

## Indice:

Descrizione	1
Lista materiali	1
Schema elettrico	2
Macchina a stati finiti	2,3
Task & scheduling	3,4
Librerie usate	4
Difficolta riscontrate	4
Foto & video	5

## **Descrizione**

questo progetto consiste nel creare un "radar" usando il sensore ultrasonico attaccato ad un servo motore che gira in un range di angoli. Quando il radar trova un oggetto che si avvicina troppo (per esempio più vicino di 50 cm) questo si fermerà e punterà il laser che è attaccato al secondo servo motore all'oggetto troppo vicino. Se l'oggetto sparisce dal radar cercherà prima nella zona vicina all'oggetto per assicurarsi che non si è leggermente spostato e poi tornerà a fare il suo scan normale.

## **Lista materiali**

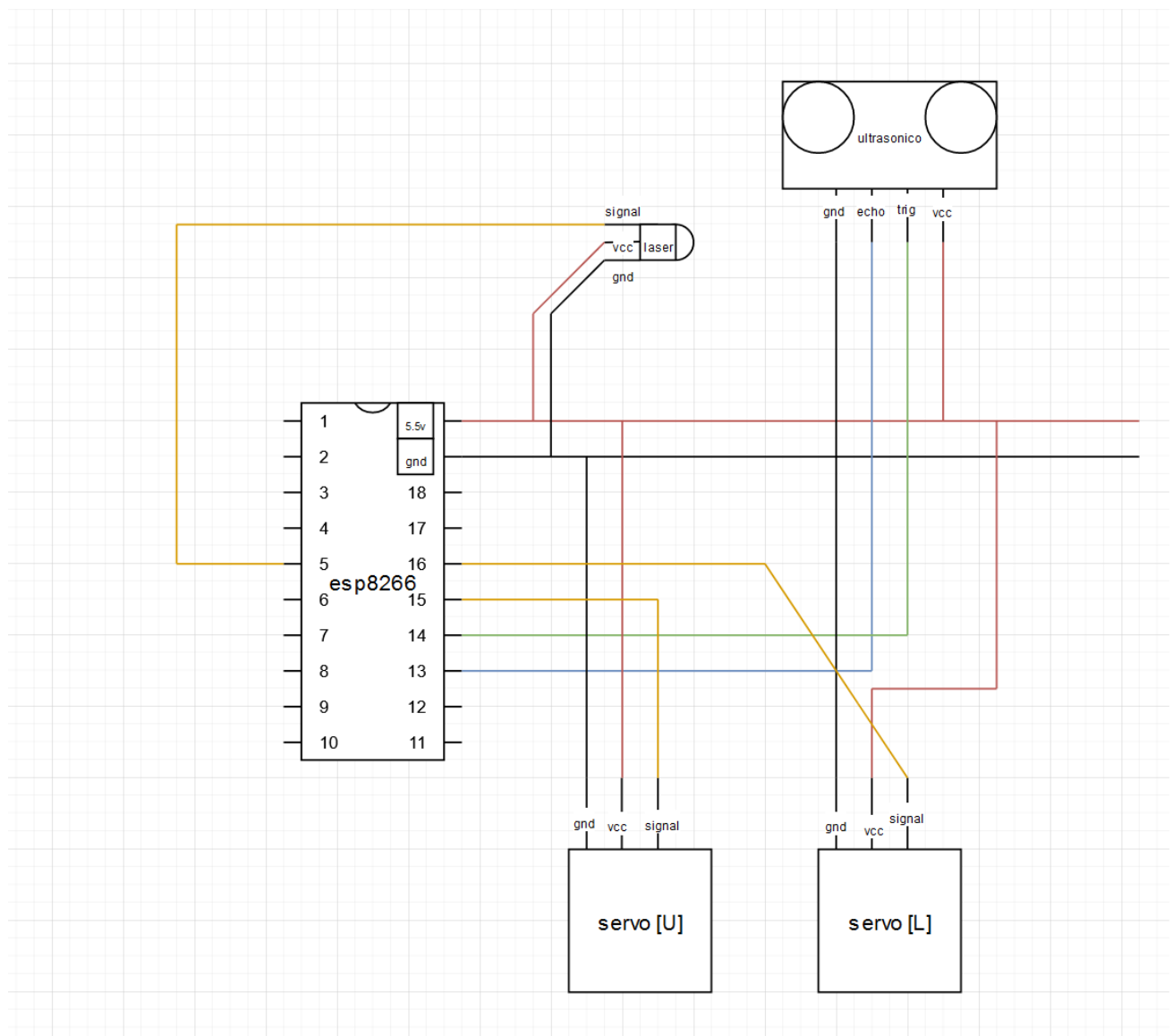
esp8266 1x

servo motori 2x

sensore ultrasonico 1x

laser rosso 1x

## Schema elettrico

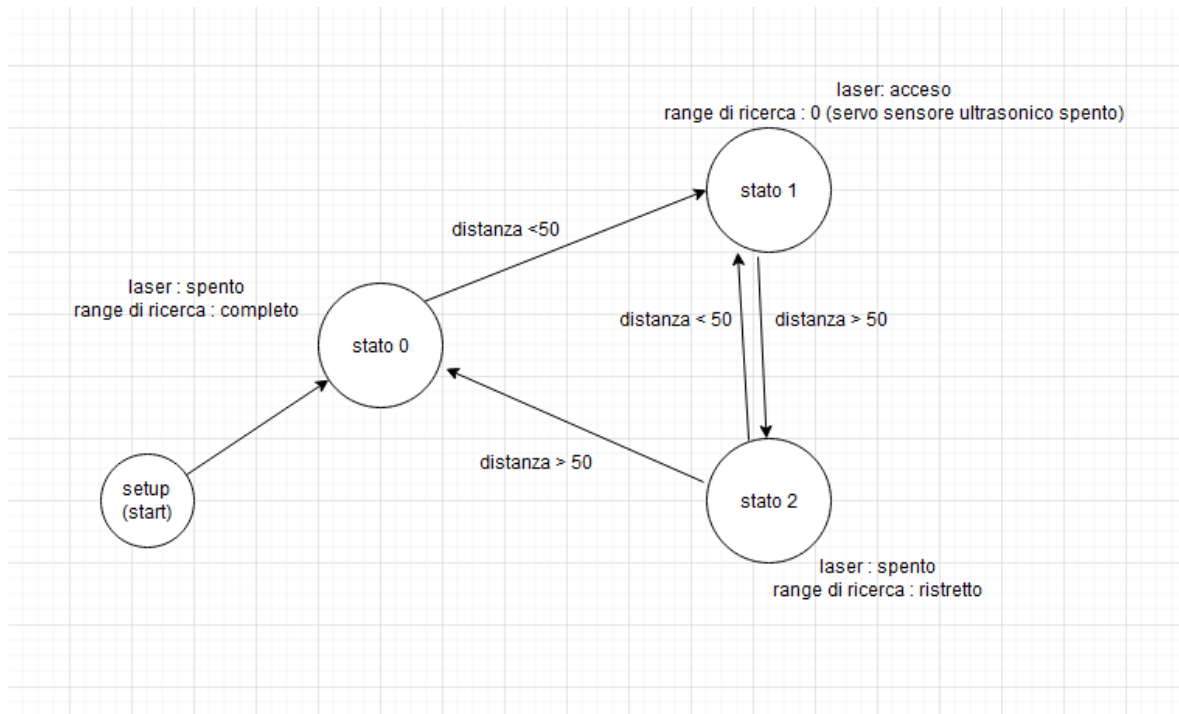


## Macchina a stati finiti

Il software è stato costruito come una macchina a stati finiti.

I 3 stati sono:

- 1) Scan : lo stato a riposo della macchina, il sensore ultrasonico gira e cerca bersagli più vicini di 50 cm. Il laser è spento.
- 2) Il sensore ultrasonico ha trovato un bersaglio, gira il laser nella direzione del bersaglio e lo accende.
- 3) Il sensore ultrasonico ha perso il suo bersaglio, inizialmente cerca in un'area ristretta che aumenta piano piano, se ritrova il bersaglio rientra nello stato 1, se non lo ritrova entra nello stato 0.



### **Task**

Negli stati 0 e 2 il sistema deve eseguire 2 task:

- 1) Girare il servo del sensore ultrasonico
- 2) Misurare la distanza usando il sensore ultrasonico

La task che gestisce il movimento del servo deve essere veloce da eseguire per permettere l'eseguimento della task che misura la distanza in modo da non creare zone cieche.

Inoltre, le 2 task hanno anche il compito di gestire la macchina a stati finiti.

Scan\_task: viene eseguita ogni 250 ms, esegue la funzione scan\_task\_func.

-scan\_task\_func:

-stato 0: gira il servo di 3 gradi ogni volta che viene eseguita, se raggiunge l'angolo minimo o massimo cambia direzione.

-stato 1: ferma il servo del sensore ultrasonico.

-stato 2: gira il servo in un'area ristretta cercando di riprendere il bersaglio, ogni volta che viene eseguito gira il servo del sensore ultrasonico e ogni 100 volte che viene eseguito aumenta l'area di ricerca. Se l'area di ricerca è grossa quanto il range di ricerca dello stato 0 rientra nello stato 0.

Get\_target\_task: viene eseguita ogni 100 ms, esegue la funzione get\_target\_task\_func.

- get\_target\_task\_func:

sempre: misura la distanza dall'oggetto più vicino. Se trova qualcosa più vicino di 50 cm

entra nello stato 1.

-stato 1: sposta il servo del laser nella direzione dell'oggetto vicino e accende il laser.

Se perde il bersaglio entra nello stato 2.

### **Librerie**

Servo.h

TaskScheduler.h (con \_TASK\_PRIORITY)

### **Difficoltà riscontrate**

La difficoltà principale riscontrata è stata scrivere la funzione

```
scan(Servo myServo, int* position, bool* direction, int* max, int* min)
```

In modo che non risultasse bloccante per la task : Get\_target\_task e farla sembrare comunque abbastanza fluida in modo da creare l'illusione del multitasking.

Anche trovare un modo per attaccare il sensore ultrasonico al servo in modo verticale non è stato facile e tuttora risulta "ballerino" a causa del peso dei cavi.

## Foto & video

<https://youtube.com/shorts/WL-3jvF4G0s?feature=share>

