

Progetto Logico Sequenziale

Reti Logiche

Programma delle prossime lezioni

Circuiti sequenziali



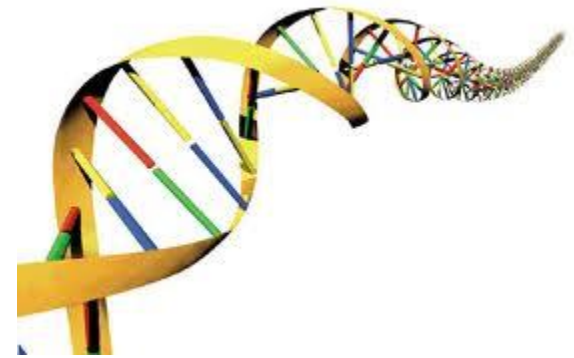
```
graph TD; A[Circuiti sequenziali] --> B[Elementi di memoria (cenni)]; B --> C[Macchine a stati finiti]; C --> D[Circuiti sequenziali fondamentali]; D --> E[Metodo di progetto RTL];
```

Elementi di memoria (cenni)

Macchine a stati finiti

Circuiti sequenziali fondamentali

Metodo di progetto RTL

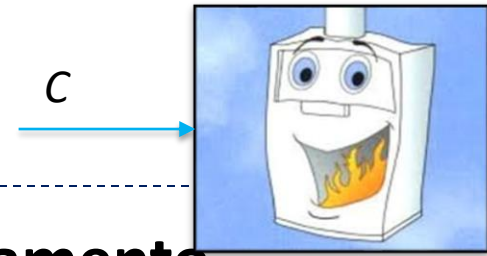


Circuiti sequenziali

Perché

- ▶ **Fin ad ora abbiamo realizzato funzioni Booleane**
 - ▶ $f(x_1, \dots, x_n) : B^n \rightarrow B$
 - ▶ Associazione tra combinazioni delle variabili di ingresso e valore di uscita
 - ▶ Espressioni e porte logiche usate per fare un circuito
- ▶ **Le funzioni Booleane sono combinatorie**
 - ▶ Significa che ad ogni combinazione di ingresso è associata sempre la stessa uscita
 - ▶ Se una combinazione di ingresso si presenta due volte in istanti differenti, l'uscita sarà la stessa in entrambi gli istanti (a parte il transitorio)
 - ▶ Il circuito guarda solo la combinazione di ingresso presente per calcolare l'uscita
- ▶ **Questo modello è insufficiente in certi frangenti**

Controllo di una caldaia



- ▶ **Si vuole controllare una caldaia per riscaldamento**
 - ▶ La caldaia dispone di un ingresso digitale C
 - ▶ Quando a 1 si accende
 - ▶ Quando a 0 si spegne
- ▶ **Vogliamo controllare la temperatura dell'ambiente**
 - ▶ Accendiamo la caldaia quando si scende sotto una soglia di temperatura, e la spegnamo quando si sale sopra una seconda soglia
 - ▶ Usiamo due soglie per evitare di accendere e spegnere in continuazione
- ▶ **Un sensore ci fornisce la temperatura**
 - ▶ Codifica per esempio in binario

Andamento e struttura

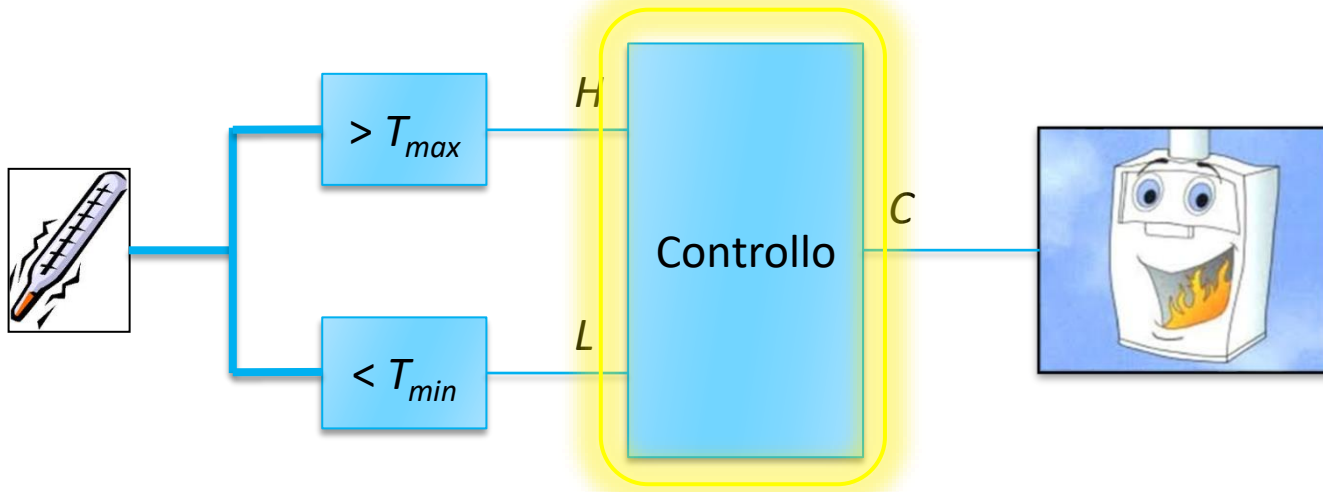
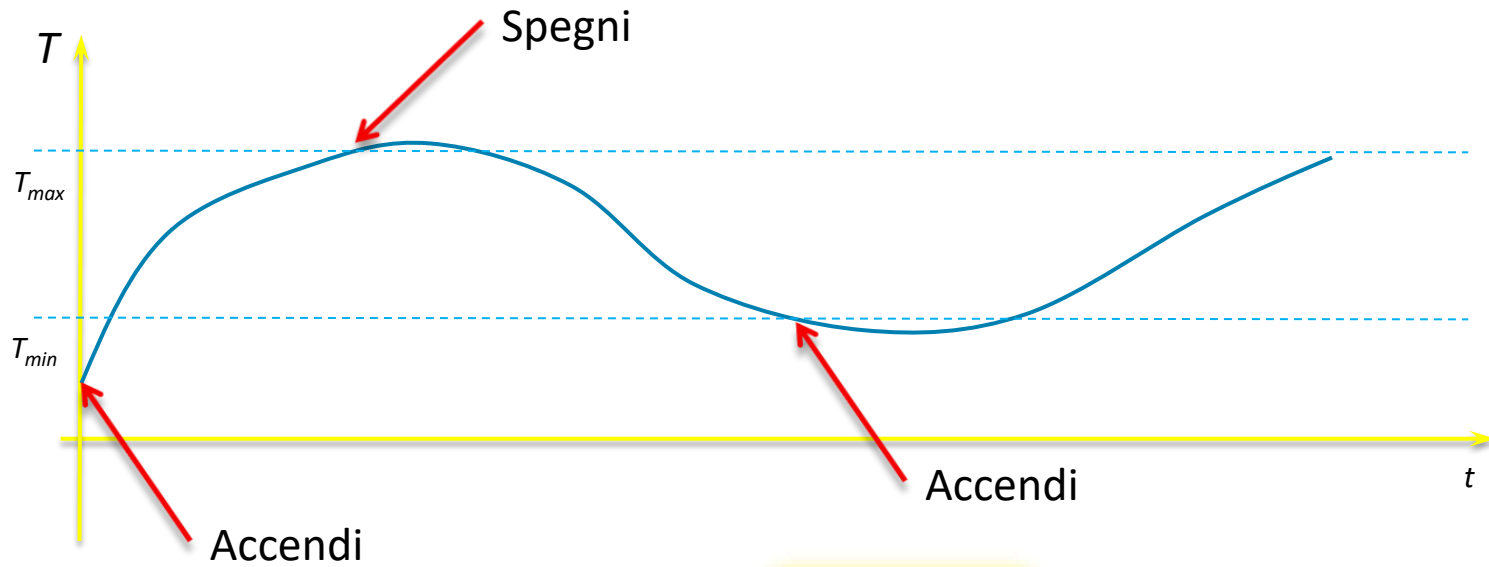


Tabella della verità

► Condizioni

- $H = 1$ quando $T > T_{max}$
- $L = 1$ quando $T < T_{min}$

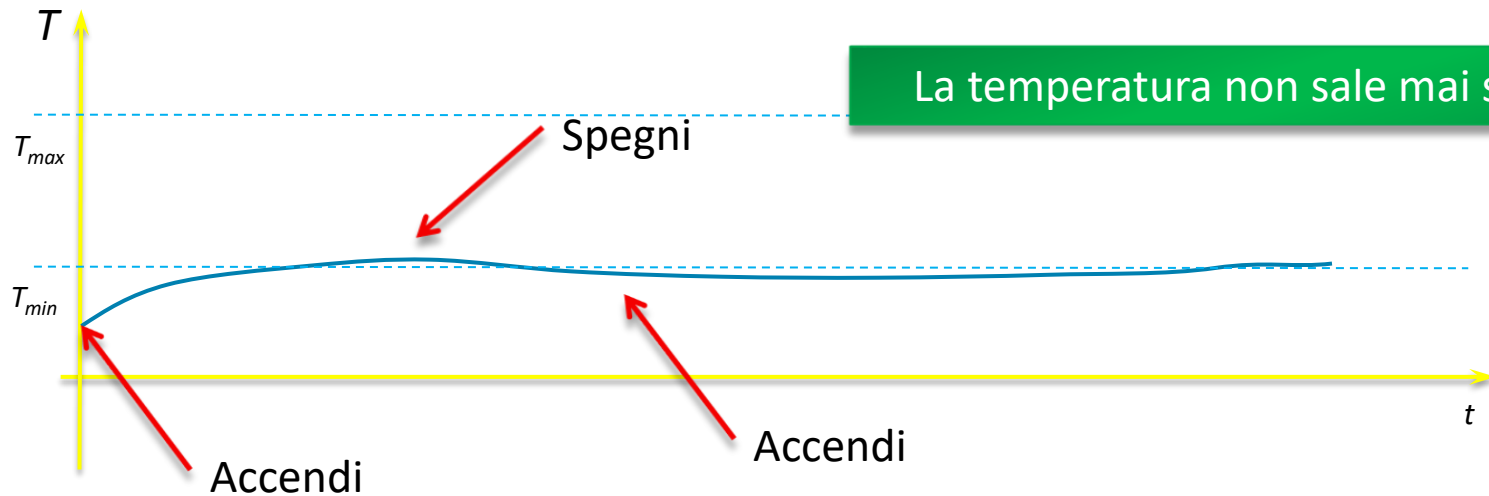
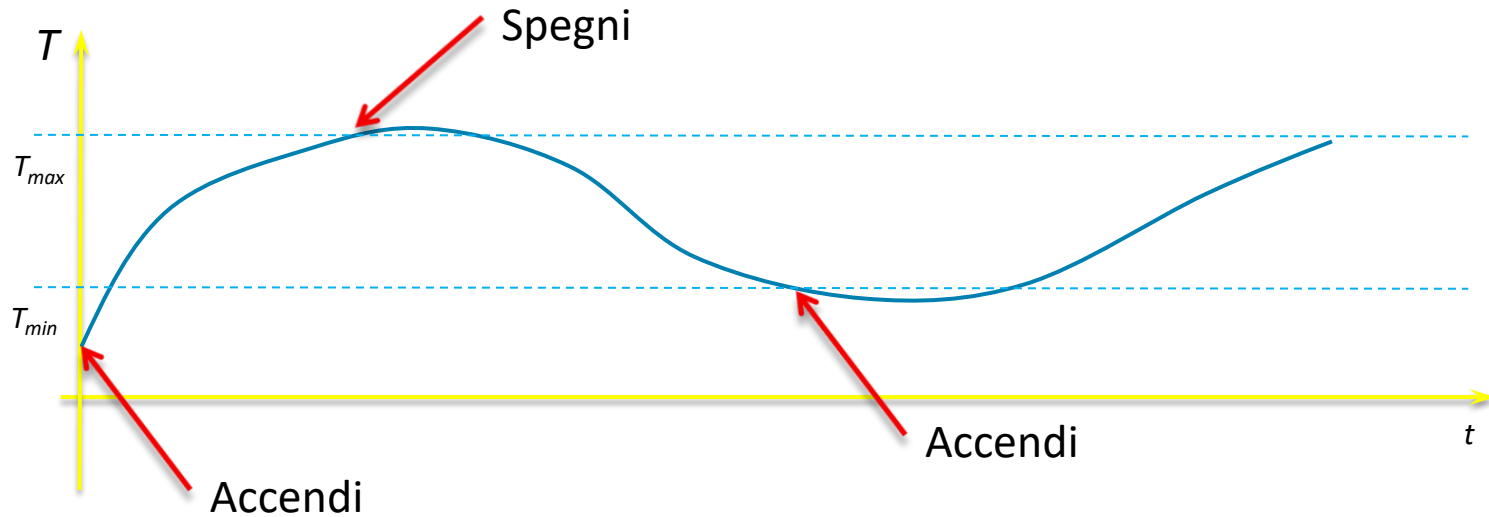
► Possiamo costruire la tabella della verità

- Quando $H = 1$ devo spegnere perché fa caldo
- Quando $L = 1$ devo accendere perché fa freddo
- Notare che H ed L non sono mai contemporaneamente a 1, o la temperatura dovrebbe avere due valori!
- Quindi posso mettere un don't care per $H = 1$ ed $L = 1$

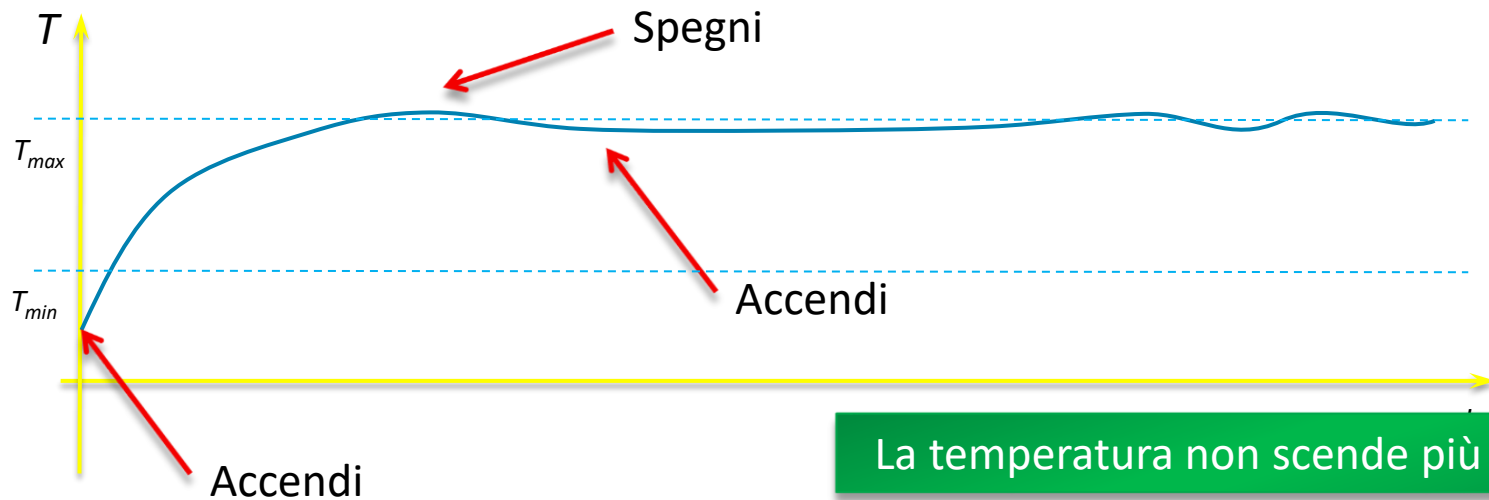
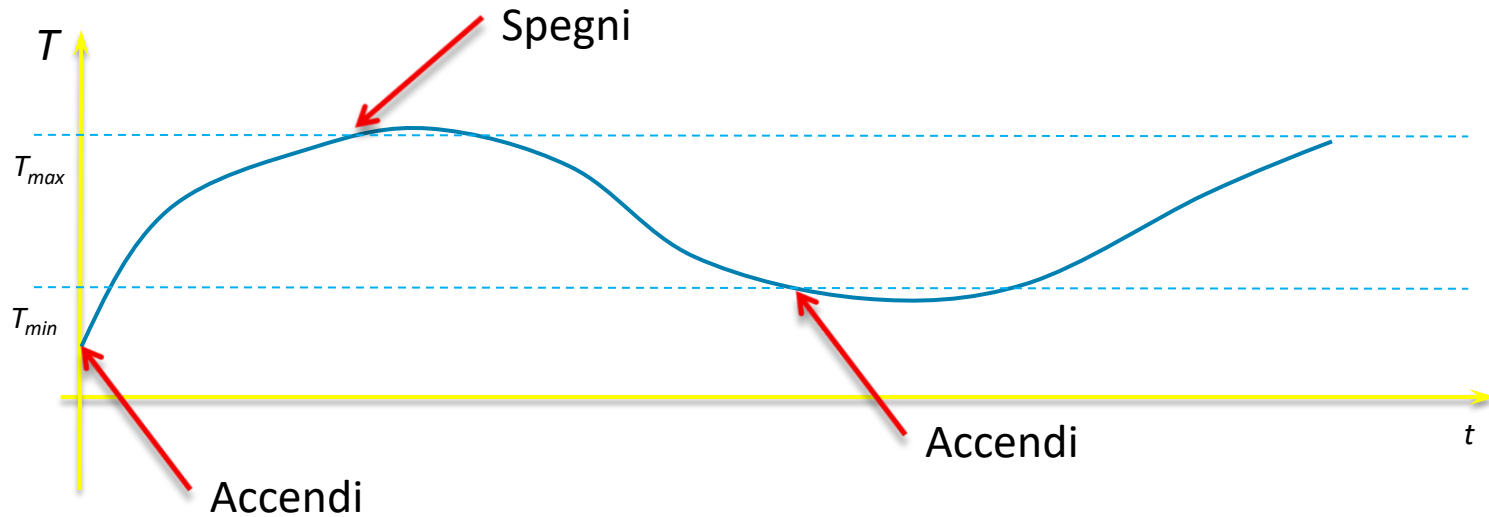
► Ma cosa metto per $H = 0$ ed $L = 0$?

H	L	C
0	0	
0	1	1
1	0	0
1	1	-

Spenta: $C = 0$



Accesa: $C = 1$



La temperatura non scende più sotto T_{max}

Tabella della verità

► Condizioni

- $H = 1$ quando $T > T_{max}$
- $L = 1$ quando $T < T_{min}$

► Possiamo costruire la tabella della verità

- Quando $H = 1$ devo spegnere perché fa caldo
- Quando $L = 1$ devo accendere perché fa freddo
- Notare che H ed L non sono mai contemporaneamente a 1, o la temperatura dovrebbe avere due valori!
- Quindi posso mettere un don't care per $H = 1$ ed $L = 1$

► Ma cosa metto per $H = 0$ ed $L = 0$?

- Spenta: però se fa freddo la temperatura non sale mai sopra T_{min} perché spengo la caldaia
- Accesa: analogamente non si scende mai sotto T_{max}

► Occorrono due valori diversi

- Accesa se stavo già scaldando
- Spenta se non stavo scaldando

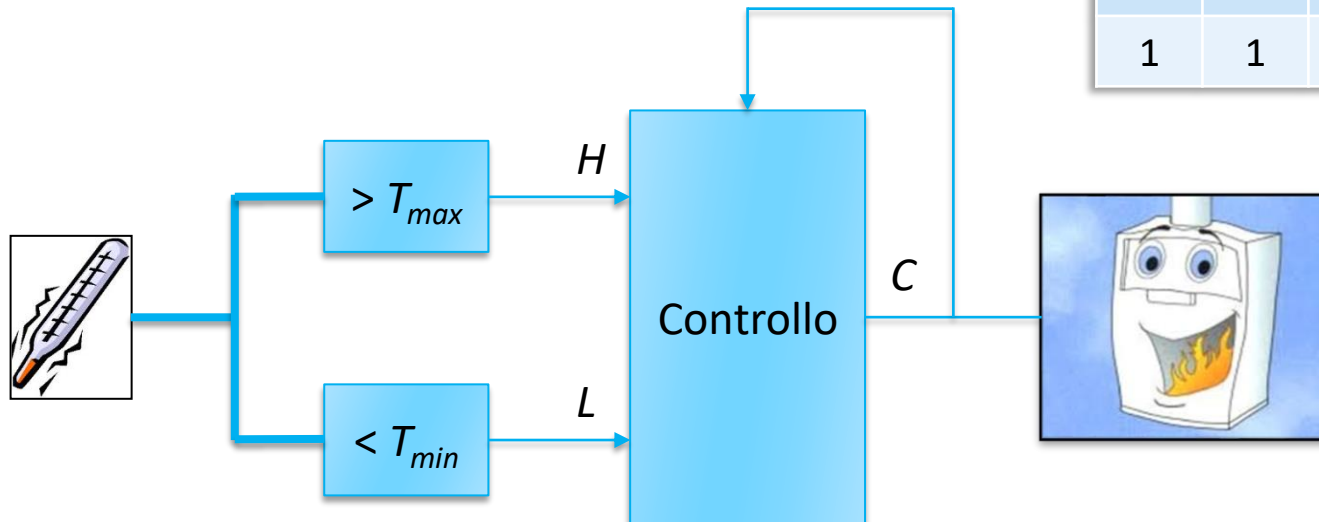
H	L	C
0	0	
0	1	1
1	0	0
1	1	-

Poiché si devono assegnare due valori diversi per lo stesso input, la funzione NON è combinatoria

Come fare?

- **Occorre guardare ciò che si stava facendo**
 - Se la caldaia era accesa, deve rimanere accesa
 - Se era spenta deve rimanere spenta
 - Devo riportare l'uscita come ingresso del controllore
 - In questo modo il circuito torna ad essere (quasi) combinatorio

<i>H</i>	<i>L</i>	<i>C</i>	<i>C</i>
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	-
1	1	1	-



Mappa e circuito

► Equazione

- $C = L + CH'$
- Notate che C dipende da C
- La caldaia deve accendersi o se la temperatura è troppo bassa (L), oppure se era accesa (C) e la temperatura non è troppo alta (H')

► Cosa è cambiato nel circuito?

- C'è un ciclo (anello)
- Il circuito ha ora un feedback
- Bisogna fare attenzione alla stabilità
 - L'equazione potrebbe non avere soluzioni, oppure averne tante
- Il feedback porta nel circuito la capacità di memoria

$C \backslash HL$	00	01	11	10
0	0	1	-	0
1	1	1	-	0

$$C = L + CH'$$

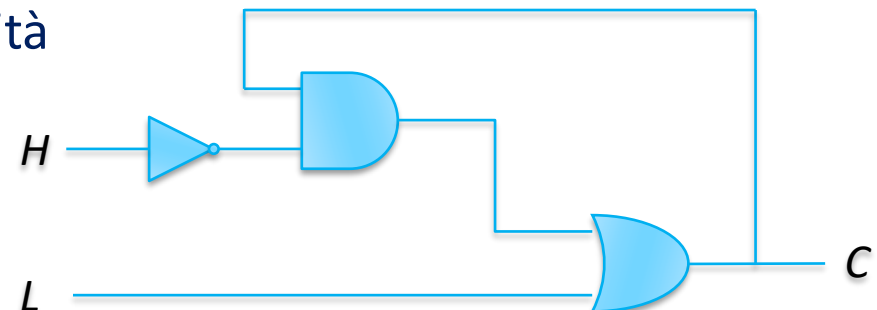
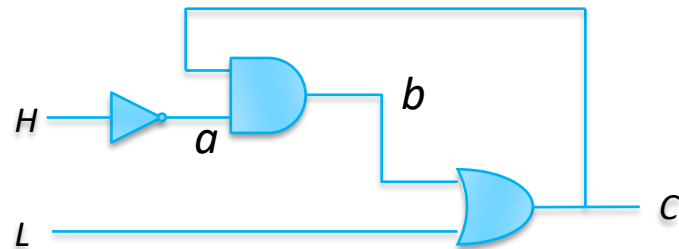
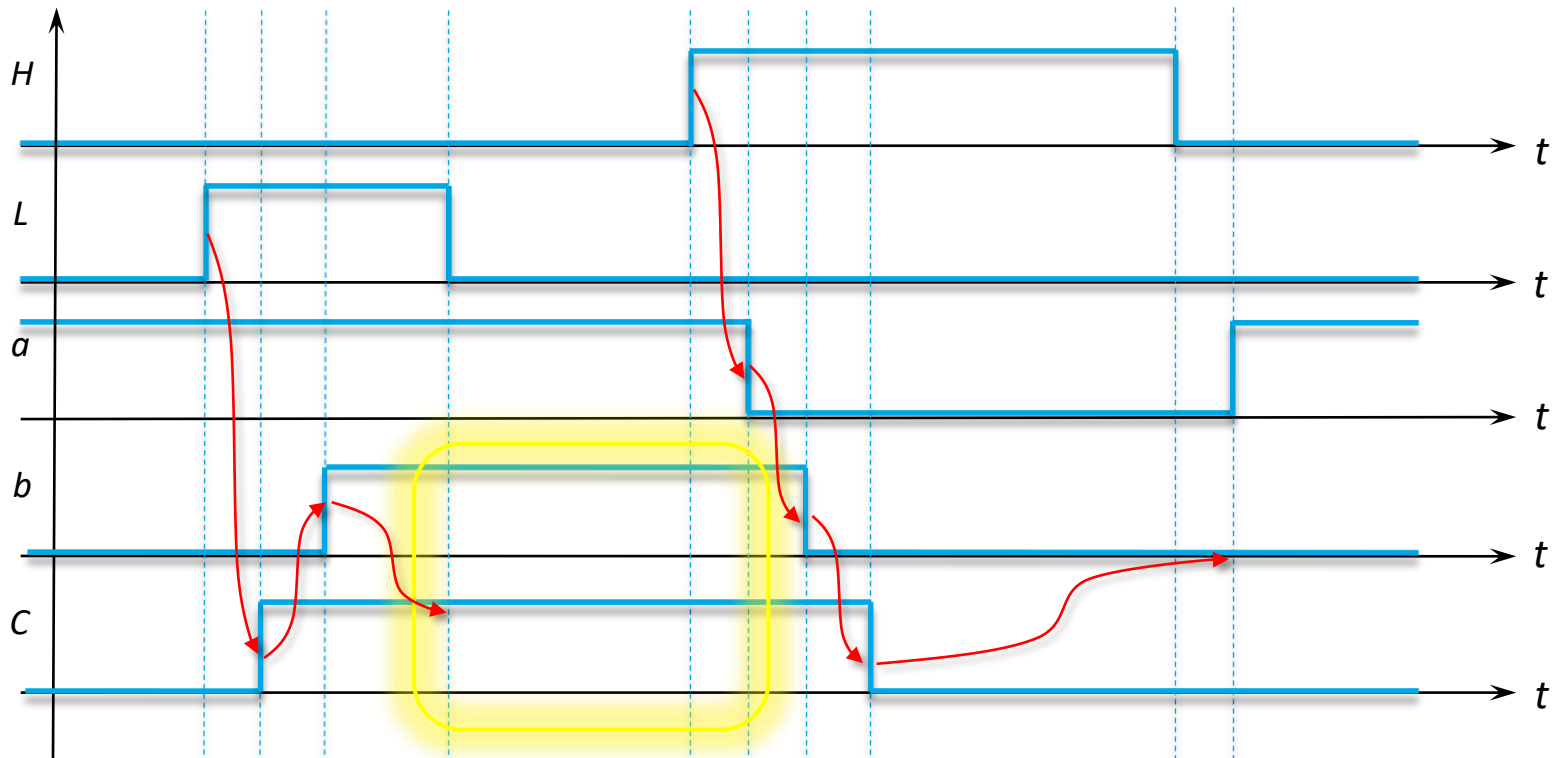


Diagramma temporale



- Il ciclo tra C e b si autosostiene
- L'uscita mantiene il valore anche quando l'ingresso sparisce

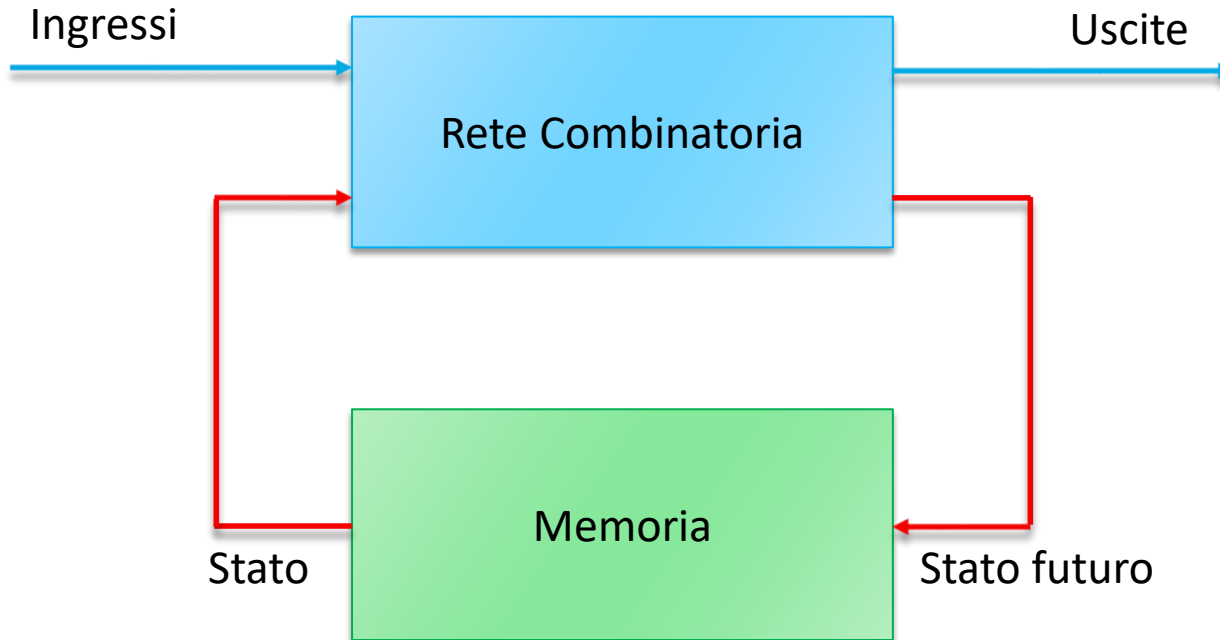


- ▶ **Le funzioni Booleane sono combinatorie**
 - ▶ Ad ogni combinazione di ingressi corrisponde sempre la stessa uscita
 - ▶ Vogliamo però poter fare dei circuiti che non soddisfano questa proprietà
 - ▶ Per farlo, facciamo in modo che l'uscita dipenda sia dall'ingresso corrente, sia da quelli passati (che, per esempio, hanno determinato il valore dell'uscita corrente)
- ▶ **Le funzioni che dipendono dal passato si chiamano sequenziali**
 - ▶ Il valore dell'uscita dipende *dalla sequenza* dei valori di ingresso, e non solo dall'ingresso corrente
 - ▶ Per uno stesso ingresso, l'uscita può essere differente a seconda della storia passata
- ▶ **Per realizzare funzioni sequenziali abbiamo bisogno di una memoria**
 - ▶ La si realizza riportando l'uscita o un nodo interno all'ingresso
 - ▶ Si crea un feedback che mantiene la storia dei valori

Circuiti sequenziali

- ▶ **I circuiti sequenziali dispongono di memoria**
 - ▶ La si può realizzare ad-hoc come nell'esempio visto
 - ▶ Le equazioni possono però risultare molto complesse
 - ▶ Le mappe anche diventano complesse
- ▶ **Conveniente separare la memoria come elemento a parte**
 - ▶ Conteniamo gli anelli di feedback in circuiti di cui conosciamo bene il comportamento
 - ▶ Evitiamo condizioni di instabilità
 - ▶ La restante parte è combinatoria e la sappiamo fare bene

Circuiti sequenziali



- ▶ **I valori contenuti nella memoria codificano lo stato del circuito**
 - ▶ Lo stato concorre insieme agli ingressi a definire i valori delle uscite
 - ▶ La parte combinatoria calcola di volta in volta il nuovo valore dello stato

Sincroni ed asincroni

- ▶ **I circuiti sequenziali si dividono in due categorie**
 - ▶ Si differenziano da come e quando può cambiare lo stato
- ▶ **Circuiti asincroni**
 - ▶ Lo stato può cambiare in ogni istante a causa di un cambiamento dello stato futuro
 - ▶ Questo, a sua volta, può variare a causa di variazioni sugli ingressi o sullo stato presente
 - ▶ Il controllo della caldaia che abbiamo realizzato è asincrono
- ▶ **Circuiti sincroni**
 - ▶ Lo stato viene aggiornato solo in corrispondenza di eventi su uno speciale segnale di sincronizzazione detto *clock*
 - ▶ Facilita il progetto perché non vi è più dipendenza reciproca immediata tra stato presente e stato futuro
 - ▶ **Ci consente di disaccoppiare la lettura e il calcolo dello stato**