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 valuri, 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 AND B) OR (B AND NOT(C))

A B C	A and B	B and not(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:

Somme di Prodotti (SOP)
Prodotti di Somme(duale)

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

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

Razionale della seconda forma canonica

F = (A AND B) OR
(B AND NOT(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 = (A AND B) OR
(B AND NOT(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. $\overline{A}\,\overline{B}\,\overline{C}$).

j indica il numero progressivo in base 10.

Possibile espressione della seconda forma canonica:

$$\overline{F} = \sum_{i=1}^{W} M_i$$

$$W = 2^N$$

$$Q + W = 2^N$$

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

La seconda forma canonica: prodotto di somme

F = (A AND B) OR
(B AND NOT(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

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

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

$$F = \prod_i \overline{M_i}$$

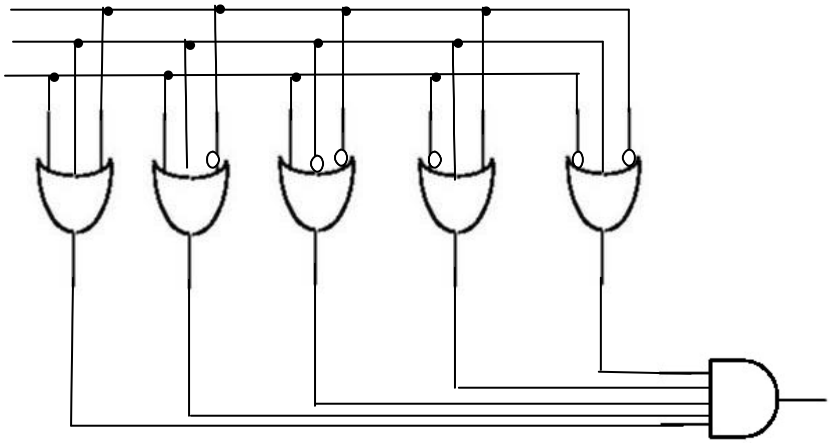
$$\overline{F} = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,C + \overline{A}\,B\,C + A\,\overline{B}\,\overline{C} + A\,\overline{B}\,C$$

Applicando il secondo teorema di De Morgan:

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

F = 1 quando nessun fattore si annulla

Il circuito della seconda forma canonica: POS



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

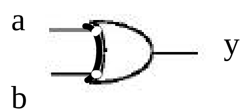
Sommario

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

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

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

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

Multiplexer

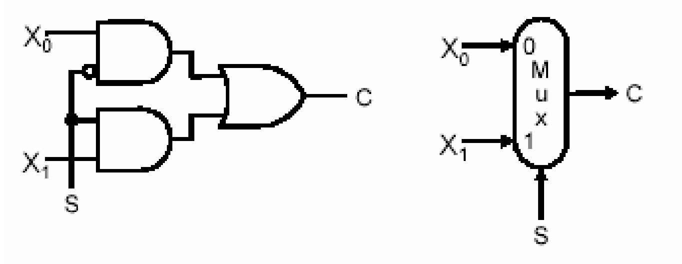
- E' caratterizzato da n linee di input (data),
- k linee di controllo (select).
- In base alla linea di controllo viene connessa all'uscita la linea di ingresso selezionata.
- Quante linee di controllo, k, servono?

$k = \text{ceil}(\log_2 n)$

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

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

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

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

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: $C = (S+x_0+x_1)(S+x_0+\overline{x_1})(\overline{S}+x_0+x_1)(\overline{S}+\overline{x_0}+x_1) =$

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

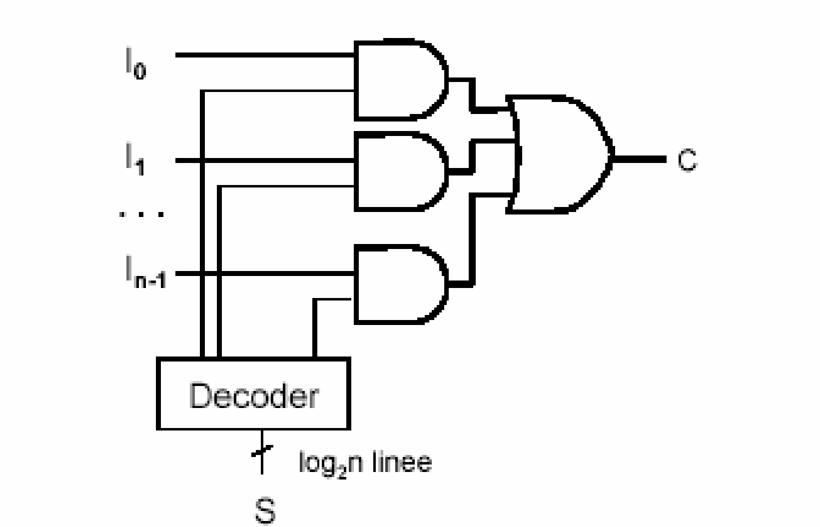
$b = S+x_0$

$[(b+x_1)(b+\overline{x_1})]\ [(a+x_0)(a+\overline{x_0})] =$

$[b+x_1\overline{x_1}]\ [a+x_0\overline{x_0}] = ba =$

$= \overline{S}\ x_0 + S\ x_1$ 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!).

• usato per inidirizzare la memoria.

Decodificatore (decoder)

• e' caratterizzato da n linee di input e 2^n linee di output

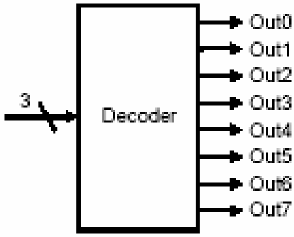
• il numero binario espresso dalla configurazione delle linee di input è usato per asserire la sola linea di output di ugual indice.

• 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 inidirizzare 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:

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

...

$U_7 = A\ BC$

$U_j = m_j$

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.