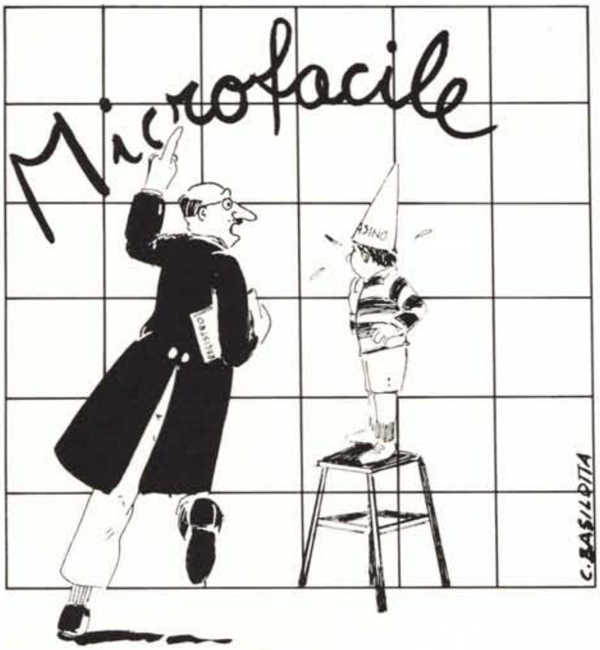
di Tommaso Pantuso



Famiglie logiche (2)

Basandoci sulle nozioni precedentemente apprese, questo mese vedremo che, fissando una operazione logica come fondamentale, è possibile ricavarne altre. Inoltre impareremo ad esprimere una funzione logica mediante una relazione algebrica.

Il Nand come operazione fondamentale

L'ultima operazione esaminata la volta scorsa è stata quella di Nand, ricavata combinando opportunamente l'operazione di And e di Not. Riprendiamo quest'ultimo punto e facciamo vedere come, praticamente — cioè impiegando delle porte logiche — sia possibile ricavare delle altre operazioni di notevole interesse nel campo dell'algebra della logica.

Abbiamo già accennato al modo in cui ottenere, utilizzando dei Nand, un And ed un Inverter (quest'ultimo, se ben ricordate, realizza l'operazione di Not). Nella figura 1 riportiamo lo schema di una porta Nand collegata in modo tale da implementare, in maniera alternativa a quella illustrata la volta scorsa, l'operazione di inversione. Consideriamo innanzitutto una relazione fondamentale a cui risponde l'operazione di And, ricordando ancora una volta che la sua tabella della verità è la seguente:

A	B	AND
0	0	0
1	0	0
0	1	0
1	1	1

La relazione è questa:

$$A \cdot 1 = A$$

dove con il simbolo "∗" è indicato l'And di A con 1. Praticamente, se osservate la figura 1b) e le ultime due combinazioni riportate nella tabella della verità (01 = 0 e 11 = 1), capirete ben presto il significato dell'operazione a cui viene sottoposta la variabile A. In altre parole, se prendiamo una variabile binaria ed eseguiamo l'And di quest'ultima con "1", in base alla risposta della tabella della verità otteniamo, come risultato, ancora il valore di A. Utilizzando una porta logica dotata di due ingressi e tenendo uno di essi costantemente in condizione 1, collegandolo ad esempio alla tensione di +5 volt, se poniamo sull'altro una certa condizione (0 o 1), essa verrà traslata in uscita senza essere modificata in alcun modo. Quindi, se in ingresso A è "0", l'uscita assumerà lo stato "0" e analogamente per l'"1". Facciamo un passo avanti. Se neghiamo entrambi i membri presenti nella relazione precedente, otteniamo:

$$\overline{A \cdot 1} = \overline{A}$$

e questa situazione è rappresentata nella figura 1a) dove troviamo lo stesso collegamento effettuato precedentemente, ma questa volta implementato su una porta Nand che opera la negazione della quantità A∗1. È facile rendersi conto che, in questo modo, avremo realizzato un circuito invertitore in quanto, qualunque variabile posta in ingresso, la ritroveremo "Negata" in uscita. Con quest'approccio al problema, abbiamo mostrato come mettere in pratica un risultato ottenuto per via teorica.

Passiamo ora alle nuove funzioni, cui accennavano poc'anzi, facendo vedere prima come esse vengono ricavate praticamente mediante l'utilizzo di porte logiche e dandone più avanti una giustificazione teorica.

Or e Nor

Osserviamo la figura 2 dove, per semplicità, abbiamo indicato gli inverter con il loro simbolo grafico, intendendo che esso può essere ricavato sfruttando uno dei modi di collegamento appresi per una porta Nand. Poniamo ora in ingresso le nostre variabili A e B e guardiamo insieme come esse cambiano aspetto durante il percorso e quale sarà lo stato dell'uscita Y.

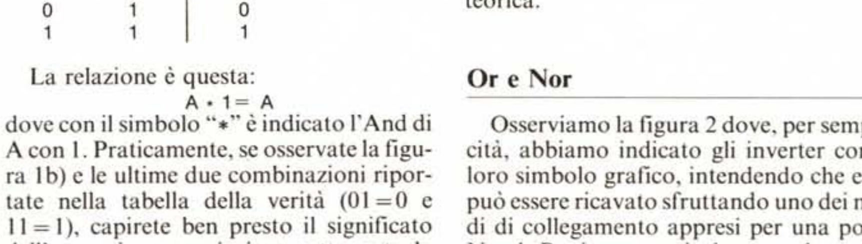


Figura 1 - a) proprietà del prodotto che sfruttata con un Nand (b) permette di realizzare un inverter.

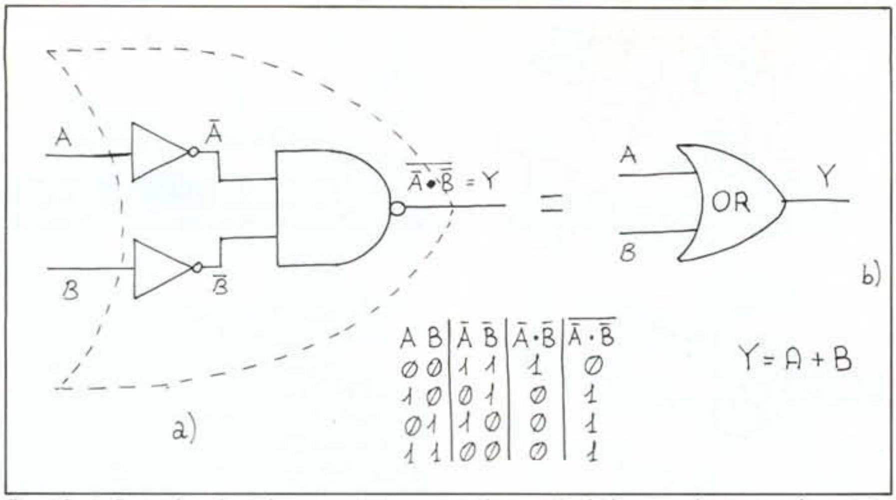


Figura 2 - a) Sintesi di un Or mediante un circuito costituito da porte Nand (da cui si può pensare anche costituito l'inverter). b) simbolo grafico con cui nei circuiti si rappresenta una porta Or.

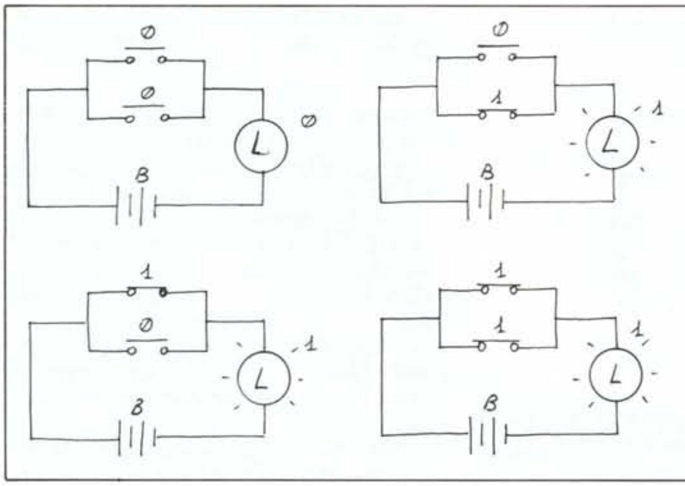


Figura 3
Esempio di circuito a pulsanti con il quale verificare l'operazione Or.

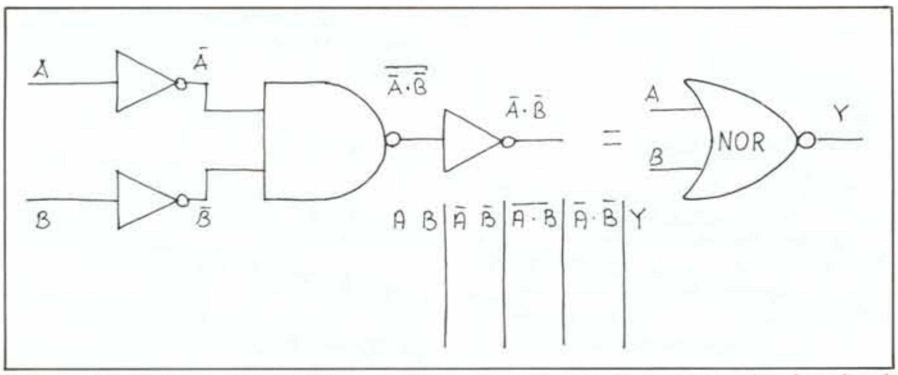


Figura 4 - Se nel circuito della figura 2a) poniamo un invertitore sull'uscita, ne otteniamo un altro che realizza la funzione di Nor. Il simbolo grafico del Nor è riportato sulla destra.

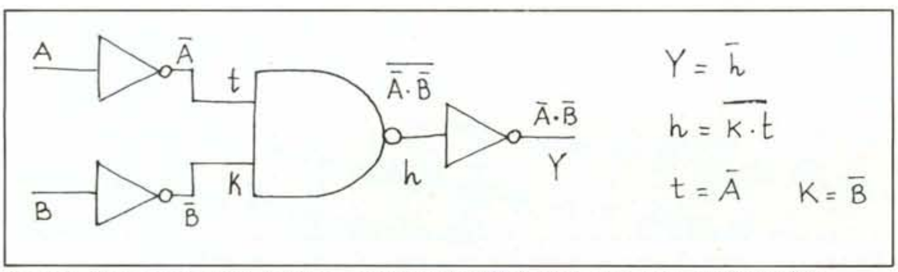


Figura 5 - Un metodo per ricavare un'espressione algebrica della funzione svolta da un circuito qualsiasi.

Per prima cosa, A e B passano attraverso due invertitori che eseguono su di esse un'operazione di Not invertendone lo stato: sulle loro uscite troveremo allora rispettivamente $\bar{A}$ e $\bar{B}$. A questo punto, le due quantità vengono introdotte in una porta Nand che ne fa il "prodotto" e lo nega. Riassumendo, la condizione che assumerà l'uscita Y in funzione delle quantità poste in ingresso sarà

$$\overline{\bar{A}\cdot\bar{B}}.$$

Da questa o dall'osservazione della figura 2, ricaviamo la tabella della verità che riportiamo di seguito:

A	B	OR
0	0	0
1	0	1
0	1	1
1	1	1

Questa tavola riassuntiva è quella che identifica l'operazione di Or (inclusivo) o disgiunzione logica: d'ora in poi la indicheremo con il simbolo "+". Una porta logica che realizza praticamente in un circuito l'operazione di Or ha per simbolo grafico quello riportato nella sezione b) della figura 2. Per fare invece il solito esempio con i pulsanti, riferendoci alla figura 3, osserviamo che l'Or può essere svolto da due pulsanti posti in parallelo nel circuito indicato: non è difficile vedere che, perché la lampadina sia accesa, basta che uno solo dei pulsanti sia premuto.

È facile intuire che da questa operazione se ne può ricavare un'altra, semplicemente invertendo l'uscita del circuito della figura 2. Otterremo allora quello della figura 4 in cui i simboli grafici vanno interpretati con le solite modalità. La tabella della verità che ne deriva, rappresenta, come detto, un'altra importante operazione, quella di Nor, ed è la seguente:

A	B	NOR
0	0	1
1	0	0
0	1	0
1	1	0

Relazioni algebriche

Fino a questo momento abbiamo visto qual è la tabella della verità di un certo circuito più o meno complesso assegnando dei valori agli ingressi e andando a verificare lo stato dell'uscita.

Vogliamo ora soffermarci su un punto importante che riguarda i circuiti digitali esaminando il modo in cui, partendo da un semplice esame circuitale, sia possibile ricavare un'espressione algebrica che sintetizzi un comportamento.

Per rendere l'esempio più vicino a noi, partiamo dallo schema della figura 4 e modifichiamolo come in figura 5. In essa, alcuni punti del circuito sono contrassegnati da lettere minuscole che rappresentano il risultato delle operazioni svolte fino a quel punto. Dall'osservazione della figura, andando a ritroso dall'uscita Y verso gli ingressi, potremmo affermare intanto che:

$$Y = \overline{h} \qquad (1)$$

nello stesso modo, essendo h la combinazione degli ingressi del Nand, il quale opera sui risultati t e k, diremo che:

$$h = \overline{k \cdot t}\,; \qquad (2)$$

infine, è immediato verificare che

$$t = \overline{A} \qquad (3)$$

e

$$k = \overline{B} \qquad (4)$$

Sostituendo ora la (2) nella (1) otteniamo:

$$Y = \overline{\overline{k \cdot t}} = k \cdot t$$

ed inserendo in questa le espressioni di t e k —(1) e (2)— ricaviamo:

$$Y = \overline{A} \cdot \overline{B}$$

Se ora sostituiamo in questa espressione le varie combinazioni degli ingressi avremo la seguente tabella della verità

A B	$\overline{A}\ \overline{B}$	$\overline{A} \cdot \overline{B} = Y$
0 0	1 1	1
0 1	1 0	0
1 0	1 0	001
1 1	0 0	0

la quale altro non è che la tabella della verità di un Nor. Come è quindi facile concludere, la condizione dell'uscita di un circuito nel quale si combinano un certo numero di variabili può essere calcolata sfruttando una certa funzione algebrica assegnata.

Come ulteriore esempio, consideriamo il circuito della figura 6. Sfruttando le lettere indicate nei vari punti non è difficile osservare che valgono le seguenti relazioni:

$$Y = h \cdot k$$
$$h = t + g$$
$$k = z + x.$$

Tenendo ancora presente che:

$$t = \overline{A} \quad ; z = A$$

e

$$g = \overline{B} \quad ; x = B$$

con lo stesso procedimento precedente si ricava che:

$$Y = (A + B) \cdot (\overline{A} + \overline{B})$$

Avremo ancora modo di esaminare quest'ultimo circuito che prende il nome di Or esclusivo.

Riassumendo.

A questo punto possiamo fare un breve riepilogo di quanto appreso fin'ora.

Abbiamo visto, nel corso di questi primi articoli, come definire cinque operazioni molto importanti nell'algebra logica e quindi nei circuiti digitali. Mediante esse, viene data una risposta diversa alle combinazioni di due variabili poste in ingresso.

Le tabelle che seguono, sintetizzano queste operazioni:

A B	AND	NAND	OR	NOR
0 0	0	1	0	1
1 0	0	1	1	0
0 1	0	1	1	0
1 1	1	0	1	0

NOT	
0	1
1	0

che possono essere sintetizzate, come ormai sappiamo dalle seguenti espressioni algebriche:

And $Y = A \cdot B$
Nand $Y = \overline{A \cdot B}$
Or $Y = A + B$
Nor $Y = \overline{A + B}$
Not $Y = \overline{A}$

La sintesi di tutto quanto esaminato fin'ora è racchiusa in queste cinque relazioni alle quali noi siamo voluti giungere con un approccio abbastanza pratico. Esse sono di carattere generale nel senso che, ciascuna di esse, può operare su un numero illimitato di operandi (figura 7). L'operazione di And stabilisce che l'uscita è vera (1) solo se tutti gli ingressi sono veri. L'operazione di Nand afferma invece che l'uscita è vera solo fino a quando almeno uno degli ingressi è falso (0). Con l'Or, avremo un'uscita vera finché almeno uno degli ingressi sarà vero ed infine, con il Nor, l'uscita sarà vera solo se tutti gli ingressi sono falsi.

Abbiamo anche visto come, sfruttando opportune combinazioni di alcune delle operazioni descritte, sia possibile passare dall'una all'altra con molta semplicità. Ad esempio, combinando opportunamente la sola operazione di Nand sfruttando porte elementari che svolgano questa funzione, è possibile ricavare dei circuiti che svolgano le altre operazioni.

Sarebbe ora bello trovare una relazione che, in un certo senso, racchiuda un metodo per passare da un'operazione all'altra seguendo un procedimento analitico. Questa relazione esiste ed è proprio quella che esamineremo la prossima volta. Essa metterà le basi per la realizzazione, partendo da una tabella della verità o da una espressione algebrica, del corrispondente circuito applicativo.

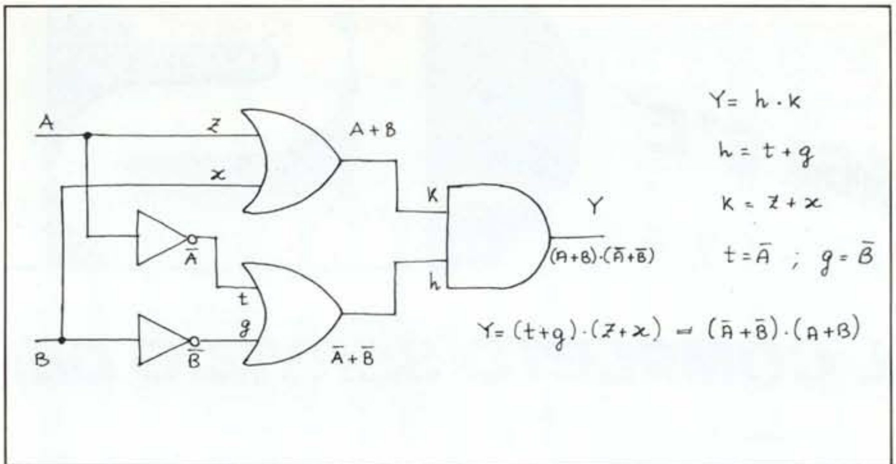


Figura 6 - Un altro esempio di sintesi di una espressione algebrica che rappresenta il comportamento del circuito.

Figura 7
Esempio di portata a più di due ingressi: Nand a tre ingressi.