

Corso di Calcolatori Elettronici I

Minimizzazione di funzioni incompletamente specificate

Prof. Roberto Canonico



Università degli Studi di Napoli Federico II
Dipartimento di Ingegneria Elettrica
e delle Tecnologie dell'Informazione
Corso di Laurea in Ingegneria Informatica
Corso di Laurea in Ingegneria dell'Automazione

Funzioni incompletamente specificate

- Nei problemi di progetto, è possibile, in alcune circostanze, che il valore di una funzione booleana per alcune n-uple di valori delle sue variabili possa essere indifferentemente 0 o 1
 - Il valore può essere irrilevante ai fini del funzionamento del sistema descritto dalla funzione
 - Può esserci una dipendenza tra le variabili che esclude alcune combinazioni
-

Funzioni incompletamente specificate

- Si parla pertanto di “**punti di non specificazione**” o *don't care*
 - Due funzioni si dicono **compatibili** se assumono gli stessi valori, eccetto al più nei punti di non specificazione
 - Se i punti di non specificazione sono k le funzioni compatibili sono 2^k
 - Due funzioni compatibili “speciali”
 - f_0 = vale 0 **in tutti** i k punti di non specificazione
 - f_1 = vale 1 **in tutti** i k punti di non specificazione
-

Funzione incompletamente specificata: esempio

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>w</i>	<i>x</i>	<i>y</i>	<i>z</i>
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	–	–	–	–
1	0	1	1	–	–	–	–
1	1	0	0	–	–	–	–
1	1	0	1	–	–	–	–
1	1	1	0	–	–	–	–
1	1	1	1	–	–	–	–

Tabella 3.3 - Tabella di decodifica da codice BCD a Eccesso 3. I trattini indicano condizioni di indifferenza.

da: G. Bucci. Calcolatori Elettronici – Architettura e organizzazione. © McGraw-Hill, 2009

Presenza di don't care

- I don't care possono essere sfruttati per minimizzare ulteriormente la struttura di una funzione logica
 - *si può cercare tra tutte le funzioni compatibili quella che ha costo minimo*
-

Presenza di don't care

- Notate che '1' nella tabella di verità consentono di ottenere implicanti più "ampi"
 - D'altro canto, un maggior numero di '0' nella tabella di verità riduce il numero di mintermini da coprire
 - conviene considerare i d.c. come '1' quando si cercano gli implicanti, e come '0' quando si ricerca la copertura
 - **Metodo:** si determinano tutti i PI della funzione compatibile f_1 (esclusi quelli che coprono solo d.c.) e si imposta con questi il problema di copertura degli 1 della funzione compatibile f_0
-

Definizione di on-set , don't care set, off-set

- Sia $f(x_0, x_1, \dots, x_{n-1})=f(X)$ una funzione di n variabili
- Si definiscono i seguenti insiemi
 - ON-Set $\Sigma = \{X_i | f(X_i) = 1\}$
 - Don't Care-set o DC-Set $\Delta = \{X_i | f(X_i) = - \}$
 - OFF-Set $\phi = \{X_i | f(X_i) = 0\}$
- Valgono le relazioni

$$\Sigma \cup \Delta \cup \phi = B^n; \Sigma \cap \Delta = \emptyset; \Sigma \cap \phi = \emptyset; \Delta \cap \phi = \emptyset$$

- Due dei tre insiemi sono sufficienti a definire in modo completo e univoco una funzione
-

Esercizio 1 (1/3)

Minimizzare la funzione $f(x_3, x_2, x_1, x_0)$ specificata come segue:

- ON-Set={0100,1010,1011,1101,1110,1111}
- DC-Set={0011,0101,0110,0111}

Per applicare il metodo di McCluskey, si considerano i punti dell'insieme ON-Set \cup DC-Set e li si ordina in classi per numero di 1 crescente:

{{0100}{1010,0011,0101,0110}{1011,1101,1110,0111}{1111}}

Esercizio 1 (2/3)

4	0100	✓	4,5	010-	✓	4,5,6,7	01--	A
3	0011	✓	4,6	01-0	✓	3,7,11,15	--11	B
5	0101	✓	3,7	0-11	✓	5,7,13,15	-1-1	C
6	0110	✓	3,11	-011	✓	6,7,14,15	-11-	D
10	1010	✓	5,7	01-1	✓	10,11,14,15	1-1-	E
7	0111	✓	5,13	-101	✓	(c)		
11	1011	✓	6,7	011-	✓			
13	1101	✓	6,14	-110	✓			
14	1110	✓	10,11	101-	✓			
15	1111	✓	10,14	1-10	✓			
(a)			7,15	-111	✓			
			11,15	1-11	✓			
			13,15	11-1	✓			
			14,15	111-	✓			
			(b)					

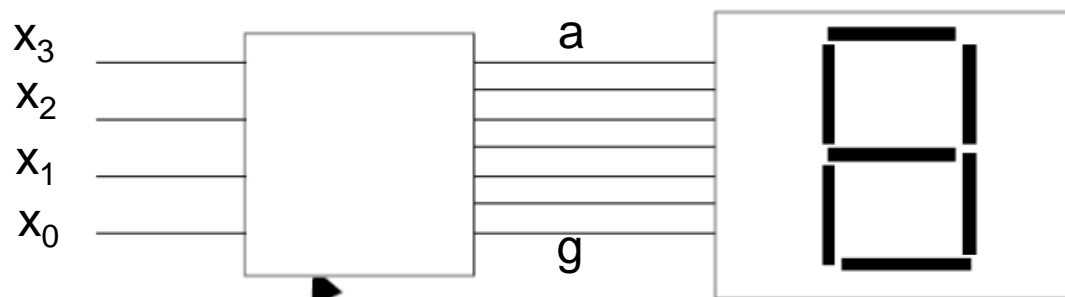
Esercizio 1 (3/3)

	4	10	11	13	14	15
A	X					
B			X			X
C				X		X
D					X	X
E		X	X		X	X

$$F = A + C + E =$$
$$= !xy + yv + xz$$

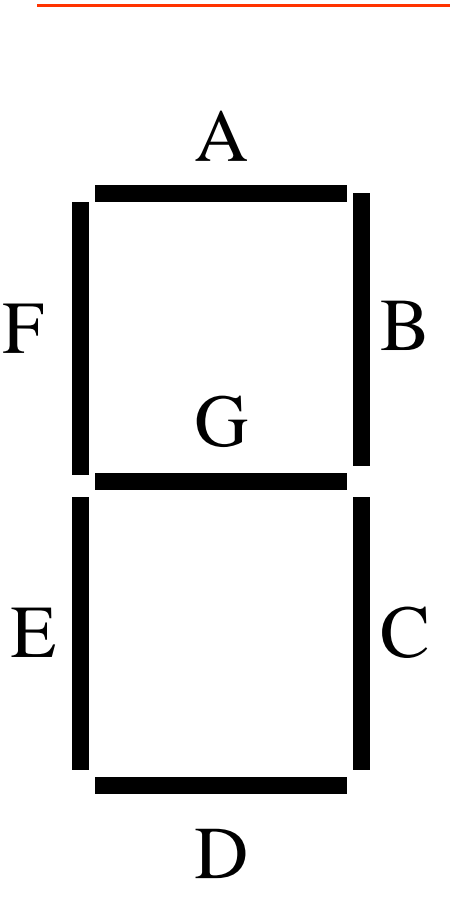
Esercizio 2 (1)

- Progettare una rete combinatoria che riceve in ingresso una cifra decimale codificata in binario (*codice BCD*) e produce in uscita sette segnali (uscite: *a, b, c, d, e, f, g*) che accendono (valore 1) o spengono (valore 0) i sette segmenti di un display decimale



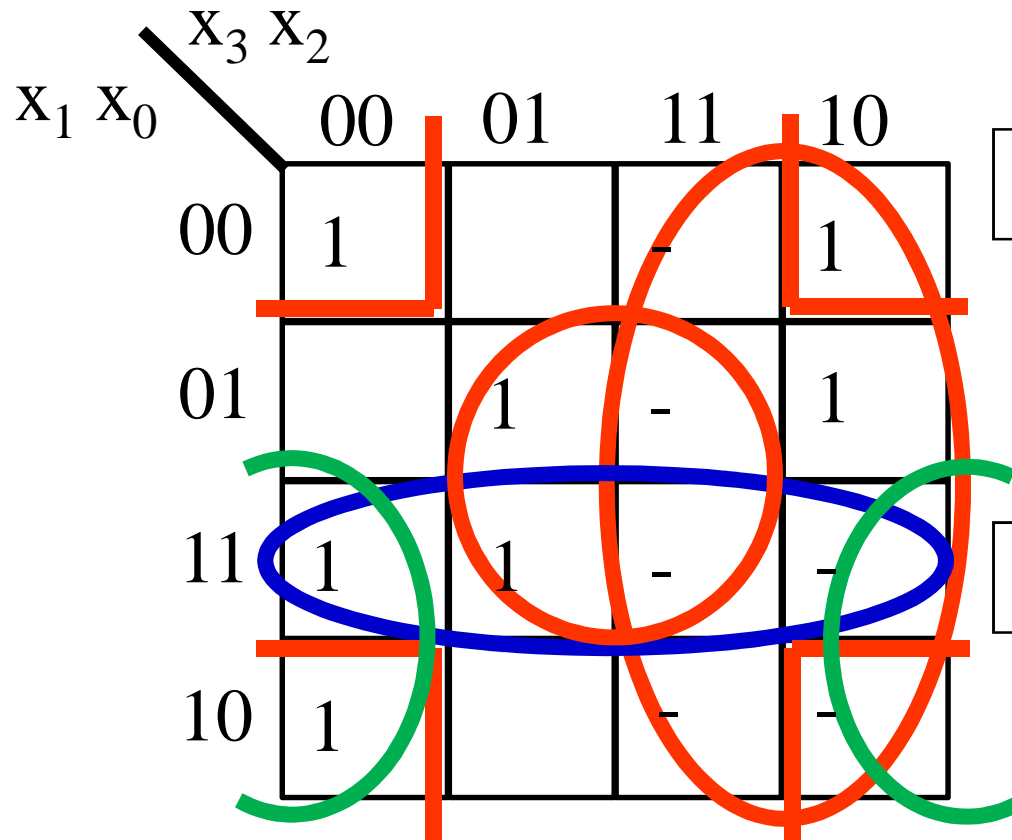
Transcodificatore per display a 7 segmenti

Esercizio 2 (2)



x3	x2	x1	x0	A	B	C	D	E	F	G
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	0	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0	0	1	1

Minimizzazione della funzione A



$$A = x_3 + x_2 x_0 + \overline{x_2} \overline{x_0} + x_1 x_0$$

oppure

$$A = x_3 + x_2 x_0 + \overline{x_2} \overline{x_0} + \overline{x_2} x_1$$