



**Politecnico
di Torino**

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Meccanica
A.a. 2024/2025
Sessione di Laurea Ottobre 2025

Automazione e supervisione di una linea di trasporto virtuale con PLC Siemens

Relatore:
Prof. Terenziano Raparelli

Candidato:
Dario Giarraputo

Correlatore:
Prof. Luigi Mazza

Correlatore estero:
Prof. Ramon Piedrafita

Abstract

Negli ultimi anni, le industrie hanno subito una grande trasformazione grazie allo sviluppo di tecnologie digitali, come i sensori, sistemi di controllo avanzati, piattaforme di software o di robot dedicate. Questi dispositivi hanno reso i processi produttivi e di trasporto più efficienti, sicuri, flessibili e velocemente adattabili alle varie esigenze del mercato. Infatti, l'Industria 4.0 ha segnato un passaggio fondamentale verso la digitalizzazione e l'automazione industriale, con l'obiettivo di migliorare la competitività delle aziende attraverso l'ottimizzazione delle risorse, la riduzione dei costi operativi e l'aumento della qualità del prodotto.

L'introduzione di strumenti come i controllori logici programmabili (PLC), ossia dispositivi elettronici di programmazione e controllo, rappresentano un elemento fondamentale per la gestione e la supervisione delle linee di produzione e dei sistemi di trasporto interni alle aziende. La loro architettura è basata su un bus, il quale è in grado di collegare e gestire rapidamente e simultaneamente informazioni di input e di output, sia analogici che digitali; con la finalità di migliorare la comunicazione tra le varie piattaforme in modo automatico, viene garantita rapidità di risposta e diagnosi preventiva in caso di mal funzionamento o anomalie.

Lo scopo della presente attività di tesi è quello di sviluppare una linea di trasporto pacchi di tre dimensioni diverse all'interno di un'area industriale simulata, tenendo in considerazione lo spazio disponibile e la praticità di utilizzo. Tale linea di trasporto è stata opportunamente disegnata e progettata attraverso l'uso di un software chiamato Factory I/O, tale software di simulazione in ambiente tridimensionale permette di creare e controllare scenari realistici e di verificarne rapidamente il funzionamento. In questa parte, sono stati opportunamente inseriti sensori, robot e macchinari al fine di garantire un flusso di lavoro corretto e continuo.

Successivamente si è passati alla stesura di un codice in TiaPortal sia in linguaggio Ladder che a blocchi, in grado di gestire l'uscita randomica dei pacchi e di comunicare prontamente con un PLC fisico presente in laboratorio, adibito alla trasmissione dei segnali tra i software, così da regolare e gestire il processo. È stata inoltre implementata, sempre attraverso quest'ultimo software, un'interfaccia HMI, cioè un dispositivo di comunicazioni uomo-macchina, che permette all'operatore di monitorare l'andamento ed il funzionamento del sistema attraverso un display. Tale dispositivo verrà progettato anche in modo tale da permettere all'operatore di fermare o controllare in maniera manuale il processo. Questa funzionalità risulta cruciale non solo per motivi di sicurezza, come nel caso di emergenze o malfunzionamenti, ma anche per attività di manutenzione.

Sebbene tale attività sia stata sviluppata interamente in un ambiente simulato al computer, il progetto può essere introdotto o utilizzato per molteplici impianti reali, grazie alla versatilità e la flessibilità del codice di TiaPortal, unite alla capacità dei PLC di gestire simultaneamente un elevato numero di segnali. In prospettiva, l'integrazione con sistemi di visione artificiale, robot collaborativi e piattaforme di raccolta dati potrebbe ulteriormente ampliare il lavoro svolto rendendolo ancora più efficiente.

Sommario

Abstract	ii
Indice delle immagini	iv
Capitolo 1.....	1
Introduzione ai sistemi automatici	1
1.1 Contesto e motivazioni della ricerca.....	1
1.2 Evoluzione dell'automazione industriale e delle linee di produzione	1
1.3 Cenni storici sui PLC	2
1.4 Il ruolo dei PLC nei sistemi di automazione industriale	2
1.5 Differenza tra PLC Modicon e Siemens.....	4
1.6 Composizione di un PLC Siemens	4
1.7 Vantaggi e svantaggi dei PLC, sicurezza, affidabilità e comunicazione via Ethernet	6
1.8 Ethernet come metodo di comunicazione.....	7
1.9 Obiettivi della tesi	7
Svolgimento della prova in laboratorio	8
Software utilizzati: TIAPortal e Factoryio.....	19
3.1 Linguaggio SFC	20
3.2 Qualificatori delle azioni	21
3.3 Spiegazione del codice di TiaPortal.....	24
Comando Manuale attraverso l'uso dell'HMI.....	44
Comando d'emergenza e Rearme	53
Conclusioni e sviluppi futuri	59
Bibliografia	61

Indice delle immagini

Figura 1: PLC di laboratorio.....	3
Figura 2: Sistema completo della linea di trasporto	8
Figura 3: Tipologia di pacco.....	9
Figura 4: WheelSorter	9
Figura 5: Tipologie di Conveyors.....	10
Figura 6: Sensore di prossimità	12
Figura 7: Pusher per pacchi	13
Figura 8: Robot 2 (XZ).....	14
Figura 9: TurnTable	15
Figura 10: Robot 1 (XYZ).....	16
Figura 11: Pallettizzatore	17
Figura 12: Almacen di ingresso	18
Figura 13: Connessione con il computer di riferimento	19
Figura 14: Connessione con il PLC di riferimento	20
Figura 15: Variabili di input e output di Factoryio	22
Figura 16: Variabili di Factoryio per il movimento delle macchine speciali	23
Figura 17: Variabili importate su TiaPortal	24
Figura 18: Tabella dei codici realizzati.....	25
Figura 19: SFC del WheelSorter parte 1.....	25
Figura 20: SFC del WheelSorter parte 2.....	26
Figura 21: SFC del WheelSorter parte 3.....	27
Figura 22: Conveyor iniziale del tratto di destra	28
Figura 23: Seconda parte dei conveyor di destra.....	28
Figura 24: Terza parte dei conveyor di destra.....	29
Figura 25: Quarta parte dei conveyor di destra	29
Figura 26: SFC Robot 1 parte 1	30
Figura 27: SFC Robot 1 parte 2.....	31
Figura 28: SFC Robot 1 parte 3.....	31
Figura 29: SFC Robot 1 parte 4.....	32
Figura 30: SFC Robot 1 parte 5.....	32
Figura 31: SFC del Turntable	34
Figura 32: SFC del pallettizzatore parte 1	35
Figura 33: SFC del pallettizzatore parte 2	35
Figura 34: SFC del pallettizzatore parte 3	36
Figura 35: SFC del pallettizzatore parte 4	36
Figura 36: Contatori del pallettizzatore	37
Figura 37: Main "Nastro"	38
Figura 38: SFC dell'Almacen parte 1	39
Figura 39: SFC dell'Almacen parte 2	40
Figura 40: SFC dell'Almacen parte 3	40
Figura 41: FC dell'Almacen.....	41
Figura 42: Main dell'Almacen parte 1	42
Figura 43: Main dell'Almacen parte 2	43
Figura 44: HMI di laboratorio	44
Figura 45: Progettazione dell'HMI virtuale	45
Figura 46: Comunicazione tra PLC e HMI	46

Figura 47: Screen 0 dei sensori e degli attuatori	47
Figura 48: Screen dei sensori e degli attuatori del palletizzatore	47
Figura 49: Screen 0 del control manual.....	48
Figura 50: Screen del Control manual del Robot 2	48
Figura 51: Screen del control de la maquina	49
Figura 52: SFC per il controllo manuale parte1	50
Figura 53: SFC per il controllo manuale parte 2	50
Figura 54: SFC per il controllo manuale parte 3	51
Figura 55: Codice SCL per il comando manuale del robot 1	51
Figura 56: Codice SCL per il comando manuale del robot 2	51
Figura 57: Codice SCL per il comando manuale dell'alamcen di ingresso	52
Figura 58: Main degli Almacenes.....	54
Figura 59: Main per la progettazine del comando d'emergenza	55
Figura 60: Interlock	55
Figura 61: Interlock e Supervisione.....	56
Figura 62: Codice di emeregenza per il Robot 2	57
Figura 63: Codice d'emergenza Almacen IN	57
Figura 64: Startup per l'emergenza	58

Capitolo 1

Introduzione ai sistemi automatici

1.1 Contesto e motivazioni della ricerca

Negli ultimi anni, l'industria manifatturiera ha subito una grande trasformazione grazie allo sviluppo di tecnologie che hanno reso i processi produttivi più efficienti, sicuri e flessibili. L'Industria 4.0 ha segnato un passaggio fondamentale verso la digitalizzazione e l'automazione industriale, con l'obiettivo di migliorare la competitività delle aziende attraverso l'ottimizzazione delle risorse e la riduzione dei costi operativi. In questo contesto, l'utilizzo dei controllori logici programmabili PLC, ossia dispositivi elettronici di programmazione e controllo, rappresentano un elemento chiave per la gestione delle linee di produzione e dei sistemi di trasporto interni alle aziende. L'automazione delle linee di produzione e distribuzione è diventata una priorità per molte imprese, soprattutto nei settori logistici. L'integrazione di PLC ha permesso la riduzione di errori umani e una maggiore affidabilità nei processi, garantendo risparmi in termini di tempo e risorse e migliorandone allo stesso tempo la sicurezza. Un aspetto critico nelle aziende moderne è la gestione dei sistemi di trasporto interni, ossia l'organizzazione e il controllo della movimentazione di merci all'interno degli stabilimenti produttivi. Questa tesi, infatti, si propone di approfondire il ruolo dell'automazione nelle linee di trasporto e distribuzione interna, focalizzandosi in particolare sulla realizzazione di una linea di distribuzione pacchi interna basata sull'uso di PLC. Lo studio nasce dall'osservazione di linee di trasporto preesistenti in un ambiente aziendale reale, con l'obiettivo di analizzarne le criticità e proporre modifiche migliorative per incrementare l'efficienza, l'affidabilità e la flessibilità del sistema, con un occhio alla sostenibilità dei processi.

1.2 Evoluzione dell'automazione industriale e delle linee di produzione

Per raggiungere un tale livello di automazione, ora richiesto nelle aziende, sono trascorsi molti anni; all'interno dei quali possono essere individuate quattro fasi principali, ognuna caratterizzata da idee e tecnologie innovative per il periodo di appartenenza. Le principali tappe dell'evoluzione industriale sono le seguenti:

- Prima Rivoluzione Industriale (XVIII-XIX secolo): caratterizzata dall'introduzione della macchina a vapore, quindi da una prima meccanizzazione dei processi produttivi, che ha permesso la nascita delle prime fabbriche.
- Seconda Rivoluzione Industriale (XIX-XX secolo): caratterizzata dalla produzione di massa con l'introduzione della catena di montaggio e

dell'elettificazione industriale, che ha permesso di aumentare la velocità di produzione e ridurre i costi.

- Terza Rivoluzione Industriale (XX secolo): caratterizzata dall'introduzione dell'elettronica, dell'informatica e dell'automazione nei processi produttivi, che ha permesso di rendere i sistemi industriali più adattabili.
- Quarta Rivoluzione Industriale (Industria 4.0, XXI secolo): basata sull'integrazione di sistemi cyber-fisici, intelligenza artificiale e automazione avanzata, che rende la produzione ancora più flessibile. [1]

Nel contesto dell'Industria 4.0, la produzione diventa sempre più intelligente e connessa, grazie all'utilizzo di PLC, SCADA (Supervisory Control and Data Acquisition) e Big Data. L'obiettivo è quello di creare sistemi di produzione intelligenti in grado di adattarsi opportunamente alle condizioni di mercato e alle esigenze produttive, migliorando la competitività e la sostenibilità dei processi [2].

1.3 Cenni storici sui PLC

Il concetto di PLC è nato negli anni '60 con l'introduzione dei primi controllori programmabili, che hanno permesso la sostituzione di sistemi basati su relè elettromeccanici. I primi PLC furono sviluppati per risolvere le difficoltà legate alla complessità dei cablaggi e alla necessità di adattamento frequente alle modifiche nei processi produttivi. Il primo PLC fu creato da un ingegnere della Modicon, un'azienda che poi è stata acquisita da Schneider Electric. Il Modicon 084 fu il primo PLC progettato per il controllo dei processi industriali nel settore automobilistico, specificamente per la catena di montaggio General Motors. Durante gli anni '70, i PLC si diffusero rapidamente e vennero standardizzati, con l'introduzione di linguaggi di programmazione come il Ladder, che riproduceva la logica dei relè elettromeccanici in una forma grafica facilmente interpretabile. Oggi i PLC sono dispositivi complessi con capacità di connessione avanzate, che possono interagire con sistemi di supervisione, sensori, attuatori e dispositivi IoT. La loro continua evoluzione ha portato all'integrazione di funzionalità come la gestione della manutenzione predittiva, il supporto a reti industriali complesse e l'automazione di processi in tempo reale.

1.4 Il ruolo dei PLC nei sistemi di automazione industriale

I PLC sono dispositivi elettronici programmabili adibiti al controllo e alla gestione dei processi industriali. Grazie alla loro versatilità, è possibile usufruirne in molteplici settori, dalla produzione industriale alla logistica interna, in quanto sono in grado di:

- Automatizzare e coordinare il funzionamento delle macchine e delle linee di produzione;
- Monitorare e regolare i parametri di processo in tempo reale, consentendo una gestione precisa di lavoro;

- Ridurre gli errori e aumentare l'affidabilità del sistema, grazie all'eliminazione di operazioni manuali e a un controllo automatico delle varie fasi produttive;
- Migliorare la sicurezza degli impianti grazie a controlli avanzati e sistemi di allarme;
- Integrare diversi dispositivi, come sensori, attuatori e sistemi di supervisione, creando soluzioni automatizzate altamente integrate.

Nell'ambito dei sistemi di trasporto interni, i PLC svolgono un ruolo fondamentale nel controllo e nella gestione delle linee di movimentazione automatizzate, in questo caso in linea di trasporto e stoccaggio pacchi. Grazie alla loro versatilità e programmabilità, i PLC permettono di adattare facilmente i flussi di trasporto alle esigenze produttive, ottimizzando le diverse fasi della catena di approvvigionamento interna [3].



Figura 1: PLC di laboratorio

1.5 Differenza tra PLC Modicon e Siemens

Nel mondo dell'automazione industriale, i PLC si dividono principalmente in due grandi famiglie: i Modicon e i Siemens. Sono entrambi molto usati, ma hanno caratteristiche diverse che li rendono più adatti a certi contesti piuttosto che ad altri.

I PLC Modicon sono stati sviluppati dalla Modicon (oggi Schneider Electric) negli anni '60. Questi PLC sono spesso utilizzati in ambienti di produzione di piccole e medie dimensioni. Offrono una programmazione semplice e una buona integrazione con i dispositivi di automazione esistenti. Modicon è noto per la sua modularità e flessibilità, consentendo alle aziende di configurare sistemi che possono crescere con le esigenze produttive. I PLC Modicon sono spesso utilizzati nelle applicazioni in cui è richiesta una rapida implementazione e un'interfaccia semplice. Invece, i PLC Siemens, in particolare quelli della serie S7, sono noti per le loro prestazioni elevate e l'affidabilità. Questi PLC sono ideali per applicazioni industriali di grande scala e per ambienti di produzione complessi. La linea Siemens S7 offre una vasta gamma di opzioni di configurazione, rendendola una scelta popolare in vari settori, dalle linee di produzione automatizzate alle applicazioni in ambito chimico e automobilistico. Inoltre, i PLC Siemens sono compatibili con il software di programmazione TIA Portal, che consente di gestire in modo efficiente la programmazione, il monitoraggio e la diagnostica di impianti complessi.

In sintesi, i PLC Modicon sono più adatti per applicazioni di dimensioni più contenute, mentre i PLC Siemens sono preferiti per le installazioni di grandi dimensioni e ad alta complessità, grazie alla loro capacità di gestire sistemi complessi e alla loro affidabilità comprovata [4].

1.6 Composizione di un PLC Siemens

Un PLC Siemens è costituito da due parti profondamente diverse tra loro:

- L'insieme dei circuiti elettronici e delle apparecchiature necessarie per fornire le varie funzioni (hardware);
- L'insieme delle istruzioni che formano il programma utente sulla base delle esigenze specifiche (software).

Il principio di funzionamento del sistema PLC può essere sintetizzato nello schema sotto riportato: il PLC riceve segnali di ingresso che giungono dai sensori sul campo, elabora tali segnali secondo il programma ed emette in uscita segnali di comando.

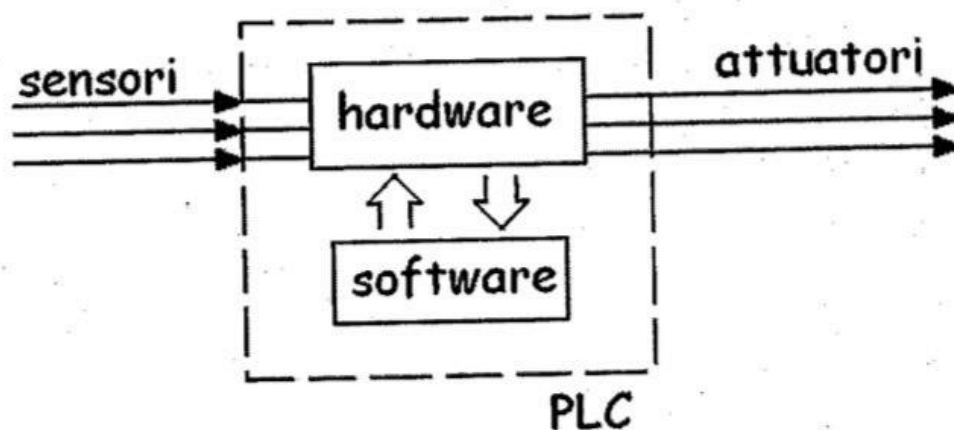


Figura 2: Schema di base del PLC [5]

L'hardware di un PLC è composto da diversi elementi, ognuno con un ruolo specifico nel controllo del processo industriale. Il cuore del sistema è rappresentato dal bus, un canale che permette a tutti i componenti di comunicare tra loro in modo coordinato. A questo bus sono collegati vari moduli che, lavorando insieme, rendono possibile il funzionamento del PLC. Il componente principale è la CPU (Central Processing Unit), ovvero l'unità centrale di elaborazione. È la CPU che esegue ciclicamente il programma memorizzato, analizza lo stato degli ingressi e, in base a questi, determina i valori da assegnare alle uscite. Il PLC dispone di due tipi principali di memoria: la ROM, utilizzata per contenere in modo permanente il programma, e la RAM, che serve per l'elaborazione e l'immagazzinamento temporaneo dei dati durante il funzionamento. Accanto a questi elementi troviamo i moduli di ingresso e uscita, sia digitali che analogici, che permettono al PLC di interfacciarsi con il mondo esterno, ricevendo segnali da sensori e inviando comandi ad attuatori. È presente anche una porta di programmazione, che consente di collegare un computer per caricare o modificare il programma, tramite un apposito software. Completano la struttura i moduli di comunicazione, fondamentali per permettere al PLC di scambiare dati con altri dispositivi o con altri PLC, e l'alimentatore, che fornisce l'energia necessaria al funzionamento di tutto il sistema.

Grazie alla loro struttura modulare, i PLC Siemens possono essere facilmente espansi e personalizzati in base alle specifiche esigenze applicative, rendendoli una soluzione versatile e ampiamente adottata in diversi settori industriali. [5]

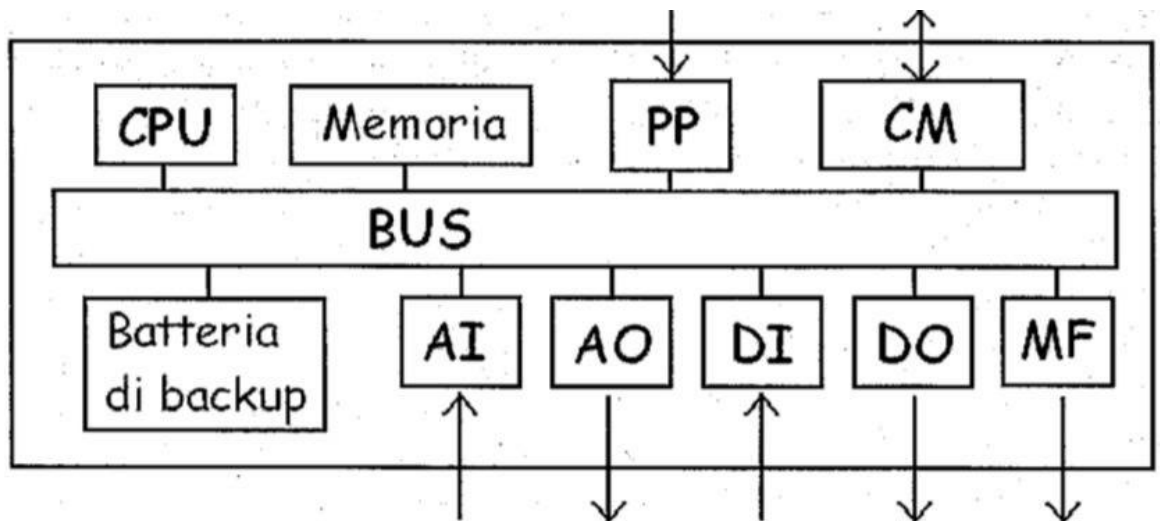


Figura 3: Parte Hardware del PLC [5]

1.7 Vantaggi e svantaggi dei PLC, sicurezza, affidabilità e comunicazione via Ethernet

I PLC presentano numerosi vantaggi tra questi vi è l'affidabilità, in quanto sono dispositivi progettati per poter lavorare in ambienti industriali di ogni tipo, con condizioni di temperatura diverse, vibrazioni, polvere e umidità. La loro capacità di lavorare ininterrottamente e di resistere a condizioni ambientali diverse li rende particolarmente adatti per l'automazione di linee di produzione o di trasporto. A differenza dei sistemi tradizionali, i PLC eliminano la necessità di collegamenti a filo, riducendo notevolmente il rischio di guasti meccanici e aumentando la durata dell'impianto. Un altro vantaggio è la loro modularità, caratteristica che consente di adattare facilmente il sistema alle esigenze produttive che possono variare in base al mercato e alla domanda di quel determinato prodotto o operazione. I moduli di ingresso e uscita analogici e digitali, i moduli di comunicazione e le unità di elaborazione centrale (CPU) possono essere facilmente sostituiti o aggiornati senza compromettere l'intero sistema. Inoltre, i PLC offrono una facile programmazione grazie all'utilizzo di linguaggi standardizzati, come il Ladder o l'SFC, che rendono il processo di configurazione intuitivo anche per gli operatori meno esperti. Un ultimo vantaggio è la sicurezza: i PLC sono progettati per ridurre i rischi di guasti attraverso la continua supervisione dei processi e la gestione delle anomalie. Grazie alla loro capacità di monitorare in tempo reale il comportamento dei macchinari e dei dispositivi ad essi collegati, i PLC possono arrestare immediatamente i processi in caso di situazioni pericolose, evitando danni agli impianti o incidenti agli operatori, anticipandone la manutenzione attraverso il segnale di opportuni sensori. Questa funzione di stop di emergenza è una delle ragioni principali per cui i PLC sono considerati indispensabili in ambito industriale. I PLC possono infatti raccogliere e analizzare dati relativi ai componenti della macchina e alle performance dei sistemi in tempo reale, permettendo agli operatori di agire prima che si verifichi un guasto, riducendo così i costi legati alla manutenzione straordinaria e aumentando la disponibilità dell'impianto.

Nonostante i numerosi vantaggi, l'utilizzo dei PLC presenta anche alcuni svantaggi. I PLC per operazioni ad alte prestazioni e con funzionalità avanzate sono abbastanza costosi. I costi per l'acquisto, l'installazione e la manutenzione possono essere significativi, specialmente per le piccole e medie imprese. Inoltre, i PLC possono essere meno flessibili rispetto ad altri dispositivi di controllo, come i PC industriali, quando si tratta di applicazioni che richiedono una gestione complessa dei dati o l'analisi in tempo reale di grandi volumi di informazioni. Inoltre, sebbene i PLC siano estremamente affidabili, in alcune circostanze l'interfaccia di programmazione può essere poco intuitiva per chi non ha esperienza in ambito di automazione. Ciò può comportare errori durante la programmazione, con conseguenti difficoltà nella diagnosi di malfunzionamenti o nella risoluzione di problemi.

1.8 Ethernet come metodo di comunicazione

Per collegamento via Ethernet si intende un metodo che permette di collegare dispositivi diversi del sistema di riferimento, come attuatori, sensori e controllori, attraverso l'utilizzo, non di cavi, ma di una rete virtuale opportunamente realizzata. Attraverso l'utilizzo di questa modalità, si garantisce un'elevata affidabilità e rapidità d'esecuzione nelle varie operazioni da svolgere, in quanto si diminuisce l'ingombro causato dalla presenza di cavi e si prevengono possibili errori dovuti al guasto di essi. In questa tesi, la comunicazione Ethernet viene utilizzata per collegare opportunamente il PLC con il computer, sul quale è stato realizzato il modello virtuale della linea di trasporto pacchi di riferimento mediante Factoryio. Attraverso tale collegamento, sarà possibile simulare l'esecuzione del sistema intervenendo prontamente sulle zone di errore, così facendo si può garantire l'affidabilità e l'efficacia del sistema prima di realizzare il progetto fisico.

1.9 Obiettivi della tesi

Questa ricerca si pone l'obiettivo di implementare una linea di distribuzione pacchi interna a un'azienda, basandosi su sistemi già esistenti ma introducendo alcune modifiche mediante software come FactoryIo e TIAPortal. In particolare, gli obiettivi principali sono:

- Pensare ad una linea di trasporto pacchi interna ad un'azienda efficace;
- Rappresentare il sistema su Factoryio;
- Progettare la parte automatica, manuale e di emergenza attraverso TIAPortal;
- Progettare l'HMI del sistema sempre tramite TIAPortal.

Capitolo 2

Svolgimento della prova in laboratorio

Dopo aver approfondito la parte teorica di riferimento, si può passare alla descrizione concreta della linea di trasporto realizzata.

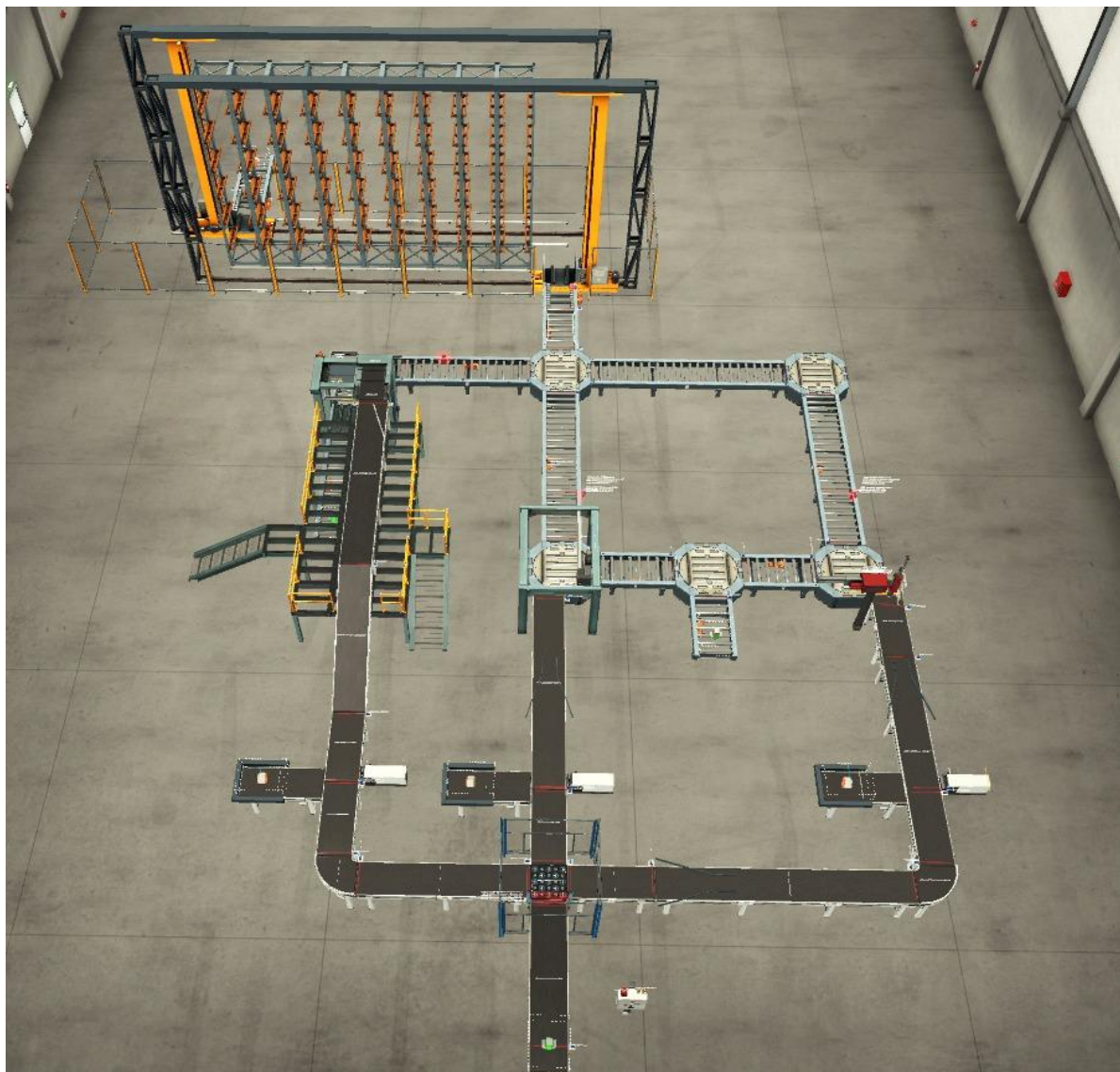


Figura 4: Sistema completo della linea di trasporto

Il sistema parte da un emettitore di pacchi, da cui compaiono i pacchi da smistare, suddivisi in tre categorie: piccoli, medi e grandi.



Figura 5: Tipologia di pacco

Questi pacchi vengono trasportati lungo un primo Belt-Conveyor fino a raggiungere un componente centrale del sistema: il WheelSorter.

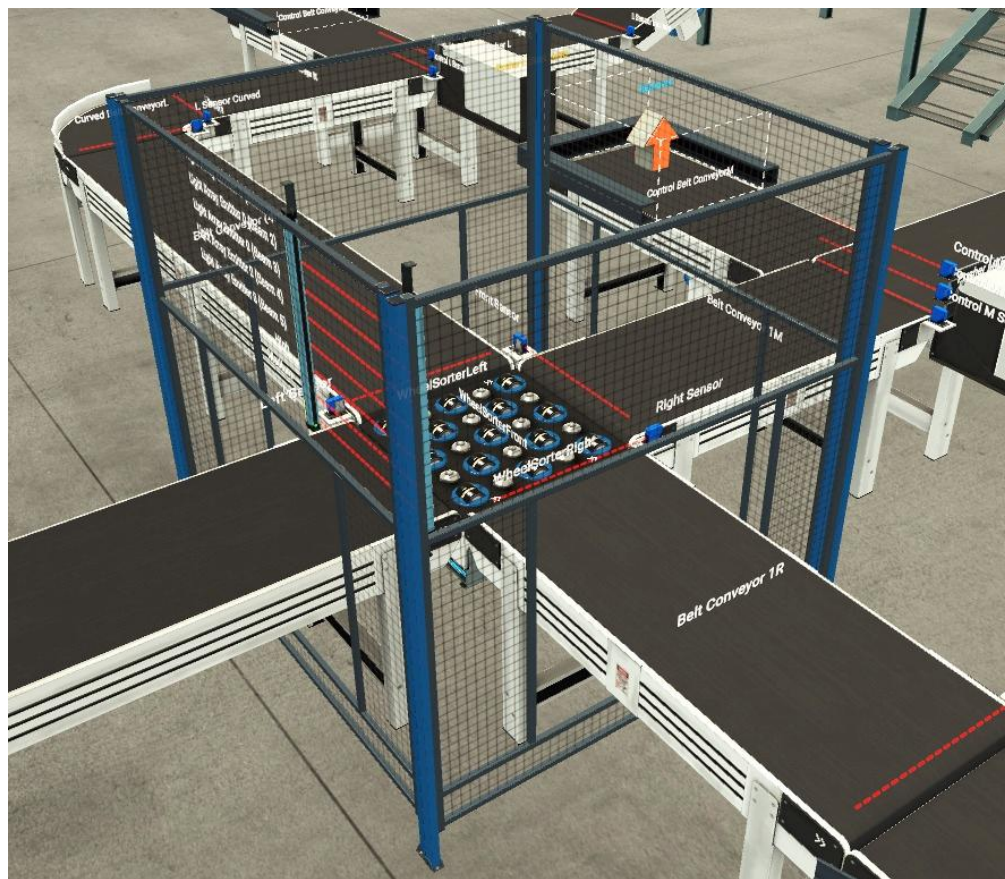


Figura 6: WheelSorter

Si tratta di un deviatore dotato di rulli orientabili che, grazie al segnale di un sensore di altezza posizionato poco prima, è in grado di capire la dimensione del pacco e indirizzarlo nella direzione corretta. Il meccanismo è semplice ma efficace: il sensore rileva l'altezza, il WheelSorter orienta le sue ruote per spingere il pacco verso la linea giusta.

Il sistema è stato progettato per smistare i pacchi in tre direzioni diverse: quelli piccoli vengono deviati verso sinistra, quelli grandi verso destra, mentre i pacchi medi proseguono in avanti. Una volta deviati, i pacchi continuano lungo la loro linea di trasporto dedicata, composta da una serie di nastri trasportatori (belt conveyor) di dimensioni differenti per la gestione degli spazi all'interno dell'area di lavoro. In questo progetto oltre al Belt-Conveyor sono stati usati anche i Roller-Conveyor.

- Il Belt conveyor, o nastro trasportatore, è un elemento di movimentazione sul quale gli oggetti scorrono sopra un nastro continuo, solitamente realizzato in gomma o materiale sintetico. Questa tipologia di trasporto permette di realizzare un movimento fluido e uniforme, utilizzato specialmente per oggetti di piccole dimensioni o leggeri, poiché garantisce ottimi livelli di aderenza e controllo durante lo spostamento.
- Il Roller conveyor, o rulliera, è realizzato da una serie di rulli disposti in parallelo su cui i materiali avanzano per rotolamento. I rulli in questione sono motorizzati, ossia formati da un azionamento che consente la rotazione attiva dei rulli. Questa tipologia di Conveyor è più adatta per oggetti rigidi, pesanti o con una base piatta, poiché il movimento risulta meno continuo ma comunque efficace. Rispetto al nastro, la rulliera è generalmente più economica e semplice da mantenere, ma offre un controllo minore sul posizionamento preciso dei carichi. [6]



Figura 7: Tipologie di Conveyors

Ciascun nastro si attiva al passaggio del pacco, grazie a un sensore di prossimità posizionato a monte: quando il sensore rileva il pacco, il relativo segnale digitale

cambia stato da 0 a 1, ovvero da contatto normalmente aperto a chiuso, consentendo così l'attivazione del nastro successivo.

I sensori di prossimità vengono utilizzati per rilevare la presenza di un oggetto nella rispettiva zona di visione del sensore stesso, senza che vi sia un contatto fisico diretto. Negli impianti automatizzati, questi sensori vengono impiegati per controllare il passaggio dei materiali e per sincronizzare le azioni dei vari componenti del sistema. Ne esistono diverse tipologie, ciascuna basata su un principio di funzionamento differente, a seconda della natura dell'oggetto da rilevare e dell'ambiente operativo.

Nel progetto in esame, si fa uso di sensori di prossimità ottici, poiché caratterizzati da un'ottima combinazione tra versatilità e affidabilità nella rilevazione, in questo caso pacchi su linee di trasporto. Questi sensori si basano sull'emissione e la ricezione di un raggio luminoso, solitamente nell'infrarosso, e rilevano la presenza di un oggetto quando questo interrompe il fascio.

Esistono diverse configurazioni di sensori ottici: a barriera, a riflessione diretta e a riflessione con catarifrangente.

- I sensori a barriera sono costituiti da un emettitore e un ricevitore posizionati uno di fronte all'altro; il sensore rileva l'oggetto nel momento in cui questo interrompe il fascio tra i due.
- Quelli a riflessione diretta sono formati da un emettitore e ricevitore integrati nello stesso corpo: la presenza viene rilevata quando la luce emessa colpisce un oggetto e viene riflessa verso il ricevitore.
- I sensori con catarifrangente funzionano in modo simile a quelli a riflessione, ma richiedono la presenza di un elemento adibito a quest'ultima, posto di fronte al sensore; l'interruzione del fascio tra il sensore e il catarifrangente indica la presenza di un oggetto.

Per la seguente linea di trasporto, la scelta è ricaduta sui sensori ottici a riflessione diretta, per via di una combinazione di semplicità, affidabilità e praticità d'installazione. Rispetto alle altre configurazioni disponibili, questi sensori offrono molteplici vantaggi in applicazioni dove è richiesto il rilevamento rapido e preciso del passaggio di oggetti.

A differenza dei sensori a barriera, i quali richiedono il posizionamento di due dispositivi contrapposti (un emettitore e un ricevitore), i sensori a riflessione diretta presentano tutto integrato in un unico corpo, riducendo la complessità meccanica e semplificandone il montaggio.

Per quanto riguarda i sensori con catarifrangente, quelli a riflessione diretta non necessitano di alcun elemento aggiuntivo (come lo specchio riflettente da posizionare di fronte), il che elimina eventuali problemi legati alla sporcizia, all'orientamento o alla manutenzione di componenti passivi. In ambienti dove il passaggio continuo di pacchi può generare polvere, vibrazioni o leggere

imprecisioni meccaniche, la presenza di un unico dispositivo, robusto e autonomo, migliora l'affidabilità complessiva del sistema.

Inoltre, i sensori a riflessione diretta moderni sono in grado di rilevare oggetti con superfici opache, scure o irregolari, rendendoli particolarmente adatti a gestire pacchi di diversa forma, colore o materiale, come avviene tipicamente in una linea di smistamento.



Figura 8: Sensore di prossimità

Come anticipato, i pacchi seguono percorsi diversi in base alla loro dimensione, ma in tutte e tre le direzioni è stato inserito un pusher, pensato per correggere eventuali errori nello smistamento da parte del WheelSorter. Può capitare, ad esempio, che un pacco piccolo venga deviato verso il percorso sbagliato, quello progettato per i pacchi grandi, poiché per qualche motivo non è stato riconosciuto correttamente dal sensore di altezza a monte del wheelSorter oppure potrebbe essere dovuto ad un errore di movimentazione del WheelSorter stesso. In questi casi entra in gioco il pusher, che riesce a rilevare in modo indipendente l'altezza del pacco grazie a una serie di sensori ottici aggiuntivi, sistemati su diversi livelli. Se il sensore più alto non si attiva, significa che il pacco in realtà non è della taglia prevista per quel percorso. A quel punto il pusher interviene attraverso la movimentazione di un elemento metallico, la cui estensione spinge il pacco fuori dal percorso principale, facendolo finire su un nastro secondario. Questo Conveyor lo trasporta poi fino al ricevitore virtuale, dove il pacco viene considerato da scartare o da gestire separatamente.



Figura 9: Pusher per pacchi

Sulla base di quanto detto finora, si passa adesso alla trattazione dei tre possibili percorsi che i pacchi possono intraprendere: destra, sinistra e avanti. Il percorso di destra, dedicato ai pacchi di grandi dimensioni, presenta un restringimento subito dopo il pusher. Tale restringimento è costituito da due elementi laterali, in plastica rigida, posizionati ai lati del nastro per indirizzare correttamente i pacchi verso la zona finale del nastro. Arrivato alla fine del nastro, si attiva il Roller-Stop adibito a stoppare il funzionamento del Conveyor. Questo stop consente al robot di prendere il pacco con precisione. Il pacco si ferma in una zona ben precisa grazie all'uso di un sensore posto al di sotto del robot, che rileva la presenza e facilita la presa.

Il robot impiegato alla fine di questo percorso è il Two-Axis Robot, il quale presenta due gradi di libertà, che gli permettono di spostarsi lungo l'asse orizzontale X e l'asse verticale Z. Tale configurazione è ideale per operazioni di Pick and Place di base. Il robot è costituito da un attuatore di presa, chiamato Gripper, adibito alla presa dei pacchi, tale presa avviene una volta raggiunta la posizione prestabilita, precedentemente misurata da un sensore posizionato alla base del robot. Il movimento operativo consiste in uno spostamento orizzontale ed uno verticale, opportunamente misurati, per raggiungere l'oggetto di riferimento; infine, vi è l'attivazione dell'organo di presa. La semplicità del movimento riduce sia il tempo di programmazione che la possibilità di errore, mantenendo comunque una buona flessibilità operativa. Inoltre, la struttura compatta del Two-Axis Robot è ideale in applicazioni in cui gli spazi di lavoro sono ridotti, come quelli tipici di una linea di smistamento. Completata la presa, il robot solleva l'oggetto e rilascia il pacco, abbassandosi nuovamente, dopo una rotazione di 90 gradi, su un pallet opportunamente posizionato sul turntable corrispondente così da garantire la continuazione della linea di trasporto.

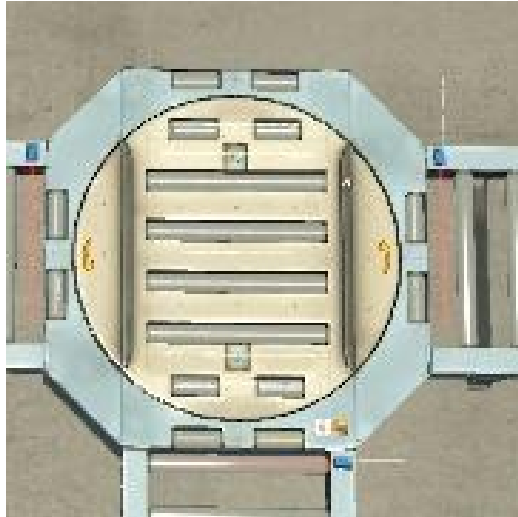


Figura 11: TurnTable

Se nel percorso di destra è presente un Two-Axis Robot, nel percorso centrale superato il pusher ed il restringimento, si giunge ad un ulteriore roller-stop il quale si attiva non appena viene attivato il sensore posizionato poco prima. Attivato il roller-stop, inizia la movimentazione del secondo robot, il Three-Axis Robot, definito “Robot_1” nella parte di codice.

Il Three-Axis Robot è un robot cartesiano che presenta tre gradi di libertà che gli permettono di muoversi lungo gli assi X, Y e Z, corrispondenti ai movimenti orizzontale, verticale e in profondità rispetto alla struttura di riferimento. In pratica, può raggiungere un qualsiasi punto nello spazio delimitato dalla sua struttura. Esso viene usato per eseguire delle semplici operazioni di pick and place, cioè, prendere un pacco da una posizione definita e depositarlo dove richiesto in modo preciso. Il movimento avviene sulla base di coordinate impostate per ciascun asse, e una volta raggiunta la posizione desiderata, il robot attiva la presa per afferrare il pacco o la disattiva per rilasciarlo, in questo caso su un pallet opportunamente posizionato, come spiegato in precedenza. Questo tipo di robot è utilizzato soprattutto quando viene richiesta un’elevata flessibilità nei movimenti, in quanto è in grado di lavorare su diversi livelli e direzioni.



Figura 12: Robot 1 (XYZ)

Dopo aver analizzato il percorso centrale e di destra, si passa ora alla descrizione del tratto di sinistra, destinato ai pacchi di piccole dimensioni, dove, dopo una sequenza di belt conveyor e l'intervento di un pusher, si raggiunge l'elemento più rilevante di questo tratto, cioè il pallettizzatore.

Questo sistema è pensato per automatizzare e facilitare il processo di impilamento, ottimizzando i tempi, riducendo l'errore umano e garantendo un'esecuzione precisa e ripetibile. In sostanza la movimentazione del pallettizzatore inizia con l'attivazione del comando di rotazione (turn) per orientare correttamente i pacchi e successivamente quello di spinta (push) per far scorrere i pacchi nella posizione esatta, secondo una configurazione di tre file da due pacchi ciascuna. Una volta che tutti e sei i pacchi sono stati collocati, viene attivato il comando open plate, che apre la piattaforma e lascia cadere i pacchi direttamente sul pallet, già posizionato correttamente al di sotto. Il pallet stesso viene movimentato tramite i comandi elevator up ed elevator down, che permettono il sollevamento e l'abbassamento dell'elevatore di riferimento.



Figura 13: Palletizzatore

Dopo aver analizzato anche il funzionamento del palletizzatore, si arriva all'ultima piattaforma rotante, ovvero un altro Turntable, il quale ha il compito di indirizzare i pallet provenienti dai tre percorsi possibili verso l'Almacén. Il Turntable decide in quale direzione ruotare grazie alla presenza di tre sensori posizionati alla fine di ciascun percorso: non appena un pallet attiva uno di questi sensori, la piattaforma ruota nella direzione indicata. Nel caso in cui due pallet attivino contemporaneamente i sensori di fine percorso, si verifica una situazione di conflitto. In questo scenario, il Turntable darà precedenza a uno dei due pallet in base alla logica stabilita nel codice: sarà infatti il programma a determinare quale percorso ha priorità, risolvendo così la coincidenza in modo controllato e prevedibile. L'obiettivo principale della Turntable è dunque quello di indirizzare il pallet verso l'Almacén, che rappresenta l'ultimo elemento nella catena di trasporto analizzata.

L'almacén è il luogo destinato alla conservazione e gestione, in questo caso, dei pacchi. Non si tratta solo di uno spazio fisico in cui risiedono prodotti in attesa di essere utilizzati o spediti, ma è un luogo che sta alla base della logistica della produzione, in quanto in esso sono presenti sensori che segnalano da quanto tempo i pacchi sono stati depositati, dove sono stati depositati e quando e dove devono uscire dal magazzino. L'almacén può essere suddiviso in aree funzionali, come la

zona di ricevimento, quella di stoccaggio e la zona di spedizione. Ogni spostamento di materiale all'interno di queste aree deve essere tracciato per garantire affidabilità e rapidità nelle operazioni. La funzione dell'almacén, quindi, non è soltanto conservare, ma anche organizzare e ottimizzare i flussi di materiale all'interno di un sistema produttivo o distributivo. La sua efficienza incide direttamente sull'efficacia dell'intera azienda poiché, un magazzino mal gestito può generare ritardi, errori di spedizione o accumuli inutili di merce. Al contrario, un magazzino ben progettato e controllato è in grado di garantire continuità operativa, velocità di risposta e riduzione dei costi. Per questo motivo, parlare di almacén oggi significa parlare anche di automazione, informazione e controllo.

In sintesi, l'almacén è il cuore silenzioso della logistica in quanto, come detto prima, è essenziale per far sì che tutto il resto possa funzionare in modo ordinato ed efficiente.

Una volta orientato il pallet verso l'Almacén, quest'ultimo (rilevata la presenza tramite il sensore di riferimento) si attiva uscendo dalla condizione di riposo. Raggiunta la posizione prestabilita mediante coordinato sull'asse X e Z, le forks entrano in azione per prelevare il pallet. Dopo averlo sollevato, le forks tornano nella posizione iniziale, attivando così il middle limit. L'attivazione di questo sensore consente all'Almacén di muoversi lungo gli assi X e Z per posizionare il pallet nel punto esatto di stoccaggio.

Una volta completato il posizionamento, l'Almacén di ingresso è pronto a ripetere il ciclo. Invece, l'Almacén di uscita ha il compito di prelevare il pallet già depositato e trasferirlo verso il conveyor di uscita finale.



Figura 14: Almacen di ingresso

Capitolo 3

Software utilizzati: TIAPortal e Factoryio

Come accennato in precedenza, i software utilizzati in questo testo sono Factoryio e TIAPortal. Il primo è un software di simulazione tridimensionale specializzato per impianti industriali. Esso ti dà la possibilità di inserire nello spazio definito elementi come: robot, sensori, nastri trasportatori e altro, in modo da creare l'impianto o la struttura desiderata. Tale software è ideale per operazioni che richiedono la simulazione di un impianto da automatizzare attraverso un PLC, in quanto riesce a coordinarsi perfettamente con il codice del PLC modulato su TIAPortal. A differenza di Factoryio, TIAPortal è un software creato da Siemens che serve per programmare i PLC, in quanto è possibile scrivere il codice che gestisce tutta la linea produttiva o di trasporto pensata o realizzata.

La coordinazione dei due Software permette di poter visualizzare rapidamente l'efficacia del codice realizzato e di verificare la correttezza della struttura in esame.

Per creare il file di codice su TiaPortal, è necessario creare il PLC virtuale seguendo i passaggi proposti e bisogna scegliere il PLC di riferimento. In questo caso è stato scelto quello presente in laboratorio, cioè il PLC Siemens SIMATIC S7-1500 ma in un contesto reale bisogna far riferimento al catalogo presente sul sito ufficiale della Siemens. [7]

Sono fondamentali i blocchi di CommunicationSetup e le proprietà del file di codice, poiché il primo serve per mettere in relazione il codice con il computer sul quale è presente l'impianto da simulare ed il secondo serve per collegare il Computer con il PLC di riferimento, così da rendere possibile l'automatizzazione del processo da simulare.

```
1 (*Introducir aqui la direccion IP donde se ejecuta Factory IO.  
2 Normalmente es la IP de vuestro ordenador laboratorio*)  
3 "Client_RD".CONNECT.  
4 RemoteAddress.ADDR[1] := 10;  
5 "Client_RD".CONNECT.  
6 RemoteAddress.ADDR[2] := 3;  
7 "Client_RD".CONNECT.  
8 RemoteAddress.ADDR[3] := 17;  
9 "Client_RD".CONNECT.  
10 RemoteAddress.ADDR[4] := 131;  
11
```

Figura 15: Connessione con il computer di riferimento

Figura 16: Connessione con il PLC di riferimento

3.1 Linguaggio SFC

Per la stesura del codice dell'impianto su TIAPortal è stato utilizzato un linguaggio SFC, Sequential Functional Chart, uno dei linguaggi di programmazione più usati per PLC. Tale linguaggio è costituito da tre elementi di base fondamentali:

- La fase;
- La transizione;
- L'arco orientato.

Le fasi rappresentano uno stato attivo del sistema. Ad ognuna di esse viene associata una variabile booleana, detta Marker della fase, con valori 0 o 1 a seconda che la fase sia attiva o disattiva. Ad ogni fase possono, anche, essere associate una o più azioni che rappresentano gli effettivi movimenti operativi del sistema o della macchina in esame. Graficamente, la fase viene rappresentata con un rettangolo con il nome della stessa da non confondere con la fase iniziale, che sarà rappresentata da una doppia cornice o rettangolo.

La transizione, invece, indica la condizione da superare per il passaggio da una fase a quella successiva. Come per la fase, anche per la transizione, è importante associare una condizione, intesa come funzione booleana di variabili booleane. È fondamentale tenere in considerazione che una transizione è superabile quando tutte le fasi a monte sono attivate e la condizione logica ad essa associata è vera. Esse possono essere rappresentate in diversi modi, ma quello usato nel codice di riferimento è rappresentato mediante una barretta posta sull'arco orientato.

L'arco orientato può essere considerato l'elemento cruciale del linguaggio SFC in quanto collega fra loro le fasi stabilendone la sequenza. Il diagramma SFC si realizza verticalmente e l'orientamento va dal bordo inferiore di una fase al bordo superiore.

Il Sequential Functional Chart è caratterizzato da due regole principali: cioè, che tra due fasi collegate ci deve essere sempre una transizione, e che tra due transizioni deve esserci sempre una fase. [5]

3.2 Qualificatori delle azioni

Esistono diversi qualificatori di azioni, ognuno dei quali indica e presenta caratteristiche e informazioni differenti.

Il qualificatore più usato in questo codice è sicuramente quello di tipo N, Non-Stored, in quanto esso indica che l'azione termina quando la fase di riferimento si disattiva.

Inoltre, sono stati usati i qualificatori R e S, che rappresentano rispettivamente il reset ed il set, il reset indica il fatto che termina un'azione attivata da altri qualificatori, come quello S; invece, il qualificatore di Set indica che l'azione continua anche quando la fase diventa inattiva e termina, solo nel momento in cui viene opportunamente resettata.

Per completezza della trattazione si riporta di seguito una tabella che rappresenta i vari qualificatori utilizzabili.

Qualificatore	Nome	Descrizione
nessuno		Di default, ha lo stesso significato di "N"
N	<i>Non-stored</i>	L'azione termina quando la fase diventa disattiva
R	<i>Reset</i>	Termina un'azione attivata con i qualificatori S, SD, SL o DS
S	<i>Set</i>	L'azione continua anche quando la fase diventa inattiva, e termina quando l'azione viene resettata
L	<i>Time Limited</i>	L'azione comincia quando la fase diventa attiva e continua finché la fase diventa inattiva oppure trascorre un certo intervallo di tempo
D	<i>Time Delayed</i>	L'azione ritardata nel tempo. Se la fase è ancora attiva dopo il tempo prefissato di temporizzazione l'azione comincia e termina quando la fase si disattiva.
P	<i>Pulse</i>	L'azione comincia quando la fase diventa attiva (P1) o disattiva (P0) e viene eseguita una sola volta.
SD	<i>Stored and Time Delayed</i>	L'azione comincia dopo un ritardo anche se la fase diventa inattiva e continua finché non è resettata.
DS	<i>Time Delayed and Stored</i>	L'azione ritardata nel tempo. Se la fase è ancora attiva dopo il tempo prefissato di temporizzazione l'azione comincia e continua finché non è resettata. E' simile a SD, tranne per il fatto che l'azione non si attiva se la fase termina prima del tempo prefissato.
SL	<i>Stored and Time Limited</i>	L'azione comincia quando la fase diventa attiva e continua finché non viene resettata o non trascorre un certo intervallo di tempo

Figura 17: Qualificatori delle azioni [5]

DRIVER		Modbus TCP/IP Server		START		CONFIGURATION	CLIENT
SENSORS				ACTUATORS			
2A PMP (Dripper Rotating)		FACTORY I/O (Running)	Input 0	Coil 0	FACTORY I/O (Pause)		2A PMP (On/Off)
2A PMP (Item Selection)		FACTORY I/O (Paused)	Input 1	Coil 1	FACTORY I/O (Reset)		2A PMP Dripper CW
2A PMP (Rotating)		FACTORY I/O (Reset)	Input 2	Coil 2	FACTORY I/O (Run)		2A PMP Rotate CCW
2A PMP X Position (V)		Emergency Stop 0	Input 3	Coil 3	Alarm Siren 0		2A PMP Rotate CW
2A PMP Z Position (V)		Manual	Input 4	Coil 4	Reset light		2A PMP Rotate CCW
Box at entry		Emergency Stop 0	Input 5	Coil 5	Reset Button 0 (Light)		2A PMP X Set Point (V)
Clamped		Manual	Input 6	Coil 6	Reset Light 0 (Flash)		2A PMP Z Set Point (V)
Control 1 Sensor 1		Reset	Input 7	Coil 7	Black Light 0 (Steady)		Alarm Siren 0
Control 1 Sensor 2		START BUTTON	Input 8	Coil 8	Black Light 0 (Yellow)		Belt Conveyor 1M
Control 1 Sensor 3		Stop Button 0	Input 9	Coil 9	Start Button 0 (Light)		Belt Conveyor 1R
Control 1 Sensor 4		Entry Item	Input 10	Coil 10	Stop Button 0 (Light)		Belt Conveyor 2L
Control 1 Sensor 5		Front Sensor	Input 11	Coil 11	Warning Light 0		Belt Conveyor 2R
Control 1 Sensor 6		Left Sensor	Input 12	Coil 12	Box Emitter		Belt Conveyor 4M
Control 1 Sensor 7		Right Sensor	Input 13	Coil 13	Entry Belt Conveyor		Belt Conveyor 4R
Control 1 Sensor 8		High	Input 14	Coil 14	WheelSensorLeft		Belt Conveyor 4M 1
Control 1 Sensor 9		Medium	Input 15	Coil 15	WheelSensorLeft		Belt Conveyor 4R 1
Control 1 Sensor 10		Low	Input 16	Coil 16	WheelSensorRight		Belt Conveyor 4M 2
Control 1 Sensor 11		Light Array Emitter 0 (Beam 1)	Input 17	Coil 17	Belt Conveyor 4M 1		Belt Conveyor 4R 1
Control 1 Sensor 12		Light Array Emitter 0 (Beam 2)	Input 18	Coil 18	Curved Belt ConveyorL		Belt Conveyor 4M 2
Control 1 Sensor 13		Light Array Emitter 0 (Beam 3)	Input 19	Coil 19	Belt Conveyor 3L		Belt Conveyor 4R 2
Control 1 Sensor 14		Light Array Emitter 0 (Beam 4)	Input 20	Coil 20	Control Belt ConveyorL		Box Emitter
Control 1 Sensor 15		Light Array Emitter 0 (Beam 5)	Input 21	Coil 21	Pusher L		Clamp
Control 1 Sensor 16		L Sensor 2	Input 22	Coil 22	unlabeled part		Control Belt ConveyorM
Control 1 Sensor 17		L Sensor Curved	Input 23	Coil 23	Belt Conveyor 4L		Control Belt ConveyorR
Control 1 Sensor 18		Control L Sensor 1	Input 24	Coil 24	Belt Conveyor IndividedL		Conveyor Bracket IN
Control 1 Sensor 19		Control L Sensor 2	Input 25	Coil 25	Belt Conveyor 4M 2		Curved Belt ConveyorL
Control 1 Sensor 20		Pusher L (Front Limit)	Input 26	Coil 26	Clamp		Curved Belt ConveyorR
Control 1 Sensor 21		Pusher L (Back Limit)	Input 27	Coil 27	Elevator up		Curved Belt ConveyorL
Control 1 Sensor 22		L Sensor Individed	Input 28	Coil 28	Elevator down		Curved Roller Conveyor L

Questa prima immagine mostra come le variabili definite di input, come i sensori, vengono posizionate sulla sinistra. Invece, sulla destra sono presenti gli output, o coil, come gli attuatori del sistema in esame.

Invece, nell'immagine sotto riportata, sono presenti tutte quelle variabili adibite alla movimentazione delle macchine complesse del sistema, come robot e almacén, e ai loro rispettivi sensori.

	Input 127	Coil 127	
FACTORY I/O (Time Scale)	Input Reg 0	Holding Reg 0	FACTORY I/O (Camera Position)
RFID Reader L Command ID	Input Reg 1	Holding Reg 1	Digital Display 0
RFID Reader L Read Data	Input Reg 2	Holding Reg 2	RFID Reader L Command
RFID Reader L Status	Input Reg 3	Holding Reg 3	RFID Reader L Write Data
P&P X Position (V)	Input Reg 4	Holding Reg 4	RFID Reader L Memory Index
P&P Y Position (V)	Input Reg 5	Holding Reg 5	P&P X Set Point (V)
P&P Z Position (V)	Input Reg 6	Holding Reg 6	P&P Y Set Point(V)
RFID Reader M Command ID	Input Reg 7	Holding Reg 7	P&P Z Set Point (V)
RFID Reader M Read Data	Input Reg 8	Holding Reg 8	RFID Reader M Command
RFID Reader M Status	Input Reg 9	Holding Reg 9	RFID Reader M Memory Index
2A P&P X Position (V)	Input Reg 10	Holding Reg 10	RFID Reader M Write Data
2A P&P Z Position (V)	Input Reg 11	Holding Reg 11	2A P&P X Set Point (V)
RFID Reader R Command ID	Input Reg 12	Holding Reg 12	2A P&P Z Set Point (V)
RFID Reader R Read Data	Input Reg 13	Holding Reg 13	RFID Reader R Command
RFID Reader R Status	Input Reg 14	Holding Reg 14	RFID Reader R Memory Index
RFID Reader 3 Command ID	Input Reg 15	Holding Reg 15	RFID Reader R Write Data
RFID Reader 3 Status	Input Reg 16	Holding Reg 16	RFID Reader 3 Command
RFID Reader 3 Read Data	Input Reg 17	Holding Reg 17	RFID Reader 3 Write Data
Stacker Crane IN X Position (V)	Input Reg 18	Holding Reg 18	RFID Reader 3 Memory Index
Stacker Crane IN Z Position (V)	Input Reg 19	Holding Reg 19	Stacker Crane IN X Set Point (V)
Stacker Crane 0 X Position (V)	Input Reg 20	Holding Reg 20	Stacker Crane IN Z Set Point (V)
Stacker Crane 0 Z Position (V)	Input Reg 21	Holding Reg 21	Stacker Crane 0 X Set Point (V)
	Input Reg 22	Holding Reg 22	Stacker Crane 0 Z Set Point (V)

Figura 19: Variabili di Factoryio per il movimento delle macchine speciali

Dopo aver inserito tutti i drivers, è necessario importare il file in TIA Portal. In caso di conflitti o ridondanze nella numerazione delle variabili, bisogna sistemarle rispettando comunque le caratteristiche e i vincoli di base del PLC utilizzato.

208	Roller_Conveyor_60	Tabla de variables	Bool	%M230.2	☐	☒	☒	☒	
209	Turntable_2R_Turn	Tabla de variables	Bool	%M230.3	☐	☒	☒	☒	
210	Turntable_2R_Roll(+)	Tabla de variables	Bool	%M230.4	☐	☒	☒	☒	
211	Turntable_2R_Roll(-)	Tabla de variables	Bool	%M230.5	☐	☒	☒	☒	
212	Roller_Conveyor_61	Tabla de variables	Bool	%M230.6	☐	☒	☒	☒	
213	Turntable_OUT_Roll(+)	Tabla de variables	Bool	%M230.7	☐	☒	☒	☒	
214	Turntable_OUT_Roll(-)	Tabla de variables	Bool	%M231.0	☐	☒	☒	☒	
215	Turntable_OUT_Turn	Tabla de variables	Bool	%M231.1	☐	☒	☒	☒	
216	Conveyor_Stacker_IN	Tabla de variables	Bool	%M231.2	☐	☒	☒	☒	
217	RFID_Reader_3_Execute_Com...	Tabla de variables	Bool	%M231.3	☐	☒	☒	☒	
218	Roller_Conveyor_Stacker_IN	Tabla de variables	Bool	%M231.4	☐	☒	☒	☒	
219	Stacker_Crane_IN_(Left)	Tabla de variables	Bool	%M231.5	☐	☒	☒	☒	
220	Stacker_Crane_IN_(Right)	Tabla de variables	Bool	%M231.6	☐	☒	☒	☒	
221	Loading_Conveyor_0	Tabla de variables	Bool	%M231.7	☐	☒	☒	☒	
222	Roller_Conveyor_20	Tabla de variables	Bool	%M232.0	☐	☒	☒	☒	
223	Remover_1_(Remove)	Tabla de variables	Bool	%M232.1	☐	☒	☒	☒	
224	Stacker_Crane_0_(Left)	Tabla de variables	Bool	%M232.3	☐	☒	☒	☒	
225	Stacker_Crane_0_(Right)	Tabla de variables	Bool	%M232.4	☐	☒	☒	☒	
226	FACTORY_I/O_(Time_Scale)	Tabla de variables	Int	%MW240	☐	☒	☒	☒	
227	RFID_Reader_L_Command_ID	Tabla de variables	Int	%MW242	☐	☒	☒	☒	
228	RFID_Reader_L_Read_Data	Tabla de variables	Int	%MW244	☐	☒	☒	☒	
229	RFID_Reader_L_Status	Tabla de variables	Int	%MW246	☐	☒	☒	☒	
230	P&P_X_Position_(V)	Tabla de variables	Int	%MW248	☐	☒	☒	☒	
231	P&P_Y_Position_(V)	Tabla de variables	Int	%MW250	☐	☒	☒	☒	
232	P&P_Z_Position_(V)	Tabla de variables	Int	%MW252	☐	☒	☒	☒	
233	RFID_Reader_M_Command_ID	Tabla de variables	Int	%MW254	☐	☒	☒	☒	
234	RFID_Reader_M_Read_Data	Tabla de variables	Int	%MW256	☐	☒	☒	☒	
235	RFID_Reader_M_Status	Tabla de variables	Int	%MW258	☐	☒	☒	☒	
236	AP&PXPosition	Tabla de variables	Int	%MW260	☐	☒	☒	☒	
237	AP&PZPosition	Tabla de variables	Int	%MW262	☐	☒	☒	☒	
238	RFID_Reader_R_Command_ID	Tabla de variables	Int	%MW264	☐	☒	☒	☒	
239	RFID_Reader_R_Read_Data	Tabla de variables	Int	%MW266	☐	☒	☒	☒	
240	RFID_Reader_R_Status	Tabla de variables	Int	%MW268	☐	☒	☒	☒	

Figura 20: Variabili importate su TiaPortal

Come si osserva dall'immagine sopra riportata, le variabili si distinguono in due tipologie: le "Bool" e le "Int". Bool rappresenta la variabile booleana cioè, una variabile che può assumere solo due valori True (1) e False (0), invece, la Int è una variabile numerica intera. Inoltre, si può osservare come ad ogni variabile venga affidato una sorta di codice identificativo, costituito da un numero e da un acronimo che può essere M o MW. M indica la memory bit, cioè, rappresenta una singola variabile booleana; invece, MW indica una parola di memoria e serve per memorizzare valori numerici (Int).

3.3 Spiegazione del codice di TiaPortal

La stesura del codice prevede come step iniziale quello di andare a creare un blocco, attraverso la sezione "Add new device", di tipo FB di linguaggio SFC. Una volta creato sarà possibile inserire le varie sequenze di codice, ognuna delle quali rappresenterà una macchina dell'impianto. Come si osserva dall'immagine sotto riportata sono stati realizzati ben 25 codici.

un determinato evento. Si parla di sequenze divergenti e convergenti nei punti in cui la struttura in parallelo inizia e finisce. Una regola di base per realizzare una struttura del genere è quella che prima del ramo delle alternative, non deve essere presente alcuna transizione e alla fine, deve essere presente un'unica condizione comune, che in questo caso è quella del salto, la cui funzione è quella di far ripartire il ciclo dalla fase iniziale.

La prima transizione rappresenta l'inizio del ciclo, infatti è costituito da "Factoryio_Running" che indica l'inizio della comunicazione con il software di simulazione e poi vi sono le condizioni dei vari sensori che, se non attivati non permettono l'attivazione della fase successiva. La fase successiva è contraddistinta dall'azionamento del conveyor di ingresso, la cui fine presenta il WheelSorter. La struttura in parallelo serve al WheelSorter ad indirizzare il pacco in una direzione piuttosto che in un'altra, in base al segnale proveniente dai sensori di altezza. In questi casi, per garantire che le condizioni di scelta siano mutuamente esclusive, le transizioni comprendono il negato delle variabili che definiscono le condizioni precedenti. Infatti, la transizione di sinistra presenta il contatto normalmente aperto dei pacchi di medie dimensioni ed il contatto normalmente chiuso delle altre due tipologie di pacchi; gli altri due rami saranno il viceversa, così da coprire tutte le possibili combinazioni.

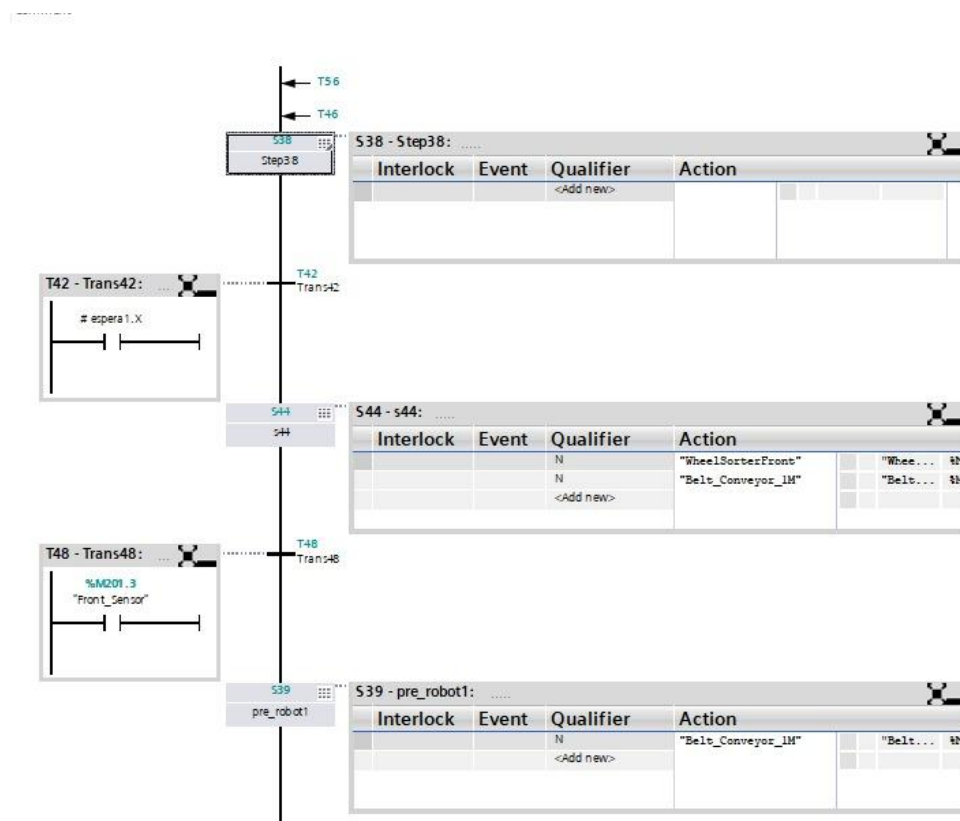


Figura 23: SFC del WheelSorter parte 2

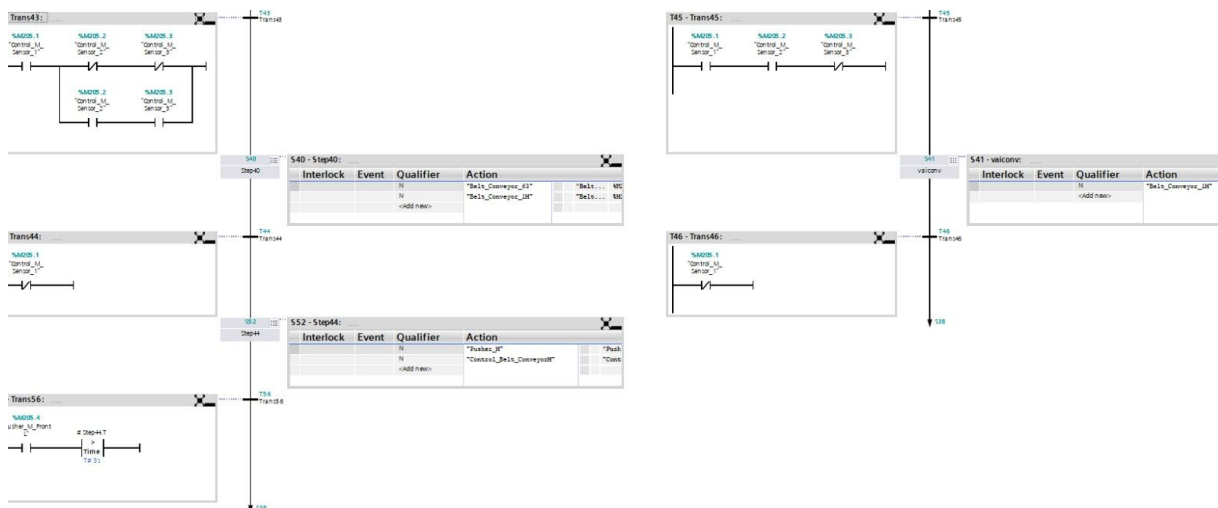


Figura 24: SFC del WheelSorter parte 3

È fondamentale tenere in considerazione che i codici sono stati realizzati uno per ciascuna macchina, così da evitare la possibile sovrapposizione del segnale quando due pacchi passano consecutivamente uno all'altro. Così facendo, si attiva un Conveyor non appena il relativo sensore rileva la presenza del pacco. Per la comunicazione tra un conveyor e l'altro sarà cruciale inserire negli step iniziali, o il sensore di riferimento o lo step.X. In questo contesto, .X indica che la condizione successiva si attiverà non appena sarà superata la fase di riferimento.

Tale suddivisione, dunque, è necessaria perché, se si utilizzasse un codice unico per tutto il sistema, si rischierebbe la sovrapposizione dei segnali che potrebbe portare allo stop per errore del sistema.

Si riportano di seguito le immagini del codice che evidenziano quanto appena detto.

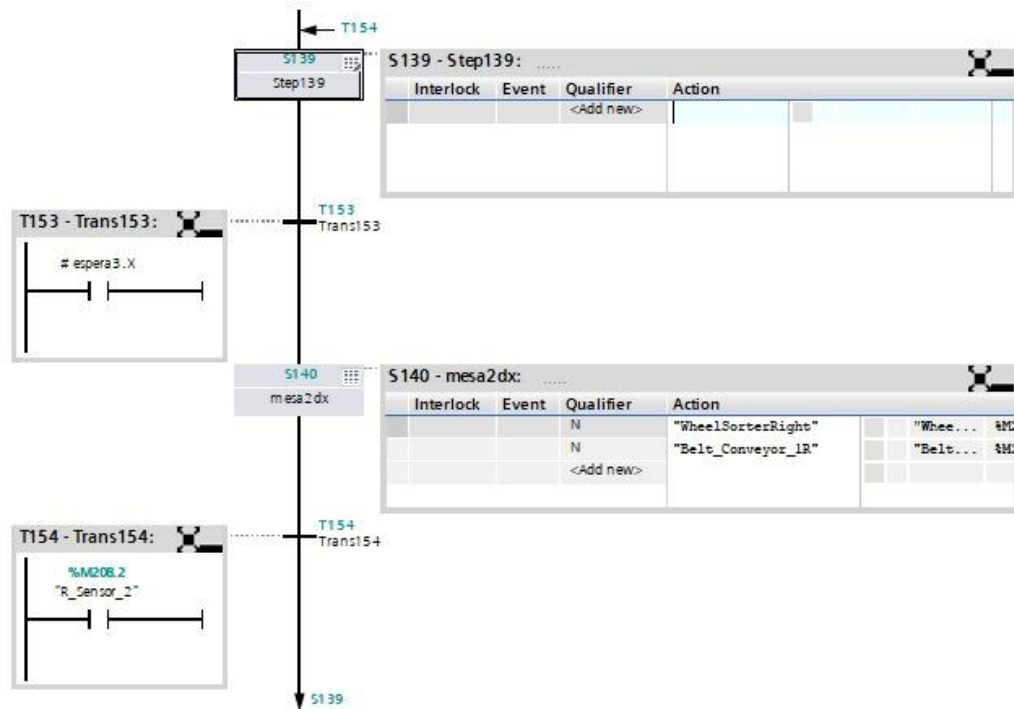


Figura 25: Conveyor iniziale del tratto di destra

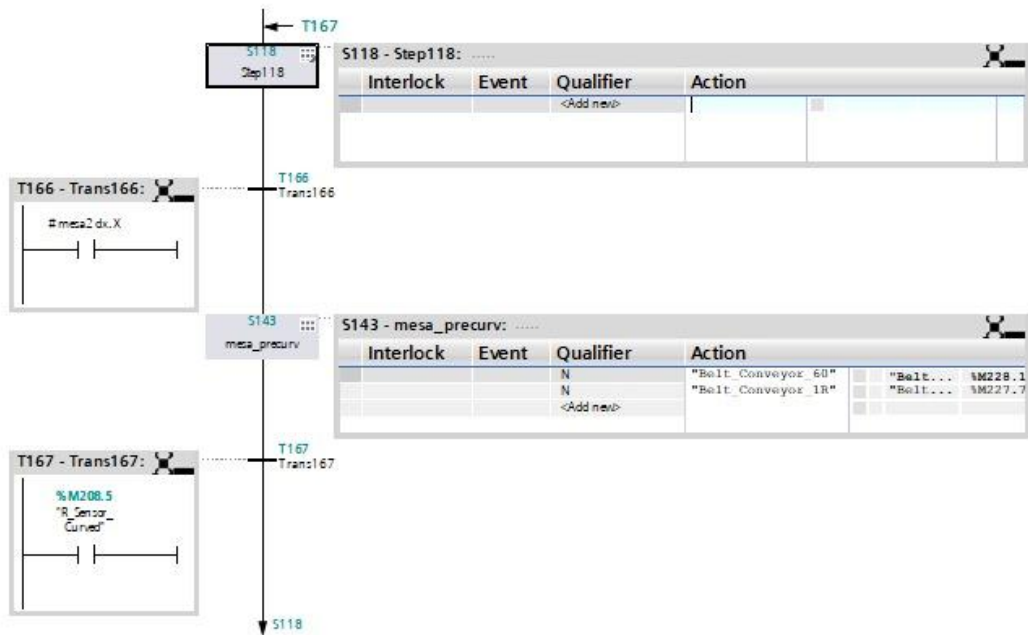


Figura 26: Seconda parte dei conveyor di destra

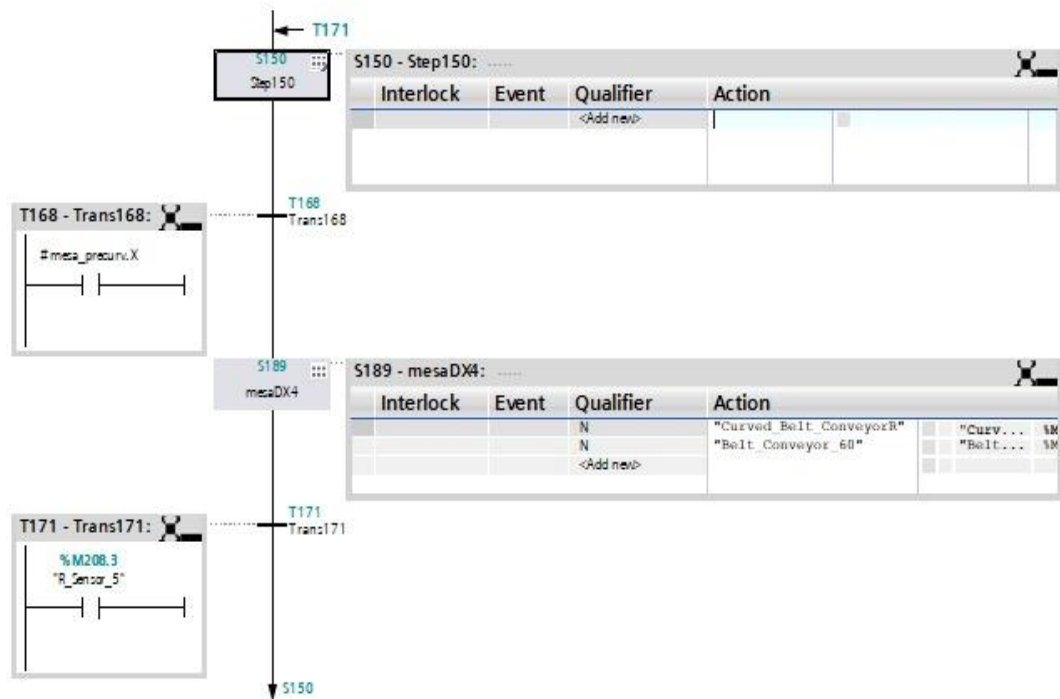


Figura 27: Terza parte dei conveyor di destra

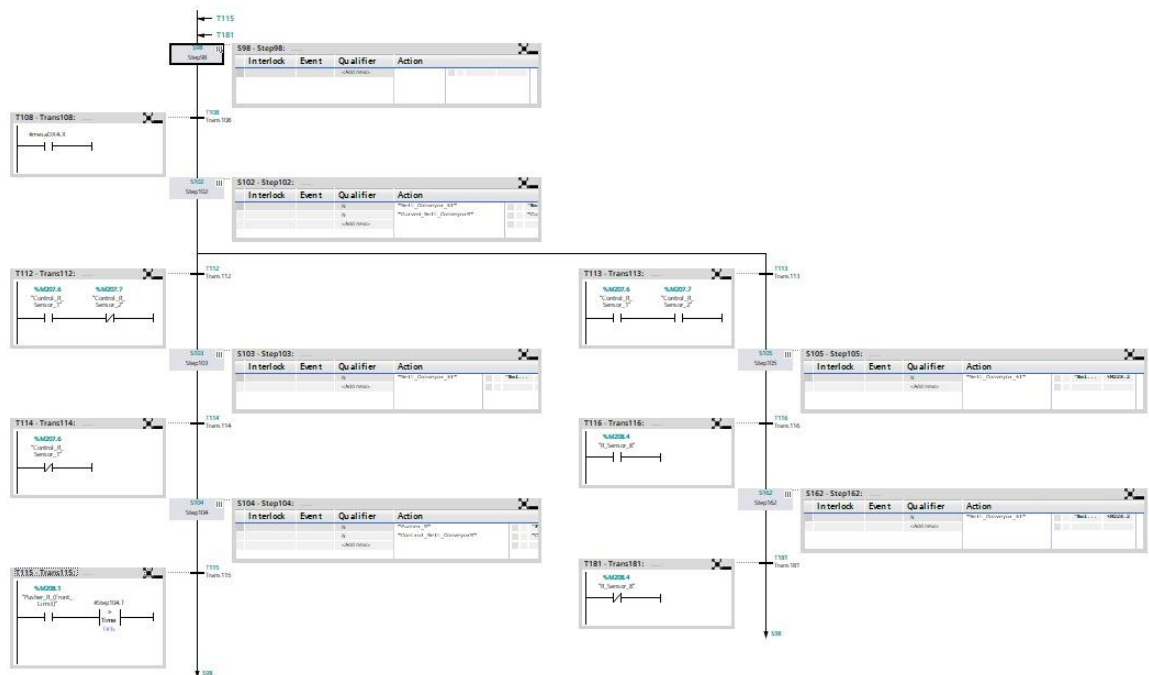


Figura 28: Quarta parte dei conveyor di destra

Nella parte successiva al WheelSorter, prima di raggiungere il Robot_1, è presente il pusher. Come si evince dal codice, il compito del pusher è quello di deviare sul conveyor di uscita i pacchi non corretti, in base ai sensori che si attivano. Al contrario, se il pacco è corretto, si attiverà l'altro ramo previsto dal programma,

consentendo al pacco di proseguire lungo il nastro fino a raggiungere il robot a tre assi.

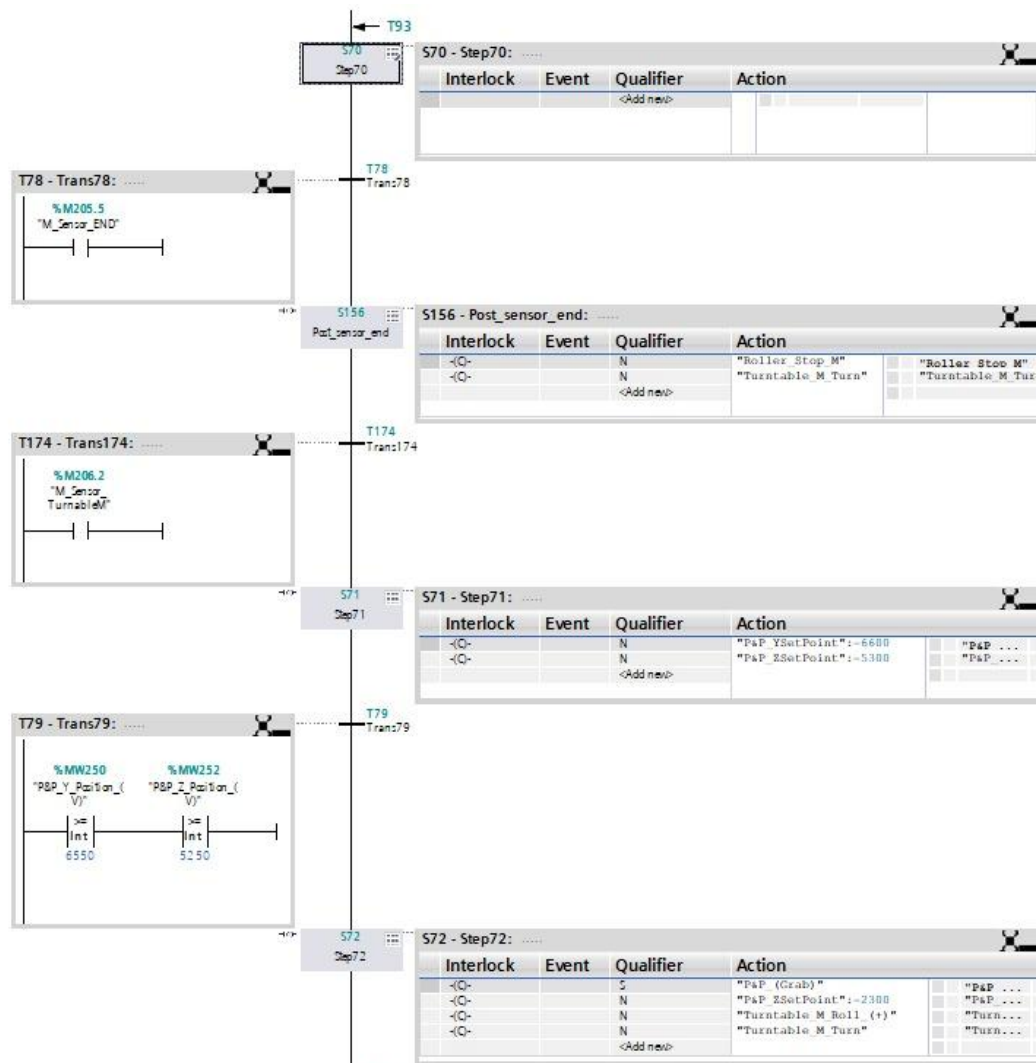


Figura 29: SFC Robot 1 parte 1

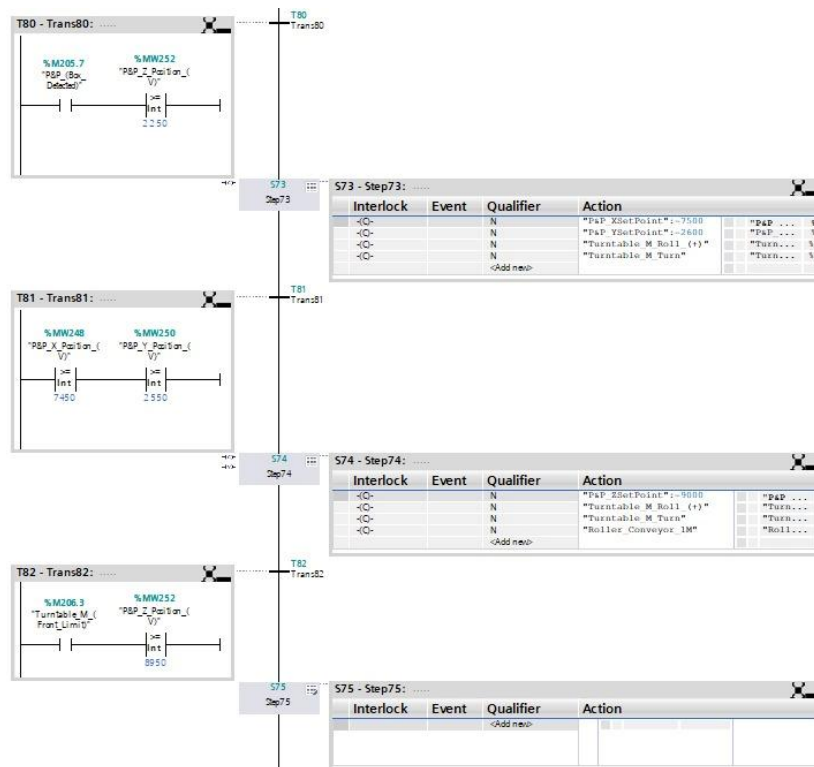


Figura 30: SFC Robot 1 parte 2

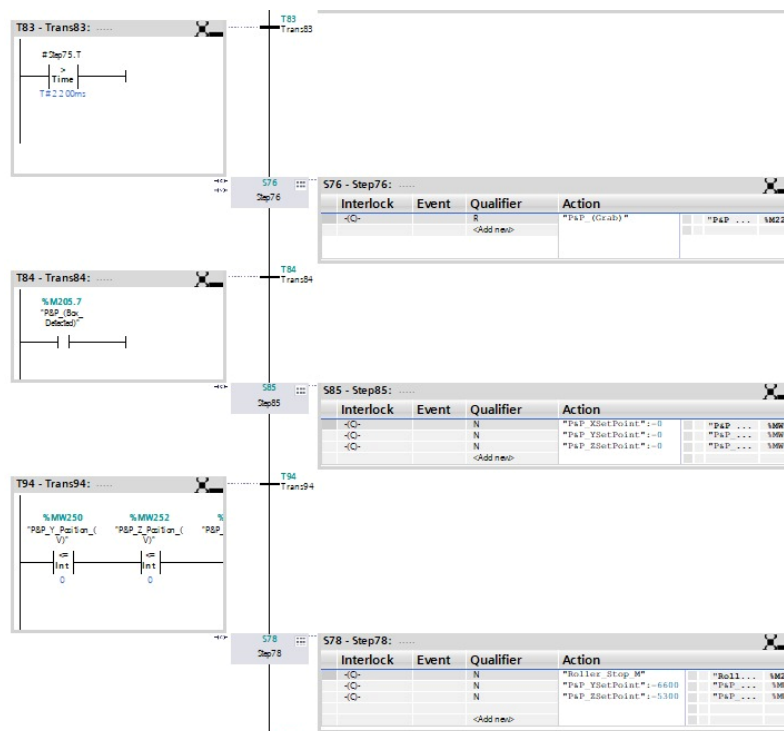


Figura 31: SFC Robot 1 parte 3

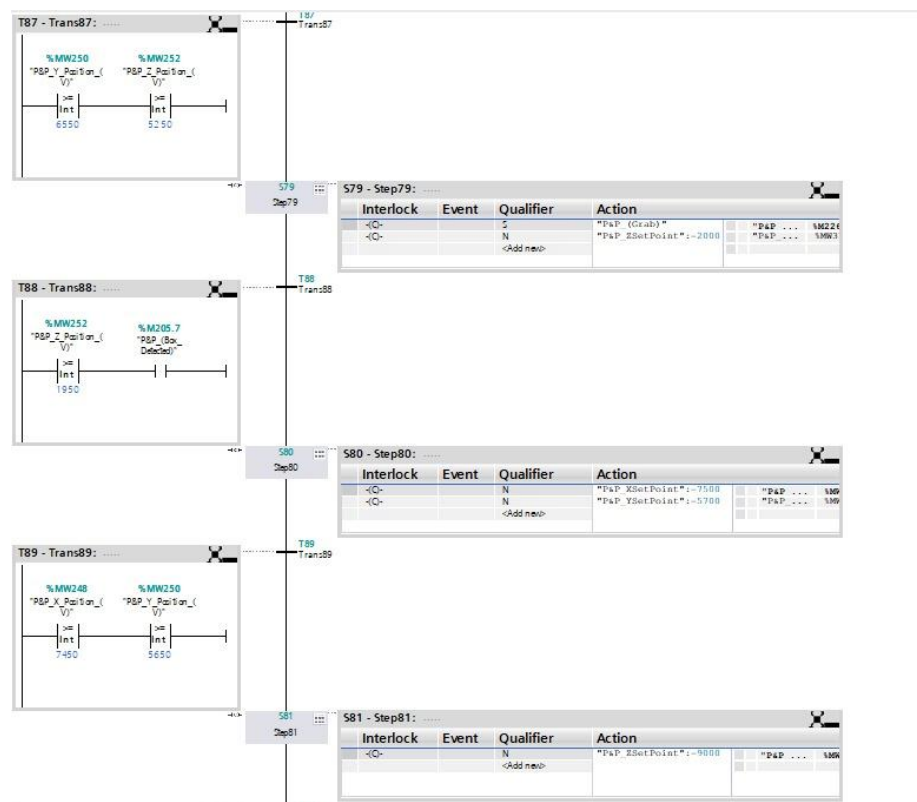


Figura 32: SFC Robot 1 parte 4

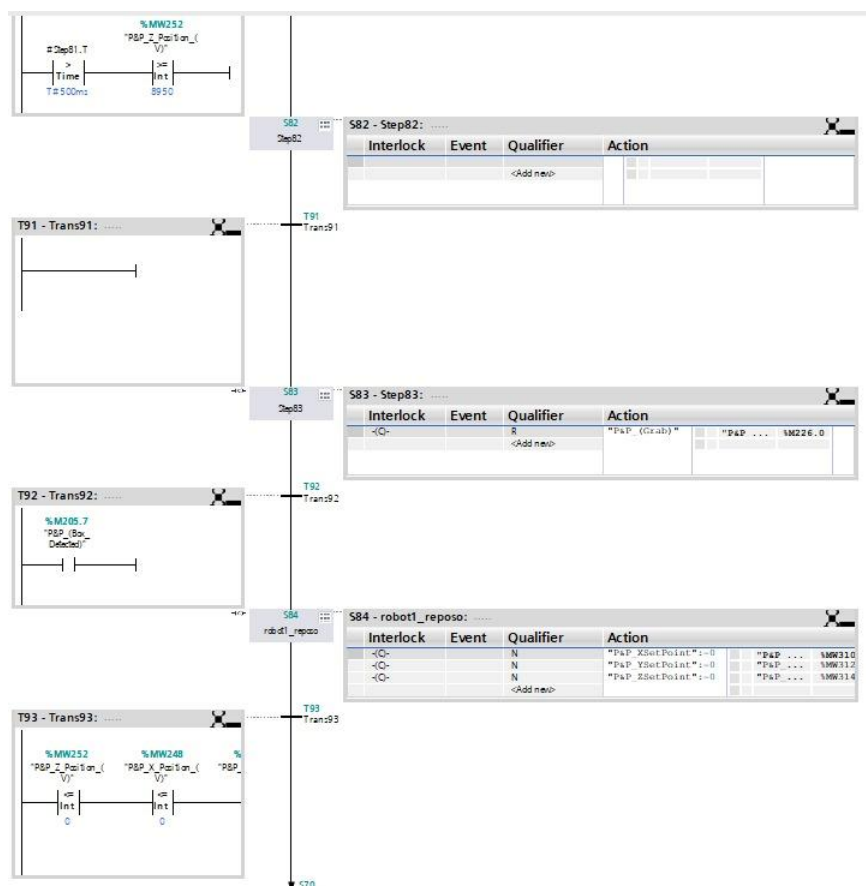


Figura 33: SFC Robot 1 parte 5

In questa parte di codice, oltre alle nozioni dette prima, è presente il temporizzatore e, successivamente, i comandi di Set_point e Position. I comandi di Set_point servono per la movimentazione del robot in esame, invece, Position è il comparatore di controllo della posizione impostata. È fondamentale che il comparatore venga inserito opportunamente nella transizione successiva alla fase di movimentazione del robot. Un comparatore è un elemento logico utilizzato per confrontare due valori e determinare se una certa condizione è soddisfatta. Questo confronto restituisce un risultato booleano, cioè vero o falso. A differenza dei comparatori, i temporizzatori sono dispositivi di controllo a tempo; possono essere elettrici o elettronici. Sono dotati di un cronometro, che serve a quantificare l'intervallo di tempo che intercorre tra due o più operazioni. A tal riguardo, possono essere divisi in due grandi categorie: quelli che ritardano l'attivazione di un'uscita dopo che l'ingresso è diventato attivo (TON) e quelli che mantengono l'uscita attiva per un certo tempo dopo che l'ingresso è stato disattivato (TOF).

Nel caso del progetto in esame, sono stati utilizzati unicamente temporizzatori di tipo TON. [8]

Subito dopo il Robot_1, come si può vedere dall'immagine del sistema generale, è presente un turntable, il quale, in base alla fase in cui si trova il robot, ha il compito di ruotare non appena si attiva il sensore "M_sensor_turntable". A quel punto si attiveranno in contemporanea il roller conveyor di prima e quello del turntable stesso, così da permettere l'ingresso del pallet. Una volta che il pallet ha raggiunto il sensore di Back_limit, il turntable potrà ruotare nuovamente e tornare alla posizione di partenza, in attesa che il robot completi le operazioni di pick and place per i due pacchi di riferimento. Il robot 1, infatti, deve prima posizionare un pacco sulla destra del pallet e successivamente posizionarne un altro sulla sinistra, senza creare sovrapposizioni e rispettando i limiti del pallet. Una volta caricato opportunamente, il pallet sarà guidato verso l'uscita del turntable attraverso la simultanea attivazione del roller del turntable e dell'altro conveyor.

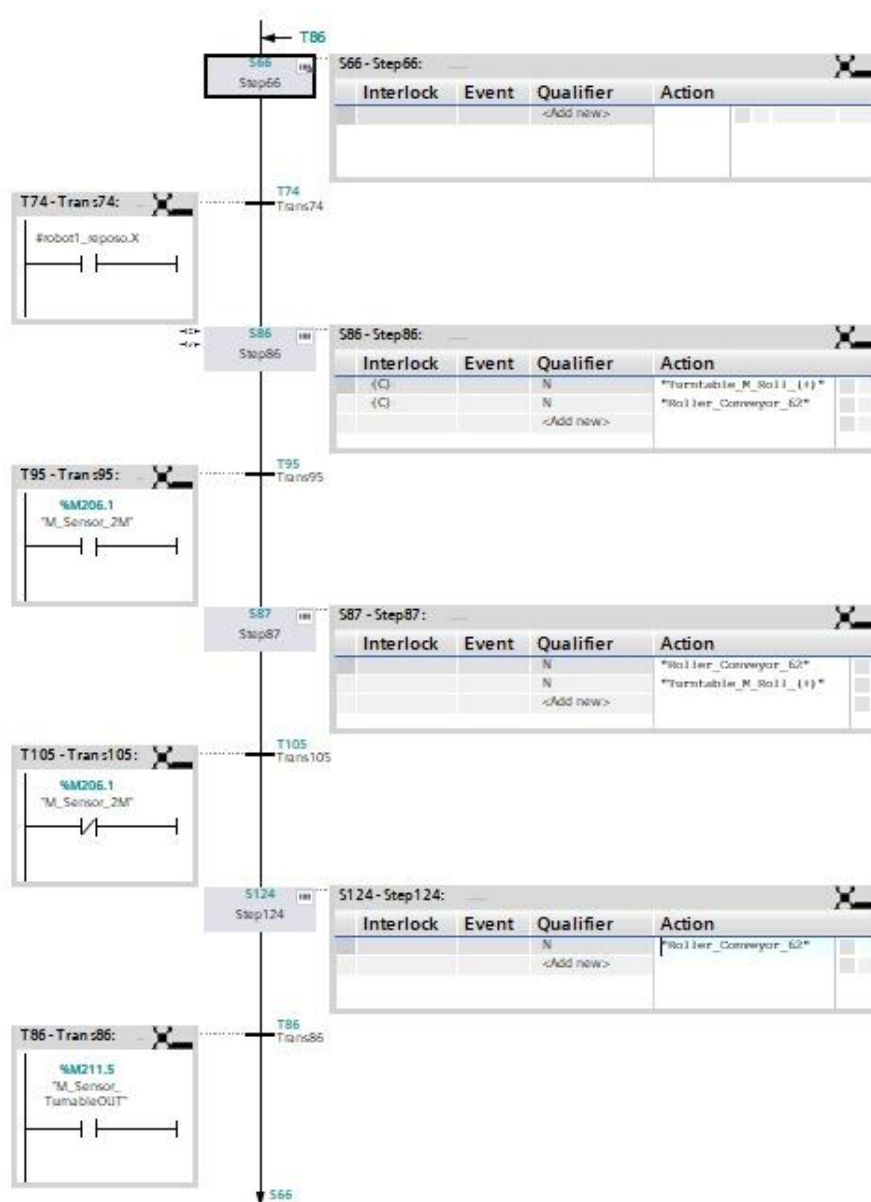


Figura 34: SFC del Turntable

Adesso si tratterà la parte di codice relativa al palletizzatore, il quale è adibito all'impilamento dei pacchi sul pallet di riferimento, in modo da creare 3 colonne da 2 pacchi ciascuno. Per far in modo che esso funzioni correttamente, sarà fondamentale l'uso del contatore. Il contatore è un dispositivo utilizzato, come dice la stessa parola, per contare eventi all'interno di un ciclo di lavoro. Serve, ad esempio, per tenere traccia di quante volte si è verificata una determinata condizione, come il passaggio di un oggetto davanti ad un sensore. I contatori possono essere classificati in due grandi famiglie: i CTU e i CTD. I CTU sono progettati in modo tale da contare in avanti ogni volta che riceve un impulso in ingresso, quando raggiunge un valore preimpostato, attiva l'uscita. Invece, i CTD partono da un valore predefinito che scende verso lo zero ad ogni impulso.

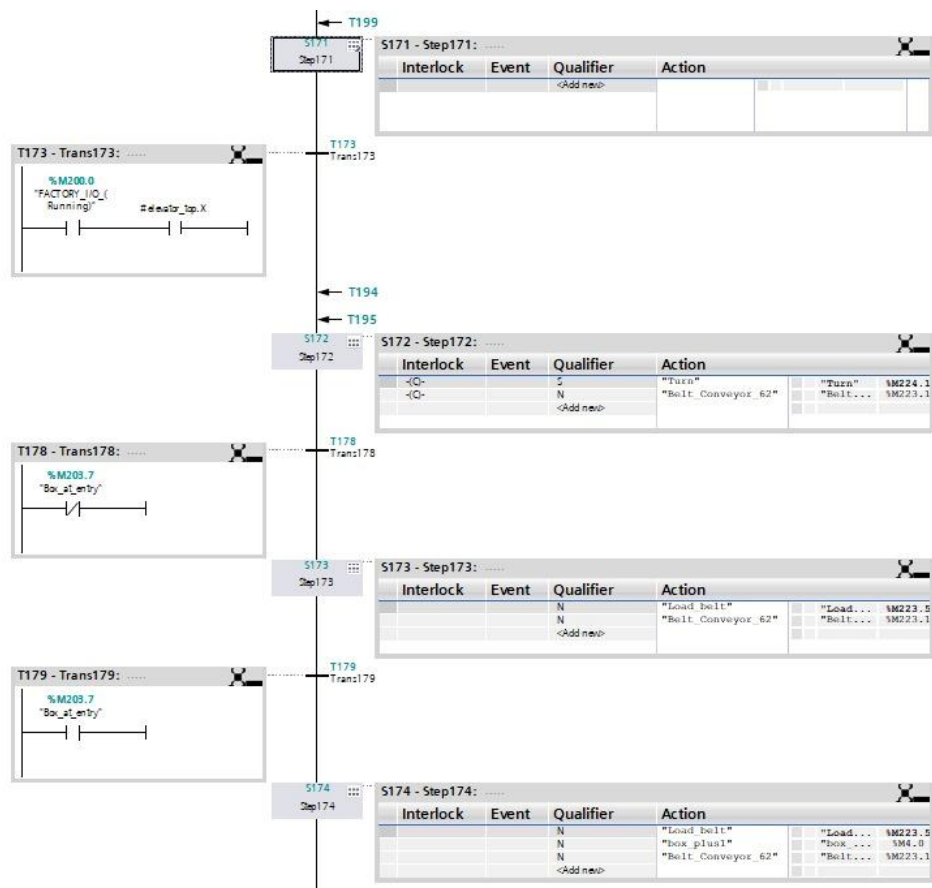


Figura 35: SFC del palletizzatore parte 1

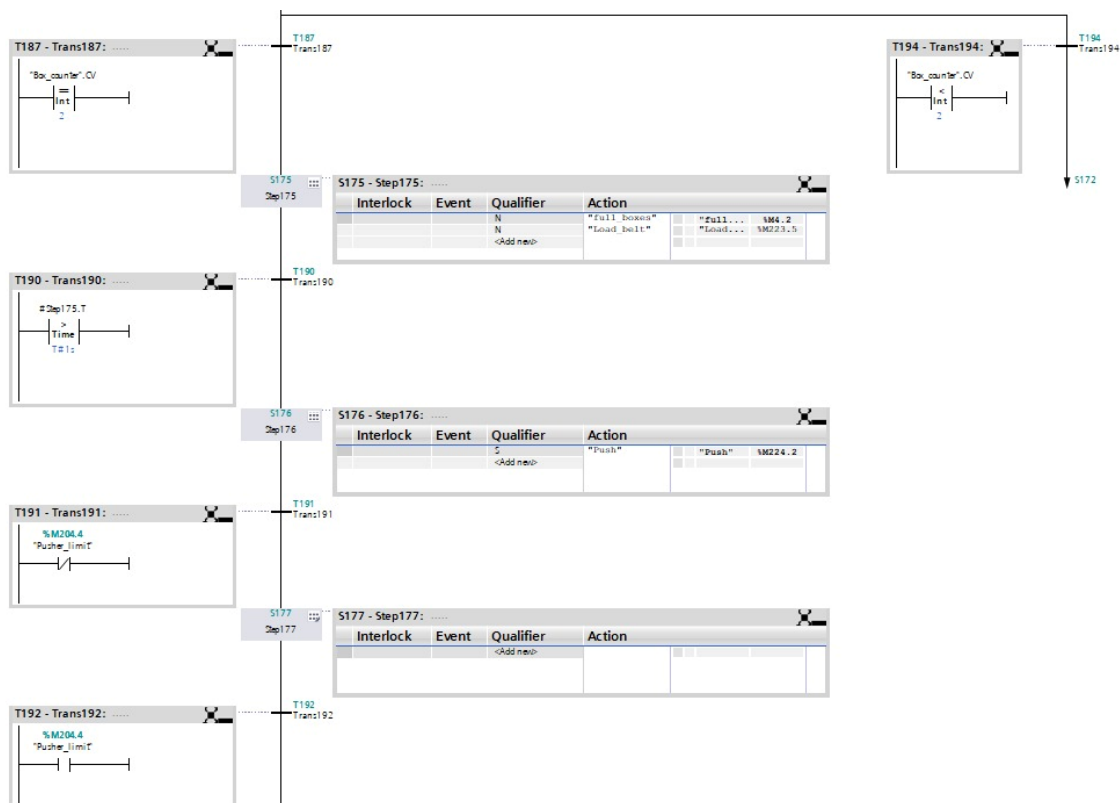


Figura 36: SFC del palletizzatore parte 2

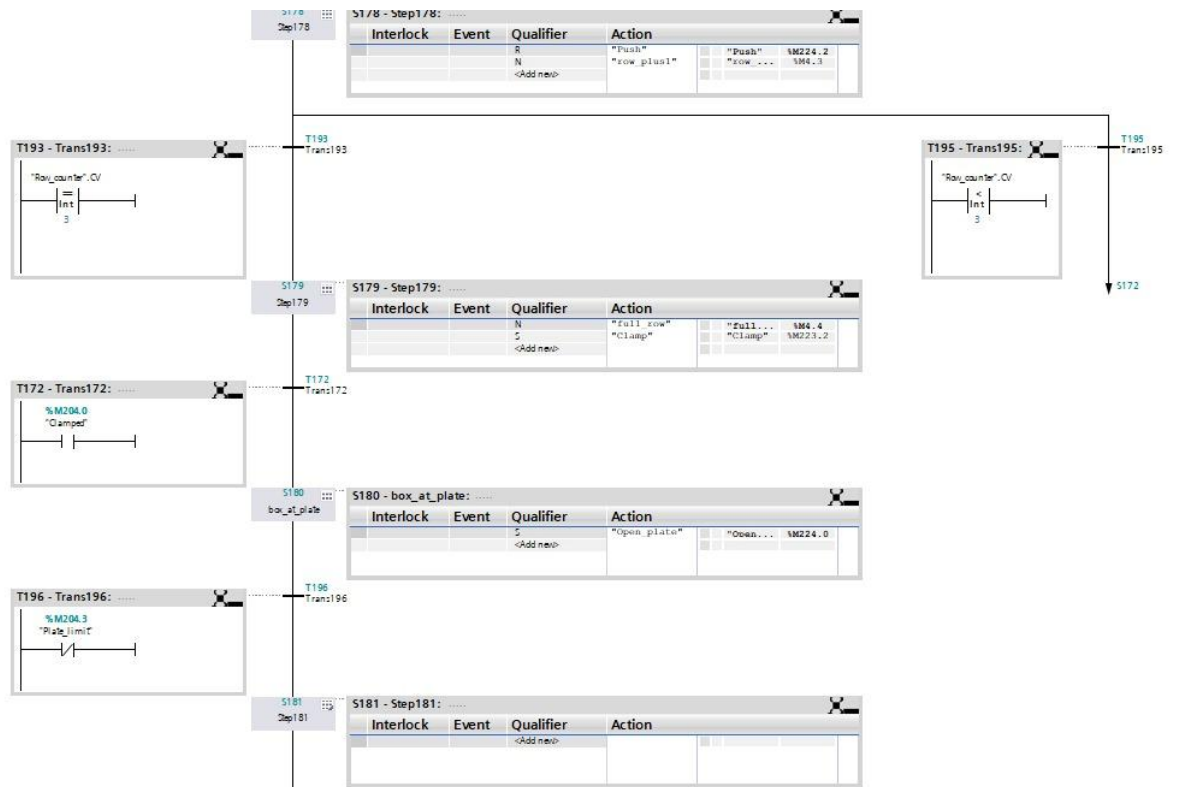


Figura 37: SFC del palletizzatore parte 3

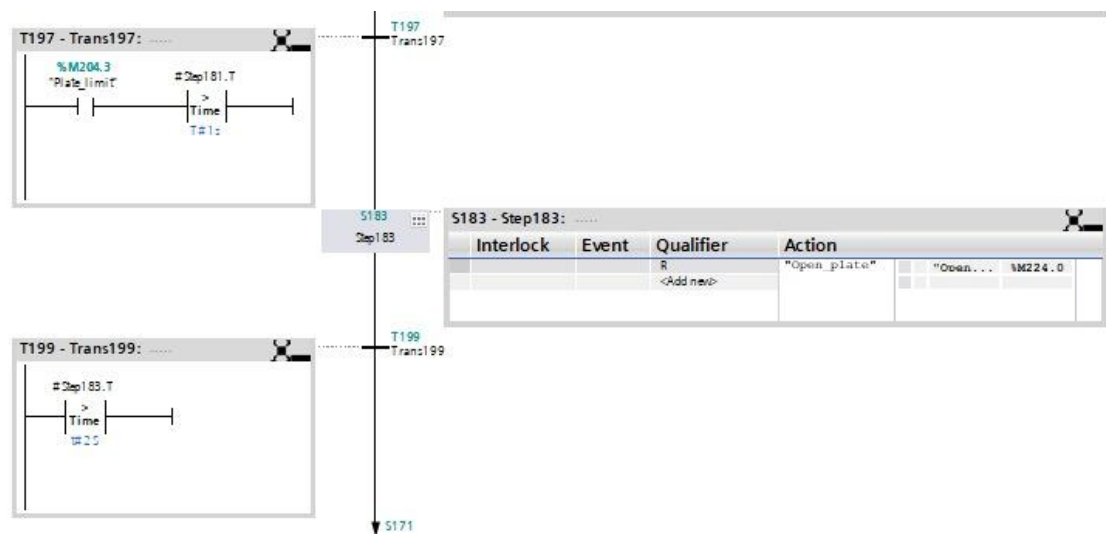


Figura 38: SFC del palletizzatore parte 4

I contatori su TiaPortal verranno visualizzati nel seguente modo:

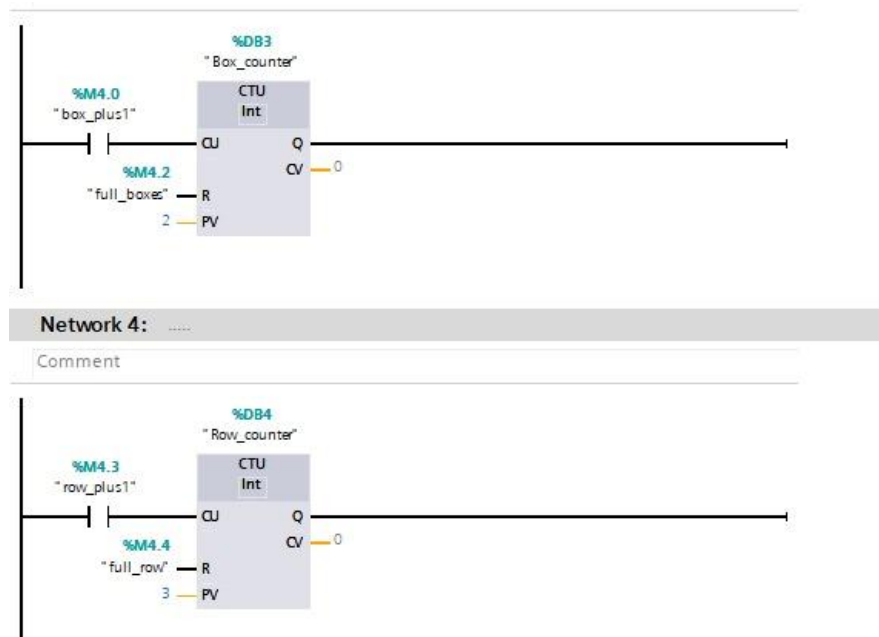


Figura 39: Contatori del palletizzatore

Il primo contatore ha il compito di registrare il numero di pacchi che superano il Turn, elemento adibito al loro posizionamento. Il secondo, invece, ha il compito di contare quante volte viene attivato il push, e ha la funzione di spostare i pacchi sulla piattaforma. Solo quando entrambi i contatori torneranno a zero, si attiverà l'open plate, così da far cadere i pacchi sul pallet.

Viene riportata unicamente la parte di codice relativa al palletizzatore; l'uscita e l'elevatore, invece, sono gestiti in un'altra sezione del programma, ma tutte e tre le parti sono collegate tra loro e concorrono al funzionamento complessivo del sistema.

Avendo trattato tutti gli elementi principali del codice, si passa al contestualizzare del Main con il blocco DB.

Tutte le informazioni ed i dati relativi a questa parte di codice vengono memorizzate in un blocco chiamato "Nastro" che è presente nel Data Block "Nastro_DB". Tale blocco permette di controllare e monitorare il comportamento del sistema, rendendo il codice ordinato.

Nella parte sinistra del blocco si trovano tutti gli ingressi, cioè i segnali che provengono principalmente da sensori, invece, la parte destra rappresenta le uscite del blocco, cioè le informazioni restituite dal sistema in risposta agli ingressi.

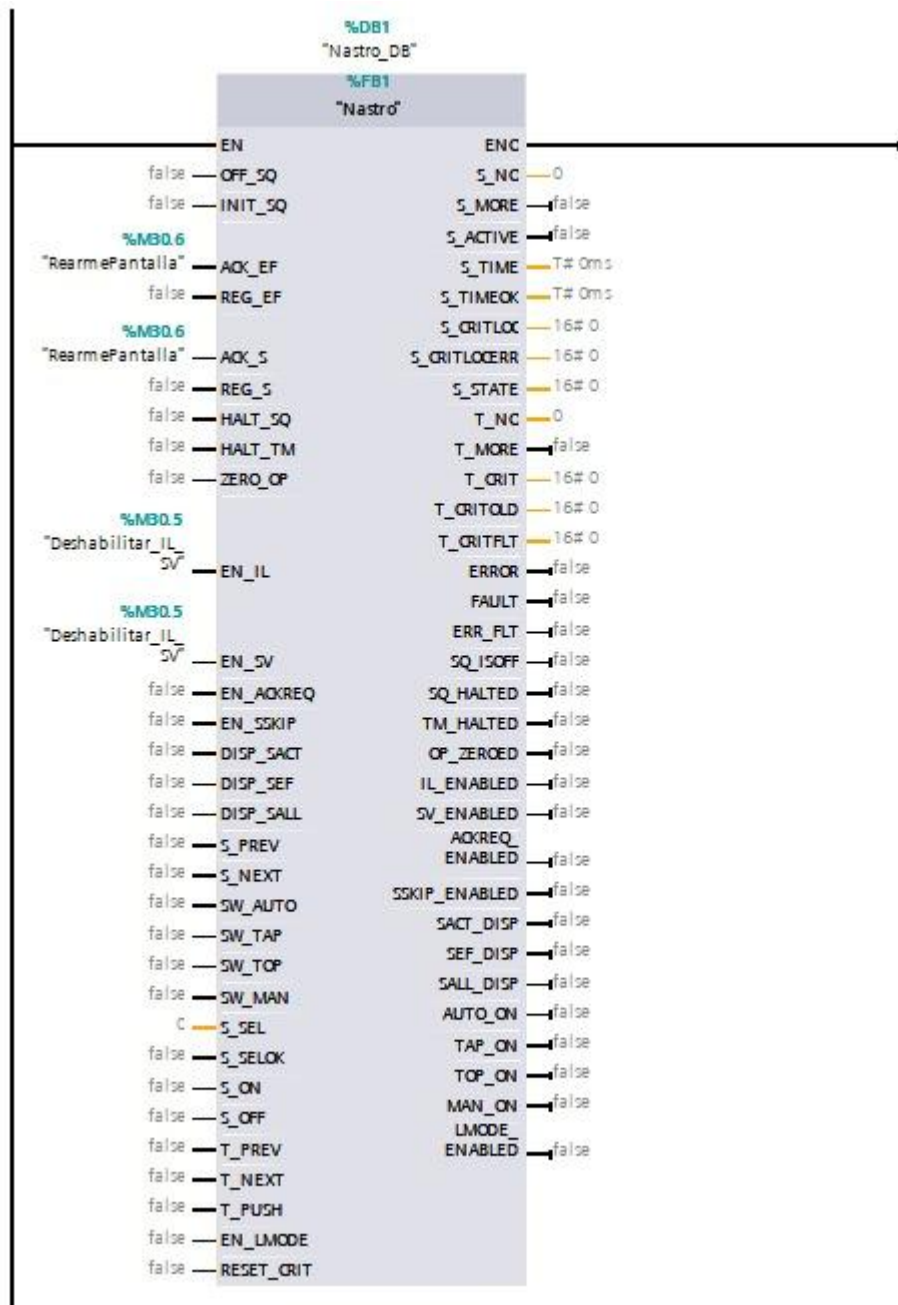


Figura 40: Main "Nastro"

Le variabili collegate al blocco verranno approfondite in un secondo momento, quando si andrà a trattare la parte manuale dell'HMI.

Successivamente, è stato creato un altro blocco di codice SFC così da implementare il codice dell'almacén in ingresso ed in uscita.

Come accennato nel paragrafo precedente, l'almacén è la parte del sistema destinata allo stoccaggio temporaneo dei pacchi. Serve per mantenere ordinati i prodotti in attesa che vengano prelevati dall'altro Almacén di uscita.

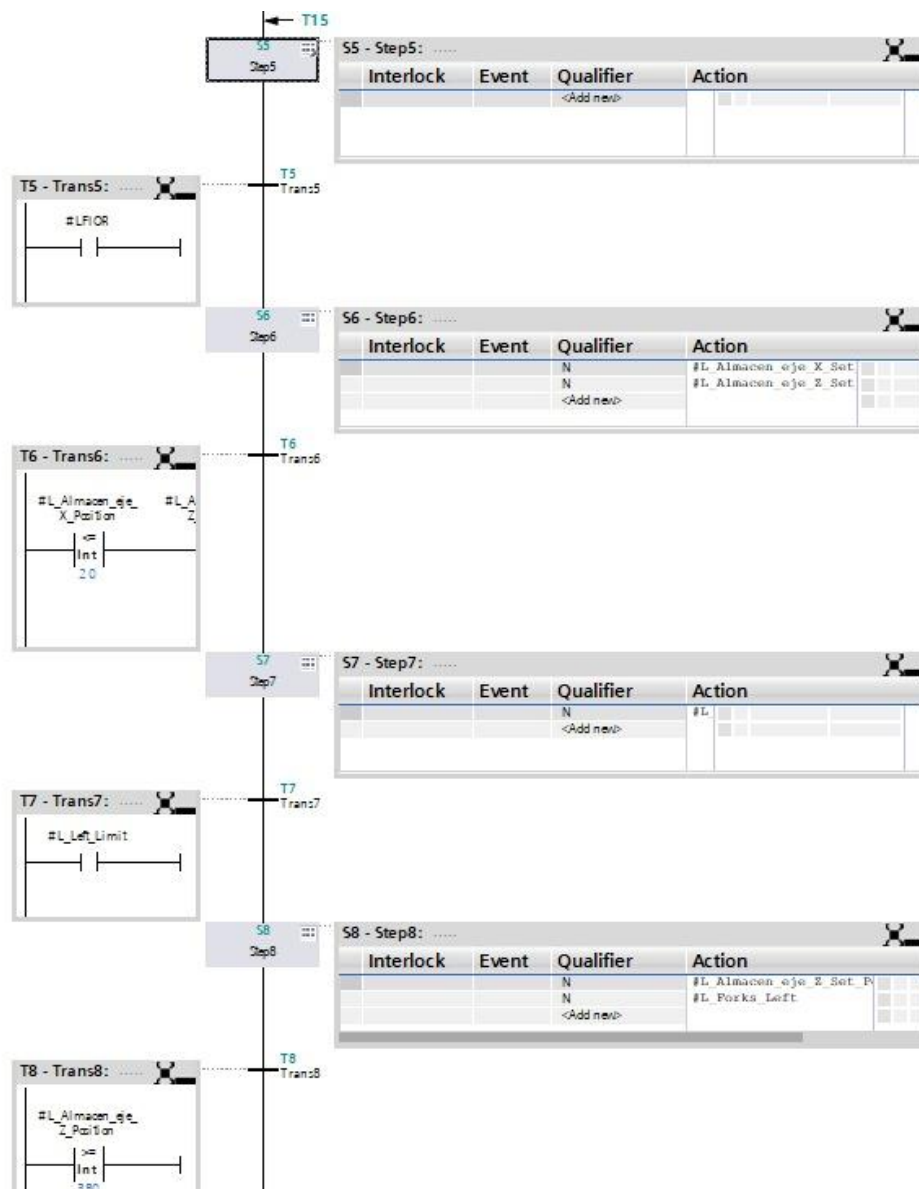


Figura 41: SFC dell'Almacen parte 1

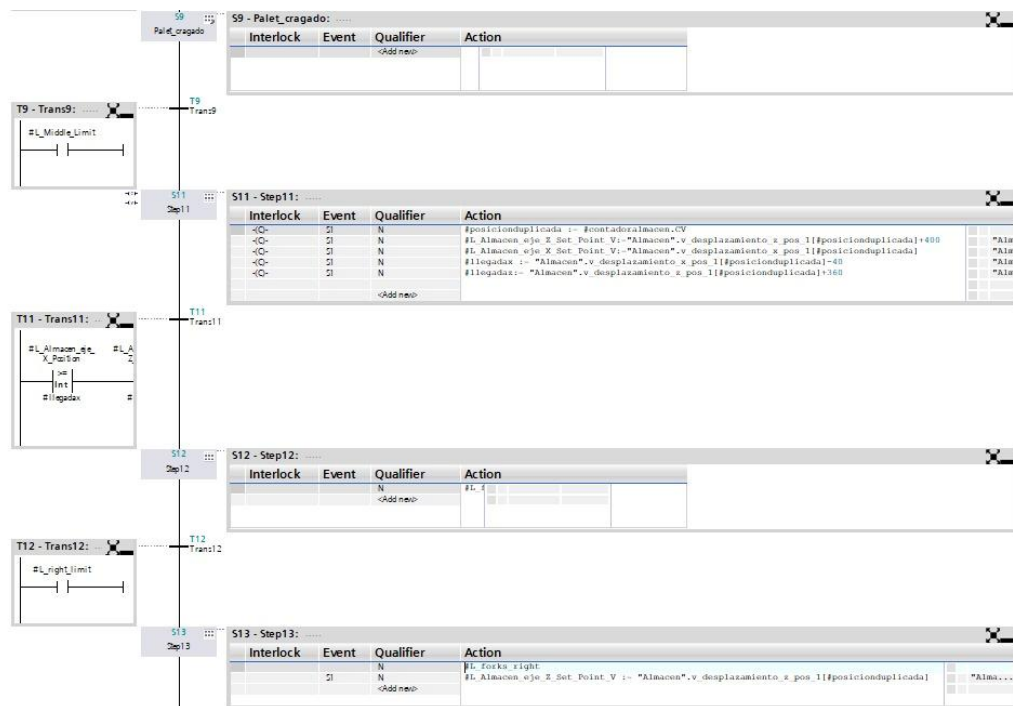


Figura 42: SFC dell'Almacen parte 2

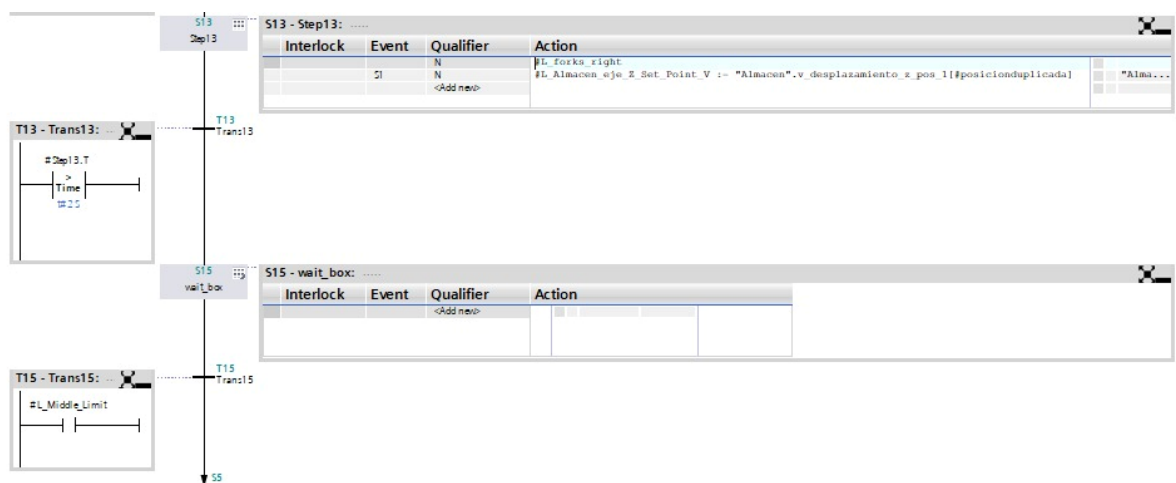


Figura 43: SFC dell'Almacen parte 3

Viene riportato unicamente il codice dell'almacén in ingresso in quanto quello in uscita presenta le stesse informazioni di base, ma semplicemente cambia il posizionamento dei blocchi.

A differenza dei codici presenti sul blocco "Nastro", quelli relativi agli almacén sono caratterizzati da variabili locali. Questa scelta consente di gestire le informazioni che fanno riferimento unicamente al funzionamento interno del blocco in esame, evitando che siano visibili dall'esterno. Così facendo, si riesce a non appesantire eccessivamente la memoria globale con dati temporanei, rendendo il codice più efficiente e sicuro.

Per questo elemento del sistema, oltre al codice SFC a blocchi, bisogna realizzare un ulteriore blocco FC.

Il codice rappresentato nell'immagine sottostante è scritto in Structured Text ed è utilizzato per misurare la disposizione spaziale dei pacchi. Avendo preso visione dell'almacén su Factoryio, è possibile impostare la posizione di partenza, visualizzabile dalla riga 1 alla 4, invece, dalla riga 6 alla 8, vengono indicate le distanze tra le colonne e tra le file del magazzino. Successivamente, si riportano il numero di righe e colonne che verranno prese in considerazione durante la simulazione.

La parte più importante di questo Text è il doppio ciclo for, in cui il primo è dedicato alle colonne ed il secondo alle righe. In questi cicli for, vengono calcolate le nuove coordinate, per ogni ciclo, sommando alle coordinate precedenti la nuova distanza tra colonne o righe, così da elaborare progressivamente tutte le posizioni possibili dove depositare e prelevare pacchi. L'indice che registra tutte le posizioni è l'Almacén_i_pos, il quale aumenta ad ogni iterazione, così da riempire i vettori delle posizioni relativi all'asse X e Y.

```

1  "Almacen".v_desplazamiento_x_pos_l[0] := 0;
2  "Almacen".v_desplazamiento_z_pos_l[0] := 0;
3  "Almacen".v_desplazamiento_x_pos_l[1] := 1430;
4  "Almacen".v_desplazamiento_z_pos_l[1] := 300+(2*1700);
5
6  "Almacen".Almacen_v_distanciaEntreColumnas := 1070;
7  "Almacen".Almacen_v_distanciaEntreFilas := 1700;
8  "Almacen".Almacen_v_coger_pallet := 400;
9
10 "Almacen_n_filas" := 3;
11 "Almacen_n_columnas_palletizer" := 9;
12
13 "Almacen_i_pos" := 1;
14 FOR "Almacen_i_x" := 0 TO "Almacen_n_columnas_palletizer" - 1 DO
15
16     FOR "Almacen_i_z" := 0 TO "Almacen_n_filas" - 1 DO
17         "Almacen".v_desplazamiento_x_pos_l["Almacen_i_pos"] := "Almacen".v_desplazamiento_x_pos_l[1] +
18         "Almacen_i_x" * "Almacen".Almacen_v_distanciaEntreColumnas;
19         "Almacen".v_desplazamiento_z_pos_l["Almacen_i_pos"] := "Almacen".v_desplazamiento_z_pos_l[1] -
20         "Almacen_i_z" * "Almacen".Almacen_v_distanciaEntreFilas;
21         "Almacen_i_pos" := "Almacen_i_pos" + 1;
22     END_FOR;
23
24 END_FOR;

```

Figura 44: FC dell'Almacen

Esattamente come la prima parte, è necessaria la progettazione di un “Data_Block” adibito alla registrazione e gestione delle informazioni dei due magazzini.

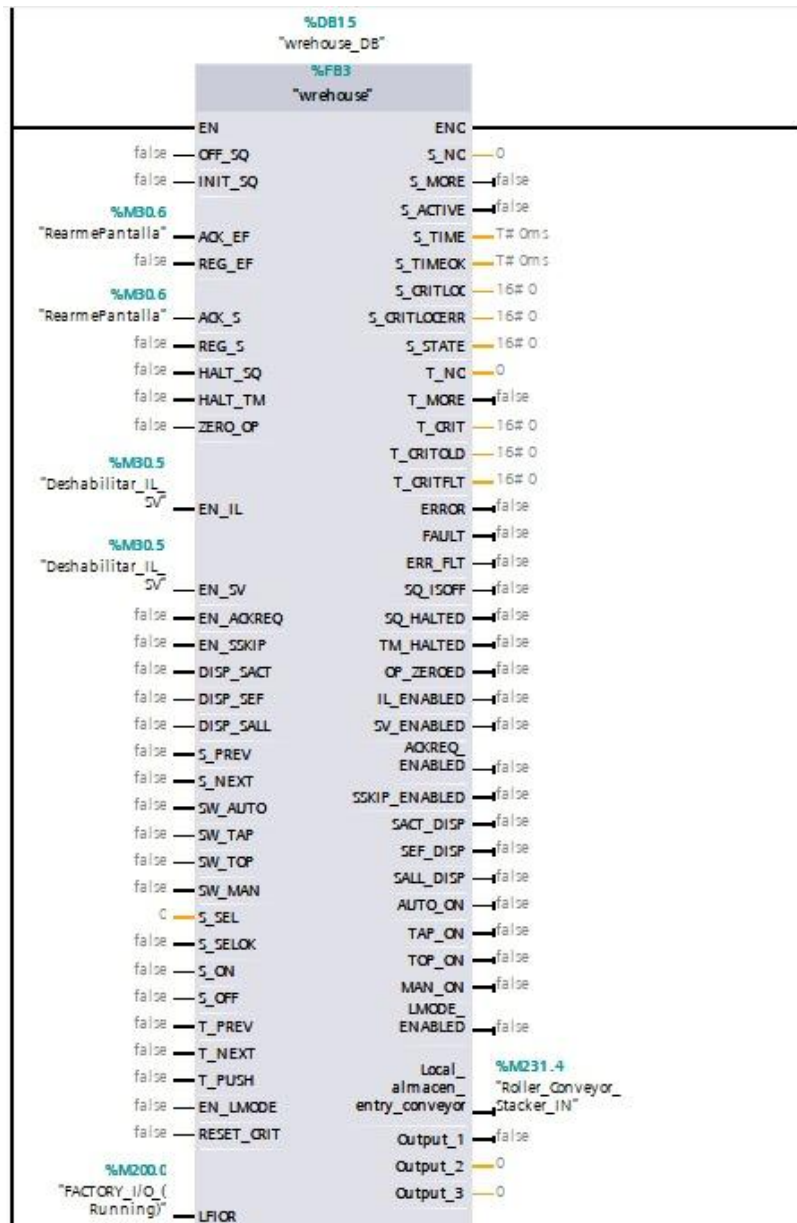


Figura 45: Main dell'Almacen parte 1

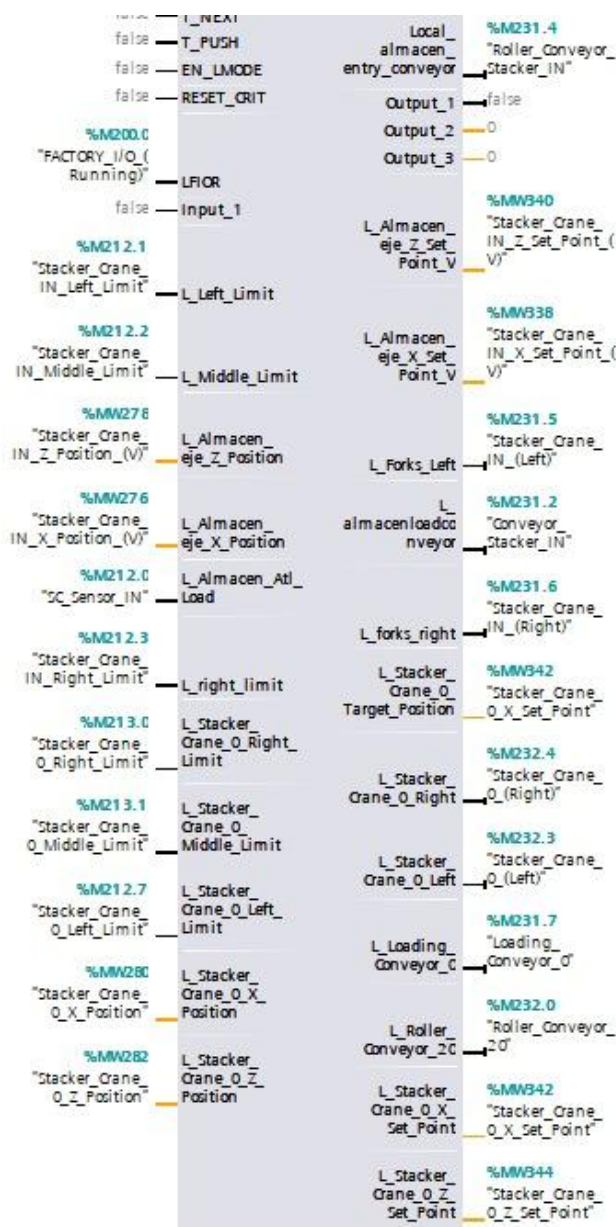


Figura 46: Main dell'Almacen parte 2

Stavolta, oltre a presentare i collegamenti iniziali, che verranno trattati successivamente, sono state inserite tutte quelle informazioni di posizione e movimento dell'almacén di ingresso e uscita nella parte destra, ed i relativi sensori nella parte di sinistra. Ogni variabile locale è collegata alla sua variabile globale, per far comprendere al sistema quale sia la variabile di partenza, la quale permette di comunicare con altri blocchi.

Capitolo 4

Comando Manuale attraverso l'uso dell'HMI

Nei capitoli precedenti si è trattato di come automatizzare il sistema attraverso la scrittura di codici SFC su TiaPortal. Adesso si parlerà dell'importanza di poter controllare la linea di trasporto in esame anche attraverso comandi gestiti da un operatore opportunamente specializzato.

Negli ultimi anni si cerca sempre di più ad automatizzare il processo, così da renderlo più veloce ed efficiente, ma resta cruciale la possibilità di controllare il sistema in modo manuale, così da gestire eventuali malfunzionamenti. In questi casi, l'operatore deve avere la possibilità di intervenire direttamente per risolvere il problema. Anche durante la manutenzione è utile comandare le varie parti singolarmente e manualmente. Tutto ciò può essere gestito tramite l'HMI (Human Machine Interface), un dispositivo che permette all'operatore di comunicare direttamente con il sistema automatizzato. Esso è costituito da un display attraverso il quale l'operatore può visualizzare lo stato dell'impianto, ricevere messaggi diagnostici ed inviare comandi manuali.



Figura 47: HMI di laboratorio

La programmazione dell'HMI inizia con la creazione dell'interfaccia virtuale su TiaPortal. Per far ciò, si accede alla sezione “Add new device”, poi si va su HMI e si sceglie il modello di riferimento. Nel caso in esame, si farà riferimento a quello presente in laboratorio. Una volta selezionato il corretto HMI si può modificare e personalizzare il display come meglio si desidera. La parte più importante, prima di confermare la creazione, è quella della Root Screen, dove si creano i tre screens principali, che verranno successivamente riempiti da bottoni e luci di utilizzo.

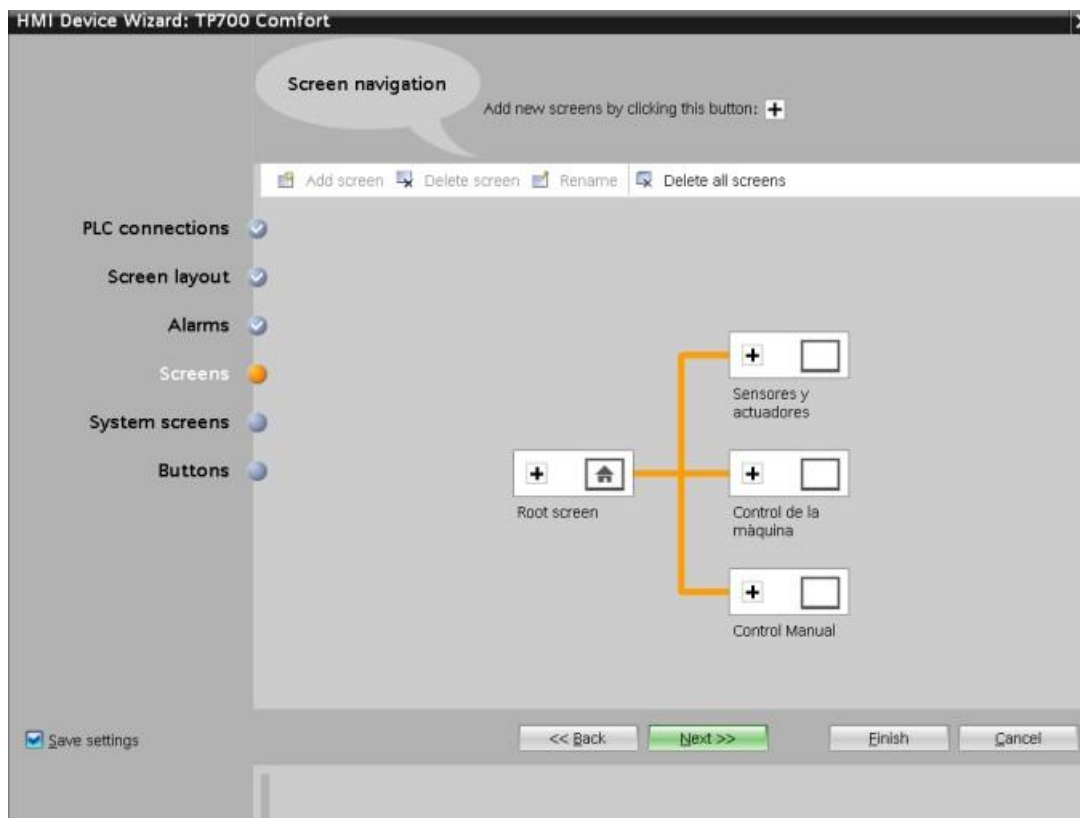


Figura 48: Progettazione dell'HMI virtuale

Avendo ufficializzato la creazione del dispositivo, si passa alla verifica della comunicazione tra il PLC e l'HMI attraverso la sezione “Device Network”. Da qui si è osservato che i due dispositivi avessero terminali differenti e, quindi, andando sulle proprietà di entrambi, è necessario cambiare manualmente l'indirizzo IP, ricordando che l'HMI del laboratorio può avere come ultimo valore solamente 220 o 221.

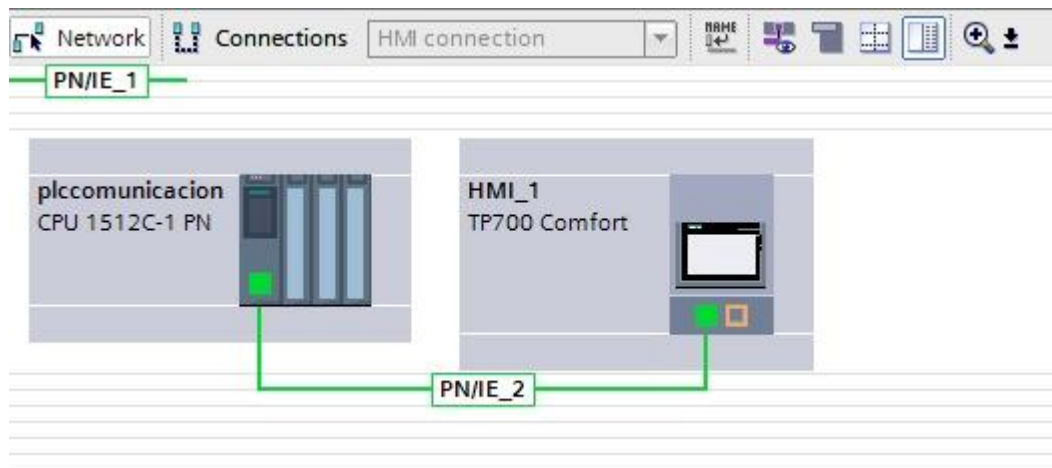


Figura 49: Comunicazione tra PLC e HMI

La progettazione dell'HMI si basa sulla realizzazione di bottoni e luci di utilizzo all'interno dei vari display. Come detto in precedenza si distinguono 3 sezioni fondamentali che sono:

- Sensori e Attuatori;
- Controllo Manuale;
- Controllo della Macchina.

È importante che queste sezioni siano opportunamente organizzate, in modo tale da permettere la comprensione e l'utilizzo anche agli operatori che non conoscono il nome delle variabili o che non conoscono al meglio la struttura del sistema in esame. Per questo motivo, verranno inseriti più screens per ogni sezione, così che ogni screen presenti le informazioni relative ad una macchina o ad una zona di lavoro ben definita.

La prima sezione, quella dei sensori e attuatori, è caratterizzata da un primo screen, chiamato Screen 0, in cui sono presenti i bottoni, ognuno dei quali, se attivato, apre lo screen di una macchina piuttosto che di un'altra. In questa sezione, ciascuna schermata relativa alle varie zone di lavoro presenta luci di utilizzo, che si accendono unicamente nel momento in cui il relativo sensore o attuttore entra in funzione durante la simulazione.

I bottoni presenti nella schermata vengono configurati accedendo alla sezione "Properties" e selezionando l'opzione "Change", dove si imposta la variabile dello schermo con cui si vuole cambiarlo. Questa configurazione consente di aprire la schermata di riferimento e mantenerla attiva fino a quando non si resetta il bit.

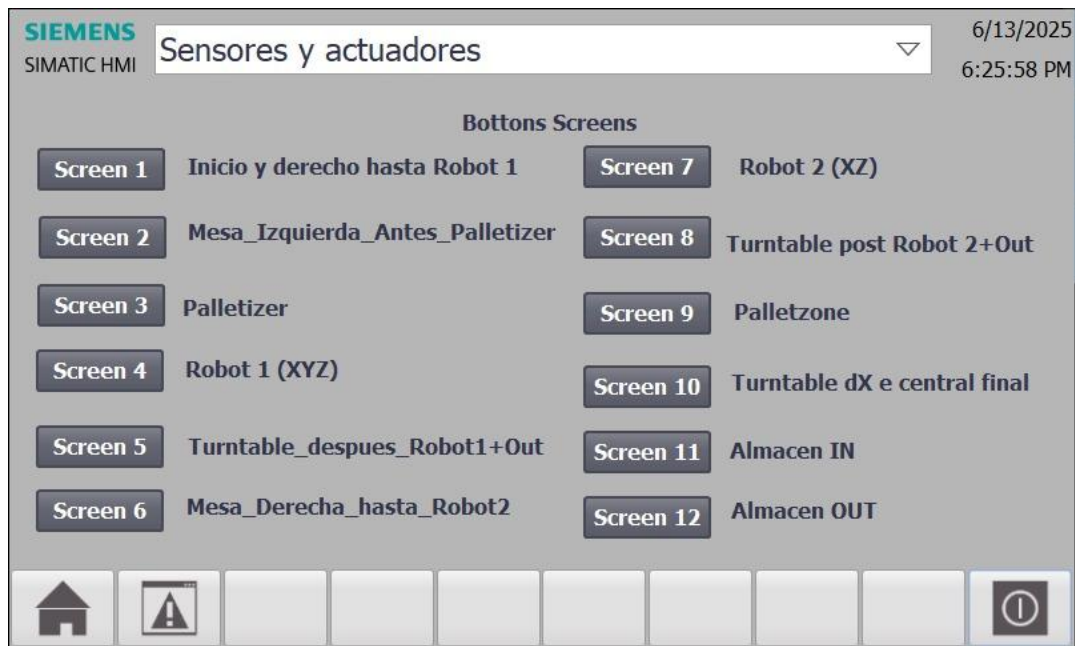


Figura 50: Screen 0 dei sensori e degli attuatori

Ogni Screens, diverso dal primo, presenta un bottone che permette di ritornare al menu principale. Tali luci sono progettate inserendo la variabile di riferimento nella sezione “Display” e poi in “Appearance”.

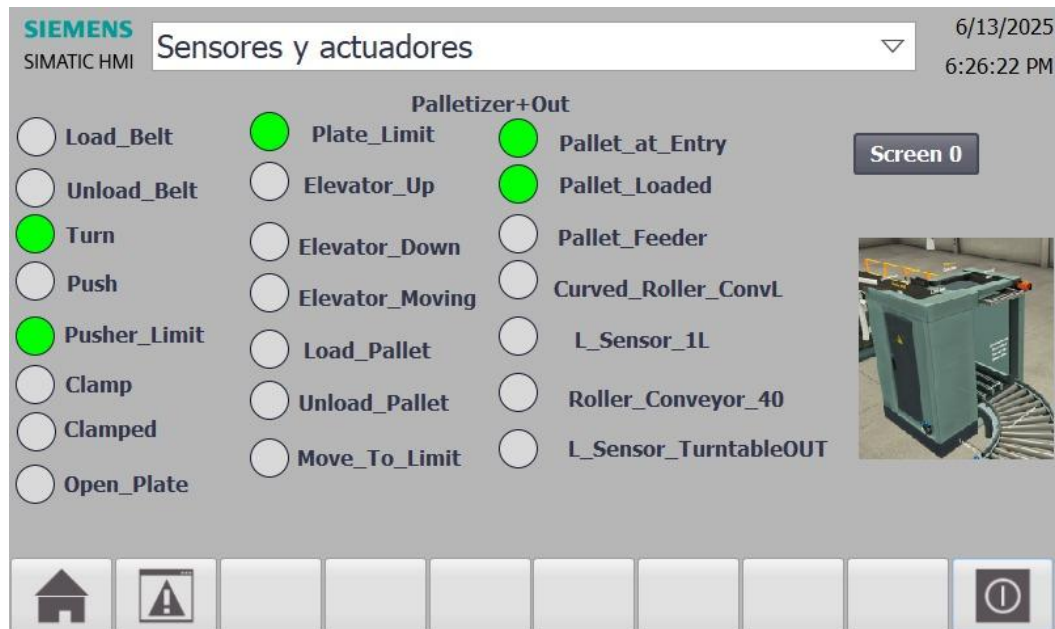


Figura 51: Screen dei sensori e degli attuatori del palletizzatore

Per la realizzazione della schermata “Control Manual”, si configurano i bottoni accedendo alla sezione “Events” e selezionando l’opzione “Edit Bit”, dove si impostano le azioni Set Bit e Reset Bit. Questa modalità permette di attivare l’attuatore di riferimento solo mentre si tiene premuto il bottone: non appena viene

rilasciato, l'attuatore si riferma. Infatti, in questa sezione si inseriscono unicamente gli attuatori delle varie macchine.

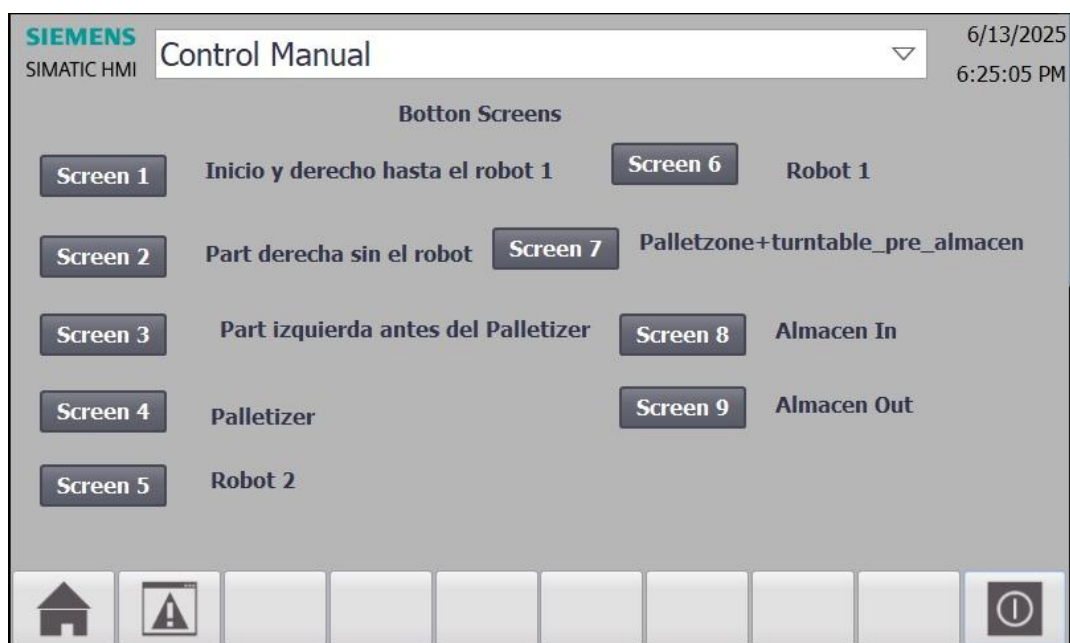


Figura 52: Screen 0 del control manual

Si riporta l'immagine dello screen 5 relativo al robot 2. Oltre ai bottoni di set bit e reset bit, sono presenti degli elementi di visualizzazione numerici che servono ad impostare un valore di spostamento del robot lungo uno dei due assi manualmente. Inoltre, come detto nella sezione dei sensori e attuatori, anche qui è presente un bottone che serve per riportarsi alla schermata iniziale di questa sezione.

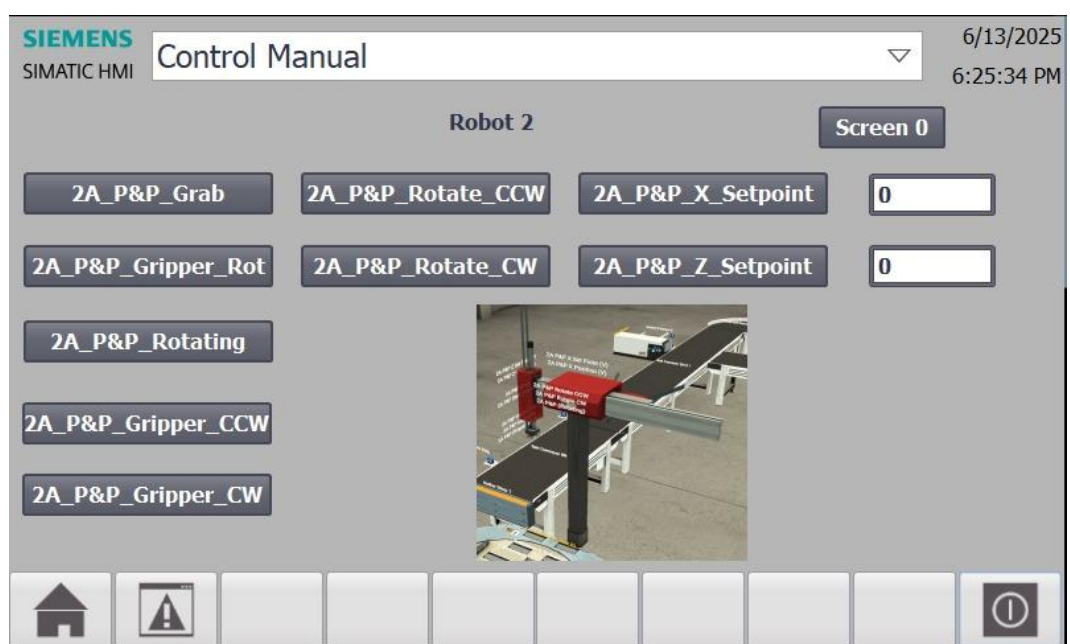


Figura 53: Screen del Control manual del Robot 2

Infine, l'ultima sezione riguarda il "Control de la maquina". Quest'unica schermata è caratterizzata da bottoni di gestione, che possono essere "Automatico", "Manual" e così via. Questi bottoni sono progettati utilizzando l'impostazione invertir Bit, la quale impone, non appena premuto, l'attivazione del comando di riferimento. Con il pulsante "Automatico" il sistema partirà in maniera automatica, invece, con manuale sarà possibile utilizzare la sezione di "control Manual" per far funzionare il sistema.

Nella schermata vi sono altri bottoni che verranno trattati nella parte dell'emergenza.

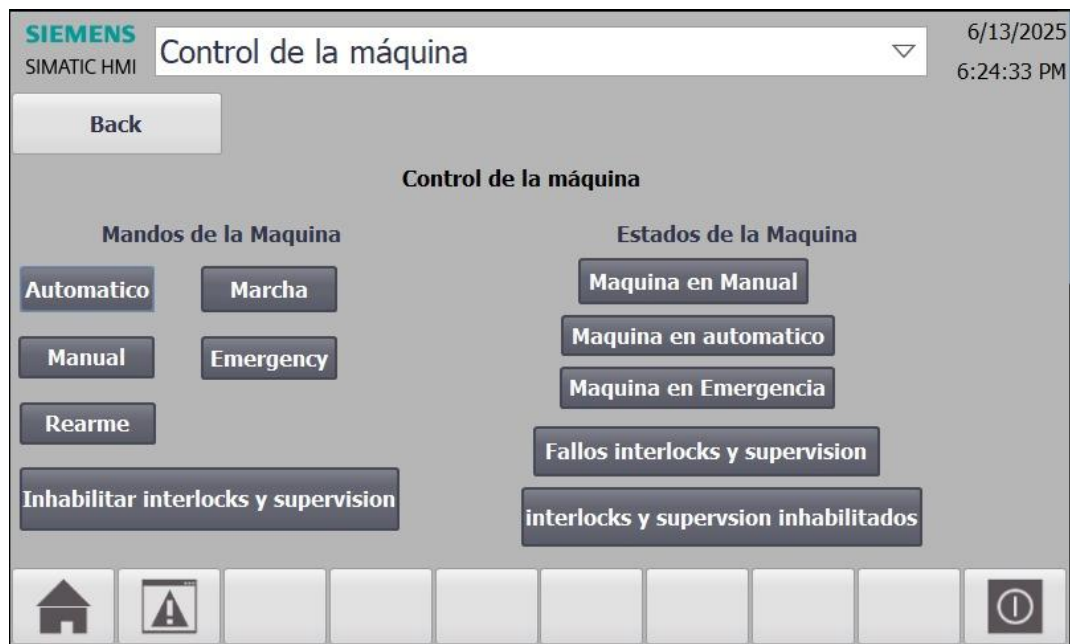


Figura 54: Screen del control de la maquina

Per completare la parte del controllo manuale, si è dovuto realizzare un codice SFC in modo tale che vi sia un collegamento pratico tra il bottone premuto e il codice di simulazione.

Tale codice è strutturato da due rami divergenti, basati sul fatto che si preme uno dei due bottoni principali, ovvero "Automatico" e "Manual". La parte relativa del bottone manuale è abbastanza semplice, caratterizzata semplicemente da temporizzatori e dalle posizioni iniziali delle macchine speciali del sistema. Il ramo di destra, invece, è dedicato al collegamento della variabile con il comando manuale corrispondente.

Per questo motivo, è fondamentale inserire tutti gli attuatori del sistema, in modo da garantirne il controllo completo nella modalità manuale.

Nel codice sono presenti anche funzioni di tipo "Call", che servono per richiamare un blocco FC. Si tratta di un modo per organizzare il codice richiamando sezioni separate. Questo significa che tutte le logiche contenute in quei blocchi funzionali vengono eseguite in contemporanea. Un altro comando presente è il "False", che

viene utilizzato per disattivare un comando, assegnando alle variabili di riferimento un valore booleano falso, cioè pari a 0.

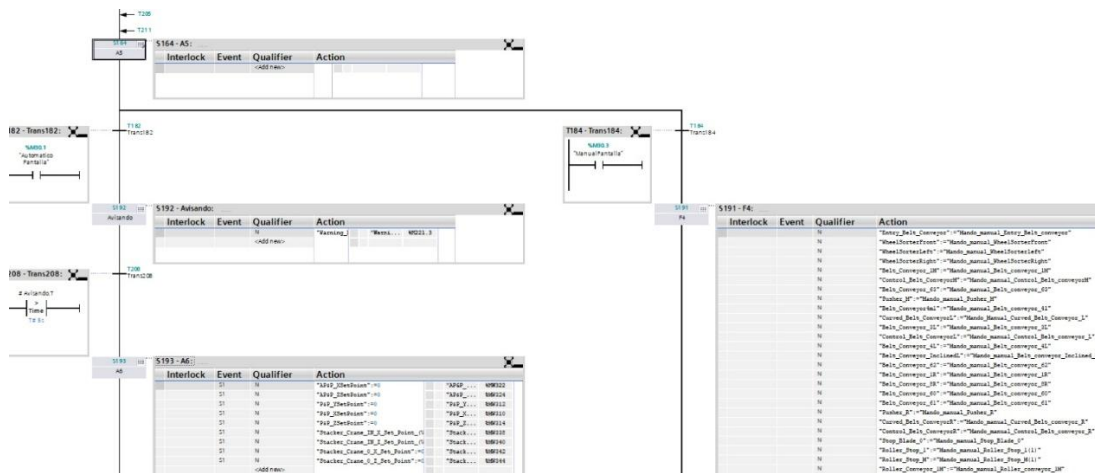


Figura 55: SFC per il controllo manuale parte1



Figura 56: SFC per il controllo manuale parte 2

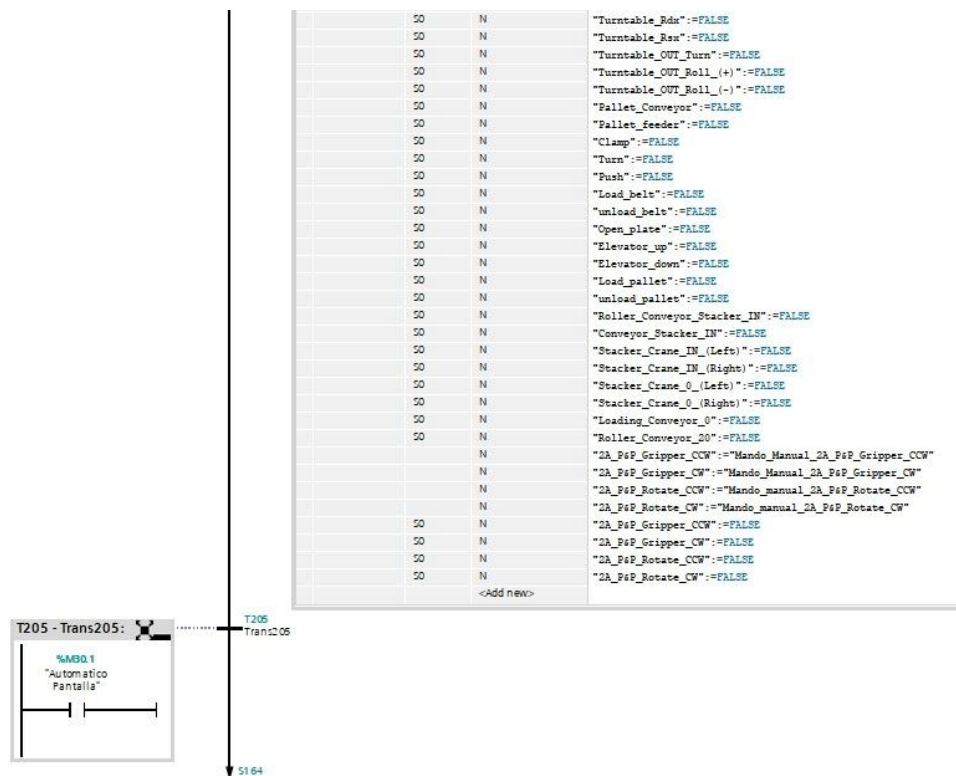


Figura 57: SFC per il controllo manuale parte 3

Si rappresentano i codici collegati alla funzione “Call” scritti in linguaggio SCL.

```

1 IF "Mando_manual_P&P_X_Set_Point" THEN
2   "P&P XSetPoint" := "Mando manual P&P X Set Point(1)";
3 END_IF;
4 IF "Mando_manual_P&P_Y_Set_Point" THEN
5   "P&P YSetPoint" := "Mando manual P&P Y Set Point B";
6 END_IF;
7 IF "Mando_manual_P&P_Z_Set_Point" THEN
8   "P&P ZSetPoint" := "Mando manual P&P Z Set Point B";
9 END_IF;

```

Figura 58: Codice SCL per il comando manuale del robot 1

```

1 IF "Mando_manual_2A_P&P_X_Set_Point" THEN
2   "AP&P XSetPoint" := "Mando manual 2A P&P X Set Point B";
3 END_IF;
4 IF "Mando_manual_2A_P&P_Z_Set_Point" THEN
5   "AP&P ZSetPoint" := "Mando manual 2A P&P Z Set Point B";
6 END_IF;

```

Figura 59: Codice SCL per il comando manuale del robot 2

```

1 IF "Mando_manual_Stacker_Crane_IN_X_Set_point" THEN
2   "Stacker Crane IN X Set Point (V)" := "Mando manual Stacker Crane IN X Set point B";
3 END_IF;
4 IF "Mando_manual_Stacker_Crane_IN_Z_Set_point" THEN
5   "Stacker Crane IN Z Set Point (V)" := "Mando manual Stacker Crane IN Z Set point B";
6 END_IF;

```

Figura 60: Codice SCL per il comando manuale dell'alamcen di ingresso

Queste sequenze di codice servono per assegnare manualmente i valori di Setpoint sugli assi X, Y e Z, sia per gli Almacens e che per i Robot. Nel dettaglio, il primo ciclo IF gestisce il Setpoint lungo X, il secondo lungo Y e l'ultimo lungo Z.

Capitolo 5

Comando d'emergenza e Rearme

Nelle linee di produzione, soprattutto quelle più grandi che potrebbero presentare più errori e malfunzionamenti, è fondamentale avere la possibilità di inserire sull'HMI un pulsante di stop e di riavvio del sistema.

Come si può osservare dalla struttura generale dell'impianto, all'inizio della linea di trasporto è presente un pannello di controllo con molteplici pulsanti. Alcuni di questi sono: il pulsante per l'emergenza ed il pulsante per il riavvio. La presenza di questo pannello è obbligatoria secondo le norme di sicurezza, come la EN ISO 13850 e la EN 60204-1, ed è fondamentale poiché permette all'operatore di intervenire in modo diretto ed immediato con la linea di trasporto. Esso, inoltre, permette di mantenere al minimo la funzionalità anche in caso di guasti al sistema digitale. La progettazione del pannello fisico e la disposizione dei pulsanti devono rispettare alcuni criteri di chiarezza, affinché l'operatore non svolga errori di interpretazione o lettura.

Tuttavia, nonostante la presenza del pannello fisico, è importante avere tali comandi anche sull'HMI, così da intervenire rapidamente in situazioni critiche, senza doversi recare al pannello di controllo e perdere tempo ad individuare il bottone corretto. L'HMI, in questo caso, consente di fermare il sistema (nel caso si preme il pulsante di emergenza) non solo in modo rapido, ma permette anche la supervisione del sistema a 360 gradi. Questo consente di essere più consapevoli, dato che l'interfaccia fornisce feedback delle varie zone del sistema.

Per la progettazione di questi due pulsanti bisogna intervenire sul Main e collegare opportunamente gli ingressi, forniti dalle variabili, con i terminali di riferimento.

Dall'immagine sottostante si può notare come il Main_Block, è caratterizzato da 4 ingressi principali: ACK_EF, ACK_S, EN_IL ed EN_SV.

- L'ACK_EF è un segnale di conferma degli errori di fine corsa, esso serve per resettare gli errori generati da condizioni anomale rilevate sui sensori di finecorsa.
- L'ACK_S, a differenza dell'ACK_EF, fa riferimento ad una sezione di reset per confermare un messaggio di fine sequenza.
- L'EN_IL è un segnale di abilitazione della modalità "IL" (Interlock). Quando è attivo, il sistema ad eseguire certe operazioni soggette a condizioni di sicurezza.
- L'EN_SV è un segnale di abilitazione della modalità "SV", che si riferisce alla supervisione del sistema.

I primi due ingressi saranno relativi al comando di Rearme, o riavvio, gli ultimi due, invece, sono predisposti per il funzionamento dell'emergenza in quanto collegati ad elementi particolari come gli interlock e la supervisione.

Si riporta unicamente l'immagine del codice "Nastro", in quanto, quello relativo agli Almacenes presenta lo stesso principio di funzionamento.

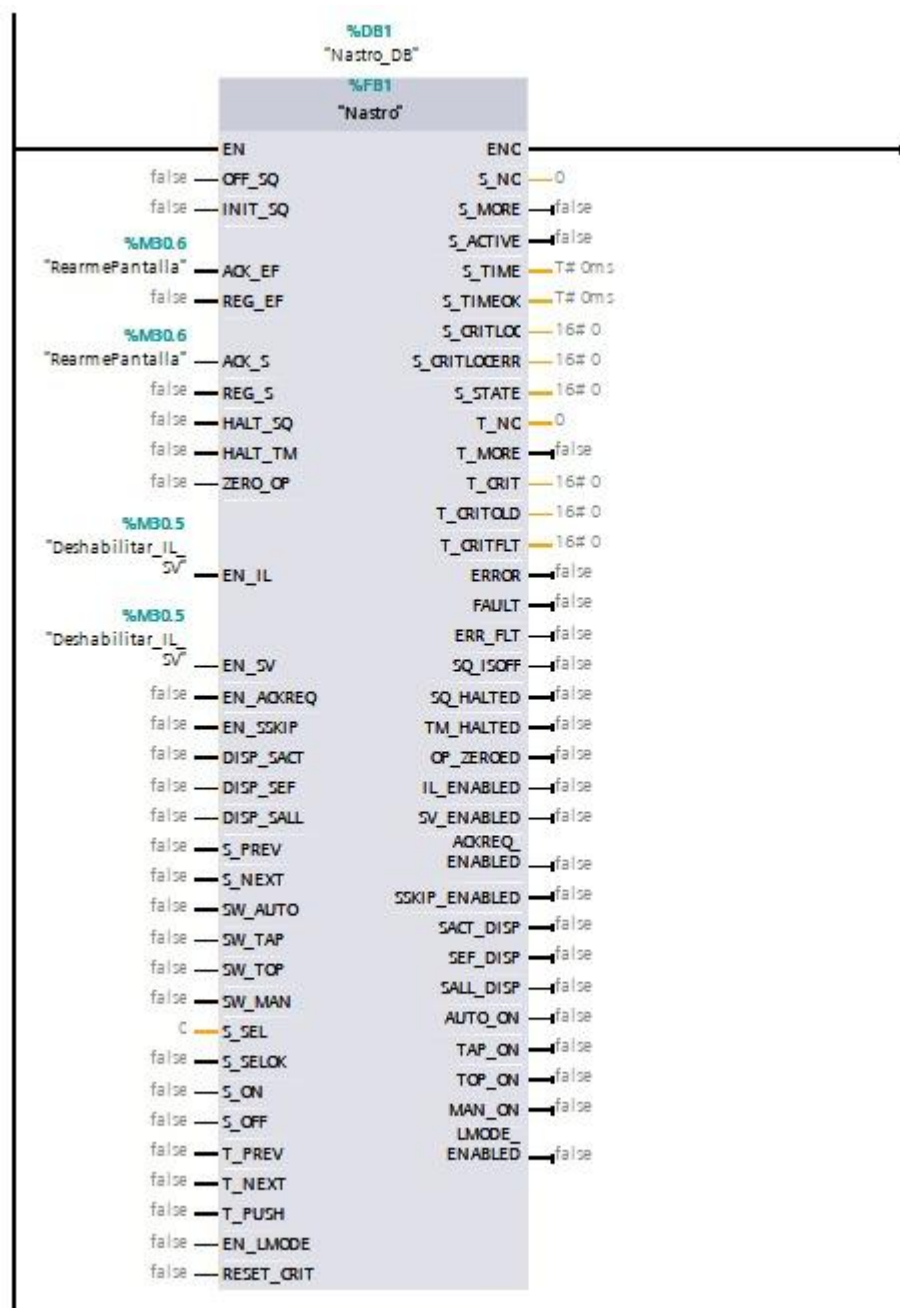


Figura 61: Main degli Almacenes

Gli Interlock agiscono bloccando l'avvio di un'operazione quando non sono rispettate determinate condizioni. Quindi l'attivazione del tasto di emergenza permette di far passare l'interlock da un segnale 1 ad uno di 0 così da fermare la macchina o il sistema di riferimento. Questo significa che, in caso di emergenza,

l'interlock impedisce alla macchina di ripartire fino a quando l'operatore non ne conferma il ripristino.

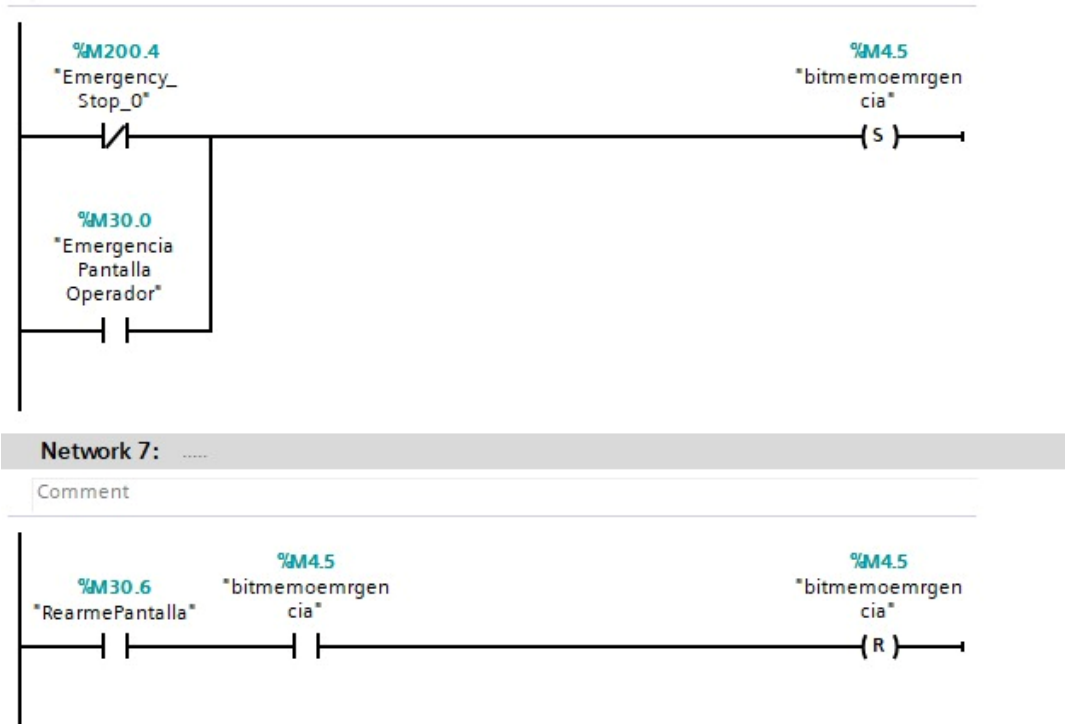


Figura 62: Main per la progettazine del comando d'emergenza

Gli interlock dovranno essere inseriti in ogni azione che comporta il movimento di una macchina e sarà progettato in linguaggio SCL. Come mostra l'immagine sottostante, di una macchina qualunque del sistema, esso è caratterizzato da un contatto negato del “Bitmemoemergencia”, una global memory creata per collegare il pulsante di emergenza con il codice di riferimento.

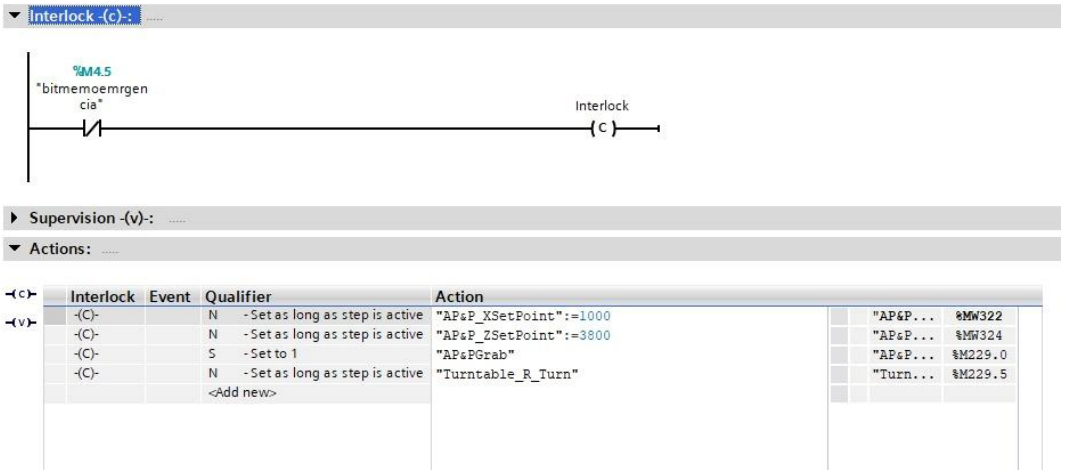


Figura 63: Interlock

Un altro elemento cruciale per questa sezione è la supervisione, che rappresenta una sorta di livello superiore di controllo. Tale funzione si basa sull'utilizzo di un temporizzatore di tipo U, il quale serve a misurare il tempo in modo cumulativo. In questo caso, è impiegato per disabilitare l'attività di una macchina nel caso in cui

l'operazione di movimento in corso superi il tempo massimo impostato, a causa di un'anomalia o malfunzionamento del sistema.

È importante inserire la supervisione unicamente in quelle operazioni in cui è possibile contare il tempo di esercizio, come la movimentazione lungo gli assi dei robot o degli almacén, oppure il tempo di rotazione ed indirizzamento dei vari Turntable.

Per finalizzare la supervisione, è inoltre importante inserire nell'interlock il contatto negato della fase di riferimento con il suffisso .VA, così da indicarne il valore attuale. Quindi, tale variabile si attiva quando la fase è attiva.

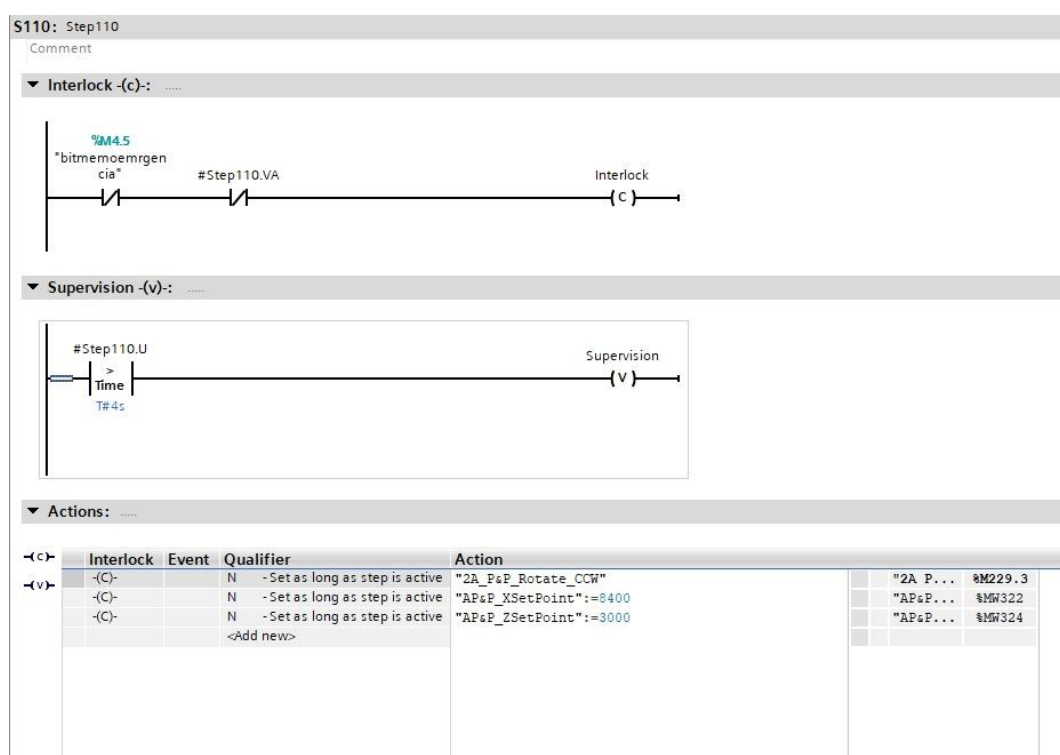


Figura 64: Interlock e Supervisione

Bisogna tenere presente che si sta facendo riferimento ad una simulazione. Questo implica che non ci sarà mai un effettivo malfunzionamento di una macchina o del sistema, poiché, al massimo può trattarsi di un problema di codice o di pesantezza del sistema per i software. Proprio per questo motivo, nella sezione "Main" sono state realizzate alcune righe di codice per ogni step di movimento, in linguaggio SCL, lungo gli assi delle cosiddette macchine speciali, così da arrestare l'esercizio di queste macchine, se in funzione, in una posizione ben precisa. Il linguaggio SCL, Structured Control Language, consiste nel linguaggio testuale di TIAPortal basato su istruzioni testuali come If, Else e end. Quindi, il comando dell'emergenza serve semplicemente per simulare il suo effettivo funzionamento.

Network 10: Robot 2

Comment

```
1 IF "Mastro_DB".Step109.L1 AND "APsP_XSetPoint" - "APsFXPosition" > 300 THEN
2   "Posicionrobotx" := "APsFXPosition";
3   "APsP_XSetPoint" := "Posicionrobotx" + 200;
4 END_IF;
5 IF "Mastro_DB".Step109.L1 AND "APsP_ZSetPoint" - "APsPZPosition" > 300 THEN
6   "Posicionrobotz" := "APsPZPosition";
7   "APsP_ZSetPoint" := "Posicionrobotz" + 200;
8 END_IF;
9 IF "Mastro_DB".Step110.L1 AND "APsP_XSetPoint" - "APsFXPosition" > 300 THEN
10  "Posicionrobotx" := "APsFXPosition";
11  "APsP_XSetPoint" := "Posicionrobotx" + 200;
12 END_IF;
13 IF "Mastro_DB".Step110.L1 AND "APsP_ZSetPoint" - "APsPZPosition" > 300 THEN
14  "Posicionrobotz" := "APsPZPosition";
15  "APsP_ZSetPoint" := "Posicionrobotz" + 200;
16 END_IF;
17 IF "Mastro_DB".Step196.L1 AND "APsP_XSetPoint" - "APsFXPosition" > 300 THEN
18  "Posicionrobotx" := "APsFXPosition";
19  "APsP_XSetPoint" := "Posicionrobotx" + 200;
20 END_IF;
21 IF "Mastro_DB".Step196.L1 AND "APsP_ZSetPoint" - "APsPZPosition" > 300 THEN
22  "Posicionrobotz" := "APsPZPosition";
23  "APsP_ZSetPoint" := "Posicionrobotz" + 200;
24 END_IF;
25
26 IF "Mastro_DB".Step122.L1 AND "APsP_ZSetPoint" - "APsPZPosition" > 300 THEN
27  "Posicionrobotz" := "APsPZPosition";
28  "APsP_ZSetPoint" := "Posicionrobotz" + 200;
29 END_IF;
30 IF "Mastro_DB".robot_riposos.L1 AND "APsP_XSetPoint" - "APsFXPosition" > 300 THEN
31  "Posicionrobotx" := "APsFXPosition";
32  "APsP_XSetPoint" := "Posicionrobotx" + 200;
33 END_IF;
34 IF "Mastro_DB".robot_riposos.L1 AND "APsP_ZSetPoint" - "APsPZPosition" > 300 THEN
35  "Posicionrobotz" := "APsPZPosition";
36  "APsP_ZSetPoint" := "Posicionrobotz" + 200;
37 END_IF;
```

Figura 65: Codice di emergenza per il Robot 2

Network 11: Almacen_IN

Comment

```
1 IF ("warehouse_DB".Step6.L1) AND ("Stacker_Crane_IN_X_Set_Point_(V)" - "Stacker_Crane_IN_X_Position_(V)") > 500 THEN
2   "PosicionAlmaX" := "Stacker_Crane_IN_X_Position_(V)";
3   "Stacker_Crane_IN_X_Set_Point_(V)" := "PosicionAlmaX" + 900;
4 END_IF;
5 IF ("warehouse_DB".Step6.L1) AND ("Stacker_Crane_IN_Z_Set_Point_(V)" - "Stacker_Crane_IN_Z_Position_(V)") > 500 THEN
6   "PosicionAlmaZ" := "Stacker_Crane_IN_Z_Position_(V)";
7   "Stacker_Crane_IN_Z_Set_Point_(V)" := "PosicionAlmaZ" + 900;
8 END_IF;
9 IF ("warehouse_DB".Step8.L1) AND ("Stacker_Crane_IN_Z_Set_Point_(V)" - "Stacker_Crane_IN_Z_Position_(V)") > 500 THEN
10  "PosicionAlmaZ" := "Stacker_Crane_IN_Z_Position_(V)";
11  "Stacker_Crane_IN_Z_Set_Point_(V)" := "PosicionAlmaZ" + 900;
12 END_IF;
13 IF ("warehouse_DB".Step11.L1) AND ("Stacker_Crane_IN_X_Set_Point_(V)" - "Stacker_Crane_IN_X_Position_(V)") > 500 THEN
14  "PosicionAlmaX" := "Stacker_Crane_IN_X_Position_(V)";
15  "Stacker_Crane_IN_X_Set_Point_(V)" := "PosicionAlmaX" + 900;
16 END_IF;
17 IF ("warehouse_DB".Step11.L1) AND ("Stacker_Crane_IN_Z_Set_Point_(V)" - "Stacker_Crane_IN_Z_Position_(V)") > 500 THEN
18  "PosicionAlmaZ" := "Stacker_Crane_IN_Z_Position_(V)";
19  "Stacker_Crane_IN_Z_Set_Point_(V)" := "PosicionAlmaZ" + 900;
20 END_IF;
21 IF ("warehouse_DB".Step13.L1) AND ("Stacker_Crane_IN_X_Set_Point_(V)" - "Stacker_Crane_IN_X_Position_(V)") > 500 THEN
22  "PosicionAlmaX" := "Stacker_Crane_IN_X_Position_(V)";
23  "Stacker_Crane_IN_X_Set_Point_(V)" := "PosicionAlmaX" + 900;
24 END_IF;
25 IF ("warehouse_DB".Step13.L1) AND ("Stacker_Crane_IN_Z_Set_Point_(V)" - "Stacker_Crane_IN_Z_Position_(V)") > 500 THEN
26  "PosicionAlmaZ" := "Stacker_Crane_IN_Z_Position_(V)";
27  "Stacker_Crane_IN_Z_Set_Point_(V)" := "PosicionAlmaZ" + 900;
28 END_IF;
```

Figura 66: Codice d'emergenza Almacen IN

Infine, per completare la sezione dell'emergenza, bisogna creare una riga di codice in un blocco di "StartUp" che permetta di tenere la funzione dell'emergenza sempre attiva in attesa che venga premuto il pulsante fisico o digitale.

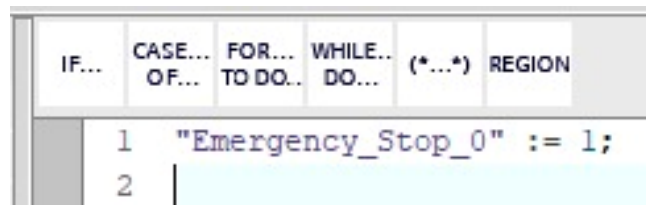


Figura 67: Startup per l'emergenza

Conclusioni e sviluppi futuri

Questo progetto di tesi ha come obiettivo quello di sviluppare una linea di trasporto pacchi che sia in grado di smistarli in base all'altezza, di trasportarli lungo i conveyors, di posizionarli opportunamente su dei pallet attraverso robot e palletizzatore, ed infine di effettuare lo stoccaggio di essi in maniera automatica mediante l'uso combinato di software come TiaPortal e Factoryio. Attraverso tale combinazione di Software, è possibile simulare il codice scritto su TiaPortal e di visualizzarlo su Factoryio. La possibilità di lavorare in un ambiente virtuale ha permesso di operare con maggiore facilità, prevenendo possibili errori e correggendoli rapidamente, poiché è possibile visualizzare direttamente il punto in cui si presentano. Dalla stesura del codice emerge l'importanza che hanno le varie condizioni e step in linguaggio SFC, SCL e Ladder, così come di temporizzatori, contatori ed il blocco chiamato "Data_Block", per l'automazione e la gestione della linea di produzione in esame. Oltre la parte di automazione, si è analizzata come poter realizzare la parte manuale, attraverso l'uso dell'HMI, un'interfaccia dotata di pulsanti virtuali adibiti alla gestione e movimentazione delle varie macchine o zone di lavoro.

Con la progettazione dell'HMI si è compreso quanto sia importante la comunicazione tra la parte automatica e quella manuale, quest'ultima utilizzata in caso di manutenzione o controlli specifici. Il sistema, inoltre, deve essere in grado di valutare possibili errori in maniera autonoma per questi sono stati inseriti interlock e supervisioni, oltre all'inserimento di pulsanti come l'emergenza ed il riavvio del sistema.

Oltre agli aspetti tecnici trattati, questo progetto evidenzia la complessità dello studio che sta dietro la progettazione di una linea di produzione: dalla scelta e configurazione dei Software, alla valutazione di spazi disponibili, fino all'uso di specifici dispositivi come sensori e Robot. In aggiunta, si mette in risalto l'importanza di interagire con il sistema sia in modalità automatica, sia manuale.

Tutte le considerazioni fatte rappresentano la base teorica e pratica per l'effettiva creazione di una linea di produzione. Nella realtà industriale, tuttavia, bisogna attenzionare altri fattori come l'analisi dei costi, la fattibilità della realizzazione e le varie norme di sicurezza.

Per quanto riguarda i possibili sviluppi futuri, il progetto potrebbe essere modificato e ampliato nei seguenti modi:

- Utilizzo di più PLC, così da rendere più complesso e completo, oltre a renderlo più veloce.
- Migliorare la comunicazione tra il PLC ed il disegno 3D, cambiando opportunamente la configurazione sul Factoryio.
- Aumentare la sicurezza attraverso l'aggiunta di moduli di sicurezza come i fail-safe.

- Implementare la tracciabilità dei pacchi tramite l'uso di codici a barre o RFID, così da associare a ogni unità informazioni specifiche sul contenuto, la destinazione e il tempo di transito.

Bibliografia

- [1] M. Hermann, T. Pentek e B. Otto, Design principles for industries 4.0 scenarios, International conference on system sciences (HICSS), 2016.
- [2] K. Umachandran, Industry 4.0: the new industrial revolution, IGI Global, 2019.
- [3] W. Bolton, «An Introduction to PLCs,» in *Programmable logic Controllers*, ISA-International Society of Automation, 2015.
- [4] M. O. A. Ephrem Ryan Alphonsus, «ScienceDirect,» 2016. [Online]. Available: <https://doi.org/10.1016/j.rser.2016.01.025>.
- [5] F. C. Vladimir Viktorov, Automazione dei sistemi meccanici, C.L.U.T, 2010.
- [6] A. H. Ivy Road, «Monk: Conveyors Sustems and Autonotation,» 2018. [Online]. Available: https://www.monk-conveyors.com/?utm_source=google&utm_medium=cpc&utm_campaign=2143866740.
- [7] Siemens, «Siemens,» [Online]. Available: <https://new.siemens.com/global/en/products/automation/systems/industrial-plc/s7-1200.html>.
- [8] N. Fondelli, «ok!Mugello,» [Online]. Available: https://www.okmugello.it/articolo/cose-un-temporizzatore-e-come-funziona_70154.
- [9] C. SECCHI, Sensori e trasduttori, Ingegneria e Tecnologie dei Sistemi di Controllo, 2005.