# Impostazioni e taratura

Una volta montata ARI V3 occorre configurarla per fargli conoscere come è il suo corpo. Le informazioni che servono sono:

La *Carreggiata o Baseline*: distanza tra le ruote in mm



Lo *Sviluppo della ruota* in mm, la sua circonferenza che è pari a 3.14 volte il diametro.

I *ppr* (Pulses Per Revolution) degli encoder. Il numero di tacche che ha il disco dell’encoder.

Con questi numeri si ricavano i numeri del robot.

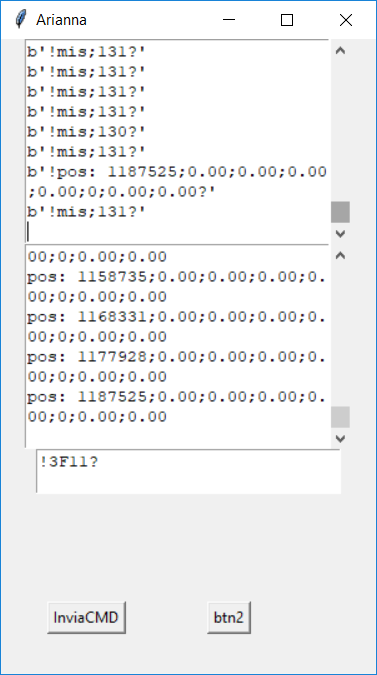
Baseline = Carreggiata

Giro Ruota = sviluppo ruota[mm]/(4\*ppr)

ED = 1

ED\_BASE = 1

Calcolati questi numeri tramite la CLI[[1]](#footnote-1) vi imposto su ARI V3. Scritto il comando lo si invia con *InviaCmd*.



*I comandi hanno la seguente forma:*

*il primo numero indica l’azione. 3 significa scrittura, 1 lettura.  
dal terzo carattere si indica quale azione fare, nell’esempio sotto F1 significa scrivi ED\_BASE  
i caratteri successivi contengono l’argomento del comando, di norma il valore.*

*Ad ogni comando inviato viene ritornata una risposta “comprensibile”.  
es: invio “3F11” mi viene ritornato “F1\_ED\_BASE: 1.000000”*

Il comando è il seguente:

Fnxxx: imposta dei parametri del robot. "n" indica quale parametro, "xxx" è il valore.

F0xx ED

F1xx ED\_BASE

F2xx BASELINE mm

F3xx GIRO\_RUOTA mm = sviluppo ruota[mm]/(4\*ppr)

N.B. questi valori vanno attivati con un 3E3

Esempio:

carreggiata = 220 mm  
diametro ruota = 50 mm  
ppr = 20

sviluppo ruota = 3.14\*50 = 157 mm

Giro Ruota = sviluppo ruota[mm]/(4\*ppr) = 157/(4\*20) = 1.9625 mm

Comandi da inviare:

E2 carica i valori di default.  
E3 attiva i valori  
E0 salva i valori in E2prom

I valori di default sono quelli sotto riportati.  
  
dovendo modificare i valori sotto sono riportati i singoli comandi.

ED = 1 inviare **F01**. **N.B. 0 è “zero” non “O” maiuscolo**ED\_BASE = 1 inviare **F11**   
BASELINE = 220 inviare **F2220**   
GIRO\_RUOTA =1.9625 inviare **F31.9625**

inviare E3 per attivare i valori. La risposta conterrà la lista dei parametri attiva. Questo parametro serve anche per mostrare i parametri in uso.

E0 per salvarli nella EEprom[[2]](#footnote-2) di ARI V3. Cosi facendo verranno ricaricati ad ogni accensione.

# Verifica dei dati.

Per vedere se i numeri “quadrano” muoviamo ARI. Facciamogli fare un movimento e vediamo come va. Mi serve un metro per misurare lo spazio percorso.

## Verifica distanza

Accendo e spengo ARI.

**3D1000** definisco un percorso di 1000 mm.  
**3R1** la faccio muovere in linea retta. ARI dovrebbe partire e fermarsi dopo un metro circa.

Se la distanza percorsa non è corretta aggiusto GIRO\_RUOTA opportunamente.. formula

## Verifica centratura

È probabile che il percorso non sia rettilineo e tenda a curvare verso destra o sinistra. Il fenomeno è dovuto alle ruote che pur simili hanno differenze minime. Aggiustando il parametro ED si corregge questo comportamento, ED infatti modifica il diametro delle ruote sinistra e destra rendendoli diversi.   
ED si modifica a piccoli passi, qualche percento alla volta, per esempio 1.01 oppure 0.99. Se lo si aumenta ARI va verso destra, diminuendolo ruota verso sinistra.

Per attivare il valore ED = 1.02 la sequenza è:

passo 1: impostazione parametro

**3F01.02** per impostare il nuovo valore 1.02  
**3E3** per attivare i valori, la risposta

Passo 2:Test

**3D1000** definisco un percorso di 1000 mm.  
**3R1** la faccio muovere in linea retta.

Muove verso destra? Diminuisco ED e torno al passo 1   
Muove verso sinistra? aumento ED e torno al passo 1

Muove diritta? Proseguo e salvare i dati

**3E0** salvo i dai in memoria EEprom.

Ex: EEprom. Esegue operazioni su dei parametri di taratura. Vedi procedura DataEEprom.

E0 SCRIVI i parametri in E2prom,

E1 LEGGI i parametri in E2prom,

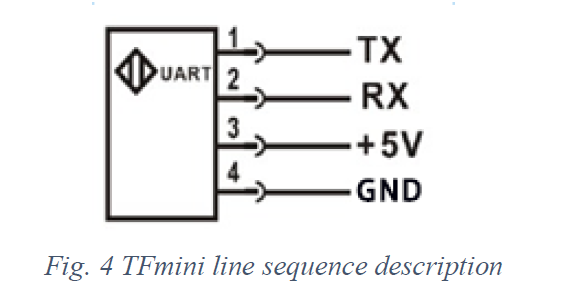
E2 rispristina in valori di DEFAULT,

E3 attiva e mostra i parametri CORRENTI,

# Tabelle di collegamento

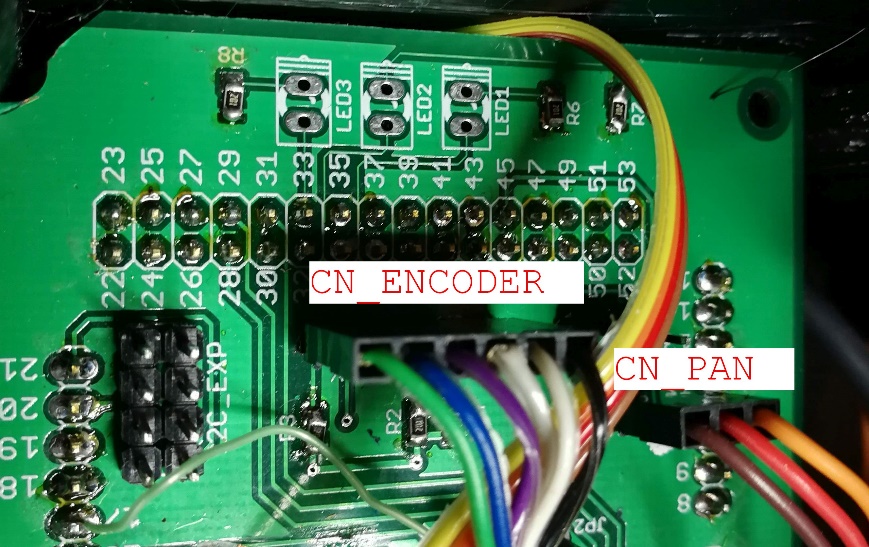
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **J\_ENC** | **filo** | **JbasettaMotore** | |  |
| 1 | Nero | Emitter-Katode | Dx | Gnd |
| 2 | Bianco | Collector | Dx | Encoder |
| 3 | Grigio | Anode | Dx | Alimentazione diodo |
| 4 | Viola | Emitter-Katode | Sx | Gnd |
| 5 | Blu | Collector | Sx | Encoder |
| 6 | verde | Anode | Sx | Alimentazione diodo |
|  |  |  |  |  |
|  |  |  |  |  |

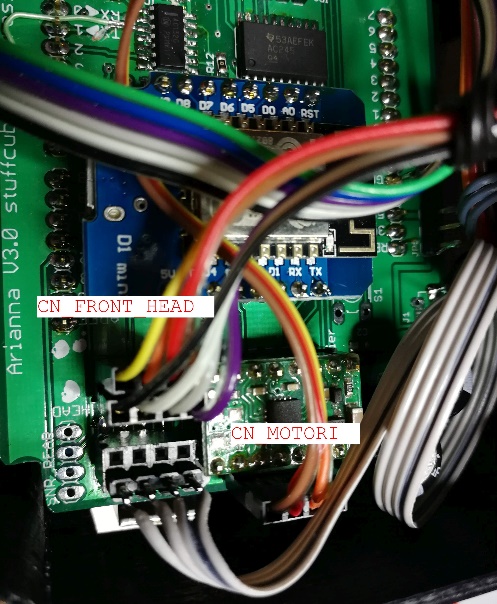
|  |  |  |
| --- | --- | --- |
| CN\_FRONT\_HEAD | | |
| 1 | Marrone | Rosso 5V |
| 3 | Rosso | Nero Gnd |
| 5 | Arancio | Bianco Rx1 (lidar tx) |
| 7 | Giallo | Verde Tx1 (lidar rx) |
| 2 | Viola | IR rx pin |
| 4 | Grigio | Nc |
| 6 | Bianco | Led pointer |
| 8 | Nero | 3V3 IR power supply |



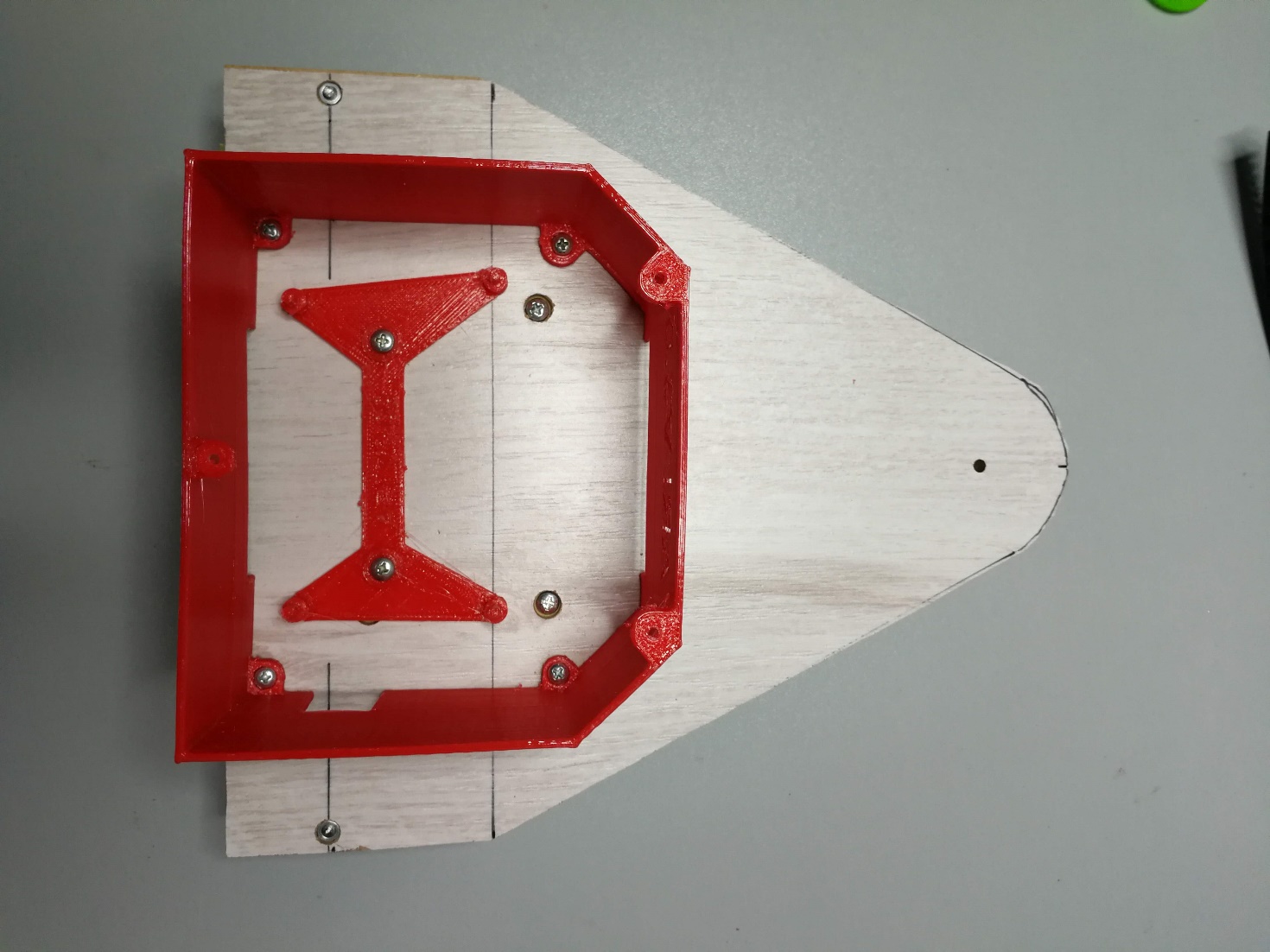
Collegamento motori

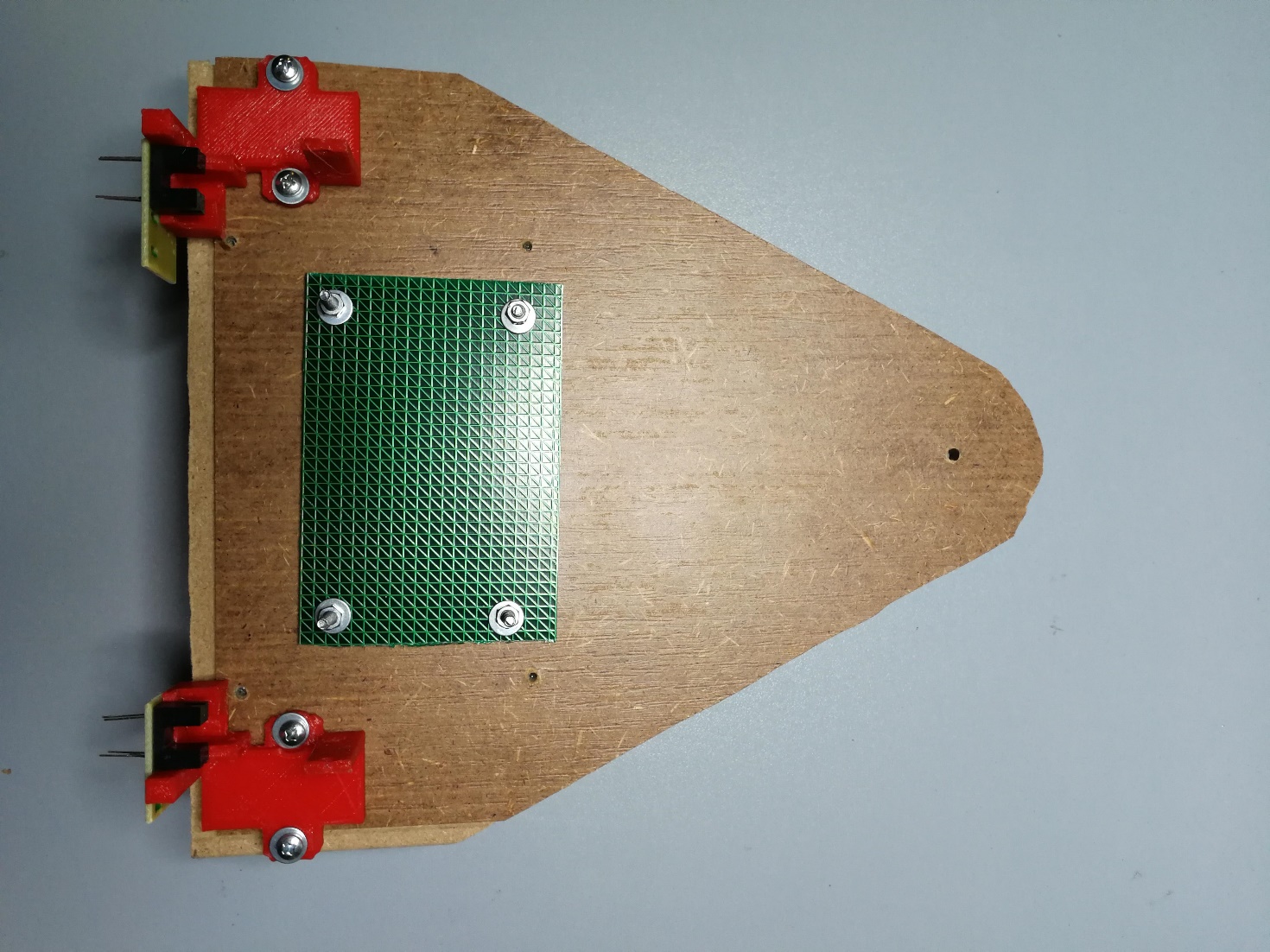
|  |  |  |
| --- | --- | --- |
| MOTORS | | |
| 1 | Marrone | Sotto sx |
| 2 | Rosso | Sopra sx |
| 3 | Arancio | Sopra dx |
| 4 | Giallo | Sotto dx |





# Sequenza montaggio

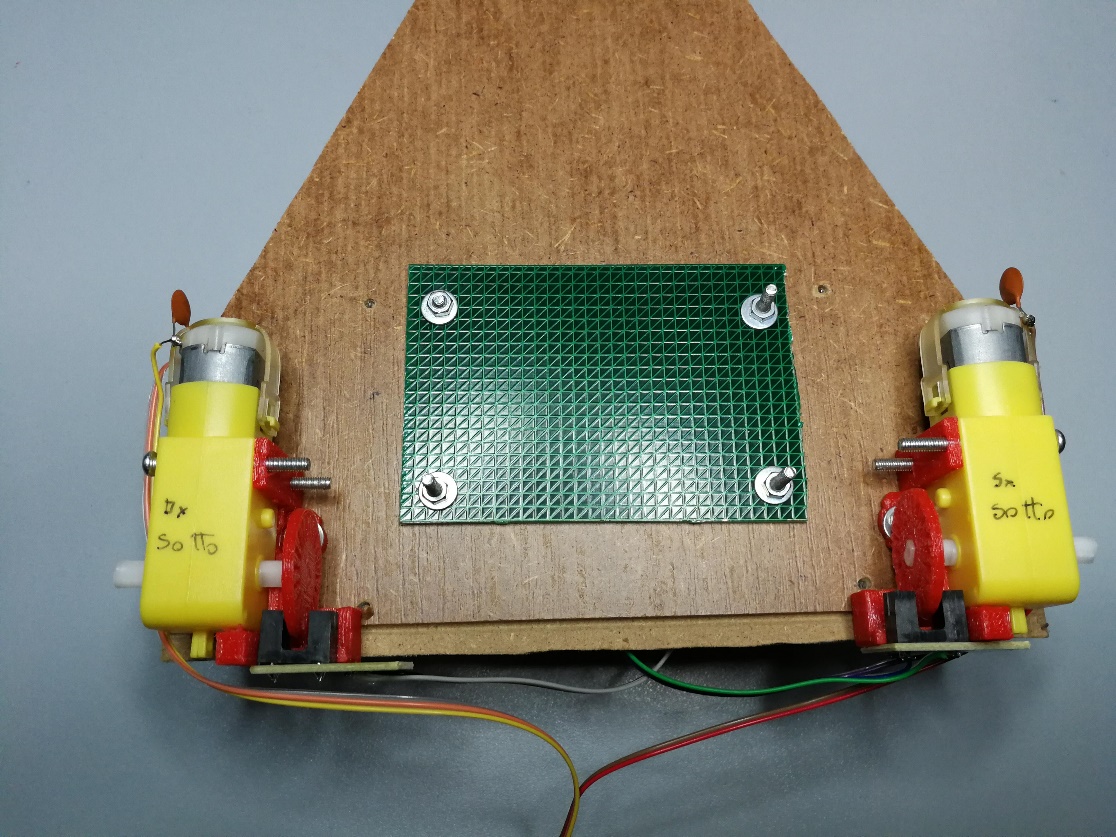


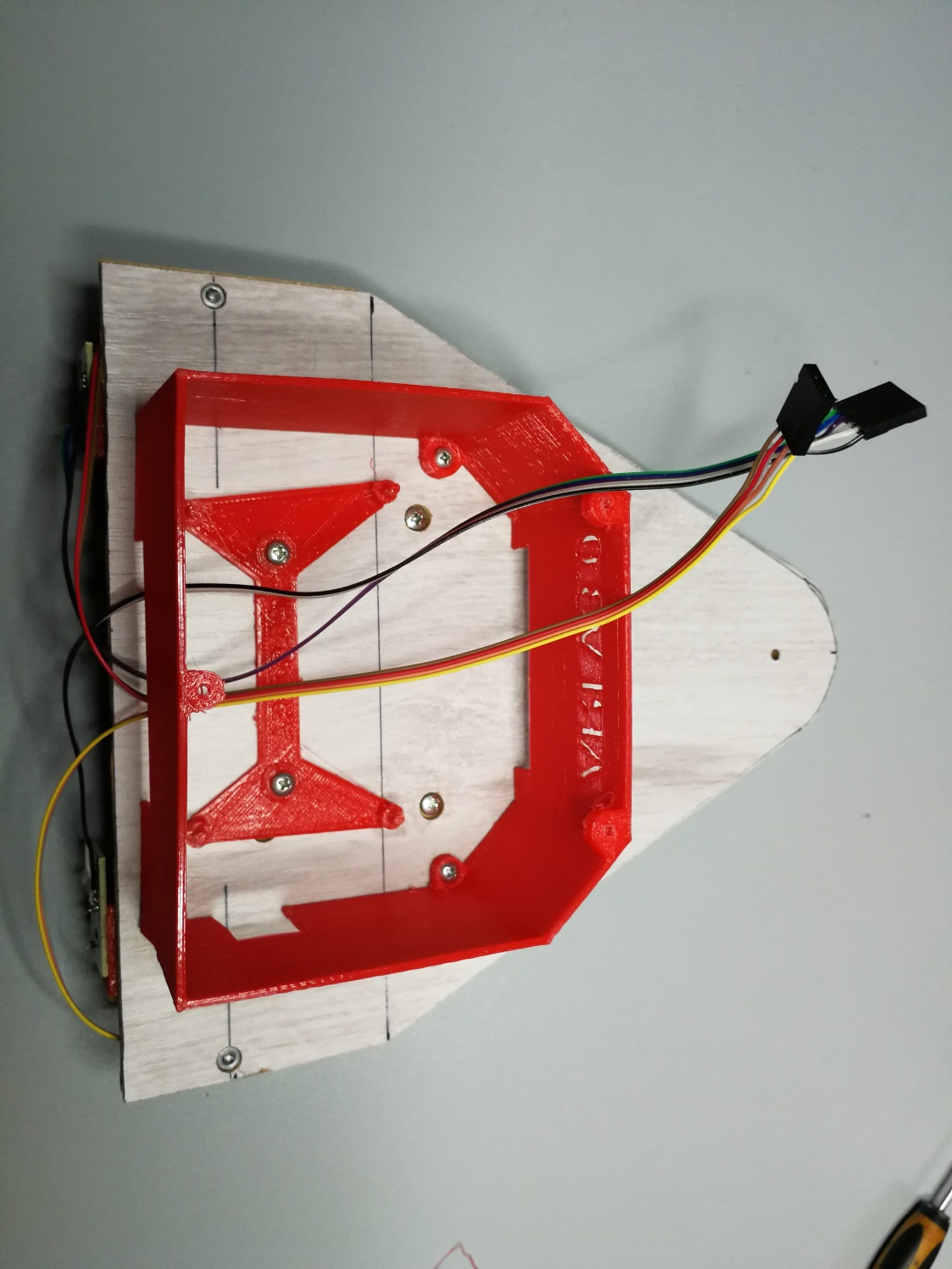


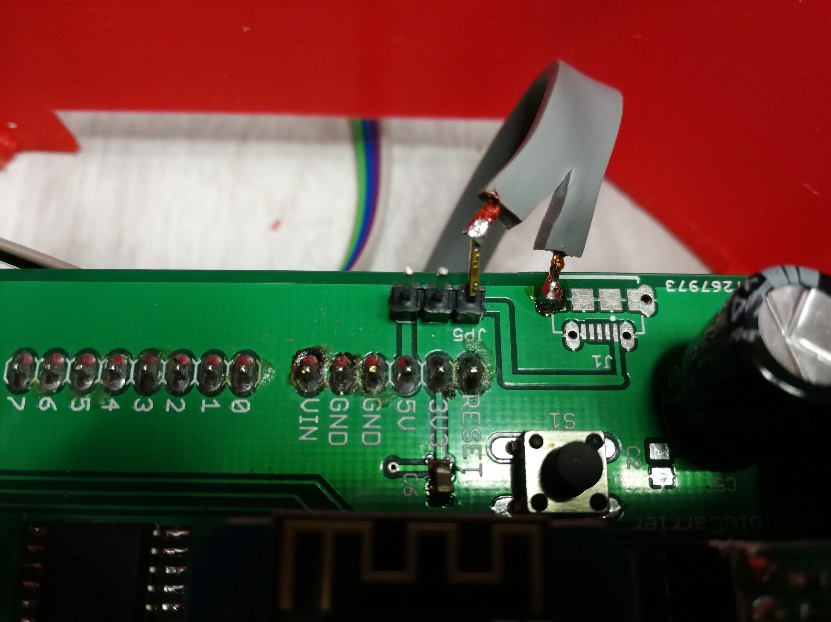
|  |  |
| --- | --- |
|  |  |



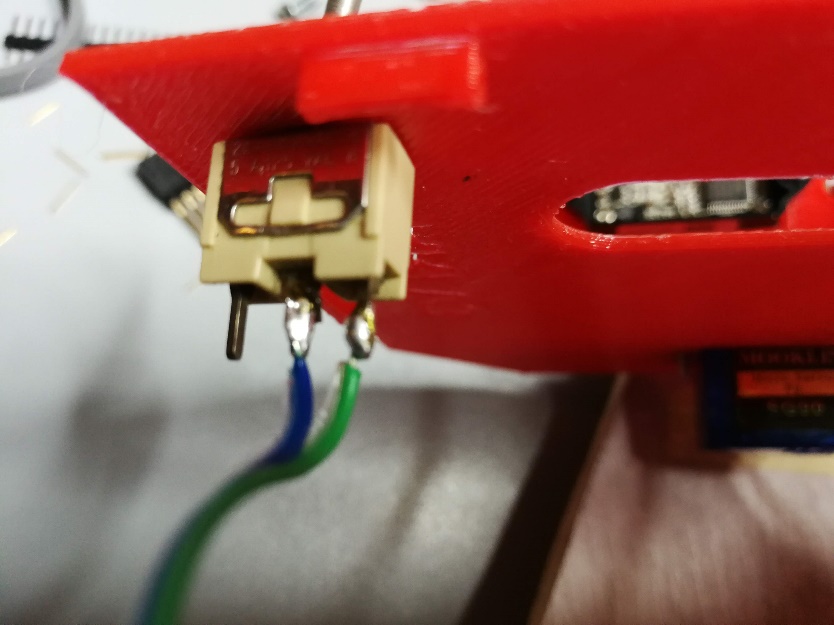
Indicare il nottolino che deve stare all’interno











1. CLI, Command Line Interface. È la finestra con la quale inviamo I comandi digitandoli sulla tastiera. [↑](#footnote-ref-1)
2. EEprom oppure E2prom. Electrically Erasable Programmable Read-Only Memory. Una memoria dove i dati possono essere letti, scritti e conservati anche senza tensione, cioè a chip spento. [↑](#footnote-ref-2)