5 *ELEMENTI DI MEMORIA*

5.1 Funzioni sequenziali

Consideriamo il circuito di fig. 5.1.1.

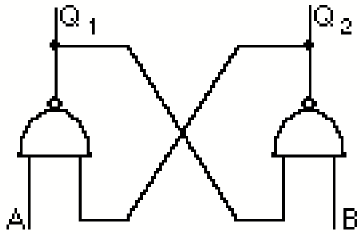

Figura 5.1.1

Costruiamone la tavola della verità, tabella 5.1.1, nel modo usuale usato per le funzioni combinatorie.

Tabella 5.1.1

A	B	Q_2	Q_1
0	0	1	1
1	0	1	0
0	1	0	1
1	1	X	$\overline{X}$

Trattandosi di due circuiti NAND, è sufficiente che un ingresso sia **0** perché l'uscita sia **1**, tuttavia nel caso che A e B siano entrambi **1** non possiamo dire quale sia lo stato di Q_1 e Q_2: sappiamo solo che uno deve essere il completamento dell'altro. In questo caso per definire lo stato delle uscite è necessario stabilire quale degli ingressi ha lasciato per ultimo lo stato **0**, livello attivo per le linee d'ingresso. Possiamo dunque dire che quando gli ingressi sono entrambi ad **1**, il circuito conserva *memoria* di ciò che è avvenuto precedentemente.

Per la descrizione dei circuiti con memoria, piuttosto che introdurre la variabile tempo in modo esplicito, conviene considerare l'asse dei tempi diviso in intervalli discreti scanditi da un orologio che fornisca, all'occorrenza, un segnale di sincronizzazione *t* (fig. 5.1.2).

Figura 5.1.2

Tutti i circuiti di memoria che considereremo avranno questa sincronizzazione che trasferirà in uscita l'effetto dei segnali d'ingresso: lo stato delle variabili d'ingresso *all'istante* di tempo n-1 definirà quindi quello dalla funzione d'uscita *nell'intervallo* n. Queste funzioni booleane, dette *sequenziali*, vengono scritte come:

$$Q^n = f(A,B,\ldots N)^{n-1} \tag{5.1.1}$$

Notiamo che gli intervalli di tempo non devono necessariamente avere la stessa durata. Un tipico caso di circuito con memoria è quello di un circuito che conti eventi esterni: in questo caso l'orologio sarà il segnale dell'evento stesso. In base a questa convenzione conviene descrivere il circuito di fig. 5.1.1 con una nuova tabella 5.1.2.

Tabella 5.1.2

A^n	B^n	Q_1^{n+1}	Q_2^{n+1}
0	0	?	?
0	1	1	0
1	0	0	1
1	1	Q_1^n	$\overline{Q}_1^n$

Si noti che il primo stato della tabella è "proibito" nel senso che se all'istante n l'orologio trasferisse questo stato in uscita non sarebbe definito lo stato di Q_1 e Q_2, nell'intervallo di tempo (n+1).

5.2 Equazioni caratteristiche

Modifichiamo il circuito di fig. 5.1.1 secondo lo schema di fig. 5.2.1a il cui simbolo logico è dato da fig. 5.2.1b.

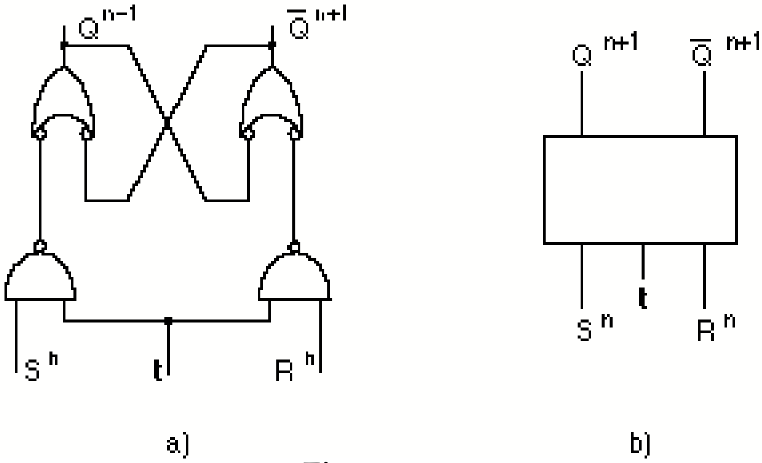

Figura 5.2.1

Si vede che il circuito è insensibile al variare di *S* o di *R* se *t* non è presente, mentre solo all'istante *t* *S* e *R* modificano *Q*. La tavola della verità di questo circuito, detto flip-flop set-reset (S-R), è la tabella 5.2.1.

Tabella 5.2.1

S^n	R^n	Q_1^{n+1}	Q_2^{n+1}
0	0	Q_1^n	$\overline{Q}_1^n$
0	1	0	1
1	0	1	0
1	1	?	?

La funzione Q^{n+1} che se ne deriva è data da:

$$Q^{n+1} = (Q\overline{S}\overline{R} + S\overline{R})^n \tag{5.2.1}$$

che insieme alla condizione:

$$SR = 0 \tag{5.2.2}$$

descrive il funzionamento del flip-flop. La (5.2.2) si può sommare a (5.2.1) in modo da ottenere:

$$Q^{n+1} = (S + Q\overline{R})^n \tag{5.2.3}$$

che, associata alla condizione (5.2.2), è detta *equazione caratteristica* del flip-flop S-R. Il S-R è il primo elemento di memoria che siamo in grado di costruire partendo da semplici circuiti NAND. Tuttavia l'indeterminazione di Q in caso di presenza simultanea, al tempo di t, di S e R rende questo flip-flop praticamente inutilizzabile. Al fine di eliminare questa indeterminazione, *inventiamo*, definendo la tavola della verità 5.2.2, un altro flip-flop che chiameremo J-K.

Notiamo che per ora non ci preoccupiamo di come possa essere realizzato circuitalmente il J-K.

Tabella 5.2.2

J^n	K^n	Q^{n+1}
0	0	Q^n
1	0	1
0	1	0
1	1	$\overline{Q}^n$

L'equazione caratteristica del J-K si ricava dalla tabella 5.2.2 e sarà:

$$Q^{n+1} = (Q\overline{J}\overline{K} + J\overline{K} + \overline{Q}JK)^n \tag{5.2.4}$$
$$= (Q\overline{K} + \overline{Q}J)^n \tag{5.2.5}$$

Figura 5.2.2

Analogamente a quanto fatto per il J-K possiamo inventare un altro tipo di flip-flop, detto D (delay), definendolo attraverso la tavola della verità 5.2.3.

Tabella 5.2.3

D^n	Q^{n+1}
0	0
1	1

Anche in questo caso non ci preoccupiamo per ora di come sia circuitalmente realizzabile il flip-flop D. Dalla tabella 5.2.3 ricaviamo l'equazione caratteristica:

$$Q^{n+1} = D^n \tag{5.2.6}$$

I simboli logici del J-K e D sono in fig. 5.2.2.

I flip-flop J-K e D sono di fatto gli elementi di memoria più largamente utilizzati usati sebbene sia possibile, in linea di principio, definire altri tipi di flip-flop, anche a più di due ingressi.

5.3 Equazioni applicative

Le *equazioni caratteristiche* viste nel paragrafo precedente descrivono il comportamento dei singoli elementi di memoria. Un generico circuito digitale di tipo sequenziale sarà realizzato con questi elementi di memoria che tuttavia dovranno essere condizionati a comportarsi in modo tale da realizzare le funzioni sequenziali volute. In forma generale un circuito di tipo sequenziale sarà realizzato come in fig. 5.3.1, ove un opportuno numero di flip-flop sarà controllato negli ingressi da funzioni logiche di tipo combinatorio.

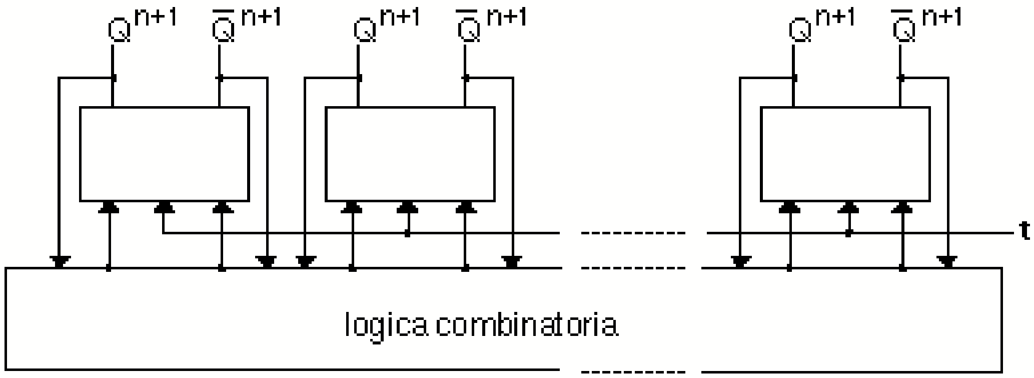

Figura 5.3.1

Un classico esempio di circuito sequenziale è costituito da un circuito che conti impulsi. Per realizzarlo sarà necessario un certo numero di flip-flop che abbiamo come segnale di orologio *t*, il segnale da contare. Se *n* sono gli elementi di memoria, 2^n saranno gli stati associabili alle loro possibili combinazioni **0**, **1**. Il contatore dovrà passare ad ogni impulso *t* di orologio dello stato di contenuto *p* (peso della combinazione di *n* zeri e uno) a quello di contenuto (*p* + 1).

In generale per realizzare un sistema logico di tipo sequenziale sarà quindi necessario scrivere, mediante le tavole della verità, le *equazioni applicative* che descrivono il comportamento del sistema e, una volta scelti gli elementi di memoria (S-R, J-K etc.), si cercheranno le funzioni che poste in ingresso a questi realizzino la sequenza voluta.

Essenzialmente si tratta di risolvere, per gli ingressi delle memorie, il *sistema booleano* costituito dalle *equazioni applicative* e dalle *equazioni caratteristiche*.

È facile verificare che qualsiasi equazione applicativa può essere messa nella forma:

$$Q^{n+1} = (Qg_1 + \overline{Q}g_2)^n \tag{5.3.1}$$

Le soluzioni del sistema fra la (5.3.1) e l'equazione caratteristica generica:

$$Q^{n+1} = f(I_1, I_2, \ldots, I_k, Q^n) \qquad (5.3.2)$$

dell'elemento utilizzato nel progetto saranno del tipo:

$$I_m = p_m(g_1, g_2, Q) \qquad (5.3.3)$$

Le soluzioni per gli ingressi per tali sistemi vanno cercate con metodi tabellari.

Troveremo, nel prossimo paragrafo, le soluzioni generali per i vari tipi di elementi di memoria fin qui considerati.

5.4 Funzioni di ingresso per gli elementi di memoria

Risolviamo anzitutto il sistema:

$$Q^{n+1} = (S + Q\overline{R})^n ; SR = 0 \tag{5.4.1}$$
$$Q^{n+1} = (Qg_1 + \overline{Q}g_2)^n . \tag{5.4.2}$$

Per trovare la soluzione generale per gli ingressi di un flip-flop S-R, scriviamo la tabella 5.4.1 che definisce Q^{n+1} in funzione di g_1, g_2, Q^n e la tabella 5.4.2 che ci dice quali devono essere S e R per essere compatibili con gli stati di Q^n e Q^{n+1} della tabella 5.4.1 e con la (5.4.1).

Tabella 5.4.1

g_1^n	g_2^n	Q^n	Q^{n+1}
0	0	0	0
1	0	0	0
0	1	0	1
1	1	0	1
0	0	1	0
1	0	1	1
0	1	1	0
1	1	1	1

Tabella 5.4.2

Q^n	Q^{n+1}	S	R
0	0	0	a_0
0	0	0	a_1
0	1	1	0
0	1	1	0
1	0	0	1
1	1	b_0	0
1	0	0	1
1	1	b_1	0

In definitiva perché siano soddisfatte contemporaneamente la (5.4.1) e (5.4.2) dovrà valere all'istante n la tabella 5.4.3 dalla quale si ricavano le soluzioni generali per gli ingressi S e R.

Notiamo che i parametri $\square$ e $\square$ possono essere indifferentemente **0** oppure **1** e pertanto potremo porli al valore che renda più semplici le funzioni S ed R. Si vede comunque che le

Tabella 5.4.3

g_1	g_2	Q	S	R
0	0	0	0	a_0
1	0	0	0	a_1
0	1	0	1	0
1	1	0	1	0
0	0	1	0	1
1	0	1	b_0	0
0	1	1	0	1
1	1	1	b_1	0

soluzioni più semplici:

$$S = g_2\overline{Q} \qquad (5.4.3)$$
$$R = \overline{g}_1 Q \qquad (5.4.4)$$

si trovano ponendo tutti gli a_n e i b_n uguali a **0**.

Applicando lo stesso procedimento al sistema:

$$Q^{n+1} = (Q\overline{K} + \overline{Q}J)^n \qquad (5.4.5)$$
$$Q^{n+1} = (Qg_1 + \overline{Q}g_2)^n \qquad (5.4.6)$$

scriviamo direttamente, con un procedimento analogo al precedente, la tabella 5.4.4 che ci permette di trovare le equazioni degli ingressi per il J-K.

Tabella 5.4.4

g_1^n	g_2^n	Q^n	Q^{n+1}	J^n	K^n
0	0	0	0	0	a_0
1	0	0	0	0	a_1
0	1	0	1	1	a_2
1	1	0	1	1	a_3
0	0	1	0	b_0	1
1	0	1	1	b_1	0
0	1	1	0	b_2	1
1	1	1	1	b_3	0

Ricaviamo quindi:

$$J = g_2 \qquad (5.4.7)$$
$$K = \overline{g}_1 \qquad (5.4.8)$$

ponendo $a_0 = a_2 = \mathbf{1}$, $a_1 = a_3 = \mathbf{0}$, $b_0 = b_1 = \mathbf{0}$ e $b_2 = b_3 = \mathbf{1}$.

Per il flip-flop D si ricava con lo stesso procedimento, l'equazione dell'ingresso:

$$D = g_1 Q + g_2 \overline{Q} \tag{5.4.9}$$

Ci poniamo ora il problema della realizzazione circuitale del flip-flop di tipo J-K. L'equazione caratteristica del J-K si può considerare l'equazione applicativa per un flip-flop di tipo S-R, che sappiamo costruire, il che equivale a risolvere il problema della realizzazione circuitale del J-K come un problema di logica sequenziale da risolvere con S-R.

Nel nostro problema l'applicativa è caratterizzata (equazione del J-K) da:

$$g_1 = \overline{K} \tag{5.4.10}$$
$$g_2 = J \tag{5.4.11}$$

quindi le equazioni degli ingressi sono:

$$S = g_2 \overline{Q} = J\overline{Q} \tag{5.4.12}$$
$$R = \overline{g}_1 Q = KQ \tag{5.4.13}$$

e il J-K si realizza come in fig. 5.4.1.

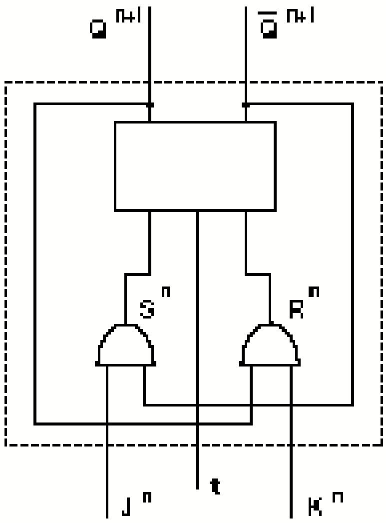

Figura 5.4.1

Analogamente si realizza il D con J-K o S-R. Notiamo solo che gli schemi di fig. 5.2.1 e 5.4.1 sono delle notevoli semplificazioni degli elementi di memoria reali disponibili nelle varie famiglie di circuiti logici. Nel capitolo successivo, dedicato ai contatori, sarà ulteriormente chiarito il metodo di progettazione di circuiti logici sequenziali.

5.5 Flip-flop Master-Slave

Un flip-flop molto comune è quello della figura 5.5.1, detto flip-flop Master-Slave.

Per analizzarne il funzionamento, supponiamo che, inizialmente, il segnale *t* sia a livello logico **1**. In queste condizioni il flip-flop costituito dalle porte *nd*1, *nd*2, *nd*3, *nd*4 è nello stato di memoria perché l'uscita di *inv*2, a livello logico **0**, forza le linee *a* e *b* a livello **1**. Il flip-flop formato da *nd*5, *nd*6, *nd*7, *nd*8 sarebbe in condizioni di cambiare stato , ma ciò non avviene perché le linee *c* e *d* del primo flip-flop sono bloccate dall'uscita di *inv*2.

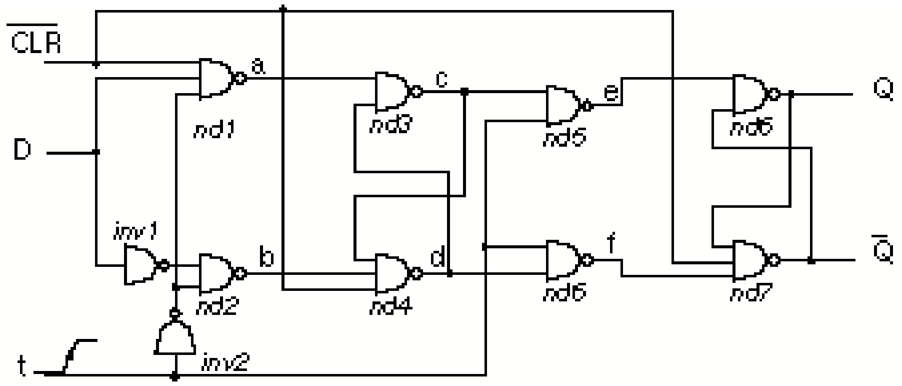

Figura 5.5.1

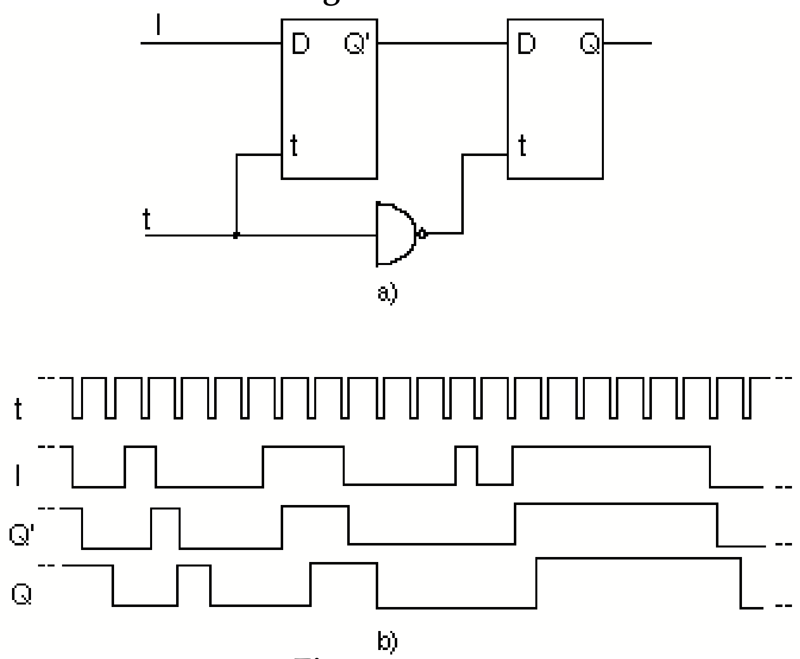

Figura 5.5.2

Quando *t* va a livello **0**, il primo flip-flop è in grado di seguire lo stato della linea *D*, ma il secondo flip-flop è adesso bloccato perché le linee *e* ed *f* sono entrambe a **1**. Nel momento che *t* torna al livello **1**, il primo flip-flop si blocca nuovamente e il secondo copia l'ultimo stato che aveva assunto il primo.

Si può dire che il dato, *D*, viene acquisito quando *t* è basso e viene trasferito in *Q* alla *transizione* di *t* da **0** a **1**.

Il flip-flop della figura, è anche fornito di un azzeramento asincrono che forza *D* a **0**, quando la linea $\overline{CLR}$ è posta a **0**.

Il flip-flop Master-Slave è descritto funzionalmente nella figura 5.5.2a, dove si sono usati due flip-flop D che commutano sul fronte positivo dell'orologio.

Nella figura 5.5.2b è descritto il comportamento del flip-flop, all'uscita *Q*, in funzione di un segnale si sincronismo periodico, *t*, e di un segnale d'ingresso *I*.

5.6 Esercizi

a)-Realizzare un flip-flop J-K avendo a disposizione un flip-flop D.

b)-Realizzare con un flip-flop D un elemento di memoria con equazione caratteristica:

$$Q^{n+1} = \overline{Q}^n$$

c)-Scrivere la tavola della verità del flip-flop la cui equazione caratteristica è:

$$Q^{n+1} = [Q \oplus (M \oplus N)]^n$$

Trovare la soluzione generale per gli ingressi M e N. Realizzare il flip-flop descritto sia con un elemento J-K che D.

d)-Studiare in modo analogo a primas il flip-flop la cui equazione caratteristica è:

$$Q^{n+1} = [\overline{Q} \oplus (M \oplus N)]^n$$