



Azionamenti e sistemi per l'automazione e la domotica

Luca Sani

Dipartimento di Sistemi Elettrici e Automazione

Università di Pisa

tel. 050 2217364

email luca.sani@dsea.unipi.it

IEC 1131-3: Functional Block Diagram

Caratteristiche Principali:

- Linguaggio grafico
- Si basa sulla rappresentazione delle funzioni logiche mediante porte logiche elementari
- Si tratta di un linguaggio a più alto livello (in termini rappresentativi) rispetto al IL e quindi è più facilmente interpretabile.
- E' il linguaggio grafico adottato per rappresentare i circuiti dell'elettronica digitale.

Regole di esecuzione

Un programma scritto in linguaggio FBD viene eseguito valutando le l'uscita dei singoli blocchi in funzione dello stato logico degli ingressi.

Il flusso di dati può andare sempre e solo da sinistra a destra, senza possibilità di inversione.

L'ordine di valutazione dei blocchi di codice è quello che procede dal primo blocco in alto verso l'ultimo blocco in basso.

Quando l'ultimo blocco viene valutato, si inizia nuovamente a valutare il primo (dopo aver aggiornato le uscite e letti gli ingressi).

Struttura del Linguaggio FBD

- Il linguaggio FBD traduce ogni funzione logica in una combinazione di porte logiche, ossia una rete logica.
- Si hanno tante reti logiche quanto sono le funzioni logiche
- Ad ogni operazione logica della funzione corrisponde un blocco nella rete associata
- Gli ingressi del blocco sono gli operandi dell'operazione logica considerata.
- L'uscita è il risultato della funzione logica.

Functional Block Diagram Codifica S7

Nel PLC Siemens il linguaggio di programmazione Instruction List è chiamato FUP.

Ogni variabile è individuata dal seguente codice:

X A.B

X lettera che identifica il tipo di elemento
A.B indirizzo dell'elemento

Elemento	Simbolo	Esempio
INGRESSI	E	(E 0.4)
USCITE	A	(A 3.4)
VARIABILI INTERNE	M	(M 1.4)
TEMPORIZZATORI	T	(T 2.4)
CONTATORI	C	(C 0.4)

Set istruzioni del FUP

- Operazioni logiche combinatorie di bit
- Operazioni di confronto
- Operazioni di conversione
- Operazioni di conteggio
- Operazioni di blocchi di dati
- Operazioni di conteggio
- Operazioni di blocchi di dati
- Operazioni di salto
- Funzioni con numeri in virgola mobile
- Funzioni con numeri in virgola fissa
- Operazioni di controllo del programma
- Operazioni di scorrimento e rotazione
- Operazioni di temporizzazione
- Operazioni logiche a parola

Sommario delle operazioni logiche combinatorie di bit

Le operazioni logiche combinatorie a bit operano con le due cifre: 1 e 0.

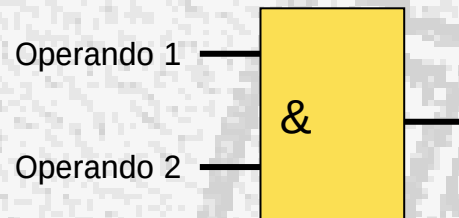
Le operazioni logiche combinatorie a bit interpretano gli stati di segnale di 1 e 0, e li combinano secondo la logica booleana per eseguire una varietà di funzioni.

Queste combinazioni producono un risultato di 1 o 0 che è chiamato “risultato logico combinatorio” (RLC).

Università di Pisa - Dipartimento di
Sistemi Elettrici e Automazione

Blocco AND

Simbolo

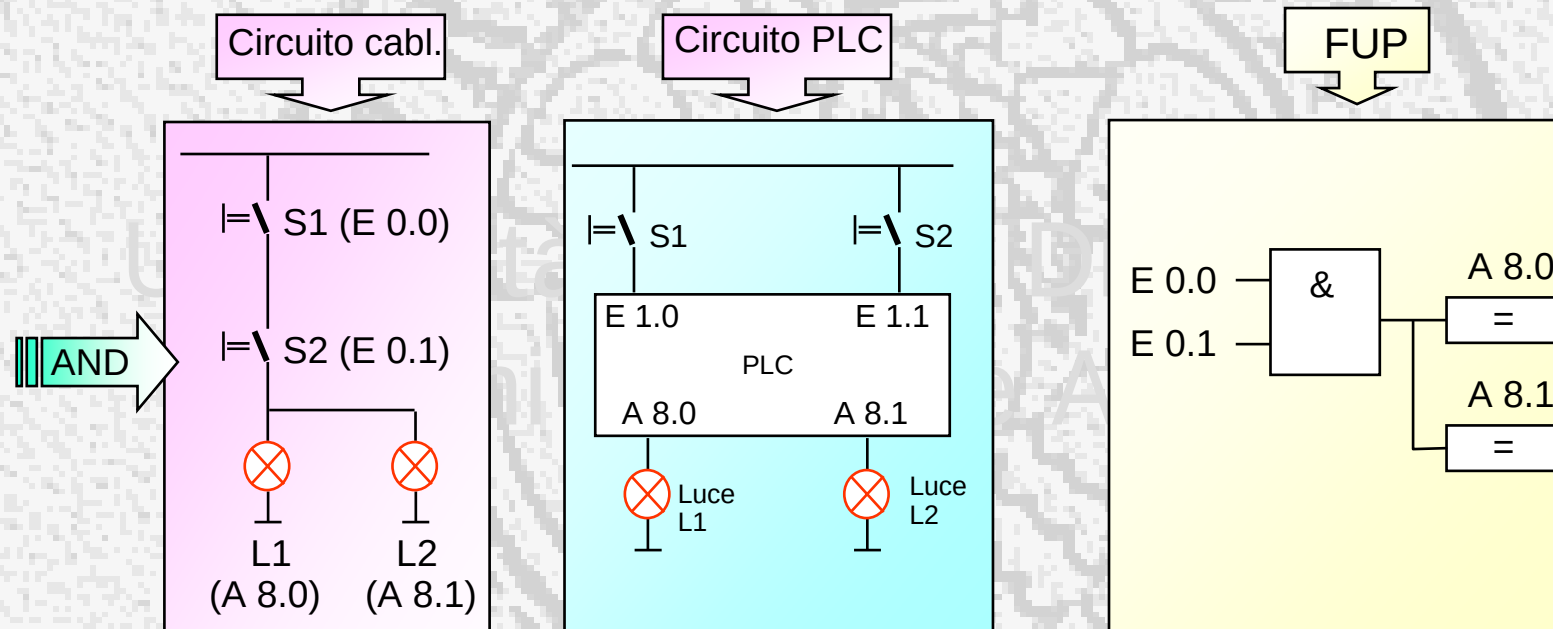


con Operando E, A, M, L, D, T, Z

L'utente può utilizzare l'operazione **AND** per interrogare lo stato di segnale di due o più operandi specificati negli ingressi di un box AND.

Se lo stato di segnale di tutti gli operandi è 1, la condizione viene soddisfatta e l'operazione dà il risultato 1. Se lo stato di segnale di un operando è 0, la condizione non viene soddisfatta e l'operazione dà il risultato 0.

Operazione logica binaria: AND

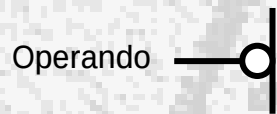


$$E 0.0 * E 0.1 = A 8.0$$

$$E 0.0 * E 0.1 = A 8.1$$

Negazione di un ingresso binario

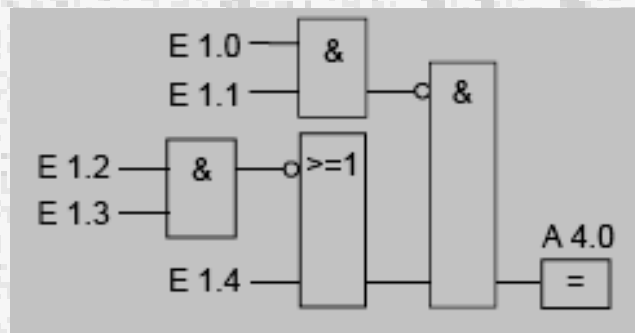
Simbolo



con Operando E, A, M, L, D, T, Z

L'operazione **Nega ingresso binario** nega l'RLC.

Esempio

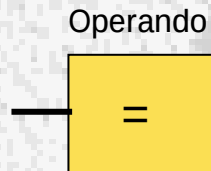


L'uscita A 4.0 è 1 quando:

- lo stato di segnale di E 1.0 AND E 1.1 NON è 1.
- AND lo stato di segnale di E 1.2 AND E 1.3 NON è 1
- OR lo stato di segnale di E 1.4 NON è 1.

Blocco =

Simbolo



con Operando E, A, M, D, L

L'operazione **Assegnazione** assegna il risultato logico combinatorio all'operando associato al blocco.

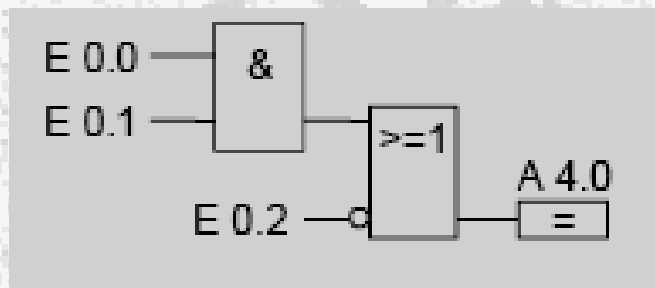
Se le condizioni delle combinazioni logiche vengono soddisfatte lo stato di segnale del box di uscita è 1; altrimenti è 0.

Si può posizionare il box **Assegnazione** soltanto all'estremità destra di una stringa logica. È tuttavia possibile utilizzare più box.

Per creare un'assegnazione negata utilizzare l'operazione Nega ingresso.

Blocco =

Esempio

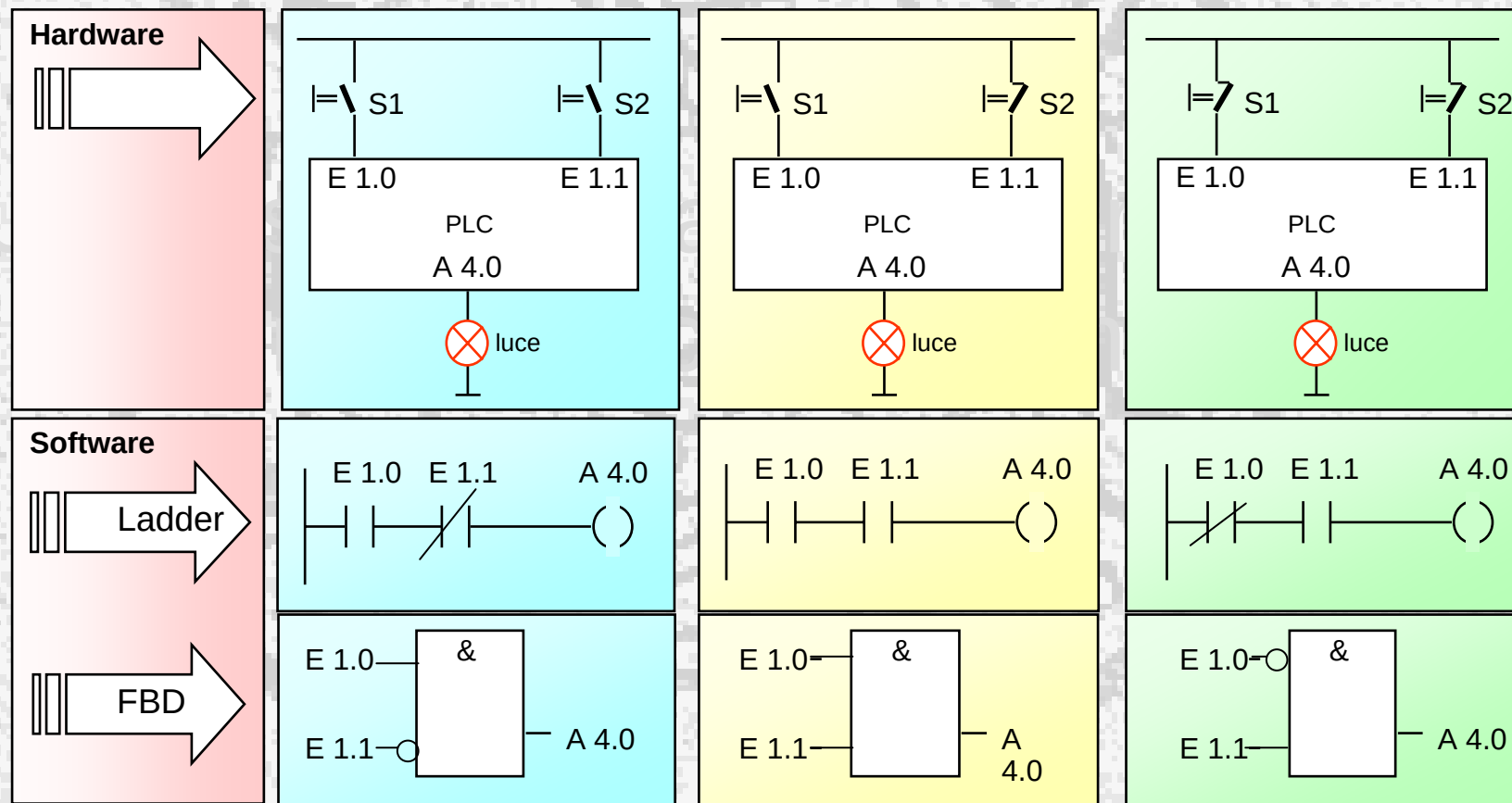


Lo stato di segnale dell'uscita A 4.0 è 1 quando:

- le uscite E 0.0 AND E 0.1 hanno lo stato di segnale 1.
- Oppure (OR) E 0.2 = 0.

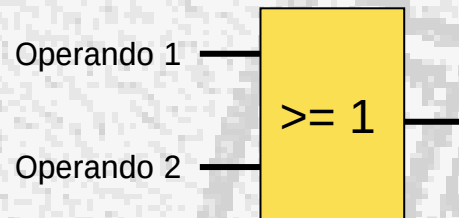
Esercizio

In tutti e tre i casi la luce deve accendersi solo quando S1 è attivato e S2 non è attivato!



Blocco OR

Simbolo

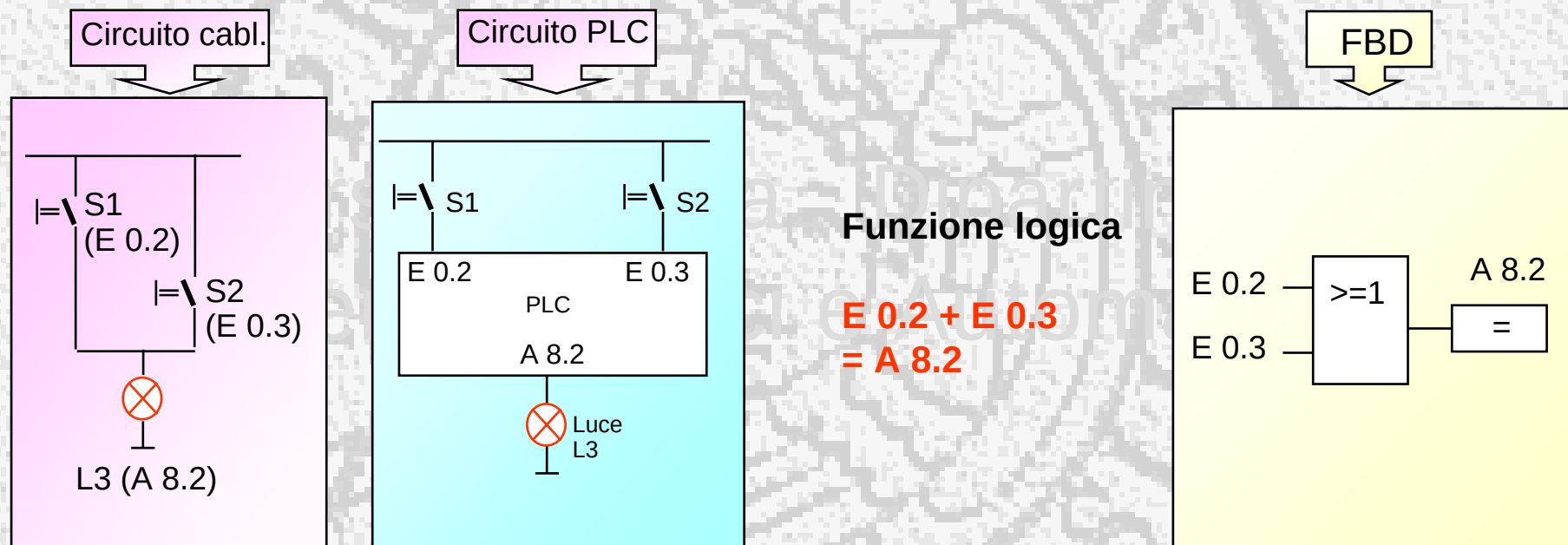


con Operando E, A, M, L, D, T, Z

L'utente può utilizzare l'operazione **OR** per interrogare gli stati di segnale di due o più operandi specificati negli ingressi di un box OR.

Se lo stato di segnale di uno degli operandi è 1, la condizione posta viene soddisfatta e l'operazione dà il risultato 1. Se lo stato di segnale degli operandi è 0, la condizione posta non viene soddisfatta e l'operazione dà il risultato 0.

Operazione logica binaria: AND

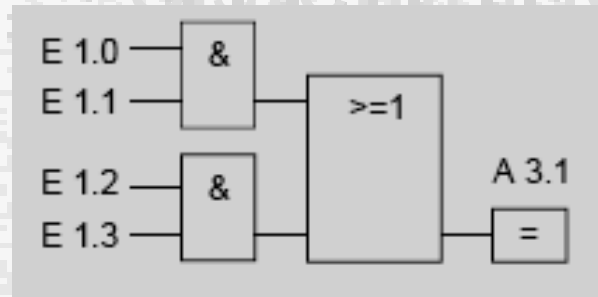


Combinazione logica AND prima di OR

L'operazione **AND prima di OR** può essere utilizzata per interrogare il risultato di un'interrogazione di segnale secondo la tabella della verità OR.

In un'operazione **AND prima di OR** lo stato di segnale è 1 quando viene soddisfatta almeno una combinazione logica AND.

Esempio



L'uscita A 3.1 ha lo stato di segnale 1 se viene soddisfatta almeno una combinazione logica AND.

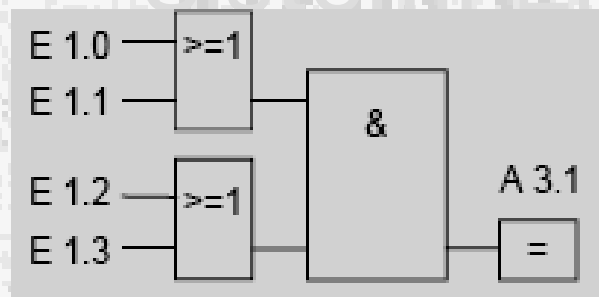
L'uscita A 3.1 ha lo stato di segnale 0 se non viene soddisfatta alcuna combinazione logica AND.

Combinazione logica OR prima di AND

L'operazione **OR prima di AND** può essere utilizzata per interrogare il risultato di un'interrogazione di segnale secondo la tabella della verità AND.

In un'operazione **OR prima di AND** lo stato di segnale è 1 quando viene soddisfatta almeno una combinazione logica OR.

Esempio

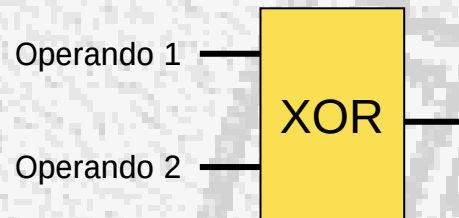


L'uscita A 3.1 ha lo stato di segnale 1 se vengono soddisfatte entrambi le combinazioni logiche OR.

L'uscita A 3.1 ha lo stato di segnale 0 se non viene soddisfatta almeno una delle combinazioni logiche OR.

Blocco XOR

Simbolo

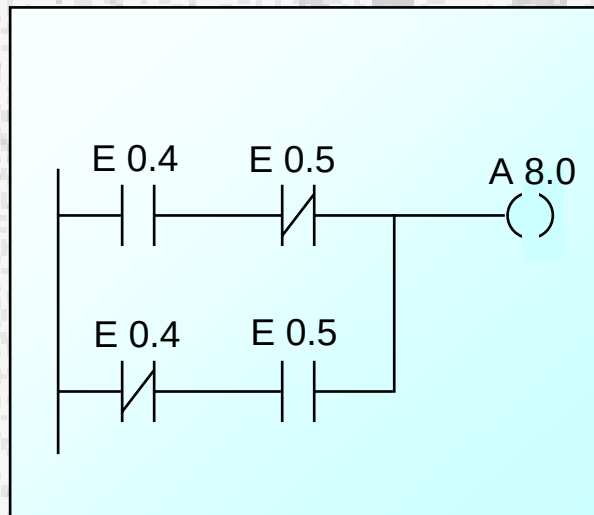


con Operando E, A, M, L, D, T, Z

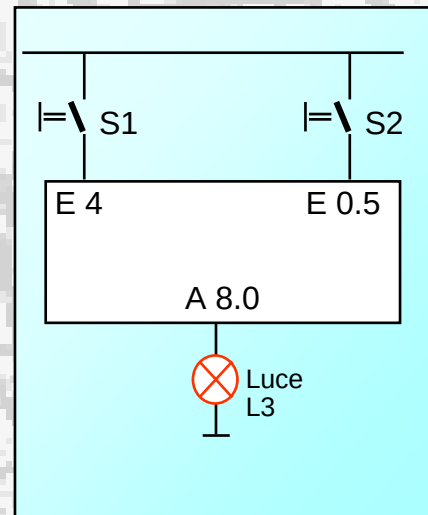
L'operazione **OR esclusivo** può essere utilizzata per interrogare il risultato di un'interrogazione di segnale secondo la tabella della verità OR esclusivo. In una combinazione logica OR esclusivo lo stato di segnale è 1 quando lo stato di segnale di uno dei due operandi specificati è 1. Con gli elementi XOR per l'interrogazione di più di due operandi il risultato logico combinatorio condiviso è "1", se un numero dispari di operandi interrogati dà come risultato dell'interrogazione "1".

Operazioni logiche binarie: OR Esclusivo (XOR)

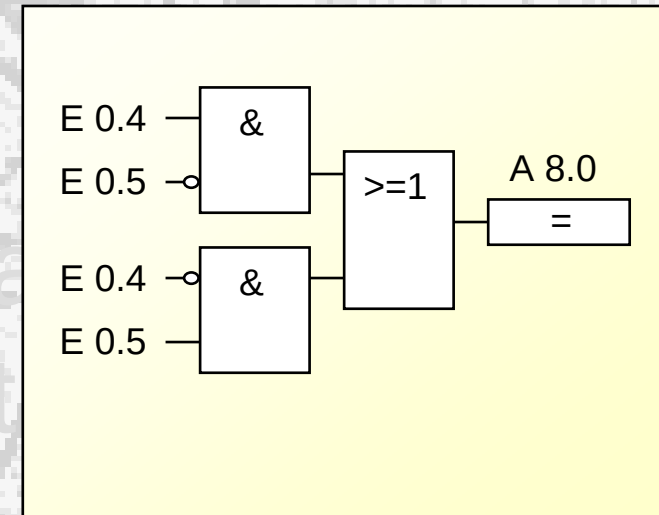
Ladder



Circuito PLC

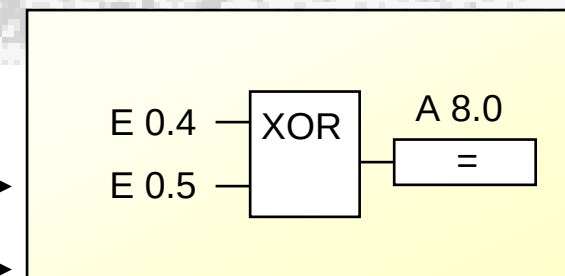
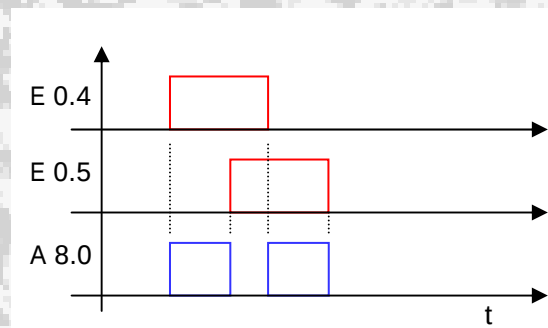


FBD



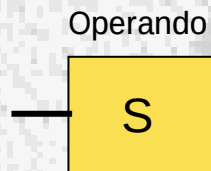
Funzione logica

$$(E\ 0.4 * E\ 0.5) + (E\ 0.4 * E\ 0.5) = A\ 8.0$$



Set Bobina

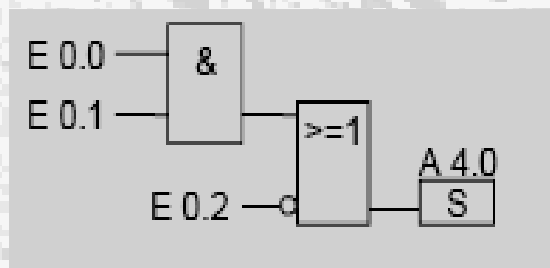
Simbolo



L' Operando indica il bit da resettare.
Aree di memoria E, A, M, T, Z, D, L

L'operazione **Set bobina** viene eseguita soltanto se $RLC = 1$. Se $RLC = 1$, questa operazione setta a 1 l'operando specificato. Se $RLC = 0$, l'operazione non ha alcun effetto sull'operando specificato. L'operando rimane inalterato.

Esempio

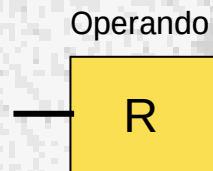


Lo stato di segnale della bobina A 4.0 viene impostato su 1 quando:

- le uscite E 0.0 AND E 0.1 hanno lo stato di segnale 1
- OR lo stato di segnale dell'uscita E 0.2 = 0.

Reset Bobina

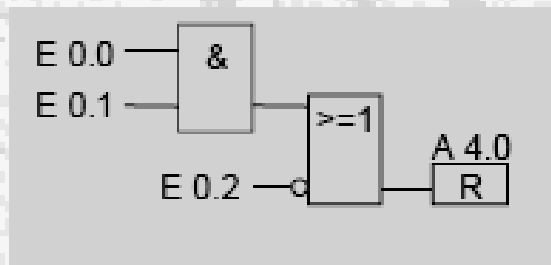
Simbolo



L' Operando indica il bit da resettare.
Aree di memoria E, A, M, T, Z, D, L

L'operazione **Reset bobina** viene eseguita soltanto se $RLC = 1$. Se $RLC = 1$, questa operazione resetta a 0 l'operando specificato. Se $RLC = 0$, l'operazione non ha alcun effetto sull'operando specificato. L'operando rimane inalterato.

Esempio

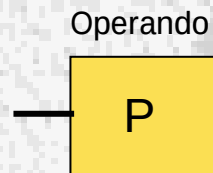


Lo stato di segnale della bobina A 4.0 viene resettato a 0 quando:

- le uscite E 0.0 AND E 0.1 hanno lo stato di segnale 1.
- OR lo stato di segnale dell'uscita E 0.2 = 0.

Blocco Rilevamento del fronte di salita di un segnale

Simbolo



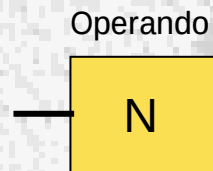
Con Operando E, A, M, D, L

L'operazione **Rileva fronte RLC di salita** riconosce un cambiamento di segnale da 0 a 1 (fronte di salita) nell'operando specificato.

Lo stato logico del segnale in ingresso al blocco (RLC) viene messo a confronto con quello dell'operando (il quale rappresenta il valore dello stesso segnale al poasso di scansione precedente). Se lo stato di segnale dell'operando è 0 e l'RLC prima dell'operazione è 1, l'RLC a valle del blocco è posto a 1 (impulso), altrimenti 0. L'RLC in ingresso al blocco viene salvato nell'operando.

Blocco Rilevamento del fronte di discesa di un segnale

Simbolo

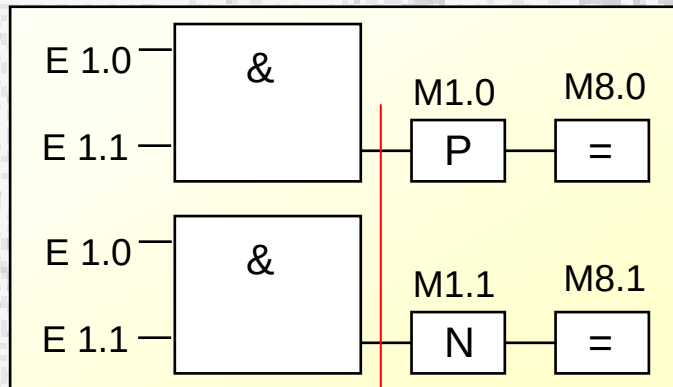


Con Operando E,A, M, D, L

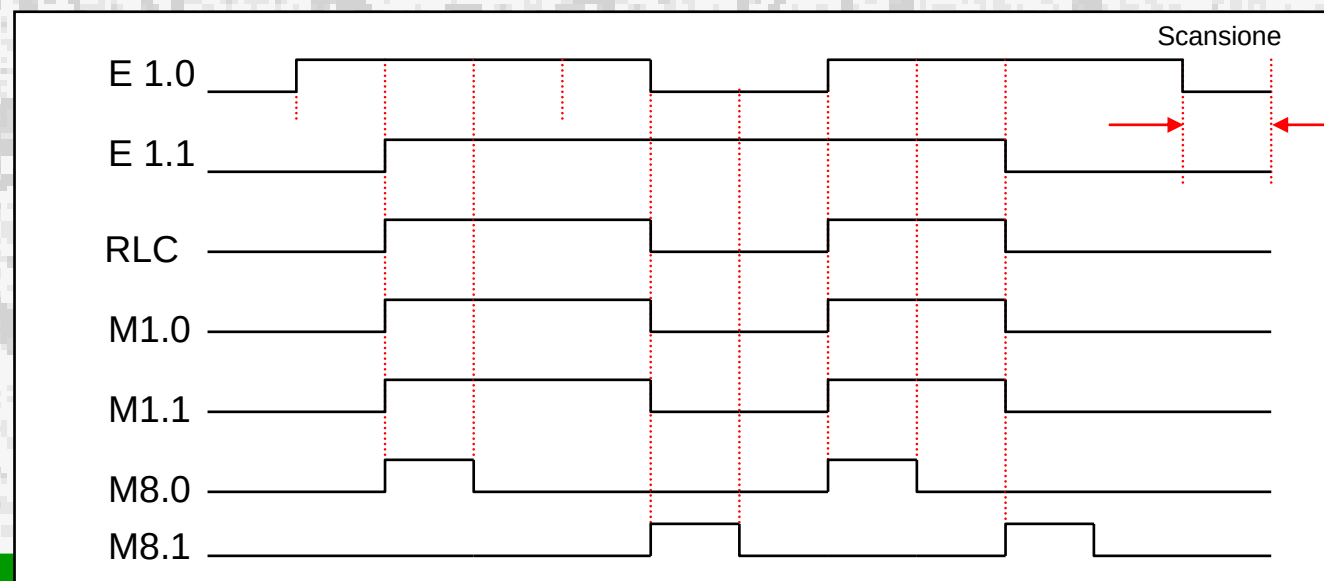
L'operazione **Rileva fronte RLC di discesa** riconosce un cambiamento di segnale da 1 a 0 (fronte di discesa) nell'operando specificato.

Lo stato logico del segnale in ingresso al blocco (RLC) viene messo a confronto con quello dell'operando (il quale rappresenta il valore dello stesso segnale al poasso di scansione precedente). Se lo stato di segnale dell'operando è 1 e l'RLC prima dell'operazione è 0, l'RLC a valle del blocco è posto a 1 (impulso), altrimenti 0. L'RLC in ingresso al blocco viene salvato nell'operando.

Rilevazione del fronte



RLC = Risultato Logico Combinatorio



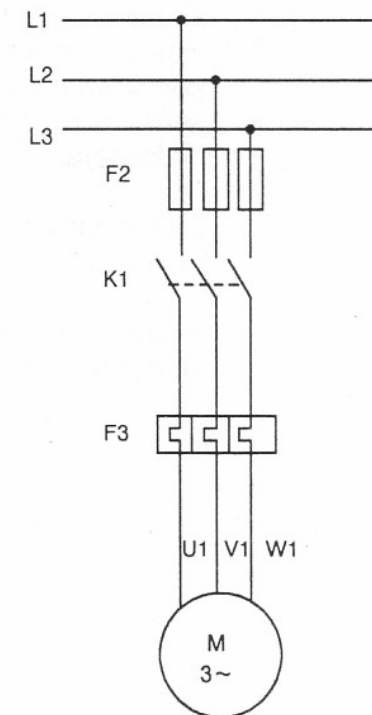
Esempio 1. Avviamento/Arresto di un motore

Descrizione del problema

Un motore asincrono viene inserito premendo il pulsante S1 (pulsante di marcia) e disinserito con il pulsante S0 (pulsante di arresto).

Le lampade di segnalazione H1 e H2 visualizzano lo stato di attività del motore.

Il motore è protetto con un interruttore termico di sovracorrente e con fusibili per il corto circuito.



Esempio 1. Avviamento/Arresto di un motore

Si definiscono gli ingressi e si associano ai corrispondenti ingressi del PLC.

Simbolo	Variabile PLC	Descrizione
S0	E 0.1	OFF Pulsante N.A.
S1	E 0.2	ON Pulsante N.A.

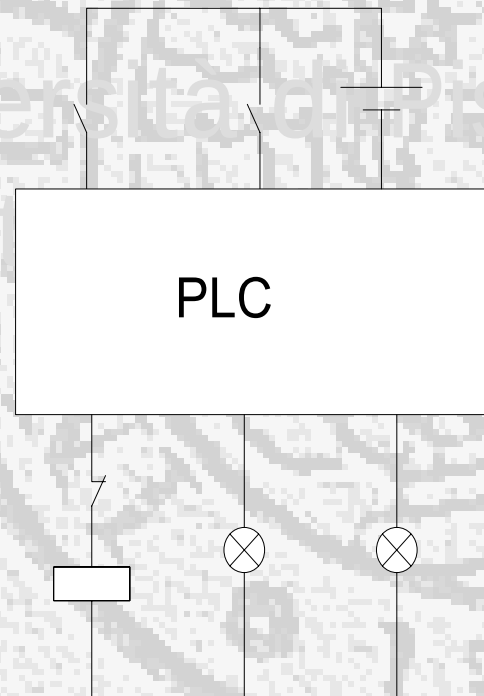
Esempio 1. Avviamento/Arresto di un motore

Si definiscono le uscite e si associano alle corrispondenti uscite del PLC.

Simbolo	Variabile PLC	Descrizione
K1	A 2.0	Bobina contattore motore
H1	A 2.1	Segnalazione motore fermo
H2	A 2.2	Segnalazione motore in marcia

Esempio 1. Avviamento/Arresto di un motore

Lo schema di collegamento al PLC è il seguente:



Per motivi di sicurezza il contatto normalmente chiuso dell'interruttore di sovracorrente F3 deve essere collegato a monte della bobina del contattore K1. Se si attiva l'interruttore di sovracorrente, la bobina del contattore K1 viene disinserita direttamente via hardware senza l'intervento del PLC. L'interruttore termico di sovracorrente deve essere dotato di un dispositivo per il ripristino manuale per evitare un successivo inserimento del contattore K1 dovuto al suo ripristino automatico.

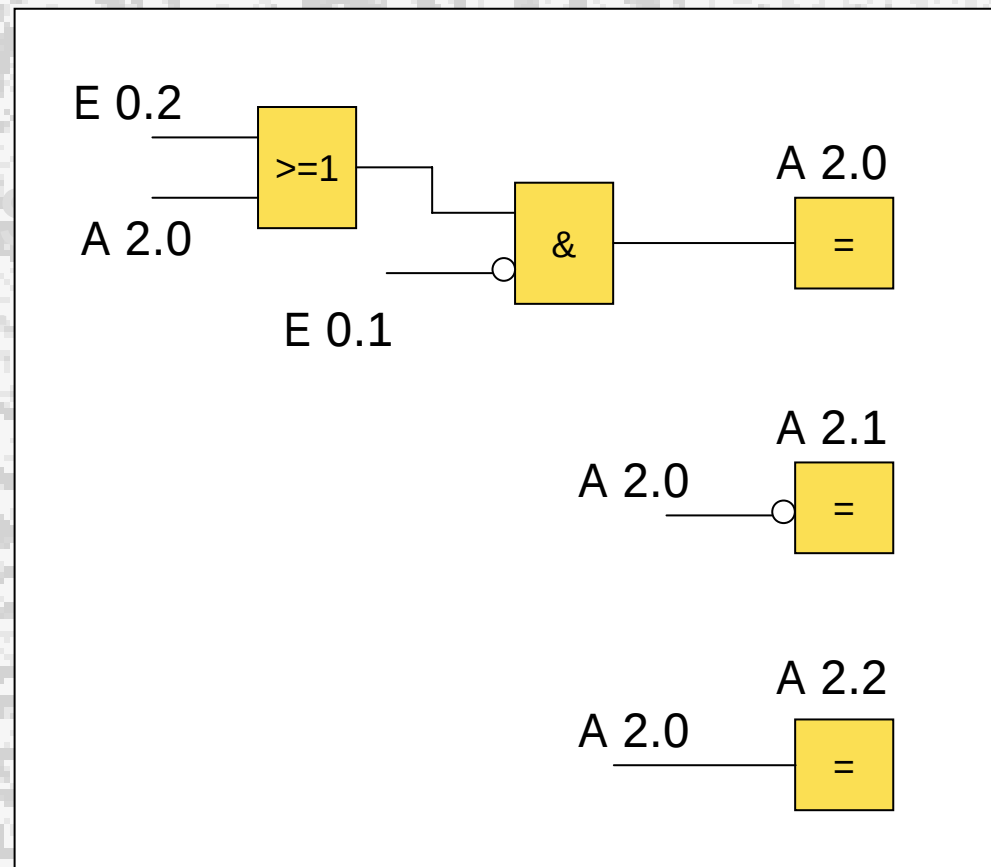
Esempio 1. Avviamento/Arresto di un motore

Il programma FUP e le funzioni logiche associate sono:

$$(E\ 0.2 + A\ 2.0) * \overline{E\ 0.1} = A\ 2.0$$

$$\overline{A\ 2.0} = A\ 2.1$$

$$A\ 2.0 = A\ 2.2$$



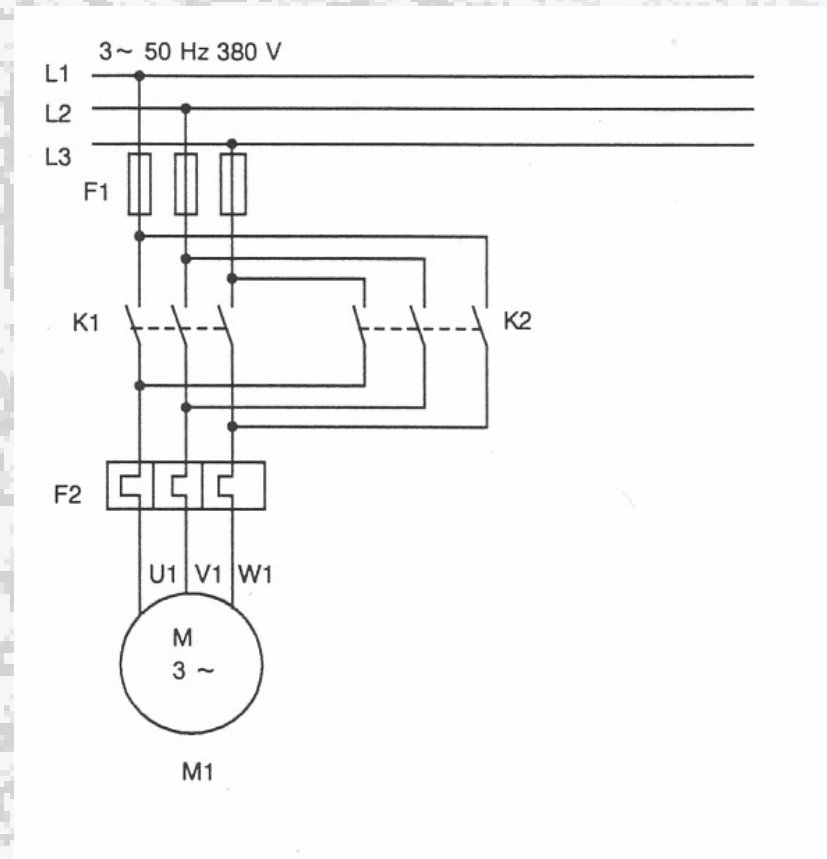
Esempio 2. Circuito invertitore per motore trifase

Descrizione del problema

Un motore asincrono viene inserito premendo il pulsante S1 per il funzionamento “marcia avanti”, S2 per “marcia indietro”, e disinserito con il pulsante S0.

Le lampade di segnalazione H1, H2 e H3 visualizzano lo stato di attività del motore.

Il motore è protetto con un interruttore termico di sovracorrente e con fusibili per il corto circuito.



Esempio 2. Circuito invertitore per motore trifase

Si definiscono gli ingressi e si associano ai corrispondenti ingressi del PLC.

Simbolo	Variabile PLC	Descrizione
S0	E 0.1	OFF Pulsante N.A.
S1	E 0.2	Pulsante Marcia avanti N.A.
S2	E 0.3	Pulsante Marcia indietro N.A.

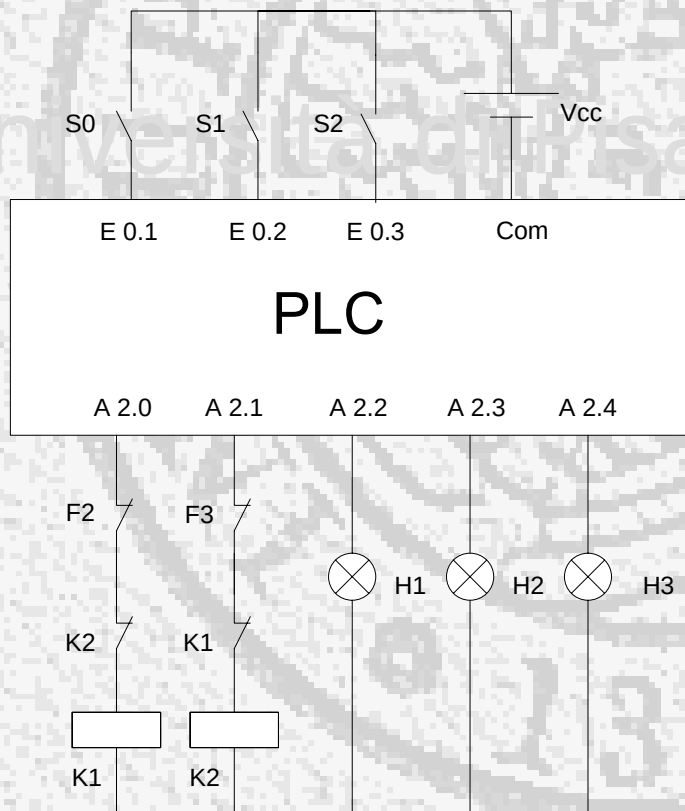
Esempio 2. Circuito invertitore per motore trifase

Si definiscono le uscite e si associano alle corrispondenti uscite del PLC.

Simbolo	Variabile PLC	Descrizione
K1	A 2.0	Bobina contattore funz. 'marcia avanti'
K2	A 2.1	Bobina contattore funz. 'marcia indietro'
H1	A 2.2	Segnalazione motore fermo
H2	A 2.3	Segnalazione motore in 'marcia avanti'
H3	A 2.4	Segnalazione motore in 'marcia indietro'

Esempio 2. Circuito invertitore per motore trifase

Lo schema di collegamento al PLC è il seguente:



I contatti dell'interruttore termico devono essere collegati sulle linee di alimentazione dei due contattori.

Inoltre un contattore non deve poter essere inserito fintanto che l'altro non è completamente disinserito. La concomitanza di entrambi i comandi di eccitazione potrebbe verificarsi a causa delle velocissime reazioni di comando elettronico rispetto alle reazioni delle parti meccaniche in movimento.

Esempio 2. Circuito invertitore per motore trifase

Il programma FUP e le funzioni logiche associate sono:

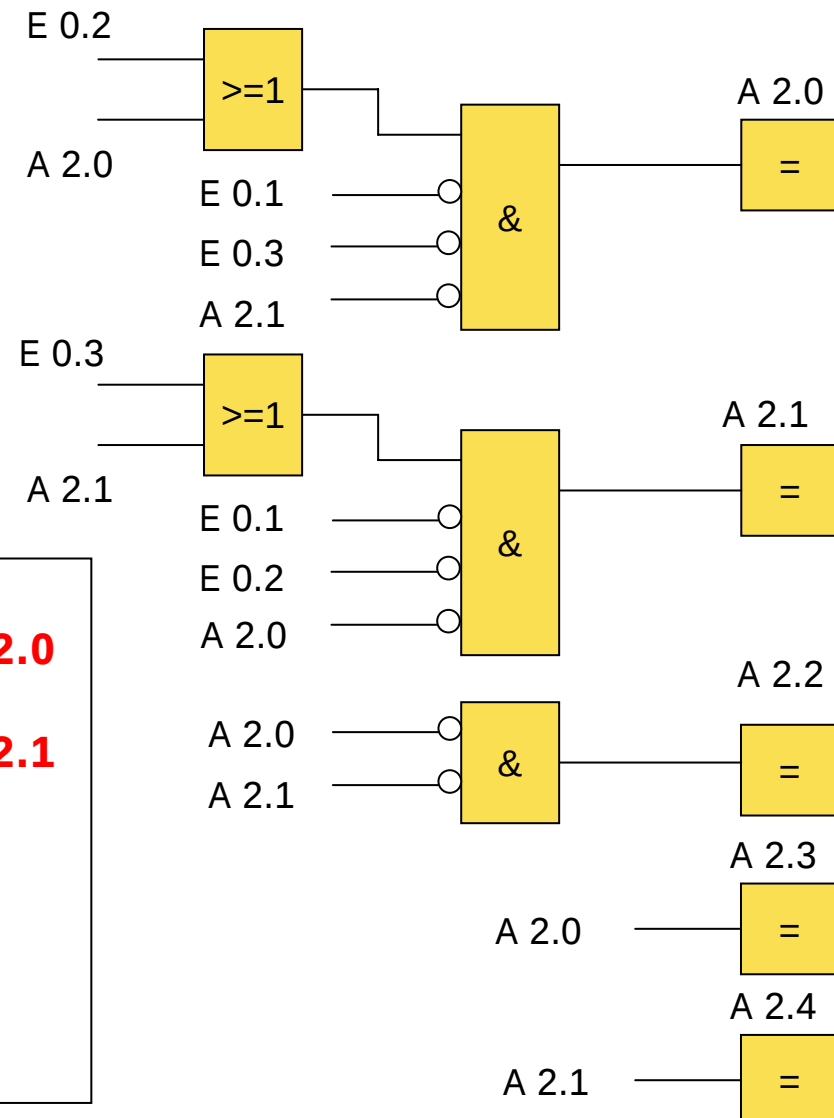
$$(E\ 0.2 + A\ 2.0) * \overline{E\ 0.1} * \overline{E\ 0.3} * \overline{A\ 2.1} = A\ 2.0$$

$$(E\ 0.3 + A\ 2.1) * \overline{E\ 0.1} * \overline{E\ 0.2} * \overline{A\ 2.0} = A\ 2.1$$

$$\overline{A\ 2.0} * \overline{A\ 2.1} = A\ 2.2$$

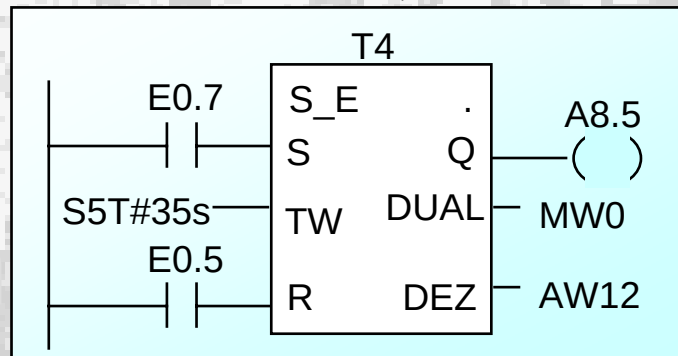
$$A\ 2.0 = A\ 2.3$$

$$A\ 2.1 = A\ 2.4$$

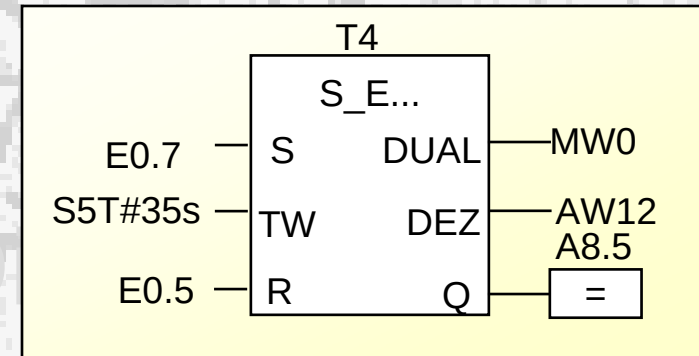


Temporizzatori: Ritardato all'Eccitazione (SE)

Ladder



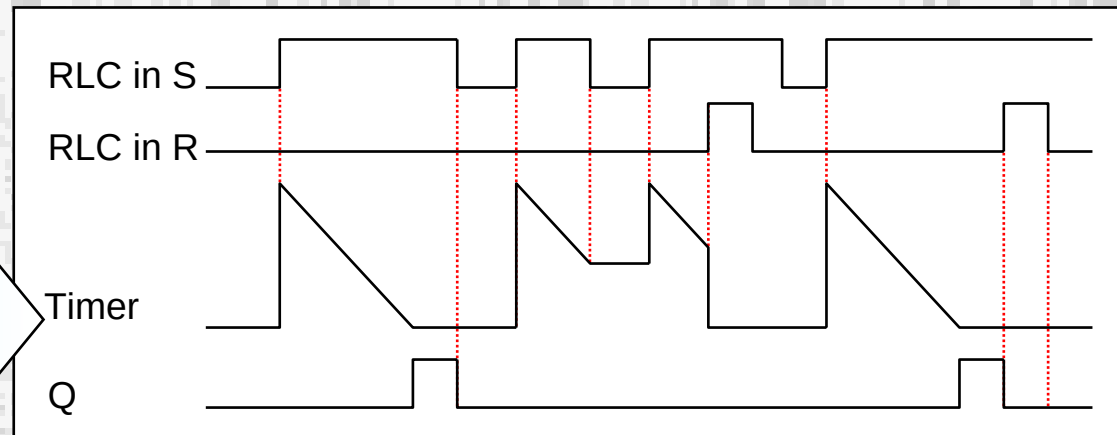
FUP



Esempio

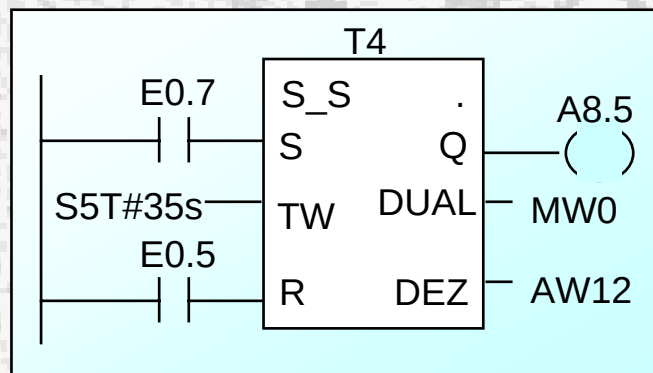
Timer

Q

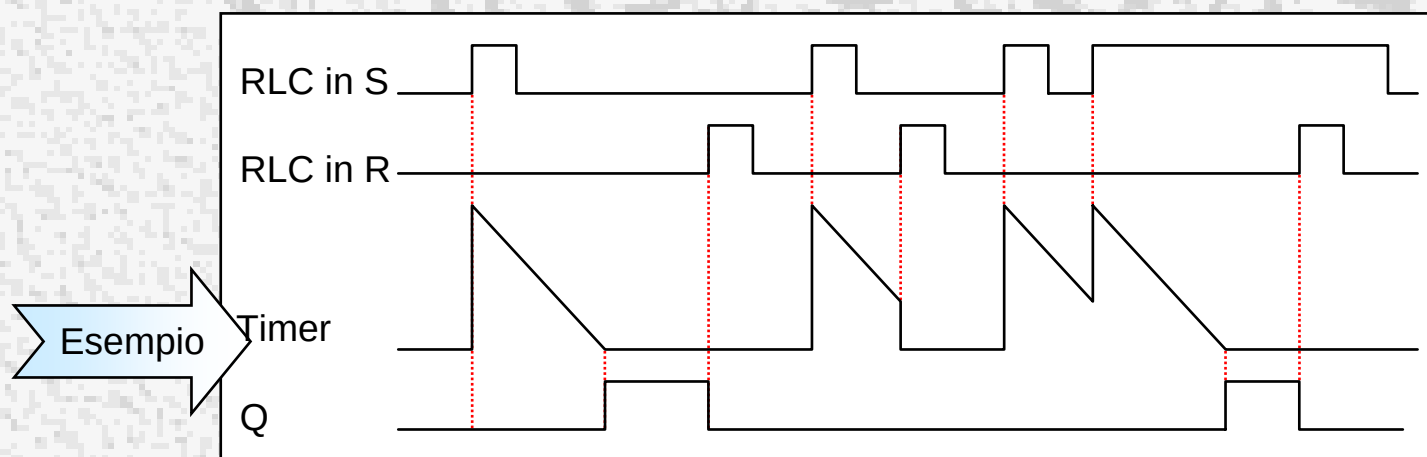
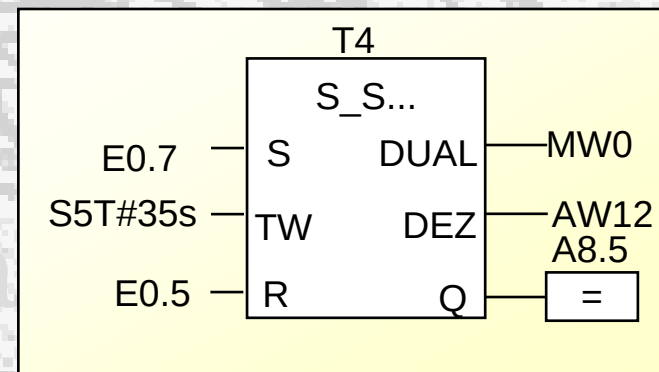


Temporizzatori: Ritardato all'Eccitazione con Memoria (SS)

Ladder

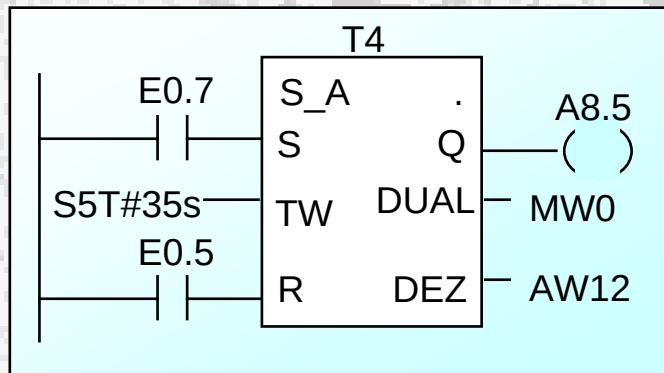


FUP

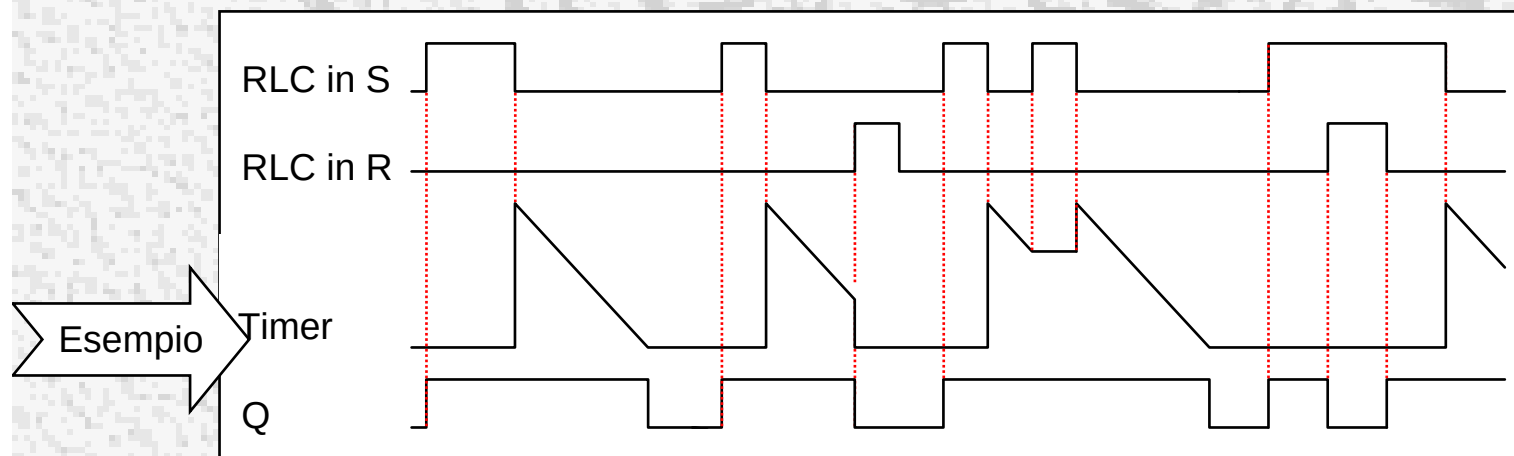
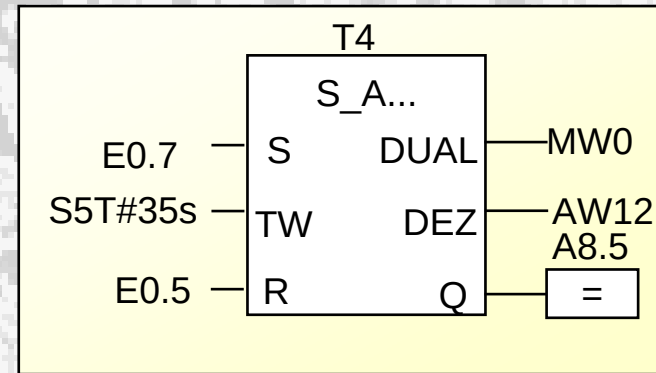


Temporizzatori: Ritardato alla Diseccitazione (SA)

Ladder

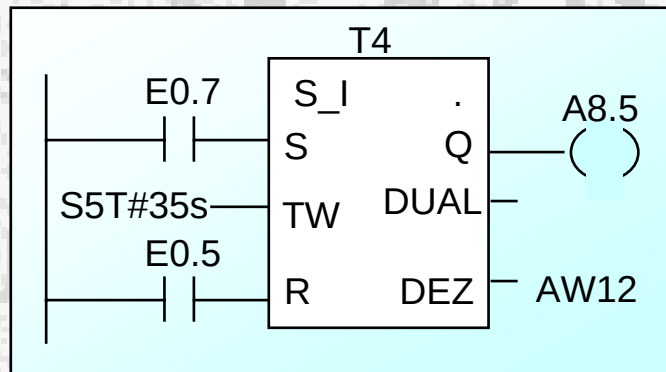


FUP

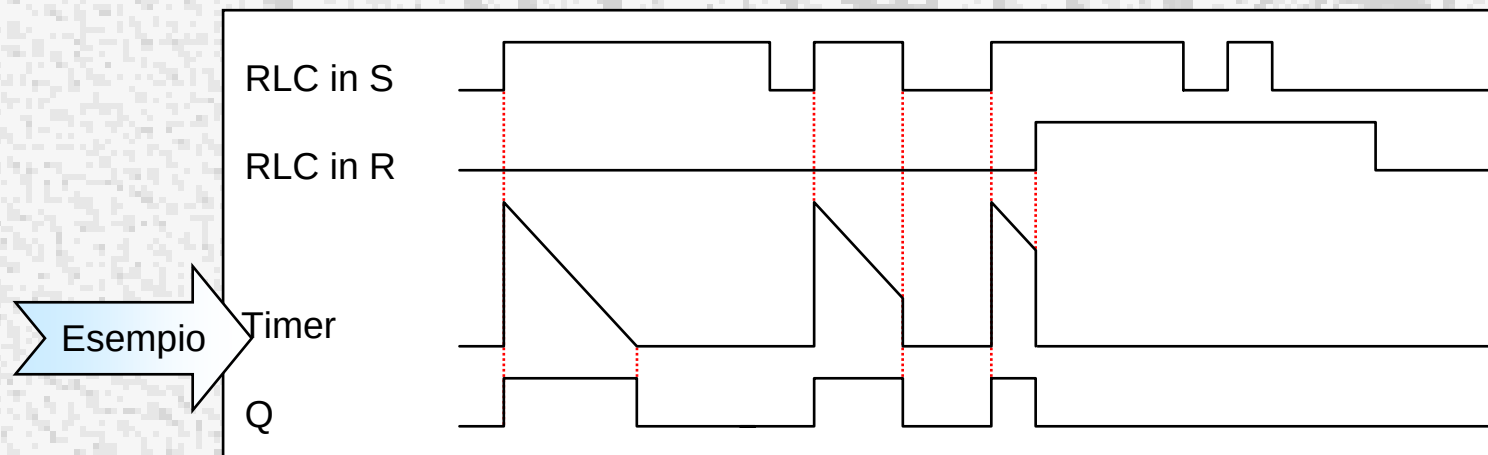
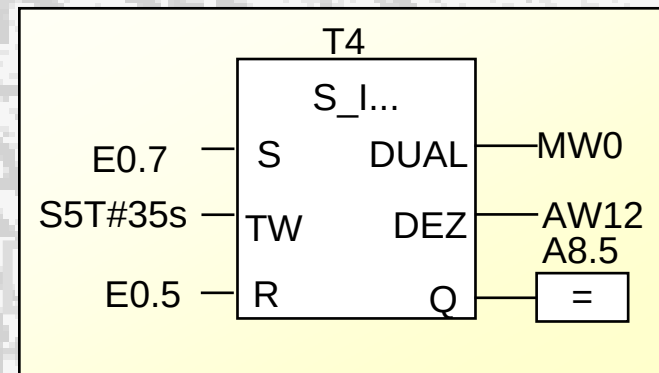


Temporizzatori: Impulsivo (SI)

Ladder

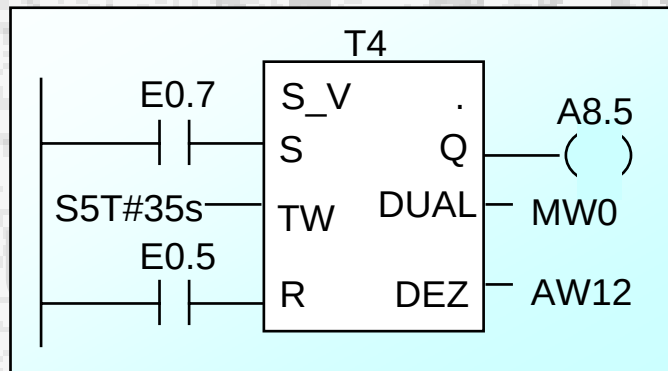


FUP

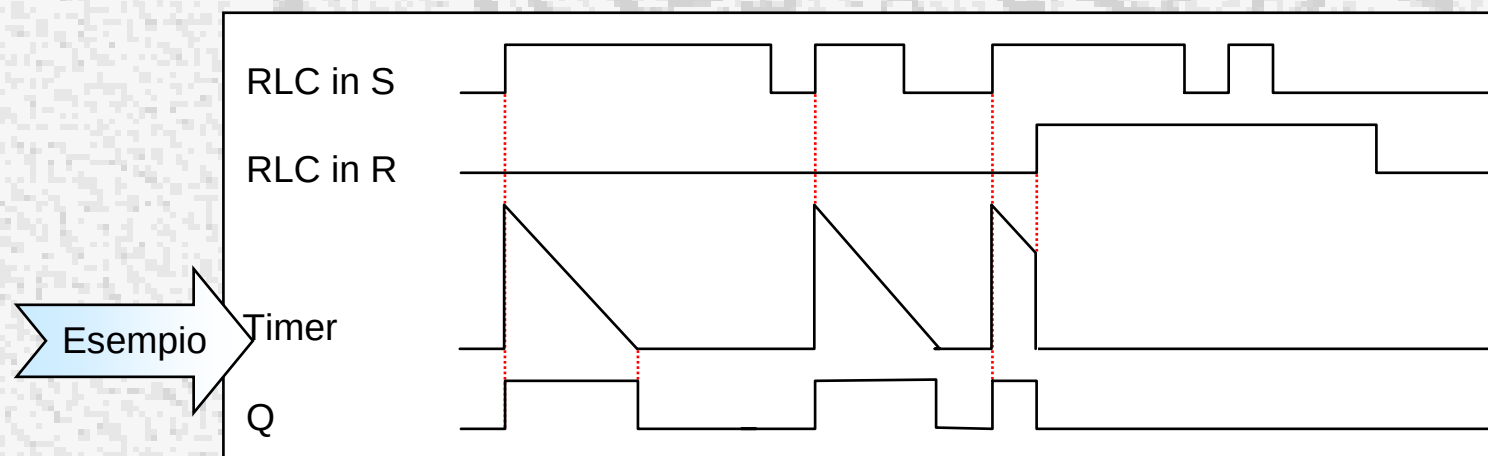
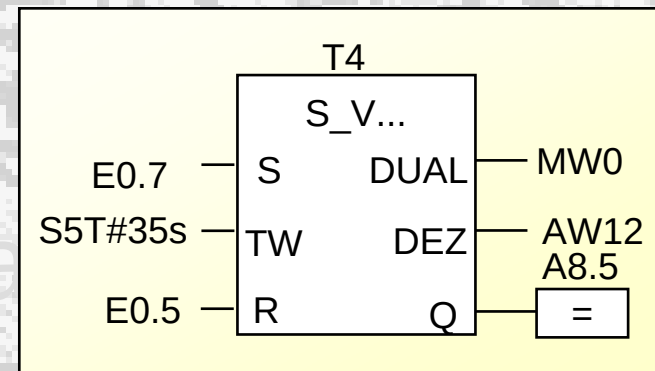


Temporizzatori: ad Impulso Prolungato (SV)

Ladder

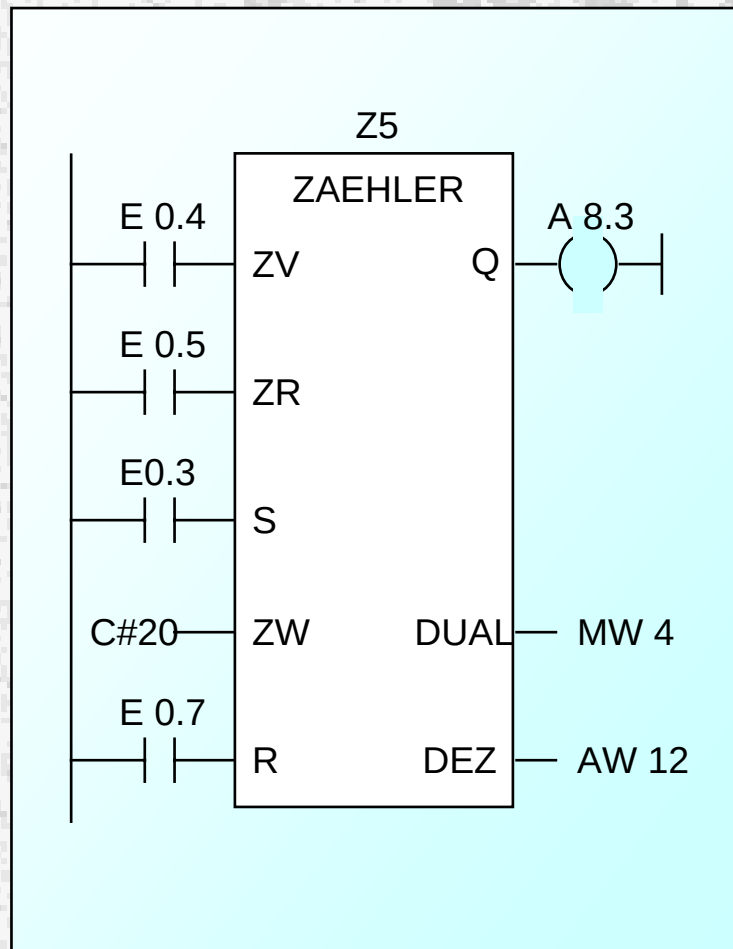


FUP

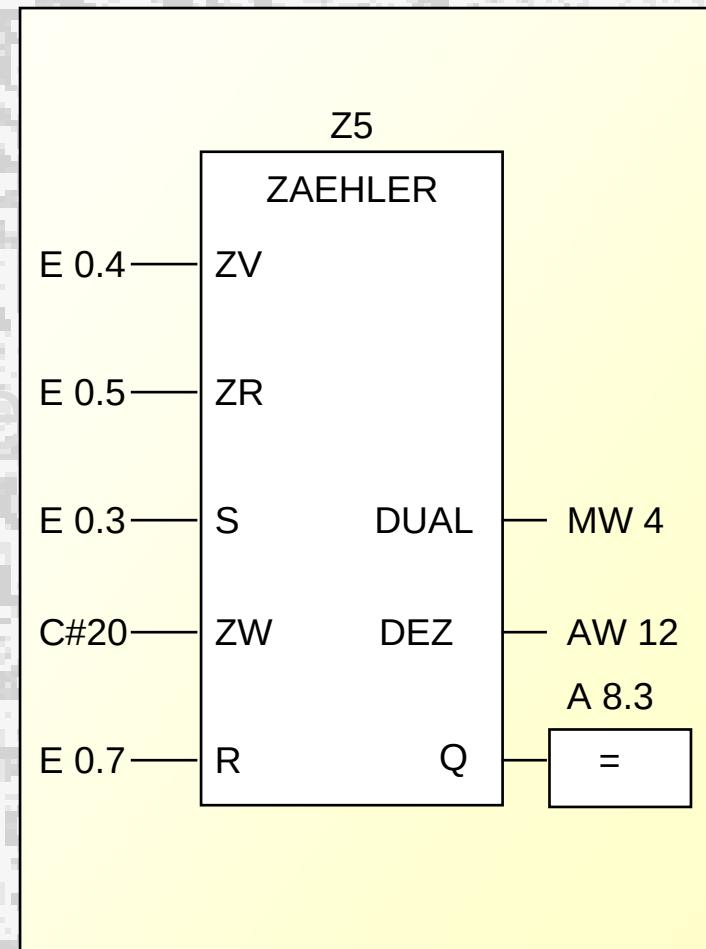


Contatori S5 in STEP 7

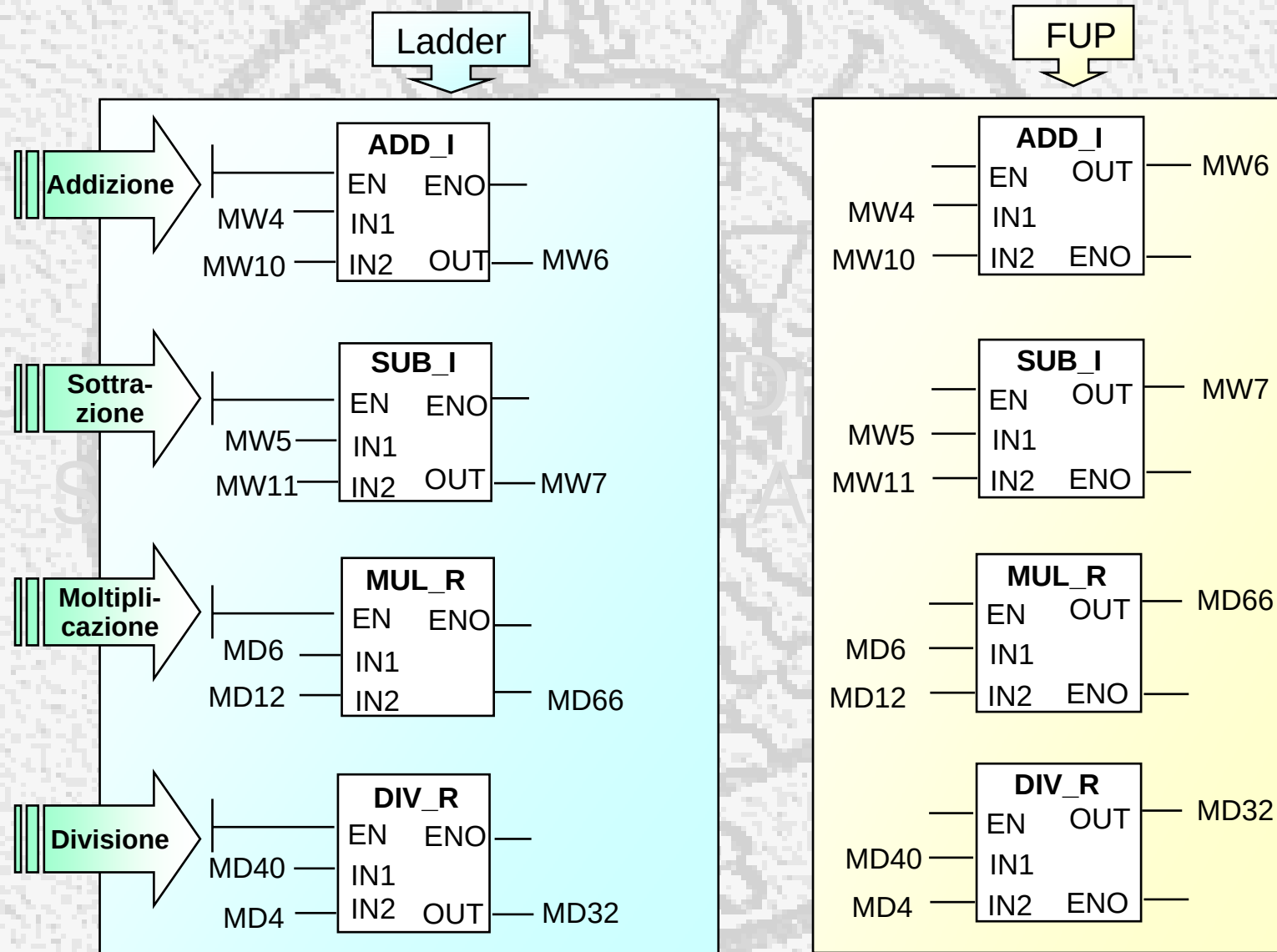
Ladder



FUP



Funzioni Matematiche di base



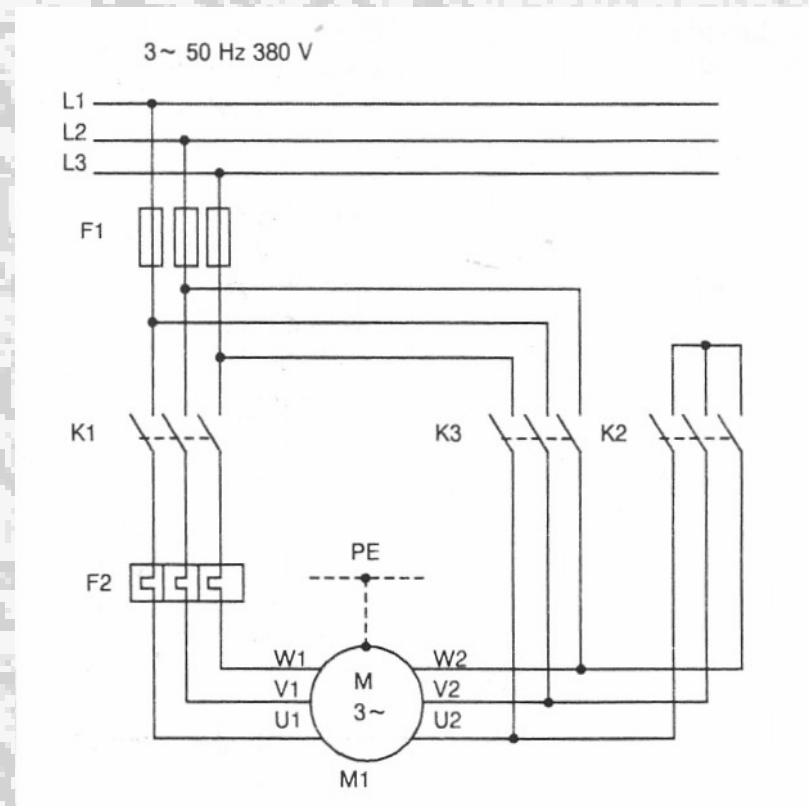
Esempio 3. Avviamento Stella-Triangolo

Descrizione del problema

Un motore asincrono viene inserito premendo il pulsante S1 (pulsante di marcia) e disinserito con il pulsante S0 (pulsante di arresto). Il motore si avvia con gli avvolgimenti statorici a stella. Dopo 10 s si ha commutazione degli avvolgimenti a triangolo.

Le lampade di segnalazione H1, H2 e H3 visualizzano lo stato di attività del motore.

Il motore è protetto con un interruttore termico di sovracorrente e con fusibili per il corto circuito.



Esempio 3. Avviamento Stella-Triangolo

Si definiscono gli ingressi e si associano ai corrispondenti ingressi del PLC.

Simbolo	Variabile PLC	Descrizione
S0	E 0.1	OFF Pulsante N.A.
S1	E 0.2	ON Pulsante N.A.

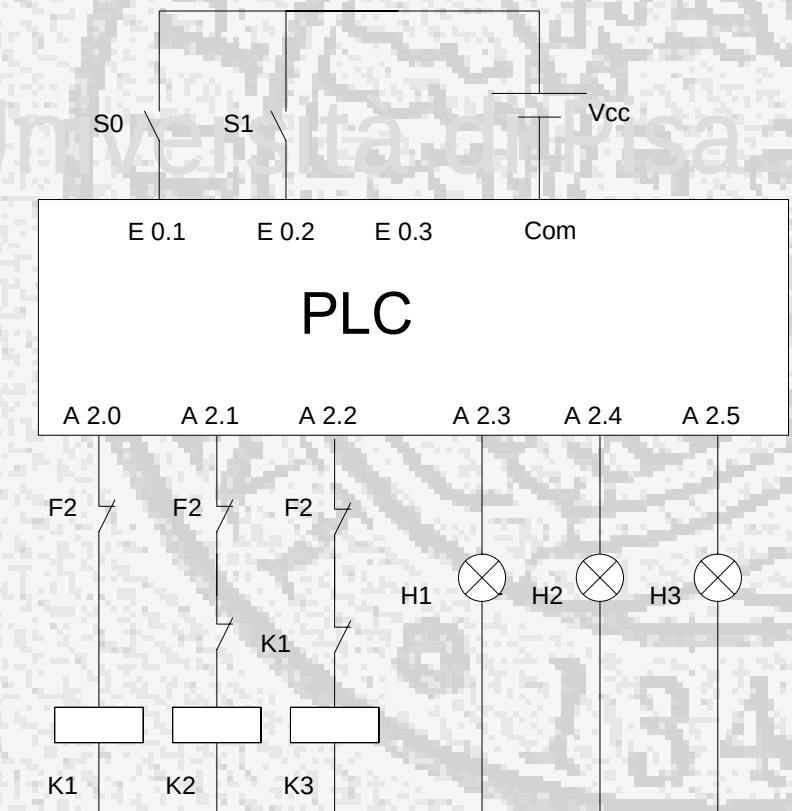
Esempio 3. Avviamento Stella-Triangolo

Si definiscono le uscite e si associano alle corrispondenti uscite del PLC.

Simbolo	Variabile PLC	Descrizione
K1	A 2.0	Bobina contattore generale
K2	A 2.1	Bobina contattore a stella
K3	A 2.2	Bobina contattore a triangolo
H1	A 2.3	Segnalazione motore fermo
H2	A 2.4	Segnalazione collegamento a stella
H3	A 2.5	Segnalazione collegamento a triangolo

Esempio 3. Avviamento Stella-Triangolo

Lo schema di collegamento al PLC è il seguente:

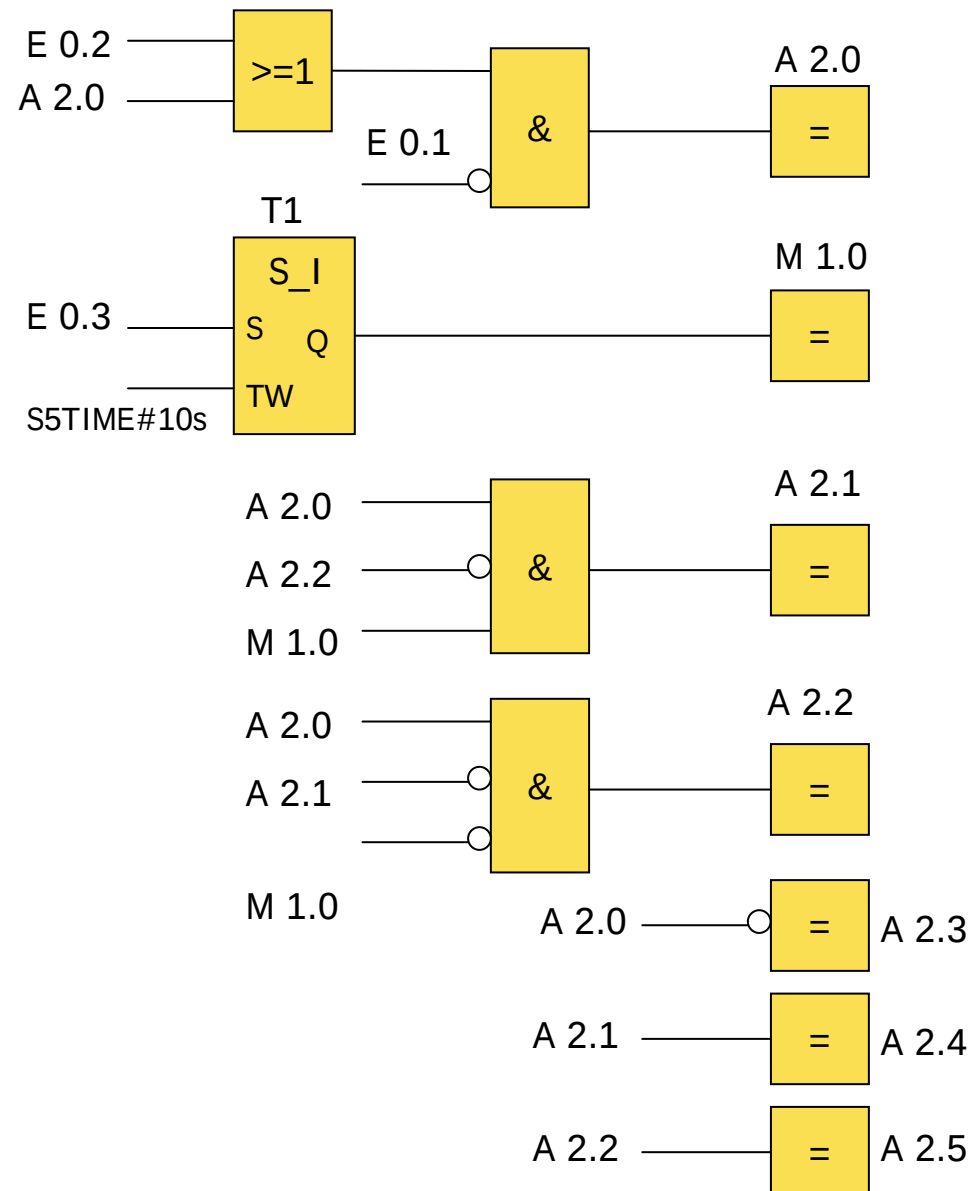
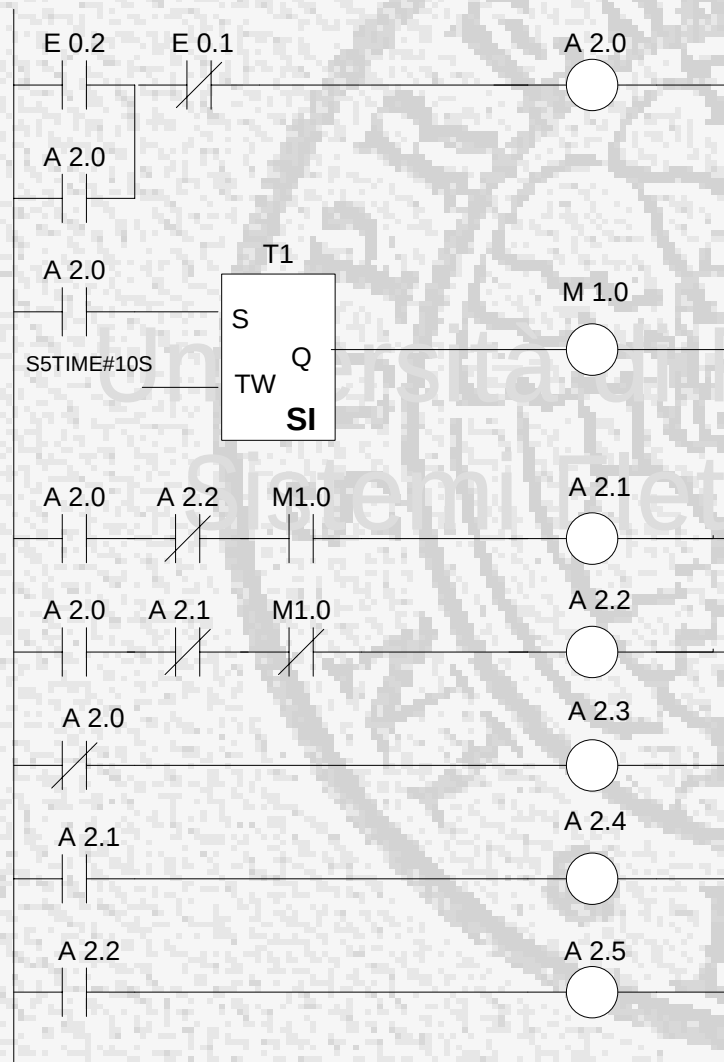


I contatti dell'interruttore termico devono essere collegati sulle linee di alimentazione dei tre contattori.

Inoltre un contattore di commutazione non deve poter essere inserito fintanto che l'altro non è completamente disinserito. La concomitanza di entrambi i comandi di eccitazione potrebbe verificarsi a causa delle velocissime reazioni di comando elettronico rispetto alle reazioni delle parti meccaniche in movimento.

Esempio 3. Avviamento/Arresto di un motore

Schema ladder



Applicazione

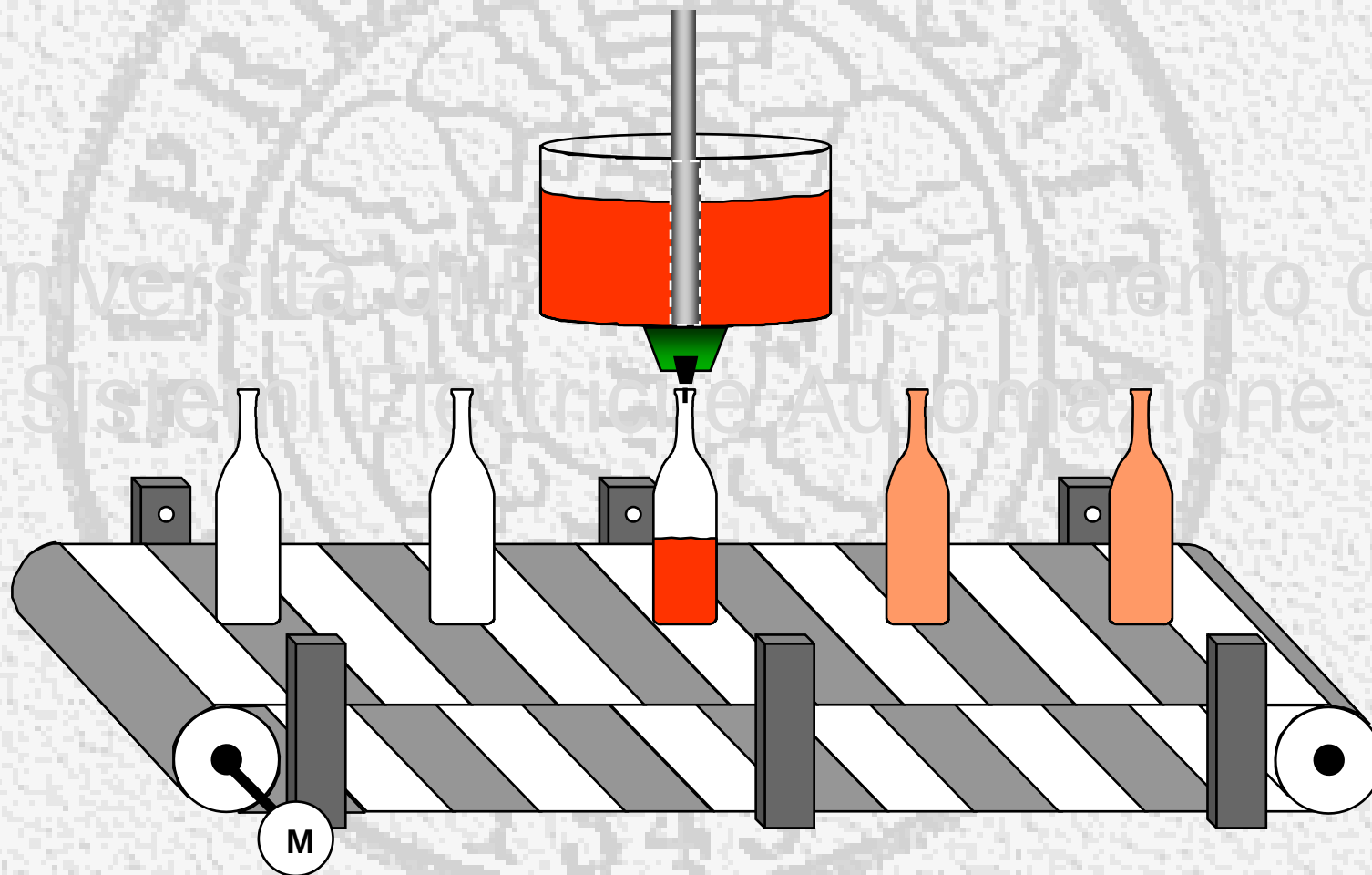
- Gestione di un impianto di imbottigliamento

Schema ladder

Linguaggio testuale

Functional Block diagram

Esercizio: Gestione di un impianto d'imbottigliamento



Descrizione dell'impianto

L'impianto deve essere acceso con un pulsante di Start e arrestato con un pulsante di Stop

Sono previste due modalità di funzionamento: Manuale e Automatico, selezionabili con un selettore.

In entrambe le modalità deve essere possibile comandare il movimento Avanti o Indietro del nastro trasportatore

Step n° 1: individuazione delle condizioni di funzionamento

Il nastro trasportatore si muove in avanti/indietro in modalità MANUALE se:

- a) l'impianto è acceso
- b) è selezionata la modalità manuale
- c) è dato il comando Avanti / Indietro

Il nastro trasportatore si muove in avanti/indietro in modalità AUTOMATICA se:

- a) l'impianto è acceso
- b) è selezionata la modalità automatica
- c) è dato il comando Avanti / Indietro

Step n° 1: individuazione delle condizioni di funzionamento

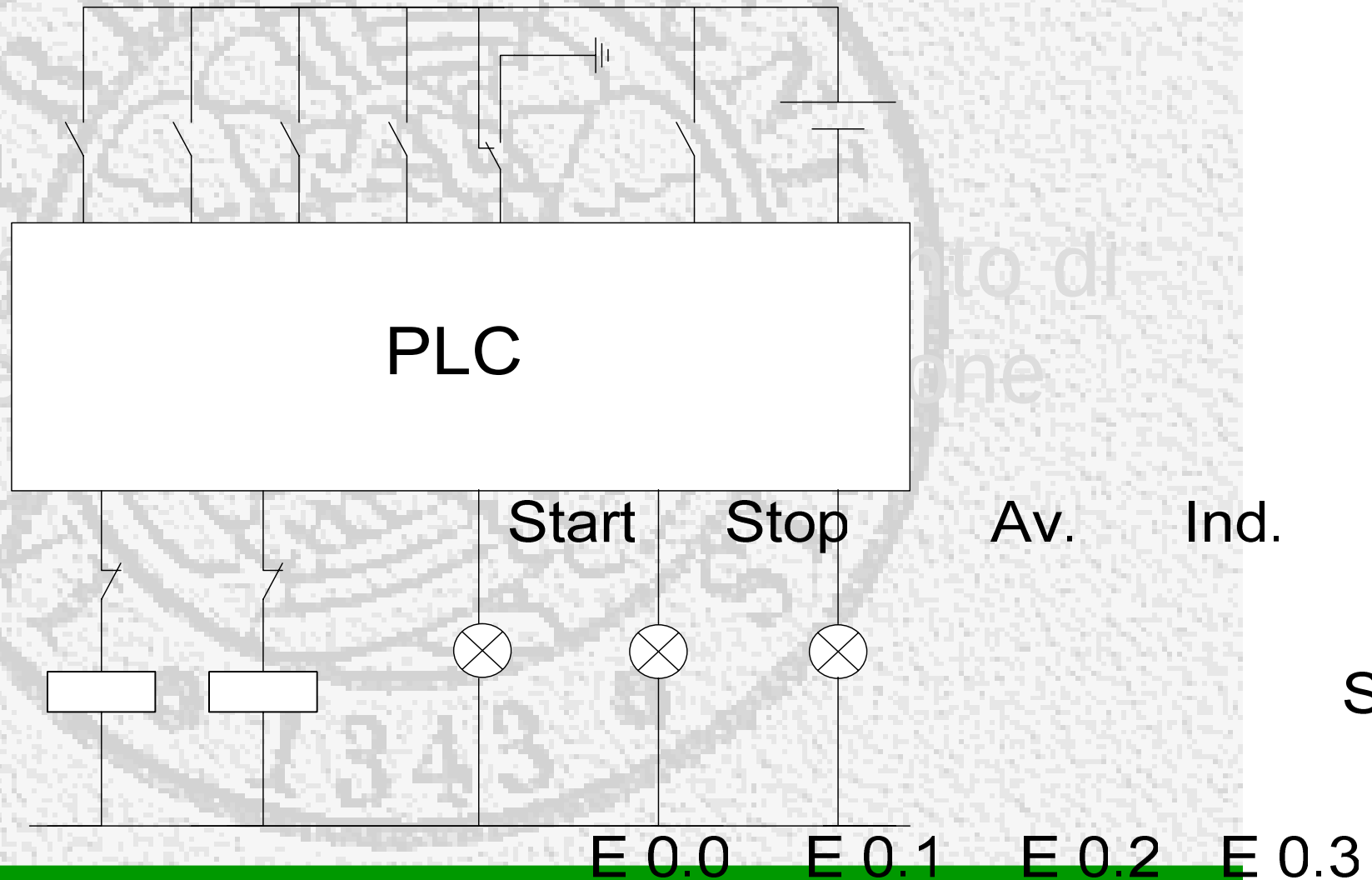
La selezione MANUALE / AUTOMATICA è realizzata con un commutatore a due stati e da un pulsante di conferma per ogni modalità

Università di Pisa - Dipartimento di
Sistemi Elettrici e Automazione

Step n° 2: definizione degli ingressi e delle uscite

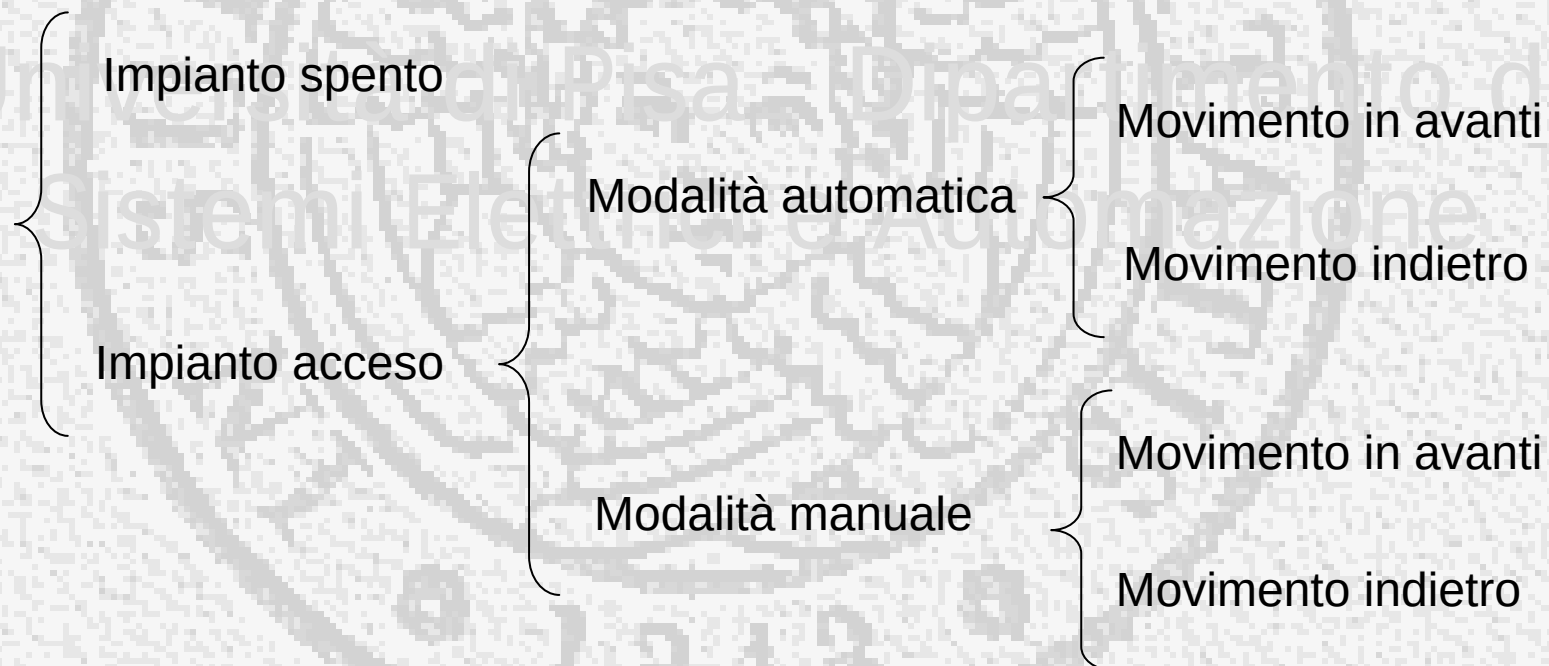
Ingressi	Uscite
<u>Impianto ON/OFF</u>	A 4.1 Led impianto acceso
E 0.0 = Start (Pulsante NA)	A 4.2 Led Modalità manuale
E 0.1 = Stop (Pulsante NA)	A 4.3 Led Modalità automatica
<u>Modo Manuale/Automatico</u>	A 8.5 Nastro trasportatore avanti
E 0.4 = Manuale/Automatico (Selettore)	A 8.6 Nastro trasportatore indietro
E 0.5 = Comando di selezione (puls. NA)	
E 0.2 = Comando avanti (puls. NA)	
E 0.3 = Comando indietro (puls. NA)	

Step n° 3: definizione dei collegamenti al PLC



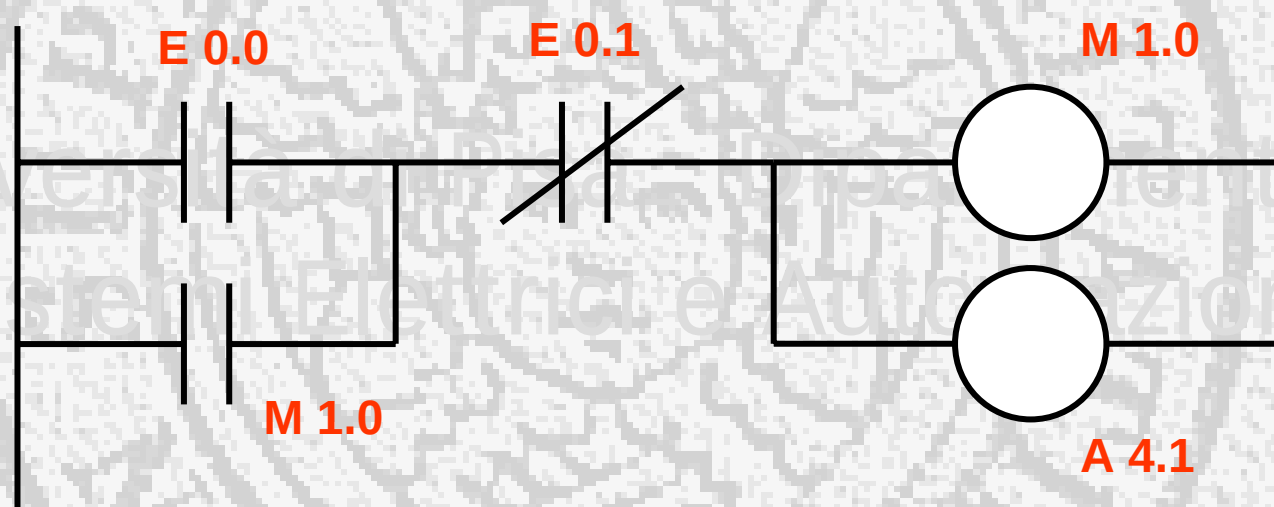
Step n° 4: definizione degli stati dell'impianto

Si definiscono le configurazioni in cui l'impianto si viene a trovare:



Accensione / arresto dell'impianto

L'impianto è acceso se viene premuto lo Start, non è premuto lo Stop e deve rimanere acceso anche se si rilascia il pulsante di Start.



Funzione Logica

$$(E0.0 + M1.0) * \overline{E0.1} = M1.0$$

$$(E0.0 + M1.0) * \overline{E0.1} = A4.1$$

Si utilizza una variabile di appoggio M 1.0

Accensione / arresto dell'impianto

Funzione Logica

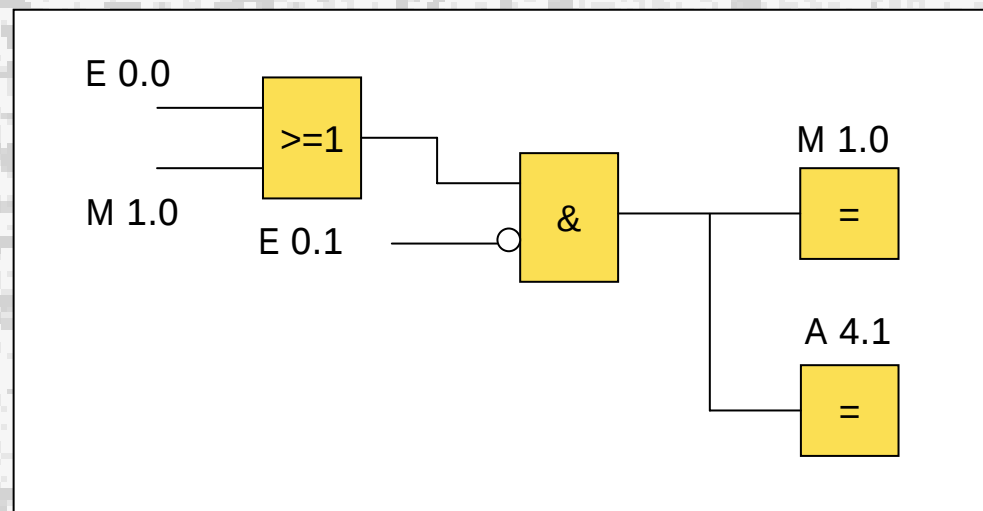
$$(E0.0 + M1.0) * \overline{E0.1} = M1.0$$

$$(E0.0 + M1.0) * \overline{E0.1} = A4.1$$

Codice AWL

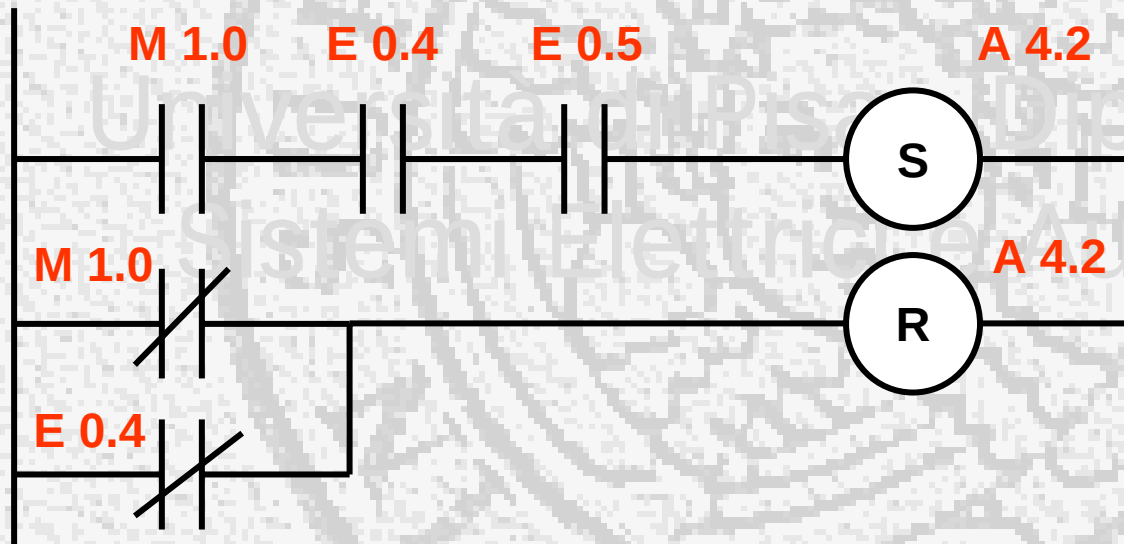
```
U(  
O E0.0  
O M1.0  
)  
UN E0.1  
=M1.0  
=A4.1
```

Schema FBD



Selezione modalità Manuale

Viene selezionata la modalità manuale se l'impianto è attivo, se il selettore è in modalità manuale e se si preme il pulsante di selezione. Viene annullata la modalità manuale se l'impianto è spento o se è selezionata la modalità automatica.



Funzione Logica

$$M1.0 * E0.4 * E0.5 = S(A\ 4.2)$$

$$\overline{(M1.0 + E0.4)} = R(A\ 4.2)$$

Selezione modalità Manuale

Funzione Logica

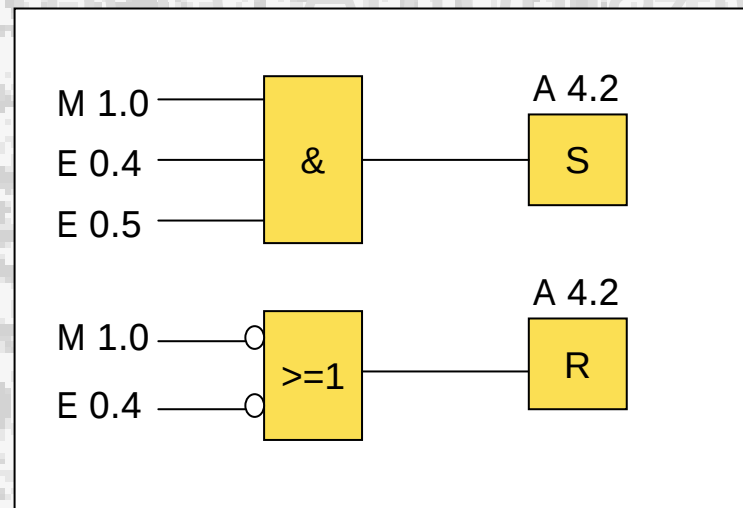
$M1.0 * E0.4 * E0.5 = S(A\ 4.2)$

$(M1.0 + E0.4) = R(A\ 4.2)$

Codice AWL

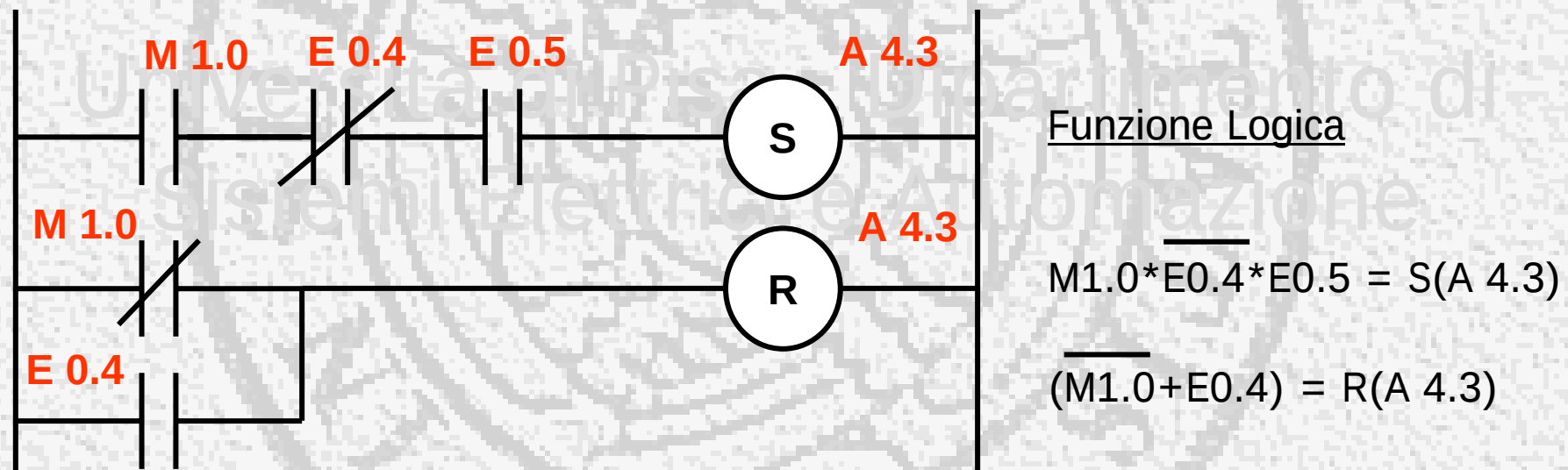
U M1.0
U E0.4
U E0.5
S A4.2
ON M1.0
ON E0.4
R A4.2

Schema FBD



Selezione modalità Automatica

Viene selezionata la modalità automatica se l'impianto è attivo, se il selettore è in modalità automatica e se si preme il pulsante di selezione. Viene annullata la modalità manuale se l'impianto è spento o se è selezionata la modalità manuale.



Selezione modalità Manuale

Funzione Logica

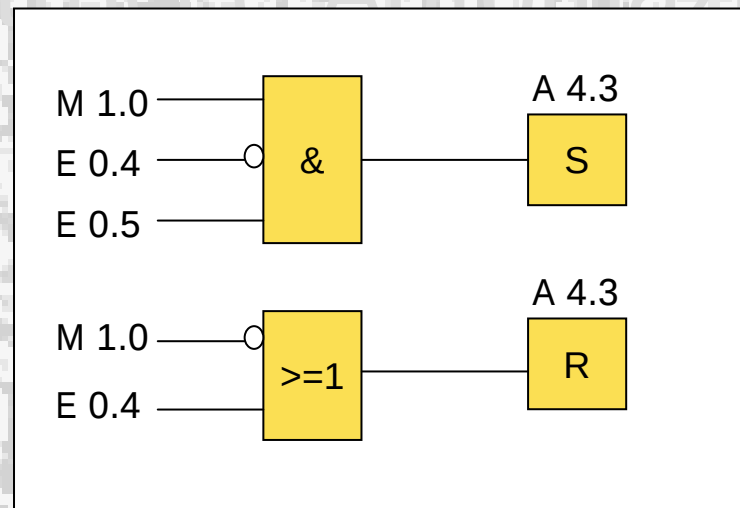
$$\overline{M1.0} * \overline{E0.4} * E0.5 = S(A\ 4.3)$$

$$\overline{(M1.0 + E0.4)} = R(A\ 4.3)$$

Codice AWL

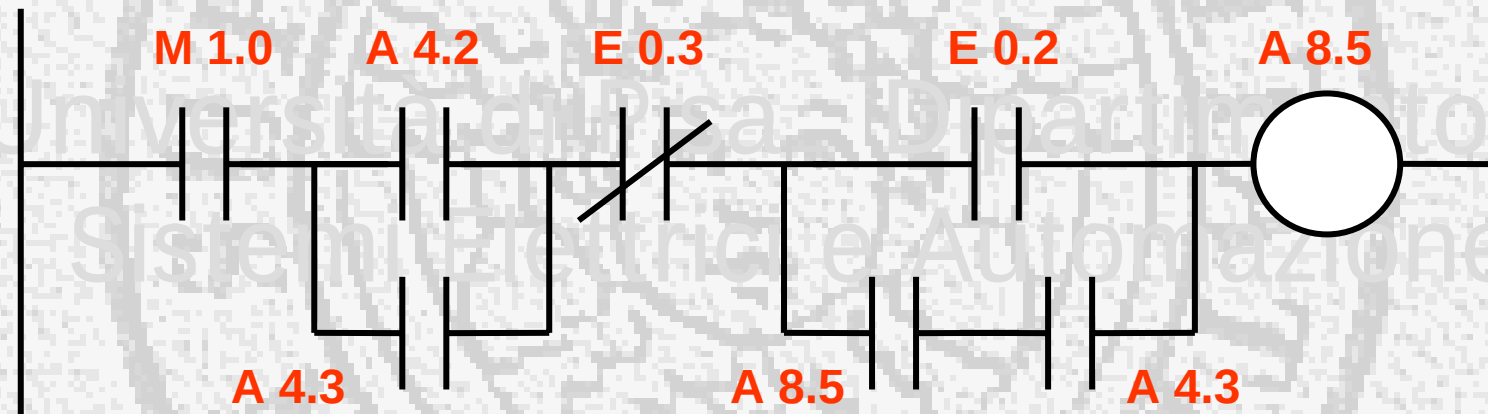
U M1.0
UN E0.4
U E0.5
S A4.3
ON M1.0
O E0.4
R A4.3

Schema FBD



Movimento Avanti

Si ha il movimento avanti se l'impianto è attivo, è selezionata una delle modalità, non è selezionato il movimento indietro, è selezionato il movimento in avanti. Se è stata selezionata la modalità automatica il movimento avanti deve rimanere attivo anche dopo che si rilascia il pulsante avanti.



Funzione Logica

$$M1.0 \cdot (A4.2 + A4.3) \cdot E0.3 \cdot (E0.2 + (A8.5 \cdot A4.3)) = A8.5$$

Movimento Avanti

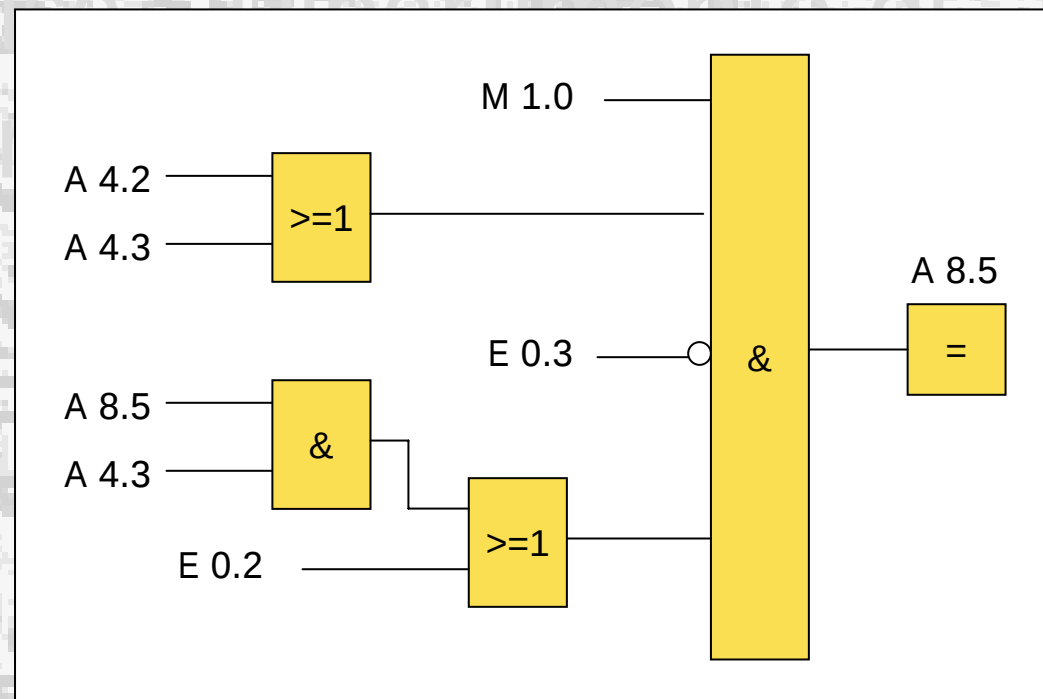
Funzione Logica

$$M1.0 * (A4.2 + A4.3) * \overline{E0.3} * (E0.2 + (A8.5 * A4.3)) = A8.5$$

Schema FBD

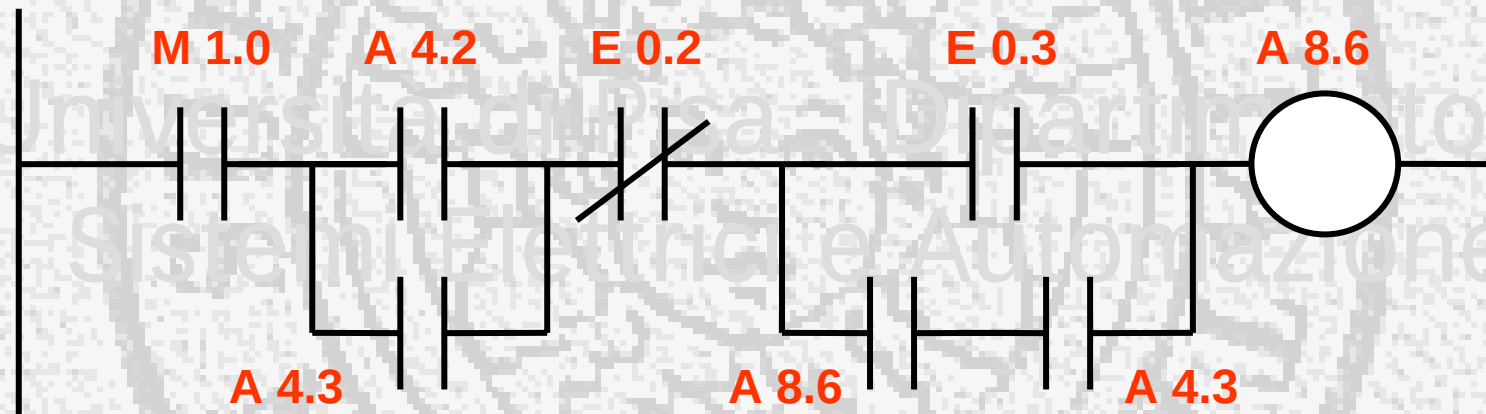
Codice AWL

U M1.0	O E0.2
U(	O(
O A4.2	U A8.5
O A4.3	U A4.3
)	)
UN E0.3	)
U(	= A8.5



Movimento Indietro

Si ha il movimento indietro se l'impianto è attivo, è selezionata una delle modalità, non è selezionato il movimento avanti, è selezionato il movimento indietro. Se è stata selezionata la modalità automatica il movimento indietro deve rimanere attivo anche dopo che si rilascia il pulsante avanti.



Funzione Logica

$$M1.0 \cdot (A4.2 + A4.3) \cdot E0.2 \cdot (E0.3 + (A8.6 \cdot A4.3)) = A8.6$$

Movimento Indietro

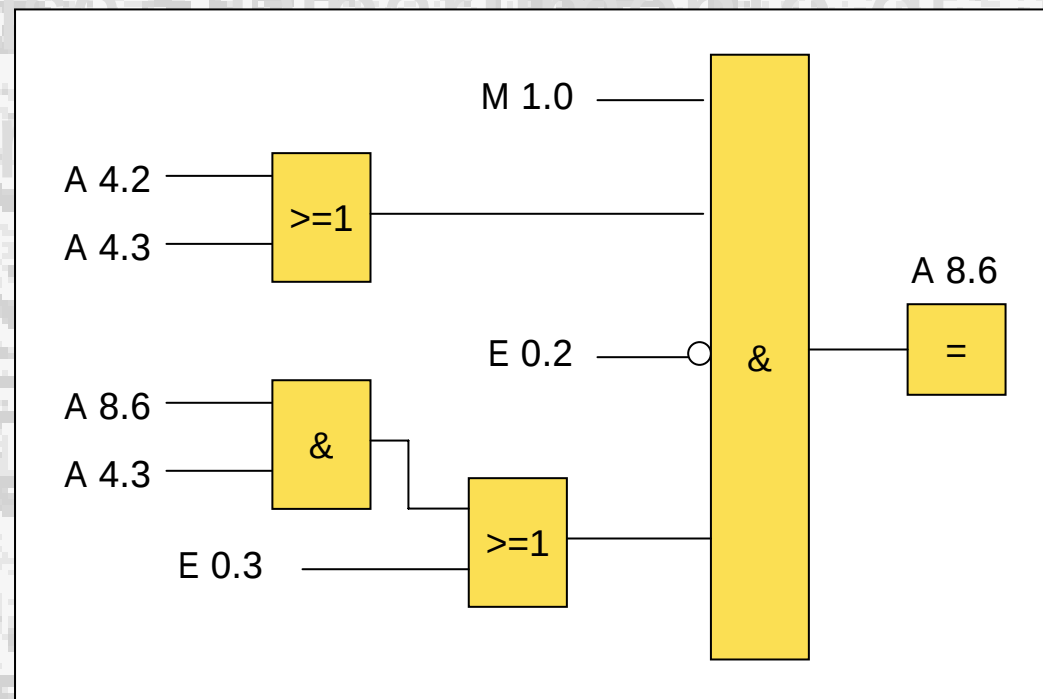
Funzione Logica

$$M1.0 * (A4.2 + A4.3) * \overline{E0.2} * (E0.3 + (A8.6 * A4.3)) = A8.6$$

Schema FBD

Codice AWL

U M1.0	O E0.3
U(	O(
O A4.2	U A8.6
O A4.3	U A4.3
)	)
UN E0.2	)
U(	= A8.6



Applicazione

- Nastro trasportatore

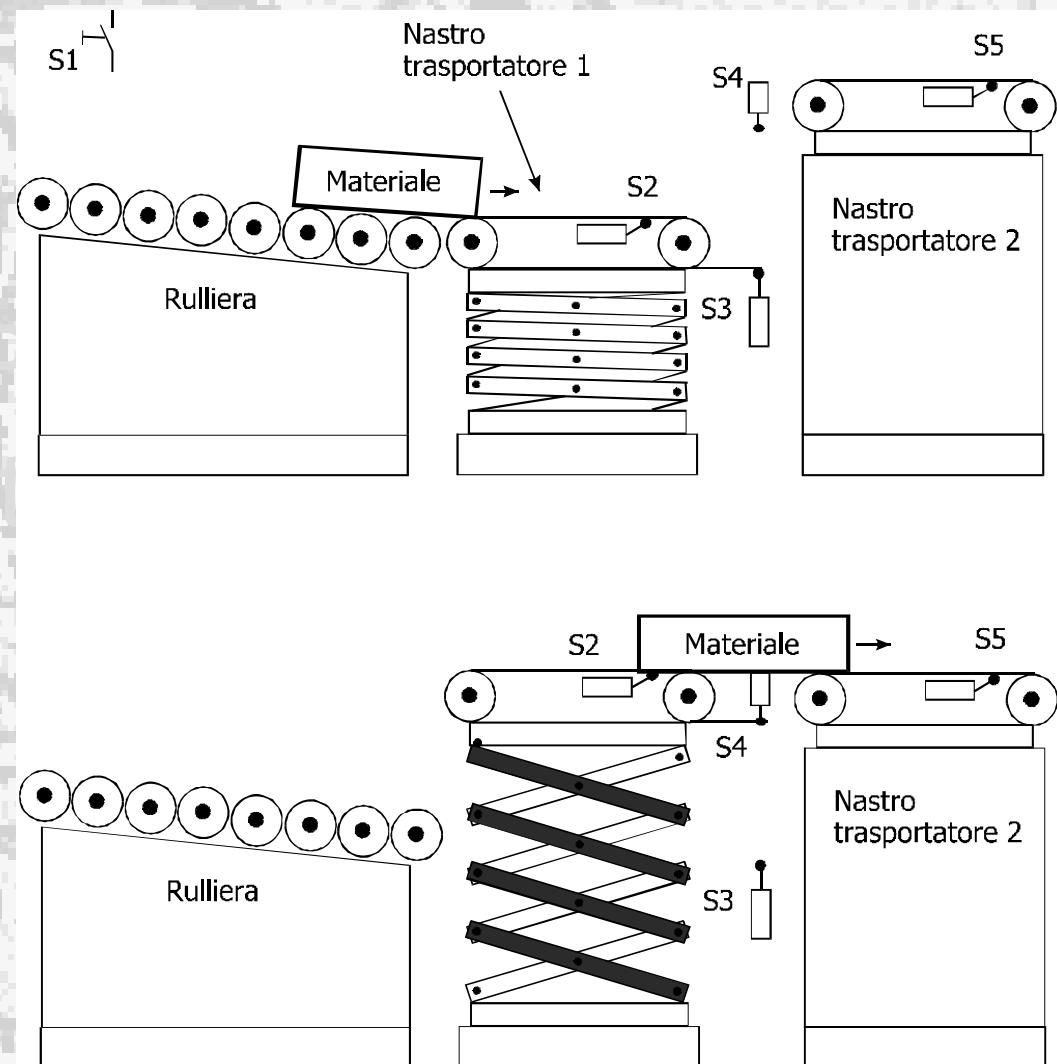
Schema ladder

Linguaggio testuale

Functional Block diagram

Descrizione dell'impianto

Si consideri il seguente processo industriale costituito da due nastri trasportatori e da un piano elevatore:



Descrizione dell'impianto

Quando viene premuto il pulsante S1, si inserisce il nastro trasportatore 1 del piano elevatore. Il materiale passa dalla rulliera inclinata sul nastro trasportatore 1.

Non appena il materiale aziona il finecorsa S2, il nastro trasportatore 1 viene arrestato ed il piano elevatore si solleva.

Quando viene azionato il finecorsa S4, il piano elevatore si ferma e i nastri trasportatori 1 e 2 si mettono in movimento.

Quando il materiale aziona l'interruttore di finecorsa S5, ambedue i nastri si arrestano; il piano elevatore si abbassa fino a raggiungere la posizione di attivazione del finecorsa S3.