม หรือใช้สายสัญญาณเพิ่มไม่มาก

### 2.2.1 Universal Asynchronous Receiver Transmitter (UART)

UART ใช้การสื่อสารแบบอนุกรมโดยเป็นแบบ Asynchronous ใช้สายสัญญาณเพียง 2 เส้น คือ Tx และ Rx ในการรับ-ส่งข้อมูล ดังรูปที่ 2.6 โดยเส้น Tx ใช้ส่งสัญญาณออกจากอุปกรณ์ และเส้น Rx ใช้รับสัญญาณเข้ามาในอุปกรณ์ เมื่อนำไปต่อใช้งานจะต้องต่อไขว้กันระหว่างอุปกรณ์ กล่าวคือ เส้น Tx ของอุปกรณ์ชิ้นหนึ่งจะต้องต่อเข้าไปยังเส้น Rx ของอุปกรณ์อีกชิ้นหนึ่ง และต้องกำหนดความเร็วและรูปแบบในการสื่อสาร ดังรูปที่ 2.7 ให้ตรงกันบนอุปกรณ์ทั้งสอง จึงจะสามารถรับ-ส่งข้อมูลได้ถูกต้อง ทั้งนี้ความเร็วสูงสุดที่รองรับได้มักขึ้นอยู่กับอุปกรณ์ที่จะเชื่อมต่อด้วย อุปกรณ์ที่ใช้ UART ในการสื่อสารที่พบได้ทั่วไป เช่น Bluetooth, Microcontroller



รูปที่ 2.6 การเชื่อมต่อระบบ UART



รูปที่ 2.7 8-bit UART Data Format

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

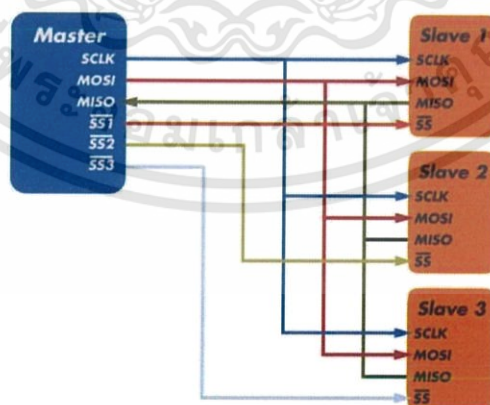
### 2.2.2 Serial Peripheral Interface (SPI) [3]

SPI ดังรูปที่ 2.8 เป็นวิธีการสื่อสารแบบ Synchronous ที่ใช้ในการติดต่อสื่อสารกับอุปกรณ์ SPI ทำงานในรูปแบบที่กำหนดให้อุปกรณ์ตัวหนึ่งทำหน้าที่เป็นผู้นำ (Master) ในขณะที่อีกตัวหนึ่งทำหน้าที่เป็นผู้ตาม (Slave) การสื่อสารสามารถรับและส่งสัญญาณได้พร้อมกัน หรือเรียกว่า Full-duplex ในการสื่อสารแบบ SPI ไม่มีมาตรฐานกำหนดรูปแบบการจัดเรียงข้อมูลตายตัว บางครั้งผู้ใช้จำเป็นต้องตั้งค่าอุปกรณ์ให้มีรูปแบบการจัดเรียงข้อมูลที่ตรงกัน SPI ใช้สายสัญญาณอย่างน้อย 3 เส้น ได้แก่

1. Serial Clock (SCLK) ทำหน้าที่กำหนดจังหวะในการส่งสัญญาณ
2. Master Output, Slave Input (MOSI) ส่งสัญญาณจาก Master ไปยัง Slave
3. Master Input, Slave Output (MISO) ส่งสัญญาณจาก Slave ไปยัง Master

ระบบอาจมีการเพิ่มสาย Slave Select (SS) ไปยังอุปกรณ์ต่อพ่วงเพิ่มเติมแต่ละตัว โดยอุปกรณ์ที่เพิ่มเข้ามาจะต้องทำหน้าที่เป็น Slave เท่านั้น เพื่อให้ Master สามารถรับส่งข้อมูลไปยังอุปกรณ์หลายชิ้นได้ จำนวนสาย Slave Select จะเท่ากับจำนวน Slave ในระบบ ยกเว้นกรณีที่มี Slave เพียงตัวเดียว ซึ่งไม่จำเป็นต้องใช้ Slave Select โดย Master จะส่งสัญญาณ Slave Select เป็น LOW เมื่อต้องการให้อุปกรณ์ Slave ชิ้นนั้นทำงาน

อุปกรณ์ที่ทำหน้าที่ Master เป็นผู้ควบคุมสัญญาณ Serial Clock ซึ่งกำหนดความเร็วของการสื่อสาร



รูปที่ 2.8 Serial Peripheral Interface

### 2.2.3 I<sup>2</sup>C

I<sup>2</sup>C เป็นโปรโตคอลที่ใช้การสื่อสารแบบ Synchronous คิดค้นโดย NXP โดยอาจมีการใช้ชื่ออื่นในการเรียก เช่น 2-Wire สำหรับในประเทศไทยมักใช้คำว่า “ไอ-สแควร์-ซี” (I-squared-C) หรือ “ไอ-ทู-ซี” (I-two-C) ใช้สายจำนวน 2 เส้นในการสื่อสาร ดังรูปที่ 2.9 คือ

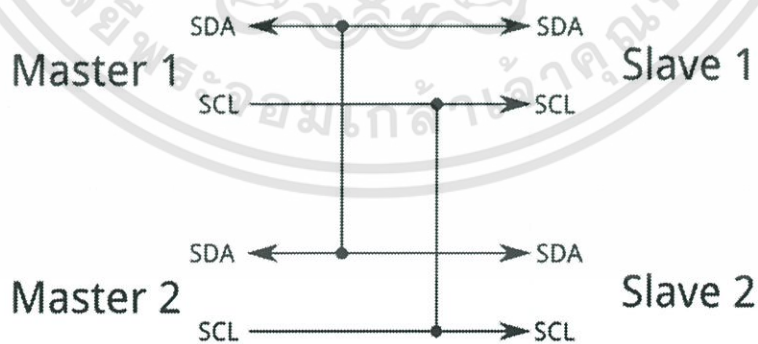
1. Serial Data (SDA) ใช้ในการรับส่งข้อมูล
2. Serial Clock (SCL) ใช้ในการควบคุมการรับส่งข้อมูลของอุปกรณ์ปลายทาง

I<sup>2</sup>C สามารถต่อเพิ่มอุปกรณ์ได้โดยไม่ต้องเพิ่มสายสัญญาณ

ปกติการเชื่อมต่อ I<sup>2</sup>C จะมีตัวต้านทานค่าประมาณ 4.7k $\Omega$  – 10k $\Omega$  ต่อแบบ Pull-up ที่ขาสัญญาณทั้ง 2 และส่งสัญญาณโดยการ Open-drain ทำให้ I<sup>2</sup>C สามารถใช้สื่อสารระหว่างอุปกรณ์ที่ใช้แรงดันไฟฟ้าแตกต่างกันได้

โปรโตคอลแบบ I<sup>2</sup>C รองรับการเชื่อมต่ออุปกรณ์มากกว่า 1 ตัว โดยแต่ละตัวจะมีรหัสประจำตัวที่เรียกว่า Address เป็นตัวแบ่งแยก อุปกรณ์หนึ่งสามารถกำหนดเป้าหมายในการสื่อสารได้ โดยการระบุรหัสประจำตัวก่อนการรับ-ส่งข้อมูล

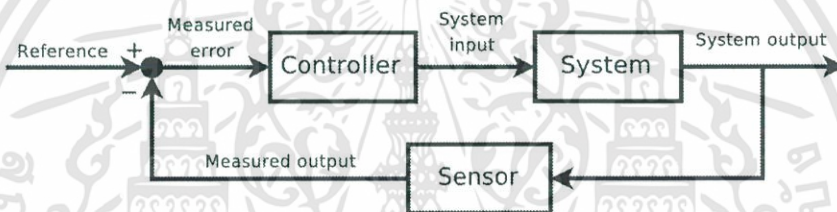
โปรโตคอลแบบ I<sup>2</sup>C ทำงานลักษณะของการถามตอบ โดยหากจะมีการสื่อสาร ตัวผู้รับข้อมูลจะต้องส่งสัญญาณไปเรียกยังอุปกรณ์เป้าหมายก่อน จึงจะมีการส่งข้อมูลกลับมา ในบางครั้งสามารถส่งข้อมูลเพียงอย่างเดียวและไม่ต้องการข้อมูลตอบกลับก็ได้



รูปที่ 2.9 การเชื่อมต่อระบบ I<sup>2</sup>C

## 2.3 ทฤษฎีระบบควบคุมวงปิด

หลักการควบคุมป้อนกลับ (Feedback Control Systems) ดังรูปที่ 2.10 เป็นหลักการพื้นฐานที่ใช้ในการควบคุมระบบพลวัตอย่างแพร่หลาย โดยเฉพาะการป้อนกลับแบบลบ (Negative Feedback) เพราะสัญญาณจากเซนเซอร์ (Measured Value) จะถูกนำไปหักล้างจากสัญญาณอ้างอิง (Reference Input) ได้เป็นผลต่างระหว่างค่าที่ต้องการ และสัญญาณจากอุปกรณ์ตรวจวัด (Sensor) เพื่อที่จะนำไปสร้างสัญญาณความคลาดเคลื่อน (Error) ซึ่งจะนำไปป้อนสู่ตัวควบคุม (Controller) และตัวควบคุมจะสร้างสัญญาณควบคุม (Control Signal) ป้อนสู่ระบบพลวัต (System) หลังจากนั้นจะนำสัญญาณขาออกของระบบพลวัตที่วัดได้จากตัวตรวจวัด มาป้อนสู่ระบบป้อนกลับต่อไปเช่นนี้เรื่อยๆ



รูปที่ 2.10 ระบบควบคุมป้อนกลับแบบลบ

ระบบควบคุมแบบป้อนกลับมีความได้เปรียบเหนือกว่าระบบควบคุมแบบวงเปิด ดังต่อไปนี้

1. สามารถกำจัดการรบกวนได้ เช่น ผลจากแรงเสียดทานที่ไม่ได้รวมอยู่ในแบบจำลองทางคณิตศาสตร์ของระบบ
2. สามารถรับประกันสมรรถนะได้มากขึ้นแม้กับแบบจำลองทางคณิตศาสตร์ที่มีตัวแปรที่มีความไม่แน่นอนอยู่ด้วย เช่น กรณีที่ผลจากการที่แบบจำลองทางคณิตศาสตร์ไม่สามารถอธิบายระบบได้อย่างสมบูรณ์แบบ
3. ระบบที่ไม่มีเสถียรภาพโดยธรรมชาติอยู่แล้ว สามารถทำให้มีเสถียรภาพได้หากติดตั้งตัวควบคุมที่เหมาะสม
4. ระบบมีความคงทนต่อความเปลี่ยนแปลงมากขึ้น แม้ในกรณีที่พารามิเตอร์ของระบบมีการเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ระบบสามารถปรับค่าสัญญาณขาออกตามสัญญาณอ้างอิงได้ดีมากขึ้นในปัญหา ระบบปรับค่าตาม

## 2.4 ทฤษฎีการทำงานของ LIDAR

LIDAR เป็นวิธีการสำรวจซึ่งวัดระยะทางถึงเป้าหมาย โดยการฉายเป้าหมายด้วยลำแสงความเข้มสูง (LASER) เป็นจังหวะและวัดแสงที่สะท้อนกลับมาด้วยตัวรับ ดังรูปที่ 2.11 แล้วใช้ความแตกต่างระหว่างเวลากลับคืนของลำแสง, ความเร็วของแสง และความยาวคลื่น เพื่อคำนวณระยะห่างระหว่างเป้าหมายกับแหล่งกำเนิดแสง



รูปที่ 2.11 การตรวจจับแสงที่สะท้อนกลับของ LIDAR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### วิธีการดำเนินงาน

#### 3.1 การวิเคราะห์ปัญหา

การวิจัยเริ่มต้นโดยการวิเคราะห์ความรุนแรงและรูปแบบของปัญหา ได้แก่ ความคลาดเคลื่อนของตำแหน่งจากระบบ GPS และขีดความสามารถในการรักษาตำแหน่งของตัวโดรน ซึ่งข้อมูลจะถูกใช้เพื่อเลือกวิธีการแก้ปัญหาที่เหมาะสมที่สุด

เบื้องต้น มีหลักการที่เป็นไปได้ 3 วิธี ได้แก่

1. สร้างแท่นลงจอดให้มีความกว้างรองรับความคลาดเคลื่อนของตำแหน่งทั้งหมด
2. ติดตั้งระบบเสริมเพื่อลดความคลาดเคลื่อนของตำแหน่ง
3. ใช้ระบบเคลื่อนที่ภาคพื้นในการเปลี่ยนตำแหน่ง เช่น รถลาก

#### 3.2 การกำหนดวิธีการแก้ไข

##### 3.2.1 การกำหนดหลักการ

ข้อมูลสมรรถภาพของ DJI Matrice 100 ซึ่งอ้างอิงจาก User Manual ดังตารางที่ 3.1 Hovering Accuracy หรือความคลาดเคลื่อนของตำแหน่งในแนวระดับคือ 2.5 เมตร ดังนั้น วิธีการสร้างแท่นลงจอดให้รองรับความคลาดเคลื่อนทั้งหมดจึงไม่เหมาะสมเพราะจะต้องมีความกว้างถึง 5 เมตร และจากการสอบถามผู้ใช้งาน สภาพพื้นที่ที่ระบบจะถูกนำไปใช้งานไม่สามารถถูกกำหนดได้ และอาจมีความต้องการติดตั้งหลายตำแหน่งซึ่งมีสภาพพื้นที่แตกต่างกัน ดังนั้นวิธีการที่ถูกเลือกมาใช้งานจึงเป็นวิธีติดตั้งระบบเสริมเพื่อลดความคลาดเคลื่อนของตำแหน่ง

### ตารางที่ 3.1 ค่าขีดจำกัดของ DJI Matrice 100

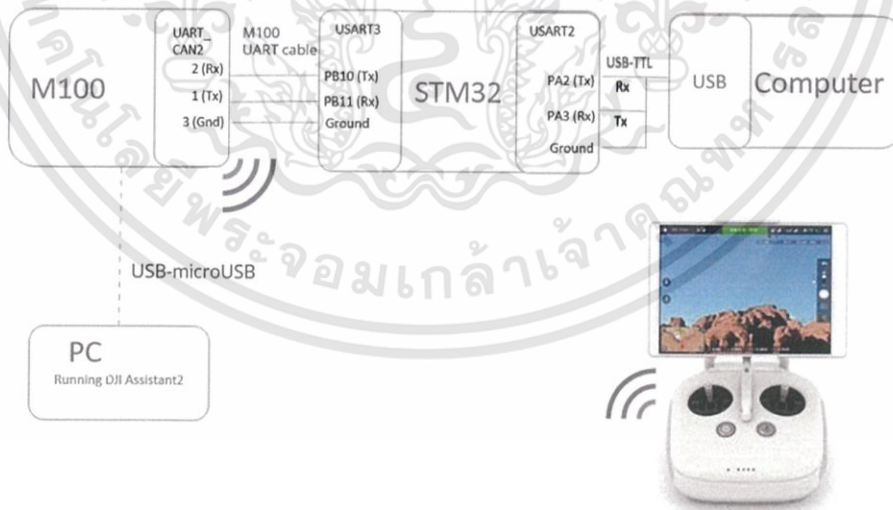
Performance	
Hovering Accuracy (P-Mode, with GPS)	Vertical: 0.5 m, Horizontal: 2.5 m
Max Angular Velocity	Pitch: 300°/s, Yaw: 150°/s
Max Pitch Angle	35°
Max Speed of Ascent	5 m/s
Max Speed of Descent	4 m/s
Max Wind Resistance	10 m/s

### 3.2.2 การกำหนดรูปแบบของระบบ [5]

จากการตัดสินใจเลือกวิธีติดตั้งระบบเสริมเพื่อลดความคลาดเคลื่อนของตำแหน่ง จำเป็นต้องตรวจสอบว่า N1 Flight Controller ของ DJI Matrice 100 รองรับการติดตั้งระบบเสริมหรือไม่, มีการสื่อสารข้อมูลแบบใด และระบบเสริมนั้นมีสิทธิ์ในการเข้าควบคุมโดรนมากเพียงใด

ข้อมูลจากเว็บไซต์ของ DJI แสดงวิธีการติดตั้งระบบเสริมรูปแบบต่างๆ ได้แก่

1. ระบบเสริมบนตัวโดรน (Onboard Computer) ดังรูปที่ 3.1
2. ระบบเสริมบนอุปกรณ์สื่อสารเคลื่อนที่ เช่น Smartphone (Mobile Device)



รูปที่ 3.1 การเชื่อมต่อ Onboard Computer กับ Flight Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งระบบที่น่าสนใจคือระบบ Onboard Computer เพราะมีการเชื่อมต่อกับ Flight Controller ผ่านสายสัญญาณ ซึ่งมีความหน่วงเวลาของสัญญาณน้อยกว่าการส่งข้อมูลผ่านคลื่นวิทยุไปยัง Mobile Device ซึ่งความเร็วจำเป็นสำหรับการส่งสัญญาณควบคุมตำแหน่งที่แม่นยำ ในขณะที่เทียบกันการเลือกใช้ Onboard Computer ทำให้ Sensor ในการหาตำแหน่งจำเป็นต้องอยู่บนตัวโดรนด้วย

### 3.2.3 การเลือก Onboard Computer

ข้อมูลจากเว็บไซต์ของ DJI ระบุว่า Onboard Computer ที่รองรับ ได้แก่

1. Microcontroller ตระกูล STM32
2. Computer ที่ใช้ระบบปฏิบัติการ Linux

จากการทดลองใช้ Microcontroller STM32F429 Discovery ดังรูปที่ 3.2 พบว่าการ Setup ระบบมีความซับซ้อนมากและฐานข้อมูลในการเรียนรู้ถูกปิดกั้นด้วย Paywall ดังนั้นจึงทดลองใช้ Computer ที่ใช้ระบบปฏิบัติการ Linux โดยเริ่มทดลองกับ Raspberry Pi 3 ดังรูปที่ 3.3 ซึ่งหาซื้อได้ง่ายและมีราคาไม่สูง สามารถค้นหาข้อมูลเกี่ยวกับการแก้ไขปัญหาต่างๆ ได้ง่ายจาก Community และ Forum



รูปที่ 3.2 STM32F429 Discovery



รูปที่ 3.3 Raspberry Pi 3

### 3.2.4 การเลือก Sensor

จากการสืบค้นข้อมูลเบื้องต้น ระบบที่นิยมใช้ในการหาค่าตำแหน่งที่แม่นยำในพื้นที่กลางแจ้งคือ

1. GPS แบบ Real-Time Kinematic (RTK) ดังรูปที่ 3.4 ซึ่งอาจมีความแม่นยำถึง  $\pm 0.02$  เมตร
2. Reference Point ซึ่งกำหนดโดยติดตั้งอุปกรณ์บนจุดที่กำหนด และใช้ Sensor ในการหาระยะทางของ Reference Point

ระบบ Real-Time Kinematic ยังไม่มีการใช้งานอย่างแพร่หลาย จึงมีราคาสูงมาก และส่วนใหญ่ถูกใช้ในการทหาร การติดตั้ง Reference Point จึงถูกนำมาใช้



รูปที่ 3.4 ระบบ RTK [4]

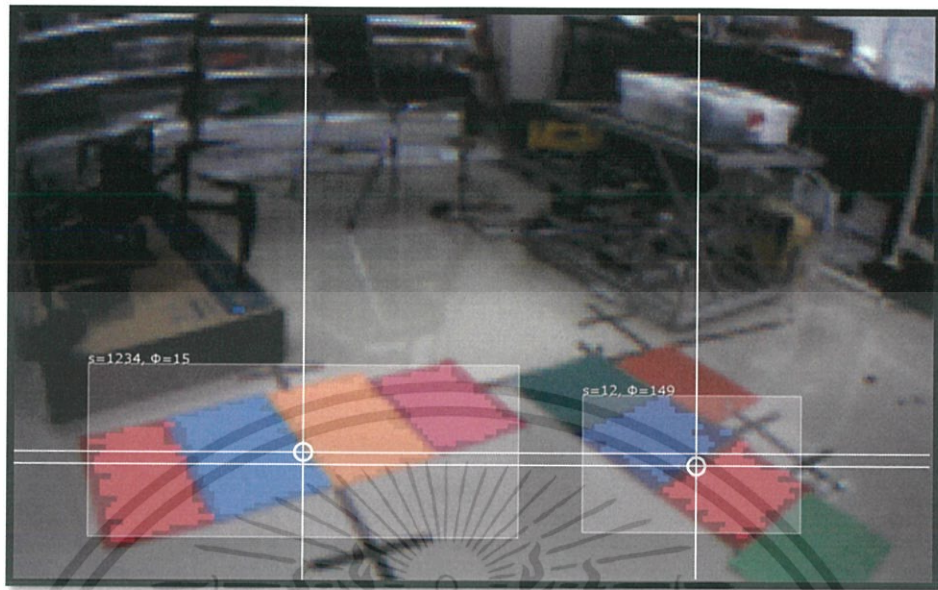
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบ Reference Point ที่นิยมมากที่สุดคือ การใช้การประมวลผลภาพ (Image Processing) เนื่องจากที่ราคาถูกและมีความแม่นยำเทียบเท่ากับรูปแบบอื่นๆ ซึ่ง Sensor ที่ถูกเลือกคือ กล้อง Pixy ดังรูปที่ 3.5 เนื่องจาก

1. มีขนาดเล็ก น้ำหนักเบา
2. มีการทำ Image Processing ในตัว และส่งข้อมูลที่ประมวลผลแล้ว ดังรูปที่ 3.6 ไปยัง Onboard Computer ได้ทันที
3. ใช้งานง่าย ไม่ต้องศึกษาการสร้างโปรแกรมสำหรับทำ Image Processing
4. ดัดแปลงให้ใช้ Infrared LED เป็น Reference Point ได้ซึ่งเหมาะกับสภาพแสงสว่างที่ไม่แน่นอน
5. รองรับ Interface หลายชนิด (UART, SPI, I2C, Analog)



รูปที่ 3.5 กล้อง Pixy



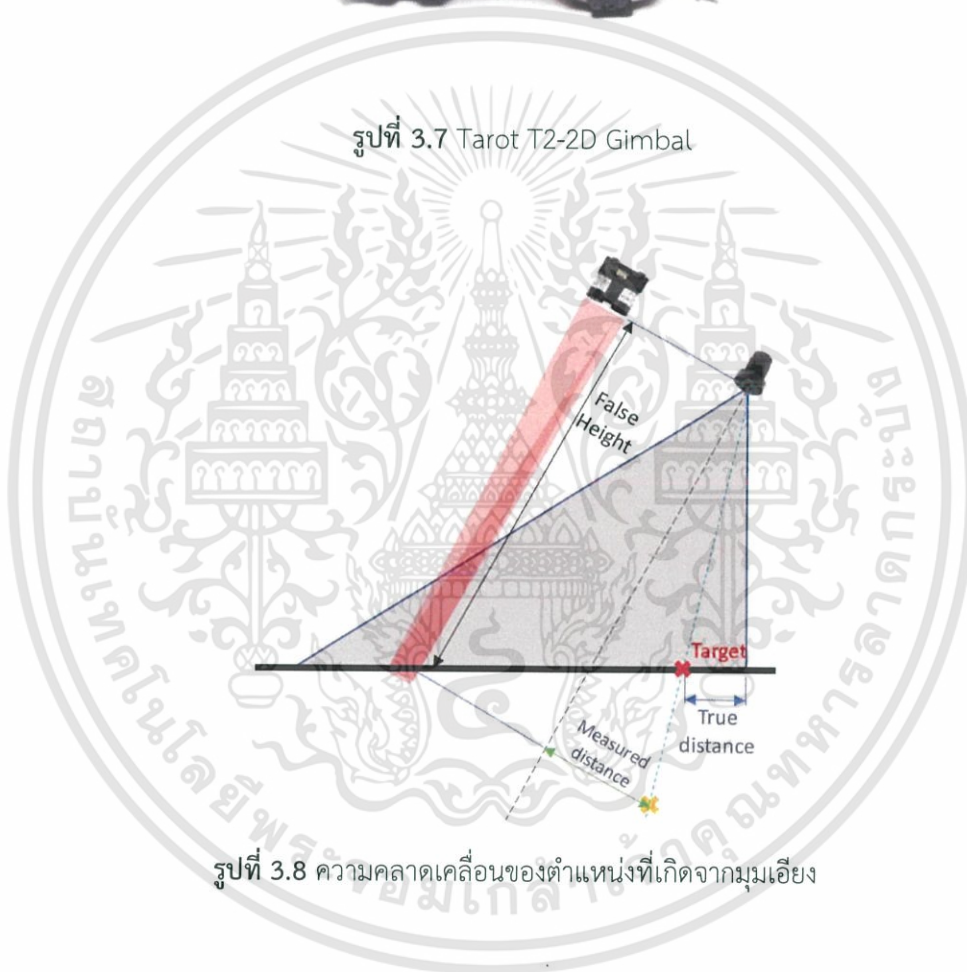
รูปที่ 3.6 ข้อมูลที่ Pixy ประมวลผลแล้ว

การใช้กล้องในการหาระยะของ Reference Point ในแนวระดับ เมื่อได้รับข้อมูลตำแหน่งในภาพแล้ว ต้องนำข้อมูลมาคำนวณเป็นระยะห่างทางกายภาพที่แท้จริง ซึ่งตัวแปรที่ใช้ในการคำนวณที่สำคัญคือ ระยะห่างจากกล้องไปยังเป้าหมาย และมุมระหว่างจอร์บภาพและพื้นผิวที่วาง Reference Point

การแก้ความคลาดเคลื่อนของตำแหน่งที่เกิดจากมุมเอียง สามารถนำค่ามุมเอียงที่วัดได้จาก IMU ของ DJI Matrice 100 เพื่อคำนวณหาตำแหน่งที่แท้จริงได้ แต่ในระบบการหมุน 3 มิตินั้นมีความซับซ้อนมาก จำเป็นต้องศึกษาระบบ Quaternion Orientation ซึ่งเป็นคณิตศาสตร์ขั้นสูงเพื่อหลีกเลี่ยงปัญหาที่อาจเกิดขึ้นจากความไม่เข้าใจ จึงแก้ปัญหาโดยใช้ Gimbal เพื่อรักษาระดับของ Sensor ให้หันลงพื้นโลกตลอดเวลา ทำให้ค่ามุมเอียง ถูกควบคุมไว้ด้วยการใช้ Gimbal ดังรูปที่ 3.7 เพื่อให้ Sensor อยู่ในแนวเดียวกับพื้นโลกตลอดเวลา และไม่เกิดความคลาดเคลื่อนจากมุมเอียง ดังรูปที่ 3.8



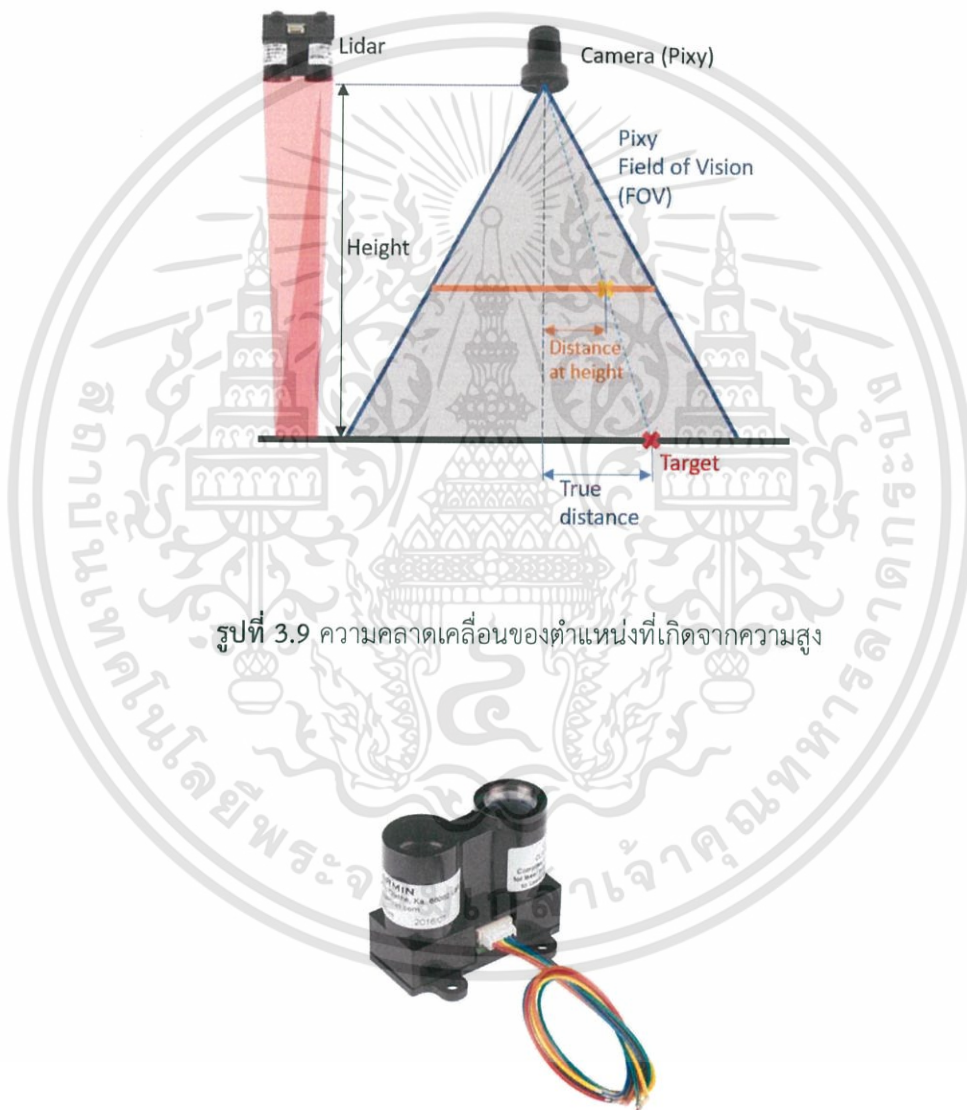
รูปที่ 3.7 Tarot T2-2D Gimbal



รูปที่ 3.8 ความคลาดเคลื่อนของตำแหน่งที่เกิดจากมุมเอียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำนวณความคลาดเคลื่อนจากความสูง ดังรูปที่ 3.9 จะต้องทราบระยะห่างระหว่างอุปกรณ์รับภาพกับพื้นผิวที่วาง Reference Point แต่จากตารางที่ 3.1 ความแม่นยำในการวัดความสูงของ Barometer บน DJI Matrice 100 คือ 0.5 เมตร ซึ่งอาจทำให้การทำงานไม่แม่นยำ จึงจำเป็นต้องเพิ่ม LIDAR ดังรูปที่ 3.10 ซึ่ง LIDAR จะต้องติดตั้งอยู่บน Gimbal เช่นเดียวกับกล้อง Pixy สำหรับวัดความสูงจากพื้น ซึ่งมีความแม่นยำถึง 0.05 เมตร และทำงานบนพื้นผิวขรุขระได้ดีกว่าระบบแบบ Ultrasonic



รูปที่ 3.9 ความคลาดเคลื่อนของตำแหน่งที่เกิดจากความสูง

รูปที่ 3.10 Garmin LIDAR Lite V3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

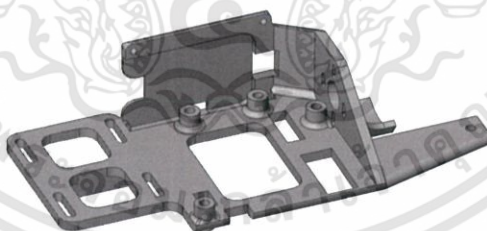
### 3.3 การดำเนินการแก้ไขปัญหา

#### 3.3.1 การติดตั้ง Sensor

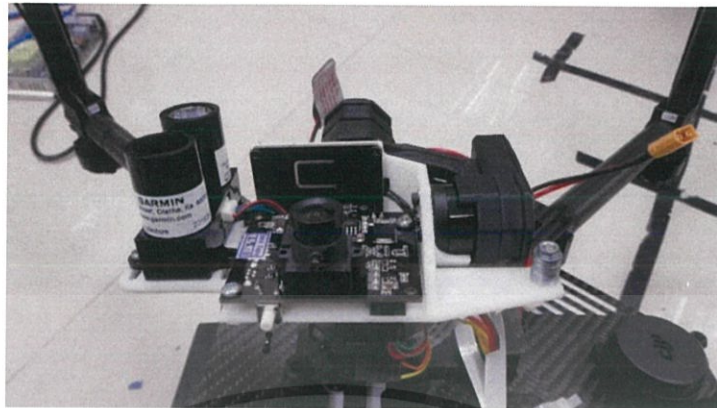
การติดตั้งอุปกรณ์บน Gimbal ชนิด Brushless จำเป็นต้องให้จุด Center of Gravity อยู่ในแกนเดียวกันกับแกนของมอเตอร์ แทนวาง Sensor ดังรูปที่ 3.11 จึงถูกออกแบบให้สามารถเลื่อนตำแหน่งอุปกรณ์ได้ เพื่อให้สามารถแก้ไขการกระจายน้ำหนักได้หากมีการเปลี่ยนแปลงอุปกรณ์ แต่ข้อจำกัดทางกายภาพของ Tarot T2-2D Gimbal ทำให้น้ำหนักถ่วงไปด้านหลังมากเกินไป แทนวาง Sensor ถึงถูกออกแบบใหม่ให้สามารถติดตั้งตุ้มถ่วงน้ำหนักได้ ดังรูปที่ 3.12 จึงได้ระบบที่ได้สมดุล ดังรูปที่ 3.13



รูปที่ 3.11 การพัฒนาโครงสำหรับติดตั้ง Sensor โดยเครื่องพิมพ์ 3 มิติ



รูปที่ 3.12 โครงสำหรับติดตั้ง Sensor ที่ติดตั้งตุ้มถ่วงน้ำหนักได้



รูปที่ 3.13 Pixy และ LIDAR ที่ติดตั้งบน Gimbal พร้อมน้ำหนักถ่วง (ขวา)

### 3.3.2 การเขียนโปรแกรม

โปรแกรมที่ใช้สั่งการระบบจะต้องประมวลผลบน CPU ของ Raspberry Pi 3 และส่งเป็นคำสั่งควบคุมการบินไปยัง Flight Controller ซึ่ง DJI ได้เปิดให้ผู้ใช้งาน Download API สำหรับส่งคำสั่งควบคุมการบินไว้ในรูปแบบของโปรแกรม C++ และ ROS\_CPP Node

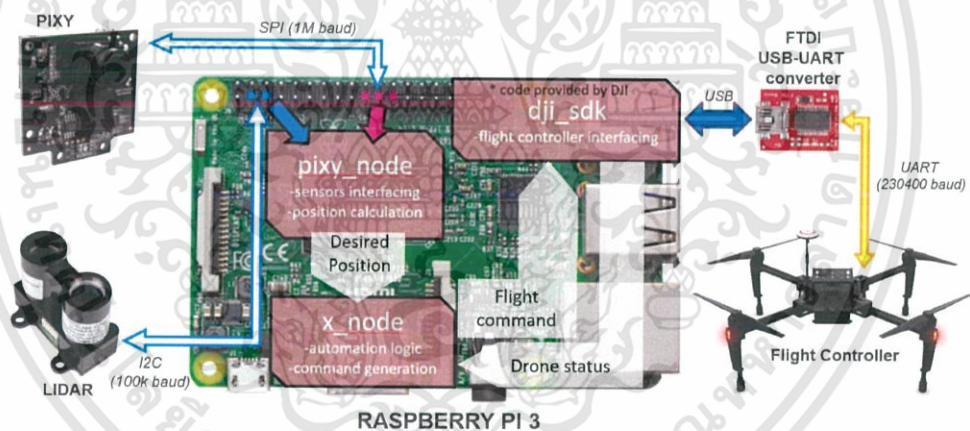
Robot Operating System หรือ ROS เป็นรูปแบบการจัดการทำงานโปรแกรม ที่ออกแบบให้โปรแกรมมีการทำงานแยกเป็นหลายส่วน เรียกว่า Node โดยแต่ละ Node รับ-ส่งสารข้อมูลต่างๆ ผ่าน Topic และ Service

ROS ถูกเลือกมาใช้เนื่องจากเหตุผลหลักดังนี้

- มี Documentation ที่ครบถ้วน
- มีการอธิบายขั้นตอนการ Install
- มี Tutorial สำหรับผู้ใช้ใหม่
- มี Community ที่มีผู้ช่วยตอบและแก้ไขปัญหาดังกล่าวโดยไม่เสียค่าใช้จ่าย

บนระบบ Onboard System ดังรูปที่ 3.14 นี้ โปรแกรมถูกออกแบบให้แบ่งเป็น 4 Node ได้แก่

1. `pixy_node` ทำหน้าที่รับและประมวลผลข้อมูลจากกล้อง Pixy และ LIDAR แล้วส่งค่าระยะทางไปยังเป้าหมายให้ `x_node` มีแผนผังการทำงานดังรูปที่ 3.15
2. `x_node` ทำหน้าที่ประมวลผลสถานะของโดรน เพื่อตัดสินใจการทำงานขั้นตอนต่างๆ ตั้งแต่การเริ่ม Take-off, การบินไปยังเป้าหมายเพื่อเก็บข้อมูล และการกลับสู่ตำแหน่งลงจอด แล้วสร้างคำสั่งที่เหมาะสมไปยัง `dji_sdk` มีแผนผังการทำงานดังรูปที่ 3.16
3. `bms_node` ทำหน้าที่ควบคุมวงจรชาร์จแบตเตอรี่ และสื่อสารกับระบบภาคพื้น มีแผนผังการทำงานดังรูปที่ 3.17
4. `dji_sdk` ซึ่งเป็น API ของ DJI ทำหน้าที่รับคำสั่งจาก `x_node` เพื่อส่งต่อไปยัง Flight Controller พร้อมทั้งรับข้อมูลสถานะต่างๆ ของโดรน จาก Flight Controller และส่งกลับไปยัง `x_node`



รูปที่ 3.14 แผนผัง Onboard System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

pixy\_node คือ ส่วนสำคัญที่สุดในการเพิ่มความแม่นยำในการลงจอด ซึ่งทำงานด้วยระบบควบคุมวงปิดแบบลบ (Negative Feedback Closed Loop Control) อย่างง่าย ใช้จุดอ้างอิง (Reference Point) ที่จุดกึ่งกลางของจอร์รับภาพของกล้อง Pixy และตำแหน่งของป้ายแถบสีที่คำนวณความคลาดเคลื่อนแล้ว เป็นค่าที่วัดได้ (Measured Output) ค่าที่ส่งไปยัง Flight Controller ผ่าน x\_node ซึ่งเป็นผลต่างของตำแหน่ง (Error) จึงเป็นค่าระยะทางในหน่วยเมตร ซึ่ง Flight Controller จะประมวลผลเป็นการควบคุมมอเตอร์เอง โปรแกรมนี้จึงไม่ต้องมีการคำนวณที่ซับซ้อน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

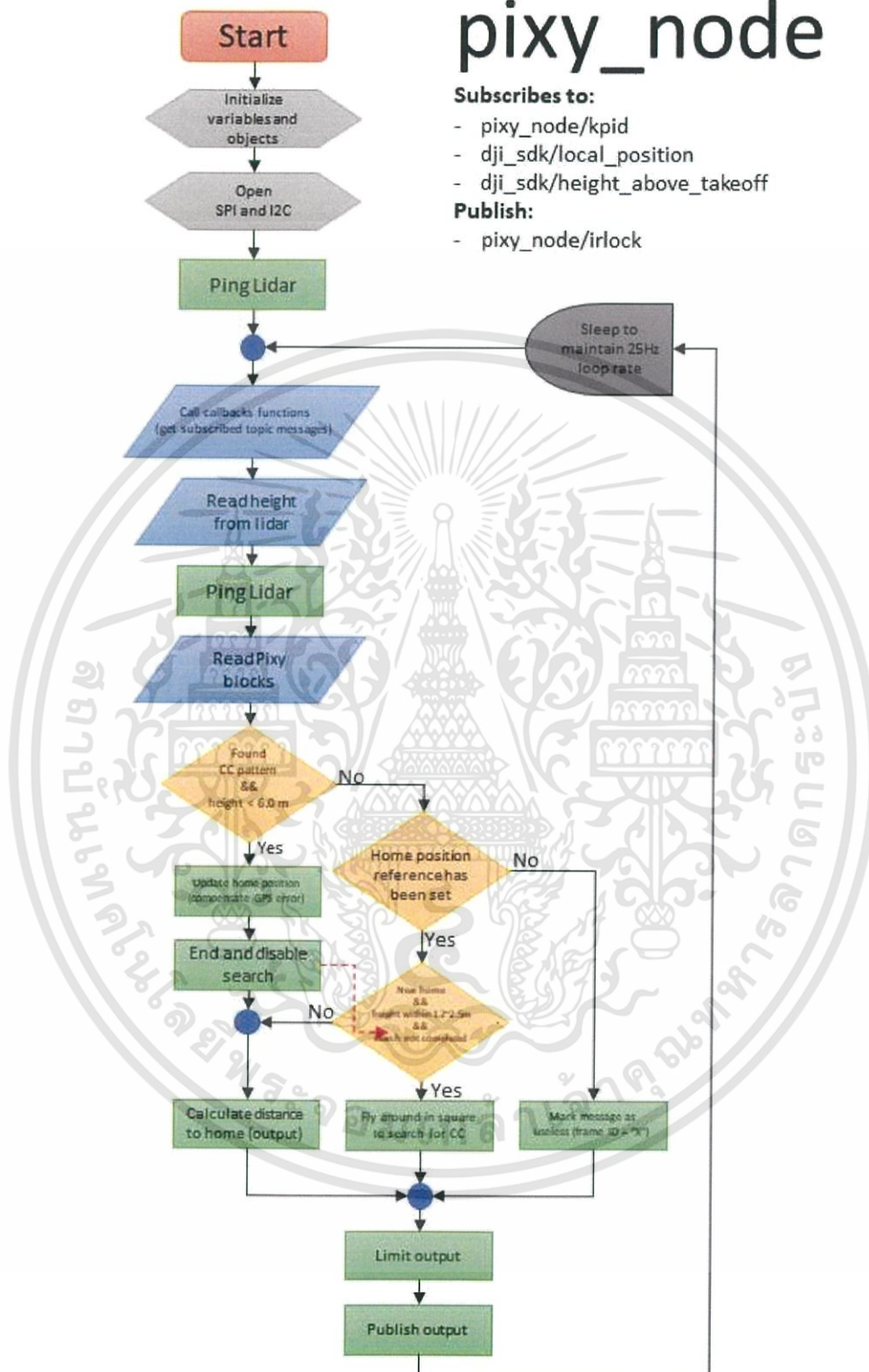
# pixy\_node

## Subscribes to:

- pixy\_node/kpid
- dji\_sdk/local\_position
- dji\_sdk/height\_above\_takeoff

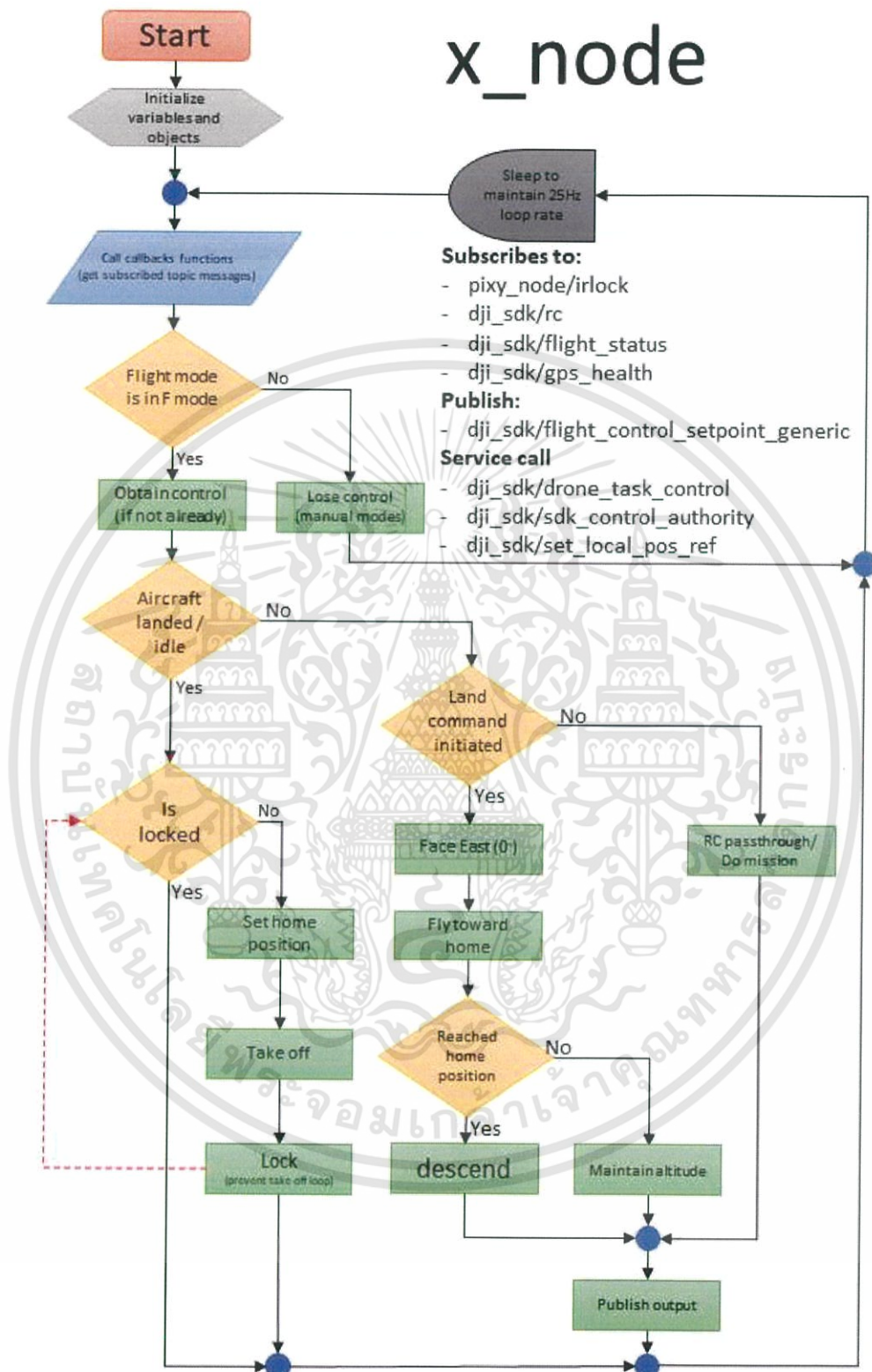
## Publish:

- pixy\_node/irlock



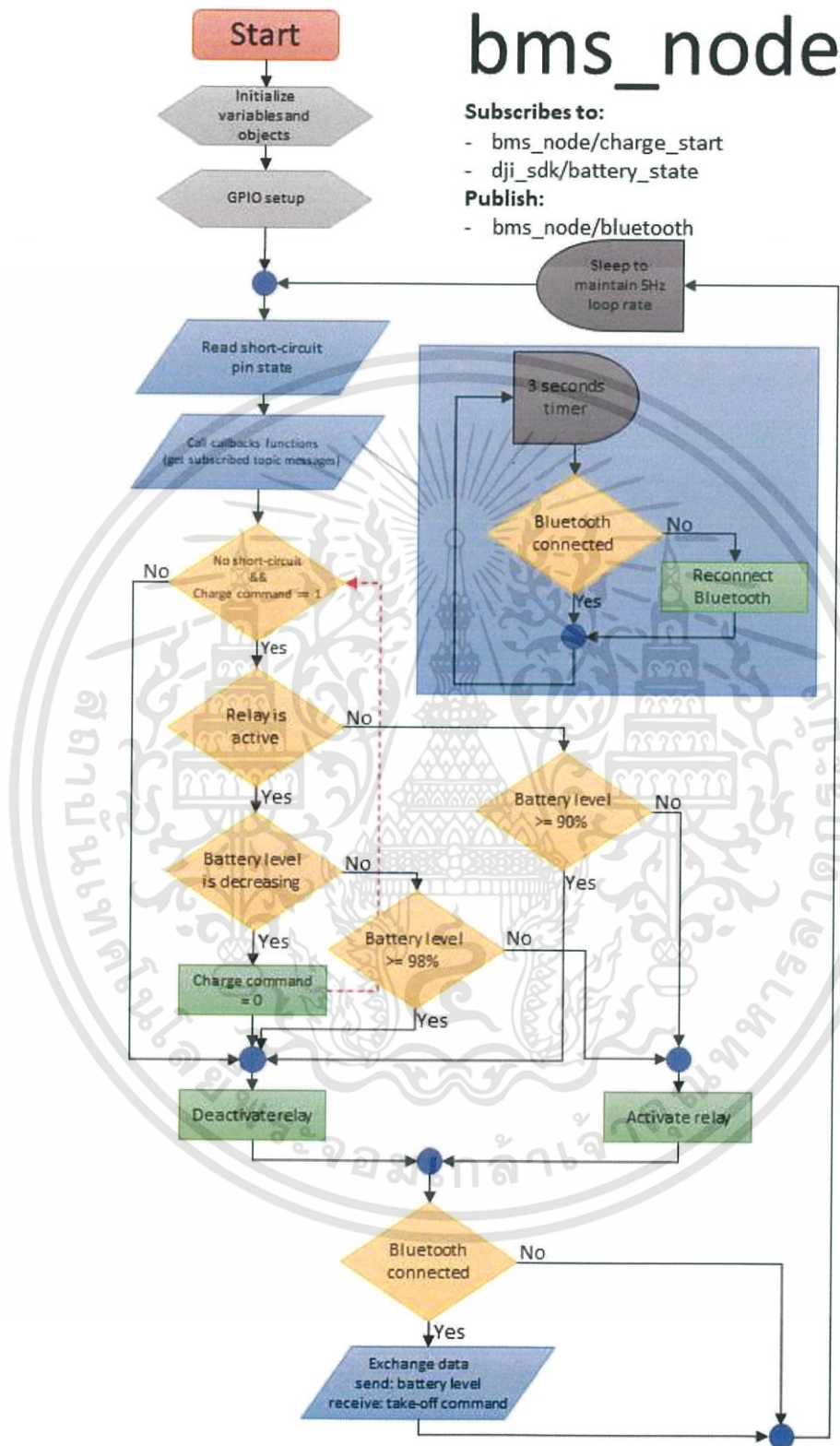
รูปที่ 3.15 Flowchart ของ pixy\_node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 Flowchart ของ x\_node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



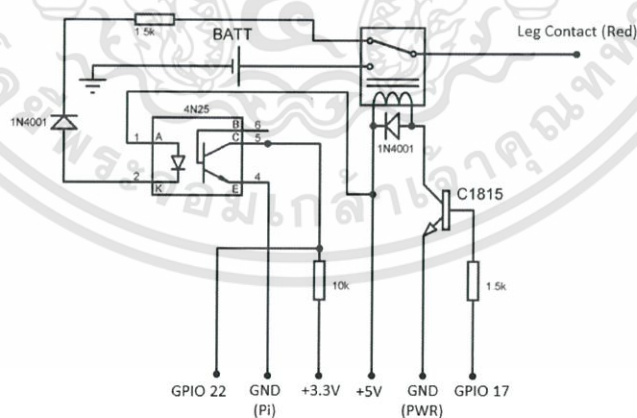
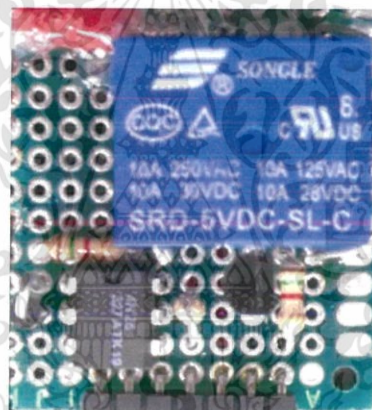
รูปที่ 3.17 Flowchart ของ bms\_node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.3 การสร้างวงจรไฟฟ้า

กล้อง Pixy และ LIDAR สามารถเชื่อมต่อและใช้งานร่วมกับ Raspberry Pi 3 ได้โดยตรงผ่านการเชื่อมต่อแบบ SPI และ I<sup>2</sup>C ตามลำดับ

ส่วนวงจรสำหรับควบคุมการชาร์จ ดังรูปที่ 3.18 นั้น จำเป็นต้องใช้ Relay เนื่องจากสามารถรองรับกระแสและแรงดันไฟฟ้าได้สูง แต่มีความต้านทานต่ำ ซึ่ง Relay มักมีแรงดันไฟฟ้าขั้นต่ำในการทำงานที่ 5 โวลต์ แต่ GPIO ของ Raspberry Pi 3 นั้นทำงานที่ 3.3 โวลต์ จึงต้องสร้างวงจร Amplifier ขึ้น และเพื่อให้ Raspberry Pi 3 สามารถรับรู้ถึงสถานะการจ่ายไฟของระบบชาร์จ ซึ่งอาจสูงถึง 26 โวลต์ จึงจำเป็นต้องมีการใช้วงจรช่วยป้องกันไฟฟ้าแรงสูงเข้าไปสู่ Onboard Computer โดย Opto-isolator ถูกนำมาใช้งาน

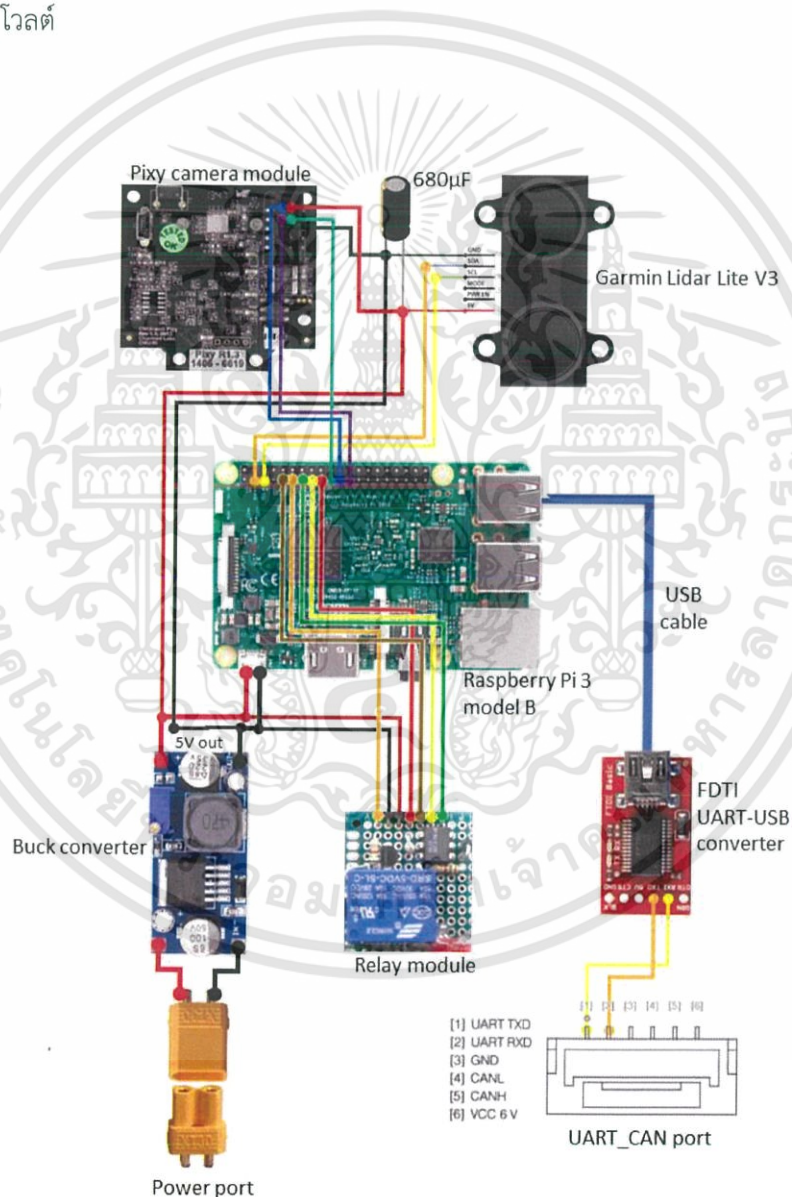


รูปที่ 3.18 วงจรควบคุมการชาร์จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรในรูปที่ 3.18 ทำหน้าที่ควบคุมการเชื่อมต่อ Contact สำหรับชาร์จแบตเตอรี่ โดย NC เชื่อมต่อผ่าน Opto-isolator ไปยัง Raspberry Pi 3 เพื่อใช้ตรวจสอบการลัดวงจร และการสลับขั้วของวงจรชาร์จแบตเตอรี่

อุปกรณ์ทุกชิ้นบน Onboard System แสดงในรูปที่ 3.19 ใช้ Power Input 5 โวลต์ ซึ่งได้รับจาก Power Port ของ DJI Matrice 100 ผ่าน Buck Converter เพื่อปรับลดแรงดันไฟฟ้า เนื่องจาก Power Port ดังกล่าว เชื่อมต่อโดยตรงกับแบตเตอรี่ของ DJI Matrice 100 ซึ่งมีแรงดันไฟฟ้าประมาณ 20 ถึง 26 โวลต์



รูปที่ 3.19 วงจรทั้งหมดของ Onboard System

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.4 การสร้างแท่นลงจอด

จากการทดลอง Reference Point ที่กล้อง Pixy สามารถตรวจจับได้ดีที่สุดคือ ป้ายแถบสีแดง และสีน้ำเงิน ส่วนความสูงที่สามารถตรวจจับได้ขึ้นกับขนาดของป้ายแถบสี (CC) แต่ขนาดที่ใหญ่เกินไปทำให้ป้ายบางส่วนอยู่นอกมุมมองของกล้องขณะบินอยู่ต่ำ ขนาดที่เหมาะสมจึงอยู่ที่ประมาณ  $8 * 8$  เซนติเมตรต่อสี่ ซึ่งสามารถตรวจจับได้ที่มีความสูง 2 เมตร ถึง 0.15 เมตร

ขั้วสำหรับชาร์จทำจากแผ่นวงจรเคลือบ Nickel-plated PCB ขนาด  $30 * 30$  เซนติเมตร วางห่างจากกันวัดจากศูนย์กลาง 60 เซนติเมตร ซึ่งทำให้ขั้วชาร์จบนตัวไดรอนไม่สามารถแตะบนแผ่นวงจรเดียวกันได้ ป้องกันการความเป็นไปได้ในการลัดวงจร แผ่นวงจรดังกล่าวถูกติดตั้งบนแผ่น Acrylic สีเหลี่ยมจัตุรัส กว้าง 1 เมตร หนา 5 มิลลิเมตร และเสริมความแข็งแรงด้วยโครงอะลูมิเนียม ดังรูปที่ 3.20



รูปที่ 3.20 แท่นลงจอดสำหรับทดลอง

## บทที่ 4

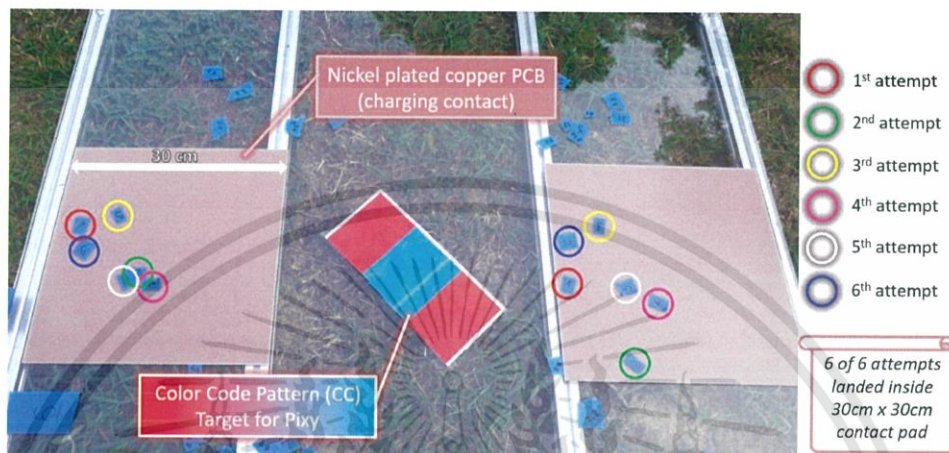
### ผลการดำเนินงาน

การทดลองจำลองการบินแบบ Waypoint โดยควบคุมด้วยมือให้โดรนบินขึ้น สูง 15 เมตร และออกห่างจากจุด Take Off อย่างน้อย 40 เมตรไปในทิศทางต่างๆ แล้วเปิดใช้งานระบบอัตโนมัติ สังเกตการลงจอดโดยไม่เข้าควบคุม จนกว่าใบพัดจะหยุดหมุนและเข้าสู่สถานะปลอดภัย (Disarmed) ดังรูปที่ 4.1



รูปที่ 4.1 การทดลองลงจอดแบบแม่นยำ

ผลการทดลองที่ถูกรับที่ไว้ ดังรูปที่ 4.2 ทั้งหมด 6 การลงจอด ในทั้ง 6 การลงจอด ขั้วสำหรับ  
ชาร์จสัมผัสกับแผ่น PCB บนแท่นชาร์จ



รูปที่ 4.2 ตำแหน่งสัมผัสของขั้วในการลงจอดแต่ละครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

# สรุปผลและข้อเสนอแนะ

### 5.1 สรุปผล

ผลการทดลองเบื้องต้นเป็นที่น่าพอใจ ระบบสามารถทำงานได้ตามความต้องการทั้งหมด สามารถแก้ไขปัญหที่เกิดขึ้นได้เอง และสามารถป้องกันปัญหาจากความผิดพลาดที่อาจเกิดขึ้นได้ ปัญหาที่เคยเกิดขึ้นทั้งหมดได้ถูกวิเคราะห์และแก้ไขที่ต้นเหตุอย่างถูกต้อง อย่างไรก็ตามระบบถูกทดสอบในสภาพแวดล้อมที่เหมาะสมคือ แสงลมน้อย และมีแสงสว่างมาก ระบบอาจไม่สามารถทำงานได้อย่างถูกต้องในสภาวะอื่น หรือช่วงเวลาอื่น

### 5.2 ปัญหาและวิธีการแก้ไขปัญหา

1. ข้อมูลที่จำเป็นสำหรับการใช้งานมีน้อยมาก โดยเฉพาะวิธีการใช้งาน API ซึ่งข้อมูลทั้งหมดเป็นภาษาอังกฤษ และ Tutorial บางส่วนไม่ถูกต้อง  
วิธีการแก้ไขปัญหา : ไม่ค้นหาข้อมูลด้วยภาษาไทย วิเคราะห์ข้อความที่ระบบแจ้งเมื่อเกิดความผิดพลาด และการศึกษาวิธีการใช้งาน API โดยการอ่าน Source Code
2. การตรวจสอบการลัดวงจรเมื่อมีการจ่ายไฟสลับขั้วทำให้ Raspberry Pi 3 ชาร์ต  
วิธีการแก้ไขปัญหา : ใช้ Opto-isolator ป้องกันแรงดันไฟฟ้าที่อันตรายเข้าไปยัง CPU
3. วงจรภายในแบตเตอรี่หยุดการทำงานและหยุดจ่ายพลังงานเมื่อชาร์จเต็มแล้ว  
วิธีการแก้ไขปัญหา : สิ้นสุดการชาร์จที่ 98 เปอร์เซ็นต์ แล้วเริ่มการชาร์จเมื่อประมาณประจุลดต่ำกว่า 90 เปอร์เซ็นต์
4. บางครั้งป้ายแถบสีหลุดอยู่นอกมุมมองในการรับภาพของกล้อง ทำให้ไม่สามารถระบุตำแหน่งที่แม่นยำได้  
วิธีการแก้ไขปัญหา : เพิ่มขั้นตอนการบินวนรอบๆ พิกัดจาก GPS เพื่อเพิ่มพื้นที่ที่สามารถตรวจจับป้ายแถบสีได้
5. โปรแกรมสั่งการไม่สามารถเริ่มเองได้หลังจากระบบปิดตัวอย่างสมบูรณ์  
วิธีการแก้ไขปัญหา : สร้าง Startup Script เพื่อจำลองการสั่งเริ่มโปรแกรมทุกครั้งที่เปิดระบบใหม่
6. กล้อง Pixy ทำงานได้ไม่ดีเมื่อความสว่างเปลี่ยนแปลงไปหลายๆ  
วิธีการแก้ไขปัญหา : ไม่ใช้ระบบ Auto Exposure และทำการ Calibrate แสงในสภาพแสงที่อยู่ตรงกลางระหว่างสภาวะทำการที่มีมืดที่สุดและสว่างที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3 ข้อเสนอแนะ

1. ควรมีการทดลองระบบเพิ่มเติม เพื่อให้มั่นใจว่าระบบมีความน่าเชื่อถือ ซึ่งจำเป็นสำหรับการนำระบบไปใช้งานจริงในอุตสาหกรรม
2. ระบบยังมีข้อจำกัดในหลายๆ ด้าน ซึ่งควรมีการแก้ไข เช่น ระบบยังไม่มีกรับมือกรณีที่อุปกรณ์ขึ้นไต่ขึ้นหนึ่งซาร์ต, แสงสว่างจำเป็นต้องใกล้เคียงกับค่าที่ใช้ในการ Calibrate, ระบบสูญเสียความแม่นยำเมื่ออากาศยานอยู่ใกล้กับพื้นดินมากๆ
3. สามารถลดน้ำหนักของระบบลงได้โดยการคำนวณความคลาดเคลื่อน จากมุมเอียง ด้วยคณิตศาสตร์แทนการใช้ Gimbal
4. วัสดุที่ใช้ยึดติดอุปกรณ์หลายชิ้นยังเป็นเพียงต้นแบบ ผลิตขึ้นด้วยวัสดุ PLA ซึ่งไม่ทนทาน หากนำไปใช้ในระยะเวลาควรเลือกใช้วัสดุอื่น เช่น ABS



## เอกสารอ้างอิง

- [1] Msardonini. **The Basic Physics of the Quad-Copter** (Online). 22 มกราคม 2562.  
Available at: <https://beaglequad.wordpress.com/2015/10/03/the-basic-physics-of-the-quad-copter/>
- [2] Sonthaya Nongnuch. **ESP32 เบื้องต้น :: การสื่อสารแบบอนุกรม** (Online).  
22 มกราคม 2562.  
Available at: <https://www.ioxhop.com/article/68/esp32-เบื้องต้น-บทที่-7-การสื่อสารแบบอนุกรม>
- [3] Wisit P. **การติดต่อสื่อสารด้วย SPI : Serial Peripheral Interface** (Online).  
22 มกราคม 2562.  
Available at: <http://www.123microcontroller.com/Hardware-Interfacing/SPI-Serial-Peripheral-Interface-communication>
- [4] **RTK-G + Datalink Pro Pack** (Online). 22 มกราคม 2562.  
Available at: <https://store.dji.com/product/rtk-g-datalink-pro-pack>
- [5] **Hardware Setup Guide** (Online). 17 สิงหาคม 2561.  
Available at: <https://developer.dji.com/onboard-sdk/documentation/development-workflow/hardware-setup.html>

## ประวัติผู้เขียน

ชื่อ-นามสกุล นายพิชญ์ หุตะเจริญ  
 วัน เดือน ปีเกิด 8 กุมภาพันธ์ พ.ศ.2540  
 ที่อยู่ปัจจุบัน 24 ซ.วชิรธรรมสาธิต 37 ถ.สุขุมวิท 101/1 แขวงบางจาก เขตพระโขนง  
 จังหวัดกรุงเทพมหานคร 10260  
 เบอร์โทรศัพท์ 081-101-1332  
 Email Pitt\_4231@hotmail.com

## ประวัติการศึกษา

พ.ศ. 2547-2548 สำเร็จการศึกษาระดับประถมศึกษาชั้นปีที่ 1-2  
 จาก โรงเรียนสาธิตบางนา สมุทรปราการ  
 พ.ศ. 2549-2552 สำเร็จการศึกษาระดับประถมศึกษาชั้นปีที่ 3-6  
 จาก โรงเรียนสารสาสน์วิเทศร่มเกล้า กรุงเทพมหานคร  
 พ.ศ. 2553-2555 สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้น  
 จาก โรงเรียนสารสาสน์วิเทศร่มเกล้า กรุงเทพมหานคร  
 พ.ศ. 2556-2558 สำเร็จการศึกษาระดับมัธยมศึกษาตอนปลาย  
 สายการเรียนคณิตศาสตร์-วิทยาศาสตร์  
 จาก โรงเรียนสารสาสน์วิเทศร่มเกล้า กรุงเทพมหานคร  
 พ.ศ. 2559-2561 ศึกษาต่อบัณฑิตศึกษา หลักสูตรวิศวกรรมศาสตรมหาบัณฑิต  
 สาขาวิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์  
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
 กรุงเทพมหานคร

## ประวัติการทำงาน

พ.ศ. 2561 ฝึกงานที่ บริษัท เอสซีจี แพคเกจจิ้ง จำกัด (มหาชน) ฝ่ายออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้