La seconda forma canonica
Circuiti notevoli

Prof. Alberto Borghese
Dipartimento di Scienze dell'Informazione
borghese@dsi.unimi.it

Università degli Studi di Milano

Sommario

La seconda forma canonica.

Circuiti combinatori notevoli.

Circuiti combinatori

- Circuiti logici digitali in cui le decisioni logiche dipendono solo da una combinazione degli input.
- Circuiti senza memoria. Ogni volta che si inseriscono in ingresso gli stessi valori, si ottengono le stesse uscite. Il risultato non dipende dallo stato del circuito.
- I circuiti combinatori descrivono delle funzioni Booleane. Il loro funzionamento può essere descritto come tabella della verità.
- Come nelle funzioni algebriche, il risultato è aggiornato immediatamente dopo il cambiamento dell'input (si suppone il tempo di commutazione trascurabile, tempo di attesa prima di guardare l'output sufficientemente ampio per permettere a tutti i circuiti la commutazione).

Funzione come espressione logica o come tabella delle verità

$F = A\,B + B\,\overline{C}$

A B C	A and B	B and $\overline{C}$	F
0 0 0	0	0	0
0 0 1	0	0	0
0 1 0	0	1	1
0 1 1	0	0	0
1 0 0	0	0	0
1 0 1	0	0	0
1 1 0	1	1	1
1 1 1	1	0	1

Forme canoniche

- Esiste un metodo per ricavare automaticamente un circuito che implementi una tabella di verità?
- Esistono 2 forme canoniche (equivalenti) che garantiscono di poter realizzare una qualunque tabella di verità con solo due livelli di porte OR, AND e NOT:

1) Somme di Prodotti (SOP)

$$F = \sum_{i=1}^{Q} m_i$$

$$F = \overline{A}\,B\,\overline{C} + A\,B\,\overline{C} + A\,B\,C$$

2) Prodotti di Somme (POS)

Razionale della seconda forma canonica

$$F = \overline{A}\,B\,\overline{C} + A\,B\,\overline{C} + A\,B\,C$$

A B C	F
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	1
1 1 1	1

F = 0

iif

(A = 0 B = 0 C = 0)
OR
(A = 0 B = 0 C = 1)
OR
(A = 0 B = 1 C = 1)
OR
(A = 1 B = 0 C = 0)
OR
(A = 1 B = 0 C = 1)

Verso la seconda forma canonica

$F = \bar{A} B \bar{C} + A B \bar{C} + A B C$

A B C	F
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	1
1 1 1	1

Maxtermine, M_j,: e' un prodotto di tutte le variabili di ingresso al quale corrisponde un valore 0 per la funzione. (e.g. $\bar{A}\ B\ \bar{C}$).

j indica il numero progressivo in base 10.

Possibile espressione della seconda forma canonica: $\bar{F} = \sum_{i=1}^{W} M_i$

$W \le 2^N$

$Q + W = 2^N$

$\bar{F} = \bar{A}\,\bar{B}\,\bar{C} + \bar{A}\,\bar{B}\,C + \bar{A}\,B\,C + A\,\bar{B}\,\bar{C} + A\,\bar{B}\,C$

La seconda forma canonica: prodotto di somme

$\bar{F} = \sum_{i=1}^{W} M_i \qquad M \le 2^N$

$\bar{F} = \left(\sum_{i=1}^{M} M_i\right) ==> F = \overline{\left(\sum_{i=1}^{M} M_i\right)}$

$F = \prod_i \overline{M_i} \qquad M_0 = \bar{A}\bar{B}\bar{C} => \bar{M_0} = A+B+C$

A B C	F
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	1
1 1 1	1

$\bar{F} = \bar{A}\,\bar{B}\,\bar{C} + \bar{A}\,\bar{B}\,C + \bar{A}\,B\,C + A\,\bar{B}\,\bar{C} + A\,\bar{B}\,C$

Applicando il secondo teorema di De Morgan:

$\boxed{F = (A+B+C)(A+B+\bar{C})(A+\bar{B}+\bar{C})(\bar{A}+B+C)(\bar{A}+B+\bar{C})}$

F = 1 quando nessun fattore si annulla

Razionale della seconda forma canonica

Product of sums

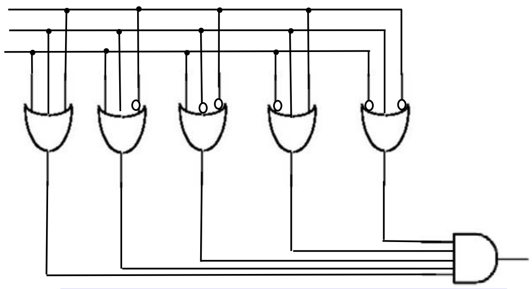
A B C	F
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	1
1 1 1	1

F = 1

iif

!(A = 0 B = 0 C = 0)
AND
!(A = 0 B = 0 C = 1)
AND
!(A = 0 B = 1 C = 1)
AND
!(A = 1 B = 0 C = 0)
AND
!(A = 1 B = 0 C = 1)

Il circuito della seconda forma canonica: POS



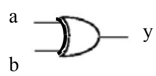
$\boxed{F = (A+B+C)(A+B+\bar{C})(A+\bar{B}+\bar{C})(\bar{A}+B+C)(\bar{A}+B+\bar{C})}$

Sommario

La seconda forma canonica.

Circuiti combinatori notevoli.

XOR



a b	y
0 0	0
0 1	1
1 0	1
1 1	0

SOP: $y = \bar{a}b + a\bar{b}$

POS: $y = (a+b)(\bar{a}+\bar{b})$

$y = (a+b)(\bar{a}+\bar{b}) = (a\bar{a}) + (b\bar{a}) + (a\bar{b}) + (b\bar{b}) =$

$= \bar{a}b + a\bar{b}$ cvd

$\boxed{y = a \oplus b}$

Multiplexer

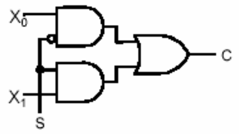
- E' caratterizzato da n linee di input (data),
- k linee di controllo (selezione).
- In base alla linea di controllo viene connessa all'uscita la linea di ingresso selezionata (cf. ROM).
- Quante linee di controllo, k, servono?

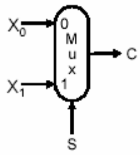
  k = ceil ($\log_2 n$)

Esempio: con 4 linee di input (da 0 a 3), se sulle linee di controllo c'è 11, in uscita si avrà il valore presente sulla linea 3

Multiplexer

S	x_0	x_1	C
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1





Il segnale di selezione S, "apre" la porta opportuna.

Sintesi della funzione Mux

S	x_0	x_1	C
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

SOP: $C = \bar{S}\, x_0\, \bar{x}_1 + \bar{S}\, x_0\, x_1 + S\, x_0\, \bar{x}_1 + S\, x_0\, x_1$

$= \bar{S}\, x_0 + S\, x_1$ cvd

$= \overline{\overline{\bar{S}\, x_0 + S\, x_1}} = \overline{\overline{(\bar{S}\, x_0)}\;\overline{(S\, x_1)}} =$

$\overline{(S + \bar{x}_0)(\bar{S} + \bar{x}_1)} = \overline{(\cancel{S\bar{S}} + \bar{x}_0\bar{S} + \bar{x}_1 S + \bar{x}_0\bar{x}_1)} =$

$\overline{(\bar{x}_0\bar{S} + \bar{x}_1 S + \bar{x}_0\bar{x}_1)} = (S + x_0)(\bar{S} + x_1)(x_0 + x_1)$

Sintesi della funzione Mux nella forma POS

S	x_0	x_1	C
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

POS: $C = (S+x_0+x_1)(S+x_0+\bar{x}_1)(\bar{S}+x_0+x_1)(\bar{S}+\bar{x}_0+x_1) =$

Definisco: $a = \bar{S}+x_1$

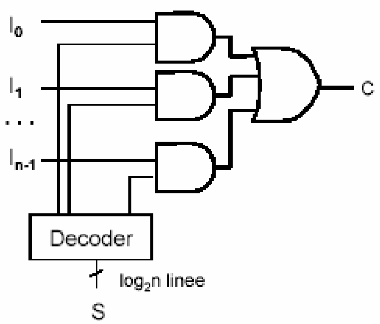
$b = S+x_0$

$[(b+x_1)(b+\bar{x}_1)]\; [(a+x_0)(a+\bar{x}_0)] =$

$[b+x_1\bar{x}_1]\; [a+x_0\bar{x}_0] = ba =$

$= \bar{S}\, x_0 + S\, x_1$ cvd

Mux a più vie.



Una sola porta alla volta viene aperta dal segnale S. Le porte sono mutuamente esclusive.

Decodificatore (decoder)

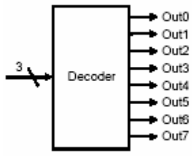
- E' caratterizzato da n linee di input e 2^n linee di output
- il numero binario espresso dalla configurazione delle linee di input è usato per asserire la sola linea di output di ugual indice.
- es.: con 4 linee di input e 16 di output (da 0 a 15), se in ingresso arriva il valore 0110, in uscita si alza la linea di indice 5 (la sesta!).
- utilizzato per indirizzare la memoria (cf. ROM).

Decodificatore (decoder)

- e' caratterizzato da n linee di input e 2^n linee di output
- il numero binario espresso dalla configurazione delle linee di input è usato per asserire la sola linea di output di ugual indice.
- es.: con 4 linee di input e 16 di output (da 0 a 15), se in ingresso arriva il valore 0110, in uscita si alza la linea di indice 5 (la sesta!).
- usato per indirizzare la memoria.

La funzione decoder

Decoder



a. A 3-bit decoder

A	B	C	U_0	U_1	U_2	U_3	U_4	U_5	U_6	U_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Le funzioni di uscita sono 2^n per n input:

$U_0 = \sim A \sim B \sim C$

… $U_j = m_j$

$U_7 = A\ BC$

Comparatore

- E' caratterizzato da 2 insiemi di n linee di ingresso ciascuna e un output.
- L'output vale 1 se i due insiemi di bit hanno uguale valore, 0 se sono diversi.

A_0	B_0	C_0
0	0	1
0	1	0
1	0	0
1	1	1

A_1	B_1	C_1
0	0	1
0	1	0
1	0	0
1	1	1

…

$$C = C_0\, C_1 \ldots C_{n-1}$$

$$C_k = \overline{a_k \oplus b_k}$$

Sommario

La seconda forma canonica.

Circuiti combinatori notevoli.