

---

# บทที่ 1 การทดลอง

---

## 1.1 การทดลองที่ 1 การทำ Binding Remote กับ Receiver

### จุดประสงค์

เพื่อทำการจูนสัญญาณของ Remote กับ Receiver ให้ตรงกัน

### อุปกรณ์

1. Remote Flysky FS-TH9X
2. Receiver Flysky FS-R9B

### วิธีการทดลอง

1. ทำการต่อสาย Binding ที่ช่อง BIND ของ Receiver
2. ทำการต่อ Battery ที่ช่อง BAT ของ Receiver



3. จะสังเกตเห็นไฟที่ Receiver กระพริบ



4. จากนั้นกดปุ่ม Bind Range Test ที่ด้านหลังรีโมทค้างไว้แล้วเปิดสวิตช์ที่รีโมท



5. จะสังเกตเห็นไฟที่ Receiver หยุดกระพริบเป็นอันว่าสามารถใช้งานได้



6. จากนั้นทำการถอดสาย Binding และสาย Battery ออกจากตัว Receiver
7. ทำการปิดสวิตช์ที่รีโมท
8. ทำการต่อ Battery ที่ Receiver จากนั้นทำการเปิดสวิตช์ที่รีโมท
9. สังเกตไฟที่ตัว Receiver ถ้าติดก็แสดงว่าใช้งานได้ปกติ

#### จากการทดลองจนสัญญาณ

ปรากฏว่า Remote กับ Receiver สามารถเชื่อมต่อกันได้

## 1.2 การทดลองที่ 2 การโปรแกรมการทำงานของ Remote Flysky FS - TH9X

### จุดประสงค์

เพื่อที่จะทำให้สามารถทำการ Calibrate ได้ง่ายยิ่งขึ้น

### อุปกรณ์

1. Remote Flysky FS-TH9X

### วิธีการทดลอง

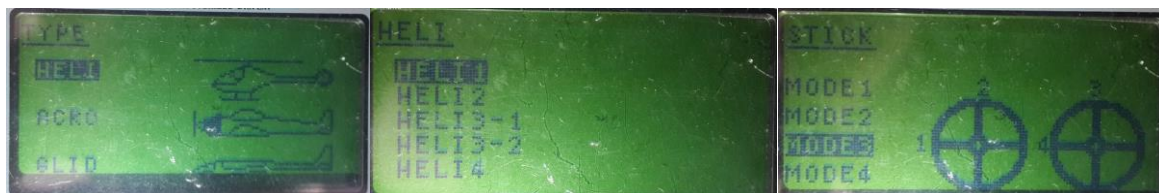
- 1.ทำการเปิดสวิตช์ตัวรีโมท และทำการดันสวิตช์ทุกตัวให้อยู่ตำแหน่งเริ่มต้นแล้วจะแสดงหน้าจอ ดังภาพ



- 2.ทำการเลือกโหมดของการส่งสัญญาณเข้าสู่ตัวรับสัญญาณ โดยให้ทำการเลือกเป็น PPM (Pulse Position Modulation) เนื่องจากต้องทำการส่งสัญญาณเพื่อใช้ในการควบคุม ESC



- 3.ทำการเลือกโหมดการบิน โดยให้ทำการเลือกโหมดเป็น HELI -> HELI 1 ซึ่งจะมีการควบคุม 4 ทิศทางหรือ 4 ช่องสัญญาณ Ch1 -> Roll , Ch2 -> Pitch , Ch3 -> Throttle และ Ch4 -> Yaw และทำการเลือกตำแหน่งของตำแหน่งควบคุม



4.ทำการเลือกช่องสัญญาณเพิ่มเติม ซึ่งใช้ในการเป็นสวิทช์ควบคุมการใช้งานเป็นโหมดต่าง ๆ เช่น Stabilize , Loiter , RTL เป็นต้น



5.จากนั้นทำการตรวจสอบว่าหลังจากการโปรแกรมข้างต้นนั้นทำงานถูกต้องหรือไม่ โดยสามารถดูได้ที่ DISPLAY ของรีโมทได้เบื้องต้น ซึ่งจะทำให้การ Calibrate Radio ซึ่งจะแสดงช่วงของสัญญาณที่ส่งออกจากตัวรีโมทเข้าสู่ตัวรับสัญญาณว่ามีค่าเท่าใดบ้าง



### 1.3 การทดลองที่ 3 การรีเซ็ตค่าเริ่มต้น (Default)ให้กับบอร์ด ArduPilotMega

#### จุดประสงค์

เพื่อเป็นการรีเซ็ตค่าเริ่มต้นให้เป็นค่า Default และ reset boot loader เพื่อป้องกันการเกิด error ต่าง ๆ

#### อุปกรณ์

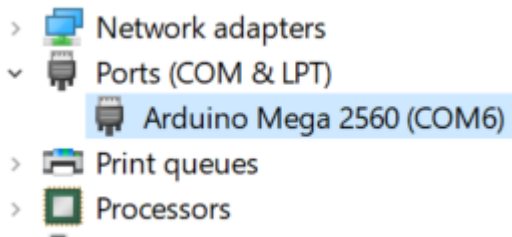
- 1.บอร์ด ArduPilotMega
- 2.สาย Micro USB
- 3.โปรแกรม Arduino IDE

#### วิธีการทดลอง

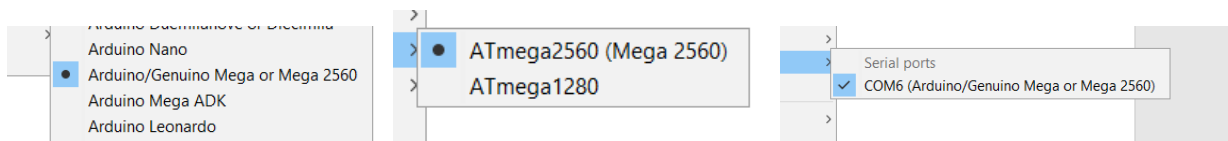
- 1.ทำการเสียบสาย Micro USB ระหว่างตัวบอร์ดกับพอร์ท USB
- 2.ทำการเปิดโปรแกรม Arduino IDE ขึ้นมา



- 3.ทำการติดตั้ง Driver ของบอร์ด ArduPilotMega



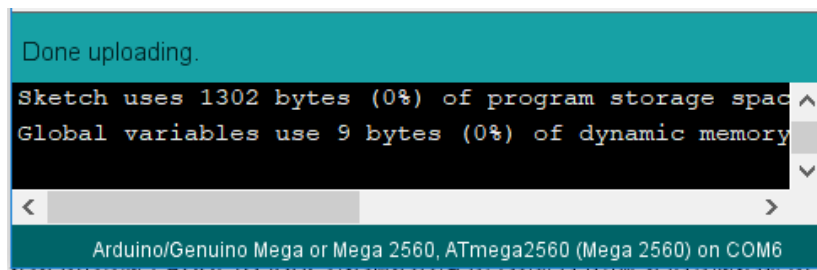
- 4.ทำการเลือก board, processor และ port ให้ตรงกับที่ใช้โดยไปที่เมนู Tools



- 5.หลังจากตั้งค่าในโปรแกรมแล้วก็ให้ไปที่เมนู File -> Examples -> EEPROM -> eeprom\_clear



6. จากนั้นทำการ compile และ burn ลงบอร์ด ArduPilotMega



7. เมื่อขึ้นคำว่า Done uploading แสดงว่า burn สำเร็จแล้ว ไฟตรงบอร์ดจะขึ้นสีเขียวค้างไว้และไม่มีการกระพริบใดๆ เนื่องจากไม่มีตัว Boot loader บอร์ดจึงไม่สามารถทำงานได้ จำเป็นต้องทำการลงเฟิร์มแวร์ใหม่ก่อน

## 1.4 การทดลองที่ 3 การติดตั้งเฟิร์มแวร์ให้กับบอร์ด ArduPilotMega

### จุดประสงค์

เพื่อเป็นการติดตั้งเฟิร์มแวร์เพื่อให้บอร์ด ArduPilotMega สามารถทำงานได้

### อุปกรณ์

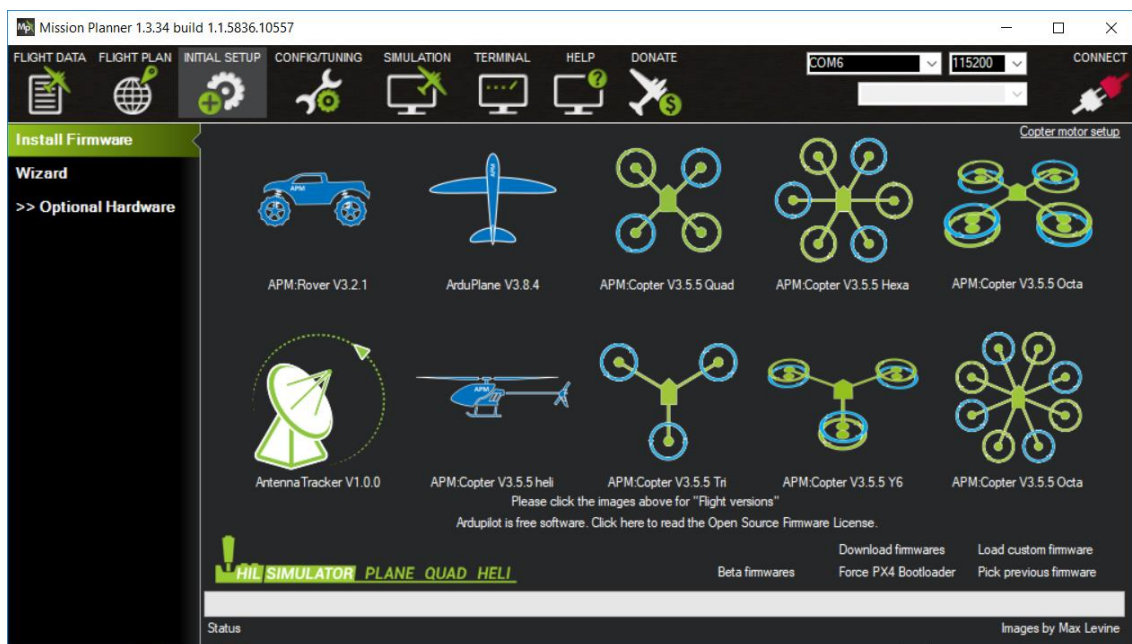
- 1.บอร์ด ArduPilotMega
- 2.สาย Micro USB
- 3.โปรแกรม Mission Planner

### วิธีการทดลอง

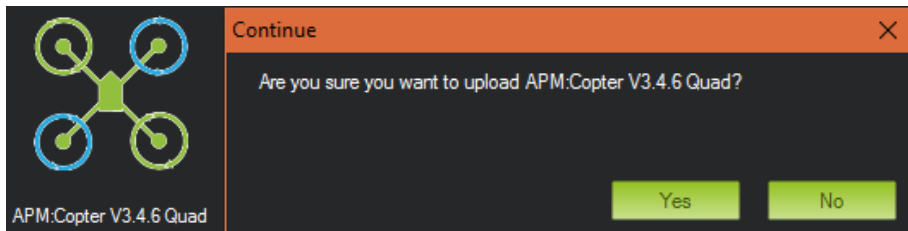
- 1.ทำการเสียบสาย micro USB ตรงตัวบอร์ดและเสียบเข้า Port USB
- 2.ทำการเปิดโปรแกรม Mission Planner ขึ้นมา



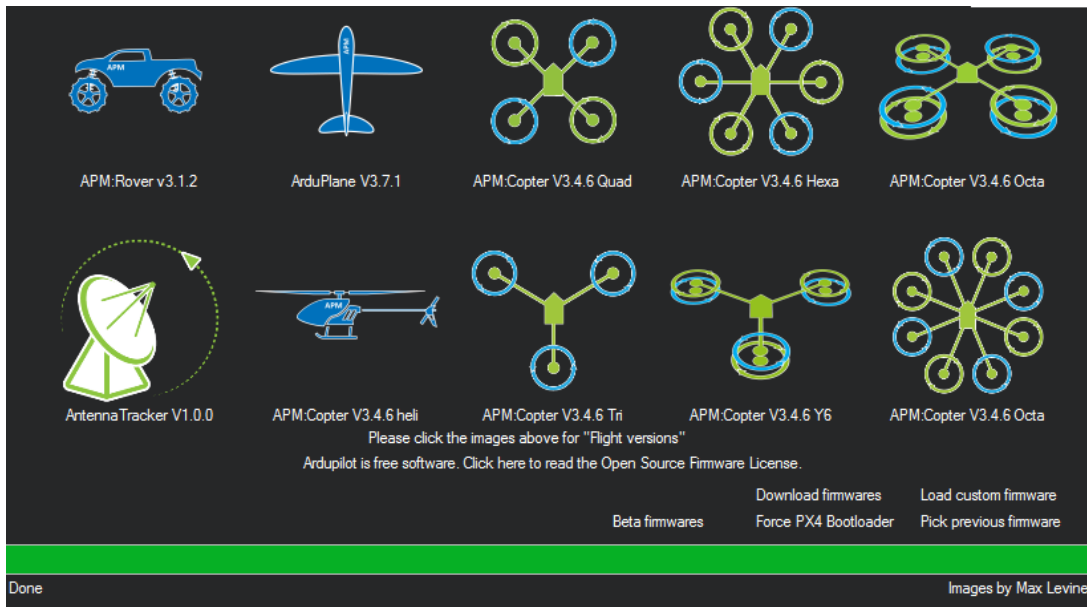
- 3.จากนั้นคลิกที่แถบเมนู INITIAL SETUP



- 4.คลิกที่คำสั่ง Install Firmware จากนั้นเลือกชนิดของอุปกรณ์ โดยในที่นี้จะทำการเลือกเป็น Copter Quad จากนั้นคลิกที่ Yes เพื่อทำการลง Firmware



5.จากนั้น รอให้ลง firmware ให้เรียบร้อยจนสังเกตเห็นคำว่า “Done” ก็เป็นอันเสร็จ

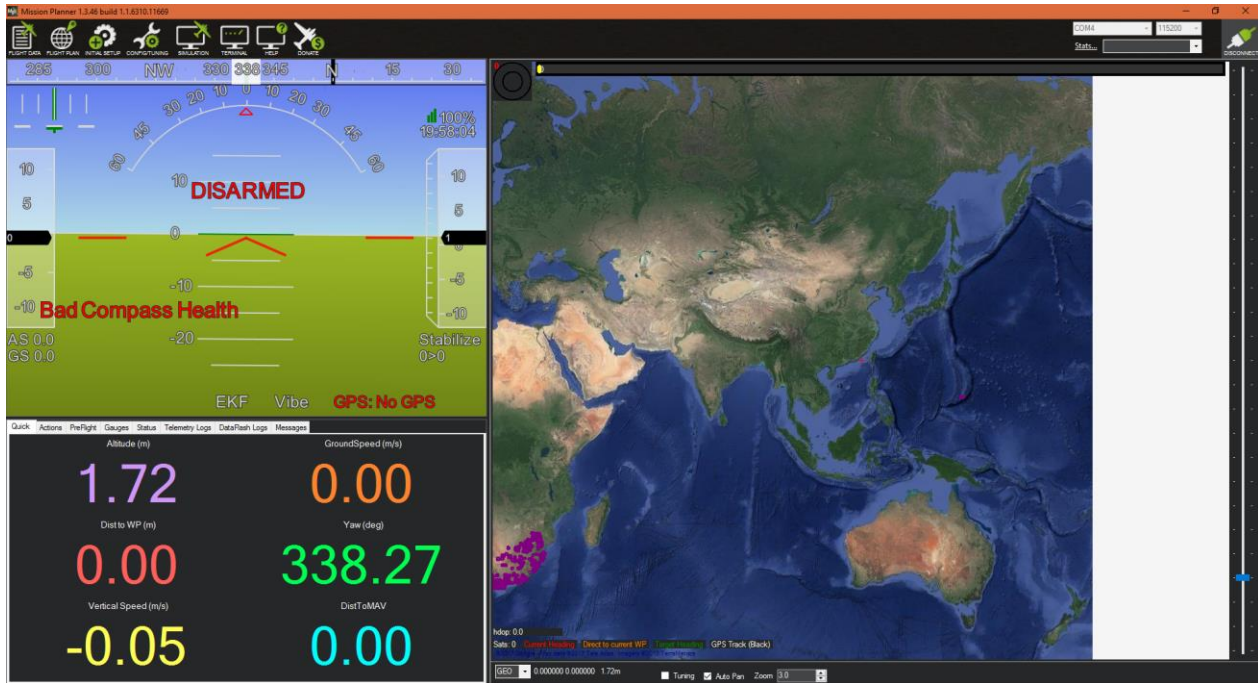


6.จากนั้นคลิกที่แถบเมนู FIGHT DATA ทำการเลือก port ที่เชื่อมต่อจากนั้นคลิกที่ Connect



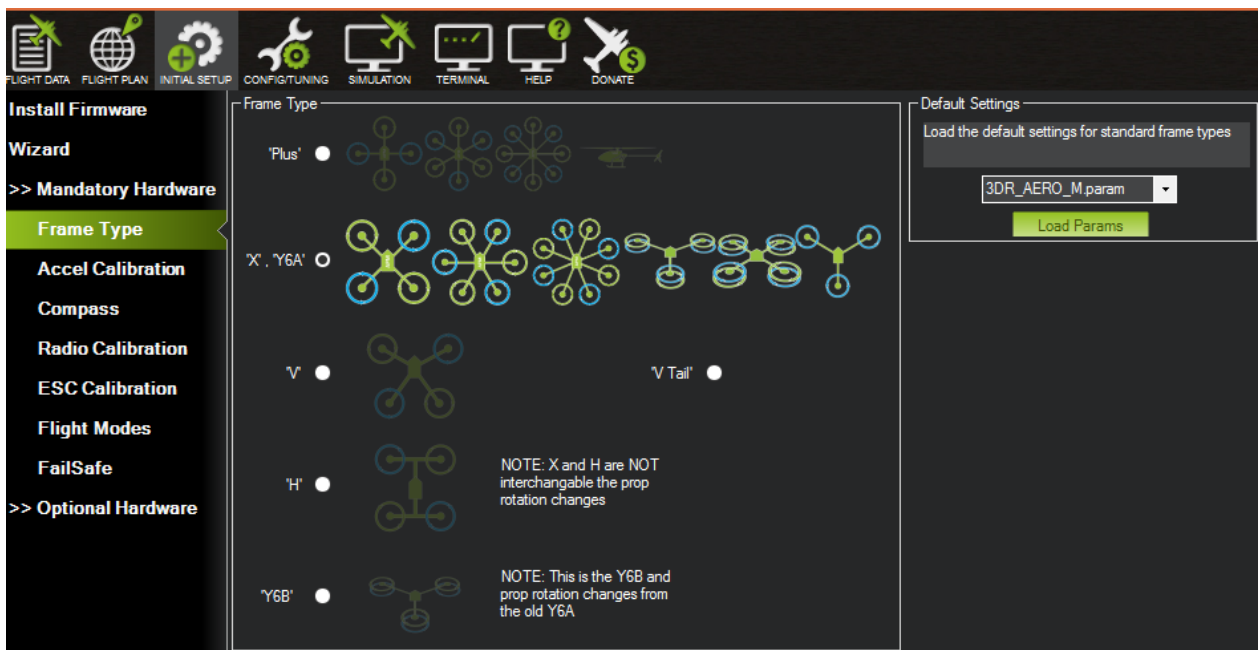
7.จะปรากฏข้อมูลต่าง ๆ ของบอร์ดแสดงในโปรแกรมดังรูป





8. จากนั้นกลับไปเมนู INITIAL SETUP เพื่อทำการตั้งค่าของบอร์ด

9. สามารถตั้งค่าได้ที่เมนู >>Mandatory Hardware เช่น การตั้งค่า Frame Type(ชนิดของเฟรม)



10. ก็เป็นอันเสร็จสำหรับการลงเฟิร์มแวร์ให้กับบอร์ด ArduPilotMaga

## 1.5 การทดลองที่ 5 การทำ Accel calibration ของบอร์ด ArduPilotMega

### จุดประสงค์

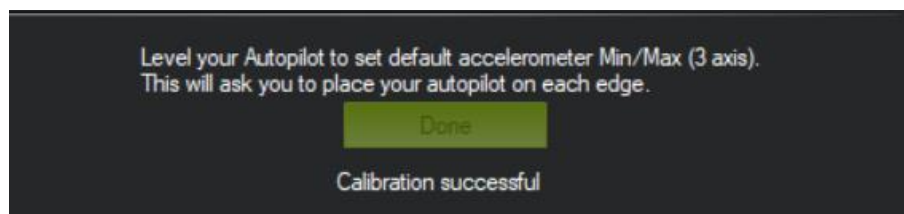
เพื่อทำการ Accel Calibrate ของบอร์ด ArduPilotMega เพื่อเป็นการกำหนดค่าเริ่มต้นของแต่ละแกน

### อุปกรณ์

- 1.บอร์ด ArduPilotMega
- 2.สาย Micro USB
- 3.โปรแกรม MissionPlanner

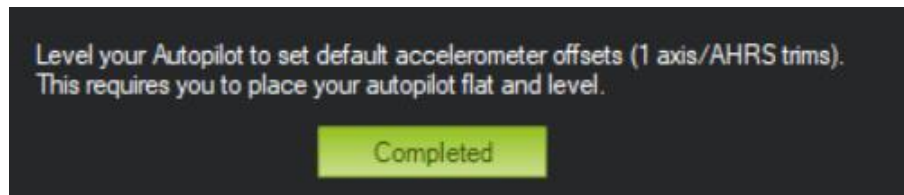
### วิธีการทดลอง

- 1.ทำการเสียบสาย micro USB ตรงตัวบอร์ดและเสียบเข้า Port USB
  - 2.ทำการเชื่อมต่อเข้ากับโปรแกรม Mission Planner
  - 3.ไปที่เมนู INTIAL SETUP -> Mandatory Hardware -> Calibrate Accel
- ในส่วนแรก Calibrate Accel (3-axis) คือ การให้ตัวบอร์ดจำแกนในแต่ละด้านของบอร์ด เช่น front, left, right, down, up และ back
- 4.จากนั้นทำการพลิกบอร์ดตามที่โปรแกรมกำหนดโดยให้หันลูกศรที่ติดอยู่ที่บอร์ดไปตามทิศทางต่าง ๆ



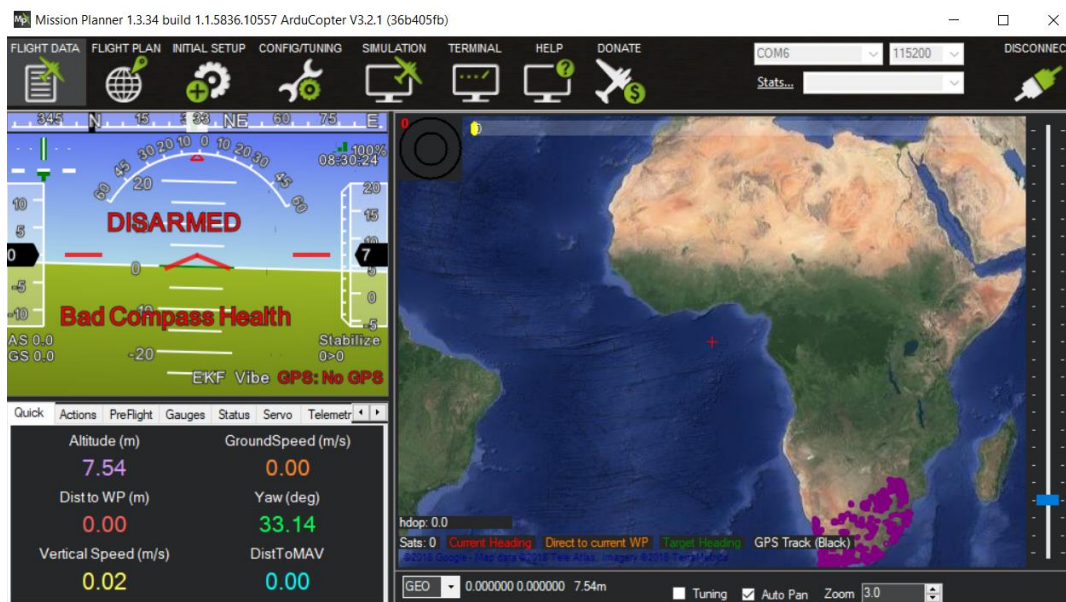
- 5.หากทำการ calibrate สำเร็จแล้วก็จะขึ้นข้อความว่า Calibration successful

- ในส่วนที่สอง Calibrate Level (1-axis) คือ การให้ตัวบอร์ดจำแกน (0,0,0) โดยจะต้องตั้งบอร์ดให้ระนาบกับพื้น



6. หากทำการ calibrate สำเร็จแล้วก็จะขึ้นข้อความว่า Completed

7. จากนั้นกลับมาที่เมนู FLIGHT DATA และทำการ connect เพื่อตรวจสอบความถูกต้องหลังจากการทำการ calibrate



6. จะเห็นว่าหลังจากการทำ calibrate ที่ (0,0,0) แล้วโปรแกรมสามารถอ่านค่าจาก Sensor ได้อย่างถูกต้องและมีความแม่นยำมากขึ้น

roll	0.795522	vz	0	groundspeed2	0	ay	4
pitch	-1.348084	vien	0	groundcourse2	0	az	-1000
yaw	116.8339	altoffsethome	0	satcountB	0	accelsq	1.000552
SSA	0	gpsstatus	1	gpstime	1/1/2513 (	gx	-1
AOA	0	gpshdop	99.99	altd1000	0.00202	gy	1
groundcourse	0	satcount	0	altd100	0.0202	gz	2
lat	0	lat2	0	airspeed	0	gyrosq	2.44949
lng	0	lng2	0	targetairspeed	0	mx	0
alt	2.02	altasl2	0	lowairspeed	False	my	0
altasl	-17	gpsstatus2	0	asratio	0	mz	0
vx	0	gpshdop2	0	groundspeed	0	magfield	0
vy	0	satcount2	0	ax	-33	ax2	0

## 1.6 การทดลองที่ 6 ทำการ Calibrate Compass ของบอร์ด ArduPilotMega

### จุดประสงค์

เพื่อทำการเซตค่าเริ่มต้นให้กับเข็มทิศและจีพีเอสในทิศทางการหมุนด้านต่าง ๆ ของบอร์ด

### อุปกรณ์

- 1.บอร์ด ArduPilotMega
- 2.Module GPS NEO-7N
- 3.สาย Micro USB
- 4.โปรแกรม MissionPlanner

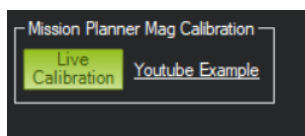
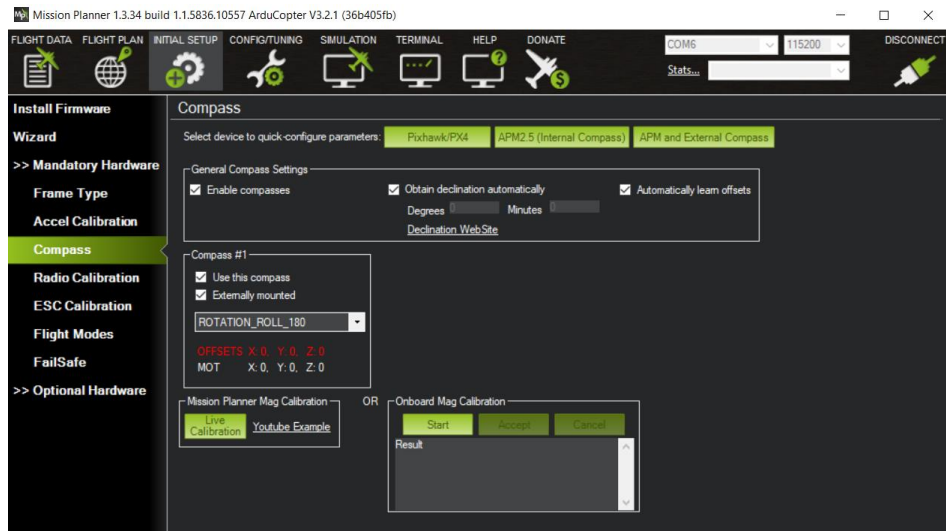
### วิธีการทดลอง

- 1.ทำการเสียบโมดูล GPS ที่มี compass ภายนอกเข้ากับ port I2C และ port GPS
- 2.ปรับให้ลูกศรของบอร์ด ArduPilotMega กับ Module ไปในทิศทางเดียวกัน

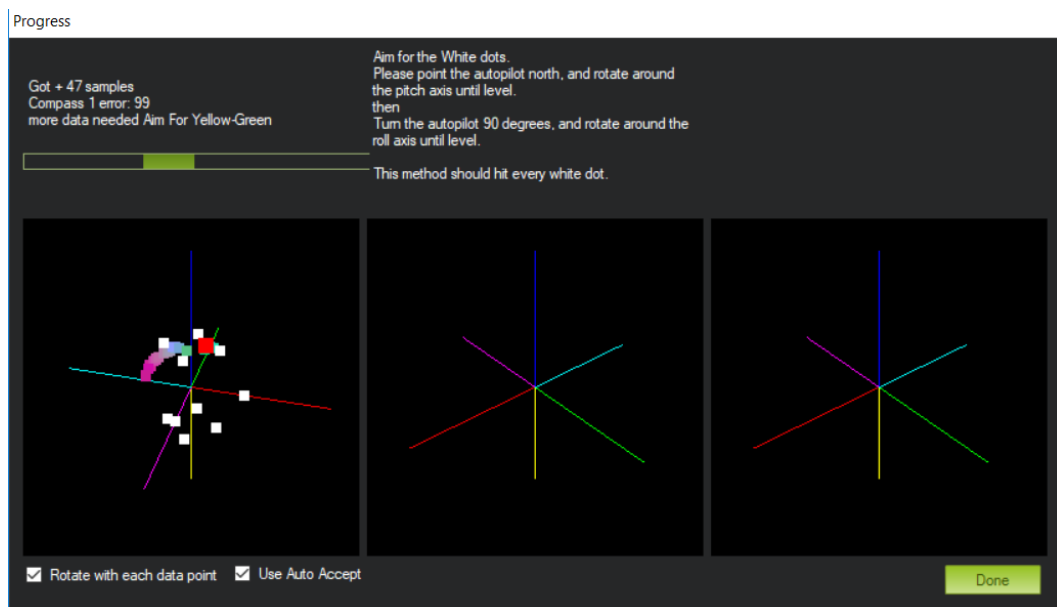


- 3.จากนั้นไปที่เมนู INITIAL SETUP -> Mandatory Hardware -> Compass -> APM and External Compass

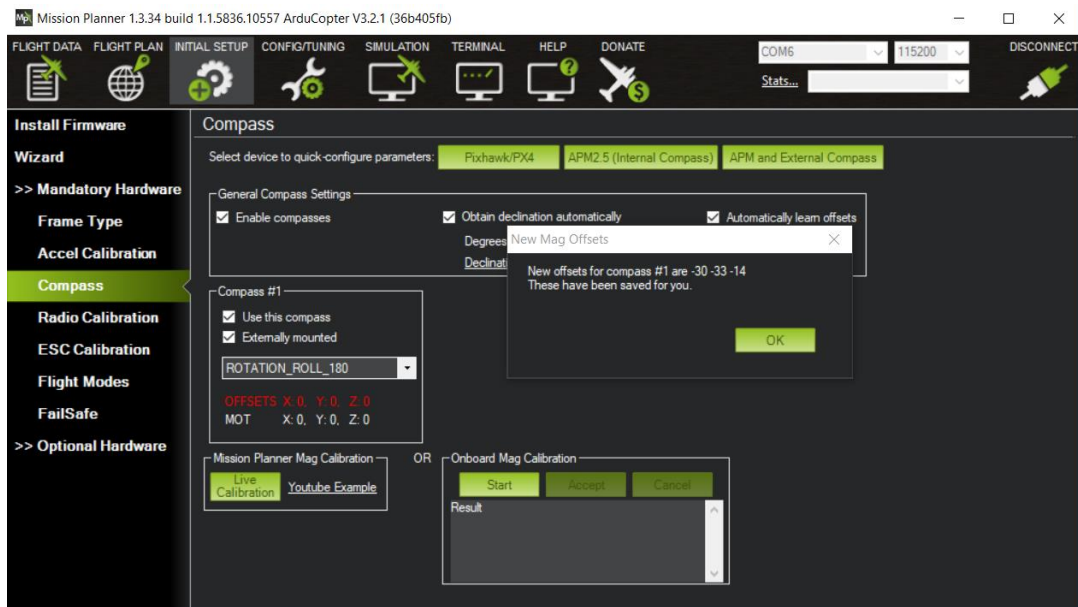




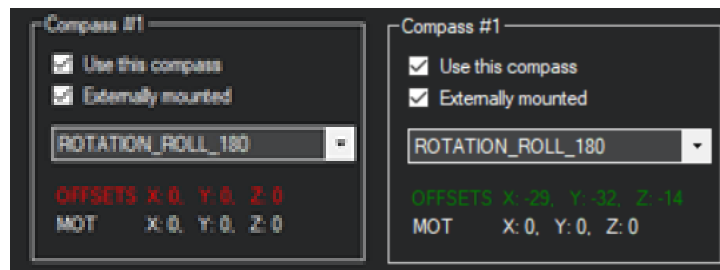
4.จากนั้นทำการกดที่ Live Calibration



5.ให้ทำการจับบอร์ดและหมุนไปในแนวต่าง ๆ เพื่อให้บอร์ดจดจำค่าของแนวและด้านต่าง ๆ ของเข็มทิศ



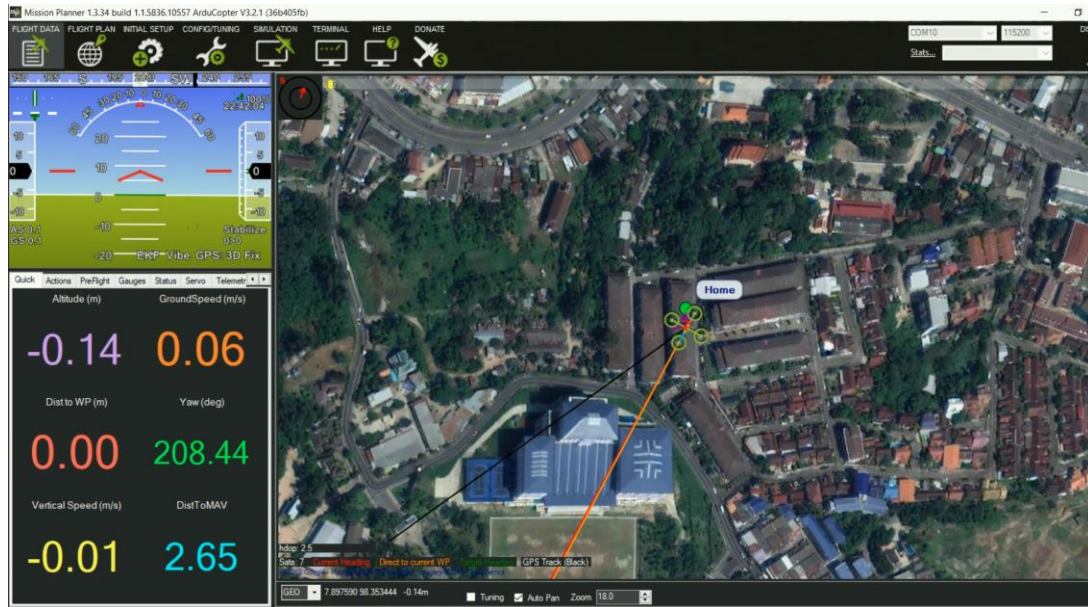
6. เพื่อสำเร็จแล้ว ก็จะมีข้อความขึ้นมาแจ้งเตือนและระบุตำแหน่งของเข็มทิศ



ก่อนและหลังการ Calibrate Compass



7. จะเห็นว่าคำว่า bad compass หายไปเนื่องจากทำการ calibrate compass เรียบร้อยแล้ว และที่แสดง GPS : no fix เนื่องจากอยู่ภายในอาคารจึงไม่สามารถรับสัญญาณจากดาวเทียมได้



8.เมื่อ GPS สามารถสัญญาณจากดาวเทียมได้จะแสดงสถานะ GPS : 3D FIX พร้อมทั้งแสดงตำแหน่งที่อยู่ ณ ปัจจุบันบนแผนที่ และยังสามารถแสดงถึงเขตการป้องกันหรือเขตห้ามทำการบินได้อีกด้วย โดยจะอยู่ใกล้ ๆ กับสนามบินที่กองบินต่าง ๆ เพื่อเป็นการป้องกันจากการบุกรุกจากอากาศยานไร้คนขับหรือโดรน

## 1.7 การทดลองที่ 7 การ Calibrate ESC

### จุดประสงค์

เพื่อทำการเซ็ตค่าเริ่มต้นของช่วงสัญญาณที่ส่งให้ ESC เพื่อใช้ในการควบคุมมอเตอร์เพื่อจะทำให้มอเตอร์สามารถทำงานได้พร้อมกัน

### อุปกรณ์

- 1.Remote Flysky FS-TH9X
- 2.Receiver Flysky R9B
- 3.ESC 30A
- 4.Battery 11.1 V (3 cell)
- 5.Motor Brushless
- 6.สาย Signal

### วิธีการทดลอง

- 1.ทำการดัน Throttle ขึ้นมาสุด
- 2.ทำการเสียบแบตเตอรี่แล้วรอประมาณ 2 วินาที แล้วถอดแบตเตอรี่ออก
- 3.ทำการเสียบแบตเตอรี่ใหม่ โดย ESC จะมีเสียงบี๊บ หรือ บี๊บ บี๊บ
- 4.พอได้ยินเสียงบี๊บ หรือ บี๊บ บี๊บ แล้วดัน Throttle ลงมาสุด ก็จะได้ยินเสียง บี๊บ หรือ บี๊บ บี๊บ อีกครั้ง
5. เป็นอันเสร็จ และสามารถทดสอบได้โดยการดัน Throttle ดูว่ามอเตอร์หมุนทุกตัวหรือไม่



## 1.8 การทดลองที่ 8 ทำการ Calibrate Radio ของบอร์ด ArduPilotMega

### จุดประสงค์

เพื่อทำการเซ็ตค่าเริ่มต้นของสัญญาณ PWM และสูงสุดที่ส่งมาจาก receiver กับบอร์ด ArduPilotMega

### อุปกรณ์

- 1.บอร์ด ArduPilotMega
- 2.Module GPS NEO-7N
- 3.สาย Micro USB
- 4.โปรแกรม MissionPlanner
- 5.Receiver + Remote
- 6.สาย Signal

### วิธีการทดลอง

- 1.ทำการเสียบสายสัญญาณระหว่าง Receiver กับ board ArduPilotMega
- 2.ไปที่เมนู INTIAL SETUP -> Mandatory Hardware -> Radio Calibration



การ calibrate วิทยุเป็นการทำเพื่อให้ตัวโดรนทราบว่าช่วงความกว้างของสัญญาณวิทยุมีค่าเป็นเท่าใด โดยค่าที่โปรแกรมบันทึกลงไปมี 5 ช่องสัญญาณคือ ช่อง Roll, ช่อง Pitch, ช่อง Yaw , ช่อง Throttle และช่อง AUX ใช้สำหรับการปรับโหมดในการควบคุมของโดรน โดยเราสามารถโยกคันโยกที่ตัววิทยุเพื่อดูค่าต่าง ๆ จากโปรแกรมได้

## 1.9 การทดลองที่ 9 ทำการตั้งค่าโหมดการบินของบอร์ด ArduPilotMega

### จุดประสงค์

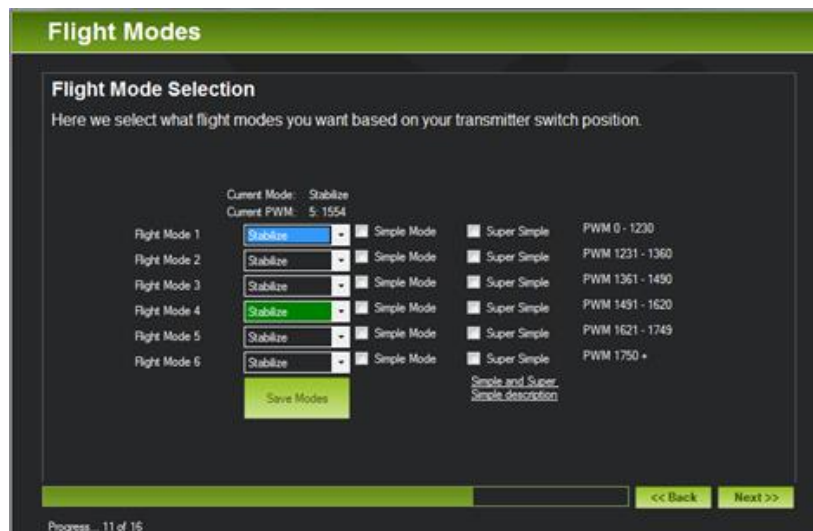
เพื่อทำการเช็คค่าเริ่มต้นของสัญญาณ PWM และสูงสุดที่ส่งมาจาก receiver กับบอร์ด ArduPilotMega

### อุปกรณ์

- 1.บอร์ด ArduPilotMega
- 2.Module GPS NEO-7N
- 3.สาย Micro USB
- 4.โปรแกรม MissionPlanner
- 5.Receiver + Remote
- 6.สาย Signal

### วิธีการทดลอง

- 1.ทำการเสียบสายสัญญาณระหว่าง Receiver กับ board ArduPilotMega
- 2.ไปที่เมนู INITIAL SETUP -> Mandatory Hardware -> flight mode



โหมด Stabilize จะเป็นโหมดที่ทำให้ตัวโดรนเข้าสู่การที่โดรนมีเสถียรภาพสูง

โหมด Loiter จะเป็นการลือคตำแหน่งโดรนโดยพิกัด GPS

โหมด RTL จะเป็นการใช้งาน Return to home โดยตั้งพิกัด GPS เพื่อจอด

โหมด LAND จะเป็นการใช้งานการ Landing ของตัวโดรน

รูปดังกล่าวเป็นหน้าโปรแกรมที่แสดงสถานะของโดรนตอนปัจจุบันหลังจากทำการ Setup เบื้องต้นเพื่อให้โดรนสามารถขึ้นบินได้ โดยโปรแกรมจะแสดงระดับของตัวโดรน ความเร็ว องศาการหมุน โดยเมื่อทำการ config บอร์ด APM สำเร็จแล้วก็จะสามารถทำการ ARM ได้ โดยทำการเลื่อนตำแหน่งคันบังคับที่รีโมทลง จะขึ้นสถานะ ARM ซึ่งเป็นการบ่งบอกว่าโดรนของเราสามารถบินได้แล้ว

## 1.10 การทดลองที่ 10 การทดสอบการบินของโดรนโดยไม่มีการจูนค่า

### จุดประสงค์

เพื่อทำการเช็คการบินของตัวโดรนก่อนที่จะทำการปรับจูนค่า

### อุปกรณ์

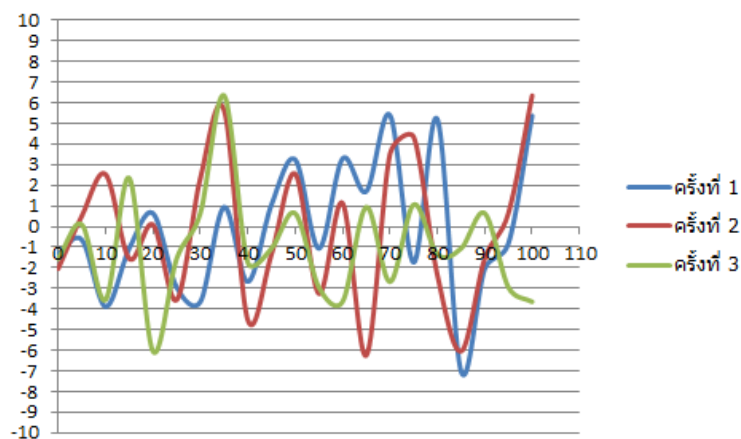
- 1.ตัวโดรนที่ทำการประกอบเสร็จเรียบร้อยแล้ว

### วิธีการทดลอง

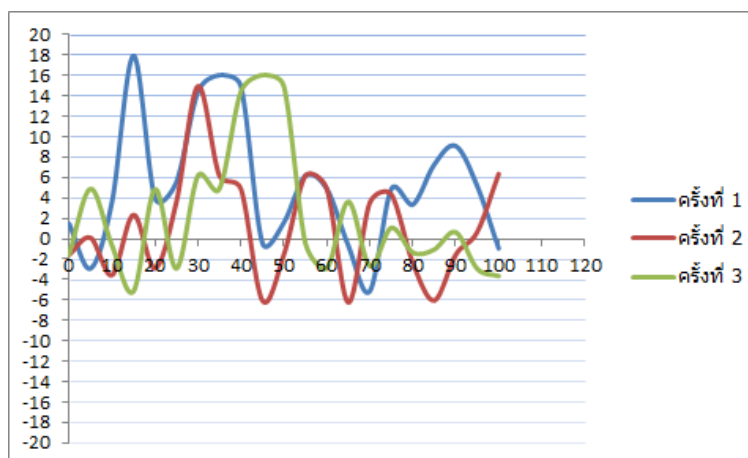
- 1.ทำการตรวจเช็คอุปกรณ์ที่ตัวโดรนให้เรียบร้อยแล้ว
- 2.ทำการ ARM เพื่อเป็นการปลดล๊อคระบบความปลอดภัยของโดรนเพื่อป้องกันการเกิดอุบัติเหตุ
- 3.เริ่มทำการบินโดยเก็บค่า Roll กับ Pitch ผ่านตัว Telemetry

### ผลการทดลอง

#### ที่แกน Roll



#### ที่แกน Pitch



## 1.11 การทดลองที่ 11 การทดสอบการบินของโดรนโดยไม่มีแรงปะทะจากแรงลม (จูนค่าแล้ว)

### จุดประสงค์

เพื่อทำการเช็คการบินของตัวโดรนหลังจากการปรับจูนค่าเรียบร้อยแล้ว(ไม่มีแรงลมปะทะ)

### อุปกรณ์

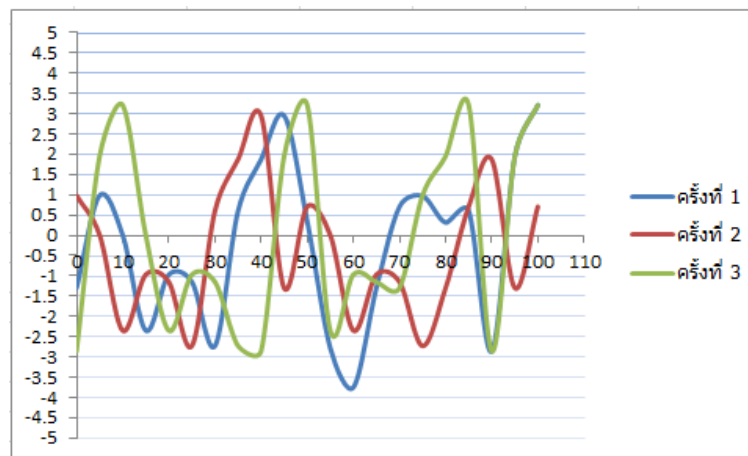
- 1.ตัวโดรนที่ทำการประกอบเสร็จเรียบร้อยแล้ว

### วิธีการทดลอง

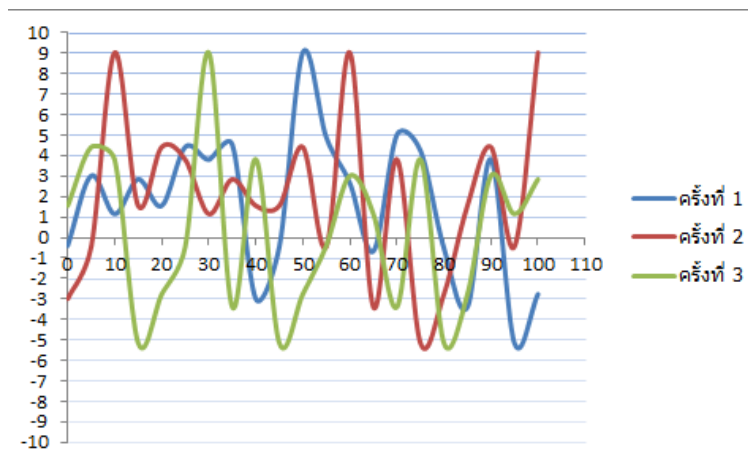
- 1.ทำการตรวจเช็คอุปกรณ์ที่ตัวโดรนให้เรียบร้อยแล้ว
- 2.ทำการ ARM เพื่อเป็นการปลดล๊อคระบบความปลอดภัยของโดรนเพื่อป้องกันการเกิดอุบัติเหตุ
- 3.เริ่มทำการบินโดยเก็บค่า Roll กับ Pitch ผ่านตัว Telemetry

### ผลการทดลอง

#### ที่แกน Roll



#### ที่แกน Pitch



## การทดลองที่ 12 การทดสอบการบินของโดรนโดยมีแรงลมปะทะจากลมของพัดลม

### จุดประสงค์

เพื่อทำการเช็คการบินของตัวโดรนหลังจากการปรับจูนค่าเรียบร้อยแล้ว โดยมีการปะทะจากแรงลมของพัดลม

### อุปกรณ์

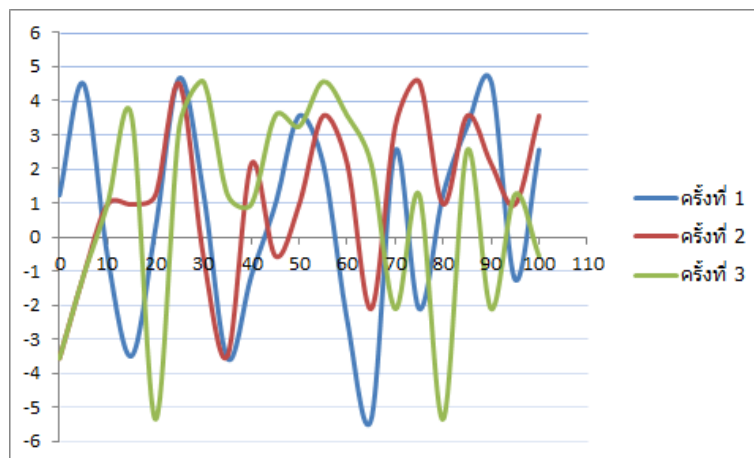
- 1.ตัวโดรนที่ทำการประกอบเสร็จเรียบร้อยแล้ว
- 2.พัดลมขนาด 12 นิ้ว

### วิธีการทดลอง

- 1.ทำการตรวจเช็คอุปกรณ์ที่ตัวโดรนให้เรียบร้อยแล้ว
- 2.ทำการ ARM เพื่อเป็นการปลดล็อคระบบความปลอดภัยของโดรนเพื่อป้องกันการเกิดอุบัติเหตุ
- 3.เริ่มทำการบินโดยและทำการเพิ่มแรงลมเข้ามากระทบกับตัวโดรนทั้งด้านบนใบพัดและด้านใต้ใบพัด
- 4.ทำการเก็บค่า Roll กับ Pitch ผ่านตัว Telemetry

### ผลการทดลอง

#### ที่แกน Roll



#### ที่แกน Pitch

