
pyDrone

发行版本 *v1.0*

CaptainJackey

2022 年 07 月 27 日

Contents

1	pyDrone 用户手册	3
1.1	硬件资源	3
1.2	教程资料	3
1.3	组装和飞行视频	3
1.4	构造函数	5
1.5	动作	5
1.6	WiFi 连接	6
2	关于 pyDrone	9
2.1	项目简介	9
2.2	购买方式	9
2.3	联系我们	9
	索引	11

-MicroPython 开源四轴飞行器-

1.1 硬件资源

- [pyDrone 原理图 \(PDF\)](#)

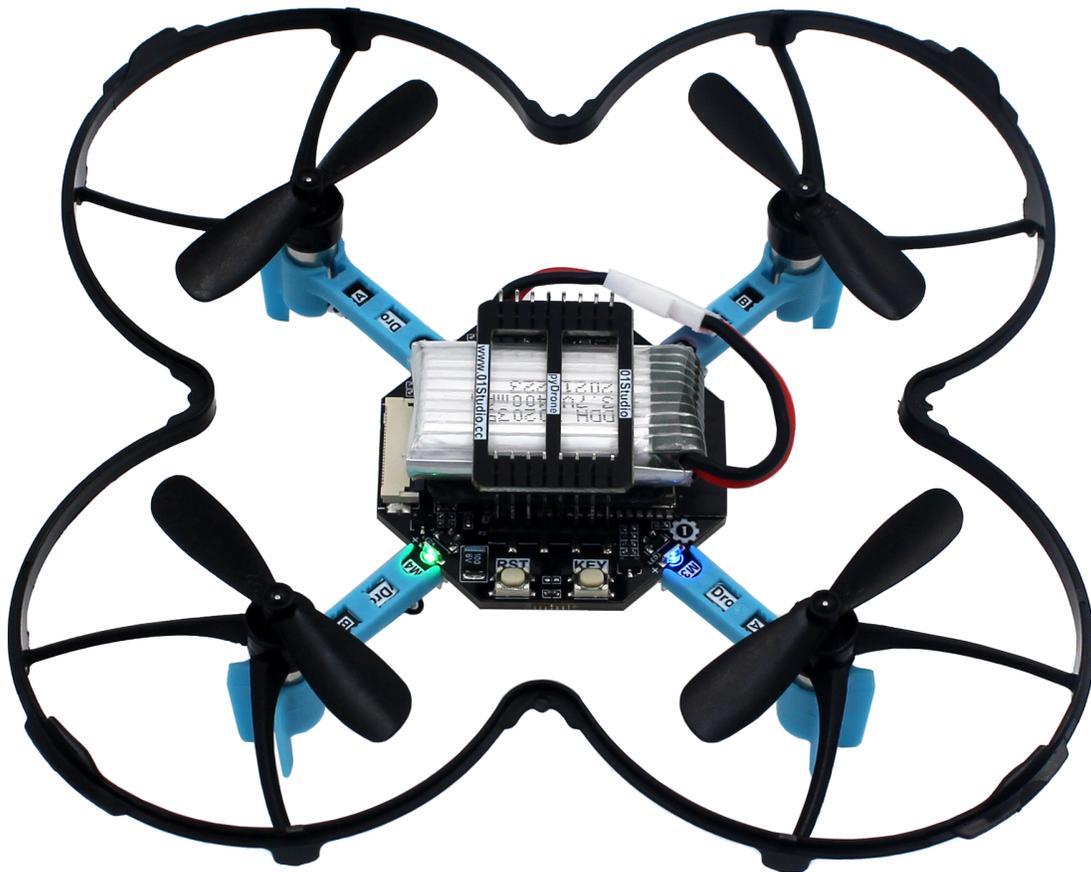
1.2 教程资料

pyDrone 主控为 ESP32-S3，基于 MicroPython 实现，01Studio 提供完整的入门教程和代码相关资料：

https://download.01studio.cc/zh_CN/latest/project/pyDrone/pyDrone.html

1.3 组装和飞行视频

- 组装和飞行视频教程：<https://b23.tv/BV1ct4y1V7qG>



1.4 构造函数

`class drone.DRONE (flightmode=0)`

构建四轴对象。

- `flightmode` 飞行模式。
 - 0 - 无头模式。表示以周围环境为参照系，飞行的前后左右方向跟机身无关，通常用于室内飞行；
 - 1 - 有头模式。表示以四轴机身作为参照系，飞行的前后左右方向跟机身有关。四轴自转后保持机头方向为正前方，通常用于室外飞行。

示例:

```
import drone

d = drone.Drone(flightmode = 0) #构建四轴对象,无头飞行模式。
```

1.5 动作

`DRONE.read_cal_data ()`

获取校准数据。返回 3 个融合的 X/Y/Z 校准数据值，当 3 个值均少于 5000 时候，校准通过。

`DRONE.read_calibrated ()`

获取校准状态。校准通过返回 1，不通过返回 0。对象构建后自动开始校准，请尽量将四轴水平放置以便提高校准速度。校准时间通常在启动后 10 秒内完成。

`DRONE.take_off (distance=80)`

起飞:

- `distance` 起飞后悬停高度，默认 80cm。可以设置范围 30-2000cm:

`DRONE.landing ()`

降落：四轴飞行器缓慢降落，这时候可以控制前后左右方向，降到底面后电机停止转动。

`DRONE.stop ()`

停止：所有电机立刻停止转动，用于突发情况迫降。

`DRONE.control (rol=0, pit=0, yaw=0, thr=0)`

四轴飞行器姿态控制:

- `rol` Roll 横滚角，控制四轴左右运动。范围: -100 ~ 100，“-”表示左，正表示右，绝对值越大，角度/油门越大。

- `pitch` Pitch 俯仰角, 控制四轴前后运动。范围: -100 ~ 100, “-” 表示后, 正表示前, 绝对值越大, 角度/油门越大。
- `yaw` Yaw 偏航角, 控制四轴自转运动。范围: -100 ~ 100, “-” 表示逆时针自转, 正表示顺时针自转, 绝对值越大, 角度/油门越大。
- `thrust` Thrust 推力, 控制四轴上下运动。范围: -100 ~ 100, “-” 表示下降, 正表示抬升, 绝对值越大, 油门越大。

`DRONE.read_states()`

读取四轴飞行器状态信息。返回 9 个数据的元组。

- 1、roll 值, 范围 [-18000 ~ 18000], 角度放大 100 倍。
- 2、pitch 值, 范围 [-18000 ~ 18000], 角度放大 100 倍。
- 3、yaw 值, 范围 [-18000 ~ 18000], 角度放大 100 倍。
- 4、遥控器 roll 控制量, 范围 [-1000 ~ 1000]
- 5、遥控器 pitch 控制量, 范围 [-1000 ~ 1000]
- 6、遥控器 yaw 控制量, 范围 [-200 ~ 200]
- 7、遥控器 Thrust 控制量, 范围 [0 ~ 100], 百分比, 摇杆回中时候约为 50, 即 50%。
- 8、电池电量, 单位 10mV。
- 9、相对高度, 单位 cm (与校准时候的相对高度)。

1.6 WiFi 连接

The network module:

```
import network

wlan = network.WLAN(network.STA_IF) # 创建 station 接口
wlan.active(True)                  # 激活接口
wlan.scan()                         # 扫描允许访问的SSID
wlan.isconnected()                 # 检查创建的station是否已经连接到AP
wlan.connect('ssid', 'password') # 连接到指定ESSID网络
wlan.config('mac')                 # 获取接口的MAC地址
wlan.ifconfig()                    # 获取接口的 IP/netmask(子网掩码)/gw(网关)/DNS 地址

ap = network.WLAN(network.AP_IF) # 创建一个AP热点接口
ap.config(essid='ESP-AP')         # 激活接口
ap.config(max_clients=10)         # 设置热点允许连接数量
ap.active(True)                   # 设置AP的ESSID名称
```

连接到本地 WIFI 网络的函数参考:

```
def do_connect():
    import network
    wlan = network.WLAN(network.STA_IF)
    wlan.active(True)
    if not wlan.isconnected():
        print('connecting to network...')
        wlan.connect('essid', 'password')
        while not wlan.isconnected():
            pass
    print('network config:', wlan.ifconfig())
```

一旦网络建立成功,你就可以通过 `socket` 模块创建和使用 TCP/UDP sockets 通讯,以及通过 `urequests` 模块非常方便地发送 HTTP 请求。

2.1 项目简介

由 01Studio(01 科技) 发起的 MicroPython 开源四轴飞行器。

Micropython 是指使用 Python 做各类嵌入式硬件设备编程。MicroPython 发展势头强劲，01Studio 一直致力于 Python 嵌入式编程，特此推出 pyDrone 开源项目，旨在让 MicroPython 变得更加流行。使用 MicroPython，你可以轻松地实现四轴飞行器的起飞、降落、悬停、移动、自转等各种姿态和动作。

开源项目地址：<https://github.com/01studio-lab/pyDrone>

2.2 购买方式

- 官方淘宝店：<https://01studio.taobao.com/>

2.3 联系我们

- 社区论坛：<https://bbs.01studio.cc/>
- 邮箱：support@01studio.cc
- 商务合作：18123953882（微信同号）

C

`control()` (DRONE 方法), 5

D

`drone.DRONE` (内置类), 5

L

`landing()` (DRONE 方法), 5

R

`read_cal_data()` (DRONE 方法), 5

`read_calibrated()` (DRONE 方法), 5

`read_states()` (DRONE 方法), 6

S

`stop()` (DRONE 方法), 5

T

`take_off()` (DRONE 方法), 5