

# Gulliver: A self balancing robot

Ciao a tutti, io mi chiamo Davide e sono uno studente della università di Salerno, oggi voglio presentarvi un progetto già visto ma basato interamente su un microcontrollore molto economico, ovvero lo ESP8266.

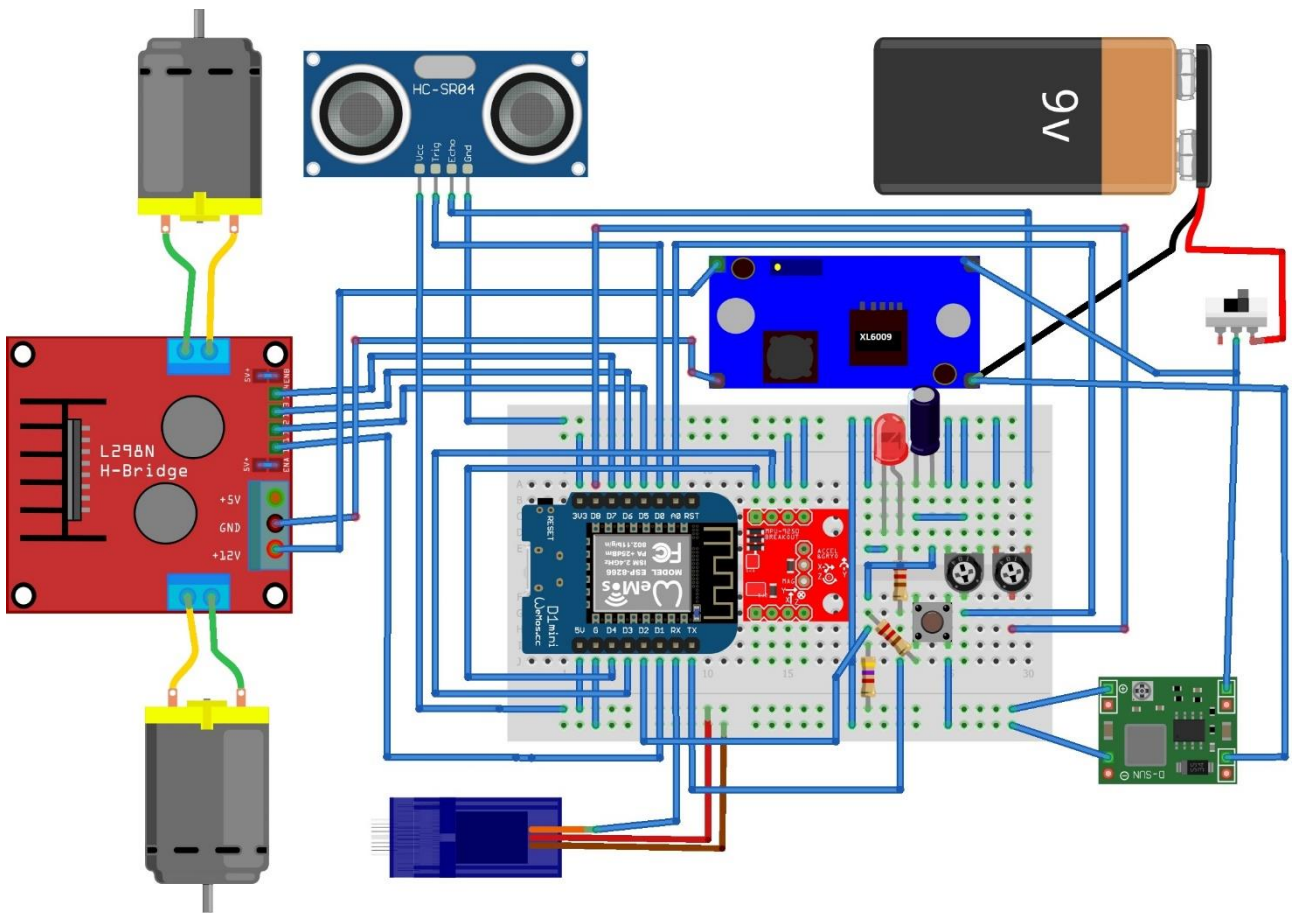
Lo ESP8266 è un microcontrollore che viene spesso usato come modulo wi-fi per arduino, ma non tutti sanno che questo microcontrollore può essere programmato esattamente come una board di arduino usando l'IDE e il linguaggio di arduino.

Ciò che differenzia questo microcontrollore da quello di arduino è che può funzionare ad una frequenza maggiore, infatti mentre un arduino 1 può funzionare a soli 16Mhz lo ESP8266 può lavorare fino a 160Mhz, ovviamente molta della potenza computazionale è sfruttata per la gestione della connessione WI-FI, infatti esso può funzionare sia come access point che come un normale dispositivo che si collega ad un modem, oppure entrambe contemporaneamente, la connessione è di tipo 802.11 b/g/n.

Ma quanto costa questo modulo? Meno di un clone di arduino 1

Esistono varie versioni di board che usano questo microcontrollore, quella usata per questo progetto è il Wemos D1 mini, una schedina leggermente più corta di un arduino nano che può essere montata su una bread board.

## Schema:



fritzing

## Lista dei componenti:

Wemos D1 mini

HC-SR04

Sevo motore da 9g SG90

MPU9250

1 interruttori

Trimmer 1Kohm

Trimmer 47Kohm

Interfaccia di potenza basata su un modulo pre-assemblato che usa un L298

2 motori N20 200RPM 12V

Batteria lipo a 2S da 610mAh

Breadboard da 400 punti con le linee di alimentazione staccati

Un bottone normalmente aperto

Condensatore elettrolitico da 10uF

Resistenza da 220Ohm

Resistenza da 4,7KOhm

Resistenza da 10KOhm

Step down converter MP1584EN regolato a 5V

Step up converter XL6009 regolato a 13,4V

### Struttura:

La struttura del self balancing robot è stata progettata in 123D Design e successivamente stampata con una stampante 3D, se siete interessati al design potete trovarlo qui:

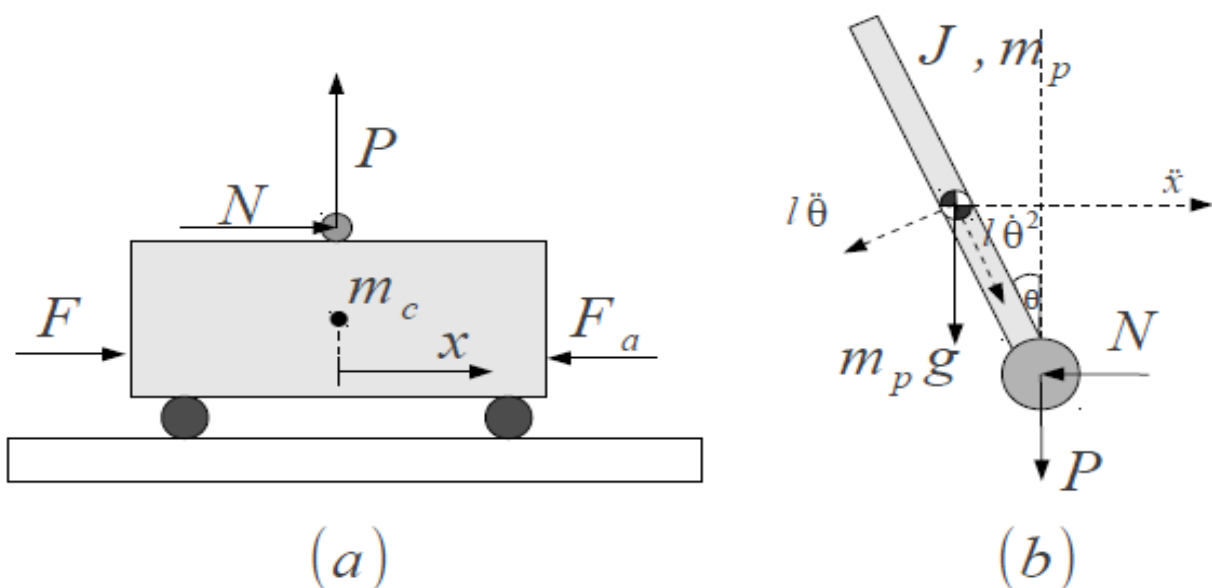
<https://www.thingiverse.com/thing:3084400>

### L'idea:

La mia idea era quella di realizzare un self balancing robot, interamente basato sul ESP8266 senza l'ausilio di microcontrollori esterni, che fosse in grado di evitare ostacoli e che potesse essere controllato da remoto tramite il WI-FI.

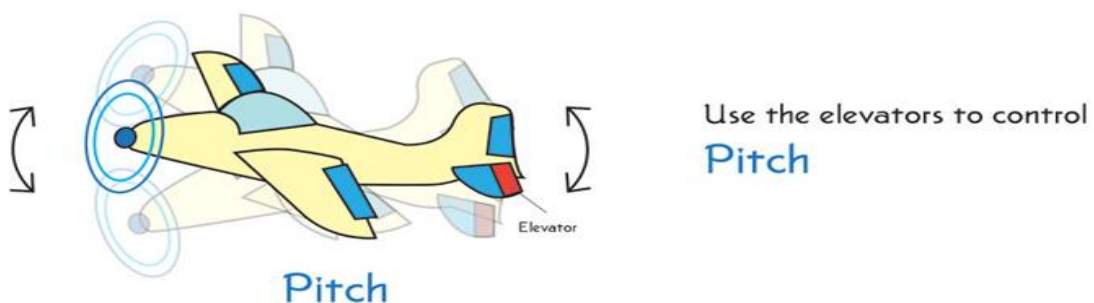
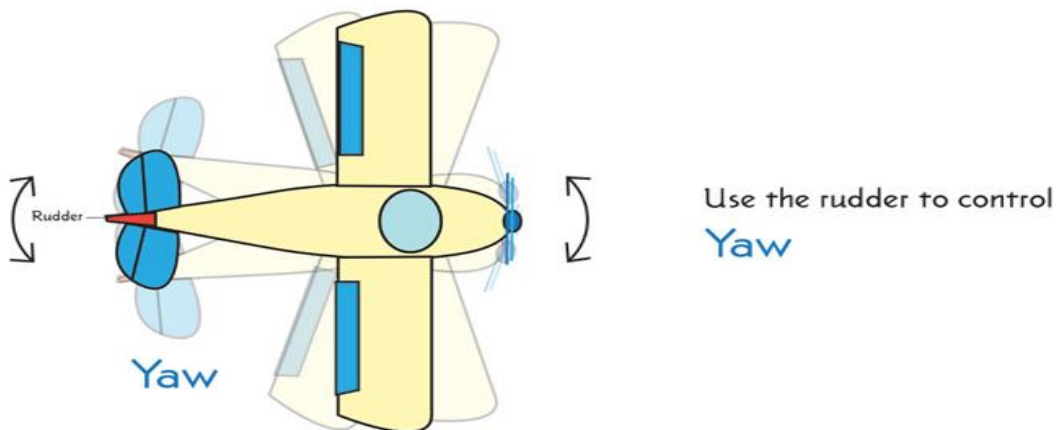
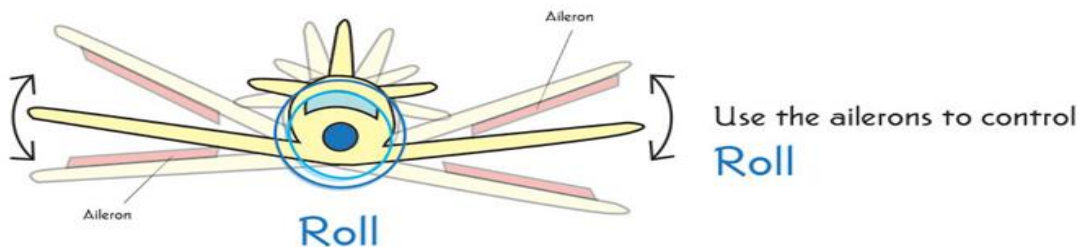
Il principio fisico alla base del self balancing robot è il pendolo inverso.

Nel caso di un pendolo, grazie alla forza di gravità porta il pendolo a fermarsi esattamente al centro mentre nel caso del pendolo inverso, la gravità è il nostro peggior nemico in quanto dobbiamo contrastarla spostando la base del robot, nei limiti del sistema che ci permette di raggiungere tale scopo ovvero i motori + sensore inerziale.



## Come fa il robot a mantenere l'equilibrio?

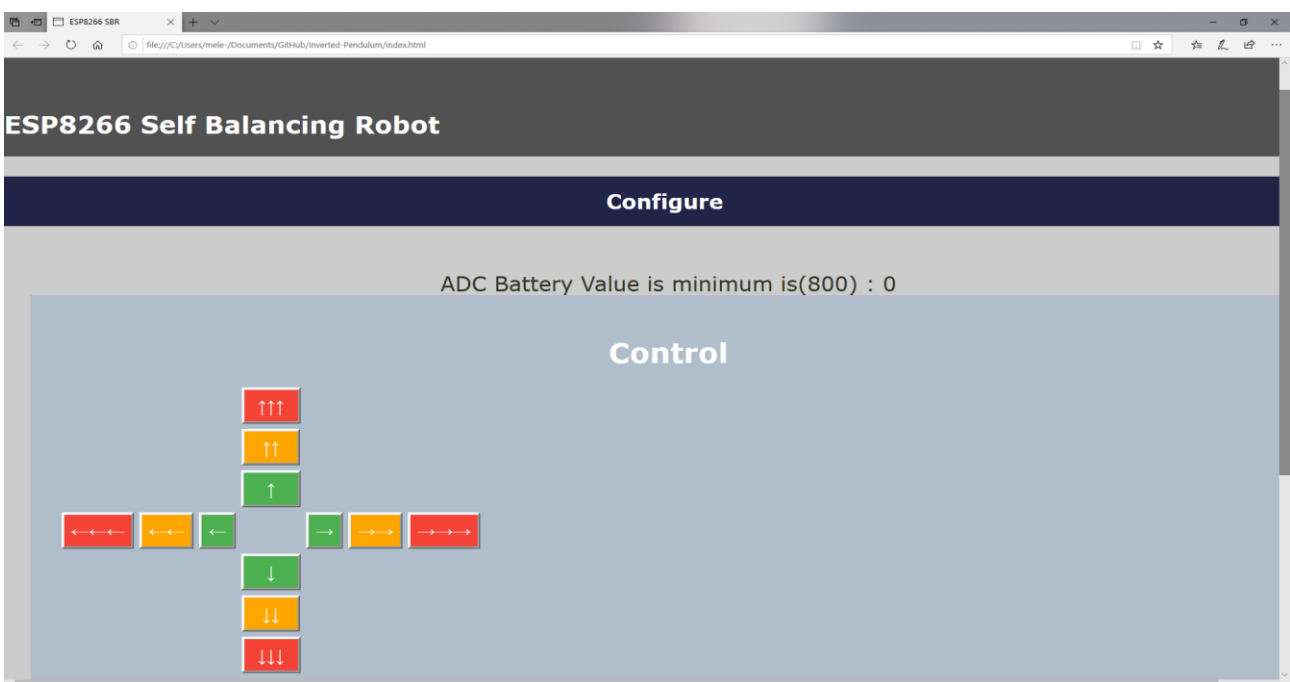
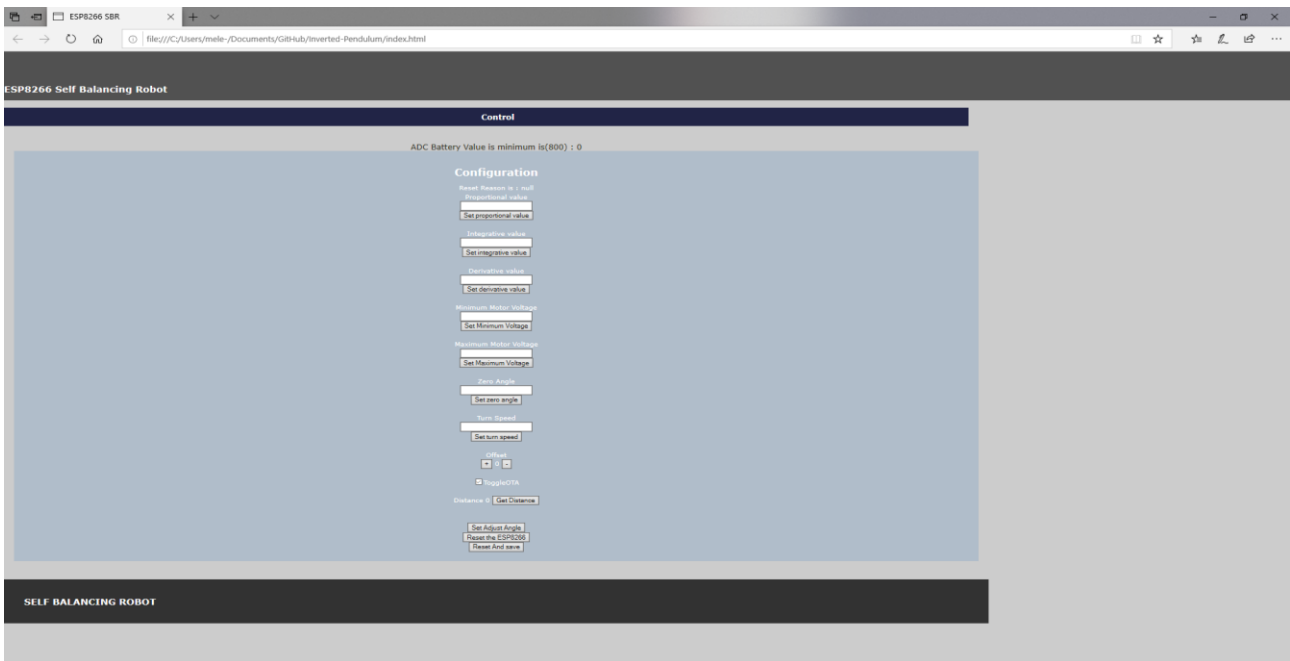
Per poter conoscere la posizione nello spazio del robot abbiamo bisogno di un sensore inerziale, nel nostro caso è un sensore a 9DOF ovvero a nove gradi di libertà, che semplificando è costituito da un accelerometro, un giroscopio ed un magnetometro, 3 assi per ognuno, nel nostro progetto useremo solo 6 gradi di libertà, ovvero l'accelerometro e il giroscopio. Il sensore in questione si chiama MPU9250. Ho fatto questa piccola introduzione perché non è possibile estrarre i dati di posizione direttamente leggendo i dati dal sensore, ma è necessario calcolarli per ottenere tre dati particolari, lo yaw il pitch e il roll, l'immagine sotto spiega bene cosa sono, questi dati sono molto usati per la regolazione di un drone ad esempio, nel nostro caso invece avremo bisogno solo del pitch, questo dato va in ingresso ad un sistema di controllo chiamato PID, composto da 3 parametri ed infine ai motori.



## Che centra questo progetto con il tema proposto per il linux day?

Questo microcontrollore è stato pensato per progetti di tipo IOT ovvero un dispositivo in grado di collegarsi ad internet o ad una rete locale, in particolare, su questo microcontrollore è presente un web server asincrono il quale gestisce le richieste dal client, sul microcontrollore è stata caricata una single web page, che usa javascript e web socket per la comunicazione con il server, infatti questo progetto il microcontrollore è configurato come access point al quale è possibile accedere al SSID ESP8266SBR e password Algorithm, infine aprendo un browser da qualsiasi dispositivo accedendo inserendo l'IP 192.168.4.1.

Questi sono degli screenshot del sito:



Adesso darò una dimostrazione pratica del funzionamento del robot, per ogni informazione o aiuto nella realizzazione del progetto o per ulteriori spiegazioni potete contattarmi, andando sul link qui sotto troverete il progetto completo e i miei contatti

# **GIT-HUB LINK:**

**<https://github.com/Dave4675/Inverted-Pendulum>**