

Classe :4A MAN - 3A MAN - 5BIPMM

A.S. : 2021-2022

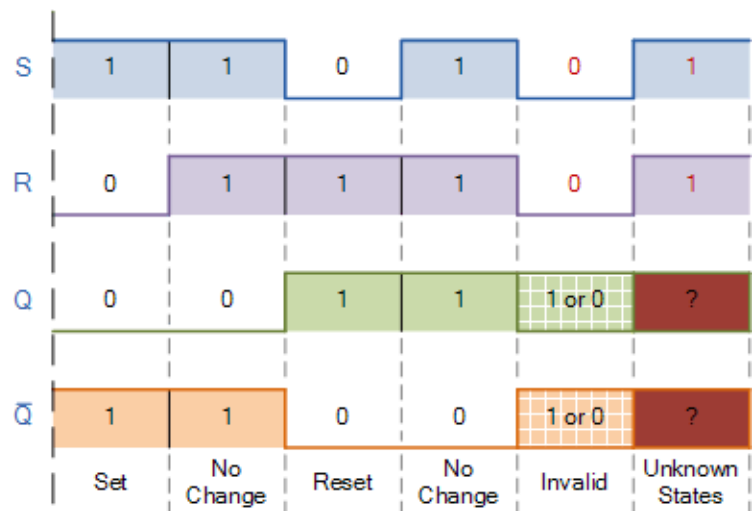
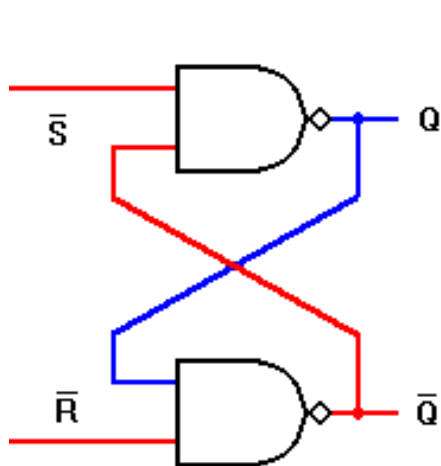
Docente :Tufoni Franco

Disciplina : Tecnologie elettriche-elettroniche e applicazioni

Reti Sequenziali

Flip – Flop

Open Source



Indice	
1.1 Premessa	2
1.2 Flip Flop SR (Set-Reset)	3
1.2.1 Flip Flop SR con porte NOR	3
1.2.2 Flip Flop SR con porte NAND	4
1.2.3	5
1.3	6
1.3.1	
1.3.2	
1.4 Flip Flop D	8
1.5 Flip Flop T	10

1.1 Premessa

I Flip Flop (FF) sono circuiti digitali sequenziali che hanno il compito di memorizzare un bit (0,1). Un circuito digitale si dice sequenziale se l'uscita dipende dagli ingressi applicati e dallo stato precedente della stessa uscita.

Un circuito sequenziale, pertanto, deve ricordare il suo stato precedente e quindi deve possedere uno o più elementi di memoria.

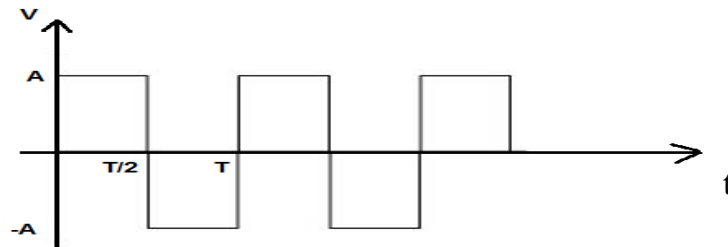
Rientrano in questa categoria i contatori, i registri e le memorie in generale.

I circuiti digitali si dividono in due fondamentali categorie:

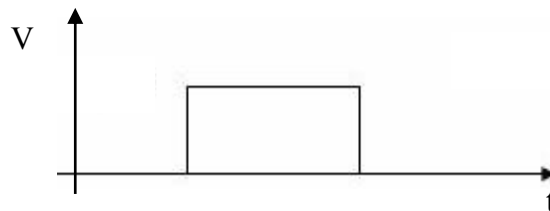
- **Combinatori:** il valore dell'uscita dipende solo dal valore dei bit applicati in ingresso, ad esempio le porte logiche, circuiti con porte logiche, multiplexer, decoder.
- **Sequenziali:** il valore dell'uscita dipende dal valore dei bit applicati in ingresso e dal suo stato precedente, ad esempio Flip Flop, contatori, registri, memorie.

I Flip Flop (FF) sono noti, anche, come multivibratori bistabili perché ciascuno degli stati logici (0 e 1) può essere reso stabile nel tempo. I multivibratori si dividono in:

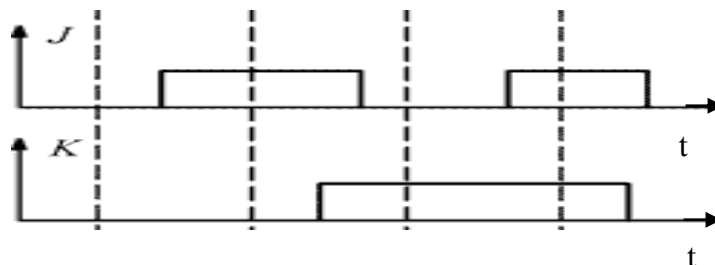
- **Astabili:** nessuno stato stabile, ad esempio i generatori di onde quadre o i lampeggiatori.



- **Monostabili:** un solo stato stabile, ad esempio temporizzatori o le luci delle scale.



- **Bistabili:** due possibili stati stabili, ad esempio una cella di memoria o un interruttore.

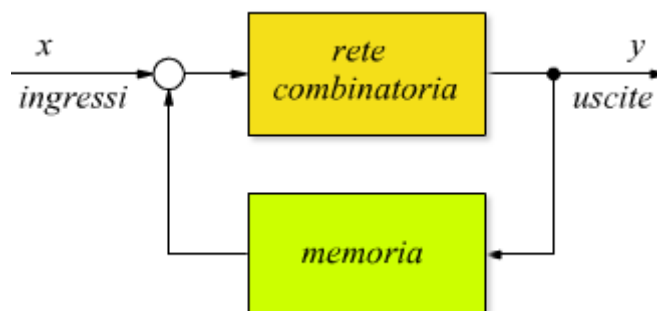


Nelle reti combinatorie, il valore dell'uscita dipende esclusivamente dal valore assunto dagli ingressi in quel dato istante.

Nelle reti sequenziali, il valore dell'uscita non dipende solo dal valore assunto dagli ingressi in un dato istante, ma anche dai livelli logici assunti dalle uscite negli istanti precedenti. Si può dire che nelle reti combinatorie il valore dell'uscita non dipende dal tempo, mentre nelle reti sequenziali la variabile tempo riveste una grande importanza.

Se in un determinato istante t_n gli ingressi e le uscite si trovano in un determinato stato, è possibile che nell'istante successivo t_{n+1} una variazione dei valori logici di ingresso determini un nuovo stato delle uscite. Se lo stato delle uscite dipende, sia dallo stato degli ingressi al tempo t_{n+1} sia da quello delle uscite all'istante precedente t_n , allora, si può dire che il circuito è in grado di conservare memoria della sequenza degli stati assunti dagli ingressi e dalle uscite.

Le strutture circuitali che possono realizzare queste funzionalità di memoria sono molteplici, ma sono tutte caratterizzate da un collegamento di reazione (feedback) che riporta in ingresso lo stato delle uscite.



Anche il modo in cui vengono effettuate le variazioni in ingresso possono cambiare.

Nel caso del latch, la semplice variazione in ingresso determina una variazione dell'uscita. In alcune configurazioni come il latch con abilitazione il circuito accetta la variazione degli ingressi solo se l'ingresso supplementare di abilitazione è attivato.

In altri casi, come il flip-flop è invece presente un segnale di cadenza (clock) in corrispondenza del quale vengono scansionati (triggerati) gli ingressi e soltanto in corrispondenza di tale controllo avvengono le eventuali variazioni delle uscite.

Questi dispositivi rappresentano gli elementi minimi per memorizzare un bit (0 o 1) e possono essere usati per costruire reti sequenziali più complesse, come contatori, registri ed automi.

Differenza tra Latch e Flip Flop

Il latch modifica lo stato logico dell'uscita al variare del segnale di ingresso, mentre il flip-flop, basato sulla struttura del latch, cambia lo stato logico dell'uscita solamente quando il segnale di clock è nel semiperiodo attivo.

1.2 Flip Flop SR (Set-Reset) (Latch)

Il più semplice dispositivo memoria è il flip flop set-reset.

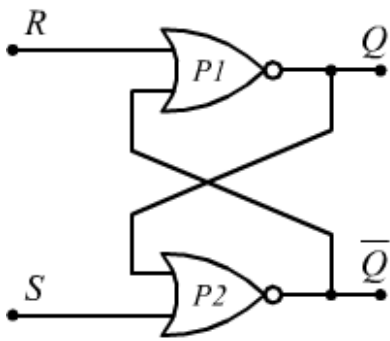
- **Set** = memorizzare
- **Reset** = azzerare

Esso possiede due ingressi denominati Set (S) e Reset (R) e due uscite denominate Q e Q negato. Nel Flip Flop (Latch) l'uscita dipende dagli ingressi e dallo stato precedente.

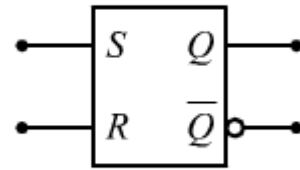
Lo stato precedente viene spesso indicato con Q_0 , oppure Q_n .

1.2.1 Flip Flop SR con porte NOR (Latch)

In figura il simbolo logico, la tabella della verità e il circuito con porte logiche NOR.



S	R	Q_{n+1}	
0	0	Q_n	memoria
0	1	0	reset
1	0	1	set
1	1	X	non ammesso



Funzionamento

- **Combinazione SR = 00**

Combinazione di riposo l'uscita conserva lo stato precedente ($Q = Q_n$)

- **Combinazione SR = 01**

L'uscita Q si porta a 0 indipendentemente dallo stato precedente (fase di Reset)

- **Combinazione SR = 10**

L'uscita Q si porta a 1 indipendentemente dallo stato precedente (fase di Set, memorizzazione)

- **Combinazione SR = 11**

Combinazione da evitare nel Flip Flop di tipo NOR perché l'uscita Q non è diversa da Q negato.

Porta logica NOR $Y = \overline{A+B}$

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

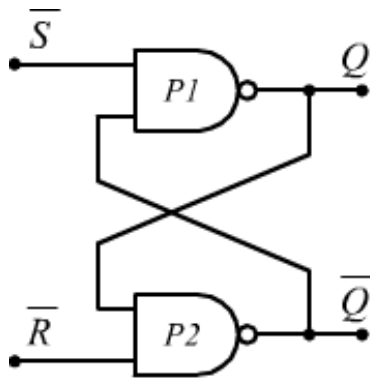


Tale Flip-Flop viene spesso utilizzato per funzionare nel modo seguente.

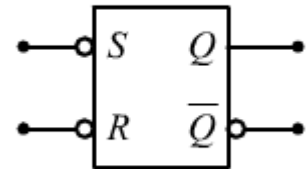
- Se si vuole memorizzare 1 si pone: $S=1$ e $R=0$. Successivamente si torna nello stato di riposo: $S=0$ e $R=0$. In tal caso l'uscita conserva lo stato precedente: $Q=Q_0=1$.
- Se si vuole memorizzare 0 si pone: $S=0$ e $R=1$. Successivamente si torna nello stato di riposo: $S=0$ e $R=0$. In tal caso l'uscita conserva lo stato precedente: $Q=Q_0=0$.

1.2.2 Flip Flop SR con porte NAND (Latch)

E' possibile costruire il latch anche con le porte NAND. La verifica dell'operatività di questo dispositivo può essere fatta in modo analogo a quanto visto per il latch a porte NOR; bisogna notare però, che in questo caso gli ingressi sono attivi bassi (S ed R).



\bar{S}	\bar{R}	Q_{n+1}	
0	0	X	non ammesso
0	1	1	set
1	0	0	reset
1	1	Q_n	memoria



Infatti per ottenere, ad esempio, la condizione di reset occorre attivare R portandolo a 0. Lo stato di memoria, quindi lo stato inattivo di riposo è stavolta abbinato ai livelli logici di ingresso $S=R=1$; in questa condizione, infatti, P1 e P2 invertono gli stati presenti sugli ingressi di reazione e quindi mantengono le uscite nello stato precedente. Applicando agli ingressi lo stato logico 0, la situazione sulle uscite diventa $Q=\bar{Q}=1$: questo stato di ambiguità dove il valore dell'uscita è uguale a quello della sua complementare non deve pertanto essere usato.

Porta logica NAND $Y = \overline{A \cdot B}$

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0



Q_{n+1} = Stato attuale
 Q_n = Stato precedente

SEGNALE DI CLOCK

Il clock è un segnale digitale (onda quadra o rettangolare) caratterizzato dai seguenti parametri.

1. T = Periodo (s)
2. F = Frequenza (Hz) $f = 1/T$ (numero di oscillazioni in un secondo)
3. Livello basso (0, Low)
4. Livello alto (1, High)
5. Fronte di salita = passaggio da 0 a 1 (da basso ad alto)
6. Fronte di discesa = passaggio da 1 a 0 (da alto a basso)
7. Tempo livello alto (t_H)
8. Tempo livello basso (t_L) $T = t_L + t_H$
9. Duty cycle (ciclo utile) spesso indicato in percentuale e viene determinato tramite la seguente formula $D\% = t_H/T * 100$

Se $D\%$ è uguale al 50% significa che il livello alto è uguale al livello basso ($t_L = t_H$)

Il segnale si chiama onda quadra.

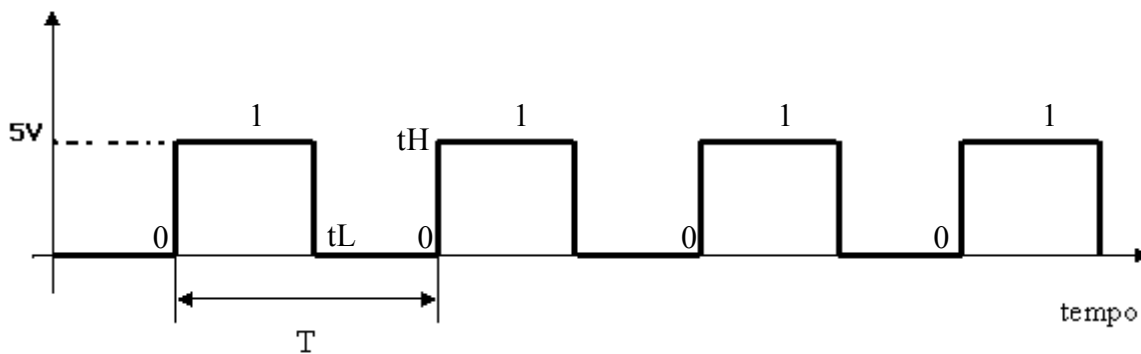
Se t_H è diverso da t_L il Duty Cycle è diverso del 50%

Il segnale si chiama onda rettangolare.

Si possono verificare due casi:

a) $t_H > t_L \rightarrow D\% > 50\%$

b) $t_H < t_L \rightarrow D\% < 50\%$



FLIP-FLOP SINCRONIZZATI

Spesso l'eventuale cambiamento di stato di un flip-flop non si fa coincidere con l'istante in cui si modificano i valori dei bit di ingresso ma con l'istante in cui un ulteriore ingresso, detto ingresso di sincronismo o ingresso di clock e denominato con la sigla CK, va da 1 a 0 oppure da 0 a 1.

Un flip-flop che funziona col clock prende il nome di flip-flop *sincronizzato*.

I flip-flop esaminati nelle figure 1 e 2, invece, essendo privi di ingresso di clock, sono denominati flip-flop Set Reset *asincroni*.

Il flip-flop si dice *attivo sul livello* quando il clock è caratterizzato da un livello logico (0 o 1) applicato all'omonimo ingresso.

Il flip-flop si dice *attivo su fronti* (edge triggered) quando l'eventuale modifica dello stato di uscita dipende dal *fronte positivo* o di salita (PET = Positive Edge Triggered) o dal *fronte negativo* o di discesa (NET = Negative Edge Triggered).

Sui dice, anche, che l'uscita si aggiorna nel:

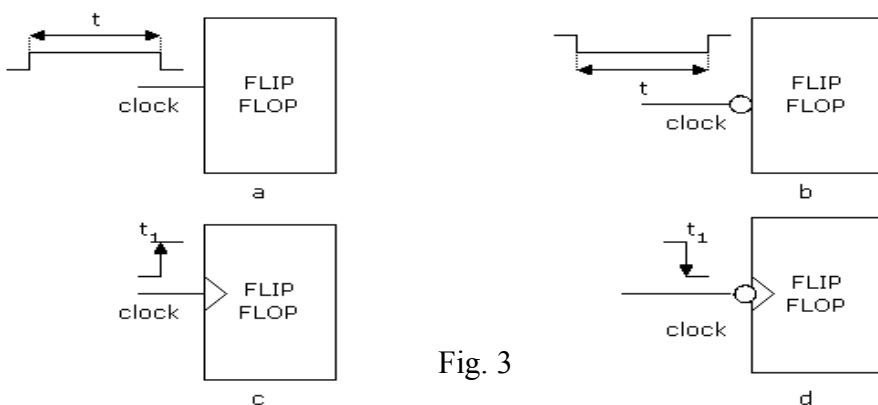


Fig. 3

Fig.3 - Flip-flop sensibili ai livelli ed alle transizioni..

Il flip-flop "a" è sensibile al livello logico 1 e funziona nell'intervallo di tempo t .

Il flip-flop "b" è sensibile al livello logico 0 durante il tempo t .

Il flip-flop "c" è di tipo PET e funziona solo nell'istante t_1 .

Il flip-flop "d", infine, è di tipo NET e funziona solo nell'istante t_1 .

4. Flip-flop JK

I Flip Flop visti in precedenza hanno sempre una condizione non ammessa (00 nel FF set-reset a porte nand, 11 nel FF set-reset a porte nor).

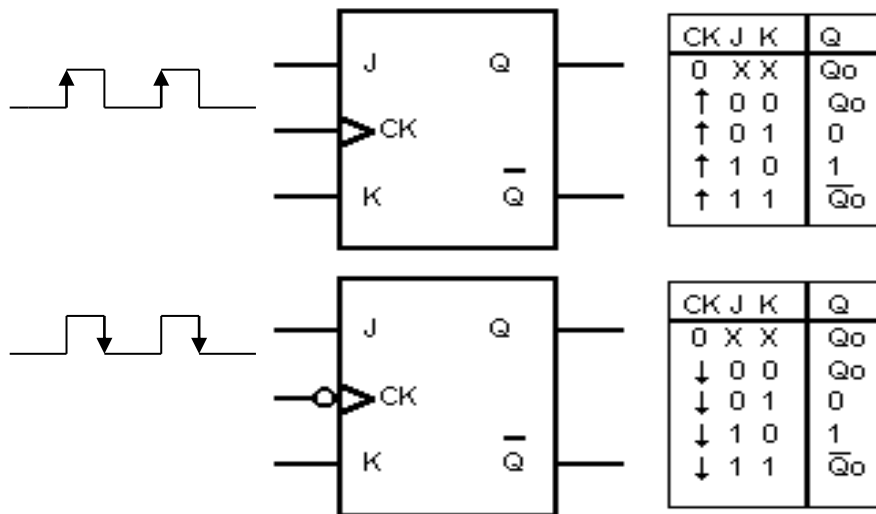
I FF-JK sono strutturati in modo tale da eliminare la condizione non ammessa.

Il FF-JK è un dispositivo integrato a due ingressi denominati J e K che operano in modo analogo alle entrate S ed R di un FF-SR con la differenza che se:

J=1 e K=1 l'uscita commuta sempre allo stato precedente negato, se è presente 0 passa 1, mentre se è presente 1 passa 0.

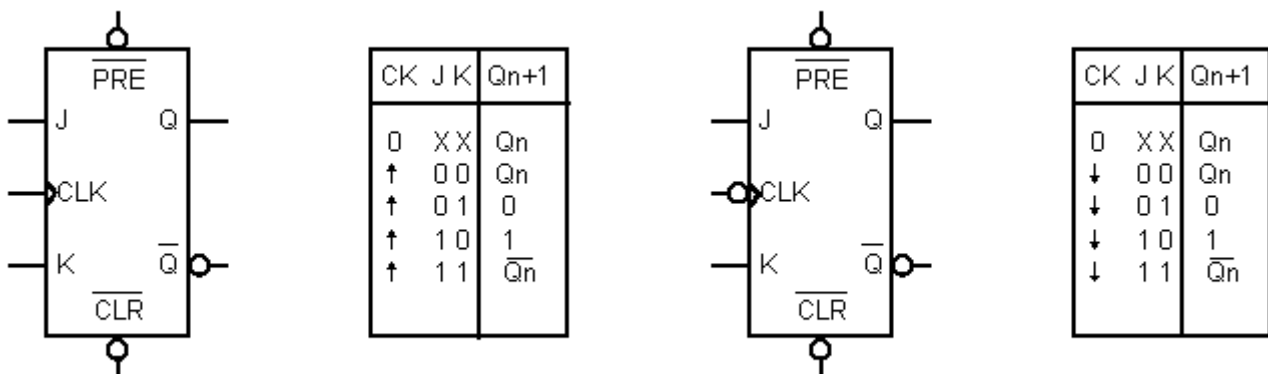
Per evitare commutazioni multiple, nel caso si lasci a lungo la combinazione J=1 e K=1, il FF-JK deve essere necessariamente sincronizzato con un segnale di clock.

In figura sono riportati 2 FF-JK con le rispettive tabelle di funzionamento, il primo sincronizzato sul fronte di salita e il secondo sincronizzato sul fronte di discesa.



Il primo flip flop JK si dice di tipo PET (Positive Edge Triggered), il secondo, invece, si dice di tipo NET (Negative Edge Triggered).

Si mostrano, in fig.6, i simboli logici e le tabelle della verità di due diversi flip-flop JK, il primo di tipo PET ed il secondo di tipo NET, con ingressi asincroni di preset (PRE) e di clear (CLR).

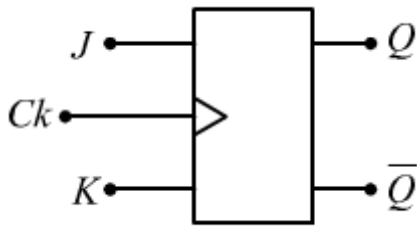


La maggior parte dei flip-flop integrati presentano, inoltre, ingressi asincroni di preassegnazione a 1 (PRE) e/o di azzeramento (CLR clear).

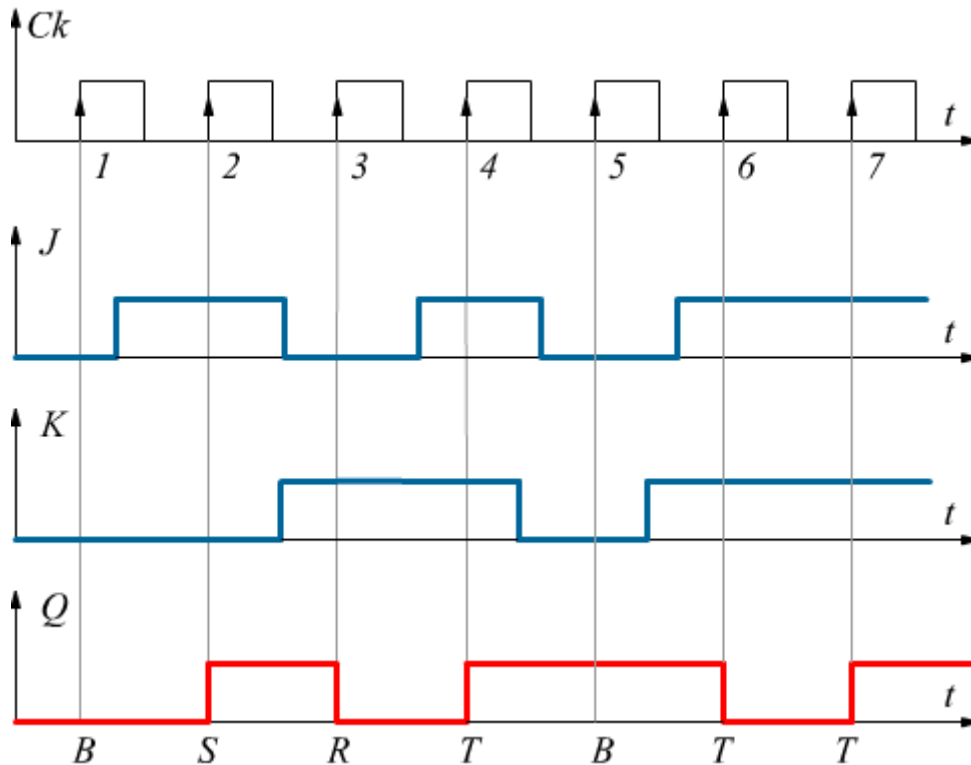
Quelli indicati in figura sono attivi bassi: se, ad esempio, si applica un livello logico basso sulla linea PRE, il flip-flop memorizza 1 indipendentemente dal clock; se, invece, si applica un livello logico basso sulla linea CLR, il flip-flop memorizza 0 indipendentemente dal clock.

Esempio FF-JK Positive Edge Triggered (PET)

Simbolo, tabella e diagrammi temporali



CK	J	K	Q
0	X	X	Q_0
↑	0	0	Q_0 (Block)
↑	0	1	0 (Reset)
↑	1	0	1 (Set)
↑	1	1	\bar{Q}_0 (Toggle)



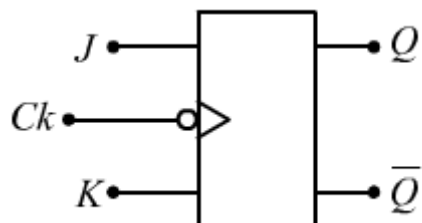
Il nome edge-triggered sta a significare che le eventuali commutazioni possono verificarsi solo in presenza del segnale di clock, in particolare, in corrispondenza dei fronti di salita oppure dei fronti di discesa di tale segnale.

Nell'esempio in alto è riportato il principio di funzionamento di un FF-JK a logica positiva (commutazioni sui fronti di salita del clock). Come si nota se:

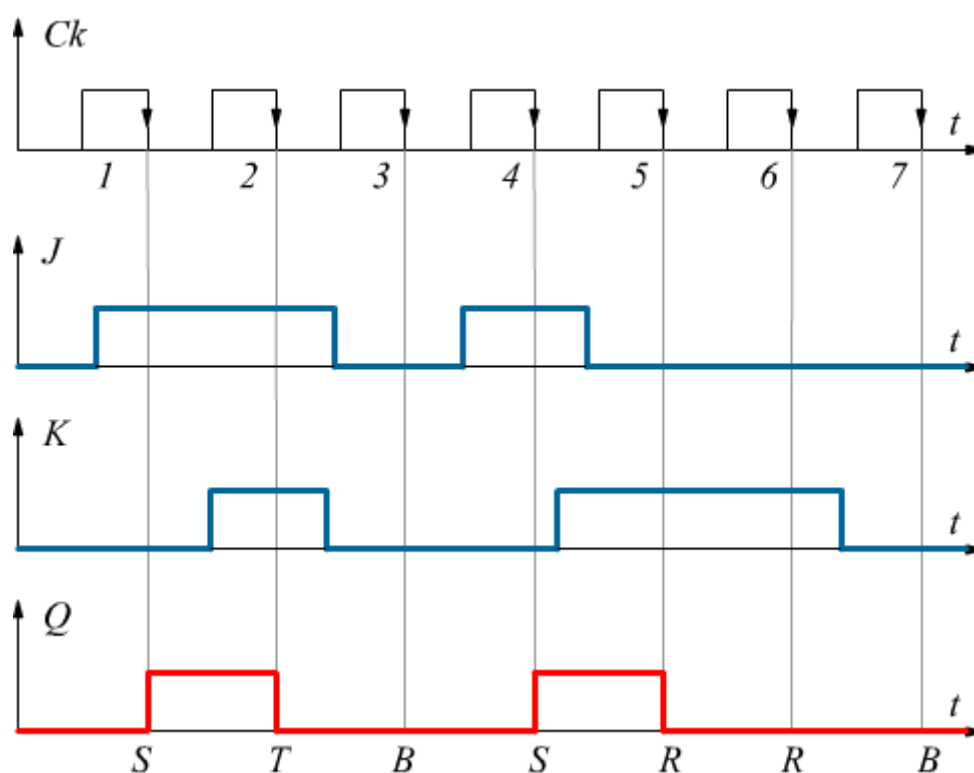
- **J=1 e K=0** si ha **Q=1** ed è la condizione di **SET**
- **J=0 e K=1** si ha **Q=0** ed è la condizione di **RESET**
- **J=0 e K=0** l'uscita Q riconferma lo stato in cui si trovava l'uscita prima del fronte di salita dell'impulso di clock, ed è la condizione di **BLOCK**
- **J=1 e K=1** l'uscita Q commuta rispetto allo stato in cui si trovava l'uscita prima del fronte di salita dell'impulso di clock, ed è la condizione di **TOGGLE** (commutazione rispetto allo stato precedente -**stato precedente negato**).

Esempio FF-JK Negative Edge Triggered (NET)

Simbolo, tabella e diagrammi temporali



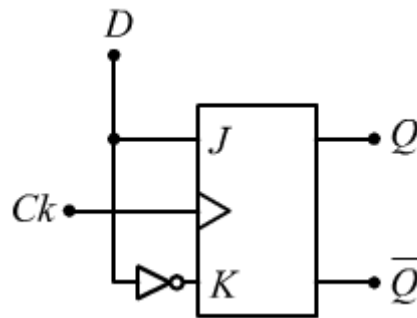
CK	J	K	Q
0	X	X	Q_0
↓	0	0	Q_0 (Block)
↓	0	1	0 (Reset)
↓	1	0	1 (Set)
↓	1	1	\bar{Q}_0 (Toggle)



Flip Flop tipo D

Il flip flop D si trova integrato e deriva dal JK ha un solo ingresso denominato D.

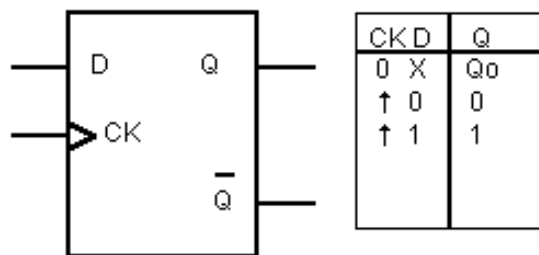
All'interno dell'integrato tra il terminale J e K è collegata una porta NOT come indicato in figura



Il flip flop D, spesso sincrono, funziona nel seguente modo:

- l'uscita Q assume lo stesso valore applicato all'ingresso D quando è attivo il segnale di clock;
- quando non è attivo il clock l'uscita conserva lo stato precedente indipendentemente dal valore applicato all'ingresso D.

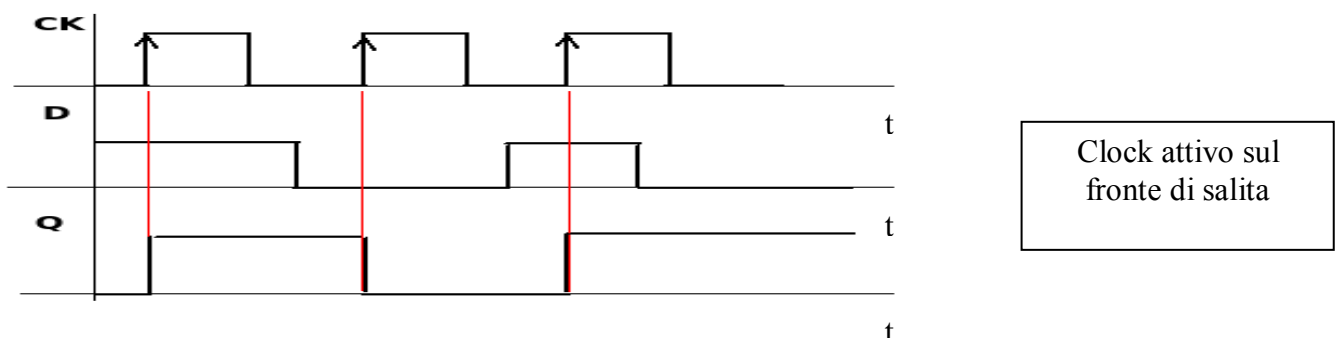
In figura è riportato il simbolo logico e la sua tabella di funzionamento



Il simbolo D è l'abbreviazione di delay (ritardo), rappresenta il ritardo tra l'uscita Q e l'ingresso D.

Il flip-flop D rappresenta l'elemento fondamentale delle memorie centrali degli elaboratori (almeno a livello concettuale).

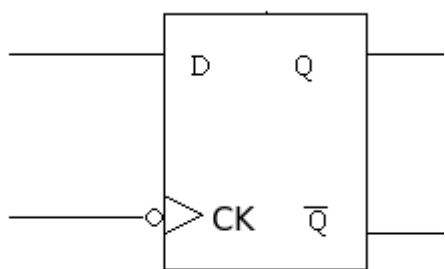
In figura è riportato un diagramma temporale del Flip Flop D che mostra un esempio di funzionamento.



L'uscita Q è uguale all'ingresso D in corrispondenza del fronte di salita del clock (tratto verticale indicato in rosso).

In tutti gli altri istanti l'uscita mantiene lo stato precedente.

Nella figura seguente è riportato un esempio di diagramma temporale con il Clock attivo sul fronte di discesa.



Simbolo

CK	D	Q
0	X	Q ₀
↓	0	0
↓	1	1

Tabella

Si noti il simbolo di negazione (il "pallino") sull'ingresso CK che sta a indicare che il segnale è attivo sul fronte di discesa.

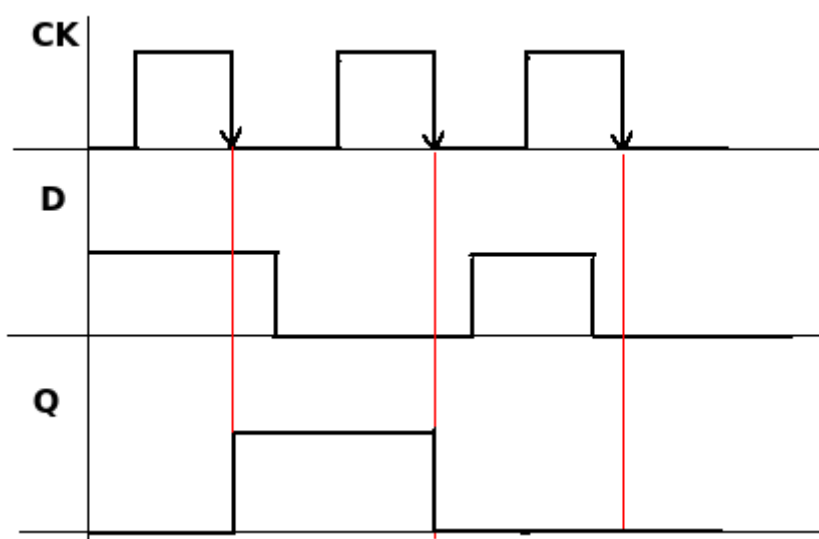
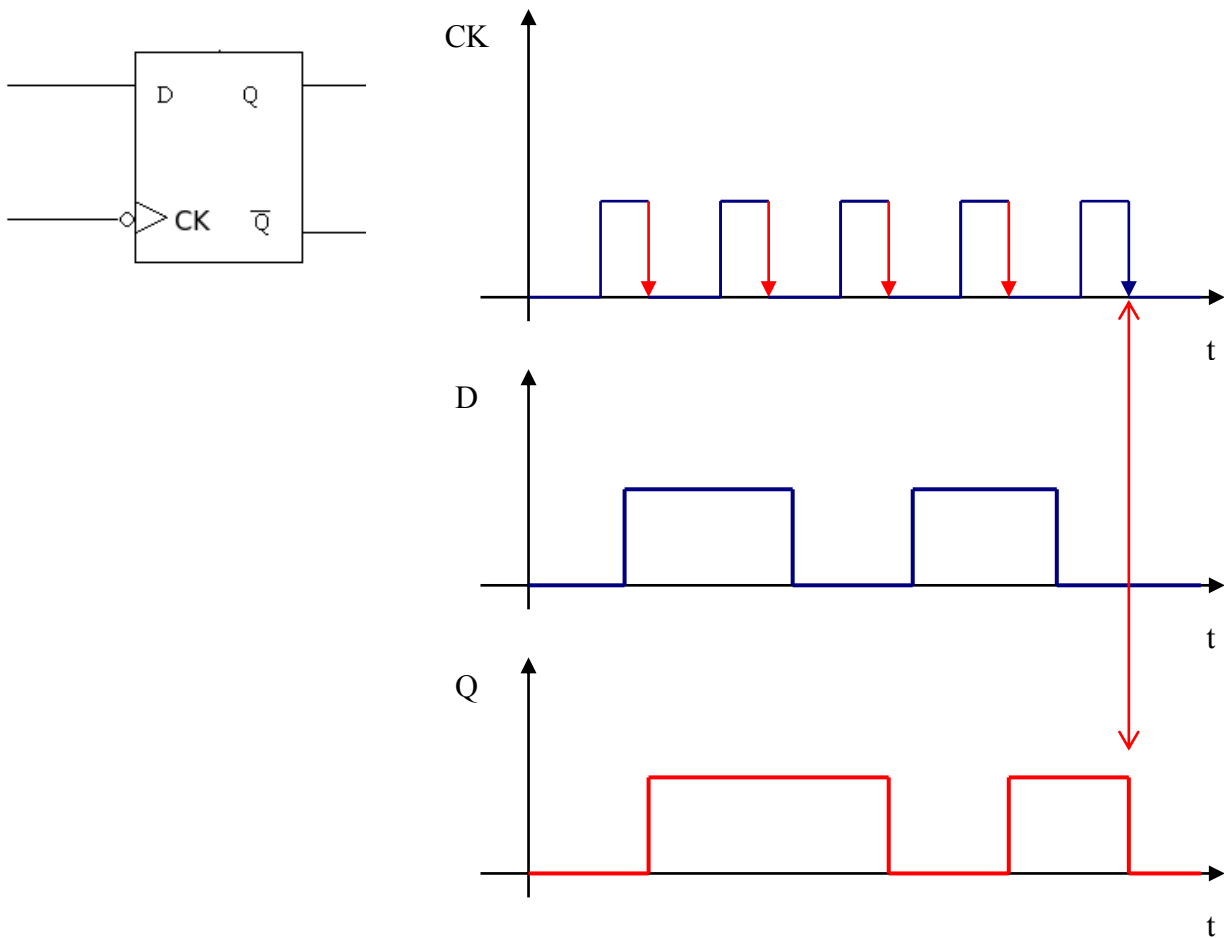


Diagramma temporale

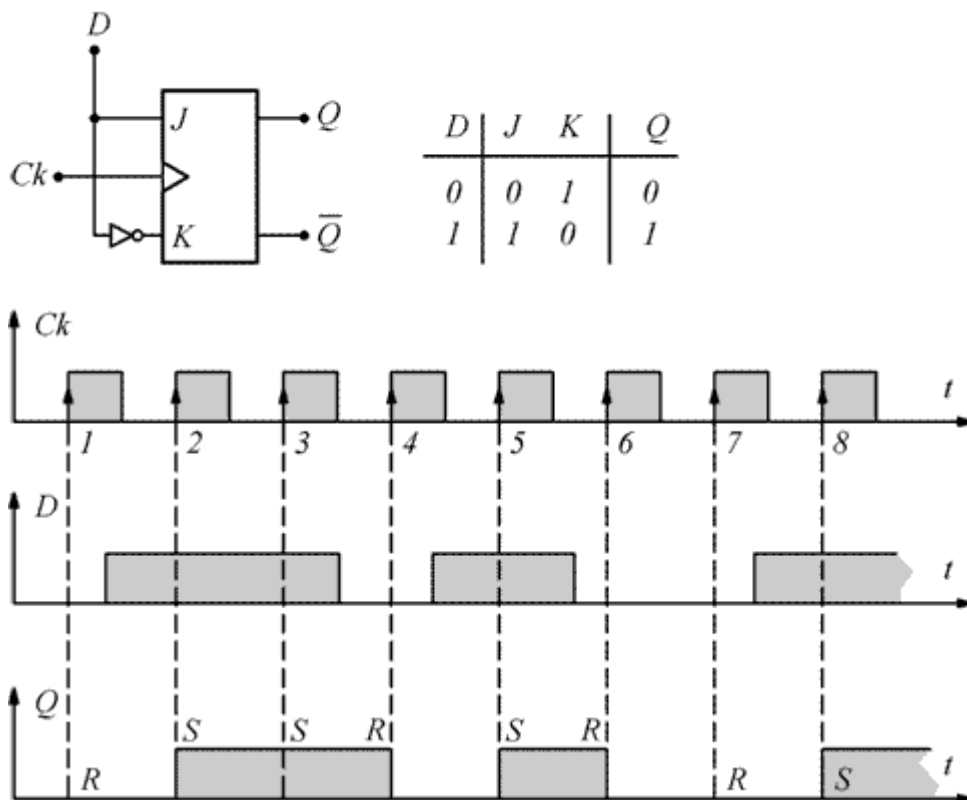
Esercizio-1 Flip Flop D:

Assegnato il Flip Flop D di figura determinare il diagramma temporale completo



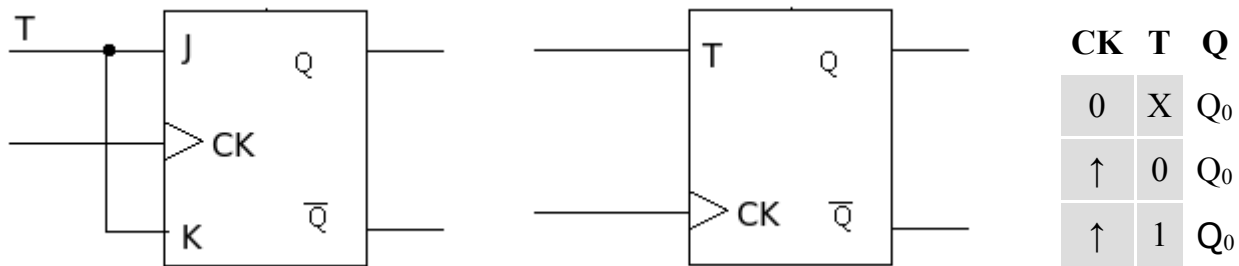
Esercizio-2 Flip Flop D:

Assegnato il Flip Flop D di figura determinare il diagramma temporale completo.



Flip flop T

Il Flip Flop T (toggle) può essere facilmente ottenuto dal FF JK collegando insieme i due ingressi:



Simbolo elettrico e tabella di verità del flip flop T sono mostrati in figura

Questo flip flop presenta solo due modi di funzionamento:

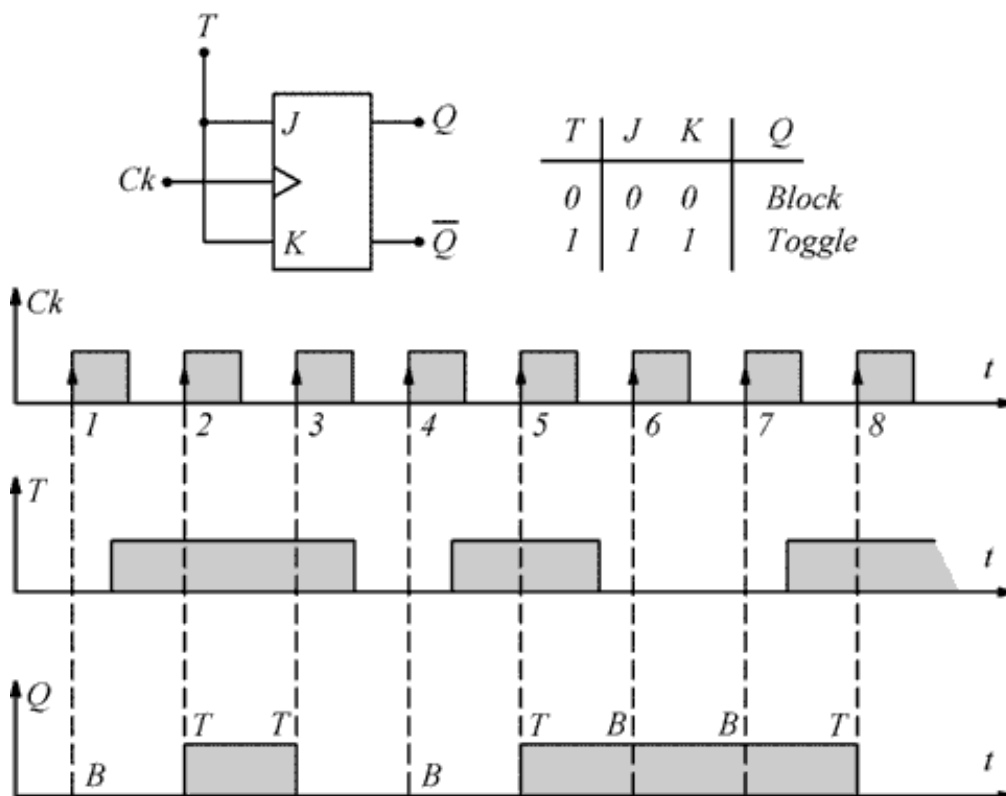
1. se $T=0$ ($J=0$ e $K=0$) il Flip Flop memorizza lo stato precedente (Q_0);
2. se $T=1$ ($J=1$ e $K=1$) il flip flop cambia stato (toggle), l'uscita è uguale allo stato precedente negato (Q_0 negato)

Il Flip Flop T è sempre sincrono, quindi il funzionamento è sempre sincronizzato col segnale di clock, nel nostro esempio il segnale è attivo sul fronte di salita.

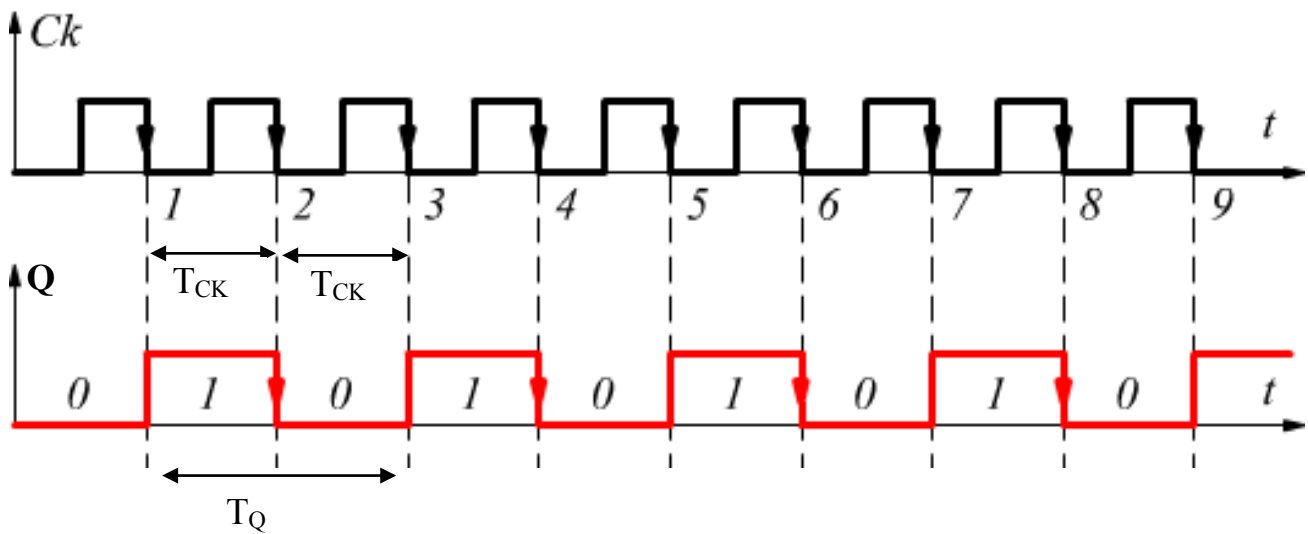
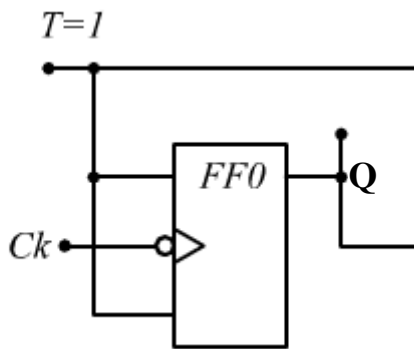
Quando il clock non è attivo qualsiasi valore viene inserito sull'ingresso T (simbolo x sulla tabella) l'uscita è uguale allo stato precedente Q_0 (vedi tabella)

Il Flip Flop T non è disponibile come circuito integrato, dato che può essere realizzato collegando insieme i due ingressi di un Flip Flop JK.

In figura è riportato un diagramma temporale che mostra il funzionamento descritto.



Flop T - Divisore di frequenza



T_{CK} = Periodo segnale di Clock

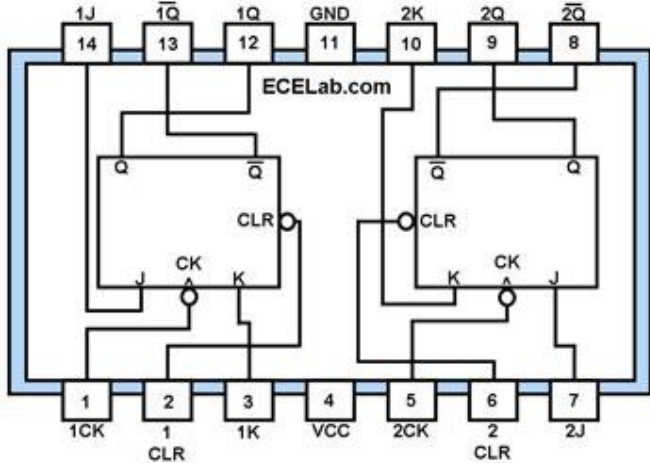
T_Q = Periodo segnale di uscita

$$T_Q = 2 T_{CK}$$

$$f_Q = 1/2 f_{CK}$$

Flip Flop integrati

Il Flip Flop Set-Reset e il flip flop T non sono disponibili come circuiti integrati, mentre si trovano diverse versioni del flip flop JK per tutte le principali famiglie logiche. A titolo di esempio mostriamo la piedinatura del 7473, un integrato TTL contenente due ff JK *pilotati sul fronte di discesa* del clock (si noti l'ingresso asincrono di *clear*):



Contatore digitale (digital counter): definizioni generali

In elettronica si definisce **contatore** (*counter*) un dispositivo in grado di *contare* il numero di impulsi di clock, fornendo in uscita una sequenza di valori numerici binari corrispondenti agli impulsi contati.

Nel caso più semplice e più frequente i valori numerici prodotti in uscita dal contatore partono da zero incrementandosi ogni volta di uno (*contatore a incremento*) oppure vanno all'indietro partendo da un valore massimo e decrementando di uno tale valore ad ogni fronte del clock (*contatore a decremento*). In entrambi i casi al termine del conteggio la sequenza riparte dal valore iniziale ciclicamente.

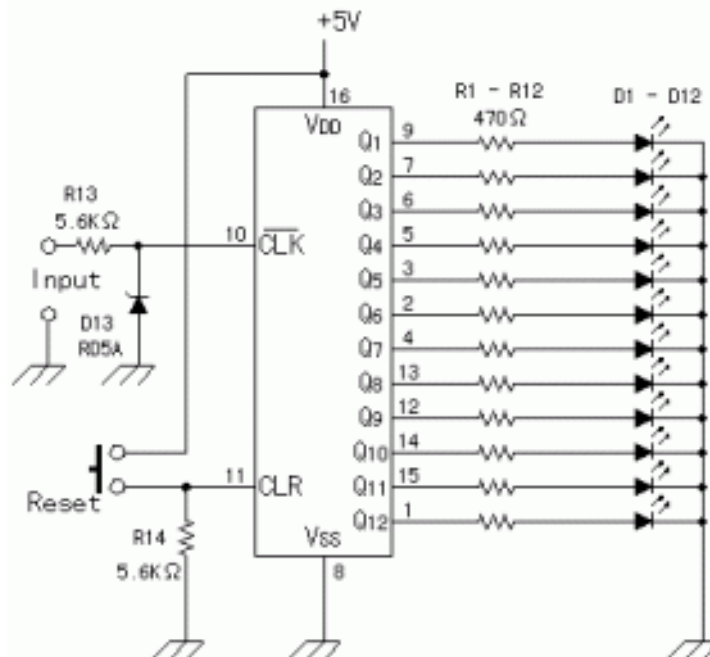
Per esempio un contatore a incremento a 3 bit produrrà in uscita la sequenza:

000 → 001 → 010 → 011 → 100 → 101 → 110 → 111 → 000 → 001 → 010 etc etc

Si dice **modulo** del contatore il numero di *combinazioni binarie* (dette anche *stati logici*) prodotte in uscita. Il contatore a 3 bit dell'esempio precedente ha modulo 8, in quanto produce in uscita 8 diverse combinazioni binarie (da 000 a 111). In generale in un contatore binario il *modulo M* è legato al *numero di bit n* dalla formula:

$$M = 2^n$$

A titolo di esempio la figura seguente mostra un circuito realizzato con l'integrato 4040B della serie CMOS, in cui il contatore produce l'accensione di 12 LED secondo la sequenza binaria di conteggio:



Contatore binario/divisore di frequenza

CONTATORE BINARIO ASINCRONO

Il contatore binario asincrono è il contatore di più facile da realizzare; in genere è formato dalla connessione di più flip flop, ad un solo ingresso, di solito JK usati come T, disposti in modo tale da poter effettuare il conteggio cambiando di stato ogni qualvolta giunge al suo ingresso un impulso.

Lo stato logico delle uscite dei bistabili istante per istante fornisce sotto forma binaria il numero degli impulsi entrati nel contatore.

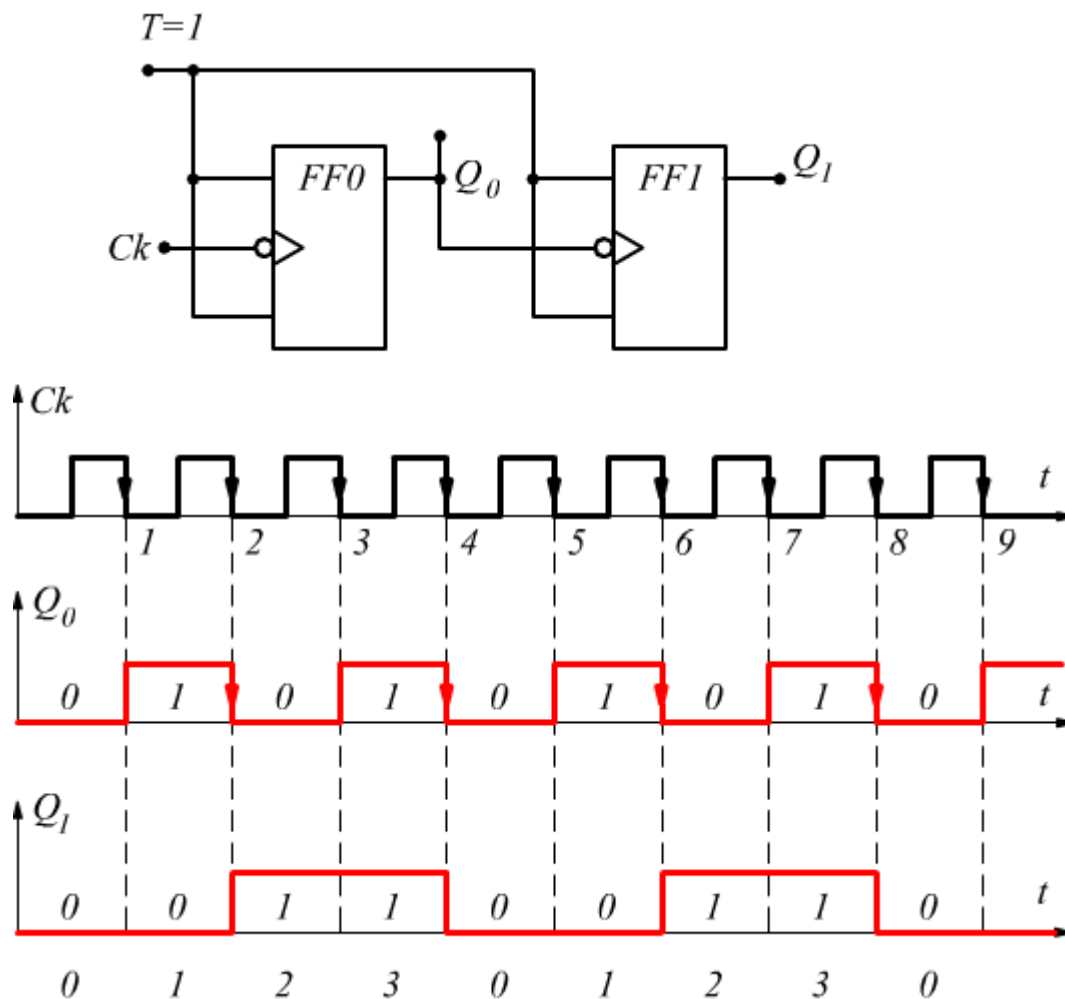
Il numero dei flip flop necessari per realizzare il contatore dipende dalla quantità degli impulsi da contare prima che esso ritorni allo stato iniziale, essendo il funzionamento ciclico. Se si volesse ad esempio realizzare un contatore modulo trentadue capace cioè di contare 31 impulsi per poi ritornare allo stato iniziale con il trentaduesimo, debbono essere utilizzati cinque flip flop in grado di fornire $2^5 = 32$ diverse configurazioni di uscita (ogni flip flop può assumere due stati diversi) ciascuno dei quali può essere associata ad uno stato logico del contatore.

In definitiva se K sono gli impulsi da contare, compreso quello di riporto allo stato iniziale, si ha che.

$$2^n = K$$

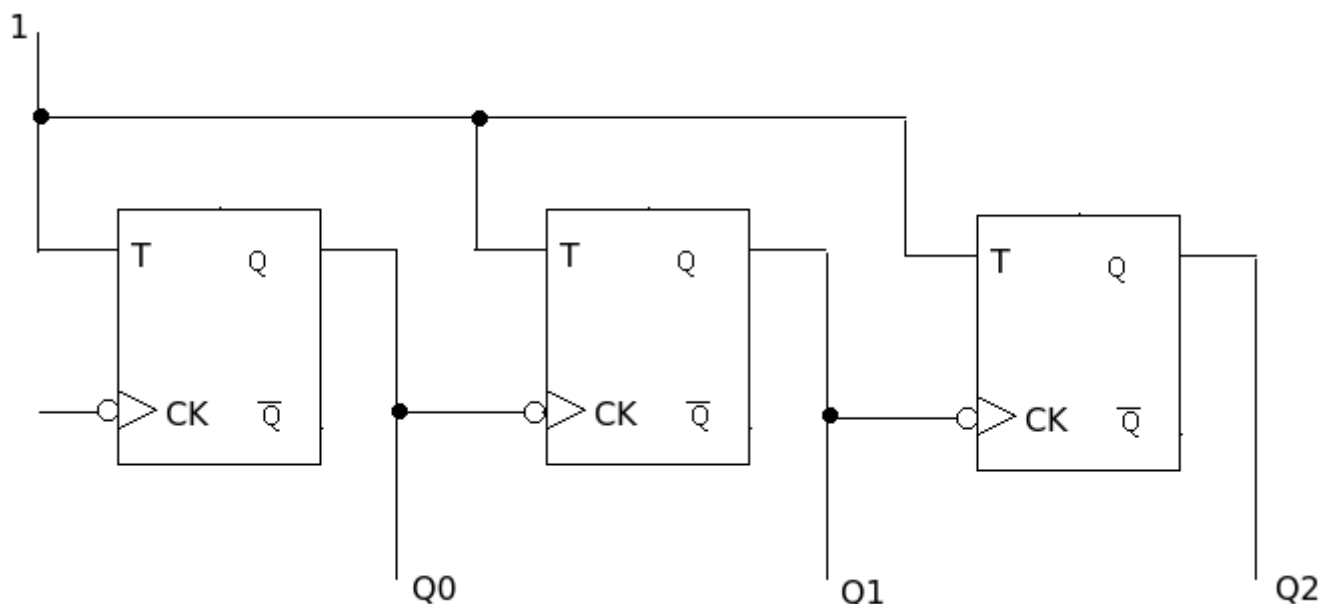
con n numero dei flip flop necessari per poter effettuare il conteggio.

Supponiamo di voler costruire un contatore binario a modulo 4; si avrebbe $2^n = 4 \rightarrow n = 2$ ci vogliono due flip-flop.

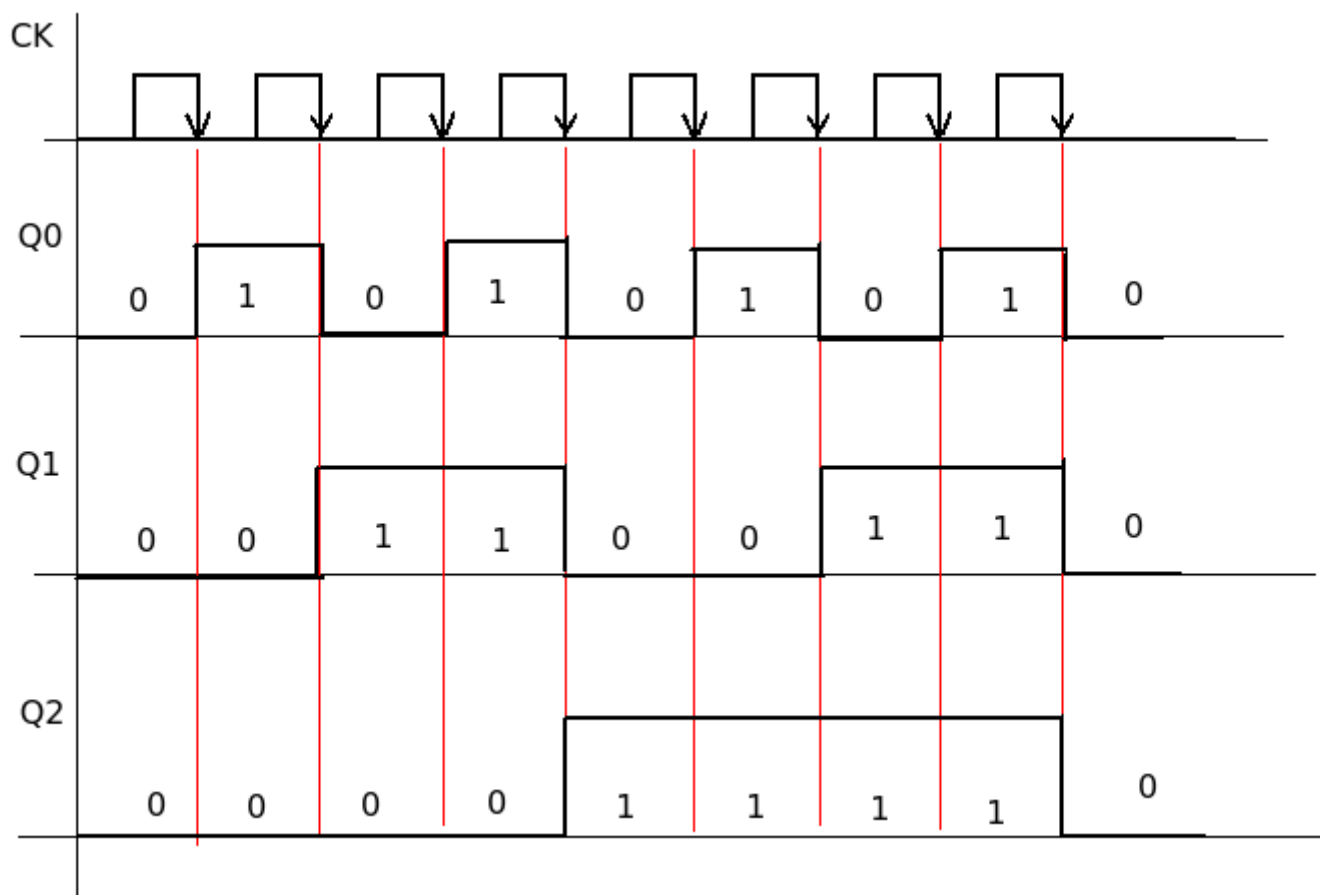


I due flip flop usati, sono sufficienti per ottenere le quattro combinazioni diverse che rappresentano i quattro stati del contatore. Poiché le quattro combinazioni 00 01 10 11 sono la rappresentazione dei primi quattro numeri del sistema binario le uscite dei flip flop sono anche le uscite del contatore .
 Per il funzionamento del dispositivo gli ingressi J e K debbono essere collegati allo stato logico 1, infatti soltanto in questo stato le uscite dei bistabili commutano ad ogni impulso inviato all'ingresso di clock ($J=K=1=toggle$).
 Il contatore grazie agli ingressi asincroni di clear può essere forzato a partire dalla condizione $Q_1=Q_2=0$ (azzeramento del contatore).
 Esaminando il disegno si nota che l'uscita Q_1 del primo flip flop si comporta da clock per il secondo. Qui sotto riportiamo la sequenza di conteggio.

Consideriamo lo schema seguente, realizzato con tre flip flop T (si rammenta qui che il FF T non è disponibile come integrato autonomo, ma può essere realizzato facilmente a partire da un ff JK):



Si osservi che l'ingresso T di tutti i FF è a 1: dunque tutti i FF si trovano in modalità commutazione (*toggle*). Si noti inoltre che il primo FF riceve il clock, mentre i FF successivi usano come clock l'uscita Q del FF precedente. Infine tutti i FF sono pilotati sul fronte di discesa del clock. Il diagramma temporale è il seguente:



Osserviamo che Q0 è un'onda quadra la cui frequenza è la metà (il periodo è il doppio) della frequenza del clock. Analogamente Q1 ha una frequenza pari a un quarto e Q2 a un ottavo della frequenza del clock. Per questa ragione il circuito è anche detto **divisore di frequenza**.

E' interessante osservare la sequenza dei valori binari prodotti sulle uscite Q2, Q1 e Q0. Considerando Q2 come bit più significativo (MSB, *Most Significant Bit*) e Q0 come bit meno significativo (LSB, *Least Significant Bit*), vediamo che in uscita vengono prodotti tutti i valori da 000 a 111 (*sette* in base dieci). Si tratta dunque di un contatore a otto stati (cioè un **contatore modulo 8**).

Aggiungendo ulteriori flip flop collegati in cascata allo stesso modo, è possibile aumentare i bit e dunque il modulo del contatore (per esempio con 5 flip flop si può realizzare un contatore modulo $2^5=32$).

Bibliografia:

www.edutecnica.it