

# Dronă ESP32

## Introducere

Proiectul constă în realizarea unei mini drone controlată de un microcontroller ESP32-C3. Scopul proiectului este de a crea o dronă simplă, cu funcționalități de bază, ce poate zbura stabil prin controlul motoarelor în funcție de datele primite de la un giroscop (MPU6050). Ideea a pornit din dorința de a învăța mai bine despre controlul motoarelor, interfațarea cu senzori și utilizarea ESP32 în aplicații embedded. Credem că acest proiect este util atât pentru noi, pentru aprofundarea practică a cunoștințelor, cât și pentru alți studenți sau pasionați care vor să construiască un microdron low-cost, open-source.

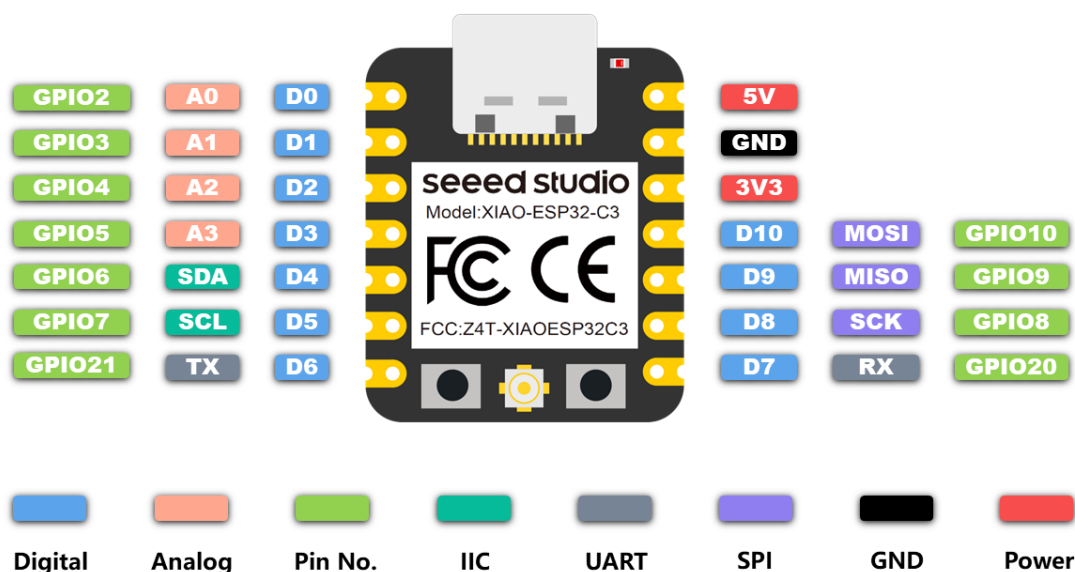
## Descriere generală

Schema bloc generală:



Descriere module și interacțiuni:

- ESP32-C3 WROOM



- MPU6050 (Giroscop și accelerometru)

Se conectează la ESP32-C3 prin protocol I2C (SCL, SDA). Trimite date despre mișcările dronei, astfel încât ESP32 să poată ajusta viteza fiecărui motor pentru stabilizare.

### • Driver Motoare Custom (MOSFETs + Diodă + Rezistor)

Primește semnale PWM de la ESP32 și controlează alimentarea celor 4 motoare coreless. Fiecare motor are un circuit driver format din:

1. MOSFET N SI2300
2. Diodă 1N4148 (pentru protecție)
3. Rezistor 10kOhmi (pull-down)
4. Condensator 100nF (pentru filtrare)

### • Motoare Coreless (x4)


1. **FL (Front-Left)** - CW (Clockwise)
2. **FR (Front-Right)** - CCW (Counter-Clockwise)
3. **BL (Back-Left)** - CCW
4. **BR (Back-Right)** - CW

Aceste motoare sunt aranjate astfel încât să creeze stabilitate prin rotație opusă două câte două.

### • Baterie 3.7V LiPo

Alimentează întreaga dronă. Tensiunea ei este monitorizată de ESP32 pentru a preveni descărcarea excesivă.

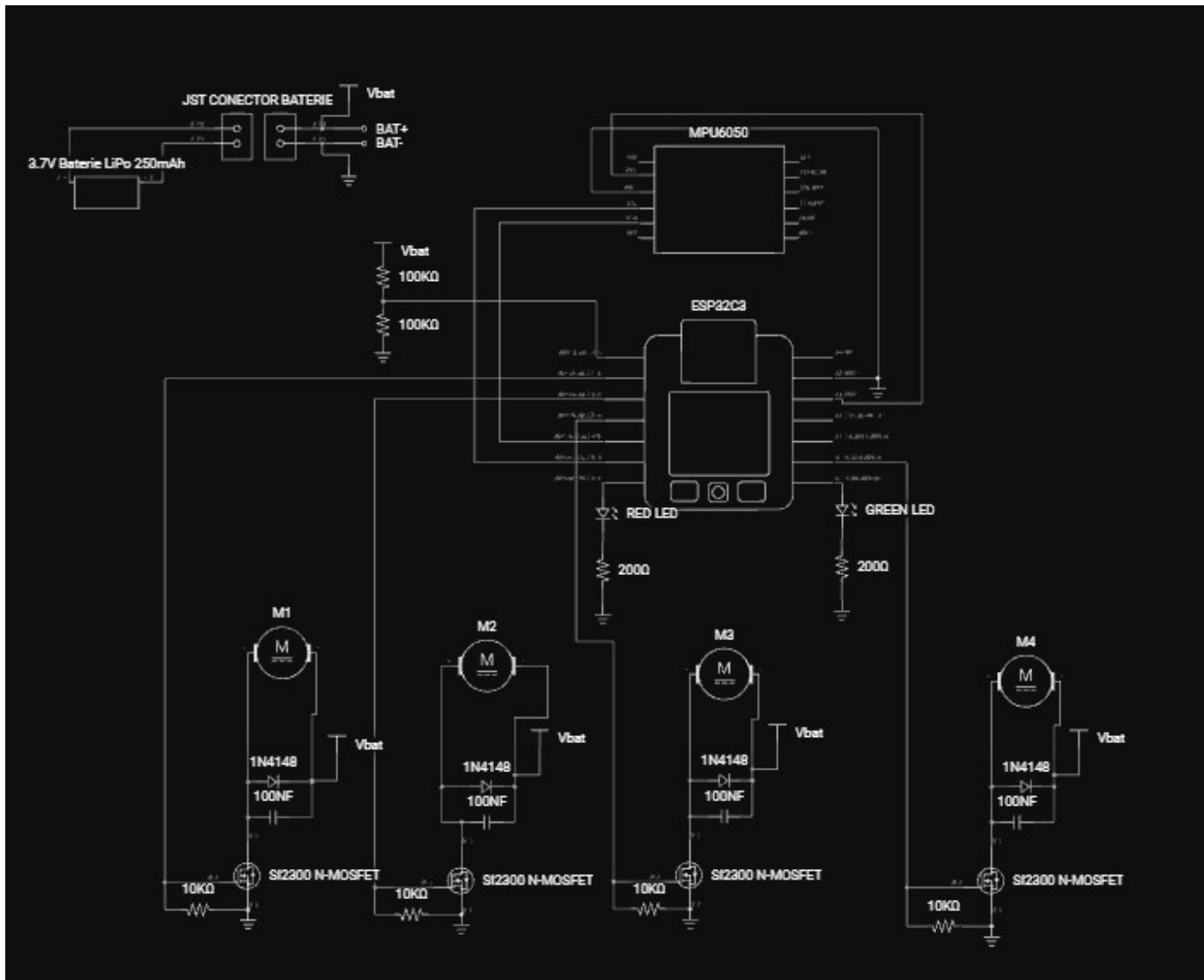
### • Divizor rezistiv pentru monitorizarea bateriei

Format din 2 rezistențe de 100kOhmi, conectate între Vbat și GND, cu punctul median legat la pinul A0 al ESP32. Acesta scade tensiunea bateriei la un nivel sigur pentru citirea de către ESP32. 

## Hardware Design

**Lista de piese:** [Fișierul BOM cu toate componentele necesare](#)

**Schema electrică:**



## Documentație Software

### Stadiul actual al implementării software

Proiectul este funcțional și bazat pe codul open-source [ESP-Drone](<https://github.com/espressif/esp-drone/tree/master>), adaptat pentru ESP32-C3. Au fost făcute modificări la nivelul pinilor conform schemei electrice proprii și s-a configurat ADC-ul pentru monitorizarea bateriei. Conexiunea Wi-Fi funcționează, comenzile sunt recepționate prin UDP, iar drona răspunde corect la comenzi.

## Alegerea bibliotecilor

Am folosit **ESP-IDF**, deoarece este framework-ul oficial Espressif, oferă control total asupra hardware-ului, este bine documentat și include unelte integrate pentru build, flash și debug.

## Element de noutate

Proiectul aduce noutate prin adaptarea codului ESP-Drone pe microcontrolerul **ESP32-C3**, cu o configurație personalizată a pinilor și integrarea unui sistem de monitorizare a bateriei cu ADC. Controlul se face wireless prin aplicație mobilă sau gamepad.

## Utilizarea funcționalităților din laborator

- PWM pentru controlul motoarelor - ADC pentru citirea tensiunii bateriei - Wi-Fi în mod AP și comunicație UDP - Task-uri și cozi din FreeRTOS - Debug serial și monitorizare loguri

## Structura proiectului și validare

Proiectul este împărțit pe fișiere:

- **wifi\_esp32.c** - conexiune Wi-Fi și UDP
- **main.c** - inițializare și logica principală
- **config.h** - definiții pini și parametri
- **adc\_esp32.c** - citire și conversie ADC
- **motors.c** - interpretare comenzi și control motoare
- **led\_esp32.c** - controlul ledurilor

Validarea s-a făcut prin conectarea aplicației mobile și testarea comenzilor de zbor, cu verificare vizuală și în serial monitor.

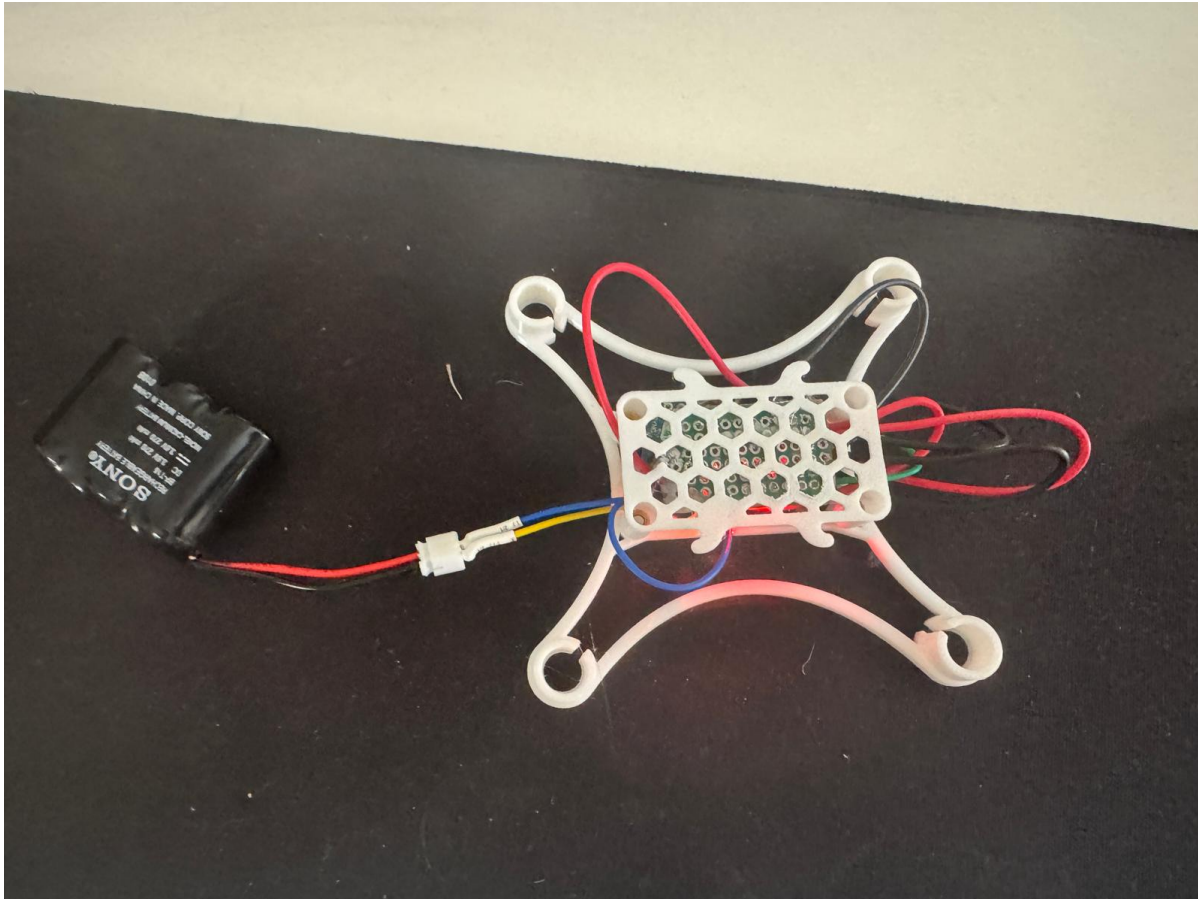
## Calibrarea senzorilor

Am calibrat ADC-ul pe baza valorilor cunoscute de tensiune (3.3V și 4.2V). Tensiunea citită este convertită în procente pentru a estima nivelul bateriei. Valorile sunt filtrate pentru stabilitate.

# Optimizări

- Dimensiuni reduse pentru buffer-ele UDP - Eliminarea componentelor nefolosite din proiect - Sincronizare eficientă între task-uri cu cozi - Inițializare Wi-Fi rapidă fără DHCP

## Rezultate Obținute



## Download

Arhiva cu codul sursă: [Descarcă drona.zip](#)

GitHub: [Vezi proiectul pe GitHub](#)

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

## Bibliografie/Resurse

[ESP-Drone - GitHub ESP-IDF Documentation \(ESP32-C3\)](#) [Random Nerd Tutorials - ESP Projects ESP-IDF ADC API Reference ESP32-C3 Resources Getting Started with ESP32C3](#)

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2025/vradulescu/constantin.matei03>



Last update: **2025/05/27 10:11**