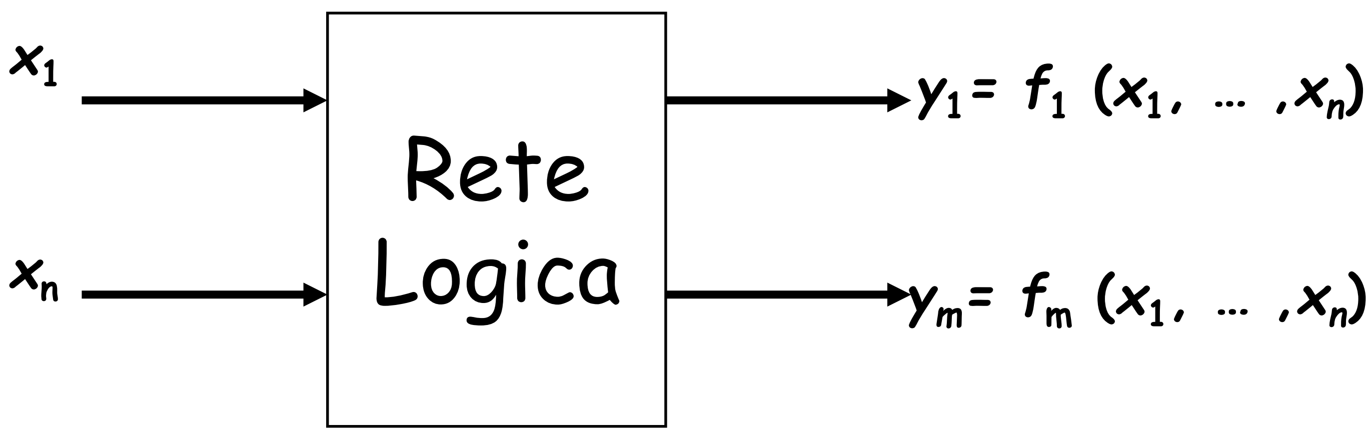
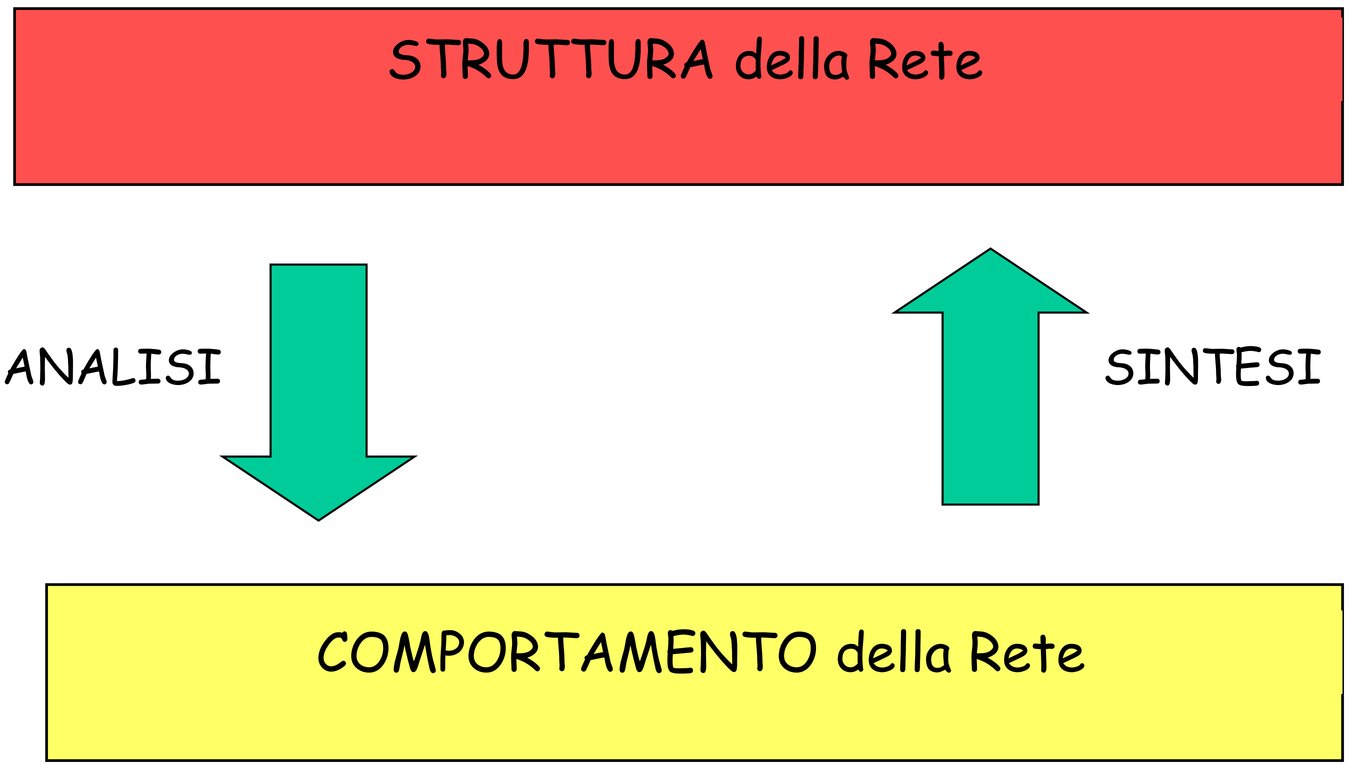
Reti Combinatorie: sintesi

Rete Logica

- Una rete logica è un circuito elettronico digitale in grado di realizzare una o più funzioni di commutazione



Analisi e Sintesi di reti logiche



Nomenclatura

• Consideriamo funzioni di commutazione di n variabili espresse come somme di prodotti (OR di AND)

$$Y = P_1 + P_2 + \dots + P_K$$

• Pi prodotto di $k \leq n$ variabili diretta o negate (una variabile diretta o negata è chiamata **letterale**), ex n=4

$$Y = X_3X_1 + X_4X_2X_1 + X_4X_3X_2X_1$$

• P_i si chiama **implicante** della funzione, $P_i \rightarrow y$.

se ogni volta che $P_i = 1$ allora si ha che y=1, ex

$$X_4X_3X_2X_1 \rightarrow y$$

• **Implicante primo**: implicante per il quale non è possibile eliminare un letterale dalla sua espressione ed ottenere ancora un implicante, ex

$$X_4X_2X_1$$

• **Espressione minima**: espressione nella quale non possono essere eliminati né un letterale né un termine senza alterare la funzione rappresentata dall'espressione stessa.

$$Y = X_3X_1 + X_4X_2X_1$$

Sintesi di reti combinatorie

- Una rete combinatoria realizza una funzione di commutazione
- Nell'attività di progetto é necessario tenere conto sia delle **prestazioni** che del **costo**.
- **Necessità**: rete logica il più veloce possibile.
- Quindi a parità di velocità é necessario **ottimizzare** il costo.
- Data una tabella di verità è possibile ricavare più **espressioni equivalenti** che la rappresentano.

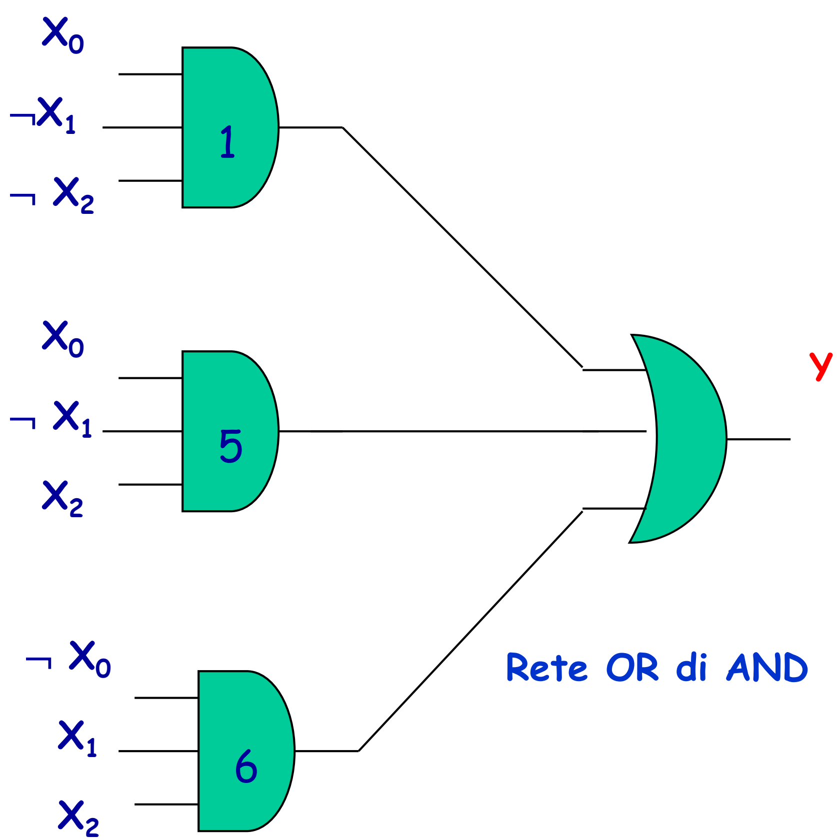
Forme canoniche come reti

Ogni funzione Y può essere espressa come somma canonica, cioè è possibile realizzarla usando 2 livelli di porte logiche (OR – AND)

VELOCITA' MASSIMA

$Y = \Sigma(1,5,6)$

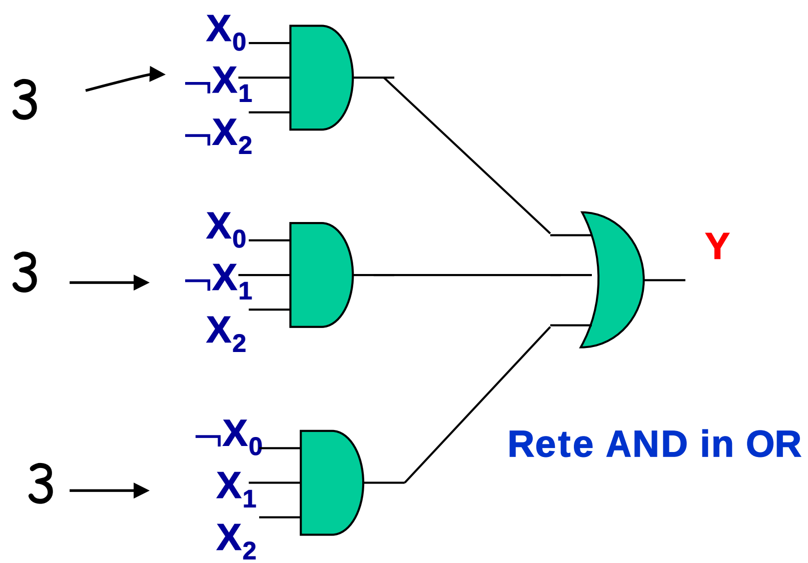
001 101 110



Rete OR di AND

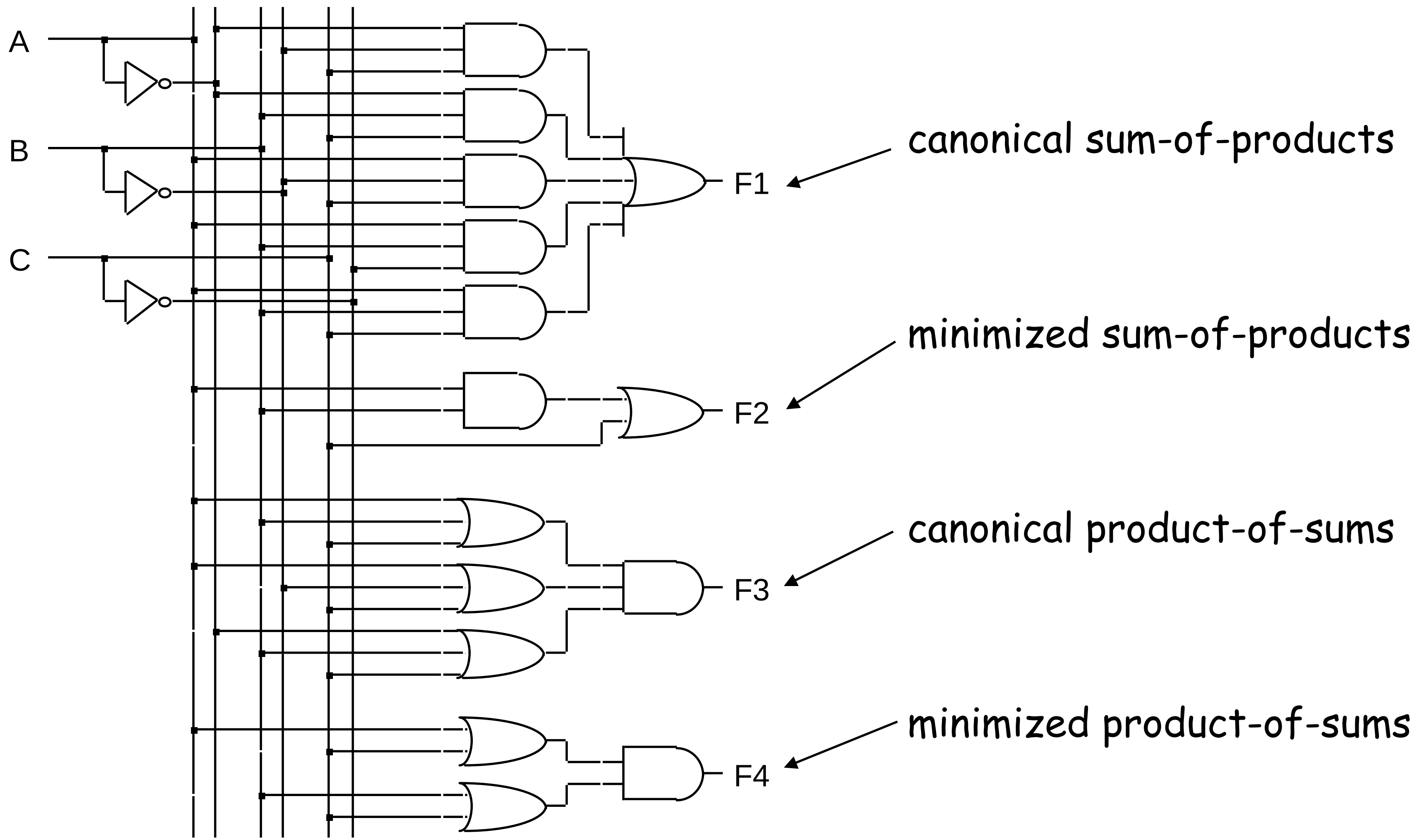
Costo di una funzione

- Costo: Somma del numero di letterali e degli implicanti



3+3+3+3=12

Alcune possibili realizzazioni di F = AB + C



Mappe di Karnaugh (MK)

- Le mappe di Karnaugh sono tabelle che permettono la rappresentazione e la semplificazione delle funzioni di commutazione fino a quattro variabili. E' possibile usarle, con qualche difficoltà, anche per funzioni di cinque e sei variabili

- Le mappe di Karnaugh per le funzioni di 2, 3, 4, 5 variabili sono divise in tante caselle (o "celle") quanti sono i corrispondenti mintermini (4, 8, 16, 32).

MK per 2 variabili

Tabella di verità

X_0	X_1	Y
0	0	
0	1	
1	0	
1	1	

Y		X_0 = 0	X_0 = 1
X_1	0	0	1
	1	2	3

MK per 2,3,4 variabili

Y

X_1 \ X_0	0	1
0	0	1
1	2	3

Y

X_2 \ $X_1 X_0$	00	01	11	10
0	0	1	3	2
1	4	5	7	6

Y

$X_3 X_2$ \ $X_1 X_0$	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

- Le caselle adiacenti corrispondono a configurazioni delle variabili di ingresso che differiscono di <u>un solo</u> bit
- Anche le caselle sulle due colonne estreme sono da considerarsi adiacenti, come se la mappa fosse originariamente su una sfera che è stata tagliata e spianata.

Esempio

$$Y = \Sigma(1,5,6) = \neg x_3 \neg x_2 \neg x_1 x_0 + \neg x_3 x_2 \neg x_1 x_0 + x_3 x_2 \neg x_1 \neg x_0$$

0001->m_1 0101->m_5 1100->m_{12}

Y

$x_3 x_2$ \ $x_1 x_0$	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

Y

$x_3 x_2$ \ $x_1 x_0$	00	01	11	10
00	0	1	0	0
01	0	1	0	0
11	0	0	0	1
10	0	0	0	0

Se la funzione è data come somma di mintermini, basta scrivere 1 in tutte le celle corrispondenti ai mintermini della somma

Semplificazione

	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

Y

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

- $m_1 + m_5 = \overline{X_3}\,\overline{X_2}\,\overline{X_1}X_0 + \overline{X_3}X_2\overline{X_1}X_0$
  $= \varphi\, \overline{X_2} + \varphi\, X_2$
  $= \varphi\, (\overline{X_2} + X_2) = \varphi$

$\varphi = \overline{X_3}\,\overline{X_1}X_0$

m_1 ed m_5 non sono implicanti primi, mentre φ è un implicante primo

In una mappa K un implicante primo corrisponde ad un raggruppamento di 2^i **celle adiacenti (cubi)**, sia orizzontalmente o verticalmente, non incluso in altri raggruppamenti

Esempi

Y

$X_3 X_2$ \ $X_1 X_0$	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	0	0	0
10	0	0	0	0

$(\neg X_3 X_0)$

Y

$X_3 X_2$ \ $X_1 X_0$	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	0	0	0	0
10	0	0	0	0

$(\neg X_3 \neg X_0)$

Y

$X_3 X_2$ \ $X_1 X_0$	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	0	1	1	0
10	0	1	1	0

(X_0)

Y

$X_3 X_2$ \ $X_1 X_0$	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	1	0	0	1
10	1	0	0	1

$(\neg X_0)$

Altre definizioni

- ***Implicante primo essenziale:*** implicante primo rappresentato da un cubo che copre almeno un 1 non coperto da altri implicanti primi

- ***Cuore di una funzione:*** insieme degli implicanti primi essenziali

Esempi

Y $X_1 X_0$

	00	01	11	10
00			1	1
01		1	1	
11		1	1	
10		1	1	

Y $X_1 X_0$

$X_3 X_2$	00	01	11	10
00	1	1	1	1
01		1	1	1
11	1	1	1	1
10		1		

Y $X_1 X_0$

	00	01	11	10
00		1	1	
01	1	1	1	
11	1	1	1	
10		1	1	

Y $X_1 X_0$

$X_3 X_2$	00	01	11	10
00	1			1
01	1			1
11	1	1	1	1
10	1	1	1	1

Algoritmo per la minimizzazione

1. Si segnano con 1 le caselle relative ai mintermini della funzione

2. Si identificano gli implicanti primi essenziali e si disegnano i relativi cubi. Se sono coperti tutti i mintermini si va al passo 4, altrimenti al 3.

3. Si coprono i restanti mintermini con il minor numero possibile di implicanti
4. Fine della procedura

- Commento: non sistematicità del passo 3

Esempio

Y

$X_3 X_2$ \ $X_1 X_0$	00	01	11	10
00	0	1	1	0
01	0	1	1	1
11	0	1	0	0
10	1	1	0	0

Funzioni parzialmente specificate

Funzioni in cui non sono possibili alcune configurazioni delle variabili di ingresso o non interessa il valore di uscita per alcune configurazioni di ingresso

Esempio: date quattro variabili di commutazione codificanti i numeri 0..9 la funzione è vera quando il numero è divisibile per 3.

Tabella di verità e MK di una funzione parz. spec.

x_3 x_2 x_1 x_0	f
0 0 0 0	1
0 0 0 1	0
0 0 1 0	0
0 0 1 1	1
0 1 0 0	0
0 1 0 1	0
0 1 1 0	1
0 1 1 1	0
1 0 0 0	0
1 0 0 1	1
1 0 1 0	d.c.c.
1 0 1 1	d.c.c.
1 1 0 0	d.c.c.
1 1 0 1	d.c.c.
1 1 1 0	d.c.c.
1 1 1 1	d.c.c.

Realizzare un circuito che riconosca se un numero compreso tra 0 e 9 sia divisibile per 3.

Y

$X_1 X_0$

$X_3 X_2$	00	01	11	10
00	1	0	1	0
01	0	0	0	1
11	-	-	-	-
10	0	1	-	-

Algoritmo per la minimizzazione

1. Si segnano con 1 le caselle relative ai mintermini e con – le d.c.c. (don’t care condition) della funzione.

2. Si identificano gli implicanti primi essenziali rappresentati da cubi costituiti da 1 e – ed aventi almeno un 1. Se sono coperti tutti i mintermini si va al passo 4, altrimenti al 3.

3. Si coprono i restanti mintermini con il minor numero possibile di cubi aventi le dimensioni massime e costituiti da 1 e -.
4. Fine della procedura

- Commento: non sistematicità del passo 3