By T Maker

ESP32 LEARNING KIT

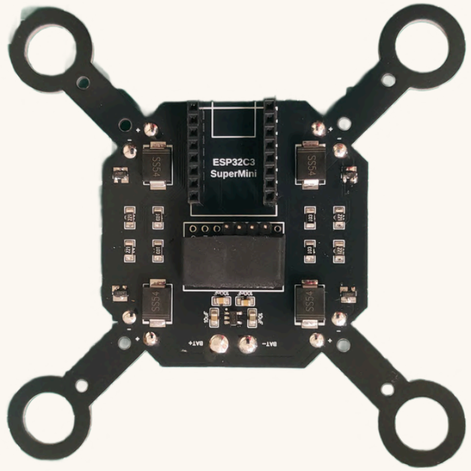
DragonFly - Spirit

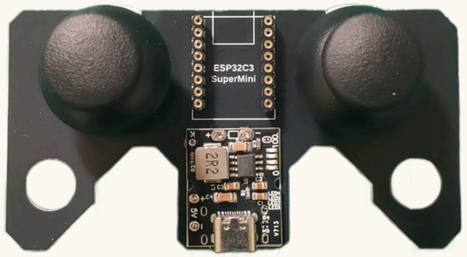
Agenda

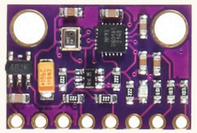
overview

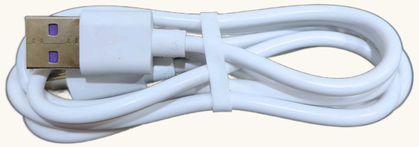
- Rover Robot Learning Kit
- Quadcopter Learning Kit

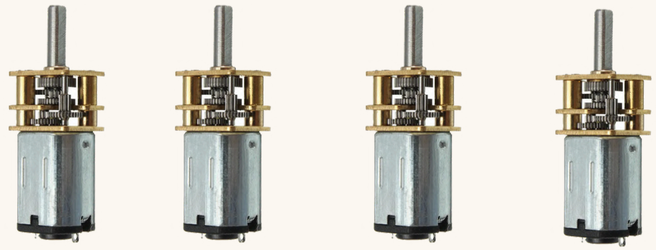
ALL COMPONENTS









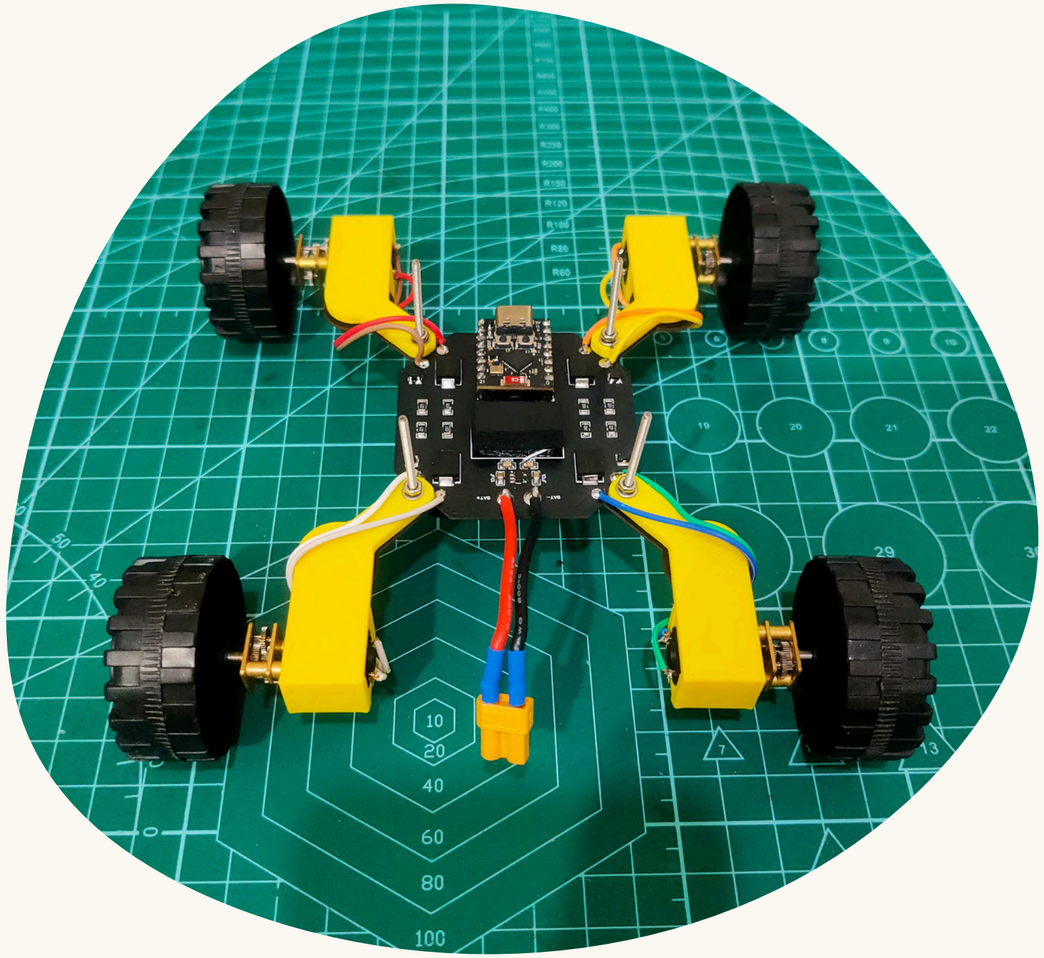


You need to get it yourself

X2

ESP32 C3 Supermini

Rover Robot Learning Kit

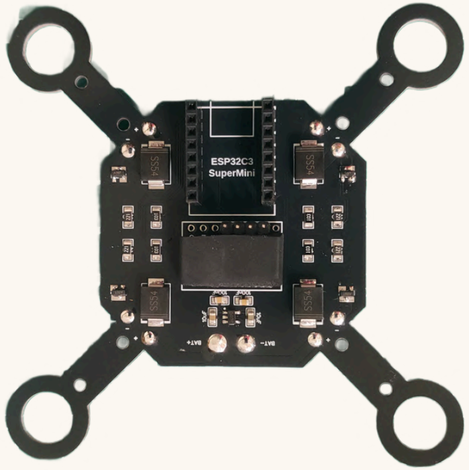


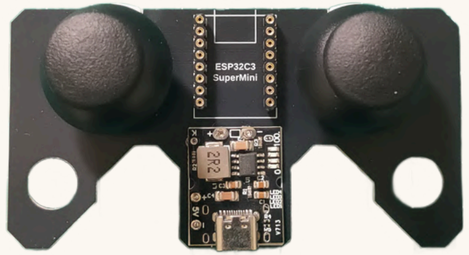
Agenda

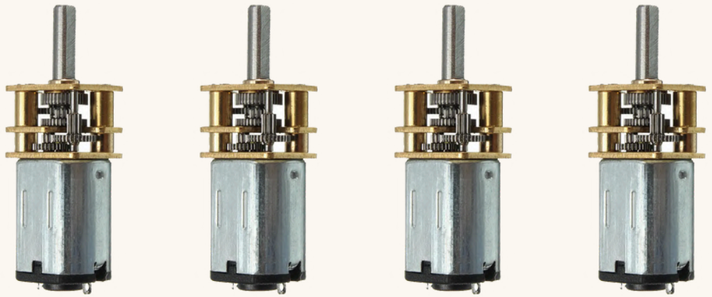
Rover Robot

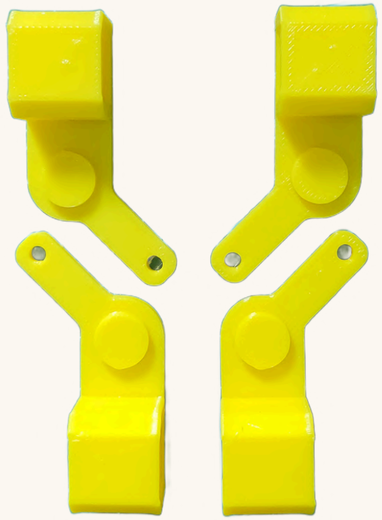
- Component
- Assembly
- Coding
- User manual

ROVER ROBOT COMPONENTS







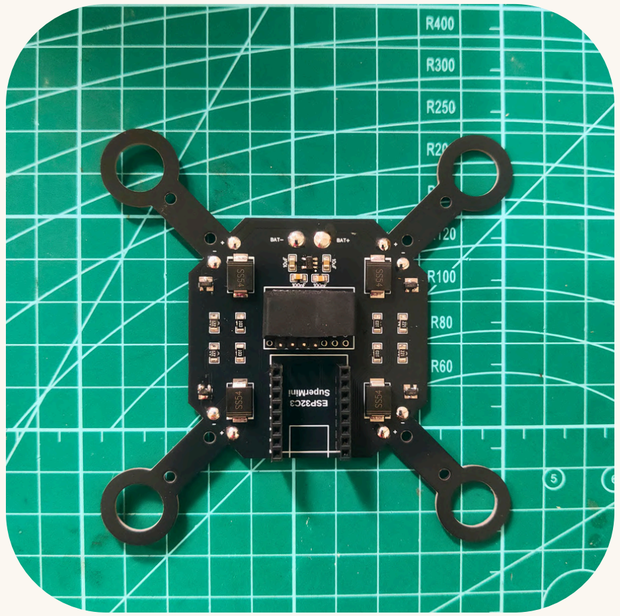


You need to get it yourself

X2

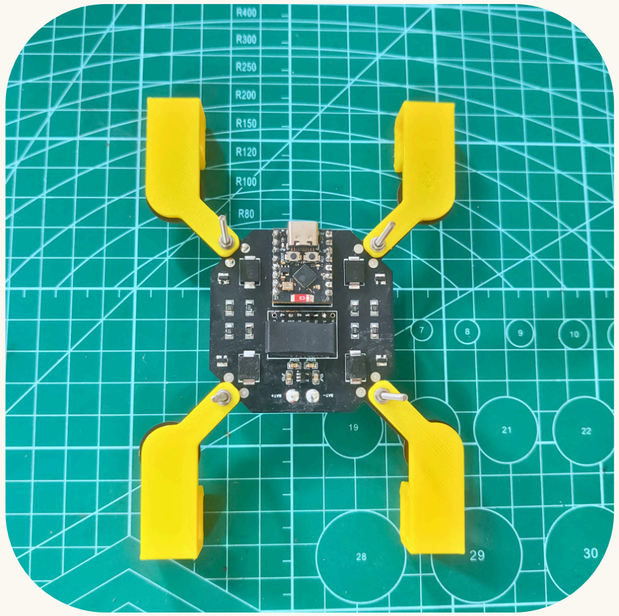
ASSEMBLY

1

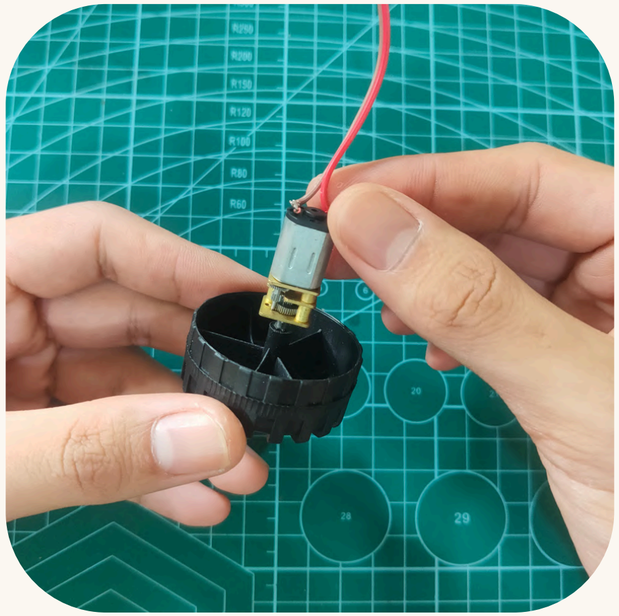


Start

2

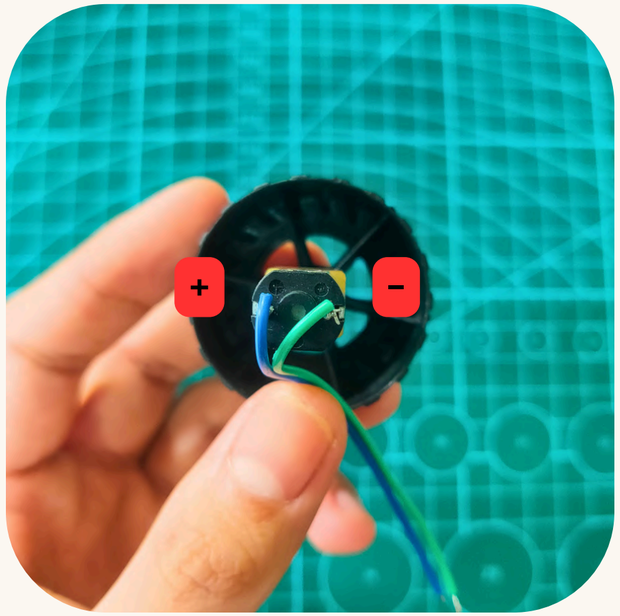


3.1



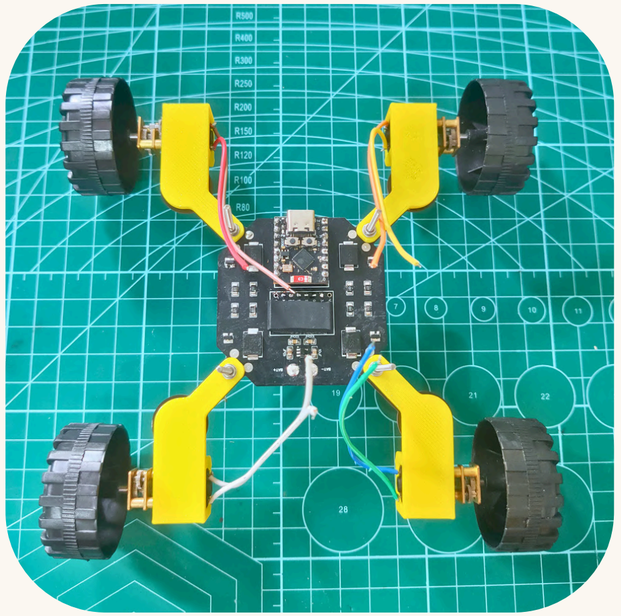
ASSEMBLY

3.2

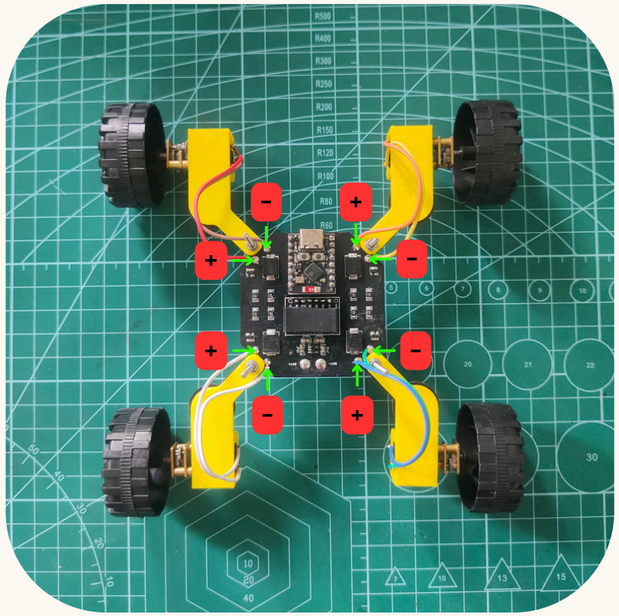


Remember motors pole

4



5

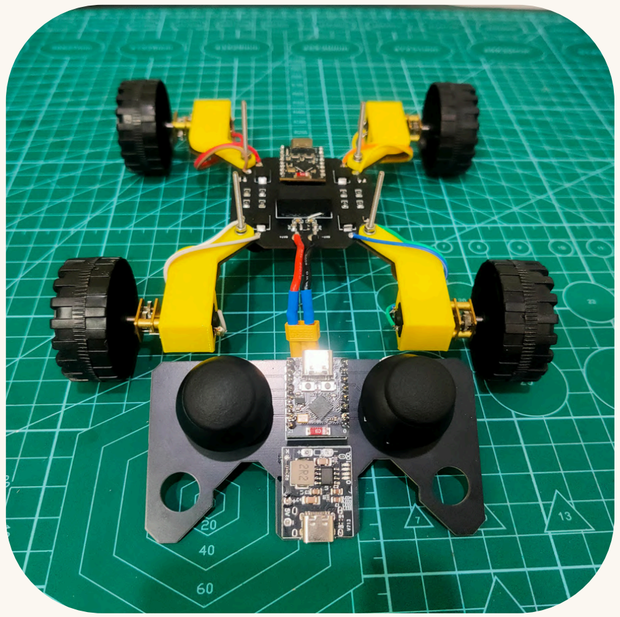


Connect the poles of motors to each pad

ASSEMBLY

6

7



I'm ready to.... ^_^

Rover Robot

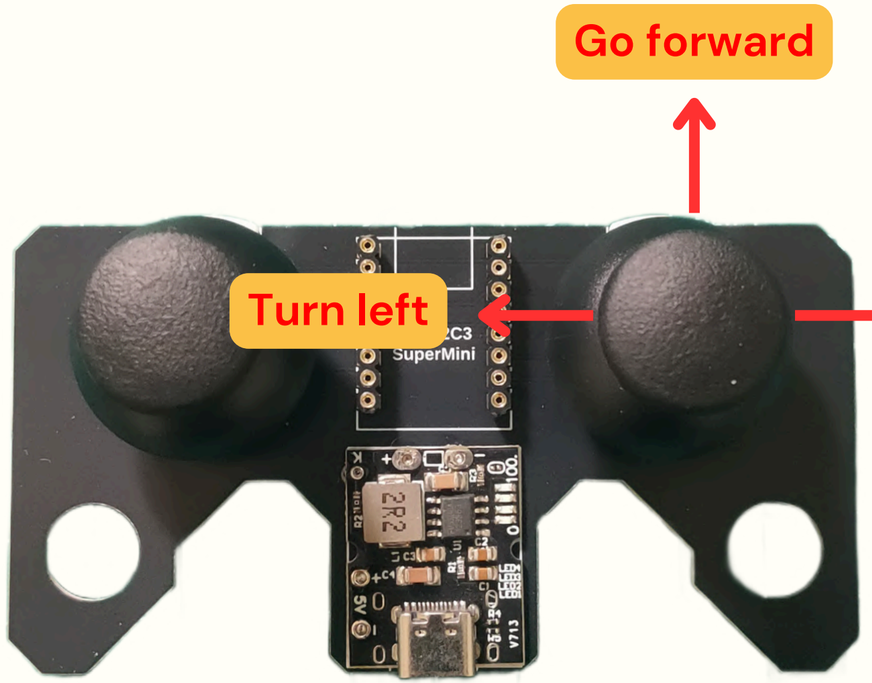
Coding

ARDUINO IDE

```
// ****************************

#include <WiFi.h>
#include <esp_now.h>

// ****************************

// Motors
#define MA 5
#define MB 3
#define MC 1
#define MD 7
#define LEDC_TIMER_10_BIT 10
#define LEDC_BASE_FREQ 20000
```

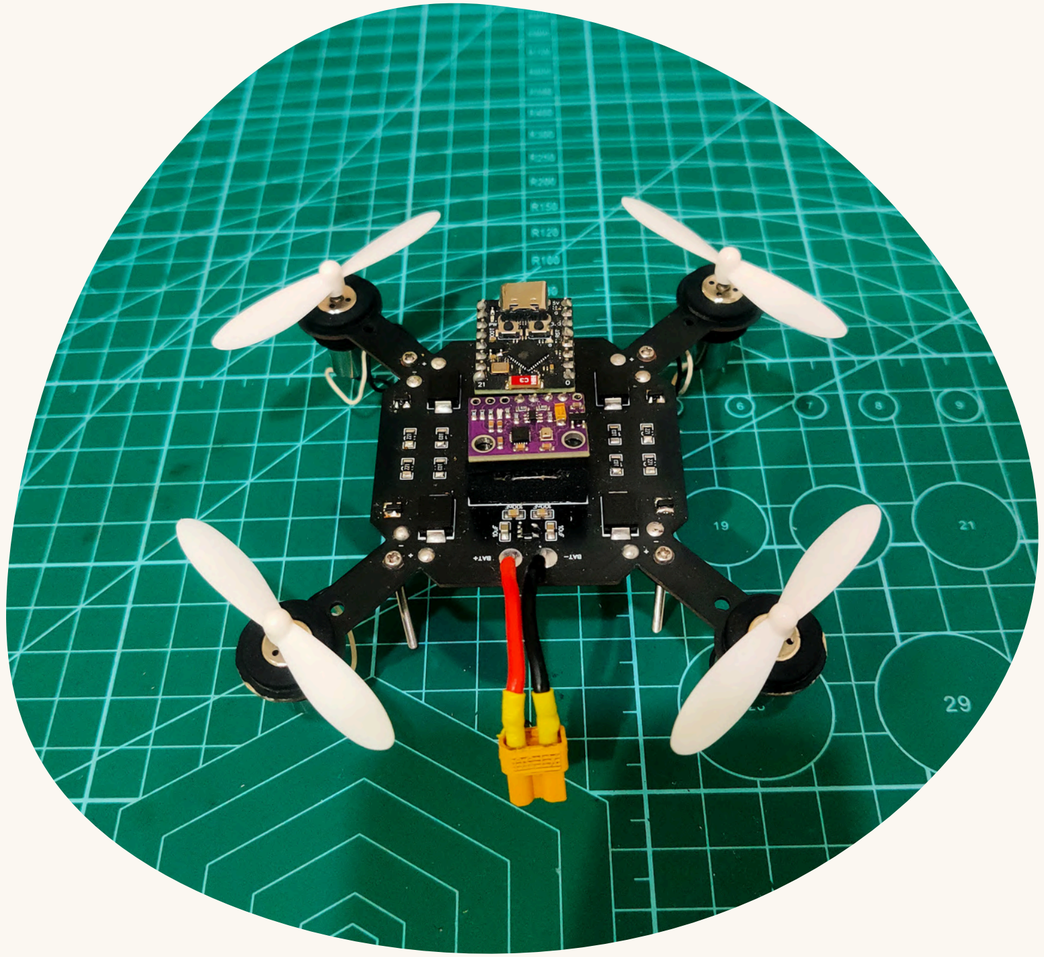
User manual

CONTROLLING



Back side : single press ==> power on | double press ==> power off

Quadcopter Learning Kit



Agenda

1. Introduction

- What is a UAV ?
- Quadcopter
- Ready-to-Fly (RTF) Drone
- DIY Drone

Agenda

2. Principle

- Drone Dynamics
- Comunication
- Flight Controller

Agenda

DragonFly-Spirit-Quadcopter

- Component
- Assembly
- Coding
- User manual

What is a UAV ?

A UAV stands for Unmanned Aerial Vehicle. It's essentially an aircraft that flies without a human pilot on board. UAVs are controlled either remotely by a person (via radio signals, computers, or controllers) or autonomously using pre-programmed flight plans and onboard sensors.

Photo : wikipedia.org/wiki/Unmanned_aerial_vehicle

Type of UAV

Fixed wing rotor

Photo : U.S. Air Force website

Single rotor

Photo : velos-rotors.com

Bi rotor

Photo : www.d-botix.com

Type of UAV

Tricopter

Photo : makezine.com

Quadcopter

Photo : www.t-drones.com

Hexacopter

Photo : www.viewprouav.com

Type of UAV

Hybrid VTOL (Vertical Take-Off and Landing)

Photo : www.unmannedsystemstechnology.com

Quadcopter

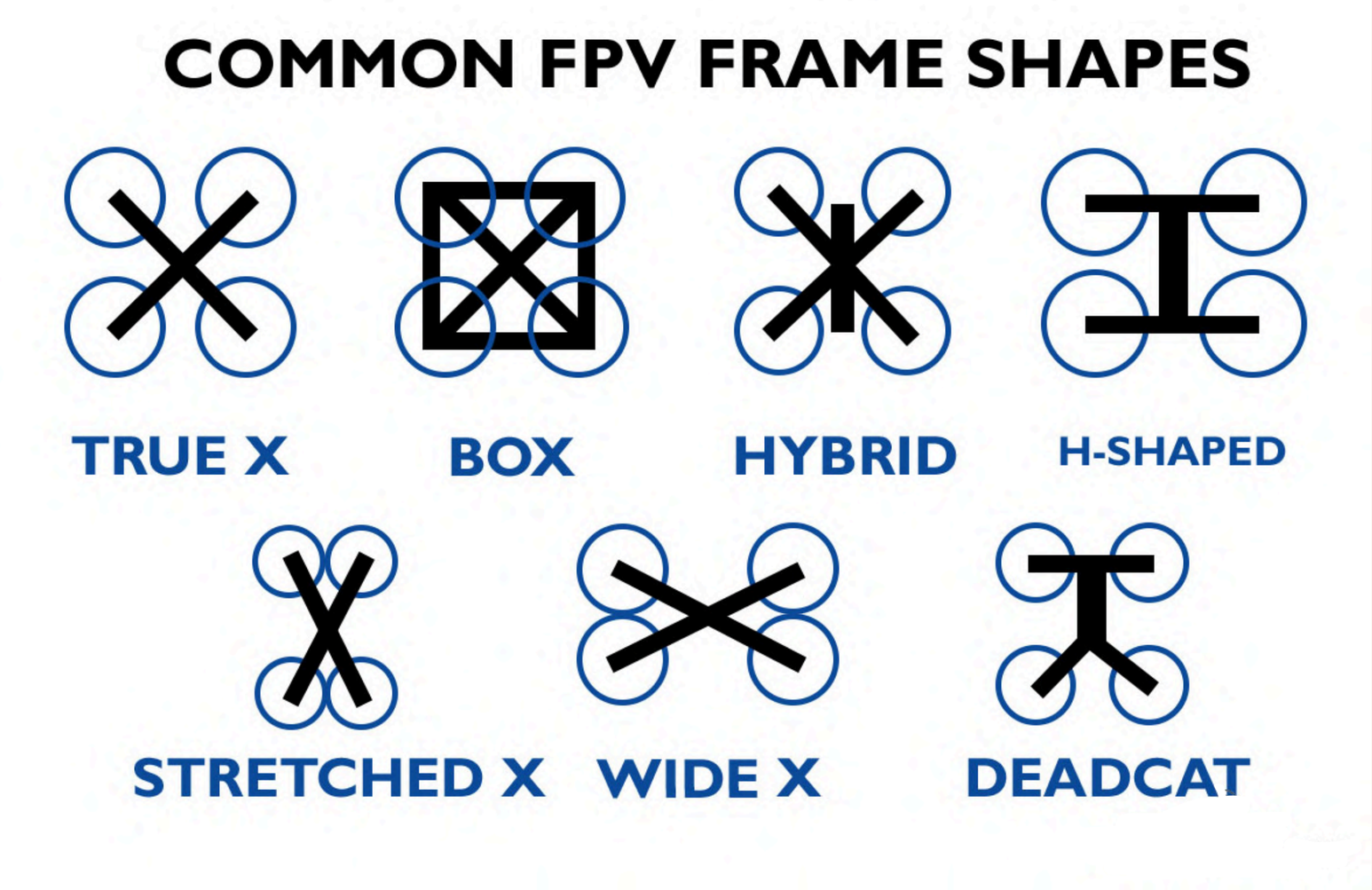


Photo : www.fpv24.com

Commercial grade

Ready-to-Fly (RTF) Drone

Toys grade

DIY Drone

Racing

Photo : drone-gigs.com

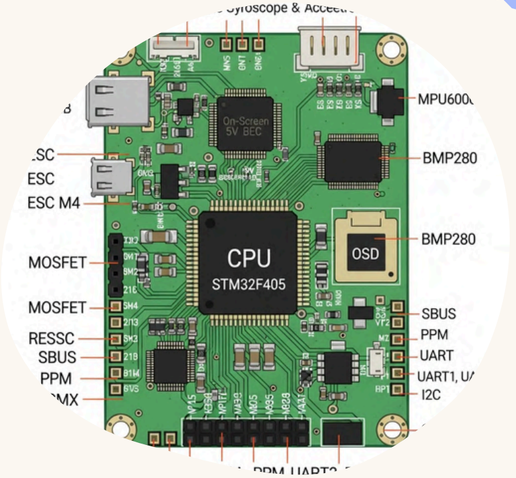
Maker

Photo : makezine.com

PRINCIPLE

Drone Dynamics

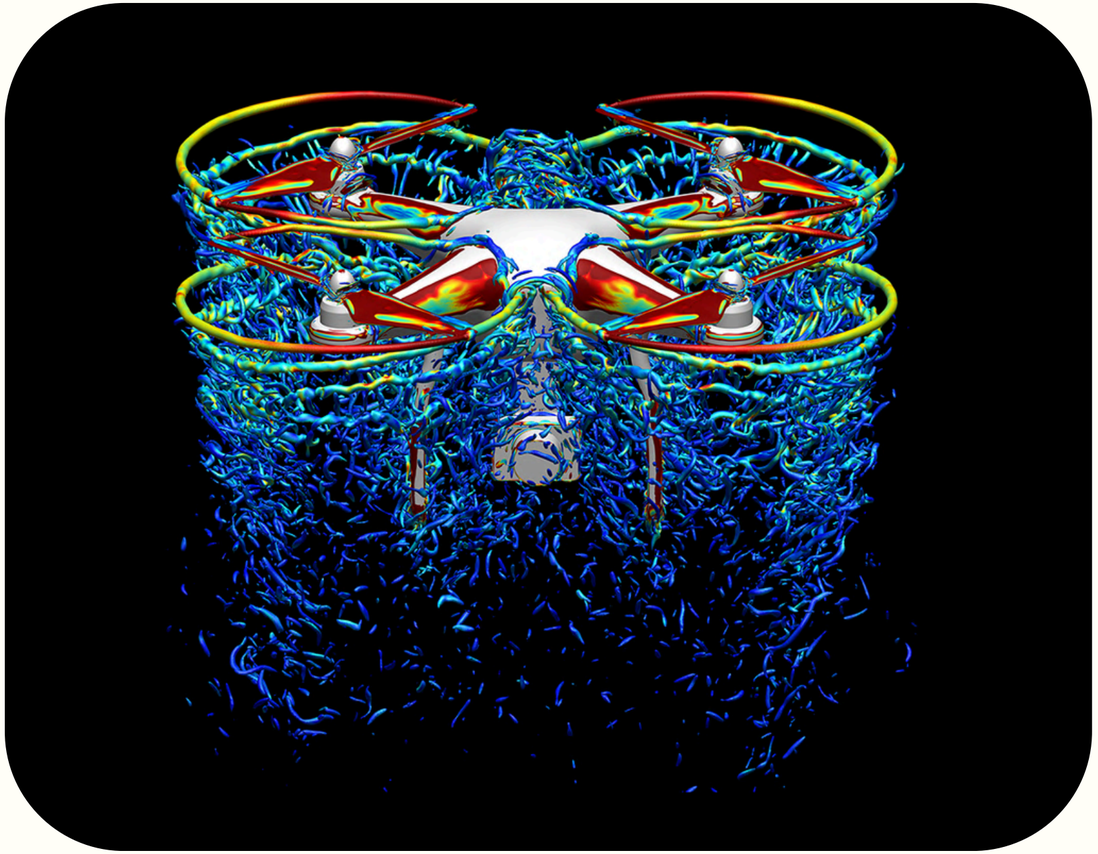
Communication



Flight controller

Drone Dynamics

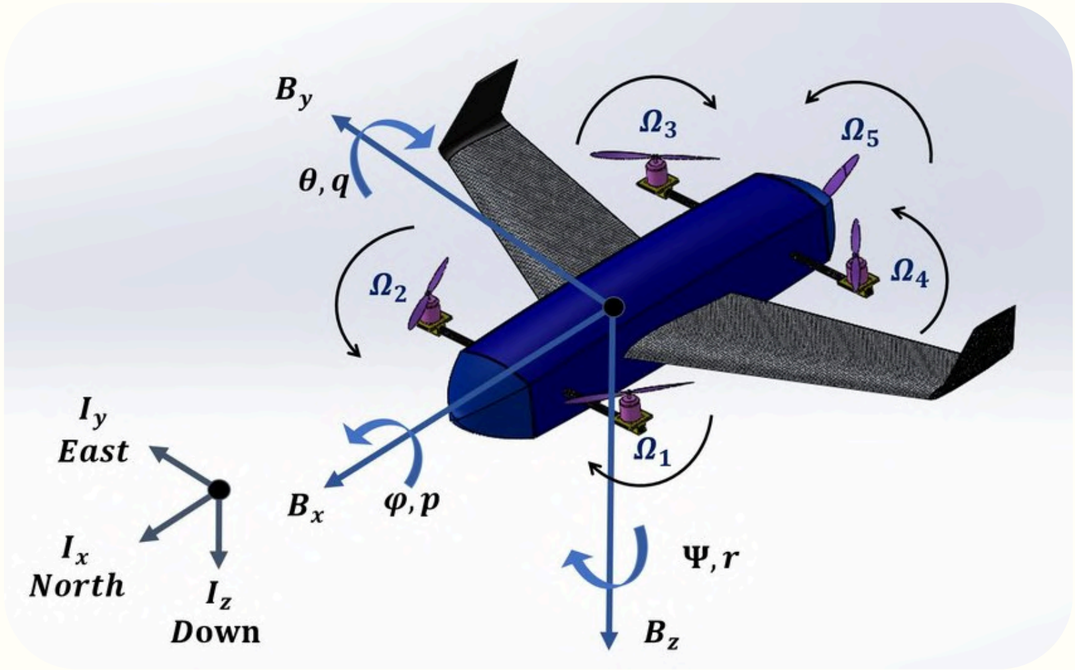
1. Frames of Reference
2. Degrees of Freedom (DOF)
3. Forces and Torques
4. Equations of Motion
5. Motor-Propeller



FRAMES OF REFERENCE

1. Inertial/Fixed Frame
2. Body Frame

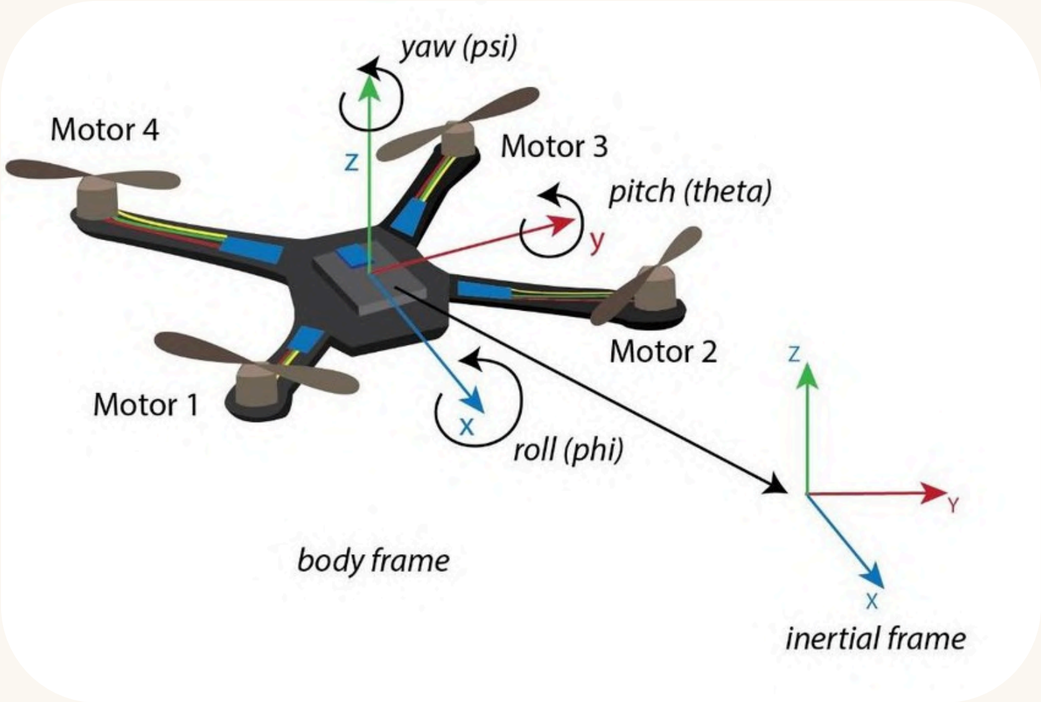
Inertial/Fixed Frame



Origin: launch location (on the Earth's surface).
Axes: fixed relative to Earth at that point in time (north-east-down, or some ENU/NED convention).
Use: good for describing trajectories in a "global" sense.

Photo : www.researchgate.net

Body Frame



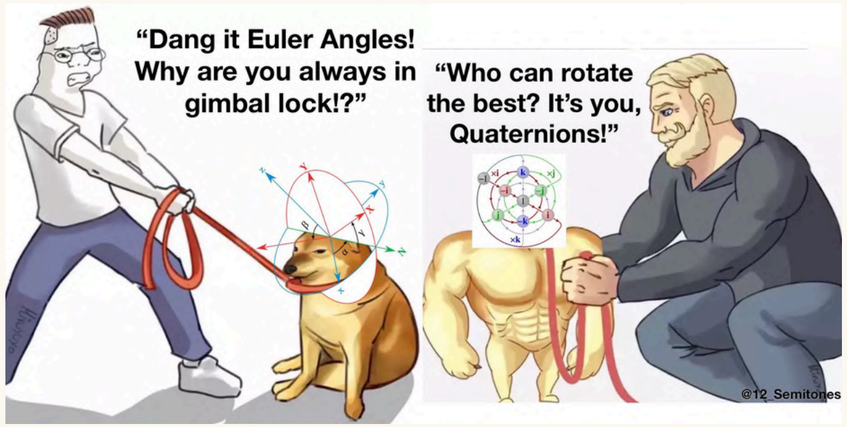
Origin: the vehicle's center of gravity (CoG).
Axes: aligned with the actual vehicle geometry – typically x forward, y right, z down (aircraft convention) or z up (spacecraft sometimes).
Use: sensors (IMU, gyros, accelerometers) and control forces naturally live here.

Photo : www.researchgate.net

- Rotation matrix

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{R}_\phi \mathbf{R}_\theta \mathbf{R}_\psi \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ -\cos\psi\sin\psi + \sin\phi\sin\theta\cos\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & \sin\phi\cos\theta \\ \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi & \cos\phi\cos\theta \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

- Quaternion



$q_1 = [w1\ x1\ y1\ z1]$ $\quad q_2 = [w2\ x2\ y2\ z2]$

$q_1 \otimes q_2 = [q_1q_2w\ \ q_1q_2x\ \ q_1q_2y\ \ q_1q_2z]$

$q_1q_2w = (w1w2 - x1x2 - y1y2 - z1z2)$

$q_1q_2x = (w1x2 + x1w2 + y1z2 - z1y2)$

$q_1q_2y = (w1y2 - x1z2 + y1w2 + z1x2)$

$q_1q_2z = (w1z2 + x1y2 - y1x2 + z1x2)$

Photo : ewpratten.com, www.autonomousrobotslab.com, ece.montana.edu

Degrees of Freedom (DOF)

6 DOF

- Translational : x, y, z
- Rotational : roll (φ), pitch (Θ), yaw (ψ)

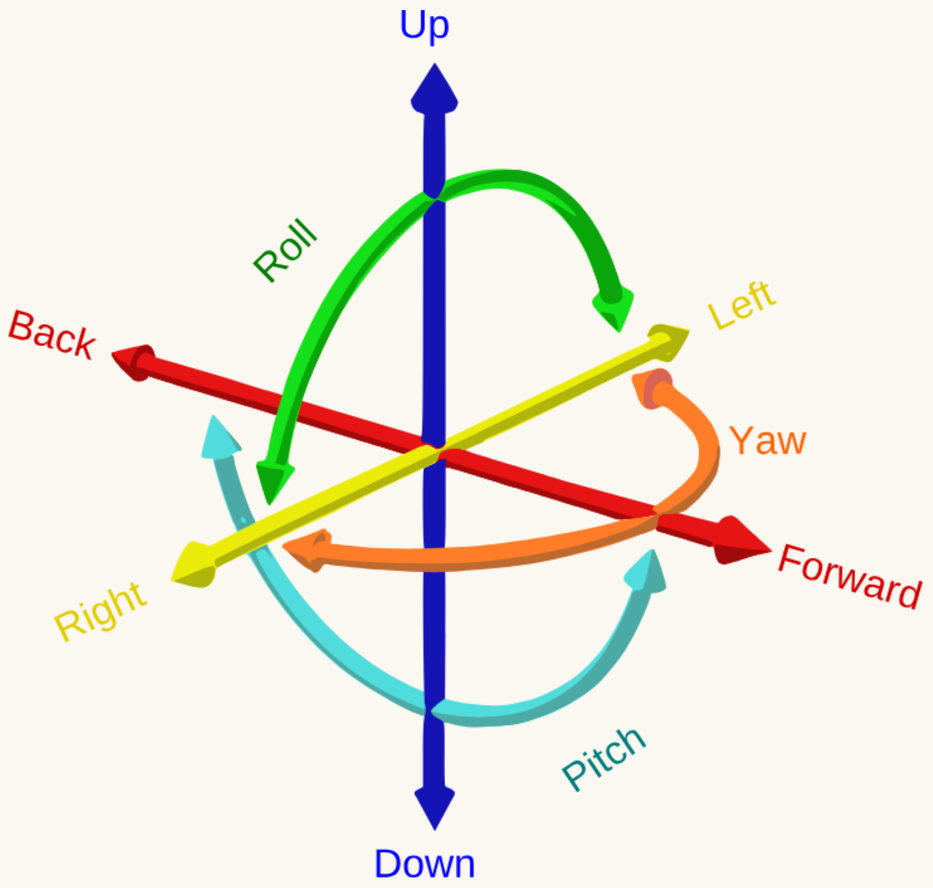


Photo : en.wikipedia.org

FORCE & TORQUE

GRAVITY

WEIGHT

TORQUE

ROLL , PTICH , YAW

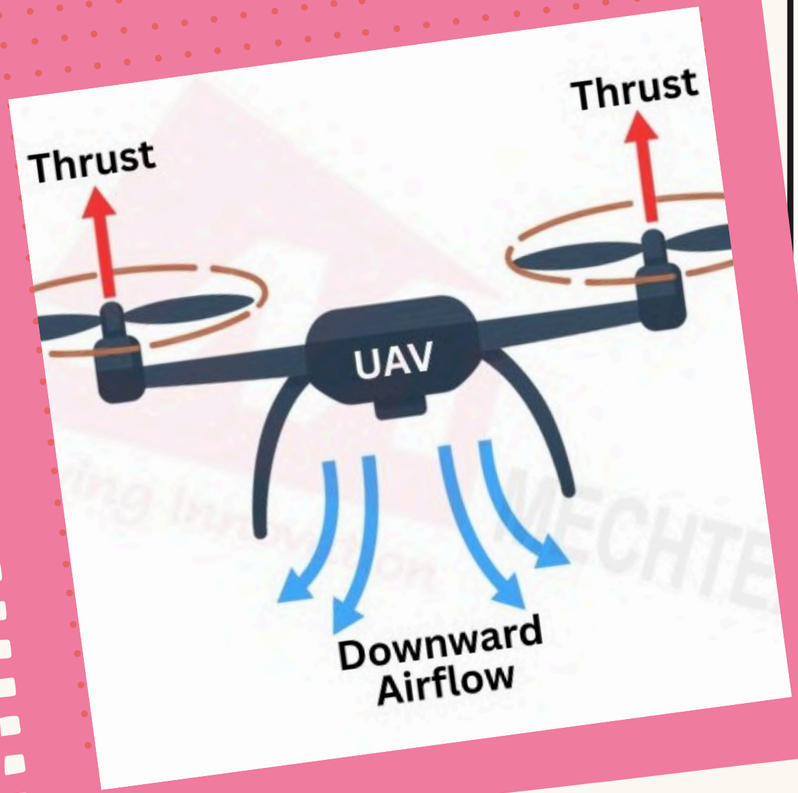
THRUST

LIFT

DRAG

AERODYNAMIC DRAG

THRUST



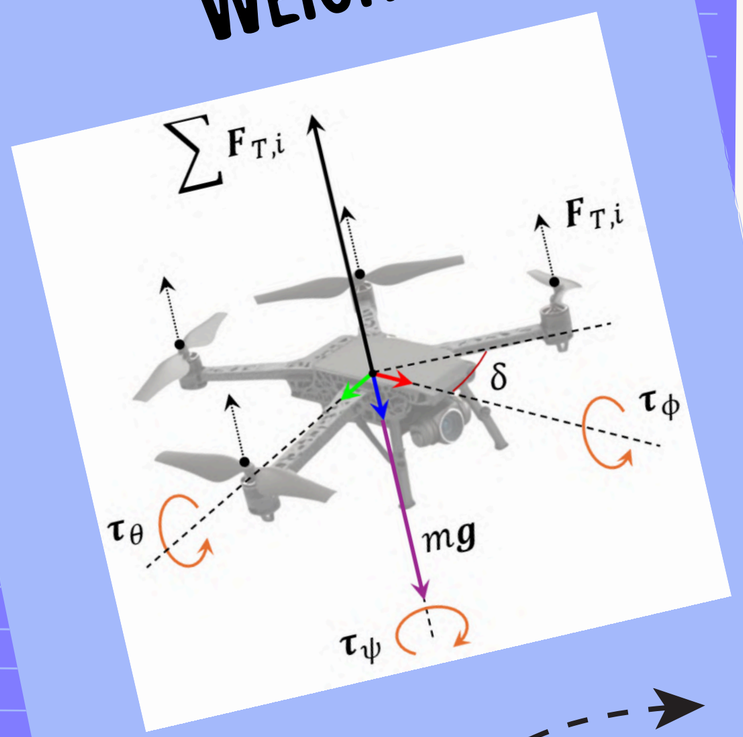
LIFT FORCE

$$L = T \cdot \cos\phi \cdot \cos\theta$$

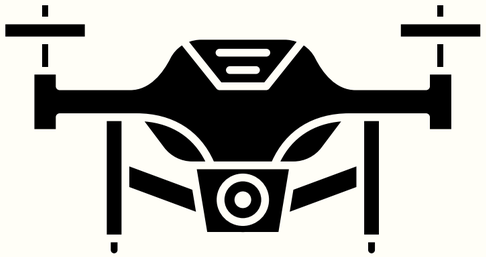
Photo : https://mechtex.com/blog/drone-motor-thrust , https://robotics.stackexchange.com/

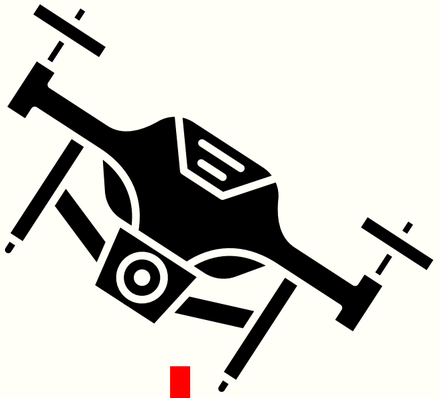
GRAVITY

WEIGHT



$$W = m \cdot g$$





W

W

Photo : https://cfdflowengineering

DRAG FORCE

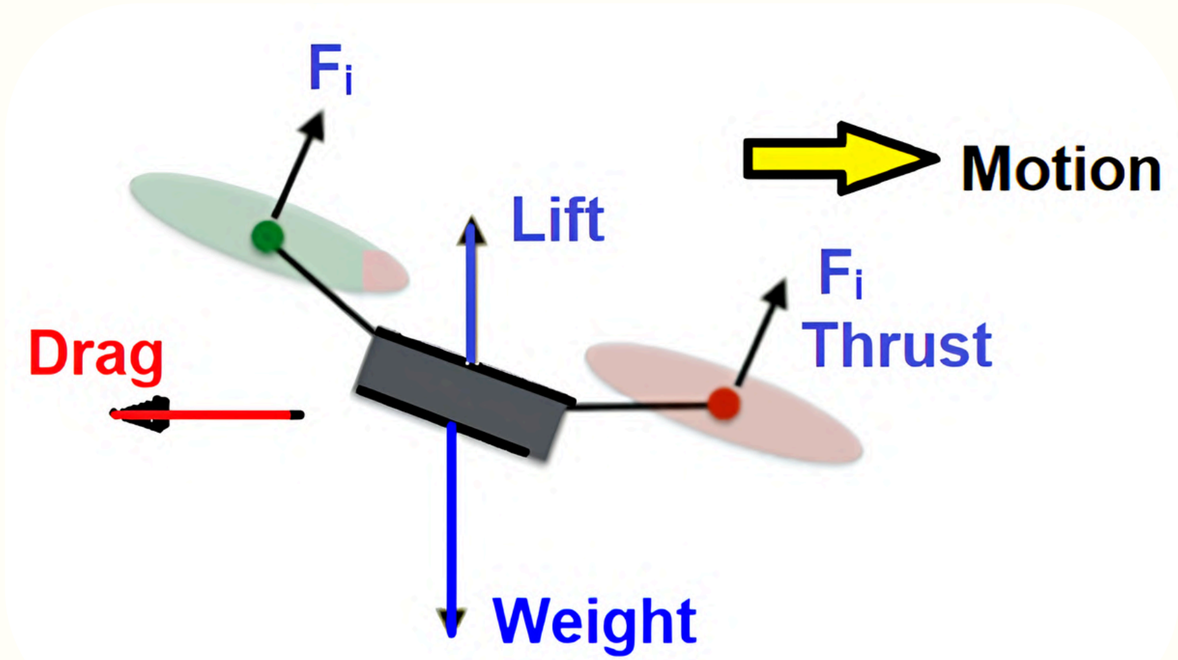


Photo : https://cfdflowengineering

EQUATION

$$F_D = \frac{1}{2}\rho v^2 C_D A$$

Note : F_D = Drag force

Torque

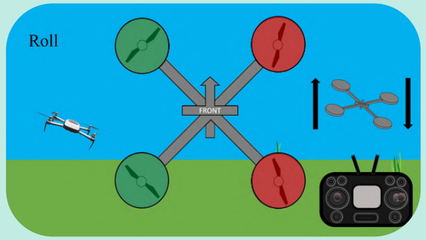
$$\tau_{\phi} = (T_{left} - T_{right}) \cdot l$$

$$\tau_{\theta} = (T_{front} - T_{back}) \cdot l$$

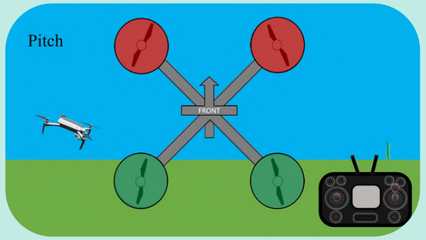
$$\tau_{\psi} = (M_{ccw} - M_{cw}) \cdot l$$

NED Frame

Roll Torque (φ-X axis)



Pitch Torque (θ-Y axis)



Yaw Torque (ψ-Z axis)

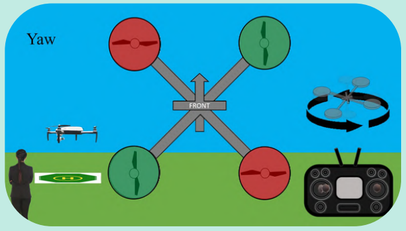


Photo : https://www.lesics.com/how-does-drones-work.html , Youtube:Tactical Aviation

Translational (Newton's 2nd law)

$$m\ddot{r} = F_{thrust} + F_{gravity} + F_{drag}$$

- $\ddot{r}$ = linear acceleration
  acceleration(inertial frame)
  === > $\ddot{x}, \ddot{y}, \ddot{z}$ < ===
- Rotation matrix or Quaternion

Equations of MOTION

Rotational (Euler's equations)

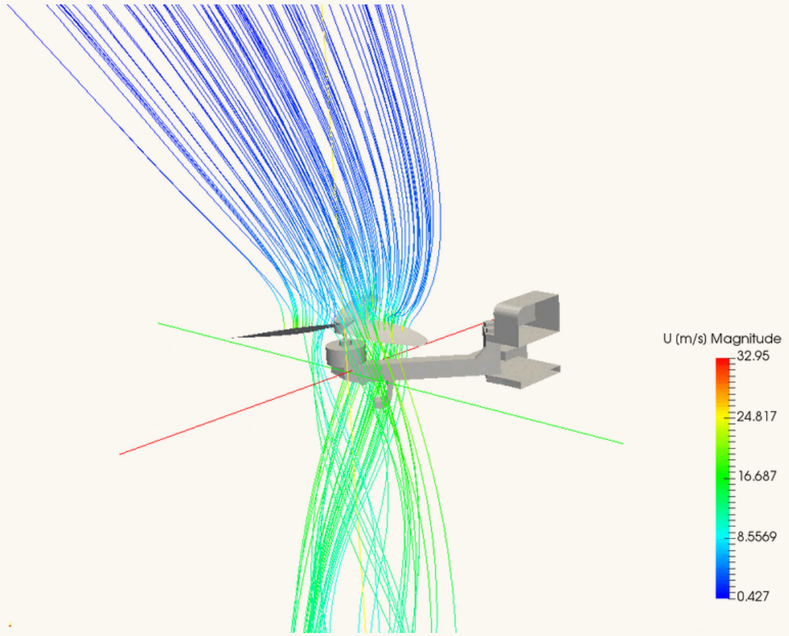
$$I\dot{\omega} + \omega \times (I\omega) = \tau$$

- I = inertia matrix (kg·m²)
- ω = angular velocity vector in body frame
- τ = torque vector

$$m\ddot{\mathbf{r}} = R_{b \to i} \begin{bmatrix} 0 \\ 0 \\ -\sum T_i \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} - k_d \dot{\mathbf{r}}$$

$$\begin{cases} I_x \dot{p} = \tau_\phi + (I_y - I_z) qr \\ I_y \dot{q} = \tau_\theta + (I_z - I_x) pr \\ I_z \dot{r} = \tau_\psi + (I_x - I_y) pq \end{cases}$$

Motor & Propeller



Linear flow	$L = 0.5\rho V^2 S C_L$
Rotational flow	$T = k_f \omega^2,\ k_f = 0.5\rho C_L \int_0^R c(r) r^2 dr$

Photo : https://www.simscale.com/

COMUNICATION

Control link

Telemetry link

2.4 GHz

2.4 / 5.8 GHz

NRF24L01	2.4 GHz ISM	100–300 m	1–10 ms	Cheap, plenty of libraries, low latency, moderate range, poor penetration through obstacles
ESP-NOW	2.4 GHz Wi-Fi	100–300 m	1–10 ms	Easy with ESP32, supports multiple nodes, low latency, medium bandwidth, no router needed
DSMX / DSM2	2.4 GHz	500–1000 m	~10 ms	Professional RC standard, anti-jamming, requires Spektrum TX/RX, more expensive than DIY
LoRa	433/868/915 MHz	1–15 km	50–200 ms	Long-range, good obstacle penetration, low bandwidth, high latency – not suitable for aggressive flight

VTX Type	Frequency	Range (LoS)	Latency	Pros	Cons
Analog FPV	5.8 GHz	200–1000 m	20–40 ms	Very low latency, cheap	Low resolution, interference
Digital FPV	5.8 GHz / 2.4 GHz	500–2000 m	30–100 ms	HD video, stable	Higher latency, costly
DIY Wi-Fi VTX	2.4 GHz Wi-Fi	50–200 m	100–300 ms	Easy integration, cheap	High latency, short range

Flight Controller

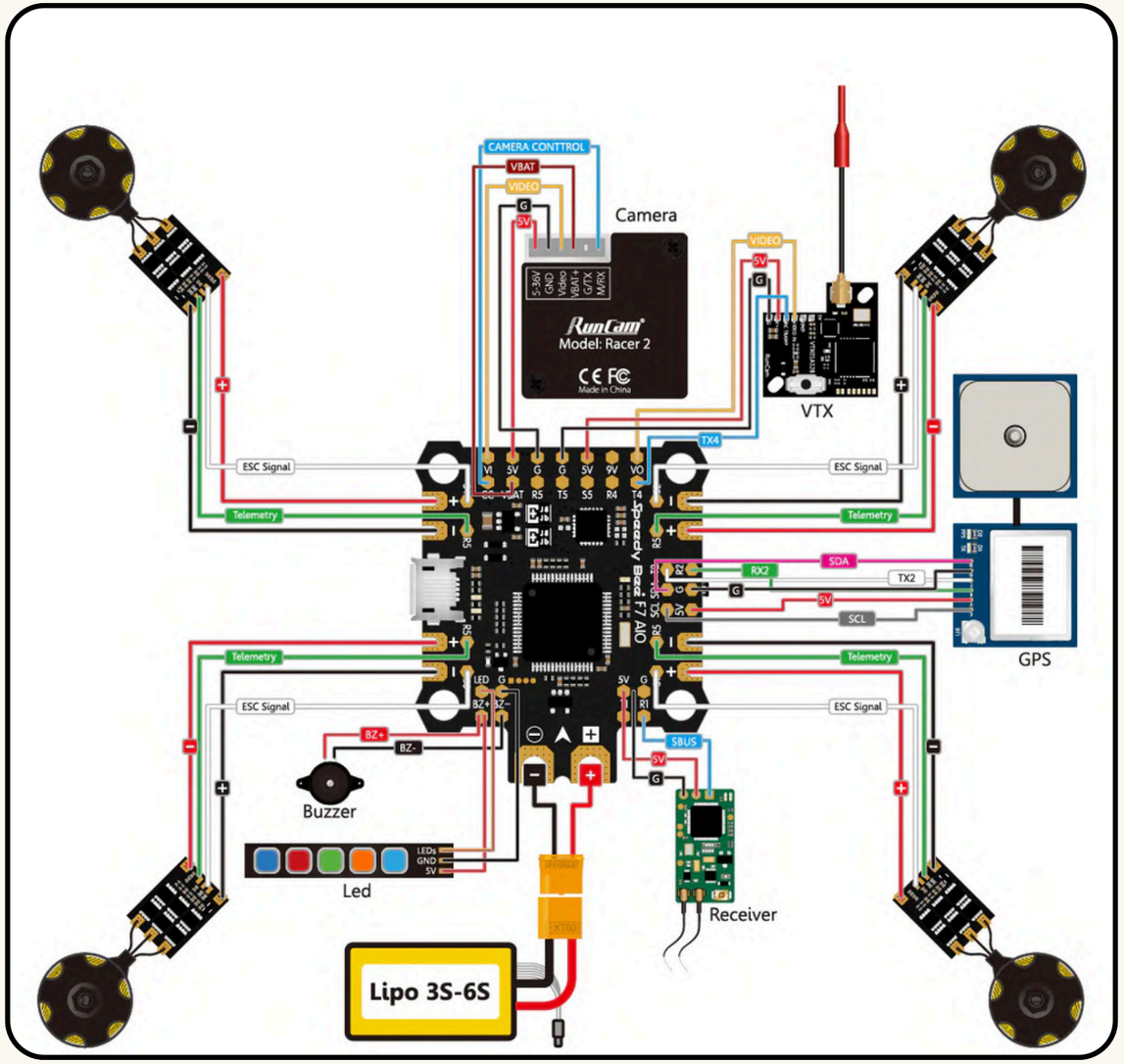
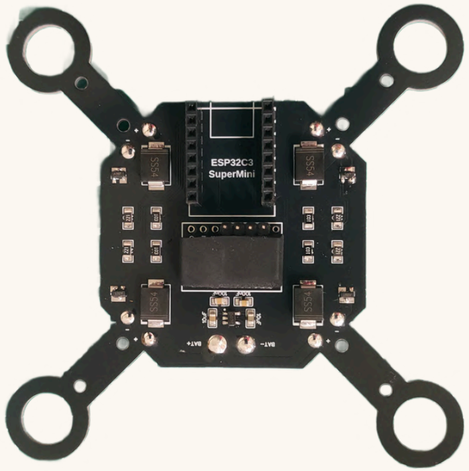
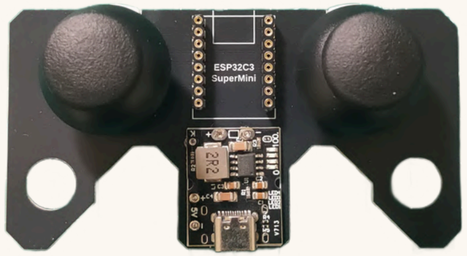
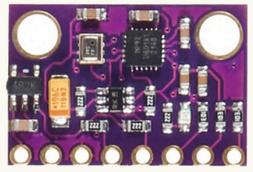


Photo : https://dronewolf.darkwolf.io/

QUADCOPTER COMPONENTS







You need to get it yourself

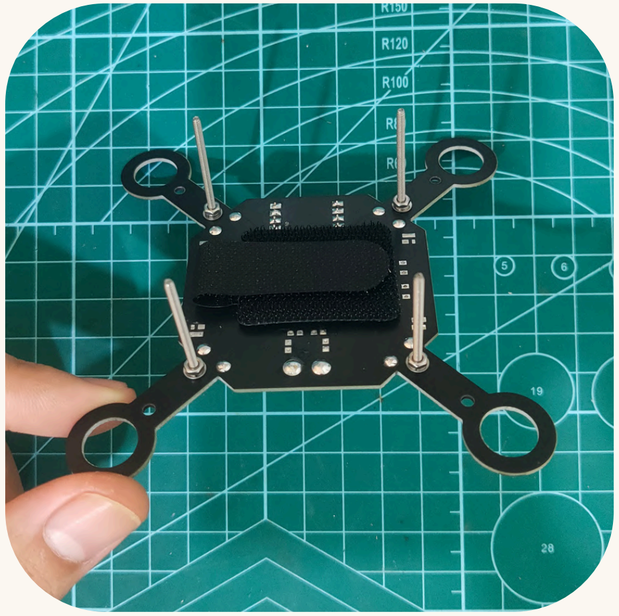
X2

ASSEMBLY

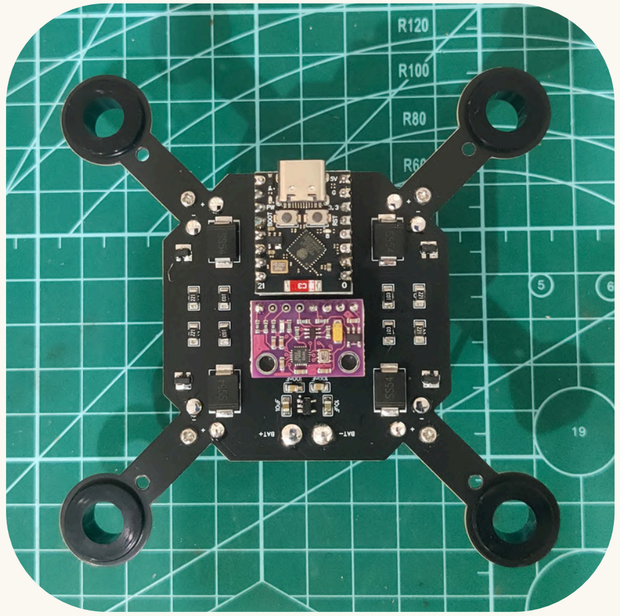
1

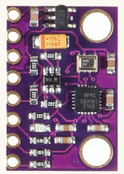
Start

2



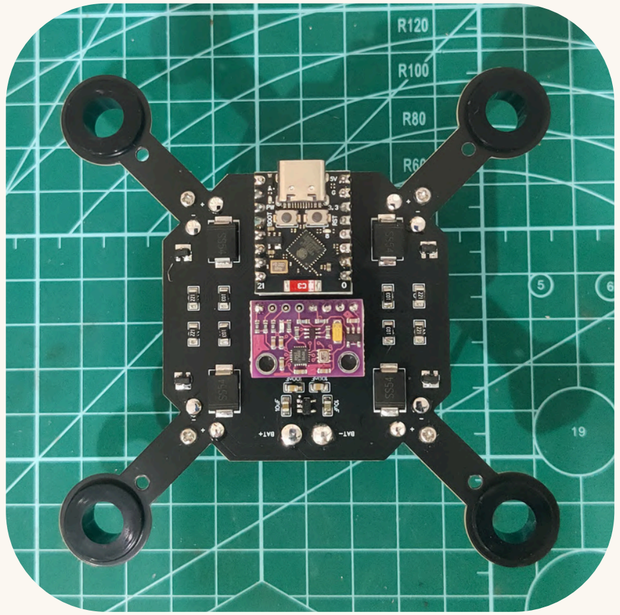
3



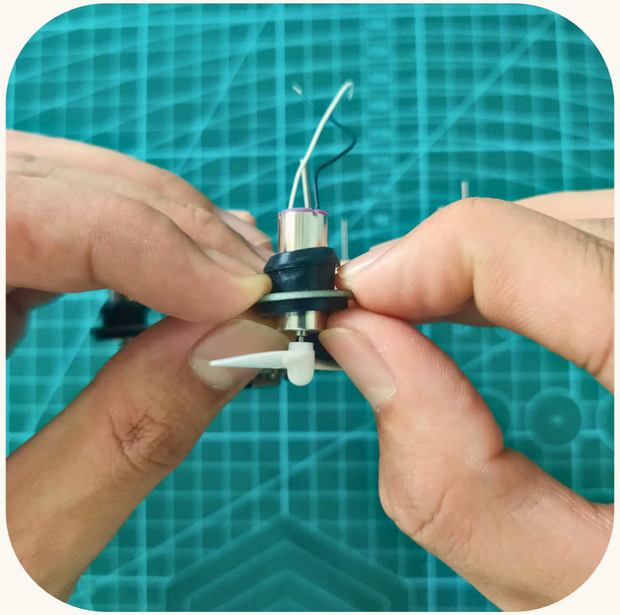


ASSEMBLY

4

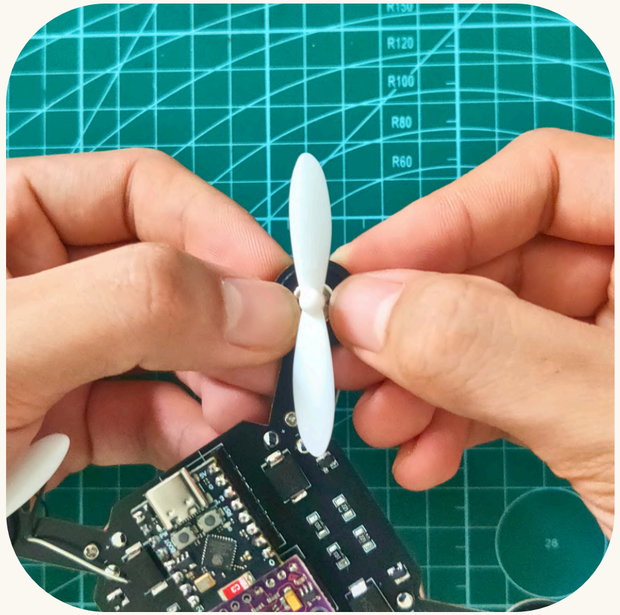


5.1



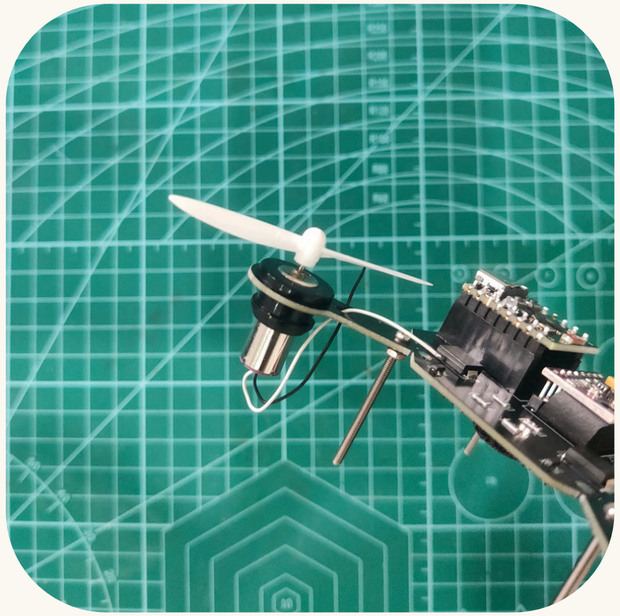
Following 5.4

5.2

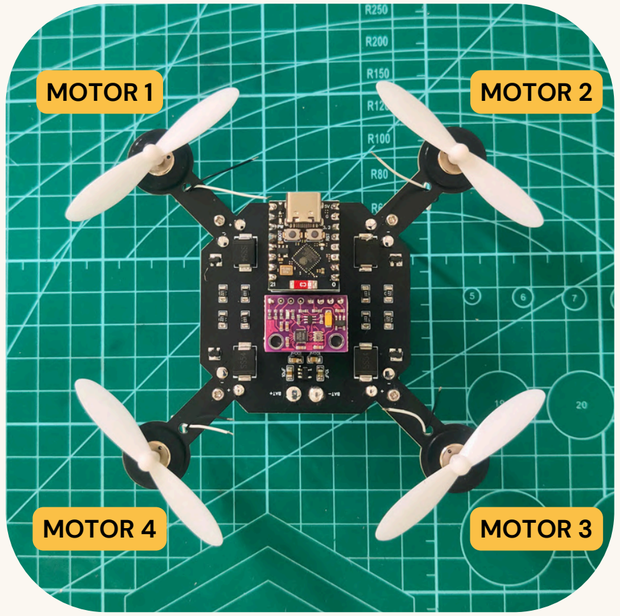


ASSEMBLY

5.3



5.4



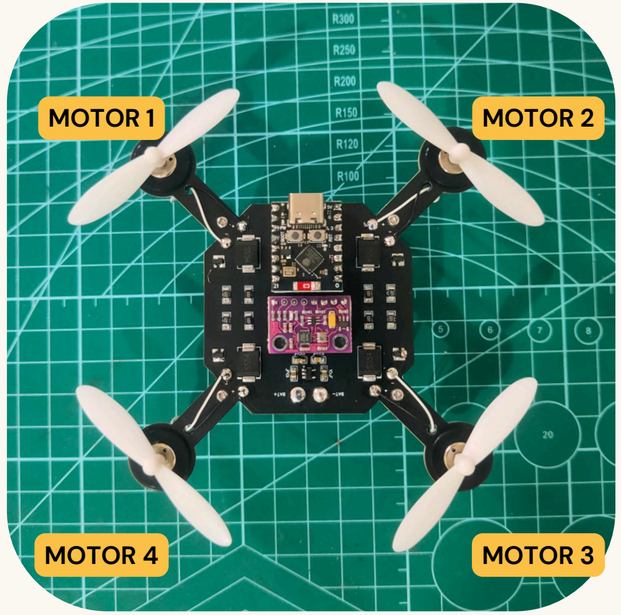
MOTOR 1 : PROPELLER A

MOTOR 2 : PROPELLER B

MOTOR 3 : PROPELLER A

MOTOR 4 : PROPELLER B

5.5



MOTOR 1 : (BLACK) >> (+) , (WHITE) >> (-)

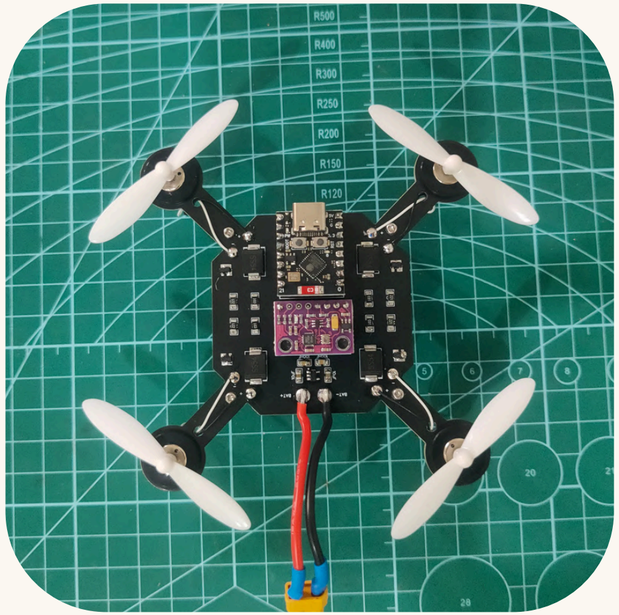
MOTOR 2 : (BLACK) >> (-) , (WHITE) >> (+)

MOTOR 3 : (BLACK) >> (+) , (WHITE) >> (-)

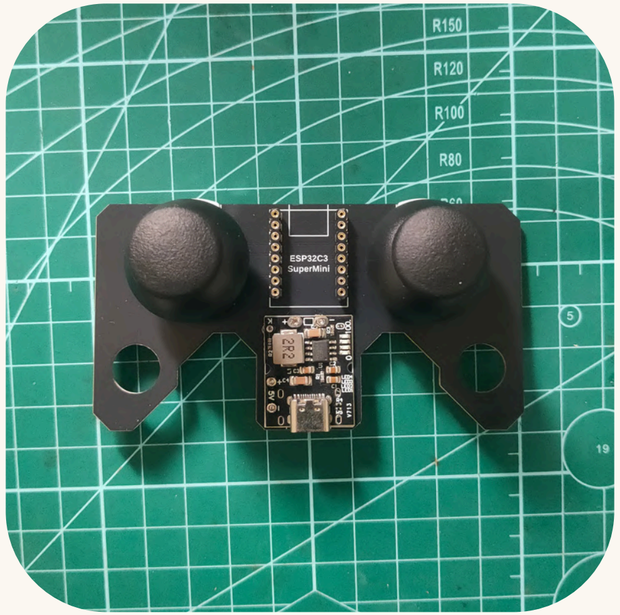
MOTOR 4 : (BLACK) >> (-) , (WHITE) >> (+)

ASSEMBLY

6

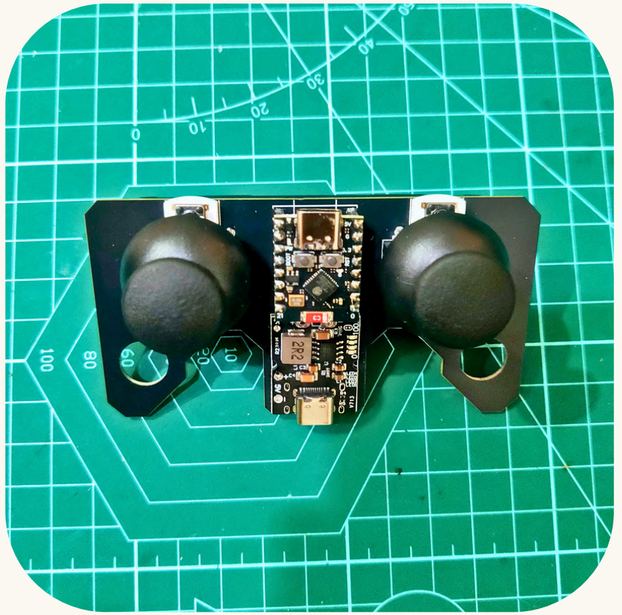


7.1



Start

7.2



ASSEMBLY

7.3

Back

7.4

7.5

ASSEMBLY

7.6

+

7.7

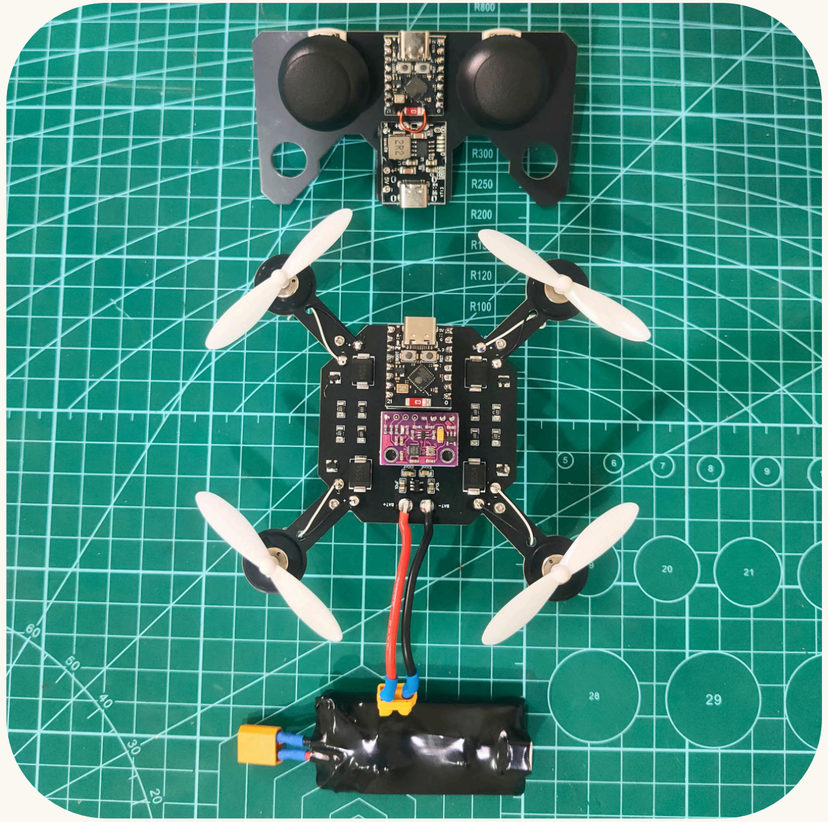
-

7.8

WARNING !!!

ASSEMBLY

Completed



Airplane Coding

ARDUINO IDE

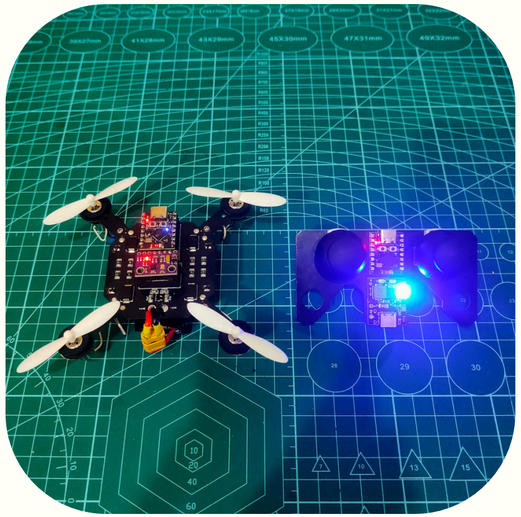
```
#include <WiFi.h>
#include <esp_now.h>
#include <Wire.h>
#include <math.h>
#include <MadgwickAHRS.h>
#include <Adafruit_BMP280.h>
#include <MPU9250_WE.h>
#define MPU9250_ADDR 0x68
```

User manual

FIRMWARE INSTALLATION

Setup

1



Power on

2

Double press left button

(open web server)

3

Open WiFi

Setup

4

Select : DragonFly Spirit

Passkey : 12345678

5

Copy Mac Address & exit to step 6

6

Open : Controller Calibration & paste and save Mac

Passkey : 12345678

Setup

7

Calibrating & get joy stick value

(push & press all axis)

8

Press Save Min/Max button

(do this after step 7)

9

Press Calibrate Center button

(do this when all button are center position)

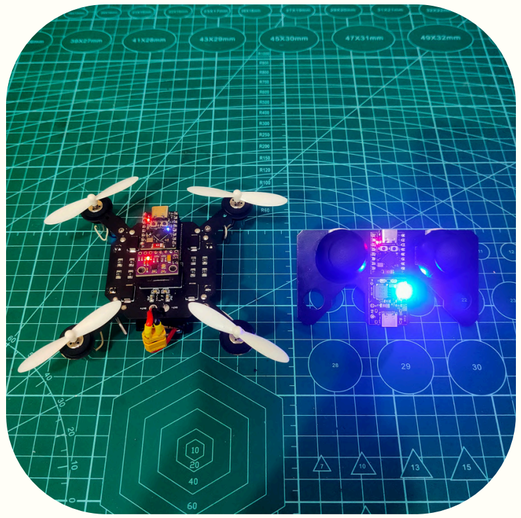
Setup

10

Double press right button

(close web server)

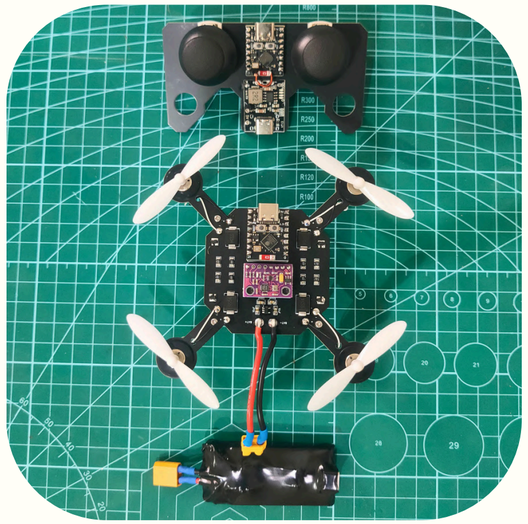
11



Restart system

(power off & power on)

12



I’m ready to.... ^_^

Controller Tuning

All data in realtime

Controller Tuning

min value of XL | Yaw rate

min value of YL | Altitude rate

min value of XR | Roll angle

min value of YR | Pitch angle

max value of XL | Yaw rate

max value of YL | Altitude rate

max value of XR | Roll angle

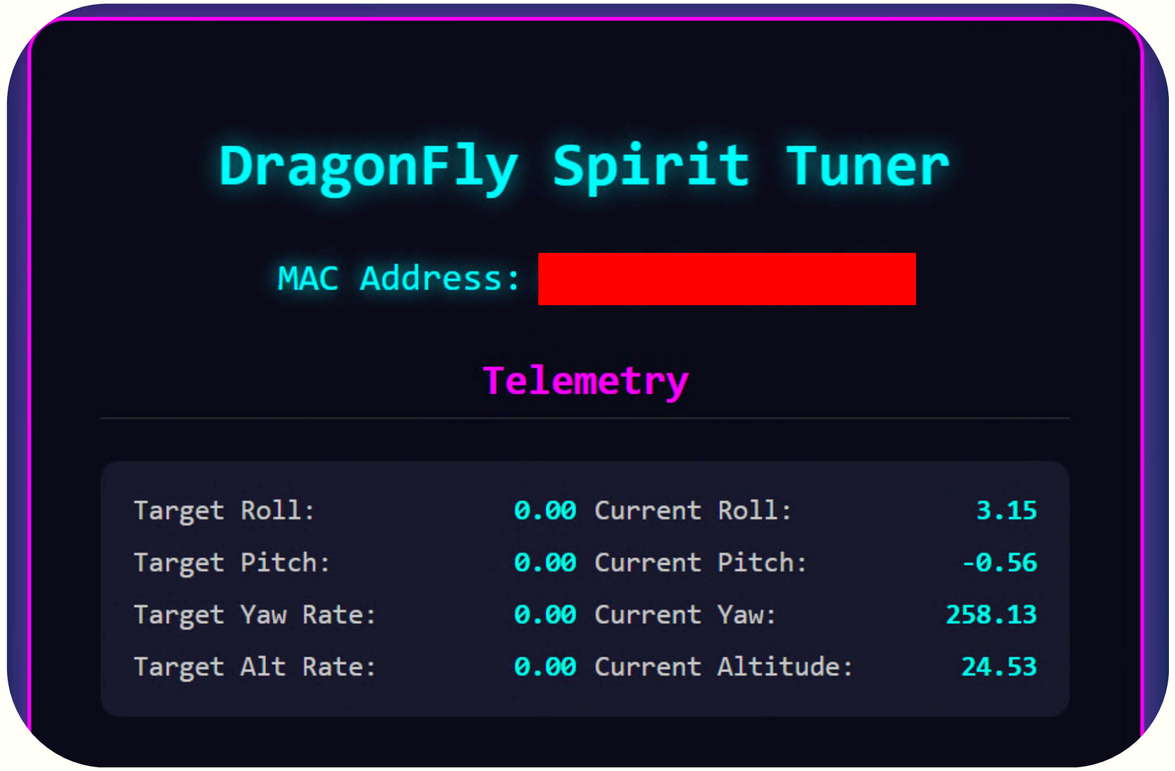
max value of YR | Pitch angle

Unit

- XL : deg/s
- YL : cm/s
- XR : deg
- YR : deg

Quadcopter Tuning

All data in realtime



Quadcopter Tuning

PID for Inner loop

PID Roll Rate

Kp: 1.20

Ki: 0.00

Kd: 0.03

PID Pitch Rate

Kp: 1.20

Ki: 0.00

Kd: 0.04

PID Yaw Rate

Kp: 3.00

Ki: 0.00

Kd: 0.03

PID Altitude Rate (m/s)

Kp: 45.00

Ki: 0.00

Kd: 1.50

Quadcopter Tuning

PID for Outer loop

PID Roll Angle

Kp: 10.00

Ki: 0.00

Kd: 0.00

PID Pitch Angle

Kp: 9.00

Ki: 0.50

Kd: 0.00

PID Yaw Angle

Kp: 4.00

Ki: 0.00

Kd: 0.00

PID Altitude

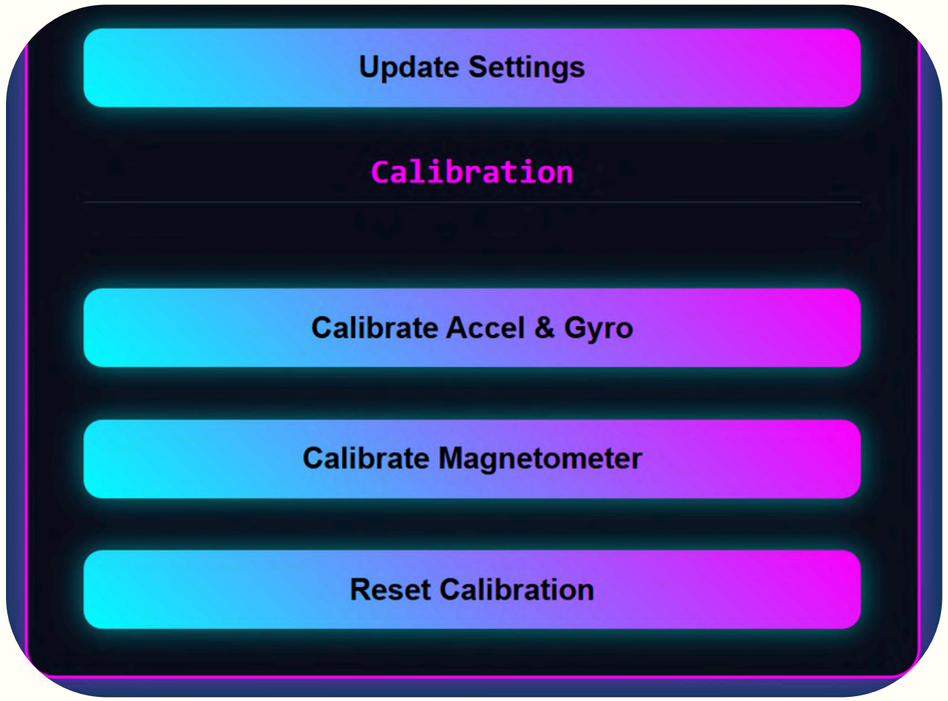
Kp: 1.00

Ki: 0.00

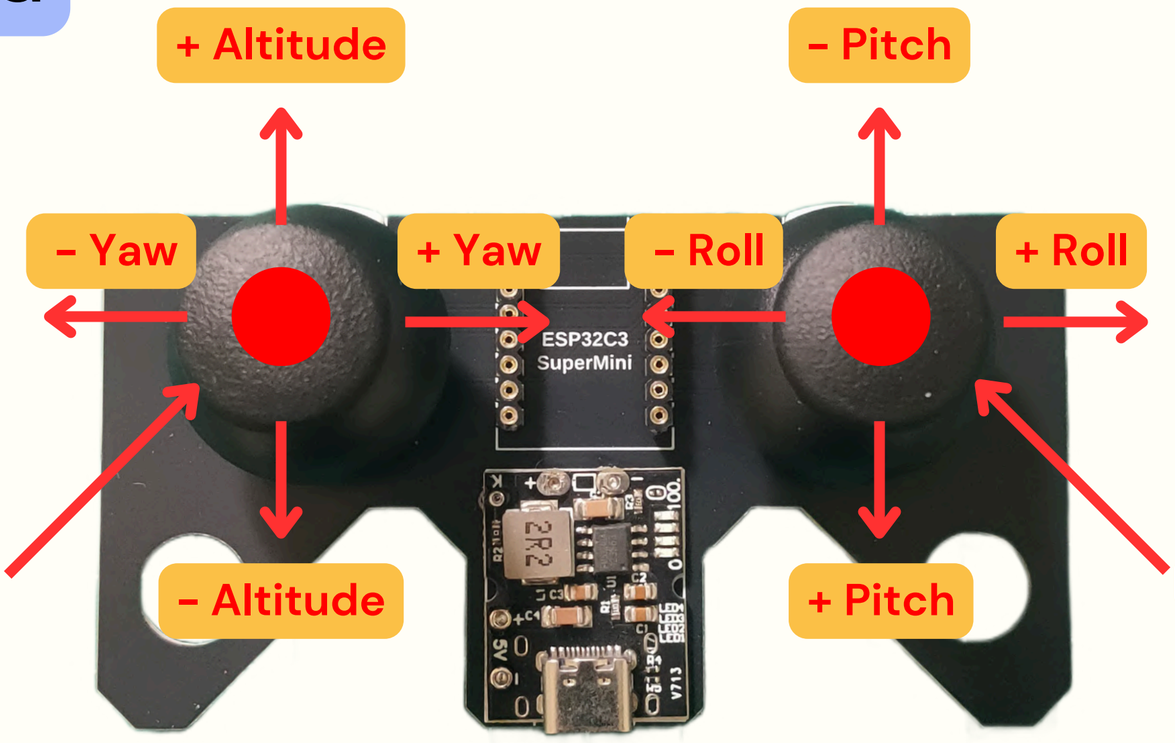
Kd: 0.40

Quadcopter Tuning

Initial parameters & All buttons

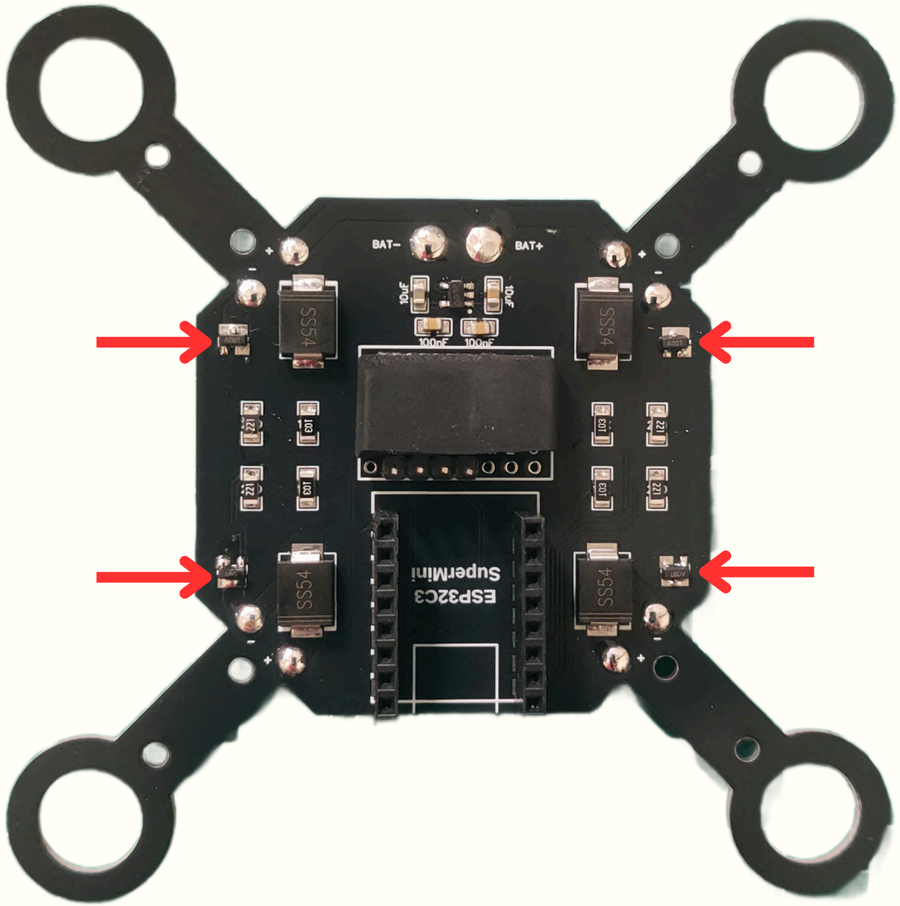


CONTROLLING



Back side : single press ==> power on | double press ==> power off

Fix & Repair


BAT−
BAT+
10uF
10uF
100nF
100nF
SS54
SS54
SS54
SS54
ESP32C3
SuperMini


A09T
A09T
A09T
A09T

ข้อชี้แจงและคำเตือนการใช้งาน

- ชุดนี้เป็นอุปกรณ์เพื่อการเรียนรู้ด้านอิเล็กทรอนิกส์และการเขียนโปรแกรมเท่านั้น (Educational STEM Kit) ไม่ใช่ชุดโดรนสำเร็จพร้อมบินใดๆ ทั้งสิ้น
- ผู้จัดจำหน่ายไม่ได้จำหน่ายอุปกรณ์วิทยุคมนาคมหรืออากาศยานไร้คนขับในเชิงพาณิชย์
- ชุดนี้จัดส่งเป็น ชิ้นส่วน (Parts Only) สำหรับการทดลองและประกอบด้วยตนเอง เพื่อการศึกษาเรียนรู้ โดยไม่มีบอร์ดสื่อสาร ESP32 C3 supermini รวมอยู่ในชุด และไม่ใช่อุปกรณ์สำเร็จรูปพร้อมใช้งาน เน้นเพื่อศึกษาเรียนรู้เท่านั้น

รายละเอียดสำคัญ

- ชุดนี้ไม่มี กล้องหรืออุปกรณ์บันทึกภาพ
- น้ำหนักรวมของโครงสร้างและอุปกรณ์เมื่อประกอบเสร็จ ไม่เกิน 2 กิโลกรัม
- ไม่มีโมดูลสื่อสาร (เช่น ESP32 C3 supermini หรือ Wi-Fi module) ติดตั้งหรือส่งไปให้
- ผู้ใช้ต้องจัดหาและติดตั้งบอร์ด ESP32 C3 supermini หรือไมโครคอนโทรลเลอร์ด้วยตนเอง

คำเตือนด้านกฎหมาย

- คลื่นความถี่ที่ใช้โดย ESP32 อยู่ในย่านที่ กสทช. อนุญาตให้ใช้งานได้โดยไม่ต้องขออนุญาตรายบุคคล (unlicensed band) แต่ต้องเป็นไปตามมาตรฐานทางเทคนิคของอุปกรณ์นั้น ๆ ผู้ใช้ควรเลือกบอร์ดที่ผ่านการรับรองมาตรฐานจากผู้จำหน่ายที่ถูกต้อง
- หากผู้ใช้ประกอบเป็นเครื่องบินบังคับวิทยุ หรืออากาศยานและทำการบินจริง
  – ไม่มีกล้อง และน้ำหนักไม่เกิน 2 กิโลกรัม → ปัจจุบันไม่อยู่ในขอบเขตที่ต้องขึ้นทะเบียนกับสำนักงานการบินพลเรือนแห่งประเทศไทย (CAAT)
  – หากติดกล้อง หรือน้ำหนักเกิน 2 กิโลกรัม → ต้องดำเนินการขึ้นทะเบียนกับ CAAT และจดทะเบียนเครื่องวิทยุกับ กสทช. ก่อนนำไปบิน
  – ผู้จัดจำหน่ายไม่มีส่วนรับผิดชอบต่อการใช้งานที่ฝ่าฝืนกฎหมายหรือระเบียบการบินใด ๆ
- หากผู้ใช้ประกอบเป็นหุ่นยนต์ Rover , รถบังคับ → ปัจจุบันไม่อยู่ในขอบเขตที่ต้องขึ้นทะเบียนกับสำนักงานการบินพลเรือนแห่งประเทศไทย (CAAT) แต่ยังต้องใช้อุปกรณ์วิทยุคมนาคมที่ได้รับมาตรฐานเช่นเดิม
- การปรับแต่งหรือดัดแปลงเพื่อใช้งานนอกเหนือจากวัตถุประสงค์เพื่อการเรียนรู้ ถือเป็นความรับผิดชอบของผู้ใช้โดยตรง
- ควรใช้เพื่อการทดลองหรือการเรียนรู้ในพื้นที่จำกัด เช่น ห้องเรียนหรือพื้นที่ในร่ม (indoor use only) เพื่อความปลอดภัย
- ห้ามใช้งานในพื้นที่สาธารณะ สนามบิน พื้นที่ห้ามบิน หรือสถานที่ราชการโดยไม่ได้รับอนุญาต
- ชุดนี้มีวัตถุประสงค์เพื่อการศึกษา การทดลอง และการสร้างสรรค์ผลงาน Maker เท่านั้น ไม่ได้ออกแบบหรือรับรองสำหรับการใช้งานในเชิงพาณิชย์ อุตสาหกรรม หรือการบินจริง

By T Maker

END

Thank you