

Altri integrati dello stesso tipo sono il 74137 (come il 74138 ma fornito di latch per memorizzare i dati in ingresso) e il 74139 (doppio decoder/demux a due ingressi e quattro uscite con un ingresso dati/abilitazione).

Esistono alcuni integrati che svolgono solo la funzione di decoder decimale: possiedono dieci uscite e ricevono in ingresso quattro bit in codice BCD (7442) o in codice eccesso-3 (7443).

L'espansione del numero delle uscite, che può essere ottenuta collegando opportunamente più integrati, viene lasciata al lettore come esercizio.

ESEMPIO 5.2

Per controllare il funzionamento di una centrale remota, si vogliono trasmettere alla centrale principale quindici segnali d'allarme, caratterizzati da un certo ordine d'importanza, che saranno visualizzati mediante quindici LED rossi.

Essendo disponibile un cavo con soli cinque fili, s'individui la soluzione circuitale che consenta l'accensione, nella centrale principale, del LED rosso che segnala l'allarme di maggiore gravità presente nella centrale remota.

Nel caso di assenza di allarmi dovrà accendersi un LED verde. (Si trascurino i problemi elettrici dovuti alla lunghezza del collegamento e si suppongano tutti i segnali attivi a livello ALTO).

Soluzione

Tramite un priority encoder si codifica, con un numero binario di quattro bit, l'allarme a maggior priorità presente in ingresso. L'ingresso a più bassa priorità dell'encoder, è collegato al livello attivo (H) e viene codificato in assenza di allarmi. Le quattro uscite del codificatore, più il riferimento di massa per le tensioni, sono collegate alla centrale principale tramite un cavo a cinque fili. In ricezione il decoder attiva l'uscita corrispondente alla codifica binaria rilevata; all'uscita 0 è collegato il LED verde, che segnala la condizione di assenza di allarmi, mentre le restanti uscite sono collegate ai LED rossi.

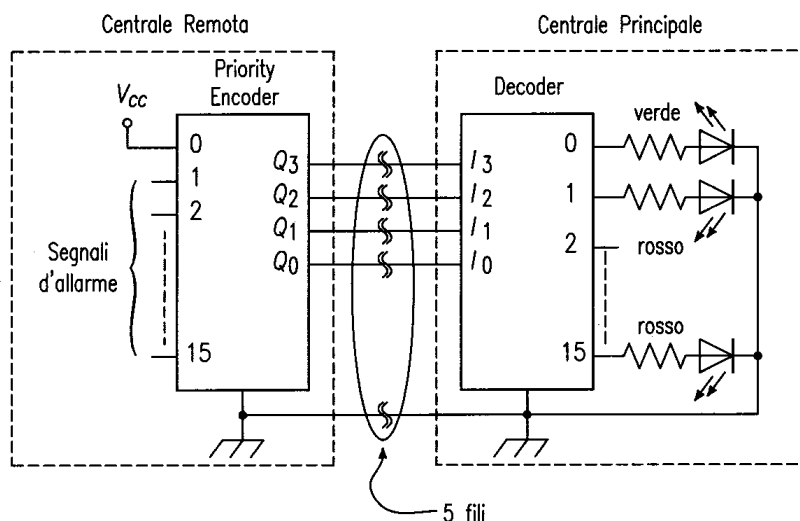



Fig. 5.14

5.4 Generazione di funzioni logiche con decoder e multiplexer

Nel cap. 3 si è affrontato il progetto delle reti combinatorie studiando le tecniche che consentono di passare da una tabella della verità ad una rete logica che la soddisfi. Si è rilevata anche la necessità di ottimizzare la rete per utilizzare il minimo numero di circuiti

integrati. Un modo per ridurre il numero degli integrati, consiste nel generare la funzione logica sfruttando decoder o multiplexer.

5.4.1 Utilizzo del decoder come generatore di funzioni

 Dallo schema a porte logiche di un decoder (fig. 5.9b) si nota come ad ogni uscita corrisponda un mintermine delle variabili d'ingresso. È quindi evidente che, per realizzare la funzione in forma canonica SP, sarà sufficiente porre in OR le uscite del decoder corrispondenti alle righe della tabella che presentano un 1 nella variabile d'uscita.

Se il decoder utilizzato possiede le uscite in logica negativa si dimostra, attraverso il teorema di De Morgan, che alla porta OR è necessario sostituire una porta NAND.

ESEMPIO 5.3

Generare la funzione logica specificata dalla tabella della verità di fig. 5.15.a, utilizzando un decoder.

Soluzione

Si pongono in OR le uscite del decoder corrispondenti alle righe della tabella che presentano il valore 1 della variabile d'uscita.

C	B	A	Z	
0	0	0	1	→ Y ₀
0	0	1	0	
0	1	0	0	
0	1	1	0	
1	0	0	1	→ Y ₄
1	0	1	0	
1	1	0	1	→ Y ₆
1	1	1	0	

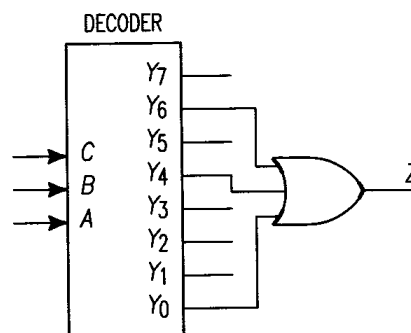


Fig. 5.15

a)

b)

Impiegando un decoder si può ottenere un risparmio sul numero di integrati utilizzati, quando si hanno più variabili d'uscita funzioni delle stesse variabili d'ingresso; è così sufficiente aggiungere all'integrato una porta OR per ogni variabile che s'intende sintetizzare, come mostrato nell'esempio 5.4.

ESEMPIO 5.4

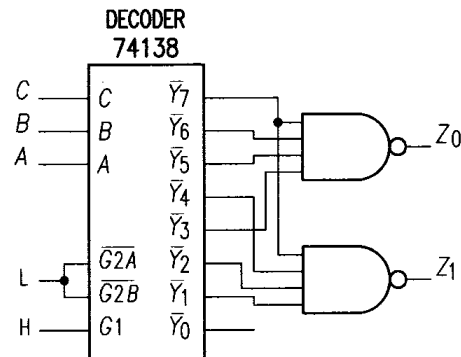
Utilizzando il decoder integrato 74138, si realizzi un circuito che segnali tramite un numero binario di due cifre, la quantità di 1 presenti sui tre ingressi. Ingressi e uscite vanno considerati in logica positiva.

Soluzione

La tabella della verità corrispondente al problema posto è rappresentata in fig. 5.16.a.

C	B	A	Z ₁	Z ₀
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

a)



b)

Fig. 5.16

Si procede per ognuna delle variabili d'uscita come nell'esempio 5.3, tenendo conto che, essendo le uscite del decoder 74138 in logica negativa, bisogna utilizzare due porte NAND.

5.4.2 Utilizzo del multiplexer come generatore di funzioni

Analizzando lo schema a porte logiche di un multiplexer (fig. 5.2), si nota come esso contenga una rete AND-OR. Ciò consente di realizzare la funzione logica corrispondente ad una data tabella della verità, utilizzando uno dei tre metodi seguenti:

⚙️ a) Mux con tanti ingressi di selezione quante sono le variabili d'ingresso della funzione

- 1) Si collegano le variabili d'ingresso agli ingressi di selezione del mux.
- 2) Si collegano al valore logico 1 gli ingressi dati del mux corrispondenti agli 1 della tabella della verità.
- 3) Si collegano al valore logico 0 gli ingressi dati del mux corrispondenti agli 0 della tabella della verità.

In questo modo la porta AND abilitata dalla combinazione delle variabili d'ingresso poste sui terminali di selezione, consentirà al valore logico presente sull'ingresso dati corrispondente, di passare verso l'uscita come specificato dalla tabella della verità.

ESEMPIO 5.5

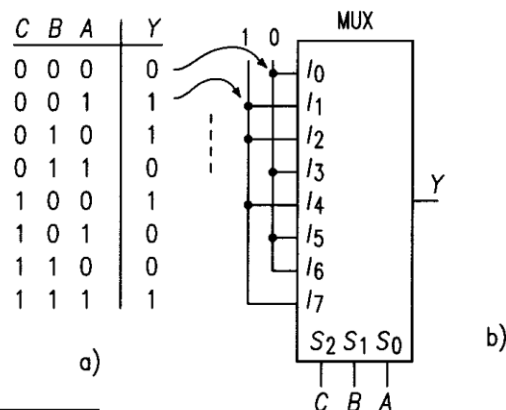


Fig. 5.17

Generare la funzione logica specificata dalla tabella in fig. 5.17a, utilizzando un multiplexer.

Soluzione

La soluzione è rappresentata nella fig. 5.17.



b) Mux con numero d'ingressi di selezione pari al numero delle variabili d'ingresso della funzione meno uno

- 1) Si inviano le $n-1$ variabili d'ingresso più significative (C e B nell'esempio 5.6) agli ingressi di selezione del mux.
- 2) Si suddivide la tabella in tante parti quante sono le possibili combinazioni delle $n-1$ variabili d'ingresso più significative; in ogni porzione queste variabili hanno valore costante, mentre quella meno significativa (A) commuta.
- 3) Si confronta poi, in ogni porzione, il valore della variabile meno significativa (A) con quello dell'uscita Y : a seconda della relazione tra A e Y ($Y = A$, $Y = \bar{A}$, $Y = 0$ o $Y = 1$) si collega l'ingresso dati del mux, corrispondente alla porzione di tabella esaminata, a A , \bar{A} , 0 o 1 .

Questa tecnica, rispetto alla precedente, consente, a parità di funzione da sintetizzare, l'utilizzo di un mux con un ingresso di selezione in meno e metà ingressi dati, richiedendo solo l'aggiunta di qualche porta NOT.

ESEMPIO 5.5

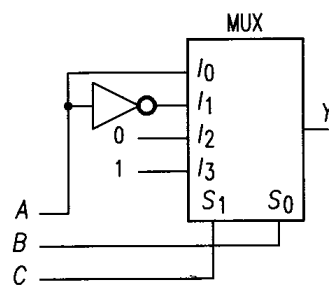
Generare la funzione logica con tre variabili d'ingresso, specificata dalla tabella della verità in figura, utilizzando un mux con due ingressi di selezione.

Soluzione

- 1) Si collegano le variabili d'ingresso B e C agli ingressi di selezione S_0 ed S_1 .
- 2) Si suddivide la tabella nelle quattro parti individuate da combinazioni identiche dei valori delle variabili B e C .
- 3) Per ogni porzione s'individua la funzione che lega Y con A .
- 4) Si collega l'ingresso corrispondente ad ogni porzione della tabella, in base alla funzione individuata nel punto precedente.

	C	B	A	Y	
I_0	0	0	0	0	$Y = A$
	0	0	1	1	
I_1	0	1	0	1	$Y = \bar{A}$
	0	1	1	0	
I_2	1	0	0	0	$Y = 0$
	1	0	1	0	
I_3	1	1	0	1	$Y = 1$
	1	1	1	1	

a)



b)

Fig. 5.18



c) Mux con numero d'ingressi di selezione pari al numero delle variabili d'ingresso della funzione meno due

- 1) S'invisano le $n-2$ variabili d'ingresso più significative (D e C nell'esempio 5.7) agli ingressi di selezione del mux.
- 2) Si suddivide la tabella in tante parti quante sono le possibili combinazioni delle $n-2$ variabili d'ingresso più significative; in ogni porzione queste variabili hanno valore costante mentre quelle meno significative (B e A) commutano.
- 3) Per ogni porzione si ricava, utilizzando le mappe di Karnaugh, la funzione di commutazione che lega le due variabili meno significative con l'uscita; nella funzione si sostituisce alla variabile d'uscita quella d'ingresso data individuata dagli ingressi di selezione.
- 4) Si collega ogni ingresso dati del mux alla corrispondente rete logica, ricavata al punto 3.

La tecnica c) consente l'utilizzo di un mux con due ingressi di selezione in meno rispetto al numero delle variabili d'ingresso, però necessita di una rete logica esterna che richiede un certo numero di circuiti integrati.

ESEMPIO 5.7

Generare la funzione logica con quattro variabili d'ingresso, specificata dalla tabella della verità in figura, utilizzando un mux con due ingressi di selezione.

Soluzione

- 1) Si collegano le variabili d'ingresso D e C agli ingressi di selezione S_1 e S_0 .
- 2) Si suddivide la tabella nelle quattro parti individuate da combinazioni identiche dei valori delle variabili D e C .
- 3) Per ogni porzione s'individua la funzione di commutazione che lega Y con A e B .
- 4) Si collegano i quattro ingressi dati ($I_0 \div I_3$) con le uscite delle reti corrispondenti alle funzioni ricavate al punto precedente.

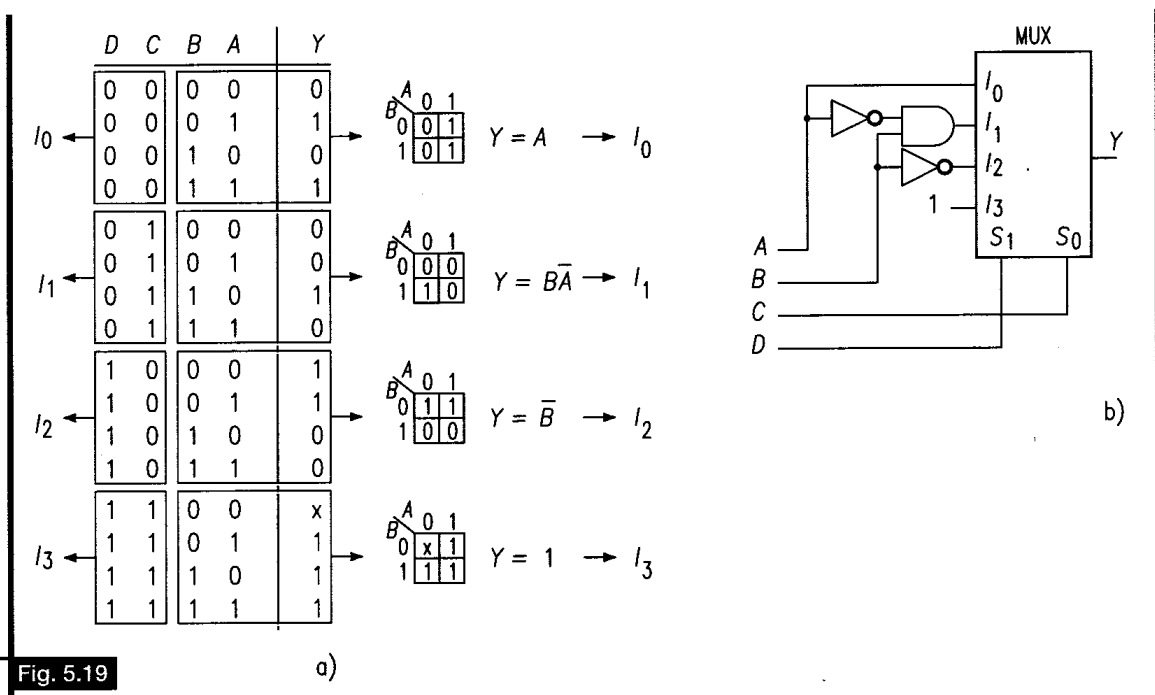


Fig. 5.19

Procedendo in maniera simile è possibile sintetizzare funzioni con tre o quattro variabili in più rispetto agli ingressi di selezione; questo comporta una rete esterna complessa, ma consente di affrontare il progetto di circuiti con un numero di variabili d'ingresso elevato, non risolvibili con il semplice uso delle mappe di Karnaugh.

Ad esempio è possibile realizzare funzioni con sette variabili d'ingresso, utilizzando un mux con quattro ingressi di selezione e portando ai sedici ingressi dati le combinazioni delle tre variabili d'ingresso meno significative, ricavate con la tecnica descritta.

Il quadro globale delle tecniche per il progetto di reti logiche combinatorie, descritte nel presente paragrafo e nel cap. 3, è riportato in fig. 5.20.

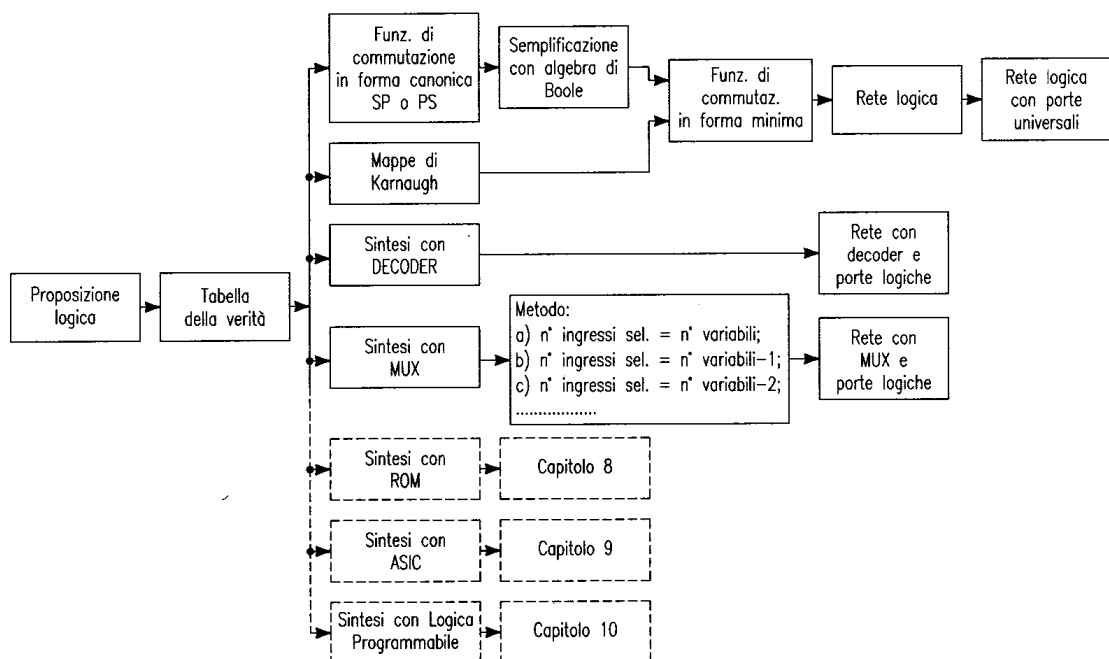


Fig. 5.20

Schema complessivo delle tecniche di progetto di reti combinatorie.