



Ministero dell'Istruzione, dell'Università e della Ricerca



Istituto di Istruzione Superiore "Benedetto Castelli"

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale

Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it



TROFEO SMART PROJECT CATEGORIA PROMESSE



Ministero dell'Istruzione, dell'Università e della Ricerca



Istituto di Istruzione Superiore “Benedetto Castelli”

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale

Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it

Virtual PLC



Sommario

Sommario	3
OBBIETTIVO	5
DESCRIZIONE PROGETTO.....	5
PROGRAMMA PLC	6
Variabili programmazione CX_PROGRAMMER	6
Processi con allegato programma plc.....	8
Programma “SCELTA”	8
Marcia arresto manuale di un motore asincrono trifase (MAT)	10
Inversione di marcia manuale di un motore (MAT)	12
Avviamento stella-triangolo di un motore MAT.....	14
Semaforo pedonale	17
Cancello automatico.....	21
Garage automatico	24
Controllo livello di un serbatoio	29
Miscelatore di due liquidi	32
Ciclo elettropneumatico con un cilindro a doppio effetto.....	39
Ciclo elettropneumatico con due cilindri a doppio effetto	41
Ciclo elettropneumatico con tre cilindri a doppio effetto.....	43
PROGRAMMAZIONE CX-SUPERVISOR	46
Variabili programmazione Cx_supervisor.....	46
Layout applicazione	51
Marcia arresto di un motore MAT.....	52
Inversione di marcia di un motore MAT.....	53
Avviamento stella-triangolo di un motore MAT.....	55
Semaforo pedonale	56
Cancello automatico.....	57
Garage automatico	59
Controllo livello di un serbatoio	65
Miscelatore di due liquidi	68
Ciclo elettropneumatico con un cilindro a doppio effetto.....	71
Ciclo elettropneumatico con due cilindri a doppio effetto	73
Ciclo elettropneumatico con tre cilindri a doppio effetto.....	73



Ascensore	74
Plc ingressi uscite.....	75
CIO	76
PARTECIPANTI.....	86



OBBIETTIVO

L' obiettivo che ci siamo proposti di raggiungere è quello di creare un software con vari processi simulati per permettere ai futuri programmati alcuni processi in cui cimentarsi, soprattutto per quando riguarda la parte di programmazione del PLC.

Il software verrà poi fornito anche a degli allievi che si approcciano allo studio del PLC fornendogli un software completo per la simulazione. Il progetto si configura con finalità prevalentemente didattiche.

In particolare vengono forniti vari ambienti che simulano tipici processi industriali e su cui gli allievi si cimentano per migliorare le loro abilità nella programmazione di controllori.

DESCRIZIONE PROGETTO

Il progetto è stato realizzato utilizzando un PLC OMRON della serie CJ1-M con una cpu 23 e con l'utilizzo dei seguenti software: CX-PROGRAMMER (per programmare il plc) ,CX-SUPERVISOR (per programmare la parte grafica) e MICROSOFT EXCEL (per creare i vari listati delle variabili).

Il progetto è stato costruito in modo da fornire un software articolato in più moduli dove vengono simulati processi indicati di seguito.

Ciascun modulo è strutturato in parte descrittiva che funge da manuale di istruzione dove si possono trovare le principali caratteristiche di funzionamento.

All'avvio dell'applicazione si apre sulla pagina di presentazione dove appaiono tutti i processi presenti per la simulazione e alcuni pulsanti per le informazioni generali:

- On/Off per dare l'abilitazione al PLC
- START per far avviare il processo selezionato
- STOP per arrestare il programma
- PLC I/O per poter visualizzare lo stato degli ingressi e delle uscite del PLC
- INFO sono contenute tutte le informazioni riguardanti il processo in esecuzione

Sulla nuova schermata che appare oltre al processo sono presenti alcuni pulsanti generali:

- CHIUDI per chiudere il processo
- I/O per monitorare i vari ingressi e uscite su SUPERVISOR
- START per far partire il processo selezionato
- STOP per arrestare il programma
- INFORMAZIONI PROCESSO sono contenute tutte le informazioni riguardanti il processo in esecuzione
- MEMORIA per monitorare lo stato I/O del PLC e l'area delle memorie counter e timer



PROGRAMMA PLC

Il programma del PLC è stato svolto mediante la programmazione a blocchi. Quindi ogni processo appartiene ad un blocco

Le variabili di ingresso e di uscita sono le stesse perché si è pensato che uno studente svolge un singolo processo alla volta; per risolvere questo problema abbiamo scritto un programma in ST(structure text) che in base al processo aperto setta un bit di un array a true e tutti gli altri a false; questo avviene dopo la trasmissione dei dati da parte di CX-SUPERVISOR dove ogni pulsante di apertura di processo è collegato ad uno script per il settaggio di una variabile a un determinato valore.

Ogni ciclo ha delle lampade di segnalazione per tenere informato l'operatore/studente su quello che sta avvenendo nel ciclo selezionato.

Variabili programmazione CX_PROGRAMMER

NOME VARIABILE	TIPO	INDIRIZZO	COMMENTO
A	INT	W5	PER AZZERAMENTO VARIABILI
CancelloAutomatico	FUNCTION BLOCK		
CICLO	BOOL[15]	W0.00	
ControlloLivello	FUNCTION BLOCK		
ENABLE	BOOL	W0.15	
IN00	BOOL	0.00	IN00
IN01	BOOL	0.01	IN01
IN02	BOOL	0.02	IN02
IN03	BOOL	0.03	IN03
IN04	BOOL	0.04	IN04
IN05	BOOL	0.05	IN05
IN06	BOOL	0.06	IN06
IN07	BOOL	0.07	IN07
START	BOOL	0.08	INStart
IN09	BOOL	0.09	IN09
IN10	BOOL	0.10	IN10
IN11	BOOL	0.11	IN11
IN12	BOOL	0.12	IN12
IN13	BOOL	0.13	IN13
IN14	BOOL	0.14	IN14
IN15	BOOL	0.15	IN15
IN16	BOOL	1.00	IN16
MarciaArresto	FUNCTION BLOCK		
MAT2	FUNCTION BLOCK		
MAT3	FUNCTION BLOCK		
Miscelatore	FUNCTION		



BLOCK			
OLD_OPZIONE	INT	W10	VECCIO VALORE DI OPZIONE
OPZIONE	INT	W15	SCELTA DEL CICLO
OUT00	BOOL	2.00	OUT00
OUT01	BOOL	2.01	OUT01
OUT02	BOOL	2.02	OUT02
OUT03	BOOL	2.03	OUT03
OUT04	BOOL	2.04	OUT04
OUT05	BOOL	2.05	OUT05
OUT06	BOOL	2.06	OUT06
OUT07	BOOL	2.07	OUT07
OUT08	BOOL	2.08	OUT08
OUT09	BOOL	2.09	OUT09
OUT10	BOOL	2.10	OUT10
OUT11	BOOL	2.11	OUT11
OUT12	BOOL	2.12	OUT12
OUT13	BOOL	2.13	OUT13
OUT14	BOOL	2.14	OUT14
OUT15	BOOL	2.15	OUT15
OUTCOUNTER1	INT	D60	uscita visione counter1
OUTCOUNTER2	INT	D70	uscita visione counter2
OUTCOUNTER3	INT	D80	uscita visione counter3
OUTCOUNTER4	INT	D90	uscita visione counter4
OUTCOUNTER5	INT	D100	uscita visione counter5
OUTCOUNTER6	INT	D110	uscita visione counter6
OUTTIMER1	INT	D0	uscita visione timer1
OUTTIMER2	INT	D10	uscita visione timer2
OUTTIMER3	INT	D20	uscita visione timer3
OUTTIMER4	INT	D30	uscita visione timer4
OUTTIMER5	INT	D40	uscita visione timer5
OUTTIMER6	INT	D50	uscita visione timer6
pos3	FUNCTION BLOCK		
SemaforoPedonale	FUNCTION BLOCK		
GARAGE	FUNCTION BLOCK		
TERM1	BOOL	W100.00	
TERM2	BOOL	W100.01	
RESET	BOOL	W100.02	
VARIABILE	INDIRIZZO	COMMENTO	TIPO
CICLO[]	0	reset uscite	BOOL
	1	ciclo mat 1	BOOL
	2	ciclo mat 2	BOOL
	3	ciclo mat 3	BOOL
	4	ciclo semaforo pedonale	BOOL



	5	ciclo cancello automatico	BOOL
	6	ciclo	BOOL
	7	ciclo	BOOL
	8	ciclo	BOOL
	9	ciclo	BOOL
	10	ciclo miscelatore	BOOL
	11	ciclo controllo livello	BOOL

Processi con allegato programma plc

Programma “SCELTA”

Descrizione

Questo algoritmo è stato scritto in linguaggio strutturato e serve per selezionare il processo da far eseguire.

Inoltre setta le variabili a zero di ingresso e uscita nel cambio di processo per evitare che ci siano complicazioni per eventuali settaggi precedenti.

Programma

Per ottimizzare il programma il reset delle variabili viene svolto solo al cambio di ciclo.

```

1.  '(*DOPO AVER RICEVUTO IL VALORE DI POSIZIONE SI ESEGUE LA VERIFICA COMPARANDO I VALORI PRE
CENTI
2. 'CON QUELLI ATTUALI:
3. '-SE CORRISPONDONO TUTTO OK
4. '-SE DIVERSI VENGONO AGGIORNATI AI VALORI ATTUALI ( CICLO[] )
5. 'SUCCESSIVAMENTE MODIFICANO GLI ENABLE DEI BLOCCHI*)
6. If ENABLE Then
7.
8.     If OPZIONE = OLD_OPZIONE Then          '(* SE LA VARIABILE OPZIONE NON E' CAMBIATA NO
N ESEGUE NIENTE *)
9.
10.    OLD_OPZIONE := OPZIONE;
11.
12. Else
13.     '(* AZZERAMENTO USCITE NEL CAMBIO DI STATO*)
14.     OUT00 := False;
15.     OUT01 := False;
16.     OUT02 := False;
17.     OUT03 := False;
18.     OUT04 := False;
19.     OUT05 := False;
20.     OUT06 := False;
21.     OUT07 := False;
22.     OUT08 := False;
23.     OUT09 := False;
24.     OUT10 := False;
25.     OUT11 := False;
26.     OUT12 := False;
27.     OUT13 := False;
28.     OUT14 := False;
29.     OUT15:= False;
30.
31.     '(* AZZERAMENTO USCITE NEL CAMBIO DI STATO*)

```



```

32.           IN00 := False;
33.           IN01 := False;
34.           IN02 := False;
35.           IN03 := False;
36.           IN04 := False;
37.           IN05 := False;
38.           IN06 := False;
39.           IN07 := False;
40.           'IN08 := False;
41.           'IN09 := False;
42.           IN10 := False;
43.           IN11 := False;
44.           IN12 := False;
45.           IN13 := False;
46.           IN14 := False;
47.           IN15:= False;
48.           IN16 := False;
49.

50.           '(*SE LA VARIABILE OPZIONE E' CAMBIATA SI VERIFICA IL VALORE E SI
CAMBIA IL BIT CORRISPONDENTE *)
51.           For A := 1 To 11 By 1 Do          '(*CICLO PER ANDARE A CAMBIARE I VALORI DE
I BIT*)
52.
53.               If A = OPZIONE Then          '(*CONFRONTO TRA INDICE E POSIZION
E*)
54.
55.                   CICLO[OPZIONE] := TRUE;
56. '(*SE IL VALORE E' UGUALE METTO A 1 IL BIT *)
57.
58.               Else
59.
60.                   CICLO[A] := False;
61. '(*SE IL VALORE E' DIVERSO METTO A 0 IL BIT *)
62.
63.               End If;
64.
65.           End For ;
66.
67.           OLD_OPZIONE := OPZIONE;        '(*AGGIORNO IL VALORE DI OLD_OPZIONE*)
68.
69.       End If;
70.
71. End If;;

```

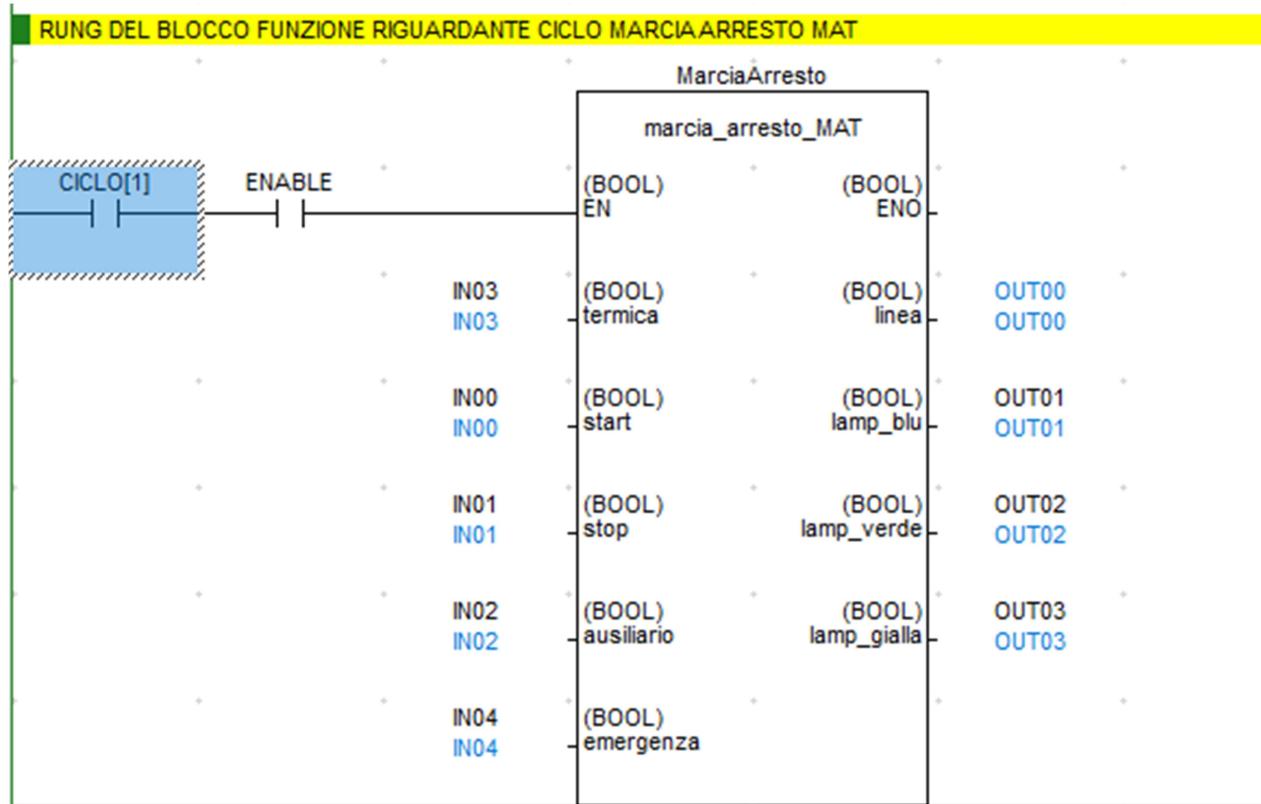


Marcia arresto manuale di un motore asincrono trifase (MAT)

Descrizione

Il processo consiste nell'avviamento di un motore MAT tramite un pulsante di start e un successivo arresto tramite il pulsante di stop. Nella simulazione si possono aggiungere dei temporizzatori per rendere ciclico l'avviamento.

Blocco



Variabili

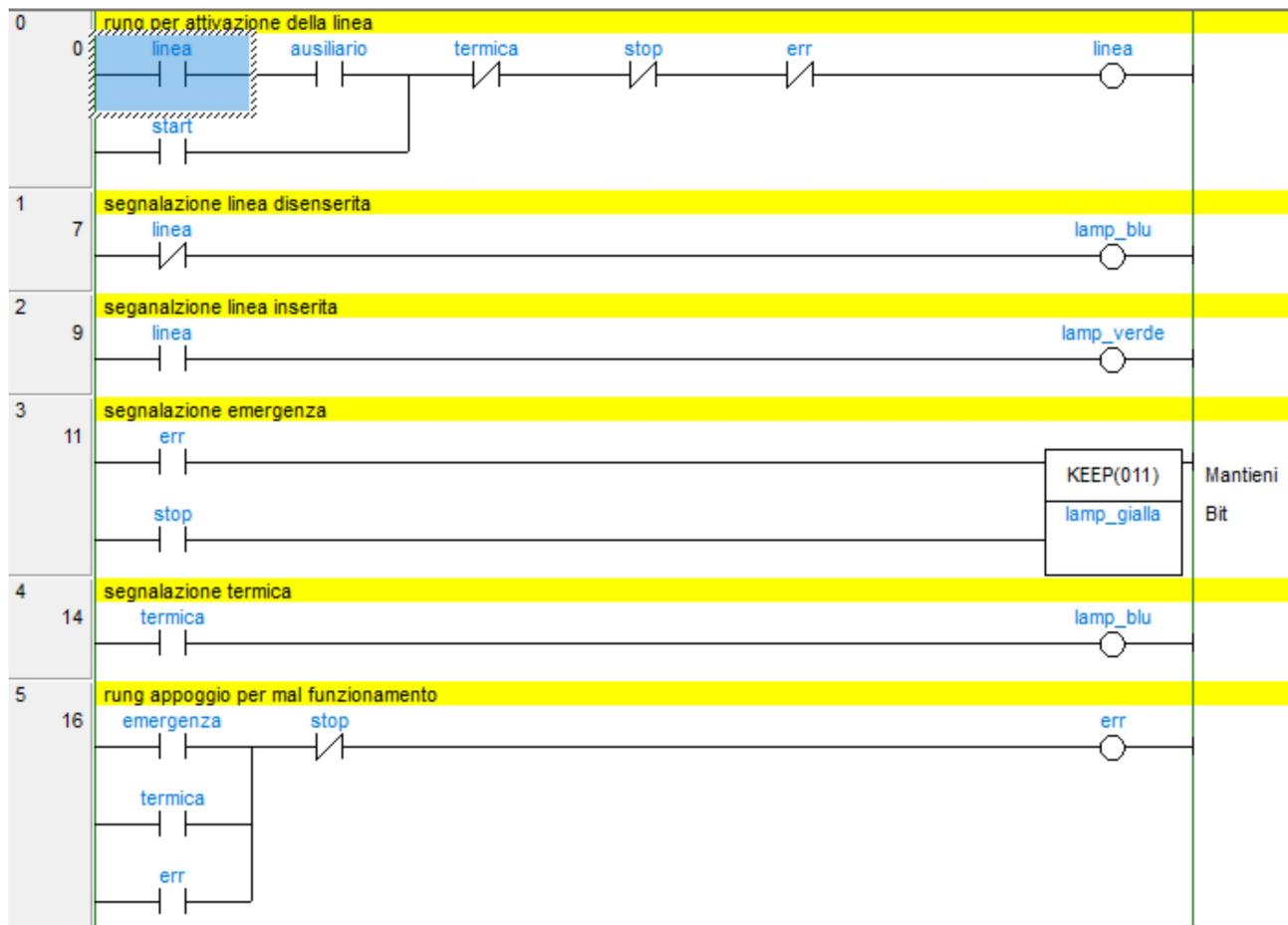
Simbolo	Indirizzo	Commento
IN00	stop	Pulsante (NC) di stop
IN01	stop	Pulsante (NO) di start
IN02	ausiliario	Contatto (NO) ausiliario di KM
IN03	termica	Contatto (NC) relè termico
IN04	emergenza	Pulsante (NC) simulatore emergenza esterna
OUT00	linea	Bobina contattore di linea KM
OUT01	Lamp_blu	Lampada segnalazione blu
OUT02	Lamp_verde	Lampada segnalazione verde
OUT03	Lamp_giallo	Lampada segnalazione gialla



Schema ladder del blocco

Nel rung 0 viene attivata l'uscita (OUT00) per dare l'avvio al motore MAT tramite un pulsante di start(IN00) e un successivo arresto tramite il pulsante di stop(IN01).

Inoltre il motore può essere arrestato tramite l'inserimento di una termica o di un'emergenza che vengono disinserite dal pulsante di stop per mantenere lo stato si usa una variabile appoggio err.



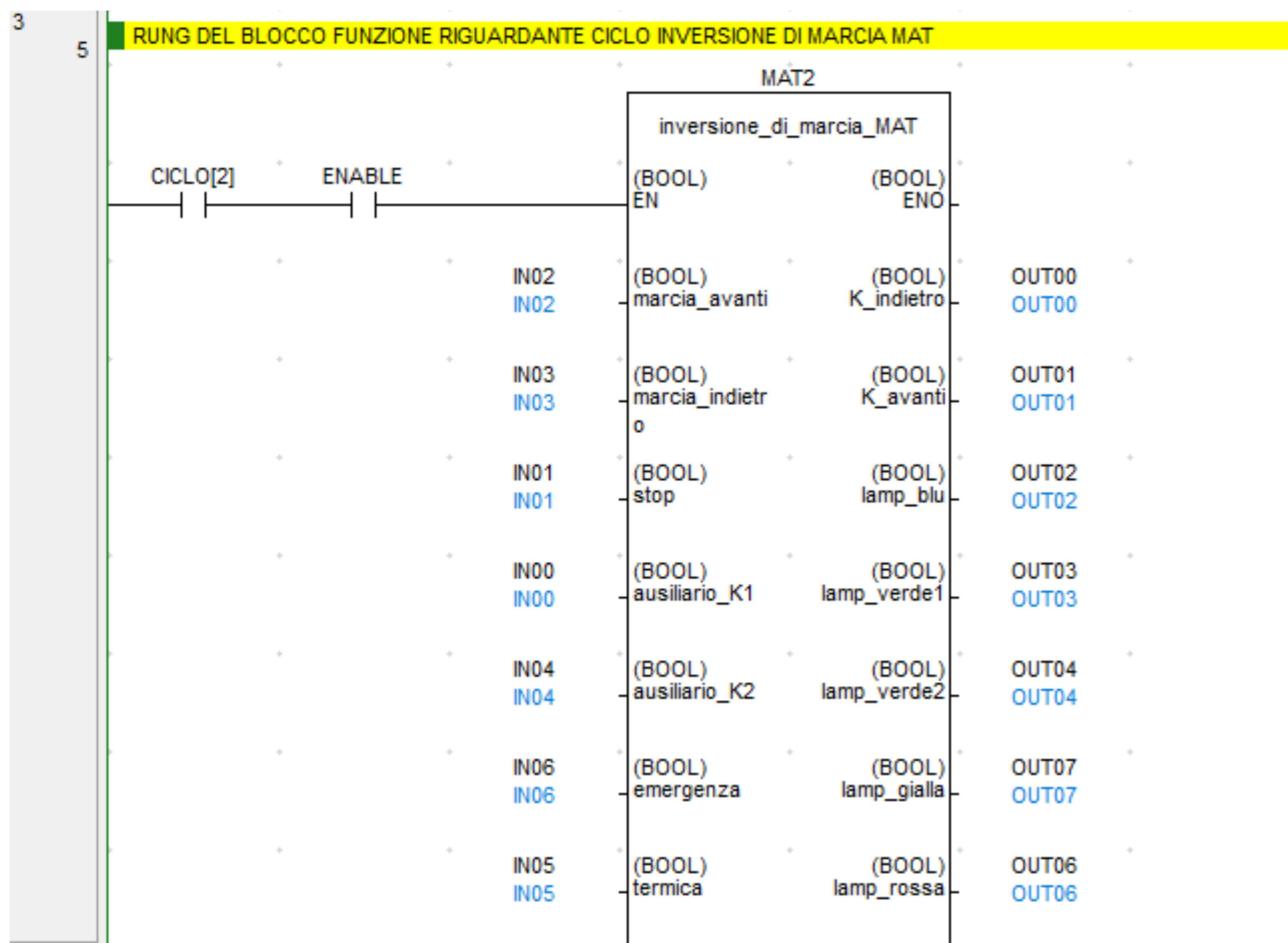


Inversione di marcia manuale di un motore (MAT)

Descrizione

Il processo consiste nell'avviare un motore (MAT) e poi invertire il senso di rotazione.

Blocco



Variabili

Simbolo	Indirizzo	Commento
IN00	Ausiliariook1	Contatto (NO) ausiliario del contattore KM1
INN01	stop	Pulsante (NC) di Stop
IN02	Marcia avanti	Pulsante (NO) marcia avanti
IN03	Marcia indietro	Pulsante (NO) marcia indietro
IN04	Ausiliariook1	Contatto (NO) ausiliario del contattore KM2
IN05	termica	Contatto (NC) relè termico
IN06	emergenza	Pulsante (NC) simulatore emergenza esterna
OUT00	K avanti	Contattore KM1 marcia avanti
OUT01	K indietro	Contattore KM2 marcia indietro
OUT02	Lamp blu	Lampada segnalazione blu
OUT03	Lamp verde1	Lampada segnalazione verde
OUT04	Lamp verde2	Lampada segnalazione verde
OUT07	Lamp gialla	Lampada segnalazione gialla
OUT06	Lamp rossa	Lampada segnalazione rossa

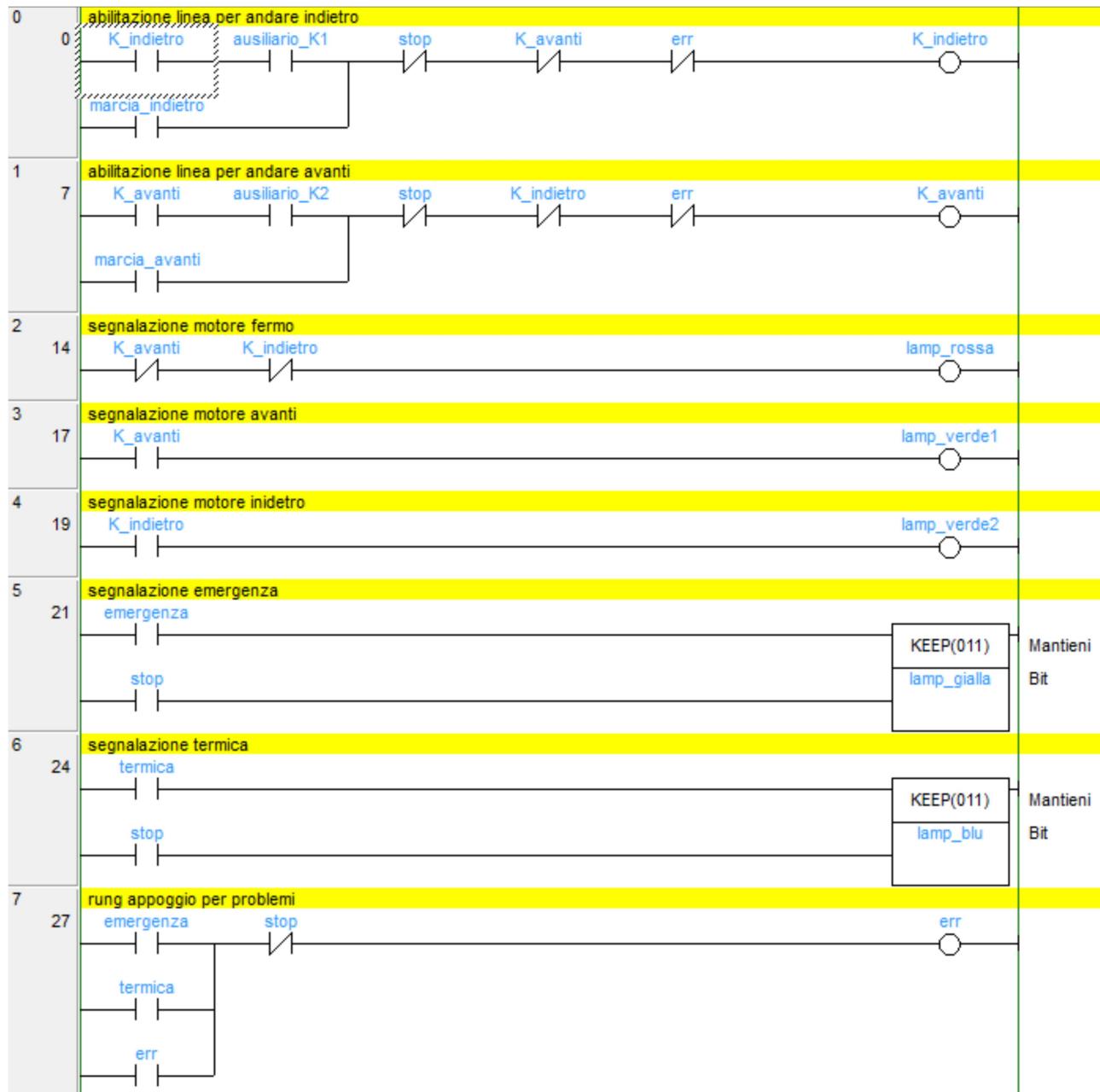


Schema ladder del blocco

Tramite l'utilizzo di due pulsanti è possibile decidere il senso di marcia del motore che lo manterrà finché non viene premuto il pulsante di stop.

Con INO2 il motore è in marcia avanti mentre con INO3 il motore è in marcia indietro.

Inoltre il motore può essere arrestato tramite l'inserimento di una termica o di un'emergenza che vengono disinserite dal pulsante di stop per mantenere lo stato si usa una variabile appoggio err.



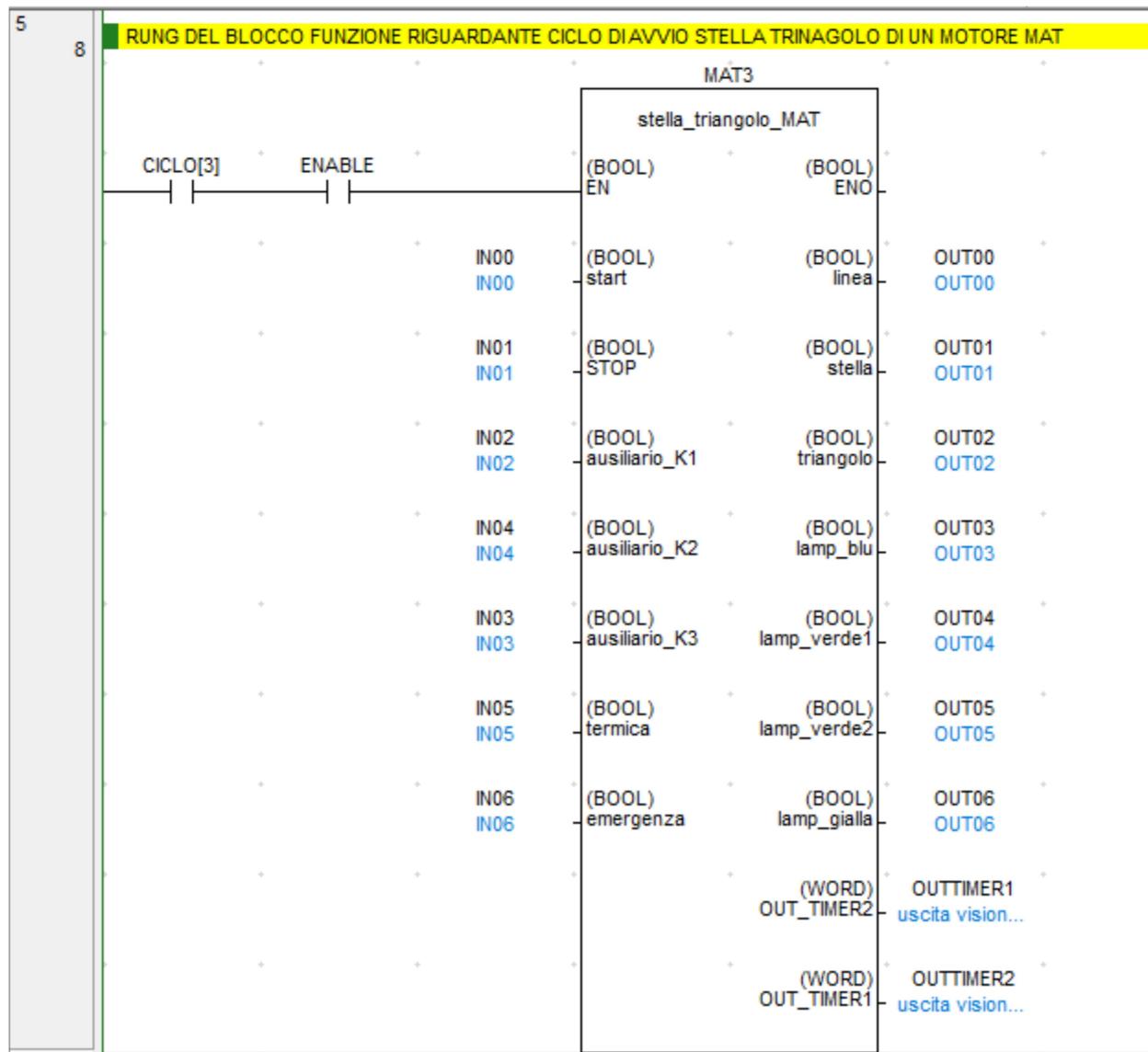


Avviamento stella-triangolo di un motore MAT

Descrizione

Il processo consiste nell'avviare un motore MAT prima a stella e successivamente a triangolo.

Blocco





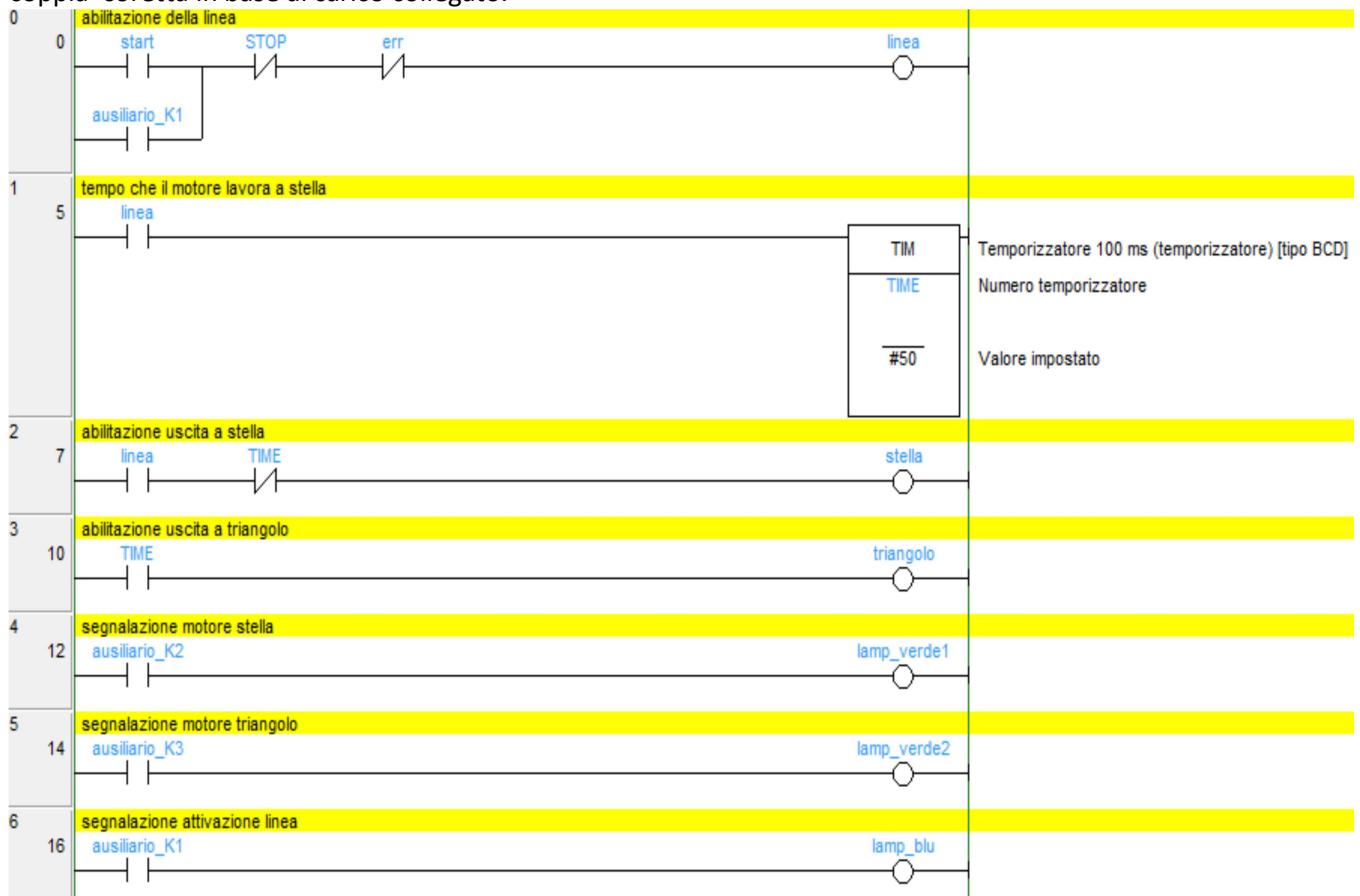
Variabili

Simbolo	Indirizzo	Commento
IN00	Stop	Pulsante (NC) Stop
IN01	Start	Pulsante (NO) Start
IN02	Ausiliari K1	Contatto (NO) ausiliario contattore linea KM1
IN03	Ausiliari K3	Contatto (NO) ausiliario contattore triangolo KM3
IN04	Ausiliari K2	Contatto (NO) ausiliario contattore stella KM2
IN05	Termica	Contatto (NC) relè termico
IN06	Emergenza	Pulsante (NC) simulatore emergenza esterna
OUT00	Linea	Contattore di linea
OUT01	Stella	Contattore stella
OUT02	triangolo	Contattore triangolo
OUT03	Lamp blu	Lampada segnalazione blu
OUT04	Lamp verde1	Lampada segnalazione verde
OUT05	Lamp verde2	Lampada segnalazione verde
OUT06	Lamp gialla	Lampada segnalazione gialla

Schema ladder del blocco

Una volta che la linea è attiva il motore parte a stella e dopo un tempo definito dal temporizzatore TIME passa ad un collegamento a triangolo.

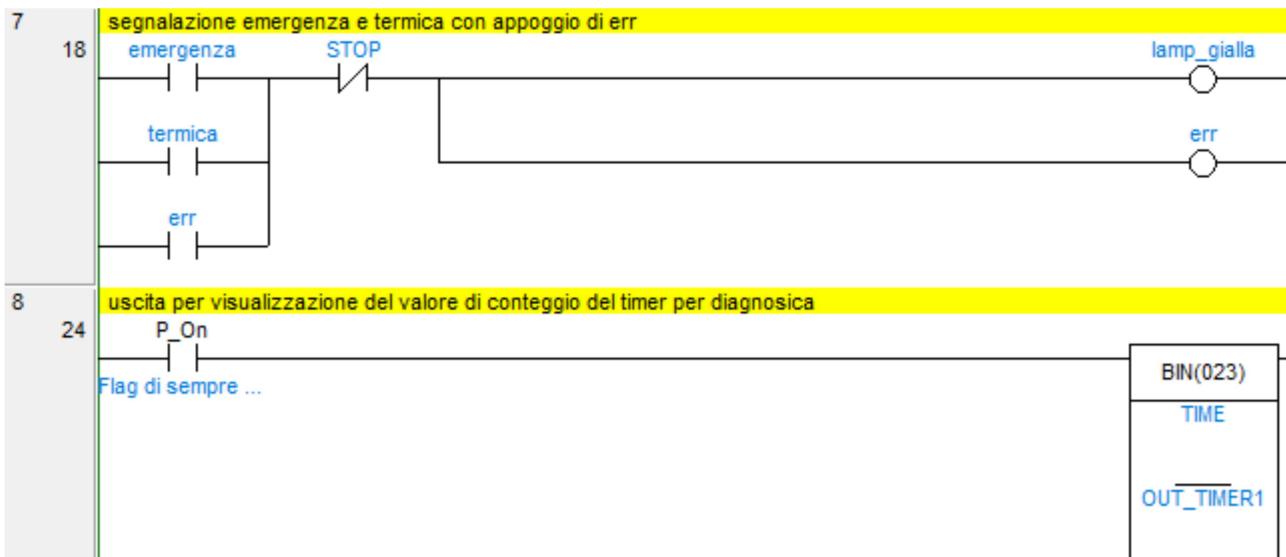
Questo ciclo viene utilizzato per avviare un motore trifase con una corrente di spunto minore ma con una coppia elevata poi nel passaggio a triangolo la corrente assorbita resta uguale e fornisce in uscita una coppia coretta in base al carico collegato.





Istituto di Istruzione Superiore “Benedetto Castelli”

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale
Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it



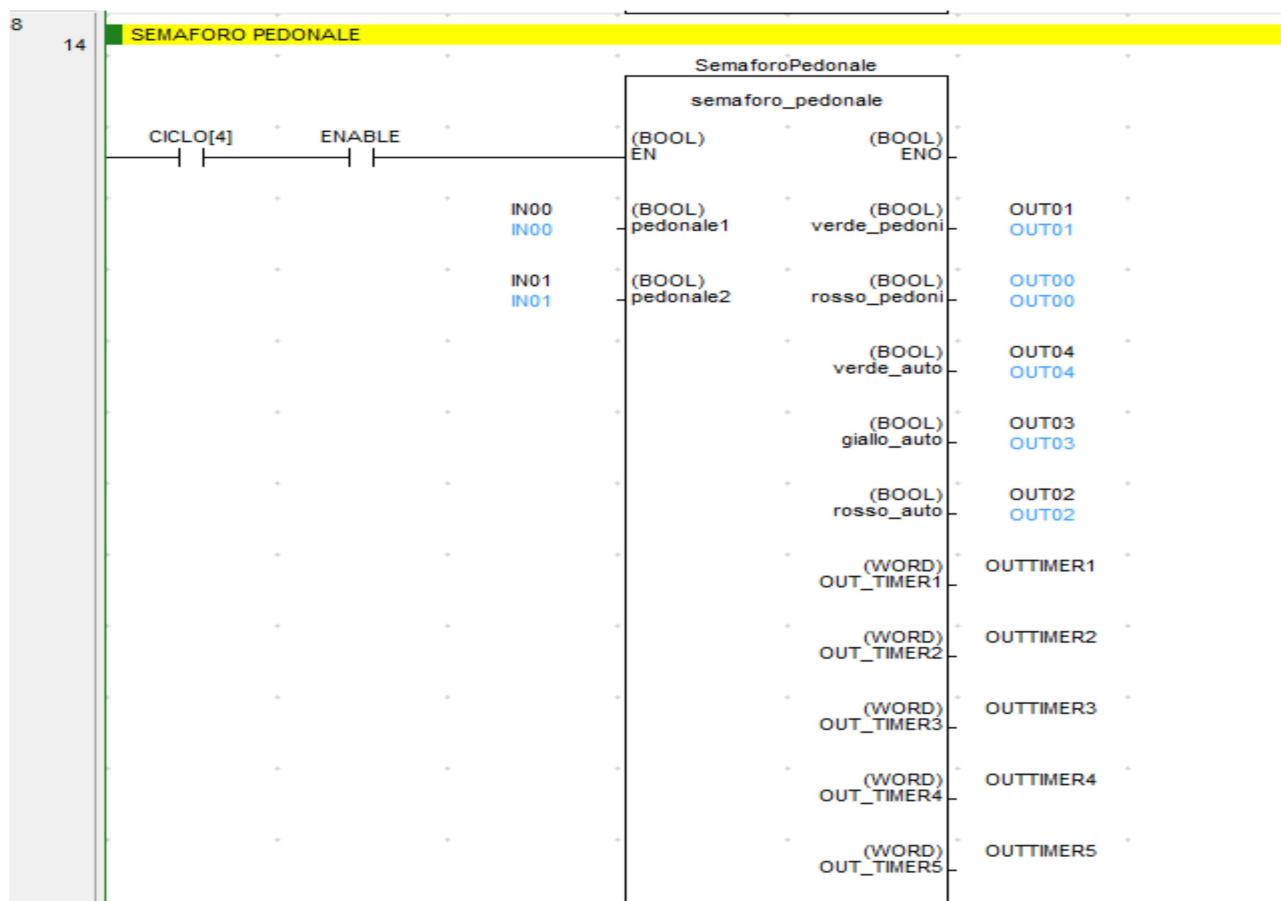


Semaforo pedonale

Descrizione

Il processo serve a simulare un passaggio pedonale su una strada. Quando un pedone preme il pulsante per attraversare parte la sequenza fermando le automobili e poi dando il verde ai pedoni. Dopo un tempo prestabilito ai pedoni tornerà il rosso e alle automobili tornerà il verde.

Blocco



Variabili

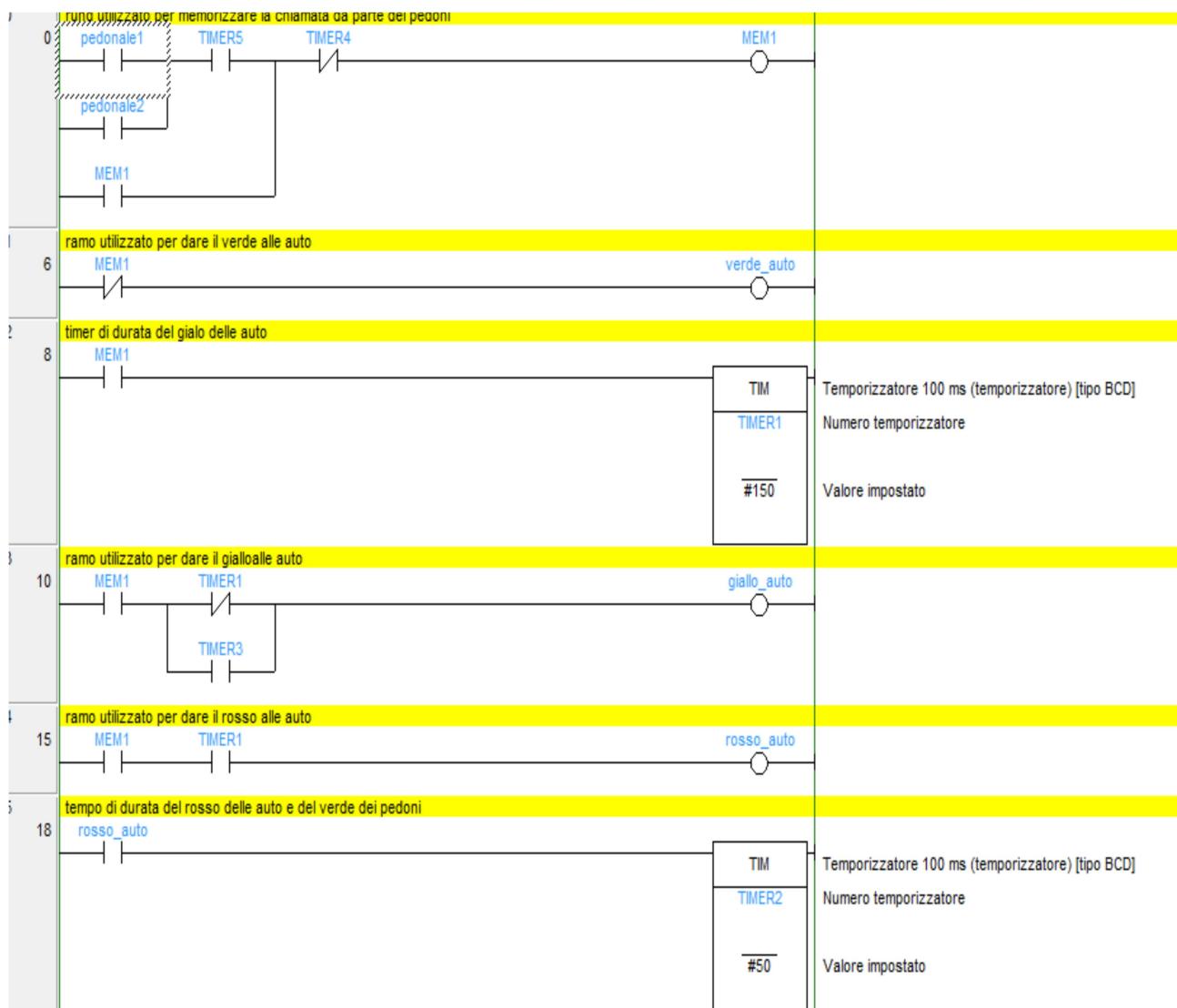
Simbolo	Indirizzo	Commento
IN00	Pedonale1	Pulsante (NO) prenotazione pedone
IN01	Pedonale2	Pulsante (NO) prenotazione pedone
OUT00	Rosso pedoni	Rosso pedoni
OUT01	Verde pedoni	Verde pedoni
OUT02	Rosso auto	Rosso autoveicoli
OUT03	Giallo auto	Giallo autoveicoli
OUT04	Vere auto	Verde autoveicoli



Schema ladder del blocco

Nel blocco si sviluppa una sequenza che in base allo stato dei vari timer va ad attivare le uscite delle lampade del semaforo.

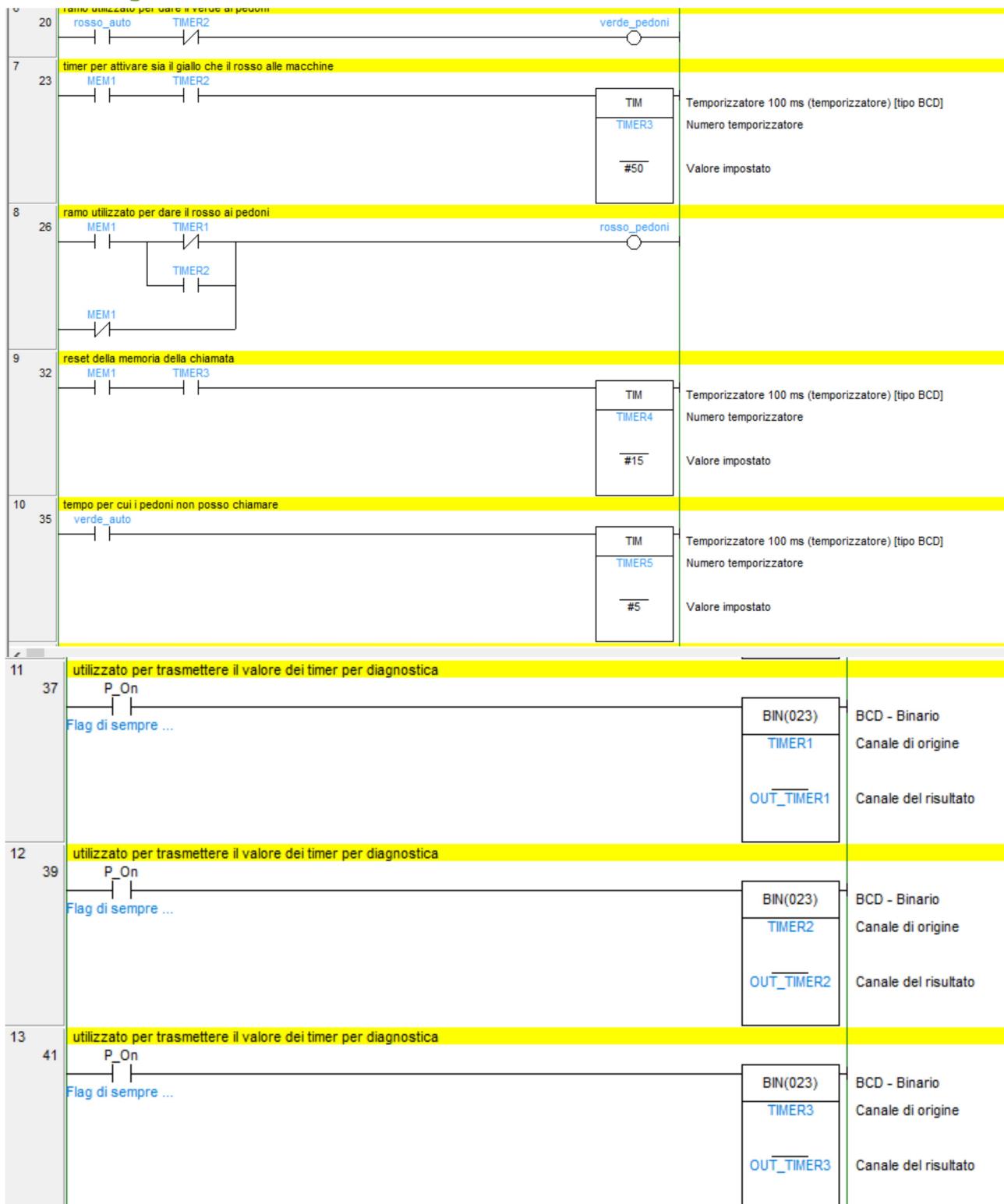
Il ciclo si attiva dopo che viene premuto uno dei due pulsanti del pedonale





Istituto di Istruzione Superiore "Benedetto Castelli"

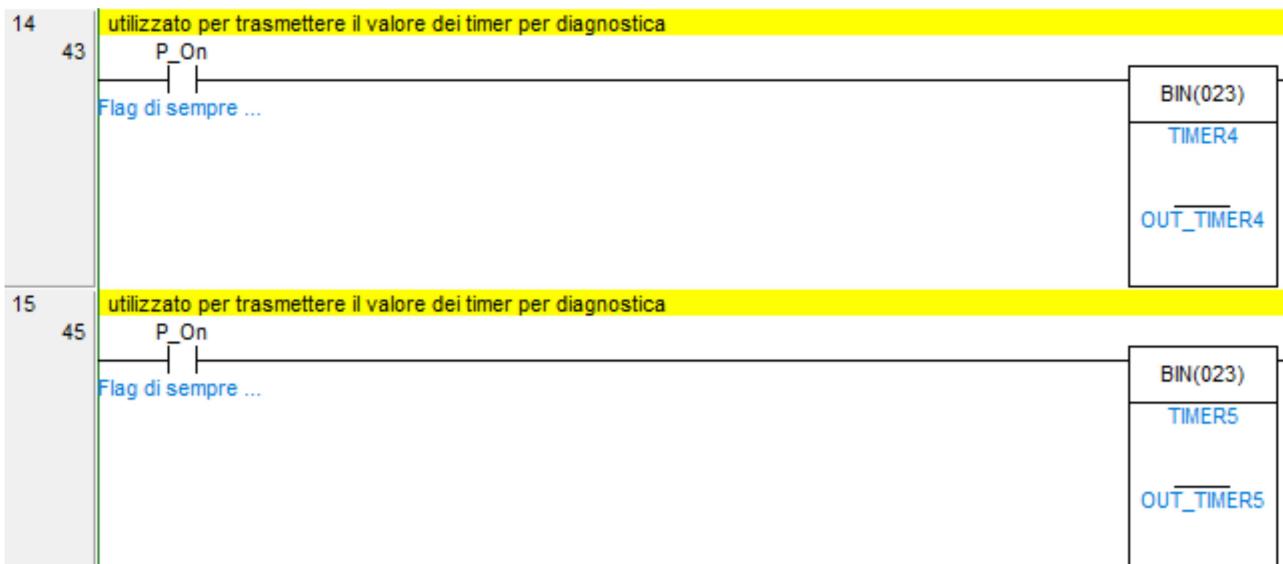
Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale
Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it





Istituto di Istruzione Superiore “Benedetto Castelli”

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale
Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it





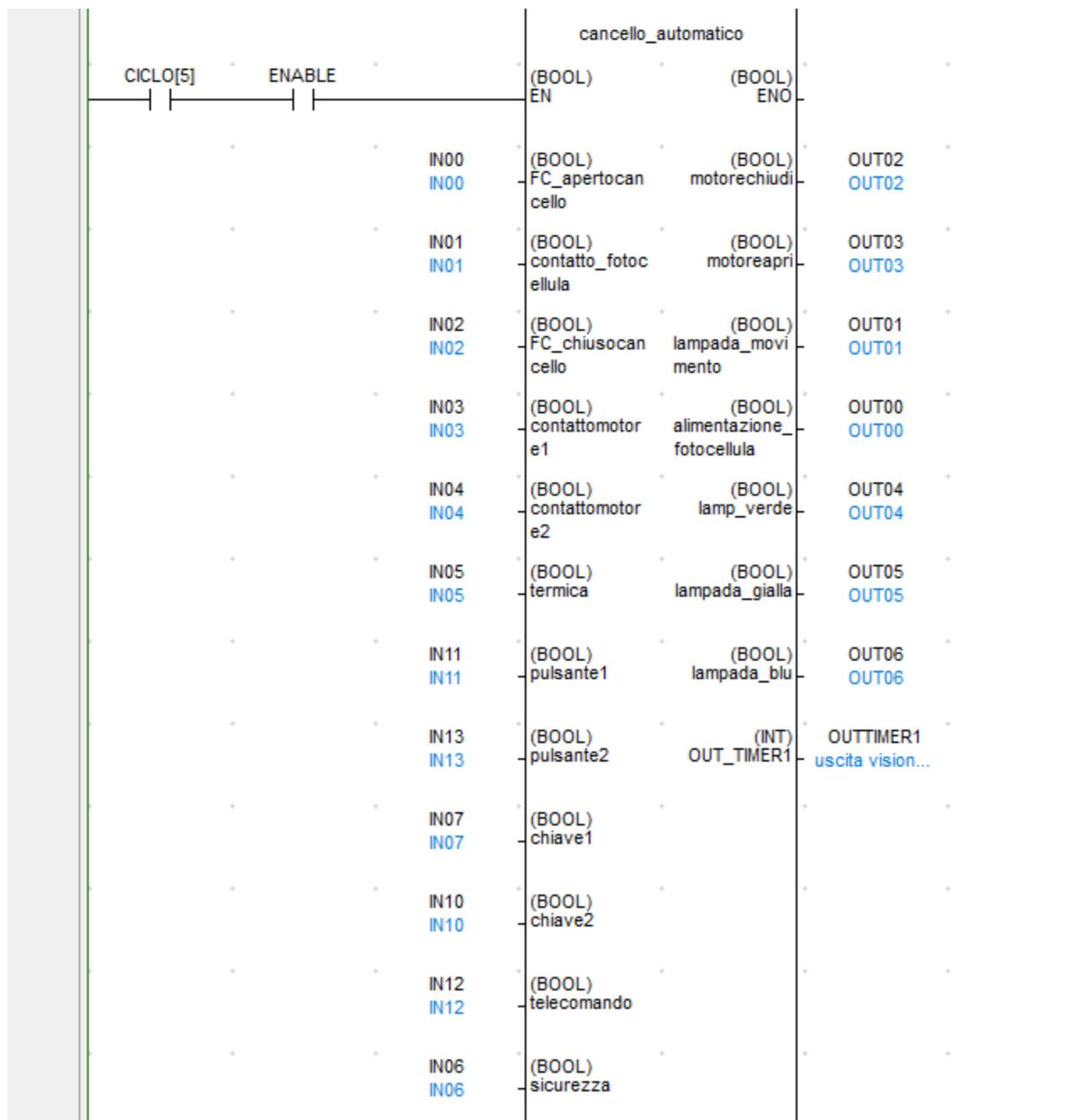
Cancello automatico

Descrizione

Il processo serve per simulare un cancello automatico che si attiva mediante chiave o un controllo manuale o mediante un telecomando che permette di svolgere un ciclo automatico .

Inoltre è presente una fotocellula che durante la chiusura se attivata riapre il cancello.

Blocco





Variabili

Simbolo	Indirizzo	Commento
IN00	FC apertocancello	Contatto (NC) fine corsa apertura cancello
IN01	Contato fotocellula	Contatto (NO) fotocellula B
IN02	FC chiusocancello	Contatto (NC) fine corsa chiusura cancello
IN03	Contattoremotore1	Contatto (NO) ausiliario KM1
IN04	Contattoremotore2	Contatto (NO) ausiliario KM2
IN05	termica	Contatto (NC) relè termico
IN06	sicurezza	Contatto (NC) costa pneumatica
IN07	Chiavel1	Contatto (NO) selettore chiave posizione 1
IN10	Chiave2	Contatto(NO) selettore chiave posizione 2
IN11	pulsante	Pulsante (NO)
IN12	telecomando	Pulsante (NO) telecomando
IN13	Pulsantel	Pulsante (NC)
OUT00	Alimentazione fotocellula	Alimentazione fotocellula
OUT01	Lampada movimento	Lampada segnalazione movimento
OUT02	motorechiudi	Contattore chiusura
OUT03	motoreapri	Contattore apertura
OUT04	Lampada verde	Lampada segnalazione verde
OUT05	Lampada gialla	Lampada segnalazione gialla
OUT06	Lampada blu	Lampada segnalazione blu

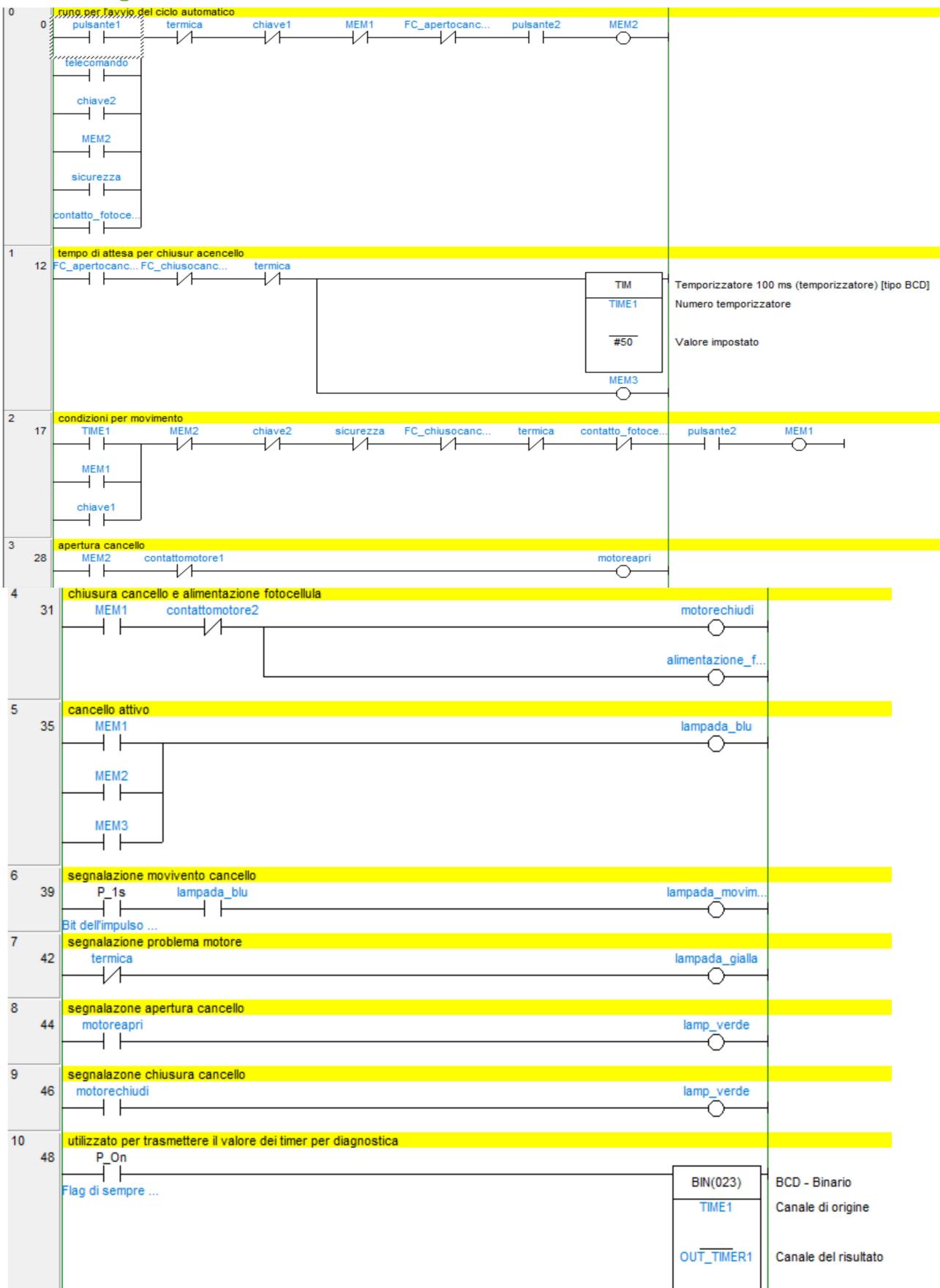
Schema ladder del blocco

Il ciclo del cancello si attiva tramite un telecomando / selettore a chiave / pulsante.

Il cancello si apre con la rotazione del motore, il finecorsa lo ferma e attiva il timer di pausa ciclo.

Trascorso il tempo prefissato si attiva la chiusura del cancello che può essere interrotta nel caso venga attivata la fotocellula che fa riaprire il cancello.

Mentre il cancello è in movimento c'è la lampada di segnalazione gialla lampeggiante.





Garage automatico

Descrizione

Serve per la simulazione di un garage automatico a sei posti auto.

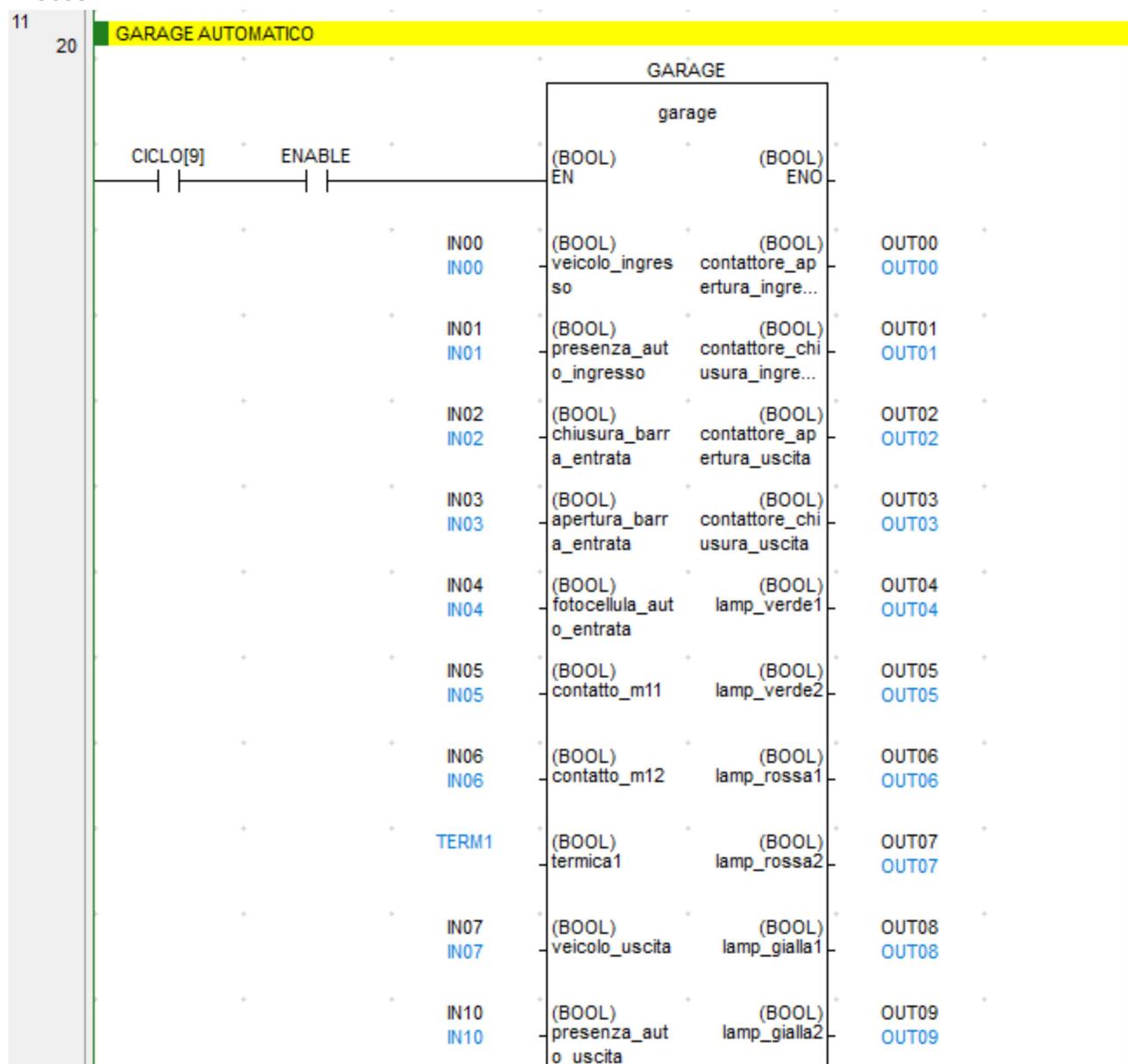
Nel garage, l'entrata e l'uscita sono regolate da due sbarre; entrambe vengono chiuse e aperte tramite un motore (rispettivamente M1 e M2); ogni motore è azionato da due contattori che ne permettono l'inversione di marcia. Per ogni contattore è disponibile anche un contatto ausiliario.

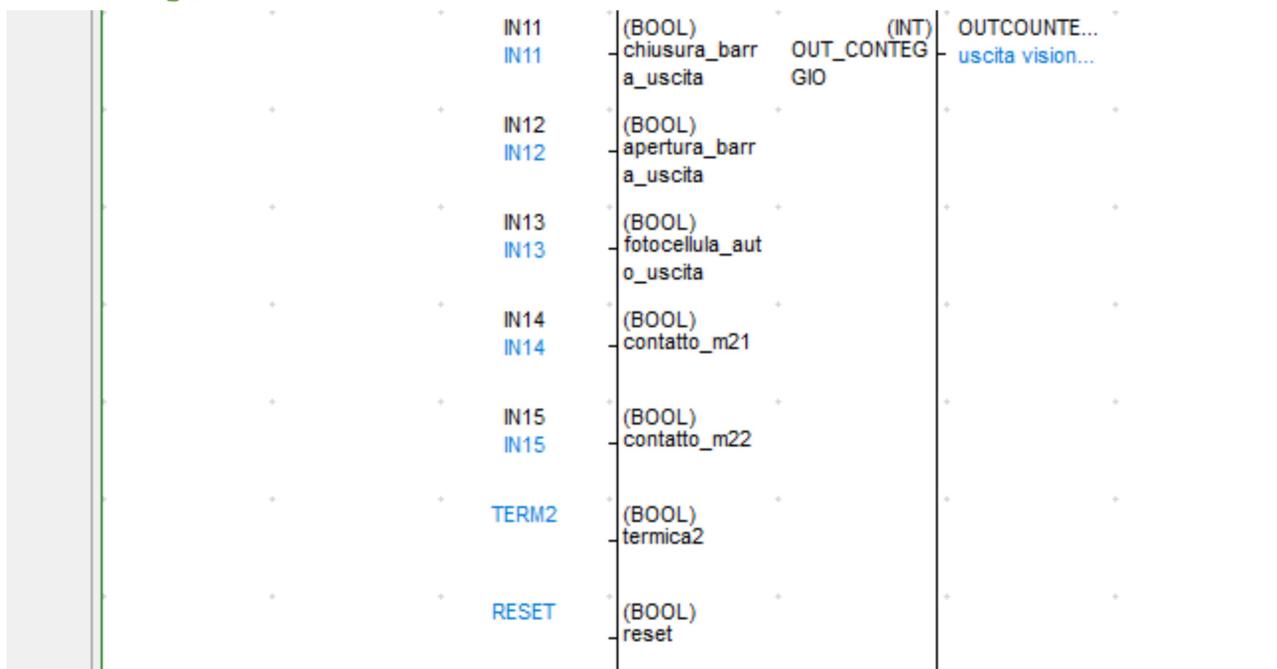
Dopo ogni sbarra è installata una fotocellula (entrata B1, uscita B2) per rilevare l'avvenuto passaggio del veicolo e che serve per controllare il conteggio dei posti disponibili.

Il movimento di ogni sbarra viene segnalato con il lampeggio delle lampade

Nel garage, il semaforo posto in ingresso sarà verde, in presenza di posti liberi; viceversa passerà al rosso quando il garage è pieno. Il semaforo d'uscita, normalmente rosso, passerà al verde quando la sbarra è aperta per segnalare al veicolo il consenso all'uscita e ripasserà poi al rosso quando il veicolo avrà superato la fotocellula B2.

La predisposizione di un veicolo in entrata viene eseguita azionando con un clic del mouse il pulsante “Nuovo Ingresso”; scegliendo poi il posto a cui è destinato mediante i pulsanti entrata; la predisposizione di uno dei veicoli per l'uscita viene attuata con un clic del pulsante uscita.

**Blocco**



Variabili

Simbolo	Indirizzo	Commento
IN00	veicolo ingresso	Pulsante (NO) richiesta biglietto entrata
IN01	presenza_auto_ingresso	Sensore (NO) magnetico presenza auto all'ingresso
IN02	chiusura_barra_entrata	Fine corsa (NC) chiusura sbarra ingresso
IN03	apertura_barra_entrata	Fine corsa (NC) apertura sbarra ingresso
IN04	fotocellula_auto_entrata	Fotocellula (NO) entrata auto terminata
IN05	contatto_m11	Contatto ausiliario (NO) contattore KM1
IN06	contatto_m12	Contatto ausiliario (NO) contattore KM2
TERM1	termical1	Contatto (NC) relè termico motore M1
IN07	veicolo_uscita	Pulsante (NO) pagamento effettuato in uscita
IN10	presenza_auto_uscita	Sensore (NO) magnetico presenza auto all'uscita
IN11	chiusura_barra_uscita	Fine corsa (NC) chiusura sbarra uscita
IN12	apertura_barra_uscita	Fine corsa (NC) apertura sbarra uscita
IN13	fotocellula_auto_uscita	Fotocellula (NO) uscita auto terminata
IN14	contatto_m21	Contatto ausiliario (NO) contattore KM4
IN15	contatto_m22	Contatto ausiliario (NO) contattore KM3
TERM2	termica2	Contatto (NC) relè termico motore M2
OUT00	contattore_apertura_ingresso	Contattore apertura sbarra ingresso
OUT01	contattore_chiusura_ingresso	Contattore chiusura sbarra ingresso
OUT02	contattore_apertura_uscita	Contattore chiusura sbarra uscita
OUT03	contattore_chiusura_uscita	Contattore apertura sbarra uscita
OUT04	lamp_verde1	Lampada Verde semaforo ingresso
OUT05	lamp_verde2	Lampada verde semaforo uscita
OUT06	lamp_rossa1	Lampada rossa semaforo ingresso
OUT07	lamp_rossa2	Lampada Rossa semaforo uscita
OUT08	lamp_gialla1	Lampeggiatore sbarra ingresso in movimento
OUT09	lamp_gialla2	Lampeggiatore sbarra uscita in movimento

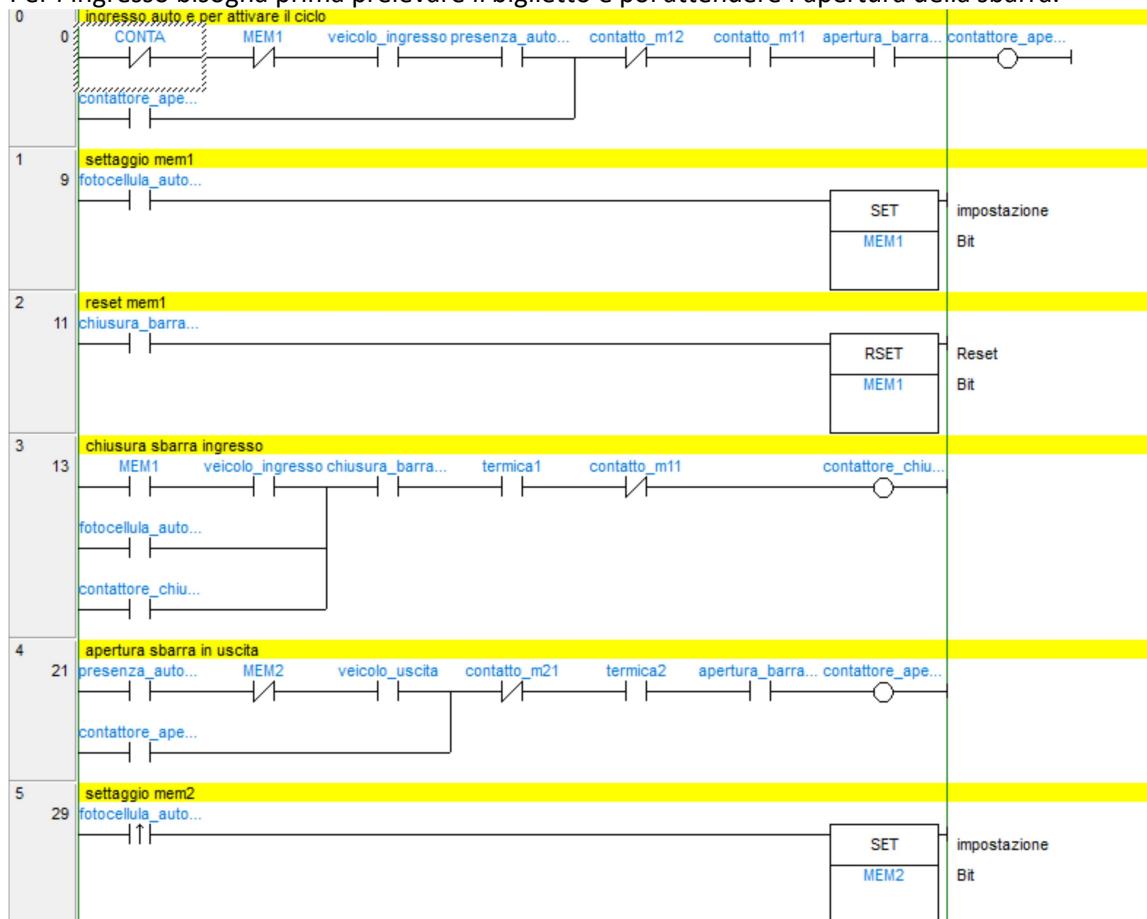


Schema ladder del blocco

Il processo conta il numero di macchine in ingresso al garage che viene gestito da un contatore up/down una volta che è arrivato al valore impostato blocca l'ingresso dei veicoli finché il contatore non verrà decrementato.

Prima dell'uscita della macchina si deve effettuare il pagamento che consente l'apertura della sbarra.

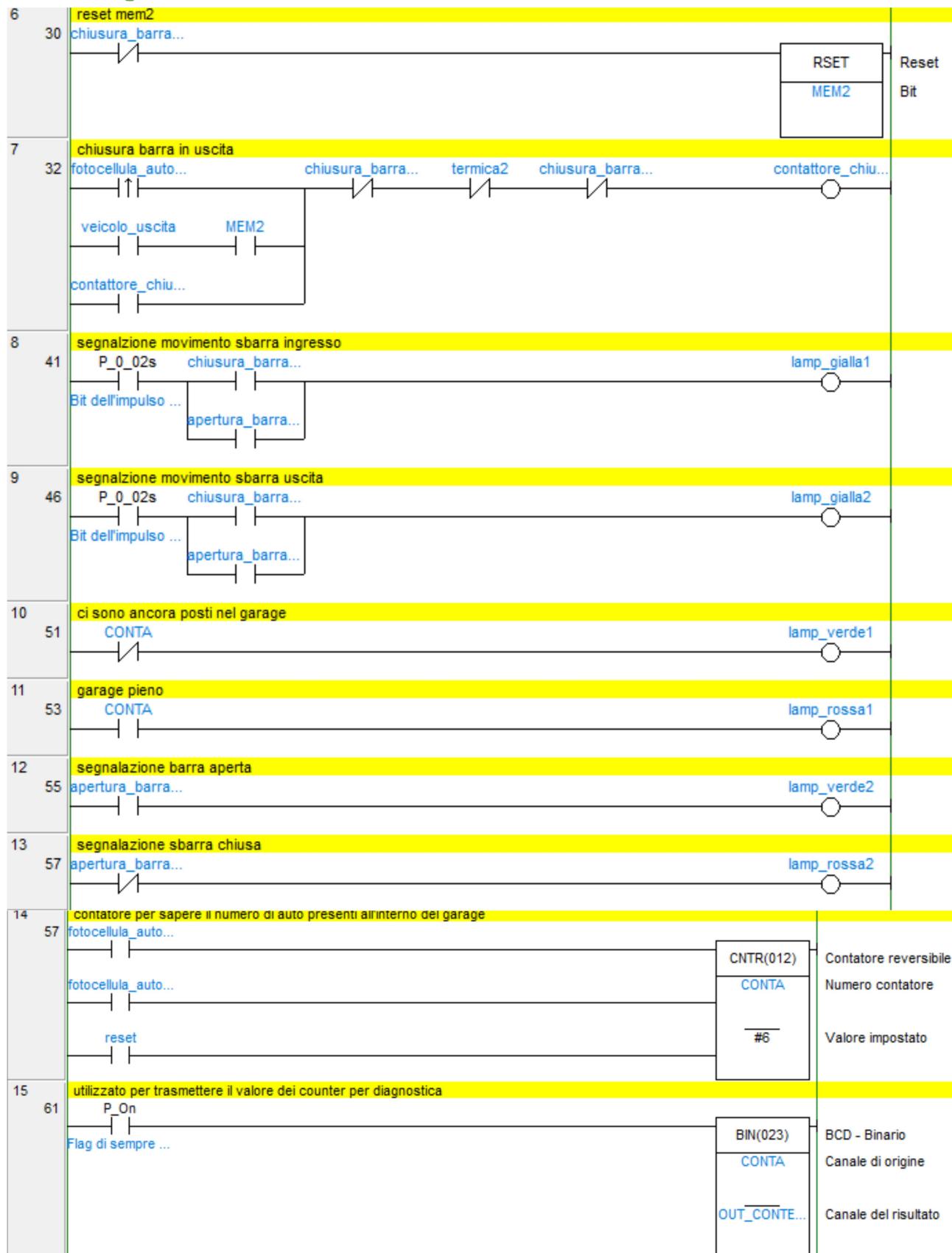
Per l'ingresso bisogna prima prelevare il biglietto e poi attendere l'apertura della sbarra.





Istituto di Istruzione Superiore "Benedetto Castelli"

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale
Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it



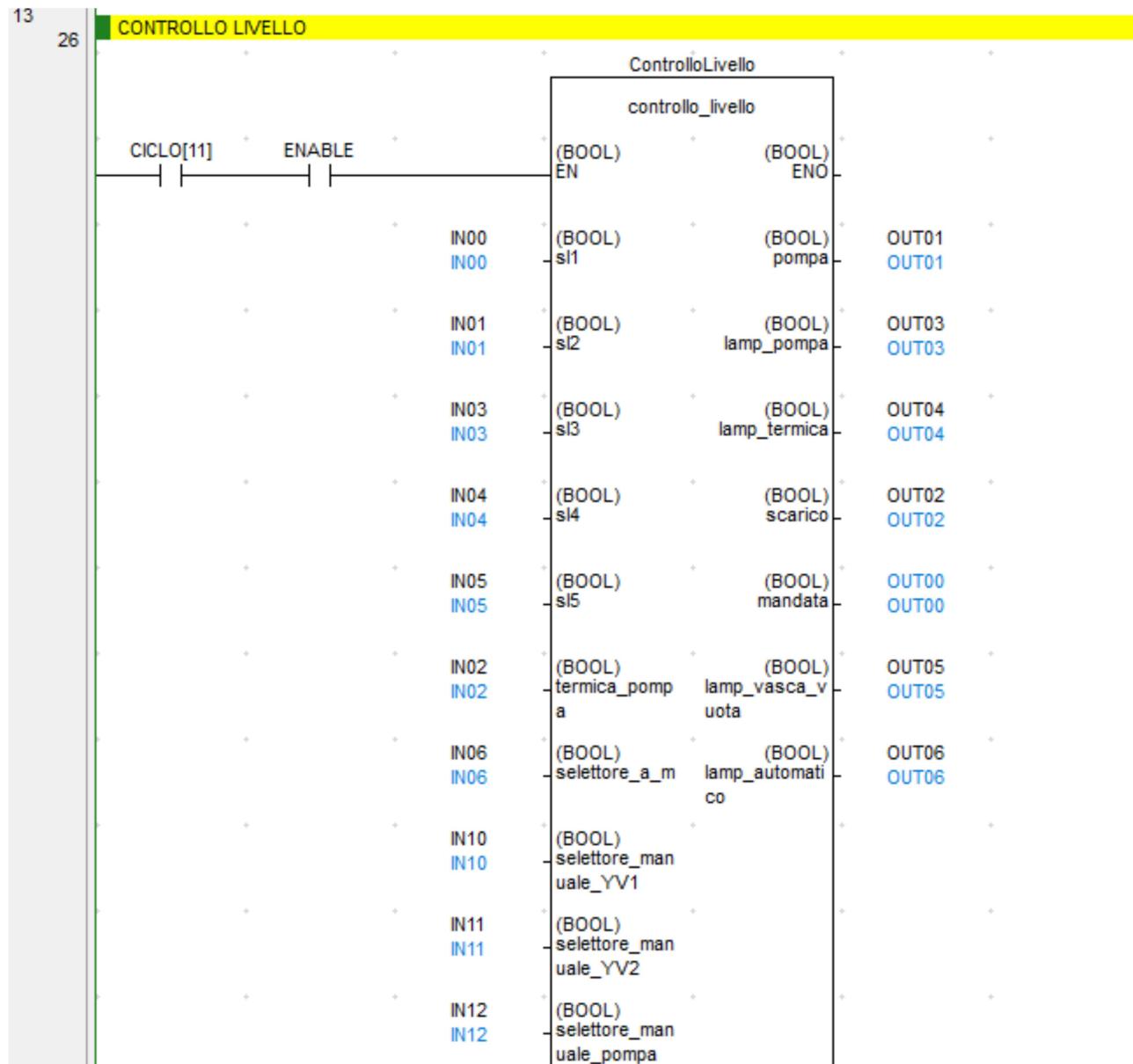


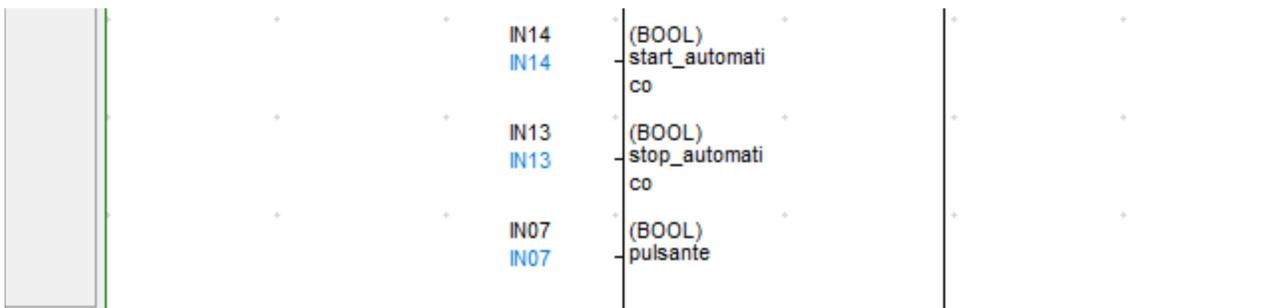
Controllo livello di un serbatoio

Descrizione

Il controllo di livello serve per simulare il corretto funzionamento di un serbatoio nel mantenere costante il livello prefissato.

Blocco





Variabili

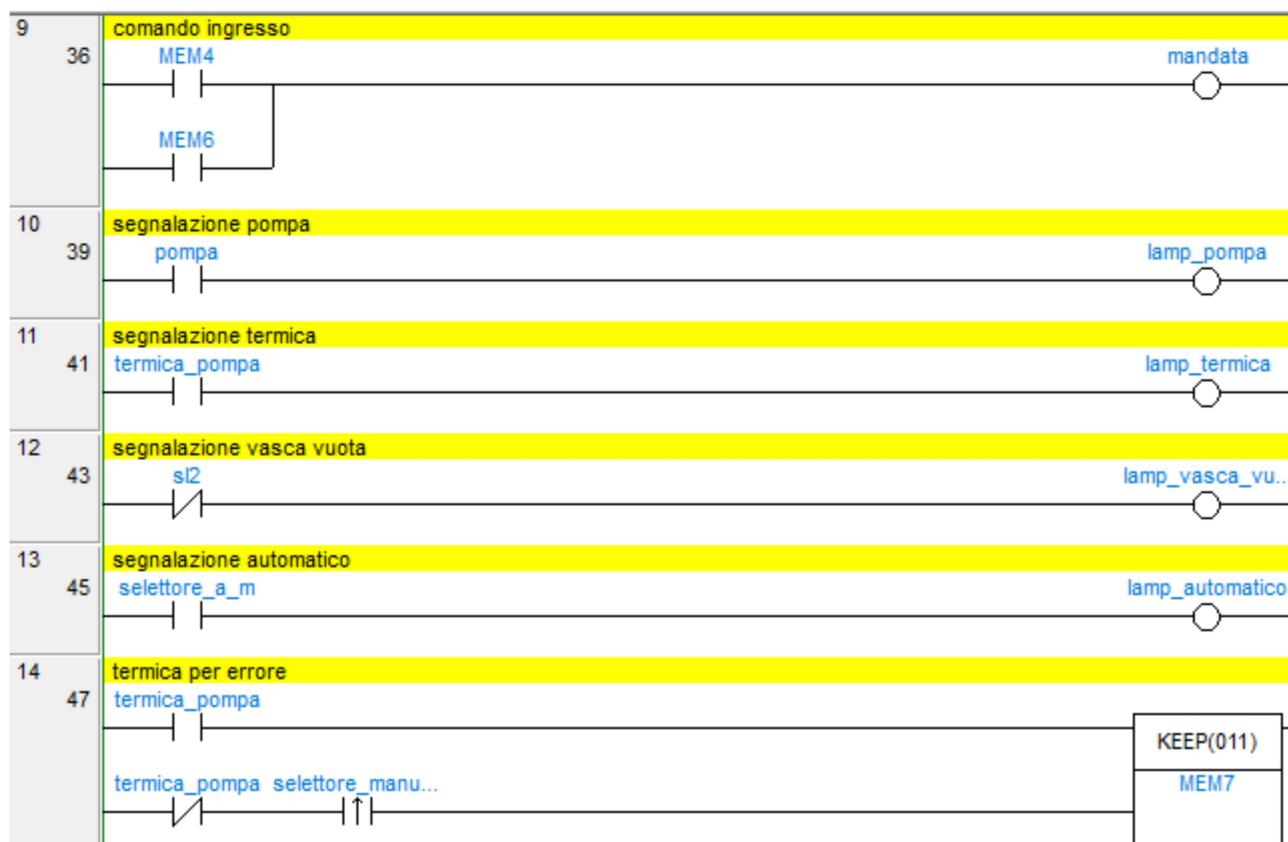
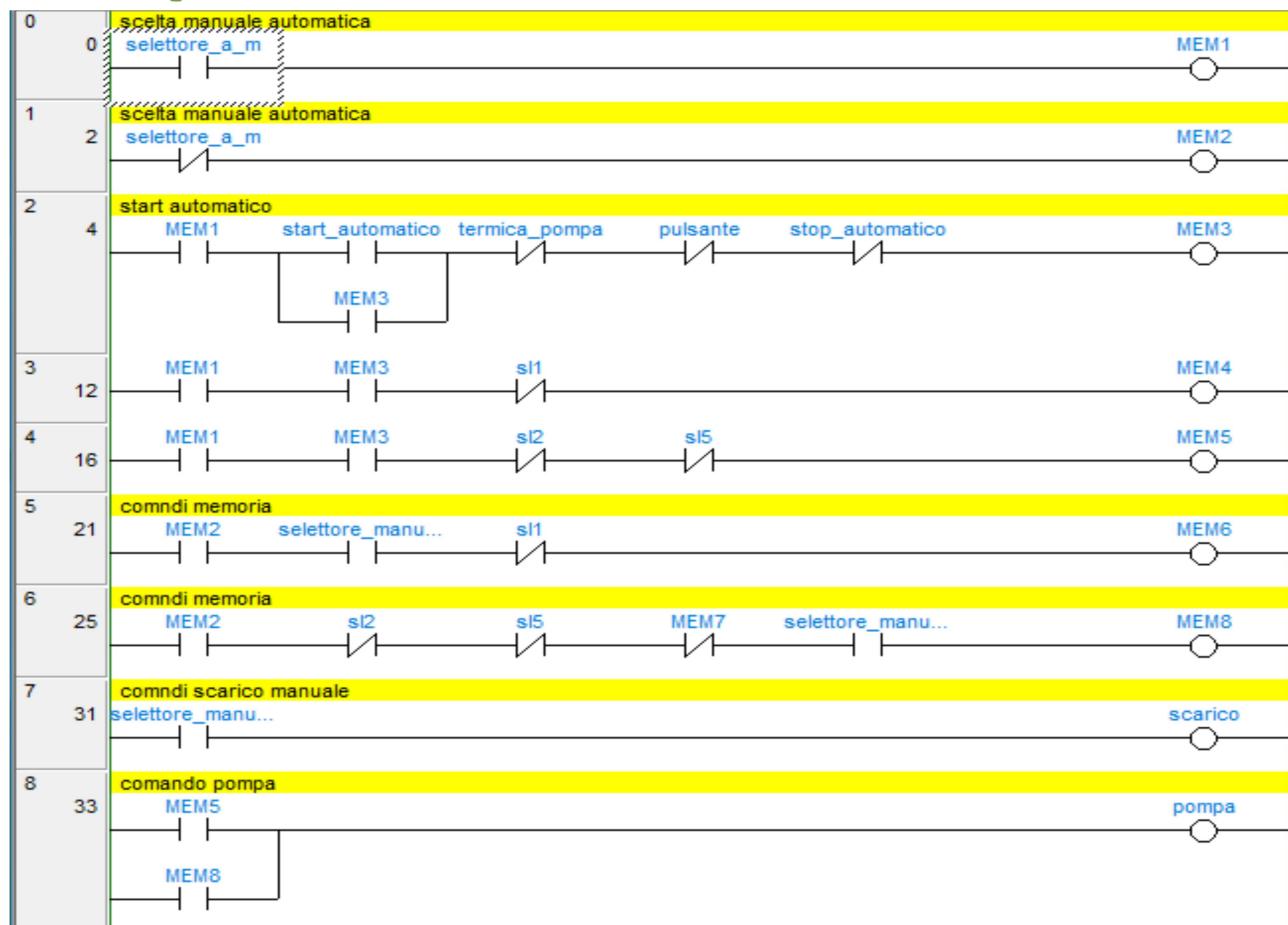
Simbolo	Indirizzo	Commento
IN00	S11	Sensore (NC) livello Max vasca di carico
IN01	S12	Sensore (NC) livello Min vasca di carico
IN02	Termica_pompa	Contatto (NC) relè termico motore pompa P1
IN03	S13	Sensore (NC) livello Min serbatoio
IN04	S14	Sensore (NC) livello 1/2 serbatoio
IN05	S15	Sensore (NC) livello Max serbatoio
IN06	Selettore_a_m	Selettore (NO) Automatico/Manuale
IN07	Pulsante	Pulsante (NC) simulazione emergenza esterna
IN10	Selettore_manuale_YV1	Selettore (NO) comando manuale elettrovalvola YV1
IN11	Selettore_manuale_YV2	Selettore (NO) comando manuale elettrovalvola YV2
IN12	Selettore_manuale_pompa	Selettore (NO) comando manuale pompa P1
IN13	StopAutomatico	Pulsante (NC)
IN14	StartAutomatico	Pulsante (NO)
OUT00	Mandata	Elettrovalvola vasca di carico
OUT01	Pompa	Contattore motore KM1
OUT02	Scarico	Elettrovalvola scarico serbatoio
OUT03	Lamp_pompa	Lampada segnalazione verde
OUT04	Lamp_termica	Lampada segnalazione gialla
OUT05	Lamp_vasca_scarico	Lampada segnalazione rossa
OUT06	Lamp_automatico	Lampada segnalazione blu

Schema ladder del blocco

Si può decidere attraverso un selettore se attivare il ciclo automatico o manuale e attraverso un altro selettore se scaricare il serbatoio oppure no.

Per sapere il livello della vasca di carico si usano due sensori per il livello massimo (IN00) e per il minimo (IN01) mentre per il serbatoio principale abbiamo tre sensori che determinano i livelli di riempimento del serbatoio (IN03-IN04-IN05) min,medio,max.

Nel caso si attivi la protezione termica o l'emergenza, il sistema si ferma e solo dopo aver premuto il pulsante di stop con un successivo start riparte.





Miscelatore di due liquidi

Descrizione

Il funzionamento di questo processo serve per monitorare il mescolamento di due liquidi diversi.

Attraverso il selettore "IN05" si sceglie la modalità di funzionamento del processo.

Processo automatico:

Le percentuali dei due liquidi sono preimpostate all'inizio del ciclo su cx_programmer, con la pressione del pulsante start si avvia il riempimento del serbatoio e raggiunto il livello impostato si attiva la miscelazione e riscaldamento del composto.

Manualmente premendo il pulsante di scarico si svuota il serbatoio fino al raggiungimento del livello minimo controllato attraverso il sensore "MIN".

Processo manuale:

L'ingresso del primo liquido è gestito tramite un pulsante "IN10" che attiva l'apertura dell'elettrovalvola "OUT00".

L'ingresso del secondo liquido si deve premere il pulsante "IN12" che attiva l'apertura dell'elettrovalvola "OUT03".

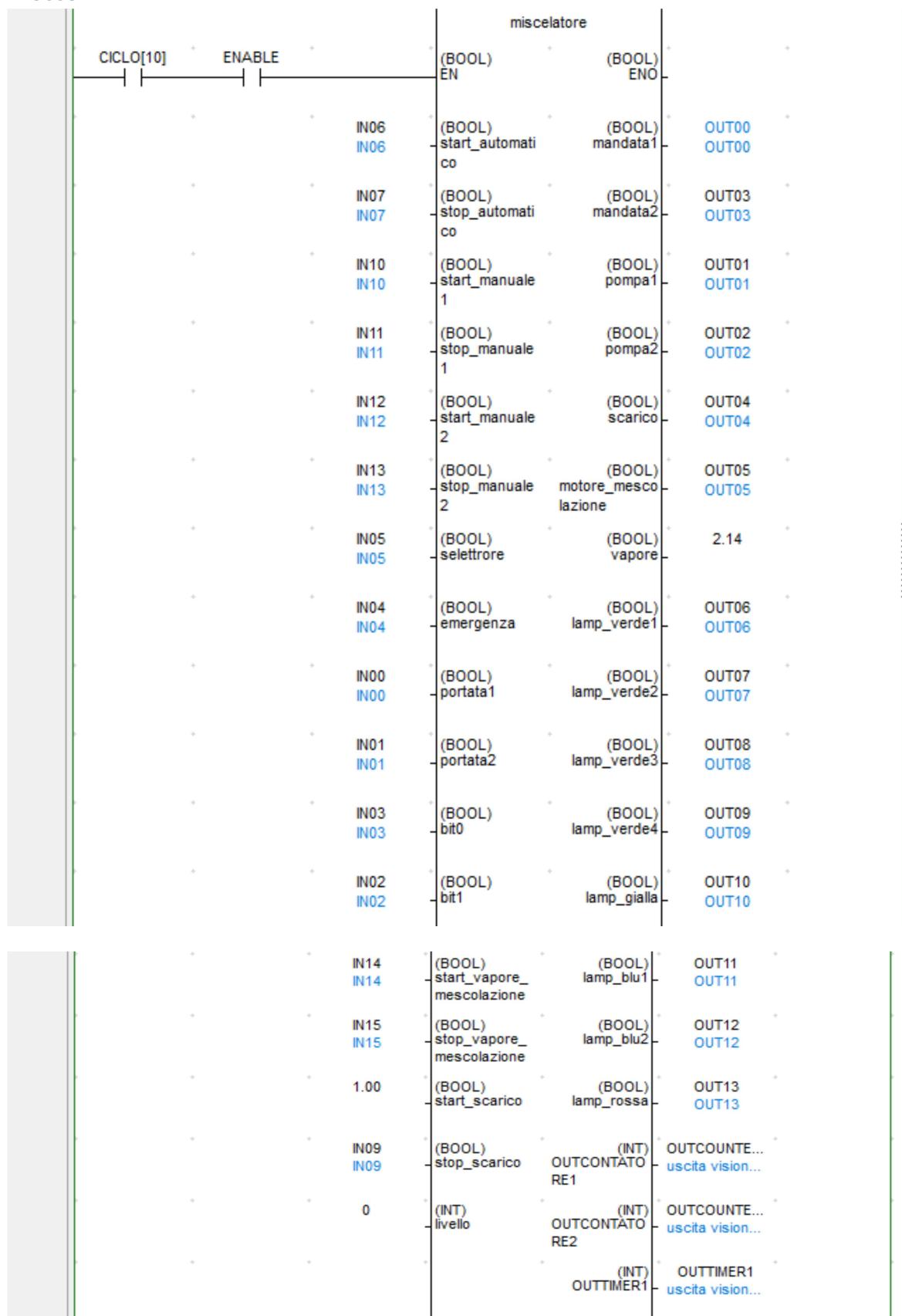
Mentre per lo scarico si usano rispettivamente i pulsanti "IN11-IN13" che disattivano "OUT00-OUT03".

Nel caso non si premano i pulsanti di stop, al raggiungimento del livello massimo, le elettrovalvole di carico si chiudono automaticamente perché si controlla il livello massimo tramite il sensore MAX.

Una volta riempito il serbatoio si può far iniziale la miscelazione e il riscaldamento del composto.

Per quanto riguarda lo scarico lo si può svolgere mentre è ancora attiva la miscelazione oppure disattivando il vapore e il motore del mescolatore e procedendo allo scarico attraverso il pulsante "IN16", per arrestarlo si deve premere il pulsante "IN09".

Con il pulsante di emergenza "IN04" si simulano le possibili cause accidentali di malfunzionamento, l'intero processo si blocca istantaneamente e si può riprendere dopo il ripristino del sistema.

**Blocco**

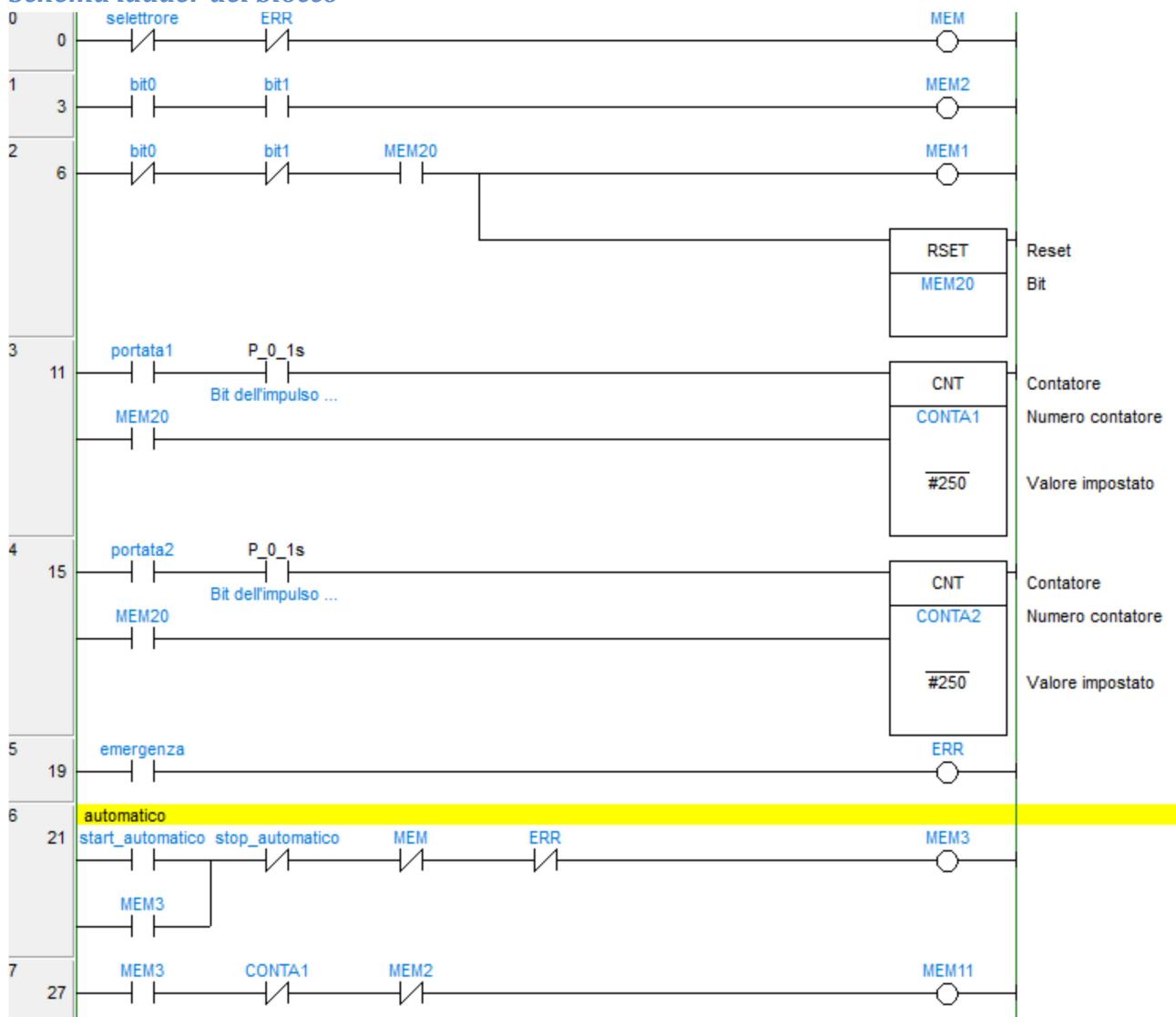


Variabili

Simbolo	Indirizzo	Commento
IN00	Portata1	Contatto (NO) trasduttore digitale di portata Prod. 1
IN01	Portata2	Contatto (NO) trasduttore digitale di portata Prod. 2
IN02	Bit1	Bit '1' trasduttore livello serbatoio
IN03	Bit0	Bit '0' trasduttore livello serbatoio
IN04	emergenza	Pulsante (NC) simulatore emergenza esterna
IN05	selettore	Selettore (NO) commutazione 'Manuale/Automatico'
IN06	StartAutomatico	Pulsante (NO) 'Start Automatico'
IN07	StopAutomatico	Pulsante (NC) 'Stop Automatico'
IN10	Start_manuale1	Pulsante (NO) start manuale 'Carico Prodotto 1 '
IN11	Stop_manuale1	Pulsante (NC) stop manuale 'Carico Prodotto 1 '
IN12	Start_manuale2	Pulsante (NO) start manuale 'Carico Prodotto 2 '
IN13	Stop_manuale2	Pulsante (NC) stop manuale 'Carico Prodotto 2 '
IN14	Star_vapore_mescolazione	Pulsante (NO) start manuale 'Agitatore-Vapore '
IN15	Stop_vapore_mescolazione	Pulsante (NC) stop manuale 'Agitatore-Vapore '
IN16	Start_scarico	Pulsante (NO) start manuale 'Scarico Serbatoio '
IN09	Stop_scarico	Pulsante (NC) stop manuale 'Scarico Serbatoio '
OUT00	Mandata1	Elettrovalvola mandata prodotto 1
OUT01	Pompal	Motore pompa P1
OUT02	Pompa2	Motore pompa P2
OUT03	Mandata1	Elettrovalvola mandata prodotto 2
OUT04	scarico	Elettrovalvola scarico serbatoio
OUT05	Motore mescolazione	Motore agitatore miscela
OUT06	Lamp_verde1	Lampada segnalazione Verde
OUT07	Lamp_verde2	Lampada segnalazione Verde
OUT08	Lamp_verde3	Lampada segnalazione Verde
OUT09	Lamp_verde4	Lampada segnalazione Verde
OUT10	Lamp_giallo	Lampada segnalazione Gialla
OUT11	Lamp_blu1	Lampada segnalazione Blu
OUT12	Lamp_blu2	Lampada segnalazione Blu
OUT13	Lamp_rosso	Lampada segnalazione Rossa
OUT14	vapore	Elettrovalvola vapore



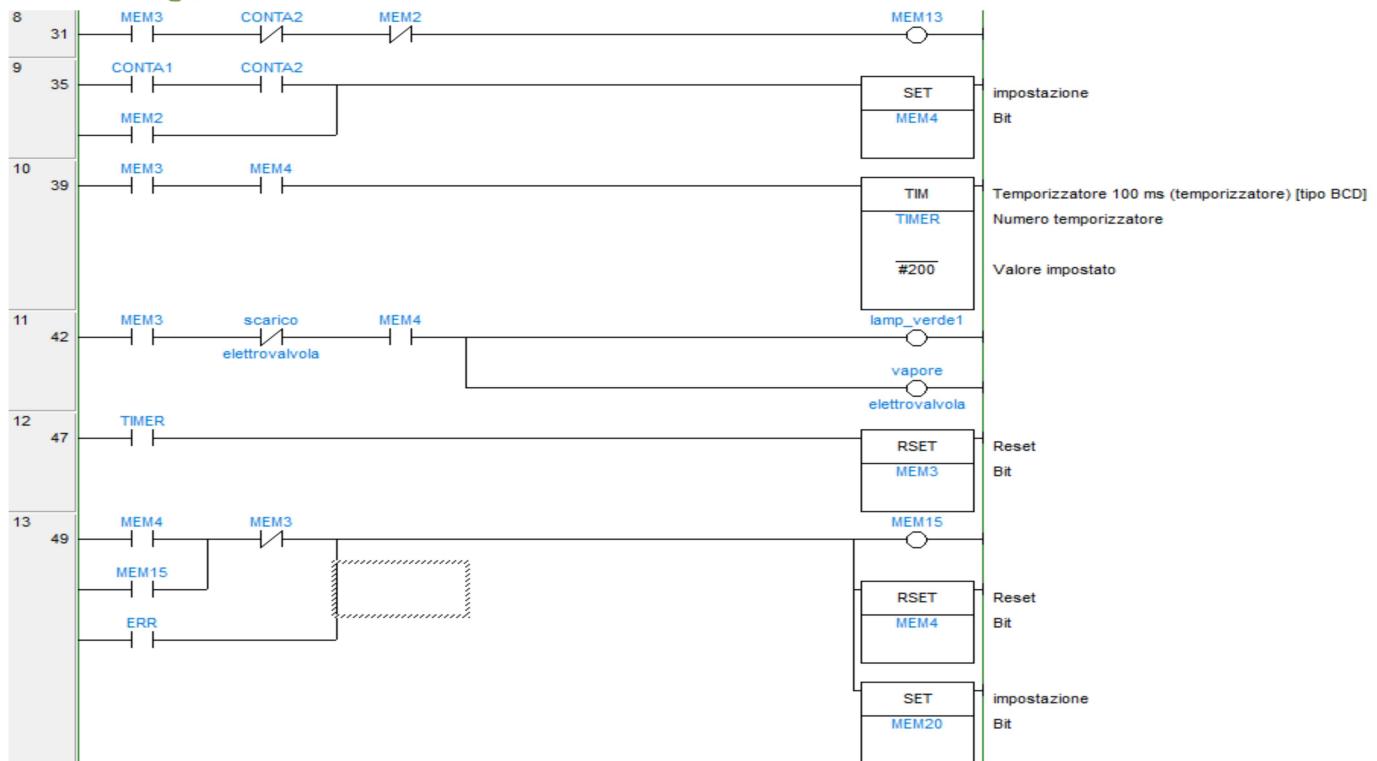
Schema ladder del blocco





Istituto di Istruzione Superiore "Benedetto Castelli"

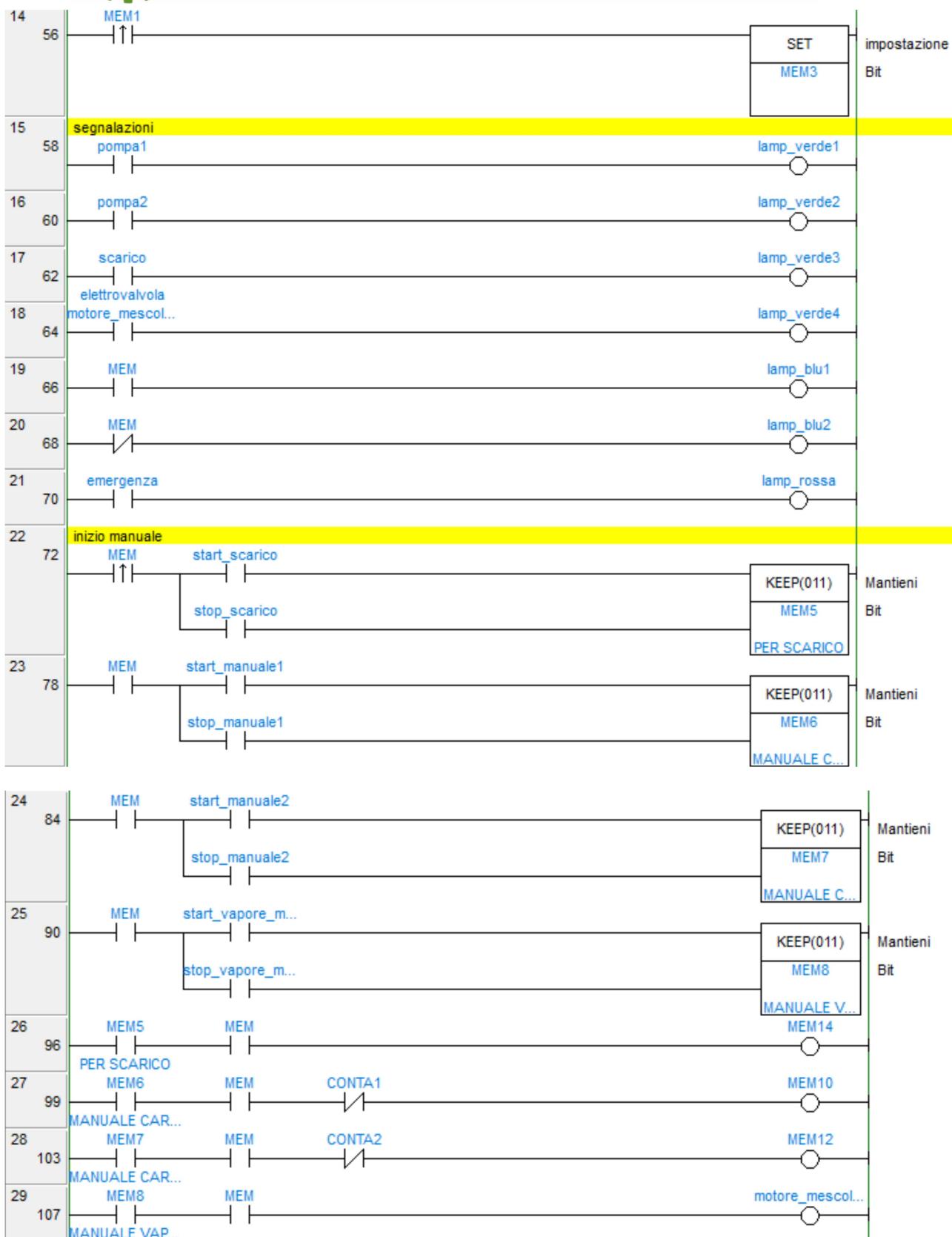
Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale
Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it





Istituto di Istruzione Superiore "Benedetto Castelli"

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale
Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it





Istituto di Istruzione Superiore “Benedetto Castelli”

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale
Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it

30	utilizzato per trasmettere il valore dei counter per diagnostica	P On	Flag di sempre ...	BIN(023)	BCD - Binario
109				CONTA1	Canale di origine
				OUTCONTA...	Canale del risultato
31	utilizzato per trasmettere il valore dei counter per diagnostica	P On	Flag di sempre ...	BIN(023)	BCD - Binario
111				CONTA2	Canale di origine
				OUTCONTA...	Canale del risultato
32	utilizzato per trasmettere il valore dei timer per diagnostica	P On	Flag di sempre ...	BIN(023)	BCD - Binario
113				TIMER	Canale di origine
				OUTTIMER1	Canale del risultato



Ciclo elettropneumatico con un cilindro a doppio effetto

Descrizione

Il processo simula il movimento di un cilindro a doppio effetto comandato da una elettrovalvola bistabile.

Il cilindro a doppio effetto può essere mosso direttamente dall'operatore nei due versi facendolo scorrere lungo il suo asse e quindi si ha il completo controllo dell'organo meccanico da governare.

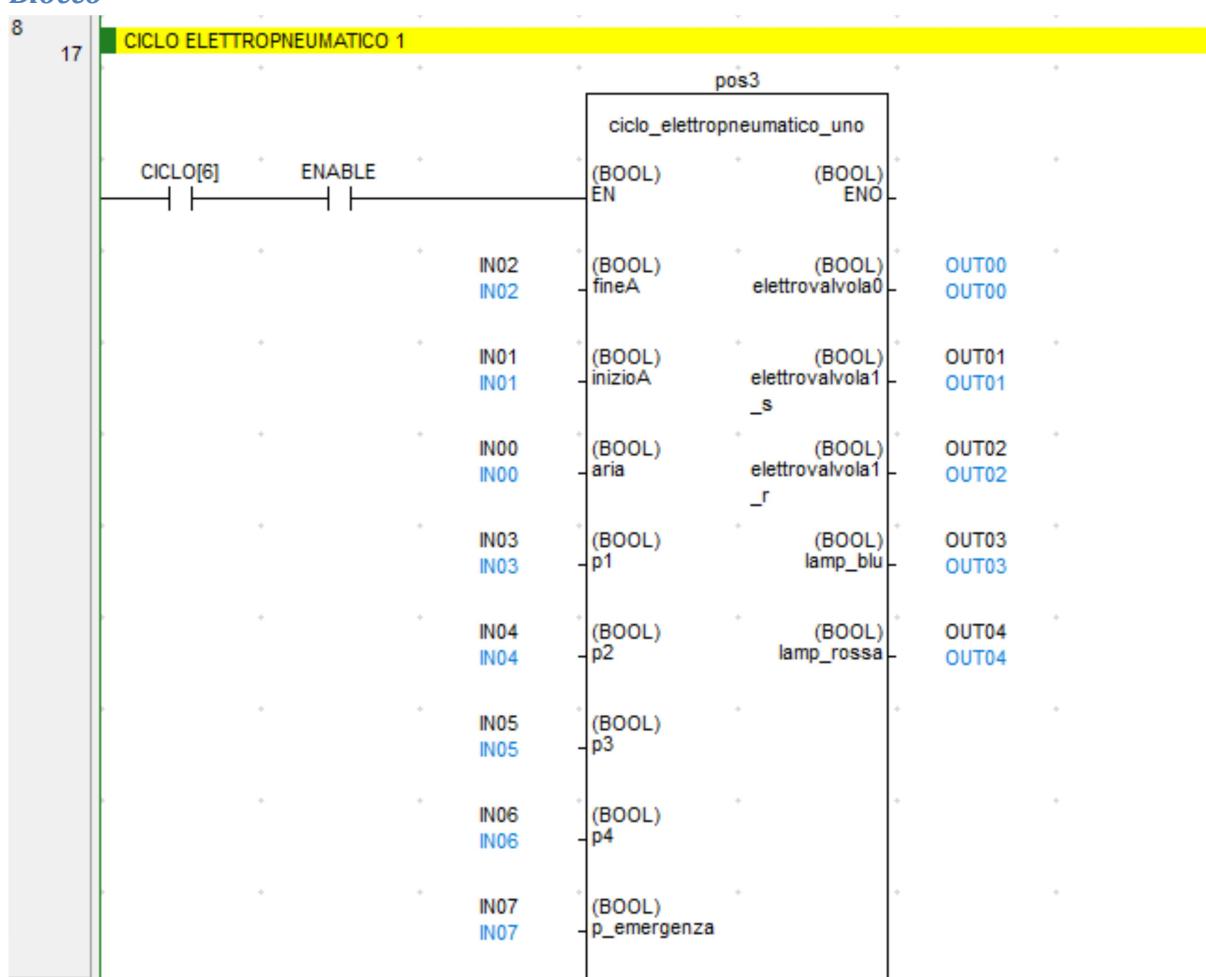
Una valvola ha un funzionamento simile a un rubinetto che permette il passaggio di un fluido attraverso un condotto nella valvola.

Con il termine “elettrico” si indica che l’attuatore è comandato mediante un segnale elettrico e non manuale o meccanico.

Per elettrovalvola bistabile si intende che l’elettrovalvola ha due stati che sono comandati da due segnali elettrici .

Dopo aver inserito l’aria e premuto lo start il pistone esce andando a premere il finecorsa IN02 dopodiché ritorna in posizione iniziale dove c’è il finecorsa IN01; compiendo un ciclo avanti indietro.

Blocco



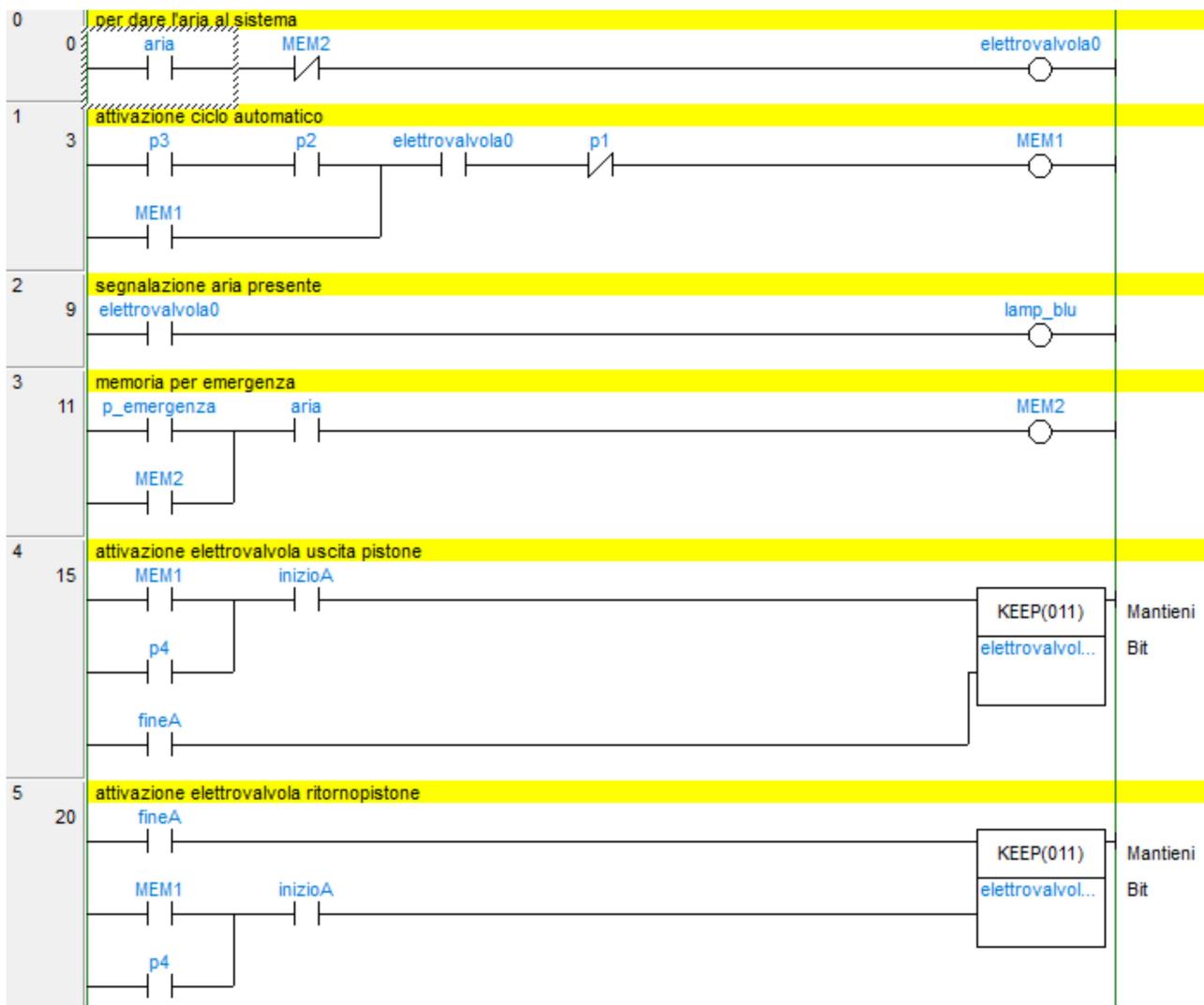


Variabili

Simbolo	Indirizzo	Commento
IN00	Aria	Selettore (NO)
IN01	inizioA	Fine corsa (NO) pistone posizione 0
IN02	fineA	Fine corsa (NO) pistone posizione 1
IN03	P1	Pulsante (NO)
IN04	P2	Pulsante (NO)
IN05	P3	Pulsante (NC)
IN06	P4	Pulsante (NO)
IN07	P_emergenza	Pulsante (NC)
OUT00	Elettrovalvola0	Bobina Set elettrovalvola 0Y1
OUT01	Elettrovalvolal s	Bobina Set elettrovalvola YV1
OUT02	Elettrovalvolal r	Bobina Reset elettrovalvola YV1
OUT03	Lamp blu	Lampada blu
OUT04	Lamp rossa	Lampada rossa

Schema ladder del blocco

Il contatto aria serve per azionare l'aria dell'intero sistema; i due blocchi keep l'andata e il ritorno del pistone.





Ciclo elettropneumatico con due cilindri a doppio effetto

Descrizione

Il processo simula due cilindri a doppio effetto con due valvole bistabili, seguendo il seguente ciclo:

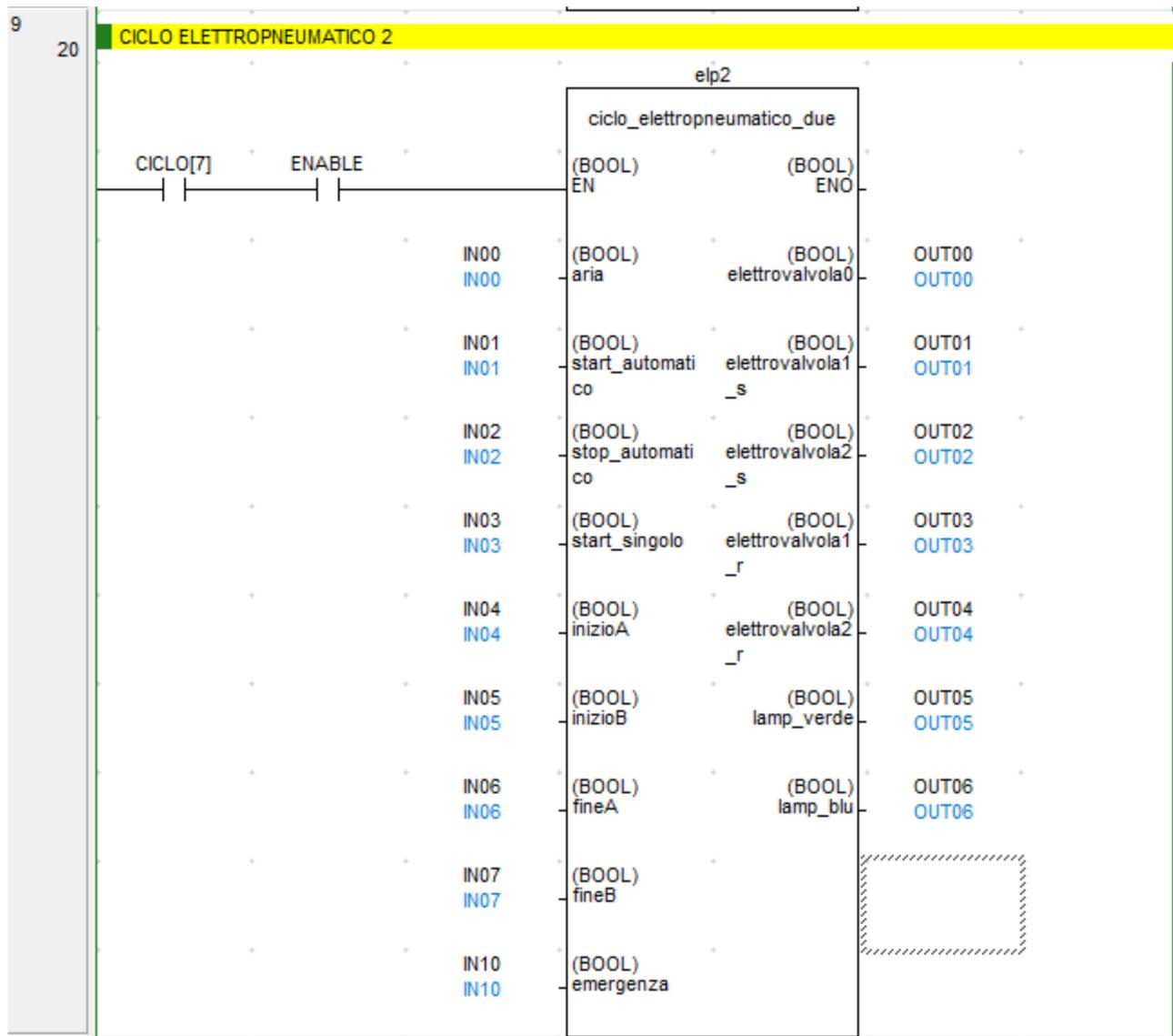
A+ B+ A- B-

Dopo aver inserito l'aria e premuto lo start il pistoneA esce andando a premere il finecorsa IN02 dopodiché esce il pistoneB che fa ritornare il posizionA e poi il pistoneB, compiendo un ciclo continuo.

Fino a quando non viene premuto il pulsante "IN02" stop_automatico che fa arrestare il ciclo.

Si può decidere tra un ciclo automatico o farne eseguire uno singolo con "IN03".

Blocco





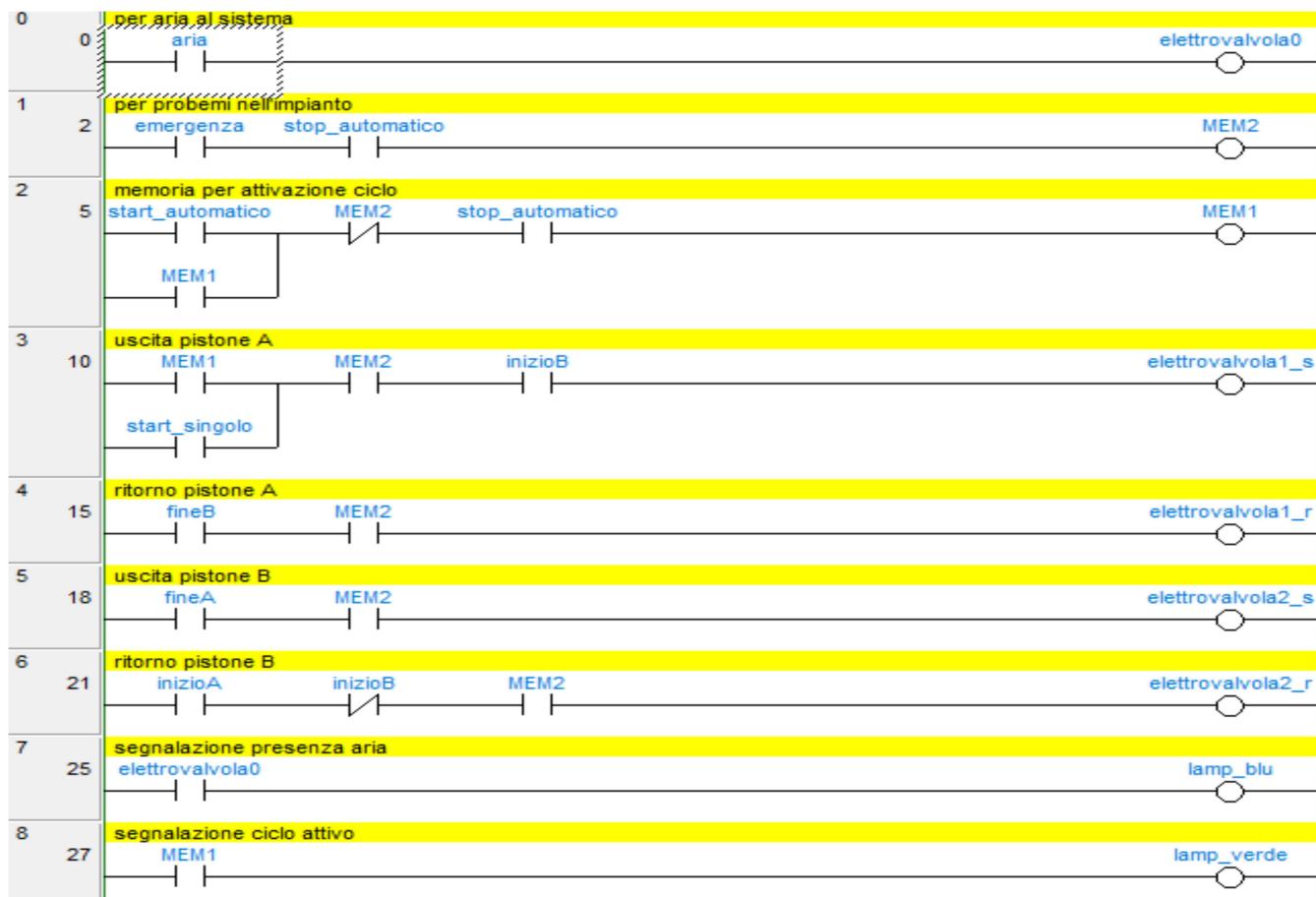
Variabili

IN00	aria	Selettore (NO) ON/OFF aria compressa
IN01	Start_automatico	Pulsante (NC)
IN02	Stop_automatico	Pulsante (NO)
IN03	Ciclo_singolo	Pulsante (NO)
IN04	inizioA	Micro (NO) posizione pistone A-
IN05	inizioB	Micro (NO) posizione pistone A+
IN06	fineA	Micro (NO) posizione pistone B-
IN07	fineb	Micro (NO) posizione pistone B+
IN10	emergenza	Pulsante (NC) simulazione emergenza esterna
OUT00	Elettrovalvola0	Bobina comando valvola 0V (12)
OUT01	Elettrovalvola1_s	Bobina set valvola 1V1 (14)
OUT02	Elettrovalvola1_r	Bobina reset valvola 1V1 (12)
OUT03	Elettrovalvola2_s	Bobina set valvola 2V1 (14)
OUT04	Elettrovalvola_r	Bobina reset valvola 2V1 (12)
OUT06	Lamp_blu	Lampada segnalazione blu
OUT05	Lamp_verde	Lampada segnalazione verde

Schema ladder del blocco

Il contatto aria serve per azionare l'aria dell'intero sistema;

Per programmare il processo si è usato uno schema pneumatico integrato con un comando elettrico.





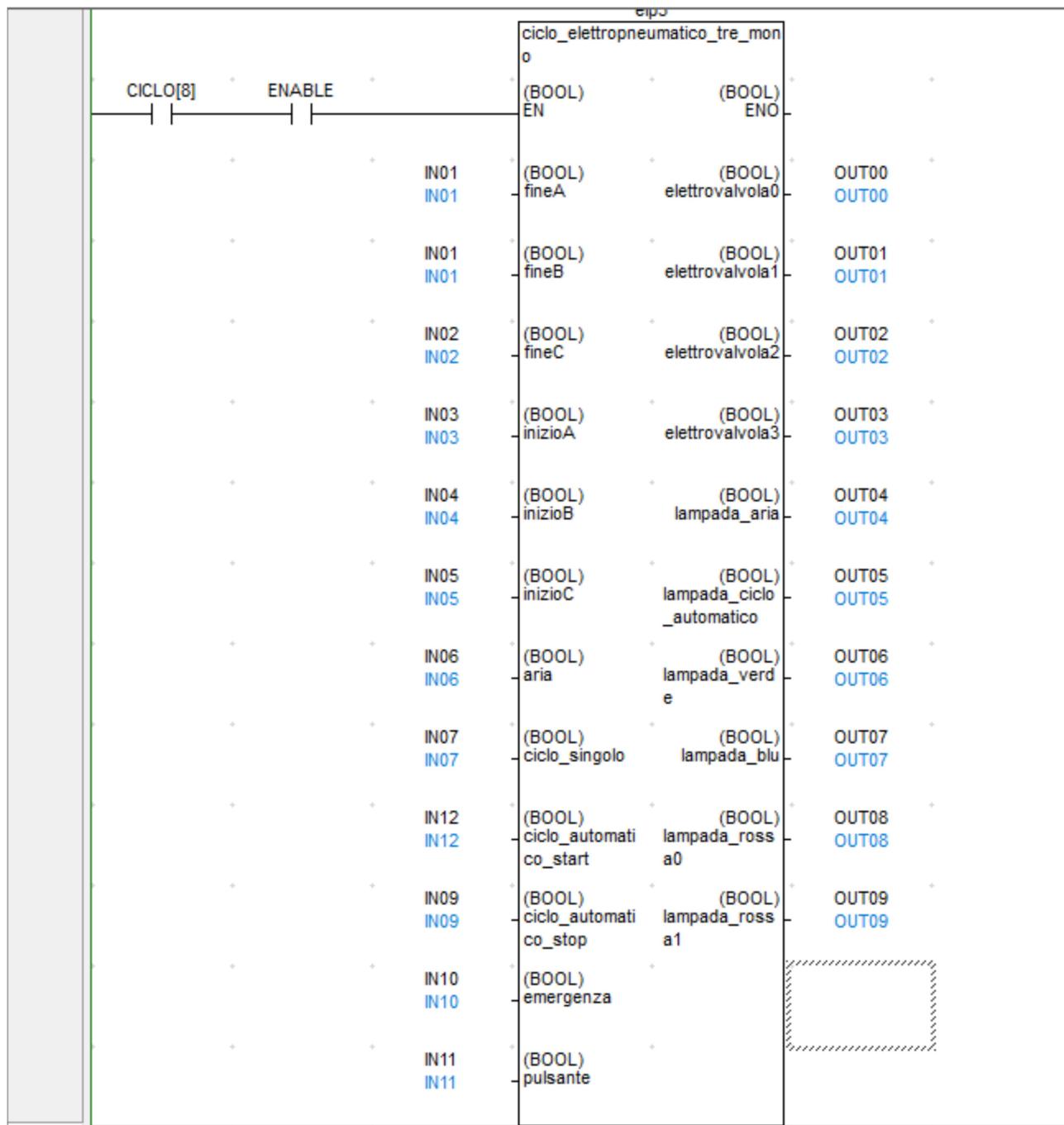
Ciclo elettropneumatico con tre cilindri a doppio effetto

Descrizione

Il ciclo comprende tre pistoni A,B,C che si muovo seguendo il seguente ciclo:

A+ B+ C+ A- B- C-

Blocco



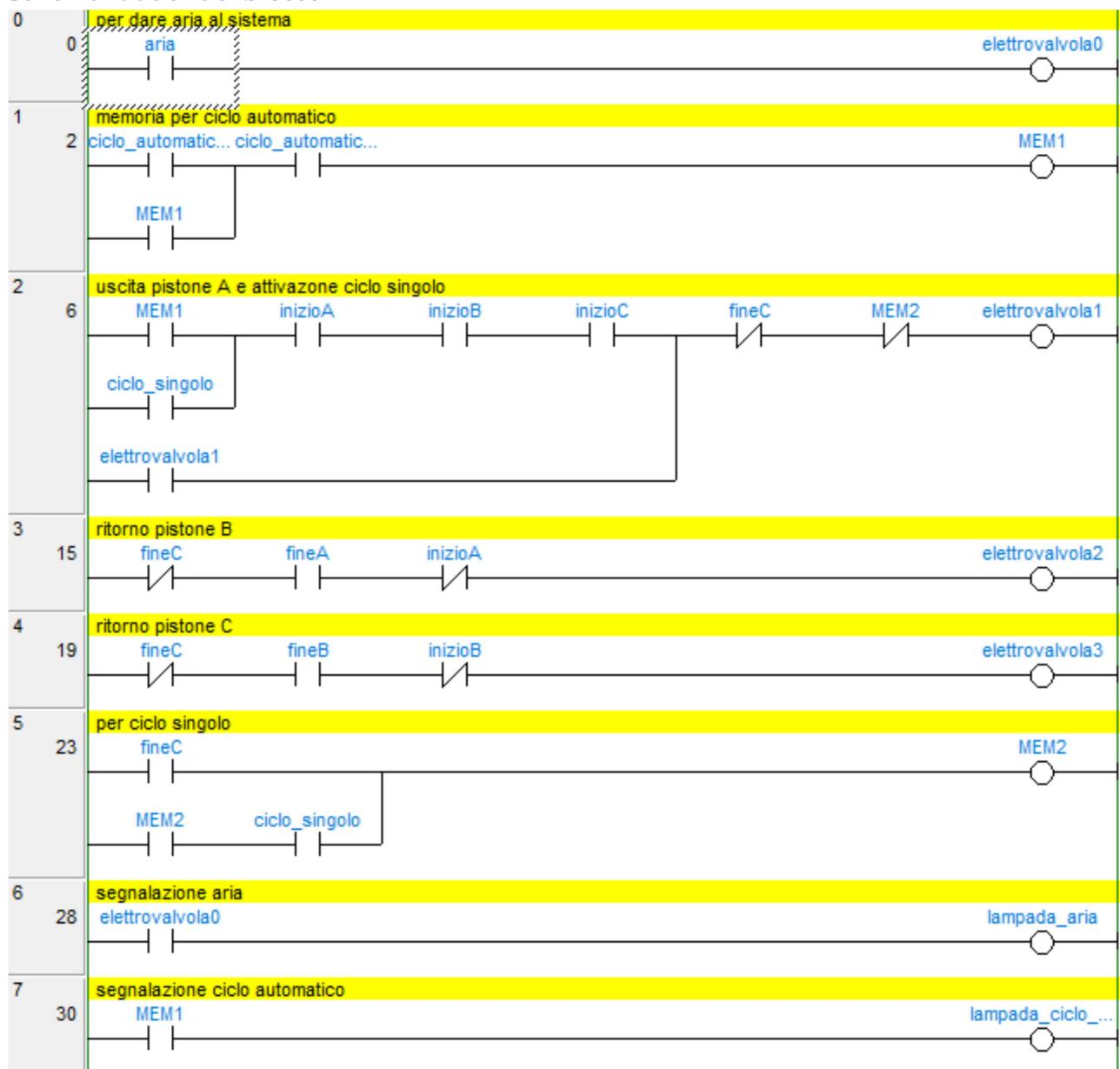


Variabili

Simbolo	Indirizzo	Commento
IN03	Inizio a	Fine corsa (NO) posizione - pistone A
IN00	Fine a	Fine corsa (NO) posizione + pistone A
IN04	Inizio b	Fine corsa (NO) posizione - pistone B
IN01	Fine b	Fine corsa (NO) posizione + pistone B
IN05	Inizio c	Fine corsa (NO) posizione - pistone C
IN02	Fine c	Fine corsa (NO) posizione + pistone C
IN06	aria	Selettore (NO) comando aria
IN07	Ciclo singolo	Pulsante (NO)
IN12	Start_ciclo_autoatico	Pulsante (NO)
IN09	Stop ciclo autoatico	Pulsante (NC)
IN10	emergenza	Pulsante (NC) simulazione emergenza esterna
IN11	PULSANTE	Pulsante (NC)
OUT00	elettrovalvola0	Bobina elettrovavola 0V
OUT01	elettrovalvola1	Bobina elettrovavola 1V1
OUT02	elettrovalvola2	Bobina elettrovavola 2V1
OUT03	elettrovalvola3	Bobina elettrovavola 3V1
OUT05	Lampada_ciclo automatico	Lampada segnalazione blu
OUT06	Lamp blu	Lampada segnalazione blu
OUT07	Lamp verde	Lampada segnalazione verde
OUT08	Lamp rossa0	Lampada segnalazione rossa
OUT09	Lamp rossal1	Lampada segnalazione rossa
OUT04	Lampada aria	Lampada aria



Schema ladder del blocco





PROGRAMMAZIONE CX-SUPERVISOR

Nella parte di visione si sono simulati tutti i processi sopra elencati aggiungendone altri che gli studenti possono programmare a loro discrezione.

In alcuni processi non sono stati necessari script perché le movimentazioni o segnalazioni sono state rese possibili grazie ad un interfacciamento diretto.

Variabili programmazione Cx_supervisor

NOME	INDIRIZZO	COMMENTO	TIPO	PLC
I0	0,00	INGRESSO1	BOOL	PLC
I1	0,01	INGRESSO2	BOOL	PLC
I2	0,02	INGRESSO3	BOOL	PLC
I3	0,03	INGRESSO4	BOOL	PLC
I4	0,04	INGRESSO5	BOOL	PLC
I5	0,05	INGRESSO6	BOOL	PLC
I6	0,06	INGRESSO7	BOOL	PLC
I7	0,07	INGRESSO8	BOOL	PLC
I8	0,08	INGRESSO9	BOOL	PLC
I9	0,09	INGRESSO10	BOOL	PLC
I10	0,10	INGRESSO11	BOOL	PLC
I11	0,11	INGRESSO12	BOOL	PLC
I12	0,12	INGRESSO13	BOOL	PLC
I13	0,13	INGRESSO14	BOOL	PLC
I14	0,14	INGRESSO15	BOOL	PLC
I15	0,15	INGRESSO16	BOOL	PLC
Q0	2,00	USCITA1	BOOL	PLC
Q1	2,01	USCITA2	BOOL	PLC
Q2	2,02	USCITA3	BOOL	PLC
Q3	2,03	USCITA4	BOOL	PLC
Q4	2,04	USCITA5	BOOL	PLC
Q5	2,05	USCITA6	BOOL	PLC
Q6	2,06	USCITA7	BOOL	PLC
Q7	2,07	USCITA8	BOOL	PLC
Q8	2,08	USCITA9	BOOL	PLC
Q9	2,09	USCITA10	BOOL	PLC
Q10	2,10	USCITA11	BOOL	PLC
Q11	2,11	USCITA12	BOOL	PLC
Q12	2,12	USCITA13	BOOL	PLC
Q13	2,13	USCITA14	BOOL	PLC
Q14	2,14	USCITA15	BOOL	PLC
Q15	2,15	USCITA16	BOOL	PLC
TIMER1	W40	INGRESSO TIMER1	INT	PLC



TIMER2	W50	INGRESSO TIMER2	INT	PLC
TIMER3	W60	INGRESSO TIMER3	INT	PLC
TIMER4	W70	INGRESSO TIMER4	INT	PLC
TIMER5	W80	INGRESSO TIMER5	INT	PLC
TIMER6	W90	INGRESSO TIMER6	INT	PLC
CONTATORE1	W100	INGRESSO CONTATORE 1	INT	PLC
CONTATORE2	W110	INGRESSO CONTATORE 2	INT	PLC
CONTATORE3	W120	INGRESSO CONTATORE 3	INT	PLC
CONTATORE4G	W130	INGRESSO CONTATORE 4	INT	PLC
CONTATORE5	W140	INGRESSO CONTATORE 5	INT	PLC
ARRAY_VISIONE[]		ARRAY PER LA VISIONE	INT	PLC
ARRAT-BOOL[]		ARRAY VISIONE BOLEANO	BOOL	PLC
ARRAY VISIONE2		ARRAY PER LA VISIONE	INT	PLC

Definizione utilizzo array per la visione

Definizione di ARRAY_VISIONE con la specificazione di dove sono stati utilizzati i vari indirizzi dell'array.

NOME VARIABILE	INDIRIZZO	COMMENTO	TIPO
ARRAY_VISIONE	0	SCELTA TIMER 1	INT
	1	VISIONE TIMER 1	INT
	2	SCELTA TIMER 2	INT
	3	VISIONE TIMER 2	INT
	4	SCELTA COUNTER 1	INT
	5	VISIONE COUNTER 1	INT
	6	SCELTA COUNTER 2	INT
	7	VISIONE COUNTER 2	INT
	8	SCELTA VARIABILE 1	INT
	9	VISIONE VARIABILE 1	INT
	10	SCELTA VARIABILE 2	INT
	11	VISIONE VARIABILE2	INT
	12	SCELTA VARIABILE 3	INT
	13	VISIONE VARIABILE 3	INT
	14	SCELTA VARIABILE 4	INT
	15	VISIONE VARIABILE 4	INT
	16	BYTE 1 INGRESSO	INT
	17	BYTE 2 INGRESSO	INT
	18	BYTE 1 USCITA	INT
	19	BYTE 2 USCITA	INT
	20	MOVIMENTO CANCELLO	INT
	21	APPOGGIO IO	INT
	22	APPOGGIO I1	INT
	23	APPOGGIO I2	INT



Istituto di Istruzione Superiore “Benedetto Castelli”

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale

Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it

24	APPOGGIO I3	INT
25	APPOGGIO I4	INT
26	APPOGGIO I5	INT
27	APPOGGIO I6	INT
28	APPOGGIO I7	INT
29	APPOGGIO I8	INT
30	APPOGGIO I9	INT
31	APPOGGIO I10	INT
32	APPOGGIO I11	INT
33	APPOGGIO I12	INT
34	APPOGGIO I13	INT
35	APPOGGIO I14	INT
36	APPOGGIO I15	INT
37	APPOGGIO Q0	INT
38	APPOGGIO Q1	INT
39	APPOGGIO Q2	INT
40	APPOGGIO Q3	INT
41	APPOGGIO Q4	INT
42	APPOGGIO Q5	INT
43	APPOGGIO Q6	INT
44	APPOGGIO Q7	INT
45	APPOGGIO Q8	INT
46	APPOGGIO Q9	INT
47	APPOGGIO Q10	INT
48	APPOGGIO Q11	INT
49	APPOGGIO Q12	INT
50	APPOGGIO Q13	INT
51	APPOGGIO Q14	INT
52	APPOGGIO Q15	INT

Definizione di ARRAY_VISIONE2 con la specificazione di dove sono stati utilizzati i vari indirizzi dell'array.

ARRAY VISIONE2	0	Elettrovalvola sopra	INT
2000	1	Elettrovalvola sotto	INT
	2	Pistone	INT
	3	SPOSTAMENTO MACCHINA X	INT
	4	SPOSTAMENTO MACCHINA Y	INT
	5	NUMERO MACCHINE	INT
	6	CANCELLA INGRESSO	INT
	7	CANCELLA USCITA	INT
	8	Y INCREMENTO	INT



	9	X INCREMENTO	INT
	10	LIQUIDO CARICA	INT
	11	LIQUIDO SERBATOIO	INT
	12	INCREMENTO CARICA	INT
	13	INCREMENTO SERBATOIO	INT
	14	INCREMENTO SERBATOIO MISCELATORE	INT
	15	INCREMENTO Incremento SERBATOIO MISCELA-TORE	INT
	16	incremento Elp1 andata	INT
	17	incremento Elp1 ritorno	INT
	18	Contatore1	INT
	19	Contatore 1	INT
	20	Contatore 2	INT
	21	Contatore 2	INT

Definizione di ARRAT_BOOL con la specificazione di dove sono stati utilizzati i vari indirizzi dell'array.

ARRAT-BOOL	0		BOOL
	1	SCELTA I/Q 1	BOOL
	2	SCELTA I/Q 2	BOOL
	3	SCELTA I/Q 3	BOOL
	4	SCELTA I/Q 4	BOOL
	5	SELETTORE CANCELLA	BOOL
	6	MACCHINA MOVIMENTO	BOOL
	7	MACCHINA POSIZIONE 1	BOOL
	8	MACCHINA POSIZIONE 2	BOOL
	9	MACCHINA POSIZIONE 3	BOOL
	10	MACCHINA POSIZIONE 4	BOOL
	11	MACCHINA POSIZIONE 5	BOOL
	12	MACCHINA POSIZIONE 6	BOOL
	13	supporto	BOOL
	14	POWER	BOOL
	15		BOOL
	16		BOOL
	17		BOOL
	18		BOOL
	19		BOOL
	20		BOOL
	21	MAT1IO	BOOL
	22	MAT2IO	BOOL
	23	MAT3IO	BOOL
	24	ELP1IO	BOOL
	25	ELP2IO	BOOL



	26	ELP3IO	BOOL
	27	SEMAFOROIO	BOOL
	28	CANCELLOIO	BOOL
	29	GARAGEIO	BOOL
	30	CONTROLLOLIVELLOIO	BOOL
	31	MISCELATOREIO	BOOL
	32		BOOL
	33		BOOL
	34		BOOL
	35		BOOL
	36		BOOL
	37		BOOL
	38		BOOL
	39		BOOL
	40	MACCHINA ORIZZONTALE	BOOL
	41	APPOGGIO VISIONE GARAGE	BOOL
	42	APPOGGIO VISIONE GARAGE	BOOL
	43	APPOGGIO VISIONE GARAGE	BOOL
	44	APPOGGIO VISIONE GARAGE	BOOL
	45	APPOGGIO VISIONE GARAGE	BOOL
	46	APPOGGIO VISIONE GARAGE	BOOL
	47	APPOGGIO VISIONE GARAGE	BOOL



Layout applicazione

Informazione pagina

Il layout permette di mettere la simulazione del plc in run/stop, visualizzare lo stato logico degli ingressi delle uscite del processo in esecuzione e avere delle informazioni sul VIRTUAL_PLC.

La funzione touch, consente di selezionare attraverso il singolo box, il processo che si desidera avviare scegliendo fra quelli disponibili.

Il “ciclo elettropneumatico con tre cilindri a doppio effetto” è stato predisposto per essere implementato attraverso il semplice collegamento delle variabili tra plc e visione e programmazione dello script.

Dell’“ascensore” esiste solo l’immagine del processo con i comandi ai quali l’utente potrà collegarsi attraverso l’uso di opportune variabili con un algoritmo da programmare secondo le proprie esigenze.

Pagina





Marcia arresto di un motore MAT

Informazione Processo

Il pannello mette a disposizione un M.A.T. (M1) alimentato attraverso un contattore (KM) e protetto da un relè termico.

Start I0.00: per avviare il motore

Stop I0.01: per arrestare il motore

Emergenza I0.04: simula un'emergenza di guasto

Termica I0.03: simula l'intervento di una protezione termica



Chiudi

Pagina

Marcia e Arresto di un Motore Trifase

Comandi		
Off	Start	I0.00
Off	Stop	I0.01
Off	Emergenza	I0.04

Segnalazioni	
Segnalazione Alimentazione	I0.02
Segnalazione in linea	I0.00
Segnalazione emergenza	I0.03
Segnalazione Termica	I0.01

OMRON CJ1M	
SYSMAC	
24 DC IN	24 DC OUT
Q0.00	Q0.00
Q0.01	Q0.01
Q0.02	Q0.02
Q0.03	Q0.03
Q0.04	Q0.04
Q0.05	Q0.05
Q0.06	Q0.06
Q0.07	Q0.07
Q0.08	Q0.08
Q0.09	Q0.09
Q0.10	Q0.10
Q0.11	Q0.11
Q0.12	Q0.12
Q0.13	Q0.13
Q0.14	Q0.14
Q0.15	Q0.15
Q1.00	Q1.00
Q1.01	Q1.01
Q1.02	Q1.02
Q1.03	Q1.03
Q1.04	Q1.04
Q1.05	Q1.05
Q1.06	Q1.06
Q1.08	Q1.08
Q1.09	Q1.09
Q1.10	Q1.10
Q1.11	Q1.11
Q1.12	Q1.12
Q1.13	Q1.13
Q1.14	Q1.14
Q1.15	Q1.15
Q2.00	Q2.00

Script

```

1. if Q20 =True then
2. I02=True
3. else
4. I02=False
5. End If

```



Inversione di marcia di un motore MAT

Descrizione

Nella pagina sottostante viene rappresentata la soluzione attraverso l'uso dei diagrammi di stato.

Si è adotto questo metodo risolutivo, solo in questo processo, come esempio di risoluzione sistematica.

Il pannello mette a disposizione un M.A.T. (M1) e due contattori (KM1, KM2) che permettono di invertire il senso ciclico delle fasi sull'alimentazione consentendo quindi, l'inversione del senso di rotazione del motore.

Stop I0.01: per arrestare il motore

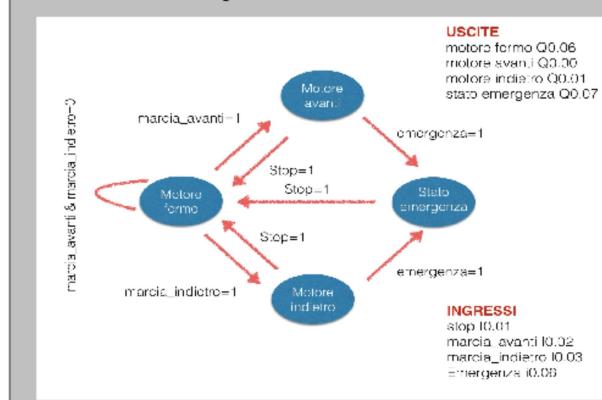
Emergenza I0.06: simula un emergenza di guasto

Marcia avanti I0.02 : per far ruotare in senso orario il motore

Marcia indietro I0.03 : per far ruotare in senso antiorario il motore

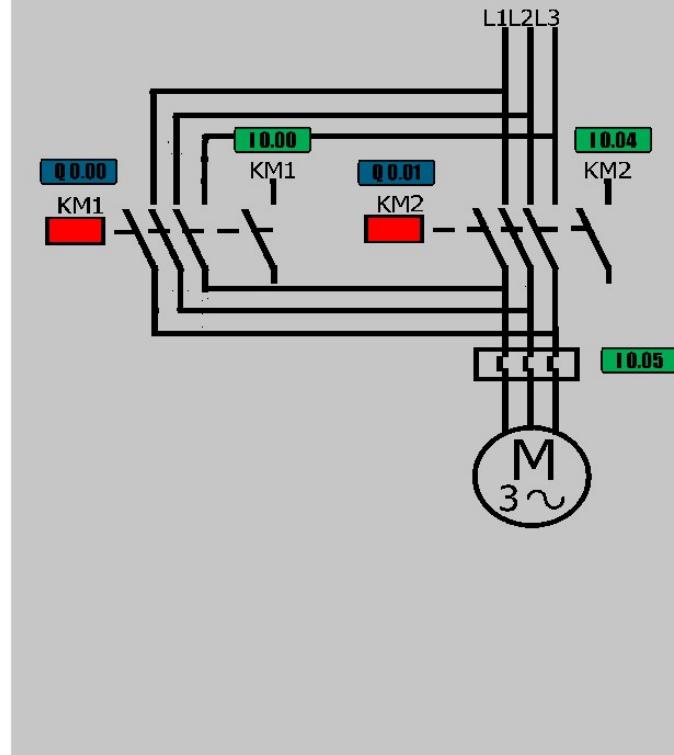


Chiudi



Pagina

Inversione di Marcia di un Motore Trifase



Comandi

	Stop	I0.01
	Marcia Avanti	I0.02
	Marcia Indietro	I0.03
	Emergenza Esterna	I0.06



Segnalazioni

	Segnalazione emergenza	Q0.07
	Segnalazione Marcia Avanti	Q0.03
	Segnalazione Marcia Indietro	Q0.04
	Segnalazione Alimentazione	Q0.06



Script

```
1. IF Q21= TRUE THEN  
2. I04=TRUE  
3. ELSE  
4. I04=FALSE  
5. End If  
6.  
7. IF Q20= TRUE THEN  
8. I00=TRUE  
9. ELSE  
10. I00=FALSE  
11. End If
```



Avviamento stella-triangolo di un motore MAT

Descrizione

Il pannello mette a disposizione un M.A.T. (M1) e tre contattori (KM1, KM2, KM3) che permettono, opportunamente eccitati, di collegare il motore alla linea di alimentazione con due collegamenti diversi delle fasi.

Per ogni contattore è disponibile anche un contatto ausiliario NO KM1a, KM2a, KM3a collegati agli ingressi del PLC, per il controllo dell'effettivo stato fisico dei contattori stessi

Start I0.00: per avviare il motore

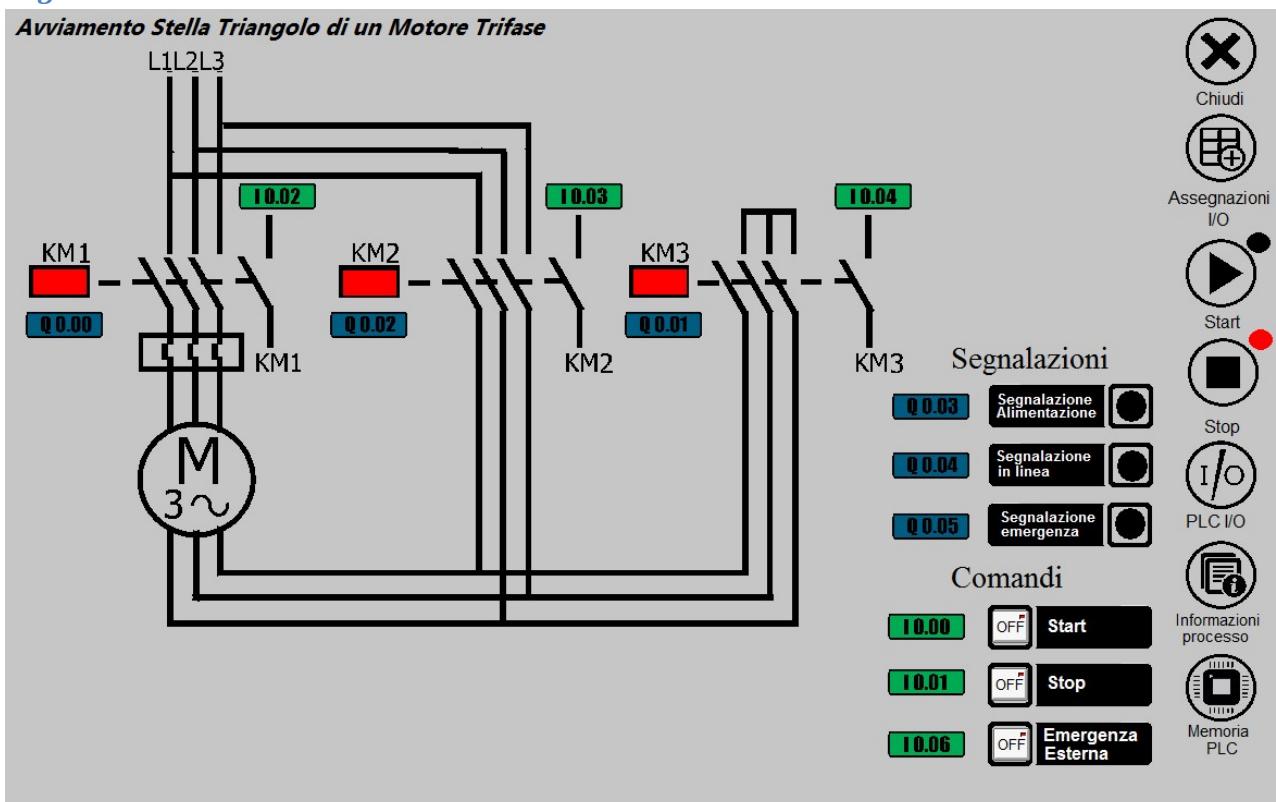
Stop I0.01: per arrestare il motore

Emergenza I0.06: simula un emergenza di guasto



Pagina

Avviamento Stella Triangolo di un Motore Trifase



Script

```

1. IF Q20 = TRUE THEN
2. I02=TRUE
3. ELSE
4. I02=FALSE
5. End If
6.
7. IF Q21= TRUE THEN
8. I04=TRUE
9. ELSE
10. I04=FALSE
11. End If
12.
13. IF Q22= TRUE THEN
14. I03=TRUE
15. ELSE
16. I03=FALSE
17. End If

```



Semaforo pedonale

Descrizione

Un passaggio pedonale viene regolato da un semaforo; due unità semaforiche regolano il traffico delle auto mentre le due unità regolano il traffico pedonale.

I semafori automobilistici sono, come di consueto, dotati di lampade di segnalazione rosse, gialle e verdi.

I semafori pedonali hanno ciascuno una lampada verde ed una rossa e un pulsante per la richiesta del verde per i pedoni.

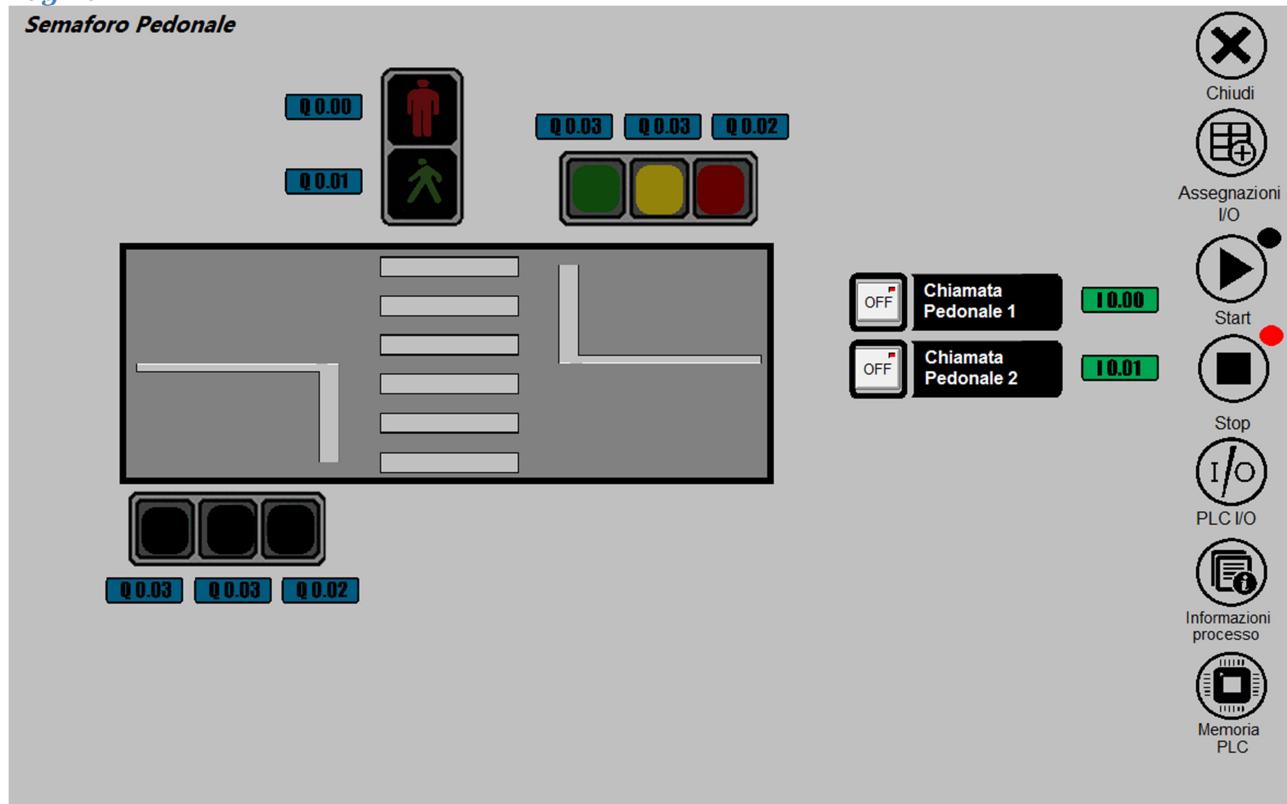
Chiamata Pedonale1 I0.00 e Chiamata Pedonale2 I0.01: servono a simulare la chiamata pedonale



Chiudi

Pagina

Semaforo Pedonale





Cancello automatico

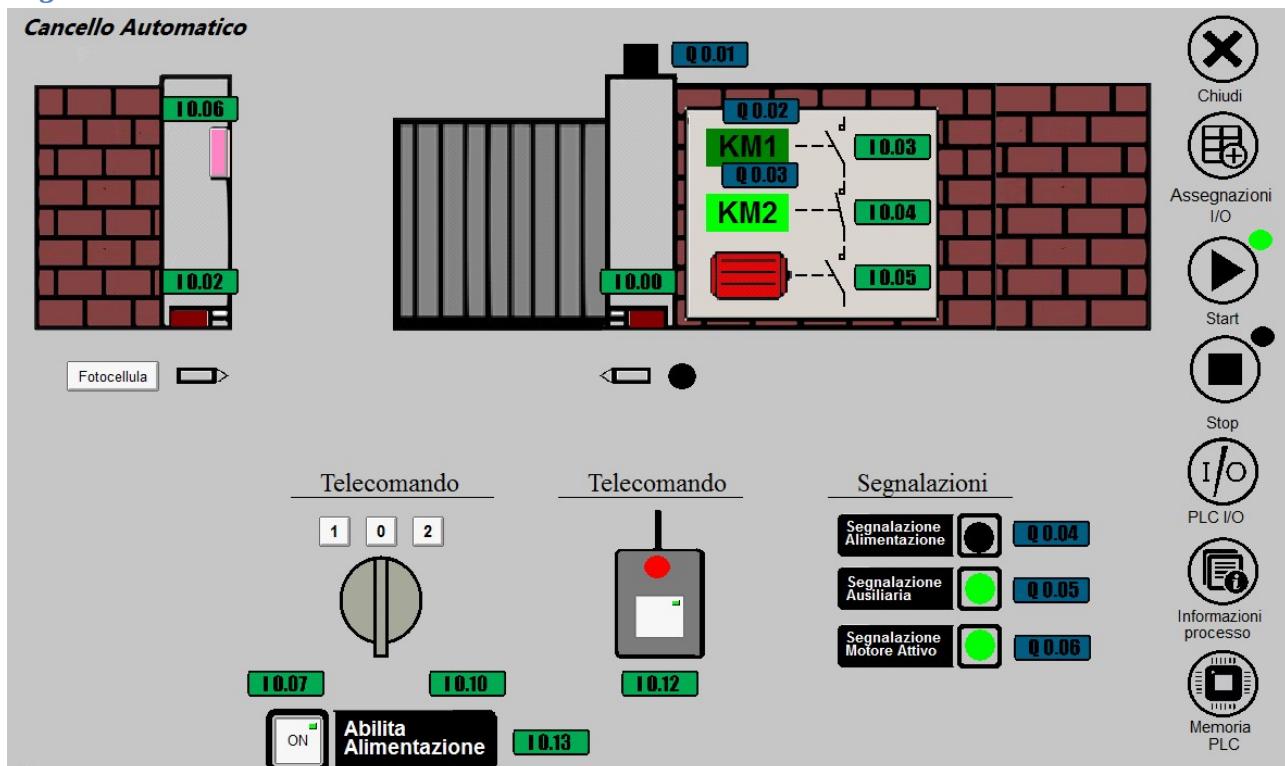
Descrizione

Il pannello mette a disposizione la movimentazione di un cancello. Il movimento del cancello avviene per opera di un M.A.T. comandato da due contattori (KM1, KM2) che permettono l’azionamento del motore nei due sensi di marcia. Una fotocellula che rileva l’interposizione fra proiettore (alimentato dall’uscita del PLC Q0.0) e ricevitore che fornirà un livello logico alto all’ingresso I0.1, quando si interporrà un ostacolo nel suo raggio di azione; la simulazione dell’ostacolo può essere effettuata clicando con il mouse sul pulsante omonimo. Una costa pneumatica, posta nella parte fissa del cancello che interviene nel caso di un’azione di esterna di opposizione al movimento di chiusura del cancello; essa è dotata di un contatto (NC) collegato all’ingresso I0.6 del PLC (l’ingresso è a livello 0 quando interviene la protezione); la simulazione del suo intervento può essere effettuata facendo clic con il mouse sul suo simbolo (rettangolo rosa).



Chiudi

Pagina



Script

Si usa array visione20 per la movimentazione del cancello e in base alla sue posizione si attivano i finecorsa.

```

6. if Q23=True then
7. ARRAY_VISIONE(20)=ARRAY_VISIONE(20)+5
8. End If
9.
10. If Q22=True then
11. ARRAY_VISIONE(20)=ARRAY_VISIONE(20)-5
12. End If
13.
14. if ARRAY_VISIONE(20)=0 then
15. I02=TRUE
16. ELSE
17. I02=FALSE
18. End If
19.

```



```
20. if ARRAY_VISIONE(20)>= 75 then
21. I00=TRUE
22. ELSE
23. I00=FALSE
24. End If
25.
26.
27. 'GRAFICA MOTORE ON 0 OFF
28. if Q23 or Q22 then
29. ARRAT_BOOL(5)=True
30. else
31. ARRAT_BOOL(5)=False
32. End If
33.
34.
35. if Q22 =True then
36. I03=True
37. else
38. I03=False
39. End If
40.
41. if Q23 =True then
42. I04=True
43. else
44. I04=False
45. End If
```



Garage automatico

Descrizione

In un garage, l'entrata e l'uscita sono regolate da due sbarre; entrambe vengono chiuse e aperte tramite un motore (rispettivamente M1 e M2); ogni motore è azionato da due contattori che ne permettono l'inversione di marcia. Per ogni contattore è disponibile anche un contatto ausiliario NO.



Chiudi

- un sensore di prossimità (entrata SQ1, uscita SQ4) che rileva la presenza di un veicolo;
- un pulsante (SB1 entrata, SB2 uscita) per la richiesta di entrata e di uscita dei veicoli.

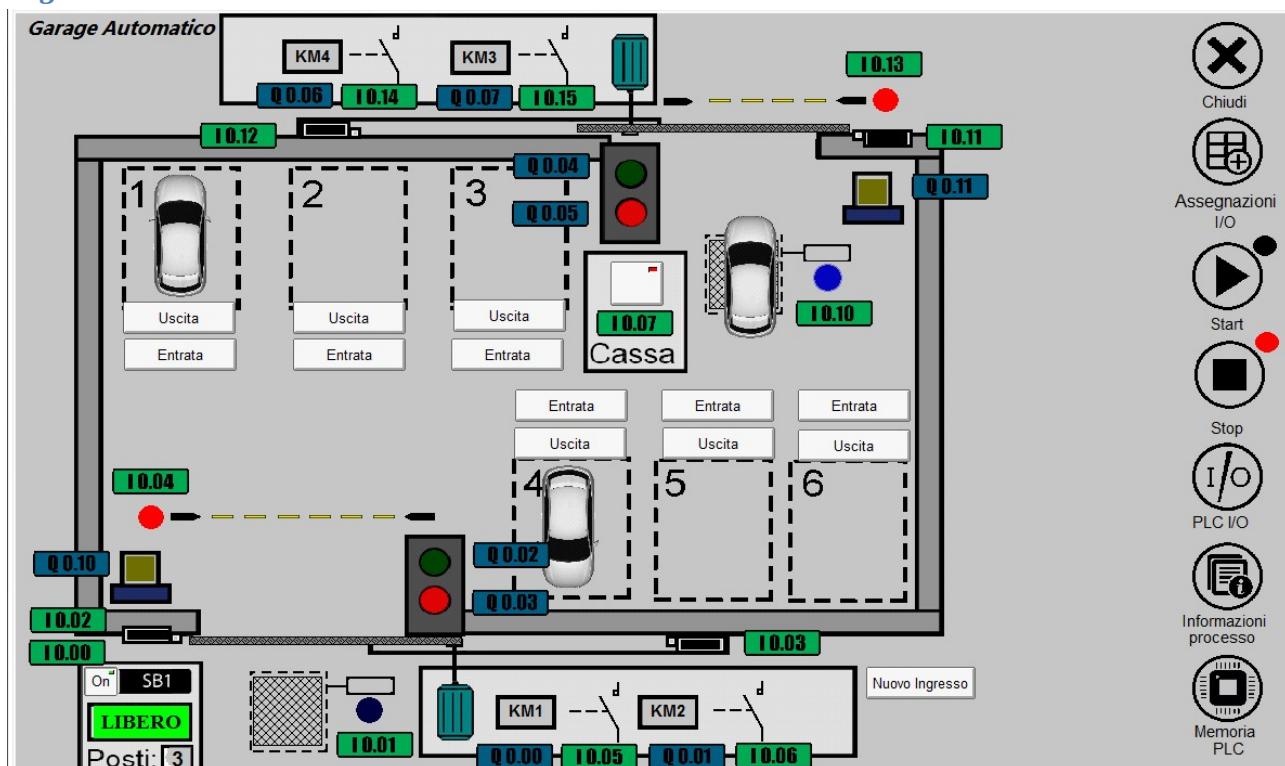
Dopo ogni sbarra è installata una fotocellula (entrata B1, uscita B2) per rilevare l'avvenuto passaggio del veicolo.

La predisposizione di un veicolo in entrata viene eseguita azionando con un clic del mouse il pulsante ‘Nuovo Ingresso’; scegliendo il posto mediante i pulsanti entrata la predisposizione di uno dei veicoli per l'uscita viene attuata con un clic sul pulsante uscita

Il movimento di ogni sbarra viene segnalato con il lampeggio delle lampade.

Nel garage, il semaforo posto in ingresso sarà verde, in presenza di posti liberi; viceversa passerà al rosso quando il garage è pieno. Il semaforo d'uscita, normalmente rosso, passerà al verde quando la sbarra è aperta per segnalare al veicolo il consenso all'uscita e ripasserà poi al rosso quando il veicolo avrà superato la fotocellula B2.

Pagina



Script

```

1. 'SENSORE AUTO1
2. if (ARRAY_VISIONE2(3)=325) and (ARRAY_VISIONE2(4)=0) then
3. I01=True
4. else
5. I01=False
6. End If
7.
8. 'SENSORE AUTO2
9. if (ARRAY_VISIONE2(3)=1375) and (ARRAY_VISIONE2(4)=2000) then
10. I10=True
11. else

```



```

12. I10=False
13. End If
14.
15. 'FOTOCELLULA 1
16. if (ARRAY_VISIONE2(3)=325) and (ARRAY_VISIONE2(4)=600) then
17. I04=True
18. else
19. I04=False
20. End If
21.
22. 'FOTOCELLULA 2
23. if (ARRAY_VISIONE2(3)=1375) and (ARRAY_VISIONE2(4)=2700) then
24. I13=True
25. else
26. I13=False
27. End If
28.
29.
30. 'FINECORS 1
31. IF ARRAY_VISIONE2(6)=0 THEN
32. I02=TRUE
33. Else
34. I02=FALSE
35. End If
36.
37.
38. IF ARRAY_VISIONE2(6)=1200 THEN
39. I03=TRUE
40. Else
41. I03=FALSE
42. End If
43.
44. 'CANCELLA APERTURA 1
45. IF Q20 AND I03=FALSE Then
46. ARRAY_VISIONE2(6)=ARRAY_VISIONE2(6)+25
47. End If
48.
49. IF Q21 AND I02=FALSE Then
50. ARRAY_VISIONE2(6)=ARRAY_VISIONE2(6)-25
51. End If
52.
53. 'FINECORS 2
54. IF ARRAY_VISIONE2(7)=0 THEN
55. I11=TRUE
56. Else
57. I11=FALSE
58. End If
59.
60.
61. IF ARRAY_VISIONE2(7)=1200 THEN
62. I12=TRUE
63. Else
64. I12=FALSE
65. End If
66.
67. 'CANCELLA APERTURA 2
68. IF Q22 AND I12=FALSE Then
69. ARRAY_VISIONE2(7)=ARRAY_VISIONE2(7)+25
70. End If
71.
72. IF Q23 AND I11=FALSE Then
73. ARRAY_VISIONE2(7)=ARRAY_VISIONE2(7)-25
74. End If

```



```
75.  
76.    'Movimento Y  
77.    ARRAY_VISIONE2(4)=ARRAY_VISIONE2(4)+ ARRAY_VISIONE2(8)  
78.    'Movimento X  
79.    ARRAY_VISIONE2(3)=ARRAY_VISIONE2(3)+ ARRAY_VISIONE2(9)  
80.  
81.    'ENTRATA MACCHINA 1  
82.    if I00=True and ARRAT_BOOL(32)=True then  
83.  
84.        if ARRAT_BOOL(7)=False then  
85.            'Macchina 1  
86.            if ARRAY_VISIONE2(4)<= 1400 then  
87.                ARRAY_VISIONE2(8)=10  
88.            Else  
89.                ARRAY_VISIONE2(8)=0  
90.            End If  
91.  
92.            if ARRAY_VISIONE2(4)>= 1400 and ARRAY_VISIONE2(3)>= 90 then  
93.                ARRAY_VISIONE2(9)=-5  
94.                if ARRAY_VISIONE2(4)<= 2000 then  
95.                    ARRAY_VISIONE2(8)=10  
96.                else  
97.                    ARRAY_VISIONE2(8)=0  
98.                end if  
99.            else  
100.                ARRAY_VISIONE2(9)=0  
101.  
102.                if ARRAY_VISIONE2(3)= 85 and ARRAY_VISIONE2(4)= 1880 then  
103.                    ARRAT_BOOL(7)=True  
104.                    ARRAT_BOOL(6)=False  
105.                End If  
106.            End If  
107.  
108.        End If  
109.    End If  
110.  
111.  
112.    'ENTRATA MACCHINA 2  
113.    if I00=True and ARRAT_BOOL(33)=True then  
114.        if ARRAT_BOOL(8)=False then  
115.            if ARRAY_VISIONE2(4)<= 1500 then  
116.                ARRAY_VISIONE2(8)=10  
117.            Else  
118.                ARRAY_VISIONE2(8)=0  
119.            End If  
120.  
121.            if ARRAY_VISIONE2(4)= 1510 then  
122.                ARRAT_BOOL(8)=True  
123.                ARRAT_BOOL(6)=False  
124.            End If  
125.        End If  
126.    End If  
127.  
128.    'ENTRATA MACCHINA 3  
129.    if I00=True and ARRAT_BOOL(34)=True then  
130.        if ARRAT_BOOL(9)=False then  
131.            if ARRAY_VISIONE2(4)<= 1500 then  
132.                ARRAY_VISIONE2(8)=10  
133.            Else  
134.                ARRAY_VISIONE2(8)=0  
135.            End If  
136.  
137.        if ARRAY_VISIONE2(4)= 1510 and ARRAY_VISIONE2(3)< 800 then
```



Istituto di Istruzione Superiore "Benedetto Castelli"

Istituto Tecnico Industriale, Istituto Professionale, Scuola in Ospedale

Via Cantore, 9 25128 Brescia tel 030/3700267 fax 030/395206 e-mail segreteria@itiscastelli.it

```

138.     ARRAY_VISIONE2(9)=10
139.     Else
140.     ARRAY_VISIONE2(9)=0
141.     End If
142.     if ARRAY_VISIONE2(4)= 1510 and ARRAY_VISIONE2(3)= 805 then
143.         ARRAT_BOOL(9)=True
144.         ARRAT_BOOL(6)=False
145.     End If
146. End If
147. End If
148.
149. 'ENTRATA MACCHINA 4
150.     if I00=True and ARRAT_BOOL(35)=True then
151.         if ARRAT_BOOL(9)=False then
152.             if ARRAY_VISIONE2(4)<= 1200 then
153.                 ARRAY_VISIONE2(8)=10
154.             Else
155.                 ARRAY_VISIONE2(8)=0
156.             End If
157.
158.
159.
160.         if ARRAY_VISIONE2(4)= 1210 and ARRAY_VISIONE2(3)<= 800 then
161.             ARRAT_BOOL(6)=False
162.             ARRAT_BOOL(40)=True
163.                 ARRAY_VISIONE2(9)=10
164.             Else
165.                 ARRAY_VISIONE2(9)=0
166.             End If
167.
168.         if ARRAY_VISIONE2(4)= 1210 and ARRAY_VISIONE2(3)= 805 then
169.             ARRAT_BOOL(10)=True
170.             ARRAT_BOOL(40)=False
171.
172.         End If
173.     End If
174. End If
175.
176. 'ENTRATA MACCHINA 5
177.     if I00=True and ARRAT_BOOL(36)=True then
178.         if ARRAT_BOOL(9)=False then
179.             if ARRAY_VISIONE2(4)<= 1200 then
180.                 ARRAY_VISIONE2(8)=10
181.             Else
182.                 ARRAY_VISIONE2(8)=0
183.             End If
184.
185.         if ARRAY_VISIONE2(4)= 1210 and ARRAY_VISIONE2(3)<= 1300 then
186.             ARRAT_BOOL(6)=False
187.             ARRAT_BOOL(40)=True
188.                 ARRAY_VISIONE2(9)=10
189.             Else
190.                 ARRAY_VISIONE2(9)=0
191.             End If
192.
193.         if ARRAY_VISIONE2(4)= 1210 and ARRAY_VISIONE2(3)= 1305 then
194.             ARRAT_BOOL(11)=True
195.             ARRAT_BOOL(40)=False
196.         End If
197.     End If
198. End If
199.
200.
```



```
201.    'ENTRATA MACCHINA 6
202.    if I00=True and ARRAT_BOOL(37)=True then
203.        if ARRAT_BOOL(9)=False then
204.            if ARRAY_VISIONE2(4)<= 1200 then
205.                ARRAY_VISIONE2(8)=10
206.            Else
207.                ARRAY_VISIONE2(8)=0
208.            End If
209.
210.        if ARRAY_VISIONE2(4)= 1210 and ARRAY_VISIONE2(3)<= 1500 then
211.            ARRAT_BOOL(6)=False
212.            ARRAT_BOOL(40)=True
213.            ARRAY_VISIONE2(9)=10
214.        Else
215.            ARRAY_VISIONE2(9)=0
216.        End If
217.
218.        if ARRAY_VISIONE2(4)= 1210 and ARRAY_VISIONE2(3)= 1505 then
219.            ARRAT_BOOL(12)=True
220.            ARRAT_BOOL(40)=False
221.        End If
222.    End If
223. End If
224.
225. 'USCITA MACCHINA 1
226. If ARRAT_BOOL(41)=True then
227.     ARRAY_VISIONE2(8)=-10
228.
229.     if ARRAY_VISIONE2(4)<= 1600 Then
230.         ARRAY_VISIONE2(3)=1375
231.         ARRAY_VISIONE2(4)=2000
232.         ARRAY_VISIONE2(8)=0
233.         ARRAT_BOOL(41)=False
234.     End If
235. End If
236.
237. 'USCITA MACCHINA 2
238. If ARRAT_BOOL(42)=True then
239.     ARRAY_VISIONE2(8)=-10
240.
241.     if ARRAY_VISIONE2(4)<= 1600 Then
242.         ARRAY_VISIONE2(3)=1375
243.         ARRAY_VISIONE2(4)=2000
244.         ARRAY_VISIONE2(8)=0
245.         ARRAT_BOOL(42)=False
246.     End If
247. End If
248.
249. 'USCITA MACCHINA 3
250. If ARRAT_BOOL(43)=True then
251.     ARRAY_VISIONE2(8)=-10
252.
253.     if ARRAY_VISIONE2(4)<= 1600 Then
254.         ARRAY_VISIONE2(3)=1375
255.         ARRAY_VISIONE2(4)=2000
256.         ARRAY_VISIONE2(8)=0
257.         ARRAT_BOOL(43)=False
258.     End If
259. End If
260.
261. 'USCITA MACCHINA 4
262. If ARRAT_BOOL(44)=True then
263.     ARRAY_VISIONE2(8)=10
```



```
264.  
265.    if ARRAY_VISIONE2(4)= 1300 Then  
266.        ARRAY_VISIONE2(3)=1375  
267.        ARRAY_VISIONE2(4)=2000  
268.        ARRAY_VISIONE2(8)=0  
269.        ARRAT_BOOL(44)=False  
270.    End If  
271. End If  
272.  
273. 'USCITA MACCHINA 5  
274. If ARRAT_BOOL(45)=True then  
275.     ARRAY_VISIONE2(8)=10  
276.  
277.     if ARRAY_VISIONE2(4)= 1300 Then  
278.         ARRAY_VISIONE2(3)=1375  
279.         ARRAY_VISIONE2(4)=2000  
280.         ARRAY_VISIONE2(8)=0  
281.         ARRAT_BOOL(45)=False  
282.     End If  
283. End If  
284.  
285. 'USCITA MACCHINA 6  
286. If ARRAT_BOOL(46)=True then  
287.     ARRAY_VISIONE2(8)=10  
288.  
289.     if ARRAY_VISIONE2(4)= 1300 Then  
290.         ARRAY_VISIONE2(3)=1375  
291.         ARRAY_VISIONE2(4)=2000  
292.         ARRAY_VISIONE2(8)=0  
293.         ARRAT_BOOL(46)=False  
294.     End If  
295. End If  
296.  
297.  
298. If Q25=True Then  
299.     ARRAY_VISIONE2(8)=10  
300.     ARRAT_BOOL(6)=True  
301. End If  
302.  
303. If ARRAY_VISIONE2(4)=3000 Then  
304.     ARRAT_BOOL(6)=False  
305. End If
```



Controllo livello di un serbatoio

Descrizione

Il pannello permette di simulare un impianto per il riempimento di un serbatoio così configurato:

- vasca di carico alimentata attraverso l'elettrovalvola YV1 e dotata di due sensori di livello con contatti NC per l'indicazione di livello massimo (SL1) e livello minimo (SL2);
- pompa P1 che, pescando l'acqua nella vasca di carico, alimenta il serbatoio; il motore della pompa è alimentato attraverso un contattore KM1 provvisto di relè termico di protezione FR1. La simulazione dell'intervento del relè termico può essere effettuata con un clic del mouse sull'immagine del motore che cambierà il suo colore da verde (scuro quando non è alimentato, chiaro quando è alimentato) a rosso. Un successivo clic sull'immagine del motore ripristina il relè termico;
- serbatoio, alimentato dalla pompa P1, provvisto di tre sensori di livello con contatti NC che indicano rispettivamente: livello Minimo (SL3), livello 1/2 (SL4) e livello Max (SL5);
- lo scarico del serbatoio viene effettuato mediante l'elettrovalvola YV2.

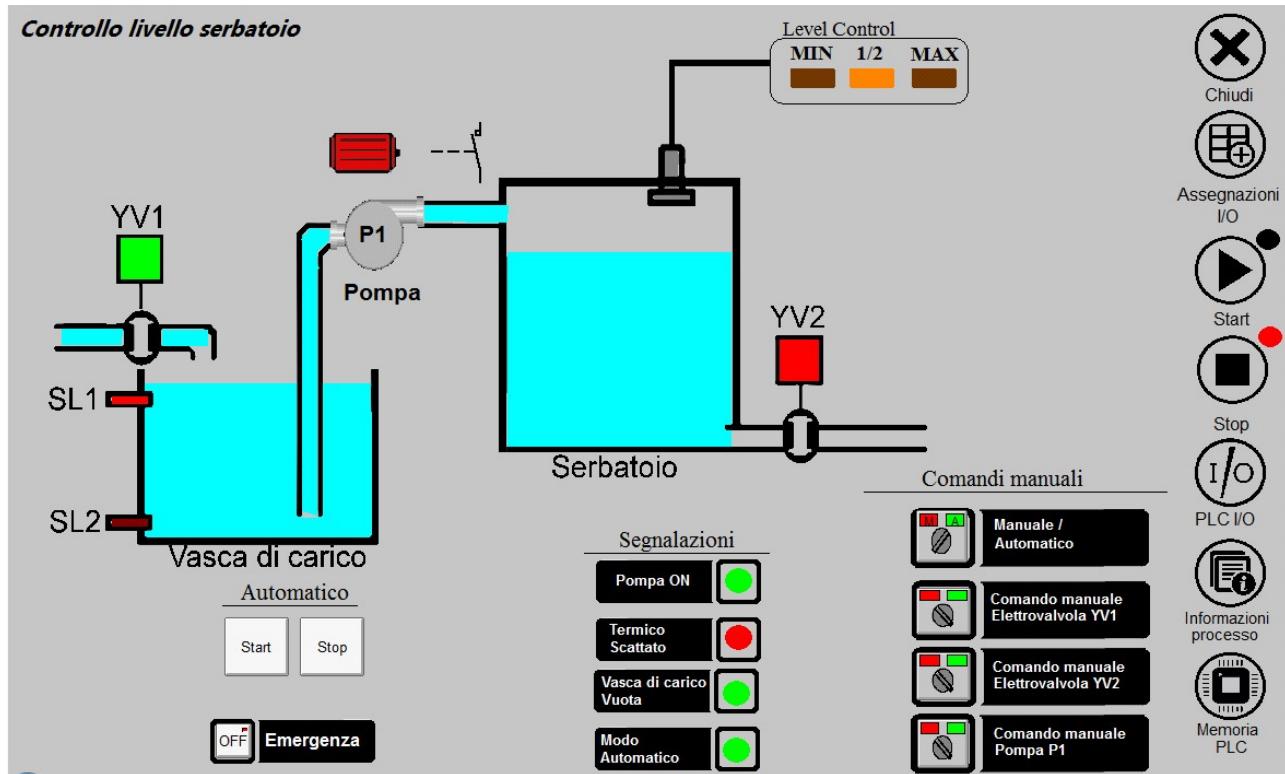
Per il comando e le segnalazioni del controllo sono disponibili:

- quattro selettori ON/OFF (SB2, SB3, SB4, SB7),
- due pulsanti NC (SB1, SB5),
- un pulsante NO (SB6),
- quattro lampade di vari colori.



Chiudi

Pagina



Script

Utilizzato per il controllo delle variabili sia in ingresso che in uscita ma anche per la gestione della visualizzazione dei due livelli dei due serbatoi.

```

1. If ARRAY_VISIONE2(10)=300 Then
2. I01= True
3. Colour "CONTROLLO LIVELLO SERBATOIO", "SL2", &H20000ff
4. Else
5. I01= False

```



```
6. Colour "CONTROLLO LIVELLO SERBATOIO", "SL2", &H2000080
7. End If
8.
9. If ARRAY_VISIONE2(10)>= 2800 Then
10. I00= True
11. Colour "CONTROLLO LIVELLO SERBATOIO", "SL1", &H20000ff
12. Else
13. I00= False
14. Colour "CONTROLLO LIVELLO SERBATOIO", "SL1", &H2000080
15. End If
16.
17. 'MIN
18. If ARRAY_VISIONE2(11)=200 Then
19. Colour "CONTROLLO LIVELLO SERBATOIO", "MIN", &H20080ff
20. I03= True
21. Else
22. Colour "CONTROLLO LIVELLO SERBATOIO", "MIN", &H200376f
23. I03= False
24. End If
25.
26.
27. 'META
28. If ARRAY_VISIONE2(11)>1500 Then
29. Colour "CONTROLLO LIVELLO SERBATOIO", "META", &H20080ff
30. I04= True
31. Else
32. Colour "CONTROLLO LIVELLO SERBATOIO", "META", &H200376f
33. I04= False
34. End If
35.
36. 'MAX
37. If ARRAY_VISIONE2(11)>= 2900 Then
38. Colour "CONTROLLO LIVELLO SERBATOIO", "MAX", &H20080ff
39. I05= True
40. Else
41. Colour "CONTROLLO LIVELLO SERBATOIO", "MAX", &H200376f
42. I05= False
43. End If
44.
45. 'CARICA
46. ARRAY_VISIONE2(10)=ARRAY_VISIONE2(10)+ARRAY_VISIONE2(12)
47.
48. If Q20= True AND Q21=FALSE Then
49. ARRAY_VISIONE2(12)=100
50. Else If Q20=FALSE AND Q21= True Then
51. ARRAY_VISIONE2(12)=-50
52. Else If Q20= True AND Q21= True Then
53. ARRAY_VISIONE2(12)=50
54. Else
55. ARRAY_VISIONE2(12)=0
56. End If
57. End If
58.
59. End If
60.
61.
62. 'SCARICA
63. ARRAY_VISIONE2(11)=ARRAY_VISIONE2(11)+ARRAY_VISIONE2(13)
64.
65. If Q22= True AND Q21=FALSE Then
66. ARRAY_VISIONE2(13)=-25
67. Else If Q22=FALSE AND Q21= True Then
68. ARRAY_VISIONE2(13)=25
```



69. Else

70. ARRAY_VISIONE2(13)=0

71. End If

72. End If



Miscelatore di due liquidi

Descrizione

Il pannello permette di simulare un impianto di miscelazione di due liquidi; è composto da:

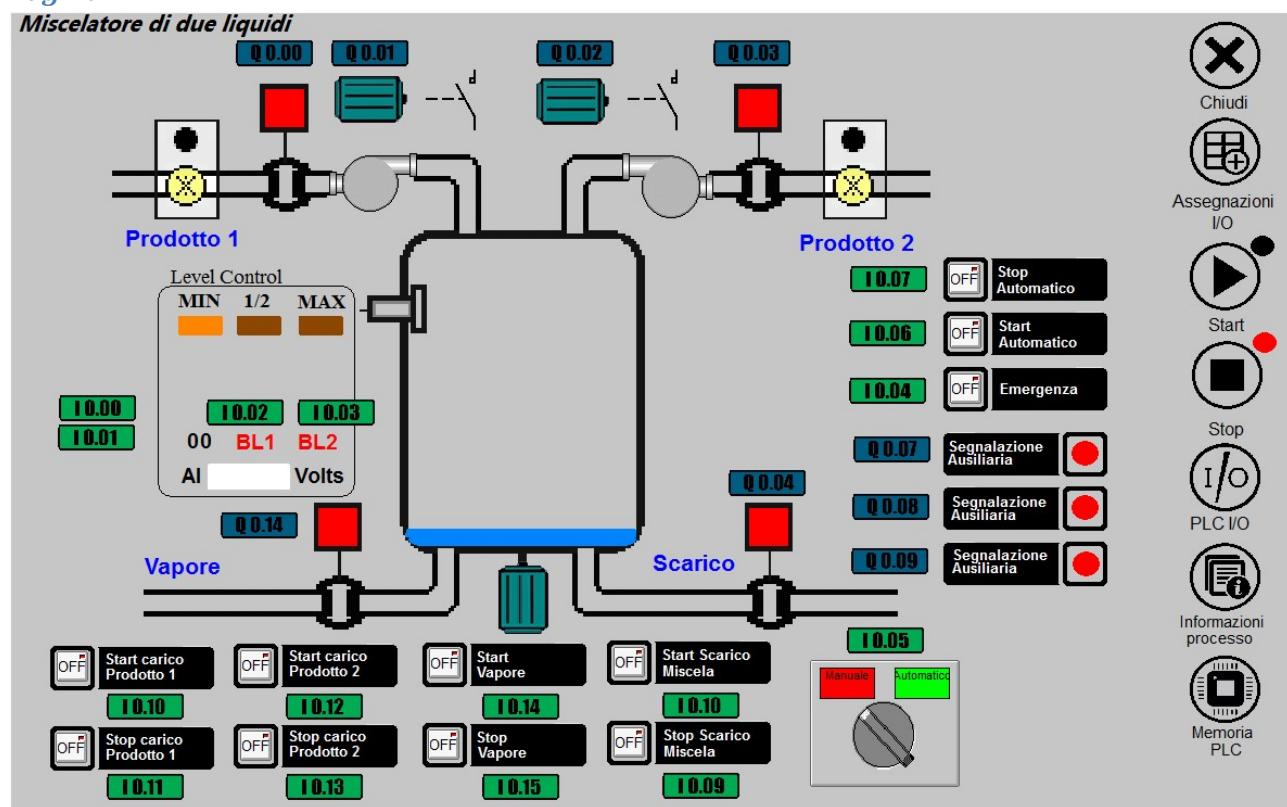
- un serbatoio di raccolta e miscelazione del prodotto
- un agitatore del liquido azionato dal motore M3;
- un indicatore di livello digitale che fornisce su due bit i livelli: serbatoio vuoto, serbatoio metà e serbatoio pieno secondo gli stati logici della seguente tabella:
- una condotta che immette vapore per il riscaldamento del liquido, controllata dall'elettrovalvola YV4;
- una condotta di scarico del liquido controllata dall'elettrovalvola YV3;
- due condotte per il carico, separato, dei due liquidi da miscelare; ogni condotta è controllata da un'elettrovalvola (YV1, YV2) e da una pompa (P1, P2). Su ogni condotta è installato un trasduttore digitale di flusso che fornisce un impulso per ogni litro di prodotto che lo attraversa;
- una serie di pulsanti e selettori per il comando dell'impianto; per l'uso dei singoli elementi si può fare riferimento alla etichette di assegnazione simboli.



Chiudi

Pagina

Miscelatore di due liquidi



Script

Script legato alle segnalazioni dei vari trasduttori di livello.

```

1. 'MIN
2. IF ARRAY_VISIONE2(14)<0 THEN
3. Colour "MISCELATORE DI DUE LIQUIDI", "MIN", &H20080ff
4. Else
5. Colour "MISCELATORE DI DUE LIQUIDI", "MIN", &H200458a
6. End If
7.
8. 'META'
9. IF ARRAY_VISIONE2(14)>= 1400 and ARRAY_VISIONE2(14)<= 1600 THEN
10. Colour "MISCELATORE DI DUE LIQUIDI", "META", &H20080ff
11. Else
12. Colour "MISCELATORE DI DUE LIQUIDI", "META", &H200458a
13. End If

```



```
14.  
15. 'MAX  
16. IF ARRAY_VISIONE2(14)>= 3000 THEN  
17. Colour "MISCELATORE DI DUE LIQUIDI", "MAX", &H20080ff  
18. Else  
19. Colour "MISCELATORE DI DUE LIQUIDI", "MAX", &H200458a  
20. End If  
21.  
22. ARRAY_VISIONE2(14)=ARRAY_VISIONE2(14)+ARRAY_VISIONE2(15)  
23.  
24. IF Q20=TRUE AND Q23=FALSE THEN  
25. ARRAY_VISIONE2(15)=1  
26.  
27. End If  
28.  
29. IF Q20=FALSE AND Q23=TRUE THEN  
30. ARRAY_VISIONE2(15)=1  
31.  
32. End If  
33.  
34. IF Q20=TRUE AND Q23=TRUE THEN  
35. ARRAY_VISIONE2(15)=2  
36.  
37. End If  
38.  
39. IF Q20=FALSE AND Q23=FALSE THEN  
40. ARRAY_VISIONE2(15)=0  
41. End If  
42.  
43. IF Q24 THEN  
44. ARRAY_VISIONE2(15)=-1  
45. End If  
46.  
47.  
48. 'trasduttori  
49. IF ARRAY_VISIONE2(14)<= 0 then  
50. I02=False  
51. I03=False  
52. END IF  
53.  
54. IF ARRAY_VISIONE2(14)>-300 AND ARRAY_VISIONE2(14)<= 1400 then  
55. I02=False  
56. I03=True  
57. END IF  
58.  
59. IF ARRAY_VISIONE2(14)>1400 AND ARRAY_VISIONE2(14)<= 2800 then  
60. I02=True  
61. I03=False  
62. END IF  
63.  
64. IF ARRAY_VISIONE2(14)>2800 then  
65. I02=True  
66. I03=True  
67. END IF  
68.  
69. If Q20 Then  
70. I00=True  
71. else  
72. I00=False  
73. End IF  
74.  
75. If Q23 Then  
76. I01=True
```



77. **else**
78. I01=False
79. End IF



Ciclo elettropneumatico con un cilindro a doppio effetto

Descrizione

Il pannello mette a disposizione un cilindro pneumatico a doppio effetto (A) e un'elettrovalvola 5/2 bistabile (1V1) per il suo controllo.

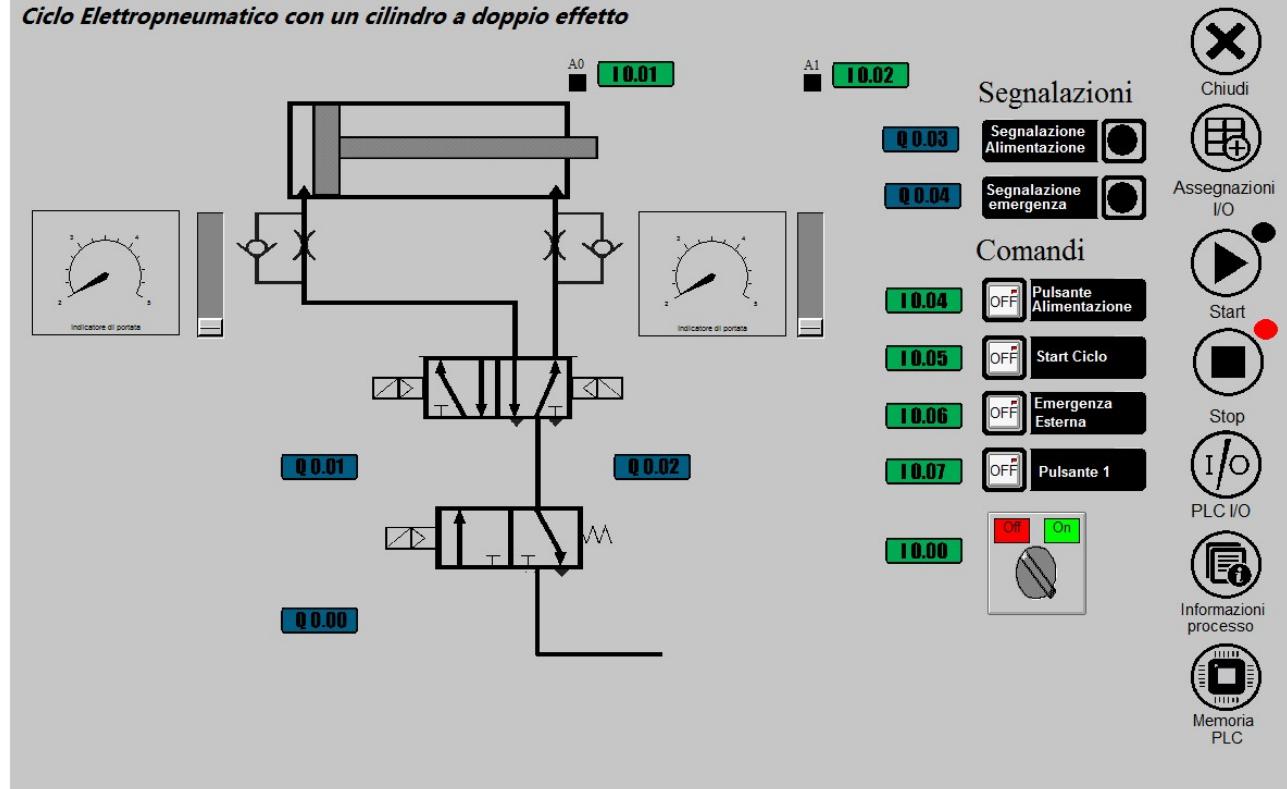
Nel circuito è presente anche un'elettrovalvola del tipo 3/2 monostabile (0V) che comanda la distribuzione dell'aria nel circuito. Sulle mandate di alimentazione del cilindro A sono presenti due strozzatori unidirezionali per la regolazione della velocità delle due corse (A+ e A-) del pistone. La regolazione avviene, impostando un numero sugli slider. Chiudi

Per rilevare la posizione del pistone sono presenti due microinterruttori magnetici (a0, a1) con contatti NO.



Pagina

Ciclo Elettropneumatico con un cilindro a doppio effetto



Script

```

1. If Q20 Then
2. ARRAY_VISIONE2(0)=1550
3. End If
4.
5. If Q21 Then
6. ARRAY_VISIONE2(1)=1325
7. else
8. ARRAY_VISIONE2(1)=0
9. End If
10.
11. If ARRAY_VISIONE2(2) <= 0 Then
12. I01=True
13. Else
14. I01=False
15. End If
16.

```



```
17. If ARRAY_VISIONE2(2) >= 38 Then
18. I02=True
19. Else
20. I02=False
21. End If
22.
23. If Q21 and I02=False Then
24. ARRAY_VISIONE2(2) = ARRAY_VISIONE2(2) + ARRAY_VISIONE2(16)
25. End If
26.
27. If Q22 and I01=False Then
28. ARRAY_VISIONE2(2) = ARRAY_VISIONE2(2) - ARRAY_VISIONE2(17)
29. End If
```

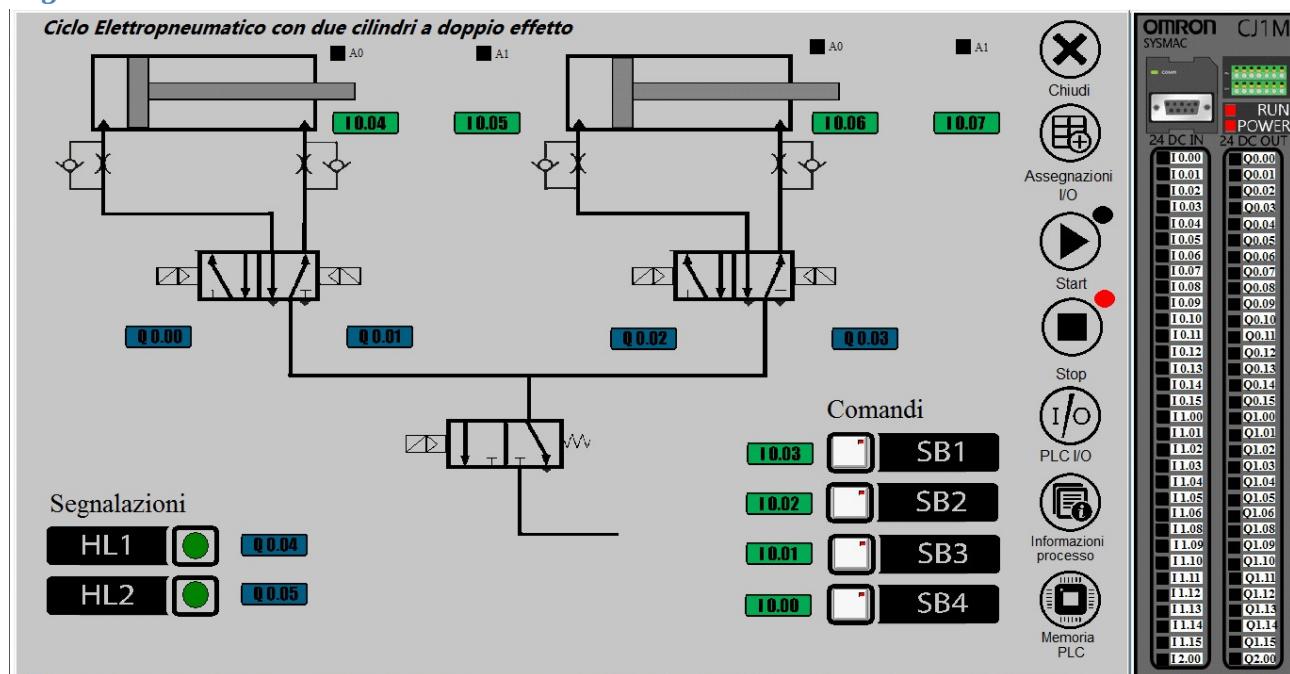


Ciclo elettropneumatico con due cilindri a doppio effetto

Descrizione

Il processo serve per descrivere simulare due pistoni a doppio effetto comandate con valvole bistabili.

Pagina

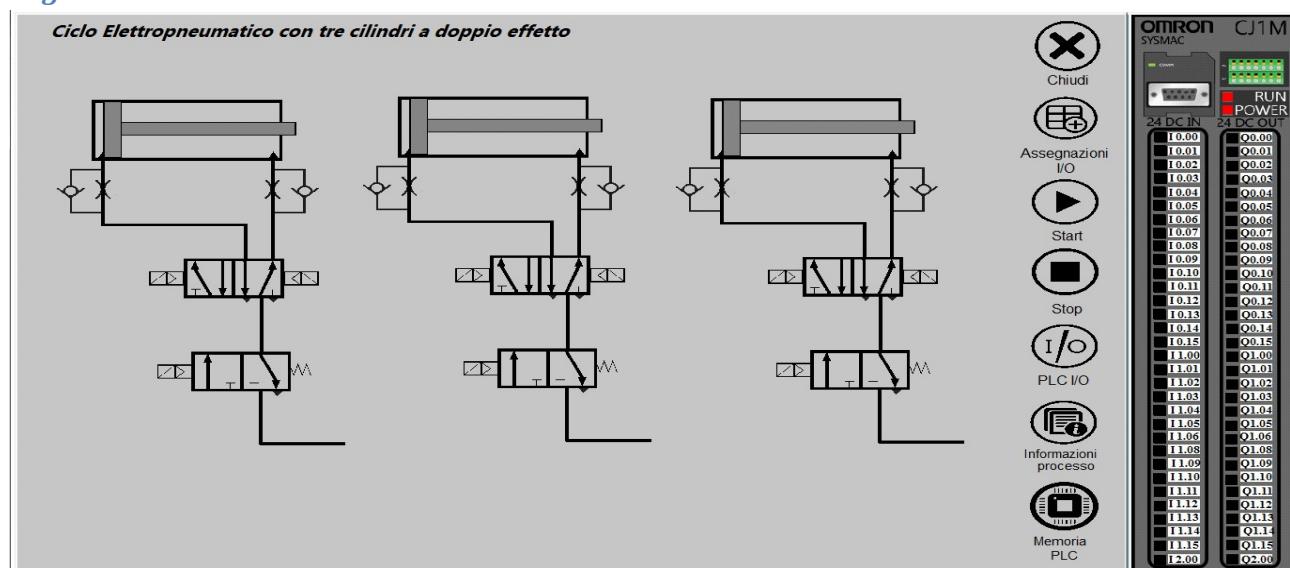


Ciclo elettropneumatico con tre cilindri a doppio effetto

Descrizione

Simula come nei casi precedenti un ciclo elettropneumatico con tre pistoni

Pagina



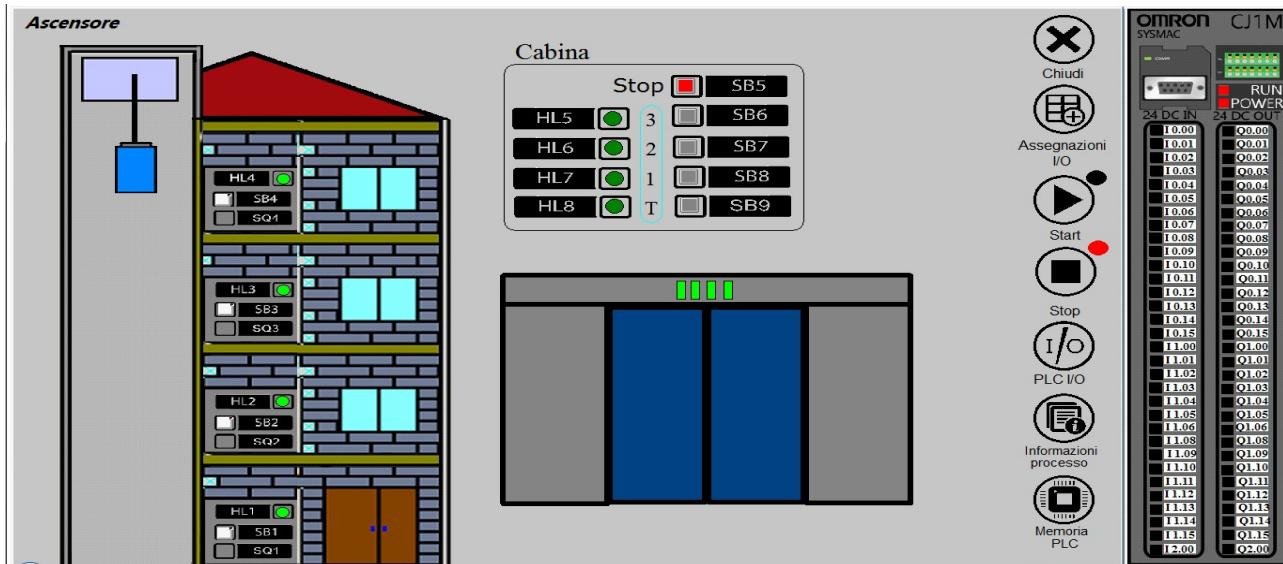


Ascensore

Descrizione

Serve a simulare un ascensore didattico a 4 piani.

Pagina





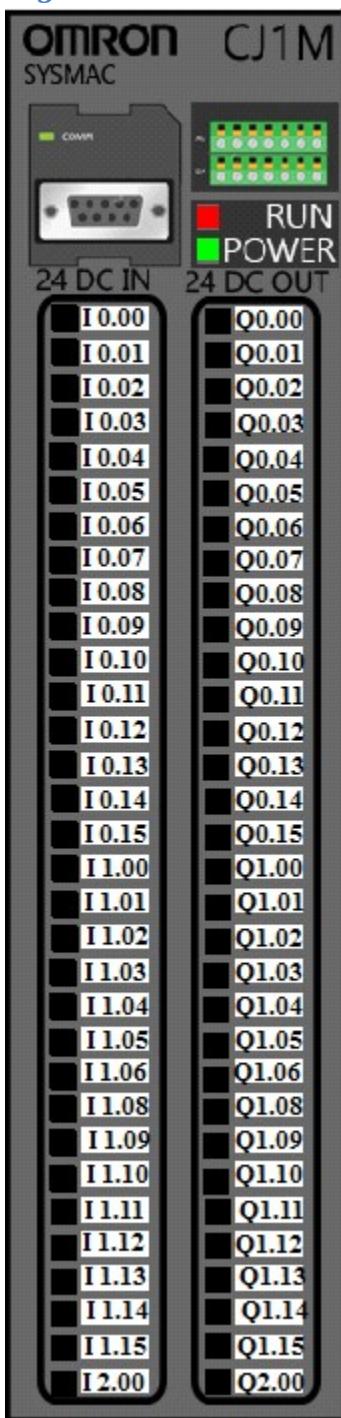
Plc ingressi uscite

Descrizione

La pagina serve per poter vedere velocemente quali ingressi e quali uscite sono attive nel momento della visualizzazione.

Sono state anche simulate la fase di run o stop della cpu e dell'alimentazione(power) de plc.

Pagina





CIO

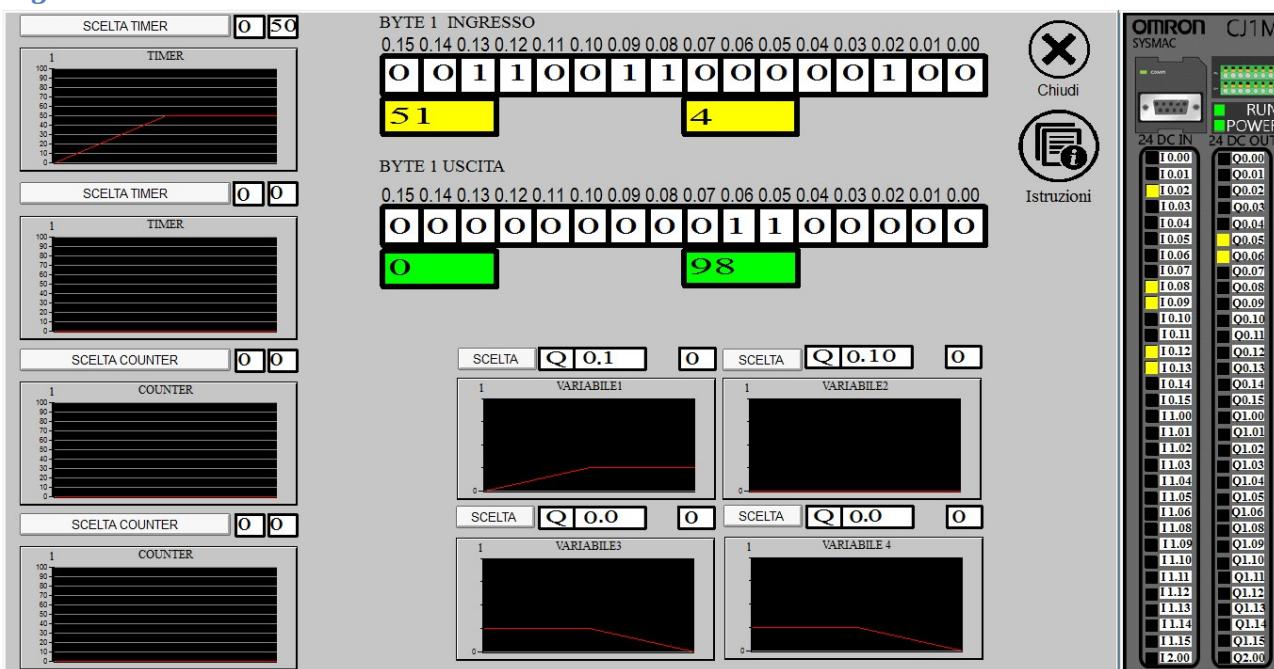
Descrizione

La pagina viene utilizzata per monitorare lo stato dei timer dei contatori tramite un grafico.

Nella pagina si possono vedere rappresentati gli ingressi e le uscite suddivisi in byte con la rappresentazione sia in binario che in decimale.

Nel fondo pagina ci sono quattro grafici che permettono di vedere un ingresso o un uscita alla volta; questo è possibile tramite una scelta prima se si vuole visualizzare lo stato degli input o degli output poi inserendo nell'apposito spazio un numero si va a decidere l'indirizzo della variabile da visualizzare.

Pagina



Script

```

1. ARAY_VISIONE(16)= ARRAY_VISIONE(21)+ ARRAY_VISIONE(22)+ARRAY_VISIONE(23)+ARRAY_VISIONE(24)
+ARRAY_VISIONE(25)+ARRAY_VISIONE(26)+ARRAY_VISIONE(27)+ARRAY_VISIONE(28)
2.
3. ARAY_VISIONE(17)= ARRAY_VISIONE(29)+ ARRAY_VISIONE(30)+ARRAY_VISIONE(31)+ARRAY_VISIONE(32)
+ARRAY_VISIONE(33)+ARRAY_VISIONE(34)+ARRAY_VISIONE(35)+ARRAY_VISIONE(36)
4.
5. ARAY_VISIONE(18)= ARRAY_VISIONE(37)+ ARRAY_VISIONE(38)+ARRAY_VISIONE(39)+ARRAY_VISIONE(40)
+ARRAY_VISIONE(41)+ARRAY_VISIONE(42)+ARRAY_VISIONE(43)+ARRAY_VISIONE(44)
6.
7. ARAY_VISIONE(19)= ARRAY_VISIONE(45)+ ARRAY_VISIONE(46)+ARRAY_VISIONE(47)+ARRAY_VISIONE(48)
+ARRAY_VISIONE(49)+ARRAY_VISIONE(50)+ARRAY_VISIONE(51)+ARRAY_VISIONE(52)
8.
9. 'CONVERSIONE INGRESSI PER BYTE1
10.
11. If I00 =True Then
12.     ARRAY_VISIONE(21)= 1
13. Else
14.     ARRAY_VISIONE(21)= 0
15. End If
16.

```



```

17. If I01 =True Then
18.     ARRAY_VISIONE(22)= 2
19. Else
20.     ARRAY_VISIONE(22)= 0
21. End If
22.
23. If I02 =True Then
24.     ARRAY_VISIONE(23)= 4
25. Else
26.     ARRAY_VISIONE(23)= 0
27. End If
28.
29. If I03 =True Then
30.     ARRAY_VISIONE(24)= 8
31. Else
32.     ARRAY_VISIONE(24)= 0
33. End If
34.
35. If I04 =True Then
36.     ARRAY_VISIONE(25)= 16
37. Else
38.     ARRAY_VISIONE(25)= 0
39. End If
40.
41. If I05 =True Then
42.     ARRAY_VISIONE(26)= 32
43. Else
44.     ARRAY_VISIONE(26)= 0
45. End If
46.
47. If I06 =True Then
48.     ARRAY_VISIONE(27)= 64
49. Else
50.     ARRAY_VISIONE(27)= 0
51. End If
52.
53. If I07 =True Then
54.     ARRAY_VISIONE(28)= 128
55. Else
56.     ARRAY_VISIONE(28)= 0
57. End If
58.
59. 'CONVERSIONE INGRESSI PER BYTE2
60.
61. If Start =True Then
62.     ARRAY_VISIONE(29)= 1
63. Else
64.     ARRAY_VISIONE(29)= 0
65. End If
66.
67. If Power =True Then
68.     ARRAY_VISIONE(30)= 2
69. Else
70.     ARRAY_VISIONE(30)= 0
71. End If
72.
73. If I10 =True Then
74.     ARRAY_VISIONE(31)= 4
75. Else
76.     ARRAY_VISIONE(31)= 0
77. End If
78.
79. If I11 =True Then

```



```

80.    ARRAY_VISIONE(32)= 8
81. Else
82.    ARRAY_VISIONE(32)= 0
83. End If
84.
85. If I12 =True Then
86.    ARRAY_VISIONE(33)= 16
87. Else
88.    ARRAY_VISIONE(33)= 0
89. End If
90.
91. If I13 =True Then
92.    ARRAY_VISIONE(34)= 32
93. Else
94.    ARRAY_VISIONE(34)= 0
95. End If
96.
97. If I14 =True Then
98.    ARRAY_VISIONE(35)= 64
99. Else
100.   ARRAY_VISIONE(35)= 0
101. End If
102.
103. If I15 =True Then
104.    ARRAY_VISIONE(36)= 128
105. Else
106.    ARRAY_VISIONE(36)= 0
107. End If
108.
109. 'CONVERSIONE USCITE PER BYTES
110.
111. If Q20 =True Then
112.    ARRAY_VISIONE(37)= 1
113. Else
114.    ARRAY_VISIONE(37)= 0
115. End If
116.
117. If Q21 =True Then
118.    ARRAY_VISIONE(38)= 2
119. Else
120.    ARRAY_VISIONE(38)= 0
121. End If
122.
123. If Q22 =True Then
124.    ARRAY_VISIONE(39)= 4
125. Else
126.    ARRAY_VISIONE(39)= 0
127. End If
128.
129. If Q23 =True Then
130.    ARRAY_VISIONE(40)= 8
131. Else
132.    ARRAY_VISIONE(40)= 0
133. End If
134.
135. If Q24 =True Then
136.    ARRAY_VISIONE(41)= 16
137. Else
138.    ARRAY_VISIONE(41)= 0
139. End If
140.
141. If Q25 =True Then
142.    ARRAY_VISIONE(42)= 32

```



```

143. Else
144.     ARRAY_VISIONE(42)= 0
145. End If
146.
147. If Q26 =True Then
148.     ARRAY_VISIONE(43)= 64
149. Else
150.     ARRAY_VISIONE(43)= 0
151. End If
152.
153. If Q27 =True Then
154.     ARRAY_VISIONE(44)= 128
155. Else
156.     ARRAY_VISIONE(44)= 0
157. End If
158.
159. 'CONVERSIONE USCITE PER BYTE4
160.
161. If Q28 =True Then
162.     ARRAY_VISIONE(445)= 1
163. Else
164.     ARRAY_VISIONE(45)= 0
165. End If
166.
167. If Q29 =True Then
168.     ARRAY_VISIONE(46)= 2
169. Else
170.     ARRAY_VISIONE(46)= 0
171. End If
172.
173. If Q30 =True Then
174.     ARRAY_VISIONE(47)= 4
175. Else
176.     ARRAY_VISIONE(47)= 0
177. End If
178.
179. If Q31 =True Then
180.     ARRAY_VISIONE(48)= 8
181. Else
182.     ARRAY_VISIONE(48)= 0
183. End If
184.
185. If Q32 =True Then
186.     ARRAY_VISIONE(49)= 16
187. Else
188.     ARRAY_VISIONE(49)= 0
189. End If
190.
191. If Q33 =True Then
192.     ARRAY_VISIONE(50)= 32
193. Else
194.     ARRAY_VISIONE(50)= 0
195. End If
196.
197. If Q34 =True Then
198.     ARRAY_VISIONE(51)= 64
199. Else
200.     ARRAY_VISIONE(51)= 0
201. End If
202.
203. If Q35 =True Then
204.     ARRAY_VISIONE(52)= 128
205. Else

```



```

206.    ARRAY_VISIONE(52)= 0
207. End If
208.
209. REM VISIONE PER IL PRIMO TIMER1
210.
211. Select Case ARRAY_VISIONE(0)
212. Case 1:
213.    ARRAY_VISIONE(1) = TIMERIN1
214. Case 2:
215.    ARRAY_VISIONE(1) = TIMERIN2
216. Case 3:
217.    ARRAY_VISIONE(1) = TIMERIN3
218. Case 4:
219.    ARRAY_VISIONE(1) = TIMERIN4
220. Case 5:
221.    ARRAY_VISIONE(1) = TIMERIN5
222. Case 6:
223.    ARRAY_VISIONE(1) = TIMERIN6
224. Case Else
225.    ARRAY_VISIONE(1) = TIMERIN1
226. End Select
227.
228.
229. REM VISIONE PER IL SECONDO TIMER2
230.
231. Select Case ARRAY_VISIONE(2)
232. Case 1:
233.    ARRAY_VISIONE(3) = TIMERIN1
234. Case 2:
235.    ARRAY_VISIONE(3) = TIMERIN2
236. Case 3:
237.    ARRAY_VISIONE(3) = TIMERIN3
238. Case 4:
239.    ARRAY_VISIONE(3) = TIMERIN4
240. Case 5:
241.    ARRAY_VISIONE(3) = TIMERIN5
242. Case 6:
243.    ARRAY_VISIONE(3) = TIMERIN6
244. Case Else
245.    ARRAY_VISIONE(3) = TIMERIN2
246. End Select
247.
248.
249. REM VISIONE PER IL PRIMO COUNTER1
250.
251. Select Case ARRAY_VISIONE(4)
252. Case 1:
253.    ARRAY_VISIONE(5) = COUNTERIN1
254. Case 2:
255.    ARRAY_VISIONE(5) = COUNTERIN2
256. Case 3:
257.    ARRAY_VISIONE(5) = COUNTERIN3
258. Case 4:
259.    ARRAY_VISIONE(5) = COUNTERIN4G
260. Case 5:
261.    ARRAY_VISIONE(5) = COUNTERIN5
262. Case 6:
263.    ARRAY_VISIONE(5) = COUNTERIN6
264. Case Else
265.    ARRAY_VISIONE(5) = COUNTERIN1
266. End Select
267.
268. REM VISIONE PER IL SECONDO COUNTER2

```



```

269.
270. Select Case ARRAY_VISIONE(6)
271. Case 1:
272.     ARRAY_VISIONE(7) = COUNTERIN1
273. Case 2:
274.     ARRAY_VISIONE(7) = COUNTERIN2
275. Case 3:
276.     ARRAY_VISIONE(7) = COUNTERIN3
277. Case 4:
278.     ARRAY_VISIONE(7) = COUNTERIN4G
279. Case 5:
280.     ARRAY_VISIONE(7) = COUNTERIN5
281. Case 6:
282.     ARRAY_VISIONE(7) = COUNTERIN6
283. Case Else
284.     ARRAY_VISIONE(7) = COUNTERIN2
285. End Select
286.
287.
288. REM VISUALIZZAZIONE VARIABILE 1
289.
290. If ARRAT_BOOL(1) Then
291.     ARRAY_VISIONE(53)= 0
292.     Select Case ARRAY_VISIONE(8)
293.     Case 0:
294.         ARRAY_VISIONE(9) = I00
295.     Case 1:
296.         ARRAY_VISIONE(9) = I01
297.     Case 2:
298.         ARRAY_VISIONE(9) = I02
299.     Case 3:
300.         ARRAY_VISIONE(9) = I03
301.     Case 4:
302.         ARRAY_VISIONE(9) = I04
303.     Case 5:
304.         ARRAY_VISIONE(9) = I05
305.     Case 6:
306.         ARRAY_VISIONE(9) = I06
307.     Case 7:
308.         ARRAY_VISIONE(9) = I07
309.     Case 8:
310.         ARRAY_VISIONE(9) = Start
311.     Case 9:
312.         ARRAY_VISIONE(9) = I09
313.     Case 10:
314.         ARRAY_VISIONE(9) = I10
315.     Case 11:
316.         ARRAY_VISIONE(9) = I11
317.     Case 12:
318.         ARRAY_VISIONE(9) = I12
319.     Case 13:
320.         ARRAY_VISIONE(9) = I13
321.     Case 14:
322.         ARRAY_VISIONE(9) = I14
323.     Case 15:
324.         ARRAY_VISIONE(9) = I15
325.     Case Else
326.         ARRAY_VISIONE(9) = I00
327.     End Select
328. Else
329.     ARRAY_VISIONE(53)= 0
330.     Select Case ARRAY_VISIONE(8)
331.     Case 0:

```



```
332.      ARRAY_VISIONE(9) = Q20
333.  Case 1:
334.      ARRAY_VISIONE(9) = Q21
335.  Case 2:
336.      ARRAY_VISIONE(9) = Q22
337.  Case 3:
338.      ARRAY_VISIONE(9) = Q23
339.  Case 4:
340.      ARRAY_VISIONE(9) = Q24
341.  Case 5:
342.      ARRAY_VISIONE(9) = Q25
343.  Case 6:
344.      ARRAY_VISIONE(9) = Q26
345.  Case 7:
346.      ARRAY_VISIONE(9) = Q27
347.  Case 8:
348.      ARRAY_VISIONE(9) = Q28
349.  Case 9:
350.      ARRAY_VISIONE(9) = Q29
351.  Case 10:
352.      ARRAY_VISIONE(9) = Q30
353.  Case 11:
354.      ARRAY_VISIONE(9) = Q31
355.  Case 12:
356.      ARRAY_VISIONE(9) = Q32
357.  Case 13:
358.      ARRAY_VISIONE(9) = Q33
359.  Case 14:
360.      ARRAY_VISIONE(9) = Q34
361.  Case 15:
362.      ARRAY_VISIONE(9) = Q35
363.  Case Else
364.      ARRAY_VISIONE(9) = Q20
365. End Select
366. End If
367.
368.
369. REM VISUALIZZAZIONE VARIABILE 2
370.
371.
372. If ARRAT_BOOL(2) Then
373.     ARRAY_VISIONE(54)= 0
374.     Select Case ARRAY_VISIONE(10)
375.     Case 0:
376.         ARRAY_VISIONE(11) = I00
377.     Case 1:
378.         ARRAY_VISIONE(11) = I01
379.     Case 2:
380.         ARRAY_VISIONE(11) = I02
381.     Case 3:
382.         ARRAY_VISIONE(11) = I03
383.     Case 4:
384.         ARRAY_VISIONE(11) = I04
385.     Case 5:
386.         ARRAY_VISIONE(11) = I05
387.     Case 6:
388.         ARRAY_VISIONE(11) = I06
389.     Case 7:
390.         ARRAY_VISIONE(11) = I07
391.     Case 8:
392.         ARRAY_VISIONE(11) = I08
393.     Case 9:
394.         ARRAY_VISIONE(11) = I09
```



```

395. Case 10:
396.     ARRAY_VISIONE(11) = I10
397. Case 11:
398.     ARRAY_VISIONE(11) = I11
399. Case 12:
400.     ARRAY_VISIONE(11) = I12
401. Case 13:
402.     ARRAY_VISIONE(11) = I13
403. Case 14:
404.     ARRAY_VISIONE(11) = I14
405. Case 15:
406.     ARRAY_VISIONE(11) = I15
407. Case Else
408.     ARRAY_VISIONE(11) = I00
409. End Select
410. Else
411.     ARRAY_VISIONE(54)= 0
412.     Select Case ARRAY_VISIONE(10)
413.     Case 0:
414.         ARRAY_VISIONE(11) = Q20
415.     Case 1:
416.         ARRAY_VISIONE(11) = Q21
417.     Case 2:
418.         ARRAY_VISIONE(11) = Q22
419.     Case 3:
420.         ARRAY_VISIONE(11) = Q23
421.     Case 4:
422.         ARRAY_VISIONE(11) = Q24
423.     Case 5:
424.         ARRAY_VISIONE(11) = Q25
425.     Case 6:
426.         ARRAY_VISIONE(11) = Q26
427.     Case 7:
428.         ARRAY_VISIONE(11) = Q27
429.     Case 8:
430.         ARRAY_VISIONE(11) = Q28
431.     Case 9:
432.         ARRAY_VISIONE(11) = Q29
433.     Case 10:
434.         ARRAY_VISIONE(11) = Q30
435.     Case 11:
436.         ARRAY_VISIONE(11) = Q31
437.     Case 12:
438.         ARRAY_VISIONE(11) = Q32
439.     Case 13:
440.         ARRAY_VISIONE(11) = Q33
441.     Case 14:
442.         ARRAY_VISIONE(11) = Q34
443.     Case 15:
444.         ARRAY_VISIONE(11) = Q35
445. Case Else
446.     ARRAY_VISIONE(11) = Q20
447. End Select
448. End If
449.
450.
451.
452. REM VISUALIZZAZIONE VARIABILE 3
453.
454.
455. If ARRAT_BOOL(3) Then
456.     ARRAY_VISIONE(55)= 0
457.     Select Case ARRAY_VISIONE(12)

```



```

458.    Case 0:
459.        ARRAY_VISIONE(13) = I00
460.    Case 1:
461.        ARRAY_VISIONE(13) = I01
462.    Case 2:
463.        ARRAY_VISIONE(13) = I02
464.    Case 3:
465.        ARRAY_VISIONE(13) = I03
466.    Case 4:
467.        ARRAY_VISIONE(13) = I04
468.    Case 5:
469.        ARRAY_VISIONE(13) = I05
470.    Case 6:
471.        ARRAY_VISIONE(13) = I06
472.    Case 7:
473.        ARRAY_VISIONE(13) = I07
474.    Case 8:
475.        ARRAY_VISIONE(13) = I08
476.    Case 9:
477.        ARRAY_VISIONE(13) = I09
478.    Case 10:
479.        ARRAY_VISIONE(13) = I10
480.    Case 11:
481.        ARRAY_VISIONE(13) = I11
482.    Case 12:
483.        ARRAY_VISIONE(13) = I12
484.    Case 13:
485.        ARRAY_VISIONE(13) = I13
486.    Case 14:
487.        ARRAY_VISIONE(13) = I14
488.    Case 15:
489.        ARRAY_VISIONE(13) = I15
490.    Case Else
491.        ARRAY_VISIONE(13) = I00
492.    End Select
493. Else
494.     ARRAY_VISIONE(55)= 0
495.     Select Case ARRAY_VISIONE(12)
496.     Case 0:
497.         ARRAY_VISIONE(13) = Q20
498.     Case 1:
499.         ARRAY_VISIONE(13) = Q21
500.     Case 2:
501.         ARRAY_VISIONE(13) = Q22
502.     Case 3:
503.         ARRAY_VISIONE(13) = Q23
504.     Case 4:
505.         ARRAY_VISIONE(13) = Q24
506.     Case 5:
507.         ARRAY_VISIONE(13) = Q25
508.     Case 6:
509.         ARRAY_VISIONE(13) = Q26
510.     Case 7:
511.         ARRAY_VISIONE(13) = Q27
512.     Case 8:
513.         ARRAY_VISIONE(13) = Q28
514.     Case 9:
515.         ARRAY_VISIONE(13) = Q29
516.     Case 10:
517.         ARRAY_VISIONE(13) = Q30
518.     Case 11:
519.         ARRAY_VISIONE(13) = Q31
520.     Case 12:

```



```
521.      ARRAY_VISIONE(13) = Q32
522.  Case 13:
523.      ARRAY_VISIONE(13) = Q33
524.  Case 14:
525.      ARRAY_VISIONE(13) = Q34
526.  Case 15:
527.      ARRAY_VISIONE(13) = Q35
528.  Case Else
529.      ARRAY_VISIONE(13) = Q20
530. End Select
531. End If
532.
533.
534. REM VISUALIZZAZIONE VARIABILE 4
535.
536.
537. If ARRAT_BOOL(4) Then
538.      ARRAY_VISIONE(56)= 0
539.      Select Case ARRAY_VISIONE(14)
540.      Case 0:
541.          ARRAY_VISIONE(15) = I00
542.      Case 1:
543.          ARRAY_VISIONE(15) = I01
544.      Case 2:
545.          ARRAY_VISIONE(15) = I02
546.      Case 3:
547.          ARRAY_VISIONE(15) = I03
548.      Case 4:
549.          ARRAY_VISIONE(15) = I04
550.      Case 5:
551.          ARRAY_VISIONE(15) = I05
552.      Case 6:
553.          ARRAY_VISIONE(15) = I06
554.      Case 7:
555.          ARRAY_VISIONE(15) = I07
556.      Case 8:
557.          ARRAY_VISIONE(15) = Start
558.      Case 9:
559.          ARRAY_VISIONE(15) = I09
560.      Case 10:
561.          ARRAY_VISIONE(15) = I10
562.      Case 11:
563.          ARRAY_VISIONE(15) = I11
564.      Case 12:
565.          ARRAY_VISIONE(15) = I12
566.      Case 13:
567.          ARRAY_VISIONE(15) = I13
568.      Case 14:
569.          ARRAY_VISIONE(15) = I14
570.      Case 15:
571.          ARRAY_VISIONE(15) = I15
572.      Case Else
573.          ARRAY_VISIONE(15) = I00
574. End Select
575. Else
576.      ARRAY_VISIONE(56)= 0
577.      Select Case ARRAY_VISIONE(14)
578.      Case 0:
579.          ARRAY_VISIONE(15) = Q20
580.      Case 1:
581.          ARRAY_VISIONE(15) = Q21
582.      Case 2:
583.          ARRAY_VISIONE(15) = Q22
```



```
584. Case 3:  
585.     ARRAY_VISIONE(15) = Q23  
586. Case 4:  
587.     ARRAY_VISIONE(15) = Q24  
588. Case 5:  
589.     ARRAY_VISIONE(15) = Q25  
590. Case 6:  
591.     ARRAY_VISIONE(15) = Q26  
592. Case 7:  
593.     ARRAY_VISIONE(15) = Q27  
594. Case 8:  
595.     ARRAY_VISIONE(15) = Q28  
596. Case 9:  
597.     ARRAY_VISIONE(15) = Q29  
598. Case 10:  
599.     ARRAY_VISIONE(15) = Q30  
600. Case 11:  
601.     ARRAY_VISIONE(15) = Q31  
602. Case 12:  
603.     ARRAY_VISIONE(15) = Q32  
604. Case 13:  
605.     ARRAY_VISIONE(15) = Q33  
606. Case 14:  
607.     ARRAY_VISIONE(15) = Q34  
608. Case 15:  
609.     ARRAY_VISIONE(15) = Q35  
610. Case Else  
611.     ARRAY_VISIONE(15) = Q20  
612. End Select  
613. End If
```

PARTECIPANTI

STUDENTI: MILANESI MARCO

MIRANDOLA EDOARDO

DOCENTE: TRIMARCHI GIUSEPPE