



Technical Report

Reporting CNR (full name and address): Industrial Technology and Automation Institute - (ITIA)
Vigevano Laboratory
Via Pisani, 1
I-27029 Vigevano (PV), Italy
Tel.: (+39 0381) 692652 Fax: (+39 0381) 693021

Responsible Person: Emanuele Carpanzano

Project Name:

Document Title: Simulatore della Linea Molecolare implementato in ISAGRAF PRO

Document Ref.No.: 2MaCS/TR-03/CS201 **Page:** 1

Date of issue: 14.10.2003 **Status:** Approved **No. of pages:** 31

Client(s): **Revision:** A

Author(s): A. Cataldo, (ITIA)

Distribution: ITIA-CNR

Keywords: Impianti Manifatturieri, Architettura dei sistemi di Controllo, Programmable Logic Control, Simulazione Eventi Discreti

Sommario:

L'impianto manifatturiero di Vigevano è caratterizzato da un'innovativa linea di trasporto molecolare basata sull'integrazione di sei terne le quali costituiscono un percorso continuo lungo il quale i vari semilavorati di scarpe si muovono [1]. Ciascuna di queste terne è costituita da tre dispositivi rotanti. Il primo è la "Tavola" che possiede 12 slots ed è utilizzata sia per la movimentazione dei semilavorati verso i vari gruppi di lavorazione che per il rientro delle forme verso il magazzino. Il secondo è la "Isola" che possiede 24 slots ed attorno alla quale sono dislocate le varie macchine operatrici. Infine il "Manipolatore" ha 3 slots e serve per spostare sulle varie Tavole e Isole i semilavorati e le forme.

Obiettivo di questo report è di descrivere la definizione del modello di simulazione ad eventi discreti della linea molecolare nonché la relativa implementazione software effettuata mediante ISAGRAF PRO. Inoltre tale obiettivo si propone di esplicitare il nome ed il significato delle variabili utilizzate per l'evoluzione del modello di simulazione. Tali variabili di fatto costituiscono l'interfaccia del modello di simulazione della linea molecolare con il mondo esterno, cioè con il controllo (sia esso automatico, sia esso implementato per mezzo di comandi impartiti dall'utente che utilizza tale modello di simulazione). Ciò è stato progettato volutamente in questo modo, per permettere una più facile migrazione delle strategie di controllo (testate con il simulatore) sul sistema di controllo reale della linea molecolare.

Scopo di questo report è quello di rendere disponibile una documentazione del simulatore ad eventi discreti della linea molecolare a chiunque debba fare uso di tale simulatore, per verificare, mediante simulazione, le strategie di controllo da applicare alla linea molecolare. Non ultimo, la descrizione della definizione e dell'implementazione del modello di simulazione permette di conoscere più in dettaglio il processo della linea di trasporto molecolare.

Sviluppi futuri di tale lavoro riguardano un'eventuale modellazione delle stazioni operatore disposte lungo la linea di movimentazione molecolare e relativa documentazione.

INDEX

1 GENERALITÀ.....	4
1.1. INTRODUZIONE.....	4
1.2. OBIETTIVO.....	4
1.3. SCOPO.....	4
1.4. STRUTTURA DEL DOCUMENTO.....	4
2. SPECIFICA FUNZIONALE DELLA LINEA MOLECOLARE NON CONTROLLATA.....	5
2.1. DESCRIZIONE GENERALE.....	5
2.2. ROTAZIONE DEI DISPOSITIVI ROTANTI.....	6
2.3. AZIONAMENTO DEGLI SPINTORI.....	7
3. DEFINIZIONE DEL MODELLO DI SIMULAZIONE DELLA LINEA MOLECOLARE.....	8
3.1. MODULI BASE.....	8
3.1.1. <i>Modulo Rotazione Dispositivo Rotante</i>	8
3.1.2. <i>Modulo Azionamento Spintore</i>	8
3.2. MODULI COMPOSTI.....	10
3.2.1. <i>Modulo Tavola</i>	10
3.2.2. <i>Modulo Isola</i>	11
3.2.2.1. Modulo Isola con 1 Spintore.....	12
3.2.2.2. Modulo Isola con 2 Spintori.....	12
3.2.3. <i>Modulo Manipolatore</i>	13
3.2.3.1. Modulo Manipolatore con 2 Spintori.....	13
3.2.3.2. Modulo Manipolatore con 3 Spintori.....	14
3.2.4. <i>Modulo Terna</i>	14
3.3. MODULO GESTIONE DELL'INFORMAZIONE ASSOCIATA AGLI SLOTS.....	16
4. IMPLEMENTAZIONE DEL SIMULATORE CON ISAGRAF PRO.....	18
4.1. SEQUENZE BASE.....	18
4.1.1. <i>Sequenza Rotazione Dispositivo Rotante</i>	18
4.1.2. <i>Sequenza Azionamento Spintore</i>	19
4.2. SEQUENZE COMPOSTE.....	19
4.2.1. <i>Sequenza della Tavola</i>	19
4.2.2. <i>Sequenza dell'Isola</i>	20
4.2.3. <i>Sequenza del Manipolatore</i>	21
4.2.4. <i>Sequenza Gestione dell'Informazione associata agli Slots</i>	21
4.3. PROGRAMMA DELLA TERNA.....	23
4.4. PROGRAMMA DEL SIMULATORE.....	23

5. VERIFICA DEL MODELLO DELLA LINEA MOLECOLARE ATTRAVERSO SIMULAZIONE.....	25
6. TABELLA RIASSUNTIVA SEGNALI DI INTERFACCIA SIMULATORE – SISTEMA DI CONTROLLO	26
7. CONCLUSIONI E LAVORI FUTURI.....	28
8. ACRONIMI E DEFINIZIONI.....	29
8.1. TERMINI UTILIZZATI IN CONTROLLI AUTOMATICI	29
8.2. ACRONIMI	30
9. RIFERIMENTI E LINKS	31
9.1. RIFERIMENTI	31
9.2. INTERNET LINKS	31

1 GENERALITÀ

1.1. INTRODUZIONE

L'impianto manifatturiero di Vigevano è caratterizzato da un'innovativa linea di trasporto molecolare basata sull'integrazione di sei terne le quali costituiscono un percorso continuo lungo il quale i vari semilavorati di scarpe si muovono [1]. Ciascuna di queste terne è costituita da tre dispositivi rotanti. Il primo è la "Tavola" che possiede 12 slots ed è utilizzata sia per la movimentazione dei semilavorati verso i vari gruppi di lavorazione che per il rientro delle forme verso il magazzino. Il secondo è la "Isola" che possiede 24 slots ed attorno alla quale sono dislocate le varie macchine operatrici. Infine il "Manipolatore" ha 3 slots e serve per spostare sulle varie Tavole e Isole i semilavorati e le forme.

1.2. OBIETTIVO

Obiettivo di questo report è di descrivere la definizione del modello di simulazione ad eventi discreti della linea molecolare nonché la relativa implementazione software effettuata mediante ISAGRAF PRO. Inoltre tale obiettivo si propone di esplicitare il nome ed il significato delle variabili utilizzate per l'evoluzione del modello di simulazione. Tali variabili di fatto costituiscono l'interfaccia del modello di simulazione della linea molecolare con il mondo esterno, cioè con il controllo (sia esso automatico, sia esso implementato per mezzo di comandi impartiti dall'utente che utilizza tale modello di simulazione). Ciò è stato progettato volutamente in questo modo, per permettere una più facile migrazione delle strategie di controllo (testate con il simulatore) sul sistema di controllo reale della linea molecolare.

1.3. SCOPO

Scopo di questo report è quello di rendere disponibile una documentazione del simulatore ad eventi discreti della linea molecolare a chiunque debba fare uso di tale simulatore, per verificare, mediante simulazione, le strategie di controllo da applicare alla linea molecolare. Non ultimo, la descrizione della definizione e dell'implementazione del modello di simulazione permette di conoscere più in dettaglio il processo della linea di trasporto molecolare.

1.4. STRUTTURA DEL DOCUMENTO

In questo documento viene presentata innanzitutto una panoramica del funzionamento della linea di trasporto molecolare, evidenziando i sensi di rotazione dei dispositivi rotanti e la convenzione utilizzata per la numerazione degli slots.

Successivamente viene illustrato il criterio che ha guidato alla concezione del modello di simulazione evidenziandone i concetti di base.

Quindi vengono spiegate le scelte adottate nell'implementazione del modello di simulazione, utilizzando l'ambiente di programmazione e sviluppo ISAGRAF PRO.

Viene inoltre verificata la corrispondenza tra i risultati di simulazione del modello e le specifiche funzionali della linea molecolare. Tale simulazione è stata condotta sempre in ambiente ISAGRAF PRO.

A conclusione del lavoro svolto viene illustrata la tabella delle variabili che costituiscono l'interfaccia tra il modello del simulatore ed il sistema di controllo della linea molecolare.

2. SPECIFICA FUNZIONALE DELLA LINEA MOLECOLARE NON CONTROLLATA

2.1. DESCRIZIONE GENERALE

La linea di trasporto molecolare è costituita da sei terne modulari adiacenti come mostrato in Figura 1 al fine di consentire il trasporto, da una cella all'altra, dei semilavorati e delle forme sopra le quali vengono costruite le scarpe.

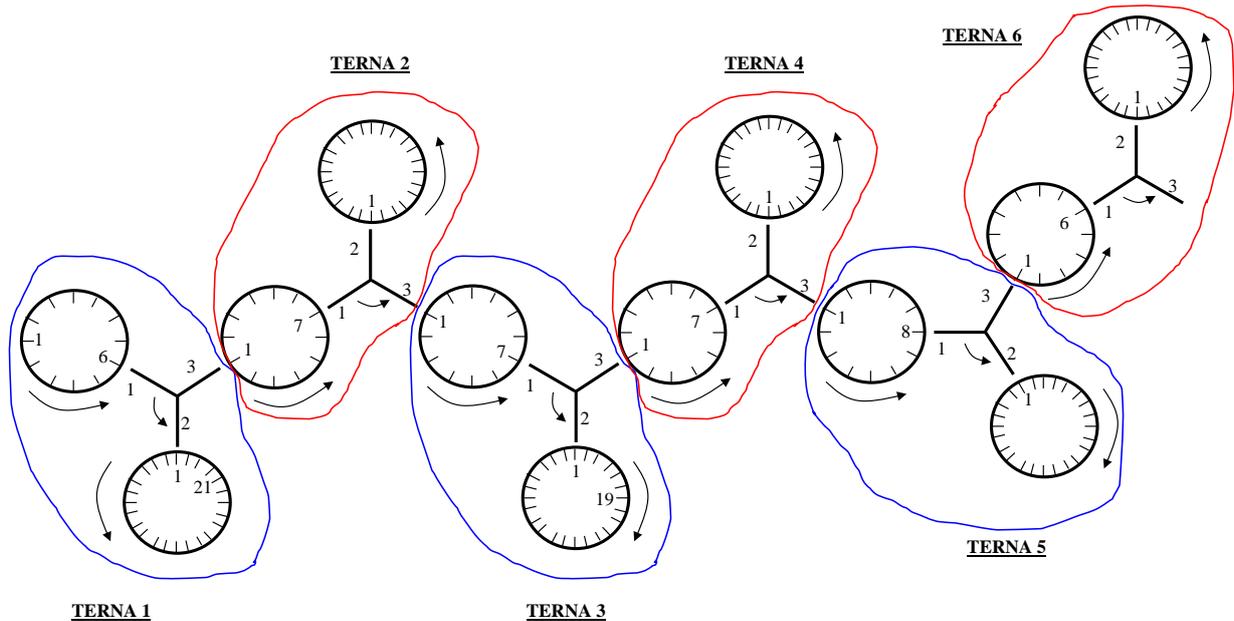


Figura 1

Ciascuna terna è a sua volta costituita da tre dispositivi rotanti chiamati Tavola, Isola e Manipolatore.

La Tavola possiede 12 slots ed è utilizzata sia per la movimentazione dei semilavorati verso i vari gruppi di lavorazione, sia per il rientro delle forme verso il magazzino.

L'Isola possiede 24 slots ed attorno ad essa sono dislocate le varie macchine operatrici.

Il Manipolatore possiede 3 slots e serve per spostare sulle varie Tavole e Isole i semilavorati e le forme.

Al fine di identificare le posizioni dei vari slots è stata stabilita la seguente convenzione:

- Lo slot della Tavola che si trova allineato con il braccio del manipolatore della terna precedente è nella posizione 1; gli altri slots sono in posizioni progressive seguendo il senso anti-orario.
- Lo slot del Manipolatore che si trova allineato con la Tavola della stessa terna si trova in posizione 1; gli altri slots sono in posizioni progressive seguendo il senso anti-orario.
- Lo slot dell'Isola che si trova allineato con il braccio del manipolatore della stessa terna è nella posizione 1; gli altri slots sono in posizioni progressive seguendo il senso anti-orario.

La disposizione meccanica delle varie terne pone subito il progettista del sistema di controllo della linea molecolare di fronte a delle scelte. In particolare si nota che il manipolatore delle terne 1, 3, 5, ruotando in senso anti-orario, sposta un generico slot sequenzialmente nelle posizioni 1-

2-3-1-2-3 ... cioè prima è allineato con la Tavola della stessa terna (1), poi con l'Isola della stessa terna (2), quindi con la Tavola della terna successiva (3). Nelle terne 2, 4, 6 avviene invece che il generico slot occupa in sequenza le posizioni 1-3-2-1-3-2 ... cioè prima è allineato con la Tavola della stessa terna (1), poi si allinea con la Tavola della terna successiva (3) quindi con l'Isola della stessa terna (2).

Fermo restando la convenzione di numerazione delle posizioni degli slots, le alternative per procedere con il controllo della linea molecolare sono:

- Far ruotare i manipolatori delle terne 2, 4, 6 in senso orario, invertendo la marcia dei motori elettrici.
- Tener conto del particolare sopra esposto nella gestione delle informazioni associate agli slots.

Si è optato per la seconda soluzione, quindi il modulo che dovrà tener traccia del contenuto di informazione di ciascuno slot dovrà gestire questi differenti comportamenti tra i diversi manipolatori.

Va sottolineato il fatto che dal punto di vista del comportamento meccanico non vi è alcuna differenza tra i vari Manipolatori perché i comandi di rotazione ed i segnali di retroazione sono identici per ogni Manipolatore, indipendentemente dal senso di rotazione del dispositivo. La differenza è localizzata solamente nella gestione delle informazioni degli slots.

Un'altra particolarità meccanica della linea di trasporto molecolare consiste nel fatto che le Isole 3 e 5 ruotano in senso orario, a differenza di tutte le altre. Questa forzatura deriva dalla sequenza delle operazioni da eseguire sui semilavorati. In particolare il layout delle macchine operatrici disposte attorno alle Isole 3 e 5 è tale per cui le operazioni da eseguire seguono in sequenza un senso orario attorno all'Isola, da cui la rotazione opposta rispetto le altre Isole.

Anche per le Isole non vi è alcuna differenza dal punto di vista del comportamento meccanico perché i comandi di rotazione ed i segnali di retroazione sono identici per ogni Isola, indipendentemente dal senso di rotazione del dispositivo. La differenza è localizzata solamente nella gestione delle informazioni degli slots.

2.2. ROTAZIONE DEI DISPOSITIVI ROTANTI

Tutti i dispositivi rotanti (Tavole, Isole e Manipolatori) sono dotati di meccanismi per la rotazione del tutto simili tra loro. Essi sono composti da un azionamento elettrico che imprime il moto di rotazione e di un meccanismo globoidale che centra con precisione la posizione di arresto del dispositivo rotante. L'azionamento elettrico è dotato di un minimo di automazione di bordo che interagisce con il sistema di controllo dell'impianto.

Per far ruotare il generico dispositivo è necessario settare a 1 un segnale digitale elettrico. In particolare l'azionamento legge lo stato alto di tale segnale in ingresso ed imprime al dispositivo rotante una rotazione equivalente ad uno slot. Contemporaneamente l'automazione locale del dispositivo (integrata nel processo) abbassa il segnale digitale di uscita di fine rotazione (tale segnale digitale di uscita di fine rotazione è alto quando il dispositivo rotante è fermo).

In base al dispositivo che ruota, l'angolo di rotazione tra una posizione di arresto e l'altra varia. Per la Tavola (12 slots) ogni rotazione di uno slot equivale a 30°. Per l'Isola abbiamo una rotazione di 15° (24 slots), mentre per il Manipolatore una rotazione di 120° (3 slots).

Non appena la rotazione di uno slot è conclusa, circa 1500ms dopo lo start, l'automazione di bordo del dispositivo alza nuovamente il segnale digitale in uscita di fine rotazione. Tale segnale rimane alto fino a che non riprende un'altra rotazione.

Se il segnale di comando della rotazione viene mantenuto alto dal sistema di controllo, l'azionamento, dopo la rotazione di uno slot ed il relativo arresto, effettua una nuova rotazione, dopo una pausa di circa 500ms.

Il cronogramma che illustra quanto detto sopra è mostrato in Figura 2.

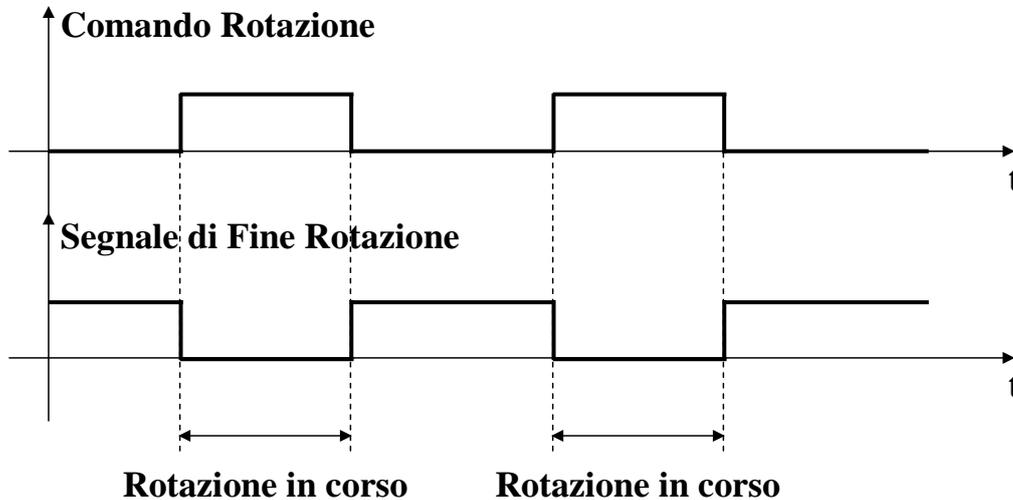


Figura 2

2.3. AZIONAMENTO DEGLI SPINTORI

Il meccanismo del generico spintore è realizzato per mezzo di un cilindro e pistone ad aria compressa e di un'elettrovalvola che comanda il flusso dell'aria nel cilindro.

L'elettrovalvola è di tipo monostabile, pertanto quando viene alzato il segnale elettrico digitale di comando essa apre il passaggio dell'aria in coda al cilindro spingendo il pistone verso l'estremità opposta del cilindro stesso cioè verso il fine-corsa. In tal modo il pistone, attraverso lo stelo ad esso collegato, esercita una spinta assiale sulle forme da spostare.

Il tempo necessario al pistone per completare l'intera corsa è di circa 500ms.

Non appena il segnale elettrico di comando elettrovalvola si abbassa, essa torna nello stato di riposo aprendo il passaggio dell'aria nella testa del cilindro, spingendo il pistone indietro verso l'inizio-corsa.

Se accade che durante l'uscita del pistone il segnale di comando viene abbassato, il pistone inverte il movimento e ritorna nella posizione di riposo iniziale. Ciò è evidenziato in quanto la modellazione ad eventi discreti dell'azionamento degli spintori trascura questo possibile comportamento fisico a vantaggio di una maggiore semplificazione del modello matematico che descrive il processo. Nel paragrafo relativo alla modellazione di tali azionamenti verrà ulteriormente dettagliato questo aspetto.

Al fine di rendere nota al sistema di controllo la posizione del pistone, indipendentemente dal segnale di comando fornito all'elettrovalvola, due sensori magnetici sono applicati agli estremi del cilindro in modo da indicare l'uno l'inizio-corsa (pistone dentro), l'altro il fine-corsa (pistone fuori).

3. DEFINIZIONE DEL MODELLO DI SIMULAZIONE DELLA LINEA MOLECOLARE

3.1. MODULI BASE

Nella fase di definizione del modello di simulazione della linea molecolare si è cercato di utilizzare alcuni concetti della programmazione ad oggetti al fine di ottenere una progettazione efficace ed efficiente del simulatore, pur sapendo di non utilizzare un linguaggio di programmazione ad oggetti, quale l'SFC. In particolare sono stati utilizzati come criteri la modularità e l'incapsulamento delle informazioni.

Il concetto di modularità è di per sé insita nel processo fisico da modellare. Infatti i componenti fisici che interagiscono con il sistema di controllo, e che sono quindi da modellare per simularne il comportamento, sono gli spintori e gli azionamenti dei dispositivi rotanti. E' risultato naturale quindi progettare il modello del simulatore mediante moduli, cominciando da quelli base (spintori e azionamenti di rotazione) per arrivare a quelli composti (Tavola, Isola e Manipolatore).

Il criterio dell'incapsulamento è utile per poter stabilire in modo chiaro le interfacce dei vari moduli definendole del tutto simili a quelle che presentano i vari dispositivi reali. Inoltre tale concetto consente di circoscrivere le funzionalità dei vari moduli software, permettendo anche il lavoro in parallelo di diversi progettisti.

3.1.1. Modulo Rotazione Dispositivo Rotante

La funzionalità della rotazione dei dispositivi rotanti può essere inglobata in un modulo base denominato RDR (Rotazione Dispositivo Rotante), vedi Figura 3.

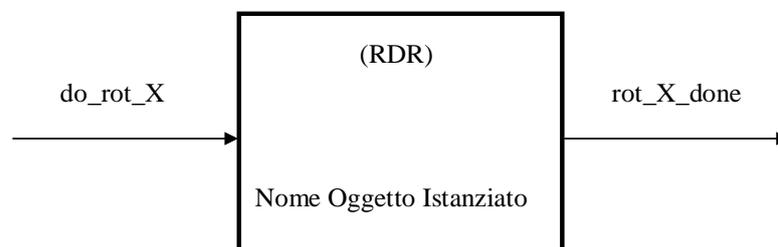


Figura 3

Esso presenta un segnale d'ingresso digitale `do_Rot_X` che rappresenta il comando di rotazione proveniente dal sistema di controllo d'impianto.

Inoltre presenta un segnale d'uscita digitale `rot_X_done` il quale, quando a 1, segnala che il dispositivo è fermo.

La lettera X inserita nella nomenclatura dei segnali di ingresso e uscita può assumere i valori T, I o M relativamente al dispositivo Tavola, Isola o Manipolatore a cui è riferito.

Il comportamento del segnale d'uscita, relativamente a quello di ingresso, è lo stesso descritto nelle funzionalità dei dispositivi rotanti.

3.1.2. Modulo Azionamento Spintore

Lo spintore può essere modellato per mezzo di un modulo chiamato AS (Azionamento Spintore), vedi Figura 4.

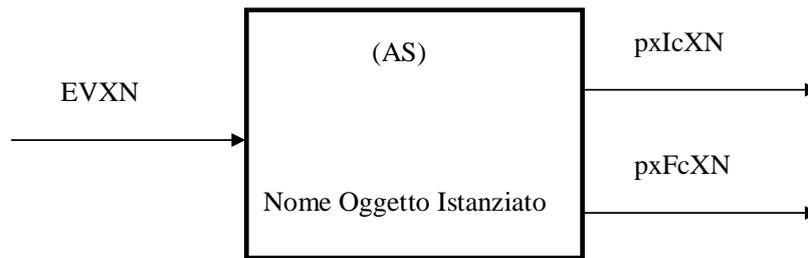


Figura 4

Esso presenta un segnale d'ingresso EVXN relativo al comando dato dal sistema di controllo d'impianto all'elettrovalvola.

In prima battuta, per semplificare il modello del simulatore rispetto al comportamento reale dello spintore, non si è tenuto conto dell'eventuale cambio di valore del segnale di ingresso all'elettrovalvola prima che il pistone abbia completato la sua corsa. Ciò è abbastanza sensato in quanto un controllo ben progettato prima fornisce l'azione di controllo relativa all'uscita del pistone e solo successivamente al raggiungimento del fine-corsa viene negato tale comando, così da richiamare il pistone indietro. Tale approssimazione permette di prefissare il tempo necessario al pistone per completare la corsa. Tuttavia con un modello così semplificato non si è in grado di scoprire un eventuale baco del programma di controllo relativo a questa funzione, cosa invece fattibile utilizzando un dimostratore fisico che riproduce in scala il processo reale. Tale dimostratore infatti non può prescindere dalla fisica dei componenti.

Il blocco AS presenta inoltre due segnali di uscita che rappresentano le posizioni di inizio-corsa e fine-corsa del pistone, pXlcXN (proximity inizio-corsa) e pXFcXN (proximity fine-corsa).

Le funzionalità di questo blocco rispecchiano quelle del funzionamento dello spintore reale. Il pistone in posizione nominale si trova in corrispondenza dell'inizio-corsa il cui segnale pXlcXN sarà a 1. Non appena viene alzato a 1 il comando dell'elettrovalvola, viene resettato il segnale di inizio-corsa e, dopo 500ms, viene settato a 1 il segnale di fine-corsa pXFcXN.

Riportato a 0 il segnale di comando dell'elettrovalvola, il segnale di fine-corsa viene resettato e, dopo 500ms, viene settato a 1 quello di inizio-corsa, vedi Figura 5.

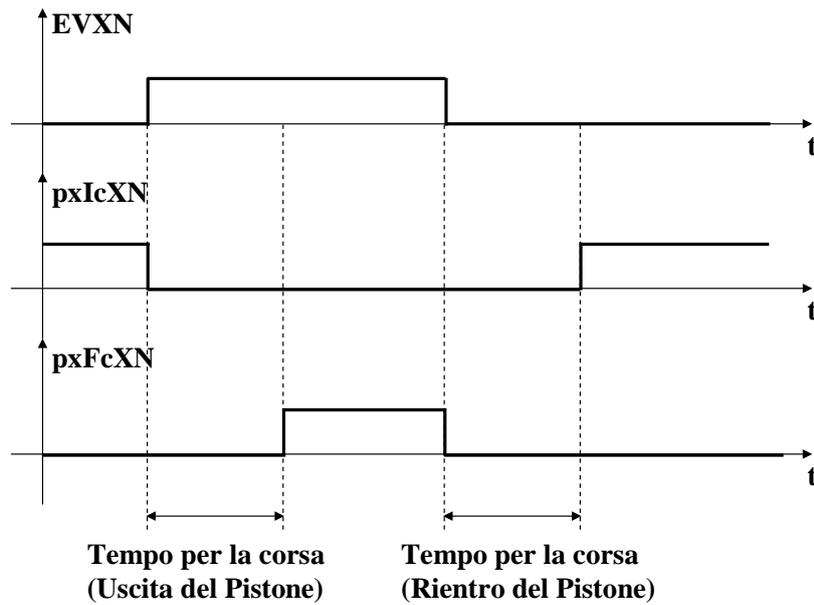


Figura 5

Le lettere X ed N inserite nella nomenclatura dei segnali di ingresso e uscita possono assumere rispettivamente i valori T, I o M in riferimento al dispositivo Tavola, Isola o Manipolatore e 1, 2 o 3 in riferimento allo spintore 1, 2 o 3 associato.

3.2. MODULI COMPOSTI

Tali moduli utilizzano sostanzialmente i moduli di base per raggruppare le funzionalità di rotazione dei dispositivi rotanti e le funzionalità degli spintori. In tal modo si può associare un modulo composto ai dispositivi elementari che compongono la linea molecolare: la Tavola, l'Isola ed il Manipolatore.

3.2.1. Modulo Tavola

Il modulo tavola che deve presentare le funzionalità di rotazione e le funzionalità relative a due spintori è composto dai moduli base RDR e AS, vedi Figura 6.

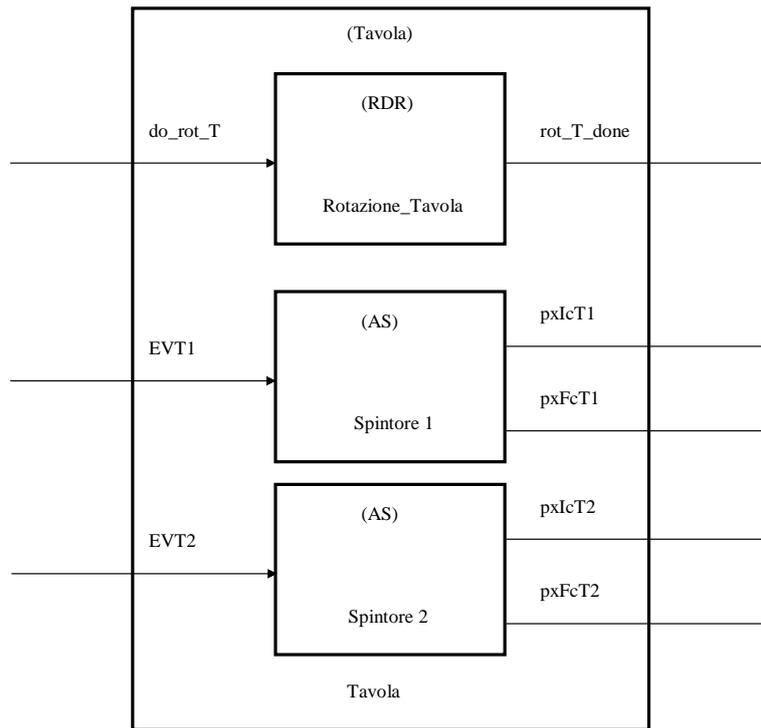


Figura 6

I segnali di ingresso del modulo composto Tavola consistono nel comando di rotazione della Tavola `do_rot_T` e nei due segnali che comandano le elettrovalvole associate ai due spintori `EVT1` e `EVT2` (dove 1 è riferito allo spintore 1 e 2 allo spintore 2 di ogni Tavola, vedi Figura 1).

I segnali di uscita del modulo consistono nel feedback di rotazione avvenuta `rot_T_done` e nei quattro segnali dei proximity switches posti sui due spintori (`pxIcT1` e `pxFcT1` per lo spintore 1, `pxIcT2` e `pxFcT2` per lo spintore 2).

3.2.2. Modulo Isola

Il modulo composto Isola, come per il modulo composto Tavola, presenta le funzionalità di rotazione e quelle associate agli spintori. Le isole 2, 4, 5 e 6 hanno un solo spintore, mentre le Isole 1 e 3 ne hanno due.

Ciò implica il fatto di avere due tipi di moduli composti per l'Isola: un modulo con un solo spintore ed un modulo con due (il quale è del tutto simile al modulo composto Tavola).

3.2.2.1. Modulo Isola con 1 Spintore

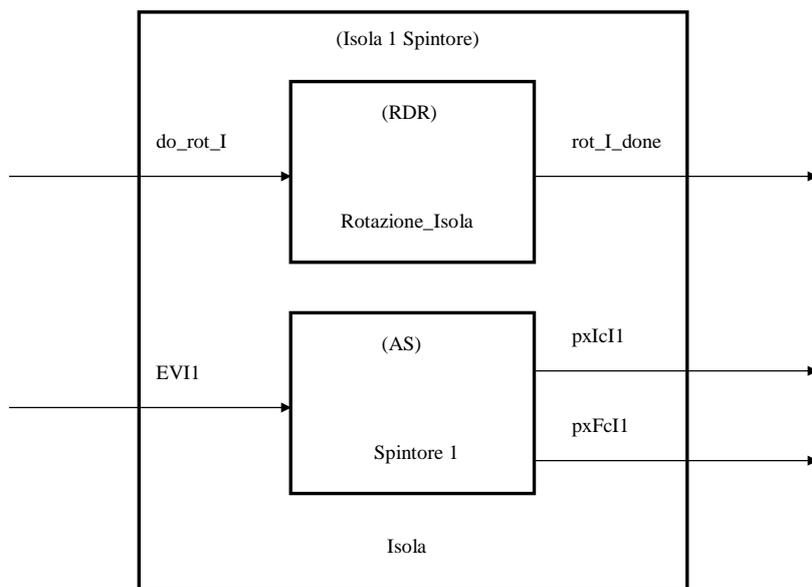


Figura 7

I segnali di ingresso del modulo composto Isola 1 Spintore consistono nel comando di rotazione dell'Isola `do_rot_I` e nel segnale che comanda l'elettrovalvola associata allo spintore `EVI1` (dove 1 è riferito allo spintore 1 di ogni Isola).

I segnali di uscita del modulo consistono nel feedback di rotazione avvenuta `rot_I_done` e nei due segnali dei proximity switches posti sullo spintore (`pxIc11` e `pxFc11` per lo spintore 1).

3.2.2.2. Modulo Isola con 2 Spintori

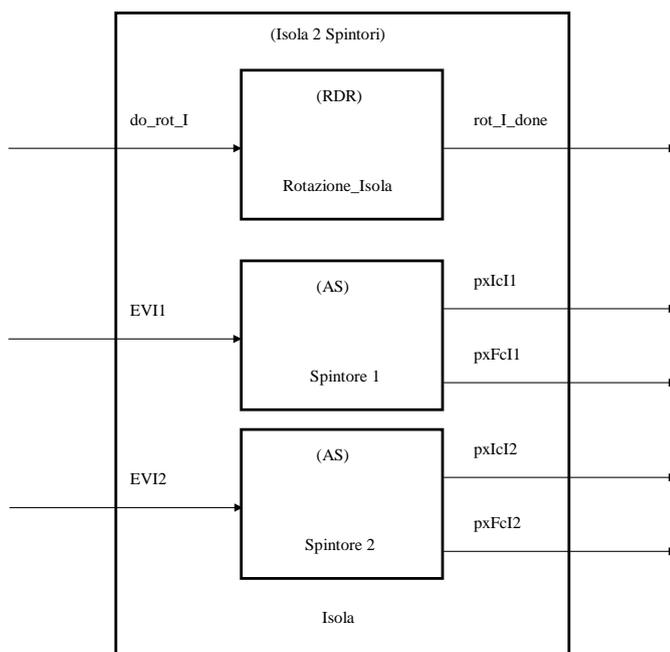


Figura 8

I segnali di ingresso del modulo composto Isola 2 Spintori consistono nel comando di rotazione dell'Isola `do_rot_I` e nei due segnali che comandano le elettrovalvole associate ai due spintori `EVI1` e `EVI2` (dove 1 è riferito allo spintore 1 di ogni Isola e 2 allo spintore 2 delle Isole che ne dispongono, cioè la 1 e la 3).

I segnali di uscita del modulo consistono nel feedback di rotazione avvenuta `rot_I_done` e nei quattro segnali dei proximity switches posti sui due spintori (`pxlcl1` e `pxFcl1` per lo spintore 1, `pxlcl2` e `pxFcl2` per lo spintore 2).

3.2.3. Modulo Manipolatore

Il modulo Manipolatore possiede anch'esso la funzionalità di rotazione e le funzionalità associate agli spintori. Tutti i Manipolatori hanno tre spintori, tranne il Manipolatore della terna 6 che ne ha soltanto due.

Come per il modulo composto Isola si hanno due tipi di moduli Manipolatore.

3.2.3.1. Modulo Manipolatore con 2 Spintori

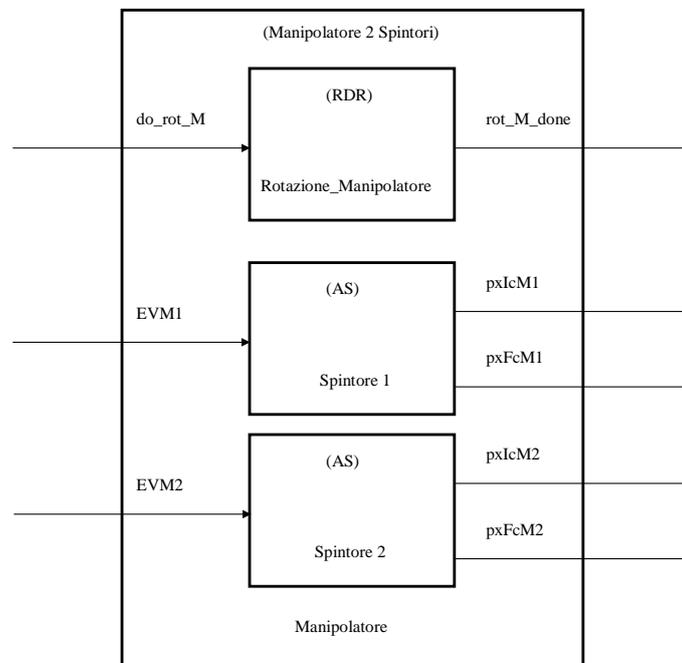


Figura 9

I segnali di ingresso del modulo composto Manipolatore 2 Spintori consistono nel comando di rotazione del Manipolatore `do_rot_M` e nei due segnali che comandano le elettrovalvole associate ai due spintori `EVM1` e `EVM2` (dove 1 è riferito allo spintore 1 e 2 allo spintore 2 di ogni Manipolatore).

I segnali di uscita del modulo consistono nel feedback di rotazione avvenuta `rot_M_done` e nei quattro segnali dei proximity switches posti sui due spintori (`pxlcM1` e `pxFcM1` per lo spintore 1, `pxlcM2` e `pxFcM2` per lo spintore 2).

3.2.3.2. Modulo Manipolatore con 3 Spintori

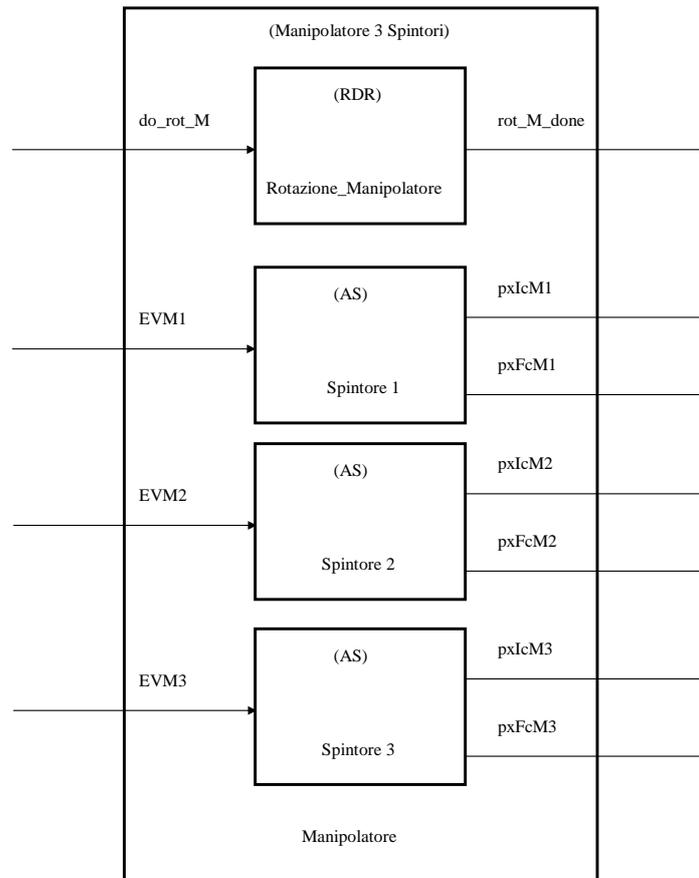


Figura 10

I segnali di ingresso del modulo composto Manipolatore 3 Spintori consistono nel comando di rotazione del Manipolatore `do_rot_M` e nei due segnali che comandano le elettrovalvole associate ai tre spintori `EVM1`, `EVM2` e `EVM3` (dove 1 è riferito allo spintore 1, 2 allo spintore 2 e 3 allo spintore 3 dei Manipolatori che hanno anche il terzo spintore).

I segnali di uscita del modulo consistono nel feedback di rotazione avvenuta `rot_M_done` e nei sei segnali dei proximity switches posti sui tre spintori (`pxIcM1` e `pxFcM1` per lo spintore 1, `pxIcM2` e `pxFcM2` per lo spintore 2, `pxIcM3` e `pxFcM3` per lo spintore 3).

3.2.4. Modulo Terna

Il modulo terna contiene i moduli composti di Tavola, Isola e Manipolatore, specifici per la terna da modellare (infatti non tutte le terne sono esattamente uguali in termini di numero di spintori).

Il modulo terna presenta segnali di I/O di interfaccia che sono la sommatoria di tutti quelli dei moduli composti utilizzati per costituirlo.

A rigor di logica bisognerebbe definire un modulo Terna per ogni tipologia di terna esistente. In particolare si ha:

- Terna A (da utilizzarsi per istanziare i moduli per le terne 1 e 3)
 - Modulo Tavola
 - Modulo Isola con 2 Spintori
 - Modulo Manipolatore con 3 Spintori

- Terna B (da utilizzarsi per istanziare i moduli per le terne 2, 4 e 5)
 - Modulo Tavola
 - Modulo Isola con 1 Spintore
 - Modulo Manipolatore con 3 Spintori
- Terna C (da utilizzarsi per istanziare i moduli per la terna 6)
 - Modulo Tavola
 - Modulo Isola con 1 Spintore
 - Modulo Manipolatore con 2 Spintori

Per non appesantire eccessivamente la trattazione della presente documentazione, si illustra solamente il tipo Terna A in Figura 11.

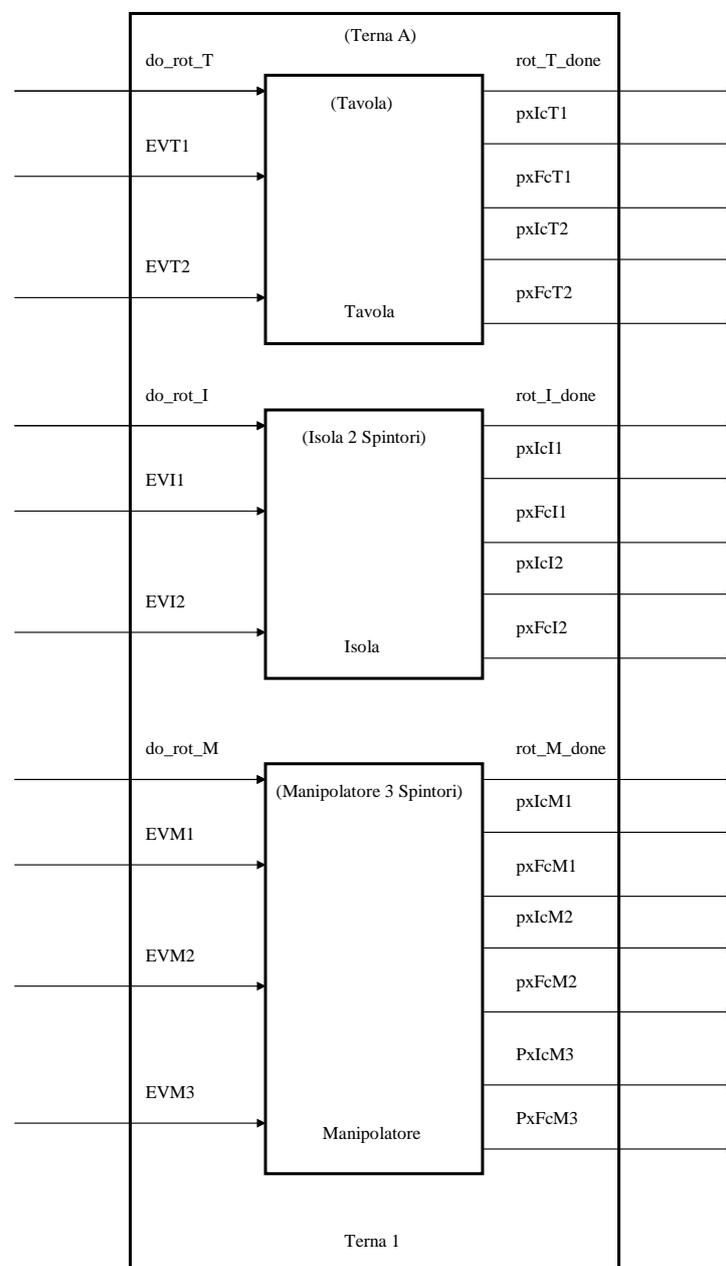


Figura 11

Per poter verificare la correttezza del modello del processo fisico attraverso la simulazione è stato necessario realizzare un modulo di gestione dell'informazione associata agli slots (descritto successivamente) ed inserirlo nel modulo terna, in quanto la simulazione non prevede al momento l'interfacciamento del modello del simulatore al sistema di controllo d'impianto.

3.3. MODULO GESTIONE DELL'INFORMAZIONE ASSOCIATA AGLI SLOTS

Il modulo di gestione dell'informazione associata agli slots è concettualmente separato dagli altri moduli che descrivono il comportamento fisico dei dispositivi in relazione alle azioni di controllo fornite dal sistema di controllo d'impianto.

Infatti l'informazione di ciascuno slot non è detto che debba fare parte del modello del simulatore del processo fisico. In particolare potrebbe essere il sistema di controllo a detenere l'informazione associata agli slots; così come potrebbe essere il simulatore di processo ad aggiornare tale informazione dopo ogni spostamento delle diverse forme di scarpe che circolano sulla linea di trasporto molecolare. E' una scelta che deve essere fatta, ma che non condiziona assolutamente il resto del modello del simulatore. Questo è il motivo per cui la gestione delle stringhe è stata sviluppata e presentata separatamente dal modello fisico dell'impianto descritto attraverso i moduli precedenti.

E' ovvio che il modulo che gestisce l'informazione associata agli slots deve necessariamente ricevere in ingresso tutte quelle azioni di controllo e quei feedback che tornano dal simulatore di processo e che interessano direttamente il movimento delle forme. Per esempio ad ogni comando di rotazione di un dispositivo rotante, il modulo di gestione delle informazioni deve aggiornare il contenuto di tutti gli slots in relazione alla nuova posizione raggiunta.

Così come al comando di uscita di uno spintore, e quindi allo spostamento di una forma, l'informazione associata agli slots coinvolti nello spostamento della forma deve essere aggiornata.

Anche l'arrivo di una forma sul braccio del Manipolatore (proveniente dalla Tavola della terna successiva) deve essere tenuto conto, anche se ciò non coinvolge alcun segnale di controllo della terna attuale in quanto è lo spintore della Tavola della terna successiva che in questo caso sta lavorando.

Nel caso in cui il modulo di gestione dell'informazione associata agli slots faccia parte del simulatore di processo, esso deve essere considerato e sviluppato come un modulo a sé stante e non deve interferire minimamente con gli altri moduli che descrivono le funzionalità meccaniche dei vari dispositivi. In tal caso va evidenziato il fatto che per ottenere il completo incapsulamento dell'informazione i codici che identificano le varie forme di scarpe in lavorazione vengono scambiati tra una terna ed una adiacente attraverso variabili di interfaccia. Così facendo l'informazione associata agli slots di una terna rimane confinata nel relativo modulo terna. Dato che tale modulo non deve fornire alcun feedback di sorta al sistema di controllo d'impianto in quanto non è un dispositivo fisico, esso non ha segnali di uscita.

I segnali di ingresso di tale modulo sono elencati qui di seguito:

(segnali provenienti dal sistema di controllo d'impianto)

do_rot_T

do_rot_I

do_rot_M

EVT1

EVT2

EVI1

EVI2 (se nella terna considerata è presente il secondo spintore dell'Isola)

EVM1

EVM2

EVM3 (se nella terna considerata è presente il terzo spintore del Manipolatore)

(feedback forniti dal simulatore)

get_shoe (quando diventa 1 significa che è giunta una forma con scarpa sullo slot 1 della Tavola)

get_last (quando diventa 1 significa che è giunta una forma senza scarpa sul braccio 3 del Manipolatore)

get_shoe_Lav1 (quando diventa 1 significa che è giunta la forma con scarpa in lavorazione sullo slot 21 dell'Isola facente parte della terna 1 / slot 19 dell'Isola facente parte della terna 3)

pxlcT1 (resetta la variabile di scambio relativa al codice della forma senza scarpa movimentata verso la terna precedente)

pxlcM3 (resetta la variabile di scambio relativa al codice della forma con scarpa movimentata verso la terna successiva)

4. IMPLEMENTAZIONE DEL SIMULATORE CON ISAGRAF PRO

In questo capitolo viene illustrata l'implementazione dei vari moduli presentati precedentemente attraverso l'uso dell'ambiente di sviluppo ISAGRAF PRO e per mezzo del linguaggio di programmazione SFC (Sequential Functional Chart) [2].

4.1. SEQUENZE BASE

Le sequenze base sviluppate nell'ambiente ISAGRAF PRO per mezzo del linguaggio di programmazione SFC implementano le funzionalità dei diversi moduli base. Tale linguaggio di programmazione non è ad oggetti e non consente di sfruttare a pieno le caratteristiche di modularità ed incapsulamento del software proprie dei moduli funzionali presentati precedentemente. Tuttavia tali concetti hanno aiutato a progettare il modello del simulatore di processo in modo più efficiente.

4.1.1. Sequenza Rotazione Dispositivo Rotante

La sequenza di rotazione di un dispositivo rotante di fatto deve resettare a 0 e poi settare nuovamente a 1, dopo un tempo prestabilito, il segnale di fine rotazione del dispositivo, vedi Figura 12.

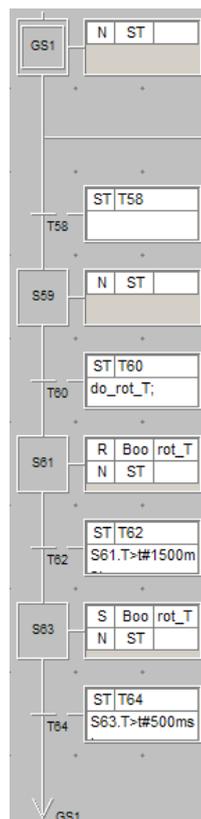


Figura 12

Dopo la prima transizione sempre verificata e la prima fase a cui non è associata alcuna azione, questo per poter svolgere la sequenza ogni volta che è richiesta la rotazione del dispositivo rotante, viene eseguito il test (T60) sulla richiesta di rotazione. Quando questa è verificata, viene resettato a zero il segnale di fine rotazione. Dopo 1500ms (tempo stimato per la rotazione di uno slot) tale segnale viene settato nuovamente a 1. La sequenza riparte dallo stato iniziale e torna a verificare la transizione T60.

4.1.2. Sequenza Azionamento Spintore

La sequenza del generico spintore deve settare e resettare i segnali di inizio-corsa e fine-corsa, a secondo del segnale di comando dato all'elettrovalvola, vedi Figura 13.

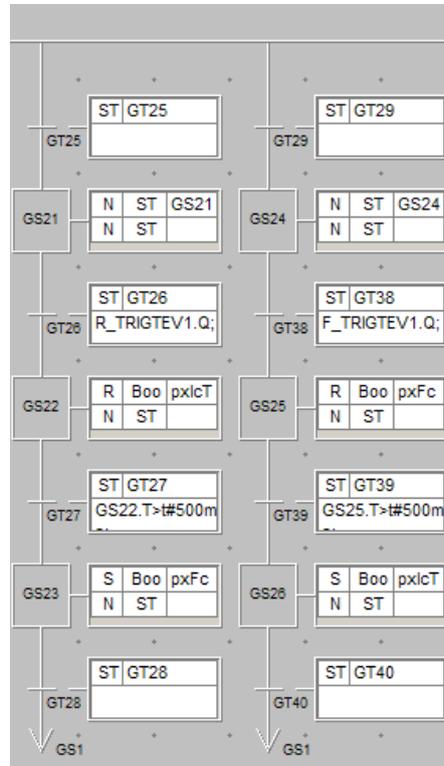


Figura 13

Il primo ramo della sequenza attende il fronte di salita positivo (GT26) della transizione del comando dato all'elettrovalvola (uscita spintore). Quindi viene resettato il segnale di inizio-corsa. Dopo un tempo di 500ms, tempo stimato per l'uscita dello spintore, viene settato a 1 il segnale di fine-corsa.

La seconda sequenza funziona in modo complementare. Viene atteso il fronte negativo (GT38) della transizione del comando dato all'elettrovalvola (rientro spintore). Quindi viene resettato il segnale di fine-corsa. Dopo un tempo di 500ms, tempo stimato per il rientro dello spintore, viene settato a 1 il segnale di inizio-corsa.

Come anticipato nella descrizione del comportamento fisico dello spintore, tale sequenza non prevede l'implementazione software di tale comportamento. Infatti se il comando all'elettrovalvola fosse resettato prima che la sequenza di uscita pistone fosse terminata, inizierebbe comunque l'esecuzione della sequenza rientro pistone. Si otterrebbe così come risultato dell'esecuzione del codice software un comportamento simulato diverso da quello fisico del dispositivo.

4.2. SEQUENZE COMPOSTE

Le sequenze composte sono realizzate a partire dalle sequenze di base sopra esposte al fine di implementare le funzionalità dei dispositivi che compongono la linea molecolare.

4.2.1. Sequenza della Tavola

Il programma della Tavola presenta la funzione di rotazione e quella di due spintori, vedi Figura 14.

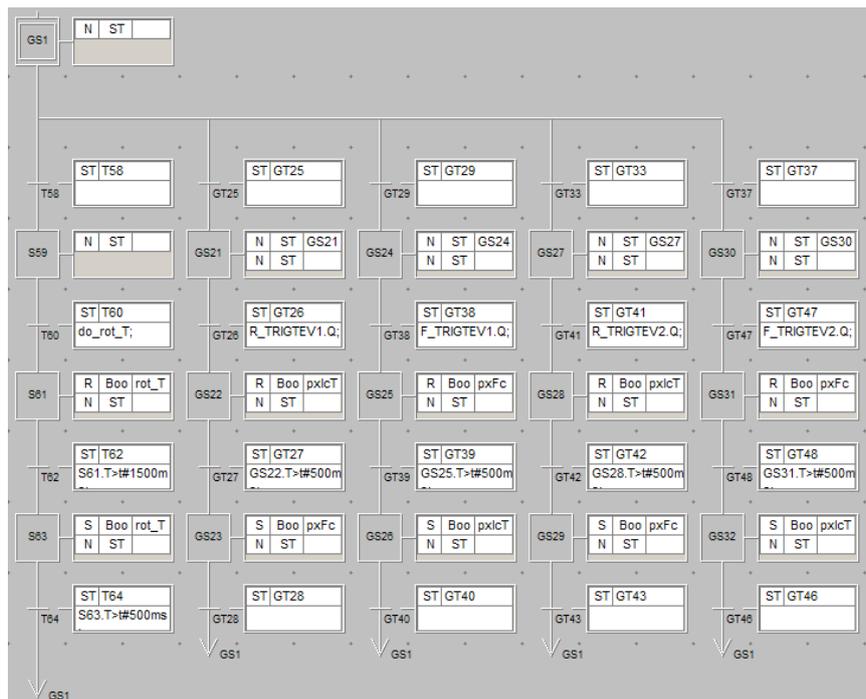


Figura 14

- Il primo ramo della sequenza composta implementa la rotazione della Tavola.
- Il secondo e terzo ramo implementano le funzionalità dello spintore numero 1.
- Il quarto e quinto ramo implementano le funzionalità dello spintore numero 2.

4.2.2. Sequenza dell'Isola

Il programma dell'Isola presenta la funzione di rotazione e quella di uno o due spintori, in base all'Isola modellata. Nel caso specifico si considera un'Isola con 2 spintori, come per la terna 1 e 3. La rappresentazione del relativo programma è illustrata in Figura 15.

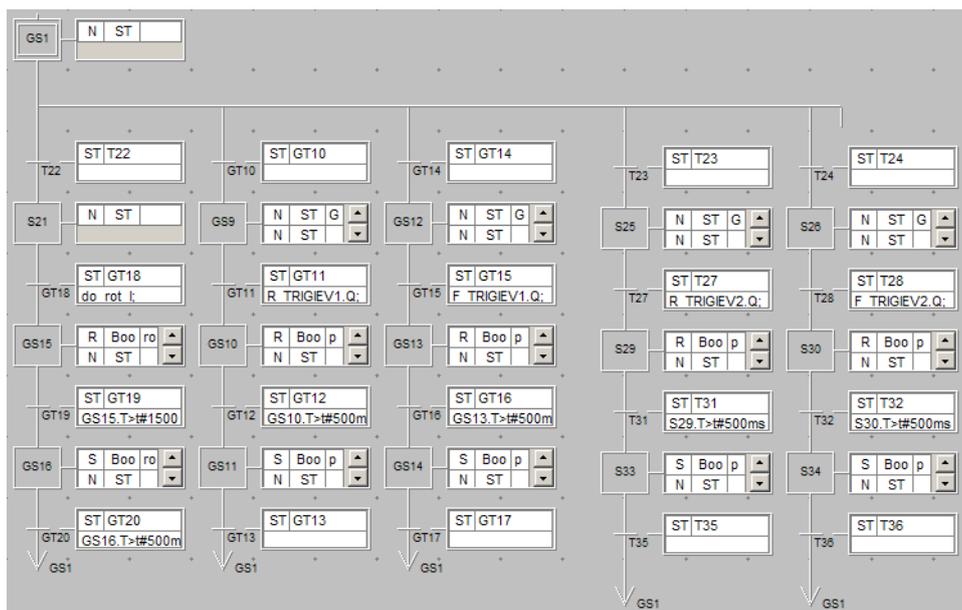


Figura 15

Il primo ramo della sequenza composta implementa la rotazione dell'Isola.

Il secondo e terzo ramo implementano le funzionalità dello spintore numero 1.

Il quarto e quinto ramo implementano le funzionalità dello spintore numero 2.

4.2.3. Sequenza del Manipolatore

Il programma del Manipolatore presenta la funzione di rotazione e quella di due o tre spintori, in base al Manipolatore modellato. Nel caso specifico si considera un Manipolatore con 3 spintori, come per tutte le terne tranne la 6. La rappresentazione del relativo programma è illustrata in Figura 16.

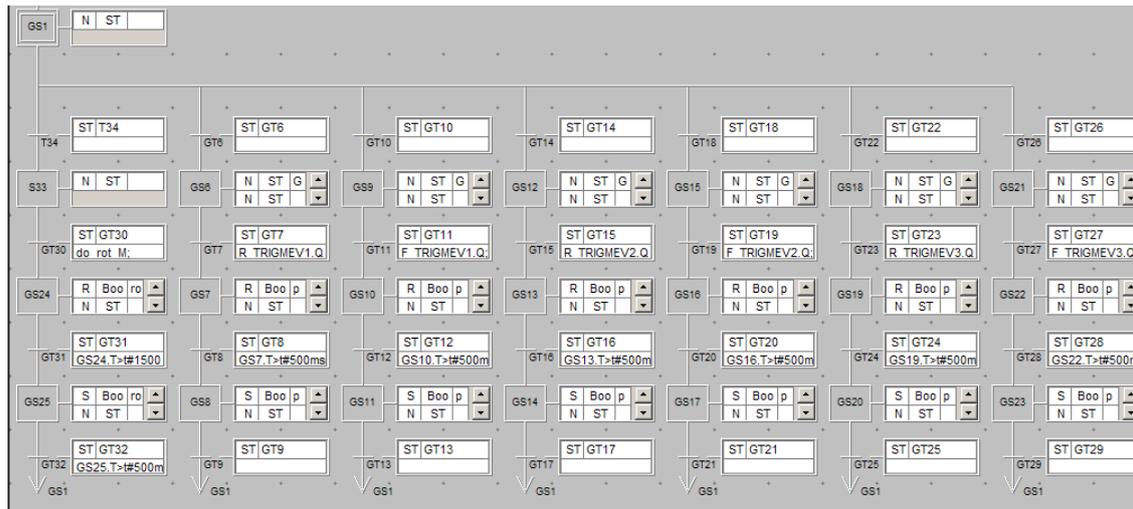


Figura 16

Il primo ramo della sequenza composta implementa la rotazione del Manipolatore.

Il secondo e terzo ramo implementano le funzionalità dello spintore numero 1.

Il quarto e quinto ramo implementano le funzionalità dello spintore numero 2.

Il sesto e settimo ramo implementano le funzionalità dello spintore numero 3.

4.2.4. Sequenza Gestione dell'Informazione associata agli Slots

Tale programma ha il compito di gestire l'informazione associata agli slots nell'intera terna, vedi Figura 17.

Dato che le terne non sono tutte uguali bensì presentano particolarità in termini di numero di spintori, di senso di rotazione di Isole e Manipolatori (i quali, anche se ruotano tutti in senso antiorario, necessitano di trattare l'informazione associata ai relativi slots diversamente in base alla relativa posizione fisica nel layout d'impianto), i relativi programmi di gestione dell'informazione saranno specifici per ognuna di esse o per gruppi di esse.



Figura 17

Il primo ramo della sequenza composta si occupa di gestire l'informazione durante la rotazione della Tavola. Al comando di rotazione viene splittata opportunamente in due la stringa che contiene le informazioni dalla Tavola, quindi viene ricomposta, ottenendo di fatto la rotazione di tale stringa circolare.

Il secondo ramo aggiorna la stringa della Tavola e la variabile `give_Last_code` come conseguenza dell'attivazione dello spintore 1 della Tavola stessa.

Il terzo ramo aggiorna la stringa della Tavola e del Manipolatore come conseguenza dell'attivazione dello spintore 2 della Tavola stessa.

Il quarto ramo della sequenza composta si occupa di gestire l'informazione durante la rotazione dell'Isola.

Il quinto ramo aggiorna la stringa dell'Isola e del Manipolatore come conseguenza dell'attivazione dello spintore 1 dell'Isola stessa.

Il sesto ramo della sequenza composta si occupa di gestire l'informazione durante la rotazione del Manipolatore.

Il settimo ramo aggiorna la stringa del Manipolatore e della Tavola come conseguenza dell'attivazione dello spintore 1 del Manipolatore stesso.

L'ottavo ramo aggiorna la stringa del Manipolatore e dell'Isola come conseguenza dell'attivazione dello spintore 2 del Manipolatore stesso.

Il nono ramo aggiorna la stringa del Manipolatore e della variabile `giveShoeCode` come conseguenza dell'attivazione dello spintore 3 del Manipolatore stesso.

Il decimo ramo aggiorna la stringa della Tavola e la variabile `getShoeCode` come conseguenza del comando, dato dal sistema di controllo, di acquisizione di una forma con scarpa dalla terna precedente.

L'undicesimo ramo aggiorna la stringa del Manipolatore e la variabile `getLastCode` come conseguenza del comando di acquisizione di una forma dalla terna successiva.

Il dodicesimo ramo aggiorna la stringa dell'Isola come conseguenza dell'attivazione dello spintore 2 dell'Isola stessa (per le Isole che ne sono dotate).

Il tredicesimo ramo aggiorna la stringa dell'Isola come conseguenza del comando, dato dal sistema di controllo, di acquisizione della scarpa lavorata dalla macchina operatrice (per le sole Isole che ne sono dotate).

Il quattordicesimo ramo resetta la variabile di scambio giveLastCode utilizzata nello scambio dei codici scarpa verso la terna precedente.

Il quindicesimo ramo resetta la variabile di scambio giveShoeCode utilizzata nello scambio dei codici scarpa verso la terna successiva.

4.3. PROGRAMMA DELLA TERNA

I programmi presentati precedentemente vengono combinati assieme in un unico programma per implementare le funzionalità di un'intera terna, sempre per rispettare il principio di modularità della progettazione del simulatore, vedi Figura 18.



Figura 18

A parte programmi di servizio definiti dall'ambiente ISAGRAF PRO, è possibile vedere i quattro programmi implementati e precedentemente esposti:

Gest_Str	Gestione dell'informazione associata agli slots della terna
ECC_T	Programma della Tavola
ECC_I	Programma dell'Isola
ECC_Man	Programma del Manipolatore

Si può notare come i vari programmi sono indipendenti l'uno dall'altro. Infatti i programmi ECC_T, ECC_I e ECC_M implementano le funzionalità fisiche rispettivamente della Tavola, Isola e Manipolatore, funzionalità che sono proprie di ciascun dispositivo.

Il programma Gest_Str, come detto in precedenza, è di per sé indipendente dagli altri programmi.

4.4. PROGRAMMA DEL SIMULATORE

I vari programmi specializzati per le diverse terne, legati tra loro da variabili di binding che sono le sole variabili di scambio tra i vari programmi delle terne, formano l'intero programma del simulatore della linea molecolare, vedi Figura 19.

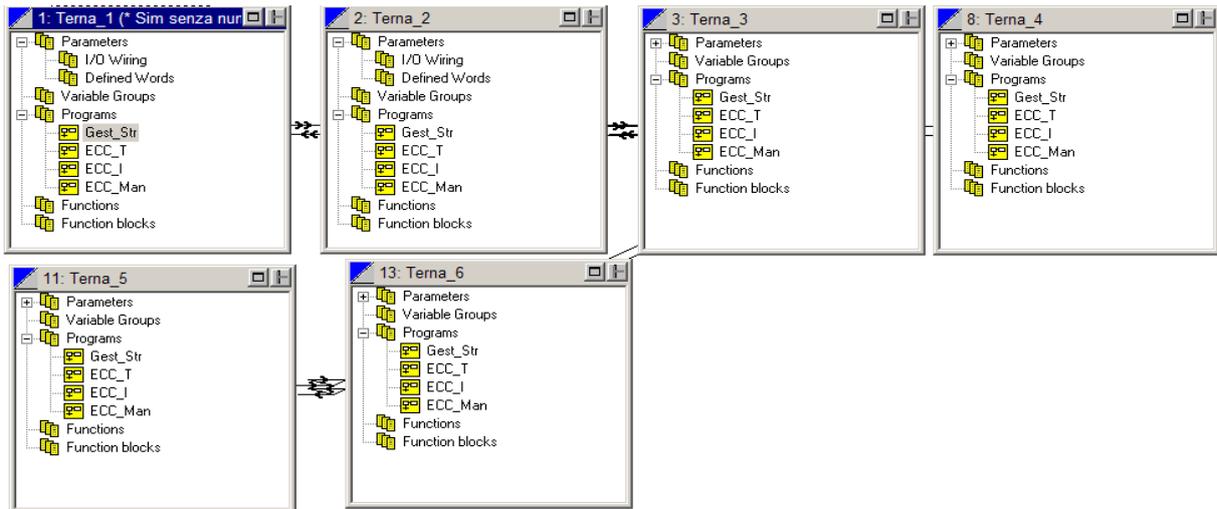


Figura 19

Come detto in precedenza, vengono utilizzate solo variabili di binding (visualizzate dalle frecce nere che collegano i vari programmi delle terne) tra le varie terne per mantenere valido il principio di modularità nella progettazione del simulatore.

5. VERIFICA DEL MODELLO DELLA LINEA MOLECOLARE ATTRAVERSO SIMULAZIONE

Il comportamento funzionale del modello di simulazione progettato ed implementato è stato verificato attraverso la simulazione condotta nell'ambiente ISAGRAF PRO.

Tale ambiente non consente di costruire un'interfaccia grafica attraverso la quale l'utente può seguire l'evolvere delle variabili simulate a fronte degli ingressi forniti al modello. Ciò che questo ambiente consente è di ottenere un pannello grafico nel quale sono visualizzati i valori delle variabili booleane ed i valori alfanumerici delle variabili stringhe simulate. Per ottenere questo risultato bisogna associare le variabili definite Input / Output ai canali virtuali di I/O specificati nell'ambiente ISAGRAF PRO.

Un'alternativa al pannello grafico è quella di leggere il valore delle variabili simulate (scritte da ISAGRAF PRO in un apposito dizionario) ed utilizzarle in un opportuno ambiente di programmazione che permette di realizzare l'interfaccia grafica del simulatore desiderata.

Di seguito sono elencate le variabili interne utilizzate per la memorizzazione dell'informazione degli slots e che, appropriatamente lette e rappresentate graficamente, permettono di visualizzare in modo dinamico l'evolvere del modello simulato.

Nome Variabile Simulata	Tipo	Significato	Dispositivo	Terna di Riferimento
Variabili Interne da usare per la rappresentazione grafica				
T	Stringa (36 bit)	Stringa che descrive il contenuto informativo degli slot	Tavola	1, 2, 3, 4, 5, 6
M	Stringa (9 bit)	Stringa che descrive il contenuto informativo degli slot	Manipolatore	1, 2, 3, 4, 5, 6
I	Stringa (72 bit)	Stringa che descrive il contenuto informativo degli slot	Isola	1, 2, 3, 4, 5, 6

6. TABELLA RIASSUNTIVA SEGNALI DI INTERFACCIA SIMULATORE – SISTEMA DI CONTROLLO

Qui di seguito è riportata la tabella che riassume i segnali di interfaccia tra il simulatore ed il sistema di controllo d'impianto.

Nel caso in cui l'interfacciamento tra il simulatore di linea ed il sistema di controllo avvenga per mezzo di segnali fisici, è evidente che il nome di tali variabili non è significativo, in quanto basta rinominare i segnali associati ai vari morsetti elettrici (a cui sono associate le variabili del simulatore) per poter disporre di un'altra nomenclatura.

Per convenzione si considerano Input per il simulatore le azioni di controllo che provengono dal sistema di controllo. Output per il simulatore i segnali di feedback che esso fornisce al sistema di controllo.

Nome Variabile Simulata	Tipo	Significato	Dispositivo	Terna di Riferimento
Digital Input (comando per il Simulatore)				
do_rot_T	Boolean	Comando di rotazione	Tavola	1, 2, 3, 4, 5, 6
EVT1	Boolean	Comando di apertura Elettro-Valvola Uscita Spintore 1	Tavola	1, 2, 3, 4, 5, 6
EVT1	Boolean	Comando di apertura Elettro-Valvola Rientro Spintore 1	Tavola	1, 2, 3, 4, 5, 6
EVT2	Boolean	Comando di apertura Elettro-Valvola Uscita Spintore 2	Tavola	1, 2, 3, 4, 5, 6
EVT2	Boolean	Comando di apertura Elettro-Valvola Rientro Spintore 2	Tavola	1, 2, 3, 4, 5, 6
Digital Input (comando per il Manipolatore)				
do_rot_M	Boolean	Comando di rotazione	Manipolatore	1, 2, 3, 4, 5, 6
EVM1	Boolean	Comando di apertura Elettro-Valvola Uscita Spintore 1	Manipolatore	1, 2, 3, 4, 5, 6
EVM1	Boolean	Comando di apertura Elettro-Valvola Rientro Spintore 1	Manipolatore	1, 2, 3, 4, 5, 6
EVM2	Boolean	Comando di apertura Elettro-Valvola Uscita Spintore 2	Manipolatore	1, 2, 3, 4, 5, 6
EVM2	Boolean	Comando di apertura Elettro-Valvola Rientro Spintore 2	Manipolatore	1, 2, 3, 4, 5, 6
EVM3	Boolean	Comando di apertura Elettro-Valvola Uscita Spintore 3	Manipolatore	1, 2, 3, 4, 5
EVM3	Boolean	Comando di apertura Elettro-Valvola Rientro Spintore 3	Manipolatore	1, 2, 3, 4, 5
Digital Input (comando per l'Isola)				
do_rot_I	Boolean	Comando di rotazione	Isola	1, 2, 3, 4, 5, 6
EVI1	Boolean	Comando di apertura Elettro-Valvola Uscita Spintore 1	Isola	1, 2, 3, 4, 5, 6
EVI1	Boolean	Comando di apertura Elettro-Valvola Rientro Spintore 1	Isola	1, 2, 3, 4, 5, 6
EVI2	Boolean	Comando di apertura Elettro-Valvola Uscita Spintore 2	Isola	1, 3
EVI2	Boolean	Comando di apertura Elettro-Valvola Rientro Spintore 2	Isola	1, 3
Digital Output (feedback dal Simulatore)				
rot_T_done	Boolean	Rotazione avvenuta (1 transizione segnale, 1 slot ruotato)	Tavola	1, 2, 3, 4, 5, 6

pxIcT1	Boolean	Spintore 1 in Posizione Inizio Corsa (Pistone dentro)	Tavola	1, 2, 3, 4, 5, 6
pxFcT1	Boolean	Spintore 1 in Posizione Fine Corsa (Pistone fuori)	Tavola	1, 2, 3, 4, 5, 6
pxIcT2	Boolean	Spintore 2 in Posizione Inizio Corsa (Pistone dentro)	Tavola	1, 2, 3, 4, 5, 6
pxFcT2	Boolean	Spintore 2 in Posizione Fine Corsa (Pistone fuori)	Tavola	1, 2, 3, 4, 5, 6
rot_M_done	Boolean	Rotazione avvenuta (1 transizione segnale, 1 slot ruotato)	Manipolatore	1, 2, 3, 4, 5, 6
pxIcM1	Boolean	Spintore 1 in Posizione Inizio Corsa (Pistone dentro)	Manipolatore	1, 2, 3, 4, 5, 6
pxFcM1	Boolean	Spintore 1 in Posizione Fine Corsa (Pistone fuori)	Manipolatore	1, 2, 3, 4, 5, 6
pxIcM2	Boolean	Spintore 2 in Posizione Inizio Corsa (Pistone dentro)	Manipolatore	1, 2, 3, 4, 5, 6
pxFcM2	Boolean	Spintore 2 in Posizione Fine Corsa (Pistone fuori)	Manipolatore	1, 2, 3, 4, 5, 6
pxIcM3	Boolean	Spintore 3 in Posizione Inizio Corsa (Pistone dentro)	Manipolatore	1, 2, 3, 4, 5
pxFcM3	Boolean	Spintore 3 in Posizione Fine Corsa (Pistone fuori)	Manipolatore	1, 2, 3, 4, 5
rot_I_done	Boolean	Rotazione avvenuta (1 transizione segnale, 1 slot ruotato)	Isola	1, 2, 3, 4, 5, 6
pxIcI1	Boolean	Spintore 1 in Posizione Inizio Corsa (Pistone dentro)	Isola	1, 2, 3, 4, 5, 6
pxFcI1	Boolean	Spintore 1 in Posizione Fine Corsa (Pistone fuori)	Isola	1, 2, 3, 4, 5, 6
pxIcI2	Boolean	Spintore 2 in Posizione Inizio Corsa (Pistone dentro)	Isola	1, 3
pxFcI2	Boolean	Spintore 2 in Posizione Fine Corsa (Pistone fuori)	Isola	1, 3

Per completezza di documentazione qui di seguito vengono riportate due variabili interne utilizzate per l'inserimento del codice scarpa nello slot numero 1 della prima Tavola, visto che il magazzino non è stato modellato.

Nome Variabile Simulata	Tipo	Significato	Dispositivo	Terna di Riferimento
Variabili Interne da usare per inizializzare lo slot n° 1 della Tavola della 1° Terna				
getShoeCode	Stringa (3 bit)	Codice scarpa che entra in linea	Tavola	1
getShoe	Boolean	Comando di assegnazione del valore di "getShoeCode" allo slot n° 1 sulla Tavola della Terna n° 1	Tavola	1

7. CONCLUSIONI E LAVORI FUTURI

Nel presente report è stato descritto il modello di simulazione della linea molecolare, la relativa implementazione nel linguaggio di programmazione SFC operata nell'ambiente di sviluppo ISAGRAF PRO nonché le variabili da utilizzare nell'interfaccia con il sistema di controllo della linea di trasporto.

Sviluppi futuri di tale lavoro riguardano un'eventuale modellazione delle stazioni operatore disposte lungo la linea di movimentazione molecolare e relativa documentazione.

8. ACRONIMI E DEFINIZIONI

8.1. TERMINI UTILIZZATI IN CONTROLLI AUTOMATICI

Advanced control—Process control strategies beyond PID loop control, such as feed forward, dead-time compensation, lead/lag, adaptive gain, neural networks, and fuzzy logic.

Fieldbus architecture—Control architecture that uses digital, serial, multi drop, two-way communications between and among intelligent field devices and control/monitoring systems.

Human-machine interface—Method of displaying machine status, alarms, messages, and diagnostics, often graphical display on a personal computer, providing operator feedback.

IEC 61131—International standard for machine control programming tools. Part Three provides five languages with standard commands and data structure, allowing changes to programming software with less extensive training.

IEC 61499—International standard for industrial-process measurement and control systems. Part One provides functional blocks allowing to describe functional control systems architecture.

Intelligent field devices—Microprocessor-based devices capable of providing multiple process variables, device performance information, diagnostic results, and execution of assigned control functions.

Intelligent I/O modules—I/O module that provides intelligent, on-board processing of input values to control output values, bypassing the PLC or control controller for routine decision making.

Internet—Global collection of industrial, commercial, academic, government, and personal computer networks that exchange information.

Interoperability—When products are replaceable by a similar product from another vendor.

MES—Manufacturing Execution System delivers information-enabling optimisation of production activities from order to goods. It guides, initiates, responds to, and reports on plant activities.

Microsoft Windows Operating Systems—The most widely used operating systems for personal computers. Microsoft NT is a desktop and server package for enterprise-wide applications. Microsoft 95 is a self-contained operating system a built-in and enhanced version of DOS. Microsoft CE is a compact version of Windows for handheld PCs and embedded devices.

Object-oriented software—Software that uses and reuses parcels of code to build applications modelled on object techniques including COM/DCOM, Java, and CORBA standards.

OLE for process control (OPC)—Object linking & embedding (OLE) that treats data as collections of objects to be shared by applications supporting OLE specifications. OPC provides extensions to OLE to support process control data sharing.

Open controller—Controller that looks like a traditional PLC but is a PC operating in a Windows environment with software control.

Open systems—Hardware/software designs in which a degree of interchangeability and connectivity give users choices. Systems complying with the seven layers of the ISO-proposed open-system interconnect, 7-layer model.

PC control—Software-configured control strategy using standard personal computer hardware and software.

PID (Proportional, integral, derivative control)—An intelligent I/O module or program instruction which provides automatic closed-loop operation of process control loops.

Programmable Logic Controller (PLC)—A solid-state control system with user-programmable memory for storage of instructions to implement specific control and automation functions.

S88—An international standard developed by ISA that uses object-oriented concepts to define terminology and models for batch control processes.

Soft logic—Controller is the software, which can run on a variety of personal-computer form factors. Most useful in applications requiring high data collection and processing as well as communications to other networks.

8.2. ACRONIMI

MES - Manufacturing Execution System

OPC – Ole for Process Control

PID – Proportional Integral Derivative

PLC – Programmable Logic Controller

SFC - Sequential Functional Chart

CN – Controllo Numerico

9. RIFERIMENTI E LINKS

9.1. RIFERIMENTI

- [1] Transport Cell Control System Strategy, F. Airoidi, G. Carlino, E.Carpanzano, A. Cataldo
- [2] Programming industrial control systems using IEC 1131-3, R.W. Lewis

9.2. INTERNET LINKS

ISAGRAF PRO <http://www.altersys.com/index.asp>