Reti Logiche

September 25, 2019

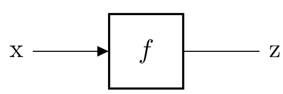
23 Sep 2019, 16:14

1 Propietá delle reti loiche

- Intrconnessione
- Scomposizione Uno schema logico puó essere rappresentato come composizione di reti piú semplici, (si arresta quando si arriva a *reti logiche elemnetari*)
- Elementari Descrivono causa effetto

Nei sistemi pratici la storia passata del sistema deve essere rappresentata attraverso un insieme finito di segnali. (i.e. il numero passato di stati della rete deve essere finito) (slide 36. 01_Modelli)
La funzione di uscita della rete dipende da ingresso corrente e stato interno della rete.

- Funzione di uscita o funzione macchina: $z(t) = f(x(t), y(t))$ (tabella, lookup-table)
- Next step function: $Y(t) = g(x(t), y(t))$, $g(x, y)$ chiamata **Funzione di stato**
- Funzione di marcatura dello stato: $y(t + \Delta t) = Y(t)$ o $y(t) = Y(t - \Delta t)$



Nelle reti logiche combinatorie manca la dipendena dal tempo (slide 41)
Reti sequenziali corrispondono al modello generale, contengono retroazione e ritardi
Rete senza ingressi provenienti dall'eseterno viene chiamata: rete autonoma. ($y(t) = y(t + 1)$: la rete é costante). Altra alternativa é un comportamento ciclico.
Se: $z = f(x)$ e $Y = g(x, y)$: rete combinatoria, dipende solo da z
Rete sequenziale con ingresso non attaccato a g é rete combinatoria (sostituire con figura)
Rete sequenziale con uscita non attaccata a g é rete combinatoria (sostituire con figura)
Sintesi multiuscita pg. 29 slide _2_RetiCombinatorie_I_parte
Numero funzioni di n variabili $F(n) = 2^{2^n}$, slide 30

x_1	x_2	x_3	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Metodo somma di prodotti: prendo solo righe con 1, le scrivo come prodotti e le sommo

$$\left.\begin{aligned} f_3 &= \bar{x_1}x_2x_3 \\ f_5 &= x_1\bar{x_2}x_3 \\ f_6 &= x_1x_2\bar{x_3} \\ f_7 &= x_1x_2x_3 \end{aligned}\right\} f = \bar{x_1}x_2x_3 + x_1\bar{x_2}x_3 + x_1x_2\bar{x_3} + x_1x_2x_3$$

Applicbile a qualsiasi tabella combinatoria (Espressione canonica *sum of products* SP)

1.1 Espressione canonica Prodotto di Somme: PS

Data una tabella della veritá, consideriamo gli zeri della funzione (OFF-set). Per ogni zero possiamo costruire un termine somma in cui compaiono come variabili dipendenti tutte le variabili della funzione di partenza. Le variabili compaiono in forma vera se compaiono come zeri, vengono prese in forma complementata se compaiono come 1 nella riga della tabella.

$$\left.\begin{aligned} s_0 &= x_1x_2x_3 \\ s_1 &= x_1x_2\bar{x_3} \\ s_2 &= x_1\bar{x_2}x_3 \\ s_4 &= \bar{x_1}x_2x_3 \end{aligned}\right\} S = (x_1x_2x_3)(x_1x_2\bar{x_3})(x_1\bar{x_2}x_3)(\bar{x_1}x_2x_3)$$

Si riescono a sintetizzare reti piú ottimizzate attraverso le mappe di Karnugh o alg. Quiaf-McCluskeyt
N letterari: si applica su soltanto al max 2 livelli. Conta il numero di morsetti del 1^o livello (somma: N_{let} = 12, prodotto 12, altro 6)

2 24 Sep 2019, 9:02

Prendendo in considerazione l'esempio del Full-Adder (slide 4)

2.1 Espressione canonica ***SoP*** delle funzioni S_0 e C_1

$S_0 = \bar{x_0}\bar{y_0}c_1 + \bar{x_0}y_0\bar{c_0} + x_0\bar{y_0}\bar{c_0} + x_0y_0c_0 = m_1 + m_2 + m_4 + m_7 = \sum m_3(1,2,4,7)$
$C_1 = \bar{x_0}y_0c_0 + x_0\bar{y_0}c_0 + x_0y_0\bar{c_0} + x_0y_0c_0 = m_3 + m_5 + m_6 + m_7 = \sum m_3(3,5,6,7)$
Un modo piú preciso sarebbe: $S(x_0, y_0, c_0)$, esplicitando l'ordine in cui vengono prese le variabili
$\sum m_3$ indica la sommatoria dei mintermini di 3 variabili

2.2 Espressione canonica ***PoS***

$S_0(x_0, y_0, c_0) = (x_0 + y_0 + c_0)(x_0 + \bar{y_0} + \bar{c_0})(\bar{x_0} + y_0 + \bar{c_0})(\bar{x_0} + \bar{y_0} + c_0) = M_0M_3M_5M_6 = \prod M_3(0,3,5,6)$

2.3 Teorema di Shannon

Ogni espressione logica di n variabili puó essere espressa come:

$$f(x_1, x_2, \ldots, x_n) = x_1 \cdot f(0, x_2, \ldots, x_n) + x_1 \cdot f(1, x_2, \ldots, x_n)$$

Dall'espressione di una funzione riusciamo a riottenere la somma dei mintermini, utilizzando il th. di Shannon:

$$
\begin{aligned}
z &= x_1x_2 + x_1x_3 + x_2x_3 = \\
&= \bar{x_1}(0 \cdot x_2 + 0x_3 + x_2x_3) + x_1(1x_2 + 1x_3 + x_2x_3) = \\
&= \bar{x_1}(x_2x_3) + x_1(x_2 + x_3 + x_2x_3) = \\
&= \bar{x_2}(x_1(0x_3) + x_1(0 + x_3 + 0x_3)) + x_2(\bar{x_1}(1x_3) + x_1(1 + x_2 + 1x_3)) = \\
&= \bar{x_2}(x_1x_3) + x_2(\bar{x_1}x_3 + x_1) = \\
&= \bar{x_3}(\bar{x_2}x_1 0 + x_2\bar{x_1} 0 + x_2x_1) + x_3(\bar{x_2}x_1 1 + x_2\bar{x_1} 1 + x_2x_1) = \\
&= \bar{x_2}x_2x_1 + x_3\bar{x_2}x_1 + x_3x_2\bar{x_1} + x_3x_2x_1 = \\
&= m_3 + m_5 + m_6 + m_7
\end{aligned}
$$

Le leggi di de morgan dimostrano che le famiglie (And-Not) e (Or-Not) sono funzionalmente complete.
DECODER: ad n ingresi corrispondono 2^n uscite

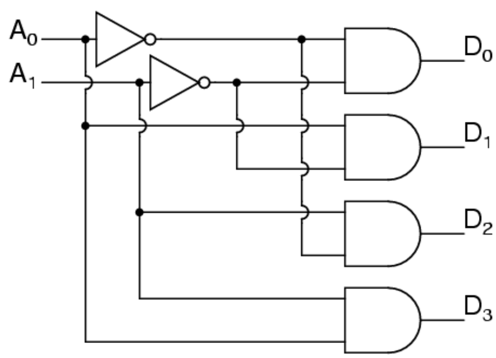


Figure 1: decoder

Espressione generale SP: $f(x_{n-1}, \ldots, x_1, x_0) = \sum_{i=0,2^n-1} m(i)f(i)$
Espressione generale PS: $f(x_{n-1}, \ldots, x_1, x_0) = \prod_{i=0,2^n-1}(M(i) + f(i))$
Il decoder sta in corrispondenza con le espressioni canoniche, il multiplexer sta un corrispondenza con le espressioni generali.

3 25 Sep 2019, 14:46

- Il segnale RC (RC_0 e RC_1) dipendono da $n-1$ variabili. e non da n variabili.
- RC_0 e RC_1 potrebbero essere semplificabili

Una funzione di n variabili puó essere espressa come una combinazione di due funzioni di n-1 variabili attraverso un Multiplexer.

$$f(x_1, x_2, \ldots, x_n) = \bar{x_1}\bar{x_2}f(0, 0, \ldots, x_n) + \bar{x_1}x_2f(0, 1, \ldots, x_n) + x_1\bar{x_2}f(1, 0, \ldots, x_n) + x_1x_2f(1, 1, \ldots, x_n)$$

Funzione semplificata di due variabili con Th. di Shannon.

AB \ CD	00	01	11	10
00	0	1	0	0
01	1	1	1	1
11	1	0	1	1
10	0	1	0	0

$$
\begin{aligned}
f_0 &= \bar{C}D \\
f_1 &= 1 \\
f_3 &= C + \bar{D} \\
f_2 &= \bar{C}D
\end{aligned}
$$

AB \ CD	00	01	11	10
00	0	1	0	0
01	1	1	1	1
11	1	0	1	1
10	0	1	0	0

$$
\begin{aligned}
f_0 &= B + \bar{C}D \\
f_1 &= B\bar{D} + BC + \bar{B}\bar{C}D
\end{aligned}
$$

Supponendo di dover realizzare la funzione con un multiplexer a 4 vie, utilizzando esclusivamente un mux e non altri componenti:
Nel caso in cui fossimo partiti dalle variabili CD:

$$
\begin{aligned}
f_0 &= B \\
f_1 &= \bar{A} + \bar{B} = A \bar{\cdot} B \\
f_2 &= B \\
f_3 &= B
\end{aligned}
$$

- Espressione canonica SP
  - AND per mintermini + OR
  - Decoder + or
- Espressione canonica PS
  - OR per Maxtermini + AND
  - Decoder a NAND + AND
- Espressione generale SP e PS: MuX
- Espressione parziali: (1 MUx + altro)

a	b	c	Sum	Count
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Count $= \bar{a}bc + a\bar{b}c + ab\bar{c} + abc = \bar{a}bc + a\bar{b}c + ab$

$$\begin{cases} N_{op} = 5 \\ N_m = 3 \cdot 4 + 4 = 16 \end{cases} \rightarrow \begin{cases} N_{op} = 4 \\ N_m = 11 \end{cases}$$

Definiamo come **distanza di Hamming** ($\mathcal{D}_H$) il numero di bit che differiscono in 2 configurazioni binarie
Es: $\mathcal{D}_H(110, 111) = 1$
Se $\mathcal{D}_H(C_i, C_j) = 1 \Rightarrow C_i \,\mathrm{e}\, C_j$ sono **adiacenti**.

a \ bc	00	01	11	10
0	0	0	1	0
1	0	1	1	1

Raggruppamento rettangolare di 2 caselle 2^1 caselle ($2^k = 2^1$) (Raggruppamento di ordine 1)
$C_{out} = I_1 + I_2 + I_3 = ac + ab + bc \qquad (N_{op} = 4, N_m = 9)$

ab \ cd	00	01	11	10
00	0	1	0	0
01	1	1	1	1
11	1	-	1	1
10	0	1	0	0

E= 0

ab \ cd	00	01	11	10
00	-	0	0	-
01	1	0	-	0
11	1	-	1	-
10	0	1	0	-

E= 1

$Z(a, b, c, d, e)$
'-' rappresenta una condizione di indifferenza.
Nei raggruppamenti prendo in considerazione le variabili che rimangono costanti

$$Z(a, b, c, d, e) = b\bar{e} + \bar{c}d\bar{e} + b\bar{c}\bar{d} + a\bar{c}d + ab$$

$N_{lett} = 13$, ingressi del primo livello
$N_{op} = 6$, numero mintermini + 1
$N_m = 18$, tutte le lettere che compaiono + mintermini

3.1 Sintesi prendendo in considerazione gli zeri

Uno 0 di una funzione é un maxtermine, ed esprime un solo 0. L'obbiettivo é costruire termini somma (Implicati)

ab \ cd	00	01	11	10
00	0	1	0	0
01	1	1	1	1
11	1	-	1	1
10	0	1	0	0

$E = 0$

ab \ cd	00	01	11	10
00	-	0	0	-
01	1	0	-	0
11	1	-	1	-
10	0	1	0	-

$E = 1$

$$Z = (B+\bar{C})(A+\bar{D}+\bar{E})(\bar{C}+D+\bar{E})(B+D) = (B \downarrow \bar{C}) \downarrow (A \downarrow \bar{D} \downarrow \bar{E}) \downarrow (\bar{C} \downarrow D \downarrow \bar{E}) \downarrow (B \downarrow D)$$

$N_{op} = 5$
$N_m = 14$
Sintesi di $Z(A, B, C, D, E)$

- mediante soli MUX a 4 vie
- 1 porta logica + MUX a 4 vie