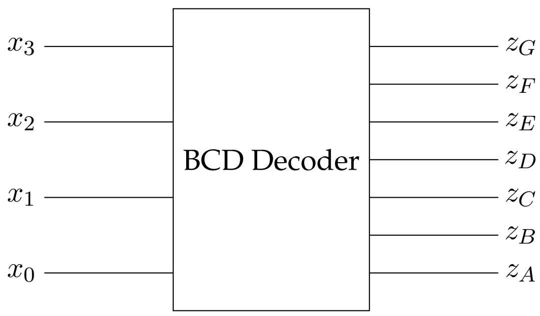
1 Lezione del 11-10-24

1.1 Sintesi di leggi non completamente specificate

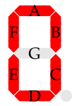
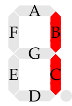
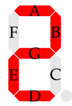
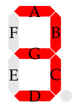
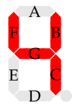
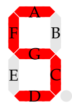
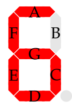
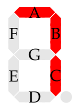
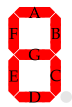
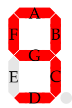
Applichiamo quanto abbiamo detto sulla sintesi di reti in forma SP a costo minimo, nel caso particolare in cui la legge non è completamente specificata (*don't care*).

Prendiamo in esempio un decodificatore BCD a 7 segmenti, simile a quello che si potrebbe trovare ad accendere le tracce di un display a cristalli liquidi.



Questo componente ha 4 variabili di ingresso, interpretate come una cifra in base 2 j, e 7 uscite, che specificano quali tracce accendere per visualizzare la cifra ottenuta in base 10. Si ha che con 4 variabili di ingresso si indirizzano 16 possibili configurazioni dei segmenti, quando ne abbiamo bisogno solo 10 (una per ogni cifra decimale). Le configurazioni di ingresso scartate si dicono quindi non specificate.

La tabella di verità della rete sarà quindi:

j	x_3	x_2	x_1	x_0	z_G	z_F	z_E	z_D	z_C	z_B	z_A	**Display**
0	0	0	0	0	0	1	1	1	1	1	1	
1	0	0	0	1	0	0	0	0	1	1	0	
2	0	0	1	0	1	0	1	1	0	1	1	
3	0	0	1	1	1	0	0	1	1	1	1	
4	0	1	0	0	1	1	0	0	1	1	0	
5	0	1	0	1	1	1	0	1	1	0	1	
6	0	1	1	0	1	1	1	1	1	0	1	
7	0	1	1	1	0	0	0	0	1	1	1	
8	1	0	0	0	1	1	1	1	1	1	1	
9	1	0	0	1	1	1	0	1	1	1	1	
10	1	0	1	0	–	–	–	–	–	–	–	
11	1	0	1	1	–	–	–	–	–	–	–	
12	1	1	0	0	–	–	–	–	–	–	–	
13	1	1	0	1	–	–	–	–	–	–	–	
14	1	1	1	0	–	–	–	–	–	–	–	
15	1	1	1	1	–	–	–	–	–	–	–	

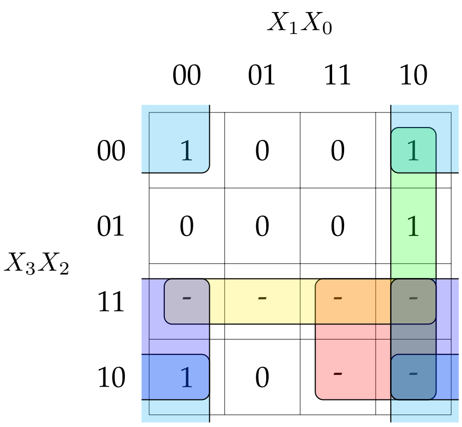
Visto che vogliamo sintetizzare reti su uscite singole, prendiamo la tabella di verità della rete sull'uscita z_E (le altre uscite richiederanno procedimenti simili):

j	x_3	x_2	x_1	x_0	z_E
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	–
11	1	0	1	1	–
12	1	1	0	0	–
13	1	1	0	1	–
14	1	1	1	0	–
15	1	1	1	1	–

Disegnamo quindi la mappa di Karnaugh. Quando si disegnano mappe di Karnaugh con elementi indeterminati, questi si interpretano diversamente a seconda che si stiano cercando i sottocubi principali, o che si stiano classificando:

- **Ricerca dei sottocubi principali:** si prendono come 1. Questo ci permette di prendere i sottocubi più grandi possibili nella ricerca dei primali (è irrilevante se si vanno a impostare uscite non specificate a 1).

  Si trova quindi:



	x_3	x_2	x_1	x_0
A	1	-	1	-
B	-	-	1	0
C	1	1	-	-
D	-	0	-	0
E	1	-	-	0

- **Classificazione dei sottocubi principali:** si prendono come 0. Così si evita di conservare implicanti che siano rilevanti su uscite non specificate (sarebbe inutile).

  Si ha quindi che i sottocubi A e C prendono solo SO1 indeterminati, ergo si scartano. Restano B e D essenziali, ed E a questo punto eliminabile, in quanto è già compreso in questi.

La sintesi completa dela rete è allora:

$$z_E = x_1\overline{x_0} + \overline{x_2 x_0}$$

1.2 Sintesi in forma PS

Abbiamo usato finora la forma SP (somma di prodotti). Esiste la duale, ovvero la forma PS (prodotto di somme). Per trovare questa forma, esiste un metodo parallelo a quello studiato per la SP, dove si parte dal considerare i maxtermini invece che dei mintermini, cioè scegliendo sottocubi negli elementi che valgono 0 della mappa di Karnaugh.

Non considereremo questo metodo, ma un'alternativa più veloce:

Algoritmo 1 per la sintesi in forma PS

Input: una legge combinatoria F
Output: la sintesi in forma PS di F
Si ricava $\overline{F}$ complementando F
Si reallizza una sintesi SP della legge $\overline{F}$
Si ottiene una sintesi di F aggiungendo un invertitore in uscita alla rete SP che sintetizza $\overline{F}$
Si applicano i teoremi di de Morgan, da destra verso sinistra

Algebricamente, l'ultimo passaggio significa scrivere $\overline{F}$ in forma SP:

$$\overline{z} = P_1 + ... + P_k$$

dove P_i sono prodotti di variabili di ingresso, e applicare de Morgan come:

$$z = \overline{\overline{z}} = \overline{P_1 + ... + P_k} = \overline{P_1} \cdot ... \cdot \overline{P_k}$$

A questo punto si applica di nuovo de Morgan, come:

$$\overline{P_i} = \overline{\prod x_j} = \sum \overline{x_j}$$

1.2.1 Dualità fra forme SP e PS

Con il procedimento presentato abbiamo che se $\overline{F}$ è in forma canonica SP, allora F è in forma canonica PS. Se la sintesi SP di $\overline{F}$ costa C, allora la sintesi PS di F costa C. Quindi se la sintesi SP di $\overline{F}$ è a costo minimo fra tutte le possibili sintesi SP, lo è anche la sintesi PS di F fra tutte le possibili sintesi PS. Se fosse il contrario, applicando de Morgan più volte avrei sintesi di costo sempre minore, violando la dualità.

A questo punto sappiamo effettuare la sintesi a costo minimo in forma SP di una qualsiasi legge F, e ponendo di sintetizzare prima $\overline{F}$ in forma SP, sappiamo anche trovare la sintesi a costo minimo in forma PS della stessa legge. Non possiamo determinare con sicurezza quale fra queste due sintesi ha costo minimo in generale, quindi bisogna controllare per forza la tabella della verità.

Troviamo ad esempio la sintesi in forma PS del BCD a 7 segmenti visto prima. Si ha che la negazine di F, su z_E, è:

j	x_3	x_2	x_1	x_0	z_E	$\overline{z_E}$
0	0	0	0	0	1	0
1	0	0	0	1	0	1
2	0	0	1	0	1	0
3	0	0	1	1	0	1
4	0	1	0	0	0	1
5	0	1	0	1	0	1
6	0	1	1	0	1	0
7	0	1	1	1	0	1
8	1	0	0	0	1	0
9	1	0	0	1	0	1
10	1	0	1	0	–	–
11	1	0	1	1	–	–
12	1	1	0	0	–	–
13	1	1	0	1	–	–
14	1	1	1	0	–	–
15	1	1	1	1	–	–

Ricaviamo quindi la mappa di Karnaugh:

X_3X_2 \ X_1X_0	00	01	11	10
00	0	1	1	0
01	1	1	1	0
11	-	-	-	-
10	0	1	-	-

	x_3	x_2	x_1	x_0
A	-	-	-	1
B	1	1	-	-
C	1	-	1	-
D	-	1	0	-

Si ha che B e C sono inutili, in quanto comprendono solo indeterminati. Restano allora A e D, entrambi essenziali, ergo la sintesi SP di $\overline{F}$ è:

$$\overline{F} = \overline{z_E} = x_0 + x_2\overline{x_1}$$

che neghiamo per ottenere nuovamente F:

$$F = z_E = \overline{x_0 + x_2\overline{x_1}}$$

A questo punto si può applicare de Morgan, prima sulla somma e poi sul prodotto a destra, per ottenere:

$$= \overline{x_0} \cdot \overline{x_2\overline{x_1}} = \overline{x_0} \cdot (\overline{x_2} + x_1)$$

Cioè è la sintesi di z_E in forma PS, che notiamo essere meno costosa della sintesi in forma SP, di due porte logiche in meno.