



# **CONTROLLO E SUPERVISIONE ONLINE DI UN IMPIANTO INDUSTRIALE**

**Istituto di Istruzione Superiore  
"Italo Calvino"**

**GEIS01400Q**

Indirizzo informatico e telecomunicazioni  
Indirizzo Elettronico ed elettrotecnico  
Liceo Scientifico opzione Scienze Applicate

via Borzoli 21 - 16153 **Genova**  
Tel. +39 010 6508778 +39 010 6504672



### CONTROLLO E SUPERVISIONE ONLINE DI UN IMPIANTO INDUSTRIALE

Nel mondo industriale sta crescendo la necessità di avere a disposizione, in ogni luogo, informazioni cruciali sullo stato di un impianto, ad esempio per intervenire in caso di emergenza, per realizzare manutenzioni predittive, ridurre i periodi di inattività, incrementare la produttività, abbassare i costi di manutenzione e intervento,...

Oggi è possibile realizzare tutto questo connettendo i dispositivi che controllano l'impianto (PLC, IPC,...) ad una rete **IoT (Internet of Things)** permettendo così di gestirli da remoto.

Uno degli **obiettivi del progetto** è dimostrare come sia possibile **controllare** e **supervisionare** sia **da locale** sia **da remoto** un **impianto industriale**, nella fattispecie **costituito da tre macchine** che lavorano dei manufatti cilindrici ferrosi.

La prima macchina è una catena di lavorazione dotata di una fresatrice e di un trapano, la seconda macchina è una punzonatrice e l'ultima un robot cilindrico che serve a movimentare i pezzi tra le prime due e il deposito.

Tutto l'impianto è controllato mediante un sistema a **Intelligenza Distribuita**. Rispetto a un sistema a **Intelligenza Concentrata** dove tutto l'impianto sarebbe guidato da un unico controllore (PLC), nei sistemi a "Intelligenza Distribuita" ciascuna macchina è comandata autonomamente da un proprio PLC. Questa scelta consente uno sviluppo modulare del progetto, riduce i tempi di sviluppo e messa in servizio (**commissioning**) e, in caso di guasti, limita i tempi di fermo macchina.

La **supervisione locale**, realizzata mediante un pannello operatore (**HMI**) touch screen, consente di conoscere, istante per istante, lo stato dell'impianto, il numero di pezzi attualmente in lavorazione e quelli prodotti in totale. Inoltre degli appositi contatori consentono di predire l'usura degli utensili impiegati e anticipare così gli interventi di manutenzione.

Per la **supervisione remota** abbiamo adottato tecnologie tipiche dell'**Industria 4.0**, quali l'**IoT** e la **cyber security**. In questo modo potremo connetterci all'impianto praticamente da ogni luogo, protetti, nel contempo, da attacchi e violazioni informatiche indesiderate.

Questo sistema di supervisione consente agli addetti locali di avere sotto controllo il funzionamento di tutto l'impianto e di pianificarne gli interventi di manutenzione ordinaria. In caso di guasti o di anomalie non previste, i tecnici locali potranno richiedere assistenza al supporto tecnico remoto che, con l'ausilio di una videocamera, potrà eventualmente operare su ciascuna macchina dell'impianto.

Se l'intervento remoto non fosse risolutivo, il centro assistenza potrà sempre inviare ai tecnici locali la soluzione al problema sotto forma di istruzioni, disegni tecnici o filmati che illustrano le modalità di intervento sulla parte da riparare o sostituire.

Grazie a queste tecnologie è possibile ottimizzare la qualità degli interventi, ridurre i tempi di riparazione e **risparmiare sulle spese di trasferta** del personale specializzato che non si dovrà più recare fisicamente dove si trova l'impianto.

Indirettamente si otterrebbero anche dei **vantaggi per l'ambiente**. Non sarebbe più necessario ricorrere a voli aerei né a lunghi e ripetuti trasferimenti in auto. Inoltre aumenterebbe la **qualità della vita dei lavoratori** che mediante il **telelavoro** potrebbero comodamente operare da casa propria nei tempi e nei modi che preferiscono. Diminuirebbe la probabilità di **infortuni**, sia sul luogo di lavoro sia durante il trasferimento. Si ridurrebbe la necessità di doversi recare in zone di lavoro a rischio (per guerre, malattie, epidemie,...).

Una **possibile evoluzione di questo progetto** potrebbe essere quella di impiegare visori per la **Realtà Aumentata (AR)** e per la **Realtà Virtuale (VR)** o, meglio, per entrambi i sistemi (**Realtà Mista, MR**).

La multimedialità di questi metodi consentirà ai tecnici locali di eseguire interventi più rapidi e consapevoli. Non dovranno perdere tempo per consultare il laptop con una mano e telefonare con l'altra per chiamare il centro assistenza e ottenere chiarimenti.

Semplicemente guardando con questi particolari occhiali il punto dove si è verificata l'anomalia, i tecnici potranno vedere in sovraimpressione indicazione su come operare oppure ottenere informazioni sulla validità di quanto eseguito, messaggi di errore e suggerimenti di correzione.

Questi dispositivi potranno essere utili anche ai fini dell'**addestramento e formazione degli addetti**. Una persona che indossa questo visore può collegarsi in remoto con più persone e svolgere una lezione dal vivo senza bisogno di spostarsi o viaggiare ("**on work**"). Rappresentano pertanto un chiaro vantaggio in termini di risparmio di tempo e denaro e un incremento di velocità ed efficienza.

## Specifiche generali

Il **ciclo di lavorazione** consiste nelle operazioni sequenziali di fresatura, foratura e punzonatura dei manufatti.

Il **pannello operatore** (HMI) consente di:

- visionare lo stato delle singole macchine dell'impianto;
- visualizzare il numero di pezzi attualmente in produzione e il numero di quelli già prodotti;
- segnalare quando intervenire per la sostituzione degli utensili;
- controllare manualmente la posizione nello spazio dell'end effector del robot;
- arrestare una determinata macchina (sia per motivi di sicurezza sia per interventi di manutenzione);
- riprendere la lavorazione dal punto in cui era stata interrotta;
- arrestare con un unico comando di emergenza tutto l'impianto.

Le stesse funzioni sono rese disponibili su **cellulare, tablet o PC remoti** mediante accesso (login) con credenziali di sicurezza. In ogni caso, per ragioni di sicurezza, non sarà possibile accedere via Internet ai singoli PLC dell'ambiente di produzione, ma solo al gateway dell'impianto.

## Specifiche tecniche

Ciascuna macchina dell'impianto è comandata da un **PLC Siemens S7-1200**. La loro coordinazione avviene via **rete Profinet**, mediante il **protocollo di comunicazione Siemens S7**.

La **supervisione locale** dell'impianto è stata realizzata usando un pannello operatore **HMI Siemens KTP700 Basic**, collegato mediante uno **switch Siemens SCALANCE XB005** alla stessa rete Profinet dei PLC.

La **supervisione remota** è stata invece implementata utilizzando un **gateway Siemens IOT2020**, anch'esso collegato alla rete Profinet dell'impianto.

Questo gateway consente di implementare una vera e propria rete **IoT** (**I**nternet **o**f **T**hings) dove, mediante un comune browser internet, sarà possibile interagire con l'hardware desiderato in modo controllato e sicuro.

La **cyber security** ovvero l'implementazione della sicurezza delle comunicazioni e della riservatezza dei dati scambiati via Internet, è stata realizzata connettendo il gateway dell'impianto e i dispositivi remoti mediante un **tunnel VPN** dedicato.

Di seguito saranno approfondite le principali caratteristiche del sistema.

## 1. CONTROLLO

### Catena di lavorazione con 2 utensili

Questa macchina ha il compito di fresare e successivamente trapanare dei manufatti cilindrici. È costituita da:

- Una fresatrice
- Un trapano
- 4 nastri trasportatori monodirezionali
- 2 slider bidirezionali
- 5 barriere fotoelettriche (NC)
- 4 contatti di finecorsa, 2 per ogni slider, (NO)

I nastri trasportatori e i due slider consentono di movimentare i pezzi da lavorare mentre le cinque barriere fotoelettriche indicano la loro posizione nell'impianto.

Il sinottico che segue (Figura 1) schematizza sinteticamente la macchina (vista in pianta) e rappresenta lo stato della catena di lavorazione in cui sono (per esempio) presenti contemporaneamente quattro pezzi (cerchi grandi):

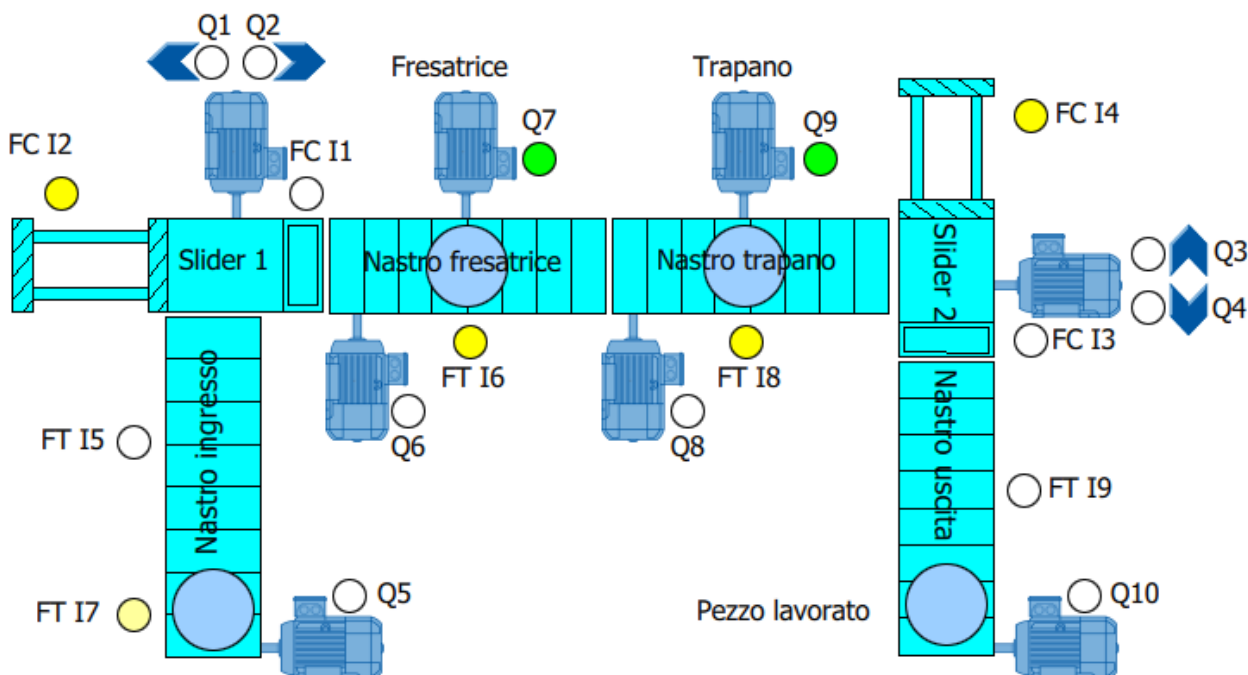


Figura 1

I cerchi piccoli segnalano lo stato dei sensori (fototransistor delle barriere e finecorsa degli slider) e dei motori (nastri trasportatori e utensili). Il colore giallo rappresenta un sensore attivato, il colore verde un motore acceso.

Per sviluppare e simulare in modo più realistico il programma di controllo della catena di lavorazione, abbiamo impiegato un modello in scala prodotto dalla **Fischertechnik** e rappresentato in Figura 2:

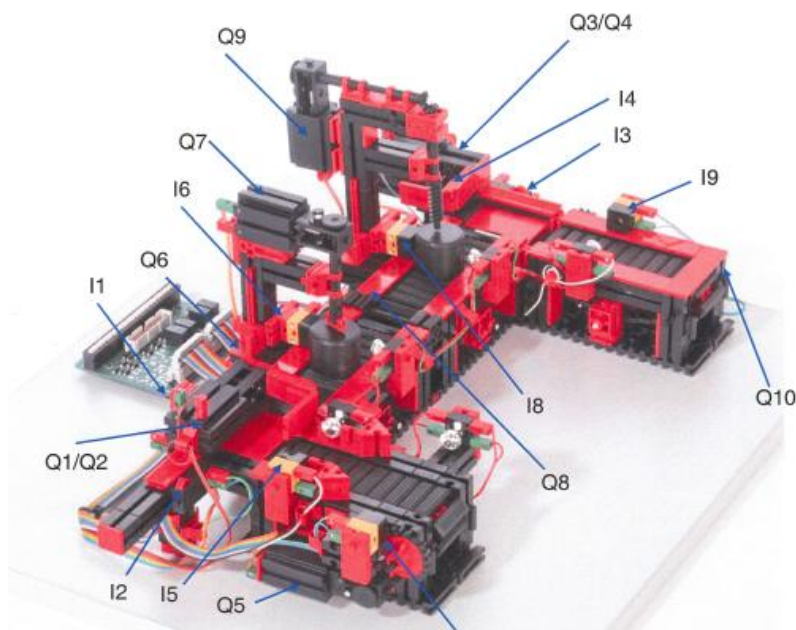


Figura 2

Per controllare questa macchina abbiamo impiegato un PLC **Siemens S7-1214C AC/DC/RLY** che, tra l'altro, dispone di 14 ingressi e 10 uscite digitali (a relè) oltre a una connessione di rete Profinet.

Le connessioni elettriche tra ingressi/uscite del modello Fischertechnik e il PLC che lo controlla sono elencate nella seguente **tabella delle associazioni**:

Pin modello	Funzione	I/O PLC	I/O modello
1	Alimentazione (+) attuatori	L+	24V DC
2	Alimentazione (+) sensori	L+	24V DC
3	Alimentazione (-)	M	0V
4	Alimentazione (-)	M	0V
5	Finecorsa slider 1 avanti (NO)	I0.0	I1
6	Finecorsa slider 1 indietro (NO)	I0.1	I2
7	Finecorsa slider 2 avanti (NO)	I0.2	I3
8	Finecorsa slider 2 indietro (NO)	I0.3	I4
9	Fototransistor slider 1 (NC)	I0.4	I5
10	Fototransistor fresatrice (NC)	I0.5	I6
11	Fototransistor stazione di carico (NC)	I0.6	I7
12	Fototransistor trapano (NC)	I0.7	I8
13	Fototransistor barriera d'uscita (NC)	I1.0	I9
14			
15	Motore slider 1 indietro	Q0.0	Q1
16	Motore slider 1 avanti	Q0.1	Q2
17	Motore slider 2 indietro	Q0.2	Q3
18	Motore slider 2 avanti	Q0.3	Q4
19	Motore nastro trasportatore d'ingresso	Q0.4	Q5
20	Motore nastro trasportatore fresatrice	Q0.5	Q6
21	Motore fresatrice	Q0.6	Q7
22	Motore nastro trasportatore trapano	Q0.7	Q8
23	Motore trapano	Q1.0	Q9
24	Motore nastro trasportatore d'uscita	Q1.1	Q10

### Specifiche di funzionamento

Il processo di lavorazione svolto da questa macchina parte dallo stato in cui tutti i motori sono spenti ed entrambi gli slider sono arretrati.

1. Quando il fototransistor della stazione di carico rileva un pezzo da lavorare (I7 OFF), dopo 1" si avvia il nastro trasportatore d'ingresso (Q5 ON) che lo trasporta fino al traguardo intermedio (fototransistor I5 OFF);
2. per fare arrivare il pezzo fino allo slider 1 il nastro trasportatore d'ingresso rimane in funzione ancora per 1";
3. trascorso questo tempo il nastro trasportatore d'ingresso si arresta (Q5 OFF) mentre lo slider 1 spinge il pezzo (Q2 ON) sul nastro trasportatore della fresatrice;
4. quando lo slider 1 arretra (Q2 OFF, Q1 ON), il nastro trasportatore della fresatrice si avvia (Q6 ON) fino a trasportare il pezzo sotto l'utensile (fototransistor I6 OFF), quindi si arresta (Q6 OFF);
5. la fresatura (Q7 ON) dura 3" dopodiché vengono simultaneamente avviati i nastri trasportatori della fresatrice (Q6 ON) e del trapano (Q8 ON) fino a quando il pezzo non arriva sotto al trapano (fototransistor I8 OFF);
6. il pezzo viene forato (Q9 ON) per 3";
7. dopo la foratura si aziona nuovamente per 1" il nastro trasportatore del trapano (Q8 ON) per fare arrivare il pezzo sullo slider 2;
8. dopo questo intervallo di tempo il nastro trasportatore del trapano si arresta (Q8 OFF) mentre lo slider 2 spinge il pezzo (Q4 ON) sul nastro trasportatore d'uscita;
9. quando lo slider 2 arretra (Q4 OFF, Q3 ON), il nastro trasportatore d'uscita si avvia (Q10 ON) fino a portare il pezzo in corrispondenza della barriera d'uscita (fototransistor I9 OFF);
10. dopo che il pezzo è stato intercettato dal fototransistor I9, il nastro trasportatore d'uscita viene azionato ancora per 1" (Q10 ON) per fare arrivare il pezzo nella zona di prelevamento del robot;
11. trascorso questo tempo, il PLC della catena di lavorazione **comunica (via rete Profinet)** al PLC che controlla il robot che il pezzo è stato lavorato ed è pronto per essere prelevato.

## I.I.S. ITALO CALVINO - Genova

In una prima versione avevamo pensato di scrivere il programma come una semplice sequenza di stati e transizioni, dove ogni operazione era consequenziale a quella precedente.

Successivamente, per ottimizzare il processo, abbiamo implementato dei parallelismi che hanno consentito la lavorazione contemporanea di più pezzi (massimo quattro).

In questa seconda soluzione, dopo che lo slider 1 ha spinto un pezzo sul nastro trasportatore della fresatrice ed è tornato indietro, è possibile posizionare un nuovo pezzo nella stazione di carico.

Terminata la fresatura, entrambi i pezzi avanzano nella catena, il primo arriva sotto al trapano, il secondo viene spinto dallo slider 1 sotto la fresatrice.

Quando lo slider 1 ritorna indietro è possibile inserire il terzo pezzo nella stazione di carico.

Terminate le operazioni di fresatura e foratura, il pezzo forato avanza fino al termine della catena di lavorazione dove potrà essere prelevato dal robot, quello fresato finisce sotto al trapano e quello in attesa nella stazione di carico viene spinto sotto la fresatrice.

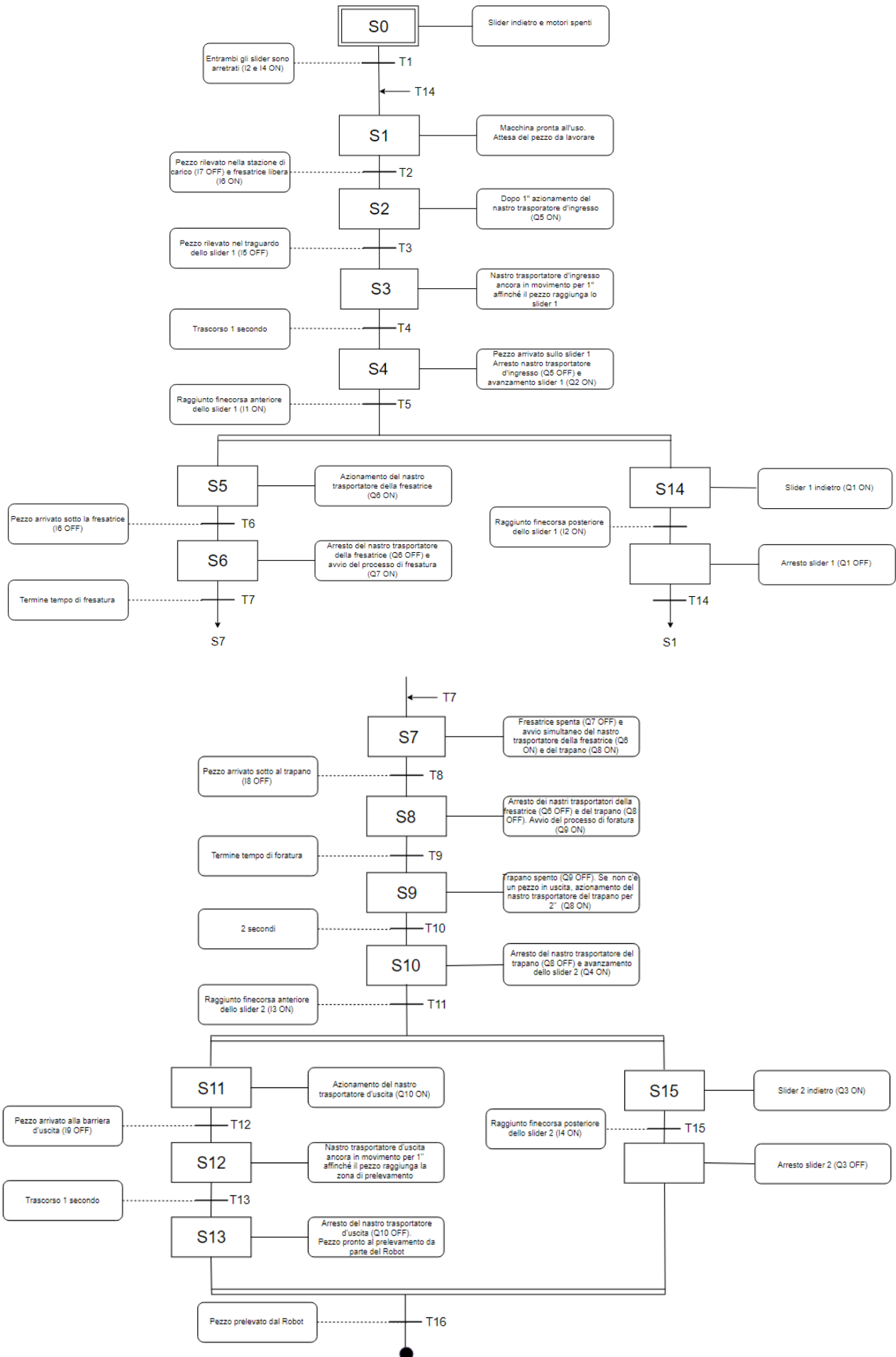
Una volta che lo slider 1 è tornato indietro, mentre i due pezzi intermedi sono in lavorazione, il nastro trasportatore d'ingresso potrà ricevere un nuovo pezzo da lavorare.

Quando il robot preleverà il pezzo lavorato, la catena di lavorazione continuerà a lavorare i pezzi intermedi spostandoli sui vari nastri trasportatori liberando un nuovo posto nella stazione di carico...

Per descrivere e comprendere meglio questo processo abbiamo impiegato un **diagramma funzionale - tecnologico**:



# I.I.S. ITALO CALVINO - Genova



Per programmare il PLC che controlla questo impianto sarebbe stato naturale scegliere il linguaggio S7-GRAPH (SFC) perché ci avrebbe consentito di implementare direttamente, con semplicità, il diagramma funzionale-tecnologico visto in precedenza.

Poiché il PLC scelto non prevede questo **linguaggio di programmazione**, abbiamo ripiegato sul linguaggio **KOP** (LD) utilizzando la **metodologia batch o della sequenza di passi**.

Seguendo le semplici regole di questa tecnica è stato relativamente agevole implementare il diagramma funzionale-tecnologico (che descrive gli stati e le transizioni che fanno evolvere il processo) in linguaggio KOP.

### Punzonatrice

Questa seconda macchina ha il compito di punzonare i manufatti cilindrici, già fresati e forati, provenienti dalla catena di lavorazione e trasportati qui dal robot. È costituita da:

- 1 nastro trasportatore bidirezionale
- 1 punzonatore
- 2 barriere fotoelettriche (NC)
- 2 contatti di finecorsa (NO)

Il nastro trasportatore consente di movimentare il pezzo dalla zona in cui viene depositato dal robot a quella dove sarà punzonato, e viceversa.

Il sinottico rappresentato in Figura 3 rappresenta un esempio di stato della macchina punzonatrice. Il pezzo depositato dal robot è stato rilevato nella barriera di carico dal fototransistor I1 (luce gialla). Il motore del nastro trasportatore si è avviato (luce verde, direzione avanti) per portarlo verso la barriera di lavorazione:

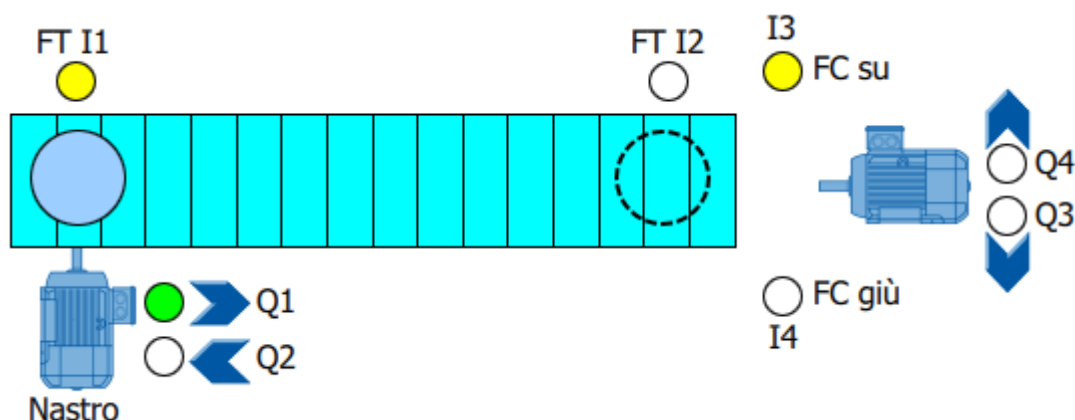


Figura 3

Il cerchio tratteggiato indica la posizione sotto al punzonatore dove il pezzo sarà lavorato.

Anche in questo caso nello sviluppo del programma che controlla la macchina punzonatrice ci siamo avvalsi di un modello in scala prodotto dalla **Fischertechnik** e rappresentato in Figura 4:

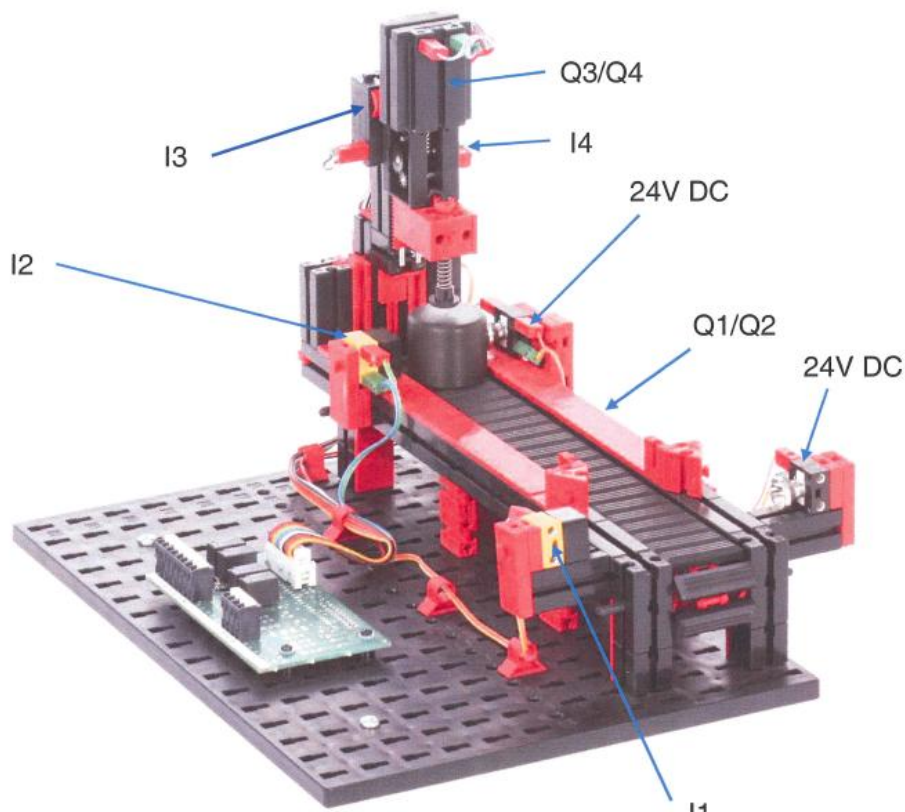


Figura 4

Per controllare questa macchina abbiamo impiegato un PLC **Siemens S7-1214C DC/DC/DC** che dispone di 14 ingressi, 10 uscite digitali (a transistor) ed una connessione di rete Profinet.

Le connessioni elettriche tra ingressi/uscite del PLC e del modello Fischertechnik controllato sono elencate nella seguente **tabella delle associazioni**:

Pin modello	Funzione	I/O PLC	I/O modello
1	Alimentazione (+) attuatori	L+	24V DC
2	Alimentazione (+) sensori	L+	24V DC
3	Alimentazione (-)	M	0V
4	Alimentazione (-)	M	0V
5	Fototransistor barriera ingresso (NC)	I0.1	I1
6	Fototransistor barriera lavorazione (NC)	I0.2	I2
7	Finecorsa punzonatore in alto (NO)	I0.3	I3
8	Finecorsa punzonatore in basso (NO)	I0.4	I4
15	Motore nastro trasportatore avanti	Q0.1	Q1
16	Motore nastro trasportatore indietro	Q0.2	Q2
17	Motore punzonatrice su	Q0.3	Q3
18	Motore punzonatrice giù	Q0.4	Q4

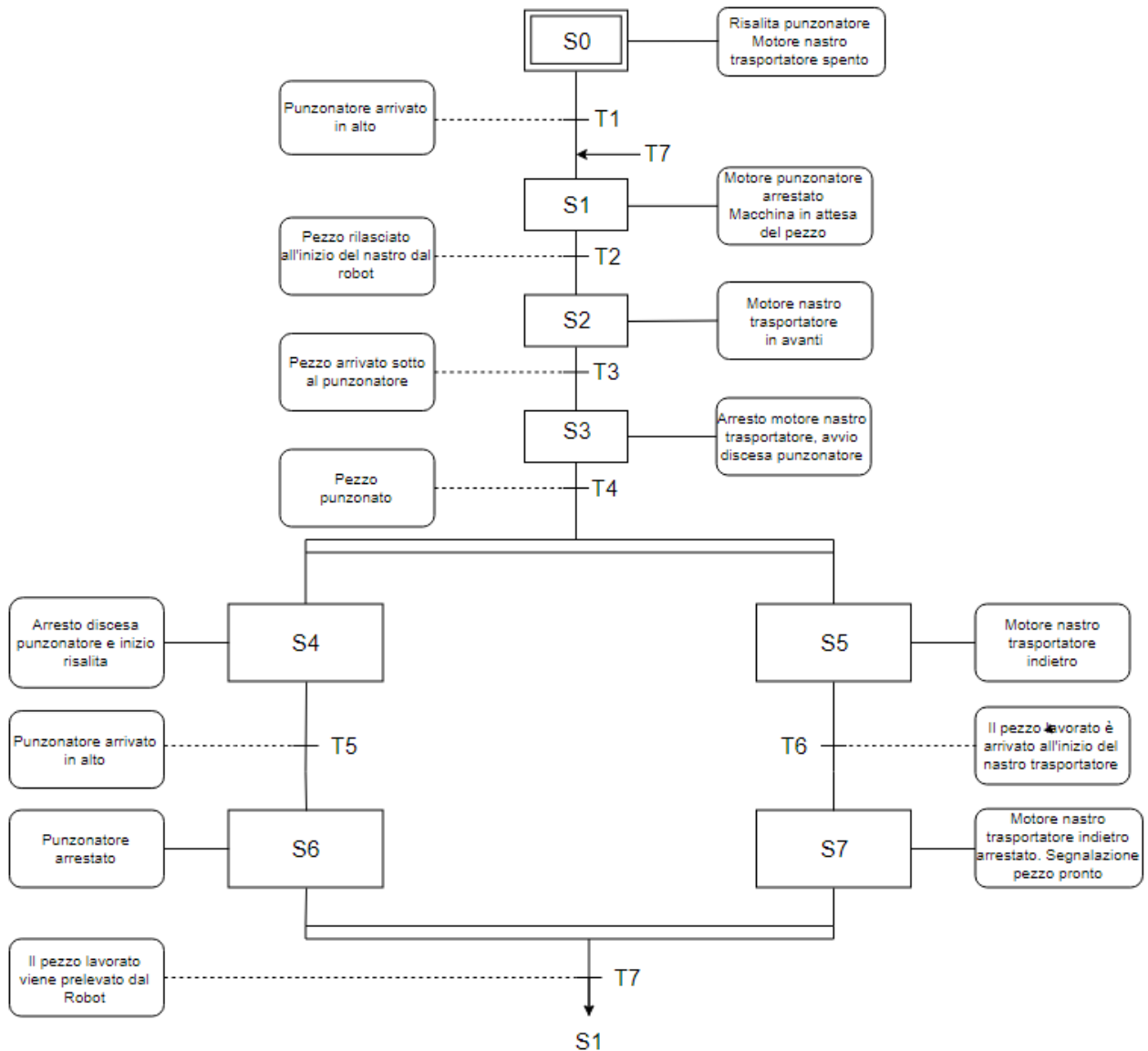
### Specifiche di funzionamento

Il processo di lavorazione della punzonatrice parte dallo stato in cui il punzonatore è in alto e tutti i motori sono spenti.

## I.I.S. ITALO CALVINO - Genova

1. Quando il fototransistor della barriera d'ingresso rileva il pezzo da lavorare depositato dal robot (I1 OFF), il nastro trasportatore d'ingresso si avvia in avanti (Q1 ON), portandolo verso il punzonatore;
2. quando il pezzo arriva alla barriera di lavorazione (fototransistor I2 OFF) il nastro trasportatore si arresta (Q1 OFF) e l'utensile scende per punzonare il pezzo (motore Q4 ON);
3. una volta punzonato il manufatto (finecorsa I4 ON), mentre il punzonatore risale (Q3 ON), il nastro trasportatore inverte la direzione di marcia (Q2 ON) per trasportare il pezzo verso la barriera d'ingresso;
4. quando il punzonatore arriva in alto (finecorsa I3 ON) il suo motore si arresta (Q3 OFF);
5. una volta che il pezzo è arrivato alla barriera d'ingresso (I1 OFF) il motore del nastro trasportatore si ferma (Q2 OFF). Il PLC imposta a TRUE un merker di stato per segnalare al robot che il pezzo è pronto per essere prelevato;
6. quando il PLC che controlla il robot trova TRUE questo merker di stato, comanda al robot di prelevare il pezzo lavorato per portarlo verso il deposito;
7. a questo punto la catena sequenziale ricomincia dal punto 1.

Anche in questo caso, il **diagramma funzionale** del processo di lavorazione ci consente di descrivere in modo più schematico e chiaro l'evoluzione degli stati della macchina punzonatrice:



Partendo da questo diagramma funzionale è stato agevole scrivere il programma di controllo della macchina punzonatrice in **linguaggio KOP** utilizzando ancora la **metodologia batch (sequenza di passi)**.

## Robot cilindrico

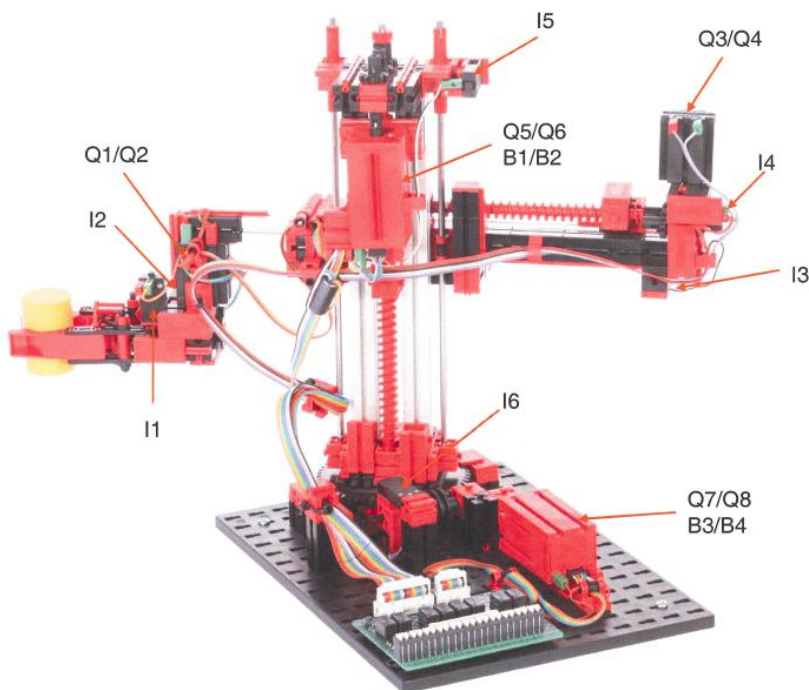


Figura 5

Questo robot (in figura 5 il modello in scala della **Fischertechnik** impiegato per lo sviluppo del programma) consente di muovere nello spazio l'end effector che manipola il pezzo in lavorazione.

L'end effector è montato all'estremità di un braccio che si può muovere sia orizzontalmente sia verticalmente e che può ruotare intorno ad un asse.

È costituito da:

- 4 motori bidirezionali
- 2 encoder in quadratura
- 2 encoder meccanici
- 4 contatti di finecorsa (NO)

### Specifiche tecniche

I dati che seguono sono stati ricavati sperimentalmente perché la documentazione Fischertechnik fornita insieme al modello è piuttosto scarna e non fornisce le specifiche degli encoder in quadratura montati sui motori dell'asse Y e del rotore, né quelle degli encoder meccanici del braccio (asse X) e della pinza montata sull'end effector.

**Braccio** (asse X): elongazione massima 90mm. Encoder meccanico con 4 camme e un pulsante (NO): 80 impulsi dal finecorsa (riferimento 0 mm) al massimo allungamento (90 mm). Risoluzione: 1,125 mm.

**Asse Y:** (orientato verso il basso) lunghezza corsa 150 mm. Encoder incrementale in quadratura: 2250 impulsi dal finecorsa superiore (riferimento 0 mm) al massimo abbassamento (150 mm). Risoluzione:  $6,7 \cdot 10^{-2}$  mm (15 impulsi/mm).

**Rotore:** rotazione massima 335°. Encoder incrementale in quadratura: 4020 impulsi dal finecorsa (riferimento 0°) alla rotazione massima (335°). Risoluzione: 5' ( $8,3 \cdot 10^{-2}$  gradi, 12 impulsi/grado).

**Pinza** (end effector): diametro con apertura massima 57 mm. Diametro con pinza chiusa 15 mm. Finecorsa attivato quando la pinza è completamente aperta. Encoder meccanico: 14 impulsi, dalla massima apertura (0) alla chiusura (14). Risoluzione: 3 mm.

Per controllare questo robot abbiamo impiegato un PLC **Siemens S7-1214C AC/DC/RLY** che dispone di 14 ingressi, 10 uscite digitali (a relè), 6 contatori veloci (**HSC**) oltre a una connessione di rete Profinet.

Le connessioni elettriche tra ingressi/uscite del PLC e del modello Fischertechnik controllato sono elencate nella seguente **tabella delle associazioni**:

Pin modello	Funzione	I/O PLC	I/O modello
1	Alimentazione (+) attuatori	L+	24V DC
2	Alimentazione (+) sensori	L+	24V DC
3	Alimentazione (-)	M	0V
4	Alimentazione (-)	M	0V
5	Home pinza (NO) (HSC1 reset)	I0.3	I1
6	Clock pinza (HSC1 clock)	I0.0	I2
7	Home braccio (NO) (HSC2 reset)	I0.1	I3
8	Clock braccio (HSC2 clock)	I0.2	I4
9	Home asse Y (NO) - HSC4 (reset)	I0.5	I5
10	Home rotore (NO) - HSC5 (reset)	I1.2	I6
11	A - Encoder asse Y - HSC4 (canale A)	I0.6	B1
12	B - Encoder asse Y - HSC4 (canale B)	I0.7	B2
13	A - Encoder rotore - HSC5 (canale A)	I1.0	B3
14	B - Encoder rotore - HSC5 (canale B)	I1.1	B4
15	Apri pinza	Q0.1	Q1 (M1)
16	Chiudi pinza	Q0.2	Q2 (M1)
17	Braccio avanti	Q0.3	Q3 (M2)
18	Braccio indietro	Q0.4	Q4 (M2)
19	Asse Y giù	Q0.5	Q5 (M3)
20	Asse Y su	Q0.6	Q6 (M3)
21	Ruota in senso orario	Q0.7	Q7 (M4)
22	Ruota in senso antiorario	Q1.0	Q8 (M4)

### Connessione degli encoder al PLC

Abbiamo collegato i canali A e B dell'encoder incrementale dell'**asse Y** ai corrispondenti ingressi (I0.6, I0.7) del contatore veloce **HSC4** del PLC. Il finecorsa dello stesso asse è collegato all'ingresso di reset (I0.5). Il valore del conteggio è disponibile come doppia word (32 bit) nella locazione di memoria ID1012 ("Conteggio encoder asse Y").



I canali A e B dell'encoder incrementale del **rotore** li abbiamo collegati ai corrispondenti ingressi (I1.0, I1.1) del contatore veloce **HSC5** del PLC. Il finecorsa del rotore (0°) è collegato all'ingresso di reset (I1.2). Il valore del conteggio è disponibile come doppia word (32 bit) nella locazione di memoria ID1016 ("Conteggio encoder Rotore").

Entrambi questi contatori veloci sono configurati come contatori A/B che utilizzano l'ingresso di resettaggio esterno (costituito dai finecorsa).

Gli impulsi di clock dell'encoder meccanico del **braccio** li abbiamo contati mediante il contatore monofase veloce, **HSC2**. Abbiamo collegato l'uscita dell'encoder all'ingresso di clock del contatore (I0.2) e il finecorsa dello stesso asse all'ingresso di reset (I0.1). Il valore del conteggio è disponibile come doppia word (32 bit) nella locazione di memoria ID1004 ("Conteggio encoder Braccio").

Il conteggio degli impulsi di clock dell'encoder meccanico della pinza l'abbiamo invece effettuato con un normale contatore software Up/Down (CTUD), con il finecorsa collegato all'ingresso di reset. Come ingressi digitali sono stati opportunamente scelti I0.0 (clock) e I0.3 (reset) che coincidono con quelli dei rispettivi ingressi del contatore veloce HSC1. In questo modo sarà possibile gestire la posizione della pinza anche utilizzando questo contatore (se abilitato).

Per evitare conteggi spuri dovuti ai rimbalzi dei contatti degli encoder meccanici del braccio e della pinza, abbiamo scelto di applicare ai corrispondenti **filtri digitali** degli ingressi del PLC la costante di tempo di default (6,4 ms).

Per gli ingressi digitali collegati ai segnali di clock degli encoder incrementali, invece, considerando la frequenza relativamente alta degli impulsi di clock, per non falsare il valore del conteggio, come costante di tempo di questi filtri abbiamo scelto un valore molto più piccolo (0,1 ms).

### Constatazioni sperimentali

Dopo alcune prove abbiamo osservato che la pinza, a causa della scarsa precisione meccanica, non sempre riusciva ad afferrare con sicurezza il pezzo da trasportare. Pertanto l'abbiamo sostituita con un semplice elettromagnete (considerato che i manufatti da trasportare sono di origine ferrosa).

L'alimentazione dell'elettromagnete<sup>1</sup> è stata connessa all'uscita Q0.2 del PLC (ex "Chiudi pinza"), mentre gli ingressi I0.0 ("Clock pinza") e I0.3 ("Home pinza") saranno ignorati dal programma di gestione.

---

<sup>1</sup> Questo progetto è solo un esempio. Sappiamo che nella realtà industriale, per trasportare materiali ferrosi, normalmente s'impiegano dei magneti permanenti. Questa soluzione aumenta la sicurezza operativa perché in caso di caduta della tensione di alimentazione il carico non



### API (Application Programming Interface)

Per gestire con semplicità (ed ad alto livello) la movimentazione degli assi del robot, abbiamo pensato di scrivere prima un insieme di funzioni di interfaccia che ci consentissero di interagire (a basso livello) con l'hardware del robot (**firmware**).

#### Braccio

Se all'ingresso "Lunghezza" di questo FB forniamo il valore della coordinata x ( $0 \div 80$  mm dall'origine) il braccio si allungherà o contrarrà fino a raggiungerla.

#### AsseY

L'ingresso "Altezza" di questo FB determina la coordinata Y ( $0 \div 150$  mm dall'origine) del braccio. L'asse è orientato verso il basso.

#### Rotore

Questo FB farà ruotare il braccio del robot di tanti gradi quanti quelli specificati all'ingresso "Angolo" ( $0 \div 335^\circ$  rispetto all'origine).

#### Pinza (non usata in questo versione del progetto)

Fornendo all'ingresso "Chiusura" del FB un numero compreso tra 0 (massima apertura) a 14 (massima chiusura), la pinza si aprirà o chiuderà fino a raggiungere quel grado di chiusura.

Le prime tre funzioni agiscono sui motori degli assi corrispondenti, la quarta sui motori della pinza dell'end effector.

I contatori veloci (HSC) impiegati nelle prime tre funzioni generano un **interrupt** quando il valore del loro conteggio raggiunge il valore di riferimento. Le rispettive routine di servizio (OB) arrestano semplicemente i motori dell'asse interessato.

Anche tutti i finecorsa del robot generano un **interrupt** quando vengono attivati. La relativa routine di servizio (OB) arresta il motore dell'asse (o della pinza) che l'ha generato e azzerà il contatore della posizione.

Poiché queste funzioni contengono diversi calcoli matematici (ad esempio per convertire i conteggi dei contatori in coordinate fisiche e viceversa), ci è sembrato più agevole scriverle in **linguaggio SCL** (ST).

### Funzioni di controllo

Impiegando queste "primitive" abbiamo potuto scrivere delle funzioni "più evolute" (ad alto livello):

### Move

**Posiziona l'end effector** del robot nel punto dello spazio specificato nell'ingresso "Coordinates" ( $x, y, \theta$ ). Il valore di ritorno vale TRUE quando è stato raggiunto il punto desiderato.

### Grab

Abbassa (o alza) il braccio ed **afferra** il pezzo. Ritorna TRUE quando il pezzo è stato afferrato.

### Drop

Abbassa (o alza) il braccio e **rilascia** il pezzo. Ritorna TRUE quando il pezzo è stato rilasciato.

### Home

Riporta tutti gli assi nella loro **posizione di riferimento** (coordinate 0 mm, 0 mm, 0°).

### Electromagnet

Se il suo ingresso è TRUE attiva l'elettromagnete dell'end effector, altrimenti lo spegne.

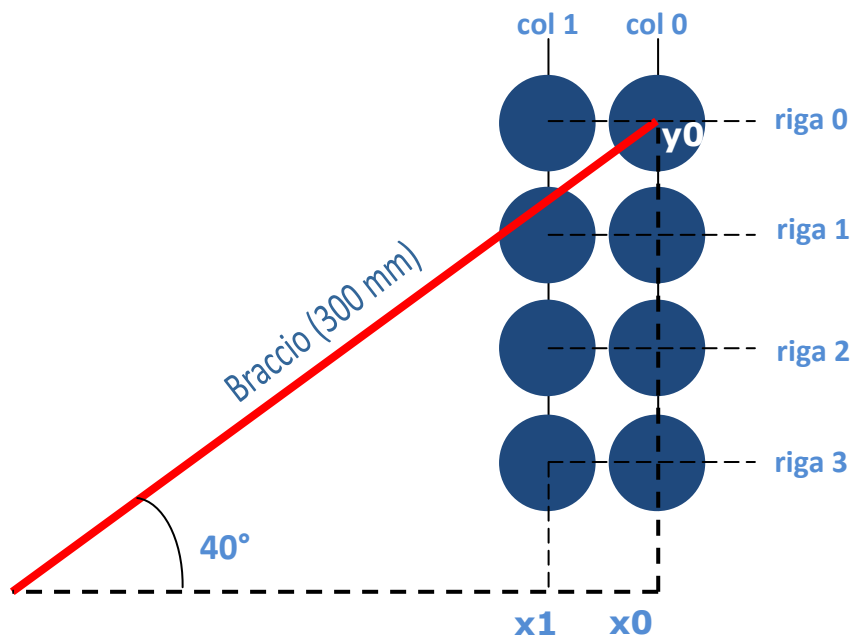
### Coordinates

Ritorna i valori delle **coordinate attuali dell'end effector** del robot, rispettivamente, del braccio, dell'asse Y e del rotore.

### Piece coordinates

Questa funzione, dato in ingresso il numero progressivo del pezzo prodotto, ritorna le coordinate polari  $x$  e  $\theta$  dove depositarlo.

I pezzi lavorati saranno rilasciati nel deposito, disposti su 2 colonne di 4 pezzi ciascuna, a formare un lotto di 8 pezzi, come rappresentato nella figura qui a fianco.



## I.I.S. ITALO CALVINO - Genova

Le coordinate polari ( $r, \theta$ ) del primo pezzo da depositare, quello più distante (riga 0, colonna 0), valgono 300 mm e  $40^\circ$ . La distanza è stata misurata dall'asse di rotazione del robot, mentre l'angolo è riferito all'asse delle  $X^2$ .

**La lunghezza effettiva del braccio** del robot è stata ricavata sommando la parte fissa (210 mm) a quella mobile ( $0 \div 90$  mm).

Il **diametro** dei pezzi è di **30 mm**. La **distanza** tra i centri di due pezzi adiacenti è di **35 mm**.

Partendo da questi dati e dal numero progressivo  $n$  del pezzo prodotto, la funzione calcola le coordinate polari che il robot dovrà raggiungere per depositarlo al posto giusto.

Abbiamo arbitrariamente numerato da 0 a 3 i pezzi della colonna 0 (partendo dall'alto) e da 4 a 7 (sempre partendo dall'alto) quelli della colonna 1.

Se  $n$  è il numero del pezzo prodotto, per individuare in quale posizione (da 0 a 7) del lotto va collocato, è sufficiente trovare il resto della divisione tra  $n$  e 8 (numero dei pezzi del lotto):

$$\text{pos} := n \text{ MOD } 8;$$

da questo numero si può facilmente individuare la colonna  $\text{col}$  dove andrà posato il pezzo:

$$\text{col} := \text{pos} / 4;$$

Poiché  $\text{pos}$  è una variabile  $\text{Int}$ , anche il risultato di questo rapporto sarà di tipo  $\text{Int}$ , ovvero  $\text{col}$  potrà valere solo 0 oppure 1.

Il valore della riga ( $0 \div 3$ ) è dato da:

$$\text{riga} := \text{pos} - (\text{col} * 4);$$

Ora che sappiamo dove sarà disposto il pezzo all'interno del lotto, occorre convertire questa posizione nelle corrispondenti coordinate polari del robot.

Osservando attentamente la figura sopra si può ipotizzare che il primo pezzo (riga 0, col 0) sia posto all'estremità superiore del cateto di un triangolo rettangolo la cui ipotenusa è lunga come la massima estensione del braccio (300 mm) e l'angolo opposto vale  $40^\circ$ .

---

<sup>2</sup> In realtà, quando il braccio del robot si trova nella posizione della linea rossa, il suo effettivo angolo di rotazione è  $220^\circ$  ( $180^\circ + 40^\circ$ ), ma per comodità di rappresentazione abbiamo deciso di riferirlo al primo quadrante.

## I.I.S. ITALO CALVINO - Genova

Pertanto il valore di  $x_0$  è dato da:

$$x_0 := \text{braccio}_{\max} * \text{COS}(40^\circ);$$

Questo valore rappresenta la **coordinata X della prima colonna**. La **coordinata Y della riga 0**,  $y_0$ , è data da:

$$y_0 := \text{braccio}_{\max} * \text{SIN}(40^\circ);$$

Considerato che i centri degli altri pezzi distano tra loro 35 mm, a questo punto è facile dedurre le rispettive coordinate cartesiane. Ad esempio la **coordinata y di un pezzo su una riga** vale:

$$y := y_0 - \text{riga} * 35;$$

mentre la **coordinata X della seconda colonna**,  $x_1$ , è semplicemente:

$$x_1 := x_0 - 35;$$

Applicando lo stesso ragionamento che abbiamo fatto per il primo pezzo a tutti gli altri, ovvero ipotizzando che la loro posizione coincida con l'estremità superiore del cateto di un triangolo rettangolo, è facile convertire le loro coordinate cartesiane nelle corrispondenti coordinate polari da passare al robot. Se ad esempio il pezzo si trova nella colonna 1 in  $(x_1, y)$  si ha:

$$\text{rotazione} := \text{ATAN}(y / x_1);$$

$$\text{braccio} := x_1 / \text{COS}(\text{rotazione});$$

Dove le funzioni ATAN e COS calcolano, rispettivamente, l'arcotangente e il coseno del loro argomento.

Il valore di *rotazione* è in radianti, mentre per il nostro robot occorre esprimerlo in gradi:

$$\#rotazione := \#rotazione * 180.0 / \text{REAL}\#3.1415 + 180.0;$$

Si noti che al risultato abbiamo aggiunto  $180^\circ$  per ottenere la rotazione corretta del robot.

Anche la coordinata X del braccio deve essere espressa al netto del suo valore minimo (210 mm):

$$\text{braccio} := \text{braccio} - 210.0;$$

La terza coordinata da fornire al robot è l'altezza alla quale deve depositare il pezzo, che nel caso del nostro esempio vale sempre 130 mm.

## Coordinamento dei PLC via rete Profinet

I PLC che controllano la macchine dell'impianto, il pannello operatore e il gateway IOT2020 comunicano fra loro attraverso una rete **Profinet** alla quale sono fisicamente connessi mediante uno switch Siemens SCALANCE XB005.

L'indirizzo IP della rete è 192.168.0.0/24, quello dei singoli dispositivi è rappresentato in figura 6.

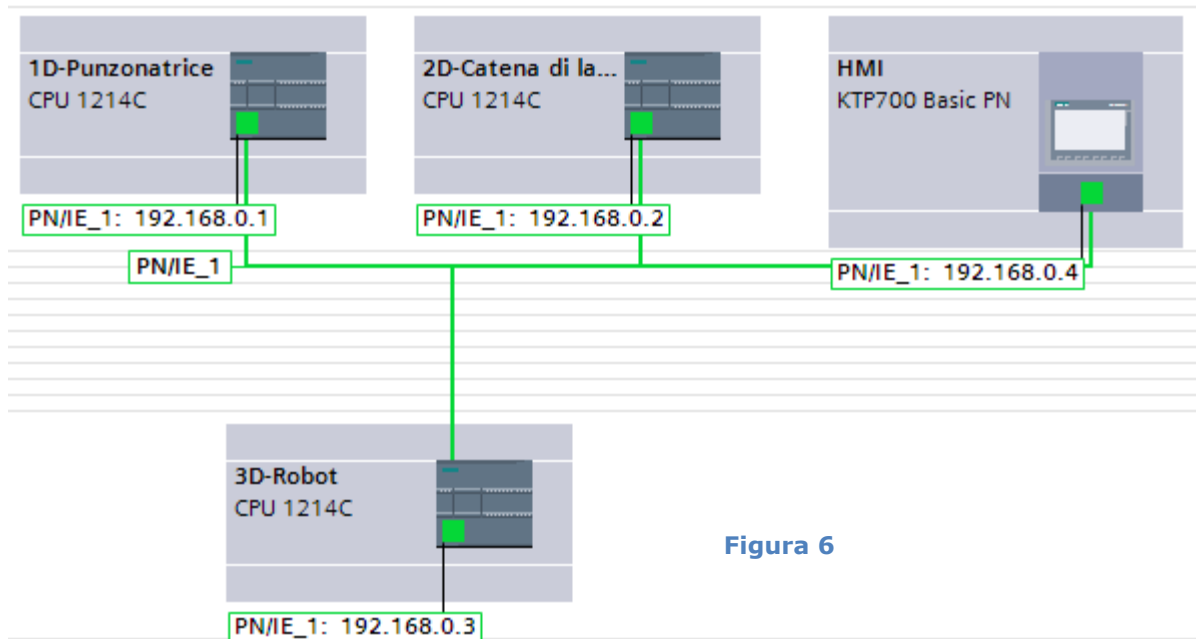


Figura 6

L'indirizzo del gateway (qui non rappresentato) è 192.168.0.12).

Il **coordinamento** delle tre macchine dell'impianto avviene mediante la comunicazione, via rete, del valore di determinate variabili di stato. A questa funzione provvedono le funzioni **GET/PUT** di **TIA Portal** che, per comunicare, utilizzano il protocollo di rete **Siemens S7**.

In particolare:

- quando la **catena di lavorazione** ha terminato di lavorare un pezzo, il PLC che la controlla imposta a TRUE (mediante la funzione **PUT**) la variabile di stato "PN - Pezzo pronto catena di lavorazione" nel PLC che controlla il **robot**;
- quando il **robot** preleva il pezzo dalla catena di lavorazione, il PLC che lo controlla imposta a TRUE per 1" (mediante la propria funzione **PUT**) la variabile di stato "PN - Pezzo prelevato robot" nel PLC che controlla la **catena di lavorazione**;
- quando il **robot**, dopo avere depositato il pezzo sul nastro trasportatore della punzonatrice, ha debitamente allontanato il suo end effector, il PLC che lo controlla imposta a TRUE (mediante la funzione **PUT**) la variabile di stato "PN - Pezzo libero" nel PLC che controlla la **punzonatrice**;

- quando il pezzo lavorato dalla **punzonatrice** è pronto per essere prelevato dal robot, il PLC che la controlla imposta a TRUE la propria variabile di stato "PN - Pezzo punzonato". Il PLC che controlla il **robot** legge questa variabile mediante la propria funzione **GET** e procede al prelevamento del pezzo.

La figura 7 schematizza lo scambio delle informazioni via rete descritte sopra che consentono il coordinamento dei tre PLC.

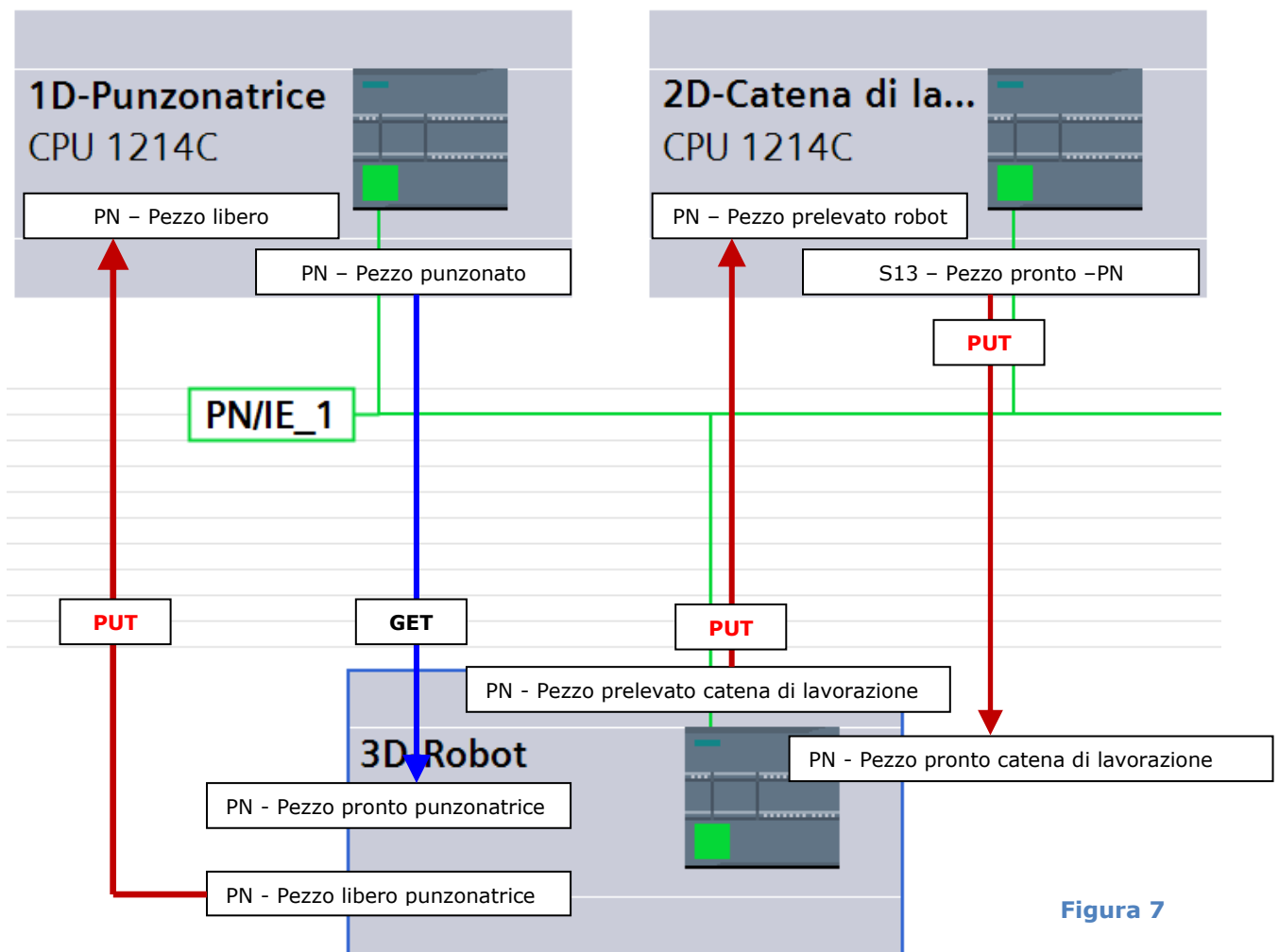


Figura 7

### Specifiche di funzionamento

Quando un pezzo ha attraversato la catena di lavorazione ed è arrivato al termine del nastro trasportatore d'uscita, il robot deve afferrarlo, trasportarlo sulla macchina punzonatrice e rilasciarlo sul corrispondente nastro trasportatore.

Una volta punzonato, il robot riafferra il pezzo e lo trasporta nel deposito dei pezzi lavorati. Successivamente ritorna alla catena di lavorazione dove rimane in attesa di un nuovo prelevamento.

Il comportamento di questo robot può essere descritto da una semplice catena sequenziale.

## I.I.S. ITALO CALVINO - Genova

All'accensione della macchina tutti gli assi si portano nell'origine del sistema di riferimento del robot (home) in modo da azzerare i rispettivi contatori di posizione (tutti i finecorsa attivati);

1. il robot sposta il proprio end effector al termine del nastro trasportatore d'uscita della catena di lavorazione, pronto a prelevare il pezzo lavorato;
2. quando il pezzo è pronto per essere prelevato, il PLC che controlla la catena di lavorazione lo segnala a questo PLC (via rete Profinet) impostando a TRUE il merker locale "PN - Pezzo pronto catena di lavorazione". A quel punto il braccio del robot scende all'altezza del pezzo, e l'end effector lo afferra;
3. il robot porta il pezzo prelevato dalla catena di lavorazione al nastro trasportatore della punzonatrice. Simultaneamente (sempre via rete, mediante la funzione PUT) il controller del robot indica al PLC che controlla la catena di lavorazione che il pezzo è stato prelevato (impostando a TRUE per 1 secondo il merker "Pezzo prelevato catena di lavorazione");
4. quando il pezzo si trova sospeso sopra il nastro trasportatore della punzonatrice, il robot abbassa il suo end effector per depositare delicatamente il pezzo sul nastro;
5. l'end effector del robot risale un po', ad una altezza di sicurezza, per non intralciare il movimento del pezzo;
6. il pezzo liberato dal robot adesso è pronto per essere lavorato dalla macchina punzonatrice.

Questo stato ("PN - Pezzo libero punzonatrice" TRUE) è comunicato via rete Profinet, mediante la funzione PUT, al PLC che controlla la punzonatrice.

Quando il pezzo punzonato è pronto per essere prelevato (variabile di stato letta, via rete Profinet mediante la funzione GET, nel PLC che controlla la punzonatrice), l'end effector del robot si abbassa un po' e lo afferra;

7. il robot trasporta il pezzo prelevato dalla macchina punzonatrice fino al deposito dei pezzi lavorati;
8. quando il pezzo arriva all'altezza del deposito l'end effector del robot si abbassa e rilascia il pezzo.

Quindi il ciclo ricomincia dal punto 1.

## 2. SUPERVISIONE LOCALE

### Pannello operatore (HMI)

Per implementare questa funzionalità abbiamo impiegato un HMI **Siemens KTP700 Basic** che dispone di un touch-screen a colori da 7" ed è connesso via rete Profinet ai PLC che controllano le singole macchine.

Grazie a questo terminale grafico possiamo intervenire, in modo semplice e intuitivo, sulle singole macchine per effettuare interventi di **manutenzione** (quali, per esempio, la sostituzione degli utensili), sospenderne il funzionamento (**stop**) e riprenderlo (**resume**) dal punto in cui era stato interrotto.

In caso di **emergenza**, abbiamo previsto anche un arresto di sicurezza (**fail-safe**) che, mediante un relè esterno, interrompe l'alimentazione comune delle bobine dei contattori di **tutti i PLC<sup>3</sup>** dell'impianto e, per ulteriore sicurezza, ne disattiva anche le uscite via software.

Una volta ripristinate le condizioni di sicurezza, per il riavvio del ciclo di lavorazione occorre prima intervenire fisicamente su un pulsante elettromeccanico di AVVIO che ripristina l'alimentazione alle bobine dei contattori d'uscita. Successivamente occorre ripristinare lo stato delle tre macchine mediante il pannello operatore.

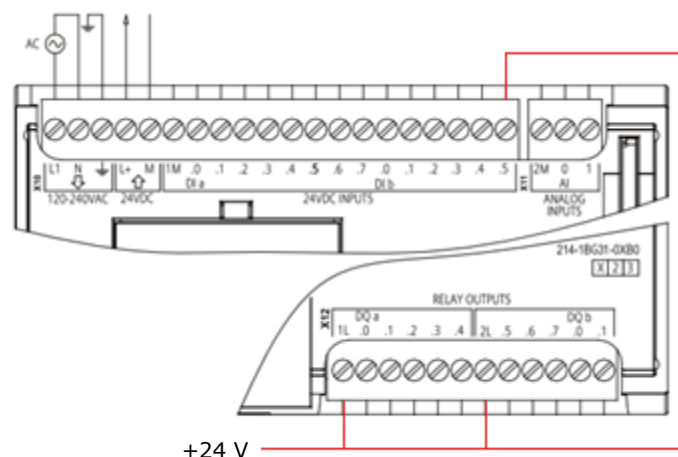
Per questo **HMI** abbiamo previsto **quattro pagine** interattive.

1. la pagina principale, **Home** (figura 8), ci consente mediante dei semplici indicatori luminosi, di verificare in un colpo d'occhio lo stato di funzionamento (attiva/disattiva) di tutte le macchine dell'impianto. Il colore verde indica che la macchina funziona regolarmente, il rosso lampeggiante che è ferma<sup>4</sup>.

Tre pulsanti ci permettono di accedere ai sinottici delle singole macchine attraverso i quali sarà possibile visualizzare lo stato di sensori e attuatori e monitorare lo stato di avanzamento della lavorazione dei manufatti.

Due indicatori numerici indicano, rispettivamente, il numero di pezzi che sono attualmente in lavorazione nell'impianto e quelli prodotti.

**Il tasto funzione F8** attiva lo stato di **emergenza** descritto sopra. In caso di pericolo tutte le macchine potranno essere immediatamente arrestate anche premendo questo pulsante.



<sup>3</sup> Per lo standard EN 60204-1 (che regola i requisiti per l'avvio di un dispositivo) non è possibile integrare i pulsanti di arresto solo in un pannello HMI. Per l'arresto e l'avvio dell'impianto saranno quindi previsti anche dei normali pulsanti elettromeccanici.

<sup>4</sup> L'indicatore del robot prevede anche un terzo stato: la modalità di controllo manuale. In questo caso l'indicatore lampeggerà con colore giallo



Questa funzionalità è stata realizzata collegando il comune dell'alimentazione a 24 V delle uscite del PLC all'ingresso I1.5 (Power ON) di ogni PLC. Quando questa alimentazione si interrompe, l'assenza di segnale a questo ingresso provoca la disattivazione automatica di tutte le uscite dei tre PLC. In questo modo ci assicuriamo che al ripristino dell'alimentazione non si creino situazioni di pericolo per gli operatori a causa dell'azionamento improvviso degli attuatori dell'impianto.

La segnalazione dello **stato di emergenza** avviene mediante il lampeggio del segnale di pericolo sopra il logo dell'azienda nella pagina Home del pannello HMI.

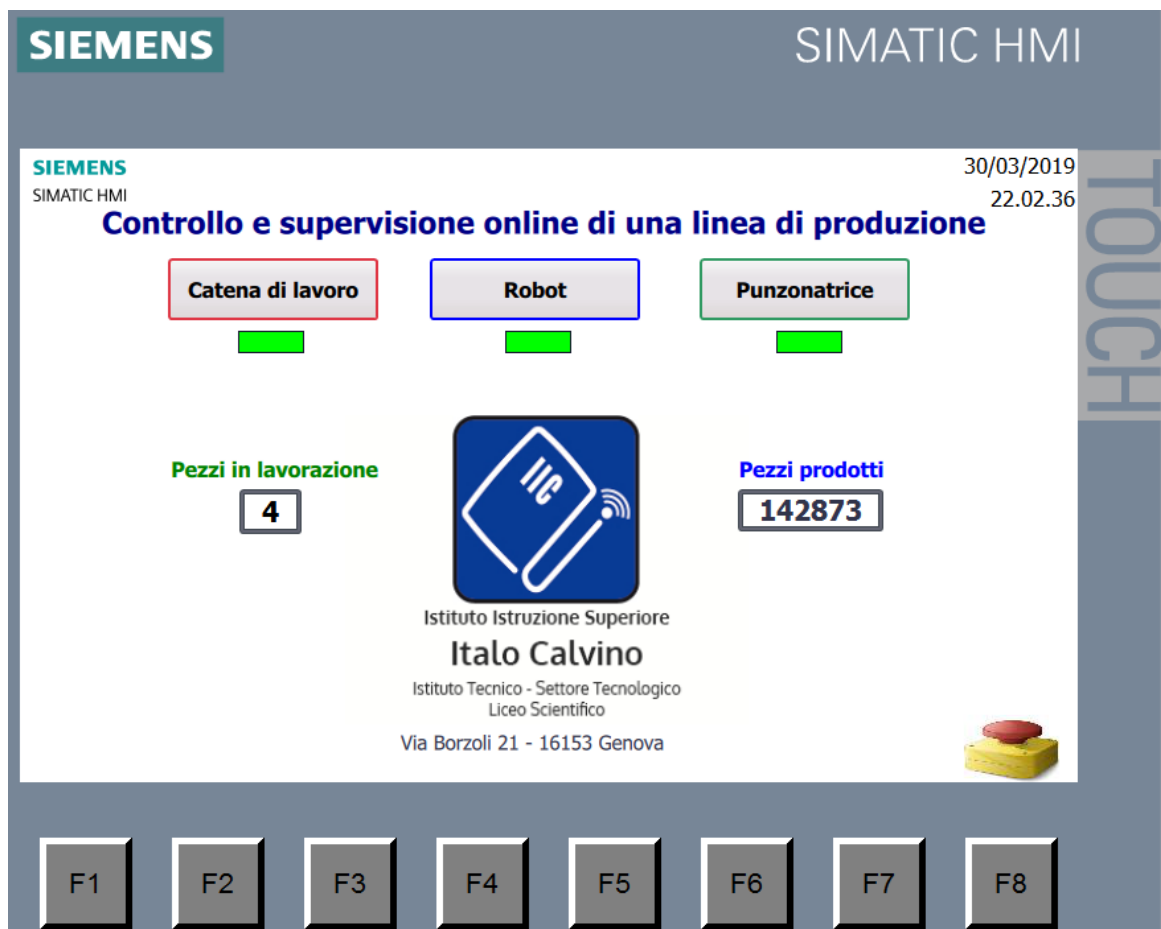


Figura 8

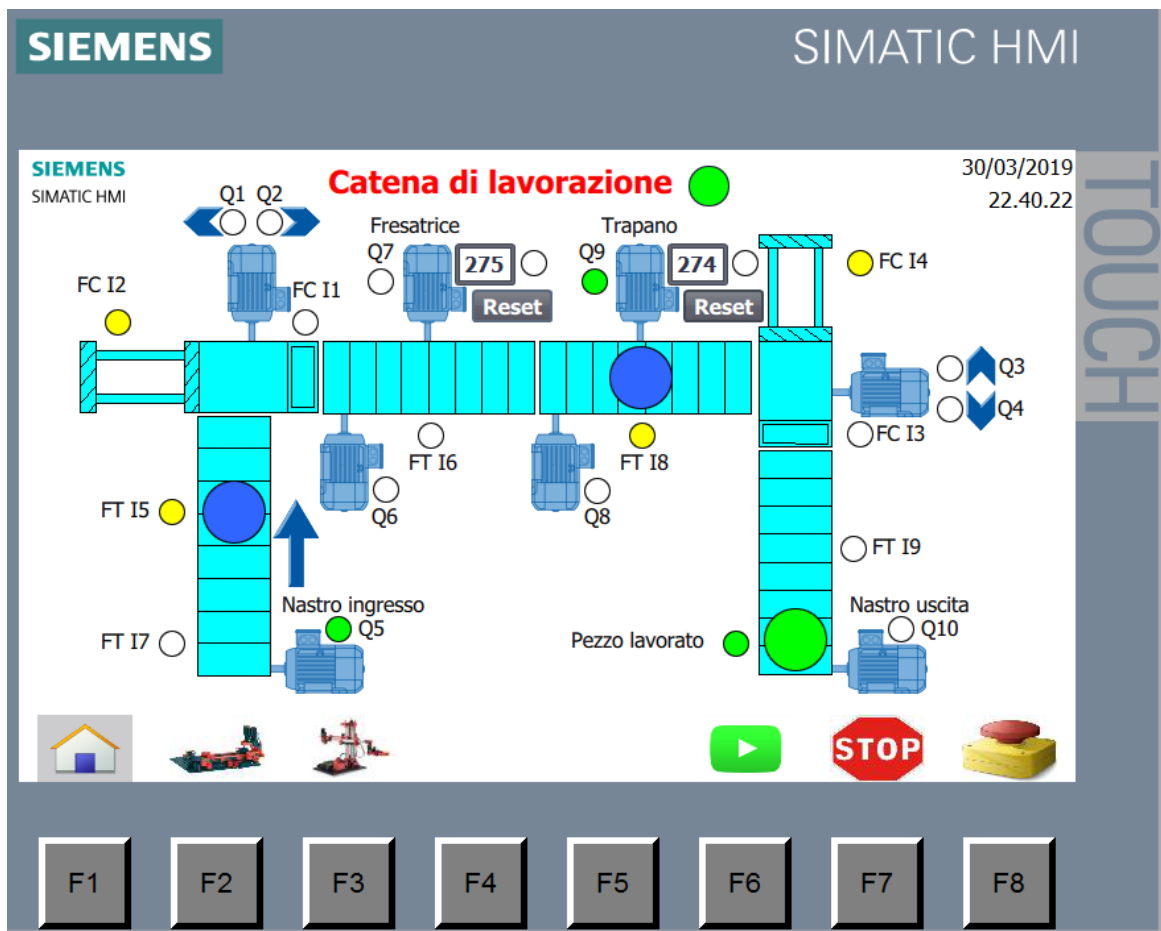
L'indicazione del numero di **pezzi in lavorazione** nell'impianto è stata realizzata aggiungendo un contatore Up/Down (CTUD) al programma del PLC che controlla la catena di lavorazione (segmento 31). L'incremento del conteggio avviene quando il pezzo (depositato sul nastro trasportatore d'ingresso) inizia il ciclo di lavorazione. Si ha un decremento quando il robot ha rilasciato il pezzo nel deposito. Quest'ultimo evento viene comunicato a questo PLC via rete Profinet.

Per indicare il numero di **pezzi prodotti** abbiamo aggiunto un contatore in avanti (CTU) al programma che controlla il robot (segmento 10). Il contatore si incrementa quando il pezzo lavorato viene rilasciato nel deposito.

2. La pagina **Catena di lavorazione** presenta il sinottico di questa macchina (figura 9). Vi sono rappresentati i nastri trasportatori, gli slider, i motori, gli indicatori di stato dei sensori e dei motori, le posizioni dei pezzi in lavorazione, gli indicatori del numero di fresature e forature effettuate,...

I cerchi piccoli rappresentano gli indicatori di stato dei sensori (finecorsa e fototransistor) e degli attuatori (motori). Quando sono disattivati sono tutti di colore bianco. Quando sono attivati, gli indicatori di stato dei sensori diventano gialli, quelli dei motori, verdi.

Figura 9



I cerchi grandi di colore blu rappresentano la posizione dei pezzi in lavorazione. Quando un pezzo arriva alla fine della catena di lavorazione ed è in attesa di essere prelevato dal robot, il cerchio che lo rappresenta lampeggia (verde/blu).

Quando i nastri trasportatori e gli slider sono in funzione, delle frecce indicano la direzione del loro movimento.

Accanto alla fresatrice e al trapano sono presenti degli indicatori numerici che visualizzano il numero di fresature e forature effettuate. Quando il conteggio arriva ad un valore prefissato per ciascun tipo di utensile, la spia di segnalazione accanto al contatore lampeggia (bianco/rosso) e la

macchina si arresta automaticamente (stato di STOP) per consentire all'operatore di sostituire l'utensile usurato.

Per implementare queste due funzionalità, abbiamo introdotto due contatori CTU nel programma del PLC che controlla la catena di lavorazione (segmenti 29 e 30). Quando questi contatori raggiungono il numero di lavorazioni massimo previsto per quell'utensile, la loro uscita Q porta nello stato di STOP la macchina.

Una volta effettuata la manutenzione, per riprendere la normale attività dal punto in cui era stata sospesa, occorre prima resettare il relativo contatore e poi premere il tasto funzione RESUME (F6) del pannello operatore.

I **tasti funzioni F1, F2 e F3** consentono la navigazione tra le varie pagine del pannello operatore. F1 consente di accedere alla pagina principale, Home, F2 al sinottico della macchina punzonatrice e F3 a quella del robot, come raffigurato dalle corrispondenti icone subito sopra ai pulsanti.

Il tasto funzione **F7 (STOP)**, arresta la macchina nello stato in cui si trova disattivando tutte le uscite.

Il tasto **F8 (EMERGENZA)** si comporta come quello omonimo presente nella pagina Home, descritto nell'introduzione di questo capitolo. Inoltre, riporta la navigazione alla pagina principale.

Per ripristinare lo stato della macchina com'era al momento dell'arresto (anche per manutenzione) o prima dell'intervento di emergenza occorre premere il tasto funzione **F6 (RESUME)**.

Questa pagina consente anche di **visualizzare la posizione e lo stato di lavorazione dei pezzi** presenti nell'impianto. In figura 9 è rappresentato un esempio in cui nella catena di lavorazione sono presenti contemporaneamente tre pezzi. Uno (qui rappresentato dal cerchio verde) è in attesa di essere prelevato dal robot, il secondo è in lavorazione sotto al trapano (fototransistor I8 attivato e motore Q9 acceso), il terzo si sta muovendo verso il primo slider (motore Q5 acceso) ed è stato rilevato dal fototransistor I5.

Entrambi gli slider sono in posizione arretrata (finecorsa I2 e I4 attivati). Al termine della foratura l'indicatore numerico accanto al trapano indicherà 275.

La spia verde accanto al titolo della pagina indica che la macchina sta funzionando regolarmente. In caso di arresto, diventa rossa e lampeggia.

**N.B.:** Le funzionalità dei tasti F1, F6, F7 e F8 sono identiche a quelle delle pagine con i sinottici del robot e della macchina punzonatrice.

I tasti F2 e F3 degli altri sinottici hanno comunque la stessa funzione di navigazione tra le pagine delle altre macchine.

3. La pagina **Punzonatrice** consente di accedere al sinottico che rappresenta questa macchina (figura 10).

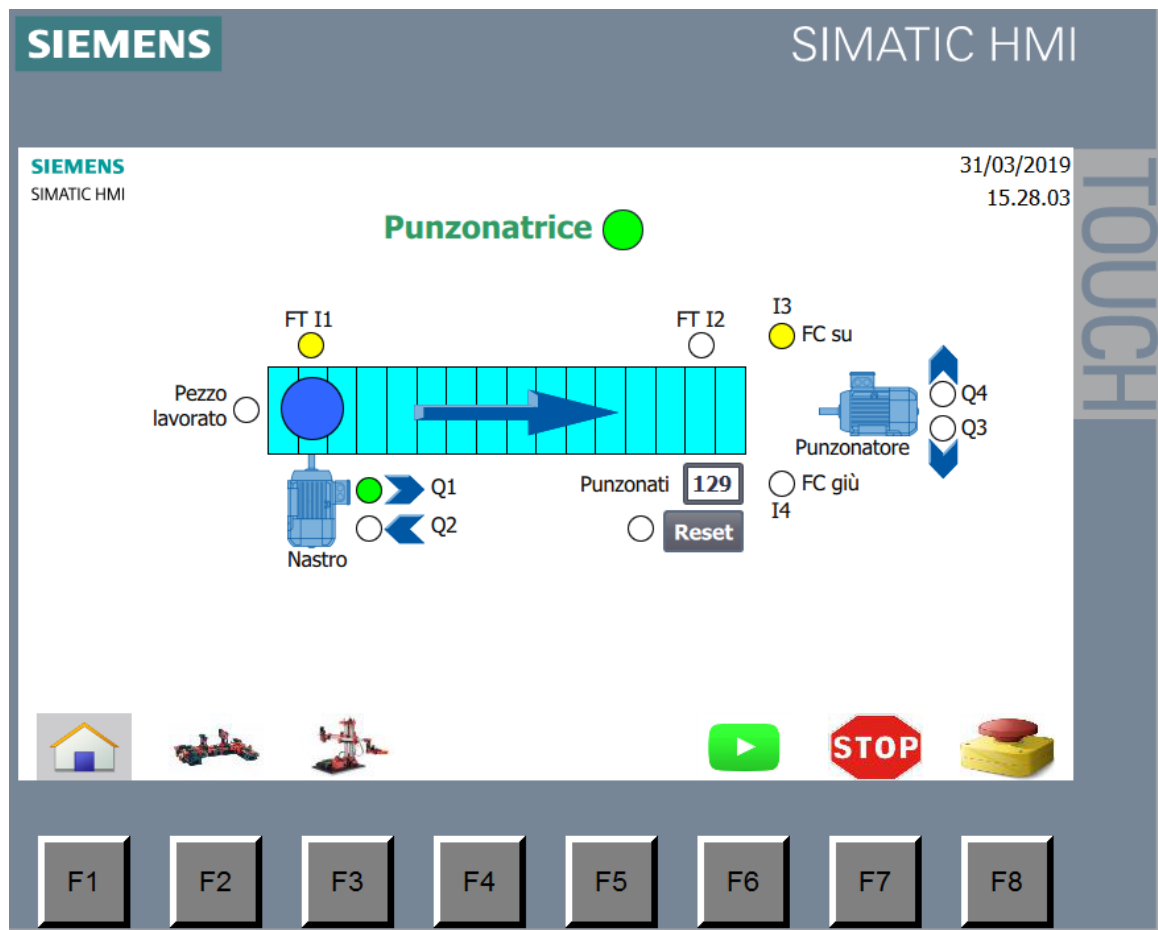


Figura 10

Nell'esempio in figura è rappresentato un pezzo che è stato rilevato dal fototransistor I1 all'ingresso del nastro trasportatore. Il motore che ne comanda il movimento lo fa spostare in avanti (spia Q1 accesa), verso l'utensile. La freccia sul nastro trasportatore indica la direzione del movimento.

Il motore raffigurato sulla destra è quello che comanda la discesa e la risalita del punzonatore.

Quando il conteggio dei pezzi punzonati raggiunge il valore massimo consentito per quell'utensile, la spia di segnalazione accanto al contatore lampeggia (bianco/rosso). Il pezzo viene riportato all'ingresso della macchina che subito dopo si arresta automaticamente (stato di STOP). A quel punto l'operatore può intervenire per sostituire l'utensile usurato.

Per aggiungere questa funzionalità al programma del PLC che comanda la macchina punzonatrice, abbiamo impiegato un contatore in avanti (CTU) che si incrementa di un'unità ad ogni punzonatura (segmento 13).

La richiesta di **manutenzione** può essere segnalata anche da una **lampada esterna**, collegata all'**uscita Q1.0** (Manutenzione segnalazione), che si accende a **intermittenza** (2 volte al secondo)<sup>5</sup>.

I tasti funzioni di questa pagina svolgono gli stessi compiti di quelli degli altri sinottici.

Se la spia verde accanto al titolo della pagina diventa rossa e lampeggia significa che la macchina è ferma (tutte le uscite del PLC che la controlla sono disattivate), altrimenti è verde.

4. Il sinottico della pagina **Robot** (figura 11) consente di visualizzare la posizione attuale dei tre assi del robot e lo stato del suo end effector (elettromagnete attivato oppure no).

Se un asse si sta muovendo una freccia ne indica la direzione. Come negli altri sinottici le spie verdi indicano i motori attivi (e il verso di rotazione). Le spie gialle, invece, l'attivazione dei finecorsa.

I tre display numerici (evidenziati da un bordo verde) visualizzano le coordinate attuali degli assi (in mm e gradi).

La spia sull'end effector diventa verde quando l'elettromagnete è attivo.

La spia accanto al titolo della pagina lampeggia di rosso se il robot è disattivato, di giallo se il robot è in modalità manuale oppure è verde fissa se il robot è controllato automaticamente (modalità di funzionamento normale).

---

<sup>5</sup> Per la catena di lavorazione non abbiamo previsto segnalatori esterni perché tutte le 10 uscite digitali del PLC sono utilizzate per controllare la macchina. Se richiesto, è comunque sempre possibile aggiungere al PLC un modulo DQ aggiuntivo (ad esempio il DQ8 x relè)

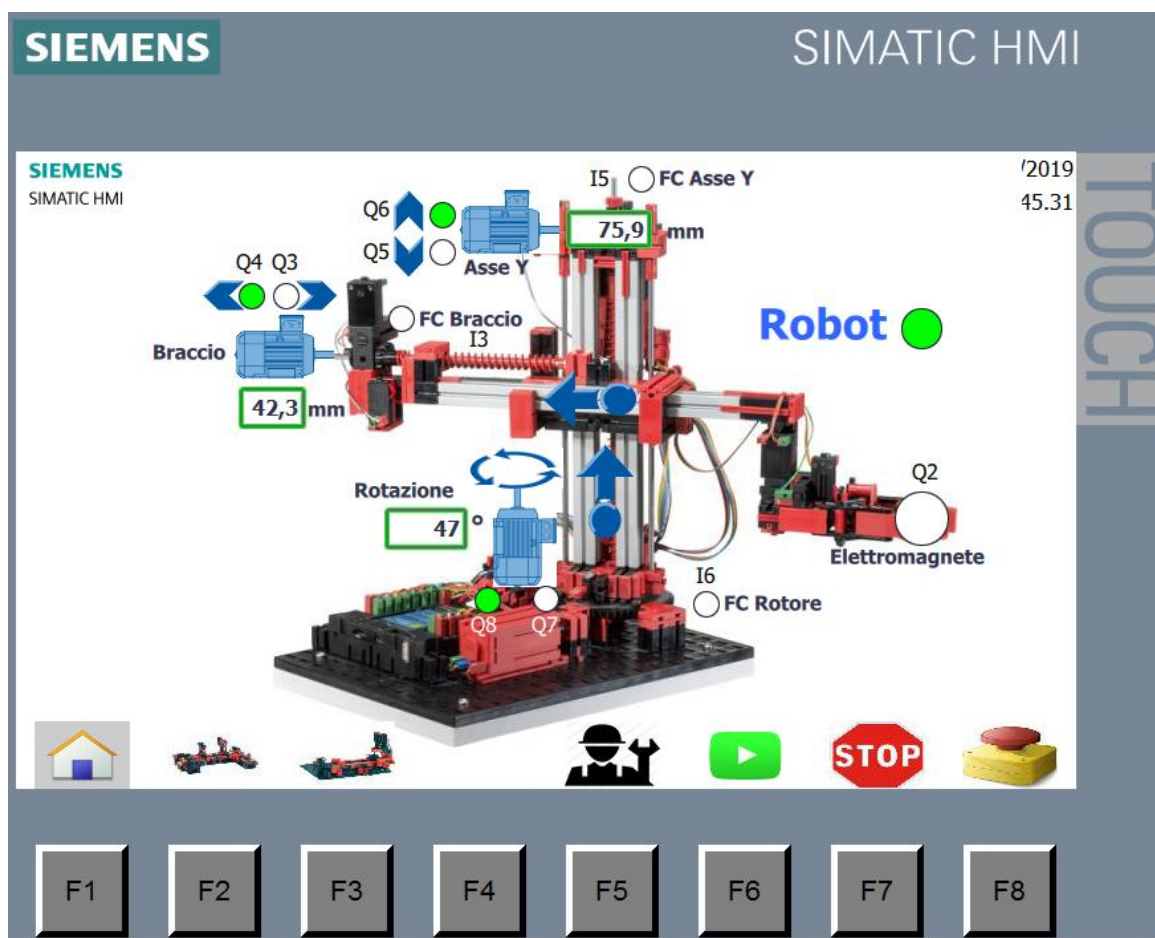


Figura 11

In figura 11 è rappresentato un esempio in cui il braccio del robot si sta contemporaneamente, contraendo (motore Q4), alzando (motore Q6) e ruotando in senso antiorario (motore Q8). Le direzioni dei movimenti sono evidenziate dalle rispettive frecce.

L'elettromagnete è disabilitato. Le coordinate attuali (x, y e  $\theta$ ) dell'end effector sono, rispettivamente, 42.3 mm, 75.9 mm e 47°. Ricordiamo che l'origine dell'asse Y è in alto, in corrispondenza del suo finecorsa.

Se si preme il tasto funzione **F5 (MANUALE)**, è possibile disattivare la modalità automatica e controllare i singoli assi del robot e il suo end effector manualmente.

In figura 12 è rappresentato un esempio di questa modalità. Si noti la spia gialla accesa accanto al nome della pagina per indicare all'operatore che il robot è in attesa di coordinate.

Quando si entra nella modalità manuale, accanto agli indicatori numerici che visualizzano le coordinate attuali degli assi, compaiono anche delle caselle numeriche (bordate di rosso, con nome SET). Qui, mediante il touch pad del

pannello operatore, è possibile immettere i valori delle coordinate desiderate.



Figura 12

Oltre alla posizione degli assi, in modalità manuale è possibile controllare, mediante un semplice interruttore, anche l'elettromagnete. Nell'esempio di figura abbiamo portato l'interruttore sullo schermo HMI nello stato ON e, come si può dedurre dall'accensione della relativa spia di segnalazione, l'elettromagnete dell'end effector si è attivato.



### 3. SUPERVISIONE REMOTA (IoT)

Come abbiamo accennato nella parte iniziale di questo documento, **il vero valore aggiunto** di questo progetto è la possibilità di replicare su un terminale remoto quale un PC, un tablet e addirittura su uno smartphone, la maggior parte delle funzionalità disponibili sul pannello HMI posto in prossimità dell'impianto.

Da remoto sarà anche possibile arrestare le singole macchine, controllare gli assi del robot, effettuare degli arresti di emergenza, acquisire altri dati da sensori intelligenti, monitorare e ottimizzare la produzione,...

Ovviamente tutte queste attività devono poter essere svolte solo da personale autorizzato, in modo protetto. Per questo saranno adottate le principali misure di sicurezza informatica quali l'autenticazione degli utenti e la cifratura delle informazioni.

In ogni caso, da remoto non sarà possibile intervenire direttamente sui PLC che condividono la rete Profinet dell'ambiente di produzione, ma l'accesso sarà mediato da firewall, router e dal **gateway Siemens IOT2020**.

#### Siemens IOT2020

Prodotto da Siemens, è un gateway **IoT** progettato e certificato per lavorare in ambienti industriali 24/7 (figura 13). Può essere impiegato per elaborare, analizzare e inviare dati ai più svariati tipi di dispositivi e di reti.

Basato sulla **CPU Intel Quark® x1000** e il sistema operativo open source **Yocto Linux**, dispone di un'interfaccia integrata **mPCIe** che consente di collegare molti dei moduli di espansione disponibili sul mercato (ad esempio per WLAN/Bluetooth).

Un'interfaccia **compatibile con Arduino Uno R3** e con l'**Arduino shield eco system** consente impiegare le stesse shield e di caricarvi la maggior parte degli sketch direttamente dall'IDE di Arduino.



Figura 13



Inoltre, poiché è in grado di utilizzare molti protocolli disponibili e aperti quali Modbus, **Profinet**, REST o MQTT, l'IOT2020 si può interfacciare via Ethernet, o mediante adattatore USB, ad hardware di terze parti come **PLC** e sensori di varie marche.

Può impiegare diversi linguaggi di programmazione ad alto livello come Java, C++ e Python. Inoltre possono anche essere installati **Node-RED**, MQTT e SQLITE3 per connettere e controllare periferiche locali e remote come, ad esempio, i sensori intelligenti.

Di seguito vedremo in dettaglio come abbiamo utilizzato l'ambiente di sviluppo grafico Node-RED sul gateway IOT2020 per realizzare l'interfaccia di supervisione remota del nostro impianto industriale.

### Node-RED



**Node-RED** è uno tra i più noti tool di **flow-based programming** per l'**Internet of Things**. Nasce con l'obiettivo di dare a tutti, anche a chi non è esperto di programmazione, la possibilità di collegare tra loro diversi dispositivi (con eventuali **Node-RED** relativi sensori ed attuatori). Inoltre dispone di API e servizi online per poter realizzare sistemi altamente integrati e complessi in modo semplice ed intuitivo.

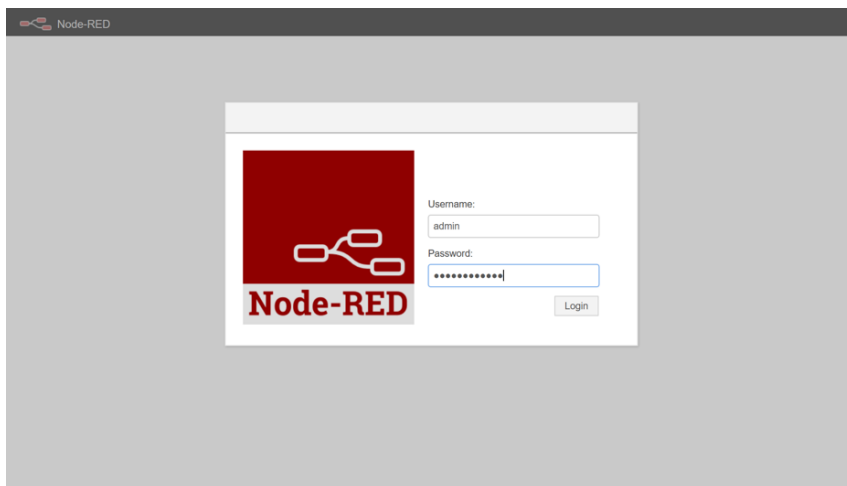
Dopo aver installato il Sistema Operativo Yocto-Linux sulla SD dell'IOT2020 e dopo averlo collegato alla rete, è possibile installarvi Node-RED attraverso una sessione SSH.

Una volta connessi al dispositivo tramite un client SSH (ex. PuTTY) scriviamo i seguenti comandi:

```
sudo apt-get update
sudo apt-get upgrade
bash <(curl -sL https://raw.githubusercontent.com/Node-RED/raspbian-deb-package/master/resources/update-nodejs-and-nodered)
sudo systemctl enable nodered.service
Node-RED-start
```

A questo punto il servizio è installato e si avvierà automaticamente.

Se adesso, mediante un browser internet, ci colleghiamo all'indirizzo IP di rete dell'IoT2020 alla porta 1880 (<http://192.168.0.12:1880>) apparirà la seguente interfaccia:



Se la restrizione dell'accesso è abilitata, ci verranno richiesti username e password.

Di default tale funzione non è attiva, per abilitarla rimandiamo alla documentazione disponibile al seguente [link](#).

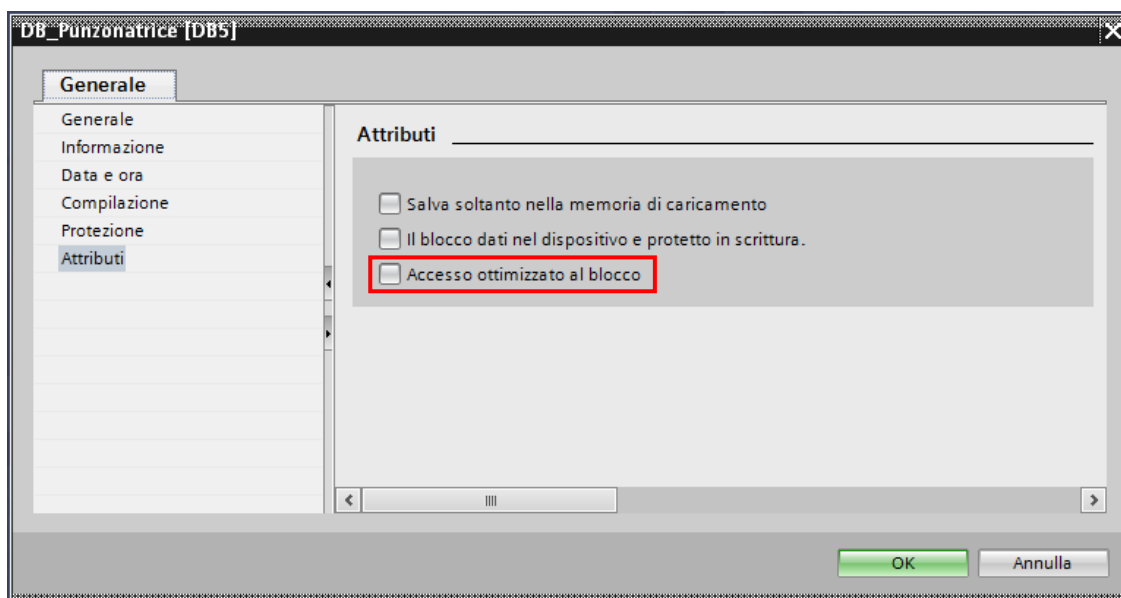
Per poter **comunicare con i PLC Siemens S7-1200** (che utilizzano appunto il protocollo **S7**) è necessario installare un plugin aggiuntivo che ci consentirà la lettura e la scrittura nei **database (DB)** dei controllori.

Per aggiungerlo alla suite di Node-RED è necessario andare sul menu "*Manage Palette*" e cercare **Node-RED-contrib-s7**.

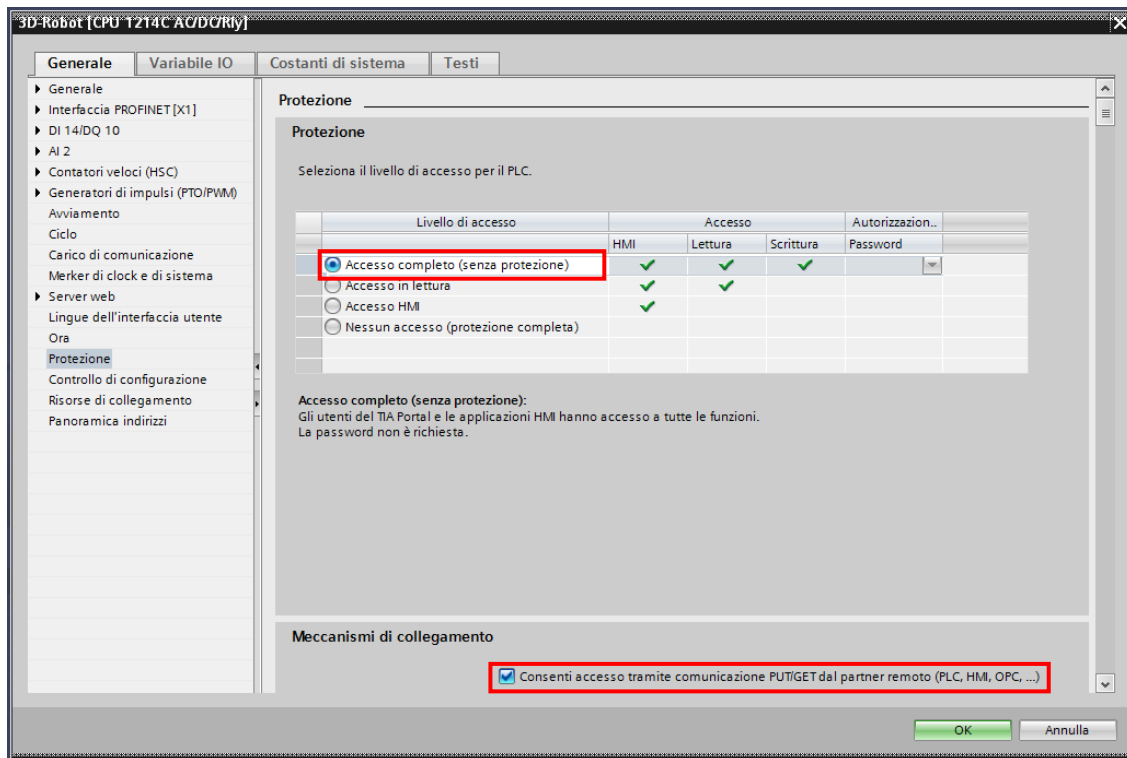
Questo plugin usa la versione "base" del protocollo S7, mentre le nuove versioni di firmware dei PLC S7-1200 impiegano una versione "estesa".

Per poter accedere ai database dei PLC sono pertanto necessari alcuni **passaggi di configurazione aggiuntivi**:

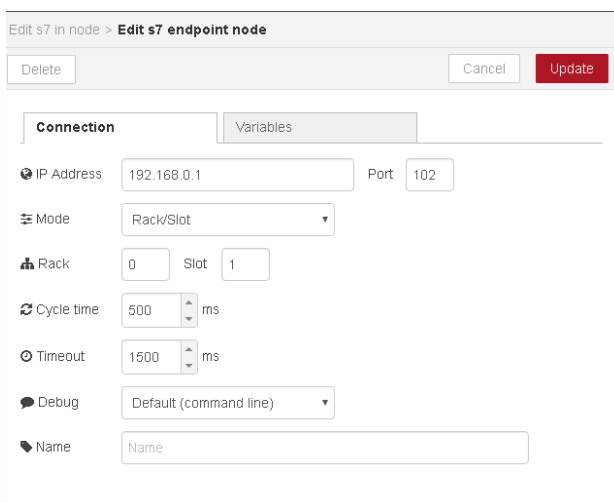
- Nelle proprietà dei DB a cui vogliamo accedere disabilitare la casella di controllo "Accesso al blocco ottimizzato":



- Nella sezione "Protezione" delle proprietà della CPU, abilitare la casella di controllo "Permetti l'accesso con PUT / GET":



Adesso possiamo aggiungere i nodi che permetteranno lo scambio dei dati con i PLC posti sulla stessa rete a cui è connesso l'IOT2020.



Nella scheda **Connection** occorre specificare l'indirizzo IP e la porta (default 102) del PLC con cui vogliamo comunicare.

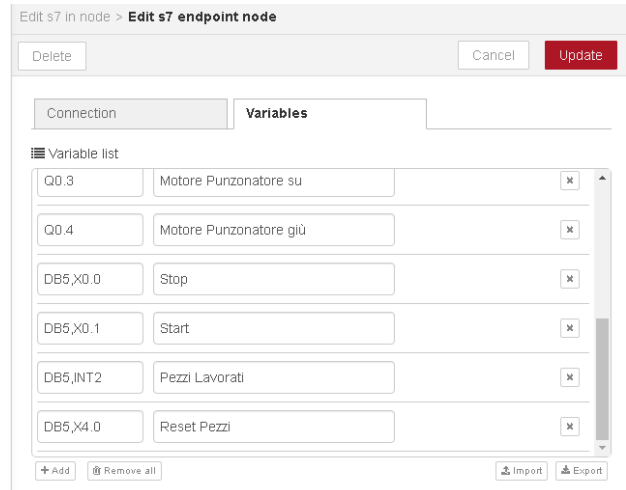
I valori di *Rack (Telaio di montaggio)* e *Slot (Posto connettore)* vengono definiti nella configurazione hardware del PLC, visibile in TIA Portal nella scheda *Generale* delle *proprietà* del PLC (default *Rack=0* e *Slot=1*).

La frequenza di aggiornamento delle variabili viene definita dal valore di *Cycle time*, mentre in caso di errore, trascorso il tempo di *timeout*, sarà effettuato un nuovo tentativo di riconnessione.

L'insieme di queste informazioni identifica un modo univoco un **nodo**.

Nella scheda **Variables** possiamo inserire la lista delle variabili che vogliamo controllare.

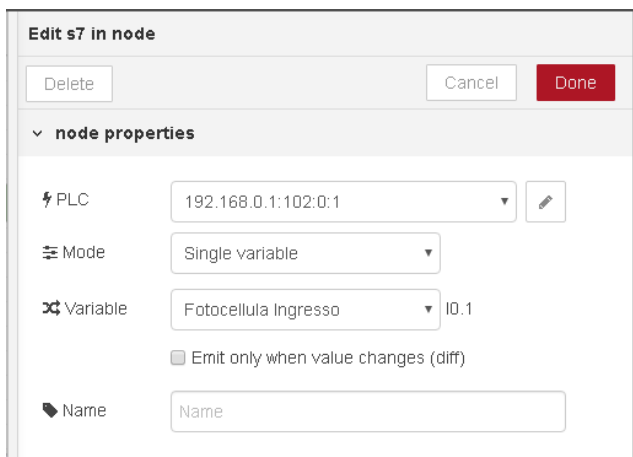
Nella casella di testo a sinistra va inserito l'indirizzo della variabile, mentre in quella di destra il suo nome (può anche essere diverso da quello definito nella tabella delle variabili del PLC).



Come si può notare, in alcuni casi la sintassi utilizzata per accedere ai campi dei database è leggermente diversa da quella usata in TIA Portal. Nella tabella che segue ne sono elencati alcuni esempi:

Indirizzo	Equivalente TIA Portal	Tipo di dati	Descrizione
DB5,X0.1	DB5.X0.1	Bool	bit 1 del byte 0 di DB 5
DB5,B1 o DB5,BYTE1	DB5.DBB1	Byte	Byte 1 (0-255) di DB 5
DB5,I2 o DB5,INT2	DB5.DBW2	Int	signed int a 16-bit dal byte 2 di DB 5
DB57,WORD4	DB57.DBW4	UInt	unsigned int a 16-bit dal byte 4 di DB 57
DB13,DI5 o DB13,DINT5	DB13.DBD5	DInt	signed int a 32-bit dal byte 5 di DB 13
DB19,DW6 o DB19,DWORD6	DB19.DBD6	UDint	unsigned int a 32-bit dal byte 6 di DB 19
DB21,DR7 o DB21,REAL7	DB19.DBD6	Real	Floating point a 32-bit dal byte 7 di DB 21

Per leggere i valori delle variabili del PLC è necessario usare il **blocco Input**:

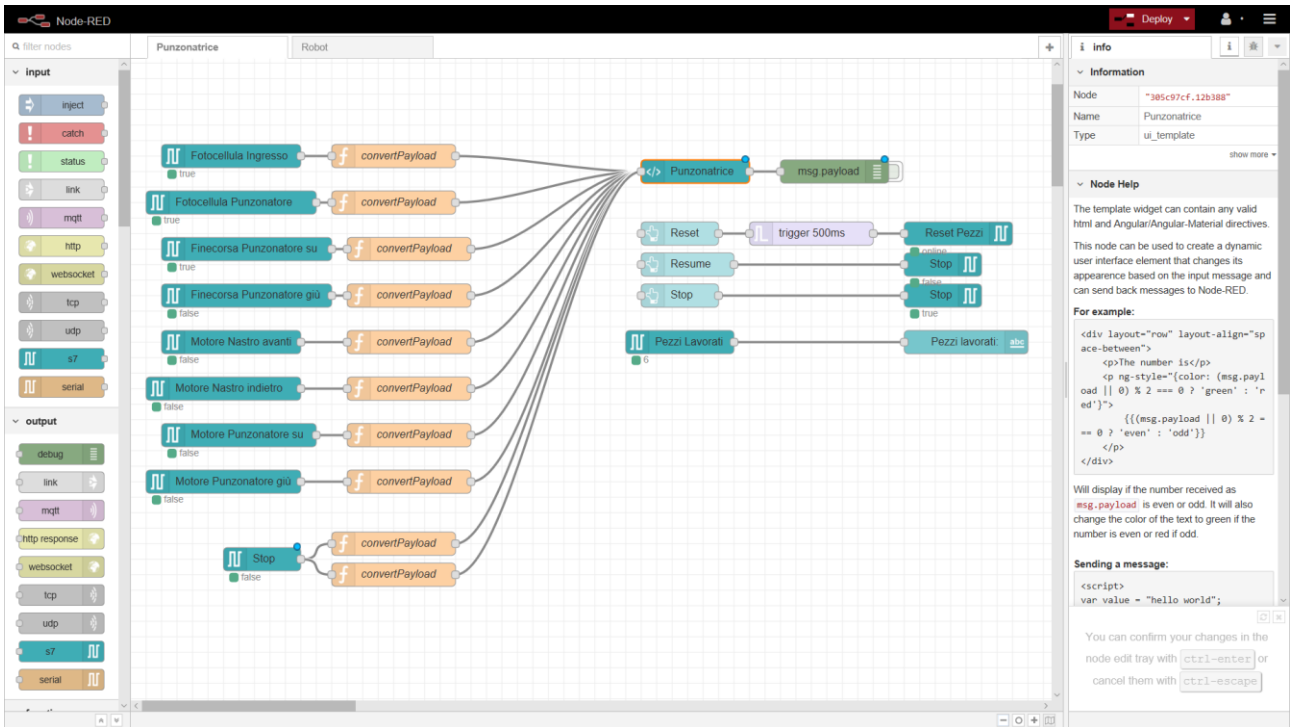


Occorre specificare quale **nodo** si vuole monitorare, la modalità con cui si va a leggere una variabile e il relativo nome.

Con l'opzione "Emit only when value changes (diff)" è possibile generare un messaggio solo quando c'è un cambiamento di stato e non ogni volta che viene aggiornato l'elenco delle variabili (definito nel *refresh time* del nodo S7).

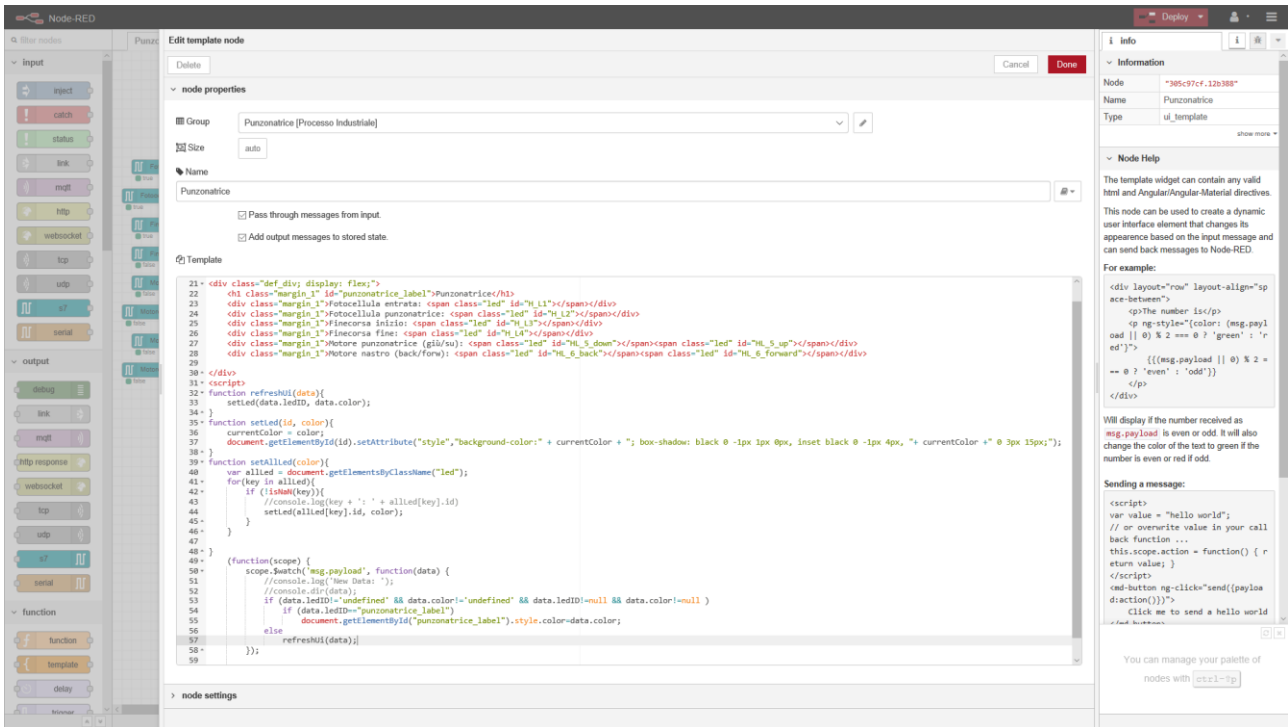
Per scrivere i valori all'interno delle variabili del PLC è necessario usare il **blocco Output**, configurato in modo simile a quello del blocco input.

Quello rappresentato nella seguente figura è il **flow** completo per la supervisione della macchina punzonatrice. Le variabili vengono lette dal PLC che la controlla per poi essere convertite in un **payload JSON** inviato a una **pagina HTML** contenente tutti i controlli:



Nella finestra **Edit function node** è possibile convertire una variabile booleana in un payload JSON contenente l'ID associato ad un LED della dashboard e specificare come cambiare il suo colore.

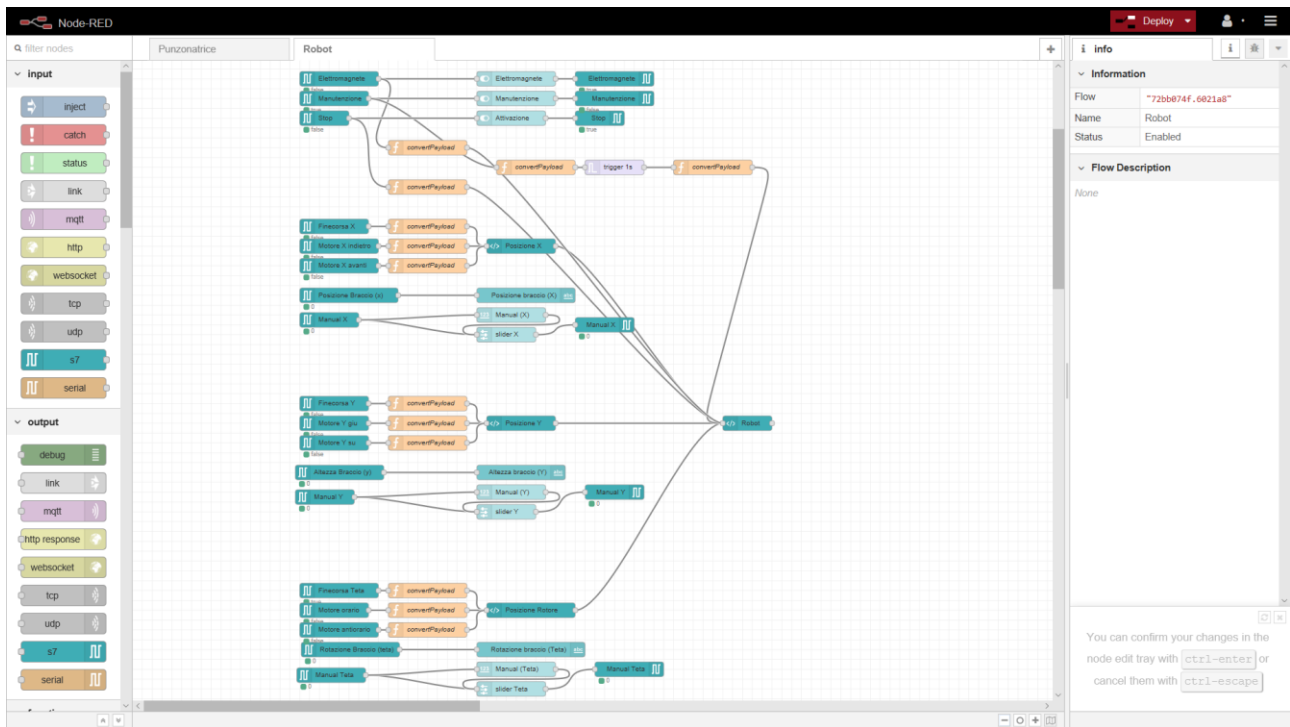
```
1 ledID = "H_L1";
2 if (msg.payload)
3   msg.payload={ledID:ledID, color:"gray"};
4 else
5   msg.payload={ledID:ledID, color:"yellow"};
6 return msg;
```



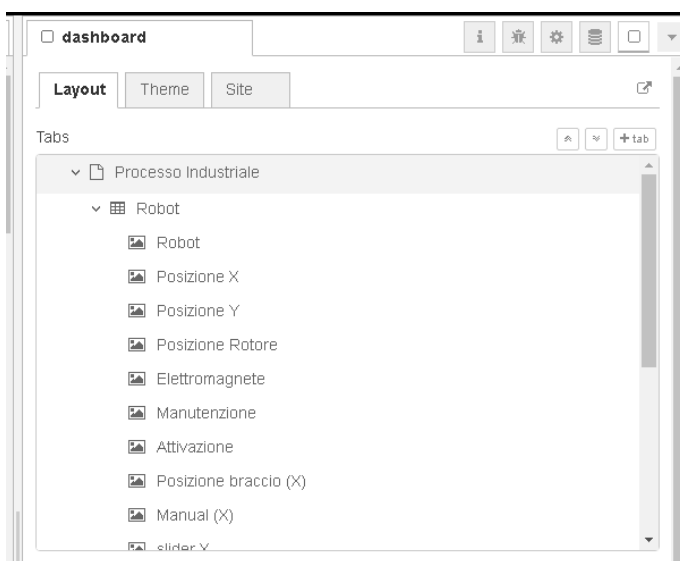
Il **blocco Frame** è composto dai controlli visualizzati nella Dashboard. Il codice è paragonabile a quello di una pagina HTML, con la possibilità di utilizzare **Javascript** (nella figura qui sotto un particolare).

```
Template
30 </div>
31 <script>
32 function refreshUI(data){
33   setLed(data.ledID, data.color);
34 }
35 function setLed(id, color){
36   currentColor = color;
37   document.getElementById(id).setAttribute("style","background-color:" + currentColor + "; box-shadow: black 0 -1px 1px 0px,
38 inset black 0 -1px 4px, "+ currentColor + " 0 3px 15px);");
39 }
40 function setAllLed(color){
41   var allLed = document.getElementsByClassName("led");
42   for(key in allLed){
43     if (!isNaN(key)){
44       //console.log(key + ': ' + allLed[key].id);
45       setLed(allLed[key].id, color);
46     }
47 }
48 </script>
49 (function(scope) {
50   //setAllLed("yellow");
51   scope.$watch('msg.payload', function(data) {
52     //console.log('New Data: ');
53     //console.dir(data);
54     if (data.ledID!="undefined" && data.color!="undefined" && data.ledID!=null && data.color!=null )
55       if (data.ledID=="punzonatrice_label")
56         document.getElementById("punzonatrice_label").style.color=data.color;
57     else
58       refreshUI(data);
59   });
60 })(scope);
61 </script>
```

Grazie alle seguenti funzioni è possibile associare ad ogni led un ID, che in questo modo potrà essere controllato attraverso il payload proveniente dalla lettura delle variabili dei PLC.



Per il **Robot** abbiamo aggiunto la **possibilità di controllarlo da remoto**, ad esempio, per manutenzione. Pertanto, per modificare le coordinate del suo end effector, abbiamo impiegato dei blocchi **"Slider"** e **"Numeric Control"**.



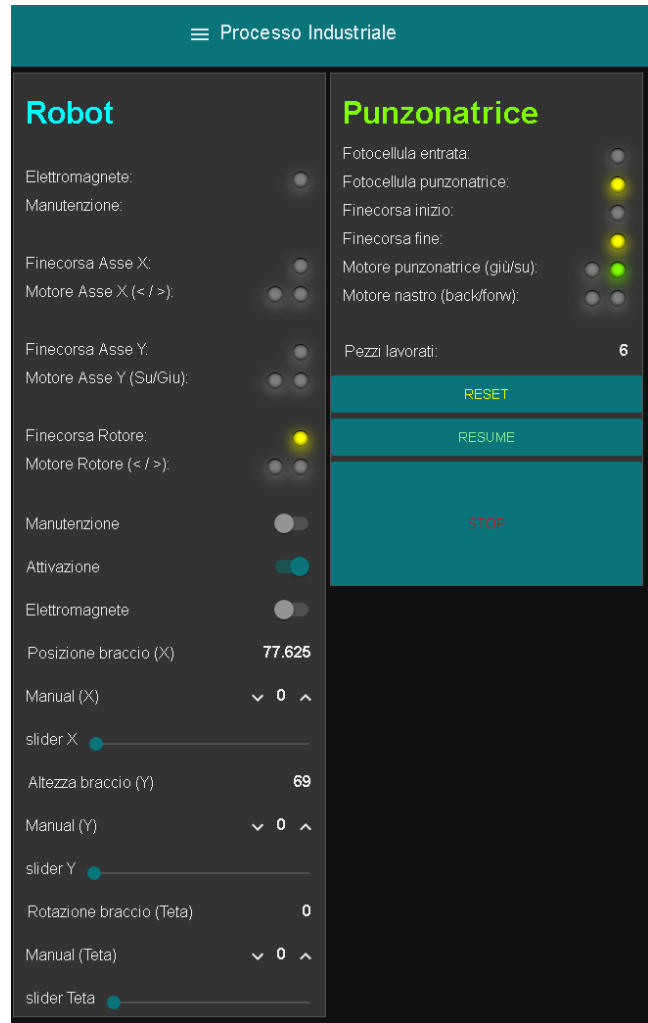
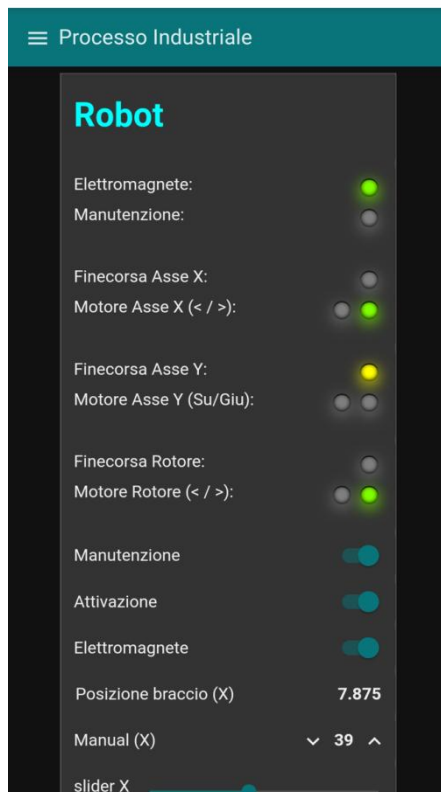
Tutti gli oggetti vengono successivamente riordinati all'interno della pagina web in base all'ordine assegnato nella sezione **Layout** della dashboard.

Impostando su un browser internet l'indirizzo IP del gateway IOT2020 (nel nostro caso 192.168.0.12) seguito dal numero della porta di default, 1880, viene visualizzata la pagina web generata da Node-RED (<http://192.168.0.12:1880/ui>).

In questo modo è possibile monitorare da remoto lo stato del Robot e della Punzonatrice e, se necessario, intervenire in caso di emergenza o di manutenzione.

12:56 27,5KB/s 4G

192.168.0.12:1880/ui/#/1



Dal momento che vengono utilizzati dei layout dinamici, è possibile visualizzare correttamente la pagina web anche da dispositivi con schermo più piccolo, come palmari e smartphone che, grazie al controllo touch, permettono un controllo più versatile del sistema.



### 4. CYBER SECURITY

Come accennato in precedenza, per motivi di sicurezza, i dispositivi connessi alla rete aziendale non devono avere accesso diretto alla rete di produzione, condivisa dai PLC.

Per accedere alle risorse della rete aziendale, gli utenti devono autenticarsi, solitamente fornendo le proprie credenziali (username e password). Se invece accedono tramite una connessione wireless, devono conoscere anche il nome SSID della rete Wi-Fi e la corrispondente password d'accesso, come richiesto dal **protocollo di sicurezza WPA2**.

Una volta ottenuto l'accesso, **gli utenti autorizzati** avranno la possibilità di raggiungere (attraverso un router/firewall) **solo il gateway IOT2020** e, da lì, mediante un browser internet, controllare e supervisionare l'impianto.

Per ulteriore sicurezza abbiamo deciso di non pubblicare su Internet la pagina web creata con Node-RED, ma di renderla accessibile solo dalla rete aziendale.

Gli utenti che hanno la necessità di connettersi da remoto, ovvero dall'esterno della rete aziendale, potranno accedere alla pagina web di supervisione dopo aver effettuato una connessione VPN mobile.

Una **connessione VPN** (Virtual Private Network) prevede la creazione di un **tunnel virtuale** che, **attraverso Internet**, consente agli utenti remoti di accedere alla rete aziendale con le stesse modalità degli utenti locali.

I dati che transitano attraverso questo canale di comunicazione sono cifrati con protocolli di sicurezza praticamente inviolabili. Inoltre, per accedere alla VPN occorre fornire un'autenticazione aggiuntiva.

In questo modo è sufficiente una connessione dati **4G** o un qualsiasi altro accesso a Internet, per connettere in modo sicuro e da qualsiasi parte del mondo, smartphone, tablet e PC alla rete aziendale.

Premesso ciò, **abbiamo simulato questa situazione** aggiungendo alla nostra piccola rete di produzione un **router** con due interfacce di rete. Una collegata alla LAN dei PLC (192.168.0.0/24), l'altra alla rete didattica (172.16.0.0/16), come mostrato in figura 14.

Abbiamo quindi chiesto al nostro amministratore di rete di aggiungere al router/firewall dell'istituto una regola di routing che instradasse tutti i pacchetti destinati all'IOT2020 (192.168.0.12), e solo a quel dispositivo della rete di produzione, verso l'interfaccia 172.16.0.254 del nostro router (percorso evidenziato in rosso)<sup>6</sup>.

---

<sup>6</sup> Diversamente, il router dell'istituto scarterebbe quei pacchetti perché appartengono ad una rete privata che non saprebbe come raggiungere

In questo modo, attraverso il nostro router, sarà possibile arrivare al gateway IOT2020, anche se è su una rete diversa da quella d'istituto.

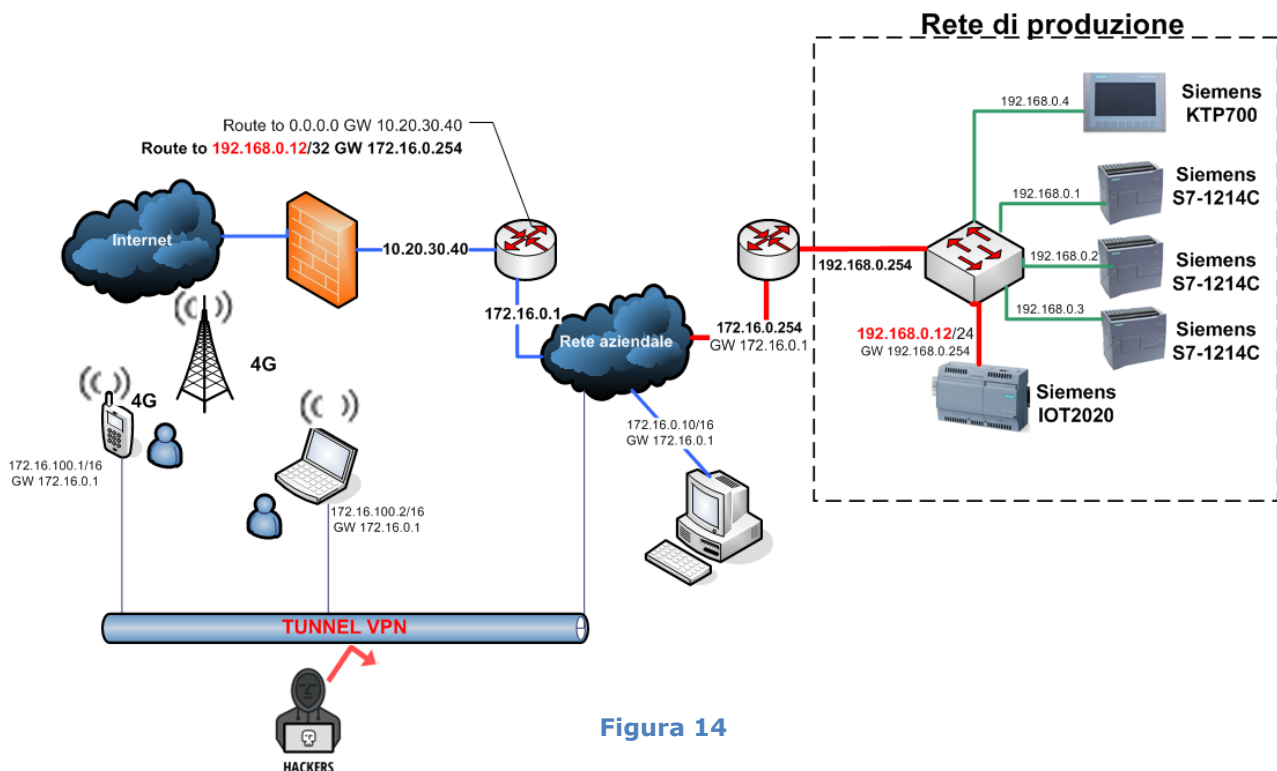


Figura 14

I pacchetti eventualmente destinati agli altri dispositivi della rete di produzione, invece, saranno automaticamente scartati dal router dell'istituto perché non esistono altre regole che contengono quelle destinazioni.

Per verificare se da remoto è possibile raggiungere la pagina web di supervisione creata con Node-RED sul gateway IOT2020, abbiamo innanzitutto configurato i nostri smartphone come client VPN.

Ci siamo quindi collegati, mediante la rete dati 4G dei nostri provider, al server VPN dell'istituto (lo stesso router/firewall). Fornite le corrette credenziali di autenticazione, il nostro smartphone ha acquisito automaticamente un indirizzo IP della LAN.

A quel punto digitando sul browser dello smartphone l'URL <http://192.168.0.12:1880> siamo riusciti a visualizzare la pagina di accesso di Node-RED anche dal nostro smartphone.

### Conclusione

In questo progetto abbiamo voluto realizzare un esempio applicativo, facilmente riproducibile, dove utilizzando tecnologie tipiche dell'**Industria 4.0**, quali l'**IoT** e la **cyber security**, è possibile, in sicurezza, e per qualsiasi necessità, essere sempre connessi da ogni luogo al proprio impianto di produzione.