

Elementi di Elettronica Digitale

Premessa

Le principali parti elettroniche dei computer sono costituite da circuiti digitali che, come è noto, elaborano segnali logici basati sullo 0 e sull'1.

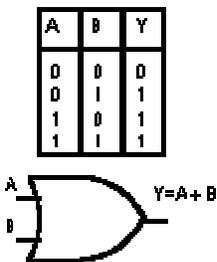
I mattoni fondamentali dei circuiti logici sono le porte logiche in grado di soddisfare un'algebra particolare, detta *algebra binaria*, sviluppata dallo scienziato George Boole (1815-1864).

In questo articolo si introdurranno le porte logiche fondamentali in grado di svolgere le operazioni di somma, prodotto e negazione indicate, rispettivamente, con OR (somma logica), AND (prodotto logico) e NOT (negazione o complementazione).

Si descriveranno, inoltre, le porte logiche derivate NOR (OR seguito da un NOT), NAND (AND seguito da un NOT), XOR (noto come OR esclusivo o circuito di anticoincidenza), XNOR (noto come NOR esclusivo o circuito di coincidenza).

Ogni porta logica ha una o più variabili di ingresso ed una sola variabile di uscita. Le variabili di ingresso e di uscita sono di tipo digitale per cui è possibile inserire in una tabella tutte le possibili combinazioni che si possono verificare tra le variabili di ingresso. L'uscita assume il valore 0 oppure il valore 1 in corrispondenza di ciascuna combinazione delle variabili di ingresso in funzione della definizione assegnata.

1. Somma logica OR

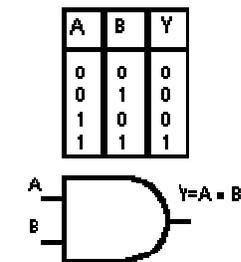


Si effettua su due o più variabili, l'uscita assume lo stato logico 1 se almeno una variabile di ingresso è allo stato logico 1.

Nel caso di due variabili di ingresso A e B, detta Y la variabile di uscita, si scrive: $Y=A+B$ e si legge A or B.

Nella seguente figura si mostra la tabella della verità con le quattro possibili combinazioni tra A e B ed il simbolo logico relativo ad una porta OR a due ingressi. Nella colonna Y si sono posti i valori assunti dall'uscita Y che soddisfa la definizione della porta OR.

2. Prodotto logico AND



Si effettua su due o più variabili, l'uscita assume lo stato logico 1 solo se tutte variabile di ingresso sono allo stato logico 1.

Nel caso di due variabili di ingresso A e B, detta Y la variabile di uscita, si scrive: $Y=A \cdot B$ e si legge A and B.

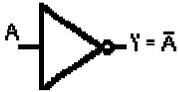
Nella seguente figura si mostra la tabella della verità con le quattro possibili combinazioni tra A e B ed il simbolo logico relativo ad una porta AND a due ingressi. Nella colonna Y si sono posti i valori assunti dall'uscita Y che soddisfa la definizione della porta AND.

3. Negazione

A	Y
0	1
1	0

Si effettua su una sola variabile. L'uscita assume il valore logico opposto a quello applicato in ingresso.

Detta A la variabile di ingresso la negazione si scrive: $Y = \bar{A}$ e si legge A negato oppure A complementato.



Nella seguente figura si mostra la tabella della verità con le due possibili combinazioni di A ed il simbolo logico relativo ad una porta NOT. Nella colonna Y si sono posti i valori assunti dall'uscita Y che soddisfa la definizione della porta NOT.

4. Somma logica negata NOR

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

Si effettua su due o più variabili, l'uscita assume lo stato logico 0 se almeno una variabile di ingresso è allo stato logico 1. In tutti gli altri casi $Y=1$.

Per due variabili di ingresso A e B: $Y = \overline{A+B}$ e si legge A nor B.



Nella seguente figura si mostra la tabella della verità con le quattro possibili combinazioni tra A e B ed il simbolo logico relativo ad una porta NOR a due ingressi. Nella colonna Y si sono posti i valori assunti dall'uscita Y che soddisfa la definizione della porta NOR.

5. Prodotto logico negato NAND

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

Si effettua su due o più variabili, l'uscita assume lo stato logico 0 se tutte le variabili di ingresso sono allo stato logico 1. In tutti gli altri casi $Y=1$.

Per due variabili di ingresso A e B: $Y = \overline{A \cdot B}$ e si legge A nand B.



Nella seguente figura si mostra la tabella della verità con le quattro possibili combinazioni tra A e B ed il simbolo logico relativo ad una porta NAND a due ingressi. Nella colonna Y si sono posti i valori assunti dall'uscita Y che soddisfa la definizione della porta NAND.

6. OR esclusivo XOR

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

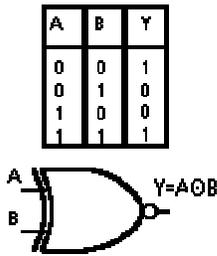
A differenza delle precedenti porte logiche, l'XOR opera su due soli ingressi. L'uscita vale 1 se gli ingressi assumono valore diverso, vale 0 se gli ingressi sono tra loro uguali.

La funzione logica si scrive: $Y = A \oplus B$ e si legge A or esclusivo B oppure A diverso da B.



Nella seguente figura si mostra la tabella della verità con le quattro possibili combinazioni tra A e B ed il simbolo logico relativo ad una porta XOR. Nella colonna Y si sono posti i valori assunti dall'uscita Y che soddisfa la definizione della porta XOR.

7. NOR esclusivo XNOR



Anche la porta XNOR opera su due soli ingressi.

L'uscita vale 1 se gli ingressi assumono valore uguale, vale 0 se gli ingressi sono tra loro diversi.

La funzione logica si scrive: $Y = A \otimes B = \overline{A \oplus B}$ e si legge A nor esclusivo B oppure A coincidente con B.

Nella seguente figura si mostra la tabella della verità con le quattro possibili combinazioni tra A e B ed il simbolo logico relativo ad una porta XNOR. Nella colonna Y si sono posti i valori assunti dall'uscita Y che soddisfa la definizione della porta XNOR.

8. NAND e NOR come operatori completi

Le porte logiche precedentemente esaminate si dividono in porte logiche fondamentali e porte logiche derivate.

Quelle fondamentali sono: somma logica OR, prodotto logico AND e negazione NOT.

Le porte logiche derivate sono: la somma logica negata NOR, il prodotto logico negato NAND, l'OR esclusivo XOR o circuito di anticoincidenza e il NOR esclusivo XNOR o circuito di coincidenza.

Le porte logiche fondamentali costituiscono i mattoni dell'elettronica digitale.

Con esse è possibile realizzare qualsiasi circuito digitale dal più semplice al più complesso.

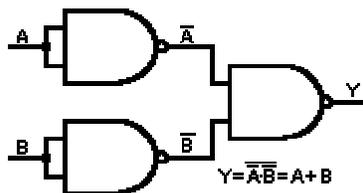
È, inoltre, possibile verificare che anche con le sole porte NOR o con le sole porte NAND è possibile realizzare qualsiasi circuito digitale. È sufficiente, infatti, verificare che con le sole porte NAND, ad esempio, si può realizzare il NOT, l'OR e l'AND.

Infatti collegando tra loro gli ingressi A e B della porta NAND come in figura si ottiene il NOT:



Per ottenere l'AND è sufficiente far seguire la porta NAND da un NOT.

Per ottenere l'OR, infine è sufficiente far precedere ciascun ingresso del NAND da un NOT come in fig.2. Infatti, per il noto teorema di De Morgan, la somma logica è il complemento del prodotto dei complementi.



9. Progettazione di un semplice circuito digitale combinatorio.

Si vuole realizzare un circuito digitale combinatorio con tre ingressi A B C ed una uscita Y.

L'uscita deve valere 1 se almeno due ingressi sono uguali ad 1. Se, invece, vale 1 un solo ingresso o nessuno di essi l'uscita Y deve valere 0 (problema della maggioranza).

Risoluzione

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

Compiliamo una tabella, nota come tabella della verità, in cui in ciascuna riga applichiamo le possibili combinazioni binarie tra gli ingressi A B C.

Tali combinazioni sono: $2^3 = 8$.

In corrispondenza di ciascuna di esse sarà possibile, grazie alla formulazione del problema, assegnare all'uscita Y il valore 0 o il valore 1.

Dalla tabella osserviamo che l'uscita Y assume il valore 1 in corrispondenza delle combinazioni ABC pari a 011, 101, 110 e 111.

Nell'ultima combinazioni tutti gli ingressi sono a 1 per cui, a maggior ragione, risulta Y=1. Per le altre combinazioni l'uscita Y vale 0 come richiesto dal problema.

Come si realizza il circuito?

Sarà, innanzitutto, necessario ricavare l'espressione algebrica dalla tabella della verità.

Il procedimento è il seguente: si considerano le combinazioni delle variabili di ingresso che rendono l'uscita Y uguale a 1. La combinazione di ingresso da considerare conterrà il prodotto tra A B C e ciascuna variabile sarà considerata in forma naturale o in forma negata a seconda se assume il valore 0 o 1. La prima combinazione che rende l'uscita uguale a 1 è: ABC=011, cioè A=0, B=1 e C=1. Allora A dovrà essere considerata in forma negata (complementata), mentre B e C in forma naturale (cioè senza negazione). I quattro termini che rendono la variabile di uscita Y uguale a 1 devono, poi, essere sommati (somma logica !!) tra di loro.

In definitiva si ottiene:

$$Y = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

Questa espressione si può semplificare algebricamente applicando le regole dell'algebra di Boole che solo parzialmente coincidono con quelle dell'algebra ordinaria.

In particolare, per la semplificazione che si intende effettuare, si applica la regola che aggiungendo quante volte si vuole un termine già esistente in una espressione, il valore dell'espressione non cambia ($A+A=A$). L'altra regola è che la somma logica tra una variabile e la stessa variabile negata vale 1 ($A + \overline{A} = 1$).

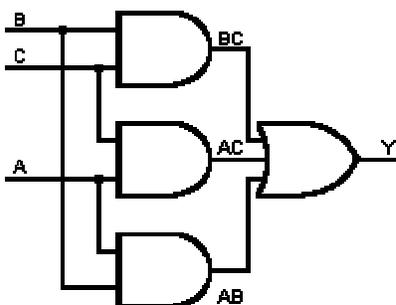
Nel nostro caso aggiungiamo, nella espressione della funzione Y, il termine ABC per altre due volte. In tal caso il primo termine ha BC in comune con ABC, il secondo ha AC in comune con l'altro ABC aggiunto ed infine il terzo termine ha AB in comune con l'ultimo ABC aggiunto.

Attraverso la messa in evidenza si ottiene:

$$Y = BC(A + \overline{A}) + AC(B + \overline{B}) + AB(C + \overline{C})$$

ove: $A + \overline{A} = 1$ e così pure $B + \overline{B} = 1$ e $C + \overline{C} = 1$; l'espressione si riduce a:

$$Y = BC + AC + AB$$



L'espressione così minimizzata è una somma di prodotti. Il circuito logico da realizzare sarà costituito da tre porte AND a due ingressi e da una porta OR finale a tre ingressi. In figura si mostra il circuito risolutivo.

Per verificarne il funzionamento è sufficiente attribuire alle variabili di ingresso A B C ciascuna delle otto combinazioni mostrate nella precedente tabella ed eseguendo la somma logica tra i tre termini prodotto BC, AC e AB.

10. Multiplexer

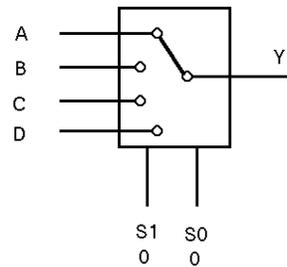
Sono dispositivi in grado di trasferire in uscita una sola di 2^n possibili linee di ingresso.

La selezione della linea di ingresso da trasferire in uscita dipende da n linee di controllo.

Il multiplexer è assimilabile ad un commutatore a posizioni multiple; la posizione dipende dai bit di controllo.

Si mostra la tabella della verità e il simbolo equivalente di un multiplexer a 4 ingressi A B C D e 2 linee di controllo (o selezione) S1 e S0.

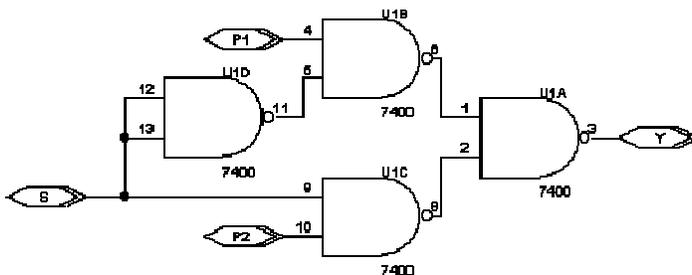
Ingressi di selezione		uscita
S ₁	S ₀	Y
0	0	A
0	1	B
1	0	C
1	1	D



La funzione di uscita Y del multiplexer vale:

$$Y = A \cdot \bar{S}_1 \cdot \bar{S}_0 + B \cdot \bar{S}_1 \cdot S_0 + C \cdot S_1 \cdot \bar{S}_0 + D \cdot S_1 \cdot S_0$$

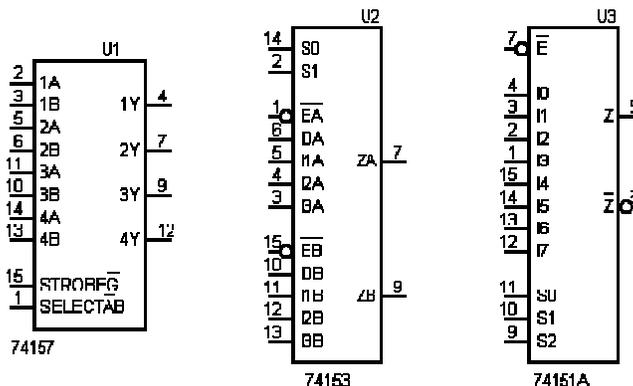
Nel caso di un multiplexer a due ingressi P1 e P2 ed un ingresso di selezione S, l'uscita Y assume la seguente espressione, trasformata applicando il teorema di De Morgan che trasforma una somma in prodotto e viceversa purché si complementi ciascun termine e l'espressione finale:



$$Y = P_1 \cdot \bar{S} + P_2 \cdot S = \overline{\overline{P_1 \cdot \bar{S}} \cdot \overline{P_2 \cdot S}}$$

Si mostra il circuito logico risolutivo, realizzato a porte NAND.

In commercio sono disponibili circuiti integrati che contengono multiplexer a 2 ingressi, 4 ingressi, 8 ingressi e 16 ingressi. Il multiplexer 74157 contiene 4 MUX a 2 ingressi, il 74153 contiene 2 MUX a 4 ingressi, il 74151 contiene 1 MUX a 8 ingressi. Si riporta la piedinatura dei citati multiplexer.



11. Flip - Flop

11.1. Premessa

I flip flop sono circuiti digitali sequenziali che hanno il compito di memorizzare un bit. 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.

I circuiti digitali si dividono in due fondamentali categorie:

- combinatori (il valore dell'uscita dipende solo dal valore dei bit applicati in ingresso);
- sequenziali (il valore dell'uscita dipende anche dal suo stato precedente).

I flip-flop 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);
- monostabili (un solo stato stabile - ad esempio i temporizzatori);
- bistabili (due possibili stati stabili - ad esempio una cella di memoria).

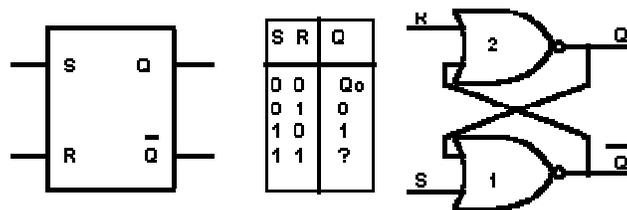
11.2. Flip-flop SR (Set-Reset)

Il più semplice dispositivo di memoria è il flip-flop Set-Reset. Esso possiede due ingressi denominati Set e Reset ed una uscita indicata con Q.

I circuiti digitali che realizzano il flip flop sono dotati, spesso, anche dell'uscita \bar{Q} . Occorre precisare, inoltre, che in un dispositivo di memoria, l'uscita dipende non solo dalla particolare combinazione assunta dalle variabili di ingresso ma anche dallo stato precedente assunto dall'uscita Q. Tale stato precedente verrà indicato con Q_0 .

11.2.1. Flip-flop SR con porte NOR

Alla luce di quanto detto si mostra in figura il simbolo logico, la tabella della verità e la soluzione circuitale a porte logiche NOR di un flip-flop S-R.



Combinazione SR=00.

Essa è nota come combinazione di riposo poiché l'uscita conserva lo stato precedente ($Q=Q_0$).

Combinazione SR=01.

Ponendo R=1, l'uscita Q si porta a 0 indipendentemente dallo stato precedente.

Combinazione SR=10.

Ponendo S=1, l'uscita Q si porta a 1 indipendentemente dallo stato precedente.

Combinazione SR=11.

Tale combinazione va evitata poiché da un punto di vista logico è una incongruenza: infatti non ha senso comandare il flip-flop per memorizzare lo 0 (R=1) oppure l'1 (S=1).

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: $Q=Q_0=0$.

Verifichiamo, infine, che il circuito realizzato con le porte NOR in figura 1 si comporta da flip-flop S-R. Dobbiamo ricordare, a tal fine, la tabella della verità della porta NOR (somma logica negata). Ponendo $S=0$ e $R=1$ si deve verificare che $Q=0$. Infatti l'uscita Q della porta 2 va a 0 poiché l'ingresso $R=1$.

La porta 1 risulta pilotata con gli ingressi uguali a 0 per cui la sua uscita va a 1. Si noti che le due uscite sono complementari tra loro.

Torniamo nella combinazione di riposo portando $R=0$ e lasciando $S=0$.

Poiché l'uscita Q precedentemente era stata portata a 0 ($Q_0=0$), l'uscita della porta 1 rimane a 1 e di conseguenza l'uscita della porta 2 rimane a 0 ($Q=0$).

Poniamo, ora: $S=1$ e $R=0$. L'uscita della porta 1 va a 0 e di conseguenza, poiché $R=0$, l'uscita della porta 2 si porta a 1: $Q=1$.

Torniamo nella combinazione di riposo portando $S=0$ e lasciando $R=0$.

Poiché l'uscita Q precedentemente era stata portata a 1 ($Q_0=1$), l'uscita della porta 1 rimane a 0 e di conseguenza l'uscita della porta 2 rimane a 1 ($Q=1$).

Se, infine applichiamo $S=1$ e $R=1$, le uscite di entrambe le porte andranno a 0. Conseguenze: le due uscite, in questo caso, non sono più l'una il complemento dell'altra, ed inoltre portando contemporaneamente S ed R a 0 entrambe le uscite si porteranno ad 1 e poi a 0 e così via. In realtà, a causa dei diversi tempi di ritardo di propagazione del segnale elettrico in ciascuna porta, uno dei due NOR propagherà l'1 in uscita prima dell'altra porta. In conclusione diventa aleatorio il valore dell'uscita Q che, pertanto, potrà trovarsi o a 0 o a 1. Anche per questo motivo è bene evitare l'applicazione dell'ultima combinazione della tabella della verità: $S=1$ e $R=1$.

11.2.2. Flip-flop a porte NAND

Si riporta in figura lo schema logico del flip-flop realizzato a porte NAND.

Qui gli ingressi sono indicati con le lettere A e B che, sostanzialmente, si comportano come gli ingressi S ed R del precedente flip-flop a porte NOR con la differenza che gli ingressi sono attivi in logica negativa. In altre parole i due flip-flop coincidono nella funzione logica purché si ponga:

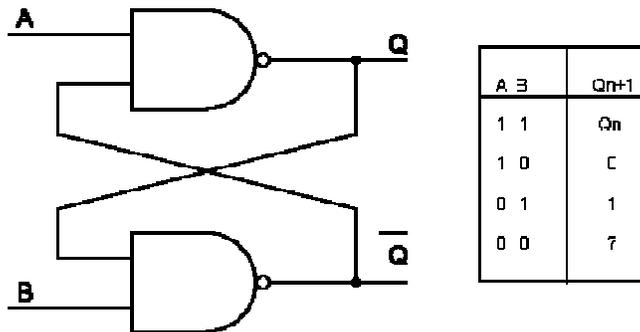
$$A = \bar{S}; \quad B = \bar{R}$$

Ponendo $AB=10$ si realizza la funzione di reset per cui l'uscita Q si porta a 0.

Ponendo $AB=01$ si realizza la funzione di set per cui l'uscita Q si porta a 1.

Ponendo $AB=11$ si realizza la funzione di memoria per cui l'uscita conserva il precedente valore memorizzato.

$AB=00$ è la combinazione da evitare sia per incongruenza logica sia perché porterebbe entrambe le uscite al valore 1 e quindi non sarebbero una il complemento dell'altra.



11.2.3. 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 in precedenza, 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:

PET quando il clock passa da 0 a 1;

NET quando il clock passa da 1 a 0.

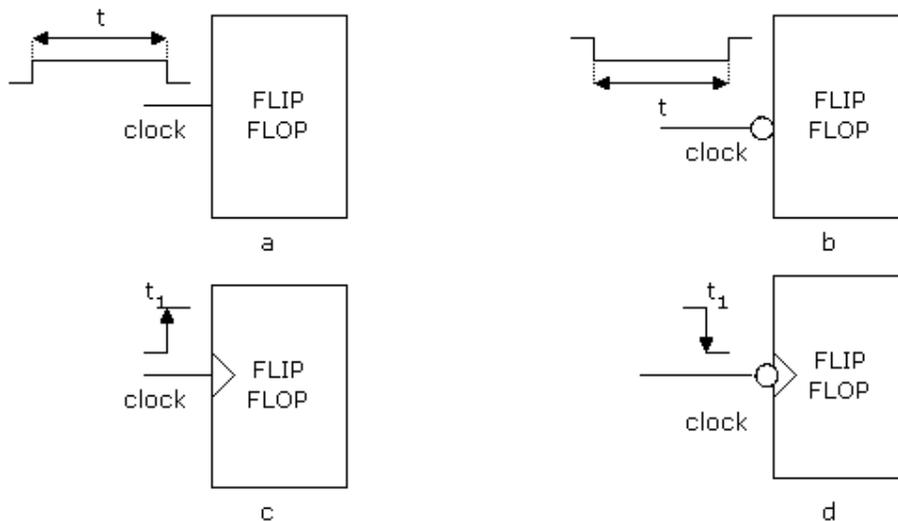


Fig.3 - Flip-flop sensibili ai livelli ed alle transizioni.

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 .

11.2.4. Flip-flop S-R sincronizzato

In fig.4 si mostra lo schema logico del flip-flop Set Reset sincronizzato. Esso è costituito da due porte logiche NAND, dette porte pilota, e da altre due porte NAND che realizzano il flip-flop S R vero e proprio del tipo mostrato nella precedente fig.2.

Se il clock CK è al livello logico 1 le porte pilota si comportano da NOT e quindi gli ingressi S ed R sono effettivamente coincidenti con gli omonimi ingressi del generico flip-flop Set Reset.

Se, invece, il clock CK è al livello logico basso, le uscite delle due porte pilota sono al livello logico 1 indipendentemente dai valori applicati agli ingressi S e R.

Per tale combinazione la "latch" a porte NAND conserva lo stato precedente e quindi il flip flop è insensibile ai comandi esterni applicati.

Nella tabella della verità si riporta l'uscita futura Q_{n+1} in funzione del clock, degli ingressi S ed R e dallo stato presente Q_n . X rappresenta indifferentemente sia lo stato logico 0 che lo stato logico 1.

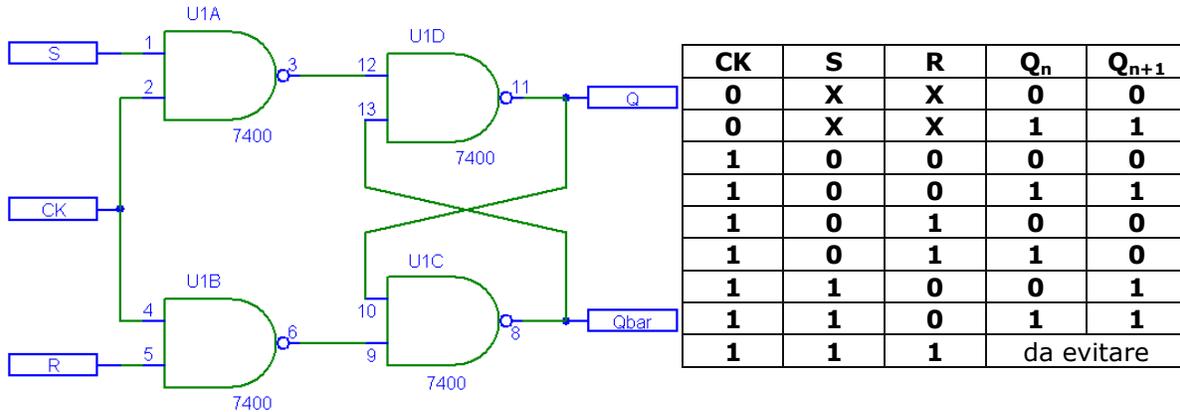


Fig.4. - Flip-flop SR sincronizzato con porte NAND. U1A e U1B costituiscono le porte pilota. Le porte U1C e U1D rappresentano il flip-flop vero e proprio. Con Qbar si indica l'uscita Q (Q barrata).

11.2.5. Flip-flop JK

Il flip-flop JK è un dispositivo a due entrate denominate J e K che operano in modo analogo alle entrate S ed R di un flip-flop Set Reset con la differenza che se $J=1$ e $K=1$ l'uscita commuta, cioè se lo stato presente è 0, lo stato futuro è 1 e viceversa.

Per evitare commutazioni multiple, nel caso si lasci a lungo la combinazione $J=1$ e $K=1$, tale flip-flop deve essere necessariamente sincronizzato.

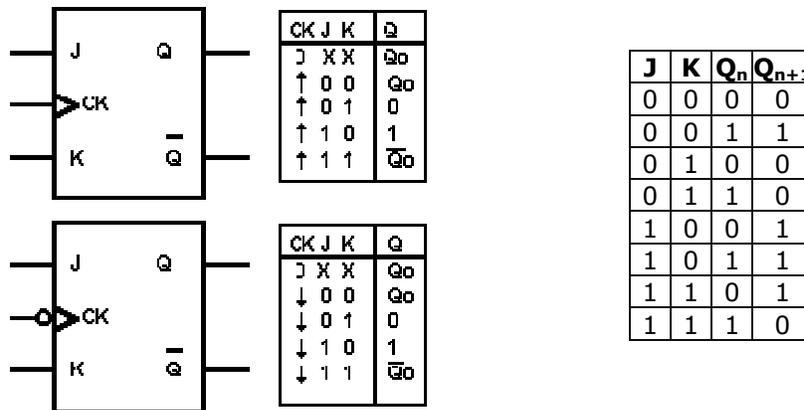


Fig.5. - Flip-flop JK e tabelle della verità. A destra si mostra la tabella della verità che prescinde dalla presenza del clock e che mostra l'uscita futura in presenza degli ingressi esterni J e K e dallo stato presente Q_n .

In fig.5 si mostrano due flip-flop JK sincronizzati. Il primo adegua le uscite quando al clock si applica un segnale digitale che passa da 0 a 1 (transizione positiva indicata con una freccia verso l'alto nella tabella della verità). Il secondo adegua le uscite quando al clock si applica un segnale

digitale che passa da 1 a 0 (transizione negativa indicata con una freccia verso il basso nella tabella della verità).

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

I flip-flop JK si realizzano con una soluzione circuitale costituita da porte logiche secondo uno schema interno più complesso rispetto a quello visto per il flip-flop SR realizzato a porte NOR e a porte NAND.

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).

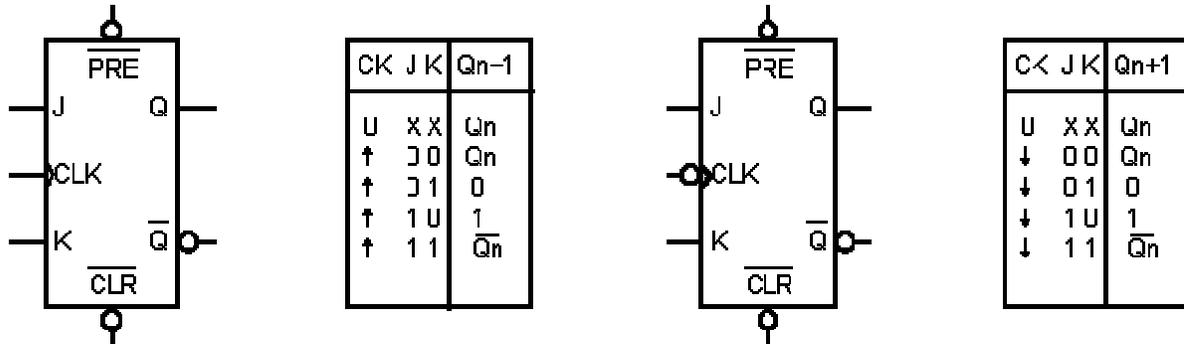


Fig.6 - Simbolo del flip-flop JK PET e relativa tabella della verità (a sinistra); Simbolo del flip-flop JK NET e relativa tabella della verità (a destra).

Si noti il simbolo del clock nei due casi. 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.

11.2.6. Flip-flop D

Presenta un solo ingresso denominato D. Spesso è di tipo sincronizzato e funziona nel seguente modo: quando è attivo l'ingresso di clock l'uscita assume lo stesso valore applicato all'ingresso D. In assenza di comando di clock l'uscita conserva lo stato precedente indipendentemente dal valore applicato all'ingresso D.

In fig.7 si mostra il simbolo logico e la tabella della verità del flip-flop D.

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

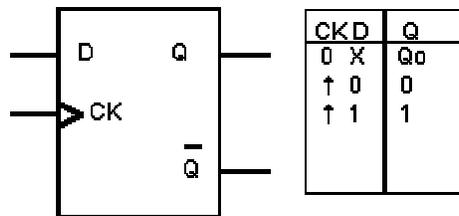


Fig.7 - Flip-flop D e relativa tabella della verità.

11.2.7. Flip-flop T

E' un flip-flop avente una sola entrata denominata T e le solite due uscite: Q e la sua complementata \bar{Q} . L'uscita Q conserva lo stato precedente se $T=0$ e commuta se $T=1$ quando si applica il segnale di clock. I flip-flop T, al contrario dei JK, D e SR, non sono disponibili commercialmente in forma integrata poiché si possono ottenere direttamente dai flip-flop JK semplicemente collegando tra loro gli ingressi J e K. E' utilizzato come dispositivo divisore di frequenza di clock per due e nei contatori digitali. Si mostra in fig.8 il collegamento di un flip-flop NET di tipo JK per ottenere un flip-flop di tipo T e la relativa tabella della verità.

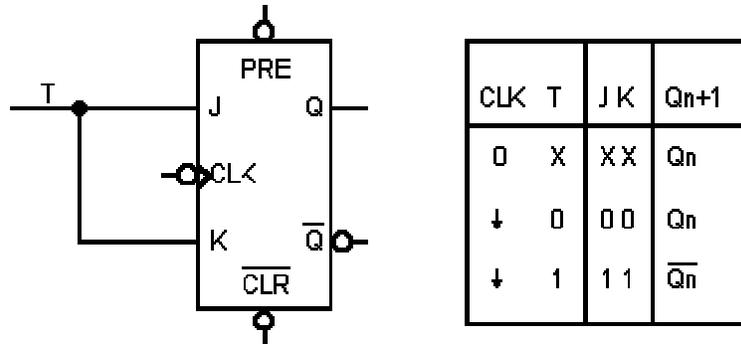


Fig.8 - Simbolo del flip-flop T e relativa tabella della verità. Dalla tabella si nota che solo quando il clock passa dal livello logico alto al livello logico basso si ha:

$$Q_{n+1} = Q_n \text{ se } T=0 \text{ e } Q_{n+1} = \bar{Q}_n \text{ se } T=1.$$

12. Diagramma degli stati

Le possibili modalità di funzionamento di un flip-flop e, più in generale, di qualsiasi circuito digitale con uno o più flip-flop, possono essere descritte in diverse modalità: tabella della verità, equazione caratteristica, tabella delle transizioni, diagramma degli stati. Le varie modalità sono tra loro perfettamente equivalenti per cui da ciascuna rappresentazione è possibile ricavare qualsiasi altra. In questa sede si vuole ricavare il diagramma degli stati del flip-flop J-K.

Il procedimento per ricavare il diagramma degli stati degli altri flip-flop è del tutto equivalente.

Intanto diciamo subito che per diagramma degli stati si intende una rappresentazione grafica dei vari stati o valori logici che il sistema digitale sequenziale può assumere. Nel caso del singolo flip-flop, che possiede una sola uscita Q, i possibili stati sono due: 0 e 1.

I due valori si inseriscono in altrettanti cerchietti ognuno dei quali rappresenta il punto di arrivo o il punto di partenza di un arco orientato che rappresenta la transizione dello stato interno del circuito. L'arco orientato è condizionato dagli ingressi esterni che, per il flip-flop JK, sono gli ingressi J e K.

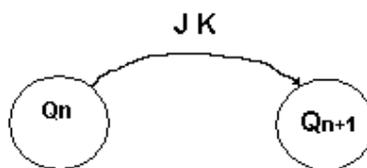


Fig.9. - L'arco orientato individua il passaggio di stato condizionato dagli ingressi J e K

In fig.10 si mostra il diagramma degli stati del flip-flop JK. Per poterlo costruire si è tenuta presente la tabella della verità rappresentata nella parte destra della fig.5. Per ogni combinazione si tiene conto dei valori degli ingressi J e K e di quelli dello stato presente Q_n e futuro Q_{n+1} .

Per la prima combinazione, ad esempio, si ha: $J = K = 0$ e $Q_n = Q_{n+1} = 0$.

Si deve disegnare, pertanto, un arco orientato che parte dallo stato 0 (stato presente Q_n) e termina nello stesso stato 0 (stato futuro Q_{n+1}).

Si procede con lo stesso ragionamento per le altre combinazioni.

Per l'ultima combinazione si ha: $J = K = 1$, $Q_n = 1$ e $Q_{n+1} = 0$.

L'arco orientato parte dallo stato 1 e termina nello stato 0. Sull'arco orientato si scrivono i due valori degli ingressi J e K: 11.

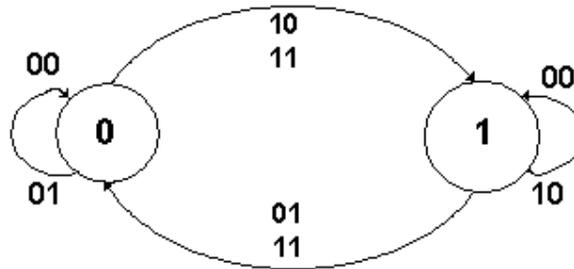


Fig.10. - Diagramma degli stati del flip-flop JK.

13. Tabella delle transizioni

Si indica col nome di tabella delle transizioni la mappa di Karnaugh in cui si inseriscono i valori che assume lo stato futuro Q_{n+1} dell'uscita in funzione degli ingressi e dello stato presente Q_n . Nel caso del flip-flop JK gli ingressi da considerare sono J e K.

Si mostra, in fig.11, la tabella delle transizioni del flip-flop JK. Essa si ricava dalla tabella della verità di fig.5 o, indifferentemente, dal diagramma degli stati della precedente fig.10.

		JK			
		00	01	11	10
Q_n	0	0	0	1	1
	1	1	0	0	1

Fig.11. - Tabella delle transizioni per un flip-flop JK.

Le celle in cui compare un bit coincidente col valore di Q_n (celle con sfondo giallo) rappresentano uno stato stabile poiché un impulso di clock non modifica lo stato di uscita Q.

Sono stabili i due stati della prima colonna (JK=00), lo stato superiore della seconda colonna e lo stato inferiore della quarta colonna. Gli altri sono stati instabili.

La funzione minimizzata che si ricava dalla precedente tabella delle transizioni prende il nome di equazione caratteristica e vale:

$$Q_{n+1} = J\bar{Q}_n + \bar{K}Q_n$$

Essa si ricava raggruppando le due celle adiacenti della prima riga in cui compare 1 e le due celle della seconda riga poste in prima e quarta colonna.

14. Applicazioni dei flip-flop

I flip-flop trovano applicazione in tutti i circuiti digitali sequenziali, cioè in quei circuiti in cui le uscite, oltre a dipendere dagli ingressi esterni, dipendono anche dallo stato interno assunto in precedenza. In altre parole trova applicazione in tutti i dispositivi di memoria: contatori, centralina dei cancelli automatici, ascensori, lavatrici, antifurto, generatori di sequenze binarie e, in particolare, in tutte le applicazioni di automazione industriale.

14.1. Comandi separati di marcia e di arresto

A titolo d'esempio si mostra in fig.12 un flip-flop SR con ingressi in logica negativa realizzato con due porte NAND comandato da due pulsanti, normalmente aperti, indicati con PM e PA. Il circuito consente i comandi separati per la marcia e l'arresto di apparati di potenza.

Quando i pulsanti sono nello stato di riposo, gli ingressi del flip-flop sono al livello logico alto: l'uscita conserva lo stato precedente.

Per mettere in marcia il circuito di potenza, ovvero porre $Q=1$, si deve pigiare il pulsante di marcia PM.

Ciò provoca il collegamento a massa del pin 1 e quindi l'uscita Q si porta al livello logico alto: il circuito di potenza viene attivato.

Rilasciando il pulsante l'uscita conserva lo stato $Q=1$ e l'impianto rimane in marcia.

Per arrestare l'impianto si deve pigiare il pulsante di arresto PA. Ciò provoca il collegamento a massa del piedino 5 e quindi $Q=0$.

L'impianto si arresta. Rilasciando il pulsante l'uscita conserva lo stato $Q=0$ e l'impianto rimane nello stato di arresto.

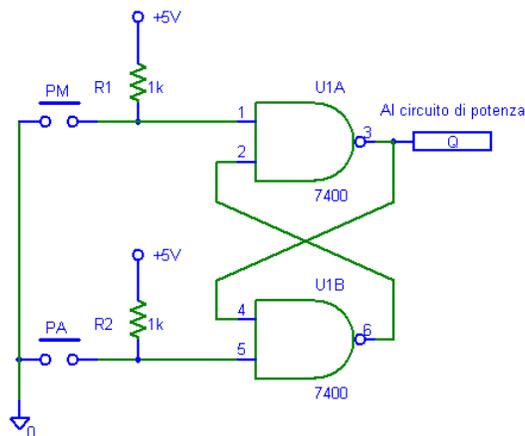


Fig.12. - Flip-flop SR per la marcia e l'arresto di apparati.

14.2. Circuito antirimbalo

Una variante del circuito precedente consente la realizzazione di un circuito in grado di sopprimere rimbaldi meccanici dovuti ai contatti mobili di interruttori. I rimbaldi multipli, infatti, generano un numero imprecisato di impulsi indesiderati che potrebbero far funzionare in modo anomalo il circuito dipendente dall'interruttore. Si pensi, ad esempio, al contatore "elimina code" dei

supermercati. Quando il salumiere preme il pulsante, il display a due cifre deve mostrare il numero successivo a quello corrente grazie alla presenza di un circuito contatore digitale. Se, però, l'interruttore non è protetto da circuito antirimbalo, alla pressione dello stesso da parte del salumiere, si generano più impulsi che verrebbero conteggiati e quindi anziché visualizzare il numero successivo, vedremmo sui display un numero sicuramente più grande. Se, ad esempio, il display mostra il numero 30 ed alla pressione dell'interruttore si generano 5 impulsi, il successivo numero visualizzato sul display sarà 35 anziché 31.

Un possibile circuito antirimbalo si realizza con lo schema logico di fig.13 che impiega un flip-flop S R a porte NAND.

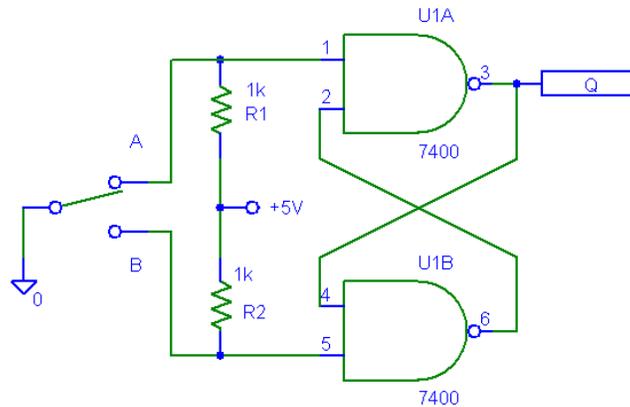


Fig.13.- Il deviatore seguito dal flip-flop prende il nome di interruttore antirimbalo.

Il deviatore può assumere due posizioni: A e B. Il contatto mobile è collegato alla massa e perciò porta il livello logico 0 all'ingresso collegato. Supponiamo che il deviatore passi da A a B e successivamente ritorni nella posizione A.

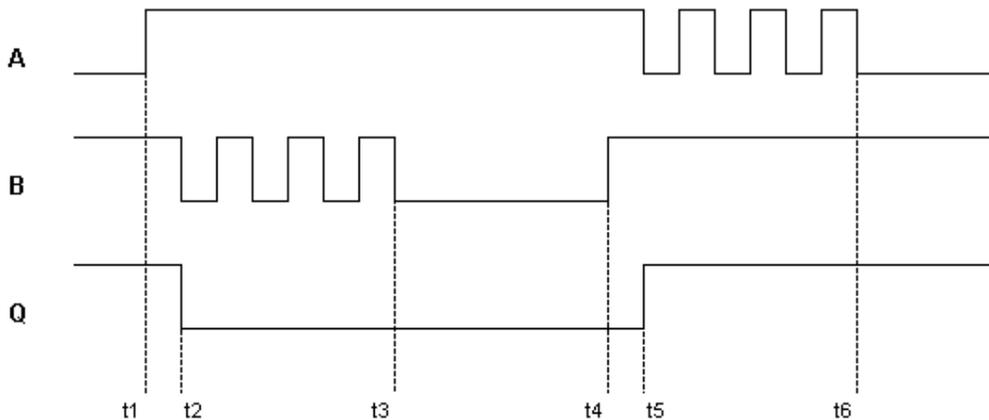


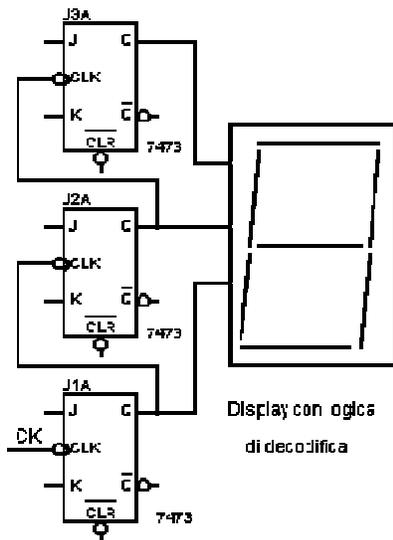
Fig.14. - Forme d'onda nei punti A, B e sull'uscita Q.

Nell'istante t1 il contatto mobile lascia il punto A e questo si porta al livello logico 1. Nell'istante t2, terminata la sua corsa, il contatto mobile tocca il punto B che si porta al livello 0. A causa della sua elasticità, la lamella vibra causando gli impulsi visibili tra gli istanti t2 e t3. In t3 la lamella tocca definitivamente il punto B. Nell'istante t4 si decide di riportare l'interruttore nella posizione A. Quando in t4 il contatto mobile lascia il punto B, questo si porta a 1 e nell'istante t5 raggiunge per la

prima volta il punto A che si porta a zero. A causa dell'elasticità della lamella del contatto mobile, si creano su A delle vibrazioni che generano gli impulsi visibili sulla linea A nell'intervallo di tempo compreso tra t_5 e t_6 . In t_6 il contatto mobile tocca definitivamente il punto A che rimane definitivamente al livello logico. Nella fig.14 si è supposto che il rimbalzo meccanico produca 3 impulsi spuri prima di raggiungere definitivamente il livello logico 0. Esaminando le forme d'onda di A e B nei vari intervalli di tempo, si deduce facilmente la forma d'onda dell'uscita Q che, come si vede nella fig.14, è esente dai rimbalzi meccanici.

14.3 Contatore a tre bit

Si mostra in fig.15 un collegamento tra 3 flip-flop JK NET che realizzano un contatore binario a tre bit capace di mostrare su un display, ciclicamente, cifre da 0 a 7 sotto l'applicazione di un fronte di discesa applicato al clock. Gli impulsi di clock CK si applicano all'ingresso CLK del flip-flop JK denominato U1A.



L'uscita Q del flip-flop U1A si applica all'ingresso CLK del flip-flop U2A e, analogamente, l'uscita Q del secondo flip-flop si applica all'ingresso CLK del flip-flop U3A. Gli ingressi J e K di tutti i flip-flop sono tenuti ad 1 per cui ad ogni fronte di discesa applicato agli ingressi CLK l'uscita Q cambia di stato. Indicando con $Q_3 Q_2 Q_1$ le uscite dei 3 flip-flop e supponendo 000 lo stato iniziale del contatore si ha, dopo l'applicazione di successivi impulsi di clock: 001, 010, 011, 100, 101, 110, 111

Applicando tali linee ad un display dotato di circuito di decodifica interno si avrà sul visualizzatore la comparsa delle cifre da 0 a 7. Dopo il 7 riapparirà nuovamente lo 0 e così via. Le linee CLR, se portate allo 0 logico, azzerano il contatore indipendentemente dal clock (comando asincrono).

Fig.15 - Contatore a 3 bit con visualizzazione su un display.

15. Decoder per il pilotaggio di display a 7 segmenti

Il display della precedente figura 15 contiene al suo interno una logica di decodifica che consente di pilotare correttamente i sette segmenti del display a LED.



Fig.16. Denominazione dei segmenti di un display e segmenti accesi per ottenere le cifre da 0 a 9.

I sette segmenti sono denominati da **a** a **g** come nella figura a sinistra. Per visualizzare i vari numeri da 0 a 9 occorre attivare alcuni dei 7 segmenti.

Per ottenere il simbolo 0 devono essere accesi i segmenti **a, b, c, d, e, f** e spento il segmento **g**.

Per ottenere il 9 il segmento **e** deve essere spento e tutti gli altri accesi.

Si noti che per ottenere il numero 8 è necessario accendere tutti i segmenti.

Associando lo stato 1 al segmento acceso e lo stato logico 0 al segmento spento, il circuito di decodifica accetta in ingresso il codice binario a 4 bit DCBA e fornisce in uscita 7 linee che pilotano i corrispondenti ingressi del display.

Il decodificatore, pertanto, deve soddisfare la seguente tabella della verità in cui con N si indica il numero decimale corrispondente alla configurazione binaria che è possibile realizzare con i quattro bit DCBA e con display si indica la sagoma fornita dal display quando è pilotata da una delle 16 combinazioni di ingresso.

Si noti che le combinazioni corrispondenti ai numeri compresi tra 10 e 15 hanno uscite indicate con x (non importa) perché non previste dal codice di ingresso.

Il progettista del decodificatore potrà a suo piacimento sostituire alla x il valore 0 o 1 per meglio ottimizzare le funzioni logiche di uscita.

Nell'ultima colonna si mostrano le configurazioni rappresentate dal display in funzione del codice binario applicato all'ingresso del decodificatore.

N.	ingressi				uscite							display
	D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	0	1	1	1	1	1	1	0	0
1	0	0	0	1	0	1	1	0	0	0	0	1
2	0	0	1	0	1	1	0	1	1	0	1	1
3	0	0	1	1	1	1	1	1	0	0	1	1
4	0	1	0	0	0	1	1	0	0	1	1	1
5	0	1	0	1	1	0	1	1	0	1	1	1
6	0	1	1	0	1	0	1	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0	1
8	1	0	0	0	1	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1	1
10	1	0	1	0	x	x	x	x	x	x	x	?
11	1	0	1	1	x	x	x	x	x	x	x	?
12	1	1	0	0	x	x	x	x	x	x	x	?
13	1	1	0	1	x	x	x	x	x	x	x	?
14	1	1	1	0	x	x	x	x	x	x	x	?
15	1	1	1	1	x	x	x	x	x	x	x	?

In commercio sono disponibili circuiti integrati decodificatori in grado di pilotare correttamente un display LED a sette segmenti.

I decodificatori integrati più utilizzati sono il CMOS 4511, il TTL 7448 ed il vecchio 9368; quest'ultimo è in grado di consentire la visualizzazione del codice esadecimale (da 0 a F) sul display e non necessita di resistenze di limitazione di corrente né di resistenze di pull-up per il pilotaggio del display.