

UNIVERSITÀ DEGLI STUDI DI ROMA "LA SAPIENZA"
DIPARTIMENTO DI FISICA

Corso di "Laboratorio di Fisica" per studenti in Scienze Biologiche

NOZIONI DI SISTEMI DIGITALI

F.Meddi, C.Rossi

1 Introduzione alla logica combinatoria

Grazie all'algebra usuale si possono risolvere, per esempio, problemi formulati nella forma seguente: "qual'è quel numero che moltiplicato per 4 è pari alla somma fatta tra il suo quadrato ed il numero 4 stesso?". Basterà indicare questo numero incognito con x ed impostare una equazione algebrica.

$$4 \cdot x = x^2 + 4 \implies x = 2$$

Oltre a questa "algebra dei numeri" è stata sviluppata nel secolo scorso da G.Boole "l'algebra logica" che è in grado di manipolare delle "frasi dicotomiche". Con "frase dicotomica" si intende una affermazione che può risultare soltanto o "vera" o "falsa": non viene considerata perciò quella "indeterminata". Per esempio: "oggi Francesca è andata a cavallo" è una frase che potrà essere o vera o falsa, ma non "grigia".

A ciascuna "frase" si sostituiscono dei simboli compatti, quali le lettere (A, B, C,...) al solo scopo di una loro manipolazione piu' agevole. Queste sono le *variabili logiche*. Si introducono dunque due simboli con significato opposto da associare a queste variabili logiche corrispondenti alle due possibilità per le seguenti "frasi dicotomiche" :

frase vera \implies "1"

frase falsa \implies "0"

Tali simboli inoltre, nulla hanno a che fare nè con la numerazione di tipo decimale, nè con quella di tipo binario. Schematicamente si intende perciò che: A=1 equivale a dire che la frase associata al simbolo A, ovvero alla variabile logica A, è vera, mentre B=0 equivale a dire che la frase associata al simbolo B, ovvero alla frase logica B, è falsa.

Vediamo un esempio di un problema risolvibile in termini dell'algebra delle frasi di Boole. Si tratta del classico problema degli aperitivi:

- Ci sono tre persone : Aldo, Bruno e Carlo.

Se Aldo ordina un aperitivo lo ordinerà anche Bruno.

O Bruno o Carlo ordinano sempre un aperitivo, ma mai insieme.

Aldo o Carlo o entrambi ordinano sempre un aperitivo.

Se Carlo ordina un aperitivo lo ordina anche Aldo.

La questione è : se si trovano tutti e tre simultaneamente al bar, cosa succede? Vedremo in seguito come si può risolvere.

1.1 Funzioni di piu' variabili logiche e tabella della verità

Date n variabili logiche A, B, C... si può definire una funzione booleana o funzione logica indicata con $f(A, B, C..)$ di tali variabili mediante una tabella detta tabella della verità composta di tante righe quante sono le disposizioni con ripetizione di classe n dei due valori "0", "1" per le n variabili: cioè 2^n righe. Inoltre ad ogni riga viene associato il valore binario "0", "1" della funzione $f(A, B, C..)$ stessa nel modo richiesto dal particolare problema. Una volta stabilita la tabella della verità associata alla funzione booleana relativa a un dato problema, si può ricavare una funzione logica per la f in termini delle variabili logiche e di alcuni operatori logici che piu' avanti introdurremo. Questa espressione, finalmente, è suscettibile di essere tradotta in un circuito logico la cui uscita risulterà vera o falsa cosi' come la f della tabella della verità in corrispondenza delle stesse configurazioni di "1" o "0" per le n variabili logiche che intervengono nel problema.

Si sottintende, naturalmente, che si hanno a disposizione vari circuiti logici semplici detti porte logiche che manipolano le variabili logiche in ingresso fornendo un'uscita identica a quella che si ottiene applicando l'operatore logico corrispondente sulle variabili di ingresso.

1.2 Operatori logici

Introduciamo le definizioni di alcuni operatori logici elementari che agiscono sulle variabili logiche.

- Inversione logica o complementazione (NOT) $\bar{A} = C$

La frase C è la negazione della frase A, cioè la frase C è vera (C=1) se la frase A è falsa (A=0).

A	\bar{A}
0	1
1	0

Per esempio nella frase *"se questa sera pioverà non potrò osservare il cielo col telescopio"* si può individuare la negazione logica:

frase A: *"questa sera pioverà"*

frase B: *"osservazione astronomica"*

Se A=1 allora B=0 e viceversa se A=0 allora B=1.

- Prodotto logico (AND) $A \cdot B = C$

La frase C è vera (C=1) se lo sono simultaneamente sia A che B.

A	B	A·B
0	0	0
0	1	0
1	0	0
1	1	1

Per esempio, *"se domani ci sarà il sole e se Claudio avrà tempo, potremo andare a pescare insieme"*. Schematizziamo le frasi A e B :

frase A: *"domani ci sarà il sole"*

frase B: *"Claudio avrà tempo"*

Quindi solo se A=1 e B=1, la frase C=*"andare a pesca con Claudio"* risulterà vera (C=1); altrimenti C=0.

- Somma logica (OR) $A + B = C$

La frase C è vera (C=1) se lo è o la frase A o la frase B, includendo il caso che lo siano entrambe (somma logica inclusiva).

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

Per esempio *"solo se vinco al totocalcio o al gioco del lotto mi potrò comprare un'auto"*. Schematizziamo le frasi A e B :

frase A: *"vincita al totocalcio "*

frase B: *"vincita al lotto"*

frase C: *"acquisto auto "*

Dunque $C=1$ se $A=1$ oppure $B=1$ (per inciso: posso comprare la macchina anche se vinco sia al totocalcio, sia al lotto!!!)

Esiste un teorema generale che garantisce che utilizzando solo i tre operatori NOT, AND e OR si può comporre qualsiasi discorso logico: AND, NOT e OR sono cioè tre operatori logici fondamentali. D'altra parte, utilizzando unicamente operatori logici, che ora definiremo, di tipo "NAND" oppure "NOR" si può realizzare ciascuno dei tre operatori logici fondamentali.

• Implicazione "se" $A \rightarrow B$

Vediamo come si può rendere l'implicazione "se" in una frase logica in termini dei tre operatori fondamentali "NOT", "AND", "OR".

Se la frase A è vera, allora anche la B è vera : $A \rightarrow B$

La frase implicativa si può rendere nella forma algebrica seguente: $\bar{A} + B = 1$ verificabile scrivendo la tabella della verità.

A	B	$\bar{A} + B$
0	0	1
0	1	1
1	0	0
1	1	1

Nelle prime due righe, se $A = 0$ e $\bar{A} + B = 1$ allora B può essere o "0" o "1".

Nella terza riga, se $A = 1$ ma $\bar{A} + B = 0$ allora $B = 0$ (non c'è implicazione).

Infine nella quarta riga, se $A = 1$ e $\bar{A} + B = 1$ allora $B = 1$ (c'è implicazione). Infatti, se $A=1$, si avrebbe $\bar{1} + B = 1$, cioè DEVE essere $B=1$ affinché sia soddisfatta la relazione di implicazione "se".

• Operatore "NAND" $\overline{A \cdot B} = C$

La negazione del prodotto logico tra A e B dà l'operatore "NAND"

A	B	$A \cdot B$	$\overline{A \cdot B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

• Operatore "NOR" $\overline{A + B} = C$

La negazione della somma logica fra A e B dà l'operatore "NOR"

A	B	$A + B$	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

• Operatore "XOR" $A \oplus B = C$

Si tratta dell'operatore somma logica che esclude però il caso che siano vere entrambe simultaneamente le frasi (somma logica esclusiva): viene chiamato perciò "OR" esclusivo.

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

Sempre studiando la tabella della verità è verificabile anche che: $A \oplus B = A \cdot \overline{B} + \overline{A} \cdot B$

A	B	$A \cdot \overline{B}$	$\overline{A} \cdot B$	$A \cdot \overline{B} + \overline{A} \cdot B$	$A \oplus B$
0	0	0	0	0	0
0	1	0	1	1	1
1	0	1	0	1	1
1	1	0	0	0	0

Infatti, la colonna 6 corrisponde alla definizione data per l'operatore "XOR", mentre la 5 è costruita a partire dalle colonne 3 e 4.

1.3 Alcune proprietà dell'algebra logica

Si riportano ora alcune proprietà dell'algebra logica, verificabili facilmente mediante la tabella della verità.

- proprietà commutativa rispetto alla somma logica $\rightarrow A+B = B+A$
- proprietà commutativa rispetto al prodotto logico $\rightarrow A \cdot B = B \cdot A$
- proprietà associativa rispetto alla somma logica $\rightarrow (A+B)+C = A+(B+C)$
- proprietà associativa rispetto al prodotto logico $\rightarrow (A \cdot B) \cdot C = A \cdot (B \cdot C)$
- 1^a proprietà distributiva (simile a quella dell'algebra solita) $\rightarrow A \cdot (B+C) = A \cdot B + A \cdot C$
- 2^a proprietà distributiva (diversa da quella dell'algebra solita) $\rightarrow (A+B) \cdot (A+C) = A + (B \cdot C)$
- Teoremi di De Morgan :

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

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

Vediamo, per esempio, una verifica per i teoremi di De Morgan:

A	\overline{A}	B	\overline{B}	$A+B$	$\overline{(A+B)}$	$A \cdot B$	$\overline{(A \cdot B)}$	$\overline{A} \cdot \overline{B}$	$\overline{A} + \overline{B}$
0	1	0	1	0	1	0	1	1	1
0	1	1	0	1	0	0	1	0	1
1	0	0	1	1	0	0	1	0	1
1	0	1	0	1	0	1	0	0	0

Infatti : le colonne 6 e 9 coincidono (primo teorema di De Morgan) e così pure le colonne 8 e 10 (secondo teorema di De Morgan).

La trasformazione circuitale dei teoremi di De Morgan in termini di NOT, AND e OR è:

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

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

1.4 Considerazioni pratiche: circuiti logici integrati

L'algebra logica di Boole viene tradotta nella pratica circuitale tramite opportuni circuiti elettronici integrati che realizzano le funzioni logiche richieste mediante porte logiche appartenenti a una ben determinata famiglia logica di circuiti integrati, per esempio "TTL" (" Transistor Transistor Logic") cioè una logica basata sull'utilizzo di transistor al silicio di tipo bipolare.

Gli operatori logici visti finora sono disponibili in questa famiglia sotto forma di circuito integrato a 14 pin contenenti ciascuno 4 sezioni identiche di porte logiche a seconda del tipo.

La verità o falsità di una frase si tradurrà in due fasce di tensione separate: una prossima a +5V (essendo la tensione di alimentazione +5V) e l'altra a 0V (tensione di riferimento per l'alimentazione). In questo senso i valori delle tensioni non appartenenti a nessuno dei due intervalli menzionati sono da ritenersi dovuti a malfunzionamenti. Schematicamente:

Elenchiamo ora alcuni parametri che definiscono le porte logiche realizzate in tecnologia TTL.

Si definisce in generale "fan out" la capacità di un terminale di uscita di una porta logica di pilotare in parallelo simultaneamente più terminali di ingresso appartenenti ad un'unica o a più porte logiche relative ad integrati della stessa famiglia logica.

Nel caso "TTL":

a) "fan out" relativo al livello logico "1" $\geq \frac{400\mu A}{40\mu A} = 10$

b) "fan out" relativo al livello logico "0" $\geq \frac{16mA}{1.6mA} = 10$

Quindi nel caso della famiglia TTL si vede che il "fan out" vale 10. Infatti dalla figura si vede come lo stadio di uscita al livello logico "1" sia in grado di erogare al più $400\mu A$ (mantenendo il livello di tensione in uscita nella fascia permessa di 2.4-5.V), mentre il singolo stadio di ingresso richiede al più solo $40\mu A$ per riconoscere tale livello.

Schematicamente:

equivale a:

ovvero:

quindi per n carichi in parallelo:

$$\Delta V = \frac{R_i V_o}{n(R_u + R_i/n)} \quad \text{ovvero, come è ovvio, al crescere di } n \text{ diminuisce } \Delta V.$$

Altri parametri caratteristici sono:

- a) il tempo di propagazione tipico di un segnale logico fra ingresso e uscita della porta $\simeq 10\text{ns}$;
- b) la potenza media dissipata da una porta logica $\simeq 10\text{mW}$.

1.5 Espressione algebrica per una funzione logica

Dalla tabella della verità che rappresenta una funzione booleana "f", si può scrivere una espressione analoga a quelle usate nell'algebra tradizionale nel seguente modo:

1) Consideriamo le sole righe dove $f=1$.

2) Associamo a ciascuna di queste righe un termine, detto "prodotto fondamentale", che viene costruito dal prodotto logico delle variabili che compaiono nella tabella, prendendone la forma vera o negata, a seconda se valgono "1" o "0" nella riga considerata. Questi prodotti logici così costruiti, automaticamente valgono "1" nella riga a cui si riferiscono.

3) L'espressione algebrica di "f" sarà data dalla somma logica ("OR") di questi prodotti fondamentali relativi alle sole righe della tabella in cui "f" vale "1".

Per esempio:

A	B	C	f(A,B,C)	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\bar{A} \cdot B \cdot C$
1	0	0	0	
1	0	1	1	$A \cdot \bar{B} \cdot C$
1	1	0	1	$A \cdot B \cdot \bar{C}$
1	1	1	0	

Le prime tre colonne rappresentano tutte le possibili configurazioni degli stati logici delle tre variabili A,B,C; la quarta corrisponde a quanto richiesto dal problema schematizzato nell'esempio tramite la tabella della verità, mentre nell'ultima sono mostrati i soli prodotti fondamentali che intervengono. Dunque:

$$f(A, B, C) = \bar{A} \cdot B \cdot C + A \cdot \bar{B} \cdot C + A \cdot B \cdot \bar{C}$$

1.6 Alcuni problemi

1) Dato l'andamento nel tempo, sia delle variabili logiche in ingresso ad una rete combinatoria, che del segnale di uscita da questa, ricavare la funzione logica f realizzata dalla rete stessa, nonché la sua traduzione circuitale.

La strategia è risalire prima alla tabella della verità associata al diagramma temporale e da questa ricavare poi l'espressione esplicita di $f(A, B)$ per poter disegnare il circuito logico.

A	B	f(A,B)
0	0	1
0	1	0
1	0	0
1	1	1

Una espressione logica che soddisfi la tabella della verità è la seguente:

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

e la sua traduzione circuitale è la seguente:

Si tratta della cosiddetta "funzione equivalenza" che richiede di avere "1" solo quando entrambi gli ingressi sono allo stesso stato logico (cioè o entrambi "0", oppure entrambi "1"). In termini di frasi logiche, significa che solo se entrambe le due frasi sono vere o solo se entrambe sono false allora la frase risultante sarà vera.

2) Capire una data rete logica combinatoria dato il suo schema circuitale, ricavandone un'espressione esplicita per la funzione logica che è rappresentata dal circuito. La strategia può essere la seguente: Organizzare prima lo schema logico in modo che abbia ingressi a sinistra e si sviluppi verso destra.

Quindi individuare, per esempio numerando, le uscite delle porte combinatorie presenti nella rete, a parte l'ultima che sarà f .

Per ultimo, scrivere la f procedendo da destra verso sinistra, esplicitando di volta in volta, le variabili presenti.

Applichiamo queste regole allo schema seguente:

Possiamo così ricavare l'espressione della $f(A,B,C,D,E,F,G)$:

$$f = [A + B \cdot (C + D)] \cdot [\bar{B} + E \cdot (F + G)]$$

la sua tabella della verità sarà formata da $2^7 = 128$ righe essendo 7 le variabili logiche coinvolte

3) Un esempio di sintesi di una rete logica combinatoria che svolga una funzione di sicurezza:

Vogliamo impedire la partenza di un ascensore se si realizza almeno una delle condizioni seguenti: a) porta aperta; b) carico eccessivo. Naturalmente ci vuole il riconoscimento dell'avvenuta attivazione di un pulsante per la richiesta di partenza da parte dell'utilizzatore. Introduciamo le variabili logiche A, B, C:

A=1 porta chiusa

A=0 porta aperta

B=1 carico eccessivo

B=0 carico nella norma

C=1 pulsante attivato (ovvero richiesta di partenza dell'ascensore)

C=0 pulsante non attivato

La funzione richiesta sarà $f(A, B, C)$ con gli ovvi significati:

f=1 l'ascensore può partire

f=0 l'ascensore non può partire

Per prima cosa va scritta la tabella della verità associata al circuito di sicurezza per l'ascensore.

A	B	C	f(A,B,C)
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	0
1	1	1	0

La partenza (f=1) avviene solo se sono verificate tutte e tre le tre condizioni seguenti: porta chiusa (A=1), carico nella norma (B=0), richiesta di partenza (C=1). Una possibile espressione che soddisfi la tabella della verità è:

$f = A \cdot \bar{B} \cdot C$ e lo schema circuitale equivalente è il seguente:

L'espressione ricavata per la f può essere trasformata sfruttando De Morgan in modo che sia realizzabile usando solo circuiti logici di tipo "NOR"

$$A \cdot \bar{B} \cdot C = A \cdot \bar{B} \cdot C = \bar{\bar{A} + \bar{B} + \bar{C}} = \bar{\bar{A} + B + \bar{C}}$$

Ovvero, ricordando la tavola della verità della porta "NOR", si può ridisegnare il circuito nella forma:

4) Riprendiamo ora il problema degli aperitivi accennato nell'introduzione e proviamo a scrivere la tabella della verità. Introduciamo tre variabili logiche in corrispondenza delle frasi seguenti:

frase A : Aldo prende l'aperitivo

frase B : Bruno prende l'aperitivo

frase C : Carlo prende l'aperitivo

Le frasi di definizione del problema erano:

a) se $A=1$ allora $B=1$ quindi A implica B : $\overline{A} + B = 1$

b) o $B=1$ o $C=1$, mai $B=1$ e $C=1$: $B \oplus C = 1$

c) o $A=1$ o $C=1$ o anche $A=1$ e $C=1$: $A + C = 1$

d) se $C=1$ allora $A=1$ quindi C implica A : $\overline{C} + A = 1$

Abbiamo dunque 4 funzioni logiche che debbono essere soddisfatte simultaneamente:

a) $F_a = \overline{A} + B$

b) $F_b = B \cdot \overline{C} + \overline{B} \cdot C$

c) $F_c = A + C$

d) $F_d = \overline{C} + A$

A	B	C	F_a	F_b	F_c	F_d
0	0	0	1	0	0	1
0	0	1	1	1	1	0
0	1	0	1	1	0	1
0	1	1	1	1	1	0
1	0	0	0	0	1	1
1	0	1	0	1	1	1
1	1	0	1	1	1	1
1	1	1	1	0	1	1

Pertanto la soluzione del problema è data da:

$A=1$: Aldo prende l'aperitivo

$B=1$: Bruno prende l'aperitivo

$C=0$: Carlo non prende (mai, sigh !) l'aperitivo

ed la traduzione circuitale è quella indicata nella figura.

Solo per $A=1, B=1, C=0$ si avrà $F_a = 1, F_b = 1, F_c = 1, F_d = 1$

2 Introduzione alla logica sequenziale

Nei circuiti rappresentanti la logica combinatoria che abbiamo visto, lo stato logico dell'uscita dipende esclusivamente dalla configurazione delle variabili logiche presente ai suoi ingressi e si aggiorna secondo la tabella della verità continuamente, teoricamente in modo istantaneo.

Al contrario, nei circuiti rappresentanti la logica sequenziale le uscite vengono riportate in ingresso con opportuni "feedback" ed hanno la peculiarità di ricordarsi dello stato precedente. Lo stato finale sarà perciò dipendente non solo dalla configurazione delle variabili presente agli ingressi ma anche dalla storia degli stati precedenti. Da ciò, evidentemente, il nome di "sequenziale" attribuito a questi circuiti logici. Il più semplice circuito sequenziale è il MULTIVIBRATORE BISTABILE, detto anche "flip-flop" ("FF"). Con esso si intende un circuito logico caratterizzato dall'aver due stati stabili che si riflettono sulle due uscite presenti Q, \bar{Q} . Così nello stato "1" del "FF" si ha che l'uscita Q vale "1" e simmetricamente l'uscita \bar{Q} vale "0", il contrario nello stato "0" del "FF". Il passaggio da uno all'altro dei due stati può avvenire inviando un impulso all'ingresso detto di comando o di orologio o di "clock".

Nello schema tipico del "FF", oltre ai così detti "ingressi sincroni" rispetto al segnale di comando, vi sono delle linee di ingresso che vanno attivate in modo asincrono rispetto al segnale di comando stesso.

Queste linee permettono di forzare lo stato del "FF" in uno dei due possibili ("1" o "0" logico) indipendentemente dai segnali presenti a tutti gli altri ingressi, predisponendo (inizializzando) il "FF" prima dell'arrivo dei segnali di comando. Uno degli utilizzi di tali "FF" può essere quello di unità fondamentale di conteggio nei circuiti detti appunto contatori di impulsi, o anche come divisore di frequenza del segnale di comando stesso. D'altra parte, la presenza di due stati stabili nei "FF", permette di applicarli come elementi di memoria base negli elaboratori numerici, per esempio organizzandone otto per raccogliere l'informazione contenuta nel cosiddetto singolo "byte", o 16 per la "word" o 32 per la "long word".

Una configurazione utile di "FF" è quella detta "MASTER-SLAVE". In essa si hanno sostanzialmente due stadi di "FF" identici, disposti in cascata con la peculiarità che per esempio, se il primo "FF" è sensibile agli impulsi positivi del segnale di comando, allora il secondo lo sarà agli impulsi negativi del segnale di comando. In altre parole, il segnale di comando del primo "FF" viene complementato per divenire quello del secondo. In questo modo l'uscita dello stadio "MASTER" viene memorizzata nello stadio "SLAVE" nella seconda parte del periodo del segnale di comando stesso.

2.1 Circuito fondamentale di memoria in un sistema digitale

Sistema con reazione
chiamato "FLIP-FLOP" costituente una
cella di memoria ad 1 bit.

Come già detto, gli stati stabili possibili sono solo due:

stato "1": $Q = 1, \quad \bar{Q} = 0 \longrightarrow B = 1, \quad A = 0$
stato "0": $Q = 0, \quad \bar{Q} = 1 \longrightarrow B = 0, \quad A = 1$

Pertanto non è possibile la situazione in cui entrambe le uscite sono nel medesimo stato, cioè entrambe 0 o entrambe 1.

In pratica, perché il circuito precedente possa funzionare con i normali circuiti integrati della serie

"TTL", bisogna passare dalla porta ad un ingresso ad una con piu' ingressi. Si potrà perciò scrivere "1" o "0" nella cella di memoria sostituendo le due porte "NOT" con due porte "NAND" a due ingressi.

Nota: se $A=B=1$ si riottiene perfettamente la cella di memoria ad un bit precedente.

Qui viene riportato solo lo schema circuitale completo relativo al cosiddetto "FF" di tipo "set-reset" cadenzato, cioè con l'ingresso di "clock" tipico delle reti logiche sequenziali sincrone ed i due ingressi asincroni di preassegnazione dello stato del "FF" indicati con "PRESET" e "CLEAR"

2.2 Schema di principio di uno strumento di misura digitale per intervalli temporali (cronometro digitale)

In linea di principio possiamo pensare all'isocronismo delle piccole oscillazioni di periodo T_o di un pendolo meccanico semplice come ad un primo esempio di misura quantizzata di un intervallo temporale Δt in termini del numero n di intervalli elementari T_o ("digit"): $n \cdot T_o < \Delta t < (n + 1)T_o$

Evidentemente piu' è piccolo il "digit" temporale T_o , migliore risulta la misurazione di Δt . Invece di un oscillatore meccanico di periodo T_o pensiamo ora di avere un circuito che fornisce una successione di onde quadre (oscillatore) che vengono inviate all'ingresso di un circuito digitale costituito da vari "FLIP-FLOP" collegati fra loro in modo da contarle.

Immaginiamo che ad un certo istante gli stati di 4 "FLIP-FLOP" siano "1011". Questa configurazione può essere associata con la parola binaria "1011" corrispondente al numero decimale 11, in base ai pesi binari dei vari "digit" che, da destra a sinistra, sono : 1, 2, 4, 8, 16, 32..

Pertanto: $1 \times 1 + 1 \times 2 + 0 \times 4 + 1 \times 8 = 11$

All'arrivo dell'impulso successivo lo stato dei 4 "FF" cambia in "1100", corrispondente al decimale 12. Evidentemente, se riusciamo a condizionare l'arrivo degli impulsi tramite una porta logica avremo effettivamente il conteggio fra l'istante di inizio e quello di fine, istanti variabili a seconda delle nostre necessità. Vediamo un possibile schema a blocchi di principio per questo cronometro digitale.

In questo esempio il "display" mostrerà 5, cioè $\Delta t = 5 \times T_o$.

Alcuni dettagli ulteriori:

1. I segnali di inizio e fine misura sono ottenibili per esempio con interruttori di tipo meccanico, o meglio, di tipo elettronico.
2. IL generatore di impulso singolo per il segnale detto "GATE" può essere semplicemente costituito da un "FLIP-FLOP".
3. L'oscillatore libero si può pensare realizzato con un invertitore logico ("NOT") la cui uscita viene riportata all'ingresso attraverso un ritardo T_o , per esempio facendo propagare il segnale attraverso un cavo coassiale: tipicamente $\simeq 5\text{ns/m}$.
4. Il contatore a sua volta può essere di tipo asincrono con il segnale di CLOCK condizionato che pilota l'ingresso di comando del primo "FLIP-FLOP" detto "LSB", ovvero digit meno significativo; l'uscita Q del primo "FLIP-FLOP" pilota a sua volta l'ingresso di comando del secondo e così via in cascata fino all'ultimo "FF", detto "MSB", ovvero digit più significativo.
5. Il blocco denominato DISPLAY in realtà deve avere un primo stadio di conversione da BINARIO a DECIMALE in modo che la visualizzazione risulti comprensibile in modo veloce da tutti gli utenti.

2.3 Schema di principio di uno strumento di misura digitale per differenze di potenziale (voltmetro digitale)

La sua realizzazione più semplice prende il nome di "convertitore analogico-numericò a conteggio o a gradinata". In effetti il segnale (V_x) da convertire in digitale viene confrontato con uno di riferimento (V_{DAC}) generato internamente al voltmetro tramite un "convertitore di tipo digitale-analogico" ("DAC") e continuamente variato secondo una rampa prefissata fino a quando il confronto non dà esito positivo. Questa generazione a rampa viene ottenuta semplicemente inviando l'uscita di un oscillatore libero all'ingresso di un contatore. L'uscita di quest'ultimo va a formare la parola digitale da convertire in tensione analogica dal "DAC". Infine questa uscita del "DAC" (V_{DAC}) ed il segnale analogico da convertire (V_x) vengono inviati ai due ingressi di un comparatore il quale non è altro che un amplificatore differenziale ad alto guadagno, la cui uscita (V_{comp}) può essere in una prima fascia di tensione identificabile con "1" logico se $V_x - V_{DAC} > 0$ ovvero può essere in una seconda fascia di tensione identificabile con "0" logico se $V_x - V_{DAC} < 0$.

In tal modo, finché si ha "1" all'uscita del comparatore gli impulsi dell'oscillatore libero possono incrementare il contatore e quindi si permette alla rampa (V_{DAC}) di salire. Appena si ha "0" all'uscita del comparatore il contatore viene bloccato tramite l'utilizzo di una semplice porta "AND" e di conseguenza la salita della rampa si arresta.

Il valore raggiunto dal contatore è quello visualizzabile, cioè rappresenta la conversione in digitale del segnale V_x . È sottinteso che V_x non deve variare durante il processo di conversione del voltmetro. Questo si realizza o perché il segnale V_x è veramente variabile in maniera lenta o perché è stato, come si dice, "campionato" e memorizzato in modo che risulti costante durante il processo di conversione.

.

3 Introduzione ai calcolatori

I metodi numerici di trattamento dell'informazione giocano un ruolo sempre maggiore nella scienza e nella tecnica moderna.

Il rapido sviluppo di metodi numerici moderni è cominciato negli anni '40 quando furono costruite le prime macchine elettroniche di calcolo numerico con dei relais elettromeccanici, in seguito sostituiti con tubi elettronici.

Già negli anni '50 comparvero calcolatori elettronici della 2^a generazione costruiti con transistor anzichè con valvole. Grazie allo sviluppo della tecnologia di fabbricazione dei dispositivi a stato solido, si sono potuti integrare circuiti sempre piu' complessi diminuendo il numero di componenti e quindi anche i costi per la realizzazione dei calcolatori. Non ultimo vantaggio è anche la diminuzione del numero medio di guasti nell'unità di tempo migliorando così l'affidabilità del sistema.

I microcircuiti integrati sono entrati sempre piu' nei sistemi elettronici numerici e nei calcolatori elettronici della 3^a generazione che prende forma nel corso degli anni '60 e '70. È infatti del 1971 la comparsa del primo microprocessore da parte della ditta INTEL, costruito in tecnologia MOS a canale P appartenente alla famiglia dei cosiddetti circuiti a grande scala di integrazione ("LSI").

Verso la fine degli anni '70 vengono realizzati calcolatori elettronici della cosiddetta 4^a generazione basati prima sulla tecnologia di fabbricazione "LSI", e poi su quella a scala di integrazione molto grande ("VLSI").

La classificazione sul grado di integrazione funzionale che determina il livello di complessità dei microcircuiti integrati riguarda il numero complessivo dei componenti, per esempio in termini di porte logiche fondamentali equivalenti di tipo "NAND" che si trovano sulla singola piastrina di silicio e che formano il microcircuito. Si ha:

SSI : scala di integrazione bassa	≤ 10
MSI : scala di integrazione media	≤ 100
LSI : scala di integrazione grande	≤ 1000
VLSI : scala di integrazione molto grande	> 1000

Tornando ai calcolatori, in generale possiamo dire che il loro campo di applicazione è duplice: sia come strumento di calcolo vero e proprio, che come controllore di processo per esempio di tipo industriale. Quest'ultima applicazione gode degli evidenti vantaggi offerti dalla possibilità di implementare algoritmi di controllo anche complicati tramite programma e quindi di poter modificare il sistema, aggiornando il solo programma ("software") senza dover modificare i circuiti ("hardware").

In breve il trattamento dell'informazione rappresentata sotto forma binaria si effettua mediante l'esecuzione di una sequenza determinata di operazioni aritmetiche e logiche.

La tipica struttura di un sistema numerico generico è basata sulle seguenti parti:

1. UNITÀ DI INGRESSO-USCITA ("I/O")

Questa permette di introdurre e prelevare l'informazione, ovvero di scambiarla con il mondo esterno (stampanti, sensori,...).

2. UNITÀ DI OPERAZIONE ("ALU")

Questa realizza il trattamento dell'informazione introdotta dall'esterno; in generale si potrà trattare di operazioni sia di tipo puramente aritmetico, che di operazioni logiche.

3. UNITÀ DI MEMORIA ("M")

In questa sezione dovrà essere immagazzinata tutta l'informazione necessaria al funzionamento del sistema. Si tratterà sia dei dati da manipolare, che del programma per il loro trattamento, nonché dei risultati finali.

4. UNITÀ DI COMANDO ("C")

Questa sezione determina l'ordine di esecuzione delle singole operazioni che si devono succedere al fine di realizzare il trattamento voluto dei dati di ingresso secondo il programma formato dalle

istruzioni. Queste ultime contengono l'informazione sul tipo di operazione da fare (CODICE OPERATIVO) e su dove trovare e mettere i dati (INDIRIZZO della memoria).

L'insieme delle due parti, quella di comando e quella operativa viene detta PROCESSORE ed è il cuore di qualunque sistema numerico.

Va precisato un aspetto talvolta nascosto nell'uso di un calcolatore. L'insieme delle istruzioni contenute in un programma, sia quelle relativamente semplici, che quelle più complesse, richiedono ciascuna l'esecuzione di più operazioni elementari (MICROOPERAZIONI) conseguenti alla traduzione della singola istruzione in più (MICROISTRUZIONI). Dunque ad ogni istruzione va pensato in sostituzione un MICROPROGRAMMA composto da una successione di MICROISTRUZIONI che determinano la sua realizzazione effettiva.

3.1 Trasmissione dati digitali

In un sistema di acquisizione dati per l'elaborazione di questi tramite un calcolatore, le informazioni possono essere scambiate con le unita' periferiche sfruttando due tecniche diverse di trasmissione: quella parallela o quella seriale.

In breve nella trasmissione di tipo seriale i dati vengono scambiati "un bit alla volta" con una successione temporale ben definita, mentre in quella di tipo parallelo i bit vengono scambiati in gruppi, per esempio di 8 bit (chiamato byte). L'evidente vantaggio della comunicazione seriale è quello di richiedere un numero limitato di linee di collegamento, mentre quello della parallela è nella maggiore velocità di funzionamento.

Naturalmente al crescere della distanza, cioè della lunghezza dei cavi di collegamento, risulta più vantaggioso un collegamento seriale, che, seppure più lento dell'altro, consente un cablaggio decisamente più semplificato.

In linea generale entrambi i modi di trasmissione possono essere di due tipi: trasmissione sincrona o asincrona. Nella sincrona viene trasmesso, simultaneamente all'informazione, su di un cavo di collegamento separato, un segnale di 'clock' al fine di sincronizzare trasmettitore e ricevitore tra di loro, obbligando così il trasferimento dei dati ad avvenire ad intervalli regolari di tempo.

Nel caso della trasmissione seriale di tipo asincrono, non essendoci la trasmissione di un segnale di "clock", la sincronizzazione si realizza in due passi:

a) per prima cosa sia il trasmettitore che il ricevitore devono lavorare con "clock" della stessa frequenza (cioè avere lo stesso "baud rate" o numero di bit trasmessi in un secondo, per esempio 9600); b) affinché questi siano in fase fra loro bisogna fare in modo che il ricevitore "sappia" quando ha inizio la trasmissione in modo da sincronizzare la lettura dei vari bit. Ciascun carattere in trasmissione è preceduto da quello che viene chiamato "bit di start" seguito dal gruppo di bit di dato e terminato da uno o più "bit di stop". Il riconoscimento del bit di start da parte del ricevitore permette a questo di sincronizzarsi con il trasmettitore. Terminata la trasmissione del singolo pacchetto, l'intervallo di tempo che passa prima dell'invio del pacchetto di bit successivo è completamente libero, con conseguente nuova perdita di sincronizzazione. Questa verrà ristabilita al momento del riconoscimento del "bit di start" del pacchetto successivo.

L'aver dovuto introdurre nel pacchetto di "bit di dato" trasmessi, dei bit di sincronizzazione, porta ad aumentare, a parità di informazione scambiata, il numero totale di bit da trasmettere con conseguente maggiore lentezza della trasmissione stessa.

Un esempio di questo tipo è lo standard seriale EIA RS232-C (Electronic Industries Associated), utilizzato diffusamente per trasmettere informazioni ai terminali video.

Nel caso della trasmissione parallela asincrona c'è la necessità che il trasmettitore "avverta" il ricevitore di avere posto l'informazione sulle linee parallele e quella che il ricevitore "risponda" al trasmettitore, informandolo di avere acquisito il dato e di essere "disponibile" ad acquisire il successivo. Il problema si risolve con quello che si chiama "hand shake" (stretta di mano). In altre parole il trasferimento dei dati avviene quando il trasmettitore e il ricevitore sono informati sull'andamento della trasmissione. Dunque in un collegamento di questo tipo, oltre alle linee di dati c'è bisogno di altre due linee dedicate che svolgano questa funzione di informare "l'interlocutore" dell'esito dell'operazione di trasferimento.

Esempio di questo tipo sono lo standard parallelo CENTRONIX (utilizzato nelle stampanti) e quello denominato IEEE 488 (Institute of Electrical and Electronic Engineers), inizialmente sviluppato dalla Hewlett-Packard con il nome di HP-IB e utilizzato nella realizzazione di strumenti di misura.

1. INTRODUZIONE ALLA LOGICA COMBINATORIA

FUNZIONI DI VARIABILI LOGICHE E TABELLA DELLA VERITA'

OPERATORI LOGICI

PROPRIETA' DELL'ALGEBRA LOGICA

CONSIDERAZIONI PRATICHE: CIRCUITI LOGICI INTEGRATI

ESPRESSIONE ALGEBRICA PER UNA FUNZIONE LOGICA

PROBLEMI

2. INTRODUZIONE ALLA LOGICA SEQUENZIALE

CIRCUITO FONDAMENTALE DI MEMORIA PER UN SISTEMA DIGITALE

CRONOMETRO DIGITALE

VOLTMETRO DIGITALE

3. INTRODUZIONE AI CALCOLATORI

TRASMISSIONE DATI DIGITALI