



UNIVERSITA' degli STUDI di ROMA
TOR VERGATA



Facoltà di Ingegneria
Corso di Laurea in Ingegneria dell'Automazione

Relazione Gruppo 7
alias
Gruppo Frisbee



Corso di Robotica con Laboratorio A.A. 2006-2007

Relazione del lavoro compiuto su SCORBOT ER-V
“Impariamo a Scrivere”

Francesco Mecocci, Giacomo Perantoni, Matteo Lugliolini

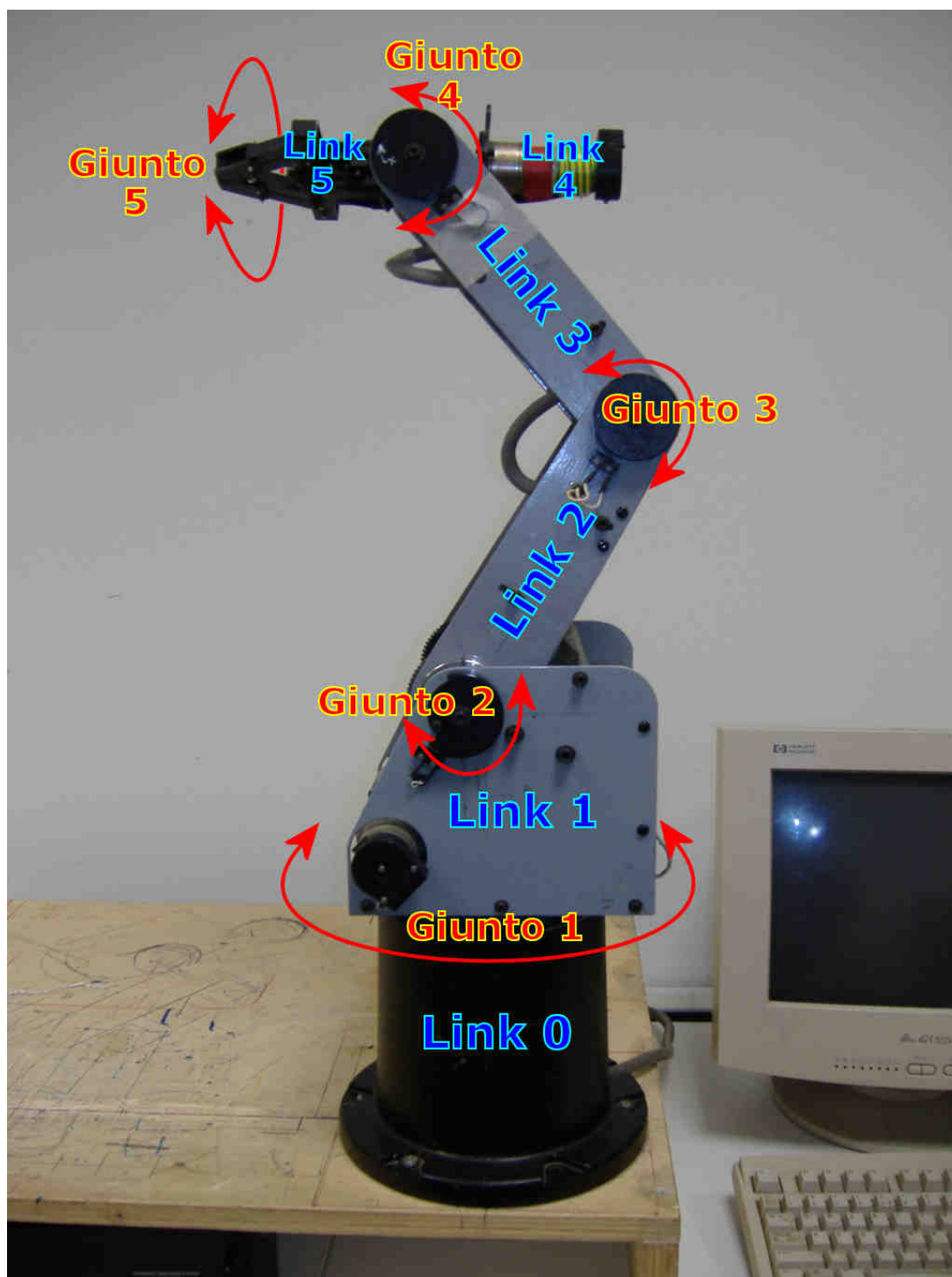
Indice

1. Introduzione.....	pag. 3
2. Cinematica diretta.....	pag. 4
2.1. Problema della cinematica diretta.....	pag. 4
2.2. Scelta dei sistemi di riferimento.....	pag. 7
2.3. Trasformazioni di coordinate omogenee.....	pag. 10
3. Sistema meccanico e di controllo.....	pag. 14
3.1. Movimenti dei giunti.....	pag. 14
3.2. Controllo del robot.....	pag. 15
4. Calcolo passi encoder – radianti.....	pag. 17
4.1. Rapporti encoder – radianti.....	pag. 17
4.2. Angoli di Home.....	pag. 20
5. Misurazioni effettuate.....	pag. 25
5.1. Parametri di Link.....	pag. 25
5.2. Parametri di Giunto.....	pag. 27
5.3. Rapporti “Passi encoder - Angoli”.....	pag. 27
5.4. Risultato delle misurazioni.....	pag. 28
6. Cinematica inversa.....	pag. 29
6.1. Semplificazioni.....	pag. 29
6.2. Calcolo degli Angoli di Giunto.....	pag. 30
6.3. Campo di Applicabilità.....	pag. 34
7. Controllo della Posizione del Robot.....	pag. 35
7.1. Controllo sui limiti dei valori per gli angoli di D-H.....	pag. 35
7.2. Controllo sull’impatto con il piano di lavoro.....	pag. 36
7.3. Controllo sull’impatto con il cilindro di base.....	pag. 37
7.4. Controllo sull’impatto con il link 2.....	pag. 38
8. Inseguimento di traiettorie.....	pag. 40
8.1. Equazioni parametriche di una retta nello spazio.....	pag. 40
8.2. Normalizzazione della distanza tra punti successivi.....	pag. 40
8.3. Inseguimento di traiettorie su un piano.....	pag. 42
8.4. Traiettorie rettilinee su un piano.....	pag. 45
8.5. Traiettorie circolari su un piano.....	pag. 45
8.5.1. Equazioni parametriche di traiettorie circolari.....	pag. 45
8.5.2. Normalizzazione lunghezza arco associato a due punti successivi.....	pag. 46
8.6. Esempio di scrittura di una lettera.....	pag. 47
9. Implementazione della funzione di cinematica diretta.....	pag. 49
10. Implementazione della funzione di cinematica inversa.....	pag. 51
11. Descrizione dello pseudolinguaggio di movimentazione del robot.....	pag. 52
12. Controllo Manuale.....	pag. 55
13. Preparazione per la dimostrazione.....	pag. 56
14. Appendice.....	pag. 58
14.1. Sistemi coordinati tridimensionali destrorsi.....	pag. 58
14.2. Convenzione sulla misura degli angoli nello spazio.....	pag. 58
14.3. Angoli di Eulero.....	pag. 59
14.4. Coordinate Omogenee.....	pag. 60
14.5. Convenzione di Denavit - Hartenberg.....	pag. 61
15. Magic Moments.....	pag. 64

1. Introduzione

Un manipolatore meccanico o robot è una sequenza di segmenti rigidi, detti Link, connessi da Giunti prismatici, o di rotazione associati rispettivamente ad uno spostamento lineare, o angolare di due Link adiacenti.

Lo SCORBOT-ER V è un manipolatore meccanico o robot costituito da cinque Link mobili (sei Link se si considera anche il Link fisso iniziale), collegati da cinque Giunti rotazionali, come evidenziato in figura. I Giunti prendono il nome di BASE, SPALLA, GOMITO, BECCHEGGIO, ROLLIO o rispettivamente Giunti 1, 2, 3, 4, 5.



2. Cinematica diretta

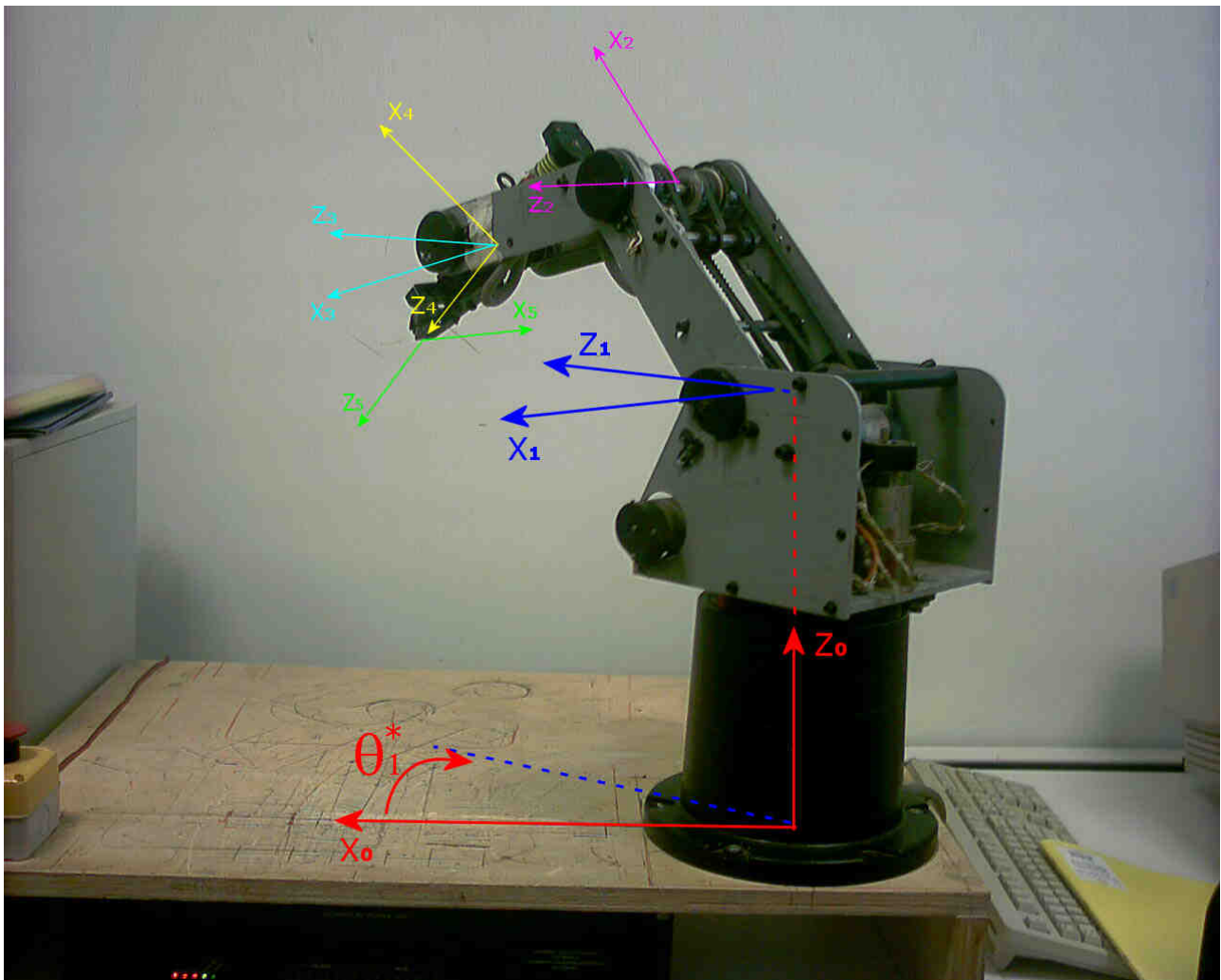
2.1 Problema della cinematica diretta

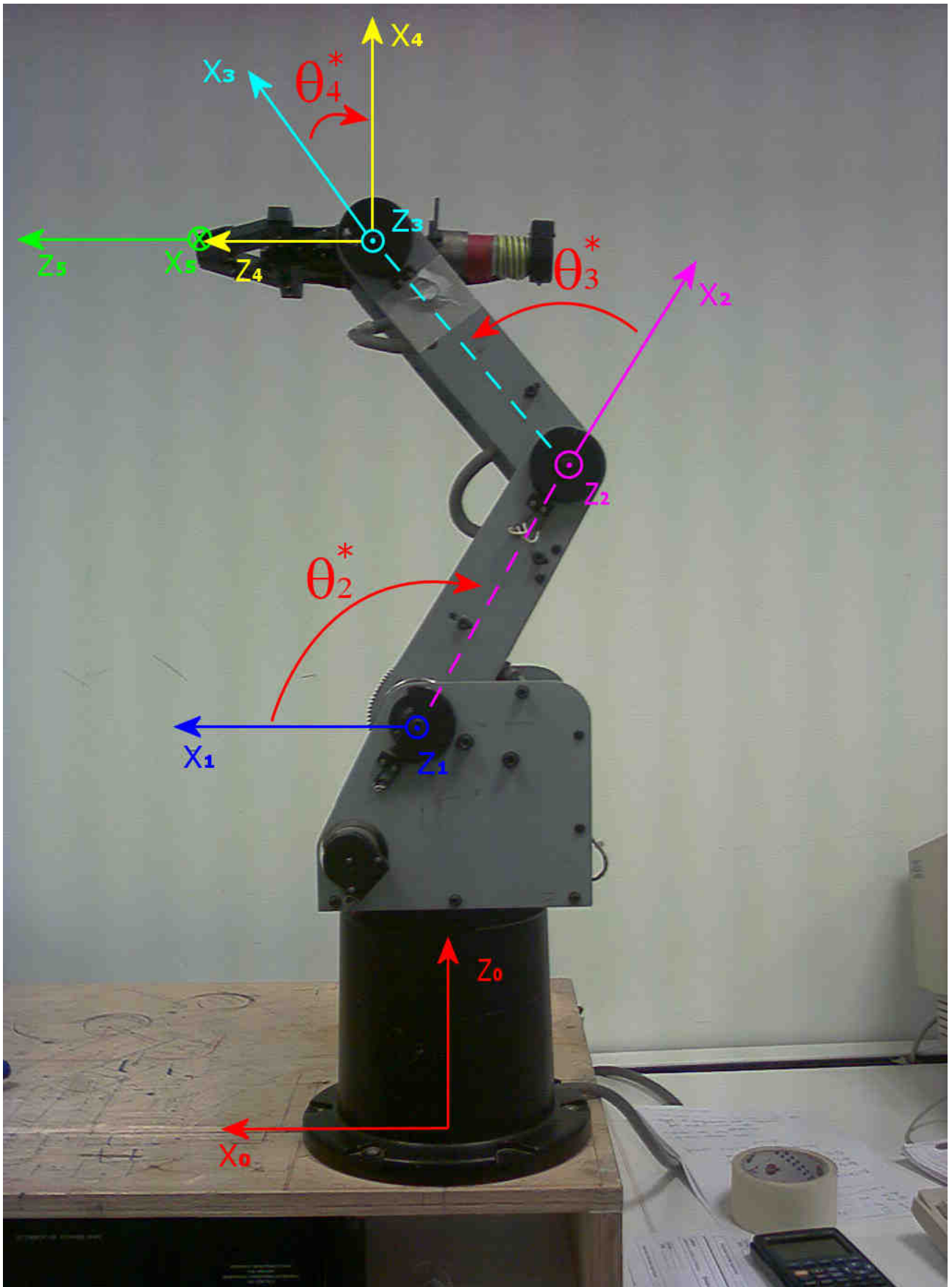
Lo SCORBOT-ER è un manipolatore provvisto di cinque gradi di libertà, tanti quanti sono i suoi Giunti. Ogni grado di libertà può essere rappresentato da una variabile reale. Si può quindi definire per lo SCORBOT-ER il proprio spazio dei Giunti Q nel quale è definito il vettore:

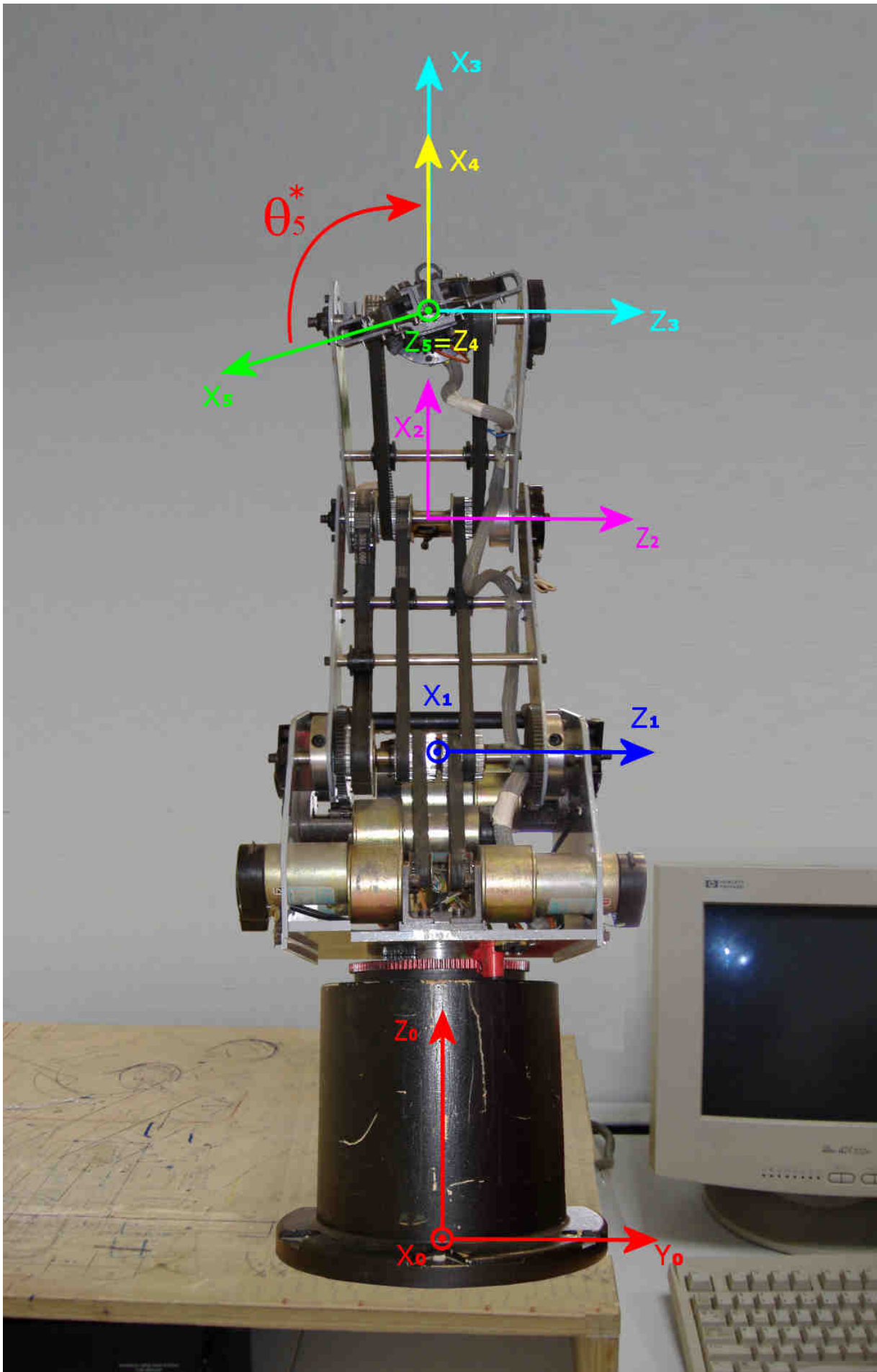
$$q = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \end{bmatrix}$$

Le componenti del vettore q sono tutte variabili reali angolari, in quanto i Giunti dello SCORBOT-ER sono tutti rotazionali. Queste cinque variabili $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ rappresentano i cinque gradi di libertà del robot. La scelta dei sistemi di coordinate rispetto ai quali definire gli angoli $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ è stata operata mediante la convenzione di Denavit-Hartenberg (D-H). Secondo questa convenzione viene associato un sistema di coordinate cartesiane destrorso ad ogni Link.

(si rimanda alla sezione Appendice per approfondimenti sugli argomenti Sistemi di coordinate destrorsi, e Convenzione di Denavit-Hartenberg)







Il problema della cinematica diretta è quello di esprimere la posizione e l'orientamento del sistema di riferimento associato al Link terminale del robot, rispetto al sistema di riferimento iniziale. Posizione ed orientamento del Link terminale, detto anche Link di effettore, saranno funzioni degli angoli $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$. Per quanto riguarda la posizione del Link di effettore, vanno caratterizzate le coordinate, rispetto a $(Oxyz)_0$, dell'origine del sistema di riferimento $(Oxyz)_5$. Per quanto riguarda l'orientamento del Link di effettore, questa va caratterizzata mediante una opportuna terna di angoli che descrive la rotazione complessiva di $(Oxyz)_5$ rispetto a $(Oxyz)_0$. Per far questo sono state formalizzate le relazioni tra i successivi sistemi di riferimento, procedendo come esposto nel seguito.

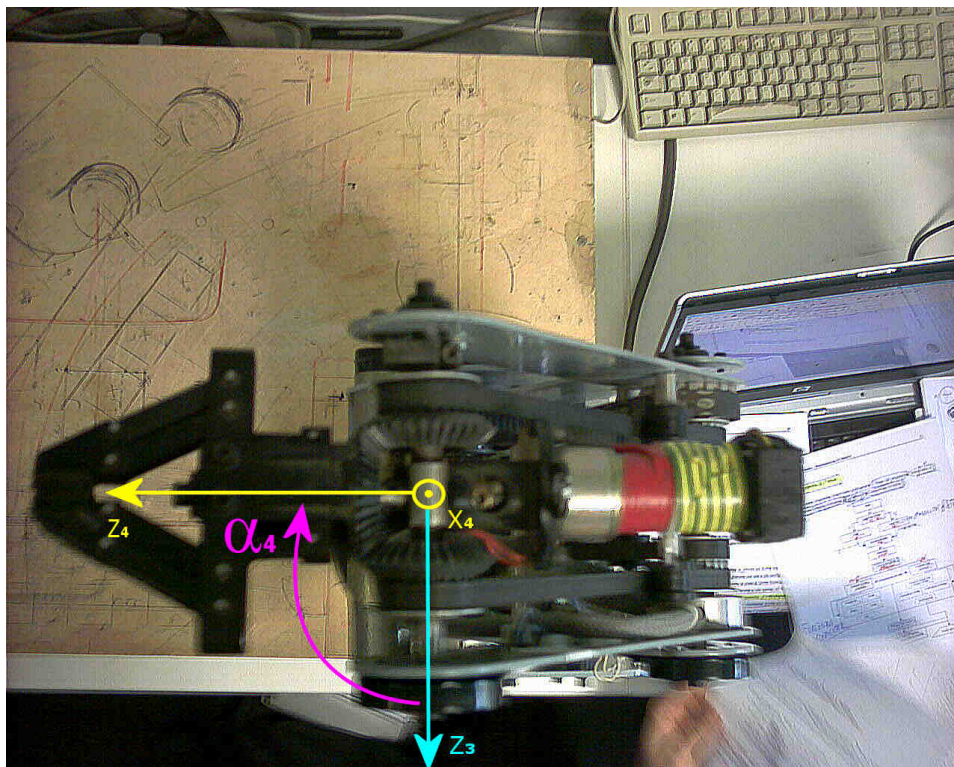
2.2 Scelta dei sistemi di riferimento

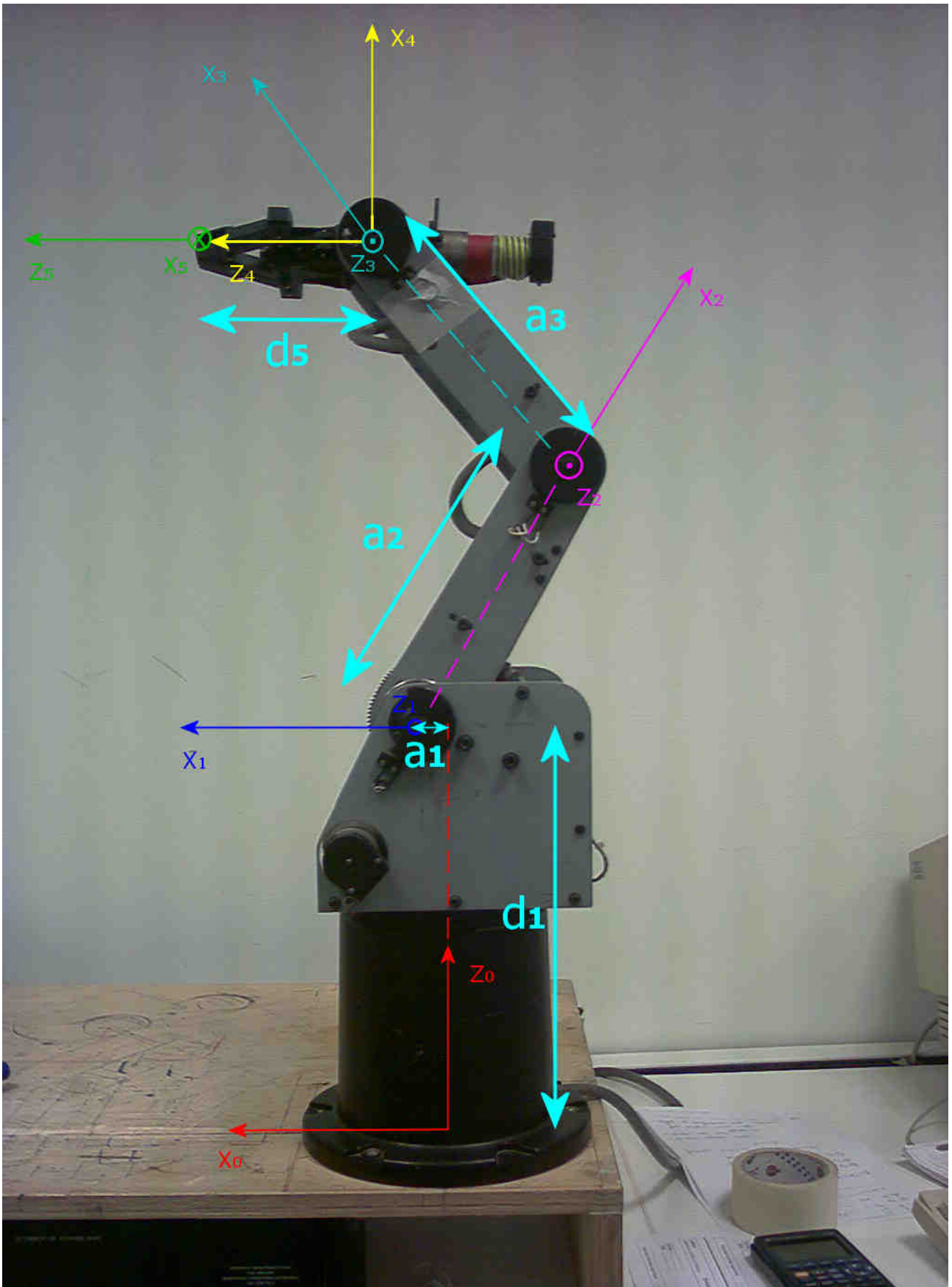
Stabilita la convenzione di D-H è possibile compilare una tabella composta da cinque righe, tante quante i gradi di libertà dello SCORBOT-ER, e quattro colonne relative ai parametri di Link ed ai parametri di Giunto. Questa tabella, detta di Denavit-Hartenberg, specifica, per ogni Giunto, i relativi parametri di Link e parametri di Giunto.

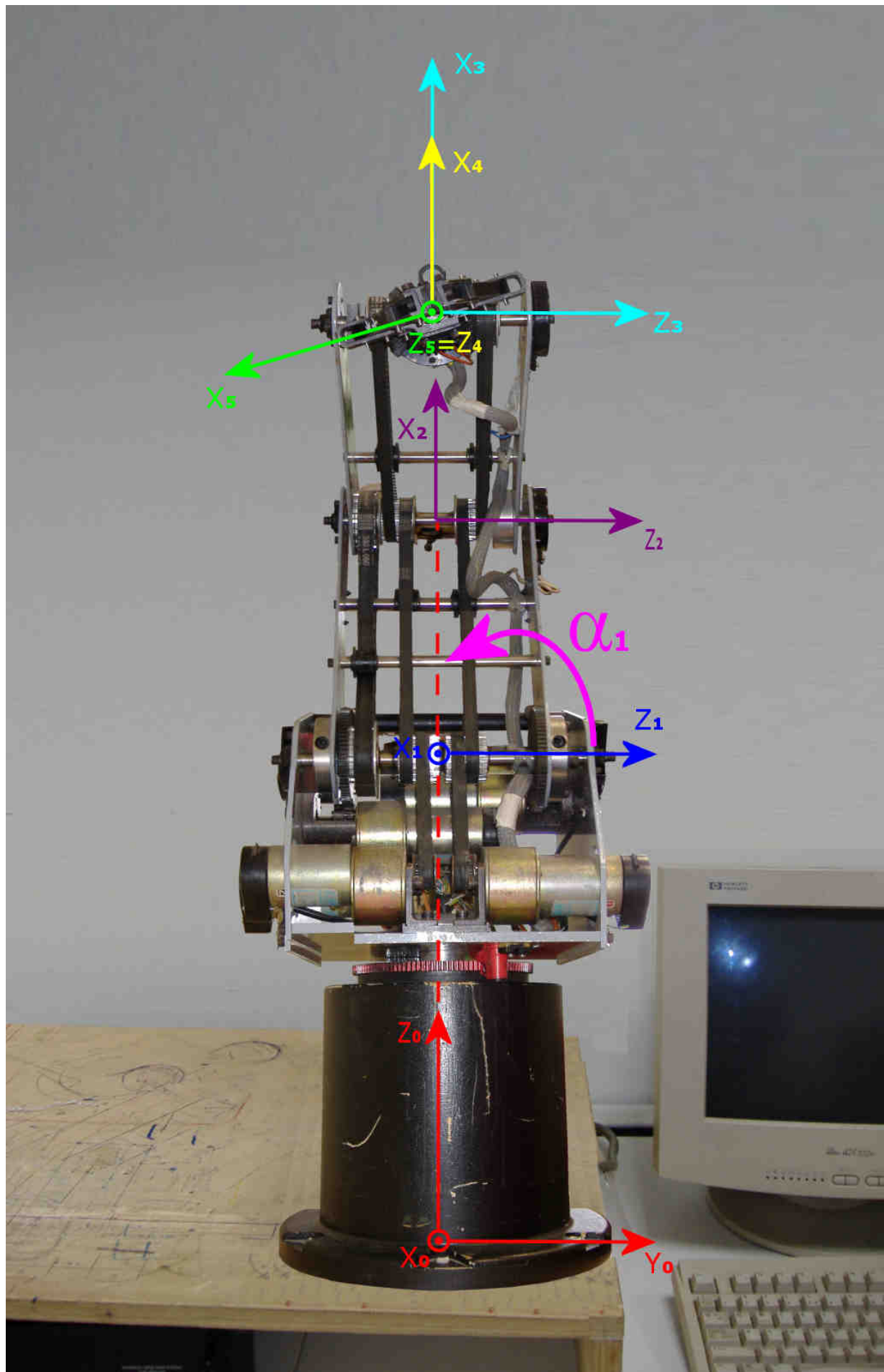
	a	α	d	θ
Giunto 1	a_1	α_1	d_1	θ_1
Giunto 2	a_2	0	0	θ_2
Giunto 3	a_3	0	0	θ_3
Giunto 4	0	α_4	0	θ_4
Giunto 5	0	0	d_5	θ_5

Tabella di Denavit-Hartenberg

Esclusa l'ultima colonna questa tabella è composta da elementi costanti che sono stati misurati direttamente sul robot.







2.3 Trasformazioni di coordinate omogenee

Definiti i sistemi di riferimento per ogni Giunto, e misurati i parametri di Giunto e di Link fissi è stato possibile mediante le coordinate omogenee, ricavare le trasformazioni di coordinate omogenee p da un sistema di riferimento $(Oxyz)_i$ al successivo $(Oxyz)_{i+1}$ (si rimanda alla sezione Appendice per approfondimenti sull'argomento Coordinate omogenee):

$$\begin{aligned}
 p_0 &= {}^0T_1 p_1 & {}^0T_1 &= \begin{bmatrix} c1 & 0 & -s1 & a_1c1 \\ s1 & 0 & c1 & a_1s1 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_1 &= {}^1T_2 p_2 & {}^1T_2 &= \begin{bmatrix} c2 & -s2 & 0 & a_2c1 \\ s2 & c2 & 0 & a_2s1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_2 &= {}^2T_3 p_3 & {}^2T_3 &= \begin{bmatrix} c3 & -s3 & 0 & a_3c3 \\ s3 & c3 & 0 & a_3s3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_3 &= {}^3T_4 p_4 & {}^3T_4 &= \begin{bmatrix} c4 & 0 & -s4 & 0 \\ s4 & 0 & c4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_4 &= {}^4T_5 p_5 & {}^4T_5 &= \begin{bmatrix} c5 & -s5 & 0 & 0 \\ s5 & c5 & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Nelle precedenti relazioni sono state operate le sostituzioni simboliche:

$$\begin{aligned}
 c1 &= \cos(\theta_1) & s1 &= \sin(\theta_1) \\
 c2 &= \cos(\theta_2) & s2 &= \sin(\theta_2) \\
 c3 &= \cos(\theta_3) & s3 &= \sin(\theta_3) \\
 c4 &= \cos(\theta_4) & s4 &= \sin(\theta_4) \\
 c5 &= \cos(\theta_5) & s5 &= \sin(\theta_5) \\
 c23 &= \cos(\theta_2 + \theta_3) & s23 &= \sin(\theta_2 + \theta_3) \\
 c234 &= \cos(\theta_2 + \theta_3 + \theta_4) & s234 &= \sin(\theta_2 + \theta_3 + \theta_4)
 \end{aligned}$$

Operando opportune sostituzioni si è ricavata la trasformazione dalle coordinate omogenee p_5 del sistema di riferimento $(Oxyz)_5$ solidale al Link 5 o di effettore, e le coordinate omogenee p_0 del sistema di riferimento di $(Oxyz)_0$ solidale al Link 0 ed al piano di lavoro.

$$p_0 = {}^0T_5 p_5$$

$${}^0T_5 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5$$

$${}^0T_5 = \begin{bmatrix} c1 \cdot c234 \cdot c5 + s1 \cdot s5 & -c1 \cdot c234 \cdot s5 + s1 \cdot c5 & -c1 \cdot s234 & c1 \cdot (-s234 \cdot d_5 + a_3 \cdot c23 + a_2 \cdot c2 + a_1) \\ s1 \cdot c234 \cdot c5 - c1 \cdot s5 & -s1 \cdot c234 \cdot s5 - c1 \cdot c5 & -s1 \cdot s234 & s1 \cdot (-s234 \cdot d_5 + a_3 \cdot c23 + a_2 \cdot c2 + a_1) \\ -s234 \cdot c5 & s234 \cdot s5 & -c234 & -c234 \cdot d_5 - a_3 \cdot s23 - a_2 \cdot s2 + d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Mediante l'ultima relazione ricavata, $p_0 = {}^0T_5 p_5$ e le proprietà delle coordinate omogenee, è stato possibile esplicitare le coordinate dell'origine O_5 di $(Oxyz)_5$, nonché l'orientamento dei singoli assi (mediante le coordinate dei relativi versori) coordinati dello stesso sistema, rispetto a $(Oxyz)_0$.

$$O_5 = O_e = \begin{bmatrix} x_e \\ y_e \\ z_e \end{bmatrix} = \begin{bmatrix} c1 \cdot (-s234 \cdot d_5 + a_3 \cdot c23 + a_2 \cdot c2 + a_1) \\ s1 \cdot (-s234 \cdot d_5 + a_3 \cdot c23 + a_2 \cdot c2 + a_1) \\ -c234 \cdot d_5 - a_3 \cdot s23 - a_2 \cdot s2 + d_1 \end{bmatrix}$$

$$\hat{x}_e = \begin{bmatrix} c1 \cdot c234 \cdot c5 + s1 \cdot s5 \\ s1 \cdot c234 \cdot c5 - c1 \cdot s5 \\ -s234 \cdot c5 \end{bmatrix}$$

$$\hat{y}_e = \begin{bmatrix} -c1 \cdot c234 \cdot s5 + s1 \cdot c5 \\ -s1 \cdot c234 \cdot s5 - c1 \cdot c5 \\ s234 \cdot s5 \end{bmatrix}$$

$$\hat{z}_e = \begin{bmatrix} -c1 \cdot s234 \\ -s1 \cdot s234 \\ -c234 \end{bmatrix}$$

Conoscendo l'orientamento degli assi cartesiani sarebbe stato possibile determinare una terna di angoli opportuna che descriva univocamente l'orientamento di $(Oxyz)_5$ rispetto a $(Oxyz)_0$. Tuttavia è risultato più immediato ricavare questa terna di angoli come semplice combinazione lineare degli angoli di Giunto $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$. In particolare ricorrendo alla convezione degli angoli di Eulero ZYZ, si ricava la terna suddetta:

$$\mathcal{E} = \begin{bmatrix} \mathcal{E}_1 \\ \mathcal{E}_2 \\ \mathcal{E}_3 \end{bmatrix}$$

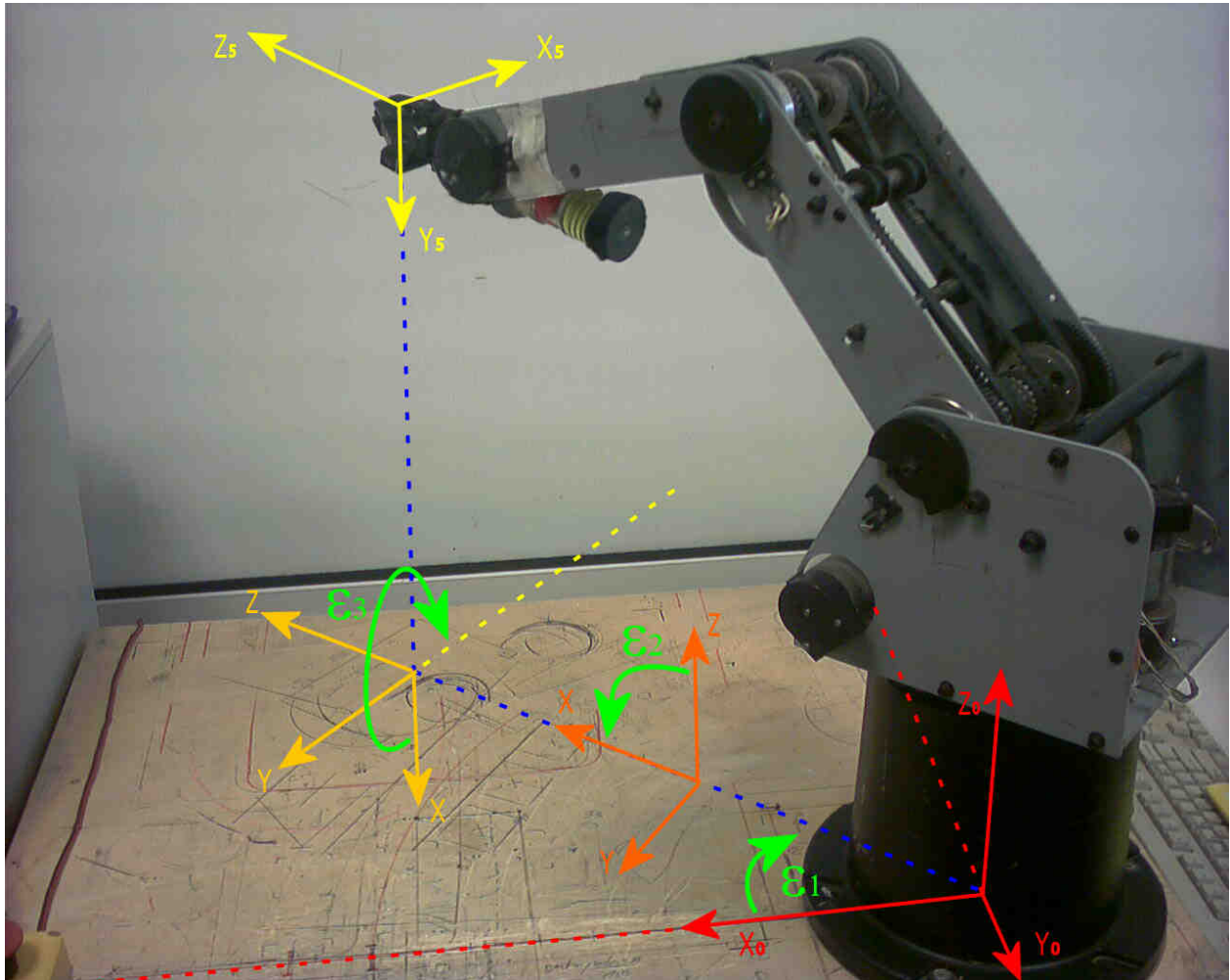
$$\mathcal{E}_1 = \theta_1$$

$$\mathcal{E}_2 = \theta_2 + \theta_3 + \theta_4 + \pi$$

$$\mathcal{E}_3 = \theta_5 + \pi$$

Ruotando il sistema di riferimento $(Oxyz)_0$ di un angolo ε_1 attorno all'asse z_0 , successivamente ruotando lo stesso sistema di un angolo ε_2 attorno all'asse y_0 corrente, ed infine ruotando ancora lo stesso sistema di un angolo ε_3 attorno all'asse z_0 corrente, tale sistema viene ad assumere lo stesso orientamento del sistema $(Oxyz)_5$. Nella sezione dedicata alla cinematica inversa viene spiegato come si possono ricavare i valori degli angoli ε_1 , ε_2 , ε_3 dalla combinazione lineare degli angoli θ_1 , θ_2 , θ_3 , θ_4 , θ_5 espressa nelle precedenti formule.

(si rimanda alla sezione Appendice per approfondimenti sull'argomento Angoli di Eulero)



E' stato così risolto il problema della cinematica diretta. Il problema in sé è stato possibile risolverlo per via teorica una volta conosciuta la struttura del robot in esame, ossia numero e tipo di Giunti, e parametri di Giunto e di Link.

E' stato così risolto il problema della cinematica diretta. Il problema in sé è stato possibile risolverlo per via teorica una volta conosciuta la struttura del robot in esame, ossia numero e tipo di Giunti, e parametri di Giunto e di Link.

Si definisce equazione di cinematica diretta la seguente applicazione:

$$[O_e, \varepsilon]' = f[q]$$

Dai risultati esposti prima si può ricavare l'espressione esplicita dell'equazione di cinematica diretta:

$$\begin{bmatrix} x_e \\ y_e \\ z_e \\ \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} = \begin{bmatrix} c1 \cdot (-s234 \cdot d_5 + a_3 \cdot c23 + a_2 \cdot c2 + a_1) \\ s1 \cdot (-s234 \cdot d_5 + a_3 \cdot c23 + a_2 \cdot c2 + a_1) \\ -c234 \cdot d_5 - a_3 \cdot s23 - a_2 \cdot s2 + d_1 \\ \theta_1 \\ \theta_2 + \theta_3 + \theta_4 + \pi \\ \theta_5 + \pi \end{bmatrix}$$

I primi problemi pratici che si sono presentati immediatamente sono stati:

- La misura effettiva dei parametri fissi di Giunto e di Link $a_1, a_2, a_3, \alpha_1, \alpha_4, d_1, d_5$.
- La misura delle posizioni angolari dei Giunti nella loro configurazione di home, descritta in seguito, ossia gli angoli $\theta_{1,home}, \theta_{2,home}, \theta_{3,home}, \theta_{4,home}, \theta_{5,home}$, assunti rispettivamente da $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ nella configurazione di Home.
- La misura del rapporto passi encoder – radianti relativamente ad ogni Giunto.

3. Sistema meccanico e di controllo

3.1 Movimenti dei giunti

I movimenti rotatori dei giunti vengono ottenuti mediante cinque motori in corrente continua, alimentati da una unità di potenza, la quale viene interfacciata ad un personal computer mediante una scheda PCI (Keithley DDA06).

Per ogni motore è presente un rapporto di riduzione, diverso per ognuno di essi.

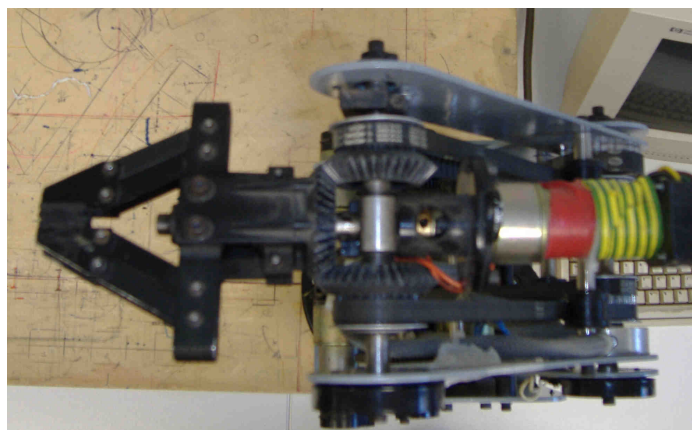
Sull'albero di ogni motore è montato un encoder incrementale, il quale fornisce all'unità di controllo, quando il motore è in rotazione, il tipico segnale ad onda quadra prodotto dall'encoder. L'unità di controllo gestisce questo segnale ricavandone l'informazione, codificata in passi encoder, della posizione angolare del motore. La posizione angolare è relativa ad una configurazione nota detta di Home.

Ad ogni motore è associato uno switch elettrico, che viene attivato quando il motore è nella sua posizione di Home. Quando lo switch è attivo il contatore di passi encoder del singolo giunto è azzerato. Ad ogni spostamento rotatorio dalla configurazione di Home del giunto corrisponderà un valore diverso da zero di passi encoder. Questo valore non è ovviamente espresso in radianti, ma il legame che esiste tra le variazioni di impulsi encoder misurata dalla unità di potenza e variazione di angolo geometrico θ misurata in radianti, relativamente ad uno stesso spostamento del giunto, è di tipo lineare (grazie alla struttura dell'encoder il cui passo di campionamento è omogeneo).

I giunti di BASE, SPALLA e GOMITO vengono azionati ognuno da un singolo motore (nell'ordine motore 1, 2, 3). Per cui la posizione angolare per ognuno dei giunti è espressa direttamente dall'informazione fornita dal relativo encoder incrementale.

POSIZIONE GIUNTO 1 = IMPULSI ENCODER 1
POSIZIONE GIUNTO 2 = IMPULSI ENCODER 2
POSIZIONE GIUNTO 3 = IMPULSI ENCODER 3

I giunti di BECCHEGGIO e ROLLIO sono invece azionati da due motori montati in un sistema differenziale. Quindi i singoli movimenti del giunto di BECCHEGGIO o di ROLLIO vengono realizzati mediante la combinazione dei movimenti dei due motori. Il movimento dei due motori in senso opposto produce il movimento di BECCHEGGIO, mentre se essi ruotano nello stesso senso viene prodotto il movimento di ROLLIO.



E' evidente quindi che l'informazione generata dagli encoder montati sui due singoli motori non corrisponde alle posizioni dei singoli giunti di BECCHEGGIO e ROLLIO. La loro posizione è invece espressa da una combinazione lineare dei passi encoder fornita da entrambi gli encoder incrementali.

$$\text{POSIZIONE GIUNTO 4} = (- \text{IMPULSI ENCODER 4} + \text{IMPULSI ENCODER 5}) / 2$$

$$\text{POSIZIONE GIUNTO 5} = (\text{IMPULSI ENCODER 4} + \text{IMPULSI ENCODER 5}) / 2$$

3.2 Controllo del robot

Il controllo del robot avviene attraverso il personal computer che alloggia la scheda di comunicazione con l'unità di potenza. L'unità di potenza non svolge il compito di controllo dei motori che azionano i giunti del robot, ma si limita ad applicare la tensione richiesta ai motori. Il segnale della tensione richiesta per ciascun motore è in ingresso all'unità di potenza e proviene dal personal computer attraverso la scheda DDA06.

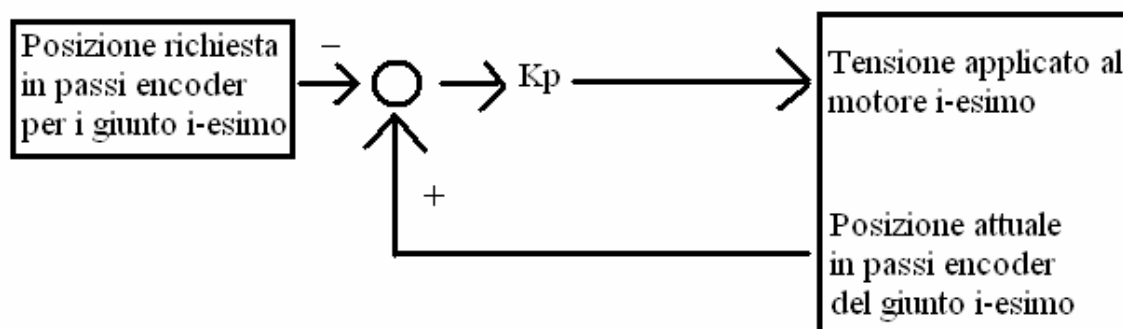
Quindi il compito del controllo degli azionamenti del robot è di competenza di un software applicativo che va eseguito sul personal computer. Il lavoro di implementare ex novo la legge di controllo dei cinque motori, e di gestire la comunicazione del segnale di tensione richiesta di ognuno di essi attraverso la scheda PCI di comunicazione, non rientrava nei compiti del corso.

Sul computer era già presente un software applicativo, sviluppato in ANSI C, che gestisce:

- La comunicazione con la scheda DDA06.
- La legge di controllo dei singoli motori.

La legge di controllo implementata nel software applicativo di base comanda non la posizione in passi encoder di ogni singolo motore, bensì la posizione di ogni giunto. In particolare questo non comporta nessuna differenza per i giunti 1, 2, e 3 per cui posizione del giunto corrisponde a posizione del motore relativo. Comporta altresì una differenza per i giunti 4 e 5 per i quali non risulta vera la corrispondenza tra posizione del giunto e posizione del motore relativo. Venendo comandata direttamente la posizione del giunto in passi encoder, non è necessario (per eseguire un movimento di BECCHEGGIO o di ROLLIO) comandare i singoli motori 4 e 5 affinché eseguano una rotazione nello stesso senso (per il movimento di ROLLIO), o in senso opposto (per il movimento di BECCHEGGIO). Richiedendo una posizione di giunto nel caso di BECCHEGGIO e ROLLIO la legge di controllo, già implementata, provvede ad assegnare le corrette tensioni ad ambo i motori 4 e 5.

La legge di controllo, implementata sul software di base e eseguita per ogni giunto, può essere così rappresentata:



Essa viene eseguita da una funzione dell'applicativo, che una volta attivata, viene richiamata periodicamente con una frequenza rigidamente prefissata (viene eseguita in Real Time). La funzione della legge di controllo viene invocata indipendentemente dal flusso del restante programma. Data la frequenza con la quale viene invocata (ogni 20 o 30 millisecondi) e la velocità con cui viene eseguita (è composta da poche istruzioni C), si può ammettere che durante l'esecuzione dell'esecutivo di base, sia in esecuzione in background una funzione che gestisce il controllo del robot e che condivide, con la funzione principale main(), delle variabili globali quali posizioni richieste e attuali dei giunti e tensioni applicate ai motori.

Tramite l'applicativo di base è possibile:

- Visualizzare la posizione, in passi encoder, dei singoli motori.
- Specificare, in passi encoder, una posizione richiesta per ciascun giunto.

La legge di controllo si adopera, ogni volta che vengono richieste delle posizioni di giunto a comandare i motori perché posizionino il robot nella configurazione richiesta.

4. Calcolo dei rapporti passi encoder – radianti

4.1 Rapporti encoder – radianti

Utilizzando l'applicativo di base è stato fin da subito possibile calcolare il rapporto esistente tra le variazioni della misura di posizione dei giunti in passi encoder e in radianti, a seguito di un movimento di un giunto.

Concluse queste misurazioni, la cui metodologia è descritta in una successiva sezione appositamente dedicata, si sono ricavate le seguenti relazioni:

$$\begin{aligned}\Delta(\theta_1) &= \text{rapporto}_1 \text{ impulso}_1 \\ \Delta(\theta_2) &= \text{rapporto}_2 \text{ impulso}_2 \\ \Delta(\theta_3) &= \text{rapporto}_3 \text{ impulso}_3 \\ \Delta(\theta_4) &= \text{rapporto}_4 \text{ impulso}_4 \\ \Delta(\theta_5) &= \text{rapporto}_5 \text{ impulso}_5\end{aligned}$$

Dove per impulso_i si intendono i passi encoder dell'i-esimo angolo di giunto, comandati attraverso la modifica dell'i-esima componente dell'array *ROBOTDefCtrlPosReq[]* (posizione richiesta) e restituiti dalla funzione *ROBOTGetLinkEncoder(i)* (posizione attuale). Per $\Delta(\theta_i)$ si intende la variazione dell'i-esimo angolo di giunto espressa in radianti.

Dal momento che eseguendo le suddette misurazioni avveniva che comandando i passi encoder di alcuni giunti, veniva modificata la posizione angolare di un altro giunto oltre a quello selezionato, è parso da subito evidente che le relazioni riportate sopra andavano corrette, inserendo ad alcune di esse la dipendenza delle variazioni dell'angolo di giunto dai passi encoder di un altro giunto.

In particolare per i giunti di BASE, SPALLA, ROLLIO (1, 2, 5), si è osservato come la variazione dell'angolo θ_i rispettivo dipendesse solo dalla modifica del relativo valore *ROBOTDefCtrlPosReq[i]*. Appurato ciò si sono potute confermare per tali giunti le relazioni suddette:

$$\begin{aligned}\Delta(\theta_1) &= \text{rapporto}_1 \text{ impulso}_1 \\ \Delta(\theta_2) &= \text{rapporto}_2 \text{ impulso}_2 \\ \Delta(\theta_5) &= \text{rapporto}_5 \text{ impulso}_5\end{aligned}$$

Riguardo ai restanti giunti, di GOMITO e BECCHEGGIO (3, 4), si è potuto osservare come:

- La variazione angolare $\Delta(\theta_3)$ fosse provocata anche dalla modifica di *ROBOTDefCtrlPosReq[2]* e quindi dal successivo movimento del giunto di SPALLA (giunto 2)
- La variazione angolare $\Delta(\theta_4)$ fosse provocata anche dalla modifica di *ROBOTDefCtrlPosReq[3]* e quindi dal successivo movimento del giunto di GOMITO (giunto 3)

*(Ricordiamo che in ANSI C l'indice di un array parte dal numero 0 e non dal numero 1, quindi *ROBOTDefCtrlPosReq[2]* e *ROBOTDefCtrlPosReq[3]* appena citati andrebbero corretti diminuendo di un'unità il loro indice. Nel programma di base e in quello successivamente sviluppato nell'elaborato del progetto viene fatto uso corretto degli indici in C. Tuttavia per una maggiore immediatezza e chiarezza, nella seguente relazione si fa uso di indici che partono da 1, per cui, ad esempio con *ROBOTDefCtrlPosReq[2]* ci riferiamo alla posizione richiesta in passi*

encoder per il giunto 2. Nell'implementazione effettiva del programma C la variabile suddetta per il giunto 2 sarà invece *ROBOTDefCtrlPosReq[1]*).

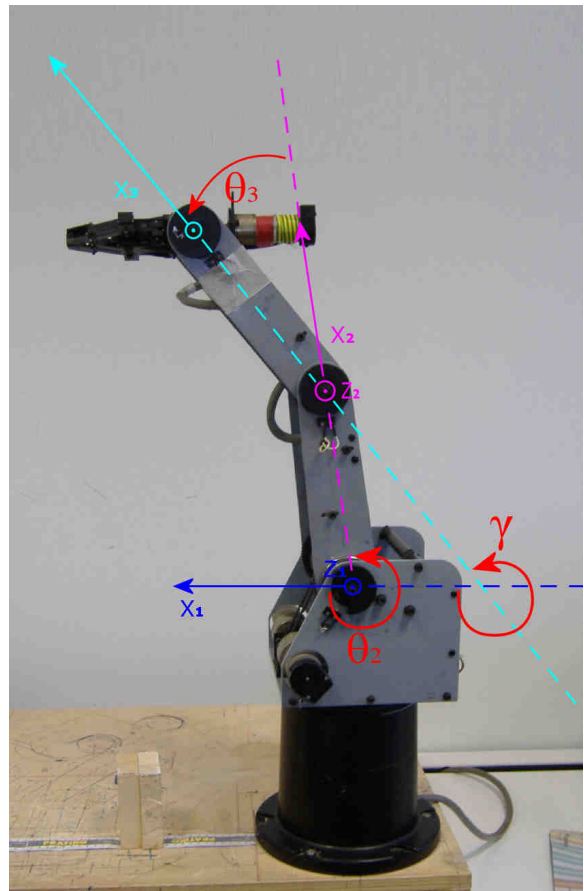
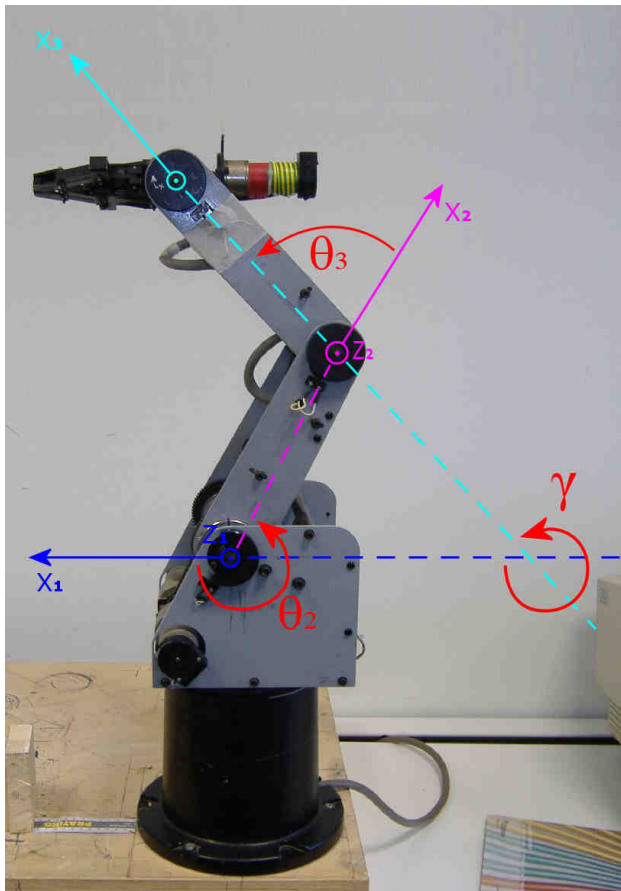
Andava quindi formalizzata la dipendenza di $\Delta(\theta_3)$ da $impulso_2$ e la dipendenza di $\Delta(\theta_4)$ da $impulso_3$. Invece che procedere per mezzo di ulteriori misurazioni alla determinazione degli ulteriori rapporti di conversione passi encoder – radianti, si è notato che per entrambe le situazioni essi potevano essere ottenuti per via formale, grazie all'osservazione del moto del robot.

Si è in effetti evidenziato come:

- Nel movimento del giunto 2, o di SPALLA, l'orientamento del sistema di riferimento $(Oxyz)_3$ rispetto a $(Oxyz)_0$ rimane invariato. In particolare quindi l'asse x_3 mantiene la stessa inclinazione rispetto al piano di lavoro. Se ne evince, anche graficamente come riportato, che la somma $\gamma = \theta_2 + \theta_3$ rimane costante. Si ricava quindi:

$$\begin{aligned}\gamma &= \theta_2 + \theta_3 \\ \Delta\gamma/\Delta\theta_2 &= \Delta\theta_2/\Delta\theta_2 + \Delta\theta_3/\Delta\theta_2 \\ 0 &= 1 + \Delta\theta_3/\Delta\theta_2 \\ \Delta\theta_3/\Delta\theta_2 &= -1 \\ \Delta\theta_3 &= -\Delta\theta_2\end{aligned}$$

Ossia durante un movimento del giunto 2, la variazione dell'angolo di giunto θ_3 è uguale ed opposta a quella che avviene su θ_2 . In altre parole, come si può anche vedere bene dalle immagini, durante un movimento del giunto 2 all'aumentare di θ_2 , θ_3 diminuisce, e viceversa.

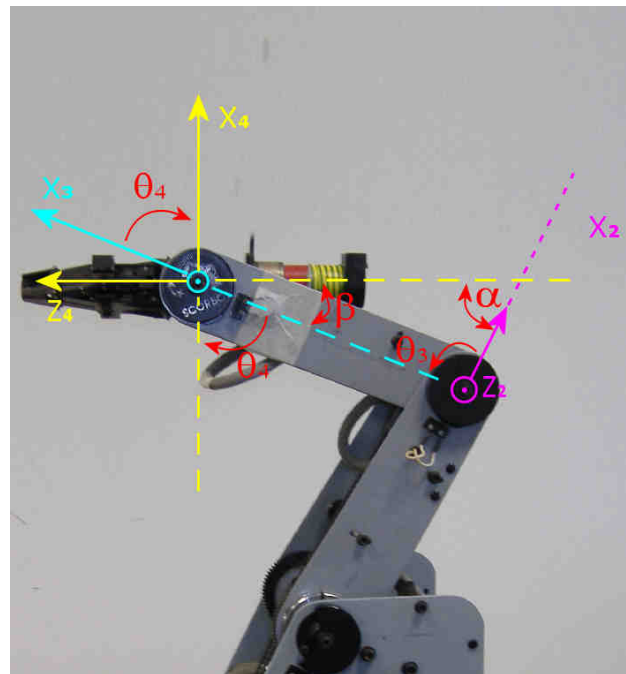
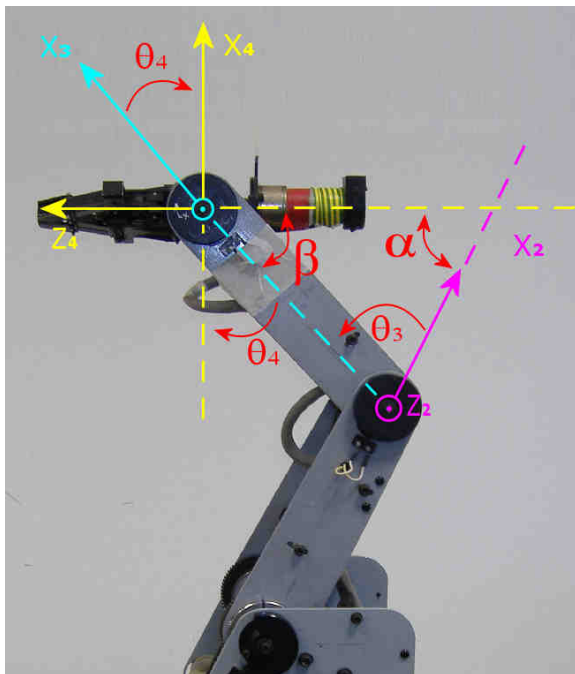


Ricordando che durante un movimento del giunto 2 si ha che $\Delta(\theta_2) = \text{rapporto}_2 \text{ impulso}_2$, si può allora correggere la relazione suddetta sulla variazione di θ_3 :

$$\Delta(\theta_3) = \text{rapporto}_3 \text{ impulso}_3 - \Delta(\theta_2)$$

$$\Delta(\theta_3) = \text{rapporto}_3 \text{ impulso}_3 - \text{rapporto}_2 \text{ impulso}_2$$

- Nel movimento del giunto 3, o di GOMITO, l'orientamento del sistema di riferimento $(Oxyz)_4$ rispetto a $(Oxyz)_0$ rimane invariato. In particolare quindi l'asse z_4 mantiene la stessa inclinazione rispetto al piano di lavoro, nonché rispetto a x_2 (il sistema $(Oxyz)_2$ durante un movimento del giunto 3 non si muove rispetto a $(Oxyz)_0$). Se ne evince, anche graficamente come riportato, che la somma $\theta_3 + \theta_4$ rimane costante. Questo si può ricavare anche formalmente grazie a considerazioni sulla somma degli angoli interni del triangolo formato dagli assi x_2, x_3, z_4 :



$$\beta = \pi/2 + \theta_4 (*)$$

$$\pi = \alpha + \beta + \theta_3$$

$$\pi = \alpha + \pi/2 + \theta_4 + \theta_3$$

$$\Delta\pi/\Delta\theta_3 = \Delta\alpha/\Delta\theta_3 + \Delta\pi/2/\Delta\theta_3 + \Delta\theta_4/\Delta\theta_3 + \Delta\theta_3/\Delta\theta_3$$

$$0 = 0 + \Delta\theta_4/\Delta\theta_3 + 1 (**)$$

$$\Delta\theta_4 \Delta\theta_3 = -1$$

$$\Delta\theta_4 = -\Delta\theta_3$$

(*) θ_4 in figura è negativo perché orario

(**) α non varia durante un movimento del giunto 3, poiché z_4 come già spiegato mantiene la stessa inclinazione rispetto a x_2

Ossia durante un movimento del giunto 3 la variazione dell'angolo di giunto θ_4 è uguale ed opposta a quella che avviene su θ_3 . In altre parole, come si può anche vedere bene dalle immagini, durante un movimento del giunto 3 al diminuire di θ_3 , θ_4 aumenta, e viceversa (si ricorda sempre che in figura θ_4 è negativo).

Ricordando che durante un movimento del giunto 3 si ha che $\Delta(\theta_3) = \text{rapporto}_3 \text{ impulso}_3$, si può allora correggere la relazione suddetta sulla variazione di θ_3 :

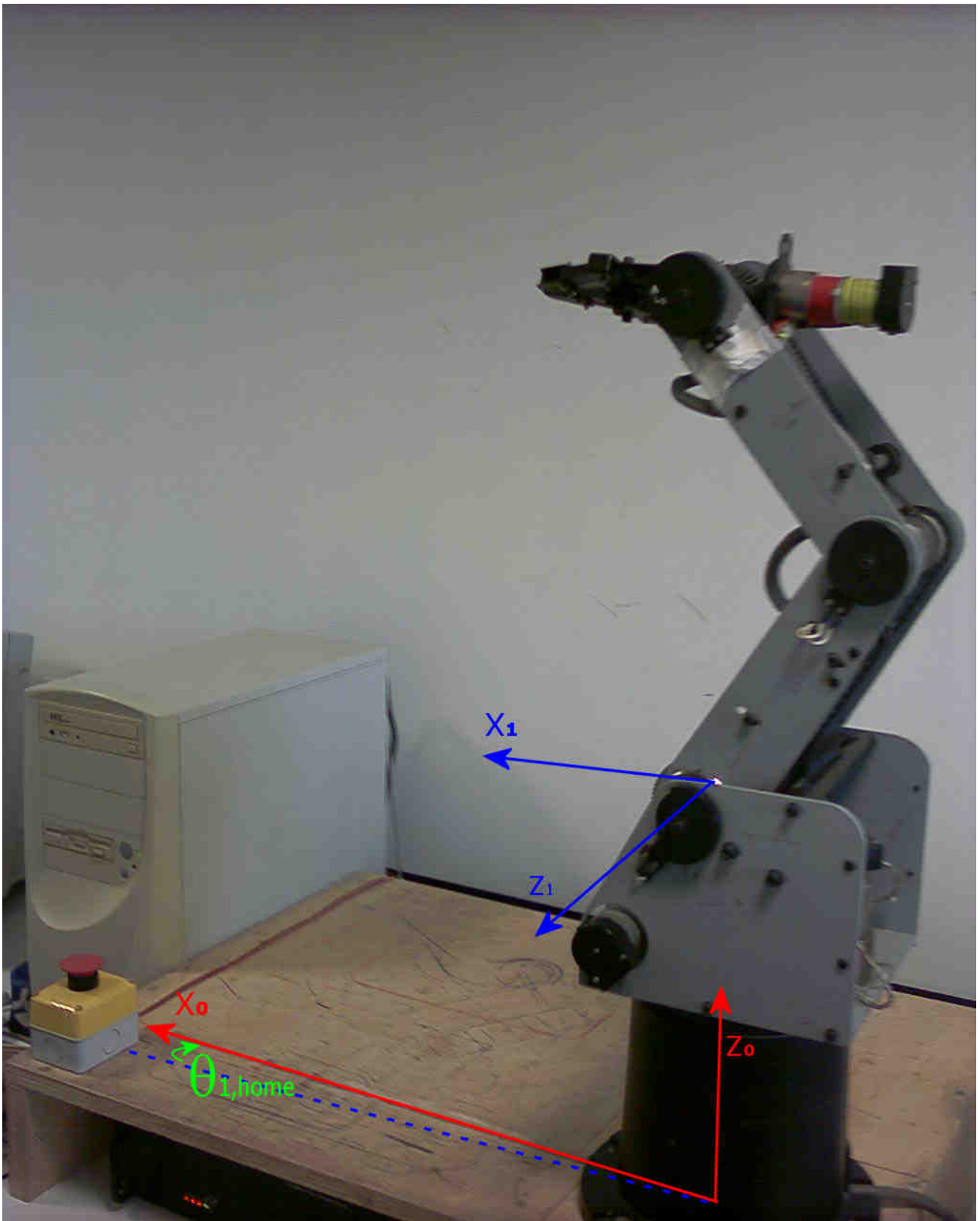
$$\begin{aligned}\Delta(\theta_4) &= \text{rapporto}_4 \text{ impulso}_4 - \Delta(\theta_3) \\ \Delta(\theta_4) &= \text{rapporto}_4 \text{ impulso}_4 - \text{rapporto}_3 \text{ impulso}_3\end{aligned}$$

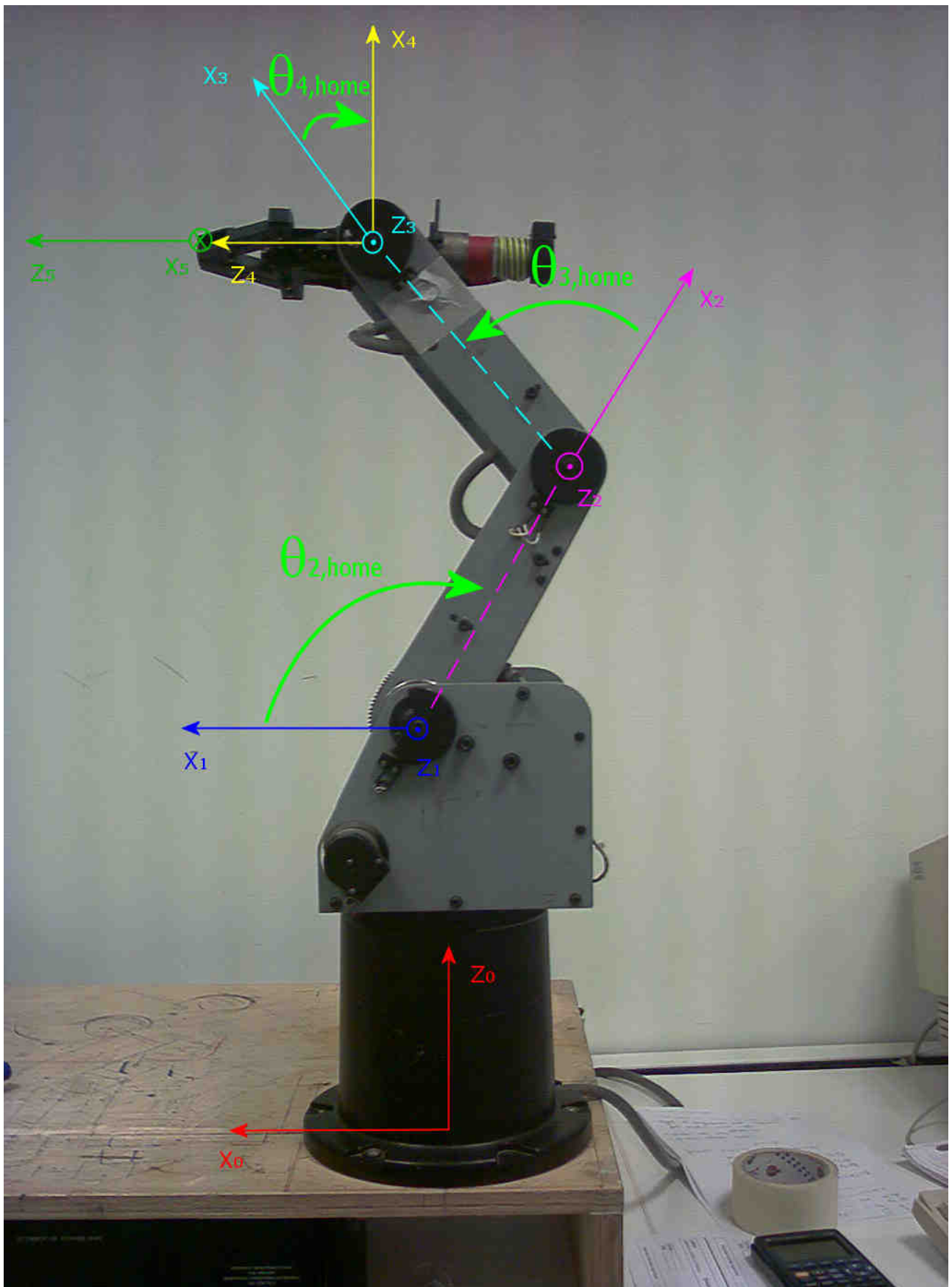
Le relazioni sono state quindi corrette nelle seguenti:

$$\begin{aligned}\Delta(\theta_1) &= \text{rapporto}_1 \text{ impulso}_1 \\ \Delta(\theta_2) &= \text{rapporto}_2 \text{ impulso}_2 \\ \Delta(\theta_3) &= \text{rapporto}_3 \text{ impulso}_3 - \text{rapporto}_2 \text{ impulso}_2 \\ \Delta(\theta_4) &= \text{rapporto}_4 \text{ impulso}_4 - \text{rapporto}_3 \text{ impulso}_3 \\ \Delta(\theta_5) &= \text{rapporto}_5 \text{ impulso}_5\end{aligned}$$

4.2 Angoli di Home

Per evitare di considerare spostamenti relativi dei giunti è stato necessario correggere ancora le relazioni, sostituendo alle variazioni $\Delta(\theta_i)$ espressioni del tipo $\theta_i - \theta_{i,\text{home}}$, dove $\theta_{i,\text{home}}$ sono gli angoli di giunto che vengono assunti dal robot nella configurazione di Home, ossia quando tutti gli impulso_i sono azzerati.





$$\begin{aligned}
\theta_1 - \theta_{1,home} &= \text{rapporto}_1 \text{ impulso}_1 \\
\theta_2 - \theta_{2,home} &= \text{rapporto}_2 \text{ impulso}_2 \\
\theta_3 - \theta_{3,home} &= \text{rapporto}_3 \text{ impulso}_3 - \text{rapporto}_2 \text{ impulso}_2 \\
\theta_4 - \theta_{4,home} &= \text{rapporto}_4 \text{ impulso}_4 - \text{rapporto}_3 \text{ impulso}_3 \\
\theta_5 - \theta_{5,home} &= \text{rapporto}_5 \text{ impulso}_5
\end{aligned}$$

$$\text{impulsi} = \begin{bmatrix} \text{impulso}_1 \\ \text{impulso}_2 \\ \text{impulso}_3 \\ \text{impulso}_4 \\ \text{impulso}_5 \end{bmatrix}$$

$$q_{home} = \begin{bmatrix} \theta_{1,home} \\ \theta_{2,home} \\ \theta_{3,home} \\ \theta_{4,home} \\ \theta_{5,home} \end{bmatrix}$$

$$q = H \text{ impulsi} + q_{home} \quad (\text{pippo})$$

Invertendo l'ultima relazione ricavata:

$$\begin{aligned}
G &= H^{-1} \\
\text{impulsi} &= G (q - q_{home}) \quad (\text{pluto})
\end{aligned}$$

$$G = \begin{bmatrix} \frac{1}{\text{rapporto}_1} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\text{rapporto}_2} & 0 & 0 & 0 \\ 0 & \frac{1}{\text{rapporto}_3} & \frac{1}{\text{rapporto}_3} & 0 & 0 \\ 0 & \frac{1}{\text{rapporto}_4} & \frac{1}{\text{rapporto}_4} & \frac{1}{\text{rapporto}_4} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{\text{rapporto}_5} \end{bmatrix}$$

$$H = \begin{bmatrix} \text{rapporto}_1 & 0 & 0 & 0 & 0 \\ 0 & \text{rapporto}_2 & 0 & 0 & 0 \\ 0 & -\text{rapporto}_2 & \text{rapporto}_3 & 0 & 0 \\ 0 & 0 & -\text{rapporto}_3 & \text{rapporto}_4 & 0 \\ 0 & 0 & 0 & 0 & \text{rapporto}_5 \end{bmatrix}$$

Ricavata questa equazione è stato possibile conoscere quali valori imporre a *ROBOTDefCtrlPosReq[i]*, al fine di posizionare il robot in una configurazione q' dello spazio dei giunti nota. In altre parole è stato possibile conoscere quale valore di encoder, comandare ad ogni motore che aziona i giunti affinché questi ultimi formino gli angoli θ_1' , θ_2' , θ_3' , θ_4' , θ_5' richiesti.

5. Misurazioni Effettuate

Dopo aver scelto i sistemi di riferimento per ogni giunto, rispettando le regole di caratterizzazione di Denavit - Hartenberg, si passa alla misura sul campo dei parametri di link e dei parametri di giunto, che andranno inseriti nella tabella D-H per il calcolo delle matrici di trasformazione.

Un manipolatore meccanico, nel nostro caso lo Scrobot ER, consiste in una sequenza di segmenti rigidi chiamati link, connessi a dei giunti rotazionali che ne permettono il movimento. Ogni coppia link-giunto è caratterizzato da quattro parametri a , α , d , θ .

5.1 Parametri di Link:

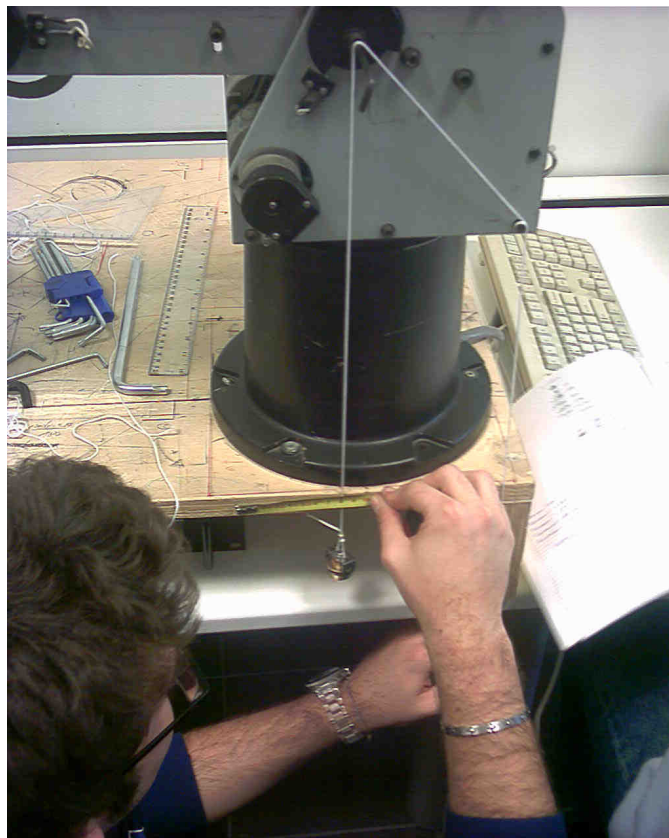
Seguendo la convenzione D-H, secondo la quale a è la lunghezza della normale tra due assi di giunto consecutivi, e α l'angolo di twist tra essi, sono state eseguite delle misurazioni sul robot. Ogni misura è stata effettuata per un minimo di tre volte, al fine di minimizzare l'errore di misura.



La particolare conformazione dello Scrobot ha permesso una facile misurazione dei parametri a_2 e a_3 in quanto coincidenti con la lunghezza dei relativi link 2 e 3. In figura si vede la misurazione del parametro a_3 , ottenuta appoggiando un righello sui perni dei giunti 3 e 4 (che grazie alla struttura del robot sono coincidenti con gli assi di rotazione dei giunti), e misurando tale lunghezza con un metro a nastro di alluminio (usato per tutte le misurazioni).



In modo analogo si vede la misurazione del parametro a_2 , dove si può osservare l'utilizzo di una chiave a brugola inserita nel centro del perno del giunto superiore, in modo tale da poter permettere una misurazione corretta della distanza dal perno del giunto inferiore, che si trovava altrimenti su un piano differente, rendendo difficoltosa la misura.



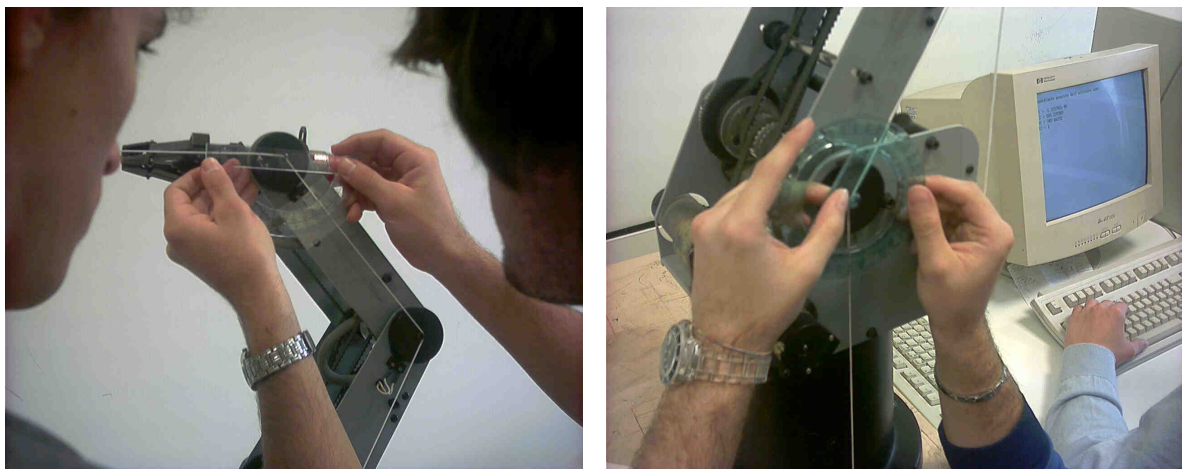
E' stata invece prestata particolare attenzione nella misura del parametro a_1 , che vede i primi due assi di giunto perpendicolari tra loro, con il giunto 1 posto al centro del cilindro di base, in una posizione pertanto scomoda. Utilizzando un filo a piombo si è visto che i due assi non sono coincidenti, ma si trovano spostati di un piccolo offset corrispondente proprio ad a_1 .

Per tale misurazione è stato disegnato un quadrato circoscritto alla circonferenza di base, e supposto l'asse del primo giunto passante per il centro di tale circonferenza, è bastato misurare la distanza dal filo a piombo dalla metà del lato del quadrato.

Non è stata invece necessaria alcuna misurazione per gli angoli α_i , che sono stati calcolati semplicemente tenendo conto dei sistemi di riferimento scelti, e sono risultati essere di -90° o 0° .

5.2 Parametri di giunto:

Ricordando che d indica la distanza tra le intersezioni delle normali sull'asse del giunto preso in esame, e θ l'angolo di giunto (cioè l'angolo tra due link adiacenti), siamo andati a misurare tali parametri con il robot in posizione di Home (essenziale per la determinazione di un sistema di riferimento assoluto).



In figura vediamo le misurazioni degli angoli $\theta_{2,home}$ e $\theta_{4,home}$, effettuate tramite un goniometro e l'utilizzo del filo a piombo, passante sui perni dei giunti, per la determinazione della giusta incidenza degli angoli.

Durante queste misurazioni si è dovuto tenere conto dei giochi meccanici (non indifferenti) dovuti ai riduttori applicati ai motorini elettrici.

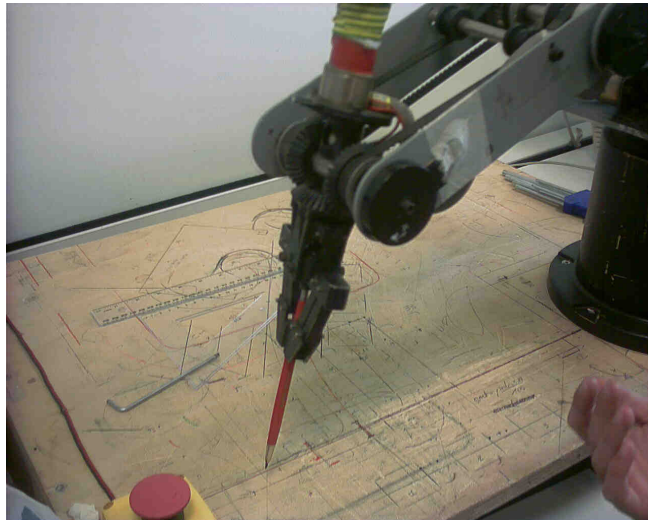
Si è osservato che nei movimenti del robot sullo spazio di lavoro tali giochi venivano “annullati” dal peso dei singoli bracci, che per effetto della gravità si posizionavano a fine corsa. Pertanto nelle misurazioni in posizione di Home sono stati portati i bracci fino a fine corsa del gioco, nella posizione che avrebbero assunto in movimento.

5.3 Rapporti “Passi encoder – Angoli”:

Da ultimo, ma non meno importante, si è proceduto alla misurazione dei rapporti di correlazione tra passi encoder (e quindi impulsi elettrici inviati ai motori) e angoli di spostamento effettivo.

Partendo da posizioni conosciute, e facendo compiere movimenti con angoli facilmente misurabili, siamo andati a leggere il valore degli encoder riportato dal programma dimostrativo, ed anche qui tenendo conto dei vari scarti, abbiamo stabilito i vari rapporti.

In figura ad esempio vediamo la misurazione del rapporto 1, dove abbiamo fatto compiere al motore di base una rotazione di 90° (cioè dall'asse x all'asse y del sistema di riferimento di base)



Per il rapporto 2 è stato utilizzato il massimo angolo possibile (sfruttando i fine corsa), mentre il rapporto 3 è stato misurato facendo compiere al link 2 un angolo di 180° .

Per la misurazione dei rapporti 4 e 5 (movimenti di pitch e roll) è bastato invece notare che trattandosi di movimenti generati dagli stessi due motori (anche se con verso opposto), dovevano necessariamente essere uguali. Premesso ciò, abbiamo quindi misurato il rapporto 4 facendo compiere al link 5 sempre un angolo di 180° , e lo abbiamo uguagliato al rapporto 5, evitando così una difficile misurazione, con conseguenti risultati non precisi. (misurare l'angolo di roll è infatti particolarmente difficoltoso, a causa della conformazione del robot).

5.4 Risultato delle misurazioni

Vengono quindi riportati i risultati delle misure effettuate per determinare i parametri costanti di giunto e di link, e per determinare i rapporti passi encoder – angoli radianti.

	a	α	d	θ
Giunto 1	$a_1 = 1.6 \text{ cm}$	$\alpha_1 = -\pi/2$	$d_1 = 34.9 \text{ cm}$	θ_1
Giunto 2	$a_2 = 22.15 \text{ cm}$	0	0	θ_2
Giunto 3	$a_3 = 22.15 \text{ cm}$	0	0	θ_3
Giunto 4	0	$\alpha_4 = -\pi/2$	0	θ_4
Giunto 5	0	0	$d_5 = 14.6 \text{ cm}$	θ_5

Tabella di Denavit-Hartenberg

$$\text{rapporto}_1 = -\pi / (21.35 \cdot 180.0)$$

$$\text{rapporto}_2 = -\pi / (16.8 \cdot 180.0)$$

$$\text{rapporto}_3 = +\pi / (16.8 \cdot 180.0)$$

$$\text{rapporto}_4 = -\pi / (4.18 \cdot 180.0)$$

$$\text{rapporto}_5 = +\pi / (4.18 \cdot 180.0)$$

6. Cinematica Inversa

In sistemi di movimentazione automatica, come nello SCORBOT-ER V, è utile avere la possibilità di far raggiungere all'effettore del robot una posizione indicata da coordinate cartesiane e in generale anche da tre angoli di orientamento che, insieme alle coordinate, fissino una data configurazione per il sistema di effettore all'interno dello spazio di lavoro destrorso del robot.

Per garantire questa possibilità è necessario risolvere il problema detto della cinematica inversa di un robot, ovvero calcolare i parametri di ogni giunto relativi alla posizione e all'orientamento desiderati per l'effettore.

6.1 Semplificazioni

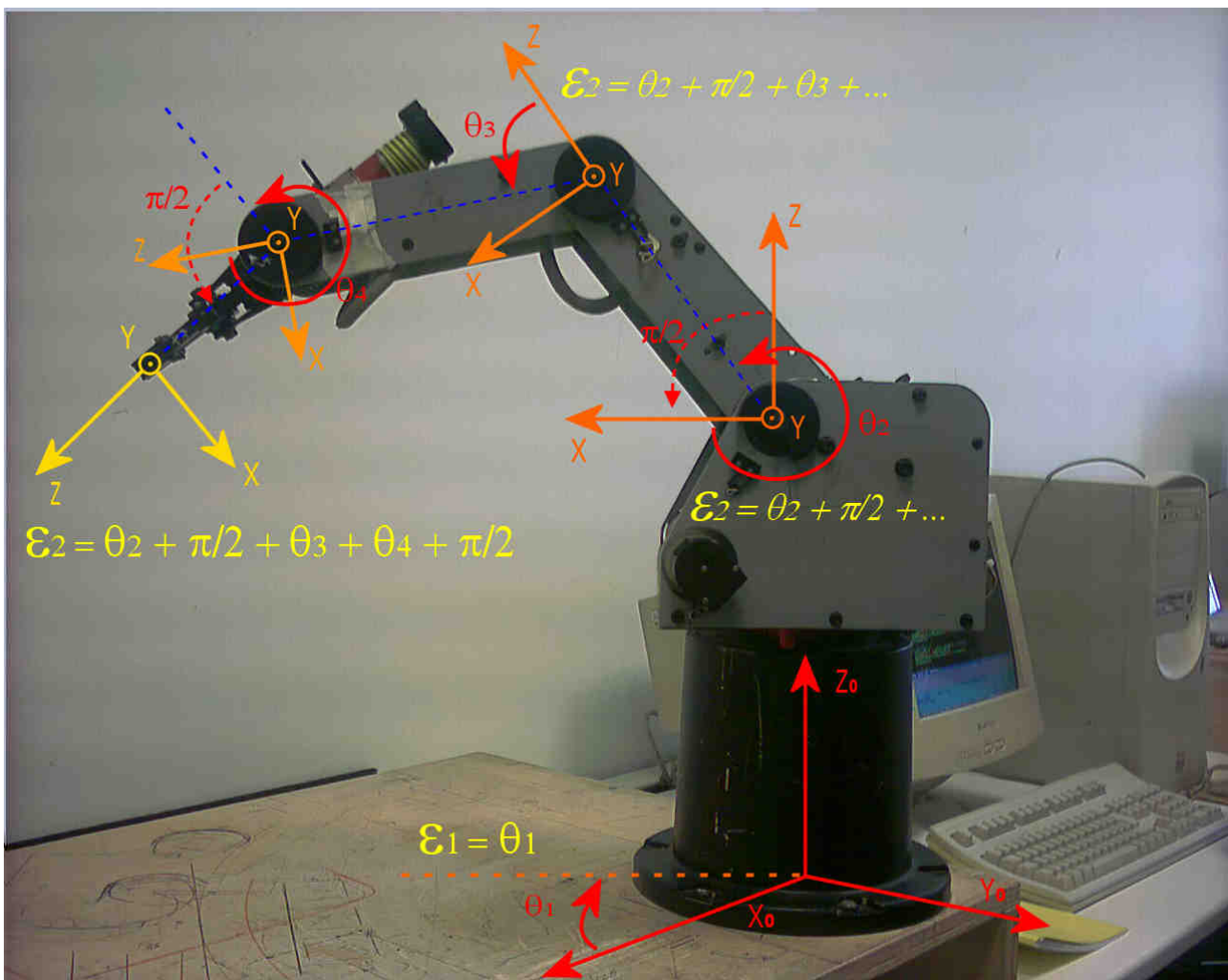
Analizzando in particolare il nostro robot, il problema generale si riduce a problemi minori dopo aver effettuato delle considerazioni:

l'orientamento del sistema di effettore può essere descritto dalla terna di angoli di Eulero ZYZ; in questo modo si verificano le corrispondenze

$$\varepsilon_1 = \theta_1 \quad (1)$$

$$\varepsilon_2 = (\theta_2 + \pi/2) + \theta_3 + (\theta_4 + \pi/2) \quad (2)$$

$$\varepsilon_3 = \theta_5 + \pi \quad (3)$$



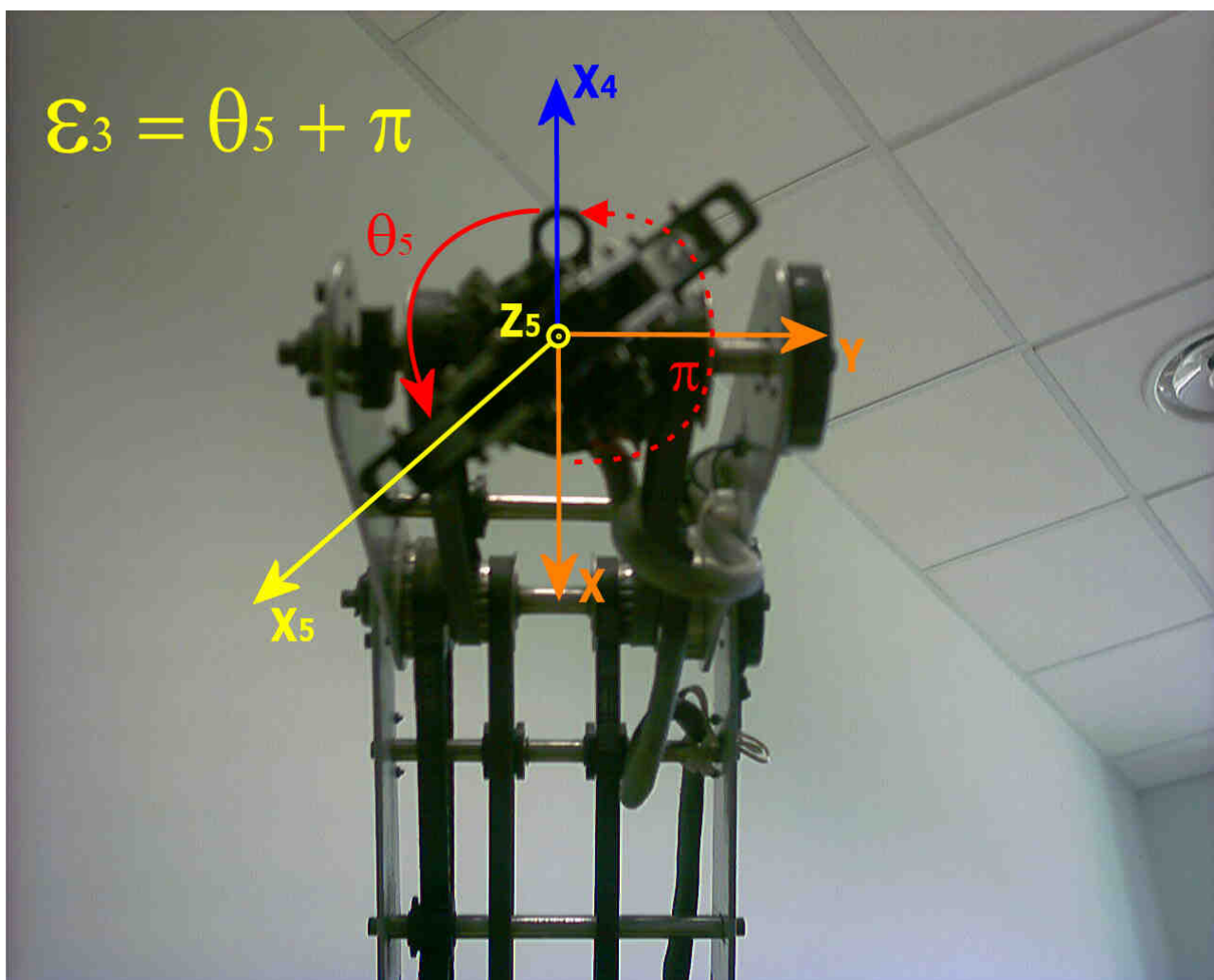
derivanti dalle seguenti argomentazioni:

il movimento del giunto 1 corrisponde a una rotazione intorno all'asse Z del sistema di base;

gli assi di rotazione dei giunti 2, 3 e 4 sono sempre perpendicolari all'asse X del sistema di riferimento $(Oxyz)_1$ per motivi strutturali e per questo l'insieme delle tre rotazioni può essere assimilato a una sola rotazione somma delle 3 (considerando nel calcolo solo l'orientamento dei sistemi e non la posizione);

il giunto 5 effettua una rotazione intorno a un'asse sempre complanare agli assi Z_1 , Z_2 e Z_3 che quindi, dopo aver effettuato alcuni accorgimenti per il calcolo di ε_2 , può essere considerato l'asse Z del sistema di riferimento finora ottenuto.

(si rimanda alla sezione Appendice per approfondimenti sull'argomento Angoli di Eulero)

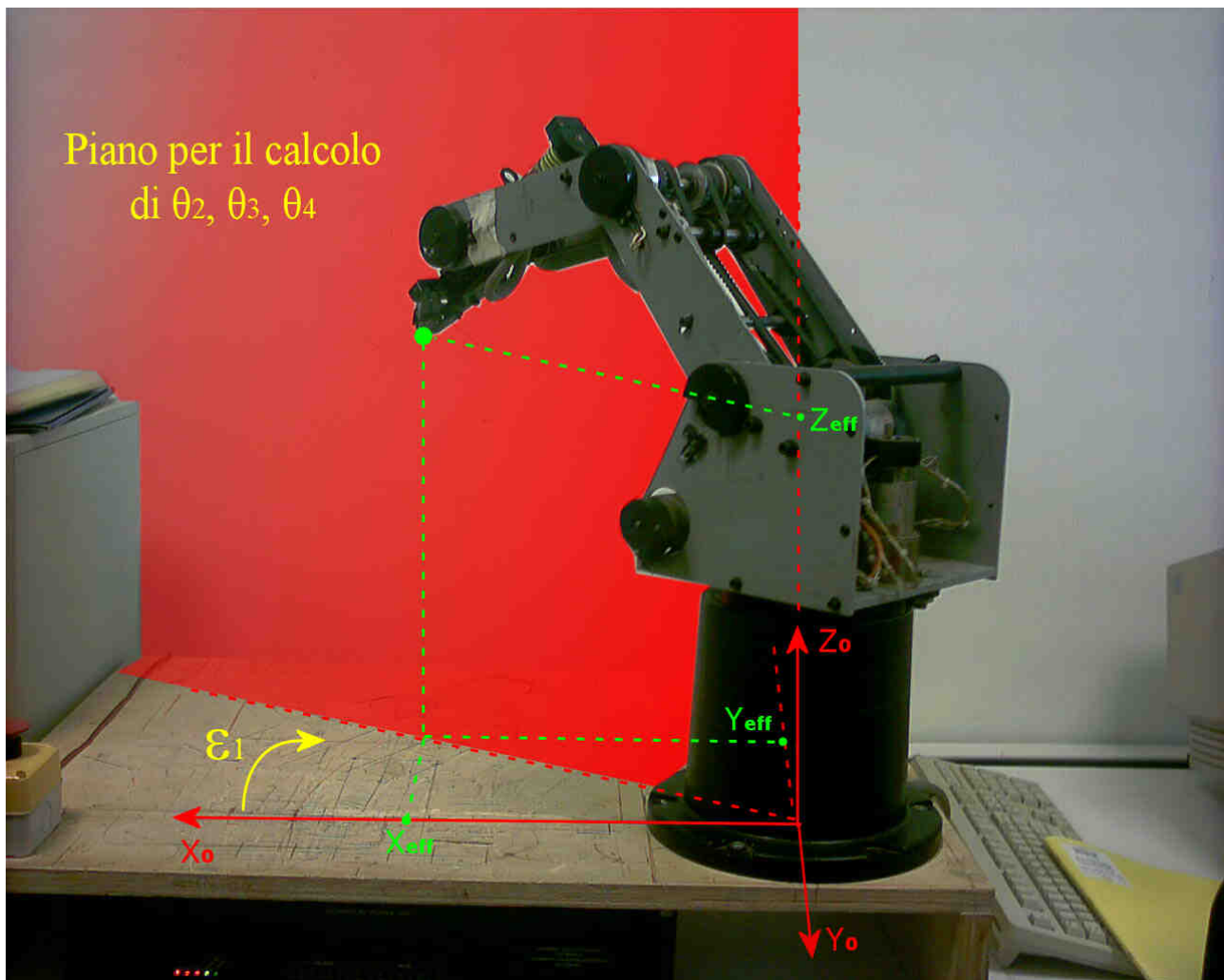


6.2 Calcolo degli Angoli di Giunto

Per motivi strutturali, la proiezione di O_5 sul piano X_0Y_0 si trova sempre sulla proiezione dell'asse X_1 sullo stesso piano, per questo le coordinate x_5 e y_5 desiderate obbligano la scelta di eulerio1:

$$\theta_1 = \varepsilon_1 = \text{atan2}(y_e, x_e).$$

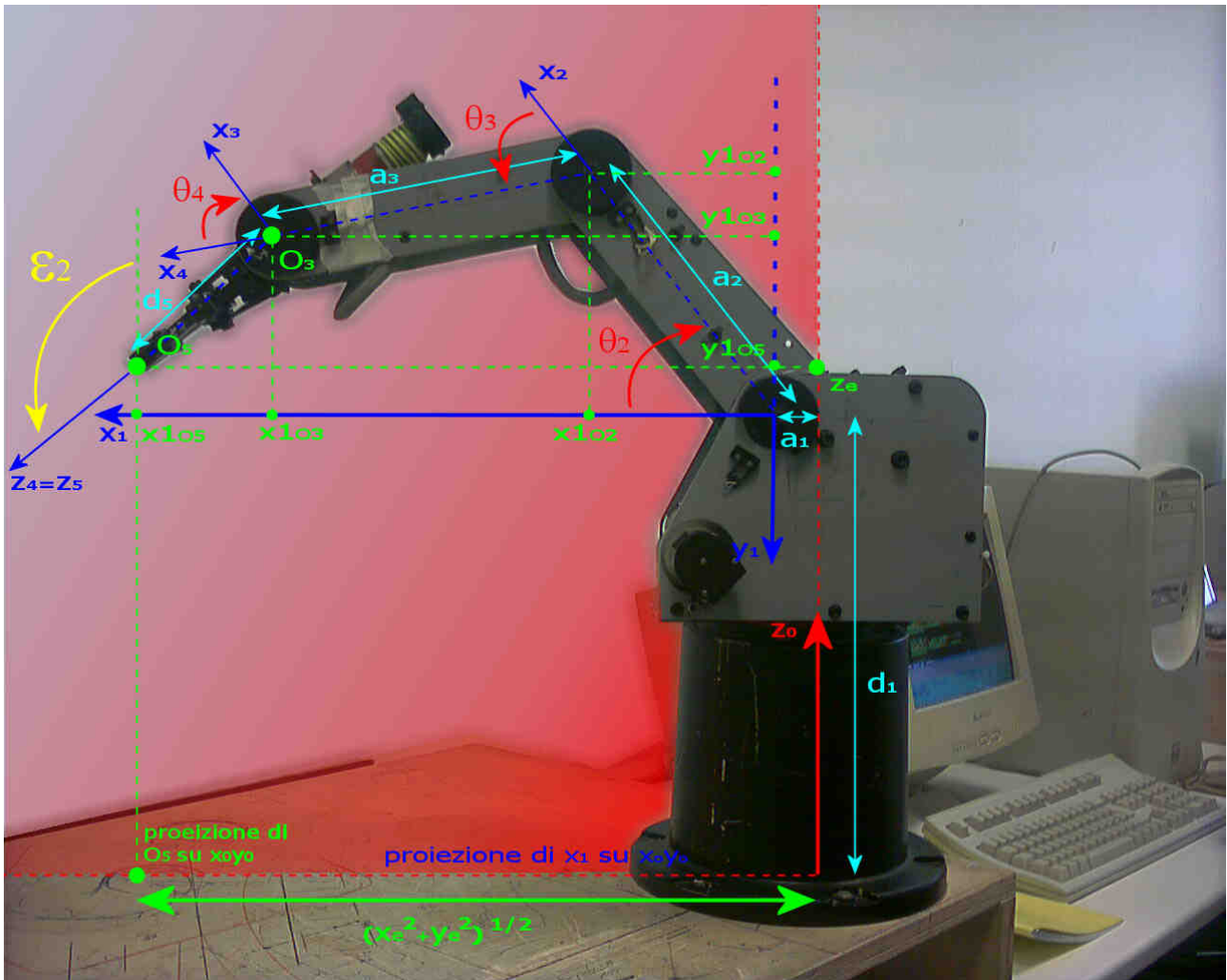
Se le coordinate X e Y dell'effettore sono entrambe pari a 0 l'angolo $\theta_1 = \text{eulerio1}$ viene scelto pari a 0° (la scelta è arbitraria) per evitare di incorrere in un caso critico della funzione $\text{atan2}(y_e, x_e)$.



L'effettore del robot ha dunque 5 gradi di libertà (e non 6 come per un generico corpo rigido nello spazio tridimensionale) nello spazio di lavoro destrorso e quindi il problema della cinematica inversa deve essere risolto per 5 parametri:

le 3 coordinate cartesiane x_e, y_e, z_e e gli angoli ϵ_2 e ϵ_3 .

Stabilito ϵ_1 , per ricavare dei possibili valori dei tre angoli che compaiono nell'equazione (2) è sufficiente valutare la configurazione planare dei sistemi di riferimento $(Oxyz)_1, (Oxyz)_2$ e $(Oxyz)_3$ sul piano X_1Y_1 .



Date le coordinate di effettore e l'angolo ε_2 , la posizione e l'orientamento del link 5 sul piano X_1Y_1 sono univocamente determinati e di conseguenza sono determinate anche le coordinate planari di O_3 conoscendo le coordinate planari di O_5 :

$$\begin{aligned}x_{1o_5} &= \sqrt{x_e^2 + y_e^2} - a_1 \\y_{1o_5} &= -(z_e - d_1)\end{aligned}$$

Si possono quindi ricavare valori possibili per θ_2 e θ_3 valutando esclusivamente le coordinate di O_3 :

$$\begin{aligned}x_{1o_3} &= x_{1o_5} - d_5 \sin(\varepsilon_2) \\y_{1o_3} &= y_{1o_5} + d_5 \cos(\varepsilon_2)\end{aligned}$$

Calcolando le coordinate di O_5 in funzione della lunghezza dei link e degli angoli di giunto si ottiene:

$$x_{1o_3} = a_2 \cos(\theta_2) + a_3 \cos(\theta_2 + \theta_3) \quad (4)$$

$$y_{1o_3} = a_2 \sin(\theta_2) + a_3 \sin(\theta_2 + \theta_3) \quad (5)$$

$$\begin{aligned}x_{1o_3}^2 + y_{1o_3}^2 &= a_2^2 (\cos^2(\theta_2) + \sin^2(\theta_3)) + a_3^2 (\cos^2(\theta_2 + \theta_3) + \sin^2(\theta_2 + \theta_3)) + \\&+ 2a_2a_3 (\cos(\theta_2)\cos(\theta_2 + \theta_3) + \sin(\theta_2)\sin(\theta_2 + \theta_3))\end{aligned}$$

Con opportune semplificazioni si ha:

$$x_{1o_3}^2 + y_{1o_3}^2 = a_2^2 + a_3^2 + 2a_2a_3 \cos(\theta_3)$$

$$\cos(\theta_3) = \frac{x_{1o_3}^2 + y_{1o_3}^2 - a_2^2 - a_3^2}{2a_2a_3}$$

$$\sin(\theta_3) = \pm \sqrt{1 - \cos^2(\theta_3)} \quad (+ \text{ per gomito alto, } - \text{ per gomito basso})$$

Utilizzando le formule di somma per seno e coseno nelle equazioni (4) e (5):

$$x_{1o_3} = a_2 \cos(\theta_2) + a_3 (\cos(\theta_2) \cos(\theta_3) - \sin(\theta_2) \sin(\theta_3))$$

$$y_{1o_3} = a_2 \sin(\theta_2) + a_3 (\cos(\theta_2) \sin(\theta_3) + \sin(\theta_2) \cos(\theta_3))$$

Rappresentabili in forma matriciale come

$$\begin{bmatrix} x_{1o_3} \\ y_{1o_3} \end{bmatrix} = \begin{bmatrix} -a_3 \sin(\theta_3) & a_2 + a_3 \cos(\theta_3) \\ a_2 + a_3 \cos(\theta_3) & a_3 \sin(\theta_3) \end{bmatrix} \begin{bmatrix} \sin(\theta_2) \\ \cos(\theta_2) \end{bmatrix}$$

oppure

$$\begin{bmatrix} \sin(\theta_2) \\ \cos(\theta_2) \end{bmatrix} = \begin{bmatrix} -a_3 \sin(\theta_3) & a_2 + a_3 \cos(\theta_3) \\ a_2 + a_3 \cos(\theta_3) & a_3 \sin(\theta_3) \end{bmatrix}^{-1} \begin{bmatrix} x_{1o_3} \\ y_{1o_3} \end{bmatrix}$$

A questo punto è possibile calcolare θ_2 e θ_3 utilizzando la funzione matematica atan2:

$$\theta_2 = \text{atan2}(\sin(\theta_2), \cos(\theta_2))$$

$$\theta_3 = \text{atan2}(\sin(\theta_3), \cos(\theta_3))$$

Dopo aver effettuato tali calcoli, gli angoli di giunto θ_2 e θ_3 , per la configurazione richiesta a gomito alto, diventano noti e l'angolo di giunto θ_4 si ricava sostituendo i valori di θ_2 e θ_3 ricavati nell'equazione (2)

$$\theta_4 = \varepsilon_2 - \theta_2 - \theta_3 - \pi$$

Infine si ottiene il valore di θ_5 utilizzando l'equazione (3):

$$\theta_5 = \varepsilon_3 - \pi$$

E' stato così risolto il problema della cinematica inversa. A partire da una terna di coordinate e una coppia di angoli, rispettivamente posizione cartesiana richiesta per l'origine O_5 e orientamento del sistema di riferimento (Oxyz)₅ entrambe rispetto a (Oxyz)₀, è stato possibile conoscere la quintupla $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5$ di angoli di giunto che permettesse di raggiungere la configurazione richiesta.

L'orientamento dell'effettore, come già detto, viene descritto solo da una coppia di angoli ($\varepsilon_2, \varepsilon_3$) e non da una terna ($\varepsilon_1, \varepsilon_2, \varepsilon_3$), come nel caso più generale, poiché il primo angolo di Eulero ε_1 risulta univocamente fissato dalla scelta della terna di coordinate cartesiane dell'effettore.

Si definisce equazione di cinematica inversa la seguente applicazione:

$$q = f[O_e, \varepsilon_2, \varepsilon_3]'$$

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \end{bmatrix} = \begin{bmatrix} a \tan 2(y_e, x_e) \\ a \tan 2(\sin(\theta_2), \cos(\theta_2)) \\ a \tan 2(\sin(\theta_3), \cos(\theta_3)) \\ \varepsilon_2 - \theta_2 - \theta_3 - \pi \\ \varepsilon_3 - \pi \end{bmatrix}$$

La formulazione del tutto esplicita della funzione di cinematica inversa è alquanto complessa e si ricava dai passaggi descritti sopra. Anche nel programma sviluppato si è evitato di implementare esplicitamente la funzione ma sono stati svolti tutti i passaggi.

6.3 Campo di Applicabilità

Condizione necessaria per la risoluzione del problema della cinematica inversa

$$\left| \frac{x_{1o_3}^2 + y_{1o_3}^2 - a_2^2 - a_3^2}{2a_2a_3} \right| < 1$$

in caso tale condizione non fosse soddisfatta non sarebbe possibile ottenere una configurazione del robot relativa alla posizione e all'orientamento richiesti per l'effettore (quindi fuori dallo spazio di lavoro destrorso).

Se le coordinate desiderate per l'effettore si trovassero fuori dallo spazio di lavoro destrorso ma comunque nello spazio di lavoro del robot sarebbe possibile far raggiungere all'effettore le coordinate richieste non considerando i vincoli relativi all'orientamento, in particolare all'angolo ε_2 .

7. Controllo della Posizione del Robot

Al fine di evitare possibili danneggiamenti delle parti meccaniche ed elettriche, dovuti a un utilizzo non previsto per il robot, è stata creata nel programma di gestione una procedura di controllo sulle operazioni richieste dall'utente.

Per mezzo di questa procedura si intende prevedere se la configurazione futura del robot sia ammissibile in accordo con le possibilità meccaniche dovute alla struttura del robot.

Nel caso in cui il robot non potesse effettuare le operazioni richieste, il programma bloccherebbe l'invio di segnali all'unità di potenza fino a quando non ricevesse una nuova operazione eseguibile, facendo visualizzare sullo schermo un messaggio di errore che chiarifichi il motivo dell'interruzione.

La procedura di controllo prevede diversi casi di errore:

7.1 Controllo sui limiti dei valori per gli angoli di D-H

- Questo controllo verifica se tutti i valori per gli angoli di giunto, relativi alla posizione successiva, sono compresi nei limiti massimi e minimi calcolati sperimentalmente per tali angoli:

$$-138^{\circ} < \theta_1 < 170^{\circ}$$

$$-127^{\circ} < \theta_2 < 30^{\circ}$$

$$-150^{\circ} < \theta_3 < 160^{\circ}$$

$$-200^{\circ} < \theta_4 < 20^{\circ}$$

Se un angolo di giunto non fosse consentito verrebbe stampato a schermo l'errore:
"Angolo Theta i non consentito."

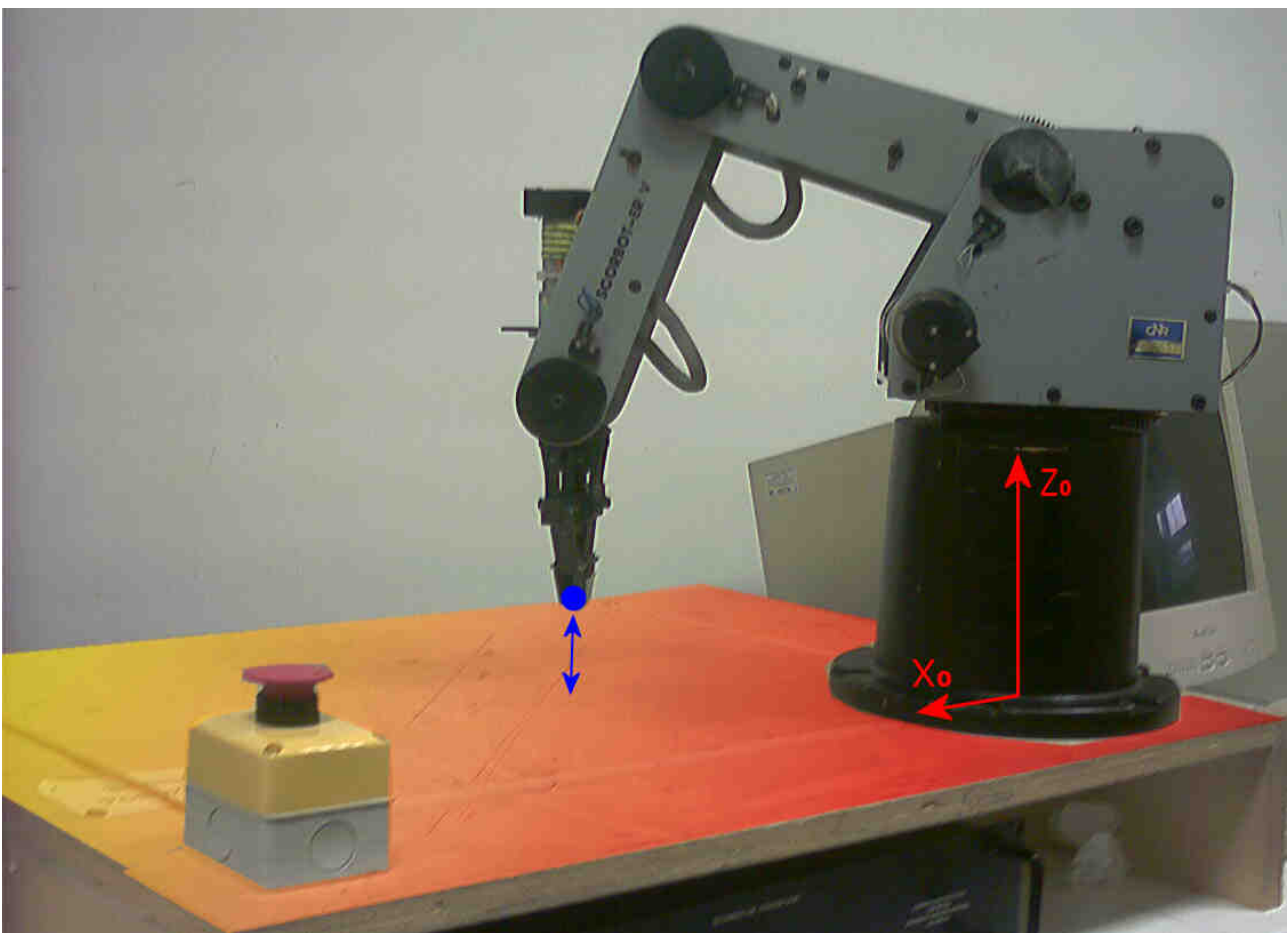
dove i è l'indice del primo angolo che non appartiene all'intervallo possibile.

7.2 Controllo sull'impatto con il piano di lavoro

- Questo controllo è finalizzato a impedire che una parte del robot tenti di posizionarsi al di sotto del piano di lavoro;
a poter cercare di raggiungere tale posizione sono in particolare l'effettore o la coda della pinza, per questo si esegue una verifica sulla coordinata z di entrambe le parti. Se almeno una parte provasse ad attraversare il piano di lavoro verrebbe stampato a schermo l'errore:

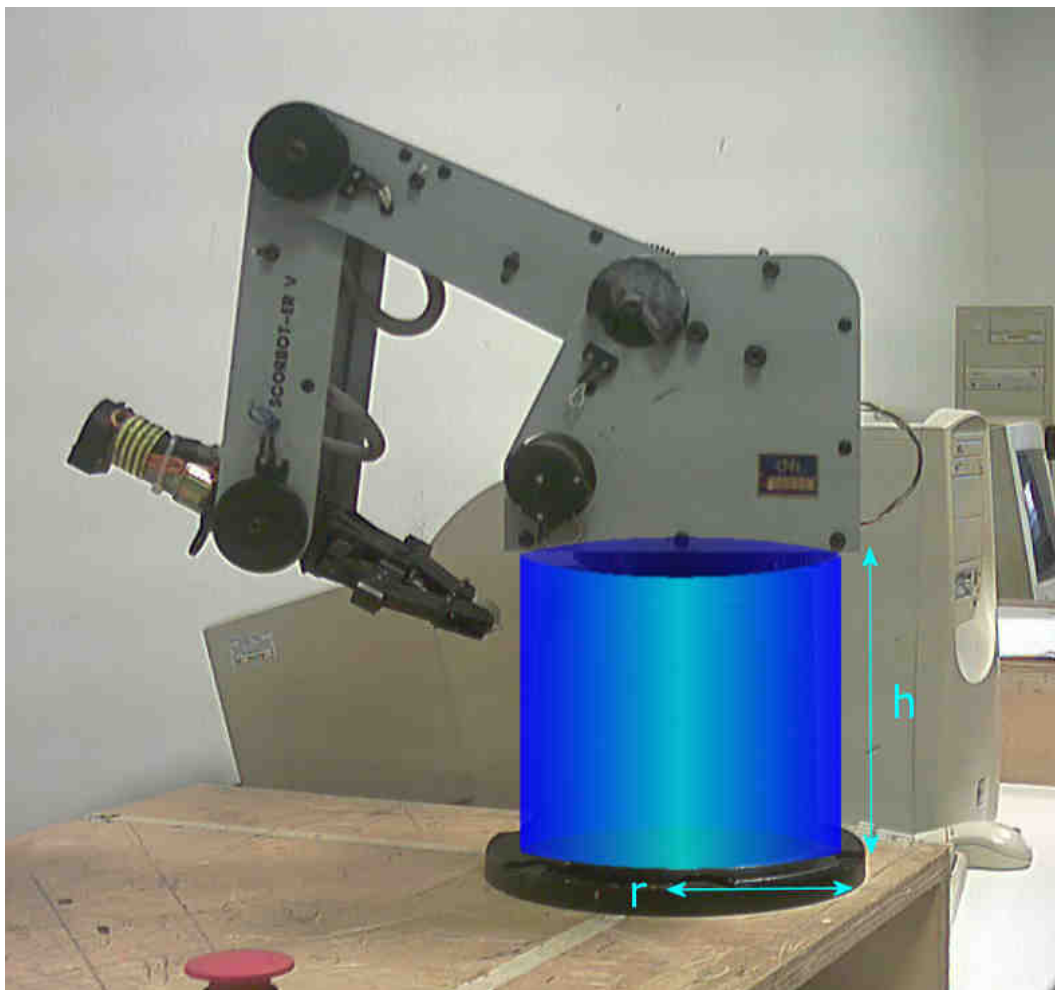
"Il robot si scontrerebbe con il piano di lavoro."

Il controllo viene eseguito sull'origine O_5 ed O_3 ed inoltre sull'estremità posteriore dell'effettore. Di questi punti viene controllata la non negatività della coordinata z nel sistema di riferimento di base $(Oxyz)_0$.



7.3 Controllo sull'impatto con il cilindro di base

- In questa fase si cerca di evitare che il robot colpisca il cilindro di sostegno posizionato alla sua base;
l'esame viene effettuato anche in questo caso dall'effettore alla coda della pinza:
si verifica prima che la coordinata z del punto da controllare sia minore dell'altezza del cilindro per poi controllare che tale punto sia esterno alla sezione circolare del cilindro relativa all'altezza z.
Se almeno una parte provasse a penetrare nel cilindro di base verrebbe stampato a schermo uno dei seguenti errori:
“L'effettore si scontrerebbe con il cilindro di base.”
“La coda della pinza si scontrerebbe con il cilindro di base.”
“Una parte della pinza si scontrerebbe con il cilindro di base.”



Per effettuare questo controllo si verifica che, quando $z_e < h$:

$$x_e^2 + y_e^2 > r^2$$

Lo stesso controllo viene effettuato con l'estremità posteriore del link di effettore.

7.4 Controllo sull'impatto con il link 2

- Tale controllo evita che l'effettore o la coda della pinza si scontrino o attraversino il link 2;

quello che si verifica è che la distanza del punto da controllare dalla retta passante per O_1 e O_2 sia maggiore di metà dello spessore dello scheletro metallico del link 2 (spessore simmetrico rispetto alla retta).

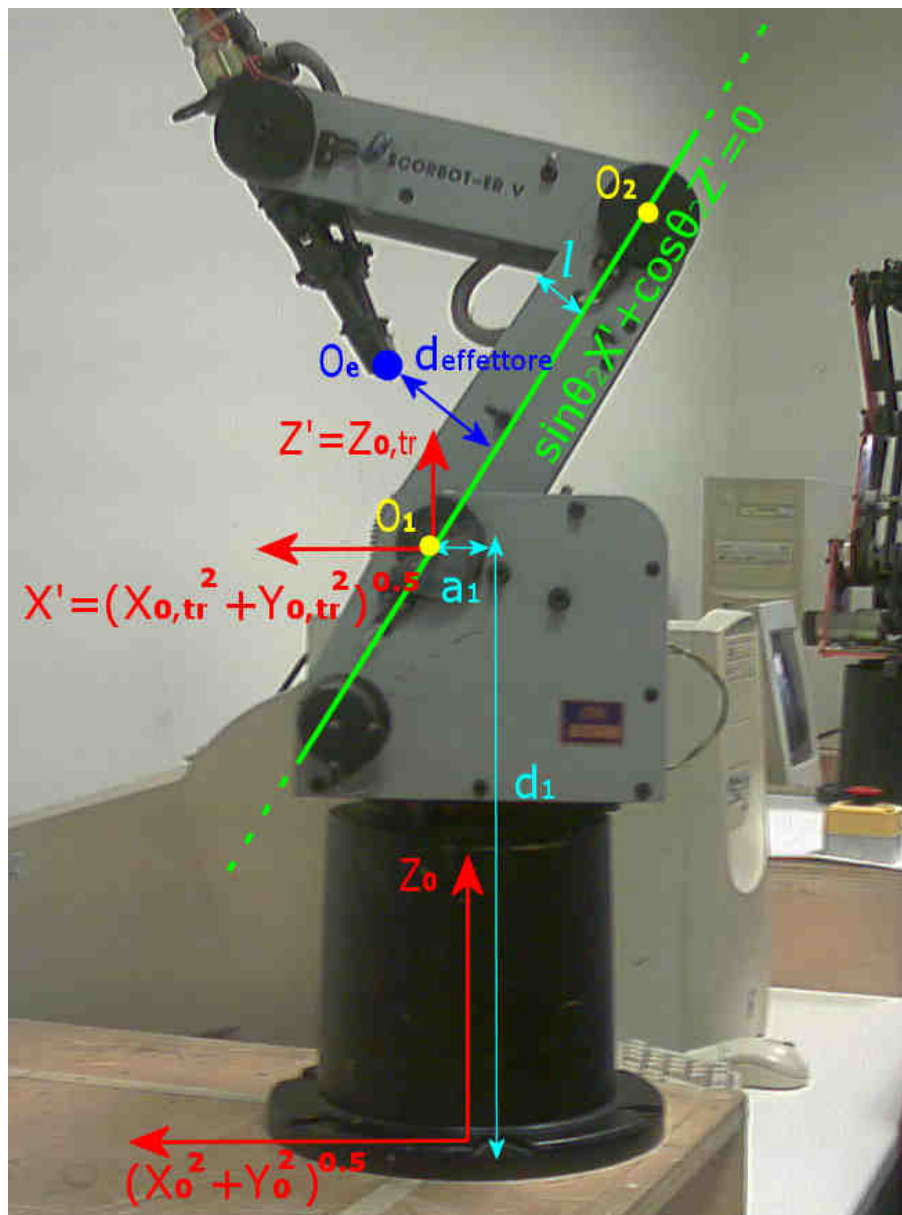
La verifica descritta è effettuata solo se l'angolo θ_3 è maggiore di 90° o minore di -90° (solo in questi casi infatti può avvenire lo scontro con il giunto).

Se almeno una parte tentasse di raggiungere il link 2 verrebbe stampato a schermo uno dei seguenti errori:

"L'effettore si scontrerebbe con il link 2."

"La coda della pinza si scontrerebbe con il link 2."

"Una parte della pinza si scontrerebbe con il link 2."



Per effettuare questo controllo si è centrato un sistema di riferimento nell'origine O_1 . Questo sistema di riferimento ha un'asse orizzontale x' parallelo all'asse $(x_0^2+y_0^2)^{0.5}$ a distanza d_1 , e un asse verticale z' parallelo a z_0 a distanza a_1 . In questo sistema di riferimento si ha che:

$$\begin{aligned} O_1 &= (0, 0) \\ O_2 &= (a_2 c_2, -a_2 s_2) \\ O_e &= (-d_5 s_{234} + a_3 c_{23} + a_2 c_2, -d_5 c_{234} - a_3 s_{23} - a_2 s_2) = (x_e', z_e') \end{aligned}$$

dove si ha:

$$\begin{aligned} c_2 &= \cos(\theta_2) & s_2 &= \sin(\theta_2) \\ c_{23} &= \cos(\theta_2 + \theta_3) & s_{23} &= \sin(\theta_2 + \theta_3) \\ c_{234} &= \cos(\theta_2 + \theta_3 + \theta_4) & s_{234} &= \sin(\theta_2 + \theta_3 + \theta_4) \end{aligned}$$

Questi risultati si ottengono da opportuni calcoli effettuati mediante le trasformazioni di coordinate omogenee che regolano le coordinate dei sistemi di riferimento $(Oxyz)_i$.

La retta passante per le origini O_1 e O_2 ha espressione:

$$\sin(\theta_2) \cdot x' + \cos(\theta_2) \cdot z' = 0$$

La distanza dell'origine O_e da questa retta è:

$$d_{\text{effettore}} = \frac{|\sin(\theta_2) \cdot x_{O_e}' + \cos(\theta_2) \cdot z_{O_e}'|}{\sqrt{\sin^2(\theta_2) + \cos^2(\theta_2)}} = |\sin(\theta_2) \cdot x_{O_e}' + \cos(\theta_2) \cdot z_{O_e}'|$$

Effettuando il controllo:

$$d_{\text{effettore}} > l$$

e ripetendo gli stessi calcoli e lo stesso controllo per l'estremità posteriore dell'effettore, si evita l'impatto con il link 2.

Le procedure di controllo della posizione del robot vengono eseguite prima dell'avvio di ogni procedura di movimentazione per testare la possibilità di eseguire o meno l'operazione richiesta dall'utente prima di far iniziare il movimento del robot.

Quando invece, mediante l'apposita funzionalità sviluppata nel progetto, si sta tracciando una traiettoria con l'estremità anteriore dell'effettore, le procedure di controllo vengono eseguite anche durante il movimento. La tecnica utilizzata per inseguire una traiettoria viene esposta nella seguente sezione.

8. Inseguimento di traiettorie

8.1 Equazioni parametriche di una retta nello spazio

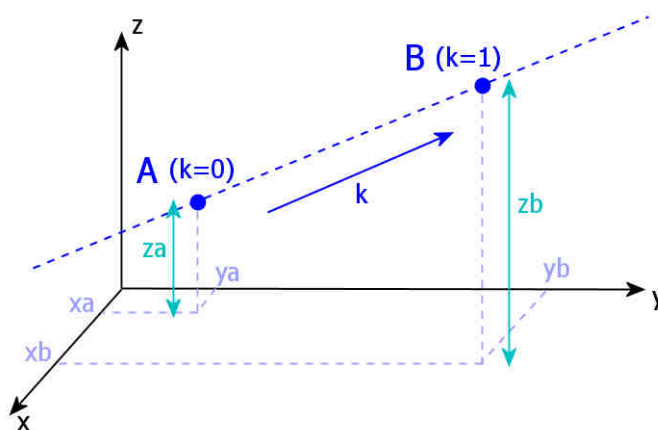
L'inseguimento di una traiettoria realizzato si basa sull'utilizzo ripetuto della funzione di cinematica inversa. Una traiettoria viene percorsa mediante un susseguirsi di applicazioni della funzione di cinematica inversa su una serie finita di punti successivi appartenenti alla traiettoria, con un orientamento fisso dell'effettore. Il primo problema da risolvere è stato quello di trovare delle equazioni, descriventi la traiettoria, dalle quali ricavare agevolmente le coordinate dei successivi punti, componenti la traiettoria desiderata.

Le traiettorie realizzate sono state di tipo rettilineo e di tipo circolare.

Le equazioni scelte per descrivere le traiettorie sono state le equazioni parametriche. Esse esprimono separatamente le coordinate di un punto generico della traiettoria in funzione di un unico parametro. Ad ogni punto della traiettoria, o luogo geometrico, è associato univocamente un valore del parametro.

Per una traiettoria rettilinea nello spazio tridimensionale, le equazioni parametriche sono:

$$\begin{aligned}x &= x_a + k \cdot (x_b - x_a) \\y &= y_a + k \cdot (y_b - y_a) \\z &= z_a + k \cdot (z_b - z_a)\end{aligned}$$



dove A è il punto (x_a, y_a, z_a), B (x_b, y_b, z_b), e k è il parametro variabile. Si può vedere che A è il punto con associato il valore del parametro k=0, e B il valore k=1. Si è così stabilito un senso di percorrenza della traiettoria. Sulla retta, spostandosi su punti successivi nella direzione AB il valore di k cresce, mentre decresce nella direzione opposta BA.

8.2 Normalizzazione della distanza tra punti successivi

Dalle equazioni parametriche si ricavano quindi punti successivi su di una traiettoria facendo crescere o diminuire il valore del parametro k. Inoltre risulta costante la distanza lineare, calcolata sulla traiettoria, tra punti successivi per incrementi costanti di k. Utilizzando le coordinate dei punti successivi, ricavate dalle equazioni parametriche, ed invocando la funzione di cinematica inversa in

maniera consecutiva su tali punti è stato possibile tracciare una traiettoria con l'estremità effettore del robot.

Nel generare la serie punti nello spazio di lavoro, su cui posizionare consecutivamente l'estremità dell'effettore del robot si è proceduto alla normalizzazione della distanza tra un punto ed il successivo. Questo è stato fatto per fare in modo che la precisione con la quale la traiettoria veniva inseguita non dipendesse dalla distanza tra i punti A e B per la traiettoria rettilinea.

Si è scelto per comodità che A fosse il punto di partenza e B quello di arrivo.

Le equazioni dalle quali si ricavano punti successivi a distanza unitaria su una traiettoria rettilinea AB sono:

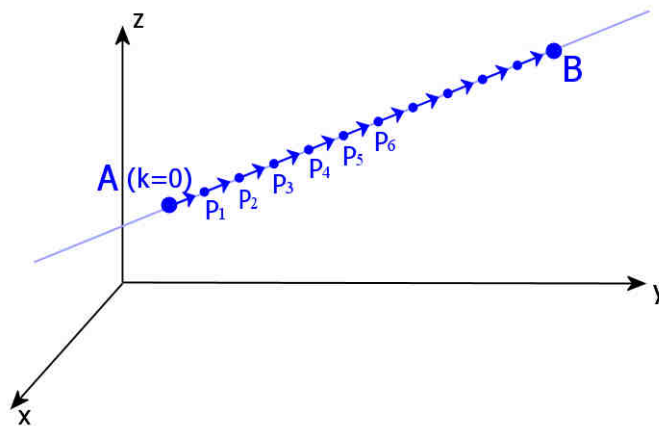
$$\begin{aligned}x &= x_a + k \cdot (x_b - x_a) / \text{norma} \\y &= y_a + k \cdot (y_b - y_a) / \text{norma} \\z &= z_a + k \cdot (z_b - z_a) / \text{norma}\end{aligned}$$

Dove:

$$\text{norma} = ((x_b - x_a)^2 + (y_b - y_a)^2 + (z_b - z_a)^2)^{0.5}$$

Ossia il valore norma è la distanza AB

Partendo da $k=0$, e quindi dal punto A, ed aumentando unitariamente il suo valore le equazioni appena scritte descrivono una serie di punti successivi che viaggiano in direzione AB sulla retta, tutti a distanza unitaria. Il programma sviluppato per l'inseguimento di una traiettoria rettilinea sfrutta queste equazioni per ricavare i successivi punti su cui posizionare l'effettore mediante la funzione di cinematica inversa, partendo dal punto A, fino ad arrivare a B.



Per eseguire spostamenti, su una traiettoria rettilinea, tra punti ad una distanza arbitraria *passo* si sono modificate le equazioni nel modo seguente:

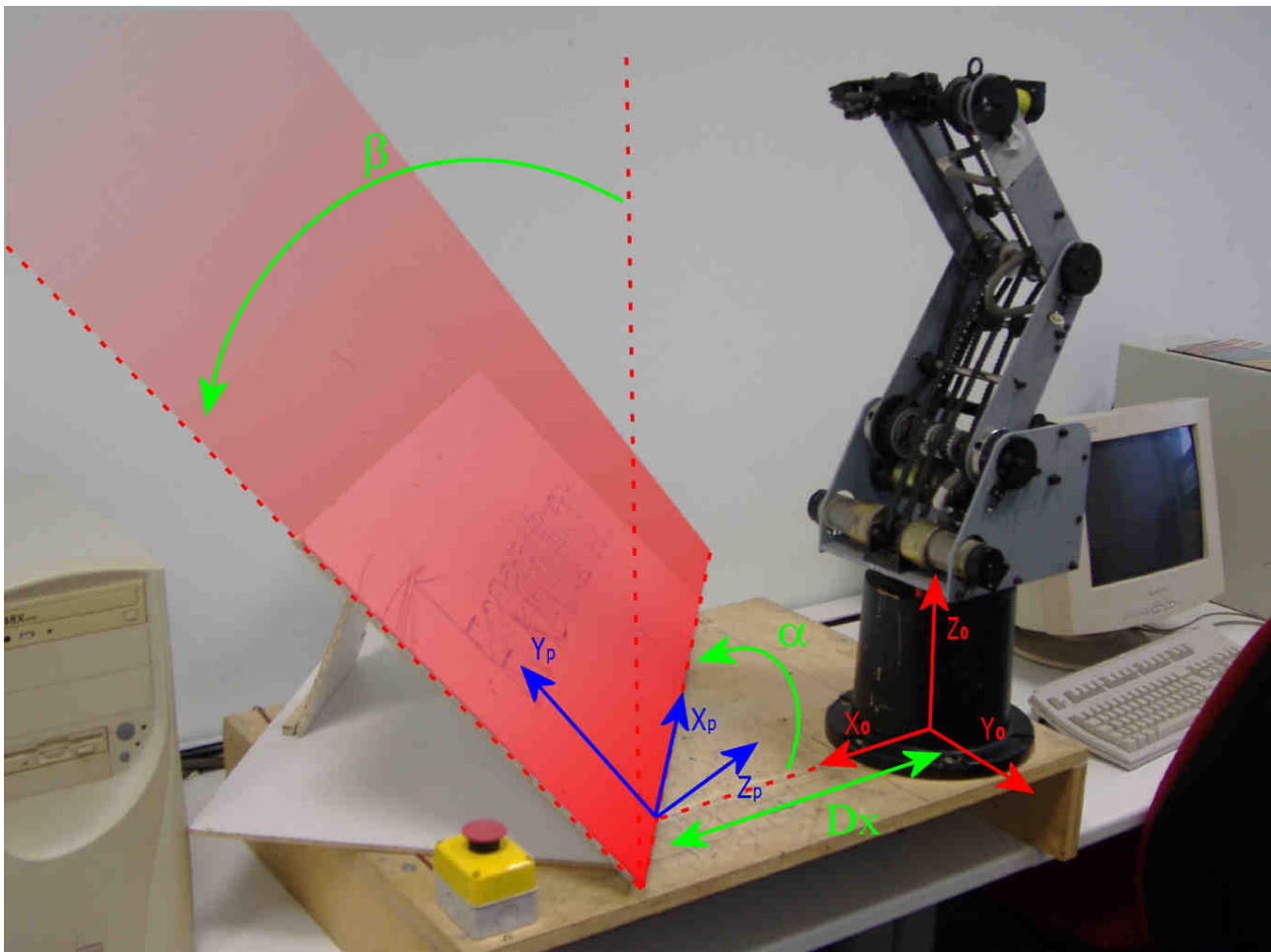
$$\begin{aligned}x &= x_a + k \cdot \text{passo} \cdot (x_b - x_a) / \text{norma} \\y &= y_a + k \cdot \text{passo} \cdot (y_b - y_a) / \text{norma} \\z &= z_a + k \cdot \text{passo} \cdot (z_b - z_a) / \text{norma}\end{aligned}$$

8.3 Inseguimento di traiettorie su un piano

Oltre all'inseguimento di una retta nello spazio si è realizzato anche l'inseguimento di traiettorie planari, residenti cioè totalmente su un piano dato. Le traiettorie planari realizzate sono state di tipo rettilineo e circolare. In particolare per una traiettoria circolare si capisce come sia necessario l'identificazione del piano su cui descrivere la traiettoria.

Per prima cosa si è proceduto alla descrizione del piano sul quale descrivere le traiettorie. Questo è stato fatto ponendo sul piano un sistema di riferimento cartesiano $(Oxyz)_P$, i cui assi x_P e y_P giacciono sul piano P e il cui asse z è uscente dal piano in direzione del robot.

Il piano P è stato individuato in relazione al sistema di riferimento di base $(Oxyz)_0$ tramite tre parametri α , β , Dx , che sono rispettivamente due angoli ed una distanza lineare. L'angolo α è l'angolo compreso tra l'asse x_0 (nella direzione delle x_0 decrescenti) e la retta di intersezione tra il piano P e il piano x_0y_0 . L'angolo β è l'angolo individuato dal piano P e la normale al piano di lavoro ovvero x_0y_0 . Dx è la distanza lineare staccata sull'asse x_0 dal piano P.

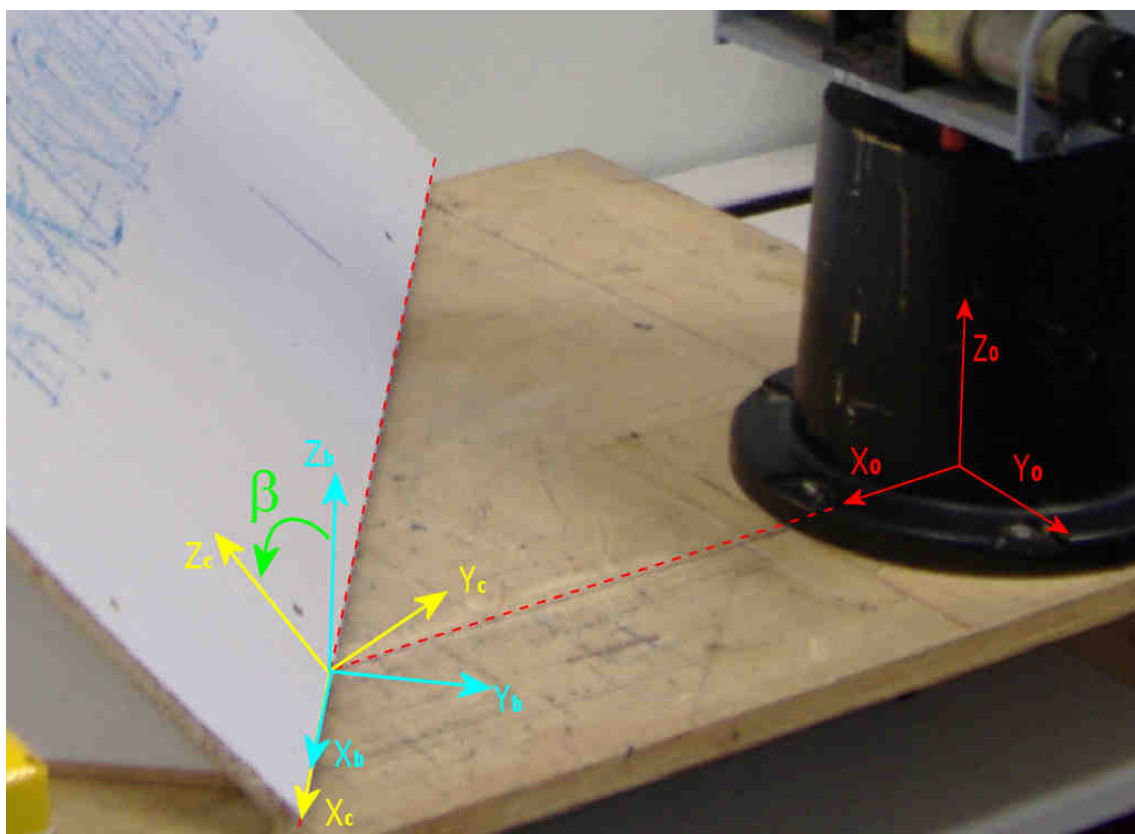
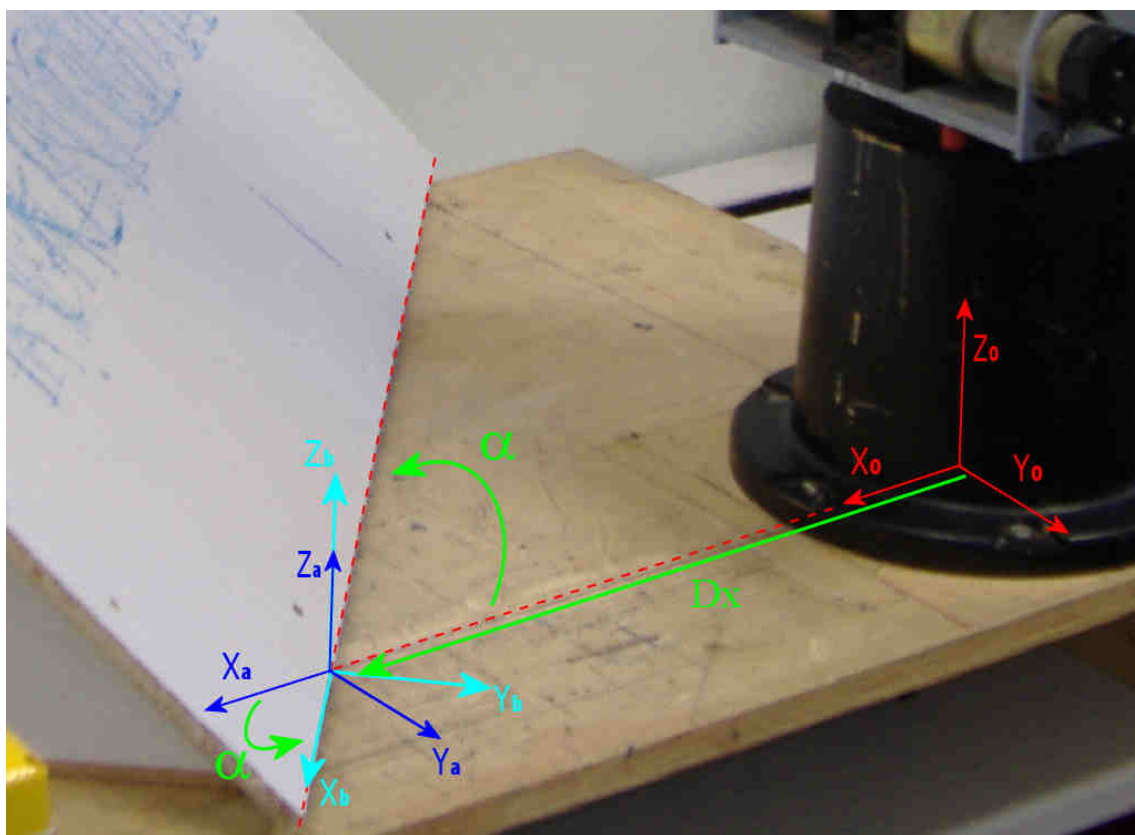


Per descrivere la trasformazione di coordinate tra il sistema di riferimento $(Oxyz)_0$ e $(Oxyz)_P$ si è ricorso alle trasformazioni di coordinate omogenee.

Le successive trasformazioni di coordinate per sovrapporre il sistema di riferimento $(Oxyz)_0$ su $(Oxyz)_P$ e quindi descrivere la trasformazione di coordinate omogenee sono:

1. Traslazione lungo l'asse x di Dx

2. Rotazione intorno all'asse z corrente di α
3. Rotazione intorno all'asse x corrente di β
4. Rotazione intorno all'asse y corrente di π
5. Rotazione intorno all'asse x corrente di $-\pi/2$



Le matrici di trasformazione omogenea che descrivono le cinque trasformazioni sopra elencate sono:

$$\begin{aligned}
 p_0 &= {}^0T_a p_h & {}^0T_a &= \begin{bmatrix} 1 & 0 & 0 & Dx \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_a &= {}^aT_b p_b & {}^aT_b &= \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_b &= {}^bT_c p_c & {}^bT_c &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\beta) & -\sin(\beta) & 0 \\ 0 & \sin(\beta) & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_c &= {}^cT_d p_d & {}^cT_d &= \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 p_d &= {}^dT_P p_P & {}^dT_P &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Operando opportune sostituzioni si è ricavata la trasformazione dalle coordinate omogenee p_P del sistema di riferimento $(Oxyz)_P$ fissato sul piano P, e le coordinate omogenee p_0 del sistema di riferimento di $(Oxyz)_0$.

$$\begin{aligned}
 p_0 &= {}^0T_P p_P \\
 {}^0T_P &= {}^0T_a {}^aT_b {}^bT_c {}^cT_d {}^dT_P \\
 {}^aT_b &= \begin{bmatrix} -\cos(\alpha) & \sin(\alpha)\sin(\beta) & -\sin(\alpha)\cos(\beta) & Dx \\ -\sin(\alpha) & -\cos(\alpha)\sin(\beta) & \cos(\alpha)\cos(\beta) & 0 \\ 0 & \cos(\beta) & \sin(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Identificato il piano P tramite il sistema di riferimento cartesiano $(Oxyz)_P$, si sono potute esprimere le traiettorie tramite le equazioni parametriche. Le equazioni parametriche saranno ora relative alle coordinate $(Oxyz)_P$, e la coordinata z_P del sistema di riferimento $(Oxyz)_P$ sarà costantemente nulla, per descrivere appunto una traiettoria residente sul piano. Quindi per una traiettoria sul piano P le

uniche coordinate variabili nel sistema di riferimento $(Oxyz)_P$ saranno la coordinata x_P e la coordinata y_P . Le coordinate in $(Oxyz)_P$ della serie di punti generati dall'applicazione delle equazioni parametriche saranno convertite per mezzo dell'equazione $p_0 = T_{0P} p_P$ in coordinate del sistema $(Oxyz)_0$, per poi essere utilizzate nella funzione di cinematica inversa, e per descrivere quindi la traiettoria sul piano P.

8.4 Traiettorie rettilinee su un piano

Per la descrizione di traiettorie rettilinee sul piano P, la trattazione è identica a quella delle traiettorie nello spazio, con la differenza che l'ultima coordinata z è costantemente nulla.

$$\begin{aligned}x_P &= x_a + k \cdot (x_b - x_a) / \text{norma} \\y_P &= y_a + k \cdot (y_b - y_a) / \text{norma} \\z_P &= 0\end{aligned}$$

dove A è il punto $(x_a, y_a, 0)$, B $(x_b, y_b, 0)$, espressi nelle coordinate del sistema $(Oxyz)_P$, e k è il parametro variabile, e norma è sempre la distanza AB.

E' stata anche realizzata una funzione che mantenendo costanti le coordinate x_P e y_P , varia la coordinata z_P per posizionarsi su un'altezza differente del piano:

$$\begin{aligned}x_P &= x_a \\y_P &= y_a \\z_P &= z_a + k \cdot (z_b - z_a) / \text{norma}\end{aligned}$$

8.5 Traiettorie circolari su un piano

8.5.1 Equazioni parametriche di traiettorie circolari

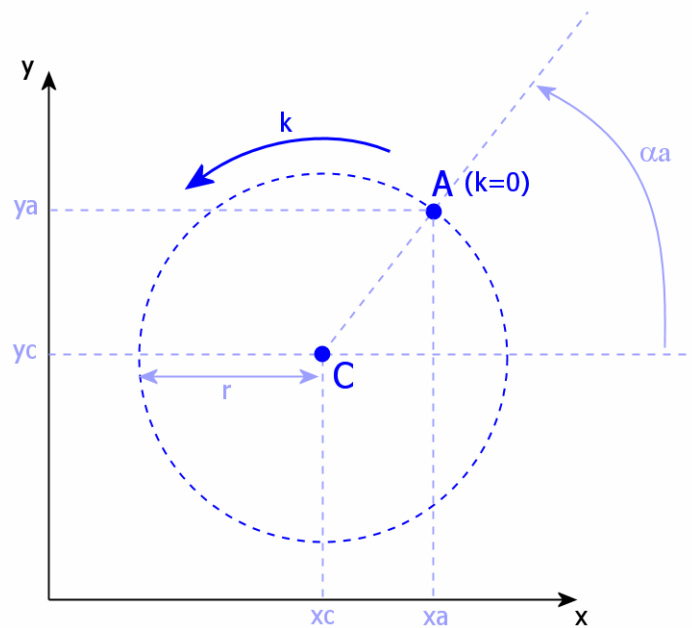
Per una traiettoria circolare nello spazio bidimensionale (il piano P) le equazioni parametriche sono:

$$\begin{aligned}x_P &= x_c + r \cdot \cos(k + \alpha_a) \\y_P &= y_c + r \cdot \sin(k + \alpha_a)\end{aligned}$$

la terza coordinata z_P rimane costantemente nulla per mantenersi sul piano P.

$$z_P = 0$$

A è il punto $(x_a, y_a, 0)$, C $(x_c, y_c, 0)$, espressi nelle coordinate del sistema $(Oxyz)_P$, ed α_a è l'angolo iniziale che forma la semiretta parallela all'asse x_P positivo applicata in C, con la semiretta AC. Si può vedere che A è il punto con associato il valore del parametro $k=0$, mentre per valori di k crescenti il punto si sposta sulla circonferenza in senso antiorario, mentre per valori di k decrescenti in senso orario. Si è così stabilito un senso di percorrenza della traiettoria. Sulla circonferenza, spostandosi su punti successivi in senso antiorario il valore di k cresce, mentre decresce spostandosi in senso orario.

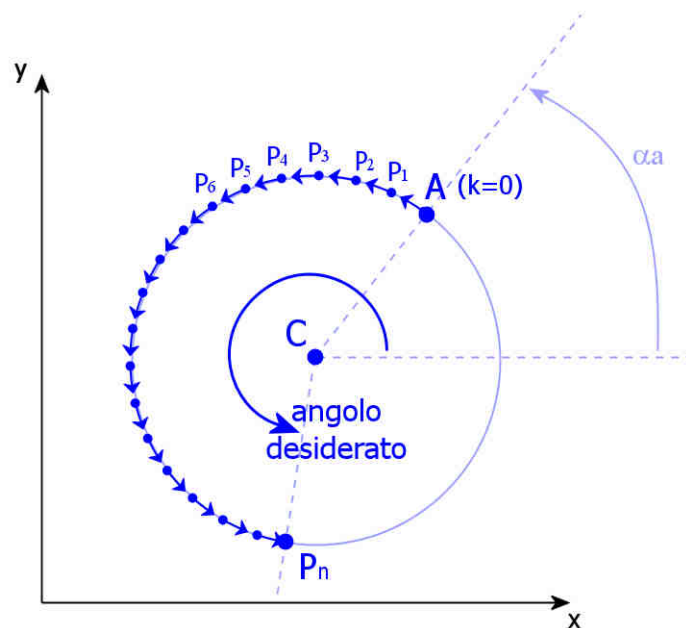


Per le traiettorie circolari invece di un punto di arrivo si è scelto di specificare un angolo desiderato da raggiungere (sempre formato dalla semiretta parallela all'asse x positivo applicata in C, con la semiretta AC, dove A si è spostato dell'angolo desiderato dalla posizione iniziale) .

Partendo da $k=0$, e quindi dal punto A, ed aumentando unitariamente il suo valore le equazioni appena scritte descrivono una serie di punti successivi che viaggiano in senso antiorario, tutti associati ad archi di circonferenza di lunghezza unitaria. Diminuendo unitariamente il valore di k , il senso di rotazione sarà invece orario. Il programma sviluppato per l'inseguimento di una traiettoria circolare sfrutta queste equazioni per ricavare i successivi punti su cui posizionare l'effettore mediante la funzione di cinematica inversa, partendo dal punto A, fino ad arrivare al punto P che individua, sulla circonferenza, un angolo pari a quello desiderato.

8.5.2 Normalizzazione della lunghezza dell'arco associato a due punti successivi

Nel generare la serie punti della traiettoria circolare, su cui posizionare consecutivamente l'estremità dell'effettore del robot, a seguito della trasformazione di coordinate omogenee, si è proceduto alla normalizzazione della lunghezza archi di circonferenza percorsi. Questo è stato fatto per fare in modo che la precisione con la quale la traiettoria veniva inseguita non dipendesse dalla distanza tra i punti dalla distanza tra i punti A e C (raggio della circonferenza).



Le equazioni dalle quali si ricavano punti successivi su una traiettoria circolare, la cui lunghezza dell'arco di circonferenza associato è unitaria, sono:

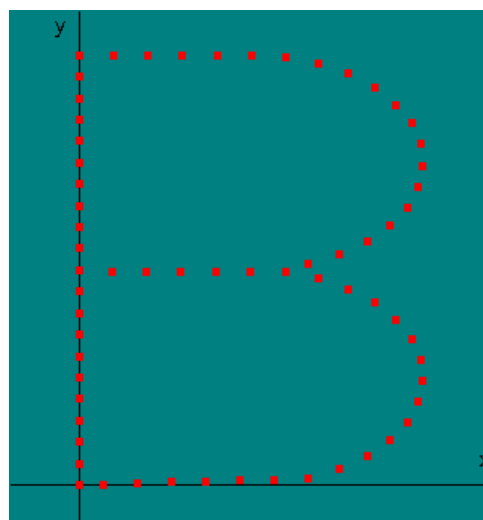
$$\begin{aligned}x &= x_c + r \cdot \cos(k / r + \alpha_a) \\y &= y_c + r \cdot \sin(k / r + \alpha_a)\end{aligned}$$

Per eseguire spostamenti, su una traiettoria circolare, tra punti associati ad archi di circonferenza di lunghezza arbitraria *passo*, si sono modificate le equazioni nel modo seguente:

$$\begin{aligned}x &= x_c + r \cdot \cos((k \cdot \text{passo}) / r + \alpha_a) \\y &= y_c + r \cdot \sin((k \cdot \text{passo}) / r + \alpha_a)\end{aligned}$$

8.6 Esempio di scrittura di una lettera

La scrittura di una lettera sul piano P è stata realizzata come una applicazione successiva di traiettorie rettilinee e circolari sul piano P. Un esempio viene qui riportato. Lettera B:



1. Ci si posiziona su un punto del piano P ad esempio $(0, 0, 0)$
2. Si effettua una traiettoria rettilinea da $(0, 0, 0)$ fino a $(0, 2, 0)$
3. Si effettua una traiettoria rettilinea da $(0, 2, 0)$ fino a $(0.5, 2, 0)$
4. Si effettua una traiettoria circolare di centro $(0.5, 2, 0)$ fino a $(0.5, 1.5, 0)$. Il punto di partenza è quello precedentemente raggiunto $(0.5, 2, 0)$ relativo ad un angolo di partenza di $\pi/2$. L'angolo di arrivo è $3/2\pi$. Il senso di rotazione è orario.
5. Si effettua una traiettoria rettilinea da $(0.5, 1, 0)$ fino a $(0, 1, 0)$
6. Si effettua una traiettoria rettilinea da $(0, 1, 0)$ fino a $(0.5, 1, 0)$
7. Si effettua una traiettoria circolare di centro $(0.5, 1, 0)$ fino a $(0.5, 0.5, 0)$. Il punto di partenza è quello precedentemente raggiunto $(0.5, 1, 0)$ relativo ad un angolo di partenza di $\pi/2$. L'angolo di arrivo è $3/2\pi$. Il senso di rotazione è orario.
8. Si effettua una traiettoria rettilinea fino da $(0.5, 0, 0)$ a $(0, 0, 0)$

9. Implementazione della funzione di cinematica diretta

Si riporta nella presente relazione un codice ad alto livello, che trascura alcuni particolari del linguaggio ANSI C ma non omette alcun passaggio, per permettere una comprensione più immediata della tecnica di programmazione utilizzata.

Il primo passo, come detto, è consistito nell'implementazione dell'equazione (pluto) a pagina 23, tramite la funzione *calcola_impulsi()*. Questa funzione utilizza i valori di *θ1desiderato*, *θ2desiderato*, *θ3desiderato*, *θ4desiderato*, *θ5desiderato* per calcolare i corretti passi encoder per ogni giunto. I valori da essa restituiti non vengono immediatamente inseriti nei campi del vettore *ROBOTDefCtrlPosReq[]*, ma vengono salvati nelle variabili *impulso[i]*. Questo perché la modifica dei valori *ROBOTDefCtrlPosReq[i]*, viene effettuata con accorgimenti specifici da una funzione separata, la funzione *daje()*.

```
impulso[] calcola_impulsi ( θdesiderato[], θhome[], rapporto[] )
{
    impulso1 = (θ1desiderato - θ1home)/rapporto1;
    impulso2 = (θ2desiderato - θ2home)/rapporto2;
    impulso3 = (θ2desiderato - θ2home)/rapporto3 + (θ3desiderato - θ3home)/rapporto3;
    impulso4 = (θ2desiderato - θ2home)/rapporto4 + (θ3desiderato - θ3home)/rapporto4 + (θ4desiderato - θ4home)/rapporto1;
    impulso5 = (θ5desiderato - θ5home)/rapporto5;

    return impulso[];
}
```

La funzione *daje()* richiede come argomenti due valori *tolleranza* e *secondi*, nonché il i valori dell'array *impulso[]*, risultato di *calcola_impulsi()*. Essa, prima di andare a modificare i valori di *ROBOTDefCtrlPosReq[]*, attende che finisca un precedente movimento. Questa attesa viene realizzata attendendo che, per ogni giunto, il valore assoluto della differenza tra posizione attuale e posizione richiesta in passi encoder ($|ROBOTGetLinkEncoder(i) - ROBOTDefCtrlPosReq[i]|$) diventi minore o uguale ad una certa soglia che viene specificata in *tolleranza*. Al fine di evitare che il programma rimanga bloccato nel ciclo, ad esempio a causa di un movimento non concluso, il ciclo while esegue un controllo (realizzato sfruttando il periodo di Real Time) sul tempo trascorso dall'inizio del ciclo. Nel caso che sia trascorso, dall'inizio del ciclo, un tempo superiore o uguale a *secondi*, questo viene fatto terminare tramite l'istruzione C *break*. Una volta terminato il ciclo, per una delle due ragioni sopra esposte, viene comandata ai giunti di 1, 2, 3, 4 la posizione in passi encoder calcolata e memorizzata in *impulso[]* da *calcola_impulsi()*. Viene poi realizzata un'attesa uguale a quella precedente relativamente solo al giunto 4 o di BECCHEGGIO, prima di inserire in *ROBOTDefCtrlPosReq[5]* il valore *impulso[5]*.

Le due operazioni di modifica di *ROBOTDefCtrlPosReq[4]* e di *ROBOTDefCtrlPosReq[5]* non vengono eseguite simultaneamente per evitare che i movimenti dei giunti di BECCHEGGIO e di ROLLIO avvenga contemporaneamente. Infatti essendo i movimenti dei due giunti realizzati mediante la diversa combinazione degli stessi motori, la modifica simultanea delle posizioni di giunti provoca un movimento non desiderato (un misto dei due movimenti o uno solo di essi da come si è potuto osservare prima di introdurre questo accorgimento essenziale).

Infine prima di terminare la funzione viene di nuovo atteso che tutti i giunti abbiano raggiunto la posizione richiesta, dopodiché la funzione *daje()* termina.

```
void daje( impulso[], tolleranza, secondi )
{
    while ( |ROBOTGetLinkEncoder(i) - ROBOTDefCtrlPosReq[i]| > tolleranza  ∀i = 1,2,3,4,5 )
```

```

    {
        if ( é trcorso dall'inizio del ciclo while un tempo  $\geq$  secondi )
            esci dal ciclo while;
    }

    ROBOTDefCtrlPosReq[1] = impulso1;
    ROBOTDefCtrlPosReq[2] = impulso2;
    ROBOTDefCtrlPosReq[3] = impulso3;
    ROBOTDefCtrlPosReq[4] = impulso4;

    while ( |ROBOTGetLinkEncoder(4) - ROBOTDefCtrlPosReq[4]| > tolleranza )
    {
        if ( é trcorso dall'inizio del ciclo while un tempo  $\geq$  secondi )
            esci dal ciclo while;
    }

    ROBOTDefCtrlPosReq[5] = impulso5

    while ( |ROBOTGetLinkEncoder(i) - ROBOTDefCtrlPosReq[i]| > tolleranza  $\forall i = 1, 2, 3, 4, 5$  )
    {
        if ( é trcorso dall'inizio del ciclo while un tempo  $\geq$  secondi )
            esci dal ciclo while;
    }
}

```

Lo stesso programma, al termine del movimento, calcola gli angoli di giunto q effettivamente raggiunti, inserendo nell'equazione (pippo) a pagina 23, nel vettore *impulsi*, i valori restituiti dalla funzione *ROBOTGetLinkEncoder()*. La funzione adibita a questo compito è *calcola_theta_attuali()*.

```

thetauale[] calcola_theta_attuali (  $\theta_{home}[]$ , rapporto[] )
{
     $\theta_{1attuale} = \text{ROBOTGetLinkEncoder}(1) * \text{rapporto1} + \theta_{1home}$ ;
     $\theta_{1attuale} = \text{ROBOTGetLinkEncoder}(2) * \text{rapporto2} + \theta_{2home}$ ;
     $\theta_{1attuale} = \text{ROBOTGetLinkEncoder}(3) * \text{rapporto3} + \theta_{3home} - \text{ROBOTGetLinkEncoder}(2) * \text{rapporto2}$ ;
     $\theta_{1attuale} = \text{ROBOTGetLinkEncoder}(4) * \text{rapporto4} + \theta_{4home} - \text{ROBOTGetLinkEncoder}(3) * \text{rapporto3}$ ;
     $\theta_{1attuale} = \text{ROBOTGetLinkEncoder}(5) * \text{rapporto5} + \theta_{5home}$ ;

    return  $\theta_{attuale}[]$ ;
}

```

Conoscendo la quintupla di angoli di giunto effettivamente raggiunti dal robot, il programma sviluppato passa allo svolgimento dell'equazione di cinematica diretta, risalendo quindi alla posizione e all'orientamento dell'effettore, rispetto a (Oxyz)₀, al termine del movimento.

```

coordinata_effettore[] calcola_coordinate_effettore (  $\theta_{attuale}[]$ , a1, a2, a3, d1, d5 )
{
     $\text{coordinata\_effettore1} = \cos(\theta_{attuale1}) * ( - \sin( \theta_{attuale2} + \theta_{attuale3} + \theta_{attuale4} ) * d5 + a3 * \cos( \theta_{attuale2} + \theta_{attuale3} ) + a2 * \cos( \theta_{attuale2} ) + a1 )$ ;
     $\text{coordinata\_effettore2} = \sin( \theta_{attuale1} ) * ( - \sin( \theta_{attuale2} + \theta_{attuale3} + \theta_{attuale4} ) * d5 + a3 * \cos( \theta_{attuale2} + \theta_{attuale3} ) + a2 * \cos( \theta_{attuale2} ) + a1 )$ ;
     $\text{coordinata\_effettore3} = - \cos( \theta_{attuale2} + \theta_{attuale3} + \theta_{attuale4} ) * d5 - a3 * \sin( \theta_{attuale2} + \theta_{attuale3} ) - a2 * \sin( \theta_{attuale2} ) + d1$ ;
};

eulero[] calcola_eulero (  $\theta_{attuale}[]$  )
{
     $\text{eulero1} = \theta_{attuale1}$ ;
     $\text{eulero2} = \theta_{attuale2} + \theta_{attuale3} + \theta_{attuale4} + \pi$ ;
     $\text{eulero3} = \theta_{attuale5} + \pi$ ;
};

```

10. Implementazione della Funzione di Cinematica Inversa

La funzione che risolve il problema della cinematica inversa ha come parametri in ingresso un vettore delle le coordinate e un vettore degli angoli di Eulero desiderati per il sistema di riferimento di effettore.

Se la posizione richiesta appartiene allo spazio di lavoro destrorso il vettore degli angoli di giunto desiderati viene modificato, e la funzione restituisce il valore 1 (che indica che il calcolo di tali angoli è andato a buon fine), altrimenti tale vettore resta invariato, la funzione restituisce il valore 0 e viene stampato a schermo il messaggio di errore: "Cinematica Inversa fallita;".

Per fare in modo che il robot si porti alla configurazione richiesta deve poi essere invocata la funzione di cinematica diretta con la relativa procedura di movimentazione dopo aver controllato che gli angoli di giunto rappresentino una posizione consentita (controllo della posizione).

```
risultato cinematica_inversa (coord_eff_desid[], ε_desid[], θdesiderato[])
{
    xIO3 = ( coord_eff_desid12 + coord_eff_desid22 ) - a1 - (d5 sin(ε_desid2))0.5;
    yIO3 = coord_eff_desid3 - d1 + (d5 cos(ε_desid2));

    c3 = (xIO32 + yIO32 - a22 - a32) / 2 a22;

    if ( (c3>1) || (c3<-1) ) //se non siamo nel campo di applicabilità
    {
        return 0; //restituisce risultato=0
    }

    s3 = (1-c32)0.5;

    θdesiderato3 = atan2(s3, c3);

    if ( coord_eff_desid1=0 && coord_eff_desid2=0 ) //caso critico di atan2()
        θdesiderato1 = 0.0;
    else
        θdesiderato1 = atan2(coord_eff_desid1, coord_eff_desid2);
        //individuo il piano per la cinematica inversa

    s2 = -xIO3 (a3 sin(θdesiderato3) / (2 a2 a3 cos(θdesiderato3) + a22 + a32)) + yIO3 ((a3 cos(θdesiderato3) + a2) / (2 a2 a3 cos(θdesiderato3) + a22 + a32));
    c2 = yIO3 (a3 sin(θdesiderato3) / (2 a2 a3 cos(θdesiderato3) + a22 + a32)) + xIO3 ((a3 cos(θdesiderato3) + a2) / (2 a2 a3 cos(θdesiderato3) + a22 + a32));

    θdesiderato2 = atan2(s2, c2);

    θdesiderato4 = ε_desid2 - θdesiderato2 - θdesiderato3 - π;

    θdesiderato5 = ε_desid3 - π;

    return 1; // restituisci risultato=1, cinematica inversa realizzata
};
```

11. Descrizione dello pseudolinguaggio di movimentazione del robot

Segue una descrizione dei comandi dello pseudolinguaggio da noi implementato, con possibilità di essere chiamati direttamente dal prompt durante l'esecuzione del programma, o essere scritti in modo sequenziale in un file di testo, da dare poi in lettura al programma stesso.

In neretto sono evidenziati i comandi, con la relativa sintassi e tipo di parametri; alla fine di un comando è necessaria la pressione del tasto [INVIO], senza lasciare spazi alla fine dei parametri.

La lista dei comandi è accessibile anche dal menu principale del programma.

mg $\theta_1 \theta_2 \theta_3 \theta_4 \theta_5$	comando Muovi Giunti , richiede gli angoli di giunto e posiziona i giunti del robot nella posizione desiderata.
mc $x_e y_e z_e \varepsilon_2 \varepsilon_3$	comando Muovi Coordinate , richiede le coordinate della punta dell'effettore nel sistema di riferimento di base $(Oxyz)_0$ e gli angoli di Eulero ε_2 ed ε_3 che descrivono l'orientamento del link di effettore, e posiziona se possibile il robot nella configurazione richiesta.
e2 ε_2 passo_angolare	comando Eulero2 , richiede l'angolo ε_2 di orientamento del link di effettore rispetto alla normale al piano di base posiziona il robot nella configurazione richiesta senza alterare nel movimento le coordinate della punta dell'effettore. Richiede inoltre il passo angolare con il quale raggiungere la posizione desiderata.
im tolleranza secondi	comando Imposta Movimento , richiede: <ul style="list-style-type: none">• <i>tolleranza</i>: tolleranza ammessa in passi encoder per iniziare il prossimo movimento• <i>secondi</i>: attesa per uscire dal ciclo di attesa di raggiungimento della posizione (per evitare loop infiniti)
ga	comando Gomito Alto , imposta che la posizione richiesta con il comando ' mc ' venga raggiunta dall'effettore con il gomito del robot rivolto verso l'alto.
gb	comando Gomito Basso , imposta che la posizione richiesta con il comando ' mc ' venga raggiunta dall'effettore con il gomito del robot rivolto verso il basso.
kp K_p	comando Kp , imposta il valore della costante proporzionale K_p della legge di controllo del robot al valore desiderato.
ms real_time_clock	comando Millisecondi , imposta il periodo con il quale viene invocata la legge di controllo del robot al valore desiderato.
go	comando Gripper Open , apre la pinza del robot.
gc	comando Gripper Close , chiude la pinza del robot.
hm	comando Home , porta il robot nella posizione di Home.
rh	Comando Set Home , considero che la Home non e' stata effettuata e che i giunti siano in posizioni non corrette.
sh	comando Reset Home , considero che la Home e' stata effettuata e che i giunti siano nelle posizioni corrette.
rs	comando Reset , azzerà il conteggio dei passi encoder.
cm	Comando Manuale comando diretto del robot mediante tastiera. E' possibile modificare passi encoder, angoli di giunto, posizione e orientamento di effettore.
mn	Come cm

to x_A y_A z_A x_B y_B z_B ε_2

comando **Track Originale**, prende:

- x_A y_A z_A coordinate del punto iniziale A
- x_B y_B z_B coordinate del punto finale, nel sistema di riferimento di base
- ε_2 angolo di orientamento del link di effettore rispetto alla normale al piano di base

Si posiziona sul punto A e traccia quindi la traiettoria rettilinea da A a B.

te x_B y_B z_B

comando **Track Evoluzione**, richiede x_B y_B z_B coordinate del punto finale, nel sistema di riferimento di base. Traccia quindi la traiettoria rettilinea dal punto in cui si trova attualmente l'effettore al punto B.

tp x_A y_A z_A

comando **Track Partenza**, richiede le coordinate del punto A iniziale e posiziona l'effettore ad una distanza h (impostata con il comando 'is') dal punto A, va usata prima del comando 'sc' mentre non e' necessaria per 'tr' e 'tc' che la richiamano automaticamente. Le coordinate di A vanno espresse nel sistema di riferimento del piano.

tP x_A y_A z_A

comando **Track Partenza Evoluzione**, chiede le coordinate del punto A iniziale e posiziona l'effettore in A. Le coordinate di A vanno espresse nel sistema di riferimento del piano.

tr x_A y_A z_A x_B y_B z_B

comando **Track Retta**, richiede x_A y_A z_A coordinate del punto iniziale A, e x_B y_B z_B coordinate del punto finale, nel sistema di riferimento del piano (Oxyz)_P impostato con il comando 'ip'. Traccia quindi la traiettoria rettilinea richiesta da A a B. Non necessita di usare prima 'tp' perché si posiziona prima in A poi esegue la traiettoria rettilinea.

tc x_A y_A z_A x_C y_C z_C γ_{ASS} senso

comando **Track Circonferenza**, richiede:

- x_A y_A z_A coordinate del punto iniziale circonferenza, nel sistema di riferimento (Oxyz)_P del piano impostato con il comando 'ip'.
- x_C y_C z_C coordinate del centro della circonferenza, nel sistema di riferimento del piano (Oxyz)_P impostato con il comando 'ip'.
- γ_{ASS} l'angolo assoluto da raggiungere, il senso di rotazione
- *senso*: senso di rotazione (+1 antiorario, -1 orario)

Traccia quindi la traiettoria circolare richiesta. Non necessita di usare prima 'tp' perché si posiziona prima in A poi esegue la traiettoria circolare.

tC x_C y_C z_C γ_{REL}

comando **Track Circonferenza Evoluzione**, richiede:

- x_C y_C z_C coordinate del centro della circonferenza, nel sistema di riferimento del piano (Oxyz)_P impostato con il comando 'ip'.
- γ_{REL} angolo relativo da coprire dalla posizione iniziale attorno a C (il senso di rotazione è direttamente specificato dal segno dell'angolo, antiorario se è positivo orario se è negativo)

Traccia quindi la traiettoria circolare richiesta. Necessita di usare prima 'tP' per posizionarsi sul punto iniziale.

ip α β Dx

comando **Imposta Piano**, richiede i due angoli di inclinazione α e β e l'intercetta Dx , con l'asse x_0 , del piano sul quale verranno tracciate le traiettorie rettilinee e circolari con i comandi '**tr**', '**tc**', '**tC**'.

is font h passo millisecondi

comando **Imposta Scrittura**, richiede:

- *font*: grandezza dei caratteri
- *h*: altezza dal piano di lavoro durante il posizionamento
- *passo*: distanza lineare, calcolata lungo la traiettoria, tra due punti successivi
- *millisecondi*: attesa tra l'assegnazione di due punti successivi di una traiettoria

gl font

comando **Grandezza Lettere**, richiede *font*, grandezza dei caratteri

iu lunghezza_utensile

comando **Imposta Utensile**, richiede la distanza alla quale spostare la punta dell'effettore dalla posizione iniziale, indispensabile se si usa uno strumento.

sc 'stringa'

comando **Scrivi**, il robot scrive la frase '*stringa*' sul piano impostato. Va prima impostato il piano di scrittura con '**ip**' e posizionato l'effettore nel punto di partenza della scrittura con '**tp**'.

ef <file>

comando **Esegui File**, interpreta successivamente ogni riga del file di testo <*FILE*> come se fosse un comando digitato dal prompt.

12. Controllo Manuale

Per consentire all'utente un'interazione diretta col robot è stata implementata nel programma una funzione di controllo in tempo reale del movimento.

Avviata tale funzione si può gestire direttamente tramite tastiera il cambiamento delle grandezze variabili per il sistema di movimentazione: coordinate dell'effettore, angoli di Eulero, angoli di giunto e posizione dei motori.

La variazione delle grandezze controllabili può essere effettuata attraverso la pressione dei tasti di SHIFT: lo SHIFT di destra aumenta il valore della grandezza mentre lo SHIFT di sinistra lo diminuisce.

Per cambiare la grandezza da modificare con gli SHIFT è utilizzato un sistema di scelta intuitivo, che identifica la grandezza corrente come combinazione di due selezioni effettuabili tramite la pressione di certi tasti.

Di seguito è riportato uno schema dei tipi di grandezza associati ai tasti di scelta:

Tasto	Tipo
"m"	posizione dei motori (dei link per BECCHEGGIO e ROLLIO)
"g"	angoli di giunto
"c"	coordinate dell'effettore
"e"	angoli di Eulero

Tasto	Grandezza
"1" o "x"	Tipo->1 oppure coordinata X dell'effettore
"2" o "y"	Tipo->2 oppure coordinata Y dell'effettore
"3" o "z"	Tipo->3 oppure coordinata Z dell'effettore
"4"	Tipo->4 (non per gli angoli di Eulero)
"5"	Tipo->5 (non per gli angoli di Eulero)

Ad esempio se si preme il tasto "c" seguito dal tasto "3" la pressione di uno SHIFT servirà a modificare il valore della coordinata Z dell'effettore.

Come negli altri casi in cui l'utente richieda un movimento del robot, prima di inviare segnali elettrici al robot viene eseguita la funzione di controllo per la posizione futura (a meno che non sia stata mai eseguita la funzione di home dall'avvio del programma).

Nella finestra di dialogo è stampata una guida per l'utilizzo dei caratteri speciali utilizzabili durante il controllo manuale.

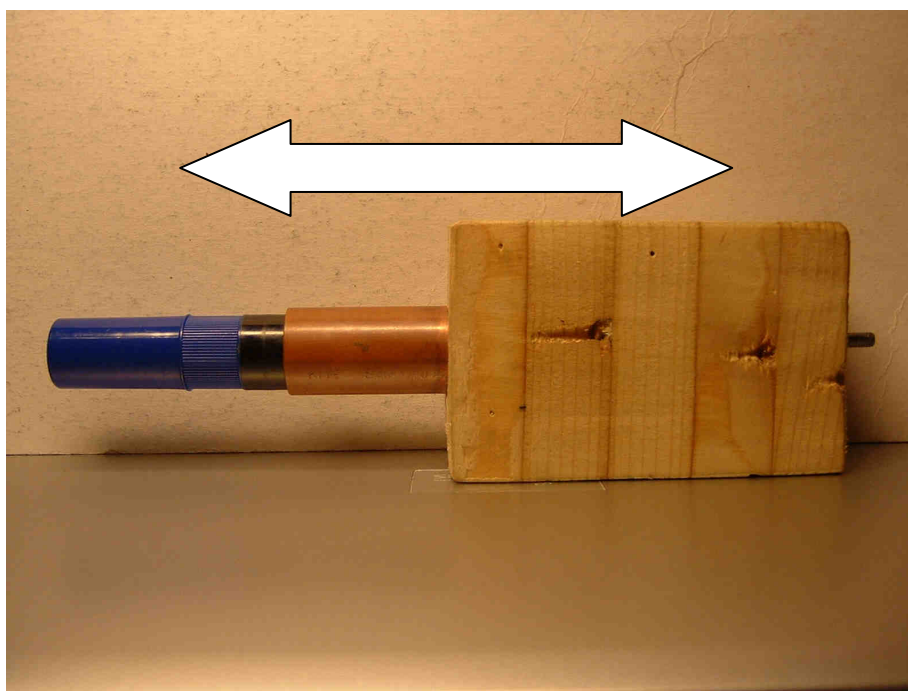
Per uscire dal Controllo Manuale è necessario digitare il tasto ESC.

13. Preparazione per la dimostrazione

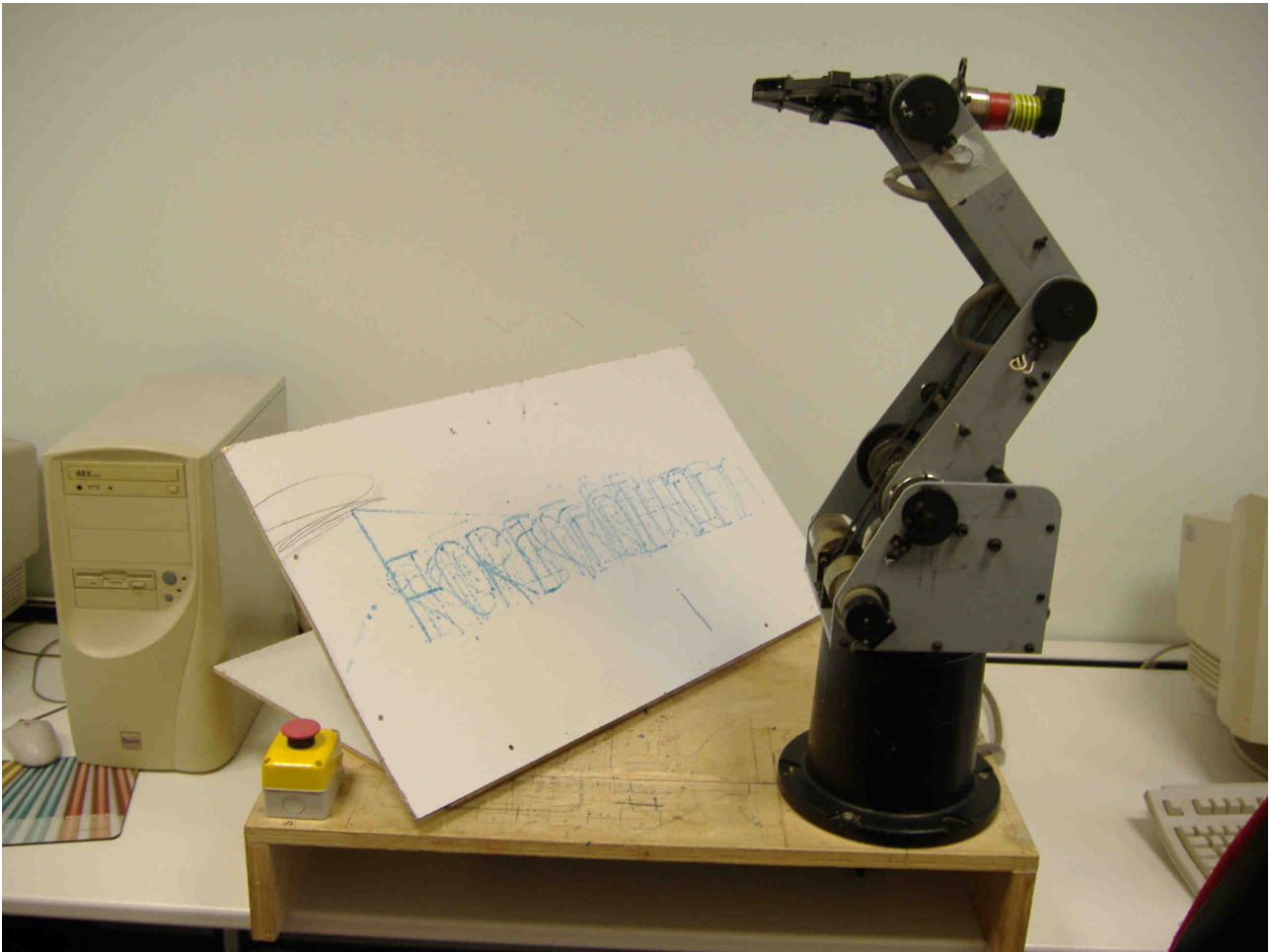
La dimostrazione finale è servita come strumento per verificare tutto il lavoro compiuto, ed in particolare la funzionalità di inseguimento di traiettorie rettilinee o circolari lungo qualsiasi piano impostato dall'utente. Tramite quindi l'utilizzo di queste funzioni, sono state implementate le lettere dell'alfabeto, per permettere allo Scorbot ER-V di scrivere qualsiasi frase, con caratteri di qualsiasi dimensione.

Si è poi dovuto fronteggiare dei problemi di ordine pratico, primo fra tutti la scelta dello strumento con cui far scrivere il robot, ed il relativo supporto. Dopo varie prove, e scartando varie ipotesi (lavagna con gessetto, pastelli a cera su carta, pennello e tempere su tela...), la scelta è ricaduta su un pennarello a punta larga.

Purtroppo, il gioco meccanico ed i vari errori di misura, portavano spesso ad impercettibili variazioni che non permettevano al pennarello di tracciare una linea uniforme e continua. È stato subito chiaro che serviva un sistema di ammortizzamento sul pennarello, risolto con una molla ed un tubo di rame. Il tubo di rame è esattamente del diametro del pennarello, e costituisce quindi la guida su cui esso scorre senza sbandamenti laterali. All'estremità opposta della punta del pennarello è poi collegata una molla che permette alla punta di rimanere sempre attaccata al foglio. Il tutto viene poi completato da un parallelepipedo di legno che permette una presa facile e sicura al robot, come vediamo nella figura sottostante.



Il robot può scrivere su qualunque piano impostabile dell'utente. Quindi, per sottolineare questo fatto, si è scelto di posizionare il foglio per la dimostrazione su un piano inclinato a 45°, realizzato appositamente con delle tavole di legno.



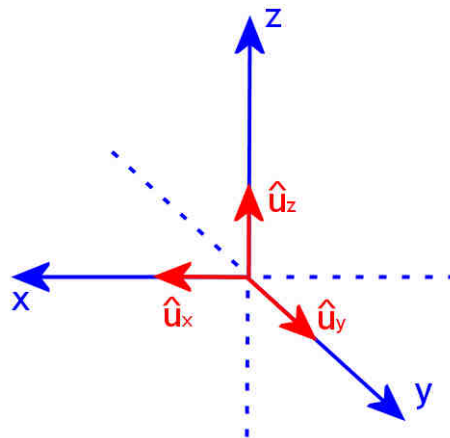
14. Appendice

14.1 Sistemi coordinati tridimensionali destrorsi

Un sistema di assi cartesiani tridimensionale destrorso è una terna di tre vettori \hat{u}_x , \hat{u}_y , \hat{u}_z (che definiscono con direzione e verso gli assi coordinati x, y, z) che soddisfa le seguenti proprietà:

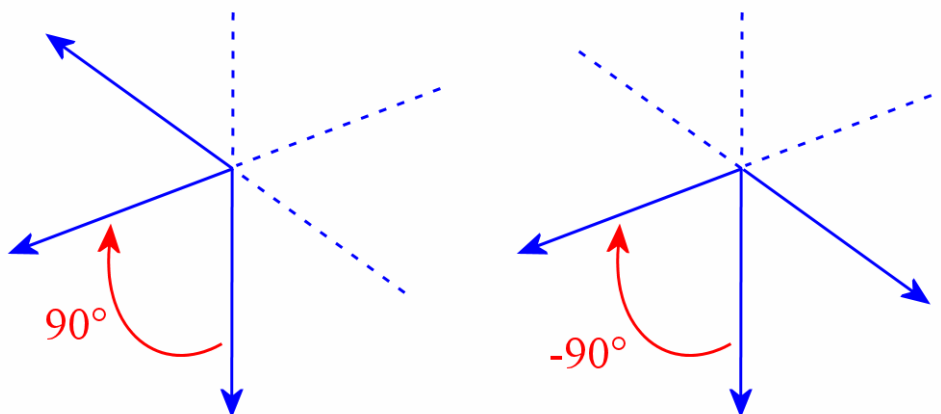
- sono mutuamente ortogonali tra loro e quindi il loro prodotto scalare è nullo ($\langle \hat{u}_x, \hat{u}_y \rangle = \langle \hat{u}_x, \hat{u}_z \rangle = \langle \hat{u}_y, \hat{u}_z \rangle = 0$)
- sono di lunghezza unitaria e quindi la loro norma è unitaria ($|\hat{u}_x| = |\hat{u}_y| = |\hat{u}_z| = 1$) definendo per questo l'unità di misura degli assi associati x, y, z
- il prodotto scalare dei primi due vettori \hat{u}_x , \hat{u}_y restituisce il terzo versore \hat{u}_z ($\hat{u}_x \times \hat{u}_y = \hat{u}_z$)

I sistemi coordinati definiti nella presente relazione sono tutti di tipo destrorso.



14.2 Convenzione sulla misura degli angoli nello spazio

Nello spazio tridimensionale gli angoli tra due assi si calcolano positivi in senso antiorario avendo di fronte la freccia del restante asse, e negativi in senso orario sempre avendo di fronte la freccia del restante asse.

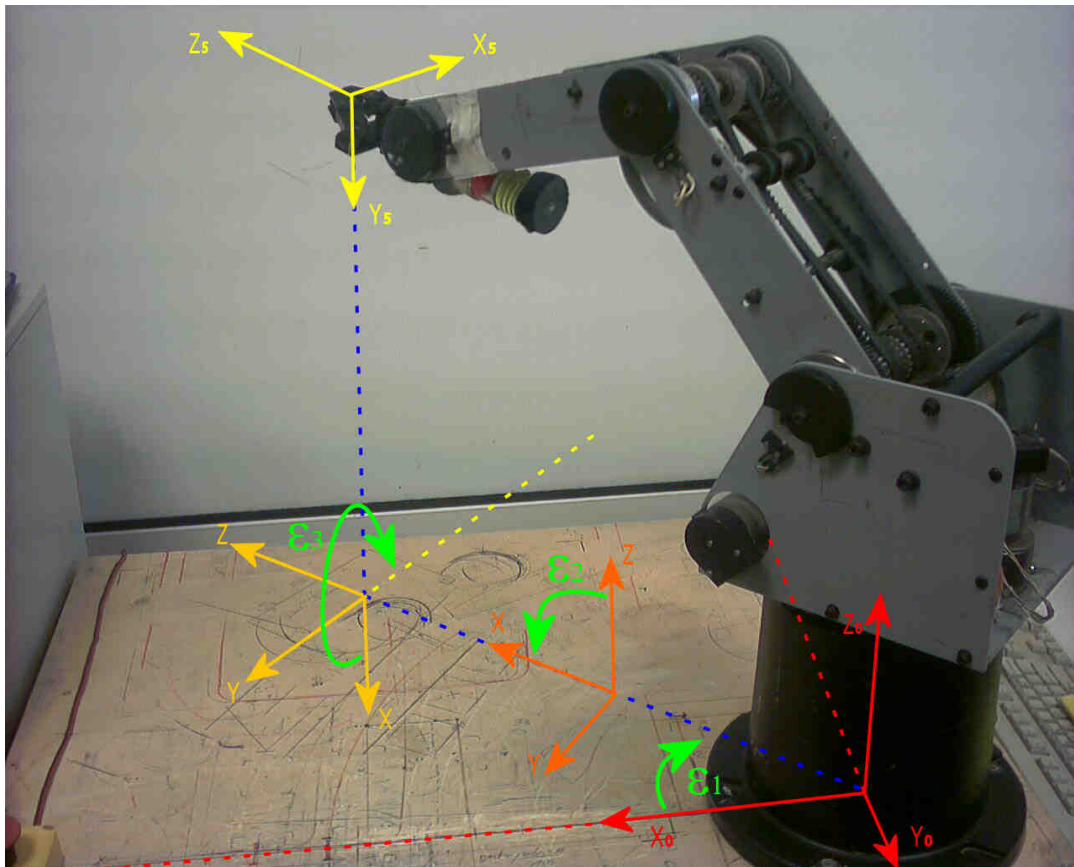


14.3 Angoli di Eulero

Per descrivere l'orientamento di un sistema di riferimento cartesiano tridimensionale rispetto ad un altro si può ricorrere all'utilizzo di una terna di angoli che rappresentino tre rotazioni intorno a tre assi coordinati. Si cerca cioè quella terna minima di rotazioni che permetta al primo sistema di riferimento di assumere lo stesso orientamento del secondo sistema. Gli angoli associati alle tre rotazioni sono detti angoli di Eulero. Ogni rotazione successiva va effettuata attorno ad un asse diverso dal precedente, affinché rappresenti una rotazione indipendente rispetto a quella che la precede.

Una possibile scelta della terna di angoli di Eulero è quella di scegliere tre angoli ε_1 , ε_2 , ε_3 che rappresentino le seguenti rotazioni sufficienti a far assumere al sistema di riferimento $(Oxyz)_0$ in figura quello del sistema $(Oxyz)_5$:

- rotazione di un angolo ε_1 del sistema $(Oxyz)_0$ intorno all'asse z_0 per posizionare l'asse x_0 sullo stesso piano dell'asse z_5
- rotazione di un angolo ε_2 del sistema $(Oxyz)_0$ intorno al proprio asse y_0 corrente (a seguito cioè della prima rotazione) per allineare l'asse x_0 sulla stessa direzione dell'asse z_5
- rotazione di un angolo ε_3 del sistema $(Oxyz)_0$ intorno al proprio asse z_0 per allineare gli assi x_0 e y_0 rispettivamente agli assi x_5 e y_5 e far quindi raggiungere a $(Oxyz)_0$ lo stesso orientamento di $(Oxyz)_5$



Questa che è una delle possibili scelte degli angoli Eulero viene detta convenzione ZYZ poiché la scelta delle successive rotazioni ricade sugli assi coordinati z , y , e z .

14.4 Coordinate Omogenee

Le trasformazioni di coordinate tra due sistemi di riferimento tridimensionali $(Oxyz)_0$ e $(Oxyz)_1$ che hanno lo stesso orientamento (gli assi coordinati rispettivi sono paralleli) possono essere descritte nel modo seguente:

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

dove

$$\vec{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

è il vettore che applicato nell'origine O_0 del sistema di riferimento $(Oxyz)_0$ punta sull'origine O_1 del sistema di riferimento $(Oxyz)_1$, ossia il vettore O_0O_1 .

Lo spostamento rigido che fa sovrapporre il sistema di riferimento $(Oxyz)_0$ a $(Oxyz)_1$ si dice traslazione.

Le trasformazioni di coordinate tra due sistemi di riferimento tridimensionali $(Oxyz)_1$ e $(Oxyz)_2$ che hanno le origini coincidenti possono essere descritte nel modo seguente:

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = R \cdot \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}$$

dove

$$\hat{x}_2 = \begin{bmatrix} r_{1,1} \\ r_{2,1} \\ r_{3,1} \end{bmatrix}, \quad \hat{y}_2 = \begin{bmatrix} r_{1,2} \\ r_{2,2} \\ r_{3,2} \end{bmatrix}, \quad \hat{z}_2 = \begin{bmatrix} r_{1,3} \\ r_{2,3} \\ r_{3,3} \end{bmatrix}$$

sono le coordinate rispetto al sistema di riferimento $(Oxyz)_1$ dei versori del sistema $(Oxyz)_2$.

Lo spostamento rigido che fa sovrapporre il sistema di riferimento $(Oxyz)_1$ a $(Oxyz)_2$ si dice rotazione.

Lo spostamento rigido che fa sovrapporre un sistema di riferimento tridimensionale $(Oxyz)_0$ ad un sistema $(Oxyz)_2$ che ha diverso stesso orientamento del primo e origine non coincidente con il primo è sempre ottenibile come combinazione di uno spostamento rotazionale e di uno traslatorio.

Le trasformazioni di coordinate tra i due sistemi di riferimento sono:

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

le quali posso essere espresse in maniera più compatta:

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & v_x \\ r_{2,1} & r_{2,2} & r_{2,3} & v_y \\ r_{3,1} & r_{3,2} & r_{3,3} & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = {}^0T_1 \cdot \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix}$$

In questo modo aggiungendo alle coordinate cartesiane una componente ulteriore sempre costante e pari ad 1 è possibile descrivere mediante una sola trasformazione algebrica una composizione di uno spostamento rotatorio e di uno rotatorio. Le nuove coordinate qui definite vengono dette coordinate omogenee e la trasformazione espressa dall'equazione di sopra viene detta trasformazione di coordinate omogenee dal sistema $(Oxyz)_0$ a $(Oxyz)_2$.

Ad esempio le coordinate omogenee dell'origine O_2 nel sistema di riferimento $(Oxyz)_0$, avendo O_2 in $(Oxyz)_2$ coordinate omogenee $[0, 0, 0, 1]'$, sono:

$$\begin{bmatrix} x_{0 \ O_2} \\ y_{0 \ O_2} \\ z_{0 \ O_2} \\ 1 \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & v_x \\ r_{2,1} & r_{2,2} & r_{2,3} & v_y \\ r_{3,1} & r_{3,2} & r_{3,3} & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{2 \ O_2} \\ y_{2 \ O_2} \\ z_{2 \ O_2} \\ 1 \end{bmatrix} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & v_x \\ r_{2,1} & r_{2,2} & r_{2,3} & v_y \\ r_{3,1} & r_{3,2} & r_{3,3} & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \\ 1 \end{bmatrix}$$

Particolarmente utile per la caratterizzazione delle trasformazioni di coordinate omogenee tra i sistemi di riferimento definiti per la descrizione della cinematica del robot risulta la seguente trasformazione di coordinate omogenee, rappresentativa della sequenza di spostamenti elencata:

1. Traslazione di una lunghezza d lungo l'asse z
2. Rotazione di un angolo θ attorno all'asse z corrente
3. Traslazione di una lunghezza a lungo l'asse x corrente
4. Rotazione di un angolo α attorno all'asse x corrente

L'equazione esplicita che lega le coordinate del sistema $(Oxyz)_1$ a quelle di $(Oxyz)_2$, questo ultimo ottenuto a seguito dei quattro spostamenti elencati sopra è:

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\cos(\alpha) \cdot \sin(\theta) & \sin(\alpha) \cdot \sin(\theta) & a \cdot \cos(\theta) \\ \sin(\theta) & \cos(\alpha) \cdot \cos(\theta) & -\sin(\alpha) \cdot \cos(\theta) & a \cdot \sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix}$$

Tale trasformazione viene utilizzata per descrivere la trasformazione di coordinate omogenee sei sistemi di riferimento definiti sul robot in accordo con la convenzione di Denavit – Hartenberg.

14.5 Convenzione di Denavit – Hartenberg

E' una convenzione che descrive una regola per definire un sistema di riferimento cartesiano associato ad ogni link di un manipolatore meccanico. Una volta definiti questi sistemi di riferimento è possibile, mediante le trasformazioni di coordinate omogenee, risalire alla posizione cartesiana di ogni singolo link rispetto al primo sistema di riferimento (fisso) al variare della configurazione assunta dal robot (a seguito cioè di un movimento dei giunti).

Regole di tracciamento dei sistemi di riferimento:

- $(Oxyz)_0$ Sistema di riferimento di base (fisso e solidale con il piano di lavoro)
 - scegliere l'asse z_0 come l'asse del primo giunto
 - scegliere il centro O_0 arbitrariamente su z_0 (ad esempio al livello del piano d'appoggio)
 - scegliere l'asse x_0 arbitrariamente
- $(Oxyz)_i$ con $1 \leq i < N$ (N numero di link) Sistema di riferimento del link i (solidale con il link i)
 - scegliere l'asse z_i come l'asse del giunto $i+1$
 - scegliere il centro O_i come segue:
 - se gli assi z_{i-1} e z_i sono sghembi sull'intersezione della normale comune a z_{i-1} e z_i con z_i
 - se gli assi z_{i-1} e z_i sono paralleli arbitrariamente
 - se gli assi z_{i-1} e z_i sono secanti sulla loro intersezione
 - scegliere il centro x_i come segue:
 - se gli assi z_{i-1} e z_i sono sghembi o paralleli, come il prolungamento della normale comune a z_{i-1} e z_i
 - se gli assi z_{i-1} e z_i sono secanti, perpendicolare al piano individuato da z_{i-1} e z_i
- $(Oxyz)_N$ Sistema di riferimento di effettore, solidale all'ultimo link
 - scegliere z_N nella direzione di avvicinamento
 - scegliere O_N al centro della pinza o sul punto terminale dell'effettore
 - scegliere x_N nella direzione di apertura della pinza

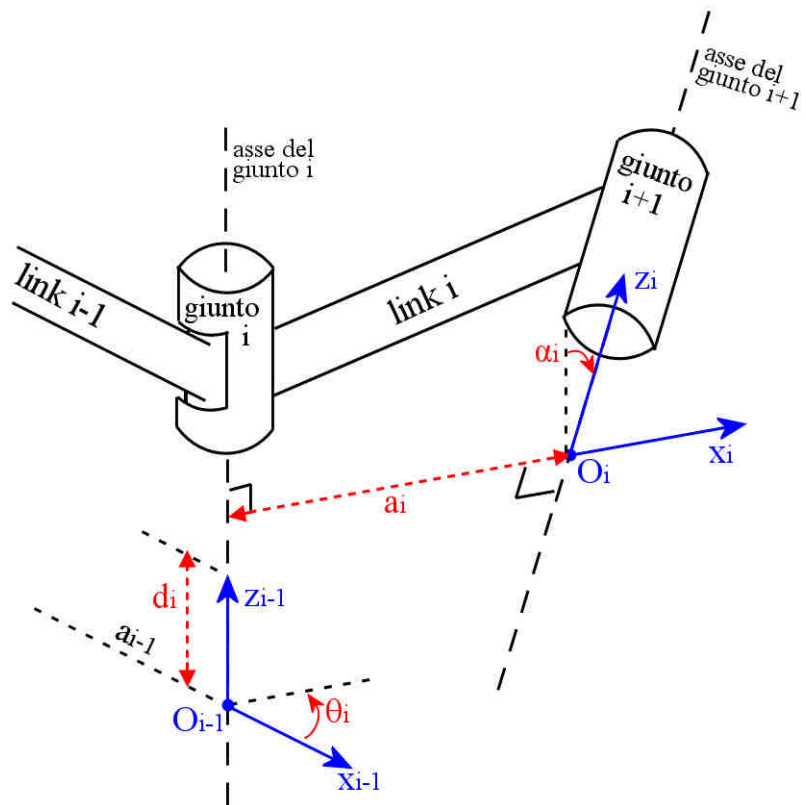
Una volta definiti gli $N+1$ sistemi di riferimento di D-H si possono definire per ogni coppia giunto - link quattro parametri:

- (a_i, α_i) Parametri di Link che dipendono dalla struttura del link
- (d_i, θ_i) Parametri di Giunto che dipendono dalla posizione relativa dei link adiacenti

I Parametri di Link e di Giunto sono:

- a_i è la distanza tra gli assi z_{i-1} e z_i .
- α_i è l'angolo tra gli assi z_{i-1} e z_i .
- d_i è la distanza misurata su z_{i-1} tra x_{i-1} e x_i .
- θ_i è l'angolo tra x_{i-1} e x_i .

Per un giunto prismatico dei quattro parametri solo d_i sarà variabile, mentre per un giunto rotazionale solo θ_i .



15. Magic Moments

Il lavoro in laboratorio è stato davvero impegnativo, costruttivo, educativo, come ben testimoniato da queste foto che ritraggono i componenti del gruppo nell'estenuante attività di sviluppo.

