# Occhiali Mouse

Gli occhiali sono costituiti da un esp32 con batteria, un modulo giroscopio MPU-6050 e uno o più pulsanti

## Giroscopio

Sensore utilizzato: MPU-6050 / GY-521
Questo giroscopio ha due funzioni:

- **Accelerometro**

Misura di quanto il dispositivo è inclinato rispetto all’inclinazione zero data nel momento della calibrazione

- **Giroscopio**

Misura con che velocità il dispositivo sta ruotando attorno agli assi relativi al dispositivo rispetto alla velocità zero data nel momento della calibrazione

Per ora ho scelto di utilizzare solo i dati del giroscopio, in modo da far corrispondere uno spostamento del dispositivo a uno spostamento del mouse.

### Calibrazione

Per la calibrazione del giroscopio (senza accelerometro) è sufficiente che il dispositivo resti immobile: **non importa l'inclinazione** e non occorre che sia in bolla poiché la rotazione si riferisce sempre agli assi relativi del dispositivo.
La calibrazione serve solo a settare la velocità rotazionale zero.

### Montatura

È **necessario** invece che il giroscopio sia dritto quando viene indossato:
Non importa il verso in cui viene montato, gli assi possono essere scambiati tra loro nel codice.
Ma è importante che questi assi siano dritti rispetto agli assi di rotazione della testa quando il dispositivo viene utilizzato (indipendentemente dagli assi terrestri), altrimenti una rotazione verso destra della testa corrisponderebbe a una rotazione su più assi del giroscopio.

### Collegamento

Per il funzionamento base utilizzare questi 4 pin: VIN (3.3v o 5v), GND, SCL, SDA

## Pulsante

Ogni pulsante va collegato a GND e a un GPIO

## Micropython vs Circuitpython

Su git sono presenti due versioni del codice per gli occhiali: per esp32 e per raspberry pi pico

Micropython supporta esp32, che ha il bluetooth, ma ha meno librerie di circuitpython Per micropython potete usare la IDE Thonny

Circuitpython supporta esp32-s2 e raspberry pi pico. Quest'ultimo non ha il bluetooth. Le librerie di circuitpython permettono di usare facilmente HID usb o bluetooth e sensori come il giroscopio MPU6050 Per circuitpython potete usare la IDE Mu

## Raspberry pi pico

Scaricare l'ultima versione delle librerie di circuitpython:
https://circuitpython.org/libraries

Scaricare il firmware stable di circuitpython per raspberry pi pico:
https://circuitpython.org/board/raspberry_pi_pico/

Installare circuitpython sul raspberry pi pico:
https://learn.adafruit.com/getting-started-with-raspberry-pi-pico-circuitpython/circuitpython

Consiglio di usare l'IDE Mu per modificare il codice e vedere l'output
Il file che viene lanciato al boot è code.py

Collegare il giroscopio al raspberry tramite VCC (3v3), GND, SCL (GP27) e SDA (GP26)
Salvare il file code.py presente su git nel raspberry

**Calibrazione**
Ogni giroscopio ha un leggero errore di misurazione che va compensato
Salvare il codice nel file calibration.py (su git) nel raspberry con il nome code.py ed eseguire mostrando l'output
Lo script farà varie misurazioni durante le quali il giroscopio deve essere immobile e dritto, e calcolerà la media dei valori dando come risultato una serie di valori di offset che vanno salvati e copiati nel codice degli occhiali

From:
https://wiki.csgalileo.org/ - **Galileo Labs**

Permanent link:
**https://wiki.csgalileo.org/projects/iotaiuto/occhiali**

Last update: **2022/01/31 13:48**