

Esercitazione in Laboratorio di Robotica

1. Accendere il PC e l'unità di potenza (tasto sul retro)
2. Attendere la fine del boot del PC finché non appare il prompt di MS-DOS
3. Portarsi nella directory $C : \backslash ROBOTICA \backslash ROBOT99 \backslash SCORB \backslash$ digitando:
cd robot99
cd scorb
4. Lanciare l'eseguibile sbdemo.exe digitando *sbdemo*
5. Una volta che il programma si è avviato ed è apparsa la schermata principale, in basso è riportato il menu dei comandi disponibili:

- I tasti da 1 a 5 permettono di selezionare il giunto che si vuole muovere
- I tasti + e - permettono di modificare lo step (in alcune postazioni conviene portarlo a 1, per default è 3)
- I tasti shift (destra e sinistra) permettono di muovere il giunto selezionato (premerli per **breve tempo** per evitare escursioni eccessive del robot!)
- Sulla schermata principale sono riportati i passi encoder dei vari motori: il tasto F1 permette di mandare il motore del giunto selezionato al passo encoder desiderato. Con il tasto *return* si azzerano tutti i passi encoder. ATTENZIONE: i passi encoder dei vari motori non rappresentano direttamente gli angoli di giunto per due motivi. Innanzitutto occorre trovare la costante di proporzionalità tra passi encoder e angoli, diversa in generale per ogni giunto. Secondo, per via della struttura meccanica dello Scorb, la dipendenza di un angolo dai passi encoder è piuttosto complessa. Si riporta sotto la corrispondenza (in entrambi i versi) tra passi encoder p_i , $i = 0, 1, \dots, 4$ ed angoli di giunto θ_i , $i = 1, 2, \dots, 5$, in cui le costanti vanno determinate (vedi punto successivo):

$$\begin{array}{ll} \theta_1 = -\alpha \cdot p_0 & p_0 = -\theta_1/\alpha \\ \theta_2 = -\beta \cdot p_1 & p_1 = -\theta_2/\beta \\ \theta_3 = \beta(p_1 + p_2) & p_2 = (\theta_2 + \theta_3)/\beta \\ \theta_4 = -\beta \cdot p_2 + \gamma(p_3 - p_4) & p_3 = 0.5[(\theta_2 + \theta_3 + \theta_4)/\gamma + \theta_5/\delta] \\ \theta_5 = \delta(p_3 + p_4) & p_4 = 0.5[\theta_5/\delta - (\theta_2 + \theta_3 + \theta_4)/\gamma] \end{array} \iff$$

Il valore di p_5 è invece legato al movimento di apertura e chiusura della pinza (ottenibile pigiando rispettivamente i tasti O e C) e non rientra tra gli scopi della presente esercitazione.

6. Determinare innanzitutto le costanti positive α , β , γ e δ presenti nelle formule precedenti (si può far uso del goniometro ma anche muovere i vari giunti di angoli noti facili da valutare, come per esempio $\pi/2$). È probabile che si troverà $\delta = \gamma$.
7. Misurare la lunghezza dei vari link del robot (delle grandezze cioè che compaiono nella tabella di Denavit Hartenberg dello Scorb). I valori riportati nella dispensa potrebbero non essere precisi o variare da robot a robot.
8. Portare il robot nella configurazione in cui tutti gli angoli di giunto sono nulli (secondo la convenzione utilizzata a lezione), azzerare i passi encoder (pigiando *return*) e fissare quindi anche il sistema di riferimento della base L_0 (sempre in accordo con quello definito nella dispensa).
9. Muovere ora il robot a piacere e verificare se, in base alla lettura dei passi encoder a schermo, applicando le equazioni precedenti di conversione passi encoder \rightarrow angoli e quelle della cinematica diretta, la pinza del robot ha raggiunto la posizione rilevabile a mano col metro. Eventualmente verificare anche l'orientamento della pinza (per esempio verificare se l'asse z_5 è diretto così come ci si aspetta dalle formule).
10. Si scelga una posizione e un orientamento (in termini di angolo di beccheggio β_d ed eventualmente anche di angolo di rollio ω_d) desiderati per la pinza. Applicare le formule della cinematica inversa e quelle di conversione angoli \rightarrow passi encoder. Vedere se portando ciascun motore al passo encoder determinato (a tale scopo si può usare la funzione F1), la pinza raggiunge la configurazione desiderata.
11. Potrebbe anche essere interessante modificare il valore del parametro K_p (usando il comando F2), per default pari a 30. Tale parametro ha lo stesso significato introdotto negli sketch di Processing: è la costante della legge di controllo proporzionale. Valori troppo elevati possono portare ad oscillazioni e all'instabilità (schiacciare il pulsante rosso!). Valori troppo piccoli possono non essere sufficienti a far muovere il robot.