

I flip flop - 1

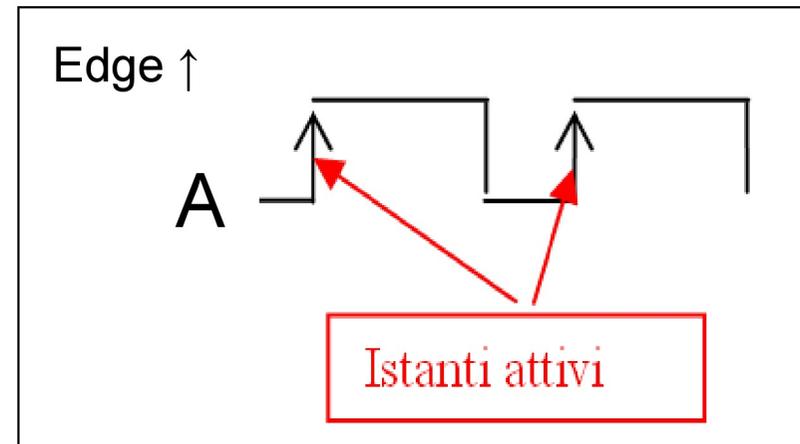
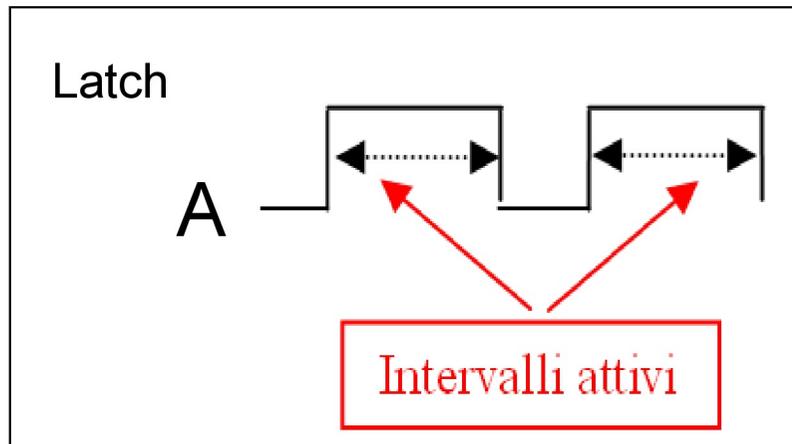
Generalità

- Elementi fondamentali (semplici reti sequenziali)
 - per la memorizzazione
 - per la costituzione di registri
 - per la costruzione di reti sequenziali
 - Memorizzano un bit, presentando in uscita (Q) 2 stati stabili:
 - stato di *SET*, o alto, o stato “1”: $Q=1$;
 - stato di *RESET*, o basso, o stato “0”: $Q=0$
 - Hanno in ingresso diversi segnali da cui traggono il nome, per esempio:
 - flip-flop RS: ingressi R (reset) e S (set)
 - flip-flop D: ingresso D (dato)
 - flip-flop T: ingresso T (trigger)
 - flip-flop JK: ingressi J e K
-

I flip flop - 2

Tempificazione

- Flip-flop abilitato (o sincronizzato): possiede un segnale di ingresso (A) che ne abilita il funzionamento
- Un flip-flop abilitato può essere:
 - **Latch**: cattura gli ingressi in tutto il periodo in cui A è attivo (es., $A=1$)
 - **Edge triggered**: cattura gli ingressi in corrispondenza di una variazione di A (fronte di salita: $0 \rightarrow 1$, o di discesa $1 \rightarrow 0$)



I flip flop - 3

Struttura

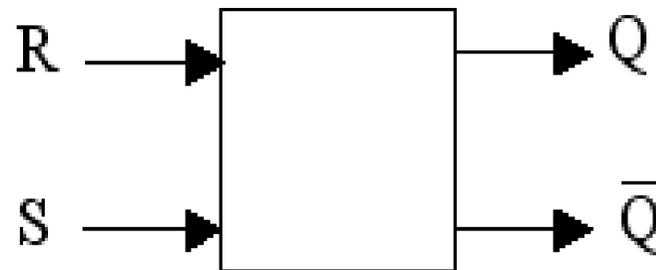
- Un Flip-flop è caratterizzato da:
 - **Struttura interna**, che può essere di una rete asincrona (vedi esempio di RS) o sincrona (vedi in seguito).
 - Una **equazione di stato**, che (al di là della struttura) indica l'uscita seguente in funzione dell'uscita precedente e degli ingressi.
-

I flip flop - 4

- Flip-flop
 - a memorizzazione dell'ingresso (RS, D)
 - a commutazione (T, JK)
 - Misti
-

Flip-flop RS

- Memorizza un bit di informazione
- Ingressi impulsivi
 - **Vincolo $RS=0$** (mai alti contemporaneamente)
 - Impulso su S (set) $\rightarrow Q=1, !Q=0$
 - Impulso su R (reset) $\rightarrow Q=0, !Q=1$
 - Per $R=S=0$ mantiene l'uscita precedente
- Escluse transizioni $RS = \dots \rightarrow 00 \rightarrow 01 \rightarrow 10 \rightarrow 00 \dots$ (a lee multiple)



Flip flop RS fondamentale

Descrizione comportamento

SR					uscita
stati	00	01	11	10	Q
Q ₀	Q ₀	Q ₀	--	Q ₁	0
Q ₁	Q ₁	Q ₀	--	Q ₁	1

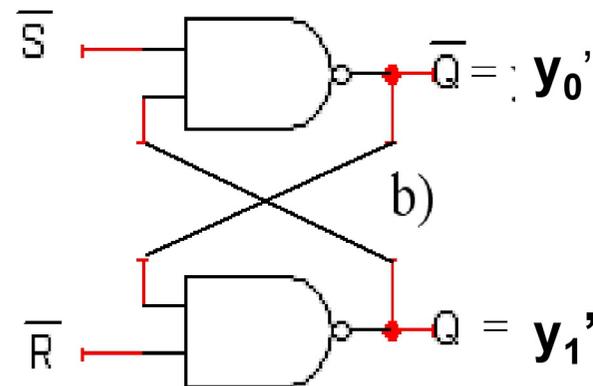
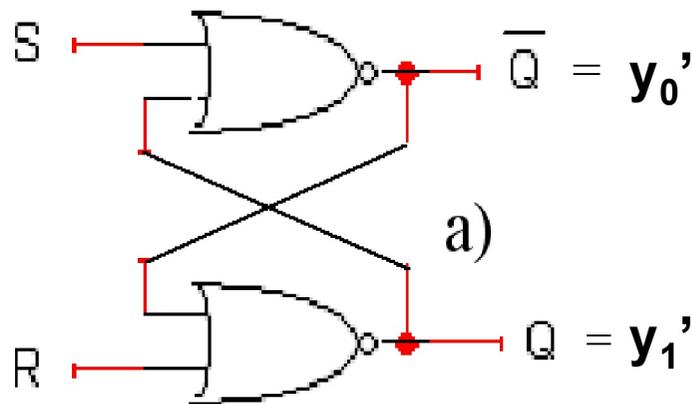
Dati tre ingressi R,S e Q_p (stato precedente)

SR				
Q _p	00	01	11	10
0	0	0	-	1
1	1	0	-	1

Equazione di stato

$$Q = S + Q_p \bar{R}$$

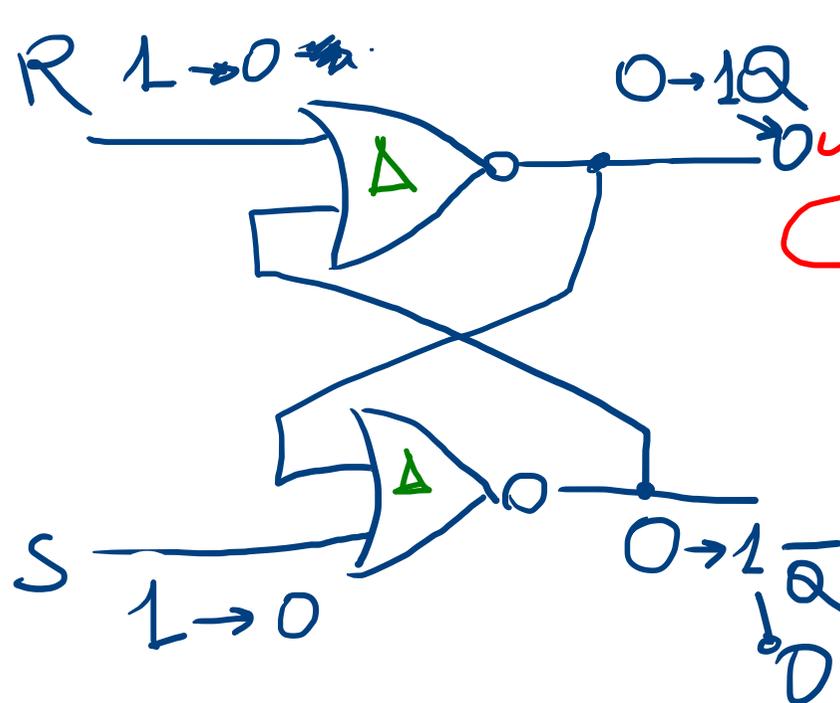
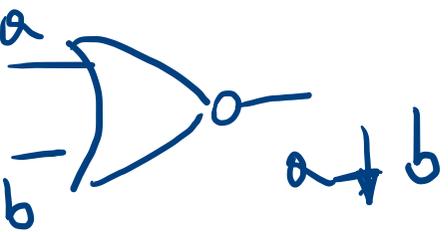
Flip-flop RS: analisi del funzionamento



La fig. mostra la rete che realizza il flip-flop fondamentale RS in logica NOR (a) e NAND (b). Si noti preliminarmente che la rete è realizzata con 2 variabili di stato invece dell'unica che sarebbe possibile realizzando direttamente la equazione di stato. La realizzazione del flip flop in logica NOR (NAND) può facilmente ottenersi progettandolo come rete sequenziale asincrona in cui il passaggio dallo stato stabile 01 (10) allo stato stabile 10 (01) avviene attraverso lo stato instabile 00 (11).

BISTABILE
RS
FONDAMENTALE

a	b	a ↓ b
0	0	1
0	1	0
1	0	0
1	1	0



RESET

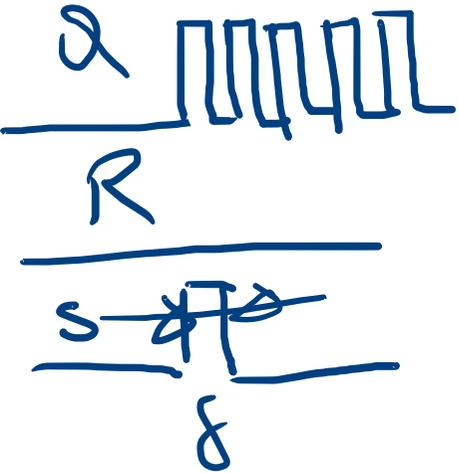
SET

RS	00	01	11	10	Q
10	10	00	00	10	0
00	11	01	00	10	0
01	01	01	00	00	1
11	00	00	00	00	1

Annotations: A green arrow points to the '00' state in the first row. A red arrow points to the '10' state in the first row. A red arrow points to the '01' state in the second row. A red arrow points to the '01' state in the third row. A red arrow points to the '00' state in the fourth row. A green arrow points to the '11' state in the second row. A green arrow points to the '01' state in the third row. A green arrow points to the '00' state in the fourth row. The text 'VIETATA' is written in red above the '11' column. The text 'RS = 0' is written in red at the bottom right.

RS = 0

Flip-flop RS: stati stabili e transizioni



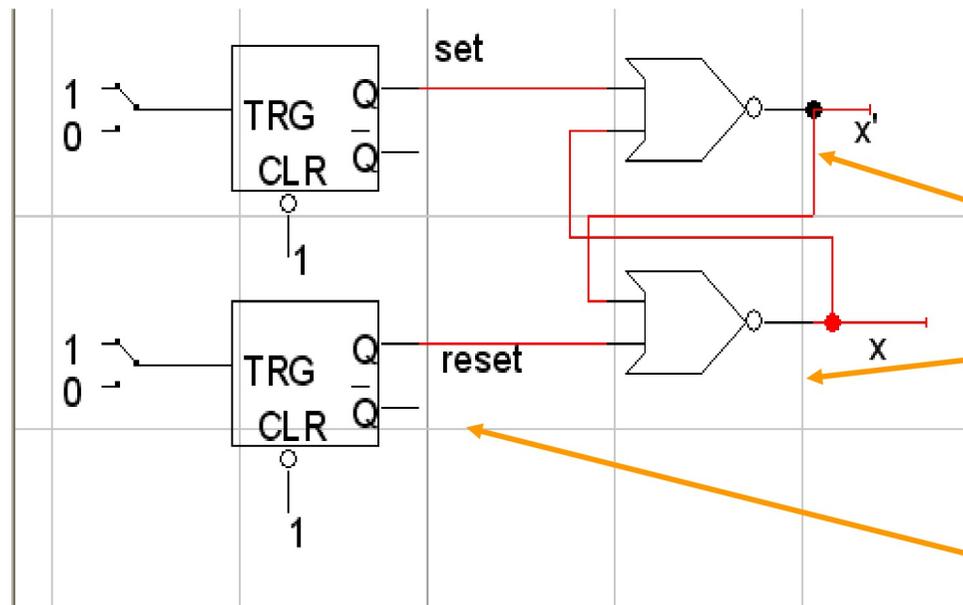
RS \downarrow

$y_0' y_1'$	$\bar{Q} Q$	00	01	11	10
10	10	00	00	00	10
00	11	01	00	00	10
01	01	01	00	00	00
11	00	00	00	00	00

The table shows the next state of the RS flip-flop based on the current state and inputs. Red circles highlight stable states (10, 01, 00) and blue arrows indicate transitions between states.

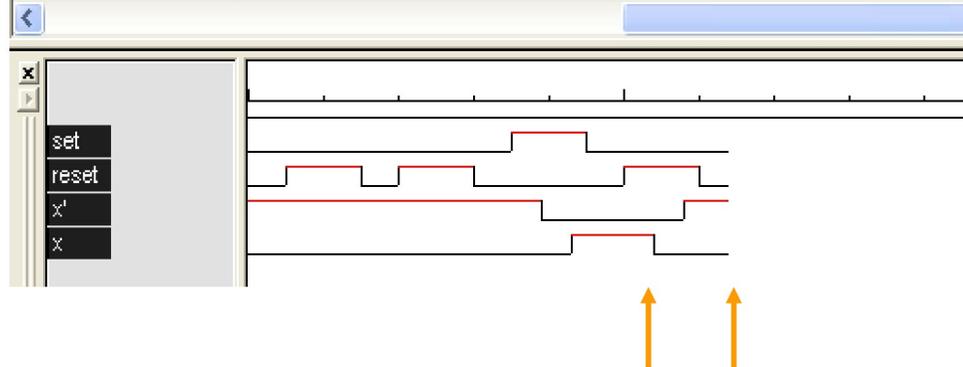
Se $\delta > 2\Delta \rightarrow$ NO OSCILLAZIONE

Flip-flop RS: tempificare durata dell'input



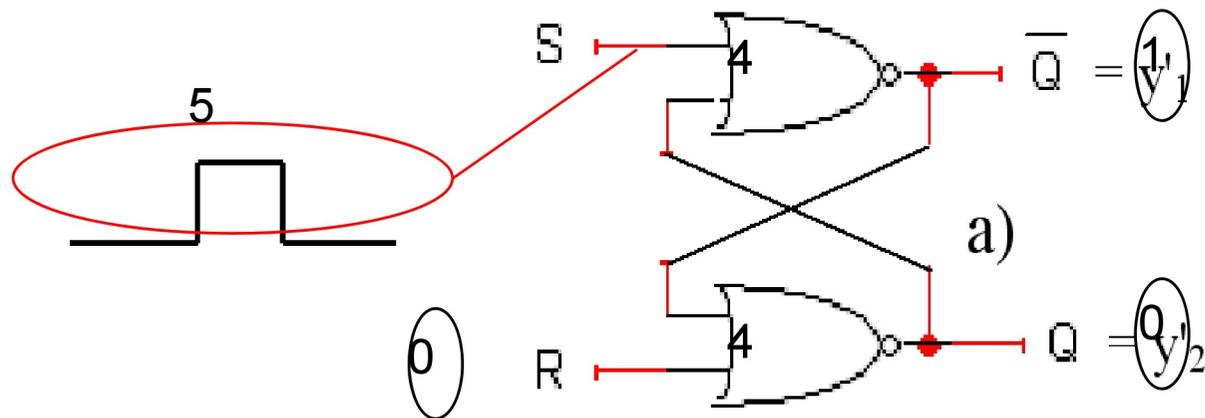
Le NAND hanno un ritardo di 4 unità di tempo

Gli impulsi in ingresso hanno durata **10** unità di tempo



Che succede per tempificazione errata?

- o Si parte da stato **10** sotto ingressi **RS=00**
- o Ritardo di porte = 4 unità di tempo τ
- o Input $d=5\tau$ (errore di progetto)

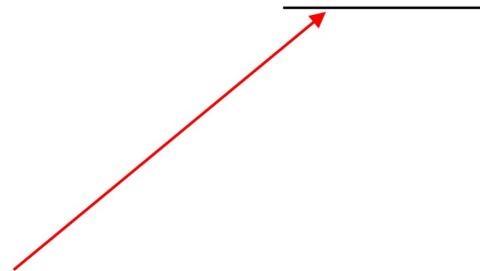


Tra 5τ e 8τ lo stato è **00**, sotto l'ingresso **RS=00**

La macchina comincia ad oscillare tra gli stati **00** e **11** sotto l'ingresso **RS=00**

Flip-flop RS: tempificare durata dell'input

RS = ...00 → 01 → 00...
 Stato = ...10 → 00 → .. → 11 → 00 → 11 →



La macchina non ha il tempo di spostarsi sul nuovo stato stabile

$y_0' y_1'$

RS

	00	01	11	10
10	10	00	00	10
00	11	01	00	10
01	01	01	00	00
11	00	00	00	00



Tempificazione nel caricamento dei registri: latch

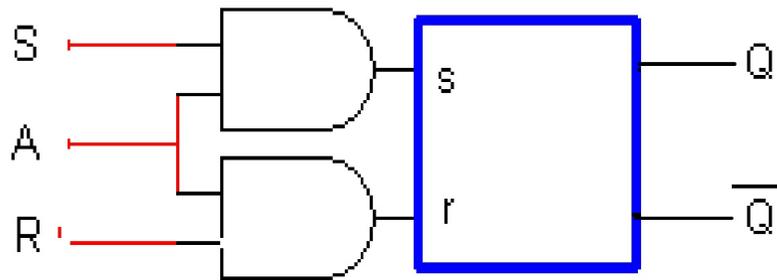
Con riferimento ai registri a sincronizzazione esterna, il modello fondamentale è anche detto di *registro latch*,

Un registro di tipo latch cattura l'ingresso-dati per tutto l'intervallo δ in cui è attivo il segnale di abilitazione s

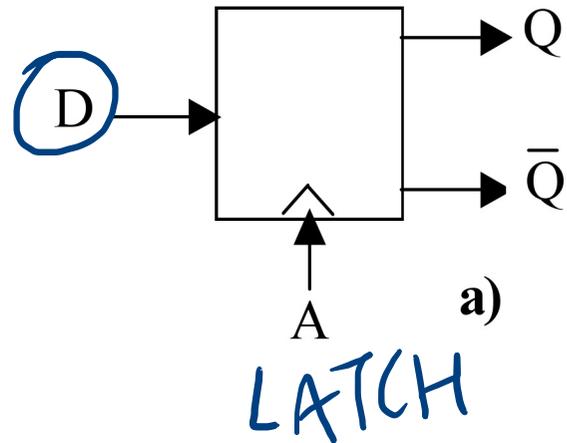
Il registro fondamentale a sincronizzazione esterna può infatti funzionare anche nell'ipotesi di sequenza di ingresso non impulsiva: una variazione di stato può infatti avvenire anche mentre è $s=1$, in concomitanza con una variazione di D . Ad esempio (vedi tabella 5.3b), con la sequenza $n_1x_1x_2$ il registro raggiungerebbe l'uscita u_2 . Il segnale di abilitazione agisce, in tal caso, come *livello abilitante* nel senso che, durante tutto il tempo che è attivo, ogni variazione sugli ingressi D viene "catturata" e trasferita nel registro.

Un RS abilitato con tempificazione latch

Possibile realizzazione di un flip-flop RS abilitato con tempificazione di tipo latch a partire da un RS asincrono



Flip flop D

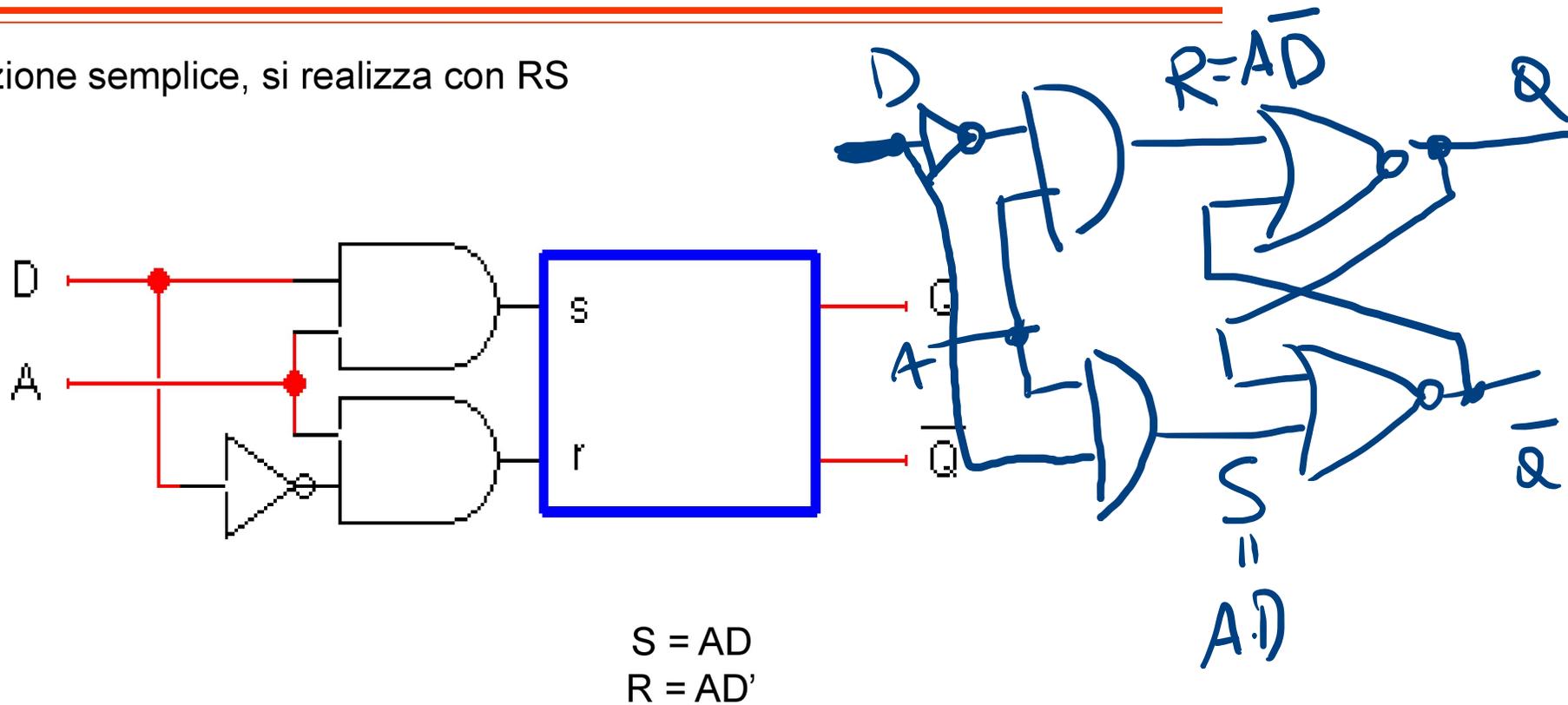


Equazione di stato:

$$Q = AD + \bar{A}Q_p$$

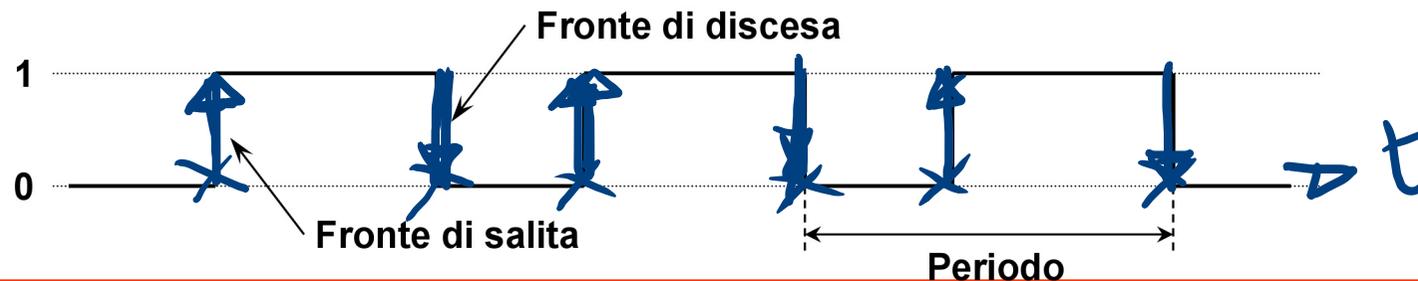
Flip-flop D: realizzazione come latch

Di concezione semplice, si realizza con RS



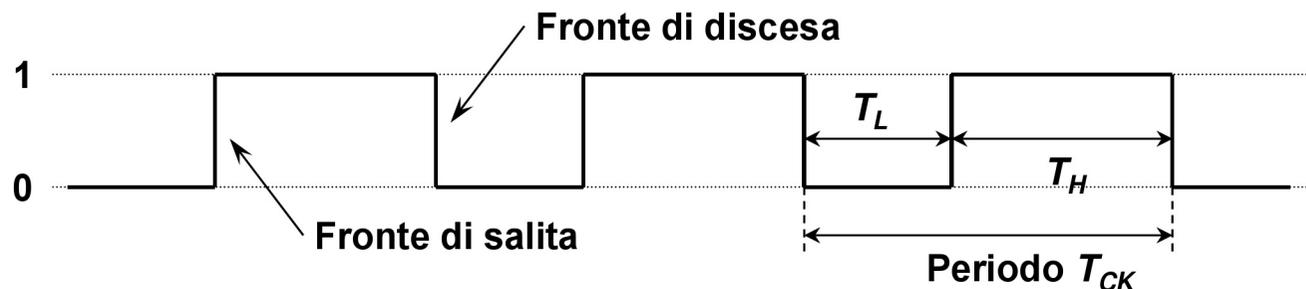
Il segnale di clock

- Il valore del segnale elettrico viene letto o *campionato* in istanti determinati
- Gli istanti in cui deve essere *campionato* il segnale elettrico sono scanditi da un apposito segnale detto *clock*
- Un *clock* ha le seguenti caratteristiche:
 - E' un segnale binario
 - E' un segnale periodico
- Spesso nei sistemi digitali, tutti i componenti con memoria (flip-flop e memorie) “leggono” gli ingressi nello stesso istante, scandito dal segnale di clock

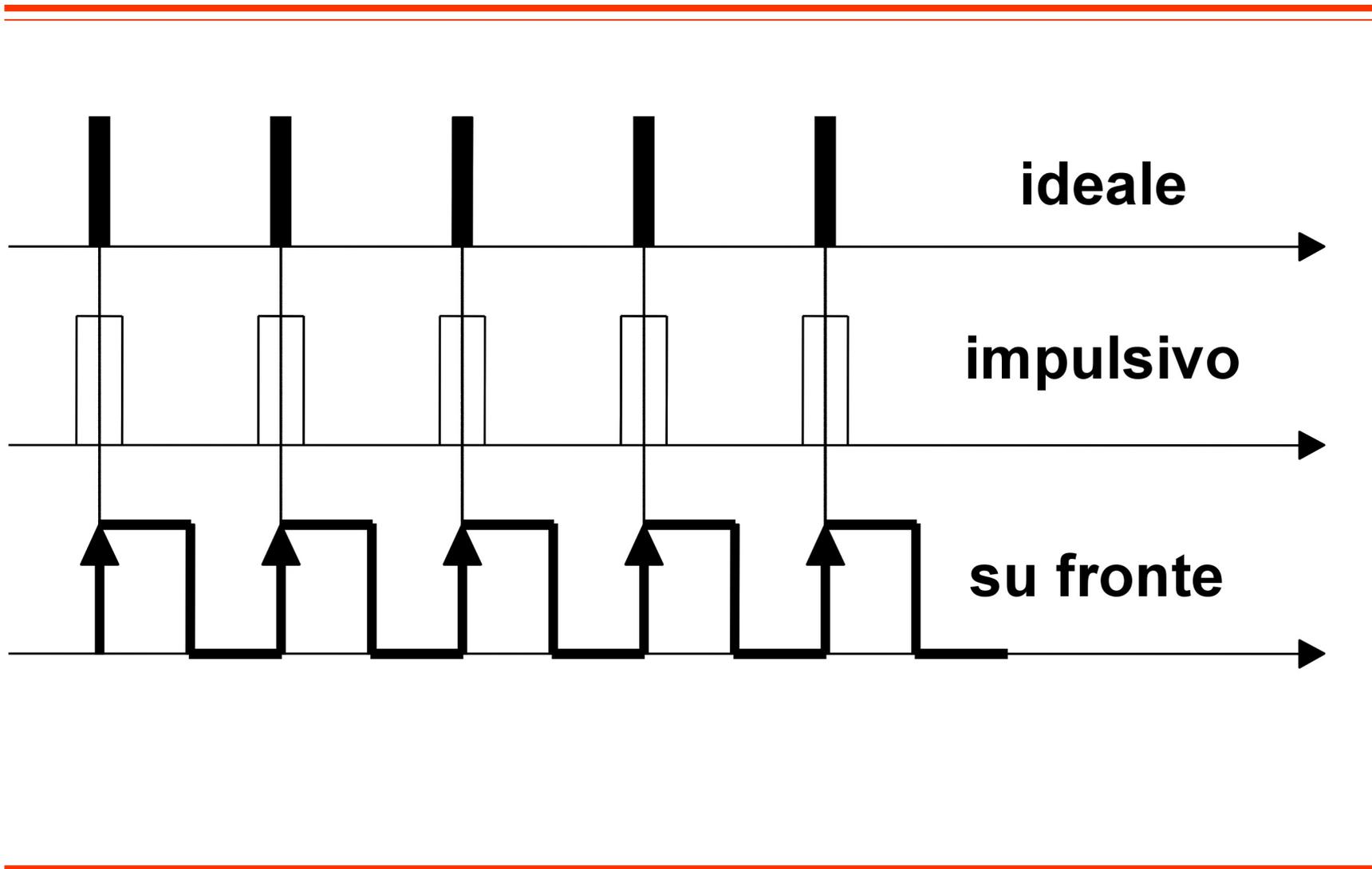


Il concetto di tempo

- Nel periodo T_{CK} , o *ciclo di clock*, il segnale assume:
 - Il valore logico 1 per un tempo T_H
 - Il valore logico 0 per un tempo T_L
- Il rapporto T_H / T_{CK} è detto *duty-cycle*
- Il passaggio dal valore 0 al valore 1 è detto *fronte di salita*
- Il passaggio dal valore 1 al valore 0 è detto *fronte di discesa*
- In genere, uno o entrambi i fronti del clock attivano i componenti con memoria (flip-flop e memorie)



Clock



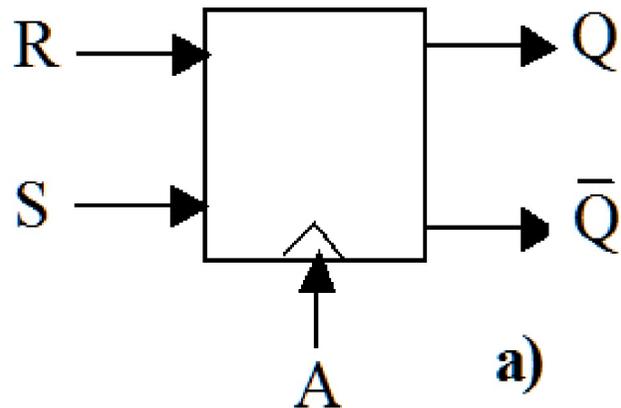
Flip-flop edge triggered

Un flip-flop, e di conseguenza un registro, edge-triggered è sensibile ad una variazione del segnale di abilitazione A e non durante tutto il periodo in cui A è attivo.

Edge-triggered sul fronte di salita (ETs): sensibile al fronte basso-alto (0 -> 1) del segnale A

Edge-triggered sul fronte di discesa (ETd): sensibile al fronte alto-basso (1 -> 0) del segnale A

RS: sincronizzazione latch ed edge triggered



q_{ij} è lo stato in cui presento il valore i e mi preparo ad a presentare in uscita il valore j , in corrispondenza del prossimo fronte di salita (o discesa) del clock

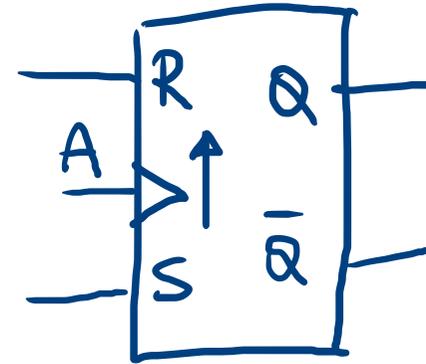
RS	A=0				A=1				Q
	00	01	11	10	00	01	11	10	
stat	q_{00}	q_{01}	--	q_{00}	q_{00}	q_{00}	--	q_{00}	0
q_0	q_0	q_0	--	q_0	q_0	q_1	--	q_0	0
q_1	q_1	q_1	--	q_1	q_1	q_1	--	q_0	1

Latch

RS	A=0				A=1				Q
	00	01	11	10	00	01	11	10	
stat	q_{00}	q_{01}	--	q_{00}	q_{00}	q_{00}	--	q_{00}	0
q_{01}	q_{00}	q_{01}	--	--	--	q_{11}	--	--	0
q_{10}	q_{11}	--	--	q_{10}	--	--	--	q_{00}	1
q_{11}	q_{11}	q_{11}	--	q_{10}	q_{11}	q_{11}	--	q_{11}	1

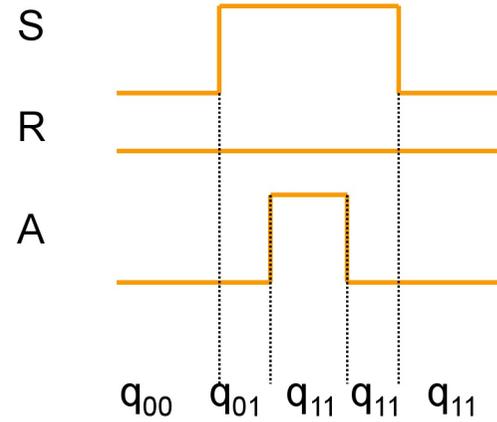
A variazione del fronte (edge-triggered)

Un RS sincronizzato – edge-triggered



RS	A=0				A=1				Q
	00	01	11	10	00	01	11	10	
stati	Q ₀₀	Q ₀₁	Q ₀₀	0					
Q ₀₁	Q ₀₀	Q ₀₁	--	--	--	Q ₁₁	--	--	
Q ₁₀	Q ₁₁	--	--	Q ₁₀	--	--	--	Q ₀₀	1
Q ₁₁	Q ₁₁	Q ₁₁	--	Q ₁₀	Q ₁₁	Q ₁₁	--	Q ₁₁	

Handwritten annotations: Blue arrows and circles indicate state transitions. A blue bracket on the left labels the top two rows as 'RESET' and the bottom two rows as 'SET'. Red arrows point to the clock input 'A' and the 'Q' output column.



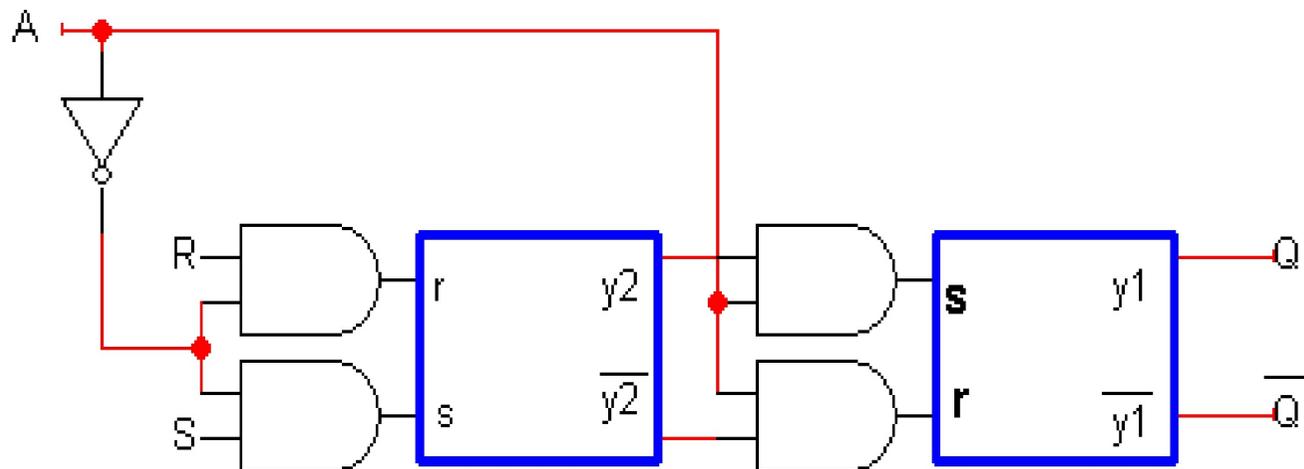
A	S	R	Q
0	-	-	Q _p
↑	0	0	Q _p
↑	0	1	0
↑	1	0	1
1	-	-	Q _p

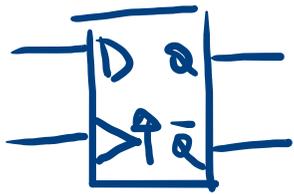
Handwritten annotations: Red arrows point to the 'Q' column and the second row of the table.

RS edge-triggered

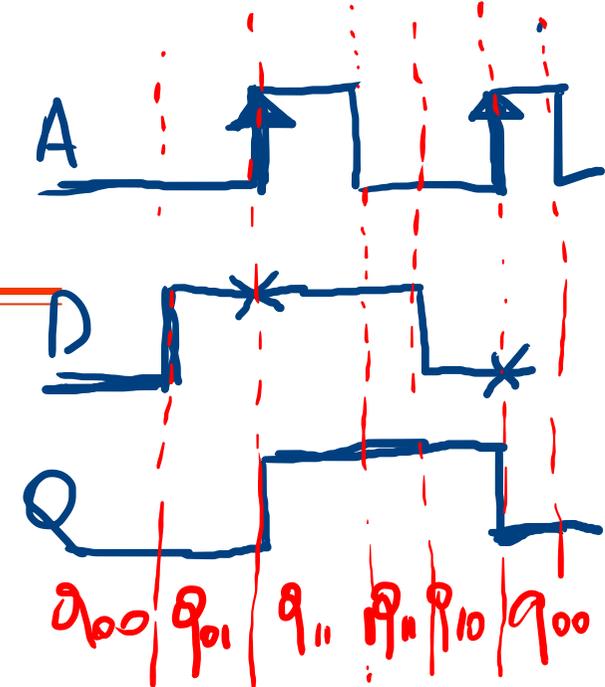
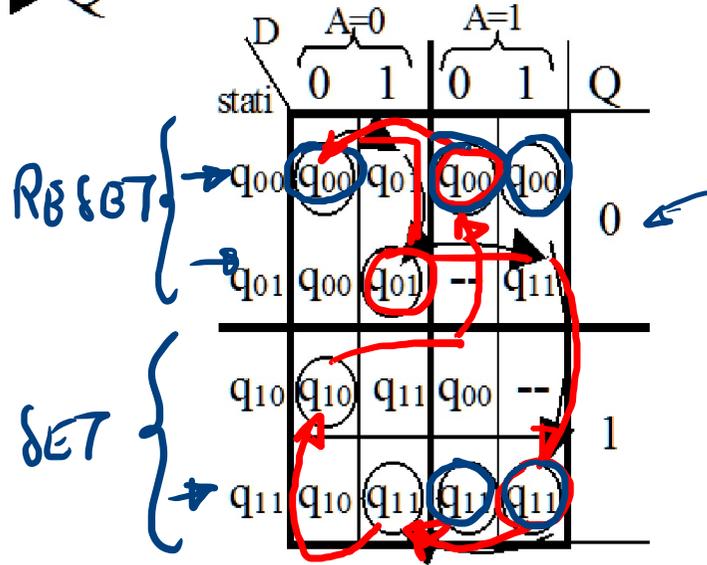
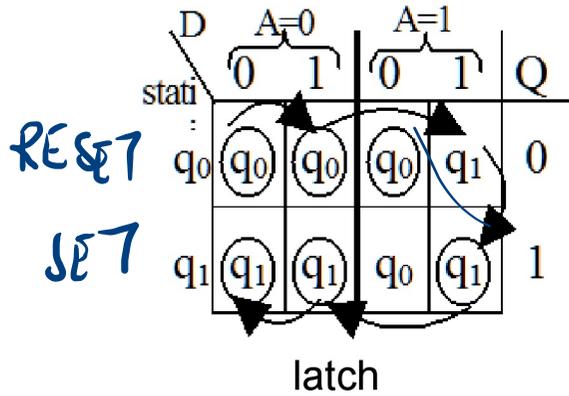
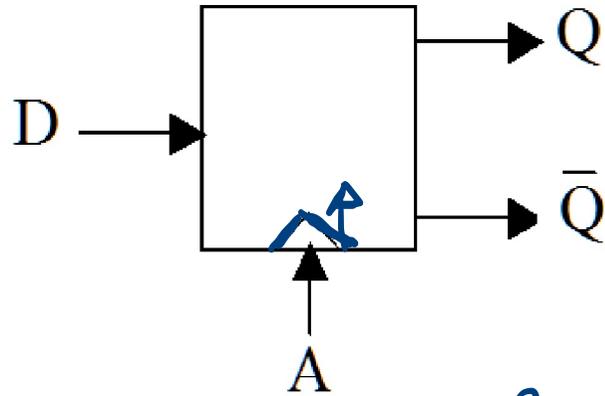
Possibile realizzazione di un flip-flop RS sincronizzato edge-triggered sul fronte di salita:

- quando $A=0$, il latch di sinistra “cattura” il segnale su R o su S e
- appena A passa da 0 ad 1 (fronte di salita), y2 è ricopiato in $y1=Q$ mentre gli ingressi del latch di sinistra restano neutri lasciandolo bloccato

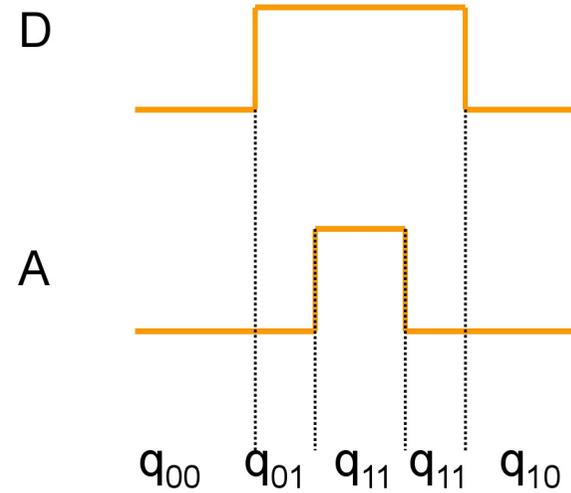
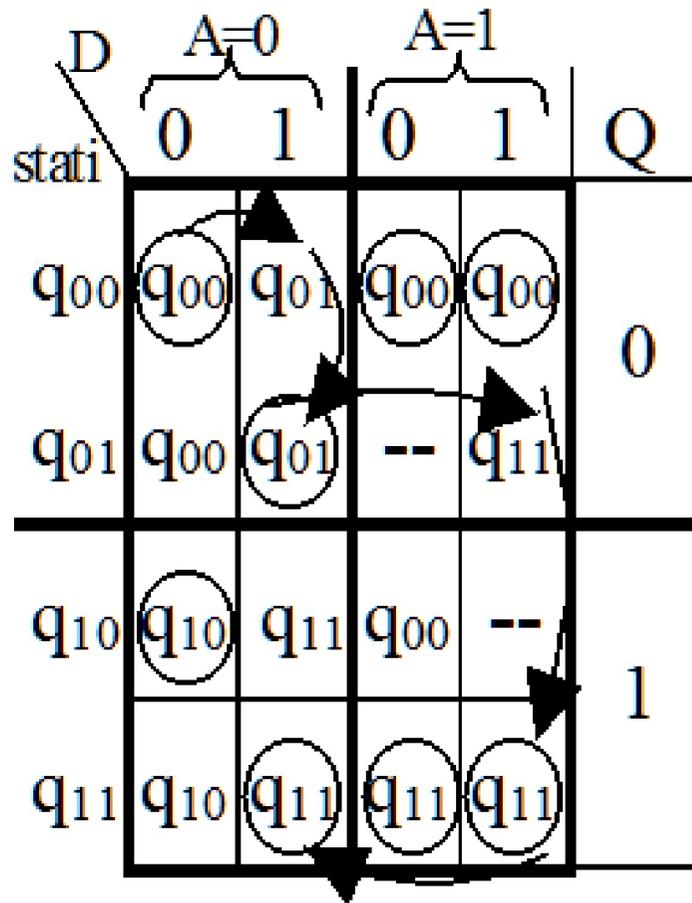




Flip flop D



Flip flop D

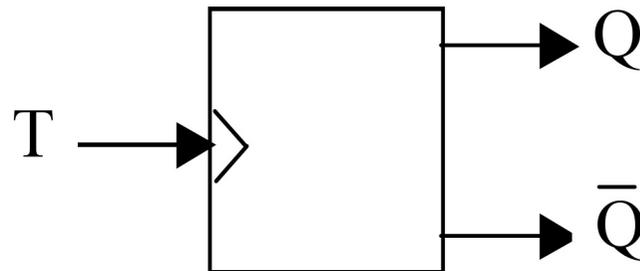


A	D	Q
0	-	Q _p
↑	0	0
↑	1	1
1	-	Q _p

Flip-Flop a commutazione

- un valore di ingresso provoca la *commutazione* dello stato piuttosto che indicarne esplicitamente il valore
 - due tipi
 - **T**: ha solo l'ingresso di commutazione
 - **JK**: analogo all'RS, con ingressi $JK=11$ corrispondenti all'ingresso di commutazione
-

Flip-Flop T

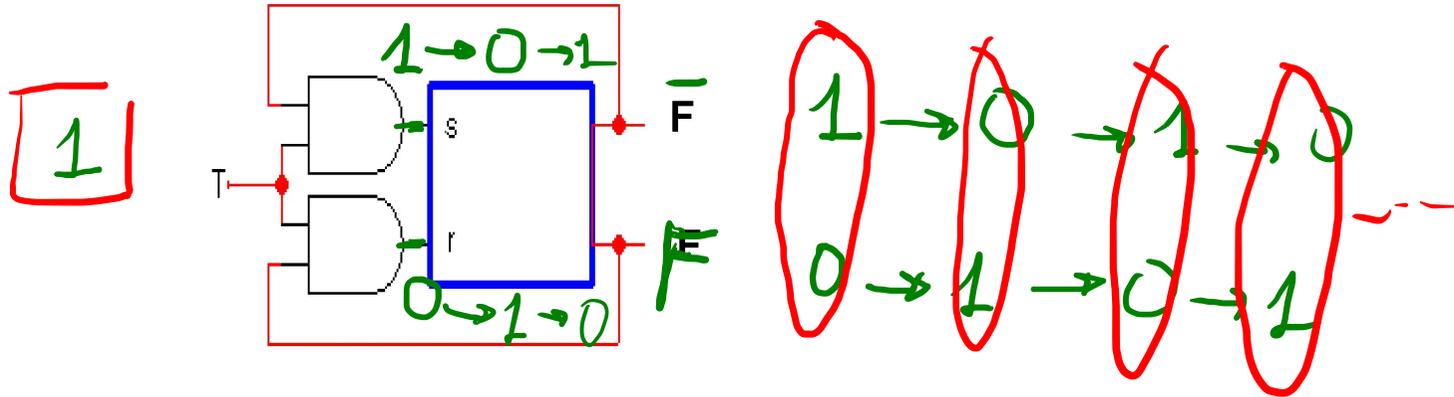


- Equazione di stato: $Q = \bar{Q}_p T + Q_p \bar{T}$
 - La macchina che implementa questo funzionamento deve essere necessariamente asincrona o sincrona impulsiva
-

Flip-Flop T sincrono con RS latch: tempificazione

- Equazioni caratteristiche

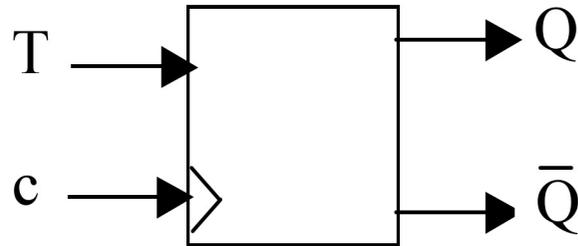
$$R = T \cdot F$$
$$S = T \cdot \overline{F}$$



- La corretta tempificazione dipende dalla durata di T

NON UTILIZZABILE IN PRATICA

Flip-Flop T abilitato



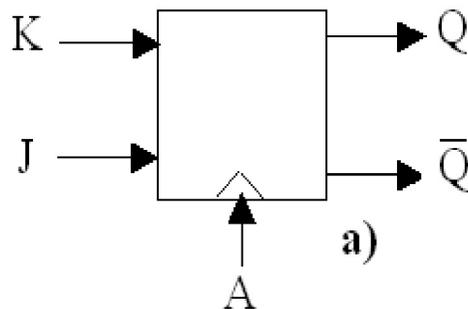
L'impulso sull'abilitazione c fa commutare il flip-flop se T è attivo.

In sostanza, T fa da abilitazione, mentre c diventa l'ingresso di commutazione

Flip-Flop JK

E' un flip-flop sincronizzato definito come segue:

- quando l'ingresso impulsivo di sincronizzazione A è attivo, i due ingressi a livello J , K si comportano rispettivamente come S , R se sono attivi separatamente..
- mentre producono la commutazione se lo sono simultaneamente.



$$Q = j\bar{Q}_p + Q_p\bar{k} = AJ\bar{Q}_p + Q_p\bar{K} + Q_p\bar{A}$$

Flip-Flop JK

KJ	A=0				Q
stati	00	01	11	10	
q ₀	q ₀	q ₁	q ₁	q ₀	0
q ₁	q ₁	q ₁	q ₀	q ₀	1

JK sincrono impulsivo

KJ	A=0				A=1				Q
stati	00	01	11	10	00	01	11	10	
q ₀₀	q ₀₀	q ₀₁	q ₀₁	q ₀₀	0				
q ₀₁	q ₀₀	q ₀₁	q ₀₁	q ₀₀	--	q ₁₁	q ₁₁	--	
q ₁₀	q ₁₁	q ₁₁	q ₁₀	q ₁₀	--	--	q ₀₀	q ₀₀	1
q ₁₁	q ₁₁	q ₁₁	q ₁₀	q ₁₀	q ₁₁	q ₁₁	q ₁₁	q ₁₁	

JK asincrono edge-triggered