



CIRCUITI LOGICI 1

Definizioni

1. **Variabile logica** (o binaria): è una variabile che può assumere solo due valori, 1 o 0.

Sinonimi di 1 e 0 sono V (vero) e F (falso)
Si usa la variabile logica per rappresentare il funzionamento di dispositivi con solo 2 stati significativi, come per esempio una lampada (accesa/spenta), un interruttore (aperto/chiuso).

Definizioni

- 2. Proposizione logica:** è una frase che può essere solo vera o falsa : “l’interruttore è chiuso”, “il motore è acceso”, “l’ombrello è aperto” ecc.
- 3. Operatori logici:** permettono di collegare le proposizioni logiche per costruirne di più complesse;
i principali operatori logici sono il **NOT** (negazione), l'**AND** (prodotto logico), e l'**OR** (somma logica)

Proposizioni collegate da operatori logici

4. “*La chitarra non (NOT) è accordata*” (assume valore vero quando la frase senza il non è falsa)
5. “*fuori è buio e (AND) la luce è accesa*” (assume valore vero quando sono vere entrambe le proposizioni)
6. “*è presente il titolare o (OR) è presente un incaricato*” (assume valore vero quando almeno una delle proposizioni è vera).

Definizioni

7. **Funzione logica:** esprime il legame tra una variabile binaria detta **dipendente** e altre variabili binarie dette **indipendenti**.

“la bancarella rimane aperta nei giorni feriali, se non piove , di giorno”

Lo stato della bancarella rappresenta la variabile dipendente, mentre le tre variabili feriale/festivo, pioggia/non pioggia, giorno/notte sono quelle indipendenti

Definizioni

8. **Algebra di Boole** : è l'insieme degli strumenti matematici che consentono di rappresentare le funzioni logiche sotto forma di espressioni algebriche e semplificarle mediante opportune regole.

Nell'algebra di Boole si rappresentano le **variabili logiche** con lettere maiuscole (**ABC**), l'operatore **NOT** con un trattino posto sopra la lettera (\bar{A}), l'operatore **AND** con il prodotto tra le variabili ($A \cdot B =$ prodotto logico), l'operatore **OR** con la somma tra le variabili ($A + B =$ somma logica)

Dalla funzione logica all'espressione algebrica

“accendo la lampada quando è buio e c'è qualcuno nella stanza”

variabile dipendente **Y** : stato della lampada (accesa=1, spenta=0);

variabili indipendenti **A, B** :

- la luminosità **A** della stanza (buio=1, luce=0)
- la presenza di persone **B** (presenza=1 assenza=0) ;

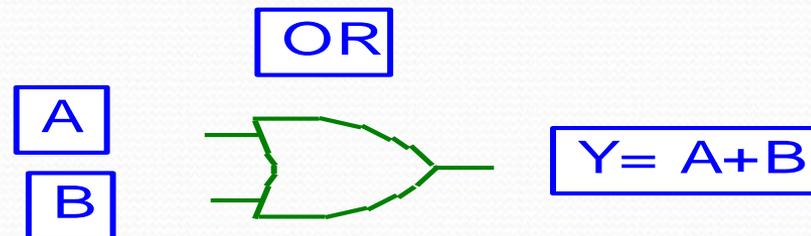
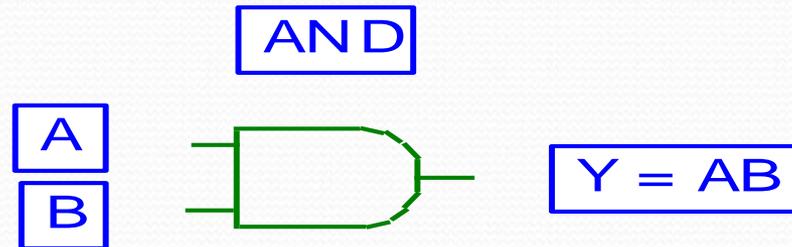
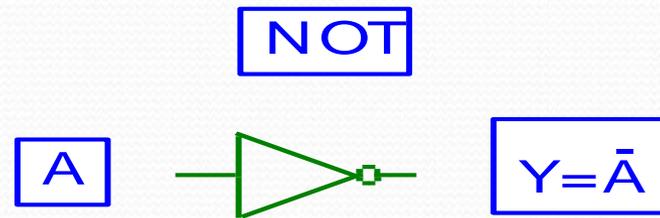
il legame logico è costituito dalla congiunzione **e** (operatore **AND**)

quindi : **$Y = A \cdot B$**

Definizioni

9. **Porte logiche (logic gates)** : sono la forma elettronica degli operatori dell'algebra di Boole, ricevono in ingresso segnali elettrici digitali (tensioni alte o basse) e forniscono in uscita il valore di tensione associato al risultato dell'operazione

Simboli grafici delle porte logiche elementari



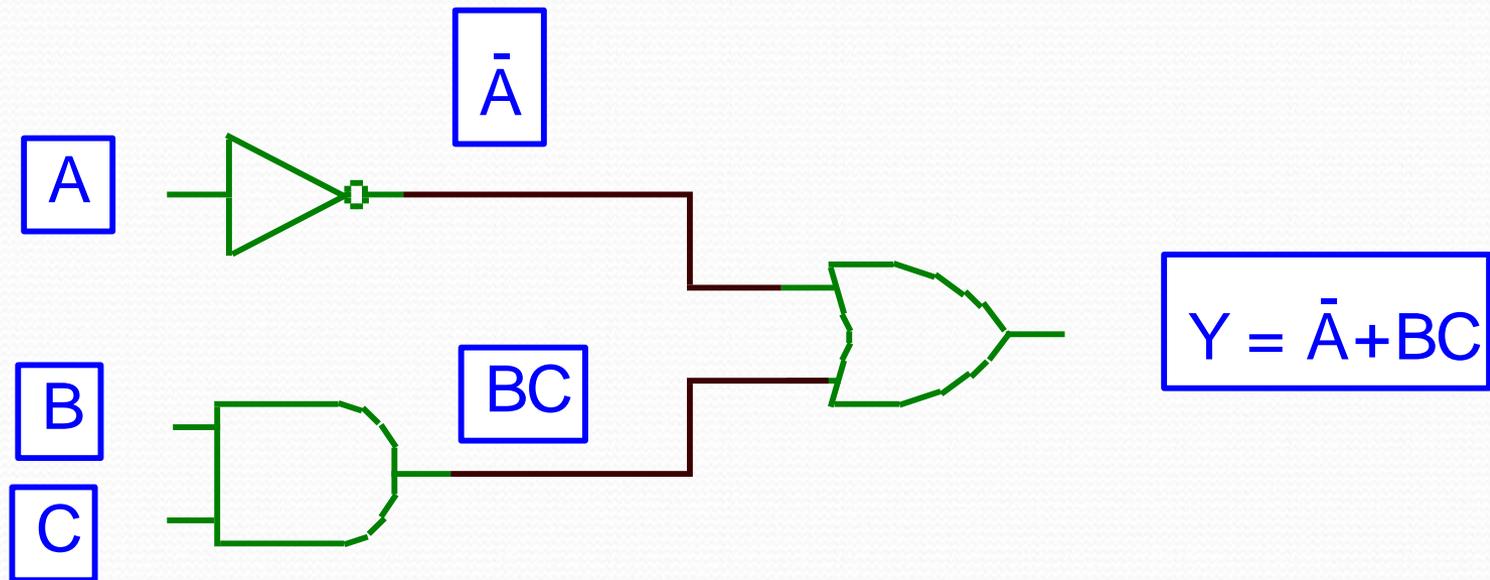
Definizioni

10. Rete logica: è un circuito costituito da porte logiche collegate tra loro

11. Rete logica combinatoria : rete digitale in cui i valori delle uscite in un certo istante di tempo, dipendono solo dai valori assunti dagli ingressi nel medesimo istante e non dai valori precedenti (rete priva di memoria)

12. Rete sequenziale: rete digitale in cui i valori delle uscite dipendono anche dai valori precedenti degli ingressi (reti con memoria ; contengono latch e flip-flop)

Esempio di rete logica



Definizioni

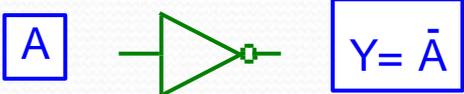
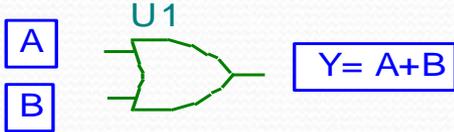
13. Funzione di commutazione: è l'espressione algebrica che lega una variabile d'uscita con gli ingressi di una rete logica.

14. Tavola di verità : esprime la relazione tra ingressi e uscite di una rete, come la funzione di commutazione.

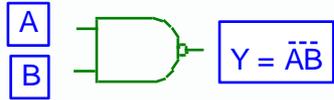
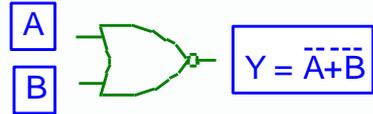
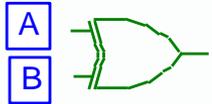
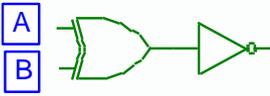
Nella colonna di sinistra si scrivono tutte le possibili combinazioni degli ingressi e in quella di destra i valori della variabile d'uscita.

Se la rete ha n ingressi la tabella è costituita da 2^n righe

Porte logiche fondamentali

Porta	Simbolo e funzione	Tavola Verità	Descrizione
NOT		$ \begin{array}{cc} A & Y \\ 0 & 1 \\ 1 & 0 \end{array} $	L'uscita ha valore complementare all'ingresso (negazione)
AND		$ \begin{array}{ccc} A & B & Y \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{array} $	L'uscita vale 1 solo se tutti gli ingressi valgono 1 (prodotto logico)
OR		$ \begin{array}{ccc} A & B & Y \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array} $	L'uscita vale 1 se almeno un ingresso vale 1 (somma logica)

Porte logiche fondamentali

Porta	Simbolo e funzione	Tabella	Descrizione															
NAND	 $Y = \overline{A \cdot B}$	<table style="border-collapse: collapse; margin-left: auto; margin-right: auto;"> <tr><td style="padding: 2px 10px;">A</td><td style="padding: 2px 10px;">B</td><td style="padding: 2px 10px;">Y</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> </table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0	L'uscita ha valore 0 solo e tutti gli ingressi sono a 1)
A	B	Y																
0	0	1																
0	1	1																
1	0	1																
1	1	0																
NOR	 $Y = \overline{A + B}$	<table style="border-collapse: collapse; margin-left: auto; margin-right: auto;"> <tr><td style="padding: 2px 10px;">A</td><td style="padding: 2px 10px;">B</td><td style="padding: 2px 10px;">Y</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	0	L'uscita vale 0 se almeno uno degli ingressi è a 1
A	B	Y																
0	0	1																
0	1	0																
1	0	0																
1	1	0																
EXOR	 $Y = A \oplus B = \overline{A} \cdot B + A \cdot \overline{B}$	<table style="border-collapse: collapse; margin-left: auto; margin-right: auto;"> <tr><td style="padding: 2px 10px;">A</td><td style="padding: 2px 10px;">B</td><td style="padding: 2px 10px;">Y</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	0	L'uscita vale 0 se i due ingressi sono uguali tra loro, vale 1 se sono diversi
A	B	Y																
0	0	0																
0	1	1																
1	0	1																
1	1	0																
EXNOR	 $Y = \overline{A \oplus B} = \overline{A} \cdot \overline{B} + A \cdot B$	<table style="border-collapse: collapse; margin-left: auto; margin-right: auto;"> <tr><td style="padding: 2px 10px;">A</td><td style="padding: 2px 10px;">B</td><td style="padding: 2px 10px;">Y</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td></tr> <tr><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">0</td><td style="padding: 2px 10px;">0</td></tr> <tr><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td><td style="padding: 2px 10px;">1</td></tr> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	1	L'uscita vale 1 se i due ingressi sono uguali tra loro, vale 0 se sono diversi
A	B	Y																
0	0	1																
0	1	0																
1	0	0																
1	1	1																