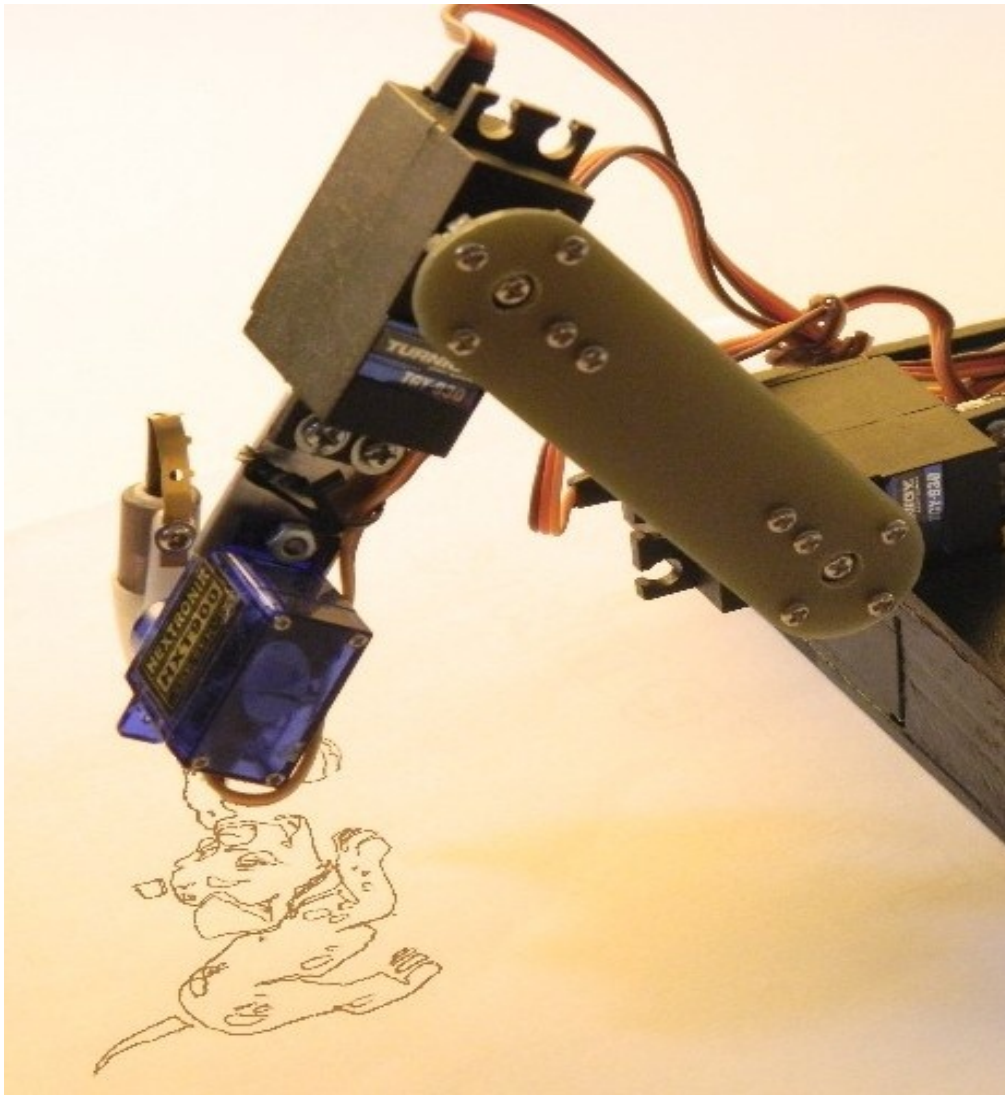


Robot Arm

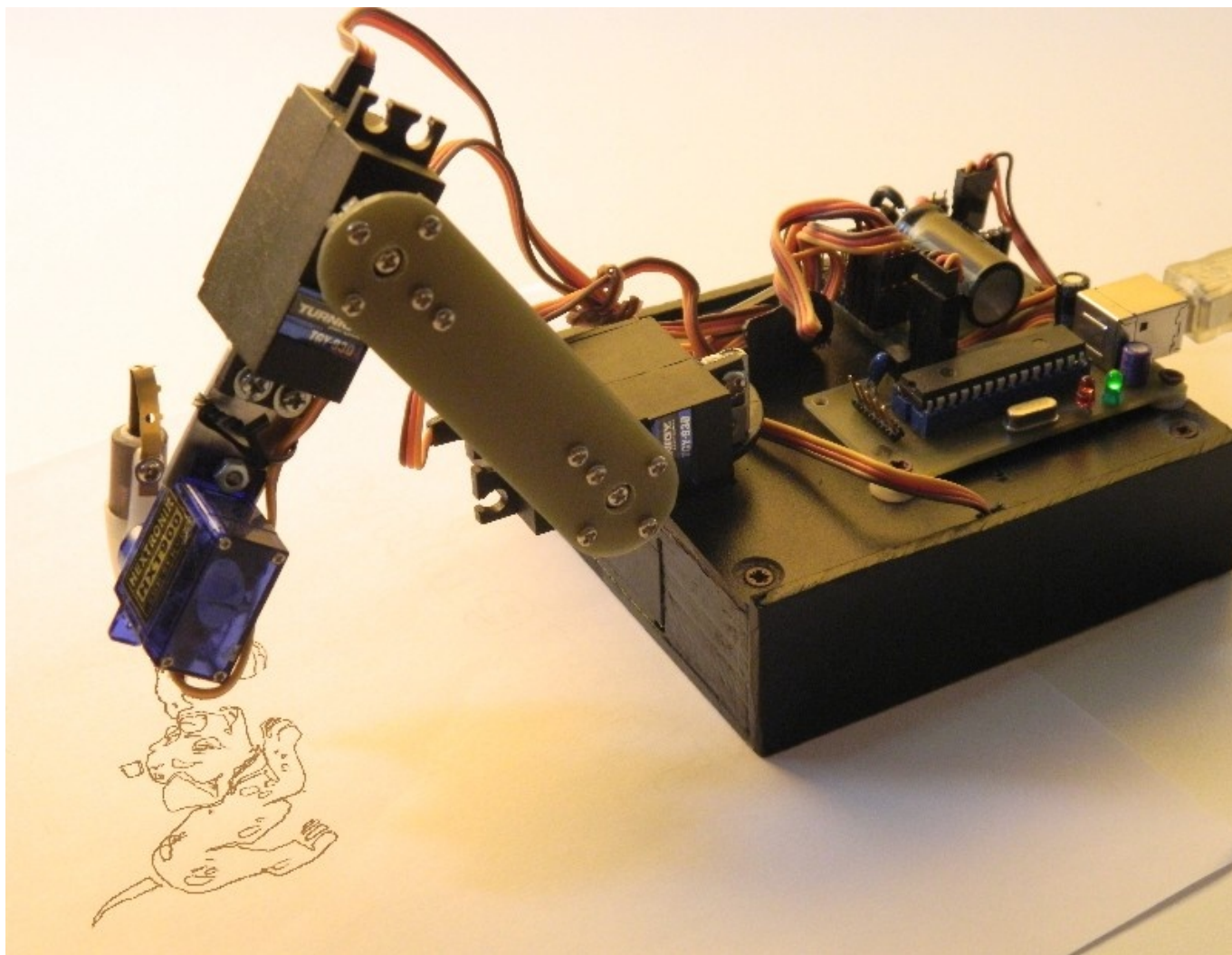
Un braccio robotico economico con meccanica essenziale



Perché un altro braccio robotico ?

Esistono molti bracci robotici in commercio: perché proporne un altro ancora? Perché questo progetto mostra un approccio innovativo che riduce la meccanica quasi a zero e permette a chiunque di costruirsi un piccolo braccio robotico in casa, spendendo pochissimo. Inoltre, e sembra incredibile, si ottiene una precisione maggiore di quella di prodotti commerciali da 200/500 Euro, abbastanza per scrivere, disegnare e fare piccole lavorazioni.

Vista di insieme



Una vista di insieme del RobotArm che ha appena eseguito un piccolo disegno.

Video del RobotArm in azione

I prossimi quattro video mostrano la buona precisione nella esecuzione di un piccolo disegno e la **eccezionale ripetibilità** che viene dimostrata ripassando il disegno con errori intorno al decimo di millimetro, praticamente invisibili anche con un forte ingrandimento. In alcuni video la traccia audio è stata amplificata per evidenziare i rumori di ambiente e dimostrare che la velocità di spostamento è quella reale.

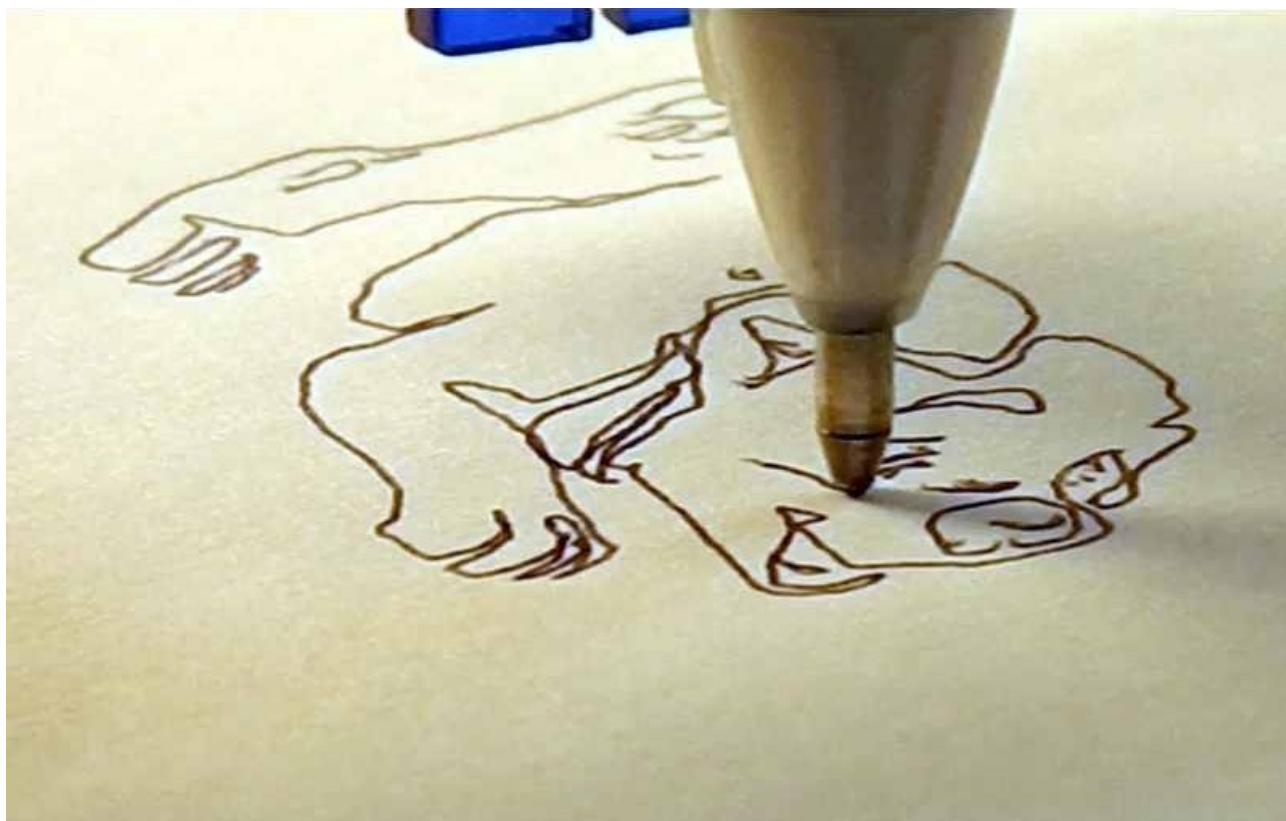
[RobotArm_Precision_1.wmv](#) Un video che mostra la buona precisione nel disegno.

[RobotArm_Precision_2.wmv](#) Un video, ingrandito, sulla precisione nel disegno.

[RobotArm_RepeatabilityTest1.wmv](#) Un video sulla ripetibilità di posizionamento.

[RobotArm_RepeatabilityTest2.wmv](#) Secondo video sulla ripetibilità di posizionamento.

[RobotArmMilling.wmv](#) - Il RobotArm che esegue una piccola lavorazione con la fresa.



Caratteristiche del RobotArm versione LowCost - 9 euro

Questa versione è composta da quattro servocomandi HXT900, tre piccole piastrine di alluminio e una base in legno.

Costo della meccanica: 9 euro (1)
Precisione: 1.5 mm (2)
Ripetibilità: 0.8 mm (2)
Velocità dei "rapidi" 500 mm/sec

- (1) 4 servo da 2 euro più un euro di alluminio e legno
- (2) Su un campo di azione di 100 mm x 50 mm x 40 mm

Caratteristiche del RobotArm versione DeLuxe - 33 euro

Questa versione è composta da tre servocomandi digitali TGY930, un servocomando HXT900, tre piccole piastrine di alluminio e una base in legno.

Costo della meccanica: 33 euro (1)
Precisione: 0.4 mm (2)
Ripetibilità: 0.1 mm (2)
Velocità dei "rapidi" 900 mm/sec

- (1) 3 servo da 10 euro, uno da 2 euro più un euro di alluminio e legno
- (2) Su un campo di azione di 160 mm x 80 mm x 50 mm

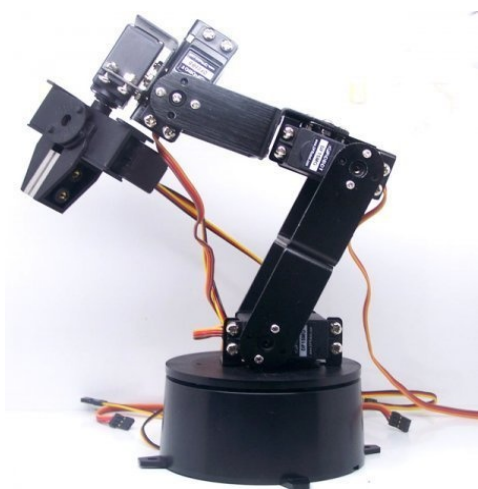
Confronto con i bracci robotici commerciali

Esistono molti bracci robotici in commercio, dai più semplici che si possono comprare, o costruire, a partire da 200 Euro, fino ai robot industriali, che riescono a spostare centinaia di chili con velocità, precisione e prezzi impressionanti.

Qui di seguito un buon esempio di piccolo braccio robotico, ben studiato, che negli USA costa 199 Dollari ma che in Italia costa circa **280 Euro (solo meccanica e motori - in kit di montaggio)**, a cui occorre aggiungere l'elettronica (es. Arduino da 25 Euro):

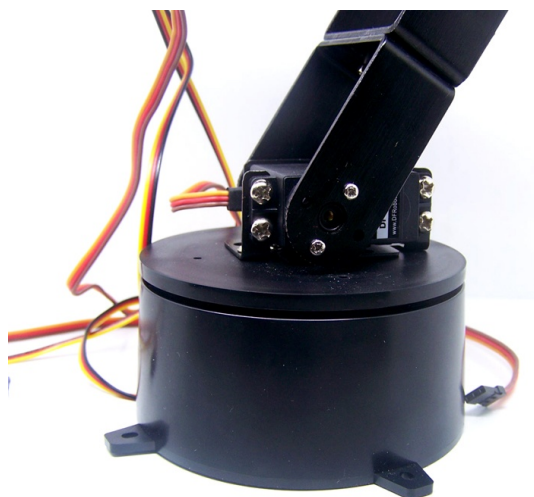
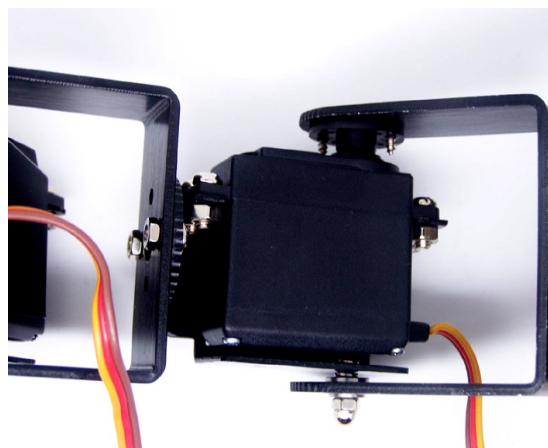
[ArduinoGripper-6-dof-programmable-clamp-robot-arm](#)

E qui si può vederlo al lavoro: [Video YouTube 9w6tKcZzhYo](#)



Questo braccio robotico, costruito con molta cura per aumentare la stabilità meccanica, dispone anche di una base con cuscinetto reggispira ed è un buon esempio di quello che si trova attualmente in commercio nella fascia dai 200 ai 500 Euro.

Il costruttore non dichiara le caratteristiche di precisione e ripetibilità, ma si può vedere dai filmati che questi valori sono intorno ai **5 mm**, sufficienti per prendere e spostare piccoli oggetti, ma senz'altro **troppi per scrivere o disegnare**.



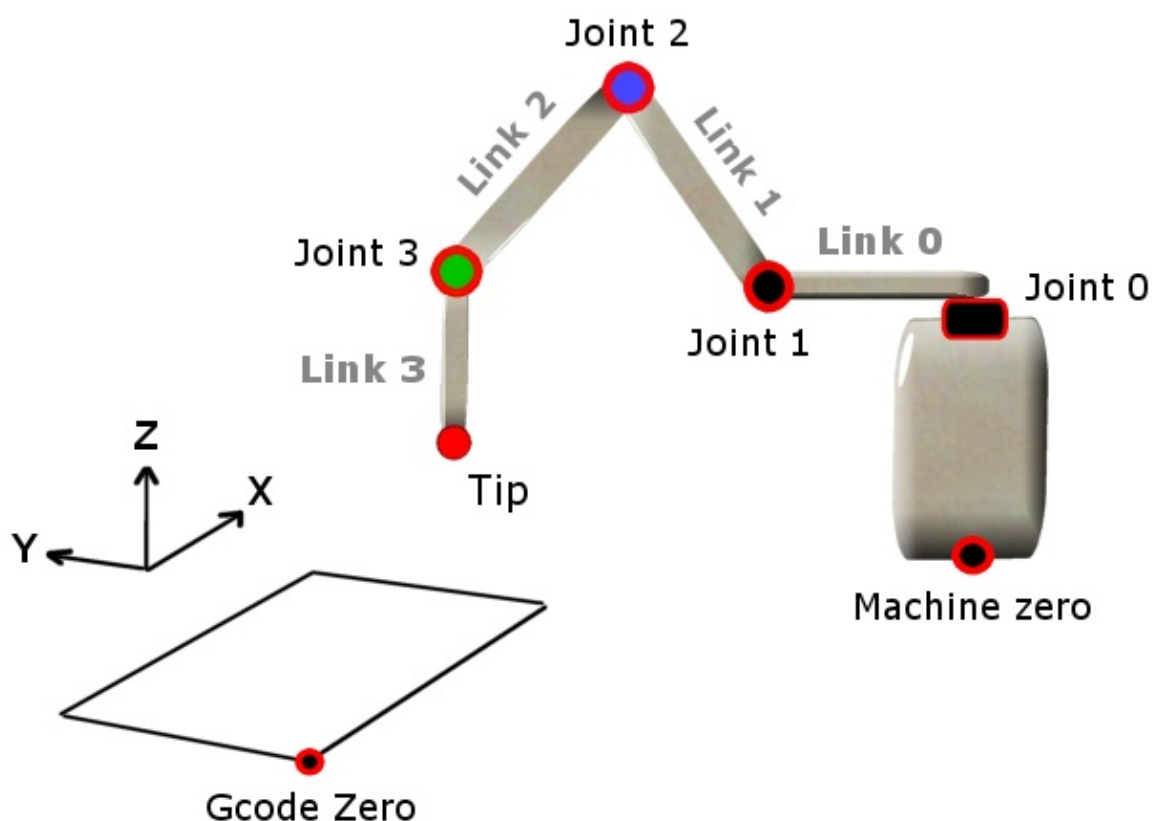
Particolari delle staffe, con il secondo perno di rotazione dal lato opposto rispetto all'asse del servocomando e della base rotante dotata di cuscinetti reggispira.



Qui iniziano le istruzioni per la costruzione del RobotArm

I nomi usati nella robotica

Ecco lo schema di base di un braccio robotico "articolato" a 4 giunti, "Joint", "Link" e "Tip" sono i nomi comunemente usati nella robotica.

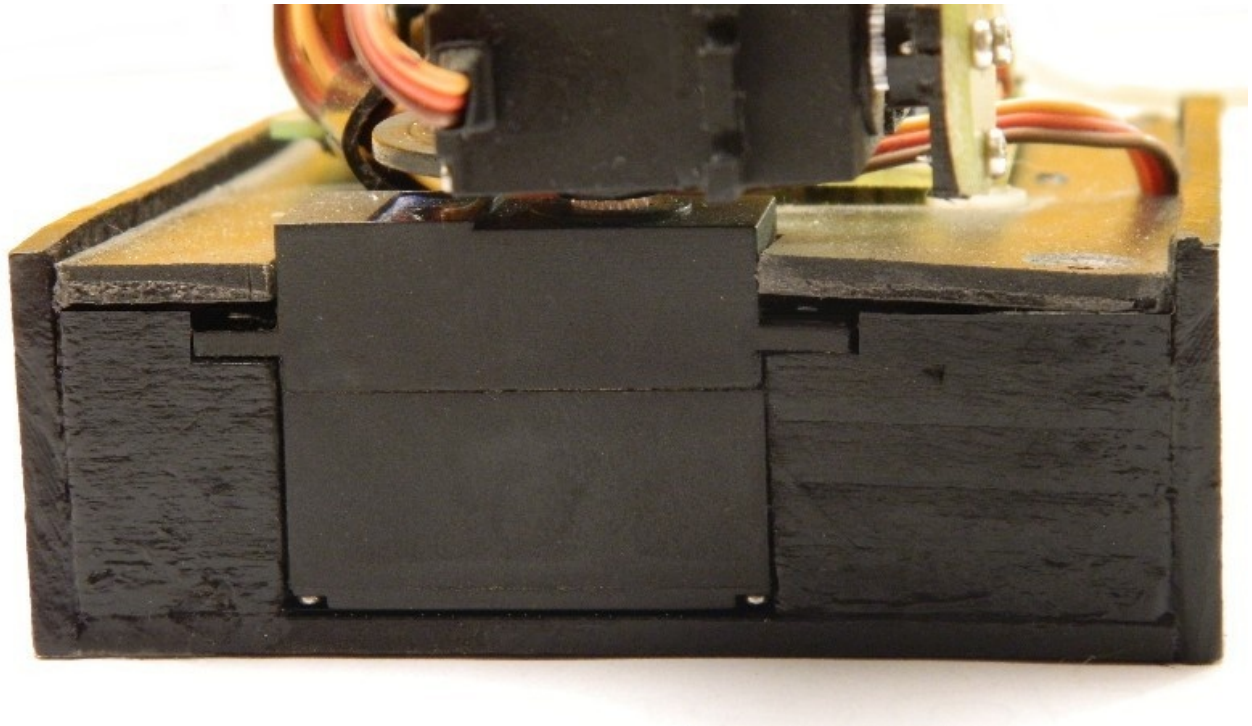


I **"Joint"** sono gli assi di rotazione che, nel nostro caso, coincidono con i perni dei servocomandi.

I **"Link"** sono i bracci rigidi che collegano i servocomandi tra di loro.

Il **"Tip"** è la punta che, nel nostro caso, viene mantenuta sempre verticale dal Joint 3.

Particolari meccanici - il Servocomando di base

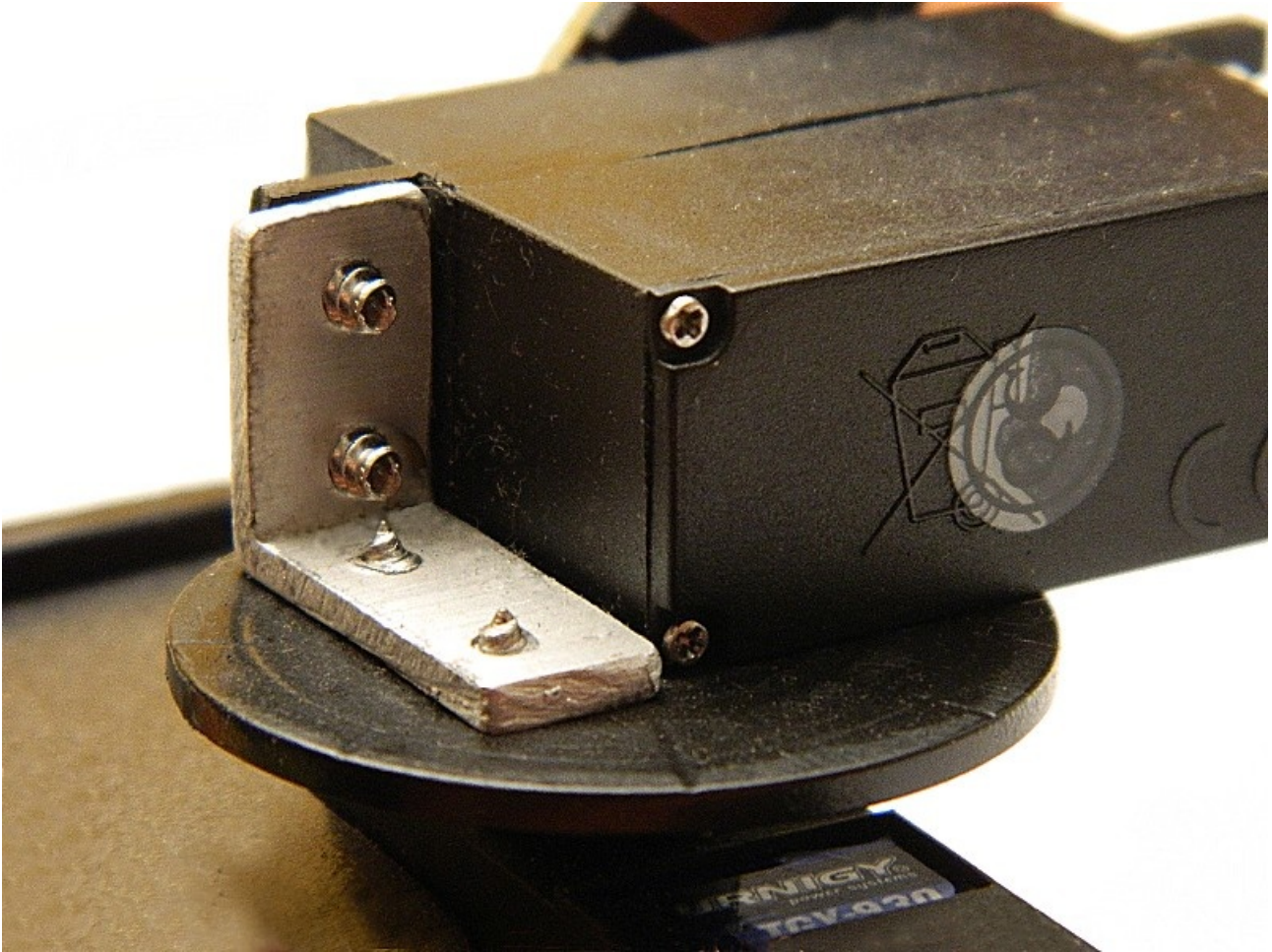


Il primo servocomando è avvitato nella base di legno, cercate di posizionarlo con l'asse ben verticale. Sulla destra si vede il suo cavetto che entra attraverso una fessura praticata nel coperchio.

La base è una scatola di legno riempita di sferette di acciaio e pesa abbastanza da tenere ben fermo il foglio (al posto delle sfere di acciaio si potrebbero usare piccole pietre, sabbia, ghiaia, pezzi di piombo, piastre di ferro, viti o dadi).

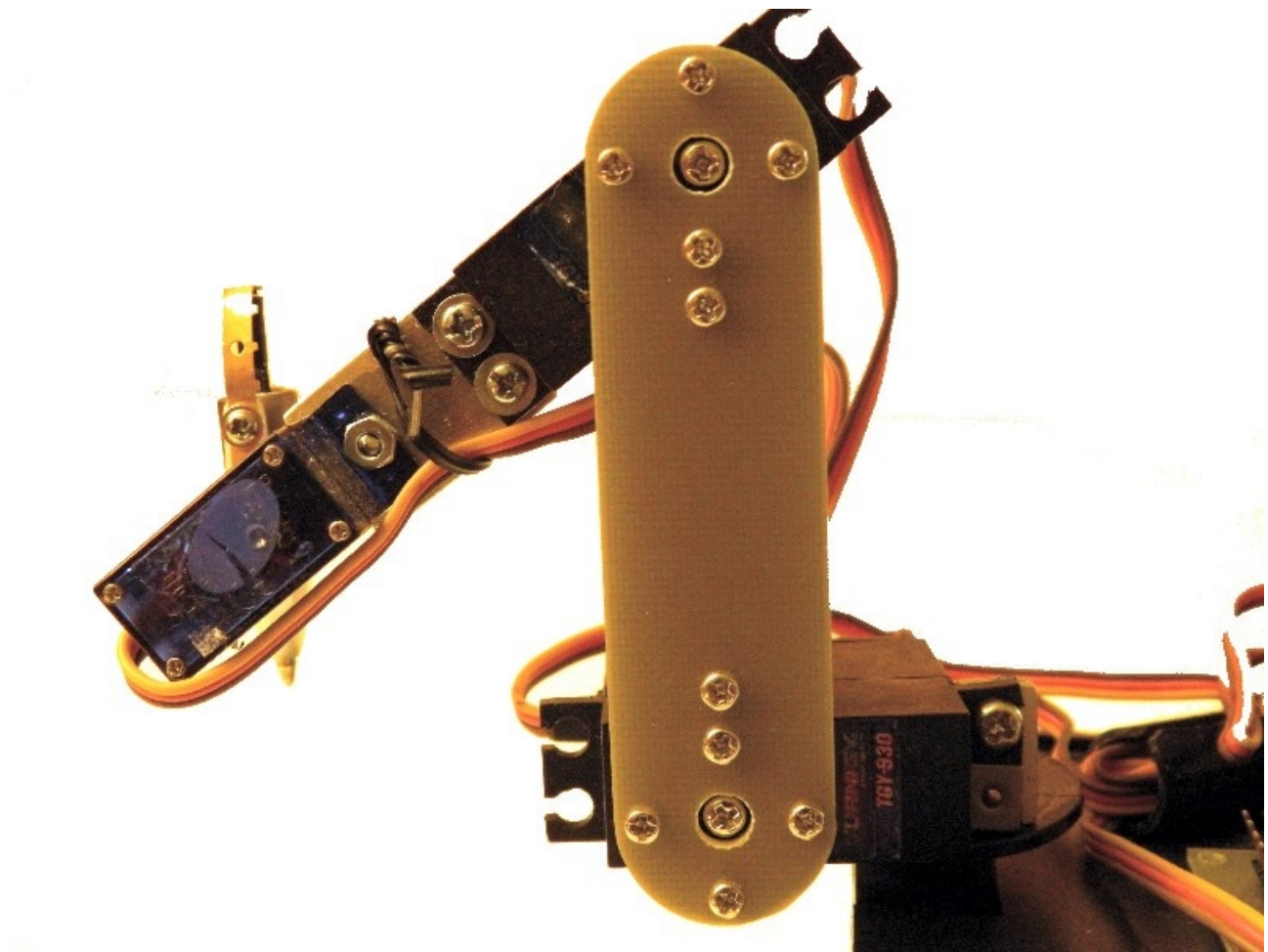
Sotto alla scatola, nei quattro angoli, si incollano dei gommini morbidi che danno maggiore stabilità alla base e aiutano a fare buona presa sulla carta. I gommini devono essere sottili perché posizionando il servocomando di base più in basso possibile si ottiene una maggiore precisione.

Particolari meccanici - il "Link o"



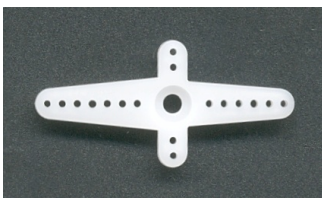
Il "Link o" è costituito da una piccola "L" in alluminio. La base nera rotonda è una delle staffe sempre in dotazione con i servocomandi. Anche le quattro viti parker sono spesso nel sacchetto del servocomando. Consiglio però, quando si ordinano i servocomandi, di ordinare da HobbyKing anche qualche sacchetto da 100 viti parker nei formati più piccoli (1.5 e 2 mm) Sono viti difficili da trovare altrove, costano poco e saranno utili in molti casi.

Particolari meccanici - il "Link 1"

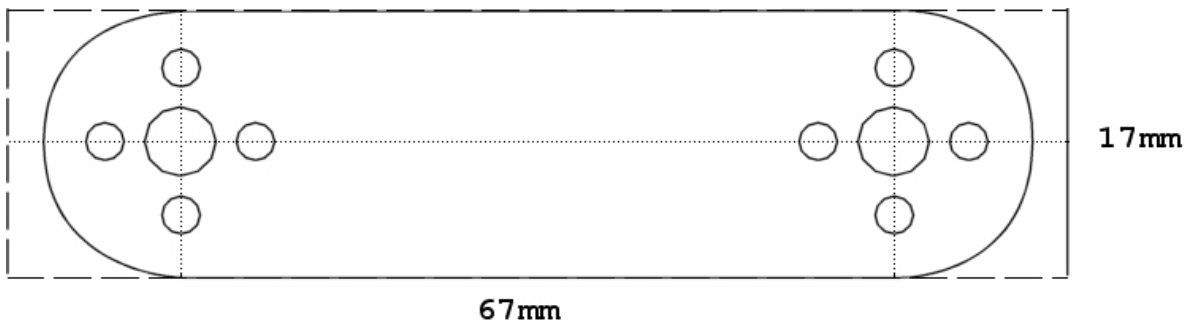


Il "Link 1" è il pezzo meccanico più *'complesso'*, si fa per dire, di tutto il RobotArm.

Il "Link 1" viene avvitato con quattro piccole viti parker alle crocette, che sono sempre fornite con i servocomandi. Uno dei bracci lunghi della crocetta in nylon deve essere tagliato con le tronchesine per non farlo sporgere. La crocetta viene poi avvitata al servocomando attraverso il foro centrale.



Per forare con precisione potrebbe essere utile stampare una "dima" su carta, come la seguente, e poi sovrapporla al pezzo da forare attaccandola con scotch trasparente.



Servo: HXT900
Link: 1

Scale: 100 pixel/cm

Central to central: 45mm
Central hole: 4.5mm
Lateral holes: 2.5mm
Lateral holes dist: 6.7mm
Lateral dist. diagonal: 9.5mm

Questa è la versione per i servo HXT900. A seconda dei servo usati, i fori dovranno essere spostati leggermente, usate come riferimento le piastrine in plastica che sono sempre allegate ai servo.

Le mie prime piastrine erano in alluminio poi ho scoperto che è ancora più facile farle con la vetronite ramata: ve la consiglio.

Come scegliere la lunghezza dei Link 1 e 2

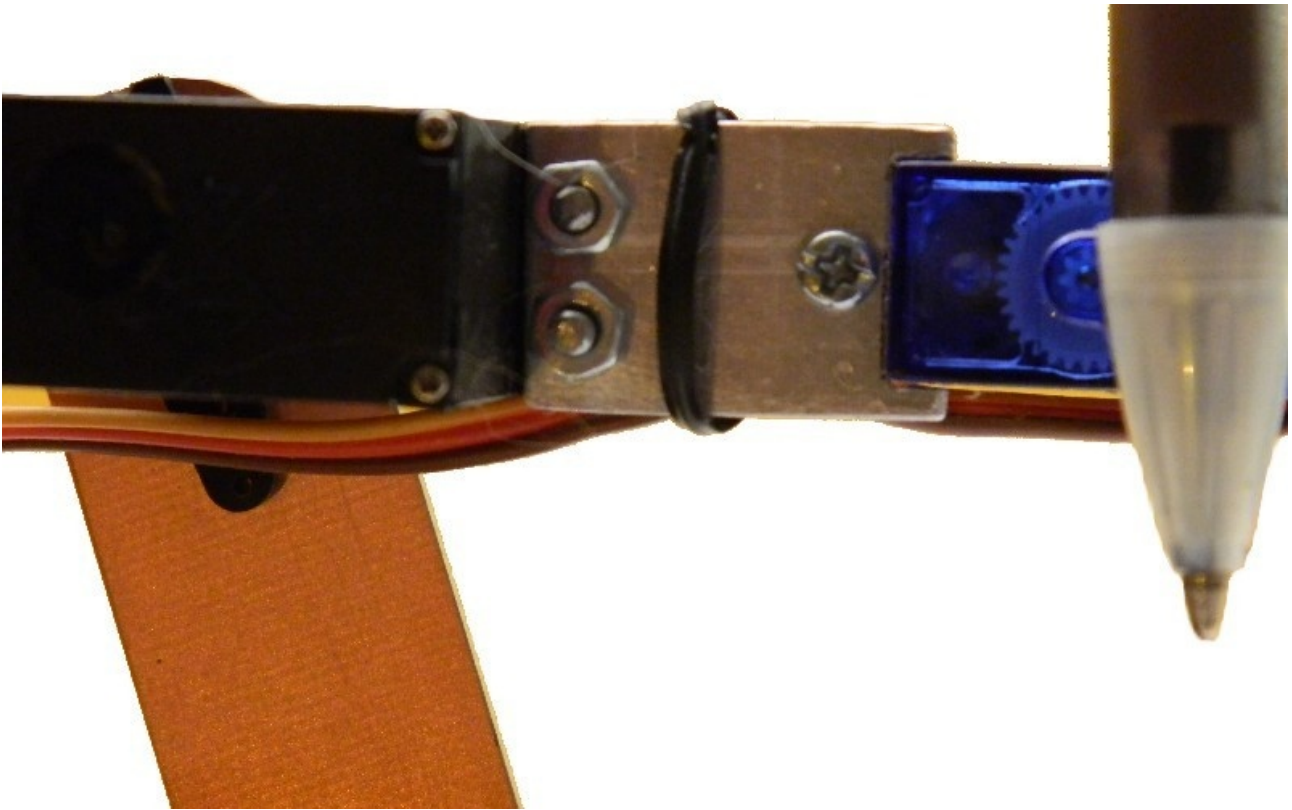
La lunghezza del "Link 1" può essere aumentata per avere più raggio di azione (ma una precisione minore); oppure può essere ridotta per avere minor raggio di azione e precisione maggiore.

Quando si decide la lunghezza del "Link 1" è bene fare in modo che corrisponda a quella del "Link 2" così durante i "rapidi" (servo alla massima velocità) la punta non tenderà a salire o a scendere.

Dato che la lunghezza del "Link 2" non può diminuire sotto a un certo valore (devono starci i servo "2" e "3" uno dopo l'altro), sono consigliabili le misure seguenti: 60 mm per i servo TGY930 e 45 mm per i più piccoli HXT900. Queste sono le misure minime e consentono la massima precisione, se si accetta di perdere precisione si possono allungare i Link 1 e 2 a piacere e coprire un raggio di azione anche molto grande (ricordarsi di mantenere i due link sempre uguali tra di loro)

La lunghezza dei "Link" si misura sempre tra i centri dei perni di rotazione.

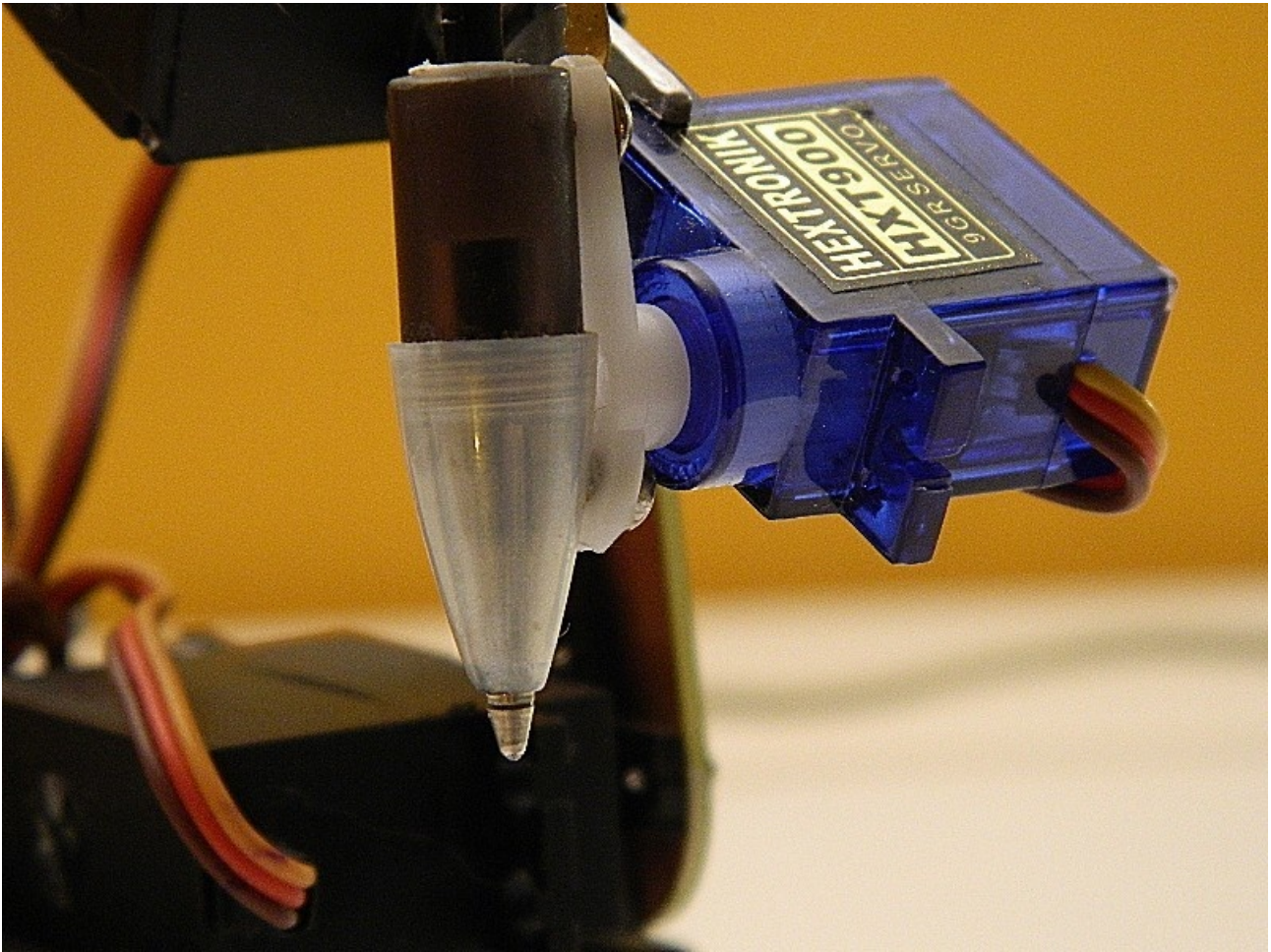
Particolari meccanici - il "Link 2"



Il "Link 2" è costituito da una piccola piastrina rettangolare di alluminio con tre fori e da tre viti da 2 mm con dado (se si usano i servo piccoli i fori e le viti sono solo due).

Nella parte destra si può notare un'asola, fatta con la lima, delle stesse dimensioni del servo HXT900 (azzurro nella foto), che serve per tenerlo ben fermo anche senza stringere molto la vite; ma è una finezza che si può anche evitare.

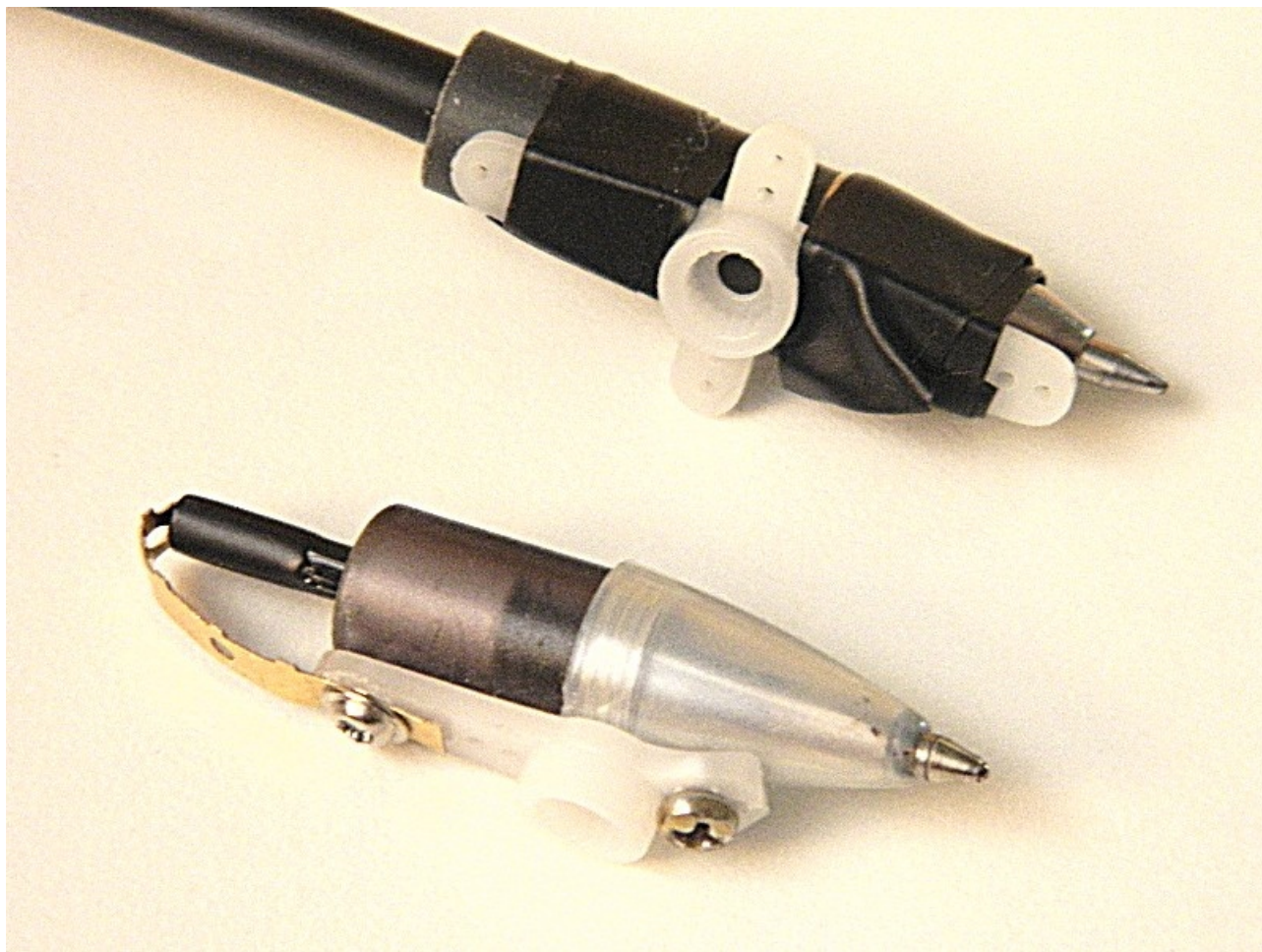
Particolari meccanici - il "Link 3"



Il "Link 3" è semplicemente il collegamento tra l'asse dell'ultimo servo e la penna.

Nel programma di controllo questo "Link" ha più parametri degli altri perché è necessario definire con precisione dove effettivamente si trova la punta, rispetto ai centri di rotazione, in tutte e tre le direzioni ortogonali X, Y e Z.

Il collegamento con la penna



Per realizzare il "pennino" si prende una biro molto morbida e la si attacca con due giri di nastro isolante alle normali crocette, che sono sempre in dotazione con i servo.

Se poi si alleggerisce la parte superiore e si aggiunge un elastico o una molla che spingono l'anima in basso si ottiene una scrittura molto migliore, meglio anche accorciare la penna per evitare che possa intralciare i movimenti.

Prima di trovare una soluzione valida, le penne di casa faranno tutte una brutta fine...

I pennarelli sono poco funzionali, sono pesanti, non si possono alleggerire, e si seccano velocemente.

Le penne a gel con tratto largo sono le migliori, ma sono difficili da alleggerire.



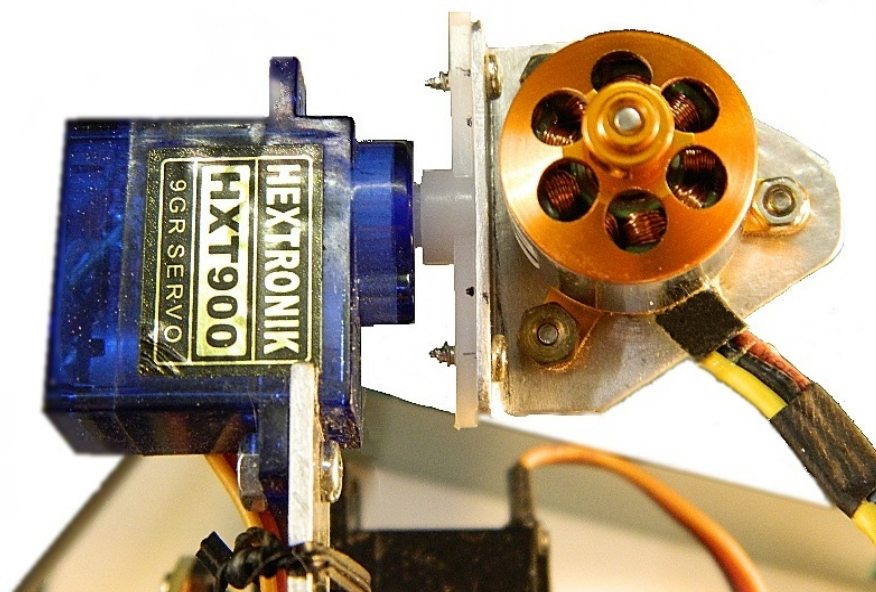
Possibili alternative alla penna

Scrivere con una penna è un compito abbastanza inutile, le stampanti sono più veloci e più precise. Ma un braccio robotico può fare anche molte altre cose, prendere piccoli oggetti per mezzo della depressione dell'aria, con un servocomando, una siringa e un lungo tubicino, o addirittura improvvisarsi MAKER ed effettuare piccole lavorazioni con un motore brushless e una fresa.

Guardate questo video! [RobotArmMilling.wmv](#)

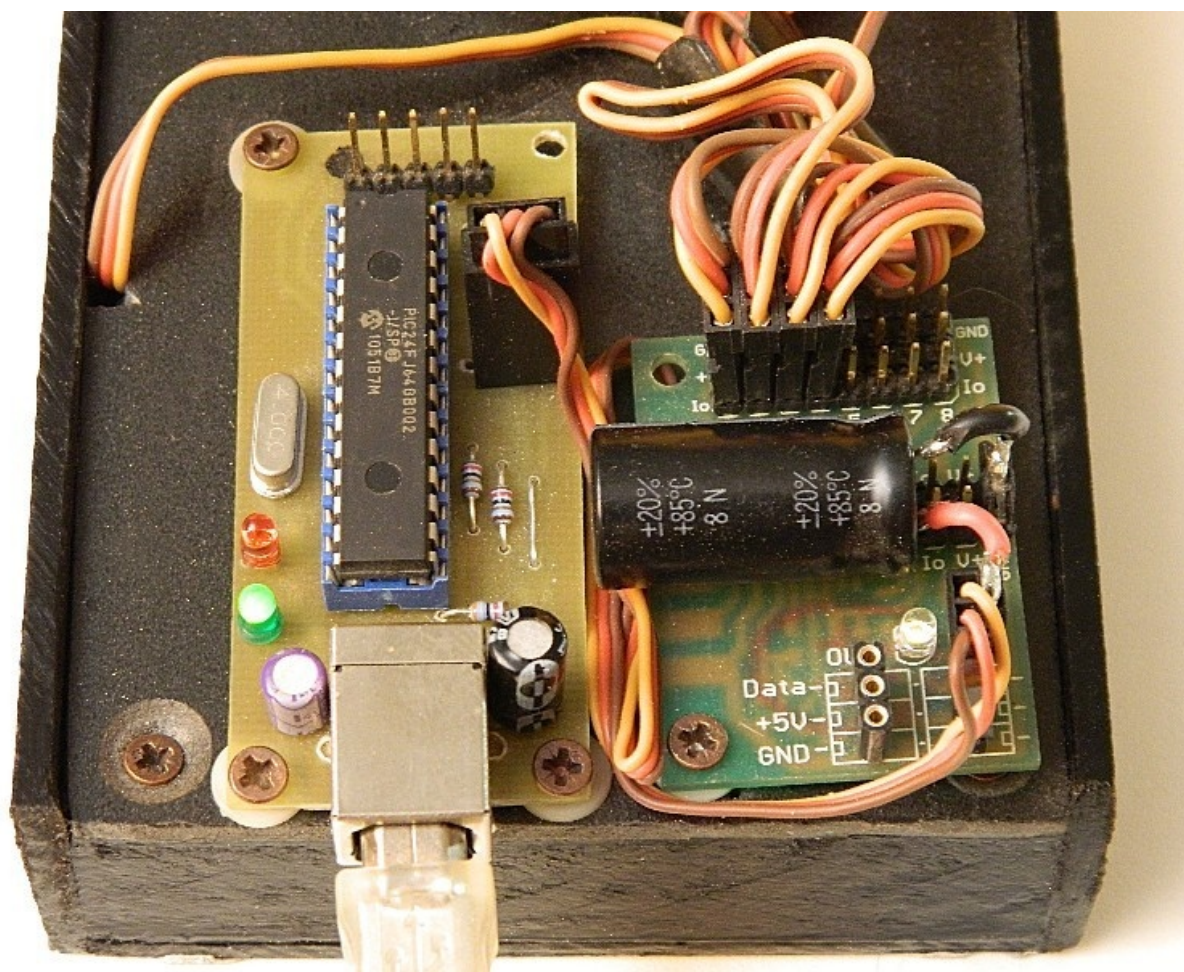


Con i **giunti in plastica e nylon** attuali la rigidità del RobotArm è scarsa per cui si possono lavorare solo materiali molto morbidi come la cera, il sapone e legni morbidi come la balsa. Inoltre si deve lavorare a bassa velocità e la precisione è scarsa. Però, dato che esistono servocomandi come il TGY930, con perni metallici dotati di due cuscinetti e rigidissimi, sostituendo i giunti con **equivalenti in ottone** si farebbe un salto di qualità. Per ottenere prestazioni ancora maggiori esistono servo ancora più potenti e robusti (attenzione che devono anche essere precisi e con una "dead-band" molto bassa).



Ecco un modo semplice per fissare un motore brushless. Il motore è un "Turnigy C1822-2100" (7 euro) e viene pilotato da un controller "Turnigy PLUSH-10A" (8 euro) che a sua volta viene regolato in velocità da 0 a 30000 giri da un segnale standard per servocomandi in arrivo da uno dei canali del modulo "Slave-Servo". La fresa è una "Proxxon 28 321" in carburo di tungsteno, da 0.8mm che costa circa 4 euro. Motore e controller sono reperibili presso www.hobbyking.com, la fresa presso qualunque buon ferramenta o "Brico". Per fissare la punta si prende un tondino in ferro o ottone e lo si fora da parte a parte con punta piccola, si allarga da un lato del diametro esatto della punta, e dall'altra del diametro del motore. Poi si controlla che i due perni entrino sforzando leggermente, si mettono due gocce di adesivo anaerobico (ad esempio Bostik anaerobico ferma-viti) e si piantano i due perni. La punta deve essere più corta possibile.

Elettronica di collegamento con il PC



Questa immagine mostra l'elettronica di controllo, che è molto semplice.

Il modulo a sinistra è il "Master" che interfaccia la porta USB del PC con la linea seriale che va al modulo "Slave" il quale, a sua volta, controlla i quattro servocomandi (potrebbe pilotarne fino a un massimo di 10).

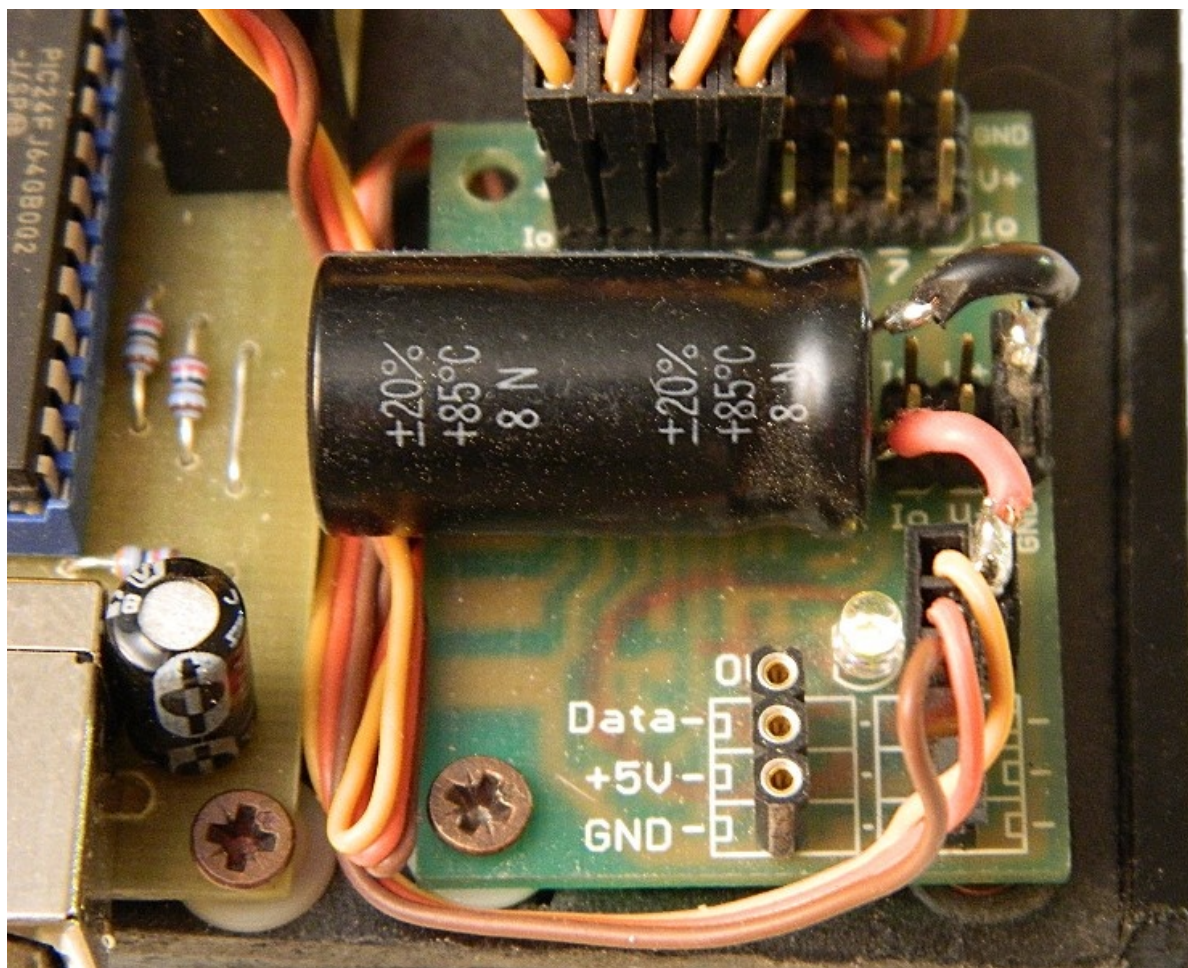
In basso si vede la presa USB, quando si collega il cavo al PC il RobotArm viene immediatamente riconosciuto, senza fastidiose installazioni di driver con tutti i problemi collegati.

Sul sito www.theremino.com nella sezione "Technical" si possono trovare gli schemi elettrici, i piani di montaggio e molte altre informazioni utili (tutto in Open Source)

Condensatore aggiuntivo

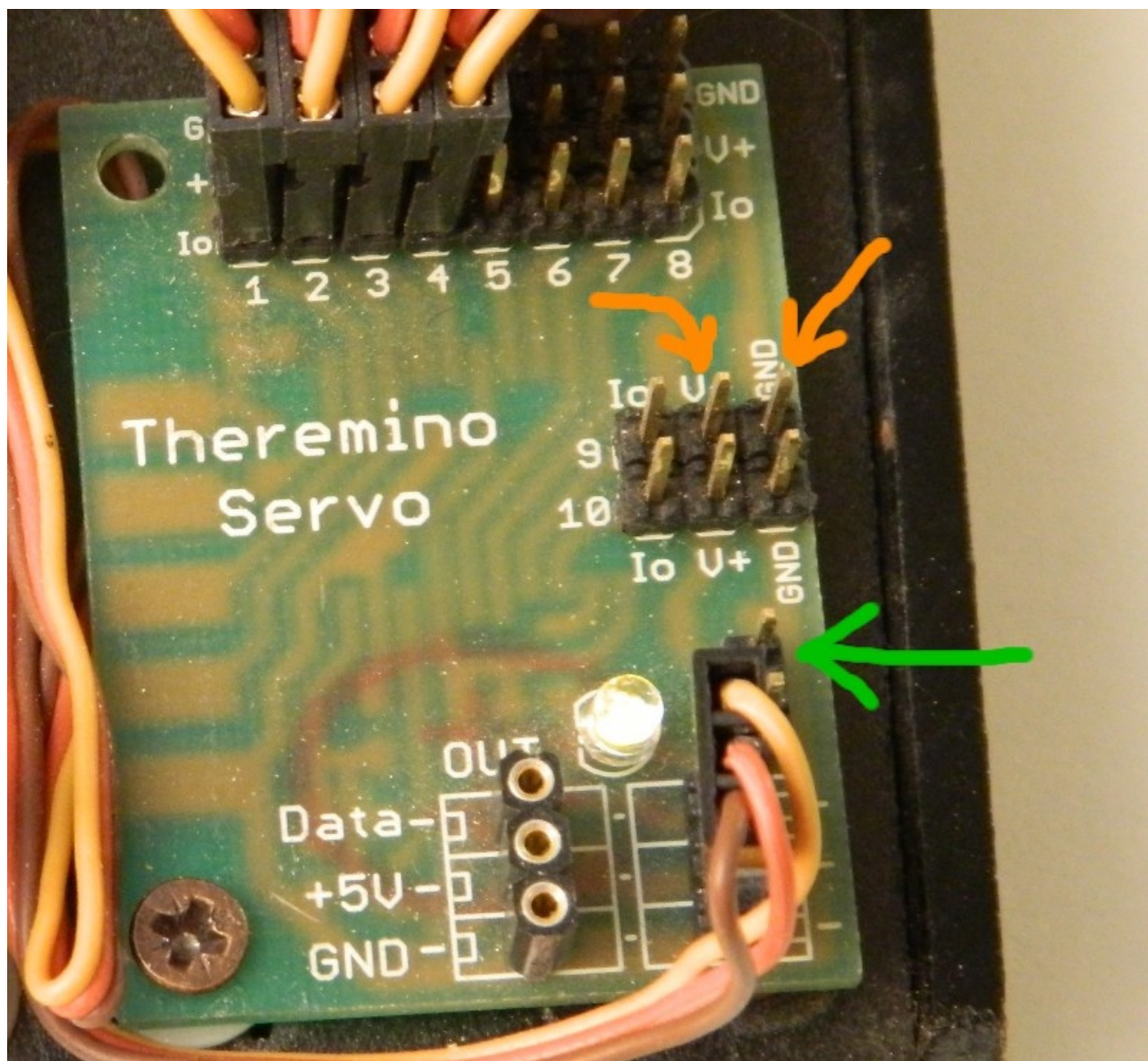
La massima corrente prelevabile dalla porta USB è di 500 mA. Poiché i servocomandi, soprattutto quelli digitali, assorbono molta corrente durante lo spunto, potrebbe accadere di mandare in protezione la porta USB o di avere cali temporanei di tensione in grado di produrre errori di trasmissione.

Quindi, per alimentare il RobotArm senza alimentatore esterno è bene aggiungere un condensatore elettrolitico da 2200 uF da almeno 6 Volt tra il +5 e GND.



In questa immagine si vede il condensatore collegato al modulo "Slave"

I servo da me adottati consumano poco, quasi tutti i servo digitali di classe superiore (più potenti, ma non necessariamente più precisi) consumano così tanto che anche il condensatore non è più sufficiente. Se si sperimentano frequenti errori di trasmissione o al limite completi blocchi della trasmissione quando si muovono i motori, questo è sicuramente dovuto al 5 Volt che cede e che, nei casi peggiori, porta addirittura al reset dei micro-controllori con conseguente perdita della comunicazione. In questi casi, per ripristinare la comunicazione, si deve scollegare l'USB, ricollegarla, e rifare il riconoscimento dei moduli. Se vi succede spesso c'è una sola cura, un alimentatore esterno da 5 Volt - 2 Ampere (oppure 5 Volt - 5 Ampere se state usando servo da 10 Kg/cm o più). Per collegare un alimentatore esterno si taglia il filo rosso della prolunga che collega il "Master" con lo "Slave" e si collega l'alimentatore al +5" e "GND" dello "Slave".



Questa immagine mostra i punti di collegamento del condensatore. Si perde la possibilità di usare gli InOut 9 e 10, ma gli 8 pin rimanenti bastano e avanzano anche per chi volesse aggiungere altri attuatori, come ad esempio una pinza o una pipetta a vuoto per piccoli oggetti.

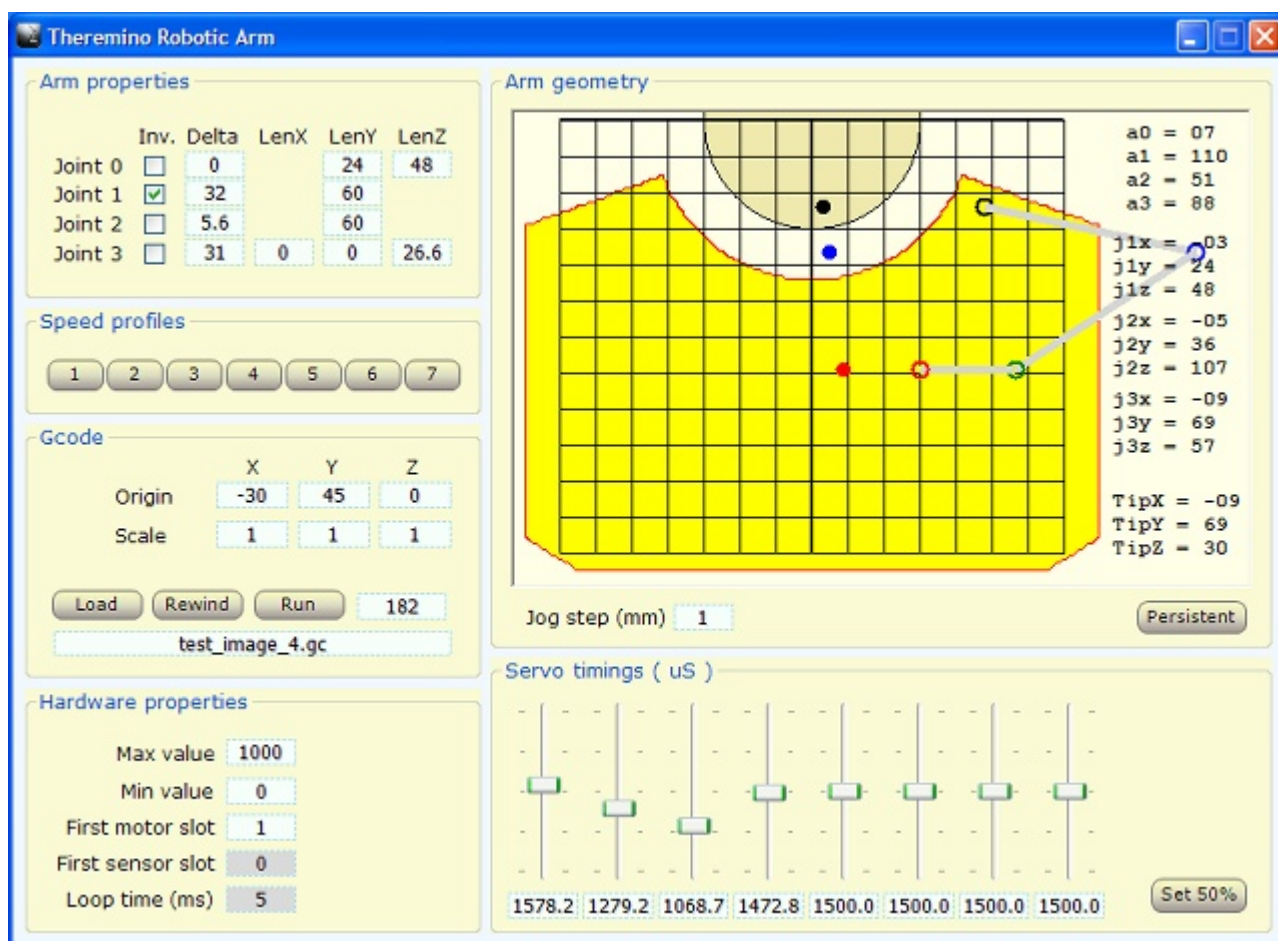
Nel mio caso, ho collegato il negativo ai due pin marcati GND e per il positivo ho fatto due fori e aggiunto due pin indicati dalla freccia verde in modo da permettere un collegamento più comodo, senza piegare troppo i fili.

Normalmente invece, il condensatore va inserito sui due pin "GND" e i due "V+" indicati dalle frecce di colore arancio e non è quindi necessario forare e saldare i due pin aggiuntivi.



Il software

L'applicazione "Theremino_Arm" (per Windows e Linux) è reperibile sul sito www.theremino.com nella pagina "Downloads/Uncategorized".



Questo software è Open Source con licenza "GNU General Public License Version 3", per cui, nella pagina di download è disponibile anche la versione completa di tutti i sorgenti. Un buon programmatore potrebbe aggiungere assi o cambiare il tipo di robot ad esempio in robot "scara" (ma dovrebbe modificare le formule di cinematica diretta e inversa che attualmente valgono solo per un "articolato" a 4 assi). Tenete conto anche che questa applicazione non pilota direttamente i moduli di InOut ma scrive solo negli "Slot" (Interprocess Communication) da lì in poi ci vuole l'HAL (Hardware Abstraction Layer) che si occupa dell'hardware (anch'esso Open Source).

Nel riquadro "Arm properties" si devono impostare le lunghezze esatte LenX, LenY e LenZ dei quattro Links, misurandole con un calibro tra i centri dei perni di rotazione. Nelle quattro caselle "Delta" si deve impostare un angolo di taratura (in gradi) e qui ci vuole davvero molta esperienza.

Due brevi suggerimenti: fate in modo che il braccio reale abbia gli stessi angoli di quello del programma (che è rappresentato con linee grigie e pallini nero, blu, verde e rosso e visto lateralmente), ma nel contempo fate attenzione che i cursori non vadano a fondo scala in alto o in basso riducendo così l'area di lavoro. Per centrare i cursori si deve smontare il "Joint" dal servo, spostarlo di uno o più denti e poi ritarare il suo delta.

Io solitamente faccio la taratura un po' a caso, per approssimazioni successive, ma qualcuno potrebbe preferire la check-list della prossima pagina, precisa, ma difficile da seguire e anche molto noiosa.

Taratura delle lunghezze e degli angoli

E' bene fare le seguenti operazioni senza pennino. Con la rotella del mouse si alza e abbassa la punta (asse Z) e muovendo il mouse sull'area gialla, con il tasto sinistro premuto, si muove la punta in orizzontale (assi X e Y). Per muovere i servo uno per uno si possono anche usare i cursori in basso, con il mouse, o con le frecce SU e GIU che permettono spostamenti micrometrici (un microsecondo per volta), utili anche per provare i servo e valutare la loro Dead-Band.

Prima di tutto si inseriscono le lunghezze dei "Link" misurandole con il calibro da centro a centro degli assi di rotazione dei motori. (misure in millimetri)

Casella "Joint 0 - LenY" = distanza orizzontale tra gli assi del motore zero (nella base) e motore 1

Casella "Joint 0 - LenZ" = distanza in verticale tra la carta e l'asse del motore 1

Casella "Joint 1 - LenY" = lunghezza del "Joint 1" (tra gli assi del motore 1 e motore 2)

Casella "Joint 2 - LenY" = lunghezza del "Joint 2" (tra gli assi del motore 2 e del motore 3)

Caselle "Joint 3 - LenX, LenY e LenZ" = spiazziamenti della punta scrivente nelle tre direzioni. (normalmente, con punta centrata, "Joint 3 - LenX e LenY" si impostano a zero)

Poi si impostano "Delta" e "Inv" con i valori delle immagini seguenti.

Arm properties					
	Inv.	Delta	LenX	LenY	LenZ
Joint 0	<input type="checkbox"/>	0		26.5	48
Joint 1	<input checked="" type="checkbox"/>	38		45	
Joint 2	<input checked="" type="checkbox"/>	0		45	
Joint 3	<input type="checkbox"/>	30	-10	0	23

Valori per servo HXT900 con links da 45mm

Arm properties					
	Inv.	Delta	LenX	LenY	LenZ
Joint 0	<input type="checkbox"/>	0		24	48
Joint 1	<input checked="" type="checkbox"/>	32		60	
Joint 2	<input type="checkbox"/>	5.6		60	
Joint 3	<input type="checkbox"/>	30	0	0	24.1

Valori per servo TGY930 con links da 60mm

Con questi valori approssimativi si centrano i link sui perni dei motori (svitandoli e avvitandoli in posizione ruotata in modo che gli angoli fisici siano approssimativamente uguali a quelli mostrati sullo schermo dalle linee grige). Poi si affinano i valori di "Joint 1 - Delta" e "Joint 2 - Delta" in modo che gli angoli fisici siano esattamente uguali a quelli mostrati su schermo. (questa operazione è più facile se si posiziona il punto rosso con il mouse in modo che il primo link sia verticale)

A questo punto muovendo il punto rosso con il mouse, la punta del braccio robotico dovrebbe muoversi in modo sensato. Se così non avviene vuol dire che alcuni assi ruotano al contrario (esistono modelli di servo che girano al contrario, oppure alcuni servo sono stati montati dal lato opposto rispetto al mio progetto). Lasciamo perdere le caselle "Inv.0" e "Inv.3" che, per ora, non provocano grossi danni anche se sono sbagliate e proviamo a invertire la "Inv.1" e la "Inv.2" fino a che si riesce ad ottenere il comportamento giusto.

Se la punta del braccio non si muove su piani ortogonali i rettangoli diventano dei trapezi e le linee rette diventano delle curve evidenti allora è bene ricontrollare tutte le lunghezze dei "Link" perché una di esse è stata impostata con un errore grossolano.

Se la punta del braccio non si muove parallelamente al piano di base si deve regolare il valore del "Join 1 - Delta" nel verso che alza la punta, se allontanando la punta dalla base questa si abbassa, e viceversa se si alza.

Se un servo tende ad arrivare a fondo corsa l'area di lavoro si riduce ed è bene svtarlo, avvitarlo in posizione ruotata e rifare la taratura del suo "Delta".

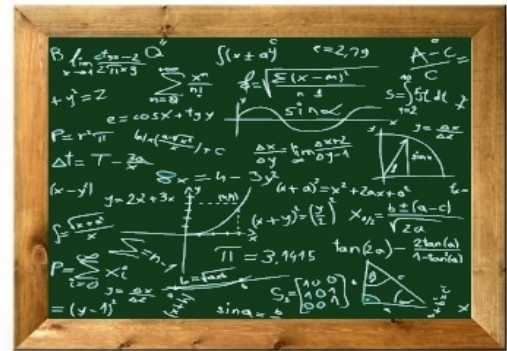
Infine si mette il pennino e si fa in modo che stia sempre verticale. Poi, con la rotella del mouse, si fa scendere la punta fino a che il valore "Tip Z" segni zero e si regola il valore della casella "Joint 3 - Len Z" in modo che sfiori la carta senza premere troppo.

Cinematica diretta e inversa

La **cinematica diretta** calcola dove si trova la punta a partire dagli angoli dei "Joint".

La **cinematica inversa** calcola gli angoli che si devono dare ai "Joint" per far andare la punta dove si vuole.

In tutti e due i casi i "Link" rappresentano un collegamento rigido e quindi sono delle costanti.



Trovare formule corrette in rete si è rivelato molto difficile tanto che alla fine ci siamo dovuti mettere in due, con carta penna e Derive6 (programma che aiuta i non-matematici a semplificare le formule), e abbiamo dovuto lavorare per molti giorni prima di riuscire a far quadrare il tutto in un algoritmo ben funzionante.

Ecco i riferimenti ad alcuni siti sulla cinematica, usateli solo per apprendere i principi in quanto, come spiegato sopra, tutte le formule copiate da internet sono risultate incomplete o contenenti errori: www.learnaboutrobots.com/inverseKinematics.htm
e: freespace.virgin.net/hugo.elias/models/m_ik.htm

Avrei voluto scrivere qui le formule finali ricavandole dal software, ma mi sono accorto che tradurre le sei pagine di algoritmi in formule matematiche è praticamente impossibile. In un algoritmo si possono mettere limiti ai valori intermedi e imporre vincoli (ad esempio questo progetto impone che la penna sia sempre verticale) I vincoli riducono le possibili soluzioni da infinite a due sole, di cui una evidentemente sbagliata perché si trova sotto al tavolo.

Io non sono in grado di tradurre tutto questo in notazione matematica, teoricamente si potrebbe fare ma la formula risultante sarebbe mostruosa e incomprensibile.

Per cui chi è veramente interessato ai particolari della cinematica dovrà andare a cercarli nel codice sorgente della applicazione "RobotArm", modulo "RobotArm", funzioni "InverseKinematicCompute" e "DirectKinematicCompute"

Tenete conto che queste funzioni valgono solo per un robot "articolato" con quattro "Joint" e quattro "Links" e con la stessa geometria del RobotArm di questo progetto. Fortunatamente non vi è alcun motivo di aggiungere o togliere motori. Le aggiunte banali come una pinza per prendere oggetti non modificano la geometria di base, quindi gli algoritmi continueranno a funzionare. Però variazioni sostanziali di geometria, come ad esempio la struttura "scara", richiederebbero algoritmi completamente diversi.

Un'ultima avvertenza: la funzione "DirectKinematicCompute" non è completa per cui non ho potuto usarla per calcolare l'area utile (in giallo) ed ho dovuto usare al suo posto un sistema di approssimazioni successive che è scomodo e molto più lento. Se qualcuno riesce a finire questa funzione con successo mi scriva a "livio@freetw.net", grazie.

Dove reperire i componenti hardware

I servocomandi HXT900 e i TGY930 sono reperibili presso www.hobbyking.com dove si possono trovare anche moltissimi altri modelli di servo. Chi volesse fare esperimenti può provare anche servocomandi diversi ma, difficilmente ne troverà di più adatti di questi, che sono stati selezionati acquistandone una quindicina di modelli, tra i più promettenti.

A mio avviso il modello HXT900 è imbattibile come rapporto qualità, prezzo, costa solo 1.9 Euro ed è abbastanza preciso. Invece il TGY930 è in assoluto il più preciso tra tutti quelli che ho provato, con una dead-band anche tre o quattro volte inferiore rispetto a tutti i modelli digitali provati, anche se assai più costosi.

Inoltre il modello TGY930 ha due cuscinetti sul perno principale, non ha gioco sul perno ed è molto potente (3.6 Kg/cm), pur consumando abbastanza poco da poter essere alimentato dall'USB. ATTENZIONE: se non trovate il TGY930 cercatelo con la sigla MG930 (è lo stesso servo).

Per la comunicazione con il PC via USB ho usato i moduli di InOut del sistema Theremino, un "Master" e uno "Slave Servo". Chi non volesse usare questi moduli potrebbe usare un hardware di InOut diverso, ad esempio un Arduino, avvantaggiandosi ugualmente del software Theremino_Arm.

In ogni caso l'applicazione Theremino_Arm non necessita di modifiche perché, essendo modulare, scrive negli "Slot", facilmente leggibili anche da uno sketch di Arduino. Informazioni su come usare gli slot sono reperibili sul sito www.theremino.com in: "Downloads/Foundations/ Interprocess Communications" e nella pagina "Technical/Communications" nel capitolo "Gli SLOT".

Note finali ed eventuali miglioramenti

La ripetibilità di posizionamento del RobotArm è eccezionale ma sarebbe desiderabile una maggiore precisione. L'imprecisione è dovuta per il 99% al riferimento interno del servo che, purtroppo, è solo un potenziometro molto piccolo. La scarsa precisione genera linee non perfettamente dritte e un po' tremolanti. Per aumentare la precisione prima di tutto si scartino i servo HXT900 in favore dei TGY930 che sono molto migliori. Poi si potrebbero sperimentare altri modelli (ci vuole esperienza per valutare la loro precisione e la dead-band, usate un bastoncino da circa 25 centimetri, una riga millimetrata, e muovete il servo con i cursori e le frecce SU/GIU di 1 uS per volta) Probabilmente esistono servo più precisi dei TGY930, se qualcuno li trova lo comunichi, per piacere, a: "livio@freetw.net"

L'applicazione RobotArm è sicuramente migliorabile, consideratela come un esempio e una collezione di funzioni utili. La lettura del GCode e la traduzione in segmenti sono rudimentali e probabilmente manca il segmento di chiusura di ogni percorso (visibile nei test - i cerchi non si chiudono bene) Sarebbe anche carino salvare i parametri di lavoro (zoom e traslazioni) individualmente per ogni GCode. Se fate miglioramenti inviatemeli. (è quasi pronta la versione 1.2 : [scaricare da qui](#))

Ho scoperto con la pratica che i parametri del riquadro "Arm properties", dopo una buona taratura, non hanno più bisogno di correzioni, tranne i due seguenti:

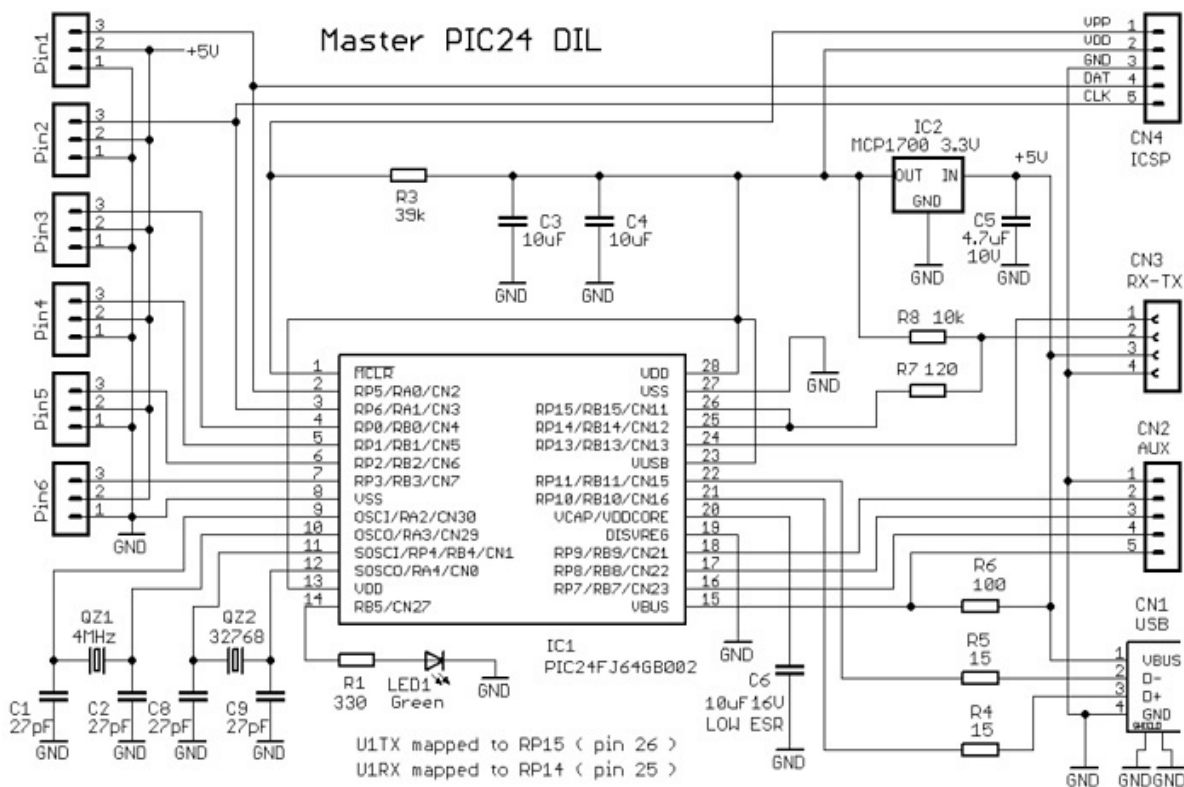
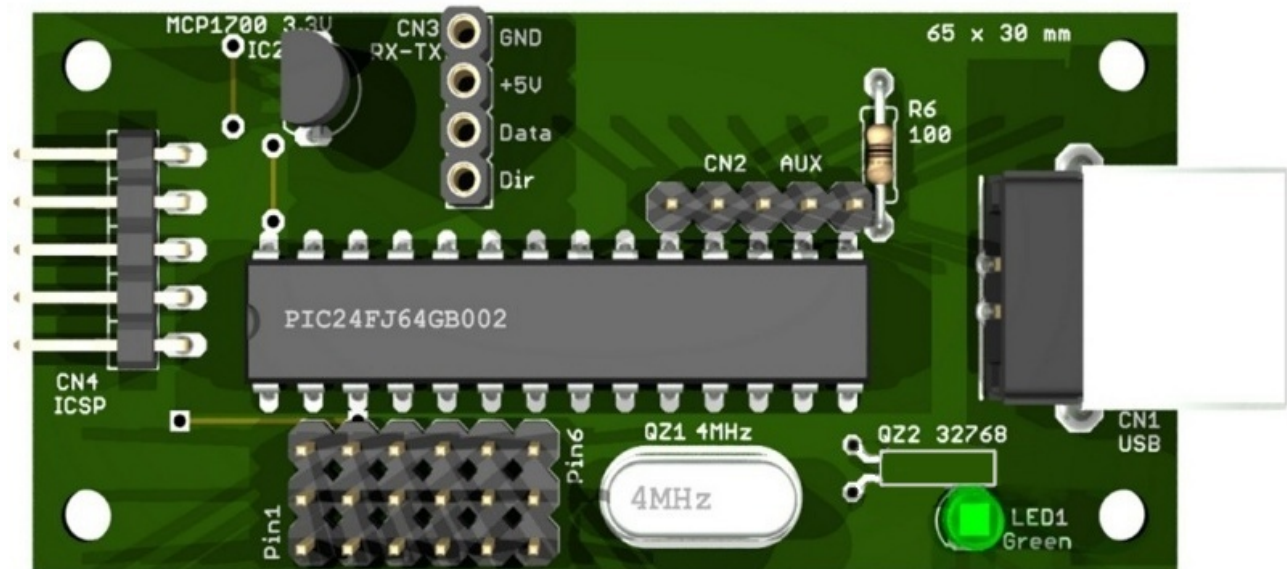
"Joint 1 - Delta" = Correzione per la penna che si alza o abbassa allontanandola dalla base.

"Joint 3 - LenZ" = Correzione della altezza della penna, modificabile anche mentre scrive.

Fine

A questo punto la trattazione sarebbe finita, ma un vero MAKER deve farsi tutto da solo, anche le viti parker, e non accetterà mai di comprare i moduli Master e Slave di produzione industriale. Il MAKER preferisce spendere il doppio per i componenti e lavorare un mese di più, ma deve arrivare alla fine in modo completamente "DIY" e "Open". Non importa la spesa e il tempo, quello che il MAKER costruisce non è un robot, ma la propria conoscenza, che vale più di ogni altra cosa.

Quindi questa è la fine, ma non per tutti. I veri, impavidi, MAKERS possono continuare e farsi anche l'elettronica completamente "DIY" scaricando lo zip: [MasterDIL DIY.zip](#)

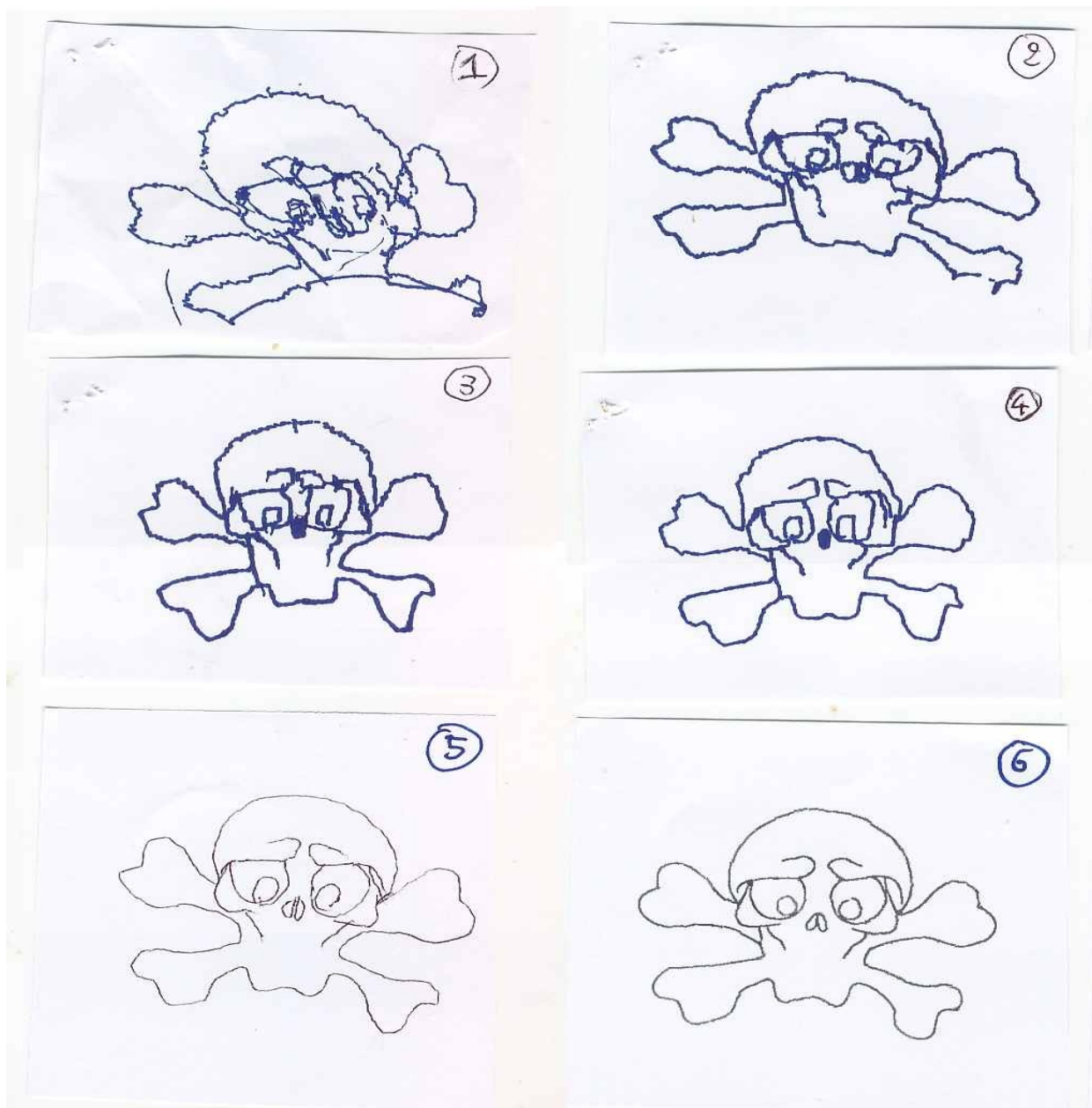


Il file ZIP contiene i progetti completi del PCB (in formato Eagle) e del firmware (MpLab) con schemi e immagini in 3D. Se vi serve aiuto scrivetemi: livio@freetw.net

Auguri a tutti i coraggiosi

Intraprendere la costruzione di questo braccio robotico è un progetto ambizioso, non tanto per la parte meccanica che si risolve con due squadrette e qualche vite, ma perché è necessaria una buona dose di abilità per riuscire a tararlo bene. Molti si perderanno per strada, altri ci riusciranno, ma tutti avranno imparato qualcosa sulla robotica.

Per farvi coraggio, o demoralizzarvi del tutto, guardate con attenzione l'immagine della pagina seguente, magari ingranditela. Per passare dalla qualità del provino "1" alla "6" ci sono voluti alcuni mesi!



Da "1" a "4" con pennarello e servo HXT900, la "5" e la "6" con i servo TGY930, la "5" con penna a sfera normale e la "6" con penna a gel e... mesi di esperienza.

Buon MAKE a tutti.
Livio Cicala & Roberto Cena