# ADDIZIONE BINARIA

Vogliamo realizzare un dispositivo che effettui l'addizione binaria tra due bit. Avremo bisogno di due ingressi che nominiamo A e B. Analizziamo ora tutti i possibili valori.

1. $A = 0;\ B = 0 \quad \rightarrow \quad A + B = 0;$
2. $A = 0;\ B = 1 \quad \rightarrow \quad A + B = 1;$
3. $A = 1;\ B = 0 \quad \rightarrow \quad A + B = 1;$
4. $A = 1;\ B = 1 \quad \rightarrow \quad A + B = 0 \quad con\, riporto\, di\ 1.$

Abbiamo bisogno anche di due uscite: una, che chiamiamo S che rappresenta la somma ed una, che chiamiamo C (Carry bit) che rappresenta il riporto.
Questo dispositivo prende il nome di *half-adder*.

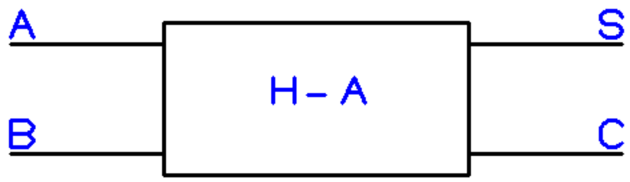


*Figura 1 Half-Adder*

Scriviamo la tabella della verità:

| A | B | S | C |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

Si trova:

$$\begin{cases} S = A \oplus B \\ C = A \cdot B \end{cases}$$ [1]

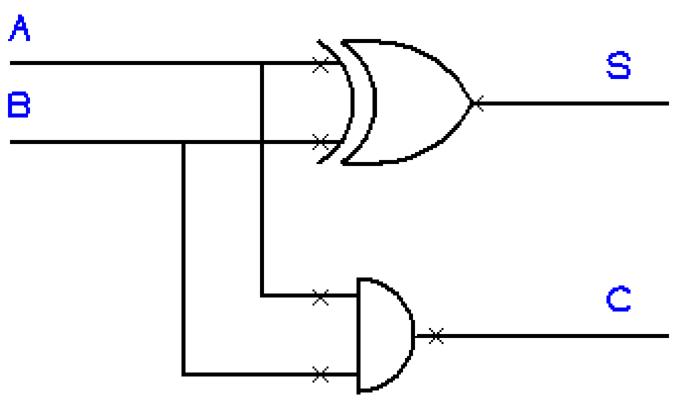


*Figura 2 Schema di un Half-Adder.*

[1] Il simbolo ⊕ indica l'operatore XOR. Il simbolo · indica l'operatore AND.

Adesso vogliamo realizzare un dispositivo in grado di effettuare la somma di due operandi costituiti da più bit. Questa operazione va eseguita *bit per bit* a partire da quello meno significativo. Abbiamo bisogno di un dispositivo con tre ingressi per tener conto dell'eventuale riporto (Carry) e due uscite (la somma e il riporto).

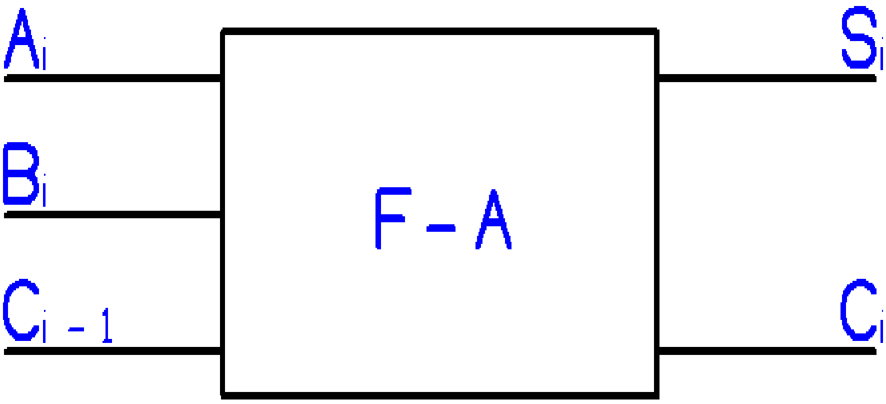


*Figura 3Full-Adder*

Scriviamo la tabella della verità:

| $A_i$ | $B_i$ | $C_{i-1}$ | $S_i$ | $C_i$ |
|---|---|---|---|---|
| 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 |

Scriviamo l'espressione booleana delle due funzioni $S_i$ e $C_i$. Esistono vari modi per ricavare l'espressione booleana di una funzione a partire dalla tabella della verità. Qui uso il metodo che costruisce l'espressione booleana in forma di OR (simbolo +) di tanti termini quante sono le combinazioni dei valori degli ingressi in corrispondenza dei quali la funzione assume il valore 1. Per ciascuna di queste combinazioni il corrispondente termine è costituito dall'AND (simbolo ·) di tutte le variabili presenti in ingresso negate oppure no a seconda che, nella particolare combinazione considerata, esse compaiano con il valore 0 oppure 1.
Consideriamo la funzione $S_i$.
Prima riga:

$$A_i = 0 \;\; B_i = 0 \;\; C_{i-1} = 0 \quad \rightarrow \quad S_i = 0$$

Questa combinazione non fornisce un contributo nell'espressione di $S_i$.
Seconda riga:

$$A_i = 0 \;\; B_i = 0 \;\; C_{i-1} = 1 \quad \rightarrow \quad S_i = 1$$

Questa espressione fornisce il termine:

$$\overline{A_i} \cdot \overline{B_i} \cdot C_{i-1}$$

Continuando con questo metodo si ottiene:

$$S_i = \overline{A_i}\cdot\overline{B_i}\cdot C_{i-1} + \overline{A_i}\cdot B_i\cdot\overline{C_{i-1}} + A_i\cdot\overline{B_i}\cdot\overline{C_{i-1}} + A_i\cdot B_i\cdot C_{i-1}$$

Semplifichiamo questa espressione. Raccolgo il fattore comune dai termini evidenziati:

$$S_i = (\overline{A_i}\cdot\overline{B_i} + A_i\cdot B_i)\cdot C_{i-1} + (\overline{A_i}\cdot B_i + A_i\cdot\overline{B_i})\cdot\overline{C_{i-1}}$$

Ricordando che:

$$\bar{A}\cdot\bar{B} + A\cdot B = \overline{A\oplus B} \quad e \quad \bar{A}\cdot B + A\cdot\bar{B} = A\oplus B$$

Troviamo:

$$S_i = \left(\overline{A_i\oplus B_i}\right)\cdot C_{i-1} + (A_i\oplus B_i)\cdot\overline{C_{i-1}}$$

Applicando nuovamente la stessa regola si ha:

$$S_i = (A\oplus B)\oplus C_{i-1}$$

Consideriamo adesso la finzione $C_i$:
Utilizzando il metodo vsto prima si trova:

$$C_i = \overline{A_i}\cdot B_i\cdot C_{i-1} + A_i\cdot\overline{B_i}\cdot C_{i-1} + A_i\cdot B_i\cdot\overline{C_{i-1}} + A_i\cdot B_i\cdot C_{i-1}$$

Raccogliamo i fattori comuni dei termini evidenziati:

$$C_i = (\overline{A_i}\cdot B_i + A_i\cdot\overline{B_i})\cdot C_{i-1} + A_i\cdot B_i\cdot(\overline{C_{i-1}} + C_{i-1})$$

Ma:

$$\overline{C_{i-1}} + C_{i-1} = 1$$

Quindi:

$$C_i = (A_i\oplus B_i)\cdot C_{i-1} + A_i\cdot B_i$$

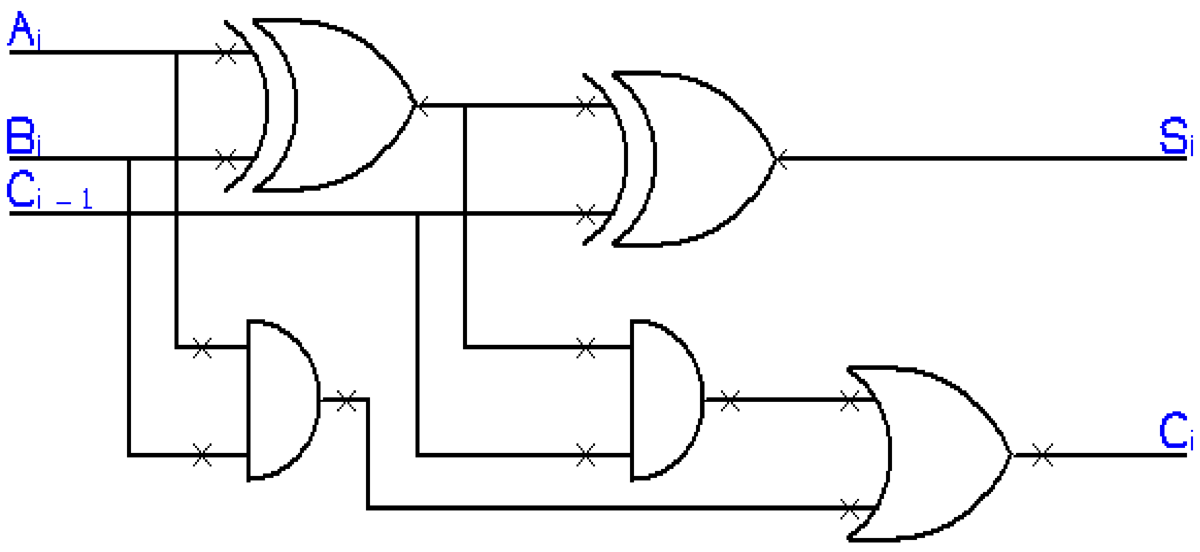


*Figura 4 Schema di un Full-Adder*

Per sommare operandi costituiti da più di un bit dobbiamo usare un Full-Adder per ogni coppia di bit. Si ottiene un sommatore binario (Binary Adder).

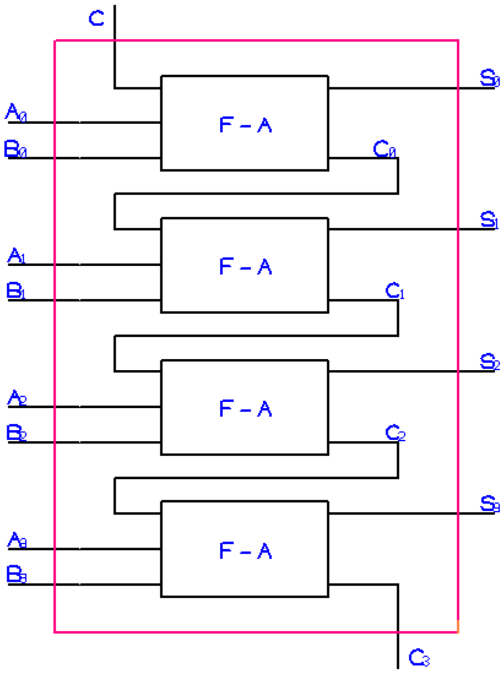


*Figura 5 Un sommatore binario a 4 bit.*

Questo file può essere scaricato gratuitamente. Se pubblicato citare la fonte.

Matilde Consales