

Guia per instal·lar MicroPython i programar el M5Core2 amb Thonny

PAS 1: Esborrar la memòria del M5Core2

Obre el terminal del Mac i escriu:

```
esptool.py --chip esp32 --port /dev/cu.wchusbserialXXXXX erase_flash
```

Canvia **XXXXX** pel nom del teu port (ex: `/dev/cu.wchusbserial1586A0319471`).

PAS 2: Instal·lar el firmware MicroPython per ESP32

Baixa el firmware oficial per ESP32:

- Fitxer recomanat: `ESP32_GENERIC-20230426-v1.20.0.bin`

Carrega'l al dispositiu:

```
esptool.py --chip esp32 --port /dev/cu.wchusbserialXXXXX --baud 115200 write_flash -z 0x1000 ESP32_GENERIC-20230426-v1.20.0.bin
```

PAS 3: Configura Thonny

1. Obre **Thonny**.
 2. Ves a **Thonny > Preferences > Interpreter**.
 3. Tria:
 - **Interpreter:** `MicroPython (ESP32)`
 - **Port:** `/dev/cu.wchusbserialXXXXX`
 4. Fes clic a **OK**.
-

PAS 4: Escriu o carrega el teu programa

Exemple de codi:

```
from machine import Pin
import time
```

```
led = Pin(10, Pin.OUT) # Prova amb un pin compatible amb Bottom2
```

```
while True:
    led.on()
    print("LED ON")
    time.sleep(0.5)
    led.off()
    print("LED OFF")
    time.sleep(0.5)
```

1. Desa'l com `test_m5core2.py`.
2. Obre el fitxer a Thonny.
3. Fes clic a **Run (F5)** per executar-ho.



Resultat esperat

- La consola de Thonny mostra "LED ON / LED OFF".
- Si el LED està connectat correctament, hauria de parpellejar.

Neus, ja tens un sistema estable per programar en Python el teu M5Core2. A partir d'aquí pots fer servir tot el que sàpigues de Python per controlar pantalles, mòduls, LEDs i servos!

Qualsevol dubte, simplement torna a obrir aquest document!

Podem fer servir el goPLus2 per fer servir un servo?