Corso di Informatica Generale per Fisica

Elementi (minimi) di struttura degli elaboratori elettronici

Mauro Brunato

24 febbraio 2009

Sommario

Scopo di questa dispensa è quello di formare un'idea di come funziona un elaboratore elettronico partendo *dal basso*, ovvero dai principi fondamentali e dalle sue componenti di base (in altre parole, si spiega come costruire un calcolatore con alcuni semplici pezzi di metallo e *molto* filo di rame).

Questa dispensa è soggetta a variazioni e aggiustamenti durante il corso del bimestre di insegnamento; l'ultima versione piò sempre essere recuperata dalla sezione "risorse" della pagina del corso di Informatica Generale per Fisica sul sito

`http://dit.unitn.it/~brunato/info1/`

Per distinguere le versioni è sufficiente confrontare la data riportata sotto il titolo con quella scritta nell'elenco delle risorse (non è dunque necessario scaricare il file tutte le volte).

Indice

Capitolo 1

L’aritmetica binaria

Un calcolatore utilizza, per la rappresentazione interna dei numeri, l’aritmetica *binaria*, ovvero in base 2. Questa base è stata scelta perché richiede l’uso di due soli simboli, 0 e 1.

Infatti un calcolatore utilizza grandezze *elettriche* (come la tensione o l’intensità di corrente elettrica) per rappresentare le cifre di un numero, e le grandezze elettriche tendono a risentire di disturbi di vario genere; di conseguenza, usare dieci livelli di tensione diversi per rappresentare le dieci cifre del sistema decimale non sarebbe pratico e si presterebbe a errori di “sconfinamento” tra una cifra e l’altra (se un 6 è rappresentato da una tensione di 6V e un 7 da una di 7V, è facile che un po’ di rumore elettromagnetico trasformi casualmente una cifra nell’altra).

Due soli simboli possono essere rappresentati dai valori di tensione più piccolo e più grande disponibili nel sistema, quindi sono più separati e meno confondibili anche in presenza di forti interferenze elettromagnetiche.

Ricordiamo che il sistema decimale è di tipo posizionale: ogni cifra ha un peso determinato dalla sua posizione nel numero

$$1492_{10} = 2 \cdot 10^0 + 9 \cdot 10^1 + 4 \cdot 10^2 + 1 \cdot 10^3.$$

In caso di ambiguità, indichiamo la base di rappresentazione con un piccolo pedice. Allo stesso modo, ecco l’interpretazione di un numero in base 2:

$$100101_2 = 1 \cdot 2^0 + 0 \cdot 2^1 + 1 \cdot 2^2 + 0 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5 = 37_{10}.$$

Notare come la cifra meno significativa sia sempre, per convenzione, quella più a destra.

Una cifra binaria è detta “bit” (**bi**nary digi**t**); un gruppo di otto cifre binarie costituisce (solitamente) la più piccola unità di rappresentazione di un numero in un elaboratore, ed è detto “byte”. Naturalmente, se un numero richiede meno di otto bit è sufficiente “allungarlo” con degli zeri sul lato sinistro.

Il più piccolo numero rappresentabile con otto bit è, ovviamente, lo zero (in altro

37	1
18	0
9	1
4	0
2	0
1	1

Figura 1.1: Trasformazione di un numero decimale nella notazione binaria.

luogo si parlerà della rappresentazione di numeri negativi); il più grande è

$$11111111_2 = \sum_{i=0}^{7} 2^i = 2^8 - 1 = 255_{10}.$$

In generale, il più grande numero rappresentabile con n cifre binarie è

$$\sum_{i=0}^{n-1} 2^i = 2^n - 1.$$

Per convertire un numero decimale nell'equivalente rappresentazione binaria, è sufficiente seguire le seguenti "regole":

1. Scrivere il resto della divisione per 2 accanto al numero (traduzione: scrivere 1 se il numero è dispari, 0 se è pari)

2. Scrivere il quoziente della divisione per 2 sotto al numero

3. Se il quoziente non è nullo, riapplicare l'algoritmo a partire dal punto 1, altrimenti fermarsi.

In figura 1.1 è mostrata la conversione del numero 37_{10}. Notare che il risultato va letto dal basso verso l'alto.

1.1 Operazioni aritmetiche in base 2

Le regole per la somma di due numeri espressi in base 2 ricalcano quelle in base 10: si sommano le cifre corrispondenti dei due numeri, muovendosi da destra verso sinistra, aggiungendo l'eventuale riporto della colonna precedente. In questo caso, naturalmente, le cifre da sommare e il riporto possono essere soltanto 0 o 1. In tabella 1.1 sono elencati tutti i possibili casi.

Come esempio, riportiamo la somma dei due numeri 10001101_2 e 00101100_2:

$$\begin{array}{cccccccc c}
 & & & {\scriptstyle 1} & {\scriptstyle 1} & & & & \\
1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & + \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & = \\
\hline
1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 &
\end{array}$$

Tabella 1.1: Regole per la somma

Primo addendo	Secondo addendo	Riporto precedente	Somma	Riporto
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

Anche operazioni più complesse come la moltiplicazione e la divisione sono simili alle analoghe operazioni in base 10, ma più semplici per via del ridotto numero di possibili risultati parziali (la moltiplicazione si riduce a una serie di moltiplicazioni per 0 e per 1, seguite da una somma; nella divisione, si tratta di decidere se un numero sta in un altro zero o una volta...).

Capitolo 2

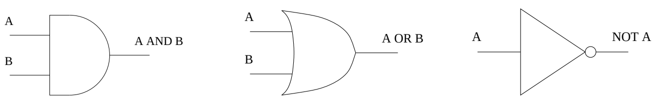
Operazioni e circuiti logici

Consideriamo l'operazione di moltiplicazione tra due cifre (binarie); vi sono quattro possibili combinazioni per i valori dei due operandi, e il risultato dell'operazione è 1 se e solo se entrambi gli operandi sono 1; La tabella che riporta il risultato dell'operazione per tutte le combinazioni degli operandi è riportata a sinistra in figura 2.1. Si noti che vi sono altre possibili interpretazioni per quest'operazione:

- Il connettivo logico di *congiunzione* fra due proposizioni, se si interpretano le cifre binarie come *valori di verità* (1 vuol dire vero, 0 falso); in altri termini, la proposizione è vera se e solo se entrambe le componenti sono vere. Per questa ragione, l'operazione è anche nota come AND (oppure con il simbolo logico "&") e in generale tutte le operazioni tra bit prendono il nome di operazioni "logiche".

- Il risultato è il minimo tra i due operandi, operazione denotata solitamente con il simbolo "$\wedge$".

Il circuito elettrico che realizza quest'operazione è denominato "porta AND", e il suo simbolo è riportato nella colonna di sinistra di figura 2.1. Si tratta di un cicuito elettrico la cui uscita ha valore 1 se e solo se entrambi gli ingressi hanno valore 1 (da



AND ($\wedge$, $\&$)

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

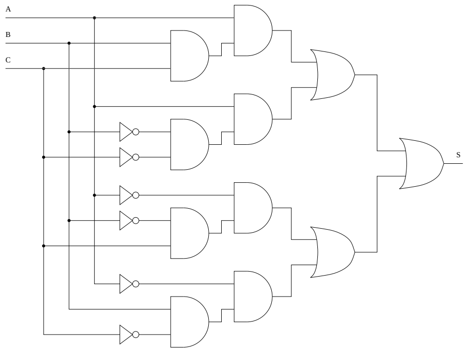
OR ($\vee$, $|$)

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

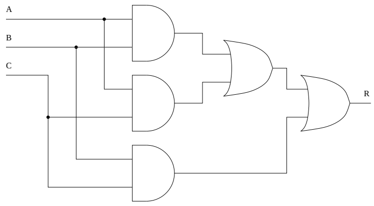
NOT ($\bar{\cdot}$)

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

Figura 2.1: Tavole di verità delle principali operazioni logiche.



(a) Somma di A, B e C



(b) Riporto della somma di A, B e C

Figura 2.2: Circuiti logici per il calcolo della somma e del riporto di cifre binarie.

un punto di vista elettronico, il “valore” di una linea di ingresso o di uscita dipende dalla modalità scelta per rappresentare le cifre binarie).

Un’altra operazione elementare è quella corrispondente alla disgiunzione logica (detta “OR” in elettronica digitale), ovvero al massimo tra i due operandi, rappresentata nella colonna centrale di figura 2.1.

Infine, l’operazione di inversione, o negazione (“NOT”), è riportata a destra.

2.1 Uso delle operazioni logiche

Le porte come la AND, la OR e la NOT, dette anche “porte logiche”, possono anche essere considerate delle rappresentazioni grafiche; come una formula si ottiene combinando delle operazioni elementari, così può essere rappresentata collegando tra loro con delle linee (che in pratica rappresentano collegamenti elettrici) le corrispondenti porte.

Ad esempio, la tavola di verità di figura 1.1 può essere sintetizzata in due formule, la prima per la somma e la seconda per il resto. Chiamiamo A e B i due operandi, e C il resto del passaggio precedente. È facile verificare che la somma (riportata nella penultima colonna della tavola 1.1) è data dalla formula

$$S = (A \wedge B \wedge C) \vee (A \wedge \bar{B} \wedge \bar{C}) \vee (\bar{A} \wedge B \wedge \bar{C}) \vee (\bar{A} \wedge \bar{B} \wedge C).$$

In un certo senso, la formula è un elenco di tutti i casi in cui il risultato dev’essere 1. Allo stesso modo, il resto (riportato nell’ultima colonna della tavola 1.1) è dato da

$$S = (A \wedge B) \vee (A \wedge C) \vee (B \wedge C).$$

Le stesse formule possono essere rappresentate graficamente come in figura 2.2, dove si adotta la convenzione che due fili collegati (rappresentati con un pallino nero nell’intersezione) rappresentano lo stesso valore (fisicamente sono equipotenziali), mentre un incrocio senza pallino nero sta a significare che non c’è contatto.

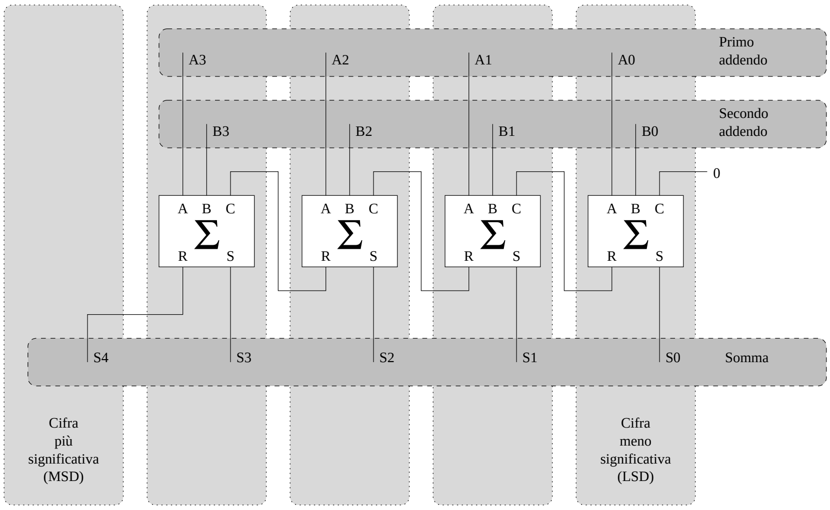


Figura 2.3: Combinazione di più circuiti di figura 2.2 per realizzare un sommatore a 4 bit.

Infine, la somma fra due numeri di quattro bit è riportata in figura 2.3. I rettangoli contrassegnati con la lettera Σ contengono i circuiti rappresentati in figura 2.2, e saranno chiamati nel seguito "sommatori elementari", in quanto ciascuno di essi esegue uno dei passaggi dell'addizione. Si noti che il riporto generato da ogni sommatore elementare (uscita R) viene collegato a uno degli ingressi del sommatore successivo (ingresso C). Fanno eccezione il l'ingresso C del sommatore elementare meno significativo, che non riceve nessun riporto ed è posto permanentemente a 0, e l'ultima uscita di riporto (quella del sommatore elementare più significativo), che viene usata come quinta cifra della somma (la somma di due numeri a quattro bit può generare, come accade in base 10, un numero a 5 bit).

Combinando opportunamente le tre porte logiche che abbiamo visto si ottengono circuiti che eseguono funzioni più complesse, delle quali il sommatore è un semplice esempio. Combinando tra loro questi circuiti complessi, si ottiene un elaboratore elettronico.

Capitolo 3

Intermezzo: come si realizza una porta logica

Quindi, il primo passo per capire “come si fa” un calcolatore elettronico è la realizzazione delle porte logiche AND, OR e NOT.

Supponiamo di disporre di un generatore di tensione (una batteria, ad esempio). Diremo che una linea elettrica vale 1 quando si trova a un potenziale positivo (in parole povere: è collegata al polo positivo di una batteria) rispetto a un valore di riferimento (il polo negativo della batteria). La linea varrà 0 quando non è collegata.

Possiamo anche dotarci di un sistema per “leggere” il valore di una linea elettrica: una lampadina collegata per un capo al polo negativo della batteria, e per l’altro capo alla linea in questione. Quando la linea elettrica è collegata al polo positivo, la lampadina si accende (figura 3.1).

Vediamo come realizzare una porta OR, ovvero un circuito con due linee elettriche in cui la lampadina si accende non appena una delle due linee è collegata al polo positivo. La soluzione è presentata in figura 3.2 a sinistra. È sufficiente collegare le due linee di ingresso (A e B) all’uscita, che va alla lampadina. È chiaro che basta collegare una delle due linee al polo positivo (cioè “porle a 1”) perché l’uscita vada a 1 (cioè la lampadina si accenda).

La porta AND è leggermente più complessa, in quanto la lampadina deve accendersi soltanso se *entrambe* le linee sono a 1. Il problema può essere risolto da un sistema elettromeccanico (parte centrale di figura 3.2) in cui l’ingresso A fa capo a un interruttore *aperto*, che cioè non conduce la corrente elettrica. L’interruttore è composto da una lamina ferromagnetica che può essere attirata da un elettromagnete collegato all’ingresso B. Perché la lampadina si accenda, è necessario dunque che l’elettromagnete sia attivato (ingresso B a 1), in modo che la lamina dell’interruttore lo chiuda, e che contemporaneamente l’ingresso A sia posto a 1.

La porta NOT si basa su un principio simile, solo che l’interruttore è normalmente chiuso (collega i suoi due estremi), quindi finché l’elettromagnete non è collegato la lampadina è accesa. Non appena colleghiamo a 1 l’ingresso A, l’elettromagnete attira la lamina dell’interruttore, scollegando la lampadina dalla batteria.

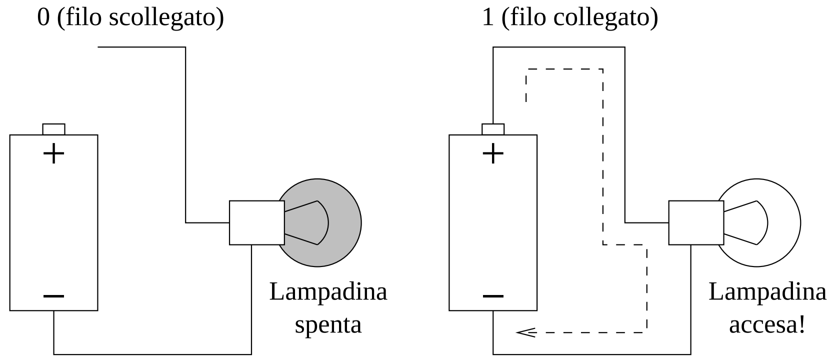
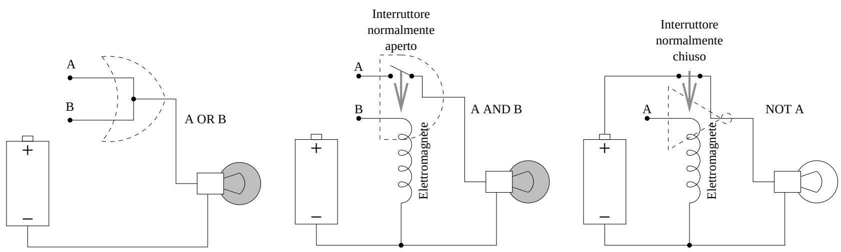


Figura 3.1: Rappresentazione di una cifra binaria su una linea elettrica e sua "verifica" per mezzo di una lampadina.



(a) Porta OR (b) Porta AND (c) Porta NOT

Figura 3.2: Realizzazione di porte logiche con mezzi elettromeccanici.

In linea di principio è possibile combinare insieme molte porte realizzate come in figura 3.2 sostituendo alla lampadina di una porta l'ingresso di un'altra porta; così facendo si possono realizzare i sommatori e gli altri componenti di un calcolatore elettronico.

La combinazione di un interruttore controllato da un elettromagnete (detta *relé*) era effettivamente utilizzata dai calcolatori elettromeccanici degli anni 30. I relé sono stati in seguito soppiantati da sistemi allo stato solido (senza parti in movimento) basati su altri principi fisici.

Si noti che non è necessario utilizzare l'elettricità per realizzare un sistema di calcolo, e che le porte logiche possono essere realizzate (una volta stabilito come rappresentare le cifre binarie) da dispositivi meccanici, idraulici, termici, quantistici, biologici. L'elettronica è stata scelta in quanto più veloce, affidabile e miniaturizzabile rispetto alle altre categorie di fenomeni fisici. La meccanica quantistica e la biologia molecolare (il DNA) sono altri due campi promettente per quanto riguarda la realizzazione di elaboratori non elettronici, ma i fenomeni su cui si basano sono difficilmente controllabili, e dovrà passare ancora molto tempo prima di avere un computer quantistico o "in provetta". Limitiamoci dunque, nel seguito, agli elaboratori elettronici.

Capitolo 4

Struttura di un elaboratore elettronico

Gli elaboratori moderni sono strutturati secondo lo schema di figura 4.1. La componente principale è detta *Unità di elaborazione centrale* (Central Processing Unit, CPU). La sua funizone è quella di eseguire operazioni su dati numerici. I dati sono contenuti nella memoria centrale, strutturata come una lunga sequenza di *celle*, ciascuna in grado di contenere un numero a 8 bit. Ciascuna cella è identificata da un *indirizzo* numerico, solitamente a 32 bit.

Le istruzioni che la CPU è in grado di eseguire sono molto semplici, e consistono di operazioni aritmetiche sui dati contenuti in particolari celle di memoria, nello spostamento di dati o nel confronto fra valori numerici. Anche le istruzioni sono codificate numericamente nella memoria centrale: la CPU le legge in sequenza e le esegue.

Per visualizzare il risultato dell'elaborazione, o per ricevere dati dall'utente, la CPU è in grado di ricevere dati numerici dalle *periferiche di ingresso/uscita* (input/output, I/O), tipicamente lo schermo video, la tastiera, il mouse, la stampante eccetera. Anche se si tratta di dispositivi *alfanumerici*, in grado cioè di rappresentare anche lettere, segni di interpunzione, immagini e altro, internamente all'elaboratore tutto è rappresentato da *codici numerici*.

Infine, è necessario notare che la memoria centrale è tipicamente *volatile*: perde il suo contenuto allo spegnimento della macchina. Per questo e per altri motivi un calcolatore è sempre dotato di un dispositivo di memorizzazione di massa, nel quale i dati sono memorizzati stabilmente per mezzo di grandezze fisiche non volatili (tipicamente la magnetizzazione di un materiale o l'accumulo di cariche elettriche). Quando il calcolatore viene acceso, o più in generale quando un nuovo programma viene eseguito, la sequenza di istruzioni (il "codice") corrispondente viene trasferito dalla memoria di massa alla memoria centrale. Lo stesso avviene per i dati su cui il programma opera (ad esempio, un documento che dev'essere modificato con Word). Tipiche memorie di massa sono il disco rigido (hard disk), i dischetti, i CD-ROM, le memorie flash (utilizzate soprattutto nelle macchine fotografiche digitali), le "chiavette" USB.

La volatilità della memoria centrale è dovuta alla sua velocità: è fondamentale che

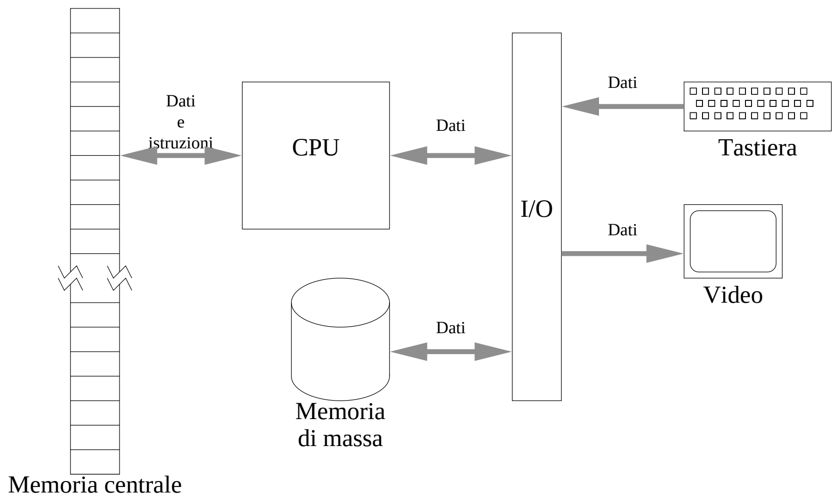


Figura 4.1: Struttura di massima di un elaboratore elettronico.

il trasferimento di dati e istruzioni tra la CPU e la memoria centrale sia il più veloce possibile, e la velocità impone l’uso di una tecnologia che costringe ad alimentare elettricamente la memoria se si desidera che i dati siano mantenuti; le memorie di massa sono tipicamente molto più lente, ma più capienti e permanenti.