4RSAA

Regola del Gottardo

Applicata in TIA PORTAL V21
Vol. 1
Prima edizione 2026

Questa pubblicazione è rivolta alla comunità scientifica, quindi, non è un libro didattico utile all'apprendimento della programmazione dei PLC di Siemens tramite TIA PORTAL.
Lo scopo è quello, tramite un'edizione monografica, di definire un assioma, svilupparne le implicazioni teoriche e dimostrare i teoremi derivati.

Scritto edito e pubblicato

Da

ing. Prof. Dott. Marco Gottardo PhD

Collana di pubblicazioni per l'automazione industriale.

Regola del Gottardo

Marco Gottardo, C.F. GTTMRC68R06G224I,
Via Colombo 14, 30030 Vigonovo (VE) Italia.
E-mail: ad.noctis@gmail.com

ISBN-13: 9798251990522
Casa editrice: G-Tronic di Marco Gottardo editore

Indice:

PREFAZIONE 4
Inquadramento formale del problema *5*
Assiomatizzazione in Algebra Booleana *6*
FORMALIZZAZIONE DELL'AUTORITENUTA 6
Equazione ricorsiva dello stato *6*
Prima dimostrazione della regola del Gottardo *7*
Proprietà strutturale: priorità del reset *8*
Confronto con latch SR classico *8*
Forma Canonica di Shannon *9*
FORMA MINIMALE 9
Interpretazione nel PLC Siemens (TIA Portal) *9*
Dimostrazione di stabilità del sistema *10*
Teorema di Sicurezza (formulazione) *10*
Estensione: forma matriciale nello spazio di stato *11*
Sintesi didattica applicativa *11*
Applicazione in campo partendo dalle morsettiere d'ingresso *12*
Gli elementi e i simboli fondamentali della logica funzionale *15*
Regola del Gottardo con dimostrazione *16*
DIMOSTRAZIONE CON TECNICA CONTRONOMINALE 17
Enunciato tecnico corretto *19*
Analisi del perché serve l'inversione *19*
Forma corretta dell'autoritenuta *20*
Traduzione logia della dimostrazione *21*
Sintesi operativa (da tecnico impiantista) *22*
Dimostrazione in logica proposizionale rigorosa *23*
Tesi (Regola del Gottardo *24*
Caso con TON (come nello schema allegato) *25*
Errore tipico in TIA Portal *26*
Analisi critica del libro (ISBN 9798345038970) *30*
Valutazione complessiva *30*
Formalizzazione secondo IEC 61131-3 ..... *31*
Modello logico dell'auto-ritenuta *31*
Implementazione corretta in Structured Text *31*
Implementazione errata (violazione) *32*
FORMALIZZAZIONE DELLA REGOLA DEL GOTTARDO IN ALGEBRA BOOLEANA ASSIOMATICA 38
Modellazione come Sistema a Stati Finiti (FSM) *41*
Analisi con Doppio Canale – Categoria 3/4 ..... *43*
Conclusioni Ingegneristiche *45*
LA TESI DI GOTTARDO 46
Fondamenti Logico-Formali della Regola del Gottardo nei Sistemi PLC e di Sicurezza *46*
CONCLUSIONI GENERALI 52

Prefazione

La presente pubblicazione non è concepita come un'opera divulgativa o libro di testo, ma come un trattato accademico, o un volume monografico, finalizzato a introdurre e discutere una specifica proposizione teorica destinata alla valutazione della comunità scientifica.
L'obiettivo principale del lavoro è presentare e formalizzare il seguente assioma: "Quando un contatto è delegato allo sgancio di un'autoritenuta, esso viene acquisito nel software in uno stato opposto rispetto a quello rappresentato nel funzionale."
La trattazione ha lo scopo di analizzare tale proposizione, verificarne la coerenza logica e proporla come assioma all'interno del quadro teorico delineato dall'autore.
La formalizzazione dell'assioma costituisce il primo passo verso una possibile strutturazione più ampia del modello concettuale, all'interno del quale la proposizione potrà essere successivamente sviluppata e dimostrata in forma di teorema.
La presente opera intende inoltre stabilire una formulazione chiara e formalmente riconoscibile della tesi proposta[1], con l'intento di attribuirne la paternità in modo esplicito e tracciabile nel contesto della letteratura scientifica.
Pur adottando un linguaggio deliberatamente semplice e diretto, il contenuto mantiene il livello di precisione terminologica e di rigore argomentativo richiesto negli ambiti scientifici e accademici, al fine di garantire chiarezza espositiva senza compromettere la formalità del discorso teorico.

[1] **Tesi:** Formulazione per la definizione degli stati logici dei contatti di sgancio delle auto ritenute in base allo stato indicato negli schemi funzionali. Quando un contatto fisico in campo è delegato allo sgancio di un'autoritenuta, questo viene acquisito nel software al contrario di come è rappresentato nel funzionale. Ne è conseguenza la necessità di regole chiare e ben definite per la rappresentazione funzionale.
[2] Nello schema elettrico gli ingressi e le uscite sono mostrati come dei rettangoli, organizzati in righe di otto, ovvero un byte. Dispongono di un campo indirizzo e un campo per la Tags (etichetta simbolica della variabile). A colpo d'occhio, se il rettangolo è in basso nella pagina si tratta di ingressi se è in alto si tratta di uscite. Negli schemi elettrici non sono riportate le variabili interne quali Merker e quelle definite nei DB.
[3] In informatica implementare è sinonimo di realizzare, quindi programmare.
[4] I segnali di ingresso possono essere di natura monostabile o bistabile. I segnali monostabili sono impulsivi, ad esempio quelli generati dai pulsanti, che generano lo stato TRUE solo per il tempo in cui l'operatore permane con il dito. I segnali

Inquadramento formale del problema

Nel controllo industriale discreto, la funzione di autoritenuta (self-holding o seal-in) realizza una memoria bistabile comandata da due ingressi:

- S: comando di avviamento (SET)
- R: comando di arresto (RESET)
- Q: stato dell'uscita (bobina)

Nel formalismo PLC (ladder o FBD), la struttura classica è:

- Contatto NO, normalmente aperto in campo, di Start
- Contatto NC, normalmente chiusi in campo, di Stop
- Contatto NO di autoritenuta in parallelo allo Start
- Bobina di uscita

Un primo assioma, necessario all'applicazione della regola del Gottardo (terminologia didattica diffusa in ambito tecnico italiano) stabilisce:
In presenza simultanea dei comandi di eccitazione e diseccitazione, la diseccitazione deve prevalere.
In termini logici:

$$\boldsymbol{S = 1 \ \wedge \ R = 1 \Rightarrow Q = 0}$$

Questa è una regola di priorità del reset, fondamentale per motivi di sicurezza funzionale.

Assiomatizzazione in Algebra Booleana
Consideriamo l'algebra booleana assiomatica:

$$\mathcal{B} = (\boldsymbol{B}, +, \cdot, ', \mathbf{0}, \mathbf{1})$$

con i seguenti assiomi fondamentali:

1. **Commutatività**
$$\boldsymbol{A} + \boldsymbol{B} = \boldsymbol{B} + \boldsymbol{A}; \boldsymbol{A} \cdot \boldsymbol{B} = \boldsymbol{B} \cdot \boldsymbol{A}$$

2. **Associatività**
$$(\boldsymbol{A} + \boldsymbol{B}) + \boldsymbol{C} = \boldsymbol{A} + (\boldsymbol{B} + \boldsymbol{C})$$

3. **Distributività**
$$\boldsymbol{A} \cdot (\boldsymbol{B} + \boldsymbol{C}) = \boldsymbol{A} \cdot \boldsymbol{B} + \boldsymbol{A} \cdot \boldsymbol{C}$$

4. **Complementazione**
$$\boldsymbol{A} + \boldsymbol{A}' = \mathbf{1}; \boldsymbol{A} \cdot \boldsymbol{A}' = \mathbf{0}$$

5. **Identità**
$$\boldsymbol{A} + \mathbf{0} = \boldsymbol{A}; \boldsymbol{A} \cdot \mathbf{1} = \boldsymbol{A}$$

Formalizzazione dell'autoritenuta
Equazione ricorsiva dello stato

Lo stato dell'uscita al tempo $\boldsymbol{k} + \mathbf{1}$**è:**

$$\boldsymbol{Q_{k+1}} = (\boldsymbol{S} + \boldsymbol{Q_k}) \cdot \boldsymbol{R}'$$

Questa è la formalizzazione canonica dell'autoritenuta con priorità al reset.
Interpretazione:

- $\boldsymbol{S} + \boldsymbol{Q_k}$→ eccitazione iniziale o mantenimento
- $\boldsymbol{R}'$→ condizione di non reset

Prima dimostrazione della regola del Gottardo

Vogliamo dimostrare formalmente:

$$\boldsymbol{S = 1 \land R = 1 \Rightarrow Q_{k+1} = 0}$$

Sostituiamo nell'equazione:

$$\boldsymbol{Q_{k+1} = (1 + Q_k) \cdot 1'}$$

Poiché:

$$\boldsymbol{1 + Q_k = 1}$$

e

$$\boldsymbol{1' = 0}$$

segue:

$$\boldsymbol{Q_{k+1} = 1 \cdot 0 = 0}$$

Dimostrato.

Proprietà strutturale: priorità del reset

Dimostriamo ora che la funzione è equivalente a una forma esplicita di priorità:

$$\boldsymbol{Q_{k+1} = S \cdot R' + Q_k \cdot R'}$$

Applicando la distributività:

$$\boldsymbol{(S + Q_k) \cdot R' = S \cdot R' + Q_k \cdot R'}$$

Raccogliendo:

$$\boldsymbol{Q_{k+1} = R' \cdot (S + Q_k)}$$

Osservazione:
Se $\boldsymbol{R = 1 \Rightarrow R' = 0}$, allora:

$$\boldsymbol{Q_{k+1} = 0}$$

indipendentemente da Se $\boldsymbol{Q_k}$.
Questa è la formalizzazione matematica della regola del Gottardo.

Confronto con latch SR classico

Latch SR senza priorità:

$$\boldsymbol{Q_{k+1} = S + Q_k \cdot R'}$$

Se $\boldsymbol{S = R = 1}$:

$$\boldsymbol{Q_{k+1} = 1 + Q_k \cdot 0 = 1}$$

Qui prevale il SET → non conforme alla regola del Gottardo.

Forma Canonica di Shannon

Applichiamo l'espansione rispetto a R:

$$Q_{k+1} = R \cdot f_{R=1} + R' \cdot f_{R=0}$$

Calcoliamo:

- **Se** $R = 1 \Rightarrow Q_{k+1} = 0$
- **Se** $R = 0 \Rightarrow Q_{k+1} = S + Q_k$

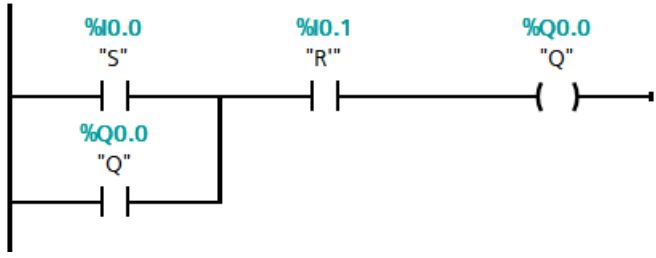
Quindi:

$$Q_{k+1} = R \cdot 0 + R' \cdot (S + Q_k)$$
$$Q_{k+1} = R' \cdot (S + Q_k)$$

Forma minimale

Interpretazione nel PLC Siemens (TIA Portal)

Nel ladder di TIA Portal:



La struttura implementa esattamente:

$$Q = (S + Q) \cdot R'$$

Nel ciclo di scansione:

1. Lettura ingressi
2. Valutazione logica
3. Aggiornamento immagine uscite

La priorità del reset è garantita strutturalmente, non temporalmente.

Dimostrazione di stabilità del sistema

Cerchiamo gli stati stazionari:

$$\boldsymbol{Q_{k+1} = Q_k = Q}$$
$$\boldsymbol{Q = (S + Q) \cdot R'}$$

Caso 1: $R = 1$

$$\boldsymbol{Q = 0}$$

Unico equilibrio.
Caso 2: $R = 0$

$$\boldsymbol{Q = S + Q}$$

Se $\boldsymbol{S = 0 \Rightarrow Q = Q}$(memoria)
Se $\boldsymbol{S = 1 \Rightarrow Q = 1}$
Sistema coerente.

Teorema di Sicurezza (formulazione)
Teorema:
La funzione booleana

$$\boldsymbol{Q_{k+1} = (S + Q_k) \cdot R'}$$

è l'unica funzione minimale a tre variabili che soddisfa:

1. Autoritenuta
2. Reset dominante
3. Assenza di stati proibiti
4. Idempotenza rispetto a $R = 1$

Dimostrazione (sintetica):
Imponendo i vincoli nella tabella di verità e minimizzando tramite mappe di Karnaugh si ottiene un'unica copertura implicante minima equivalente alla forma sopra riportata.

Estensione: forma matriciale nello spazio di stato
Possiamo scrivere:

$$\boldsymbol{Q_{k+1} = f(Q_k, S, R)}$$

Sistema dinamico discreto non lineare su campo booleano:

$$\boldsymbol{Q_{k+1} = R'S + R'Q_k}$$

Strutturalmente equivalente a:

$$\boldsymbol{Q_{k+1} = R'(S + Q_k)}$$

che rappresenta un sistema logico del primo ordine con ingresso di annullamento dominante.

Sintesi didattica applicativa

La regola del Gottardo non è un artificio grafico ladder, ma:

- una proprietà strutturale della funzione booleana
- un vincolo di priorità nel dominio discreto
- una condizione di sicurezza funzionale
- una scelta di progettazione coerente con normative di sicurezza macchina

La sua formalizzazione assiomatica consente:

- analisi formale
- dimostrazione di correttezza
- verifica simbolica
- sintesi automatica

Applicazione in campo partendo dalle morsettiere d'ingresso.

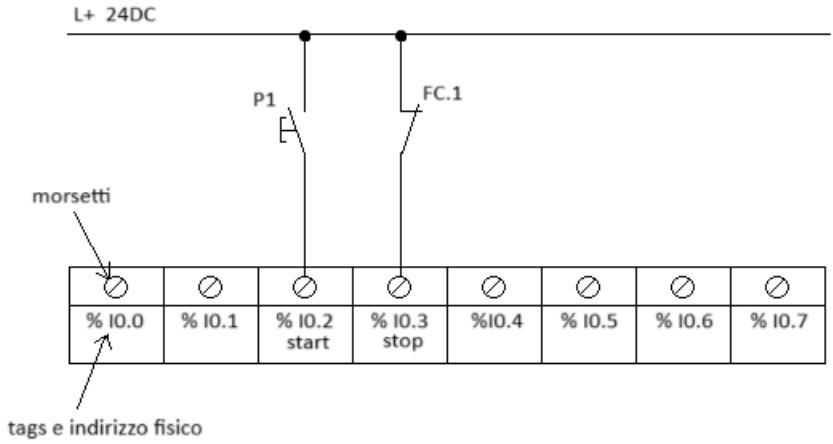
Il lavoro del programmatore inizia con una fase d'intervista, in cui si raccolgono, documentano e ufficializzano, tramite firme del committente, le specifiche tecniche dell'impianto da realizzare.

Queste non dovrebbero cambiare in corso d'opera, cosa che però puntualmente avviene, scombussolando in itinere il lavoro già svolto.

Quando s'inizia il lavoro di programmazione dell'impianto o della macchina, vanno forniti al programmatore due documenti, lo schema elettrico e il P&Id.

Il primo mostra le connessioni tra gli apparati sensoriali e gli ingressi del PLC nonché tra le uscite e gli attuatori o i loro organi di comando (relè, teleruttori, elettrovalvole, ecc.).

Nell'immagine è mostrato uno stralcio di un tipico schema elettrico[2] fornito al programmatore nella fase iniziale dello sviluppo del lavoro.

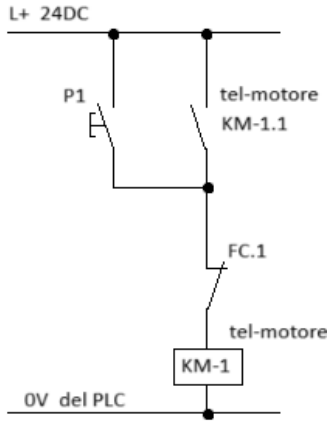


Il pulsante P1 porta il segnale del comando di marcia all'ingresso %I0.2, quindi il terzo bit della prima morsettiera di input, mentre il fine corsa elettromeccanico è connesso al successivo ingresso digitale %I0.3.

Probabilmente il programmatore realizza un "comando di auto ritenuta", che mantiene il movimento fino a limite assegnato.

Non dovrebbe comparire sullo schema elettrico quanto riportato in questo disegno:

[2] Nello schema elettrico gli ingressi e le uscite sono mostrati come dei rettangoli, organizzati in righe di otto, ovvero un byte. Dispongono di un campo indirizzo e un campo per la Tags (etichetta simbolica della variabile). A colpo d'occhio, se il rettangolo è in basso nella pagina si tratta di ingressi se è in alto si tratta di uscite. Negli schemi elettrici non sono riportate le variabili interne quali Merker e quelle definite nei DB.

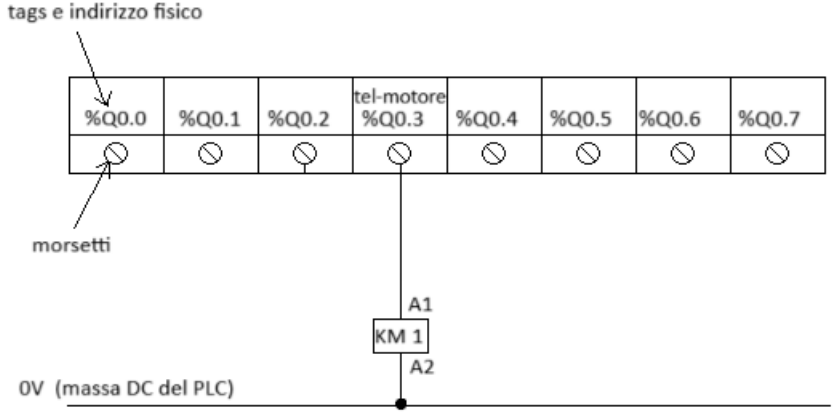


Questo non compare nello schema perché rappresenta la logica internamente implementata[3].
Si tratta di un Flip Flop elettromeccanico in grado di memorizzare un bit fintanto che non si realizza l'evento di sgancio.
In questo caso, il comando monostabile[4] di marcia è memorizzato nel bit interno KM-1, dichiarato preferibilmente all'interno di un Data Block, piuttosto che come Merker nella Tags Table.

Alla pressione del pulsante P1, il segnale positivo, a 24V DC, viene portato al nodo sottostante.
Al nodo Incontra un dispositivo elettromeccanico cablato normalmente chiuso, quindi il segnale lo attraversa eccitando la bobina.
La bobina eccitata chiude il suo contatto posto in parallelo al pulsante di comando.
L'operatore può rilasciare il pulsante ma la bobina viene ritenuta dal suo stesso contatto posto in parallelo.
Una qualche parte mobile, della macchina, andrà a intercettare FC.1 che tagliando la linea sgancia l'auto ritenuta.
Come avremo occasione di vedere in più parti del testo, questa simbologia, pur avendo forti somiglianze, non è uno schema elettrico e assume il nome di schema funzionale.
Molte cose, che sarebbero d'obbligo per gli schemi elettrici sono trascurabili negli schemi funzionali, ad esempio molte numerazioni.
Non ha senso numerare un cavo che in realtà è inesistente dato che viene rappresentato tramite software.
Gli stessi collegamenti hanno un diretto riscontro con il software, ad

[3] In informatica implementare è sinonimo di realizzare, quindi programmare.
[4] I segnali di ingresso possono essere di natura monostabile o bistabile. I segnali monostabili sono impulsivi, ad esempio quelli generati dai pulsanti, che generano lo stato TRUE solo per il tempo in cui l'operatore permane con il dito. I segnali bistabili invece sono tipici degli interruttori. Ad ogni azione corrisponde una transizione da uno stato stabile a quello complementare ove vi rimane fino a una successiva azione dell'operatore. In automazione si predilige l'azione monostabile e l'eventuale auto ritenuta realizzato nel software cosi da rendere gli automi auto azionanti quando si verificano le condizioni. Inoltre i sistemi auto ritenuti sono più idonei agli sganci automatici durante gli interventi di sicurezza.

esempio il parallelo rappresenta l'operazione booleana[5] OR e la serie l'operazione AND.
Lo schema elettrico, delle uscite, da un significato al collegamento della bobina del teleruttore.



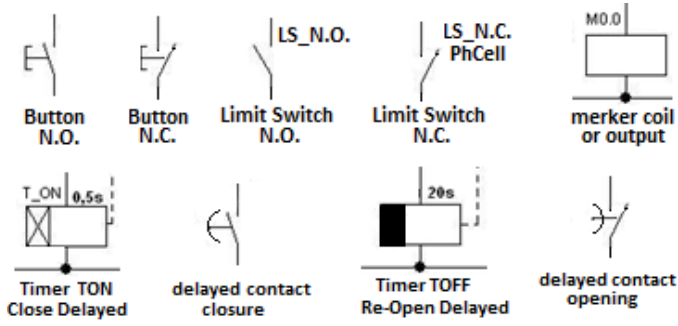
È in questo schema che il tratto di filo compreso tra il morsetto %Q0.3 e il morsetto A1 del teleruttore va numerato, anche se qui non presente per semplicità espositiva.
Gli stessi morsetti A1 e A2 risulteranno nello schema reale rappresentati in modalità standard come il "pallino barrato" a cui si abbina la dicitura coerente con il resto dello schema, ad esempio X1-1 ecc.

[5] Le operazioni booleane sono quelle contemplate nell'algebra di Boole. George Boole fu un matematico e logico inglese nato nel 1815, molto prima dell'avvento dei computer. Egli applicò la normale algebra ai sistemi di numerazione in base due, postulando alcuni elementi mancanti ed essenziali, come ad esempio la tautologia, che permette la semplificazione dei segnali ridondanti. Facendo un esempio, è una tautologia l'espressione "oggi piove oppure non piove", realizzabile tramite il parallelo "OR" delle due possibilità esprimibili con lo stesso contatto nella situazione normalmente aperta o normalmente chiusa. Dato che questo parallelo è sempre vero (TRUE), può essere eliminato o meglio sostituito con un filo che dalla linea di potenza +24V D.C. si collega direttamente alla bobina del relè di chiusura dell'espressione. Si implementa così la più importante delle minimizzazioni booleane, come si può incontrare nella tecnica delle mappe di Karnaugh. A compendio delle regole di base delle logiche booleane vi è il contributo di un altro scienziato, De Morgan, che introduce le uguaglianze e quindi la sostituibilità tra OR e AND se soggetti a opportuni vincoli di negazione dei segnali.

Gli elementi e i simboli fondamentali della logica funzionale.
La logica funzionale esprime le azioni che il controllore deve intraprendere ed è un utile ausilio per lo sviluppo del software. Gli elementi di base sono:

1. Linea di Potenza, orizzontale in alto, indicata con L+ oppure L se alimentata in continua o in alternata. Non essendo un circuito elettrico ma un algoritmo da implementare nella logica interna del PLC, tramite il software, è più sensato sia sempre continua.
2. Linea di ritorno, indicata con M (massa come chiusura di una linea L+) oppure N (neutro come chiusura di una linea in alternata).
3. Calate, che derivano verso il basso il flusso logico. Nel vero funzionale ogni segmento parte da una calata e non deriva orizzontalmente dal segmento precedente.
4. Contatto in chiusura, (ovvero cablato normalmente aperto), con aspetto di contatto pulito disegnato a sinistra della calata.
5. Contatto in apertura (ovvero cablato normalmente chiuso), con aspetto di contatto pulito disegnato alla destra della calata.
6. Bobina temporizzata in ritardo all'eccitazione e suo contatto in apertura e in chiusura.
7. Bobina temporizzata in ritardo alla diseccitazione e suo contatto in apertura e chiusura.
8. Contatore, rappresentato come bobina di carico rettangolare ma con gli ingressi up, down, set e reset. Va inoltre rappresentato il valore di preset.

I simboli utili per la stesura di un funzionale sono riassunti nell'immagine sottostante. Come possiamo vedere, i triangoli sui fine corsa come le barrette di chiusura non sono essenziali dato che i vari contatti rappresentano solo lo stato booleano di un bit interno o presente alla morsettiera, non l'oggetto elettromeccanico in campo.

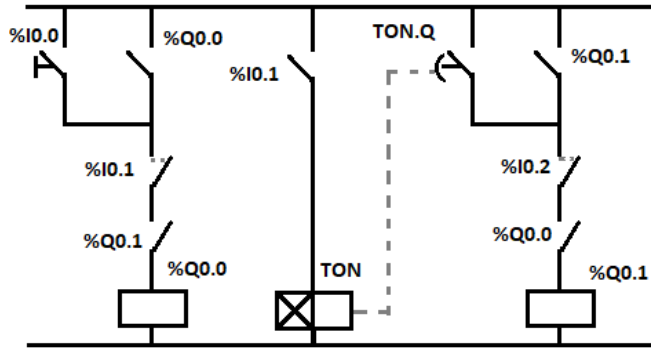


Dato che, in questo ambito, lo schema funzionale non è uno schema elettrico ma un algoritmo a cui attenersi per implementare il programma Ladder, la quantità di simboli e le regole del disegno si semplificano notevolmente.

In primo luogo i simboli presenti nel funzionale rappresentano i segnali che gli oggetti elettromeccanici forniscono.

Lo schema funzionale è una rappresentazione schematica e intuitiva del programma che dovrà essere implementato con il software. Svolge la medesima funzione del diagramma di flusso, "flow chart"[6], nei linguaggi a alto livello.

Solo in prima approssimazione il funzionale ruotato di 90 gradi rappresenta l'implementazione del software. In generale questo non è vero. A tal proposito va infatti considerata la **Regola del Gottardo.**

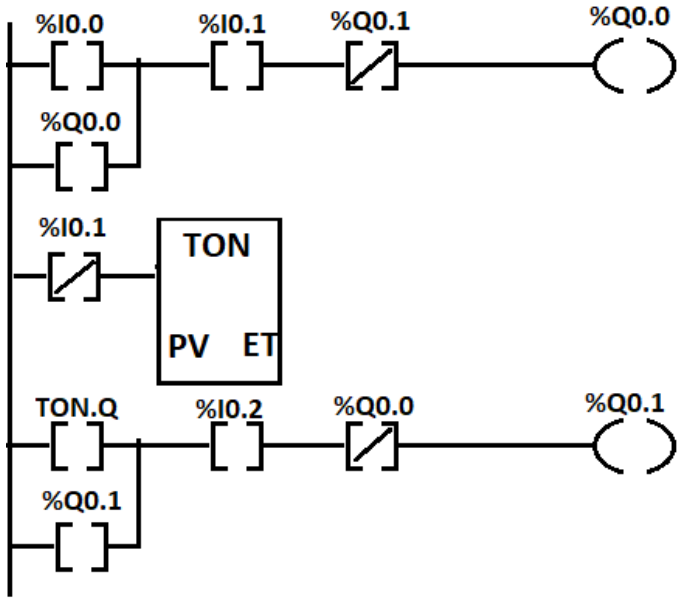


Regola del Gottardo: Quando un contatto* è delegato allo sgancio di un'auto ritenuta, viene acquisito nel software al contrario di come è rappresentato nel funzionale.

***si intende contatto fisico di tipo fine corsa elettromeccanico. Quando il medesimo segnale perviene da una virtualizzazione o è in qualche modo simulato la regola ne viene confermata in quanto l'eventuale contatto negato presente allo sgancio non è un contatto reale o non è un fine corsa elettromeccanico.**

[6] I flow chart non devono essere applicati nella fase di sviluppo dei programmi dei PLC perché inadeguati. Questi sono sostituiti, ove possibile, con gli schemi funzionali. I set di simboli per i flow chart sono ovale per partenze e arrivi, rettangoli di vario tipo per le azioni, rombi per i decisionali di tipo IF, frecce per indicare le direzioni dei flussi. Per analogia gli schemi, o diagrammi, funzionali hanno un loro set limitato di simboli, il contatto aperto a sinistra della calata, il contatto chiuso a destra della calata, la bobina, le bobine temporizzate, mentre le logiche sono realizzate con paralleli per le OR e serie per le AND.

Lo schema funzionale proposto porta all'implementazione del codice standardizzato IEC 61131 sotto riportato.
L'attenzione dovrebbe essere raccolta dai contatti %I0.1 e %I0.2 che nel funzionale sono mostrati sul lato destro della calata, quindi normal 1 -> pressed 0.
In questi segmenti Ladder sono invece presentati normalmente aperti, e si trova spiegazione nella appena citata "regola del Gottardo"[7].
La necessità della regola nasce dopo una trentennale esperienza in ambito didattico e non è meno banale dell'affermare che due rette parallele non si incontrano mai.



Nota sulla sicurezza intrinseca:
Un sistema in autoritenuta si trova in sicurezza intrinseca quando la rottura o la mancanza del sensore di fine corsa invia al controllore il medesimo stato dell'intervento.

Dimostrazione con tecnica contronominale: Per la Regola del Gottardo applichiamo la tecnica di dimostrazione contronominale che consiste nel partire dalla negazione della tesi per arrivare alla negazione dell'ipotesi, e non a una contraddizione con l'ipotesi.
Sia la **tesi** la necessita di negare il contatto di sgancio dell'auto ritenuta

[7] Fintanto che non se ne dimostri il contrario l'autore usa lo status di ricercatore Ph.D. per validarne l'esistenza e la dimostrazione qui formulata.

rispetto a come è rappresentato nel funzionale.
La negazione della tesi è che non sia negato lo stato del contatto fisico in campo.

L'**ipotesi** è però che i contatti fisici in campo delegati allo sgancio dell'auto ritenuta siano cablati normalmente chiusi.

Ipotizziamo, per assurdo, di negare lo stato in campo del contatto fisico delegato allo sgancio.

Si osserva che nessun segnale perviene al morsetto delegato alla sua lettura.

Essendo questo connesso, secondo la logica interna del software, tramite combinazione booleana AND, il risultato combinatorio, qualunque sia lo stato del pulsante di start, sarà zero, impedendo l'aggancio della bobina in auto ritenuta.

La **negazione dell'ipotesi** è che non bisogna negare lo stato del contatto di sgancio delegato all'auto ritenuta rispetto a come questo sia mostrato nel funzionale.

Non negando lo stato di questo contatto esso implementa il teorema della negazione doppia booleana[8] realizzato dall'affermazione "non(non(sgancio) = sgancio", e quindi il motore non si avvia in quanto risulta sganciato.

la doppia negazione permette di eliminare dall'espressione una negazione, ovvero porre lo stato del contatto nel segmento ladder step 7 aperto anziché chiuso, validando definitivamente la regola del Gottardo, fino a prova contraria.

C.V.D.

Nota bene: L'ipotesi cita "contatti chiusi in campo" quindi non si deve applicare quando i contatti suddetti non esistano, ovvero quando siano virtualizzati in un pannello HMI.

Dimostrazione del contrario: Il contatto di interblocco non va invertito in quanto non soddisfa le ipotesi iniziali dimostrando la tesi.

[8] Dimostrabile sia tramite i sistemi deduttivi di Hilbert che in logica proposizionale classica anche con il solo ausilio delle tautologie. Secondo le regole della doppia negazione, teoremizzate nelle algebre booleane e combinatorie allora la doppia negazione permette di eliminare dall'espressione una negazione validando definitivamente la regola del Gottardo, fino a prova contraria.

Enunciato tecnico corretto

Nel caso di **contatto fisico di finecorsa elettromeccanico cablato normalmente chiuso (NC)** e delegato allo **sgancio di un'auto-ritenuta**, vale la seguente regola:

Nel software del PLC il contatto deve essere programmato negato rispetto alla sua rappresentazione funzionale.

In pratica:

- In campo → contatto NC (chiuso a riposo).
- In Ladder → va inserito come contatto normalmente aperto (—| |—) associato al bit di ingresso.

Questo perché il PLC legge **livelli logici elettrici**, non la simbologia funzionale.

Analisi del perché serve l'inversione

Stato fisico reale (finecorsa NC)

Stato fisico	Morsetto PLC	Bit di ingresso
Non premuto	1	TRUE
Premuto	0	FALSE

Ma funzionalmente:

- Premuto = condizione di sgancio
- Non premuto = macchina abilitata

Quindi il significato funzionale è invertito rispetto al livello logico letto.

Forma corretta dell'auto-ritenuta

Equazione corretta:
Q_{n+1}= (START ∨ Q_n) ∧ INGRESSO

Dove:

- INGRESSO è il bit letto dal PLC (NC cablato).
- Quando il finecorsa viene premuto → INGRESSO = 0 → la AND annulla tutto → sgancio garantito.

Se il programmatore lo negasse nel software:
Q_{n+1}= (START ∨ Q_n) ∧ ¬INGRESSO

allora:

- A riposo → INGRESSO = 1 → ¬1 = 0 → la macchina non parte mai.
- Premuto → INGRESSO = 0 → ¬0 = 1 → la macchina potrebbe partire con finecorsa attivo (errore grave).

Ed è esattamente questo il cuore della dimostrazione contronominale riportata nel testo.

Corollario: Q_{n+1} rappresenta lo stato assunto dall'uscita nel ciclo di scansione successivo alla valutazione dello stato logico della locazione di memoria, secondo quanto previsto dal runtime del PLC. Nei sistemi PLC conformi al modello di esecuzione dei linguaggi della norma IEC 61131-3, gli ingressi sono campionati all'inizio del ciclo di scansione e copiati nell'immagine di processo degli ingressi, mentre le assegnazioni alle uscite aggiornano l'immagine di processo delle uscite durante l'esecuzione del programma. Solo al termine del ciclo l'immagine delle uscite viene trasferita ai morsetti fisici. Questo implica che il programma può osservare lo stato aggiornato di una bobina nello stesso ciclo, mentre il campo lo vedrà nel ciclo successivo.

Traduzione logica della dimostrazione

Ipotesi:
Il contatto fisico è NC in campo.

Tesi:
Non va negato nel software.

Dimostrazione contronominale:

- Se lo negassi nel software,
- allora impediresti l'aggancio della ritenuta oppure abiliteresti la macchina in condizione di sgancio,
- quindi l'ipotesi di corretto funzionamento sarebbe falsa,
- dunque non si deve negarlo.

Il riferimento alla doppia negazione:

$$\neg(\neg S) = S$$

serve solo a formalizzare che:

NC fisico + contatto NO in ladder = doppia negazione implicita → comportamento coerente.

Quando NON si applica

La nota del libro è corretta:

La regola vale **solo se il contatto è fisico elettromeccanico NC cablato in campo**.

Non si applica quando:

- Il segnale è virtuale (HMI).
- Il segnale è simulato.
- Il contatto è NO fisico.
- Si tratta di interblocchi logici interni.

Sintesi operativa (da tecnico impiantista)

Se il dispositivo di sicurezza è:

- NC fisico cablato → NON invertire nel PLC.
- NO fisico → valutare secondo funzione.
- Virtuale → nessuna regola del Gottardo.

Possiamo ora sviluppare i seguenti punti:

- riscrivere la dimostrazione in forma rigorosa di logica proposizionale pura (senza narrativa),
- Mostrare il caso con timer TON come nello schema presentato in precedenza
- Analizzare l'errore tipico in TIA Portal quando si confonde rappresentazione funzionale e stato elettrico.

fermo restando la regola del Gottardo, già presente nel testo con ISBN: 9798345038970, procediamo sviluppando l'analisi di tutti e tre i punti elencati.

Manteniamo come assioma la formulazione della Regola del Gottardo, già presente in molti testi e procediamo in modo strutturato sui tre punti richiesti.

Dimostrazione in logica proposizionale rigorosa

Definizioni formali

Sia:

- F = stato fisico del contatto NC in campo
  F=1 → circuito chiuso (condizione normale)
  F=0 → circuito aperto (finecorsa premuto)
- I = bit letto dal PLC
  Essendo cablato NC:

$$I=F$$

- Q = uscita in auto-ritenuta
- S = start

Auto-ritenuta corretta:

$$Q_{n+1}= (S \vee Q_n) \wedge I$$

Tesi (Regola del Gottardo)

Non bisogna negare I nel software.

Dimostrazione contronominale

Supponiamo falsa la tesi:
Si nega il contatto nel software.

Allora:

$$Q_{n+1} = (S \vee Q_n) \wedge \neg I$$

Caso 1 — condizione normale (F = 1 → I = 1):

$$Q_{n+1} = (S \vee Q_n) \wedge 0 = 0$$

La ritenuta non può agganciarsi → comportamento non conforme.

Caso 2 — finecorsa attivo (F = 0 → I = 0):

$$Q_{n+1} = (S \vee Q_n) \wedge 1$$

La macchina può avviarsi in condizione di sgancio → violazione funzionale grave.

Quindi la negazione della tesi implica negazione dell'ipotesi di corretto funzionamento.

C.V.D.

Caso con TON (come nello schema allegato)

Nel tuo schema compare un **TON** che ritarda lo sgancio o l'abilitazione.

Ricordiamo:

TON.Q=1 solo dopo PT con IN = 1

Se il finecorsa NC viene negato nel software:

- In condizioni normali → IN = 0 → TON non parte mai.
- In condizioni di sgancio → IN = 1 → TON parte quando non dovrebbe.

Conseguenze:

- Temporizzazioni invertite.
- Autoritenuta che si aggancia con finecorsa premuto.
- Ritardi di sgancio anziché di avviamento.

Con timer la violazione diventa ancora più insidiosa perché introduce dinamica temporale incoerente.

Errore tipico in TIA Portal

Errore frequente negli studenti:

Confondere:

- Simbolo grafico funzionale (NC disegnato)
- Stato logico letto dal PLC

In TIA:

- Il contatto —|/|— non significa “contatto NC fisico”.
- Significa “condizione logica negata”.

Molti programmatori:

1. Cablaggio NC in campo.
2. Inseriscono —|/|— in ladder.
3. Ottengono doppia inversione errata.

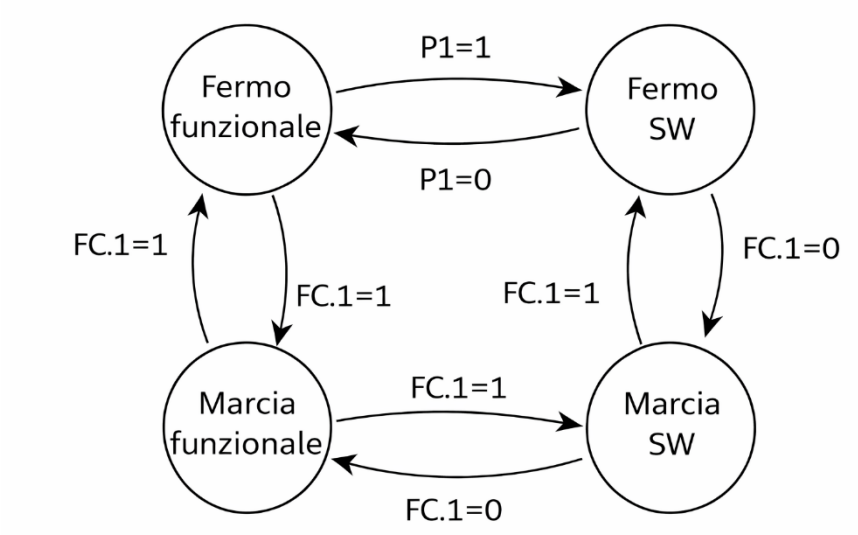
Effetto:

- Macchina che non parte.
- Oppure parte in condizione di sicurezza aperta.
- Diagnostica fuorviante.

La Regola del Gottardo serve esattamente a evitare questa ambiguità tra livello fisico e livello booleano.

Rappresentazione con automa a stati finiti

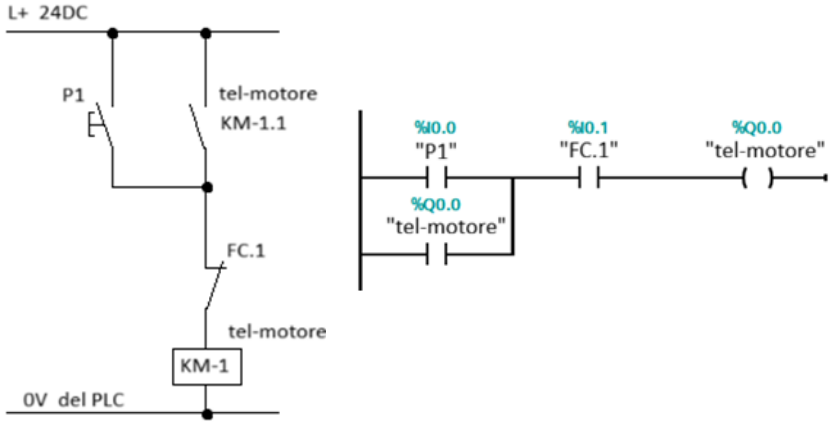
L'automa a stati finiti rappresentato nel grafo formalizza il comportamento di un circuito con **autoritenuta elettromeccanica** quando esso viene acquisito all'interno di un sistema di controllo programmabile secondo il principio denominato *regola del Gottardo*. Tale principio afferma che, qualora un contatto sia delegato allo sgancio di un'autoritenuta, la sua acquisizione logica nel software avviene con polarità inversa rispetto alla rappresentazione funzionale presente nello schema pseudo elettromeccanico.



Regola del Gottardo

Il sistema considerato è costituito da un circuito di comando di un teleruttore, indicato con KM-1, dotato di autoritenuta realizzata mediante un contatto ausiliario dello stesso teleruttore in parallelo al pulsante di avvio P1.
La catena di arresto è realizzata tramite il contatto FC.1 posto in serie alla bobina.
Nel dominio funzionale elettromeccanico il comportamento del sistema è di tipo bistabile: l'attivazione momentanea del comando di avvio eccita la bobina del teleruttore, che mantiene poi il proprio

stato tramite il contatto di autoritenuta; l'apertura del contatto di arresto interrompe invece la corrente nella bobina determinando il ritorno allo stato di riposo.
Quando tale comportamento viene implementato all'interno di un PLC, la logica deve essere interpretata alla luce del modello di scansione ciclica e dell'utilizzo delle immagini di processo degli ingressi e delle uscite.
In questo contesto l'automa introduce una distinzione concettuale tra lo **stato funzionale del circuito fisico** e lo **stato software interno al controllore**, producendo così quattro stati distinti che descrivono la combinazione delle due rappresentazioni.
Gli stati denominati *Fermo funzionale* e *Marcia funzionale* descrivono il comportamento atteso del sistema fisico, mentre gli stati *Fermo SW* e *Marcia SW* rappresentano lo stato della variabile logica interna che realizza l'autoritenuta nel programma PLC.
Nel funzionamento dell'automa, la pressione del pulsante P1 determina una transizione dal dominio funzionale di fermo allo stato software corrispondente, nel quale la logica di autoritenuta viene acquisita dal controllore.



Tale passaggio riflette il fatto che, durante il ciclo di scansione, l'attivazione dell'ingresso provoca l'assegnazione della bobina logica che implementa l'autoritenuta.

Una volta stabilita questa condizione, l'automa evolve verso lo stato di marcia, in cui sia il modello funzionale sia il modello software risultano coerenti nel rappresentare l'eccitazione del teleruttore.
La presenza del contatto FC.1 introduce invece la condizione di sgancio dell'autoritenuta. Nel dominio funzionale l'apertura del contatto comporta immediatamente l'interruzione della catena di alimentazione della bobina. Tuttavia, nel dominio software, tale evento è acquisito come variazione dell'ingresso che provoca la transizione verso lo stato di fermo soltanto attraverso l'aggiornamento della logica di autoritenuta nel ciclo di scansione. In questa fase emerge il principio espresso dalla regola del Gottardo: il contatto che funzionalmente rappresenta la condizione di arresto viene trattato nel software con polarità logica opposta rispetto alla rappresentazione elettrica, poiché la variabile interna che mantiene l'autoritenuta deve essere disabilitata quando il contatto risulta aperto.
L'automa rappresenta pertanto una formalizzazione del disallineamento semantico tra la rappresentazione elettromeccanica del circuito e la sua implementazione logica nel PLC.
Attraverso la distinzione tra stati funzionali e stati software è possibile descrivere in modo rigoroso il processo mediante il quale l'autoritenuta viene stabilita e successivamente sganciata, evidenziando come l'acquisizione degli ingressi e l'aggiornamento delle uscite nel ciclo di scansione introducano una trasformazione logica che giustifica l'inversione concettuale del contatto di arresto. In questo senso l'automa fornisce una base formale per interpretare la regola del Gottardo come proprietà emergente del modello di esecuzione dei controllori programmabili e della traduzione dei circuiti a relè in logica software.

Analisi critica del libro (ISBN 9798345038970)

Valutazione tecnica, non editoriale.

Punti di forza

- Introduce formalizzazione logica (contronominale, doppia negazione).
- Collega cablaggio fisico e algebra booleana.
- Stimola ragionamento strutturato, non solo “programmazione per imitazione”.
- Buono per studenti ITI/ITS che devono capire il perché.

Valutazione complessiva

Dal punto di vista tecnico:

- Concettualmente solido.
- Corretta impostazione logica.
- Utile per formazione professionale tecnica.

Dal punto di vista didattico:

- Buono per studenti motivati.
- Può risultare complesso per chi non ha basi di logica formale.
- Avrebbe beneficiato di maggiore rigore simbolico e meno retorica.

Non è un manuale industriale avanzato, ma per il target ITI/IPSIA/ITS è sopra la media dei testi introduttivi.

Formalizzazione secondo IEC 61131-3

La norma IEC 61131-3 definisce i linguaggi (LD, FBD, ST, SFC) e il modello di esecuzione ciclica.

Modello logico dell'auto-ritenuta

Variabili:

```
VAR
    Start     : BOOL;   // Pulsante NO
    StopNC    : BOOL;   // Ingresso da contatto NC fisico
    Motor     : BOOL;   // Uscita in autoritenuta
END_VAR
```

Assunzione fisica (Regola del Gottardo – ipotesi del libro):

- Contatto fisico STOP cablato NC.
- A riposo → ingresso = TRUE.
- Attivato → ingresso = FALSE.

Implementazione corretta in Structured Text

```
Motor := (Start OR Motor) AND StopNC;
```

Questa forma è:

- deterministica,
- reset-dominant,
- coerente con logica fail-safe.

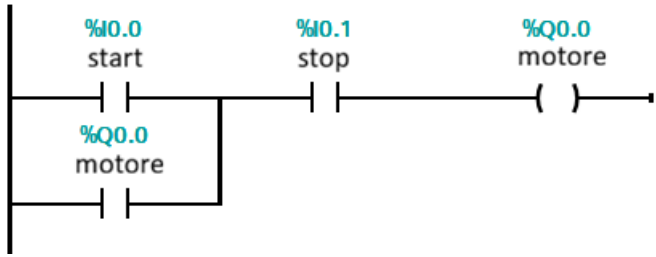
Implementazione errata (violazione)

```
Motor := (Start OR Motor) AND NOT StopNC;
```

Conseguenze:

- A riposo → NOT StopNC = FALSE → Motor non parte mai.
- In condizione di stop → NOT StopNC = TRUE → possibile avvio pericoloso.

Forma Ladder equivalente corretta



Nota: il contatto StopNC è **NO in ladder**, perché rappresenta il livello logico letto, non la natura fisica del dispositivo.

Tabella di verità completa

Consideriamo:

$$Q_{n+1} = (S \vee Q_n) \wedge I$$

Dove:

- S = Start
- I = StopNC
- Qn = stato precedente

Caso corretto

S	Q_n	I (NC)	Q_{n+1}	Significato
0	0	1	0	Fermo
1	0	1	1	Avvio
0	1	1	1	Ritenuta
1	1	1	1	Ritenuta
X	X	0	0	STOP sempre dominante

Proprietà fondamentale:

$$I = 0 \Rightarrow Q_{n+1} = 0$$

Reset dominante garantito.

Caso errato (negazione software)

$$Q_{n+1} = (S \vee Qn) \wedge \neg I$$

S	Q_n	I	Q_{n+1}
0	0	1	0
1	0	1	0 □
0	1	1	0 □
1	0	0	1 □
0	1	0	1 □

Violazioni:

- Impossibilità di avvio in stato normale.
- Possibile avvio con STOP attivo.

Analisi in ottica sicurezza funzionale

Riferimenti normativi:

- EN ISO 13849-1
- IEC 62061

Principio di sicurezza rilevante

Fail-safe principle:

La perdita di energia o l'apertura del circuito deve portare allo stato sicuro.

Con contatto NC:

- Rottura filo → ingresso = 0

- PLC legge FALSE
- AND annulla l'uscita
- Motore si arresta

Se si viola la Regola del Gottardo:

- Rottura filo → ingresso = 0
- NOT 0 = 1
- Possibile consenso alla marcia

Questo compromette:

- Categoria del circuito
- Performance Level (PL)
- Integrità funzionale

Dominanza del reset

Nei sistemi safety il reset deve essere:

- prioritario,
- non mascherabile,
- non autoripristinante.

La forma corretta:

$$Q = Fset \wedge Fsafe$$

dove:

$$Fsafe = \bigwedge_i Ii$$

Ogni ingresso di sicurezza è in AND globale.

Collegamento con PLC safety

In PLC fail-safe (es. CPU F):

- I contatti di sicurezza sono trattati come segnali “high = safe”.
- La negazione arbitraria nel software rompe la coerenza SIL/PL.

La Regola del Gottardo, in questo senso, non è solo didattica ma coerente con architetture certificate.

Valutazione tecnica approfondita del libro

Analisi più dettagliata.

Correttezza concettuale

Corretta interpretazione del rapporto:

- livello elettrico
- livello logico
- rappresentazione ladder

Corretta impostazione reset-dominant

Rigorosità formale

Parzialmente rigorosa:

- Usa contronominale correttamente.
- Ma manca formalizzazione sistematica con tabelle di verità.
- I richiami ai sistemi di Hilbert sono teoricamente corretti ma non sviluppati formalmente.

Livello didattico

Per target ITI/IPSIA/ITS:

- Superiore alla media.
- Stimola ragionamento logico.
- Introduce concetti di algebra booleana applicata.

Limite:

- Linguaggio talvolta ridondante.
- Formalismo non sempre lineare.

Valutazione complessiva

Dal punto di vista tecnico-professionale:

È un buon testo didattico operativo con ambizione logico-formale.

Non è un manuale di sicurezza funzionale avanzata, ma è ben strutturato per:

- comprendere l'autoritenuta,
- evitare errori tipici in PLC,
- sviluppare ragionamento booleano applicato.

Formalizzazione della Regola del Gottardo in Algebra Booleana Assiomatica

Definizioni e Ipotesi

Sia definita un'auto-ritenuta comandata da:

- $S \in \{0,1\}$ — comando di START (NO)
- $F \in \{0,1\}$ — stato fisico del contatto di sgancio NC
- $I \in \{0,1\}$ $I \in \{0,1\}$ — valore logico letto dal PLC
- $Q_n \in \{0,1\}$ — stato della bobina al ciclo n
- Q_{n+1} — stato al ciclo successivo

Ipotesi fisica (contatto NC cablato):

$$I - F$$

con:

- $F=1 \rightarrow$ contatto chiuso (condizione normale)
- $F=0 \rightarrow$ contatto aperto (sgancio)

Auto-ritenuta corretta

Definizione:

$$Q_{n+1} = (S \vee Q_n) \wedge I$$

Assiomi dell'Algebra Booleana utilizzati

Si considerano gli assiomi classici (Huntington):

1. Commutatività
   $A \vee B = B \vee A$
   $A \wedge B = B \wedge A$
2. Distributività
   $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$

3. Elementi neutri
   $A \wedge 1 = A$
   $A \vee 0 = A$
4. Elementi nulli
   $A \wedge 0 = 0$
   $A \vee 1 = 1$
5. Complementazione
   $A \wedge \neg A = 0$
   $A \vee \neg A = 1$

Proprietà di Reset Dominante

Proposizione 1
La funzione è reset-dominant rispetto a I.

Dimostrazione

Sia $I = 0$.

$Q_{n+1} = (S \vee Q_n) \wedge 0$

Per assioma 4:

$A \wedge 0 = 0$

quindi:

$Q_{n+1} = 0$

Indipendentemente da S e Qn

C.V.D.

Violazione della Regola

Supponiamo la negazione software:

$Q_{n+1} = (S \vee Q_n) \wedge \neg I$

Sia I = 1 (condizione normale).

$Q_{n+1} = (S \vee Q_n) \wedge 0 = 0$

La funzione perde la possibilità di set.

Sia I = 0 (sgancio).

$Q_{n+1} = (S \vee Q_n) \wedge 1 = S \vee Q_n$

Il reset non è più dominante.

Conclusione formale:
La negazione software distrugge la proprietà di dominanza del reset.

Modellazione come Sistema a Stati Finiti (FSM)

Definizione del sistema

Definiamo la macchina a stati:

$M = (X,U,\delta)$

dove:

- $X = \{OFF,ON\}$
- $U = \{S,I\}$
- $\delta : X \times U \rightarrow X$

2.2 Funzione di transizione corretta

Stato attuale	S	I	Stato successivo
OFF	0	1	OFF
OFF	1	1	ON
ON	0	1	ON
ON	1	1	ON
*	*	0	OFF

Proprietà:

$I = 0 \Rightarrow \delta(x, S, I) = OFF$

Lo stato OFF è assorbente rispetto a $I = 0$.

Caso errato (negazione software)

Stato	S	I	Stato successivo
OFF	1	1	OFF
OFF	1	0	ON

Si genera transizione pericolosa:

$(OFF, S=1, I=0) \rightarrow ON$

Il sistema non è monotono rispetto alla variabile di sicurezza.

Proprietà formale violata

Definiamo proprietà di sicurezza:

$I = 0 \Rightarrow X = OFF$

Nel sistema errato questa proprietà non è invariante.

Analisi con Doppio Canale – Categoria 3/4

Riferimento: EN ISO 13849-1

Architettura Categoria 3

Caratteristiche:

- Due canali indipendenti.
- Monitoraggio di coerenza.
- Un singolo guasto non porta alla perdita della funzione di sicurezza.

Definiamo:

- I_1 = canale A
- I_2 = canale B

Funzione di sicurezza:

$F_{safe} = I_1 \wedge I_2$

Auto-ritenuta:

$Q_{n+1} = (S \vee Qn) \wedge I_1 \wedge I_2$

Proprietà di tolleranza al guasto

Caso: rottura filo su canale A

$I_1=0$

Allora:

$Q_{n+1} = 0$

Sistema sicuro.

Caso con inversione errata software
$Q_{n+1} = (S \vee Qn) \wedge \neg I_1 \wedge \neg I_2$

Rottura filo:

$I_1=0 \Rightarrow \neg I_1=1$

Possibile consenso alla marcia.

Perdita di categoria.

Categoria 4

Richiede:

- Ridondanza
- Diagnostica continua
- Rilevazione accumulo guasti

Modello logico:

$F_{safe} = (I_1 \wedge I_2) \wedge D$

dove D = diagnostica.

La Regola del Gottardo è condizione necessaria per mantenere la semantica “1 = safe”.

Invertire nel software altera la mappatura semantica tra:

- livello elettrico,
- livello logico,
- livello di certificazione PL.

Conclusioni Ingegneristiche

1. La Regola del Gottardo garantisce formalmente:
    - reset dominante,
    - invarianza dello stato sicuro,
    - monotonia rispetto alla variabile di sicurezza.
2. In algebra booleana è dimostrabile tramite assiomi classici.
3. In teoria degli automi è proprietà di stato assorbente[9].
4. In sicurezza funzionale è requisito strutturale per mantenere PL e categoria SIL[10].

---

[9] Nel contesto della teoria degli automi, uno **stato assorbente** (in inglese *absorbing state*) è uno stato di un automa a stati finiti caratterizzato dalla proprietà che, una volta raggiunto, il sistema non può più evolvere verso stati differenti. Formalmente, dato un automa deterministico $A = (Q, \Sigma, \delta, q_0)$, uno stato $q_a \in Q$è detto assorbente se per ogni simbolo di ingresso $x \in \Sigma$vale $\delta(q_a, x) = q_a$. Tale proprietà implica che tutte le transizioni uscenti dallo stato conducono nuovamente allo stato stesso, rendendolo un punto di stabilità del sistema dinamico descritto dall'automa. In termini modellistici, gli stati assorbenti sono frequentemente utilizzati per rappresentare condizioni terminali, condizioni di errore non recuperabile oppure configurazioni di equilibrio dalle quali non è prevista alcuna ulteriore evoluzione del comportamento del sistema.

[10] Il **Performance Level (PL)** e il **Safety Integrity Level (SIL)** sono metriche normative utilizzate per quantificare il livello di affidabilità richiesto alle funzioni di sicurezza nei sistemi di controllo. Il Performance Level, definito nella norma ISO 13849-1, esprime la capacità di una funzione di sicurezza di ridurre il rischio attraverso cinque livelli discreti, indicati da PL a (più basso) a PL e (più elevato), determinati in base alla probabilità media di guasto pericoloso per ora e alla struttura architetturale del sistema. Il Safety Integrity Level, definito nella norma IEC 61508 e adottato in ambito industriale anche dalla IEC 62061, rappresenta analogamente la probabilità che una funzione di sicurezza esegua correttamente il proprio compito quando richiesta; esso è articolato in quattro livelli crescenti, SIL 1–SIL 4, associati a intervalli quantitativi di probabilità di guasto pericoloso. Entrambe le classificazioni costituiscono strumenti di valutazione del rischio e di progettazione dei sistemi di sicurezza, consentendo di specificare e verificare il grado di integrità richiesto alle funzioni di protezione.

La tesi di Gottardo
Fondamenti Logico-Formali della Regola del Gottardo nei Sistemi PLC e di Sicurezza

Una trattazione in algebra booleana, teoria degli automi, logica temporale e sicurezza funzionale

Abstract

La cosiddetta ***Regola del Gottardo***, formulata nel contesto della programmazione PLC con contatti fisici normalmente chiusi delegati allo sgancio di un'auto-ritenuta, può essere interpretata non soltanto come una regola pratica di officina, ma come una proprietà strutturale dimostrabile in algebra booleana, modellabile tramite automi a stati finiti e verificabile mediante logica temporale formale.

Nel presente lavoro si dimostra che tale regola garantisce la dominanza del reset, la preservazione dell'invariante di sicurezza e la coerenza semantica tra livello elettrico e livello logico, risultando condizione necessaria — benché non sufficiente — per il mantenimento dei requisiti di sicurezza previsti dalle norme EN ISO 13849-1 e IEC 62061.

Struttura logica della funzione di autoritenuta

Consideriamo un sistema di comando motore con pulsante di avviamento normalmente aperto e contatto di arresto fisico normalmente chiuso cablato in serie, letto da un PLC conforme alla IEC 61131-3.

Indichiamo con:

- S il comando di start,
- F lo stato fisico del contatto NC,
- I il valore logico letto dal PLC,
- Qn lo stato della bobina al ciclo n.

Poiché il contatto è NC cablato in campo, la variabile letta coincide con lo stato elettrico reale:

$$I = F$$

con la convenzione:

$F=1 \Rightarrow$ circuito chiuso (condizione normale)
$F=0 \Rightarrow$ circuito aperto (sgancio)

La funzione di autoritenuta corretta è definita da:

$$Q_{n+1} = (S \vee Q_n) \wedge I$$

Tale espressione incorpora la struttura classica di latch con reset dominante implicito nella congiunzione finale.

Dimostrazione in algebra booleana assiomatica

L'algebra booleana classica, fondata sugli assiomi di Huntington, fornisce gli strumenti per una dimostrazione rigorosa.

Si vuole dimostrare che la funzione precedente possiede la proprietà di dominanza dello sgancio.

Sia $I = 0$. Allora:

$$Q_{n+1} = (S \vee Q_n) \wedge 0$$

Per l'assioma dell'elemento nullo:

$$A \wedge 0 = 0$$

segue immediatamente:

$$Q_{n+1} = 0$$

La variabile di uscita è indipendente da S e Qn.
Il reset è dominante.

Si analizzi ora la funzione ottenuta negando il contatto nel software:

$$Q_{n+1} = (S \vee Q_n) \wedge \neg I$$

Ponendo I=0 si ottiene:

$$Q_{n+1} = (S \vee Q_{n)} \wedge 1 = S \vee Q_n$$

Il reset perde dominanza.
La proprietà di sicurezza non è più una conseguenza assiomatica della struttura.

Ne consegue che la Regola del Gottardo non è una convenzione grafica, ma una proprietà strutturale della funzione booleana.

Modellazione come sistema a stati finiti

Si definisce l'automa deterministico:

$M = (X, U, \delta)$

dove lo spazio degli stati è X= {OFF, ON}, l'insieme degli ingressi è U= {S, I} e la funzione di transizione è:

$\delta(x, S, I) =$

- OFF se I=0
- ON se I=1 ∧ (S=1 ∨ x=ON)
- OFF altrimenti

La proprietà fondamentale del sistema è che lo stato OFF è assorbente rispetto alla condizione I=0. Formalmente:

$$I = 0 \Rightarrow \delta(x, S, I) = OFF \ \forall x, S$$

Se si introduce la negazione impropria del contatto, l'automa ammette la transizione:

$$(OFF, S=1, I=0) \rightarrow ON$$

La condizione di sicurezza non è più invariante.
Il sistema perde monotonia rispetto alla variabile di sicurezza.

Verifica mediante logica temporale

La proprietà desiderata può essere espressa in Logica Temporale Lineare (LTL).

Sia definita la proposizione atomica:

$\text{Safe}: = (Q = \text{OFF})$

La proprietà fondamentale diventa:

$G(I = 0 \rightarrow \text{Safe})$

dove G è l'operatore "globally".

Nel modello corretto, la formula è valida in tutti i cammini temporali.

Nel modello errato, esiste un cammino tale che:

$$I = 0 \wedge S = 1 \wedge Qn = \text{OFF}$$

Ma:

$$Q_{n+1} = \text{ON}$$

La formula LTL è falsificata.
La regola può quindi essere vista come condizione necessaria alla validità dell'invariante di sicurezza in logica temporale.

In termini CTL:

$$AG(I = 0 \rightarrow AX(Q = \text{OFF}))$$

risulta vera solo nella formulazione corretta.

Estensione a sistemi a doppio canale Categoria 3/4

Nel quadro normativo della EN ISO 13849-1, la Categoria 3 richiede ridondanza con rilevazione dei guasti.

Si introducono due canali indipendenti I_1 e I_2, entrambi NC fisici.

La funzione diventa:

$$Q_{n+1}= (S \vee Q_n) \wedge I_1 \wedge I_2$$

Si supponga un guasto singolo con rottura filo sul primo canale:

$$I_1=0$$

Per proprietà assiomatica:

$$Q_{n+1}=0$$

La funzione di sicurezza è preservata.

Se invece si negano i segnali nel software:

$$Q_{n+1}= (S \vee Q_n) \wedge \neg I_1 \wedge \neg I_2$$

La rottura filo produce:

$$I_1= 0 \Rightarrow \neg I_1 = 1$$

Il guasto non è più intrinsecamente sicuro.
La struttura perde la proprietà fail-safe e può compromettere il Performance Level.

In Categoria 4, dove è richiesta anche la rilevazione dell'accumulo di guasti, la coerenza semantica "1 = condizione sicura" è fondamentale per l'analisi probabilistica e la corretta implementazione dei meccanismi diagnostici.

Analisi probabilistica e relazione con SIL

Nel contesto della IEC 62061, la probabilità di guasto pericoloso dipende dalla capacità del sistema di portarsi automaticamente in stato sicuro in caso di fault.

Con contatti NC e funzione AND finale, la probabilità di guasto pericoloso è funzione della probabilità di guasto simultaneo dei canali.

Negando nel software, la rottura singola non genera arresto certo.
La probabilità di guasto pericoloso aumenta, modificando il PFHd e quindi il SIL conseguibile.

La Regola del Gottardo diventa dunque condizione strutturale per mantenere basso il tasso di guasto pericoloso.

Conclusioni generali

La Regola del Gottardo, lungi dall'essere una convenzione grafica o una semplice abitudine didattica, può essere interpretata come:

una proprietà algebrica di dominanza del reset,
una proprietà strutturale di un automa deterministico,
un invariante esprimibile in logica temporale,
una condizione necessaria per il mantenimento del Performance Level nei sistemi ridondanti.

La sua validità è rigorosa entro l'ipotesi di contatto fisico NC cablato in campo.
Al di fuori di tale ipotesi, la regola permane in validità, sulla base della dimostrazione contronominale, e deve essere riconsiderata alla luce della semantica del segnale.