

CADDEL

Spediz. in abbonamento postale GR. II/70 L. 2.000  
(...)

# 22 CORSO PRATICO COL COMPUTER

421743

diretto da GIANNI DEGLI ANTONI

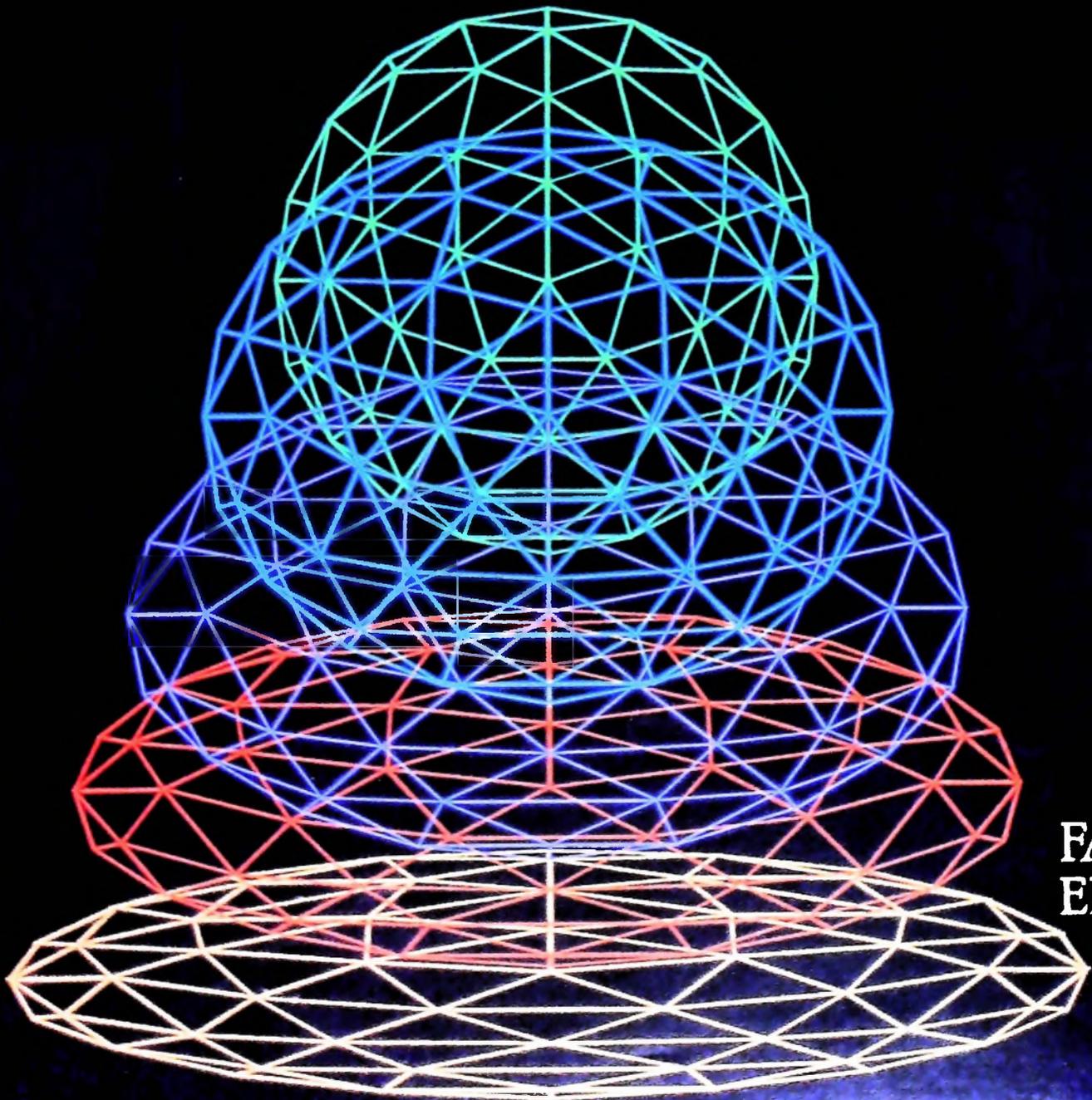
è una iniziativa  
**FABBRI EDITORI**

in collaborazione con  
**BANCO DI ROMA**

e **OLIVETTI**



BATTERY LOW

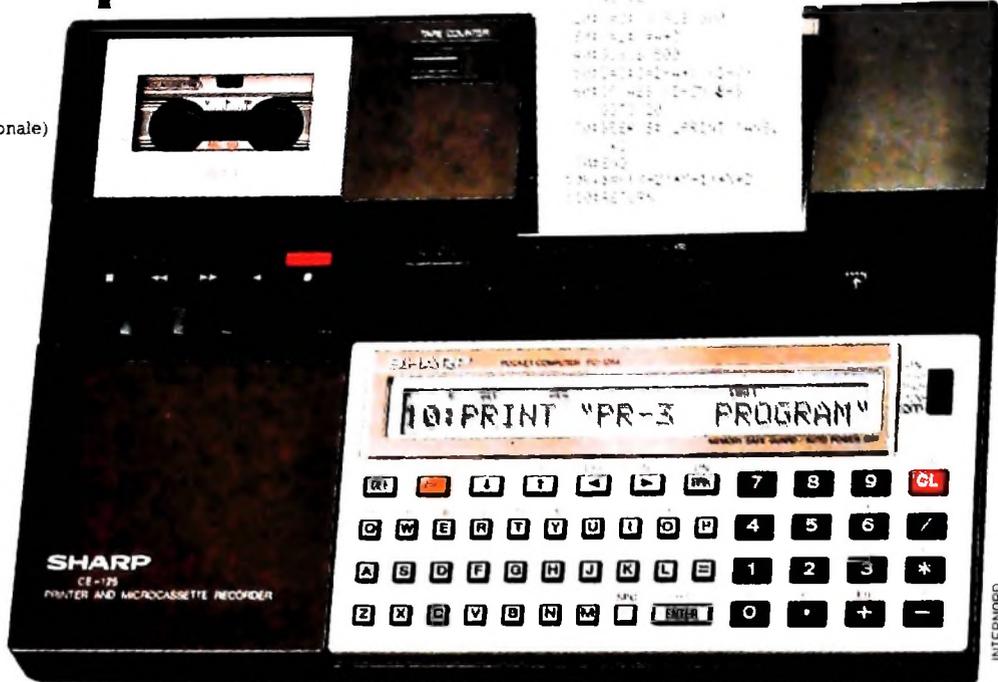


**FABBRI  
EDITORI**

# Un nuovo Computer tascabile da Sharp.

## La giusta combinazione tra semplicità d'uso e disponibilità di programmi, tra superiorità del sistema e reale portatilità

Il PC-1251 costa solo  
L. 299.500 + IVA  
(Il sistema stampante/registratore a microcassette è opzionale)



### SHARP PC-1251

#### Basic esteso con comandi ed istruzioni estremamente versatili

Il Basic realizzato dalla Sharp comprende dei nuovi comandi che vi consentono le più vaste applicazioni. Il comando DIM, ad esempio vi permette di trattare matrici bidimensionali di variabili. Si possono effettuare operazioni con le stringhe. Avete a disposizione il comando IN-KEYS per effettuare inserimenti da tastiera senza fermare il programma. Grazie al BASIC esteso questo Computer, il più tascabile di tutti, appartiene alla categoria dei Personal Computer: ideale per neofiti, versatile per gli esperti.

#### C-MOS e CPU a 8-bit per rapide elaborazioni

Per mezzo del CPU a 8 bit si ottiene una veloce elaborazione dei dati, mentre l'utilizzazione dei C-MOS permette un funzionamento con basso consumo di corrente per una più lunga durata delle batterie.

#### 24 KBytes di ROM, 4,2 KBytes di RAM. Nonostante le sue dimensioni il PC-

1251 ha 24 KBytes di ROM e 4,2 di RAM, di cui 3,7 a disposizione dell'utente. Con questa grande capacità di memoria potete scrivere o caricare da nastro anche lunghi programmi.

#### Protezione della memoria

Le pile mantengono l'alimentazione alle memorie RAM proteggendole anche a macchina spenta. Grazie a questa caratteristica i programmi che avete in memoria sono sempre pronti per essere utilizzati.

#### Funzione Pass

Potete assegnare un codice segreto ad ogni programma che limita l'accesso alle sole persone che lo conoscono.

#### Visore a 24 caratteri con matrice a punti 5x7

Il visore è in grado di visualizzare contemporaneamente fino a 24 caratteri, ognuno dei quali chiaramente rappresentato per mezzo di una matrice da 5x7 punti. La luminosità del display è regolabile.

#### Stampante/Registratore a microcassette CE-125 (opzionale)

La CE-125 è una unità speciale comprendente sia la stampante che il registratore a microcassette, realizzata per funzionare con il PC-1251. Vi consente sia una stampa veloce che il trasferimento su nastro dei programmi. Le ridotte dimensioni del sistema consentono d'avere la potenza del PC-1251 e la versatilità della CE-125 sempre e ovunque con voi.

#### Registratore a microcassette

Non è necessario avere un registratore esterno quando dovete "salvare" o ricaricare un programma. Usate semplicemente una microcassetta e tutto è risolto. Questo sistema elimina gli errori di trasferimento.

I programmi sono la chiave per il successo di un Computer. Col PC-1251 siete già sulla giusta strada. Perché? Perché potete utilizzare sia i già collaudati programmi del PC-1211 che i nuovi programmi realizzati per il PC-1251 o quelli che voi stessi realizzerete grazie alla semplicità del BASIC.

#### Elenco dei programmi inseriti nel manuale del PC-1251

- Radici di un'equazione secondo il metodo di Newton
- Percentuale, varianza, deviazione standard
- Intersezioni tra cerchi e rette
- Calcolo del numero dei giorni
- Pratica con la tastiera
- Atterraggio morbido
- Prova di memoria
- Caccia alla pulce
- Rotazione doppia

#### Elenco dei programmi inseriti nel manuale della CE-125

- Prodotto di matrici
- Integrazione numerica secondo il metodo di Simpson
- Coefficiente di correlazione e regressione lineare
- Istogrammi
- Conversione  $\Delta - Y$  e  $Y - \Delta$
- Curva clotoidale
- Incroci
- Ordinamento
- Calcolo delle rate

- Bionimo (semigrafico)
- Ritratto stampato

#### Programmi su microcassette CE-12A/B/C

Le microcassette sono disponibili opzionalmente  
CE-12A - 19 programmi di statistica e giochi  
CE-12B - 20 programmi di ingegneria e giochi  
CE-12C - 20 programmi di matematica e giochi

Direttore dell'opera  
GIANNI DEGLI ANTONI

Comitato Scientifico  
GIANNI DEGLI ANTONI  
Docente di Teoria dell'informazione, Direttore dell'Istituto di Cibernetica dell'Università degli Studi di Milano

UMBERTO ECO  
Ordinario di Semiotica presso l'Università di Bologna

MARIO ITALIANI  
Ordinario di Teoria e Applicazione delle Macchine Calcolatrici presso l'Istituto di Cibernetica dell'Università degli Studi di Milano

MARCO MAIOCCHI  
Professore Incaricato di Teoria e Applicazione delle Macchine Calcolatrici presso l'Istituto di Cibernetica dell'Università degli Studi di Milano

DANIELE MARINI  
Ricercatore universitario presso l'Istituto di Cibernetica dell'Università degli Studi di Milano

Curatori di rubriche  
TULLIO CHERSI, ADRIANO DE LUCA (Professore di Architettura dei Calcolatori all'Università Autonoma Metropolitana di Città del Messico), GOFFREDO HAUS, MARCO MAIOCCHI, DANIELE MARINI, GIANCARLO MAURI, CLAUDIO PARMELLI, ENNIO PROVERA

Testi  
ADRIANO DE LUCA, GIANCARLO MAURI, CLAUDIO MORDA, Eidos (TIZIANO BRUGNETTI), Etnoteam (ADRIANA BICEGO)

Tavole  
Logical Studio Communication  
Il Corso di Programmazione e BASIC è stato realizzato da Etnoteam S.p.A., Milano  
Computergrafica è stato realizzato da Eidos, S.c.r.l., Milano  
Usare il Computer è stato realizzato in collaborazione con PARSEC S.N.C. - Milano

Direttore Editoriale  
ORSOLA FENGLI

Coordinatore settore scientifico  
UGO SCAIONI

Redazione  
MARINA GIORGETTI  
LOGICAL STUDIO COMMUNICATION

Art Director  
CESARE BARONI

Impaginazione  
BRUNO DE CHECCHI  
PAOLA ROZZA

Programmazione Editoriale  
ROSANNA ZERBARINI  
GIOVANNA BREGGÈ

Segretaria di Redazione  
RENATA FRIGOLI  
LUCIA MONTANARI

**NEL PROSSIMO NUMERO  
IN OMAGGIO  
IL DECIMO POSTER  
"LA STORIA  
DELL'INFORMATICA"**

Corso Pratico col Computer - Copyright (c) sul fascicolo 1984 Gruppo Editoriale Fabbri, Bompiani, Sonzogno, Etas S.p.A., Milano - Copyright (c) sull'opera 1984 Gruppo Editoriale Fabbri, Bompiani, Sonzogno, Etas S.p.A., Milano - Prima Edizione 1984 - Direttore responsabile GIOVANNI GIOVANNINI - Registrazione presso il Tribunale di Milano n. 135 del 10 marzo 1984 - Iscrizione al Registro Nazionale della Stampa n. 00262, vol. 3, Foglio 489 del 20/9/1982 - Stampato presso lo Stabilimento Grafico del Gruppo Editoriale Fabbri S.p.A., Milano - Diffusione Gruppo Editoriale Fabbri S.p.A. via Mecenate, 91 - tel. 50951 - Milano - Distribuzione per l'Italia A & G Marco s.a.s. via Forzezza 27 - tel. 2526 - Milano - Pubblicazione periodica settimanale - Anno I - n. 22 - esce il giovedì - Spedizione in abb. postale - Gruppo 11/70 - L'Editore si riserva la facoltà di modificare il prezzo nel corso della pubblicazione, se costretto da mutate condizioni di mercato.

concessionaria  
per l'Italia

MELCHIONI

**TUTTA LA POTENZA DI UN COMPUTER  
NEL PALMO DELLA TUA MANO**

Per ulteriori informazioni scrivete a  
MELCHIONI - Divisione Pocket Computer - 20135 MILANO - Via P. Colletta 37

**SHARP**

# CICLI DI ISTRUZIONI

L'unità di controllo, come un direttore d'orchestra, organizza l'accesso alle istruzioni e la loro esecuzione.

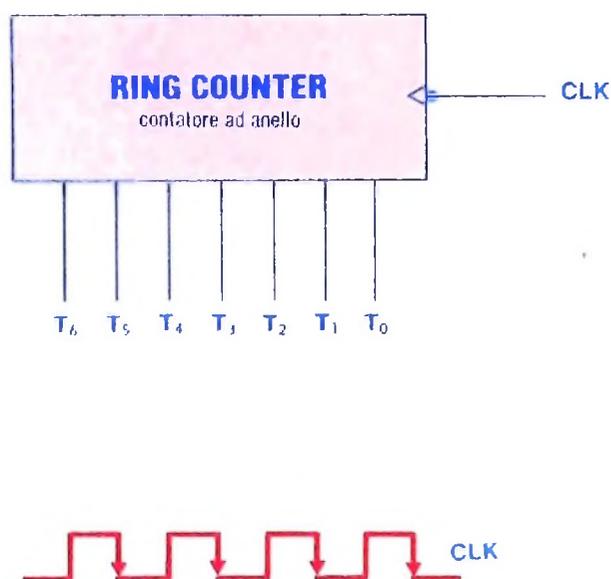
Il controllo, come abbiamo avuto occasione di dire, è come un direttore di orchestra. Una delle sue funzioni principali consiste nell'accesso alle istruzioni (FETCH CYCLE) e nella esecuzione delle istruzioni stesse (EXECUTE CYCLE).

Per poter realizzare questa prassi abbiamo bisogno di un circuito speciale chiamato "contatore ad anello" (*ring counter*) (figura 1).

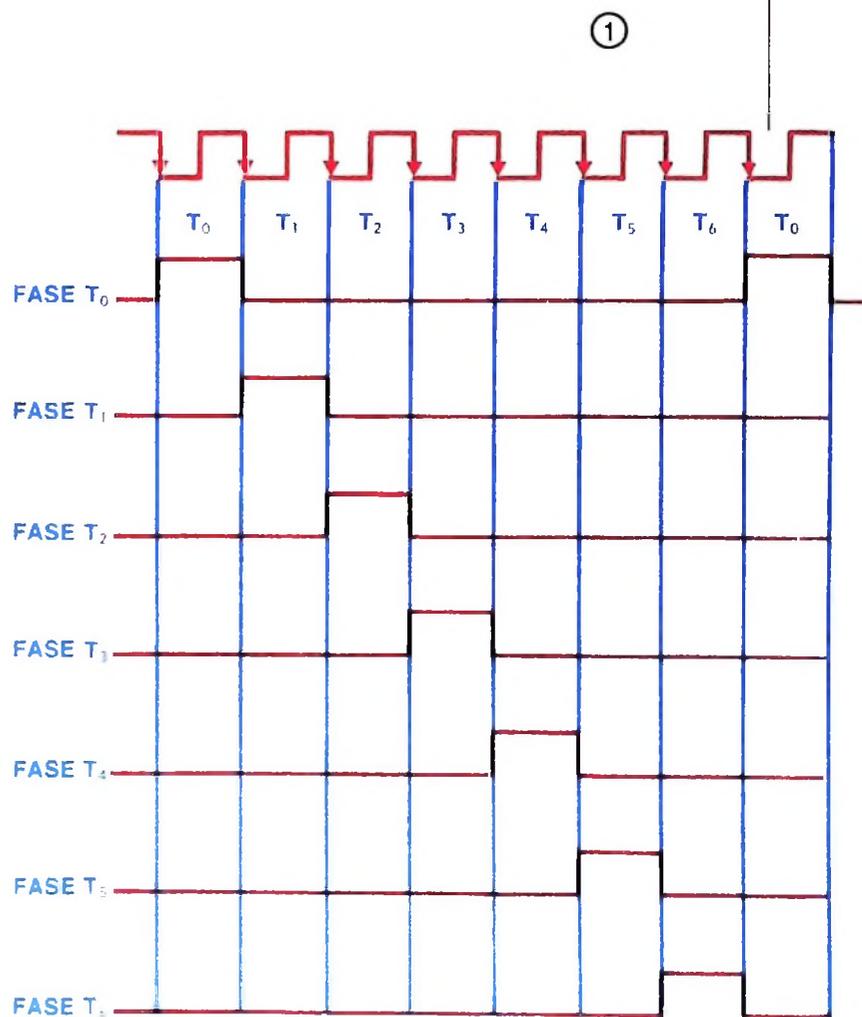
Questo contatore ha due entrate: con  $CLR=1$ , tutte le uscite da  $T_1$  a  $T_6$  vanno a zero, tranne  $T_0$  che resta uguale a uno. La seconda entrata è una entrata di clock (orologio) ed ha lo

stato attivo *negativo*; di conseguenza a ogni transizione negativa l'uscita uguale a 1 si sposta di un posto, come mostra la figura, cioè abbiamo un "uno viaggiante" che gira ad anello per tutte le uscite.

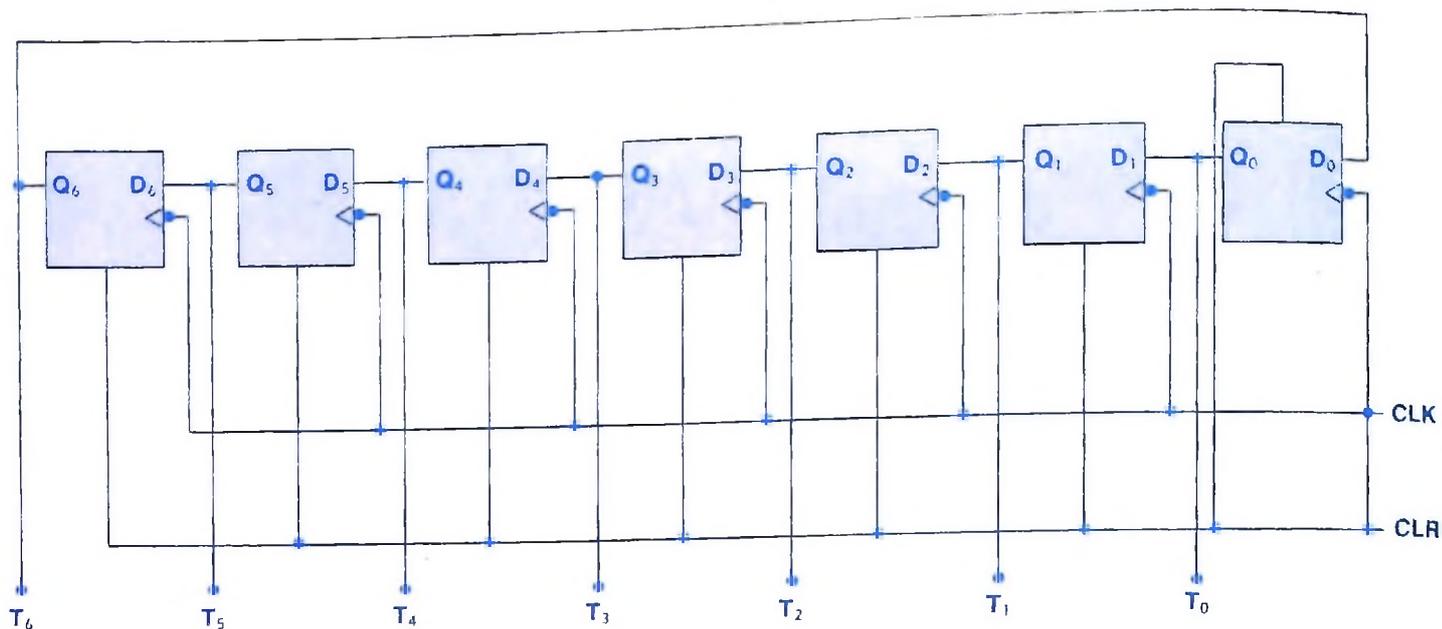
La forma di un circuito simile è data nella figura 2: come possiamo vedere, esso è composto da sette flip-flop di tipo D, con le uscite chiuse ad anello. Il comando  $CLR$  è collegato al  $RESET$  di tutti, tranne il primo, che è collegato al  $SET$ , cosicché quest'ultimo avrà la sua uscita  $T_0=1$  (al clock), mentre le altre sono a zero.



Il contatore ad anello (in alto) è un circuito fondamentale per l'attività dell'unità di controllo. Ha due entrate e sette uscite. Una delle entrate è per il segnale di clock, raffigurato qui sopra, che è attivo nella transizione negativa (dal livello alto al livello basso). Con l'entrata  $CLR = 1$  tutte le uscite del contatore vanno a 0, tranne  $T_0$ , che resta uguale a 1. A ogni impulso di clock, come si può vedere nel diagramma a destra, l'1 scorre di una posizione, verso le uscite successive.

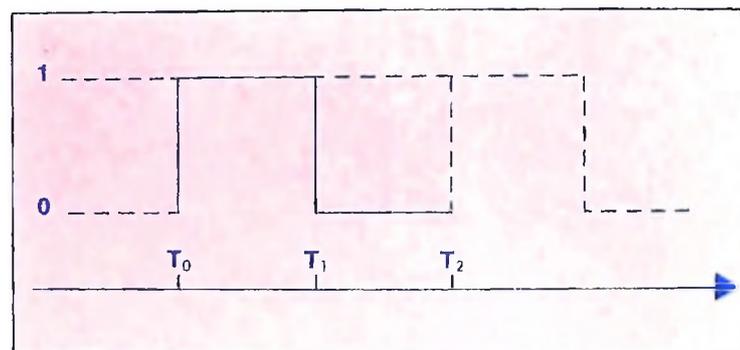


2



In alto, una realizzazione del contatore ad anello, mediante sette flip-flop di tipo D, con le uscite chiuse ad anello. Sotto, andamento della forma d'onda del clock.

3



### Ciclo di accesso

Per completare un ciclo di accesso è necessario completare due fasi,  $T_0$  e  $T_1$ . Analizziamo come avviene questo processo: il controllo, attraverso un sistema interno, genera una parola di controllo, che varia di contenuto in relazione all'azione che sta compiendo e al clock. Prima, però, bisogna definire il sistema di clock: prendiamo in considerazione la forma d'onda del clock (figura 3). In un ciclo abbiamo quattro diverse configurazioni: la prima, quella in  $T_0$ , è una transizione positiva, la seconda, da  $T_0$  a  $T_1$ , è un valore alto continuo, la terza in  $T_1$  è una transizione negativa e l'ultima, da  $T_1$  a  $T_2$ , è un valore basso, continuo. Attraverso questi quattro stati diversi del segnale di clock possiamo sincronizzare nel tempo diversi processi. Prima di tutto, tutti i comandi di ENABLE (ABILITAZIONE) come pure il comando VMA vengono emessi ed eseguiti nella parte bassa continua del clock, mentre i comandi di LOAD e W/R, emessi pure nella parte bassa del clock, vengono effettuati con la transizione positiva del clock; come vedremo in seguito, questa "tempistica" ci assicura il giusto funzionamento di tutti i registri rispettando i tempi necessari. Naturalmente, nessuno ci vieta di scegliere altri stati attivi del clock, come per esempio la transizione negativa, dato che ci sono moltissimi registri che funzionano con stati attivi negativi; in realtà, ognuno può scegliere quello che vuole: l'importante è tener presente le necessità che occorre soddisfare in base alla velocità del sistema. Comunque, la tendenza, che sembra ormai affermata, è che tutti i segnali di controllo siano attivi negativi; nell'UAMICRO I, invece, sono stati usati attivi positivi, più facili da capire. Torniamo ora alla parola di controllo e alle fasi della figura

4. Nella fase  $T_0$ , abbiamo due comandi uguali a 1 mentre tutti gli altri sono zero. Con  $E_{PC} = 1$  l'indirizzo del PC (*program counter*, il contatore di programma), che punta alla memoria, viene messo nel bus degli indirizzi, nella parte bassa continua del clock, mentre con  $LN = 1$  (essendo questo un comando di LOAD) l'indirizzo viene caricato nel MAR (*memory auxiliary register*, registro ausiliario di memoria) nella transizione positiva che segue. Nella fase  $T_1$  abbiamo  $VMA = 1$ , cioè gli indirizzi sono validi. Esso viene emesso quando il clock si incontra nella fase continua bassa, cosicché i dati possono passare al bus; in questa stessa fase vengono fuori  $L_{IR}$  e  $I_{PC} = 1$ , per cui alla transizione positiva del clock, da una parte il PC si incrementa e punta all'indirizzo della memoria che segue, dall'altra, con  $L_{IR}$  abilitato, il contenuto del bus (una istruzione in questo caso) viene immagazzinato nel registro IR dove inizia il processo di esecuzione. Possiamo sintetizzare le due fasi nei tempi nella seguente forma:  
 $T_0 = PC \rightarrow MAR$  (fase di indirizzo)  
 $T_1 = (RAM) \rightarrow IR$  e  $PC + 1 \rightarrow PC$  (fase di memoria e incremento).

4

FASE	$E_A$	$L_A$	$S_0$	$E_0$	$L_0$	$E_0$	$I_{PC}$	$E_{PC}$	$L_{AR}$	$E_{AR}$	$L_{IR}$	LM	VMA	W/R
$T_0$	0	0	0	0	0	0	0	1	0	0	0	1	0	0
$T_1$	0	0	0	0	0	0	1	0	0	0	1	0	1	0

Gli stati della parola di controllo nelle fasi  $T_0$  e  $T_1$ .

### Ciclo di esecuzione

Il registro IR (*instruction register*, registro delle istruzioni) è il registro, come abbiamo detto, dove incomincia il processo di esecuzione. In esso, attraverso opportuni circuiti logici, che vedremo più avanti, viene analizzato il codice in ingresso, al fine, dapprima, di individuare l'istruzione e, immediatamente dopo, di inviarla nel registro di controllo. In base al tipo di questa istruzione, il controllo emette una sequenza di parole adatte a eseguirla. Vediamo come: supponiamo che l'istruzione entrata fosse la LDA XX (figura 5). Essendo questa una istruzione MRI (*memory register instruction*, istruzione del registro di memoria), bisogna tornare in memoria a prelevare l'operando; il PC sta già puntando alla locazione di memoria seguente (figura 6), per cui la fase  $T_2$  consiste in  $E_{PC} = 1$  che emette l'indirizzo, e con  $L_M = 1$  che al clock immagazzina l'indirizzo nel MAR (in questa nuova figura abbiamo adottato la tecnica di indicare solo i comandi che sono nello stato attivo tralasciando tutti gli altri). Nella fase  $T_3$  convalidiamo l'indirizzo con VMA, carichiamo il contenuto della MAR con  $L_{AR}$  e, infine, incrementiamo il PC con  $I_{PC}$ .

Nella fase  $T_4$ , con  $E_{AR}$  abilitiamo il contenuto di AR e con  $L_M$  lo carichiamo nel MAR. Nella fase  $T_5$ , con VMA convalidiamo l'indirizzo, mentre con  $L_A$  carichiamo il contenuto della memoria dell'accumulatore A. La fase  $T_6$  non comporta nessuna operazione, però è un passo necessario per completare il circolo del contatore ad anello. Una domanda naturale in questo caso è quella di chiedere se non è possibile saltare le fasi non utilizzate e così guadagnare in velocità; questo è infatti possibile e lo faremo nell'UAMICRO II e III, che sono microprocessori più evoluti.

Vediamo l'istruzione LDB XX (figura 7): questa istruzione è molto simile alla precedente tranne che nella fase  $T_5$  dove il segnale è  $L_B$  che carica il dato nell'accumulatore B come voleva l'istruzione.

### Istruzione STA XX

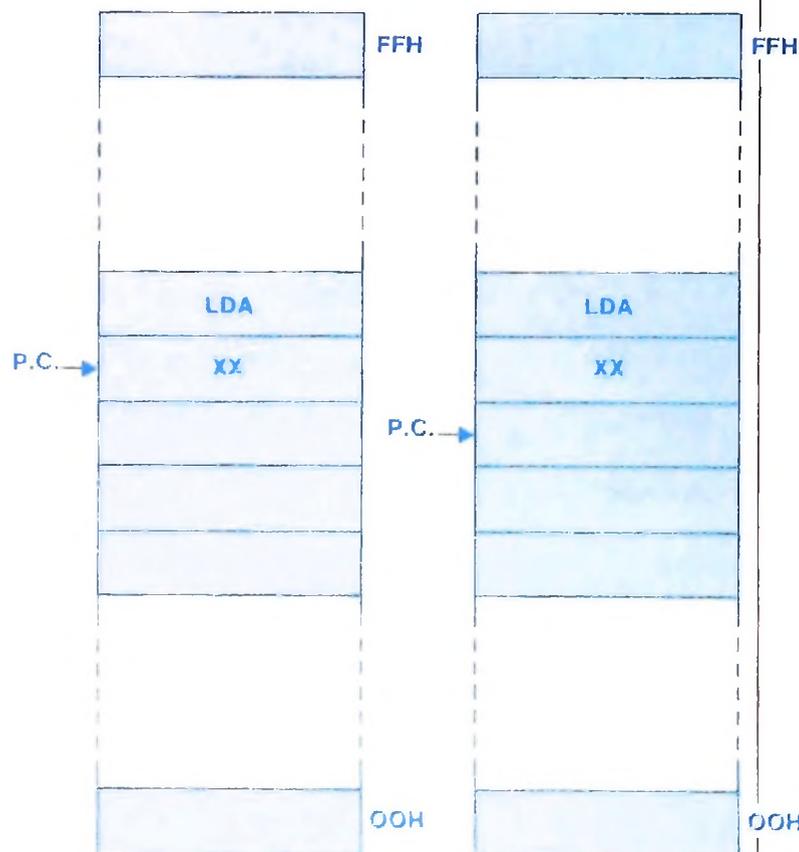
Anche questa istruzione (figura 8) ha le stesse fasi delle altre tranne la  $T_5$  dove, in questo caso, essendo un trasferimento dall'accumulatore alla memoria, c'è bisogno di un  $E_A$ , di una

5

ISTRUZIONE LDA XX		
FASE	CON	
$T_2$	$E_{PC}$	$L_M$
$T_3$	$I_{PC}$	$L_{AR}$ VMA
$T_4$	$E_{AR}$	$L_M$
$T_5$	VMA	$L_A$
$T_6$		

Come si attivano i comandi nelle varie fasi per l'istruzione LDA XX.

6



Il diagramma mostra come il PC (contatore di programma) si autoincrementa e punta alla posizione successiva. A sinistra prima, a destra dopo l'incremento.

7

ISTRUZIONE LBD XX			
FASE	CON		
T <sub>2</sub>	E <sub>PC</sub>	L <sub>M</sub>	
T <sub>3</sub>	I <sub>PC</sub>	L <sub>AR</sub>	VMA
T <sub>4</sub>	E <sub>AR</sub>	L <sub>M</sub>	
T <sub>5</sub>	VMA	L <sub>B</sub>	
T <sub>6</sub>			

9

ISTRUZIONE STB XX			
FASE	CON		
T <sub>2</sub>	E <sub>PC</sub>	L <sub>M</sub>	
T <sub>3</sub>	VMA	L <sub>AR</sub>	I <sub>PC</sub>
T <sub>4</sub>	E <sub>AR</sub>	L <sub>M</sub>	
T <sub>5</sub>	VMA	E <sub>B</sub>	W/R
T <sub>6</sub>			

8

ISTRUZIONE STA XX			
FASE	CON		
T <sub>2</sub>	E <sub>PC</sub>	L <sub>M</sub>	
T <sub>3</sub>	VMA	L <sub>AR</sub>	I <sub>PC</sub>
T <sub>4</sub>	E <sub>AR</sub>	L <sub>M</sub>	
T <sub>5</sub>	VMA	E <sub>A</sub>	W/R
T <sub>6</sub>			

10

ISTRUZIONE ADD XX			
FASE	CON		
T <sub>2</sub>	E <sub>PC</sub>	L <sub>M</sub>	
T <sub>3</sub>	I <sub>PC</sub>	L <sub>AR</sub>	VMA
T <sub>4</sub>	E <sub>AR</sub>	L <sub>M</sub>	
T <sub>5</sub>	L <sub>B</sub>	VMA	
T <sub>6</sub>	E <sub>U</sub>	L <sub>A</sub>	

Come si attivano i comandi nelle varie fasi per le istruzioni LBD XX, STA XX, STB XX e ADD XX.

convalida dell'indirizzo con VMA e di un comando di scrittura W/R (bisogna ricordare che questo segnale svolge due funzioni: quando è uguale a 1 permette la scrittura, quando è 0 la lettura; ecco perché nelle precedenti istruzioni non appariva).

### Istruzione STB XX

Uguale alla precedente (figura 9), tranne che nella fase T<sub>5</sub>, con il comando E<sub>B</sub> che abilita l'accumulatore B.

### Istruzione ADD XX

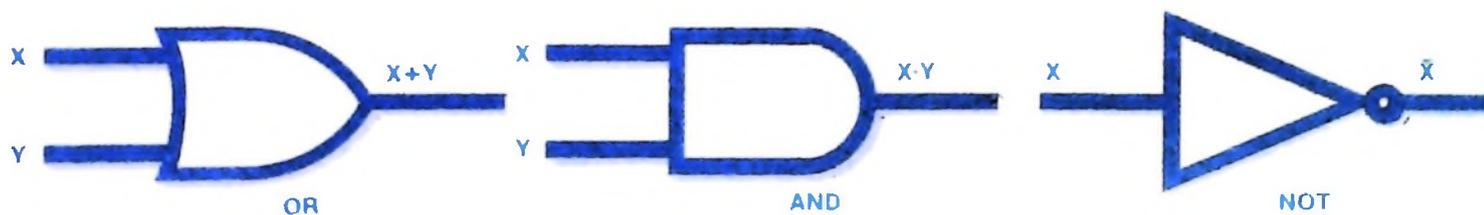
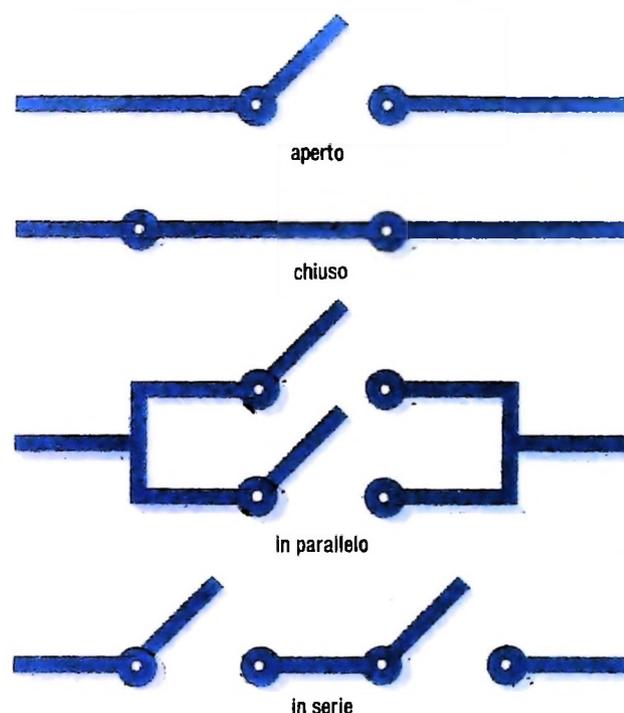
Con questa istruzione (figura 10) incominciamo ad entrare nel campo delle istruzioni che vengono definite come "istruzioni composte". Se l'analizziamo un po' attentamente, adottando la stessa tecnica usata finora, vediamo che ci troviamo in presenza di più istruzioni elementari messe assieme; per esempio, se prendiamo le operazioni a una a una (quella di caricare l'accumulatore B con un dato preso in memoria e

quella di sommarlo con il contenuto dell'accumulatore A), possiamo notare che il processo rappresenta l'esecuzione di due istruzioni, rendendo necessario tornare tre volte in memoria a prendere dati e istruzioni, mentre con l'istruzione ADD XX si va in memoria due volte, realizzando di conseguenza un risparmio di tempo e una minore occupazione di memoria. La tendenza delle istruzioni dei microprocessori moderni, dovuta sia al progresso tecnologico che all'esperienza acquisita, è di diventare sempre più composte. È interessante vedere com'è perfetta la sincronia dei tempi per la realizzazione della somma e il deposito in A senza creare confusioni e errori. Analizziamo la fase T<sub>6</sub>: il comando S<sub>U</sub> = 0 vuol dire sommare, quindi, essendo questo comando asincrono (perciò presente nella parte bassa del clock), la somma fra i due addendi viene eseguita immediatamente, con il comando E<sub>U</sub> = 1, anche questo asincrono. Il risultato è continuamente presente nel bus, mentre il comando L<sub>A</sub> = 1 è un comando sincrono, quindi necessita del clock per la sua realizzazione, per cui alla transizione positiva che segue il risultato della somma viene caricato in A (distruggendo il valore precedente) e l'operazione così viene portata felicemente a termine.

# RETI LOGICHE

Come l'algebra di Boole ci può servire per descrivere circuiti elettronici che realizzano funzioni logiche.

Prendiamo un circuito elettrico elementare, con un solo interruttore che può assumere due stati ben distinti - chiuso e aperto - in corrispondenza dei quali nel circuito circola o meno corrente. Circuiti elementari di questo genere possono essere combinati in vario modo per creare circuiti complessi, dal comportamento più vario. Due circuiti elementari possono essere collegati in parallelo, in modo che nel circuito risultante passi corrente se almeno uno dei due interruttori è chiuso: oppure, possono essere collegati in serie, dando un circuito in cui la corrente passa solo se ambedue gli interruttori sono chiusi. Due circuiti elementari in parallelo realizzano un analogo della funzione di "somma logica" (la "o" della logica degli enunciati, l'unione dell'algebra degli insiemi), mentre due circuiti in serie realizzano un analogo del "prodotto logico" (la "e" della logica degli enunciati, l'intersezione dell'algebra degli insiemi). Se conveniamo di considerare "complemento" di un circuito X un circuito  $\bar{X}$  che abbia il comportamento opposto (vi circola corrente nelle stesse condizioni in cui in X non circola, e viceversa) e aggiungiamo al nostro armamentario un circuito sempre aperto e un circuito sempre chiuso, abbiamo tutti gli elementi per una "algebra



dei circuiti", che altro non è se non un'algebra booleana, a cui possiamo estendere tutto ciò che abbiamo detto sin qui sull'algebra di Boole.

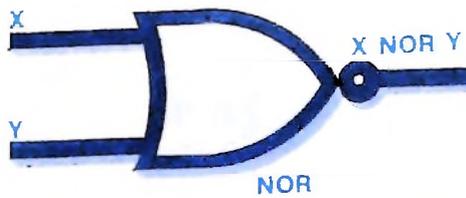
## Le porte logiche

Come siano effettivamente realizzati gli interruttori, in questo contesto, poco importa: ciò che è pertinente è solo il tipo di funzione svolta. Convenzionalmente, si sono scelti alcuni

simboli grafici per identificare i tre tipi fondamentali di circuiti che realizzano le operazioni booleane di somma, prodotto e complementazione, cui, nel linguaggio tipico dell'elettronica, si dà il nome di "porte" OR, AND e NOT, rispettivamente. (Or, and e not sono vocaboli inglesi che significano rispettivamente "o", "e" e "non".) In queste raffigurazioni (si veda qui sopra) si può immaginare il circuito come una "scatola nera" con uno o più ingressi (equivalenti agli interruttori del nostro esempio) e una uscita. Nelle effettive applicazioni elettroniche, poi, i due stati logici non sono rappre-



X	Y	X NAND Y
1	1	0
1	0	1
0	1	1
0	0	1



X	Y	X NOR Y
1	1	0
1	0	0
0	1	0
0	0	1

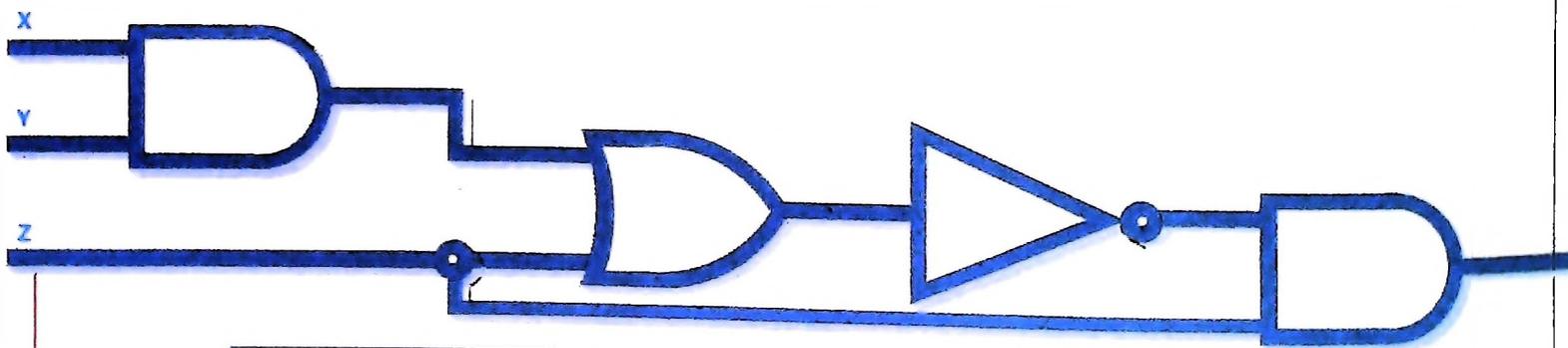
Nella pagina precedente, in alto, un interruttore elementare aperto e chiuso e la combinazione di due interruttori in parallelo e in serie. Sotto, la rappresentazione grafica delle "porte logiche" fondamentali OR, AND e NOT. Qui è data invece la rappresentazione grafica delle porte logiche NOR e NAND con la relativa tavola di verità. Le porte NOR e NAND hanno una particolarità: ciascuna di esse è sufficiente da sola per realizzare qualunque rete logica: le funzioni NOT, AND e OR possono essere definite in termini di NAND oppure di NOR. Con un solo tipo di porta, quindi, si può realizzare qualunque rete logica.

sentati necessariamente dal passaggio o meno di corrente, ma da due livelli di tensione distinti (che possono essere 0 e +5 oppure -5 e +5 volt, per esempio): per questo spesso i due stati logici 0 e 1 sono interpretati come "tensione bassa" e "tensione elevata".

Una combinazione qualsiasi di queste porte fondamentali costituisce una *rete logica*, il cui comportamento può essere completamente specificato da un'espressione booleana; e, viceversa, data un'espressione booleana, si può sempre realizzare una rete logica di cui quell'espressione descriva il comportamento. Nell'illustrazione in basso in questa pagina si può vedere una rete logica molto semplice, con la relativa espressione booleana e la corrispondente forma algebrica.

### Dalla tavola di verità alla rete logica

La corrispondenza fra le diverse "concretizzazioni" dell'algebra booleana astratta ci consente di usare l'una o l'altra secondo necessità o comodità e di trasferire i risultati raggiunti dall'una all'altra senza difficoltà di principio. Se non esistono problemi in teoria, in pratica però la cosa può non essere immediata. Per esempio, si può presentare la necessità di realizzare una rete logica corrispondente a una data tavola di verità: si è individuata, cioè, una funzione logica, e la si vuole "implementare" in un dispositivo fisico. Come si può fare? Dobbiamo risalire all'espressione booleana corrispondente e, per questo, notiamo che le combinazioni che ci interessano,



X	Y	Z	F
1	1	1	1
1	1	0	1
1	0	1	0
1	0	0	0
0	1	1	0
0	1	0	1
0	0	1	0
0	0	0	0

XYZ+  
XYZ+  
XYZ

$$(NOT ((X AND Y) OR Z)) AND Z$$

$$(XY + Z) - Z$$

Una rete logica e la sua descrizione, nella forma della logica degli enunciati e in quella algebrica. A sinistra, una tavola di verità e la relativa espressione booleana, come "somma di prodotti".

nella tavola di verità, sono quelle che danno 1 (il "vero") in uscita. Prendiamo la tavola di verità riportata in basso nella pagina precedente. Quello che ci dice è che la funzione logica F dà valore 1 se X, Y e Z sono 1, oppure se X e Y sono 1 e Z è 0, oppure, ancora, se Y è 1 e X e Z sono 0. Possiamo tradurre questa osservazione in una formula:

$$F = (X \text{ AND } Y \text{ AND } Z) \text{ OR } (X \text{ AND } Y \text{ AND NOT } Z) \text{ OR } (\text{NOT } X \text{ AND } Y \text{ AND NOT } Z)$$

o, per usare la notazione algebrica, più concisa in queste situazioni:

$$F = XYZ + XY\bar{Z} + \bar{X}Y\bar{Z}$$

Chi avesse dubbi sulla correttezza del procedimento può provare a calcolare la tavola di verità di questa formula e verificare che il risultato finale è identico all'ultima colonna della tavola di verità che definiva F. Le righe della tavola che

corrispondono ai valori 0 in uscita non portano alcun contributo alla formula.

### La minimizzazione

Con tavole di verità più ampie si possono ottenere formule lunghe e complesse, ma, come succede con le espressioni algebriche, queste formule potrebbero essere "semplificabili": potrebbero cioè esistere formule equivalenti più semplici.

Due formule booleane sono equivalenti quando hanno la stessa tavola di verità (e rappresentano quindi la stessa funzione logica) e fra due formule equivalenti sarà ovviamente preferibile la più semplice, perché può essere implementata con un minor numero di componenti (è un elementare principio di economia).

Minimizzare un'espressione booleana è quindi un passo importante per ottenere reti logiche efficienti e per questo sono stati studiati vari metodi, che consentono di rendere mecca-

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

	$\bar{B}$	B	$\bar{B}$	B
$\bar{A}$	A = 0 B = 0 C = 0	A = 0 B = 1 C = 1	A = 0 B = 0 C = 0	A = 0 B = 1 C = 1
A	A = 1 B = 0 C = 0	A = 1 B = 1 C = 1	A = 1 B = 0 C = 0	A = 1 B = 1 C = 1
	$\bar{C}$	C	$\bar{C}$	C

	C		C		
$\bar{A}$	A = 0 B = 0 C = 0 D = 0	A = 0 B = 0 C = 0 D = 1	A = 0 B = 0 C = 1 D = 1	A = 0 B = 0 C = 1 D = 0	$\bar{B}$
A	A = 0 B = 1 C = 0 D = 0	A = 0 B = 1 C = 0 D = 1	A = 0 B = 1 C = 1 D = 1	A = 0 B = 1 C = 1 D = 0	B
	$\bar{D}$	D	$\bar{D}$	D	

Le mappe di Karnaugh per la rappresentazione di espressioni con due, tre e quattro variabili. Ciascun quadratino rappresenta una possibile combinazione di valori delle variabili. Ciascun quadratino ha quindi lo stesso significato di una riga in una tavola di verità: data una espressione booleana, si dà valore 1 ai quadratini corrispondenti alle righe della sua tavola di verità per le quali l'espressione viene valutata "vera". Gli altri quadratini avranno valore 0 (oppure possono essere lasciati vuoti).

nico il procedimento. Si può procedere, in effetti, anche come si fa con le espressioni algebriche, usando per esempio le tautologie, che stabiliscono l'equivalenza di due espressioni di diversa complessità (le leggi di De Morgan, per esempio), ma è necessaria una buona conoscenza delle leggi logiche e, peraltro, non si può mai essere del tutto sicuri di aver raggiunto la formula più semplice possibile.

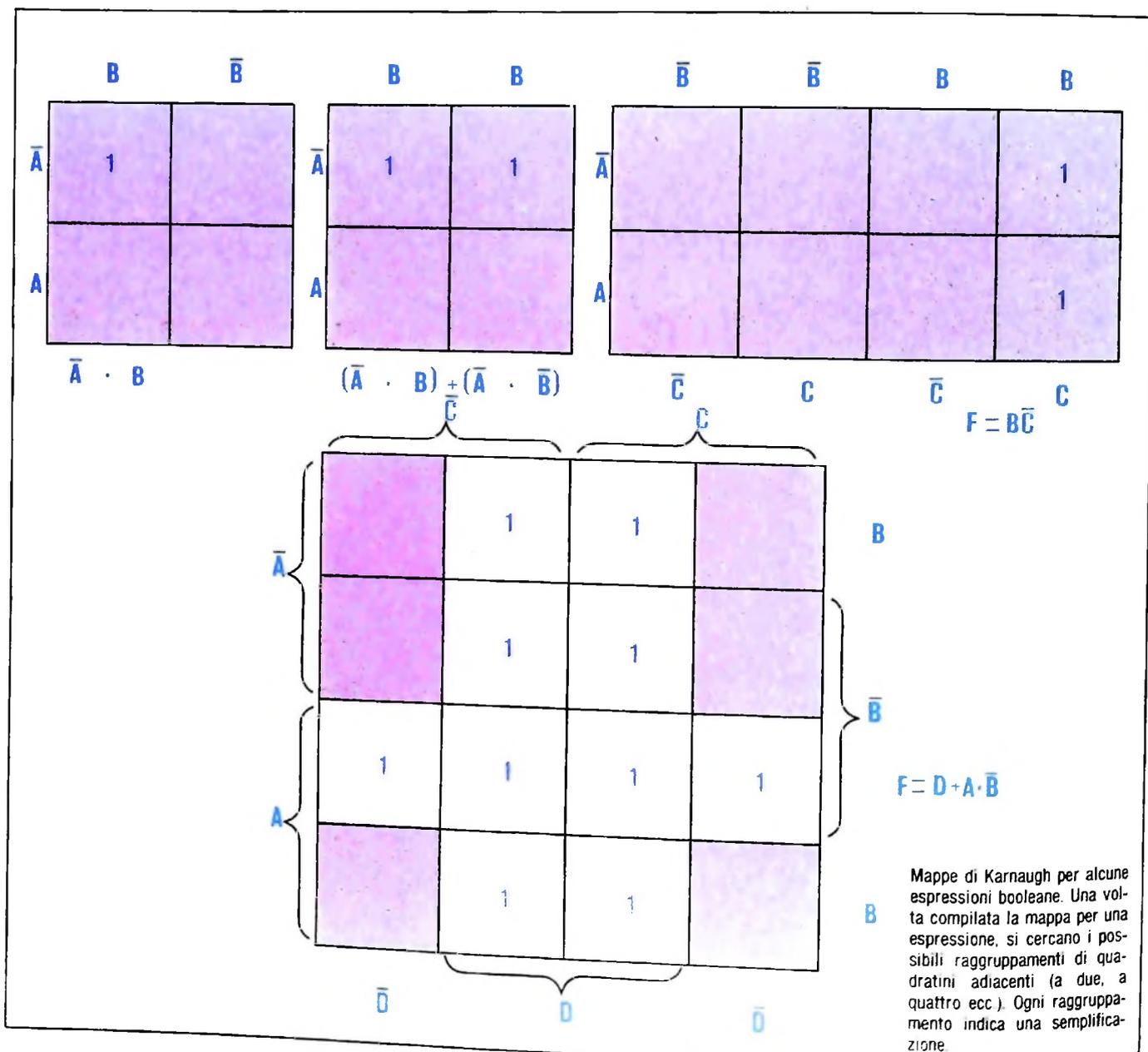
### Mappe di Karnaugh

Le mappe di Karnaugh costituiscono un metodo grafico per minimizzare espressioni booleane, che risulta maneggiabile per espressioni contenenti al massimo sei variabili distinte. Cominciamo con il caso più semplice, quello di due variabili: si costruisce una griglia di quattro caselle (si veda la figura nella pagina precedente), ciascuna delle quali rappresenta una delle possibili combinazioni di valori delle variabili A e B. Tutte le funzioni booleane di due variabili sono rappresentabili in questa griglia: si mette un 1 nelle caselle corrispondenti alle combinazioni di valori di A e B che rendono

vera la funzione considerata. Per comodità le colonne e le righe sono etichettate con le variabili e le loro negazioni: se c'è un 1 in una casella, le cifre 1 sono gli elementi semplici che etichettano la riga e la colonna al cui incrocio si trova quella casella. Pertanto, se in una casella c'è un 1, nella espressione booleana comparirà il prodotto logico di quei due elementi.

Dato uno di questi diagrammi, l'espressione corrispondente è la somma dei prodotti logici indicati dalle cifre 1.

Per procedere alla minimizzazione, bisogna raggruppare le caselle adiacenti a gruppi di due o di quattro. Le caselle che restano isolate non consentono semplificazioni, mentre quelle che possono essere raggruppate indicano possibilità di semplificazione (l'analogo di fattori che si elidono a vicenda). Così, per esempio, in questa pagina vediamo una mappa di Karnaugh per una espressione a tre variabili, in cui compaiono due regioni adiacenti contenenti 1: in questo caso la formula non minimizzata sarebbe  $ABC + \bar{A}BC$ : si può vedere come la funzione non dipenda in realtà da A, ma solo da B e da C, perché se B è 1 e C è 0 la funzione dà il valore 1, indipendentemente dal valore di A, mentre in ogni altro caso dà il valore 0.



Mappe di Karnaugh per alcune espressioni booleane. Una volta compilata la mappa per una espressione, si cercano i possibili raggruppamenti di quadratini adiacenti (a due, a quattro ecc.). Ogni raggruppamento indica una semplificazione.

## Lezione 21

**Programmi, testi e l'uso del comando MERGE**

Un calcolatore mette a disposizione in generale un certo insieme di strumenti per facilitare la costruzione di programmi che vanno dai TEXT EDITOR (cioè gli strumenti che permettono di costruire, correggere, modificare i programmi pensati come testi), ai COMPILATORI e INTERPRETI (che mettono a disposizione i linguaggi di programmazione), alla GESTIONE DELLE LIBRERIE (cioè una collezione di strumenti che permette di memorizzare programmi o sottoprogrammi, di richiamarli, di inserirli in altri, e così via, recuperando fatiche già fatte).

Anche l'M10 ha a disposizione, nel suo piccolo, qualche tipo di "gestione di libreria". Infatti, abbiamo imparato che, con il comando SAVE, possiamo registrare i nostri programmi dotandoli di un nome, che risulterà poi uno dei possibili nomi selezionabili con il menù.

**Programmi o testi di programmi?**

In realtà, esistono diverse forme del comando SAVE: accanto a quella che conosciamo, che permette di "salvare" il programma nella memoria dell'M10, ce ne sono altre che permettono di registrarlo su una cassetta magnetica, e altre ancora.

Tuttavia, anche per la registrazione del programma in memoria esistono due versioni differenti; supponiamo di voler costruire un piccolo sottoprogramma come il seguente:

```
List
1000 'Mette in M il maggiore tra X e Y
1010 IF X>Y THEN M=X ELSE M=Y
1020 RETURN
Ok
```

Possiamo salvare questo sottoprogramma con il solito comando:

```
save "max"
Ok
```

che inserisce il programma in memoria identificandolo con il suffisso .BA, come compare dal menù.

Tale suffisso indica che il modo con cui il programma è stato registrato non è quello con cui sarebbe stato registrato un testo qualunque costruito con il comando TEXT, tanto che tale programma è eseguibile (anzi, l'esecuzione è automaticamente inne-

scata dalla sua selezione a partire dal menù), ma non è operabile come un testo (per esempio spostandosi su di esso con i vari cursori).

È possibile, tuttavia, fare una delle seguenti operazioni:

- trasformare temporaneamente il programma in un testo, per poter effettuare correzioni veloci;
- registrare direttamente il programma come testo.

### **Il comando EDIT**

Proviamo a richiamare il nostro sottoprogramma MAX selezionandolo con il menù: l'esecuzione è innescata automaticamente, e, non essendo il sottoprogramma richiamato da nessuno, il messaggio

```
PRG Error in 1020  
Ok
```

viene notificato, in quanto non è possibile eseguire l'istruzione RETURN. Possiamo allora effettuare la lista, inserire nuove linee, cancellarne di vecchie, ma non possiamo "passeggiare" sul programma con il cursore. Proviamo però a dare il comando:

```
edit
```

e vedremo che, dopo una breve pausa in cui l'M10 trasforma la versione del programma in un testo di caratteri, ci viene visualizzato sullo schermo il nostro programma, proprio come se lo stessimo costruendo con la funzione TEXT: sono evidenziati tutti gli "a capo" (con il triangolo nero con la punta rivolta verso sinistra), è evidenziata la posizione del "cursore" (il rettangolo lampeggiante), possiamo scorrere il testo con le quattro frecce di posizionamento del cursore, possiamo fare qualunque modifica proprio come operando su un testo.

Il tasto F8 fa comparire un messaggio lampeggiante "WAIT", che chiede di aspettare un istante, in quanto è in atto la ritrasformazione del testo di caratteri in una forma di programma BASIC eseguibile, dopo di che tutto torna come prima.

Il comando EDIT permette di effettuare modifiche su un programma senza dover ritrascrivere ogni volta le righe da modificare, ma intervenendo localmente.

### **La registrazione del testo di un programma**

Visto che un programma ha questa ambivalenza di poter essere pensato come un algoritmo eseguibile o come un testo fatto di caratteri, è possibile chiederne la regi-

strazione direttamente in quest'ultima forma. Se infatti usiamo, per il salvataggio, il comando **SAVE** nella forma:

```
save "subm",A
Ok.
```

la registrazione in memoria avviene come testo.

Il carattere che compare nel comando è l'iniziale di ASCII, acronimo di American Standard Code for Information Interchange, ovvero il nome di uno standard di rappresentazione dei caratteri: dire di salvare con l'opzione A è chiedere la memorizzazione in **CARATTERI ASCII**, e non in forma di programma.

Infatti, tornando al menù, vediamo che il nostro sottoprogramma ha ricevuto il suffisso **.DO**, tipico dei testi.

Se lo selezioniamo, non avremo affatto l'innesco della solita esecuzione automatica, ma ci verrà semplicemente visualizzato il testo, proprio come per qualunque altro testo costruito con **TEXT**.

## Il comando **MERGE**

A cosa può servire avere un programma come testo? Per esempio, a costruirci una libreria per comporre programmi a partire da altri.

Consideriamo il seguente esempio; selezioniamo il comando **BASIC** e dopo esserci accertati che non ci sono programmi presenti, inseriamo il seguente:

```
10 INPUT "Inserire due valori";X,Y
20 GOSUB 1000
30 PRINT "Il maggiore e' ";M
40 END
```

Evidentemente, si tratta di un programma parzialmente specificato, che avrebbe bisogno del sottoprogramma che abbiamo precedentemente costruito, e che, come testo, abbiamo memorizzato con il nome di **SUBM**.

È possibile, con il comando **MERGE**, fare sì che tutte le istruzioni del sottoprogramma vengano accodate a quelle del nostro programma principale, ottenendo così il risultato voluto con un completo recupero del modulo precedentemente scritto e salvato:

```
merge "subm"
Ok.
```

Infatti, chiedendo la lista del programma, verificiamo:

```
10 INPUT "Inserire due valori";X,Y
20 GOSUB 1000
30 PRINT "Il maggiore e' ";M
40 END
1000 'Mette in M il maggiore tra X e Y
1010 IF X>Y THEN M=X ELSE M=Y
1020 RETURN
```

Eseguendo, otteniamo:

```
run
Inserire due valori? 3,5
Il maggiore e' 5
Ok.
```

In realtà il comando MERGE non effettua un accodamento, ma un mescolamento delle istruzioni dei due testi, rispettando l'ordine dei numeri di linea crescenti; se però noi abbiamo l'accortezza di costruire dei sottoprogrammi sempre a partire da numeri di linea alti (1000, 2000 ecc.), ecco che possiamo metterci a disposizione una vera e propria biblioteca di moduli che possiamo usare per comporre più facilmente nuovi programmi, con la garanzia che tali moduli non avranno istruzioni che si mescoleranno con quelle del nostro programma, ma ne resteranno da queste rigorosamente divise, in fondo al testo.

### Cosa abbiamo imparato

In questa lezione abbiamo visto:

- l'uso del comando EDIT
- la memorizzazione dei programmi con il comando SAVE con l'opzione A
- il comando MERGE
- l'uso del comando MERGE per il reimpiego di sottoprogrammi da considerare come moduli standard.

# IL COMPUTER IN BANCA

Il sistema bancario è uno degli esempi più concreti di ambiente in cui si è sviluppata l'applicazione delle nuove tecnologie informatiche.

L'informatica e la telematica, oltre a indurre cambiamenti nell'organizzazione del lavoro d'ufficio, hanno la possibilità di creare una situazione in cui la circolazione della carta verrà ridotta o addirittura eliminata, arrivando alla cosiddetta "paperless society". Un uso esteso di tecnologie elettroniche e di telecomunicazioni dovrebbe cioè condurre all'eliminazione della documentazione cartacea, sostituita dalla trasmissione del medesimo contenuto informativo entro reti di comunicazione, basate su tecnologie telematiche, colleganti un grande numero di elaboratori.

Anche l'oggetto "denaro" può essere visto come portatore di un particolare contenuto informativo e come tale trasmissibile al pari di qualsiasi altra informazione. A partire da queste considerazioni un possibile sviluppo della "paperless

society" va verso una società in cui gli scambi commerciali avvengono senza circolazione di carta moneta, la cosiddetta "cashless society". La moneta elettronica non rappresenterebbe una nuova forma di moneta, bensì una forma diversa di circolazione del denaro basata sull'uso di tecnologie elettroniche.

Una situazione in cui aumentano sia le interazioni tra l'utenza privata e la banca (i cosiddetti servizi di pagamento a domicilio delle utenze: bollette di luce, gas, telefono ecc.), sia quelle tra le aziende di credito, rende necessario l'uso di reti interbancarie di telecomunicazione nonché l'integrazione delle reti locali di elaborazione dei singoli istituti e la possibilità di accedere a banche-dati pubbliche o private (il servizio VIDEOTEL, per esempio). Da parte dell'utente il siste-

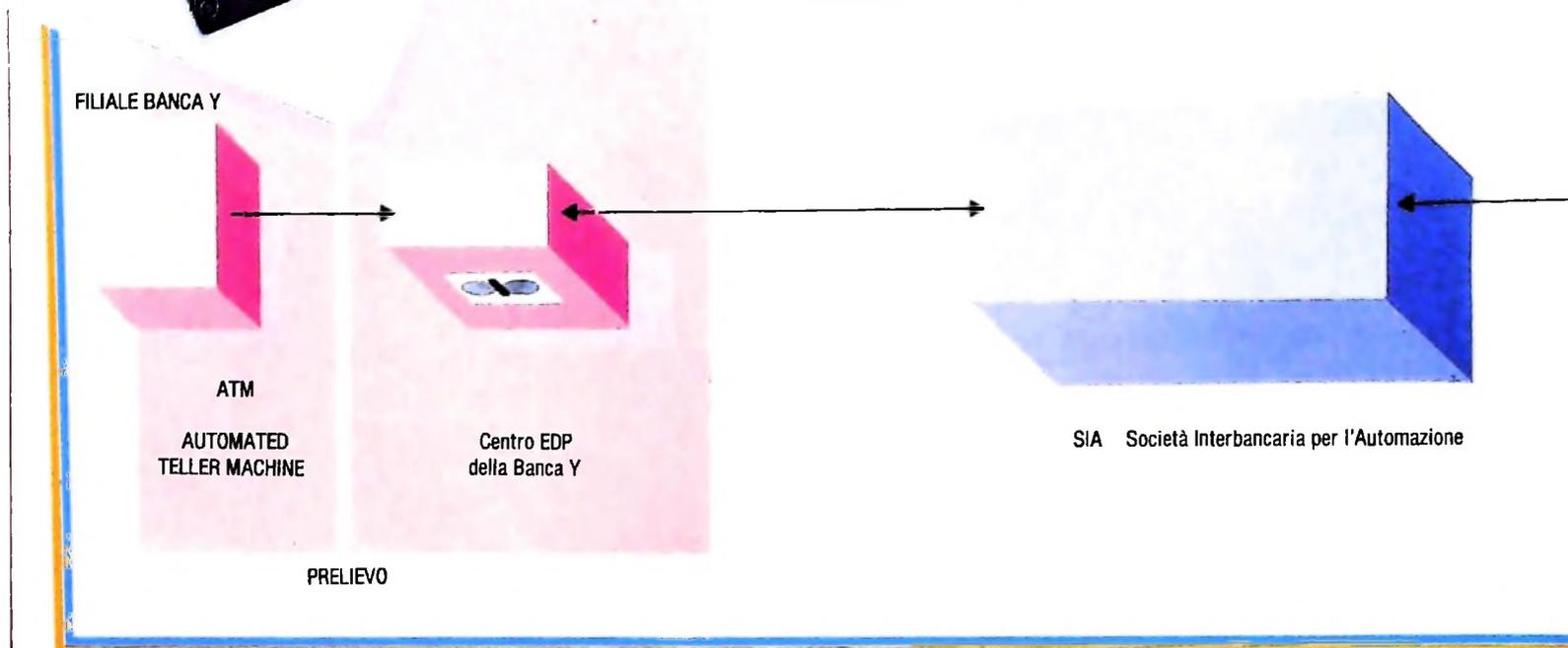


CAMBAZARD/OVERSEAS



**LE "AUTOMATED TELLER MACHINES", UN ESEMPIO DI TRASFERIMENTO ELETTRONICO DI FONDI**

Schema di interconnessione fra i sistemi distribuiti di istituti bancari che offrono servizi automatizzati all'utente.



ma banca viene sempre più visto come ente fornitore di servizi, e l'automazione di queste attività comporta l'abolizione parziale o totale del rapporto tradizionale cliente/impiegato. L'aspetto nuovo è costituito dal concetto di autotransazione, ovvero della possibilità da parte dell'utente di eseguire in proprio delle operazioni sotto la guida e il controllo di terminali molto sofisticati e in assenza di operatori specializzati.

**Sviluppo dell'automazione delle attività bancarie**

L'automazione delle attività del sistema banca deriva dalla necessità di fronteggiare una domanda di servizi costantemente in aumento in termini sia di numero di clienti che di numero di operazioni per cliente, a cui si è aggiunta anche l'esigenza di seguire o addirittura promuovere nuove forme di intermediazione e trattamento di grandi masse di denaro. Da un punto di vista più strutturale l'attività bancaria ha caratteristiche adatte a essere trattate automaticamente, quali:

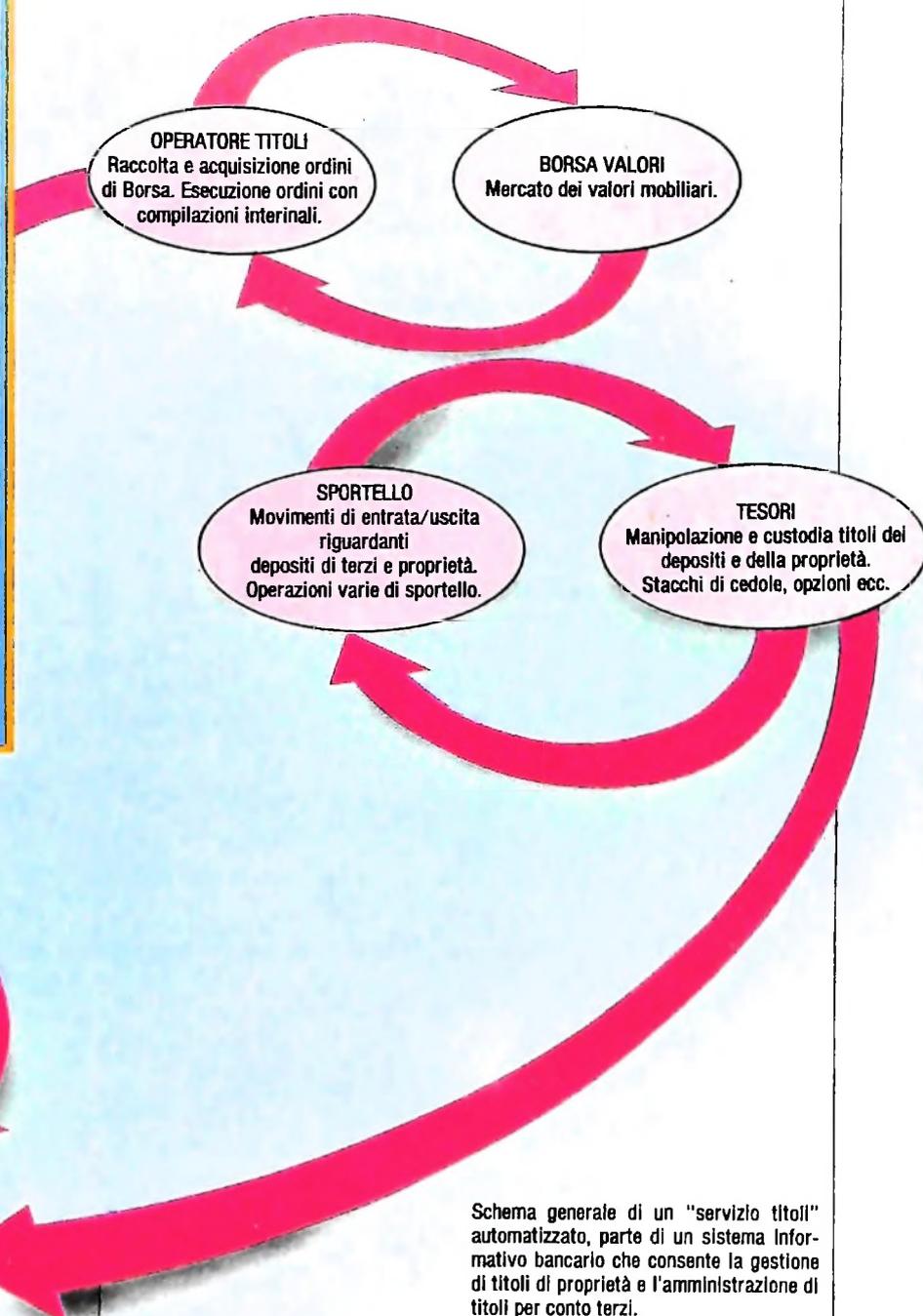
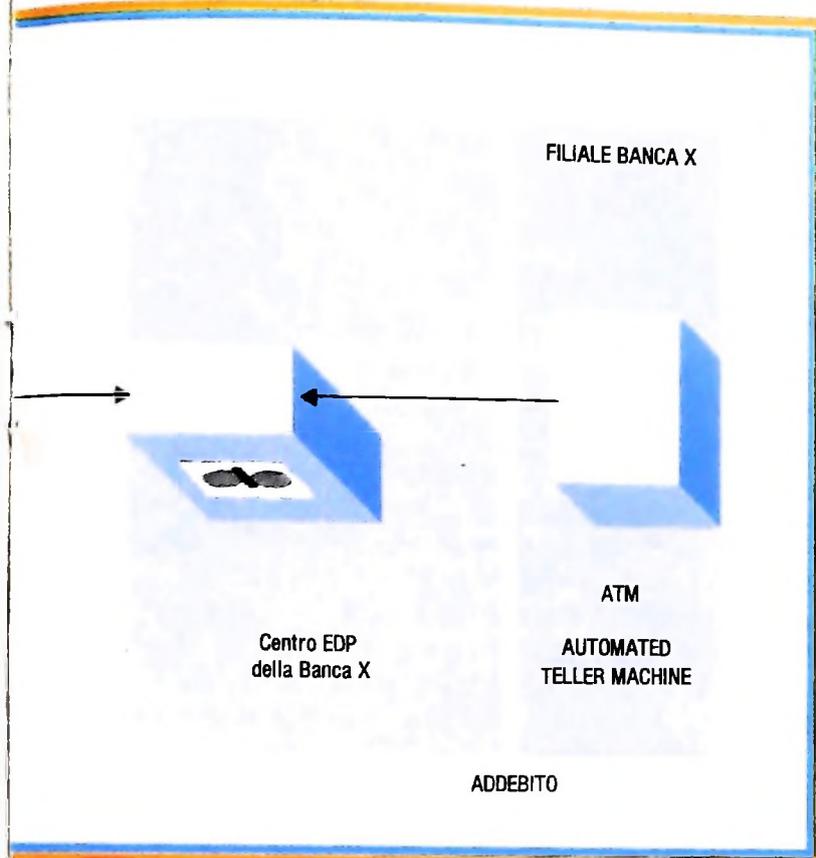
- ripetitività di talune operazioni che possono essere facilmente standardizzate;
- grossi volumi di calcolo;
- fornitura di servizi di genere diverso in termini di complessità e di costi;
- necessità di conservare e memorizzare grandi quantità di informazioni;
- necessità di accedere rapidamente alle informazioni e di poter stabilire relazioni tra le stesse.

Esiste quindi all'interno del sistema banca un intenso traffico di informazioni, spesso basato sullo scambio di una grande quantità di documentazione cartacea, dovuto alla tipica natura di sistema distribuito che caratterizza una banca do-

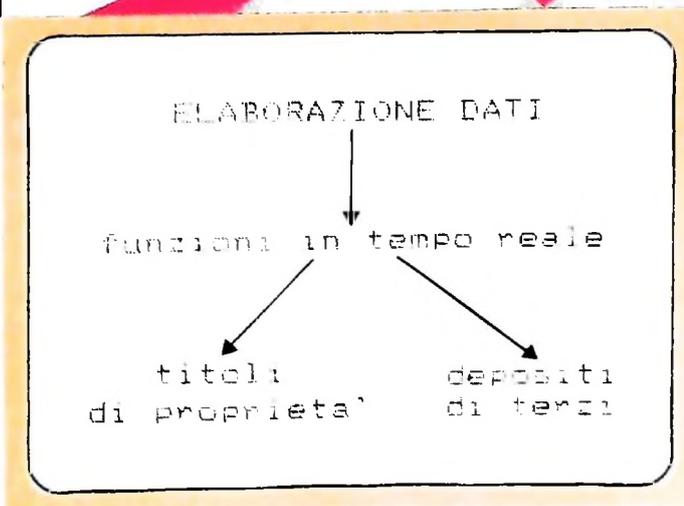
tata di filiali sparse sul territorio. L'automazione in banca si è sviluppata attraverso l'introduzione dei centri meccanografici che all'inizio erano dedicati alla gestione centralizzata di conti correnti e depositi a risparmio. Le attività principali di questi centri erano di contabilità, selezione, classificazione, e il supporto principale per la registrazione dei dati era la scheda perforata.

Dai primi centri meccanografici si è quindi passati all'introduzione di veri e propri calcolatori e le banche sono state tra i primi enti a utilizzare grandi centri di calcolo basati su elaboratori di grandi dimensioni, rappresentando una notevole parte dell'intero mercato del settore informatico. Quello bancario è stato quindi uno dei primi settori a utilizzare le possibilità offerte dalla rivoluzione microelettronica, essendo uno degli ambienti in cui sono stati applicati i concetti di informatica distribuita, di rete locale di calcolatori, di trasmissione dati in tempo reale. La conseguenza immediata è stata un miglioramento netto dei servizi prestati, che ha eliminato una quantità di passaggi intermedi molto costosi in termini di tempo. Un tipico esempio consiste nell'uso, ormai generalizzato, della banda magnetica presente sugli assegni, sulla quale viene registrato un particolare codice che permette di semplificare, per esempio, la seguente procedura:

— un assegno compilato da un utente di una Cassa di Risparmio, ricevuto come attestato di pagamento e quindi versato fuori piazza presso una Banca Popolare, deve raggiungere l'Istituto Centrale delle Banche Popolari, da qui essere trasmesso al corrispondente Istituto per le Casse di Risparmio e quindi tornare alla filiale che lo ha emesso per avere il riscontro finale della copertura del suddetto assegno. Tutte queste operazioni possono essere svolte automaticamente senza l'intervento diretto dell'operatore umano.



Schema generale di un "servizio titoli" automatizzato, parte di un sistema informativo bancario che consente la gestione di titoli di proprietà e l'amministrazione di titoli per conto terzi.



### Trasferimento elettronico dei fondi

Le ricerche che si sono sviluppate nel campo dei servizi bancari e interbancari hanno prodotto metodologie e tecniche generalmente raggruppate sotto il nome di "Electronic Fund Transfer Systems". Il trasferimento elettronico dei fondi va inteso come quel sistema elettronico in grado di sostituire, nei pagamenti tra due parti, un flusso di impulsi elettronici al flusso di contanti o di assegni.

Un esempio tipico di applicazione è costituito dalle cosiddette "Automated Teller Machines" (ATM) utilizzate per prelevare, depositare o trasferire fondi da un conto ad un altro. L'uso di questi terminali avviene fornendo l'utente di una particolare tessera a banda magnetica su cui sono registrate le informazioni relative all'utente stesso e allo stato corrente

## Sigle delle Società e Servizi Bancari per l'automazione in Italia

ABI	Associazione Bancaria Italiana
CIPA	Convenzione Interbancaria per i Problemi dell'Automazione (prima iniziativa nel settore)
SIA	Società Interbancaria per l'Automazione (svolge ruolo di coordinatrice del trattamento elettronico di incassi e pagamenti)
IPACRI	Istituto per l'Automazione delle Casse di Risparmio
STACRI	Sistema di Teletrasmissione Automatica delle Casse di Risparmio
SETIF	Sistema Elettronico per il Trasferimento Interbancario di Fondi (gestito da SIA)
CARISMAT	Rete di sportelli automatici delle Casse di Risparmio
BANCOMAT	Servizio nazionale interbancario di sportelli automatici

del suo rapporto con la banca. Inserita tale tessera nella ATM, all'utente viene richiesto di fornire la propria "password" (parola d'ordine) in modo da essere riconosciuto dal sistema e poter essere autorizzato ad eseguire una delle operazioni previste; questo passaggio è essenziale per poter proseguire nella procedura e un suo insuccesso provoca l'espulsione della tessera e in alcuni casi il sequestro della stessa, dopo alcuni tentativi infruttuosi, da parte della ATM.

Nel caso in cui invece la password venga riconosciuta, l'utente può proseguire, scegliendo il tipo di transazione desiderata mediante la selezione di uno dei tasti funzionali che la ATM presenta sulla tastiera, e quindi definire l'entità della transazione stessa attraverso i tasti numerici.

Le Automated Teller Machines sono sostanzialmente terminali, collegati ad un calcolatore centrale, in grado di:

— trasferire fondi da un conto a un altro nell'ambito di una stessa banca o tra banche diverse, aggiornando i saldi correnti dei rispettivi conti;

— accreditare, ovvero addebitare, fondi sul conto corrente della banca in cui è installata la particolare ATM che si sta utilizzando, o presso un'altra banca collegata.

Il collegamento dei terminali ATM con l'elaboratore centrale può avvenire secondo due modalità fondamentali che sono di tipo "on-line" e di tipo "off-line". In quest'ultimo caso il terminale ha capacità di elaborazione autonome e si collega periodicamente, per gli eventuali aggiornamenti, con l'elabo-

ratore centrale tramite un modem (modulatore/demodulatore), strumento che permette di utilizzare una linea telefonica per la trasmissione dei dati. In questo modo l'elaboratore centrale viene sollevato dal compito di dover gestire completamente il terminale, e diventa possibile la collocazione remota di questi terminali, per esempio presso un'azienda per il pagamento dei dipendenti. Uno dei problemi maggiori consiste nella necessità di dover alimentare con carta moneta il terminale, nel caso questo sia adibito alla distribuzione di denaro contante.

Nel caso invece di collegamento "on-line", l'elaboratore centrale esegue subito tutte le operazioni richieste e anche in questo caso il collegamento può essere realizzato tramite un modem, oppure collegando direttamente l'elaboratore al terminale nel caso questo sia collocato presso la banca.

Un particolare tipo di apparecchiatura per il trasferimento elettronico dei fondi è rappresentato dai cosiddetti "POS (Point Of Sale) terminals", "terminali punto di vendita", la cui diffusione su larga scala dovrebbe essere uno degli stadi intermedi per l'introduzione a livello di massa del denaro elettronico. L'idea consiste nel creare un sistema di terminali POS collegati agli elaboratori delle banche e distribuiti presso gli esercizi commerciali. Con meccanismi più o meno analoghi a quelli descritti per la ATM, utilizzando un'analogia tessera magnetica, verrebbero trasferiti fondi dal conto bancario del cliente a quello del commerciante. Anche in questo caso il terminale può essere collegato on-line e quindi a ogni operazione viene generata automaticamente la chiamata telefonica che permette di collegare via modem il terminale alla banca; oppure ciò avviene solo periodicamente per gli aggiornamenti, ad esempio di una "lista nera" delle password non abilitate che permetta di individuare eventuali scorrettezze da parte del portatore della tessera magnetica.

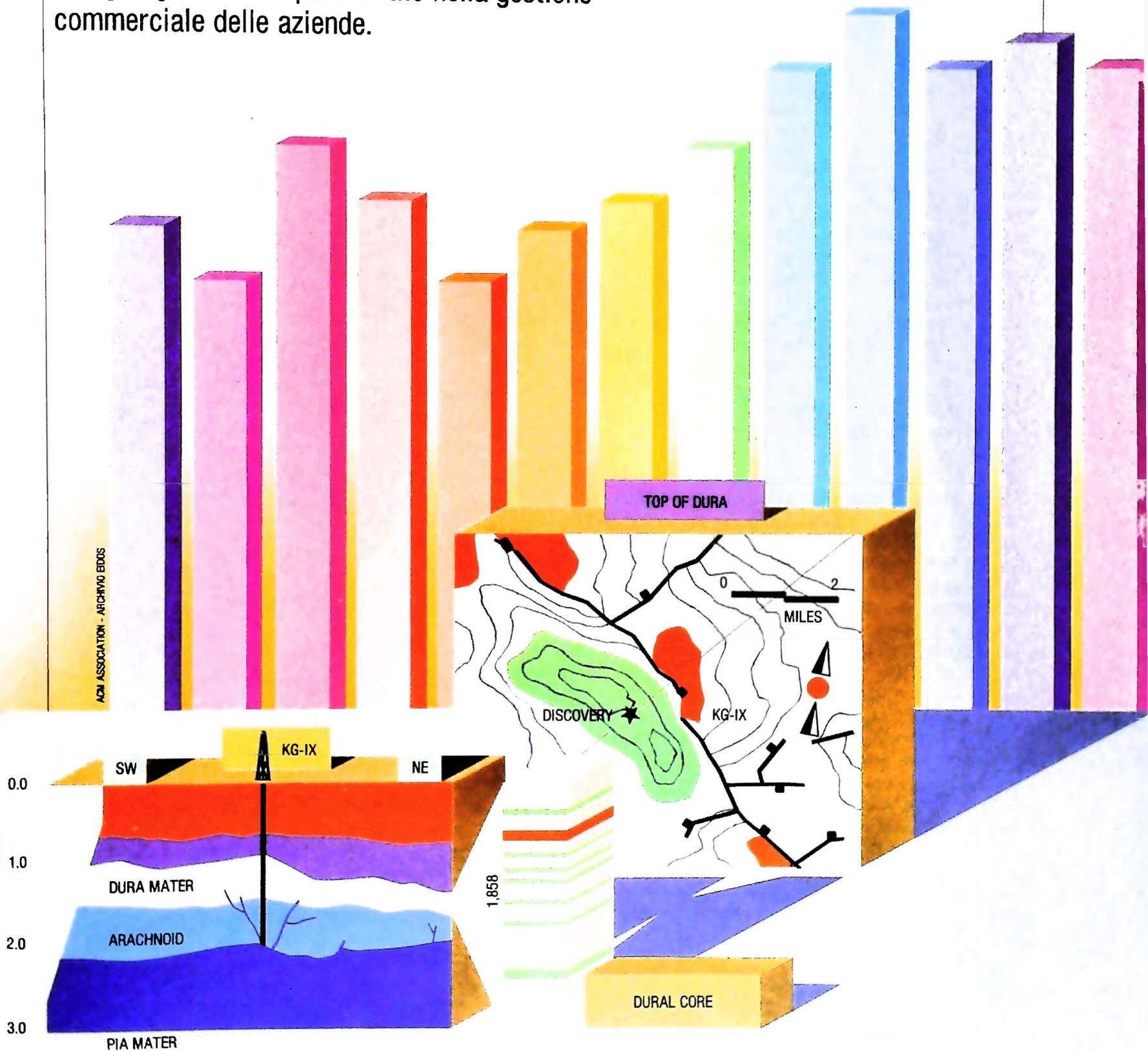
La realizzazione di questo tipo di servizi prevede il collegamento in rete di un grande numero di istituti bancari, come per esempio è avvenuto nel nostro paese con la realizzazione del sistema BANCOMAT. Questo sistema allo stato attuale permette il prelievo di banconote da parte di un cliente di una qualsiasi delle banche collegate alla rete BANCOMAT, presso un qualsiasi terminale installato in una qualunque filiale degli istituti associati al servizio BANCOMAT.

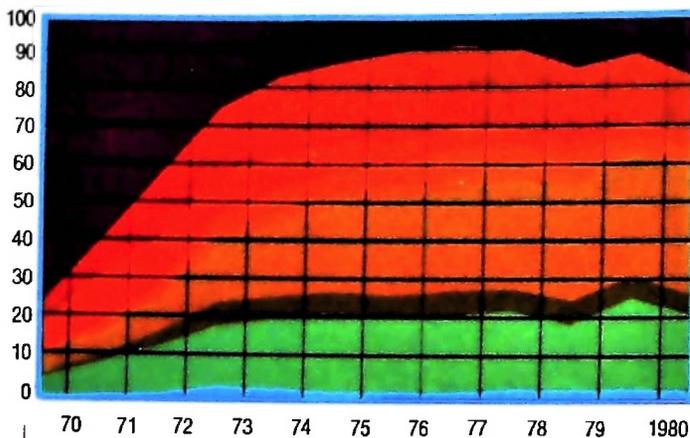
Un'altra rete bancaria è il sistema SWIFT (Society for Worldwide Interbank Financial Fund Transfer) realizzato principalmente per operazioni a livello interbancario internazionale e che collega le banche dell'Europa occidentale, del Canada e degli USA. Per operazioni in Europa che utilizzano questa rete internazionale la singola banca deve collegarsi al Concentratore Nazionale di messaggi che dialoga con il centro elaboratore dati dello SWIFT a Bruxelles (o in caso di guasto con quello di Amsterdam), che funziona da centro di smistamento e da cui il messaggio viene diramato alla banca estera destinataria, inviando inoltre al mittente la conferma di messaggio ricevuto.

In un prossimo capitolo vedremo alcune delle linee di tendenza dello sviluppo di nuovi tipi di servizi verso l'utente e dei problemi che si pongono sul piano giuridico, normativo e di sicurezza delle informazioni.

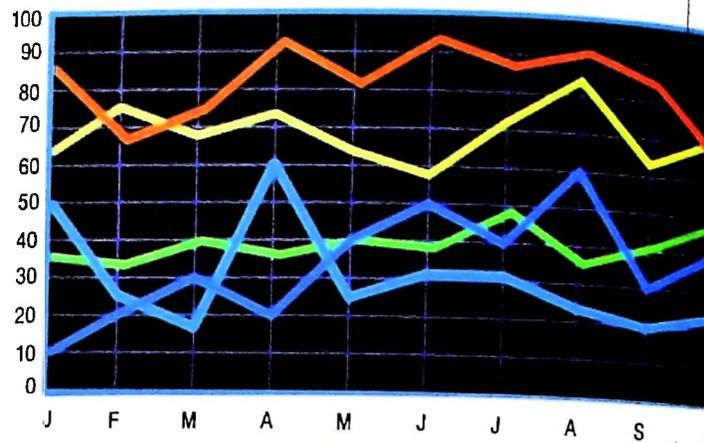
# BUSINESS GRAPHICS: I DIAGRAMMI A BARRE

Il segno grafico computerizzato nella gestione commerciale delle aziende.





AGM ASSOCIATION - ARCHIVIO EIDOS



“Business graphics” è un’espressione inglese, ormai entrata nell’uso comune, che definisce una particolare modalità d’impiego della grafica al computer.

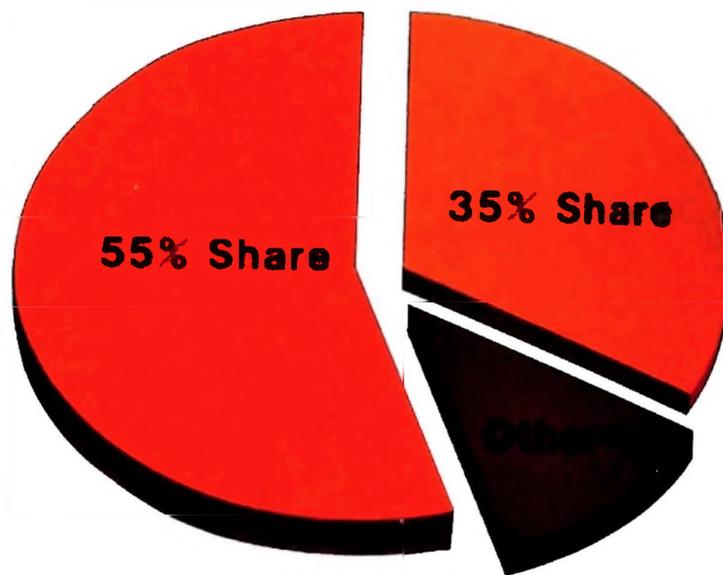
“Business” in inglese significa “affari”, “commercio”, quindi per “business graphics” si intende un particolare utilizzo del segno grafico computerizzato il cui obiettivo è quello di essere un efficace strumento di controllo per la gestione commerciale di un’azienda.

Tutti hanno visto almeno un diagramma in cui per esempio vengono rappresentate con barre colorate i dati relativi alle vendite di un certo prodotto mese per mese, oppure una circonferenza colorata a spicchi che simboleggia la globalità delle vendite, ove su ciascuno spicchio è specificata la percentuale che esso rappresenta rispetto all’intera circonferenza e che può corrispondere per esempio alle vendite ottenute in una certa regione o in un certo periodo.

Questi metodi di rappresentazione dei dati non sono stati introdotti dall’avvento del computer, ma esistevano già decine di anni fa. Nella gestione e nel controllo economico-commerciale di una azienda, vale infatti il famoso motto secondo il quale “un bel grafico a colori è più chiaro di mille parole o di un intero listato di numeri”. L’introduzione del computer ha semplicemente dato libero sfogo alla fantasia di rappresentazione dei dati: i grafici tridimensionali a barre, a torta, a scatola oppure a linee producono una percezione delle informazioni immediata e inequivocabile, generando così un linguaggio universalmente comprensibile. Il computer con le sue possibilità grafiche e cromatiche ha cioè ottimizzato le rappresentazioni e perfezionato le tecniche della business graphics, conferendo a questo settore della computergrafica una sua propria dignità metodologica.

Molto è stato scritto e detto sull’incredibile velocità di diffusione della business graphics al computer e sulle altrettanto strabilianti previsioni per il suo futuro. Utenti e venditori di tale prodotto, ossia manager, professionisti e commercianti da una parte e programmatori e costruttori di computer dall’altra, sono parimenti eccitati da tali prospettive.

Ma quale è la prerogativa che ha consentito alla business graphics di crearsi in breve tempo un così vasto mercato? Si potrebbe dire, adottando un’espressione economica, che essa è un’estensione del sistema informativo gestionale di un’a-



AGM ASSOCIATION - ARCHIVIO EIDOS

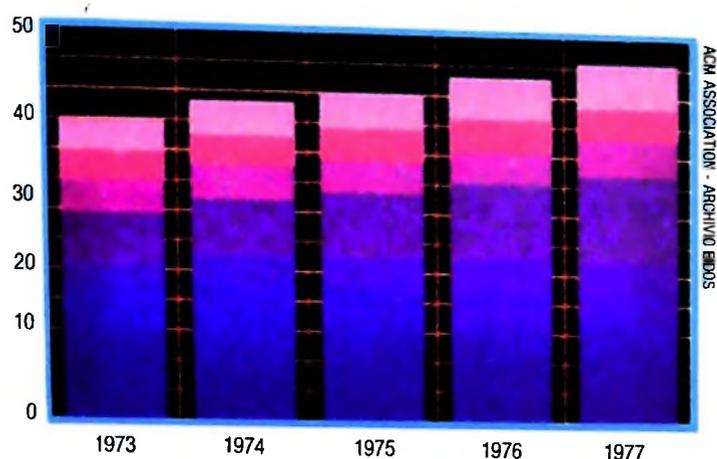
Nella pagina precedente, un’immagine in cui la fantasia del realizzatore ha avuto libero sfogo. La tridimensionalità, la simulazione di una sorgente luminosa e la realizzazione delle ombre esulano dalla necessità di rappresentazione dei

dati, ma conferiscono più chiarezza ai grafici presenti. In queste due pagine, vari tipi di grafici per la rappresentazione di dati. Il colore ha sempre un’importanza fondamentale nella percezione immediata delle informazioni.

zienda e come tale è diventata uno strumento in grado di offrire al manager in un solo colpo d’occhio l’andamento dell’intera azienda o di alcuni suoi settori.

Uno dei grandi vantaggi offerto dal computer è quello di poter confrontare contemporaneamente dati omogenei, ma riferiti a periodi di tempo diversi, di poterli elaborare ed evidenziarne i risultati. Tutto ciò può essere visualizzato su uno schermo definendo una serie di viewport, all’interno delle quali si rappresentano uno o più grafici da confrontare. L’utilità di questi sistemi è rafforzata dall’esistenza di speciali apparecchiature per riprodurre direttamente su pellicola i risultati visualizzati sullo schermo.

Ma la business graphics è ormai divenuta anche uno strumento per la gestione dei propri affari personali. Sempre più



numeroso sono le persone che sul proprio personal o home computer hanno qualche programma che consente loro, per esempio, di gestire le entrate e le uscite, di fare proiezioni sulle disponibilità finanziarie o di confrontare i redditi percepiti in vari anni.

Insomma, la business graphics sta entrando, ma forse sarebbe meglio dire che è già entrata, a far parte di quell'ampia gamma di strumenti di controllo che la tecnologia ci consente di inventare.

## I diagrammi a barre

Una curiosità che subito si manifesta è quella di veder funzionare un grafico per qualche applicazione personale immediata. Per esempio, ci può interessare vedere l'andamento dei nostri guadagni mese per mese, oppure confrontare le entrate mensili di quest'anno con quelle dell'anno precedente, o anche verificare gli scostamenti fra preventivi e consuntivi.

Questi problemi vengono risolti generalmente mediante l'utilizzazione dei cosiddetti "diagrammi a barre".

Forniamo qui un esempio di programma che accetta in input 12 coppie di dati e disegna un diagramma a barre mese per mese.

Ci sembra di poter dire che il programma è veramente semplice, per cui ci limiteremo a commentarne brevemente la modalità d'utilizzo.

Alla richiesta di input che compare dopo il disegno delle barre nere si risponde con un ENTER per ottenere la seconda serie di barre. L'ultima richiesta di input (istruzione 340) serve per non far cancellare automaticamente il disegno dalla istruzione di END. Per terminare si batterà ENTER.

In una prossima lezione vedremo invece un esempio di diagramma "a torta", come vengono comunemente chiamati i diagrammi costituiti da circonferenze colorate a spicchi.

Lo scopo dei diagrammi a torte e a barre è, ripetiamo, essenzialmente quello di offrire una visione d'insieme dell'andamento di una qualsiasi attività: sia essa di vendita, di produzione o di qualsiasi altro tipo, purché quantificabile in numeri che andranno a determinare l'ampiezza degli spicchi di torta o l'altezza delle barre.

```

5  REM QUESTO PROGRAMMA DISEGNA UN
6  REM GRAFICO A BARRE E PERMETTE DI
7  REM CONFRONTARE DUE SERIE DI DATI
8  REM OMOGENEI (PER ES. RIFERITI AI
9  REM MESI DEL 1983 E DEL 1984)
10 CLS
20 DIM Y(12,2)
25 INPUT "VUOI USARE I DATI GIA'
    ESISTENTI (S/N)";R$
30 IF R$="S" THEN 82
33 REM INSERIMENTO DATI MESE PER MESE
35 FOR I=1 TO 12
40 PRINT "MESE";I
50 INPUT "Q.TA'(0-53),Q1,Q2";Q1,Q2
55 IF Q1<54 AND Q2<54 GOTO 70
60 PRINT "Q.TA' ECCESSIVA"
65 GOTO 40
70 Y(I,1)=Q1 : Y(I,2)=Q2
77 NEXT I
78 GOTO 108
79 REM ASSEGNAZIONE AUTOMATICO DEI DATI
80 REM MENSILI PER DIMOSTRAZIONE
82 Y(1,1)=10 : Y(1,2)=15 : Y(2,1)=40
85 Y(2,2)=39 : Y(3,1)=50 : Y(3,2)=45
88 Y(4,1)=38 : Y(4,2)=51 : Y(5,1)=49
91 Y(5,2)=37 : Y(6,1)=23 : Y(6,2)=40
94 Y(7,1)=29 : Y(7,2)=26 : Y(8,1)=39
97 Y(8,2)=34 : Y(9,1)=52 : Y(9,2)=40
100 Y(10,1)=47 : Y(10,2)=30 : Y(11,1)=38
103 Y(11,2)=40 : Y(12,1)=27 : Y(12,2)=30
108 CLS
110 REM DISEGNO ASSI DI RIFERIMENTO
113 LINE (18,0)-(18,55),1
115 LINE (18,63)-(235,63),1
120 PRINT TAB(4) "GE FE MA AP MA GI";
130 PRINT " LU AG SE OT NO DI"
140 PRINT "54"
150 PRINT
160 PRINT "38"
170 PRINT
180 PRINT "22"
181 LINE (17,8)-(19,8)
182 LINE (17,24)-(19,24)
183 LINE (17,40)-(19,40)
185 REM STAMPA DEL GRAFICO RELATIVO AI
186 REM DATI Q1.
190 K=0
200 FOR I=23 TO 221 STEP 18
210 K=K+1
215 IF Y(K,1)=0 THEN 250
220 X=I : Y=62-Y(K,1)
240 LINE (X,Y)-(X+5,62),1,BF
250 NEXT I
255 REM STAMPA DEL GRAFICO RELATIVO AI
256 REM DATI Q2
260 INPUT
270 K=0
280 FOR I=29 TO 227 STEP 18
290 K=K+1
295 IF Y(K,2)=0 THEN 330
300 X=I : Y=62-Y(K,2)
320 LINE (X,Y)-(X+6,63),1,B
330 NEXT I
340 INPUT
350 END

```

## Medicina e informatica

Medicina e informatica è un binomio frequentemente utilizzato. L'aiuto alla difesa della salute che l'informatica è in grado di dare non si limita alla gestione dei dati dei pazienti, dei letti degli ospedali e al controllo di apparecchiature sofisticate. Nel campo della grafica e della elaborazione di immagini si stanno ottenendo risultati straordinari. Le ricerche in questo settore sono particolarmente avanzate negli Stati Uniti e in Giappone. Si tratta di ricerche finalizzate alla ricostruzione della figura umana, sia nella sua forma esteriore che nelle sue componenti anatomiche.

Ricerche nate in parte per lo studio ergonomico relativo alle capsule spaziali e in parte per meglio utilizzare e analizzare le informazioni provenienti da apparecchiature sofisticate quali il TAC (tomografia assiale computerizzata) vengono oggi indirizzate alla creazione di "contesti" utili al medico per lo studio di problemi chirurgici, per la simulazione di interventi, per il "monitoraggio" di pazienti sotto terapie sia chimiche sia fisiche.

Per poter fare ciò il metodo che si adotta consiste nel ricostruire la forma tridimensionale di una parte anatomica, ad esempio il cranio, e nel visualizzarlo su un display. L'informazione può provenire o dai dati di un tomografo assiale, nel qual caso si hanno dati sulla presenza di tessuti anomali, oppure può essere il prodotto di elaborazioni geometriche condotte su una forma ideale tipo.

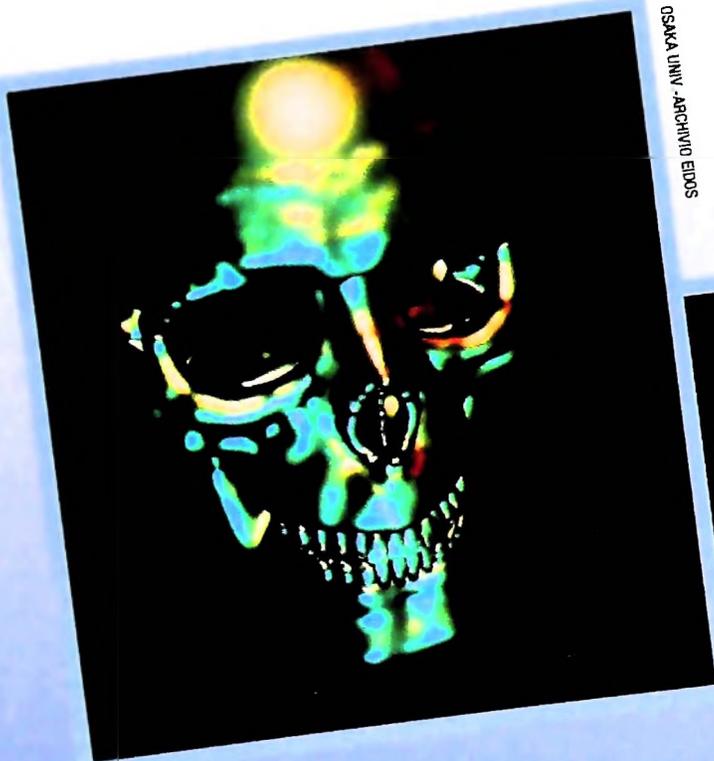
Nel primo caso si possono effettuare infinite analisi sull'immagine presentata dall'elaboratore, estraendo dati significativi, quali le dimensioni e l'estensione della zona sospetta, controllare nel tempo l'evoluzione della terapia in

modo automatico, identificare con maggiore precisione la presenza di tessuti anomali in modo semiautomatico. Tutti questi possono essere indubbi aiuti sia nella diagnosi sia nella terapia.

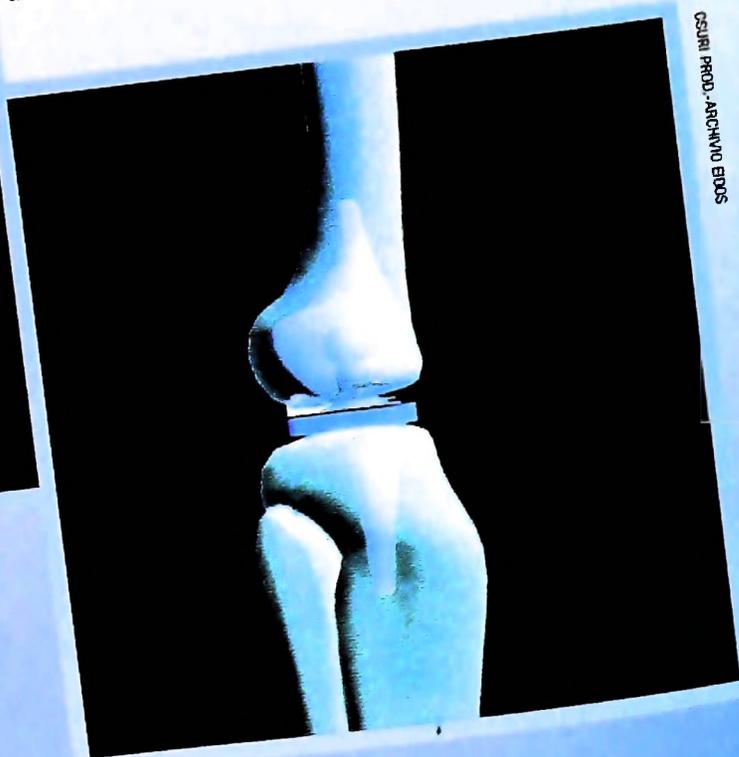
Studi particolarmente interessanti riguardano la simulazione di interventi chirurgici: si tratta in questo caso di pianificare un intervento su un paziente inesistente, a partire dal modello geometrico delle parti interessate, integrato con dati eventualmente rilevati ancora con TAC o altre tecniche. Il medico, interagendo con l'elaboratore, indica il percorso degli strumenti chirurgici, identifica i punti rischiosi, addestra se stesso e i propri collaboratori a seguire le varie fasi dell'intervento in modo accurato e sincronizzato, senza alcun danno al paziente o a cavie vive.

I vantaggi di queste procedure sono indiscutibili e si basano, come ormai abbiamo potuto vedere in molte altre applicazioni, sulla capacità dell'elaboratore di essere strumento di simulazione del reale. In questi casi la simulazione diventa sempre più perfezionata, dando al medico la percezione visiva di ciò che potrebbe accadere durante il suo operato.

Altri campi di ricerca particolarmente sofisticati sono quelli microbiologico e genetico. In questi settori il computer ha immesso la sua enorme potenzialità di simulazione del reale, che consente di ricostruire e di studiare le molecole più complesse come il DNA. L'ingegneria genetica, una fra le più recenti e più affascinanti branche della biologia, ha ricevuto un impulso fondamentale nel suo sviluppo dalla possibilità di utilizzare le capacità grafiche messe a disposizione dalla tecnologia del computer.



OSAKA UNIV. - ARCHIVIO EIDOS



OSAKA UNIV. - ARCHIVIO EIDOS

Ricostruzione tridimensionale di un cranio e modellazione geometrica tridimensionale per lo studio di una protesi al menisco. L'immagine del cranio è stata costruita al computer pixel per pixel, rilevando direttamente dal cranio del paziente, mediante speciali sensori, le "quote" delle coordinate spaziali. Simulazioni come quella a destra permettono di valutare con estrema precisione la conformazione di una protesi.

— UN NUOVO MODO DI USARE LA BANCA. —

# Conto corrente più

## TANTI PENSIERI IN MENO CON IL CONTO CORRENTE "PIU" DEL BANCO DI ROMA.

Essere cliente del Banco di Roma vuol dire anche essere titolari del conto corrente "più". Un conto corrente più rapido: perché già nella maggior parte delle nostre filiali trovate gli operatori di sportello che vi evitano le doppie file.

Più comodo, perché potete delegare a noi tutti i vostri pagamenti ricorrenti: dai mutui all'affitto, dalle utenze alle imposte.

Più pratico, perché consente l'utilizzo del sistema di prelievo automatico Bancomat e l'ottenimento della carta di credito.

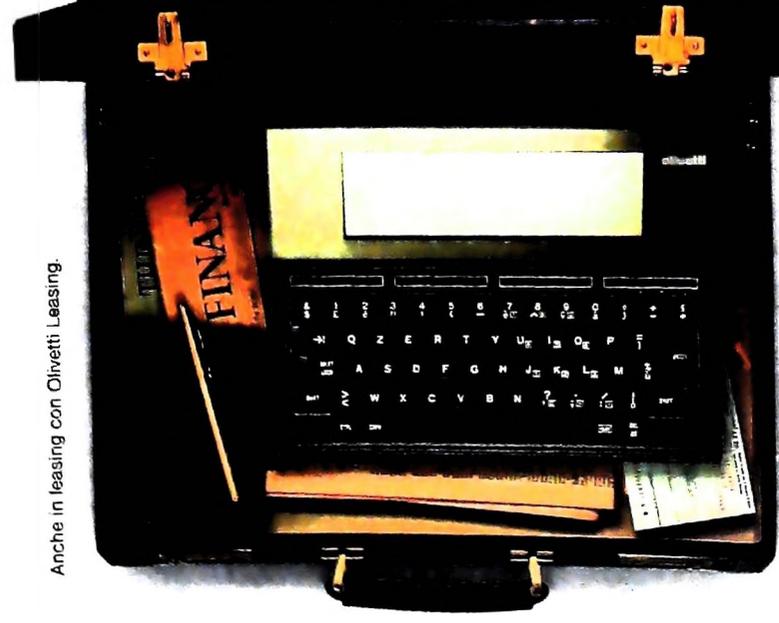
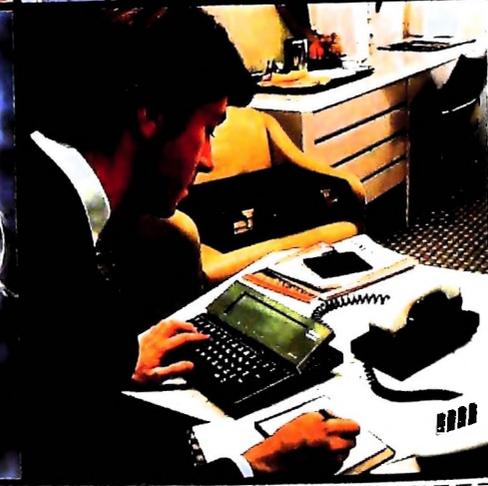
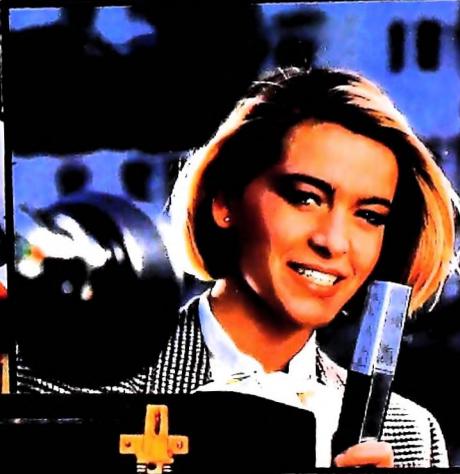
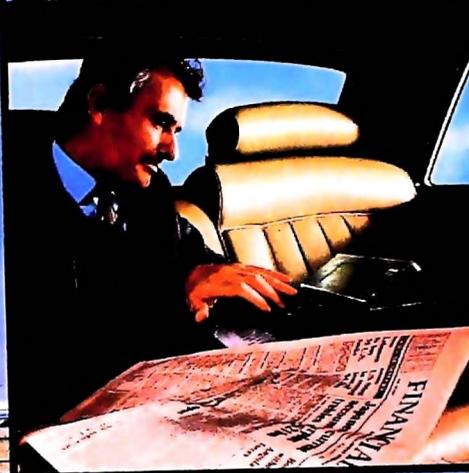
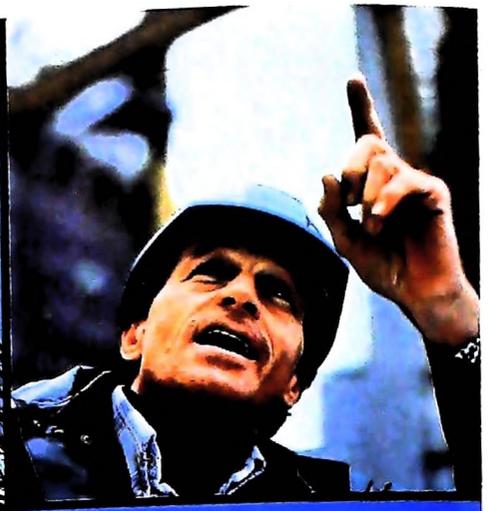
Più esclusivo, perché potete usufruire del servizio Voxintesi, attraverso il quale chiedere direttamente al nostro elaboratore il saldo del vostro conto corrente con una semplice telefonata: in qualsiasi ora come in qualsiasi giorno, anche festivo.

Più sicuro, perché con una minima spesa potrete assicurarvi contro furti e scippi mentre vi recate in banca o ne uscite.

Veniteci a trovare, ci conosceremo meglio.

 **BANCO DI ROMA**  
CONOSCIAMOCI MEGLIO.





Anche in leasing con Olivetti Leasing.

## PERSONAL COMPUTER OLIVETTI M10 L'UFFICIO DA VIAGGIO

Olivetti M10 vuol dire disporre del proprio ufficio in una ventiquattre. Perché M10 non solo produce, elabora, stampa e memorizza dati, testi e disegni, ma è anche capace di collegarsi via telefono per spedire o ricevere informazioni.

Qualunque professione sia la vostra, M10 è in grado, dovunque vi troviate, di offrirvi delle capacità di soluzione davvero molto grandi. M10: il più piccolo di una grande famiglia di personal.

# olivetti

Per informazioni e prezzi al negozio contrassegnato da "Olivetti M10 Punto di Vendita" o  
inviare l'acomposta Olivetti, Div. di Personal Computer, Via Meravigli 12, 20123 Milano

NUMERO VERDE  
VIA N  
CAMPUS  
TELEFONO