Architettura degli Elaboratori
4 - Reti Combinatorie e Algebra di Boole

Ugo Dal Lago

Dipartimento di Scienze dell'Informazione
Università degli Studi di Bologna

Anno Accademico 2006/2007

Sommario

Il Livello Logico-Digitale

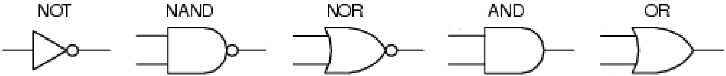
- Il ciclo di lezioni che andiamo ad iniziare riguarda il **livello logico-digitale**.
- Come abbiamo già detto in precedenza, in questo livello i programmi sono costitutiti da **circuiti digitali** (o **reti logiche**), i cui componenti fondamentali sono detti **porte logiche**.
- Esistono due classi di reti logiche:
  - Le **reti combinatorie**, che non hanno uno stato interno e che verranno studiate in questa parte del corso. Esempio: **ALU**.
  - Le **reti sequenziali**, che hanno uno stato interno e che verranno studiate nella prossima parte del corso. Esempio: **registri**.
- Il livello logico-digitale sta **alla base** delle gerarchia di macchine virtuali che abbiamo introdotto all'inizio del corso.
  - I circuiti sono entità concrete e tangibili, per cui non vanno tradotti o interpretati.
  - D'altra parte, il funzionamento interno delle porte logiche si può comprendere solo ad un livello inferiore, detto **livello dei dispositivi**.

Circuito Digitale

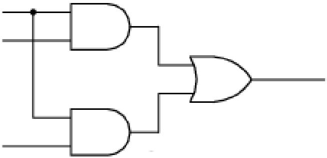
- Un **circuito digitale** è un'interconnessione di porte logiche.
- Nei fili che interconnettono le porte logiche in un circuito sono presenti solo due valori logici:
  - Il valore **0**, rappresentato da un segnale compreso tra 0 e 1 volt.
  - Il valore **1**, rappresentato da un segnale compreso tra 3 e 5 volt.
- Esistono almeno cinque tipi diversi di porte logiche: **NOT**, **NAND**, **NOR**, **AND**, **OR**.
- Ogni porta logica ha uno o più **ingressi** e un'**uscita**.
- Ogni circuito ha uno o più **ingressi** e una o più **uscite**.
- Gli ingressi di un circuito possono essere connessi:
  - Ad uno degli ingressi di una porta logica.
  - Ad una delle uscite del circuito.
- In un circuito, l'uscita di una porta logica può essere connessa:
  - Ad uno degli ingressi di un'altra porta logica.
  - Ad una delle uscite del circuito.

Notazione Grafica per i Circuiti Digitali

Porte Logiche



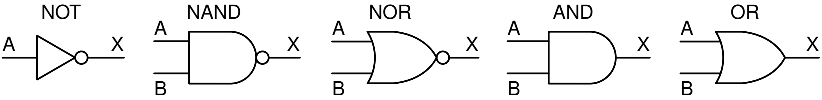
Circuito di Esempio



Reti Combinatorie

- Le reti combinatorie sono circuiti digitali che **non contengono cicli**.
  - In altre parole, non possono esistere cammini ciclici all'interno del circuito.
- Un **input** per il circuito consiste in una sequenza di valori binari, uno per ogni ingresso del circuito.
- Un **output** per un circuito consiste in una sequenza di valori binari, uno per ogni uscita del circuito.
- L'output di una rete combinatoria in corrispondenza ad un certo input si può calcolare come segue:
  - Per ogni porta logica, se sono noti i valori binari in corrispondenza dei suoi ingressi, allora sarà noto anche il valore binario in corrispondenza della sua uscita.
  - Le regole che permettono di determinare, per ogni porta logica, quale sia il valore della sua uscita in corrispondenza di ciascun valore per le entrate dipendono solo dal tipo di porta.
- Si può supporre, a questo punto, che il **tempo** necessario alla valorizzazione dell'uscita di una porta logica sia trascurabile.

Porte Logiche - Funzione Calcolata



A	**X**
0	1
1	0

(a)

A	**B**	**X**
0	0	1
0	1	1
1	0	1
1	1	0

(b)

A	**B**	**X**
0	0	1
0	1	0
1	0	0
1	1	0

(c)

A	**B**	**X**
0	0	0
0	1	0
1	0	0
1	1	1

(d)

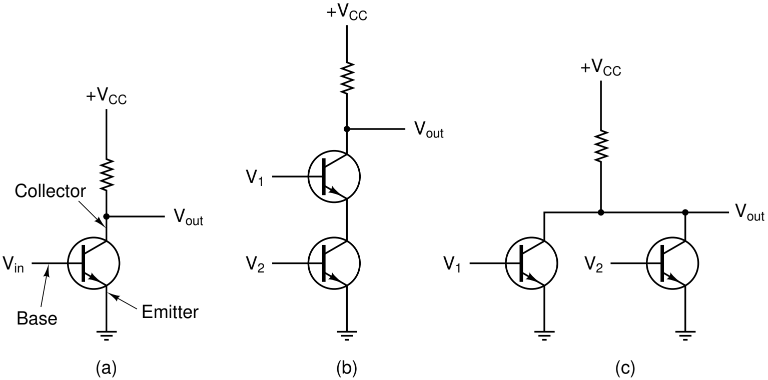
A	**B**	**X**
0	0	0
0	1	1
1	0	1
1	1	1

(e)

Come Realizzare le Porte Logiche?

- I dettagli sul **funzionamento interno** delle porte logiche esulano dagli scopi di questo corso.
- Una piccola digressione risulta però utile, a questo punto.
- Il **transistor** è un dispositivo elettronico che può funzionare come un velocissimo interruttore binario.
- I transistor hanno tre connessioni verso il mondo esterno: il **collettore**, la **base** e l'**emettitore**.
- Quando la tensione sulla base scende sotto il valore critico, il transistor si comporta come una **resistenza infinita**.
- Quando la tensione supera un valore critico, il transistor si comporta come un **conduttore ideale**.
- In questo modo è facilmente realizzabile una porta NOT. Due transistor sono invece sufficienti a costruire una porta NAND oppure una porta NOR.
- Esistono due tecnologie principali nella costruzione delle porte logiche:
  - La tecnologia **bipolare** (TTL, ECL,...).
  - La tecnologia **MOS** (PMOS, NMOS, CMOS,...).

Transistor: Esempi



L'Algebra di Boole

- Studieremo l'Algebra di Boole perché quest'ultima rappresenta un utile strumento per l'analisi e la sintesi delle reti combinatorie e sequenziali.
- Le **Algebre di Boole** sono strutture algebriche che soddisfano alcune proprietà.
  - Lo studio delle Algebre di Boole in generale richiederebbe troppo tempo.
- A noi interessa una particolare Algebra di Boole, che consiste nell'insieme $B = \{0, 1\}$ e in tre operazioni su questo insieme:
  - L'operazione binaria di **addizione**, indicata con $+$ oppure con OR. Valgono le seguenti identità: $0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$ e $1 + 1 = 1$.
  - L'operazione binaria di **moltiplicazione**, indicata con $\cdot$, con AND oppure tramite la semplice giustapposizione degli operandi. Valgono le seguenti identità: $0 \cdot 0 = 0$, $0 \cdot 1 = 0$, $1 \cdot 0 = 0$ e $1 \cdot 1 = 1$.
  - L'operazione unaria di **negazione**, indicata con il simbolo NOT oppure con una linea sopra il relativo operando. Valgono le seguenti identità: $\overline{0} = 1$ e $\overline{1} = 0$.

Proprietà dell'Algebra di Boole

Elemento neutro	$1A = A$	$0 + A = A$
Assorbimento	$0A = 0$	$1 + A = 1$
Idempotenza	$AA = A$	$A + A = A$
Complementazione	$A\overline{A} = 0$	$A + \overline{A} = 1$
Commutativa	$AB = BA$	$A + B = B + A$
Associativa	$(AB)C = A(BC)$	$(A + B) + C = A + (B + C)$
Distributiva	$A + BC = (A + B)(A + C)$	$A(B + C) = AB + AC$
Assorbimento	$A(A + B) = A$	$A + AB = A$
De Morgan	$\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}\,\overline{B}$
Doppia Negazione	$\overline{\overline{A}} = A$	

Espressioni Booleane

- Le **variabili booleane** sono $X, Y, Z, A, B, C, \ldots$. Si suppone che le variabili booleane prendano un valore in B.
  - Abbiamo appena utilizzato le variabili booleane nel dare le proprietà dell'Algebra di Boole, che valgono qualunque sia il valore sostituito a A, B o C.
- A partire dalle costanti 0 e 1 e dalle variabili booleane, possiamo costruire **espressioni booleane** utilizzando le operazioni di addizione, moltiplicazione e negazione.
  - Esempi di espressioni booleane sono $A + BC$, $\overline{A + B + \overline{C}} + C$, etc.
- Una volta assegnato un valore di verità in B ad ognuna delle variabili in un espressione booleana, è possibile valutare l'espressione in corrispondenza di tale assegnamento.
- Il valore di un'espressione in corrispondenza di tutti i possibili assegnamenti è riassunto nella sua **tabella di verità**.
- Due espressioni booleane (contenenti le stesse variabili booelane) si dicono **equivalenti** quando il valore delle rispettive tabelle di verità coincide in ogni riga.

Tabelle di Verità: Esempi

$AB + AC$

A	B	C	AB	AC	$AB + AC$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	0	0	0
1	0	1	0	1	1
1	1	0	1	0	1
1	1	1	1	1	1

$A(B + C)$

A	B	C	$B + C$	$A(B + C)$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Manipolazioni Algebriche

- Un modo molto semplice per trasformare un'espressione booleana in un'altra espressione espressione ad essa equivalente consiste nell'impiego delle proprietà algebriche dell'algebra booleana.
  - Se si dimostra che esiste una catena di uguaglianze (indotte dalle proprietà dell'algebra booleana) che permettono di riscrivere un'espressione in un'altra, allora le due espressioni sono equivalenti.
- In questo modo è possibile, in particolare, semplificare un'espressione booleana data in un'espressione più semplice (ma equivalente).
- Esempi:

$$AB + AC = A(B + C)$$

$$\begin{aligned} \overline{A}BC + \overline{A}B\overline{C} + AC &= \overline{A}B(C + \overline{C}) + AC \\ &= \overline{A}B \cdot 1 + AC \\ &= \overline{A}B + AC \end{aligned}$$

Funzioni Booleane

- Date n variabili booleane $A_1, \ldots, A_n$, una **funzione booleana** associa un assegnamento di verità per una variabile C ad ogni assegnamento di verità alle variabili $A_1, \ldots, A_n$.
- Un modo esplicito per definire una funzione booleana consiste nel dare la sua tavola di verità, che elencherà il valore di C per ogni possibile valore di verità per $A_1, \ldots, A_n$.
- Esempio:

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

- Data una funzione booleana, il suo **complemento** è definito come la funzione booleana (sulle stesse variabili) che vale 0 quando la funzione di partenza vale 1 e, viceversa, vale 1 quando la funzione di partenza vale 0.

Dalle Espressioni alle Funzioni

- Ad ogni espressione booleana E può essere **associata** una funzione booleana F, che si indica spesso con

$$F = E$$

- Il valore della funzione F su un assegnamento di verità alle variabili booleane in E sarà semplicemente il valore di E in corrispondenza di tale assegnamento.
- Esempi:

$$F = A + BC$$
$$F = A(B + \overline{C}) + A\overline{B}$$

- In generale, espressioni booleane **diverse** possono corrispondere alla **stessa** funzione booleana.
  - Ad esempio, le tabella di verità di $F = AB + AC$ e di $F = A(B + C)$ sono uguali.
  - Più in generale, tutte le espressioni equivalenti corrispondono alla stessa funzione.

Dalle Funzioni alle Espressioni

- Un **letterale** è una variabile booleana oppure la negazione di una variabile booleana.
- Data una funzione booleana, si può **sempre** costruire una espressione booleana corrispondente (tra le tante possibili).
- In particolare, si può procedere considerando ogni riga della tabella di verità relativa alla funzione di partenza.
- Si associa **ad ogni riga** il prodotto di letterali ad essa corrispondente.
- L'espressione che cerchiamo sarà una somma di prodotti di letterali. Se la funzione vale 1 in una certa riga, allora il corrispondente prodotto di letterali farà parte della somma. Viceversa, se la funzione vale 0 in una certa riga, allora il corrispondente prodotto di letterali non farà parte della somma.
- In questo modo si possono ottenere espressioni booleane molto complesse...
  - ... che possono però essere semplificate tramite l'uso delle proprietà

Dalle Funzioni alle Espressioni: Esempio

- Consideriamo la funzione booleana introdotta in precedenza.
- In corrispondenza di ogni riga della tabella di verità, scriviamo il prodotto di letterali:

A	B	C	F	
0	0	0	0	$\overline{A}\,\overline{B}\,\overline{C}$
0	0	1	0	$\overline{A}\,\overline{B}\,C$
0	1	0	1	$\overline{A}\,B\,\overline{C}$
0	1	1	0	$\overline{A}\,B\,C$
1	0	0	0	$A\,\overline{B}\,\overline{C}$
1	0	1	0	$A\,\overline{B}\,C$
1	1	0	1	$A\,B\,\overline{C}$
1	1	1	1	$A\,B\,C$

- Le righe in cui F vale 1 sono la terza, la penultima e l'ultima. Possiamo quindi concludere che l'espressione booleana che ci interessa è

$$\overline{A}B\overline{C} + AB\overline{C} + ABC$$

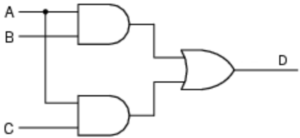
- Utilizzando le proprietà dell'algebra booleana otteniamo:

$$\begin{aligned} F &= \overline{A}B\overline{C} + AB\overline{C} + ABC = \overline{A}B\overline{C} + AB \\ &= B(\overline{A}\,\overline{C} + A) \end{aligned}$$

Dai Circuiti alle Funzioni

- Ad ogni uscita di ogni rete combinatoria corrisponde una funzione booleana.
  - Prima di tutto associamo n variabili booleane $A_1, \ldots, A_n$ agli n ingressi e una variabile C all'uscita che ci interessa
  - Per ogni assegnamento di valori di verità alle variabili booleane $A_1, \ldots, A_n$, il corrispondente valore di C sarà quello ottenuto valutando il circuito.
  - In questo modo, si possono associare funzioni booleane ad ogni uscita del circuito. Diremo che il circuito in questione **implementa** tali funzioni booleane.
- Due circuiti si dicono **equivalenti** se implementano la stessa funzione.
- Una volta costruita una funzione, si potrà poi ricavare un'espressione booleana

Dai Circuiti alle Funzioni: Esempio



$\Rightarrow$

A	B	C	D
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$\Downarrow$

$$A\overline{B}C + AB\overline{C} + ABC$$

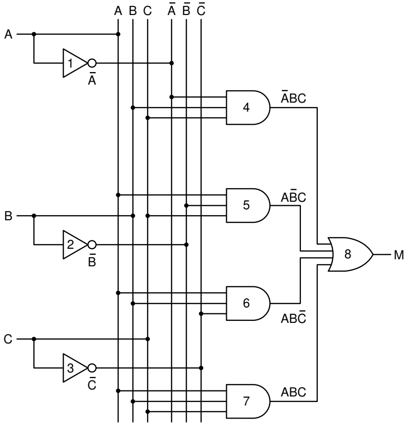
Dalle Espressioni ai Circuiti

- Ad ogni sequenza di espressioni booleane sulle stesse variabili corriponde un circuito.
  - Data un'espressione booleana E contenente le variabili booleane $A_1, \ldots, A_n$, un circuito con n entrate e un'uscita si può costruire seguendo la struttura di E.
  - La funzione implementata dal circuito sarà la funzione corrispondente ad E.
  - Partendo da una sequenza di espressioni, si potrà costruire un **singolo** circuito con più uscite.
- Per ottenere le espressioni booleane di partenza, si può procedere applicando ad altrettante funzioni booleane la procedura vista in precedenza.
  - In questo modo, si può passare da n funzioni booleane su m variabili ad un circuito con m ingressi ed n uscite.

Dalle Funzioni ai Circuiti: Esempio

A	B	C	M
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

(a)



(b)

$\Downarrow$ $\Uparrow$

$$\overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$