



UNIVERSITA' POLITECNICA DELLE MARCHE

Corso di Laurea Triennale in
INGEGNERIA INFORMATICA E AUTOMAZIONE

**Studio della sostituzione di un robot
pneumatico ad 1 asse con un robot
antropomorfo a 6 assi**

**Study of the replacement of a 1-axis
pneumatic robot with a 6-axis
anthropomorphic robot**

**Relatore:
Prof. Andrea Bonci**

**Candidato:
Igor Nociaro
Mat. s1083746**

Anno Accademico 2022-2023

Prefazione

Questo elaborato è il culmine del mio tirocinio durato circa 4 mesi presso l'azienda Elettromeccanica Pantanetti, ormai affermata nell'ambiente industriale in diversi settori ed esperta nel campo calzaturiero. In particolare il loro lavoro si concentra nella progettazione di quadri elettrici su misura di macchine costruite da altre aziende partner e talvolta, come in questo caso, l'aggiornamento di vecchi macchinari non più adatti agli standard di qualità odierni. Nel tempo di permanenza ho seguito passo per passo il processo che si cela tra il colloquio con il cliente e la messa in servizio.

Comau e Siemens sono due costanti nei progetti di qualsiasi cliente, al momento della stesura di questo testo, Pantanetti elettromeccanica ha installato il 35° robot Comau di cui 9 predisposti per la colata del poliuretano. Per questo molte informazioni che leggerete provengono dalla diretta esperienza dei dipendenti. Nonostante la sostituzione del "distributore a un asse" con il braccio industriale sia l'argomento principale della tesi è interessante conoscere, anche in minima parte, il resto del macchinario per semplificare la comprensione di alcune scelte progettuali che comprendono anche la messa in sicurezza e la certificazione.

Sommario

Isocianato e poliolo, conosciuti dagli addetti ai lavori come ISO e POL, sono i due materiali reagenti che formano il poliuretano. Vengono inseriti contemporaneamente in una camera di miscelazione e una volta che la centrifuga inizia a mischiare i materiali bisogna fare in fretta. Pochi secondi e il poliuretano si espande, prendendo la forma del suo contenitore. Fortunatamente la camera di miscelazione ha un ugello da cui far uscire il composto, è stata studiata in questo modo. Tutto il processo è stato pensato per essere ripetibile già dagli anni '80. Il boom del settore calzaturiero ha portato macchine che permettono di automatizzare la colata, dalle *banane* semiautomatiche si è passati alle rotative per aumentare la produzione. Puntare sulla quantità ha tolto attenzione alla tecnica di colata e ad oggi è rimasto lo stesso di cinquant'anni fa. Ora ai grandi marchi interessa la suola di qualità, peso consistente, con dettagli ben visibili e privi di imperfezioni. Il vecchio "distributore ad un asse" pneumatico non è più una soluzione: l'imprecisione con cui cola all'interno dello stampo forma molte bolle e provoca una grande quantità di scarti. Supponendo che non ci siano problemi nella composizione dei materiali e nel metodo di colata, uno dei modi di migliorare un macchinario del poliuretano è fluidificare i movimenti del colatore, limitando in questo modo i difetti sulla suola finale. In questa tesi si seguono da vicino i passaggi per sostituire il robot colatore, con particolare attenzione ai benefici ottenuti.

Indice

Elenco delle figure	iv
0 Introduzione	1
0.1 Obiettivi	2
0.2 Cenni teorici	3
0.2.1 Protocollo Profinet	3
0.2.2 Il PLC	4
0.2.3 Tipi di motori e come avviene il controllo	5
0.3 Software utilizzati	7
0.3.1 Aucotec ELCAD	7
0.3.2 TIA Portal	8
0.3.3 RoboShop	9
0.3.4 SolidWorks	9
1 Breve illustrazione del macchinario	10
1.1 Tavola rotante	11
1.2 Robot	12
1.3 Testa di colata	12
1.3.1 Pompe di servizio	14
1.4 Funzionamento	17
1.4.1 Ricette e cambio mix	17
1.4.2 Taratura del poliuretano	18
2 Robot antropomorfo 6 assi NJ-60-2.2	19
2.1 Cinematica	21
2.1.1 Sistemi di riferimento	22
2.2 Programmazione	24
2.2.1 Comunicazione tra PLC e robot	25
2.3 Preparazione e sostituzione	26

3	Sicurezza in generale	29
3.1	S.T.O. - Safe Torque Off e sicurezza del robot	30
3.2	Accorgimenti nella costruzione	31
4	Conclusioni	33
4.1	Possibili sviluppi	33
	Appendice A Programmazione del robot	35
A.1	Diagramma di flusso di un task generico	35
A.2	Main	36
A.3	Punto e tratto	42
A.4	Task lavaggio	46
A.5	Task precolata	46
A.6	Task taratura in sicurezza	47
A.7	Comunicazione con il PLC	50
	Appendice B Comunicazione lato PLC	52
B.1	DB dati ricevuti	52
B.2	DB dati trasmessi	53
B.3	Ricezione e trasmissione in AWL	54
B.4	Multiplexer scambio dati	55
	Bibliografia	60

Elenco delle figure

1	Single shot process [1]	1
2	Robot pneumatico ad un asse	2
3	Accuratezza e ripetibilità [15]	3
4	Configurazione della rete Profinet di questo progetto	4
5	ET 200SP con schede di I/O	5
6	Occupazione della memoria nella CPU 1513	5

7	Unità di controllo e moduli di potenza utilizzati	7
8	Programmatore portatile TP5	9
9	Vista dall'alto del robot con CAD	10
10	Un altro robot colatore	12
11	Testa di colata senza coperchio di sicurezza	13
12	Centrifuga prima di essere assemblata	14
13	Pistone con valvola di accesso dei materiali alla camera di colata	15
14	Bocchettoni di ricircolo per tubi di ISO e POL	16
15	Pulsantiera di controllo	17
16	Impostazioni di una singola stazione	18
17	Specifiche tecniche del robot NJ60 [4]	19
18	Assi di un generico robot antropomorfo	20
19	Area di lavoro SMART-5 NJ60 2.2	22
20	Terne di riferimento del sistema robot [5]	23
21	Passi per la creazione del frame [5]	23
22	Robot in azienda per prove	26
23	Robot nella sua posizione definitiva	28
24	Esempio di matrice di rischio [8]	29
25	Categorie di stop [14]	30

0 Introduzione

Come stampare il poliuretano non è un segreto, chiunque lavori nel settore calzaturiero conosce la sua composizione e i metodi con cui si producono soles. Il macchinario più popolare utilizza un poliuretano formato da due a cinque ingredienti [2], i fondamentali Isocianato (ISO) e Poliolo (POL), più gli additivi che fungono da catalizzatori ed eventualmente, i pigmenti di colore. Tutto viene inserito contemporaneamente all'interno di un cilindro metallico e mescolato da una centrifuga a diecimila giri (figura 1), il cilindro non ha bisogno di essere grande perché il materiale una volta dentro scende senza incontrare ostacoli, il composto che ne fuoriesce è poco viscoso e inizierà ad espandersi in pochi secondi. Il poliuretano è considerato un materiale moderno e il metodo del miscelatore ad uno stadio

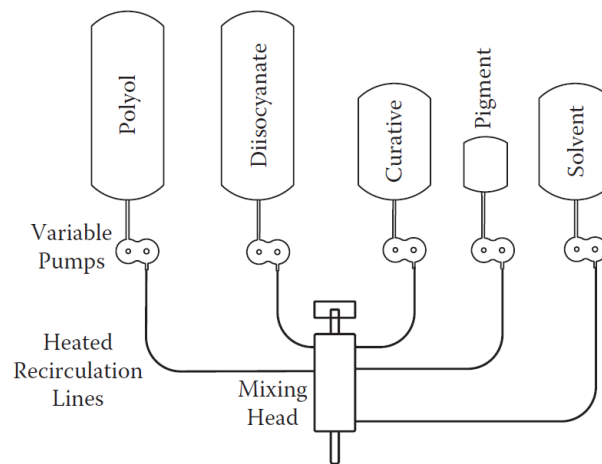


Figura 1: Single shot process [1]

non si è mai aggiornata dal suo concepimento. Come è possibile allora migliorare il prodotto finale se non ci sono difetti nei metodi di colata? Come riempire lo stampo non è un aspetto da sottovalutare. Le soles con geometrie particolari o più semplicemente con il tacco, richiedono che più materiale sia depositato dove il volume da riempire è maggiore. Qui entra in gioco il robot antropomorfo, difatti fare movimenti precisi e veloci è la sua specialità. Il miscelatore viene così installato sul polso del robot, pronto ad eseguire tutti i movimenti richiesti senza nessuna difficoltà.

0.1 Obiettivi

Lo scopo del lavoro è ammodernare il macchinario, non solo sostituendo il colatore ma anche rimpiazzare i vecchi motori delle pompe con nuovi motori più affidabili e assumere il controllo della macchina rotativa. Data la portata del progetto, ci si concentrerà soprattutto nella sostituzione del robot, i motivi della scelta, la sua programmazione e le sicurezze impiegate.

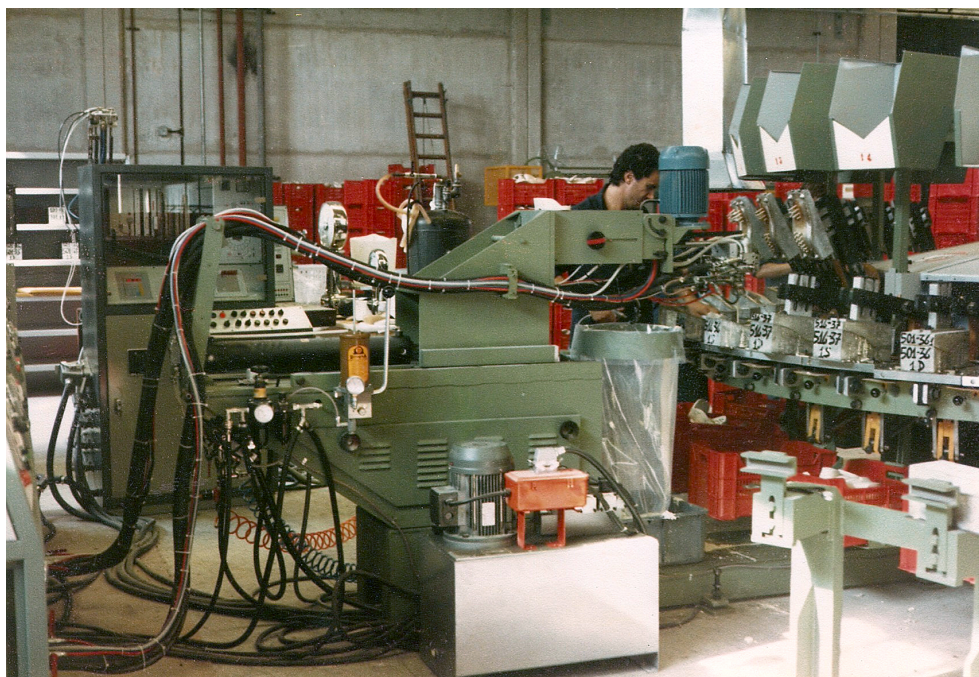


Figura 2: Robot pneumatico ad un asse

Prima di iniziare occorre definire la differenza tra accuratezza, precisione e ripetibilità (figura 3). Facendo una analogia con il colatore, l'accuratezza è quanto più vicino al luogo desiderato dello spazio si può posizionare l'ugello di colata, mentre con precisione si intende la capacità di collocarlo in una certa posizione e orientamento con un errore inferiore a un determinato valore. Nella maggior parte delle applicazioni la dote più importante per il movimento di un sistema è la ripetibilità, ovvero la dispersione massima della posizione raggiunta dovuta all'errore, calcolati come dice il nome, ripetendo la medesima traiettoria molteplici volte. Gli errori di posizionamento si producono in seguito a diverse limitazioni della struttura, come possibili disallineamenti o la presenza di giochi meccanici, oppure dalla risoluzione dei sensori di posizione e altri fattori. Se un sistema ha una buona ripetibilità

(quindi precisione) si possono applicare delle compensazioni sul movimento in modo da migliorarne l'accuratezza. La precisione del robot pneumatico (figura 2) degrada con l'utilizzo e l'errore raggiunto poteva essere anche più di un centimetro! Oltre al chiaro vantaggio della ripetibilità nelle operazioni, il robot antropomorfo è molto uniforme nei movimenti, è decisamente più veloce e permette di eseguire le operazioni di taratura del poliuretano in modo comodo e sicuro.

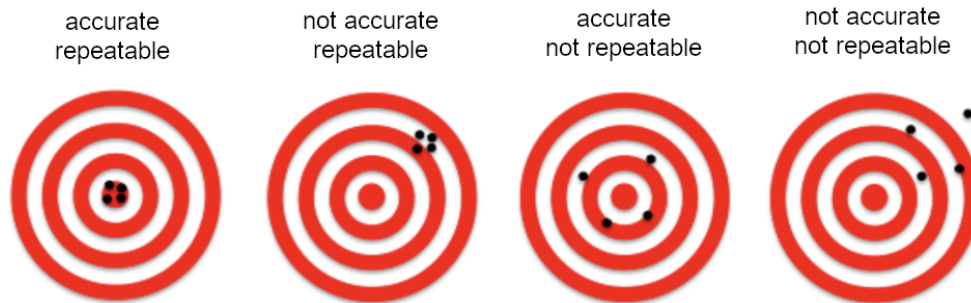


Figura 3: Accuratezza e ripetibilità [15]

0.2 Cenni teorici

Protocolli, componenti e software utilizzati.

0.2.1 Protocollo Profinet

Il protocollo Profinet è un protocollo Siemens evoluzione del protocollo seriale Profibus. Fisicamente utilizza la configurazione dei fili e il connettore RJ45 di un comune cavo Ethernet ma con protezioni aggiuntive dai disturbi elettromagnetici. Oltre alla schermatura esterna, ogni *twisted pair* ne ha una propria. A tutti i dispositivi Profinet viene assegnato un indirizzo IP (TCP/IP) e collegato direttamente al controllore (PLC) o in cascata con altri dispositivi. La comunicazione avviene in RT (real-time), ovvero con tempi di ciclo di 10 ms, e IRT (isochronous real-time) che abbassa i tempi fino a 1 ms rendendo il Profinet adatto anche alle applicazioni di *motion control* in cui si richiede un uso simultaneo di molti assi [12]. Nella rete locale costruita si può teoricamente assegnare qualsiasi indirizzo al PLC, l'importante è che tutti i dispositivi abbiano la stessa maschera sottorete, tuttavia anche per le reti locali esistono delle direttive da seguire. Lo IANA definisce delle classi di indirizzi IP per uso "privato" che non vengono passate all'esterno del router (ad esempio 192.168.x.x per la rete

domestica) [12].

In caso di necessità è possibile collegare alla rete un componente aggiuntivo che permette, attraverso una VPN, di connettersi al PLC comodamente dalla rete aziendale. Quando il cliente ha un problema, con questa soluzione è possibile verificare lo stato della rete e caricare aggiornamenti del software risparmiando sul tempo del viaggio.

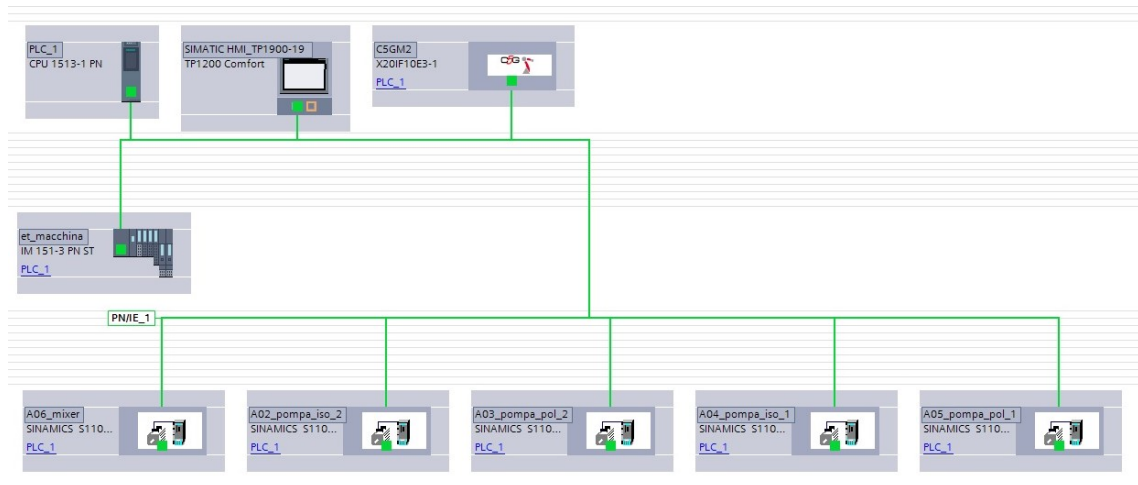


Figura 4: Configurazione della rete Profinet di questo progetto

0.2.2 Il PLC

Quando si parla di PLC, si parla con ogni probabilità anche di CPU. Non è un caso infatti che chi frequenta gli ambienti industriali utilizzi questi termini come sinonimi ma che talvolta vengono considerati - impropriamente - come due componenti separati. Nel vecchio quadro elettrico (precedente alla sostituzione) veniva utilizzato un ET 200S (6ES7151-8AB01-0AB0) cioè un sistema di periferia modulare nel fattore forma della serie di CPU Siemens 300, con moduli di ingresso (6ES7131-6BF01-1BA0) e moduli di uscita (6ES7132-6BF01-1BA0). Questo modello di ET 200S è una CPU, si può quindi utilizzare stand alone senza l'ausilio di altri componenti oltre alle schede di espansione. Con l'aggiunta del robot antropomorfo, il programma nel PLC si è appesantito ed è stato necessario fare delle valutazioni per cambiare il 200S, la cui memoria di programma si è più che saturata. Il costo dei moduli I/O, la potenza di calcolo, la dimensione massima del programma e la ritentività (dimensione della memoria non volatile) sono i fattori che hanno portato alla scelta di un ET 200SP (6ES7155-



Figura 5: ET 200SP con schede di I/O

6AU01-0BN0), riutilizzando in questo modo i moduli I/O e una CPU 1513 (6ES7513-1AL02-0AB0). Questo ET viene chiamato PLC anche se non ha una CPU al suo interno ed è dunque una semplice periferia modulare. Nel catalogo Siemens esistono ET che hanno al loro interno CPU della serie 1500 ma che non avrebbero comunque raggiunto le specifiche per far funzionare il programma correttamente. Nella figura 6 si nota come la memoria abbia un giusto compromesso di impiego, considerando il fatto che è stato utilizzato il modello più povero tra quelli della serie 1500. CPU ed ET comunicano tra loro per mezzo di un cavo Profinet.

Memoria utilizzata PLC_1												
Oggetti	Memoria di carica...	Memoria di progra...	Memoria di lavoro ...	Memoria a ritenzion	Risorse Motion Con...	I/O	DI	DO	AI	AO		
1	32 %	71 %	5 %	72 %	0 %		96 %	86 %	0 %	3 %		
2												
3	Totale:		12 MB	307200 byte	1572864 byte	90784 byte	800	Configurato:	48	36	132	136
4	Occupato:		3974891 byte	218634 byte	77136 byte	65208 byte	0	Occupato:	46	31	0	4
5	Dettagli											
6	▶ OB	60453 byte	2110 byte									
7	▶ FC	3119749 byte	214688 byte									
8	▶ FB	30980 byte	1836 byte									
9	▶ DB	656093 byte		77136 byte	65208 byte							
10	Oggetti per Motion Technology											
11	▶ Tipi di dati	38859 byte			0 byte	0						
12	Variabili PLC		68757 byte		0 byte							

Figura 6: Occupazione della memoria nella CPU 1513

0.2.3 Tipi di motori e come avviene il controllo

Questa applicazione richiede carichi di lavoro particolari. I 5 motori elettrici devono essere correttamente selezionati dal catalogo Siemens che comprende tra i motori a "bassa tensione" (fino a 5,3

MW) una gamma professionale di brushless. 4 motori, dedicati alle pompe dosatrici, non possono mai fermarsi ed è necessario che siano precisi perché devono regolare il flusso del materiale, il motore rimanente invece necessita di raggiungere la velocità massima in poco tempo per poi fermarsi bruscamente, ripetendo questo ciclo anche ogni due secondi. Tra i parametri, abbiamo bisogno di conoscere in tempo reale la velocità di marcia; ciò si consegue attraverso un resolver (un encoder relativo analogico) già racchiuso all'interno dei motori brushless, il suo cavo di alimentazione e dei dati sono separati dalla parte di potenza del motore e vengono gestiti dalla unità di controllo (CU).

Per muovere e comandare un motore elettrico si utilizza un azionamento, cioè l'insieme di una CU e un modulo di potenza (PM). Mentre il PM si sceglie in base alla potenza del motore, nella scelta delle CU si considera il numero di motori da controllare. Il vantaggio è la modularità del sistema: una stessa CU può essere attaccata ad un PM da 5 kW o ad uno da 100kW risparmiando sui costi. Normalmente per controllare più motori si utilizza una CU come la CU320, la quale permette di controllare fino a 4 motori ma bisogna stare attenti alla contemporaneità. Se un azionamento può controllare quattro motori non per forza può gestirli simultaneamente.

In questo caso per via dei tempi di consegna e delle disponibilità in magazzino si è deciso di utilizzare 5 CU305 (6SL3040-0JA01-0AA0, ogni singolo motore è controllato da una CU) con 5 PM240. Nella figura 7 vengono mostrati i 5 azionamenti all'interno del quadro elettrico che nonostante il fattore forma sia identico, hanno potenze differenti. I 4 motori delle pompe sono da 1,1 kW e hanno un PM (6SL3210-1PB17-4AL0) da 1,5 kW mentre il motore del mixer è da 2,5 kW con un PM (6SL3210-1PE21-4AL0) da 4 kW. Come si può notare, il modulo di potenza del mixer è sovradimensionato; questo è giustificato dal fatto che il lavoro compiuto dal motore è molto gravoso, fermarsi velocemente richiede anche che venga attaccato all'azionamento un sistema di "recupero dell'energia" fatto con delle resistenze, il cui valore e potenza complessiva per quello specifico azionamento sono valori consigliati da Siemens.

Una volta che il motore brushless con encoder è stato cablato si esegue una procedura di autoapprendimento, possibile anche senza PLC. L'azionamento inizia a far muovere brevemente il motore compiendo accelerazioni e decelerazioni e dopo qualche minuto, con i dati ricevuti dal motore, riesce a ricavarsi i tre valori del regolatore PID necessari per seguire correttamente il profilo di velocità (un trapezio). Anche se è un procedimento che si dovrebbe fare con il motore nella posizione definitiva e quindi con una certa massa inerziale, si preferisce eseguirlo a vuoto per poi andare a cambiare

lievemente i parametri del PID.



Figura 7: Unità di controllo e moduli di potenza utilizzati

0.3 Software utilizzati

Il programma del robot e il programma del PLC sono differenti. Mentre l'unità di controllo del robot fa le sue ottimizzazioni del movimento, il segnale per eseguire il task viene dato dal PLC.

0.3.1 Aucotec ELCAD

CAD per schemi elettrici, veniva utilizzato da Siemens molto tempo fa ma ora utilizzano tutti un altro software con una licenza per niente economica. Per questo alcune aziende ancora lo utilizzano, con molte difficoltà visto che il catalogo componenti non è sempre aggiornato. Molti accorgimenti per la sicurezza dell'impianto vengono fatti in questa fase della progettazione. Il neutro si aggiunge soltanto se nell'impianto sono presenti elementi riscaldanti perché utilizzano una alimentazione monofase, in

questo caso non ne abbiamo bisogno, l'unica alimentazione da fornire attraverso il quadro è la trifase per il robot (insieme ai 24V in continua per il PLC).

Ogni parte del quadro elettrico acquisisce un proprio codice, se si ha bisogno di trovare un componente dallo schema elettrico nella pagina =0+H1.D4, si conosce, in ordine: impianto, locazione (numero di armadietto) e funzione. Viceversa, aprendo il quadro elettrico, si legge l'etichetta presente sul componente e si può risalire alla pagina esatta che lo utilizza. Infine, dopo la messa in servizio, lo schema elettrico viene sempre stampato e il cliente viene fornito di una sua copia.

0.3.2 TIA Portal

Totally Integrated Automation Portal è un software di Siemens che permette di programmare PLC e interfaccia utente dalla stessa piattaforma. I molteplici linguaggi di programmazione possibili da utilizzare sono strutturati e non strutturati. L'SCL è quello che si potrebbe trovare più intuitivo perché segue la sequenza dei comandi scritti, realizzandoli uno dopo l'altro similmente ad altri linguaggi come Java e C. Diversamente, KOP e FUP (dal tedesco) sono due linguaggi drag and drop, rispettivamente a contatti(Ladder) e a porte logiche(FBD) più adatti alla programmazione real-time e a chi non ha competenze specifiche. Esiste un quarto linguaggio chiamato AWL. Quest'ultimo è di basso livello e molto simile all'assembly tuttavia è ancora ampiamente utilizzato per la semplicità con cui permette di eseguire alcune funzioni. Ogni parte di linguaggio deve essere contenuto all'interno di un blocco e questo ne facilita anche la lettura:

- organizzativo(OB) - gestione di altri blocchi
- programma(FC) - codice fondamentale al funzionamento dell'applicazione
- funzione(FB) - contiene spesso parti complesse e ripetitive di codice
- database(DB) - strutture dati utilizzate dal programma

Il main è un OB in cui si mettono i blocchi che il PLC esegue in modo parallelo. Una volta programmate le FC e le FB necessarie se ci dimenticassimo di aggiungerle al main, tutto il progetto risulterebbe vuoto.

0.3.3 RoboShop

E' il software proprietario di Comau e utilizza il PDL2, un linguaggio di programmazione strutturato simile al Pascal. Oltre ai tipi di dato standard ha un tipo chiamato POSITION, contiene tre componenti di posizione (X,Y,Z), tre angoli di Eulero per l'orientamento (E1,E2,E3) di tipo REAL ed una componente di tipo STRING [6]. Con il tipo JOINTPOS invece si indica sempre una posizione nello spazio ma definita da un vettore di REAL, ogni componente rappresenta l'angolo di un giunto del robot. La programmazione può essere eseguita con lo stesso linguaggio anche sull'unità TP5 (figura 8) fornita insieme al robot.



Figura 8: Programmatore portatile TP5

0.3.4 SolidWorks

L'utilizzo di un CAD è ormai necessario per una parte consistente del lavoro. Non solo per la progettazione di cosa andrà poi attaccato al polso del robot ma anche della posizione di cavi e tubi in modo che non ne compromettano il movimento. Altre cose da tenere in considerazione sono l'area di raggiungibilità del braccio antropomorfo (figura 19) e soprattutto verificare in quali punti potrebbe urtare oggetti, compresa la gabbia di protezione. Quando poi si andrà ad installare il robot dal cliente, tutta questa fase di pianificazione si traduce in una messa in servizio più rapida e con minori equivoci.

1 Breve illustrazione del macchinario

L'area di lavoro del macchinario si può concettualmente dividere in sezioni:

- Tavola rotante
- Testa di colata
- Robot
- Serbatoi e pompe materiale

Intorno alla macchina rotativa si concentrano la totalità dei lavoratori, poi il macchinario si sviluppa lungo una linea, con il robot colatore e i serbatoi di materiale subito dopo.

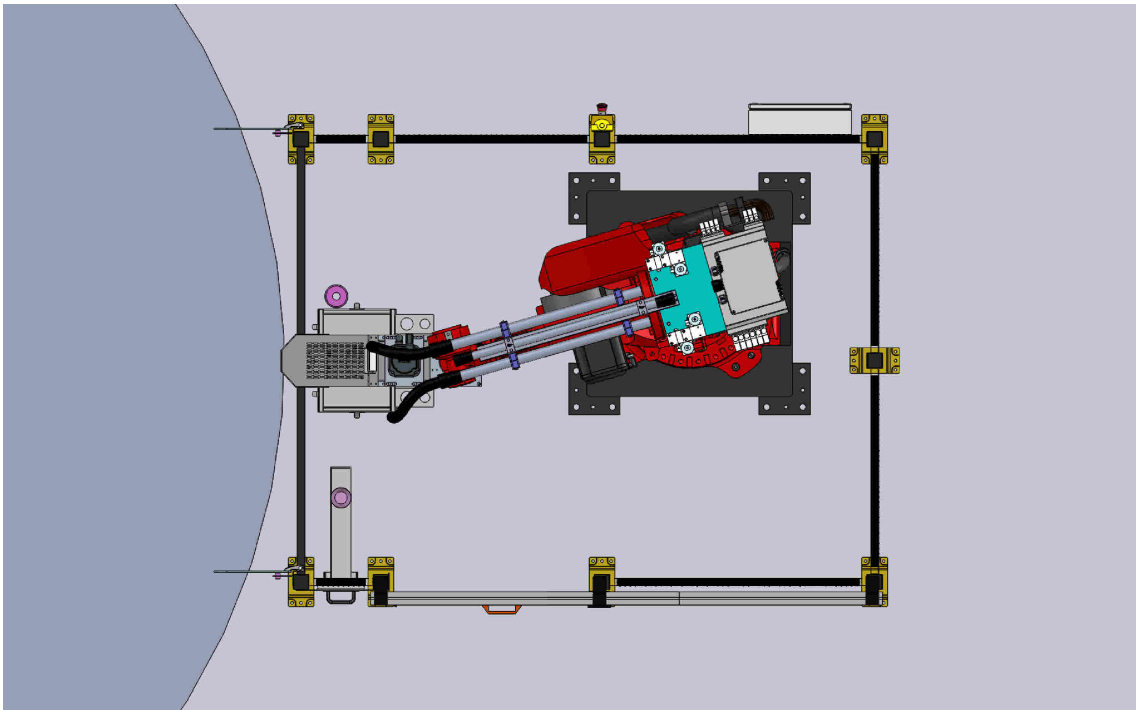


Figura 9: Vista dall'alto del robot con CAD

Nella figura 9 si evidenzia come la gabbia di protezione sia molto contenuta. Generalmente non è così, in questo caso la posizione dei quattro serbatoi da 200l ciascuno non ha permesso di fare altrimenti. Neanche spostare il robot è un'opzione. L'antiquata macchina rotativa oleodinamica apre e chiude

gli stampi sempre nella stessa porzione di tavola lasciando poco margine di manovra. Alla fine si è optato per posizionare il robot leggermente angolato per diminuire lo spazio occupato con l'elemento terminale sempre rivolto verso la macchina. Per definire il colatore non esiste un linguaggio unificato, spesso si considera come l'insieme di robot e testa di colata, mentre la testa di colata comprende il sistema formato da valvola e mixer.

1.1 Tavola rotante

La macchina rotativa in questione è a 40 stazioni. Significa che a ogni angolo giro della tavola eseguito a pieno regime di produzione, si stamperanno 40 soles, 20 per la scarpa destra e 20 per la sinistra ma niente vieta di produrre altri oggetti in poliuretano utilizzando stampi che non siano di scarpe. Il numero di stazioni (o posti) non influisce in modo significativo sulla programmazione del PLC perché ogni stazione ha il proprio blocco funzione. Non cambia neanche se la macchina è a doppio giro (al secondo giro viene fatta una colata di un diverso mix di poliuretano con la prima suola ancora nello stampo) infatti in questo caso il primo posto del secondo giro sarà contato come quarantunesimo posto invece di ripartire da uno.

La gestione del movimento rotativo viene fatta dai componenti al centro della macchina, chiamato centro tavola e comprende dunque anche un collettore che permette ai cavi elettrici di ruotare liberamente senza intrecciarsi. Nella sostituzione non si va a modificare il controllo però si prendono alcuni segnali provenienti dal centro tavola per verificarne il movimento.

Stampo

Un altro fattore che influisce sulla qualità della suola finale è la precisione con cui lo stampo è stato fatto. L'alluminio viene lavorato da una macchina a controllo numerico e generalmente è formato da una base e dal coperchio. Tuttavia per soles particolari è facile che la complessità (e il costo) dello stampo aumenti, la base viene divisa in due o più parti unite da cerniere e viene aggiunta la possibilità di mettere un film in polipropilene, fatto poi aderire allo stampo per mezzo di una pompa per il vuoto.

La temperatura deve mantenersi tra 54°C e 77°C [13] anche se alcuni produttori di soles la tengono a valori mediamente superiori. La misurazione avviene con l'utilizzo di una sonda Pt100 ed è stato trovato uno stratagemma per evitare di metterne una per ogni stampo: l'inerzia termica è simile tra tutte le

forme di alluminio quindi è stata aggiunta una forma fittizia al centro tavola, quando quest'ultima raggiunge la temperatura prestabilita, tutti gli stampi saranno circa a quella temperatura.

Essendo lo stampo riempito con un liquido a bassa viscosità è possibile modellare il poliuretano a forme molto complesse e una volta avvenuta la polimerizzazione a stampo chiuso, si può estrarre senza che il materiale venga piegato o rotto perché le tensioni interne al materiale non ci sono più a differenza di altri materiali plastici [13].

1.2 Robot

La figura 10 mostra un robot di un precedente lavoro, con l'unica differenza da quello utilizzato nel colore della scocca. Di questo particolare modello se ne parla in modo più approfondito nel capitolo 2.



Figura 10: Un altro robot colatore

1.3 Testa di colata

Mentre il blocco per miscelare il poliuretano viene acquistato da un costruttore, tutto il resto della testa di colata è stata progettata al CAD, compresa la giunzione per il polso del robot. Una volta avvenuta la fabbricazione della meccanica da una ditta esterna viene assemblata in azienda. Un grande vantaggio di questo metodo è conoscere a priori peso e centro di massa della testa di colata, in questo

modo si può scegliere in anticipo il robot antropomorfo adatto e di conseguenza accelerare anche i tempi di consegna che possono rivelarsi di diversi mesi.

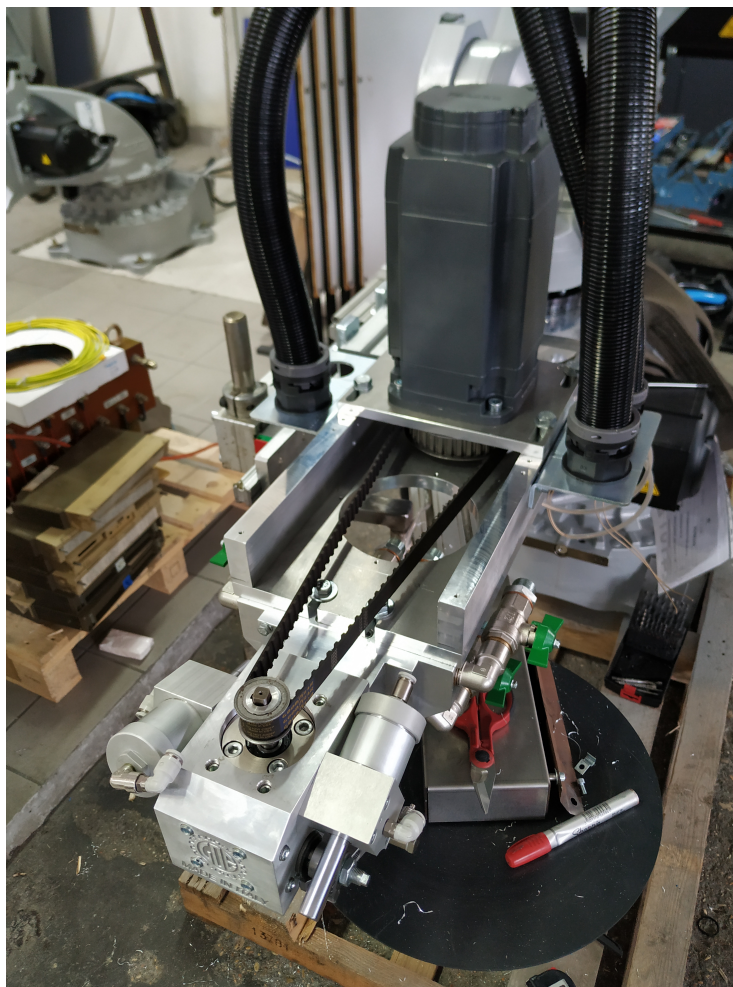


Figura 11: Testa di colata senza coperchio di sicurezza

Le coordinate X, Y, Z del colatore visualizzate sul pannello hanno una particolarità: guardando al movimento della testa sembrano non rispettare i movimenti, ed in effetti è così. Per ragioni pratiche e storiche nell'evoluzione delle macchine si utilizza un sistema di riferimento non matematicamente corretto. I primi colatori come quello sostituito avevano soltanto un asse avanti/dietro ed essendo l'unico la coordinata X era quella che indicava il loro spostamento (avanti positivo), più avanti, alcuni nuovi colatori avevano un asse aggiuntivo Y, che però non poteva sconvolgere quello che fino ad allora

era stato l'asse per colare. Quindi, considerata l'origine nell'angolo sinistro più vicino dello stampo, si è rimasti ad avere X positivo in avanti e Y positivo a destra. Con l'avvento del terzo asse e rispettando la regola della mano destra, si riconosce che sarebbe eccessivo osservare il carrello andare verso l'alto e allo stesso tempo vedere il valore Z decrescere. In sostanza si mantiene questo sistema di riferimento anche per evitare che l'operatore a seguito dell'aggiornamento della macchina si confonda con le nuove coordinate. All'interno del PLC i calcoli vengono eseguiti normalmente ma sul pannello vengono visualizzati con X e Y invertiti.



Figura 12: Centrifuga prima di essere assemblata

1.3.1 Pompe di servizio

Un grande problema per le macchine del poliuretano è la solidificazione delle componenti di ISO e POL se non vengono continuamente mosse. Se la pompa materiale si fermasse, la sostanza ha due-tre ore prima di diventare inutilizzabile per indurimento (tempo riferito ad una giornata estiva). Ma non è l'unica delle difficoltà: anche la temperatura gioca un ruolo fondamentale. Il sistema prevede un circuito di ricircolo dal serbatoio dove è contenuto il materiale, al mixer (figura 14) attraverso dei tubi riscaldati da una guaina metallica (funge da resistenza e applicando una tensione ai capi si verifica

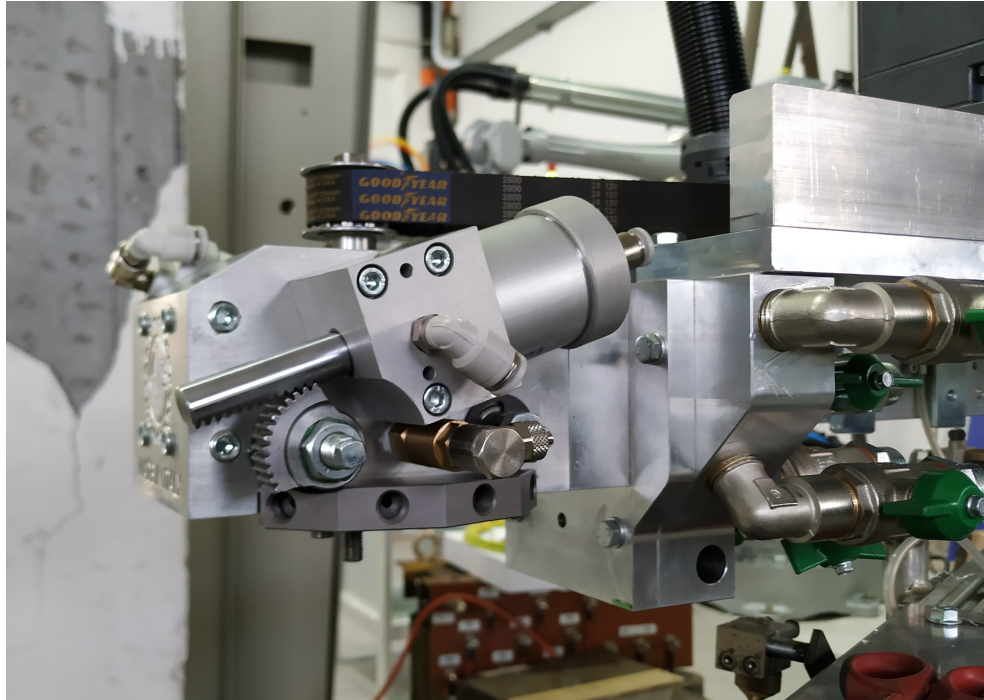


Figura 13: Pistone con valvola di accesso dei materiali alla camera di colata

l'effetto Joule). Il tutto si deve trovare ad una temperatura tra i 24°C e i 49°C [13]. I serbatoi di ISO e POL hanno un misuratore di pressione con soglia massima e minima. La minima si mette per avere un segnale di allarme in più ma non è strettamente necessaria perché non è pericoloso avere pressione bassa come quella alta in cui c'è effettivamente un rischio di esplosione. La pressione bassa si verifica quando c'è una perdita e l'unico danno che si può fare è imbrattare gli stampi o il pavimento.

Questi serbatoi hanno una capienza di 200l e il peso del materiale al loro interno non influisce sulla portata ma se ci fosse stato un serbatoio da 30 000l sarebbe stato necessario introdurlo nei calcoli perché non più trascurabile. Il peso specifico γ del materiale fornito dal cliente e la portata massica \dot{m} voluta, permettono di calcolare il numero di giri al minuto che deve compiere il motore secondo questa formula:

$$RPM = \frac{\dot{m}_{iso} * i * 60}{cc * \gamma_{iso}}$$

Dove cc è la cilindrata della pompa, i il rapporto di riduzione tra motore e pompa (in questo caso 1/10) e 60 è un moltiplicatore per trasformare i giri al secondo in RPM. Analogamente si calcolano i

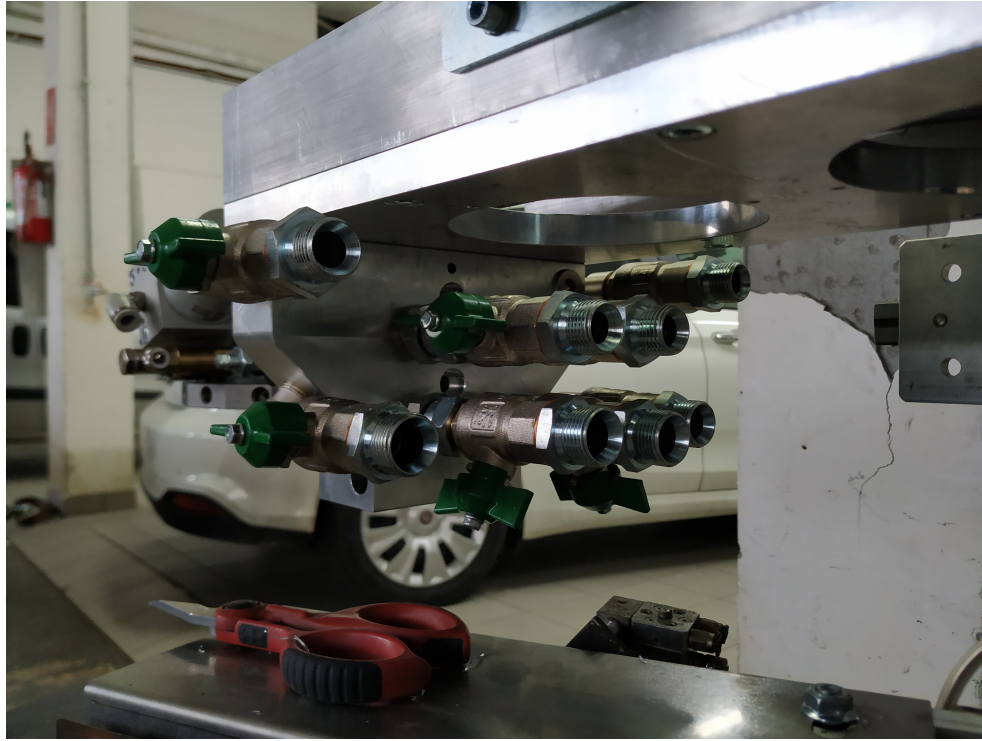


Figura 14: Bocchettoni di ricircolo per tubi di ISO e POL

giri al minuto per il POL.

ISO e POL vengono sempre considerati in quest'ordine: se ad esempio il rapporto dei materiali è 2:1 si conoscono già tutte le informazioni, sono due parti di ISO e una parte di POL. Poi si calcola semplicemente la portata totale:

$$\dot{m}_{tot} = \dot{m}_{iso} + \dot{m}_{pol}$$

Infine, una volta che l'operatore ha inserito il peso della suola, si quantifica il tempo che la valvola di figura 13 deve rimanere aperta:

$$t_c = \frac{P}{\dot{m}_{tot}}$$

Per i colori non ci sono tutte queste problematiche ma il ricircolo si fa lo stesso in modo che il dosaggio sia più preciso. Il peso del colore (ricavato dalla percentuale impostata dall'operatore), viene sottratto dal peso di colata in modo che il peso finale della suola venga rispettato.

1.4 Funzionamento

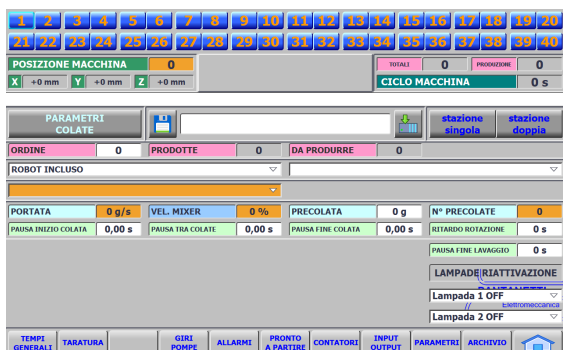
L'operatore del macchinario ha a disposizione la pulsantiera di figura 15 per realizzare la totalità delle operazioni. Oltre ai pulsanti di start e stop ciclo, ci sono i pulsanti per avviare le missioni del robot, la scelta del mix di poliuretano e un interruttore a chiave per entrare nella gabbia e permettere al manutentore di fare la sostituzione dell'ugello di colata.



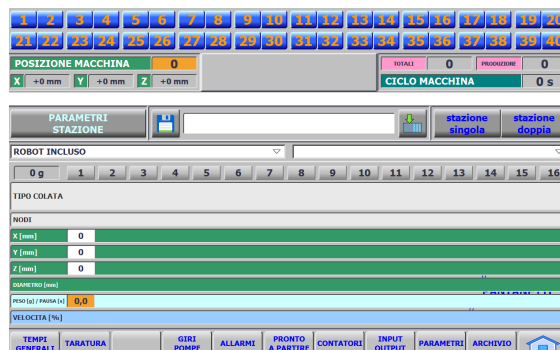
Figura 15: Pulsantiera di controllo

1.4.1 Ricette e cambio mix

Una ricetta, esattamente a come accade in cucina, è una sequenza di procedimenti e ingredienti che si segue con l'intenzione di arrivare ogni volta allo stesso risultato, che sia una pietanza oppure come in questo caso la suola di una scarpa. Ciascuna stazione della macchina rotativa ha la propria ricetta; anche il solo fatto di riempire uno stampo con il numero 42 invece di un numero 39, ha bisogno del ritocco di alcuni parametri. Dall'interfaccia grafica del pannello si possono modificare e salvare: peso del materiale, portata $[g/s]$, velocità del robot, punto iniziale e percorso di colata. La percentuale di colore e la scelta del tipo di mix si preferisce deciderli a priori, anche se capita spesso di dover passare dal mix "morbido" al mix "duro" (o viceversa) durante la produzione.



(a) Parametri stazione



(b) Parametri colata

Figura 16: Impostazioni di una singola stazione

In tal caso, tutte le volte che si cambia mix, si esegue una operazione di lavaggio che consiste nel pulire la camera di colata con un solvente. Ci si posiziona sopra un sacco vicino al macchinario e una volta spruzzato all'interno della camera con la centrifuga in movimento, il materiale rimanente scivola nel sacco insieme al solvente che in seguito verrà riciclato.

1.4.2 Taratura del poliuretano

- Taratura su sacco: un operatore entra nella gabbia di sicurezza per aprire la camera di miscelazione, riempie due bicchierini con rispettivamente ISO e POL e deve verificare che il loro peso sia identico a quello inserito nelle impostazioni dei materiali. Se diverso potrebbero esserci problemi con la portata, motori delle pompe antiquati, densità del materiale non esatta, ostruzioni, ecc. Si procede con la modifica di un moltiplicatore sul touch panel.
- Taratura su bicchiere: viene riempito un bicchiere di carta per verificare il peso del mix e si esegue una prova di resistenza alla compressione, poi il bicchiere si taglia verticalmente a metà per fare un ultimo esame visivo delle bolle formatesi nel processo. Consente di controllare anche errori di portata. Dopo la messa in servizio della macchina si controlla a campione (a discrezione del cliente, in genere ogni turno di lavoro) attraverso un apposito sportello sulla gabbia di protezione.

La taratura del colore invece non è molto importante perché quello che conta è l'effetto visivo finale, si esegue soltanto se il tubo si è otturato per qualche motivo.

2 Robot antropomorfo 6 assi NJ-60-2.2

I robot differiscono tra loro sia per la struttura fisica sia per le loro prestazioni e controllo. Le principali caratteristiche che determinano le diverse tipologie di bracci industriali sono i gradi di libertà, la ripetibilità, la capacità di carico al polso (payload) e l'accessibilità (ovvero l'area di lavoro).

Per questa applicazione qualsiasi robot del catalogo Comau ha una ripetibilità sufficiente, quindi possiamo escluderla dai criteri di scelta. Dalle informazioni ricavate al CAD, la testa di colata pesa, compresa di tubi, circa 40kg e non ha bisogno di essere orientata perché deve rimanere parallela al pavimento, una proprietà caratteristica del robot PAL 4 assi. La taglia più piccola di questa tipologia è il PAL-180-3.1, con un payload di 180kg ci si trova ampiamente all'interno della specifica ma con uno spazio disponibile di circa 2 metri e mezzo tra la macchina rotativa e i serbatoi si può affermare che lo sbraccio di 3100mm è esagerato.

Specifiche tecniche NJ-60-2.2

Numero di assi	6
Capacità di carico massima al polso (kg)	60
Carico aggiuntivo sull'avambraccio (kg)	20
Sbraccio orizzontale massimo (mm)	2258
Coppia sull'asse n. 4 (Nm)	221
Coppia sull'asse n. 5 (Nm)	221
Coppia sull'asse n. 6 (Nm)	118
Corsa sull'asse n. 1	+/- 180° (170°/s)
Corsa sull'asse n. 2	-60° / +125° (150°/s)
Corsa sull'asse n. 3	0° / -165° (165°/s)
Corsa sull'asse n. 4	+/- 2700° (265°/s)
Corsa sull'asse n. 5	+/- 123° (250°/s)
Corsa sull'asse n. 6	+/- 2700° (340°/s)
Ripetibilità (mm)	0.06
Flangia di accoppiamento all'attrezzo	ISO 9409 - 1 - A100
Peso del robot (kg)\	645
Grado di protezione	IP65 / IP67
Posizione di fissaggio	Floor / Ceiling / Sloping (max 45°)

Figura 17: Specifiche tecniche del robot NJ60 [4]

Alla fine la decisione è ricaduta sul robot antropomorfo 6 assi NJ-60-2.2 (figura 17), molto meno

ingombrante e nonostante la grande differenza con il PAL sul carico al polso, hanno un costo simile per via dei motori aggiuntivi. Se ci fosse stato più spazio la scelta non sarebbe stata immediata come in questo caso ma non bisogna dimenticare che un robot più grande implica una gabbia di protezione con un perimetro più lungo quindi una spesa economica maggiore. Con il braccio antropomorfo nascono però alcuni inconvenienti. Quando esegue i movimenti utilizza tutti gli assi disponibili (figura 18) provocando rotazioni del colatore che non devono avvenire. Pertanto è necessario eseguire una procedura aggiuntiva: Comau dà la possibilità di bloccare o vincolare gli assi a livello software, limitandone la corsa o sincronizzandoli con altri assi. Si blocca così l'asse 4 e si vincolano l'asse 5 parallelo al terreno e l'asse 6 in direzione della macchina rotativa facendo produrre alla testa di colata solo movimenti di traslazione.

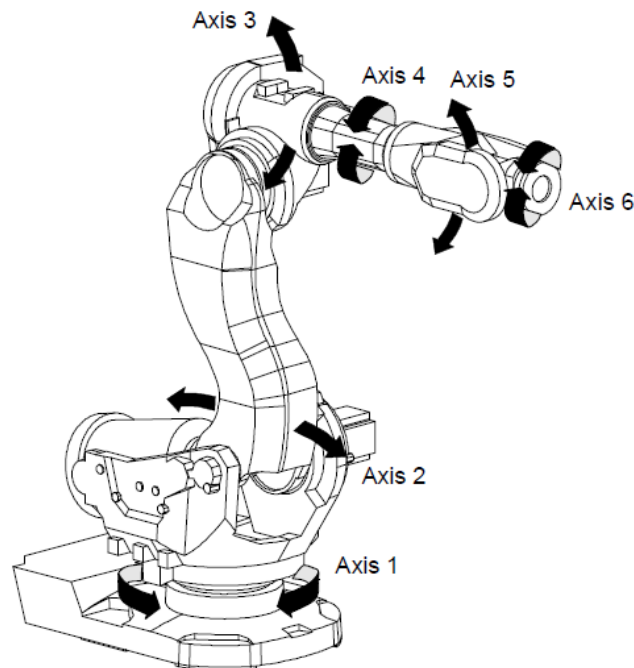


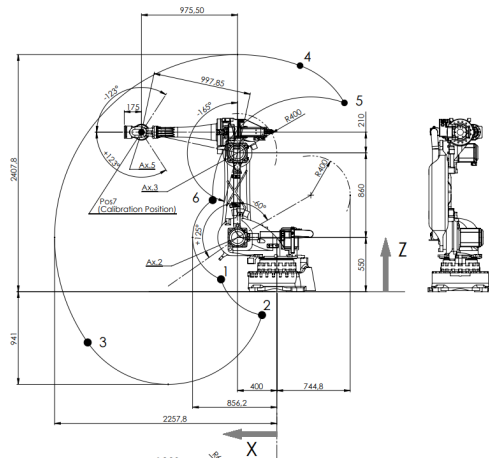
Figura 18: Assi di un generico robot antropomorfo

Il robot viene fornito insieme all'unità di controllo C5GPlus che può interpolare fino a 13 assi (senza accessori)[3] dunque oltre ai 6 dedicati al robot è possibile aggiungere motori a cui serve l'interpolazione con il braccio industriale.

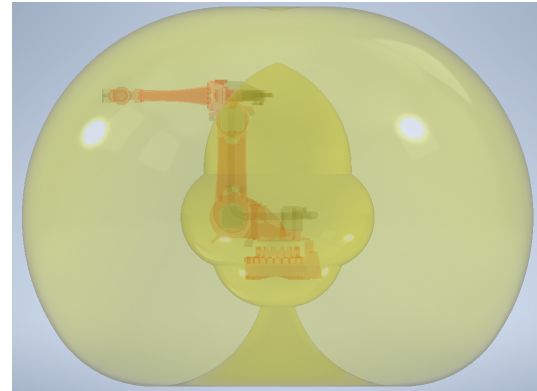
2.1 Cinematica

Per raggiungere un punto nello spazio ad un robot bastano tre gradi di libertà (X, Y, Z). Se però aggiungiamo la possibilità di orientare l'elemento terminale saranno necessari altri tre gradi di libertà ($E1, E2, E3$). Lo spazio raggiunto sarà ugualmente tridimensionale ma sarà possibile raggiungerlo in configurazioni diverse (ridondanza). Le traiettorie o azioni che compie un robot sono il risultato dei movimenti degli elementi articolati che costituiscono la sua struttura. Il sistema di controllo deve fornire a ogni attuatore che muove la rispettiva articolazione gli ordini necessari affinché tutti insieme permettano al robot di compiere le traiettorie desiderate o il posizionamento in un punto dello spazio o un orientamento determinato. per questa ragione è necessario considerare due spazi di lavoro, lo spazio delle configurazioni, le cui coordinate sono le variabili delle articolazioni del robot, e lo spazio cartesiano, definito dalle coordinate X, Y, Z sulle quali si definisce la posizione e l'orientamento dell'elemento terminale in ogni momento. Solitamente, la programmazione di applicazioni si esegue nello spazio cartesiano e ciò permette una facile interpretazione da parte dell'utente sia per porre oggetti nell'ambiente di lavoro sia per definire una traiettoria oppure, in generale, per programmare un compito determinato. Al contrario, il controllo del robot dovrà compiersi per ogni articolazione nello spazio delle configurazioni, permettendo a ogni attuatore di agire nel modo appropriato. Sarà quindi necessario determinare le trasformazioni geometriche che mettono in relazione i due spazi, che saranno funzione della struttura del robot. La funzione matematica che applica la trasformazione necessaria per conoscere la posizione e l'orientamento dell'elemento terminale nello spazio cartesiano partendo dalla posizione di ogni articolazione prende il nome di *trasformazione cinematica diretta*. Tuttavia, il problema di controllare un robot che deve realizzare una traiettoria definita nello spazio cartesiano (coordinate $X, Y, Z, E1, E2, E3$) implica la realizzazione di una *trasformazione cinematica inversa* per ottenere i valori delle variabili articolari ($\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5, \Theta_6$) che saranno i segnali di ingresso ai controllori di ogni attuatore del robot. La complessità algoritmica di questa trasformazione dipenderà dal numero di articolazioni angolari del robot poiché ogni rotazione implica una rotazione degli assi di riferimento dell'articolazione successiva che analiticamente è di maggiore complessità rispetto alla semplice traslazione degli assi di riferimento nel caso delle articolazioni lineari [11].

Tutti i modelli di robot hanno l'area di lavoro (figura 19) al di sotto del piano della base per questo è consueto alzare il robot fissandolo ad un basamento di una certa altezza. Nonostante Comau



(a) Disegno tecnico[4]



(b) Vista laterale al CAD

Figura 19: Area di lavoro SMART-5 NJ60 2.2

definisce il payload del robot NJ60 pari a 60kg in ogni punto della working area, esso varia in ragione della posizione del carico nello spazio, sarà maggiore per traiettorie vicine all'asse centrale rispetto a configurazioni con una maggiore distanza dall'asse.

2.1.1 Sistemi di riferimento

L'unità di controllo C5GPlus ha tre fondamentali variabili di sistema (\$BASE, \$TOOL e \$UFRAME) che consentono di descrivere le trasformazioni di coordinate (figura 20). Se si immagina la terna WORLD come una stanza di officina, \$BASE rappresenta la posizione del robot in quella stanza, \$TOOL descrive il TCP (Tool Center Point) rispetto alla posizione della flangia e \$UFRAME la posizione del pezzo da lavorare rispetto alla terna mondo. Tutte le POSITION che si memorizzano sono riferite al frame ed è quindi necessario realizzarne uno. In questo caso l'unico frame che ci è utile creare è il piano dello stampo. Prima di tutto bisogna definire l'attrezzatura montata sulla flangia cambiando la variabile \$TOOL con tre coordinate di posizione e tre angoli di rotazione, o in altre parole lo spiazamento (la distanza nello spazio) del TCP dal centro della flangia e il suo orientamento. In questo l'utilizzo del CAD facilita di molto il procedimento. Per permettere all'algorithmo di migliorare il movimento è fondamentale definire tutte le variabili del carico:

- \$TOOL_MASS

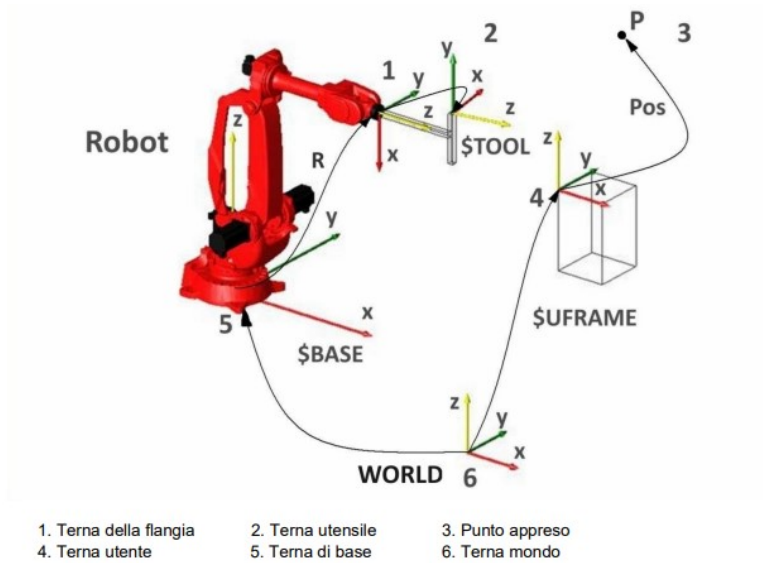


Figura 20: Terne di riferimento del sistema robot [5]

- \$TOOL_CNTR
- \$TOOL_INERTIA

Ora è possibile calcolare il frame utilizzando l'ugello del colatore come TCP. Si utilizza il programmatore portatile TP5 per muovere il robot e memorizzare i tre punti che andranno a definire il piano del frame (figura 21). Una volta creato si verifica che il robot raggiunga tutti i punti controllando che non

```
PROGRAM setframe
VAR corner, x, xy : POSITION
BEGIN
    $UFRAME := POS(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)
    $TOOL := ... -- correttamente definito
    -- Muovere il TCP su tre punti del pezzo mediante i tasti
    -- di jog e memorizzare le POSITION corner, x e xy premendo
    -- il tasto MOD del TP.
    -- Quindi eseguire la seguente istruzione.
    $UFRAME := POS_FRAME(corner, x, xy)
END setframe
```

Figura 21: Passi per la creazione del frame [5]

passi in punti di singolarità o che semplicemente non faccia movimenti insoliti.

2.2 Programmazione

La traiettoria rappresenta un movimento del braccio da una posizione iniziale a una posizione finale. L'istruzione di movimento è MOVE a cui si possono associare le parole riservate dei tre tipi di traiettorie possibili:

- traiettoria articolare: JOINT
- traiettoria lineare: LINEAR
- traiettoria circolare: CIRCULAR

Nel caso di interpolazione giunti (JOINT), gli angoli di ogni giunto del braccio vengono interpolati linearmente dai valori iniziali a quelli finali. Gli assi si muovono partendo insieme ciascuno dalla propria posizione iniziale e arrivando insieme ciascuno alla relativa posizione finale. Il percorso seguito dal TCP, sebbene ripetibile, non è prevedibile [5]. Si può impostare la variabile predefinita `$JNT_MTURN` a FALSE per percorrere la strada minima (cioè minore di 180°). I movimenti fra due posizioni mediante interpolazione giunti sono sempre possibili. Le altre due traiettorie sono calcolate nello spazio cartesiano. Nella interpolazione lineare il TCP si sposta lungo una linea retta dalla posizione iniziale a quella finale. Invece nella interpolazione circolare il centro dell'utensile segue una traiettoria ad arco dalla posizione iniziale a quella finale. Per definire l'arco occorre specificare una posizione aggiuntiva, la posizione di VIA. Se non viene specificato nulla, MOVE TO utilizzerà il valore della variabile predefinita (`$MOVE_TYPE` (di default JOINT) [6]. Oltre alle parole riservate per definire le traiettorie si possono aggiungere alla fine dell'istruzione alcune clausole:

- ADVANCE - fa in modo che l'interprete continui l'esecuzione del programma appena terminati i calcoli relativi alla traiettoria, pochi millisecondi dopo l'inizio del moto;
- WITH - può essere usata per specificare altre caratteristiche del moto come velocità, condizioni di arresto e gestori di condizione;
- BY - clausola utilizzata per avvicinare la posizione in ZTool, distanza misurata lungo il versore negativo di approccio della terna utensile;
- AWAY - come la clausola precedente ma per allontanarsi dalla posizione in ZTool.

La clausola BY viene usata ad esempio nel programma del lavaggio (appendice A.4) dove il colatore provoca qualche schizzo e quindi il robot deve scendere all'interno del sacco di un paio di centimetri senza correre il rischio di andare ad urtare i bordi del cestino. Ogni task eseguito dal robot può essere ricondotto ad un diagramma a blocchi generico mostrato nell'appendice A.1. Durante la missione, un allarme, uno stop o l'emergenza fanno fermare il robot in modi differenti: con l'emergenza si blocca immediatamente tutto il macchinario compreso il braccio antropomorfo (tranne le pompe materiale); con lo stop invece il macchinario si ferma ma il robot finisce l'istruzione che sta eseguendo spegnendo il colatore; con il segnale di allarme, dipende tutto dal livello di gravità. Se è dannoso come l'allarme provocato dalla pressione alta di una pompa, il task si interrompe e il robot raggiunge la posizione di fuori ingombro.

Per il percorso della suola si possono potenzialmente aggiungere fino a 30 punti, non che ne siano necessari così tanti, anzi è raro arrivare ad utilizzarne 10, ma si è scelto di rendere il programma scalabile per future applicazioni. Al cliente questo numero è limitato dall'interfaccia grafica del pannello che permette di aggiungere al massimo 16 punti.

2.2.1 Comunicazione tra PLC e robot

Quando si compra un nuovo robot Comau, la sua scheda di comunicazione Profinet permette di inviare e ricevere di default 32+32 Byte per ogni tempo di ciclo. Quindi la prima operazione che si compie è predisporre la scheda per ricevere e trasmettere 64+64 byte, il massimo consentito. Per questo progetto, il telegramma utilizzato (la combinazione di variabili per "riempire" i 64 Byte) comprende:

- 16 booleani di sistema, $\$DIN[1] \dots \$DIN[16]$
- 16 booleani utente, $\$DIN[1001] \dots \$DIN[1016]$
- 30 interi utente, $\$FMI[1] \dots \$FMI[30]$

In alcuni linguaggi di programmazione un intero corrisponde ad una occupazione di memoria di 4 Byte ma non qui: nel PLC un tipo di dato int equivale a 2 Byte perciò si hanno 60 Byte di interi e 4 Byte di booleani.

Nonostante ciò, i punti di colata non riescono ad entrare tutti nel telegramma. Sono definiti da tre interi ciascuno (X, Y, Z)(eventualmente anche di un intero per il diametro, nel caso di un percorso

circolare) e sono diversi per ogni stazione. Per questo motivo si è deciso di utilizzare un multiplexer, ovvero si trasmette al robot anche un indice che aumenta di uno ad ogni messaggio inviato: per ciascun valore di questo puntatore il PLC invierà un punto differente finché non ci saranno più punti da mandare. Questa lettura di punti avviene allo start ciclo e quando la macchina esegue il passo per andare alla stazione successiva, impiegando al massimo 1 secondo. Nell'appendice B si osservano i tipi di dati passati dal multiplexer (B.2) e l'invio e la ricezione dei telegrammi, scritti e letti dalla memoria con un indice assoluto (B.3).

2.3 Preparazione e sostituzione

Al momento dell'arrivo del robot in azienda, si scarica dal furgone per mezzo del carroponte e si avvita la base a dei fori nel pavimento fatti precedentemente in quella che viene chiamata "test area" (figura 22). Realizzato il cablaggio e la configurazione software iniziale si cercano di emulare le posizioni degli

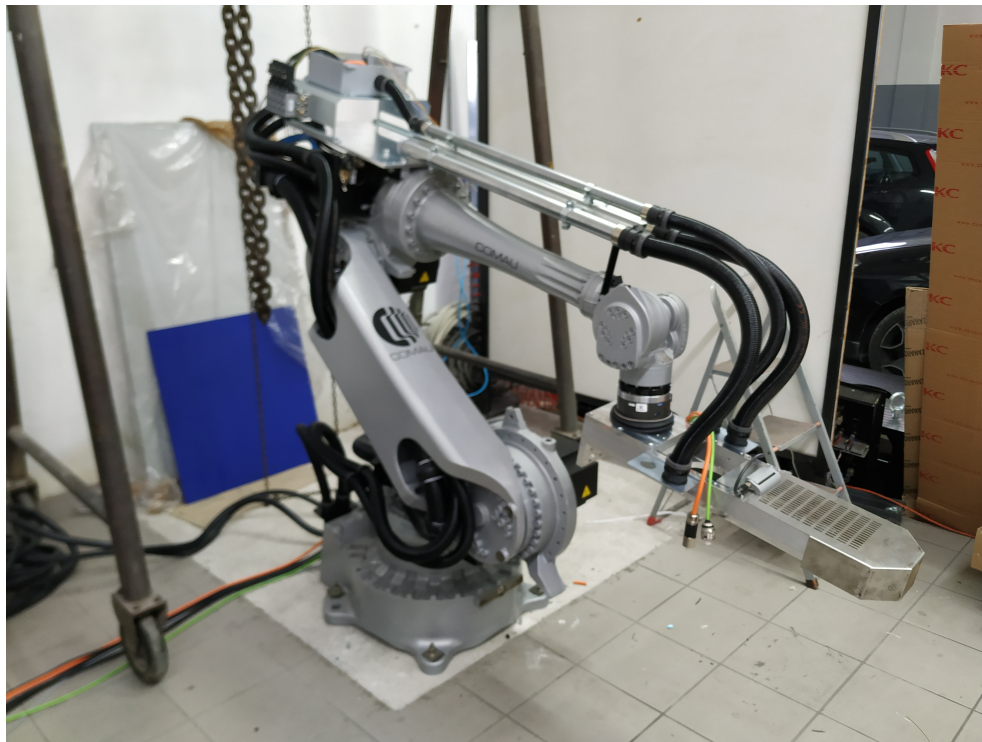


Figura 22: Robot in azienda per prove

oggetti come disposti nell'azienda di destinazione per poi eseguire il programma PUNTI.COD. Questo

insieme di istruzioni sono comandi di movimento consecutivi tra tutte le posizioni che utilizzerà il programma robot definitivo. Si fa partire l'algoritmo in modalità manuale dal dispositivo TP5, in modo che ad ogni riga di codice si verifica che il movimento del braccio sia corretto ed eventualmente si riposiziona il punto nello spazio. Fino a questo momento il robot ha operato soltanto con la sua unità di controllo senza il quadro elettrico della macchina che viene assemblato parallelamente durante il periodo delle prove. Una volta che il quadro viene finito si realizza un collaudo generale compreso di robot e ci si mette d'accordo con il cliente per la messa in servizio.

Il braccio antropomorfo viene disposto in posizione di trasporto (viene avviata una procedura che mette i giunti ad un certo angolo) e caricato sul furgone insieme alla gabbia di protezione, quadro elettrico, pompe materiale e tutti i cavi. Il tempo di montaggio dal cliente si è protratto per circa 4 giorni compresi gli aggiustamenti del software e una nuova esecuzione del programma PUNTI.COD per memorizzare le posizioni reali dei punti.



Figura 23: Robot nella sua posizione definitiva

3 Sicurezza in generale

Al giorno d'oggi la sicurezza è parte integrante di ogni lavoro. Un macchinario ben ideato e costruito potrebbe seriamente limitare danni a oggetti e persone. Ma una buona progettazione non riguarda soltanto la coscienza e il buonsenso di chi lo consegna: esistono delle norme in continuo aggiornamento che riguardano la costruzione di macchine industriali e a cui ci si affida, anche se è spiacevole riportarlo, per evitare problemi legali in caso di incidenti. Una prima distinzione da fare è tra pericolo e rischio. Mentre il pericolo è qualcosa di assoluto, il rischio è "calcolabile" ed è sempre riferito all'applicazione reale. La macchina deve essere sicura quando viene utilizzata secondo quello che il produttore ha previsto come comportamento corretto. Se una persona valica la gabbia di protezione di un robot aggirando le emergenze non deve essere considerata nella valutazione del rischio perché si tratta di un comportamento non ragionevolmente prevedibile.

Severità del Danno		Classe di Probabilità				
		Molto Improbabile 3-4	Improbabile 5-7	Possibile '8-10	Probabile 11-13	Molto Probabile 14-15
Morte o Disabilità (>10%) Permanente	4	Basso	Elevato	Grave	Grave	Grave
Danno Irreversibile Servono > 6 mesi per rimettersi	3	Basso	Medio	Elevato	Grave	Grave
Danno Reversibile Servono < 6 mesi per rimettersi	2	Basso	Basso	Medio	Grave	Grave
Danno Minore Perdita al massimo di 2 giorni di lavoro	1	Basso	Basso	Basso	Medio	Elevato

Figura 24: Esempio di matrice di rischio [8]

Dopo aver individuato i pericoli e i limiti della macchina, si forma una tabella a zone (figura 24) che definisce il rischio inaccettabile (rosso e arancione) con riduzione del rischio obbligatoria, e rischio accettabile (verde). Questa matrice fa parte delle norme EN ISO 12100 [8] che a loro volta sono una categoria (tipo A) delle norme armonizzate (oltre 800 norme tecniche) alla Direttiva Macchine [9]. Perdersi nell'universo delle norme è semplice, per questo motivo chiedere la consulenza di un certificatore esterno fa parte della prassi. Nell'ordine di far seguire le normative, egli propone modifiche del macchinario già nella fase iniziale di progettazione e a realizzazione conclusa, redige il fascicolo tecnico.

3.1 S.T.O. - Safe Torque Off e sicurezza del robot

Lo STO è una fermata certificata, ovvero si certifica che il motore una volta fermo non può accidentalmente ripartire senza prima aver premuto il pulsante di inserzione comandi. IEC 60204-1 [14] distingue tre categorie di stop (figura 25):

- categoria 0 - arresto mediante rimozione immediata dell'alimentazione degli azionamenti (fermata incontrollata);
- categoria 1 - un arresto controllato con l'energia disponibile nell'azionamento e poi rimozione dell'alimentazione a motore fermo;
- categoria 2 - un arresto controllato con l'energia rimanente disponibile nell'azionamento.

Quando disponibile, si preferisce sempre eseguire lo stop di tipo 1, altrimenti si realizza uno stop di tipo 0, replicabile anche interrompendo elettromeccanicamente i cavi tra il motore e l'azionamento.

Alcune volte con motori che spostano grandi inerzie meglio non applicare immediatamente lo STO ma piuttosto arrivare ad una velocità tale da poter comunque rimuovere in sicurezza l'alimentazione dell'azionamento.

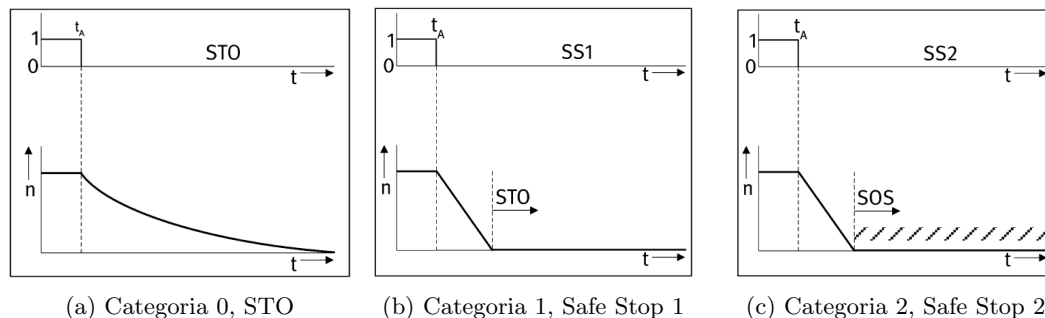


Figura 25: Categorie di stop [14]

Il braccio antropomorfo possiede dei freni in ogni giunto che si attivano quando si realizza uno STO, fermandolo rapidamente. EN ISO 10218-1 sono le norme armonizzate di tipo C che riguardano i requisiti di sicurezza dei robot industriali [7]. La gabbia di protezione del robot dovrebbe essere almeno grande quanto l'area di accessibilità formata dal braccio completamente disteso e compreso di utensile. Se come in questo caso lo spazio operativo è maggiore dello spazio protetto, secondo la normativa deve

essere fornito un mezzo per stabilire uno “spazio limitato” intorno al robot utilizzando dei dispositivi di limitazione che possono essere meccanici, elettromeccanici oppure eseguite da software. La soluzione da prediligere è utilizzare un software certificato, tuttavia questo comporta dei costi aggiuntivi (di licenza) che si valutano insieme al cliente, spesso incline a considerare soprattutto i dispositivi più economici. Pertanto in questo progetto sono stati impiegati dei blocchi meccanici sugli assi (più specificamente sui tre assi con il raggio di movimentazione maggiore, come da norma), che restringono in modo “fisico” e “brutale” lo spazio di movimentazione del robot. Durante il normale funzionamento, il robot non andrà a sbattere sui blocchi meccanici ma questo accadrà al verificarsi di un’avarìa.

Per i robot Comau esiste una prestazione opzionale chiamata Collision Detection [5]. Questo algoritmo fornisce al sistema la capacità di arrestare il movimento del braccio, nel momento in cui entrino in azione forze di disturbo sui giunti. Un lavoratore non riesce a provocare un disturbo abbastanza grande da innescarlo, infatti Comau attesta che nasce per limitare danni, in caso di urto, soltanto alla meccanica del braccio. Per far funzionare bene la Collision Detection bisogna definire adeguatamente il carico al polso come spiegato nel capitolo 2.1.1.

L’ultimo sistema di sicurezza applicato ma non imposto dalle norme è un cilindro pneumatico anticollisione. Questo dispositivo viene inserito come congiunzione tra la testa di colata e il polso del robot e fornisce una protezione agli urti e ai sovraccarichi dell’elemento terminale.

3.2 Accorgimenti nella costruzione

Tutte le volte che sono presenti pulsanti con la funzione di stop in un impianto è consuetudine utilizzare contatti normalmente chiusi (NC) in modo che in caso di un falso contatto o un’interruzione del cavo, ci si accorga della rottura. Per i funghi di emergenza invece si seguono le norme UNI EN ISO 13850 [10]. Si utilizza un doppio contatto NC per ogni pulsante, poi cablato ad un dispositivo safety, come un relè di sicurezza o un PLC failsafe. Le emergenze in sé non sono una funzione di sicurezza e non riducono il rischio. In altri termini non è possibile decidere che il rischio è accettabile se sono presenti, intorno ad una macchina pericolosa, più funghi di emergenza.

A ridosso della macchina rotativa, la gabbia di protezione del robot possiede, sostenuti da cerniere di sicurezza, dei pannelli in plexiglas che si aprono nel verso di rotazione della giostra. Se per colpa di un guasto uno stampo rimanesse aperto o se un dipendente si incastrasse, all’apertura degli sportelli

il macchinario va in allarme, bloccando la rotazione e il robot. Ci si trova al limite della certificazione, infatti avere protezioni tra macchina e robot significherebbe che la tavola rotante non è sicura e che se dovesse venire certificata probabilmente non verrebbe approvata.

Un'altra cerniera di sicurezza è installata sul coperchio della testa di colata (figura 11) e serve per impedire alla centrifuga di partire durante le operazioni di taratura. Sebbene ci siano altre protezioni per questa procedura, si è deciso di conferire al manutentore anche una certezza "visiva" di essere al sicuro.

4 Conclusioni

Poco prima della messa in servizio, per una esigenza del cliente, la gabbia di protezione è stata spostata di circa 30cm alla destra del robot con la conseguenza di dover accorciare anche il cassetto della taratura in sicurezza. Dopo questa manovra, il certificatore ha fortemente consigliato di aggiungere una parte in più di intelaiatura nell'ampio spazio che si era creato tra la tavola rotante e cancello, in modo da evitare che qualcuno potesse entrare nell'area di lavoro del robot accovacciandosi, ed eludere così le sicurezze. Sempre per altre esigenze del cliente, è stato tolto il cilindro pneumatico anticollisione, comunque non essenziale per la certificazione.

La precisione raggiunta con il braccio antropomorfo era stata ampiamente anticipata. Considerando il fatto che la costruzione dei due robot sono avvenute in periodi di tempo distanti di quasi cinquant'anni, le premesse sulla ripetibilità del distributore ad un asse erano a dir poco disastrose. Ora, bisognerebbe fare un confronto tra le produzioni di soles del robot pneumatico con quelle del nuovo robot appena sostituito, come il numero di soles scartate o la velocità di produzione, dati che senza dubbio dimostrerebbero i risultati positivi ottenuti. Purtroppo, il cliente non ha fornito questi dati, si può però fare affidamento sul fatto che questi robot utilizzati per la colata sono richiesti continuamente sia a livello locale che all'estero, che sia la stessa azienda che ne richiede un altro o di nuove aziende che per passaparola, ne sono venute a conoscenza.

4.1 Possibili sviluppi

Alcune aziende che producono macchine del poliuretano stanno progettando tavole rotanti con il cambio stampi automatizzato. L'unico addetto alla macchina carica e scarica gli stampi da uno scaffale a ridosso della gabbia di protezione, comunicante con il robot antropomorfo. Il numero di robot è due: uno per l'estrazione e l'inserimento degli stampi sulla tavola, ed un altro per la colata delle soles.

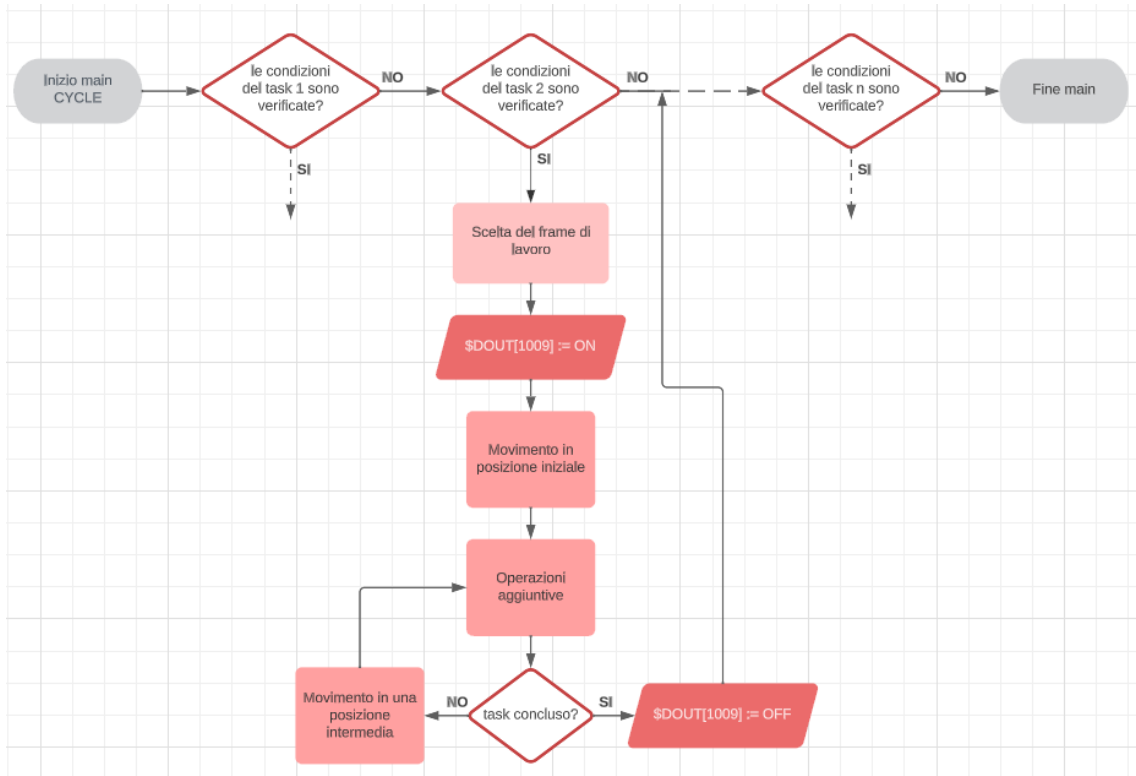
La maggior parte degli sviluppi però, avverranno dal punto di vista della sicurezza. In questo momento si sta iniziando ad utilizzare lo standard OPC UA per l'industria 4.0. Con l'OPC UA si semplificheranno gli scambi di dati tra PLC, interfacce utente, server, client e altri macchinari. Per quanto riguarda i robot, la maggior parte delle applicazioni richiederà una forma di certificazione che nel caso di Comau è il Comau RoboSAFE (una funzione aggiuntiva con licenza). All'accensione e dopo una certa quantità di ore di lavoro il robot esegue automaticamente la procedura di test dei freni, con

accelerazioni e decelerazioni, In questo modo si certifica il corretto funzionamento e che non sia stato manomesso niente mentre il braccio industriale era spento.

Una macchina industriale che oggi è considerata sicura, in futuro potrebbe non esserlo più: la tendenza che si percepisce è raggiungere un livello esasperato di sicurezza tale da comprendere nella valutazione del rischio anche comportamenti non ragionevolmente prevedibili (con costi intollerabili).L'auspicio di molti è che esista un limite alle normative oltre il quale prevalga il buonsenso perché il vero pericolo sono i vecchi macchinari ancora in funzione, che se dovessero venire nuovamente certificati andrebbero ricostruiti da zero.

Appendice A Programmazione del robot

A.1 Diagramma di flusso di un task generico



Per avviare un task si verificano un insieme di valori booleani. Alcuni hanno la funzione di semaforo, ad esempio per eseguire il task di precolata bisogna prima aver effettuato il task di lavaggio, mentre altri sono bit di start che possono provenire dalla pulsantiera oppure vengono direttamente generati dal software. Una volta all'interno del task si avverte il PLC che la missione è iniziata mettendo il bit \$DOUT[1009] a 1 per poi azzerarlo non appena si conclude.

A.2 Main

```
1 PROGRAM pr_MAIN STACK = 2048
2 VAR
3
4 ROUTINE a_tool(ai_t_idx : INTEGER) EXPORTED FROM tt_tool
5 ROUTINE call_main EXPORTED FROM pr_MAIN
6 ROUTINE condizioni EXPORTED FROM lib_ini
7 ROUTINE inizializzazione EXPORTED FROM lib_ini
8 ROUTINE esegui_movimento EXPORTED FROM pr_mov
9
10 ROUTINE pos_primo_punto EXPORTED FROM pr_primo
11 ROUTINE pos_lavaggio EXPORTED FROM pr_lav
12 ROUTINE pos_precolata EXPORTED FROM pr_pre
13 ROUTINE pos_cambio_mixer EXPORTED FROM pr_cmi
14 ROUTINE pos_fuori_ingombro EXPORTED FROM pr_fuori
15 ROUTINE allarme(numero_allarme : INTEGER) EXPORTED FROM lib_ini
16 ROUTINE pos_taratura EXPORTED FROM pr_tar
17 ROUTINE pos_taratura_in_sicurezza EXPORTED FROM PR_TAR_SICURA
18
19 ROUTINE call_main
20 BEGIN
21
22 ----- Ricevo %Velocita' da PLC
23 $WORD[4] := $FMI[4]
24 IF ($WORD[4] <= 0) OR ($WORD[4] > 100) THEN
25     $WORD[4] := 20 -- Imposto Velocita' di Default
26 ENDIF
27
28 ----- Ricevo %Velocita' tra Colate da PLC
29 $WORD[1] := $FMI[1]
30 IF ($WORD[1] <= 0) OR ($WORD[2] > 100) THEN
31     $WORD[1] := 20 -- Imposto Velocita' di Default
32 ENDIF
33
34 IF NOT $FDIN[9] THEN --- Controllo chiave in Aut.
35     $WORD[4] := 100
```

```

36  ENDIF
37
38  $ARM_OVR := $WORD[4]  -- %Vel. Spostamento da PLC
39
40  $DOUT[1009] := OFF  -- missione in corso
41
42  IF $DIN[1006] = ON THEN  -- Programma INIZIO COLATA
43    IF ($FMO[4] = 4) OR ($FMO[4] = 5) THEN  -- codice ultima missione
44      $DOUT[1002] := OFF  -- Reset Programma FI
45      $DOUT[1003] := OFF  -- Reset Programma Posizione PRECOLATA
46      $DOUT[1004] := OFF  -- Reset Programma Posizione LAVAGGIO
47      $DOUT[1005] := OFF  -- Reset Programma Posizione CAMBIO MIXER
48      $DOUT[1006] := OFF  -- Reset Programma Posizione INIZIO COLATA
49      $DOUT[1007] := OFF  -- Reset Programma Posizione FINE PROGRAMMA COLATA
50      $DOUT[1008] := OFF  -- OK per Peso Parziale
51      $DOUT[1013] := OFF  -- Reset Programma Posizione TARATURA
52      $DOUT[1014] := OFF  -- Reset Programma TARATURA IN SICUREZZA
53      BIT_ASSIGN($FMO[19], 1, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA
54      BIT_ASSIGN($FMO[19], 2, FALSE)  --POSIZIONE TARATURA RAGGIUNTA
55      BIT_ASSIGN($FMO[19], 3, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
56      BIT_ASSIGN($FMO[19], 4, FALSE)  --POSIZIONE LAVAGGIO RAGGIUNTA
57      $FMO[4] := 1  -- Codice Programma in corso
58      $FMO[5] := 0  -- Numero punto fatto
59      pos_primo_punto
60      $DOUT[1006] := ON  -- Programma INIZIO COLATA Terminato
61    ELSE
62      allarme(4027)  -- Missione non Possibile
63    ENDIF
64  ENDIF
65
66  IF $DIN[1007] = ON THEN  -- Programma COLATA
67    IF ($FMO[4] = 1) OR ($FMO[4] = 2) THEN  -- codice ultima missione
68      IF ($FMI[5] >= 1) AND ($FMI[5] <= 30) THEN  -- controllo numero punti
69        IF $FMI[5] > $FMO[5] THEN  -- esegui solo se il contatore punti e' inferiore
70          $DOUT[1002] := OFF  -- Reset Programma FI
71          $DOUT[1003] := OFF  -- Reset Programma Posizione PRECOLATA

```

```

72      $DOUT[1004] := OFF    -- Reset Programma Posizione LAVAGGIO
73      $DOUT[1005] := OFF    -- Reset Programma Posizione CAMBIO MIXER
74      $DOUT[1006] := OFF    -- Reset Programma Posizione INIZIO COLATA
75      $DOUT[1007] := OFF    -- Reset Programma Posizione FINE PROGRAMMA COLATA
76      $DOUT[1008] := OFF    -- OK per Peso Parziale
77      $DOUT[1013] := OFF    -- Reset Programma Posizione TARATURA
78      $DOUT[1014] := OFF    -- Reset Programma TARATURA IN SICUREZZA
79      BIT_ASSIGN($FMO[19], 1, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA
80      BIT_ASSIGN($FMO[19], 2, FALSE)  --POSIZIONE TARATURA RAGGIUNTA
81      BIT_ASSIGN($FMO[19], 3, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
82      BIT_ASSIGN($FMO[19], 4, FALSE)  --POSIZIONE LAVAGGIO RAGGIUNTA
83      $FMO[4] := 2    -- Codice Programma in corso
84      esegui_movimento
85      $DOUT[1007] := ON    -- Programma COLATA Terminato
86      ENDIF
87      ELSE
88          allarme(4026)    -- Numero Punti Errati
89      ENDIF
90      ELSE
91          allarme(4027)    -- Missione non Possibile
92      ENDIF
93  ENDIF
94
95  IF $DIN[1004] = ON THEN    -- Programma LAVAGGIO
96      IF ($FMO[4] = 3) OR ($FMO[4] = 4) OR ($FMO[4] = 5) OR ($FMO[4] = 6) OR ($FMO[4] =
97          7) OR ($FMO[4] = 9) THEN    -- codice ultima missione
98          $DOUT[1002] := OFF    -- Reset Programma FI
99          $DOUT[1003] := OFF    -- Reset Programma Posizione PRECOLATA
100         $DOUT[1004] := OFF    -- Reset Programma Posizione LAVAGGIO
101         $DOUT[1005] := OFF    -- Reset Programma Posizione CAMBIO MIXER
102         $DOUT[1006] := OFF    -- Reset Programma Posizione INIZIO COLATA
103         $DOUT[1007] := OFF    -- Reset Programma Posizione FINE PROGRAMMA COLATA
104         $DOUT[1008] := OFF    -- OK per Peso Parziale
105         $DOUT[1013] := OFF    -- Reset Programma Posizione TARATURA
106         $DOUT[1014] := OFF    -- Reset Programma TARATURA IN SICUREZZA
107         BIT_ASSIGN($FMO[19], 1, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA
108         BIT_ASSIGN($FMO[19], 2, FALSE)  --POSIZIONE TARATURA RAGGIUNTA

```

```

108     BIT_ASSIGN($FMO[19], 3, FALSE)    --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
109     BIT_ASSIGN($FMO[19], 4, FALSE)    --POSIZIONE LAVAGGIO RAGGIUNTA
110     $FMO[4] := 3    -- Codice Programma in corso
111     pos_lavaggio
112     $DOUT[1004] := ON    -- Programma LAVAGGIO Terminato
113     ELSE
114         allarme(4027)    -- Missione non Possibile
115     ENDIF
116 ENDIF
117
118 IF $DIN[1003] = ON THEN    -- Programma PRECOLATA
119     IF ($FMO[4] = 3) OR ($FMO[4] = 4) OR ($FMO[4] = 5) OR ($FMO[4] = 6) OR ($FMO[4] =
120         7) OR ($FMO[4] = 9) THEN    -- codice ultima missione
121         $DOUT[1002] := OFF    -- Reset Programma FI
122         $DOUT[1003] := OFF    -- Reset Programma Posizione PRECOLATA
123         $DOUT[1004] := OFF    -- Reset Programma Posizione LAVAGGIO
124         $DOUT[1005] := OFF    -- Reset Programma Posizione CAMBIO MIXER
125         $DOUT[1006] := OFF    -- Reset Programma Posizione INIZIO COLATA
126         $DOUT[1007] := OFF    -- Reset Programma Posizione FINE PROGRAMMA COLATA
127         $DOUT[1008] := OFF    -- OK per Peso Parziale
128         $DOUT[1013] := OFF    -- Reset Programma Posizione TARATURA
129         $DOUT[1014] := OFF    -- Reset Programma TARATURA IN SICUREZZA
130         BIT_ASSIGN($FMO[19], 1, FALSE)    --POSIZIONE PRECOLATA RAGGIUNTA
131         BIT_ASSIGN($FMO[19], 2, FALSE)    --POSIZIONE TARATURA RAGGIUNTA
132         BIT_ASSIGN($FMO[19], 3, FALSE)    --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
133         BIT_ASSIGN($FMO[19], 4, FALSE)    --POSIZIONE LAVAGGIO RAGGIUNTA
134         $FMO[4] := 4    -- Codice Programma in corso
135         pos_precolata
136         $DOUT[1003] := ON    -- Programma PRECOLATA Terminato
137     ELSE
138         allarme(4027)    -- Missione non Possibile
139     ENDIF
140 ENDIF
141
142 IF $DIN[1002] = ON THEN    -- Programma FI
143     $DOUT[1002] := OFF    -- Reset Programma FI
144     $DOUT[1003] := OFF    -- Reset Programma Posizione PRECOLATA

```

```

144   $DOUT[1004] := OFF    -- Reset Programma Posizione LAVAGGIO
145   $DOUT[1005] := OFF    -- Reset Programma Posizione CAMBIO MIXER
146   $DOUT[1006] := OFF    -- Reset Programma Posizione INIZIO COLATA
147   $DOUT[1007] := OFF    -- Reset Programma Posizione FINE PROGRAMMA COLATA
148   $DOUT[1008] := OFF    -- OK per Peso Parziale
149   $DOUT[1013] := OFF    -- Reset Programma Posizione TARATURA
150   $DOUT[1014] := OFF    -- Reset Programma TARATURA IN SICUREZZA
151   BIT_ASSIGN($FMO[19], 1, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA
152   BIT_ASSIGN($FMO[19], 2, FALSE)  --POSIZIONE TARATURA RAGGIUNTA
153   BIT_ASSIGN($FMO[19], 3, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
154   BIT_ASSIGN($FMO[19], 4, FALSE)  --POSIZIONE LAVAGGIO RAGGIUNTA
155   $FMO[4] := 5    -- Codice Programma in corso
156   pos_fuori_ingombro
157   $DOUT[1002] := ON    -- Programma FI Terminato
158 ENDIF
159
160 IF $DIN[1005] = ON THEN    -- Programma CAMBIO MIXER
161   IF ($FMO[4] = 3) OR ($FMO[4] = 4) OR ($FMO[4] = 5) OR ($FMO[4] = 6) OR ($FMO[4] =
162     7) OR ($FMO[4] = 9) THEN    -- codice ultima missione
163     $DOUT[1002] := OFF    -- Reset Programma FI
164     $DOUT[1003] := OFF    -- Reset Programma Posizione PRECOLATA
165     $DOUT[1004] := OFF    -- Reset Programma Posizione LAVAGGIO
166     $DOUT[1005] := OFF    -- Reset Programma Posizione CAMBIO MIXER
167     $DOUT[1006] := OFF    -- Reset Programma Posizione INIZIO COLATA
168     $DOUT[1007] := OFF    -- Reset Programma Posizione FINE PROGRAMMA COLATA
169     $DOUT[1008] := OFF    -- OK per Peso Parziale
170     $DOUT[1013] := OFF    -- Reset Programma Posizione TARATURA
171     $DOUT[1014] := OFF    -- Reset Programma TARATURA IN SICUREZZA
172     BIT_ASSIGN($FMO[19], 1, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA
173     BIT_ASSIGN($FMO[19], 2, FALSE)  --POSIZIONE TARATURA RAGGIUNTA
174     BIT_ASSIGN($FMO[19], 3, FALSE)  --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
175     BIT_ASSIGN($FMO[19], 4, FALSE)  --POSIZIONE LAVAGGIO RAGGIUNTA
176     $FMO[4] := 6    -- Codice Programma in corso
177     pos_cambio_mixer
178     $DOUT[1005] := ON    -- Programma CAMBIO MIXER Terminato
179 ELSE
180   allarme(4027)    -- Missione non Possibile

```



```

180     ENDIF
181 ENDIF
182
183 IF $DIN[1013] = ON THEN    -- Programma TARATURA
184     IF ($FMO[4] = 3) OR ($FMO[4] = 5) OR ($FMO[4] = 6) OR ($FMO[4] = 7) OR ($FMO[4] =
185     9) THEN    -- codice ultima missione
186         $DOUT[1002] := OFF    -- Reset Programma FI
187         $DOUT[1003] := OFF    -- Reset Programma Posizione PRECOLATA
188         $DOUT[1004] := OFF    -- Reset Programma Posizione LAVAGGIO
189         $DOUT[1005] := OFF    -- Reset Programma Posizione CAMBIO MIXER
190         $DOUT[1006] := OFF    -- Reset Programma Posizione INIZIO COLATA
191         $DOUT[1007] := OFF    -- Reset Programma Posizione FINE PROGRAMMA COLATA
192         $DOUT[1008] := OFF    -- OK per Peso Parziale
193         $DOUT[1013] := OFF    -- Reset Programma Posizione TARATURA
194         $DOUT[1014] := OFF    -- Reset Programma TARATURA IN SICUREZZA
195         BIT_ASSIGN($FMO[19], 1, FALSE)    --POSIZIONE PRECOLATA RAGGIUNTA
196         BIT_ASSIGN($FMO[19], 2, FALSE)    --POSIZIONE TARATURA RAGGIUNTA
197         BIT_ASSIGN($FMO[19], 3, FALSE)    --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
198         BIT_ASSIGN($FMO[19], 4, FALSE)    --POSIZIONE LAVAGGIO RAGGIUNTA
199         $FMO[4] := 7    -- Codice Programma in corso
200         pos_taratura
201         $DOUT[1013] := ON    -- Programma PRECOLATA Terminato
202     ELSE
203         allarme(4027)    -- Missione non Possibile
204     ENDIF
205 ENDIF
206
207 IF $DIN[1014] = ON THEN    -- Programma TARATURA SICURA
208     IF ($FMO[4] = 3) OR ($FMO[4] = 9) THEN    -- codice ultima missione
209         $DOUT[1002] := OFF    -- Reset Programma FI
210         $DOUT[1003] := OFF    -- Reset Programma Posizione PRECOLATA
211         $DOUT[1004] := OFF    -- Reset Programma Posizione LAVAGGIO
212         $DOUT[1005] := OFF    -- Reset Programma Posizione CAMBIO MIXER
213         $DOUT[1006] := OFF    -- Reset Programma Posizione INIZIO COLATA
214         $DOUT[1007] := OFF    -- Reset Programma Posizione FINE PROGRAMMA COLATA
215         $DOUT[1008] := OFF    -- OK per Peso Parziale
216         $DOUT[1013] := OFF    -- Reset Programma Posizione TARATURA

```

```

216     $DOUT[1014] := OFF    -- Reset Programma TARATURA IN SICUREZZA
217     BIT_ASSIGN($FMO[19], 1, FALSE)    --POSIZIONE PRECOLATA RAGGIUNTA
218     BIT_ASSIGN($FMO[19], 2, FALSE)    --POSIZIONE TARATURA RAGGIUNTA
219     BIT_ASSIGN($FMO[19], 3, FALSE)    --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
220     BIT_ASSIGN($FMO[19], 4, FALSE)    --POSIZIONE LAVAGGIO RAGGIUNTA
221     $FMO[4] := 9    -- Codice Programma in corso
222     pos_taratura_in_sicurezza
223     $DOUT[1014] := ON    -- Programma TARATURA IN SICUREZZA Terminato
224     ELSE
225         allarme(4027)    -- Missione non Possibile
226     ENDIF
227 ENDIF
228
229 END call_main
230
231 BEGIN
232
233     condizioni
234     inizializzazione
235
236 CYCLE
237     call_main
238     DELAY 1    -- Pausa di 1mS
239 END pr_MAIN

```

A.3 Punto e tratto

```

1  ----- PROGRAMMA PUNTO -----
2  ROUTINE programma_punto(numero_punto : INTEGER)
3
4  BEGIN
5
6     $DOUT[1009] := ON    -- missione in corso
7     posizioneTemp := ARM_POS
8
9     posizioneTemp.X := punto[numero_punto].pos.X
10    posizioneTemp.Y := punto[numero_punto].pos.Y
11    posizioneTemp.Z := punto[numero_punto].pos.Z

```

```

12
13 controllo_limite(posizioneTemp)
14
15 $ARM_OVR := $WORD[1] -- %Vel. "TRA COLATE" da PLC per raggiungere la posizione
    del punto
16
17 MOVE JOINT TO (posizioneTemp),
18     WITH CONDITION[1], -- Controllo Colata
19 ENDMOVE
20
21 $DOUT[1008] := ON -- OK per Peso Parziale
22
23 $FMO[5] := numero_punto
24 $DOUT[1009] := OFF -- missione in corso
25
26 WAIT FOR $DIN[1008]+ OR $DIN[1008] OR $DIN[1009] --Aspetto OK dalla colata del
    peso parziale OPPURE ricevo l'OK perche' e' caduto il ciclo
27
28 END programma_punto
29
30 ----- PROGRAMMA PERCORSO -----
31 ROUTINE programma_percorso(numero_punto : INTEGER)
32
33 BEGIN
34
35     posizioneTemp := ARM_POS
36
37     $DOUT[1009] := ON -- missione in corso
38     IF numero_punto <> 1 THEN
39
40         $DOUT[1009] := ON -- missione in corso
41
42         posizioneTemp.X := punto[numero_punto].pos.X
43         posizioneTemp.Y := punto[numero_punto].pos.Y
44         posizioneTemp.Z := punto[numero_punto].pos.Z
45         controllo_limite(posizioneTemp)
46

```

```

47   $ARM_OVR := $WORD[1]  -- %Vel. "TRA COLATE" da PLC per raggiungere la posizione
    inizio percorso
48
49   MOVE JOINT TO (posizioneTemp)
50
51   $DOUT[1009] := OFF  -- missione in corso
52
53   ENDIF
54
55   $DOUT[1008] := ON  -- Posizione OK per Colata (Peso Parziale)
56
57   numero_tratti := punto[numero_punto].nodi - 1
58
59   IF numero_tratti + numero_punto > punti_massimi_impostabili THEN  -- Controllo che
    la lunghezza del percorso non superi i punti massimi
60     -- se la supera forzo la lunghezza fino all'ultimo punto impostabile
61     numero_tratti := punti_massimi_impostabili - numero_punto
62
63   ENDIF
64
65   lunghezza_percorso := numero_tratti + 1
66
67   override_percorso := 100
68
69   IF PATH_LEN(percorsso) > 0 THEN
70     NODE_DEL(percorsso, 1, PATH_LEN(percorsso))
71   ENDIF
72
73   percorsso.COND_TBL[1] := 1
74
75   NODE_APP(percorsso, lunghezza_percorso)
76   FOR idx_posizione := 0 TO numero_tratti DO
77
78     indice_punto := numero_punto + idx_posizione
79     indice_nodo := indice_punto - numero_punto + 1
80
81     posizioneTemp.X := punto[indice_punto].pos.X

```

```

82     posizioneTemp.Y := punto[indice_punto].pos.Y
83     posizioneTemp.Z := punto[indice_punto].pos.Z
84
85     override_percorso := punto[indice_punto].override    -- 100    ----- Ricevo %
      Velocita' da PLC
86     IF override_percorso < 1 THEN
87         override_percorso := 1    -- Imposto Velocita' di MINIMA
88     ENDIF
89     IF override_percorso > 100 THEN
90         override_percorso := 100    -- Imposto Velocita' di MASSIMA
91     ENDIF
92
93     controllo_limite(posizioneTemp)
94     percorso.NODE[indice_nodo].$MAIN_POS := posizioneTemp    --- punto[numero_punto +
      idx_posizione].pos
95     percorso.NODE[indice_nodo].$MOVE_TYPE := LINEAR
96     percorso.NODE[indice_nodo].$SEG_FLY := TRUE
97     percorso.NODE[indice_nodo].$COND_MASK := 1
98     percorso.NODE[indice_nodo].$SPD_OPT := SPD_LIN    -- oppure SPD_CONST
99     percorso.NODE[indice_nodo].$SEG_OVR := override_percorso
100  ENDFOR
101
102  WAIT FOR $DIN[1008]+ OR $DIN[1008] OR $DIN[1009] --Aspetto OK dalla colata del peso
      parziale OPPURE ricevo l'OK perche' e' caduto il ciclo
103
104  $ARM_OVR := 100    -- $WORD[5]    -- %Vel. da PLC // Modifica Velocita
105
106  $DOUT[1009] := ON    -- missione in corso
107
108  MOVE ALONG percorso
109
110  $FMO[5] := numero_punto + numero_tratti
111  $DOUT[1009] := OFF    -- missione in corso
112
113  $ARM_OVR := $WORD[4]    -- %Vel. Spostamento da PLC
114
115  END programma_percorso

```

A.4 Task lavaggio

```
1 PROGRAM pr_lav PROG_ARM = 1
2
3 VAR posizione_lavaggio : JOINTPOS EXPORTED FROM PUNTI
4   posizione_intermedio : JOINTPOS EXPORTED FROM PUNTI
5   jnt_attuale : JOINTPOS NOSAVE
6
7 ROUTINE ToolFrame(ai_tool, ai_frame, ai_arm : INTEGER()) EXPORTED FROM TT_TOOL GLOBAL
8 ROUTINE pos_lavaggio EXPORTED FROM pr_lav
9
10 ROUTINE pos_lavaggio
11 BEGIN
12   ToolFrame(10, 10, 1)
13
14   $DOUT[1009] := ON    -- missione in corso
15
16   jnt_attuale := ARM_JNTP
17
18   MOVEFLY RELATIVE VEC(0, 0, 100) IN BASE ADVANCE
19
20   MOVEFLY LINEAR NEAR posizione_lavaggio BY 100 ADVANCE  -- Avvicina di 100mm in Z
21     Tool
22   MOVE JOINT TO posizione_lavaggio
23
24   $DOUT[1009] := OFF  -- missione in corso
25
26 END pos_lavaggio
27 BEGIN
28
29   pos_lavaggio
30
31 END pr_lav
```

A.5 Task precolata

```
1 PROGRAM pr_pre STACK = 2048
```

```

2 VAR
3   posizione_precolata : JOINTPOS EXPORTED FROM punti
4   posizione_intermedio : JOINTPOS EXPORTED FROM punti
5   jnt_attuale : JOINTPOS NOSAVE
6
7 ROUTINE pos_precolata EXPORTED FROM pr_pre
8
9 ROUTINE ToolFrame(ai_tool, ai_frame, ai_arm : INTEGER()) EXPORTED FROM TT_TOOL GLOBAL
10
11 ROUTINE pos_precolata
12 BEGIN
13
14   ToolFrame(10, 10, 1)
15
16   $DOUT[1009] := ON    -- missione in corso
17
18   jnt_attuale := ARM_JNTP
19
20   MOVEFLY RELATIVE VEC(0, 0, 100) IN BASE ADVANCE
21
22   MOVEFLY LINEAR NEAR posizione_precolata BY 100 ADVANCE    -- Avvicina di 100mm in Z
23   Tool
24
25   MOVE JOINT TO posizione_precolata
26
27   $DOUT[1009] := OFF    -- missione in corso
28 END pos_precolata
29
30 BEGIN
31   pos_precolata
32 END pr_pre

```

A.6 Task taratura in sicurezza

```

1 PROGRAM PR_TAR_SICURA STACK = 2048
2 VAR
3   jnt_taratura_in_sicurezza : JOINTPOS EXPORTED FROM punti

```

```

4   jnt_passaggio_1 : JOINTPOS EXPORTED FROM punti
5   jnt_passaggio_2 : JOINTPOS EXPORTED FROM punti
6   jnt_passaggio_3 : JOINTPOS EXPORTED FROM punti
7   jnt_precolata_in_sicurezza : JOINTPOS EXPORTED FROM punti
8
9   posizione_fuori_ingombro : JOINTPOS EXPORTED FROM punti
10
11  posizione_lavaggio : JOINTPOS EXPORTED FROM punti
12  posizione_precolata : JOINTPOS EXPORTED FROM punti
13
14  ROUTINE pos_taratura_in_sicurezza EXPORTED FROM PR_TAR_SICURA
15
16  ROUTINE ToolFrame(ai_tool, ai_frame, ai_arm : INTEGER()) EXPORTED FROM TT_TOOL GLOBAL
17
18  ROUTINE pos_taratura_in_sicurezza
19  BEGIN
20
21    ToolFrame(10, 10, 1)
22
23    $DOUT[1009] := ON    -- missione in corso
24
25    MOVEFLY RELATIVE VEC(0, 0, 100) IN BASE ADVANCE
26
27    MOVEFLY JOINT NEAR jnt_precolata_in_sicurezza BY 100 ADVANCE    -- Avvicina di 100mm
        in Z Tool
28
29    MOVE JOINT TO jnt_precolata_in_sicurezza
30
31    BIT_ASSIGN($FMO[19], 1, TRUE)    --POSIZIONE PRECOLATA RAGGIUNTA
32
33    WAIT FOR BIT_TEST($FMI[19], 1, TRUE)    --FINE PRECOLATE
34
35    IF BIT_TEST($FMI[19], 2, TRUE) THEN    --TARATURA SU BICCHIERE ATTIVA
36
37        MOVEFLY RELATIVE VEC(0, 0, 200) IN BASE ADVANCE
38
39        MOVEFLY LINEAR NEAR jnt_taratura_in_sicurezza BY 100 ADVANCE    -- Avvicina di 100

```



```

mm in Z Tool
40
41 MOVE LINEAR TO jnt_taratura_in_sicurezza
42
43 BIT_ASSIGN($FMO[19], 2, TRUE) --POSIZIONE TARATURA RAGGIUNTA
44
45 WAIT FOR BIT_TEST($FMI[19], 3, TRUE) --FINE TARATURA
46
47 MOVEFLY RELATIVE VEC(0, 0, 100) IN BASE ADVANCE
48
49 ENDIF
50
51 MOVEFLY LINEAR NEAR posizione_precolata BY 200 ADVANCE -- Avvicina di 100mm in Z
Tool
52
53 MOVE LINEAR TO posizione_precolata
54
55 BIT_ASSIGN($FMO[19], 3, TRUE) --POSIZIONE PRECOLATA RAGGIUNTA X SPURGO
56
57 WAIT FOR BIT_TEST($FMI[19], 4, TRUE) --FINE SPURGO
58
59 MOVEFLY RELATIVE VEC(0, 0, 100) IN BASE ADVANCE
60
61 MOVEFLY JOINT NEAR posizione_lavaggio BY 100 ADVANCE -- Avvicina di 100mm in Z
Tool
62
63 MOVE JOINT TO posizione_lavaggio
64
65 BIT_ASSIGN($FMO[19], 4, TRUE) --POSIZIONE LAVAGGIO RAGGIUNTA
66
67 $DOUT[1009] := OFF -- missione in corso
68
69 END pos_taratura_in_sicurezza
70
71 BEGIN
72 pos_taratura_in_sicurezza
73 END PR_TAR_SICURA

```

A.7 Comunicazione con il PLC

```
1 PROGRAM Com_PLC NOHOLD
2
3 TYPE def_punto = RECORD
4     pos : POSITION
5     diametro : INTEGER
6     override : INTEGER
7     tipocolata : INTEGER
8     nodi : INTEGER
9     incremento_spirale : INTEGER
10    ripetizione_spirale : INTEGER
11 ENDRECORD
12
13 VAR punto : ARRAY[30] OF def_punto EXPORTED FROM lib_ini
14 VAR multiplexer : INTEGER (0) NOSAVE
15 VAR letto : INTEGER (0) NOSAVE
16
17 ROUTINE allarme(numero_allarme : INTEGER) EXPORTED FROM lib_ini
18
19 BEGIN
20     multiplexer := 0
21     letto := 0
22 CYCLE
23
24     multiplexer := $FMI[16]
25
26     IF (multiplexer < 0) OR (multiplexer > 30) THEN
27         allarme(4039) -- Missione non Possibile
28     ENDIF
29
30     IF (multiplexer > 0) AND (multiplexer <= 30) AND (letto = 0) THEN
31
32         $WORD[101] := $FMI[6]
33         punto[multiplexer].pos.X := $WORD[101]
34         $WORD[102] := $FMI[7]
35         punto[multiplexer].pos.Y := $WORD[102]
```

```

36     $WORD[103] := $FMI[8]
37     punto[multiplexer].pos.Z := $WORD[103]
38     $WORD[104] := $FMI[9]
39     punto[multiplexer].diametro := $WORD[104]
40     $WORD[105] := $FMI[10]
41     punto[multiplexer].override := $WORD[105]
42     $WORD[106] := $FMI[11]
43     punto[multiplexer].tipocolata := $WORD[106]
44     $WORD[107] := $FMI[12]
45     punto[multiplexer].nodi := $WORD[107]
46     $WORD[108] := $FMI[13]
47     punto[multiplexer].incremento_spirale := $WORD[108]
48     $WORD[109] := $FMI[14]
49     punto[multiplexer].ripetizione_spirale := $WORD[109]
50
51     $FMO[16] := multiplexer
52     letto := 1
53
54 ELSE
55     IF multiplexer = 0 THEN
56         $FMO[16] := 0
57         letto := 0
58     ENDIF
59 ENDIF
60
61 DELAY 1    -- Pausa di 1mS
62
63 END Com_PLC

```

Appendice B Comunicazione lato PLC

B.1 DB dati ricevuti

1	Static		
2	NON_ALLARMI_PRESENTI	Bool	
3	RISPOSTA_MOTORI_ACCESSI	Bool	
4	START_NO_HOLD_su_Prog_Mo	Bool	
5	SEL_CHIAVE_IN_AUT	Bool	
6	MOT_IN_COPPIA_DA_TP5	Bool	
7	DB_VAR25	Bool	
8	DB_VAR26	Bool	
9	DB_VAR27	Bool	
10	DB_VAR	Bool	
11	DB_VAR13	Bool	
12	DB_VAR14	Bool	
13	DB_VAR15	Bool	
14	Riservato	Bool	
15	SEL_CHIAVE_IN_SEM_o_AUT	Bool	
16	SISTEMA_PRONTO	Bool	
17	BIT_OSCILLATORE	Bool	
18	POS_X	Int	
19	POS_Y	Int	
20	POS_Z	Int	
21	POS_OK_X_ROTAZIONE	Bool	
22	POS_FUORI_INGOMBRO	Bool	
23	POS_PRECOLATA	Bool	
24	POS_LAVAGGIO	Bool	
25	POS_CAMBIO_MIXER	Bool	
26	POS_PRIMO_PUNTO	Bool	
27	POS_PROGRAMMA_COLATA	Bool	
28	POS_X_Peso_Parziale	Bool	
29	Mov_in_Corso	Bool	
30	Prog_MAIN_Attivo	Bool	
31	POS_USCITA_STAMPO	Bool	
32	POS_PREP_USCITA_STAMPO	Bool	
33	POS_TARATURA_PESO_SPECIFICO	Bool	
34	POS_TARATURA_SICURA	Bool	
35	DOUT[1015]	Bool	
36	DOUT[1016]	Bool	
37	MISSIONE_IN_CORSO	Int	
38	CONTATORE_PUNTI	Int	
39	FMO[6]	Int	
40	FMO[7]	Int	
41	FMO[8]	Int	
42	FMO[9]	Int	
43	FMO[10]	Int	
44	FMO[11]	Int	
45	FMO[12]	Int	
46	FMO[13]	Int	
47	FMO[14]	Int	
48	FMO[15]	Int	
49	MULTIPLEXER_Indice	Int	
50	FMO[17]	Int	
51	FMO[18]	Int	
52	FMO[19]bit9	Bool	
53	FMO[19]bit10	Bool	
54	FMO[19]bit11	Bool	
55	FMO[19]bit12	Bool	
56	FMO[19]bit13	Bool	
57	FMO[19]bit14	Bool	
58	FMO[19]bit15	Bool	
59	FMO[19]bit16	Bool	
60	POS_PRECOLATA_RAGGIUNTA	Bool	
61	POS_BICCHIERE_RAGGIUNTA	Bool	
62	POS_PRECOLATA_X_SPURGO_R...	Bool	
63	POS_LAVAGGIO_RAGGIUNTA	Bool	
64	FMO[19]bit5	Bool	
65	FMO[19]bit6	Bool	
66	FMO[19]bit7	Bool	
67	FMO[19]bit8	Bool	
68	FMO[20]	Int	
69	FMO[21]	Int	
70	FMO[22]	Int	
71	FMO[23]	Int	
72	FMO[24]	Int	
73	LIMITI_MAX_X_mm	Int	
74	LIMITI_MAX_Y_mm	Int	
75	LIMITI_MAX_Z_mm	Int	
76	LIMITI_min_X_mm	Int	
77	LIMITI_min_Y_mm	Int	
78	LIMITI_min_Z_mm	Int	

B.2 DB dati trasmessi

1	Static	
2	DRIVE_ON	Bool
3	NOT_DRIVE_OFF	Bool
4	START	Bool
5	NOT_HOLD	Bool
6	U1	Bool
7	U2	Bool
8	U3	Bool
9	U4	Bool
10	Reset_Allarmi	Bool
11	Vel_di_Sicurezza	Bool
12	DB_VAR14	Bool
13	DB_VAR15	Bool
14	DB_VAR16	Bool
15	DB_VAR17	Bool
16	DB_VAR18	Bool
17	DB_VAR19	Bool
18	SET_VELOCITA_TRA_COLATE	Int
19	FMI[2]	Int
20	FMI[3]	Int
21	DIN[1001]	Bool
22	POS_FUORI_INGOMBRO	Bool
23	POS_PRECOLATA	Bool
24	POS_LAVAGGIO	Bool
25	POS_CAMBIO_MIXER	Bool
26	POS_PRIMO_PUNTO	Bool
27	POS_START_PROG_COLATA	Bool
28	START_COLATA	Bool
29	FINE_PROGRAMMA_MOV	Bool
30	ABILITO_MOV_IN_COLATA	Bool
31	POS_USCITA_STAMPO	Bool
32	POS_PREP_USCITA_STAMPO	Bool
33	POS_TARATURA_PESO_SPECIFI...	Bool
34	POS_TARATURA_SICURA	Bool
35	DIN[1015]	Bool
36	PREP_USCITA_ABILITATA	Bool
37	SET_VELOCITA_DI_SPOSTAM	Int
38	SET_CONTATORE_PUNTI	Int

39	Multiplexer	Struct
40	POS_X	Int
41	POS_Y	Int
42	POS_Z	Int
43	DIAMETRO	Int
44	OVERRIDE	Int
45	TIPO_COLATA	Int
46	NODI	Int
47	INCREMENTO_SPIRALE_mm	Int
48	RIPETIZIONE_INCREMENTI	Int
49	ris3	Int
50	Indice	Int
51	NUMERO_FRAME	Int
52	FMI[18]	Int
53	FMI[19]bit9	Bool
54	FMI[19]bit10	Bool
55	FMI[19]bit11	Bool
56	FMI[19]bit12	Bool
57	FMI[19]bit13	Bool
58	FMI[19]bit14	Bool
59	FMI[19]bit15	Bool
60	FMI[19]bit16	Bool
61	FINE_PRECOLATE	Bool
62	VAL_SU_BICCHIERE	Bool
63	FINE_TARATURA_SU_BICCHIERE	Bool
64	FINE_SPURGO	Bool
65	FMI[19]bit5	Bool
66	FMI[19]bit6	Bool
67	FMI[19]bit7	Bool
68	FMI[19]bit8	Bool
69	FMI[20]	Int
70	FMI[21]	Int
71	FMI[22]	Int
72	FMI[23]	Int
73	FMI[24]	Int
74	FMI[25]	Int
75	FMI[26]	Int
76	FMI[27]	Int
77	FMI[28]	Int
78	FMI[29]	Int
79	FMI[30]	Int

B.3 Ricezione e trasmissione in AWL

```
1      L      1256
2      SLW    3
3      LAR1
4      //#####
5
6      AUF    "
          DB902_RX_da_ROBOT_Profinet"
7
8      L PED [ AR1 , P#0.0 ]
9      T      %DBD0
10
11     L PED [ AR1 , P#4.0 ]
12     T      %DBD4
13
14     L PED [ AR1 , P#8.0 ]
15     T      %DBD8
16
17     L PED [ AR1 , P#12.0 ]
18     T      %DBD12
19
20     ...
21
22     L PED [ AR1 , P#52.0 ]
23     T      %DBD52
24
25     L PED [ AR1 , P#56.0 ]
26     T      %DBD56
27
28     L PED [ AR1 , P#60.0 ]
29     T      %DBD60
```

```
1      L      1256
2      SLW    3
3      LAR1
4      //#####
5
6      AUF    "
          DB901_TX_a_ROBOT_Profinet"
7
8      L      %DBD0
9      T PAD [ AR1 , P#0.0 ]
10
11     L      %DBD4
12     T PAD [ AR1 , P#4.0 ]
13
14     L      %DBD8
15     T PAD [ AR1 , P#8.0 ]
16
17     L      %DBD12
18     T PAD [ AR1 , P#12.0 ]
19
20     ...
21
22     L      %DBD52
23     T PAD [ AR1 , P#52.0 ]
24
25     L      %DBD56
26     T PAD [ AR1 , P#56.0 ]
27
28     L      %DBD60
29     T PAD [ AR1 , P#60.0 ]
```

B.4 Multiplexer scambio dati

```
1      ON      #Tx
2      SPBN   INIZ
3      L      0
4      T      #INDICE_MULTIPLEXER
5      T      "DB901_TX_a_ROBOT_Profinet".Multiplexer.Indice
6
7      L      0
8      T      #NUM_SCRITTURE_ESEGUITE
9  INIZ: NOP 0
10
11     U(
12     L      "DB902_RX_da_ROBOT_Profinet".MULTIPLEXER_Indice
13     L      "DB901_TX_a_ROBOT_Profinet".Multiplexer.Indice
14     ==I
15     )
16     U      #Tx
17     SPBN   AZZ
18     L      0
19     T      "DB901_TX_a_ROBOT_Profinet".Multiplexer.Indice
20  AZZ:  NOP 0
21
22  //#####
23
24     U(
25     L      "DB902_RX_da_ROBOT_Profinet".MULTIPLEXER_Indice
26     L      0
27     ==I
28     )
29     U(
30     L      "DB901_TX_a_ROBOT_Profinet".Multiplexer.Indice
31     L      0
32     ==I
33     )
34     U      #Tx
35     SPBN   AGG
```

```

36
37 //#####
38 // indirizzi Iniziali Dati
39
40 L 176 // DBW 120
41 T #L_Ind0_Coord_X
42
43 L 264 // DBW 180
44 T #L_Ind0_Coord_Y
45
46 L 352 // DBW 240
47 T #L_Ind0_Coord_Z
48
49 L 440
50 T #L_Ind0_Diametro
51
52 L 528
53 T #L_Ind0_Velocita
54
55 L 616
56 T #L_Ind0_Tipo
57
58 L 660
59 T #L_Ind0_Nodi
60
61 L #INDICE_MULTIPLEXER
62 L 1
63 +I
64 T #INDICE_MULTIPLEXER
65
66 O(
67 L #INDICE_MULTIPLEXER
68 L #MaxPunti
69 >I
70 )
71 O(
72 L #INDICE_MULTIPLEXER

```



```

73     L     1
74     <I
75     )
76     SPBN  CTRM
77     L     1
78     T     #INDICE_MULTIPLEXER
79
80     L     #NUM_SCRITTURE_ESEGUITE
81     L     1
82     +I
83     T     #NUM_SCRITTURE_ESEGUITE
84
85     CTRM: NOP 0
86
87     L     #DB_Stazione
88     T     #L_Stazione
89
90     AUF DI [ #L_Stazione]
91
92     L     #INDICE_MULTIPLEXER
93     L     1
94     -I
95     T     #L_multiplexerByte
96     L     2
97     *I
98     T     #L_multiplexer
99
100    L     #L_Ind0_Coord_X
101    L     #L_multiplexer
102    +I
103    SLW   3
104    LAR1
105    L DIW [ AR1 , P#0.0 ]
106    L     "DB800_OFF_SET_CORD_ROBOT_xyz".off_set_COORDINATA_X
107    +I
108    T     #L_CORDINATA
109    CALL  "FC616_CTRL_Limiti_INT"

```

```

110      Input   :=#L_CORDINATA
111      min     :="DB902_RX_da_ROBOT_Profinet".LIMITI_min_X_mm
112      Max     :="DB902_RX_da_ROBOT_Profinet".LIMITI_MAX_X_mm
113      OUTPUT  :="DB901_TX_a_ROBOT_Profinet".Multiplexer.POS_X
114
115      L       #L_Ind0_Coord_Y
116      L       #L_multiplexer
117      +I
118      SLW    3
119      LAR1
120      L DIW [ AR1 , P#0.0 ]
121      L       "DB800_OFF_SET_CORD_ROBOT_xyz".off_set_COORDINATA_Y
122      +I
123      T       #L_CORDINATA
124      CALL    "FC616_CTRL_Limiti_INT"
125      Input   :=#L_CORDINATA
126      min     :="DB902_RX_da_ROBOT_Profinet".LIMITI_min_Y_mm
127      Max     :="DB902_RX_da_ROBOT_Profinet".LIMITI_MAX_Y_mm
128      OUTPUT  :="DB901_TX_a_ROBOT_Profinet".Multiplexer.POS_Y
129
130      L       #L_Ind0_Coord_Z
131      L       #L_multiplexer
132      +I
133      SLW    3
134      LAR1
135      L DIW [ AR1 , P#0.0 ]
136      L       "DB800_OFF_SET_CORD_ROBOT_xyz".off_set_COORDINATA_Z
137      +I
138      T       #L_CORDINATA
139
140      CALL    "FC616_CTRL_Limiti_INT"
141      Input   :=#L_CORDINATA
142      min     :="DB902_RX_da_ROBOT_Profinet".LIMITI_min_Z_mm
143      Max     :="DB902_RX_da_ROBOT_Profinet".LIMITI_MAX_Z_mm
144      OUTPUT  :="DB901_TX_a_ROBOT_Profinet".Multiplexer.POS_Z
145
146      L       #L_Ind0_Diametro

```

```

147     L     #L_multiplexer
148     +I
149     SLW  3
150     LAR1
151     L DIW [ AR1 , P#0.0 ]
152     T     "DB901_TX_a_ROBOT_Profinet".Multiplexer.DIAMETRO
153
154     L     #L_Ind0_Velocita
155     L     #L_multiplexer
156     +I
157     SLW  3
158     LAR1
159     L DIW [ AR1 , P#0.0 ]
160     T     "DB901_TX_a_ROBOT_Profinet".Multiplexer.OVERRIDE
161
162     L     #L_Ind0_Tipo
163     L     #L_multiplexerByte // #L_multiplexer
164     +I
165     SLW  3
166     LAR1
167     L DIB [ AR1 , P#0.0 ]
168     T     "DB901_TX_a_ROBOT_Profinet".Multiplexer.TIPO_COLATA
169
170     L     #L_Ind0_Nodi
171     L     #L_multiplexer
172     +I
173     SLW  3
174     LAR1
175     L DIW [ AR1 , P#0.0 ]
176     T     "DB901_TX_a_ROBOT_Profinet".Multiplexer.NODI
177
178 // #####
179
180     L     #INDICE_MULTIPLEXER
181     T     "DB901_TX_a_ROBOT_Profinet".Multiplexer.Indice
182
183 AGG:  NOP 0

```

Bibliografia

- [1] Ian Clemitson. *Polyurethane Casting Primer*. CRC Press, 2012.
- [2] I.R. Clemitson. *Castable Polyurethane Elastomers*. CRC Press, first edition edition, 2008.
- [3] Comau. C5g plus. <https://www.comau.com/it/competencies/robotics-automation/controls-auxiliary-equipment/c5g-plus/>. Accessed on 2023-01-30.
- [4] Comau. Nj-60-2.2: Una soluzione perfetta per applicazioni con media capacità di carico. <https://www.comau.com/it/competencies/robotics-automation/robot-team/nj-60-2-2/>. Accessed on 2023-01-30.
- [5] Comau. Unità di controllo c5gplus - programmazione del movimento. Manuale di istruzioni. Accessed on 2023-09-20.
- [6] Università degli Studi di Napoli Federico II. Pdl2. <http://www.prisma.unina.it/courses/pdl2.pdf>. Accessed on 2023-09-19.
- [7] GT Engineering. 5.12 - limitazione degli assi. <https://www.gt-engineering.it/normative-tecniche/normative-en-iso/en-iso-10218-1/5-7-limitazione-degli-assi/>. Accessed on 2023-10-10.
- [8] GT Engineering. En iso 12100: L'analisi del rischio. <https://www.gt-engineering.it/normative-en-iso/analisi-rischio-en-12100>. Accessed on 2023-01-27.
- [9] GT Engineering. Le norme armonizzate di tipo a, b, c. <https://www.gt-engineering.it/approfondimenti/sicurezza-macchine/le-norme-armonizzate-di-tipo-a-b-c/>. Accessed on 2023-10-10.
- [10] GT Engineering. Pulsanti di emergenza e la funzione di arresto. <https://www.gt-engineering.it/approfondimenti/sicurezza-macchine/pulsanti-emergenza-funzione-di-arresto/>. Accessed on 2023-10-10.
- [11] Alicia Casals Josep Amat. *I Robot del Futuro*, volume 67 of *Una passeggiata nel cosmo*. RBA Italia, 2022.

- [12] Raimond Pigan and Mark Metter. *Automating with PROFINET: Industrial Communication Based on Industrial Ethernet*. Wiley-VCH, 2008.
- [13] Donald Dominick Matthew Rosato Rosato Rosato. *Plastic Product Material and Process Selection Handbook*. Elsevier Science, 1 edition, 2004.
- [14] C. Werner, H. Zilligen, B. Köhler, and R. Apfeld. *Safe drive controls with frequency inverters*. IFA Report. DGUV/IFA, 2019.
- [15] Xerion. Understanding precision: accuracy, repeatability, precision. <https://xeryon.com/technology/understanding-precision/#:~:text=Accuracy%20is%20how%20close%20a,properties%20of%20a%20positioning%20system>. Accessed on 2023-01-17.