

Angelo Raffaele Meo

Aurora Martina Neri

**L'INFORMATICA RACCONTATA
AI GRANDI E AI PICCINI**



PREMESSA

Questo libro è destinato ai ragazzi di età compresa tra i 10 e i 14 anni circa. Abbiamo pensato a loro perché vengono in contatto ogni giorno con gli strumenti informatici e di cui spesso ne conoscono solo superficialmente il funzionamento.

Speriamo che possa risultare utile anche a quei docenti, educatori, genitori che condividono con noi l'idea che oggi insegnare l'informatica ai ragazzi sia non solo importante ma indispensabile al fine di contribuire a creare in loro maggiore autonomia, sicurezza e capacità di gestire il proprio futuro.

Ci auguriamo che possa essere gradito soprattutto ai ragazzi, spesso avviati dal sistema scolastico a una conoscenza troppo "*applicativa*" dell'informatica.

Il libro è stato creato anche per essere fruito on line sulle piattaforme per la didattica oggi più diffuse.

L'edizione attuale presenta una caratteristica innovativa rispetto alle edizioni precedenti. Infatti, il testo è integrato da alcuni videogiochi didattici che fanno diretto riferimento agli algoritmi descritti nei singoli capitoli del testo.

I videogiochi potranno scaricati dal sito <http://www.gaminaction.com/clients/politecnico/release/> (username: poli, password: poli) ed essere eseguiti su qualunque personal computer in ambiente Windows, Mac e Linux. Inoltre, potranno essere giocati "on line" ai seguenti indirizzi:

<http://www.gaminaction.com/clients/politecnico/gioco01>

<http://www.gaminaction.com/clients/politecnico/gioco02>

<http://www.gaminaction.com/clients/politecnico/gioco03>

Nell'edizione attuale i giochi sono finalizzati alle esercitazioni relative al calcolo binario e operazioni connesse. Alla fine del corso, insegnanti e allievi potranno verificare il livello dell'apprendimento con una versione originale del Gioco dell'Oca all'indirizzo:

<http://www.gaminaction.com/clients/politecnico/gioco06/>

Gli Autori

Prima edizione: aprile 2012

Seconda edizione: luglio 2014

Quest'opera è stata rilasciata con licenza Creative Commons Attribuzione - Non commerciale - Condividi allo stesso modo 3.0 Italia. Per leggere una copia della licenza visita il sito web <http://creativecommons.org/licenses/by-nc-sa/3.0/it/> o spedisce una lettera a Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Indice

Parte n. 1 - Il COMPUTER

1. L'aritmetica degli elefanti (e dei calcolatori elettronici)
2. La logica degli elefanti e degli uomini
3. I circuiti elettronici elementari (AND, OR, NOT)
4. La codifica ASCII, bit e byte
5. La memoria degli elefanti
6. Da un registro all'altro
- 6S - Scheda di approfondimento: trasferimenti multipli tra registri
- 7S - Scheda di approfondimento: L'Addizionatore (Half adder e Full adder)
- 8S - Scheda di approfondimento: La memoria centrale
- 9S - Scheda di approfondimento: L'unità di controllo
10. Traduzione da un linguaggio ad un altro

Parte n. 2 - La RETE

1. La trasmissione dei dati
2. Le reti di calcolatori
3. Internet
4. I servizi più importanti di Internet
5. Internet del futuro



Introduzione ai giochi didattici

JIMMY JUMBO A MONKEY TOWN

Sul pianeta degli animali gli Elefanti e le Scimmie convivono pacificamente nelle loro isole, Elephantville e Monkey Town. La vita scorre tranquilla fino a quando le scimmie, avendo mangiato banane in grande quantità, cominciano ad avere problemi con le bucce, che ormai invadono ogni angolo della loro città!

Invece di cercare una soluzione costruttiva e magari chiedere aiuto agli elefanti, le scimmie decidono di costruire un grosso macchinario simile a un aspirapolvere-cannone per raccogliere le bucce e spararle direttamente su Elephantville.

La macchina spara-bucce è un mix tra un cannone e una palma, alto e minaccioso: il suo nome in codice è BANANA SPIT.

Gli elefanti, sorpresi dalla pioggia di scarti di banana, decidono di mandare una squadra scelta, capitanata dal simpatico Jimmy Jumbo, a cercare di neutralizzare la malefica macchina spara-bucce! Ce la faranno i nostri orecchiuti eroi?



Minigioco 1 - Elefanti in fuga

Una volta arrivati alla base delle scimmie gli elefanti vengono subito individuati (anche perché per loro non è facilissimo agire discretamente) e ingabbiati!

La scimmia soldato che li ha catturati sghignazza, ma Jimmy si accorge che in cima alla gabbia c'è un tasto di apertura, che possono raggiungere con la proboscide.

Tutti gli elefanti premono il pulsante ma la scimmia ha creato un codice di sicurezza per bloccare il meccanismo: per uscire gli elefanti dovranno decrittare il codice binario.

E' difficile perché la scimmia guardiana ad ogni tentativo riuscito richiude le celle fino a quando il macchinario, date le numerose ripetizioni, si guasta e i nostri eroi sono di nuovo liberi, mentre Mickey Monkey, che controlla la situazione attraverso un monitor, va su tutte le furie!

Jimmy Jumbo e la sua squadra riescono a fuggire raggiungendo la porta blindata di ingresso della grande Banana Spit.

Minigioco 2 - La porta blindata

Entrati nel locale tecnico, Jimmy si avvicina alla porta blindata per tentare di aprirla, mentre gli altri bloccano l'ingresso (con la loro mole non è difficile).

Per poter entrare, Jimmy deve riuscire a inserire la giusta combinazione numerica, convertendo i numeri binari in numeri decimali.

L'elefante decifra il codice binario mentre le scimmie collegate continuano a mandare alla porta nuovi codici per riuscire a evitare lo sblocco.

Alla fine del gioco, gli elefanti riescono ad entrare attraverso la porta blindata e a raggiungere il cervellone della macchina.

Minigioco 3 - Attacco hacker

Ma Banana Spit ovviamente non gradisce l'intrusione e, in pieno stile Odissea nello Spazio, "cattura" in un ambiente senza gravità gli elefanti.

Per riuscire a neutralizzare la minaccia e a tornare liberi, il grande Jimmy deve risolvere le operazioni binarie! Ogni calcolo riuscito spegne un pezzetto del grande cervellone. Ad ogni errore purtroppo Jimmy deve ricominciare da capo!





1. L'aritmetica degli elefanti (e dei calcolatori elettronici)

Noi uomini usiamo da molti anni, per contare, quello che con parole difficili viene generalmente chiamato "sistema di numerazione decimale". In parole più semplici, noi usiamo dieci simboli diversi

0, 1, 2, 3, 4, 5, 6, 7, 8, 9

per rappresentare i dieci numeri da 0 a 9; poi, per continuare a contare, scriviamo

10

che vuol dire una (1) decina, ossia un pacco di dieci cose, e zero (0) unità, ossia un pacco vuoto. E continuiamo scrivendo

11

che vuol dire "un pacco di dieci cose più una cosa", e poi

12

che vuol dire "un pacco di dieci cose più due cose", e così via.

Così, ad esempio,

37

vuol dire "3 pacchi di dieci cose + 7 cose". Proseguendo

356

vuol dire "3 pacchi di cento cose + 5 pacchi di dieci cose + 6 cose".

Molto probabilmente noi uomini usiamo per contare il sistema decimale perché abbiamo dieci dita, che costituiscono il primo calcolatore messo a disposizione dall'uomo più di centomila anni fa.

Ma quello non è l'unico sistema di calcolo possibile. Ad esempio, i contadini contano le uova a dozzine e noi stessi diciamo che un'ora è composta da 60 minuti e un minuto da 60 secondi.

Gli elefanti che hanno uno strumento molto maneggevole, la proboscide, ma sfortunatamente per loro, una sola proboscide, usano un sistema di numerazione diverso, che, per quella che potrebbe sembrare una combinazione (ma non lo è), è anche quello usato dai calcolatori elettronici. Quando un elefante vuole indicare una cosa, solleva la proboscide come vediamo nella Fig. 1-1.



Fig. 1-1



Quando invece vuole indicare "nessuna cosa" abbassa la proboscide come nella seguente Fig. 1-2.

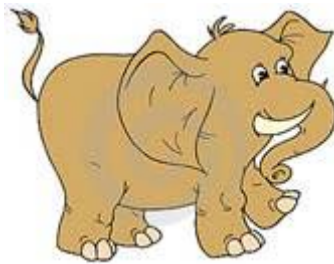


Fig. 1-2

Supponiamo ora che un elefante voglia indicare "tre cose". Per non essere costretto a alzare la proboscide per tre volte, l'elefante usa un metodo che oggi noi chiamiamo codice. Questo metodo è molto simile a quello del sistema decimale, ma usa componenti di versi, come quelli che vediamo.



unità



coppia o duetta



quadrupletta



ottupletta



sedici-pletta

(ed altre ancora 32-pletta, 64-pletta...)



Come nel sistema decimale, anche nel sistema degli elefanti i componenti assumono un valore che dipende dalla loro posizione. Nella numerazione decimale, partendo da destra abbiamo le "unità", le "decine", le "centinaia", eccetera. Nella numerazione binaria allo stesso modo, partendo da destra abbiamo le "unità", le coppie o "duette", le "quadruplette", le "ottuplette", eccetera.

Usando questi componenti e sfruttando la loro posizione partendo da destra, l'elefante prima alzerà la proboscide per indicare che vuole un'unità e poi alzerà di nuovo la proboscide per indicare che vuole una coppia.





	
	
<p>SU</p>	<p>SU</p>

Fig. 1-3

Vediamo un altro esempio.

Supponiamo che l'elefante voglia indicare il numero 5. Allora nell'ordine eseguirà queste operazioni:

- a) prima alzerà la proboscide per indicare che vuole una unità (ossia la componente da un rametto)
- b) poi abbasserà la proboscide per indicare che non vuole nessuna coppia
- c) infine, alzerà la proboscide per indicare che vuole una quadrupletta (ossia una componente da quattro rametti).

La Fig. 5-4 illustra la rappresentazione del numero 5 e la Fig. 5-5 la rappresentazione del numero 11.



SU	GI U'	SU

Fig. 5.4

Segue la rappresentazione del numero 11.

SU	GI U'	SU	SU

Fig. 5.5







I calcolatori elettronici di oggi usano lo stesso metodo per ragioni di sicurezza di funzionamento. Infatti, per un calcolatore è più facile controllare, ad esempio, se in un certo circuito c'è corrente oppure no, che non verificare il valore di una corrente che possa assumere dieci livelli di diversi.

Noi uomini, per interpretare meglio il comportamento degli elefanti e dei calcolatori usiamo i simboli 0 e 1, dove 0 vuol dire "proboscide giù", o "assenza di corrente" in un filo elettrico, e 1 significa "proboscide su", o "presenza di corrente" nel filo elettrico corrispondente.







Proviamo a scrivere gli esempi precedenti seguendo l'interpretazione degli umani.

Cominciamo con il numero 3.

	
	
SU ossi a 1	SU ossi a 1

Quindi 3 cose si rappresentano con 1 1.

Vediamo ora come gli umani rappresentano il numero 5.

		
		
SU ossi a 1	GIU' ossi a 0	SU ossi a 1



Il numero 5 è rappresentato da 1 0 1.

E, infine, vediamo come si scrive il numero 11.

<p>SU ossi a 1</p>	<p>GIU' ossi a 0</p>	<p>SU ossi a 1</p>	<p>SU ossi a 1</p>

Quindi il numero 11 sarà 1 0 1 1.



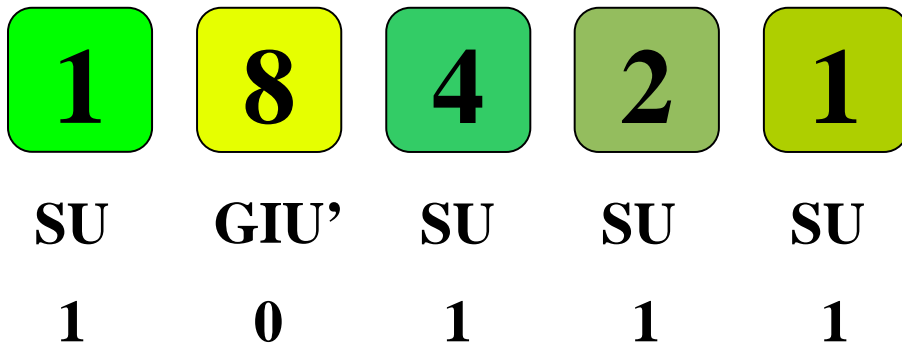
Eserciti amoci a rappresentare i numeri decimali con l'aritmetica degli elefanti

1. In quale modo gli elefanti rappresentano i due valori 0 e 1?
2. Qual è il numero binario che corrisponde al numero decimale 23?

Ripensiamo ai mucchietti di rami e alla loro posizione partendo da sinistra e cominciamo a domandarci se ci può essere utile il mucchietto di 32 rametti. La risposta è: NO, quel mucchietto è troppo grande e non ci serve.

Prendiamo allora il mucchietto appena più piccolo, quello da 16 rametti. Questo ci serve e lo teniamo.

A questo punto, dal nostro numero 23 dobbiamo togliere i 16 rametti del mucchietto, ne rimangono 7, e quindi ora dobbiamo rappresentare il numero 7. Sfortunatamente il mucchietto da 8 rametti è troppo grosso e così non lo utilizziamo. Allora prendiamo il mucchietto da 4 rametti e poi, visto che rimangono ancora 3 rametti, prendiamo quello da 2 e poi quello da 1. Così, per rappresentare il numero 23 utilizziamo questi mucchietti:



Adesso seguendo lo stesso metodo, prova a rappresentare in binario altri numeri.

3. Rappresenta i numeri da 1 a 9.
(Ad esempio rappresentiamo il numero 9).



4. Rappresenta i numeri: 10, 20, 30, 40, 50, 60, 70, 80, 90, 100
(Ad esempio rappresentiamo il numero 40).



32	16	8	4	2	1
SU	GIU'	SU	GIU'	GIU'	GIU'
1	0	1	0	0	0

5. Rappresenta i numeri 64, 128, 256, 512, 1024, 65536

Eserciti amoci a trovare i numeri decimali corrispondenti ai numeri binari

1. Qual è il numero decimale corrispondente al numero binario

1 1 0 1 1 ?

Iniziamo mettendo sotto ogni numero il mucchietto di rametti che ci sembra utile, partendo da destra.

1	1	0	1	1
1	8	4	2	1
SU	SU	GIU'	SU	SU

e poi facciamo la somma: $16 + 8 + 2 + 1 = 27$.

2. Quali sono i numeri decimali che corrispondono a:

1 0 0,

1 0 0 0,

1 0 0 0 0,

1 0 0 0 0 0,

1 0 0 0 0 0 0.



Le operazioni dell'aritmetica degli elefanti

Proviamo a calarci nella testa degli elefanti per imparare la loro aritmetica.

Per cominciare vediamo la tabellina elementare:

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 10$$

In quest'ultimo caso scriveremo 0 con riporto di 1, come vedremo in seguito.

Supponiamo ora di dover eseguire l'operazione che noi scriveremmo $1 + 3$.

Gli elefanti scriverebbero

$$\begin{array}{r} 01 + \\ 11 \end{array}$$

perché, come abbiamo visto, quelli che per noi sono i numeri 1 e 3 per loro sono 01 e 11.

Cominciamo a sommare le due cifre più a destra:

$$\begin{array}{r} 01 + \\ 11 \\ \hline \end{array} ?$$

Noi sappiamo che $1 + 1$ è uguale a 2, ma gli elefanti non conoscono il simbolo 2. Per loro 2 è uguale a 10. Allora gli elefanti scriverebbero:

$$\begin{array}{r} (1) \\ 01 + \\ 11 \\ \hline 0 \end{array}$$

usando la logica del riporto.

A questo punto gli elefanti sommerebbero $1 + 0 + 1$, che per noi è 2 ma per loro è 10. Così il risultato finale diventa

$$\begin{array}{r} (1) \\ 01 + \\ 11 \\ \hline 100 \end{array}$$

che corrisponde a quel numero che noi chiamiamo "4".



Vediamo un esempio con numeri più complessi.

<pre> (1)(1) (1)(1) 1 0 1 0 1 1 + 0 1 1 0 0 1 ----- 1 0 0 0 1 0 0 </pre>	<pre> 43 + 25 ----- 68 </pre>
--	--

Passiamo alla moltiplicazione, che è più facile, perchè coincide con la nostra moltiplicazione. Infatti,

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

per loro, come per noi.

Applichiamo questa tabellina alla moltiplicazione di due numeri binari lunghi.

```

  1001 x
   110 =
-----
  0000
 1001
1001
-----
110110
    
```

Vediamo un altro esempio.

```

  1101 x
   101 =
-----
  1101
 10000
1101
-----
1000001
    
```

Analizziamo ancora un esempio un po' più complesso.



$ \begin{array}{r} 10110 \times \\ 11101 = \\ \hline 10110 \\ (1) 00000 \\ (1) 10110 \\ (1) 10110 \\ (1) 10110 \\ (1) 10110 \\ 10110 \\ \hline 100111110 \end{array} $	$ \begin{array}{r} 22 \\ 29 \\ \hline 198 \\ 44 \\ \hline 638 \end{array} $
---	---

Passiamo alla sottrazione che è un po' più difficile. Cominciamo dalla tabella elementare che nei primi tre casi è uguale alla nostra

$0 - 0 = 0$

$1 - 0 = 1$

$1 - 1 = 0$

Il quarto caso è quello difficile. Infatti,

$0 - 1 = ?$

per eseguire questa operazione occorre che lo 0 si faccia prestare qualcosa e la cosa più piccola da prestare è 10 (che è il nostro 2).

Eseguiamo allora la sottrazione

$$\begin{array}{r}
 1^{(1)}0 - \\
 0 1 = \\
 \hline
 0 1
 \end{array}$$

Infatti, lo 0 in alto a destra si è fatto prestare un 1 (in alto a sinistra). Ma l'1 in alto è in realtà 10, ossia 2.

Vediamo qualche altro esempio più complicato.

$ \begin{array}{r} (1) \quad (1) \\ 1 0 1 1 0 1 \\ 0 1 1 0 1 1 \\ \hline 0 1 0 0 1 0 \end{array} $	$ \begin{array}{r} 45 - \\ 27 \\ \hline 18 \end{array} $
--	---



Esercizi

1. Quali sono le tre operazioni nell'aritmetica degli elefanti?
2. Qual è il valore di $1 + 1$ nell'aritmetica degli elefanti?
3. Eseguire le seguenti operazioni e poi trasformare i numeri in numeri decimali

$$00101010 + 00011111 =$$

$$\begin{array}{r} 0010\ 1010\ + \\ 0001\ 1111\ = \\ \hline 0100\ 1001 \end{array}$$

$$00100011 - 00010110 =$$

4. Quali sono le operazioni che mancano all'aritmetica degli elefanti rispetto alla nostra?
5. Con quale ordine devono essere risolte le operazioni di un'espressione aritmetica degli elefanti?
6. Indica quali delle seguenti affermazioni sono vere e giustifica la tua risposta.
 - a) per indicare il valore 1 gli elefanti sollevano la proboscide.
 - b) nell'aritmetica degli elefanti $1 + 1 = 2$
 - c) nell'aritmetica degli elefanti le addizioni hanno la precedenza sulle moltiplicazioni.



2 . La logica degli elefanti (e degli uomini)

Gli elefanti usano la proboscide, non soltanto per contare, ma anche per dire SI oppure NO. Se la mamma chiede al suo elefantino "Vuoi mangiare un rametto tenero?" l'elefantino risponde SI alzando la sua piccola proboscide oppure risponde NO abbassandola.

Gli elefanti, così come gli uomini, hanno una logica, ossia un modo di ragionare molto raffinato. In questa logica compaiono tre operazioni fondamentali.

La prima operazione è quella che viene chiamata dagli studiosi "disgiunzione logica" ma che noi, che non amiamo le parole difficili, chiameremo OPPURE.

Vediamo un esempio di ragionamento di un elefantino che si domanda se debba fare il bagnetto nello stagno, sapendo che il farlo o non farlo, dipende solo da due eventi, che l'acqua sia fresca oppure no, e che lui stesso sia sporco oppure no.

Prepariamo una tabella che riassume tutti i casi possibili del suo ragionamento.

ACQUA_FRESCA	ELEFANTI NO_SPORCO	BAGNETTO
NO	NO	NO
NO	SI	SI
SI	NO	SI
SI	SI	SI

Gli studiosi dei calcolatori prediligono i due simboli 0 e 1 anche per impostare ragionamenti, e quindi:

"Se una cosa, o una frase è vera, ossia è SI, scriviamo 1; se invece è falsa, ossia è NO, scriviamo 0".

Nota bene che in questo caso i due simboli 0 e 1 non hanno alcun valore di quantità e non sono neanche i simboli che abbiamo usato nell'aritmetica degli elefanti (e dei calcolatori).

Introducendo i due simboli 0 e 1 la tabella dell'operazione logica OPPURE (in inglese OR) è la seguente:

ACQUA_FRESCA	ELEFANTI NO_SPORCO	BAGNETTO
0	0	0
0	1	1
1	0	1
1	1	1



Sicuramente avete notato che abbiamo attaccato le due parole "acqua" e "fresca" con il simbolo di sottolineatura "_", così come abbiamo attaccato le due parole "elefantino" e "sporco". Abbiamo usato questo simbolo per ricordare che "acqua_fresca" è un'unica identità, ossia una proposizione che può essere vera (SI o 1) *oppure* falsa (NO o 0).

Questo modo di rappresentare le proposizioni che possono essere vere o false ci servirà quando studieremo un linguaggio di programmazione, ossia un linguaggio che servirà a dire al calcolatore cosa deve fare.

Gli studiosi dei calcolatori usano anche il simbolo + per indicare l'operazione OPPURE. E' una scelta infelice, perché il + logico non ha alcuna parentela con il + dell'aritmetica.

Usando il nuovo simbolo +, il ragionamento logico dell'elefantino può essere scritto come:

bagnetto = acqua_fresca + elefantino_sporco

La seconda operazione che vogliamo descrivere è stata chiamata dagli studiosi della logica "congiunzione logica". Per semplicità noi la chiameremo "E INOLTRE" (in inglese AND).

Vediamo un esempio di ragionamento dell'elefantino, che con parole semplici potremmo scrivere così:

"Se vedo una tigre e inoltre la mamma è lontano, devo scappare"

La tabella che riassume tutti i casi possibili e che riassume il suo ragionamento è la seguente:

TI GRE_PRESENTE	MAMMA_LONTANA	DEVO-SCAPPARE
NO	NO	NO
NO	SI	NO
SI	NO	NO
SI	SI	SI

Usando i due simboli 0 e 1 tanto amati dagli studiosi dei calcolatori la stessa tabella, ossia la tabella dell'operazione E INOLTRE (in inglese AND), diventa:

TI GRE_PRESENTE	MAMMA_LONTANA	DEVO-SCAPPARE
0	0	0
0	1	0



1	0	0
1	1	1

Gli studiosi dei calcolatori usano * (che nei linguaggi di programmazione è il simbolo della moltiplicazione) per indicare l'operazione E INOLTRE.

Anche questa è una scelta infelice, perché la moltiplicazione logica non è parente della moltiplicazione aritmetica, anche se, per una pura combinazione, la tabella della moltiplicazione logica e quella della moltiplicazione aritmetica coincidono.

Usando il simbolo *, il ragionamento dell'elefantino può essere scritto come:

devo_scappare = tigre_presente * mamma_lontana

Vediamo ora una terza, e ultima, definizione di operazione logica.

È l'operazione che gli studiosi chiamano "negazione" o "complementazione" e che noi chiameremo "operazione NON" (in inglese NOT).

Partiamo, come nel caso delle due operazioni logiche precedenti, dal ragionamento dell'elefantino, che sa che è molto pericoloso andare a fare il bagnetto nello stagno quando c'è la tigre in giro.

La tabella che riassume il ragionamento dell'elefantino è la seguente:

TI GRE_PRESENTE	BAGNETTO
NO	SI
SI	NO

Usando i due simboli 0 e 1 la stessa tabella, o "tabella della negazione", diventa

TI GRE_PRESENTE	BAGNETTO
0	1
1	0

Per rappresentare con un unico simbolo questa operazione di complementazione (o negazione) utilizzeremo una barretta



orizzontale e sopra l'espressione che vogliamo complementare. Ad esempio, scriveremo:

$$\text{bagnetto} = \overline{\text{tigre_presente}}$$

che vuol dire che "quando tigre_presente è = 0, allora bagnetto = 1 e quando tigre_presente = 1, bagnetto = 0.

Le tre operazioni logiche AND, OR, NOT, che abbiamo visto posso essere combinate fra loro in molti modi diversi per rappresentare ragionamenti più complessi. Ad esempio, possiamo scrivere

$$\text{bagnetto} = \overline{\text{tigre_presente}} * (\text{acqua_fresca} + \text{elefantino_sporco})$$

Qui le due parentesi delimitano il primo ragionamento elementare:

"se l'acqua è fresca oppure l'elefantino è sporco, sarebbe opportuno fare il bagno"

Ma questa condizione non è sufficiente, perché potrebbe esserci la tigre in giro, nel qual caso non si deve assolutamente andare allo stagno.

La tabella che riassume tutti i casi possibili è la seguente

TIGRE-PRESENTE	ACQUA-FRESCA	ELEFANTINO-SPORCO	BAGNETTO
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Certamente avrete notato un simpatico cagnetto che qualche volta ha commentato gli argomenti trattati. E' il cane di uno degli autori di questo libro e si chiama Moon (parola inglese che si pronuncia "Muun" e significa "luna"). E' bruttino, ma molto intelligente e conosce bene l'Informatica.

Spesso litiga con altri cani, ma non litiga mai con le cagnette e con i cuccioli. Perché?

Vediamo ora qualche altro esempio.

"resto a casa se piove OPPURE non ho finito i compiti"



casa = finito_compiti + piove

“prendo il gelato se non fa freddo e ho i soldi per comprarlo”

gelato = freddo * soldi

“per scrivere occorre un foglio E INOLTRE (una penna OPPURE una matita)”

scrivere = foglio * (penna + matita)

Perché abbiamo messo una parentesi nella frase?

Perché, se non avessimo messo la parentesi, il significato sarebbe stato molto diverso. Ossia la frase avrebbe potuto significare che per scrivere occorre (un foglio E INOLTRE una penna) oppure una matita. E nel secondo caso il foglio non sarebbe stato necessario.

Per comprendere bene questo concetto esaminiamo con attenzione le due tabelle seguenti che riassumono le due diverse proposizioni.

“Per scrivere occorre un foglio E INOLTRE (una penna OPPURE una matita)”

	FOGLIO	PENNA	MATITA	SCRIVERE
1	NO	NO	NO	NO
2	NO	NO	SI	NO
3	NO	SI	NO	NO
4	NO	SI	SI	NO
5	SI	NO	NO	NO
6	SI	NO	SI	SI
7	SI	SI	NO	SI
8	SI	SI	SI	SI

“Per scrivere occorre (un foglio E INOLTRE una penna) OPPURE una matita”

	FOGLIO	PENNA	MATITA	SCRIVERE
1	NO	NO	NO	NO
➔ 2	NO	NO	SI	SI
3	NO	SI	NO	NO
➔ 4	NO	SI	SI	SI
5	SI	NO	NO	NO
6	SI	NO	SI	SI
7	SI	SI	NO	SI
8	SI	SI	SI	SI

Notate che la seconda e la quarta fila della seconda tabella sono sbagliate. E' vero, con la matita potrei scrivere sul muro ma non è bello e non è quello che intendevamo dire con il nostro esempio.

Nella logica dei calcolatori se, in una frase, si trovano sia un +



La prima operazione da eseguire è il *, così come avviene nelle operazioni aritmetiche. Con parole difficili si vuol dire che il * ha la priorità più alta.

Esercizi

1. Trova il complemento di 0 nell'aritmetica degli elefanti.
2. Descrivi il significato dell'operazione di complementazione.
3. Qual è il significato della barra posta sopra il valore 0 o il valore 1.
4. E' vera la seguente affermazione:
- per trovare il complemento di un valore è sufficiente aggiungere 1 al valore stesso.

5. Qual è il risultato delle seguenti operazioni?

$$1 * \overline{1} = \quad [0]$$

$$0 * \overline{0} = \quad [0]$$

$$1 + \overline{1} = \quad [1]$$

$$0 + \overline{0} = \quad [1]$$

6. Calcola il valore delle espressioni seguenti.

$$1 + 1 + 1 =$$

$$1 * 1 + 1 * 1 =$$

$$1 + 0 * 1 =$$

$$1 + 1 * 0 =$$

$$0 * 1 + 0 + 1 * 0 =$$

$$0 * \overline{0} + 1 * 1 + 0 * 1 =$$

$$1 + 1 + 1 * 0 + 0 * \overline{1} + 1 =$$

$$1 + 1 + 1 + \overline{1} + 1 + 1 * 0 =$$

7. $1 + (1 * \overline{0} + 1 + 1) * 0 =$

8. $(1 + 0 * 1) * (0 + 1 * 1) + 1 * (\overline{1} + \overline{1} * 1 * 0)$

9. $\overline{((1 + 0) * (0 * 0 + 1) + \overline{1})} * 0 + (0 + 1 + 1)$

10. $\overline{((1 + 0 + \overline{1} * 0) * (1 + \overline{0} * 1 * 1) + 1 + 0) + 1 * 1}$

11. $1 + (0 * \overline{1} + ((\overline{0} * 1) + (1 * \overline{0}) + 0 * 0 + 1) * (0 + 1))18$



3. I circuiti elettronici elementari (AND, OR, NOT)

Un calcolatore elettronico è composto da miliardi di circuiti elettronici elementari del tipo di quelli che vedremo tra poco.

Il circuito elettronico OPPURE (OR)

Vediamo un primo esempio.



Fig. 3-1

Questo circuito genera nel punto C un segnale che dipende dai segnali nei punti A e B.

Cos'è un segnale?

Per comprendere cosa è un segnale dobbiamo sapere che i diversi punti di ingresso e uscita di un circuito elettronico possono trovarsi in due stati diversi, che potremmo chiamare "acceso" e "spento". Un punto è acceso quando una formichina prenderebbe la scossa se lo toccasse; è spento quando una formichina può toccarlo senza prendere la scossa. Abbiamo fatto l'esempio della formichina perché il livello di quella che gli elettricisti chiamano "tensione elettrica" è troppo basso (in un punto acceso) per far male agli uomini, ma è sufficiente per dare la scossa alle formiche.

Torniamo al circuito di Fig. 3-1, nel quale, come abbiamo visto, il segnale nel punto C dipende dai segnali nei punti A e B. In linguaggio tecnico si dice che A e B sono "ingressi" del circuito mentre C è l'"uscita".

In parole più semplici, se altri circuiti applicano due segnali ("acceso" e "spento") ai due punti A e B, dopo un intervallo di tempo molto piccolo, molto più piccolo di un miliardesimo di secondo, il livello del segnale nel punto C diventerà alto o basso (ossia acceso o spento) a seconda dei livelli dei segnali in A e B.

A sua volta il punto C potrà diventare l'ingresso di un altro circuito.

Il circuito disegnato in Fig. 3-1 ha il seguente comportamento.

Caso 1.

Quando i due segnali A e B sono entrambi nello stato "spento" anche il segnale C è "spento". (Fig. 3-2)

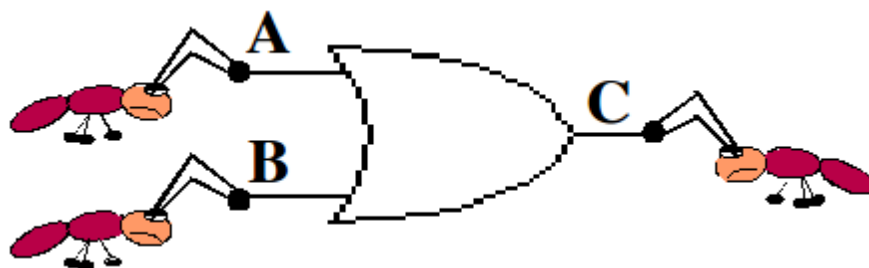


Fig. 3-2

Caso 2.

Il segnale A è nello stato "spento" mentre B è "acceso". In questo caso, anche il segnale C è "acceso". Sia la formichina in C sia quella in B prenderanno la scossa. (Fig. 3-3)

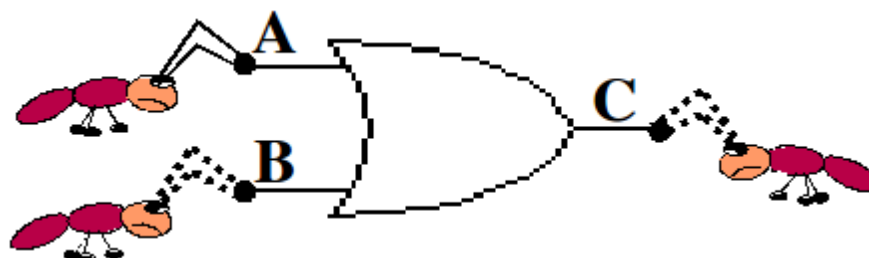


Fig. 3-3

Caso 3.

A è "acceso"; B è "spento". C è "acceso" anche in questo caso. (Fig. 3-4)

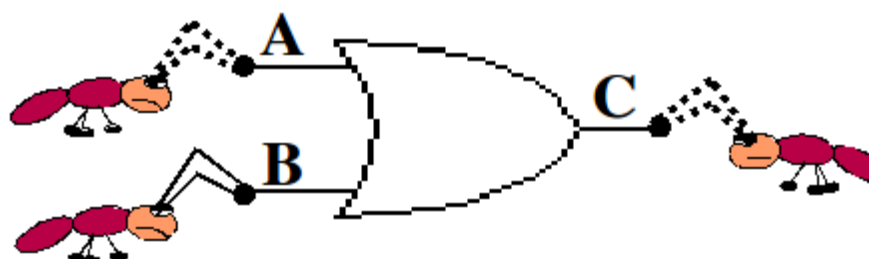


Fig. 3-4

Caso 4.

A e B sono entrambi "accesi". Allora anche C è "acceso". (Fig. 3-5)

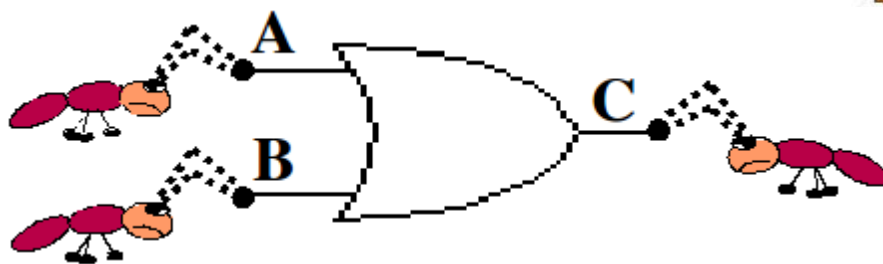


Fig. 3-5

Generalmente nello studio dei circuiti elettronici, si dice che un segnale vale 1 quando è nello stato "acceso" e vale 0 quando è nello stato "spento". Così, ad esempio, nel caso 2 dell'esempio precedente, si scrive

$$A = 0 \quad B = 1 \quad C = 1.$$

Adottando questa convenzione, il comportamento del circuito rappresentato in Fig. 3-1 è riassunto da questa tabella.

	A	B	C
Caso 1	0	0	0
Caso 2	0	1	1
Caso 3	1	0	1
Caso 4	1	1	1

Ma questa è esattamente la tabellina dell'operazione logica **OPPURE** (in inglese OR), mostrata nel capitolo 2 dall'esempio:

$$\text{BAGNETTO} = \text{ACQUA_FRESCA} + \text{ELEFANTINO_SPORCO}$$

Il circuito elettronico **E_INOLTRE** (AND)

Consideriamo ora il circuito rappresentato in Fig. 3-6, che ha un comportamento diverso da quello di Fig. 3-1. Esaminiamo separatamente i 4 casi possibili.

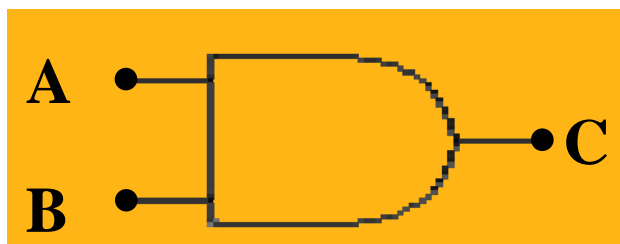


Fig. 3-6

Caso 1



Quando A e B sono "spenti", anche C è "spento". (Fig. 3-7)

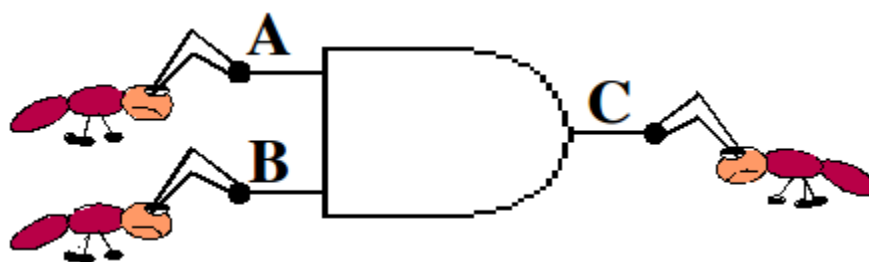


Fig. 3-7

Caso 2

Se A è "spento" e B è "acceso", C è "spento". (Fig. 3-8)

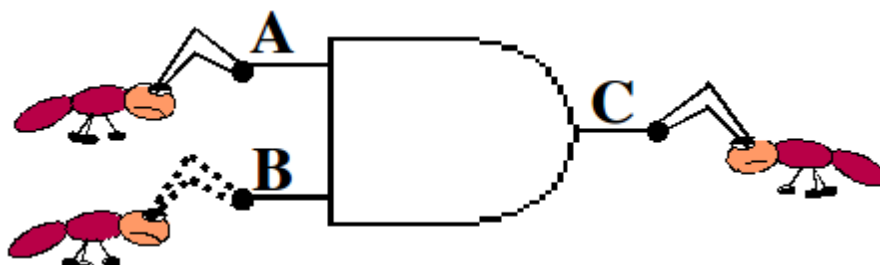


Fig. 3-8

Caso 3

Se A è "acceso" e B è "spento", C è "spento". (Fig. 3-9)

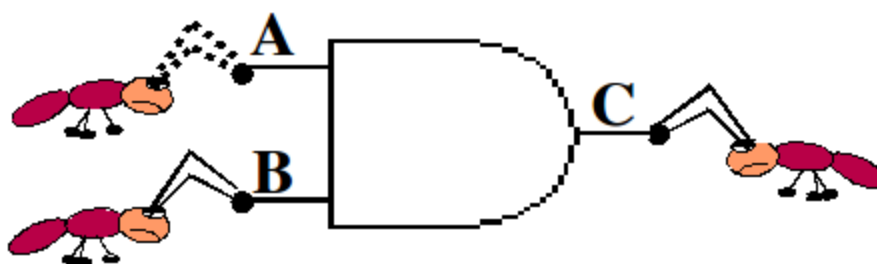


Fig. 3-9

Caso 4

Se A e B sono entrambi "accesi", anche C è "acceso". (Fig. 3-10)

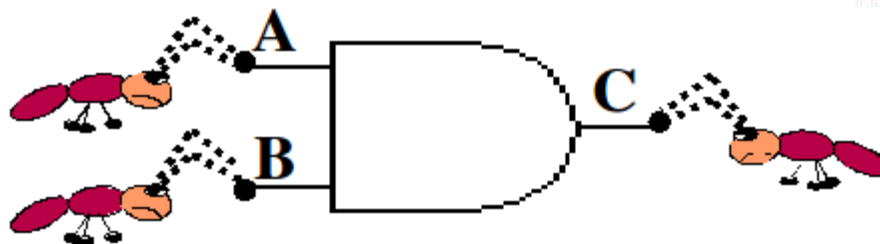


Fig. 3-10

Con la convenzione secondo la quale un segnale "acceso" vale 1 e un segnale "spento" vale 0, il comportamento del circuito rappresentato in Fig. 6-1 è riassunto dalla tabellina seguente.

	A	B	C
Caso 1	0	0	0
Caso 2	0	1	0
Caso 3	1	0	0
Caso 4	1	1	1

Questa è esattamente la tabellina dell'operazione logica E_INOLTRE (in inglese AND) che abbiamo esaminato nell'esempio:

$$\text{DEVO_SCAPPARE} = \text{TIGRE_PRESENTE} * \text{MAMMA_LONTANA}$$

Il circuito elettronico NON (NOT)

I circuiti elettronici di un calcolatore contengono anche un terzo tipo di componente elementare, che è generalmente schematizzato in Fig. 3-11

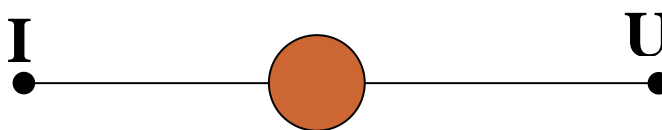


Fig. 3-11

Questo circuito elementare ha un solo nodo in ingresso (I) e un solo nodo di uscita (U).

Il valore di U dipende dal valore di I nel modo seguente:

quando I è "acceso", U è "spento" (Fig. 3-12)

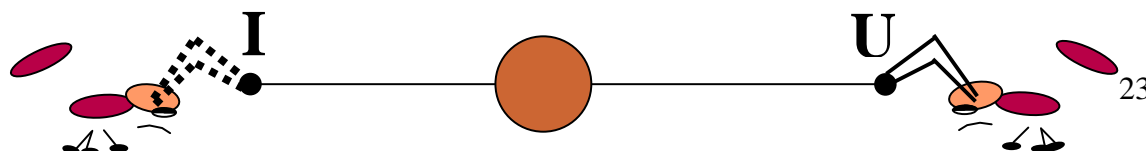




Fig. 3-12

quanto I è "spento", U è "acceso" (Fig. 3-13)

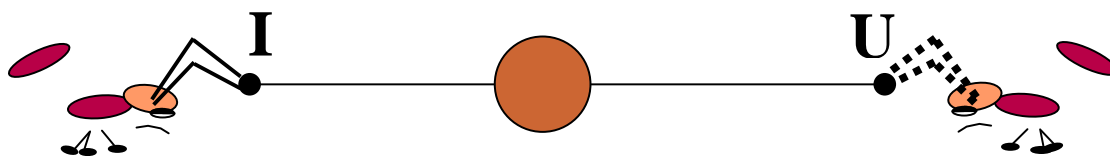


Fig. 3-13

	I	U
Caso 1	0	1
Caso 2	1	0

Che altro non è che la tabellina dell'operazione logica NON (in inglese NOT) che abbiamo presentato con l'esempio:

$$\text{BAGNETTO} = \overline{\text{TIGRE_PRESENTE}}$$

Alcuni circuiti complessi

Mettendo insieme più circuiti elementari è possibile realizzare circuiti complessi come quello di Fig. 3-14, che attua il ragionamento già studiato:

$$\text{BAGNETTO} = \overline{\text{TIGRE_PRESENTE}} * (\text{ACQUA_FRESCA} + \text{ELEFANTINO_SPORCO})$$

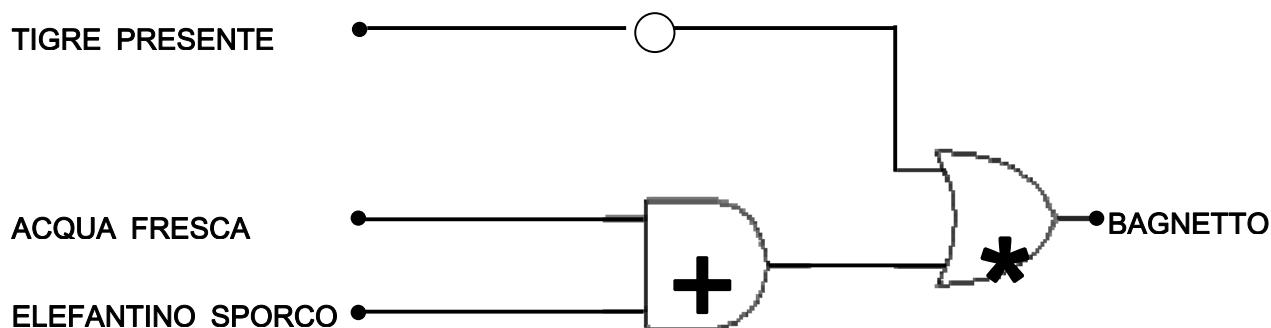


Fig. 3-14

Un secondo esempio è quello di Fig. 3-15, che mostra l'attuazione del ragionamento:



$$\text{SCRIVERE} = (\text{PENNA} + \text{MATITA}) * \text{FOGLIO}$$

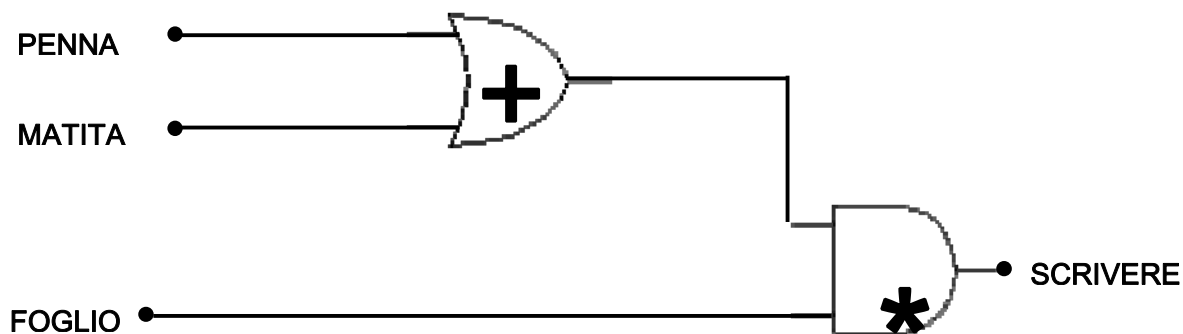


Fig. 3-15

Il concetto di funzione logica

Nel circuito di Fig. 3-14 il valore del nodo chiamato "BAGNETTO" dipende dai valori dei nodi chiamati "TIGRE_PRESENTE", "ACQUA_FRESCA" e "ELEFANTINO_SPORCO".

Usando un linguaggio più raffinato, possiamo dire che i nodi "TIGRE_PRESENTE", "ACQUA_FRESCA", "ELEFANTINO_SPORCO", "BAGNETTO" sono variabili nel senso che il valore di ciascuno di questi nodi può variare, ossia valere 1 qualche volta e valere 0, qualche altra volta.

Inoltre, la variabile "BAGNETTO" è funzione delle variabili "TIGRE_PRESENTE", "ACQUA_FRESCA", "ELEFANTINO_SPORCO", in quanto il suo valore dipende dai valori delle altre variabili.

Per esercizio provate a indicare le variabili e le funzioni del circuito di Fig. 3-15.

Circuiti elementari a molti ingressi

Consideriamo il circuito elementare rappresentato in Fig. 3-16. Il suo comportamento è descritto dalla tabella seguente.



Fig. 3-16

A	B	C	S
0	0	0	0



0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

E' chiaro che questo circuito è un'estensione del circuito OPPURE (OR) rappresentato in Fig. 3-1. Infatti, S vale 0 quando A, B e C sono tutti 0; vale 1 in tutti gli altri casi.

Analogamente, in un circuito OPPURE (OR) a quattro ingressi A, B, C e D, l'uscita S vale 0 quando tutti gli ingressi sono 0 e vale 1 in tutti gli altri casi.

Consideriamo ora il circuito elementare rappresentato in Fig. 3-17. Il suo comportamento è descritto dalla seguente tabella.

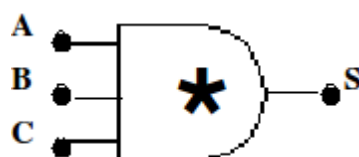


Fig. 3-17

A	B	C	S
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



Questo circuito è un'estensione del circuito E_INOLTRE (AND) rappresentato in Fig. 3-6. Infatti, S vale 1 quando tutti gli ingressi A, B e C valgono 1, e S vale 0, in tutti gli altri casi.

Più in generale, in un circuito E_INOLTRE (AND) a molti ingressi l'uscita vale 1 quando tutti gli ingressi valgono 1 e vale 0 in tutti gli altri casi.

Esercizi

1. Disegnare il circuito logico che attui la funzione: "resto a casa se piove oppure non ho finito i compiti".
2. Disegnare il circuito logico che attui la funzione: "prendo il gelato se non fa freddo e ho i soldi per comprarlo".
3. Disegnare il circuito logico che attui la funzione: "prendo il taxi se devo andare lontano e ho i soldi per pagarlo".
4. Disegnare il circuito logico che attui la funzione: "resto a casa se piove o nevicata o fa freddo".
5. Disegnare il circuito logico che attui la funzione: "la mia squadra vincerà il campionato se avrà buoni calciatori e un buon allenatore e molta fortuna".
6. Calcolare il valore della funzione rappresentata in Fig. 3-18 per i valori A=0 B=1 C=1 D=0.
7. Calcolare il valore della funzione rappresentata in Fig. 3-18 per i valori A=0 B=1 C=1 D=1.
8. Calcolare il valore della funzione rappresentata in Fig. 3-18 per i valori A=1 B=0 C=0 D=1.

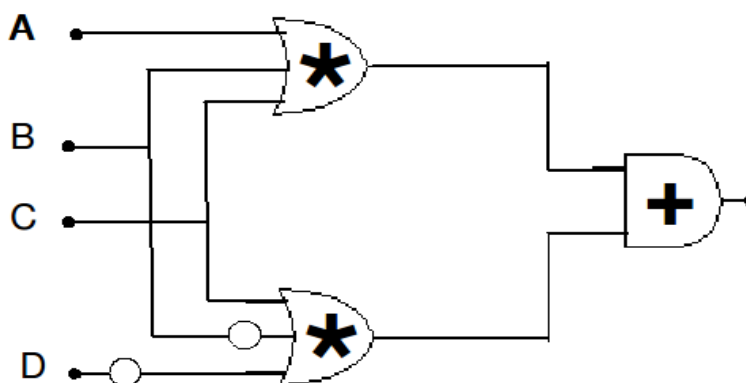


Fig. 3-18



4 . La codi fi ca ASCII. Bi t e byte

Se il semaforo è rosso, ci fermiamo e, per poter attraversare la strada, aspettiamo che diventi verde; se uno spettacolo ci è piaciuto, applaudiamo battendo le mani, altrimenti fischiamo; se incontriamo un amico gli stringiamo la mano; se ci è antipatico, allunghiamo la lingua.

Sono questi esempi di gesti o segni attraverso i quali riceviamo o trasmettiamo messaggi secondo convenzioni di verse.

Le convenzioni che assegnano un particolare significato ad un insieme di gesti o di segni sono chiamate codici.

Il galateo, ad esempio, è un codice di comportamento civile, il codice della strada stabilisce le regole per la circolazione e ci sono tanti altri codici di cui sicuramente hai già sentito parlare come il codice a barre, usato per il riconoscimento dei prodotti commerciali, il codice delle targhe automobilistiche, usato per l'identificazione dei veicoli, il codice Morse, il codice postale o il codice fiscale.



Calcola il codice fiscale

REPUBBLICA ITALIANA
MINISTERO DELLE FINANZE

CODICE FISCALE:

COGNOME:

NOME: SESSO:

LUOGO DI NASCITA:

PROVINCIA: DATA DI NASCITA:



Fig. 4-1

Esempi di codici

I codici si possono inventare dichiarando la convenzione che li regola. Un esempio di codice numerico potrebbe essere il seguente: sostituiamo le lettere dell'alfabeto con i numeri corrispondenti



alla loro posizione:

A = 1 B = 2 C = 3 D = 4 e così via.

In questo modo la parola COMPUTER diventa 3 13 11 14 19 18 5 16.

Come abbiamo visto, il calcolatore preferisce usare i numeri binari.

Uno dei codici più diffusi, che utilizzano solo le due cifre 0 e 1, è il codice ASCII.

ASCII è l'acronimo di "American Standard Code for Information Interchange", ovvero "Codice Standard Americano per lo Scambio dell'Informazione". Il codice ASCII è usato da tutti i calcolatori e **codifica tutti i caratteri con 8 bit** (un byte), come vedremo più oltre.

La tabella del codice ASCII inizia da 0 e arriva a 255, quindi ha 256 caratteri, perché con 8 bit si scrivono 256 caratteri diversi. Ricorda che 256 è uguale a 2^8 .

Nella tabella sono riportati i vari caratteri in codice ASCII corrispondenti ai singoli numeri del sistema di numerazione decimale, binario.

Esaminando nel dettaglio la tabella del codice ASCII, vediamo che dal numero 32 al numero 126 sono codificati i caratteri normali, mentre da 0 a 31 e con il 127 sono codificati i cosiddetti caratteri di controllo, codici che non corrispondono a un carattere, ma che possono avere un effetto particolare, ad esempio, andare a capo, generare un suono e così via. I caratteri dal 128 al 256 non hanno sempre lo stesso significato, vengono usati per le lettere accentate, i simboli e alcuni caratteri grafici particolari.

Per tradurre i testi in binario bisogna usare questo codice standard con il quale si possono rappresentare tutti i caratteri, i numeri e i caratteri di controllo.

Attenzione. Quando codifichiamo un testo dobbiamo codificare anche la punteggiatura e gli spazi vuoti tra le parole, non solo i caratteri e i numeri.

Ad esempio, analizziamo la frase

MI CHIAMO LUCA

Tra una parola e l'altra c'è uno spazio. Quindi la frase che dovrò codificare sarà questa:

MI ⓂCHI AMOⓂLUCA

dove Ⓜ rappresenta lo spazio tra una parola e l'altra.

Vediamo ancora:

ILⓂMI OⓂNUMEROⓂE' Ⓜ0115645403 (a capo)

H0Ⓜ10Ⓜanni (a capo)



Proviamo adesso a tradurre una di queste frasi in codice ASCII.

MI ħCHI AMOħLUCA (a capo)

```
010011010100100100100000010000110100100001001001010000010100110101
001111001000000100110001010101010000110100000100001101
```

Come vedete un testo diventa una lunga sequenza di 0 e 1 e il loro significato dipende dal codice con cui sono stati scritti. In questo caso, se andate a consultare la tabella ASCII vedrete che corrispondono esattamente ai caratteri della frase che dovevamo tradurre, come è stato indicato qui di seguito:

```
01001101 M
01001001 I
00100000 ħ
01000011 C
01001000 H
01001001 I
01000001 A
01001101 M
01001111 O
00100000 ħ
01001100 L
01010101 U
01000011 C
01000001 A
00001101 a capo
```

Anche i numeri possono essere rappresentati come codifica ASCII delle cifre decimali.

```
Ad esempio:   48   00110000   110000
              dec  ASCII      binario
```

A questo punto sappiamo scrivere un numero in tre modi diversi: decimale, binario e ASCII.

Esistono dei "programmi" sul calcolatore che permettono di vedere e modificare i testi scritti in ASCII.

I loro nomi comuni sono "editor" o "word processor" e alcuni esempi dei loro nomi propri sono OPENOFFICE, WORD, PFE, NOTEPAD.

A questo punto siamo capaci di leggere una fila di bit come quella dell'esempio che abbiamo esaminato prima.

Esercizi

1. Cos' è un codice?
2. Che cos' è il codice ASCII?
3. Com' è codificato il numero 6 nel codice ASCII?



4. Scrivere il proprio nome, cognome, indirizzo, età in codice ASCII.

5. Scrivere i numeri da 0 a 9 in binario e ASCII.

Bit e Byte

Una cifra binaria, 0 oppure 1, rappresenta la più piccola unità di informazione che possiamo comunicare al computer, corrispondente semplicemente ad un segnale basso (spento) o alto (acceso), e viene chiamata bit.

La parola **bit** è la contrazione delle due parole "binary digit", o "cifra binaria", ma in inglese significa anche un pezzetto, una piccola porzione di qualcosa. Nel nostro caso rappresenta il singolo dato che viene comunicato al computer, usando il "suo" linguaggio elettronico.

Bit è la più piccola unità di informazione che possiamo comunicare al computer, corrispondente ad una cifra binaria, 0 oppure 1.

La grande maggioranza dei computer utilizza per molte funzioni importanti, come la comunicazione con l'utente, un codice a 8 cifre binarie.

Una sequenza di 8 bit viene chiamata **byte**.

Byte è un insieme di 8 bit, ovvero 8 cifre binarie, generalmente utilizzato per indicare un elemento di informazione.

Così possiamo dire che un byte viene utilizzato per indicare il codice di uno specifico elemento di informazione.

In Fig. 4-2, ad esempio, è registrato un byte a 8 bit che rappresenta il numero binario 10010101.

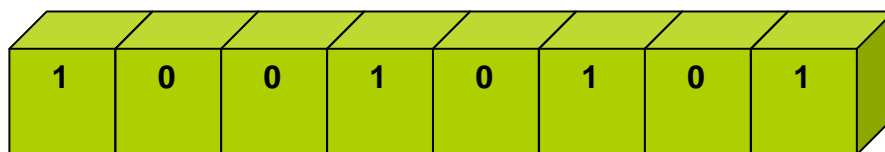


Fig. 4-2

A questo punto siamo già in grado di sapere quanti dati può memorizzare un computer!

Se un computer ha una memoria di 640 KB (kilobyte), significa che può memorizzare circa 640.000 caratteri.

Attenzione



In informatica K non significa proprio 1000, come siamo soliti pensare quando parliamo, ad esempio, di un chilogrammo o di un chilometro, ma 1024, cioè 2^{10} .

In questo modo 1 KB indica 1024 byte e 640 KB indicano 640×1024 byte.

Esercizi

1. Che cos'è un bit?
2. Che cos'è un byte?
3. A quanti bit corrisponde un byte?
4. Che cosa si intende dicendo che un calcolatore ha una memoria di 516 kbyte?



5 . La memoria degli elefanti

Abbiamo visto che gli elefanti possono indicare con la proboscide alzata o abbassata, rispettivamente le cifre 1 e 0. Abbiamo anche visto come queste cifre vengono usate, nella logica degli elefanti e degli uomini.

Inoltre, gli elefanti, opportunamente addestrati, sono in grado di usare lo stesso codice per indicare i numeri del sistema binario usato dall'uomo e dal calcolatore.

Una fila di elefanti sufficientemente lunga può indicare qualsiasi numero del sistema binario. Ad esempio in Fig. 5-1 è indicato il numero binario 0101, cioè il numero 5.

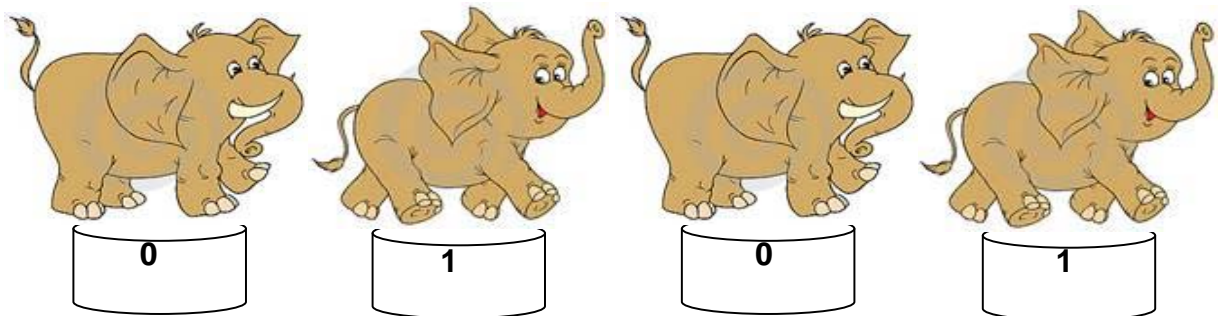


Fig. 5-1

Il gioco del Flip-Flop

Qualche volta al circo elefanti e scimmiette ammaestrate giocano al "flip-flop". Una scimmietta si mette a giocare con la coda dell'elefante: ogni tanto la solleva, poi l'abbassa e così via. Ad un certo istante il domatore schiocca la frusta e allora l'elefante regola la posizione della sua proboscide su quella della coda: solleva la proboscide se in quell'istante la coda è sollevata oppure l'abbassa se la coda è abbassata.

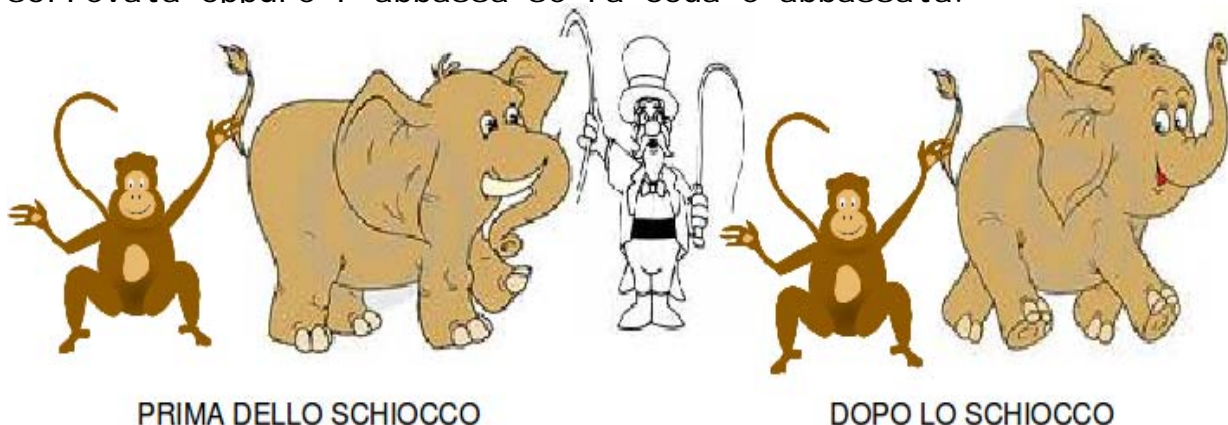


Fig. 5-2

Da quell'istante la scimmietta può continuare a giocare con la coda dell'elefante, ma la proboscide rimane sempre nella stessa posizione fino a un nuovo schiocco di frusta, quando l'elefante



regolerà nuovamente la posizione della sua proboscide sul valore "alto" o "basso" della coda.

In pratica l'elefante ha nel gioco del flip-flop due soli "stati" possibili: lo stato "flip", indicato dalla proboscide alzata, e lo stato "flop", indicato dalla proboscide abbassata. Osserva che la proboscide alzata o abbassata "ricorda" la posizione della coda all'istante in cui il domatore ha schioccato la frusta e la ricorda per sempre, a meno di un nuovo schiocco di frusta.

Questo è un segno della prodigiosa memoria degli elefanti (o dei calcolatori?)

Abbiamo visto che la proboscide alzata viene usata dagli elefanti per indicare il valore "1", mentre la proboscide abbassata viene usata per indicare il valore "0". Lo stesso codice vale anche per la posizione della coda.

Allora possiamo dire che lo stato della proboscide dell'elefante, alzata o abbassata, ricorda il valore "1" oppure "0" impostato dalla scimmietta sulla coda al momento dell'ultimo schiocco di frusta. Da quell'istante in poi, sino al successivo schiocco di frusta, la scimmietta può variare il "valore della coda", ma l'elefante conserverà lo stesso stato e indicherà con la proboscide sempre lo stesso valore.

Il Flip-Flop elettronico

Il gioco del Flip-Flop è molto importante perchè rappresenta il principio fondamentale della memoria del calcolatore. Il circuito disegnato in Fig. 5-3 attua elettronicamente il gioco che abbiamo visto e si chiama proprio "Flip-Flop".

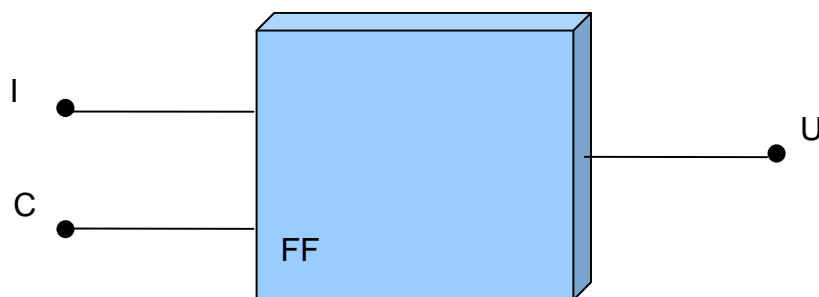


Fig. 5-3

Questo circuito ha due morsetti d'ingresso I e C, ed un morsetto di uscita U. Il morsetto di ingresso I corrisponde alla coda dell'elefante, il morsetto di ingresso C corrisponde invece alla frusta del domatore e il morsetto di uscita U corrisponde alla proboscide dell'elefante. La caratteristica fondamentale di questo circuito è quella di avere soltanto due stati. Nello stato "Flip" il livello di tensione sul morsetto di uscita U è alto (valore logico "1") e corrisponde all'elefante con la proboscide alzata.



Nello stato "Flop" il livello di tensione sul morsetto di uscita è basso (valore logico "0") e l'elefante ha la proboscide abbassata.

Ai due morsetti di ingresso I e C vengono applicati segnali elettrici con caratteristiche di diverse.

All'ingresso I viene applicato un segnale per cui la tensione assume un livello alto o basso (ossia i valori logici 1 o 0) per lunghi intervalli di tempo.

All'ingresso C è invece applicato, ogni tanto, un "impulso" di tensione, ossia un segnale molto breve, migliaia di volte più breve di un fulmine e, ovviamente, milioni di volte meno intenso. Qualcosa come una brevissima scossetta, l'analogo dello schiocco di frusta.

Nell'istante in cui arriva l'impulso all'ingresso C, il flip-flop entra nello stato "flip" se l'ingresso I è alto (valore logico "1") oppure entra nello stato "flop" se l'ingresso I è basso (valore logico "0").

Nel primo caso l'uscita U è alta (valore "1"); nel secondo caso è bassa (valore "0"), come in fig. 5-4.

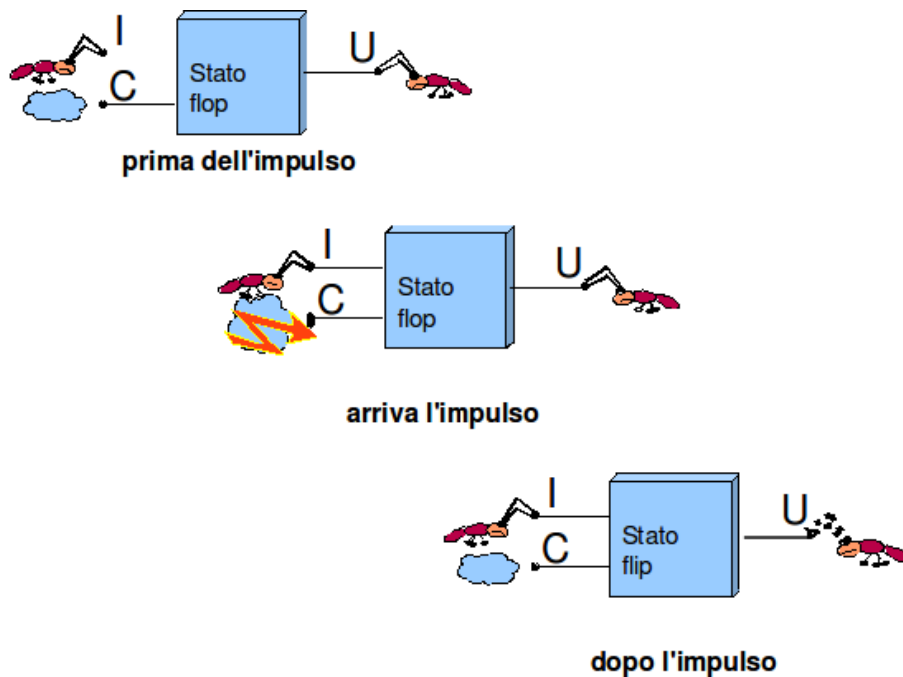


Fig. 5-4

Il flip-flop rimarrà nello stesso stato, ossia "ricorderà" il valore 1 oppure 0 sino all'istante in cui verrà applicato un nuovo



impulso all'ingresso C.

In quel momento il flip-flop conserverà il vecchio stato o cambierà il proprio stato in base al valore di ingresso I in quell'istante.

Se non venisse più applicato alcun impulso al morsetto C, il flip-flop rimarrebbe nello stesso stato per sempre e l'uscita conserverebbe per sempre il suo valore.

In sostanza il flip-flop "ricorda" il valore dell'ingresso I nell'ultimo istante in cui è stato applicato un impulso a C.

Quindi il flip-flop è la più piccola unità di memoria, fatta per ricordare un bit. Se l'uscita è 1, il valore ricordato è 1; altrimenti è 0.

Certamente avete notato che il morsetto di ingresso è stato chiamato I, iniziale della parola "ingresso", mentre il morsetto di uscita è stato chiamato U, iniziale della parola "uscita".

Probabilmente vi siete domandati perchè il morsetto dove arriva l'impulso di tensione è stato chiamato C.

La lettera C è l'iniziale della parola inglese "clock", che significa "orologio". Infatti, come vedremo meglio in seguito, ogni calcolatore ha un "clock" che dà il tempo al calcolatore, come avviene in un orologio. La differenza è rappresentata dal fatto che un orologio avanza ogni secondo, mentre un calcolatore ha un "clock" velocissimo, che produce milioni o miliardi di impulsi al secondo.

In futuro, se continuerai a studiare informatica, potrai vedere come sono fatti i flip-flop. Devi comunque sapere che questi sono a loro volta costituiti dai circuiti logici OR, AND e NOT che abbiamo già visto, variamente interconnessi.

Il Registro

Un registro è l'analogo di una fila di elefanti (e di scimmiette) i quali contemporaneamente sentono lo schiocco di frusta e regolano insieme la posizione delle proboscidi sulla posizione delle rispettive code.

Osserviamo la Fig. 5-5 e vediamo una fila di flip-flop i cui morsetti C sono tutti collegati fra loro, in modo che un impulso applicato ad uno di essi sia contemporaneamente ed istantaneamente applicato a tutti.

Una serie di flip-flop come questa costituisce un **registro** ed è proprio la versione elettronica della fila di elefanti.

Una serie di flip-flop con i morsetti C in comune costituisce un



registro.

Allo stesso modo in cui un flip-flop ricorda un singolo bit (0 oppure 1), così un registro ricorda una successione di bit, ossia un numero.

Ad esempio, in Fig. 5-5 tutti i flip-flop sono nello stesso stato "flop" con l'eccezione del secondo da sinistra che è nello stato "flip". Quindi il registro memorizza il numero binario 01000000, che corrisponde al numero decimale 64.

Ovviamente per scrivere 01000000 nel registro si sono applicati i valori 0 1 0 0 0 0 0 0 agli ingressi I_7, I_6, \dots, I_0 dei singoli flip-flop nello stesso istante in cui è stato applicato l'ultimo impulso sui morsetti C.

In un calcolatore ci sono molti registri, disseminati un po' ovunque, come vedremo. Essi costituiscono una prima forma di memoria del calcolatore. Infatti, il registro indicato in Fig. 5-5 "ricorda" il valore in esso memorizzato sino al momento in cui arriverà un nuovo impulso sul morsetto C e nuovi valori saranno memorizzati nei diversi flip-flop del registro.

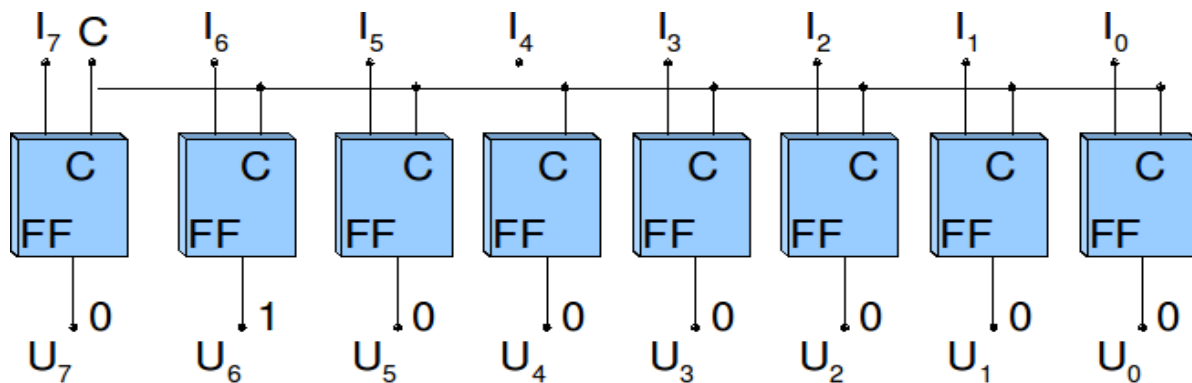


Fig. 5-5

Osservate che i morsetti di uscita sono stati chiamati U_7, U_6, U_5, U_0 andando nell'ordine decrescente da 7 a 0, e non viceversa.

La ragione è semplice: se un registro contiene un numero, la cifra più significativa viene scritta per prima come avviene nel sistema decimale ove si scrivono le migliaia, poi le centinaia, poi le decine e infine le unità.

ESERCIZI

1. Come è possibile, con una fila di elefanti, indicare un numero binario.



2. Come si comporta l'elefante con la sua proboscide nel gioco del flip-flop, quando la scimmietta muove la sua coda?
3. Qual è la funzione dello schiocco di frusta nel gioco del flip-flop?
4. A quale valore corrisponde la proboscide abbassata dell'elefante?
5. Specifica i due stati possibili del flip-flop.
6. Nel flip-flop elettronico, qual è il corrispondente del colpo di frusta che si usa nel gioco del flip-flop?
7. A quale valore logico corrisponde lo stato "flop" di un circuito logico sequenziale?
8. Spiega con parole tue la funzione dell'ingresso C di un circuito logico sequenziale flip-flop.
9. Qual è l'uscita del flip-flop, se l'ingresso I è alto, nell'istante immediatamente successivo all'applicazione dell'impulso all'ingresso C?
10. Come si comporta il flip-flop finché non viene applicato un nuovo impulso al morsetto C?
11. Quali valori può ricordare il flip-flop? (0 o 1)
12. Che cos'è un registro?
13. Che cosa ricorda un registro? (un numero; nel caso in cui i flip-flop siano 8 un byte)
14. Quali sono i flip-flop nello stato flip e quali sono invece quelli nello stato flop su un registro a otto flip-flop che ha memorizzato il numero 50?
15. Disegna un registro composto da 4 flip-flop che abbia memorizzato il numero binario 0110. Quali sono i flip-flop nello stato "flip"?
16. Come è possibile memorizzare un numero su un registro? (Si devono applicare i valori 0 o 1, corrispondenti al numero che si vuole memorizzare, agli ingressi dei singoli flip-flop nell'istante in cui è stato applicato l'ultimo impulso sui morsetti C).



6 . Da un registro all'altro

Osserva la Fig. 6-1: sono rappresentati un registro A di 4 bit (ossia 4 flip-flop) e un registro B, anch'esso di 4 bit. Gli ingressi del registro A per ora non sono importanti, osserviamo invece le 4 uscite del registro A.

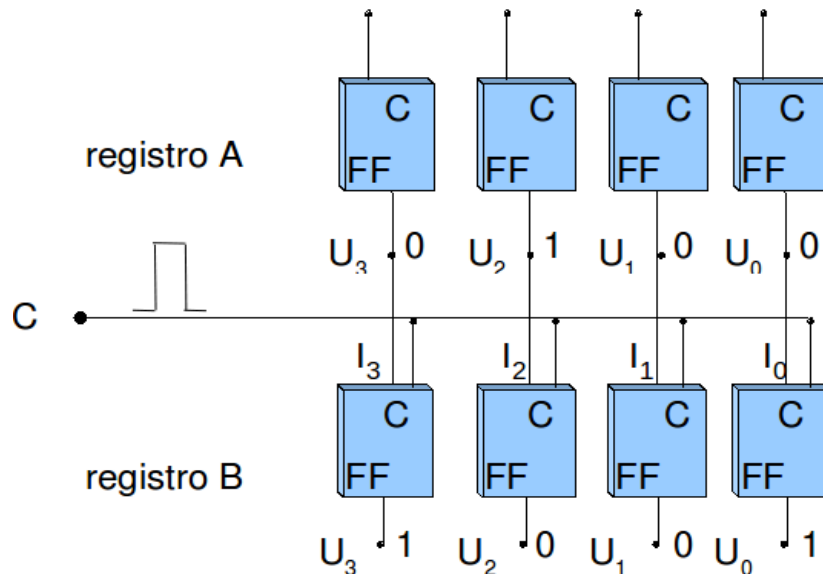


Fig. 6-1

Supponiamo che le 4 uscite, chiamate U_3, U_2, U_1 e U_0 , siano direttamente collegate ai 4 ingressi del registro B, I_3, I_2, I_1 e I_0 .

In un certo istante i 4 flip-flop del registro A sono negli stati flop, flip, flop e flop, come si vede dai valori delle uscite ($U_3=0, U_2=1, U_1=0$ e $U_0=0$).

Invece, i 4 flip-flop del registro B sono negli stati flip, flop, flop e flip ($U_3=1, U_2=0, U_1=0$ e $U_0=1$).

A contiene quindi il numero binario 0100 e B il numero 1001.

Osservate che abbiamo chiamato U_3, U_2, U_1 e U_0 i morsetti di uscita dei due registri A e B andando nell'ordine decrescente da 3 a 1, e non viceversa.

La ragione è semplice: se un registro contiene un numero, la cifra più significativa viene scritta per prima come avviene nel sistema decimale dove si scrivono prima le migliaia, poi le centinaia, poi le decine e infine le unità.



Ad un certo istante arriva un impulso sul morsetto C del registro B, che come abbiamo detto coincide con il morsetto C di tutti i flip-flop di B.

I 4 flip-flop di B ricopiano quindi al loro interno i valori degli ingressi corrispondenti.

Ad esempio, il flip-flop di sinistra, che era nello stato flip, passa nello stato flop, perchè il suo ingresso I_3 è 0 nell'istante in cui arriva l'impulso di C. In tal modo la sua uscita U_3 che valeva 1, diventa 0.

In sostanza, dopo l'impulso il registro B contiene anch'esso 0100, come A. Il valore di A è stato ricopiato o trasferito in B, mentre A conserva il suo valore. (Fig. 6-2)

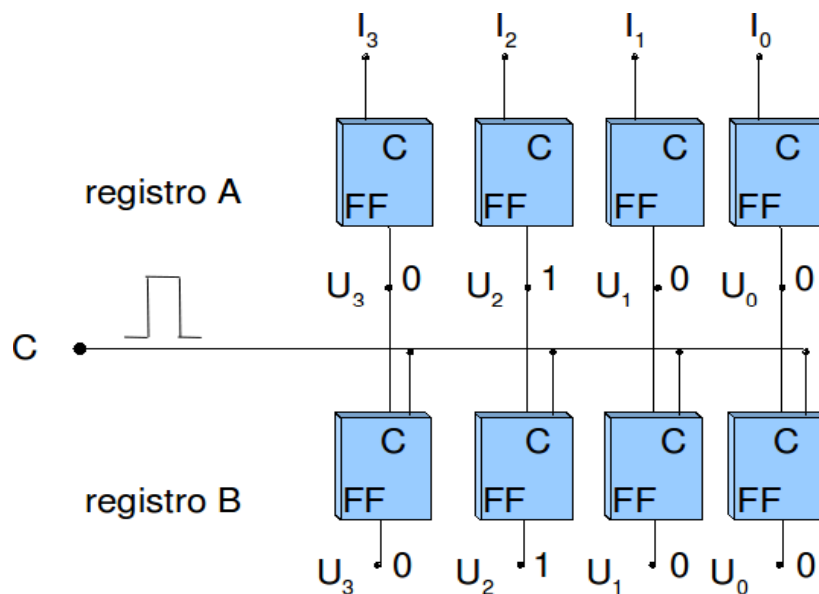


Fig. 6-2

Il circuito di Fig. 6-1 può essere complicato con l'aggiunta di un segnale di "abilitazione" o di "controllo".

Vediamo di cosa si tratta.



Nelle figure 6-3 e 6-4 è rappresentato lo stesso circuito logico AND che abbiamo descritto nel capitolo 2. Qui tuttavia è utilizzato in modo diverso, avendo aggiunto un segnale T che opera come segnale di controllo.

Per comprendere la nuova modalità di funzionamento dobbiamo ricordare che l'uscita di quel circuito AND vale 1 quando entrambi gli ingressi sono uguali a 1. Supponiamo ora che al morsetto C sia applicato un segnale come quello indicato in fig. 6-3, mentre all'ingresso I è applicato il segnale 1.

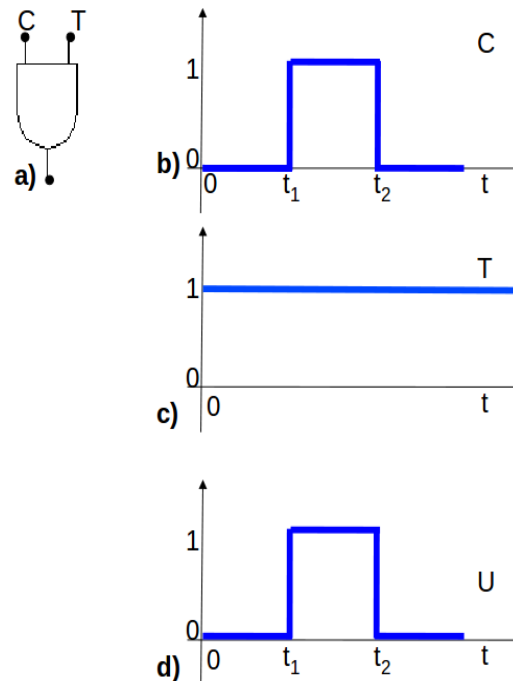


Fig. 6-3

Alcune figure saranno probabilmente una novità per te. Esse sono chiamate "diagrammi temporali" perchè riportano il valore di una grandezza al variare del tempo. Così il diagramma in Fig. 6-3 b) ci dice dall'istante iniziale, o istante 0, sino all'istante t_1 il segnale C vale 0; poi, dall'istante t_1 all'istante t_2 , C è uguale a 1; e dopo t_2 C vale sempre 0. Invece il segnale T vale sempre 1 (Fig. 6-3 c).

Siccome $0 \times 1 = 0$ e $1 \times 1 = 1$, l'uscita U vale 1 soltanto



nell'intervallo di tempo fra t_1 e t_2 (Fig. 6-3 d).

Il segnale C è un segnale molto breve, ma ha una forma molto simile a quella rappresentata in Fig. 6-3 b).

In conclusione, quando il segnale T vale 1 l'impulso di orologio, o clock, passa attraverso il circuito, come si vede bene dal confronto fra la fig. 6-3 b) e la fig. 6-3 d).

Tenete presente che in un personal computer la durata degli impulsi di clock, ossia dei rettangolini che vediamo in Fig. 6-3 b) e 6-3 d), è nettamente inferiore al nanosecondo, ossia al miliardesimo di secondo.

Viceversa, quando il segnale T vale 0, siccome $0 \times 0 = 0$ e $1 \times 0 = 0$, l'uscita U vale sempre 0, come mostrato in fig. 6-4.

Quindi se $T = 0$, il segnale non passa attraverso il circuito, come mostrato in Fig. 6-4

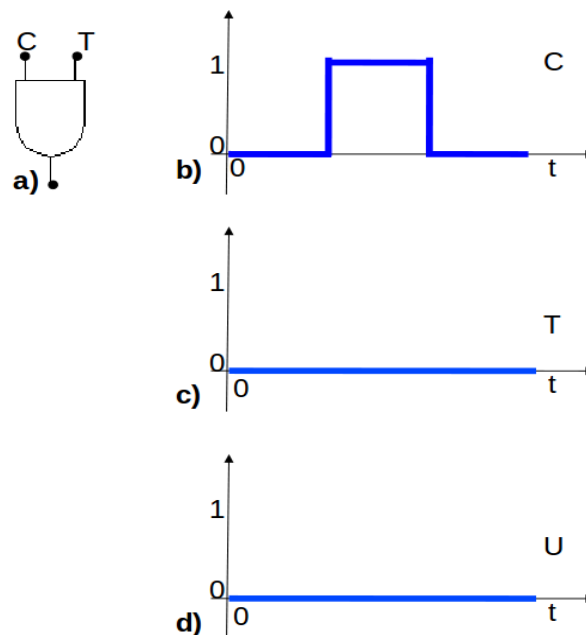


Fig. 6-4

Esami ni amo ora la Fig. 6-5.

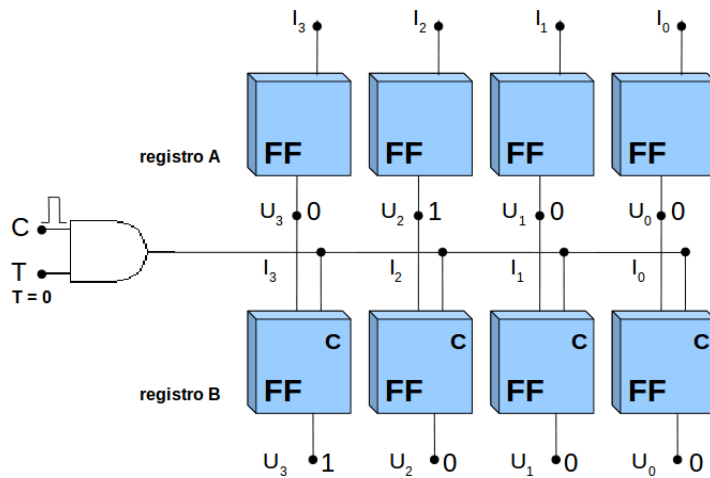


Fig. 6-5

Supponiamo, ad esempio, che il registro A contenga 0100 mentre il registro B contiene 1000.

Supponiamo inoltre che il segnale di abilitazione T sia uguale a 0.

In un certo istante arriva l'impulso di clock sul morsetto C, ma questo impulso, essendo $T = 0$, non passa attraverso il circuito AND, come mostrato in Fig. 6-4, e non arriva ai morsetti C dei vari flip-flop del registro B.

Di conseguenza l'impulso di clock non produce alcun risultato e il registro B conserva il suo valore iniziale.

Consideriamo ora la Fig. 6-6 supponendo che i registri A e B inizialmente contengano gli stessi valori che abbiamo visto in Fig. 6-5 (A = 0100 e B = 1000) e che sia $T = 1$.

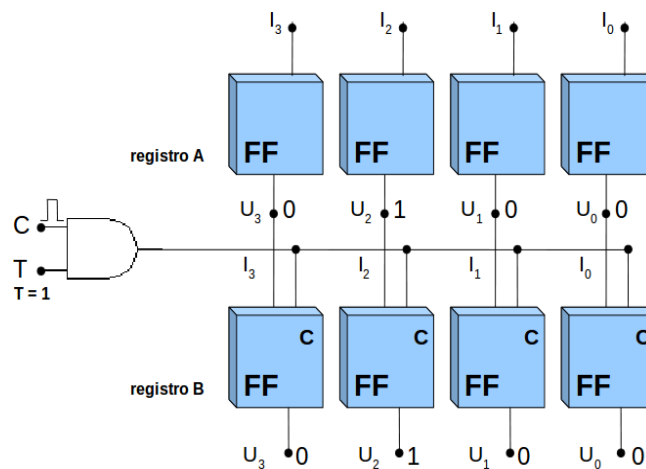


Fig. 6-6



A un certo punto arriva l'impulso di clock sul morsetto C, ma a differenza di quanto illustrato in Fig. 6-5, ora l'impulso di clock passa attraverso il circuito di abilitazione, essendo $T = 1$, e arriva sui morsetti C di tutti i flip-flop del registro B.

Di conseguenza dopo l'arrivo dell'impulso di clock, il valore di ogni flip-flop del registro A sarà ricopiato sul corrispondente flip-flop del registro B, per cui B assumerà lo stesso valore di A.

Il segnale T è prodotto da altri circuiti, come ad esempio, l'unità di controllo che descriveremo nel capitolo 9. Sono questi circuiti esterni rispetto ai due registri A e B che decidono quando l'impulso di clock deve determinare il trasferimento del contenuto di A nel registro B.

Possiamo semplificare i disegni e rappresentare il circuito di Fig. 6-6 con lo schema di Fig. 6-7.

Osserva che REG A (4) indica che A è un registro di 4 bit e che la freccia a doppio tratto indica un fascio di fili (chiamato "bus") costituito in questo caso da 4 collegamenti.

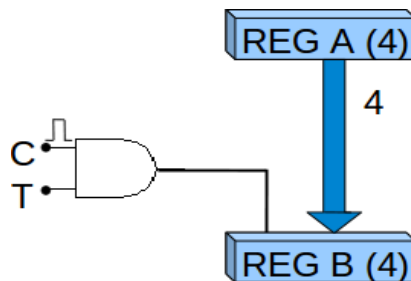


Fig. 6-7

Esercizi

1. Quale numero contiene un registro a 4 bit se i suoi 4 flip-flop sono negli stati:

fl op fl op fl ip fl op	[0010, 02]
fl ip fl ip fl op fl op?	[1100, 12]
fl op fl ip fl op fl op	[0100, 04]
fl ip fl ip fl ip fl op	[1110, 14]
fl ip fl op fl op fl ip	[1001, 09]
fl ip fl op fl op fl op	[1000, 08]



2. Spiega come avviene il trasferimento del valore contenuto in un registro ad un altro registro i cui ingressi sono collegati alle corrispondenti uscite del primo.
3. Disegna un circuito che trasferisca il valore 1110 dal registro A al registro B che contiene 1000. (vedi Fig. 6-1)
4. Due registri che sono collegati fra loro attraverso le uscite di uno e le entrate dell'altro possono contenere valori diversi in un certo istante?
5. Prova a tracciare lo schema di un circuito simile a quello di Fig. 6-1 per registri a 8 bit.



6S . SCHEDA DI APPROFONDIMENTO - TRASFERIMENTI MULTIPLI TRA REGISTRI

Trasferimento da molti a uno

Osserviamo come si comporta il circuito disegnato in Fig. 6S-1 con $T_A = 1$, $T_B = 0$ e $T = 1$.

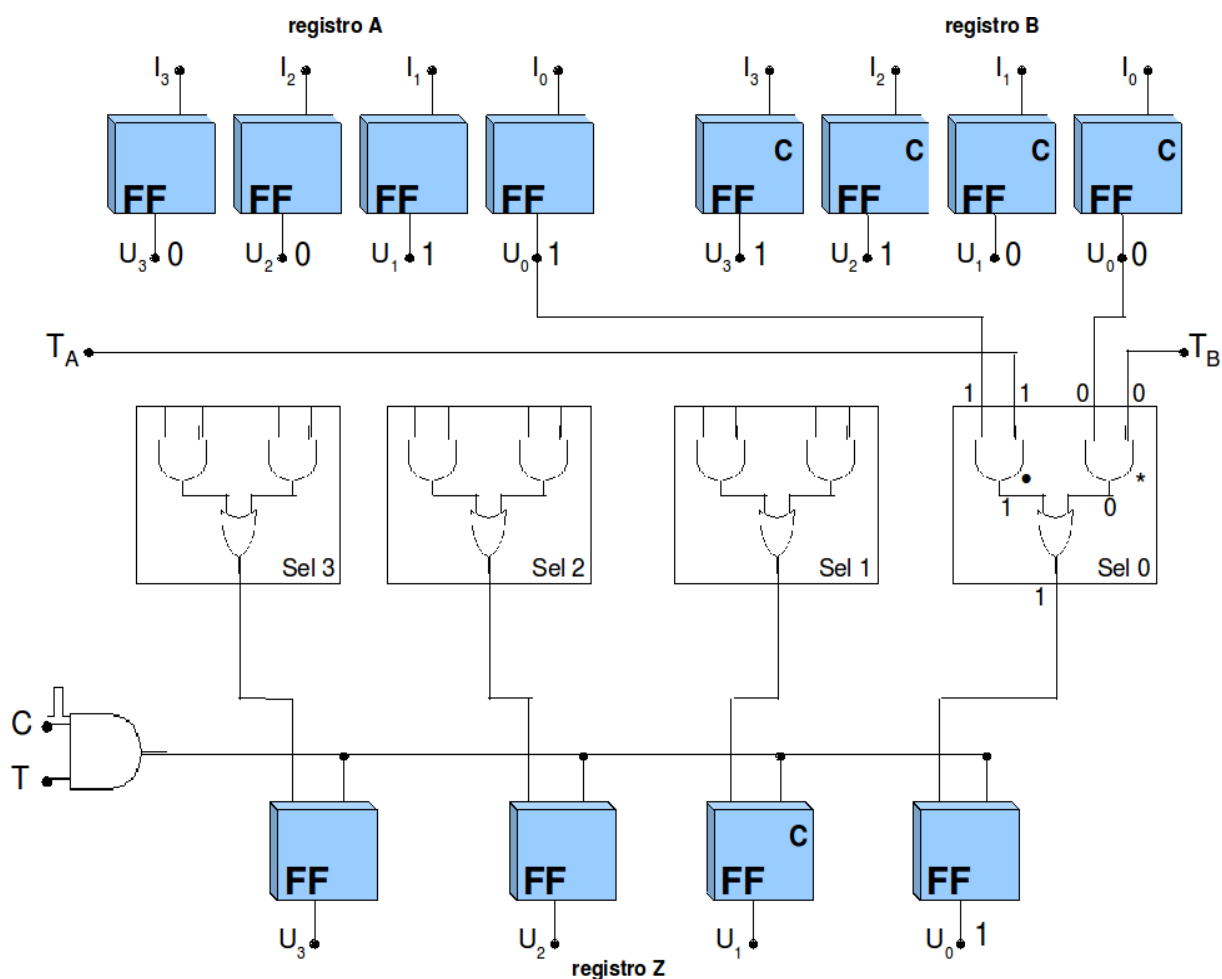


Fig. 6S-1

Cominciamo a esaminare il circuito che abbiamo chiamato Sel 0, trascurando tutti gli altri collegamenti del circuito.

Scopriamo subito che l'uscita del circuito AND in alto a destra, quello con l'*, vale 0 perchè $T_B = 0$ e inoltre l'ingresso proveniente dall'uscita U_0 del registro B fosse stata uguale a 1, l'uscita del circuito AND contrassegnato con * sarebbe stata uguale a 0, perchè $T_B = 0$.



Invece l'uscita del circuito AND contrassegnato con \bullet vale 1, perchè sia T_A sia l'uscita U_0 del registro A valgono 1. Se l'uscita U_0 del registro A fosse stata uguale a 0, l'uscita del circuito AND contrassegnato con \bullet sarebbe stata uguale a 0.

Vediamo quindi che quando $T_A = 1$ e $T_B = 0$, l'uscita del circuito AND contrassegnato con \bullet è sempre uguale all'uscita U_0 del registro A, mentre l'uscita del circuito AND contrassegnato con $*$ è sempre uguale a 0.

Di conseguenza l'uscita del circuito OR di **Sel 0**, quando $T_A = 1$ e $T_B = 0$ è esattamente uguale all'uscita U_0 del registro A. Analogamente se fossero $T_A = 0$ e $T_B = 1$, l'uscita di Sel 0 sarebbe uguale all'uscita U_0 del registro B.

In sostanza, il circuito **Sel 0** "seleziona" l'uscita U_0 del registro A o la corrispondente uscita U_0 del registro B. Quando:

$T_A = 1$ e $T_B = 0$ sull'uscita Sel 0 "passa" l'uscita U_0 di REG A

$T_A = 0$ e $T_B = 1$ sull'uscita Sel 0 "passa" l'uscita U_0 di REG B

Sempre nel caso in cui $T_A = 1$ e $T_B = 0$, vediamo cosa succede se anche $T = 1$.

È evidente che l'uscita di **Sel 0**, quando arriva un impulso sul morsetto C, viene trasferita sul quarto flip-flop del registro Z

I circuiti Sel 1, Sel 2 e Sel 3 operano nello stesso modo, lavorando però sulle uscite U_1 , U_2 e U_3 dei due registri A e B.

Il circuito di Fig. 6S-1 può essere disegnato in modo molto più semplice utilizzando lo schema di Fig. 6S-2.

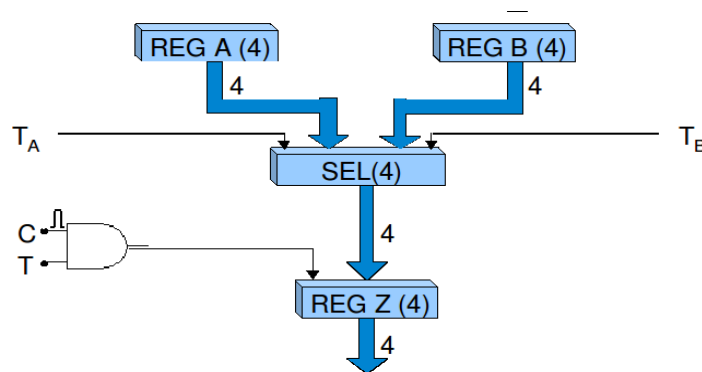


Fig. 6S-2



Si noti che i circuiti che stanno a monte del circuito di Fig. 6S-1 (generalmente, i circuiti dell'unità di controllo che vedremo nel capitolo 9) si comportano in modo tale che siano sempre $T_A = 0$ e $T_B = 1$, oppure $T_A = 1$ e $T_B = 0$. Non succede mai che siano contemporaneamente $T_A = 1$ e $T_B = 1$.

In sostanza, se $T = 0$, non avviene alcun trasferimento perchè l'impulso di clock non arriva al registro Z. Se invece $T = 1$, sul registro Z sarà trasferito il valore di REG A se $T_A = 1$ oppure, in alternativa, il valore di REG B se $T_B = 1$.

In Fig. 6S-2 abbiamo aggiunto una novità rispetto ai componenti della Fig. 6-7: il selezionatore SEL(4).

SEL(4) riceve due "bus" in ingresso: quello proveniente da REG A e quello proveniente da REG B. Entrambi sono di 4 fili, come indicato dalla scritta SEL(4).

Si possono realizzare selezionatori anche molto più complicati di quello di segnato in Fig. 6S-1.

Ad esempio, nel selezionatore disegnato in Fig. 6S-3 i morsetti T_A T_B T_C e T_D hanno il compito di ordinare la selezione desiderata, precisando al circuito SEL quale dei quattro bus di ingresso deve essere collegato.

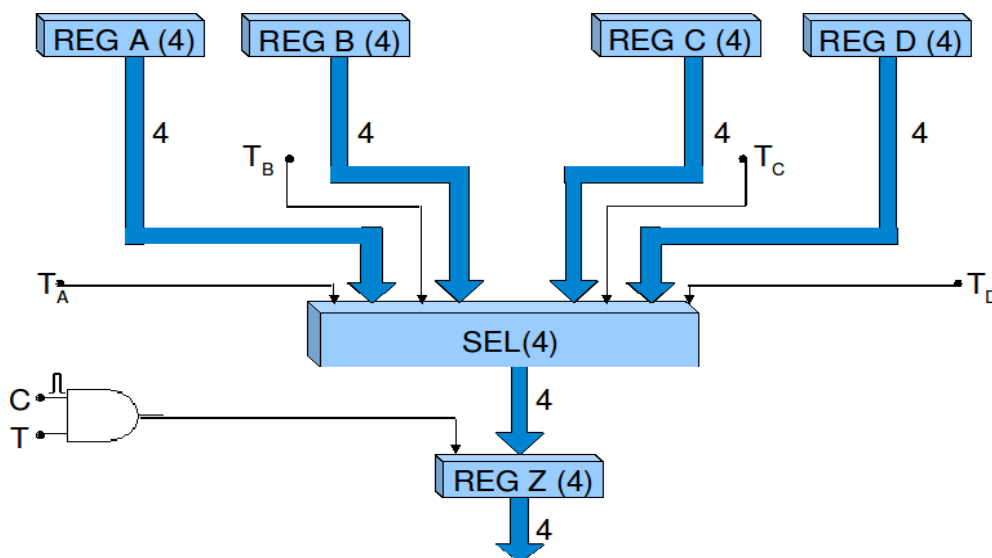


Fig. 6S-3

Quando T_A è uguale a 1 (e $T_B = T_C = T_D = 0$) i valori dei singoli flip-flop di REG A attraversano il circuito SEL(4) e diventano gli ingressi dei flip-flop di REGZ.



Quando T_B è uguale a 1 (e $T_A = T_C = T_D = 0$) i valori dei flip-flop di REG B diventano gli ingressi di REG Z.

Si noti che soltanto uno dei quattro segnali T_A , T_B , T_C e T_D è attivo in un certo istante.

Il morsetto T ha un compito più semplice, che abbiamo già visto: semplicemente T ordina la ricopiatura dell'informazione in ingresso sul registro che segue.

I circuiti selezionatori del tipo di quelli rappresentati in Fig. 6S-1 e 6S-3 sono molto importanti.

Questi circuiti sono l'analogo degli scambi ferroviari, in quanto "selezionano" quale dei due o dei quattro binari di ingresso deve essere portato all'uscita.

Trasferimento da uno a molti

Il trasferimento da un registro A a uno o più registri destinazione (cioè da uno a molti) è più facile del trasferimento che abbiamo ora visto, da più registri a un registro selezionato (cioè da molti a uno).

Osservando la figura 6S-4, se $T_X = 1$, quando arriva l'impulso sul morsetto C, il contenuto di REG A viene trasferito su REG X. Se $T_Y = 1$, quando arriva l'impulso sul morsetto C, il contenuto di REG A viene trasferito su REG Y.

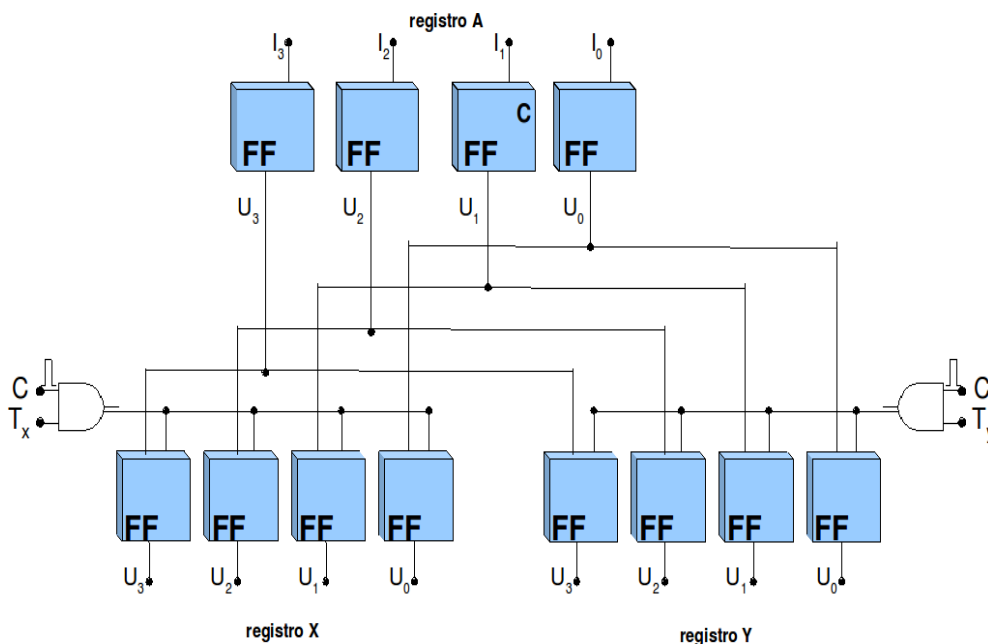


Fig. 6S-4



Inoltre, se $T_x = T_y = 1$, il contenuto di REG A viene trasferito su entrambi REG X e REG Y.

Lo schema sintetico del trasferimento uno a molti è riportato in Fig. 6S-5.

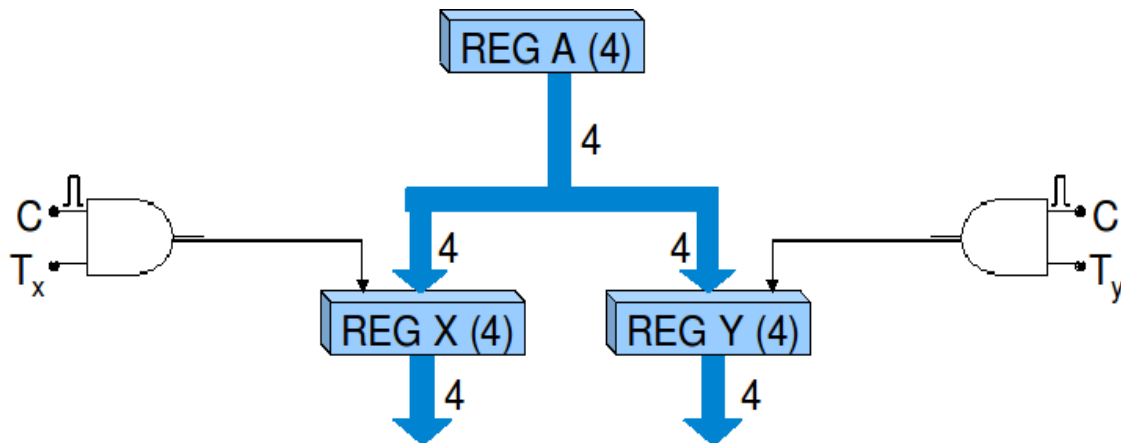


Fig. 6S-5

Esercizi

1. Quale numero contiene un registro a 4 bit se i suoi quattro flip-flop sono negli stati flip, flip, flop e flip?
2. Spiega come avviene il trasferimento del valore contenuto in un registro ad un altro registro i cui ingressi sono collegati alle corrispondenti uscite del primo.
3. Disegna un circuito per il trasferimento di un valore da un registro ad un altro, nel caso di registri ad 8 bit (vedi Fig. 6S-1).
4. Due registri che sono collegati fra loro, attraverso le uscite di uno e le entrate dell'altro (vedi Fig. 6S-1), possono contenere valori diversi in un certo istante?
5. Spiega come funziona il meccanismo di "abilitazione" o di "controllo" per il trasferimento del contenuto di un registro ad un altro registro.
6. Traccia lo schema di un circuito simile a quello di Fig. 6-7, con registri a 8 bit.
7. Completa il disegno di Fig. 6S-1, tracciando i collegamenti mancanti.
8. Spiega il comportamento del circuito Sel0 del circuito di Fig. 6S-1.
9. Spiega come avviene il trasferimento da un registro selezionato



fra molti registri a un registro destinazione (vedi Fig. 6S-1).

10. Spiega come avviene il trasferimento da un registro a uno o più registri destinazione (vedi Fig. 6S-4).

11. Disegna il circuito che esegue il trasferimento da un registro selezionato fra quattro registri, tutti di due bit.

12. Disegna il circuito di Fig. 6S-3 rappresentando separatamente i singoli flip-flop dei cinque registri REG A, REG B, REG C, REG D e REG E.



7 . Scheda di approfondimento - L'addizionale (Half Adder e Full Adder)

In questo capitolo impareremo come è fatto quel componente fondamentale di un calcolatore che si chiama **addizionale**.

Il compito dell'**addizionale**, che è un circuito, è eseguire la somma di due numeri, scritti in codice binario.

Supponiamo che i due numeri da sommare siano contenuti in due registri chiamati REG A e REG B (Fig. 7-1).

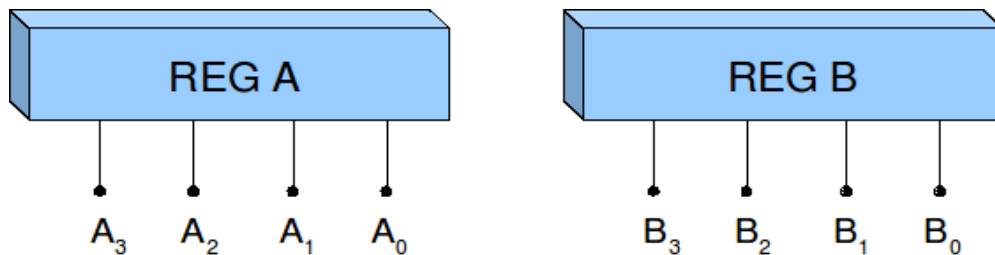


Fig. 7-1

Quando eseguiamo una somma partiamo da destra, ossia dalle cifre "meno significative", come avviene anche nel sistema decimale.

Quindi la prima operazione da eseguirsi sarà la somma di A_0 con B_0 definita dalla seguente tabella:

0	+	0	=	0	
1	+	0	=	1	
0	+	1	=	1	
1	+	1	=	0	(con il riporto di 1, perché in binario $1 + 1 = 10$)

Confronta questa tabella con la seguente tabella che illustra il funzionamento del circuito quando $A_0 = 1$ e $B_0 = 1$ con il riporto di 1.

$$\begin{array}{r}
 \text{riporti} \quad 101 \text{ —————} R_0 \\
 \hline
 \text{REG A} \quad 0101 \text{ ———+——} A_0 \\
 \text{REG B} \quad 0101 \text{ ———=——} B_0 \\
 \hline
 \quad \quad 1010 \text{ —————} S_0
 \end{array}$$



Il circuito che eseguirà questa operazione elementare, ossia la somma di $A_0 + B_0$, avrà due ingressi, A_0 e B_0 , e due uscite, l'uscita S_0 contenente la cifra binaria meno significativa della somma e il riporto R_0 . Di conseguenza il circuito elementare che esegue la somma di A_0 con B_0 sarà costituito da due parti. La prima parte eseguirà il calcolo di S_0 , definito dalla tabella

A_0	B_0	S_0
0	0	0
0	1	1
1	0	1
1	1	0

L'uscita di questo componente sarà uguale a 1 in due casi: quando $A_0 = 0$ e $B_0 = 1$ e quando $A_0 = 1$ e $B_0 = 0$.

E' facile verificare che questo componente dovrà essere fatto come indicato in Fig. 7-2.

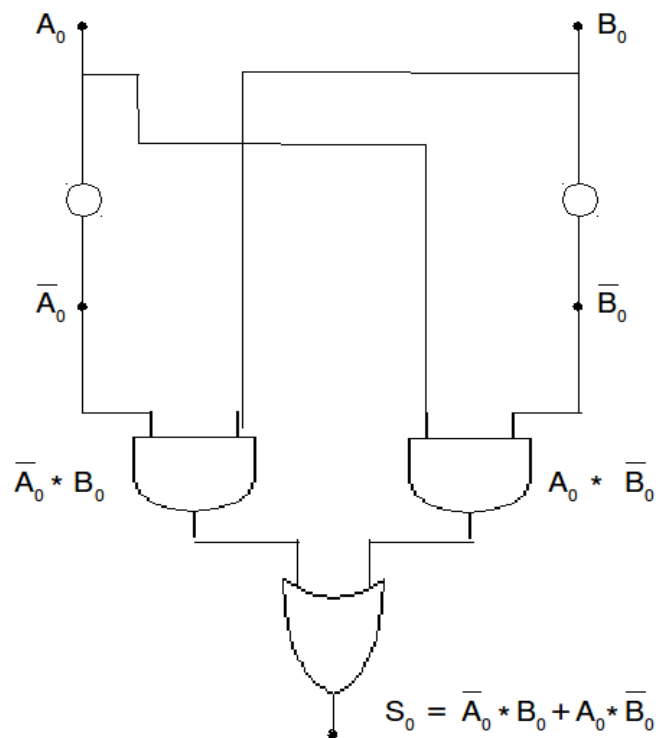


Fig. 7-2



Il secondo componente che esegue il calcolo del riporto S_0 vale 1 soltanto quando entrambi A_0 e B_0 sono uguali a 1. Quindi questo componente sarà costituito da un circuito elementare di tipo AND. (Fig. 7-3).

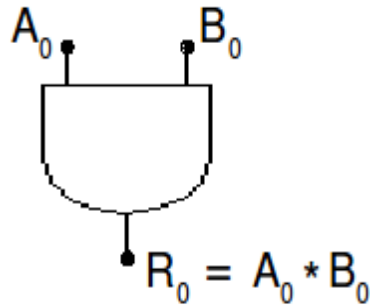


Fig. 7-3

In conclusione, il circuito che calcola la somma S_0 e il riporto R_0 è rappresentato in Fig. 7-4.

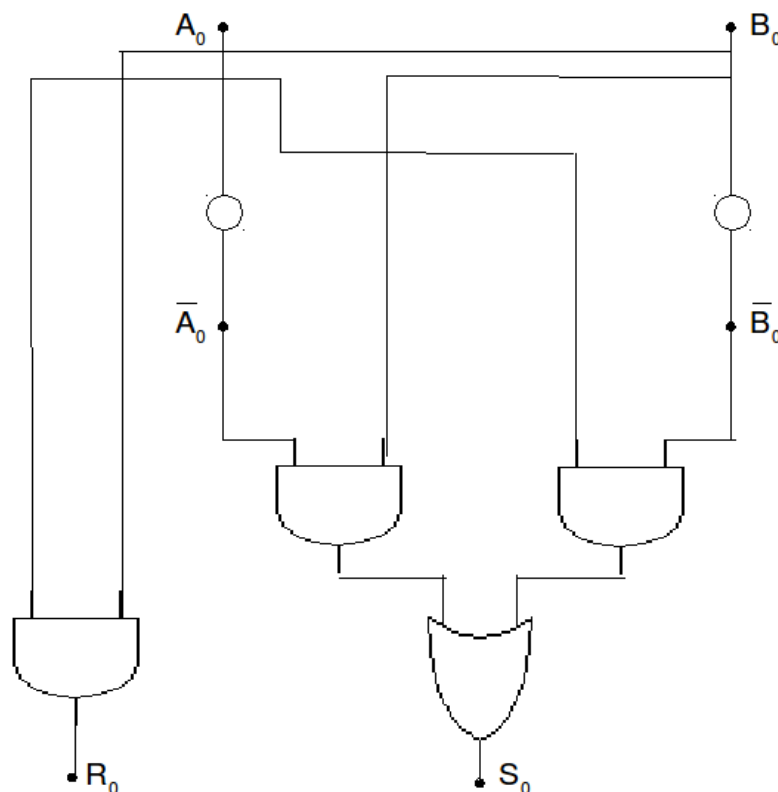


Fig. 7-4

Il circuito di Fig. 7-4 esegue soltanto la somma di A_0 e B_0 , non dovendo calcolare il contributo di un riporto precedente.



Full Adder

Più complicato è il circuito che esegue la somma di A_1 con B_1 e con il riporto R_0 della somma di A_0 con B_0 .

Questo è un circuito con 3 ingressi (A_1 , B_1 e R_0) e con due uscite (S_1 e il riporto R_1) per una nuova somma elementare. Nella tabella seguente vediamo un caso particolare del funzionamento di questo circuito.

riporti	111			R_0
				R_1
REG A	0011		+	A_1
REG B	0111		=	B_1
	1010			S_1

Il circuito ha due componenti. Il primo componente deve comportarsi come mostrato nella tabella

A_1	B_1	R_0	S_1
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

L'uscita S_1 assumerà il valore 1 in quattro casi distinti:

$$A_1 = 0 \quad B_1 = 0 \quad R_0 = 1$$

$$A_1 = 0 \quad B_1 = 1 \quad R_0 = 0$$

$$A_1 = 1 \quad B_1 = 0 \quad R_0 = 0$$

$$A_1 = 1 \quad B_1 = 1 \quad R_0 = 1$$

Di conseguenza il componente sarà costituito come indicato in Fig. 7S-5.

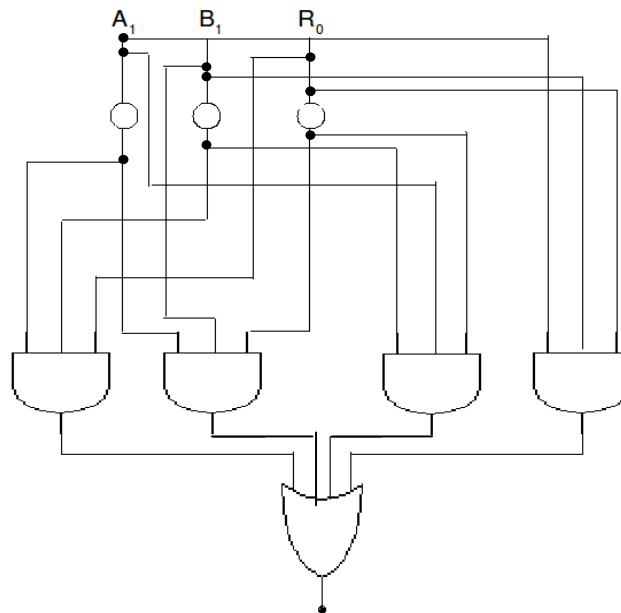


Fig. 7S-5

Il secondo componente esegue il calcolo del riporto R_1 e si comporta come indicato nella seguente tabella.

A_1	B_1	R_0	R_1
0	0	0	0
0	0	1	0
0	1	1	1
1	0	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

R_1 assume il valore 1 nei quattro casi seguenti:

$$A_1 = 0 \quad B_1 = 1 \quad R_0 = 1$$

$$A_1 = 1 \quad B_1 = 0 \quad R_0 = 1$$

$$A_1 = 1 \quad B_1 = 1 \quad R_0 = 0$$

$$A_1 = 1 \quad B_1 = 1 \quad R_0 = 1$$

Quindi possiamo scrivere

$$R_1 = \overline{A_1} * B_1 * R_0 + A_1 * \overline{B_1} * R_0 + A_1 * \overline{B_1} * \overline{R_0} + A_1 * B_1 * R_0$$

E' difficile dimostrare, ma siamo certi che ci riuscirete, che R_1 può anche essere scritto nella forma:



$$R_1 = B_1 * R_0 + A_1 * R_0 + A_1 * B_1$$

Infatti, è sufficiente che due dei tre valori A_1 , B_1 e R_0 siano entrambi uguali ad 1 perchè si generi un nuovo riporto.

Di conseguenza il componente elementare che esegue il calcolo di R_1 sarà fatto come indicato in Fig. 7S-6. In conclusione, il circuito che calcola la somma $A_1 + B_1$ e R_0 producendo S_1 e il riporto R_1 per la somma successiva $(A_2 + B_2 + R_1)$ è rappresentato in Fig. 7S-7.

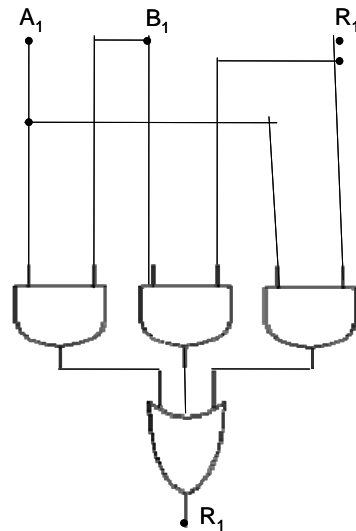


Fig. 7S-6

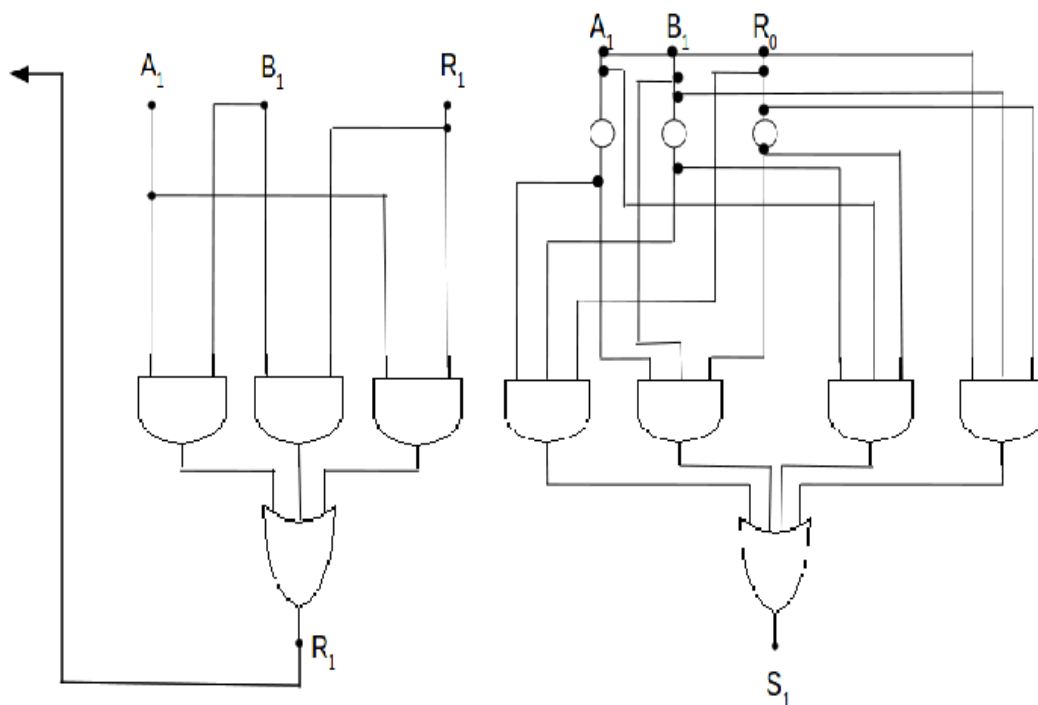


Fig. 7S-7



in Fig. 7-4, che esegiva soltanto la somma di A_0 e B_0 e non doveva lavorare sul riporto.

Per questa ragione, il circuito di Fig. 7-4 è chiamato "Half Adder" (semi addizionario) mentre quello di Fig. 7S-7 è chiamato "Full Adder" (addizionario completo). Le due denominazioni non sono molto precise ma ormai in tutto il mondo i circuiti di Fig. 7-4 e 7S-7 sono chiamati così.

Il segnale di uscita di R_1 diventerà l'ingresso di un successivo "Full Adder" che eseguirà la somma di A_2 , B_2 e R_1 e genererà un nuovo riporto R_2 .

In conclusione l'addizionario dei contenuti di REG A e REG B sarà fatto come in Fig. 7S-8.

Notate che l'addizionario lavora sempre e in ogni istante esegue la somma di REG A e REG B, ma soltanto quando arriva l'impulso di clock e il segnale di abilitazione T vale 1, il risultato della somma viene trasferito nel registro disegnato in fondo alla Fig. 7S-8. Questo sarà fatto da cinque flip-flop perchè la somma di due numeri binari di 4 cifre può richiedere 5 bit.

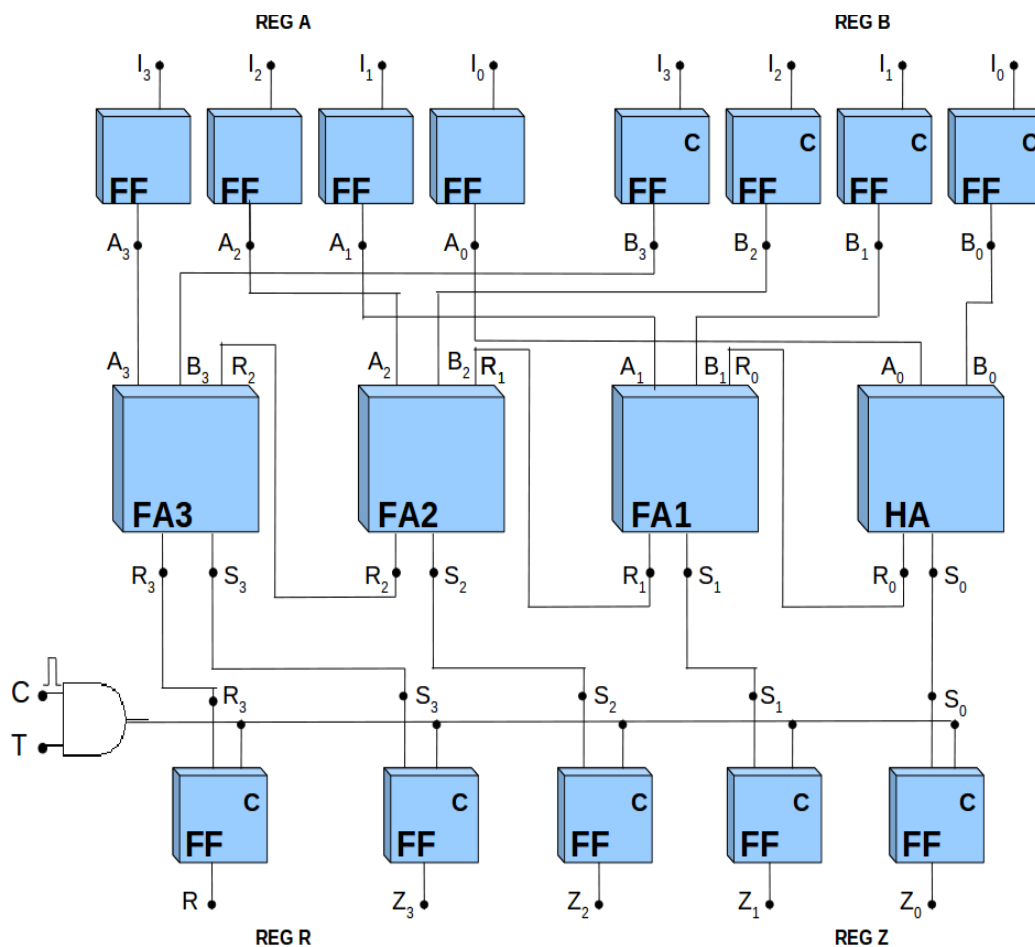


Fig. 7S-8



Esercizi

1. Completa gli schemi delle somme seguenti:

a) riporti
REG A 1011 +
REG B 0110 =

b) riporti
REG A 1100 +
REG B 0111 =

c) riporti
REG A 0111 +
REG B 1110 =

d) riporti
REG A 1001 +
REG B 1011 =

2. Qual è il compito del circuito HA dell'Addizzatore?

3. Se il circuito HA (half adder) riceve in ingresso i due valori $A_0 = 1$ e $B_0 = 0$ quali sono i valori S_0 e R_0 prodotti in uscita?

4. Qual è il compito dei circuiti FA dell'Addizzatore?

5. Quali sono i 3 ingressi del circuito full adder?



9. SCHEDA DI APPROFONDIMENTO - L'unità di controllo

Lo schema di Fig. 9-1 illustra le cinque parti fondamentali di un calcolatore programmabile. Questo schema spiega perfettamente anche l'organizzazione della grande maggioranza dei calcolatori attuali.

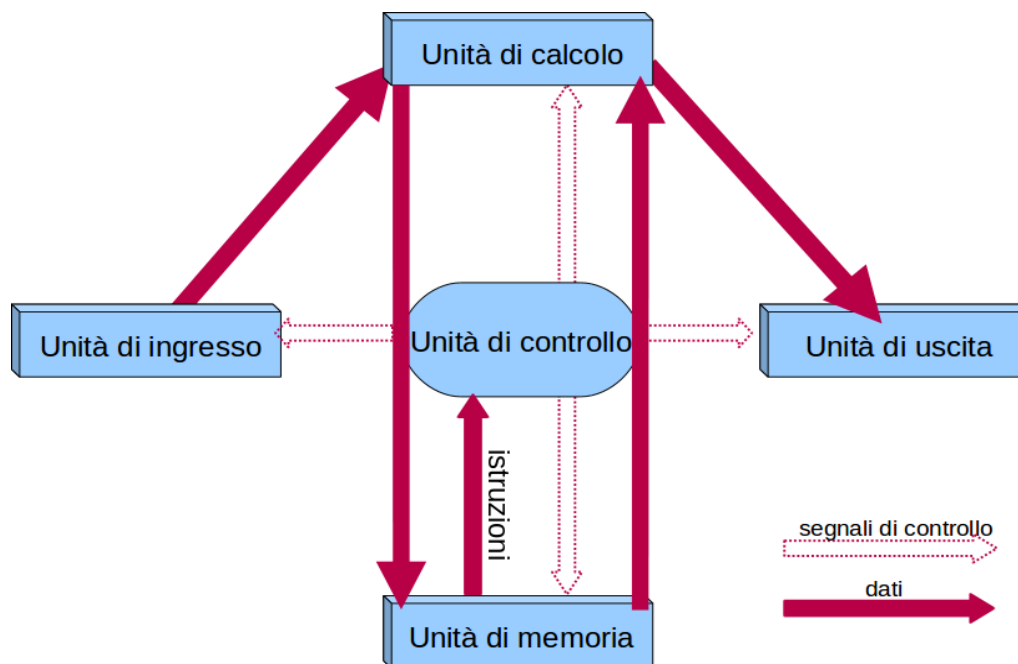


Fig. 9-1

Descriviamo brevemente i compiti delle singole unità che compaiono nello schema.

L'unità di calcolo

L'unità di calcolo (detta anche "unità aritmetica") è quella parte del calcolatore in cui vengono svolte le operazioni aritmetiche o logiche che risolvono un dato problema. Si potrebbe dire che l'unità di calcolo è l'analogo della calcolatrice tascabile.

Nell'unità di calcolo passano i dati in ingresso o in uscita. Infatti, i programmi e i dati su cui lavorare sono letti da un'unità di ingresso che li trasferisce, su ordine dell'unità di controllo, all'unità di calcolo, generalmente a un registro A. Chiameremo questo registro REGA o "registro accumulatore" dell'unità di calcolo. Analogamente i risultati del calcolo sono generalmente inviati, sempre su comando dell'unità di controllo, dall'unità di calcolo all'unità di uscita.



L'unità di ingresso

L'unità di ingresso è costituita da una pluralità di dispositivi. Il più importante è la tastiera, attraverso la quale l'utente introduce nel calcolatore dati numerici o alfabetici, cioè numeri o lettere, ed eventuali comandi.

Un secondo dispositivo di ingresso è il mouse (o "topo"), che può essere pensato come un'estensione della tastiera, attraverso il quale vengono inviati altri comandi oltre quelli inviati dalla tastiera.

Un terzo dispositivo è lo "scanner" che serve ad acquisire un'immagine, come, ad esempio, una pagina di un libro, e a trasferirla nella memoria del calcolatore.

L'elenco dei dispositivi di ingresso non finisce qui. Infatti, molti altri dispositivi possono trasferire dati alla memoria del calcolatore. Ad esempio: gli strumenti per misurare la temperatura, la pressione o altre grandezze fisiche. Molti telefoni cellulari e molte macchine fotografiche possono essere collegati a un calcolatore come dispositivi di ingresso, ed eventualmente di uscita. Infine, molti dispositivi progettati per contenere enormi volumi di informazione - anche moltissimi miliardi di caratteri - possono essere utilizzati come unità di ingresso.

L'unità di uscita

Anche l'unità di uscita è costituita da una ricca pluralità di dispositivi: i più importanti sono ovviamente il video e la stampante. Inoltre, molti dei dispositivi sopra elencati possono operare sia come unità di ingresso sia come unità di uscita.

La memoria centrale

Abbiamo già visto che la memoria centrale contiene sia il programma sia i dati su cui il programma lavora. Un calcolatore che utilizzasse due memorie centrali distinte, una per i dati e una per il programma, sarebbe certamente più semplice e facile da comprendere. Tuttavia, i calcolatori di oggi preferiscono usare una memoria unica per avere una maggiore flessibilità.

Infatti, ci sono problemi come quelli della contabilità delle aziende, che sono relativamente semplici e sono risolvibili con programmi corti, ma lavorano su grandi volumi di dati. Altri problemi, come quelli della guida di un'astronave dalla Terra a Marte, richiedono invece pochi dati, ma programmi lunghi e complessi.

Proprio per questo, se la memoria dei dati e del programma è unica, con lo stesso tipo di calcolatore possiamo risolvere sia i problemi semplici con tanti dati, sia quelli difficili con pochi dati.



Il compito dell'unità di controllo

Abbiamo visto che il funzionamento dell'unità di calcolo e della memoria centrale è comandato da opportuni segnali di controllo. Ad esempio se andiamo a rivedere nella Fig. 7S-8 il funzionamento dell'addizionatore, che è il nucleo centrale dell'unità aritmetica, scopriamo che il risultato della somma dei contenuti di REGA e di REGB viene trasferito su REGZ se il segnale sul morsetto T assume il valore 1 quando arriva un impulso sul morsetto C. I segnali indicati con la lettera T nelle Fig. 6-3, 6-4, 6-5, 6-6, 6-7, 6S-1 e seguenti sono tutti prodotti dall'unità di controllo.

Allo stesso modo, se riesaminiamo il funzionamento della memoria centrale descritto nella Fig. 8-4, vediamo che la lettura di una cella di memoria di un dato indirizzo è comandata dal segnale MEMR e dal segnale CLK.

Qualunque operazione elementare del calcolatore è basata su trasferimenti fra registri e questi trasferimenti sono determinati da segnali di controllo.

Il compito dell'unità di controllo consiste proprio nella generazione e nella trasmissione a tutte le altre unità del calcolatore di segnali di controllo, quei segnali che in Fig. 9-1 sono stati disegnati tratteggiati, per distinguerli dai segnali che portano invece dati da elaborare o elaborati.

L'orologio

Abbiamo già accennato al fatto che il funzionamento di qualunque unità del calcolatore è cadenzato da un "clock" (o orologio) che produce un lunga successione di impulsi C, a intervalli uguali di tempo, come i secondi di un cronometro. Poiché, come abbiamo visto, i trasferimenti fra registri avvengono quando arriva l'impulso C, un calcolatore è tanto più veloce quanto più rapido è il suo orologio.

Il generatore di impulsi di orologio o "clock generator" è una specie di piccolo cuore del calcolatore. Ma mentre il cuore dell'uomo batte all'incirca ogni secondo, il cuoricino del calcolatore batte milioni o miliardi di volte al secondo.

Ricorda che la parola "mega" significa "un milione" e che un herz equivale a un colpo al secondo. Quindi un calcolatore che vada alla velocità di 33 megahertz, ha il cuore che batte 33 milioni di volte al secondo.

Nel momento in cui questa pagina viene scritta alcuni tra i nuovi calcolatori operano alla frequenza di clock pari a qualche decina di gigahertz, ossia qualche decina di miliardi di impulsi al secondo, ma la velocità di clock aumenta sempre più velocemente.



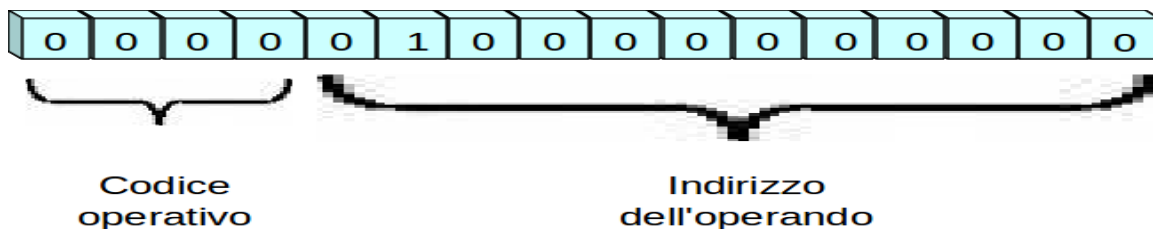
Tieni presente che occorrono alcuni impulsi di orologio per eseguire un'istruzione; così un calcolatore che abbia un orologio da 33 megahertz esegue da 5 a 10 milioni di istruzioni al secondo. L'orologio può essere considerato uno dei circuiti fondamentali dell'unità di controllo, che ha appunto il compito di inviare, oltre ai segnali di controllo, anche gli impulsi di orologio a tutte le altre unità della macchina.

Il Programma

Abbiamo visto che nella memoria centrale sono allocati non soltanto i dati da elaborare e i risultati parziali o finali del calcolo, ma anche il programma. Il programma è una successione di ordini elementari per il calcolatore, ossia di "istruzioni", scritte di norma una dopo l'altra.

Vediamo il semplice esempio di un programma, o frammento di programma, che esegue la somma dei dati contenuti nelle celle di memoria 1024 e 1025 e trasferisce il risultato nella cella di memoria 1026. Questo è un programma costituito da 3 istruzioni.

1a istruzione



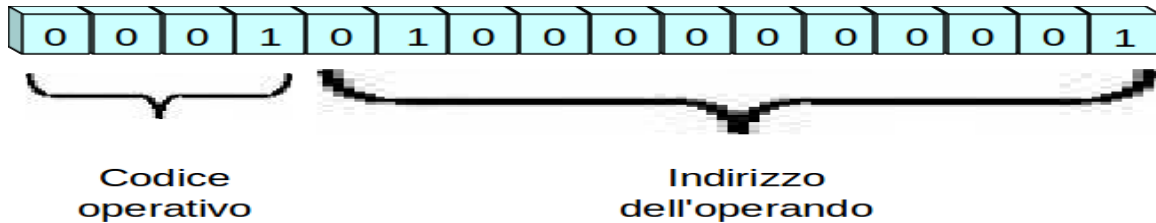
Questa istruzione, come molte altre istruzioni, è costituita da due parti distinte: il codice operativo che dice che cosa deve fare quell'istruzione, e l'indirizzo dell'operando, che specifica su quale operando lavorare.

Nel nostro esempio, il codice operativo, 0000, è il codice dell'istruzione che ordina il trasferimento al registro A (REGA) del contenuto della cella di memoria indicata nel campo "indirizzo dell'operando"

Osservate che l'indirizzo dell'operando scritto in codice binario, è l'equivalente del numero 1024. Quindi questa istruzione ordina il trasferimento nel registro REGA del contenuto della cella di memoria che ha indirizzo 1024.



2a istruzione

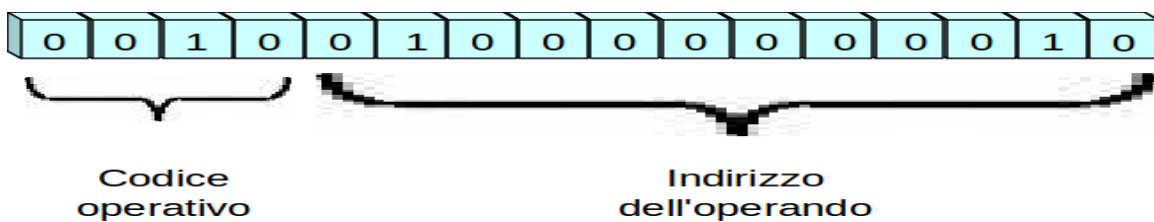


Il nuovo codice operativo, 0001, ordina l'esecuzione della somma del contenuto del registro A e del contenuto della cella di memoria il cui indirizzo è stato specificato nel campo "indirizzo dell'operando".

Nel nostro esempio, l'istruzione ordinerà la somma del contenuto di REGA (nel quale, dopo la prima istruzione, è stato trasferito il contenuto della cella di memoria 1024) con il contenuto della cella di memoria 1025. Il risultato della somma rimarrà nel registro A.

3a istruzione

Il codice operativo di questa istruzione, 0010, ordina il trasferimento del contenuto di REGA nella cella di memoria il cui indirizzo è precisato nel campo di operando. Nel nostro caso questo indirizzo vale 1026 in codice decimale.



Di conseguenza, poichè in REGA stava la somma delle celle di memoria 1024 e 1025, dopo l'esecuzione di questa terza istruzione, la somma starà nella cella di memoria di indirizzo 1026.

Acquisizione di un'istruzione

La Fig. 9-2 rappresenta lo schema di principio di un'unità di controllo. Il registro in alto è chiamato "program counter" o "P.C.", contatore di programma. È un registro lungo come il registro della memoria, MAR, e ha il compito di tenere il segno dell'istruzione a cui è arrivata l'esecuzione di un programma. Se in un certo istante, il contenuto del contatore di programma è uguale a 1000, questo significa che l'istruzione che deve essere



eseguita in quel momento è quella che è scritta nella memoria centrale e all'indirizzo 1000.

Il contatore di programma è collegato al registro di indirizzo della memoria, MAR, attraverso un fascio di fili detto "ADDBUS" o "Address bus", in quanto trasporta generalmente indirizzi.

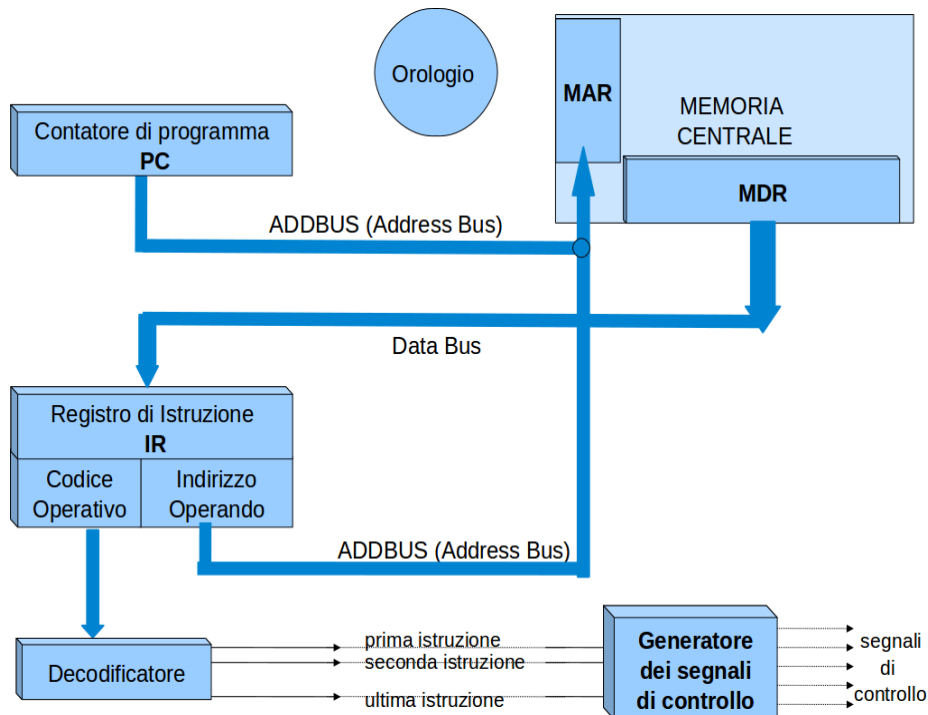


Fig. 9-2

Utilizzando opportuni segnali di comando, l'unità di controllo inizia l'esecuzione di un'istruzione, eseguendo le seguenti operazioni:

1. *trasferisce il contenuto del contatore di programma (PC) nel registro di indirizzo della memoria (MAR): da PC a MAR*
2. *trasmette il segnale di controllo MEMR (memory read) che ha come effetto il trasferimento della cella di memoria il cui indirizzo è scritto in MAR, ossia dell'istruzione che deve essere eseguita, nel registro di dato della memoria (MDR): da cella di memoria a MDR*
3. *trasferisce lungo il DATABUS il contenuto di MDR sul registro IR (registro di istruzione), che quindi conterrà l'istruzione da eseguire: da MDR a IR*
4. *incrementa di 1 il contenuto di PC (contatore di programma) in modo che questo "punti" all'istruzione successiva, ossia contenga l'indirizzo della prossima istruzione che deve essere eseguita: PC + 1 entro PC*



Le operazioni 1, 2, 3 e 4 costituiscono la fase di "acquisizione di un'istruzione". Questa fase viene chiamata nei manuali "fetch cycle" o "ciclo di ricerca" (dell'istruzione che deve essere eseguita).

Esecuzione di un'istruzione

Come si vede in Fig. 9-2, il codice operativo di un'istruzione e il suo campo di indirizzo, dopo aver raggiunto insieme il registro IR, sono trattati separatamente.

Infatti, il codice operativo viene inviato in ingresso a un decodificatore (Fig. 8-1) che ha tante uscite quanti sono i diversi tipi di istruzione. In un certo istante una sola uscita sarà uguale a 1, quella che corrisponde al tipo di istruzione che deve essere eseguita.

Ad esempio, dopo l'acquisizione dell'istruzione

```
0000 0100 0100 0000
```

sarà attiva soltanto l'uscita numero 0, corrispondente al codice 0000 dell'istruzione "trasferisci in A dalla cella di memoria indicata".

L'indirizzo dell'operando viene invece inviato ancora una volta sul bus degli indirizzi ADDBUS per accedere alla memoria centrale. Più esattamente, il contenuto del campo di indirizzo dell'istruzione viene trasferito, sull'ADDBUS, al registro di indirizzo della memoria per accedere alla cella indicata nello stesso campo di indirizzo (nel nostro caso, 0100 0100 0000)

In pratica, la seconda fase dell'esecuzione di un'istruzione, chiamata "execute cycle" (ciclo di esecuzione), consiste nelle seguenti operazioni elementari:

1. il codice operativo viene decodificato e la sua uscita è inviata al "generatore dei segnali di controllo" (Fig. 9-2). Nel nostro caso, poichè il codice operativo è 0000, l'uscita del decodificatore chiamata "prima istruzione" in Fig. 9.2 assume il valore 1, mentre tutte le altre uscite del decodificatore assumono il valore 0.
2. il campo di indirizzo dell'istruzione viene trasmesso, tramite l'ADDBUS, al MAR della memoria centrale. Nel nostro caso, i 12 fili dell'ADDBUS assumono i valori 0100 0100 0000.
3. i segnali di controllo sono generati dal "generatore dei segnali di controllo" e trasmessi all'unità di calcolo, alla memoria centrale, alle unità di ingresso e di uscita e alla stessa unità di controllo, per completare l'esecuzione dell'istruzione



La fase 3, la più complessa, è attuata dal circuito "generatore dei segnali di controllo" che, istante per istante, istruzione dopo istruzione, produrrà la corretta sequenza dei segnali di controllo necessari per il funzionamento del calcolatore. Nel nostro caso, poiché l'istruzione di codice 0000 richiede soltanto il trasferimento del dato contenuto nella cella di memoria il cui indirizzo è scritto nel campo "indirizzo operando" (0100 0100 0000) e poiché l'indirizzo dell'operando è già stato trasferito nel MAR, il generatore dei segnali di controllo dovrà soltanto attivare il segnale di controllo MEMR. Quindi l'ingresso chiamato "prima istruzione" in Fig. 9-2 sarà collegato direttamente all'uscita MEMR del generatore dei segnali di controllo.

Esercizi

1. Quali sono le 5 parti fondamentali di un calcolatore?
2. Qual è il ruolo dell'unità aritmetica nel calcolatore?
3. Qual è il ruolo della memoria centrale nel calcolatore?
4. Perché è preferibile un'unica memoria centrale nel calcolatore,
5. Cosa fa l'unità di controllo nel calcolatore?
6. Che cosa si intende dicendo che un calcolatore va alla velocità di 12 gigahertz?
7. Cosa fa il contatore di programma nell'unità di controllo?
8. Quali operazioni esegue l'unità di controllo durante il "fetch cycle"?
9. Quali operazioni esegue l'unità di controllo durante l'"execute cycle"?
10. Definisci un piccolo insieme di istruzioni e assegna a ciascuna il suo codice operativo. Definisci un piccolissimo programma e scrivilo in binario.



10. Traduzione da un linguaggio ad un altro

I programmatori dei primi calcolatori, quelli costruiti a cavallo degli anni cinquanta del secolo scorso, per impartire ordini alla macchina e scrivere programmi, usavano un linguaggio fatto soltanto di successioni di "1" e "0". Ad esempio, per chiedere al calcolatore di eseguire la somma di due numeri scritti in due celle della memoria, quei programmatori dovevano scrivere esattamente, bit dopo bit, le tre istruzioni che abbiamo visto nel capitolo 9. Quindi per un compito così semplice, i programmatori di allora dovevano scrivere 3x16 cifre binarie, corrispondenti a 3 istruzioni ciascuna costituite da 16 bit (4 bit per il codice operativo e 12 bit per l'indirizzo dell'operando)

Il linguaggio, fatto soltanto di zeri e uni, che il calcolatore è in grado di interpretare e tradurre in comandi elementari (come le tre istruzioni che abbiamo visto nel capitolo 9) è chiamato "linguaggio macchina".

Per rendere più semplice e rapido il lavoro dei programmatori, si pensò di introdurre linguaggi di programmazione artificiale che consentissero di scrivere rapidamente ordini sintetici, ciascuno dei quali potesse corrispondere anche decine o centinaia di istruzioni del linguaggio macchina.

Ad esempio, nel linguaggio Python si può scrivere:

```
print math.sqrt ((79 + 123) * (1250 - 570))
```

Poiché la macchina comprende soltanto il linguaggio macchina, composto da uni e zeri, si scrissero programmi speciali per tradurre nel linguaggio macchina i programmi scritti in nel linguaggio artificiale. Questi programmi speciali erano scritti ovviamente in linguaggio macchina.



Fig. 10-1

Il programmatore prima di eseguire il suo programma doveva fare molte cose.



- 1) Trascrivere il programma scritto nel linguaggio artificiale prescelto su un supporto che la macchina sapesse leggere, come ad esempio le schede perforate, che ormai non si usano più da molto tempo. Ogni scheda conteneva un'istruzione, ossia un ordine elementare, che era scritto in un codice sotto forma di buchi messi in posizioni diverse sulla scheda. Questo programma era chiamato "sorgente", perchè era la sorgente dell'informazione nel successivo processo di traduzione.

In Fig. 10-1 abbiamo rappresentato lo schema del processo di traduzione. Notate che i rettangoli con l'angolo superiore sinistro tagliato rappresentano le schede perforate, perché queste avevano proprio quella forma.

In Fig. 10-2 sono rappresentate alcune schede perforate e loro frammenti.

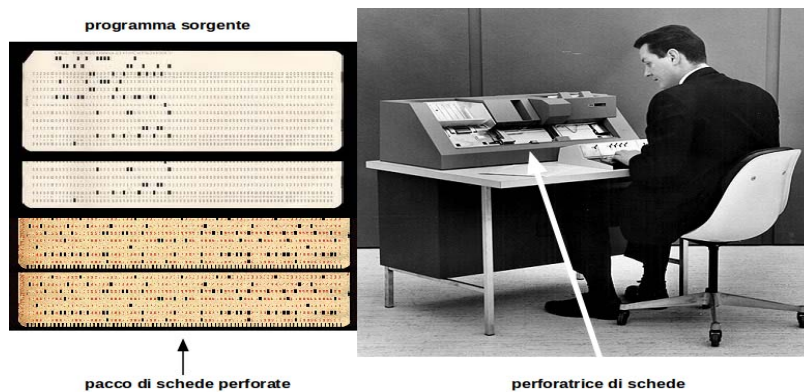


Fig. 10-2

- 2) Prelevare da un'armadio dell'ufficio il programma traduttore, che era stato scritto in linguaggio macchina e che era anch'esso trascritto su schede. Il pacco di schede veniva introdotto nel calcolatore che leggeva le schede e trasferiva il loro contenuto nella memoria centrale (Fig. 10-3).

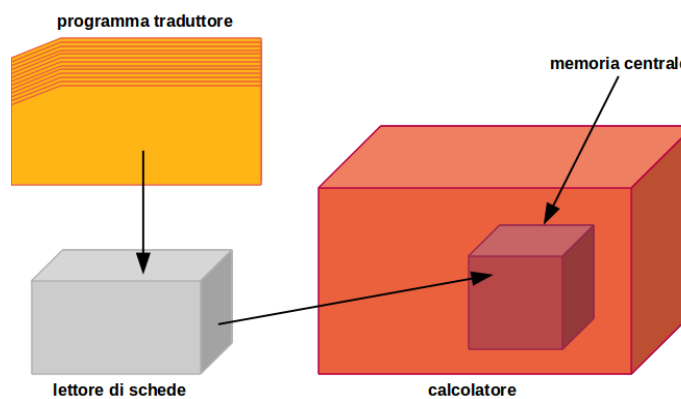


Fig. 10-3



3) Mettere nel lettore di schede il programma "sorgente" e ordinare al calcolatore di eseguire il programma traduttore. Il programma traduttore leggeva il programma "sorgente" e lo traduceva nel "programma oggetto", ossia produceva con l'aiuto di un'opportuna perforatrice, un terzo pacco di schede contenente un programma scritto in linguaggio macchina e quindi interpretabile ed eseguibile dal calcolatore (Fig. 10-4).

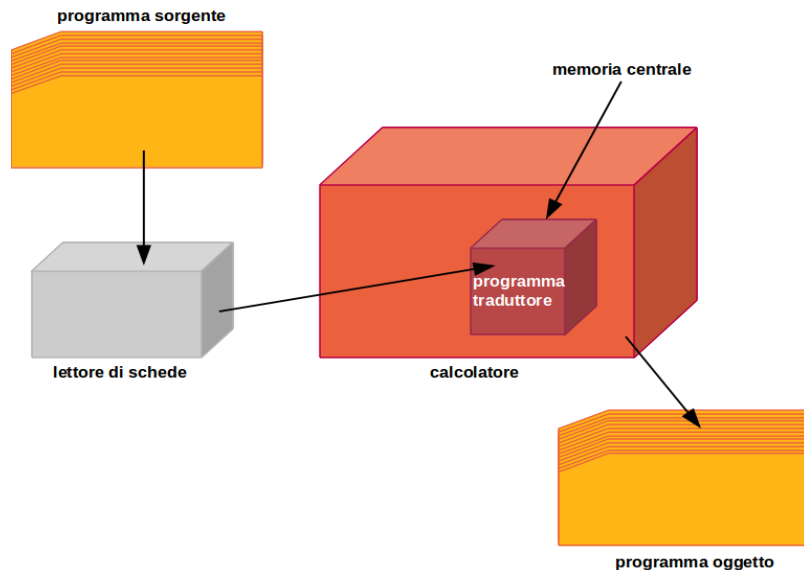


Fig. 10-4

4) A questo punto il programmatore introduceva nel lettore un pacco di schede contenente il programma oggetto e ordinava al calcolatore di caricarlo in memoria.

5) Infine, il programmatore ordinava al calcolatore l'esecuzione del programma "oggetto" che, essendo scritto in linguaggio macchina, poteva essere eseguito.

Oggi tutte queste 5 operazioni vengono eseguite automaticamente, in modo quasi invisibile per noi. Il programmatore digita sulla tastiera il suo programma nel linguaggio che preferisce (ad esempio python) che viene automaticamente trasferito nella memoria del calcolatore. Quando il programmatore impartisce l'ordine di esecuzione del programma (start o run), il calcolatore carica il programma traduttore, traduce il programma "sorgente" e infine esegue il programma "oggetto".

Ovviamente, poiché le schede perforate non si usano più, i vari programmi operano automaticamente soltanto su "file" contenuti nella memoria del calcolatore.

Ad esempio, nel linguaggio Python si può scrivere:

```
print math.sqrt ((79 + 123) * (1250 - 570))
```



che ordina il calcolo della radice quadrata di un'espressione aritmetica e che corrisponde a molte istruzioni di linguaggio macchina.

I linguaggi naturali, come il mio, sono molto più complicati di quelli artificiali come Python.



Esercizi

1. Che cos'è il "linguaggio macchina"?
2. Cos'è il "programma sorgente"?
3. Perché un tempo era necessario trascrivere un programma sulle schede?
4. Com'è cambiato oggi il lavoro dei programmatori, rispetto al passato, nello scrivere un programma e impartire ordini al calcolatore?
5. Secondo voi a un'istruzione aritmetica del tipo

`a = math.sqrt(25.3 * b + 39.5)`

quante istruzioni in linguaggio macchina corrispondono?



1 - LA TRASMISSIONE DEI DATI

Un calcolatore può trasmettere dati, ossia insiemi di bit, a un altro calcolatore oppure a molti altri calcolatori, usando canali di comunicazione molto diversi. Riassumiamo le caratteristiche dei canali più importanti.

In prima approssimazione i canali di comunicazione possono essere suddivisi in tre grandi gruppi.

Al primo gruppo appartengono i conduttori metallici. La trasmissione dei dati è effettuata con lo spostamento di microscopiche particelle elementari chiamate "elettroni".

Al secondo gruppo appartengono le fibre ottiche, nelle quali l'informazione è trasferita con la luce. Tutto avviene come quando si trasmettono segnali accendendo e spegnendo una lampadina o un faro, come nelle comunicazioni marine, ma nelle fibre ottiche i segnali luminosi sono piccolissimi e rapidissimi (migliaia di miliardi di accensioni e spegnimenti ogni secondo).

Al terzo gruppo appartengono i canali di comunicazione via segnali radio, così come avviene nella radio, nella televisione e nei telefoni cellulari. In lingua inglese tali canali sono chiamati "wireless" che significa "senza fili".

CONDUTTORI METALLICI

- Il doppino di rame

È il classico cavetto usato ancora oggi nelle nostre case per collegare i telefoni (Fig. 1-1).



Fig. 1-1 Il doppino di rame

È costituito da due fili di rame attorcigliati e isolati fra loro da due opportune guaine protettive.



I fili sono due perché, per funzionare, ogni circuito elettrico deve chiudersi su se stesso e quindi un filo serve per portare la corrente che tornerà indietro sull'altro filo. Si possono usare tecniche molto diverse per trasmettere i bit; per esempio, in Fig. 1-2, si è mostrato come un segnale elettrico che vada su e giù a intervalli regolari di tempo possa trasmettere un carattere o "byte" fatto dagli otto bit seguenti: 11100101. Un opportuno bit di "start" può essere utilizzato dal trasmettitore per dire al ricevitore che sta per trasmettere un carattere, mentre un bit di "stop" viene utilizzato per chiudere la trasmissione di quel carattere.

Utilizzando un doppino di rame, oppure un fascio di doppini, si possono trasmettere da milioni a miliardi di bit ogni secondo.

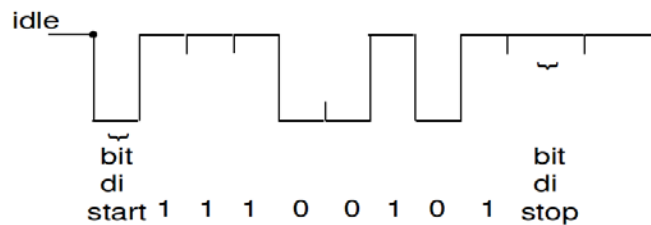


Fig. 1-2 Un modo per trasmettere un carattere

I doppini, così come, in generale, i conduttori metallici, costano poco e sono facili da installarsi, in quanto possono essere piegati senza rompersi.

- Il cavo coassiale

E' il ben noto cavetto generalmente utilizzato per trasmettere il segnale televisivo dall'antenna al televisore (Fig. 1-3). E' costituito da un filo centrale e da una o più guaine protettive per evitare che nel cavo entrino segnali di disturbo provenienti da altri fili o dall'esterno.

Su un cavo coassiale si possono trasmettere centinaia di milioni di bit al secondo.

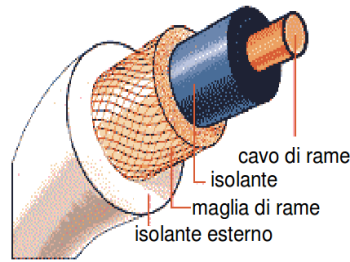


Fig. 1-3 Il cavo coassiale

LA FIBRA OTTICA

E' un sottilissimo filo di vetro che trasmette i bit come rapidissimi e piccolissimi lampi di luce. Il diametro del filo è dell'ordine di poche decine di micrometri o "micron". Perché il micrometro è il milionesimo di metro, il diametro di una fibra è dell'ordine di poche decine di millesimi di millimetro, ossia cento volte più sottile di un capello. La velocità di trasmissione è molto alta, dell'ordine di molti miliardi di bit al secondo, che possono essere trasportati a distanze di centinaia di chilometri.

Di solito le fibre ottiche sono impaccate in fasci contenenti molte decine di fibre come mostrato in Fig. 1-4.

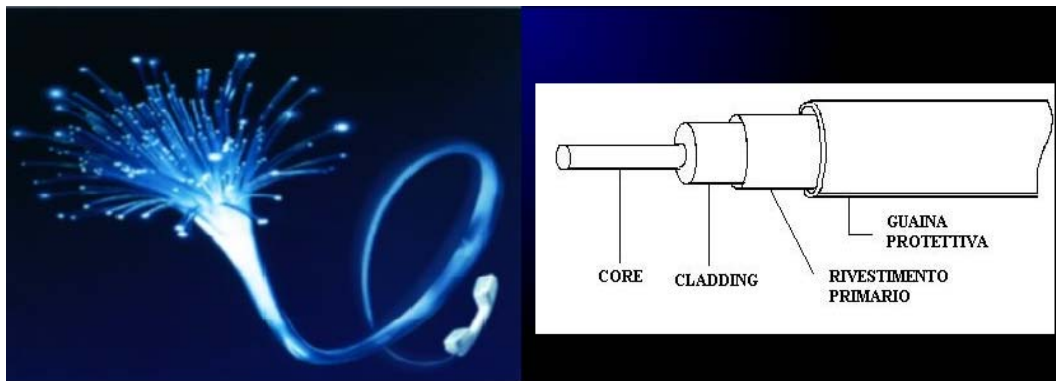


Fig. 1-4 Un fascio di fibre ottiche e lo schema della fibra

Un difetto della fibra è rappresentato dal fatto che è molto fragile e non può essere piegata. In compenso, le fibre più veloci possono raggiungere velocità dell'ordine di 10 mila miliardi di bit al secondo. La velocità media delle fibre ottiche, secondo calcoli recenti, è destinata a raddoppiare ogni anno.



TRASMISSIONE VIA RADIO (WI RELESS)

I bit possono anche essere trasmessi nell'aria, come ad esempio, da un edificio ad un altro, oppure nello spazio vuoto, come nelle comunicazioni tra un'astronave e una stazione di controllo a terra.

Sono disponibili tecnologie diverse per la trasmissione nell'etere, caratterizzate dall'impiego di frequenze diverse, ove la frequenza del segnale indica la rapidità con cui varia la sua intensità. Un esempio importante è il cosiddetto Wi-Fi, che è molto spesso impiegato nelle comunicazioni fra i calcolatori allocati in una stessa stanza o in uno stesso edificio.

La telefonia cellulare e la televisione digitale satellitare o terrestre sono un esempio di trasmissione di bit a distanza via etere.

Come già accennato, la trasmissione via etere non richiede di posare un cavo o una fibra fra i due calcolatori che si collegano. Inoltre consente velocità trasmissive molto alte, dell'ordine di decine di milioni di bit al secondo. Ha tuttavia un grave difetto: il segnale non riesce ad attraversare le pareti molto spesse degli edifici e risente molto dell'interferenza di altre trasmissioni o di disturbi esterni.

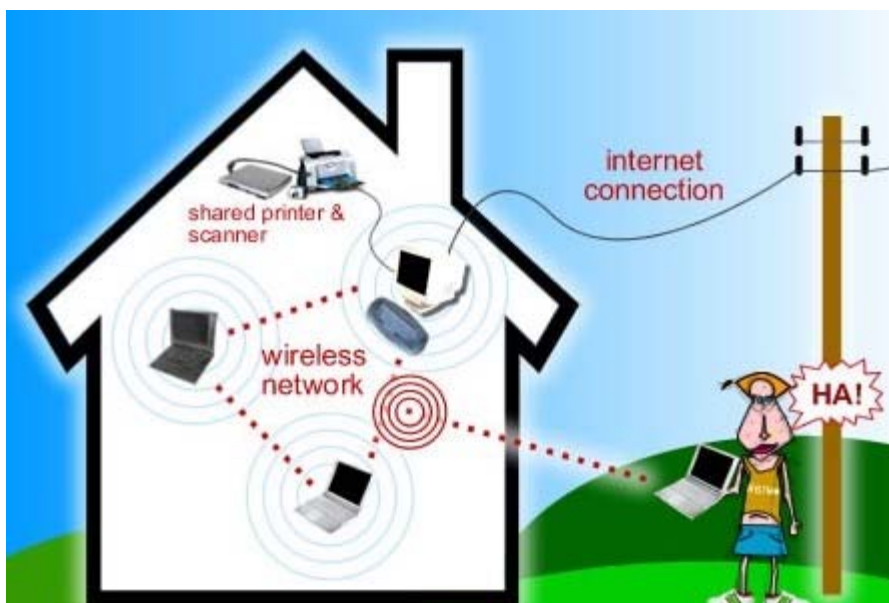


Fig. 1-5 La trasmissione di bit via etere



2 - LE RETI DI CALCOLATORI

Una rete di calcolatori è un insieme di calcolatori interconnessi fra loro da canali di comunicazione per mezzo dei quali un qualunque calcolatore può trasmettere dati a un qualunque altro calcolatore.

VARIE TOPOLOGIE

La topologia non è la scienza dei topi di cui è grandissima esperta Nana, la gattina di uno degli autori di questo libro. Con la parola "topologia" si intende la modalità con cui diversi "luoghi" (traduzione della parola greca "topos") sono interconnessi fra loro. Riportiamo alcuni esempi.

La Figura 2-1 rappresenta una **rete magliata**, caratterizzata dal fatto che ogni calcolatore è collegato a qualche altro con una connessione diretta, chiamata anche "**punto-a-punto**". Si noti che lo schema di Fig. 2-1, come gli altri schemi che seguiranno, rappresenta un'approssimazione che non mette in evidenza un fatto importante. Infatti, le reti funzionano in virtù dell'impiego di apparati elettronici che si occupano della trasmissione e ricezione dei dati. Qualche volta questi apparati sono immersi nella scatola che contiene il calcolatore, come avviene nella sezione a destra di Fig. 2-1, 2-2 e 2-3; più spesso invece gli apparati di comunicazioni sono esterni.

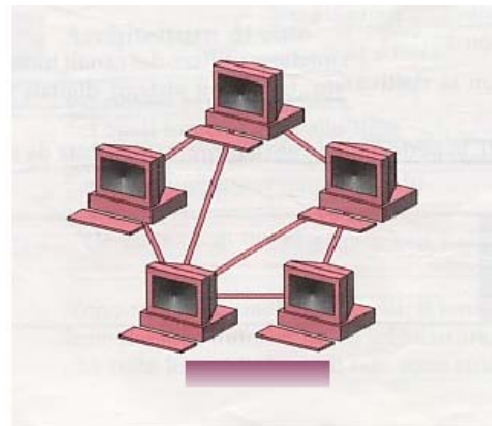
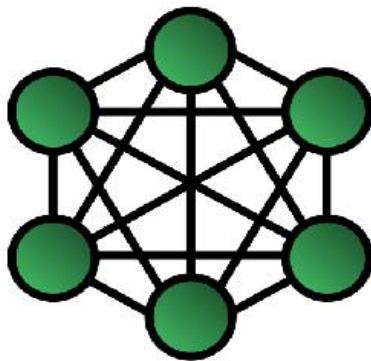


Fig. 2-1 Esempio di rete magliata

In Fig. 2-2 vediamo invece una "**rete a stella**", caratterizzata da un calcolatore centrale, di norma dotato di grandi capacità di calcolo e memoria o un apparato di comunicazioni, al quale sono interconnessi tutti gli altri. Quando uno dei calcolatori periferici deve trasmettere dati ad un altro dei calcolatori



periferici, trasmette quei dati al calcolatore centrale che si occupa di trasferirli al calcolatore periferico al quale quei dati sono stati destinati.

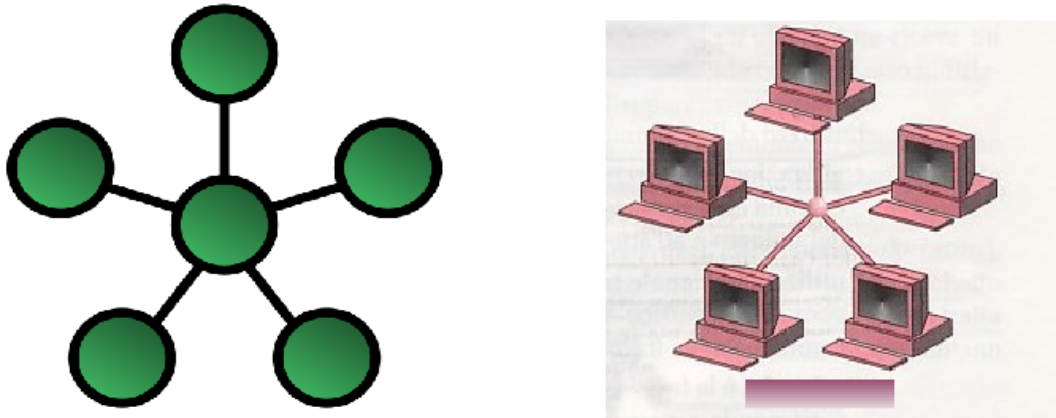


Fig. 2-2 Esempio di rete a stella

Nelle cosiddette "reti a bus" (Fig. 2-3) tutti i calcolatori sono interconnessi ad un grande, unico canale di comunicazione, chiamato appunto "bus". Come per gli altri tipi di reti, ogni calcolatore è contraddistinto da un indirizzo diverso da quello di ogni altro. Quando il calcolatore **A** vuole inviare un pacchetto di dati al calcolatore **B**, trasmette quei dati antepo-
nendo agli stessi l'indirizzo di **B**. Tutti i calcolatori hanno accesso ai dati trasmessi, ma soltanto il calcolatore **B**, riconoscendo dall'indirizzo che quei dati sono proprio diretti a lui, li legge e li utilizza.

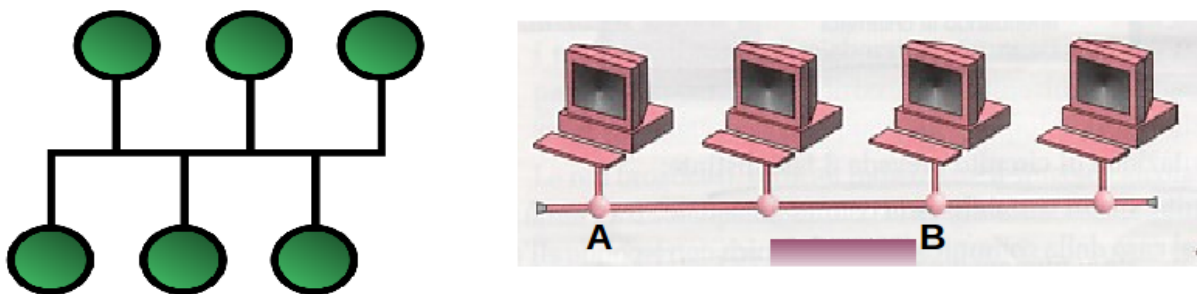


Fig. 2-3 Esempio di rete a bus

Un meccanismo analogo viene adottato nelle cosiddette "reti ad anello". Anche in questo caso, ogni calcolatore è contraddistinto



da un indirizzo diverso da quello di tutti gli altri. Quando, ad esempio, il calcolatore **A** deve trasmettere qualche dato al calcolatore **E**, premette a quei dati l'indirizzo di **E** e trasmette "indirizzo più dati" al suo vicino **B**. Il calcolatore **B** vede che i dati non sono diretti a lui e quindi li trasferisce al suo vicino **C**, che, a sua volta, li trasmette a **D**, sino a quando i dati non arrivano al calcolatore di destinazione **E** (Fig. 2-4).

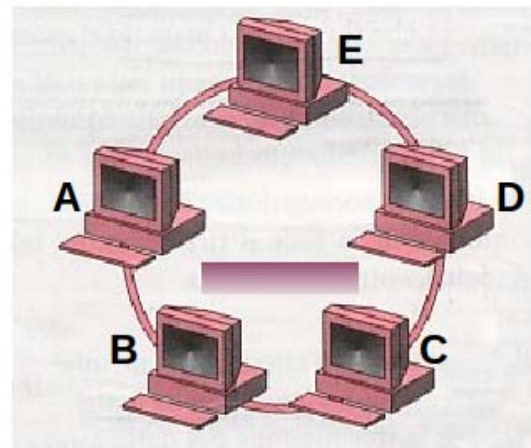
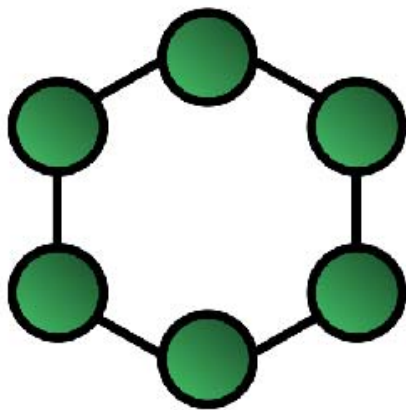


Fig. 2-4 Rete ad anello

Si noti che per semplicità di realizzazione i canali di interconnessione sono unidirezionali, ossia, ad esempio, **A** può trasmettere dati a **B** ma non può trasmettere dati a **E** direttamente.



RETI LOCALI E RETI GEOGRAFICHE

Si distinguono generalmente i seguenti tipi di rete.

LAN (Local Area Network)

- Una rete locale interconnette tutti i calcolatori di un'aula, un laboratorio o un appartamento.

WAN (Wide Area network)

- Una rete geografica consente invece l'interconnessione di calcolatori disposti su un vasto territorio geografico e, al limite, sull'intero pianeta ove abitiamo.

Le reti locali possono essere magliate, oppure a stella, oppure a bus, oppure ad anello. Negli ultimi anni si sono andate diffondendo anche le soluzioni radio o "wireless", basate generalmente sulla cosiddetta tecnologia "Wi-Fi.". Molto probabilmente nei prossimi anni si diffonderanno anche le reti locali basate su fibra ottica.

Le reti geografiche sono generalmente eterogenee, nel senso che non sono caratterizzate da un'unica topologia e da un'unica tecnologia. Molto spesso le reti geografiche contengono al loro interno una pluralità di reti locali che si interconnettono fra loro o con altri calcolatori attraverso opportuni apparati chiamati generalmente "router", ossia "instradatori" del traffico. Ad esempio, in Fig. 2-5 vediamo l'interconnessione di una rete locale a bus ad un'altra rete a bus attraverso una rete geografica.

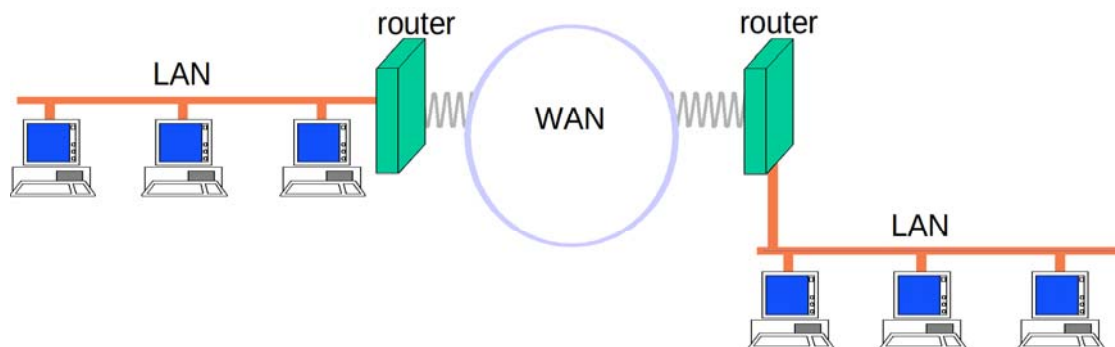


Fig. 2-5 La connessione di due reti locali a bus

Comunque, la topologia di una rete geografica con migliaia di utenti è simile a un albero, ove le foglie sono i calcolatori interconnessi, i rametti a cui sono attaccate le foglie sono le reti locali, mentre i rami più grossi e il tronco sono le cosiddette "dorsali", ossia i canali di comunicazione più importanti, quelli che portano i grossi volumi di dati da un'area geografica ad un'altra. Ora le dorsali sono costituite



prevalentemente da connessioni in fibra ottica, perché soltanto le fibre ottiche sono in grado di trasmettere le migliaia di miliardi di bit al secondo che devono essere trasferiti sulla dorsale di una rete geografica importante. Oggi tutti i mari sono solcati da migliaia di fibre ottiche che una flotta di navi oceanografiche continua a posare sui fondali.

LA RETE TELEFONICA PER LA TRASMISSIONE DEI DATI

La rete telefonica mondiale, nata molti anni prima delle reti di calcolatori, può essere utilizzata oggi per trasmettere dati da un calcolatore ad un altro.

Per collegare due calcolatori attraverso la rete telefonica occorre inserire fra ciascuno dei due calcolatori e il cavo telefonico un opportuno strumento chiamato "modem" ossia "modulatore e demodulatore" (Fig. 2-6).

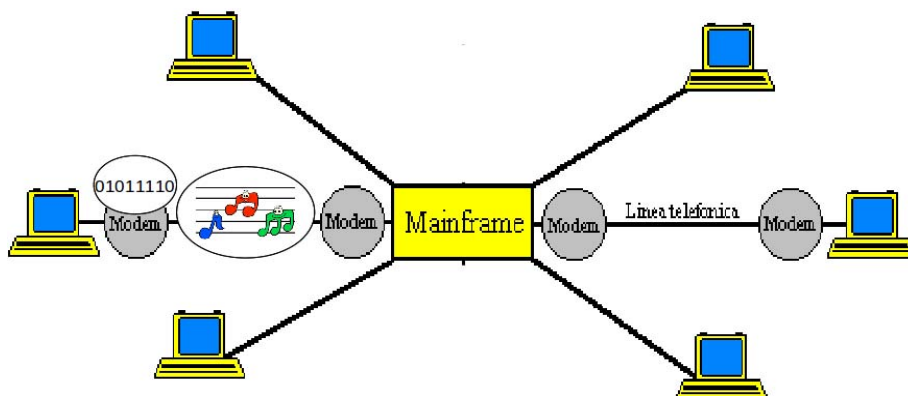


Fig. 2-6 La trasmissione dati attraverso la rete telefonica

Il Modem ha un ruolo molto importante. Infatti la rete telefonica è stata progettata per trasmettere la voce ed è quindi in grado di trasmettere i suoni, purché questi suoni non siano troppo acuti. Quindi il Modem è uno strumento metà informatico e metà musicale, nel senso che nella fase di trasmissione ha il compito di convertire i bit in uscita dal calcolatore in brevi suoni da trasmettere sulla rete telefonica, mentre nella fase di ricezione converte le note musicali ricevute nei bit corrispondenti.

L'uso della rete telefonica per trasmettere dati ha un grave difetto: il numero di bit al secondo, ossia il volume di informazione che viene trasmesso nell'unità di tempo, è, nei casi migliori, dell'ordine delle decine di migliaia di bit al secondo, corrispondenti a poche decine di migliaia di caratteri al secondo. Come vedremo in seguito, tali velocità di trasmissione



dati sono insufficienti per applicazioni importanti come la trasmissione di video.

Questo grave difetto è compensato da un grande pregio: la rete telefonica è diffusa su tutta la terra. Rappresenta l'equivalente di un investimento globale dell'umanità dell'ordine di mille miliardi di euro, ed è proprio la sua pervasività che ha determinato il suo successo anche per la trasmissione dei dati.

Un esempio importante di trasmissione dati basata su rete telefonica e modem è il fax, uno degli strumenti di comunicazione più diffusi nel mondo industrializzato. Il verbo "faxare", cioè trasmettere in fax, verbo regolare della prima coniugazione, è entrato nei dizionari della lingua italiana.

Reti pubbliche di trasmissione dati a commutazione di circuito e di pacchetto

Molte aziende e pubbliche amministrazioni non possono permettersi il lusso di comprarsi una rete privata, dedicata soltanto all'interconnessione dei loro calcolatori. Per questa ragione, una trentina di anni fa, iniziò la realizzazione delle prime reti pubbliche per la trasmissione dei dati. Due punti di vista si scontrarono fin dal primo momento: la commutazione di circuito e la commutazione di pacchetto.

Le reti di trasmissione dati a commutazione di circuito (Fig. 2-7) realizzano una connessione fisica diretta fra l'utente chiamante e l'utente chiamato, così come avviene nella antica rete telefonica, ma con una migliore qualità di trasmissione. L'inconveniente della commutazione di circuito è rappresentato dal fatto che le risorse impegnate nella connessione sono spesso sottoutilizzate: ad esempio, mentre un utente pensa e non trasmette, la linea di collegamento risulta occupata ma non utilizzata né dai due interlocutori né da altri.

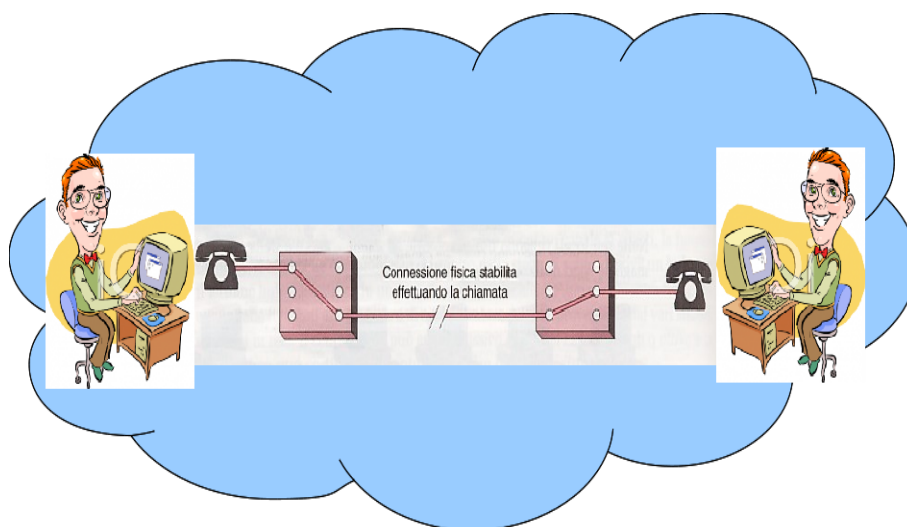


Fig. 2-7 Rete a commutazione di circuito



Alla commutazione di circuito si contrappone la commutazione di pacchetto (Fig. 2-8). Una rete a commutazione di pacchetto è caratterizzata, anziché da centrali di commutazione, come quelle della rete telefonica, da calcolatori collegati fra loro da linee del tipo delle linee "punto a punto". In fase di trasmissione i dati, che sono tutti bit, sono divisi in tanti pacchetti. Ad esempio, una frase di un utente composta da centomila bit potrebbe essere suddivisa in dieci pacchetti da diecimila bit ciascuno. I pacchetti saranno poi inviati al calcolatore più vicino e da questo al secondo e così via, sino ad arrivare a destinazione. Così uno stesso collegamento viene utilizzato per una pluralità di trasmissioni. Al fine di migliorare l'efficienza globale della rete, i pacchetti potranno seguire percorsi diversi. Lo scopo principale di questa seconda soluzione, intrinsecamente più complicata, è la piena utilizzazione delle risorse trasmissive. Il suo inconveniente principale è rappresentato dalla variabilità dei tempi di consegna dei singoli pacchetti. Così vi è il pericolo che un segmento di voce arrivi in ritardo rispetto al segmento precedente, compromettendo la capacità di comprendere bene il messaggio vocale. In sostanza, la soluzione appare molto efficiente per la trasmissione di dati numerici, ma molto complicata per la trasmissione di voce o di video.

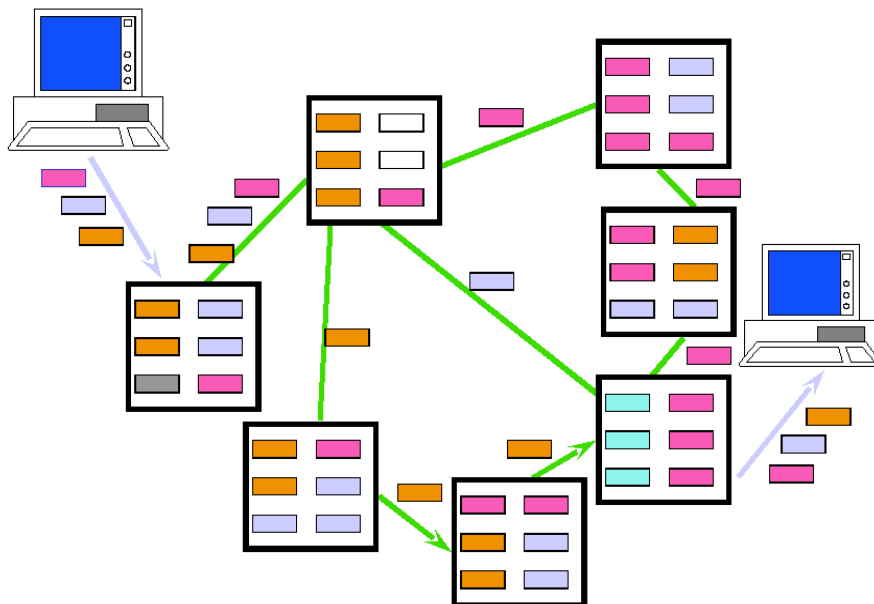


Fig. 2-8 Rete a commutazione di pacchetto



3 - INTERNET

La rete delle reti

La storia di Internet inizia nel 1960 ed è una storia di continue scoperte e invenzioni. All'incirca a metà di questa storia, i progettisti di Internet si diedero un obiettivo ambizioso: diventare la grande rete delle reti. Infatti, in quel momento operavano molte reti diverse, locali e geografiche, caratterizzate da topologie diverse, progettate e prodotte da aziende diverse.

I sistemi costituiti dai vari calcolatori e dalle reti di comunicazione non potevano scambiarsi dati perché ogni rete adottava regole diverse per organizzare i dati da trasmettere, per collegare le sorgenti e le destinazioni dei flussi di dati, per specificare gli indirizzi dei calcolatori coinvolti nella comunicazione e così via. Inoltre ciascuno dei produttori delle diverse reti, molto geloso delle sue soluzioni tecniche, non gradiva che calcolatori di altri produttori venissero collegati alla loro rete.

Si decise allora di operare secondo lo schema rappresentato in Fig. 3-1.

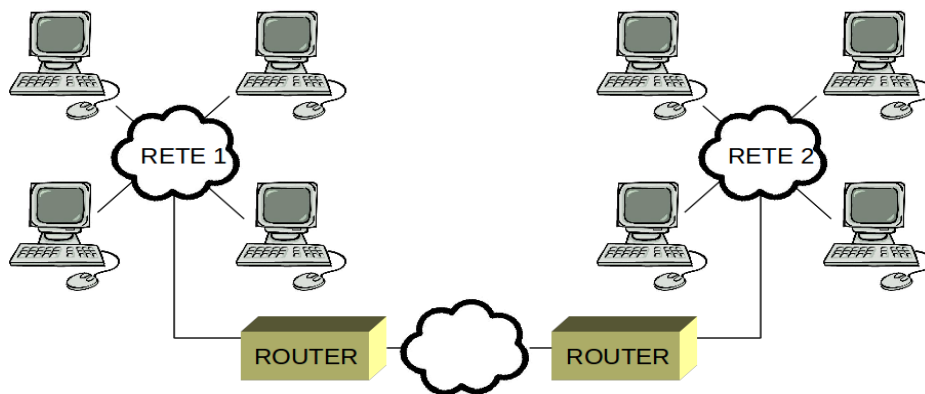


Fig. 3-1

Alla rete di sinistra chiamata RETE 1 in figura, una delle reti preesistenti ad Internet, si aggiunse un apparato, chiamato ROUTER in Fig. 3-1, che colloquiava con altri calcolatori della stessa RETE 1, secondo le regole adottate dai progettisti di quella rete, ma che fosse collegato a un canale di comunicazione "da Rete a Rete". Alla rete a destra chiamata "RETE 2" nella figura, si aggiunse un altro apparato, capace di scambiare i dati con gli altri calcolatori della stessa RETE 2 secondo le regole prestabilite per questa. Si collegarono poi fra loro i due router con un canale opportuno. Si ricorda che "route" significa "strada" e "router" significa "instradatore".

I router non sono concettualmente o strutturalmente diversi dagli altri calcolatori; sono soltanto calcolatori specializzati nella trasmissione dei dati. Quando un calcolatore della RETE 1 vuol trasmettere un'insieme di dati a un calcolatore della RETE 2, trasmette quei dati al router della RETE 1 secondo le modalità della stessa RETE 1. Il router della RETE 1 trasmette allora quei



dati al router della RETE 2 e questi convoglia i dati ricevuti al calcolatore di destinazione utilizzando la modalità di comunicazione della RETE 2. In questo modo qualunque calcolatore della RETE 1 può trasmettere dati a qualunque calcolatore della RETE 2.

Possiamo estendere lo schema. Ad esempio, la RETE 2 potrebbe utilizzare un secondo router per collegarsi ad una terza rete chiamata "RETE 3". In virtù di questo collegamento, qualunque calcolatore della RETE 1 potrebbe trasmettere dati a qualunque calcolatore della RETE 3 passando attraverso la RETE 2. Questa RETE 2 mette le proprie risorse di comunicazione a disposizione di calcolatori che appartengono a reti diverse; in cambio, altre reti, in altre circostanze, si metteranno a disposizione di comunicazioni che interessano i calcolatore della RETE 2. E' la collaborazione tra router il principio fondamentale dei fornitori e degli utenti dei servizi di Internet.

Il lungo viaggio dei trenini dei messaggi

Nonostante il grande numero di calcolatori collegati, dell'ordine di diversi miliardi, nel mondo, la tecnologia di Internet è relativamente semplice. Consideriamo dapprima il caso della trasmissione di un messaggio breve. In vacanza avete conosciuto una bambina americana che vi ha lasciato il suo indirizzo di posta elettronica ed è sufficientemente simpatica da meritare di essere ricordata. Così, tornando a casa le mandate una "mail", ossia una lettera con un breve messaggio: "E' stato molto bello giocare con te al mare". La vostra amica non ha ancora un suo indirizzo di posta elettronica e quindi mandate il messaggio alla sua mamma che si chiama Mary Smith e lavora nello staff segretariale del Presidente degli Stati Uniti. Il suo indirizzo di posta elettronica sarà: "mary.smith@white-house.gov".

Come vedremo tra breve, l'indirizzo "white-house.gov" del calcolatore dove finirà il messaggio sarà convertito in una sequenza di bit, uno o zero, sequenza più idonea all'elaborazione automatica, ma la conversione dell'indirizzo scritto come sequenza di caratteri alfabetici a un numero binario non cambia la sostanza delle cose per cui supponiamo, per ora, che l'indirizzo non subisca alcuna modificazione.

Il nostro calcolatore raccoglie dalla tastiera il testo del messaggio e lo incolla all'indirizzo del destinatario, la mamma della vostra amica.

Oggi sono attivi nel mondo numerosi calcolatori, chiamati I.S.P. ("Internet Service Provider" ossia "Fornitori di Servizi di Internet"), che hanno il compito di trasmettere i pacchetti di dati ricevuti da uno dei loro "client" ("clienti") a qualunque altro calcolatore collegato a Internet.

Se il vostro calcolatore è collegato al calcolatore del centro che vi fornisce il servizio di collegamento a Internet (Internet Provider) via Modem e telefono (Fig. 3-2), il messaggio arriva attraverso la linea telefonica al primo router, ossia al calcolatore del vostro "I.S.P."

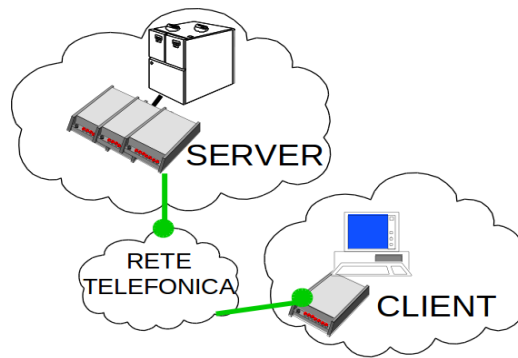


Fig. 3-2

Tutto avviene come in una grande rete ferroviaria (Fig. 3-3).

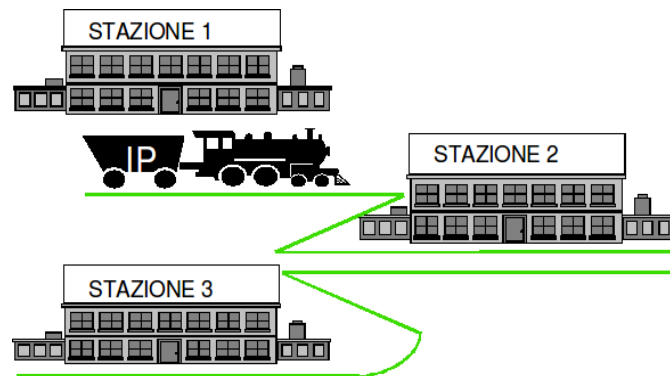


Fig. 3-3

Il capostazione della stazione di partenza, ossia il vostro calcolatore, invia un trenino con un unico vagoncino contenente un carico di bit, il vostro messaggio. Sulla locomotiva c'è un piccolo carico di servizio, l'indirizzo di destinazione. Quel trenino è chiamato "pacchetto IP", dalle iniziali di "Internet Protocol". Il trenino IP arriva alla sua prima stazione intermedia, ossia al calcolatore del vostro I.S.P. Il capostazione si affaccia alla locomotiva e legge l'indirizzo di destinazione. L'ultima parte dell'indirizzo (.gov) indica la rete degli enti governativi degli Stati Uniti, per cui il messaggio dovrà essere inoltrato in America. Il capostazione consulta una grande tabella, che gli indica, per ogni destinazione finale, la destinazione successiva a cui inviare il trenino. Così il trenino viene inviato a Milano, presso un istituto di ricerca dell'Università, da cui devono transitare tutti i trenini partiti da Torino con destinazione negli Stati Uniti. Il capostazione di Milano farà lo stesso lavoro, e inoltrerà il trenino a Bologna, e così, di stazione in stazione, sino all'arrivo alla stazione finale, il calcolatore dove, come vedremo, risiede la casella postale della mamma di Mary.

Un vagoncino IP ha una capacità limitata, mediamente pari a circa 1500 caratteri. Di conseguenza, il breve messaggio per Mary può essere contenuto in un unico vagoncino, ma il milione di bit



necessari per rappresentare un'immagine con un'ottima risoluzione non sta in un vagoncino e richiede un treno composto da moltissimi vagoncini. Sfortunatamente, la rete ferroviaria di Internet non ama convogliare treni con più di un vagoncino; per questa ragione il capostazione di partenza, ossia il vostro computer, dovendo trasmettere cento vagoncini, come sua prima operazione scompone il treno in cento trenini, con cento locomotive diverse, ciascuna delle quali trascina un diverso vagoncino. Il lungo treno di partenza è chiamato "pacchetto TCP", come "Transport Control Protocol", una sigla misteriosa che non ha un profondo significato (Fig. 3-4).

I cento trenini IP in cui il capostazione ha scomposto il lungo treno TCP sono autonomi e viaggiano ciascuno per proprio conto, secondo la logica della "Commutazione di pacchetto" che abbiamo descritto nel capitolo 2. Spetta al capostazione della destinazione finale, il computer che ospita la posta di Mary, raccogliere i singoli vagoncini e incollarli nell'ordine corretto, ricostruendo il lungo treno di partenza.

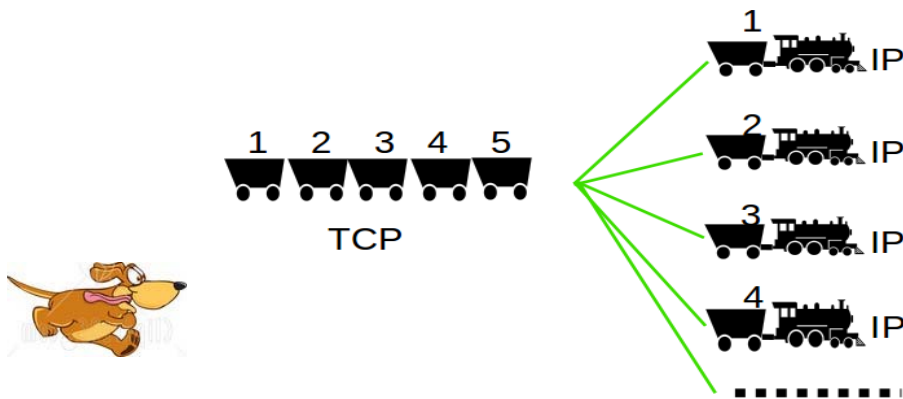


Fig. 3-4

Questo meccanismo costituisce il principale difetto di Internet, probabilmente l'unico serio. I vari trenini IP che portano il carico del lungo treno TCP partono in sequenza ma possono arrivare in ordine sparso. Inoltre se uno solo dei trenini elementari si perde, il capostazione di arrivo non è più in grado di ricostruire il treno. In questo caso, deve mandare un treno al capostazione di partenza per chiedere la ritrasmissione del treno perduto. L'arrivo in ordine sparso non pone alcun problema alla trasmissione di messaggi scritti, ma costituisce un ostacolo difficilissimo alla trasmissione di un messaggio vocale o di un filmato. Nel caso del messaggio vocale, ad esempio, tutti i frammenti che compongono una parola, come, ad esempio, la sillaba "ma" della parola "mamma", devono arrivare entro un decimo di secondo ed essere riprodotti nell'altoparlante del computer di destinazione nell'ordine

corretto, senza discontinuità, perché altrimenti il messaggio sarebbe incomprensibile. Per questo la trasmissione della voce e del video su Internet ha rappresentato e rappresenta tuttora uno dei temi più interessanti della ricerca tecnologica.



Abbiamo accennato al fatto che l'utente di Internet usa indirizzi simbolici e mnemonici, come `white-house.gov`, ma che i calcolatori della rete interpretano le destinazioni dei messaggi solo se queste sono scritte in codice binario, come sequenze di uni o zeri. Spetta a un calcolatore specializzato, chiamato "Domain Name Server" (DNS), il compito di tradurre gli indirizzi simbolici negli indirizzi fisici scritti in binario (Fig. 3-5)

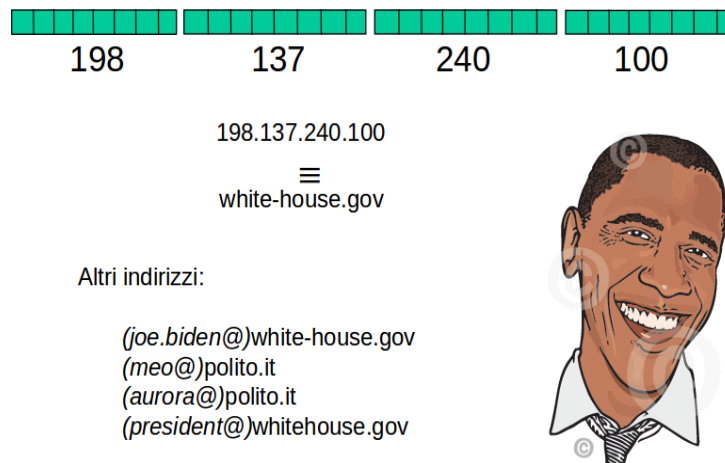


Fig. 3-5

Il lavoro del "Domain Name Server" è complesso ed è eseguito con la collaborazione di molti calcolatori distribuiti su tutta la rete.

Un indirizzo fisico è composto da quattro numeri binari, ciascuno dei quali è costituito da 8 bit, per un totale di 32 bit. Con 32 bit si possono distinguere oltre 4 miliardi di indirizzi diversi, che, nel momento in cui i comitati internazionali, incaricati di definire le regole di Internet, specificarono le modalità di indirizzamento, sembravano ampiamente sufficienti a coprire tutte le esigenze per molti anni. Da alcuni anni, invece, gli indirizzi cominciano a scarseggiare. I comitati internazionali hanno già definito un nuovo standard chiamato IPv6 (versione 6 del pacchetto IP), che consentirà la specificazione di molte centinaia di miliardi di indirizzi diversi. Questa cifra può sembrare pazzesca se pensiamo che gli utenti della rete siano gli uomini; nella realtà i soggetti interessati alle comunicazioni su Internet sono i calcolatori, che diventano ogni giorno più piccoli e numerosi. Domani, ogni forno, frigorifero, condizionatore, tostapane, videoregistratore e ogni altro elettrodomestico ospiterà uno o più processori caratterizzati ciascuno dal proprio indirizzo IPv6.



4-parte 2-I SERVIZI PIÙ IMPORTANTI DI INTERNET

Usando il meccanismo dei trenini IP e spesso anche quello della scomposizione e ricomposizione dei lunghi treni TCP, Internet riesce a mettere a disposizione degli utenti molti servizi utili. Soffermiamoci sui più importanti.

La posta elettronica

Nei primi tempi della storia di Internet la posta elettronica veniva attuata inviando direttamente un messaggio da un calcolatore ad un altro. Questa soluzione, molto semplice, aveva un difetto: se il calcolatore del destinatario era spento, il messaggio si perdeva. I gatti, che sono sempre a zozzo, furono i primi ad accorgersi di questo difetto.

Per questa ragione si installarono nella rete un certo numero di "mail server", che dovevano essere sempre accesi. Nella Fig. 4-1 sono rappresentati due mail server, chiamati "caniegatti.it" e "gattiproletari.it", ma nella realtà di oggi i mail server attivi sono molti milioni. In Fig. 4-1 sono rappresentati soltanto 4 dei molti "client" (clienti del servizio di posta) dei due mail server; due appartengono a due cani, "nazibau@caniegatti.it" e "nazibob@caniegatti.it", e due appartengono ai due gatti "miao@caniegatti.it" e "randagio mao@gattiproletari.it".

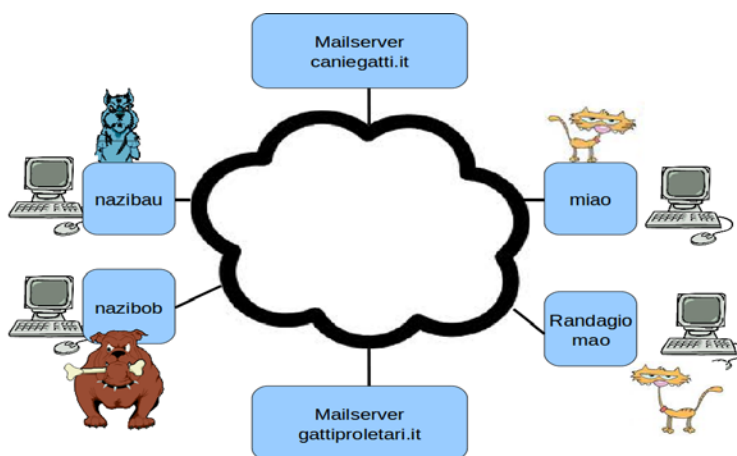


Fig. 4-1

I mail server ospitano le caselle postali dei loro clienti, che sono memorizzate nelle memorie di massa dei due server. In Fig 4-2 sono stati rappresentati i primi messaggi delle tre caselle postali appartenenti ai tre clienti nazibau@caniegatti.it, nazibob@caniegatti.it e randagio mao@gattiproletari.it.

Ogni casella postale è suddivisa in due campi: il primo contiene l'indirizzo del mittente, il secondo il contenuto del messaggio. Osservate che il mittente del messaggio contenuto nella mail box di randagio mao è un gatto che utilizza il mail server "gattiproletari.it" (diverso dal server di posta degli altri tre



protagonisti della nostra storia).



Fig. 4-2

Per comprendere il senso dei messaggi di Fig. 4-2, tenete presente che Nazibau è un cane militante nel movimento dei "Cani Arrabbiati", che predica il ritorno allo stato randagio, Nazibob è un cane del movimento dei "Nazi canischi", che vogliono la soppressione dei gatti e la deportazione dei cani bastardi, mentre Miao è un povero gattino che rischia di diventare vittima di Nazibau e Nazibob.

E' importante ricordare che quando inviate un messaggio ad un vostro amico, questo viene memorizzato nella sua mail box anche se il suo calcolatore è spento. Quando il vostro amico vorrà leggere il vostro messaggio, accenderà il suo calcolatore e si collegherà al suo mail server che trasmetterà il vostro messaggio al suo calcolatore.

FTP - Il trasferimento di file.

Il trasferimento di file, o "file transfer", può essere visto come una variante della posta elettronica. E' idoneo al trasferimento senza errori di documenti molto lunghi, come progetti, filmati, articoli scientifici o interi volumi. Costituisce ormai uno dei meccanismi standard per la diffusione della documentazione nell'ambito della comunità scientifica mondiale. (Fig. 4.3)

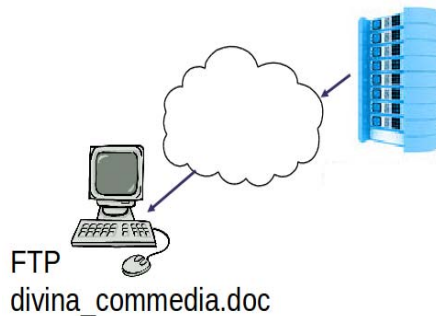


Fig. 4.3



TELNET - L'accesso a un calcolatore remoto.

Un opportuno servizio di Internet, chiamato Telnet, offre all'utente la possibilità di collegarsi a un calcolatore remoto, anche a molte migliaia di chilometri di distanza, e di ordinarne l'esecuzione di un programma (Fig. 4.4). Dal punto di vista dell'utente, tutto avviene come se egli fosse collegato alla consolle del calcolatore remoto. Nella realtà tale servizio è ottenuto attraverso lo scambio di una successione di messaggi sulla rete fra l'utente e il calcolatore remoto.

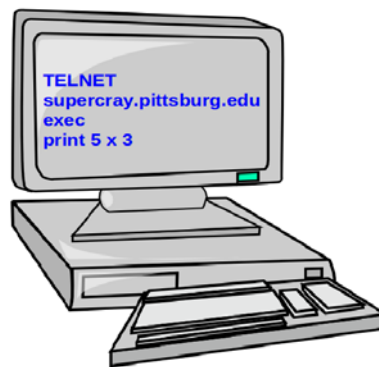


Fig. 4.4

Qualche volta, come nel caso illustrato in figura 4.4, che mostra il collegamento a un supercomputer per eseguire l'operazione 5 x 3, Telnet rende disponibili in tutto il mondo risorse di calcolo preziose. Qualche altra volta il collegamento rende possibile il movimento di una telecamera remota e l'acquisizione delle immagini catturate o più banalmente l'accesso al calcolatore che gestisce le vendite di un grande magazzino.

La navigazione

Centinaia di milioni di calcolatori in tutto il mondo mettono a disposizione di tutti enormi volumi di informazione. Molti programmi sono stati sviluppati per rendere più facile e proficua la navigazione nel grande mare di questi archivi o "data base server".

Oggi il più importante dei meccanismi per il "netsurfing", ossia per navigare nel grande mare della conoscenza contenuta in Internet, adotta uno standard che è stato proposto nel 1992 da alcuni ricercatori del C.E.R.N., il grande istituto europeo di ricerche che ha sede a Ginevra. A quella soluzione fa riferimento la nota sigla W.W.W. (World Wide Web o "la ragnatela estesa come il mondo").

Una pagina W.W.W. contiene generalmente un certo numero di parole sottolineate o immagini o altri segni che fanno esplicito riferimento ad altri concetti o ad altri dati. Ciascuno di quei riferimenti contiene l'indirizzo dell'archivio dove i nuovi dati potranno essere trovati ma l'utente non sarà stato costretto a



leggere e reimpostare il nuovo indirizzo, essendo sufficiente spostare il cursore su quella parola o su quella immagine e premere il pulsante del mouse, per ottenere l'accesso automatico al nuovo indirizzo.

Vediamo un semplice esempio. Supponiamo che la mamma vi abbia finalmente dato il permesso di adottare un cane. Un amico vi ha detto che trovate un'ampia scelta di cani da adottare su molte pagine W.W.W. Scegliete una pagina a caso – www.adottauncane.net – e vi collegate a quella pagina. Vi comparirà la pagina rappresentata in Fig. 4.5, contenente le immagini di molti cani. Fate “click” sul nome del cane che vi piace di più e automaticamente vi comparirà la descrizione delle caratteristiche di quel cane.

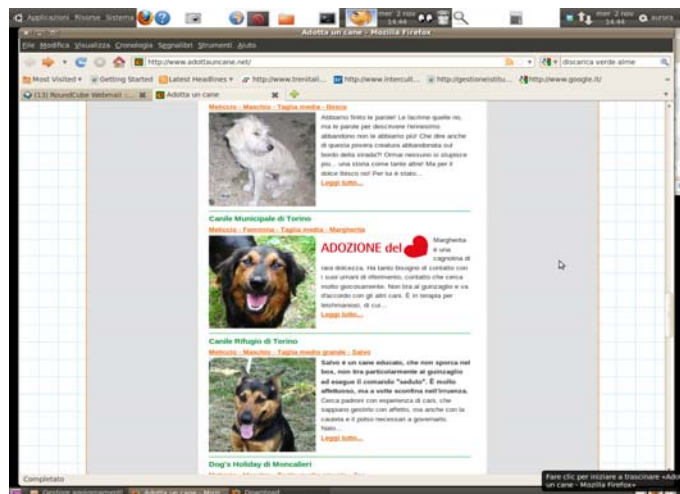


Fig. 4-5

Un ipertesto è un documento nel quale sono disseminate opportune ancore (**ri riferimenti ad altri documenti**), utilizzate come puntatori ad altre sezioni del documento o ad altri documenti. Il meccanismo del W.W.W. attua un grande ipertesto mondiale, ove ogni documento contiene una pluralità di riferimenti ad altri documenti allocati nello stesso sito o in altri siti della rete.

Il W.W.W. tratta, come suoi componenti fondamentali, non soltanto i testi e i numeri, ma anche le immagini, i suoni, i filmati. Inoltre il meccanismo è molto potente e flessibile, per cui l'interrogazione può essere facilmente integrata con altre funzionalità della rete, in modo da attuare un'ampia gamma di applicazioni, dall'assistenza tecnica alla consultazione di archivi e all'electronic banking, che è l'accesso remoto al conto corrente dei vostri genitori in banca.

Motori di ricerca

Talvolta non è noto a priori l'indirizzo del sito da dove iniziare la ricerca dei dati a cui si è interessati. Per risolvere questo problema sono stati attivati opportuni programmi che sfogliano tutti i documenti disponibili in molti milioni di data base



server.

Questi calcolatori e i programmi relativi sono chiamati motori di ricerca perché a tempo pieno interrogano tutti i data base server della rete, sintetizzando poi i risultati del proprio lavoro in enormi indici. Nel nostro caso, un indice è una tabella che contiene tutte le parole che compaiono nei miliardi di pagine di Internet; questa tabella precisa, per ogni parola, gli indirizzi di tutti i documenti che contengono quella parola al loro interno. Quando un utente desidera raccogliere informazioni su un determinato argomento, si collega a uno dei siti ove è allocato il grande indice ed esprime la propria richiesta utilizzando una o più parole chiave. Ad esempio, se fosse interessato ad acquistare un windsurf nuovo, egli si collegherebbe al motore di ricerca, esplicitando la parola "windsurf" nell'apposita finestrella destinata alle parole chiave. Il motore di ricerca proporrebbe allora una sequenza di siti e documenti contenenti informazioni sui windsurf. Come secondo esempio, provate a collegarvi a Google (www.google.com) che è forse il più importante dei motori di ricerca oggi disponibili. Vi comparirà una finestra nella quale scriverete la parola chiave per la ricerca – cane Moon di Angelo Raffaele Meo -. Il motore di ricerca vi darà i puntatori ad alcuni documenti, fra cui un manualino di programmazione in Python sul quale compaiono alcune immagini del cane Moon.

Newsgroup, chat e phone.

Le newsgroup sono grandi "tazebao" elettronici ove i membri di una comunità di utenti che condividono l'interesse per un determinato problema, o gruppo di interesse, espongono liberamente le proprie opinioni. Nel corso delle ultime elezioni politiche molti italiani residenti all'estero hanno condiviso sulle News con i residenti in Italia la passione per il dibattito politico.

Le chat sono conversazioni in tempo reale fra gli utenti della rete. Un calcolatore ospita la conversazione; a quel calcolatore e dallo stesso sono trasmessi i messaggi di tutti gli iscritti alla conversazione. Eventualmente il dibattito può essere disciplinato da un utente "chairman" o moderatore, che controlla la validità dei messaggi ricevuti prima di irradiarli agli altri partecipanti alla discussione. All'inizio le chat erano prevalentemente testuali, per cui i messaggi erano costituiti esclusivamente da sequenze di caratteri, attualmente sono multimediali, con messaggi audio e video contenenti, fra l'altro, i volti dei partecipanti al dibattito (Fig. 4.6).



Fig. 4.6



Il talk o messenger è la particolareggiata chat a due soli utenti, con l'obiettivo di rendere il colloquio particolarmente rapido. Come per le chat, nei primi tempi la comunicazione era soltanto testuale. Successivamente, sono stati sviluppati prodotti per scambiarsi messaggi vocali. Infine, la telefonia vocale è diventata videotelefonata, per cui i due utenti non soltanto colloquiano fra loro ma anche si vedono (Fig. 4.7). Un noto programma per attuare la telefonia vocale e la videotelefonata è Skype.



Fig. 4.7

I siti delle reti sociali (o "social networks")

Una rete sociale è un gruppo di persone che hanno interessi comuni. Un esempio di rete sociale potrebbe essere il gruppo dei tifosi della Juventus.

Negli ultimi anni sono state sviluppate numerose soluzioni tecnologiche basate su Internet per consentire lo scambio di messaggi fra i componenti di una rete sociale (i cosiddetti "Social network sites").

L'utente di un sito di rete sociale innanzitutto descrive il proprio profilo, ossia i propri interessi, e la lista dei cosiddetti "contatti", ossia degli amici con cui intende colloquiare. Eventualmente egli potrà anche visitare la lista degli amici dei suoi amici.

E' importante sapere che lo scambio di informazioni personali sui siti delle reti sociali può essere pericoloso, in quanto facilmente tutte ciò che viene scritto potrebbe diventare di pubblico dominio.

Vediamo alcuni esempi di reti sociali.

Facebook.

Facebook è un archivio contenente le fotografie e le informazioni relative a tutti i membri di una comunità, come, ad esempio, tutti gli studenti di un'università.

Nel momento in cui scriviamo gli utenti di Facebook sparsi in tutto il mondo sono più di 500 milioni, mentre gli utenti italiani



sono 18 milioni.

Gli amici possono scambiarsi messaggi, fotografie, video ed altre informazioni, sia in forma privata sia volendo, in forma pubblica, ossia autorizzando la pubblicazione in una bacheca pubblica dei contenuti dei messaggi.

My Space ("il mio spazio")

E' un sito sociale orientato prevalentemente agli appassionati di musica e spettacolo. Consente in particolare lo scambio di spettacoli musicali o cinematografici, ossia su My Space è possibile ascoltare musica o vedere video.

Twitter (dall'inglese "cinguettare")

E' caratterizzato dalla possibilità per un utente di entrare in contatto con altri utenti attraverso opportuni cinguettii (o "twitt"), ossia messaggi istantanei che potranno essere pubblicati nella pagine di un altro utente.

LinkedIn

E' un social network di tipo "professionale", nel senso che partecipano ad esso professionisti, come programmatori o progettisti, che vogliono entrare in contatto con altri professionisti.

Flickr

E' una comunità di appassionati di fotografie che intendono condividere i frutti del loro lavoro.

You Tube

E' la ben nota comunità di appassionati di video che intendono condividere filmati di ogni genere, dal gattino, che giocando con i pesci, cade nella loro vasca alla importante lezione univertaria oppure al comizi o rivoluzionario.





5 – Internet del futuro

I PROGRESSI DELLE TECNOLOGIE DELL'INFORMAZIONE

Tutte le tecnologie informatiche (dette anche "tecnologie dell'informazione") sono molto migliorate negli ultimi anni. Qualche studioso ha calcolato che i dati che rappresentano il progresso di questo settore crescono di cento volte ogni dieci anni e che questi progressi avvengono senza che i costi delle tecnologie crescano.

Vediamo tre esempi importanti che si riferiscono ai tre capitoli principali delle tecnologie dell'informazione.

La velocità di calcolo

Nel momento in cui scriviamo queste righe (fine 2011), un calcolatore veloce può eseguire un miliardo di istruzioni aritmetiche al secondo, lavorando su numeri decimali con virgola. Un esempio di istruzione aritmetica con virgola potrebbe essere:

3,5419 moltiplicato per 258,1722

e quest'istruzione potrebbe essere eseguita in un milionesimo di secondo.

In Fig. 5-1 vediamo il prototipo di una unità aritmetica veloce che sarà utilizzata nei calcolatori della prossima generazione. E' fatta da ottanta microscopici calcolatorini, tutti incollati su un unico supporto, capaci di lavorare insieme. Tra un paio di anni quell'unità aritmetica, o unità come quella, diventeranno il cuore dei calcolatori della prossima generazione, rendendo possibili velocità di calcolo dell'ordine di un "teraflop", ossia di mille miliardi di istruzioni aritmetiche decimali al secondo. (Ricordiamo che "chilo" vuol dire mille, "mega" vuol dire un milione, "giga" vuol dire un miliardo, "tera" vuol dire mille miliardi).

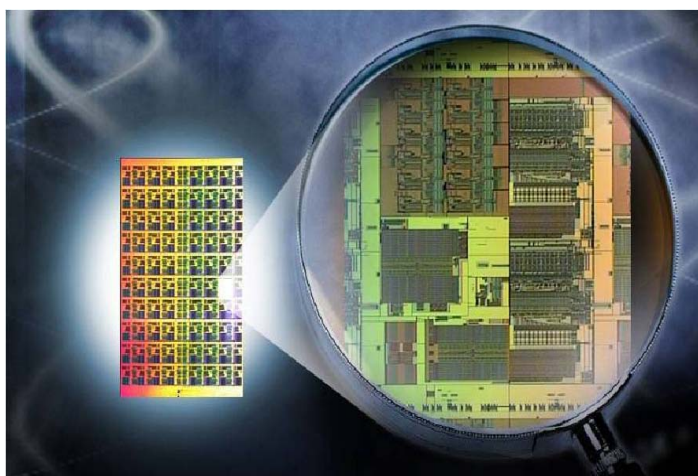


Fig. 5-1



La memoria

La ben nota "chiavetta" (o "pen drive") che la vostra maestra e il vostro professore utilizzano per conservare grandi volumi di dati è una memoria che costa pochi euro e contiene alcuni miliardi di caratteri. Nei laboratori di ricerca sono già disponibili i prototipi di chiavette capaci di memorizzare 256 GigaByte, ossia 256 miliardi di caratteri, pari al contenuto di una biblioteca di 150 mila volumi (Fig. 5-2).



Fig. 5-2

La trasmissione dati

Le fibre ottiche, che, come abbiamo visto, sono le componenti fondamentali per la trasmissione dei dati a grande distanza, diventano ogni giorno più veloci. Oggi, ciascuno dei sottilissimi fili di vetro che vediamo in Fig. 5-3 (con diametro uguale a dieci millesimi di millimetro) può trasmettere, con rapidissimi, piccolissimi lampi di luce, enormi volumi di informazione, dell'ordine dei terabit al secondo, ossia migliaia di miliardi di bit al secondo.

Su un fascio di fibre come quello mostrato in Fig. 5.3, possono viaggiare tutte le telefonate del mondo, su distanze dell'ordine di mille chilometri. Si tenga anche presente che il costo di una fibra è di un centesimo di dollaro per ogni metro di lunghezza.



Fig. 5-3



In un libro del 1997 rivolto ai bambini e ai ragazzi di quegli anni, parlavamo di un sogno: la cosiddetta "TV on demand" via Internet. Ossia sognavamo la possibilità di collegarsi a un "video server" remoto per ricevere in tempo reale il filmato dell'ultima udienza del Papa, o la finale del torneo di tennis femminile di Wimbledon del 1970.

Nel 1997 affermavamo che la velocità con cui venivano trasmessi i dati sulla Rete ci consentiva soltanto la trasmissione del campionato mondiale delle tartarughe (Fig. 5.4). Oggi, in virtù della velocità dei canali di trasmissione dati disponibili, e delle fibre ottiche in particolare, la TV via Internet è diventata un'importante realtà mondiale.

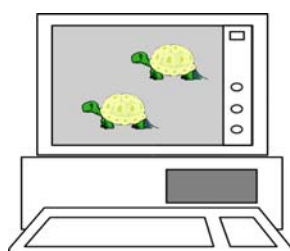


Fig. 5-4

Criptografia e autenticazione

La trasmissione di dati sulla Rete non è intrinsecamente sicura. Ad esempio, il vostro papà, che è molto bravo in informatica, potrebbe intercettare il messaggio inviato da uno di voi a un amico ed evitare una grave marachella (Fig. 5.5).

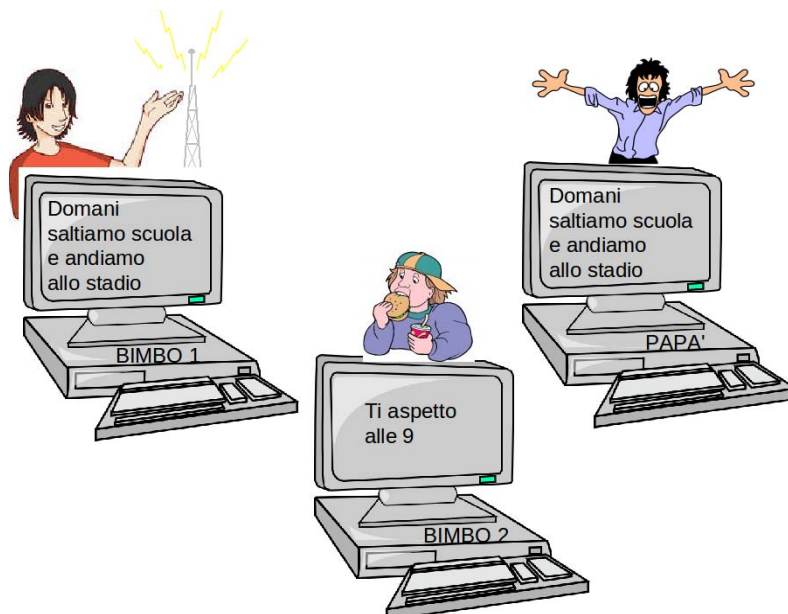


Fig. 5-5



Inoltre, come superare il limite evidenziato nella vignetta di un noto settimanale americano (Fig. 5.6)?

Ossia, come verificare l'autenticità del soggetto con il quale sto colloquiando?



Fig. 5-6

La soluzione a questi due problemi è venuta dalla matematica. Infatti, i matematici hanno inventato soluzioni molto intelligenti per "crittografare" i messaggi, ossia per rendere impossibile la comprensione del contenuto del messaggio da parte di chiunque non conosca il codice segreto della trasmissione. Inoltre, i matematici hanno inventato tecniche raffinate di "firma elettronica", certamente più sicura della firma sulla carta, che consentono di verificare se si sta colloquiando con un cane o con il Presidente degli Stati Uniti. In virtù di queste tecniche, il commercio elettronico è diventato una realtà importante per milioni di uomini.

Il global positioning system (G.P.S.)

La Fig. 5.7 schematizza il principio di funzionamento di un apparato G.P.S., ossia di un piccolo dispositivo in grado di determinare la sua posizione sulla superficie della Terra sulla base di dati ricevuti da alcuni satelliti.

In prima grossolana approssimazione, tale principio di funzionamento può essere descritto nei termini seguenti. L'antenna ricevente dell'apparato a terra "vede" almeno tre di una schiera di satelliti orbitanti a quote relativamente basse. Dal primo di questi tre satelliti l'apparato del G.P.S. riceve un messaggio del tipo: "Io sono il satellite S. Sono le ore H". Lo stesso apparato conosce la posizione P nello spazio occupata da S all'ora H, e da questa informazione deduce che la propria posizione è attualmente allocata su una sfera con centro in P e raggio pari alla distanza ricoperta da un'onda elettromagnetica in un tempo pari al tempo



intercorso fra l'ora H dichiarata dal satellite e l'ora H^* in cui il messaggio è stato ricevuto. L'apparato del G.P.S. allocato a bordo del mezzo mobile ripete lo stesso calcolo sui dati di almeno due altri satelliti determinando la propria posizione come intersezione di almeno tre sfere. Comunque, al fine di migliorare la precisione del rilevamento si usano anche i dati di almeno un quarto satellite.

Attualmente il G.P.S. consente precisioni nella determinazione della propria posizione dell'ordine dei centimetri.

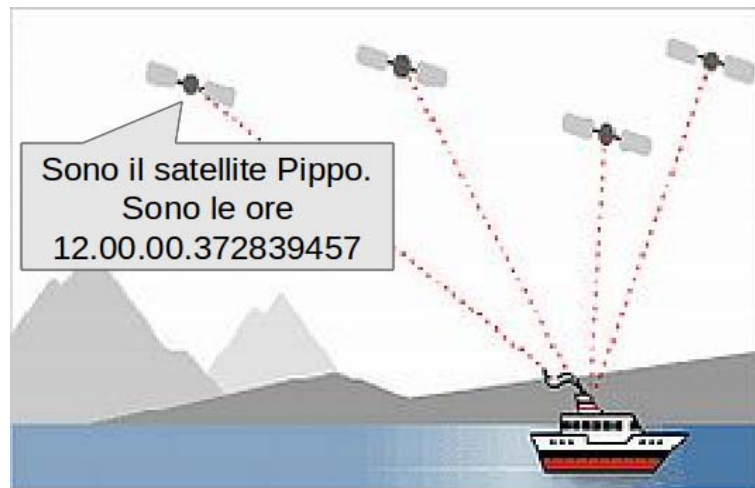


Fig. 5-7

L'intelligenza artificiale e i sistemi esperti

Abbiamo visto che un calcolatore può eseguire un miliardo di istruzioni aritmetiche decimali in un secondo, mentre un bravo matematico può eseguire una di quel miliardo di istruzioni in un minuto. Sfortunatamente alla bravura del calcolatore in matematica corrisponde un'incredibile incapacità a risolvere problemi anche banali, se questi richiedono intuito o fantasia. Ad esempio, un bimbo di due anni riconosce perfettamente la mamma dalla zia, mentre un calcolatore non sempre ci riesce, a meno che la mamma sia bionda e la zia bruna.

L'intelligenza artificiale è quel capitolo dell'informatica che studia come dotare un calcolatore di qualità come la fantasia, l'intuito, la creatività, che sono tipiche dell'intelligenza naturale, degli uomini o anche soltanto degli animali. È una scienza molto interessante ma anche molto difficile, che non ha prodotto negli ultimi anni i risultati che si aspettavano alcuni decenni fa.

Un'area importante dell'intelligenza artificiale è quella dei cosiddetti "sistemi esperti", ossia di quegli automatismi, di norma programmati su calcolatore, che incorporano le conoscenze di uomini "esperti" in determinate aree professionali e sono dotati di meccanismi intelligenti per la formulazione di diagnosi o indicazioni operative. Ad esempio, un sistema esperto per la medicina produce una diagnosi sulla base delle indicazioni della cartella clinica e dei dati del laboratorio. Un sistema di ausilio alle decisioni finanziarie identifica gli investimenti più redditizi analizzando i dati dei bilanci e l'evoluzione nel tempo



dei mercati.

L'insieme delle cose che un sistema esperto può fare è così ampio da rendere molto difficile una classificazione esauriente. Vediamo pochi esempi. Il primo è quello dei sistemi diagnostici di apparati meccanici, elettrotecnici, elettronici o informatici, anche di notevole complessità. Nel caso di Fig. 5-8, avendo preso fuoco tutto l'impianto, un'opportuna segnalazione arriva sul video del calcolatore che opera il controllo.



Fig. 5-8

Il secondo esempio riguarda i sistemi di ausilio, che collaborano con gli scienziati e i tecnici nella progettazione di macchine o impianti, prevalentemente meccanici, elettrotecnici, elettronici, e anche, più raramente, civili e aeronautici.

Il terzo esempio è quello dei sistemi di controllo delle operazioni elementari di fabbricazione, assemblaggio e movimentazione in impianti produttivi complessi.





Informatica e Internet oggi e domani

L'informatica rappresenta una delle più importanti rivoluzioni scientifiche e tecnologiche degli ultimi cinquanta anni, e forse, in prospettiva, dell'intera storia dell'umanità.

L'informatica pervade ormai quasi tutti i prodotti industriali, dalla lavatrice al forno a microonde, dal giocattolo alla pompa di benzina.

La metà del costo di un'automobile è rappresentato da componenti informatiche e vi sono aeroplani che sono guidati da molte decine di calcolatori che controllano ogni movimento del velivolo con una velocità con cui nessun pilota, per quanto bravo, potrebbe operare.

Sino a qualche anno fa, i pezzi di un'automobile, di un trattore, di qualunque oggetto meccanico erano prodotti a mano, da artigiani molto bravi che comandavano tutte le operazioni elementari delle macchine che lavoravano ogni pezzo. Oggi molti pezzi sono prodotti, in modo completamente automatico, da macchine complicate, dove un certo numero di utensili lavorano contemporaneamente sulle diverse superfici di ogni pezzo.

Sino a qualche anno fa, centinaia di operai lavoravano nelle linee dove venivano montate le singole parti delle macchine; oggi, poche decine di uomini sono sufficienti per controllare il lavoro delle linee di assemblaggio automatiche, dove operano migliaia di robot (Fig. 5-10).

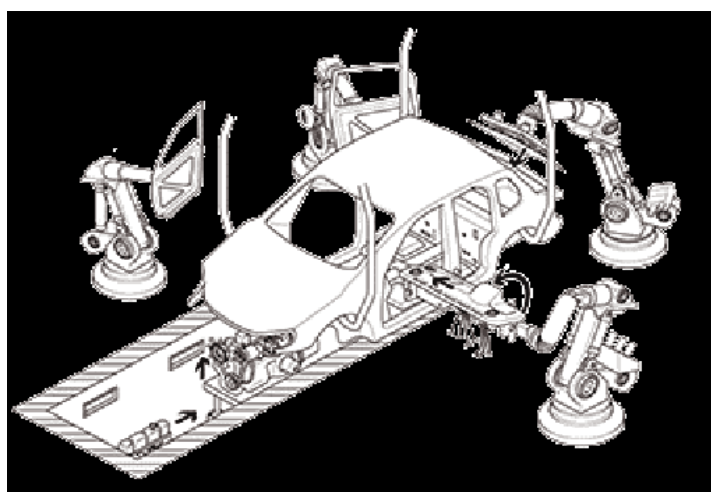


Fig. 5-10

Per quanto concerne Internet, ogni giorno scopriamo nuove fantastiche applicazioni già operative ora o destinate a rivoluzionare molte attività dell'uomo nell'arco di pochi anni. Ci limitiamo qui a pochi esempi.

Il telelavoro

L'impiegato o il dirigente di un ufficio può lavorare dalla propria abitazione o dal luogo della vacanza utilizzando i servizi della Rete per esaminare i documenti dell'ufficio o per colloquiare in videoconferenza con i colleghi e i collaboratori.



La teleassistenza tecnica

Il tecnico di un qualunque apparato di un'abitazione o di un'azienda, molto spesso, può procedere alla riparazione operando in remoto (Fig. 5-11).



Fig. 5-11

Il commercio elettronico e la banca in Rete

Oggi molti siti specializzati ci consentono di acquistare beni anche molto costosi, operando sulla Rete. Accedendo ai servizi di una banca via Internet è possibile pagare il bene acquistato che i servizi postali faranno arrivare alla nostra casa con le tecniche tradizionali.

La teledidattica

Sino a pochi anni or sono la teledidattica era attuata attraverso trasmissioni televisive oppure attraverso la distribuzione di video cassette. Oggi migliaia di lezioni e corsi sono disponibili in Rete oppure sono trasmesse attraverso Internet (Fig. 5-12). Entro una decina di anni tutti i libri del mondo potranno essere letti via Internet.

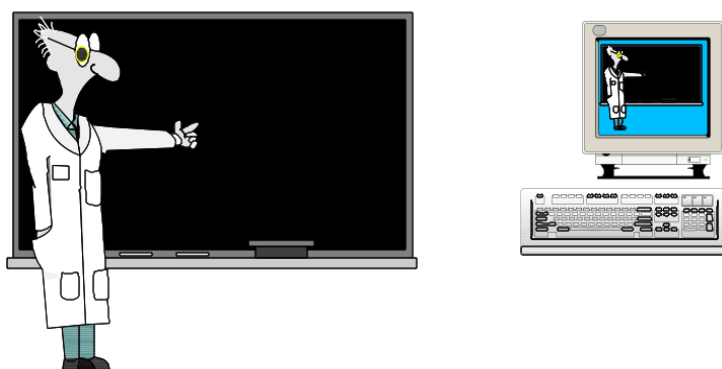


Fig. 5-12

La telemedicina

E' molto probabile che presto i medici potranno visitare i loro pazienti utilizzando opportuni sensori che trasmetteranno attraverso la Rete i valori della respirazione, della pressione del sangue, della frequenza cardiaca (Fig. 5-13).



Fig. 5-13

La telechirurgia

Arriverà poco dopo la telemedicina, ma diventerà anch'essa un'importante realtà. Gli esperimenti di telechirurgia, rivolti soprattutto alle esigenze dei paesi più poveri, sono già stati iniziati.

La telerobotica

Entro una decina di anni moltitudini di robot comandati via Internet scaveranno gallerie, opereranno nelle miniere, cercheranno le mine esplosive nei campi, collaboreranno nei lavori domestici (Fig. 5-14). Molto probabilmente gli eserciti arruoleranno robot guerrieri. Speriamo che gli uomini comprendano presto che le guerre sono il male supremo, perché i robot guerrieri potrebbero produrre enormi distacchi.



Fig. 5-14