IL MIO COMPUTER

CORSO PRATICO DI INFORMATICA: MICRO, MINI E PERSONAL COMPUTER



ISTITUTO GEOGRAFICO DE AGOSTINI

IL MIO COMPUTER

CORSO PRATICO DI INFORMATICA: MICRO, MINI E PERSONAL COMPUTER

volume primo

ISTITUTO GEOGRAFICO DE AGOSTINI

Direzione: Achille Boroli - Adolfo Boroli
Direzione editoriale: Mario Nilo; *settore fascicoli:* Jason Vella

Redazione dell'edizione italiana a cura di:
Edizioni Remo Sandron - Firenze

Redazione dell'edizione inglese:
Richard Pawson (*Editor*); David Tebbutt (*Consultant Editor*); Chris Cooper (*Executive Editor*); David Whelan (*Art Director*); Gareth Jefferson, Roger Ford (*Collaborators*)
Designed and Produced by Bunch Partworks Limited
Traduzione a cura di: Daniel Quinn

Coordinamento grafico: Otello Geddo
Coordinamento fotografico a cura del Centro Iconografico dell'Istituto Geografico De Agostini

Titolo originale dell'opera: The home computer course
© Orbis Publishing Limited, Londra, 1983.
© Istituto Geografico De Agostini S.p.A., Novara, 1984.
Stampato in Italia - IGDA Officine Grafiche, Novara, 1984.

IL MIO
COMPUTER

Il computer, fino a ieri strumento di lavoro di una ristretta cerchia di esperti, è oggi entrato negli uffici, nelle fabbriche, nelle scuole e perfino nelle case. Molti sono i possessori di personal computer che apprezzano il loro apparecchio per la soddisfazione che dà e per il fascino che esercita; se costoro già sono consapevoli dei vantaggi di una sua conoscenza sistematica, numerose altre persone hanno invece validissime ragioni, a volte imposte dal bisogno, di acquisire i principi fondamentali del computing.
Si sostiene spesso che non è necessario sapere come funziona un computer, esattamente come non si richiedono nozioni di ingegneria meccanica per la guida di un'automobile; ma non c'è dubbio che una conoscenza sufficientemente ampia di questa materia è basilare già oggi e lo sarà ancora di più in un futuro molto vicino, in qualsiasi società evoluta: sgombera innanzitutto il campo da quel timore che deriva dal non sapere e che non permette di distinguere il possibile dall'impossibile.
In ogni caso è certo che, da un punto di vista molto pratico e immediato, il livello di vita dell'umanità e di ciascuno di noi dipende in larga misura dai vantaggi offerti dal computer.
La sua enorme diffusione a tutti i livelli non pone alternative: occorre aggiornarsi e imparare a dialogare con questi apparecchi.
IL MIO COMPUTER *è un corso pratico che gradualmente insegna a comunicare con il computer, a scrivere programmi, a conoscere i principi del suo funzionamento; aiuta a orientarsi nella fitta giungla del software, a valutare le prestazioni di questo o di quel microcomputer; spiega i termini tipici del "gergo" e le differenze tra i linguaggi di programmazione più diffusi.*
Di pari passo, esamina di volta in volta gli enormi cambiamenti apportati dall'informatica all'ambiente in cui viviamo e le prospettive per le future applicazioni del computer.
Il corso completo è svolto su due livelli: il primo, particolarmente adatto ai principianti, affronta temi di carattere più generale, offrendo un ampio panorama della materia, oltre all'insegnamento del linguaggio BASIC. Il secondo livello, che riguarda la teoria della programmazione in genere, il "linguaggio macchina" e gli altri linguaggi disponibili sugli apparecchi oggi più diffusi, mette in grado, attraverso moltissimi esempi concreti, di scrivere programmi nel linguaggio più adatto.
IL MIO COMPUTER *è, quindi, una guida chiara, semplice e completa al nuovo e affascinante mondo dell'informatica.*

Sommario

1 Cos'è un computer
5 Guida all'acquisto di un microcomputer
6 I più diffusi in Europa
8 Installare il micro
9 Computer e software
13 Lo sapevate che...?
14 Il mondo dei giochi
16 Carta d'identità
17 Lo Spectrum
20 Obbedire... e basta!
24 La nuova rivoluzione
26 L'artista elettronico
28 Bit e byte
30 L'insegnante elettronico
33 Il Commodore 64
36 Il tasto giusto
38 Gira e rigira
41 La controrivoluzione
44 Dipingere coi numeri
46 Lo scatto che conta
48 Lo sapevate che...?
49 Dietro le... barre
50 L'Oric-1
52 I puntini sulle i
54 Quando 1 + 1 = 10
56 Velocità di risposta
58 Memoria di ferro
60 Pronta cassa
61 La parola giusta
64 Lo sapevate che...?
65 I micro in movimento
66 Messaggio ricevuto!
68 Vero o falso?
70 Il Lynx
72 Appuntamento col chip
74 Via con la stampa

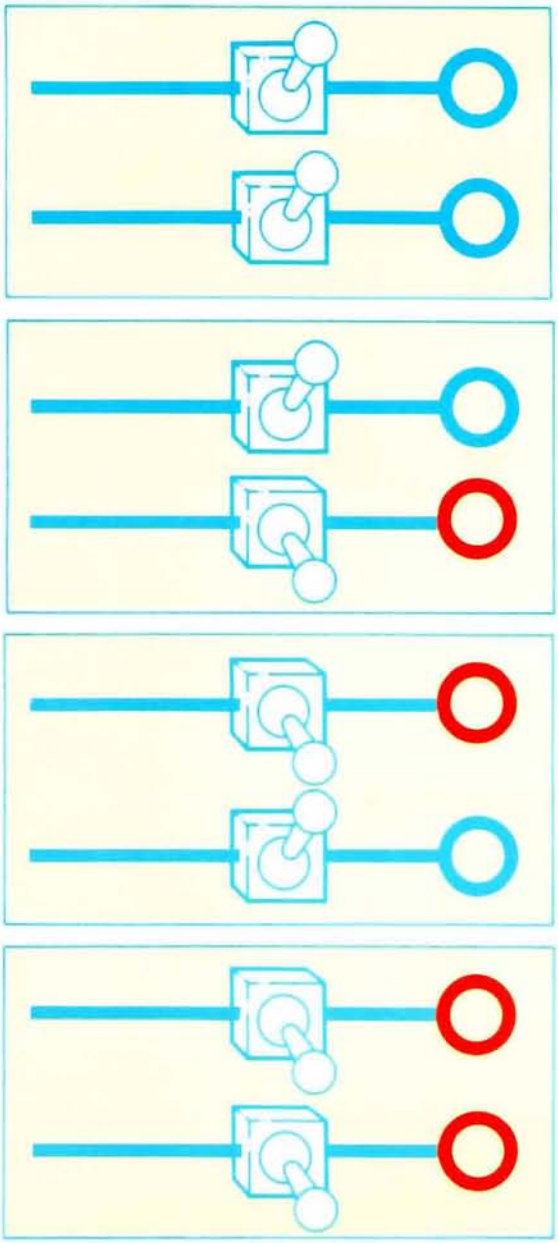
77 Parliamo di routine
79 Quando meno vale più
81 Obiettivo Scuola
84 Codice segreto
86 Dal pallottoliere al computer
89 Il BBC modello B
92 Le porte logiche ed i sommatori
94 La registrazione dei programmi
96 I tipi di memoria
98 Quanto manca a Natale?
101 Le carriere nell'informatica
104 I diagrammi di flusso
106 Scienza domestica
108 L'anello di congiunzione
109 Atari 400 e Atari 800
112 Il dialogo digitale
114 Le unità a dischi
116 Le variabili indicizzate
119 Il "complemento a due"
120 Sir Clive Sinclair
121 La fabbricazione dei chip
124 Database: chi cerca... trova
126 I micro e la medicina
128 Le leggi del pensiero
130 Il Dragon 32
132 Monitor o televisore?
134 La struttura di un programma
138 Il cervello di un computer
140 John von Neumann
141 Musica e computer
144 La CPU e gli indirizzi
146 Le funzioni intrinseche
150 Il Commodore VIC-20
152 La grafica "a sprite"
155 Steve Wozniak
156 Le penne ottiche
158 Gli "spreadsheet"
161 I giochi d'avventura
164 I "micromondi"
166 L'era dei portatili
169 L'Epson HX-20
172 Un numero... a caso
176 Una penna... a motore
179 Il codice esadecimale
180 Chuck Peddle
181 Immagini animate
184 Interpreti e compilatori
186 Parola al... computer
188 Peek e Poke
189 Il TI99/4A
192 I pirati del software
194 Le variabili a due dimensioni
198 I plotter
200 Alan Turing
201 Pronti al decollo?
204 Organizzazione dei dati
206 Le interfacce
209 Affidarsi al caso
210 Lo ZX81
212 Facciamo il punto
216 Gli accoppiatori acustici
218 I network
220 Charles Babbage
221 Avversario: il computer
224 I "Floppy tape"
226 Le qualità che contano
230 L'Aquarius
232 Una rubrica telefonica... computerizzata
236 Cosa sono i "buffer"?
238 I computer analogici
240 Herman Hollerith

Cos'è un computer?

I computer pensano e possiedono una conoscenza? La risposta è di vitale importanza per una vera comprensione di questi apparecchi



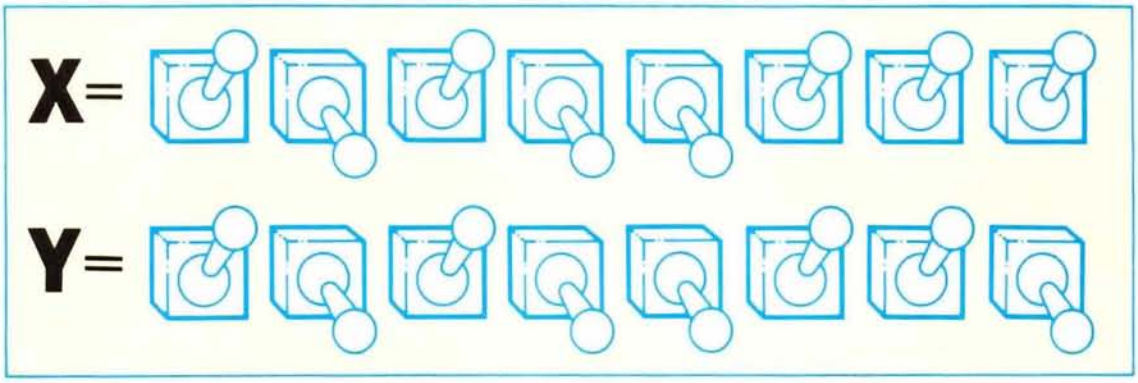
Rispondere alla domanda «cos'è un computer?» non è così facile come rispondere a «cos'è un televisore?» o a «cos'è una lavatrice?», poiché un computer — diversamente da altri apparecchi elettrici — non ha un'unica funzione. I computer digitali, anche quelli di bassissimo costo, costituiscono una nuova generazione di apparecchi che possono avere un'infinità di applicazioni a seconda delle istruzioni (programmi) loro impartite.
Il concetto di "programmabilità" non è del tutto estraneo all'ambiente domestico: basti pensare alle lavatrici e a certi tipi di cucine, che offrono la possibilità di scegliere tra diverse alternative di lavoro. Queste alternative sono veri e propri "programmi" predeterminati dal fabbricante. Nel caso dei computer, però, è possibile variare il programma a piacimento, adattandolo così ai più disparati impieghi: elaborazione dei testi, divertimento e giochi, gestione di contabilità e così via.
Com'è possibile che un computer possa svolgere compiti così diversi? È ciò che impareremo col progredire del corso; per adesso limi-



Interruttori e numeri

I computer usano circuiti elettrici per rappresentare i numeri. I circuiti consistono principalmente in interruttori. Un interruttore può essere acceso o spento. Due interruttori offrono 4 combinazioni di acceso/spento. La combinazione spento/spento equivale a 00, spento/acceso è 01, acceso/spento è 02, acceso/acceso è 03. Impiegando più interruttori è possibile elaborare numeri anche molto grandi in poche frazioni di secondo

tiamoci ad esaminare brevemente alcuni principi di base.
Ai fini di questa prima spiegazione, un computer può essere paragonato ad una scatola piena di minuscoli interruttori collegabili tra loro secondo più combinazioni. Un computer è, in realtà, qualcosa di molto più complesso, grazie soprattutto all'enorme progresso tecnologico: basti sapere che anche nei più piccoli computer sono spesso presenti più di 250.000 piccoli "interruttori". Ciò è oggi possibile usando il più recente frutto della tecnologia elettronica: il "chip al silicio".
Ognuno di questi piccoli interruttori può essere chiuso (**on**) o aperto (**off**). Chiunque abbia una minima conoscenza di calcolo combinatorio è in grado di comprendere che le variazioni possibili sono immensamente numerose. Con grande vantaggio per l'utente, i computer vengono venduti corredati di appositi programmi interni alla macchina, capaci di gestire convenientemente tutte le possibili combinazioni di questi interruttori, permettendo di "parlare" al computer mediante semplici parole della lingua inglese.
Molte persone si stupiscono nello scoprire che di per sé il computer non possiede alcuna conoscenza "utile". Ci si meraviglia che non sappia quale sia la capitale dell'Afghanistan o l'altezza del Monte Bianco. Nonostante la vasta diffusione di questi apparecchi, infatti, è ancora diffusa l'errata convinzione che il computer sia un "cervello elettronico" sede di tutto lo scibile umano.
Siamo ancora lontani da una simile realtà: il "cervello" del computer (il chip di silicio) non conosce né l'alfabeto né, tantomeno, l'aritmetica. Anzi, la sua conoscenza si limita ad alcune centinaia di combinazioni numeriche: qualsiasi tipo d'informazione deve essere necessariamente convertito in numeri per essere usato dal computer. Quei minuscoli interruttori, cui abbiamo accennato prima, servono, appunto, a memorizzare numeri: una combinazione di interruttori chiusi/aperti rappresenta un numero nel sistema binario (nel quale esistono soltanto i valori 0 ed 1). Il fatto che il computer possa ricordare, o più precisamente "conservare", un'informazione è la base del suo stesso funzionamento.
Per fare un esempio, la memoria di uno Spectrum della Sinclair può conservare una serie di informazioni equivalente a circa sei pagine dattiloscritte (pari a circa 20.000 caratteri). Se poi si usano memorie di massa (come nastri magnetici), la quantità di dati memorizzabili non ha più limiti.
Oltre a conservare i numeri nella propria memoria, un computer può manipolarli, eseguendo su di essi semplici operazioni aritmetiche: può sommarli, sottrarli, confrontarli tra loro e spostarli in altre zone della memoria. Tutte le applicazioni pratiche di un computer si basano su queste poche e semplici operazioni. Supponiamo di voler conservare un testo nella memoria: potremmo inventare un codice tale da associare un numero ad ogni lettera dell'alfabeto: saremmo così in grado di manipolare non solo numeri ma anche parole. Oppure, supponiamo di voler disegnare un'immagine sullo schermo del computer: faremmo riferimento ad un reticolo opportunamente numerato, in modo da conservare nella memoria del computer quale punto dello schermo deve essere acceso o spento. Per nostra fortuna, i codici necessari esistono già e i programmi per gestirne la conversione vengono forniti assieme al computer.
Un programma per computer può essere definito come una particolare sequenza di istruzioni elementari (somma, sottrazione, confronto, ecc.) che, una volta eseguite dal computer, consentono di svolgere compiti molto più complessi. Le istruzioni elementari vengono immesse nella memoria del computer sotto forma — ancora una volta — di numeri. Ciò può far nascere un lecito dubbio: com'è possibile immettere dati nel computer se, come abbiamo detto, il computer non è in grado di far niente senza che vi sia prima un programma che permette tale inserimento?



Un codice per le lettere ed i numeri

Un gruppo di 8 interruttori consente 256 combinazioni di acceso/spento, un numero sufficiente per codificare tutti i caratteri riportati sulla tastiera del computer (lettere dell'alfabeto, caratteri numerici, punteggiatura e simboli vari). L'illustrazione mostra le combinazioni usate nel codice ASCII per rappresentare le lettere X ed Y

L'ovvia risposta è che tra i vari programmi forniti dai fabbricanti del computer ve ne sono di speciali, che sovrintendono appunto a funzioni quali la decodifica dei caratteri selezionati sulla tastiera, la visualizzazione dei caratteri sul video e molti altri.
Grazie a questi "programmi residenti" (immessi dai costruttori) l'utente può introdurre nuove sequenze di istruzioni usando la tastiera. Questo processo di immissione di codici numerici altro non è che "programmazione in linguaggio macchina", di cui avremo occasione di parlare più avanti nel corso. Il linguaggio macchina è decisamente difficile da apprendere e da usare, cosicché alcuni ingegnosi programmatori hanno scritto dei "linguaggi" che traducono semplici parole inglesi di uso corrente (ad esempio: PRINT, BEEP, LOAD e LIST) in gruppi di istruzioni, in linguaggio macchina, comprensibili ed eseguibili dal computer. Oggi anche i più piccoli computer portatili contengono questi linguaggi e tra essi il più diffuso è il BASIC. È possibile in tal modo scrivere programmi, usando normali parole al posto di poco comprensibili sequenze numeriche.
Quando si scrive un programma in BASIC, (sia esso per un nuovo gioco o per una qualche applicazione commerciale) non ci si accorge della grande quantità di operazioni elementari che il computer esegue ad ogni semplice comando BASIC. A questo livello è molto facile scrivere un programma che, per esempio, possa rispondere a quale sia la capitale del Belgio. In altre parole, il cervello elettronico conosce soltanto ciò che gli viene insegnato. Non può scoprire niente da solo. Ma allora i computer sono davvero utili? Certo, per la grande quantità di informazioni che essi possono immagazzinare ed elaborare a velocità irraggiungibili dall'uomo. C'è da aggiungere che non sempre tutte le informazioni di partenza devono essere introdotte manualmente: possono far parte di programmi già pronti. Al contrario, si possono acquistare programmi per elaborare le informazioni immesse dall'utente, come nel caso dei **word processor** per l'elaborazione dei testi, o dei **database**, che permettono di archiviare qualsiasi tipo di informazioni e di richiamarle successivamente applicando un criterio selettivo. Per esempio, un database può contenere tutte le informazioni relative ai volumi di una biblioteca. Con pochi comandi e quasi istantaneamente, il computer può estrarre informazioni del tipo «Quali libri di G.B. Shaw pubblicati a Londra prima del 1926 ci sono in biblioteca?».
Il fatto che il computer non comprenda che numeri è forse più un vantaggio che uno svantaggio. Se i computer elaborassero effettivamente le nozioni umane, essi sarebbero enormemente più complessi di quanto non lo siano già adesso e, probabilmente, sarebbe necessario un computer specifico per ogni applicazione. Inoltre, come si potrebbe memorizzare nel computer la definizione di un colore?
Una volta appurato che non è affatto neces-

SOFTWARE, PERCHÉ?

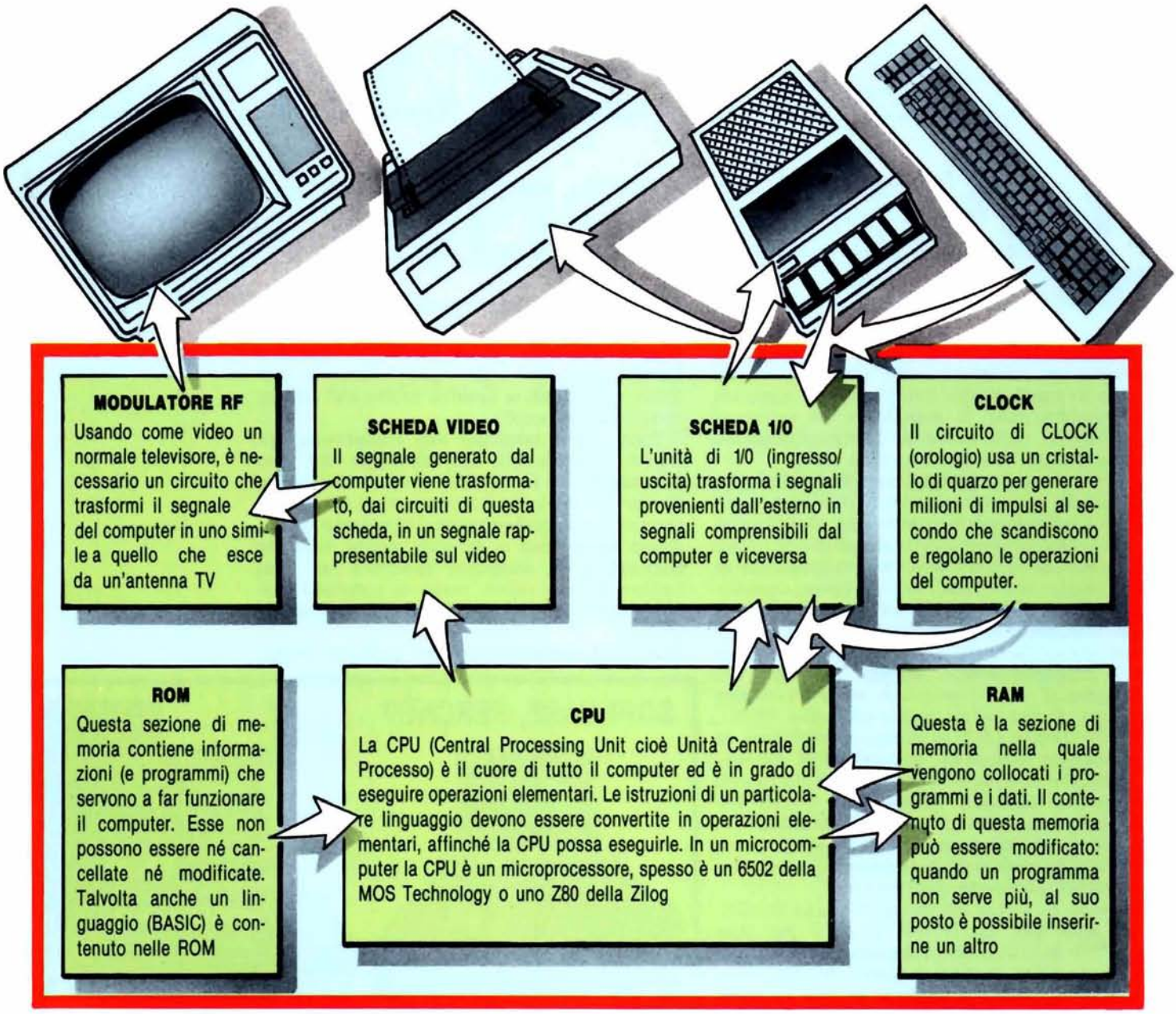


Il "micro" ha la natura dell'insegnante... è utile in ufficio...

per la scienza... e per il divertimento.

Il computer è uno strumento flessibile e versatile, ma è dal software che esso trae la propria potenza. Lo stesso apparecchio può essere adoperato dall'uomo d'affari usando programmi finanziari, dal tecnico con programmi di statistica, oppure semplicemente per il divertimento con i giochi più fantasiosi. È il software che determina l'utilizzazione del computer

TONY LODGE



COSA ACCADE ALL'INTERNO

Per il funzionamento di un sistema completo è necessario collegare tra loro le varie unità. I chip di silicio, cuore del computer, sono nell'unità centrale e talvolta anche nella tastiera

sario che il computer "capisca" le informazioni da elaborare — così come farebbe un uomo — si può comprendere perché un solo computer possa svolgere i compiti più disparati. Tutto si riconduce all'abilità del programmatore, che deve essere capace di convertire e ridurre il problema da risolvere in numeri. Volendo produrre musica con un computer, sarebbe impensabile memorizzare i suoni veri e propri: procederemo, invece, descrivendo ciascuna nota della scala musicale con un numero proporzionale al tono o alla frequenza della nota. Potremo quindi convogliare i segnali elettrici generati dal computer verso un altoparlante, anziché su di un video, per ascoltare il risultato. Similmente, in un videogioco, possiamo spostare astronavi o sparare razzi attraverso lo schermo, semplicemente manipolando opportunamente dei numeri che rappresentano la forma e la posizione dell'astronave o del razzo. Immagini, movimenti, colori, forme e suoni possono tutti essere rappresentati con numeri elaborabili dal computer e, successivamente, riconvertiti in segnali su video o attraverso altoparlante. Quindi alla domanda iniziale «cos'è un computer?» si può rispondere che il computer è un apparecchio in grado di memorizzare segnali elettrici che rappresentano numeri. Alcuni di questi numeri vengono interpretati come istruzioni per manipolare gli altri numeri. Il computer esegue migliaia di queste istruzioni al secondo con meticolosità, precisione, senza mai stancarsi e senza commettere errori (salvo che ve ne siano nel lavoro del programmatore). Il risultato di queste operazioni produce nuovi numeri, che vengono tradotti in forme comprensibili all'uomo. È il lavoro del programmatore che rende i computer utilizzabili e, spesso, indispensabili.

Guida all'acquisto di un microcomputer

Come sopravvivere nella giungla delle offerte

Videoregistratori, televisori, e complessi hi-fi hanno un elemento in comune: ognuno di questi apparecchi ha un'unica e precisa funzione. Anche se esistono modelli più o meno sofisticati di altri, un complesso stereo servirà sempre e soltanto a riprodurre suoni, mentre una lavatrice non può che lavare biancheria e un televisore ricevere e visualizzare le immagini TV.

Il discorso cambia totalmente nel caso di un computer, perché centinaia di persone possono acquistare lo stesso modello di computer ed ognuna di esse utilizzarlo per qualcosa di totalmente diverso. Ecco perché l'acquisto di un computer implica molta più attenzione.

Il primo passo è quello di annotarsi con cura tutte le funzioni che vogliamo far svolgere all'apparecchio. Volendo imparare i fondamenti della programmazione in BASIC, un apparecchio come lo Spectrum della Sinclair o come l'Oric-1 può andar bene. Se, invece, vogliamo divertirci con giochi, o usare un word processor o tenere una contabilità dovremo indirizzare la scelta su apparecchi come il BBC od il Commodore 64. Anche elementi quali il prezzo e l'affidabilità devono essere tenuti ben presenti.

L'elenco deve essere il più completo possibile, per evitare di trovarsi poi con un computer che non risponde alle nostre esigenze.

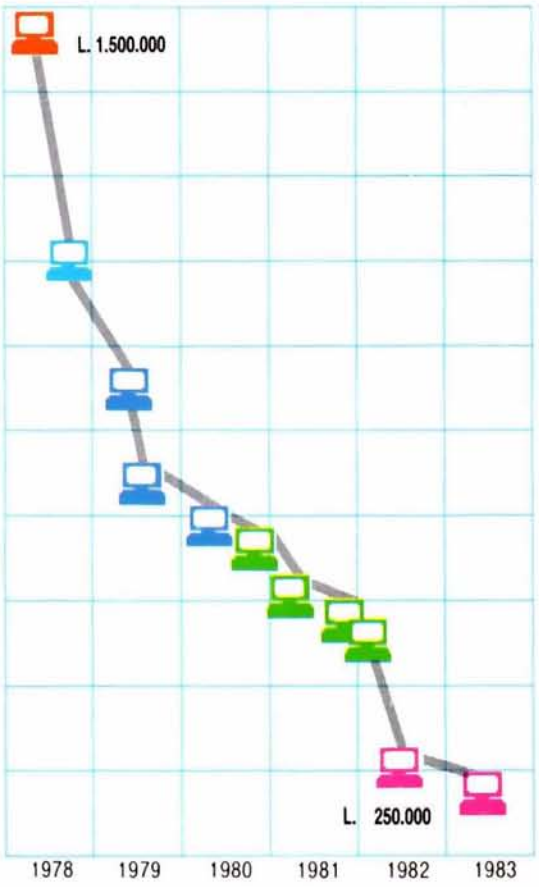
Spesso il computer scelto non è che un modulo base, al quale vanno aggiunte unità accessorie (per la conservazione dei programmi o dei dati) perché sia veramente utilizzabile con profitto. Unità accessorie tipiche sono registratori a cassette o unità a dischetti. Alcuni computer necessitano, inoltre, di speciali monitor, mentre altri si possono collegare direttamente ad un normale televisore. In questo caso, potrebbe rendersi necessario l'acquisto di un secondo televisore, poiché non sempre tutta la famiglia è disposta ad osservare incomprensibili e misteriosi messaggi sul video di casa, al posto del suo programma TV preferito! Altro punto ancora è la stampante, nell'eventualità che si desideri avere su carta copie dei programmi o dei dati elaborati.

In tutti i casi, è buona norma informarsi sulla quantità di **software** disponibile per il computer in questione: per alcuni apparecchi vi è abbondanza di giochi, ma scarsità di applicazioni "serie" o viceversa.

Inoltre, deve essere considerata la quantità di memoria necessaria per i programmi che intendiamo usare: più i programmi sono complessi e sofisticati, maggiore è la quantità di memoria impiegata. Per i word processor, ad esempio, è necessaria molta memoria per conservare il testo da elaborare. In genere 32K di RAM sono sufficienti per la maggior parte delle applicazioni medie, mentre 16K bastano per dei videogiochi con buona grafica. Di regola conviene acquistare il massimo della memoria consentita dal... portafoglio.

Alcuni degli **home computer** più completi (BBC, Commodore 64 ed Atari 800) trovano applicazione anche in piccole attività commerciali. A questi tre apparecchi, infatti, è possibile collegare unità a dischetti, stampanti e **modem** (per l'allacciamento via telefono ad altri computer).

Riassumendo, è necessario avere le idee ben chiare su ciò che vogliamo fare col computer, prima di acquistarne uno.



LA CADUTA NEL PREZZO DEI MICRO

La tabella mostra visivamente il sensibile calo dei prezzi di computer, con almeno 16K di RAM, negli ultimi anni

I PIÙ DIFFUSI IN EUROPA



ATARI 400

MEMORIA STANDARD: 16 Kbyte

ESPANDIBILITÀ: Non espandibile
COLLEGABILE A: Registratore a cassette ATARI, unità dischi ATARI, stampante, joystick

ASPETTI POSITIVI: Basso costo. Molto software disponibile

ASPETTI NEGATIVI: Tastiera 'a sfioramento'. Il BASIC viene venduto separatamente

COMMENTI: L'applicazione più adatta è per i giochi



ATARI 800

MEMORIA STANDARD: 48 Kbyte

ESPANDIBILITÀ: Non espandibile
COLLEGABILE A: Registratore a cassette ATARI, TV, monitor, unità dischi ATARI, stampante, joystick

ASPETTI POSITIVI: Tastiera di tipo tradizionale. Ampia scelta di software, compatibile con l'ATARI 400

ASPETTI NEGATIVI: Collegabile soltanto a registratore (o unità dischi) ATARI. La versione del BASIC non è tra le più standard

COMMENTI: Dotato di tastiera migliore del modello 400, può essere adoperato per piccole attività commerciali



BBC MICRO

MEMORIA STANDARD: Mod. A: 16 Kbyte; Mod. B: 32 Kbyte
ESPANDIBILITÀ: Fino a 32 Kbyte
COLLEGABILE A: Registratore a cassette, TV, monitor, unità dischi, joystick, sistemi PRESTEL, CEEFAX, ed in ECONET con altri BBC

ASPETTI POSITIVI: Ampia scelta di software. Buoni effetti sonori, se amplificati. Il BASIC è versatile

ASPETTI NEGATIVI: Selezione limitata di colori nella grafica. Manuali non sempre chiari. I suoni generati necessitano di amplificazione

COMMENTI: Un computer dalle buone possibilità, dalle interessanti espansioni



COLOUR GENIE

MEMORIA STANDARD: 32 Kbyte

ESPANDIBILITÀ: Non espandibile
COLLEGABILE A: Registratore a cassette, TV, monitor, stampante, impianto HI-FI, joystick, cartucce aggiuntive

ASPETTI POSITIVI: Tastiera di tipo tradizionale. Effetti sonori chiari

ASPETTI NEGATIVI: Il funzionamento non è tra i più veloci. Il software disponibile non è molto. La grafica è un po' scadente

COMMENTI: È forse il computer a colori più economico sul mercato



OLIVETTI M10

MEMORIA STANDARD: 8 Kbyte
ESPANDIBILITÀ: Fino a 32 Kbyte

COLLEGABILE A: Stampante, modem, plotter, registratore a cassette, lettore di codici a barre
ASPETTI POSITIVI: Ottima portabilità (alimentazione interna o esterna). Il display è incorporato

ASPETTI NEGATIVI: Le prestazioni sarebbero ancora migliori se fosse dotato di registratore incorporato
COMMENTI: È un buon apparecchio del suo genere, ottimo se deve essere usato in viaggio



LYNX

MEMORIA STANDARD: 48 Kbyte
ESPANDIBILITÀ: Fino a 192 Kbyte

COLLEGABILE A: Registratore a cassette, TV, monitor, stampante, unità dischi, joystick
ASPETTI POSITIVI: Immagini nitide sullo schermo. Programmi di utilità già incorporati. Tastiera di tipo tradizionale

ASPETTI NEGATIVI: Il BASIC è del tutto particolare. La memoria è espandibile soltanto dal costruttore
COMMENTI: Un apparecchio ben costruito, con una tastiera piacevole all'uso



NEW BRAIN

MEMORIA STANDARD: 32 Kbyte
ESPANDIBILITÀ: 2 Mbyte (2000 Kbyte!)

COLLEGABILE A: Registratore a cassette, TV, monitor, unità dischi, stampante, modem
ASPETTI POSITIVI: Il BASIC ha molte caratteristiche interessanti. Ha notevoli capacità per l'elaborazione dei testi. È semi-portatile

ASPETTI NEGATIVI: Connettori non standard e facilmente danneggiabili. Non è tra i più adatti al principiante
COMMENTI: È un apparecchio ampiamente espandibile, molto competitivo nella gamma dei portatili. È dotato di una porta seriale RS232



ORIC-1

MEMORIA STANDARD: 16/18 Kbyte
ESPANDIBILITÀ: Non espandibile

COLLEGABILE A: Registratore a cassette, TV, monitor, modem, mini unità dischi
ASPETTI POSITIVI: Tastiera con disposizione tradizionale. È disponibile una stampante a 4 colori. Effetti sonori chiari

ASPETTI NEGATIVI: Limitata scelta di software

COMMENTI: Un apparecchio di buona costruzione, adatto per principianti. La tastiera è facile da usare

IAN McKINNELL

I PIÙ DIFFUSI IN EUROPA



COMMODORE 64

MEMORIA STANDARD: 64 Kbyte
ESPANDIBILITÀ: Non espandibile
COLLEGABILE A: Registratore a cassette, TV, monitor, unità dischi, stampante, joystick
ASPETTI POSITIVI: Ottima e veloce gestione della grafica sullo schermo. Effetti audio molto buoni
ASPETTI NEGATIVI: Nessuno di rilievo
COMMENTI: Può essere usato sia per lo svago che per applicazioni serie, grazie ai molti accessori



COMMODORE VIC20

MEMORIA STANDARD: 5 Kbyte
ESPANDIBILITÀ: Fino a 32 Kbyte
COLLEGABILE A: Registratore a cassette, TV, monitor, unità dischi, stampante, joystick
ASPETTI POSITIVI: Tastiera di tipo tradizionale. ROM a cartuccia opzionali. Buoni effetti audio. Ampia scelta di software
ASPETTI NEGATIVI: La memoria (nella versione standard) non è molta
COMMENTI: Molto diffuso. Necessita della espansione di memoria per applicazioni utili



DRAGON 32

MEMORIA STANDARD: 32 Kbyte
ESPANDIBILITÀ: Fino a 64 Kbyte
COLLEGABILE A: Registratore a cassette, TV, monitor, unità dischi, stampante, joystick
ASPETTI POSITIVI: Tastiera di tipo tradizionale. Il BASIC è molto veloce. Il software è generalmente adatto al Tandy Color Computer e viceversa
ASPETTI NEGATIVI: Il connettore per l'alimentazione si può facilmente danneggiare. I connettori per le periferiche non sono standard
COMMENTI: Forse non uno dei migliori apparecchi è comunque abbastanza diffuso.



EPSON HX20

MEMORIA STANDARD: 16 Kbyte
ESPANDIBILITÀ: Fino a 32 Kbyte
COLLEGABILE A: Registratore a cassette, stampante, modem, cartucce estraibili
ASPETTI POSITIVI: Tastiera di tipo tradizionale. Display, stampante e registratore a microcassette incorporati. Buon software disponibile
ASPETTI NEGATIVI: Il display incorporato visualizza soltanto 4 linee alla volta
COMMENTI: Uno fra i migliori portatili, dotato anche di una porta seriale RS232



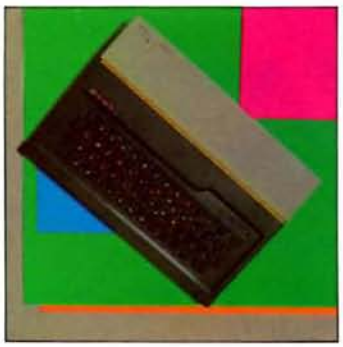
SINCLAIR ZX81

MEMORIA STANDARD: 16 Kbyte
ESPANDIBILITÀ: Oltre 56 Kbyte
COLLEGABILE A: Registratore a cassette, TV, mini stampante
ASPETTI POSITIVI: Possibilità di creare caratteri grafici. Prezzo accessibilissimo
ASPETTI NEGATIVI: Le piccole dimensioni e la tastiera "a sfioramento" creano impaccio nell'uso
COMMENTI: Questo computer non è molto sofisticato, ma sono disponibili numerosi accessori per migliorare le prestazioni



SINCLAIR SPECTRUM

MEMORIA STANDARD: 16 o 48 Kbyte
ESPANDIBILITÀ: Fino a 48 Kbyte
COLLEGABILE A: Registratore a cassette, TV, micro unità a dischi, modem, mini stampante, joystick
ASPETTI POSITIVI: Prezzo molto contenuto per un computer a colori. Molto software disponibile (soprattutto giochi)
ASPETTI NEGATIVI: Tastiera non molto affidabile. Effetti audio appena udibili. Talvolta ha problemi nel caricamento di programmi. Si scalda molto
COMMENTI: È una scelta eccellente per il principiante, per la facilità d'uso



SORD M5

MEMORIA STANDARD: 4 Kbyte
ESPANDIBILITÀ: Fino a 36 Kbyte
COLLEGABILE A: Registratore a cassette, TV, monitor, stampante, cartucce estraibili, joystick
ASPETTI POSITIVI: Di ottima fattura, è anche dotato di una spia per l'alimentazione
ASPETTI NEGATIVI: I tasti sono "gommosi" al tatto. Il BASIC è un poco limitato
COMMENTI: Le espansioni sono piuttosto care. È stata annunciata una migliore versione di BASIC



TI 99/4A

MEMORIA STANDARD: 16 Kbyte
ESPANDIBILITÀ: Fino a 52 Kbyte
COLLEGABILE A: Registratore a cassette, TV, monitor, joystick
ASPETTI POSITIVI: Tastiera di tipo tradizionale
ASPETTI NEGATIVI: Il BASIC è molto lento. Non è disponibile un'ampia scelta di software
COMMENTI: La lentezza del BASIC e la scarsezza di software ne hanno impedito una larghissima diffusione

Installare il micro

Accessori necessari per comunicare con il computer, per conservare programmi o per giocare

STAMPANTE

È necessaria, se vogliamo che il computer riporti su carta i listati dei programmi o i risultati di elaborazioni. I differenti prezzi delle stampanti dipendono dalla qualità e dalla velocità di stampa

UNITÀ A DISCHI

Sui dischi si memorizzano programmi e/o dati. Le unità a dischi sono più costose dei registratori a nastro, ma offrono la possibilità di conservare le informazioni e di accedere ad esse in modo più rapido ed efficiente; sono indispensabili per le applicazioni commerciali



TELEVISORE

Un normale televisore può essere utilizzato per visualizzare messaggi od immagini generate dai programmi. Quando si adopera la tastiera, ogni carattere battuto viene visualizzato sullo schermo. Impiegando un vero monitor da computer la resa è di qualità nettamente superiore

REGISTRATORE A CASSETTE

Un normale registratore a cassette è il mezzo più economico per conservare e rileggere programmi e/o dati che, altrimenti, con lo spegnersi del computer, andrebbero persi



IL COMPUTER

Il computer è il cuore del sistema; però necessita di parti accessorie, collegabili mediante appositi connettori, per comunicare con l'esterno. Spesso la tastiera — simile a quella di una macchina da scrivere, ma con alcuni tasti in più — è incorporata nel contenitore del computer

LA SFERA

Questo accessorio, che viene generalmente adoperato nei video-games per muovere immagini sullo schermo, è più facile e veloce da usare dei joystick

JOYSTICK

Piccoli strumenti, simili a quelli delle macchinette delle sale-giochi, servono in genere per comandare il movimento di immagini sullo schermo (ad esempio: astronavi nei videogiochi); alcuni hanno anche dei pulsanti utilizzabili per comandi speciali

MARK WATKINSON

Computer e software

Senza software, il computer non può funzionare. Vediamo come scegliere quello più adatto

Il software è la metà invisibile di un sistema computerizzato: senza di esso il computer non è che un ammasso inservibile di circuiti elettronici.
All'interno dei chip di un computer esistono migliaia o addirittura milioni di microscopici interruttori elettronici che vanno opportunamente aperti o chiusi. Come nel caso di un normale interruttore elettrico, è ovvio che occorre un intervento per accenderlo o spegnerlo; è il software che dice al computer quali e quanti di questi interruttori devono essere aperti o chiusi ed in quale successione ciò deve avvenire.
Il termine software indica proprio l'insieme di istruzioni che permettono il funzionamento di un computer. Queste istruzioni non sono altro che numeri, che la CPU (il cuore del computer; vedere pag. 4) interpreta, provvedendo a chiudere o aprire gli interruttori giusti.
Questi numeri però sono capiti dal computer soltanto se espressi in forma binaria (vedere pag. 28).
Essi sono il risultato di una lunga catena di eventi, la cui origine è ciò che il programmatore aveva in mente quando ha iniziato a scrivere il programma.
Un programma per computer (il termine "programma" indica una particolare parte di software) può esistere sotto molte forme diverse.
L'unica indicazione certa è che la sua forma finale deve essere espressa in modo comprensibile al computer.
Facciamo un esempio concreto. Un tecnico che volesse far funzionare l'impianto di un semaforo stradale usando un computer, dovrebbe scrivere un programma per generare le corrette sequenze di verde/giallo/rosso. Ma prima di scrivere il programma, si prepara una concisa descrizione della sequenza



ROM

La ROM (Read Only Memory, ossia "Memoria a sola lettura") è uno dei tanti tipi di memoria per computer: consente di memorizzare programmi in maniera permanente. La maggior parte dei personal computer viene venduta con una ROM che contiene il BASIC. Spesso è possibile acquistare ROM aggiuntive con altri linguaggi o programmi

FLOPPY DISK

Il software può essere memorizzato su dischetto magnetico. La lettura e la scrittura vengono fatte su tracce, come nei dischi musicali, ma usando una testina simile a quella dei registratori. I dischi offrono maggior capacità e velocità rispetto ai nastri, ma comportano l'acquisto di costose unità di lettura/scrittura (vedere a pag. 8)

CASSETTE

Il software viene spesso venduto registrato su cassette magnetiche del tutto simili a quelle musicali. Nella maggior parte dei casi si tratta di giochi. Un programma viene caricato nel computer usando un lettore di cassette. A caricamento avvenuto, il lettore può essere spento

CARTUCCE

Una cartuccia è, sostanzialmente, una ROM convenientemente inscatolata. Alcuni computer hanno speciali connettori dove infilare la cartuccia. Questo sistema viene in genere usato nel caso dei linguaggi

Un computer funziona soltanto se è alimentato col software (sequenze di istruzioni). In questa illustrazione vengono presentati i diversi tipi di supporto fisico necessari per conservare e per fornire il software. Ciascun tipo di supporto ha particolari caratteristiche

delle operazioni (ad esempio: il segnale giallo deve accendersi contemporaneamente al verde; poi il verde ed il giallo si spengono e si accende il rosso).
È ovvio che istruzioni sotto forma di linguaggio corrente non si possono utilizzare direttamente con il computer: esse vanno tradotte in un programma usando un particolare linguaggio, ad esempio il BASIC. Le frasi originali devono essere riespresse con i termini propri del linguaggio BASIC. Il BASIC non è altro che un programma particolare (programma interprete) necessario per convertire le istruzioni BASIC in istruzioni adatte al computer. Il prodotto finale di questa conversione è ciò che si chiama "linguaggio macchina".
Il software offerto in vendita, anche se scritto con linguaggi diversi, è sempre codificato in linguaggio macchina.
Talvolta resta permanentemente in una memoria del computer (ROM), ma più spesso è inciso su cassette o dischetti magnetici, che rappresentano quindi il supporto fisico del software.
Il software, per poter essere utilizzato dal computer, deve essere trasferito dalla cassetta o dal dischetto nella memoria centrale del computer.
Effettuato il "caricamento del programma" (così viene chiamata l'operazione di trasferimento) il programma può essere "lanciato" (eseguito).

L'acquisto del software

Un computer senza software è, come abbiamo già detto, inutilizzabile, ma a seconda del programma inserito, esso può elaborare testi, tenere una contabilità o diventare un videogame.
Sta all'utente decidere quale applicazione farne: partendo dal problema che si vuole affrontare, si arriva a scegliere il programma adatto.
Durante questo processo di selezione, si analizzano a fondo i vari requisiti. Se si parte dal principio di voler offrire ai nostri figli un diversivo ai loro normali giochi, cercheremo il programma tra le varie guerre spaziali, invasioni galattiche o simulazioni più fantasiose (vedere pag. 14).
Si deve poi vedere se quel particolare programma esiste in una versione per il computer che noi possediamo: infatti le varie marche di computer sono così differenti tra loro che virtualmente non esiste alcuna compatibilità.
Un programma scritto per l'Atari 800 non funziona sullo Spectrum, cosicché la scelta è limitata ai programmi adatti al nostro computer.
Ma ciò non basta: occorre tener conto della quantità di memoria fisicamente disponibile nel computer. Se il programma scelto necessita di più memoria esso comporta l'acquisto di memoria aggiuntiva.

FIRMWARE

Il termine hardware indica la parte "tangibile" del computer: i circuiti, la tastiera, ecc...
Di contro il software consiste nella parte "invisibile" del computer: "sequenze di istruzioni" o "programmi".
Ma esiste un altro termine: il firmware.
Inizialmente 20-30 anni fa, i programmi venivano conservati su nastri di carta perforati (simili a quelli dei telex). Poi vennero usati i nastri magnetici ed infine i dischetti. Negli anni '70 è stato introdotto l'uso di ROM specifiche (vedere pag. 4). Sulle ROM viene memorizzato un programma in modo permanente. Questa particolare combinazione di software e di hardware viene chiamata firmware

ELABORARE TESTI

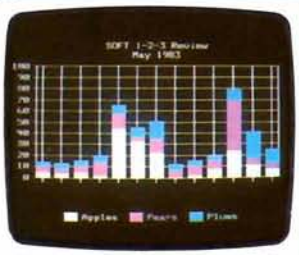


Word Processor

L'uso di un word processor consente di avere una macchina da scrivere "intelligente".
Anche il dattilografo più bravo commette errori, ma con un word processor è possibile correggerli prima che i testi vengano stampati. Un primo risultato evidente è un enorme risparmio di tempo e di carta a vantaggio di una maggiore efficienza.
La tastiera del computer sostituisce quella della macchina da scrivere, e sullo schermo appaiono i caratteri battuti sulla tastiera.
La similitudine si ferma qui, perché da questo punto entra in gioco la versatilità del computer: gli errori possono essere corretti istantaneamente sullo schermo.
Le parole possono venir sostituite, spostate o cancellate del tutto. Ciò vale anche per interi paragrafi: si può decidere di riorganizzare l'intero testo (sempre lavorando sullo schermo) e giungere a stamparlo soltanto quando si è raggiunta la stesura definitiva.
In ogni caso il testo può essere conservato su disco o su cassetta per permettere soste durante il lavoro o per riutilizzare il testo in tempi successivi, come avviene nel caso di circolari periodiche, oppure per testi ricorrenti (contratti di vendita, conferme di ordini, ecc.)



CONTABILITÀ GENERALE



Package di contabilità

Poiché un computer può eseguire operazioni aritmetiche, non stupisce affatto che gran parte dei programmi disponibili siano scritti per svolgere mansioni di calcolo. La gamma di programmi in questo campo è veramente impressionante: si va dalla gestione di una contabilità semplificata a programmi per la gestione completa di tutti gli aspetti contabili ed amministrativi di una attività commerciale. Generalmente — per gestire le notevoli quantità di dati che vengono usati in queste applicazioni — è necessario l'uso di una o più unità a dischi.

In genere questi programmi interagiscono con l'utente mediante l'uso di sistemi domanda/risposta: il programma presenta sul video una o più richieste e, in base alle risposte fornite dall'operatore, vengono eseguite le necessarie elaborazioni sui dati. Applicazioni tipiche sono: la fatturazione automatica, la gestione automatica dei riordini per il magazzino, la preparazione di bilanci, ecc...

I prezzi dei programmi possono anche raggiungere l'entità di qualche milione

ARCHIVIARE



Database

Un computer può vagliare quantità grandissime di informazioni in tempi brevissimi. Maggiore è la mole di informazioni da elaborare, più diventa indispensabile l'uso di un computer. Nella forma più elementare, un database può contenere una piccola rubrica di numeri telefonici.

Anche se l'uso di un database è decisamente facile, le funzioni che esso svolge possono essere notevolmente complesse. Facciamo l'esempio di uno studioso di botanica, che stia scrivendo un trattato sui funghi. Egli può conservare su archivi magnetici (**files**) tutte le informazioni relative alle varie specie di funghi, assieme a riferimenti bibliografici, ecc.

Prima dell'avvento dei computer, un simile lavoro richiedeva la compilazione di un gran numero di schede. La ricerca di una particolare informazione richiedeva un notevole lavoro. Oggi è possibile memorizzare tutto su uno o più dischetti e raggiungere quella particolare informazione, nel termine di pochi secondi. Inoltre un database offre la possibilità di selezionare le informazioni secondo criteri che noi stessi fissiamo: nell'esempio trattato, far stampare un elenco di titoli nei quali appaiano le parole "funghi velenosi", oppure una lista dei funghi diffusi in una certa zona del paese sono operazioni di una facilità estrema. Per utilizzare tutte le risorse di un database è indispensabile l'uso di unità a dischi

CALCOLO NUMERICO



Lavagne elettroniche

Le lavagne elettroniche (**spreadsheets**) sono uno strumento grazie al quale è possibile rispondere a domande del tipo «cosa accadrebbe se...» e che, senza computer, implicano laboriosi calcoli. Qualsiasi attività commerciale coinvolge tutta una serie di elementi variabili legati l'uno all'altro (costo delle materie prime e della manodopera, scarti di produzione e via dicendo).

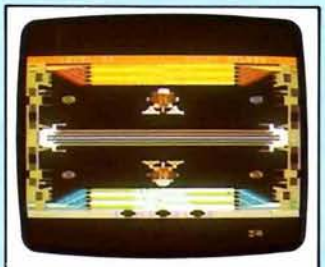
La possibilità di costruire dei modelli, partendo da valori reali, e variare poi alcuni di questi, può consentire l'esame di situazioni ideali (prospezioni) in base alle quali impostare l'andamento generale dell'attività.

Ogni valore essenziale viene inserito all'incrocio di una riga e di una colonna in una specie di tabella. Ogni incrocio può essere collegato ad un altro utilizzando formule matematiche (ad es.: i valori di ogni riga della colonna C sono ottenuti sottraendo i rispettivi valori della colonna B da quelli della colonna A).

Una volta costruito il modello è possibile variare uno o più valori per vedere immediatamente quanto questi nuovi dati possano influenzare la situazione generale.

Gli utenti di questi tipi di programmi sono in genere uomini d'affari, ingegneri o scienziati, che si trovano nella necessità di elaborare quantità di dati fortemente variabili

DIVERTIMENTO



Giochi

I computer, oltre che per applicazioni serie, possono essere adoperati anche per il tempo libero. Esiste una vasta scelta di giochi eseguibili al computer: dai classici scacchi, alla dama, al bridge, al poker, per arrivare ai più sofisticati **war games** o giochi strategici, che richiedono grandissima abilità e concentrazione.

Alcuni programmi sono simulazioni di eventi reali, come nel caso di simulazione del volo di un aereo, altri — pur rimanendo nella sfera del divertimento — si avvicinano al campo della didattica.

Questo genere di software è fortemente interattivo: stimola continuamente l'operatore con la richiesta di dati o istruzioni per rendere vivace e brillante il procedere dell'azione. Alcuni tasti assumono funzioni speciali (per posizionare immagini o sparare "proiettili" nelle scene create sullo schermo). In certi giochi viene fatto uso di particolari strumenti di comando (joysticks), per facilitare lo spostamento di immagini sul video

SOFTWARE PRONTO ALL'USO

La maggior parte del software in commercio viene venduta su dischetto o su cassetta, assieme al manuale che ne spiega l'uso. L'Apple Writer è un programma abbastanza tipico per l'elaborazione di testi. Viene venduto su dischetto ed il manuale, redatto con molta cura, include una guida di facile consultazione per i principianti. Purtroppo non sempre i manuali sono chiari e facili: questo è un aspetto da considerare con grande attenzione al momento dell'acquisto

Come indicazione di massima, più un programma è completo e sofisticato, più memoria viene usata.
Altro fattore è la forma fisica del supporto (vedere riquadro), che deve essere quella adatta al computer: alcuni programmi vengono venduti su cassetta, altri su dischetto. Se un programma è venduto soltanto su dischetto e non possediamo l'unità a dischi, non sarà possibile utilizzare il programma senza prima acquistare l'unità a dischi (in genere abbastanza costosa).
Alcuni programmi, particolarmente i giochi, comportano l'acquisto di speciali "cloche" (joystick). Per applicazioni più serie può essere necessaria, invece, una stampante per avere delle copie su carta di dati o programmi.
Infine, va tenuto conto del prezzo: alcuni giochi costano poche migliaia di lire, ma programmi per applicazioni commerciali possono costare anche milioni.



FAUSTO DORELLI

Categorie di software

Per alcuni aspetti, i giochi formano un gruppo a parte: dopo tutto, un gioco serve soltanto a divertire. La maggior parte del software è studiata e realizzata per facilitare e velocizzare un particolare lavoro. I campi nei quali gli esperti di software si applicano per migliorare l'efficienza o la rapidità di processi ripetitivi sono innumerevoli. Prendiamo il caso di una segretaria: quante volte deve ribattere a macchina il testo di una lettera o di una relazione prima di arrivare alla versione finale?
Con l'impiego di un **word processor**, che consente di elaborare testi sul video prima della stampa definitiva, è possibile un enorme risparmio di tempo, energia e... carta.
Un altro settore che ben si presta alla computerizzazione è quello amministrativo. Una gran parte dell'immenso lavoro necessario per calcolare paghe e stipendi o per tenere in perfetto ordine i libri contabili può essere svolto molto più rapidamente e semplicemente con l'uso del software adatto. Molti dei programmi di questo genere sono molto specifici e — quasi sempre — non basta un solo programma per svolgere tutte le mansioni. Esistono così dei gruppi di programmi (o **packages**) per la gestione delle paghe, che calcolano accuratamente gli stipendi e stampano le buste-paga; altri packages curano la gestione del magazzino, memorizzando le quantità di articoli acquistati e venduti (alcuni prevedono anche una procedura automatica di riordino per articoli sottoscorta o addirittura calcolano automaticamente il fabbisogno dei materiali in base ai consumi).
Un'altra importante applicazione in cui i computer sono degli assi è l'archiviazione, la selezione e l'estrazione di informazioni. Programmi di questo genere si chiamano **database**. Un buon database può cancellare il ricordo di disordinati scaffali pieni di ingombranti carte e consente inoltre la ricerca di informazioni anche con criteri selettivi impossibili da eseguire a mano.
L'ultimo grosso gruppo di software comprende le cosiddette "lavagne elettroniche". Con questi programmi è possibile costruire con estrema facilità complessi quadri finanziari (ad esempio: bilanci di previsione, budget ecc..), evitando ingenti sprechi di carta e di tempo.
Tutti i vari tipi di software ai quali abbiamo qui accennato sono venduti già "pronti per l'uso", ovvero sono stati studiati per risolvere tutta una serie di problemi di larga applicazione. Ma in taluni casi, quando i problemi o le procedure non rientrano nello standard, è necessario ricorrere all'opera — in genere notevolmente costosa — di un programmatore, che scriva un programma specifico. In alternativa, per programmi non troppo complessi, si può imparare a programmare da soli. Armati di un linguaggio come il BASIC, possiamo sbizzarrirci in un'impresa che ci costerà soltanto il tempo impiegato per risolvere il problema.

Lo sapevate che...?

Notizie e curiosità che difficilmente riuscireste a trovare altrove

A cosa serve un "Home Computer", oltre che per giocare?

Gli "Home Computer" (computer da casa) possono essere impiegati, in piccole attività commerciali, per tenere la contabilità, per l'elaborazione di testi e lettere. Un altro uso, per esempio da parte di associazioni, potrebbe essere quello per tenere aggiornata la situazione dei soci. Queste sono applicazioni abbastanza diffuse, ma più



generalmente si possono raggruppare in: elaborazione di testi, calcolo numerico e gestione di piccole banche-dati.

Le dimensioni ed i prezzi dei computer sembrano calare continuamente. Continuerà così?

L'avanzamento della tecnologia permette la costruzione di computer sempre più piccoli, perché i componenti interni hanno un ingombro sempre minore, mentre un fattore che rimarrà costante è la dimensione della tastiera. Quindi, fino a quando non verrà inventato un sistema di introduzione dei dati, sarà difficile che le dimensioni possano calare ulteriormente. Anche per quanto riguarda i prezzi (alcuni



computer non costano più di 200/300.000 lire) non sono previsti grandi cambiamenti.

Il BASIC è difficile da imparare?

No. Di per sé non è un linguaggio difficile. Superata la barriera della lingua inglese, il BASIC non è in fondo che un piccolo «sotto-linguaggio» dalle regole abbastanza semplici. Ma, sebbene il linguaggio non presenti molte difficoltà, la stesura di un lungo e complicato programma può essere un lavoro molto complesso.

Quando si dovrebbe imparare il BASIC.

Il migliore momento per cominciare lo studio del BASIC è ... quando è necessario usarlo! Può darsi che ciò non avvenga mai, se si usano programmi già pronti. Ma avendo l'idea di scrivere piccoli programmi per conto proprio, allora è bene applicarsi subito allo studio del BASIC.

Perché un monitor costa più di un televisore?

Un monitor è un apparecchio progettato per visualizzare caratteri generati da un computer e, poiché durante la maggior parte del tempo che si passa al computer gli occhi sono fissi sullo schermo, le immagini devono essere particolarmente nitide e gradevoli alla vista.



Quanta elettricità consuma un Home computer?

Meno di una lampada da 60 watt.

È possibile usare il computer per la didattica, come suggerisce talvolta la pubblicità?

Sì. I computer possono essere adoperati nell'insegnamento sia dei bambini che degli adulti. Nei paesi anglosassoni lo Stato spesso interviene concretamente affinché ogni scuola abbia in dotazione almeno un computer, per avvicinare i giovani all'uso di questi ap-



parecchi. La chiave per usare il computer nell'insegnamento risiede in un buon software: esistono già molti programmi scritti a questo fine, ed anche un linguaggio sperimentale (il LOGO), che permette anche ai più giovani l'esplorazione delle enormi capacità potenziali di un computer. In questo caso i bambini imparano, allo stesso tempo, anche le basi della programmazione.
Questo argomento viene affrontato più in dettaglio a pag. 30.

Alcune tastiere di computer sono del tutto simili a quelle delle macchine da scrivere, mentre altre hanno tasti in gomma: che differenza c'è?

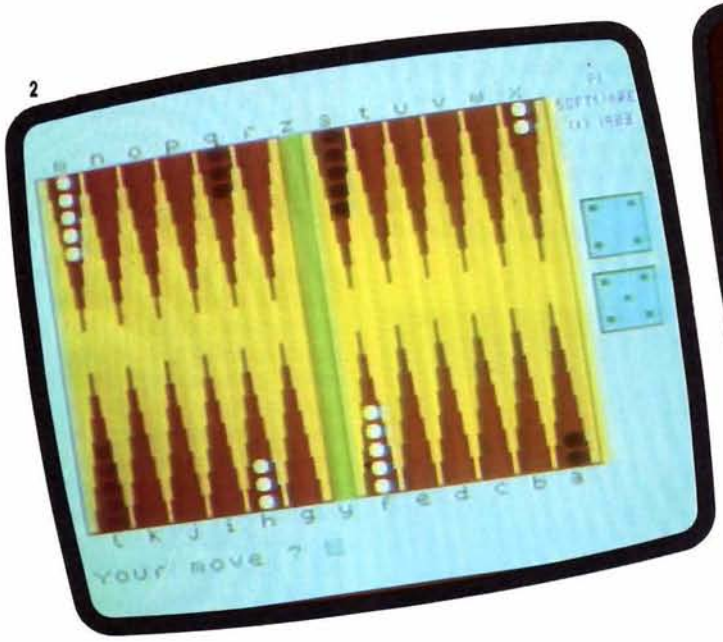
Quasi nessuna per chi non è un bravo dattilografo. È solo questione di praticità nella costruzione.

Il mondo dei giochi

Caverne, draghi, voli spaziali, combattimenti. Avventura ma anche didattica sul computer

C'è ormai un'infinità di gente che si diverte giocando col computer: non si può dire che la fantasia manchi ai creatori di programmi il cui scopo è lo svago: un grande caleidoscopio di colori, forme e situazioni immaginarie si sprigionano dallo schermo del computer!
Grazie ai prezzi sempre più bassi dei personal e home computer, i video-games da salegiochi stanno per essere soppiantati dai computer-games, molto più raffinati, divertenti e — cosa da non trascurare — meno mangiasoldi. È stato detto che nei paesi anglosassoni, dove la diffusione dei personal e degli home ha già raggiunto cifre considerevoli, la gente si avvicina al computer prima per giocarci, e poi — eventualmente — per farci qualcosa di più serio. In fondo, che male c'è?











Se questo può servire a sdrammatizzare almeno un poco il primo impatto con gli apparecchi del futuro, che ben vengano i giochi.

Avventure

Non tutti i giochi hanno bisogno di suoni ed immagini riccamente elaborate per essere divertenti: ce ne sono molti, indicati col generico termine di "avventure", che descrivono luoghi e situazioni a parole senza per questo perdere il loro fascino. Si potrebbero definire come "romanzi a finale aperto", in cui l'intervento dell'operatore modifica via via la trama dell'avventura fino ad avere sempre nuovi finali.
Il computer, dopo aver descritto un ambiente di partenza, attende le istruzioni dell'operatore: queste possono essere di movimento (vai a nord, gira a destra, ecc.) e di azione (prendi la lampada, accendi un fiammifero, ecc.). Le situazioni e gli ambienti cambiano quindi ad ogni partita.
L'ambientazione di queste avventure può essere un intrico di grotte popolate da mostri (come nel noto *Dungeons and Dragons*) oppure regioni stellari costellate di alieni amici o nemici o, ancora, una semplice casa di campagna, dove risolvere il mistero di un delitto. Qualunque sia lo scenario, il giocatore deve esplorare luoghi, affrontare ostacoli ed aguzzare l'ingegno per risolvere misteri. Un buon gioco d'avventura è come un buon romanzo, con in più un coinvolgimento diretto del lettore. Inizialmente questo genere venne sviluppato sui grossi computer, ma sono oggi disponibili molte versioni per i piccoli personal.

Giochi da tavolo

Era inevitabile che la maggior parte dei più diffusi giochi da tavolo venisse riproposta su



computer. Uno dei primi vantaggi è che il computer è un compagno di gioco sempre pronto, abile e paziente. È anche possibile allenarsi contro la macchina per migliorare la propria tecnica, come nel caso degli *Scacchi*, un gioco che si presta benissimo ad esser fatto sul computer, sfruttando le capacità grafiche dello schermo.
Altri giochi disponibili, tra i molti in commercio, sono: il *Backgammon*, la *Dama* ed il *Bridge*.

Imparare giocando

Molti fornitori di software si sono dedicati allo sviluppo di giochi didattici, nei quali vengono presentati enigmi o semplici problemini da risolvere, assegnando punteggi in base all'esattezza delle risposte. Alcuni di questi giochi implicano esercitazioni didattiche quali la compilazione di parole, semplici operazioni aritmetiche e addirittura introducono i concetti fondamentali della legge della domanda e dell'offerta.
Un gioco molto adatto ad un pubblico infantile è, ad esempio, *La tartaruga* (*The Turtle*): con l'uso di semplici istruzioni, viene fatta muovere sullo schermo l'immagine di una tartaruga, e la "scia" che essa lascia costruisce pian piano disegni anche complessi. Spesso al computer viene collegata una specie di piccola tartaruga a motore, che riproduce su di un grande foglio di carta i movimenti (e quindi il disegno) fatti sul video.





Come in Sala Giochi

È forse il campo più vasto e più conosciuto, e — per usare un termine corrente — più gettonato. Invasioni spaziali, atterraggi lunari e guerre galattiche vengono simulate sul video del computer, accompagnate da fantasiosi effetti sonori.
Tutti questi giochi d'azione sono ora disponibili sui piccoli personal computer, sfruttandone al massimo le capacità grafiche.

Grande strategia

Un altro settore di giochi trasportato su computer è quello dei giochi tattici. Ponderatezza e riflessione sono gli elementi tipici di questo genere di giochi, nel quale eccellono i cosiddetti war games (giochi di guerra).
Il giocatore può diventare un generale e comandare le truppe nello scenario di battaglie realmente avvenute e fedelmente ricostruite al computer. Oppure può essere il re di un paese immaginario e doverne amministrare oculatamente le risorse difendendolo dagli attacchi nemici: in altre parole amministrarlo saggiamente.
Anche se questi restano soltanto dei giochi, il giocatore viene spesso coinvolto a tal punto da essere nelle condizioni di dover sviluppare le proprie capacità decisionali ed abituarsi a previsioni a lungo termine.



1. Magia. Entriamo in un mondo misterioso. Alcuni personaggi possono aiutarci, ma quale scegliere?

2. Backgammon. Il nostro avversario è dotato di una potenza formidabile: inesorabile logica del computer

3. ABC Dragon. Un gioco educativo per i piccoli

4. Zaxxon. Un classico delle sale-giochi. Lo schermo diventa un campo di battaglia per astronavi

5. War games. Siamo in grado di cambiare il corso della storia

6. Simulatore di volo. Diventiamo tutti esperti piloti

Diventare piloti

Alcuni programmi simulano in maniera quasi perfetta il volo: riproducendo fedelmente il cruscotto di un aereo o di una navetta spaziale, il giocatore può far decollare l'apparecchio, deve mantenerlo in volo ed infine farlo atterrare secondo le normali regole della realtà.
Nei giochi di simulazione, la fedeltà con cui si riproducono certe condizioni è fondamentale e talvolta l'unica differenza con la realtà è la possibilità per il giocatore di... sbagliare senza pagarne le conseguenze!

Carta d'identità

Come interpretare la scheda unita alla presentazione dei computer

Tastiera

La tastiera è progettata per essere di uso facile e piacevole. I tasti sono disposti come in una normale macchina da scrivere. Ogni carattere può essere automaticamente ripetuto semplicemente tenendo premuto il corrispondente tasto. È possibile digitare sia maiuscole che minuscole e, con apposita tastierina a parte, anche i numeri

Video

Il set di caratteri ASCII (America Standard Code for Information Interchange) è un gruppo standard di lettere, numeri e simboli espressi in forma di codice numerico. Il set è utilizzato da gran parte dei piccoli computer. Il numero di linee e di caratteri per linea varia a seconda della marca, al pari della possibilità di usare un normale televisore od un monitor come terminale

Memoria

Le cifre indicano la capacità della memoria, espressa in Kilobyte, ossia migliaia di byte. La ROM (Read Only Memory, Memoria a sola lettura) contiene programmi e dati necessari al funzionamento dell'apparecchio e talvolta anche lo stesso BASIC in forma permanente, mentre la RAM (Random Access Memory, Memoria ad accesso casuale) è la zona di memoria utilizzabile per i programmi ed i dati dell'utente

Interfacce

Sono disponibili diversi connettori e relativi circuiti per collegare stampanti, unità di lettura/scrittura, modem, ecc.

CPU

La CPU è l'Unità Centrale di Processo, il cuore del computer. Z80 è la sigla di un particolare microprocessore (chip) enormemente diffuso, prodotto dalla Zilog. Le operazioni della CPU vengono scandite da un orologio interno (clock), che lavora ad una frequenza di 2,2 milioni di oscillazioni al secondo

Caratteristiche di un computer "TIPICO"

Memoria	16 Kbyte di ROM, 32 Kbyte di RAM, è capace di indirizzare 48 Kbyte di RAM
Video	Può visualizzare il set di caratteri ASCII su 25 linee di 80 caratteri, su di un monitor oppure su un normale televisore
CPU	Chip Z80 con frequenza di 2.2 Mhz
Tastiera	Design ergonomico, formato QWERTY, ripetizione automatica dei tasti, maiuscole, minuscole e tastiera numerica
Interfacce	Ce ne sono per stampante, per mangianastri, per unità a dischi e per comunicazioni
BASIC	Comandi per grafica e effetti sonori, autodiagnosi degli errori e possibilità di programmazione strutturata
Grafica	Secondo formati compatibili col Teletext e col Viewdata, risoluzione massima 640 × 256, effetti tridimensionali
Effetti sonori	Sintetizzatore musicale, 5 ottave ed uscita HI-FI
Periferiche disponibili	Unità a cassette, unità a dischetti, unità a disco fisso, stampanti, plotter, digitiser, joystick, modem e sintetizzatori della voce
Linguaggi disponibili	FORTH, PASCAL, LOGO, LISP, PROLOG, ASSEMBLER

BASIC

Il linguaggio fornito col computer dispone di comandi per l'uso di effetti sonori e grafici. Controlla automaticamente la validità delle istruzioni e l'eventuale presenza di errori formali nel programma. Alcuni comandi supplementari del BASIC consentono di sfruttare convenientemente i programmi, in modo da permetterne una verifica più immediata

Effetti sonori

Singole note possono essere "suonate" con un'estensione di 5 ottave, ed il segnale proveniente dal computer può essere riprodotto da un complesso ad Alta Fedeltà

Periferiche disponibili

Registratore a cassette, una o più unità a dischetti o un disco fisso servono a memorizzare programmi e dati. Per l'input e l'output di testi ed immagini si possono usare una stampante, un plotter o un digitiser. Per alcuni giochi può essere conveniente collegare una coppia di joystick. Il modem è un apparecchio che consente ai computer di scambiare informazioni via cavo telefonico

Grafica

Il terminale video (display) è in grado di riprodurre immagini trasmesse dai sistemi Teletext e Viewdata su di uno schermo di 256 linee e 640 punti per linea. È possibile rappresentare immagini tridimensionali

Linguaggi disponibili

Questo è l'elenco dei linguaggi che possono essere utilizzati in alternativa al BASIC. A seconda dell'applicazione finale, un linguaggio può essere preferibile ad un altro. L'ASSEMBLER è un linguaggio molto più difficile del BASIC, ma consente una velocità di esecuzione molto più alta

Lo Spectrum

Anatomia di un microcomputer di diffusione europea: tecnologie a confronto

Lo Spectrum della Sinclair è un piccolo personal computer dotato di una grafica a colori, di un generatore di effetti sonori e di una notevole memoria. Il principale motivo del suo grande successo è indubbiamente il prezzo molto basso: lo Spectrum in versione 16K è stato il primo computer a colori ad avere un prezzo di vendita inferiore alle 400.000 lire. Ad un prezzo così accessibile, questo apparecchio ha avuto una larghissima diffusione, invogliando la produzione sia di software che di accessori specifici.

È possibile estendere la memoria dello Spectrum fino a 48K, consentendo in tal modo anche la stesura di lunghi e complessi programmi nel BASIC fornito di corredo. Si trovano moltissimi programmi già pronti su cassetta; ciò non toglie che molti possessori di Spectrum si applichino allo sviluppo di programmi per conto proprio. Il BASIC dello Spectrum — come del resto accade con tutte le altre marche — ha alcune caratteristiche sue proprie, pur mantenendo una struttura sufficientemente vicina al normale standard del linguaggio, ormai familiare al programmatore BASIC.

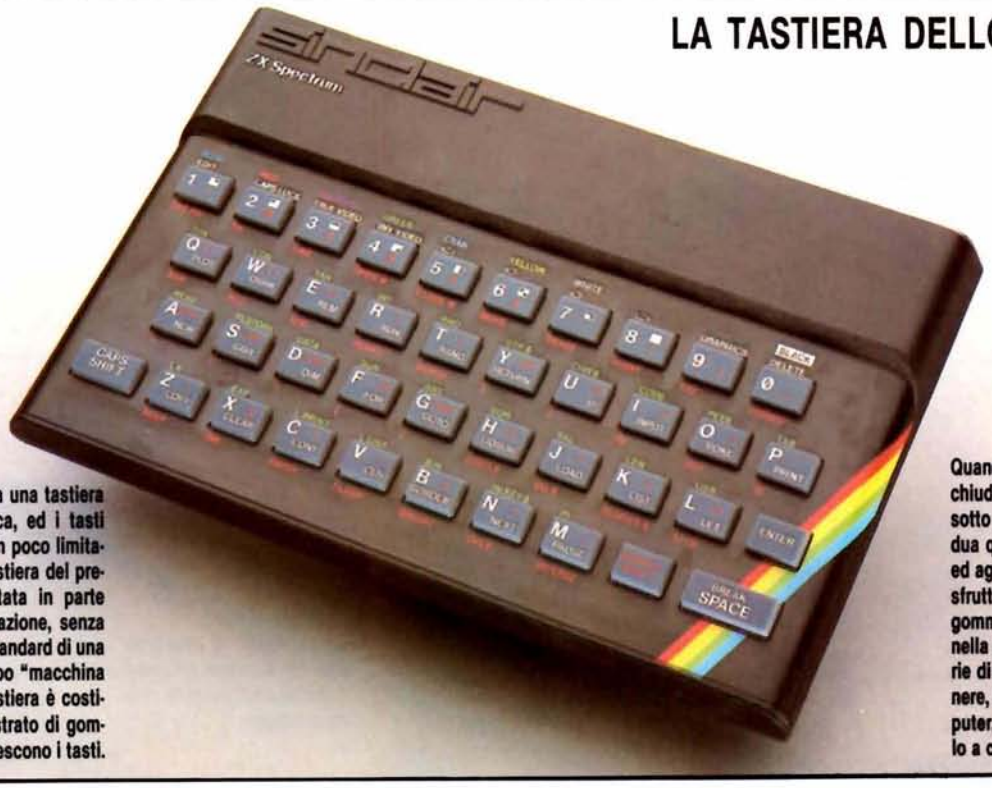
Le capacità grafiche e sonore dello Spectrum possono essere controllate mediante opportuni comandi del BASIC, ma, mentre la riproduzione di effetti sonori è piuttosto limitata, ciò che offre la grafica è addirittura stupefacente per un apparecchio così piccolo. Con l'uso dei comandi PAPER (carta) ed INK (inchiostro) si possono costruire immagini alterando il colore dello sfondo e dei disegni: con ulteriori comandi si possono disegnare figure perfino complesse.

Alcune parti dell'immagine possono anche lampeggiare.

Numerosissimi programmi in commercio per lo Spectrum sono giochi: dall'ormai famoso *Invasori Spaziali* fino a sofisticati programmi di simulazione, la cui verosimiglianza col reale è veramente impressionante.

La possibilità di espandere lo Spectrum con apparecchi accessori proviene in parte dalla stessa Sinclair (la stampante ed i microdischetti), ma il mercato offre anche tutta una varietà di altri accessori: joystick, interfacce per il controllo di strumenti, tastiere aggiuntive di formato standard, sintetizzatori di voce e via dicendo.

LA TASTIERA DELLO SPECTRUM



Lo Spectrum monta una tastiera piuttosto economica, ed i tasti hanno una corsa un poco limitata. Rispetto alla tastiera del precedente ZX81 è stata in parte migliorata la prestazione, senza però arrivare allo standard di una vera tastiera del tipo "macchina da scrivere". La tastiera è costituita da un unico strato di gomma, dal quale fuoriescono i tasti.

Quando si preme un tasto, si chiude un contatto posto al di sotto di esso. Il computer individua quale tasto è stato premuto ed agisce di conseguenza. Viene sfruttata l'elasticità del blocco di gomma per far ritornare il tasto nella posizione normale. Una serie di accorgimenti di questo genere, nel progetto di questo computer, hanno permesso di produrlo a costi eccezionalmente bassi



Il Microdrive ZX

Il Microdrive ZX si allaccia allo Spectrum mediante un'interfaccia collegata sul retro dell'apparecchio. L'unità offre 100K di memoria di massa ed il tempo di accesso si aggira sui 3,5 secondi



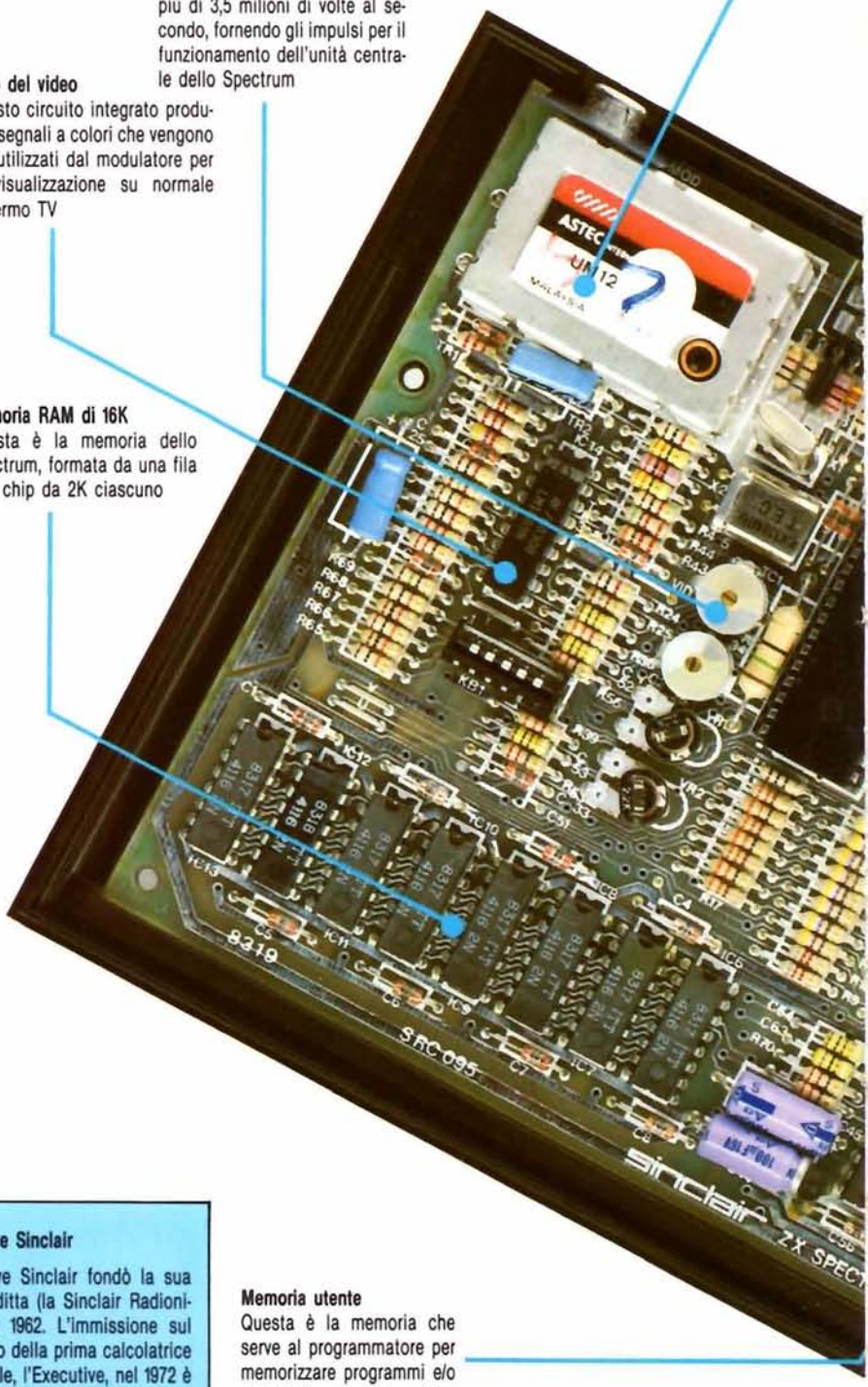
La stampante ZX

La stampante ZX si collega direttamente allo Spectrum. Il formato di stampa è di 9 righe per pollice, con la possibilità di riprodurre anche i simboli grafici. Il contenuto dello schermo può essere copiato su carta con il comando COPY

Clock
L'orologio interno (clock) oscilla più di 3,5 milioni di volte al secondo, fornendo gli impulsi per il funzionamento dell'unità centrale dello Spectrum

Chip del video
Questo circuito integrato produce i segnali a colori che vengono poi utilizzati dal modulatore per la visualizzazione su normale schermo TV

Memoria RAM di 16K
Questa è la memoria dello Spectrum, formata da una fila di 8 chip da 2K ciascuno



Memoria utente
Questa è la memoria che serve al programmatore per memorizzare programmi e/o dati. Può essere estesa fino a 48K

Connettore per tastiera
La tastiera viene allacciata a questo connettore

Dissipatore
Questa lastra di alluminio disperde il calore generato durante un prolungato uso del computer



Sir Clive Sinclair

Sir Clive Sinclair fondò la sua prima ditta (la Sinclair Radionics) nel 1962. L'immissione sul mercato della prima calcolatrice tascabile, l'Executive, nel 1972 è una conferma della capacità di miniaturizzazione di Sinclair, oltre a quella di uno spiccato gusto nel design e di una elevata intuizione commerciale. Nel 1979 Sinclair lasciò la Sinclair Radionics per fondare la Sinclair Research, dove nel 1980 venne sviluppato lo ZX80, seguito ad un solo anno di distanza dallo ZX81. Ambedue erano monocromi, ma nel 1982 è stato lanciato lo Spectrum a colori. Durante il 1983 Sinclair ha stabilito una propria sede di ricerca a Cambridge

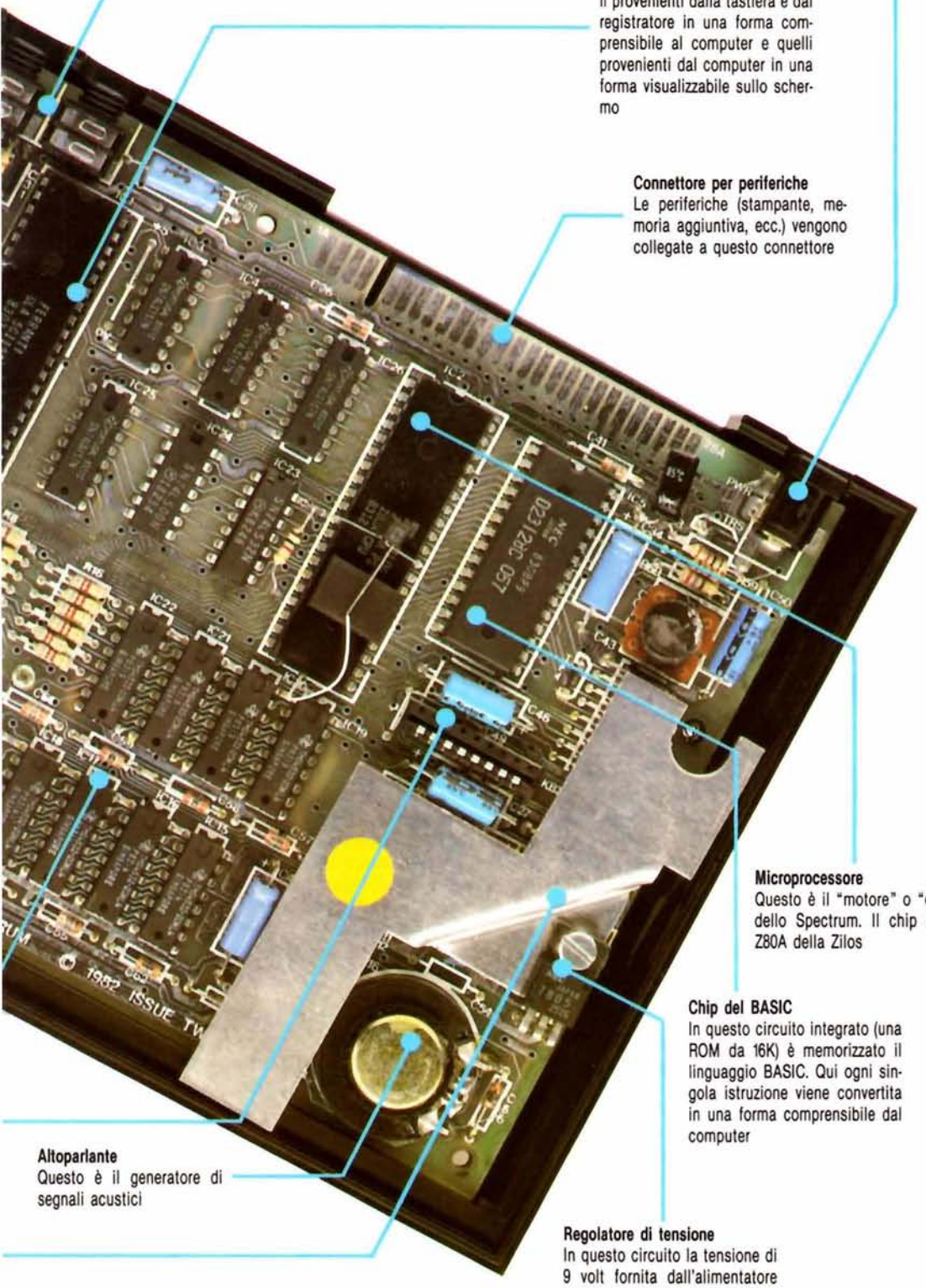
Modulatore RF
Questo circuito converte i segnali generati dal circuito video in segnali simili a quelli irradiati dalle emittenti TV

Spinotti per microfono e auricolare
Vengono utilizzati per trasferire informazioni da e verso il registratore a cassette

Presa di alimentazione
Qui si allacciano i nove volt occorrenti al funzionamento dello Spectrum e prodotti dall'apposito alimentatore

Circuito Input/Output
Questo circuito converte i segnali provenienti dalla tastiera e dal registratore in una forma comprensibile al computer e quelli provenienti dal computer in una forma visualizzabile sullo schermo

Connettore per periferiche
Le periferiche (stampante, memoria aggiuntiva, ecc.) vengono collegate a questo connettore



Microprocessore
Questo è il "motore" o "cuore" dello Spectrum. Il chip è uno Z80A della Zilos

Chip del BASIC
In questo circuito integrato (una ROM da 16K) è memorizzato il linguaggio BASIC. Qui ogni singola istruzione viene convertita in una forma comprensibile dal computer

Altoparlante
Questo è il generatore di segnali acustici

Regolatore di tensione
In questo circuito la tensione di 9 volt fornita dall'alimentatore esterno viene trasformata nelle varie tensioni richieste dai circuiti dello Spectrum

SPECTRUM SINCLAIR

DIMENSIONI

mm. 232 x 144 x 30

PESO

552 grammi

CLOCK

3,5 MHz

MEMORIA

16 Kbyte di RAM estendibile a 48 Kbyte. 16 Kbyte di ROM contengono il BASIC

VIDEO

24 linee di 32 caratteri o 192 x 256 punti per la grafica ad alta risoluzione. 8 colori disponibili

CPU

Z80A

TASTIERA

40 tasti mobili su unica base di plastica

INTERFACCE

Connettore per TV, connettore per registratore a cassette (senza comando a distanza), connettore a 28 piedini per periferiche

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

FORTH, PASCAL, LISP, LOGO, PROLOG

ACCESSORI COMPRESI

Alimentatore, cavi di connessione per TV e per registratore a cassette, un nastro con programma dimostrativo, 2 manuali

DOCUMENTAZIONE

Insieme allo Spectrum vengono forniti due manuali: quello per l'installazione e quello, più consistente, sulla programmazione in BASIC. Il secondo manuale ha una parte introduttiva dedicata alla tastiera che potrebbe essere migliorata, mentre la sezione relativa al BASIC mostra, anche con largo uso di esempi di programmazione, come sfruttare le capacità dello Spectrum.
Una serie di appendici costituisce una guida pressoché completa allo Spectrum ed al suo BASIC. Ambedue i manuali hanno una veste tipografica piacevole

Obbedire... e basta!

Il computer può far di tutto e non commette mai errori solo se "istruito" nella giusta maniera

Anche senza conoscere il funzionamento interno, chiunque è in grado di usare un computer. Questa è la prima lezione di una serie dedicata a tutto ciò che occorre sapere per scrivere un buon programma.
Spesso, dopo aver usato per un certo tempo programmi fatti da altri, nasce il desiderio di apportarvi modifiche o, addirittura, di riscriverli completamente per adattarli maggiormente alle proprie esigenze. Ma, poiché un computer non è in grado di programmarsi da solo, è necessario impartirgli una serie di istruzioni per comunicargli esattamente quali procedure deve eseguire per raggiungere i risultati che desideriamo. Questa serie di istruzioni formano ciò che si chiama "programma" e l'arte che ci permette di creare programmi si chiama "programmazione".
Non c'è niente di particolarmente difficile nella programmazione; inoltre non è indispensabile essere dei geni in matematica, sempreché non ci si voglia cimentare nella stesura di programmi pieni di calcoli complessi: per iniziare è sufficiente capire il linguaggio BASIC.

Il primo linguaggio

Assieme alla maggior parte dei personal computer viene fornito un linguaggio chiamato BASIC. Già il nome suggerisce l'idea che esso sia stato studiato per agevolare l'apprendimento delle basi della programmazione da parte dei principianti. Come ogni linguaggio umano, il BASIC ha una sua grammatica, una sua sintassi ed un suo vocabolario, anche se piuttosto esiguo. Il BASIC adopera un limitato gruppo di parole inglesi facili da riconoscere e da imparare. Come linguaggio per applicazioni generiche, è adatto sia al principiante che all'esperto programmatore.
L'unica pecca di questo linguaggio deriva dal fatto che, nel corso degli anni, molti fabbricanti di computer vi hanno arbitrariamente introdotto nuove particolari istruzioni. Come risultato si hanno oggi più versioni di BASIC, che si differenziano particolarmente per quanto riguarda i comandi impiegati per la gestione della grafica, dei colori e degli effetti sonori. Le principali variazioni comuni ai computer più diffusi verranno d'ora in poi riportate nella rubrica «Basic & Basic».
A causa delle variazioni appena menzionate, è pressoché impossibile scrivere un programma di una qualche complessità che si adatti a tutti i tipi di computer. Fortunatamente, il nucleo centrale delle istruzioni è comune a quasi tutti gli apparecchi. Inizieremo proprio da questo nucleo centrale, dal quale ci muoveremo verso programmi più complessi.

I primi passi

Cominciamo con lo scrivere un semplice programma e vediamo cosa accade. Quello che segue è un piccolo programma, che, apparentemente, presenta un computer che... commette errori! Una volta acceso il computer, digitiamo il programma esattamente come riportato, spaziature incluse. Il <CR> alla fine di ogni riga ci ricorda di premere il tasto RETURN (ritorno carrello). A seconda del computer, questo tasto è indicato con RETURN, ENTER o solo con ↵ .

```
10 REM I COMPUTER NON SBAGLIANO MAI <CR>
20 PRINT "DIGITA UN NUMERO" <CR>
30 INPUT A <CR>
40  LET A=A+1<CR>
50 PRINT "PENSO CHE IL NUMERO DIGITATO FOSSE"; <CR>
60 PRINT A <CR>
70 END
```

Finito di scrivere tutto questo, si provi ad usare il comando LIST <CR>. Il programma appena scritto dovrebbe apparire di nuovo sul video: LIST è un'istruzione per dire al computer di mostrare il listato di un programma in memoria. Se il computer ha riprodotto con esattezza il listato, possiamo provare il comando RUN (esegui). Se è stato commesso un errore nello scrivere il programma, basta riscrivere l'intera linea che contiene l'errore (numero di linea compreso!). Proviamo a digitare:

```
25 REM RIGA DI COMMENTO <CR>
```

ed usiamo di nuovo il LIST. Per cancellare la linea appena scritta, si digita il solo numero di linea, seguito da <CR>. Una volta controllato che il listato non contenga errori, lo si fa "girare" digitando RUN <CR>. A questo punto deve comparire sullo schermo:

```
DIGITA UN NUMERO
```

E noi premiamo il tasto "7" (attenzione: il computer — a meno che non si scriva un programma apposito — non riconosce la parola "sette" come un numero!).

ALTRI LINGUAGGI

Il BASIC è il linguaggio più diffuso sui microcomputer, ma non è l'unico. Nei primi anni della programmazione, gli scienziati e gli ingegneri usavano il FORTRAN. Altri linguaggi diffusi sui microcomputer sono il PASCAL, il FORTH ed il LOGO

Il Pascal

Similmente al BASIC, il PASCAL venne sviluppato principalmente come linguaggio didattico per programmatori, ma a differenza del primo, il PASCAL incoraggia la stesura di programmi con strutture più raffinate ed eleganti. Il PASCAL è disponibile anche per lo Spectrum, ad un costo accessibile, ed è adatto alla maggior parte delle applicazioni generiche

Il Forth

Le istruzioni del FORTH non si avvicinano al normale linguaggio inglese quanto il BASIC o il PASCAL, e sono di più difficile apprendimento. Al contrario, i programmi in FORTH sono più compatti e permettono la definizione di comandi "personalizzati", operazione impossibile con il BASIC

Il Logo

Questo linguaggio è relativamente giovane, ed è particolarmente indirizzato alla didattica. È sufficientemente semplice da essere usato dai giovanissimi. Può essere impiegato per insegnare la programmazione, avviando gli allievi verso un metodo logico di programmazione. Il LOGO impiega anche un sistema grafico di facile uso per disegnare immagini sullo schermo. Un accessorio meccanico (la "tartaruga") può essere collegato al computer (vedere pag. 31). Con semplici comandi si è in grado di muovere la "tartaruga" per eseguire complicati disegni

Sul video comparirà:

```
PENSO CHE IL NUMERO DIGITATO FOSSE 8
```

Il computer ha fatto un errore? Oppure non ha che eseguito un nostro ordine? Se osserviamo attentamente il programma, possiamo capire cosa abbia fatto il computer. Ecco la prima linea:

```
10 REM I COMPUTER NON SBAGLIANO MAI
```

REM sta per REMark (nota). Qualsiasi parola, che in una linea di programma segua l'istruzione REM, viene totalmente ignorata dal computer. Le REM sono molto utili per inserire commenti tra le righe di un programma. Questa particolare REM contiene il titolo del programma, ma non ci dice cosa faccia il computer. Impareremo più avanti quanto siano utili delle REM collocate opportunamente. Vediamo adesso:

```
20 PRINT "DIGITA UN NUMERO"
```

Quando il BASIC incontra l'istruzione PRINT, tutto ciò che segue viene "stampato" sullo schermo. Notare che la scritta è inclusa tra due doppi apici.
Una delle regole del BASIC è che una qualsiasi sequenza di caratteri inclusi tra due doppi apici preceduti da una PRINT apparirà sul video fedelmente riprodotta. Vedremo un'altra PRINT alla linea 60. Poi viene:

```
30 INPUT A
```

Sorvoliamo un attimo su questa linea, per vedere prima la 40:

```
40 LET A = A + 1
```

La lettera viene qui usata per rappresentare una variabile. Una variabile è paragonabile ad una piccola scatola etichettata, che può contenere un numero o delle lettere. Invece di doverci ricordare cosa c'è nella scatola, basta ricordare il nome sull'etichetta per poterne usare il contenuto. È come dire: «Mi passi la scatoletta B», invece di «Mi passi la scatoletta che contiene le viti a testa piatta da 15 mm».
In questa linea abbiamo una scatola «A». «A» è chiamata variabile, proprio perché ciò che contiene può essere variato. Possiamo assegnare un qualsiasi valore ad una variabile, ed è ciò che avviene nella linea 30:

```
30 INPUT A
```

La parola INPUT (letteralmente *immissione*) indica uno dei modi disponibili in BASIC per assegnare un particolare valore ad una variabile. Quando il BASIC incontra una INPUT, si ferma ad aspettare che qualcosa venga digitato alla tastiera. INPUT A dice al computer che abbiamo una variabile chiamata A e che qualsiasi cosa verrà digitato alla tastiera verrà assegnato a quella variabile. A questo punto, se si battono i tasti 7 <CR>, si mette il numero 7 nella scatola A.
Ovvero, usando il "gergo", abbiamo assegnato ad A il valore 7. Adesso che sappiamo cos'è una variabile ed abbiamo visto uno dei modi per assegnare un valore ad una variabile, riguardiamo la linea 40:

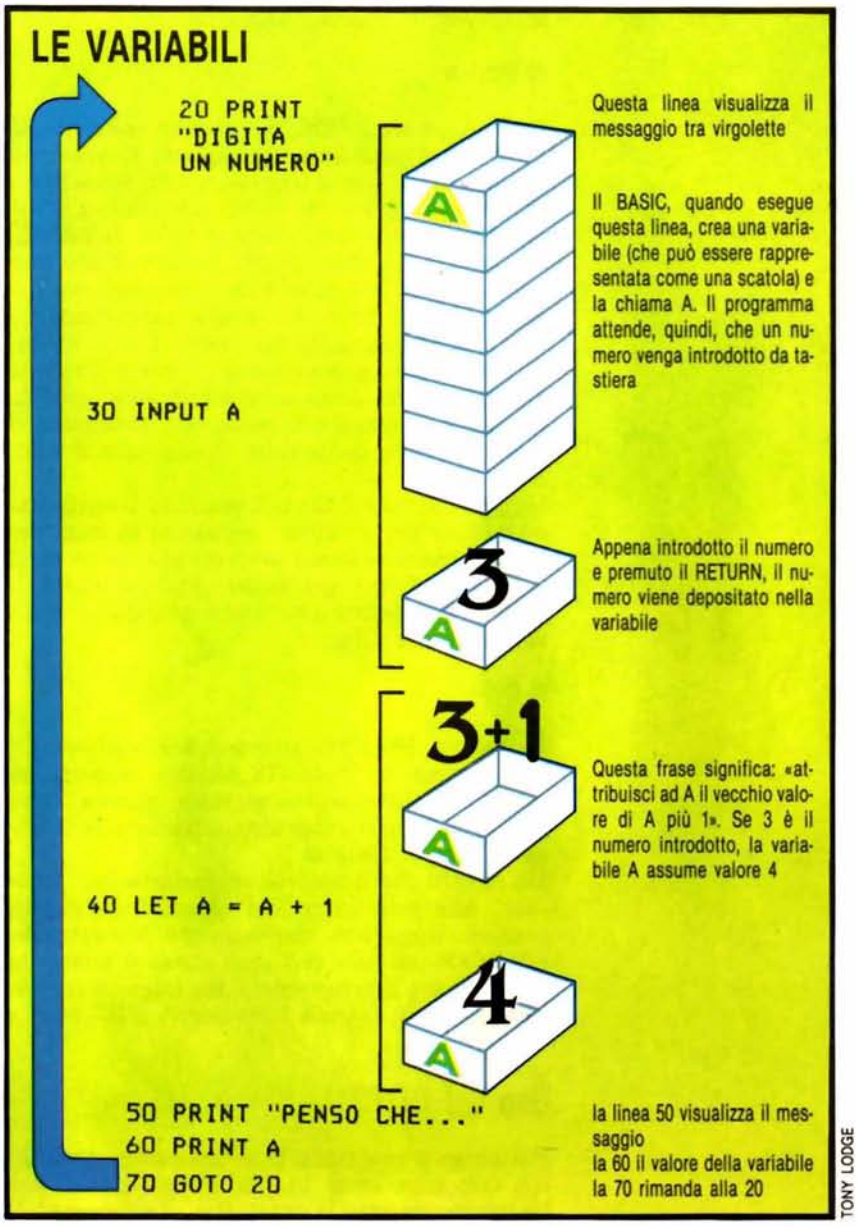
```
40 LET A = A + 1
```

Il nome della variabile alla quale viene assegnato un valore è sempre alla sinistra del segno di uguaglianza. In questo caso noi assegniamo un nuovo valore alla variabile A. In linguaggio corrente: «Assegna (LET) ad A il vecchio valore di A più 1». Il vecchio valore era 7, il nuovo sarà 8.

```
50 PRINT "PENSO CHE IL NUMERO FOSSE";
```

Ecco di nuovo una PRINT. Essa "stampa" la stringa di caratteri (ossia la sequenza di lettere o numeri) racchiusi tra doppi apici. Da

La tavola evidenzia l'uso delle "variabili" nel linguaggio BASIC e mostra anche come usare l'istruzione GOTO per formare un ciclo



Basic & Basic



Solo nello Spectrum e nello ZX81 la parola LET è obbligatoria, mentre in altri computer essa può essere omessa. La linea 20 può essere riscritta: A = A + 1



In genere l'istruzione END può essere omessa, poiché viene automaticamente aggiunta dal BASIC alla fine del programma. Nello Spectrum, nello ZX81, nell'Oric-1 e nel Dragon 32 sostituire con STOP



Sullo Spectrum e sullo ZX81 appare GO TO, sebbene si prema un solo tasto. In altri BASIC si può scrivere GOTO o GO TO indifferentemente

notare il punto e virgola alla fine della linea: serve per indicare al computer dove effettuare la stampa del prossimo elemento. Più avanti nel corso ritorneremo sull'uso del punto e virgola. Adesso vediamo:

```
60 PRINT A
```

Ancora un'altra PRINT, ma questa volta non ci sono doppi apici a racchiudere la A. Abbiamo già imparato che il computer non visualizzerà una semplice A sullo schermo, perché sarebbero necessari i doppi apici. Il BASIC, in assenza dei doppi apici, va a cercare una variabile con lo stesso nome del carattere che segue la PRINT. Se ne trova una, esso stampa il valore della variabile (se non la trova, avverte dell'errore oppure viene assunto il valore zero). Il nostro programma contiene una variabile A, cosicché il computer stamperà il valore in essa contenuto. Quale sarà il valore?
Non è 7, perché il BASIC esegue i programmi una linea dopo l'altra, seguendo la numerazione. Quando siamo arrivati alla linea 60, il valore di A era già stato cambiato in 8, e questo è il valore che viene stampato. Arriviamo infine alla:

```
70 END
```

La parola END (fine) avverte il BASIC che il programma è finito. In alcune versioni di BASIC è obbligatorio inserire questa linea alla fine di ogni programma (vedere la rubrica «Basic & Basic»).
Da notare che il nostro programma ha "lavorato" una sola volta. Per eseguirlo di nuovo occorre impartire nuovamente il comando RUN <CR>. Adesso vediamo come è possibile far "girare" il programma un numero indefinito di volte usando l'istruzione GOTO (vai).

Uso del GOTO

Più sotto è riportato il solito programmino, ma con una linea in più. Se avessimo nel frattempo spento il computer, dovremmo ribatterlo da capo, altrimenti basta inserire le linee 70 ed 80.

```
10 REM I COMPUTER NON SBAGLIANO MAI <CR>
20 PRINT "DIGITA UN NUMERO" <CR>
30 INPUT A <CR>
40 LET A=A+1 <CR>
50 PRINT "PENSO CHE IL NUMERO DIGITATO FOSSE"; <CR>
60 PRINT A <CR>
70 GOTO 20 <CR>
80 END <CR>
```

Dopo una LIST per controllare eventuali errori, facciamo partire con RUN <CR> il programma. Come nella prima versione apparirà:

```
DIGITA UN NUMERO
```

Noi premiamo un qualsiasi tasto numerico, seguito da RETURN. Il computer sommerà 1 al numero e visualizzerà il risultato:

```
PENSO CHE IL NUMERO FOSSE (il numero + 1)
```

Al che noteremo che a questo messaggio segue un nuovo DIGITA UN NUMERO. Possiamo così ripetere il processo un numero infinito di volte. La ragione va ricercata nella linea 70:

```
70 GOTO 20
```

Quando il BASIC incontra un'istruzione GOTO, invece di proseguire l'esecuzione con la linea seguente, passa al numero di linea specificato dopo la GOTO. Nel nostro caso, l'esecuzione viene rimandata alla linea 20, dove tutto il processo riparte da capo, continuando all'infinito. Anche volendo, non potremmo uscire da questo ciclo senza fine.
Naturalmente, è possibile scrivere programmi in modo da uscire da simili situazioni, e ciò verrà trattato nella prossima lezione. Nel frattempo, possiamo provare con il testo BREAK (se esiste sul computer).
Si noti che la linea 80 contenente la END, grazie all'istruzione GOTO 20, non viene mai eseguita. In alcuni BASIC, come abbiamo già detto, è obbligatorio inserire una END alla fine del programma.
Se non riusciamo ad uscire dal programma con il tasto BREAK, possiamo provare con il RESET. Se ciò ha interrotto il programma, possiamo provare di nuovo un LIST e, se questo funziona ancora dopo il RESET, non avremo bisogno di riscrivere il programma per gli esercizi che seguono. In caso contrario, il tasto di RESET sul vostro computer distrugge il contenuto della memoria e dovremo così riscrivere il programma.

Esercizi

Gli esercizi che seguono sono stati studiati per facilitare l'apprendimento ed è consigliabile eseguirli con attenzione per ottenere qualche progresso.
Prima di iniziare con gli esercizi, proviamo a fare qualche piccola modifica. Il computer

non corre alcun rischio facendo qualche errore o premendo qualche tasto sbagliato, quindi: coraggio! Per cambiare una linea, dopo aver eventualmente riscritto il programma, basta digitare il numero della linea che si desidera cambiare, scrivendo di seguito la nuova linea. Proviamo:

10 REM I COMPUTER TALVOLTA SBAGLIANO <CR>

Con un LIST vedremo che la prima linea è mutata. Proviamo a digitare semplicemente:

10 <CR>
LIST <CR>

La linea 10 non dovrebbe più esistere. Volendo la si può riscrivere (non dimenticando il numero di linea).

* Si riscriva il programma in modo che il computer stampi veramente il numero digitato. (Basta cancellare una linea!)

* Si riscriva la linea 70 in modo che l'esecuzione passi alla linea 80. Si faccia un LIST. Si esegua con un RUN. Da cosa dipende la diversa esecuzione?

* Si modifichi la linea 60 in modo che venga stampata una A (e non il contenuto della variabile).

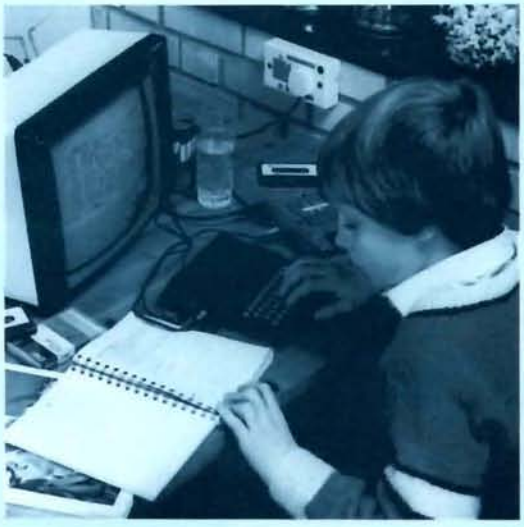
* Si ripristini la linea 60 in modo da far stampare di nuovo il valore della variabile e si cancelli la linea 10 (quella con la REM). Ci sono differenze?

* Si aggiunga un nuovo commento (usando REM) nella linea 25. Nuove linee si inseriscono semplicemente scrivendo, dopo il numero di linea, le nuove istruzioni. Si aggiunga un commento del tipo: "QUI SI ASPETTA PER UN VALORE DA TASTIERA". Fatto ciò si esegua un LIST per vedere i cambiamenti.

* Si riscriva il programma in modo che il valore venga moltiplicato per 10. Sarà necessario modificare la linea 50 in qualcosa come "IL NUMERO DIGITATO MOLTIPLICATO 10 È". Il simbolo della moltiplicazione in BASIC è * (non usare una x perché per il BASIC non è la stessa cosa).

Abbiamo imparato un bel po' di cose; come scrivere commenti (REM), come stampare una stringa di caratteri (PRINT), e come stampare il valore di una variabile; come modificare il percorso del programma con il GOTO.
In seguito vedremo come potremo uscire da un **loop** (ciclo) usando le istruzioni IF-THEN. Impareremo che è possibile far eseguire parti di programma per un numero prefissato di volte e creare dei ritardi nell'esecuzione per simulare che il computer "pensa".

...E VENNE IL BASIC



I linguaggi di programmazione sono stati creati perché l'uomo potesse comunicare con facilità con gli elaboratori: il BASIC è uno dei linguaggi oggi più diffusi perché facile da imparare ed usare. Le istruzioni consistono in semplici parole inglesi combinate, quando necessario, con elementari simboli matematici che si trovano sulla tastiera.
La facilità del BASIC è tale che, pochi minuti dopo aver montato un microcomputer appena acquistato, è già possibile iniziare a scrivere i primi semplici programmi. Il BASIC venne inventato nel 1965 da due insegnanti (Thomas Kurtz e John Kemeny) del Dartmouth College, nel New Hampshire, con il preciso scopo di semplificare i linguaggi esistenti in quel tempo. Da allora, l'impiego universale del BASIC ha portato ad alcune modifiche ed aggiunte, anche se, fondamentalmente, il nucleo centrale del BASIC è rimasto immutato.
Un programma è una sequenza di istruzioni che il computer esegue per svolgere un determinato compito. Questo compito può essere, ad esempio, la stesura di un piano finanziario mensile oppure lo spostamento di un'astronave sullo schermo del televisore. Il programma appare come una serie di linee numerate: ogni linea contiene una o più istruzioni e il numero di linea consente l'esecuzione nella corretta sequenza. I comandi si apprendono velocemente, ed anche i programmi più complicati sono composti da combinazioni e ripetizioni di questi comandi elementari.
I computer possono essere programmati anche in "linguaggio macchina" (un linguaggio considerato di "basso livello", in quanto la sua logica è più vicina a quella della macchina). Il BASIC, invece, è ad "alto livello", poiché impiega comandi più vicini al linguaggio umano. Esistono molti altri linguaggi ad "alto livello" studiati per specifiche applicazioni, ma il BASIC resta sempre il migliore avviamento alla programmazione.

La nuova rivoluzione

L'avvento del computer sta mutando la vita dell'uomo. Il futuro è già qui

La «rivoluzione del computer» si è sviluppata in coincidenza con la corsa alla conquista dello spazio. L'ingente investimento — miliardi di dollari — riversato in questo campo ha fatto sì che le migliori menti scientifiche si concentrassero nella produzione di apparecchiature sempre più sofisticate, al limite dell'impensabile. Grazie a ciò, una gran quantità di nuovi prodotti — tra i quali i circuiti elettronici miniaturizzati — è entrata a far parte della vita di tutti i giorni.

Inizialmente, la parola "computer" richiamava alla mente apparecchi molto ingombranti, pieni di migliaia di componenti elettronici, dai costi di produzione ingentissimi. Oggi gli stessi circuiti possono essere riprodotti all'interno di minuscoli chip al silicio, così piccoli da alloggiare nel contenitore di una tastiera. I circuiti integrati che fanno parte di un qualsiasi personal computer non solo sono di dimensioni ridottissime, ma anche i loro costi di produzione sono pressoché irrisori. La materia prima impiegata, il silicio, è una delle sostanze più diffuse in questo mondo: ogni granello di sabbia, infatti, è costituito quasi esclusivamente di silicio.

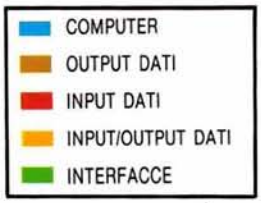
Con la diffusione dei computer, sia nei posti di lavoro che nelle case, stiamo assistendo alla seconda rivoluzione industriale.

La prima rivoluzione industriale portò a sostituire il lavoro manuale con apparecchiature meccaniche, la rivoluzione del computer porterà a grossi risparmi di tempo nel lavoro in generale ed all'automatizzazione dei processi industriali, con l'impiego di "robot".

Gli effetti a lungo termine di questo cambiamento non sono affatto prevedibili. Un fatto certo è che i modelli attuali del rapporto lavoro/tempo-libero cambieranno rapidamente. I robot, vere e proprie estensioni meccaniche dei computer, rimpiazzeranno centinaia di operai, ai quali rimarranno due sole possibilità: imparare nuove specializzazioni o rimanere senza lavoro.

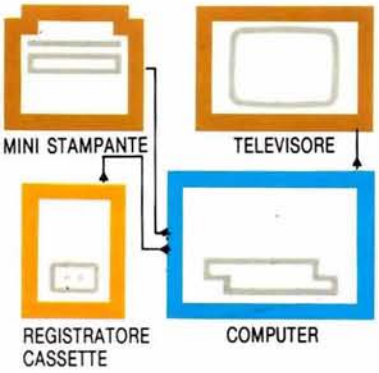
Anche gli insegnanti possono venir sostituiti usando la tecnologia odierna. Tra breve, una normale visita medica si ridurrà ad un colloquio con il terminale di un computer.

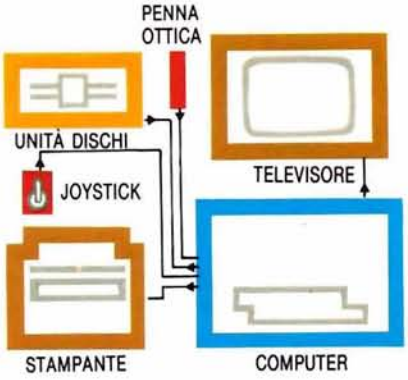
La conseguenza sociale della prima rivoluzione industriale è stata una fuga dalle campagne a beneficio dello sviluppo — talvolta squallido, anche se redditizio — del settore industriale. Siamo sull'orlo di una rivoluzione altrettanto drammatica: dato il ruolo cardine che il computer assume via via nella nostra società, la sopravvivenza dipende in gran parte dal saper comprendere e usare questo potente apparecchio.



IN CASA

Un "computer domestico" a basso costo può essere acquistato per poche centinaia di migliaia di lire, e può rappresentare un passaporto per il futuro



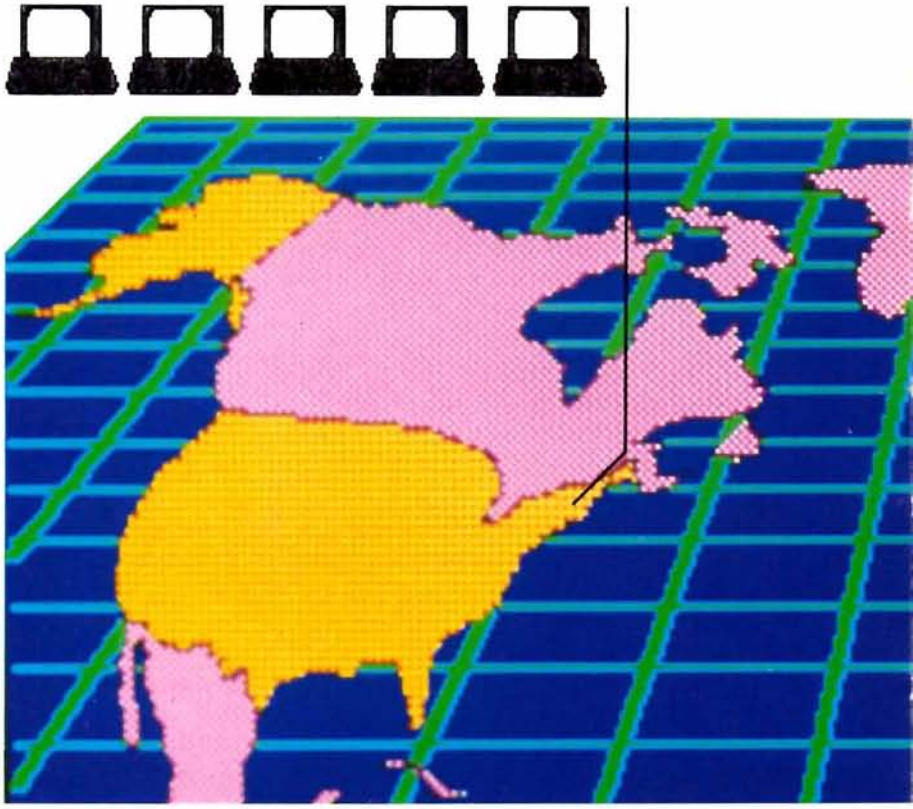


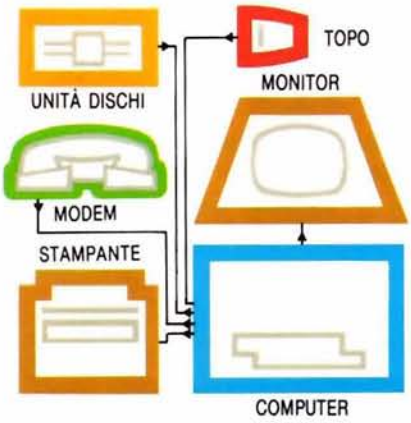
I GIOCHI

Con pochi accessori è possibile accedere al mondo dei giochi, che spesso non sono soltanto divertenti, ma anche utili alla comprensione della programmazione

LA DIFFUSIONE DEI MICRO COMPUTER

USA 12.600.000



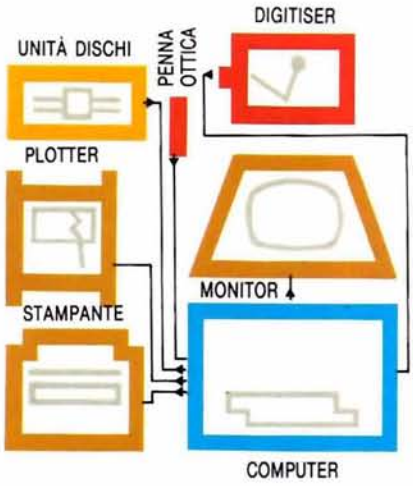


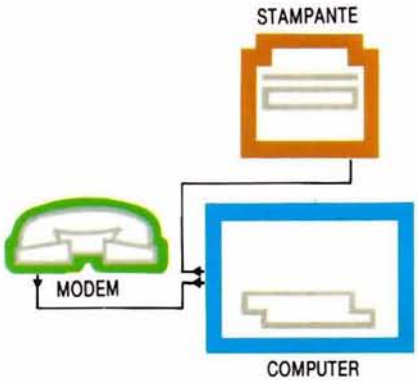
NEL LAVORO

Piccoli sistemi computerizzati aiutano — e rimpiazzano — impiegati e uomini d'affari in tutto il mondo. I **modem** consentono di allacciarsi — via telefono — ai grossi computer ed accedere ad una grande massa di informazioni. Il lavoro di dattilografia viene sostituito con i **word processor**, mentre l'uso di programmi di contabilità consente enormi risparmi rispetto alle procedure manuali, permettendo maggiore immediatezza a livello decisionale. Scaffali e scaffali di carte sono sostituiti dai dischetti magnetici, di minimo ingombro. Non sono che alcuni esempi dei cambiamenti in questo settore

LE COMUNICAZIONI

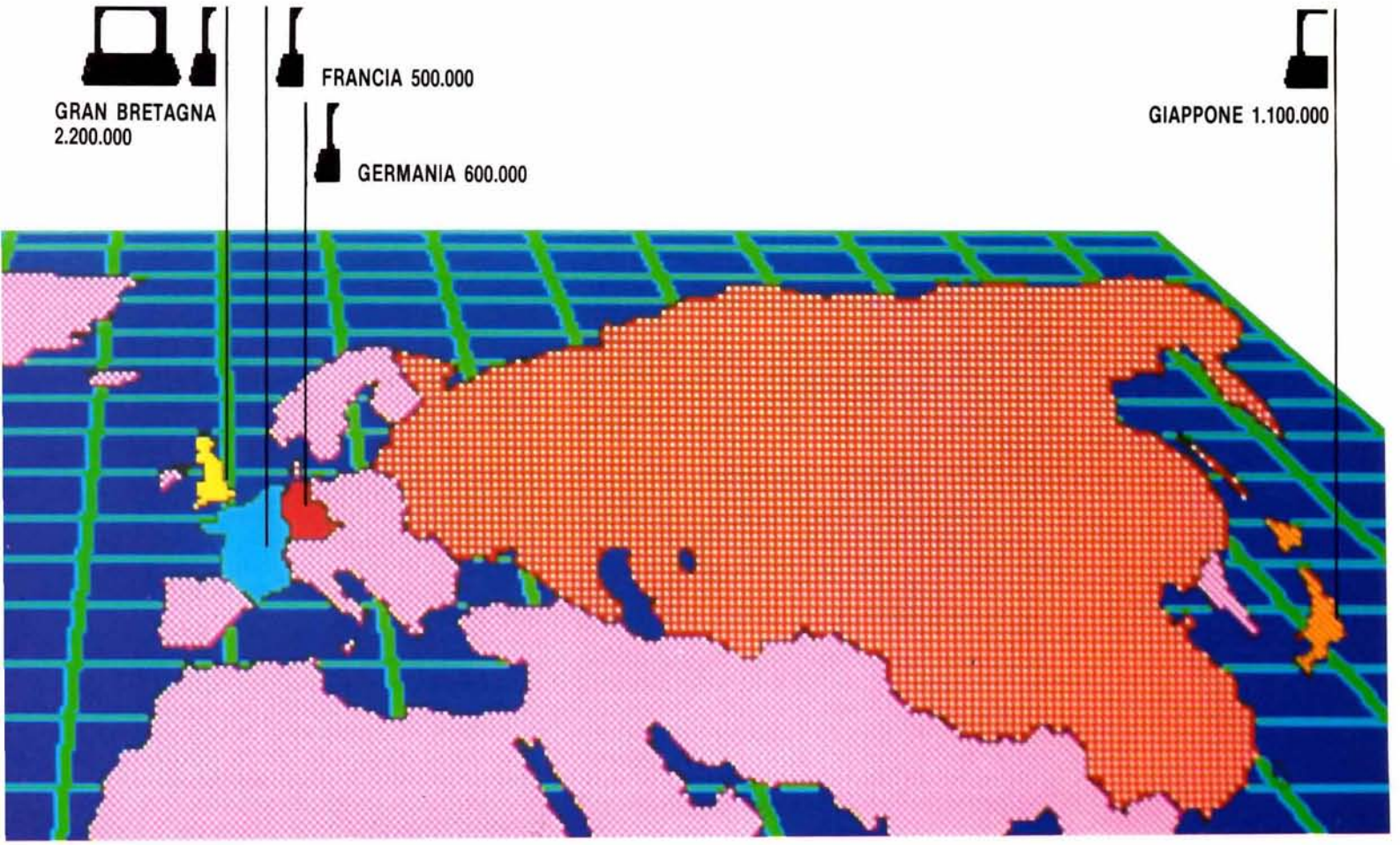
La rete telefonica ha ramificazioni in tutto il mondo. Grazie all'uso dei modem e dei computer — anche di quelli più piccoli — è possibile trasmettere o ricevere informazioni da e per ogni luogo in cui vi sia un telefono. Virtualmente è possibile, 24 ore al giorno, allacciare un piccolo computer ai "giganti". I giornalisti — per esempio — possono, da qualsiasi luogo, trasmettere il loro "pezzo" via modem alla propria redazione nel giro di pochi minuti





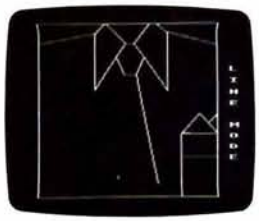
LA RICERCA

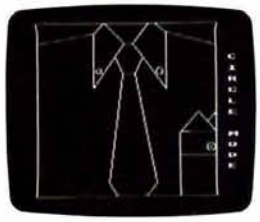
La potenza del computer non è più esclusivo appannaggio di laboratori universitari e di grosse organizzazioni commerciali: sistemi altamente sofisticati stanno entrando nelle scuole e nelle abitazioni private di tutto il mondo. Il CAD (Computer Aided Design — Progettazione al Computer) e l'AI (Artificial Intelligence — Intelligenza Artificiale), sono settori ai quali ci si può già dedicare comodamente seduti nel soggiorno di casa. **Plotter** (apparecchi per la riproduzione di disegni tecnici creati dal computer) e **digitiser** (strumenti per l'immissione dei dati relativi a figure complesse come mappe e diagrammi) hanno prezzi relativamente accessibili. L'Informatica sta divenendo una materia scolastica, ed i giovani di oggi stanno imparando a convivere con i computer di domani



L'artista elettronico

Semplici o sofisticate, le immagini al computer sono composte da migliaia di punti. Il computer deve ricordarne il colore e la luminosità

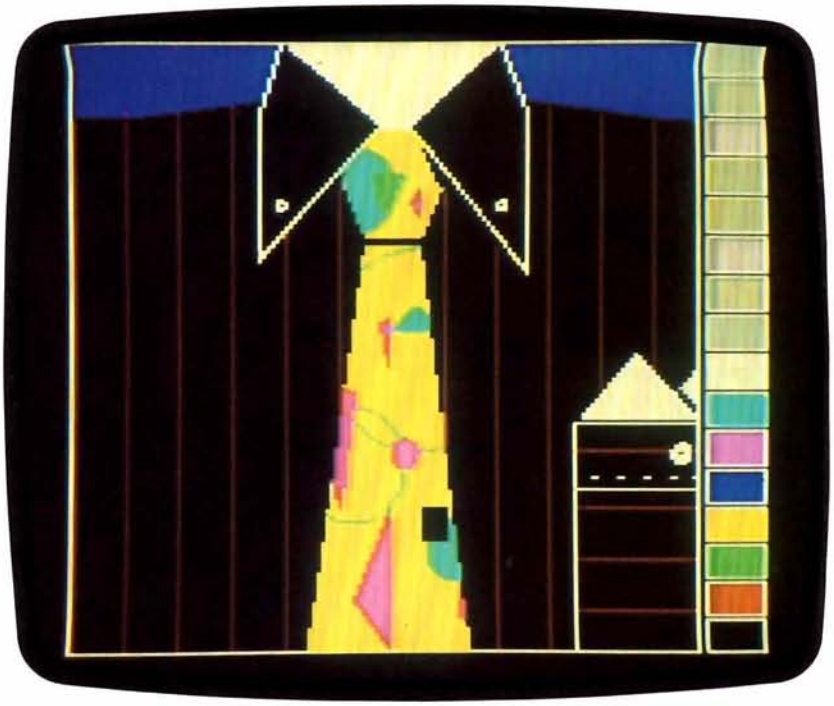






DIPINGERE COL MICRO

È facile dipingere col computer usando un buon software grafico. Alcuni programmi permettono la creazione di complesse figure mediante semplici comandi impartiti alla tastiera. Il primo passo consiste, in genere, nel disegnare le linee diritte. Poi possono venir aggiunti cerchi, triangoli, quadrati o altre figure predefinite. I colori che vengono usati per lo sfondo e per le linee del disegno si selezionano usando una "tavolozza" riprodotta sullo schermo o direttamente da tastiera. In alcuni casi il cursore può essere impiegato come pennello per colorare le diverse zone del disegno. Le figure qui accanto mostrano i vari passi nella creazione di un'immagine (è stato usato il programma Graphkey su di un BBC Modello B)



Il nostro aereo sta sfiorando i tetti delle case, un motore è fuori uso, stiamo volando dritti contro un grattacielo e la pista di atterraggio è ingombrata da un aereo in fiamme: è tutto qui, davanti ai nostri occhi, sullo schermo: una delle tante forme della grafica su computer.

Nei programmi didattici, quali quelli per insegnare la compitazione o l'aritmetica ai bambini, viene fatto largo uso della grafica per stimolare e mantenere l'interesse dell'allievo.

Nel campo del lavoro, chi utilizza un computer lo fa soprattutto per eseguire calcoli e visualizzare dei risultati in forma numerica, ma in alcuni casi anche questo tipo di informazioni può essere rappresentato più efficacemente usando la grafica (si pensi alla rappresentazione di un grafico della produzione).

Il computer ha la capacità di "ridisegnare" in breve tempo un diagramma che contenga nuovi dati (siano essi reali o di previsione) e con pari velocità è possibile ottenerne una copia su stampante. È facile immaginare quanto venga migliorata l'efficienza di un'azienda che sappia sfruttare bene un computer.

La tela del computer

«Come possono i computer creare immagini?» Per rispondere a questa domanda esaminiamo per un momento la "tela" adoperata dal computer. Un computer produce immagini sullo schermo illuminando (o "accendendo") uno o più punti in una o più posizioni sullo schermo. I punti sono organizzati in righe e colonne, cosicché — per poter individuare un qualsiasi punto — è sufficiente indicarne la riga e la colonna.

Una particolare figura è costruita illuminando certi punti e tenendo gli altri spenti, sia nel caso di schermi a colori che in quello di schermi in bianco e nero, (se lo schermo è a colori si deve attribuire anche un colore ad ogni punto).

Per visualizzare una lettera o un numero, il computer opera su una matrice di punti rettangolare o quadrata. La dimensione tipica è di otto righe di otto punti ciascuna.

Il numero di punti sullo schermo varia a seconda del computer, ed un valore tipico è di 192 righe di 256 punti ciascuna.

Per indicare il grado di definizione di un'immagine, viene impiegato il termine "risoluzio-

ne". Un computer che possa rappresentare sullo schermo 192 righe di 256 punti, possiede una risoluzione di 192×256.
Ovviamente, un alto valore di risoluzione, significando un maggior numero di punti sullo schermo, comporta una maggiore nitidezza dell'immagine.
Tutti i computer hanno un limite massimo di risoluzione, ma in alcuni esistono più livelli. Nel microcomputer BBC, ad esempio, si può lavorare in alta risoluzione (640 × 256 punti) od in bassa risoluzione (320 × 256 punti), purché ciò venga convenientemente indicato nel programma.
In apparecchi dotati di più livelli di risoluzione, è necessario inserire delle opportune istruzioni, in genere all'inizio del programma, per selezionare il modo grafico desiderato.
Le linee e le curve create da un computer su di una matrice di punti non sono effettivamente continue, come quelle disegnate da una penna. Sono, invece, dei percorsi che approssimano più o meno la linea o la curva, a seconda del grado di risoluzione. Un computer con bassa risoluzione visualizza le curve (e le linee diagonali) in modo segmentato. La visualizzazione avviene, infatti, illuminando il punto sullo schermo che più si avvicina al percorso della linea. Questa approssimazione produce, talvolta, un effetto chiamato "a gradino": la linea è tracciata con uno zig-zag più o meno marcato. Ancora una volta, più alta è la risoluzione e minore è l'effetto "gradino".
Vi è poi un'altra considerazione di carattere tecnico: l'immagine video deve essere "rinfrescata" (ripassata) continuamente, altrimenti essa apparirebbe per un solo istante e poi svanirebbe. Per questo motivo le informazioni concernenti l'immagine vengono conservate in una speciale zona di memoria chiamata "area video", da dove possono essere richiamate all'occorrenza. Con un video monocromo ad ogni punto dello schermo corrisponde un bit di memoria nell'area video ed il corrispondente punto viene "acceso" o "spento" a seconda che il bit abbia valore 1 o 0. Ne consegue che per un video di 256 × 192 punti occorrono 49.152 bit, ovvero 6 Kbyte dato che 1Kbyte è fatto di 8,192 bit.
Nel caso di un video a colori, la faccenda si complica, perché la quantità di memoria necessaria aumenta proporzionalmente al numero di colori che si intendono usare ed è per questa ragione che alcuni dei computer più economici, oltre ad offrire una grafica a bassa (o tutt'al più a media) risoluzione, non consentono di usare una gamma molto estesa di colori.
Vediamo come è possibile codificare le informazioni necessarie per rappresentare un punto colorato sullo schermo, avendo a disposizione due soli bit. Le combinazioni che si possono formare sono:

bit 1	bit 2	colore
0	0	bianco
0	1	rosso
1	0	blu
1	1	nero

Per visualizzare un'immagine a quattro colori, sono dunque necessari due bit per ogni punto dello schermo. Similmente, per otto colori ci vorranno 3 bit, mentre per 16 colori ne servono 4. Un computer con una grafica a 16 colori ed una risoluzione di 160 × 256 usa, quindi, un'area video di 160 × 256 × 4 bit, ossia 160 Kbit (20 Kbyte).

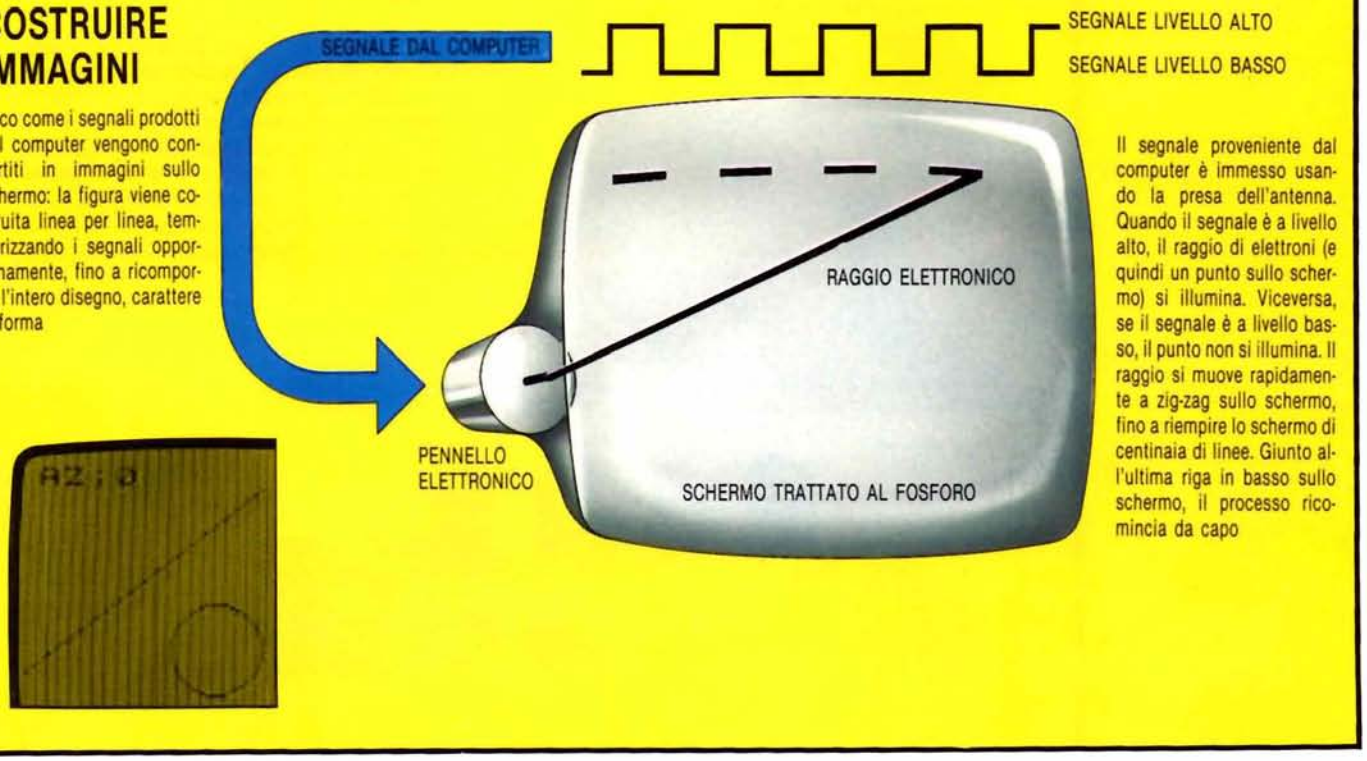


CREAZIONI GRAFICHE

Costruire immagini su di un personal può essere molto facile. Gran parte dei personal ha comandi speciali in BASIC per disegnare linee o per definire "oggetti" sullo schermo, e su alcuni, con opportuni programmi, è addirittura possibile creare immagini d'animazione

COSTRUIRE IMMAGINI

Ecco come i segnali prodotti dal computer vengono convertiti in immagini sullo schermo: la figura viene costruita linea per linea, temporizzando i segnali opportunamente, fino a ricomporre l'intero disegno, carattere o forma



Il segnale proveniente dal computer è immesso usando la presa dell'antenna. Quando il segnale è a livello alto, il raggio di elettroni (e quindi un punto sullo schermo) si illumina. Viceversa, se il segnale è a livello basso, il punto non si illumina. Il raggio si muove rapidamente a zig-zag sullo schermo, fino a riempire lo schermo di centinaia di linee. Giunto all'ultima riga in basso sullo schermo, il processo ricomincia da capo

Bit e byte

Il computer capisce solo numeri, ma numeri dall'aspetto poco familiare

Quando si parla di computer, immancabilmente compaiono le due parole **bit** e **byte**. Questi due termini indicano la forma, molto diversa da quella per noi abituale, usata dal computer per conservare ed elaborare i numeri.

Noi rappresentiamo i numeri impiegando 10 diversi simboli (da 0 a 9) ed eseguiamo calcoli sui multipli e sottomultipli di 10. Questo è noto col nome di sistema decimale, ovvero con "base" 10. I computer, invece, usano soltanto due numeri (lo zero e l'uno) per tutte le operazioni matematiche. I bit ed i byte sono modi di rappresentare le combinazioni di questi due numeri.

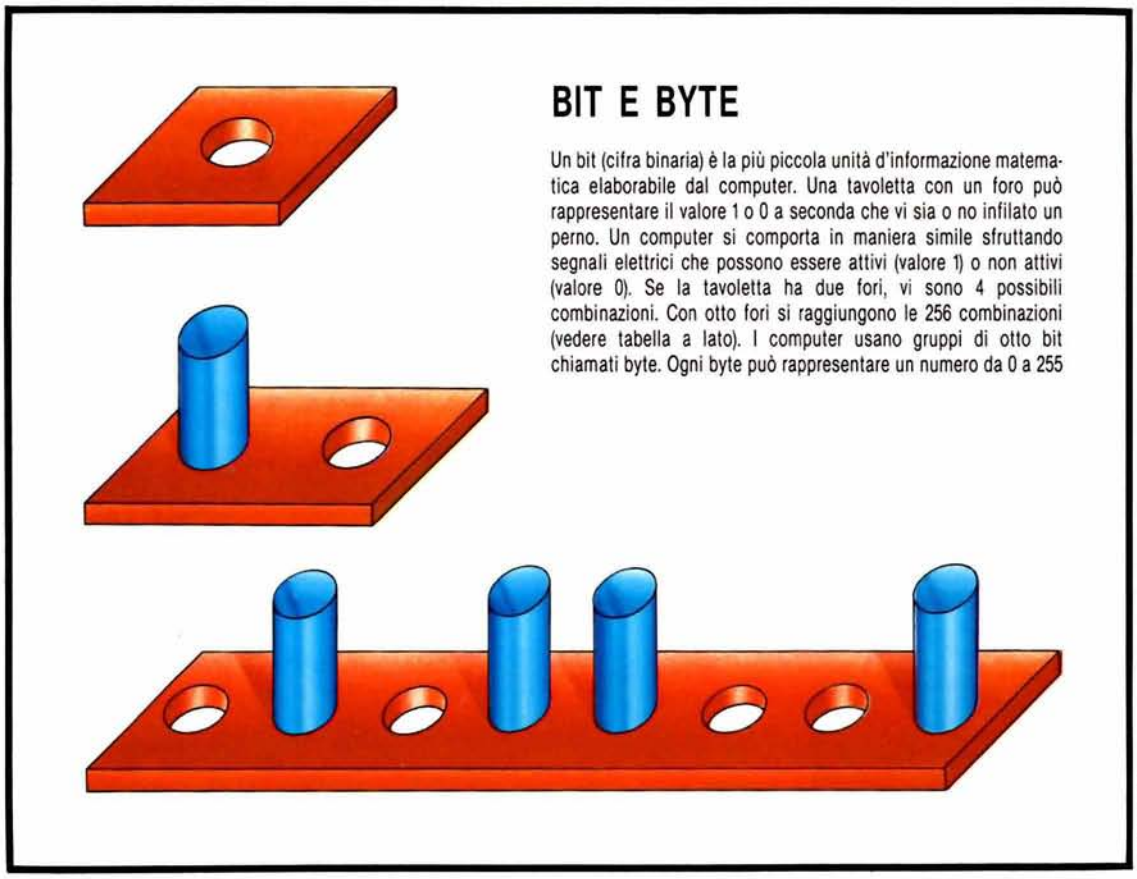
La più piccola informazione elaborabile dal computer è il bit, che può avere valore 0 o valore 1. Il byte non è che un gruppo di otto bit, e con esso è possibile rappresentare numeri di media grandezza.

Esaminiamo per primi i bit: cosa sono, e da cosa deriva il loro nome. I computer sono dei congegni elettronici, pertanto tutto ciò che essi trattano deve essere ricondotto, in ultima analisi, a dei segnali elettrici. Un singolo segnale elettrico può avere due "stati": può essere attivo (**on**) o non attivo (**off**): grazie a questo principio è possibile rappresentare numeri usando segnali elettrici.

Per meglio comprendere il concetto, osserviamo la prima illustrazione: una tavoletta munita di un foro, nel quale si può introdurre un perno. Sebbene vi sia un solo foro, è possibile rappresentare due numeri: se il foro è vuoto il valore è zero, se nel foro viene fissato il perno, il valore è uno. Analoga è la situazione nel computer, dove al posto della tavoletta viene usato un segnale elettrico: il segnale non attivo indica valore zero, quello attivo indica il valore uno. Un solo filo elettrico (o la tavoletta dell'esempio) può avere due stati: off/on, spento/acceso, senza perno/con perno, zero/uno.

Questa piccola unità d'informazione si chiama bit (letteralmente: pezzettino). Il termine deriva da **BI**nary digi**T**. Con il bit è possibile contare, ma solo da zero a uno.

Ritornando all'esempio, una tavoletta con due fori può indicare quattro diversi stati, ovvero i numeri da 0 a 3: i due fori vuoti, il



BIT E BYTE

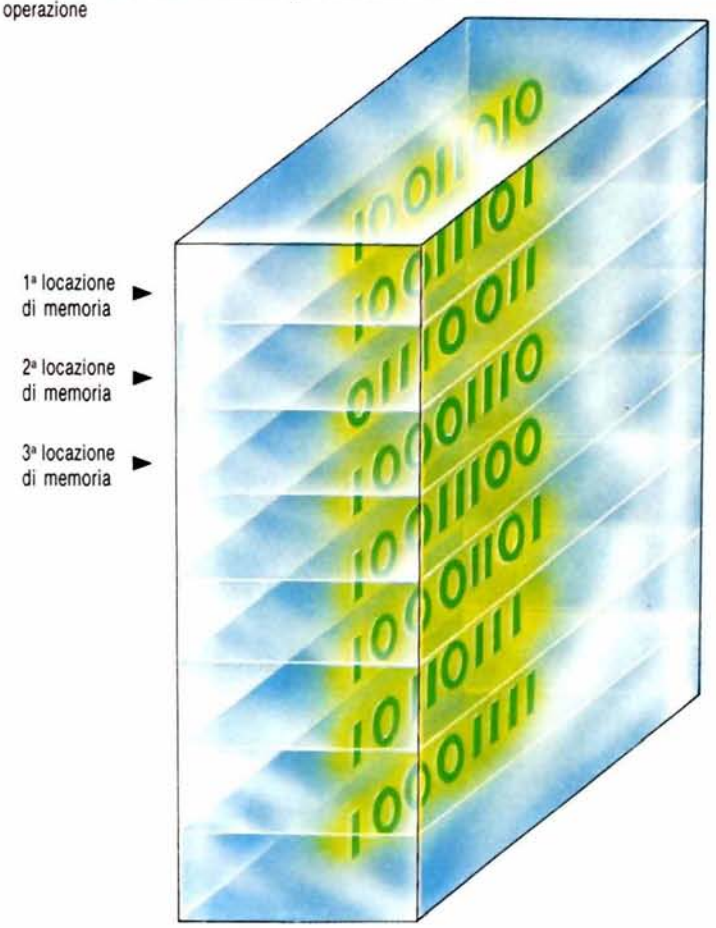
Un bit (cifra binaria) è la più piccola unità d'informazione matematica elaborabile dal computer. Una tavoletta con un foro può rappresentare il valore 1 o 0 a seconda che vi sia o no infilato un perno. Un computer si comporta in maniera simile sfruttando segnali elettrici che possono essere attivi (valore 1) o non attivi (valore 0). Se la tavoletta ha due fori, vi sono 4 possibili combinazioni. Con otto fori si raggiungono le 256 combinazioni (vedere tabella a lato). I computer usano gruppi di otto bit chiamati byte. Ogni byte può rappresentare un numero da 0 a 255

TONY LODGE

0	00000000	128	10000000
1	00000001	129	10000001
2	00000010	130	10000010
3	00000011	131	10000011
4	00000100	132	10000100
5	00000101	133	10000101
6	00000110	134	10000110
7	00000111	135	10000111
8	00001000	136	10001000
9	00001001	137	10001001
10	00001010	138	10001010
11	00001011	139	10001011
12	00001100	140	10001100
13	00001101	141	10001101
14	00001110	142	10001110
15	00001111	143	10001111
16	00010000	144	10010000
17	00010001	145	10010001
18	00010010	146	10010010
19	00010011	147	10010011
20	00010100	148	10010100
21	00010101	149	10010101
22	00010110	150	10010110
23	00010111	151	10010111
24	00011000	152	10011000
25	00011001	153	10011001
26	00011010	154	10011010
27	00011011	155	10011011
28	00011100	156	10011100
29	00011101	157	10011101
30	00011110	158	10011110
31	00011111	159	10011111
32	00100000	160	10100000
33	00100001	161	10100001
34	00100010	162	10100010
35	00100011	163	10100011
36	00100100	164	10100100
37	00100101	165	10100101
38	00100110	166	10100110
39	00100111	167	10100111
40	00101000	168	10101000
41	00101001	169	10101001
42	00101010	170	10101010
43	00101011	171	10101011
44	00101100	172	10101100
45	00101101	173	10101101
46	00101110	174	10101110
47	00101111	175	10101111
48	00110000	176	10110000
49	00110001	177	10110001
50	00110010	178	10110010
51	00110011	179	10110011
52	00110100	180	10110100
53	00110101	181	10110101
54	00110110	182	10110110
55	00110111	183	10110111
56	00111000	184	10111000
57	00111001	185	10111001
58	00111010	186	10111010
59	00111011	187	10111011
60	00111100	188	10111100
61	00111101	189	10111101
62	00111110	190	10111110
63	00111111	191	10111111
64	01000000	192	11000000
65	01000001	193	11000001
66	01000010	194	11000010
67	01000011	195	11000011
68	01000100	196	11000100
69	01000101	197	11000101
70	01000110	198	11000110
71	01000111	199	11000111
72	01001000	200	11001000
73	01001001	201	11001001
74	01001010	202	11001010
75	01001011	203	11001011
76	01001100	204	11001100
77	01001101	205	11001101
78	01001110	206	11001110
79	01001111	207	11001111
80	01010000	208	11010000
81	01010001	209	11010001
82	01010010	210	11010010
83	01010011	211	11010011
84	01010100	212	11010100
85	01010101	213	11010101
86	01010110	214	11010110
87	01010111	215	11010111
88	01011000	216	11011000
89	01011001	217	11011001
90	01011010	218	11011010
91	01011011	219	11011011
92	01011100	220	11011100
93	01011101	221	11011101
94	01011110	222	11011110
95	01011111	223	11011111
96	01100000	224	11100000
97	01100001	225	11100001
98	01100010	226	11100010
99	01100011	227	11100011
100	01100100	228	11100100
101	01100101	229	11100101
102	01100110	230	11100110
103	01100111	231	11100111
104	01101000	232	11101000
105	01101001	233	11101001
106	01101010	234	11101010
107	01101011	235	11101011
108	01101100	236	11101100
109	01101101	237	11101101
110	01101110	238	11101110
111	01101111	239	11101111
112	01110000	240	11110000
113	01110001	241	11110001
114	01110010	242	11110010
115	01110011	243	11110011
116	01110100	244	11110100
117	01110101	245	11110101
118	01110110	246	11110110
119	01110111	247	11110111
120	01111000	248	11111000
121	01111001	249	11111001
122	01111010	250	11111010
123	01111011	251	11111011
124	01111100	252	11111100
125	01111101	253	11111101
126	01111110	254	11111110
127	01111111	255	11111111

I BYTE NELLA MEMORIA

I byte sono gruppi di otto cifre binarie (bit). Ogni byte viene usato dal computer per rappresentare un numero da 0 a 255. Ogni byte viene conservato in "celle" individuali di memoria, organizzate sistematicamente, in modo che sia facile individuare la collocazione della "cella" dove si trova il byte necessario ad una certa operazione



TONY LODGE

foro a destra vuoto e l'altro pieno, il foro a destra pieno e l'altro vuoto, ambedue i fori pieni. La figura più in basso mostra una tavoletta con otto fori. Esistono ben 256 diverse permutazioni di fori vuoti/pieni (come si può vedere nella tabella a lato) in cui il foro vuoto è indicato con zero e il pieno con uno. Questi gruppi di otto cifre binarie (bit) vengono chiamati byte. Un singolo byte può quindi rappresentare 256 stati diversi (ossia i numeri da 0 a 255).

Quando si dice che un computer "memorizza" un byte, significa che un numero (tra 0 e 255) viene conservato nella memoria del computer, per essere successivamente riutilizzato. Ogni byte è come in una piccola "scatoletta", e tutte queste scatolette sono collocate una di seguito all'altra. Un'idea di questo tipo di organizzazione viene data dall'ultima illustrazione. Quando è necessario recuperare un particolare numero da una "scatoletta", al computer è sufficiente sapere in quale delle scatolette andare a cercare. Di questo parleremo nella prossima lezione.

Tutti i numeri da 0 a 255 possono essere rappresentati usando una combinazione di 0 ed 1, come si nota dalla tabella. I bit vengono conservati e ricercati dal computer a gruppi di otto. Otto bit costituiscono un byte

L'insegnante elettronico

Anche i più giovani vogliono usare il computer. Aiutiamoli così!



IAN McKINNELL

Uno degli accessori didattici più interessanti è la "Tartaruga": un piccolo robot collegato al micro computer ed azionato mediante un linguaggio chiamato LOGO. La Tartaruga, utile per insegnare, divertendo, concetti geometrici elementari (distanze, direzioni, dimensioni), viene anche usata per eseguire complicati disegni

Molti genitori si chiedono se un computer può essere utile ai loro figli, pur sapendo che è un'ottima idea far imparare ai giovani come si usano questi apparecchi.
La risposta è senz'altro affermativa, ma occorre scegliere il metodo adatto per introdurre un bambino ai concetti dell'Informatica.
In Gran Bretagna, per esempio, il Governo ha stabilito che l'uso dei computer venga insegnato fin dalle scuole elementari. Il progetto *Micros in Primary Education* prevede che ogni scuola sia in possesso di almeno un microcomputer.
In Italia è in corso un progetto per raggiungere obiettivi simili.
Gli insegnanti avranno il compito di utilizzare convenientemente le possibilità offerte dal computer.
I computer non sono solo bravi in matematica: per mezzo di buoni programmi (di cui attualmente vi è una certa scarsità) si possono insegnare ai giovanissimi materie quali la musica, la geografia, le lingue straniere oltre, naturalmente, quelle basate sulla matematica e sulla geometria.
Esistono due tipi di utilizzazione in questo campo: l'allievo può usare il computer per esplorare l'ambiente circostante oppure adoperarlo come sostituto dell'insegnante in esercitazioni sulle varie materie.
Alcuni bambini riescono a scrivere programmi in BASIC anche a nove anni, ma gli studi compiuti dallo psicologo francese Jean Piaget hanno dimostrato che prima dei 12/13 anni i bambini hanno grosse difficoltà ad apprendere concetti astratti, di cui il linguaggio BASIC abbonda.
I tentativi di insegnare il BASIC a bambini di età inferiore ai 12 anni, quindi, non hanno in genere molto successo.

I ricercatori, tenendo conto di questo ostacolo, hanno studiato un nuovo metodo per introdurre i giovanissimi all'uso del computer basandolo su concetti che sono molto più accessibili e meno astratti (vedere il riquadro sul LOGO).

La tartaruga

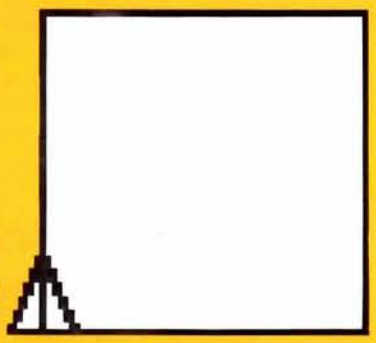
Anche i bambini più piccoli possono imparare per mezzo del computer.
La foto a lato mostra un ragazzo alle prese con una Tartaruga (così viene chiamato quella specie di piccolo robot collegato al computer).
Il principio di funzionamento della Tartaruga non è molto complicato: mediante semplici istruzioni impartite al computer, si ottengono gli spostamenti della Tartaruga nelle direzioni desiderate, con la possibilità di lasciare una "traccia", grazie alla penna incorporata nell'apparecchio. Ciò consente di generare disegni che possono essere anche molto complessi.
Usando la Tartaruga, un bambino impara non solo ad usare un computer, ma anche i primi elementi di geometria, poiché è incoraggiato a "programmare" le mosse della Tartaruga per ottenere forme specifiche. Questo metodo di "auto-istruzione" sta alla base del linguaggio LOGO e si fonda sul principio che una lezione impartita "euristicamente" (ovvero mediante prove e verifiche) è appresa più efficacemente se accompagnata da esempi pratici.
Vi sono, dunque, due diversi metodi per avvicinare i giovanissimi al computer. Con il LOGO, i ragazzi di otto/nove anni, cominciano ad usare una versione "su video" della Tartaruga, disegnando figure complesse ed "insegnando" alla Tartaruga le più diverse procedure.
Quando un bambino "insegna" alla Tartaruga come muoversi, in effetti egli sta programmando il computer. Il LOGO è un linguaggio che permette anche a bambini molto piccoli di programmare ancor prima che siano in grado di apprendere i concetti astratti contenuti negli altri linguaggi per computer. Il "gioco della Tartaruga" consente ai più piccoli un avvicinamento alla padronanza del computer e li aiuta ad esplorare l'ambiente circostante.
L'altro metodo si fonda invece sull'apparente "pazienza" del computer nell'insegnare attraverso esempi.
Bambini che trovano difficoltà nell'apprendere una materia, o un particolare aspetto di essa, sono spesso aiutati da programmi di "spiegazione e verifica". In sostanza, questi programmi pongono delle domande ed assegnano un punteggio in base alle risposte date dal bambino. Molti di questi programmi sono resi molto affascinanti ed interessanti da un brillante impiego sia della grafica che di effetti sonori. Alcuni programmi, per esempio, chiedono di indicare quale, tra un gruppo di parole, sia un avverbio oppure chiedono di formare parole usando un certo gruppo di lettere. I bambini vengono incoraggiati ad imparare, usando questi programmi: il computer non si spazientisce mai, anche insistendo a fornire risposte errate. Ciò è ancora più apprezzabile quando il bambino ha un ritmo di apprendimento piuttosto lento. Questo secondo metodo potrebbe portare alla errata conclusione che il computer può sostituire l'insegnante. Ciò non è affatto possibile, poiché il contatto umano è un elemento fondamentale nell'insegnamento e se è vero da un lato che il computer è uno strumento didattico ineguagliabile, dall'altro non esistono surrogati ad un insegnamento ben curato.
In prospettiva, la scelta di un computer da usare in casa, dovrebbe cadere sullo stesso modello che viene adottato dalla scuola, perché ciò consentirebbe ai nostri figli una omogeneità nell'uso dell'apparecchio e dei programmi, creando un saldo legame tra scuola e casa.
Il computer può anche essere usato per il divertimento, e non dovremmo impedire che i

LA LOGICA DEL LOGO

Vediamo come costruire immagini sullo schermo usando il linguaggio LOGO.
Il LOGO è un linguaggio appositamente studiato per insegnare ai giovanissimi (addirittura di quattro/cinque anni) come si programma un computer. Il programma è stato sviluppato negli anni '60 da Seymour Papert, un matematico del Massachusetts Institute of Technology che ha collaborato col pedagogista Jean Piaget.
Scopo del programma, almeno per i più piccoli, è quello di creare disegni, sia sullo schermo del computer che su grandi fogli di carta, impartendo semplici comandi ad una "Tartaruga". Il comando FORWARD 10 (AVANTI 10) indica alla tartaruga di spostarsi in avanti di 10 unità, tracciando una linea dietro di sé. Il comando RIGHT 90 (DESTRA 90) causa una rotazione di 90 gradi a destra. I comandi possono essere raggruppati in sequenze che permettono di tracciare triangoli, quadrati, cerchi ed altre figure più complesse. Si può anche fare in modo che la tartaruga "impari" una certa sequenza di comandi. Senza rendersene conto, i bambini imparano così a programmare il computer

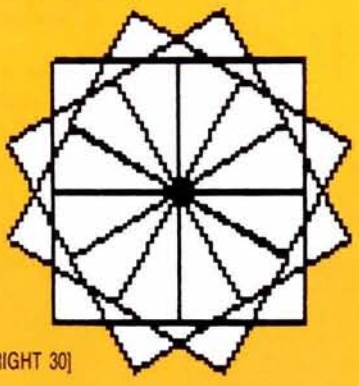
Disegnamo un quadrato:

```
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
RIGHT 90
FORWARD 50
```



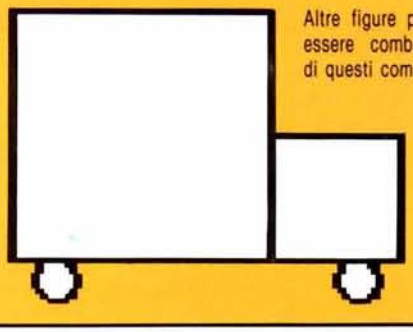
Costruiamo il comando di nome BOX per disegnare il quadrato:

```
TO BOX
REPEAT 4 [FORWARD 50 RIGHT 90]
END
```



Il comando STAR per creare una stella:

```
TO STAR
REPEAT 12 [BOX RIGHT 30]
END
```



Altre figure possono essere combinazioni di questi comandi

nostri figli lo adoperino per giocare: anche questo aspetto conduce alla familiarizzazione con l'apparecchio e con la programmazione.
I bambini di età inferiore ai sette/otto anni in genere devono essere aiutati per le operazioni di accensione del computer e delle periferiche e per far partire i programmi. Dopo di che, il bambino, in base alla sua capacità di lettura e di rispondere alle domande, può anche essere lasciato a se stesso.
I requisiti delle apparecchiature, per questo tipo di applicazioni, sono pochi, ma vale la pena di riassumerli. Innanzi tutto, il contenitore del micro computer deve essere robusto, perché spesso i bambini sottopongono le parti a bruschi scossoni e ad ogni sorta di "collaudo". Poi, il sistema deve essere facile da accendere, da collegare e da usare, pena una perdita completa d'interesse. Anche se, secondo alcuni esperti, andrebbero usate tastiere particolari, con lettere e simboli di grandezza superiore al normale, la tendenza è di impiegare tastiere normali, anche per evitare il successivo insorgere di problemi di adattamento.
Le tastiere del tipo "a sfioramento", quale quella del Sinclair ZX81, non sono particolarmente adatte per bambini di età inferiore ai nove o dieci anni, sebbene quella adottata dal Videopac 7000 della Philips può comodamente essere usata da bambini di quattro anni.

Giovani programmatori

Le difficoltà maggiori si incontrano nella scelta di programmi adatti ai giovanissimi. Programmi conservati su "musicassette" implicano un'assistenza al momento di caricare il programma, mentre è più raro che i bambini trovino difficoltà a maneggiare dei dischetti.
Ideale è l'uso di cartucce ROM, dove il programma è conservato in speciali memorie all'interno di un solido contenitore di plastica. Lo svantaggio causato dal fatto che le cartucce si possono soltanto "leggere" (non si possono aggiungere altri dati) si compensa con la facilità del loro uso e con la loro robustezza.
Un altro punto importante, se il computer verrà usato in casa, è la sua collocazione. In genere l'apparecchio non soffre alcun danno nell'essere trasferito da una stanza all'altra, ma se la messa in funzione implica ogni volta una serie di operazioni complicate per il bambino, egli perderà interesse per il computer, preferendo, magari, guardare la televisione.

La collocazione ideale

La soluzione migliore sarebbe collocare il computer, completo di tutti gli accessori, nella camera del bambino. Ciò può implicare l'acquisto di un secondo televisore, che può benissimo essere in bianco e nero. L'uso di un video a colori, infatti, oltre a tradursi in un costo non indifferente, viene considerato da gran parte degli esperti un sovrappiù. È ovvio che una collocazione permanente, che permette l'uso immediato del computer, è preferibile a qualsiasi altra soluzione transitoria.
Una volta stabilito dove tenere il computer, è opportuno fissare tutti i cavi di connessione in modo da offrire la massima sicurezza, ma al tempo stesso consentire lo spostamento degli apparecchi altrove. Può essere utile, a questo fine, acquistare due serie di cavi: una verrà tenuta fissa nella "postazione", l'altra impiegata quando si voglia usare il computer in qualche altro luogo. Inoltre, per assicurarsi che il computer sia stabile sul piano di lavoro, e per evitare danni alle parti delicate, è possibile trovare in commercio degli speciali supporti, adatti ai micro computer più leggeri (come ad esempio quelli della Sinclair). Occasionalmente, si potranno tenere divertenti ed istruttive sessioni di programmazione con la partecipazione di tutta la famiglia.

LA VISTA È IN PERICOLO?

Molti adulti, che lavorano continuamente, per ore e ore, allo schermo di un computer possono soffrire di un affaticamento alla vista. Un problema comune, in questi casi, è una certa difficoltà dell'occhio a focalizzare immagini poste a distanze diverse. I bambini, in genere, non restano mai così a lungo davanti all'apparecchio da avere questa sorta di problemi. Una buona regola, comunque, è di interrompere il lavoro al computer per qualche minuto ogni quarto d'ora. Un altro problema può sorgere da vecchi apparecchi TV a colori; alcuni emettono delle radiazioni che potrebbero essere nocive a poca distanza dagli occhi. Negli apparecchi costruiti dopo il 1970 il problema non sussiste

IL BIG TRAK

Per quanto assomigli ad un carrarmato giocattolo, il Big Trak è in effetti un interessante strumento didattico, basato sull'impiego di un microprocessore programmabile. Un bambino può fargli memorizzare un preciso percorso che può avere fino a 16 variazioni. Il carro può quindi entrare ed uscire da più ambienti, prima di ritornare alla "base". Per i bambini è un gioco, ma il computer li aiuta gradualmente ad abituarsi al concetto della programmazione



Il Commodore 64

Anatomia di microcomputer di diffusione europea: tecnologie a confronto

Il Commodore 64 è il primo di una nuova generazione di microcomputer, a cavallo tra l'utilizzazione domestica e quella commerciale. I 64 Kbyte di memoria standard sono più che sufficienti per gestire immagini grafiche di notevole complessità, e per l'impiego di "lavagne elettroniche", database, word-processor. Esistendo una certa compatibilità con i sistemi Commodore VIC-20 e PET, la scelta dei programmi disponibili è ancora più ampia.
La gamma di interfacce collegabili al Commodore 64 è tale da consentire l'uso di un grande numero di periferiche, ivi incluse la stampante e le unità a dischi per il VIC-20. Un sistema completo di computer, unità disco e stampante, può essere acquistato ad un prezzo veramente competitivo.
Due delle caratteristiche più interessanti del Commodore 64 sono: l'ottima grafica a "sprite" (vedere pag. 45) ed un sintetizzatore musicale completo, incorporato. Gli sprite sono immagini create sullo schermo con la grafica ad alta risoluzione. La loro posizione ed il loro movimento vengono modificati dal programma mediante semplici comandi, facilitando, in particolare, la gestione grafica di giochi. I particolari effetti della grafica "sprite" si potrebbero ottenere anche con una grafica normale senza sprite, ma a costo di una programmazione molto più complessa. Gli sprite sul Commodore 64 si possono muovere anche su piani diversi rispetto ad uno scenario fisso sullo sfondo o rispetto ad immagini situate su piani diversi, creando così una profondità alle immagini ed una verosimile illusione di tridimensionalità.
La sintetizzazione musicale è molto più sofisticata, rispetto agli sterili 'bip' ottenibili su apparecchi più economici: non solo sono disponibili tre voci (si possono quindi formare accordi ed armonie), ma è anche possibile controllare una serie di parametri che influiscono sul timbro e sul tono delle note generate. In altre parole il Commodore 64 è in grado di simulare una gran varietà di strumenti musicali, oltre a suoni del tutto nuovi.
Il linguaggio BASIC disponibile rappresenta un buono standard sia per il programmatore dilettante che per quello serio. Grazie alle istruzioni POKE e PEEK, è possibile sfruttare appieno le capacità grafiche e sonore del computer. Sono disponibili altri tipi di BASIC, appositamente studiati per l'impiego della grafica su questo ottimo apparecchio.



Il Commodore 64 ha un'eccellente tastiera, i cui tasti, modellati anatomicamente, ne facilitano l'uso. Oltre ai caratteri normali, sono disponibili numerosi caratteri grafici, selezionabili mediante un apposito tasto, contrassegnato col marchio Commodore. La selezione dei colori avviene premendo contemporaneamente il tasto CTRL (CONTROL) ed uno dei tasti della fila più alta. Ai quattro tasti collocati a destra della tastiera è possibile assegnare funzioni speciali mediante istruzioni di programma



1530 C2N Registratore a cassette

Questo è il registratore, molto affidabile, della Commodore. Si collega all'apposito connettore del computer. L'alimentazione è fornita dallo stesso computer, che può anche accendere e spegnere il motore



1540 Unità a dischi

Questa unità a dischi viene collegata al connettore seriale del Commodore. Un dischetto da 5 1/4 pollici offre 170 Kbyte di memoria di massa. Il tempo massimo di accesso è di 2 secondi

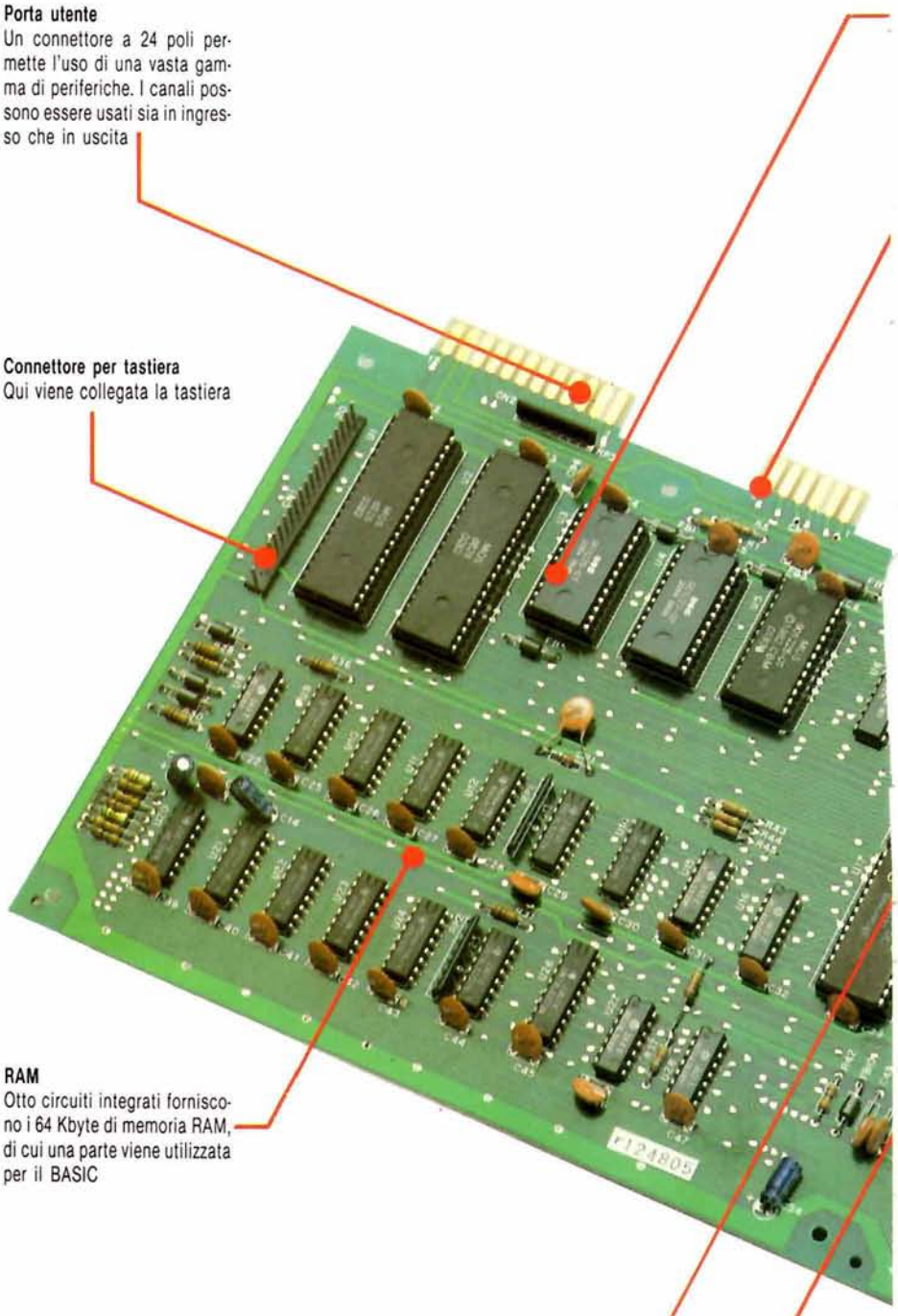


1525 Stampante

Questa stampante a matrice di punti è in grado di riprodurre tutti i caratteri generati dal Commodore 64 (inclusi quelli grafici), ad una velocità di 30 caratteri al secondo. Si collega al connettore seriale

Porta utente
Un connettore a 24 poli permette l'uso di una vasta gamma di periferiche. I canali possono essere usati sia in ingresso che in uscita

Connettore per tastiera
Qui viene collegata la tastiera



RAM
Otto circuiti integrati forniscono i 64 Kbyte di memoria RAM, di cui una parte viene utilizzata per il BASIC

Microprocessore
Il microprocessore impiegato è il 6510, che controlla sia tutte le principali operazioni di elaborazione sia parte delle operazioni di input/output

Sintetizzatore
Il chip 6581 è il circuito che genera gli effetti sonori del Commodore 64

ROM del BASIC
Questi tre chip contengono il linguaggio BASIC

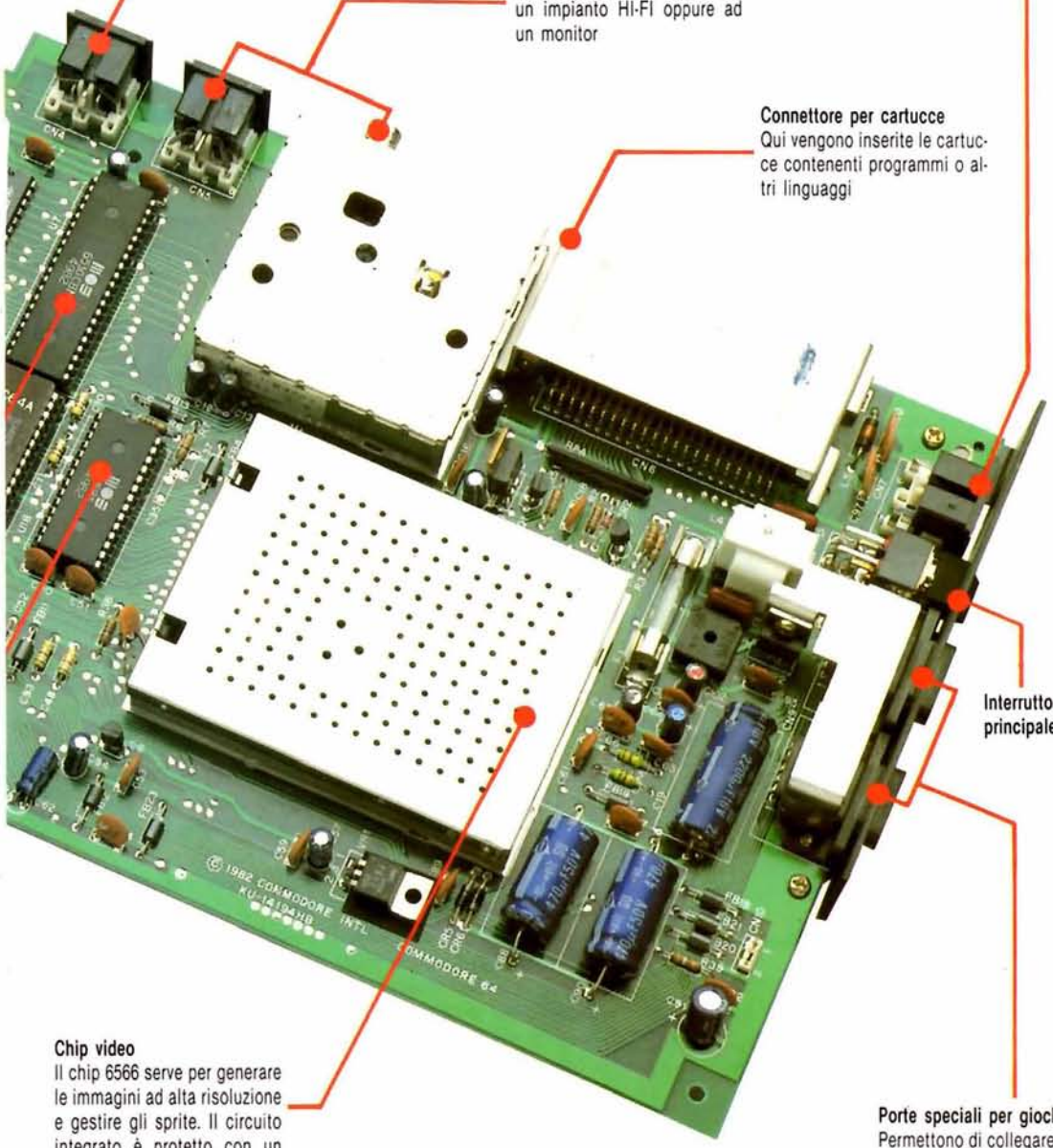
Presa per alimentazione
Qui viene collegato lo spinotto dell'alimentatore

Connettore per registratore
Qui viene collegato il registratore a cassette della Commodore. Altri tipi di registratori potrebbero venir collegati, purché opportunamente modificati

Porta seriale
Questo è il connettore per la comunicazione seriale con altre apparecchiature. I dati sono trasmessi un bit alla volta

Connettore audio/video
Il computer si può collegare ad un impianto HI-FI oppure ad un monitor

Connettore per cartucce
Qui vengono inserite le cartucce contenenti programmi o altri linguaggi



Interruttore principale

Chip video
Il chip 6566 serve per generare le immagini ad alta risoluzione e gestire gli sprite. Il circuito integrato è protetto con un dissipatore di calore

Porte speciali per giochi
Permettono di collegare penne ottiche, joystick o altri accessori per giochi

COMMODORE 64

DIMENSIONI

mm. 404 x 216 x 75

PESO

1820 grammi

CLOCK

1 MHz

MEMORIA

64 Kbyte totali. 20 Kbyte di ROM contengono il sistema operativo ed il BASIC; RAM disponibile all'utente 54 Kbyte, se non viene usato il BASIC

VIDEO

25 linee di 40 caratteri o 320 x 200 punti per la grafica ad alta risoluzione. 16 colori disponibili

CPU

6510

TASTIERA

Tipo macchina da scrivere, QWERTY, con 62 tasti più 4 tasti per funzioni speciali

INTERFACCE

Vari connettori: per TV, per monitor, per registratore a cassette, per cartucce. Interfaccia RS232. Porta utente

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

FORTH, COMAL, LOGO, PILOT, UCSD PASCAL, e altri BASIC

ACCESSORI COMPRESI

Alimentatore, cavi di connessione per TV, un manuale

DOCUMENTAZIONE

Il manuale contiene una "Guida all'uso" abbastanza chiara. Viene descritto l'apparecchio, il funzionamento ed il linguaggio. Alcuni punti potrebbero risultare leggermente difficili per il principiante, ma in linea di massima si tratta di un buon manuale. Per approfondire la conoscenza sull'uso della grafica e degli effetti sonori, sono disponibili alcune pubblicazioni supplementari, alcune edite dalla stessa Commodore

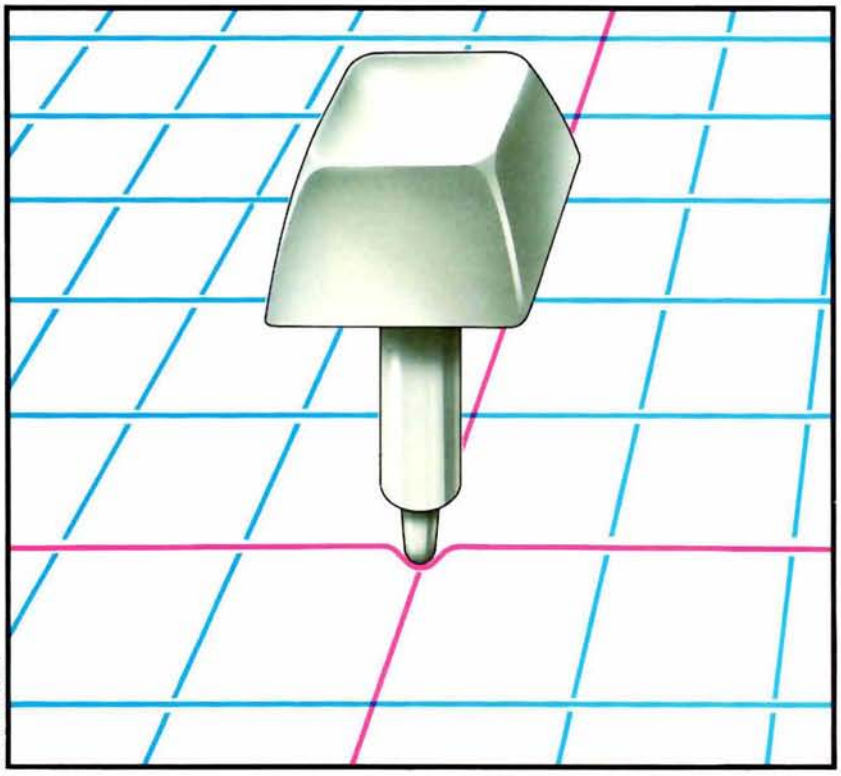
Il tasto giusto

Tutte le tastiere sembrano uguali ma l'apparenza può ingannare

La tastiera del computer è una parte molto importante del sistema. È attraverso la tastiera, infatti, che possiamo comunicare con il computer. La scelta della tastiera riveste tanta importanza quanto la scelta delle capacità grafiche o della quantità di memoria.
La disposizione dei tasti, nelle tastiere per computer, è in genere del tutto simile a quella delle macchine da scrivere e passa sotto il nome di QWERTY (le prime 6 lettere della tastiera).
Nelle prime macchine da scrivere, all'inizio di questo secolo, venne scelta questa disposizione addirittura con l'intento di frenare una battitura troppo veloce, che avrebbe potuto rovinare i delicati meccanismi delle macchine.
Così, quando negli anni '50 apparvero i primi computer, la disposizione QWERTY – sebbene alquanto scomoda per il principiante – era ormai uno standard e venne impiegata anche in questo settore.
Sul costo totale dei primi grossi computer, quello della tastiera era senza dubbio trascurabile rispetto a quello di tutti gli altri componenti. Oggi la situazione si è invece capovolta.

LA MATRICE NELLA TASTIERA

I tasti nella tastiera di un computer sono connessi elettricamente ad una griglia di fili. L'illustrazione mostra come, alla pressione di un tasto, si connetta una (ed una sola) coppia di fili, permettendo al computer di interpretare quale sia il tasto premuto



DAVID MALLOTT

Fino a quando non venne introdotto lo ZX81 della Sinclair, il costo di una tastiera di tipo tradizionale poteva incidere notevolmente su quello di un microcomputer. Le tastiere con pulsanti meccanici (quali quelle del BBC o del Dragon) usano dei veri e propri interruttori collocati sotto ciascun tasto (vedere figura alla pagina seguente). Quando viene premuto un tasto, il corrispondente interruttore chiude un circuito. Ogni tasto di questo tipo contiene numerose parti meccaniche che fanno lievitare enormemente il costo di produzione. Il BBC, ad esempio, conta 74 tasti, ma in altri apparecchi ve ne sono anche molti di più.

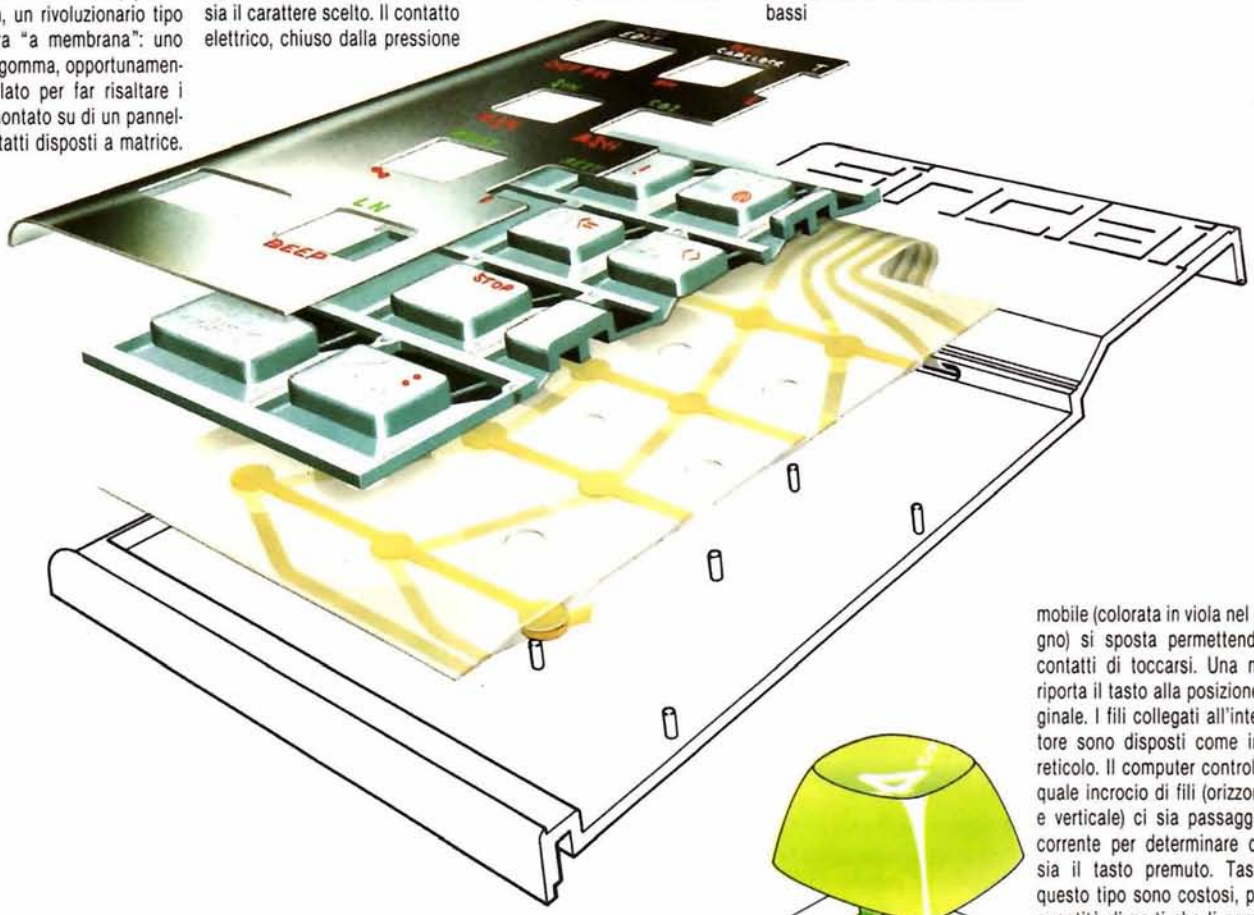
Alcuni costruttori, per aggirare questo problema, hanno impiegato un nuovo tipo di tastiera, detto "a sfioramento" (in inglese: touch-sensitive).
È il caso dello ZX81, un piccolo computer progettato principalmente per giochi e per piccole applicazioni di programmazione. Considerato che l'uso della tastiera, per questo tipo di impiego, è decisamente minimo, sembrò logico alla Sinclair sacrificare la qualità della tastiera a vantaggio di un prezzo più accessibile.
Nella tastiera dello ZX81, infatti, è stata eliminata gran parte delle minuterie meccaniche.

L'inconveniente di questo tipo di tastiere è che non si ha una conferma «tattile» di aver premuto un tasto ed è sempre necessario verificare sullo schermo se il carattere è stato registrato dal computer. Nel modello successivo della Sinclair, lo Spectrum, è stata impiegata la tastiera "a membrana" (vedere figura alla pagina seguente), che rappresenta un miglioramento, ma pur sempre un compromesso costo/qualità.

Gran parte dei computer dal costo relativamente basso (tra i quali il BBC, il Dragon e il Lynx) possiede tastiere "professionali", assolutamente indispensabili se deve essere svolta una grossa mole di lavoro alla tastiera (ad esempio se si lavora all'elaborazione di testi). L'uso di una tastiera del tutto simile a quella di una macchina da scrivere facilita ed accelera il lavoro dell'operatore.
Esiste una categoria di tastiere intermedia alle due appena citate, del tipo adottato sulle calcolatrici: i tasti sono piccoli, non sempre disposti secondo lo standard QWERTY, ma col vantaggio di avere un "tocco" più apprezzabile. Il New Brain e l'Oric-1 impiegano questo tipo di tastiera.

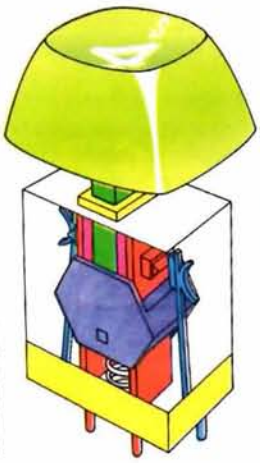
LA TASTIERA SINCLAIR

La Sinclair ha adottato, per lo Spectrum, un rivoluzionario tipo di tastiera "a membrana": uno strato di gomma, opportunamente modellato per far risaltare i tasti, è montato su di un pannello di contatti disposti a matrice. Quando si preme un tasto, il corrispondente contatto elettrico si chiude ed il computer, esaminando quale coppia di fili è in contatto, riesce a capire quale sia il carattere scelto. Il contatto elettrico, chiuso dalla pressione sul tasto, è normalmente tenuto aperto grazie ad una minuscola bolla d'aria presente tra i due fogli di plastica. Il ritorno del tasto alla posizione normale è assicurato dall'elasticità della somma. Questo sistema, pur causando qualche inconveniente nell'uso, permette però di produrre tastiere a costi veramente bassi



I TASTI MECCANICI

In genere questo tipo di tasti — simili a quelli di una macchina da scrivere — sono composti da un paio di contatti meccanici normalmente aperti. Quando si preme il tasto, una piccola parte mobile (colorata in viola nel disegno) si sposta permettendo ai contatti di toccarsi. Una molla riporta il tasto alla posizione originale. I fili collegati all'interruttore sono disposti come in un reticolo. Il computer controlla in quale incrocio di fili (orizzontale e verticale) ci sia passaggio di corrente per determinare quale sia il tasto premuto. Tasti di questo tipo sono costosi, per la quantità di parti che li compongono, ma offrono maggiore affidabilità e hanno un "tocco" più percepibile, grazie all'impiego della molla di ritorno. Un tasto di buona qualità oltre a permettere di sentire "al tatto" se il computer abbia o meno registrato il carattere, è modellato in modo da facilitare un prolungato uso della tastiera



DAVID WEEKS

In alcuni casi, per ovviare alla mancanza di conferma "tattile" nelle tastiere "a sfioramento", viene emesso un "bip" sonoro ogni volta che si preme un tasto.
I progettisti dello ZX81 e dello Spectrum hanno introdotto un metodo nuovo e molto utile per ridurre apprezzabilmente l'immissione di istruzioni BASIC: ogni tasto può servire non solo ad immettere un carattere, ma anche — grazie alla contemporanea pressione di un tasto di "funzione" — complete istruzioni BASIC di più lettere con una sola operazione.
Per esempio, l'istruzione PRINT viene immessa premendo il tasto funzione ed il tasto corrispondente alla lettera P. Un sistema simile è adottato dal Sord M5.



BASTA CON LA DATTILOGRAFIA

Fino a non molto tempo fa, si potevano immettere dati unicamente mediante l'uso di una tastiera. Molti potenziali utenti, scoraggiati dal fatto di non essere bravi dattilografi, esitavano nell'acquisto di piccoli computer: così alcuni fabbricanti hanno inventato il Mouse (Topo).
Questo piccolo apparecchio, fatto scorrere su di una superficie piana, consente facili e rapidi spostamenti del cursore sullo schermo del computer, semplificando l'operazione di immissione. Il Topo può essere adoperato, nel modo grafico, per disegnare linee o selezionare colori sullo schermo

Gira e rigira

Come ripetere gruppi di istruzioni un numero prefissato di volte

La precedente lezione è terminata con la formulazione del programma riportato qui sotto: funzionava bene, ma a causa dell'istruzione GOTO nella linea 70, il programma ripeteva il ciclo di istruzioni all'infinito. Per interromperlo, occorreva premere il tasto RESET (o BREAK) oppure spegnere il computer.
Adesso vediamo come uscire da questo ciclo infinito inserendo nel programma una verifica di condizione. In genere, il sistema usato è di controllare se un certo numero, che non useremmo altrimenti, viene digitato alla tastiera. Il programma originale richiedeva l'immissione di un numero, che sommato ad 1, veniva visualizzato sullo schermo. Se decidessimo di limitare il valore di questo numero ad un massimo di 999, digitando un numero superiore a quello limite potremmo far terminare il programma. Ecco come fare: inseriamo il programma originale ed aggiungiamo la linea

35 IF A > 999 THEN GOTO 80 <CR>

Adesso, facendo "girare" il programma, il funzionamento sarà identico a quello precedente, a meno che noi non immettiamo un numero maggiore di 999.
Proviamo con 1000 <CR>.
Perché il programma si è fermato? È stata l'istruzione IF nella linea 35. Quando il BASIC incontra una frase IF sa che deve effettuare una verifica sui valori che seguono. Il simbolo > significa "maggiore di". In altri termini, la linea 35 significa: «se (IF) la variabile A è maggiore di 999 allora (THEN) vai (GOTO) alla linea 80». Poiché noi abbiamo immesso il valore 1000 (maggiore di 999) l'esecuzione del programma passa alla linea 80 dove trova una END. Se i valori immessi non sono maggiori di 999, le istruzioni che seguono la parola THEN vengono ignorate, e l'esecuzione passa tranquillamente alla linea successiva.
Riassumendo, il gruppo di istruzioni IF-THEN intercetta una certa condizione da noi stabilita per seguire un percorso alternativo che, nel nostro caso, è il modo per terminare il programma. Quando il BASIC ha terminato l'esecuzione di un programma, ci avverte visualizzando un simbolo (**prompt**). Il prompt del BBC è il simbolo >, mentre quello del Dragon è OK e quello del Sord è READY. Qualunque sia il simbolo, con esso il BASIC ci avverte che è ora pronto ad accettare nuovi comandi.
Il trattamento dell'istruzione THEN non è identico in tutti i computer, come indicato nella rubrica "Basic & Basic".
Altri simboli per confronti sono: < (minore di), = (uguale), > = (maggiore o uguale a), < = (minore o uguale a) e < > (diverso da). Incontreremo questi simboli molto frequentemente nel progredire del corso.

```
10 REM I COMPUTER NON SBAGLIANO MAI
20 PRINT "DIGITA UN NUMERO"
30 INPUT A
40 LET A = A + 1
50 PRINT "PENSO CHE IL NUMERO FOSSE";
60 PRINT A
70 GOTO 20
80 END
```

Esercizi

* Si modifichi una delle linee, in modo che il programma termini se il valore immesso è 1000.

* Si modifichi una delle linee, in modo che il programma termini se il valore è minore di zero.

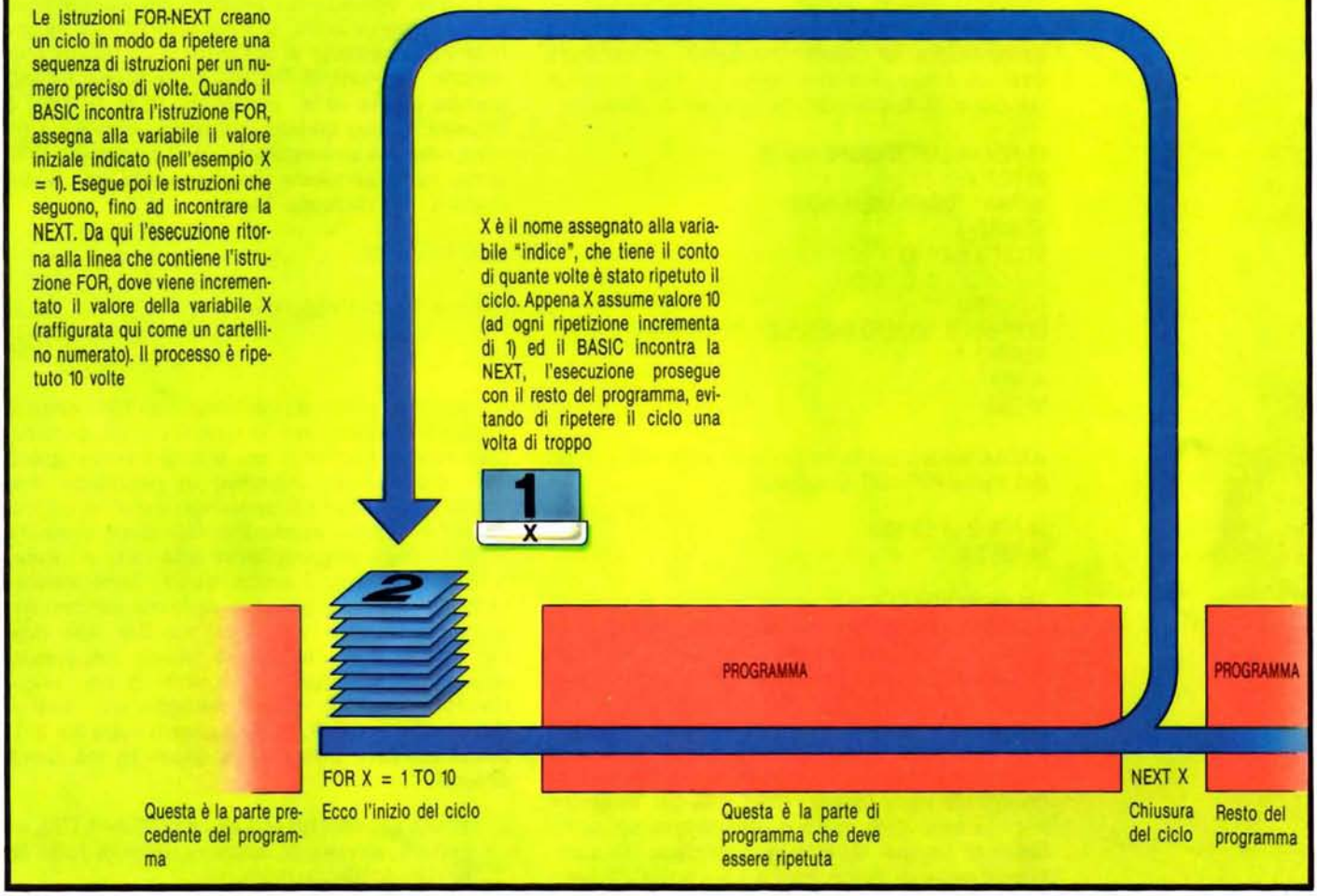
* Si modifichi la linea con il GOTO, in modo da far ripartire il programma dall'inizio se il valore di A è uguale o minore di 500. (Nota: è necessaria una nuova linea con IF-THEN ed un'altra con un GOTO).

Alla scoperta di FOR-NEXT

Nello scrivere programmi, può accadere che si voglia ripetere una serie di istruzioni per un numero prefissato di volte. Con il GOTO nella linea 70, questo numero era infinito, anche se con le istruzioni IF-THEN siamo riusciti a creare una "scappatoia".
Rivediamo il programma che moltiplica per

IL CICLO FOR-NEXT DEL BASIC

Le istruzioni FOR-NEXT creano un ciclo in modo da ripetere una sequenza di istruzioni per un numero preciso di volte. Quando il BASIC incontra l'istruzione FOR, assegna alla variabile il valore iniziale indicato (nell'esempio X = 1). Esegue poi le istruzioni che seguono, fino ad incontrare la NEXT. Da qui l'esecuzione ritorna alla linea che contiene l'istruzione FOR, dove viene incrementato il valore della variabile X (raffigurata qui come un cartellino numerato). Il processo è ripetuto 10 volte



10 il valore immesso, e facciamo in modo che venga ripetuto 8 volte:

```
10 REM MOLTIPLICA PER DIECI
20 FOR X=1 TO 8
30 PRINT "DIGITA UN NUMERO"
40 INPUT A
50 LET A=A * 10
60 PRINT "IL NUMERO MOLTIPLICATO 10 È";
70 PRINT A
80 NEXT X
90 END
```

Scritto il programma, controlliamo con un LIST che non vi siano errori e poi lo facciamo "girare" con un RUN. La richiesta di immettere un numero viene ripetuta otto volte. La ragione sta nella linea 20:

```
20 FOR X=1 TO 8
```

Questa istruzione è parte di un ciclo FOR-NEXT, che costituisce una delle strutture più utili del BASIC. Vediamone l'uso.
Nel programma appena visto, abbiamo creato una variabile chiamata X. Il nome X è puramente arbitrario e potevamo usare un qualsiasi altro nome (al di fuori di A, che è già usato nel programma). L'istruzione FOR deve essere sempre usata in abbinamento con l'istruzione NEXT, che incontriamo più avanti nel programma, subito dopo la parte che deve essere ripetuta.
Il formato dell'istruzione FOR è sempre il seguente:

```
FOR variabile = valore iniziale TO valore finale
```

Nel nostro esempio, FOR X=1 TO 8, abbiamo assegnato ad X un valore iniziale di 1. L'esecuzione del programma continua con la parte in cui viene immesso un numero, e visualizzato il risultato della moltiplicazione per 10. A questo punto incontriamo l'istruzione NEXT X, che causa un salto dell'esecuzione alla linea 20. Il valore della X viene incrementato di 1, così X vale adesso 2. La porzione di programma compresa tra la FOR e la NEXT viene di nuovo eseguita. Alla linea 80 il BASIC torna nuovamente alla 20, dove incrementa la X, che vale ora 3. Il processo viene ripetuto finché arrivata alla linea 80 con X che vale 8, l'esecuzione passa alla linea 90. In altre parole, il ciclo FOR-NEXT termina quando la variabile assume il valore finale (nel nostro caso 8).

Altri impieghi del ciclo FOR-NEXT

Talvolta, i cicli FOR-NEXT vengono impiegati per introdurre dei ritardi nell'esecuzione, altrimenti troppo veloce, di alcune istruzioni. Proviamo ad introdurre un ritardo nel nostro programma di moltiplicazione, simulando così un computer che "pensa " alla risposta da dare. Il listato del programma diventa:

```
10 REM MOLTIPLICAZIONE PER 10
20 FOR X=1 TO 8
30 PRINT "DIGITA UN NUMERO"
40 INPUT A
50 LET A=A * 10
52 FOR D = 1 TO 1000
54 NEXT D
60 PRINT "IL NUMERO MOLTIPLICATO PER 10 È";
70 PRINT A
80 NEXT X
90 END
```

Abbiamo aggiunto le linee 52 e 54 all'interno del ciclo FOR-NEXT originale:

```
52 FOR D=1 TO 1000
54 NEXT D
```

La variabile D è inizialmente posta uguale ad 1, l'esecuzione passa alla linea 54, dove si trova la corrispondente istruzione NEXT. In effetti, all'interno di questo ciclo non accade proprio niente, se non che il valore di D viene incrementato di 1. Le linee 52 e 54 vengono ripetute 1000 volte, prima di passare alla linea successiva, che visualizza il risultato. I computer sono molto veloci, ma per eseguire il ciclo ben 1000 volte è necessario un certo lasso di tempo. Questo può variare tra marche diverse di computer: ad esempio, l'Epson HX-20 impiega 2,9 secondi, mentre lo Spectrum impiega 4,5 secondi. Potremmo provare a cambiare il valore finale nella linea 52, ottenendo ritardi più o meno lunghi. Per meglio simulare un comportamento umano, potremmo aggiungere le linee:

```
56 PRINT "LASCIA CHE CI PENSI..."
57 FOR E=1 TO 1000
58 NEXT E
```

LISTiamo il programma ed eseguiamolo con un RUN. Abbiamo adesso due cicli di ritardo, che servono solo a... far trascorrere del tempo. Aggiungiamo ora:

```
51 REM QUESTO CICLO PERDE TEMPO
55 REM E QUESTO NE PERDE ANCORA DI PIÙ
```

Usiamo di nuovo la LIST ed osserviamo bene il programma. Si noti che le linee aggiunte si sono automaticamente collocate al punto giusto. Ciò ci induce a parlare dell'ultimo argomento di questa lezione: la numerazione delle linee di programma.
La numerazione da noi usata del programma originale partiva da 10. Ad ogni linea successiva abbiamo sommato 10 alla numerazione, che terminava con 90. Potevamo scegliere un'altra numerazione, ad esempio: 1, 2, 3... 9, ma allora dove avremmo inserito le linee che abbiamo aggiunto?
I programmatori hanno sempre dei ripensamenti o delle piccole modifiche da apportare: è quindi conveniente lasciarsi un poco di spazio, tra linea e linea. C'è chi addirittura segue una numerazione di 50 in 50 o di 100 in 100. Alcune versioni di BASIC possiedono un comando molto utile, chiamato AUTO. Il BBC e l'Epson hanno questo comando, mentre altri computer ne sono sprovvisti. Il comando AUTO serve per impostare una numerazione automatica. Proviamone l'uso:

```
AUTO 100, 10<CR>
```

Sullo schermo appare:

```
100
```

Il "cursore" rimane posizionato subito dopo il numero visualizzato. Il cursore è un particolare segno (talvolta un piccolo rettangolo), che indica sullo schermo la posizione ove verrà visualizzato il prossimo carattere. Possiamo a questo punto inserire normalmente le istruzioni appartenenti alla prima linea, terminando con il solito <CR>. Automaticamente, appare il numero di linea successivo (nel nostro caso 110) e così via fino alla fine del programma. AUTO può essere impiegato con o senza "argomenti". Il termine "argomento" proviene dalla matematica: nell'espressione 2 + 3 = 5, gli argomenti sono 2 e 3. Il comando AUTO può essere usato in tre modi diversi:

1. senza argomenti (ad es.: AUTO<CR>). Ciò, in genere, avvia una numerazione di 10 in 10 partendo dalla linea 10.
2. con un argomento (ad es.: AUTO 100 <CR>). Ciò causa una numerazione di 10 in 10, ma partendo dal valore indicato 100.
3. con due argomenti (ad es.: AUTO 300,50). In questo caso la numerazione parte da 300 ed è incrementata di 50 in 50.

Il numero di linea più alto si aggira normalmente intorno a 65000, il che ci lascia ampi margini anche per i programmi più lunghi. Nella prossima lezione esamineremo i vari metodi di presentazione dei dati sullo schermo e sulla stampante.

Basic & Basic

IF

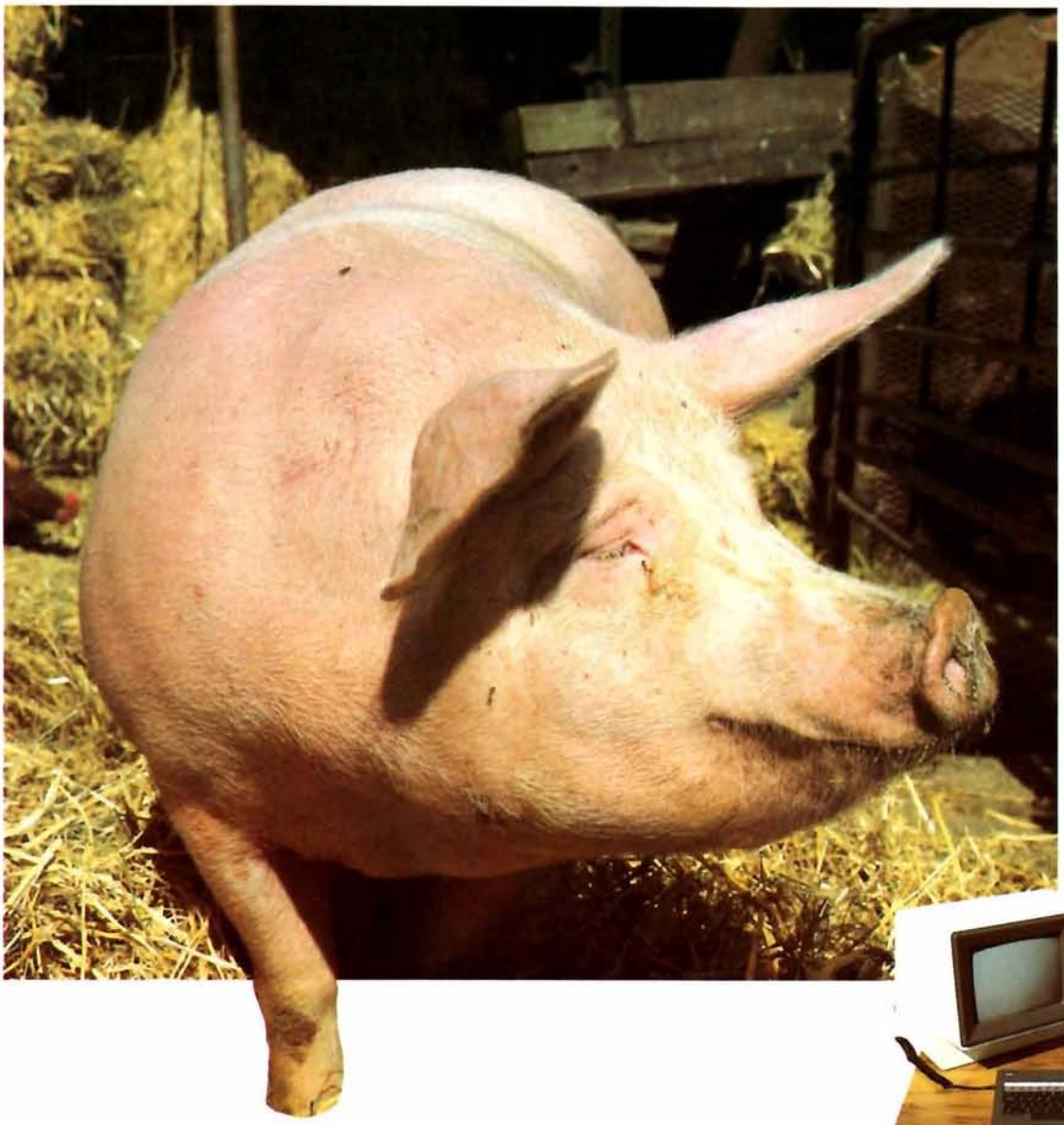
Nella gran parte dei microcomputer l'istruzione ha la forma: IF A>999 THEN 80 oppure IF A>999 GOTO 80 (Nello Spectrum e nello ZX81: IF A>999 THEN GOTO 80)

AUTO

Questo comando non è disponibile né sul Dragon 32, né sul Commodore VIC 20, né sul Commodore 64, né sullo Spectrum, né sullo ZX81

La controrivoluzione

Applicazioni del computer nelle piccole aziende



Il computer "agricolo"

Non deve sorprendere il fatto che il computer sia ugualmente utile in campagna quanto in ufficio. Un programmatore piuttosto intraprendente ha preparato un programma, chiamato "Optimiser", che riduce al minimo il costo per l'alimentazione dei suini. Il programma ha avuto un enorme successo in tutto il mondo. L'allevatore fornisce al computer i dati relativi al fabbisogno giornaliero dei suini ed i valori nutritivi dei cereali disponibili, in termini di proteine e vitamine. Il programma determina quale sia la miscela più economica da usare. Quando si considera che certi allevatori consumano oltre 400 tonnellate di mangime al mese, si può comprendere come un opportuno dosaggio possa far risparmiare enormemente.
Rimangono, tuttavia, alcuni problemi soggettivi: i gusti dell'animale, per esempio. Il suino si abitua ad una particolare dieta ed un cambiamento drastico non è mai ben accetto



Il computer è nato nei laboratori militari ed universitari. I primi esemplari vennero impiegati per fare previsioni meteorologiche o per calcolare la traiettoria, in condizioni atmosferiche avverse, di siluri sparati da navi da guerra. Non ci volle molto tempo per accorgersi che il computer aveva un enorme potenziale commerciale. Inizialmente soltanto grandi aziende erano in grado di affrontare la spesa per un computer. Oggi il prezzo di questi apparecchi è talmente accessibile, che anche le più piccole imprese ne possono acquistare uno.

Com'è possibile che un derivato della scienza militare possa essere utile, mettiamo, ad un negoziante?
La risposta è semplice: un computer è capace di conservare ed elaborare una grande quantità di dati, e riorganizzarli ordinatamente secondo le richieste dell'operatore. Gli articoli in vendita in un negozio possono essere i più disparati (giornali, penne, caramelle, ecc.). Se ognuno di essi viene "catalogato" nella memoria del computer, assieme al quantitativo giacente, al prezzo ed alla scorta minima, il negoziante, che ha regolarmen-

te registrato le vendite, oltre ad avere una situazione di magazzino sempre aggiornata, viene anche avvertito automaticamente dal computer quando la scorta di un articolo è sotto il livello "normale". Inserendo gli appositi moduli nella stampante, viene automaticamente stilato l'ordine al fornitore. Il computer può svolgere anche altri lavori, che manualmente richiedono molto tempo: calcolo di stipendi, dell'I.V.A e tenuta della contabilità in genere.

Il computer non è soltanto un apparecchio utile: in certi casi è addirittura indispensabile. L'uomo non sarebbe mai andato sulla Luna senza l'aiuto di un computer. Esistono, infatti, dei problemi nel progetto dei razzi e nel calcolo delle traiettorie risolvibili unicamente grazie al computer. Grazie ad esso, l'Umanità si appresta ad esplorare orizzonti finora sconosciuti.

Ma torniamo all'esempio del negoziante. Un'altra fonte di risparmio (e quindi di guadagno) può essere individuata nell'ottimizzazione delle attività. Poniamo che il negoziante effettui consegne a domicilio: nel corso della giornata vengono accumulati nella memoria del computer gli indirizzi dei clienti e gli articoli da consegnare. Al momento opportuno si può chiedere al computer di fornire una stampa di questi dati, organizzati, per esempio, in zone di consegna oppure in base al migliore percorso di consegna e via dicendo. A fine mese può essere stampato un riepilogo, per ciascun cliente, con i totali della merce consegnata e delle somme dovute. Il compito è ancora più facile se le ordinazioni (e relative consegne) sono periodiche. A sua volta il negoziante ha una precisa idea della quantità di merce da ordinare ai suoi fornitori.

IL SISTEMA PER LE PICCOLE IMPRESE

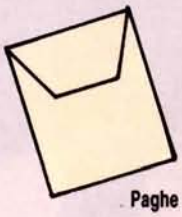


Un sistema computerizzato per una piccola impresa deve soddisfare alcune esigenze quali ad esempio: essere affidabile anche dopo ore e ore di lavoro; avere una tastiera "comoda"; permettere l'uso di unità a dischi, necessari per programmi di buon livello; essere collegabile ad una buona stampante



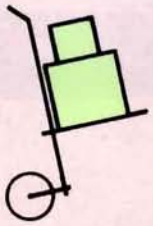
Contabilità

Tutta la contabilità dell'azienda può essere gestita usando particolari programmi. La situazione contabile risulta sempre aggiornata ed è possibile stampare bilanci di verifica in qualsiasi momento. I migliori programmi sono progettati a "moduli", che automaticamente, dai dati di prima nota o dalla emissione di una fattura, ricavano i dati sufficienti ad aggiornare l'intera contabilità



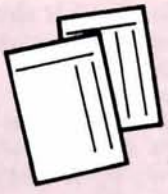
Paghe

Un programma per le paghe calcola, mensilmente, gli stipendi del personale. Le detrazioni individuali e le ritenute di Legge vengono accantonate in modo automatico, ed il tutto viene poi stampato dal computer nella "busta paga" di ciascun dipendente. Nei casi previsti, è possibile anche inserire dati riguardanti gli straordinari o eventuali somme "extra" da conteggiare adeguatamente



Magazzino

Un programma di gestione del magazzino offre la possibilità di avere sempre sotto controllo la situazione delle giacenze, delle voci "sotto scorta" e consente, inoltre, di compilare l'inventario con poche semplici istruzioni. La valorizzazione può essere fatta "a ultimo costo", "a costo medio" o LIFO, come richiesto dalla Legge in Italia. Alcuni programmi offrono anche la possibilità di inserire i dati mediante penne ottiche e l'uso del Codice a Barre, accelerando notevolmente i tempi di inserimento



Fatturazione

Un buon programma di fatturazione (immediata e/o differita) consente di emettere fatture precise, aggiornando automaticamente la situazione contabile dell'azienda. Quando vengono forniti i codici del cliente o quelli degli articoli di magazzino (se anche questo è collegato alle procedure), questi vengono automaticamente controllati e visualizzati sullo schermo, per evitare attribuzioni errate



TONY LODGE



Sotto il cofano

In molte autofficine si usano computer per le procedure di revisione delle automobili. Sullo schermo vengono presentate delle "schede" in cui l'operatore (che talvolta è il meccanico stesso) inserisce informazioni sullo stato delle varie parti del motore e della carrozzeria. Viene anche presentato un elenco dei controlli da eseguire e si evita che il meccanico "si dimentichi" qualche verifica.
I dati così raccolti possono poi essere utilizzati per la fatturazione dell'intervento.

A ciascuno il suo panino!

Un fornaio di Colnbrook, Gran Bretagna, usa un piccolo computer per gestire la sua produzione giornaliera.
Le ordinazioni vengono "passate" al computer, e gli sprechi sono ridotti al minimo.
L'apparecchio viene usato anche per tenere la contabilità e per la fatturazione.

Un taglio... alla byte!

C'è un noto parrucchiere di Londra che ha introdotto l'uso del computer nella sua attività. Egli tiene un archivio sempre aggiornato sulla... capigliatura delle sue clienti.
Colore, permanente, taglio ed altri dati essenziali vengono di volta in volta archiviati e, all'occorrenza richiamati: in tal modo la prestazione del parrucchiere risulta migliorata.
Sembra che l'innovazione abbia avuto enorme successo presso la clientela.

Un altro gruppo di programmi, fra i tanti disponibili in commercio, è quello dei "word processor", per elaborare testi. Con essi è possibile scrivere lettere, circolari, articoli o anche interi libri, memorizzando il tutto su dischetti magnetici. La scrittura e la correzione del testo viene fatta sullo schermo del computer e la stampa può avvenire quando si è terminata la stesura definitiva. Ciò comporta notevoli risparmi di carta e di tempo, offrendo inoltre la possibilità di stampare più copie di uno stesso testo (anche in momenti diversi) semplicemente richiamandolo nella memoria centrale. Un lavoro così ripetitivo è indubbiamente stancante se fatto nel modo tradizionale, mentre la macchina non si stanca mai!
Col progredire della tecnica in questo campo, saremo ben presto in grado di collegare il nostro computer ad altri situati anche a grandissima distanza, procurando in pochi minuti, lo scambio di milioni di informazioni. Il computer diverrà indispensabile quanto lo è oggi il telefono.

A cena col computer

Anche in un ristorante le applicazioni di un computer sono molteplici. Oltre alla solita contabilità ed all'emissione di ricevute fiscali, esso può essere impiegato per ottimizzare le scorte di cibo, per stampare i menù, per gestire le ordinazioni e anche... per ricordare i clienti più affezionati!
Uno dei vantaggi più apprezzabili è senz'altro l'ottimizzazione degli acquisti in base a statistiche generate dalle richieste dei clienti e dalle dosi delle ricette.

Dipingere coi numeri

Come si creano le immagini con la grafica del computer

Un dipinto su tela viene eseguito adoperando "ad arte" il pennello ed i colori. Ma come si comporta un artista che voglia creare un'immagine sullo schermo di un computer?
La grafica nei microprocessori può essere utilizzata ricorrendo ad uno dei tre sistemi principali, che si distinguono per il livello di controllo sulla definizione dell'immagine finale: grafica "a blocchi", grafica "a pixel" e grafica ad alta risoluzione.
Nella grafica ad alta risoluzione, l'artista può operare su ogni singolo "punto" sullo schermo del monitor. L'unica limitazione consiste nella quantità di memoria disponibile, dove viene conservata una replica elettronica dello schermo. Con 32 Kbyte, per esempio, ogni punto dello schermo di un monitor corrisponde ad un bit di memoria.
Nella grafica "a blocchi", l'artista deve in parte sacrificare la possibilità di operare su ogni singolo punto dello schermo, ma ha il vantaggio di poter usare un vasto numero di forme "predefinite" che facilitano la costruzione di immagini complesse. In genere, queste forme predefinite sono riportate su ciascun tasto della tastiera e sono pertanto richiamabili direttamente. Selezionando il funzionamento grafico, la tastiera diviene una "spatola" per dipingere.
In questo tipo di grafica, ogni forma è costruita all'interno di una piccola matrice di punti (generalmente 8×8 punti). In alcuni tipi di computer è addirittura possibile, con un apposito programma, definire a piacimento la forma di queste piccole immagini.
La grafica "a pixel" è una via di mezzo tra le due già esaminate: si può operare su porzioni di schermo più piccole di una matrice di 8×8

I TRE TIPI DI RISOLUZIONE



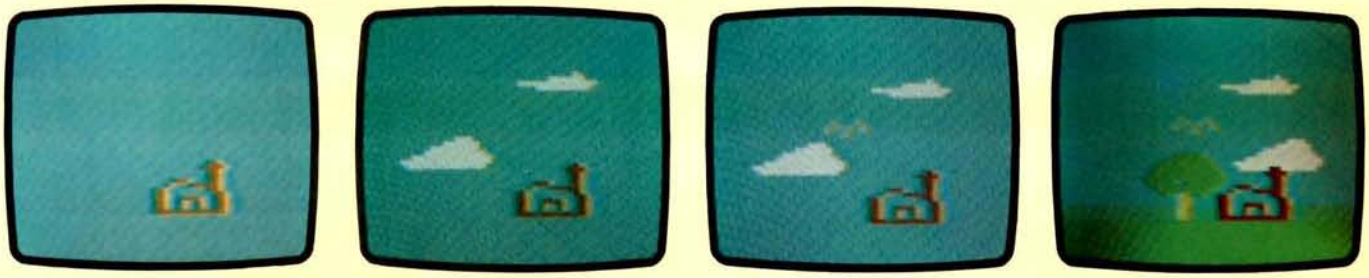
Nella grafica a bassa risoluzione (o "a blocchi"), è sufficiente 1 Kbyte di memoria per conservare tutte le caratteristiche di un'immagine creata allo schermo. Se lo schermo è suddiviso in 24 righe di 40 "blocchi" ciascuna, il numero totale di blocchi è 960, poco meno di un Kbyte (= 1024 byte). Poiché un byte è composto di otto bit, un Kbyte offre la possibilità di conservare fino a 8 diverse informazioni (con un totale di 256) su ciascun blocco da visualizzare

Nella grafica a media risoluzione, l'immagine sullo schermo è caratterizzata da un maggior numero di dettagli, richiedendo perciò una maggiore quantità di memoria. Con 8 Kbyte di memoria è possibile memorizzare la condizione acceso/spento di 65.536 punti sullo schermo. Se però lo schermo è a colori, un solo bit (acceso/spento) non basta: ce ne vorranno almeno tre per indicare anche il colore del punto, aumentando ancor più l'utilizzazione della memoria

La grafica ad alta risoluzione necessita di una gran quantità di memoria. Spesso lo schemo è suddiviso in 240 righe ed ogni riga contiene fino a 640 punti. Il numero dei punti "indirizzabili" sale così a 268.800. Se lo schermo è monocolore, sono dunque necessari 33.600 byte (268.800 : 8) per indicare se un dato punto è acceso o spento. Ovviamente, se volessimo immagini a colori la quantità di memoria da dedicare alla gestione grafica sarebbe, come minimo, il triplo

JO LAWRENCE/IAN McKINNELL

LA GRAFICA "A SPRITE"



Il nome "sprite" indica un particolare metodo di rappresentare, con un soddisfacente realismo, le immagini sullo schermo di un computer.
Questo tipo di grafica, inizialmente sviluppato dalla Texas Instruments, è oggi disponibile anche sulla maggior parte dei piccoli computer (TI99/4A, Commodore 64, Atari, Sord M5, ecc.).
Nella grafica di tipo convenzionale, le immagini sono create su di un solo "piano", così come verrebbero disegnate su di un foglio di carta.
Nella grafica "a sprite" il disegnatore può collocare le immagini anche su "strati" diversi. Il numero di strati disponibili varia da computer a computer (il Sord M5 ne offre addirittura 32).

Possiamo immaginare questi strati come fogli di plastica trasparente. Se sul foglio più vicino all'osservatore è disegnato un albero e su quello più lontano è disegnata una nuvola, possiamo far "muovere" il secondo foglio ottenendo l'illusione che la nuvola si sposti nel cielo, rimanendo "nascosta" nel passare dietro all'albero in primo piano.
L'effetto che si ottiene è dunque di un'immagine dotata di prospettiva. Gli oggetti possono essere rappresentati in forma tridimensionale e più figure possono muoversi indipendentemente l'una dall'altra anche su piani diversi. Una volta definiti i vari "soggetti" di uno scenario, è possibile farli muovere con semplici comandi, specificando direzione e velocità.

punti, ed ogni pixel può essere richiamato dalla memoria e collocato sullo schermo nella posizione desiderata.
Il BASIC usa due comandi per disegnare linee nella grafica ad alta risoluzione: MOVE (per spostare il "pennello" su di un preciso punto dello schermo) e DRAW (per disegnare una linea tra due punti). Questi comandi sono eseguiti da due numeri, che individuano la riga e la colonna dello schermo. Generalmente la numerazione delle righe parte dal settore superiore dello schermo mentre quella delle colonne va da sinistra a destra, cosicché il punto situato nella riga zero, colonna zero è, sullo schermo, quello più in alto a sinistra.
A titolo di esempio, viene riportato qui sotto un breve programma, adatto per la gestione grafica del BBC e del Lynx (vedere «Basic & Basic» qui a lato). In altri computer le istruzioni MOVE e DRAW vanno sostituite con le equivalenti funzioni descritte nel manuale operativo.

```
10 MOVE 100,50
20 DRAW 92,95
30 DRAW 57,125
40 DRAW 100,110
50 DRAW 143,125
60 DRAW 108,95
70 DRAW 100,50
```

Queste poche istruzioni, una volta eseguite, fanno comparire sullo schermo un disegno "a Trinacria", un profilo che assomiglia alla Sicilia (Trinacria è l'antico nome dell'isola). Modificando opportunamente questo semplice programma è possibile disegnare sullo schermo un'immagine qualsiasi, usando le sole istruzioni MOVE e DRAW. Le uniche limitazioni sono la risoluzione grafica del computer (che può influire in particolare sulle linee curve) e la pazienza del programmatore (ogni linea curva, infatti, va definita punto per punto)! Il seguente programma, invece, costruisce l'immagine di un cono, disegnando una serie di cerchi. L'istruzione CIRCLE si può usare sullo Spectrum, sul Dragon e sull'Oric (vedere «Basic & Basic»):

```
10 FOR K = 2 TO 40
20 CIRCLE 40 + K, 40 + K,K
30 NEXT K
```

Gli esempi riportati illustrano come sia possibile, mediante semplici comandi grafici, costruire delle immagini al computer usando coppie di numeri. Tuttavia, la creazione di immagini molto complesse richiede l'immissione di una enorme quantità di coppie riga/colonna. Per rendere questo lavoro il più semplice possibile sono stati progettati degli appositi apparecchi da collegare al computer, ad esempio il digitiser. Con questo dispositivo, l'operatore crea immagini usando una "penna ottica" direttamente sullo schermo. Il digitiser provvede a tradurre le informazioni raccolte dalla penna ottica in coordinate memorizzabili dal computer.

Basic & Basic

PROGRAMMA 1

Per il Micro BBC deve essere inserita la linea:

```
5 MODE 0
```

PROGRAMMA 2

Per l'Oric deve essere inserita la linea:

```
5 HIRES
```

e la linea 20 deve essere sostituita con:

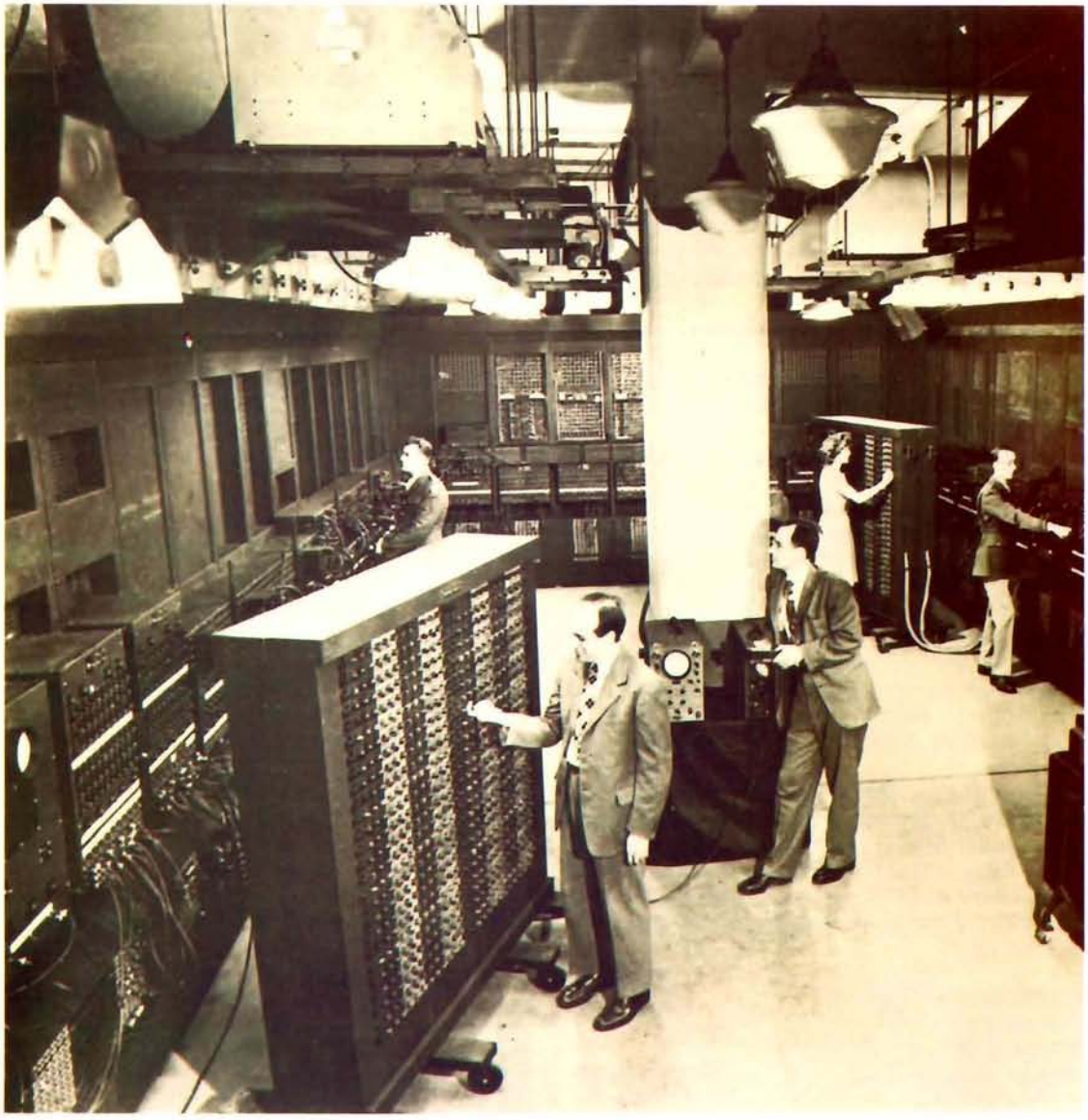
```
20 CURSET 40 + K,40 + K,0
25 CIRCLE K,1
```

Mentre per il Dragon è da inserire:

```
5 PMODE 4,1: SCREEN 1,1: COLOR 0,5: PCLES
20 CIRCLE (40 + K,40 + K), K
```

Lo scatto che conta

Gli enormi passi fatti nell'ingegneria miniaturizzata



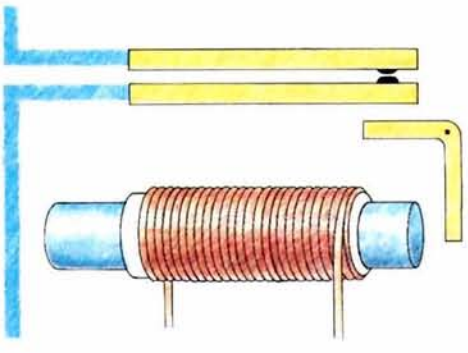
© THE SCIENCE MUSEUM

Il primo computer a valvole

Nel 1943, durante la Seconda Guerra Mondiale, un colonnello dell'esercito americano richiese il progetto di un apparecchio di calcolo per l'artiglieria. La sfida fu raccolta dall'università della Pennsylvania e l'apparecchio venne presentato nel 1946, dopo ben 7237 ore di lavoro umano.
Il macchinario venne chiamato ENIAC (Eletrical Numerical Integrator and Calculator) ed era il primo calcolatore a valvole. Esso impiegava 18.000 valvole, 1.500 relay e dissipava un calore equivalente a 200 Kilowatt. Occupava una stanza di m. 9 x 30. I limiti maggiori consistevano nell'affidabilità e nella quantità di memoria. L'ENIAC era in grado di memorizzare soltanto 20 numeri di 10 cifre e tutta la programmazione avveniva mediante allacciamenti di cavi elettrici. Quando l'ENIAC andò in pensione (nel 1952), il suo funzionamento aveva richiesto la sostituzione di ben 19.000 valvole. Infatti l'enorme calore generato faceva bruciare le valvole ad un ritmo di una ogni due minuti

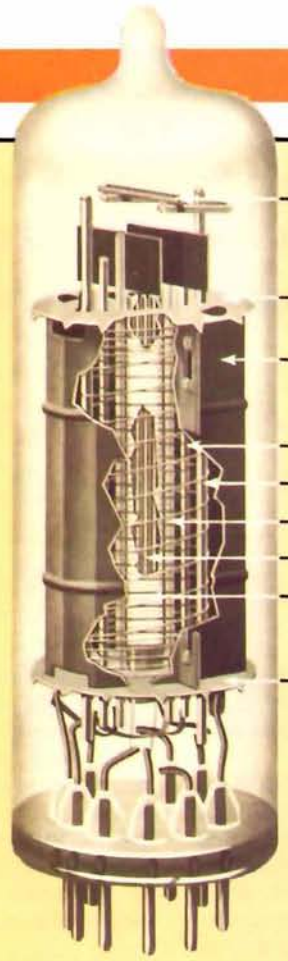
L'interruttore a relay

Quando la corrente passa attraverso la bobina avvolta sul nucleo di ferrite, si crea un campo magnetico, che attira, piegandolo, un contatto metallico. Il circuito ai due capi del relay si chiude e la corrente passa. Una molla disabilita il contatto quando il relay non è attivato



TONY LODGE

Un moderno computer contiene milioni di microscopici interruttori elettronici, fondamentali al suo funzionamento. Senza di essi, la rivoluzione tecnologica del secondo dopoguerra non sarebbe mai avvenuta.
Nel 1938 l'ingegnere Claude Shannon dimostrò che era possibile eseguire operazioni logiche mediante circuiti che impiegano interruttori elettrici. Poiché il funzionamento di un computer è fondato proprio su sequenze di operazioni logiche, occorreva inventare degli opportuni interruttori elettronici.
Un primo tentativo venne fatto con i "relay" meccanici. Sebbene impiegati nei primi tentativi, l'ingombro di simili circuiti limitava lo



Alcuni tipi di valvole, oltre ai tre componenti fondamentali (catodo, anodo e griglia), contengono altri elementi per migliorarne le prestazioni. Il principio di funzionamento, tuttavia, rimane identico

LA VALVOLA

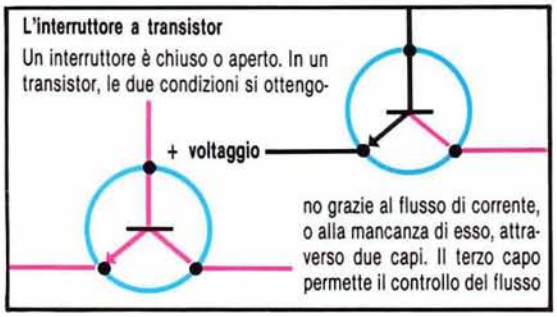
L'illustrazione mostra le varie parti di una valvola termoionica. Il tubo di vetro contiene un terminale positivo ed uno negativo (anodo e catodo), separati da una griglia metallica. Gli elettroni presenti sul catodo vengono attirati dall'anodo, causando un flusso di corrente. Questo flusso naturale può essere accresciuto in due maniere diverse. Il catodo può essere riscaldato fino all'incandescenza e la sua superficie trattata con speciali sostanze chimiche. La griglia di controllo, situata fra anodo e catodo, non interferisce con il flusso di elettroni a meno che non venga ad essa applicata una tensione negativa. In questo caso si può variare il flusso fino ad interromperlo. La valvola funziona così da interruttore applicando un'adeguata tensione alla griglia

per concessione della MULLARD LTD

sviluppo di elaboratori sufficientemente sofisticati. Inoltre, essendo in parte composti da delicati elementi meccanici, risultavano scarsamente affidabili e molto lenti.
La prima vera generazione di computer è caratterizzata dall'impiego di valvole termoioniche le quali, non avendo parti meccaniche in movimento, consentivano una maggiore affidabilità e velocità di calcolo. Tuttavia il consumo di energia elettrica era enorme (causando fra l'altro problemi non trascurabili di surriscaldamento dei componenti) e l'affidabilità non era ancora totale.
L'invenzione del transistor promosse una nuova generazione di computer. Il funzionamento del transistor, anche se teoricamente identico a quello delle valvole, offre numerosi vantaggi: basso costo di produzione, bassissimo consumo e migliori prestazioni. L'avvento del transistor segnò l'inizio delle sue applicazioni nel campo commerciale.
I computer odierni usano ancora transistor come interruttori, ma non più incapsulati singolarmente, bensì in forma "integrata". Un chip al silicio delle dimensioni di un'unghia può contenere fino ad un quarto di milione di transistor, troppo piccoli per essere visti a occhio nudo. Per quanto minuscoli, i transistor si comportano come interruttori. Potendo riunire così tanti interruttori in un singolo chip al silicio, il costo di produzione è calato in maniera colossale. Le dimensioni di un computer degli anni '50 erano tali da riempire un intero laboratorio: oggi la stessa potenza di calcolo risiede in un singolo chip, che rappresenta il cavallo di battaglia dei piccoli computer, così tanto diffusi.

L'interruttore a transistor

Un interruttore è chiuso o aperto. In un transistor, le due condizioni si ottengono grazie al flusso di corrente, o alla mancanza di esso, attraverso due capi. Il terzo capo permette il controllo del flusso



Gli inventori del transistor

Nel 1956 il Premio Nobel venne assegnato al gruppo di ricercatori il cui lavoro portò all'invenzione del transistor nel 1947. Nella foto, scattata nei laboratori della Bell Telephone (da sinistra a destra): il Dr. John Bardeen, il Dr. William Shockley ed il Dr. Walter Brittain



© ASSOCIATED PRESS LTD

Lo sapevate che...?

Notizie e curiosità che difficilmente riuscireste a trovare altrove

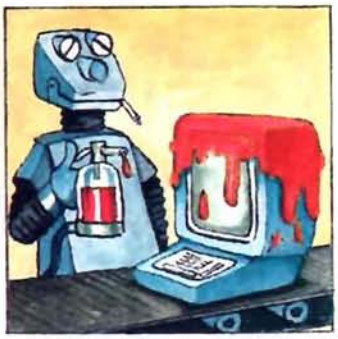
I computer provano emozioni?

Attualmente i computer non hanno, né possono avere, emozioni, perché ancora non sono intelligenti al punto da "pensare" per conto proprio. Non si sa quanto tempo occorrerà per produrre macchine realmente pensanti, ma molti esperti ritengono che ciò avverrà durante questa generazione. Se accettiamo la tesi secondo la quale il pensiero creativo è inseparabile dall'emotività, i computer di domani saranno in grado di provare "emozioni".



Qual è la differenza tra computer e robot?

I robot sono estensioni meccaniche dei computer: le braccia e gli occhi funzionano secondo le direttive del "cervello" del computer. I robot oggi impiegati per costruire auto o complessi stereo sono tutti regolati da un microprocessore, ma sono ancora abbastanza "stupidi". Posti di fronte ad una situazione imprevista, non sanno come comportarsi. In futuro, i robot possederanno circuiti più sofisticati e saranno dotati di una certa intelligenza.



Perché il software è così caro? Molti programmi per giocare al computer costano poche lire, mentre alcuni programmi applicativi costano milioni.

La stesura di programmi applicativi, in particolare di quelli per uso commerciale, richiedono il lavoro di gruppi di esperti per diversi mesi o addirittura anni. Le ditte che vendono il software devono quindi recuperare l'enorme investimento necessario allo sviluppo dei programmi. Un gioco per computer viene generalmente venduto in migliaia di esemplari, e l'incidenza degli investimenti iniziali è frazionata. Nel caso di programmi con mercato molto limitato, l'incidenza è tale da far lievitare il prezzo finale.

L'impatto del computer sulla situazione sociale è veramente drammatico?

Indubbiamente, gli effetti a lungo termine dell'avvento del microprocessore sulla società, ed in particolare nel campo del lavoro, sono difficili a pronosticarsi. Certo è che stiamo già assistendo a dei grossi cambiamenti. I computer, collegati a robot meccanici, sono in grado di sostituire l'uomo in catene di montaggio, ma anche altri tipi di mansioni fino a ieri esclusivamente riservate agli uomini vengono lentamente, ma inesorabilmente affidate a computer.
Ciò potrebbe generare, come qualche scrittore di fantascienza ha già scritto, problemi di occupazione e problemi di come impiegare il maggiore tempo libero a disposizione.



Si può usare un computer per rubare a una banca o per causare una Terza Guerra Mondiale?

Poiché i computer "dialogano" attraverso le linee telefoniche è possibile, in teoria, riuscire ad inserirsi nelle loro comunicazioni per compiere operazioni illecite. In pratica, però, vengono adoperati dei sistemi di sicurezza tali da impedire intrusioni da parte di malintenzionati. I messaggi vengono inviati in una forma codificata comprensibile soltanto a chi è in possesso della decodifica. Ogni sistema di codifica/decodifica è talmente sofisticato che richiederebbe qualche milione di anni di lavoro al computer per poter decifrarlo senza la chiave.

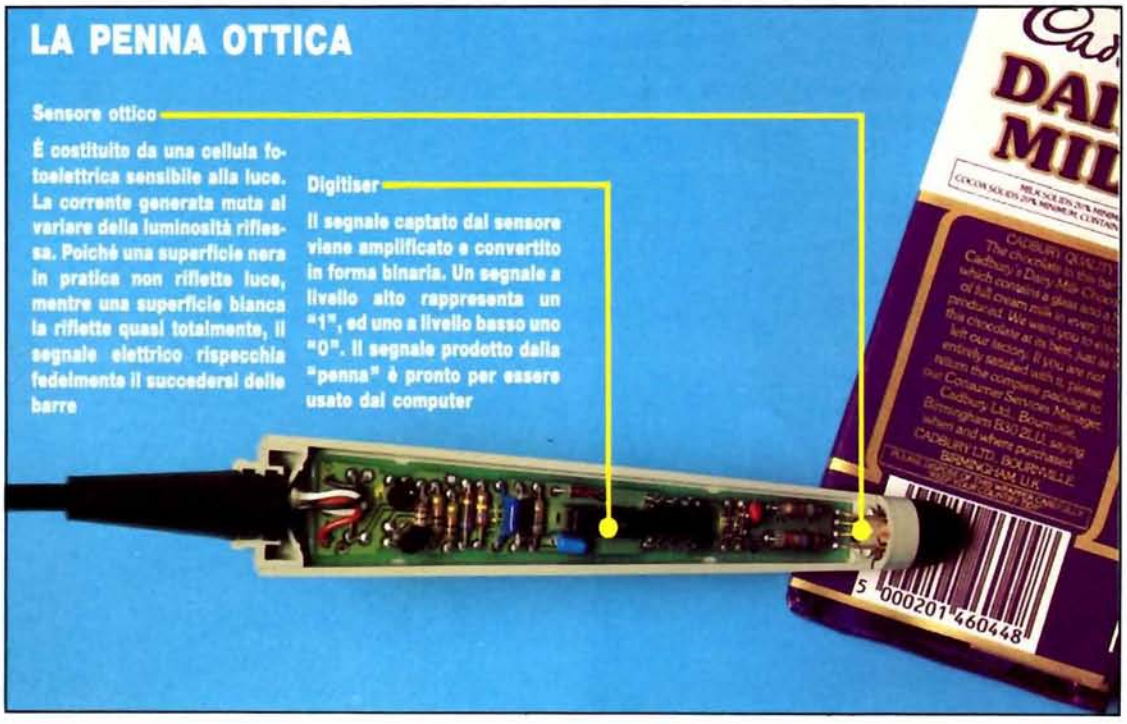


L'accesso a sistemi di comunicazioni militari è ancora più difficile, in quanto, in aggiunta ad altre precauzioni, non vengono adoperate le normali linee telefoniche, ma collegamenti via satellite e impiegando microonde. Questi sistemi non sono accessibili alla gente comune, e comunque le trasmissioni di carattere militare sono tutte codificate.

Dietro le... barre

Come il computer interpreta il codice a barre

Forse non tutti lo hanno notato, ma moltissimi prodotti in commercio hanno serie di misteriose striscette stampigliate sulla confezione.
Cosa hanno in comune una bibita in barattolo, un libro o una stecca di cioccolata? A che cosa serve, nei grandi supermercati, quell'apparecchio dotato di una strana penna che emette dei brevi "bip"? Quelle striscette sono il "codice a barre", un ingegnoso sistema che permette, grazie alle penne "magiche", una lettura istantanea, di tutti i dati relativi ad un particolare prodotto. Le informazioni vengono raccolte e successivamente passate al computer per essere elaborate. Questo metodo permette di avere sempre aggiornata la situazione di magazzino, di sapere quali articoli sono sotto scorta e quanto capitale è impegnato, momento per momento, nelle scorte.
Vediamo il caso di un libro. Ogni libro pubblicato in ogni parte del mondo, possiede un codice ISBN (International Standard Book Number). Il codice consta di uno o più caratteri per indicare la lingua in cui è scritto il libro (o la nazione in cui è stato pubblicato). Altri caratteri (da due a sette) identificano chi ha pubblicato il libro ed i restanti caratteri (da uno a sei) individuano il titolo e l'edizione del volume: in totale sono di nove cifre (più un carattere usato dal computer per controllare la validità dei precedenti).
Ai fini di questa codifica, i libri vengono numerati secondo il sistema EAN (European Article Numbering) composto di tredici cifre (per i prodotti alimentari viene spesso usato un codice più corto, di soli otto caratteri). Le prime tre cifre rappresentano il codice EAN (978 per i libri). Segue il codice ISBN, seguito a sua volta da un carattere di controllo EAN. Negli Stati Uniti ed in Gran Bretagna, il codice ISBN viene stampigliato anche in forma leggibile sia all'uomo sia a speciali lettori ottici di caratteri.
I lettori ottici di caratteri sono un altro interessante sviluppo della tecnica, che avrà grande influenza nel futuro. Esistono già oggi apparecchi in grado di leggere otticamente righe di testo. Il segnale generato da questi lettori ottici viene applicato ad un computer, che elabora l'informazione per gli scopi più disparati. Le parole lette possono essere, per esempio, visualizzate su di uno schermo, avendo evitato una delle operazioni più laboriose e noiose: l'immissione manuale del testo, per mezzo di una tastiera.



IAN McKINNELL

LA DECODIFICA



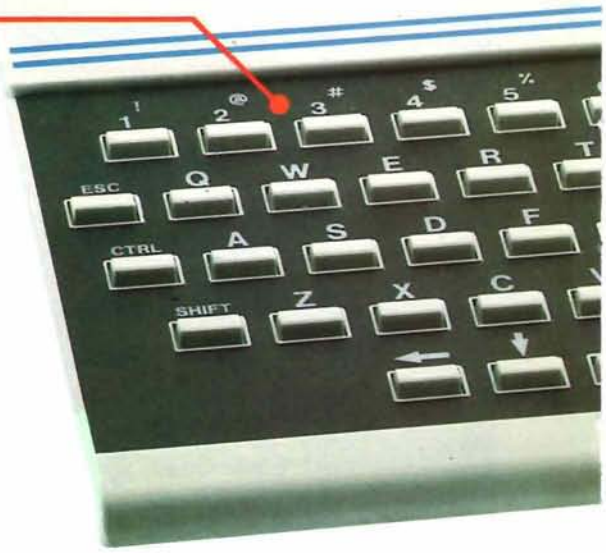
La figura mostra il numero 72 espresso in codice a barre. A prima vista non sono che piccole striscette nere di diverso spessore, ma, in questo caso, ogni gruppo consiste di cinque barre, due delle quali più larghe delle altre. La particolare posizione delle strisce più spesse rappresenta un numero. Strisce aggiuntive segnano l'inizio e la fine di ciascuna unità di informazione. Nell'esempio, il numero 72 è rappresentato in due unità: la cifra 7 e la cifra 2.
Esiste più di un sistema per la codifica a barre. Poiché una barra può essere o spessa o fine, vengono spesso associati a queste due condizioni i valori binari 0 e 1, così facilmente comprensibili al computer. Un diverso codice viene impiegato nell'Universal Product Code, dove le barre hanno più di due spessori diversi. In questo caso il numero di barre occorrenti è inferiore al precedente sistema.
La ragione di una così ampia diffusione del codice a barre sta nella facilità con la quale esso è leggibile dalle macchine: basta una frazione di secondo per far scorrere una penna ottica sul codice. Il principio di funzionamento è molto semplice: i tratti neri del codice non riflettono la luce, al contrario di quelli bianchi. Ciò viene convertito dai lettori in segnali elettrici, comprensibili dal computer

L'Oric-1

Anatomia di un microcomputer di diffusione europea: tecnologie a confronto

L'Oric-1 è un piccolo computer fabbricato in Gran Bretagna, che appartiene alla stessa categoria dello Spectrum Sinclair, sia per il prezzo che per le sue caratteristiche. Nel contenitore della tastiera, piuttosto elegante, di colore grigio, e comodamente inclinata, trovano posto i vari circuiti ed i tasti a movimento indipendente.
Sono disponibili due versioni, di cui la più completa dispone di una memoria di 48 Kbyte, sufficienti per programmi abbastanza complessi.
L'Oric ha i soliti connettori per il collegamento al televisore, al registratore o ad altre periferiche. Già predisposto per l'uso di una stampante, offre anche la possibilità di aggiungere estensioni di memoria, cartucce con programmi e, volendo, un modem.
Il BASIC è incluso come standard, ma è possibile impiegare anche altri linguaggi. Nella versione a 48K, oltre al BASIC, viene fornito anche il FORTH.
La grafica a colori e gli effetti sonori vengono gestiti da comandi BASIC. Si possono visualizzare otto colori ed è possibile definire e memorizzare caratteri di qualsiasi forma. I comandi INK e PAPER permettono di modificare il colore sia del carattere che dello sfondo sul quale esso viene rappresentato.
La generazione di effetti sonori è flessibile quanto la grafica: con speciali comandi si può creare una grande varietà di suoni ed anche far musica. Le note e gli accordi variano su più di sei ottave.
L'Oric, considerato il suo basso costo, è un computer di notevoli qualità. Le espansioni disponibili — in particolare quella che permette lo scambio di dati al telefono — ne fanno un apparecchio veramente versatile.

Tastiera
La tastiera dell'Oric ha 57 tasti indipendenti. Le lettere ed i numeri sono disposti nello standard QWERTY. I tasti ESCAPE e CONTROL sono a sinistra, mentre i tasti DELETE e RETURN si trovano a destra. I tasti di controllo del cursore e la barra spaziatrice sono collocati nella parte più in basso della tastiera. Poiché la disposizione è molto simile a quella di una normale macchina da scrivere (inglese), non è difficile raggiungere apprezzabili velocità di battitura



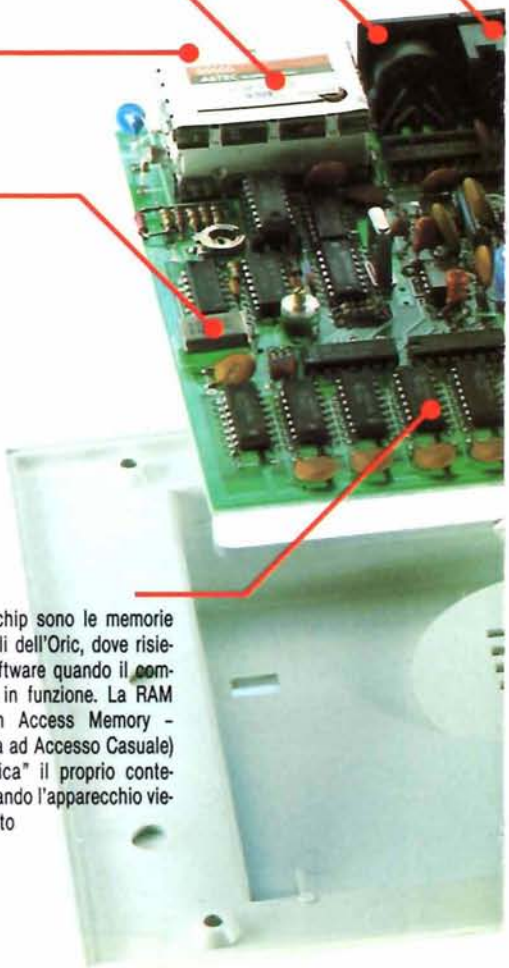
Presa RGB
L'Oric dispone di una presa per i monitor a colori che usano tre segnali separati per una migliore qualità di immagini

Presa per registratore
Il registratore a cassette si collega a questa presa

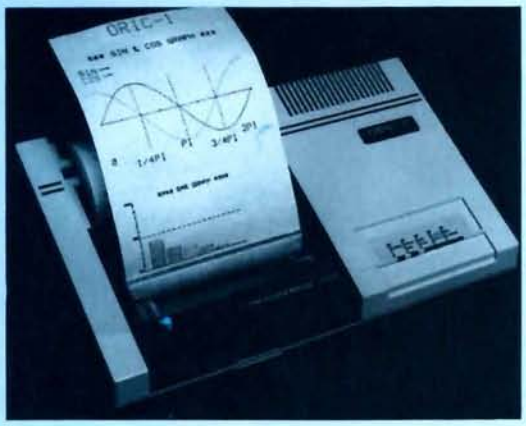
Modulatore RF
Questo particolare circuito converte il segnale generato dal computer in un segnale che può essere direttamente applicato alla presa dell'antenna di un televisore

Presa TV
Il cavo di connessione al televisore viene collegato a questa presa

Clock
Per sincronizzare tutte le operazioni dell'Oric, viene impiegato un particolare tipo di orologio (Clock) che oscilla un milione di volte al secondo



RAM
Questi chip sono le memorie principali dell'Oric, dove risiede il software quando il computer è in funzione. La RAM (Random Access Memory - Memoria ad Accesso Casuale) "dimentica" il proprio contenuto quando l'apparecchio viene spento



La stampante

La stampante, a colori, è stata disegnata con lo stesso stile dell'Oric-1. È in grado di produrre stampati e disegni in quattro colori: rosso, verde, blu e nero.
La stampante impiega quattro penne a sfera (una per colore) ed un rullo di carta (di 11 cm. di larghezza). La scrittura dei caratteri può avvenire scegliendo tra 15 diverse dimensioni e 4 angolazioni



Interfaccia per stampante
Al connettore del circuito d'interfaccia è possibile allacciare un dispositivo di stampa

Porta per espansioni
Questo connettore consente il collegamento ad un vasto numero di periferiche, compresi apparecchi di telecomunicazione

Presa per l'alimentazione
Questa è la presa per la corrente proveniente dall'alimentatore esterno

Dissipatore
I circuiti elettronici possono venir danneggiati da temperature troppo alte. Il dissipatore provvede a disperdere la temperatura generata durante il funzionamento

ROM
Nella ROM (Memoria a sola lettura) risiede in forma permanente una parte del software. Generalmente il linguaggio BASIC è conservato nelle ROM

Microprocessore
Il motore del computer è questo microprocessore, del tipo 6502A

Input/Output
Questo circuito integrato converte i segnali in ingresso in una forma comprensibile alla CPU dell'Oric ed i segnali in uscita in una forma comprensibile alla stampante (o ad altre periferiche)

Sintetizzatore
Questo speciale circuito genera tutti gli strabilianti effetti sonori

Circuito Video
Nell'Oric un solo chip sovrintende alla generazione delle immagini a colori, che risultano ottime. Il modulatore provvede ad inviare i segnali al televisore

Altoparlante
La qualità degli effetti sonori è molto apprezzabile nell'Oric, che è fornito di un altoparlante interno

ORIC-1

DIMENSIONI

mm. 280 x 178 x 150

PESO

848 grammi

CLOCK

1 MHz

MEMORIA

16 Kbyte di RAM estendibile a 48 Kbyte. 16 Kbyte di ROM contengono il BASIC

VIDEO

Ha 4 modi di funzionamento: scrittura e due modi grafici su 28 righe di 40 caratteri ed un modo grafico ad alta risoluzione (200 x 240 punti)

CPU

6502A

TASTIERA

57 tasti a movimento indipendente, con barra spaziatrice

INTERFACCE

Connettore per TV, per registratore a cassette e per amplificatore HI-FI. Interfaccia per stampante Centronics, per video RGB, connettore per espansioni

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

FORTH

ACCESSORI COMPRESI

Alimentatore, cavi di connessione per TV e per registratore a cassette, un nastro con programma dimostrativo, un manuale

DOCUMENTAZIONE

Il manuale dell'Oric comprende una brillante presentazione dell'apparecchio e del BASIC adoperato. La parte dedicata al BASIC spiega il trattamento dei dati, l'uso della grafica a colori e del generatore di effetti audio.
Sono riportati numerosi programmi dimostrativi di livello elementare e le appendici costituiscono una guida rapida alla consultazione. Si tratta, nel complesso, di un buon manuale per principianti

I puntini sulle i

Il BASIC è facile da usare, ma occhio alla punteggiatura!

Nei programmi presentati nelle prime lezioni (vedere listato a pag. 20) si sarà notato un punto e virgola in fondo alla linea 50. In questa lezione viene spiegata la funzione di questo importante particolare del BASIC. In quasi tutti i BASIC, il punto e virgola viene inserito quando si desidera concatenare la visualizzazione di più elementi di stampa. Le linee 50 e 60 di pagina 20 sono:

```
50 PRINT "PENSO CHE IL NUMERO DIGITATO FOSSE";
60 PRINT A
```

La linea 50 provvede a visualizzare il messaggio tra doppi apici, mentre la 60 visualizza il contenuto della variabile A. L'inserimento del punto e virgola provoca la visualizzazione del contenuto della variabile A sulla stessa riga ed immediatamente dopo la scritta compresa tra i doppi apici. Se l'avessimo omesso, quindi, il valore di A verrebbe visualizzato nella prima posizione della riga successiva. Il programma che segue, particolarmente adatto per dimostrare l'utilità del punto e virgola nel BASIC, serve alla conversione di temperature da gradi Celsius a Fahrenheit. Per chiarezza, d'ora in poi ometteremo il simbolo <CR> nei listati dei programmi, anche se ogni riga deve essere sempre terminata con la pressione del tasto RETURN (o ENTER).

```
10 REM CONVERSIONE DA GRADI C AD F
20 PRINT "INTRODURRE LA TEMPERATURA MINIMA"
30 INPUT L
40 PRINT "INTRODURRE LA TEMPERATURA MASSIMA"
50 INPUT H
60 FOR X = L TO H
70 LET F = X * 9 / 5 + 32
80 PRINT  X;" IN GRADI CELSIUS E' ";F;
   " IN GRADI FAHRENHEIT"
90 NEXT X
100 END
```

Impostato il programma e verificatane la correttezza con un LIST, lo si esegua con un RUN. Viene chiesto di introdurre un valore minimo: proviamo -6; poi un valore massimo: digitiamo 10. Automaticamente il programma converte, di grado in grado, tutti i valori dell'intervallo in gradi Fahrenheit. Sullo schermo dovrebbe apparire:

```
-6 In gradi celsius è 21.2 in gradi fahrenheit
-5 In gradi celsius è 23 in gradi fahrenheit
-4 In gradi celsius è 24.8 in gradi farhenheit
```

Si noti che, anche se l'incolonnamento dei valori non è molto preciso (a causa delle cifre decimali), ogni temperatura in Celsius e l'equivalente in Fahrenheit vengono visualizzate sulla stessa riga. Dopo aver fatto "girare" il programma più volte fornendo intervalli diversi, si provi a cambiare tutti i punti e virgola della linea 80 in virgole. Basta una sola prova per accorgersi che la visualizzazione è diventata piuttosto caotica.
Per poterne comprendere il perché, proviamo questo breve programma, dopo aver impartito il comando NEW (per cancellare quello precedente):

```
10 REM CONFRONTO TRA; E,
20 PRINT "QUESTA RIGA USA IL PUNTO E VIRGOLA"
30 PRINT "A"; "I"; "U"; "T"; "O"
40 PRINT "QUESTA RIGA USA LA VIRGOLA"
50 PRINT "A", "I", "U", "T", "O"
60 END
```

Quando il BASIC esegue la linea 30, sullo schermo appare AIUTO, mentre l'esecuzione della linea 50 visualizza A I U T O. Si veda la rubrica "Basic & Basic" per le varianti tra computer e computer. La virgola ha una molteplice funzione in BASIC, ma quando viene usata nel contesto di una stampa (su video o su stampante) fa sì che ogni elemento di stampa venga visualizzato secondo una elementare forma di tabulazione (è come se avessimo un tabulatore automatico ogni 8 colonne). Omettendo sia le virgole che i punti e virgola, gli elementi verrebbero visualizzati uno per riga.
Oltre a mostrarci l'uso del punto e virgola, il nostro programma di conversione ripresenta alcune istruzioni già incontrate nelle prime lezioni: nelle linee 30 e 50 alle variabili L ed H vengono assegnati, rispettivamente, il valore minimo e massimo dell'intervallo desiderato. La linea 60 forma la prima parte di un ciclo FOR-NEXT, in cui vengono adoperate direttamente le variabili L ed H al posto di costanti numeriche. Ciò facilita l'esecuzione del programma in quanto le due variabili vengono immesse, di volta in volta, mediante le istruzioni INPUT L e INPUT H. Con i valori impiegati

nell'esempio, (-6 e -4) l'inizio del ciclo equivale a: FOR X = -6 TO -4.
Nella linea 80 viene visualizzato (PRINT) il contenuto della variabile X (che inizialmente è pari alla temperatura minima e successivamente aumenta di un grado per volta fino a raggiungere la temperatura massima). Sulla stessa riga di stampa, grazie al nostro punto e virgola, viene visualizzato il messaggio tra doppi apici ed, infine, il valore della variabile F. Il valore di F è calcolato nella linea 70. La linea con l'istruzione NEXT ci assicura che il ciclo venga ripetuto fino al raggiungimento della temperatura massima da noi fornita.
Prima di illustrare alcune variazioni dell'istruzione PRINT, vale la pena di soffermarsi sulla linea:

```
70 LET F = X * 9/5 + 32
```

dove viene calcolato il valore della variabile F. Il valore di X (la temperatura in Celsius) viene moltiplicato per 9 e diviso per 5. Al risultato viene sommato 32. La formula, espressa in termini matematici, è: F = C x 9 : 5 + 32. Il BASIC usa i seguenti simboli: * per la moltiplicazione, / per la divisione, + per la somma e - per la sottrazione.
In BASIC, come del resto nella normale aritmetica, l'ordine secondo il quale vengono eseguite le operazioni è importantissimo e segue una precisa regola. Nelle quattro operazioni, la moltiplicazione viene eseguita per prima, poi viene la divisione, quindi la somma ed, infine, la sottrazione. Se una parte dell'espressione è racchiusa fra parentesi, questa viene eseguita prima delle altre. Volendo eseguire la somma prima della moltiplicazione, dobbiamo racchiudere la somma tra parentesi. Facciamo un esempio: abbiamo due conti in banca, e, se vogliamo conoscere il valore totale dei due conti espresso in dollari, scriveremo:

```
D = (C + S) * 1600
```

dove indichiamo con C il totale del primo conto, con S il totale del secondo. Questi due valori devono essere prima sommati tra loro ed il risultato deve essere poi moltiplicato per 1600 (supponendo un cambio di 1600 lire per un dollaro). La variabile D contiene il totale in dollari. Senza le parentesi, il valore nel conto S verrebbe moltiplicato per 1600, al risultato sarebbe sommato il valore (in lire) del conto C, il che produrrebbe un totale completamente errato. È sempre bene verificare che nei programmi non esistano errori di questo genere.

PRINT USING

Cerchiamo di migliorare la visualizzazione dei dati nel nostro programma di conversione. Se, infatti, forniamo un valore minimo di -10 ed un massimo di 10, l'incolonnamento dei risultati è tutto fuori posto. Ciò avviene perché il punto e virgola adoperato nel programma ci permette, sì, di visualizzare più elementi su di una stessa riga, ma non tiene conto, però, della lunghezza variabile di ciascun valore numerico.
Quasi tutti i BASIC hanno una variazione dell'istruzione PRINT: la PRINT USING. Questa ci consente di "formattare" la stampa (ossia di definire un formato di visualizzazione). Se vogliamo visualizzare il valore di una X e sappiamo che esso varia tra -99 e 99, il numero può essere incolonnato regolarmente usando l'espressione: PRINT USING "# # #"; X. I tre simboli # indicano che per il numero (segno compreso) devono essere riservate tre posizioni. Di conseguenza, se il numero risultasse di 4 o più cifre, l'incolonnamento sarebbe di nuovo compromesso. Se volessimo rappresentare un numero con una parte decimale, lo si può indicare adoperando un punto nel formato definito tra doppi apici, ad esempio: PRINT USING "# # #.# #"; X. Si usa un # per ogni carattere numerico, e se ne aggiunge uno in più se il valore può essere negativo. Attenzione: nel BASIC viene adoperata la notazione anglosassone: il punto serve ad indicare i decimali, mentre la virgola serve per separare le migliaia.
Possiamo ora modificare il programma nel seguente modo:

```
80 PRINT USING "# # #";X;
82 PRINT " IN GRADI CELSIUS È ";
84 PRINT USING "# # #";F
86 PRINT " IN GRADI FAHRENHEIT"
```

LISTiamo il programma per verifica, ed eseguiamo un RUN. Nella prossima lezione parleremo di come conservare i programmi, per non doverli riscrivere ogni volta.

Esercizi

* Si modifichi una delle linee, in modo che il programma termini se il valore immesso è 1000.
* Si modifichi la linea 84 in modo che vengano visualizzati soltanto numeri interi (senza parte decimale).

Basic & Basic

PRINT USING

Questa istruzione non è disponibile sui seguenti computer: VIC 20, Commodore 64, Spectrum, ZX81, Oric, BBC. Nel BBC, però, si può limitare il numero di cifre visualizzate usando la forma: @%: = 131594.

LA VIRGOLA

L'uso della virgola nel formato di stampa produce una visualizzazione incolonnata secondo una tabulazione fissa, generalmente di 8 spazi.
Sul BBC e sul Commodore 64 gli spazi sono 10, sul Dragon sono 16 e sull'Oric sono 4.

Quando 1 + 1 = 10

I computer eseguono calcoli prodigiosi, usando solo due numeri

Siamo talmente abituati al nostro sistema di numerazione decimale, che non ci rendiamo conto che esiste un'intera varietà di altri sistemi.
Gli antichi Romani adoperavano un metodo in cui i numeri venivano rappresentati con le lettere dell'alfabeto: X valeva il nostro 10, L era 50, C valeva 100 e via dicendo. Un simile sistema funzionava discretamente per registrare quantità numeriche, ma non era molto adatto per eseguire calcoli. Anche una semplice somma, in questa notazione, risulta difficoltosa. La principale ragione sta nel fatto che non esiste, nella numerazione romana, un'associazione tra il valore di un numero e la sua posizione. Nel sistema decimale, invece, la differenza tra due numeri, ad esempio 56 e 506, ci indica immediatamente che il numero 506 è composto da 5 centinaia, nessuna decina e 6 unità.
Ogni "colonna" (o posizione) individua un preciso arco di valori: la cifra più a destra le unità, poi le decine, le centinaia e così via... Il numero in ciascuna colonna indica quante unità, decine o centinaia ci sono nel numero totale.
Vediamo cosa c'entra tutto ciò con i computer ed il sistema binario. I computer sono apparecchi elettronici, che possono facilmente trattare valori numerici, purché questi siano trasformati in correnti elettriche. Ad esempio, una tensione di cinque Volt rappresenta un uno, mentre una tensione nulla rappresenta uno zero. Come abbiamo già appreso a pag. 28, con opportune sequenze di valori zero/uno è possibile rappresentare un qualsiasi numero di qualsiasi grandezza. Nel sistema decimale (o in base 10) il simbolo 506 è un modo conciso per rappresentare l'equivalente di, mettiamo, cinquecentosei nodi in una fune o palline in una scatola. Nel sistema binario lo stesso numero si esprime con un meno comprensibile 111111010.
Ciò accade perché nel metodo binario la posizione di ciascuna cifra indica un valore diverso da quello decimale: invece di muoversi in potenze di 10, le colonne incrementano il valore in potenze di 2.

LA STORIA DEI NUMERI



Babilonese

Gli antichi Babilonesi usavano un sistema a base 60 anziché 10. La figura mostra il numero 59 espresso in scrittura "cuneiforme".
Il sistema con base 60 aveva i suoi vantaggi e rimane tutt'oggi in uso per misurare le ore, i minuti e i valori angolari. Nel cerchio ci sono sei angoli di 60 gradi. Tutto ciò risale a più di 4000 anni fa.

Romano

Il sistema romano costituiva, invece, un passo indietro, perché la rappresentazione di numeri mediante lettere dell'alfabeto non associava la posizione al valore del numero, ostacolando così anche le più semplici operazioni aritmetiche.

Hindu

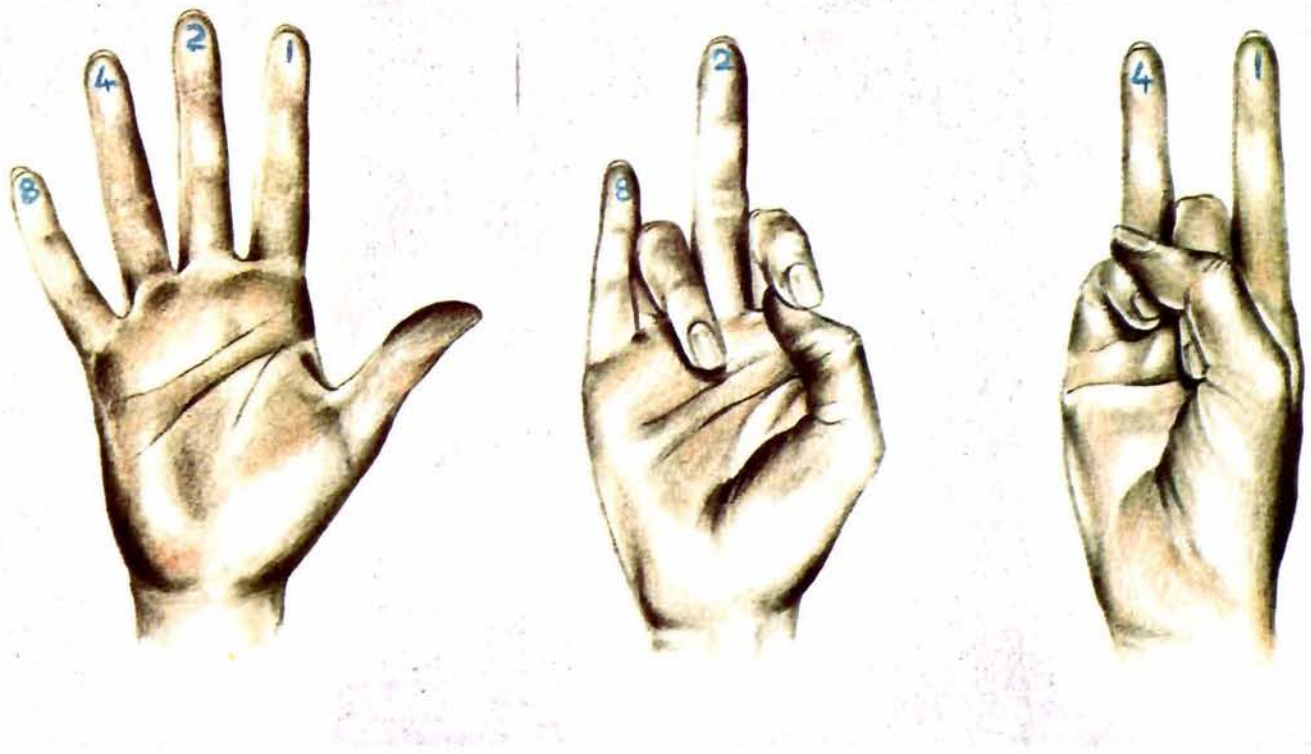
Gli Hindu adoperavano i simboli equivalenti ai numeri dall'1 al 9 ed in un secondo momento introdussero un simbolo per lo zero. L'enorme contributo di questo sistema è stato quello di aver introdotto il concetto di valore legato alla posizione del numero: il numero 30 vale 3 volte 10. Il sistema fu adottato dagli Arabi, e venne poi diffuso in Europa. Un eminente matematico arabo si chiamava Al Khowarizmi. Dalla latinizzazione del suo nome proviene il termine matematico "algoritmo", mentre dal titolo di un suo libro "Al-jabr wa'l Mugabalah" proviene il termine "algebra".

Binario

Il sistema binario è stato adottato per i computer, perché qualsiasi numero è rappresentabile da una sequenza di zero/uno

ANDY LESLIE

LA CONVERSIONE IN NUMERI BINARI



Il metodo più semplice per convertire piccoli numeri binari nell'equivalente decimale consiste nell'assegnare a ciascun dito della mano il "peso" di ciascuna colonna binaria.
Purché il numero binario sia al massimo di quattro cifre, basta piegare le dita corrispondenti alle colonne contenenti 0 per giungere al risultato.

Nella figura centrale le dita con peso 8 e 2 sono stese per rappresentare il valore binario 1010, l'equivalente è 10 (8 + 2), la figura più a destra mostra invece il valore 0101 equivalente a 5 (4 + 1). Proviamo, con questo metodo a convertire i valori binari: 1110 e 0110. Usando ambedue le mani si possono convertire numeri più grandi, in questo caso i "pesi" sulle dita della seconda mano saranno 16, 32, 64 e 128

ELAINE KEENAN

La colonna più a destra indica ancora le unità, ma appena sommiamo 1 siamo già a corto di numeri (esistendo, come si è detto i soli valori 0 ed 1). Nel sistema decimale possiamo invece incrementare questo valore fino a 9, dopo di che occorre inserire un numero nella colonna immediatamente a sinistra, per indicare che abbiamo superato le unità e siamo nelle decine. Nel sistema binario il discorso è simile, ma il valore 10 indica il decimale 2. Diamo un'altra occhiata al numero 506:

Centinaia	Decine	Unità	
5	0	6	
= 5 x 100	0 x 10	6 x 1	(= 506)

256	128	64	32	16	8	4	2	1	
1	1	1	1	1	1	0	1	0	
= 1 x 256 + 1 x 128 + 1 x 64 + 1 x 32 + 1 x 16 + 1 x 8 + 0 x 4 + 1 x 2 + 0 x 1									(= 506)

Le regole dell'aritmetica binaria sono le stes-

numeri 0 ed 1 invece dell'intervallo da 0 a 9. Proviamo con una somma. I valori tra parentesi sono l'equivalente decimale:

(3)	11
+ (5)	101
(8)	1000

(1 + 1 = 0 riporto 1)
(riporto + 1 = 0 riporto 1)
(riporto + 1 = 0 riporto 1)
(riporto + 0 = 1)

Nel metodo binario, come si è appena visto, sommando 1 e 1 ci dà 0 con riporto 1. Ecco un'altra somma spiegata, e due da usare per esercizio:

(4)	100
+ (6)	110
(10)	1010

(7)	111	(3)	11
+ (2)	10	+ (12)	1100
(?)	(?)	(?)	(?)

I numeri binari sono molto più lunghi del loro

equivalente decimale. Si provi a sommare

Velocità di risposta

Gli accessori da gioco per i computer

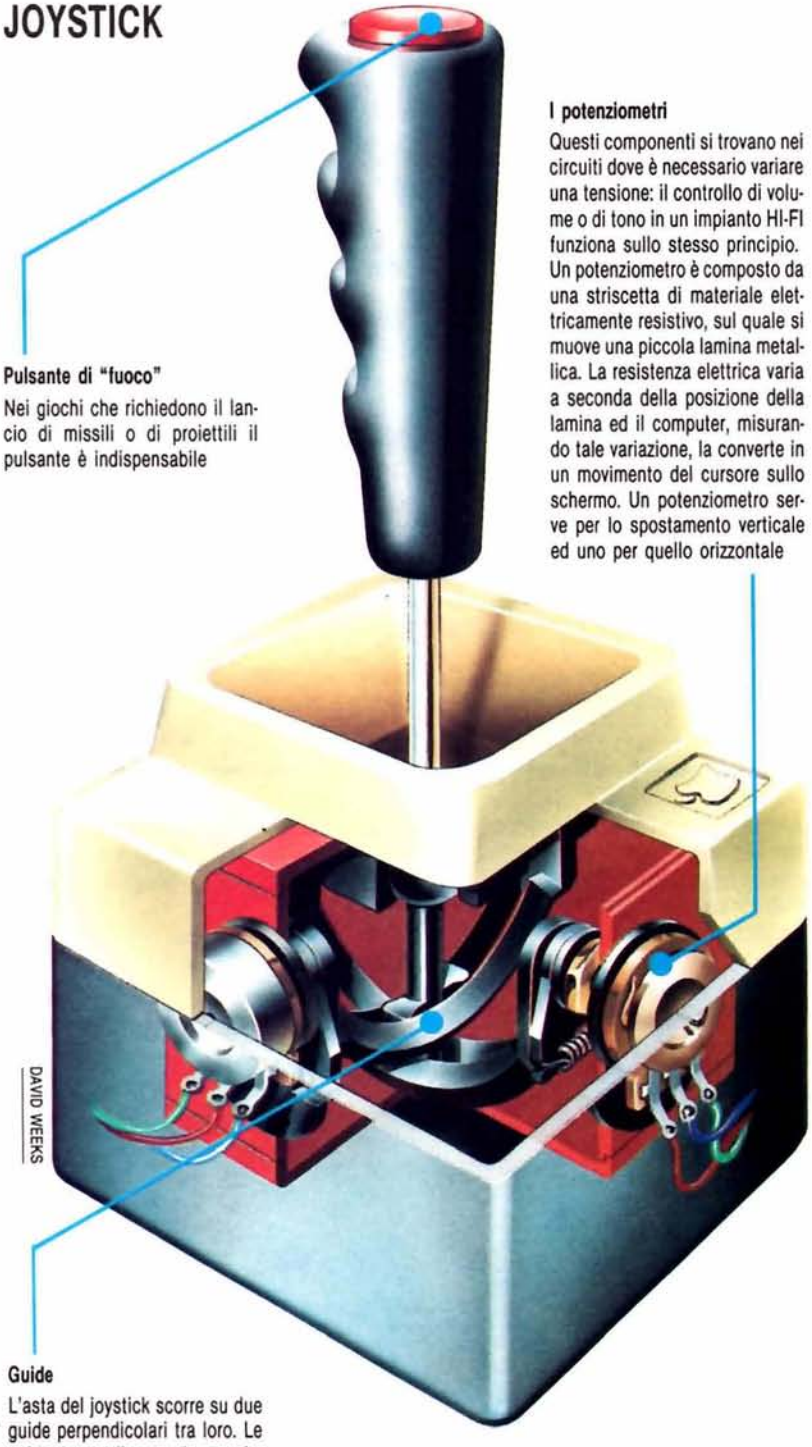


IAN McKINNELL

La maggior parte dei giochi che si fanno sul computer, ci pongono alla guida di sofisticate navi spaziali pronti a sparare missili e raggi laser contro degli obiettivi in movimento. Dalla nostra prontezza di riflessi e dalla possibilità di spostare rapidamente il nostro veicolo sullo schermo dipende il buon esito del gioco. L'uso di joystick è molto più efficace, a questo scopo, che non l'immissione di dati attraverso la normale tastiera del computer. I joystick sono degli accessori collegabili all'apparecchio, che riproducono talvolta molto realisticamente, la "cloche" di un aereo. Manovrando opportunamente il joystick possiamo spostare istantaneamente la nostra nave spaziale nelle varie zone dello schermo. Il funzionamento di un joystick è molto semplice: la cloche, spostandosi, chiude uno o più interruttori che comunicano al computer la direzione scelta. Nelle versioni più complete, al posto di semplici interruttori, vengono usati dei potenziometri, per consentire un controllo più preciso dei movimenti. Quasi tutti i joystick dispongono di uno o più pulsanti di "fuoco", per sparare le nostre munizioni.

Ai computer più economici, in genere, non è possibile collegare i joystick, se non usando una speciale "interfaccia". L'interfaccia è un particolare circuito che adatta l'uscita del joystick ad un ingresso ausiliario del computer. In questo caso, però, i programmi devono essere scritti appositamente per l'uso dei joystick e della speciale interfaccia.

JOYSTICK



Pulsante di "fuoco"
Nei giochi che richiedono il lancio di missili o di proiettili il pulsante è indispensabile

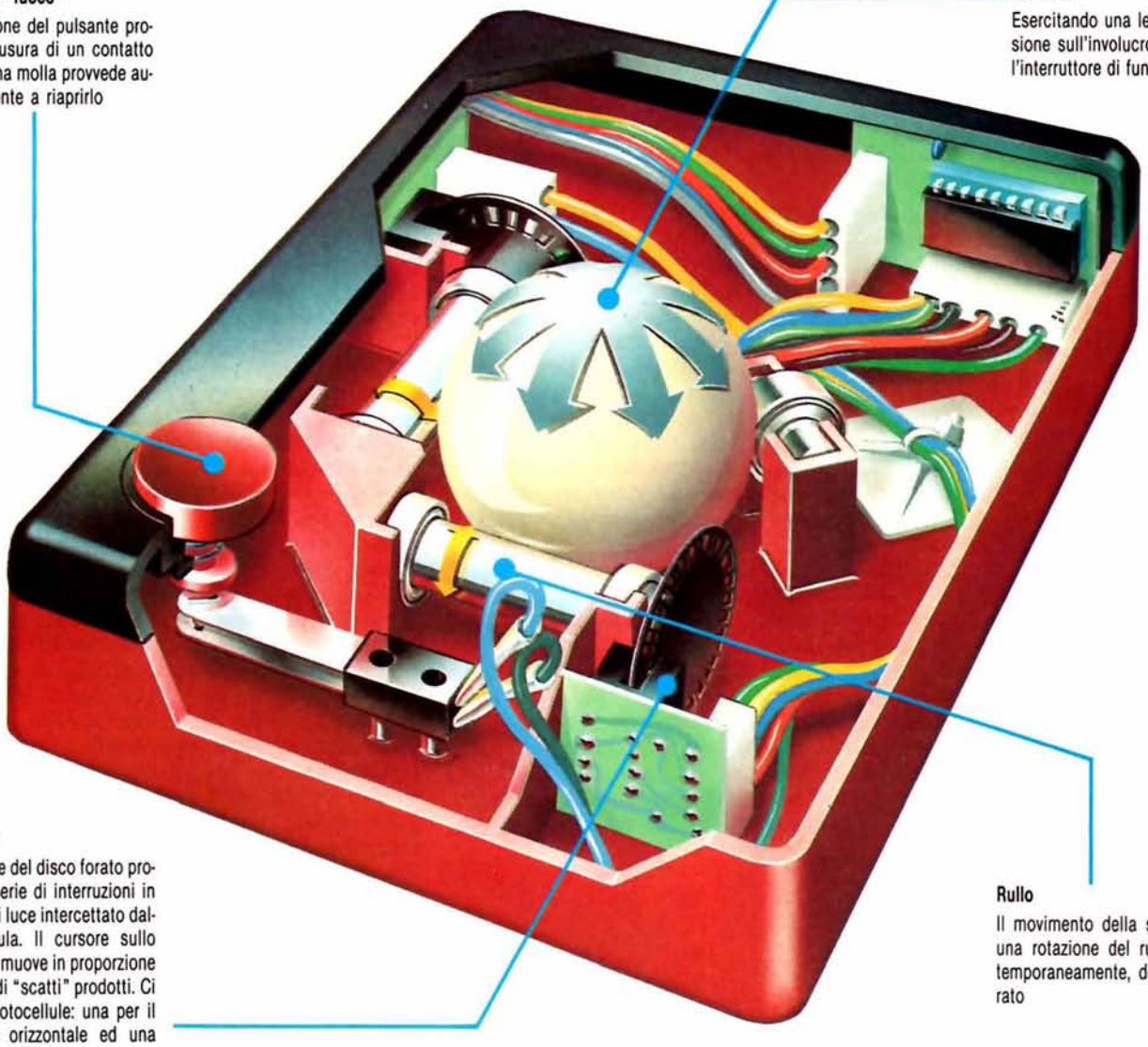
I potenziometri
Questi componenti si trovano nei circuiti dove è necessario variare una tensione: il controllo di volume o di tono in un impianto HI-FI funziona sullo stesso principio. Un potenziometro è composto da una striscetta di materiale elettricamente resistivo, sul quale si muove una piccola lamina metallica. La resistenza elettrica varia a seconda della posizione della lamina ed il computer, misurando tale variazione, la converte in un movimento del cursore sullo schermo. Un potenziometro serve per lo spostamento verticale ed uno per quello orizzontale

DAVID WEEKS

Guide
L'asta del joystick scorre su due guide perpendicolari tra loro. Le guide sono collegate ai potenziometri in modo da variarne la resistenza a seconda della posizione dell'asta

Pulsante di "fuoco"
Una pressione del pulsante provoca la chiusura di un contatto elettrico. Una molla provvede automaticamente a riaprirlo

La sfera
Esercitando una leggera pressione sull'involucro, si chiude l'interruttore di funzionamento



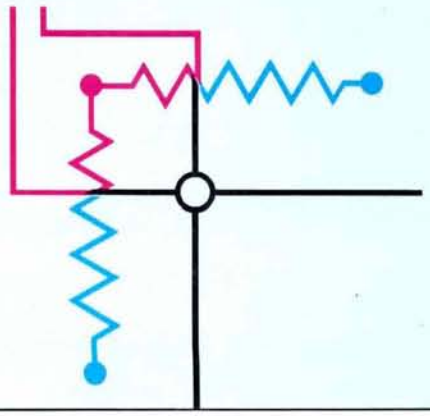
Fotocellula
La rotazione del disco forato produce una serie di interruzioni in un fascio di luce intercettato dalla fotocellula. Il cursore sullo schermo si muove in proporzione al numero di "scatti" prodotti. Ci sono due fotocellule: una per il movimento orizzontale ed una per quello verticale

Rullo
Il movimento della sfera causa una rotazione del rullo e, contemporaneamente, del disco forato

La track ball

Immaginiamo di dover muovere il cursore sullo schermo attraverso un labirinto: il percorso causa repentini cambiamenti di direzione, difficili da controllare mediante una tastiera. La Track Ball (chiamata anche Topo o Sfera) agevola notevolmente la gestione di simili spostamenti. La track ball impiega una sfera, che viene fatta scorrere su di una superficie: il movimento viene riprodotto immediatamente dal cursore sullo schermo. Nel contenitore, il movimento della sfera provoca una rotazione di due "rulli" perpendicolari tra loro: uno intercetta i movimenti in senso verticale, l'altro quelli in senso orizzontale. Il segnale elettrico generato viene ricomposto dal computer per riprodurre il movimento sullo schermo.

CIRCUITO DEL POTENZIOMETRO

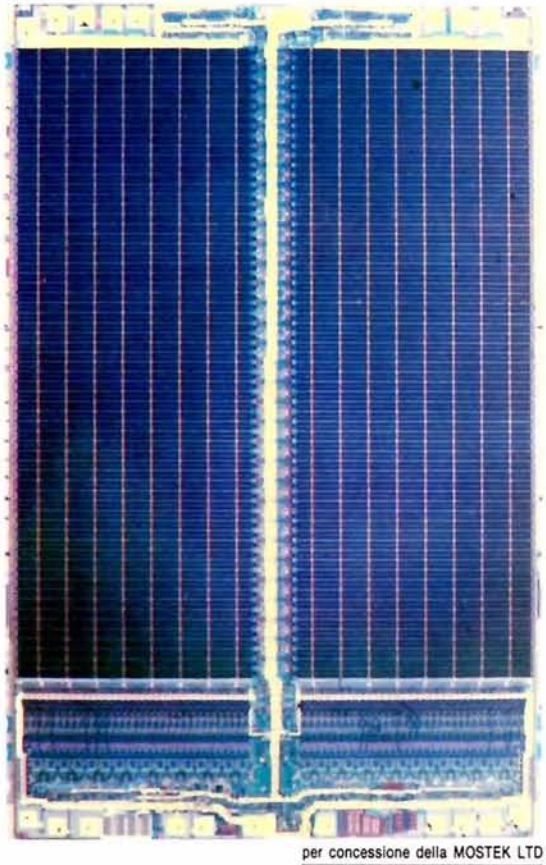


La leva del joystick è collegata a due resistenze variabili (chiamate potenziometri). Il movimento della leva genera uno spostamento del punto di contatto su una o ambedue le strisce resitive (indicate con la linea a zig-zag) variando quindi la resistenza elettrica del circuito. Il computer controlla e traduce le variazioni di tensione in spostamenti del cursore sullo schermo

Memoria di ferro

Come è organizzata e gestita la memoria di un computer

Nell'uomo la memoria è il magazzino del pensiero: vi si conservano tutti i dettagli dell'esperienza da essere riutilizzati in futuro. Nel computer il termine "memoria" ha un significato pressoché identico, anche se esistono alcune precise limitazioni.
Una memoria poco efficiente, nell'uomo, può essere fonte di difficoltà o di imbarazzo, mentre nel computer porta a conseguenze disastrose. Un computer senza memoria non serve più a nulla: non avrebbe niente su cui lavorare, né istruzioni da eseguire, poiché anche i programmi vengono conservati nella memoria.
In ambedue i casi la parola "memoria" implica due concetti: la conservazione ed il richiamo di informazioni. Conservare informazioni senza la possibilità di richiamarle è totalmente inutile, come del resto tentare di richiamarle se non si ha la capacità di conservarle.
L'analogia con la memoria umana non si ferma qui. L'uomo possiede due generi di memoria: a breve ed a lungo termine. Se dobbiamo attraversare una strada, ci ricordiamo di attendere che tutte le auto siano passate, ma, giunti sul marciapiede opposto, il ricordo delle auto viene subito cancellato: questa è memoria a breve termine.
Ma se, per esempio, tra le auto ne passa una con a bordo uomini mascherati può darsi che il fatto ci rimanga impresso nella memoria, assieme, magari, al colore ed al tipo di auto. Questa è memoria a lungo termine.
Ebbene, anche nel computer si può fare una simile distinzione. La memoria a lungo termine (detta anche "non-volatile") contiene programmi e dati che l'utente desidera conservare. A questo servono le memorie "di massa", ovvero nastri e dischi magnetici, nastri o schede perforate o cartucce ROM.
La memoria a breve termine (o "volatile") si identifica invece con la RAM interna al computer, che rimane attiva soltanto durante il funzionamento dell'apparecchio. Appena viene tolta la corrente, anche per una frazione di secondo, tutto il contenuto della memoria viene irrimediabilmente perso.
Le analogie con la mente umana si fermano a questo punto. Nel computer, infatti, i programmi ed i dati vengono trasferiti dalla memoria di massa alla memoria interna per essere usati. Inoltre il sistema adottato per il richiamo e la conservazione di queste informazioni è totalmente diverso da ciò che accade nell'uomo.
Alcuni meccanismi della mente umana sono finora ignoti, anche perché non riusciamo ancora a stabilire con esattezza come vengano distribuite e conservate le informazioni nel cervello. Del resto, per ricordare qualcosa, non ci è necessario sapere dove questo qualcosa è nel nostro cervello, né dove riporlo una volta usato.



per concessione della MOSTEK LTD

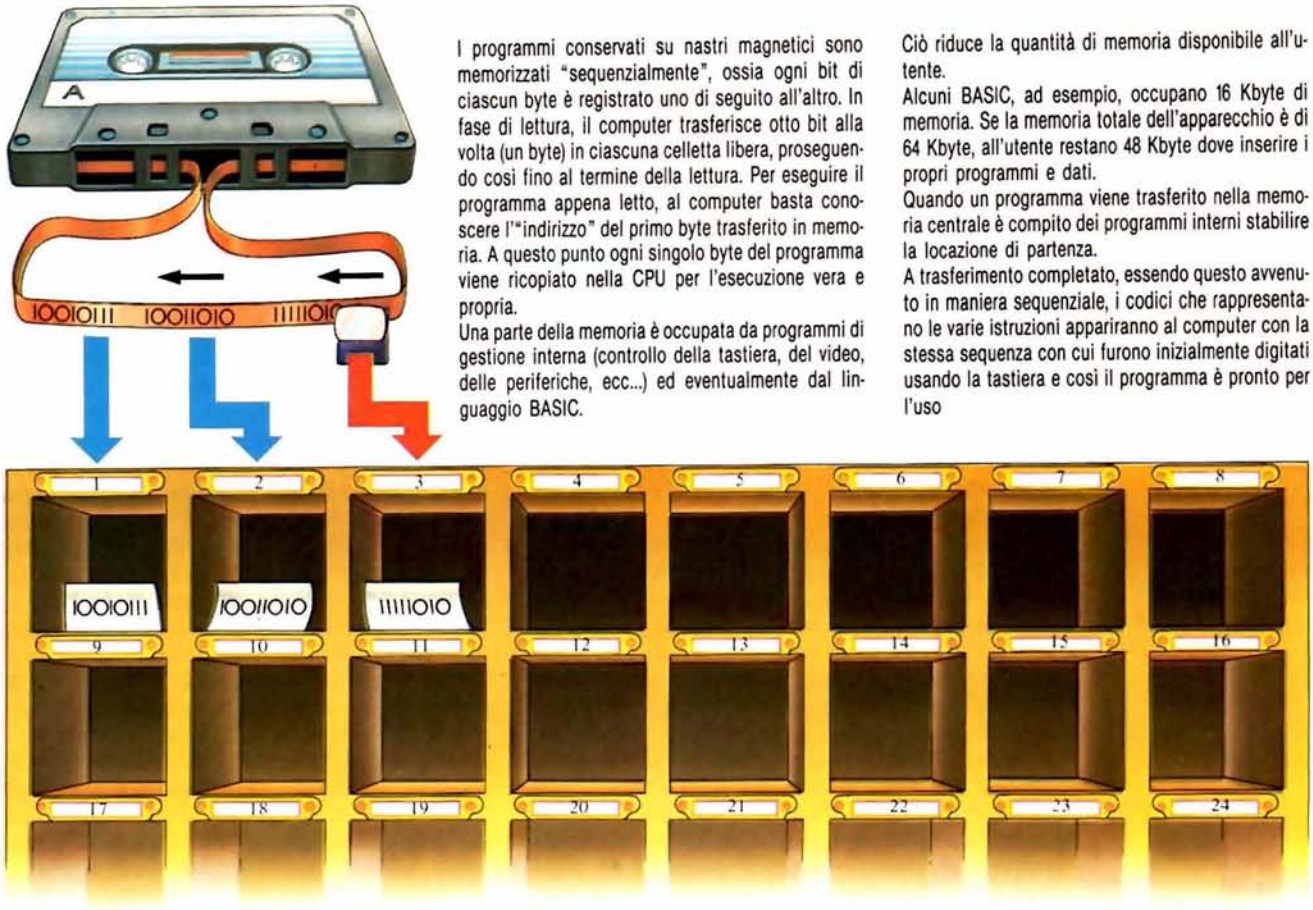
Memoria RAM

Il chip della RAM (illustrato nella foto) è una delle più importanti innovazioni tecnologiche nel campo dei computer. La Random Access Memory (Memoria ad accesso casuale) appartiene al tipo di memoria totalmente elettronica, della quale fa parte anche la Read Only Memory (Memoria a sola lettura). I nastri e i dischetti magnetici fanno parte, invece, delle memorie di tipo elettromagnetico.
La memoria RAM viene realizzata da strati di silicio, sui quali sono creati, attraverso processi fotografici e chimici, migliaia di microscopici transistor. È necessario un transistor per ogni bit di memoria.
Il tempo necessario a "scrivere" il valore di un singolo bit in una qualsiasi delle 16.383 cellette è di circa 200 nanosecondi (cinque milionesimi di secondo)

Un po' d'ordine nel caos

Nella memoria di un computer, la locazione ove ciascun elemento è conservato è di estrema importanza, poiché deve esistere un sistema per individuare un particolare byte tra le migliaia raggiungibili. Inoltre il computer deve "annotarsi" la collocazione di una particolare informazione.
La memoria umana può paragonarsi ad una scatola piena di informazioni senza alcuna particolare organizzazione. I ricordi vengono accumulati senza un apparente ordine, talvolta mescolandosi ad altri. Il nostro cervello è perfettamente in grado di muoversi con precisione in questa bolgia per ripescare un particolare ricordo al momento opportuno.

LA MEMORIA



TONY SLEEP

I programmi conservati su nastri magnetici sono memorizzati "sequenzialmente", ossia ogni bit di ciascun byte è registrato uno di seguito all'altro. In fase di lettura, il computer trasferisce otto bit alla volta (un byte) in ciascuna celletta libera, proseguendo così fino al termine della lettura. Per eseguire il programma appena letto, al computer basta conoscere l'"indirizzo" del primo byte trasferito in memoria. A questo punto ogni singolo byte del programma viene ricopiato nella CPU per l'esecuzione vera e propria.

Una parte della memoria è occupata da programmi di gestione interna (controllo della tastiera, del video, delle periferiche, ecc...) ed eventualmente dal linguaggio BASIC.

Ciò riduce la quantità di memoria disponibile all'utente.

Alcuni BASIC, ad esempio, occupano 16 Kbyte di memoria. Se la memoria totale dell'apparecchio è di 64 Kbyte, all'utente restano 48 Kbyte dove inserire i propri programmi e dati.

Quando un programma viene trasferito nella memoria centrale è compito dei programmi interni stabilire la locazione di partenza.

A trasferimento completato, essendo questo avvenuto in maniera sequenziale, i codici che rappresentano le varie istruzioni appariranno al computer con la stessa sequenza con cui furono inizialmente digitati usando la tastiera e così il programma è pronto per l'uso

La memoria di un computer, invece, può essere paragonata ad un'enorme serie di cellette numerate a compartimenti stagni. Tutto è necessariamente ordinato: ogni celletta ha un suo numero (chiamato "indirizzo"), che contiene un solo byte. Il computer conserva e ritrova le informazioni in base al loro indirizzo, e non in base a ciò che la celletta contiene.

Poiché i computer non possiedono intelligenza, non possono organizzare e gestire la loro memoria per conto proprio. La ragione per la quale i computer sembrano, in pratica, svolgere questo delicato compito, risiede nel fatto che qualcuno ha provveduto a preparare in precedenza un certo numero di istruzioni che il computer esegue di volta in volta. Vediamo cosa succede effettivamente sul nostro computer.

All'accensione, di solito, appare un messaggio sullo schermo per avvertire che il computer è pronto all'uso. In certi casi, a questo punto si possono già inserire istruzioni in un particolare linguaggio. Il messaggio iniziale ed i programmi che lo visualizzano, assieme al linguaggio, sono conservati in una zona di memoria "a lungo termine", generalmente una ROM (vedere a pagina 9).

I programmi conservati in questa memoria sovrintendono al funzionamento della tastiera, del video e di tutte le altre parti dell'elaboratore. Un gruppo di programmi provvede, inoltre, a convertire le istruzioni BASIC in codici binari, rendendoli comprensibili alla macchina.

In alcuni computer, appare anche una scritta del tipo xxxxx bytes free, dove xxxxx vale, ad esempio, 15.797 o qualche altro numero. Questo messaggio ci indica quante "cellette" libere abbiamo nella memoria. Se iniziamo a scrivere istruzioni con la tastiera, il numero di byte liberi andrà via via diminuendo, fino ad esaurimento. Il che ci porta ad esaminare un altro aspetto molto importante: l'ordine con il quale vengono conservate le informazioni.

La pressione di un tasto provoca la memorizzazione del codice binario corrispondente al carattere (vedere a pagina 3).

Ma in quale celletta va a finire allora questo codice?

La risposta è semplice: nella prima locazione di memoria libera. Il carattere successivo viene memorizzato nella locazione seguente e così via.

Il computer dispone di uno speciale contatore per sapere, in ogni momento, quale sia la prossima locazione libera. L'uso di questo contatore è controllato da un particolare programma interno.

Pronta cassa

Le banche possono offrire i propri servizi 24 ore al giorno, grazie ai microcomputer

Grazie ad una delle applicazioni del microcomputer, i clienti delle banche hanno la possibilità di ritirare contanti in qualsiasi momento del giorno e quasi ovunque sul territorio nazionale. Basta avere con sé una speciale tessera di plastica, sulla quale sono codificate alcune informazioni concernenti il conto corrente del cliente. In caso di necessità il prelievo, usando un particolare dispositivo all'esterno delle banche, è faccenda di pochi minuti.

Dietro a questi apparecchi "dispensatori" di contante si nasconde un microcomputer. Il cliente, per effettuare il prelievo, deve introdurre la sua tessera in una fessura dell'apparecchio, dove un lettore, molto simile ad un registratore a nastro, esegue una serie di controlli sui codici magnetici incisi sulla tessera: in particolare, sulla validità (potrebbe trattarsi di una tessera falsa) e sul codice di identità del cliente (chiamato PIN, da Personal Identification Number). Il cliente viene quindi invitato a digitare su di un'apposita tastiera il proprio codice segreto (il PIN, appunto) che viene automaticamente confrontato con quello indicato sulla tessera. Se il codice corrisponde, vengono abilitate le operazioni successive, altrimenti viene ripetuto l'invito a fornire il codice ed al terzo tentativo fallito, il computer provvede automaticamente a ritirare la tessera ed a bloccare ogni ulteriore operazione.

Se il codice introdotto è esatto, il cliente viene invitato a digitare la somma che intende prelevare (fino ad un massimo di 500.000 lire). Il computer controlla (usando ancora i codici sulla tessera) se il prelievo supera il limite massimo mensile di 1.500.000 lire. In caso negativo, riscrive sulla tessera il nuovo saldo e provvede a preparare le banconote necessarie. Contemporaneamente, emette uno scontrino con i dati concernenti l'operazione appena effettuata. In alcuni casi il computer che sovrintende a tutte queste operazioni è collegato al computer centrale della banca, ove gli estremi del prelievo vengono automaticamente registrati tra i movimenti del conto corrente interessato. Presto saranno disponibili anche altri servizi accessori, quali, ad esempio, l'emissione di estratti conto per conoscere la situazione del proprio conto corrente.



L'uso del servizio "BANCOMAT" è stato introdotto per consentire prelievi (ed in futuro anche altre operazioni) perfino durante l'orario di chiusura delle banche. In caso di necessità i clienti possono ritirare somme di denaro dal proprio conto in qualsiasi momento del giorno o della notte, usando un codice individuale segreto

La parola giusta

Con i programmi adatti, il computer archivia, organizza ed elabora i testi



IAN McKINNELL

Il word processor portatile
Poiché il word processor sta diventando una delle applicazioni più diffuse sui microcomputer, vengono progettati nuovi apparecchi particolarmente orientati verso questa funzione. Le caratteristiche comprendono: schermo da 80 colonne, unità a dischetti incorporate e tastiere dotate di speciali tasti "funzione" per facilitare l'uso del word processor, che spesso viene compreso nel prezzo di vendita. Alcuni apparecchi sono addirittura portatili, offrendo a giornalisti, scrittori ed uomini d'affari, la possibilità di lavorare anche durante spostamenti in treno ed in aereo

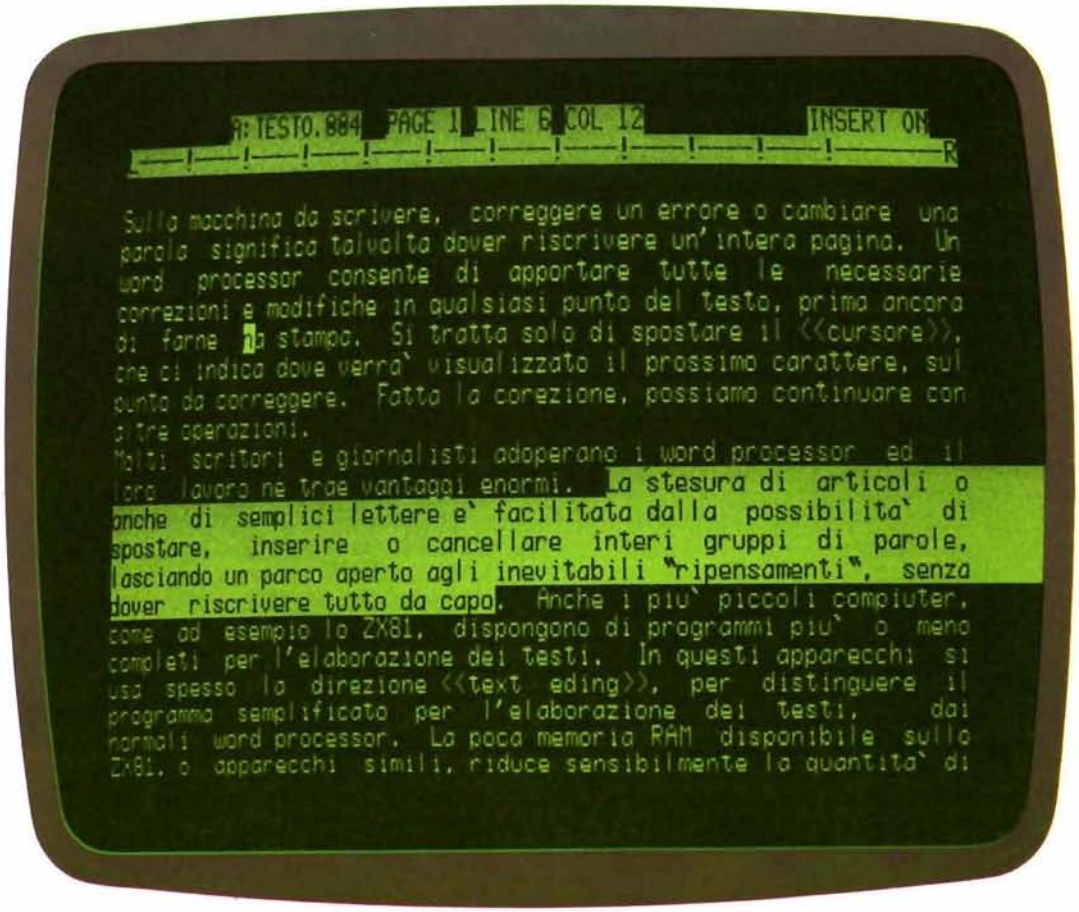
L'elaborazione dei testi è una delle utilizzazioni maggiormente apprezzabili di un computer, ma il grande pubblico si chiede ancora: «Cosa significa elaborare un testo?». Fino a poco tempo fa, il word processing veniva reclamizzato come un processo a sé stante, che necessitava di un particolare apparecchio in grado di svolgere quest'unico compito. In effetti, gli apparecchi per l'elaborazione dei testi spesso non sono che microcomputer sui quali "gira" un solo programma.
Forse, per una maggiore chiarezza, i programmi di word processing si sarebbero dovuti chiamare «dattilografia assistita dal computer» o con altra espressione analoga. Disponendo di una stampante e dell'opportuno programma, anche i piccoli computer possono servirci come unità più o meno sofisticate per l'elaborazione dei testi.
Usando il computer come word processor, sullo schermo appaiono le parole che noi componiamo alla tastiera, come se lo schermo fosse un normale foglio di carta inserito nella macchina da scrivere. Alcuni microcomputer mettono a disposizione uno schermo con 80 caratteri per riga, altri più economici costringono ad usare righe di soli 40 caratteri oppure non dispongono dei caratteri minuscoli. La quantità di testo da elaborare dipende, ovviamente, da quanta memoria è disponibile nel computer.
Un buon programma offre tutta una serie di utili comandi, alcuni automatici ed altri inseribili a piacimento. Quasi tutti hanno una

L'"EDITING" ELETTRONICO

Sulla macchina da scrivere, correggere un errore o cambiare una parola significa talvolta dover riscrivere un'intera pagina. Un word processor consente di apportare tutte le necessarie correzioni e modifiche in qualsiasi punto del testo, prima ancora di farne la stampa. Si tratta solo di spostare il //cursore//, che ci indica dove verrà visualizzato il prossimo carattere, sul punto da correggere. Fatta la correzione, possiamo continuare con altre operazioni.

Molti scrittori e giornalisti adoperano i word processor ed il loro lavoro ne trae vantaggi enormi. La stesura di articoli o anche di semplici lettere è facilitata dalla possibilità di spostare, inserire o cancellare interi gruppi di parole, lasciando un parco aperto agli inevitabili "ripensamenti", senza dover riscrivere tutto da capo. Anche i più piccoli compiuter, come ad esempio lo ZX81, dispongono di programmi più o meno completi per l'elaborazione dei testi. In questi apparecchi si usa spesso la direzione //text ediNg//, per distinguere il programma semplificato per l'elaborazione dei testi, dai normali word processor. La poca memoria RAM disponibile sullo ZX81, o apparecchi simili, riduce sensibilmente la quantità di testo sul quale si può lavorare.

I più grandi vantaggi di un sistema di word processing su computer, rispetto ai sistemi tradizionali, sono la maggiore velocità ed elasticità di lavoro consentite. Mentre elaborare testi alla macchina da scrivere comporta lunghi e non facili processi di correzione, con conseguenti necessità di ribattere (anche più volte) i testi fino alla stesura definitiva, con un word processor si lavora a velocità "elettronica", operando correzioni e modifiche al testo direttamente su di uno schermo. Sofisticati sistemi consentono anche la ricerca di determinate parole nel testo, lo spostamento di interi blocchi di testo da un punto all'altro, la correzione automatica di errori ortografici. In certi casi è addirittura possibile operare un controllo di tipo grammaticale sui testi. Non stupisce, perciò, che gradualmente, ma inesorabilmente, le antiquate macchine da scrivere vengano sostituite da questi sistemi. Nelle figure vengono confrontati il sistema tradizionale e quello che utilizza un word processor e nella prima colonna della pagina seguente, segnato da una linea rossa, il testo definitivo.



opzione chiamata "word wrap". Questa permette, una volta stabiliti i margini di un testo ovvero il numero di caratteri per riga, di non doversi più preoccupare degli "a capo": il computer si accorge da solo che il numero dei caratteri in una riga ha superato il massimo, riorganizza perciò le parole della riga spaziandole opportunamente e (se la funzione è disponibile) interrompe una parola troppo lunga nel punto giusto. I "ritorno carrello" sono necessari soltanto alla fine di un paragrafo.

Sulla macchina da scrivere, correggere un errore o cambiare una parola significa talvolta dover riscrivere un'intera pagina. Un word processor consente di apportare tutte le necessarie correzioni e modifiche in qualsiasi punto del testo, prima ancora di farne la stampa. Si tratta solo di spostare il "cursore", che ci indica dove verrà visualizzato il prossimo carattere, sul punto da correggere. Fatta la correzione, possiamo continuare con altre operazioni.

La stesura di articoli o anche di semplici lettere è facilitata dalla possibilità di spostare, inserire o cancellare interi gruppi di parole, lasciando un varco aperto agli inevitabili "ripensamenti", senza dover riscrivere tutto da capo. Molti scrittori e giornalisti adoperano i word processor ed il loro lavoro ne trae enormi vantaggi. Anche i più piccoli computer, come ad esempio lo ZX81, dispongono di programmi più o meno completi per l'elaborazione dei testi. In questi apparecchi si usa spesso la dizione "text editing", per distinguere il programma semplificato per l'elaborazione dei testi, dai normali word processor. La poca memoria RAM disponibile sullo ZX81 o su apparecchi simili riduce sensibilmente la quantità di testo sul quale si può lavorare. Inoltre, sia lo ZX81 che lo Spectrum non hanno tastiere particolarmente adatte a questo tipo di utilizzazione, in quanto la disposizione dei tasti non è di tipo tradizionale e non consentono, quindi, grandi velocità di battitura. Questo è un aspetto molto importante da considerare nella scelta di un computer (vedere a pagina 36). Una memoria RAM di 16 o 32 Kbyte è in genere sufficiente per elaborare testi di media lunghezza e, se l'apparecchio consente espansioni di memoria, tanto meglio.

Un altro aspetto da considerare è la "memoria di massa". È possibile usare nastri a cassetta per conservare testi, ma l'impiego di unità a dischetti è decisamente da preferirsi, perché in questo caso le parti di testo, eccedenti la memoria RAM a disposizione, vengono automaticamente "appoggiate" sul dischetto durante il lavoro, permettendo la stesura di testi molto più lunghi. Nel caso dello Spectrum, le micro unità a disco fornite dalla Sinclair rendono il lavoro un poco più pratico. I "microdrive" sono, in realtà, un incrocio tra un nastro a cassetta ed un dischetto. Viene impiegato un nastro magnetico senza fine, che, grazie a particolari sistemi di accesso, riduce ad un quarto i tempi di memorizzazione. Ciò nonostante, l'archiviazione di una pagina di testo richiede circa 6/7 secondi, mentre con unità a disco convenzionali il tempo impiegato è di circa un secondo.

La forza dei word processor risiede nel fatto che la fase di composizione e di correzione viene separata dalla fase di stampa del testo. Scrivendo a mano o con la macchina da scrivere, le parole vengono immediatamente impresse su carta: con un word processor, il testo viene riprodotto su carta soltanto quando la versione elaborata sullo schermo è giudicata definitiva; la gamma di stampanti disponibili sul mercato è molto vasta (vedere a pagina 74).

Molti costruttori si sono dedicati alla creazione di economici sistemi di word processing per gli "home" computer. In genere si tratta di speciali chip racchiusi in cartucce da collegare al computer. Se la memoria RAM è nell'ordine dei 32 o più Kbyte, è possibile elaborare testi di circa 5000 parole.

Poiché il chip contiene soltanto il word processor, la conservazione dei testi deve necessariamente avvenire sulla memoria di massa a disposizione e, se questa è costituita da nastri magnetici, la procedura può richiedere diversi minuti.

Alcuni sofisticati word processor possono svolgere utilissime funzioni accessorie. In certi casi, ad esempio, è possibile consultare un dizionario durante la stesura di un testo oppure, una volta redatto lo scritto, si può far eseguire un controllo per individuare errori di compilazione: se una parola non viene "riconosciuta", il programma invita l'operatore a correggerla oppure a confermare che si tratta di un nuovo vocabolo, da aggiungere al dizionario. Questo genere di funzioni richiede necessariamente l'uso di unità a dischi.

L'uso del word processor, inizialmente limitato agli uffici ed alle attività commerciali in genere, si sta rapidamente estendendo anche tra utilizzatori dilettanti. Le macchine da scrivere finiranno presto nei musei?

I word processor disponibili per l'home computer

Quasi ogni marca di home computer ha disponibile una versione più o meno sofisticata per l'elaborazione di testi. Se questa costituisce uno delle applicazioni principali del computer, occorre sceglierne uno con la tastiera di tipo tradizionale (vedere la colonna "tastiera" in tabella). La stampante è un accessorio indispensabile e ve ne sono di vari tipi (vedere a pag. 74). La tabella indica anche se il particolare word processor è venduto su nastri a cassetta o su dischetto

MICROCOMPUTER	TASTIERA	SOFTWARE	SU CASSETTA	SU DISCO
SINCLAIR SPECTRUM		WORDPROCESSOR	*	
COMMODORE VIC 20	*	VICWRITER	*	
DRAGON 32	*	GEMINI	*	
NEW BRAIN		PROPEN	*	
COMMODORE 64	*	VISAWRITE		*
ATARI 800	*	KEYWORD		*
BBC MICRO	*	VIEW		*
EPSON HX20	*	DECKMASTER	*	

Lo sapevate che...?

Notizie e curiosità che difficilmente riuscireste a trovare altrove

Cosa s'intende per computer della "quinta generazione"?

La quinta generazione è il livello al quale si stanno attualmente dedicando i progettisti di hardware e di software. I computer di questa generazione costituiranno un enorme balzo in avanti nel settore.
Il termine "quinta generazione" è stato inizialmente adottato dai Giapponesi per indicare un loro progetto a lungo termine sullo sviluppo dei computer: si tratta di apparecchi che non useranno tastiere per l'immissione di dati e che



non impiegheranno linguaggi tipo il BASIC o il PASCAL: saranno, infatti, in grado di parlare e di comprendere il linguaggio umano e, inoltre, si autoprogrammeranno per la soluzione dei problemi loro presentati.
La prima generazione fu quella dei computer totalmente elettronici, inventati subito dopo la Seconda Guerra Mondiale. La seconda generazione è caratterizzata dall'impiego di transistor al posto delle valvole termoioniche. La terza è quella degli apparecchi che impiegano speciali "circuiti integrati", i progenitori degli attuali "microprocessori", ma che hanno un costo ancora elevato. La quarta generazione, infine, è costituita dalla produzione attuale, che impiega circuiti a Larga Scala d'Integrazione (Large Scale Integration). Lo sviluppo di questi speciali circuiti ha favorito l'enorme diffusione dei piccoli computer.



Dove si trova la Silicon Valley?

Con il nome Silicon Valley si indica una zona degli Stati Uniti, intorno a San Josè (San Francisco), dove sono le sedi della maggior parte delle grandi società americane, che operano nel settore della microelettronica. E pensare che fino a circa 20 anni fa, questa zona era nota soltanto per la sua produzione di frutta!

Nei computer ci sono ancora enormi grovigli di fili elettrici?

La maggior parte dei computer di oggi vengono progettati in modo da nascondere la gran quantità di connessioni elettriche, ed hanno un aspetto esteriore molto "pulito", ma in alcuni computer progettati per la ricerca, le connessioni esterne sono estremamente importanti, e, poiché l'elettricità impiega un certo tempo per scorrere nei fili, la loro lunghezza deve essere accuratamente calcolata.



La pubblicità parla spesso di computer basati su microprocessore Z80 o 6502. Che significato hanno queste sigle?

Le sigle indicano il tipo di microprocessore impiegato: il 6502 e lo Z80 sono due chip di largo impiego nei piccoli computer. I computer, a seconda del tipo di microprocessore, usano un particolare gruppo di istruzioni fondamentali (chiamate Codice Macchina) con le quali vengono scritti i programmi di base. Ma, siccome per la programmazione vengono forniti anche linguaggi cosiddetti "evoluti" (BASIC, PASCAL, ecc.), le differenze tra i vari "linguaggi macchina" non comportano alcuna difficoltà per l'utente. La velocità di elaborazione nei microcomputer dipende talvolta più dal grado di sofisticazione del software, che non dal tipo di chip impiegato.



Il computer può aiutare nella lotta contro il crimine?

Il computer può senz'altro costituire un validissimo aiuto.
La Polizia può infatti accedere, in tempi brevissimi, a tutta una serie di dati concernenti veicoli rubati o sospetti oppure persone che hanno avuto a che fare con la giustizia. Sono pure allo studio particolari progetti per l'impiego di computer in caso di calamità o grossi incidenti. Le applicazioni, anche in questo settore, sono illimitate, ma devono essere sviluppate di pari passo a precise disposizioni legislative. Uno dei più grandi problemi (già sentito in alcune nazioni) è infatti quello di conciliare l'uso legale delle informazioni di questi archivi, dove verranno in futuro conservate informazioni su tutti i cittadini, con il concetto di libertà individuale.

I micro in movimento

Applicazioni di microprocessori nelle auto del futuro



per concessione della TOYOTA

Cruscotto dinamico

Il cruscotto della nostra prossima auto potrebbe assomigliare a quello della foto. Tutti gli indicatori e le lancette sono sostituiti da un visualizzatore controllato dal computer di bordo. Diodi luminosi (LED) indicano la velocità, il livello di carburante, ecc. con una precisione molto superiore agli strumenti tradizionali. La Austin Metro ha già adottato un cruscotto del genere

Uno dei settori nel quale si potrà presto apprezzare direttamente l'impiego dei computer è quello automobilistico. Infatti, è già possibile acquistare auto, dotate di speciali dispositivi elettronici, in grado di comunicare al guidatore quando la macchina necessita di una revisione.
Appositi sensori, collegati alle varie parti del motore, raccolgono una serie di dati (temperatura del motore, numero di giri, chilometraggio percorso, ecc.), che vengono elaborati da un microprocessore per valutare le condizioni generali del veicolo. In base ai risultati, il computer accende o spegne le lampadine "spia" inserite nel cruscotto.
Alcune auto di costruzione europea e giapponese, sono anche in grado di "parlare" al guidatore avvertendolo di allacciare le cinture di sicurezza oppure avvisandolo che la pressione dell'olio non è normale. In questo caso, oltre al microprocessore, è presente un sintetizzatore di voce umana. In base agli impulsi prodotti dal computer, il sintetizzatore riproduce su di un altoparlante gruppi di parole create elettronicamente.

Viaggiare comodamente

Un'altra applicazione del computer può essere quella del controllo delle sospensioni dell'auto. La Lotus sta progettando un sistema chiamato "sospensioni attive". Questa tecnica si avvale del computer per controllare e regolare, anche più volte al secondo, la rigidezza delle sospensioni a seconda del carico. La tenuta di strada risulta perfetta in ogni situazione.
In genere, le auto sportive hanno sospensioni "dure" per poter reggere bene la strada, a scapito della comodità dei passeggeri, sottoposti ad ogni sorta di scossoni. Del resto, sospensioni troppo "morbide" offrirebbero più comodità ma meno sicurezza. Il controllo computerizzato delle sospensioni offre sempre il migliore rapporto tra comodità e tenuta di strada.
La Honda e la Toyota stanno studiando anche dei computer "navigatori", in grado di comunicare al guidatore in quale direzione viaggiare per raggiungere una meta prefissata.
Il computer, rilevando automaticamente la velocità, la direzione e la distanza già percorsa, confronta queste letture con una mappa stradale inserita nella sua memoria.
Apposite "spie" poste sul cruscotto, avvertono il guidatore quando è necessario svoltare a destra o a sinistra.
Alcuni computer, già disponibili, servono per calcolare il consumo di carburante e l'ora approssimativa di arrivo a destinazione. Il guidatore può conoscere, istante per istante, il consumo di benzina e la velocità media "di crociera".
In alcuni casi è anche possibile "prefissare" una velocità di crociera, che il computer manterrà automaticamente agendo sull'acceleratore, salvo l'intervento diretto del guidatore.
Altri tipi di computer vengono installati (in genere su veicoli commerciali da trasporto) per controllare, ad esempio, le ore totali di guida, la velocità e la percorrenza.
Uno dei vantaggi più apprezzabili offerti dai computer per auto è il notevole risparmio di carburante.
La casa automobilistica BMW offre già una gamma di auto dotate di un sistema per determinare la migliore miscela benzina/aria a seconda delle condizioni di guida. Questo aggiustamento automatico, frutto di un esame computerizzato di tutti i fattori determinanti (velocità, marcia innestata, temperatura del motore, ecc.), viene effettuato più volte al secondo, evitando costosi sprechi di carburante.

Il futuro

Cosa possiamo dunque aspettarci in futuro dai computer installati sulle auto? In teoria è possibile affidare all'elaboratore tutte quante le funzioni di guida: il conducente si limiterà ad impostare i dati della destinazione voluta. Il computer provvederà alla guida del veicolo, impiegando speciali sensori collocati sotto il manto stradale o collegandosi direttamente con un computer centrale. Inoltre, per mantenere le distanze di sicurezza e per evitare ostacoli, la velocità dell'auto potrebbe essere controllata automaticamente in base alle rilevazioni di un piccolo radar di bordo.
Entro pochi anni gli attuali strumenti del cruscotto verranno sostituiti con apparecchi di visualizzazione controllati dal computer di bordo. Addirittura, importanti informazioni sulle condizioni del veicolo o sul percorso da seguire potrebbero essere proiettate sul parabrezza, affinché il guidatore non debba distogliere lo sguardo dalla strada. La revisione di un veicolo potrà essere fatta semplicemente collegando il computer dell'officina al veicolo: gli eventuali difetti di funzionamento o le parti in procinto di guastarsi verranno immediatamente individuati.
La foto mostra un prototipo di cruscotto progettato dalla Honda, collegato ad un computer con funzioni di "navigatore"



IAN McKINNELL

Messaggio ricevuto!

Da linguaggio umano a linguaggio elettronico

Un computer è un insieme di parti in plastica, metallo e silicio: se non viene inserito un programma nella sua memoria, non è possibile fargli svolgere alcuna funzione utile. La stesura dei programmi si chiama, appunto, "programmazione". Anche il principiante è in grado di individuare due fasi ben distinte nel processo di stesura di un programma per risolvere un determinato problema. La prima si riferisce a riformulare e convertire il problema in una forma comprensibile al computer; la seconda fase comporta l'immissione delle istruzioni e l'esecuzione del programma. Queste due fasi possono essere ulteriormente suddivise in due stadi; nel primo è il programmatore che agisce, mentre nel secondo è il computer che elabora una serie di informazioni (generalmente senza bisogno di altri interventi dell'utente).

Supponiamo di voler scrivere un programma per la preparazione delle "buste paga". In primo luogo sarà necessario comprendere a fondo e circoscrivere il problema. Quale deve essere il prodotto finale dell'elaborazione? Che genere di informazioni sono necessarie al computer per ottenere i risultati desiderati? Ciò comporta il calcolo di stipendi, di straordinari, ecc. Il passo successivo sarà quello di specificare la corretta sequenza dei calcoli, per esempio: «Come si devono calcolare le ritenute di Legge»?

Nel caso di applicazioni commerciali su larga scala, questo processo iniziale può essere affidato ad un "Analista" professionale, il cui compito è, appunto, quello di analizzare tutti gli aspetti di una procedura da automatizzare e riassumerli in una forma che consenta la vera e propria stesura di un programma. In applicazioni meno impegnative, questo lavoro può essere svolto dallo stesso programmatore.

Se il computer capisse il normale linguaggio umano, queste "specifiche preliminari" potrebbero essere date direttamente "in pasto" all'elaboratore. Ma, almeno per adesso, ciò non è possibile.

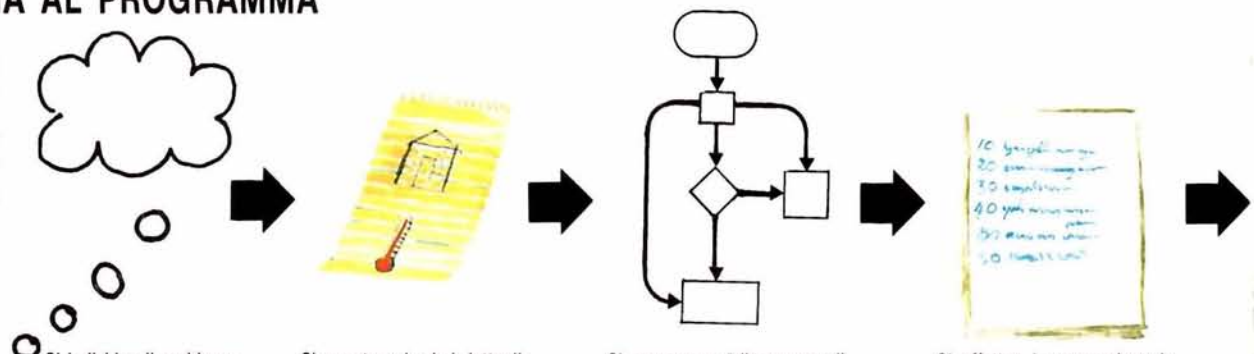
Il principiante, in genere, tende a scrivere un programma così come tradurrebbe un testo in francese o in tedesco: parola per parola dall'inizio alla fine. Un programmatore esperto, al contrario, suddivide il programma in più parti. Ad esempio, il programma per le paghe può essere separato in quattro "moduli" distinti: l'immissione dei dati mensili, la fase di calcolo, l'archiviazione di dati (per esempio: i totali progressivi delle tasse), e la stampa delle "buste paga". Ciascun modulo può essere ulteriormente suddiviso in più gruppi di funzioni. Questo metodo è noto come "programmazione strutturata". Ogni modulo elementare è molto semplice ed è composto da poche righe di istruzioni. Tutte le istruzioni (il "listato" del programma) viene infine immesso nel computer.

Un buon programmatore conserva sempre delle annotazioni su ogni stadio del programma, in modo da avere una precisa traccia di tutto il percorso che va dall'analisi del problema, scritta in linguaggio normale, fino alla sua conversione in un linguaggio di programmazione, ad esempio, il BASIC.

Ciò che accade da quando si fa partire il programma in poi, è totalmente affidato al controllo del computer ed implica, ancora

DAL PROBLEMA AL PROGRAMMA

La stesura di un programma per computer parte dal bisogno di risolvere un determinato problema. Supponiamo di voler mantenere costante la temperatura all'interno di una serra. Per giungere alla stesura del programma, si percorrono vari stadi intermedi



Si individua il problema...

Si annotano i primi dettagli...

Si prepara un "diagramma di flusso" per definire e sviluppare la struttura del programma...

Si effettua la conversione in un linguaggio, per esempio, in BASIC

una volta, una quantità di operazioni a più livelli. La maggior parte di queste operazioni viene eseguita dal computer internamente, senza segnalazioni di alcun genere, salvo quando è richiesto un particolare intervento dell'operatore (come l'immissione di dati necessari ai calcoli) o quando vengono stampati i risultati finali.

In effetti, il computer svolge anche un'altra funzione "trasparente" per l'operatore: la conversione di tutte le istruzioni scritte in linguaggio "ad alto livello" (in BASIC, per intenderci) in istruzioni del "linguaggio macchina". Negli home computer questo processo viene fatto da un interprete, conservato nelle memorie ROM.

L'interprete è un sofisticato programma in linguaggio macchina, eseguito automaticamente ogni volta che impartiamo il comando RUN: il listato del programma immesso dall'operatore viene esaminato carattere per carattere. Ogni istruzione viene confrontata con quelle contenute in un "vocabolario" interno. Se un'istruzione non viene riconosciuta, magari per via di un errore di battitura, il processo di conversione si interrompe, e viene dato avviso dell'errore (ad esempio: SYNTAX ERROR).

Se, invece, l'istruzione è presente nel vocabolario, il controllo viene passato alla sezione dell'interprete che esegue la funzione richiesta. Se l'istruzione, per esempio, è una PRINT, l'interprete si prepara alla visualizzazione della sequenza di dati che seguono la PRINT. Ecco, dunque, un altro livello di operazione: la serie di caratteri da visualizzare viene copiata in una speciale zona di memoria riservata al video, e quindi convertita in opportuni segnali elettrici da inviare al monitor o al televisore. Queste funzioni vengono svolte con incredibile rapidità, tanto che spesso sembrano contemporanee alle operazioni di calcolo.

L'introduzione dei dati alla tastiera provoca un processo del tutto simile: la tastiera viene esaminata per individuare se e quali tasti sono stati premuti, ed il corrispondente codice viene conservato in una specifica zona di memoria per essere poi utilizzato dal programma. Inoltre, poiché deve essere possibile interrompere in qualsiasi momento l'esecuzione del programma, la tastiera viene continuamente "interrogata" dal computer, anche durante le fasi di calcolo.



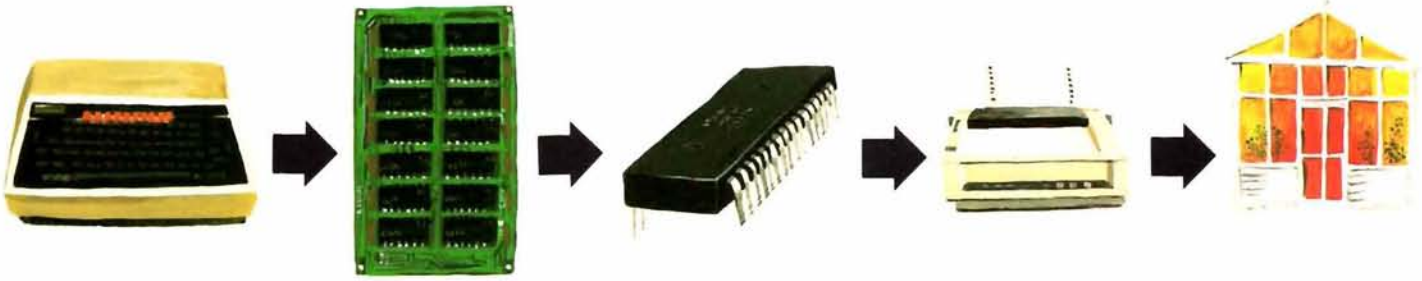
In realtà, i microprocessori dei piccoli computer possono svolgere una sola funzione per volta, cosicché le funzioni interne (scansione della tastiera, controllo della visualizzazione, ecc.) o vengono alternate a quelle relative all'interprete, o vengono gestite da sistemi "ad interruzione", dove viene generato un impulso periodico (ad esempio 50 volte al secondo) per "ricordare" al microprocessore di controllare la tastiera, il video ed altre periferiche, prima di continuare l'esecuzione del programma principale.

Vi sono, dunque, più livelli operativi nel computer, anche quando il programma scritto dall'operatore è tra i più semplici.

La tendenza attuale, nel progetto dei computer, è quella di renderne l'uso il più facile possibile ed in un futuro non molto lontano sarà addirittura possibile scrivere programmi usando il normale linguaggio parlato.

Il software nascosto

In ogni computer vi è una certa quantità di software più o meno accessibile che sovrintende continuamente ad una serie di operazioni: la tastiera viene costantemente esaminata per individuare quando viene premuto un tasto, il video è incessantemente scandito per la visualizzazione dei caratteri, come pure le uscite delle varie periferiche. Tutto ciò avviene in maniera del tutto "trasparente" per l'operatore, anche nei computer meno costosi



Il programma viene immesso nella memoria del computer usando la tastiera

L'interprete BASIC provvede a convertire le istruzioni in linguaggio macchina ad uso della CPU...

Tutte le operazioni vengono eseguite dalla CPU. I risultati vengono inviati alle periferiche (stampante, video, unità a dischi, ecc.)

Se necessario, viene prodotto un listato del programma su stampante

Se il programma è corretto in ogni sua parte, il problema è risolto

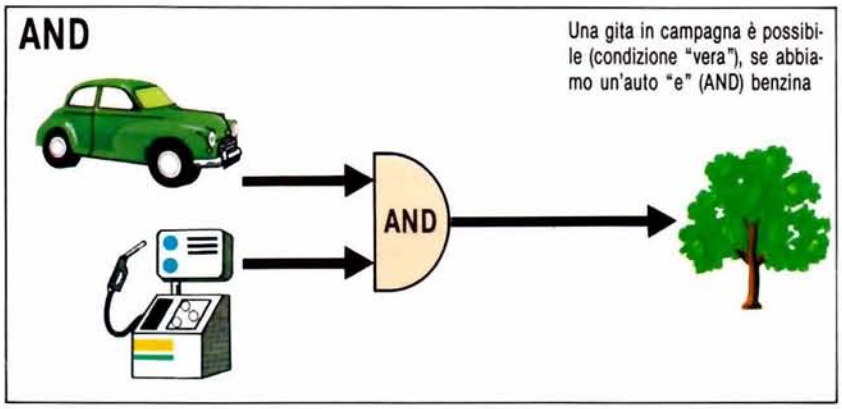
Vero o falso?

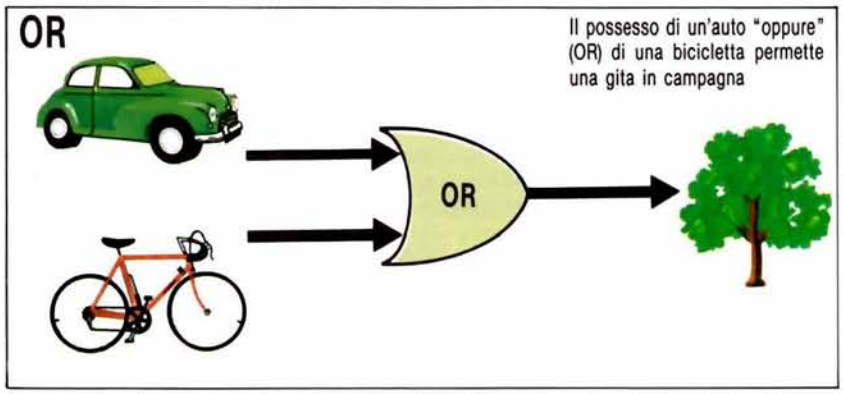
I computer forse non pensano, ma possiedono una ferrea logica

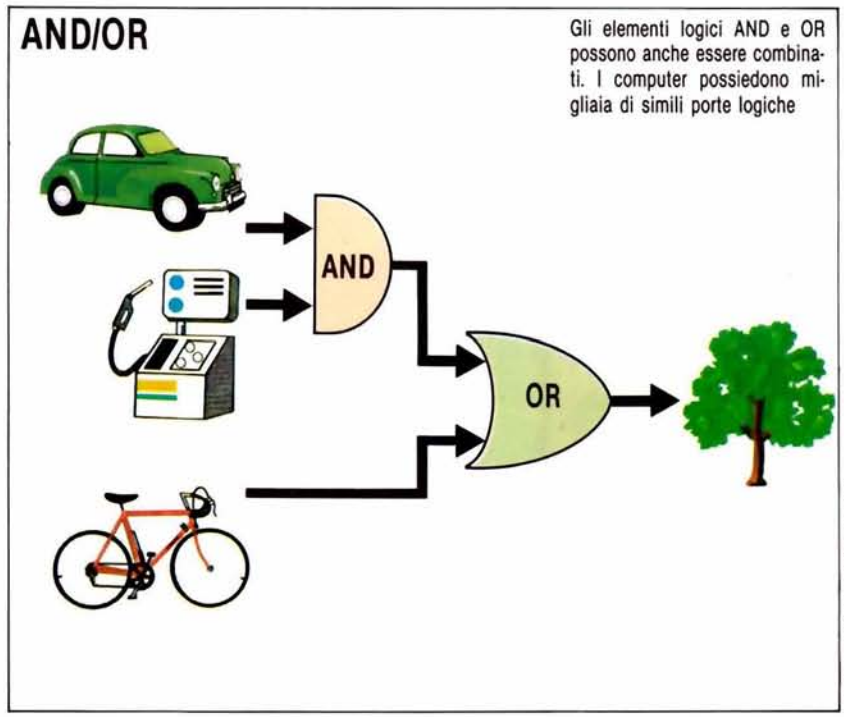
La CPU è sovente indicata come il cervello del computer: in questo chip vengono eseguiti tutti i calcoli e i confronti "logici". Come avvengono, effettivamente, questi processi? Per comprenderlo appieno, è necessario conoscere i fondamenti dell'aritmetica binaria ed il funzionamento delle "porte logiche" (logic gates).
Nei computer, le porte logiche sono semplici circuiti elettrici in grado di fare scelte e confronti. Il principio di funzionamento non è molto complesso e può essere facilmente spiegato con qualche esempio.
Vi sono tre tipi fondamentali di porte: la porta AND, la porta OR e la porta NOT. I termini AND, OR e NOT, di uso ormai universale, in italiano significano, rispettivamente: "e", "oppure", "no".

Un contatto logico

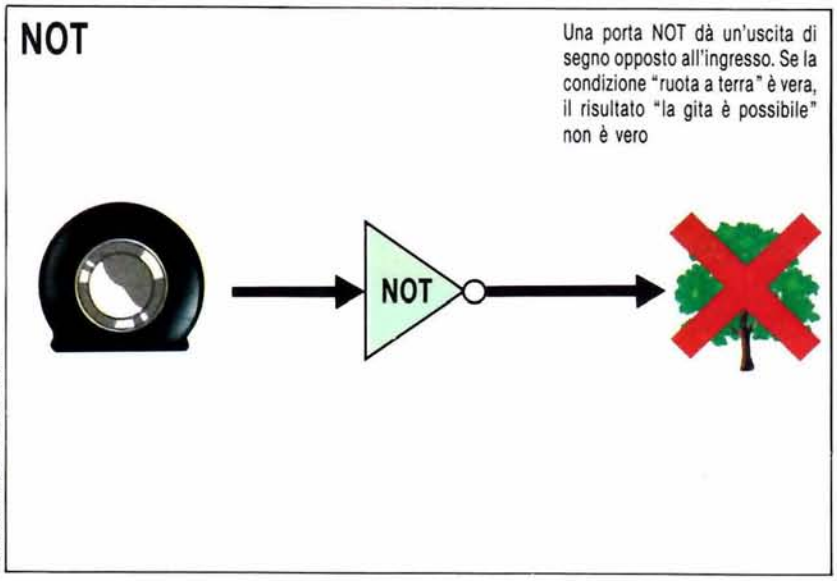
Una porta può essere considerata come un circuito con due ingressi ed un'uscita. Il valore in uscita di una porta AND è "vero" soltanto se ambedue gli ingressi sono "veri". Supponiamo, ad esempio, di voler fare una gita in campagna. Se possediamo un'auto e (AND) abbiamo benzina sufficiente, la gita è possibile. In mancanza di una delle due condizioni, la gita non è possibile. Ovvero, le due condizioni "in ingresso" devono essere ambedue "vere".
Più avanti vedremo come si possono tradurre questi concetti in equazioni ed in "tabelle della verità".
Immaginiamo adesso una situazione leggermente diversa: se possediamo un'auto "oppure" (OR) una bicicletta, la gita è possibile. Soltanto mancando ambedue i mezzi la gita è impossibile. Nel gergo dei computer, l'uscita è "falsa" solo se ambedue gli ingressi sono "falsi".
L'ultimo tipo di porta logica (NOT) è forse il più facile a comprendere: qualunque sia il valore in ingresso (condizione "vera" o "falsa") l'uscita assume il valore opposto. Se l'ingresso è "vero" l'uscita è "falsa" e viceversa. Ritornando al nostro esempio, la condizione "gomma a terra" deve essere "falsa" per poter fare la gita.
Questi elementi logici possono anche essere combinati, come mostrano le illustrazioni, che meglio servono a spiegare i concetti.
Combinazioni di AND, OR e NOT permettono di compiere scelte basate, solitamente, sulla logica convenzionale.







ANDY LESLIE/MARK WATKINSON



ANDY LESLIE/MARK WATKINSON

Può essere interessante esercitarsi a costruire combinazioni di porte per altre situazioni ipotizzabili.

Le Tavole della verità

I simboli impiegati nelle illustrazioni sono gli stessi che vengono usati nei diagrammi dei circuiti per computer. Vediamo com'è possibile riassumere le varie condizioni per una porta logica AND.
Usiamo la lettera "a" per indicare che possediamo un'auto, la lettera "b" per indicare che abbiamo benzina sufficiente e la lettera "g" per indicare che la gita è possibile. Possiamo poi usare una "V" per "vero" ed una "F" per "falso". Nella tavola della verità sono riportate tutte le possibili combinazioni d'ingresso e le conseguenti situazioni di uscita con una porta AND:

AUTO	F	V	F	V	(a)
BENZINA	F	F	V	V	(b)
GITA	F	F	F	V	(g)

La tavola della verità per 2-input AND

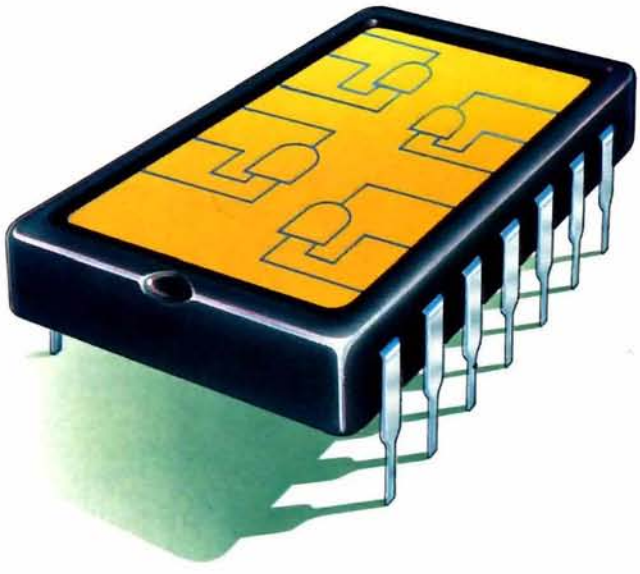
AUTO	0	1	0	1	(a)
BENZINA	0	0	1	1	(b)
GITA	0	0	0	1	(g)

La stessa tavola della verità usata con 0 e 1 per Falso e Vero

Nei computer vengono usati i numeri binari uno e zero per indicare le condizioni, rispettivamente, di "vero" e di "falso". Poiché il computer può interpretare una tensione positiva come 1 ed una tensione nulla come 0, una porta AND può essere fatta impiegando dei transistor collegati in maniera tale da fornire una tensione positiva in uscita, soltanto se sono presenti tensioni positive ai due ingressi. Se a uno o a tutti e due gli ingressi è presente una tensione nulla, anche l'uscita è nulla.
Un circuito elettronico OR ha una tensione positiva in uscita, se è positiva la tensione applicata ad uno o a tutti e due gli ingressi. Solo se ambedue gli ingressi sono a zero, anche l'uscita è a zero.
In un circuito NOT, se la tensione all'ingresso è positiva allora quella in uscita è zero e viceversa.

Il chip 7408

I chip di silicio spesso contengono migliaia di porte per operazioni logiche AND, OR e NOT. Il piccolo chip 7408 qui illustrato contiene tutti i circuiti necessari per quattro porte AND. Sull'involucro, sono riportati i simboli logici delle porte, invece dell'effettivo circuito interno



TONY LODGE

Il Lynx

Anatomia di un microcomputer di diffusione europea: tecnologie a confronto

Il Lynx è un computer prodotto in Gran Bretagna dalla Computers di Cambridge. È un apparecchio le cui dimensioni e peso sono una via di mezzo rispetto ai suoi principali rivali di stessa nazionalità (lo Spectrum Sinclair e l'Oric-1) e gli americani (Atari e Commodore). L'involucro del Lynx è forse uno tra i più eleganti del settore, rifinito in una sfumatura grigia, esteticamente pregevole.
La sua memoria di 48 Kbyte può essere estesa ad un massimo, molto apprezzabile, di 192 Kbyte. Considerando che gli apparecchi professionali offrono in genere un massimo di 64 Kbyte, si può apprezzare la versatilità del Lynx. Questo computer ha una tastiera tipo macchina da scrivere, con un "tocco" molto piacevole e sicuro (a differenza di tante altre tastiere, ci si accorge se si è premuto un tasto o no). Con il Lynx si possono visualizzare fino ad 8 colori diversi e sono a disposizione 32 caratteri grafici (anche se non indicati sulla tastiera). Il BASIC che viene fornito assieme al computer è di buon livello, ed include utili istruzioni, tra le quali l'AUTO per la numerazione automatica delle linee di programma. L'unità centrale del Lynx è il microprocessore Z80, lo stesso impiegato in moltissimi computer molto sofisticati.



La tastiera del Lynx

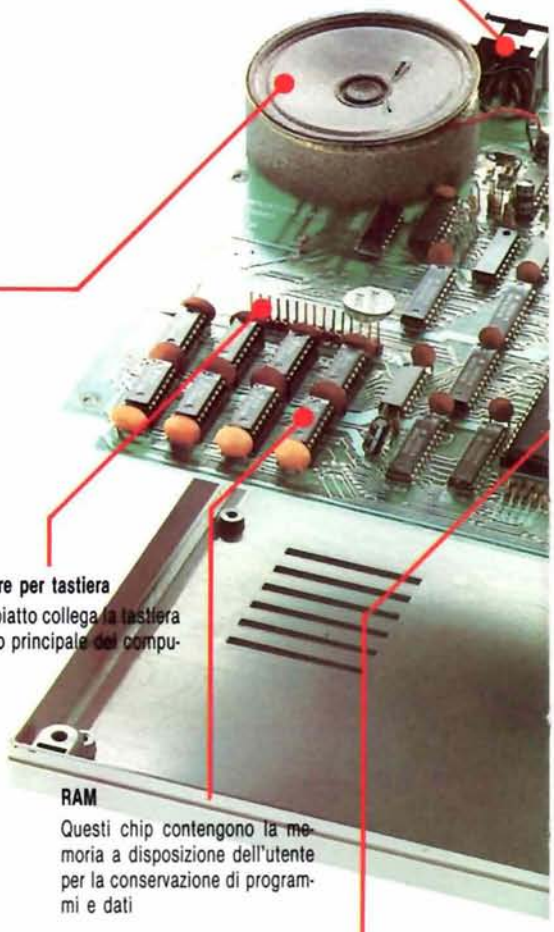
Il Lynx ha un aspetto elegante e solido nel suo contenitore di plastica grigia. Ha una tastiera professionale con 57 tasti, più la barra spaziatrice. La disposizione segue lo standard QWERTY. Le due coppie di tasti, poste alle due estremità della terza fila, servono per spostare il cursore sullo schermo quando si usa lo "screen editor". I tasti ESCAPE e CONTROL sono in alto a sinistra, mentre quelli BREAK e DELETE in alto a destra

Interfaccia parallela

Questo connettore permette di allacciarsi ad apparecchi che usano una trasmissione parallela, ad esempio le unità a disco

Presa di alimentazione

La presa, di tipo DIN, serve per collegare il Lynx all'alimentatore in corrente continua



Altoparlante

Tutti gli effetti audio generati dal Lynx vengono riprodotti da questo altoparlante

Connettore per tastiera

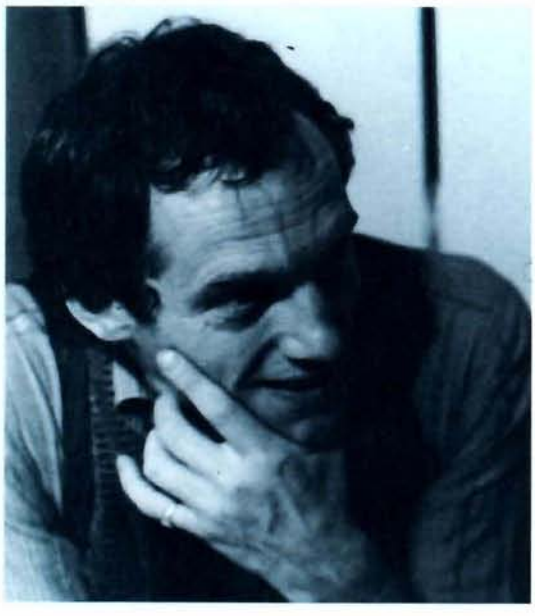
Un cavo piatto collega la tastiera al circuito principale del computer

RAM

Questi chip contengono la memoria a disposizione dell'utente per la conservazione di programmi e dati

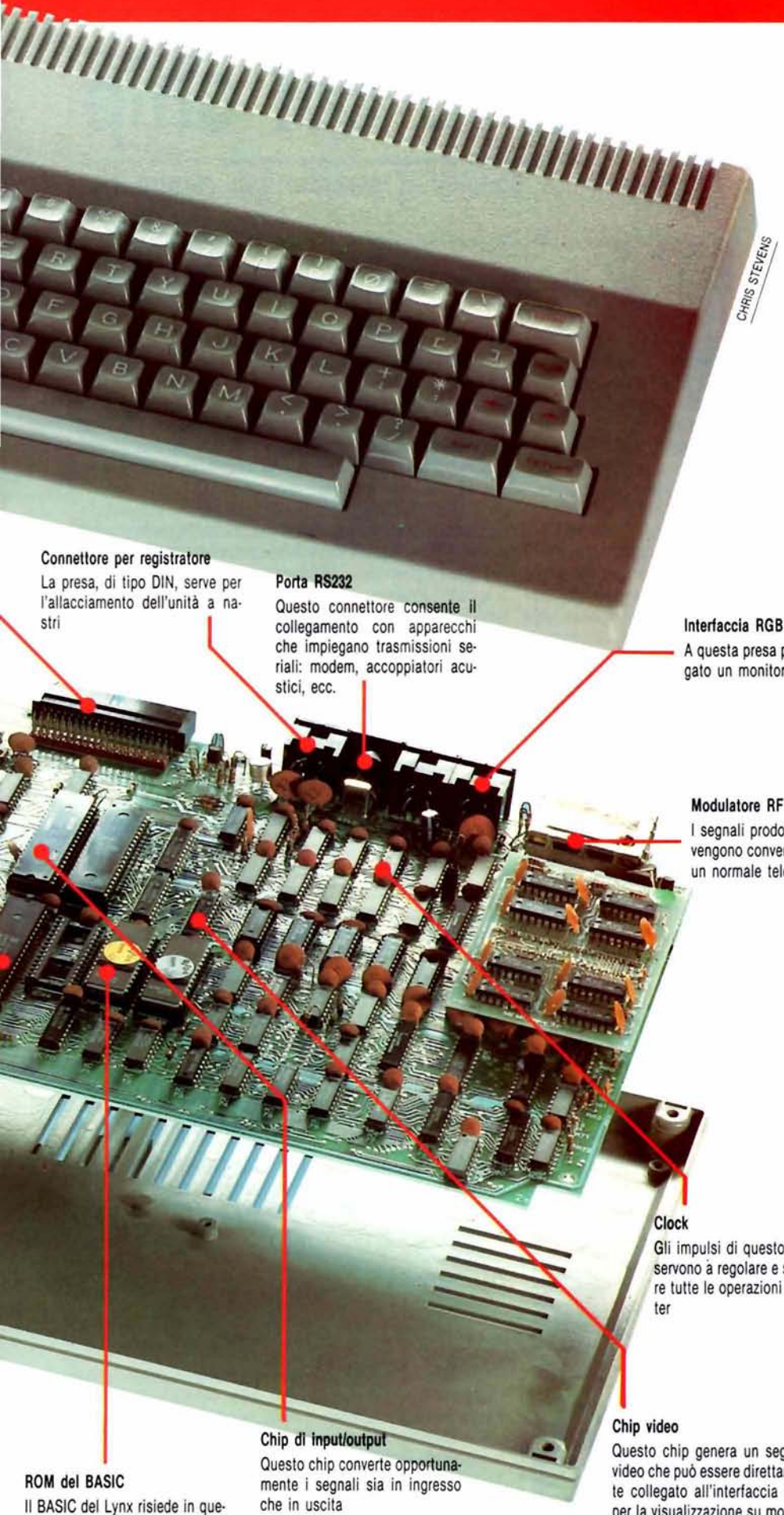
Microprocessore

L'unità centrale del Lynx è lo Zilog Z80A



John Shirreff

Il cervello, che ha dato vita alla gamma dei computer Lynx, è quello del trentacinquenne John Shirreff. Poco dopo aver lasciato l'Università di Cambridge, negli anni '60, egli si dedicò al progetto ed alla produzione di strutture gonfiabili mobili. Durante quel periodo, inoltre, suonò la batteria in diversi gruppi rock. Intrapresa la carriera di addetto alla registrazione musicale, ha coltivato a lungo la passione per l'elettronica. L'interesse per la musica e quello per l'elettronica si fusero quando egli incontrò Dave Vorhaus, titolare di uno studio di registrazione. Nel 1981 progettò un innovativo computer commerciale, basato sul chip Z80, con 64 Kbyte di memoria. L'esperienza di quel gruppo di progettisti è servita in gran misura per la produzione del Lynx



Connettore per registratore
La presa, di tipo DIN, serve per l'allacciamento dell'unità a nastri

Porta RS232
Questo connettore consente il collegamento con apparecchi che impiegano trasmissioni seriali: modem, accoppiatori acustici, ecc.

Interfaccia RGB
A questa presa può essere collegato un monitor a colori

Modulatore RF
I segnali prodotti dal chip video vengono convertiti in segnali per un normale televisore

Clock
Gli impulsi di questo oscillatore servono a regolare e sincronizzare tutte le operazioni del computer

Chip video
Questo chip genera un segnale video che può essere direttamente collegato all'interfaccia RGB per la visualizzazione su monitor oppure al modulatore RF se viene impiegato un normale televisore

Chip di input/output
Questo chip converte opportunamente i segnali sia in ingresso che in uscita

ROM del BASIC
Il BASIC del Lynx risiede in queste due memorie ROM

LYNX

DIMENSIONI

mm. 350 x 213 x 60

PESO

1564 grammi

CPU

Z80A

CLOCK

4 MHz

MEMORIA

48 Kbyte di RAM estendibile a 192 Kbyte. 16 Kbyte di ROM contengono il BASIC ed i programmi di gestione interna

VIDEO

24 linee di 40 caratteri, oppure 248 x 256 punti per la grafica. 8 colori disponibili

INTERFACCE

Connettore per TV, per monitor RGB, per video composito, connettore per registratore a cassette, connettore RS232, connettore per espansioni parallele

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

— — —

ACCESSORI COMPRESI

Alimentatore, cavi di connessione per TV e per registratore a cassette, un nastro con programmi dimostrativi, un manuale

TASTIERA

Professionale, tipo QWERTY, 57 tasti più barra spaziatrice

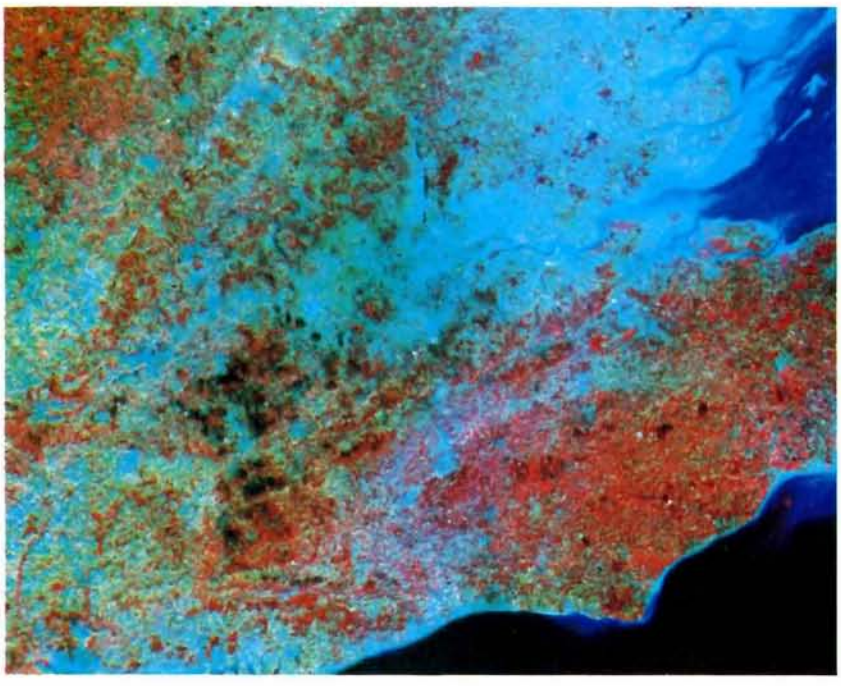
DOCUMENTAZIONE

Sebbene il manuale offra una sufficiente introduzione al BASIC usato, mancano quegli approfondimenti che i più esperti si aspettano di trovare. Le spiegazioni sul funzionamento degli effetti grafici e sonori del Lynx sono molto adatte ai principianti, mentre quelle concernenti il linguaggio macchina lo sono meno. La mancanza di un indice rende difficile il reperimento di particolari argomenti. Le illustrazioni non sono sempre chiare, specialmente quelle più tecniche

Appuntamento col chip

Grazie agli "expert system", gli specialisti avranno più tempo per la ricerca

L'intelligenza artificiale e la creazione di computer pensanti in grado di prendere decisioni come gli esseri umani fanno ancora parte della fantascienza. Conoscere totalmente e perfettamente la mente umana e il suo funzionamento è impresa ciclopica e, sebbene vi siano stati alcuni progressi, un computer del tipo *2001 Odissea nello spazio* è ancora lontano.
Ma se limitiamo il compito del computer ad una sfera più ristretta, nella quale esso simuli in parte l'intelligenza in un settore circoscritto delle attività umane, allora si ha l'impressione che il computer sia intelligente. Questa è la teoria alla base degli "expert system", ossia dei sistemi "specializzati". Il loro principio operativo sta nella possibilità che un esperto di un certo settore, ad esempio il migliore chirurgo od il miglior geologo, possa trasferire nel computer la sua conoscenza ed esperienza del suo particolare settore di lavoro. A questo punto, grazie a speciali programmi che gestiscono le informazioni accumulate, anche persone totalmente digiune della materia possono interrogare il computer ed ottenere risposte sensate. Gli "expert system" potrebbero risultare utili in più occasioni. È già stato sviluppato un programma che, in base ai sintomi presentati dai pazienti, è in grado di diagnosticare la causa di particolari dolori allo stomaco. Un altro programma, sfruttando le attuali conoscenze nel campo geologico, viene usato per individuare probabili giacimenti di molibdeno e di altri minerali. Un terzo programma, partendo da enormi quantità di dati sperimentali non organizzati, costruisce strutture verosimili di molecole organiche. Questi processi, normalmente, richiederebbero il costoso intervento di scienziati molto specializzati, i quali, grazie al computer, possono invece dedicarsi ad attività più produttive.
Ma gli "expert system" possono fare molto più che sostituire gli esperti umani. Più volte l'elaborazione delle informazioni fornite dagli esperti ha portato a risultati inattesi. In particolare, vengono spesso scoperte, tra certe informazioni, correlazioni precedentemente sfuggite all'uomo, si aprono così nuovi orizzonti alla ricerca.
Gli "expert system" sono, o perlomeno diverranno presto, un importante sviluppo delle applicazioni del computer. Con programmi di buona qualità, il computer può essere un validissimo aiuto in campi specifici. Inoltre, poiché il programma può essere usato su più computer, si può moltiplicare all'infinito l'accesso alle informazioni che, fino a poco tempo fa, appartenevano ad un solo individuo.
Ovviamente, il problema più difficile da risolvere consiste nel creare programmi dal funzionamento sicuro, capaci di sostituire la genialità dell'esperto umano.



per concessione della STANFORDS LTD

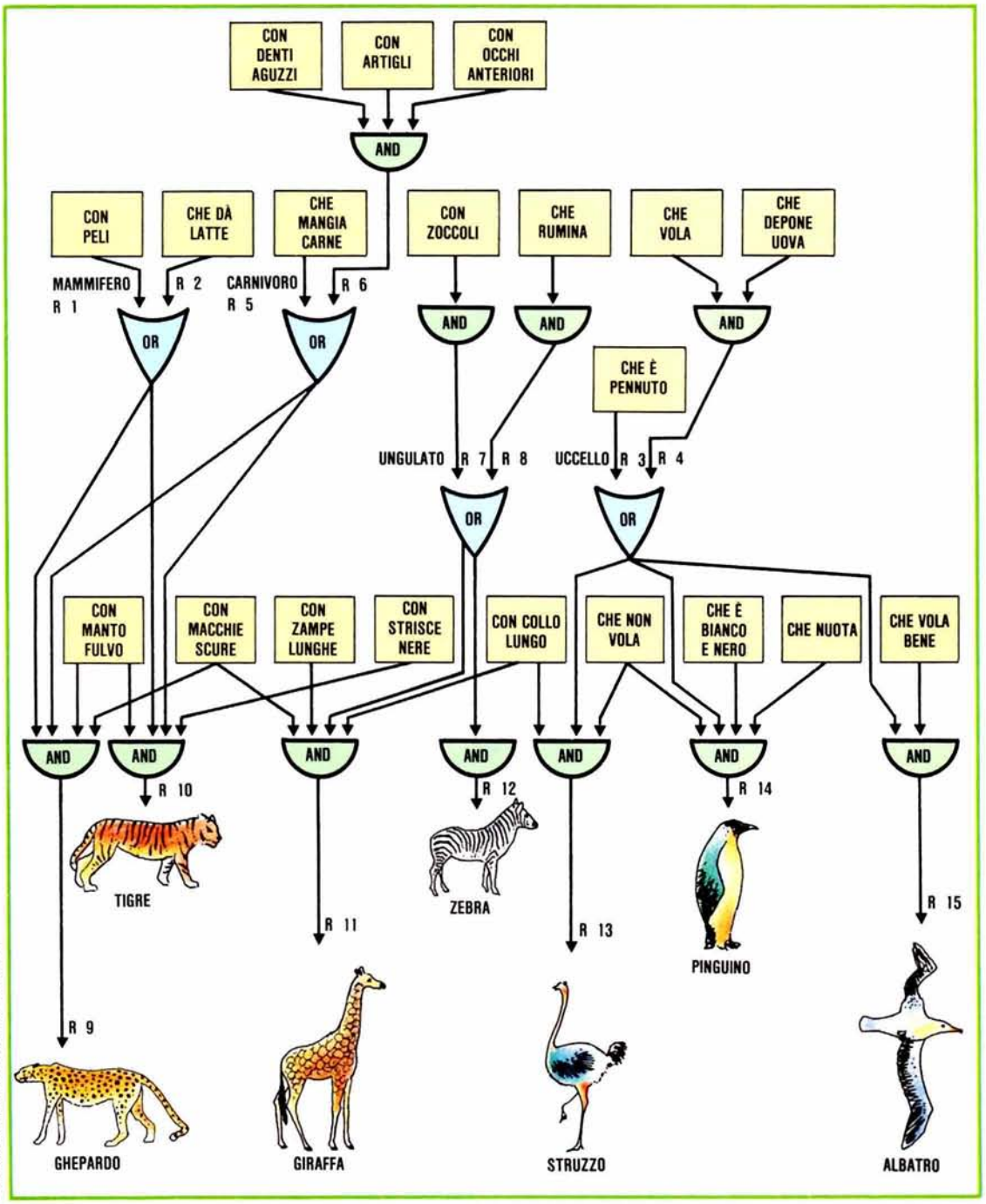
La creazione del programma

Il primo passo consiste nel tentare di comprendere come l'esperto di un certo settore ragiona nel suo lavoro. I processi della mente umana non sono sempre logici, non almeno quanto quelli di un computer, e dipendono in larga parte dall'esperienza. Se un esperto umano deve risolvere un problema del tutto

Veduta dallo spazio

Il Landsat 4 è stato lanciato nel luglio dell'82 e gira in un'orbita che, sorvolando l'intera superficie terrestre, ripassa sullo stesso punto ogni 20 giorni. Tutti i dati raccolti vengono trasmessi sotto forma di codici digitali. Grazie a speciali tecniche computerizzate, è possibile identificare oggetti di soli 40 metri di diametro. La foto mostra un rilievo, opportunamente rielaborato fotograficamente, fatto dal satellite mentre sorvolava la parte meridionale dell'Inghilterra

nuovo, egli lo paragona mentalmente a tutta una serie di problemi già risolti in passato e, grazie a particolari associazioni d'idee, ne viene tentata la soluzione.
Ma per utilizzare tutto il bagaglio conoscitivo di un medico, per esempio, dovremmo creare una struttura immensa, capace di immagazzinare e di utilizzare una quantità immensa di piccole e grandi informazioni, collegate tra loro in più di un modo. Dovremmo anche introdurre altri elementi per una migliore simulazione del comportamento umano.
I medici, basandosi sui sintomi dei pazienti, spesso esprimono delle diagnosi di cui sono sicuri soltanto in parte.
Pertanto, in un modello computerizzato andrebbe introdotto un calcolo probabilistico legato alla diagnosi.
La maggior parte degli "expert system" intavolano una specie di dialogo con l'utente, che deve descrivere i dettagli del problema da risolvere. Il metodo più ovvio e più semplice è quello in cui è lo stesso computer a rivolgere specifiche domande, offrendo al tempo stesso una gamma di possibili risposte numerate. L'utente digita il numero corrispondente alla risposta ed il computer, in base a questa, passa alla domanda successiva. Si potrebbe definire un procedimento per eliminazione, col quale si giunge, infine, alla soluzione.
Presto saranno disponibili degli "expert system" da usare su di un home computer installato nel nostro salotto.



JULIE-ANN CHAMBERS/MARK WATKINSON

UN NETWORK DI ANIMALI

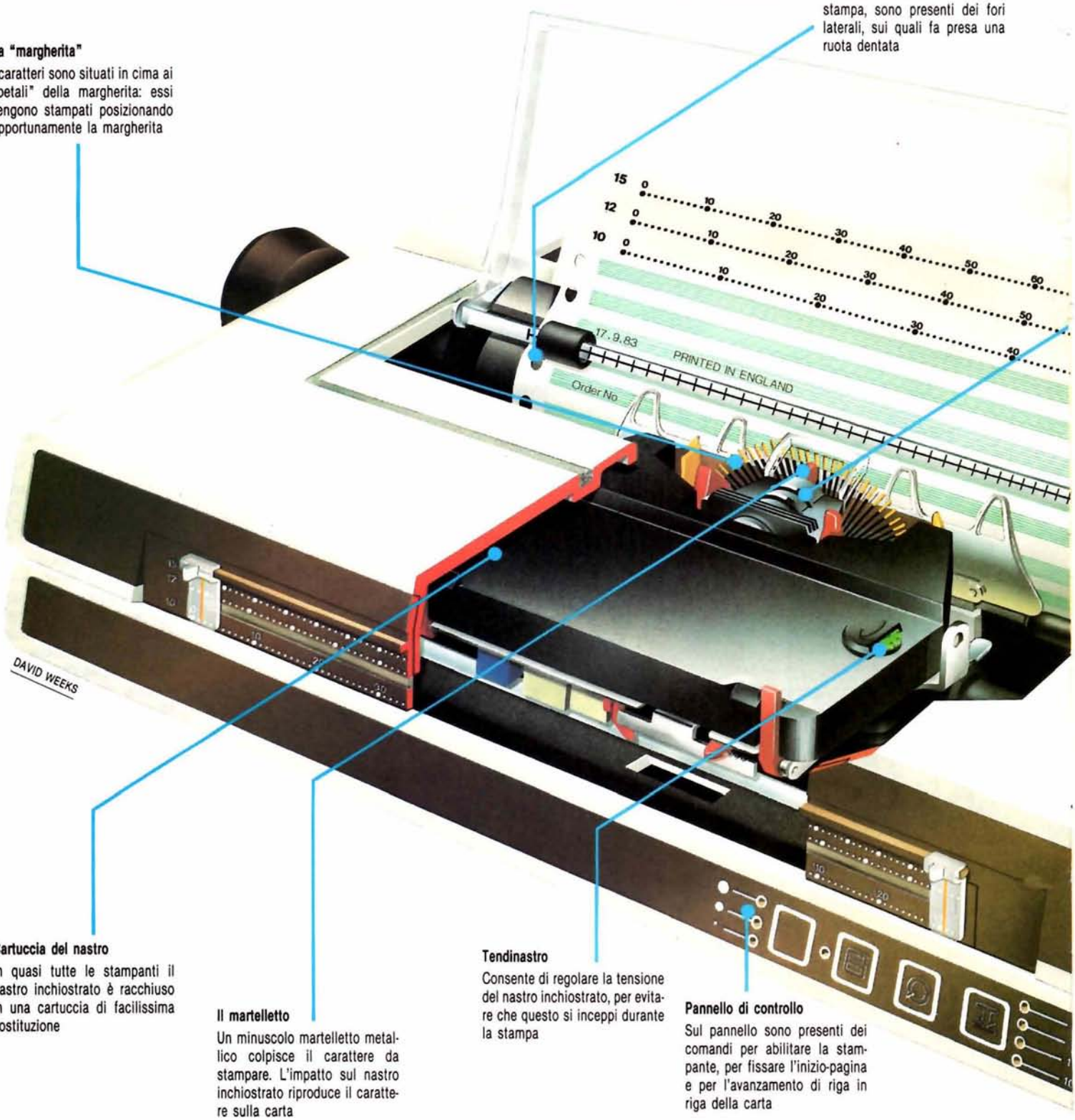
La figura a lato mostra come la "conoscenza" di un "expert system" può essere organizzata con l'uso di poche e semplici regole, che qui vengono numerate da R1 a R15. Scopo delle regole è quello di individuare il nome di un particolare animale, in base alle informazioni ad esso concernenti. Supponiamo di immettere la seguente informazione: "l'animale ha denti aguzzi". Il computer, iniziando dalla casella in alto a sinistra, deve scegliere un percorso che individui il nome dell'animale in questione. Il termine AND (dentro il primo semicerchio) indica che le tre condizioni (con denti aguzzi, con artigli e con occhi anteriori) devono essere tutte "vere", affinché possa valere la regola R6. Pertanto, il computer chiederà se l'animale ha o meno "artigli" e "occhi anteriori". Se ambedue le risposte sono affermative, il computer passa alla casella successiva, che contiene un OR. In questo caso, il computer può avanzare purché l'una oppure (OR) l'altra delle due condizioni sia "vera", il che si verifica nel nostro caso, essendo vera la R6. Il percorso porta a due condizioni AND. Per giungere al risultato finale, il computer deve chiederci se l'animale ha il manto a strisce oppure maculato. Dalla nostra risposta il computer può dirci se l'animale è una tigre o un ghepardo

Via con la stampa

Come orientarsi nella scelta di una stampante

La "margherita"
I caratteri sono situati in cima ai "petali" della margherita: essi vengono stampati posizionando opportunamente la margherita

Fori di guida
Per assicurare un corretto avanzamento della carta in fase di stampa, sono presenti dei fori laterali, sui quali fa presa una ruota dentata

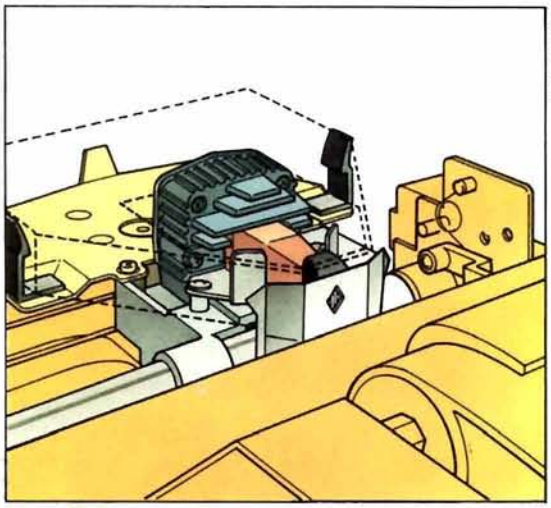


Cartuccia del nastro
In quasi tutte le stampanti il nastro inchiostrato è racchiuso in una cartuccia di facilissima sostituzione

Il martelletto
Un minuscolo martelletto metallico colpisce il carattere da stampare. L'impatto sul nastro inchiostrato riproduce il carattere sulla carta

Tendinastro
Consente di regolare la tensione del nastro inchiostrato, per evitare che questo si inceppi durante la stampa

Pannello di controllo
Sul pannello sono presenti dei comandi per abilitare la stampante, per fissare l'inizio-pagina e per l'avanzamento di riga in riga della carta

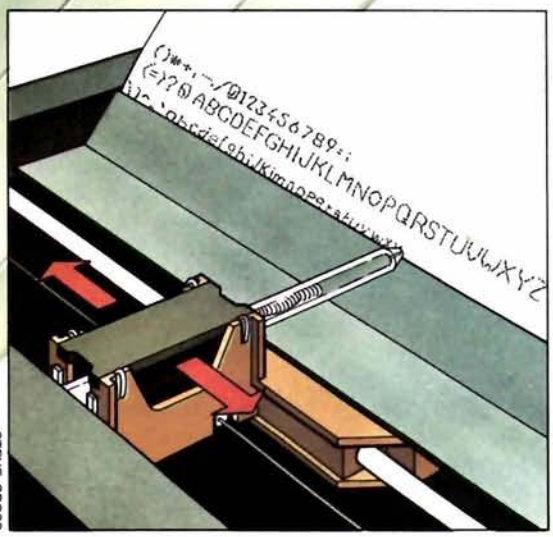


STEVE CROSS

Motore
Il motore ruota la margherita per ottenere il corretto posizionamento del carattere da stampare

La stampante "a matrice di punti"
Questo tipo di stampante impiega una fila (matrice) di aghi per la stampa dei caratteri. La combinazione di aghi che colpiscono il nastro inchiostrato, durante lo spostamento della testina di stampa, crea la forma del particolare carattere da stampare

Stampanti "a getto"
In questo caso il carattere viene generato da un minuscolo getto d'inchiostro, opportunamente deflesso da un campo elettrostatico. La qualità di stampa è tra le migliori



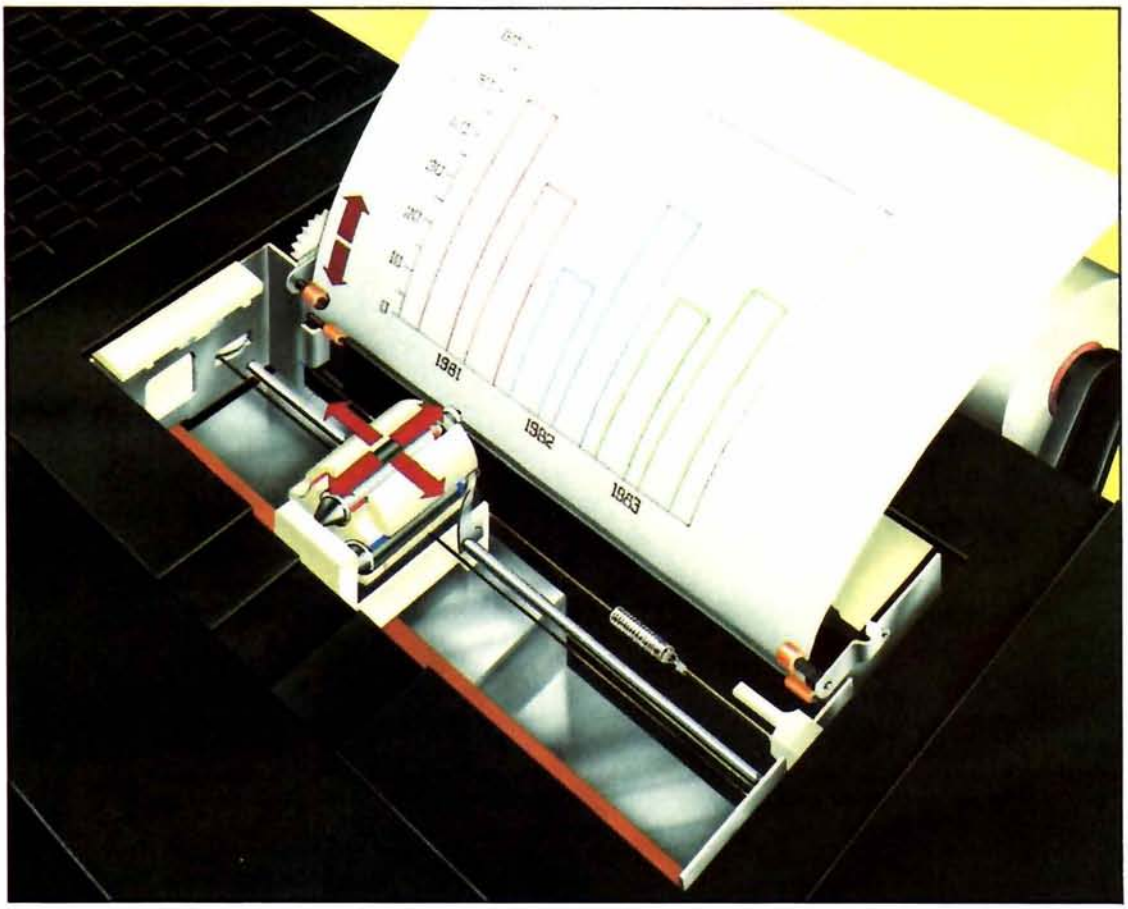
STEVE CROSS

L'acquisto di una stampante non sempre avviene contemporaneamente a quello del computer. Ciò è particolarmente vero nel caso degli home computer, che vengono comprati soprattutto a scopo didattico o ricreativo.
Viene però il momento in cui si sente la necessità di avere "su carta" il risultato di un certo calcolo oppure il listato di un particolare programma.
Ma, tra i tanti tipi di stampanti oggi disponibili, la scelta non è sempre così facile e dipende da molti fattori.

La scelta della stampante

Esistono quattro tipi principali di stampanti: quelle "a matrice di punti", quelle "a margherita", quelle "a getto" e quelle termiche.
Il tipo più comune è quello "a matrice di punti". Il principio di funzionamento si avvale di una testina mobile dotata di un certo numero di finissimi aghi. La stampa di un carattere avviene allorché il nastro inchiostrato viene colpito da una particolare combinazione degli aghi, mentre la testina scorre da un lato all'altro del foglio. Le stampanti "a matrice di punti" hanno due pregi: sono veloci e costano relativamente poco. Il rovescio della medaglia consiste nella bassa qualità di stampa, essendo questa ottenuta da serie di punti, e nel funzionamento spesso piuttosto rumoroso. Per migliorare la qualità di stampa, alcuni di questi apparecchi "ripassano" il carattere più volte, riuscendo così a nascondere quasi del tutto gli spazi tra punto e punto.
Le stampanti a matrice di punti possono andar bene per produrre listati di programmi e per la stampa di tabulati e grafici. In alcuni casi, oltre ai normali caratteri, è anche possibile stampare speciali caratteri grafici. Se però è richiesta una buona qualità di stampa, è necessario orientarsi verso altri tipi di stampante.
Per avere una stampa nitida ed omogenea, simile a quella delle macchine da scrivere elettriche, occorre scegliere una stampante "a margherita", chiamata così per via di una specie di ruota a lunghi "petali" usata per la stampa dei caratteri. In cima a ciascun "petalo" c'è un carattere e la stampa avviene facendo ruotare la "margherita" fino a che il carattere da stampare non si trova all'altezza di un martelletto elettrico. Un breve impulso provoca l'avanzamento del "petalo" ed il carattere colpisce il nastro inchiostrato. Le "margherite" sono generalmente in plastica o

Stampante "a penna"
Questa rappresenta una nuova concezione nella tecnica di stampa. La testina di stampa regge quattro speciali penne "a sfera" colorate. La carta può scorrere avanti e indietro sul rullo, consentendo di tracciare linee verticali, mentre lo spostamento della testina lungo l'asse permette il tracciamento delle linee orizzontali. Questo tipo di stampante si presta molto bene alla realizzazione di grafici ed immagini complesse. La qualità di stampa è piuttosto buona, ma, poiché alcuni colori vengono usati più di altri, occorre controllare periodicamente l'efficienza delle penne



STEVE CROSS

in metallo e, con una semplice operazione, è possibile sostituire "margherite" con diversi tipi di caratteri.
Gli svantaggi delle stampanti "a margherita" sono la lentezza ed il costo relativamente alto. Inoltre, non sono molto pratiche per produrre dei buoni grafici, in quanto occorre cambiare più "margherite" per combinare i caratteri grafici e quelli normali.
Più costose ancora sono le stampanti "a getto". In queste, un microscopico getto d'inchiostro viene elettrostaticamente deviato per generare i caratteri di stampa.
L'enorme pregio di queste stampanti è, oltre ad un'ottima qualità di stampa, l'impressionante velocità: in un solo secondo si possono stampare circa 20 metri di caratteri!
Le stampanti termiche, infine, funzionano grazie all'impiego di carta speciale, sensibile al calore. La testina di stampa trasferisce calore accostandosi alla carta, che si annerisce rapidamente secondo i contorni del carattere da stampare. Le stampanti termiche sono molto silenziose e discretamente veloci. Una delle più note è la "Silent Type" della Apple. Il prezzo delle stampanti termiche è molto ragionevole, ma è necessario usare la speciale carta termica che ha un costo ovviamente superiore alla carta normale.

L'interfaccia giusta

Quando si decide per l'acquisto di una stampante, oltre a considerare la qualità e la velocità di stampa, è necessario accertarsi che essa si possa collegare al computer in nostro possesso. In primo luogo, il connettore della stampante deve corrispondere a quello del computer. Le interfacce generalmente disponibili sui computer sono: la Centronics, l'IEEE488 e la RS232. La Centronics è anche nota come interfaccia "parallela". Sul manuale del computer (o addirittura sullo stesso connettore) è indicato il tipo di interfaccia disponibile.
Comunque, l'industria di questo settore è tristemente nota per le numerose incompatibilità e potremmo scoprire che, anche se i due connettori corrispondono perfettamente, in effetti i due apparecchi non sono compatibili. Ciò può derivare, ad esempio, dalla diversa velocità di trasferimento dei dati (indicata col termine "baud"). L'invio dei dati (bit) dal computer alla stampante può avvenire in due modi: ad uno ad uno, sequenzialmente, su di un singolo filo (come nell'interfaccia seriale RS232), oppure a gruppi, su più fili (come nelle interfacce parallele Centronics o IEEE488).
Le stampanti differiscono, infine, anche riguardo al tipo di carta impiegato. In alcune, la carta può essere inserita un foglio alla volta, come nella macchina da scrivere. In altre, la carta, sotto forma di moduli continui opportunamente perforati ai lati, è trascinata da uno speciale congegno che ne assicura un corretto scorrimento. Il sistema è simile a quello impiegato nelle macchine fotografiche per l'avanzamento del film.

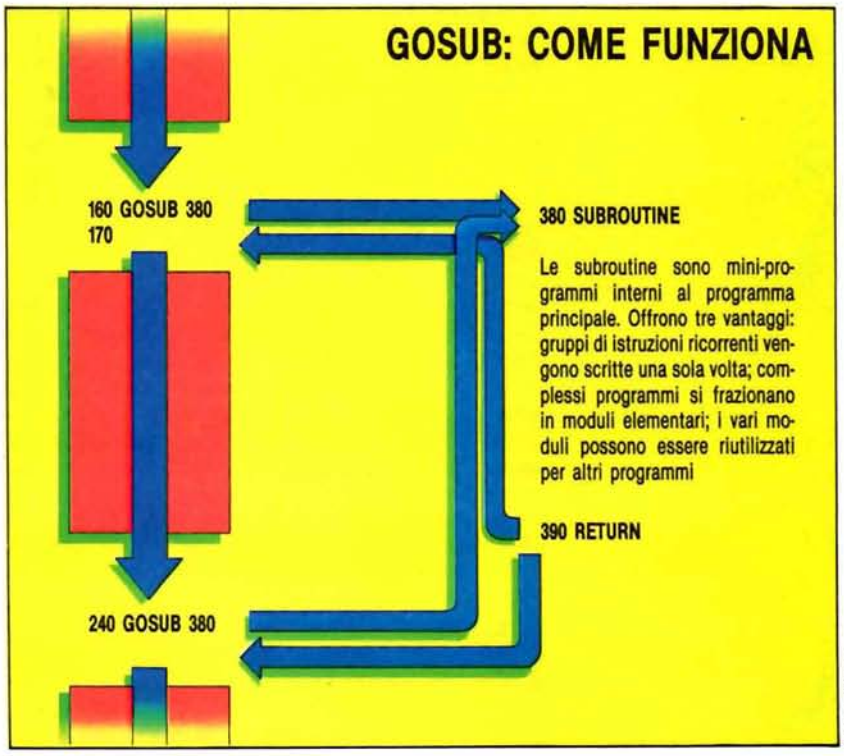
Parliamo di routine

L'uso dei sottoprogrammi in BASIC facilita la programmazione

Nelle passate lezioni del corso di programmazione in BASIC, abbiamo immesso programmi nella memoria del computer, li abbiamo eseguiti, modificati ed eseguiti di nuovo. Per inserire un nuovo programma, abbiamo ogni volta azzerato la memoria, usando il comando NEW spegnendo l'apparecchio. Se volessimo eseguire nuovamente un vecchio programma, dovremmo riscriverlo nella memoria del computer.
Per evitare un simile spreco di energie, tutte le versioni del BASIC dispongono di un comando che permette di archiviare su nastro o su disco il programma inserito. Il programma che segue, serve per calcolare il numero di piastrelle necessarie per decorare le pareti di una stanza. Una volta immesse tutte le istruzioni, possiamo "salvare" il nostro lavoro su nastro (o su disco) usando il comando SAVE, seguito dal nome del programma.

```
10 REM QUESTO PROGRAMMA CALCOLA IL NUMERO DI PIA-
   STRELLE NECESSARIE
20 REM A DECORARE UNA STANZA
30 PRINT "LUNGHEZZA LATERALE DELLA PIASTRELLA IN
   MM."
40 INPUT A1
45 LET T = 0
50 REM LA LINEA 60 CALCOLA L'AREA
60 LET A2 = A1 * A1
70 PRINT "NUMERO DI PARETI"
80 REM W VIENE USATO COME MASSIMO NEL CICLO
90 INPUT W
100 FOR X = 1 TO W
110 PRINT "LUNGHEZZA DELLA PARETE N."; X; "IN METRI"
120 REM D È LA DIMENSIONE DELLA PARETE
130 INPUT D
140 REM LA DIMENSIONE D VIENE CONVERTITA IN
150 REM MM. NELLA SUBROUTINE 380
160 GOSUB 380
170 REM LA LINEA 190 ASSEGNA A L
180 REM LA LUNGHEZZA IN MILLIMETRI DI D
190 LET L = D2
200 PRINT "ALTEZZA DELLA PARETE N."; X; "IN METRI"
210 REM LE LINEE DA 230 A 250 ASSEGNANO AD H
220 REM L'ALTEZZA DEL MURO IN MM.
230 INPUT D
240 GOSUB 380
250 LET H = D2
260 REM LA 270 ASSEGNA AD A3 L'AREA DELLA PARETE
270 LET A3 = L * H
280 REM S (TOT. PARZIALE) È L'AREA DI PARETE
290 REM DIVISO L'AREA DI UNA PIASTRELLA
300 LET S = A3/A2
310 REM A T (TOTALE) VENGONO SOMMATI I PARZIALI
320 REM AD OGNI GIRO DEL CICLO
330 LET T = T + S
340 NEXT X
350 REM STAMPA DEL TOTALE
360 PRINT T
370 END
380 LET D2 = D * 1000
390 RETURN
```

A questo punto, per conservare il programma su nastro, usiamo il comando SAVE. Ovviamente, il registratore deve essere collegato come spiegato nel manuale del computer ed essere pronto all'uso. Inseriamo una cassetta nuova nel registratore e digitiamo SAVE, seguito da uno spazio e da un nome a nostro piacimento, racchiuso fra doppi apici. Questo nome rappresenta il nome del "file"; che viene archiviato. Il termine "file", molto adoperato in programmazione, indica una raccolta di codici (siano essi programmi o dati) conservati su nastro o disco. Il nome del "file" con il quale archiviare il programma è a nostra discrezione, ma conviene usarne uno che ci ricordi la funzione del programma, ad esempio: "DECORA".
In alcuni registratori il motore è direttamente controllato dal computer, in altri si dovrà avviare la registrazione manualmente (ricordandosi, in ambedue i casi, che esso va



predisposto alla registrazione). Avendo già digitato SAVE "DECORA", possiamo premere il tasto RETURN per avviare il processo di archiviazione.
Per verificare se il programma è stato archiviato correttamente, si può azzerare la memoria del computer con il comando NEW <CR>. Riavvolto il nastro e predisposto il registratore all'ascolto, si procede al caricamento del programma (da nastro a memoria) con il comando LOAD. Il comando LOAD deve essere seguito dal nome del file (tra doppi apici), ossia: LOAD "DECORA" <CR>.
A caricamento avvenuto, appare un messaggio sullo schermo (READY oppure OK). Possiamo adesso usare il comando LIST per controllare se il programma è stato trasferito in modo corretto.

La GOSUB

Il comando GOSUB serve per deviare l'esecuzione del programma verso una "subroutine". Una subroutine è una specie di mini-programma, ossia è un programma nel programma. La subroutine impiegata nel programma "DECORA" è estremamente semplice. È stata inclusa proprio a titolo di esempio, tanto che il programma poteva essere scritto anche senza subroutine.
Il nostro programma calcola il numero di piastrelle necessarie per coprire una certa superficie. Oltre alla lunghezza della piastrella (che per semplicità è di forma quadrata), viene chiesta la lunghezza e l'altezza di ciascuna parete da coprire. Il numero totale di piastrelle viene calcolato dividendo la superficie totale da coprire per la superficie di ciascuna piastrella. La subroutine alla linea 380 converte una lunghezza da metri a millimetri, semplicemente moltiplicando per 1000. Le subroutine offrono tre vantaggi: gruppi ricorrenti di istruzioni possono essere scritti una sola volta, indipendentemente dal numero di volte che vengono usati nel programma; programmi lunghi e complessi possono essere frazionati in più moduli elementari, più maneggevoli e comprensibili; infine, le subroutine possono, all'occorrenza, essere copiate e riutilizzate in più programmi.
La nostra subroutine inizia alla linea 380 e consiste di una sola frase: LET D2 = D * 1000. La variabile D (lunghezza o altezza della parete in metri) viene moltiplicata per 1000, per cui si ottiene la misura in millimetri. Il risultato viene assegnato alla variabile D2.
L'istruzione che obbliga il programma ad eseguire la subroutine è la GOSUB, che incontriamo per la prima volta nella linea 160. Il valore di D è stato assegnato nella 130. La 160 chiama la subroutine, dove viene calcolata la variabile D2. L'istruzione RETURN, nella linea 390 è necessaria per ritornare al programma principale. L'esecuzione riprende sempre dalla linea successiva a quella che contiene la GOSUB, in questo caso dalla linea 170.
Incontriamo una seconda GOSUB nella linea 240. Questa volta, eseguita la solita subroutine, l'esecuzione riprende dalla 250. In un programma si possono usare quante subroutine si desiderano: non ci sono limiti. L'importante è ricordarsi di terminare ogni subroutine con RETURN. Si noti, inoltre, che l'istruzione END della linea 370 precede la subroutine. La END indica la fine del programma principale e serve a bloccare l'esecuzione prima dell'inizio della subroutine.
Sebbene il programma presentato in questa lezione sia più lungo dei precedenti, non è affatto più complesso. Si provi a seguirne il percorso sul listato, per capirne il funzionamento. Oltre alla subroutine, viene introdotto soltanto un nuovo elemento: nomi più lunghi delle variabili.
Per meglio seguire i vari risultati, può essere utile disegnarsi a parte dei piccoli riquadri, che rappresentino le variabili, ed annotarsi via via i valori di ciascuna variabile.
Va notato, inoltre, che la linea 300 può produrre anche valori non interi. Proviamo ad immettere i seguenti valori: 110 per la piastrella; per la parete: 2,3 metri (lunghezza) e 1,8 metri (altezza), usando una sola parete. Il risultato sarà 342,149 piastrelle. Poiché le piastrelle vengono vendute intere, il risultato non è rigorosamente esatto. Nella prossima lezione vedremo come ovviare a questo problema.

Esercizi

* Si provi ad usare 0 per la misura della piastrella. Ciò dovrebbe causare un errore e l'interruzione del programma. Perché? Come mai ciò non accade se specifichiamo una lunghezza della parete uguale a zero? (Suggerimento: moltiplicando o dividendo per zero non è la stessa cosa!).

* Il programma funziona soltanto con piastrelle quadrate. Si provi a modificare le linee 30 e 60 per trovare l'area di piastrelle rettangolari. (Il sistema è simile a quello usato per le pareti).

* Si inserisca una linea 355 alla quale far aumentare il numero totale di piastrelle del 5%, per crearci una scorta di piastrelle. (Moltiplicando un numero per 105/100 esso viene incrementato del 5%).

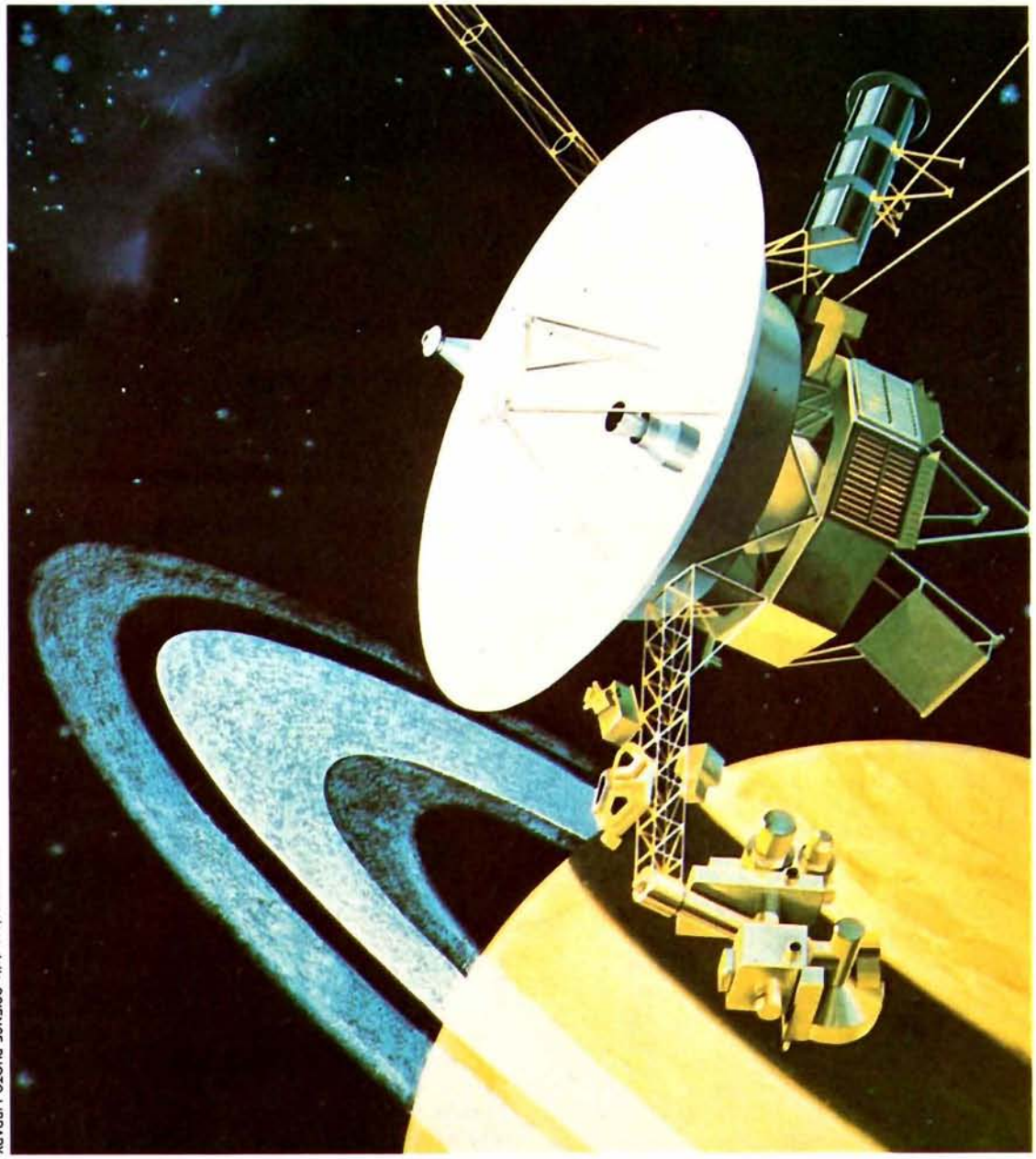
Basic & Basic

END Lo Spectrum, lo ZX81, l'Oric-1 ed il Dragon 32 non hanno questa istruzione. Sostituire con STOP

GOSUB Nello Spectrum e nello ZX81 l'istruzione viene visualizzata come GOSUB, sebbene la si digiti con un solo tasto

Quando meno vale più

I computer sottraggono... sommando!



per concessione della SCIENCE PHOTO LIBRARY

Il Voyager 2

Il Voyager è il primo esploratore spaziale che si sia spinto oltre il nostro sistema solare. Viaggiando nello spazio, ha ripreso fotografie e raccolto un'enorme serie di dati, tradotti in numeri binari dal computer di bordo. I dati venivano trasmessi verso terra all'incredibile velocità di 116.000 bit al secondo. I dati venivano quindi elaborati dai computer della NASA a Houston, Texas

Nella prima parte di questa serie abbiamo scoperto che qualsiasi numero decimale può essere rappresentato in forma binaria. I numeri binari, anche se hanno il difetto di essere molto più lunghi dei loro equivalenti decimali, sono facilmente utilizzabili dai computer, dove l'uno e lo zero possono essere rappresentati da una tensione positiva o nulla. Abbiamo anche visto che è facile sommare i numeri binari.

Sulla carta, si può eseguire altrettanto facilmente anche la sottrazione di due numeri binari, usando le stesse regole della sottrazione decimale. Già da molto tempo, però, i

progettisti di computer si sono accorti che i "sommatori" (i circuiti che eseguono le somme) possono benissimo eseguire anche le sottrazioni. Vediamo come ciò è possibile.

Il complemento a due

Uno dei metodi usati per rappresentare i numeri negativi nei computer si chiama "complemento a due". Il processo di sottrazione, secondo questo metodo, non è che un'estensione della somma. Consideriamo il seguente problema algebrico:

$$16 - 12 = 4$$

oppure

$$16 + (-12) = 4$$

Il valore 12 viene sottratto a 16, ma si può anche considerare la somma di 16 e −12. Il risultato, in ambedue i casi, è identico, cambia solo l'uso dei segni aritmetici e delle parentesi. Questo principio è stato adottato per semplificare l'operazione sui computer.
Supponiamo, per semplicità, che il nostro computer possa elaborare numeri binari di sole 5 cifre (ovviamente i computer lavorano su numeri binari molto più grandi). Il nostro computer adotta un semplice sistema per indicare i numeri negativi: la cifra iniziale viene considerata separatamente dalle altre quattro più a destra, che per il momento consideriamo tutte a 0. Se la cifra iniziale è 1 il numero rappresenta −16, se è 0 il numero rappresenta zero. Le restanti 4 cifre vengono considerate positive e seguono la normale convenzione che abbiamo visto la scorsa volta:

[]	[]	[]	[]	[]
−16 o 0	8	4	2	1

Così, ad esempio, il numero binario 01000 equivale a 8 decimale, mentre 10000 vale −16. Quanto vale 10100? La risposta è −12 (ossia −16 più 4). Con qualche altro esercizio si può vedere che possono essere rappresentati tutti i numeri tra −16 e +15 :

Sistema:	Binario	Decimale
	10000	−16
	10001	−15
	10010	−14
	10011	−13
	10100	−12
	ecc.	
	11111	−1
	00000	0
	00001	1
	00010	2
	ecc.	
	01110	14
	01111	15

Aumentando il numero di cifre oltre le 5 qui considerate, potremmo ovviamente elaborare una più ampia gamma di numeri.
Agli albori dell'aritmetica binaria, venne scoperto un semplice trucco per calcolare il complemento a due (ovvero il numero cambiato di segno) di un numero binario. Occorrono due passi.
Per prima cosa, si inverte il valore di ogni cifra: ad ogni 1 si sostituisce uno 0 e viceversa; poi si somma 1 al risultato.

Applichiamo il metodo ad un esempio, in cui usiamo il numero decimale +12, il cui equivalente binario è 01100.

	01100	(= +12)
primo passo	10011	
secondo passo	00001	(+1)
	10100	(= −12)

Vediamo ora come il computer esegue la sottrazione; per esempio: 12 − 4:

(+ 12)	01100	
(− 4)	11100	usando il complemento a due
	101000	

Si noti che adesso abbiamo un risultato di 6 cifre. Poiché il nostro computer ha solo 5 cifre, quella più a sinistra (chiamata "overflow") viene ignorata, lasciandoci con 01000, ovvero 8 decimale, la risposta giusta!

Ed ora vediamo un esempio leggermente più complesso è 4 − 12:

(+ 4)	00100	
(− 12)	10100	
	11000	(= −8)

Facciamo un ultimo esempio, con due numeri negativi, −3, −4 = −3 + (−4) = −7

(+3)	00011	
(−3)	11101	(complemento a due)
(−4)	11100	
	111001	

Il risultato è nuovamente di 6 cifre. Scartato l'"overflow", abbiamo 11001, ovvero −7 decimale.
In tutti gli esempi le sottrazioni sono state eseguite mediante la somma ed il complemento a due.
Il sistema si implementa bene sui computer, in cui, per invertire il valore di una cifra binaria viene impiegata una porta logica NOT (vedere a pagina 68).
All'uscita di una porta NOT si trova sempre il valore opposto a quello d'ingresso. Se in ingresso si ha 0, l'uscita vale 1 e viceversa. Questa funzione è richiesta dal primo passo del "complemento a due".
Vedremo nella prossima lezione in qual modo le porte logiche consentono di eseguire le somme.

Obiettivo Scuola

Nuovi criteri nell'insegnamento scolastico



IAN DOBBIE

Il computer nelle scuole elementari

Sono in fase avanzata di studio le modalità per introdurre l'uso dei computer nelle prime classi elementari. Lo scopo non è soltanto quello di far familiarizzare i bambini con le nuove tecnologie, ma anche quello di insegnare, col computer, una vasta gamma di materie tradizionali: dalla biologia alla musica, dalle lingue straniere alla geografia. Il computer si presta bene a questo ruolo, poiché si mostra sempre "paziente" con l'allievo, aiutandolo gradualmente a superare le difficoltà di apprendimento

Presto anche in Italia ogni scuola avrà un computer. Questi apparecchi non verranno impiegati solamente per l'insegnamento dell'informatica, ma anche per la maggior parte delle altre materie.

Nei paesi anglosassoni sono disponibili molti programmi specifici per l'insegnamento, alcuni dei quali, però, non incontrano il favore degli insegnanti, poiché non sempre il software è progettato adeguatamente ai programmi scolastici e all'eterogeneità delle classi.

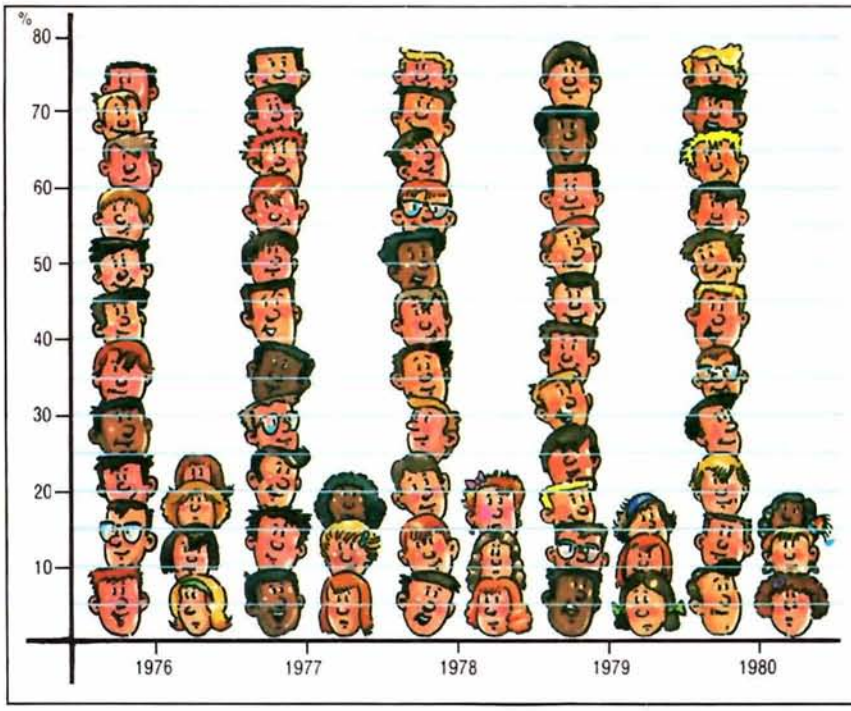
Ciò dipende, almeno in parte, dalla enorme e quanto mai improvvisa diffusione dei microcomputer economici. Attualmente, infatti, è raro incontrare programmatori che abbiano esperienza nel campo didattico, com'è altrettanto difficile trovare, nelle scuole, insegnanti con una sufficiente preparazione nel settore dell'informatica, che rischierebbero così di confondere, anziché chiarire, le idee agli allievi.

Alcuni insegnanti si cimentano nella stesura di programmi dimostrativi, che, pur funzionando per una classe, non sono adatti per altre. Oppure, sia per la frequente incompatibilità tra i vari computer, sia per l'inadeguatezza del manuale d'uso, tali programmi sono inutilizzabili presso altre scuole.

Scrivere programmi efficienti, adatti all'insegnamento, richiede, infatti, molta cura ed attenzione: occorre prevenire i dubbi e gli errori di chi è alle prime armi, offrendo ampie spiegazioni, anche sulle operazioni più semplici ed apparentemente ovvie. Ad esempio, un programma non deve bloccarsi se viene data una risposta errata o viene premuto un tasto sbagliato. Inoltre, esso deve essere di facile uso e comprensione per stimolare l'interesse dell'allievo.

Malgrado queste difficoltà di programmazione, che andranno lentamente scomparendo, il computer rimane uno strumento efficacissimo per la didattica e, anche con un software dalle limitate funzioni, è possibile che venga utilizzato con profitto nell'apprendimento di tutta una gamma di materie, compresa l'informatica.

I programmi didattici, in genere, adottano un



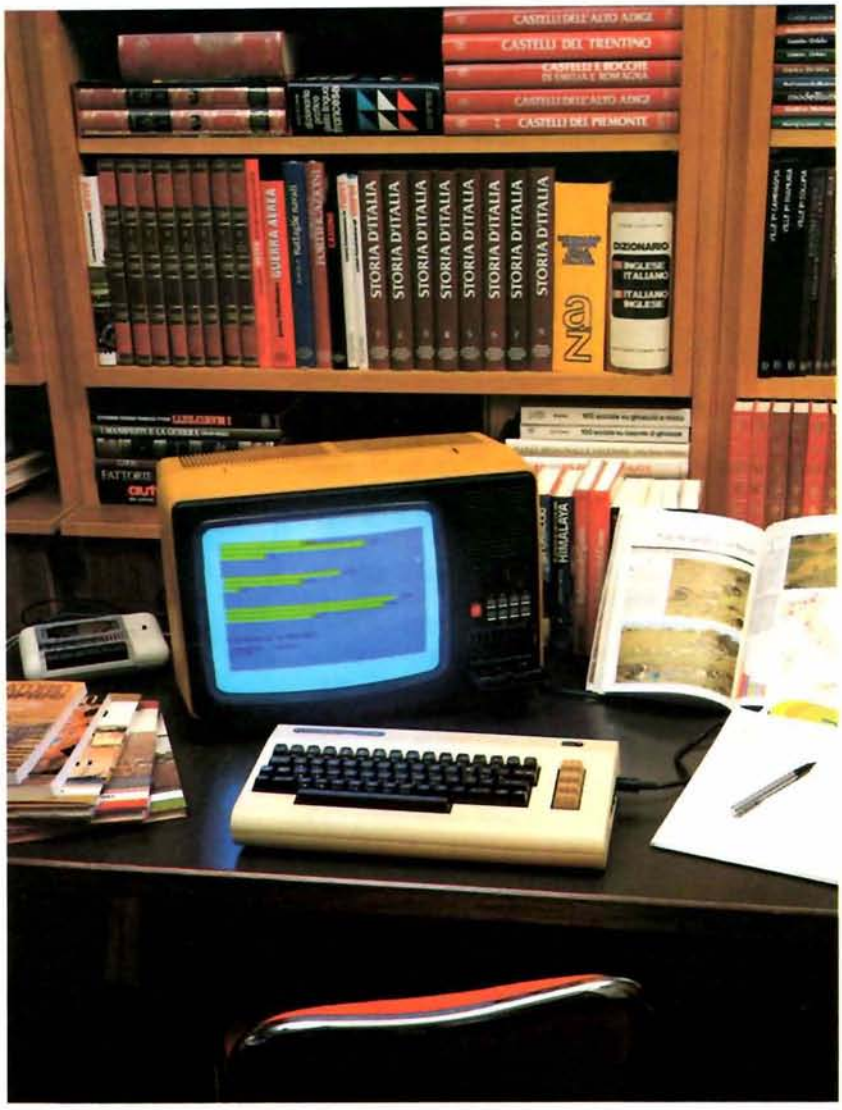
L'ESEMPIO DELLA GRAN BRETAGNA

L'illustrazione mostra, per il periodo 1976-1980, il diverso grado di interesse, che gli alunni (distintamente i maschi e le femmine) della scuola media inglese hanno rivolto all'informatica. Si può notare che la popolazione scolastica maschile è preponderante, ma gli esperti confidano in una distribuzione più omogenea nel prossimo futuro, anche grazie alla maggiore diffusione dei computer nelle scuole elementari

ANNI AXWORTHY

Computer a scuola, computer a casa

L'uso di piccoli computer economici si va sempre più diffondendo nell'ambito scolastico, dove non vengono utilizzati soltanto per l'insegnamento dell'informatica, ma anche per quello di tutte le altre materie. Alcuni genitori, inoltre, acquistano un home computer, affinché i propri figli possano esercitarsi anche a casa.



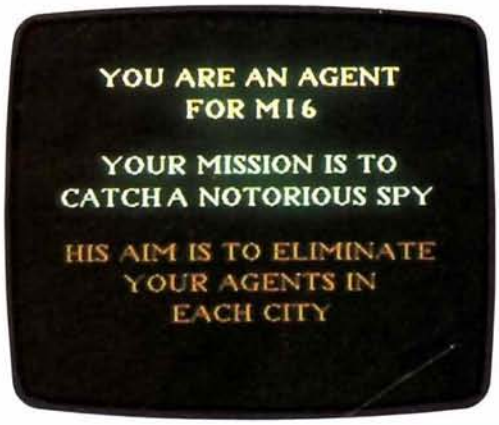
metodo basato sulla presentazione di alcuni esempi, seguiti da esercitazioni pratiche, sulle quali viene mantenuto un "punteggio".
Talvolta, se la risposta fornita dall'allievo non è esatta, viene visualizzato un messaggio d'incoraggiamento.
La scelta di un particolare programma didattico dipende da tre fattori: l'età dell'allievo, la materia di studio ed il tipo di computer disponibile.
Oggigiorno, molti genitori acquistano un "home computer", con il preciso scopo di fornire ai loro figli uno strumento didattico. In questi casi, prima di procedere all'acquisto dell'apparecchio, può essere utile informarsi sul tipo di computer adottato (o previsto) nella scuola frequentata dai figli. Se, infatti, il modello è perlomeno simile, ci sono buone probabilità di usare a casa gli stessi programmi impiegati a scuola. In certi casi, è possibile ottenere una valida assistenza (e magari una copia dei programmi) dalla scuola stessa.
Ovviamente, per gli apparecchi più diffusi esiste una più ampia scelta di software didattico, anche se, per il momento, gran parte di esso è in lingua inglese. I computer più avvantaggiati in questo settore sono: l'Apple, il PET Commodore, il Tandy, il BBC, i Sinclair ed il Texas Instruments.
Le case di software, abitualmente, offrono programmi in più versioni, adatte alle varie marche.

La scelta giusta

Per la fascia di età al di sotto degli otto anni, i programmi didattici vertono principalmente sulle nozioni basilari di aritmetica e di compitazione delle parole.

La Texas Instruments (TI), pur rimanendo essenzialmente una ditta produttrice di hardware, si è ampiamente dedicata allo sviluppo di piccoli programmi didattici, assieme alla "Scott, Foresman & Co." americana. Alcuni programmi di questa gamma, che presto dovrebbero uscire in versione italiana, sono dedicati all'insegnamento della grammatica e delle quattro operazioni aritmetiche. I programmi sono conservati su cartucce da inserire nel computer ed alcuni di essi, grazie all'impiego di uno speciale sintetizzatore vocale, sono in grado di "parlare" all'allievo. La TI offre anche una buona versione del linguaggio LOGO (vedere a pagina 30).
In Italia, alcune case editrici si stanno muovendo per offrire software didattico e presto saranno disponibili programmi per i computer più diffusi. In Gran Bretagna, dove il fenomeno ha assunto dimensioni notevoli, la quantità di software didattico cresce di giorno in giorno e la scelta è veramente ampia.
I programmi per la fascia d'età tra gli 8 e gli 11 anni sono rivolti all'approfondimento di materie elementari, stimolando, al tempo stesso, le capacità creative e deduttive del giovane. Sono anche disponibili, per questa fascia d'età, programmi per l'insegnamento della musica e delle lingue straniere.
Indubbiamente, qualunque sia l'età degli allievi, è essenziale che in futuro vi sia, in questo settore, un'assistenza notevole da parte degli organismi statali preposti all'insegnamento. Solo così sarà possibile coordinare opportunamente l'impiego didattico dei computer.
Un'ulteriore gamma di programmi è orientata verso la fascia degli 11/13 anni. Per i ragazzi di questa età, l'apprendimento è ancora ricco di scoperte ed i programmi devono riflettere questa situazione, stimolando quanto più possibile l'esplorazione del "mondo esterno". Il migliore programma attualmente disponibile è il LOGO, che è già stato adattato per la maggior parte degli "home" e "personal" computer. Questo particolare programma, che può essere usato da bambini di età compresa tra i 6 ed i 12 anni, offre l'opportunità di imparare a disegnare col computer (gli allievi apprendono, al tempo stesso, elementi di programmazione, di matematica e di geometria).
Il settore della didattica computerizzata è in costante crescita e nuovi programmi, sempre più sofisticati e completi, vengono immessi giornalmente sul mercato. Per tenersi aggiornati sulle novità, può essere utile visitare le tante mostre sui computer che, periodicamente, vengono allestite per la presentazione sia di apparecchi che di programmi. Anche molte riviste specializzate dedicano degli articoli a questo particolare argomento.
Indubbiamente, nell'immediato futuro, il ruolo degli insegnanti, ma forse ancor più quello dei genitori, cambierà enormemente.
Sia gli uni che gli altri, infatti, devono essere in grado di adeguarsi rapidamente al notevole progresso culturale implicito nell'impiego dei computer.

L'AGENTE SEGRETO



Tra i molti programmi, a metà tra il gioco e la didattica, ne esiste uno chiamato "L'agente segreto", che tra non molto dovrebbe essere disponibile in versione italiana. Si tratta di un gioco di simulazione, dove assumiamo il ruolo, appunto, di un agente segreto. Il nostro incarico consiste nell'acciuffare una spia nemica, prima che questa elimini tutti i nostri agenti. Via via vengono fornite delle tracce per localizzare la spia

La spia si sposta continuamente tra varie città, soggiornandovi solo poche ore. Se pensiamo di averla individuata in una particolare città, abbiamo la scelta di usare il treno o l'aereo per raggiungerla. La nostra scelta deve tener conto sia del tempo a disposizione, che delle spese di viaggio





Sulla mappa riprodotta sullo schermo appaiono di volta in volta le segnalazioni inviateci dai nostri agenti e anche la città di provenienza. Abbiamo inoltre la facoltà di ingaggiare nuovi agenti, ma ciò aggrava le spese

Alcune delle segnalazioni devono anche essere decifrate. Giocando, si imparano i nomi delle città europee e la loro collocazione geografica, oltre ad apprendere l'uso di orari di treni ed aerei. Inoltre, ci si abitua ad un'attenta pianificazione delle spese: infatti, rischiamo di essere "licenziati", se ... spendiamo troppo nell'operazione!



IAN McKINNELL

Codice segreto

La conversione da linguaggio "evoluto" a "codice macchina"

Benché tutti i computer siano apparentemente simili, ogni marca e modello ha caratteristiche particolari. Alcuni vengono forniti completi di un certo numero di programmi, mentre in altri i programmi vanno letti da dischetti o da nastri magnetici.
Alcuni apparecchi, inoltre, dispongono di un unico programma di gestione interna, che permette l'uso di comandi tipo SAVE e LOAD (per il caricamento e l'archiviazione di programmi), mentre in altri computer la gestione è affidata a più programmi.
Vi sono, comunque, alcuni principi di base, comuni alla maggior parte dei microcomputer: il flusso di dati tra le unità di memoria esterne (dischi e nastri) e lo schermo del computer avviene sempre sotto il controllo della tastiera. Lo stesso accade per altre unità periferiche (stampanti, plotter, ecc.). Un altro fattore comune, inoltre, è l'impiego di linguaggi cosiddetti "evoluti", come ad esempio il BASIC.
Quando si scrive un programma, immettendo le varie istruzioni mediante la tastiera, uno speciale programma interno al computer (il "sistema operativo"), trasferisce le varie informazioni digitate sia allo schermo, per la visualizzazione, sia al programma "interprete" del BASIC, per la decodifica. In altre parole, nel computer sono memorizzati tre distinti programmi: il sistema operativo, l'interprete BASIC ed il programma da noi immesso. Durante l'esecuzione del programma in BASIC, può sembrare che tutti e tre i programmi siano attivi contemporaneamente: ogni istruzione BASIC del nostro programma è tradotta, dall'interprete BASIC, in una serie di istruzioni, passate ed eseguite una alla volta dal microprocessore. Allo stesso tempo, il sistema operativo controlla se viene battuto uno dei pulsanti della tastiera, visualizzandolo se occorre.
Se l'istruzione BASIC da noi impartita implica l'uso, la lettura o la scrittura di dati su dischetto, l'interprete chiede al sistema operativo di provvedere alla funzione richiesta. L'illusione che tutto ciò avvenga contemporaneamente è dovuta all'incredibile velocità di lavoro del microprocessore, che riesce ad elaborare quasi istantaneamente le istruzioni provenienti sia dal sistema operativo che dall'interprete BASIC.
In alcuni apparecchi, per rendere ancora più rapidi i processi di elaborazione, viene impiegato, per la tastiera o altre periferiche, un sistema detto "a interruzione": la tastiera, per esempio, comunica al sistema operativo, con un particolare segnale elettrico, che è stato battuto un carattere, evitando così le perdite di tempo causate da una continua scansione della tastiera.
Il linguaggio BASIC può essere fornito sotto forma di "interprete" o di "compilatore". La differenza è sostanziale e vale la pena di chiarirla: l'"interprete", durante l'esecuzione di un programma, deve convertire ciascuna istruzione BASIC in una serie di istruzioni in linguaggio macchina, eseguibili dal microprocessore. Il "compilatore", invece, converte in linguaggio macchina l'intero programma, una volta per tutte. I programmi "compilati" offrono una maggiore velocità di esecuzione, che non i loro equivalenti "interpretati".
Il BASIC è un ottimo linguaggio per principianti, per la facilità con la quale lo si apprende e lo si usa, ma i programmatori più esperti, per velocizzare il funzionamento dei programmi, preferiscono usare il linguaggio "Assembler", più complicato e più vicino al linguaggio macchina.
In pratica, con l'Assembler, ogni istruzione del programma equivale ad una istruzione in linguaggio macchina: è pertanto indispensabile una profonda conoscenza del funzionamento del particolare microprocessore impiegato. Un programma in Assembler è costituito da istruzioni del tipo: MVI (da **M**ove **I**mmediate = Caricamento immediato) o JZ (da **J**ump on **Z**ero = "salta" in condizione Zero). Queste istruzioni sono "mnemoniche",

Il sistema operativo per dischi

Quando un programma viene archiviato su disco, i blocchi di informazioni che lo compongono non sono distribuiti uniformemente, ma a seconda degli spazi disponibili. Il sistema operativo per dischi (chiamato anche DOS, da Disk Operating System) tiene nota della distribuzione delle informazioni in modo da poterle rileggere senza generare errori. L'illustrazione mostra una piccola porzione delle informazioni archiviate su di un dischetto. A sinistra, il contenuto di ciascun byte è rappresentato in forma esadecimale, mentre a destra è rappresentato sotto forma di carattere alfabetico. I codici che non corrispondono a caratteri visualizzabili vengono rappresentati con un punto



ossia sono abbreviazioni che aiutano il programmatore a ricordare la funzione dell'istruzione vera e propria.
Impiegando l'Assembler, si lavora con un sistema numerico detto "esadecimale", ovvero a base 16. I numeri dallo 0 a 9 si scrivono nella forma normale, mentre dal 10 al 15 si adoperano le lettere A, B, C, D, E ed F. Ogni locazione di memoria contiene otto "bit", rappresentabili con due cifre esadecimali.
Per esempio, il numero binario 01011101 viene scomposto in 0101 ed 1101 e rappresentato con le due cifre esadecimali 5 e D (rispettivamente 5 e 13 in notazione decimale). È indubbiamente più facile usare l'esadecimale 5D, che non l'ingombrante 01011101. Confrontiamo brevemente i due linguaggi:

BASIC:
```
100 INPUT "Quante ore di lavoro";
ORE 200 PAGA = ORE * PAGAORARIA
```

La prima linea invita, visualizzando il messaggio tra doppi apici, ad immettere il numero di ore lavorate. Nella seconda linea, il numero appena introdotto viene moltiplicato per la paga oraria (immessa precedentemente) ottenendo il valore della paga totale.

Assembler:
```
MVI C,1
CALL 05
.
```

La prima istruzione "carica" il valore 1 in una speciale locazione di memoria del microprocessore (detta registro "C"). La seconda, passa il controllo al sistema operativo, che, eseguite particolari funzioni, ritorna al programma "chiamante" per continuare l'esecuzione. Le due istruzioni, in esadecimale, valgono:

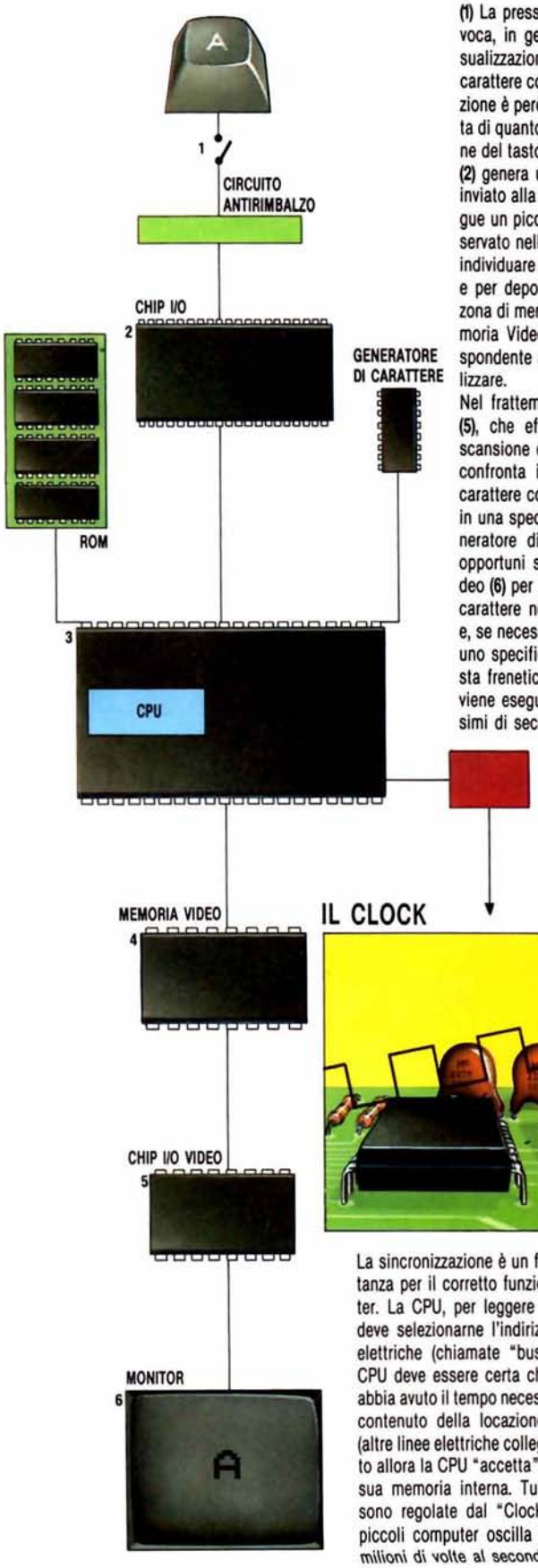
Linguaggio macchina:
```
OE 01
CD 05 00
```

La conversione delle due linee di programma BASIC, viste sopra, richiede molte istruzioni in linguaggio macchina. È perciò più comodo avvalersi di un interprete o di un compilatore che operi la conversione automatica.
Talvolta, alcuni comandi tipici di un sistema operativo sono a prima vista incomprensibili quanto le istruzioni in Assembler. Ecco un esempio tratto dal sistema operativo CP/M (Control Program for Microcomputers = Programma di Controllo per Microcomputer):

```
DIR *.BAS
```

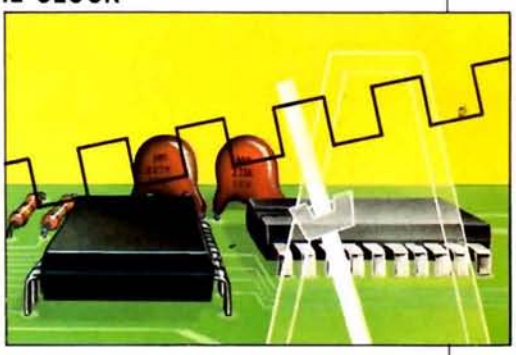
che significa: «elenca tutti i files presenti sul dischetto, che abbiano estensione BAS». DIR è l'abbreviazione di Directory (elenco).
Nonostante i comandi del CP/M non siano esempi di chiarezza, questo sistema operativo è stato installato su più di un milione di computer. Esso offre il notevole vantaggio di poter "trasportare" programmi da un computer all'altro (anche di marche diverse) senza dover apportare modifiche, purché tutte le operazioni relative alla gestione della tastiera, dei dischi e del video vengano affidate al CP/M mediante opportune istruzioni. Per questo motivo, la biblioteca di programmi disponibili per il CP/M è una delle più estese.

DAL TASTO ALLO SCHERMO



(1) La pressione di un tasto provoca, in genere, l'immediata visualizzazione sullo schermo del carattere corrispondente. La funzione è però molto più complicata di quanto sembri: alla pressione del tasto, infatti, il chip di I/O (2) genera un segnale che viene inviato alla CPU (3). La CPU esegue un piccolo programma, conservato nella memoria ROM, per individuare di quale tasto si tratti e per depositare in un'apposita zona di memoria RAM, detta Memoria Video (4), il codice corrispondente al carattere da visualizzare.
Nel frattempo, il circuito Video (5), che effettua una continua scansione della Memoria Video, confronta il codice di ciascun carattere con i codici conservati in una speciale ROM, detta "generatore di caratteri" ed invia opportuni segnali elettrici al video (6) per la visualizzazione del carattere nella giusta posizione e, se necessario, per selezionare uno specifico colore. Tutta questa frenetica serie di operazioni viene eseguita in pochi milionesimi di secondo

IL CLOCK



La sincronizzazione è un fattore di vitale importanza per il corretto funzionamento del computer. La CPU, per leggere un byte di memoria, deve selezionarne l'indirizzo su apposite linee elettriche (chiamate "bus degli indirizzi"). La CPU deve essere certa che il chip di memoria abbia avuto il tempo necessario per depositare il contenuto della locazione sul "bus dei dati" (altre linee elettriche collegate alla CPU). Soltanto allora la CPU "accetta" e ricopia il dato nella sua memoria interna. Tutte queste operazioni sono regolate dal "Clock" (orologio), che nei piccoli computer oscilla anche fino a quattro milioni di volte al secondo

STEVE CROSS

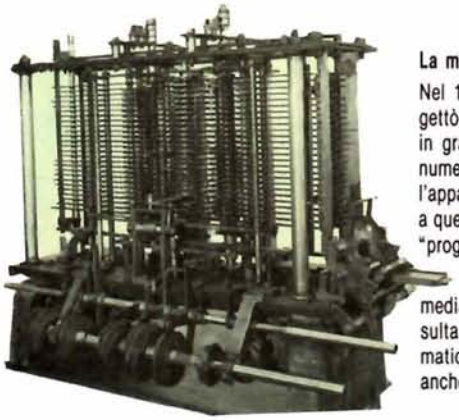
Dal pallottoliere al computer

Storia del computer dalle origini a oggi

Charles Babbage nel 1830 riuscì quasi a costruire il primo computer funzionante con un secolo di anticipo.
Babbage dové lottare contro diverse difficoltà: una di queste era costituita dal fatto che il computer doveva necessariamente essere meccanico; un'altra proveniva dall'inadeguatezza dell'ingegneria del suo tempo. Nonostante ciò, Babbage costruì una macchina che convinse il governo britannico di allora a finanziarlo inizialmente con 1.500 sterline, aumentate poi a 17.000 (oltre quattro miliardi di lire attuali).
La storia dei computer, comunque, è iniziata molto tempo prima. Fondamentalmente, un computer è un apparecchio che può essere programmato per eseguire calcoli su di un gruppo di numeri e che conserva i risultati di questi calcoli, passando, infine, ad altri calcoli. La prima forma di computer può essere considerata il pallottoliere, il cui uso, sebbene risalga a più di 4000 anni fa, è ancora oggi diffuso in alcuni paesi orientali. Grazie alla posizione delle palline, che rappresentano una forma primitiva di "memoria", si può fisicamente "vedere" una somma. Ma il pallottoliere non esegue calcoli automaticamente ed è laborioso operare su numeri molto grandi.
La prima calcolatrice meccanica venne inventata nel 1642 dal francese Blaise Pascal che, si dice, avesse voluto facilitare il lavoro del padre, ispettore delle tasse.
L'apparecchio funzionava egregiamente: il meccanismo che trasferiva i riporti dalla colonna delle unità a quella delle decine era basato sullo stesso principio impiegato nei moderni contachilometri delle auto.
Modestamente, l'inventore chiamò questo apparecchio "Pascalina".

Blaise Pascal

La "Pascalina" è stata la prima calcolatrice meccanica, progettata dal francese Blaise Pascal nel 1642. Egli voleva inventare una macchina capace di eseguire le quattro operazioni aritmetiche. La "Pascalina" impiegava speciali meccanismi per trasferire i riporti da una colonna all'altra. Pascal ottenne un brevetto dal re di Francia, ma la commercializzazione non ebbe molto successo



La macchina analitica

Nel 1834 Charles Babbage progettò una "macchina analitica" in grado di eseguire calcoli su numeri di 80 cifre. I principi dell'apparecchio sono molto simili a quelli degli odierni computer: i "programmi" venivano immessi mediante schede perforate, i risultati venivano stampati automaticamente ed era disponibile anche una limitata memoria



per concessione del SCIENCE MUSEUM

CRONOLOGIA DEL COMPUTER

1000 A.C.
Pallottoliere
Ancora in uso oggi in alcuni paesi orientali, calcola e conserva i numeri usando palline

1614
Logaritmi
Giovanni Nepero inventò questo sistema, riducendo le divisioni a sottrazioni e le moltiplicazioni a somme

1623
Sistema binario
Francesco Bacone usa per primo la numerazione in base 2

1642
La "Pascalina"
Blaise Pascal inventa la sua macchina per le somme e sottrazioni

1671
La calcolatrice
Gottfried von Leibnitz riesce a produrre una macchina per le moltiplicazioni e le divisioni

1802
Le schede perforate
Joseph Jaquard costruisce un telaio in cui la tessitura è "programmabile" grazie a schede perforate

1822
La "macchina differenziale"
La prima macchina matematica di Charles Babbage

1000 A.C. // 1614 1623 1642 1671 1802 1822

La "Pascalina" non riscosse un gran successo commerciale, tuttavia essa stimolò molto l'interesse degli scienziati di allora e, negli anni che seguirono, venne migliorata per più modi. Occorre giungere, però, a Charles Babbage e Ada Lovelace per incontrare qualcosa di veramente innovativo.

Charles Babbage nacque nel 1791 da una famiglia benestante e si rivelò ben presto un genio matematico. Fu proprio lavorando alla correzione dei numerosi errori da lui individuati nelle tavole logaritmiche, che ebbe l'idea di costruire un apparecchio di calcolo. Nel 1822 presentò alla Royal Astronomical Society il suo primo modello di "macchina differenziale": una macchina in grado di eseguire i calcoli necessari per costruire le tavole logaritmiche. Il nome della macchina derivò da una tecnica matematica chiamata "calcolo differenziale". La Royal Astronomical Society lo incoraggiò a perfezionare l'apparecchio.

Insieme con Ada Lovelace, figlia di lord Byron, intraprese un progetto molto ambizioso: la costruzione della "macchina analitica". Questo apparecchio doveva essere capace di calcolare i valori di funzioni matematiche molto più complesse dei logaritmi.

Ci furono, fin dagli inizi, difficoltà di ogni tipo e la macchina si rifiutava di funzionare. La macchina doveva essere gigantesca (a giudicare dai pochi disegni rimasti del progetto), tanto da occupare tutto l'enorme laboratorio di Babbage.

Occorreva tornire centinaia di ingranaggi, ruote, guide e mozzi e la tecnologia allora disponibile non era all'altezza. Finché si trattò di costruire un prototipo in scala ridotta, le inadeguatezze meccaniche si poterono ridurre entro limiti ragionevoli, ma in scala reale gli errori generati dalle parti in movimento resero l'apparecchio inutilizzabile.

Le intuizioni di Babbage erano giuste e, se solo egli avesse potuto disporre di una migliore tecnologia, la "macchina analitica" avrebbe funzionato. La logica di progetto e la struttura d'insieme dei moderni computer si basa ancora su concetti introdotti da Babbage, tanto che egli viene considerato uno dei fondatori dell'informatica.

In particolare, Babbage introdusse il concetto di "programmazione": insegnare, cioè, ad una macchina il metodo per eseguire calcoli matematici. Se fosse riuscito a concretizzare le sue idee, l'impero vittoriano sarebbe stato gestito da un "computer a vapore".

Ma per la conferma delle intuizioni di Babbage si dovette attendere fino al 1936, quando un giovane matematico di Cambridge, Alan Turing, pubblicò un articolo sulla rivista "On computable numbers". Il nome di Turing è rimasto pressoché sconosciuto al grande pubblico, ma il suo apporto contribuì notevolmente allo sviluppo dell'informatica.

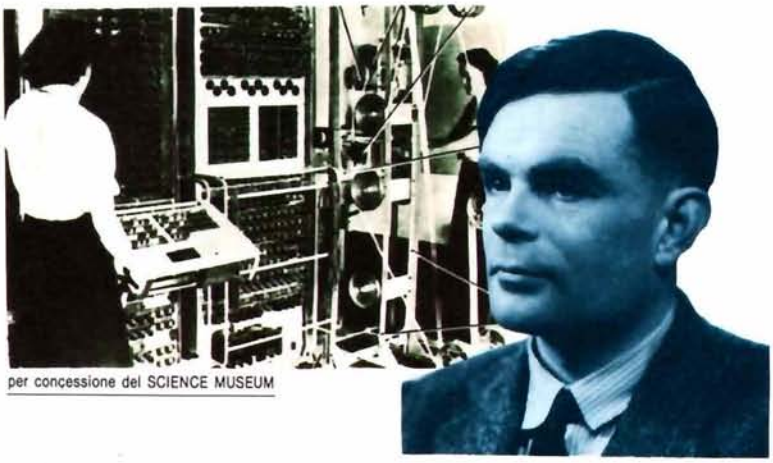
Gli scienziati erano da tempo giunti alla conclusione che la matematica non è una arte misteriosa, bensì una scienza completa-



per concessione della HMG COLLECTION

La prima programmatrice

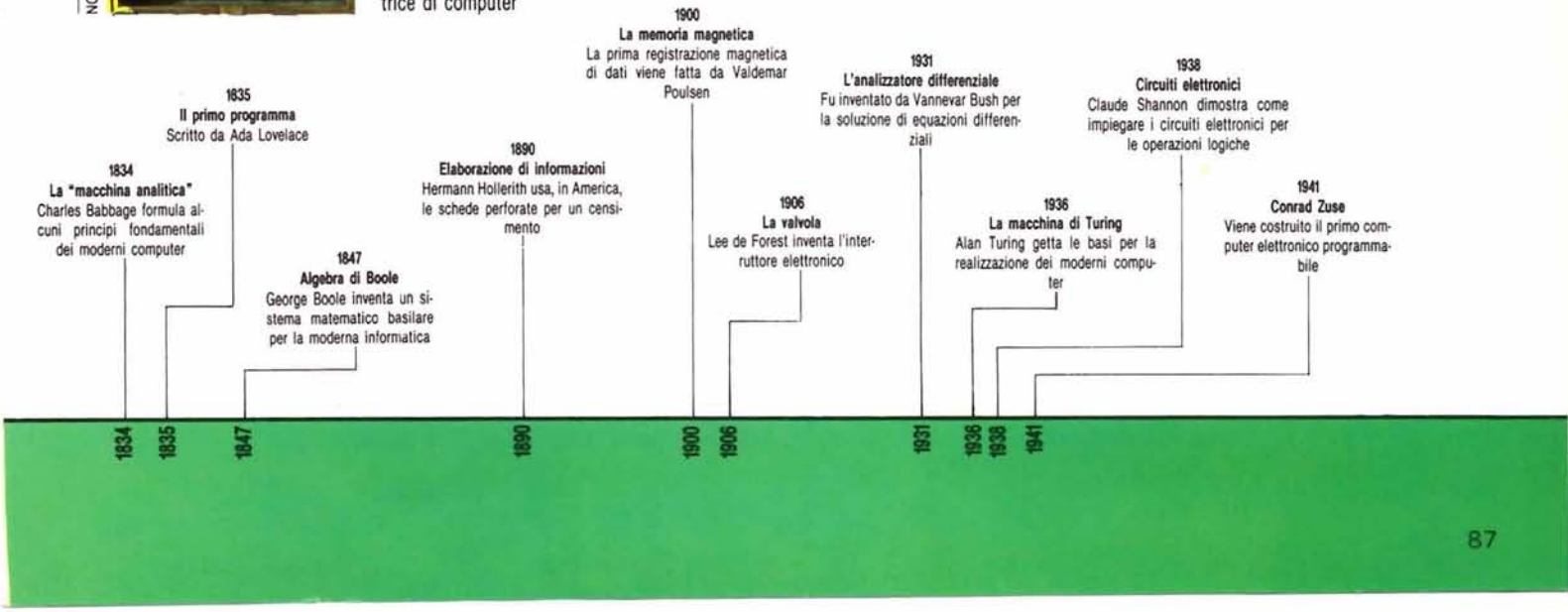
La contessa Ada Lovelace, compagna di Charles Babbage ed unica figlia legittima di lord Byron, è una delle poche donne presenti nella storia dei computer. Dotata di una mente matematica, comprese il funzionamento della "macchina analitica" di Babbage, lasciando dei precisi resoconti sul suo funzionamento. Scrisse anche alcuni programmi per quella macchina, diventando la prima programmatrice di computer



per concessione del SCIENCE MUSEUM

Alan Turing e il "Colossus"

Alan Turing dimostrò che con poche semplici istruzioni si possono risolvere anche complessi problemi. Con alcuni collaboratori, egli sviluppò il "Colossus", uno dei primi computer. La foto mostra "Colossus" in azione durante il secondo conflitto mondiale. Questo apparecchio, che conteneva migliaia di valvole ed era capace di elaborare 5000 caratteri al secondo, venne impiegato per decifrare il codice segreto tedesco "Enigma"



mente governata da precise regole logiche. Sarebbe quindi bastato "insegnare" ad una macchina quelle precise regole e, impostando i termini di un problema, ottenerne la soluzione.
Purtroppo tutti i tentativi erano fino ad allora falliti e Turing decise di affrontare il problema da un altro punto di vista. Egli esaminò attentamente ed elencò i tipi di problemi che una macchina, in grado di seguire regole logiche, avrebbe potuto risolvere.
Turing fu a capo di un gruppo di ricercatori e sviluppò l'invenzione più segreta della seconda guerra mondiale: Colossus, il primo computer elettromeccanico. Questo apparecchio riuscì a decifrare il codice segreto tedesco "Enigma", durante il conflitto.
Dopo la guerra, Turing andò negli Stati Uniti, dove partecipò al primo progetto americano di computer, l'ENIAC. L'apparecchio venne progettato dall'Università della Pennsylvania ed impiegava 18.000 valvole, (ne saltava una ogni due minuti, a causa dell'enorme calore generato dalla macchina).
La poca fama che Turing ebbe dipende anche dal fatto che egli lavorò per il servizio segreto britannico (MI6) e molto del suo lavoro era top secret. Alcuni particolari del lavoro di Turing sono stati resi noti soltanto nel 1975.

Lo sviluppo del computer subì molti progressi, il maggiore dei quali è rappresentato dall'invenzione del transistor al silicio (1947), che ha introdotto velocità di calcolo impressionanti, rispetto al tipo con le valvole.
I transistor possono sostituire le valvole in ogni loro funzione, lavorando molto più velocemente, più accuratamente e senza generare calore. Al pari delle valvole, i transistori vengono impiegati come interruttori elettronici. Durante gli anni '50 e '60 vennero costruiti potenti computer anche per applicazioni commerciali.
A metà degli anni '60, gli scienziati riuscirono a produrre circuiti miniaturizzati e, grazie anche alle ingenti somme di denaro investite nella ricerca spaziale, vennero prodotti i primi "circuiti integrati" su singoli "chip" di silicio. Questi componenti hanno fatto compiere un ulteriore balzo in avanti nella costruzione dei computer.
Il passo successivo è stato quello di riunire in un solo chip più circuiti integrati, creando il "microprocessore".
Sebbene la tecnologia dei "microchip" sia ben poco rassomigliante alla gigantesca "macchina analitica" di Babbage e Lovelace o al "Colossus" di Turing, l'"architettura" nel progetto dei moderni computer si basa ancora sui principi impostati da Babbage e la teoria enunciata da Turing rimane ancora oggi insuperata.

Lo Spectrum della Sinclair

Sviluppato nemmeno 40 anni dopo i primi giganteschi computer, che occupavano intere stanze, lo Spectrum è piccolo, compatto ed economico. Lo Spectrum è stato il primo personal computer a colori in vendita a meno di 100 sterline (circa 250.000 lire). Questo apparecchio ha avuto un enorme successo



L'"Architetto" dei moderni computer

John von Neumann prese parte al progetto ENIAC come consulente per il progetto logico del sistema. Il suo lavoro ha influenzato enormemente i successivi sviluppi del computer. Suggerì, tra l'altro, di dotare i futuri computer di programmi interni allo stesso apparecchio e di impiegare la numerazione binaria per rappresentare programmi e dati nella memoria dei computer

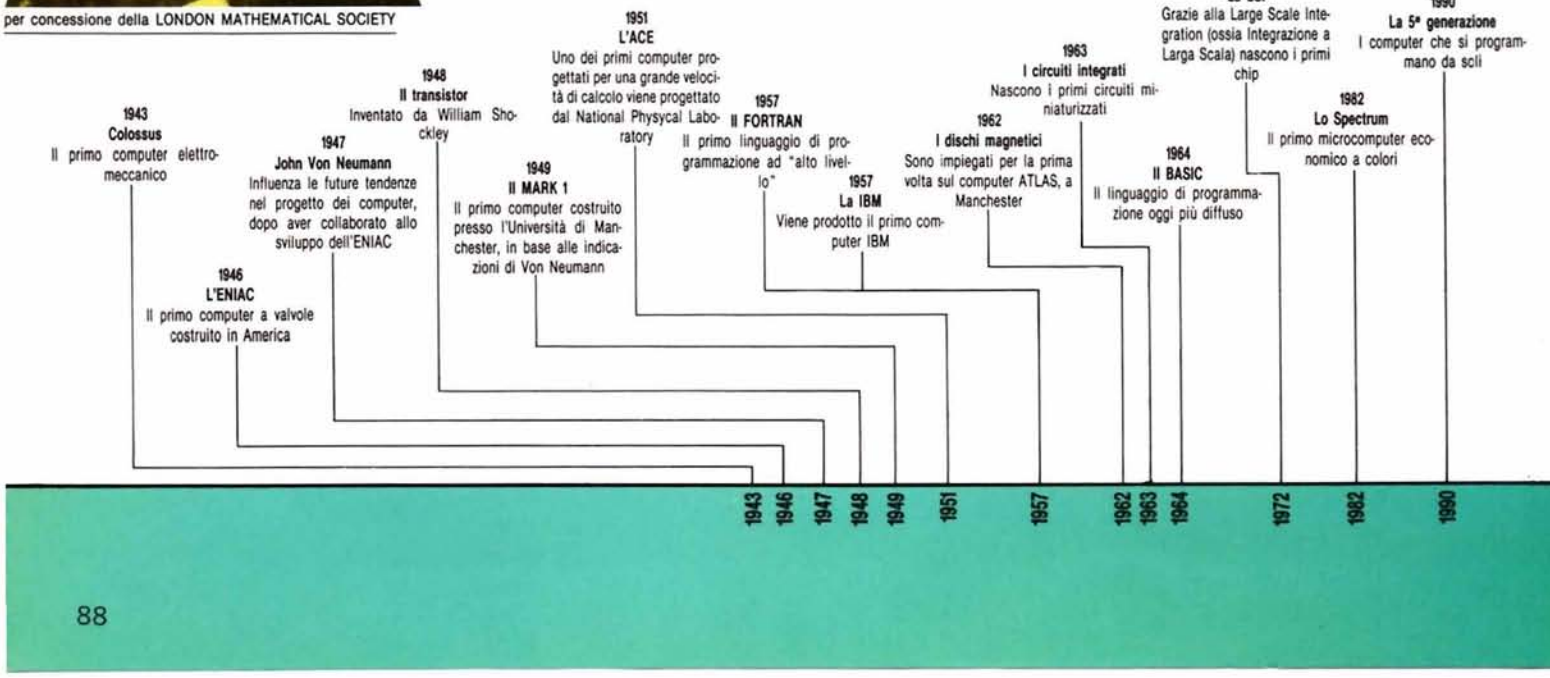


per concessione della LONDON MATHEMATICAL SOCIETY



Steve Wozniac

Nell'industria dei computer, già si parla di lui come di una "leggenda vivente". Nella foto, egli appare con la sua prima invenzione: l'Apple I (oggi un computer da collezionisti). Già a scuola egli progettava computer, e pochi altri come lui hanno contribuito alla miniaturizzazione ed alla diffusione su larga scala del computer. L'Apple II, il cui progetto è del 1976, era il primo "home computer" ad impiegare unità a dischetti e rimane ancora oggi uno dei computer più venduti. Wozniac lavora tutt'oggi a nuovi progetti per la Apple Corporation



Il BBC modello B

Anatomia di microcomputer di diffusione europea: tecnologie a confronto

Il BBC Modello B è costruito dalla Acorn Computers di Cambridge (GB) ed è commercializzato dall'ente televisivo di Stato, la BBC. Il precedente Modello A, più economico, ma molto meno sofisticato, è lentamente uscito di produzione.
In Gran Bretagna, questo apparecchio ha avuto grandissima diffusione nell'ambito scolastico, (poiché esso figura nell'elenco dei computer per la didattica, stilato dal governo). È logico, quindi, che vi sia un'ampia scelta di software didattico, che, presto, sarà disponibile anche in Italia.
Nonostante siano stati prodotti, nel frattempo, molti altri tipi di computer, le caratteristiche tecniche del BBC Modello B sono ancora considerate di ottima qualità.
In particolare, il BASIC del BBC è ricco di comandi per la gestione di speciali funzioni. Anche la stesura e la correzione dei programmi avviene in modo molto facile.
Sono disponibili ben 8 diversi "modi grafici", offrendo all'utente la scelta tra bassa, media ed alta risoluzione, (anche se in quest'ultima il numero di colori disponibili è limitato). La risoluzione massima è di 640×256 punti. Nella maggior parte delle applicazioni a livello domestico, il BBC viene collegato ad un normale televisore. Ovviamente si ottengono risultati grafici migliori impiegando un apposito monitor. Esistono comandi per disegnare linee, cerchi e per creare una vasta selezione di forme ed immagini sullo schermo.
I circuiti di alimentazione sono posti all'interno del contenitore, esteticamente molto elegante. I connettori per periferiche, posti nella parte posteriore e inferiore dell'apparecchio, sono più numerosi che in altri apparecchi simili e permettono di allacciare un vasto numero di diversi apparecchi (per esempio, anche unità a dischetti diverse da quelle fornite dalla BBC).
Oltre alle interfacce per unità a dischi, per la stampante e per apparecchi di misura analogici, c'è la possibilità di allacciarsi ad alcuni "network" (reti di computer). Questa particolare caratteristica è molto utile in applicazioni scolastiche, in quanto permette, per esempio, di usare una sola unità a disco o una sola stampante, per più computer.
È disponibile, infine, una sofisticata, anche se non molto diffusa, interfaccia accessoria (il "Tube"), che permette di usare un diverso tipo di microprocessore, per raggiungere maggiore velocità di esecuzione o per usare programmi scritti per altri computer.



LA TASTIERA DEL BBC MODELLO B

La disposizione dei tasti e la qualità della tastiera sono molto apprezzabili. Ogni tasto è sagomato per facilitare la battitura, che ha un "tocco" comodo anche a chi è abituato alla tastiera di una macchina da scrivere.
Quattro tasti speciali servono a spostare il "cursore" sullo schermo, facilitando la correzione di testi sul video.
I tasti-funzione, colorati di rosso, sono disposti nella parte alta della tastiera. Essi possono essere impiegati da programmi didattici o applicativi.
Una caratteristica molto simpatica è la presenza di tre diodi LED che, accendendosi, indicano se è in funzione il motore del registratore a nastri, se è attivo il tasto "Shift lock" (blocco delle maiuscole), o quello "Capitals Lock" (simile al precedente ma limitato alle lettere e non ai simboli)



Il "drive" per i dischi

Il drive per dischetti della BBC è un accessorio molto utile, ma il cui costo è piuttosto alto, se paragonato ad altre marche. La capacità dei dischetti è di soli 100 Kbyte. È possibile collegare anche unità a dischi di marca diversa dalla BBC

Ingresso analogico

A questo connettore è possibile collegare apparecchi di misura analogici (una sonda termica, per esempio), che permettono l'immissione diretta delle letture

Chip del video

Questo circuito trasforma le informazioni contenute nella memoria in caratteri visualizzabili sullo schermo

Chip per interfaccia

Per la connessione ad altri apparecchi vengono impiegati speciali chip, come questo 6522 della MOS Technology. Sebbene non siano veri e propri microprocessori, questi chip sono altrettanto complessi internamente



I Connettori per le periferiche

Una veduta della parte inferiore del computer, dove sono alloggiate alcune prese per gli apparecchi accessori

Porta per stampante

Qui si collega una stampante con interfaccia di tipo "parallelo"

Porta generica

Questa porta d'ingresso/uscita può essere usata dall'utente per collegare altre apparecchiature

IAN McKINNELL

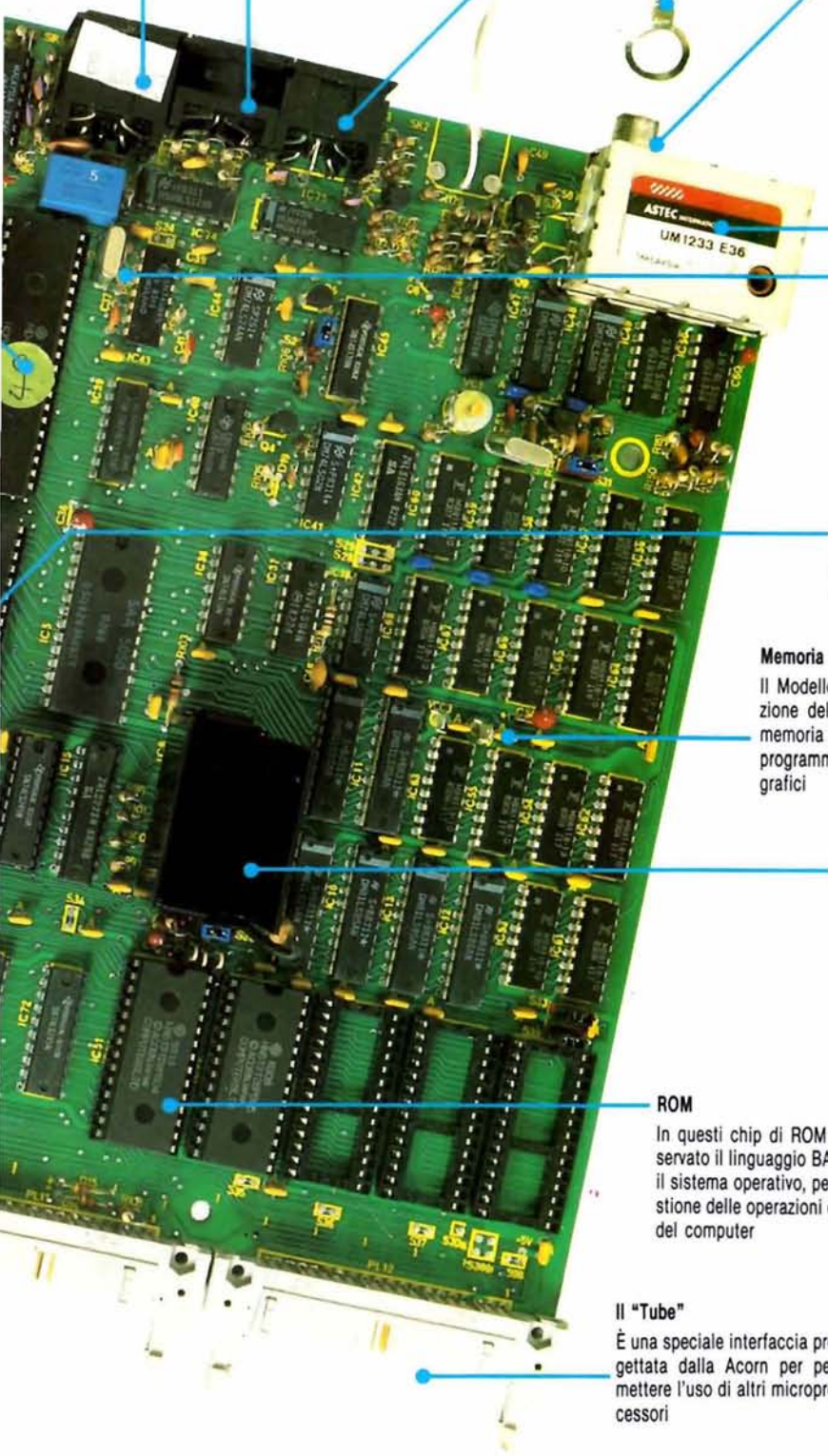
Interfaccia per registratore
I programmi possono venir conservati usando un normale registratore "a cassette" scegliendo tra due velocità: una rapida e un'altra più lenta ma più affidabile

Uscita RGB
Connettore per allacciarsi ad un monitor a colori. I tre segnali di colore (rosso, verde e blu) vengono trasmessi separatamente

Uscita per televisore
Per l'impiego con un normale televisore

Uscita video
Per l'uso di un monitor monocromo

Porta RS232
Questo è il connettore della porta seriale di trasmissione



Modulatore RF
Questo circuito converte i segnali prodotti dal computer in segnali adatti ad un televisore

Cristallo al quarzo
Questo componente serve al funzionamento del temporizzatore, che sincronizza tutte le operazioni del computer

Microprocessore
Quello impiegato nel BBC è il 6502 della MOS Technology

Memoria utente
Il Modello B mette a disposizione dell'utente 32 Kbyte di memoria RAM, per conservare programmi, dati e caratteri grafici

ULA (Uncommitted Logic Array ossia Matrice Logica Indipendente)
Questo speciale circuito, progettato appositamente, svolge una massa di operazioni per le quali, in altri computer, sono necessari più componenti. È dotato di un'aletta di raffreddamento per dissipare il calore prodotto nel funzionamento

ROM
In questi chip di ROM è conservato il linguaggio BASIC ed il sistema operativo, per la gestione delle operazioni di base del computer

Il "Tube"
È una speciale interfaccia progettata dalla Acorn per permettere l'uso di altri microprocessori

BBC Modello B

DIMENSIONI
mm. 409 x 358 x 78

PESO
3700 grammi

CPU
6502

CLOCK
2 MHz

MEMORIA
32 Kbyte di RAM e 32 Kbyte di ROM (che contengono il BASIC ed un sofisticato sistema operativo)

VIDEO
8 diversi modi grafici consentono un'ampia scelta. Massima area per testi: 32 linee di 80 caratteri. Massima risoluzione grafica: 640 x 256 punti, 16 colori disponibili in bassa risoluzione

INTERFACCE
Televisore, monitor monocromo o a colori, dischi, stampante, periferiche varie, oltre al "Tube" (vedi testo)

LINGUAGGI FORNITI
BASIC

ALTRI LINGUAGGI DISPONIBILI
FORTH, LISP, LOGO

ACCESSORI COMPRESI
Cavi di connessione per TV e per registratore a cassette, un nastro con programma dimostrativo, un manuale

TASTIERA
Tipo convenzionale con 64 tasti + 10 tasti "funzione"

DOCUMENTAZIONE
La guida al Microcomputer BBC è stata scritta dando per scontata una certa conoscenza di base del funzionamento del computer. Molti capitoli trattano in modo piuttosto sofisticato le capacità grafiche e gli effetti sonori dell'apparecchio
È inclusa anche una panoramica piuttosto completa ed esauriente (a differenza dei manuali di altri computer) sulla programmazione del chip 6502

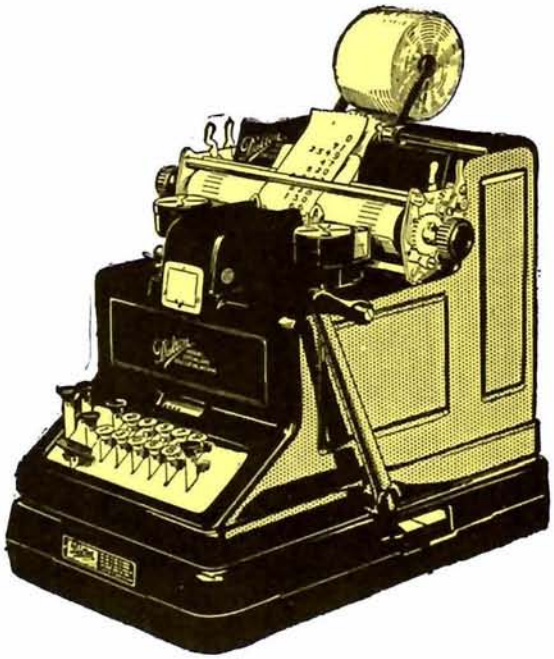
Le porte logiche e i sommatori

I circuiti elettronici per la somma dei numeri binari

Abbiamo visto precedentemente (a pag. 68), come sia relativamente semplice usare dei transistor per operare scelte logiche del tipo AND, OR e NOT. Questi stessi circuiti vengono impiegati nel microprocessore anche per le operazioni aritmetiche. Nei circuiti logici, l'ingresso può assumere i valori 1 o 0, per indicare "vero" o "falso". In genere l'assenza di tensione si associa a 0, e una tensione positiva ad 1. Nelle operazioni aritmetiche, questi valori rappresentano numeri, espressi in forma binaria.
Volendo sommare due cifre binarie, sono sufficienti due ingressi e esistono soltanto quattro combinazioni possibili (0 + 0, 0 + 1, 1 + 0, 1 + 1). Dai nostri studi sui numeri binari, abbiamo imparato che, come nel caso dei decimali, 0 + 0 = 0. E, ancora, che 1 + 0 (o 0 + 1) = 1. La differenza con l'aritmetica tradizionale sorge nel caso 1 + 1, dove il risultato è 10, ossia 0 con riporto (Carry) 1. Riassumiamo tutto ciò in una tabella:

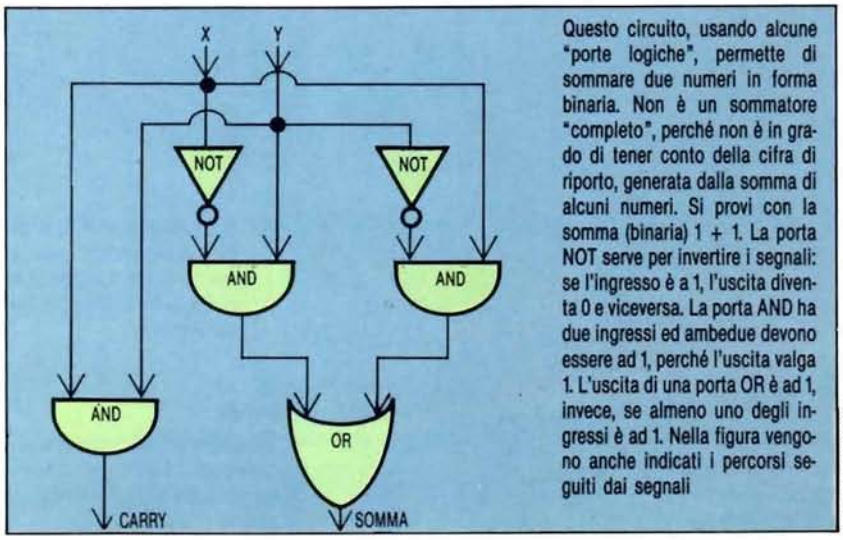
X		Y		Z
0	+	0	=	0
0	+	1	=	1
1	+	0	=	1
1	+	1	=	10

Usando una porta logica OR, otterremo una uscita "falsa" (0) se ambedue gli ingressi sono "falsi" (0) ed un'uscita "vera" (1) se uno dei due ingressi è "vero" (1 + 0 oppure 0 + 1).
Purtroppo, la soluzione non è così semplice! Infatti, se ambedue gli ingressi sono "veri" l'uscita di una porta OR risulta "vera", il che porterebbe ad un risultato aritmetico errato. L'uscita dovrebbe dare 0, con riporto di 1.
Ci serve, dunque, un circuito che ci dia un'uscita 0 se ambedue gli ingressi sono a 0, un'uscita 1 se uno dei due ingressi è a 1, ed un'uscita 0 se ambedue gli ingressi sono a 1. Se usiamo due porte AND collegate come nell'illustrazione, abbiamo quasi risolto il nostro problema. "Quasi", perché se i due ingressi sono a 1, l'uscita è 0, ma non viene generato il riporto. Aggiungendo la porta AND (quella più a sinistra nella figura) si ottiene



La calcolatrice da tavolo
Le calcolatrici meccaniche erano, fino a pochi anni fa, molto diffuse nei negozi e negli uffici. Salvo qualche modifica di poco conto, il principio di funzionamento di queste macchine è rimasto immutato per 300 anni, basandosi sull'impiego di ingranaggi e leve. Il potenziale commerciale di un apparecchio in grado di eseguire calcoli venne in parte intuito da Pascal. Egli costruì la prima addizionatrice meccanica, che venne in seguito usata da suo padre, esattore di tasse. Quando, successivamente, Leibnitz trovò la maniera per eseguire anche le moltiplicazioni e le divisioni, l'apparecchio era pronto per le applicazioni commerciali

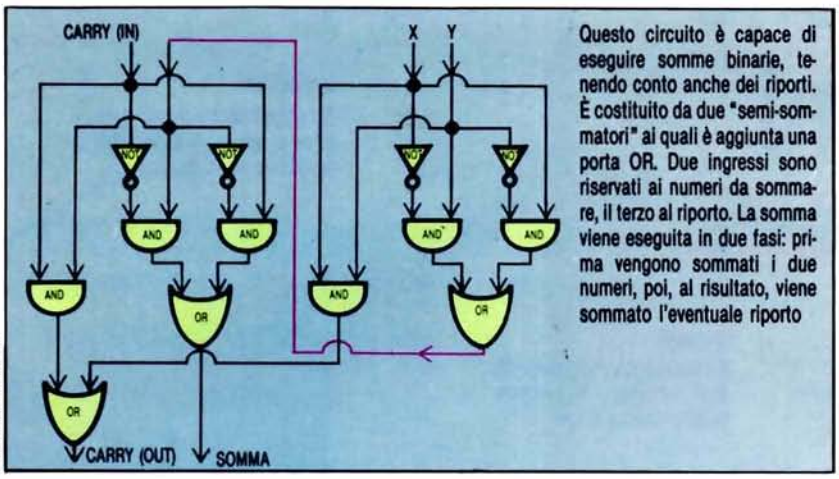
SEMI-SOMMATORE



Questo circuito, usando alcune "porte logiche", permette di sommare due numeri in forma binaria. Non è un sommatore "completo", perché non è in grado di tener conto della cifra di riporto, generata dalla somma di alcuni numeri. Si provi con la somma (binaria) 1 + 1. La porta NOT serve per invertire i segnali: se l'ingresso è a 1, l'uscita diventa 0 e viceversa. La porta AND ha due ingressi ed ambedue devono essere ad 1, perché l'uscita valga 1. L'uscita di una porta OR è ad 1, invece, se almeno uno degli ingressi è ad 1. Nella figura vengono anche indicati i percorsi seguiti dai segnali

MARK WATKINSON

SOMMATORE "COMPLETO"



Questo circuito è capace di eseguire somme binarie, tenendo conto anche dei riporti. È costituito da due "semi-sommatori" ai quali è aggiunta una porta OR. Due ingressi sono riservati ai numeri da sommare, il terzo al riporto. La somma viene eseguita in due fasi: prima vengono sommati i due numeri, poi, al risultato, viene sommato l'eventuale riporto

MARK WATKINSON

finalmente un circuito chiamato "semi-sommatore" (half adder), la cui tavola della verità risulta essere:

X (ingresso 1)	Y (ingresso 2)	C (uscita "carry")	S (uscita "somma")
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Nelle normali applicazioni, occorre sommare almeno un byte (= 8 bit) alla volta, mentre il nostro "semi-sommatore" ne somma soltanto due. Per ottenere un circuito "sommatore completo" (full adder) possiamo impiegare il "semi-sommatore", che per la colonna di numeri più a destra funziona ancora bene, ma ci serve ancora qualche modifica. Tutti i sommatori a sinistra del primo devono poter accettare tre ingressi (i due "normali", più il riporto della colonna precedente).
Facciamo un esempio:

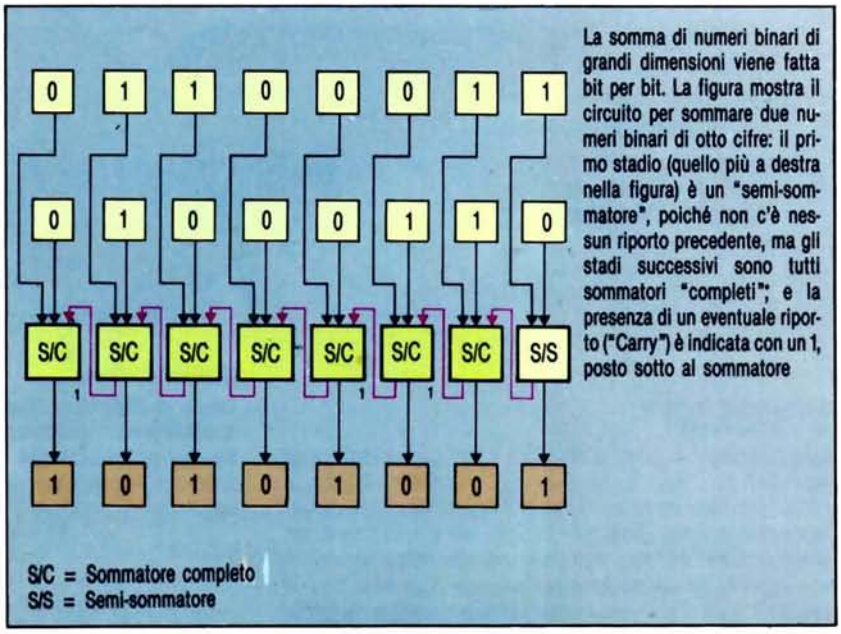
$$\begin{array}{r} 011 + \\ 111 \\ \hline 1010 \end{array}$$

Si somma la colonna più a destra: 1 + 1 = 0, riporto 1, poi la seconda: 1 + 1 = 0, con riporto 1. Sommando il riporto precedente al risultato: 1 + 0 = 1 con riporto 1. Nella terza colonna: 0 + 1 = 1. Si somma il riporto precedente: 1 + 1 = 0 con riporto 1, che scriviamo sulla quarta colonna.
In altre parole, la tavola della verità di un "sommatore completo", in grado di operare sui riporti, oltre che sui due numeri in ingresso è:

X	Y	C(in)	C(out)	S
0	0	0	0	0
1	0	0	0	1
0	1	0	0	1
0	0	1	0	1
0	1	1	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

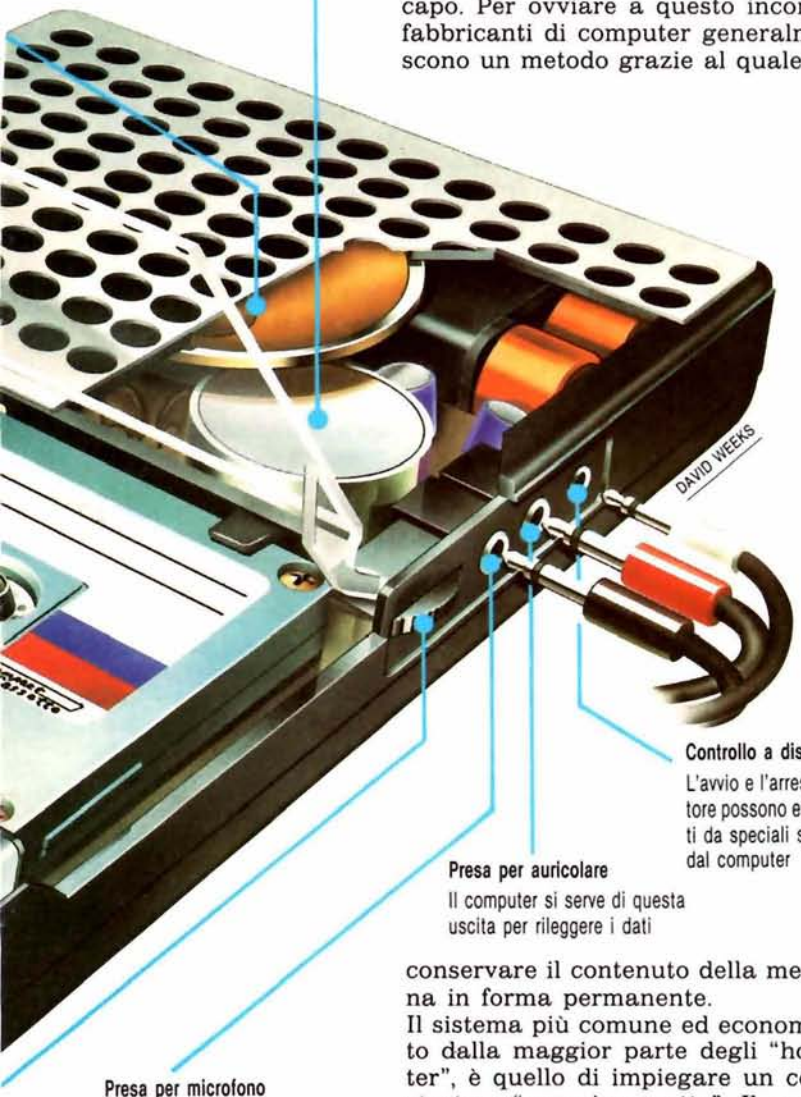
Un circuito "sommatore completo" si ottiene usando due "semi-sommatori" più una porta OR", come illustrato in figura. Il riporto in uscita "C (out)" viene collegato al riporto in ingresso "C (in)" del sommatore alla sua sinistra. Anche più sommatori completi possono essere collegati in questo modo.
Nei microcomputer, la maggior parte delle operazioni aritmetiche vengono eseguite impiegando circuiti molto simili a quelli che abbiamo appena descritti. Generalmente, questi circuiti costituiscono una parte di quelli interni all'unità centrale di processo (la CPU). Quando non esisteva ancora la tecnica dell'integrazione a larga scala, i più semplici circuiti integrati contenevano soltanto qualche porta logica. Questi erano detti integratori TTL (la sigla proviene da Transistor-Transistor Logic, logica da Transistor a Transistor).
All'interno di una CPU si trovano da due piccole zone di memoria (una di RAM ed una di ROM) a un grande numero di circuiti logici indicati con la sigla ALU (Arithmetic and Logic Unit - Unità Aritmetica e Logica). Il computer esegue tutti i confronti logici e le operazioni aritmetiche grazie ai circuiti dell'ALU.

SOMMATORE A 8 BIT



La somma di numeri binari di grandi dimensioni viene fatta bit per bit. La figura mostra il circuito per sommare due numeri binari di otto cifre: il primo stadio (quello più a destra nella figura) è un "semi-sommatore", poiché non c'è nessun riporto precedente, ma gli stadi successivi sono tutti sommatori "completi"; e la presenza di un eventuale riporto ("Carry") è indicata con un 1, posto sotto al sommatore

Motore
Il motore imprime una rotazione costante al rullo di trascinamento e provvede anche alle funzioni di avvolgimento e riavvolgimento veloci



Controllo a distanza
L'avvio e l'arresto del registratore possono essere comandati da speciali segnali generati dal computer

Presa per auricolare
Il computer si serve di questa uscita per rileggere i dati

Presa per microfono
Viene usata per immettere i segnali di registrazione provenienti dal computer. In alcuni apparecchi esiste una presa di tipo DIN per ingressi ausiliari, che andrebbe preferita a questa. È necessaria un'attenta regolazione dei comandi di volume e dei toni (se presenti) per ottenere buone registrazioni

Quando si scrive un programma, sia esso tratto da un libro, da una rivista o dalla nostra inventiva, le istruzioni di volta in volta digitate vengono conservate nella memoria RAM. Finché il computer è acceso, le informazioni rimangono in memoria, ma se dovesse mancare la corrente o se spegnessimo l'apparecchio, tutto il programma verrebbe cancellato. Ciò può essere particolarmente spiacevole, soprattutto quando il programma è di una certa lunghezza. Volendo riusare il programma, dovremmo riscriverlo tutto da capo. Per ovviare a questo inconveniente, i fabbricanti di computer generalmente forniscono un metodo grazie al quale è possibile conservare il contenuto della memoria interna in forma permanente.

Il sistema più comune ed economico, adottato dalla maggior parte degli "home computer", è quello di impiegare un comune registratore "mangiacassette". Il metodo impiegato per trasferire le informazioni tra memoria e nastro magnetico varia a seconda della marca di computer, tanto che un programma creato ed archiviato da un Commodore non può essere riletto da uno Spectrum e viceversa. Il principio di funzionamento, comunque, è abbastanza simile in tutti gli apparecchi.

Anche se viene impiegato un normale registratore, i segnali incisi non sono certamente di genere musicale, ma rappresentano, invece, numeri binari, ovvero sequenze di segnali a livello alto o basso, per indicare, rispettivamente, gli 1 e gli 0.

In genere, si scelgono due frequenze fisse: 2.400 Hz per indicare il livello 1, e 1.200 Hz per il livello 0.

Quando impartiamo il comando SAVE, per archiviare un programma, la prima cosa a venir registrata su nastro è una più o meno lunga sequenza di segnali a frequenza fissa. Questa serve, in successive riletture da nastro, per sincronizzare il computer con la velocità di registrazione. Successivamente, viene incisa una serie di modulazioni, corrispondenti al nome da noi assegnato al programma. Ogni lettera del nome occupa un byte (otto bit), pertanto vengono generati otto segnali per ciascun carattere. In più, perché il computer possa riconoscere l'inizio e la fine di una sequenza di otto bit, vengono aggiunti altri due segnali, chiamati, rispettivamente, "bit di start" e "bit di stop".

Il contenuto del programma viene registrato col medesimo sistema, anche se, in genere, viene spezzettato in blocchi di 256 byte. Spesso vengono anche registrate altre informazioni necessarie ad una corretta rilettura del programma. Una di queste informazioni è la "checksum" (somma di controllo). Il primo byte di un blocco contiene la lunghezza, in byte, del blocco, mentre l'ultimo byte contiene uno speciale valore calcolato in base alla somma del valore di ciascun byte del blocco. In fase di lettura, il computer effettua una verifica su questi valori e, in caso di errori, avverte l'operatore che la lettura è fallita. Il livello di sofisticazione nel metodo di controllo si traduce in una maggiore affidabilità dell'intero sistema.

I Baud

La velocità alla quale vengono registrate le informazioni su nastro è normalmente (ed erroneamente) indicata col termine "Baud", che deriva dal codice Baudot, usato in telegrafia. Il termine, in effetti, sta ad indicare quante volte un segnale cambia di livello nell'intervallo di un secondo. Più alta è la velocità (in genere è compresa tra 300 e 1.200 bit al secondo) più rapide saranno la scrittura e la lettura su nastro.

Purtroppo, più alta è la velocità, maggiori sono le probabilità che vengano introdotti errori. Trasferimenti a 1.200 bit al secondo offrono un buon compromesso tra velocità e sicurezza. In alcuni computer è possibile selezionare tra due velocità di trasferimento, lasciando la scelta all'utente.

La qualità delle "cassette", ovviamente, deve essere tale da offrire le massime garanzie. Si possono usare anche normali nastri musicali, ma non conviene sceglierli di durata maggiore ai C-60. Per calcolare approssimativamente la capacità di un nastro, si divide per 10 il valore della velocità di trasferimento, ottenendo così il numero di byte registrati al secondo. Si moltiplica poi questo valore per la durata, in secondi, del nastro. Una cassetta C-60 (30 minuti per lato) ad una velocità di trasferimento di 1.200 bit al secondo, offre una capacità totale di circa 432 Kbyte.

La registrazione dei programmi

Come salvare il lavoro su nastro magnetico

Altoparlante
Se il registratore è collegato ad un computer, in genere l'altoparlante viene escluso

Testina di cancellazione
Provvede all'azzeramento di precedenti registrazioni durante la fase di incisione

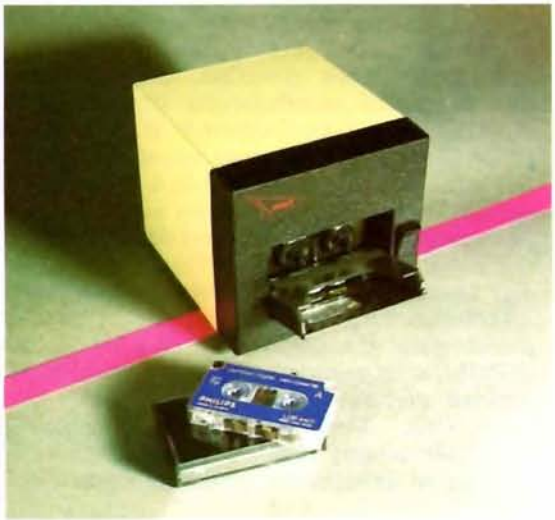
Contagiri
Un accessorio indispensabile se si conservano più programmi sullo stesso nastro

Testina di scrittura/lettura
Questa testina, a seconda del caso, scrive o legge le informazioni



Rullo di trascinamento
Il regolare scorrimento del nastro è assicurato da uno speciale rullo gommato

Controllo di volume
Questo controllo deve essere accuratamente posizionato per ottenere il caricamento di programmi in condizioni ottimali



IAN McKINNELL

L'"HOBBIT"

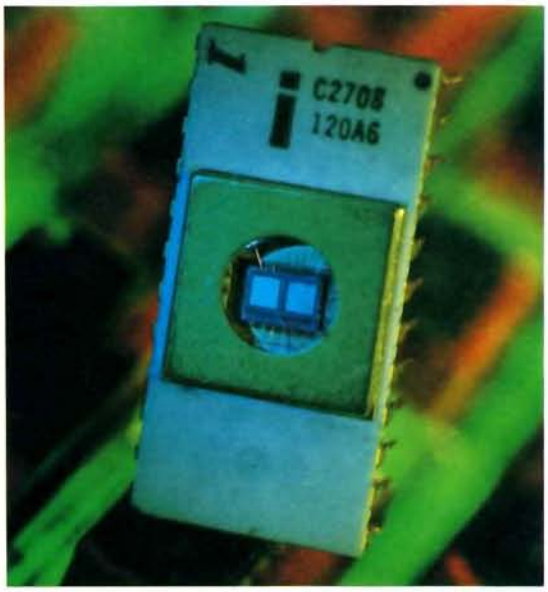
Questo apparecchio, prodotto dalla Ikon, è uno speciale registratore appositamente progettato per l'uso con il computer. Totalmente controllabile tramite software, la qualità di registrazione è notevolmente superiore ai normali registratori musicali. Non vi sono pulsanti per la selezione delle funzioni: è il computer che le controlla tutte (anche il riavvolgimento del nastro). Specificando il nome di un programma da caricare, l'"Hobbit" posiziona automaticamente il nastro per la lettura

I tipi di memoria

Cosa sono e come funzionano le RAM e le ROM

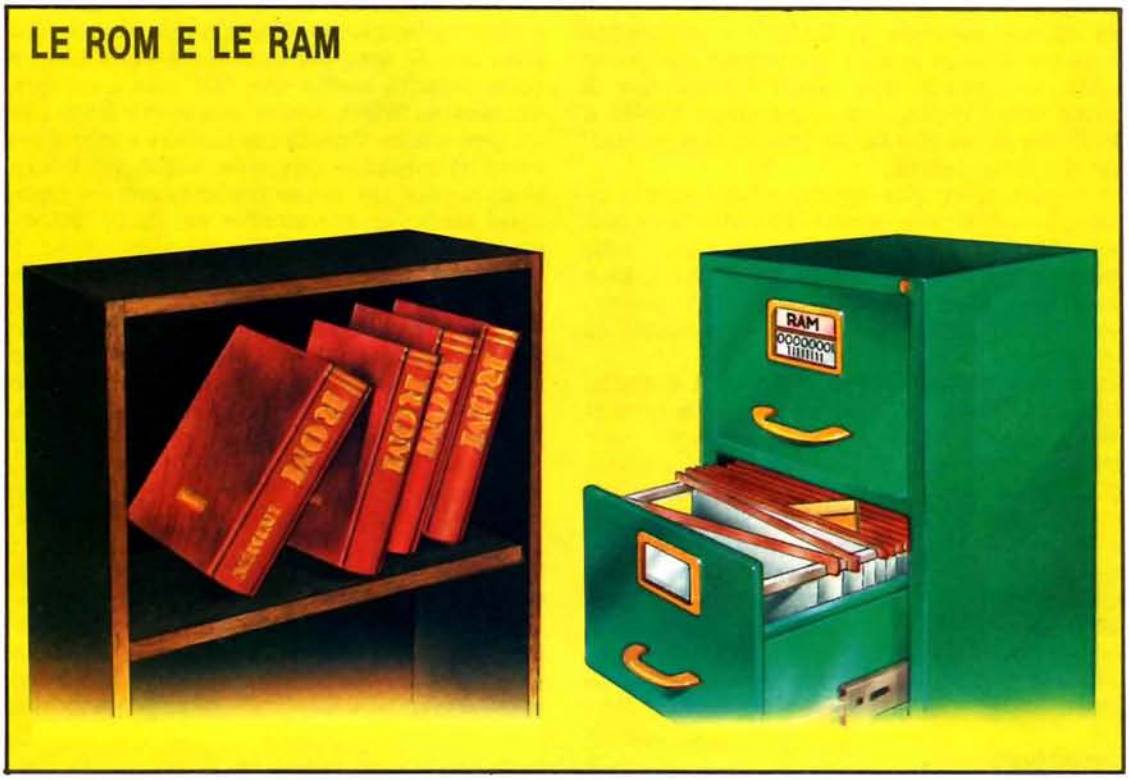
Un modo per descrivere la memoria di un computer è in termini di conservazione "a breve" ed "a lunga scadenza". La memoria "a lunga scadenza" mantiene le informazioni anche quando l'apparecchio viene spento; i dischi ed i nastri magnetici appartengono a quest'ultima categoria.
Ma i computer usano anche una memoria "a breve termine", per la conservazione provvisoria di programmi e dati. Potremmo anche distinguere la memoria tra "interna" ed "esterna" al computer. La memoria "interna" è situata dentro il contenitore dell'apparecchio e, generalmente, è di tipo elettronico, mentre quella "esterna" è parzialmente meccanica (ad esempio: una piastra di registrazione, un "drive" per dischi od un perforatore di schede).
La memoria interna è anche indicata come "principale", e quella esterna come "secondaria" o "di massa". Attualmente, la memoria interna è composta di RAM e di ROM, ambedue memorie completamente elettroniche che si presentano sotto forma di "chip" al silicio racchiuso in un contenitore rettangolare in plastica e dotato di una serie di piedini metallici. Il sistema usato dalla CPU per selezionare questi tipi di memoria è unico, ma torneremo sull'argomento più avanti.
La principale differenza tra RAM e ROM è che quest'ultima viene impiegata per custodire i programmi in modo permanente: i codici vengono incisi, con particolari processi, dal fabbricante di computer e non sono più modificabili. Le ROM si possono paragonare a "biblioteche di consultazione", dalle quali è possibile trarre programmi al momento opportuno, senza però poterli modificare. Il termine ROM proviene dall'inglese "Read Only Memory", ovvero Memoria a Sola Lettura.
Alcuni speciali tipi di ROM, chiamati EPROM, si possono azzerare usando raggi ultravioletti e riprogrammare da capo. Una ROM tipica è la 2364 della Intel. Il chip ha una capacità di 65.536 bit, organizzati come 8 Kbyte di memoria ad 8 bit. In termini matematici, 1 K equivale a 2 elevato alla decima potenza, ossia a 1024: pertanto 64 Kbyte = 64 × 1024 = 65.536.
La CPU può "indirizzare" (selezionare) uno qualsiasi degli 8.192 byte. Dallo schema elettrico della 2364 si può vedere che dei 28 piedini di connessione, 2 servono per l'alimentazione (+5 volt e massa). Dei rimanenti 26 otto sono utilizzati per trasferire i valori degli otto bit di una particolare locazione alla CPU ("bus dei dati"); uno non è utilizzato (esiste soltanto per ragioni di fabbricazione); quattro piedini servono per selezionare il funzionamento della memoria: un "pin" (piedino) serve per abilitare l'uscita ("output enable"), uno per abilitare il chip ("chip enable"), e due per selezionare il chip ("chip select"). Questi due "pin" raccolgono segnali dalla CPU per sapere quando è richiesto l'impiego del chip.
I restanti 13 pin, che vengono usati per indirizzare una particolare locazione di memoria, sono collegati al "bus degli indirizzi" del CPU. Poiché i piedini sono 13, sono possibili 2^{13} = 8.192 diverse combinazioni, permettendo di selezionare ciascuno degli 8.192 byte della ROM.

LE EPROM



Le normali memorie a sola lettura (ROM) vengono programmate a livello di fabbricazione, dopo di che è impossibile modificarne il contenuto. Nelle EPROM (Erasable Programmable Read Only Memory ossia Memorie a sola lettura programmabili cancellabili), invece, il contenuto si può cancellare, usando raggi ultravioletti. Ciò permette di riprogrammarle anche più volte. Sull'involucro delle EPROM c'è una piccola finestrella, che permette di esporre i circuiti interni all'esposizione di raggi ultravioletti. Evitando l'esposizione, il contenuto della memoria è conservato indefinitamente

Le RAM, al contrario, si possono paragonare a delle lavagne, dove si possono depositare dei dati (o programmi), cancellarli e riscriverne dei nuovi.
La struttura delle RAM è più complessa ancora di quella delle ROM, perché ogni singolo bit deve essere modificabile.
Un chip RAM abbastanza diffuso è la memoria 2114 della Intel. Ogni 2114 ha una capacità di 4.096 bit, organizzati in 1024 gruppi di quattro bit. Per ottenere 1 Kbyte di RAM sono quindi necessari due di questi chip. I piedini di collegamento sono soltanto 18, due dei quali usati per l'alimentazione. Quattro servono per l'ingresso/uscita dei dati. Uno è usato per il segnale di selezione (per far sapere al chip quando è richiesto il suo intervento) ed uno per indicare al chip se è richiesta una lettura o una scrittura. Le rimanenti 10 linee ("bus degli indirizzi") servono per la selezione delle locazioni di memoria. Volendo impiegare le 2114 in un computer per ottenere 64 Kbyte di memoria utente, sono necessari 128 chip. Esistono anche chip di RAM a più alta densità, come le memorie 4164: ne bastano 8 per ottenere 64 Kbyte di memoria utente.
Sia le RAM che le ROM diventano, col passare del tempo, sempre più economiche e compatte: siamo già arrivati a chip con 128 Kbit di capacità. Ma siamo ai limiti delle possibilità consentite dalla tecnica odierna. Probabilmente i chip del futuro saranno costruiti impiegando metodi basati su raggi elettronici oppure raggi X.
Vi sono attualmente due tipi di memoria RAM, conosciuti come RAM "dinamica" e RAM "statica". Ambedue i tipi hanno vantaggi e svantaggi, anche se oggigiorno viene maggiormente impiegata quella di tipo "dinamico". Ambedue i tipi perdono il loro contenuto se viene tolta l'alimentazione, ma la memoria "dinamica" ha bisogno di essere "rinfrescata" elettricamente ogni pochi millisecondi. Questo processo deve avvenire senza interferire con le operazioni della CPU. Ciò comporta il progetto di accurati circuiti di sincronizzazione.
La memoria dinamica ha due grossi pregi rispetto a quella statica: essa richiede l'impiego di un solo transistor per ogni bit (per la statica ne sono necessari tre), permettendo di concentrare più memoria in un singolo chip. L'altro vantaggio è il minore consumo di corrente, che consente di impiegare alimentatori più economici.
Il vantaggio principale delle memorie statiche, invece, consiste soprattutto nel fatto che non è necessario "rinfrescare" questo tipo di memoria e ciò si riflette in una maggiore semplicità dei circuiti in cui esse vengono applicate.



Memoria "a lungo termine"

La ROM può essere paragonata ad un libro. Così come non è possibile modificare il testo del libro, non si possono alterare le informazioni in essa contenute

Memoria a "breve termine"

La memoria a scrittura/lettura è paragonabile a un archivio di pratiche. Queste, infatti, possono essere aggiornate e modificate in qualsiasi momento, proprio come è possibile scrivere, cancellare e poi riscrivere sulla RAM. I dati in essa memorizzati non sono permanenti e vengono cancellati quando si spegne il computer

Quanto manca a Natale?

L'uso delle "stringhe" ed il calcolo del numero di giorni tra due date

In questa lezione viene presentato un programma, che serve un poco da riepilogo degli argomenti fin qui trattati e, al tempo stesso, introduce l'uso di alcune nuove ed utili istruzioni del BASIC.
Il programma serve per calcolare quanti giorni mancano a Natale.
Il listato inizia con un elenco delle variabili usate. Anche se non è particolarmente essenziale, la pratica di elencare le variabili rende più chiaro il funzionamento del programma, specialmente se si desidera modificarlo in tempi successivi.
In alcune versioni di BASIC, è consentito assegnare nomi di sole due lettere, mentre in altre non esiste una simile limitazione. È ovvio che l'impiego di nomi come GIORNO o MESE (invece di GI o ME) facilita la comprensione del programma.
Il programma, per prima cosa, esegue le istruzioni PRINT che cominciano alla linea 230, visualizzando una breve spiegazione sullo scopo del programma ed invitando ad immettere una data scritta secondo un preciso formato (giorno, mese e anno separati da virgole).
La prima istruzione per noi nuova è quella alla linea 300: la DIM (che sta per DIMensione). Nel nostro caso, la X è una variabile "con indice", ossia essa può contenere più elementi, ciascuno dei quali è individuato dal numero dell'indice.
L'istruzione DIM serve per assegnare una dimensione massima al numero di elementi della variabile X. In altre parole, nella linea 300, noi creiamo una variabile chiamata X, composta da 13 elementi.
Più avanti nel corso spiegheremo approfonditamente sia l'istruzione DIM che l'uso delle variabili "con indice" (in inglese "Subscripted variables").

```
310 INPUT G,M$,A
```

Questa linea contiene un normale INPUT, sennonché i valori da immettere sono tre: G ed A sono variabili numeriche che conterranno, rispettivamente, il giorno e l'anno, mentre M$, che contiene il mese, è una variabile un po' speciale, chiamata "stringa" (il simbolo "$", subito dopo la M, serve proprio per questa distinzione). In una variabile "stringa" si possono conservare sia lettere che numeri.
Facciamo un esempio: supponiamo di digitare alla tastiera 23,GENNAIO,1984. Alla variabile G viene assegnato il valore numerico 23, alla variabile M$ viene assegnata la stringa GENNAIO, mentre alla variabile A il valore numerico 1984.

```
330 GOSUB 560     REM CALCOLO DEL MESE
```

Questa istruzione serve per far eseguire il sottoprogramma (o subroutine) che inizia alla linea 560. Si noti, per inciso, che nella linea è stata inserita anche una REM: non è sempre necessario, infatti, usare una nuova linea per un commento. Questa particolare subroutine viene richiamata una sola volta nell'intero programma, per cui le poche istruzioni potevano anche essere inserite nel corpo principale del programma. Il sottoprogramma è stato creato per un motivo molto semplice: originariamente, era previsto un valore numerico anche per il mese e le istruzioni per convertire il nome del mese in un valore numerico non erano necessarie. Poi, avendo deciso di apportare questa modifica, è bastato inserire opportunamente una GOSUB nel programma originale.
È molto facile costruire piccoli "moduli" indipendenti, all'interno di un programma, usando la GOSUB e la RETURN.
La subroutine è molto semplice, ma illustra efficacemente l'elasticità del linguaggio BASIC nel trattamento delle stringhe alfanumeriche. Supponiamo di aver digitato, come mese, GENNAIO. Questa sequenza di caratteri viene assegnata alla variabile M$. La prima linea della subroutine è:

```
560 IF M$ = "GENNAIO" THEN LET M = 1
```

Il contenuto di M$ viene confrontato con i caratteri racchiusi tra doppi apici e se, come nel nostro esempio, le due parole sono identiche, il programma assegna alla variabile M il valore 1. Non si confondano tra loro le due variabili M$ ed M: M$ contiene una stringa, mentre M è una variabile numerica. Nelle

linee successive il contenuto di M$ viene via via confrontato con i nomi dei restanti mesi, ma, non trovando più alcuna corrispondenza, il valore di M rimane inalterato. In pratica, la subroutine serve per assegnare ad M il valore numerico del mese da noi immesso.
Il BASIC, giunto alla linea 680, incontra una RETURN e l'esecuzione riprende dalla linea successiva alla GOSUB, cioè dalla linea 340, dove, se M vale 0 (abbiamo sbagliato a digitare il mese) si riparte dalla linea 230.
Le linee dalla 350 alla 370, incluse, formano un ciclo FOR-NEXT, nel quale viene incrementato, di un'unità per volta, il valore della variabile I. La variabile I, che assume quindi tutti i valori da 1 a 13, serve da indice per la variabile X nella linea 360:

```
360 READ X(I)
```

In questa linea incontriamo per la prima volta l'istruzione READ. La READ è sempre usata insieme con una corrispondente istruzione DATA, che noi troviamo alla linea 510:

```
510 DATA 31,28,31,30,31,30,31,31,30,31,30,25,0
```

Questi valori, eccetto gli ultimi due, indicano il numero di giorni in ciascun mese dell'anno. Lo stesso effetto ottenuto con le due istruzioni READ e DATA, inserite nel ciclo FOR-NEXT, si potrebbe avere usando 13 linee di LET:

```
LET X(1) = 31
LET X(2) = 28
LET X(3) = 31
LET X(4) = 30
LET X(5) = 31
LET X(6) = 30
LET X(7) = 31
LET X(8) = 31
LET X(9) = 30
LET X(10) = 31
LET X(11) = 30
LET X(12) = 25
LET X(13) = 0
```

Per meglio comprendere la coppia READ/DATA, facciamo un altro esempio:

```
10 READ A,B,C
20 LET D = A + B + C
30 PRINT D
40 DATA 5,10,20
```

In questo esempio, la READ nella linea 10 legge il primo dato contenuto nella linea 40, dove abbiamo l'istruzione DATA, e ne copia il valore nella variabile A. Poi viene letto il secondo valore, che viene assegnato alla variabile B ed infine il terzo valore, che viene assegnato alla variabile C. La linea 20 assegna a D il valore della somma di A, B e C. Il risultato è visualizzato dalla linea 30.
Ma torniamo al nostro programma "natalizio". La prima volta che viene eseguito il ciclo, I vale 1. La linea 360, quindi, equivale a READ X(1). Il dato corrispondente nella linea 510 è 31, di conseguenza X(1) = 31.

```
100 REM  *** ELENCO DELLE VARIABILI ***
110 REM
120 REM   G    = CONTIENE IL GIORNO
130 REM   M$   = NOME DEL MESE
140 REM   A    = ANNO
150 REM   I    = INDICE 1
160 REM   X( ) = (CON INDICE) NUMERO DI GIORNI NEI MESI
170 REM   R    = NUMERO DI GIORNI RESTANTI (RISULTATO)
180 REM   M    = NUMERO DEL MESE
190 REM   L    = INDICE 2
200 REM   Z    = VALORE INTERO DI (A/4)
210 REM
220 REM
230 PRINT "QUESTO PROGRAMMMA CALCOLA IL"
240 PRINT "NUMERO DI GIORNI CHE MANCANO"
250 PRINT "A NATALE, PER DATE FINO AL"
255 PRINT "24 DICEMBRE."
260 PRINT
270 PRINT "DIGITARE LA DATA NEL FORMATO GIORNO,MESE,ANNO"
280 PRINT "PER ESEMPIO: 12,LUGLIO,1984"
290 PRINT
300 DIM X(13)
310 INPUT G,M$,A
320 REM
330 GOSUB 560   REM CALCOLO DEL MESE
340 IF M = 0 THEN GOTO 230   REM IN CASO DI MESE ERRATO
350 FOR I = 1 TO 13
360 READ X(I)
370 NEXT I
380 REM
390 GOSUB 750   REM VERIFICA ANNO BISESTILE
400 REM
410 LET R = X(M) - G
420 FOR L = M TO 12
430 LET M = M + 1
440 LET R = R + X(M)
450 NEXT L
460 REM
470 IF R = 1 THEN GOTO 500
480 PRINT "MANCANO ";R;" GIORNI A NATALE"
490 GOTO 520
500 PRINT "MANCA UN SOLO GIORNO A NATALE"
510 DATA 31,28,31,30,31,30,31,31,30,31,30,25,0
520 END
530 REM
540 REM
550 REM
560 IF M$ = "GENNAIO" THEN LET M = 1
570 IF M$ = "FEBBRAIO" THEN LET M = 2
580 IF M$ = "MARZO" THEN LET M = 3
590 IF M$ = "APRILE" THEN LET M = 4
600 IF M$ = "MAGGIO" THEN LET M = 5
610 IF M$ = "GIUGNO" THEN LET M = 6
620 IF M$ = "LUGLIO" THEN LET M = 7
630 IF M$ = "AGOSTO" THEN LET M = 8
640 IF M$ = "SETTEMBRE" THEN LET M = 9
650 IF M$ = "OTTOBRE" THEN LET M = 10
660 IF M$ = "NOVEMBRE" THEN LET M = 11
670 IF M$ = "DICEMBRE" THEN LET M = 12
680 RETURN
690 REM
700 REM
710 REM
720 REM NOTA BENE: QUESTA SUBROUTINE NON
730 REM            CONTROLLA GLI ANNI MULTIPLI DI 100
740 REM
750 LET A = A / 4
760 LET Z = INT(A)
770 IF A - Z = 0 THEN GOTO 790
780 RETURN
790 LET X(2) = X(2) + 1
800 RETURN
```

Al secondo "giro", I vale 2 e la linea 360 equivale a READ X(2). Poiché il secondo dato nella linea 510 è 28, avremo che X(2) = 28.
Continuando così, a ciascuno dei 13 elementi di X viene assegnato il numero di giorni di ciascun mese, con l'eccezione di X(12), dove il valore è 25, e di X(13) dove è 0.

```
390 GOSUB 750 REM    VERIFICA ANNO BISESTILE
```

Questa linea richiama la subroutine che verifica se l'anno da noi digitato è bisestile. Vediamo come funziona:

```
750 LET A = A/4
760 LET Z = INT(A)
770 IF A — Z = 0 THEN GOTO 790
780 RETURN
790 LET X(2) = X(2) + 1
800 RETURN
```

Un anno è bisestile se è divisibile per 4 senza lasciare resto. Una verifica completa dovrebbe comprendere il caso in cui l'anno è un multiplo di 100, nel qual caso esso è bisestile solo se è anche divisibile per 400, ma, per semplicità abbiamo omesso questa complicazione.
La linea 750 assegna ad A (l'anno) il vecchio valore di A, diviso 4. La variabile A sarà intera (senza decimali) solo se la divisione non dà resto.
Nella linea 760 abbiamo impiegato la funzione INT per assegnare a Z il valore intero di A. Se l'anno digitato, per esempio, è il 1985, A/4 = 496,25 e INT(A) vale 496, ossia il valore intero arrotondato per difetto.
La linea 770 sottrae ad A il valore di Z, controllando se il risultato è zero. Se ciò è vero, l'anno è bisestile (la divisione non ha dato resto) e l'esecuzione passa alla linea 790 (grazie al GOTO), dove il valore di X(2) viene incrementato di 1. Poiché X(2) è il numero dei giorni di Febbraio, avremo che in un anno bisestile questo mese ha 29 giorni.
Se il risultato della sottrazione alla linea 770 non è zero, allora X(2) non viene modificato e l'esecuzione torna alla linea 400 del programma principale mediante la RETURN.
La linea 400 contiene un'altra REM per spaziare il listato, e non implica alcuna azione particolare. L'esecuzione passa alla linea 410, dove viene calcolata R, ovvero il numero di giorni restanti nel mese digitato, sottraendo dal numero totale di giorni del mese quello della data digitata. Se avessimo immesso 12,FEBBRAIO,1983, G equivarrebbe a 12, ed M a 2. Poiché l'anno non è bisestile, X(2) vale 28. Di conseguenza, il valore di R sarebbe 28 — 12 = 16, il numero di giorni di Febbraio da includere nel computo.
La linea 420 inizia un altro ciclo, che serve per incrementare il valore di M. Qui il ciclo inizia da un valore di L = M (il mese digitato) e viene ripetuto finché L = 12. All'interno del ciclo, M viene incrementato di 1 e vengono via via sommati a R i valori X(M), che contengono il numero di giorni di ciascun mese. Al primo "giro", M viene incrementato a 3, quindi ad R (che contiene già 16) viene sommato X(3), ossia il numero di giorni in marzo. L'operazione si ripete finché L = 12.
Si noti che X(12) = 25 e non 31. Ciò è dovuto al fatto che noi vogliamo calcolare i giorni fino a Natale.

```
470 IF R = 1 THEN GOTO 500
```

Questa linea esegue un semplice controllo sul valore di R, per scegliere il messaggio appropriato da visualizzare, tra i due disponibili.
Il programma è stato scritto in una versione di BASIC adatta alla maggior parte dei computer (vedere "Basic & Basic"), ma si potrebbe incontrare qualche difficoltà nella subroutine per l'anno bisestile. In alcuni BASIC linee del tipo:

```
IF M$ = "SETTEMBRE" THEN LET M = 9
```

generano un errore e la subroutine deve essere modificata come segue:

```
560 IF M$ = "GENNAIO" THEN GOTO 900
570 IF M$ = "FEBBRAIO" THEN GOTO 910
580 IF M$ = "MARZO" THEN GOTO 920
    ....
    ....
900 LET M = 1
905 RETURN
910 LET M = 2
915 RETURN
920 LET M = 3
925 RETURN
(... e così via)
```

Questa soluzione, certamente meno elegante, dimostra, tuttavia, che esistono più strade per risolvere lo stesso problema.
Il listato contiene molte linee con la sola istruzione REM: queste servono unicamente per "spaziare" il listato, rendendolo più leggibile.

Basic & Basic

INPUT

Lo ZX81 non accetta l'immissione di più variabili separate dalla virgola. Occorrono le seguenti modifiche:
280 PRINT "GIORNO?", 310 INPUT G, 311 PRINT "MESE?"
312 INPUT M$, 313 PRINT "ANNO?", 314 INPUT A

REM

In molte versioni del BASIC, l'istruzione REM, se segue altre istruzioni in una stessa linea, deve essere preceduta da ":", per esempio:
330 GOSUB 560 : REM CALCOLO DEL MESE

END

L'istruzione END non è disponibile sui computer: ZX81, Spectrum, Oric-1 e Dragon 32. Sostituirla con STOP

Le carriere nell'informatica

Le caratteristiche professionali necessarie per questo settore



per concessione della ICL

I giganti buoni

I computer commerciali di grosse dimensioni (chiamati "mainframe" per distinguerli dai mini e micro computer) richiedono il lavoro di un'équipe di operatori altamente qualificati. Macchine di queste dimensioni sono in grado di eseguire centinaia di programmi contemporaneamente, servendo migliaia di utenti in qualsiasi parte del mondo, per mezzo di linee telefoniche e collegamenti via satellite

L'uso sempre crescente del computer, sia in casa che nelle scuole, ha richiesto una grande quantità di programmatori molto validi: tutte persone che altrimenti non avrebbero mai pensato di intraprendere una carriera nell'informatica. Ma, come sempre, una sola esperienza superficiale può essere pericolosa, specialmente se limitata al linguaggio BASIC. I requisiti di un programmatore professionista sono ben diversi da quelli di una persona che si diletta in casa propria con un piccolo computer. La scelta più ovvia, per un giovane che ha finito la scuola dell'obbligo e che ha un profondo interesse per i computer, è quella di frequentare una facoltà universitaria, per ottenere una laurea, oppure uno dei tanti corsi presso istituti privati, al termine dei quali viene rilasciato un diploma o un attestato. I laureati o diplomati si trovano spesso nella invidiabile condizione di poter scegliere fra più possibilità di lavoro offerte dal mercato.

La disoccupazione in questo settore, infatti, colpisce soltanto coloro che possiedono i più bassi livelli di preparazione, mentre la domanda per ingegneri, analisti di sistema e progettisti di sistema rimane alta.

Poiché fino ad ora i corsi d'informatica erano tenuti principalmente a livello universitario, un settore in forte espansione è quello dell'insegnamento di questa materia nella scuola d'obbligo, dove c'è una notevole carenza di insegnanti specializzati. Una carriera in questo settore può offrire molte soddisfazioni.

Il lavoro nel settore informatico si può idealmente suddividere in più livelli: del primo livello, gerarchicamente più basso, fanno parte gli impiegati specializzati, ossia coloro che hanno imparato ad usare il computer per un'applicazione particolare, come ad esempio il "word processing" o la contabilità. In genere, la mansione principale di queste persone rimane quella di segretario/a o di contabile. Tuttavia ci sono degli impiegati addetti esclusivamente al computer, come l'operatore del terminale o l'operatore del perforatore di schede. Per tali professioni è necessario avere un buon livello di cultura generale e la capacità di pensare in modo logico e organico. Nel successivo livello si possono collocare gli operatori. Anche se i computer usati su scala industriale sono per più ragioni ben diversi dagli home computer, i principi di funzionamento sono simili ed una certa familiarità con questi ultimi può essere di aiuto a chi desideri intraprendere questa professione. Gli operatori imparano presto a capire come funziona il computer e ciò costituisce un buon punto di partenza per lanciarsi nella carriera di programmatore.

Bisogna tener bene in mente, comunque, che il lavoro può essere fisicamente abbastanza stressante. Molti grandi centri, per esempio, operano per 168 ore alla settimana ed è necessario che i macchinari siano sempre sotto sorveglianza. Gli attributi principali per diventare programmatore sono una mente razionale e metodica e la capacità di concentrarsi anche sui più piccoli dettagli. Anche se è generalmente richiesta una laurea o un diploma, una predisposizione naturale al lavoro svolto in modo logico spesso conta di

LA SCELTA DELLA PROFESSIONE

L'ANALISTA



Prima di intraprendere una carriera è consigliabile esaminare da vicino le effettive possibilità di impiego ed i requisiti necessari. Il lavoro dell'**analista-sistemista** è quello di analizzare quali, tra i problemi degli utenti, è possibile risolvere mediante processi computerizzati, tenendo conto delle specifiche necessità e disponibilità degli utenti. Chi intraprende questa professione deve essere in grado di pensare in modo razionale, deve avere una certa predisposizione alle pubbliche relazioni ed un pizzico di creatività. Spesso l'analista è anche addetto alla vendita di software e deve quindi mantenere alta l'immagine commerciale della ditta per la quale lavora

GLI INGEGNERI PROGETTISTI

Forse in futuro saranno gli stessi computer a progettare la nuova generazione di macchine, ma per il momento questo compito spetta all'**ingegnere progettista**. Il suo ruolo è in parte di scienziato e in parte di tecnico: deve avvalersi delle scoperte e degli aggiornamenti teorici per migliorare e sfruttare al massimo il funzionamento delle macchine. Una laurea in ingegneria elettronica è quasi sempre indispensabile



TONY SLEEP

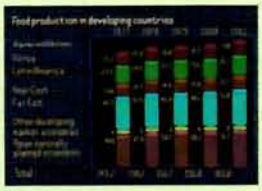
Come in qualsiasi altro reparto di una moderna organizzazione, quello EDP (Electronic Data Processing ossia elaborazione elettronica dei dati) è suddiviso in modo gerarchico: per primo viene il **manager EDP**, cioè la persona responsabile di tutto il reparto di elaborazione dei dati. Tutti coloro che hanno una professione nell'informatica partono da un'esperienza tecnica, acquistando capacità manageriali via via che avanzano nella carriera. Le tre principali aree di specializzazione sono quelle dell'operatore, del programmatore e dell'analista di sistema ed esiste un certo grado di mobilità fra le varie specializzazioni. Come succede in qualsiasi altra professione, è consigliabile intraprendere la carriera con più attestati scolastici possibili. All'inizio questo può sembrare di scarsa importanza, ma una persona con poche qualifiche trova ben presto ostacoli insormontabili.
In Gran Bretagna esistono enti statali (come la BRITISH COMPUTER SOCIETY) che, a seguito di esami, rilasciano un diploma al programmatore o all'analista, attestante la sua preparazione professionale. In Italia esistono numerosi istituti privati che operano nel settore

I PROGRAMMATORI

Il programmatore si serve della relazione preparata dall'analista e la trasforma prima in un piano "tattico" (suddividendo il lavoro da fare in tante parti più elementari) poi provvede alla codifica del programma,



nel linguaggio di programmazione più adatto.
Il **programmatore di software applicativo** è addetto alla preparazione di programmi adatti a particolari applicazioni (in genere commerciali), mentre il **programmatore di software di sistema** si occupa più specificamente del buon funzionamento di tutto il sistema per l'elaborazione di dati. Doti essenziali sono la capacità di concentrazione ed una buona dose di calma



OPERATORI

Forse uno dei lavori fisicamente più pesanti del settore è quello di operatore su computer del tipo multi-utente a multi-programmazione. Egli non solo è costretto a spostarsi spesso, durante il suo turno, portandosi dietro pacchi di dischi e di nastri o scatoloni di carta, ma deve anche essere a piena conoscenza del sistema operativo del computer e avere idee molto chiare sulla priorità dei processi in corso di elaborazione in un dato momento. Decisioni che influiscono sul lavoro dell'intera ditta (quali, ad esempio, la priorità di accesso al computer da parte dei vari reparti di una ditta) vengono prese da un operatore capo



Gli operatori

Gli operatori che svolgono la loro attività in piccole imprese, oltre alle loro normali mansioni sono spesso chiamati ad aiutare i programmatori per diagnosticare eventuali difetti di funzionamento. La cosa più importante, tuttavia, è l'assoluta conoscenza del sistema operativo del computer. La buona qualità del software facilita il lavoro dell'operatore e programmi ben congegnati possono essere usati anche da personale non specializzato, senza peraltro perdere in efficienza



KAI CHOI

FIELD ENGINEER



Di solito l'unica occasione di relax per un operatore è quando il computer si guasta ed è necessario l'intervento di un **field engineer** ("tecnico-riparatore specializzato"). Quasi tutti i computer moderni sono dotati di un sistema di auto-diagnosi incorporato e poiché la costruzione dei computer è quasi sempre di tipo "modulare", il compito del "field engineer" si è alquanto semplificato. Tuttavia, il "field engineer" deve essere esperto in elettronica digitale e deve anche essere un bravo meccanico in grado di lavorare con una sensibilità ed una precisione superiore a quelle di un orologiaio. Per intraprendere la professione di "field engineer" è quasi sempre necessario avere una laurea



Il lavoro meno impegnativo intellettualmente è quello svolto dall'**operatore del terminale**. Le uniche doti richieste sono le stesse che si esigono da una brava dattilografa: velocità e precisione. Questo lavoro può anche divenire piuttosto noioso e ripetitivo, ma, specialmente nelle imprese più piccole, la mansione è svolta saltuariamente da impiegati, che si occupano anche di altri lavori

per concessione del SUNDAY TIMES

più. Ciò può essere di conforto a molti genitori, che intravedono una carriera di programmatore per i loro figli.
Come si è già detto, però, la sola conoscenza del linguaggio BASIC non sempre è sufficiente per entrare nel mondo del lavoro.
Infatti, anche se questo linguaggio è diffusissimo sugli home computer, esso è considerato da molti professionisti un linguaggio mal strutturato ed è opinione comune che il suo uso distolga dall'abitudine di pensare razionalmente, portando a livelli pessimi di programmazione. Il fatto che la maggioranza dei giovani faccia esperienza, appunto, con degli home computer, imparando il BASIC piuttosto che linguaggi meglio strutturati, (come il LOGO o il COMAL) potrebbe portare, in futuro, a seri problemi.
Addirittura, alcune università, avallando questa opinione piuttosto negativa, accettano malvolentieri, tra gli iscritti ai loro corsi di informatica, coloro che hanno imparato il BASIC. Tuttavia, nonostante questo forte contrasto "ufficiale", molti giovani hanno trovato il modo di sfruttare appieno le proprie capacità di programmazione anche in BASIC: alcuni hanno inventato dei giochi, che hanno ottenuto un grande successo commerciale. Le case di software sono alla costante ricerca di nuovi giochi che possano incontrare il gradimento del pubblico giovanile. Alcuni dei "whizz kids" (così vengono chiamati i giovani "maghi" del gioco elettronico) guadagnano grosse cifre pur essendo giovanissimi, e con la sola conoscenza del BASIC. Altri giovani, veri portenti, si servono del linguaggio macchina (che permette di avere un maggiore controllo sul funzionamento di un microprocessore): per loro il futuro è davvero roseo. Purtroppo, molti degli articoli che appaiono sulla stampa sono scritti da persone che non possono apprezzare le differenze fra i due linguaggi e ciò che essi scrivono induce spesso molti genitori a pensare che i loro figli, fanatici del computer, sono destinati a diventare ben presto ricchi sfondati. Può anche succedere, ma certo non molto frequentemente.

Nel mondo del lavoro

Esistono due categorie di programmatori: quelli di software applicativo e quelli di software di sistema. Nel primo caso i programmi vengono sviluppati per una specifica applicazione generale (ad esempio: procedure di contabilità, di magazzino, ecc.). Nel secondo caso i programmi riguardano la gestione interna del computer (per esempio, i sistemi operativi o i programmi per la trasmissione di dati). In genere, il programmatore di software applicativo ha un rapporto diretto con gli utenti dei suoi programmi e spesso lavora in équipe per sviluppare programmi adatti ad uno specifico lavoro. Il programmatore di software di sistema, invece, svolge un lavoro specialistico, che, per la maggior parte del tempo, è "solitario", ovvero quasi sempre a "tu per tu" con il "cervello" della macchina. Al di sopra di questo livello di lavoro ne esiste uno al quale possono accedere solamente i programmatori più brillanti ed i laureati più qualificati: è il regno degli analisti e progettisti di sistema.
All'analista-sistemista spetta affrontare e risolvere a livello concettuale il problema affidatogli e decidere come questo può essere risolto dal computer.
Esaminiamo il caso di un'azienda petrolifera. Supponiamo che venga individuato un nuovo giacimento di grezzo: dopo aver stabilito, con opportuni sondaggi, la lunghezza del giacimento e che la qualità del grezzo varia molto da un punto ad un altro, l'azienda si trova a dover decidere se investire ingenti capitali (l'equivalente di miliardi di dollari) nello sfruttamento del giacimento.
La decisione finale verrà presa basandosi sulle previsioni di vendita relative a tutto il periodo di sfruttamento del giacimento (poniamo, per esempio, 20 anni). Una scelta fondamentale riguarda quale parte del giacimento debba essere sfruttata per prima.
Essendo in gioco cifre colossali, l'azienda si rivolgerà ai propri esperti di elaborazione dati per avere una prima analisi della situazione. L'analista incaricato deve consultarsi con economisti, con esperti di "marketing" del grezzo, con geologi ed altri specialisti. Dopo un complesso lavoro di preparazione, egli appronta un "modello" computerizzato del giacimento. I manager dell'azienda petrolifera, a questo punto, possono sfruttare il "modello", modificandone alcuni elementi, (quali, ad esempio, l'influenza dei costi di estrazione rispetto al prezzo di vendita) per simulare determinate situazioni e decidere, in base ai risultati, se convenga o meno procedere all'estrazione del petrolio. In altre parole il modello computerizzato dà tutte le informazioni necessarie per la decisione finale.
Esistono altri ruoli importanti nel ramo dell'informatica ma pochi sono considerati allo stesso livello dell'analista di sistema. L'unica eccezione potrebbe essere il progettista di hardware: le possibilità di lavoro per ingegneri elettronici sono le più ampie (dal negozietto di riparazioni ad istituti di ricerca), ma i campi dello sviluppo e della ricerca sono aperti esclusivamente a coloro che possiedono le più alte qualifiche nell'ingegneria elettronica.
Spesso gli analisti e i progettisti di hardware e software arrivano ad essere dirigenti d'azienda oppure si dedicano alla libera professione. Ciò costituisce certamente un avanzamento nella loro carriera, ma il loro lavoro rimane sostanzialmente immutato.
I governi di alcuni paesi europei intravedono nella microelettronica la soluzione ad alcuni problemi di disoccupazione.
Indubbiamente, anche in Italia, è necessario che la scuola si adegui ai tempi, poiché in un futuro molto prossimo le speranze di trovare un lavoro, senza conoscere l'informatica, sono davvero esigue.



David Simmonds

Il diciassettenne David Simmonds si è guadagnato circa 22 milioni di lire durante le sue ultime vacanze. È un mago della programmazione e scrive programmi per la COMMODORE (i fabbricanti dei computer PET e VIC). Diversamente dalla maggioranza dei giovani, David si è dedicato allo sviluppo di software "serio" per applicazioni commerciali. Finiti gli studi, egli spera di trovare un buon impiego in un'azienda specializzata in informatica.
David ha iniziato giocando con un computer che suo padre portò a casa, ma ben presto, abbandonati i giochi, si è messo ad imparare seriamente la programmazione. Agli inizi pubblicò alcuni programmi sulla rivista della COMMODORE, vendendo copie dei suoi programmi per pochi soldi. Quando la COMMODORE si accorse di lui, egli fu in grado di dimostrare le sue capacità. Ha ottenuto così il suo primo vero lavoro di programmatore



Eugene Evans

Ha 17 anni ma già il suo reddito annuo è di circa 80 milioni di lire. Infatti Eugene è uno dei "whizz kids" ("maghi" dei giochi elettronici). È grazie alla sua collaborazione che la IMAGINE SOFTWARE di Liverpool riesce a rimanere fra i migliori produttori di giochi elettronici. Gli alti guadagni di cui godono questi "maghi" sono formati per la maggior parte dai diritti d'autore guadagnati sulle vendite dei giochi (come succede per gli scrittori). È evidente che un giovane è la persona più indicata a inventare giochi che piacciano ai suoi coetanei, i maggiori compratori di giochi elettronici

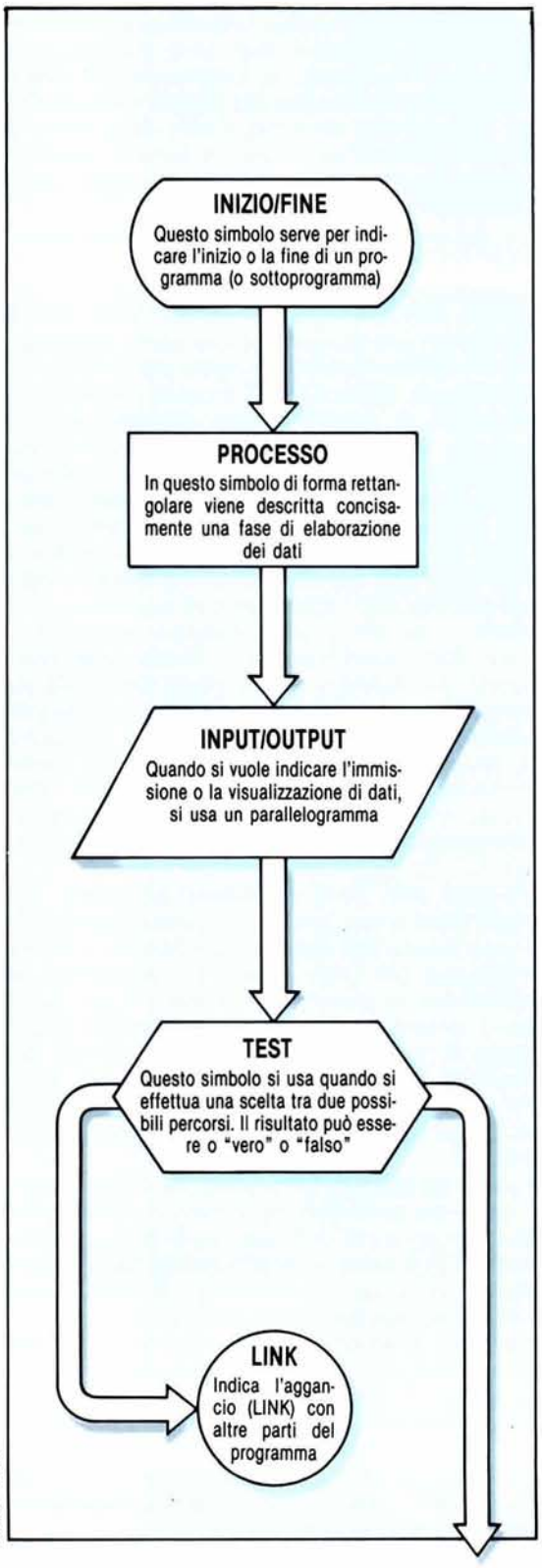
I diagrammi di flusso

Come stilare programmi ben strutturati

Il flusso delle informazioni

Lo scopo principale dei diagrammi di flusso (flowchart) è quello di indicare, in forma sintetica, il funzionamento di un programma o di una particolare fase di esso. Una chiara descrizione della procedura, in particolare quando si eseguono dei confronti per la scelta di uno fra due percorsi, è di estrema importanza. Le varie fasi del programma sono certamente più comprensibili mediante una rappresentazione grafica, piuttosto che con una serie di istruzioni.

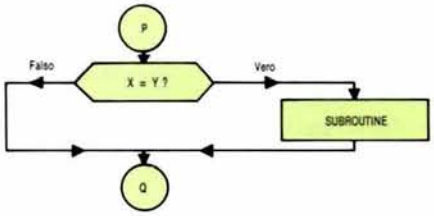
Per indicare i punti in cui si effettua una scelta o un confronto si usa un simbolo di forma romboidale od esagonale



Le varie fasi che portano alla soluzione di un problema possono essere schematizzate sinteticamente con una serie di diagrammi, collegati tra loro secondo un percorso logico. L'uso dei "diagrammi di flusso" è particolarmente utile, nella programmazione, per affrontare un problema in maniera organica e completa.

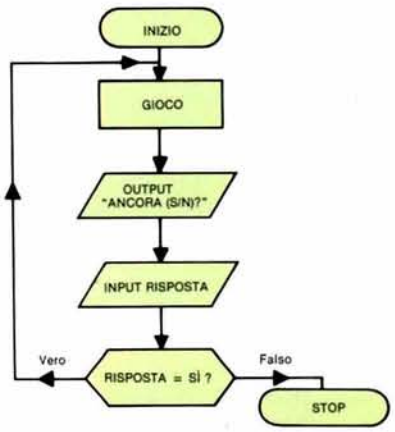
Ogni "blocco" di un diagramma di flusso rappresenta un particolare stadio nel processo di elaborazione ed i canali di collegamento tra i vari blocchi indicano i percorsi possibili da una fase all'altra della procedura. I percorsi possono essere soltanto a "senso unico", pertanto vengono usate delle piccole freccette per indicarne la direzione.

Quando una delle fasi implica una scelta, viene usato un simbolo "decisionale", di forma esagonale o romboidale. Poiché si tratta di una scelta, da questo simbolo partono due percorsi. Ecco un esempio, in cui la scelta, indicata dall'esagono, implica l'esecuzione o meno di una subroutine:



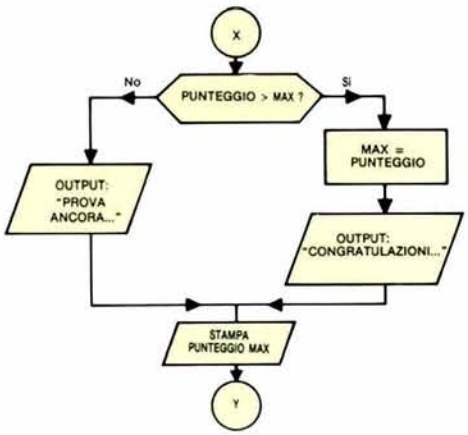

```
120 IF X = Y THEN GOSUB 300
```

Lo stesso simbolo è usato anche per indicare un'operazione di verifica posta alla fine di un ciclo di istruzioni. Nell'esempio che segue, l'esecuzione riprende dall'inizio, se alla domanda «Ancora?» si risponde di sì:



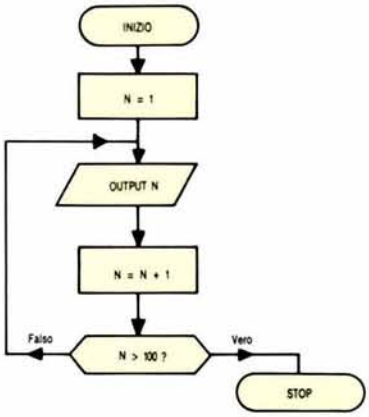
```
90 REM  INIZIO DEL GIOCO
100
110
[... parte del programma relativa al gioco]
790
800
810 PRINT "ANCORA (S/N)";
820 INPUT R$
830 IF R$ = "S" THEN GOTO 100
840 END
```

In altri casi, in base al confronto di due valori, viene deciso quale eseguire dei due gruppi di istruzioni. Nel seguente esempio, viene fatto un confronto tra il punteggio di un giocatore e quello massimo finora raggiunto:




```
1200 IF PUNTEGGIO > MAX THEN GOTO 1230
1210 PRINT "PROVACI ANCORA, DEVI ARRIVARE A";
1220 GOTO 1250
1230 LET MAX = PUNTEGGIO
1240 PRINT "CONGRATULAZIONI, IL NUOVO MASSIMO È";
1250 PRINT MAX
```

Si noti che il punteggio massimo viene stampato in ambedue i casi e che i due percorsi possibili si ricongiungono prima di effettuare questa operazione di stampa.
Nei programmi, tutte le decisioni vengono prese in base a confronti simili, dove il risultato può essere o "vero" o "falso". Non si può avere, con questo sistema, un "forse" come risultato, quindi i confronti devono essere ricondotti sempre ad una scelta tra due possibili risposte.
Tutti i linguaggi di programmazione dispongono di istruzioni più o meno simili, che servono per modificare il percorso dell'esecuzione in base al risultato di un confronto: se la condizione è "vera", il percorso viene deviato, altrimenti viene eseguita la linea di programma immediatamente seguente. In alcune versioni "ridotte" del BASIC, in cui esiste soltanto la possibilità IF...THEN, dobbiamo ricorrere a qualche espediente per far sì che l'esecuzione devii opportunamente dopo il confronto, usando una GOTO (come nella linea 1200 dell'ultimo esempio). La linea 1210 viene eseguita soltanto se il risultato del confronto alla linea 1200 è "falso".
L'istruzione GOTO (linea 1220) è necessaria affinché l'esecuzione salti le linee 1230 e 1240, che altrimenti verrebbero eseguite dopo la 1210. I due percorsi si ricongiungono, perciò, alla linea 1250.
Nel convertire i diagrammi di flusso in programmi, si è spesso indotti ad usare molte più istruzioni GOTO del necessario, a scapito dell'eleganza, ma, soprattutto, della chiarezza di programmazione. Vediamo come un diagramma di flusso ci può aiutare nella descrizione di un semplice compito: la stampa di tutti i numeri compresi tra uno e cento.




```
10 LET N = 1
20 PRINT N
30 LET N = N + 1
40 IF N > 100 THEN END
50 GOTO 20
```

L'impiego dei diagrammi di flusso serve per descrivere con precisione tutte le varie fasi del processo, ma l'efficienza del programma risultante dipende sempre dalla bravura e dall'inventiva del programmatore. Una soluzione molto più efficace ed elegante dello stesso problema si ha impiegando un semplice ciclo FOR-NEXT:

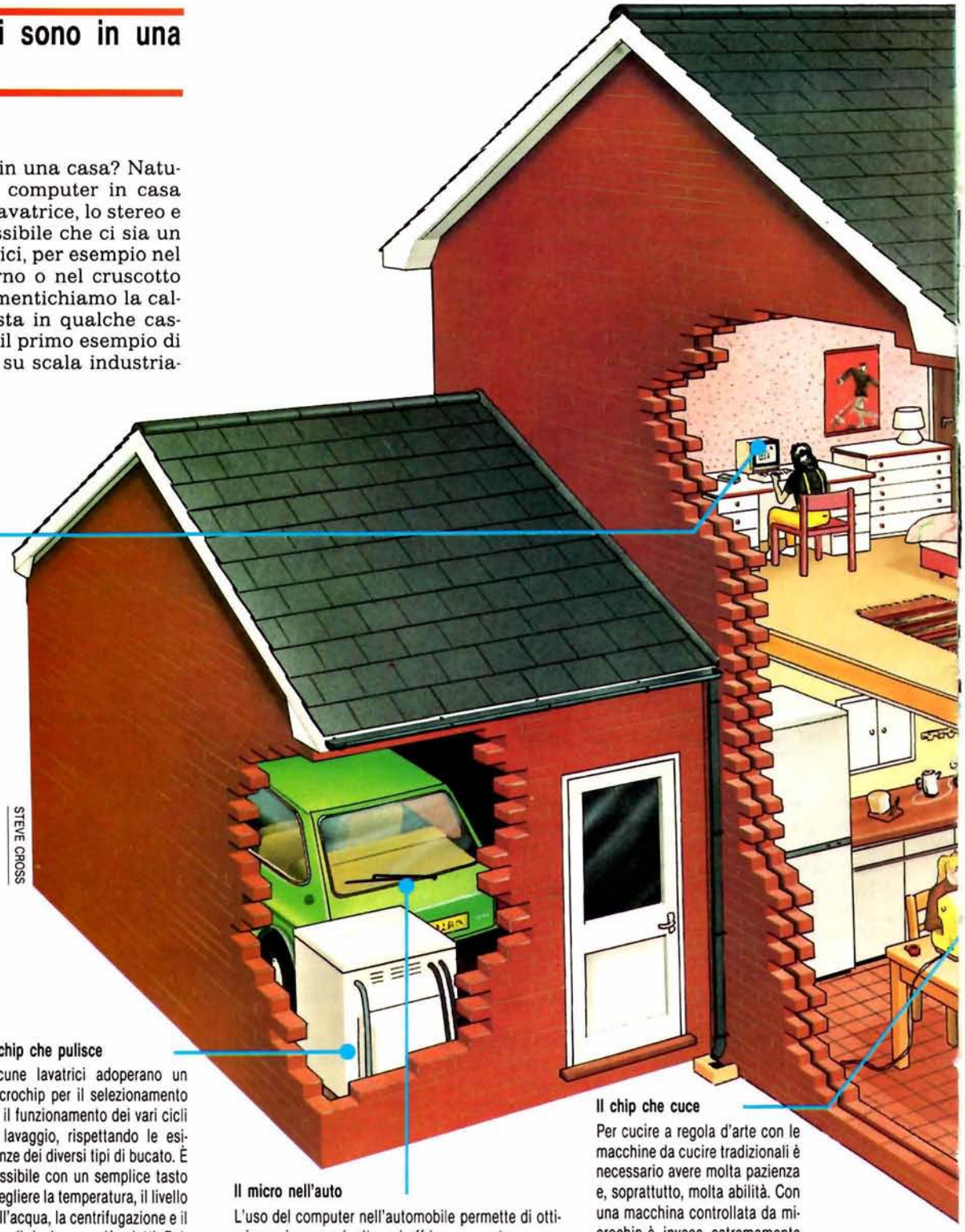
```
10 FOR N = 1 TO 100
20 PRINT N
30 NEXT N
40 END
```

Il diagramma di flusso visto più sopra, sebbene adatto a descrivere il problema, non ci può indicare questa seconda soluzione. Vedremo più avanti nel corso come rappresentare, nei diagrammi di flusso, l'impiego dei cicli FOR-NEXT.
I diagrammi di flusso sono molto utili durante l'analisi concettuale del problema da risolvere, particolarmente nei punti più ingarbugliati. I programmatori più esperti, infatti, usano i diagrammi di flusso per descrivere solo le parti più complicate del programma o quelle non riconducibili a situazioni elementari ricorrenti. I principianti spesso sottovalutano l'utilità dei diagrammi di flusso e si precipitano a tradurre le loro idee direttamente nelle istruzioni del linguaggio. I programmi scritti con questo criterio raramente funzionano alla prima prova e quasi sempre contengono subdoli errori, che si manifestano solo successivamente.

Scienza domestica

Quanti computer ci sono in una casa?

Quanti microchip ci sono in una casa? Naturalmente se c'è un home computer in casa questo ne ha uno... ma la lavatrice, lo stereo e il videoregistratore? È possibile che ci sia un chip in tanti elettrodomestici, per esempio nel regolatore termico del forno o nel cruscotto dell'automobile? E non dimentichiamo la calcolatrice elettronica, riposta in qualche cassetto, o l'orologio digitale (il primo esempio di microprocessore prodotto su scala industriale).



STEVE CROSS

Un gioco da bambini

Spesso, i bambini sanno sfruttare l'home computer meglio degli adulti, accettandolo con naturalezza, come hanno fatto per la televisione. La conoscenza del linguaggio LOGO stimola i bambini ad esplorare l'ambiente che li circonda e ad imparare da soli. Anche una versione ridotta del LOGO, che si avvalga di immagini create con la "tartaruga", può essere utile ai ragazzi più piccoli.
Esiste, in commercio, del software didattico ad alto livello. I giochi possono essere stimolanti e educativi, ma spesso il talento e l'interesse di un ragazzo si sviluppano maggiormente esercitandosi su problemi esistenti nella programmazione del computer. È possibile trovare versioni del linguaggio LOGO per quasi tutti gli home computer: ciò offre un enorme potenziale per incoraggiare ad affrontare e risolvere i problemi esistenti in molti campi al di fuori della programmazione

Il chip che pulisce

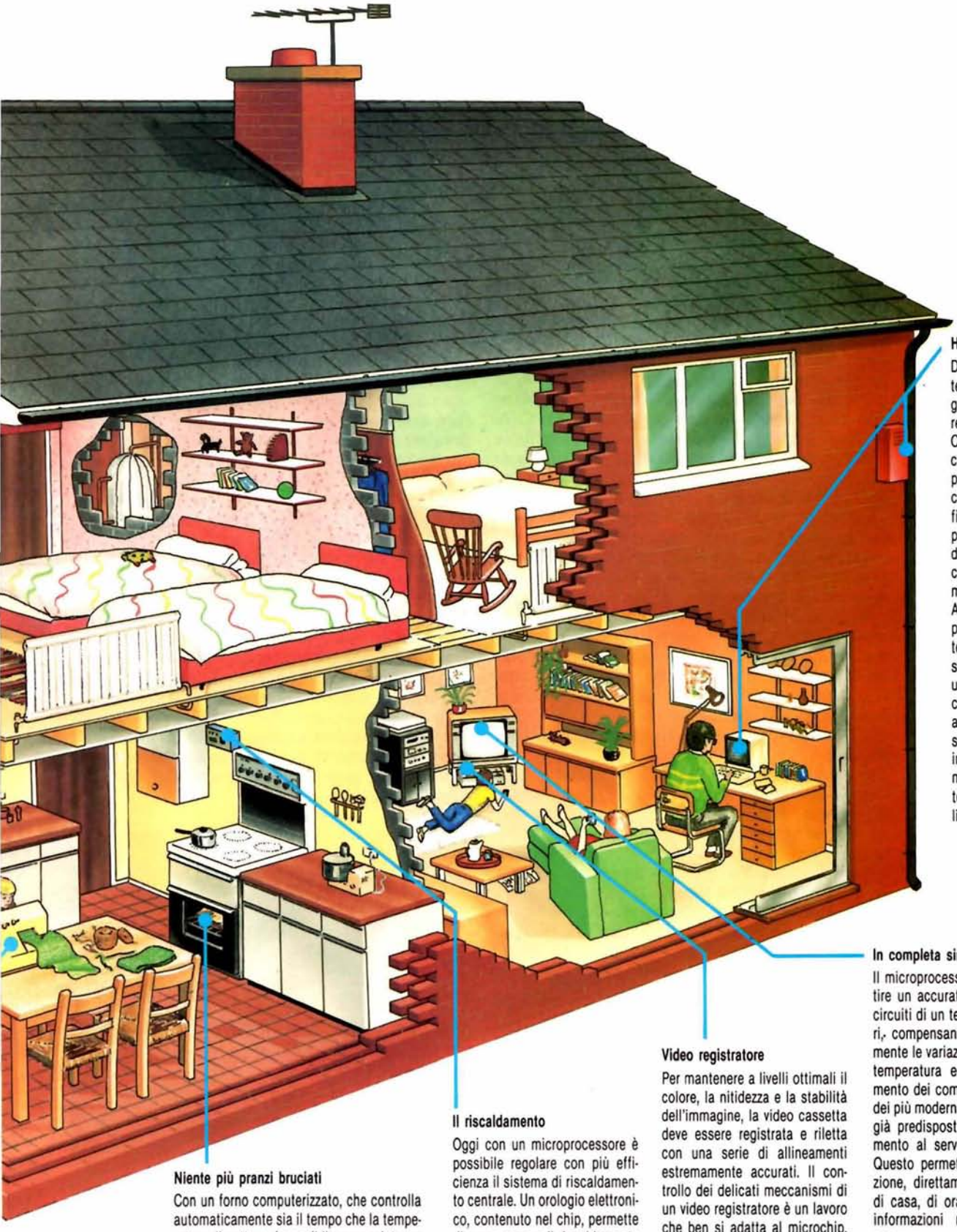
Alcune lavatrici adoperano un microchip per il selezionamento ed il funzionamento dei vari cicli di lavaggio, rispettando le esigenze dei diversi tipi di bucato. È possibile con un semplice tasto scegliere la temperatura, il livello dell'acqua, la centrifugazione e il tipo di risciacquo più adatti. Poiché, con l'impiego dei chip, vengono eliminate le parti meccaniche mobili (eccettuato il cestello naturalmente) la macchina durerà molto più a lungo e le riparazioni si ridurranno al minimo

Il micro nell'auto

L'uso del computer nell'automobile permette di ottimizzare i consumi, oltre ad offrire una vasta gamma di funzioni finora inesistenti. Il computer è in grado di calcolare il consumo del carburante, di indicare e controllare la velocità e perfino di agire come antifurto. In alcuni casi, addirittura, può parlare al guidatore, avvisandolo se il livello dell'olio o quello dell'acqua nella batteria sono al di sotto del minimo di sicurezza

Il chip che cuce

Per cucire a regola d'arte con le macchine da cucire tradizionali è necessario avere molta pazienza e, soprattutto, molta abilità. Con una macchina controllata da microchip è, invece, estremamente facile eseguire anche ricami molto complicati. Oltre ai modelli di ricamo già predisposti nel programma è possibile anche creare nuovi ricami che la memoria conserverà anche quando la macchina è disinserita



Home computer

Di solito un home computer viene acquistato per giocarci, ma esso può avere molte altre applicazioni. Chi ha una piccola attività commerciale, per esempio, può tenerci la propria contabilità, oppure chi ha figli può sfruttare il computer per applicazioni didattiche. Ma ci sono anche altre applicazioni domestiche meno comuni. Ad esempio, non esiste praticamente nessun limite alle sofisticazioni che si possono applicare ad un sistema di allarme: il computer è in grado di azionare rivelatori nascosti di vari tipi ed è perfino in grado di chiamare automaticamente un numero telefonico da noi prestabilito, in caso di emergenza

In completa sintonia

Il microprocessore può garantire un accurato controllo sui circuiti di un televisore a colori, compensando automaticamente le variazioni dovute alla temperatura ed all'invecchiamento dei componenti. Alcuni dei più moderni televisori sono già predisposti per l'allacciamento al servizio "Videotel". Questo permette la consultazione, direttamente sul video di casa, di orari dei treni, di informazioni meteorologiche, ecc.

Video registratore

Per mantenere a livelli ottimali il colore, la nitidezza e la stabilità dell'immagine, la video cassetta deve essere registrata e riletta con una serie di allineamenti estremamente accurati. Il controllo dei delicati meccanismi di un video registratore è un lavoro che ben si adatta al microchip. Inoltre, il microprocessore può essere anche programmato per effettuare la registrazione, in orari e su canali selezionabili, di programmi televisivi quando siamo assenti da casa

Il riscaldamento

Oggi con un microprocessore è possibile regolare con più efficienza il sistema di riscaldamento centrale. Un orologio elettronico, contenuto nel chip, permette di programmare il riscaldamento a seconda delle effettive esigenze. È anche possibile prefissare la temperatura nelle varie stanze della casa (per esempio nelle camere da letto e nel garage), risparmiando così sul consumo energetico

Niente più pranzi bruciati

Con un forno computerizzato, che controlla automaticamente sia il tempo che la temperatura di cottura, è possibile avere pietanze sempre cucinate alla perfezione. Dopo aver visualizzato sullo schermo la ricetta desiderata (magari in collegamento con un servizio tipo "Videotel"), ed aver preparato gli ingredienti, possiamo lasciare al computer il compito di cucinare il pranzo alla giusta temperatura e per il tempo programmato

L'anello di congiunzione

Gli apparecchi per lo scambio di dati tra computer

Il termine modem deriva dalla contrazione di "Modulatore-Demodulatore". In vendita da più di 5 anni, il modem si sta diffondendo solo adesso tra gli utenti di home computer. Molti possessori di home computer pensano che la spesa per acquistare un apparecchio per collegare il computer al telefono sia totalmente inutile. Invece, grazie al modem è possibile far "dialogare" il nostro computer con altri situati in una qualsiasi parte del mondo. L'unica condizione è che il computer all'altro capo della linea telefonica, sia esso un altro home computer o un grande elaboratore di un'università, disponga anch'esso di un modem. Il collegamento con grossi computer permette l'accesso istantaneo ad enormi "banche dati" ed a servizi informativi generici o specifici, come, ad esempio, l'andamento aggiornato della borsa. Altrimenti possiamo collegare il nostro computer con quello di un nostro amico, per scambiarci del software, per trasmettere della piccola "corrispondenza elettronica" e perfino per giocare insieme.
Il funzionamento del modem è molto simile a quello dell'interfaccia per unità a nastri, di cui la maggior parte degli home computer è fornita.
Sia l'interfaccia per unità nastri che il modem convertono gli "1" e gli "0" in audio frequenze. Nel caso dell'interfaccia tali frequenze sono facilmente registrabili su nastro magnetico (come se fossero normali segnali audio). Per quanto riguarda il modem, invece, le audio frequenze vengono semplicemente inviate attraverso le linee telefoniche e riconvertite in numeri binari dal modem ricevente. La interfaccia per unità a nastri, tuttavia, non deve fare altro che convertire i numeri binari in segnali audio per registrare sul nastro (questo processo si chiama "modulazione"), oppure svolgere il processo inverso, per la conversione dei segnali audio trasmessi dal registratore in numeri binari (chiamato "demodulazione"). Viceversa, la maggior parte dei modem è progettata per uno scambio contemporaneo d'informazioni tramite una sola linea telefonica e pertanto necessita di due diverse bande di frequenza e di quattro frequenze. Un diffuso modello di modem usa, in trasmissione, una frequenza di 1.070 Hz per lo "0" e di 1.270 Hz per l'1.
Per la ricezione usa, invece, una frequenza di 1.025 Hz per lo "0" e di 2.225 Hz per l'1.
È da notare che le due frequenze su ciascuna banda sono molto vicine tra loro: c'è solamente una differenza di 200 Hz fra l'uno e lo zero in tutt'e due i casi. Questo sistema è completamente diverso da quello adottato nell'interfaccia per unità a nastri, dove la frequenza che rappresenta l'uno è di solito il doppio di quella che rappresenta lo zero. Per poter decodificare frequenze così vicine tra loro, il modem necessita di un circuito elettronico abbastanza complesso, perciò il costo di questi apparecchi è abbastanza alto: a volte un modem può costare quanto un piccolo home computer.

L'accoppiatore acustico

Quasi tutti i modem oggi in uso sono apparecchi totalmente elettronici, e vengono direttamente collegati alle linee telefoniche per mezzo della spina del telefono. L'azienda telefonica ha stabilito dei costi e delle modalità molto precise per l'installazione dei modem, che rendono talvolta molto costoso il loro uso. Un modo più economico per risolvere il problema è di usare un accoppiatore acustico, in quanto, non essendo elettricamente collegato alla linea telefonica, non sottostà, generalmente, alla regolamentazione concernente i modem. L'accoppiatore acustico converte i segnali del computer in frequenze audio sinusoidali (ossia in suoni), che vengono amplificati e riprodotti da un piccolo altoparlante appoggiato vicino alla cornetta del telefono



IAN DOBBIE



Le macchine FAX

Le macchine FAX (la sigla proviene da "facsimile") stanno velocemente prendendo campo in Europa e negli Stati Uniti, mentre in Giappone anche le aziende più piccole, e perfino molte abitazioni private, ne posseggono una. Una FAX permette, in pochi secondi, di trasmettere documenti, disegni tecnici e illustrazioni (anche di grosse dimensioni), ad un'altra macchina FAX usando un modem incorporato ed un comune apparecchio telefonico

Atari 400 e Atari 800

Anatomia di microcomputer di diffusione europea: tecnologie a confronto



CHRIS STEVENS

LE TASTIERE DELL'ATARI

La differenza più evidente tra i due modelli Atari è la tastiera. Mentre l'Atari 800 è dotato di una tastiera di tipo tradizionale, il 400 ne ha una "a sfioramento" che, sebbene di buona qualità, non è adatta a chi è abituato alla tastiera della macchina da scrivere. Le tastiere "a sfioramento" comunque, sono spesso adoperate in quegli apparecchi (specialmente ad uso militare) impiegati in ambienti molto polverosi

Il fenomenale successo della Atari, che ha la sua principale sede a Sunnyvale (nella famosa Silicon Valley californiana), proviene principalmente dai giochi per computer. Il primo gioco fu il *Ping Pong* per televisori in bianco e nero.

In questi ultimi sei anni, l'ascesa della Atari, oggi parte della Warner Communications Group, è stata inarrestabile, fino a diventare una delle maggiori produttrici di home computer e di apparecchi per sale giochi.

Lo stile inequivocabile e l'aspetto solido e compatto dell'Atari 400 e dell'Atari 800 hanno a lungo rappresentato un modello dal quale molte altre case hanno preso spunto. L'ottima fama che accompagna questi due apparecchi è principalmente dovuta alle fenomenali possibilità grafiche e al software di buon livello di cui essi dispongono.

Le differenze tra l'Atari 400 ed il "fratello maggiore" 800 consistono, oltre che nel prezzo, in una maggiore memoria disponibile (fino a 48 kbyte contro i 16 Kbyte del 400), nel numero di connettori per cartucce (due invece di una sola) e la possibilità, per il modello 800, di essere collegato non solo ad un normale televisore, ma anche a monitor professionali.

La differenza più apprezzabile, comunque, rimane forse la tastiera, che nell'800 è di tipo tradizionale.

Per evitare inconvenienti derivanti dall'uso di registratori di bassa qualità, l'Atari vende, separatamente, un registratore "a cassette" specificatamente progettato per questo impiego. La maggior parte dei possessori di Atari, però, preferisce acquistare almeno un'unità a dischetti, che dà la possibilità di accedere ad una larga biblioteca di programmi.



L'unità a dischi

L'Atari 810, con soli 88 Kbyte di capacità, è un'unità a dischi che comincia a sentire il peso degli anni; oltretutto è relativamente lenta rispetto ad apparecchi più recenti. Nonostante ciò, rimane un accessorio quanto mai utile



Il registratore "a cassette"

Essendo stato progettato appositamente per i computer Atari, questo registratore è senz'altro molto più affidabile dei comuni registratori per uso domestico. Non ha altoparlante e non usa "cassette" di tipo normale. Non si possono, inoltre, assegnare nomi ai programmi archiviati



Il Joystick Atari

Questo è forse l'accessorio che lascia più a desiderare, essendo basato sull'impiego di interruttori, che offrono meno "elasticità" di risposta rispetto ad altri tipi in commercio

SCHEDA CPU

ANTIC

Un chip appositamente progettato per l'Atari gestisce parte delle funzioni del video

CTIA o GTIA

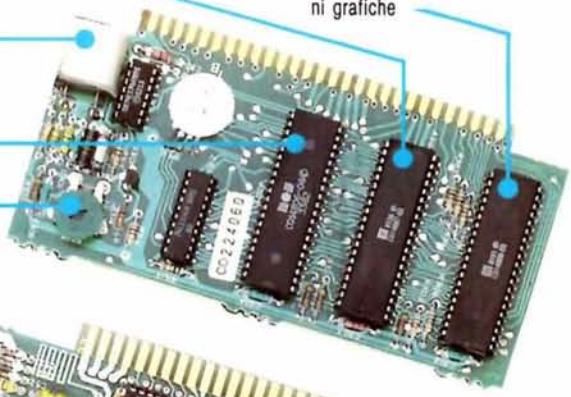
Questo chip, che si trova soltanto sugli Atari, gestisce i colori sul video ed altre funzioni grafiche

Clock del colore

Si può variare la risoluzione grafica del computer regolando la frequenza di oscillazione di questo circuito

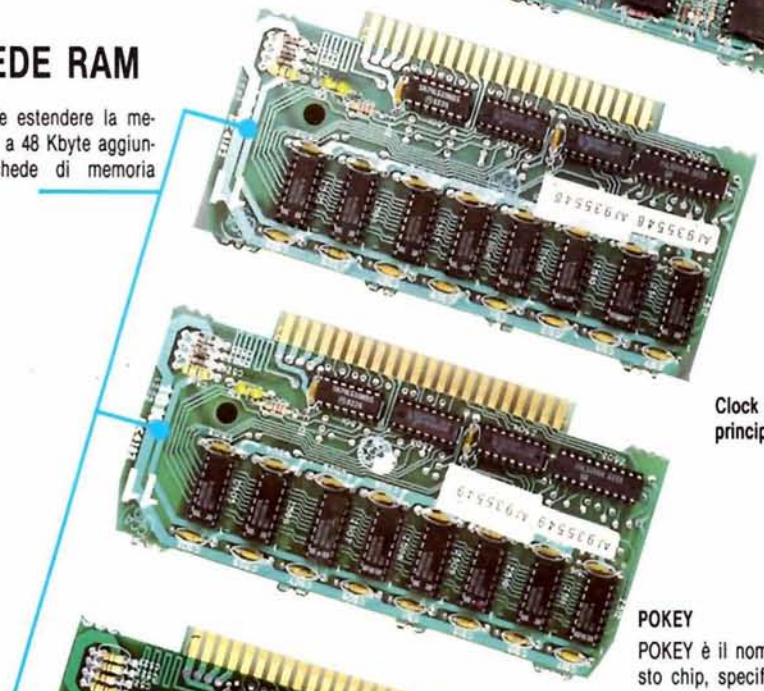
CPU 6502

Regolazione del colore



SCHEDE RAM

È possibile estendere la memoria fino a 48 Kbyte aggiungendo schede di memoria RAM



Clock principale

POKEY

POKEY è il nome di questo chip, specificatamente progettato per l'Atari, che gestisce il funzionamento della tastiera, del canale seriale di trasferimento e del generatore di effetti audio

Interfaccia per periferiche 6520

Questo chip controlla gli ingressi dei joystick



Altoparlante



L'Atari può svolgere molte funzioni accessorie sostituendo questa scheda, che contiene le ROM. Alcuni linguaggi di programmazione vengono forniti su questo tipo di scheda

SCHEDA DI PERSONALIZZAZIONE

SCHEDA PRINCIPALE

Diodi raddrizzatori
Poiché l'alimentazione è in corrente alternata, questi diodi la convertono in corrente continua, necessaria al funzionamento

Connettore per periferiche

Interruttore generale

Connettori per le espansioni RAM
Ogni scheda permette di aggiungere 16 Kbyte di memoria RAM

Selettore di canale

Connettore per la CPU
Serve a collegare la CPU, che è collocata su di un'altra scheda

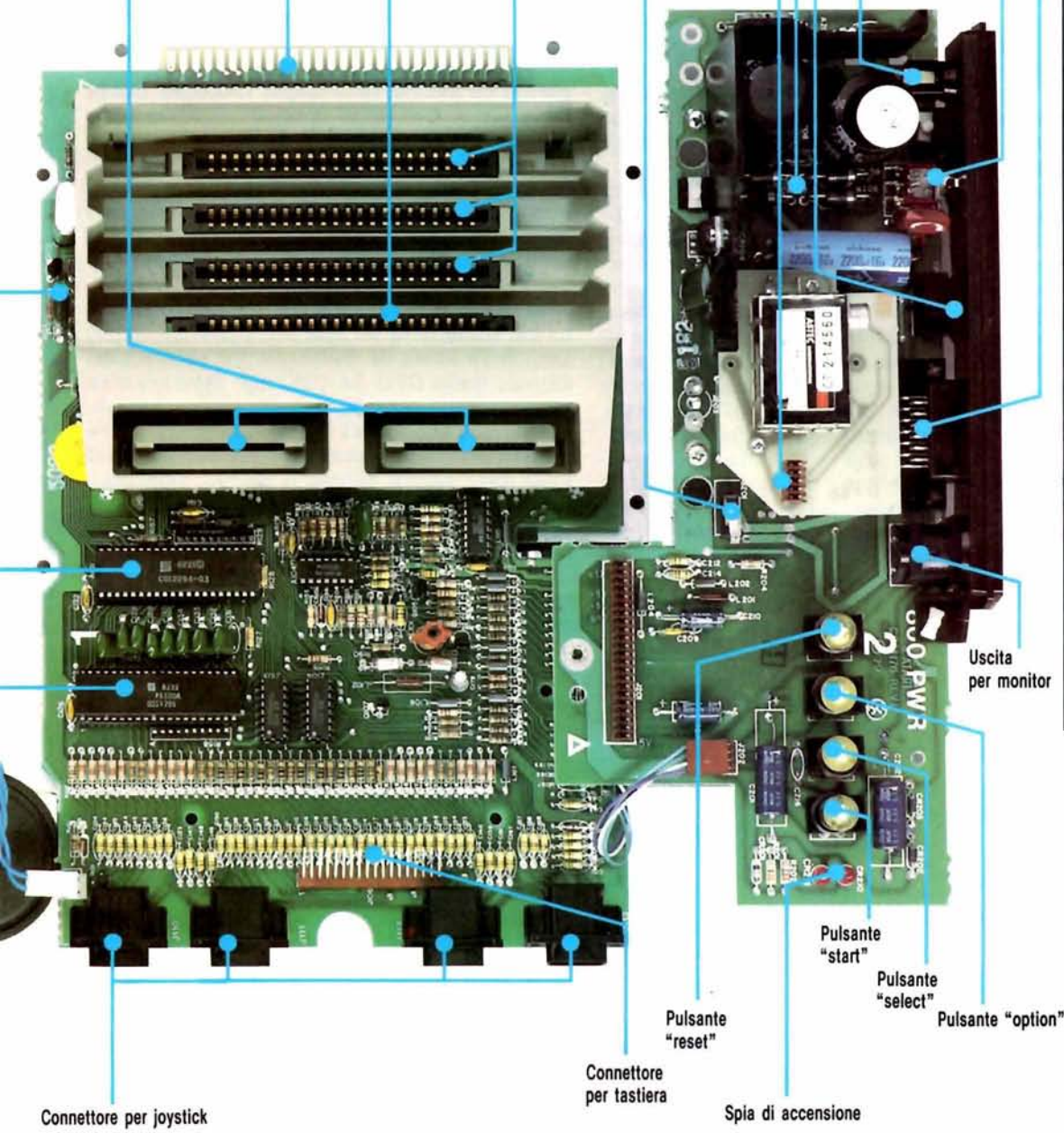
Connettore per video

Presa di alimentazione

Connettori per cartucce
Qui vengono inserite le cartucce contenenti i programmi o i giochi

Connettore per la scheda ROM

Interruttore di sicurezza
Toglie automaticamente la corrente se viene sollevato il coperchio del computer



Uscita per monitor

Pulsante "start"

Pulsante "select"

Pulsante "option"

Pulsante "reset"

Connettore per tastiera

Spia di accensione

Connettore per joystick

ATARI 800

DIMENSIONI

mm. 405 x 330 x 110

PESO

4.200 grammi

CPU

6502

CLOCK

1,79 MHz

MEMORIA

16 Kbyte di RAM espandibile a 48 Kbyte

VIDEO

Per testi, 40 x 32 caratteri; per la grafica, un massimo di 329 x 192 punti, con 16 colori di 8 tonalità ciascuno

INTERFACCE

Connettore per TV, connettore per registratore a cassette (di tipo speciale), porta seriale per periferiche e porta per joystick

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

BASIC A+, PILOT, C

ACCESSORI COMPRESI

Alimentatore, un manuale

TASTIERA

57 tasti mobili, più 3 tasti per funzioni speciali

DOCUMENTAZIONE

I manuali introduttivi sono di buona qualità e compilati con cura. Su richiesta, la Atari fornisce un manuale tecnico con tanto di diagrammi dei circuiti dell'apparecchio, oltre ad alcuni listati dei programmi interni ed altre informazioni di carattere tecnico

Il dialogo digitale

Come è regolato il flusso di dati tra il computer ed il "mondo esterno"

Con "Input/Output", spesso abbreviato in I/O, si indica lo scambio di dati, in ingresso ed in uscita, tra la CPU (il "cuore" del computer) ed il "mondo esterno". In questo contesto, per "mondo esterno" si intende un qualsiasi dispositivo collegato al computer. Non vengono incluse tra i dispositivi esterni le memorie RAM o ROM, che sono considerate parti integranti del computer. La distinzione fra ciò che è "interno" e ciò che è "esterno" al computer è alquanto arbitraria: in linea di massima, comunque, tutti i circuiti logici (vedere pag. 92) che lavorano in stretta vicinanza alla CPU ed alla memoria principale, sono considerati "interni" al computer.

Dei dispositivi esterni, che impiegano i circuiti di I/O per comunicare con il computer, fa parte una vasta gamma di periferiche: la tastiera, le unità a dischi, i joystick, la stampante, lo schermo, ecc.

Quando la CPU desidera richiamare un dato dalla memoria, deve indicare, in primo luogo, l'"indirizzo" della locazione di memoria dove è conservato il byte in questione. Analogamente, quando la CPU desidera memorizzare un byte di dati da usare in seguito, deve prima fornire l'indirizzo della locazione dove l'informazione sarà conservata. Questo processo si chiama, appunto, "indirizzamento della memoria". In breve, la CPU emette un numero binario corrispondente alla locazione di memoria su una serie di 16 "piedini di indirizzamento" della CPU. Questo gruppo di connessioni si chiama "il bus degli indirizzi". Uno speciale circuito, che fa parte delle memorie, è in grado di decodificare questi 16 segnali, selezionando la giusta locazione all'interno della memoria. Con sedici bit binari sono possibili 65.536 diverse combinazioni di 1 e 0, pertanto si possono indirizzare altrettante locazioni di memoria.

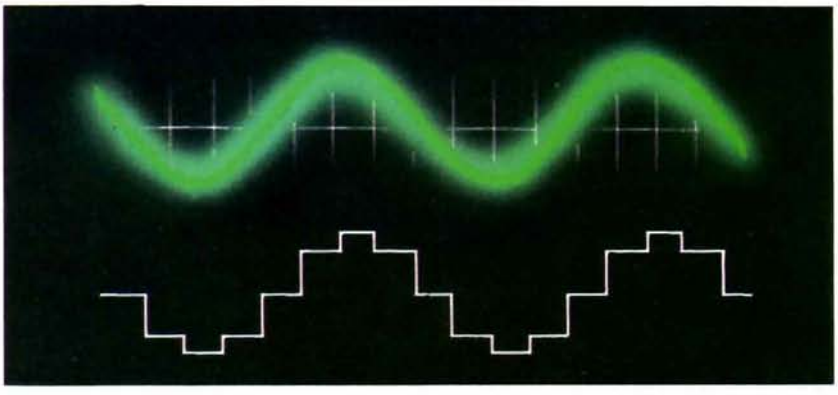
Quando, invece, il computer desidera comunicare con un congegno esterno, deve "localizzare" questo congegno in modo analogo al precedente, ma in questo caso esistono soltanto 8 linee di indirizzamento e questo limita a 256 il numero totale delle locazioni di I/O selezionabili. Questo numero è decisamente inferiore a quello ottenibile con 16 linee di indirizzamento, ma è più che sufficiente, in quanto non vengono collegati mai così tanti congegni esterni ad un computer.

Dall'analogico al digitale

In natura, la variazione di una qualsivoglia quantità non avviene, salvo rare eccezioni, per passi "discreti" o "digitali", ma piuttosto mediante un movimento "continuo", come succede, per esempio, nelle maree.

Affinché un computer possa misurare quantità variabili, occorre quindi, per prima cosa, "digitalizzare" la variazione. I convertitori Analogico/Digitali (A/D) servono proprio a questo scopo, prelevando campioni del segnale da misurare ad intervalli di tempo regolari (esempio 100 campionature al secondo). Ciascuna misurazione viene singolarmente memorizzata, sotto forma di valore digitale, in una particolare locazione di memoria. È anche possibile eseguire opportuni calcoli per riconoscere eventuali valori anomali.

La funzione di un convertitore Digitale/Analogico (D/A) è alquanto simile, ma in direzione opposta, ed impiega delle tecniche statistiche per simulare una certa continuità nel segnale di uscita

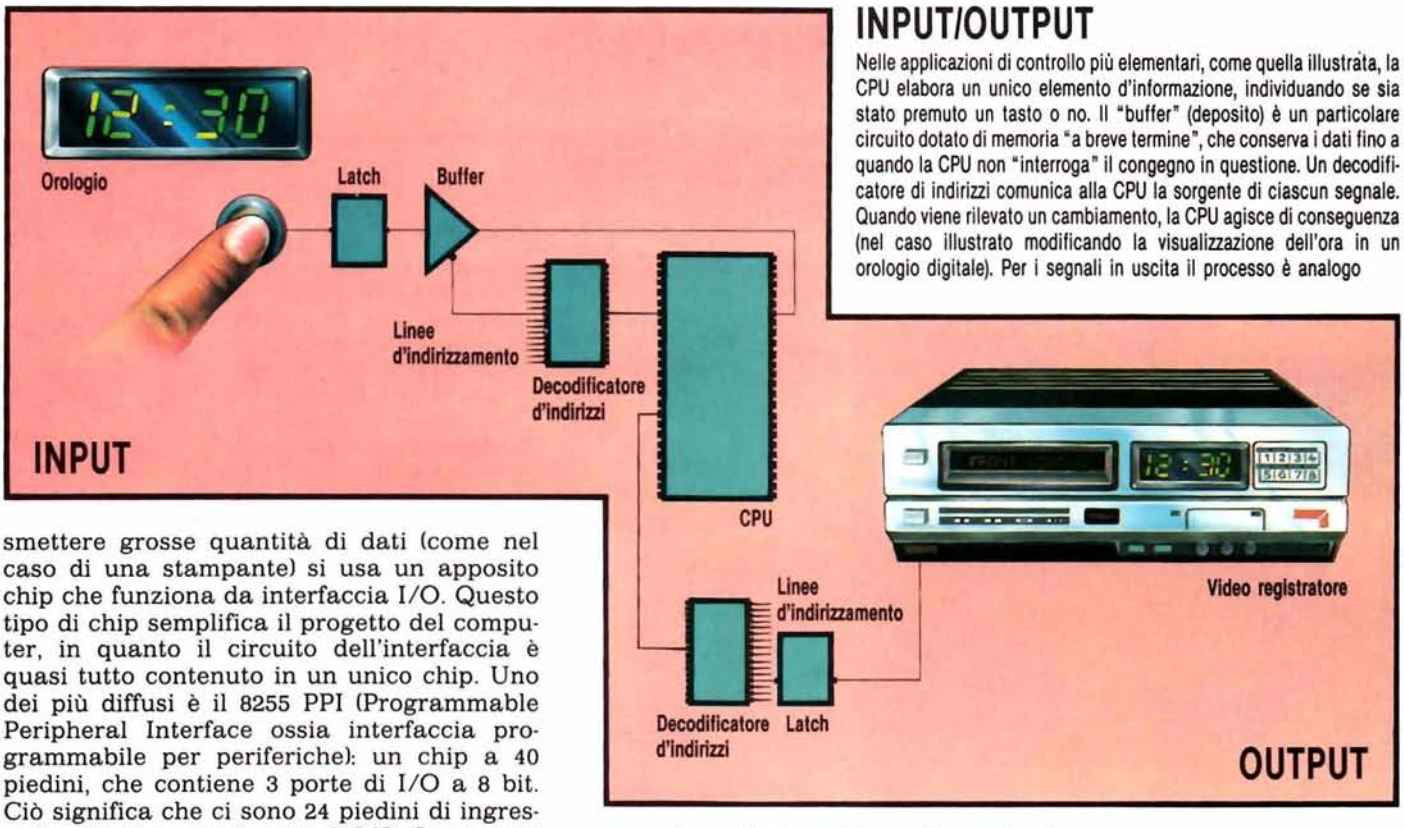


MARK WATKINSON

Il selezionamento delle periferiche

Per meglio comprendere come il computer operi la selezione di un congegno esterno e come trasmetta (o riceva) i dati, consideriamo un semplice esempio, nel quale il dispositivo periferico sia un LED (Light Emitting Diode ossia diodo emittente di luce) collocato sulla tastiera del computer. Il diodo deve restare acceso quando è attiva la funzione "maiuscole" della tastiera. Per il computer, questo LED non è altro che un congegno al quale può inviare dei dati. Nel caso di un unico LED, il dato da inviare sarà o un "1" (per l'accensione del LED) o uno "0" (per spegnere il LED). Nonostante questo LED sia di modeste pretese (ha bisogno di un solo bit d'informazione), esso deve avere un indirizzo per essere selezionato dalla CPU. La CPU non deve perdere troppo tempo per una operazione così semplice, perciò deve essere in grado di selezionare il LED, indicando contemporaneamente se il LED deve essere acceso o spento. Poniamo, per esempio, che l'indirizzo del LED sia rappresentato dal numero 32. Per selezionarlo, la CPU dovrà depositare sulle linee di indirizzamento l'equivalente, in numeri binari, di 32, ossia 00100000. Il LED in questione si avvale di uno speciale circuito di decodifica, che ignora tutte le altre combinazioni di bit sulle linee di indirizzamento, ma quando su queste compare la combinazione 00100000, il circuito di decodifica la riconosce e genera una corrente ad alto voltaggio (ossia una condizione "vera" in uscita). La restante parte del circuito necessario per il funzionamento del LED, è costituita da un piccolo chip chiamato "latch". Questo chip serve per mantenere acceso o spento il LED fino al ricevimento di nuovi dati.

La maggior parte dei congegni esterni con i quali il computer comunica ha una struttura molto più complessa di un semplice LED: la stampante rappresenta un tipico esempio di un congegno periferico. Ogni volta che il computer si mette in contatto con la stampante, i dati trasmessi rappresenteranno il codice necessario per la stampa di un intero carattere. Di solito, quando è necessario tra-



INPUT/OUTPUT

Nelle applicazioni di controllo più elementari, come quella illustrata, la CPU elabora un unico elemento d'informazione, individuando se sia stato premuto un tasto o no. Il "buffer" (deposito) è un particolare circuito dotato di memoria "a breve termine", che conserva i dati fino a quando la CPU non "interroga" il congegno in questione. Un decodificatore di indirizzi comunica alla CPU la sorgente di ciascun segnale. Quando viene rilevato un cambiamento, la CPU agisce di conseguenza (nel caso illustrato modificando la visualizzazione dell'ora in un orologio digitale). Per i segnali in uscita il processo è analogo

smettere grosse quantità di dati (come nel caso di una stampante) si usa un apposito chip che funziona da interfaccia I/O. Questo tipo di chip semplifica il progetto del computer, in quanto il circuito dell'interfaccia è quasi tutto contenuto in un unico chip. Uno dei più diffusi è il 8255 PPI (Programmable Peripheral Interface ossia interfaccia programmabile per periferiche): un chip a 40 piedini, che contiene 3 porte di I/O a 8 bit. Ciò significa che ci sono 24 piedini di ingresso/uscita, 8 per ogni porta di I/O. Ogni porta consente la trasmissione di 8 bit (l'equivalente di un byte) alla volta ad una periferica come la stampante, oppure la ricezione di altrettanti bit da una periferica in input (per esempio, dalla tastiera).

Per inviare un dato di 8 bit alla stampante, la CPU, per prima cosa seleziona l'indirizzo del chip PPI, al quale, subito dopo, invia il dato di 8 bit attraverso il "bus dei dati". Il dato viene quindi contemporaneamente depositato in una celletta di memoria interna al chip, chiamata "registro". Il chip PPI, successivamente, rende disponibile il dato sugli opportuni piedini di I/O. Un sistema analogo, ma diretto in senso inverso, permette di depositare un dato, proveniente da un congegno esterno di input, in un altro registro del chip. Il dato viene poi collocato sul "bus dei dati", quando la CPU invia un segnale di richiesta. Come si può dedurre, non è possibile, per i congegni esterni, depositare i dati direttamente e continuamente sul "bus dei dati" del computer: il "bus dei dati", infatti, serve principalmente per lo scambio di dati con la memoria o con altri congegni di I/O e questo processo deve svolgersi nel massimo ordine. Il chip di I/O, perciò, conserva temporaneamente i dati e li trasferisce sul "bus dei dati" (per essere letti dalla CPU) solamente quando la CPU stessa ne fa richiesta.

A questo punto ci si potrebbe chiedere come faccia la CPU a sapere quando un congegno esterno ha dei dati pronti per il trasferimento: esistono due diversi metodi.

Nel primo, la CPU interrompe periodicamente il programma in corso di esecuzione e dà un rapido sguardo a tutte le porte d'ingresso. Se trova che ci sono dei dati in attesa di essere ricevuti, impartisce alla porta le opportune istruzioni per il trasferimento dei dati attraverso il "bus dei dati". Questo processo di "interrogazione" dei congegni di input, che avviene quasi istantaneamente, si indica col termine "polling" (che in inglese significa, appunto, "interrogazione").

L'altro metodo si chiama "ad interruzione" ("interrupt"): il congegno esterno, appena ha un dato disponibile, emette un comando di "interruzione" direttamente alla CPU, la quale interrompe momentaneamente l'esecuzione del programma, per leggere il dato dalla porta in questione. Avremo modo di illustrare, più avanti nel nostro corso, i rispettivi vantaggi e svantaggi dei due sistemi.

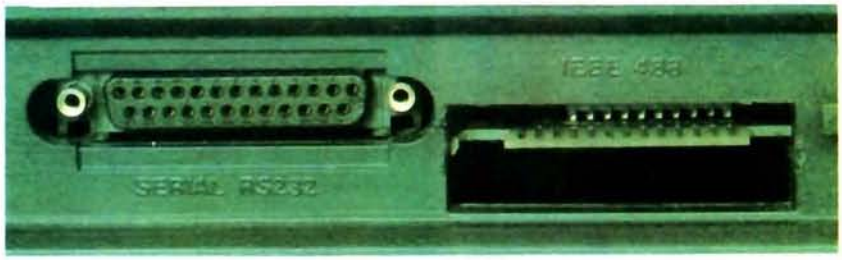
Il tipo di I/O che abbiamo descritto fino ad ora si chiama "parallelo", poiché i dati vengono ricevuti o inviati in gruppi di 8 bit alla volta (un byte), usando 8 linee di I/O.

Esiste un altro tipo di I/O, chiamato "seriale", in cui le informazioni vengono trasmesse o ricevute sequenzialmente un bit alla volta. Alcune stampanti usano interfacce seriali ed anche il trasferimento di dati mediante "modem" (vedere a pag. 108) è di tipo seriale. Il maggior pregio della comunicazione di tipo "seriale" è che bastano due soli fili, invece degli 8 necessari per quella "parallela".

Porte parallele e seriali

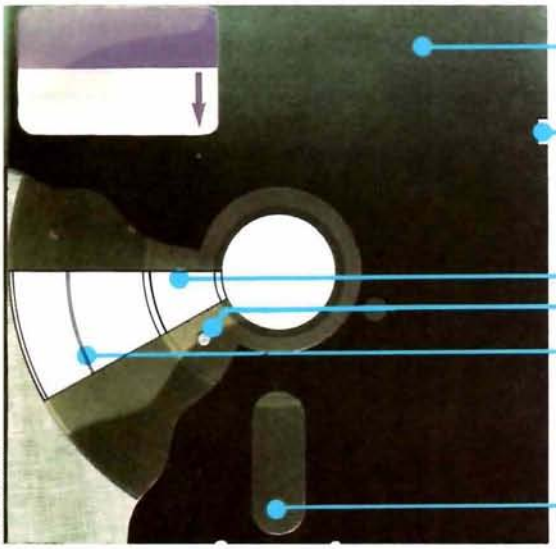
La maggior parte dei computer attuali dispone sia di porte seriali che di porte parallele. Le prime scambiano segnali ad un bit per volta, le seconde in gruppi di 8 bit (un byte) alla volta. Per il tipo più diffuso di porta seriale, la RS232C, viene impiegato un connettore sub-miniaturizzato a 25 poli (come quello a sinistra in figura) o più raramente una presa DIN, simile a quelle usate negli impianti HI-FI.

Le porte parallele generalmente seguono lo standard IEEE488, sviluppato dalla Hawlett Packard ed adottato dall'Institute of Electrical and Electronics Engineers statunitense



Le unità a dischi

I dischetti consentono maggiori velocità di elaborazione



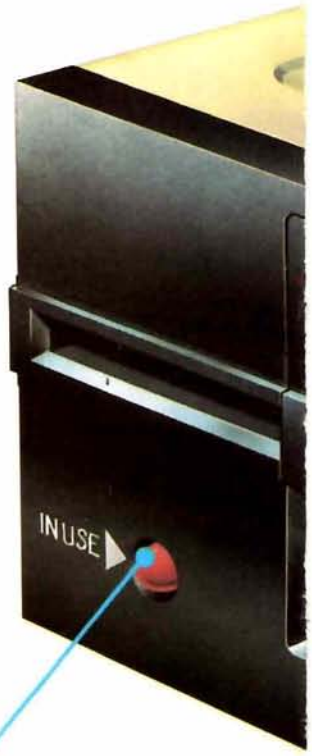
DAVID WEEKS

Il "floppy disk"

La superficie di un dischetto "formattato" è suddivisa in un numero di zone separate chiamate "tracce", che vengono suddivise a loro volta in settori. In un computer quale l'"APPLE II", per esempio, ciascuna traccia è suddivisa in 16 settori, ognuno dei quali possiede una propria zona dati ed una zona indirizzi. Il sistema operativo del computer accede ai singoli settori della traccia usando la zona indirizzi, che contiene, appunto, i rispettivi indirizzi della traccia e del settore, oltre ad un codice di controllo, che serve per verificare se l'utente sta usando il disco giusto. Questo sistema è molto simile a quello usato dal computer per accedere, grazie all'"indirizzamento", ai dati contenuti nella memoria interna

Scheda analogica

Questo circuito converte i segnali digitali generati dal computer in segnali analogici utilizzabili dalla testina di registrazione e viceversa



Lampadina spia

Questo LED indica se l'unità dischi è in funzione

Mozzo

Questo congegno serve per assicurare la centratura del disco e la sua corretta rotazione durante l'uso

Un computer si "dimentica" di tutto quello che è stato programmato nella memoria RAM, quando l'apparecchio viene spento. Ciò è, nel migliore dei casi, un fatto piuttosto spiacevole, ma può risultare un vero disastro se il lavoro di programmazione di un'intera giornata sparisce per sempre. Per ovviare a questo inconveniente, sono stati progettati dei sistemi per conservare l'intero contenuto della memoria in modo permanente. In genere, questi dispositivi si avvalgono di un supporto magnetico sul quale il programma è inciso "digitalmente" mediante una serie di frequenze audio (vedere pag. 94).

Tuttavia quando si tratta di archiviare o richiamare programmi molto lunghi o una serie di programmi di frequente impiego, viene persa una notevole quantità di tempo per ritrovare e "caricare" il programma dai nastri. Ciò dipende da due motivi: il primo è dovuto al fatto che occorre riavvolgere completamente il nastro per poter localizzare l'inizio del programma registrato (in questi casi un registratore con contagiri è di grande aiuto).

Il secondo motivo risiede nel sistema col quale il programma è conservato. Le varie sequenze di bit conservate in memoria devono essere convertite in altrettante serie di toni audio: una frequenza alta corrisponde ad un bit di valore 1 (ON), mentre una frequenza bassa rappresenta un bit di valore 0 (OFF). La velocità ottimale per il trasferimento su nastro è di 150 byte al secondo. Usando una velocità maggiore aumentano le possibilità di errore e, conseguentemente, ciò si ripercuote sull'attendibilità dei risultati.

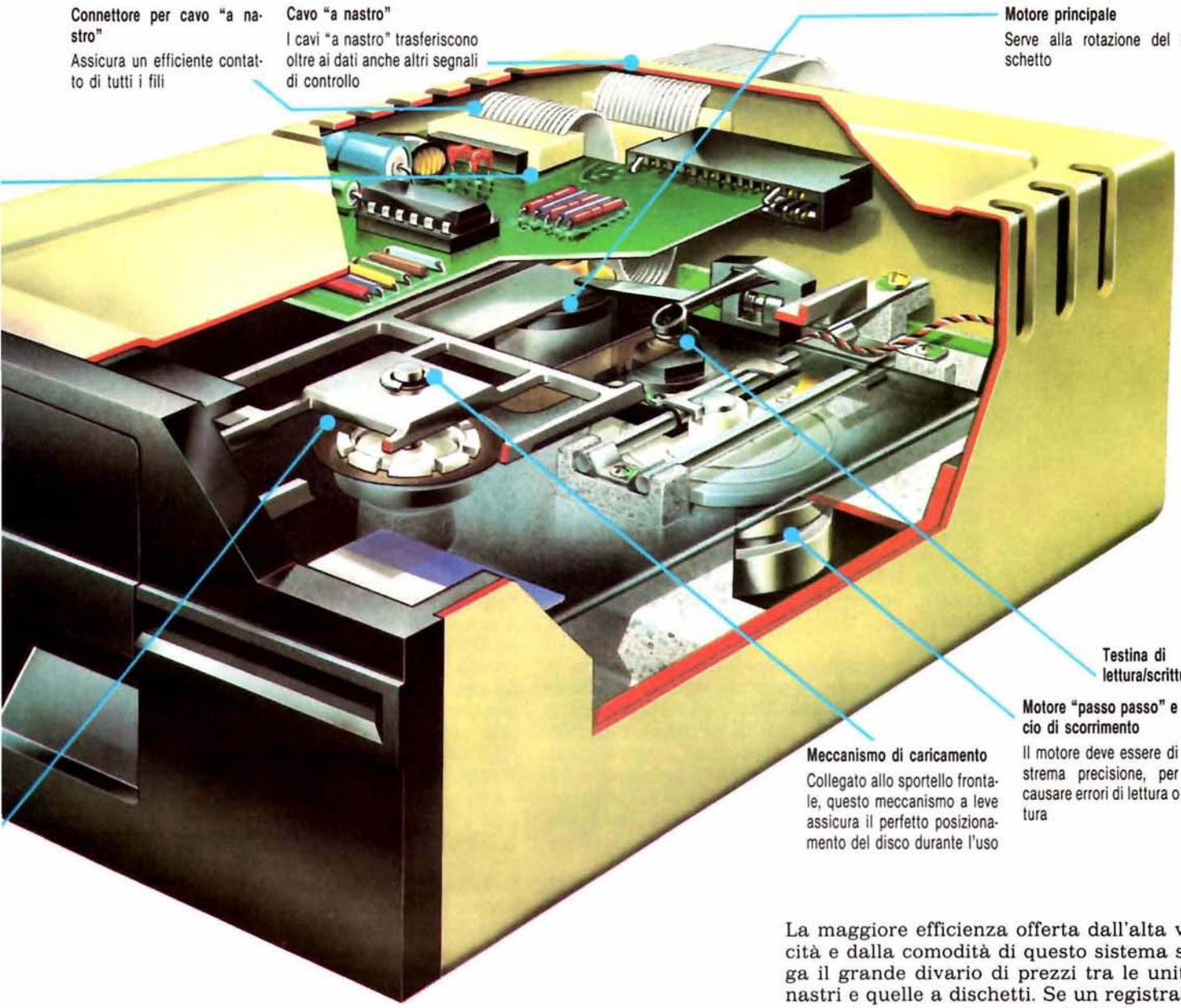
Un simile metodo è assolutamente inadeguato per applicazioni che vadano oltre i semplici giochi. Si ricorre quindi all'impiego di unità a dischetti (o "floppy disk") attualmente disponibili anche su quasi tutti i piccoli computer. Se immaginiamo di ridistribuire tutto il contenuto di un nastro "a cassetta" sulla superficie di un disco rotante, dal diametro di circa 5 pollici, si può intuire come sia più veloce localizzare qualsiasi informazione conservata sul disco. I "floppy disk" vengono inseriti nell'unità di lettura/scrittura (il "disk drive"), la cui funzione è quella di far ruotare il dischetto (dentro la sua custodia) ad una velocità costante: la lettura e la scrittura avvengono per mezzo di una testina di registrazione, di dimensioni ridottissime. La testina è in grado di muoversi avanti e indietro sulla superficie del dischetto in rotazione, diversamente da ciò che accade in un registratore audio, dov'è, invece, il nastro a scorrere avanti e indietro sulla testina.

Un'altra differenza sostanziale risiede nel metodo di registrazione dei dati. Sul nastro magnetico essi sono, in pratica, una lunga successione di byte. Un dischetto, invece, viene "formattato" prima dell'uso. L'operazio-



Come aver cura dei dischetti

I "floppy disk" sono piuttosto delicati e devono essere maneggiati con molta cura. Occorre attenersi scrupolosamente alle istruzioni della ditta produttrice





per concessione della NEWBURY DATA

Testina di lettura/scrittura
Questa è un'immagine fortemente ingrandita della testina di registrazione per dischetti. È molto simile a quella di un registratore "a cassette", ma le sue dimensioni sono microscopiche

ne di "formattamento", per la quale viene impiegato un apposito programma, consiste nel suddividere la superficie del dischetto in vari "settori" concentrici (la cui lunghezza è, in genere, di 256 byte). Ciascun "settore" ha un proprio "indirizzo", registrato in una speciale zona dello stesso dischetto, chiamata "directory" (in italiano viene usato il termine "indice" o, più raramente, "direttorio").
Quando si memorizza un programma sul dischetto, per prima cosa la testina si sposta sulla "directory", da dove, tra tutti i nomi e gli indirizzi dei programmi e degli archivi dati (files) ivi conservati, estrae le informazioni necessarie all'operazione. Nel caso che il nome del programma esista già, il contenuto precedente viene cancellato ed al suo posto viene registrato il nuovo programma. Se invece il nome non è presente nella "directory", i dati identificativi vengono inseriti nell'indice e la prima sezione libera del dischetto viene utilizzata per la memorizzazione del programma.

La maggiore efficienza offerta dall'alta velocità e dalla comodità di questo sistema spiega il grande divario di prezzi tra le unità a nastri e quelle a dischetti. Se un registratore a cassette può costare circa 50.000 lire, un'unità dischi può costare anche fino a dieci volte tanto.
Il fattore che forse influisce maggiormente sul prezzo delle unità a dischi è l'alta precisione richiesta nella loro costruzione. La testina di lettura/scrittura, talmente piccola da essere quasi invisibile a occhio nudo, deve essere "allineata" perfettamente, con tolleranze che vanno intorno ai centesimi di centimetro. Il meccanismo che muove questa microscopica testina impiega un motore "passo passo", in grado, cioè, di ruotare con precisione anche in frazioni di grado, collegato ad un braccio che trasporta la testina e la muove sulla superficie del dischetto a scatti calcolati minuziosamente.
È necessario l'impiego di congegni elettronici molto complessi sia per assicurare che il disco ruoti a velocità costante sia per ridurre l'effetto di eventuali sbalzi termici. Inoltre, il "disk drive" deve essere costruito in modo da proteggere l'unità da eventuali vibrazioni, che sono dannosissime al funzionamento.

Altrimenti, un altro sistema ricorre all'uso delle istruzioni READ ... DATA:

```
10 DIM A(5)
20 FOR X = 1 TO 5
30 READ A(X)
40 NEXT X
50 DATA 5, 10, 15, 20, 100
```

Si provi questo semplice programma. Impartito il comando RUN, si può adesso controllare il valore di ciascun elemento usando, come prima, PRINT A(1)<CR> o PRINT A(5). Oppure, potremmo inserire qualche linea di programma in più per rendere automatica la visualizzazione:

```
60 FOR L = 1 TO 5
70 PRINT A(L)
80 NEXT L
90 END
```

Si provi adesso ad eseguire il programma. Una prova successiva può consistere nel modificare i valori della frase DATA. I nuovi valori devono essere separati da virgole, esattamente come riportato nella linea 50.
Questo metodo si adatta bene se i valori non devono essere cambiati ogni volta che viene usato il programma, altrimenti quello che impiega le INPUT all'interno di un ciclo FOR-NEXT è preferibile.
Adesso impieghiamo tutte queste nozioni per scrivere un utilissimo programma di riordinamento.
Supponiamo di voler ordinare una serie di numeri per valori crescenti. Ma prima di precipitarsi sul computer a digitare istruzioni su istruzioni, cerchiamo di affrontare logicamente il problema sulla carta, descrivendo le varie fasi con semplici parole.
Partiamo da cinque numeri a caso: 4, 9, 2, 8 e 3. Riorganizzarli, manualmente, in ordine crescente non presenta difficoltà; li esaminiamo uno ad uno e, individuato quello dal valore più basso, lo copiamo da un'altra parte. Poi ripetiamo il processo per i restanti numeri.
Ma il computer ha bisogno di istruzioni ben precise, che seguano una certa logica. Per esempio: confronta il primo numero col secondo; se il primo è maggiore scambia i numeri tra loro, altrimenti passa a confrontare il secondo numero col terzo. Dopo l'ultimo confronto, controlla se vi sono stati degli scambi: se non ve ne sono stati, il lavoro è fatto, altrimenti ricomincia da capo.
Questo processo logico funziona con una serie di numeri lunga quanto si vuole. Vediamo cosa accadrebbe con i numeri del nostro esempio:

ordine originale	:	4	9	2	8	3
primo scambio	:	4	2	9	8	3
secondo scambio	:	4	2	8	9	3
terzo scambio	:	4	2	8	3	9

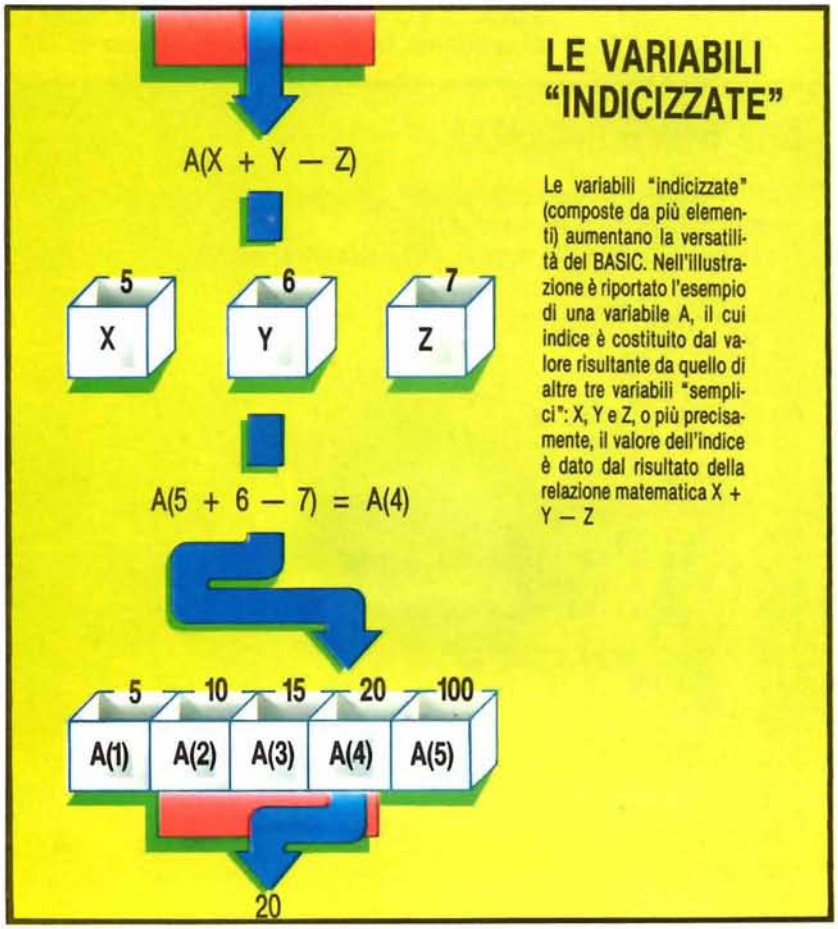
Poiché in questa prima "passata" vi sono stati scambi, il processo ricomincia da capo:

ordine precedente	:	4	2	8	3	9
primo scambio	:	2	4	8	3	9
secondo scambio	:	2	4	3	8	9

Il processo viene ripetuto ancora una volta, poiché vi sono stati scambi:

ordine precedente	:	2	4	3	8	9
primo scambio	:	2	3	4	8	9

Un nuovo "passaggio", a questo punto, rileverebbe che non vi sono stati più scambi e che quindi il lavoro è finito: infatti, i numeri sono adesso in ordine crescente.
L'impiego di variabili indicizzate permette di implementare una procedura di riordino (chiamata anche "sort") con estrema facilità. Assegniamo i valori usati nell'esempio, rispettivamente, ad A(1), A(2) ... A(5). A questo punto se alla variabile X viene assegnato il valore 1, A(X) conterrà 4 e A(X+1) conterrà 9.
Esaminiamo adesso il listato del programma: la linea 20 assegna ad N il numero massimo di elementi da riordinare (in gergo: da "sortare"). Supponiamo che gli elementi siano 5.
La linea 30 contiene la DIM che fissa la dimensione della variabile A. Poiché N vale 5, tanti saranno gli elementi di A.
Le linee da 40 a 60 costituiscono un ciclo FOR-NEXT per l'immissione dei numeri da "sortare", che possono essere, volendo, anche dei numeri con una parte decimale.
La linea 90 assegna a S il valore 0. Questa variabile viene successivamente usata per



Le variabili indicizzate

I pregi delle variabili con indice, rispetto a quelle semplici

Nel programma per il calcolo del numero di giorni mancanti a Natale, abbiamo incontrato un nuovo tipo di variabile, chiamata variabile "a indice" o "indicizzata". A differenza delle variabili "semplici", quelle "a indice" possono contenere più elementi. Il nome di una variabile semplice può essere composto da un massimo di due lettere (o una lettera ed un numero), anche se alcuni tipi di BASIC consentono di assegnare nomi più lunghi. Esempi di nomi validi per variabili semplici sono: A, B, B1, C3 e R2. Le variabili "indicizzate", invece, hanno un diverso aspetto: A(6), B(12), X(20). L'indice è rappresentato dal numero tra parentesi.
Se abbiamo precedentemente paragonato le variabili semplici a piccole scatole dotate di un'etichetta, quelle "indicizzate" possiamo immaginarle come piccole scatole a più compartimenti. Volendo creare, ad esempio, una variabile con 12 di questi "compartimenti" possiamo farlo, in BASIC, mediante l'istruzione DIM: DIM A(12) indica al BASIC che desideriamo avere 12 elementi nella variabile (indicizzata) A.
Assegnare valori alle variabili semplici non presenta alcuna difficoltà, basta usare le ormai note istruzioni LET o INPUT, ad esempio: LET A=35, LET B1=365, oppure INPUT C3. I valori di ciascun elemento di una variabile indicizzata possono essere assegnati allo stesso modo. Per esempio:

```
10 DIM A(5)
```

crea una variabile indicizzata con 5 elementi. Adesso possiamo assegnare dei valori a questi elementi:

```
20 LET A(1) = 5
30 LET A(2) = 10
40 LET A(3) = 15
50 LET A(4) = 20
60 LET A(5) = 100
```

Per apprezzare la differenza, assegniamo anche qualche valore a delle variabili semplici:

```
70 LET X = 5
80 LET Y = 6
90 LET Z = 7
```

Impartito il comando RUN, non comparirà niente sullo schermo, ma adesso ogni variabile conterrà il valore assegnato. Usando in modo diretto (cioè senza digitare alcun numero di linea) il comando PRINT, possiamo ora esaminare il contenuto delle variabili. Si provi con PRINT X<CR>. Sullo schermo apparirà immediatamente un 5. Poi proviamo con PRINT Y<CR>. Il computer risponderà visualizzando un 6. Se vogliamo controllare il contenuto degli elementi della variabile indicizzata, possiamo digitare PRINT A(1), oppure PRINT A(3), e così via.
Un aspetto molto importante delle variabili indicizzate è che lo stesso indice può essere, a sua volta, una variabile. Proviamo a digitare PRINT A(X). Questa volta sullo schermo comparirà il numero 100!
Se diamo un'occhiata alle linee di programma, si può notare che abbiamo assegnato alla variabile X il valore 5. Quindi, scrivere PRINT A(X) equivale a scrivere PRINT A(5). Che valore otterremmo digitando PRINT A(Y – X)?

L'assegnamento dei valori

Se le variabili sono del tipo "semplice" e sono poche, il metodo più facile per assegnare ad esse dei valori è di usare l'istruzione LET. Ma, poiché gli elementi di una variabile indicizzata possono anche essere molto numerosi, vediamo quali possibilità ci vengono offerte da:

```
10 DIM A(5)
20 PRINT "IMMETTI I VALORI"
30 INPUT A(1)
40 INPUT A(2)
50 INPUT A(3)
60 INPUT A(4)
70 INPUT A(5)
```

Questo metodo è laborioso quanto quello di assegnare valori mediante la LET, sebbene funzioni egregiamente. Ma se conosciamo a priori il numero di elementi, è molto più comodo usare un ciclo FOR-NEXT:

```
10 DIM A(5)
20 FOR X = 1 TO 5
30 INPUT A(X)
40 NEXT X
```

Questo programma, una volta eseguito, solleciterà l'immissione di 5 valori (si ricordi di battere il tasto RETURN dopo ogni valore !).

controllare se vi siano stati degli scambi. Se viene eseguito uno scambio, infatti, a questa variabile viene assegnato il valore 1, come vedremo alla linea 240. Dato il particolare impiego di questa variabile, essa viene spesso indicata col termine "flag" (bandiera).
La linea 100 assegna il numero massimo di passaggi da eseguire nel ciclo: poiché N vale 5, saranno necessari N − 1 = 4 confronti. Al primo passaggio attraverso il ciclo, L varrà 1 e quindi A(L) contiene il valore di A(1) ed A(L+1) contiene il valore di A(2). Al successivo passaggio, il valore di L sarà 2, e pertanto A(L) equivale ad A(2) e A(L+1) ad A(3) e via dicendo.
La linea 110 esegue un controllo per vedere se A(L) è maggiore dell'elemento A(L+1). Il simbolo > indica, appunto, "maggiore di".
Se la condizione è vera, il programma esegue il sottoprogramma che inizia alla linea 200 per effettuare lo scambio dei due valori, altrimenti il BASIC passa alla linea 120, che conclude il ciclo. Dopo quattro passaggi all'interno del ciclo, l'esecuzione passa alla linea 130 dove viene eseguito un controllo sulla variabile S. Se il suo valore è 1, indicando che sono stati effettuati scambi, l'esecuzione viene dirottata alla linea 90 per ripetere nuovamente il ciclo, altrimenti, se il valore di S è 0, i numeri sono stati riordinati ed il lavoro è concluso: non resta che passare alla stampa dei numeri riordinati (linee 140 − 170).
Nella subroutine che scambia la posizione di due elementi è necessario l'impiego di una variabile, che serve da deposito temporaneo durante lo scambio, dopodiché viene assegnato ad S il valore 1 e l'esecuzione torna al programma principale mediante una RETURN.

```
10 PRINT "QUANTI NUMERI SONO DA RIORDINARE"
20 INPUT N
30 DIM A(N)
40 FOR X = 1 TO N
45 PRINT "PROSSIMO NUMERO"
50 INPUT A(X)
60 NEXT X
70 REM
80 REM PROCEDURA DI RIORDINO (SORT)
90 LET S = 0
100 FOR L = 1 TO N − 1
110 IF A(L) > A(L+1) THEN GOSUB 200
120 NEXT L
130 IF S = 1 THEN 90
140 PRINT
150 FOR X = 1 TO N
160 PRINT "A(";X;") = "; A(X)
170 NEXT X
180 END
190 REM
200 REM SUBROUTINE PER LO SCAMBIO
210 LET T = A(L)
220 LET A(L) = A(L+1)
230 LET A(L+1) = T
240 LET S = 1
250 RETURN
```

Esercizi

* Si provi ad introdurre il calcolo della media aritmetica dei numeri immessi. La media si ottiene sommando tutti i valori e dividendo per il numero di essi. Un metodo semplice consiste nell'inserire una GOSUB subito prima della linea 170. La subroutine dovrebbe leggere ogni elemento della variabile indicizzata, sommandone il valore a quello di una variabile che contiene il totale progressivo. Esauriti tutti gli elementi, il risultato deve essere diviso per il numero di elementi.

* Con la modifica di una sola linea, il riordinamento viene fatto in ordine decrescente.

* Questo esercizio è indirizzato soprattutto ai possessori di TI99/4A, poiché le variabili ad indice del BASIC di questo apparecchio non accettano l'uso di variabili come "indice". Come deve essere modificato il programma?

* Il sistema di riordinamento impiegato nell'esempio non è l'unico. Si provi a formularne un altro.

DAL NOME... ALL'ETÀ

Ecco un esempio sull'uso delle variabili indicizzate. Per uscire dal programma digitare un asterisco: *<CR>.
Si potrebbero anche usare nomi di città e numero di abitanti...

```
10 REM *** DAL NOME ... ALL'ETA`
20 READ N
30 DIM A$(N)
40 DIM E(N)
50 FOR X = 1 TO N
60 READ A$(X)
70 READ E(X)
80 NEXT X
90 PRINT "NOME";
100 INPUT B$
110 IF B$ = "*" THEN GOTO 170
120 FOR X = 1 TO N
130 IF A$(X)=B$ THEN GOTO 190
140 NEXT X
150 PRINT "NON ESISTE!"
160 GOTO 90
170 END
180 REM --- VISUALIZZA L'ETA`
190 PRINT "ETA`:";E(X)
200 PRINT : GOTO 90
210 GOTO 90
220 DATA 6 : REM  --- NUMERO DI PERSONE
230 DATA "LUIGI", 15,"MARCO", 19,"GIOVANNI", 22
240 DATA "ANTONELLA",17,"MARTA",14,"LUCIA",18
```

Basic & Basic

IF... THEN

Se il programma deve girare su di uno Spectrum e su di uno ZX81, modificare la seguente linea:
130 IF S = 1 THEN GOTO 90

END

Non è disponibile sullo Spectrum. Modificare la linea 170 in: 170 GOTO 260 ed introdurre la linea 260 REM FINE DEL PROGRAMMA

Il "complemento a due"

Il computer e le moltiplicazioni

Abbiamo già visto come il computer risolve un'operazione di addizione: ora diamo un'occhiata alle moltiplicazioni.
La moltiplicazione di due numeri può anche essere vista come una serie di somme (ad esempio il risultato di 14 × 12, può essere ottenuto con la somma 14+14+14...per 12 volte). Questo sistema venne adottato nelle prime calcolatrici meccaniche. Tuttavia, il metodo può divenire in certi casi alquanto lungo e complicato. È stato così sviluppato un metodo che, pur mantenendo una certa semplicità di fondo, è facilmente applicabile ai computer.
Una moltiplicazione di solito si scrive così:

```
14 x 12
-------
   28
  14
-------
  168
```

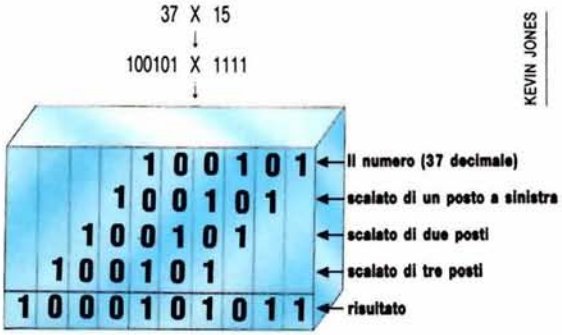
Il metodo funziona anche se la numerazione è in base 2 (o binaria):

```
101 x 11
--------
  101
 101
--------
 1111
```

Il sistema vale anche per numeri più grandi, ma, per adesso, ritorniamo all'esempio di 12 × 14, ma in notazione binaria:

```
(14 x 12 = 168)
1110 x 1100
-----------
    0000
   0000
  1110
 1110
-----------
 10101000
```

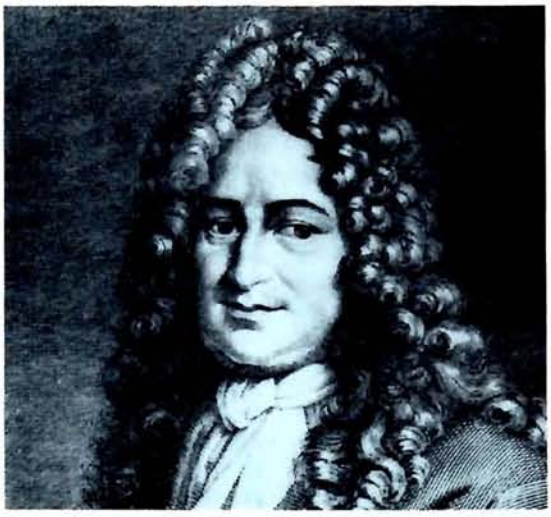
La moltiplicazione in numeri binari è molto più semplice rispetto a quella con numeri in base dieci: se si moltiplica un numero qualsiasi per 1 il numero originale rimane lo stesso (14 × 1 = 14): se si moltiplica per 0 il risultato è sempre 0 (14 × 0 = 0). Ciò vale per tutti i sistemi di numerazione, siano essi binari, decimali o altro. I matematici, osservando calcoli simili a questi, si sono accorti che i concetti di base sono di un'estrema semplicità: la moltiplicazione in binario consiste di 2 sole operazioni (lo spostamento e l'addizione). Ed in effetti il computer esegue



le moltiplicazioni impiegando proprio queste due operazioni. Per prima cosa incolonna una copia del moltiplicando nella posizione appropriata (a seconda degli 1 e degli 0 del moltiplicatore), poi somma le varie colonne. Il computer ha bisogno di una relativamente grande disponibilità di memoria per eseguire le moltiplicazioni. Nel nostro esempio, un numero binario di 4 cifre (1110) è stato moltiplicato per un altro numero binario di 4 cifre (1100) ed il risultato è lungo 8 caratteri (10101000). Come regola generale, il risultato di una moltiplicazione ha una lunghezza doppia rispetto a quella del numero più grande. Può forse sorprendere il fatto che il computer possa a volte sbagliare il risultato di una moltiplicazione. Il motivo risiede nella mancanza di spazio disponibile per memorizzare i risultati delle operazioni. Se non esiste spazio sufficiente, si verifica una condizione di "traboccamento" (chiamato "overflow") e le cifre meno significative vengono perdute, portando a risultati totalmente errati.

La moltiplicazione per spostamento

L'operazione di moltiplicazione è più semplice coi numeri binari che non con quelli in base dieci. Anche se il metodo è identico, l'operazione risulta più semplice perché vengono usati solo due numeri per la moltiplicazione (0 e 1). Quando si moltiplica un numero per 1, ovviamente, il numero originale rimane invariato. Nell'esempio riportato a lato, dove 100101 (37) è moltiplicato per 1111 (15), il numero 100101 appare 4 volte.
Su ogni riga consecutiva il moltiplicando viene spostato a sinistra di una posizione rispetto alla posizione degli 1 presenti nel moltiplicatore. L'ultima operazione che rimane da fare per ottenere il risultato è la somma di tutte le righe, che dà 1000101011 (555)



Gottfried Wilhelm Leibniz

Leibniz (1646-1716), contemporaneo di Isaac Newton, contribuì notevolmente allo sviluppo della matematica, della scienza e della filosofia. Inventò una macchina capace di fare delle moltiplicazioni e divisioni. Inoltre, considerò la possibilità di usare la matematica binaria su macchine calcolatrici, sebbene il primo accenno a numeri binari sia stato fatto da Francis Bacon nel 1623. Negli ultimi anni della sua vita ebbe delle dispute con Newton riguardo alla invenzione del calcolo infinitesimale

Sir Clive Sinclair



L'intraprendente ingegnere che ha creato lo Spectrum

La popolarità di Sir Clive Sinclair, considerato un genio della moderna elettronica, ha varcato da tempo i confini della Gran Bretagna, dove egli opera da anni e si è procurato un'ingente fortuna.
Si può tranquillamente affermare che questo quarantatreenne ex-giornalista di riviste tecniche sia una delle persone più note nel mondo del computer. Il suo operato è stato spesso paragonato a quello di Henry Ford nel settore automobilistico. Per onorare il suo successo come progettista e produttore del computer più venduto nel mondo, lo ZX81, all'inizio del 1983 egli è stato nominato "Sir" (altissima onorificenza inglese), avendo enormemente contribuito a portare la Gran Bretagna sullo stesso piano degli USA e del Giappone nella produzione di nuove tecnologie.
Sir Clive Sinclair è nato a Londra nel 1940. Finiti gli studi, a 17 anni, dopo aver frequentato il St. George's College a Weybridge, esercitò per quattro anni la professione di giornalista tecnico, dopo di che, nel 1962, fondò una società chiamata "Sinclair Electronics", che produceva e vendeva per corrispondenza scatole di montaggio per radio e amplificatori. Tuttavia la sua vera fortuna è dovuta al progetto particolarmente ingegnoso di due computer, diventati famosissimi, lo ZX81 e lo Spectrum.
Egli è stato il primo produttore di calcolatrici elettroniche a basso costo che apparvero sul mercato nel lontano 1972. Ne esisteva addirittura un modello in oro placcato. Inoltre è stato uno dei primi produttori di orologi digitali con microchip.
La sua innovazione più recente è il televisore a schermo piatto, con uno schermo a 2 pollici e dalle dimensioni paragonabili ad un libro tascabile. Tra non molto potremo trovare questo televisore in vendita ad un prezzo decisamente conveniente.
Al momento attuale sir Clive sta lavorando al suo progetto più ambizioso: la produzione di un'automobile elettrica, in particolare di una macchina economica per uso urbano.
Sir Clive è anche proprietario di una casa editrice e ha scritto lui stesso più di una dozzina di libri, tutti sull'elettronica. Nel 1981 ha fondato una nuova casa editrice chiamata "Sinclair Browne", che pubblica circa 20 libri di vario genere all'anno.
Oltre a ricevere l'onorificenza di "Sir", egli è stato anche nominato "L'uomo d'affari dell'anno" dal giornale inglese "Guardian". L'autorevole rivista "Computing" lo ha nominato "Personaggio del Computer del decennio".
I suoi hobby preferiti sono il teatro e la poesia; inoltre è membro del consiglio di amministrazione della "Cambridge Symphony Orchestra". Divide il suo tempo fra la sede della sua azienda a Cambridge ed il suo ufficio a Londra.
Sir Clive ha dei sentimenti molto patriottici e fa di tutto per convincere i suoi concittadini che anche gli inglesi sono altrettanto capaci quanto gli americani o i giapponesi nel campo industriale.

1962
Clive Sinclair costituisce la "Sinclair Radionics", a Londra, per la vendita di scatole di montaggio per radio e amplificatori

1972
Sinclair produce la prima calcolatrice tascabile, l'Executive, dal costo inferiore alle 150.000 lire. Il suo volume di affari è valutato intorno a 5 miliardi di lire

1975
Sinclair lancia sul mercato uno dei primi orologi digitali, che egli chiama il "Black watch" (l'orologio nero). Purtroppo, a causa di deficienze nell'approvvigionamento dei chip, l'affare si risolve con enormi perdite

1976
Sinclair riceve sostanziosi finanziamenti per produrre la sua televisione tascabile, lanciata sul mercato dopo un anno dedicato al progetto

1979
Sinclair forma una nuova società, la "Sinclair Research", per lo sviluppo di apparecchi di largo consumo

1980
La sua nuova ditta immette sul mercato il suo primo prodotto: il computer ZX80, il primo ad essere venduto a meno di 300.000 lire

1981
Sinclair sviluppa lo ZX81, un altro computer, venduto in più di due milioni di esemplari in meno di due anni

1982
Allo ZX81 viene affiancato il più versatile Spectrum

1983
Arriva, finalmente, la microunità a dischi per lo Spectrum, assieme alle interfacce I e II, per l'utilizzo di cartucce ROM e l'allacciamento a reti di comunicazione. Viene inoltre annunciato il televisore a schermo piatto, dopo quattro anni di progettazione

1984
È stato annunciato, per aprile, il lancio dello ZX84: un computer per applicazioni commerciali. Probabilmente esso impiegherà un tipo di schermo piatto simile a quello usato per il televisore Sinclair

La fabbricazione dei chip

I chip vengono prodotti usando una delle sostanze più comuni al mondo

Il silicio è la sostanza più diffusa sulla superficie del nostro pianeta dopo l'ossigeno, con il quale si combina per formare la silice. Chi trascorre le proprie vacanze al mare, vive a stretto contatto con questa sostanza: la sabbia. Sembra incredibile, ma la recente rivoluzione microelettronica è basata proprio sull'impiego della... sabbia.

L'importanza del silicio nell'industria microelettronica è dovuta alla particolare struttura fisica di tale elemento. Allo stato puro, il silicio non è un buon conduttore elettrico, ma ricorrendo a speciali procedimenti, che introducono un certo grado di impurità, la sua struttura elettrica può essere modificata, cosicché esso diviene un "semiconduttore". La conduzione elettrica di una sostanza è determinata dal numero di elettroni presenti negli strati esterni di ciascun atomo e dal tipo di legame che questi hanno con il nucleo. Nei metalli, il flusso di corrente è possibile grazie al precario equilibrio del legame elettronico: gli elettroni sono liberi di vagare all'interno della struttura atomica, trasferendo così, la loro carica elettrica da un atomo all'altro. Al contrario, in un isolante l'equilibrio degli elettroni è molto stabile e la corrente elettrica non riesce a fluire.

La produzione di silicio puro è un processo molto semplice: la prima fase consiste nel raffinare chimicamente il prodotto grezzo fino ad ottenere una sostanza pura al 99,99%. Durante la seconda fase, che deve avvenire in un ambiente saturo di gas inerte sterile per evitare l'introduzione di sostanze estranee, il materiale viene riscaldato fino al punto di fusione (1.410 gradi centigradi).

Il processo col quale si introducono quantità prestabilite di "impurità" (indicato col termine "drogaggio") serve a combinare il silicio puro o col fosforo, ottenendo il silicio di tipo "n" (che permette il passaggio alle cariche negative) o col boro, ottenendo il tipo "p" (che consente il flusso di cariche positive).



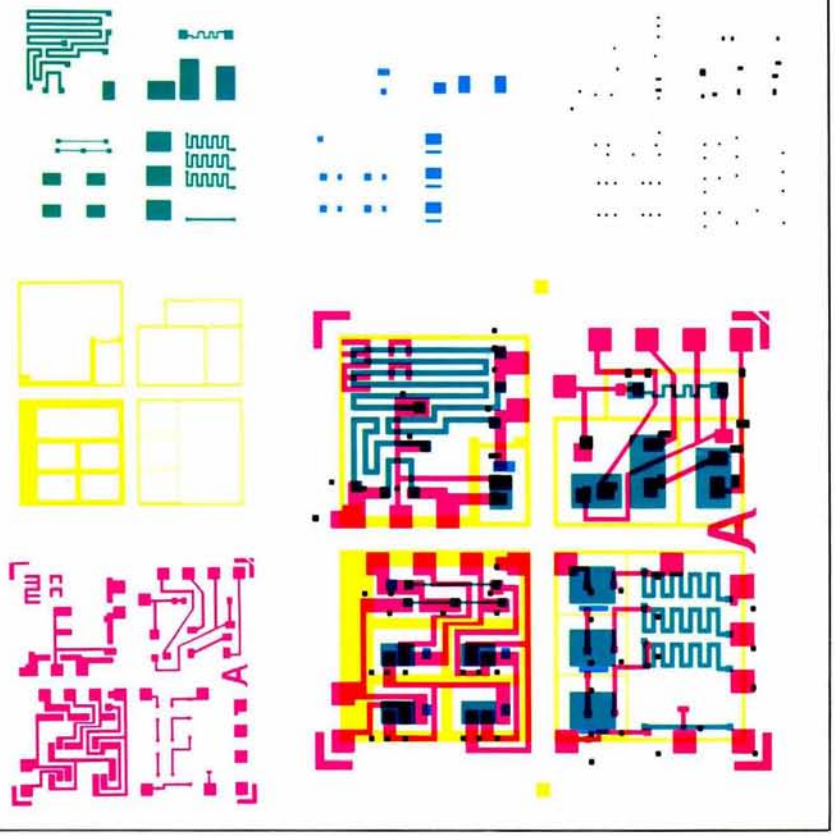
per concessione della FERRANTI

Un'infornata di chip!

La foto mostra un forno per la produzione di chip di silicio. Dopo che il silicio è stato raffinato, sezionato, levigato, mascherato e inciso chimicamente, la sua superficie viene ricoperta da uno strato di biossido di silicio. Ciò avviene mediante un riscaldamento dei "wafer" a 1050 gradi centigradi, seguito immediatamente da un'esposizione ad un flusso di ossigeno puro o di vapore surriscaldato. Mentre i chip sono ancora nel forno, dentro vasche di quarzo, sulla superficie dei "wafer" si forma uno strato di biossido di silicio, che viene successivamente rimosso durante il processo di incisione. L'intera procedura è ripetuta più volte, a seconda della complessità del circuito nel chip

GLI STRATI DEL CHIP

Il processo per la produzione del circuito integrato prevede che ciascun strato del circuito venga inciso separatamente sulla superficie del "wafer" di silicio. Questa tecnica è molto simile a quella adoperata nella stampa delle illustrazioni a colori di questa pubblicazione. In primo luogo, viene eseguita una "selezione" dei colori della fotografia, scomponendola in quattro diverse lastre (tre colori più il nero). I quattro colori sono poi stampati sovrapponendo con precisione le "selezioni".
Il processo per la fabbricazione del chip utilizza strati di silicio "drogato" ed altri materiali al posto dell'inchiostro, ma la procedura necessaria è molto più complicata di quella usata per la stampa delle "quadricromie". L'illustrazione mostra come le singole "maschere", ciascuna relativa ad uno strato del circuito, vengono sovrapposte in sequenza, al fine di costruire il microcircuito: in questo caso si tratta di un semplicissimo circuito transistorizzato



La produzione di cristalli di silicio avviene introducendo nella materia fusa un "seme" di cristallo puro, che viene poi fatto riemergere imprimendogli, contemporaneamente, una lentissima rotazione. Con questo metodo è possibile produrre cristalli del diametro di 7-10 centimetri e lunghi 60 o più centimetri. I cristalli vengono poi lavorati per ottenere un diametro di misura regolare (normalmente tra i 76 ed i 100 millimetri) e quindi tagliati in sezioni, opportunamente levigate. Il "wafer" (così viene chiamato il prodotto finale di questa lavorazione) ha uno spessore tipico di circa mezzo millimetro.
Poiché il processo è relativamente semplice e la materia prima così abbondante, è logico chiedersi come mai il silicio usato nei chip è così caro (un wafer può costare anche qualche decina di migliaia di lire). L'alto prezzo è motivato dall'assoluta necessità di escludere sostanze estranee (eccetto quelle volutamente introdotte durante il "drogaggio") nel wafer; perciò l'aria, negli impianti di fabbricazione, deve essere estremamente pura, con meno di 3.000 particelle per metro cubo. Ciò significa una sterilità 100 volte maggiore di quella impiegata nei moderni ospedali.
La fabbricazione di un circuito integrato avviene mediante un metodo di microincisione della superficie del chip. Nella produzione su larga scala ciò è ottenuto con un processo chiamato "fotolitografia", simile a quello usato per stampare le illustrazioni a colori di questa pubblicazione.
Ciascuno strato del circuito è trattato, durante l'intero processo, come un'entità separata. I disegni dei circuiti, eseguiti con un computer, vengono successivamente riprodotti su pellicola fotografica e quindi ridotti a dimensioni infinitesimali. A questo punto si costruisce una "maschera" per la microincisione del wafer.
Il wafer viene portato ad una temperatura di 1.050 gradi centigradi, in un'atmosfera di ossigeno puro. Ciò provoca la formazione di uno strato superficiale di biossido di silicio, che agisce da isolante. Questo strato isolante viene accuratamente lavorato in modo da formare delle "finestre" sul silicio puro sottostante. Questo processo è ripetuto più volte sui vari strati durante la fabbricazione del circuito integrato.
La superficie ossidata del wafer viene quindi trattata con una sostanza "fotoresistente", la cui solubilità diminuisce proporzionalmente al tempo di esposizione ai raggi ultravioletti. Tra la fonte di raggi ultravioletti e la superficie così trattata, viene interposta la "maschera" che riproduce una porzione del circuito. Dopo l'esposizione ai raggi ultravioletti, sulla superficie viene passato uno speciale solvente per rimuovere la sostanza fotoresistente dalle parti che, essendo "mascherate", non sono state esposte ai raggi. Un metodo molto simile è usato per eseguire incisioni su rame. La superficie della lastra viene, in primo luogo, ricoperta con uno strato uniforme di cera. La cera viene poi "graffiata" a seconda

del disegno da riprodurre, e quindi messa a bagno in un acido, che corrode soltanto le parti non protette dalla cera. Dopo un lavaggio per rimuovere la cera residua, il disegno risulta inciso sulla superficie del rame.
Ma torniamo al nostro circuito integrato. Completato il primo ciclo di operazioni, l'intero processo viene ripetuto con maschere e reagenti diversi fino ad ottenere il circuito definitivo, completo in ogni sua parte. Sull'intera superficie del wafer, a questo punto, viene steso un nuovo strato di biossido di silicio. A volte è necessario ripetere le operazioni di copertura, "mascheramento", esposizione ed incisione anche dieci o più volte. Le possibilità di commettere errori durante una delle tante fasi di lavorazione sono molte, tant'è vero che nelle fasi di collaudo e di verifica sulla qualità vengono scartati numerosissimi pezzi. Più è complesso ed intricato il microcircuito, maggiore è la densità dei singoli elementi all'interno del wafer e, di conseguenza, più alta è l'incidenza di errori e di scarti.
Una prima verifica viene fatta sull'intero wafer, inserendolo in un impianto collegato ad un computer che controlla ogni singolo circuito integrato tra le centinaia presenti sul wafer. Spesso, sul wafer, vengono riservate nove piccole aree, che servono esclusivamente per i test di efficienza.
Questo test non solo indica ogni errore con una particolare macchia di inchiostro, ma prende anche nota della frequenza dei difetti in ciascun wafer, individuandoli sia nell'intero wafer che nel singolo chip.
Terminata questa verifica, i singoli chip vengono tagliati dal wafer: quelli difettosi sono eliminati manualmente, mentre quelli che risultano perfetti vengono montati su microscopici supporti metallici. Con un processo computerizzato si collegano le uscite del chip (usando fili sottilissimi, quasi invisibili) ai piedini metallici che serviranno per le connessioni esterne. Il tutto viene poi sigillato in contenitori di plastica o di ceramica, dai quali fuoriescono soltanto i piedini di collegamento. Il grado di miniaturizzazione attualmente ottenibile, che dipende in massima parte dal processo fotolitografico della produzione, è più o meno pari alla lunghezza d'onda della luce, ossia circa due milionesimi di metro.
Tra non molto, grazie a nuove tecnologie basate sull'impiego dei raggi X, dovrebbe essere possibile ridurre del 50% le attuali dimensioni dei circuiti microelettronici.
Paradossalmente, quasi tutti gli aspetti della produzione dei chip (la fase di progettazione, la preparazione del prototipo, la produzione su larga scala e il test finale) sono talmente complesse, che sarebbero praticamente impossibili senza l'ausilio di un computer!
L'uso di un computer nella fase di progetto permette di richiamare (e quindi riutilizzare) sottoprogetti precedentemente memorizzati. Facciamo un esempio: nella produzione dei chip di memoria RAM, ciascun bit richiede l'impiego di una singola cella a transistor. In una memoria da 2 Kbyte, il chip deve dunque contenere 16.384 celle identiche. Il progettista può limitarsi a definire per una sola volta la struttura di una cella, chiedendo poi al computer di ripetere il medesimo "campione" 16.384 volte.
Ma, in fase di progetto, l'uso del computer non si limita a questo utile e semplice intervento. I disegnatori tecnici non lavorano più su un tavolo da disegno, ma davanti ad un terminale video, adoperando penne ottiche per "disegnare" direttamente sul monitor. Il disegno esecutivo finale è prodotto direttamente dal computer per mezzo di un plotter multicolore.
E le applicazioni non si fermano certo al settore grafico: è anche possibile creare dei "modelli di simulazione" (come nel caso dell'impianto per l'estrazione di olio grezzo, descritto a pag. 103). Un circuito elettronico può essere "simulato" al computer, offrendo al progettista la più ampia libertà d'azione nella ricerca della soluzione più adatta a risolvere un particolare problema. Si possono così evitare i considerevoli sprechi di tempo e di denaro legati alla produzione ed al collaudo di prototipi reali.
Quando si parla di "rivoluzione elettronica" s'intende l'enorme salto compiuto nella produttività, nella diminuzione dei costi e delle dimensioni degli impianti. Può essere utile ricordare, a questo punto, le caratteristiche dei vecchi computer a valvole:

notevole ingombro
tempi di elaborazione lenti
elevato fabbisogno elettrico
memoria e istruzioni limitate
alti costi di fabbricazione

Su tutti questi fattori influì positivamente l'invenzione del transistor (vedere a pag. 46), sebbene, in termini di produttività, la fabbricazione dei computer rimanesse ancora un'attività costosa. Molto lavoro, in particolare il montaggio dei componenti sui circuiti stampati, doveva essere eseguito a mano.
È stata la successiva invenzione del circuito integrato o, più precisamente, del microprocessore che ha permesso all'industria di trarre enormi vantaggi dalle nuove tecnologie disponibili per la produzione e la lavorazione computerizzate.
Alcune statistiche sono sbalorditive: nel 1959, per esempio, era possibile montare un solo componente su un circuito integrato (un diodo oppure un transistor). Nel 1978 i circuiti integrati a larga integrazione (LSI) potevano contenere più di 250.000 componenti su di un singolo chip. In un lasso ancor più breve di tempo (dal 1973 al 1983) il costo, per bit, delle memorie è diminuito di venti volte, mentre l'impiego di componenti elettronici di ogni tipo su scala mondiale è incrementato 1000 volte. Probabilmente, questo andamento continuerà negli anni a venire. Alcune stime indicano che il numero di apparecchi elettronici in uso aumenterà del 100% all'anno, durante i prossimi 3 anni.



per concessione della TEXAS INSTRUMENTS

Un innovatore della microelettronica

Jack Kilby è generalmente riconosciuto come l'inventore del circuito integrato. Mentre era alle dipendenze della Texas Instruments, nel 1958, egli costruì un circuito, composto di alcuni transistor, tutto contenuto in una "confezione" (package) di un centimetro di larghezza e poco più di un centimetro di lunghezza. I circuiti elettronici moderni contengono, in un medesimo spazio, l'equivalente di centinaia di migliaia di componenti

Database: chi cerca... trova

L'uso di un archivio elettronico permette i più ampi criteri di ricerca delle informazioni

Col termine inglese "database" si indica una raccolta di dati e di informazioni memorizzate e gestite da un computer. Capita spesso di consultare dei "database" non computerizzati: uno di questi è l'elenco telefonico, nel quale i nominativi sono riportati in ordine alfabetico. Nei "database" gestiti da un computer, però, non è necessario, anzi può essere addirittura limitativo, conservare le informazioni secondo un unico ordine logico (quello alfabetico, per esempio).

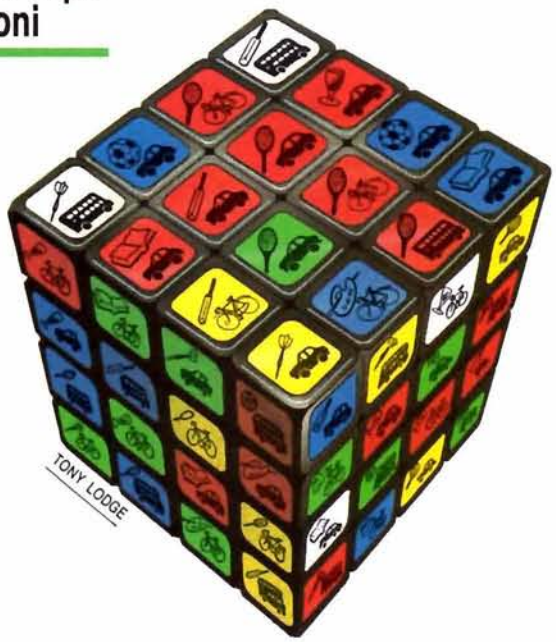
I programmi impiegati per la gestione dei database (i cosiddetti "database manager") consistono in una serie di regole, che permettono all'utente l'estrazione di parte dei dati secondo criteri selettivi. Esistono più livelli di automazione in questi programmi: si va dalla gestione di semplici "schede", fino a sistemi che impiegano speciali linguaggi.

Di solito un database computerizzato deve essere abbastanza capiente per contenere una quantità di dati di vario genere: ciò non comporta, necessariamente, l'acquisto di un grosso elaboratore. Qualsiasi computer è in grado di utilizzare un database, poiché l'unica vera limitazione sta nella capacità e nella velocità della memoria.

Per fare un esempio pratico, potremmo stilare un elenco di conoscenti (chiamandolo AMICI) completo di varie informazioni a loro relative. Utilizzando un normale schedario manuale, si otterrebbe qualcosa di simile a una rubrica personale. Ovviamente, le informazioni contenute in queste "schede" hanno caratteristiche diverse: alcune sono costituite da semplici parole e altre da numeri. Esistono, poi, dei casi particolari: il sesso della persona può essere o maschile o femminile e il suo stato civile può corrispondere a "coniugato", "celibe/nubile", "divorziato" o "vedovo/a".

Potrebbe essere utile raggruppare alcuni dei dati in liste di parole o di numeri, ad esempio: la professione, il nome della ditta per cui lavora la persona, l'indirizzo e il numero telefonico della ditta e il nome del diretto superiore, potrebbero essere tutti archiviati sotto il titolo LAVORO, mentre la marca e l'anno di fabbricazione dell'automobile posseduta li potremmo annotare sotto il titolo AUTO.

Seguendo questo criterio, si potrebbero creare elenchi separati per ciascun elemento





Un caos organizzato

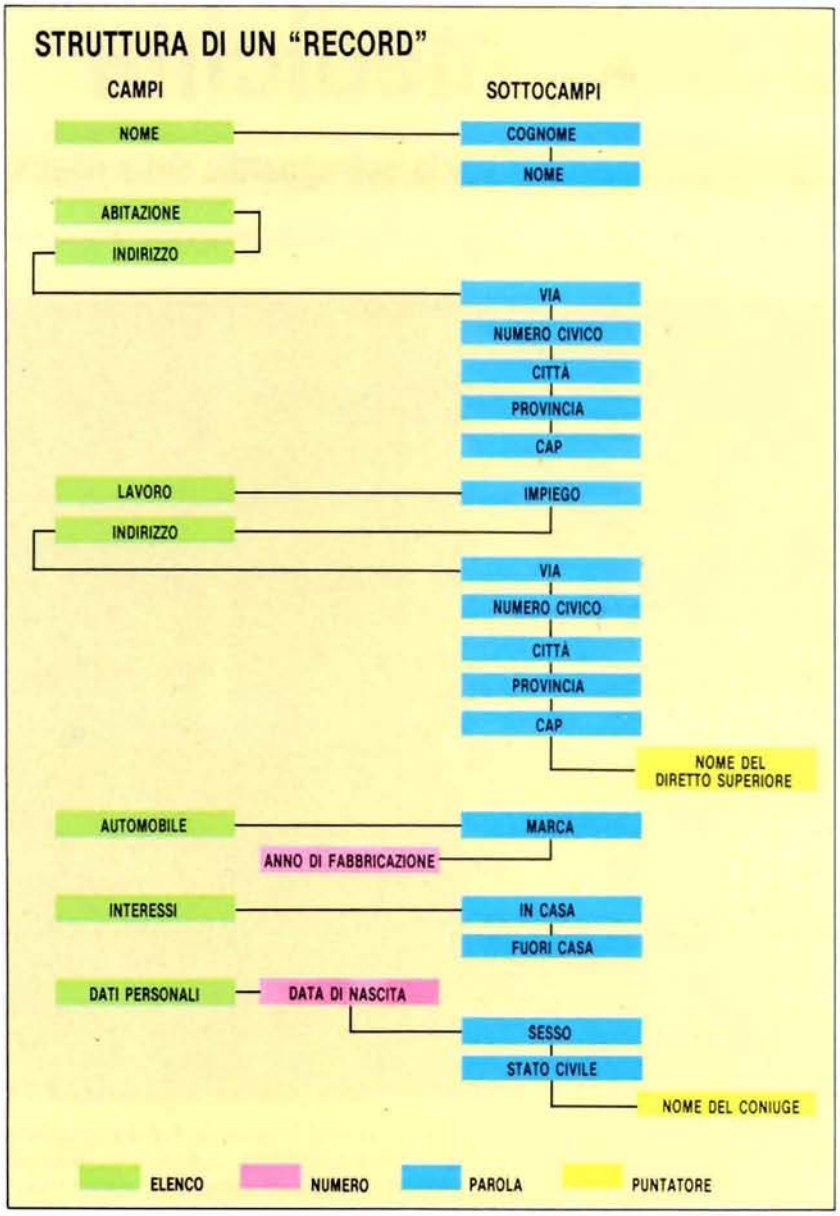
Abbiamo usato un cubo di Rubik per rappresentare un "database". Inizialmente le informazioni sono conservate alla rinfusa: tutte possono esserci utili, ma soltanto secondo la combinazione giusta. Ogni faccia del cubo rappresenta una persona e contiene uno o più simboli che indicano una sua particolare situazione o interesse. Volendo selezionare, fra tutti, coloro che possiedono un'auto, che giocano a tennis e che sono disponibili per una certa data (quadratino rosso) il computer, idealmente, riorganizza il cubo in modo da presentarci su di una sola facciata tutti i quadratini rispondenti alla particolare combinazione





LA RICHIESTA D'INFORMAZIONI

La maggior parte dei programmi per la gestione dei "database" adopera comandi e codici che costituiscono quasi uno speciale linguaggio. L'esempio riportato è abbastanza tipico. La parola ACCESS indica al programma che desideriamo "interrogare" un archivio (file) precedentemente compilato. La parola USE indica, invece, che intendiamo usufruire di un "sottoinsieme" dell'archivio intitolato AMICI e che vogliamo estrarre informazioni relative a tutti i nominativi sulle cui schede compaiono le parole AUTO e TENNIS. Il risultato è un elenco di nominativi, completi di numero telefonico, che soddisfa le condizioni imposte per la selezione





L'organizzazione delle informazioni

Un database strutturato sistematicamente conduce l'utente da un'informazione ad un'altra, offrendo più alternative sul percorso da seguire. Un normale schedario manuale, al contrario, permette di accedere a gruppi di informazioni seguendo un solo criterio, generalmente alfabetico

dell'indirizzo (uno per le città, uno per le vie, ecc.): in certe applicazioni, infatti, è necessaria la sola città di residenza, mentre il resto dell'indirizzo non è richiesto. La voce STATO CIVILE può essere ampliata, aggiungendo, eventualmente, il nome del coniuge, ma, per una maggiore completezza, potrebbe essere utile inserire un richiamo alla scheda personale del coniuge. Ma, poiché ogni scheda ha una collocazione ben precisa nell'archivio, ad essa può essere assegnato un numero di codice, ed usare questo, invece del nome, per creare un archivio a "tiro incrociato". Il numero di codice viene spesso chiamato anche "puntatore". Questa tecnica di accesso non modifica la struttura delle schede, ma permette una maggiore celerità di ricerca.

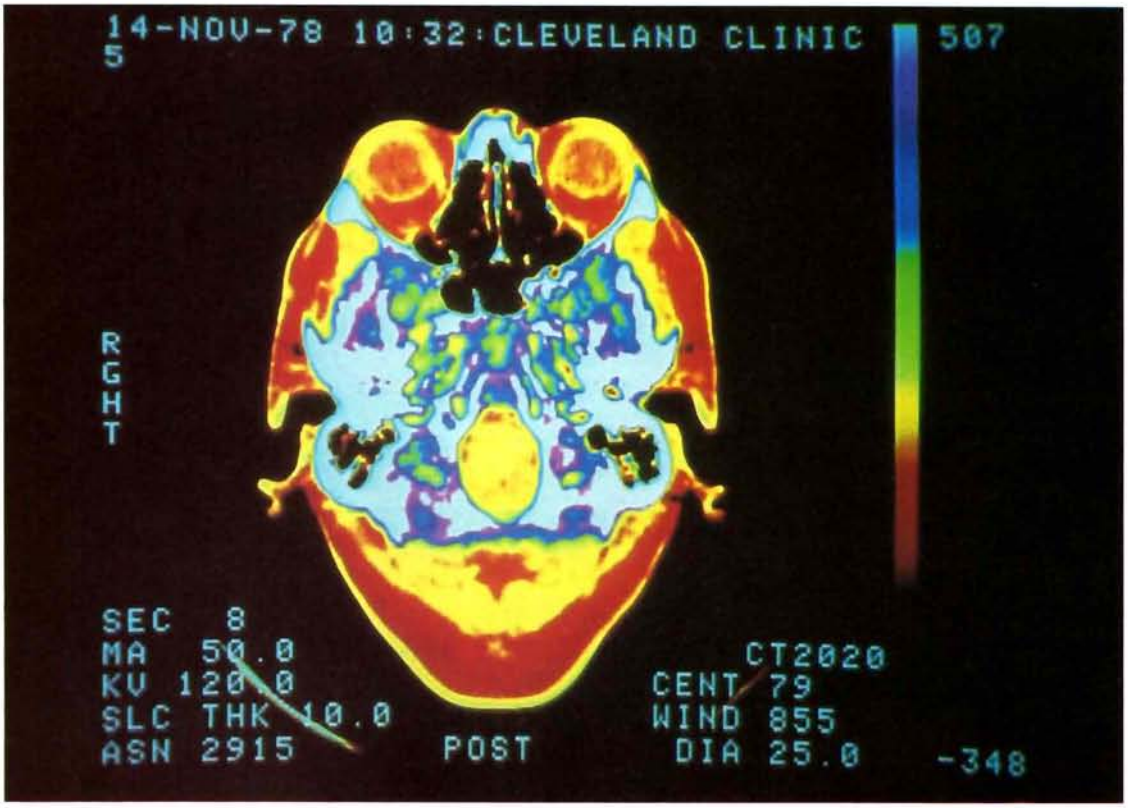
La principale differenza tra uno schedario manuale ed un database computerizzato è che al primo possiamo dare un unico tipo di ordinamento (di solito quello alfabetico). L'ordine alfabetico va benissimo per effettuare una ricerca servendosi, ad esempio, del cognome di una persona. Le cose si complicano, però, nel caso che si desideri estrarre un elenco delle persone che lavorano per una stessa ditta: in uno schedario manuale, occorre, infatti, esaminare ciascuna scheda, annotandosi i vari nominativi. Un tale processo può richiedere molto tempo, ma, soprattutto, esistono forti probabilità di omettere qualche nome o di fare degli errori.

Usando, invece, un sistema computerizzato, possiamo affidare questo compito al computer, ottenendo un elenco stampato dei nominativi che ci interessano. Alternativamente, si potrebbe chiedere al computer di riordinare le schede, in base alla professione di ogni persona. In questo caso, il database conterrebbe gli stessi dati di prima, ma strutturalmente riorganizzati per un particolare tipo di accesso: il processo di riordinamento raggrupperà tutte le schede nelle quali compare, alla voce LAVORO, il nome di una certa ditta. In altre parole, un comune schedario manuale ha una sola "chiave di ricerca", ad esempio il cognome di una persona, mentre in un database computerizzato si ha la facoltà di scegliere, fra tutte quelle che in esso compaiono, una "voce" da usare come "chiave" per operare un riordino dell'intero archivio oppure una selezione di dati da stampare.

Esistono anche dei programmi molto sofisticati che, oltre a permettere la gestione di un archivio su più "chiavi", sono in grado di produrre statistiche, conteggi ed altre utilissime funzioni. Questi "database manager" permettono al computer di selezionare i "record" (così si chiamano le "schede" computerizzate) secondo criteri del tipo: «stampa tutti i nominativi delle persone nate tra il 1945 ed il 1949, in grado di parlare almeno due lingue straniere, e che possiedano un'automobile». Appare evidente che i vantaggi di simili sistemi, in tutti i campi, ma in particolare in quelli del lavoro e della ricerca, sono enormi. Oggi, anche con computer di piccole dimensioni questi strumenti sono alla portata di tutti.

I micro e la medicina

Le numerose applicazioni del computer per la salvaguardia della nostra salute



L'esame al TAC

Le lastre a raggi-X producono un'immagine bidimensionale, nella quale tutti gli organi sono sovrapposti e solo un esperto è in grado di interpretarle. Con la tomografia assiale computerizzata (TAC), invece, viene eseguita una scansione (detta "scanning") mediante un sottile fascio di raggi-X. I dati così raccolti vengono memorizzati ed elaborati dal computer, producendo precise immagini, che offrono un quadro molto più completo e più facilmente interpretabile.
Anche se è possibile avere immagini a colori (come quella riprodotta), molti radiologi preferiscono quelle monocrome, affermando che queste ultime offrono una rappresentazione più immediata della densità relativa dei tessuti

Nel settore medico, come in tante altre professioni, capita spesso che il personale specializzato sia costretto a dedicare molto del proprio tempo a lavori relativamente semplici o ripetitivi: anche in questo campo, l'uso dei microcomputer ha portato enormi benefici.
Da quando sono stati introdotti i primi microprocessori, ci sono stati notevoli cambiamenti nel lavoro ospedaliero, in particolare nei reparti di rianimazione. Il microcomputer viene impiegato per sorvegliare le funzioni vitali dei pazienti: la pressione del sangue, le pulsazioni, il ritmo respiratorio e quello cardiaco vengono tenuti costantemente sotto controllo e si memorizza, al tempo stesso, il decorso clinico dell'ammalato. Ciò permette al personale medico di dedicarsi tranquillamente ad altri pazienti, poiché, nel caso dovessero insorgere situazioni anomale, il computer automaticamente richiederebbe l'intervento dei medici.
I computer trovano largo impiego anche nel settore amministrativo degli ospedali (e delle Unità Sanitarie Locali in genere), per la gestione anagrafica e contabile dei pazienti, e per il controllo delle scorte di farmaci.
Abbiamo già visto, in parte (a pag. 72 e 73), alcune applicazioni importanti del computer nel campo della medicina, ma vale la pena di esaminare più dettagliatamente quali vantaggi possono trarre i medici dall'impiego degli expert system.
Uno di questi sistemi, non molto sofisticato, ma che sta riscuotendo grande successo nei paesi anglosassoni, è il "Mickie", che fornisce al medico un quadro generale delle condizioni fisiche del paziente. In realtà il "Mickie" è un expert system "inesperto", in quanto non è in grado di fornire una diagnosi esauriente, ma solamente delle indicazioni molto generiche.
Le domande che il sistema pone al paziente, sono sempre formulate in modo da permettere risposte del tipo: "sì", "no" oppure "non lo so". In quest'ultimo caso, vengono fatti vari tentativi, mediante elementari spiegazioni e domande supplementari, per aiutare il paziente nella risposta.
Invece di una tastiera completa, il paziente adopera una scatola con solo quattro tasti, opportunamente contrassegnati. Il funzionamento di questo sistema, e di altri simili, è piuttosto lento, poiché dipende dal tempo impiegato dai pazienti a fornire le varie risposte. Il prossimo passo verso degli

expert system più completi, in questo ramo, probabilmente consisterà nella possibilità di aggiungere e confrontare i nuovi dati acquisiti con informazioni fornite in precedenza sul decorso clinico del paziente: se dovesse risultare che il paziente lamenta gli stessi disturbi avuti in precedenti occasioni, la diagnosi potrebbe essere emessa rivolgendo una sola domanda al paziente, per esempio: «I sintomi sono gli stessi di quattro anni fa?» Ci sono altri vantaggi offerti dalla computerizzazione delle cartelle cliniche, che forse non saltano subito agli occhi: per esempio, la segretezza. Difatti, è molto più difficile leggere una serie di dati contenuti su di un dischetto magnetico, che non su di un fascicolo di carta, pur ammettendo che si sappia far funzionare il computer.

Tuttavia gli sviluppi più significativi sono stati fatti nel campo della diagnostica. Fino a poco tempo fa, gli unici metodi disponibili per fare alcuni esami interni erano relativamente pericolosi se non, talvolta, addirittura distruttivi: le radiografie a raggi-X, per esempio, comportano dei rischi dovuti ad eccessiva esposizione alle radiazioni, mentre l'endoscopia (l'inserimento di un tubo per l'esplorazione visiva di un organo interno) può causare dei traumi ai fragili tessuti interni e non si dimentichino i casi in cui, in passato, era necessario ricorrere ad interventi chirurgici esplorativi.

Un notevole passo avanti, verso una maggiore sicurezza d'uso ed accuratezza di risultati, è rappresentato dallo sviluppo della tomografia computerizzata (il TAC). Questo metodo, invece di esporre estesamente le parti del corpo alle radiazioni, impiega un sottile fascio di raggi-X per eseguire una rapidissima scansione della parte da esaminare. Anche l'uso di altri sistemi, quali, ad esempio, gli ultrasuoni e la risonanza magnetica nucleare, fanno sperare nella possibilità di eseguire esami interni in modo completamente innocuo.

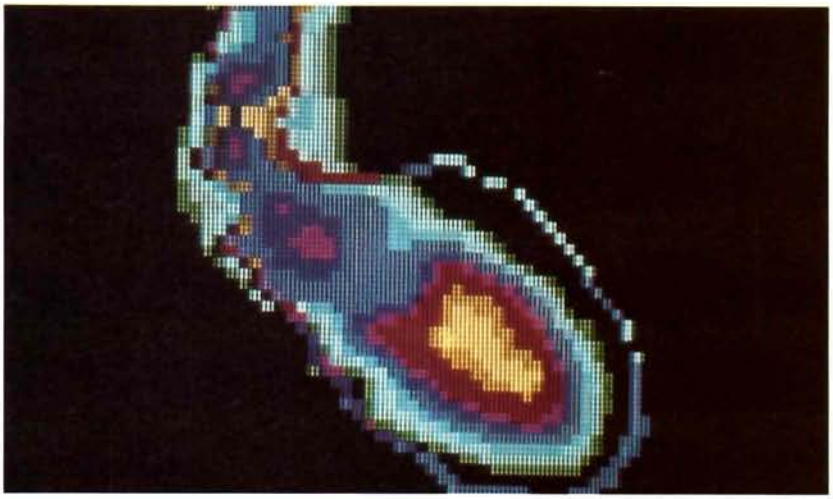
Anche la gestione delle situazioni d'emergenza si avvale dell'opera dei computer: in certi casi la vita di una persona dipende dalla celerità con la quale si ottengono alcune importanti informazioni. L'uso di archivi elettronici, rispetto a quelli tradizionali, offre, in questo senso, incalcolabili vantaggi.

Grandi passi in avanti si sono avuti anche nella cura degli handicappati fisici e mentali. Questo settore è ancora in gran parte inesplorato e si prevedono grandissimi sviluppi a breve termine. Si immagini, per esempio, la gioia di un paraplegico muto, in grado di "parlare", per la prima volta nella sua vita, grazie ad uno speciale apparecchio che, sfruttando il movimento degli occhi, mette in funzione un sintetizzatore di voce.

Vediamo, brevemente, il funzionamento di questa apparecchiatura: uno speciale "lettore" misura gli spostamenti fatti dall'occhio, mentre la persona muove lo sguardo lungo una fila di lettere. Partendo da un punto di riferimento, è possibile individuare la posizione dello sguardo e "leggere" il carattere che il paziente sta guardando. In una memoria ROM sono contenuti i fonemi necessari alla costruzione delle varie parole, che vengono poi riprodotte da un amplificatore. Se l'handicappato ha una qualsiasi possibilità di movimento, questo può essere sfruttato come base di un sistema di comunicazione computerizzata. Spesso si sostituisce la tastiera normale con un'altra costruita appositamente (per esempio con tasti di dimensioni maggiori del normale).

In alternativa, il computer può presentare all'utente una serie di domande prestabilite ad un ritmo appropriato: per esempio, il menù per il prossimo pranzo. Con la pressione di un solo tasto, al momento opportuno, l'utente comunica al computer la pietanza che egli desidera mangiare. Il computer interpreta come risposta negativa la mancata pressione del tasto. In casi del genere, la tastiera può consistere in un unico tasto, utilizzabile, per esempio, con un semplice movimento della testa.

Sistemi simili a quelli appena descritti offrono all'utente la possibilità di usare dei "word processor", producendo, anche se lentamente, del lavoro scritto.





IAN DOBBIE

Un metodo sicuro

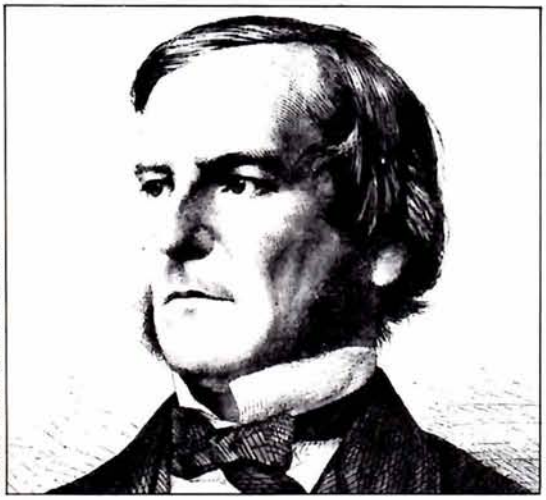
La scansione eseguita mediante NMR (ossia risonanza magnetica nucleare) dà risultati analoghi a quelli ottenuti con la tomografia computerizzata, ma, poiché non vengono impiegati i raggi-X, è molto meno pericolosa per la salute. La tecnologia in questo campo è ancora agli stadi iniziali ed, attualmente, uno "scanning" completo può richiedere anche più di un'ora

Le onde della salute

Le informazioni relative alle condizioni del paziente, ottenute tramite sensori collegati al corpo, vengono elaborate da un microprocessore e, successivamente, presentate sullo schermo di uno speciale oscilloscopio, simile a quelli usati dai tecnici per l'analisi dei circuiti elettronici

Le leggi del pensiero

George Boole e la sua teoria sulla logica matematica



per concessione della ROYAL SOCIETY

George Boole (1815-1864)
George Boole nacque molto prima che il computer elettronico venisse appena immaginato, ma rimane uno dei fondatori della logica matematica usata nei computer dei nostri tempi. Figlio di un calzolaio, egli imparò da sé la matematica nel suo tempo libero. Egli era fermamente convinto che si potessero convertire i processi decisionali in una forma matematica particolare. Nel 1847 pubblicò le sue teorie che lo resero famoso in breve tempo. Gli fu richiesto di diventare primo professore di matematica alla Nuova Università di Cork (Irlanda)

Nel 1815 — l'anno della sconfitta di Napoleone a Waterloo — si verificò un altro avvenimento significativo nella storia dell'Uomo: a Lincoln, in Inghilterra, nacque George Boole. Figlio di un calzolaio, egli divenne in seguito uno dei geni che contribuirono all'invenzione del computer. Anche se morì nel 1864, ben cento anni prima della rivoluzione del microcomputer, quest'ultima non si sarebbe mai verificata senza le sue idee.
Boole capì che i processi di ragionamento fatti dalle persone nel corso della vita di tutti i giorni possono essere descritti in relazione alla logica formale di cui furono pionieri gli antichi greci. Egli fu del parere che si può andare oltre ed esprimere il ragionamento umano in forma matematica. Boole si dedicò a dimostrare proprio questo. Imparò la matematica e cominciò le sue ricerche sulla logica delle decisioni umane.

La teoria degli insiemi

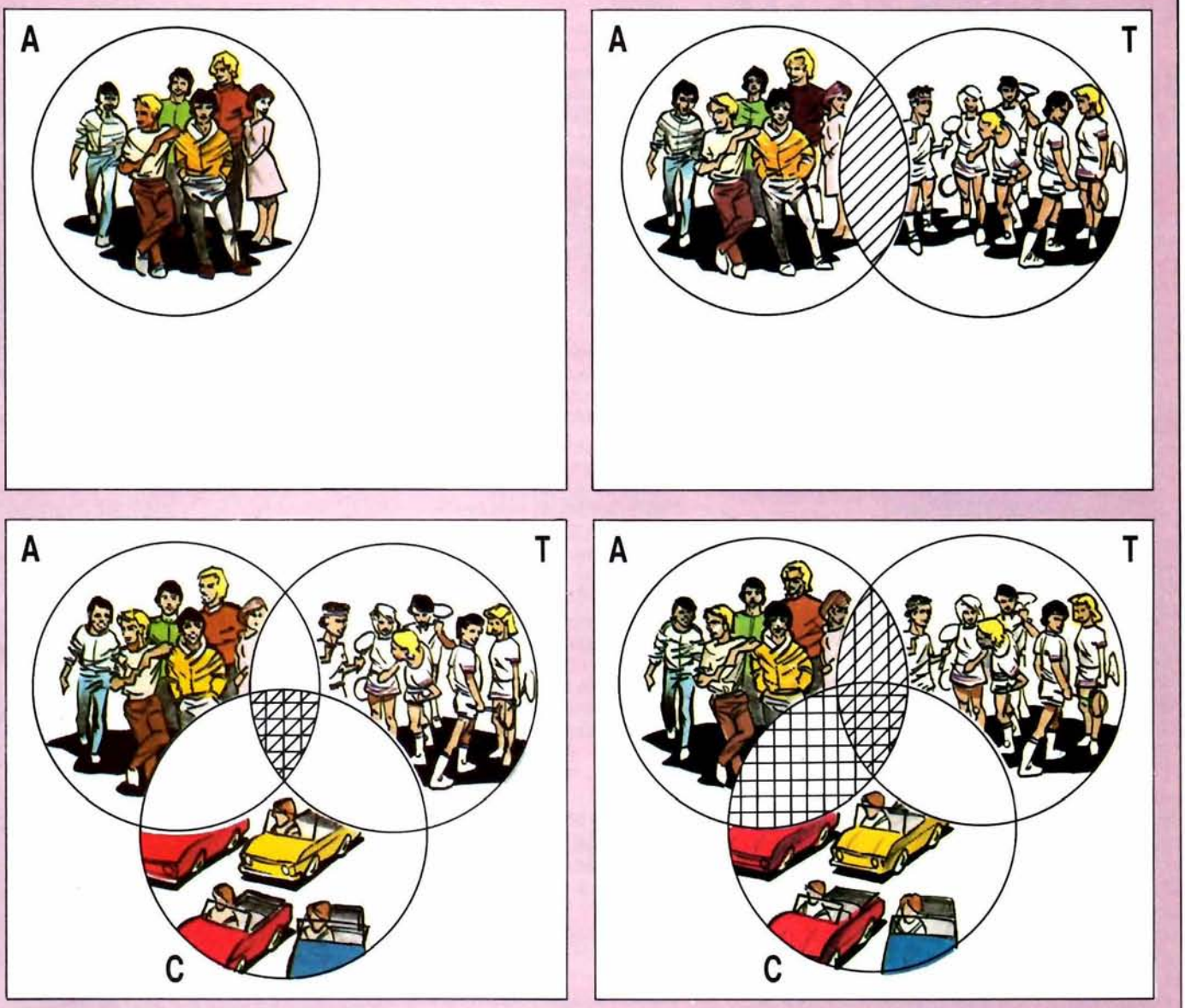
Supponiamo, per esempio, che una persona vada ad una festa con l'intenzione di ballare: entrata nella sala, essa noterà che alcune persone ballano ed altre no. Non è possibile che stiano facendo tutt'e due queste cose nello stesso momento.
La persona dovrà quindi decidere con chi ballare: la scelta potrà essere fatta tra individui di sesso maschile o femminile.
Boole avrebbe ragionato in modo diverso. Egli avrebbe suddiviso le persone presenti in due "insiemi": i maschi (M) e le femmine (F). Altri due "insiemi" avrebbero raggruppato quelli che stanno ballando (B) e quelli che sono ancora in attesa di ballare (A).

La persona con cui egli avrebbe potuto ballare doveva rispondere a due requisiti: appartenere all'insieme delle persone di sesso femminile ed anche a quello di chi è in attesa di ballare. Boole notò l'importanza di questo "anche" (in inglese AND) e lo rappresentò con il simbolo ∩ (una ∪ rovesciata). Avrebbe quindi potuto indicare, matematicamente, che le possibili partner di ballo dovevano necessariamente appartenere all'insieme F ∩ A.
Tuttavia, se si fosse trattato di trovare una persona, non per ballare, ma solamente per fare quattro chiacchiere, egli avrebbe potuto scegliere chiunque fra l'insieme M o quello F, in quanto tutte le persone nella sala sono incluse in questi due insiemi. Di nuovo, egli vide l'importanza di questo "oppure" (in inglese OR) e lo rappresentò col simbolo ∪. Nella sua algebra logica M ∪ F conteneva tutti gli uomini e le donne della sala.
Le "porte logiche" presenti nel computer vengono indicate con i termini AND e OR proprio seguendo la logica di Boole. Anche nel programma BASIC riportato più sotto, sono presenti due comandi molto utili chiamati, appunto, AND e OR. Ma prima di esaminarne il funzionamento, vale la pena di presentare un'interessante interpretazione della logica booleana, concepita dai matematici inglesi John Venn (1832-1892) e Charles Dodgson (1832-1892), noto, quest'ultimo, anche con lo pseudonimo di Lewis Carroll, col quale scrisse *Alice nel paese delle meraviglie*.
Immaginiamo di aver inserito nella memoria di un computer un elenco di nostri conoscenti. Oltre al nome, abbiamo inserito anche altre informazioni: il numero telefonico, gli hobby etc. Un bel giorno decidiamo di giocare a tennis in un campo dall'altra parte della città. A questo punto vogliamo trovare un

IL PROGRAMMA DEGLI AMICI

```
10 DIM N$(10),D$(10),F$(10),T$(10),C$(10)
20 REM NOME, TELEFONO, AMICO?, TENNIS?, AUTO?
30 PRINT "RISPONDERE SECONDO IL FORMATO:"
40 PRINT "NOME, TELEFONO, SI/NO, SI/NO, SI/NO"
20 FOR K = 1 TO 10
30 INPUT N$(K),D$(K),F$(K),T$(K),C$(K)
40 NEXT K
50 REM FASE DI RICERCA DEI NOMI
60 FOR J = 1 TO 10
70 IF F$(J)="SI" AND T$(J)="SI" AND C$(J)="SI" THEN GOSUB 100
80 NEXT J
90 END
100 PRINT N$(J),D$(J)
110 RETURN
```

I DIAGRAMMI DI VENN



Ogni riquadro rappresenta un insieme di tutte le persone elencate dal computer. Generalmente nei diagrammi di Venn questo è definito come un insieme "universale". I cerchi dentro i riquadri rappresentano gli insiemi individuali.
Gli amici sono identificati nel diagramma come le persone appartenenti all'insieme "A". Non tutte le persone giocano a tennis ma l'insieme di quelli che giocano è indicato con una "T". Siccome ci sono persone che giocano a tennis e che, contemporaneamente, sono anche degli amici, i due cerchi si sovrappongono (A ∩ T). L'insieme "C" identifica coloro che hanno l'auto. Questo insieme si sovrappone agli altri due, e le persone che, oltre ad avere un'auto, sono amici e anche giocatori di tennis sono identificati dalla zona comune ai tre cerchi (A ∩ T ∩ C). L'ultimo diagramma rappresenta gli stessi insiemi (A, T e C), ma con diverse condizioni da soddisfare. In questo caso si tratta o di giocare a tennis o di andare a fare una gita in auto. L'insieme contenente gli amici che giocano a tennis e hanno l'auto è indicato con delle linee diagonali, quello contenente gli amici che hanno l'auto è indicato con delle linee orizzontali. La combinazione di queste due aree rappresenta gli amici con cui si potrebbe giocare a tennis o fare una gita in auto

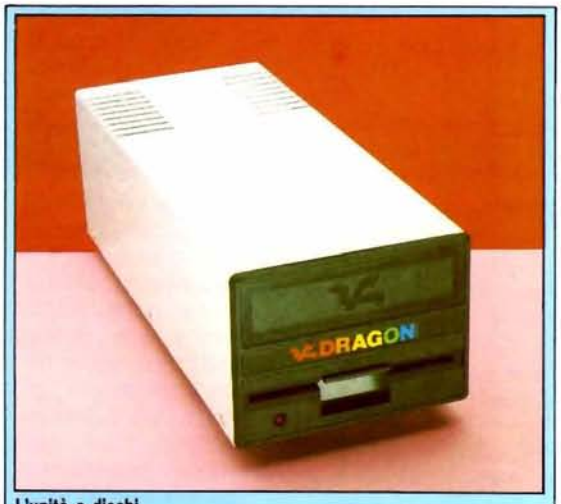
KEVIN JONES

amico (e non soltanto un conoscente) che ami giocare a tennis e, inoltre, possieda un'auto. Ci rivolgiamo al computer per avere il nome ed il numero telefonico di tutte le persone che soddisfino a tutt'e tre i requisiti: amanti del tennis, possessori di un'auto ed amici. Il programma, durante l'immissione dei dati, chiede varie informazioni relative a ciascuna conoscenza: sono amici? hanno l'auto? giocano a tennis? Supponiamo di avere 10 amici, anche se non vi è alcun limite teorico (basta ricordarsi di cambiare il 10 fra parentesi nella frase DIM sulla riga 10); l'elenco viene idealmente "sfogliato", usando le istruzioni IF...THEN alle quali è stata associata una condizione multipla. La massima parte dei BASIC consentono di specificare per le frasi IF...THEN condizioni costituite, a loro volta, da sottocondizioni collegate fra loro da istruzioni AND oppure OR. Quando l'elenco è stato completamente esaminato, sullo schermo saranno visualizzati i nomi ed i numeri telefonici di qualsiasi nostro conoscente, che soddisfa a tutt'e tre le condizioni.
In alcuni programmi si trovano delle combinazioni molto complesse delle funzioni logiche. L'algebra di Boole, considerata poco più di una curiosità ai tempi nei quali venne formulata, ha acquistato la sua vera importanza nell'era del computer.

Il Dragon 32

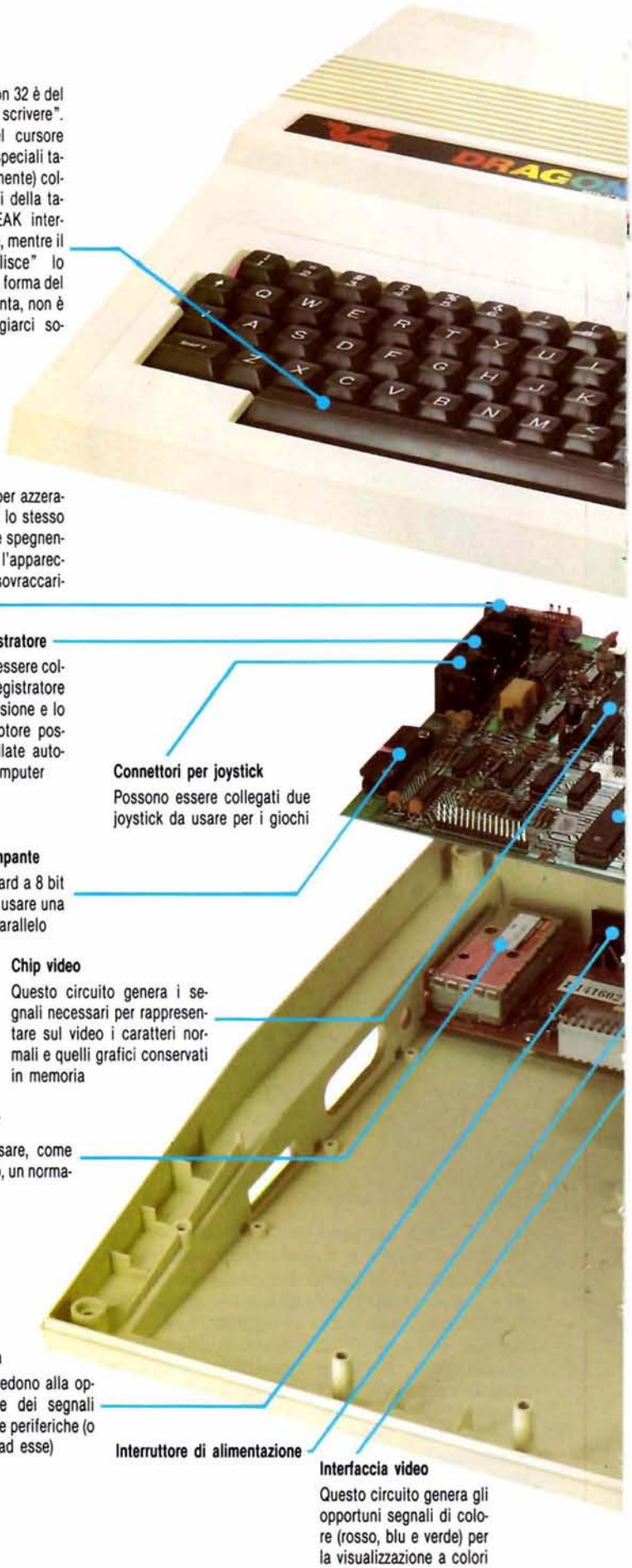
Anatomia di microcomputer di diffusione europea: tecnologie a confronto

Questa volta esaminiamo un computer progettato e realizzato nel Galles (GB) dalla Dragon Data, un'impresa originariamente fondata dalla fabbrica di giocattoli Mettoy. Il Dragon 32 venne commercializzato nel dicembre del 1982, ed ottenne un grande successo, soprattutto per l'allora notevole capacità di memoria (32 Kbyte) e per l'impiego dell'interprete BASIC della Microsoft.
Il Dragon 32 è compatibile, quasi totalmente, con il Tandy Color Computer: è possibile adoperare gli accessori Tandy e alcuni dei giochi su cartucce, mentre non c'è compatibilità... per i nastri a cassette. Ambedue gli apparecchi sono progettati intorno al chip Motorola 6809E, meno diffuso del 6502 o dello Z80.
Un sofisticato gruppo di comandi per la grafica consente un controllo più ampio delle immagini di quello offerto dalla maggioranza degli altri computer della stessa categoria. Alcuni di questi comandi sono: DRAW, CIRCLE, PAINT, COLOUR e MOVE (letteralmente: "disegna", "cerchio", "dipingi", "colora" e "muovi").
Il Dragon ha un generatore di note musicali ad una sola "voce" e non può, quindi, generare degli accordi musicali. La facilità dei comandi, in compenso, rende molto semplice la riproduzione di melodie. Sebbene il Dragon 32 abbia numerose interfacce, sono poche le periferiche disponibili, a parte i joystick.
Una scheda di espansione, per aumentare la memoria a 64 Kbyte, dovrebbe essere presto in commercio, portando l'apparecchio ad un livello simile a quello del Commodore 64.



L'unità a dischi
L'unità è composta da due parti: una scheda per il controllo dei dischi, che deve essere inserita nell'apposito connettore del computer, ed il contenitore del meccanismo di lettura/scrittura

CHRIS STEVENS



Tastiera
La tastiera del Dragon 32 è del tipo "macchina da scrivere". Gli spostamenti del cursore sono comandati da speciali tasti (poco opportunamente) collocati ai due estremi della tastiera. Il tasto BREAK interrompe l'elaborazione, mentre il tasto CLEAR "pulisce" lo schermo. Sebbene la forma del contenitore lo consenta, non è consigliabile appoggiarci sopra il video

Tasto di RESET
Questo tasto serve per azzerare la memoria ed ha lo stesso effetto che si ottiene spegnendo e riaccendendo l'apparecchio, senza, però, sovraccaricare l'alimentatore

Connettore per registratore
A questa presa può essere collegato un normale registratore a cassette e l'accensione e lo spegnimento del motore possono essere controllate automaticamente dal computer

Connettori per joystick
Possono essere collegati due joystick da usare per i giochi

Connettore per stampante
Un'interfaccia standard a 8 bit paralleli permette di usare una stampante di tipo parallelo

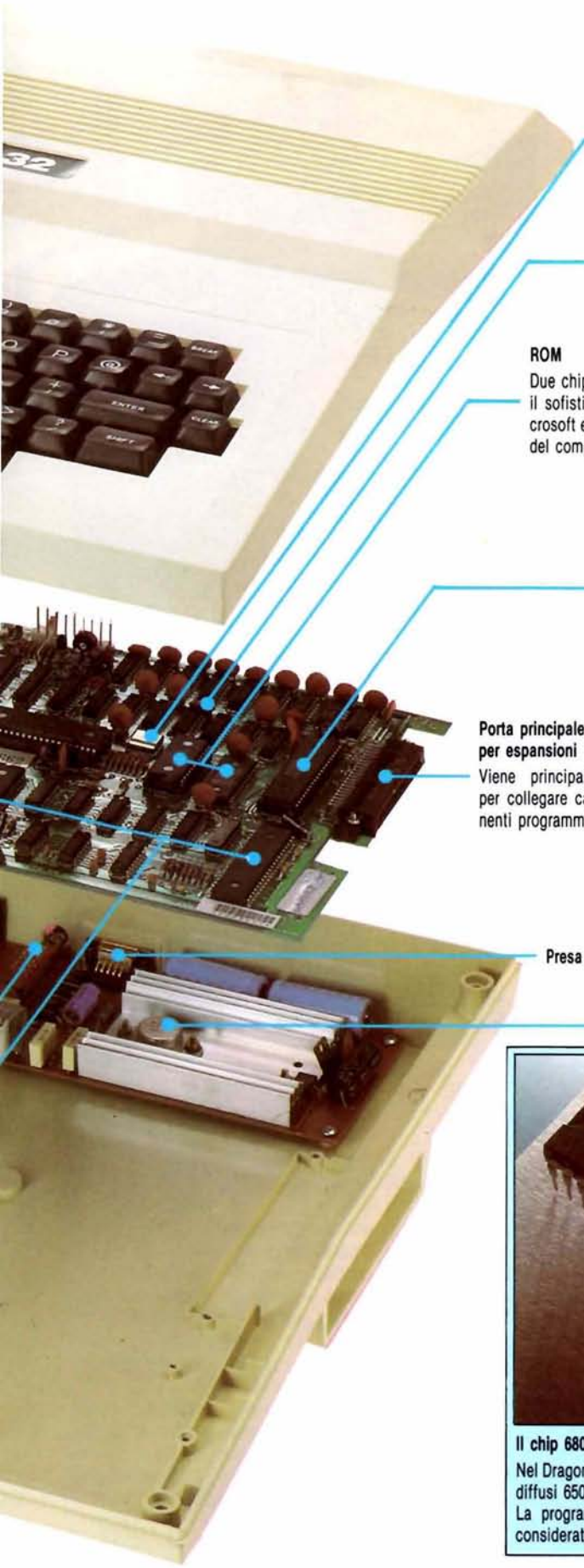
Chip video
Questo circuito genera i segnali necessari per rappresentare sul video i caratteri normali e quelli grafici conservati in memoria

Modulatore RF
Permette di usare, come terminale video, un normale televisore

Circuiti d'interfaccia
Due chip 6821 provvedono alla opportuna conversione dei segnali che provengono dalle periferiche (o che sono destinati ad esse)

Interruttore di alimentazione

Interfaccia video
Questo circuito genera gli opportuni segnali di colore (rosso, blu e verde) per la visualizzazione a colori



Oscillatore a quarzo
I segnali di temporizzazione, necessari al corretto funzionamento del computer e del circuito video, sono ottenuti impiegando cristalli al quarzo

RAM
La memoria standard del Dragon 32 è di 32 Kbyte, ottenuti impiegando 16 chip da 2 Kbyte ciascuno

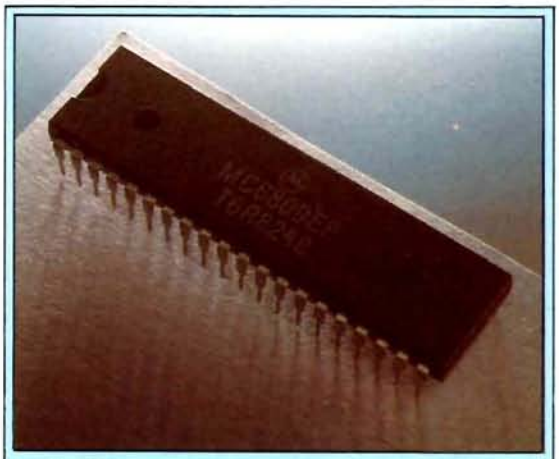
ROM
Due chip di ROM contengono il sofisticato BASIC della Microsoft ed il sistema operativo del computer

Microprocessore
Il Motorola 6809E, impiegato anche nel computer Tandy Color, è più veloce del più diffuso 6502, benché vi sia una scarsità di software per questo chip

Porta principale per espansioni
Viene principalmente usato per collegare cartucce contenenti programmi

Regolatori di tensione
Un trasformatore esterno provvede a convertire la tensione di rete ad una molto più bassa: questo circuito serve per stabilizzare la corrente, evitando dannosi sbalzi

Presa di alimentazione



Il chip 6809E
Nel Dragon 32 è impiegato il microprocessore 6809E, invece dei più diffusi 6502 e Z80.
La programmazione di questo chip in linguaggio macchina è considerata, dai programmatori, molto facile

DRAGON 32

DIMENSIONI
mm. 380 x 325 x 97

PESO
2.100 grammi

CPU
6809E

CLOCK
1 MHz

MEMORIA
32 Kbyte di RAM (in futuro estendibile a 64 Kbyte). 16 Kbyte di ROM contengono il BASIC ed il sistema operativo

VIDEO
16 linee di 32 caratteri o 192 x 256 punti per la grafica ad alta risoluzione. 8 colori disponibili

INTERFACCE
Connettore per TV, connettore per registratore a cassette, connettore per periferiche, per stampante parallela, ecc.

LINGUAGGI FORNITI
BASIC

ALTRI LINGUAGGI DISPONIBILI
Nessuno, per il momento

ACCESSORI COMPRESI
Alimentatore, cavi di connessione per TV, un manuale

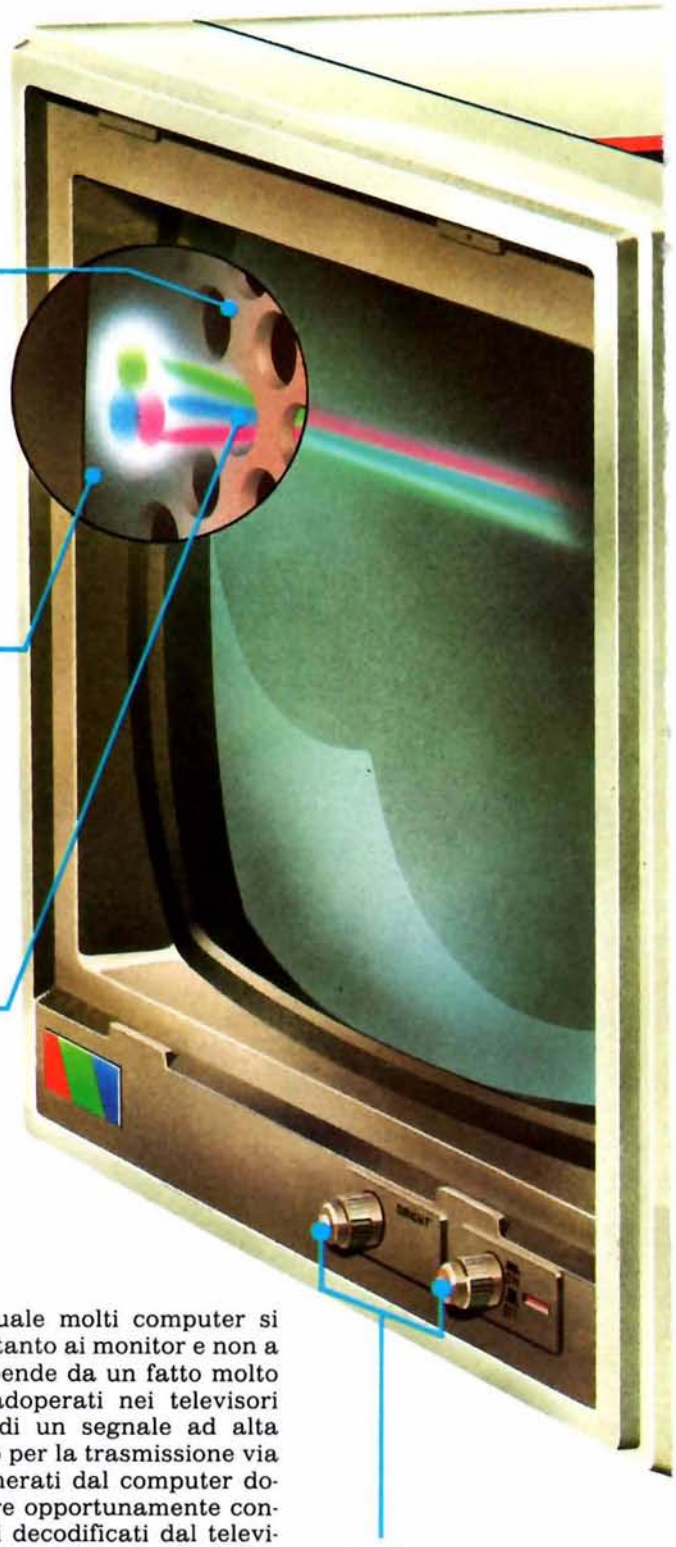
TASTIERA
Del tipo "macchina da scrivere" con 53 tasti

DOCUMENTAZIONE
Il manuale, redatto dal progettista di computer Richard Wadman è di buon livello. Sono riportati numerosi esempi sul funzionamento, assieme a molti validi suggerimenti per l'uso. È molto interessante e completa anche la parte dedicata all'uso della grafica. Sono inoltre disponibili in commercio alcuni libri (in inglese) per chi desideri approfondire la propria conoscenza sull'apparecchio e sul linguaggio usato

Monitor o televisore?

Quale scegliere per il nostro computer

Grazie al sempre crescente impiego di computer e di altri apparecchi elettronici che necessitano di uno schermo video, il prezzo dei monitor sta calando sensibilmente.
Fino a poco tempo fa un buon monitor a colori costava più di un milione e mezzo, mentre oggi si possono trovare ottimi monitor spendendo all'incirca mezzo milione. I monitor monocromi, poi, costano molto meno.
Le capacità grafiche dei microcomputer oggi disponibili (quasi tutti a colori) sono in netta evoluzione e per poterle sfruttare appieno occorre impiegare un monitor a colori. Esistono due tipi di monitor a colori: uno è noto con la sigla RGB (Red, Green, Blue, ossia: rosso, verde, blu), l'altro è chiamato video composito. In un monitor RGB i 3 "cannoni" elettronici che producono i colori vengono direttamente controllati dal computer. Anche gli impulsi, adoperati per sincronizzare il computer col monitor, vengono generati direttamente dal computer.
Servono due tipi diversi di "sincronismi": uno è necessario per ciascuna riga del quadro e l'altro per il quadro completo. Quando la visualizzazione di un quadro è stata completata, al monitor viene inviato un breve impulso affinché il "pennello elettronico", che visualizza le immagini, si riposizioni in alto a sinistra dello schermo, pronto per la successiva immagine.
Un processo simile avviene alla fine di ogni riga dell'immagine, ed il "pennello elettronico" si posiziona all'inizio della successiva riga. In un monitor RGB i segnali relativi ai tre colori, al sincronismo di riga ed a quello di schermo, vengono inviati dal computer al monitor individualmente, su altrettanti cavi.
Il funzionamento di un monitor a video composito, invece, è molto più simile a quello di un normale televisore, poiché tutti i vari segnali vengono combinati in un segnale unico, trasmesso al monitor con l'impiego di un cavo coassiale. I circuiti interni al monitor provvedono a separare ed utilizzare convenientemente i diversi segnali per la riproduzione dell'immagine.
Un monitor può essere considerato come un televisore, al quale siano stati tolti i circuiti di sintonizzazione. In effetti, è teoricamente possibile, aggiungendo un sintonizzatore ad un monitor, utilizzarlo come televisore oppure, togliendo i circuiti per la selezione dei canali, utilizzare un televisore come monitor. Ma simili interventi, oltre ad essere poco pratici, sono anche molto pericolosi, considerate le alte tensioni circolanti nei circuiti di questi apparecchi. Inoltre, l'immagine che si ottiene su di un televisore non ha mai la qualità raggiungibile con i monitor.



La griglia dello schermo
Per far sì che i punti generati dal "cannone" elettronico siano posizionati con precisione, all'interno del tubo a raggi catodici è posta una speciale griglia

Schermo fosforescente
Le immagini colorate sono, in realtà, ottenute dalla combinazione dei tre colori fondamentali. Ciascun punto dello schermo, trattato con speciali sostanze fosforescenti, si illumina quando è colpito dai raggi di elettroni prodotti dai "cannoni"

Raggi elettronici
In un tubo a raggi catodici a colori, esistono tre diversi raggi, ciascuno dei quali influisce su un determinato tipo di fosforo, generando un punto luminoso colorato

La ragione per la quale molti computer si possono collegare soltanto ai monitor e non a normali televisori dipende da un fatto molto semplice: i segnali adoperati nei televisori sono "modulati" su di un segnale ad alta frequenza, necessario per la trasmissione via etere. Gli impulsi generati dal computer dovrebbero perciò essere opportunamente convertiti, per essere poi decodificati dal televisore: ciò comporta l'aggiunta di speciali circuiti che, oltre ad influire sul prezzo, peggiorano la qualità delle immagini.
Nei monitor, al contrario, i segnali non subiscono complicati processi di conversione, e di modulazione/demodulazione, a tutto vantaggio della nitidezza dell'immagine ed, in ultima analisi, della vista dell'operatore.

Controlli
Come nei televisori, esistono diversi controlli per regolare l'immagine, soprattutto per ciò che riguarda l'intensità ed il contrasto



Circuiti ad alta tensione

Poiché i tubi a raggi catodici impiegano voltaggi altissimi, speciali circuiti convertono la tensione di rete ai livelli necessari al loro funzionamento

Circuiti principali

La conversione dei segnali provenienti dal computer e la generazione di quelli necessari per visualizzare le immagini vengono fatte da speciali circuiti contenuti su una o più schede. Alcuni segnali di sincronizzazione vengono impiegati per regolare l'alimentazione del tubo a raggi catodici

I "cannoni"

Come in un televisore a colori, anche in un monitor sono presenti tre "cannoni" elettronici (uno per ciascun colore fondamentale) collocati posteriormente al tubo

Alimentatore

Il tubo a raggi catodici necessita di una tensione molto stabile ad alto voltaggio, rendendo indispensabile l'uso di trasformatori piuttosto ingombranti

DAVID WEEKS

Anodo

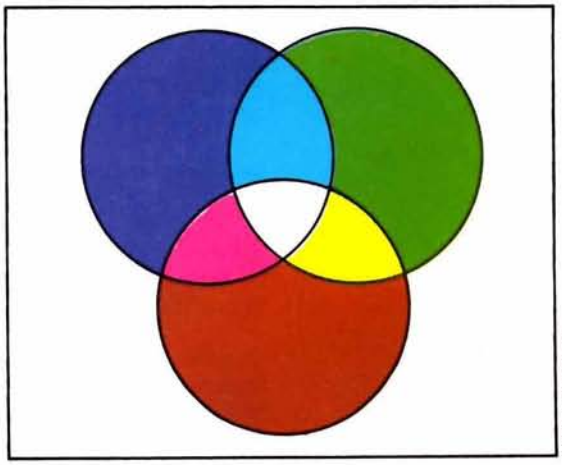
I raggi elettronici prodotti dai "cannoni" subiscono un'accelerazione mediante l'impiego di un campo elettrico ad alto voltaggio, applicato ad una speciale placca metallica collocata lateralmente al tubo

Giogo di deflessione

Il giogo di deflessione, costituito da più bobine di filo elettrico, genera un fortissimo campo magnetico che, variando molto rapidamente, "guida" il raggio elettronico sulla superficie dello schermo

LE VARIAZIONI DI COLORE

Se facciamo passare un raggio solare attraverso un prisma di vetro, esso si scompone in uno "spettro" di colori, che vanno dal rosso al blu. Se il raggio viene fatto passare attraverso un secondo prisma, esso si ricompone in un fascio di luce "bianca". Un simile processo di scomposizione e ricomposizione viene adoperato nei monitor a colori. Combinando opportunamente i raggi prodotti dai tre "cannoni" elettronici è possibile generare qualsiasi tonalità di colore



MARK WATKINSON

La struttura di un programma

I criteri da seguire per ottenere programmi ben congegnanti

Lo sviluppo e la stesura di un programma possono essere fatti in più maniere: c'è chi lavora disordinatamente, mettendosi subito a scrivere linee di programma in BASIC (spesso senza avere nemmeno una precisa idea sullo scopo del programma) e chi, al contrario, procede con ordine e metodo, partendo da un'analisi preliminare del problema. Ovviamente, i risultati migliori si hanno seguendo quest'ultimo metodo.
In questa lezione vedremo come è possibile suddividere un programma abbastanza complesso in tanti "sottoprogrammi" o "moduli" elementari, che possono essere scritti l'uno indipendentemente dall'altro. Oltre all'ovvio vantaggio di poter eseguire le verifiche sui singoli moduli, l'uso di sottoprogrammi (o "subroutine") consente uno sviluppo logico ed ordinato del programma. L'impiego di diagrammi di flusso (vedere a pag. 104), inoltre, può essere di notevole aiuto durante la stesura del programma.
Prima di metterci al computer, è sempre consigliabile stendere una sommaria descrizione del problema che abbiamo intenzione di risolvere.
Nei successivi passi, ogni fase dello sviluppo viene approfondita, con dettagli sempre più precisi, fino a giungere alla stesura del programma in istruzioni BASIC.
Supponiamo di voler scrivere un programma che, dopo aver accettato una serie di nominativi (nomi, seguiti dai cognomi), provveda a riordinarli alfabeticamente (secondo i cognomi) ed, infine, a stampare ciascun cognome, seguito dal relativo nome, secondo questo ordine alfabetico.
Per esempio, avendo digitato alla tastiera prima il nominativo CARLO ROSSI e poi ALBERTO BIANCHI, il programma dovrà stampare:

BIANCHI, ALBERTO
ROSSI, CARLO

In breve, una prima descrizione delle funzioni di "input" e di "output" è:

Passo 1
Immissione di nomi, seguiti dai cognomi, in ordine casuale
Stampa dei cognomi, in ordine alfabetico, seguiti dai nomi

Questo appunto ci serve per fissare, in due parole, cosa deve fare il programma. Adesso occorre approfondire la descrizione delle varie fasi, pur non entrando in troppi dettagli:

Passo 2
Chiedere il numero di nominativi da elaborare
Immissione dei nominativi
Scambiare di posto il nome col cognome
Riordinare alfabeticamente i nominativi
Stampare i nominativi ordinati

Si esamini attentamente questo breve elenco di funzioni, eseguendolo mentalmente, nel caso avessimo omesso qualche operazione importante. Quando siamo sicuri di non aver tralasciato niente, passiamo ad approfondire maggiormente la descrizione. Le singole funzioni riassunte nel Passo 2 sono sufficientemente elementari per essere tradotte in piccoli sottoprogrammi individuali. Possiamo, per maggiore chiarezza, assegnare un nome a ciascun sottoprogramma: la prima subroutine (per il numero dei nominativi) può essere chiamata, ad esempio, CHIEDINUM; la seconda (per l'immissione dei nominativi), IMMETTI; la terza SCAMBIA, la quarta (per il riordino) SORT e la quinta, infine, STAMPANOMI.

Passo 3.1 CHIEDINUM
Chiedi all'operatore di immettere il numero
Accetta il numero N di nominativi
Dimensiona una variabile "stringa" usando N
Passo 3.2. IMMETTI
Se il numero di nomi è minore di N, invita l'operatore ad immettere un (altro) nome
Aggiungi il nome alla variabile "stringa"
Passo 3.3. SCAMBIA
Trova la lunghezza della stringa (nome)
Trova un carattere "spazio" nella stringa
Metti i caratteri che precedono lo spazio in una variabile temporanea
Metti i restanti caratteri in un'altra variabile temporanea, aggiungendo una virgola ed uno spazio
Trasferisci il cognome, seguito dal nome, nella variabile indicizzata
Passo 3.4. SORT
Confronta il primo elemento della "stringa" indicizzata col secondo
Se il primo è più alto (alfabeticamente), scambia i due elementi
Confronta il secondo ed il terzo elemento e scambia se necessario
Continua fino ad esaurire tutti gli elementi
Ripeti i confronti ripartendo dal primo elemento, finché non vengono più eseguiti scambi

Nota Bene: La parte di programma che opera il riordino (SORT) è identica a quella già presentata in una precedente lezione.

Passo 3.5 STAMPANOMI
Stampa tutti i nominativi contenuti nella "stringa" indicizzata

A questo punto, il funzionamento del programma è stato descritto con sufficiente accuratezza in ogni sua parte, salvo quella chiamata SORT (già trattata precedentemente) e quella indicata con SCAMBIA, che vedremo più avanti. Adesso possiamo iniziare la conversione vera e propria in linee di istruzioni in BASIC.

Passo 4
1. CHIEDINUM

Le tre righe di testo del passo 3.1 si convertono direttamente in istruzioni BASIC: un messaggio viene visualizzato usando una PRINT; il numero è immesso grazie a una INPUT, e la dimensione è assegnata con una DIM:

```
PRINT "QUANTI NOMINATIVI DA ELABORARE?"
INPUT N
DIM A$(N)
RETURN
```

La variabile N contiene adesso il numero massimo di nominativi da immettere. Le variabili che devono contenere caratteri alfanumerici (stringhe) vengono individuate col simbolo $: ecco dunque che la variabile stringa indicizzata A$ viene DIMensionata con il valore di N. L'istruzione RETURN serve per far ritornare l'esecuzione al programma principale. I valori assegnati alle variabili nelle subroutine possono essere impiegati, successivamente, in qualsiasi altra parte del programma.

2. IMMETTI

Finché il numero di nominativi immessi è minore di N, l'operatore deve essere invitato ad immetterne altri, che vengono depositati nella variabile stringa indicizzata. Può essere conveniente, a questo scopo, impiegare un ciclo FOR ... NEXT:

```
FOR X = 1 TO N
PRINT "NOMINATIVO"
INPUT A$(X)
NEXT X
RETURN
```

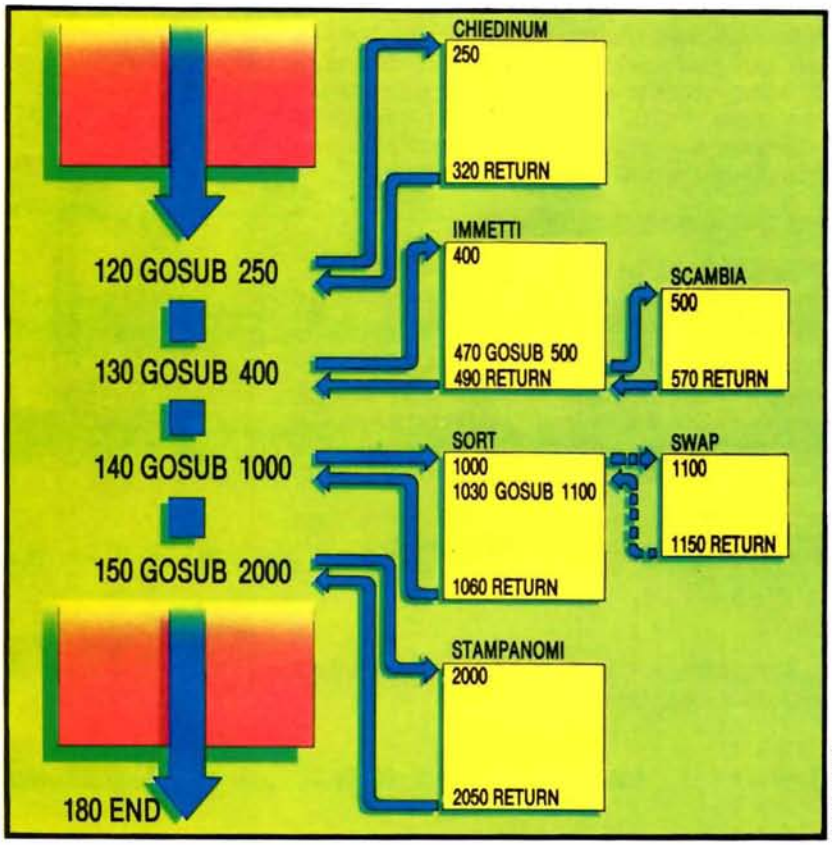
Queste poche istruzioni permettono di immettere tutti gli N nominativi. I lettori più attenti si saranno già accorti di un particolare importante: avevamo previsto, più avanti nel programma, lo scambio del nome col cognome. Ciò impone di rielaborare per una seconda volta ciascun elemento della variabile indicizzata. Perché, allora, non eseguire lo scambio durante la stessa fase di immissione dei nominativi? Infatti, basta aggiungere una linea di programma alle precedenti:

```
FOR X = 1 TO N
PRINT "NOMINATIVO"
INPUT A$(X)
GOSUB [SCAMBIO]
NEXT X
RETURN
```

Così, terminata l'immissione dei nominativi, essi saranno già pronti nel formato cognome/nome per il riordino alfabetico.

3. SCAMBIO

Per scambiare di posto il nome col cognome, dobbiamo poter individuare la posizione del carattere "spazio" che funge da "separatore" tra i due. Il BASIC dispone di speciali funzioni per eseguire questa operazione, come pure per l'estrazione ed il trasferimento di una parte della stringa in altre stringhe. Le funzioni, nel BASIC, sono speciali istruzioni che operano sul valore indicato tra parentesi subito dopo il nome della funzione. Molte delle funzioni sono predefinite nel BASIC, ma è anche possibile, da parte del programmatore, definire nuove funzioni. Una tipica funzione predefinita è SQR ( ). Questa calcola il valore della radice quadrata del numero indicato tra parentesi. Per esempio: LET A = SQR (9) assegna ad A il valore 3 (radice quadrata di 9). Torneremo sull'argomento delle funzioni più avanti nel corso.
La routine SCAMBIO impiega le funzioni LEN (per trovare la lunghezza di una stringa), INSTR (per individuare la posizione del carattere "spazio"), LEFT$ (per estrarre da una stringa un certo numero di caratteri partendo da sinistra) e RIGHT$ (per estrarre da una stringa un certo numero di caratteri partendo da destra). Per adesso non ci soffermiamo sul funzionamento di questi comandi.



4. SORT

Il funzionamento della routine di SORT è già stato trattato nel numero precedente.

5. STAMPANOMI

Questo modulo è veramente semplice:

```
FOR Q = 1 TO N
PRINT A$(Q)
NEXT Q
RETURN
```

A questo punto, la stesura del programma principale si riduce a:

```
REM PROGRAMMA PRINCIPALE
GOSUB [CHIEDINUM]
GOSUB [IMMETTI]
GOSUB [SORT]
GOSUB [STAMPANOMI]
END
```

Abbiamo racchiuso i nomi dei moduli tra parentesi quadre, poiché nella maggior parte dei BASIC occorre sostituire a questi i vari numeri di linea.

Esercizi

Avendo già passato in rassegna quasi tutte le più salienti caratteristiche del linguaggio BASIC, è tempo di verificare, con alcuni esercizi, quanto è stato effettivamente appreso. Gli esercizi sono di varia difficoltà, ma tutti risolvibili con le cognizioni finora presentate.

* **Le variabili.** Alcuni nomi di variabili riportati qui sotto servono per valori numerici, altri per stringhe alfanumeriche, altri ancora non sono validi. Si disegni un cerchietto intorno a quelli per valori numerici ed una crocetta su quelli non validi:

A B6 2Z D$ 15 X$ A12 D9 Q81 Q5 6F H$

* **Aritmetica 1.** Si scriva un piccolo programma per assegnare il valore 6 alla variabile B e per visualizzare il contenuto di B.

* **Aritmetica 2.** Si scriva un breve programma per assegnare alla variabile A il valore 5, alla variabile B il valore 7, alla variabile C il valore 9, ed alla variabile D la somma di A, B e C, visualizzando il risultato.

* **Aritmetica 3.** Si osservino le seguenti linee di BASIC e si determini il valore di C:

```
LET C = 5 + 4 + 3
PRINT C
```

* **Aritmetica 4.** Che risultati verranno stampati da questi due programmi?

```
LET A = 3
LET B = 2
LET C = 9
LET D = 4
```

```
10 REM QUESTO PROGRAMMA RIORDINA UNA LISTA
20 REM DI NOMINATIVI IN ORDINE ALFABETICO
30 PRINT "IMMETTERE IL NUMERO DI NOMI"
40 PRINT "DA RIORDINARE, POI IMMETTERE"
50 PRINT "OGNI NOMINATIVO COL FORMATO:"
60 PRINT "NOME(SPAZIO)COGNOME"
70 PRINT
80 REM
90 REM QUESTO E' IL PROGRAMMA PRINCIPALE
100 PRINT
110 PRINT
120 GOSUB 250
130 GOSUB 400
140 GOSUB 1000
150 GOSUB 2000
160 REM
170 REM FINE DEL PROGRAMMA PRINCIPALE
180 END
250 REM SUBROUTINE PER LA RICHIESTA
260 REM DEL NUMERO DI NOMINATIVI
270 PRINT "QUANTI SONO I NOMI"
280 PRINT "DA RIORDINARE"
290 PRINT
300 INPUT N
310 DIM A$(N)
320 RETURN
400 REM SUBROUTINE PER L'IMMISSIONE DEI NOMI
410 PRINT "IMMETTERE IL NOME NEL FORMATO"
420 PRINT "NOME(SPAZIO)COGNOME(CR)"
430 PRINT "AD ESEMPIO: MARIO VERDI"
440 FOR X = 1 TO N
450 PRINT "NOME "
460 INPUT A$(X)
470 GOSUB 500
480 NEXT X
490 RETURN
500 REM SUBROUTINE PER SCAMBIARE NOME/COGNOME
510 LET L = LEN(A$(X))
520 LET S = INSTR(A$(X)," ")
530 LET C$ = LEFT$(A$(X),S-1)
540 LET F$ = RIGHT$(A$(X),L-S)
550 LET F$ = F$ + ", "
560 LET A$(X) = F$ + C$
570 RETURN
1000 REM SUBROUTINE PER IL RIORDINO (SORT)
1010 LET S = 0
1020 FOR P = 1 TO N - 1
1030 IF A$(P) > A$(P+1) THEN GOSUB 1100
1040 NEXT P
1050 IF S = 1 THEN GOTO 1000
1060 RETURN
2000 REM VISUALIZZAZIONE NOMINATIVI
2010 PRINT
2020 FOR Q = 1 TO N
2030 PRINT A$(Q)
2040 NEXT Q
2050 RETURN
```

```
LET E = (A + B) * (C — D)
PRINT E

LET E = 5
LET E = E * E
PRINT E
```

* **Confronto 1.** Per quale valore della X il messaggio verrà visualizzato?

```
70 LET A = 5
80 LET B = X
90 LET R = B — A
100 IF R = 0 THEN GOTO 120
110 GOTO 10
120 PRINT "CONGRATULAZIONI! HAI VINTO"
999 END
```

* **Confronto 2.** Qual è il minimo valore della X in base al quale l'esecuzione passerà alla linea 300?

```
250 IF X > 6 * 100 THEN GOTO 300
```

* **Confronto 3.** Qual è il minimo valore della Z in base al quale verrà visualizzato il messaggio di congratulazioni?

```
340 IF Z < 10000 THEN GOTO 500
350 IF Z > = 10000 THEN GOTO 520
 :
 :
500 PRINT "PUNTEGGIO TROPPO BASSO! RIPROVA"
510 GOTO 600
520 PRINT "CONGRATULAZIONI! SEI UN ASSO"
530 GOTO 700
```

* **PRINT 1.** Supponiamo che il valore di T sia 50. Si scriva una linea di programma che visualizzi il messaggio IL VALORE DI T È 50 (Suggerimento: racchiudere il messaggio tra doppi apici, usare il punto e virgola ed il nome della variabile).

* **PRINT 2.** Dopo aver esaminato il seguente listato, completare la frase PRINT alla linea 640 in modo che venga visualizzato un messaggio del tipo:

SPIACENTE, MA IL PUNTEGGIO DI 175 È TROPPO BASSO

Attenzione: il punteggio ha un valore diverso ad ogni "giro"!

```
610 REM LA VARIABILE S CONTIENE IL PUNTEGGIO
620 IF S < = 500 THEN GOTO 640
630 GOTO 700
640 PRINT
```

* **PRINT 3.** Che messaggio apparirà eseguendo questo programma?

```
200 LET A$ = "IL CORSO 'IL MIO COMPUTER'?"
210 LET B$ = "DIMMI, TI PIACE ";
220 PRINT B$
230 PRINT A$
```

* **INPUT 1.** L'istruzione INPUT è una delle tante maniere per assegnare un valore ad una variabile. Nel seguente programma, quale valore dovremo digitare, affinché venga visualizzato il numero 12?

```
60 INPUT N
70 LET N = N * 2
80 PRINT N
```

* **INPUT 2.** Cosa verrà stampato in questo caso?

```
100 PRINT "DIGITA IL TUO NOME"
110 INPUT N$
120 PRINT "CIAO, ";N$;". IO SONO IL TUO COMPUTER"
```

Basic & Basic

Il programma presentato non può funzionare sull'ATARI 400 o 800, a causa del diverso trattamento delle stringhe.

DIM

L'istruzione esiste sullo Spectrum, ma il suo impiego è leggermente diverso, sostituire la linea 310 con:

```
310 DIM A$(N, 30)
```

Similmente, sul Lynx, si sostituisca la stessa linea con:

```
310 DIM A$ (30) (N)
```

GOTO

Nella linea 1050, l'istruzione GOTO 1000 segue immediatamente la THEN. In questo caso, molte versioni di BASIC consentono di omettere la GOTO e scrivere:

```
1050 IF S = 1 THEN 1000
```

INSTR

Questa funzione non è disponibile sullo Spectrum, sul VIC 20, sul Commodore 64, sul Lynx e sull'Oric-1. Sul Commodore, sul Lynx e sull'Oric-1 sostituire la linea 520 con:

```
515 FOR P = 1 TO L
520 CH$ = MID$ (A$(X), P, 1)
522 LET S = 0
523 IF CH$ = " " THEN LET S = P:LET P = L
525 NEXT P
```

INSTR

Nessuna di queste funzioni è disponibile sullo Spectrum, ma è possibile aggirare l'ostacolo creando delle funzioni "ad hoc" con l'impiego della frase DEF FN. Si cancelli la linea 320, aggiungendo:

```
320 DEF FN M$(X$, P, N) = X$(P TO P + N — 1)
330 DEF FN L$(X$, N) = X$(TO N)
340 DEF FN R$(X$, N) = X$(LEN (X$) — N + 1
    TO)
350 RETURN
```

LEFT$

Quindi si eliminino le linee da 510 a 560, sostituendo con:

```
510 LET D$ = A$(X)
520 LET L = LEN(D$)
530 LET S = 0
540 FOR P = 1 TO L
550 IF FN M$(D$, P) = " " THEN LET S = P
560 IF S < > 0 THEN LET P = L
570 NEXT P
580 LET C$ = FN L$(D$, S — 1)
590 LET F$ = FN R$(D$, L — S)
600 LET G$ = F$ + ", " + C$
610 LET A$(X) = G$
620 RETURN
```

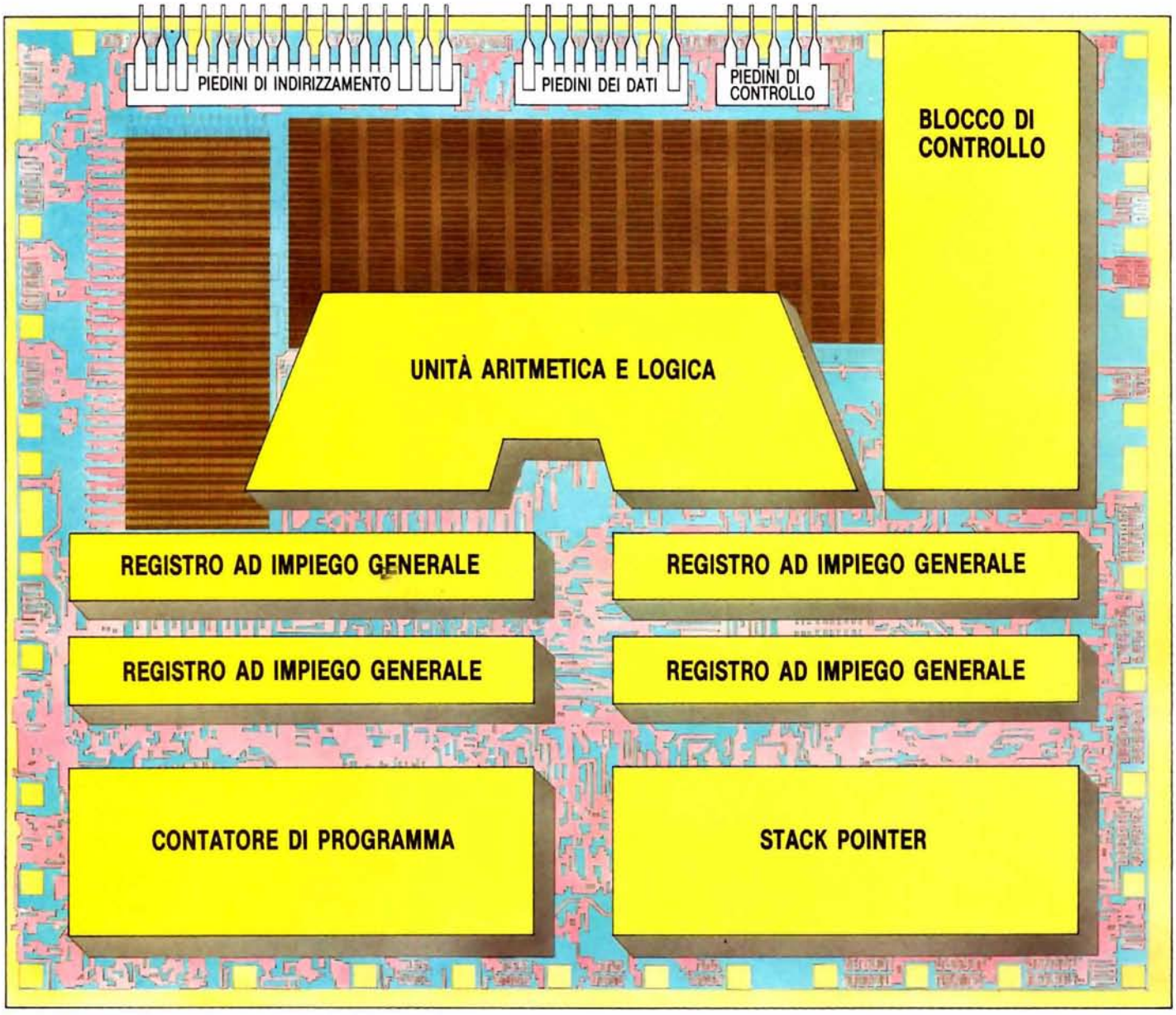
RIGHT$

END

Non disponibile sullo Spectrum, l'Oric-1 e Dragon 32. Sostituire con STOP

Il cervello di un computer

Il funzionamento dell'unità centrale di processo



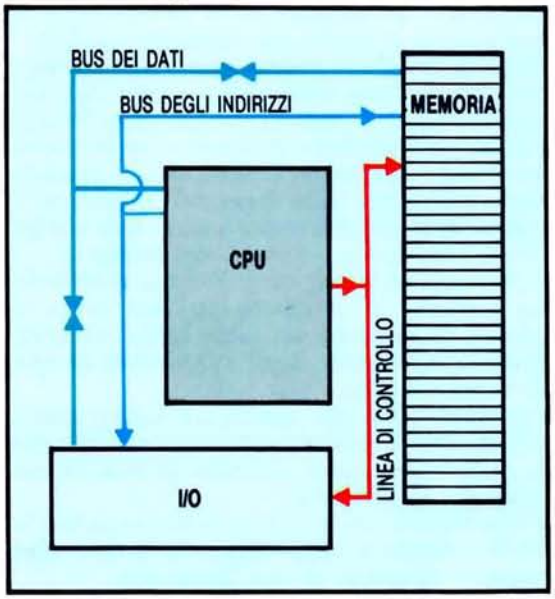
Il funzionamento del computer è controllato dall'Unità Centrale di Processo o CPU (sigla che sta per Central Processing Unit). Un computer, nella sua forma essenziale, è costituito da una CPU, da qualche chip di memoria e da un semplice circuito di I/O (input/output).

La sezione di I/O serve al computer per comunicare con il mondo esterno. In applicazioni molto semplici, per esempio, una lavatrice controllata da microprocessore, il computer deve trasmettere opportuni segnali di I/O, per regolare l'accensione del motore oppure la temporizzazione dell'impianto di riscaldamento. I chip di memoria sono necessari per conservare sia le istruzioni per la CPU sia i dati che quest'ultima deve elaborare. I dati possono includere numeri e codici binari, che rappresentano caratteri visualizzabili (lettere, numeri o simboli come & o @. Se nella memoria sono contenuti sia dati da elaborare sia istruzioni, come può la CPU distinguere fra questi due tipi d'informazione? Per rispondere a questo quesito occorre esaminare il funzionamento interno della CPU. La CPU di un microcomputer a 8-bit (la categoria alla quale appartengono quasi tutti i personal e gli home computer) è contenuta,

generalmente, in un singolo chip a 40 piedini (20 per ciascun lato).
Attraverso ogni piedino (con l'eccezione dei due usati per l'alimentazione) passano dei segnali (sia in entrata che in uscita) destinati alla CPU o alle periferiche (memorie o circuiti di I/O). Una tipica CPU ad 8 bit è dotata di 16 piedini, collegati al "bus degli indirizzi".
Ciascun piedino emette un segnale in uscita, che rappresenta un 1 o uno 0 nella numerazione binaria. Le possibili combinazioni di 1 o 0, utilizzando tutti i piedini, sono 65.536 e vengono adoperate per selezionare le varie locazioni di memoria disponibile. Altri otto piedini vengono impiegati per il trasferimento dei dati tra la CPU e la memoria e vengono collegati, appunto, al "bus dei dati".
Un certo numero di piedini, infine, serve per lo scambio di segnali di "controllo", alcuni dei quali provenienti dalla CPU, altri invece diretti ad essa. Vedremo più avanti a cosa servono questi segnali di controllo.
All'interno della CPU è presente una certa quantità di piccole celle di memoria (di 1 o 2 byte), chiamate "registri". Alcuni di questi "registri" sono riservati a funzioni speciali, altri, invece, sono usati per conservare temporaneamente i dati da elaborare: questi si chiamano "registri d'impiego generico". Ci sono altri due "blocchi" funzionali molto importanti nella CPU: l'ALU e il "blocco di controllo".
La sigla ALU sta per Arithmetic and Logic Unit (unità aritmetica e logica): questa parte della CPU esegue, difatti, le operazioni matematiche e logiche, oltre ad altre non meno importanti quali la manipolazione di bit all'interno dei byte.
Il "blocco di controllo" è uno speciale circuito, progettato appositamente per far sì che la CPU si comporti esattamente secondo le istruzioni ricevute dalla memoria. Possiamo fare un esempio specifico, usando i codici delle istruzioni della CPU dello Z80. Quando la CPU legge dalla memoria il codice 11000110, essa somma il contenuto del prossimo byte di memoria a quello conservato in uno dei "registri" interni alla stessa CPU. Volendo conservare il risultato dell'addizione in una specifica locazione di memoria, la prossima istruzione da impartire alla CPU dovrà essere 00110010, immediatamente seguita da due byte, per specificare l'indirizzo della locazione di memoria dove intendiamo conservare il risultato.
Supponiamo che il risultato dell'addizione sia 37 (in numerazione decimale) e che l'indirizzo della locazione di memoria dove intendiamo conservare il risultato sia 33.126 (decimale): il codice delle istruzioni viene interpretato dal "blocco di controllo", che provvede ad emettere sul "bus degli indirizzi" il valore binario equivalente a 33.126 (1000000101100110) e, successivamente, ad inviare opportuni segnali sui piedini di controllo, per comunicare alla memoria che c'è un dato in arrivo da memorizzare. Il "blocco di controllo", infine, provvede a depositare il valore binario equivalente a 37 (00100101) sul "bus dei dati", da dove viene prelevato dai circuiti della memoria e copiato nella locazione specificata dal "bus degli indirizzi". Se, in un momento successivo è necessario richiamare il dato (per visualizzarlo sullo schermo, per esempio), possiamo impartire un altro tipo di istruzione alla CPU per compiere l'operazione inversa: il "blocco di controllo", questa volta, provvederebbe ad indirizzare la locazione di memoria usata precedentemente, a prelevare il dato ed a copiarlo in uno dei "registri" interni alla CPU.
Il numero di registri (o celle di memoria temporanea) disponibile nella CPU varia da modello a modello. Si possono avere registri lunghi 8 bit (1 byte) o 16 bit (2 byte). Ai registri riservati a funzioni speciali vengono spesso assegnati dei nomi: "stack pointer" (puntatore dello "stack"), "program counter" (contatore di programma) e "accumulatore". I registri di uso generico vengono generalmente indicati con lettere dell'alfabeto: registro C, registro X e così via.
Uno dei più importanti registri a 16 bit è il "program counter". Questa cella di memoria contiene sempre l'indirizzo (in binario) della locazione di memoria dov'è conservata la prossima istruzione da eseguire. Quando la CPU è pronta per l'esecuzione, il contenuto del "program counter" viene depositato sul bus degli indirizzi, ed il byte in quella locazione viene ricopiato nella CPU (per mezzo del bus dei dati).
Il più importante dei registri a 8 bit è il cosiddetto "accumulatore": in esso vengono di solito temporaneamente conservati i risultati delle operazioni eseguite dall'ALU, oppure i dati che vogliamo trasferire da un luogo all'altro della memoria o visualizzare sullo schermo.
Più avanti torneremo a trattare più dettagliatamente il funzionamento della CPU: lo scopo di questa prima trattazione è stato quello di illustrare, a grandi linee, quali e quanti processi avvengano (in poche frazioni di secondo) in quel piccolo involucro nero chiamato microprocessore.

L'architettura della CPU

Il diagramma illustra le parti che compongono l'unità centrale: i "registri" di memoria, l'unità Aritmetico/Logica (che si compone di migliaia di porte logiche), ed il "blocco di controllo". Il blocco di controllo ha il compito di interpretare le istruzioni in codice binario affinché la CPU le esegua correttamente. Ad esso è affidata buona parte della gestione del "bus degli indirizzi", del "bus dei dati" e dei segnali di controllo, che servono a coordinare le molte funzioni



Un computer "minimo"

I computer più elementari sono costruiti impiegando pochi componenti: una CPU, un po' di memoria ed un circuito di I/O (input/output). Nella memoria vengono conservate le istruzioni del programma (che vengono eseguite una alla volta) ed i dati da elaborare. Il circuito di I/O serve a comunicare con il "mondo esterno". Se il computer serve, per esempio, a controllare il funzionamento di una lavatrice, attraverso il circuito di I/O verranno inviati i segnali necessari all'azionamento del cestello e dei vari meccanismi di riscaldamento e pompaggio dell'acqua.
Ogni tipo di CPU adopera un proprio gruppo di istruzioni in formato binario

John von Neumann

L'impostazione tecnica degli attuali computer deve molto a questo brillante matematico



ASSOCIATED PRESS

Una coppia affiatata
John von Neumann con la sua seconda moglie Klara, anch'essa una esperta programmatrice dei primi computer

Solo un ungherese sarebbe capace di entrare dentro una porta girevole dietro a qualcuno e uscirne prima di lui. Così si è espresso John von Neumann per illustrare quanto sono competitivi i suoi compatrioti.
Egli non costituisce un'eccezione alla regola. La sua ambizione, assieme alla sua straordinaria intelligenza, lo hanno portato alle vette più alte del mondo scientifico degli Stati Uniti.
Neumann è nato da una famiglia ebrea benestante dell'Impero austro-ungarico. Le sue straordinarie doti matematiche furono palesi sin dalla più giovane età; infatti a soli 25 anni egli aveva già due lauree e un "dottorato" ed era in grado di discutere su questioni matematiche con personaggi, quali il fisico Einstein e il matematico David Hilbert.
Neumann non ha mai trascurato i grandi problemi mondiali. Con la caduta dell'Impero austro-ungarico, dopo la prima guerra mondiale, egli adottò il titolo nobiliare di "von" e si inserì nella vita accademica della Germania sconfitta. Allo stesso tempo, si costruì molti contatti con gli Stati Uniti, trascorrendo i periodi invernali presso l'Università di Princeton nel New Jersey e quelli estivi in Europa, amministrando le proprietà di suo padre. Allo scoppio della seconda guerra mondiale era già al sicuro negli Stati Uniti.
La sua fama di eminente matematico proviene, principalmente, dalla sua teoria sugli "insiemi", che venne enormemente sottovalutata da Bertrand Russell. Von Neumann era affascinato dalla fisica quantistica e dalla teoria dei giochi. Fu l'inventore del metodo "Monte Carlo", nel quale si adoperano numeri generati a caso per risolvere equazioni matematiche.
Quando anche l'America entrò in guerra, fu immediatamente richiesto il suo intervento nel Manhattan Project, dove con entusiasmo partecipò alla produzione della bomba atomica. Mentre era di stanza a Los Alamos, faceva spesso più di 200 km per poter mangiare nel suo ristorante messicano preferito. Si dice anche che, durante gli ultimi anni che egli trascorse a Princeton, egli abbia distrutto un'auto all'anno a causa della sua guida spericolata.
Era ancora impegnato nel Manhattan Project, quando seppe dei tentativi compiuti da altri scienziati per la costruzione di un computer elettronico e fu invitato a partecipare al progetto ENIAC. Il lavoro era sotto il controllo di ingegneri elettronici ma, in qualità di primo matematico interpellato, egli affrontò i problemi da un nuovo punto di vista e preparò un rapporto, che divenne poi la base per il progetto dei computer moderni.
Dopo la fine della guerra si impegnò sempre più nella politica della difesa degli USA ma non per questo tralasciò le sue ricerche nel campo matematico e progettò il primo computer dell'Università di Princeton, chiamandolo JOHNIAC. Alla festa per celebrare il completamento dell'apparecchio, fece scolpire nel ghiaccio un modello del computer.
Purtroppo, all'età di circa 50 anni, colpito da un tumore, fu confinato su una sedia a rotelle, ed egli, che per tutta la sua vita era stato un agnostico, negli ultimi mesi di vita divenne cattolico.
Parlando della sua morte, un amico disse: «Penso che von Neumann abbia sofferto più di ogni altro uomo, quando la sua mente cessò di funzionare».
Il suo ricordo è suggellato, nella progettazione del computer, al sistema noto ancora oggi come *Architettura di von Neumann.*

1903
Janos Louis Neumann nasce a Budapest il 28 dicembre

1921
Pubblica, insieme al suo maestro, la sua prima relazione sulla matematica

1933
Diviene professore alla Princeton University, New Jersey, dove furono accolti Einstein e altri matematici fuggiti dall'Europa

1942
Pubblica un libro sulle applicazioni della teoria dei giochi nell'economia

1943
Viene nominato consulente del Manhattan Project a Los Alamos, New Mexico

1944
Viene informato del progetto segreto ENIAC

1945
Partecipa alle prime prove sulla bomba atomica

1947
Prepara il rapporto sull'ENIAC ed i progetti preliminari del nuovo computer chiamato EDVAC (Electric Discrete Variable Computer)

1951
JOHNIAC, il suo computer personale, entra in funzione all'Università di Princeton

1951-53
Presidente della American Mathematical Society, si dedica allo studio di nuove teorie

1955
Nominato dal Presidente Eisenhower Alto Commissario per l'Energia Atomica Americana

1957
Muore di cancro l'8 febbraio. Per l'occasione un intero volume del Bollettino della Società Matematica Americana viene dedicato alla sua vita ed al suo lavoro

Musica e computer

Si può creare della musica elettronica anche col computer

Il computer non serve esclusivamente per elaborare informazioni o per giocare: può essere usato anche come strumento musicale. Il processo di creare della musica con l'impiego di un elaboratore si chiama "sintesi musicale".
Il computer serve anche per rendere più vivo e interessante l'insegnamento della musica e, a lungo andare, può rivelarsi più economico (e forse più... paziente) di un maestro di musica. Più avanti spiegheremo come creare musica usando un piccolo home computer: per il momento ci limiteremo ad illustrare come ciò viene fatto nel campo professionale, poiché la maggior parte delle idee applicate ai sintetizzatori professionali finiscono prima o poi col venire estese ad apparecchi più economici, ivi inclusi i piccoli computer.
Gli strumenti musicali "automatici" hanno sempre avuto un grandissimo successo ed hanno molti aspetti in comune con il computer. La pianola, una specie di pianoforte automatico presente in molti salotti dell'epoca vittoriana, funzionava per mezzo di un rotolo di carta perforata, mentre il carillon, ancora oggi molto diffuso, funziona mediante un rullo o disco metallico ruotante, munito di piccoli piolini, che servono a far vibrare delle linguette metalliche e così si riproduce un motivo musicale.
Perfino i vecchi organetti a manovella, tanto odiati da Charles Babbage (uno dei pionieri dell'informatica), che avrebbe voluto proibirne l'uso nelle pubbliche vie, erano, per un certo verso, programmabili. Allora, i suonatori di organetti da strada inscenarono una protesta... musicale sotto le finestre di Babbage: oggi, sono le associazioni dei musicisti, che si battono per abolire l'uso dei congegni musicali programmabili: in Gran Bretagna, nel 1982, alla sezione di Londra dell'Unione Musicisti ha votato per proibire l'uso di tali congegni in sala di registrazione e in concerti dal vivo, perché, viene affermato, i sintetizzatori sono in grado di imitare tanti strumenti, anche più di uno per volta, con il rischio di togliere il lavoro a tanti musicisti.
I sintetizzatori musicali elettronici si trovano sul mercato già da diversi anni, ma l'introduzione di tecniche digitali ha aperto nuovi orizzonti. Oggi non occorre più manovrare un'infinità di tasti per riprodurre un determinato suono: è possibile, infatti, registrarlo (sia esso prodotto da uno strumento musicale, o, al limite, dallo scoppio di un palloncino), farlo analizzare dal computer, che ne scompone le frequenze, e ricomporlo, successivamente, in qualsiasi tonalità.
I suoni "digitali" hanno qualcosa in comune con le fotografie riprodotte sui giornali. Osservando attentamente una di queste, si nota che è composta da tanti piccoli puntini singoli, mentre la fotografia originale era probabilmente ricca di sfumature senza delimitazioni così nette. Nella stessa maniera il suono originale può essere scomposto in una sequenza digitale e questa tecnica si chiama "sampling" o "campionatura".
Sistemi di questo tipo sono molto costosi (gli apparecchi più noti e sofisticati sono i Fairlight ed i Synclavier), ma siccome essi sono in grado di riprodurre il suono di vari strumenti musicali, il loro prezzo risulta spesso inferiore all'ingaggio di un gruppo di musicisti.
Le apparecchiature digitali si stanno sempre più diffondendo, ma ci vorrà ancora molto tempo, prima che i sintetizzatori analogici escano definitivamente di scena. In questi ultimi viene adoperata una tecnica chiamata "sintesi per sottrazione", paragonabile al modo col quale uno scultore crea una statua da un pezzo di marmo. Si inizia con un suono principale, creato elettronicamente, che viene poi elaborato tramite diversi processi elettronici. Ciascun processo modifica il suono originale, togliendo o aggiungendo determinate frequenze, fino ad ottenere il suono desiderato. Questo metodo incoraggia il musicista a sperimentare le più diverse combinazioni di processi ed è relativamente facile per un principiante.



per concessione della EMI

Suonatori solitari

I suonatori di sintetizzatori elettronici, come Klaus Schultz (nella foto) stanno sfruttando sempre più i fantastici poteri dei loro strumenti elettronici basati su microprocessori per riprodurre, in concerto, suoni che fino a 20 anni fa avrebbero richiesto l'impiego di un'intera orchestra

Effetti speciali

I sintetizzatori musicali controllati da piccoli computer stanno diventando sempre più richiesti. Oggi sono in vendita a prezzi decisamente accessibili anche al dilettante e permettono di ottenere effetti musicali paragonabili a quelli che, solo 10 anni fa, erano possibili soltanto con delle attrezzature molto più costose. Quello illustrato nella foto è capace di elaborare nella propria memoria brani di musica letta da spartiti stampati sotto forma di codice "a barre". Molti di questi sintetizzatori "domestici" si collegano direttamente ad un home computer, avvantaggiandosi sia dello schermo che della memoria in più. Comunque, in un numero sempre crescente di home computer viene incorporata qualche forma di sintetizzatore musicale



IAN McKINNELL

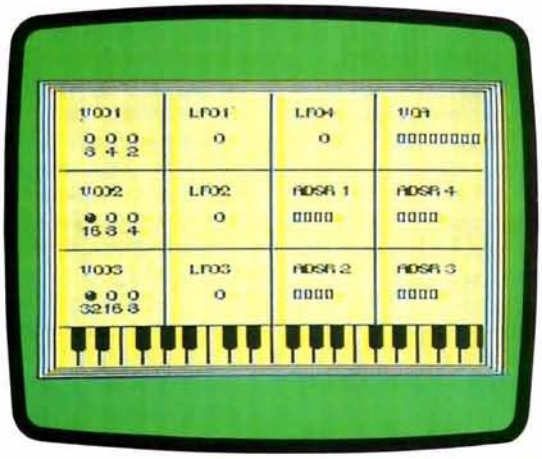
Al contrario, qualsiasi operazione su di un sintetizzatore digitale deve essere progettata con la medesima cura ed attenzione che si richiedono nella costruzione di un grande edificio, poiché viene impiegato un metodo di "sintetizzazione per addizione", ossia: il suono finale è prodotto aggiungendo più componenti, una sovrapposta all'altra. Il suono diviene appena riconoscibile solo verso la fine del processo. Tuttavia, si può anche partire scomponendo un suono "tradizionale", conservandone le componenti nella memoria del computer (oppure su dischi o nastri) per poi usarle come "mattoni" per la costruzione del suono desiderato.

A parte la capacità di creare una gamma così vasta di suoni singoli, il computer è anche in grado di conservare delle sequenze di suoni o intere composizioni musicali. I sintetizzatori più venduti sono dotati di un "sequenziatore", per conservare e richiamare una particolare sequenza di suoni. Il Fairlight, per esempio, può conservare, su dischetto, composizioni della durata di ben 30 minuti, anche composte di più "voci".

Accade sempre più spesso che, con l'impiego dei sintetizzatori più sofisticati (il cui prezzo supera i 40 milioni di lire) il compositore, i musicisti e il direttore d'orchestra sono tutti riuniti in una sola persona.

Scatole musicali

Esiste del software specializzato, che dà modo all'utente di sfruttare al massimo le capacità musicali del proprio computer. Lo schermo, nella maggior parte dei casi, viene usato per presentare un'interpretazione visiva della musica durante l'esecuzione, oppure è possibile usare la tastiera del computer come se fosse quella di un pianoforte



IAN McKINNELL

Se ascoltiamo un disco a 33 giri, ma suonato alla velocità di 45 giri, possiamo notare che i suoni hanno un timbro molto più acuto rispetto al normale. Uno dei fattori più affascinanti del sintetizzatore computerizzato è la capacità di ovviare a questo inconveniente, essendo possibile ascoltare un pezzo di musica ad una velocità maggiore o inferiore senza alterarne la tonalità o, viceversa, alterare la tonalità del pezzo mantenendo invariata la velocità.

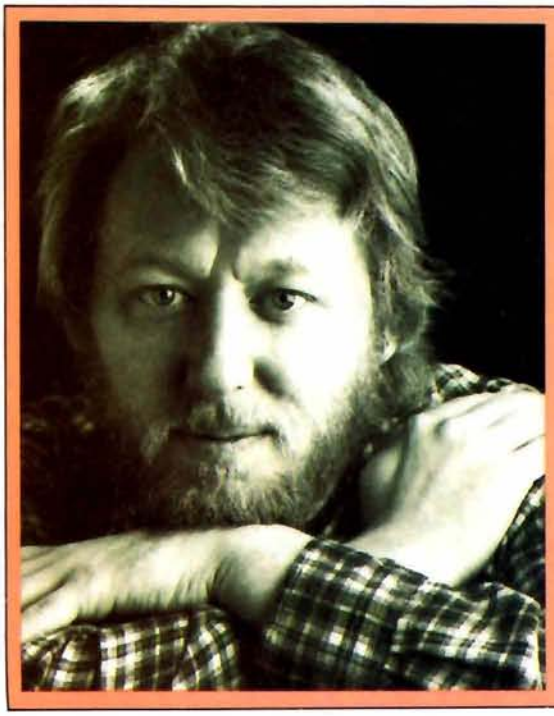
È perfino possibile prendere un brano musicale suonato, per esempio, da una tromba, duplicarlo al sintetizzatore, e miscelarlo in modo tale da ottenere lo stesso brano, ma suonato da un fagotto. I due suoni possono poi essere eseguiti contemporaneamente, o alternativamente in armonia. Questo processo si chiama "track bouncing".

Molti di questi apparecchi vengono programmati con uno speciale linguaggio per composizione musicale (nel caso del Synclavier il linguaggio è lo SCRIPT), simile al BASIC, ma un po' più difficile a comprendere. Una delle caratteristiche più sbalorditive si chiama "reverse compiling" (compilazione inversa), e consiste nel suonare normalmente un pezzo di musica sulla tastiera, mentre il computer provvede a tradurre il pezzo in linguaggio SCRIPT.

A questo punto, se l'esecuzione non è riuscita perfettamente, è possibile modificare le istruzioni del programma generato, usando il terminale del computer come se si trattasse di un programma in BASIC, e riascoltare il brano automaticamente. Un altro sistema, simpatico ma meno duttile, è quello adottato in uno dei sintetizzatori Yamaha: in base al brano eseguito manualmente, viene stampato automaticamente lo spartito tradizionale, cioè sul normale pentagramma.

I sintetizzatori trovano largo impiego anche nel mondo del cinema. Le eccezionali immagini del film di Walt Disney *Tron* sono state prodotte al computer. Ma forse non tutti sanno che anche parte della musica ed i vari effetti sonori sono stati creati usando un

microcomputer. Nel film c'è anche della musica "vera" (suonata dalla London Philharmonic Orchestra), ma tanti rumori, come quello prodotto dal dirigibile Goodyear e quello di un frigorifero sono stati registrati e modificati per mezzo di un Fairlight.
Gli effetti sonori erano così tanti, che si è reso necessario catalogarli usando un "database", il "File Manager 800 +" su di un computer Atari 800. Anche alcune voci dei personaggi sono state prodotte impiegando un piccolo computer ed un sintetizzatore, riuscendo a miscelare voce e musica. La Atari, per l'occasione, ha fornito un package di software che, fino ad allora, era stato usato soltanto all'interno della ditta per creare i sofisticati effetti sonori dei giochi elettronici per sale giochi.
L'Atari asserisce che questo "Electronic Sound Assembly System" (Sistema elettronico di composizione dei suoni) costituisce, per la creazione di effetti sonori, un enorme passo avanti, paragonabile a quello compiuto nell'elaborazione dei testi grazie ai word processor.
Esiste una gran quantità di accessori che si possono applicare ad un home computer per trasformarlo in un sofisticato sintetizzatore musicale. Alcuni di questi apparecchi aggiuntivi permettono di affidare al computer tutto il lavoro creativo e producono una sintetizzazione di alta qualità, mentre altri includono anche una tastiera tipo pianoforte.
Tra i sintetizzatori computerizzati per uso dilettantistico, quello forse più completo è il Casio CT7000. La sua attrattiva principale è un dispositivo, chiamato "sequenziatore polifonico", che, per mezzo di un registratore a cassette e una memoria RAM molto estesa, funziona quasi come un registratore professionale a più piste. Un normale "mangiacassette" ha una sola pista per lato, tutt'al più due, se è stereo: un registratore professionale ne può avere anche più di 24, permettendo di registrare anche un solo strumento per pista e combinare, con un successivo lavoro di "mixaggio", tutti i suoni insieme per riprodurre l'intera orchestra. Con il CT7000 si possono comporre delle vere e proprie sinfonie.
Il predecessore del CT7000 (il CT701) adoperava un lettore di codice "a barre" (vedere a pag. 49) per leggere la musica da speciali spartiti e trasferirla nella memoria della macchina.
Purtroppo non esistono, per il dilettante, metodi sufficientemente pratici per la compilazione di spartiti così concepiti, e quindi l'uso è esclusivamente limitato alla musica stampata dalla Casio.
Talvolta per aver successo nella musica leggera non è necessario avere un sintetizzatore molto sofisticato. Il complesso tedesco dei "Trio" nella loro canzone *Dadada* hanno usato un Casiotone VL1 del valore di circa 100.000 lire, riscuotendo più successo di Peter Gabriel che usa un Fairlight del valore di diversi milioni di lire. Il VL1 è, essenzialmente, un apparecchio monofonico (si può suonare una sola nota per volta), ma è capace di imitare diversi strumenti musicali e di memorizzare una sequenza di note. È, inoltre, possibile cambiare alcune caratteristiche delle note per creare un suono tutto particolare. Più precisamente, è possibile variare l'inviluppo della nota o "ADSR" (sigla che sta per Attack, Decay, Sustain e Release). Anche in certi home computer, come il Commodore 64 e l'Oric-1 è possibile un simile processo ed ovviamente il fatto stesso di essere programmabili in BASIC significa che possono essere impiegati anche come "sequenziatori".
Sono in vendita diversi programmi per computer, anche per quelli dalle limitate capacità musicali, scritti per comporre musica. In alcuni casi la composizione viene fatta usando un pentagramma riprodotto sullo schermo e la selezione delle note avviene mediante l'impiego di un joystick o di una penna ottica. Alla pressione di uno speciale tasto, viene suonata l'intera composizione. In altri casi, invece, sul video viene riprodotta una tastiera da pianoforte: anche qui le note vengono selezionate con una penna ottica o con un joystick oppure per mezzo della tastiera del computer.
È possibile programmare effetti musicali in BASIC anche senza l'ausilio di software speciale come quello sopra descritto, ma come succede anche per la grafica, i metodi variano da computer a computer, come pure il grado di sofisticazione dei comandi BASIC dedicati a questo compito. Il Dragon 32, per esempio, è monofonico, ma dispone di un comando PLAY, che permette di suonare una sequenza di note scritte usando le lettere dell'alfabeto (secondo la notazione musicale anglosassone). Al contrario, il Commodore 64 ha dei sofisticati circuiti incorporati per la generazione di effetti sonori, ma non possiede comandi in BASIC altrettanto facili da usare. Vedremo più avanti come è possibile usare il computer per fare musica.



LEN LEWIS

MARTIN RUSHENT

Martin Rushent è il produttore di diversi famosi complessi (gli "Altered Images", i "Dexy's Midnight Runners" ed altri) ed è anche uno dei più famosi promotori della musica prodotta al sintetizzatore. Nelle sue sale di registrazione, nel cuore della campagna inglese, ci sono nove diversi sistemi che rappresentano quanto di meglio si possa oggi trovare. Ciascun sistema si avvale di un metodo diverso di programmazione e di codici diversi per le note della scala musicale. Nonostante vari tentativi di riunirli tutti insieme in un unico "megasistema", Rushent ha dovuto ideare un programma, scritto da lui stesso, su un piccolo home computer, per convertire un pezzo di musica da un sistema ad un altro

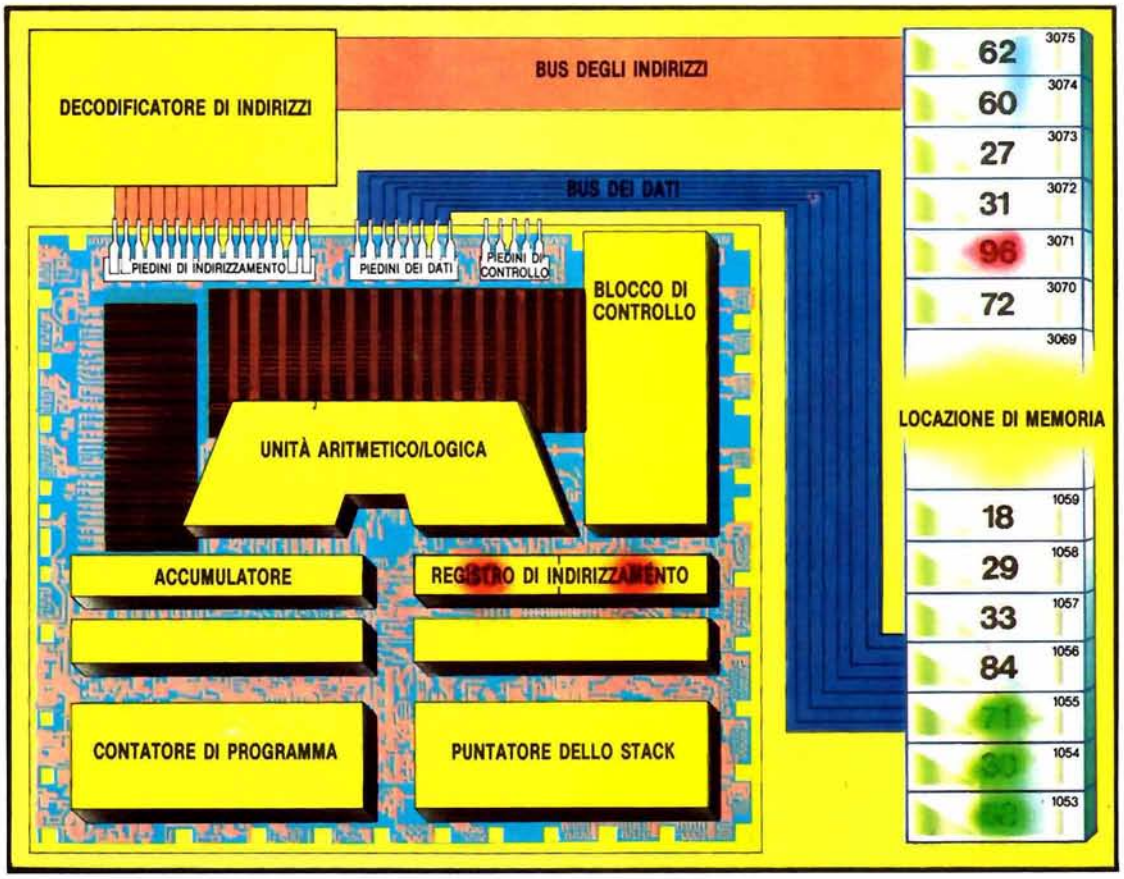
La CPU e gli indirizzi

Come fa la CPU a distinguere fra istruzioni e dati?

Per leggere le istruzioni e i dati, conservati nella memoria, la CPU seleziona le varie locazioni della stessa memoria mediante opportuni segnali sul "bus degli indirizzi", trasferendo il dato per mezzo del "bus dei dati". Ciò è vero in linea teorica, perché, in pratica, il processo è un poco più complicato.
In particolare, il byte conservato in una delle migliaia di locazioni di memoria può rappresentare sia un'istruzione (per comunicare al computer la prossima azione da intraprendere), sia un dato da elaborare. Vediamo come la CPU è in grado di distinguere fra i due casi.

Il riconoscimento dei codici

Per prima cosa, esaminiamo cos'è un'"istruzione". Essa è un particolare codice binario, che indica alla CPU una specifica sequenza di operazioni da eseguire. Per esempio, il



UNA CATENA DI EVENTI

Anche le più semplici operazioni della CPU implicano più fasi successive. La CPU legge, una alla volta, le istruzioni depositate nella memoria. Esse vengono riconosciute ed interpretate dal blocco di controllo e comportano l'esecuzione di tutta una serie di operazioni. Nella figura, dalla locazione 1053 viene letta l'istruzione 58. Questa istruzione provoca questa catena di eventi: il byte alla locazione successiva (1054) viene letto e depositato nella prima metà di uno speciale registro a 16 bit; nella seconda metà viene copiato il byte contenuto nella locazione 1055. Questi due byte rappresentano l'indirizzo al quale andare a leggere un particolare dato, perciò il valore (3071) viene depositato sul "bus degli indirizzi" e il dato ivi contenuto (96) viene infine letto e copiato dalla CPU nel registro interno, chiamato "accumulatore". A questo punto sul bus degli indirizzi viene ripristinato il valore iniziale, aumentato di 1, ossia 1056, dove la CPU sa di trovare la successiva istruzione, che verrà a sua volta interpretata ed eseguita attraverso una simile serie di operazioni. Essendo l'istruzione composta di un solo byte, quella successiva si troverà alla locazione 1057

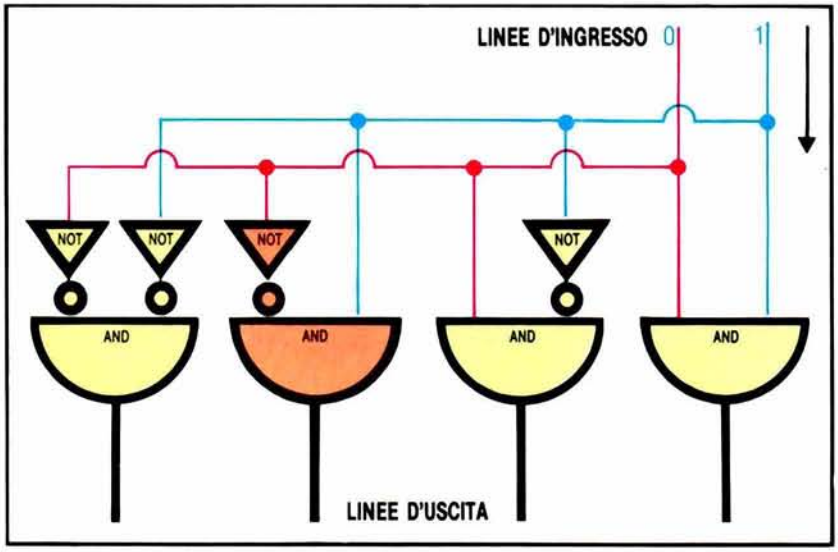
codice binario 00111010, se riconosciuto dalla CPU come istruzione e non come dato da elaborare, può significare: interpreta i prossimi due byte come indirizzo di una locazione della memoria, seleziona la locazione di memoria indicata da questi due byte, e copia il dato contenuto in essa nello speciale registro "accumulatore" della CPU.
A parole, tutto ciò appare molto complicato, ma non è che la descrizione di una delle tante istruzioni di indirizzamento tipiche della CPU Z80. Il processo appena descritto può essere più chiaro osservando l'illustrazione. Supponiamo che la CPU sappia che il prossimo byte, nella locazione di memoria 1053, sia un'istruzione e non un dato. L'indirizzo 1053 in forma binaria (0000010000011101) viene messo sul "bus degli indirizzi" ed interpretato dal "decodificatore di indirizzi", il quale provvede a selezionare un solo ed unico canale di uscita: quello corrispondente alla locazione 1053.
Il passo successivo consiste nel depositare il contenuto di detta locazione (nel nostro esempio il valore 58 decimale o 00111010 binario) sul "bus dei dati" in modo che la CPU possa leggerlo e copiarlo in uno speciale registro interno. La CPU, sapendo che il byte rappresenta un'istruzione, ne chiede l'interpretazione al blocco di controllo. A questo punto la CPU, sapendo che i prossimi due byte di memoria devono essere interpretati come dati (l'indirizzo di una locazione di memoria dalla quale leggere un byte), attenderà di leggere la prossima istruzione (contenuta nella locazione 1056) fino a quando tutte le operazioni non sono completate.
L'istruzione 58 fa sì che l'indirizzo attuale (1053) venga incrementato di 1, per "puntare" al primo dei due byte di dati, permettendo alla CPU di copiarne il valore in uno speciale registro della CPU. Questo registro è composto di 16 bit (due byte), pertanto il valore appena depositato non è che metà dell'intero indirizzo; quindi, la CPU procede con la copia del valore contenuto nella successiva locazione di memoria (la 1055).

Il trasferimento dei dati

La fase successiva (si ricordi che tutto questo processo fa parte di un'unica istruzione) consiste nel depositare l'indirizzo 3071, appena letto, sul "bus degli indirizzi". Di nuovo, il "decodificatore di indirizzi" provvede alla corretta selezione della memoria ed il contenuto della locazione indirizzata (nel nostro caso il valore 96), viene letto dalla CPU e depositato nel registro "accumulatore". Terminata questa operazione, la CPU è pronta per leggere la prossima istruzione, contenuta nella locazione 1056.
Adesso che abbiamo fatto leggere alla CPU un dato dalla memoria, come procediamo? La prossima istruzione potrebbe essere, per esempio, l'inversione del dato contenuto nell'accumulatore. Questa operazione consiste nel cambiare ogni 0 in 1 e viceversa e l'istruzione, come si è detto, è memorizzata nella locazione 1056. Poniamo che il codice corrispondente sia 84. Quando questo valore viene letto ed interpretato dalla CPU, il dato presente nell'accumulatore viene invertito: quindi, se il valore era 96 (01100000 binario), dopo l'operazione esso diverrà 10011111. Poiché l'istruzione per l'inversione dell'accumulatore è costituita di un solo byte, la CPU sa che la prossima istruzione è alla locazione 1057. Questo metodo di indirizzamento della memoria è uno dei tanti disponibili per il programmatore, per la lettura di un dato. Il principio di funzionamento della nostra CPU immaginaria è simile in quasi tutti i microprocessori e le differenze consistono, generalmente, nel differente codice impiegato da CPU di marche diverse.
Anche l'indirizzamento dei canali di I/O (Input/Output) funziona in maniera molto simile a quella descritta, anche se nella maggior parte dei chip l'indirizzo è costituito da soli 8 bit, cosicché è consentito di indirizzare un massimo di 256 diversi canali di ingresso/uscita (del resto più che sufficienti per le applicazioni più comuni).

La decodifica degli indirizzi

Le 16 linee a disposizione sul bus degli indirizzi possono selezionare fino a 65.536 singole locazioni di memoria. Le combinazioni di 0 ed 1 vengono decifrate da appositi decodificatori d'indirizzi, costituiti da semplici porte logiche o da speciali circuiti interni alle memorie. L'illustrazione mostra come sia possibile, usando due sole linee, selezionare una (e soltanto una) tra quattro linee di uscita



La decodifica dell'indirizzo operata dalla CPU (sia che riguardi una locazione di memoria o un canale di I/O) deve sempre permettere l'accesso ad un'unica locazione (o canale di I/O), lasciando tutte le altre inattive: questo meccanismo viene indicato col termine di "abilitazione".
Quando la selezione degli indirizzi avviene mediante un piccolo numero di linee, è possibile impiegare dei circuiti logici elementari: il principio di funzionamento di un decodificatore da due a quattro linee è mostrato nell'illustrazione.
Il circuito viene spesso impiegato nella decodifica dei canali di I/O, ma, poiché la sua complessità cresce notevolmente all'aumentare del numero di linee da controllare, vengono impiegati degli speciali circuiti interni alle stesse memorie. In questi casi, infatti, può essere necessario indirizzare fino a 65.536 diverse locazioni di memoria.

Le funzioni intrinseche

Il BASIC contiene molte funzioni che facilitano il lavoro del programmatore

I metodi disponibili, per calcolare la radice quadrata in un programma, sono più di uno: il meno efficace è senz'altro quello di costruire una tabella di valori fissi (simile alle tabelline pitagoriche imparate a scuola). Fortunatamente, però, quasi tutti i BASIC sono dotati di una funzione intrinseca, di facilissimo impiego, che serve proprio a questo scopo. Vediamone il funzionamento:

```
10 REM PROGRAMMA PER IL CALCOLO DELLA
20 REM RADICE QUADRATA DI UN NUMERO
30 PRINT "DIGITARE IL NUMERO DEL QUALE SI"
40 PRINT "DESIDERA LA RADICE QUADRATA"
50 INPUT N
60 LET A = SQR(N)
70 PRINT "LA RADICE QUADRATA DI "N;" È"; A
80 END
```

Si provi questo breve programma, e si potrà verificare che effettivamente viene calcolata la radice quadrata di qualsiasi numero digitato.
In BASIC una "funzione" è indicata con una parola (nel nostro caso SQR), seguita da un'espressione, racchiusa tra parentesi, sulla quale la particolare funzione deve operare. Nel nostro programma, N è il numero, immesso da tastiera, del quale si desidera trovare la radice quadrata. Nella linea 60 alla variabile A viene assegnato il valore della radice quadrata di N. La PRINT della linea 70 visualizza il risultato.
L'espressione racchiusa tra parentesi si chiama "argomento" della funzione, e può anche non essere una variabile; si provi:

```
10 PRINT SQR(25)
20 END
```

Queste istruzioni servono a visualizzare la radice quadrata di 25. Possiamo anche usare argomenti più complessi:

```
10 LET A = 10
20 LET B = 90
30 LET C = SQR(A + B)
40 PRINT C
50 END
```

Questo programma si può abbreviare in:

```
10 LET A = 10
20 LET B = 90
30 PRINT SQR(A + B)
40 END
```

Le funzioni "intrinseche" si possono considerare come piccoli sottoprogrammi, di uso generico, incorporati nel BASIC. La quantità e la qualità delle funzioni disponibili variano nelle diverse versioni di BASIC e, in alcune di queste, il programmatore ha la possibilità di definire un certo numero di funzioni a seconda delle necessità contingenti. Ora vedremo qualche altra utile funzione, distinguendo tra funzioni numeriche (quelle che hanno per argomento valori numerici) e quelle per il trattamento delle stringhe alfanumeriche (nelle quali l'argomento è una serie di caratteri).
In un precedente programma, che abbiamo chiamato DECORA (vedere a pag. 77), avevamo incontrato un piccolo problema: il valore che indicava il numero di piastrelle occorrenti poteva non essere un numero intero, per esempio: 988,24. In simili applicazioni, è più conveniente arrotondare il risultato al valore intero più vicino. In BASIC esiste una speciale funzione per far questo:

```
10 PRINT "DIGITA UN NUMERO NON INTERO"
20 INPUT N
30 PRINT "LA PARTE INTERA DEL NUMERO È:";
40 PRINT INT(N)
50 END
```

Immettendo, per esempio, il valore 3.14<CR> (attenzione ad usare il punto al posto della virgola, perché la notazione è quella anglosassone!), si ottiene il messaggio:

```
LA PARTE INTERA DEL NUMERO È: 3
```

Nel caso del programma DECORA, dovremmo sommare 1 al risultato così ottenuto, per avere sufficienti piastrelle.
In altre circostanze, può essere utile conoscere il segno del numero, ossia sapere se esso è positivo o negativo o se vale zero:

```
10 PRINT "DIGITA UN NUMERO";
20 INPUT N
30 LET S = SGN(N)
40 IF S = — 1 THEN GOTO 100
50 IF S = 0 THEN GOTO 120
60 IF S = 1 THEN GOTO 140
100 PRINT "IL NUMERO È NEGATIVO"
110 GOTO 999
120 PRINT "IL NUMERO VALE ZERO"
130 GOTO 999
140 PRINT "IL NUMERO È POSITIVO"
150 GOTO 999
999 END
```

Si noti che i valori assegnati dalla funzione SGN alla variabile S nella linea 30, e successi-

vamente verificati nelle linee 40, 50 e 60, possono essere soltanto tre: — 1 (per indicare che l'argomento della funzione è negativo), 0 (l'argomento vale 0) ed 1 (l'argomento è positivo). L'impiego della funzione SGN ci risparmia la scrittura di almeno tre linee di programma. Infatti, al posto della sola linea 30, avremmo potuto usare:

```
IF N < 0 THEN LET S = — 1
IF N = 0 THEN LET S = 0
IF N > 0 THEN LET S = 1
```

Quindi, ai fini di risparmiare tempo e fatica, conviene usare, se possibile, le funzioni intrinseche del BASIC.
Ecco qualche altra comoda funzione: la ABS, che "restituisce" il valore assoluto di un numero (ossia un valore positivo anche se il numero è negativo). Ad esempio, il valore assoluto di — 9 è 9. Si provi:

```
10 LET X = — 9
20 LET Y = ABS(X)
30 PRINT Y
40 END
```

La funzione MAX trova il valore massimo tra due numeri:

```
10 LET X = 9
20 LET Y = 7
30 LET Z = X MAX Y
40 PRINT Z
50 END
```

La funzione MIN è simile alla MAX, ma trova il valore minimo:

```
10 PRINT "DIGITA UN NUMERO";
20 INPUT X
30 PRINT "DIGITA UN ALTRO NUMERO";
40 INPUT Y
50 LET Z = X MIN Y
60 PRINT Z
70 END
```

Si noti che queste ultime due funzioni non hanno un solo argomento ma due e che questi non sono racchiusi tra parentesi. Molte versioni di BASIC possiedono anche altre funzioni più sofisticate, quali la LOG (per trovare il logaritmo di un numero), la SIN, la COS e la TAN (per trovare, rispettivamente, il seno, il coseno e la tangente di un valore). Vedremo più avanti l'utilità delle funzioni trigonometriche.
Passiamo adesso ad esaminare le funzioni per il trattamento delle stringhe alfanumeriche. Abbiamo già usato alcune di queste nel nostro programma per il riordino alfabetico di nominativi (vedere a pag. 134), ma senza spiegarne a fondo il funzionamento. Si provi:

```
10 LET A$ = "COMPUTER"
20 LET N = LEN(A$)
30 PRINT "LA PAROLA"; A$; "CONTIENE"; N;
40 PRINT "CARATTERI"
50 END
```

A cosa serve sapere quanti caratteri ci sono in una stringa? Il programma che segue è una delle applicazioni di questa funzione:

```
5 REM FRAZIONAMENTO DI UNA PAROLA
10 LET A$ = "MARCO"
20 FOR L = 1 TO 5
30 LET B$ = LEFT$(A$,L)
40 PRINT B$
50 NEXT L
60 END
```

Facendo "girare" il programma, sullo schermo si ottiene:

```
M
MA
MAR
MARC
MARCO
```

Il programma impiega la funzione LEFT$ per l'estrazione di una parte dei caratteri dalla stringa. I due argomenti della LEFT$ sono: il nome della stringa (seguito da una virgola) ed il numero di caratteri da estrarre, partendo da sinistra. Nell'esempio, A$ contiene la stringa MARCO, cosicché LEFT$(A$,1) estrae il carattere M e LEFT$(A$,2) estrae MA. Abbiamo usato un ciclo nel quale viene incrementato l'indice L (da 1 a 5) ed abbiamo usato L come secondo argomento per la funzione. Ovviamente, sapevamo già che la lunghezza della stringa era 5, ma vedremo come modificare il programma in modo che funzioni con parole di qualsiasi lunghezza. Prima diamo qualche esempio sull'impiego della funzione LEN:

```
10 REM PROVA DELLA FUNZIONE 'LEN'
20 PRINT LEN("COMPUTER")
30 END
```

INTRODUZIONE AL FORTH

Versione BASIC:

```
100 REM VISUALIZZARE 6 VOLTE "SALVE!" IN
    BASIC
110 FOR X = 1 TO 6
120 PRINT "SALVE!"
130 NEXT X
140 END
(RUN per eseguire)
```

Versione FORTH:

```
(VISUALIZZARE 6 VOLTE "SALVE!" IN FORTH)
: GRIDA ."SALVE!"; .
: CORO 6 0 DO GRIDA LOOP;
(CORO per eseguire)
```

Ambedue i programmi svolgono la stessa funzione, ma la versione BASIC assomiglia ad una ricetta, mentre quella in FORTH ad una formula magica!
In FORTH si inizia con un elenco di parole di comando (chiamato, appunto, dizionario da "insegnare" al FORTH. Nel nostro esempio, al dizionario vengono aggiunte due nuove parole: GRIDA è definita come variabile alfanumerica da visualizzare, mentre CORO è definita come un misto tra "primitive" (parole già definite nel dizionario) e la nuova parola GRIDA.
Il FORTH, inoltre, dispone di una memoria (chiamata lo stack) e della possibilità di elaborare numeri nello stack. Il FORTH esegue le stesse operazioni aritmetiche e logiche del BASIC, ma impiegando, appunto, questo deposito temporaneo invece di espressioni algebriche.
Il FORTH può sembrare un rompicapo (molti programmatori odiano questo linguaggio), ma offre un'incredibile aiuto nella programmazione. La sua versalità deriva dalla possibilità di definire nuove parole di comando, a seconda della particolare applicazione del programma. Il FORTH è particolarmente adatto per programmare piccoli robot, per esempio, perché il programmatore ha la facoltà di creare comandi come: PRENDI, SPOSTATI, TROVA, SEGUI, RITORNA, ecc.

Il computer visualizza il numero 8, che è, appunto la lunghezza della stringa. Ripetiamo l'esperimento:

```
10 REM LUNGHEZZA DI UNA STRINGA
20 LET A$ = "IL MIO COMPUTER"
30 LET L = LEN(A$)
40 PRINT L
50 END
```

Il risultato questa volta è 15: tanti sono, infatti, i caratteri racchiusi tra i doppi apici (si ricordi che anche gli spazi contano come caratteri!). Siamo adesso in grado di apportare alcune modifiche al programma per il frazionamento delle parole:

```
10 REM FRAZIONAMENTO DI UNA PAROLA
20 PRINT "IMMETTI UNA PAROLA"
30 INPUT A$
40 LET N = LEN(A$)
50 FOR L = 1 TO LEN(A$)
60 LET B$ = LEFT$(A$,L)
70 PRINT B$
80 NEXT L
90 END
```

Ad ogni "passaggio" del ciclo, L aumenta di 1, fino ad arrivare ad N, che è la lunghezza della stringa. Se immettiamo la parola "DISCHETTO", la linea 40 equivale a: LET N = LEN ("DISCHETTO"), così N vale 9. Al primo passaggio, L vale 1 e la linea 60 equivale a: LET B$ = LEFT$ ("DISCHETTO", 1), di conseguenza B$ contiene il primo carattere a sinistra della stringa, cioè "D".
Il valore di L viene incrementato a 2 nel secondo passaggio, ottenendo che a B$ venga assegnato il valore "DI", e così via.
Poiché N vale 9, ci saranno esattamente 9 passaggi, all'ultimo dei quali B$ conterrà l'intera parola immessa.
La funzione LEFT$ ha una "gemella", la RIGHT$, che serve per estrarre i caratteri di una stringa, partendo da destra.
Vediamo, infine, la funzione INSTR (anch'essa usata nel programma per il riordino alfabetico, per separare il nome dal cognome). Con la INSTR si ottiene la posizione nella quale un certo carattere (o un gruppo di caratteri) appare per la prima volta all'interno di una stringa. Per esempio:

```
10 LET A$ = "PORTOGALLO"
20 LET P = INSTR(A$, "TOGA")
30 PRINT P
40 END
```

Prima di far girare il programma, si provi ad indovinare il valore che verrà assegnato a P. Ricordandosi che la INSTR fornisce la posizione iniziale della "sotto-stringa" (TOGA) nella variabile A$, il valore di P sarà dunque 4 (contando da sinistra a destra). In alcune versioni di BASIC il nome di questa funzione è INDEX, ma il funzionamento è identico. Un altro esempio:

```
10 REM RICERCA DEL PRIMO SPAZIO IN UNA STRINGA
20 LET A$ = "IL MIO COMPUTER"
```

Basic & Basic

TAB

Sullo Spectrum non è necessario racchiudere tra parentesi gli argomenti di funzioni come la TAB, a meno che essi non siano espressioni complesse e non semplici valori o nomi di variabili

LEFT$

Nessuna di queste funzioni è disponibile sullo Spectrum, ma è possibile definirle usando l'istruzione DEF FN:

```
9900 DEF FN L$(X$, N) = X$(TO N)
9910 DEF FN R$(X$, N) = X$(LEN(X$) — N +
1 TO)
9920 DEF FN M$(X$, P, N) = X$(P TO P + N
— 1)
```

RIGHT$

MID$

Adesso si può usare:
FN L$(X$, N) al posto di LEFT$(X$, N)
FN R$(X$, N) al posto di RIGHT$(X$, N)
FN M$(X$, P, N) al posto di MID$(X$, P, N)

INSTR

Questa istruzione non è disponibile sui seguenti computer: Spectrum, VIC-20, Commodore 64 ed Oric-1, ma è possibile ricorrere a questo metodo:
sostituire:

```
20 LET P = INSTR(A$, "TOGA")
```

con:

```
20 LET X$ = A$ : LET Z$ = "TOGA" : GOSUB
9930
```

ed aggiungere:

```
9929 STOP
9930 LET U = 0 : LET X = LEN(X$) : LET V =
LEN(Z$)
9940 FOR W = 1 TO L — V + 1 : IF MID$(X$,
W, V) = Z$ THEN LET U = W
9950 IF U <> 0 THEN LET W = L — V + 1
9960 NEXT W : RETURN
```

(Per lo Spectrum sostituire la MID$ con la funzione descritta più sopra).

```
30 LET P = INSTR(A$," ")
40 PRINT P
50 END
```

Il secondo argomento della INSTR (linea 30) è un carattere spazio racchiuso tra doppi apici. Il programma dà come risultato 3, perché questa è la posizione del primo spazio nella stringa A$. Si provi a cambiare la linea 30 in:

```
30 LET P = INSTR(A$, "C")
```

Quale sarà, questa volta il risultato?
Vediamo, infine, un'utilissima funzione da adoperare assieme all'istruzione PRINT:

```
10 PRINT "QUESTA LINEA È NORMALE"
20 PRINT TAB(5); "QUESTA LINEA È TABULATA"
30 END
```

Il risultato che si ottiene da questo programma è la visualizzazione di due righe di testo, una delle quali è "tabulata" alla colonna 5. L'effetto della TAB è simile a quello ottenuto con un tabulatore in una normale macchina da scrivere.
Ecco un altro esempio:

```
10 REM USO DELLA FUNZIONE TAB
20 PRINT "DIGITA IL NUMERO DELLA COLONNA"
30 PRINT "PER LA TABULAZIONE";
40 INPUT T
50 LET W$ = "TABULAZIONE"
50 PRINT TAB(T); W$
60 END
```

Può essere utile, dopo queste spiegazioni, riguardare come sono state impiegate tutte queste funzioni nel programma di pagina 136.

Esercizi

*** Ciclo 1.** Che cosa viene visualizzato da questo programma?

```
10 LET A = 500
20 FOR L = 1 TO 50
30 LET A = A — 1
40 NEXT L
50 PRINT "IL VALORE DI A È:"; A
```

*** Ciclo 2.** Che cosa appare sullo schermo usando questo programma?

```
10 REM
20 REM QUESTO È UN CICLO DI RITARDO
30 REM QUANTO TEMPO IMPIEGA?
40 REM
50 PRINT "INIZIO"
60 FOR X = 1 TO 5000
70 NEXT X
80 PRINT "FINE"
90 END
```

*** Ciclo 3.** Che risultato otterremo facendo girare questo programma e digitando il valore 60?

```
10 PRINT "PENSA UN NUMERO E DIGITALO"
20 INPUT N
30 LET A = 100
40 FOR L = 1 TO N
50 LET A = A + 1
60 NEXT L
70 PRINT "IL VALORE DI A È:"; A
80 END
```

*** Ciclo 4.** Che cosa accade se facciamo partire questo programma?

```
10 PRINT "MI PIACE IL BASIC"
20 GOTO 10
30 END
```

*** Ciclo 5.** Quale sarà il risultato di questo programma?

```
10 FOR Q = 1 TO 15
20 PRINT "IO SONO IL TUO COMPUTER"
30 NEXT Q
40 END
```

*** READ-DATA 1.** Che risultato otterremo da questo programma?

```
10 READ X
20 READ Y
30 READ Z
40 PRINT "PROVA DELLE ISTRUZIONI READ-DATA"
50 DATA 50,100,20
60 PRINT X + Y + Z
70 END
```

*** READ-DATA 2.** E da questo?

```
100 FOR L = 1 TO 10
110 READ X
120 PRINT "X = "; X
130 NEXT L
140 DATA 1,2,3,5,7,11,13,17,19,23
```

(Le risposte sono nella prossima lezione)

Risposte agli esercizi di pagg. 136, 137
Variabili

(A) (B6) ~~Z2~~ D$ ~~X6~~ X$ (A12) (D9) (Q81) (Q5) ~~3E~~ H$

Nota bene: le risposte dipendono molto dalla versione di BASIC (alcune non consentono l'uso di suffissi numerici).

Aritmetica 1
```
10 LET B = 6
20 PRINT B
```

Aritmetica 2
```
10 LET A = 5
20 LET B = 7
30 LET C = 9
40 LET D = A + B + C
50 PRINT D
```

Aritmetica 3
17

Aritmetica 4
25
25

Confronto 1
5

Confronto 2
601 (restando nel campo dei numeri interi)

Confronto 3
10000

PRINT 1
```
PRINT "IL VALORE DI T È"; T
```

PRINT 2
```
640 PRINT "SPIACENTE, MA IL PUNTEGGIO DI "; S; "È TROPPO BASSO"
```

PRINT 3
Questo esercizio conteneva intenzionalmente un errore: infatti, il punto e virgola nella linea 210 provoca un errore del tipo "SYNTAX ERROR". La forma esatta è:

```
200 LET A$ = "IL CORSO 'IL MIO COMPUTER'?"
210 LET B$ = "DIMMI, TI PIACE"
220 PRINT B$; A$
```

che produce il messaggio:

```
DIMMI, TI PIACE IL CORSO 'IL MIO COMPUTER'?
```

INPUT 1
6

INPUT 2
```
DIGITA IL TUO NOME
(input del nome)
CIAO, (nome). IO SONO IL TUO COMPUTER
```

Il Commodore VIC-20

Anatomia di microcomputer di diffusione europea: tecnologie a confronto

La Commodore Business Machines commercializzò, nel 1977, uno dei primi personal computer: il Personal Electronic Transactor (PET). Nel 1981 ha lanciato sul mercato un altro apparecchio, dal costo notevolmente inferiore, ma dalle caratteristiche pressoché identiche: il Vic-20. Non solo è stato impiegato il medesimo modello di CPU, la 6502, ma anche il BASIC, fornito su memorie ROM, è lo stesso.

Una delle principali differenze tra i due computer consiste nelle maggiori capacità grafiche del Vic-20 (la sigla Vic proviene, appunto, da Video Interface Chip, ossia Chip d'interfaccia video). Il colore di ciascun carattere o simbolo grafico può essere scelto tra una gamma di 8 colori, mentre lo sfondo ne ha a disposizione 16.

Il set di caratteri visualizzabili è incredibilmente vasto: oltre a comprendere le maiuscole e le minuscole, sono anche inclusi due gruppi di caratteri grafici, selezionabili con appositi tasti, 62 in tutto. Esistono anche 4 tasti speciali, programmabili dall'utente, per selezionare fino ad otto particolari funzioni. La tastiera ha un aspetto molto elegante ed è comoda da usarsi, essendo stata progettata seguendo i moderni criteri ergonomici.

L'unico grande svantaggio del Vic-20 è la scarsa memoria disponibile (soli 5 Kbyte, dei quali appena 3,5 disponibili all'utente dopo il caricamento del sistema operativo). Sono, comunque, disponibili schede aggiuntive, non necessariamente della Commodore, per espandere la memoria a 32 Kbyte.

Al Vic-20 si può collegare, mediante le interfacce fornite, una vasta gamma di periferiche: joystick, penne ottiche, cartucce contenenti programmi, espansioni di memoria, stampanti, unità a dischi o a nastri e video. È disponibile anche una porta seriale RS232 per l'eventuale utilizzo di un modem o di una stampante seriale. Molti degli accessori adatti al più recente Commodore 64 (vedere a pag. 33) possono essere collegati anche al Vic-20, senza apportare alcuna modifica.

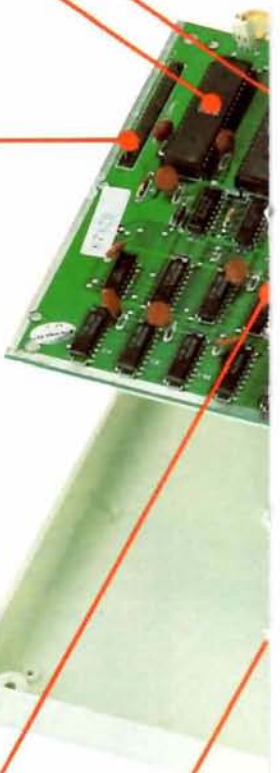


Connettore per registratore
Il Vic-20, come del resto gli altri computer Commodore, necessita di uno speciale registratore, che viene collegato a questa presa

Porta utente
Questo connettore a 24 piedini permette di collegare una vasta gamma di accessori di tipo seriale

Interfaccia per periferiche
Questi chip controllano tutte le funzioni di I/O, comprese le conversioni da seriale a parallelo e viceversa

Connettore per tastiera
La tastiera viene collegata a questo connettore

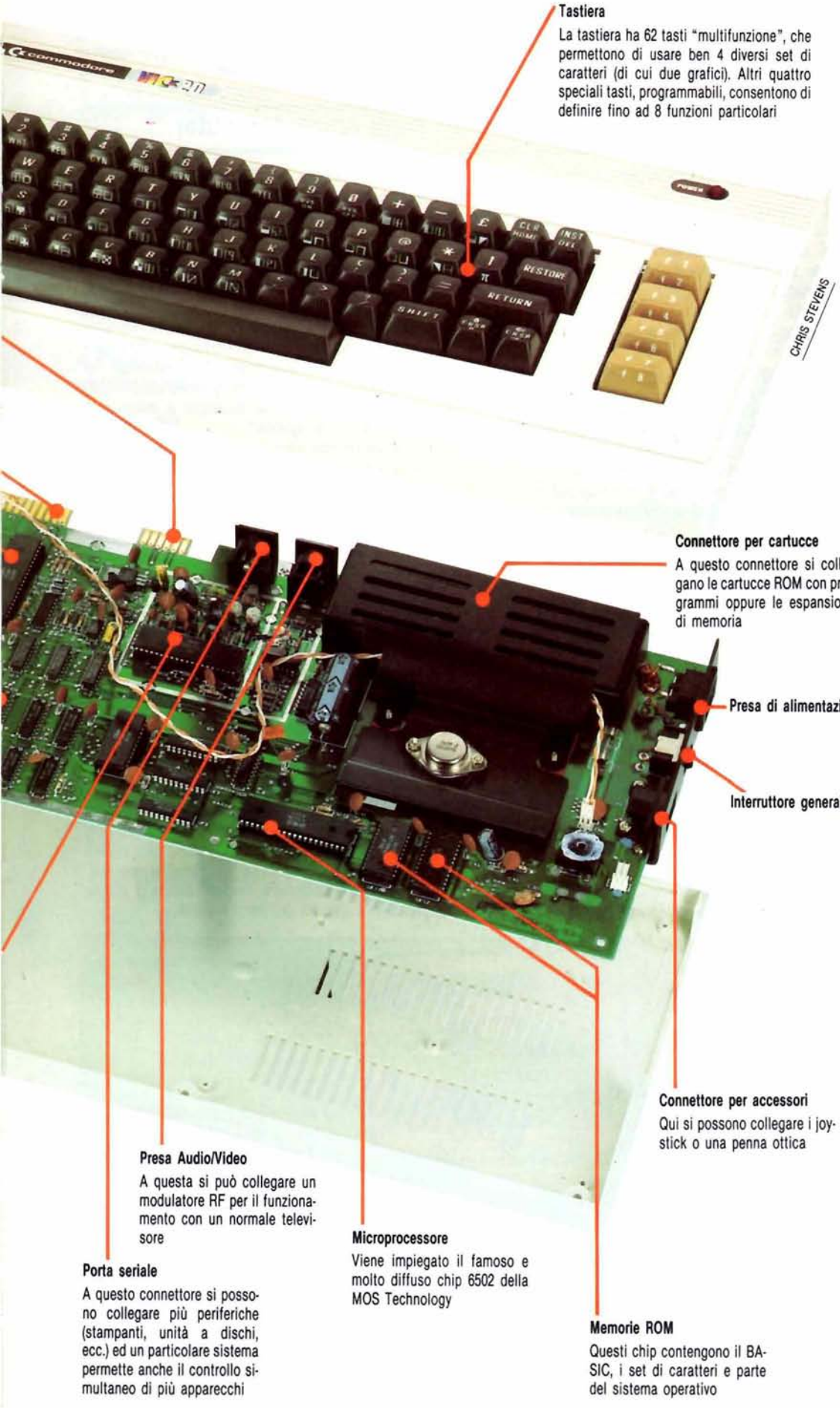


Chip di interfaccia video
Questo speciale chip controlla tutte le operazioni di visualizzazione e del generatore di effetti sonori a tre voci

Memoria utente (RAM)
Il Vic-20 ha una memoria RAM di 5 Kbyte. Con schede aggiuntive, esterne, si possono raggiungere i 32 Kbyte



Le periferiche per il Vic-20
La gamma di accessori della Commodore comprende: registratori a cassette, unità a dischetti ed una discreta scelta di stampanti, oltre agli immancabili joystick. Esistono, sul mercato, anche altri accessori, non prodotti dalla Commodore, tra i quali utilissime espansioni di memoria



Tastiera

La tastiera ha 62 tasti "multifunzione", che permettono di usare ben 4 diversi set di caratteri (di cui due grafici). Altri quattro speciali tasti, programmabili, consentono di definire fino ad 8 funzioni particolari

Connettore per cartucce

A questo connettore si collegano le cartucce ROM con programmi oppure le espansioni di memoria

Presa di alimentazione

Interruttore generale

Connettore per accessori

Qui si possono collegare i joystick o una penna ottica

Presa Audio/Video

A questa si può collegare un modulatore RF per il funzionamento con un normale televisore

Porta seriale

A questo connettore si possono collegare più periferiche (stampanti, unità a dischi, ecc.) ed un particolare sistema permette anche il controllo simultaneo di più apparecchi

Microprocessore

Viene impiegato il famoso e molto diffuso chip 6502 della MOS Technology

Memorie ROM

Questi chip contengono il BASIC, i set di caratteri e parte del sistema operativo

COMMODORE VIC-20

DIMENSIONI

mm. 404 x 216 x 75

PESO

1820 grammi

CPU

6502

CLOCK

1 MHz

MEMORIA

5 Kbyte di RAM (estendibile esternamente a 32 Kbyte)

VIDEO

23 linee di 22 caratteri, oppure 184 x 176 punti per la grafica ad alta risoluzione. 16 colori disponibili

INTERFACCE

Connettore per audio/video, connettore per registratore a cassette, 2 connettori seriali, connettore per cartucce, connettore per accessori da gioco

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

ASSEMBLER e speciali comandi BASIC

ACCESSORI COMPRESI

Alimentatore, cavo di connessione per TV ed un manuale

TASTIERA

Tastiera tipo "macchina da scrivere" dotata di 62 tasti più 4 tasti "funzione"

DOCUMENTAZIONE

Il manuale è di facile lettura e sufficientemente comprensibile per un principiante, ma non vengono trattate in maniera approfondita le grandi capacità di questo piccolo computer. In compenso, sono in vendita numerosi libri che permettono di colmare gran parte delle lacune

La grafica "a sprite"

Come si creano immagini in movimento sugli home computer

Una delle maggiori attrattive degli home computer è la loro possibilità di produrre immagini grafiche fisse e moventi, che sono vere e proprie "animazioni". In quasi tutti i microcomputer è possibile, infatti, tracciare sullo schermo linee, cerchi e figure, e colorare sia lo sfondo che le immagini in primo piano. Per creare dei giochi di simulazione o d'azione è necessario riprodurre immagini in movimento. Il metodo più semplice consiste nel costruire una serie di immagini fisse, che vengono poi fatte scorrere sullo schermo in rapida successione, per dare l'illusione del movimento. Le immagini televisive sono prodotte con un metodo analogo.

Velocità d'azione

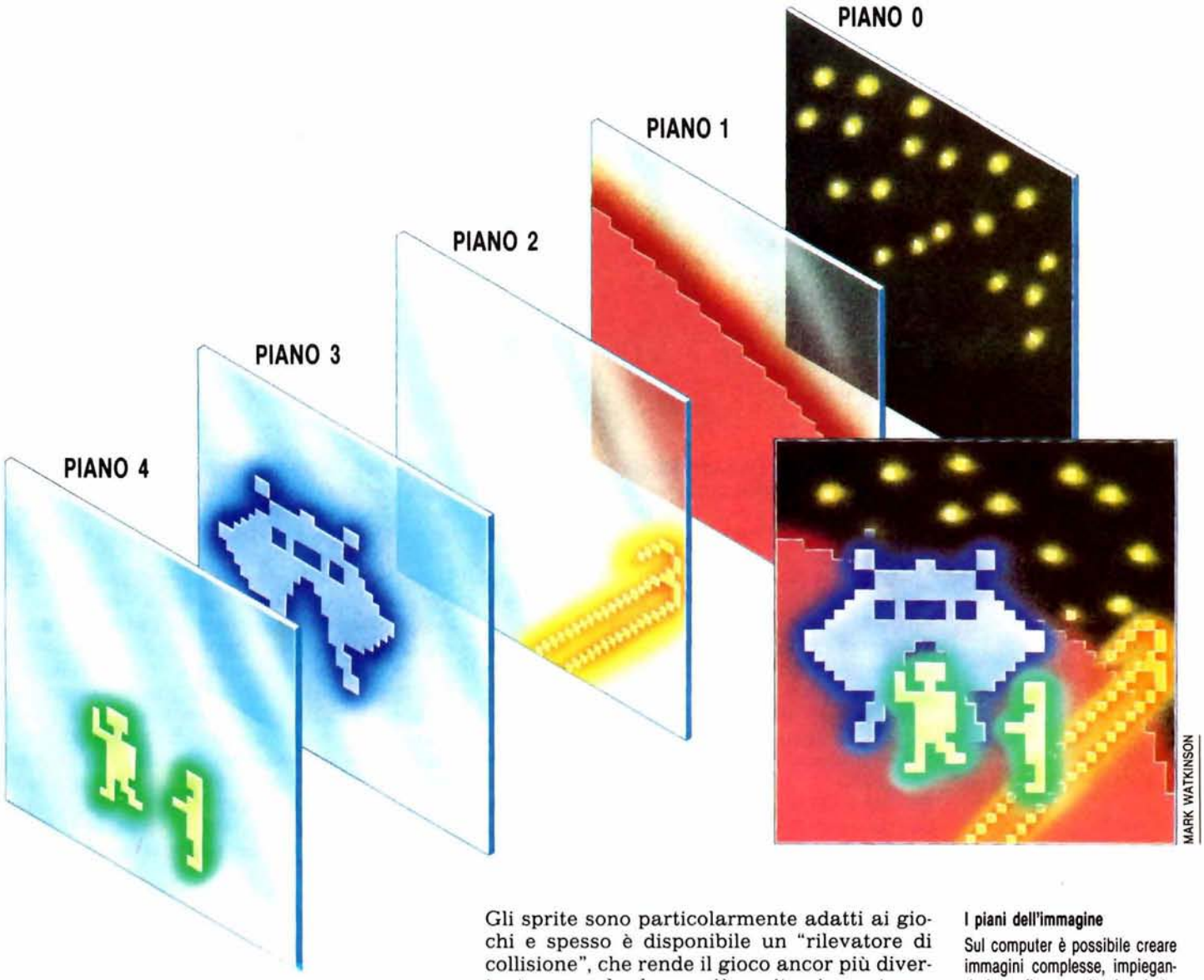
Un altro sistema per creare l'illusione di un movimento è quello di visualizzare un carattere, cancellarlo, e subito dopo visualizzarlo di nuovo in una posizione leggermente diversa da quella precedente. Se si vuole ottenere un movimento scorrevole, la distanza fra ciascuna posizione deve essere minima, ed anche il tempo occorrente per visualizzare e cancellare l'immagine deve essere il più breve possibile.
Impiegando il BASIC, spesso i movimenti risultano troppo lenti, quindi generalmente si ricorre all'uso del linguaggio ASSEMBLY, ma in questo caso occorre avere molta pratica per creare immagini senza "sfarfallio".
Inoltre, all'aumentare del numero dei singoli elementi sullo schermo, sorge il problema di controllarne lo spostamento in modo preciso. Alcuni computer, (il Commodore 64, il Sord M5, il TI99/4A e la gamma Atari in particolare), hanno risolto questo problema offrendo una grafica di animazione che utilizza le stesse tecniche adoperate nei giochi elettronici da sala giochi. Questa tecnica è nota col nome di grafica "a sprite".
L'oggetto o la forma che possiamo far muovere liberamente su tutto lo schermo si chiama sprite. L'autonomia di movimento è controllata dai valori depositati in un paio di locazioni di memoria, e che rappresentano le coordinate X e Y (la riga e la colonna) del particolare oggetto visualizzato. Di solito, il valore della X varia da 0 a 255, mentre quello della Y varia da 0 a 191. In alcuni computer è persino possibile specificare la velocità e la direzione del movimento di ciascuno sprite, lasciando al computer il compito di regolare tutto il resto.
La grafica a sprite è generalmente ottenuta per mezzo di uno speciale chip o mediante particolari circuiti interni al computer. In alcuni casi si possono acquistare delle speciali schede aggiuntive da allacciare al computer. La velocità di spostamento di ogni sprite può variare indipendentemente da quella di altri sprite, poiché esistono più piani di visualizzazione. L'immagine risultante sullo schermo, però, appare come unica e composta da più oggetti in movimento.

IL TRENINO

L'immagine del treno, qui illustrato, è stata costruita con tre sprite (la locomotiva più due carrozze) usando un programma chiamato "Spritemaker" su di un Commodore 64. Grazie alle particolari caratteristiche del programma, ciascuno sprite è stato creato usando l'intero schermo. Successivamente, i dati ad esso relativi sono stati registrati su nastro (per altri futuri impieghi) e riletti nella memoria del Commodore 64, dove con opportuni comandi POKE, è stata fissata la posizione, il colore ed, eventualmente, la velocità del movimento di ogni sprite sullo schermo. A ciascun sprite, inoltre, è stato assegnato un livello di priorità, per meglio simulare un effetto tridimensionale (ad esempio: il treno che passa "dietro" alla casetta)



IAN McKINNELL



I piani dell'immagine
Sul computer è possibile creare immagini complesse, impiegando la grafica a sprite: i vari elementi che formano l'immagine vengono collocati su piani diversi, posti consecutivamente uno dietro l'altro. L'immagine risultante sembra riprodotta su di un piano unico, ed il grande vantaggio di questo sistema sta nel fatto che gli oggetti sui vari piani sono liberi di muoversi in modo completamente indipendente. A ciascuno sprite viene assegnato un livello di priorità, affinché, quando due sprite si sovrappongono, sia visibile quello con priorità maggiore, offrendo così un effetto tridimensionale. La maggioranza dei computer dotati di grafica a sprite possiede uno speciale circuito (utilizzabile dai programmi), che serve per intercettare lo "scontro" di due o più sprite, creando così l'immagine (e talvolta anche l'effetto sonoro) di un'esplosione

È anche possibile produrre un effetto tridimensionale, in quanto gli sprite possono passare da un piano all'altro, idealmente più vicino allo spettatore o più lontano da esso. Agli sprite viene assegnato un numero di riferimento, che va da 0 al numero massimo di piani disponibili (nel Sord M5, ad esempio, questo è di 32). Quando due sprite occupano la stessa posizione, sullo schermo appare soltanto quello con il numero di riferimento più basso. Pertanto con un'attenta numerazione degli sprite è possibile ottenere dei buoni effetti tridimensionali; per esempio: l'immagine di un albero viene temporaneamente oscurata alla vista dal passaggio di un trenino, che, a sua volta, viene parzialmente nascosto mentre passa dietro all'immagine di una casetta, posta in un piano intermedio tra lo spettatore ed il treno.
Ovviamente, gli effetti migliori si ottengono colorando opportunamente ciascuno sprite. In alcuni casi anche le dimensioni dello sprite possono essere modificate cambiando il valore conservato in una particolare locazione di memoria.

Gli sprite sono particolarmente adatti ai giochi e spesso è disponibile un "rilevatore di collisione", che rende il gioco ancor più divertente: quando due o più sprite si scontrano sullo schermo (per esempio, quando un missile si scontra con una nave spaziale nemica) opportune istruzioni del programma provocano la visualizzazione di un'esplosione, modificando, al tempo stesso, il punteggio del giocatore.
Prima di poter giocare con gli sprite occorre crearli, adoperando lo stesso sistema usato per definire un nuovo carattere. Le lettere dell'alfabeto, i numeri e i simboli grafici speciali sono conservati in un chip chiamato "generatore di caratteri". Come accennato precedentemente (vedere a pag. 44), i caratteri sono generalmente formati da una matrice di 8 × 8 punti, chiamati "pixel". Le dimensioni di uno sprite variano da computer a computer, ma, di solito, esse equivalgono a quelle di più caratteri normali. Nel Commodore 64, per esempio, la massima dimensione di uno sprite è di 24 pixel in larghezza per 21 in altezza. Il metodo migliore per la costruzione di un nuovo sprite è quello di usare della carta a quadretti, delineando con una matita i margini corrispondenti alla massima possibilità del computer, e disegnando, poi, la forma deside-

Le dimensioni degli sprite

La grafica a sprite è attualmente disponibile soltanto su quattro dei più diffusi home computer: il Commodore 64, il Sord M5, TI99/4A e tutta la gamma Atari.
Sono normalmente disponibili 16 colori, anche se negli Atari se ne possono ottenere addirittura 256, perché ciascun colore ha ben 16 diversi gradi di luminosità. Nei giochi elettronici è spesso utile sapere quando due sprite (per esempio, due astronavi) si scontrano: il rilevatore di collisione serve a questo scopo, facilitando la programmazione dei giochi.
L'effetto tridimensionale è creato per mezzo di diversi piani sovrapposti: più piani ci sono, maggiormente efficace è l'effetto. La dimensione massima di ciascuno sprite è espressa in "pixel". Si possono, inoltre, ingrandire o rimpiccolire gli sprite e muoverli qua e là sullo schermo. Per semplificare il processo di creazione degli sprite, è possibile impiegare uno speciale software

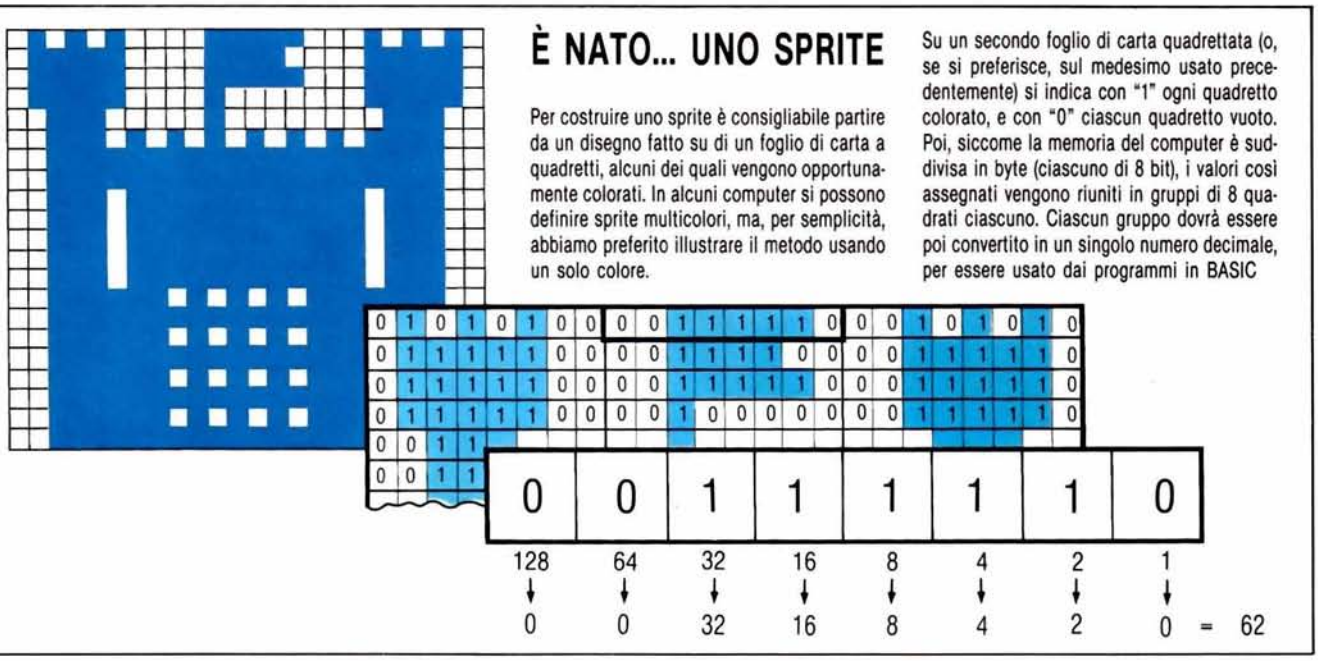
MICROCOMPUTER	COLORI DISPONIBILI	RIVELATORE DI COLLISIONE	PIANI	DIMENSIONI DELLO SPRITE	SOFTWARE
Commodore 64	16	SI	8	24 x 21	SI
Atari 800	256	SI	16	16 x 16	SI
Texas TI 99/4A	16	SI	28	32 x 32	SI
Sord M5	16	SI	32	8 x 8	NO

rata utilizzando i quadretti. Siccome sappiamo che il computer conosce ed impiega soltanto i numeri 1 e 0 (vedere a pag. 27), i quadretti "pieni" rappresenteranno un 1 e le caselle vuote uno 0.
Quasi tutti i dati elaborati dal computer sono espressi sotto forma di byte (un insieme di 8 bit), quindi dovremo consultare il manuale del computer per sapere come occorre suddividere il disegno a quadretti, per ricondurlo al formato richiesto.
Dopo di che, ciascun byte dovrà essere convertito in numeri decimali, affinché se ne possa impiegare il valore in un programma in BASIC.
La serie di numeri decimali risultante da questa operazione definisce la forma dello sprite e viene depositata, usando un programma BASIC, in particolari locazioni di memoria che dipendono dal computer impiegato. Altre istruzioni saranno necessarie per comunicare al computer l'esatta collocazione di questi dati nella memoria.
A questo punto, non rimane che specificare, attraverso semplici comandi, la posizione di ciascuno sprite sullo schermo, cambiarne il colore, ingrandirne o diminuirne le dimensioni e controllare quando due o più sprite si sovrappongono.
Sono disponibili in commercio dei programmi di software chiamati "utilities" (adatti alla maggioranza di computer con grafica a sprite), che rendono la creazione delle immagini meno difficile e noiosa: lo schermo viene suddiviso in quadretti e l'immagine desiderata si costruisce semplicemente muovendo il cursore su questa "griglia". La parte aritmetica viene svolta automaticamente dal computer e i risultati sono conservati negli appositi byte di memoria. Ad operazione compiuta, i quadretti scompaiono dallo schermo e rimane lo sprite, pronto per successive elaborazioni.
La grafica a sprite ha rivoluzionato la creazione di immagini sui computer, offrendo un metodo semplice e molto efficace per costruire giochi e simulazioni che risultano così avvincenti.

È NATO... UNO SPRITE

Per costruire uno sprite è consigliabile partire da un disegno fatto su di un foglio di carta a quadretti, alcuni dei quali vengono opportunamente colorati. In alcuni computer si possono definire sprite multicolori, ma, per semplicità, abbiamo preferito illustrare il metodo usando un solo colore.

Su un secondo foglio di carta quadrettata (o, se si preferisce, sul medesimo usato precedentemente) si indica con "1" ogni quadretto colorato, e con "0" ciascun quadretto vuoto. Poi, siccome la memoria del computer è suddivisa in byte (ciascuno di 8 bit), i valori così assegnati vengono riuniti in gruppi di 8 quadrati ciascuno. Ciascun gruppo dovrà essere poi convertito in un singolo numero decimale, per essere usato dai programmi in BASIC



MARK WATKINSON

Steve Wozniak

L'uomo che ha creato l'Apple



APPLE COMPUTER INC.

Negli Stati Uniti, Steve Wozniak è più conosciuto al grande pubblico come organizzatore di concerti rock all'aperto che non come l'uomo che costruì, da solo e nel garage della propria casa, i famosi APPLE I e II. Nel mondo dell'informatica egli è noto come il "mago dell'elettronica", che, più di ogni altro, ha contribuito ad aumentare la popolarità dei microcomputer. Le macchine prodotte da Wozniak erano le prime ad essere dotate di una grafica a colori, di una tastiera e di un video incorporati. L'APPLE I ha ottenuto un tale successo, che ne sono stati venduti oltre un milione di esemplari.

L'ascesa di Wozniak, da ingegnere automobilistico a miliardario, ha qualcosa della moderna favola californiana. Egli è infatti nato e cresciuto proprio nella Silicon Valley, in California. Suo padre, di professione ingegnere, gli insegnò le regole più elementari dell'elettronica, ma Wozniak è, fondamentalmente, un autodidatta.

Ha trascorso gli anni della sua infanzia giocando con componenti elettronici e sfruttando il suo genio tecnico per fare degli scherzi a scuola. Una volta si costruì un congegno elettronico, che, prendendo spunto da un personaggio di una rivista per ragazzi, egli chiamò la "scatola blu". Questo apparecchio era in grado di riprodurre le frequenze impiegate nel sistema telefonico americano e con esso era possibile telefonare... gratis in ogni parte del mondo. Si dice che Wozniak, usando questo apparecchietto, abbia parlato perfino col Papa in persona!

Wozniak era bravissimo in matematica e in elettronica, ma non finì mai gli studi superiori. Il suo primo lavoro fu come tecnico addetto al progetto di calcolatori nella mastodontica società americana Hewlett Packard. Ma, non avendo qualifiche sufficienti (stando ai canoni della grossa ditta), per applicarsi ai progetti che egli aveva in mente, decise di mettersi a lavorare da solo per produrre un microcomputer, che la Hewlett Packard si rifiutò di comprare. Per niente sconfitto, lasciò la società e con un amico di scuola (e compagno di scherzi) Steve Jobs, riuscì a produrre e vendere 50 esemplari del computer: era nato l'Apple I. La decisione di chiamare sia il computer che la nuova società con il nome "Apple" (che significa "mela") venne presa semplicemente perché Jobs aveva lavorato una volta in un'azienda agricola.

Nel periodo fra il 1975 ed il 1976, all'età di 26 anni, Wozniak si chiuse nel proprio garage, dove, dopo intensi periodi di creatività (che duravano giorni e notti intere) creò l'Apple II. Gli esperti considerano ancora oggi l'Apple un capolavoro di semplicità sia nel progetto che nei circuiti.

Si dice che Steve Wozniak sia capace di leggere i diagrammi dei circuiti di un chip con la stessa facilità con la quale le chiromanti leggono il destino nelle mani della gente. Una delle principali innovazioni dell'Apple II è la semplicità dell'unità disco. Prima dei progetti dovuti a Wozniak, i circuiti contenevano 30 chip: egli riprogettò il tutto usando solo 5 chip. La sua inventiva non è indirizzata tanto verso nuovi prodotti quanto verso la semplificazione dei componenti ed il modo col quale essi vengono montati. Ciò ha reso possibile la diffusione del computer anche per applicazioni domestiche.

Wozniak non si è mai dedicato all'aspetto commerciale dei computer: è sempre stato Steve Jobs, il suo socio, ad essere responsabile della commercializzazione degli Apple e della costituzione della società. La Apple, attualmente, ha ben 3300 persone alle proprie dipendenze in tutte le parti del mondo e si dice che abbia reso miliardarie almeno 50 persone. Wozniak possiede oggi soltanto il 4% delle azioni e non si è mai interessato né impegnato negli aspetti amministrativi: preferisce giocare con i computer e sfornare nuove idee.

Nell'estate 1983, dopo una pausa di 2 anni (dedicata per la maggior parte all'organizzazione di festival rock) Wozniak è ritornato alla Apple per elaborare alcuni nuovi progetti. Nessuno sa bene su che cosa stia lavorando, ma pare che si tratti di software applicativo. Egli è inoltre impegnato al progetto di una versione dell'Apple II con grafica ad alto livello, con la quale è possibile produrre anche cartoni animati. Ma Wozniak non si limita a rendere più veloci e migliori le macchine esistenti: egli è del parere che nel futuro sarà possibile progettare dei microcomputer dotati d'intelligenza, i quali, utilizzando un unico programma, saranno in grado di imparare qualsiasi cosa. Dovremo aspettare per vedere cosa riuscirà ad inventare questo ingegnere anticonformista, dotato di una fertile immaginazione.

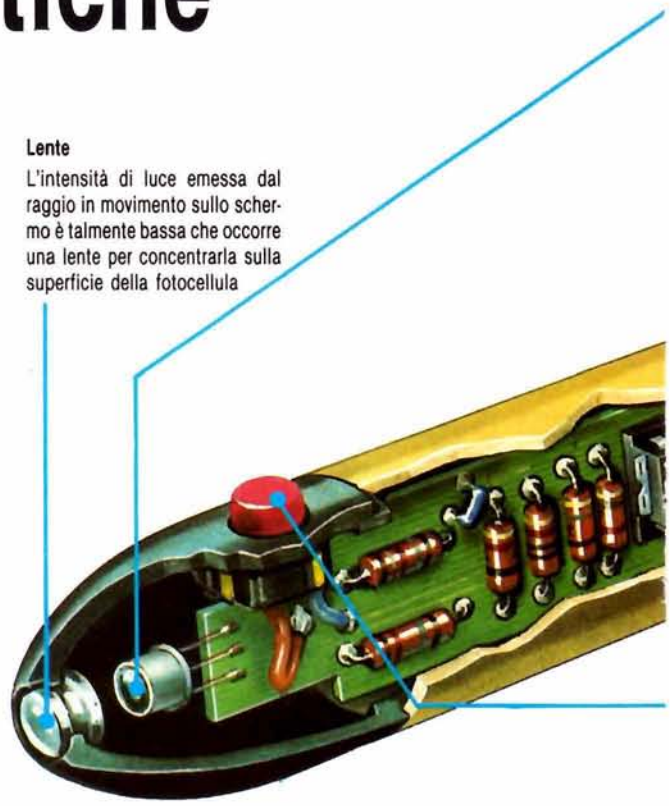
L'amministratore delegato

Steven Jobs, Amministratore Delegato della Apple Computer Inc., è socio ed amico di Wozniak. Se è vero che Wozniak è il genio elettronico della coppia, Jobs è il ragazzo prodigio degli affari. Nel 1975, quando l'Apple I fu progettato, Jobs aveva solo 20 anni, ma intuì immediatamente il potenziale di vendita della macchina. Inizialmente si rivolse agli hobbysti ed agli appassionati di computer, ma non ci volle molto tempo perché si dedicasse al mercato dei personal computer. Wozniak creò il computer, ma la produzione su grande scala e la vendita furono compito di Jobs. Il computer Apple II viene confezionato "pronto per l'uso", completo di tutti gli accessori in modo tale che chiunque è immediatamente in grado di usarlo in casa propria. I primi fondi per la costituzione della società, quando ancora Wozniak e Jobs lavoravano nel garage, furono ricavati dalla vendita del furgoncino Volkswagen di Jobs e del calcolatore programmabile di Wozniak. Ben presto Mike Marakkula, un giovane miliardario, finanziò quello che gli sembrò un progetto interessante. Quando nel 1980 la Apple divenne una società per azioni, fu una grande sorpresa, ma non deluse le aspettative: vennero venduti computer per un valore di 583 milioni di dollari nel solo 1982.

Lo sviluppo fenomenale della Apple Corporation è stato acclamato e premiato negli Stati Uniti. Steve Jobs è apparso sulla copertina della famosa rivista "TIME", indicato come uno dei più giovani uomini d'affari di grande successo negli Stati Uniti.

Le penne ottiche

Il funzionamento di questo utilissimo accessorio



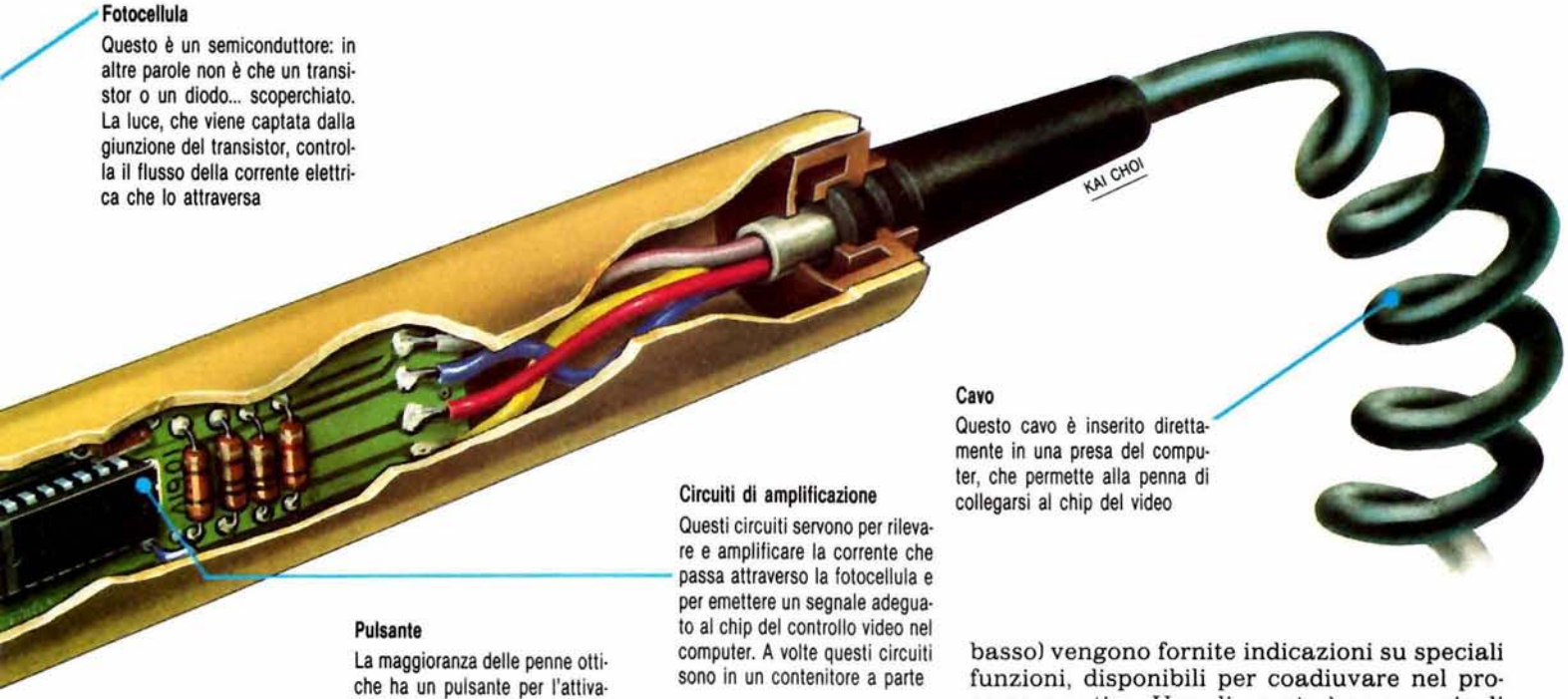
Uno dei principali motivi, che rendono la gente diffidente nei confronti dei computer, è attribuibile ad una sorta di "fobia della tastiera". Ciò dipende, forse, dal fatto che la tastiera di un computer assomiglia a quella di una macchina da scrivere e che non tutti si sentono provetti dattilografi o, peggio ancora, provano soggezione di fronte a tasti dall'aspetto poco familiare. Esiste una soluzione a questo problema: la penna ottica.
La penna ottica assomiglia molto ad una normale penna, salvo per la presenza di un filo elettrico, che serve a collegarla al computer. Ogni volta che la penna viene appoggiata sullo schermo del monitor e viene azionato il pulsante su di essa (in alcuni sistemi basta premere la penna contro lo schermo per attivare un interruttore interno), il computer è in grado di rilevare con esattezza la posizione relativa della penna sullo schermo.
Il principio di funzionamento non è molto complicato: il fotorilevatore montato sulla punta della penna genera un impulso elettrico ogniqualvolta esso rivela il passaggio del raggio elettronico, creato dal tubo a raggi catodici del monitor, durante la scansione dello schermo. Un circuito interno al chip di controllo del video calcola la posizione del raggio elettronico, nel preciso istante nel quale la penna ottica emette l'impulso.
Un impiego molto comune della penna ottica è, dunque, la selezione di una, fra più opzioni presentate sullo schermo da parte di un programma. Il computer, una volta ricevuto il segnale, è in grado di comunicare al programma la posizione della penna e ciò serve per stabilire quale carattere o simbolo sia stato scelto. Il metodo di presentare dei "menu" di opzioni, sullo schermo è molto frequente nei programmi applicativi: un menu è semplicemente un elenco di possibilità fra le quali l'utente può scegliere (come accade col menu di un ristorante). In un programma di contabilità, per esempio, un menù potrebbe contenere le seguenti voci:

1) Gestione della prima nota
2) Stampa del riepilogo contabile
3) Stampa degli estratti conto

Normalmente, l'utente indica la sua scelta premendo un tasto (1, 2 o 3) oppure digitando una parola di risposta. Se si ha una penna ottica, invece, basta appoggiarne la punta sull'opzione prescelta ed il computer, a conferma di aver ricevuto il comando, emette un segnale sonoro o un particolare simbolo sullo schermo.
Ci sono alcuni programmi piuttosto sofisticati operanti quasi esclusivamente per mezzo di menu (in inglese questi programmi si chiamano, appunto, "menu-driven", ossia guidati da menu). In questo genere di programmi la tastiera viene usata soltanto quando si devono immettere dati particolari, come il nome o l'indirizzo di una persona etc.

Programmi speciali

Programmi di questo tipo, però, devono essere compilati appositamente per uso con la penna ottica al posto della tastiera, ma a questo si può spesso ovviare scrivendo opportuni moduli di programma, da aggiungere a programmi normali, grazie ai quali, sfruttando la posizione rilevata dal chip del video, si risale alla opzione scelta. Sfortunatamente, solo poche case produttrici di software forniscono i loro programmi in versioni funzionanti con penna ottica.
L'uso della penna ottica non si limita solamente alla selezione di voci indicate sullo schermo, ma serve anche per la creazione di immagini al computer. Quasi tutti i computer, tra i quali molti home e personal, che prevedono l'uso di una penna ottica, sono dotati di software appositamente studiato per la generazione di immagini.
L'utente si trova di fronte ad uno schermo "pulito" sul quale sbizzarrirsi a disegnare ed, inoltre, in una zona dello schermo (di solito in



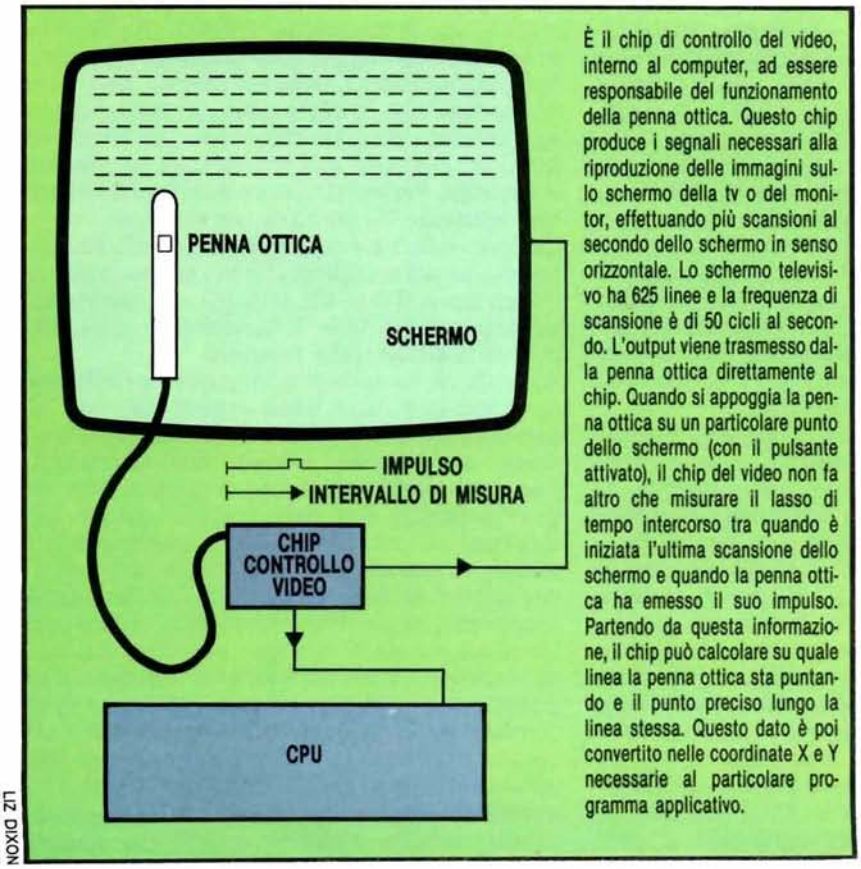
Fotocellula
Questo è un semiconduttore: in altre parole non è che un transistor o un diodo... scoperchiato. La luce, che viene captata dalla giunzione del transistor, controlla il flusso della corrente elettrica che lo attraversa

Cavo
Questo cavo è inserito direttamente in una presa del computer, che permette alla penna di collegarsi al chip del video

Circuiti di amplificazione
Questi circuiti servono per rilevare e amplificare la corrente che passa attraverso la fotocellula e per emettere un segnale adeguato al chip del controllo video nel computer. A volte questi circuiti sono in un contenitore a parte

Pulsante
La maggioranza delle penne ottiche ha un pulsante per l'attivazione, che, di solito, viene azionato con la pressione di un dito, ma in qualche caso l'accensione avviene esercitando con la penna una leggera pressione sullo schermo. Il pulsante è necessario per evitare che la penna reagisca alla luce ambientale quando non è in uso

COME FUNZIONA



È il chip di controllo del video, interno al computer, ad essere responsabile del funzionamento della penna ottica. Questo chip produce i segnali necessari alla riproduzione delle immagini sullo schermo della tv o del monitor, effettuando più scansioni al secondo dello schermo in senso orizzontale. Lo schermo televisivo ha 625 linee e la frequenza di scansione è di 50 cicli al secondo. L'output viene trasmesso dalla penna ottica direttamente al chip. Quando si appoggia la penna ottica su un particolare punto dello schermo (con il pulsante attivato), il chip del video non fa altro che misurare il lasso di tempo intercorso tra quando è iniziata l'ultima scansione dello schermo e quando la penna ottica ha emesso il suo impulso. Partendo da questa informazione, il chip può calcolare su quale linea la penna ottica sta puntando e il punto preciso lungo la linea stessa. Questo dato è poi convertito nelle coordinate X e Y necessarie al particolare programma applicativo.

LIZ DIXON

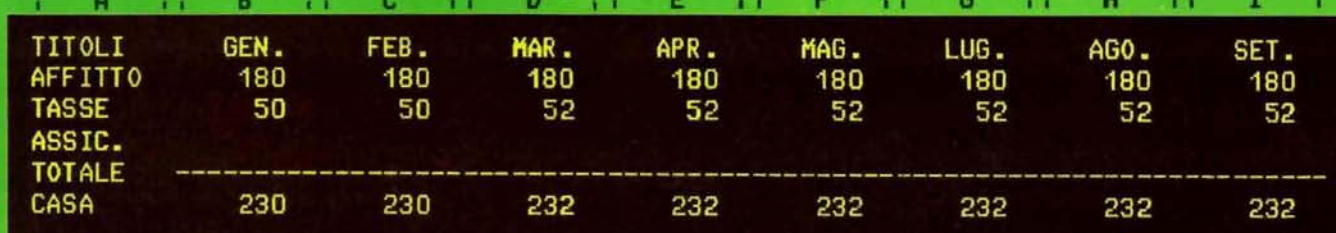
basso) vengono fornite indicazioni su speciali funzioni, disponibili per coadiuvare nel processo creativo. Una di queste è una specie di tavolozza di colori, che può essere usata proprio come fanno i pittori: si appoggia la penna ottica sopra il colore desiderato e (fino alla scelta di un nuovo colore), la penna traccerà sul disegno prodotto sullo schermo linee di tale colore. Inoltre, viene anche data la possibilità di selezionare (sempre dal fondo dello schermo) tra più spessori di linea e perfino tra più forme predefinite (cerchi, quadrati, ecc.). In poche parole, si può fare tutto quello che abbiamo già spiegato a pag. 26, ma certamente molto più velocemente che non con la tastiera.

Ultimamente, il mercato dei videogiochi si è arricchito con quelli che fanno uso della penna ottica. Siccome è molto più facile maneggiare una penna ottica, i giochi devono essere più complicati e frenetici di quelli che impiegano tasti o levette. La penna ottica, comunque, è molto indicata nei giochi tradizionali, come gli scacchi: basta mettere la penna nel quadrato dove si vuole collocare, per esempio, l'alfiere e il computer pensa a tutto il resto.

Tuttavia, le persone che più adoperano la penna ottica sono i tecnici disegnatori e gli ingegneri. I sistemi CAD (sigla che sta per Computer Aided Design, sistemi di disegno computerizzato) sono spesso citati nella pubblicità di nuove automobili. In realtà, questi sistemi computerizzati sono molto simili ad altri, ma con possibilità grafiche nettamente superiori. Quando il progettista adopera il CAD per il progetto di un nuovo apparecchio elettronico, egli ha a disposizione, sullo schermo, una selezione di componenti di cui può aver bisogno, pronti ad essere "prelevati" e piazzati sul punto più appropriato dello schermo.

La penna ottica è forse il migliore esempio di un accessorio per computer che è divertente da usare e allo stesso tempo ha molte applicazioni pratiche.

Gli "spreadsheet"



	A	B	C	D	E	F	G	H	I
1									
2	TITOLI	GEN.	FEB.	MAR.	APR.	MAG.	LUG.	AGO.	SET.
3	AFFITTO	180	180	180	180	180	180	180	180
4	TASSE	50	50	52	52	52	52	52	52
5	ASSIC.								
6	TOTALE								
7	CASA	230	230	232	232	232	232	232	232
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									
25									
26									
27									

I vantaggi offerti dalle lavagne elettroniche

Campi di gioco

Uno spreadsheet (lavagna elettronica) è suddiviso in righe e colonne. L'intersezione di una riga con una colonna si chiama "campo" o "casella" e si individua in base a delle coordinate (A1, B3, ecc.), come accade nel comune gioco della battaglia navale. Ogni campo può avere un titolo ("gennaio" per esempio), un numero (per esempio 149,89) oppure una formula. Il campo N2, per esempio contiene la formula "SUM(A2:M2)" per sommare tutte le cifre della riga 2 (da gennaio a dicembre). Il risultato di questo calcolo appare sul display come 388,4 e potrebbe rappresentare una somma espressa in milioni. Da notare che lo spreadsheet è diviso in due "finestre" di modo che la parte che va dal marzo a dicembre non è visibile.

Il cursore

Il cursore (il rettangolino illuminato che serve ad indicare dove verrà scritto il prossimo carattere) può essere spostato da un campo all'altro usando speciali comandi che, per comodità, vengono riassunti sull'ultima riga dello schermo. Una volta posizionato il cursore, si possono immettere o modificare i valori contenuti nel particolare campo.

È stato stimato che i dirigenti d'azienda trascorrono il 30 per cento del loro tempo preparando budget, un'attività piena di incognite e situazioni incerte. Prima dell'avvento del computer si usava prendere un grande foglio di carta e suddividerlo in una dozzina di colonne, ciascuna intitolata con i mesi dell'anno. Sul lato sinistro del foglio, riga per riga, si elencavano tutte le varie voci inerenti alla gestione dell'azienda e, col passare dei mesi, si riportavano le cifre incolonnandole sotto il mese in questione. Con questo sistema, ad ogni fine mese, si sommavano le cifre di tutta una colonna per controllare il totale mensile delle spese. Alla fine dell'anno si totalizzavano, voce per voce, tutte le colonne. I problemi sorgevano quando, in sede di bilancio di previsione, si commettevano errori di valutazione (specie se per difetto).

Oggi, avvalendosi di sistemi computerizzati chiamati "Spreadsheet" (Lavagne elettroniche) c'è la possibilità di ricalcolare un'intera pagina di cifre tutte le volte che si debba modificare uno qualsiasi dei singoli elementi. Se, per esempio, il costo per il trasporto della merce nel mese di gennaio è cambiato, ciò influisce su tutta la somma delle spese di gennaio, come pure sul totale della riga "Costi di trasporto" per tutto l'anno. Manualmente, ciò implica tutta una lunga serie di calcoli, che il computer può, invece, eseguire in frazioni di secondo, ripresentando immediatamente la nuova situazione. Non c'è da meravigliarsi se il software più venduto nel mondo è proprio di questo tipo.

Come succede con quasi tutti i tipi di software per applicazioni commerciali, anche i programmi per gli spreadsheet sono suddivisi in più moduli, ovvero, non tutto il programma risiede nella memoria allo stesso tempo, ma i vari moduli vengono letti dal disco soltanto quando sono necessari. Il sistema è simile a quello spiegato per le subroutine (vedere a pag. 77), ma in questo caso il sistema operativo trasferisce un singolo modulo dal dischetto ad una zona di memoria precedentemente usata da un altro modulo non più necessario. Questo metodo di "sezionamento" consente di usare programmi molto lunghi e complessi anche quando la memoria RAM del computer non è sufficiente per contenere il programma per intero. L'unico svantaggio è costituito dal tempo impiegato a fare questi trasferimenti.

Esistono molte versioni di spreadsheet (facilmente riconoscibili perché il loro nome di solito finisce in "calc"), adatti sia ai personal che agli home computer. Uno dei più noti è il "Visicalc", commercializzato per l'Apple II nel 1979. Sulla scia del successo riscosso da questo programma, è comparsa sul mercato un'infinità di sistemi analoghi per tutti i tipi di computer, anche se talvolta la qualità lascia a desiderare.

Per essere veramente utile, uno spreadsheet deve avere due caratteristiche: dimensioni adeguate al "foglio" (non necessariamente legate a quelle di ciò che appare sullo schermo perché, come vedremo, il video è solamente una "finestra" di tutto l'insieme) ed una gamma sufficiente di comandi di controllo. Ciò significa che esistono dei precisi limiti circa il tipo di macchina in grado di sfruttare al massimo le capacità di questi programmi. Indicativamente, i requisiti minimi per poter adoperare con profitto uno spreadsheet per applicazioni commerciali sono: 32 Kbyte di memoria RAM ed uno schermo da 80 caratteri per riga. Per applicazioni domestiche basta uno schermo da 40 caratteri per riga.

Chi possiede un home computer, che funziona con nastri a cassetta (come per esempio lo Spectrum o il Vic 20), troverà che molto del software disponibile è certamente utile, anche se limitato nelle funzioni.

Essendo lo spreadsheet in grado di rispondere a domande tipo "cosa accadrebbe se..." il suo impiego è veramente utile nella formulazione di semplici modelli computerizzati, come abbiamo accennato a pagina 101. In quell'esempio abbiamo parlato del lavoro dell'analista e abbiamo fatta presente la necessità di pianificare con molta cura i metodi per gestire le informazioni: questa necessità si avverte, in particolare, quando si usa uno spreadsheet. Abbiamo già visto che i database elettronici sono costituiti da una massa di informazioni disorganizzate, che vengono riordinate o selezionate in base a particolari esigenze, mentre i word processor offrono all'utente la possibilità di elaborare testi con grande facilità: il funzionamento degli spreadsheet, invece, richiede un'attenta analisi preliminare da parte dell'utente.

Se, infatti, si intende usare uno spreadsheet per controllare l'andamento delle spese personali, è utile riunire insieme tutte le spese relative alla casa (affitto, rate del mutuo, tasse comunali, ecc.) per poi usare il risultato della somma in una tabella più grande dello stesso programma. In tal caso bisogna aver cura di eseguire la somma, prima di riportare la cifra nella tabella più grande.

Uno spreadsheet mette a disposizione una certa quantità di caselle (il numero di esse varia da versione a versione). Ciascuna casella viene individuata dalle sue coordinate: orizzontalmente si usano combinazioni di lettere (A-Z, AA-ZZ o anche BA-BM) per selezionare una delle colonne (normalmente 64 in tutto), mentre verticalmente si impiegano numeri, generalmente compresi tra 0 e 256. Ciascuna casella può contenere una parola, che funge da "etichetta" (per esempio "vendite" o "profitti") oppure un valore immesso da tastiera o ottenuto da un calcolo oppure, infine, la formula per un calcolo, (come, ad esempio: B4 + B6 * B5). Siccome spesso le formule sono più lunghe dello spazio assegnato alla casella, in alcuni spreadsheet ad esse è riservata una speciale zona dello schermo. Quando si inizia il lavoro con un foglio nuovo, la dimensione delle caselle è già prefissata in un valore standard (detto anche di "default"), normalmente 8 o 9 caratteri, ma c'è la possibilità di variare, entro certi limiti, la dimensione di ciascuna casella, a seconda delle necessità. La colonna più a sinistra serve per l'immissione delle varie "voci" o titoli ed è più larga delle altre. Non è necessario decidere a priori quanti caratteri assegnare a ciascuna colonna, perché è sempre possibile intervenire successivamente (anche quando le caselle contengono dei dati). Se la lunghezza viene ridotta al di sotto di quella del dato contenuto, la parte che non appare sul video non viene cancellata, ma solamente "mascherata". L'ultimo importante elemento è il menu dei comandi che appare (in alto e/o in basso sullo schermo) quando viene impartito un particolare comando, che può essere, per esempio, il carattere "/". I comandi a disposizione servono per costruire o modificare lo schema generale del sistema, e non per inserire particolari dati, anche se la visualizzazione di questi ultimi può dipendere da alcuni di essi. Quasi tutte le versioni danno la possibilità di eseguire una vasta gamma di operazioni sui dati introdotti nel foglio. Si può, per esempio, cancellare, trasferire o copiare il contenuto di una colonna o riga in un'altra. Si può suddividere lo schermo in più "finestre" per avere contemporaneamente sul display due o più parti dello spreadsheet, normalmente troppo distanti fra loro.

Lo spostamento da una casella all'altra si ottiene con i tasti per il movimento del cursore, ma è anche possibile saltare ad una specifica casella, specificandone le coordinate. L'elaborazione, la conservazione, la cancellazione e la protezione sono operazioni comandate dai tasti di comando. È bene ribadire l'importanza di "salvare", ad intervalli regolari, il proprio lavoro su dischetto, per evitare di perderlo a seguito, per esempio, di interruzioni elettriche. La preparazione dello schema è un lavoro che richiede molto tempo (spesso molto più di quello che si impiega per inserire dati) ed è buona regola archiviare lo schema su dischetto, prima di iniziarne l'uso (se dovessimo commettere qualche irrimediabile errore, almeno questa parte di lavoro sarebbe salva).

I risultati possono essere inviati alla stampante per mezzo di opportuni comandi, avendo cura di specificare, per mezzo di parametri, quale parte del foglio desideriamo stampare. Se lo schermo è una finestra, o una porzione dell'intero foglio, ciò è vero anche per la pagina di stampa. Quando si deve stampare una finestra di larghezza maggiore a quella del foglio della stampante, si possono sempre eseguire due successive stampe e riunire il tutto in un momento successivo.

Così come è possibile suddividere lo schermo in più finestre, si può fare altrettanto in fase di stampa: ciò è particolarmente utile per evitare errori durante l'immissione dei dati, quando occorre rivedere valori già immessi. In quasi tutti i sistemi esiste la possibilità di tener fissa sullo schermo la prima linea, che riporta titoli e nomi delle colonne.

Cosa accadrebbe se...?

Quando viene modificato il contenuto di un campo, lo spreadsheet automaticamente ricalcola tutti gli altri campi collegati ad esso mediante formule matematiche definite dall'utente. La facilità di questa operazione induce l'utente a mettere alla prova i budget, per vedere l'effetto che avrebbe sull'utile se, per esempio, certi elementi dovessero cambiare. Facciamo l'esempio di un venditore di frutta:



— se il prezzo della benzina aumentasse di X%...

— ...allora le spese mensili di trasporto salirebbero di Y%...

— ...e questo, a sua volta, inciderebbe sul prezzo all'ingrosso delle mele...

— ...ed infine sul prezzo al minuto

Fino ad ora abbiamo considerato lo spreadsheet come una tabella alla quale si può accedere solamente immettendo o richiamando un dato dopo l'altro, lungo una riga o una colonna, ma sulla disposizione dei vari elementi è lasciata ampia libertà, purché, in fase di analisi, si presti particolare attenzione ad una corretta formulazione dei rapporti esistenti tra le varie caselle.

Gli spreadsheet per uso commerciale, come il Visicalc, il Supercalc e il Masterplan, offrono all'utente la possibilità di utilizzare, attraverso passaggi automatici, le informazioni immesse ed elaborate nello spreadsheet da parte di programmi di word processing o di gestione di archivi (database). Esistono poi dei programmi complementari, che permettono di visualizzare i risultati in forma grafica: grafici "a torta" o "a barre".

Abbiamo già menzionato uno degli usi più comuni di uno spreadsheet: l'analisi delle spese personali; ma un altro impiego, sempre nell'ambito domestico, è l'ottimizzazione dell'impianto del riscaldamento, dove sono molti i fattori variabili da tenere in considerazione: tipo di combustibile, numero e tipo di radiatori, capacità della caldaia, ecc. In termini molto generali, qualsiasi processo decisionale trae beneficio dall'uso di uno spreadsheet, anche perché l'utente è costretto ad analizzare tutte le possibilità esistenti.

Forse la cosa che più sbalordisce quando si vede in azione un programma di questo genere, è la velocità con cui lavora. Questa è in genere dovuta al fatto che i programmi vengono scritti in linguaggio macchina. Forse vale la pena di considerare alcuni dei problemi che andrebbero affrontati se si cercasse di scrivere un programma di questo tipo in BASIC, se non altro per dare un'idea di quanto è complesso il problema.

Tanto per cominciare, ciascuna casella deve essere definita in tre modi diversi: deve poter conservare dati sia numerici (per le operazioni matematiche) che alfanumerici ("le voci" o titoli). Deve inoltre poter conservare una formula di calcolo, ossia un gruppo di istruzioni (come, per esempio, tasse annuali/12 = tasse mensili).

In più in ciascuna cella deve essere possibile modificare la lunghezza del dato visualizzato senza peraltro perdere la parte meno significativa del dato. Ciò comporta una duplicazione di tutte le caselle: una serie viene impiegata per conservare i valori reali, l'altra per la parte di essi da visualizzare.

Appare subito evidente che la sola elaborazione dei dati diventa un lavoro molto complicato. In più occorre ricordarsi che gli spreadsheet più sofisticati possono contenere anche più di 16.000 caselle! Le tecniche in uso per scrivere software di questo tipo sono molto simili a quelle usate nella progettazione di linguaggi come il BASIC e il FORTH e vengono usate anche nei programmi per la gestione dei "database".

Tutto questo giustifica, almeno in parte, l'alto costo del software scritto appositamente per uso commerciale. Un programma tipo il Visicalc o il Supercalc può costare anche centinaia di migliaia di lire ma si deve tener conto del risparmio che si ottiene a lungo andare, grazie al suo impiego. Un'applicazione piuttosto semplice, come la preparazione di budget, che abbiamo portato come esempio, consente di risparmiare fino al 15% o 20% sulle spese generali annue.

A queste condizioni, il costo del software diviene insignificante, senza contare l'enorme risparmio di tempo e la preziosa possibilità di analizzare più a fondo le situazioni incerte.



Finestre sul mondo

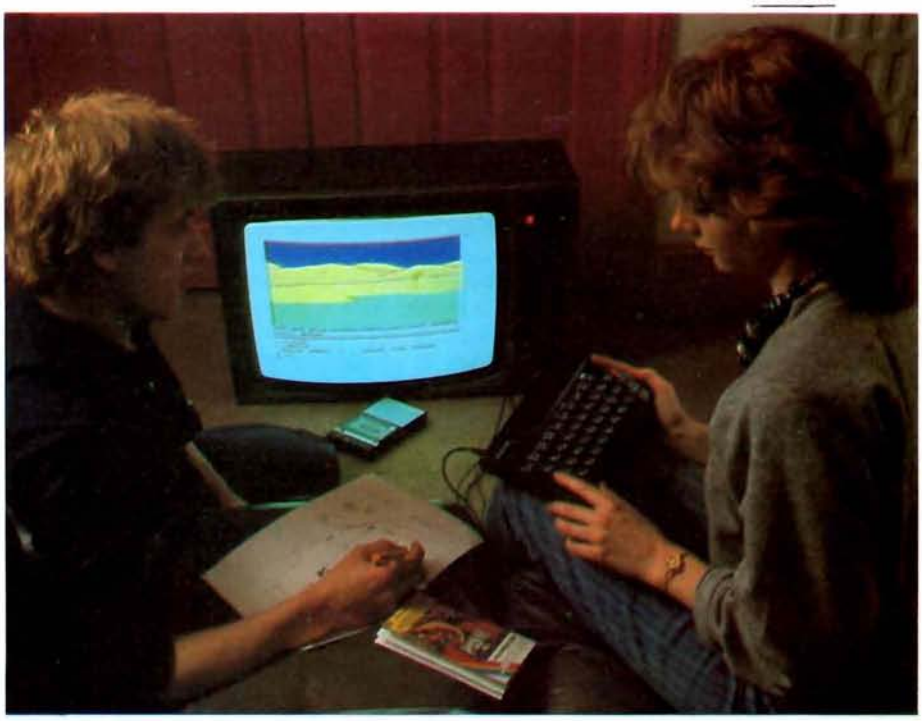
Così come possiamo muovere il cursore da sinistra a destra, in alto e in basso sullo schermo, usando i vari tasti di comando, similmente possiamo visualizzare una sola sezione dello spreadsheet. Questo ci dà modo di lavorare su "fogli" di superficie molto superiore a quella dello schermo stesso.

Alcuni computer portatili con schermo di larghezza limitata, come l'Epson HX-20 e l'Osborne-1 usano lo stesso sistema di visualizzazione in tutti i programmi.

I giochi d'avventura

Negli "Adventure games" il giocatore diventa protagonista

IAN DOBBIE



In genere, la parola "avventura" richiama alla mente un libro, un film o, in alcuni casi, un'esperienza personale. Ma esiste tutta una schiera di persone per le quali questo termine significa un nuovo modo di giocare col computer: gli "Adventure games", o giochi d'avventura.

A differenza di ciò che accade leggendo un libro, in un Adventure game il giocatore si trova a "vivere" da protagonista i pericoli, i misteri e le situazioni che fanno parte della storia: non si limita a leggere le imprese del protagonista del racconto.

Ma c'è di più: quando si legge un libro non si può modificare il corso degli eventi narrati: si può leggerlo un'infinità di volte e la storia rimane sempre la stessa. Al contrario, negli Adventure il giocatore può influire con le sue azioni sullo svolgimento della trama. La successione degli avvenimenti è diversa di volta in volta e, se il finale sia felice o meno, dipende dalla bravura di chi gioca. Comunque, il fattore predominante è la partecipazione attiva ai rischi dell'avventura, sebbene le emozioni siano vissute dai giocatori che se ne restano comodamente seduti in casa.

I giochi di questo genere sono sempre ambientati in luoghi piuttosto particolari: in strani mondi sotterranei, in "luna park", in città fantasma, su un altro pianeta oppure in paesi mitici; insomma, in qualsiasi posto in qualsiasi epoca, passata, presente o futura. Di solito l'avventura si svolge secondo un tema con un ben preciso obiettivo finale: si deve trovare, per esempio, il modo di fuggire da un pianeta nemico oppure scovare ed annientare un mago cattivo o, ancora, salvare una principessa, trovare un tesoro nascosto oppure, addirittura, risolvere un giallo e arrestare il colpevole.

Tutto ciò è già di per sé molto interessante, ma la parte più entusiasmante arriva quando si è obbligati a risolvere dei quesiti "rompicapo", che costituiscono una parte integrante dell'avventura. I quesiti si presentano sotto le più svariate forme: quella più comune implica la soluzione di un enigma prima di poter proseguire con il gioco, come nel caso in cui ci si trova a dover attraversare un ponte pericolante. A volte, invece, un enigma è stato introdotto nel gioco per trarre in inganno il giocatore. Può succedere di attraversare un canyon, superando mille pericoli, per affrontare lo strano individuo che si trova dall'altra parte e scoprire, spesso troppo tardi, che il losco figuro non è altro che la nostra immagine riflessa in un enorme specchio. La risposta giusta ad un enigma può essere utile, ma non decisiva: per esempio, quando si trova il passaggio segreto per sfuggire al mago cattivo. Altre volte, al contrario, diventa una questione di vita o di morte, come quando ci si viene a trovare sull'orlo di un precipizio senza possibilità di salire o di scendere e per di più si è senza cibo.

Quasi sempre gli enigmi si risolvono con un poco di astuzia e non sono necessarie particolari conoscenze o capacità intellettive eccezionali. Il giocatore deve sempre prestare la massima attenzione al gioco, perché tutte le soluzioni sono contenute nel contesto, oppure sono deducibili da esso. Difatti, nelle avventure ben scritte, non ci sono elementi messi lì per caso ed occorre ricordare che qualsiasi elemento del gioco ha una sua importanza, anche se, a prima vista, un oggetto, un messaggio o un personaggio sembrano estranei alla storia. Talvolta, però, qualche elemento è inserito volutamente per sviare il giocatore e rendere il gioco più entusiasmante. Che significato può avere una collezione di bottiglie di rhum, ridotte in frantumi? Oppure una strana voce che dice «Plush»? A cosa può servire un mucchio di fango maleodorante oppure un tappeto inchiodato al pavimento anche se sotto non c'è niente? Questi sono esempi tratti da giochi di avventure, nei quali simili particolari assumono grande importanza nel prosieguo del gioco.

Chi è il protagonista?

I giochi d'avventura per computer sono in commercio da svariati anni, ma l'avvento degli home computer ha generato una vera valanga di nuovi giochi. Alcuni più elementari, altri più sofisticati ed accattivanti, sono principalmente dei "solitari", ossia c'è un unico giocatore, che diviene il protagonista delle storie più incredibili e fantasiose

Un elemento comune a molte storie è un labirinto, dove tutte le stanze appaiono identiche: in questi casi l'unico modo per riconoscerle è di fare come Hänsel e Gretel, cioè lasciare degli indizi in tutte le stanze, magari lasciando cadere un oggetto diverso in ogni stanza. Ma, attenzione, perché questo trucco è ormai talmente noto, che in alcune storie è stato inserito, per esempio, un ladro che insegue il giocatore mescolando gli indizi da lui disseminati!

In alcune avventure, è importante l'ordine col quale si risolvono gli enigmi, in altre no, ed esiste un unico modo per poter raggiungere un finale trionfante.

Ci sono, poi, delle avventure che richiedono ore, o addirittura settimane, prima che siano svelati tutti i "segreti": è ovvio come in questi casi vi sia la possibilità, per il giocatore, di poter fermare il gioco in qualsiasi momento, (registrando la situazione su disco o nastro), e di poterlo successivamente riprendere dallo stesso punto. Ciò è particolarmente utile nei momenti cruciali della storia, quando, per esempio, il giocatore sta per affrontare un esercito di orchi feroci, armato di una sola lampada e di una bottiglia d'acqua. Avendo poche possibilità di farcela, gli conviene fermare il gioco, memorizzarlo, e riprenderlo quando avrà avuto la possibilità di valutare attentamente le varie alternative. A questo punto, anche se gli orchi decidessero di "mangiarsi" il giocatore per colazione, si avrà sempre la possibilità di ripetere di nuovo l'azione, magari seguendo un percorso meno pericoloso, il che è sempre meglio che ripartire da zero.

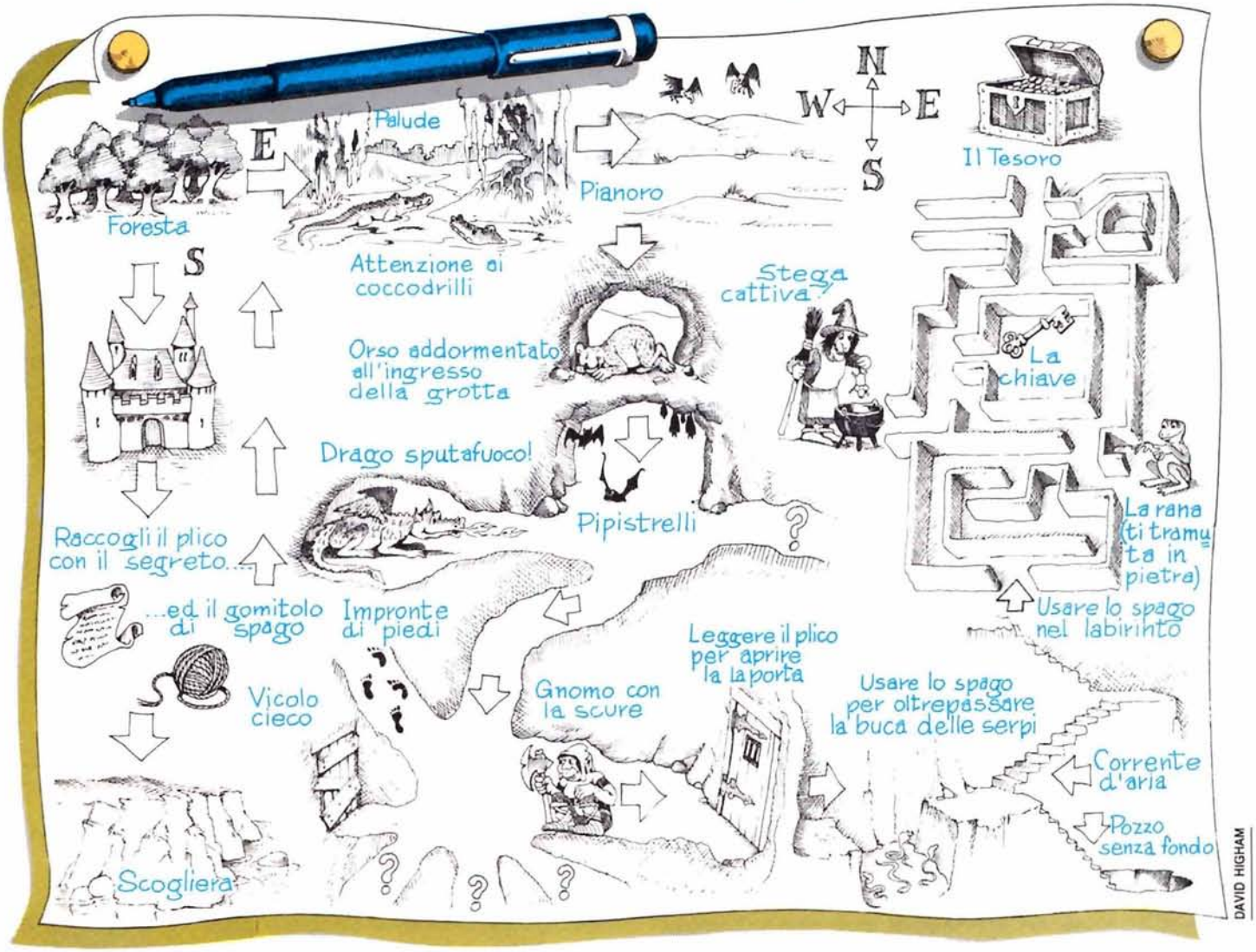
Anche nei casi in cui l'"avventuriero" viene ucciso, non vuol dire che il gioco sia necessariamente finito, perché a volte egli può tornare in vita, magari a costo di una penalizzazione e perdendo dei punti. In ogni caso, se vi è la possibilità di resuscitare, egli si troverà sempre in luoghi poco ameni: spesso in un cimitero, ma, di certo, lontano dalla civiltà.

Abbandoniamo adesso queste fantasie, e vediamo come il giocatore prenda parte e comunichi con il programma. A volte il giocatore viene interrogato direttamente, altre volte egli è rappresentato da una figura: un "omino" manovrabile tramite i tasti del computer. Il computer interpreta i comandi e riporta le risposte sul video, assieme a parte della narrazione.

Se il computer non dispone di effetti grafici, l'avventura si svolge esclusivamente attraverso il testo che appare sul video, altrimenti vengono impiegate immagini (talvolta molto

La mappa del tesoro

Nei giochi d'avventura, il giocatore si addentra nei luoghi più impervi e misteriosi, in genere alla ricerca di un tesoro da carpire. Durante il percorso, egli deve affrontare tutta una serie di difficoltà e risolvere numerosi enigmi, dai quali dipende il successo della missione. Talvolta il gioco si dipana in maniera così complessa, che si deve ricorrere ad una mappa, per non perdersi...



suggestive), che sono a completamento del testo scritto. Per la maggior parte, le immagini sono statiche, a volte fatte di poche linee, a volte più dettagliate.
Un elemento abbastanza comune nei giochi che impiegano la grafica è una mappa stilizzata oppure il disegno di un ambiente o un edificio. I personaggi e gli oggetti sono rappresentati da simboli o pupazzetti in miniatura. In questi giochi normalmente il giocatore ha pochi comandi a disposizione, spesso un solo tasto per spostare e controllare il personaggio.
Nei giochi senza grafica, il testo fornisce sempre un'indicazione chiara su tre elementi importanti: dove si trova il giocatore; cosa vede e in quale direzione può andare. Un esempio potrebbe essere il seguente: «Sei in una foresta scura. Sopra di te il cielo è oscurato dalle foglie degli alberi. C'è un sentiero che va da est a ovest. Di fronte, in direzione nord, c'è un precipizio. C'è una spada per terra, e intorno ad essa è attorcigliata una serpe». A questo punto il giocatore ha una descrizione dell'ambiente, un'idea sulla direzione da prendere e sugli oggetti a portata di mano. I comandi a disposizione del giocatore di solito sono composti di due parole (un verbo seguito da un sostantivo), anche se nelle avventure più sofisticate sono previste frasi intere. I verbi più diffusi sono: PRENDI, LASCIA, SPINGI, TIRA, GETTA, ILLUMINA, UCCIDI, MANGIA, BEVI, oppure VAI A NORD, per segnalare la direzione da seguire.
Un verbo molto importante è ESAMINA, in quanto dà al giocatore la possibilità di ottenere maggiori informazioni: «ESAMINA LA SERPE», per esempio, può fornire al giocatore informazioni del tipo: «È una serpe innocua» oppure «La serpe ti ha visto e ti vuole colpire». Alcuni verbi non necessitano di un sostantivo.
La parola INVENTARIO è spesso usata per fornire al giocatore un elenco degli oggetti che ha con sé. A volte alcuni oggetti sono contenuti in altri (l'acqua dentro la bottiglia o l'ascia dentro il sacco) mentre altri sono indosso (un anello o un mantello, per esempio). La parola PUNTEGGIO serve al giocatore per conoscere quali progressi stia facendo verso l'obiettivo. Quando il giocatore chiede aiuto (HELP) di solito il computer interviene aiutandolo a superare una difficoltà, e, allo stesso tempo, incoraggiandolo a continuare il gioco. A volte accade che, a seguito di un particolare comando, il computer comunichi al giocatore: «Non puoi fare questo per adesso...». Implicitamente, la particolare combinazione "verbo-sostantivo" del comando darà un risultato utile, ma non in questo luogo o situazione. Parte del gioco consiste spesso proprio nella scoperta di quali verbi e nomi abbiano rilevanza. Le combinazioni di parole non riconosciute solitamente generano un responso del tipo: «Non capisco».
I giochi di avventura sono corredati da un manuale e spesso anche da elementi (buste, indizi e varie "tracce") da utilizzare durante il gioco.



IAN McKINNELL

CACCIA AL TESORO

"L'Hobbit" è il nome di un gioco progettato per la grafica dello Spectrum Sinclair. Il nome deriva dall'omonimo libro di J.R.R. Tolkien, che viene dato in omaggio con la cassetta. Il giocatore si immedesima in Bilbo, che si addentra nella Terra di Mezzo, incontrando molti personaggi del libro

In certi casi, i programmi sono troppo complicati e lunghi per essere contenuti nella memoria del computer ed è per questo che ne esiste soltanto una versione su dischetto. Il programma principale viene caricato in memoria all'inizio del gioco, mentre brani del testo o immagini vengono estratte dal disco al momento opportuno. Tuttavia, grazie a nuove tecniche di registrazione, che permettono la conservazione di quantità elevate di dati in una memoria di dimensioni ridotte, si stanno diffondendo molte "avventure" anche su nastri a cassetta.
Ci sono avventure adatte ad ogni tipo di microcomputer e tutte offrono la possibilità di trascorrere un po' di tempo divertendosi con la fantasia.



IAN DOBBIE

per concessione della CENTROSOFT LTD.

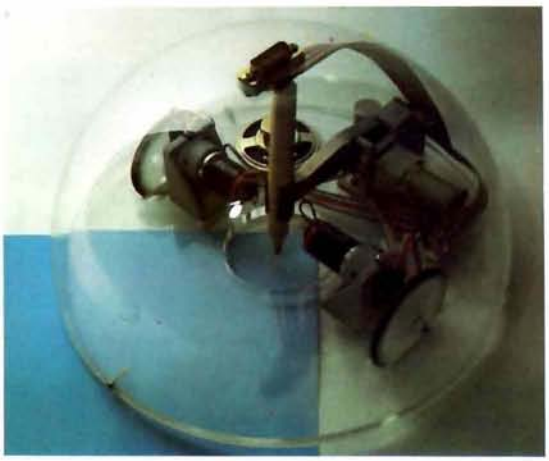
Un giallo... sempre nuovo
Una variazione sul tema delle avventure è il gioco "Deadline", nel quale il giocatore assume il ruolo di un investigatore alla ricerca di un assassino. Il gioco può durare anche molto a lungo (un paio di settimane o più!), durante le quali occorre riflettere, interrogare, ed agire in base ai vari indizi offerti

I "micromondi"

L'uso del computer a scopo didattico

La Tartaruga

La tartaruga (turtle) è stata ideata come mezzo per indurre il bambino a ragionare e, in particolare, per imparare la geometria e i rapporti interspaziali. Quando il bambino è insicuro sul modo migliore di comandare la tartaruga, di solito tende ad immedesimarsi nella tartaruga, camminando per la stanza ubbidendo alle istruzioni del LOGO. In questo modo il bambino impara direttamente da esperienze personali



IAN McKINNELL

Quando apparvero i primi microcomputer, nel 1977, gli educatori più lungimiranti non tardarono a capire quanto questo nuovo apparecchio poteva essere utile, come mezzo didattico, nella scuola. Sono molte, oggi, le scuole che hanno almeno un microcomputer e in alcune di esse l'informatica fa parte delle materie d'insegnamento. Nonostante ciò, il computer non ha influito molto sui metodi tradizionali di insegnamento.

Questo fatto è evidenziato in modo particolare dai programmi didattici per microcomputer attualmente disponibili, che dimostrano per la maggior parte un'incredibile mancanza di fantasia: quasi tutti non sono altro che "libri di testo elettronici", in quanto il computer mostra allo studente una serie di pagine scritte (proprio come quelle di un libro), ponendo poi una serie di domande per vedere se l'allievo ha compreso la materia. Automaticamente, viene assegnato un punteggio in base alle risposte: il tutto senza l'intervento dell'insegnante.

I programmi di questo genere, sebbene vengano spesso corredati da grafici colorati o perfino animati, si possono considerare soltanto un'automazione del processo già esistente, e non un'innovazione resa possibile dalle reali capacità del microcomputer.

Il discorso cambia per quanto riguarda il LOGO. Il LOGO è il frutto di studi fatti dal Prof. Seymour Papert (Istituto di Tecnologia del Massachusetts) e viene definito come "una filosofia dell'educazione e una famiglia di linguaggi per la programmazione del computer, progettata per aiutare ad implementare tale filosofia".

Molte persone, erroneamente, considerano il LOGO esclusivamente come un linguaggio per la programmazione, e, paragonandone la semplicità dei comandi e i vantaggi offerti con quelli del BASIC, arrivano alla conclusione che il LOGO è più adatto ai principianti del BASIC. Ma questo ragionamento non rende giustizia allo scopo principale del LOGO: Papert non ha ideato il suo sistema come un metodo per insegnare ai bambini l'arte della programmazione, ma lo ha concepito, invece, come "un ambiente in cui i bambini possono imparare diverse materie". In altre parole: un sistema per imparare ad imparare.

Molti aspetti di tale filosofia derivano dagli studi dell'eminente filosofo svizzero Jean Piaget. Egli ritiene che i bambini, purché allevati nell'ambiente adatto, siano in grado di imparare da soli qualsiasi materia, con la stessa facilità con la quale imparano a camminare o a parlare. La tesi di Piaget, comunque, è completamente teorica e Papert s'impegnò per creare l'ambiente pratico adatto per questo criterio di apprendimento.

Che i metodi di insegnamento tradizionale non facilitino l'apprendimento delle materie scolastiche appare evidente dal fatto che moltissimi adulti hanno "paura" di imparare qualcosa di nuovo o di apprendere nuove tecniche professionali. Il sintomo più comune di questa specie di fobia, asserisce Papert in un suo libro, è la paura della matematica.

Il motivo principale di questa diffidenza è dovuto al fatto che il metodo di insegnamento nella scuola varia poco da una materia all'altra, mentre le applicazioni di ciascuna di esse sono completamente diverse. I bambini imparano come fare le moltiplicazioni nello stesso modo con cui imparano i nomi delle capitali del mondo: a pappagallo! Il processo di apprendimento viene trattato quasi come entità separata da ciò che il bambino sta imparando.

Papert considera l'apprendimento di qualsiasi cosa nuova (sia essa volare, cucinare o una lingua straniera) come un hobby, attribuendo questo suo modo di pensare alla propria infanzia, ed, in particolare, a quando, da bambino, egli scoprì il funzionamento degli ingranaggi di un cambio. Da allora, egli ha sempre mantenuto questo atteggiamento di fronte ad ogni nuovo problema. Anche il grande Albert Einstein ha sempre detto che, dovendo affrontare qualcosa che non riusciva a capire, egli la scomponeva in tanti concetti elementari comprensibili, da quando aveva cinque anni d'età.

Il LOGO è basato su questi ragionamenti, come si può notare dall'esempio di un'applicazione pratica del LOGO, che riportiamo a lato. Il primo contatto con il LOGO avviene, di solito, per mezzo della "Tartaruga" (Tur-

tle), un apparecchio progettato per essere "un mezzo con cui ragionare" (seguendo l'esempio degli ingranaggi adoperato da Papert). La tartaruga, per i bambini più piccoli, non è altro che un piccolo robot progettato appositamente per muoversi sul pavimento di una stanza, controllato da un microcomputer mediante comandi in LOGO. In genere, la tartaruga è provvista di una matita con la quale lascia una "traccia", disegnando forme più o meno geometriche su grandi fogli stesi sul pavimento. Talvolta è anche fornita di un altoparlante e di un rilevatore di collisione.
Il passo successivo è costituito dall'impiego di una tartaruga visualizzata sullo schermo del computer, sul quale si costruiscono immagini geometriche. La tartaruga è un apparecchio ideato per insegnare i concetti basilari dei rapporti interspaziali degli oggetti, ed è possibile persino insegnare concetti molto avanzati della geometria.
Il controllo della tartaruga, comunque, non è che una delle tante applicazioni del LOGO: viene forse più pubblicizzata perché è la più interessante da osservare. Molto più importante è, invece, il concetto di sfruttare idee semplici per costruire ragionamenti più complessi o, viceversa, di scomporre i problemi complicati in tanti piccoli problemi, facilmente risolvibili.
Un significativo esempio su questo sistema di apprendimento viene dato dal dialogo di un gruppo di bambini che stanno imparando a comandare la tartaruga per creare il disegno di un fiore (vedere riquadro a lato).
Il LOGO non è l'unico linguaggio che incorpora delle strutture di questo genere, un altro è il FORTH, ma il LOGO è senz'altro l'unico progettato appositamente per i bambini piccoli. Il LOGO elimina molte formalità e procedure connesse alla programmazione in altri linguaggi. Infatti lo scopo del LOGO è proprio quello di far sì che il bambino non si renda conto che sta, in effetti, programmando un computer, ma si dedichi, invece, alla risoluzione di un particolare problema. Addirittura, in certe applicazioni didattiche, l'allievo non viene nemmeno coinvolto nella fase di programmazione: è l'insegnante che prepara una serie di "strumenti" (tools) molto sofisticati ed inerenti ad una materia scolastica o area di apprendimento specifica. In questo modo, il ragazzo è libero di esplorare la materia usando i mezzi messi a sua disposizione per scoprire tutto da sé. Queste aree di apprendimento sono definite "micromondi": ossia, ambienti circoscritti, nei quali il computer viene usato per simulare qualcosa di esistente nel mondo reale oppure qualche area della scienza.
Forse uno dei migliori esempi di micromondo è offerto dal modello LOGO relativo alle teorie della fisica di Newton. La prima legge di Newton dice che, senza l'influenza di forze esterne un corpo continuerà a muoversi nella stessa direzione ad una velocità costante, mentre i bambini sanno che, in realtà, il corpo rallenta il suo moto. Questo apparente contrasto potrebbe creare un blocco nell'apprendimento: con il LOGO si può creare un micromondo in cui tutto si comporta secondo le leggi di Newton e, ricorrendo a particolari comandi per lo spostamento degli oggetti sullo schermo, il bambino ben presto impara da sé tutte e tre le leggi di Newton.
Vale senz'altro la pena di approfondire la conoscenza del LOGO applicandolo su di un home computer. Anche se il costo di una tartaruga da pavimento è abbastanza elevato, ci sono versioni di LOGO, adatte alla maggior parte degli home computer, che manovrano una tartaruga sullo schermo.

IMPARIAMO A DISEGNARE UN FIORE

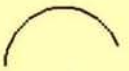
Questo dialogo offre una buona rappresentazione su come il LOGO incoraggi un gruppo di bambini a risolvere dei problemi mai affrontati prima.

```
TO CURVE
REPEAT 80
  FORWARD 1
  RIGHT 1
END
CURVE
```



«Per fare dei petali dobbiamo disegnare una linea curva...»
«Sì, ma la tartaruga si muove sempre in linea retta...»
«Allora facciamola muovere un poco, poi la spostiamo leggermente e la facciamo muovere di nuovo e così via.... dovrebbe venire fuori una linea curva»
«Va bene, la distanza minima è 1 e l'angolo minimo è pure 1. Proviamo a fare questo per 80 volte».

```
CURVE
CURVE
```



«Ecco, proprio come volevamo».
«Due di queste curve fanno un petalo, vero? Proviamo...»

```
TO PETAL
CURVE
RIGHT 90
CURVE
END
PETAL
```



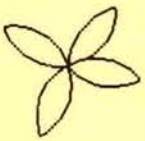
«Ma quello non assomiglia per niente ad un petalo... che è successo?»
«Ha continuato nella stessa direzione di prima. Dovevamo dirgli di andare in un'altra direzione...»
«Di quanto si deve far girare?»
«Proviamo con 90... alle volte funziona».
«Allora, facciamo un'altra parola PETAL per risparmiare tempo».

```
TO PETAL
CURVE
RIGHT 100
CURVE
END
PETAL
```



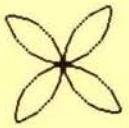
«Adesso va meglio, ma 90 non basta... cosa facciamo ora?»
«Cerchiamo di fare un calcolo preciso invece di indovinare».
«OK, ti ricordi che abbiamo imparato che, se la tartaruga gira tutto intorno ad un oggetto, deve fare un angolo completo di 360?».
«Allora sappiamo che gira di 80 nella prima curva, perciò deve fare altrettanto sulla seconda per ritornare indietro... fanno 160».
«Questo ci lascia una curva di 200 per fare la punta del petalo».
«No, non va bene perché per ritornare alla posizione da cui è partita bisogna fare la punta anche all'altro lato»
«Allora, proviamo la metà di 200»

```
PETAL
PETAL
PETAL
PETAL
```



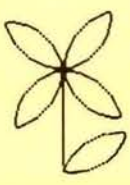
«Fantastico... quattro di questi faranno un fiore!»

```
TO FLOWER
REPEAT 4
  PETAL
  RIGHT 10
END
FLOWER
```



«Così non va mica tanto bene, è storto».
«Abbiamo dimenticato di fare le curve fra i petali».
«Allora, perché non ha disegnato un petalo sopra l'altro?»
«Perché quando ha disegnato un petalo, la tartaruga è rivolta di 100 gradi rispetto alla posizione iniziale».
«Così ogni petalo è inclinato di 100 gradi».
«A noi serve una rotazione di 90 tra petalo e petalo».

```
FLOWER
RIGHT 180
FORWARD 100
RIGHT 180
PETAL
```



«Finalmente! Fai girare la tartaruga in modo da disegnare lo stelo. 100 dovrebbe bastare».
«Facciamo anche una foglia in basso, può essere identica al petalo».
«Va bene, ma ricorda di fare l'angolo giusto questa volta... bisogna far girare la tartaruga completamente intorno a sé stessa».

LIZ DIXON

L'era dei portatili

I computer hanno raggiunto dimensioni minime, ma così piccoli a cosa servono?



IAN DOBBIE

Un giocattolo per dirigenti?

I computer portatili (come l'Epson HX-20) con capacità di memoria di 32 Kbyte, alcuni accessori periferici incorporati ed una vasta disponibilità di software, hanno dato modo ai dirigenti d'azienda di avvalersi di una grande quantità di informazioni, in qualsiasi posto essi si trovino. Ogni microcassetta può contenere fino a 230 Kbyte (ossia fino a circa 40.000 parole): l'equivalente di una banca dati di ragionevoli dimensioni, mentre la possibilità di avere a disposizione un sistema standard di "word processing" oppure una lavagna elettronica permettono al dirigente di lavorare ovunque, perfino mentre viaggia in taxi!

Lo sviluppo dei microcomputer portatili proviene da due direzioni: la grande diffusione delle calcolatrici tascabili, come la PC 1251 della Sharp e la FX700P della Casio, e l'enorme evoluzione nel processo di miniaturizzazione, che ha consentito di produrre computer di dimensioni ridotte quali, ad esempio, il Personal Computer Tandy TRS80 Modello 100 e l'Epson HX-20 (vedere a pag. 169).

Questi passi in avanti sono il frutto dei nuovi processi costruttivi dei chip, che permettono di conservare un numero sempre più grande di informazioni occupando uno spazio sempre minore. Nel 1972, con la prima apparizione dei microprocessori "single chip" (dotati di un unico chip), divenne realizzabile, almeno in teoria, la costruzione di un computer completo dalle dimensioni di un pacchetto di sigarette. In pratica, però, vi sono delle limitazioni, dovute principalmente al display ed alla tastiera, che devono essere di dimensioni tali da risultare comode all'uso.

Le calcolatrici tascabili, invece, sono diventate decisamente più piccole e, addirittura, oggi si trovano anche degli orologi digitali funzionanti anche come calcolatrice. I tasti, però, sono di dimensioni così ridotte che per fare la più semplice operazione matematica occorre usare la punta di una penna. L'operazione è talmente difficile che anche i più entusiasti non avrebbero il coraggio di elaborare un programma di 50 linee in BASIC su una simile tastiera. Per questo motivo, sarà difficile che il personal computer diventi piccolo quanto un orologio, sebbene sia diventato piccolo quanto una calcolatrice.

Le limitazioni riguardanti le dimensioni riflettono la nostra dipendenza dalla tastiera come dispositivo principale di input e il tubo a raggi catodici come dispositivo di output. Tuttavia, in un futuro non troppo lontano, sarà possibile produrre computer parlanti, di ridottissime dimensioni, con le stesse capacità di quelli "grandi": per l'input ci si servirà di un piccolo microfono e per l'output di un altoparlante o di una minuscola stampante.

Quando le calcolatrici sono divenute programmabili (per quanto in linguaggi molto particolari), il passo per incorporare l'uso di un linguaggio ad alto livello è stato breve e la scelta più ovvia è stata per il BASIC. Quasi contemporaneamente, i fabbricanti hanno incominciato ad usare una speciale RAM di tipo "non volatile" (cioè che non perde il suo contenuto quando l'apparecchio è spento) ed un chip "generatore di caratteri" più sofisticato, in grado di visualizzare sia le lettere dell'alfabeto che i numeri.

Ci sono dei computer tascabili, come quelli prodotti dalla Sharp e dalla Casio, che sono dotati di una gamma di comandi in BASIC e di una capacità di memoria paragonabili a quelle offerte dagli home computer: questi apparecchi stanno via via prendendo il posto delle "vecchie" calcolatrici programmabili. Alcuni di questi minuscoli computer offrono anche la possibilità di collegarsi direttamente a stampanti ed a registratori a cassette: quando le vendite raggiungeranno un certo livello, saranno probabilmente forniti di circuiti incorporati che permetteranno la trasmissione di informazioni via linea telefonica. Quelli prodotti dalla Casio, in particolare, incontrano il favore degli scienziati e degli ingegneri, perché presentano una vasta gamma di funzioni matematiche e scientifiche.

Esistono altri modelli, con stampante incorporata e completi di interfaccia per unità cassette dal costo di circa 300.000 lire: il PCI1151 della Sharp ne è un esempio.

Su un gradino più alto stanno modelli quali l'Epson HX-20, il Tandy 100 e il NEC 8220. Questi microcomputer offrono una completa implementazione del BASIC (quasi sempre nella versione della Microsoft), una memoria RAM da 16 a 64 Kbyte, un display incorporato, a cristalli liquidi, di misura adeguata (20 × 4 caratteri per quanto riguarda l'Epson; 40 ×

8 per le altre due marche) e la possibilità di collegamento ad una vasta gamma di strumenti periferici standard.
In effetti, se collegati ad un monitor normale (può essere necessaria un'apposita interfaccia), questi apparecchi sono paragonabili, per capacità e rendimento, a dei veri e propri home computer. Il pregio maggiore di questi piccoli computer risiede nel fatto che essi sono alimentati a batterie e, pertanto, sono portatili. L'Epson, per esempio, è alimentato con una serie di pile al nichel-cadmio, incorporate, consentendo un'autonomia di circa 50 ore. Il sistema, inoltre, è progettato in modo che, quando le pile si sono scaricate al disotto di un certo livello, l'apparecchio si spegne automaticamente per non perdere i dati già inseriti.
Diversamente, le macchine della NEC e della Tandy usano un circuito semiconduttivo complementare ad ossidi di metallo (CMOS), che richiede meno corrente, consentendo il funzionamento anche con normali pile a stilo.
L'Epson (il primo microcomputer portatile ad essere reclamizzato su larga scala) è stato concepito originariamente come apparecchio destinato ai dirigenti d'azienda, ma, indubbiamente, è più probabile che apparecchi simili vengano utilizzati fuori dell'ufficio, in quanto sono adattissimi per raccogliere dati durante spostamenti, ad esempio, nei giri d'ispezione in fabbrica, in cantiere o perfino per la strada. I dati raccolti saranno poi rielaborati al ritorno in ufficio.
La macchina della Epson offre svariati sistemi per la trasmissione dei dati: ci sono accessori che servono per collegarsi al Telex oppure, mediante un Modem, per dialogare con un'altra macchina per mezzo delle linee telefoniche. Altrimenti, ed è il caso più semplice ed economico, i dati possono essere conservati su di un nastro magnetico, che può venir spedito per posta.
Proprio questa versatilità d'impiego ha influito moltissimo sul successo dei portatili, anche in quella fascia di persone che, finora, non pensavano che un computer potesse loro servire per il lavoro di tutti i giorni. Un rappresentante, per esempio, durante il suo giro giornaliero dei clienti, può raccogliere gli ordini usando un computer portatile di queste dimensioni, invece di riempire manualmente i moduli per l'ordinazione. Il programma per l'elaborazione degli ordini potrebbe visualizzare l'elenco dei prodotti chiedendo, ed automaticamente memorizzando, la quantità ordinata per ciascun articolo. Finita la visita al cliente, il rappresentante registra l'ordine completo su cassetta e alla fine della giornata si troverà con una serie di ordinazioni registrate, pronte ad essere spedite alla sede o da trasmettere direttamente al computer principale della ditta. La stampante del microcomputer fornisce una copia dell'ordine per il rappresentante ed una per il cliente, per un'immediata conferma.
Una versione leggermente più sofisticata sarebbe in grado di segnalare il livello di disponibilità di ciascun articolo, avvisando automaticamente il rappresentante qualora l'ordine superi questo livello. In questo caso, però, occorrerebbe collegare il microcomputer con il sistema principale ad ogni ordine, per aggiornare continuamente il livello di disponibilità. Questa operazione si può fare aggiungendo un accoppiatore acustico leggero e poco costoso. A questo punto il nostro computer a buon mercato è diventato un vero terminale interattivo con la possibilità di accedere alla banca dati della ditta.
A parte l'ovvio beneficio che si trae dall'immediato accesso ad informazioni aggiornate, il notevole risparmio di tempo e di danaro proveniente da un simile sistema "a tempo reale" consente di ammortizzare in pochi mesi il costo iniziale delle apparecchiature.
Abbiamo già accennato al fatto che i computer tascabili stanno prendendo il posto delle calcolatrici programmabili: analogamente, i computer portatili stanno prendendo il sopravvento sugli apparecchi per la raccolta di dati non programmabili. Questi ultimi, infatti (chiamati anche "terminali muti", non essendo programmabili), sono in vendita da tempo ma non hanno mai avuto grande successo a causa dell'alto costo ed il loro difficile uso.
Grazie ad un software sempre migliore ed


IAN McKINNELL

Una macchina per ogni stagione

L'Epson HX-20 è il primo esempio di una nuova generazione di computer autosufficienti: offre fino a 32 Kbyte di memoria RAM e una vasta gamma di accessori periferici.
Fra i computer portatili, in vendita attualmente, ci sono il PC 1251 della Sharp (nella foto insieme alla stampante e all'unità a microcassette) e il FX700P della Casio, collegabile ad un normale registratore a cassette. Le loro caratteristiche tecniche sono leggermente inferiori rispetto all'Epson HX-20, ma anche il prezzo è più basso!
La Osborne Computer Company, dopo lo strepitoso successo ottenuto con le sue macchine "trasportabili", ha perfezionato il secondo modello (l'Executive), che ha, tra gli altri, il notevole pregio di uno schermo più grande



IAN McKINNELL

Stenografia elettronica

Il concetto di "notes elettronico" si è tradotto in realtà con l'introduzione del Microwriter, un sistema di elaborazione di testi, manovrabile con una sola mano. Qualsiasi lettera dell'alfabeto, oltre a numerosi simboli grafici, viene impostata da una combinazione dei sei tasti della macchina, permettendo, ovviamente con un po' di pratica, di "prendere appunti" velocemente ed impegnando la sola mano destra. L'output avviene direttamente sulla stampante oppure viene trasmesso a computer più grandi per una successiva elaborazione. Il Microwriter dispone di una serie di comandi per modificare il formato di stampa

I FERRI DEL MESTIERE

I computer portatili vengono oggi impiegati anche in aree finora al di fuori della portata di sistemi di elaborazione di dati:

Il giornalista memorizza sulla macchina portatile le domande che desidera fare durante una particolare intervista e, nel corso di essa, immette le risposte. L'articolo, a questo punto, può anche essere scritto durante il viaggio di ritorno a casa, per esempio in treno.



Nei laboratori, il computer mette a disposizione un modo economico e facile per conservare i dati relativi ad analisi o alla composizione di particolari prodotti.





DAVID HIGHAM

Il rappresentante può fornire un preventivo immediato, anche facendo complicati calcoli, per esempio il costo di un impianto di riscaldamento o una polizza assicurativa sulla vita.



Il controllo sulle giacenze di magazzino è molto importante nelle piccole aziende, ma nell'industria è addirittura essenziale avere un quadro sempre aggiornato sulla situazione. I microcomputer portatili con 32 o 64 Kbyte di memoria RAM, dotati di memorie di massa adeguate, e collegabili all'elaboratore centrale, sono l'ideale per questa applicazione

alle possibilità di collegamento con normali periferiche da ufficio (comprese le unità a dischi), i computer portatili dovrebbero presto divenire diffusissimi nel campo del lavoro. Inoltre, macchine portatili di questo genere costituiscono la prima vera occasione per l'impiego delle memorie "a bolle": memorie non volatili, basate su chip radicalmente diversi da quelli usuali, dotati di una grandissima capacità, sebbene di dimensioni ridottissime. Un esempio è il PC500 della Sharp: le cartucce di memoria contengono 128 Kbyte, con tempi di accesso inferiori rispetto a quelli dei dischetti. La possibilità di poter avere piccoli computer con più di un milione di byte di memoria è davvero allettante!

Simili capacità di memoria sono già disponibili in macchine quali l'Executive della Osborne, l'Ajile Hyperion e il Portico Miracle. Il costo di questi apparecchi, che sarebbe più giusto indicare come "trasportabili" (poiché funzionano principalmente con la corrente di rete) è, però, cinque o sei volte superiore a quello dell'HX-20 o del TRS100.

In questa fascia, gli apparecchi sono dotati, di solito, di due unità a dischetti, di un monitor e di una tastiera incorporati, ed impiegano un sistema operativo CP/M.

Il primo esempio di questo genere ad apparire sul mercato è stato l'Osborne 1, che, ad un prezzo di circa 4 milioni, offre anche una gamma completa di software: BASIC, Wordstar, Supercalc e altri programmi di "utilità" CP/M. L'unico "neo" è lo schermo molto piccolo (mm. 100 × 85), con caratteri fatalmente appena leggibili, sebbene sia possibile collegare ad esso un monitor di dimensioni normali. Comunque, questo fatto sembra non aver ostacolato minimamente le vendite, in quanto l'Osborne 1 è presto diventato uno tra i computer più venduti nel mondo.

Le ditte concorrenti si sono date molto da fare per offrire caratteristiche migliori, particolarmente per quanto riguarda le capacità e la velocità del microcomputer, e la quantità di memoria disponibile, finché questa categoria di macchine ha raggiunto lo stesso livello tecnico di altri computer "fissi" più sofisticati. In molti casi i "trasportabili" offrono una completa compatibilità con macchine specifiche (particolarmente con il Personal Computer della IBM), rendendoli utilissimi come "secondo computer".

Spesso, gli stessi programmi applicativi, sviluppati per il computer "fisso", possono essere adoperati anche su macchine portatili. L'unica limitazione può derivare dalle diverse dimensioni del monitor e da eventuali diversità nei codici dei caratteri grafici.

C'è, attualmente, una notevole quantità di innovazioni tecniche in questo campo e non è difficile prevedere sviluppi ancora più interessanti di quelli avvenuti fino ad ora. Già adesso si può immaginare, per esempio, di collegare lo Spectrum della Sinclair con il monitor a schermo piatto e microdrive della stessa ditta, alimentando il tutto con una serie di batterie al nichel-cadmio. Date le sue ridotte dimensioni, lo Spectrum è un apparecchio ideale per un'applicazione di questo genere, ma poiché la maggioranza dei computer opera con un voltaggio ottenibile da normali batterie, quasi tutti potrebbero diventare delle macchine portatili.

Una nuova ed interessante tendenza, nei nuovi portatili, si nota nell'Apricot della ACT. Questo computer è semi-portatile: scollegando il monitor, la parte restante si richiude come una valigetta. In realtà, la parola "portatile", applicata ai computer, è ancora un po' vaga: molti cosiddetti microcomputer sono semplicemente dotati di una maniglia per facilitarne il trasporto da un luogo all'altro, ma non sono concepiti come unità autosufficienti. Dobbiamo però riconoscere un fatto importante: l'introduzione di una vasta gamma di microcomputer portatili ha costituito un grandissimo passo in avanti verso un totale sfruttamento del computer in tutta la sua potenzialità: mettere a disposizione di tutti (anche dei non esperti) la potenza di elaborazione del computer.

L'Epson HX-20

Anatomia di microcomputer di diffusione europea: tecnologie a confronto

L'Epson HX-20 è stato il primo computer veramente portatile ad essere immesso sul mercato. Essendo programmabile in BASIC, le sue applicazioni sono molto più numerose rispetto anche alle più sofisticate calcolatrici elettroniche. La leggerezza (meno di 2 chili) ed il minimo ingombro ne fanno un apparecchio facilmente trasportabile in una normale valigetta "24 ore".

Poiché l'HX-20 è il frutto di un criterio di progetto totalmente nuovo, c'è ancora una certa scarsità di software, ma il BASIC, fornito di corredo, è di facile impiego anche per il principiante e molti, tra gli acquirenti di questo portatile, si sono scritti da soli i programmi per le più svariate applicazioni.

L'apparecchio è dotato di un visualizzatore a cristalli liquidi (simile a quello degli orologi digitali) in grado di rappresentare 4 righe di testo, ciascuna composta da 20 caratteri. In funzionamento grafico, si possono visualizzare 120 × 32 punti. Nella maggior parte delle applicazioni, il display funge da "finestra" sull'intero testo: con appositi tasti è possibile spostarla sul particolare punto che interessa esaminare.

Il meccanismo di stampa, incorporato, adopera normali rotoli di carta da calcolatrice, e le righe di stampa sono composte di 24 colonne.

Il registratore a microcassette è opzionale, e si inserisce nel contenitore principale del computer: l'aspetto globale che ne risulta è quello di un apparecchio molto compatto e solido. Il registratore è di qualità nettamente superiore ai normali registratori a cassette: il computer controlla automaticamente l'"avanzamento rapido" del nastro, facilitando la ricerca di programmi o dati. Al posto del registratore è previsto l'inserimento di cartucce allo "stato solido" (cioè contenenti circuiti elettronici integrati), anche se queste non sono ancora in commercio.

La gamma di interfacce disponibili è veramente interessante, permettendo di collegare, tra l'altro, anche una penna ottica (vedere a pag. 156).

I normali 16 Kbyte di memoria RAM possono aumentare fino a 32, usando una cartuccia da inserire nella parte posteriore dell'apparecchio.

Per il collegamento dell'apparecchio ad altri simili (oppure a computer più grandi) è disponibile sia l'hardware che il software necessario, a prezzi ragionevoli.



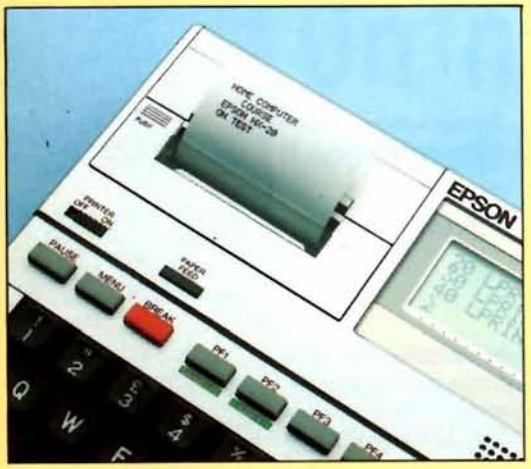
CHRIS STEVENS

Le dimensioni dell'HX-20 sono, in effetti, determinate principalmente dalla tastiera, i cui tasti sono di grandezza normale e dal "tocco" molto comodo per chi sia abituato alla macchina da scrivere.

Una particolarità consiste nella possibilità di usare parte dei tasti (U, I, O, J, K, L e M) come tastierina numerica, premendo il tasto NUM: è una comodità per chi deve immettere grandi quantità di numeri.

I pulsanti per il movimento del cursore, assieme a quello SCRN per lo spostamento della "finestra", si trovano nella parte superiore della tastiera.

Sono inoltre disponibili 5 tasti per selezionare speciali funzioni



La stampante incorporata

Piccole stampanti ad aghi sono comunemente impiegate per i registratori di cassa e per le calcolatrici da tavolo. La Epson, che ha iniziato la propria attività fabbricando stampanti, ha pensato bene di utilizzare uno dei propri modelli miniaturizzati, anche per immagini grafiche, nell'HX-20



Presa d'alimentazione

Se collegato ad un apposito alimentatore, l'HX-20 funziona con la tensione di rete, ricaricando, al tempo stesso, le batterie interne

Pulsante di RESET

Presa per lettore di codice a barre

Porta per registratore

È possibile collegare un normale registratore a cassette, sebbene i risultati migliori si abbiano usando l'apposito registratore a microcassette



La memoria di massa

Le varie funzioni del registratore a microcassette (simile a quelli impiegati nei dittafoni portatili) vengono controllate dal computer, facilitando la ricerca di programmi o dati. Il registratore si inserisce in un'apposita fessura del contenitore, e non sono necessari cavi di collegamento

Circuito stampato

Nell'illustrazione, la scheda dei componenti è stata ribaltata per mostrare i componenti. Apposite finestrelle sul fondo del contenitore permettono di accedere ai circuiti integrati più importanti, in caso di sostituzione

I microprocessori

Due microprocessori 6301 (fabbricati dalla Epson) controllano il funzionamento del computer e delle interfacce. Fatto abbastanza insolito, ciascun microprocessore contiene 4 Kbyte di ROM e 128 byte di memoria

RAM

L'HX-20 ha 16 Kbyte di memoria RAM, utilizzando otto chip da 2 Kbyte ciascuno



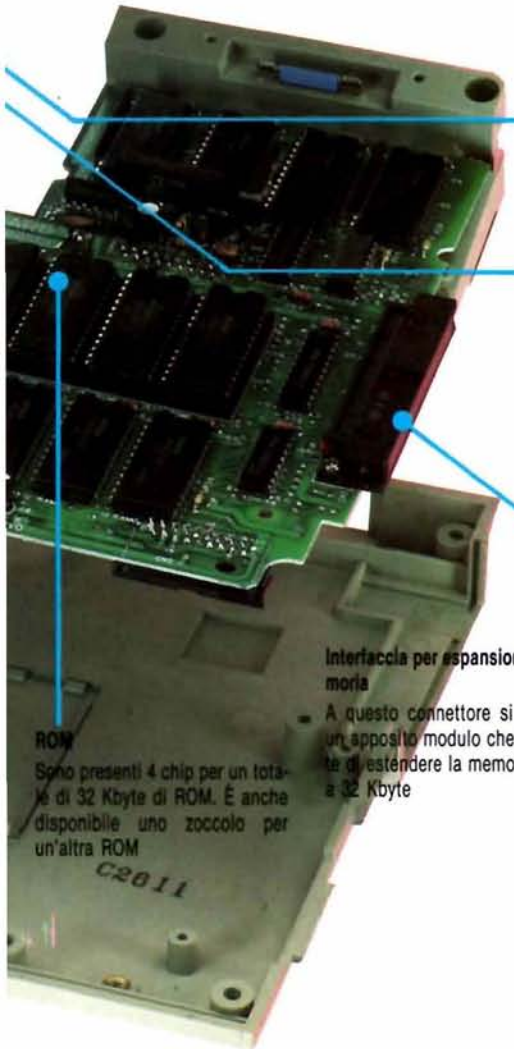
Pulsante per l'estrazione del registratore

Permette di rimuovere l'intero registratore a microcassette. Al suo posto è previsto l'uso di cartucce (non ancora in vendita) contenenti programmi specifici

Controllo di visualizzazione

Serve per ottimizzare la visualizzazione, che, con i display a cristalli liquidi, dipende fortemente dall'angolo d'incidenza della luce

Interruttore generale



Interfaccia seriale

Sebbene la porta RS232 sia anch'essa seriale, questa interfaccia è prevista per il collegamento di ulteriori periferiche

Porta seriale RS232

Questo connettore permette di collegare l'HX-20 a modem, a stampanti oppure ad altri computer per lo scambio di informazioni

Interfaccia per espansioni di memoria

A questo connettore si collega un apposito modulo che permette di estendere la memoria RAM a 32 Kbyte

ROM

Sono presenti 4 chip per un totale di 32 Kbyte di ROM. È anche disponibile uno zoccolo per un'altra ROM



Un display di nuova concezione

Il sistema di visualizzazione (generalmente un tubo a raggi catodici) influisce sensibilmente sulle dimensioni di un microcomputer. Nell'Epson HX-20 il problema è stato risolto impiegando un visualizzatore a cristalli liquidi. Simili display sono adoperati nelle calcolatrici tascabili, ma quello dell'HX-20 permette di visualizzare caratteri alfanumerici su 4 righe di 20 caratteri

EPSON HX-20

DIMENSIONI

mm. 290 x 210 x 45

PESO

1.700 grammi

CPU

6301 Hitachi

CLOCK

2,45 MHz

MEMORIA

16 Kbyte di RAM, estendibile a 32 Kbyte. 32 Kbyte di ROM espandibili a 64 Kbyte

VIDEO

Il display, del tipo a cristalli liquidi, ha 4 linee di 20 caratteri e rappresenta una "finestra" mobile di una pagina di 255 x 255 caratteri. Ogni punto del display è indirizzabile, con risoluzione massima di 120 x 32 punti

INTERFACCE

Porta per espansione di memoria, per registratore a cassetta. Connettore RS232 (standard) e porta seriale (per future unità a disco). Connettore per lettore di codice a barre

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

FORTH

ACCESSORI COMPRESI

Alimentatore, contenitore in plastica a valigetta, 2 manuali

TASTIERA

60 tasti del tipo "macchina da scrivere", 5 tasti programmabili e 4 tasti "funzione"

DOCUMENTAZIONE

L'HX-20 viene corredato di due manuali: quello operativo e quello dedicato al BASIC. Sono ambedue molto chiari e completi, anche se mancano gli indici. La versione del BASIC è quella della Microsoft, oggi molto diffusa.
Il manuale operativo comprende una sezione esauriente sulle possibili connessioni mediante la porta seriale RS232, utilissima nello scambio di informazioni con altri computer

Un numero... a caso

La generazione di numeri casuali in BASIC

Un'altra interessante funzione "intrinseca" del BASIC è la RND, che serve per la generazione di numeri casuali. Il suo impiego più comune è nei programmi di gioco, ma talvolta viene usata anche in applicazioni più "serie".
Sfortunatamente, la funzione RND è molto volubile, nel senso che il suo uso varia moltissimo da una versione all'altra di BASIC. In questo corso di programmazione è stata adottata la versione di BASIC prodotta dalla casa americana Microsoft, chiamata MBASIC. Pur non esistendo nessuno standard ufficiale riguardante il BASIC (mentre esiste, per esempio, per il linguaggio COBOL), questa versione viene spesso considerata uno standard "di fatto", essendo molto diffusa.
La principale differenza tra il MBASIC e quelli più recenti risiede nella gestione della grafica: quando venne sviluppato il MBASIC, infatti, i personal e gli home computer non erano dotati di grandi capacità grafiche. Per sfruttare appieno le funzioni grafiche, quindi, occorre studiare attentamente il manuale specifico relativo all'apparecchio acquistato. Di tutte le versioni di BASIC, differiscono maggiormente quelle adottate per lo ZX81 e lo Spectrum (della Sinclair) e per il BBC. Anche il BASIC disponibile sul TI99/4A si discosta sensibilmente dal MBASIC. Nella rubrica "Basic & Basic", vengono via via presentate le modifiche che è necessario apportare ai nostri programmi per adattarli ai vari apparecchi.
Per illustrare il funzionamento della funzione RND, abbiamo preparato un programma che simula il gioco dei dadi. Come nei precedenti programmi, anche in questo abbiamo impiegato molte subroutine (sottoprogrammi), per facilitare sia la stesura, sia la comprensione del programma.



CAROLINE HOLDEN

Il programma principale inizia con l'istruzione RANDOMIZE (linea 20): nella maggior parte delle versioni di BASIC occorre, infatti, fornire al computer un valore iniziale per la generazione di numeri casuali, altrimenti le varie sequenze di numeri rischiano di non essere, di fatto, casuali, ma ripetitive. Alla linea 50 viene chiamato il sottoprogramma che assegna un valore "a caso" alla variabile D, mediante la funzione RND. Nel nostro esempio abbiamo scritto:

```
320 LET D = INT(10 * RND)
```

ed è molto probabile che questa linea debba essere modificata per i computer che non impiegano il MBASIC (vedere "Basic & Basic").
La funzione RND ha, in genere, anche un argomento, racchiuso (come al solito) tra parentesi. Se non viene specificato alcun argomento, la funzione RND produce un numero casuale compreso tra 0 ed 1. Poiché nel nostro programma non sono utili i numeri minori di 1, possiamo moltiplicare per 10 detto valore, usando un'espressione del tipo LET A = 10 * RND, in modo tale che se, per esempio, il numero generato è 0,125455, il valore di A sarà 1,25455. Il valore di A potrà quindi variare tra 0 e 10.
Ma ai nostri fini non occorre la parte decimale del numero, che eliminiamo impiegando la funzione INT, ossia: LET A = INT(10 * RND). In alcune versioni di BASIC è possibile specificare il limite massimo del numero casuale da generare: nel Dragon 32, per esempio, si può scrivere PRINT RND(6) per visualizzare un numero generato "a caso" e compreso tra 1 e 6.
Poiché il MBASIC non consente questa possibilità, dobbiamo eseguire un controllo sul valore di D, per accertarci che sia compreso tra 1 e 6:

```
330 IF D > 6 THEN GOTO 320
340 IF D < 1 THEN GOTO 320
```

Queste due linee di programma servono per far ripetere la generazione del numero casuale qualora esso risulti esterno all'intervallo voluto.
Il sottoprogramma termina regolarmente con l'istruzione RETURN, per ritornare al programma principale, che provvede a visualizzare (linea 70) il messaggio "IL TUO PUNTEGGIO È:", seguito da una rappresentazione grafica della faccia del dado risultante dal "lancio". La faccia appropriata del dado viene selezionata dalla subroutine che inizia alla linea 390. Se il valore di D è 1, per esempio, alla linea 410 viene richiamata la subroutine per la visualizzazione della faccia 1 (linee 530-590):

```
410 IF D = 1 THEN GOSUB 530
```

La subroutine in questione non è che una serie di opportune istruzioni PRINT, che costruiscono l'immagine della "faccia" usando degli asterischi. Se il nostro computer fosse dotato di una grafica più sofisticata, potremmo riscrivere le sei subroutine di visualizzazione, per creare un'immagine più reale del dado.
Il programma, dopo aver visualizzato il "lancio" del giocatore, ne fa fare uno al computer. Il gruppo di istruzioni che servono per stabilire chi abbia ottenuto il punteggio maggiore fa parte del programma principale: la linea 200 confronta il valore di M (il risultato ottenuto dal giocatore) con quello di C (il risultato del computer) per vedere se questi sono uguali. Se lo sono, alla variabile S$ viene assegnato il valore alfanumerico "SIAMO PARI!". La linea 210 controlla, invece, se il valore di M è maggiore di quello di C: in tal caso, ad S$ viene assegnato il valore "HAI VINTO!", mentre se M è minore di C la frase assegnata a S$ sarà "HA VINTO IL COMPUTER!". La linea 240 visualizza il risultato ed il gioco è concluso. Sebbene il programma sia relativamente lungo, il principio di funzionamento è, in effetti, molto semplice: impiega una sola funzione (la RND), non ha alcun ciclo FOR-NEXT, nessuna variabile indicizzata e solo qualche frase IF...THEN.
Avendo rilevato in precedenza che l'uso della funzione RND varia da versione a versione di BASIC (in alcune manca addirittura!), vediamo se è possibile generare numeri casuali con sistemi alternativi.
Un metodo può consistere nell'impiego dell'istruzione INKEY$. Quando nell'esecuzione di un programma il BASIC, incontra questa istruzione la tastiera viene "ispezionata" per vedere se l'operatore sta premendo un tasto. Differentemente dall'istruzione INPUT, in questo caso il computer non si ferma ad attendere che il tasto venga premuto, cosicché la INKEY$ viene solitamente inserita all'interno di un ciclo FOR-NEXT, per dare tempo all'operatore di digitare un carattere. Spesso, sempre all'interno del ciclo, viene eseguito un confronto (per vedere se il tasto premuto corrisponde ad un particolare carattere). Vediamo un esempio:

```
10 PRINT "PREMERE LA BARRA SPAZIATRICE"
20 FOR X = 0 TO 1
30 LET R = R + 1
40 LET A$ = INKEY$
50 IF A$ = " " THEN GOTO 80
60 LET X = 0
70 NEXT X
80 FOR Q = 0 TO 1
90 IF R < 10 THEN GOTO 130
100 LET Q = 0
110 LET R = R/10
120 NEXT Q
130 PRINT INT(R)
140 END
```

Il valore di R è un numero casuale?
Esaminiamo il programma per conoscere la risposta.
La linea 10 visualizza il messaggio "PREMERE LA BARRA SPAZIATRICE". Prima che noi abbiamo il tempo di digitare un carattere spazio, il programma è già entrato nel ciclo FOR X = 0 TO 1 (linea 20). I valori 0 ed 1 possono sembrare piuttosto strani in questo caso, ma vedremo tra breve il perché del loro impiego. Poiché il MBASIC azzera tutte le variabili prima di eseguire un programma, la prima volta che viene eseguita la linea 30 viene assegnato ad R il valore 1. La linea 40 assegna ad A$ il valore corrispondente al tasto che viene premuto sulla tastiera. La linea 50 esegue un confronto, per vedere se il carattere premuto è uno spazio: se lo è, l'esecuzione prosegue attraverso la GOTO 80, altrimenti passa alla linea 60. In questa linea viene assegnato il valore 0 alla variabile X. Ma, poiché la variabile X rappresenta il contatore del ciclo, se X vale 0 la seguente istruzione NEXT (linea 70) fa sì che il ciclo continui a ripetersi indefinitamente, purché il contenuto di A$ sia diverso da uno spazio.
Soltanto se ad un certo punto viene premuta

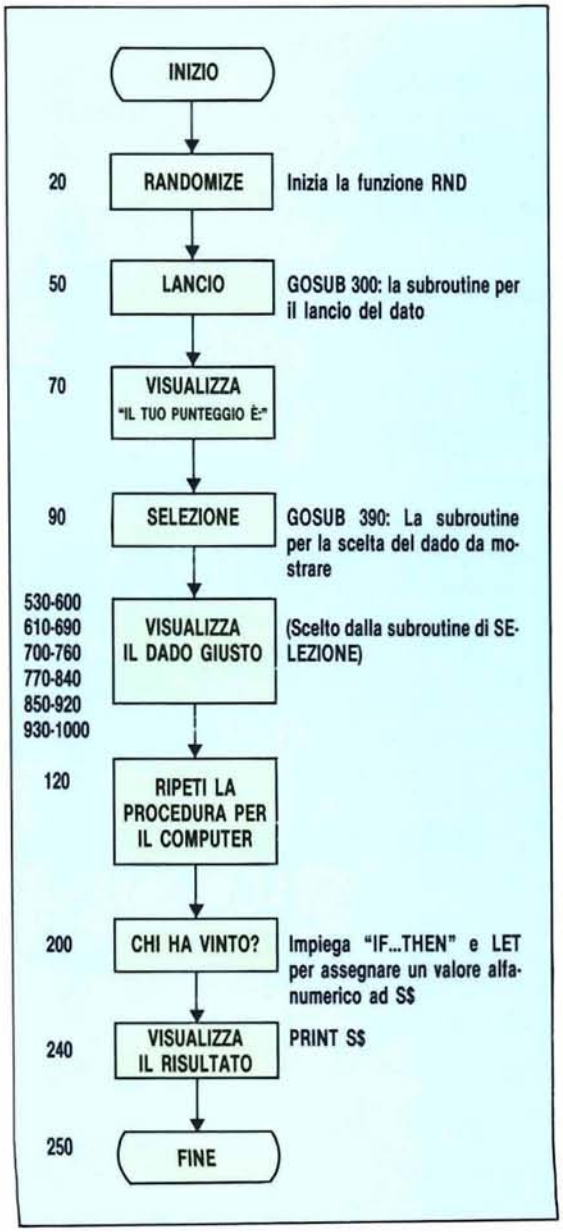


Diagramma di flusso
Il diagramma mostra, in forma sintetica, le varie fasi del programma. A sinistra di ciascun blocco sono riportati i numeri di linea corrispondenti al listato, mentre a destra è riportata una breve spiegazione su ogni fase. Non si tratta di un flowchart completo, poiché non sono indicati i blocchi "decisionali", presenti invece nel programma

```
  10 REM GIOCO DEI DADI --- PROGRAMMA PRINCIPALE
  20 RANDOMIZE
  30 REM "LANCIO" DEL GIOCATORE
  40 REM CHIAMATA DELLA SUBROUTINE DI "LANCIO"
  50 GOSUB 300
  60 LET M = D
  70 PRINT "IL TUO PUNTEGGIO E`:"
  80 REM CHIAMATA DELLA SUBROUTINE DI "SELEZIONE"
  90 GOSUB 390
 100 PRINT
 110 REM "LANCIO" DEL COMPUTER
 120 REM CHIAMATA DELLA SUBROUTINE DI "LANCIO"
 130 GOSUB 300
 140 LET C = D
 150 PRINT "IL PUNTEGGIO DEL COMPUTER E`:"
 160 REM CHIAMATA DELLA SUBROUTINE DI "SELEZIONE"
 170 GOSUB 390
 180 PRINT
 190 REM CHI HA VINTO ?
 200 IF M = C THEN LET S$ = "PARI"
 210 IF M > C THEN LET S$ = "HAI VINTO!"
 220 IF M < C THEN LET S$ = "HA VINTO IL COMPUTER!"
 230 REM  VISUALIZZA IL RISULTATO
 240 PRINT S$
 250 END
 260 REM
 270 REM
 280 REM
 290 REM
 300 REM SUBROUTINE DI "LANCIO" DEL DADO
 310 REM
 320 LET D = INT(10*RND)
 330 IF D > 6 THEN GOTO 320
 340 IF D < 1 THEN GOTO 320
 350 RETURN
 360 REM
 370 REM
 380 REM
 390 REM ROUTINE DI "SELEZIONE"
 400 REM
 410 IF D = 1 THEN GOSUB 530
 420 IF D = 2 THEN GOSUB 610
 430 IF D = 3 THEN GOSUB 690
 440 IF D = 4 THEN GOSUB 770
 450 IF D = 5 THEN GOSUB 850
 460 IF D = 6 THEN GOSUB 930
 470 RETURN
 480 REM
 490 REM
 500 REM
 510 SUBROUTINE "GRAFICA"
 520 REM
 530 PRINT "-------------"
 540 PRINT "|           |"
 550 PRINT "|           |"
 560 PRINT "|     *     |"
 570 PRINT "|           |"
 580 PRINT "|           |"
 590 PRINT "-------------"
 600 RETURN
 610 PRINT "-------------"
 620 PRINT "|           |"
 630 PRINT "|         * |"
 640 PRINT "|           |"
 650 PRINT "| *         |"
 660 PRINT "|           |"
 670 PRINT "-------------"
 680 RETURN
 690 PRINT "-------------"
 700 PRINT "|           |"
 710 PRINT "|         * |"
 720 PRINT "|     *     |"
 730 PRINT "| *         |"
 740 PRINT "|           |"
 750 PRINT "-------------"
 760 RETURN
 770 PRINT "-------------"
 780 PRINT "|           |"
 790 PRINT "| *     *   |"
 800 PRINT "|           |"
 810 PRINT "| *     *   |"
 820 PRINT "|           |"
 830 PRINT "-------------"
 840 RETURN
 850 PRINT "-------------"
 860 PRINT "|           |"
 870 PRINT "| *     *   |"
 880 PRINT "|     *     |"
 890 PRINT "| *     *   |"
 900 PRINT "|           |"
 910 PRINT "-------------"
 920 RETURN
 930 PRINT "-------------"
 940 PRINT "|           |"
 950 PRINT "| *     *   |"
 960 PRINT "| *     *   |"
 970 PRINT "| *     *   |"
 980 PRINT "|           |"
 990 PRINT "-------------"
1000 RETURN
```

la barra spaziatrice (o il tasto "spazio"), l'esecuzione passa alla linea 80: nel frattempo, all'interno del ciclo, il valore della variabile R è stato incrementato di 1 ad ogni passaggio, cosicché, quando si arriva alla linea 80, R conterrà un valore non conoscibile a priori, ma che dipende da quanto tempo è trascorso prima che sia stata premuta la barra spaziatrice.

I computer sono molto veloci e, anche se la pressione del tasto è avvenuta quasi subito dopo il messaggio iniziale, R può contenere valori piuttosto alti: come facciamo per ottenere dei risultati, mettiamo, compresi tra 1 e 10? La risposta ci viene dal resto del programma: la linea 80 inizia un nuovo ciclo, che serve a dividere R per 10, se il suo valore è maggiore di 10. Infatti, in questo ciclo alla variabile Q viene assegnato il valore 0 fintanto che R è maggiore di 10 (il metodo è identico a quello usato precedentemente). Il valore di R viene visualizzato soltanto quando esso è minore di 10. La linea 30 ci assicura l'esclusione del valore 0.

In teoria, dunque, il programma fornisce valori casuali di R compresi tra 1 e 9, ma in pratica ciò è vero? L'istruzione INT ci garantisce, rimuovendo la parte frazionaria di R, che i possibili valori di R sono: 1,2,3,4,5,6,7,8,9. La media di questi valori è 5 e, secondo le leggi sulla distribuzione dei valori casuali, la media dei valori generati deve avvicinarsi quanto più possibile alla media dei valori "generabili".

Per una verifica, potremmo far "girare" il programma più volte, annotando i valori risultanti di R, e calcolando poi la media di questi valori, oppure potremmo aggiungere qualche istruzione, affinché il programma "giri" un numero prefissato di volte (100 ad esempio) sommando, ad ogni passaggio, il valore di R ad una variabile S. Dividendo il valore di S per 100 si otterrebbe automaticamente la media dei valori generati.

Nell'effettuare questa verifica, abbiamo trovato che i valori di R, in effetti, non erano molto casuali: la ragione dipende dal fatto che il BASIC è molto veloce, ma non abbastanza!

Nel primo ciclo FOR-NEXT, ad ogni passaggio la variabile R viene aumentata di 1 e ci sono forti probabilità che, per premere la barra spaziatrice, noi facciamo trascorrere più o meno sempre lo stesso intervallo di tempo e, nelle prove da noi fatte, i valori di R tendevano di solito ad essere "bassi". Possiamo allora ricorrere ad un altro metodo che è più soddisfacente.

Consideriamo l'esempio:

```
10 PRINT "PREMERE LA BARRA SPAZIATRICE"
20 FOR X = 0 TO 1
30 LET R = R + 1
40 IF R > 9 THEN R = 1
50 IF INKEY$ = " " THEN GOTO 80
60 LET X = 0
70 NEXT X
80 PRINT R
90 END
```

In questo programma, R non assume mai valori minori di 1 o maggiori di 9. Con questo programma abbiamo generato 1000 valori di R, la cui media si aggira sul 5,014.
Rispetto ad una media di 5, questo valore ha uno scarto di solo 0,28%, che è più che accettabile.
Spesso in un programma perfettamente funzionante "sulla carta" possono esserci degli effetti imprevedibili, che si possono rilevare soltanto attraverso attente verifiche.
I programmi che abbiamo presentato per la generazione di numeri casuali, potrebbero essere abbreviati sostituendo ai cicli FOR-NEXT delle opportune GOTO, ma la ragione di questa apparente complicazione sarà presto spiegata durante le prossime lezioni del corso.

Esercizi

✻ **Funzione RND.** Si modifichi l'ultimo programma presentato, in modo che vengano generati numeri compresi tra 1 e 6 (inclusi).

✻ **Cicli e media.** Si aggiungano opportune istruzioni all'ultimo programma presentato, affinché vengano generati 100 numeri a caso, e venga calcolata la media dei 100 valori.

✻ **Sostituzione di subroutine.** Si sostituiscano le linee 50 e 130 del programma principale (quello nel riquadro) in modo da utilizzare la subroutine sviluppata nel primo esercizio.

✻ **INKEY$.** Usando questa istruzione, scrivere un programmino che "legga" un carattere dalla tastiera e visualizzi il messaggio "IL TASTO PREMUTO ERA:", seguito dal carattere effettivamente premuto.

✻ **Ciclo di ritardo.** Si scriva, utilizzando la INKEY$, un ciclo di ritardo contenente un contatore e si esamini questo valore dopo un intervallo di tempo di 10 secondi. Prima di terminare, il programma potrebbe visualizzare il messaggio "IL VALORE DI R DOPO 10 SECONDI È:", seguito dal valore del contatore.

✻ **Confronti IF...THEN.** Si scriva un breve programma per la generazione di numeri casuali compresi tra 1 e 100, che il giocatore deve via via indovinare. Il giocatore deve avere a disposizione 5 tentativi. Il programma dovrà visualizzare, ad ogni tentativo, il messaggio appropriato tra quelli seguenti: IL TUO NUMERO È TROPPO ALTO; IL TUO NUMERO È TROPPO BASSO; CONGRATULAZIONI! HAI INDOVINATO; TROPPI TENTATIVI, HAI PERSO!

Le risposte sono nella prossima lezione.

Basic & Basic

RANDOMIZE

Sul BBC e sull'Oric-1 si cancelli la linea 30 e si sostituisca la 320 con:
320 LET D = INT(10 * RND(1))

RND

Sul Dragon 32, si cancellino le linee 20, 330 e 340, sostituendo la 320 con:
320 LET D = RDN(6)

Sul Lynx sostituire la linea 20 con:
20 RANDOM

Sul Vic-20 e sul Commodore 64 sostituire la 20 con:
20 LET X = RND(— TI)

sostituendo la 320 con:
320 LET D = INT(10 * RND(1))

Sulla tastiera dello Spectrum, l'istruzione RANDOMIZE è abbreviata con RAND, anche se sullo schermo appare per intero

INKEY$

Sul Vic-20 e sul Commodore 64, sostituire la linea 40 a pag. 173 con:
40 GET A$
e la linea 50 a pag. 174 con:
50 GET A$: IF A$ = " " THEN GOTO 80

Sul BBC sostituire INKEY$ con INKEY$(10). Il numero tra parentesi indica il numero di centesimi di secondo durante i quali il computer attende un responso. Si può variare il valore aumentandolo o diminuendolo e verificare la differenza dei risultati

Risposte agli esercizi di pagg. 148 e 149

Ciclo 1

IL VALORE DI A È: 450

Ciclo 2

INIZIO
FINE

Ciclo 3

IL VALORE DI A È: 160

Ciclo 4

MI PIACE IL BASIC
MI PIACE IL BASIC
MI PIACE IL BASIC
:
:
:
(fino a che non si preme il tasto RESET o BREAK)

Ciclo 5

IO SONO IL TUO COMPUTER
(ripetuto 15 volte)

READ-DATA 1

PROVA DELLE ISTRUZIONI READ-DATA
170

READ-DATA 2

X = 1
X = 2
:
:
X = 23

Una penna... a motore

Le tartarughe ed i robot: didattica e divertimento

Robot da pavimento e tartarughe non sono soltanto strumenti didattici, ma anche fonte di divertimento!

I principi che governano l'uso del Buggy BBC, per esempio, sono identici a quelli impiegati per i robot industriali. Sebbene i robot da pavimento non siano ancora in grado di badare alle faccende di casa, potrebbero costituire la prossima generazione di collaboratori domestici.

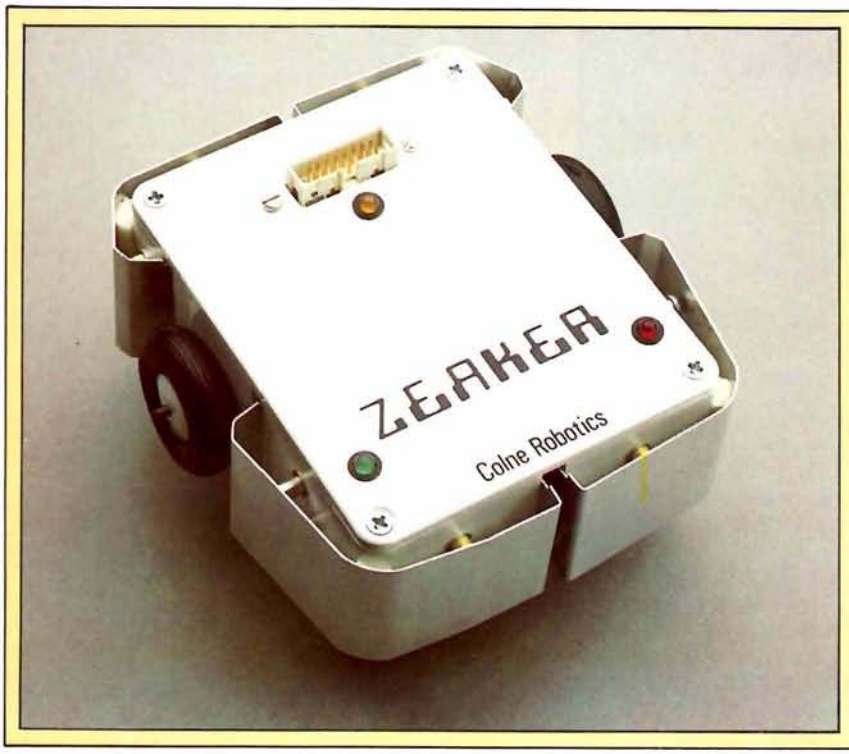
Una delle principali necessità, per un robot, è quella di potersi posizionare con esattezza all'interno di un ambiente: per questo motivo viene generalmente impiegato un motore "passo passo". Diversamente dai motori normali, un "passo passo" non si mette a girare senza sosta se si collega alla corrente elettrica, ma compie soltanto una piccola rotazione per ogni impulso inviatogli. Il preciso numero di impulsi necessari per fare una rotazione completa dipende dal particolare modello di motore.

Anche la direzione del movimento può essere controllata, impiegando più motori, tutti controllati con estrema precisione dal computer. Il robot può anche ruotare su se stesso, facendo girare le ruote in direzioni opposte.

Al tempo stesso, è altrettanto importante che il robot possa comunicare al computer eventuali ostacoli incontrati sul cammino, e particolari dispositivi (rivelatori di collisione) vengono spesso realizzati con dei paraurti collegati a minuscoli interruttori. Quando uno di questi interruttori si chiude, un segnale viene inviato al computer, che lo interpreta opportunamente.

Spesso sono necessari anche altri tipi d'informazione: per esempio, può essere utile che il robot sia in grado di seguire una linea bianca tracciata sul pavimento. Ciò si ottiene dotando il robot di una cellula fotosensibile e di una sorgente di luce. La luce riflessa dal pavimento viene "letta" dalla cellula, e l'informazione trasmessa al computer, il quale deciderà, in base alla variazione di corrente, se impartire o meno delle correzioni al movimento.

Il computer BBC è dotato di una speciale porta analogica, che consente l'uso di simili dispositivi; in altri apparecchi è necessario ricorrere a dei circuiti aggiuntivi.



LO ZEAKER

Questo apparecchio, un incrocio tra un piccolo robot ed una tartaruga, possiede rilevatori di collisione e può essere dotato di una certa "intelligenza". Lo Zeaker può spostarsi nelle quattro direzioni, oltre ad essere in grado di alzare o abbassare la punta di una penna. Il suo controllo si avvale di una particolare versione del LOGO, chiamata SNAIL LOGO (lett. LOGO per lumaca). Il suo impiego richiede l'uso di un'interfaccia analogico/digitale

CHRIS STEVENS

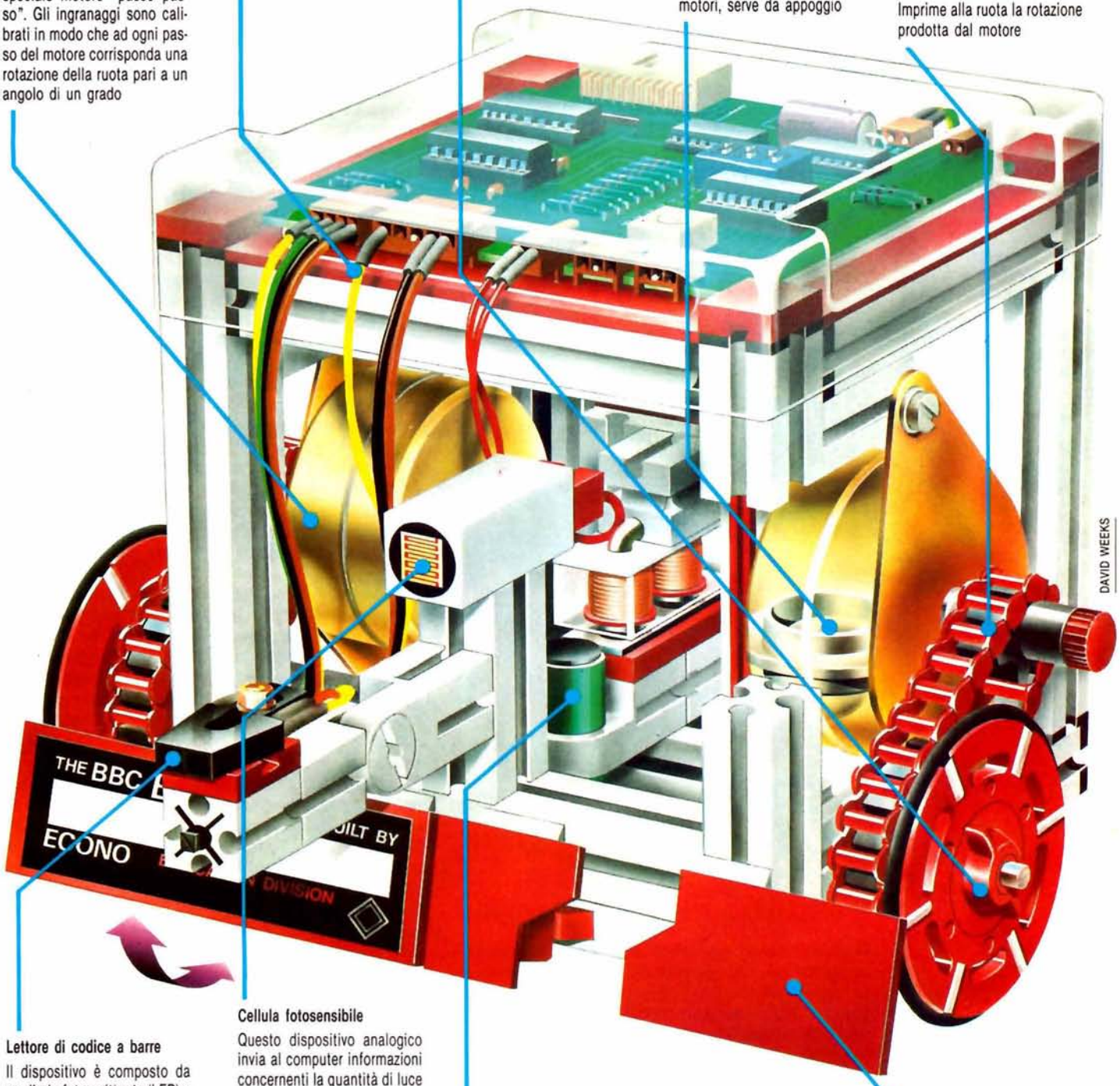
Motore "passo passo"
Ogni ruota è collegata ad uno speciale motore "passo passo". Gli ingranaggi sono calibrati in modo che ad ogni passo del motore corrisponda una rotazione della ruota pari a un angolo di un grado

Scheda di controllo
Tutti i segnali in ingresso e in uscita dal computer passano attraverso questi circuiti, per essere convertiti e smistati opportunamente

Ruota di trazione
Le ruote sono gommate, per una migliore aderenza al suolo e per permettere quindi movimenti estremamente precisi

Ruota di appoggio
Questa ruota, non collegata ai motori, serve da appoggio

Catena di trasmissione
Imprime alla ruota la rotazione prodotta dal motore



DAVID WEEKS

Lettore di codice a barre
Il dispositivo è composto da un diodo fotoemittente (LED) e da un diodo fotorilevatore (LDD). Il fascio di luce generato dal LED viene riflesso dalla superficie dove è impresso il codice a barre. Le variazioni della luce riflessa vengono rilevate dal LDD e trasmesse al computer. Inclinando il supporto, il lettore può essere impiegato per leggere particolari codici tracciati sul suolo

Cellula fotosensibile
Questo dispositivo analogico invia al computer informazioni concernenti la quantità di luce ad esso diretta. Può essere usato per individuare fonti di luce

Supporto per penna
Un accessorio utile, per certe applicazioni, può essere il supporto per penna, che viene impiegato per tracciare linee e disegni sul suolo, sotto il controllo del computer

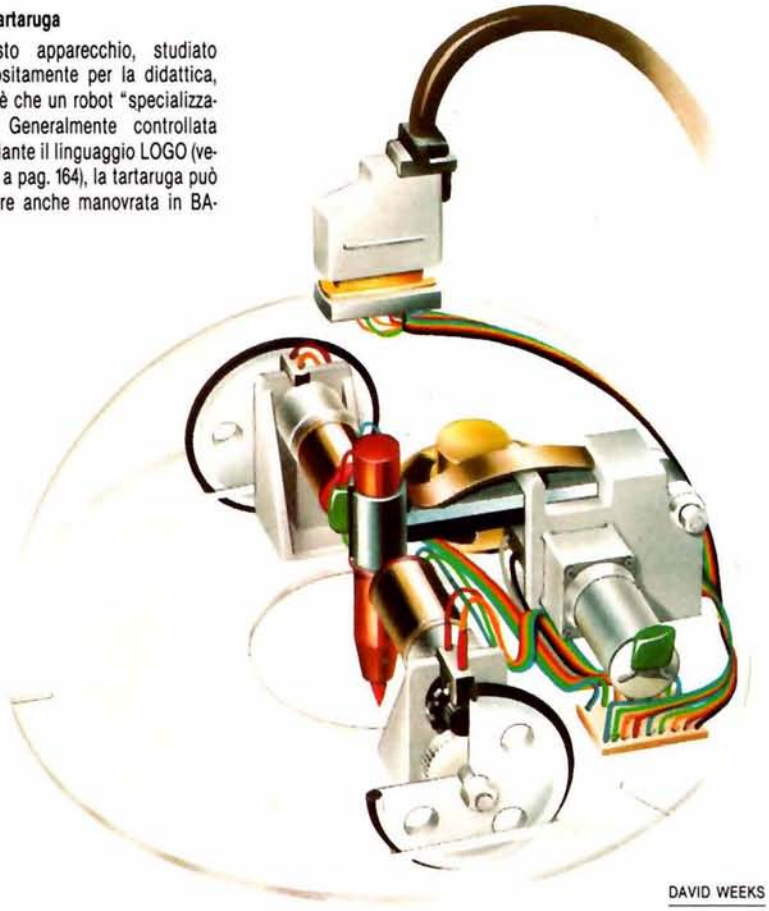
Rilevatori frontali di collisione
Collegato a minuscoli interruttori, il paraurti è in grado di comunicare se incontra ostacoli sul proprio cammino

Inoltre, con lo stesso sistema, si può far leggere al sensore un codice a barre e ciò può essere utile, per esempio, lungo gli scaffali di un magazzino: il robot si muoverà accanto agli scaffali, finché non legge il codice di un particolare prodotto. Il software fornito assieme al Buggy BBC offre anche un esempio... musicale di interpretazione di un codice a barre.

I tipi di segnali utilizzabili per lo spostamento di un robot possono essere: luci, suoni e campi magnetici. Questi ultimi vengono spesso impiegati nelle fabbriche automatizzate per indicare al robot il percorso da seguire: speciali cavi, posti sotto il pavimento, generano il campo magnetico necessario.

La tartaruga

Questo apparecchio, studiato appositamente per la didattica, non è che un robot "specializzato". Generalmente controllata mediante il linguaggio LOGO (vedere a pag. 164), la tartaruga può essere anche manovrata in BASIC



DAVID WEEKS

Il controllo di un robot è affidato a speciali programmi, che governano tutte le operazioni di Input/Output dei dati. Nel caso del Buggy BBC, vengono impiegate quattro linee per controllare il movimento dei motori. Utilizzando lo stesso cavo, il robot invia al computer le varie letture dei sensori.

I programmi in genere esaminano continuamente specifiche locazioni di memoria, dove vengono depositati i segnali ricevuti dal robot. In BASIC, il valore contenuto in una locazione di memoria può essere esaminato usando l'istruzione PEEK, mentre per modificarne il contenuto è impiegata l'istruzione POKE.

Se il computer è dotato di un convertitore analogico/digitale, l'esame dei vari dati avviene più o meno con lo stesso sistema, altrimenti i segnali devono essere prima convertiti, con appositi circuiti in valori digitali e successivamente elaborati dal computer.

Una "Tartaruga", che viene usata principalmente in associazione al linguaggio LOGO, non è che una forma elementare di robot, sebbene la distinzione fra questi due apparecchi sia sempre più nebulosa (vedere a pag. 164).

Molte delle più recenti tartarughe sono dotate di rilevatori di collisione, e molti robot vengono dotati di una penna per tracciare disegni così come fa una tartaruga. L'idea di tracciare disegni su larga scala usando un apparecchio su ruote proviene dall'insegnamento dei rapporti esistenti tra distanze, angolazioni e forme geometriche. Se una persona, per quattro volte consecutive, muove 10 passi in avanti, e si gira a sinistra di 90 gradi, avrà tracciato un grande quadrato: la tartaruga ed il robot si prestano bene a questo tipo di esperimenti.

Il Buggy BBC è fornito in scatola di montaggio ed è basato su di una meccanica Fischer Technik, alla quale è facile apportare migliorie.

Le stesse operazioni di montaggio del Buggy sono senz'altro molto istruttive, ma il vero apprendimento ha inizio quando si comincia ad usare l'apparecchio montato. Inoltre, anche se la maggior parte di simili apparecchi viene fornita completa di software, può essere molto divertente scriversi parte dei programmi da sé.

Il controllo di un robot è una buona esercitazione a livello di programmazione: il software deve essere in grado di operare in "tempo reale", esaminando tutti i dati in ingresso ed inviando, conseguentemente, opportuni segnali in uscita. È di vitale importanza, per l'apparecchio stesso, rilevare in tempo gli ostacoli (muri, scale ecc.) e reagire immediatamente.

In teoria, vi è poca differenza tra la stesura di un programma per il Buggy e quello per il controllo di un grosso impianto industriale: in ambedue i casi, si deve applicare una logica ferrea ed essere in grado di prevedere tutte le varie possibilità.

Le tecniche che si imparano lavorando con un piccolo robot possono essere veramente utili alla comprensione di sistemi basati sull'Intelligenza Artificiale. Si potrebbe addirittura programmare il robot in modo che si accorga se le proprie batterie sono scariche, e si diriga automaticamente ad una fonte di approvvigionamento, per poi riprendere il compito interrotto.

La prossima generazione di piccoli robot offrirà caratteristiche ancora più interessanti: essi saranno dotati di piccole braccia retrattili in grado di trasportare oggetti. Le attuali cellule fotosensibili saranno sostituite da minuscole telecamere (veri e propri occhi elettronici!).

Infine, circuiti per la sintetizzazione della voce li metteranno in grado di funzionare dietro semplici comandi vocali e magari anche di rispondere... a tono!

Il codice esadecimale

Contare in base 16 è un'altra specialità del computer

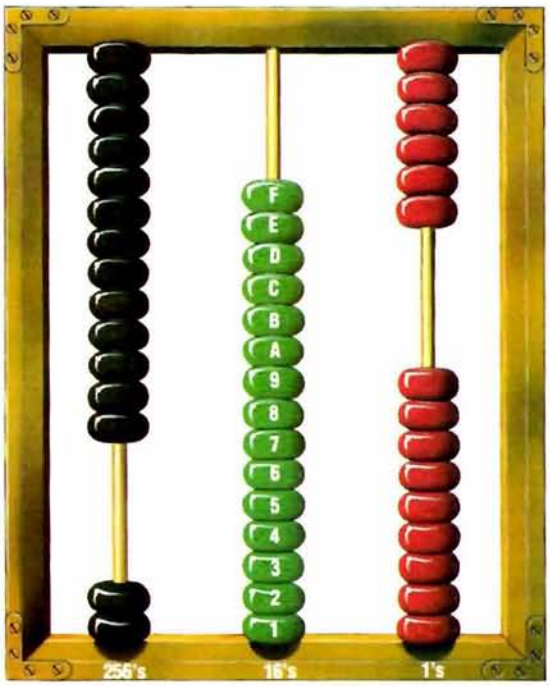
È facile capire perché si usi il sistema binario nei computer: numeri composti da una serie di 1 o di 0 si prestano egregiamente ad essere convertiti in condizioni elettriche di acceso/spento (on/off). È altrettanto facile comprendere perché l'umanità abbia adottato, quasi universalmente, il sistema decimale: abbiamo 10 dita e ciò ha influito sulla scelta del nostro sistema di numerazione.

Ma esistono numerosi altri sistemi di numerazione, tra i quali quello esadecimale (ossia con base 16): anche questo trova largo impiego nel settore informatico. La principale utilità di questo sistema proviene dalla facilità con la quale si possono convertire i numeri binari in esadecimali e viceversa. Viene spontaneo chiedersi perché, se il computer "capisce" solo numeri binari, i programmatori ricorrano ad un altro sistema numerico per la programmazione. Ci sono due motivi molto validi: in primo luogo, i numeri binari sono molto più lunghi dell'equivalente in decimali o esadecimali (per esempio il numero decimale 356 diventa in binario 0000000101100100); in secondo luogo, è molto facile commettere errori usando numeri composti esclusivamente di 1 e 0.

Il sistema esadecimale si avvale di 16 simboli diversi: i numeri dallo 0 al 9 e sei lettere dell'alfabeto. Pertanto, il numero più alto rappresentabile nella colonna delle unità equivale al numero decimale 15. I matematici potevano inventare sei simboli nuovi per indicare i valori esadecimali compresi tra il 10 ed il 15 (decimali), ma, per semplicità, hanno preferito usare le prime sei lettere dell'alfabeto. La numerazione esadecimale, quindi, procede normalmente dallo 0 al 9, dopo di ché si usano A,B,C,D,E,F. Arrivati a F (= 15 decimale) abbiamo bisogno di una nuova colonna di numeri (delle "sedicine", in contrapposizione a quella delle "decine" nella normale numerazione). Anche qui si usano gli stessi simboli a disposizione da 0 fino a F. Completata la colonna delle unità e quella delle "sedicine" occorre iniziare una terza colonna: la colonna dei "256".

La tabella in neretto illustra alcuni numeri decimali con l'equivalente in binario e in esadecimale.

Il numero 65.535 diventa in esadecimale FFFF: questo esempio ci dà subito un'idea chiara dell'utilità di questo sistema. Un numero composto di cinque cifre, nel sistema decimale, e di ben 16 cifre in quello binario, viene rappresentato in esadecimale con sole 4 cifre. A parte il vantaggio della compattezza ce n'è un altro più importante: quattro cifre in binario vengono rappresentate in esadecimale con una sola cifra, cosicché la conversione da binario in esadecimale e viceversa diventa relativamente semplice.



MARK WATKINSON

Il pallottoliere esadecimale
Un pallottoliere adatto al sistema esadecimale dovrebbe avere 15 pallottole per ogni stecca. Su quella all'estrema destra si conterebbero le unità, fino a raggiungere il valore esadecimale F. Il sistema esadecimale è basato su 16 simboli: i nove numeri ai quali fanno seguito le prime sei lettere dell'alfabeto. Come si può vedere dalla stecca nel centro, la A corrisponde al numero decimale 10, la B all'11, la C al 12, la D al 13, la E al 14 e la F al 15. Dopo il numero F non abbiamo più unità a disposizione e pertanto dobbiamo iniziare il conto su di un'altra colonna: la colonna dei sedici. Sulla colonna più a destra (quella delle unità) è stato fatto un conteggio di 9 palline. La colonna centrale (dei sedici) illustra un conteggio di F, mentre la colonna dei 256 illustra un conteggio di 2. Il numero risultante si legge perciò "2F9 Hex" ed è equivalente al numero decimale 761 (composto come segue: 256 x 2 + 16 x 15 + 1 x 9 = 761

Per trasferire un numero binario in esadecimale è necessario dividere il numero binario in gruppi di 4 bit, iniziando da destra, e convertire ciascun gruppo in esadecimale. Ecco alcuni esempi, nei quali viene impiegato il termine "hex" (abbreviazione molto diffusa di "esadecimale"):

		1110	1001	binario
		= E	9	hex
		= 233		decimale
	1111	1000	1100	binario
	= F	8	C	hex
	= 3980			decimale
0111	1110	0010	1101	binario
= 7	E	2	D	hex
= 32.301				decimale

Normalmente, nei linguaggi ad alto livello (come il BASIC, per esempio) non si usa la notazione esadecimale (come del resto quella binaria), ma in altri linguaggi a basso livello (linguaggio macchina ed Assembly) il suo impiego è basilare.

Per distinguere i numeri esadecimali dai decimali, si suole aggiungere il suffisso H (da Hexadecimal): per esempio, il numero decimale 256 è scritto, in esadecimale, 100H (nel linguaggio parlato: cento hex).

Il sistema esadecimale verrà incontrato soprattutto più avanti nel corso, quando tratteremo più da vicino il funzionamento del linguaggio macchina.

Binario	Decimale	Esadecimale
00000001	1	1
00000010	2	2
00000011	3	3
00000100	4	4
00000101	5	5
00000110	6	6
00000111	7	7
00001000	8	8
00001001	9	9
00001010	10	A
00001011	11	B
00001100	12	C
00001101	13	D
00001110	14	E
00001111	15	F
00010000	16	10
00010001	17	11
00010010	18	12
00010011	19	13
00010100	20	14
00010101	21	15

Chuck Peddle

L'uomo che ha progettato il chip 6502, il PET ed il Sirius 1

Un'intera generazione separa Chuck Peddle da altri geni dell'elettronica, quali Steve Wozniak e Steven Jobs. Il suo primo contatto con il mondo del microcomputer fu nel 1973, quando egli entrò a far parte della società Motorola, partecipando al progetto del microprocessore 6800.
Questo era il primo modello di microprocessore ad essere messo in vendita e la Motorola ne approfittò chiedendo un prezzo (200 dollari), che a Peddle sembrò decisamente tanto eccessivo che egli lasciò la ditta per prestare la sua opera nella MOS Technology.
In questa ditta, relativamente piccola, Peddle s'impegnò nella progettazione di un altro microprocessore, che sarebbe poi diventato il famoso 6502: il microprocessore che, a detta di alcuni, ebbe il più grande successo durante la prima decade dell'era del microcomputer. A quel tempo, però, nessuno poteva immaginare che il 6502 sarebbe diventato il pezzo forte di una nuova industria e avrebbe contribuito in grande misura ad una "rivoluzione" sociale di enormi proporzioni.
Una delle poche persone capaci di intuire l'importanza del microcomputer in generale, ed il potenziale del 6502 in particolare, fu Jack Tramiel, Presidente della Commodore. Fino ad allora la Commodore Business Machines produceva (senza peraltro riscuotere grande successo) una serie di macchine per ufficio e calcolatori tascabili. La Commodore era il cliente più importante della MOS Technology e comprava regolarmente da lei grandi quantità di chip "dedicati" (ossia "personalizzati") a quattro funzioni. Nonostante i problemi della Commodore per reggere la concorrenza, Tramiel credette tanto nel potenziale di vendita del 6502, che in qualche modo riuscì a trovare i mezzi finanziari per comprare lui stesso la MOS Technology. Insieme alla ditta acquistò anche l'operato di Charles Peddle, che a quell'epoca era responsabile capo per lo sviluppo del microprocessore in questione.
Durante quel periodo Peddle si rese conto che la sua invenzione poteva forse dar vita ad una strepitosa innovazione: il personal computer. Anche Wozniak e Jobs della Apple Computer Inc. stavano lavorando alle stesse applicazioni. Peddle fu così determinato nello sfruttare al massimo la nuova tecnologia che decise di associarsi con Bill Gates (fondatore della Microsoft — famosa per molti prodotti di software, tra i quali l'interprete BASIC) per tentare di acquistare la società Apple che, per pura coincidenza, era in vendita contemporaneamente alla stessa MOS Technology. Tuttavia, Wozniak e Jobs chiedevano 150.000 dollari per la società, mentre Peddle e Gates potevano arrivare a due terzi di quella cifra. Peddle, quindi, rimase alla Commodore e intraprese il progetto del PET (Personal Electronic Transactor). Questo microcomputer fu lanciato nel 1977, più o meno nello stesso periodo dell'Apple II. Il PET era sostanzialmente diverso dall'Apple, in quanto il monitor e le unità dischetti erano incorporati ed il "tocco" della tastiera era più simile a quello di un calcolatore che non ad una macchina da scrivere. Dopo poco tempo dalla sua apparizione sul mercato la Commodore ricevette ordini per 1000 unità: era nata la prima generazione di microcomputer, progettato specificamente per uso domestico.
Ci vollero ancora 3 anni prima che Peddle potesse realizzare la sua seconda ambizione: quella di avere la propria ditta produttrice di computer. Con Chris Fish (uno dei geni della finanza fautore dell'improvvisa ascesa della Commodore) entrò a far parte della Victor United, una società satellite della mastodontica Walter Kidde Corporation, e fondò la Sirius Systems Technology.
Le innovazioni nell'industria del personal computer si basavano, in quel periodo, su di un chip a 16-bit, come lo 8088 della Intel. In seguito si seppe che anche la IBM stava lavorando al progetto di un personal computer basato sullo stesso chip, ma per una serie di casi fortuiti, la Sirius fu in grado di presentare il frutto delle sue fatiche alcune settimane prima della IBM. La macchina venne accolta con grande successo ed occupò una posizione molto forte sul mercato, essendo il primo microcomputer a basso costo, prodotto su scala industriale, capace di offrire i vantaggi della nuova generazione di microcomputer a 16-bit.
Il modello Sirius 1 era relativamente economico e facile da usare, stabilendo, tra l'altro, un nuovo standard di qualità per microcomputer: una tastiera indipendente, una grafica ad alta risoluzione ed uno schermo antiriflesso. Inoltre, la capacità di indirizzamento del microcomputer a 16-bit consentiva notevoli risparmi di tempo ed un miglior utilizzo della memoria.
Chuck Peddle è riuscito, almeno in grande misura, a realizzare la sua ambizione: quella di dare al grande pubblico la possibilità di usare i microcomputer.



per concessione della SIRIUS COMPUTING

Il primo personal computer

Dopo aver progettato il microprocessore 6502, Chuck Peddle si dedicò alla realizzazione del primo "personal" computer: un apparecchio completo, utilizzabile anche da persone inesperte; basta che sia semplicemente collegato ad una presa di corrente. Il modello fu chiamato "Commodore PET" e fu messo in vendita quasi contemporaneamente all'Apple II, progettato da Steve Wozniak. La maggiore innovazione era costituita dal fatto che l'apparecchio aveva un video ed un registratore a cassette, incorporati, e che veniva fornito, di serie, l'interprete BASIC della Microsoft. Da allora il PET ha subito varie modifiche e "cure di ringiovanimento" (nella foto la versione con tastiera completa) ed è ancora molto richiesto. Una delle caratteristiche della prima versione che più colpì il grosso pubblico fu l'aspetto estetico: la macchina ricordava vagamente la forma di una testa e delle spalle umane. Il chip 6502 è, attualmente, uno dei microprocessori più diffusi tra tutti gli home computer in vendita

Immagini animate

Gli enormi sviluppi nel mondo dei cartoni animati

L'intero processo per la produzione di immagini animate, sia nel cinema che nella televisione, sfrutta il fatto che l'occhio umano, per qualche frazione di secondo, "fissa" le immagini in movimento: basta far scorrere davanti agli occhi una serie di immagini in rapida successione per creare l'illusione del movimento.
I primi tentativi, per produrre artificialmente il movimento di immagini, erano basati sull'impiego di un cilindro ruotante: sulla superficie interna venivano applicati dei disegni che l'osservatore guardava attraverso delle fessure praticate sulla superficie esterna del cilindro. L'effetto non era molto soddisfacente, ma il principio è identico a quello dei moderni film. Questo strumento, chiamato "Zeotropo", esisteva prima ancora dell'invenzione della macchina fotografica. Fu naturale in seguito, sostituire i disegni incollati all'interno del cilindro con fotografie. La successiva fase di sviluppo di questi processi, ossia le sequenze filmate, si avvalse di emulsioni fotografiche relativamente veloci, capaci di imprimere un'immagine sulla pellicola in meno di un sedicesimo di secondo (la velocità alla quale venivano proiettati i primi film cinematografici).

La simulazione del movimento

Sembrerà strano, ma ci volle molto tempo prima che nell'industria cinematografica nascesse l'idea di produrre dei disegni a mano, imprimerli su pellicola, e proiettare poi il risultato: i cosiddetti "cartoni animati". Se pensiamo che per ogni secondo di film ci vogliono 24 disegni (le pellicole moderne, infatti, si proiettano a 24 immagini al secondo) è evidente che anche per soli 5 minuti di pellicola si richiede una notevole mole di lavoro essendo necessari ben 7.200 disegni! Non ci si meravigli, quindi, che il tipo di immagine abbia, in genere, una grafica alquanto formalizzata: la ripetitività è essenziale, affinché un disegno non si discosti troppo dal successivo.
Il computer è indicatissimo per questo tipo di lavoro ripetitivo e preciso ed, in effetti, viene impiegato in questo settore, per regolare la velocità del movimento, per modificare la prospettiva, le luci e le ombre, i passaggi d'ambiente ed i ritmi di presentazione delle immagini, lasciando all'artista la possibilità di concentrarsi maggiormente sulla qualità dell'immagine. In altre parole, l'animazione sta diventando una vera arte grafica: l'artista crea immagini, il computer le anima.
Nella sua forma più semplice il processo di animazione si basa sulla grafica "a sprite" (vedere a pag. 152), per la creazione di "personaggi", che vengono poi trasferiti sullo schermo, e fatti muovere, producendo così il tipo di animazione tipica dei videogiochi. Per creare l'illusione di movimento, per esempio una persona che cammini, occorre scambiare ripetutamente due (o più) sprite. Come abbiamo già visto, la creazione di uno sprite è un lavoro piuttosto laborioso, se si considera che la qualità grafica risultante è, necessariamente, un'immagine bidimensionale molto elementare.
Nella fase successiva dell'operazione, l'ani-

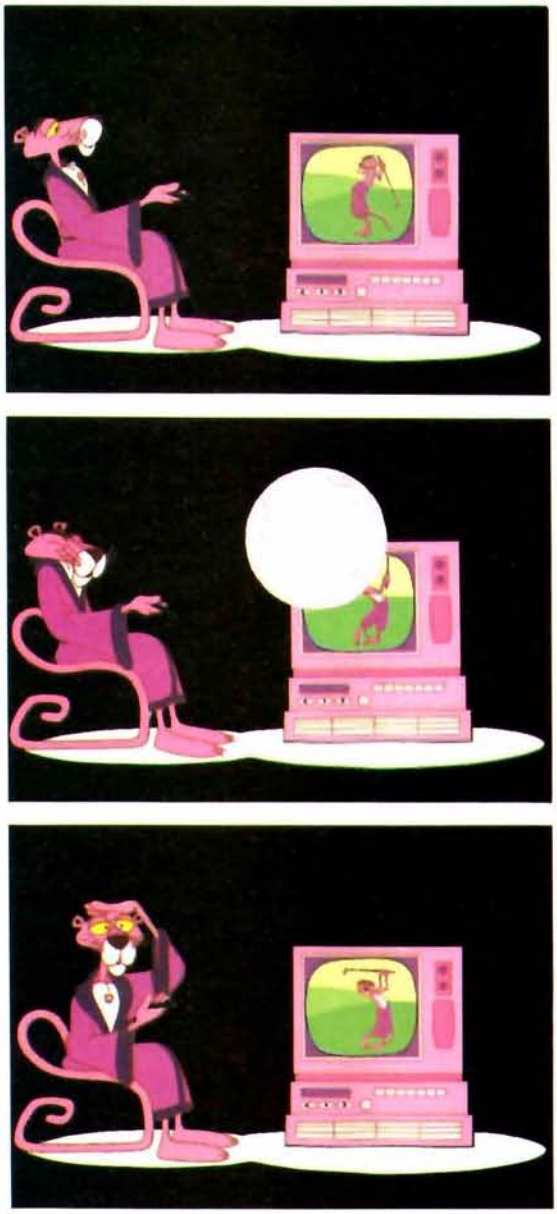


Immagine per immagine
Con il sistema di animazione convenzionale (nella foto tre esempi tratti da un film della "Pantera Rosa") l'artista è costretto a disegnare ogni singola immagine separatamente, o almeno le parti di essa che cambiano posizione o aspetto. Vengono impiegati dei grandi fogli trasparenti in modo che l'immagine completa si ottiene dalla sovrapposizione di più "strati" del disegno. L'artista si dedica alla preparazione delle scene principali della sequenza, mentre tutte quelle intermedie vengono preparate da assistenti chiamati "in-betweeners". L'intera sequenza viene poi fotografata, nel corretto ordine, con una speciale macchina fotografica

per concessione della RICHARD WILLIAMS ANIMATION LTD

matore/programmatore deve costruire un algoritmo per dare all'immagine l'impressione di profondità, secondo le leggi della prospettiva, permettendo di definire gli oggetti sullo schermo in base alle loro coordinate X, Y e Z. Inoltre, è estremamente utile che il computer non riproduca sullo schermo le linee "nascoste", (fino a poco tempo fa, infatti, le immagini erano una fitta rete di linee, ossia una "gabbia" sulla quale costruire, a mano, i vari disegni).

Il successivo passo, verso un miglioramento dell'immagine, è l'arrotondamento delle curve. Un arco circolare si può definire, in geometria, con soli 3 punti: uno posto all'inizio dell'arco, uno all'altro estremo ed il terzo situato nel punto più lontano dalla linea retta che congiunge i primi due. Naturalmente una complessa sequenza di curve va suddivisa in componenti elementari, quindi è molto importante che ci sia un sistema sufficientemente semplice, per indicare al computer che la linea in questione non è una retta, ma una curva.

Un altro fattore importante è la possibilità di creare luci ed ombre. Per prima cosa è necessario specificare la posizione della fonte di luce: in tal modo la parte dell'oggetto disegnato rivolta verso la luce sarà più illuminata, ed un adeguato ombreggiamento progressivo permette di meglio definire l'immagine, dandole una certa profondità.

Se il software è abbastanza sofisticato, è anche possibile definire più di una sorgente di luce e sarà il computer a giostrare i giochi d'ombre e di riflessi.

L'uso del colore è strettamente collegato all'ombreggiamento. Anche il più semplice home computer dispone di 8 o 16 colori diversi, ma i computer progettati appositamente per grafica di tipo professionale permettono almeno 4096 variazioni di colore. Alcuni sono limitati dalla quantità di bit che compongono la "parola" (word) del computer: se questa è di 24 bit, per esempio, il computer può avere a disposizione 16,7 milioni di variazioni. In genere, la definizione dei colori e delle ombre viene riunita in un'unica funzione del programma.

Veniamo adesso al problema della simulazione del movimento. È relativamente facile ridurre il movimento in componenti individuali se lo concepiamo come un problema di geometria solida continua, anche se l'oggetto in questione è complesso quanto lo può essere una mano in movimento. Il fattore determinante è la capacità e le dimensioni del computer impiegato. Inoltre, per produrre un'immagine di alta qualità, occorrono monitor capaci di un'alta risoluzione, generalmente di 1.000 × 1.200 pixel. Se si pensa che per ogni pixel occorre almeno un byte (8 bit) nel quale conservare le informazioni che definiscono il colore e la tonalità, ciò comporta più di un megabyte (un milione di byte) per ogni immagine sullo schermo. Non è possibile, dunque, generare immagini animate di altissima qualità sugli home computer. In effetti i disegnatori di cartoni animati adoperano computer fra i più grandi e potenti del mondo: non per niente chiedono compensi che arrivano fino a 2 milioni e mezzo di lire per ogni secondo di film prodotto!

Se iniziamo con un oggetto semplice, come un cubo, è relativamente facile capire come fare per muoverlo (e magari farlo ruotare allo stesso tempo) sullo schermo. Un cubo è definibile mediante le coordinate dei suoi otto angoli, ma lo stesso principio vale anche per oggetti più complessi. L'unica differenza risiede nella memoria necessaria per conservare tutte le coordinate e nella capacità del computer di manipolare tali informazioni in un tempo abbastanza breve per dare un'illusione di movimento sufficientemente realistica. Anche qui, come in altri campi, esiste uno stretto rapporto tra la qualità e la quantità di memoria e capacità di elaborazione disponibili. Più le immagini sono ricche di particolari, maggiore dovrà essere la disponibilità di memoria di massa.

I normali televisori domestici (con una risoluzione di circa 625 × 1000 pixel) offrono immagini più "granulose", rispetto ai monitor pro-





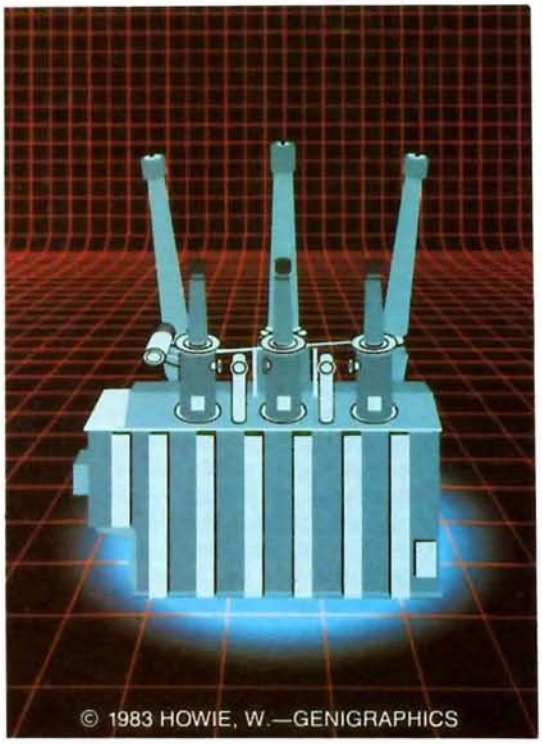


La scatola magica dei colori

Fino a poco tempo fa, la preparazione di sequenze, animate o meno, per la televisione e per il cinema era simile a quella adoperata per la riproduzione di fotografie nelle riviste: il disegno veniva elaborato su carta o su pellicola trasparente per essere poi fotografato. Il sistema "Paint Box" (scatola dei colori) della Quantel, tuttavia, ha eliminato completamente l'uso della carta e il disegno viene composto direttamente su computer e trasferito sul nastro di un videoregistratore

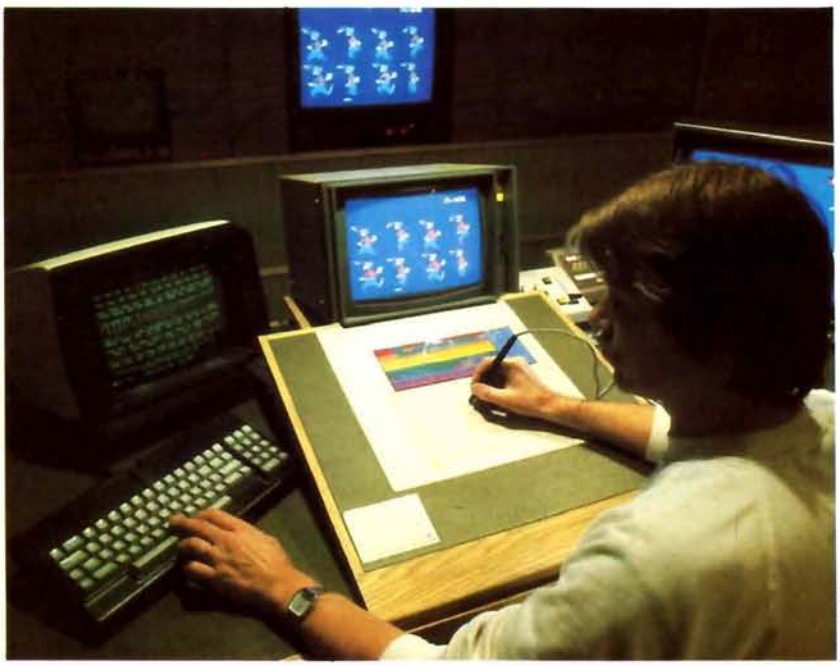
fessionali, ma si può esser certi che qualsiasi disegno eseguito con lo stesso grado di risoluzione, o ancor più dettagliato, risulterà realistico almeno quanto un'immagine televisiva. Per poter creare delle immagini molto dettagliate e aderenti alla realtà è necessario impiegare software molto sofisticato e hardware realizzato appositamente, oppure opportunamente modificato per questo genere di applicazioni.

Nei sistemi più diffusi viene adoperato un apparecchio chiamato "bit-pad digitiser": un grande tavolo da disegno dotato di un fitto reticolo di fili disposti orizzontalmente e verticalmente (a matrice). Lo scopo del reticolo è di captare la posizione di una speciale penna usata dal disegnatore affinché sullo schermo del computer venga riprodotto il disegno tracciato. L'artista è libero di copiare un disegno già esistente, di disegnare a mano libera, oppure usare squadre ed altri strumenti, proprio come se disegnasse su carta. L'immagine viene "digitalizzata" (ossia, il computer ne calcola automaticamente le



coordinate), memorizzata e, contemporaneamente, presentata sullo schermo. L'artista è libero di scegliere quale simbolo adoperare come cursore sullo schermo, proprio come, disegnando con sistemi tradizionali, si può scegliere tra una matita, una penna o un pennello. Analogamente, i colori si selezionano da una "tavolozza" riprodotta su di una zona dello schermo. Se il colore desiderato non è tra quelli indicati si "mescolano" i colori fino ad ottenere la giusta tonalità, come con la tavolozza di colori a olio. La speciale penna può essere usata anche per cancellare o sovrapporre i disegni.

Avendo creato un'immagine, vediamo ora come farla muovere. Si potrebbe ricorrere ad una meccanizzazione del processo convenzionale, usando un programma per archiviare il materiale, per colorare gli scenari e per mostrare alcune possibili sequenze. Il lavoro sarebbe certamente più rapido, ma è possibile ottenere molto di più con una programmazione adeguata. Basandosi sullo stesso principio adottato per l'arrotondamento delle curve, è possibile fabbricare tutta una serie di azioni, limitandosi a definire soltanto la scena iniziale e quella finale. Tale processo (conosciuto col nome di "tweening", letteralmente "inframezzamento") viene svolto, in uno studio di animazione tradizionale, da un assistente chiamato, appunto, "in-betweener", (inframezzatore). Oggi, quasi tutto il lavoro di animazione viene affidato ai computer. Abbiamo già fatto notare che l'introduzione del computer, in questo campo, offre all'artista la possibilità di concentrarsi maggiormente sulla qualità del disegno e sulla creazione di nuove forme ed immagini. I risultati si cominciano ad apprezzare in un numero sempre più crescente di film, nei quali sbalorditivi effetti speciali, impossibili fino ad oggi, si susseguono a ritmo incalzante, mescolando realtà e fantasia.



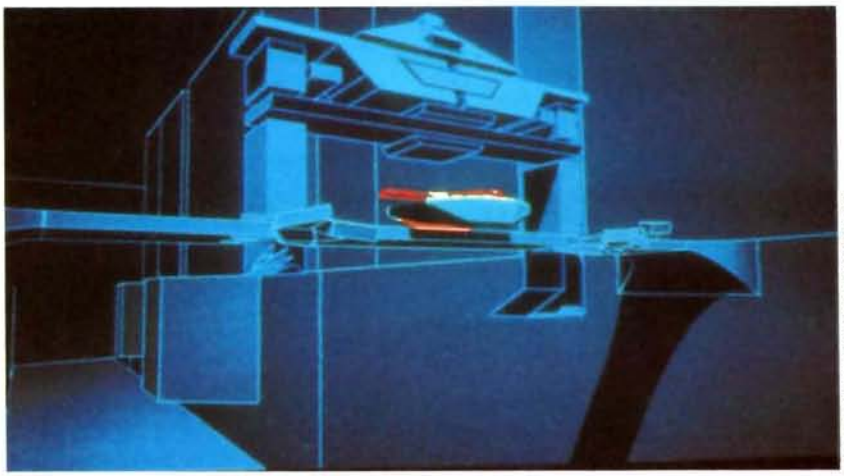
IAN DOBBIE

Vedere per credere

Grazie alla potenza di elaborazione ed alla enorme memoria dei moderni computer, è possibile creare sia su pellicola sia sullo schermo TV immagini identiche a fotografie. Con tecniche di programmazione progettate per risolvere problemi statistici e numerici, è possibile manipolare le immagini create, per renderle realistiche al massimo, riuscendo ad ingannare anche lo spettatore più accorto

TRace ON

Il film TRON, prodotto dalla Walt Disney Productions, è stato il primo a sfruttare il computer per creare la maggior parte delle immagini. Il film è ambientato parte nella realtà e parte in un gigantesco computer. Gli scenari del mondo fantastico sono stati realizzati mediante combinazioni di disegni computerizzati e di fotografie con effetti speciali



per concessione della WALT DISNEY LTD

Interpreti e compilatori

Due diversi sistemi per convertire il BASIC in codice macchina

Nei primi computer, non esisteva la tastiera: le istruzioni di programmazione venivano immesse, una alla volta, selezionando una serie di interruttori posti sul pannello frontale dell'apparecchio. Le particolari combinazioni di acceso/spento degli interruttori rappresentavano il codice binario dell'istruzione o del dato.
Era logico sostituire gli interruttori con tastiere simili a quelle delle macchine da scrivere e le varie combinazioni di interruttori con parole più comprensibili. Infatti, sono stati creati dei linguaggi ad alto livello, quali il BASIC.
Ma, poiché le unità centrali dei computer funzionano ancora con il medesimo codice binario, i programmatori hanno dovuto sviluppare particolari programmi (chiamati, come vedremo, interpreti o compilatori), per convertire i programmi scritti in linguaggi ad alto livello in codici eseguibili dalla macchina.
Nel settore dell'informatica, come del resto in tutti gli altri, un qualsiasi guadagno in potenza o velocità si paga in denaro, in tempo o in libertà d'azione. Così accade con i programmi "interpreti" e "compilatori": se, da un lato, sono validi strumenti di lavoro per il programmatore, dall'altro comportano una serie di vantaggi e svantaggi, che vedremo tra poco.
Gli interpreti sono un sistema abbastanza economico (ecco perché sono presenti nella maggior parte dei piccoli computer) per la conversione di un linguaggio ad alto livello in codice macchina. Inoltre, essendo conservati nelle memorie ROM, occupano poca memoria RAM, lasciandone di più per i programmi.
Nella maggior parte dei computer più economici, l'interprete consente di scrivere un programma in BASIC e di eseguirlo immediatamente digitando un RUN. Tutt'al più, comparirà un messaggio d'errore del tipo:

```
SYNTAX ERROR ON LINE 123
```

se abbiamo commesso qualche errore nella trascrizione del programma.
Si procede allora con un comando LIST e, individuato e corretto l'errore, si riparte. Alcuni BASIC addirittura controllano la sintassi di ogni linea di programma in fase di trascrizione.
Chissà quante volte, oramai, abbiamo eseguito queste operazioni senza mai pensare all'interprete. Eppure, sebbene la sua presenza sia volutamente, per così dire, "invisibile", l'interprete solleva il programmatore da tutta una serie di problemi (dove collocare il programma in memoria, come eseguirlo, ecc.): basta digitare qualche comando per eseguirlo, listarlo o modificarlo.
L'interprete è facile da usare, ma il suo funzionamento non è ottimale: ogni volta che impartiamo il comando RUN, infatti, l'interprete deve trovare il programma che desideriamo far "girare" ed eseguirlo, traducendo un'istruzione alla volta. Poniamo che il programma contenga un ciclo:

```
400 LET N = 0
500 PRINT N
600 LET N = N + 1
700 IF N < 100 THEN GOTO 500
```

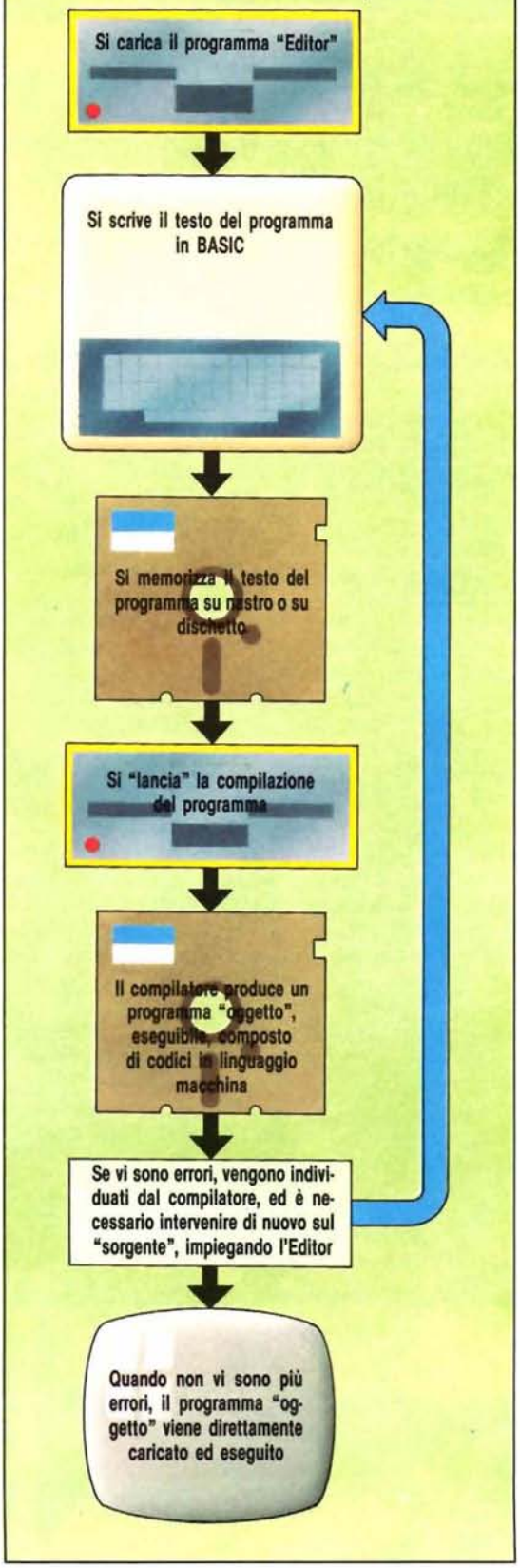
L'interprete deve tradurre ed eseguire le linee 500 e 600 ben cento volte, come se non le avesse mai incontrate prima!
I compilatori, invece, si comportano in modo assai diverso. Hanno lo svantaggio di essere più costosi, di occupare, durante la compilazione, una maggiore quantità di memoria, e di richiedere, in genere, l'impiego di unità a dischi.
Ciò che offrono si può riassumere in elasticità d'uso, potenza di elaborazione e velocità. Le poche linee di programma riportate nell'esempio, vengono tradotte una volta per tutte e quindi eseguite per cento volte, con un notevole risparmio di tempo. Però, c'è un prezzo da pagare. Supponiamo di voler scrivere un programma usando il compilatore: occorre, in primo luogo, usare un particolare programma (chiamato "Editor"), col quale si crea il testo del programma che, in questo caso, viene indicato col termine di "programma sorgente". Per conservare il "sorgente" su dischetto, occorre un nome o codice identificativo. L'Editor, in genere, chiede il nome che vogliamo assegnare al programma ed esso consiste, normalmente, di due parti: il nome da noi scelto (per esempio, MIOPROG) ed un'estensione "fissa" che, nel caso del BASIC, è BAS. Il nome completo è dunque MIOPROG.BAS (con un punto che separa le due parti). Memorizzato il sorgente, occorre impartire un comando del tipo:

```
COMPILA MIOPROG.BAS
```

Questo comando carica in memoria il compilatore BASIC, che elaborerà il testo sorgente, convertendolo in codice eseguibile. Il tempo necessario alla compilazione può variare da alcuni secondi a diversi minuti, dopodiché sul dischetto apparirà un nuovo "file": MIO-

LA COMPILAZIONE DI UN PROGRAMMA

Compilare un programma è più laborioso che non scriverlo impiegando un interprete, ma si ha il vantaggio di produrre programmi incredibilmente più veloci. Ecco, in sintesi, le fasi della compilazione:



PROG.OBJ (l'estensione OBJ indica che il file è un programma "eseguibile"), che non è altro se non la traduzione del sorgente in codice "oggetto", ossia codice macchina.
Durante la compilazione, viene eseguito un controllo sulla correttezza della sintassi e, se si rilevano degli errori, vengono visualizzati sullo schermo dei messaggi del tipo:

```
100 REED X : IF X = 3 (N + 2) LET P = Q
    1                 2       3

FATAL ERROR:—
1) //REED// UNRECOGNISED COMMAND
2) // ( // ILLEGAL OPERATOR HERE
3) ?? 'THEN' OR 'GOTO' EXPECTED HERE
```

Questi errori, in italiano, corrispondono a:

```
1) //REED// COMANDO NON RICONOSCIUTO
2) // ( // OPERATORE NON VALIDO QUI
3) ?? MANCA 'THEN' O 'GOTO'
```

Messaggi simili appaiono per ogni linea che contenga almeno un errore, e la spiegazione è senz'altro più esauriente che non in un interprete BASIC, ma a questo punto occorre ripartire dall'Editor, correggere gli errori, e ricompilare il programma. Se non emergono più errori, si lancia il programma con un comando del tipo:

```
RUN MIOPROG
```

Il programma non contiene più alcun errore di sintassi, ma può essere necessario apportare ancora delle modifiche, affinché funzioni come noi vogliamo. In questo caso, si riparte dall'Editor, si ricompila... e così via.
I pregi di un compilatore non sono particolarmente evidenti in fase di stesura del programma (nonostante la migliore diagnostica degli errori), ma soprattutto in fase di esecuzione dei programmi, che risultano spesso fino a 50 volte più veloci di quelli fatti "girare" mediante un interprete.
Limitare il confronto tra compilatori ed interpreti ai pochi esempi qui riportati è commettere un'ingiustizia nei confronti dei primi, che spesso sono indispensabili, particolarmente nelle applicazioni commerciali.
Esistono compilatori per diversi linguaggi (tra i quali il COBOL, nato esclusivamente sotto forma di compilatore).
I compilatori, infine, hanno altri due apprezzabili vantaggi. Uno consiste nel fatto che il programma sorgente può essere completato da numerosi ed ampi commenti, senza che la lunghezza del programma eseguibile (programma oggetto) ne risenta minimamente: tutti i commenti vengono "filtrati" dal compilatore.
L'altro vantaggio è particolarmente apprezzato da chi commercializza il proprio software: i programmi oggetto sono codificati in linguaggio macchina, dal quale è impossibile risalire alle istruzioni BASIC contenute nel sorgente, e ciò offre un certo grado di protezione contro... gli scopiazzatori!



Azione lenta

Nei giochi d'avventura simulati su computer, la velocità non è un fattore critico: per tali programmi non sono richiesti molti calcoli e possono anche essere scritti comodamente in BASIC interprete



Azione più veloce

Molti programmi per applicazioni commerciali (particolarmente le "lavagne elettroniche"), non possono essere scritti in linguaggio macchina, per la gran mole di calcoli necessari, né in BASIC interprete, per la tipica lentezza di calcolo. Si ricorre quindi al BASIC compilato



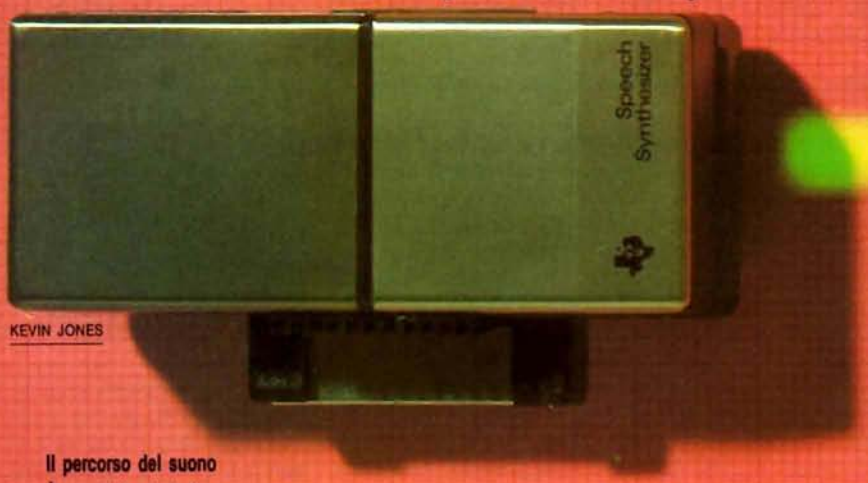
Azione velocissima

Per programmi che richiedono una grande velocità d'azione (ad esempio i giochi con largo impiego di grafica), i compilatori non sono abbastanza veloci, e deve quindi essere impiegato il linguaggio macchina, cosa non facile e laboriosa

Parola al... computer

I computer parlanti non sono più fantascienza: con un sintetizzatore di voce, anche i piccoli computer possono dialogare con l'uomo

Sebbene la scienza per il riconoscimento della parola sia ancora in via di sviluppo, la riproduzione della voce, mediante apparecchi elettronici, è già in fase avanzata. Fino a pochi anni fa, per la generazione di suoni che assomigliassero alla voce umana occorrevano computer molto potenti, dotati di una grande memoria. Oggi, con l'impiego di speciali schede accessorie, dal costo relativamente basso, anche i piccoli home computer ed i videogiochi sono in grado di "parlare".
La voce umana è composta da tre tipi di suoni completamente diversi fra loro. Il primo è il suono tipico delle vocali, prodotto dalla vibrazione delle corde vocali: la frequenza della vibrazione ma anche l'apertura della bocca e la posizione della lingua determinano quale vocale viene pronunciata. Il secondo è il suono sibilato (come in ss, sh e ff); in questo caso l'aria proveniente dai polmoni passa inostacolata attraverso le corde vocali e la frequenza del suono è determinata dalla posizione delle labbra e della lingua. Il terzo suono (se si può chiamarlo tale) è quello del silenzio o, per essere più precisi, le pause interne alle parole: non ce ne accorgiamo, ma, nel parlare, è impossibile passare con continuità da un particolare suono ad un altro completamente diverso.

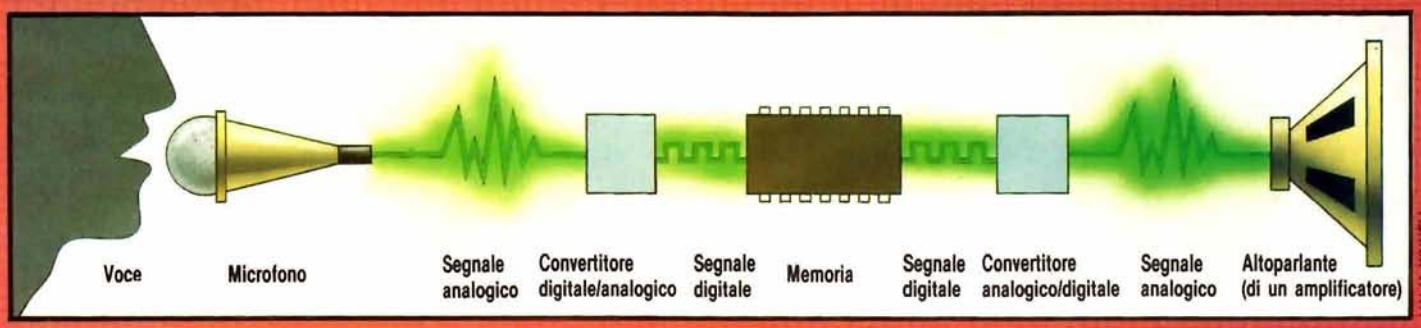


KEVIN JONES

Il percorso del suono

È possibile digitalizzare e conservare i suoni prodotti dalla voce nella memoria RAM o ROM. L'output elettrico proveniente dal microfono passa attraverso un convertitore analogico/digitale. L'output di questo convertitore è esclusivamente composto da numeri binari. La parola viene ricomposta usando un convertitore digitale/analogico, un amplificatore e un altoparlante

Esistono due modi per sintetizzare elettronicamente la voce umana. Il primo metodo, molto usato fino a questi ultimi anni, è basato sulla sintetizzazione "per regole" (synthesis "by rule"): analizzando le varie frequenze con le quali sono formate le parole, è possibile elaborare una serie di regole che ci permettono di ricreare qualsiasi parola, semplicemente ricomponendo in modo opportuno i vari suoni di base. Tanto per citare un esempio, la parola "due" può essere definita come una mescolanza di frequenze atte a riprodurre il suono della "d", per una durata di tot secondi, immediatamente seguita da altre due "miscele" per i suoni "u" ed "e".
Il termine scientifico, per indicare queste miscele di frequenze, è "fonema": qualsiasi parola è costituita da una particolare combinazione di fonemi. Quando si adoperano strumenti di sintetizzazione elettronica, le parole tendono a perdere le caratteristiche individuali della voce umana, sebbene esse rimangano riconoscibili e comprensibili. Siccome le regole per la generazione di fonemi sono memorizzate all'interno della macchina, l'utente non deve fare altro che fornire al sistema un elenco di fonemi, che vengono poi riprodotti attraverso un piccolo altoparlante. Con un po' di pratica si possono generare all'istante frasi complete, semplicemente richiamando le opportune sequenze di fonemi, conservate, nel caso del linguaggio BASIC, in variabili alfanumeriche (o "stringhe").
Il secondo metodo di sintetizzazione della parola si affida alla capacità, sia dell'orecchio che del cervello umano, di compensare, inconsciamente, gli elementi mancanti. La gamma delle frequenze trasmesse per telefono, per esempio, hanno un quinto della qualità che si pretende da un buon complesso stereo: nonostante ciò la voce del nostro interlocutore telefonico ci risulta comprensibile: è il nostro cervello che riempie automaticamente i vuoti di frequenza.
Questo secondo metodo, che si chiama "sintetizzazione digitalizzata della parola" è basato sullo stesso fenomeno. Con l'enorme calo



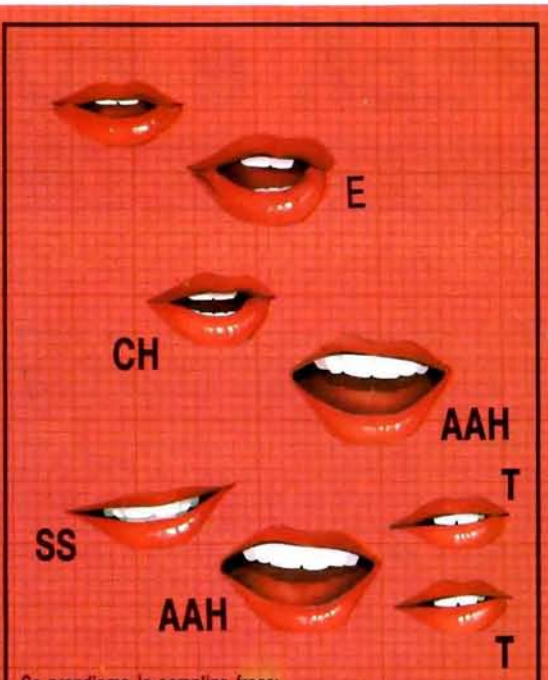
IAN McKINNELL

nel prezzo delle memorie elettroniche, è oggi possibile, mediante appositi convertitori analogico/digitali, memorizzare le parole sotto forma di dati espressi in numeri binari. I dati così raccolti vengono elaborati, riducendone (con metodi statistici) le componenti, e conservati in una memoria ROM. Sarà il nostro udito (o per meglio dire il nostro cervello) a colmare automaticamente le lacune di frequenza risultanti da questo processo.

Per poter riascoltare le parole conservate, basterà indicare al computer l'indirizzo di quella parola nella ROM e il dato digitalizzato viene estratto e riconvertito in suoni. Poiché in questo processo viene impiegata la voce di una particolare persona, la maggior parte delle inflessioni originali, del tutto individuali, rimangono intatte. Per esempio, il chip "parlante" prodotto dalla Acorn per il microcomputer BBC è stato prodotto usando la voce di un lettore di telegiornali inglese, Kenneth Kendall, e si riconosce immediatamente che la voce risultante è la sua.

Ci sono dei computer (in particolar modo il Sirius 1) che, grazie all'hardware incorporato e ad un software opportuno (fornito su dischetto), permettono all'utente di digitalizzare la propria voce impiegando un microfono. I dati immessi vengono conservati su disco (ogni secondo di parola parlata occupa circa 1 Kbyte), pronti ad esser richiamati da un qualsiasi programma applicativo, che preveda l'uso di messaggi o avvertimenti vocali.
I campi nei quali la sintetizzazione della parola può essere applicata con notevole profitto sono talmente numerosi, che è impossibile elencarli tutti. Tanto per dare un'idea: gli annunci nelle stazioni ferroviarie e negli aereoporti, apparecchi per i non vedenti, ecc. Sulla rete telefonica americana la sintetizzazione della voce è già diffusa per avvertire, per esempio, che il numero chiamato è occupato, oppure che il numero chiamato è cambiato. Molte ditte di vendita per corrispondenza si stanno equipaggiando con speciali segreterie basate sull'impiego della sintetizzazione vocale. In altri casi, il codice di un articolo ordinato viene digitato al computer il quale fornisce, vocalmente, la descrizione dell'articolo per conferma. Il computer è inoltre in grado di dare informazioni sul livello delle scorte o se si prevede un ritardo nella consegna, dando modo di modificare, di conseguenza, l'ordine.

Nell'ambito degli home computer e dei videogiochi, il sintetizzatore rende più vivi e divertenti i giochi: avvisa il giocatore del conteggio e segnala verbalmente un prossimo attacco nemico, lasciandolo libero di concentrarsi sulla tattica del gioco senza il bisogno di consultare eventuali messaggi scritti.
Esistono anche applicazioni in campo didattico: lo *Speak'n'Spell* (prodotto dalla Texas Instruments) pronuncia una serie di parole, che il bambino deve compitare sulla tastiera; se sbaglia, è invitato a ripetere l'operazione, altrimenti viene incoraggiato a riprovare con un'altra parola. Esistono anche "interpreti" parlanti in lingue straniere, che traducono e pronunciano le parole impostate alla tastiera.



Se prendiamo la semplice frase:

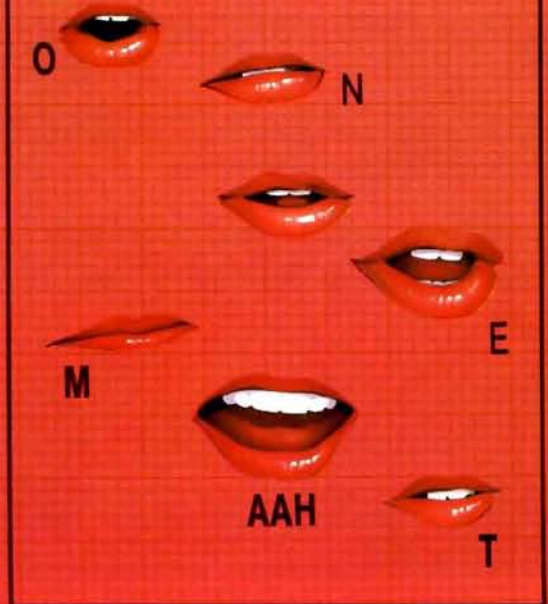
Scacco matto

è possibile scomporla in una serie di fonemi

SS / CH / A / CH / CH / O M / AAH / T / T / O

Ogni tipo di chip ha esigenze diverse e pertanto le frasi devono essere spezzate e specificate diversamente secondo quale chip viene impiegato.

Il chip prodotto dalla Votrax, per esempio, contiene circa 60 fonemi accessibili direttamente in base ad un semplice numero. Per rendere più pratico il sistema, viene fornito un gruppo di programmi che permettono di digitare la parola o frase desiderata sotto forma di fonemi dei quali essa è composta. Tuttavia, esistono strumenti, quali il Braid Speech Synthesiser o il Personal Speech System della Votrax, nei quali è possibile digitare sulla tastiera la frase (in lingua inglese) per sentirla pronunciare dal computer



TONY LODGE

PEEK e POKE

L'anello di congiunzione tra linguaggio macchina e BASIC

PEEK e POKE sono due istruzioni del BASIC molto usate da chi si dedica alla manipolazione dei bit e dei byte nella memoria, particolarmente nel campo della grafica. La PEEK serve per esaminare il contenuto di una locazione di memoria, mentre la POKE si impiega per scrivere un valore, compreso tra 0 e 255, in una specifica locazione di memoria.
Mediante queste due istruzioni, è possibile esplorare zone di memoria altrimenti inaccessibili con il BASIC, che solo a livello interno provvede a leggere e scrivere dati nella memoria. Sebbene nelle normali applicazioni non sia necessario conoscere l'ubicazione di stringhe, caratteri o di altri valori, può essere utile, in certe occasioni, esaminare il contenuto della memoria. Il breve programma che segue serve proprio a questo:

```
10 REM ESAME DELLE LOCAZIONI DI MEMORIA
20 PRINT "NUMERO DELLA LOCAZIONE DI MEMORIA IN DECI
   MALE"
30 INPUT M
40 P = PEEK(M)
50 PRINT "IL CONTENUTO DELLA LOCAZIONE ";M;" È "; P
60 GOTO 20
70 END
```

Specificando l'indirizzo della locazione che intendiamo esaminare, il programma ci fornisce il valore ivi contenuto. Vengono impiegati, nella visualizzazione, valori decimali, anche se, ovviamente, in realtà il computer lavora in sistema binario. Se desideriamo visualizzare il contenuto della memoria sotto forma di caratteri stampabili, abbiamo a disposizione la comoda funzione CHR$, e basta modificare la linea 50 in:

```
50 PRINT "IL CONTENUTO DELLA LOCAZIONE"; M; " È ";
   CHR$(P)
```

Per visualizzare il contenuto dell'intera memoria RAM, può essere usato un ciclo FOR-NEXT: si cancelli la linea 30, si modifichi la linea 20 in: FOR X = 0 TO 65535, e si cambi la 60 in: NEXT X.
I caratteri scorreranno molto velocemente sullo schermo, cosicché è opportuno introdurre un piccolo ciclo di ritardo, subito dopo la PRINT e prima della NEXT X. Si noti, inoltre, che il limite superiore del ciclo va bene per una memoria di 64 Kbyte: per 16 Kbyte il valore è 16383 (decimale), per 32 Kbyte è 32767 e per 48 Kbyte è 49151. Il nuovo programma diventa:

```
10 REM ESAME DI TUTTE LE LOCAZIONI DI MEMORIA
20 FOR X = 0 TO 65535
30 LET Y = PEEK(X)
40 PRINT "LOCAZIONE"; X; " = "; Y; " = ";
50 PRINT CHR$(Y)
60 FOR D = 1 TO 200
70 NEXT D
80 NEXT X
90 END
```

La funzione CHR$ converte valori decimali negli equivalenti caratteri, ma quelli che tra essi sono stampabili, hanno valori compresi tra 32 e 127. Nella maggior parte dei computer, i codici da 128 a 255 (il massimo valore rappresentabile con un solo byte) sono riservati a caratteri grafici. Inoltre, alcuni dei valori tra 0 e 31 costituiscono dei caratteri di controllo (ritorno carrello, pulizia dello schermo, ecc.). La loro visualizzazione può sortire effetti piuttosto strani: salti del cursore, azionamento di segnali sonori ed altri.
Con l'istruzione POKE, invece, si possono scrivere valori (tra 0 e 255) nella memoria ed essa va impiegata con grande cautela: se modifichiamo i valori di certe parti riservate di memoria, potremmo compromettere il funzionamento del BASIC e dover ricorrere al tasto di RESET (o spegnere il computer) per ripartire da capo. Conviene controllare attentamente sul manuale dell'apparecchio quali zone di memoria possiamo liberamente modificare (area "utente", come viene spesso indicata). In molti computer, la memoria "video" (quella che serve per depositare i caratteri da visualizzare) è accessibile all'utente ed in essa è possibile definire la forma di nuovi caratteri o immagini grafiche.

USO DELLA POKE

L'istruzione POKE va usata con molta attenzione, poiché con essa si modificano i valori contenuti nella memoria del computer e ciò potrebbe compromettere il funzionamento del programma o dello stesso BASIC, anche se ciò non significa che venga fatto alcun effettivo danno.
Ecco alcune POKE "sicure":

per l'Atari 400 e 800, depositando il valore 1 nella locazione 751 si "spegne" il cursore: POKE 751,1;

sul Commodore 64, si provi POKE 1024,1 (1024 è la locazione della prima memoria video);

sullo Spectrum Sinclair si provi:

```
100 FOR N = 0 TO 6 STEP 2
110 POKE USR"A" + N,BIN01010101
120 POKE USR"A" + N + 1,BIN10101010
130 NEXT N
140 PRINT "AAAAAAAA"
```

Le "A" nell'ultima linea devono essere digitate usando il modo grafico. Questo programma produce una fila di piccoli pezzi da scacchiera, ma può anche causare qualche interferenza "spuria" nell'immagine, dovuta alle frequenze in gioco

Il TI99/4A

Anatomia di un microcomputer a diffusione europea: tecnologie a confronto

Per quanto concerne il progetto e la realizzazione il TI99/4A, della Texas Instruments, è uno degli home computer più professionali. Il suo prezzo, inizialmente piuttosto alto, è calato enormemente negli ultimi tempi per via della forte concorrenza. Questo computer impiega un microprocessore a 16 bit (il TMS9900), progettato dalla stessa Texas Instruments, una casa all'avanguardia nella produzione di semiconduttori, calcolatrici e microcomputer. Il TMS9900 è stato uno dei primi microprocessori a 16 bit, ma non ha riscosso un grandissimo successo.

Il TI99/4A ha una tastiera composta da 48 tasti ed è comoda nell'uso. Sulla parte destra dell'apparecchio c'è una specie di nicchia, per alloggiare le cartucce dei programmi, chiamate dalla Texas "software allo stato solido", con evidente riferimento al fatto che le cartucce contengono vari circuiti elettronici allo "stato solido", ossia integrati. Un connettore, dall'altro lato dell'apparecchio, permette di collegare alcune periferiche, tra le quali: le unità a dischetti, le espansioni di memoria ed un'interfaccia seriale RS232. Il collegamento non è diretto, bensì è necessario l'uso di un particolare circuito esterno di espansione.

Si possono visualizzare fino a 16 diversi colori ad alta risoluzione ed è anche disponibile un generatore di effetti sonori, in grado di produrre contemporaneamente tre "voci" distinte. Tuttavia, la scarsezza di buona documentazione in merito alla programmazione in linguaggio macchina, ostacola il pieno sfruttamento della grafica e degli effetti sonori.

Ogni periferica può essere acquistata separatamente e, a differenza di ciò che si riscontra con altri computer, non esistono periferiche prodotte da altre ditte.

Il TI99/4A è stato studiato particolarmente per i principianti ed è dotato di un buon linguaggio BASIC, con la possibilità di ag-



CHRIS STEVENS

La qualità della tastiera è decisamente superiore a quella della maggior parte degli home computer, benché qualcuno trovi che il suo "tocco" sia un poco rigido per chi sia abituato ad una macchina da scrivere. I tasti non sono particolarmente numerosi, ma si ottengono funzioni aggiuntive mediante l'uso del tasto CTRL, e inoltre, il tasto FCTN permette di usare i tasti della fila superiore per funzioni definite dall'utente

giungere anche il LOGO. In America, questo computer è tra i più venduti, assieme all'Apple II, nel settore didattico.
All'atto dell'accensione, all'utente viene presentato un elenco di opzioni e, se è stata inserita una cartuccia di programmi, anche questa scelta viene inclusa tra le altre. La versione di BASIC fornita come standard è piuttosto limitata, ma è disponibile una cartuccia di espansione che aggiungendo sofisticati comandi di visualizzazione (vedere a pag. 52), la gestione della grafica "a sprite" e quella del sintetizzatore vocale, ne porta il livello a punte superiori al BASIC della Microsoft. Il sintetizzatore vocale è un accessorio molto divertente, ma per il suo funzionamento è necessario il BASIC "esteso", oppure uno speciale programma chiamato "Speech Editor".
Le periferiche disponibili, anche se care rispetto al prezzo del computer, sono numerose e molto adatte, per la loro robustezza, anche all'impiego da parte di bambini.

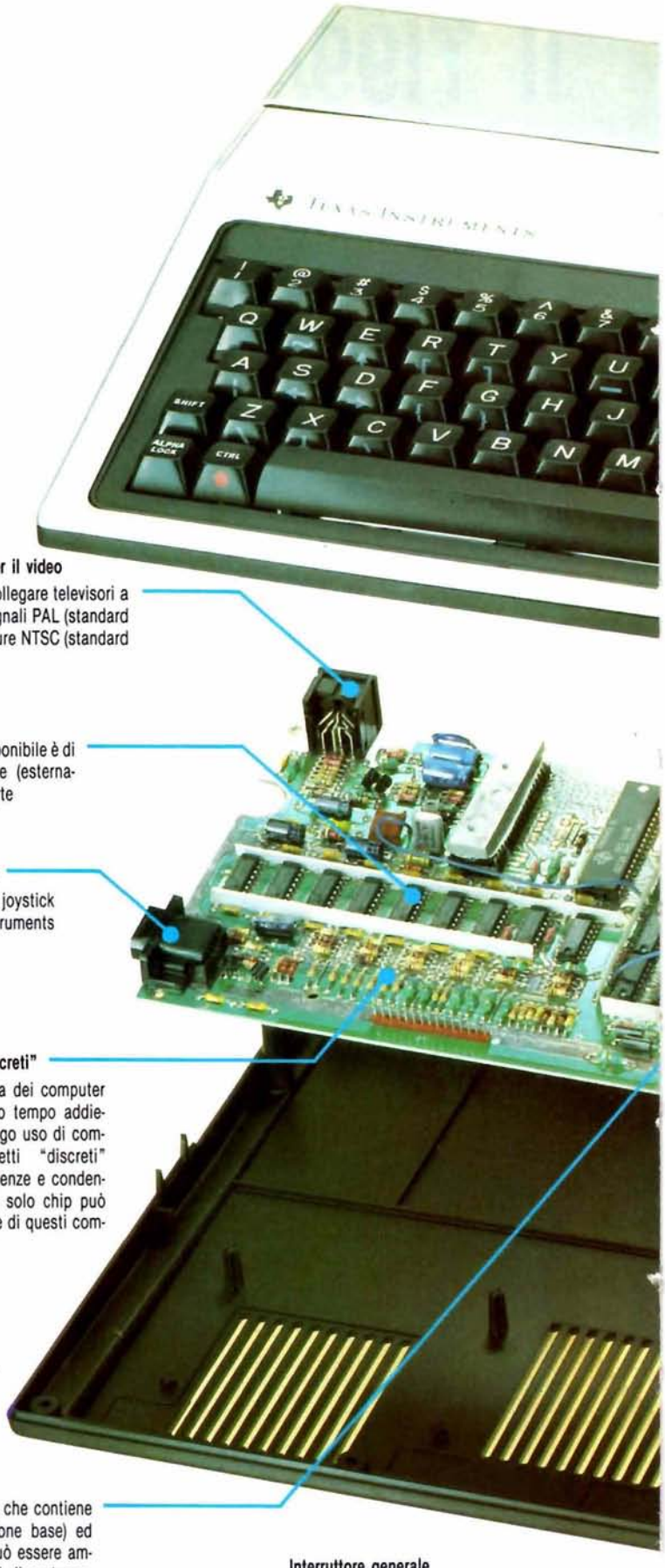


Il joystick
La Texas Instruments ha messo in vendita una coppia di joystick, collegati ad un unico connettore da inserire nel computer. In ciascun joystick ci sono quattro interruttori, il cui azionamento è comandato da una cloche



"Box" di espansione per periferiche
In questo contenitore vengono alloggiate le interfacce per le periferiche, l'alimentatore e l'unità a dischetti. C'è spazio per contenere fino ad otto diverse schede, ciascuna fornita della propria spia di accensione. Collegando una scheda per l'espansione di memoria, si possono avere fino a 52 Kbyte di RAM. Sono inoltre disponibili dei moduli che permettono l'uso di stampanti e di modem, impiegando la nota interfaccia seriale RS232

CHRIS STEVENS



Connettori per il video
È possibile collegare televisori a colori con segnali PAL (standard europeo) oppure NTSC (standard USA)

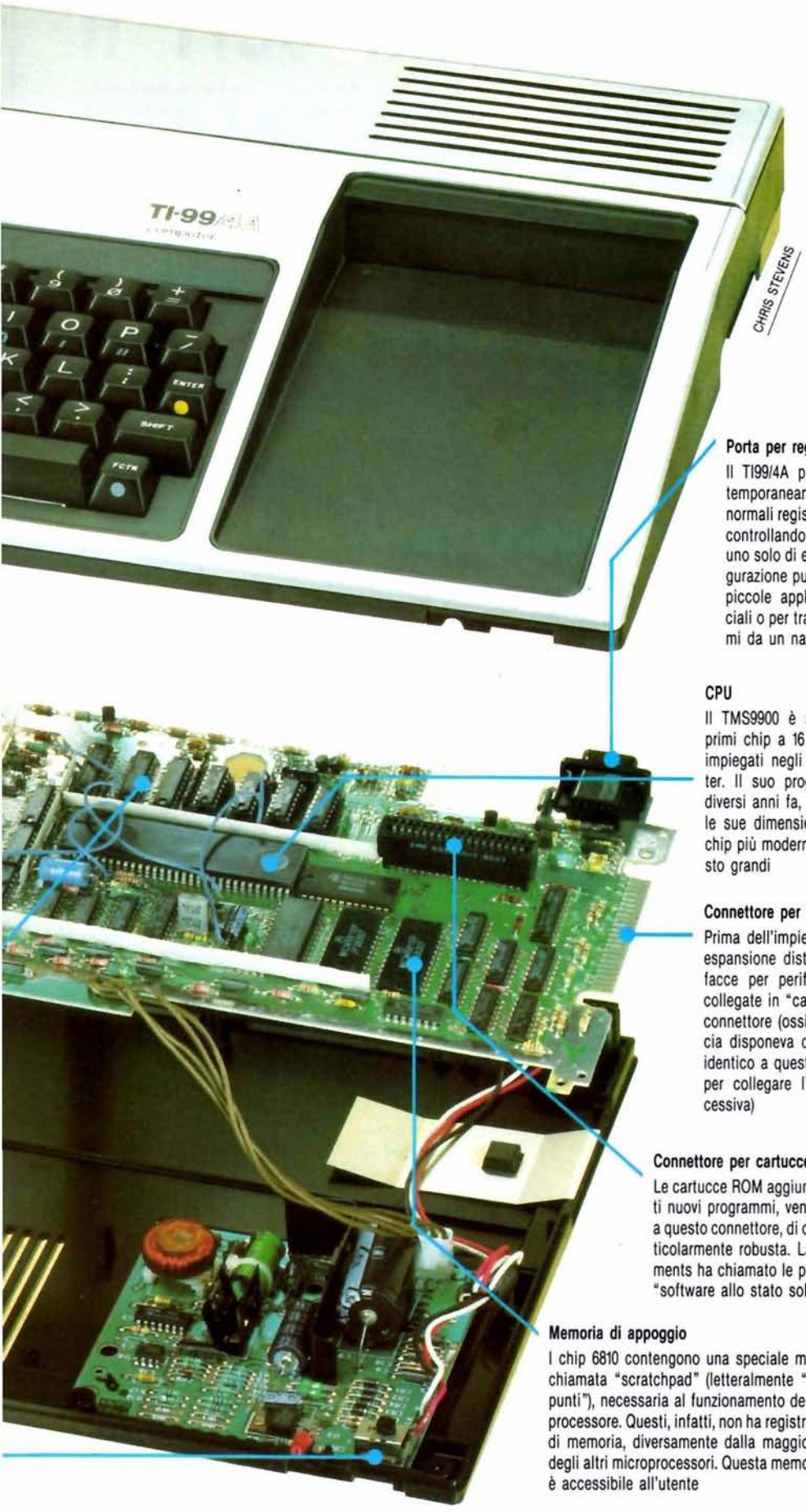
RAM
La memoria RAM disponibile è di 16 Kbyte, espandibile (esternamente) fino a 52 Kbyte

Connettore per joystick
Qui vengono collegati i joystick venduti dalla Texas Instruments

Componenti "discreti"
Una caratteristica dei computer progettati diverso tempo addietro è l'ancora largo uso di componenti cosiddetti "discreti" (transistor, resistenze e condensatori). Oggi, un solo chip può sostituire dozzine di questi componenti

ROM
La ROM esistente, che contiene il BASIC (in versione base) ed altri programmi, può essere ampliata con vari tipi di cartucce, tra le quali una contiene una versione "estesa" del BASIC

Interruttore generale
L'interruttore ha un LED incorporato, che funge da spia



Porta per registratore

Il TI99/4A può lavorare, contemporaneamente, con due normali registratori a cassette, controllando, però, il motore di uno solo di essi. Questa configurazione può essere utile per piccole applicazioni commerciali o per trasferire i programmi da un nastro all'altro

CPU

Il TMS9900 è stato uno dei primi chip a 16 bit ad essere impiegati negli home computer. Il suo progetto risale a diversi anni fa, ragion per cui le sue dimensioni, rispetto a chip più moderni, sono piuttosto grandi

Connettore per periferiche

Prima dell'impiego del "box" di espansione distaccato, le interfacce per periferiche venivano collegate in "cascata" a questo connettore (ossia, ogni interfaccia disponeva di un connettore identico a questo, ma in uscita, per collegare l'interfaccia successiva)

Connettore per cartucce ROM

Le cartucce ROM aggiuntive, contenenti nuovi programmi, vengono collegate a questo connettore, di costruzione particolarmente robusta. La Texas Instruments ha chiamato le proprie cartucce "software allo stato solido"

Memoria di appoggio

I chip 6810 contengono una speciale memoria, chiamata "scratchpad" (letteralmente "per appunti"), necessaria al funzionamento del microprocessore. Questi, infatti, non ha registri interni di memoria, diversamente dalla maggior parte degli altri microprocessori. Questa memoria non è accessibile all'utente

TI99/4A

DIMENSIONI

mm. 380 x 260 x 70

PESO

1800 grammi

CPU

TSM9900 (Texas Instruments)

CLOCK

1 MHz

MEMORIA

16 Kbyte di RAM, estendibile (esternamente) a 52 Kbyte. 26 Kbyte di ROM ed 8 Kbyte di RAM per la grafica. Inoltre 256 byte di memoria speciale sono riservate al funzionamento del microprocessore e non sono accessibili all'utente

VIDEO

24 linee di 32 caratteri, sono disponibili 16 colori. Non esiste una vera e propria grafica nella versione base, ma è possibile definire dei caratteri speciali usando una matrice di 8 x 8 punti

INTERFACCE

Connettore per monitor, per registratore a cassette, per joystick, per cartucce e per periferiche esterne

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

Extended BASIC, TI LOGO, UCSD PASCAL, TI FORTH e Assembler

ACCESSORI COMPRESI

Alimentatore, adattatore per TV, cavo per registratore a cassette, un manuale

TASTIERA

48 tasti, tipo "macchina da scrivere", i tasti numerici possono servire come tasti per funzioni speciali

DOCUMENTAZIONE

Il manuale comprende una descrizione generale dell'apparecchio e delle connessioni necessarie al funzionamento. Una parte, dedicata al BASIC, tratta per esteso i comandi disponibili, fornendo anche diversi esempi sul loro impiego. In fondo al manuale è riportato un conciso glossario sul BASIC

I pirati del software

I danni causati dalla copia abusiva dei programmi



TONY SLEEP



Cento a uno

Così come è illegale copiare abusivamente le musicassette, anche la copia dei programmi di software è un reato. Sfortunatamente per i produttori di software, tale attività non è solamente difficile da prevenire, ma anche da punire. Alcune software house sostengono che per ogni copia di un loro programma, venduta legalmente, ne esistono almeno un centinaio abusive

Duplicazione ad alta velocità

È possibile copiare le cassette di software con lo stesso sistema impiegato per riprodurre le cassette musicali: per mezzo di una copiatrice ad alta velocità. L'apparecchio consiste di una piastra "principale", dove inserire l'originale, e di un certo numero di piastre "ausiliarie", dove vengono registrate, simultaneamente, le copie. Bastano pochi secondi per registrare i due lati di una cassetta. I dischetti, invece, vengono spesso copiati, uno alla volta, usando normali unità a dischi

Un fenomeno che si va sempre più diffondendo è quello della "pirateria": inizialmente comune nel settore discografico, sta prendendo piede anche nel campo del software. La pirateria consiste nel duplicare abusivamente il contenuto di un disco (sia esso di musica leggera o di programmi per computer): tale pratica elimina sia l'acquisto del disco sia la corresponsione dei relativi diritti d'autore. Nel settore informatico, al livello più basso è commesso ogni volta che qualcuno riproduce un programma imprestatogli da un amico. A poco serve il fatto che alcuni programmi (in particolar modo quelli scritti in codice macchina) non possono essere memorizzati usando i normali comandi BASIC, poiché è possibile collegare insieme due registratori a cassette e travasare il programma da uno all'altro, perfino senza l'intervento del computer. Alcune ditte produttrici di videogiochi sostengono che per ogni copia di gioco venduta legalmente, ne esistono 100 riprodotte illegalmente. I novelli "pirati" si difendono affermando che le "software house" possono ben permettersi le perdite finanziare dovute a queste copie non autorizzate, ma certamente gli autori dei programmi, che si guadagnano da vivere con i proventi dei diritti d'autore, non trovano molto giusta questa affermazione.

Ci sono state molte polemiche a proposito di commercianti che offrono del software in leasing, a noleggio oppure con il sistema di "prova prima di acquistare" perché facilitano la copia abusiva dei programmi. Ci sono addirittura dei commercianti che, pur d'incrementare le vendite, regalano copie illegali dei giochi di maggior successo. Anche certi distributori, dai pochi scrupoli, riproducono i programmi in gran quantità, per rivenderli ad altri commercianti. Questo tipo di pirateria comporta pochi rischi, specialmente se il fornitore originale è di un'altra nazione.
Al di sopra di questo livello, la riproduzione illegale diviene molto più raffinata e più difficile da scoprire. Per esempio, c'è chi prende un programma già esistente, vi apporta alcune modifiche e poi lo vende come se si trattasse di una sua invenzione. Talvolta accade che la nuova "versione" offra dei vantaggi o nuove funzioni rispetto all'origi-

nale, ma più spesso, per camuffare l'origine del programma, ci si limita a cambiare solamente gli avvisi di copyright, oppure il modo di presentare le informazioni sullo schermo. Questo discorso comunque vale più per programmi commerciali che non per giochi.
È difficile stabilire se quest'ultimo processo di "modifica" si possa considerare "una pirateria" alla stessa stregua della riproduzione: proprio per questo motivo molte persone riescono a farla franca. Le software house (e gli autori di software in genere), non godono di alcuna protezione da parte della legge: infatti, le leggi vigenti sui diritti d'autore non perseguono la trascrizione dei programmi da un nastro ad un altro, né gli altri tipi di contraffazione di cui abbiamo parlato. Il copyright si riferisce, a quanto pare, esclusivamente alla parola scritta (con alcune eccezioni per quanto riguarda la musica), pertanto programmi scritti per essere conservati nella ROM o su cassette non appartengono a questa categoria.
Le cose si confondono ancora di più quando una ditta produttrice di software copia un'idea da un programma di successo, creandone una propria versione. In questo caso, infatti, non viene copiato il codice del programma ma, piuttosto, lo svolgimento generale di esso o il modo di interagire con l'utente. Il nuovo programma viene scritto partendo da zero, ma per avere lo stesso risultato. Il caso più clamoroso, nel campo dei giochi, è quello di *Pacman*: il gioco è apparso per la prima volta nelle sale giochi. L'Atari ne fece una versione anche per gli home computer Atari, ma ben presto esso è comparso, in numerosissime versioni, prodotte da una miriade di altre ditte. Ogni versione è leggermente diversa dalle altre, ma in ogni caso si basa sempre su di un personaggio all'interno di un labirinto. Nel giro di pochi mesi la Atari è riuscita ad eliminare tutta la concorrenza sleale ricorrendo ad azioni legali oppure, nel caso di ditte più piccole, con minacce di procedere per vie legali.
In generale, chi si guadagna da vivere scrivendo o producendo software è costretto a trovare mezzi tecnici sempre più efficaci (e possibilmente poco costosi) per tutelare i propri diritti d'autore e i propri programmi. Alcuni sostengono che se i loro prodotti vengono venduti ad un prezzo ragionevole, viene meno, in gran parte del pubblico, l'incentivo a copiarli. Per quanto riguarda i programmi più sofisticati, un buon manuale operativo, una elegante confezione ed un servizio di assistenza, continuativo nel tempo, offrono un certo grado di protezione.
Il sistema basato sulla "registrazione" degli utenti garantisce una certa tutela nel caso di software commerciale ad alto livello: l'utente deve riempire un tagliando di registrazione, contenuto nel manuale, per poter usufruire di assistenza o di un "pronto intervento" telefonico in caso di cattivo funzionamento o di aggiornamenti sui programmi.
Un metodo un po' più drastico consiste nel fornire all'acquirente una piccola "scatola nera", detta "dongle" (ovviamente sigillata), da inserire in uno dei connettori per periferiche del computer: senza questo accessorio il programma non funziona. In sostanza, la "scatola nera", grande quanto una scatoletta di fiammiferi, non contiene che un circuito elettronico, spesso nient'altro che una memoria ROM opportunamente codificata. Ad intervalli regolari il programma interroga lo speciale circuito per ottenere il codice: se non lo riceve (o il codice non è esatto) il programma si blocca. A volte il codice è diverso per ogni copia del programma e in tal caso il dongle deve essere venduto in coppia con il "suo" programma. L'unico modo di copiare illegalmente il programma è di contraffare il circuito, oppure di individuare ed eliminare del tutto la parte del programma che controlla il codice. Questa non è un'impresa impossibile, ma non è certo alla portata di tutti.
Sono state fatte molte ricerche per trovare metodi che garantiscano un alto grado di protezione contro la pirateria, senza l'impiego di hardware addizionale. Un sistema (chiamato "watermark", letteralmente "filigrana") è basato sull'inserimento di un codice magnetico (sulla cassetta o sul disco) situato o prima o dopo la registrazione del programma. Il codice non viene trasferito durante la copia, ed il programma non funziona su altro disco o cassetta che non sia l'originale. L'unica protezione economicamente conveniente per produttori di videogiochi è la cartuccia ROM (di solito piuttosto cara rispetto ai programmi su nastri magnetici). Comunque anche questa può essere manomessa. Ci sono degli strumenti capaci di copiare una cartuccia su una cassetta oppure su un nuovo tipo di cartuccia, che permette all'utente di riprogrammare il contenuto.
La contraffazione di software è simile al gioco di *Guardie e Ladri*, dove i protagonisti usano tutta la loro astuzia nel battersi continuamente gli uni contro gli altri. Probabilmente, non sarà mai possibile eliminare del tutto questo fenomeno: al massimo si potrà farlo diventare talmente costoso da renderlo un'attività marginale.

Il "dongle"

Il dongle è un piccolo apparecchietto inventato per proteggere certi programmi dalla contraffazione. Difatti il programma non funziona se non viene inserita l'apposita "scatoletta nera" in una delle interfacce del computer. La parte interna elettronica è, in genere, sigillata con resina o catrame per impedirne la manomissione



IAN MCKINNELL

Le variabili a due dimensioni

Elaborare efficacemente i dati con l'impiego di tabelle

Fino ad ora, abbiamo trattato due tipi di variabili: quelle "semplici" e quelle "indicizzate". Una variabile semplice può contenere un solo valore (numerico o alfanumerico) e nomi validi, per questo genere di variabili del BASIC, sono: N, B2, X, Y3. Le variabili indicizzate, che abbiamo precedentemente paragonato a scatole con più scomparti, possono invece contenere più valori, contraddistinti dal valore di un indice. Queste variabili, prima di essere usate, devono essere DIMensionate, ad esempio: DIM A(16) definisce una variabile indicizzata A con 16 elementi. Per inciso, in alcune versioni di BASIC il primo elemento è considerato A(0), per cui DIM A(16) definisce, in realtà, una variabile a 17 elementi.

In certe applicazioni, però, è necessario elaborare dei dati sotto forma di tabelle, come accade negli "spreadsheet", o lavagne elettroniche. I dati possono essere, ad esempio, gli elementi di un bilancio familiare, suddivisi per mese, come nella seguente tabella:

	AFFITTO	TELEFONO	LUCE	VITTO	AUTO
GEN	260000	35400	41500	240000	32000
FEB	260000	42350	22700	268800	48700
MAR	260000	37900	33900	320400	39500
APR	260000	43200	44100	292400	72250
MAG	260000	52000	41750	334200	43440
GIU	260000	48900	48900	345450	50000
LUG	260000	52300	45800	299200	43400
AGO	295000	24300	24400	370850	84200
SET	295000	72200	44200	310200	34230
OTT	295000	54200	46800	325000	32750
NOV	295000	55100	47300	325000	48300
DIC	295000	64700	48900	352500	82400

Organizzando i dati in questo modo, è possibile elaborarli con relativa semplicità. Non è difficile calcolare il totale delle spese per il mese di marzo, oppure quanto si è speso in telefonate durante tutto l'anno: basta sommare i valori di una riga o di una colonna. Questa disposizione di dati può essere collocata in una variabile bidimensionale (nel nostro caso, una variabile con 12 righe per 5 colonne).

L'uso delle variabili bidimensionali, in BASIC, non differisce molto da quello delle variabili ad una sola dimensione: è sufficiente utilizzare un secondo indice.

Nella stesura di un programma per l'elaborazione di dati simili a quelli dell'esempio, l'impiego di una variabile bidimensionale è indicatissimo. Chiameremo questa variabile A. Ovviamente, la variabile deve essere dimensionata e, poiché ci servono 12 righe e 5 colonne, si scriverà: DIM A(12,5). L'ordine delle due dimensioni è molto importante e, per convenzione, il primo valore si riferisce alle righe, il secondo alle colonne.

L'istruzione DIM ha due scopi: uno è quello di riservare una sufficiente parte della memoria per collocarvi i valori della variabile, l'altro è l'indicazione di quanti indici intendiamo usare, oltre al loro valore massimo. Con DIM X(3,5), per esempio, definiamo una variabile, a due indici (bidimensionale), con tre righe e cinque colonne.

Usando la tabella, si provi a individuare i valori corrispondenti ad A(1,1), A(1,5), A(2,1), A(3,3) ed A(12,3).

L'immissione dei dati nella variabile può avvenire con delle istruzioni LET:

```
30 LET A(1,1) = 260000
40 LET A(1,2) = 35400
50 LET A(1,3) = 41500

610 LET A(12,5) = 82400
```

ma questo è senza ombra di dubbio un metodo alquanto laborioso. Un sistema molto più semplice può essere quello di impiegare le istruzioni READ-DATA oppure INPUT, all'interno di un ciclo FOR-NEXT; vediamo un esempio con la READ:

```
10 DIM A(12,5)
20 FOR R = 1 TO 12
30 FOR C = 1 TO 5
40 READ A(R,C)
50 NEXT C
60 NEXT R
70 DATA 260000, 35400, 41500, 240000, 32000, 260000, 42350,
   22700
80 DATA 268800, 48700, 260000, 37900, 33900, 320400, 39500,
   260000
90 DATA 43200, 44100, 292400, 72250, 260000, 52000, 41750,
   334200
100 DATA 43440, 260000, 48900, 48900 345450, 50000, 260000,
    52300
110 DATA 45800, 299200, 43400, 295000, 24300, 24400, 370850,
    84200
120 DATA 295000, 72200, 44200, 310200, 34230, 295000, 54200,
    46800
130 DATA 325000, 32750, 295000, 55100, 47300, 325000, 48300
```

```
140 DATA 295000, 64700, 48900, 352500, 82400
150 END
```

Ci sono diverse osservazioni da fare su questo programma. La prima è che l'istruzione DIM si trova subito all'inizio del programma: poiché, normalmente, le variabili vengono dimensionate una sola volta all'interno di un programma, è logico collocare le DIM fra le prime istruzioni e comunque esternamente a qualsiasi ciclo. La seconda osservazione è che vi sono due cicli FOR-NEXT: uno incrementa il numero della riga, l'altro quello della colonna. Un ciclo è interno all'altro o, per dirlo in gergo, è "nidificato". Si notino i valori massimi attribuiti ai contatori: FOR R = 1 TO 12 varia il valore di R da 1 a 12, mentre FOR C = 1 TO 5, varia il valore di C da 1 a 5.
L'istruzione READ è collocata nel punto più interno ai due cicli e la parte essenziale del programma è:

```
20 FOR R = 1 TO 12
30 FOR C = 1 TO 5
40 READ A(R,C)
50 NEXT C
60 NEXT R
```

Al primo passaggio, R e C varranno ambedue 1, cosicché la linea 40 è equivalente a READ A(1,1) e verrà letto il primo valore contenuto nella frase DATA, ossia 260000. La scelta di scrivere 8 valori per ogni linea di DATA è puramente arbitraria e non influisce sull'ordine di lettura.
L'esecuzione, giunta alla linea 50, incontra una NEXT C, che chiude il primo ciclo, incrementando il valore C della colonna. La linea 40 adesso equivale a READ A(1,2). Il processo si ripete finché C non raggiunge il valore di 5: a questo punto l'esecuzione passa per la linea contenente la NEXT R, che chiude il secondo ciclo, incrementando, questa volta, il valore della riga. La linea 30 viene eseguita di nuovo, facendo ripartire il valore di C da 1. La linea 40 equivale adesso a READ A(2,1). La sequenza descritta si ripete fino a che il valore di R raggiunge 12.
I cicli nidificati, che si distinguono in cicli "interni" e cicli "esterni", sono usati molto frequentemente ed il loro impiego non presenta particolari difficoltà: è sufficiente controllare, oltre al corretto ordine degli indici, che ogni ciclo sia interamente contenuto in quello più esterno. Nel nostro esempio, il ciclo esterno (FOR R = 1 TO 12) si chiude con la seconda istruzione NEXT, alla linea 60, ed il ciclo interno (FOR C = 1 TO 5) viene eseguito per intero prima di incrementare l'indice R. È possibile "nidificare" cicli anche a più livelli, ma in pratica simili situazioni vengono evitate.
Esaminiamo adesso le frasi DATA. Si noti che i valori vengono separati da virgole (non devono però esserci virgole né davanti al primo dato, né dietro l'ultimo). Nel nostro listato abbiamo inserito anche uno spazio, tra dato e dato, che, pur non essendo strettamente necessario, facilita enormemente la rilettura (o l'eventuale modifica) dei valori. In una linea contenente l'istruzione DATA possiamo inserire quanti valori vogliamo: nel nostro esempio, abbiamo impiegato otto frasi DATA, ma avremmo potuto usare anche 5 frasi DATA di 12 valori ciascuna (senza alterare, però, l'ordine di questi). Ogni nuova linea di dati deve essere iniziata con l'istruzione DATA. I valori vengono letti uno alla volta e nell'ordine in cui sono scritti. Dobbiamo sempre assicurarci che il numero di dati corrisponda esattamente al numero di letture che facciamo eseguire al programma, altrimenti sorgerà una condizione di errore in fase di esecuzione.
Il programma presentato non serve che a riempire la variabile A con dei valori, quindi, una volta "lanciato" non visualizzerà niente. Per verificarne il corretto funzionamento, possiamo usare il comando PRINT, per esempio: PRINT A(1,1) <CR>. Dovrebbe apparire il valore 260000. I comandi, in BASIC, sono quelle istruzioni che si possono eseguire "direttamente", ossia senza dover assegnar loro un numero di linea. Alcuni comandi sono: LIST, RUN, SAVE, AUTO, EDIT e PRINT. Tornando all'esempio, cosa apparirà sullo schermo se impartiamo i seguenti comandi?

```
PRINT A(12,1)
PRINT A(1,5)
PRINT A(5,1)
PRINT A(5,5)
```

Il programma presentato, in sé, non è molto utile: acquista valore soltanto se diviene parte di un programma più completo, dove verrà probabilmente scritto sotto forma di subroutine e richiamato impiegando delle GOSUB.
Poniamo, ad esempio, di voler scrivere un programma per seguire l'andamento di un bilancio familiare. Si inizia, di regola, con una concisa descrizione "a parole" delle funzioni richieste; nel nostro caso, desideriamo che il programma calcoli, per ogni riga e per ogni colonna, il totale delle spese e la media aritmetica di esse. Il programma utilizzerà un "menu" dal quale scegliere una tra più opzioni: quindi, un primo abbozzo può essere questo:

```
PROGRAMMA PRINCIPALE
(INPUT DEI DATI)

MENU
(SELEZIONE DELLA FUNZIONE RICHIESTA)

FINE
```

Dovremo prevedere delle subroutine per il calcolo dei totali mensili (TOTMESE), di quelli per categoria di spesa (TOTCAT), della media mensile (MEDIAMESE), della media per categoria di spese (MEDIACAT). Tra parentesi è riportato il nome simbolico assegnato alle subroutine, che serve, durante questa fase, ad evitare l'uso di numeri di linea che potrebbero confonderci. Anche le operazioni concernenti il menu potrebbero essere raggruppate in una

subroutine ed il successivo "piano" potrebbe risultare questo:

```
PROGRAMMA PRINCIPALE (INPUT DEI DATI)
  MENU (SUBROUTINE)
FINE

* * * LE SUBROUTINE * * *

1 MENU
2 TOTALI
3 MEDIE

(2) TOTALI
4 TOTMESE
5 TOTCAT

(3) MEDIE
6 MEDIAMESE
7 MEDIACAT
```

Da qui si vede che la subroutine di MENU ci permette di scegliere tra calcolo dei TOTALI e calcolo delle MEDIE. Il modulo TOTALI, che è a sua volta una subroutine, offre la scelta tra TOTMESE e TOTCAT. Analogamente, MEDIE ci consente di scegliere tra MEDIAMESE e MEDIACAT. Una volta verificata la correttezza logica di questa impostazione, possiamo passare a scrivere le istruzioni che formano ciascuna subroutine. Per ciò che concerne il programma principale, l'unica istruzione consiste in una GOSUB, seguita da una END:

```
145 GOSUB * * MENU * *
```

Si noti l'impiego, ancora per questa fase, di nomi simbolici e non di effettivi numeri di linea: in altri linguaggi (o nelle versioni più sofisticate del BASIC) è possibile impiegare nomi simbolici nella codifica del programma. Vediamo di definire la subroutine di MENU (la numerazione delle linee è stata omessa perché non ancora essenziale):

```
REM * * * SUBROUTINE DI MENU * * *
PRINT "CALCOLO DEI TOTALI O DELLE MEDIE?"
PRINT "(RISPONDERE T OD M)"
INPUT L$
IF L$ = "T" THEN GOSUB * * TOTALI * *
IF L$ = "A" THEN GOSUB * * MEDIE * *
RETURN
```

Supponiamo di digitare una T (per TOTALI). Il programma chiama la subroutine TOTALI, che potrebbe essere così impostata:

```
REM * * * SUBROUTINE DEI TOTALI * * *
PRINT "SI DESIDERANO I TOTALI PER MESE"
PRINT "O PER CATEGORIA?"
PRINT "(RISPONDERE M O C)"
INPUT L$
IF L$ = "M" THEN GOSUB * * TOTMESE * *
IF L$ = "C" THEN GOSUB * * TOTCAT * *
RETURN
```

La subroutine per il calcolo dei totali mensili potrebbe avere la seguente forma:

```
REM * * LA SUBROUTINE TOTALI MENSILI * *
REM VIENE CALCOLATO E VISUALIZZATO IL TOTALE
REM DELLE SPESE DI UN MESE A SCELTA
PRINT "SELEZIONARE IL MESE"
PRINT "1=GEN 2=FEB 3=MAR 4=APR 5=MAG 6=GIU"
PRINT "7=LUG 8=AGO 9=SET 10=OTT 11=NOV 12=DIC"
PRINT "DIGITARE IL NUMERO DEL MESE RICHIESTO"
LET T = 0
INPUT M
FOR C = 1 TO 5
LET T = T + A(M,C)
```

Risposte agli esercizi di pagina 175

Funzione RND

```
40 IF R > 6 THEN LET R = 1
```

Cicli e medie

```
5 FOR L = 1 TO 100
:
:
80 LET T = T + R
90 NEXT L
100 LET A = T/100
110 END
```

Sostituzione con subroutine

Si cancellano le linee 5, 80, 90, 100 e 110 nel listato della soluzione precedente. Si modifica la numerazione delle linee 10 — 70 in, per esempio 1000, — 1070. Si verifica che la linea 40 sia identica a quella della prima soluzione e si aggiunge la linea 1080 RETURN. Il programma principale deve essere modificato con: 50 GOSUB 1000 e 130 GOSUB 1000.

INKEY$

```
10 PRINT "BATTI QUALSIASI TASTO"
20 LET A$ = INKEY$
30 IF A$ = " " THEN GOTO 20
40 PRINT "IL TASTO ERA"; A$
50 END
```

Ciclo di ritardo

```
5 PRINT "BATTI IL TASTO 'SPAZIO' TRA 10 SECONDI"
10 FOR L = 0 TO 1
20 LET R = R + 1
30 IF INKEY$ = " " THEN GOTO 60
40 LET L = 0
50 NEXT L
60 PRINT "IL VALORE DI R DOPO 10 SECONDI È"; R
70 END
```

IF...THEN

```
10 GOSUB 1000
20 PRINT "INDOVINA IL NUMERO"
30 FOR G = 1 TO 5
40 INPUT N
50 IF N > R THEN GOTO 110
60 IF N < R THEN GOTO 130
70 IF N = R THEN GOTO 150
80 NEXT G
90 PRINT "HAI ESAURITO I TENTATIVI!"
100 GOTO 500
110 PRINT "VALORE TROPPO ALTO!"
120 GOTO 80
130 PRINT "VALORE TROPPO BASSO!"
140 GOTO 80
150 PRINT "BRAVO, HAI INDOVINATO!"
500 END
1000 REM * * * GENERAZIONE DEL NUMERO A CASO * * *
     (inserire qui la subroutine scelta per questa funzione)
1020 RETURN
```

```
NEXT C
PRINT "IL TOTALE DELLE SPESE PER IL MESE"
PRINT "NUMERO"; M; "È"; T
RETURN
```

Il numero che rappresenta il mese viene assegnato alla variabile M, impiegata per selezionare la corrispondente riga nella variabile bidimensionale A.
Il ciclo FOR-NEXT serve per assegnare a T la somma delle spese contenute in ciascuna colonna della variabile A.
Se avessimo scelto il mese di marzo, per esempio, la linea interna al ciclo equivarrebbe a LET T = T + A(3,1) nel primo passaggio, mentre in quello successivo a LET T = T + A(3,2) e così via.
Con lo stesso criterio sopra esposto, si possono stilare le rimanenti subroutine.
L'impiego di variabili bidimensionali, come abbiamo già detto, risulta molto utile quando si debbano elaborare dati in forma tabulare, facciano essi parte di calcoli statistici, finanziari o altro.

RISOLUZIONE DELLE EQUAZIONI DI 2° GRADO

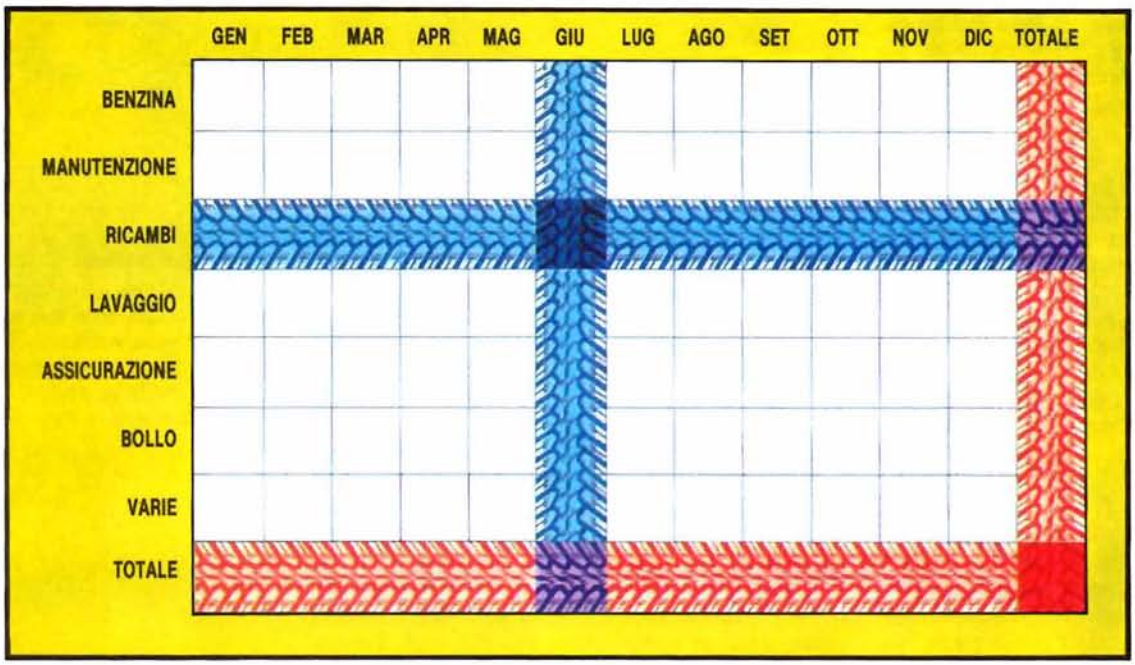
```
10 REM RISOLUZIONE EQUAZIONI DI SECONDO GRADO
20 DEF FND(X,Y,Z) = Y*Y-4*X*Z
30 REM INPUT COEFFICIENTI
40 PRINT:INPUT "DIGITA I TRE COEFFICIENTI (A = 0 -) FINE) ";A,B,C
50 IF A = 0 THEN 1000
60 LET D = FND(A,B,C)
70 IF D ) 0 THEN 200
80 IF D = 0 THEN 300
100 REM DELTA MINORE DI ZERO
110 PRINT"NON ESISTONO SOLUZIONI REALI"
120 GOTO 40
200 REM DELTA MAGGIORE DI ZERO
210 LET X = (-B-SQR(D))/(2*A)
220 LET Y = (-B+SQR(D))/(2*A)
230 PRINT "DUE SOLUZIONI REALI DISTINTE"
240 PRINT "X1 = ";X,"X2 = ";Y
250 GOTO 40
300 REM DELTA UGUALE ZERO
310 LET X = -B/(2*A)
320 PRINT "DUE SOLUZIONI REALI COINCIDENTI"
330 PRINT "X1 =  X2 = ";X
340 GOTO 40
1000 END
```

Esercizi

*** Assegnazione di valori.** Si scriva un programma per assegnare valori (Benzina, Manutenzione, Ricambi, Lavaggio, Assicurazione, Bollo, Varie.) ad una variabile bidimensionale (matrice) strutturata come nell'illustrazione.
Poi, si scriva una subroutine che permetta di visualizzare il valore corrispondente ad una qualsiasi combinazione di mese/categoria.
Infine, si scriva una subroutine che calcoli il totale di ciascuna riga e di ciascuna colonna, riportando i vari totali nelle appropriate caselle della matrice, indicate con TOTALI. La casella in basso a destra dovrà contenere il risultato generale (somma dei totali).

*** Qualche ...baco!** Il programma che segue, contiene un "baco", ossia un errore: lo si trovi.

```
10 DIM A(3,4)
20 FOR R = 1 TO 3
30 FOR C = 1 TO 4
40 READ A(R,C)
50 NEXT C
60 NEXT R
70 FOR X = 1 TO 3
80 FOR Y = 1 TO 4
100 PRINT A(Y,X)
110 NEXT Y
120 NEXT X
130 DATA 2,4,6,8,10,12,14,16,18,20,22
140 END
```



La figura mostra una matrice reticolare di 8 x 13 caselle. Le righe rappresentano i diversi elementi che concorrono alla manutenzione di un'auto, mentre le colonne sono suddivise in mesi. Usare l'illustrazione per svolgere l'esercizio **Assegnazione di valori**

I plotter

L'apparecchio ideale per utilizzare al massimo la grafica in fase di stampa

La possibilità di avere una copia su carta delle immagini che appaiono sullo schermo di un computer è essenziale in molte applicazioni: agli ingegneri, ai disegnatori tecnici e anche agli uomini d'affari occorrono diagrammi e grafici che abbiano una maggiore precisione di quella che si ottiene con normali stampanti. Lo strumento capace di riprodurre immagini con queste caratteristiche è il "plotter", un apparecchio, fino a pochi anni fa, molto costoso. Oggi si trovano in vendita apparecchi (quali la stampante/plotter a quattro penne per uso sul Tandy/CGP-115 e la stampante MCP-40 della ORIC), con un ottimo output grafico ad un prezzo alla portata di tutti.

La necessità di usare un plotter, di solito, dipende dal tipo di output generato dal computer. Un ingegnere o un disegnatore tecnico devono avere disegni accuratamente dettagliati, mentre un uomo d'affari probabilmente desidera soltanto avere dei grafici relativi alle statistiche di vendita.

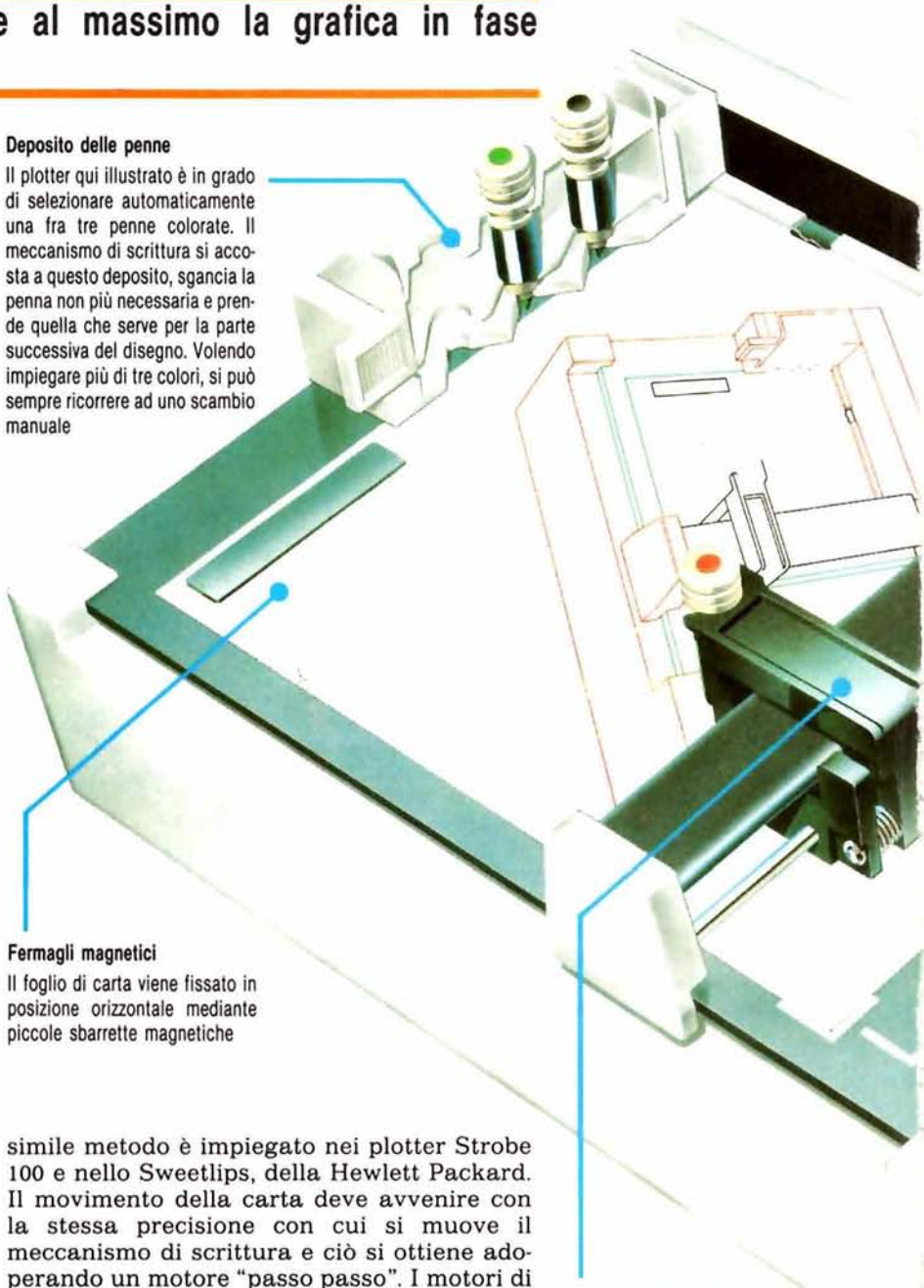
Il processo di stampa con stampanti convenzionali è assai laborioso e i risultati appaiono soltanto in bianco e nero. L'unica alternativa è quella di scattare una fotografia a colori dello schermo: una pratica che potrebbe andare bene nel caso di rappresentazioni di statistiche, ma certamente non abbastanza precisa per un disegnatore o un architetto.

Il plotter funziona con un principio completamente diverso della stampante, tracciando le linee tra diversi punti situati ovunque sulla carta, invece di riprodurre caratteri predefiniti, ad una riga per volta. Il loro funzionamento si basa sull'impiego di coordinate cartesiane: definendo le coordinate di ciascun punto di una retta, è possibile costruire anche forme complesse.

Poiché, evidentemente, è necessaria una certa libertà di movimento per collegare i vari punti, la penna del plotter è fissata su di un supporto mobile, che si può muovere sia lungo l'asse X (sinistra-destra), sia lungo l'asse Y (alto-basso). I plotter tradizionali vengono chiamati "flat bed", (ossia a "letto piatto"), poiché la carta viene fissata orizzontalmente su di un ripiano ed il meccanismo di tracciamento si muove sopra di essa. L'unico svantaggio di questo sistema è che il plotter deve per forza essere grande almeno quanto il foglio di carta.

Per ridurre la grandezza di questi apparecchi si ricorre ad un semplice stratagemma: il foglio di carta viene fatto ruotare su di un rullo (per consentire il tracciamento di linee lungo uno degli assi), mentre il meccanismo di scrittura si sposta lungo l'altro asse. Un simile metodo è impiegato nei plotter Strobe 100 e nello Sweetlips, della Hewlett Packard. Il movimento della carta deve avvenire con la stessa precisione con cui si muove il meccanismo di scrittura e ciò si ottiene adoperando un motore "passo passo". I motori di questo tipo, ad ogni impulso ricevuto, ruotano di piccolissime frazioni di grado. Una loro applicazione tipica è nelle unità a dischetti (dove provvedono al posizionamento della testina magnetica sulla superficie del disco) oppure nei robot (vedere a pagina 176).

Il sistema di collegamento del plotter al computer è generalmente lo stesso usato per collegare una stampante, almeno per quel che riguarda l'interfaccia. Esistono plotter con interfaccia seriale (RS232) o parallela (Centronics o IEEE 488). I programmi che prevedono un output su plotter sono leggermente più complicati di quelli per normali stampanti, in quanto non è sufficiente fornire



Deposito delle penne
Il plotter qui illustrato è in grado di selezionare automaticamente una fra tre penne colorate. Il meccanismo di scrittura si accosta a questo deposito, sgancia la penna non più necessaria e prende quella che serve per la parte successiva del disegno. Volendo impiegare più di tre colori, si può sempre ricorrere ad uno scambio manuale

Fermagli magnetici
Il foglio di carta viene fissato in posizione orizzontale mediante piccole sbarrette magnetiche

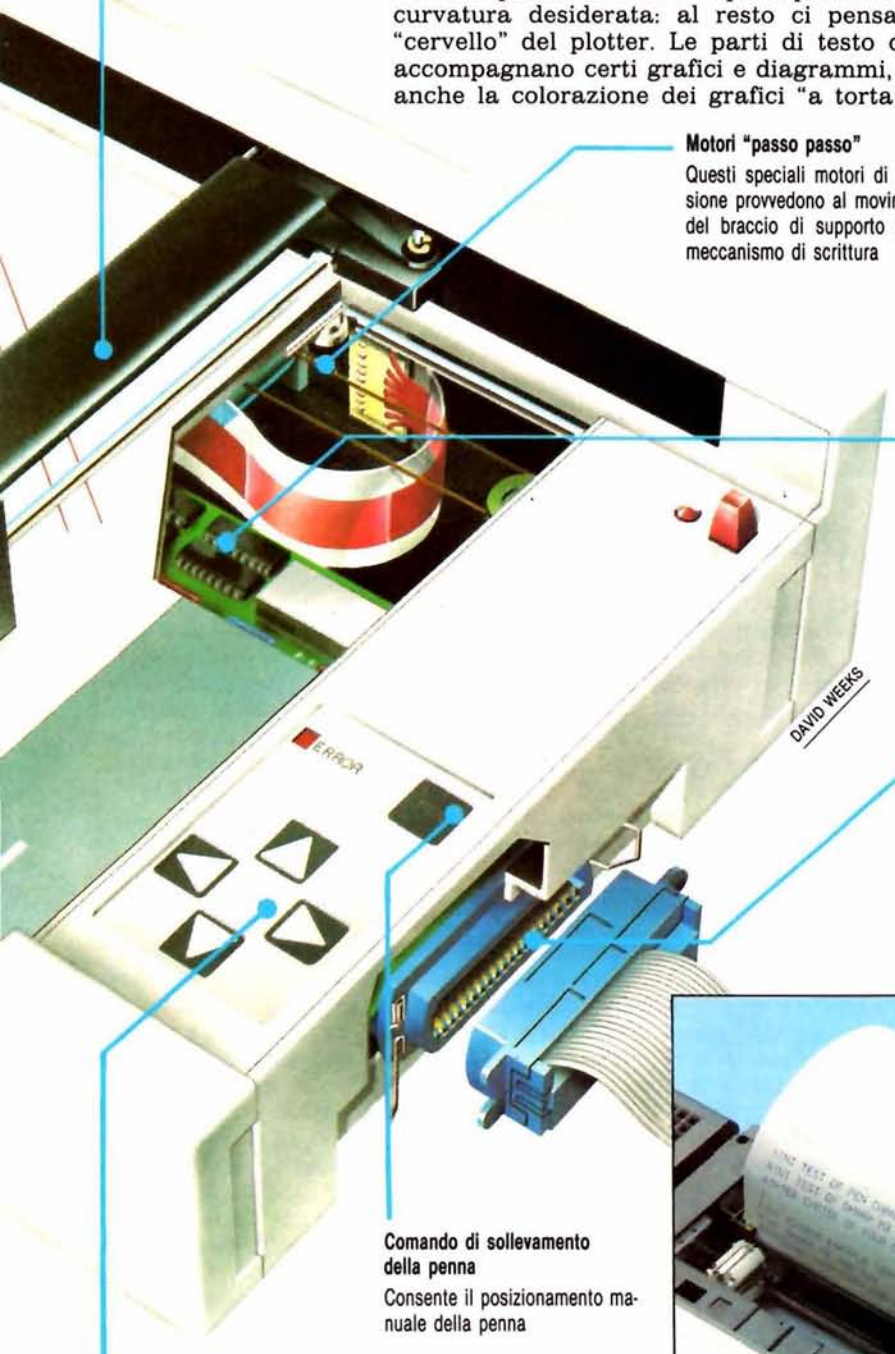
Supporto per la penna
La penna che serve per tracciare i disegni è agganciata, spesso con un sistema magnetico, al supporto mobile, che provvede automaticamente ad alzarla ed abbassarla rispetto al piano di lavoro

il dato da stampare: occorrono anche altre precise informazioni (riguardanti come e dove riprodurlo). Il plotter è considerato, in genere, un apparecchio "intelligente", in quanto è dotato di un microprocessore incorporato, che sovrintende alla conversione dei caratteri e delle istruzioni ricevute dal computer in una serie ben precisa di complicati movimenti, coi quali si ottiene la riproduzione dei disegni. Quasi tutti i plotter più sofisticati sono in grado di produrre forme complesse (come cerchi o curve), semplicemente indicando il punto iniziale dal quale partire e la curvatura desiderata: al resto ci pensa il "cervello" del plotter. Le parti di testo che accompagnano certi grafici e diagrammi, ed anche la colorazione dei grafici "a torta" o degli istogrammi sono spesso riprodotti con processi automatici.

Assieme ai plotter viene normalmente fornito l'apposito software, che permette all'utente di adoperarlo direttamente con istruzioni interne al programma applicativo, fino ad ottenere una copia su carta delle immagini che appaiono sullo schermo del computer. Quando, invece, non è disponibile il software, l'utente deve aggiungere speciali sottoprogrammi per tradurre le informazioni relative alle immagini da riprodurre nei codici adatti a pilotare il plotter. In alcuni plotter non esiste un generatore di caratteri incorporato, quindi l'utente dovrà creare anche i codici necessari per le lettere, i numeri ed i simboli. Ciò consente di creare caratteri nella forma che più ci aggrada. Le immagini e le scritte, una volta impostate, possono essere riprodotte in qualsiasi posizione, orientamento o scala, permettendo anche la creazione di un "repertorio di forme", dove può risultare utile includere quelle più frequentemente adoperate. In ogni caso, la programmazione dei plotter è abbastanza semplice e non si discosta molto dai metodi tradizionali per la creazione di disegni mediante coordinate.

Braccio di supporto

Sul braccio di supporto, mobile lungo l'asse X, si muove il meccanismo di scrittura. La combinazione dei due movimenti permette di raggiungere qualsiasi zona del foglio



DAVID WEEKS

Motori "passo passo"

Questi speciali motori di precisione provvedono al movimento del braccio di supporto e del meccanismo di scrittura

Scheda del circuito

I plotter sono, in genere, degli apparecchi "intelligenti", ossia sono dotati di un microprocessore interno che controlla e regola automaticamente tutte le funzioni in base alle istruzioni ricevute dal computer

Connettore per interfaccia

I plotter vengono collegati ai computer mediante interfacce seriali (RS232) o parallele (Centronics). Per il computer, il plotter non è che un tipo di stampante più sofisticata

Comando di sollevamento della penna

Consente il posizionamento manuale della penna

Controlli manuali di spostamento

La penna può anche essere pilotata manualmente, grazie a questi pulsanti



CHRIS STEVENS

PLOTTER/STAMPANTE A QUATTRO PENNE

Questo particolare tipo di meccanismo, adottato per la prima volta sullo Sharp CE-150, ha suscitato molto interesse, perché ha permesso di dotare anche i più piccoli computer portatili di una grafica a colori.

Il principio di funzionamento è molto semplice: il rullo di carta può ruotare sia in avanti che indietro con piccoli e precisi passi, mentre il meccanismo di scrittura, dotato di quattro penne a sfera colorate, scorre in senso orizzontale.

Il supporto delle penne viene ruotato per selezionare il colore di scrittura e quindi appoggiato sulla carta. Con questo sistema, dal costo veramente ragionevole, si ottengono anche grafici piuttosto complessi e precisi, anche se le dimensioni del rullo (in genere simile a quello adoperato per le calcolatrici) impone qualche restrizione

Alan Turing

L'informatica deve molto a questo genio matematico

Una sfida non soltanto... matematica!

Alan Turing (1912-1954) trovava ispirazione e relax facendo lunghe maratone. Egli era molto incuriosito dagli effetti provocati dallo sforzo fisico sull'agilità e la creatività mentali



Fin da giovane Alan Turing dimostrò una notevole perspicacia scientifica. Una volta scrisse alla madre, dal collegio: «Ho sempre il desiderio di creare delle cose nuove con gli elementi che ci offre la natura». I matematici, generalmente, manifestano il loro talento fin da giovanissimi ed Alan Turing non fece eccezione. Appena fu in grado di leggere e scrivere, si divertiva a trovare il minimo comune multiplo del numero dell'inno da cantare in chiesa e progettò perfino una bicicletta anfibia.

Durante il periodo in cui suo padre era di stanza in India, Turing ottenne numerosi premi scolastici, vincendo, infine, una borsa di studio per frequentare il prestigioso Kings College di Cambridge. Fu infatti durante gli anni che passò a Cambridge, prima come studente e dopo come assistente, che egli iniziò ad interessarsi ai problemi concernenti la logica matematica.

Nel 1931, il cecoslovacco Kurt Godel suscitò scalpore nel mondo scientifico, scoprendo che ci sono dei teoremi matematici che sono validi, senza possibilità di una dimostrazione. Alan Turing si impegnò per accertare quali di questi teoremi potevano, in effetti, essere più o meno convalidati da dimostrazioni matematiche.

Egli abbozzò il progetto di una macchina (senza peraltro occuparsi della sua costruzione) in grado di eseguire meccanicamente i processi normalmente svolti dai matematici. Ciascuna operazione matematica era affidata ad una macchina (per esempio, una per fare le addizioni, una per le divisioni e un'altra per le integrazioni, ecc.). Macchine di questo tipo furono chiamate successivamente "Macchine di Turing".

Turing fece molte ricerche sul funzionamento di queste macchine immaginarie, arrivando all'"incredibile" conclusione che non occorreva avere una singola macchina per ciascuna operazione matematica, ma che era possibile progettare uno strumento "universale" fatto in modo da raggruppare in sé tutte le altre, grazie ad una programmazione adeguata. Per puro caso, Turing si era imbattuto nella teoria del calcolatore programmabile.

Quando scoppiò la Seconda Guerra Mondiale, il Governo Britannico lo reclutò nel dipartimento governativo "codici e cifre", situato in Bletchley Park, Buckinghamshire. Se la guerra non fosse mai scoppiata, forse le macchine sarebbero rimaste solamente nella mente di Turing, ma a Bletchley Park c'era da fare un lavoro urgentissimo e coperto dal massimo segreto: decifrare i codici militari tedeschi.

Questi codici venivano modificati anche di giorno in giorno, occorrevano quindi delle macchine capaci di decifrare i codici prima che venissero soppiantati da altri nuovi.

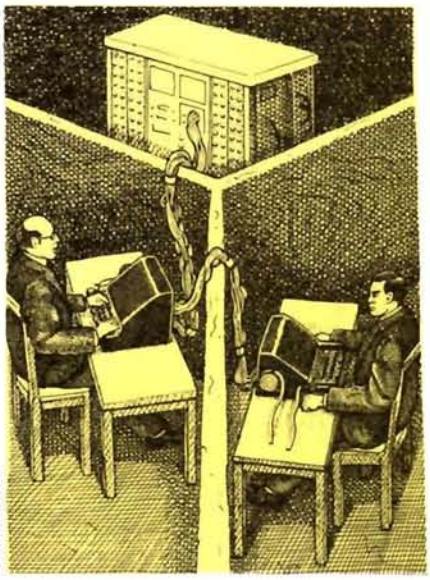
Bletchley Park divenne un grande centro per l'elaborazione di informazioni. Verso la metà della guerra, il Governo Inglese mandò Turing negli Stati Uniti d'America, per definire dei codici speciali di cui si dovevano avvalere gli Alleati per le loro comunicazioni transatlantiche.

La natura segretissima del suo lavoro in questo periodo non ci rivela molto a proposito dei suoi movimenti, ma si suppone che abbia conosciuto Von Neumann a Princeton, New Jersey. Verso la fine della guerra a Turing fu commissionato il progetto di un computer interamente britannico, nominato ACE (Automatic Computing Engine) per il National Physical Laboratory.

Il nome dato al computer deriva in parte da quello della Macchina Analitica di Babbage. Un elemento comune tra le due macchine fu senz'altro il tempo impiegato per la costruzione: anche l'ACE ne richiese molto, ma per vari aspetti esso era molto più avanzato tecnicamente dell'ENIAC (vedere a pag. 46).

Frustrato da tante lungaggini, Turing dette le dimissioni e si trasferì all'Università di Manchester per partecipare alla progettazione di un altro computer. Durante lo stesso periodo agì da consulente presso la Ferranti, una grande azienda elettronica, e contribuì notevolmente alla produzione dei primi computer in Gran Bretagna. Turing era un eccentrico, che raramente si curava delle convenzioni sociali o delle restrizioni legali. Un amico disse di lui che era "divinamente ritardato" nel vedere i difetti degli altri, ma che il suo genio scientifico era impareggiabile.

Nel 1952 venne condannato per aver commesso reati di carattere omosessuale e due anni più tardi si suicidò. Egli avrebbe forse portato un enorme contributo nel campo dell'intelligenza artificiale, se fosse vivo ancora oggi.



BOB VENABLES

Le macchine sono in grado di pensare?

Per rispondere a questa domanda Turing propose un test diventato poi famoso, chiamato da lui "il gioco dell'imitazione", più tardi conosciuto con il nome di "Test di Turing". Esso consiste nel mettere un uomo in una stanza con a disposizione una telescrivente (composta da una tastiera ed una stampante). Questa è collegata con un altro apparecchio analogo in un'altra stanza, azionato sia da un altro uomo che da un computer, oggetto della prova. Il primo uomo doveva rivolgere una domanda qualsiasi all'altro uomo o al computer, indifferentemente. Se dalle risposte date non era in grado di dedurre chi dei due stesse rispondendo (l'uomo o il computer), si poteva dunque ritenere che la macchina era dotata di una "intelligenza". D'altronde, seguendo la logica del test, non abbiamo altro modo di accertare se un'altra persona pensa o se è cosciente, al di fuori del paragonare con le nostre le sue reazioni a determinate circostanze

Pronti al decollo?

I programmi per la simulazione di volo su computer: a metà tra il gioco ed il professionale



Permesso di decollare

«Bravo alfa tre due cinque, avete il permesso di decollare sulla pista numero uno due. La velocità del vento è di uno cinque nodi, l'orientamento è di uno zero cinque. Avvisateci quando superate i 1000 metri...».
Ma in questo caso la pista davanti al Boeing 737 esiste soltanto grazie ad impulsi digitali generati dal sistema computerizzato Monoview SP3 per la simulazione di volo

Il concetto di simulazione di volo non è affatto nuovo: già dal 1940 gli allievi piloti venivano addestrati su riproduzioni di aerei, conosciuti sotto il nome di "Link Trainers": la prima generazione di simulatori di volo. Questi modelli di aereo erano piuttosto grossolani e servivano unicamente per offrire ai giovani piloti un primo contatto con l'uso degli strumenti di comando e per insegnare le principali nozioni di volo.
Con l'avvento della prima generazione di jet a più motori, quale il "Comet" della De Havilland, gli istruttori delle varie linee aeree sentirono il bisogno di disporre di sistemi più economici di addestramento, che non l'impiego, in volo, di veri aerei. La vasta gamma di apparecchi ed i maggiori consumi dei nuovi aerei, infatti, rendevano l'addestramento molto costoso. La soluzione venne dall'elettronica, con i primi simulatori di volo computerizzati, che adoperavano tecniche analogiche e digitali.
Questi simulatori erano montati su speciali arieti idraulici ed erano, in pratica, un perfetto duplicato della cabina di pilotaggio dell'aereo che gli allievi piloti dovevano imparare a manovrare. Il sistema idraulico era controllato da un computer e rispondeva ai comandi impartiti dal pilota proprio come se si trattasse di un aereo vero, consentendo anche la simulazione di particolari situazioni di emergenza, improponibili, perché troppo pericolose, durante l'addestramento in volo effettivo.
Erano riprodotti non solo i comandi per il pilota, ma anche quelli per tutto l'equipaggio di volo, perché volare, specialmente su aerei di linea, è un'attività che non dipende più dal solo pilota, ma da tutta un'équipe di persone. L'unico elemento mancante, per una completa simulazione, era la riproduzione della vista che il pilota ha davanti a sé durante la



per concessione della REDIFFUSION SIMULATION LTD.

Un aereo... su palafitta!

Esternamente, un simulatore di volo assomiglia ad una scatola, di forma piuttosto strana, montata su appoggi idraulici, ma una volta all'interno, si entra in un mondo completamente diverso: la cabina di un moderno aereo

fase di decollo e di atterraggio. Alcuni primi tentativi si basarono sulla proiezione di sequenze filmate, ma la soluzione non era certo ideale, non potendo tener conto delle manovre del pilota, diverse in ciascun caso.
Fortunatamente, proprio quando questo problema stava divenendo insostenibile, la soluzione fu trovata, di nuovo, nel campo della tecnologia elettronica. A pagina 181, abbiamo visto come la produzione computerizzata delle immagini (Computer Generated Imaging) abbia notevolmente influito sull'industria dei cartoni animati. Le stesse tecniche sono state impiegate per fornire, durante la simulazione di decolli ed atterraggi, immagini realistiche, che mostrano ai piloti la pista dell'aeroporto durante tutte le manovre.
Inoltre, poiché le informazioni per costruire le immagini animate possono essere conservate su dischetti, il sistema di simulazione è in grado di offrire al pilota la scelta su di un grande repertorio di diversi aeroporti.



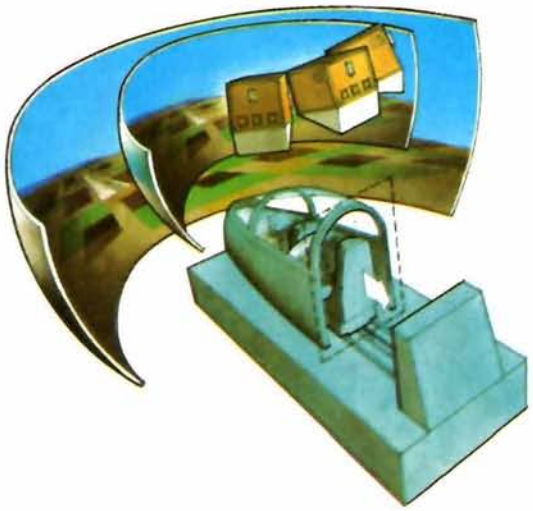
Il quadro di comando

Il simulatore di volo prodotto dalla Microsoft mette l'utente di fronte al quadro di comandi di un aereo monomotore Cessna 182 in partenza da uno dei 20 aeroporti veri degli Stati Uniti. La replica del quadro dei comandi, piuttosto complesso, appare sulla parte inferiore dello schermo, mentre la parte superiore rappresenta una veduta del "mondo esterno" in prospettiva. Il package include anche un gioco di combattimento, ma è da considerarsi piuttosto come un programma per "l'apprendimento con aiuto del computer" progettato per l'insegnamento delle principali conoscenze della navigazione aerea

Simili tecniche sono ancor più sfruttate nei simulatori di volo impiegati per l'addestramento dei piloti militari. In questo caso l'obiettivo principale non è tanto quello di abituare i piloti a volare sui diversi tipi di aereo (i moderni sistemi di volo sono così sofisticati, infatti, che gli aerei sono virtualmente autopilotati), ma piuttosto di abituare i piloti e gli osservatori militari a riconoscere i bersagli e ad imparare le tecniche di lancio dei missili aria-terra e aria-aria. Comunque sia, la necessità di base è sempre quella di offrire al pilota un'immagine, quanto più realistica possibile, della visuale che avrebbe dalla cabina di pilotaggio.
Quando abbiamo parlato delle lavagne elettroniche (pag. 158), abbiamo descritto una tecnica basata sull'impiego di "finestre": questa tecnica permette all'operatore di vedere sullo schermo del monitor, una alla volta, porzioni di un "foglio" di dimensioni maggiori di quelle dello schermo. Nei simulatori di volo più sofisticati, un simile sistema di visualizzazione è indispensabile, per dare l'illusione che l'aereo si muova veramente, rispondendo ai comandi del pilota, in relazione allo scenario esterno. Per esempio, quando il pilota effettua una virata verso il basso, la visuale davanti a lui dovrà cambiare, rispecchiando fedelmente gli effetti della manovra. Così, ogni volta, il pilota deve agire con la manovra più adatta al momento e non limitarsi a manovre imparate "a pappagallo".
Alcune scuole militari vanno ancora oltre: usufruiscono di immagini ad altissima risoluzione (come fotografie scattate da satelliti o da aerei da ricognizione in volo ad alta quota, oppure immagini preparate in base a rapporti forniti da osservatori) per la produzione di film, che mostrano, in grande dettaglio, gli aspetti fisici del terreno sopra il quale



il pilota si troverebbe a volare, a bassa quota, durante una missione di bombardamento.
L'uso di questi film, congiuntamente a quello del simulatore di volo, consente una verosimiglianza quasi completa. Il pilota affronta tutte le fasi di una sortita completa dal decollo fino al rientro alla base, ma tutto ciò senza mai alzarsi, in realtà, da terra.
Le tecniche di simulazione non vengono impiegate soltanto per l'addestramento di volo: anche la marina mercantile le sta adoperando, affinché i guardiamarina acquistino dimestichezza con l'entrata nei canali e nei porti marini principali ed i sommozzatori riescano meglio a riconoscere oggetti in condizioni di cattiva visibilità.
Inoltre, anche gli astronauti usufruiscono di simili sistemi per la simulazione dei viaggi spaziali. Le possibilità offerte dalla produzione computerizzata di immagini, combinate a



La proiezione spaziale

Impiegando tre diversi proiettori televisivi a tre fasci di luce (rossa, verde e blu) l'immagine animata del mondo esterno viene proiettata sullo schermo posto davanti al pilota. Uno specchio di collimazione provvede a compensare la distorsione causata dallo schermo curvo

quelle della simulazione di volo, permettono di riprodurre praticamente qualsiasi esperienza visiva e, contemporaneamente, aprono nuovi orizzonti anche per una nuova generazione di giochi e di passatempi.
Alcuni videogiochi, che utilizzano semplici simulatori di volo, sono un esempio lampante di questo fenomeno: il giocatore deve cercare di manovrare un aereo, a volte in condizioni avverse (magari con poco carburante a disposizione, con cattiva visibilità oppure con i motori in avaria, ecc.). Tuttavia, è sbagliato pensare che questi giochi (con la sola eccezione, forse, di quello prodotto dalla Microsoft) siano in diretta relazione con i veri simulatori di volo professionali.
Nel gioco le situazioni simulate sono spesso puramente immaginarie (come nel caso dello *Space Invaders* e *PacMan*): chi li ha progettati ha stabilito dei parametri e li ha tradotti in possibili azioni da intraprendere. A volte, invece, essi costituiscono una simulazione più aderente alla realtà, come nel caso di corse automobilistiche.
Non tutto il software "di simulazione" per microcomputer è indirizzato al semplice divertimento: ci sono anche dei programmi scritti per l'apprendimento, mediante computer, della navigazione aerea, del controllo del traffico aereo e dell'impostazione generale di volo. Per esempio, il simulatore di volo della Microsoft, disponibile per il personal della IBM, è stato progettato sia come semplice divertimento sia come un serio esercizio per futuri piloti.
Nel manuale si spiega che il programma simula tutti gli strumenti e le attrezzature previsti dal regolamento dell'Autorità Federale Americana per l'Aviazione. La parte bassa dello schermo rappresenta il quadro di



comandi di un Cessna 182 (l'aereo monomotore sul quale la simulazione è basata), mentre sulla parte più alta dello schermo appare un'immagine animata a colori a media risoluzione della veduta fuori dall'aereo. Vengono anche impiegati effetti pseudotridimensionali, ossia, viene data l'illusione di movimento in base a variazioni della prospettiva.
Nel package è inclusa anche la carta geografica di una vasta estensione di territorio, che copre un'area di circa 2,5 milioni di chilometri quadrati (in pratica tutti gli Stati Uniti, parte del Canadà, del Messico e del Mare dei Caraibi) e sono riprodotti oltre 20 diversi aeroporti nelle quattro aree principali: New York/Boston, Chicago, Los Angeles e Seattle. In teoria è possibile "volare" attraversando tutta l'America, da New York a Los Angeles, via Chicago, ma, siccome non esistono distributori per il rifornimento di carburante fuori dalle aree densamente popolate, ben presto precipiteremmo senza scampo. Anche se fosse possibile portarsi dietro sufficiente carburante, il simulatore è basato sul "tempo reale", il che significa che, volando alla velocità massima di circa 240 km all'ora, per fare tutto il viaggio occorrerebbero circa 20 ore (una simulazione che può diventare decisamente noiosa!). Per questo motivo il sistema prevede la possibilità di "saltare" da una zona ad un'altra.
Il giocatore ha molte possibilità di scelta: in quale ora del giorno desidera volare, oppure in quale stagione (automaticamente cambiano l'ora dell'alba e del tramonto), in quali condizioni atmosferiche (cielo sereno o nuvoloso), la velocità di crociera e la direzione del vento e può perfino introdurre un fattore di turbolenza nell'aria.
Il simulatore di volo della Microsoft è basato su di un aereo molto "docile" e semplice, e, grazie ad una programmazione in linguaggio macchina molto efficiente, è stato possibile condensare tutto il programma in modo da usare soli 64 Kbyte di memoria. Naturalmente, simulare la cabina di pilotaggio ed i movimenti di un jet moderno più sofisticato, diventa un'impresa non più adatta ai piccoli computer di oggi. Domani forse...



per concessione della REDIFFUSION SIMULATION LTD.

"Caccia all'aereo nemico"
Immagini di altissima qualità, generate al computer, danno la possibilità agli addetti al volo (sia all'equipaggio di piloti che ai controllori a terra) di esercitarsi nell'arte di manovrare un aereo nelle più disparate condizioni (scelte dall'istruttore). La foto, prodotta col sistema Simulations Novoview della Rediffusion, mostra l'aereo militare americano da combattimento A10



Organizzazione dei dati

Come evitare duplicazioni e sfruttare meglio la memoria

La quantità d'informazioni contenute nello schedario di una biblioteca è vastissima, ma se non viene mantenuto un rigoroso ordine logico, il valore informativo dello schedario è praticamente nullo.
L'essenza di un sistema informativo, infatti, non sta tanto nelle informazioni contenute ma nel modo in cui sono organizzate. Prendiamo ad esempio il seguente nominativo (fittizio) estratto dalle Pagine Gialle:

Alfredo Bernetti, Via Mazzini 125, Milano

Il solo nominativo, di per sé, dice poco, ma se sappiamo che è elencato sotto la voce "Ferramenta", esso assume un'importanza maggiore perché veniamo a conoscere di che cosa si occupa la persona in questione.
La struttura informativa più semplice, in un archivio, è il raccoglitore (in inglese: "file"): in esso vengono archiviate, per argomento, informazioni di vario genere.
L'etichetta o il nome, riportati sulla costola del raccoglitore, forniscono una prima indicazione sul contenuto e, se tutte le informazioni pertinenti allo stesso argomento vengono inserite nel raccoglitore, la consultazione diventa facile e rapida. Un raccoglitore può essere dedicato ad un particolare argomento generale, oppure a tutto un gruppo di informazioni, suddivise a loro volta in "sotto argomenti". Per certi versi, anche un libro può essere considerato un "raccoglitore": una biografia su Pellegrino Artusi, per esempio, viene letta nella sua globalità, mentre il suo libro di ricette è composto da una serie di singoli gruppi d'informazioni.
Se un raccoglitore è voluminoso, per trovare una particolare informazione occorre scorrere tutti gli incartamenti, fino a trovare quello giusto. Questo sistema viene definito "ricerca sequenziale" o "accesso seriale" ed il raccoglitore equivale ad un file "sequenziale" del computer. Un altro esempio di file "sequenziale" è costituito dai programmi televisivi, poiché rispettano una "scaletta" o un orario ben precisi. Questo metodo di gestire le informazioni è il più comune, sia perché è utile ed economico, ma soprattutto perché ben si adatta alla mentalità umana. Tuttavia, la consultazione diventa spesso un compito lento e laborioso, pertanto i raccoglitori vengono di solito suddivisi, internamente, in diversi scomparti, per facilitare la ricerca di un particolare dato, senza dover sfogliare tutte le cartelle in esso contenute. Lo stesso avviene nei libri, dove la suddivisione in capitoli, la numerazione delle pagine e la presenza di un indice agevolano la consultazione.
Ma nei computer si può accedere ad un file anche seguendo un altro sistema, chiamato "ad accesso diretto". Per meglio chiarire la differenza, ricorriamo ad un altro esempio. Una raccolta di canzoni incise su nastro magnetico equivale ad un file "sequenziale": infatti, per trovare una particolare canzone occorre riavvolgere il nastro e riascoltarlo fino a che non si arriva alla canzone voluta. Ma se la stessa raccolta è invece incisa su disco, basta posizionare il braccio del giradischi sul punto giusto del disco. Il disco, dunque, consente un accesso "diretto" alle informazioni in esso contenute.
L'accesso diretto, ovviamente, è possibile soltanto se si conosce l'esatta ubicazione del dato cercato: in un libro è l'indice che fornisce questa indicazione. Purtroppo, non sempre i libri contengono indici completi: la preparazione dell'indice, infatti, è un compito veramente lungo e laborioso (anche se oggi i computer aiutano anche in questo!).
I dati, che su nastro magnetico devono essere necessariamente conservati secondo una struttura più o meno sequenziale, nella memoria del computer possono anche essere organizzati in modo totalmente diverso.
Prendiamo l'esempio di un commerciante, che abbia archiviato i nominativi e gli indirizzi dei propri clienti, su nastro magnetico, in un file sequenziale. Al momento delle consegne, gli sarà estremamente utile disporre di un elenco, riordinato secondo un percorso ottimale. Il file originale, su nastro, potrebbe avere una struttura di tipo alfabetico:

Antonietti, Via Cavour 34
Benettoli, Corso Italia 65
Fontanelli, Via Roma 87a
Gensini, Piazza della Stazione 2
Ulderici, Via Milazzo 11
Vandelli, Piazza Mazzini 34r

Quando i vari nominativi ed indirizzi vengono trasferiti nella memoria interna del computer, ciascuno di essi è depositato in una locazione numerata tutta sua, e tutte le locazioni, nel loro insieme, formano un "blocco" di memoria con, volendo, un proprio nome identificativo. Il file, in memoria, potrebbe essere:

NOMEBLOCCO: Clienti
1) Antonietti, Via Cavour 34
2) Benettoli, Corso Italia 65
3) Fontanelli, Via Roma 87a
4) Gensini, Piazza della Stazione 2
5) Ulderici, Via Milazzo 11
6) Vandelli, Piazza Mazzini 34r

Grazie a questa nuova organizzazione, si può accedere ad ogni singolo dato, semplice-

Il percorso più breve

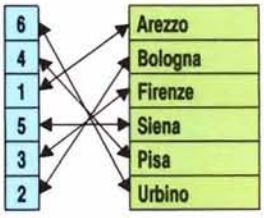
Una ditta effettua, una volta al mese, un giro di consegne in sei diverse città. Il computer può indicare quale sia il percorso più economico da seguire.

Local$(1)	Arezzo
Local$(2)	Bologna
Local$(3)	Firenze
Local$(4)	Pisa
Local$(5)	Siena
Local$(6)	Urbino

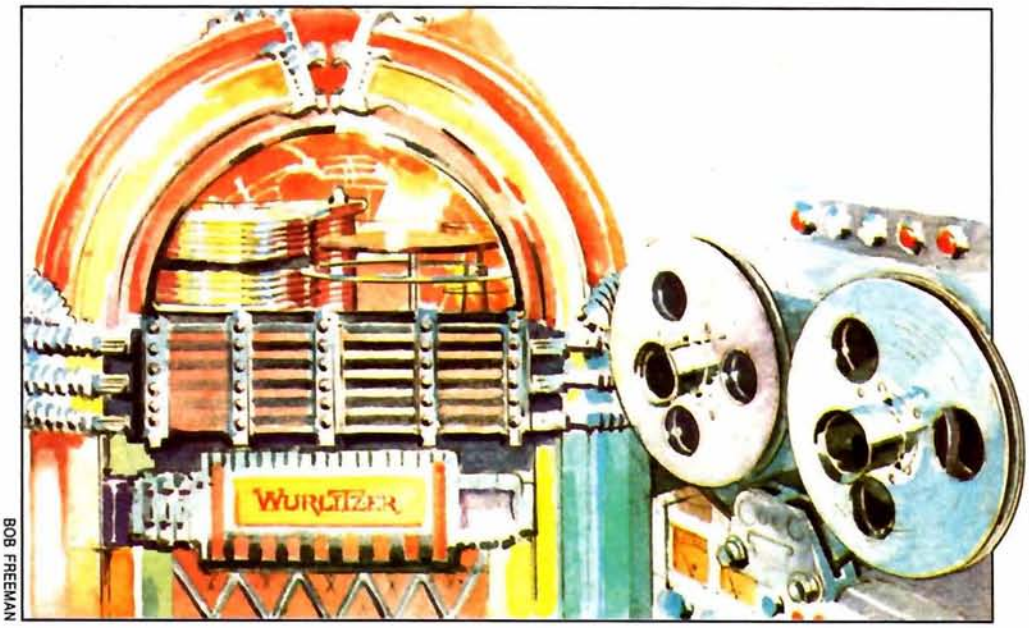
I nomi delle città, in ordine alfabetico, sono conservati in un file "sequenziale" su nastro magnetico. In questo stesso ordine vengono letti e trasferiti nella matrice alfanumerica Local$( ). Supponiamo che il computer abbia indicato, in base alle distanze, ai quantitativi di merce ed altri fattori, questo percorso ottimale:

Urbino	Local$(6)
Pisa	Local$(4)
Arezzo	Local$(1)
Siena	Local$(5)
Firenze	Local$(3)
Bologna	Local$(2)

Invece di sprecare memoria duplicando i nomi delle città disposti secondo questo nuovo ordine, il computer conserva unicamente il numero che si riferisce alle rispettive posizioni delle città nella matrice Local$( ):



La sequenza dei numeri costituisce un "indice" di rilettura della matrice Local$( ). Si può creare un'infinità di altri indici, ciascuno dei quali accede ai nomi delle città secondo i criteri più diversi. Il vantaggio di questo metodo risiede nel fatto che non è necessario riorganizzare o duplicare il file originale ma basta riorganizzare e conservare gli indici



BOB FREEMAN

SULLA PISTA GIUSTA

Un juke-box contiene circa 200 canzoni. Il suo costo è di circa 5 milioni di lire ed il suo peso è di circa un quintale. Per selezionare una canzone basta la pressione di un paio di tasti e dopo pochi secondi si può ascoltare la canzone. Un nastro magnetico per registratori professionali potrebbe contenere le stesse 200 canzoni. Un registratore a bobine costa circa un quarto del juke box e pesa, più o meno, 10 volte meno. Per selezionare una canzone, però, occorre riavvolgere il nastro, premere il tasto di ascolto ed... aspettare. Il tempo che trascorre, dalla selezione del brano all'effettivo ascolto, può essere 100 volte superiore a quello del juke-box. Il juke-box si può definire un apparecchio ad accesso "diretto". È veloce, piuttosto sofisticato e costoso. Il registratore dispone soltanto di un accesso "sequenziale", è lento, ma economico. Il registratore a cassette collegato ad un microcomputer è un apparecchio ad accesso "sequenziale", mentre l'unità a dischetti è ad accesso "diretto", benché si possa impiegare anche per conservare file sequenziali

mente in base al nome del "blocco" ed alla posizione, in esso, del dato Clienti (4) si riferisce a Gensini, Piazza della Stazione 2. Questa struttura, a chi segue le lezioni di programmazione BASIC, è già nota ed è chiamata "matrice" o "variabile indicizzata" (vedere a pag. 194).
È la struttura più comunemente impiegata per l'elaborazione di dati col computer, ed è paragonabile ad un libro che ha una sola informazione per pagina. È da notare come questa semplice modifica alla struttura dei dati porti ad una diversa interpretazione degli stessi: un elenco di nominativi e di indirizzi si è trasformato in un blocco di dati anonimi. Al computer, in effetti, non interessa il significato delle informazioni da elaborare, ma soltanto dove esse si trovino e cosa farne una volta individuate.
I dati contenuti nella matrice Clienti ( ), sono disposti in ordine alfabetico, ma avremmo fortuna se questo corrispondesse anche al migliore (o più economico) percorso per le consegne! Supponiamo che il computer, grazie ad un apposito programma, ci comunichi che il percorso migliore sia il seguente:

1) Gensini, Piazza della Stazione 2
2) Antonietti, Via Cavour 34
3) Fontanelli, Via Roma 87a
4) Benettoli, Corso Italia 65
5) Vandelli, Piazza Mazzini 34r
6) Ulderici, Via Milazzo 11

L'elenco, così formato, potrebbe essere conservato in un'altra matrice, ma ciò comporterebbe una superflua duplicazione di informazioni nella memoria e, poiché la RAM è limitata, una simile duplicazione è sconsigliabile. Ricorriamo così ad uno stratagemma: al posto dei nominativi e degli indirizzi, usiamo le posizioni occupate nella matrice Clienti ( ), cosicché il "blocco" Consegne diviene:

NOMEBLOCCO: Consegne
1) 4
2) 1
3) 3
4) 2
5) 6
6) 5

In altre parole ciò equivale a dire: «la prima consegna è al cliente indicato alla voce Clienti (4), poi a quello elencato sotto la voce Clienti (1)» e così via. L'unica informazione significativa in questa "tabella di marcia" è l'ordine con cui la consegna va effettuata e pertanto questa è l'unico dato da conservare nel nuovo "blocco" intitolato Consegne, che diviene così una matrice di "puntatori" al blocco Clienti ( ), per quanto riguarda le consegne. Quando il computer stampa la tabella di marcia, utilizza i numeri trovati in Consegne ( ) per stampare, nel corretto ordine di consegna, i nominativi e gli indirizzi contenuti in Clienti ( ).
Nell'esempio che abbiamo presentato, l'informazione principale, ossia i nominativi e gli indirizzi dei clienti, è stata elaborata, ma non modificata dalle diverse strutture di dati impiegati. Le strutture di dati, infatti, non modificano il contenuto, ma soltanto l'interpretazione delle informazioni, collegandole tra loro secondo un criterio organizzato.
Così come abbiamo riorganizzato l'accesso alla matrice Clienti ( ), conferendole la struttura finale ottenuta con la matrice di puntatori Consegne ( ), è possibile costruire anche altri indici adatti ad altre circostanze, e necessità. Parlando dei "database" (vedere a pag. 124), abbiamo fatto notare che si possono estrarre o selezionare gruppi di informazioni, grazie a speciali puntatori, contenuti nel "record" di ciascun nominativo. Sfruttando a pieno questo metodo, si potrebbe inserire un puntatore per ciascun elemento nella matrice Clienti ( ), che serva ad indicare la posizione dello stesso elemento all'interno di Consegne ( ).
Si potrebbe andare oltre e includere un puntatore ad un record che contiene le ordinazioni: così ogni ordine potrebbe essere verificato direttamente sulla tabella di marcia, oppure il computer potrebbe provvedere anche ad emettere le bolle di consegna.

Le interfacce

Non basta il solo cavo per allacciare gli apparecchi periferici al computer

Interfacce per registratore

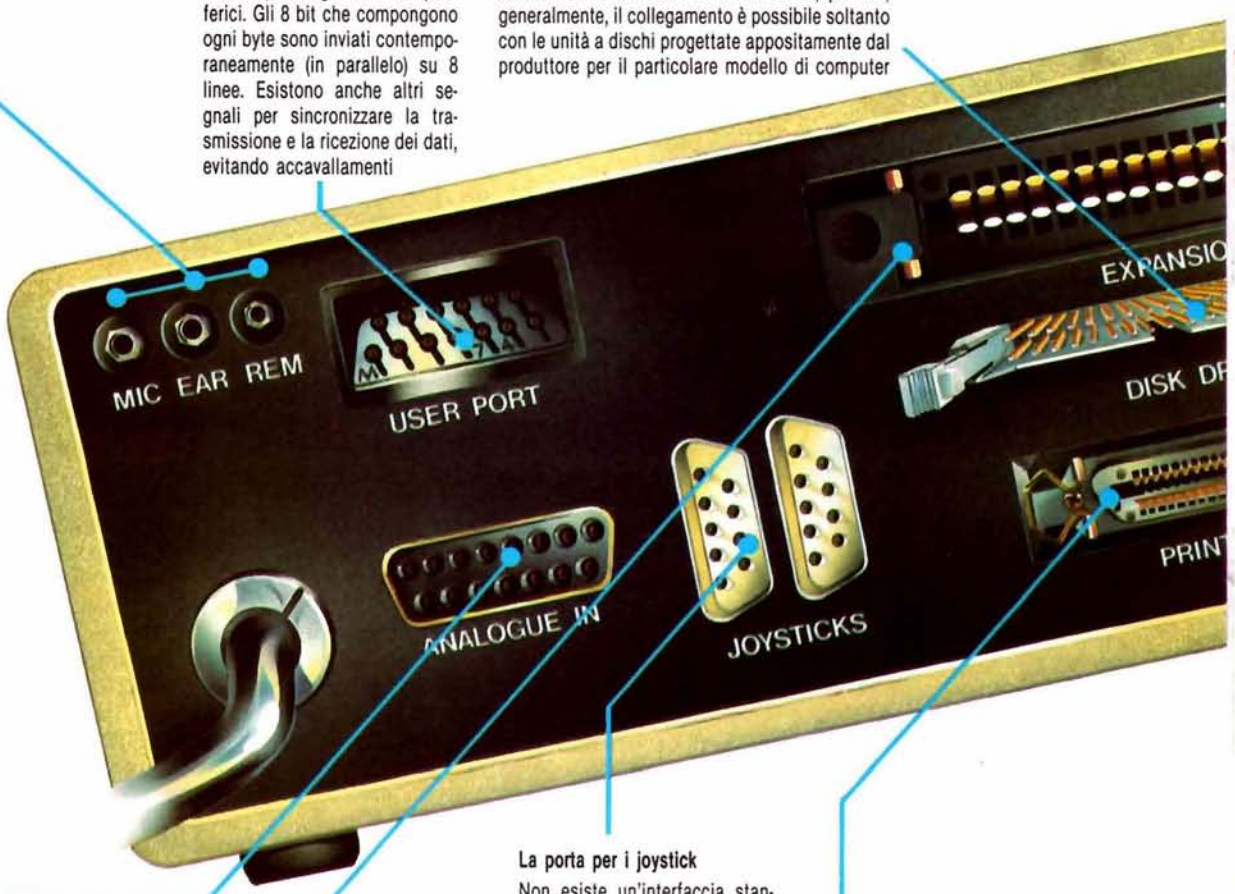
La porta o interfaccia per le cassette di cui la maggior parte di home computer è dotata, consiste, in realtà, in una specie di interfaccia seriale. Siccome i dati devono essere registrati su normale nastro magnetico, impiegando frequenze audio, non è possibile ottenere una velocità di trasmissione di dati molto elevata. Il circuito d'interfaccia preleva dalla memoria i byte di dati da registrare e li converte in un flusso di bit (valori 0 e 1), a loro volta convertiti in due diverse frequenze audio. Quando, invece, si "legge" il nastro, i segnali vengono decodificati dal circuito d'interfaccia e i risultanti bit, riorganizzati in byte, vengono depositati in memoria. Le interfacce per cassette, presenti sulla maggioranza degli home computer, sono "universali" in quanto si possono usare normali registratori a "cassetta", di tipo domestico. Le prese, in genere, sono del comunissimo tipo DIN oppure del tipo "mini jack", a "banana"

Porta parallela

Questa è una interfaccia parallela multiuso per il collegamento del computer agli strumenti periferici. Gli 8 bit che compongono ogni byte sono inviati contemporaneamente (in parallelo) su 8 linee. Esistono anche altri segnali per sincronizzare la trasmissione e la ricezione dei dati, evitando accavallamenti

L'interfaccia per unità dischi

L'unità a dischi è normalmente collegata al computer per mezzo di un'interfaccia parallela. Non esiste alcuno standard di trasmissione, per cui, generalmente, il collegamento è possibile soltanto con le unità a dischi progettate appositamente dal produttore per il particolare modello di computer



Input analogico

Questo è normalmente disponibile solo sui computer più costosi, oppure progettati per uso didattico. L'input analogico è utile nei casi in cui si desideri collegare il computer con strumenti da laboratorio (termometri, manometri o altri strumenti di misura elettronici, sensori ottici, e via dicendo). L'interfaccia dispone di una o più linee capaci di "leggere" un voltaggio (entro limiti specifici). L'utente deve evitare con cura il collegamento con apparecchi che impieghino un voltaggio superiore ai limiti previsti per evitare gravi danni all'apparecchio

La porta di espansione per la memoria

Questa porta permette l'accesso a tutte (o quasi tutte) le linee provenienti direttamente dal microprocessore, cioè i bus degli indirizzi, dei dati e di controllo. A questo connettore viene allacciata la memoria aggiuntiva oppure, in qualche caso, apparecchi periferici della stessa marca del computer. Nella maggioranza dei casi il connettore è semplicemente ricavato direttamente sulla piastra del circuito stampato, mentre in altri viene impiegata una presa simile a quelle usate per le cartucce (che, in sostanza, non sono che un'espansione della ROM)

La porta per i joystick

Non esiste un'interfaccia standard per i joystick, anche se molti produttori adoperano lo standard Atari per evitare di dover progettare joystick nuovi. Le interfacce di questo genere di solito hanno cinque linee attive, una proveniente da ciascun interruttore di "movimento" del joystick ed uno per il tasto di "fuoco". Tuttavia, i joystick di tipo analogico necessitano di interfacce diverse, capaci di sopportare voltaggi di diverso livello. La maggioranza dei computer possiedono più porte per joystick, utilizzabili eventualmente anche per altri apparecchi

L'interfaccia per la stampante

Questa interfaccia è standard, secondo il sistema adottato dalla Centronics Corporation, pertanto è abbastanza facile far funzionare una stampante con interfaccia Centronics su quasi tutti i computer. Anche i livelli e le funzioni dei segnali sono standardizzati a 0 e 5 volt per rappresentare, rispettivamente, i valori binari 0 e 1. I connettori e la disposizione dei vari segnali non sono standard e può quindi essere necessario un apposito cavo per collegare la stampante al computer

L'interfaccia per il video

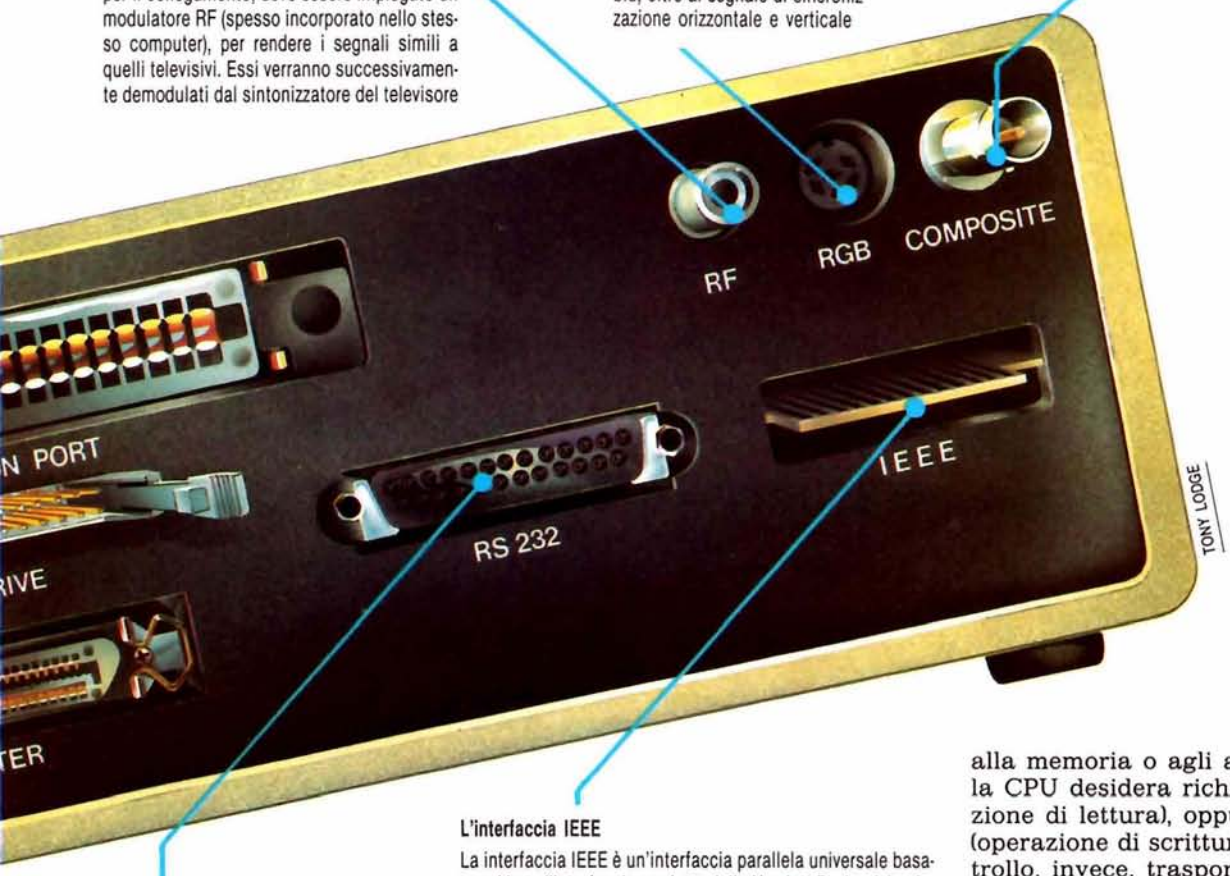
Tutti gli home computer sono progettati per il collegamento ad un video che nella maggioranza dei casi consiste in un normale televisore in bianco e nero od a colori. Se si desidera utilizzare la presa di antenna del televisore, per il collegamento, deve essere impiegato un modulatore RF (spesso incorporato nello stesso computer), per rendere i segnali simili a quelli televisivi. Essi verranno successivamente demodulati dal sintonizzatore del televisore

L'interfaccia RGB

I risultati migliori si ottengono tenendo separati i segnali di colore necessari al monitor. L'interfaccia RGB fornisce dei segnali separati per il rosso, il verde ed il blu, oltre al segnale di sincronizzazione orizzontale e verticale

L'interfaccia per video composito

Alcuni televisori e molti video monitor impiegano un segnale video composito, che non deve essere sottoposto alla fase di modulazione/demodulazione RF. Le immagini risultano di qualità nettamente superiore. Tutti gli elementi di un segnale video standard (la luminosità, il contrasto, i segnali di sincronizzazione, ecc.) sono forniti dal computer; ma questo segnale "composito" non necessita di alcuna ulteriore modulazione



L'interfaccia seriale

Diversamente dalla maggioranza delle interfacce, l'interfaccia seriale RS232, almeno in teoria, è definita in modo preciso come uno standard, secondo le norme della Electrical Industries Association. Lo standard specifica il tipo di connettore da impiegare (un connettore a 25-poli), i segnali assegnati a ciascuna linea ed i livelli dei segnali. Sfortunatamente solo pochi produttori si attengono alle norme stabilite e, talvolta, è difficile far funzionare strumenti seriali con alcuni computer. Dei 25 poli presenti sul connettore, soltanto tre, di solito, vengono effettivamente utilizzati: il polo 2 per trasmettere dati seriali dal computer allo strumento periferico, il polo 3 per l'operazione inversa ed il polo 7 per il segnale di terra. Sia lo strumento trasmittente che quello ricevente dovranno essere sincronizzati su di una specifica velocità di trasmissione dei dati ed il formato di trasmissione deve essere compatibile

L'interfaccia IEEE

La interfaccia IEEE è un'interfaccia parallela universale basata sul bus d'interfaccia prodotta dalla Hewlett-Packard, impiegata come standard dall'Istituto degli Ingegneri Elettrici e Elettronici britannico (IEEE). Lo standard ha una definizione ben precisa, sia fisica che elettrica. Diversamente da altre interfacce per dati (per esempio, la interfaccia Centronics parallela e RS232 seriale) che si possono collegare solo ad uno strumento periferico per volta, sul bus IEEE si possono collegare simultaneamente fino a 15 strumenti (incluso il computer). Fra gli apparecchi dotati dell'interfaccia IEEE ci sono stampanti, unità a dischi, plotter, voltometri ed altri strumenti di controllo. Infatti, l'interfaccia IEEE viene sfruttata molto con strumenti da laboratorio e di misura. Oggi solo pochi home computer offrono una interfaccia IEEE.

Il termine "interfaccia" è usato genericamente per indicare la porta o presa che serve per collegare al computer apparecchiature esterne. Tuttavia, per essere più specifici, il termine si riferisce al circuito elettronico (hardware) ed al software che rendono possibile il collegamento dei due apparecchi.

Internamente, il computer comunica inviando dati per mezzo dei "bus", cioè serie di conduttori paralleli, ciascuno dei quali trasporta un unico segnale in formato binario. Nella maggioranza dei microcomputer ci sono tre bus interni: il bus dei dati, dalla capacità di 8 bit; il bus degli indirizzi, da 16 bit; e il bus di controllo, con segnali tra i 5 ed i 12 bit. Alcuni segnali di controllo comunicano alla memoria o agli apparecchi periferici se la CPU desidera richiamare dei dati (operazione di lettura), oppure depositare dei dati (operazione di scrittura). Altri segnali di controllo, invece, trasportano informazioni provenienti dall'esterno e dirette alla CPU, come nel caso, ad esempio, di un apparecchio periferico che abbia dei dati da immettere e chieda l'intervento della CPU.

Internamente, il computer elabora uno alla volta i dati, generalmente composti da 8 o da 16 bit. Pertanto, se la CPU desidera leggere un dato conservato nella locazione di memoria 65535 (o FFFF in numeri esadecimali), adopera i 16 canali del bus degli indirizzi come un unico dato a 16 bit, che costituisce l'"indirizzo" della particolare locazione.

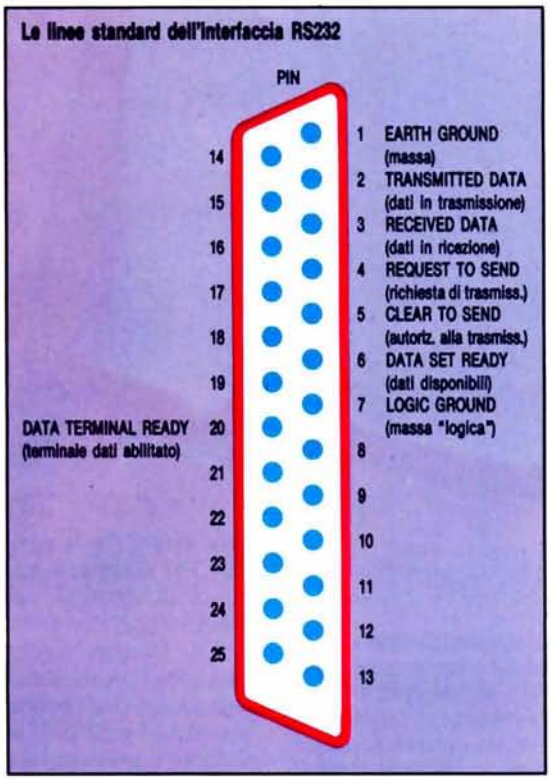
Se il contenuto di tale locazione di memoria fosse, per esempio, 182 (o B6 in numeri esadecimali), questi dati saranno trasferiti sul bus dei dati in numeri binari (10110110).

Questo tipo di trasferimento di dati a 8 o a 16 bit alla volta si chiama "in parallelo".

Molti strumenti periferici sono progettati in modo da trasmettere o ricevere dati in parallelo. Le interfacce provviste per strumenti periferici di questo tipo si chiamano "interfacce parallele" e quasi tutti i computer sono provvisti di almeno una presa o connettore specifico per apparecchi paralleli.

Comunque, non tutti gli accessori periferici ricevono o trasmettono dati in forma parallela: alcuni usufruiscono di una sola linea, trasmettendo (o ricevendo) un bit alla volta, cominciando dal bit "meno significativo" del

byte (quello più a destra) via via fino a quello "più significativo". Ciascun byte viene suddiviso in una serie di bit, trasmessi consecutivamente: all'altro capo del cavo, il byte viene ricomposto per la successiva elaborazione, mediante speciali convertitori parallelo-seriale e seriale-parallelo. Sia l'interfaccia seriale che quella parallela possono ricevere l'ordine di trasmettere dati da o verso il computer. Normalmente, i computer sono dotati di interfacce che servono esclusivamente o per la trasmissione di dati verso l'esterno (per esempio: il circuito di uscita video) o per la ricezione dei dati inviati, ad esempio, dai joystick.



L'interfaccia seriale

Come abbiamo avuto modo di dire precedentemente, il settore dei computer soffre di una forte mancanza di "standard", ossia di convenzioni sui segnali impiegati nello scambio di dati. Fortunatamente, tra le varie interfacce seriali, ve n'è una definita come "standard": l'interfaccia RS232, nella quale sia il numero che la disposizione ed il livello dei segnali sono stati stabiliti una volta per tutte. La RS232 è una interfaccia seriale "bidirezionale": una linea serve per la trasmissione, un'altra per la ricezione. I dati vengono inviati un bit alla volta sulla linea "dati trasmessi" e ricevuti sulla linea "dati ricevuti". La sequenza dei bit può assumere diversi formati, ma non ha importanza quale viene adoperato, purché i due apparecchi comunicanti impieghino il medesimo formato.
Siccome ciascun byte di informazione è trasmesso in una sequenza seriale di bit, il software dedicato alla interfaccia dovrà indicare in qualche modo quando viene inviato il primo bit e quando è stato trasmesso l'ultimo. Il metodo più comune consiste nell'iniziare la trasmissione con un "bit di start" (di valore 0), seguito da 8 bit di dati, seguiti, a loro volta, da un "bit di stop" (di valore 1).
È necessario specificare, prima dell'inizio dello scambio di dati, la velocità di trasmissione altrimenti il succedersi di segnali a livello basso ed alto, che rappresentano gli 8 bit a livello logico 0 e 1, potrebbe essere mal interpretato. La velocità della trasmissione dei dati è classificata secondo la scala Baud (dal nome di Baudot, uno scienziato francese del XIX secolo). La scala Baud prevede un arco di velocità che va dai 75 ai 9.600 bit per secondo: poiché ciascun carattere è composto di 10 bit (8 di dato + quelli di start e di stop), la velocità di trasmissione di ciascun carattere equivale ad un decimo del valore espresso in Baud.

L'interfaccia parallela

L'interfaccia parallela trasmette e riceve informazione un byte alla volta, ma oltre alle 8 linee dei dati, ne occorrono altre per indicare al computer (o agli apparecchi periferici) quando è il momento giusto per inviare un carattere. Il tipo più comune di interfaccia parallela è la "Centronics" (il cui nome deriva dalla ditta americana Centronics Corporation, produttrice di stampanti), ma anche in questo caso non tutti i produttori aderiscono a questo cosiddetto standard e esiste una certa divergenza fra il tipo di connettore e la disposizione dei segnali sui piedini del medesimo. Ad ogni buon conto, sulla maggioranza delle interfacce Centronics esistono per lo meno i seguenti segnali:

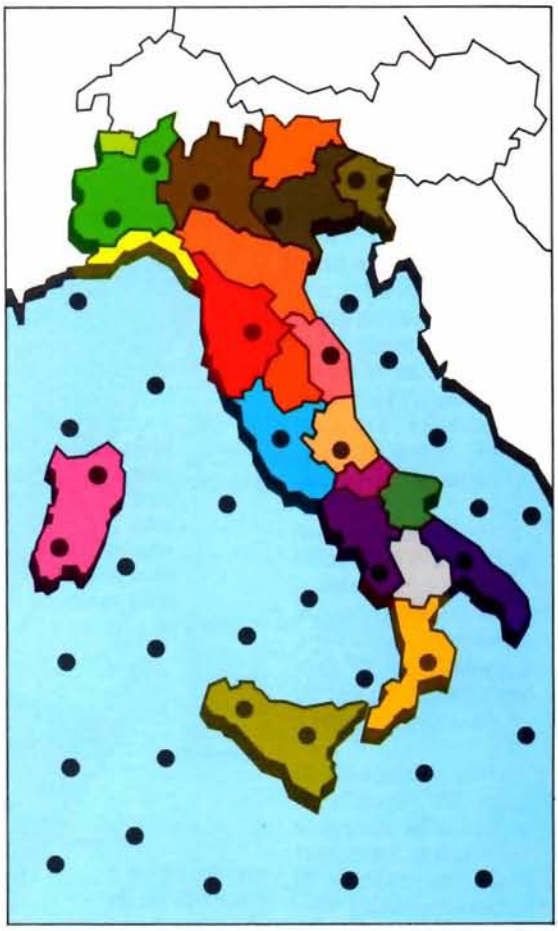
DATI 0 - 7	(8 linee per trasportare gli 8 bit del byte trasmesso)
ADK	questo segnale indica al computer se la periferica è pronta a ricevere i dati
GND	la linea di terra serve, ad ambedue gli apparecchi, per avere un riferimento a 0 volt
BUSY	questo è il segnale usato dall'apparecchio periferico per indicare al computer che non è in grado di ricevere dati (BUSY = occupato)
STROBE	con questo segnale, il computer avverte la periferica di aver pronto un dato per la lettura

Molti altri apparecchi, oltre alla stampante, si avvalgono dell'interfaccia parallela Centronics "quasi standard": in questi casi il collegamento al computer non presenta altre difficoltà se non l'acquisto di un cavo di connessione adatto ed, in generale, non occorre apportare modifiche al software dedicato al funzionamento dell'apparecchio periferico.

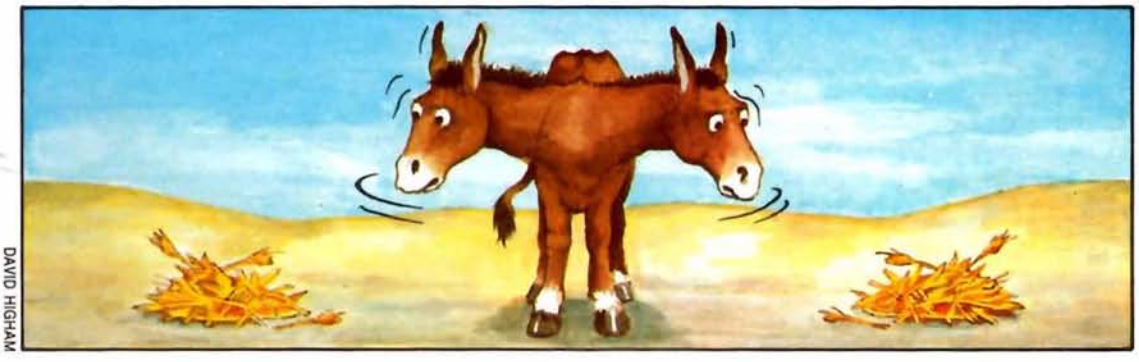
Affidarsi al caso

Alcune interessanti applicazioni dei numeri casuali

La scelta di un numero "a caso" è un'operazione facilissima per una persona, ma, anche se può sembrare un paradosso, quasi impossibile per un computer.
Un numero, per essere veramente casuale, deve essere totalmente imprevedibile, mentre in un computer, che si limita all'esecuzione di semplici comandi, il numero generato è frutto di una precisa sequenza di operazioni.
Poco importa quanto queste operazioni siano complicate: il numero prodotto rimane sempre, almeno in teoria, prevedibile (per arrivare al numero, basta seguire le stesse istruzioni impartite al computer): pertanto tale numero non può essere definito casuale.
In alcuni casi, per ottenere numeri veramente casuali, occorre affidarsi a fenomeni fisici: ERNIE (un computer adoperato in Gran Bretagna per una lotteria), per esempio, impiega un sistema basato sul movimento di elettroni liberi per "estrarre" i numeri vincenti.
Von Neumann fu il primo a concepire l'idea di numeri pseudo-casuali, ossia numeri generati secondo un processo matematico. Il suo metodo consiste nel prendere un numero di 4 cifre (diciamo 4.321) e elevarlo al quadrato. Dal risultato (nel nostro caso 18.671.041), composto di 8 cifre, si estraggono le 4 cifre centrali, (ossia 6.710) che rappresentano un primo numero casuale. Questo numero, a sua volta, è usato come "seme" per la generazione del numero successivo casuale. Il numero 6.710 elevato al quadrato, equivale a 45.024.100, le cui cifre centrali sono 0241, che formano il seme del successivo numero.
Questo processo può continuare all'infinito, ma siccome il totale di numeri diversi composti di 4 cifre è limitato a 9.999, prima o poi la sequenza inizia a ripetersi.
Un moderno computer impiega metodi più complessi per "creare" i numeri pseudocasuali. Per citare un esempio, il Microcomputer BBC è in grado di generare un numero casuale ogni 1,5 millisecondi. Se il computer dovesse generare numeri in continuazione, ci vorrebbero, in teoria, almeno 150 giorni, prima che una sequenza si ripeta.



I primi a generare elettronicamente numeri casuali furono alcuni ingegneri impegnati a simulare le oscillazioni nell'utenza delle centrali telefoniche. Attualmente, esistono molti impieghi dei numeri casuali, che vanno dal semplice videogioco alla simulazione di fenomeni fluttuanti (nella fisica o nell'economia), fino alla valutazione di difficili funzioni matematiche. In quasi tutte le applicazioni, gli attuali algoritmi per la generazione di numeri pseudocasuali può sostituire efficacemente quella di numeri veramente casuali.



DAVID HIGHAM

Il metodo "Monte Carlo"

Questo metodo, inventato da John Von Neumann, impiega numeri casuali per fornire le risposte a problemi matematici. Spesso occorre determinare l'area racchiusa in una figura non regolare, come i contorni di una nazione. Nell'illustrazione a fianco, una cartina dell'Italia è contenuta in un rettangolo di cui possiamo facilmente calcolare l'area. Si inizia col disseminare, a caso, una certa quantità di puntini sulla superficie del rettangolo. Poiché la posizione dei puntini è scelta a caso, essi cadranno un po' dovunque e si può supporre che il numero di essi sulla superficie del territorio sia proporzionale alla superficie stessa. Nel nostro esempio sono stati usati 40 puntini 16 dei quali sono caduti internamente ai confini, mentre 24 esternamente. Sapendo che l'intera superficie del rettangolo equivale a 745.000 km quadrati, si può dedurre che la superficie dell'Italia è:
16/40 x 745.000 = 298.000 km quadrati.
Usando una maggiore quantità di puntini, il calcolo della superficie risulta meno approssimato, avvicinandosi all'effettiva estensione di 301.262 km quadrati

L'asino di Buridano

Immaginiamo di mettere un asino, dotato di assoluta razionalità, in mezzo a due mucchi di fieno. L'asino non sa scegliere il mucchio più grosso, perché i due mucchi sono identici, e non sa scegliere quello a lui più vicino, perché sono ambedue alla stessa distanza. In questo caso potrebbe essere attratto da uno dei due mucchi, ma perché dovrebbe notare un mucchio piuttosto di un altro? Siccome non ha un motivo razionale per scegliere un mucchio in particolare, l'asino rimane fermo dove è e muore di fame! Similmente, un computer non è in grado di generare numeri veramente casuali, perché il suo funzionamento è basato su di una logica ferrea

Lo ZX81

Anatomia di un microcomputer a diffusione europea: tecnologie a confronto

Lo ZX81 della Sinclair è, ancora oggi, uno dei più economici computer sul mercato. Nato come naturale evoluzione dello ZX80, la principale caratteristica che lo distingue è l'incredibile compattezza della circuiteria interna: l'impiego di uno speciale chip, chiamato ULA (Uncommitted Logic Array), che, in un solo circuito integrato, racchiude l'equivalente di decine di circuiti integrati "generici". Lo ZX81 è un computer senza fronzoli, interruttori o pulsanti superflui: tutto è ridotto all'essenziale.

L'alimentatore, esterno, si collega ad una presa laterale. Sono presenti anche connettori per il registratore a cassette, per il video e per altre periferiche, come la stampante ZX. Molto economica, di pochissimo ingombro, questa stampante offre la possibilità di produrre delle "hard copy" (ossia: copie su carta) di listati di programmi o di risultati di elaborazione.

C'è da dire, però, che la carta impiegata, di tipo termico, è piuttosto cara. La visualizzazione avviene, contrariamente alla maggior parte dei computer, con l'impiego di caratteri neri (solo maiuscoli) su sfondo chiaro. Il BASIC Sinclair è alquanto diverso dalla versione Microsoft, ma più che adatto al principiante.

La tastiera dello ZX81 suscitò, a suo tempo, un certo stupore: era tra le prime tastiere "a sfioramento" montate in un computer. L'uso di simili tastiere non è particolarmente gratificante per chi sia abituato alle normali macchine da scrivere, ma non presenta problemi per l'immissione di programmi, anche per l'ingegnoso sistema adottato, basato sull'impiego di abbreviazioni dei comandi: spesso basta premere un solo tasto e sullo schermo compare, per esteso, l'istruzione BASIC corrispondente. All'accensione, sullo schermo appare una K, ad indicare che il computer attende un comando BASIC. Ogni tasto, selezionando opportunamente il modo di funzionamento, può servire per l'immissione di tre tipi diversi di informazioni: comandi BASIC (in forma abbreviata), le normali lettere dell'alfabeto ed i vari simboli grafici predefiniti.

È stato prodotto tantissimo software per questo piccolo gioiello ed, in alcuni casi, la limitata memoria disponibile non ha rappresentato un ostacolo insormontabile, ma piuttosto una sfida. Comunque, si trova in commercio un'ampia varietà di interfacce e di periferiche, tra le quali schede di memoria aggiuntive. Purtroppo, però, la connessione elettrica (realizzata direttamente sul circuito stampato del computer) non offre grandi garanzie di solidità. Sono state perfino realizzate delle tastiere a dimensione naturale, da collegare al piccolo apparecchio.

Lo ZX81 ha incontrato un enorme successo presso i giovanissimi, ed è senza dubbio un apparecchio che si presta egregiamente alla didattica.

QUANTITÀ O QUALITÀ?

Quando Clive Sinclair decise di lanciare sul mercato un computer molto economico, furono necessari alcuni compromessi sulla qualità. Il primo, forse, riguarda la tastiera, che viene considerata il punto debole sia dello ZX80 che dello ZX81. Ciò ha generato una colossale domanda di software "pronto" (da non dover digitare sulla scomoda tastiera) ed, infatti, il mercato è stato inondato da ogni tipo di programma per questi minuscoli ed economici computer. Oltre al BASIC, è possibile impiegare il linguaggio FORTH (vedere a pag. 147) od un Assembler per il microprocessore Z80

La tastiera dello ZX81
Per una maggiore economicità è stata impiegata una tastiera del tipo "a sfioramento", criticata da molti utenti sia per le dimensioni, che per l'insicurezza che proviene dal suo impiego (non si è mai sicuri che la pressione del tasto sia effettivamente avvenuta). Di contro, la tastiera consente di immettere intere istruzioni BASIC mediante la pressione di singoli tasti. Ogni tasto è associato a tre diverse funzioni



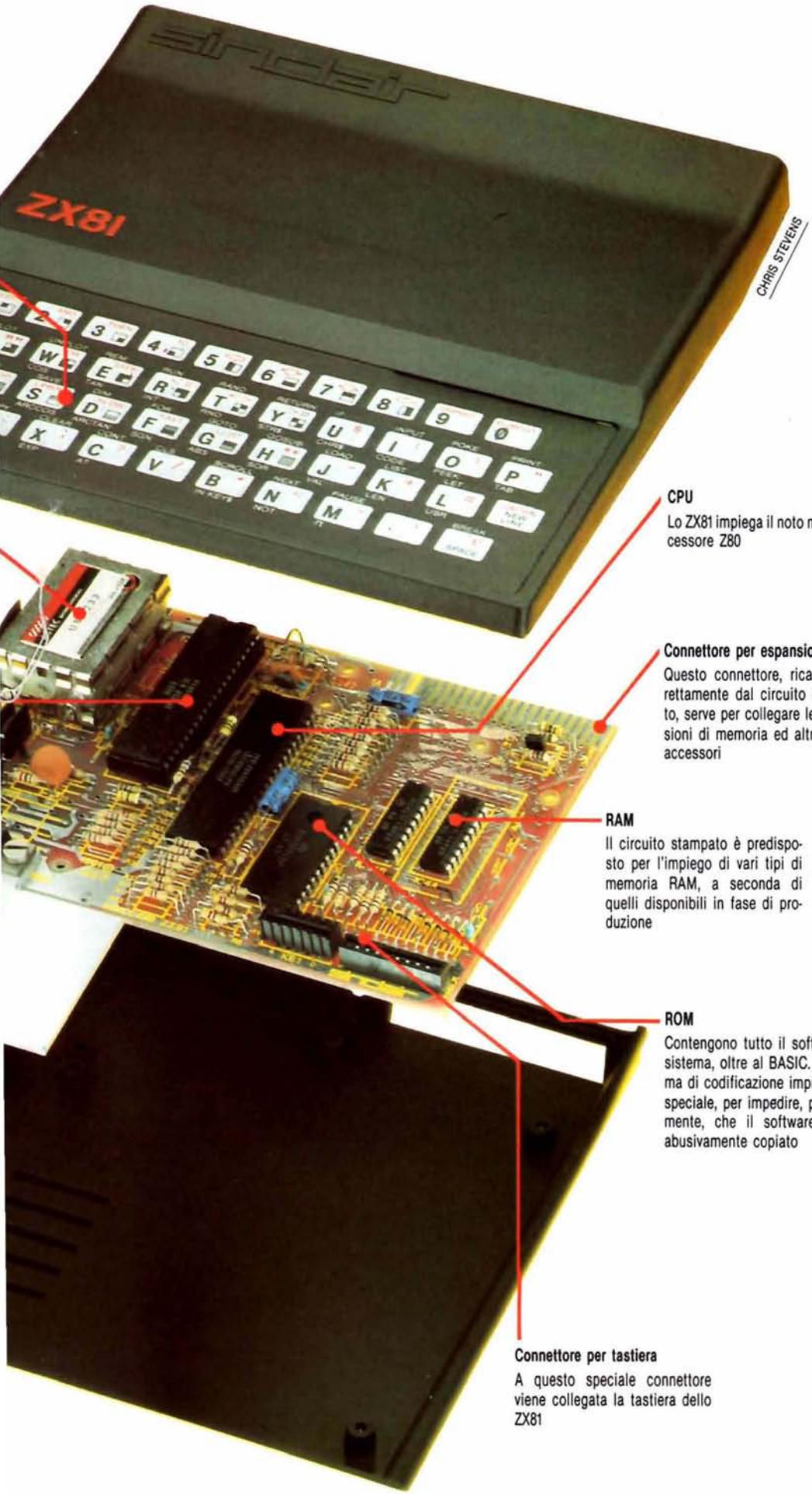
ULA (Uncommitted Logic Array)
Questo speciale chip, costruito appositamente per la Sinclair, racchiude al suo interno l'equivalente di qualche decina di circuiti integrati generici. Ciò ha permesso la produzione di un computer che impiega veramente pochi componenti (molti meno dello ZX80). Chip del genere sono molto difficili da progettare ed, in genere, servono unicamente per un solo tipo di applicazione

Modulatore RF
Emette i segnali necessari per un normale video TV



Connettori
Questi tre "jack" miniaturizzati servono per l'alimentazione, per l'ingresso e per l'uscita dei dati da e per il registratore a cassette. Alcuni programmi usano l'uscita del registratore per riprodurre suoni musicali





CPU
Lo ZX81 impiega il noto microprocessore Z80

Connettore per espansioni
Questo connettore, ricavato direttamente dal circuito stampato, serve per collegare le espansioni di memoria ed altri tipi di accessori

RAM
Il circuito stampato è predisposto per l'impiego di vari tipi di memoria RAM, a seconda di quelli disponibili in fase di produzione

ROM
Contengono tutto il software di sistema, oltre al BASIC. Il sistema di codificazione impiegato è speciale, per impedire, probabilmente, che il software venga abusivamente copiato

Connettore per tastiera
A questo speciale connettore viene collegata la tastiera dello ZX81

ZX81

DIMENSIONI

mm. 175 x 168 x 43

PESO

300 grammi

CPU

Z80A

CLOCK

1 MHz

MEMORIA

8 Kbyte di ROM contengono il BASIC ed il software di sistema 1 Kbyte di RAM, della quale 123 byte sono utilizzati dal sistema operativo, una parte dal video ed il restante per i programmi applicativi. La RAM può essere estesa a 16 Kbyte con modica spesa

VIDEO

24 linee di 32 caratteri, o 64 x 44 in modo grafico. Solo bianco e nero. Normalmente le ultime due righe dello schermo vengono usate per l'immissione delle linee di comando

INTERFACCE

Connettore per TV, connettore per registratore a cassette. Uno speciale connettore permette l'espansione della memoria

LINGUAGGI FORNITI

BASIC

ALTRI LINGUAGGI DISPONIBILI

FORTH e Assembler Z80

ACCESSORI COMPRESI

Alimentatore, cavi di connessione per TV e per registratore a cassette, documentazione

TASTIERA

40 tasti "a sfioramento". Ogni tasto possiede tre modi di funzionamento: per comandi BASIC, per normali caratteri e per simboli grafici

DOCUMENTAZIONE

Il manuale è scritto in modo da risultare comprensibile ai principianti, contiene molti esempi, esercizi e tavole riepilogative. Alcuni capitoli sono dedicati alla descrizione tecnica dell'apparecchio ed alla programmazione in linguaggio macchina

Facciamo il punto

Le strutture di controllo, alcune nuove funzioni ed un po' di ripasso

Nel corso delle prime dieci lezioni abbiamo esaminato quasi tutti gli aspetti più importanti del linguaggio BASIC: qui si propone un ripasso generale e vengono fornite indicazioni sugli argomenti che tratteremo nella prossima parte del corso.
Nei linguaggi ad "alto livello", quali il BASIC, si adopera tutta una serie di comandi, che vengono automaticamente convertiti in una forma eseguibile dal computer. Un qualsiasi programma può essere scritto impiegando una combinazione dei due modelli principali, chiamati "costrutti": i costrutti "di sequenza" e le "strutture di controllo". Tra questi, gli unici che siano essenziali in BASIC sono le frasi IF...THEN...ELSE e WHILE...DO. La maggior parte degli altri linguaggi cosiddetti "evoluti" dispongono di una scelta più ampia.
I costrutti di sequenza consentono di suddividere il procedimento principale in tanti sotto-procedimenti, che vengono eseguiti, appunto, sequenzialmente. In BASIC, questi sotto-procedimenti sono costituiti da più linee di programma, nelle quali la sequenza è indicata dal numero di linea.
Pertanto, se il procedimento consiste nel moltiplicare per 10 il valore di una variabile, si potrebbe impiegare la seguente sequenza:

```
110 INPUT N
120 LET N = N * 10
130 PRINT N
```

Oltre ai costrutti di sequenza, occorrono anche le strutture di controllo, che servono per deviare il percorso dell'esecuzione (eventualmente, in base al risultato di un calcolo o di un confronto).
In BASIC, la più semplice struttura di controllo è l'istruzione GOTO, chiamata anche "salto incondizionato": quando il BASIC incontra questa istruzione, l'esecuzione salta alla linea specificata dopo la GOTO, senza che venga verificata alcuna speciale condizione. Anche l'istruzione GOSUB costituisce un salto "incondizionato", salvo che il programma, una volta eseguita la subroutine, ritorna alla linea immediatamente successiva alla GOSUB.
Un'altra struttura di controllo disponibile in BASIC è la frase IF...THEN...ELSE, che deve seguire una precisa "sintassi" (o formato):

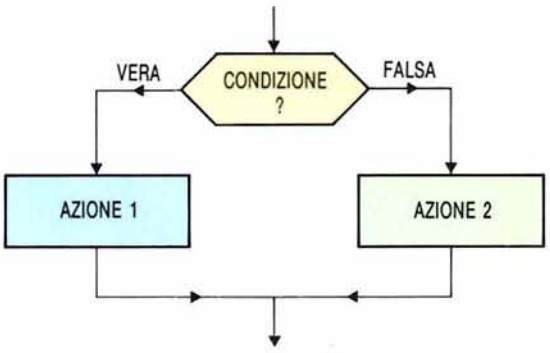
IF [condizione] THEN [azione 1] (ELSE [azione 2])

ossia:

SE [condizione] è vera, ALLORA esegui [azione 1] (ALTRIMENTI esegui [azione 2]).

L'istruzione ELSE (= altrimenti) è racchiusa tra parentesi perché non è presente, esplicitamente, in tutte le versioni di BASIC. Qualora il nostro BASIC non disponga di essa, la frase ELSE [azione 2] equivale a: ALTRIMENTI esegui la linea immediatamente seguente.
Comunque sia, la struttura IF...THEN (...ELSE) offre al BASIC la possibilità di scegliere una tra due diverse azioni da intraprendere.
Si consideri il seguente progamma, che serve per calcolare la radice quadrata di un numero immesso da tastiera, fino a quando non si digiti il valore −9999, per uscire dal programma:

```
10 PRINT "DIGITA UN NUMERO"
20 INPUT N
30 IF N = −9999 THEN GOTO 70
40 LET S = SQR (N)
50 PRINT "LA RADICE QUADRATA DI ";N;" È ";S
60 GOTO 10
70 END
```



La struttura di controllo IF...THEN...ELSE
Se la condizione è vera, viene eseguita l'AZIONE 1, se, invece, essa è falsa, viene eseguita l'AZIONE 2

La linea 30, in effetti, significa: se è vero che il numero N è uguale a −9999, allora termina il programma, altrimenti passa alla linea seguente (ossia la 40, dove viene calcolata la radice quadrata).
La struttura di controllo WHILE...DO (esegui... finché), non è disponibile in tutti i BASIC, ma può essere simulata facilmente. Questa struttura equivale alla seguente frase: «ripeti un certo gruppo di istruzioni, fino a che non sia vera la condizione specificata dopo l'istruzione WHILE». Più sinteticamente:

MENTRE [condizione] è vera ESEGUI [azione 1].

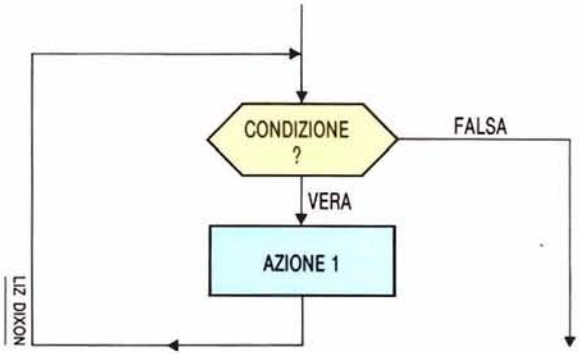
Consideriamo un semplice esempio: in un programma viene presentato il messaggio "PREMERE IL TASTO SPAZIO PER PROSEGUIRE".

Ciò può essere riassunto, a parole:

MENTRE (WHILE) lo spazio non è digitato
ESEGUI (DO) una scansione della tastiera
(resto del programma)

In BASIC:

```
250 PRINT "PREMERE IL TASTO SPAZIO PER PROSEGUIRE"
260 IF INKEY$ <  > " " THEN GOTO 260
270 (resto del programma)
```



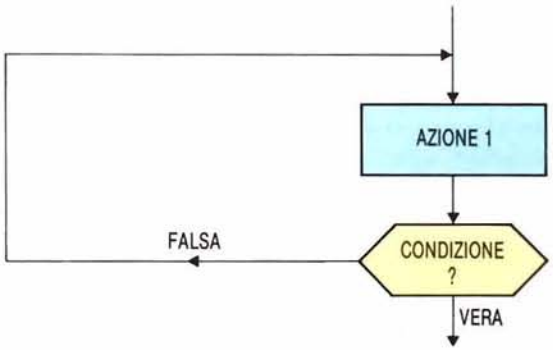
La struttura di controllo WHILE...DO
Il ciclo viene ripetuto all'infinito, finché la condizione è vera.

La linea 260 significa: SE (IF) il valore di INKEY$ è diverso (< >) da uno spazio (" "), ALLORA (THEN) torna a controllare l'input da tastiera. Un altro modo per codificare la stessa funzione, in BASIC, è:

```
250 PRINT "PREMERE IL TASTO SPAZIO PER PROSEGUIRE"
260 FOR X = 0 TO 1 STEP 0
270 IF INKEY$ = " " THEN LET X = 2
280 NEXT X
290 (resto del programma)
```

In questo frammento di programma, la scansione della tastiera avviene soltanto se non viene premuto alcun tasto (o il tasto non corrisponde al carattere spazio). In caso contrario, l'esecuzione passa alla linea 290.
Si noterà che nel ciclo FOR-NEXT è presente un elemento nuovo: STEP (passo). Se viene omessa questa istruzione, l'indice del ciclo (X nel nostro caso) viene incrementato di un'unità ad ogni passaggio: STEP serve per definire un incremento diverso. Un esempio: nel ciclo FOR Y = 1 TO 10 STEP 2, il valore della Y, al primo passaggio, sarà 1, e nei successivi passaggi, sarà 3, 5, 7 e 9. A un passaggio successivo, Y varrebbe 11, che è superiore al limite massimo di 10, per cui il ciclo termina. Possiamo anche definire dei cicli coi quali eseguire un "conto alla rovescia": FOR Y = 10 TO 1 STEP –1, causa l'indice Y ad assumere, progressivamente, i valori da 10 ad 1. L'impiego di STEP 0, nel nostro esempio, costituisce un piccolo trucco: il ciclo non avrà mai fine, a meno che il valore della X venga cambiato artificialmente (nel nostro caso all'interno della frase IF...THEN).
Un'altra struttura di controllo molto utile, facilmente simulabile in BASIC, è la "REPEAT...UNTIL" (ESEGUI...FINCHÉ). In questo caso, la condizione è collocata dopo il gruppo di istruzioni che ne formano il "corpo", cosicché esso viene eseguito almeno una volta. Consideriamo il seguente "generatore di numeri casuali":

```
10 PRINT "PREMERE IL TASTO SPAZIO"
20 FOR X = 0 TO 1 STEP 0
30 LET R = R + 1
40 IF R > 9 THEN LET R = 1
50 IF INKEY$ = " " THEN LET X = 2
60 NEXT X
70 PRINT "IL VALORE DI R È ";R
```



La struttura di controllo REPEAT...UNTIL
Il ciclo viene eseguito fino a quando la condizione è falsa ed il gruppo di istruzioni viene elaborato almeno una volta

In questo caso, il "corpo" principale (l'incremento del valore di R) viene eseguito almeno una volta, prima di controllare se la condizione (tasto spazio premuto) è vera o meno.
Un'ultima struttura di controllo, non essenziale, ma sicuramente utile, è la cosiddetta CASE (caso). Nel BASIC, viene esplicata ricorrendo alle frasi ON...GOTO oppure ON...GOSUB. Il funzionamento è molto semplice: la frase ON...GOTO permette, in una sola linea di programma, di selezionare una tra più azioni diverse, in base al valore di una variabile. Un esempio che evidenzia l'utilità di questa struttura è il seguente, nel quale, il valore numerico di una variabile viene convertito nel nome di un giorno della settimana:

```
1050 IF D = 1 THEN GOTO 2020
1060 IF D = 2 THEN GOTO 2040
1070 IF D = 3 THEN GOTO 2060
1080 IF D = 4 THEN GOTO 2080
1090 IF D = 5 THEN GOTO 3000
2000 IF D = 6 THEN GOTO 3020
2010 IF D = 7 THEN GOTO 3040
2020 PRINT "LUNEDÌ"
2030 GOTO * * * END * * *
2040 PRINT "MARTEDÌ"
2050 GOTO * * * END * * *
2060 PRINT "MERCOLEDÌ"
2070 GOTO * * * END * * *
2080 PRINT "GIOVEDÌ"
2090 GOTO * * * END * * *
3000 PRINT "VENERDÌ"
3010 GOTO * * * END * * *
3020 PRINT "SABATO"
3030 GOTO * * * END * * *
3040 PRINT "DOMENICA"
3050 GOTO * * * END * * *
```

DECIMALE	BINARIO	CARATTERE
32	00100000 =	(spazio)
33	00100001 =	!
34	00100010 =	"
35	00100011 =	#
36	00100100 =	$
37	00100101 =	%
38	00100110 =	&
39	00100111 =	'
40	00101000 =	(
41	00101001 =	)
42	00101010 =	*
43	00101011 =	+
44	00101100 =	,
45	00101101 =	-
46	00101110 =	.
47	00101111 =	/
48	00110000 =	0
49	00110001 =	1
50	00110010 =	2
51	00110011 =	3
52	00110100 =	4
53	00110101 =	5
54	00110110 =	6
55	00110111 =	7
56	00111000 =	8
57	00111001 =	9
58	00111010 =	:
59	00111011 =	;
60	00111100 =	<
61	00111101 =	=
62	00111110 =	>
63	00111111 =	?
64	01000000 =	@
65	01000001 =	A
66	01000010 =	B
67	01000011 =	C
68	01000100 =	D
69	01000101 =	E
70	01000110 =	F
71	01000111 =	G
72	01001000 =	H
73	01001001 =	I
74	01001010 =	J
75	01001011 =	K
76	01001100 =	L
77	01001101 =	M
78	01001110 =	N
79	01001111 =	O
80	01010000 =	P
81	01010001 =	Q
82	01010010 =	R
83	01010011 =	S
84	01010100 =	T
85	01010101 =	U
86	01010110 =	V
87	01010111 =	W
88	01011000 =	X
89	01011001 =	Y
90	01011010 =	Z
91	01011011 =	[
92	01011100 =	\
93	01011101 =	]
94	01011110 =	ò
95	01011111 =	_
96	01100000 =	`
97	01100001 =	a
98	01100010 =	b
99	01100011 =	c
100	01100100 =	d
101	01100101 =	e
102	01100110 =	f
103	01100111 =	g
104	01101000 =	h
105	01101001 =	i
106	01101010 =	j
107	01101011 =	k
108	01101100 =	l
109	01101101 =	m
110	01101110 =	n
111	01101111 =	o
112	01110000 =	p
113	01110001 =	q
114	01110010 =	r
115	01110011 =	s
116	01110100 =	t
117	01110101 =	u
118	01110110 =	v
119	01110111 =	w
120	01111000 =	x
121	01111001 =	y
122	01111010 =	z
123	01111011 =	{
124	01111100 =	|
125	01111101 =	}
126	01111110 =	~

Il codice ASCII
Un elenco di codici ASCII e equivalenti valori decimali e binari, compresi tra 32 e 126. Il significato dei valori non riportati varia da apparecchio ad apparecchio

Per ottenere il medesimo risultato, si può semplicemente scrivere:

```
1050 ON D GOTO 2020,2040,2060,2080,3000,3020,3040
```

Se il valore di D è 1, il programma esegue un salto alla linea 2020, mentre se D vale 2, l'esecuzione passa alla linea 2040, e così via. Potremmo adesso modificare il programma presentato a pag. 174 (lancio dei dadi) nel seguente modo:

```
REM ROUTINE DI SELEZIONE
REM USANDO ON...GOSUB
ON D GOSUB 530,610,690,760,850,930
RETURN
```

In alcune versioni di BASIC possono esserci funzioni speciali che non abbiamo descritto (ad esempio: PAINT, PAPER, INK, BEEP, ecc.): queste sono istruzioni particolari, non facenti parte del BASIC "standard" e, per lo più, connesse all'uso della grafica. La spiegazione di alcune verrà data più avanti.
Prima di chiudere questa prima parte del corso, comunque, vanno ancora chiariti alcuni particolari: il "set" di caratteri ASCII, alcune funzioni per il trattamento delle stringhe alfanumeriche ed i metodi da impiegare per creare nuove funzioni "ad hoc".
Molti sono stati i sistemi, nel corso degli anni, per codificare le lettere dell'alfabeto, i numeri e particolari simboli grafici: uno dei primi è stato l'alfabeto Morse, nel quale si impiegano combinazioni di "punti" e di "linee", ma il sistema senza dubbio più diffuso oggi è quello dell'American Standard Code for Information Interchange (Codice americano "standard" per lo scambio d'informazioni), abbreviato in ASCII.
Il codice ASCII impiega un byte per ciascuno dei 96 caratteri "stampabili" o di "controllo". In effetti gli otto bit di un byte permettono ben 256 combinazioni, molte più di quelle necessarie, cosicché se ne impiegano soltanto sette, per un totale di combinazioni pari a 128. L'ottavo bit è talvolta ignorato, ma spesso viene usato per poter rappresentare caratteri grafici o simboli speciali. Il codice ASCII, con i valori binari e decimali corrispondenti a ciascun carattere è riportato nella tabella della pagina a fianco.
Come si può notare, il codice ASCII per la lettera A equivale a 65, quello per la lettera B equivale a 66, e così via. I rispettivi caratteri, ma in minuscolo, valgono 97 e 98. Il valore di ogni carattere maiuscolo differisce da quello minuscolo di un fattore fisso 32. Ciò rende estremamente facile la conversione da maiuscolo a minuscolo e viceversa. Per far ciò in BASIC, dobbiamo impiegare due funzioni intrinseche non ancora spiegate: la ASC e la CHR$.
La funzione ASC fornisce il valore ASCII del carattere impiegato come argomento della funzione stessa, cosicché scrivere PRINT ASC ("A") visualizza sullo schermo il valore 65, mentre PRINT ASC ("b") il valore 98.
La funzione CHR$, invece, si comporta esattamente al contrario: converte il proprio argomento (numerico) nel carattere equivalente. Dunque: PRINT CHR$(65) visualizza una "A", mentre PRINT CHR$(98) una "b". Nel trattamento delle stringhe alfanumeriche le funzioni ASC e CHR$ vengono largamente impiegate assieme alle funzioni LEFT$, RIGHT$ e MID$. Ecco un piccolo programma per la conversione automatica in maiuscolo dei caratteri immessi da tastiera:

```
10 REM CONVERTITORE DA MINUSCOLO A MAIUSCOLO
20 PRINT "DIGITA UNA LETTERA"
30 INPUT C$
40 LET C = ASC (C$)
50 IF C > 90 THEN LET C = C — 32
60 PRINT CHR$(C)
```

Vedremo molte altre applicazioni di queste funzioni per il trattamento delle stringhe più avanti nel corso.
Vediamo, infine, come creare nuove funzioni "ad hoc" (o per sopperire alla mancanza di quelle finora presentate). Quasi tutti i BASIC permettono, mediante l'istruzione DEF, la creazione di nuove funzioni: qui di seguito presentiamo una funzione per calcolare il volume di una sfera (la formula matematica è $4/3\ \pi\ r^3$ dove r è il raggio della sfera e π vale 3,14159):

```
10 REM FUNZIONE PER CALCOLARE IL VOLUME DI UNA
   SFERA
20 DEF FNV (X) = 4 * 3.14159 * X * X * X/3
30 PRINT "LUNGHEZZA DEL RAGGIO"
40 INPUT R
50 PRINT "IL VOLUME DI UNA SFERA DI RAGGIO";R;" È "
60 PRINT FNV (R)
70 END
```

La funzione viene definita nella linea 20: ma esaminiamola più in dettaglio:

```
DEFinisci  identificatore della funzione
    ↓      ↓
20 DEF   FNV(X) = 4 * 3.14159 * X * X * X/3
         ↑  ↑
FuNzione   variabile "di comodo"
```

Nel definire una funzione in BASIC, le lettere FN sono seguite da un'altra lettera "identificativa" (V nel nostro caso) e da una variabile "di comodo". La variabile di comodo deve essere impiegata nell'espressione con la quale si definisce la funzione, mentre, per utilizzare la nuova funzione in altre parti del programma, si usa un valore numerico (o un'altra variabile), come alla linea 60. La nuova funzione può essere utilizzata anche così:

```
999 LET A = 66
1000 LET B = FNV(A)
1010 PRINT B
1020 LET C = 5
1030 LET D = B + FNV(C)
1040 PRINT D
1050 LET G = FNV(16)
1060 PRINT G
```

Alcuni BASIC hanno il pregio, non indifferen-

te, di permettere l'uso di più variabili in una stessa funzione, per cui una funzione per calcolare la media di due valori potrebbe essere:

```
100 DEF FNA(B,C) = (B + C)/2
110 INPUT "DIGITA DUE NUMERI";B;C
120 LET A = FNA(B,C) : REM LA FUNZIONE "MEDIA"
130 PRINT "LA MEDIA DI ";B; " E ";C; " È ";A
```

Nella linea 110 incontriamo una gradita novità: l'uso dell'istruzione INPUT anche per la visualizzazione di un messaggio racchiuso tra doppi apici. La maggioranza dei BASIC, infatti, permette questa combinazione, equivalente a:

```
110 PRINT "DIGITA DUE NUMERI";
120 INPUT A,B
```

Anche nella linea 120 abbiamo riunito in una sola linea due diverse istruzioni, separandole da un carattere ":". La maggior parte dei BASIC permette questo tipo di formato, che fa risparmiare memoria ma che, talvolta, rende i programmi meno leggibili e più soggetti ad errori.
Nella prossima parte del corso, affronteremo in maggiore dettaglio l'analisi e l'effettiva stesura di programmi, piuttosto che soffermarci sulle singole istruzioni BASIC.

Basic & Basic

ASCII
Il Dragon 32 possiede un "set" di caratteri diverso dall'ASCII, che non prevede l'uso di lettere minuscole, quindi il programma di conversione non può funzionare

DEF FN(A,B)
Sull'Oric-1, sul VIC-20, sul Dragon 32 e sul Commodore 64 non è consentito usare più di una variabile

ASC( )
Sullo Spectrum e lo ZX81 sostituire:
ASC(A$) con CODE A$ e
CHR$(65) con CHR$ 65

CHR$( )
Se l'argomento delle due funzioni è un'espressione complessa, allora le parentesi sono necessarie, altrimenti no

ON... GOSUB
Non sono disponibili sullo Spectrum, sullo ZX81 e sul Lynx.

ON... GOTO

INPUT" "
Sul computer BBC, sostituire
INPUT "(qualsiasi messaggio)";MS$
con
INPUT "(qualsiasi messaggio)",MS$

DEF FN
Sul BBC, le funzioni devono essere definite alla fine del programma, dopo la parola END o STOP e non all'inizio, come riportato nell'esempio

STEP 0
Sostituire, per l'Oric-1:
IF INKEY$ = " " THEN LET X = 2
con
IF KEY$ = " " THEN LET X = 1

Sul Dragon 32 sostituire, invece, con:
IF INKEY$ = " " THEN LET X = 1

Sul VIC-20 e sul Commodore 64 sostituire:
IF INKEY$ = " " THEN LET X = 2
con
GET A$: IF A$ = " " THEN LET X = 1

Risposte agli esercizi di pagina 197

Qualche... baco!

Otterremo un errore del tipo "OUT OF DATA", (ossia "dati insufficienti"). Infatti, i dati nella linea DATA (la 130), devono essere 12. Inoltre, c'è un secondo errore: la linea 100 deve essere:

```
100 PRINT A(X,Y)
```

altrimenti, il computer tenterà di accedere ad A(4,1), che non esiste!

Assegnazione di valori

Ecco una soluzione al problema (ma ne esistono altre!):

```
10 DIM A(8,13)
20 FOR R = 1 TO 7
30 FOR C = 1 TO 12
40 READ A(R,C)
50 NEXT C
60 NEXT R
70 REM SOMMA DEI TOTALI
80 GOSUB 300
90 REM VISUALIZZAZIONE DEI RISULTATI
100 GOSUB 200
110 PRINT "ANCORA ALTRI DATI ";
120 PRINT "(S/N)";
130 INPUT A$
140 IF A$ = "N" THEN GOTO 160
150 GOTO 100
160 END
200 PRINT "QUALE MESE ?"
210 PRINT "1  = GENNAIO, 2 = FEBBRAIO, ECC."
220 PRINT "13 = TOTALE"
230 INPUT M
240 PRINT "QUALE VOCE DI SPESA?"
250 PRINT "1 = BENZINA, ECC."
260 PRINT "8 = TOTALI"
270 INPUT X
280 PRINT "IL VALORE E' ";A(X,M)
290 RETURN
300 FOR R = 1 TO 7
310 LET T = 0
320 FOR C = 1 TO 12
330 LET T = T + A(R,C)
340 NEXT C
350 LET A(R,13) = T
360 NEXT R
370 FOR C = 1 TO 13
380 LET T = 0
390 FOR R = 1 TO 7
400 LET T = T + A(R,C)
410 NEXT R
420 LET A(8,C) = T
430 NEXT C
440 RETURN
500 REM DA QUI IN POI CI SONO GLI 84 VALORI
510 REM CONTENUTI IN FRASI 'DATA'
520 DATA 11350, 9870, ......
```

Gli accoppiatori acustici

La comunicazione tra due computer è cosa semplice con questi economici apparecchi

Collegare il computer con la stampante o l'unità a dischetti è un'operazione relativamente facile, perché, di solito, gli apparecchi sono tutti collocati in una stessa stanza, se non addirittura sulla stessa scrivania: tutt'altra cosa è collegare il computer con elaboratori più grandi situati nello stesso ufficio, oppure, dall'altra parte del mondo! Fortunatamente, esiste già un mezzo di comunicazione intercontinentale, la rete telefonica, e, collegato il computer ad essa, una parte del problema è risolto.

La rete telefonica viene già utilizzata, su larga scala, dai grossi centri di elaborazione elettronica (per esempio, dalle compagnie aeree per la prenotazione dei voli) e pertanto la tecnologia si trova già in una fase molto avanzata. Tutto ciò che occorre all'utente di un home o personal computer è una forma più economica per sfruttare questa tecnologia. Un metodo molto usato per collegare un computer alla rete telefonica è mediante il "modem".

Il funzionamento del modem è molto simile a quello dell'interfaccia per unità a nastri del microcomputer. I numeri binari 1 e 0 vengono convertiti in segnali elettrici aventi due diverse frequenze audio. Queste vengono inviate lungo la linea telefonica (processo di modulazione): all'altro capo del filo, le frequenze audio vengono riconvertite in valori binari 1 e 0 (processo di demodulazione). Il modem emette una frequenza costante (chiamata frequenza "portante") sia durante la trasmissione che durante la ricezione. La "portante" serve anche per indicare che il computer è ancora in linea e non ha "riagganciato".

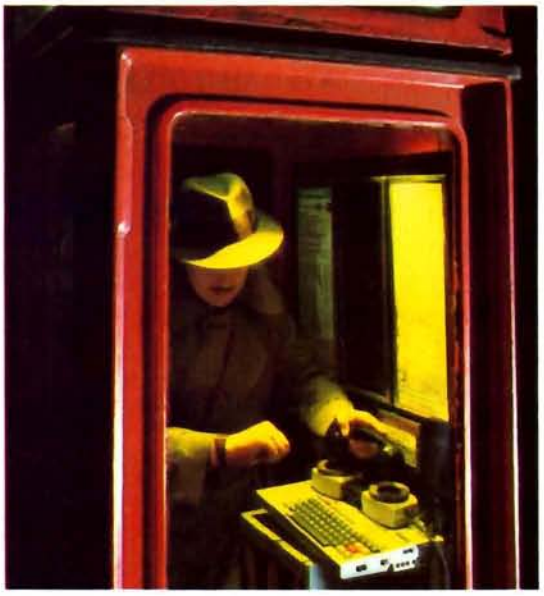
Lo svantaggio più grosso, particolarmente per l'utente di home computer, è la necessità di collegare in modo permanente il modem alla rete telefonica, monopolizzando così l'apparecchio di casa.

In alternativa, si può impiegare un accoppiatore acustico. Il sistema è basato sull'invio di frequenze audio, riprodotte su di un altoparlante da accoppiare alla cornetta del telefono. All'altro capo della linea, un microfono collegato alla parte "ricevente" della cornetta, capta il segnale trasmesso. Diversamente da quanto avviene con il modem, non è necessario tenere l'accoppiatore acustico collegato al telefono in modo permanente.

Ci sono diversi tipi di accoppiatori acustici: da strumenti simili a quello illustrato qui di fianco (di dimensioni così ridotte da essere utilizzabile con un computer portatile) a versioni grandi "da scrivania". Le versioni più sofisticate sono in grado di rispondere automaticamente alle chiamate, senza richiedere l'intervento di un operatore, riconoscendo semplicemente la frequenza "portante" dell'apparecchio chiamante. Come nell'interfaccia per registratore, i dati possono essere trasmessi a velocità diverse, ma non è consigliabile superare i 1200 caratteri al secondo (cps), perché ciò potrebbe pregiudicare l'affidabilità del sistema.

Gli strumenti più economici consentono una trasmissione/ricezione piuttosto lenta (anche soli 30 cps), mentre su quelli più costosi è possibile selezionare la velocità. È comunque indispensabile che lo strumento trasmittente e quello ricevente lavorino alla stessa velocità, altrimenti un qualsiasi scambio di dati diviene impossibile.

La notevole diffusione di piccoli computer, negli ultimi anni, ha dato incremento alla produzione di molti strumenti nuovi. Ne esistono alcuni (quali il "Sendata") che permettono a chiunque (in particolar modo a giornalisti o rappresentanti) di utilizzare un computer portatile (quali il Modello 100 della Tandy e lo HX-20 della Epson) come terminali remoti. Ciò permette di memorizzare sul computer una serie di ordini o l'articolo per un giornale e trasmetterli alla sede centrale tramite le linee telefoniche.



MARCUS WILSON-SMITH

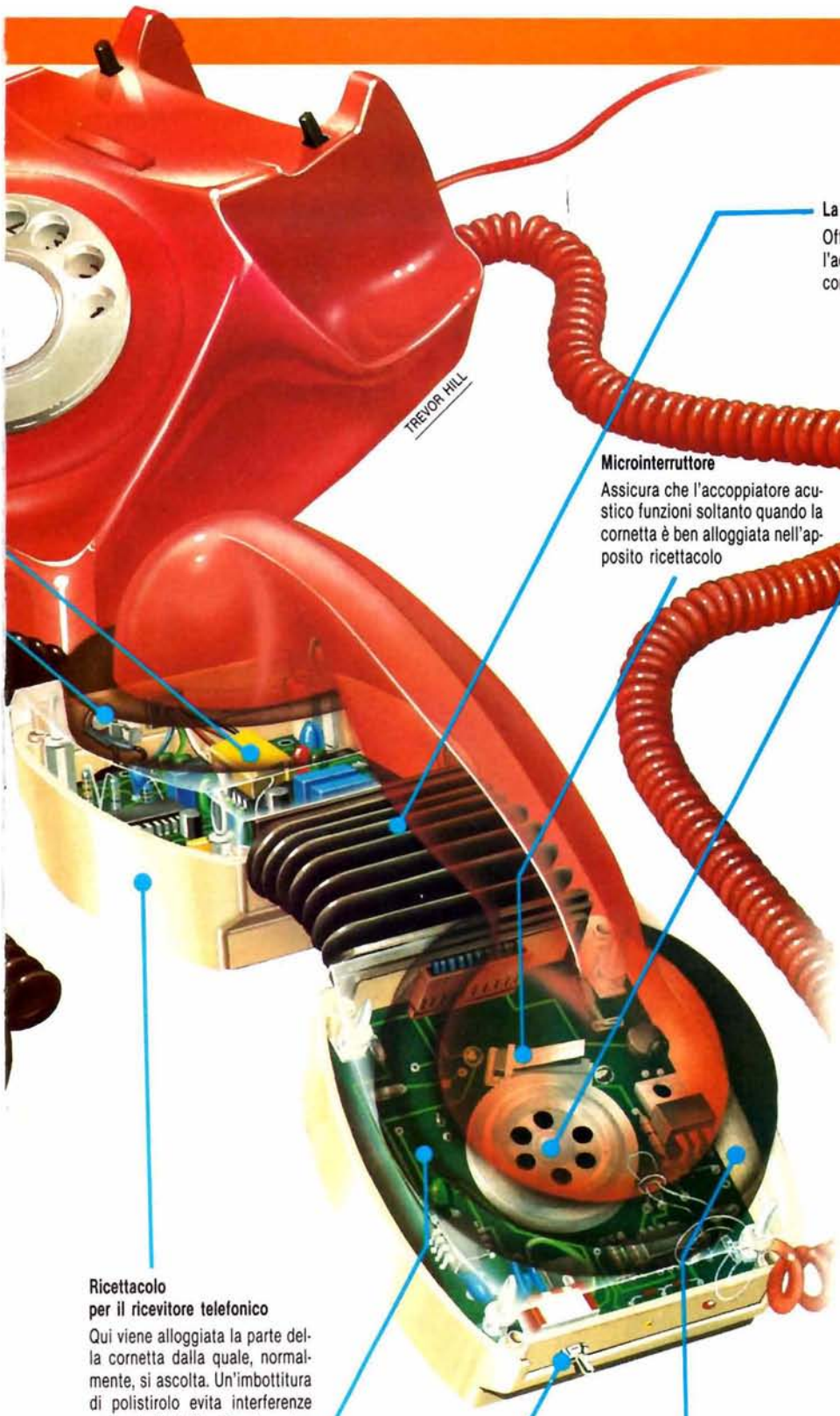


Il microfono
Serve a captare il segnale emesso dal ricevitore della cornetta telefonica, ed a trasferirlo al circuito di decodifica

Presa di alimentazione
La corrente, necessaria al funzionamento dell'accoppiatore ed alla ricarica degli accumulatori incorporati, è fornita da un apposito trasformatore

Cavo per l'interfaccia
Serve per il collegamento all'interfaccia seriale RS232 del computer

La cabina telefonica
Un accoppiatore acustico, sfruttando la rete telefonica pubblica, permette all'utente di un computer portatile di comunicare con un qualsiasi altro computer, situato anche da un'altra parte del mondo



La giuntura flessibile
Offre la possibilità di adattare l'accoppiatore alle più svariate cornette telefoniche

Microinterruttore
Assicura che l'accoppiatore acustico funzioni soltanto quando la cornetta è ben alloggiata nell'apposito ricettacolo

L'altoparlante
L'accoppiatore acustico funziona in maniera complementare alla cornetta: l'altoparlante emette segnali, captati dal microfono della cornetta

Ricettacolo per il ricevitore telefonico
Qui viene alloggiata la parte della cornetta dalla quale, normalmente, si ascolta. Un'imbottitura di polistirolo evita interferenze provenienti dall'ambiente esterno

Circuito elettronico
Questi componenti elettronici controllano lo scambio di dati tra l'interfaccia ed il microcomputer, convertendo, allo stesso tempo, i segnali binari in due frequenze diverse

Selettore di funzionamento
Questo interruttore determina se l'accoppiatore acustico deve trasmettere o ricevere

Microfono telefonico
Questa parte della cornetta viene alloggiata nell'apposito ricettacolo

Nel mondo commerciale, un computer collegato ad un accoppiatore acustico offre l'accesso ad una vasta rete di servizi informativi e centrali di elaborazione (computer bureaux). In Gran Bretagna una catena di farmacie utilizza un sistema completamente computerizzato per le ordinazioni e la fornitura dei prodotti: il personale indica la quantità e gli articoli mancanti e trasmette la richiesta al centro di elaborazione del magazzino.

Per quanto riguarda l'utente "casalingo", l'accoppiatore acustico ha il vantaggio non indifferente rispetto al modem di non occupare in modo permanente la linea telefonica. In questo modo, un dirigente, che lavori in casa propria, può inviare o ricevere informazioni per mezzo di esso, pur continuando ad usufruire normalmente del telefono.

Sempre nel campo dei piccoli computer, l'accoppiatore acustico sta diventando un'alternativa utile e economica al modem, per permettere all'utente l'accesso a banche dati pubbliche o a servizi "postali" elettronici: inviare, via telefono, un messaggio (o un intero programma) ad un amico è molto più sicuro che affidarne le sorti al servizio postale tradizionale. Il servizio di posta elettronica, già funzionante in molti paesi, offre un nuovo genere di comunicazione, che, finora, era esclusivamente riservato agli utenti di grossi computer.

Tuttavia, anche in questo settore s'incontra l'ostacolo degli "standard": non è possibile far comunicare un accoppiatore acustico europeo con uno americano, per esempio, perché negli USA viene adoperato il sistema di comunicazione BELL 103, mentre in Europa è in uso il CCITT V21, e questi sono incompatibili tra loro.

Inoltre, non sempre le cornette degli apparecchi telefonici di costruzione europea si adattano agli accoppiatori acustici fabbricati in altre nazioni: ciò può causare interferenze tali da rendere impossibile qualsiasi comunicazione.

I network

I sistemi che permettono di distribuire le informazioni a più utenti

Oggi che i microcomputer si stanno diffondendo sempre di più, sia nelle scuole che negli uffici, maggiori sono le probabilità che nello stesso edificio, o addirittura sullo stesso tavolo, si trovino più apparecchi compatibili fra di loro. In queste situazioni, viene spontaneo chiedersi se non sia possibile collegare i vari apparecchi tra loro, magari per usufruire in comune delle periferiche (unità a disco o stampanti). Infatti, accessori di questo genere sono piuttosto costosi e da un punto di vista economico conviene poterli sfruttare al massimo delle capacità.

L'utilizzo di strumenti periferici da parte di più utenti non è l'unico vantaggio che deriva dal collegamento di più macchine: si possono collegare assieme un certo numero di microcomputer per formare un "network" (rete). Se tutte le macchine o "workstations" (stazioni di lavoro) sono situate in un unico edificio, si parla di "rete locale" (Local Area Network, abbreviata in "LAN"). Fiumi di inchiostro sono stati consumati a proposito dell'abbandono dell'uso di carta negli uffici ed il primo passo verso questo obiettivo è di sostituire i sistemi di comunicazione interna (memorandum cartacei) con messaggi depositati nella memoria centrale del computer, leggibili sullo schermo del terminale. Questa specie di "servizio postale interno" è disponibile nella maggioranza delle "reti locali". Invece di interrompere il lavoro dell'operatore, sull'ultima riga del suo schermo appare l'avviso che c'è un messaggio in attesa di essere letto.

Un altro settore, per applicazioni di questo genere, è la scuola. Il testo da studiare viene diramato, dal microcomputer dell'insegnante, sul terminale di ciascun studente, oppure, l'insegnante ha la possibilità di esaminare e commentare il lavoro fatto da uno qualsiasi degli studenti: il tutto rimanendo seduto in cattedra (o addirittura da un'altra stanza dell'edificio).

Inoltre, c'è la possibilità di creare una centrale informativa, in comune fra più utenti. Forse l'applicazione più comune è la banca dati, dove vengono inserite informazioni di interesse più o meno pubblico (come il Videotel o il Televideo).

Quando più persone lavorano sullo stesso archivio di dati, occorre assicurarsi, però, che i dati in esso contenuti non vengano modificati senza che tutti gli utenti vengano informati del cambiamento.



TONY SLEEP

Un Apple per insegnante

In molti centri per la formazione tecnologica, in Gran Bretagna, vengono utilizzate reti di distribuzione come quella illustrata nella foto, dove i giovani studenti si addestrano nell'uso di computer

Per esempio, una fabbrica potrebbe impiegare un "network" di computer per informarsi sulla disponibilità di componenti o materie grezze ed è di estrema importanza che ad ogni utente vengano presentate informazioni sempre corrispondenti alla realtà: in primo luogo per evitare confusione, ma soprattutto per non vendere articoli già esauriti o per non falsare i dati di produzione.

Quando si usa un computer con una RAM sufficientemente capiente, la tentazione di "appoggiare" una parte dei dati nella memoria locale (slacciandosi quindi da quella comune a tutti gli utenti) è sempre molto forte, ma può generare tutta una serie di grossi problemi, qualora il software che gestisce l'intero sistema non sia in grado, poi, di rilevare le modifiche introdotte nelle informazioni "catturate" ed elaborate dal singolo terminale.

Attualmente il costo dell'hardware è in diminuzione, ma gli apparecchi periferici, spesso molto sofisticati, talvolta costano più dello stesso computer. Un buon esempio è costituito dai dischi tipo "Winchester": dischi rigidi, sigillati in un contenitore ermetico, che offrono una grande rapidità di accesso alle informazioni, paragonabile a quella dei computer molto più costosi. Il nome "Winchester" deriva, scherzosamente, dal nome del fucile a ripetizione Winchester 30/30, in quanto, originariamente, era costituito da due dischi "gemelli", ciascuno con una capacità di 30

megabyte. Nel software commerciale, dove è necessario elaborare un gran numero di transazioni, i dischi Winchester sono veramente utili. Inoltre, essi hanno una capacità tale da permettere a più utenti di accedere contemporaneamente. Analogamente, se stampanti a matrice di punti (vedere a pag. 74) vanno benissimo per la maggioranza delle applicazioni, nell'elaborazione di testi sorge spesso la necessità di utilizzare una stampante di qualità superiore ("a margherita", per esempio), molto più costosa. Quando è possibile, conviene far sì che più utenti possano utilizzare apparecchi simili.

I "network" offrono anche altre possibilità: ad esempio, far visionare documenti a più persone, senza il bisogno di stamparli su carta. Questo avviene già negli uffici più all'avanguardia, come le sedi di giornali o di riviste: il giornalista prepara il testo di un articolo e lo presenta, via computer, al Direttore per l'approvazione.

Successivamente, il testo magari passa al caporedattore, (sempre mediante computer), il quale provvede alla correzione di eventuali errori e fornisce indicazioni per quanto riguarda la composizione tipografica: da qui l'articolo passa alla tipografia per la stampa.

Fino a poco tempo fa, prima della costituzione di "reti locali", l'articolo da stampare doveva spesso venir corretto e ribattuto tre o quattro volte.

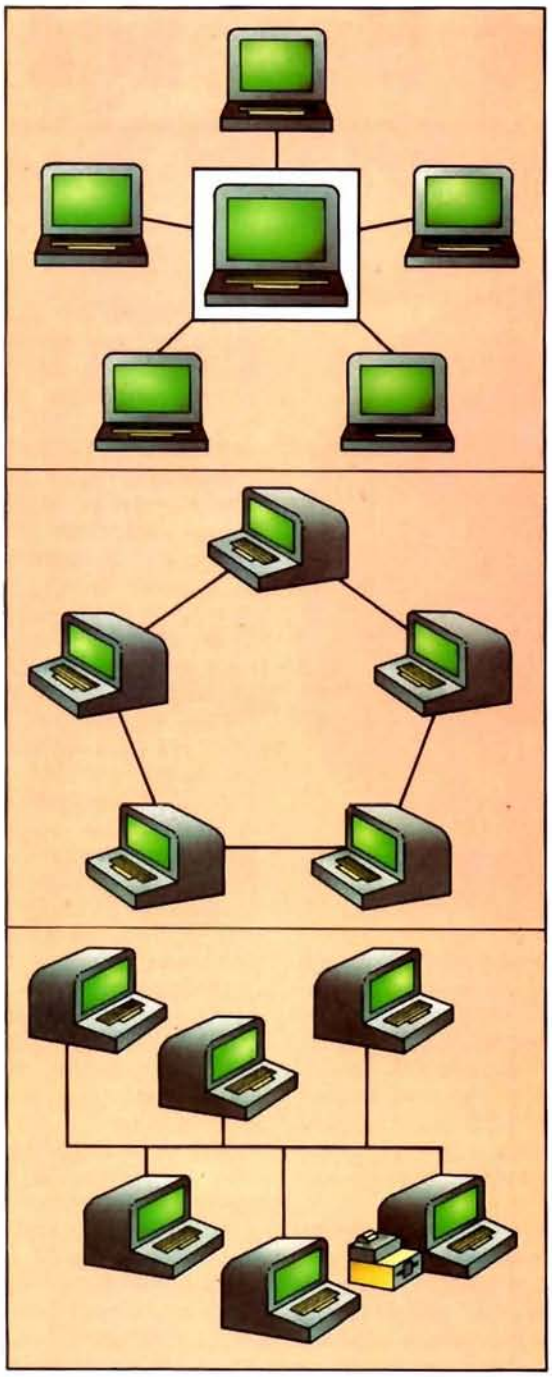
Ci sono vari sistemi "network" adatti a molti home computer, ma quello più diffuso è probabilmente l'Econet prodotto dalla Acorn per i microcomputer della BBC. Ad una delle macchine comprese nel "network" viene affidato il compito di fungere da "gestore del sistema", il che significa che essa si occupa del funzionamento dell'unità a dischi centrale e gestisce tutte le richieste di informazione. Questa unità, quando non è adibita alla funzione appena descritta, può essere impiegata come normale computer "a sé stante". Quando una sola stampante è comune a tutto il network, è necessario un apparecchio che ne controlli l'uso.

Il sistema Econet è in grado di servire fino a 254 "stazioni di lavoro", più le due stazioni di "gestione". Tuttavia, esiste una certa limitazione nell'impiego di simili "reti": in genere la distanza massima fra il "clock unit" (unità di sincronizzazione) ed i vari computer è di soli 500 metri. Il "clock unit" è l'apparecchio che sovrintende alla sincronizzazione dei dati in circolazione nel network. Il sistema Econet usa, per gli allacciamenti, due coppie di cavi e pertanto l'installazione è relativamente facile. Una coppia di cavi serve per trasportare i dati, mentre l'altra viene utilizzata per gli impulsi di sincronizzazione.

Il software piuttosto sofisticato del sistema Econet è contenuto in 8 Kbyte di EPROM, presenti in ciascun terminale. Il compito più complesso del sistema è quello di prevenire la collisione di dati: in altre parole, esso deve assicurare che non più di un solo utente della rete, alla volta, trasmetta dati. Altre marche di network per home computer presentano caratteristiche simili, ma non sono, generalmente, così complesse.

Le reti informative basate su microcomputer e su grossi calcolatori sono diventate assai diffuse negli ultimi dieci anni e non sono necessariamente limitate a scambi d'informazioni all'interno di una sola nazione: molte linee aeree, per esempio, adoperano simili sistemi, estesi ad ogni continente, per la prenotazione e l'acquisto di biglietti. In questi casi, i dati vengono trasmessi tramite le linee telefoniche o via satellite.

Con la tendenza attuale verso il sistema televisivo via cavo, si può ragionevolmente prevedere un forte incremento nell'uso di reti informative computerizzate collegate, forse, a sistemi informativi di massa tipo "Prestel" o "Videotel".



KEVIN JONES

ALCUNI TIPI DI NETWORK

A stella

Un network disposto "a stella" collega tutte le macchine con quella centrale, di controllo, che regola anche l'uso degli apparecchi periferici

Ad anello

In alcuni network i computer dei vari utenti sono collegati fra di loro in un circuito continuo. Questo tipo di network non incontra molto consenso perché i dati devono passare attraverso un certo numero di macchine (a volte più della metà del network) prima di arrivare a destinazione

A bus

La composizione di un network più sofisticato, come quello della Econet, è molto simile, concettualmente, all'architettura di un microcomputer moderno: i dati e i messaggi sono trasmessi direttamente da un utente ad un altro

Charles Babbage

Un eccentrico personaggio del XIX secolo, che ha gettato le basi della rivoluzione computerizzata



«Come vorrei che questi calcoli fossero eseguiti a vapore!». Così esclamò Charles Babbage mentre lavorava alla preparazione delle tabelle per l'Almanacco Nautico. I motori a vapore erano stati inventati nel XIX secolo, ma la navigazione in mare poneva ancora molti problemi. Le navi stabilivano la propria posizione in base alla Luna e facendo riferimento a tabelle matematiche, spesso non tanto accurate.

Nel 1812 a Babbage venne l'idea di costruire una macchina, che fu denominata "macchina differenziale", in grado di eseguire i lunghi ed elaborati calcoli necessari alla preparazione delle tabelle alle quali stava lavorando. Nel 1823 era già completata la costruzione di un piccolo prototipo e Babbage presentò al governo britannico una richiesta di finanziamento per la costruzione della macchina. Gli furono concesse 1.500 lire sterline (somma considerevole a quei tempi) e Babbage si dedicò con fervore alla realizzazione di una macchina capace di eseguire calcoli senza il minimo errore, in grado di stampare automaticamente i risultati.

Babbage impiegò il resto della sua vita a questa impresa colossale: occorrevano grandissime somme di denaro, perché il progetto comportava l'impiego di tecnologie avveniristiche per quei tempi. Fu con l'aiuto del suo amico, il Duca di Wellington (che fu anche primo ministro) che egli riuscì ad ottenere i finanziamenti necessari.

Nonostante la sicurezza, più volte espressa da Babbage, che «qualsiasi cosa facesse, la macchina non avrebbe mai commesso errori», il governo, ad un certo punto, ritirò il proprio appoggio, non prima di aver speso ben 17.000 lire sterline nell'impresa. Anche il collaboratore di Babbage, l'ingegnere Joseph Clement, si ritirò dal progetto, a seguito di una discussione sorta tra i due, portando via con sé tutti gli strumenti progettati specificamente per la costruzione della macchina.

Comunque, Babbage non si dette per vinto e passò ben presto ad un altro progetto: quello della "macchina analitica", che avrebbe dovuto svolgere tutte le funzioni della "macchina delle differenze" e anche molte di più. Per diversi aspetti questa macchina assomigliava ai moderni computer: era dotata di memorie, di un "cervello" meccanico (assimilabile alla CPU), era predisposta per la stampa dei risultati ed era perfino programmabile!

Nella prima versione, le istruzioni erano controllate da un sistema basato sull'impiego di piccoli chiodi (come avviene nei carillon), ma più tardi venne usato un sistema di schede perforate, simili a quelle adoperate da Joseph Jacquard per l'industria tessile. Babbage fece degli esperimenti usando basi numeriche diverse, ma risultò che non si traeva nessun vantaggio dall'uso del sistema binario, poiché le macchine erano completamente meccaniche.

Babbage fu assistito, nel suo progetto dalla sua compagna, la contessa Ada Lovelace, anch'essa un genio matematico. Purtroppo, i due erano sommersi da problemi, non ultimo la mancanza di denaro. Ada Lovelace perse gran parte delle proprie sostanze scommettendo sui cavalli con quello che usava definire un sistema "infallibile": dopo la sua morte, a soli 36 anni, Babbage continuò il proprio lavoro da solo.

Fu un uomo estremamente energico: inventò l'oftalmoscopio (per esaminare l'interno dell'occhio), curò la coreografia di un balletto, progettò un sistema per l'illuminazione del palcoscenico e perfino un sistema per la segnalazione in mare.

Negli ultimi anni della sua vita divenne molto irascibile: arrivò perfino a rifiutare un'onorificenza, offertagli dal governo britannico, perché se ne aspettava una più alta.

L'operato di Babbage gettò le basi della struttura dell'elaboratore elettronico moderno, ma egli non fu in grado di portare a termine i suoi progetti e la sua "macchina analitica" rimase incompiuta, principalmente a causa delle limitazioni tecniche dell'ingegneria del XIX secolo.

1792
Babbage nasce a Totnes, nel Devonshire, il 26 dicembre

1810
Entra nel Trinity College, Cambridge, per studiare matematica

1814
Si sposa con Georgina Whitemore

1822
Pubblica una recensione intitolata "Osservazioni sulle applicazioni di macchine per computare tabelle matematiche". Riceve la sua prima medaglia d'oro dalla Società Astronomica, di cui fu uno dei fondatori

1827
L'Università di Cambridge lo nomina Professore Lucasiano (la cattedra già assegnata a Newton) con uno stipendio annuo di 80 sterline: egli non insegnò mai e non prese mai la residenza all'Università

1833
Si presenta candidato alle elezioni parlamentari

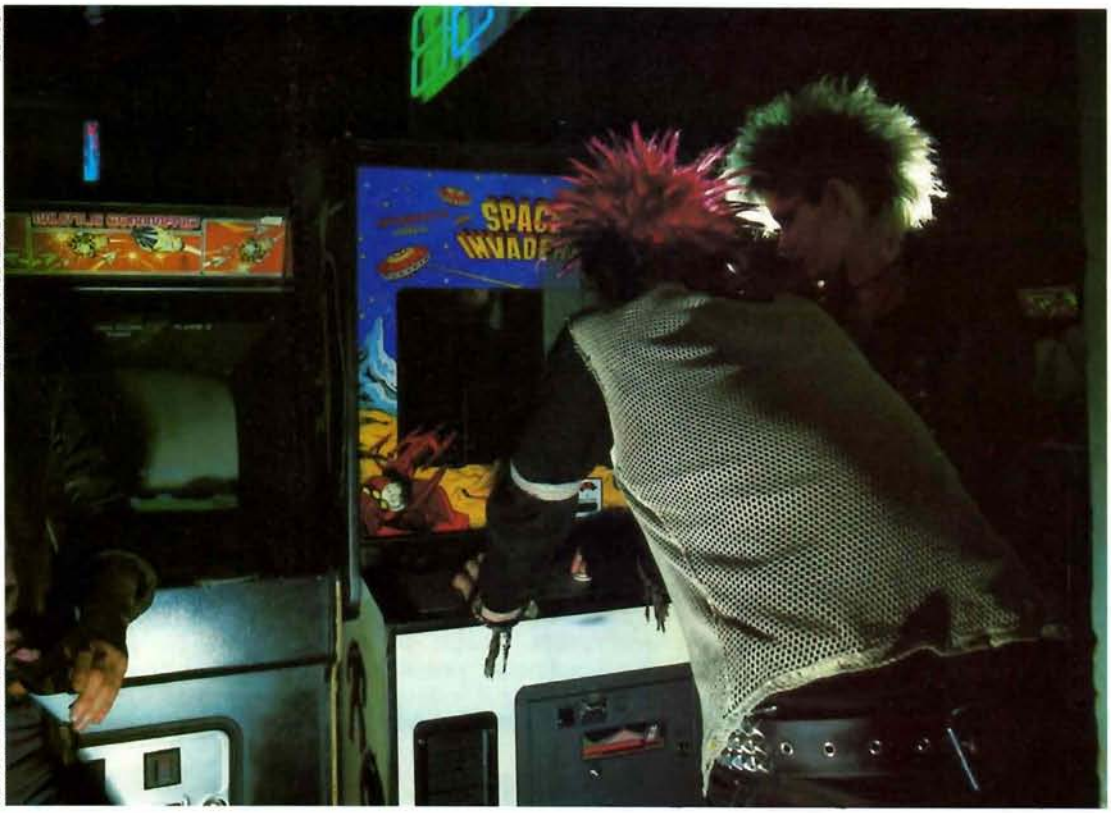
1834
Comincia il progetto per la "macchina differenziale", più tardi sospeso per l'abbandono da parte del suo assistente l'ingegnere Joseph Clement

1862
Costruisce parzialmente la "macchina differenziale", custodita oggi in un museo di South Kensington, Londra

1871
Muore il 18 ottobre

Avversario: il computer

L'evoluzione dei giochi per computer è stata molto rapida e propone, oggi, un'incredibile serie di novità



ARCADE per concessione della RUFFLER & DEITH LTD.

MARCUS WILSON-SMITH

Nelle sale giochi ed in tantissimi bar si stanno diffondendo videogiochi sofisticatissimi: emozioni "mozzafiato" per poche centinaia di lire...

Nel 1971, un giovane di nome Nolan Bushnell si dette molto da fare per persuadere i proprietari di locali e bar frequentati da giovani nella zona di Sunnyvale, California, dove abitava, a provare un nuovo tipo di gioco, da lui inventato. Il gioco era dotato di una gettoniera (il che significava un guadagno immediato per chi lo installasse nel proprio locale), ma l'aspetto più interessante era il principio di funzionamento, basato su un concetto totalmente nuovo, adesso conosciuto con il nome di "televisione interattiva".
Finalmente, il proprietario di uno dei bar acconsentì all'esperimento, ma dopo due giorni telefonò a Bushnell lamentandosi che il gioco non funzionava. Quando Bushnell arrivò sul posto scoprì che il difetto era molto semplice: la gettoniera era completamente piena!
Il *Pong* (il cui nome deriva dal gioco del ping-pong) è stato il precursore di tutti gli emozionanti giochi elettronici in voga oggi nelle sale giochi di tutto il mondo. È interessante notare che Bushnell, anche se era stato il primo ad intuire che giochi come il ping-pong potevano essere simulati con un computer, non riuscì ad afferrare del tutto la potenzialità della sua innovazione: il gioco si basava ancora sulla lotta tra due giocatori, l'uno contro l'altro, invece di avere come avversaria la sola macchina.
Stranamente, trascorsero diversi anni prima che questa nuova generazione di giochi elettronici si diffondesse: ciò avvenne nel 1977, quando la ditta giapponese Taito lanciò sul mercato lo *Space Invaders*, un gioco che ottenne subito un enorme successo.
Nello *Space Invaders*, il giocatore manovra una piccola astronave (riprodotta sulla parte inferiore dello schermo), sparando contro le astronavi nemiche che avanzano verso la sua postazione. All'occorrenza, la piccola astronave può ripararsi dietro appositi scudi "protettivi", per evitare i tiri del nemico. Il ritmo di avanzamento delle astronavi nemiche è prevedibile e viene scandito da un suono elettronico bitonale. Tuttavia, questa breve spiegazione a parole non rende bene l'idea di quanto sia irresistibile l'effetto globale del gioco.
La prima versione introdotta sul mercato utilizzava un monitor monocromo ed una grafica "a scansione", ma, in quanto ad hardware e software, non apportava alcuna innovazione di rilievo, sebbene la sua diffusione abbia generato una rivoluzione sociale paragonabile a quella dovuta alla nascita del cinema. L'hardware dello *Space Invaders* (e

giochi simili) non era molto diverso da quello degli home computer di quel periodo ed i fabbricanti di questi ultimi non tardarono ad accorgersi del nuovo mercato venutosi a creare. Le tendenze generali mutarono rapidamente: il computer poteva essere un mezzo di svago e non venir impiegato esclusivamente per l'apprendimento della programmazione.

Anche oggi, a distanza di anni, gli home computer si vendono soprattutto grazie alle loro capacità di divertire, anche se bisogna riconoscere che l'ultima generazione di giochi elettronici da sala giochi ha decisamente superato le capacità dei normali home computer. Infatti, è abbastanza normale per i giochi più evoluti disporre di una memoria da un milione di byte e di avere una grafica altamente sofisticata, paragonabile a quella degli apparecchi professionali, usati da ingegneri, architetti e progettisti.

Molte innovazioni nel campo degli home computer sono da attribuirsi ai videogiochi da sala giochi: la tendenza ad impiegare microprocessori da 16 (e perfino 32) bit deriva



dalla necessità di indirizzare più di 64 Kbyte, e da quella di ottenere una maggiore velocità di elaborazione. I produttori di videogiochi da "sala" sono stati fra i primi ad utilizzare microprocessori a 16 bit, quali il 68000 della Motorola o lo 8086 della Intel.

Una delle principali innovazioni, introdotta successivamente anche negli home computer più economici, è stata, ad esempio, la grafica "a sprite" (vedere a pag. 152). Anche alcuni chip speciali, per la generazione di immagini grafiche e di effetti sonori, quali il chip d'interfaccia video ed audio della Commodore, hanno percorso la stessa strada e così pure tre chip usati dall'Atari nella gamma 400 e 800.

L'Atari, che deve la sua fortuna al gioco inventato da Nolan Bushnell, rappresenta uno dei migliori esempi di come gli sviluppi tecnologici nel videogioco abbiano portato beneficio anche agli home computer. L'Atari fu la prima ditta a considerare l'home computer innanzitutto come un mezzo di divertimento (su ciò può aver influito, forse, il fatto che la ditta appartiene alla Warner Bros.): oltre ai microcomputer, l'Atari produce anche un apparecchio esclusivamente dedicato ai videogiochi, denominato VCS (Video Cartridge System), in grado di riprodurre, virtualmente inalterati, la maggior parte dei giochi in voga nelle sale giochi. Naturalmente, il fatto che l'Atari è uno dei produttori principali di giochi da sala è stato di grande vantaggio. Comunque ci sono altre ditte che, pur avendo interessi prioritari in altri campi, si occupano per una buona parte della produzione di videogiochi, per esempio la Columbia Broadcasting System e la Mattel, che hanno brevettato molti giochi per il mercato "domestico" da usare con apparecchi di loro produzione, rispettivamente il Colecovision e l'Intellivision.

Il videogioco, sia a livello di sala giochi sia domestico, sta rapidamente diventando una vera industria a sé stante, per certi versi simile a quella discografica: esiste perfino una specie di graduatoria dei 20 giochi più venduti. Altro elemento in comune è che spesso i giovani si scambiano tra loro i giochi, talvolta riproducendoli abusivamente.

Abbiamo già accennato al fatto che l'hardware degli apparecchi da sala giochi è decisamente superiore a quello esistente su home computer. La più recente innovazione è l'uso di un videodisco che permette di scegliere e cambiare lo sfondo del monitor. Al momento attuale, non si può prevedere se raffinatezze di questo tipo arriveranno mai sul mercato del home computer. La tecnologia esiste ed è abbastanza economica, anche se bisogna riconoscere che il software per videogiochi del tipo domestico (sia per il videogioco semplice sia per il microcomputer) non ha mai raggiunto il livello tecnologico che si riscontra su giochi da sala nella forma originale.

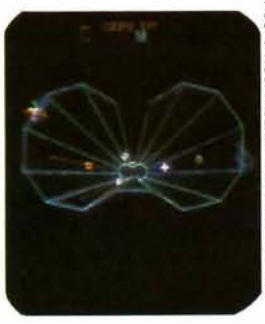
Nell'articolo dedicato alla simulazione del volo (vedere pag. 201), abbiamo notato che tutti i giochi presenti nelle sale giochi sono in qualche modo delle simulazioni, o di una situazione "vera" (come il ping-pong o una corsa automobilistica) o di una situazione immaginaria (come nel caso dello *Space Invaders*, *PacMan* e *Frogger*). Nei 12 anni dell'esistenza del videogioco, queste due correnti si

CAMPI DA GIOCO

I videogiochi da sala giochi, quale lo "Space Invaders", in cui il giocatore può spostare la propria astronave soltanto lungo una linea fissa, sono stati superati da giochi del tipo "acchiappino nel labirinto", dei quali uno molto diffuso è "PacMan". Inizialmente, le immagini venivano create con la grafica "a sprite" ed erano poco nitide. La tendenza odierna è di produrre oggetti di tipo "astratto", come nel caso del gioco "Battlezone": un giocatore "futuristico" che combatte contro carri armati e missili, oppure nel "Tempest" che si avvale di splendide immagini grafiche con effetto tridimensionale







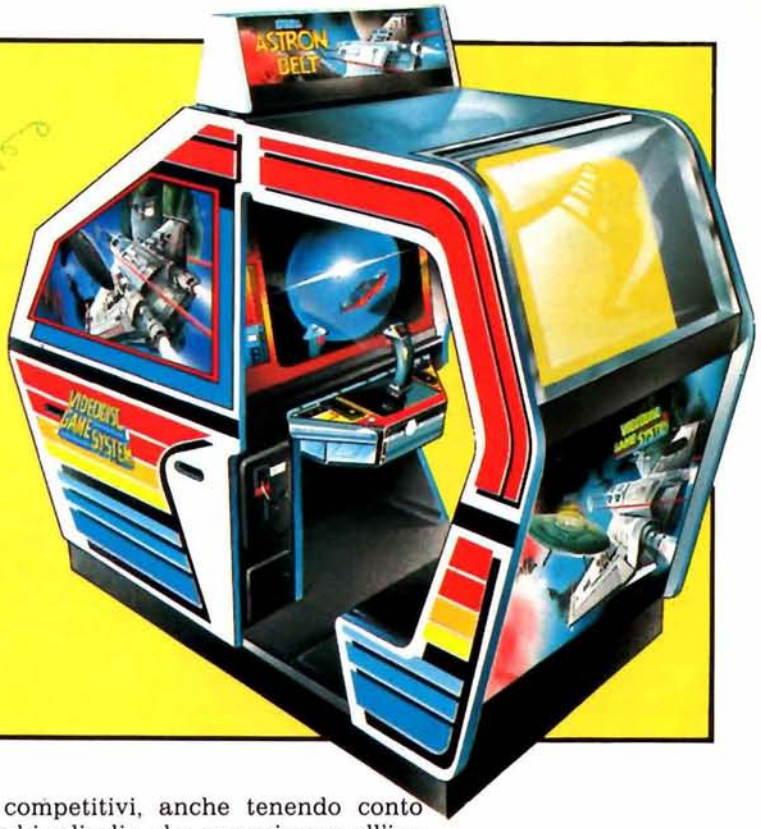
per concessione della HUTCHINSON & CO LTD.

LA NAVICELLA SPAZIALE

Il gioco "Astron Belt" è un tipico esempio della nuovissima generazione di giochi da sala giochi in cui vengono adoperati dischi laser per produrre uno sfondo semovente. Poiché il disco è impiegato in un sistema ad "accesso diretto", è possibile spostarsi con facilità da una scena ad un'altra (come nel caso di uno scontro con una navicella nemica, che risulta in un'esplosione di colori). Per il giocatore ciò costituisce un enorme passo in avanti verso il realismo. Le immagini presenti sul disco laser possono essere rappresentazioni vere oppure delle animazioni generate dal computer. L'"Astron Belt" è dotato anche di effetti sonori stereofonici e perfino di un sedile che vibra in concomitanza con le basse frequenze del circuito audio



TONY LODGE



sono sviluppate parallelamente, anche se i giochi "fantasiosi" sono quelli più diffusi.
Può essere interessante esaminare gli sviluppi derivati dai due giochi "pionieri", ossia il *Pong* e lo *Space Invaders*. Il primo gioco del tipo "a racchetta", nel quale la macchina rappresenta l'avversario, fu il *Breakout*: l'obiettivo è quello di abbattere un muro di mattoni (riprodotto sullo schermo, naturalmente!) facendovi rimbalzare una "pallina". Il giocatore deve essere abile nel manovrare la propria racchetta e tentare di colpire tutti i mattoni, per farli "scomparire". In seguito, sono arrivati altri giochi di simulazione: il golf, i biliardi e le bocce. In linea di massima, maggiore è l'accuratezza del programmatore, nel replicare fenomeni naturali e esistenti nella realtà (come per esempio la forza di gravità, quella del vento o il semplice attrito tra superfici), più entusiasmante diviene il gioco.
Al contrario, nei giochi "fantasiosi", questi criteri non hanno senso: il giocatore deve adattarsi alle regole di mondi immaginari, creati dalla fantasia del programmatore. Il passo successivo, dopo lo *Space Invaders*, è stato quello di introdurre più generi di nemici, con diversi metodi di attacco e ad intervalli irregolari. L'ultima variazione introdotta in questo tipo di giochi è stata quella di rendere mobile la "postazione" del giocatore, come avviene, per esempio, in *Defender*, considerato il migliore gioco di questo genere.
Comunque, anche i giochi "a postazione fissa" (come lo *Space Invaders*), sono progrediti, mutandosi in *Missile Command* e derivati. L'obiettivo, in questi casi, è la difesa della propria base contro l'attacco da parte di missili balistici intercontinentali.
I giochi del tipo "a labirinto", come *PacMan*, sono derivati invece dalle corse automobilistiche elettroniche dove, però, mancavano elementi competitivi, anche tenendo conto delle macchie di olio che apparivano all'improvviso sull'asfalto. Da semplici corse automobilistiche, nelle quali l'avversario principale è il tempo, si è passati a giochi elettronici che si effettuano sul tipo del gioco infantile dell'"acchiappino" fra più persone.
Le corse automobilistiche elettroniche sono anch'esse migliorate: è ora possibile ottenere una rappresentazione tridimensionale, sempre in movimento, del percorso, visto dal posto di guida, con davanti a sé il percorso che cambia continuamente. Un metodo molto simile viene adoperato nei più sofisticati giochi di simulazione di volo.
Infine, anche giochi tradizionali, quali gli scacchi, la dama ed il backgammon possiedono una loro trasposizione elettronica. In genere, questi giochi si trovano quasi esclusivamente su home computer, perché la durata del gioco è piuttosto lunga e anche perché la grafica è molto meno importante degli algoritmi impiegati per il gioco.
L'unica vera innovazione tecnica, in tutti i giochi, consiste nella generazione delle immagini: tutti i primi esemplari di giochi, e anche la maggior parte di quelli attuali, usano una grafica "a scansione", ma ve ne sono alcuni (in particolare *Asteroids*), che usano il metodo chiamato "vector scanning" (ossia: "a scansione vettoriale"), nel quale la scansione del raggio elettronico "tocca" soltanto gli oggetti rappresentati sullo schermo (e non gli sfondi).
I giochi elettronici presenti nelle sale giochi e, sempre in maggior numero, anche nei bar, sono controllati da computer probabilmente molto più potenti e complessi di quelli impiegati nella maggioranza degli apparecchi commerciali o degli home computer: segno indicativo di una nuova società?

I "Floppy tape"

Una soluzione per chi non si accontenta dei nastri "a cassetta", ma non può acquistarsi le unità a dischi

Per l'utente di home computer, i nastri "a cassetta" offrono un modo economico e quasi sempre sicuro per conservare i programmi o utilizzare il software commerciale. Tuttavia, le "cassette" presentano alcuni inconvenienti. Il primo riguarda, senza dubbio, la velocità: anche in un sistema abbastanza veloce, per esempio a 1200 bit/sec., occorrono alcuni minuti per trasferire un programma mediamente lungo o per cercare un particolare dato. Un altro grosso problema è dovuto al fatto che il nastro si muove in una direzione sola (normalmente il computer non è in grado di operare i controlli di avanzamento e di riavvolgimento). Quindi, se il programma desiderato si trova in fondo al nastro, bisogna farlo scorrere tutto prima di poter utilizzare il programma.
Con i dischetti questi problemi non esistono, ma, ovviamente, le unità a dischi sono molto più care dei normali registratori a cassette.

L'ideale, per gli utenti di home computer, sarebbe un sistema che fosse molto più veloce delle cassette ma meno costoso del disco. In effetti, strumenti con queste caratteristiche esistono e sono chiamati "floppy tape" o "stringy floppie". Inizialmente, furono progettati dalla ditta americana Exactron per il computer TRS-80 Modello 1 della Tandy. La prima versione dell'apparecchio adoperava un nastro magnetico "ad anello", inserito in una cartuccia; l'idea fu copiata da un sistema di registrazione ad 8 tracce in voga alcuni anni fa. Il principio dell'operazione è molto

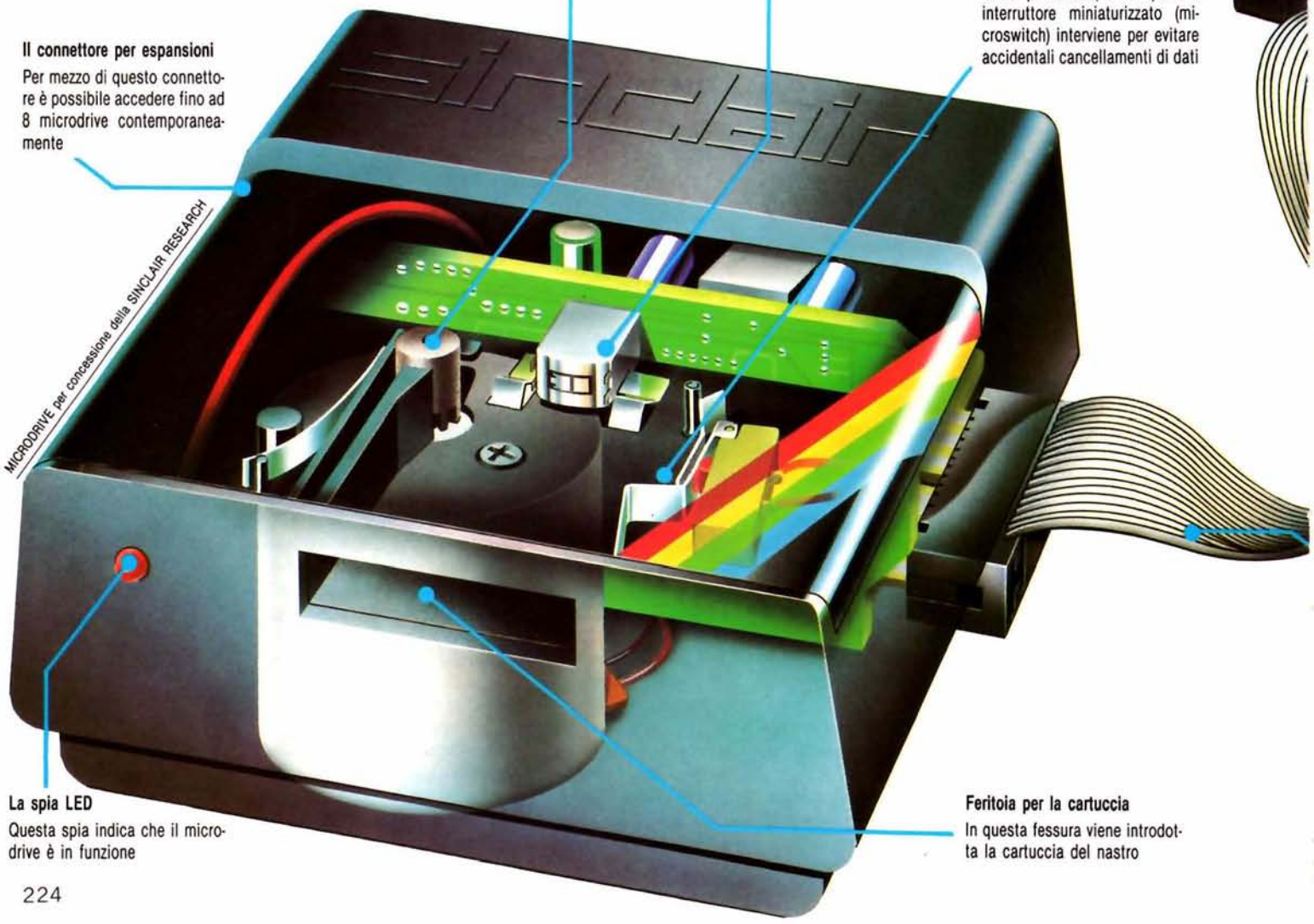
L'interfaccia 1
Questa interfaccia non solo fornisce il necessario collegamento tra il microdrive ed il computer, ma funziona anche come porta seriale per il collegamento ad una stampante oppure ad un'interfaccia per "network" (reti a più utenti), alla quale è possibile allacciare fino a 64 Spectrum

Il rullo trascinatore del nastro
Il nastro viene trascinato da questo rullo, che funziona in modo analogo a quello di un registratore a cassette

La testina di lettura/scrittura
Questa è una testina miniaturizzata e serve sia per la lettura che per la scrittura di dati ed è simile a quella che si trova sui normali registratori a cassette

Interruttore di protezione
Quando la cartuccia estraibile è stata protetta contro la scrittura, rimuovendo un'apposita linguetta di protezione, uno speciale interruttore miniaturizzato (microswitch) interviene per evitare accidentali cancellamenti di dati

Il connettore per espansioni
Per mezzo di questo connettore è possibile accedere fino ad 8 microdrive contemporaneamente



La spia LED
Questa spia indica che il microdrive è in funzione

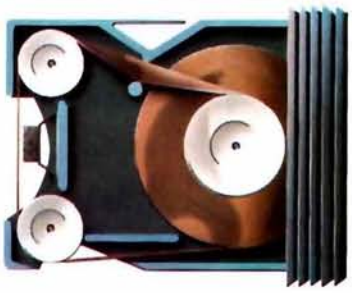
Feritoia per la cartuccia
In questa fessura viene introdotta la cartuccia del nastro

semplice: il nastro è costantemente in circolo, offrendo all'utente la possibilità di localizzare, in tempi molto brevi, un particolare programma o dato. Su di una speciale porzione del nastro viene tenuto l'indice di tutti i programmi e dei "file" memorizzati, analogamente a quanto avviene con i sistemi a dischi.



Cavo dell'interfaccia
Questo cavo collega il primo microdrive (di un'eventuale serie) all'unità per interfaccia

Le informazioni vengono registrate digitalmente e non si usano metodi basati sulle frequenze audio: pertanto i trasferimenti avvengono molto più velocemente che impiegando dei mangianastri a cassette (spesso con una velocità almeno cinque volte superiore). I floppy tape non implicano l'uso di complicate e costose interfacce: adoperano una normale porta parallela e tutto il software necessario è già incorporato nell'unità a nastri, oppure viene fornita su di una ROM da inserire all'interno del computer. Il nome stesso di queste unità descrive bene quello che esse in realtà sono: una via di mezzo fra il nastro (tape) ed il "floppy disk"; infatti, sono basate sull'impiego di nastro magnetico, ma il funzionamento è molto simile a quello dei dischi.
Sfortunatamente i primi esemplari presentavano diversi problemi, dovuti soprattutto al nastro, che, talvolta, non resisteva al continuo sforzo cui era sottoposto. Sono stati venduti solo pochi esemplari del floppy tape originale. Fino all'introduzione del Microdrive della Sinclair, questo sistema era ancora guardato con molto sospetto. Tuttavia, nel Microdrive, il principio rimane lo stesso: un rullo di nastro magnetico che scorre continuamente a contatto con una testina di scrittura/lettura. Inoltre, il nastro è molto più piccolo (circa 2 mm. di larghezza, ossia meno della metà di quello originale). Il Microdrive, comunque, sembra essere abbastanza affidabile.
Un sistema alternativo di conservazione dati, che offre un alto grado di velocità e di affidabilità, è il nastro digitale a cassette. Esistono già da tempo apparecchi professionali dal costo piuttosto elevato, ma l'introduzione del sistema "micro digital cassette", prodotto dalla Philips, comincia ad essere impiegato in apparecchi quali l'Hobbit (vedere pag. 94). In questo caso, il nastro non è "ad anello" e funziona ad una velocità eccezionale. L'indice dei programmi è situato a metà del nastro che può essere riavvolto in entrambe le direzioni di scorrimento, sotto il controllo del sistema operativo. Alcuni computer portatili, come il PC-1251 della Sharp e l'HX-20 della Epson, adoperano questo tipo di speciali microcassette.
L'unico vero problema, con tutt'e tre i sistemi di conservazione dati descritti, è dovuto al fatto che il software disponibile è molto limitato. Il PC-1251 e l'HX-20, comunque, sono capaci di trasferire programmi da un normale nastro a cassetta alla microcassetta interna.
Sia l'Hobbit che il Microdrive si possono collegare al computer contemporaneamente ad un normale registratore a cassette, permettendo perciò scambi di programmi fra i vari sistemi.
Per quanto riguarda l'affidabilità, il sistema digitale a microcassette risulta più efficace rispetto ai sistemi floppy tape o a normali cassette, ma è difficile prevedere se arriverà mai a sostituirli completamente.



LA CARTUCCIA DEL NASTRO

Il nastro contenuto dentro la speciale cartuccia, è formato da un rotolo continuo di nastro magnetico largo circa 2 mm. In genere viene impiegato nastro per videoregistratori, più robusto e durevole del normale nastro audio.
Durante l'operazione il nastro compie un giro completo in circa 7 secondi e i dati vengono trasferiti a circa 6 Kbyte/sec.: ciò costituisce un notevole passo in avanti rispetto ai 1,5 Kbyte/sec. dell'interfaccia per cassette dello Spectrum. È possibile localizzare e "caricare" un qualsiasi programma in circa 15 secondi. Su ciascun nastro si possono conservare fino a 100 Kbyte di dati, ma la Sinclair garantisce che solo 85 Kbyte sono disponibili all'utente. Prima dell'uso, occorre "formattare" il nastro: tale processo può essere avviato con un semplice comando BASIC. La formattazione serve per ottimizzare la distribuzione dei dati e per controllare quali parti del nastro siano utilizzabili o quali siano da usare per l'indice

Le qualità che contano

L'acquisto del primo computer, o dei moduli di espansione, deve essere fatto con molta attenzione

Le periferiche

Quando viene posto in vendita un nuovo computer, il relativo depliant spesso descrive la gamma di periferiche disponibili, ma a volte queste sono ancora in fase di progetto. È molto importante distinguere fra ciò che esiste davvero e quello che i produttori pensano di fare in un secondo tempo. Un computer abbinato ad un'unità a disco e stampante certamente sarà corredato da un buon software commerciale prima o poi. Un buon videogioco dovrà essere abbinato a joystick ed, eventualmente, ad una penna ottica

>>> Spectrum, BBC Model B, Tandy Color, TI99/4A
>> Atari 400 & 800, Vic-20, Commodore 64, Dragon-32
> Oric-1, Sord M5

Il monitor

Spesso viene data moltissima importanza alla grafica, ma per la maggior parte del tempo diversi utenti dovranno guardare un testo scritto oppure i listati dei programmi. Ci sono due fattori importanti in questi casi: il numero di caratteri visualizzabili sullo schermo (è da considerarsi buono un valore di 24 righe di 80 caratteri) e la leggibilità di tali caratteri. Un altro fattore molto importante è la possibilità di indirizzamento del cursore, essenziale, per esempio, per modificare il listato di un programma semplicemente spostando il cursore sui caratteri del testo e digitando le modifiche

>>> BBC Model B, Atari 400 & 800, Commodore 64
>> Oric-1, Sord M5, TI99/4A
> Spectrum, Vic-20, Dragon-32, Tandy Color



La tastiera

Se l'utente non ha intenzione di usare esclusivamente i joystick per i giochi, egli adopererà la tastiera e pertanto è importante trovarsi a proprio agio con essa. Due sono i fattori importanti: la sua affidabilità e la disposizione dei tasti. Una dattilografa, per esempio, si trova meglio con una tastiera di buona qualità, perché in quelle più economiche i tasti non offrono la certezza dell'avvenuto azionamento. Per quanto concerne il secondo fattore, sono preferibili le tastiere il più possibile simili a quelle delle macchine da scrivere, ed, inoltre, quelle nelle quali ciascun tasto ha un'unica funzione (altrimenti si rischia di far confusione). I tasti "funzione" sono utili perché si possono adoperare, all'interno dei programmi, per facilitare le operazioni più ricorrenti

>>> BBC Model B, Atari 800, Vic-20, Commodore 64
>> Dragon-32, Tandy Color, TI99/4A
> Spectrum, Atari 400, Oric-1, Sord M5

Il software commerciale

Chi decide di acquistare un computer per motivi professionali deve orientare la propria scelta su apparecchi di livello superiore, tuttavia gli home computer più costosi possono servire anche a questo tipo di uso, purché corredati da unità a disco e stampante. In tal caso, si possono impiegare programmi sofisticati per l'elaborazione dei testi, per la contabilità, per la gestione di archivi e di "lavagne elettroniche". Come nel caso dei videogiochi, la disponibilità di buon software varia a seconda della marca

>>> Atari 800, Commodore 64
>> BBC Model B, Atari 400, Vic-20, TI99/4A
> Spectrum, Dragon-32, Oric-1, Sord M5, Tandy Color

Le interfacce

Maggiore è il numero di interfacce disponibili, più possibilità esistono di collegare apparecchiature esterne. In particolare, la presenza di interfacce standard (la RS232, seriale, e la Centronics, parallela) permettono anche l'impiego di periferiche di marche diverse, offrendo una più ampia scelta

>>> BBC Modello B, Vic-20, Commodore 64
>> Atari 400 & 800, Dragon 32, Oric-1, Sord M5, Tandy Color
> Spectrum, TI99/4A



La grafica

La difficoltà nel paragonare le capacità grafiche dei vari home computer risiede nel fatto che nella maggioranza di essi è disponibile più di un sistema per la creazione di immagini. Per esempio, digitando "MODE 1" su di un computer, si potrebbe aver accesso a 16 colori su di un reticolo di 40 x 25 posizioni, mentre "MODE 7" potrebbe dare la massima risoluzione di 320 x 200 pixel, ma con soli due colori. Un fattore importante da tener presente è che su alcuni computer la massima risoluzione può essere ottenuta solamente su speciali monitor e non sul normale schermo televisivo

>>> BBC Model B, Atari 400 & 800, Commodore 64, Sord M5, TI99/4A
>> Spectrum, Dragon-32, Oric-1, Tandy Color
> Vic-20

La qualità del BASIC

Quasi tutti gli home computer sono dotati di un interprete BASIC residente nelle ROM, ma, come già accennato nella rubrica "Basic & Basic", i comandi variano molto da macchina a macchina. Tutte le versioni di BASIC della Microsoft operano più o meno allo stesso modo, mentre altre versioni differiscono, generalmente, nel trattamento delle stringhe e delle matrici e nella gestione della grafica. Un buon BASIC dovrebbe disporre di una nutrita serie di comandi utili alla grafica, quali, ad esempio: CIRCLE, DRAW, PAINT, ecc.

>>> BBC Model B, Dragon-32, Spectrum, Atari 400 & 800, Oric-1, Tandy Color, TI99/4A
>> Vic-20, Commodore 64
> Sord M5

Gli effetti audio

I computer che utilizzano il circuito audio del televisore (o addirittura un sistema HI-FI esterno) sono capaci di una maggiore qualità di riproduzione, rispetto a quelli con generatori audio incorporati: è sempre consigliabile chiedere una dimostrazione pratica, prima di acquistare un computer. I sistemi "monofonici" producono soltanto una nota alla volta (come un pianoforte suonato con un dito solo), mentre quelli a 3 o 4 voci (polifonici) possono anche riprodurre gli interi accordi. Alcuni computer sono dotati di un generatore "a rumore bianco", per la creazione di speciali effetti sonori (esplosioni, suoni "spaziali", ecc.), mentre altri possiedono veri e propri sintetizzatori musicali in scala ridotta

>>> BBC Model B, Atari 400 & 800, Commodore 64, Oric-1, Sord M5, TI99/4A
>> Vic-20
> Spectrum, Dragon-32

Il software per videogiochi

L'ampia gamma di software per giochi è fornito, non dalle ditte produttrici di computer, ma da fornitori specializzati in software. I fattori che governano la qualità e la quantità di giochi prodotti per un dato computer sono la diffusione e le caratteristiche tecniche dell'apparecchio. Di conseguenza alcuni computer sono progettati per essere usati come videogioco, mentre altri per esercitarsi nella programmazione o per uso commerciale

>>> Spectrum, Atari 400 & 800, Vic-20, TI99/4A
>> BBC Model B, Commodore 64, Dragon-32, Sord M5, Tandy Color
> Oric-1

Il registratore a cassette

Se l'utente non intende servirsi esclusivamente di cartucce ROM oppure di software basato sull'impiego dell'unità a disco, deve ricorrere ad un registratore a cassette. Solo un paio di computer sono costretti ad usare un registratore a cassette speciale: di solito, gli home computer lavorano con normali registratori a cassette di tipo "domestico". Bisogna tener presente, però, che la velocità con la quale vengono trasferiti i programmi e l'affidabilità del sistema variano considerevolmente da marca a marca. I sistemi migliori prevedono che sia il computer a controllare direttamente il funzionamento del motore del registratore

>>> Spectrum, BBC Model B
>> Dragon-32, Oric-1, Sord M5, Tandy Color, TI99/4A
> Atari 400 & 800, Vic-20, Commodore 64

La memoria

Il grado di sofisticazione dei programmi adatti ad un dato computer è in diretto rapporto con la quantità di memoria RAM: maggiore è la memoria, più dati possono essere elaborati contemporaneamente. La memoria standard, nella gamma degli home computer, varia da 1K a 64K, ma nella maggioranza dei casi è possibile ampliare la memoria con l'uso di moduli o schede aggiuntive. Spesso, il computer utilizza parte di questa memoria per il proprio uso interno, riducendone la quantità a disposizione per i programmi. In particolare, se il video è a colori e si usa la grafica ad alta risoluzione, questa porzione riservata alle operazioni interne aumenta considerevolmente

>>> Spectrum (48K), BBC Model B, Atari 800, Commodore 64, Dragon-32, Oric-1 (48K)
>> Atari 400, Tandy Color (16K), TI99/4A
> Vic-20, Sord M5 (4K)

Esiste una scelta talmente vasta di microcomputer da sconcertare chi stia procedendo all'acquisto del suo primo apparecchio o chi intenda procurarsi degli accessori per quello che già possiede.
Fino ad ora, abbiamo descritto in dettaglio alcuni degli home computer che si trovano in commercio, spiegandone il principio di funzionamento. Inoltre, abbiamo presentato le più comuni applicazioni di microcomputer nella vita odierna ed abbiamo introdotto

Caveat emptor

Ogni mese, nei negozi specializzati, appaiono nuove marche o nuovi modelli di computer e l'imbarazzo della scelta rende ancor più difficile l'acquisto. Ci sono talmente tanti fattori da tenere in considerazione che occorre prestare la massima attenzione



alcuni termini tipici del gergo "computerese". Ci sono molti aspetti da prendere in considerazione quando si acquista un microcomputer per la prima volta: il primo forse è il costo. Oggi si può acquistare uno ZX81 della Sinclair ad un prezzo incredibilmente basso, ma questo computer ha un punto debole, la tastiera: nel tentativo di ridurre al minimo le dimensioni dell'apparecchio, la Sinclair ha impiegato un'economica tastiera "a sfioramento", non molto comoda, specie per il principiante. In effetti, programmare un computer è già un compito impegnativo per un principiante, sia dal punto di vista intellettuale che per l'attenzione che egli deve prestare ai piccoli dettagli e l'uso di una tastiera che non "confermi" in modo sensibile la pressione dei tasti può costituire un'ulteriore complicazione.

D'altronde, da un apparecchio così economico non si possono pretendere le prestazioni che si hanno con i computer più grandi (per esempio, lo ZX81 non è in grado di generare colori o suoni ed, inoltre, la gamma di accessori da abbinare non è tra le più ampie). Lo stesso genere di considerazioni vale per altri computer a basso costo, quali, ad esempio, il PC-1500 della Sharp.

Esiste tutta una serie di caratteristiche, che ci si aspetta di trovare su di un home computer: una panoramica completa è riportata nelle pagine precedenti. Prima di acquistare un computer è molto utile prepararsi un elenco, in ordine di importanza, di tutte le caratteristiche concernenti la particolare applicazione dell'apparecchio. Se il computer verrà adoperato principalmente per lo svago, i fattori determinanti saranno: una buona scelta di software di base, la disponibilità di periferiche appropriate (come joystick o trackball) ed anche una buona grafica oltre alla possibilità di generare effetti sonori.

Se, invece, il computer verrà impiegato per tenere una contabilità oppure per l'elaborazione di grandi quantità di dati, il numero di righe e colonne visualizzabili sullo schermo diviene un fattore estremamente importante, assieme alla qualità della tastiera, ad una buona capacità di memoria ed alla possibilità di allacciamento con una stampante. Naturalmente, occorre anche una vasta gamma di software già pronto.

Un computer, di per sé, è un apparecchio generalmente molto versatile, ma alcuni modelli si prestano, meglio di altri, ad assolvere alcuni particolari compiti. Ecco perché occorre partire con idee ben chiare. Nell'elenco, vale la pena annotare tutto quello che ci viene in mente, magari assegnando a ciascuna voce un grado di priorità (per esempio da 1 a 10).

Se l'applicazione del computer sarà molto generica è conveniente preferire computer dotati di un buon BASIC, di una tastiera comoda, di ampie possibilità d'espansione e di una memoria piuttosto ampia.

Anche se talvolta non si ha grande scelta, può essere utile non fermarsi nel primo negozio che capita, ma visitarne più d'uno. Probabilmente, non si otterranno grandi sconti, ma a volte i rivenditori offrono qualche gioco (o programma) in omaggio. Inoltre, esiste già un mercato dell'usato: molte riviste del settore contengono diverse pagine dedicate all'usato. Differentemente da quanto accade nel settore automobilistico, in genere i rivenditori non sono inclini a vendere nuovi apparecchi offrendo di ricomprare computer usati,

quindi il mercato degli apparecchi usati è ancora molto aleatorio.
Quando l'acquisto riguarda eventuali espansioni o periferiche, da aggiungere ad un sistema già esistente, l'attenzione da prestare è anche maggiore. Se, per esempio, si acquista uno Spectrum, saremo, successivamente, legati all'acquisto di periferiche prodotte dalla Sinclair, che potrebbero risultare non sufficienti per l'applicazione voluta. Lo stesso discorso vale per le espansioni di memoria: la scelta, per chi possiede un Vic-20 della Commodore, per esempio, spazierà tra le unità prodotte dalla stessa Commodore e quelle prodotte da altre ditte, dichiarate "compatibili" (ma attenzione: talvolta la compatibilità non è del tutto veritiera!).
Nel settore delle stampanti, forse, esistono le più ampie facoltà di scelta: non solo ci sono moltissime marche, ma anche cinque o sei tipi diversi di stampa: i più comuni sono quello a matrice di punti, quello "a margherita" e quello, infine, "a getto d'inchiostro" (vedere a pag. 74). Se si intende adoperare un programma di "word processing" una stampante "a margherita" è la più indicata, poiché la qualità di stampa è simile a quella di una macchina da scrivere elettrica, mentre, se interessa la stesura di budget o la stampa di tabulati, il numero di colonne di stampa è di primaria importanza.
Esaminiamo, infine, la scelta del sistema per la conservazione dei dati, ossia la memoria "di massa". Il dilemma è se acquistare un'unità a dischi (certamente più costosa), oppure accontentarsi del normale registratore a cassette. Esiste anche una terza possibilità: i floppy tape (vedere a pag. 224). In questo caso, affrontare la spesa per un'unità a dischi può rivelarsi, a lungo termine, un risparmio notevole nei tempi di elaborazione e nella possibilità di usare programmi migliori.
L'ultimo elemento, che lascia sempre assai sconcertati, è il continuo calo dei prezzi: il potenziale acquirente si chiede se non convenga attendere qualche mese con la speranza di vedere calare il prezzo del computer che lo interessa. Tutto fa pensare che i prezzi continueranno ad abbassarsi anche nel prossimo futuro (salvo impennate del dollaro!) e seguendo la stampa specializzata è talora possibile anticipare l'entità di eventuali fluttuazioni: in ogni caso, però, la speranza di risparmiare poche migliaia di lire non è sempre un buon motivo per rimandare l'acquisto. È molto più importante, invece, avere la certezza di acquistare la macchina giusta. Come abbiamo già osservato, il modo più sicuro è di meditare bene sull'uso che si vuol fare del computer. Bisogna ricordarsi sempre che un computer è molto di più di una scatola con una tastiera da collegare alla televisione ed è molto più importante del software o delle periferiche. Il computer è un apparecchio che, se sfruttato al massimo delle sue capacità, è in grado di offrire molte ore di divertimento, di risolvere un'infinità di problemi e di gestire enormi quantità di utili informazioni.



IL SOFTWARE INNANZITUTTO

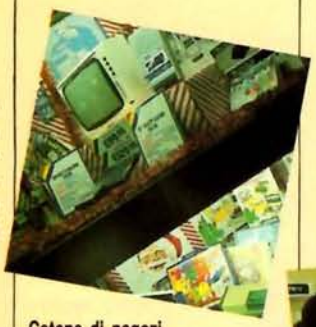
La disponibilità di un'ampia scelta di software è importante tanto quanto le caratteristiche tecniche del computer. Più diffuso è il computer, più possibilità ci sono di poter acquistare del software progettato specificamente per esso: allo stesso tempo più software esiste per un certo modello, più probabilità ci sono per la vendita

VETRINA PER GLI ACQUISTI

Rivendite di elettrodomestici

In molti di questi negozi sono in vendita piccoli computer, spesso anche a prezzi convenienti, ma, in genere, l'acquirente parte già sapendo la marca ed il modello che desidera



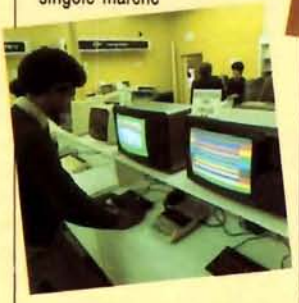


Catene di negozi

In Italia, come è già accaduto all'estero, alcune grosse catene di rivendita hanno riservato un settore ai computer (è il caso dei "Bit shop" sparsi un poco ovunque in Italia). In queste rivendite, si trova spesso personale sufficientemente preparato e l'acquisto risulta facilitato

Negozi specializzati

In quasi tutte le città sono nati dei "computer shop", negozi dedicati esclusivamente alla vendita di questi apparecchi. Il personale di vendita aiuta gli interessati nella scelta e nella prova dei computer in mostra. La scelta in genere è molto ampia e non è legata a poche singole marche





Vendita per corrispondenza

Nei cataloghi delle più conosciute organizzazioni di vendita per corrispondenza sono già apparsi i computer. Talvolta, anche gli stessi fabbricanti dei computer più economici ricorrono a questo sistema di vendita

L'Aquarius

Anatomia di un microcomputer a diffusione europea: tecnologie a confronto

L'Aquarius, dotato di un microprocessore Z80 e di una tastiera un po' particolare "a pulsanti", appartiene alla stessa categoria dello Spectrum della Sinclair. Comunque, sotto molti aspetti, l'Aquarius è una macchina molto più flessibile, soprattutto perché il "bus" di espansione interno è stato ben progettato.
Ad esso possono essere collegati svariati moduli di espansione, che vanno da piccole RAM di 4 Kbyte a moduli di espansione di grandi dimensioni. Forse il più usato fra questi è un telaietto di espansione, che permette di collegare 2 schede di memoria supplementare (o cartucce di programmi), 2 accessori di comando tipo joystick, oltre ad essere fornito di 2 canali audio supplementari. Collegando un'espansione RAM da 16 Kbyte e una cartuccia ROM, che contenga, ad esempio, il programma "Finplan", si ottiene un sistema abbastanza versatile.
La RAM di 4 Kbyte incorporata nella macchina è appena sufficiente, ma, con la possibilità di espanderla fino a 64 Kbyte, le prestazioni dell'Aquarius superano quelle dei normali home computer.
Tuttavia, la tastiera ed il video non offrono la stessa qualità di macchine più grandi: la barra spaziatrice è sostituita da un tasto, situato in una posizione insolita, ed i tasti non sono né molto sensibili né veloci nella risposta. La tastiera, quindi, non si presta bene alla battitura di testi. Il video ha 24 linee di 40 caratteri, e, anche se di dimensioni ragguardevoli, non è adeguato per piccole applicazioni commerciali.
I colori disponibili, sia per lo sfondo che per i caratteri di testo, sono 16 e, sebbene manchi all'utente la possibilità di definire caratteri speciali, i simboli visualizzabili sono 256 (caratteri maiuscoli, minuscoli ed una discreta gamma di simboli grafici). In alta risoluzione si possono rappresentare 320 × 192 pixel. L'uscita video si può collegare soltanto ad un televisore (non è previsto l'impiego di monitor "dedicati") e la qualità delle immagini è mediocre: i caratteri appaiono leggermente offuscati, ma, in compenso, l'immagine è stabile e luminosa, con una buona gamma di colori.
L'Aquarius è fornito di un generatore di effetti audio, sebbene le forme d'inviluppo non siano controllabili quanto in altri computer. Il BASIC fornito è quello "standard" della "Microsoft", ma sarà presto reperibile una versione "estesa" del BASIC ed anche un LOGO.
Uno dei più interessanti accessori per questo computer è il sistema BSR X-10 che permette di controllare fino a 255 elettrodomestici mediante i segnali generati dall'unità centrale. Non è necessario alcun impianto supplementare, poiché i segnali vengono inviati sotto forma di impulsi, direttamente sull'impianto elettrico centrale della casa. Gli impulsi non sono sufficientemente grandi da provocare disturbi nella linea di alimentazione, ma un demodulatore X-10 inserito in una qualsiasi presa di corrente della linea principale, può captare il codice e modificare, a seconda del comando ricevuto, la corrente erogata dall'apparecchio ad esso collegato. Purtroppo, questa interessantissima applicazione rischia di non essere utilizzabile in Italia, dove gli impianti elettrici, nella maggioranza delle abitazioni, non si adeguano ancora allo standard europeo.



Stampante Aquarius
Questa stampante, dal prezzo molto contenuto, impiega un meccanismo di stampa termico ed utilizza quindi carta di tipo speciale. Può stampare fino a 80 caratteri al secondo in righe di 40 caratteri. È inoltre disponibile una stampante/plotter a 4 colori



Unità di espansione
Questo apparecchio permette di utilizzare, contemporaneamente, una cartuccia di programmi ed una memoria supplementare. Consente, inoltre, l'uso di 2 comandi manuali (per i giochi) e di 2 canali audio supplementari

La tastiera
La tastiera è uno dei punti deboli dell'Aquarius. Sebbene la disposizione dei tasti si avvicini allo standard QWERTY, non è dotata di barra spaziatrice, ha un solo tasto SHIFT, il RETURN è in posizione inconsueta e le spaziature tra i tasti non corrispondono a quelle di una normale tastiera

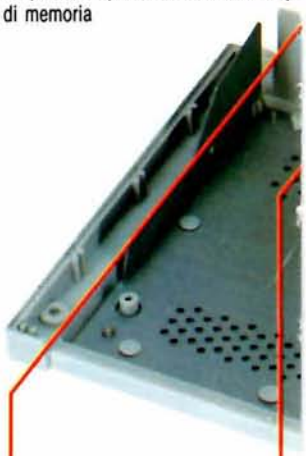


Connettore RF
Da qui esce il segnale video per il televisore. Non è previsto l'impiego di monitor "dedicati"

Presa d'alimentazione
A questa presa si collega l'alimentatore esterno

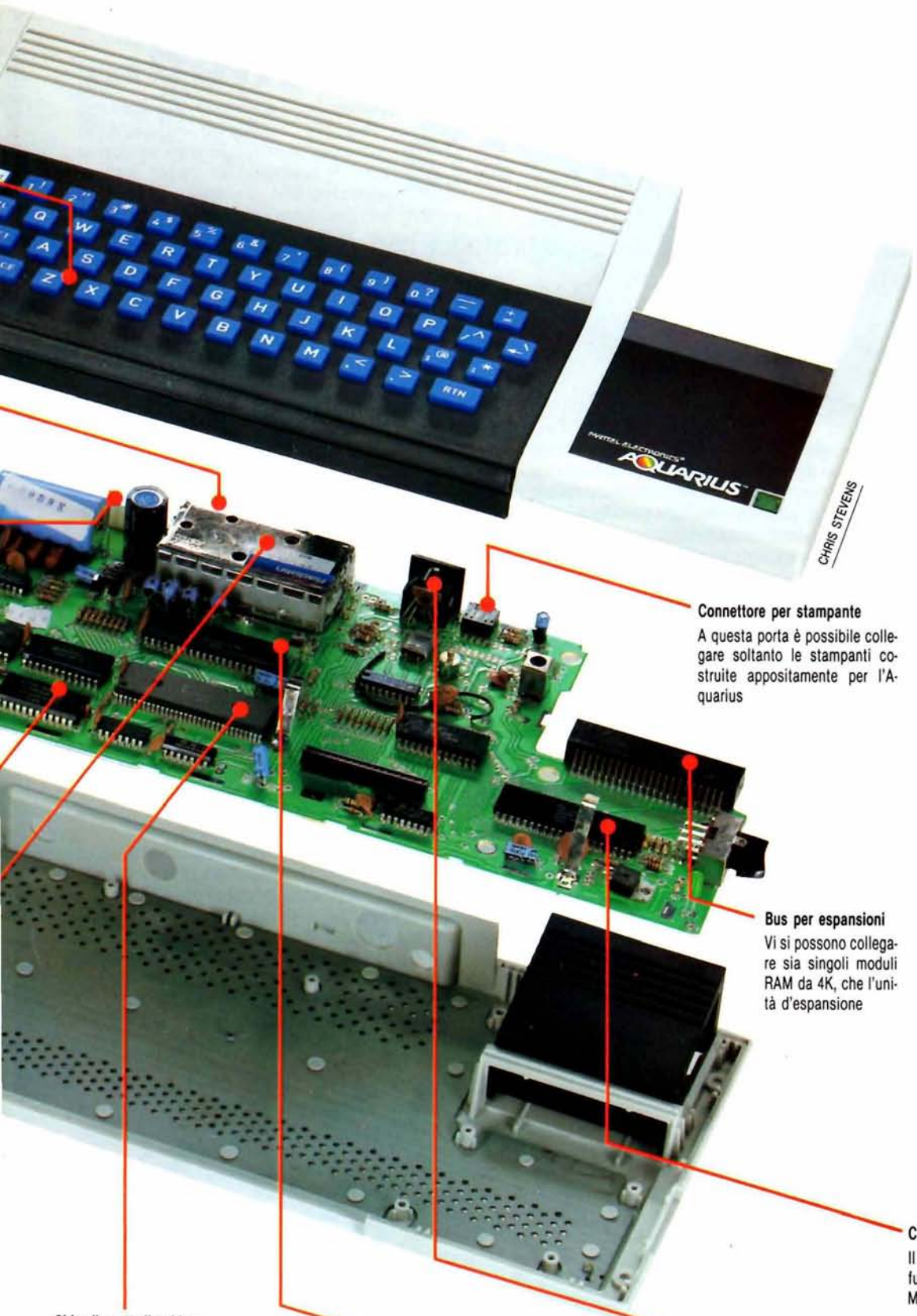


RAM
In questo chip sono contenuti 4 Kbyte di memoria



ROM
In questo chip si trova il BASIC Microsoft, versione standard, da 8 Kbyte. Le restanti ROM sono impiegate per la gestione della grafica e degli effetti audio

Modulatore
Converte i segnali interni in impulsi per la visualizzazione sullo schermo di normale televisore



AQUARIUS

DIMENSIONI

mm. 345 x 150 x 55

PESO

—

CPU

Z80

CLOCK

3,5 MHz

MEMORIA

10 Kbyte di ROM e 4 Kbyte di RAM (espandibili a 64 Kbyte)

VIDEO

24 linee di 40 caratteri, o 320 x 192 punti per la grafica ad alta risoluzione. 16 colori disponibili. 256 caratteri pre-definiti

INTERFACCE

Connettore per TV, per registratore a cassette e per stampante. Bus di espansione

LINGUAGGI FORNITI

BASIC Microsoft

ALTRI LINGUAGGI DISPONIBILI

(Annunciati) LOGO e BASIC "esteso"

ACCESSORI COMPRESI

Cavo di connessione per TV. Manuale d'installazione e manuale del BASIC

TASTIERA

49 tasti "a pulsante". Il tasto di RESET è situato in modo da impedirne una pressione accidentale

DOCUMENTAZIONE

La documentazione è particolarmente adatta ai principianti, corredata da una serie di schedine, sulle quali sono descritte le principali funzioni. Manca, per i più esperti, una parte dedicata all'approfondimento tecnico

Connettore per stampante
A questa porta è possibile collegare soltanto le stampanti costruite appositamente per l'Aquarius

Bus per espansioni
Vi si possono collegare sia singoli moduli RAM da 4K, che l'unità d'espansione

CPU
Il processore è uno Z80, che funziona ad una frequenza di 3,5 MHz

Chip di controllo video
I circuiti che controllano l'immagine video stanno assumendo un'importanza sempre crescente nell'architettura dei computer. Questo chip di regolazione automatica è più grande dello stesso microprocessore

Chip di "protezione"
Questo chip, progettato appositamente per l'Aquarius, ha lo scopo di impedire, per chiunque non sia il fabbricante, la produzione di cartucce di programmi per questo computer

Connettore per registratore
Lo spinotto per il registratore è del tipo DIN. Le linee presenti permettono al computer di controllare il movimento del motore del registratore

Una rubrica telefonica... computerizzata

Il nostro corso di programmazione prosegue con un'applicazione pratica di quanto finora studiato

È tempo di mettere a buon frutto tutti gli insegnamenti appresi finora, applicandoli allo sviluppo di un vero programma. Ovviamente, anche i brevi programmi presentati nelle altre lezioni erano "veri", ma costituivano soltanto dei piccoli moduli per eseguire solo qualche specifica funzione: piccoli meccanismi di un orologio. Adesso costruiamo l'intero orologio!

Una delle più frequenti domande che vengono rivolte agli "esperti" è: «A cosa mi serve un computer?»

La risposta non è così semplice, quanto ci si potrebbe immaginare, anzi è forse una delle più difficili.

Le applicazioni sono tante e tali che elencarle tutte è impossibile, ma, se si resta sul generico, si rischia di sentirsi rispondere che i metodi manuali sono più pratici.

Qual è dunque quella sottile distinzione tra una procedura manuale ed una computerizzata, che dimostra quanto sia indispensabile un elaboratore elettronico?

Vediamo di individuarla presentando lo sviluppo di un programma, che, a tutta prima, può sembrare abbastanza frivolo: una rubrica telefonica computerizzata.

Le normali rubriche telefoniche "manuali" sono impostate in maniera da facilitare la ricerca alfabetica di un particolare nominativo: le pagine vengono suddivise in gruppi ed identificate con una lettera dell'alfabeto. Ma i nomi, in ciascuna pagina, non vengono aggiunti seguendo un criterio strettamente alfabetico: alla lettera P, per esempio, il primo potrebbe essere Antonio Pellegrini, il secondo Michele Portinai, ma un terzo, aggiunto in un momento successivo, potrebbe essere Stefano Palumbo. Oltre ai nominativi, si registrano altri dati (indirizzo, numero di telefono, ecc.).

Una rubrica manuale, pur funzionando abbastanza bene in una ricerca alfabetica, non può fornirci un elenco di persone che abitino tutte in una stessa città, e nemmeno aiutarci a risalire ad un nominativo partendo dal solo numero telefonico.

In applicazioni domestiche, queste funzioni "accessorie" possono essere irrilevanti, ma nel campo commerciale può essere di grande aiuto poter selezionare, secondo più parametri fissati a piacimento, particolari gruppi di clienti (o di fornitori).

Supponiamo di commerciare in abbigliamento: volendo informare la nostra clientela della disponibilità di alcuni articoli per l'infanzia, sarebbe molto oneroso inviare una comunicazione a tutti i clienti, includendo anche quelli che non trattano in abbigliamento infantile.

Questo è il tipo di ragionamento che occorre fare per stabilire se e come può essere utile impiegare un computer.

Stabilito ciò, la prossima considerazione concerne l'acquisto del programma: sceglierne uno già pronto in commercio o farne uno "ad hoc"?

Se il programma esiste già e corrisponde alle nostre necessità (anche di prezzo!) la faccenda è risolta, altrimenti possiamo pensare di scrivercelo da soli, sicuri che corrisponderà alle nostre specifiche esigenze.

Lo sviluppo di un programma, come abbiamo già fatto notare altrove, passa attraverso più di uno stadio di lavoro e la prima mossa consiste nel definire chiaramente il problema, ossia nella "Definizione del problema".

Il secondo passo consiste nel trovare la soluzione del problema, esaminando attentamente quale tipo di "input" forniremo al computer ed il tipo di "output" desiderato. In queste due fasi le descrizioni devono essere abbastanza sintetiche, non troppo particolareggiate. Successivamente, esse verranno definite in sempre maggior dettaglio, fino a quando è possibile passare comodamente all'ultima fase, quella di codifica vera e propria in un particolare linguaggio di programmazione.

Noi useremo il BASIC, ma il sistema vale per qualsiasi altro linguaggio. Durante le fasi intermedie, si può impiegare (e noi lo faremo) una specie di "linguaggio fittizio" di programmazione, più elastico dei normali linguaggi da computer.

Questo metodo di programmazione si chiama "top-down", ossia "dall'alto in basso", per indicare che, partendo da una descrizione molto generica, si giunge alla codifica particolareggiata. Cercheremo di attenerci, quanto più possibile, anche ai principi della cosiddetta "programmazione strutturata", che chiariremo nel corso dello sviluppo.

Le varie fasi si possono così riassumere:

1. Chiara definizione del problema
2. Il formato dell'"input" e dell'"output" (descrizione al primo livello)
2.1 Raffinamento (descrizione al secondo livello)
2.2 Ulteriore raffinamento (descrizione al terzo livello)
3. Codifica nel linguaggio di programmazione

Prima d'imbarcarci in un qualsiasi progetto di software un po' complesso, occorre sempre individuare ed isolare il problema da risolvere. Vediamo cosa accade per la nostra rubrica computerizzata.
Per prima cosa, elenchiamo tutte le funzioni che desideriamo: decideremo in seguito se e quali sarà possibile includere effettivamente nel programma senza avere troppe complicazioni.
Noi vorremmo:

1. Visualizzare un indirizzo, numero telefonico ed un'annotazione, fornendo il nominativo da tastiera
2. Ottenere un elenco di nominativi, indirizzi e numeri telefonici, fornendo soltanto una parte del nominativo (ad esempio solo il nome di battesimo)
3. Ottenere un elenco di nominativi, indirizzi e numeri telefonici relativi ad una particolare città o zona
4. Ottenere un elenco di nominativi in base alla lettera iniziale
5. Ottenere un elenco, ordinato alfabeticamente, di tutti i nominativi presenti nella rubrica
6. Aggiungere nominativi a piacimento
7. Modificare, a piacimento, i dati relativi ad un nominativo
8. Cancellare nominativi a piacimento

Supponiamo per un momento che il programma sia già stato scritto: che forma dovrebbero avere l'input e l'output dei dati? Come vorremmo che funzionasse, dal punto di vista dell'utente?
I programmi applicativi, in generale, possono suddividersi in tre categorie, a seconda del metodo impiegato per "interagire" con l'utente: "a menu", "a comandi" o una combinazione di questi.
In un programma "a menu", quando è necessario scegliere tra diverse operazioni, viene presentato all'utente un elenco di opzioni, selezionabili digitando un numero o una lettera. Nei programmi del tipo "a comandi", l'utente deve digitare, per intero, uno o più comandi. Il vantaggio dei programmi che impiegano i "menu" è senza dubbio quello di essere facili da usare anche per i principianti, oltre a ridurre al minimo le possibilità di introdurre comandi errati.

I programmi funzionanti "a comandi", invece, offrono una maggiore velocità d'impiego per gli utenti esperti. Per il nostro programma useremo i menu, ma la scelta è puramente arbitraria.
In primo luogo dobbiamo definire il formato dei dati che verranno inseriti. Infatti, il computer dovrà accettare (ed interpretare correttamente) una qualsiasi delle seguenti forme:

C.E. Mareschi
Leonardo da Vinci
Giulia MARCHINI
Maria V.
P. D'Orazio
a.g. Lorenzetti
U Fontani
Corrado
NANNI GIUSTINELLI
Dr. Giacomo Ferretti

Può sembrare un po' eccessivo prevedere tutte queste forme, ma si pensi all'irritazione risultante dal ricevere un messaggio "NOME NON TROVATO", semplicemente perché si è scritto, per esempio, U Fontani invece di U. Fontani!
Possiamo agire in due sensi: o prevediamo l'immissione di una qualsiasi delle forme citate, affidando ad una serie di ingegnose subroutine il compito di interpretare correttamente il dato, oppure costruiamo un "filtro", che rifiuti i dati immessi, se questi non corrispondono ad un formato fissato a priori, visualizzando il messaggio "FORMATO NON CORRETTO".
Noi opteremo per la prima soluzione: più complessa, ma molto più elastica.
Per quanto riguarda la ricerca o il riordinamento alfabetico, i nominativi si considerano formati da due elementi: uno è il cognome e l'altro è costituito dagli altri nomi o iniziali. Potremmo definire il cognome come una serie di caratteri (maiuscoli o minuscoli) preceduta da un carattere "spazio" (codice ASCII 32) e terminata con un RETURN (codice ASCII 13). Primo problema: che cosa accadrebbe se immettessimo il nome "Corrado" senza farlo precedere da uno spazio? Esso verrebbe rifiutato, quindi la nostra definizione non è adatta e va cambiata.
In ogni caso, un nominativo inizia sempre con una lettera dell'alfabeto (maiuscola o minuscola), e termina sempre con un RETURN (non faremo accettare il punto come carattere terminale). Se il nominativo contiene uno spazio, daremo per scontato che l'ultimo gruppo di caratteri costituisce il cognome, mentre gli altri caratteri saranno, ai nostri fini, il nome.
Nel caso immettessimo soltanto un soprannome (ad esempio, Pippo), esso verrà considerato un cognome, anche nella fase di riordino alfabetico. Un altro problema, nell'ordinamento alfabetico è: tra i due nominativi G.B. Santini e Giovan Battista Santini, quale viene per primo? Una soluzione è quella di

ignorare la punteggiatura, riunire le iniziali e, quindi, considerare GB Santini precedente a Giovan Battista Santini.
Un altro aspetto da considerare, successivamente, sarà il formato da impiegare nell'output: dovremo decidere se il nominativo (assieme all'indirizzo ed al numero telefonico) verrà visualizzato, come risultato di una ricerca, tale e quale lo abbiamo immesso, oppure secondo un formato prefissato.
Intanto supponiamo di avere tre variabili alfanumeriche: NOMINATIVI$, NOMI$ e COGNOMI$. In NOMINATIVI$ verrà depositato il dato immesso da tastiera: come separeremo le sue componenti per assegnarle a NOMI$ e COGNOMI$? Un metodo potrebbe essere descritto così:

Convertire tutti i caratteri in maiuscole
Eliminare tutti i caratteri non alfabetici, salvo lo "spazio" separatore
Assegnare tutti i caratteri che seguono lo spazio a COGNOMI$
Assegnare tutti i caratteri che precedono lo spazio a NOMI$

Prima di precipitarsi a codificare questo metodo in istruzioni BASIC, vediamo come conviene procedere, seguendo i principi della programmazione strutturata.
Per mantenere i nostri appunti ad un livello accettabile di chiarezza (potremmo in futuro voler apportare delle modifiche al programma), impiegheremo nomi molto descrittivi sia per le variabili (ad esempio: COGNOMI$), che per particolari funzioni: INIZIO CICLO, FINE CICLO, ecc.
Quando saremo giunti alla fase di effettiva codifica, sostituiremo a queste indicazioni le istruzioni vere e proprie del BASIC.
Vediamo adesso il piano di sviluppo di questa parte del programma:

1ª Definizione del problema

INPUT
Un nominativo (con formato a piacere)

OUTPUT
1. Un nome
2. Un cognome

1° Raffinamento

1. Lettura di NOMINATIVI$
2. Conversione di tutte le lettere in maiuscole
3. Ricerca dello "spazio" separatore
4. Lettura di COGNOMI$
5. Lettura di NOMI$
6. Scarto di tutti i caratteri non alfabetici in COGNOMI$

2° Raffinamento

1. Lettura di NOMINATIVI$
2. Conversione delle lettere in maiuscole
   INIZIO
   CICLO finché ci sono caratteri in NOMINATIVI$
   Leggi un carattere di NOMINATIVI$
   SE il carattere è minuscolo
   ALLORA converti in maiuscolo
   ALTRIMENTI non eseguire la conversione
   FINE SE
   Assegna il carattere ad una variabile temporanea
   FINE CICLO
   NOMINATIVI$ = variabile temporanea
   FINE
3. Trova spazio separatore
   INIZIO
   CICLO finché ci sono caratteri in NOMINATIVI$
   SE il carattere è " " (spazio)
   ALLORA annotane la posizione in una variabile
   ALTRIMENTI passa al prossimo carattere
   FINE SE
   FINE CICLO
   FINE
4. Leggi COGNOMI$
   INIZIO
   Assegna i caratteri di NOMINATIVI$ a COGNOMI$ partendo dallo spazio separatore in poi
   FINE
5. Leggi NOMI$
   INIZIO
   CICLO finché ci sono caratteri in NOMINATIVI$ (prima dello spazio separatore)
   ESAMINA un carattere alla volta
   SE il carattere non è lettera alfabetica
   ALLORA passa oltre
   ALTRIMENTI assegna il carattere a NOMI$
   FINE SE
   FINE CICLO
   FINE
6. Scarta i caratteri non alfabetici da NOMI$
   Funzione già svolta al punto 5

Questo secondo livello di raffinamento è già molto vicino all'effettiva codifica. Proviamo a sviluppare il punto 2 ad un livello ulteriore ed a tradurlo poi in istruzioni BASIC. Possiamo impiegare un algoritmo già usato in precedenza (vedere a pag. 212):

3° Raffinamento

2. Conversione in lettere maiuscole
   INIZIO
   LEGGI NOMINATIVI$
   CICLO
   FOR L = 1 TO lunghezza della stringa
   LEGGI il carattere di posizione L
   IF (se) il carattere è minuscolo
   THEN (allora) sottrai 32 dal suo valore ASCII
   ELSE (altrimenti) passa oltre
   FINE IF
   LET VARTEMP$ = VARTEMP$ + carattere
   FINE CICLO
   LET NOMINATIVI$ = VARTEMP$
   FINE

Questo frammento di programma, scritto in pseudo-istruzioni, è sufficientemente vicino all'effettiva codifica in BASIC. Poiché il BASIC da noi impiegato non consente di usare

che un massimo di due lettere per i nomi delle variabili, impiegheremo, al posto di NOMINATIVI$ il nome N$:

```
1000 REM SUBROUTINE DI CONVERSIONE IN MAIUSCOLE
1010 INPUT "DIGITARE UN NOMINATIVO";N$ : REM A SOLO
     SCOPO DI VERIFICA
1020 LET P$ = "" : REM LA VARIABILE TEMPORANEA VIENE
     "VUOTATA"
1030 FOR L = 1 TO LEN(N$) : REM INIZIO DEL CICLO
1040 LET T$ = MID$(N$,L,1) : REM ESTRAZIONE DI UN CARAT-
     TERE
1050 LET T = ASC(T$) : REM TROVA IL VALORE ASCII DEL
     CARATTERE
1060 IF T > = 97 THEN LET T = T — 32 : REM CONVERSIONE
     IN MAIUSCOLA
1070 LET T$ = CHR$(T)
1080 LET P$ = P$ + T$ : REM P$ È LA NOSTRA VARTEMP$
1090 NEXT L : REM FINE DEL CICLO
1100 LET N$ = P$ : REM A SOLO SCOPO DI VERIFICA
2000 END : REM A SOLO SCOPO DI VERIFICA, SOSTITUIRE
     CON
2010 REM UNA 'RETURN' NEL PROGRAMMA DEFINITIVO
```

Come si può notare dal listato, in questo modulo di programma sono state inserite delle linee a puro scopo di verifica (la 1010, la 1100 e la 2000), molte frasi REM di commento ed un'istruzione END. Una volta verificato il corretto funzionamento, le linee di verifica vengono tolte e la END sostituita con una RETURN.

Esercizi

* Si sviluppino le altre parti del programma, fino allo stadio di codifica in BASIC. È buona norma usare parole in stampatello per le funzioni (INIZIO, CICLO, ecc.) e scrivere in corsivo le altre parti dei commenti.

* Si verifichi il corretto funzionamento di ciascun modulo (inserendo opportune linee di verifica) e, una volta soddisfatti del funzionamento, modificare ciascun modulo per uso come subroutine.

Esercizi di riepilogo

Gli esercizi che seguono non contengono alcun "trabocchetto", né introducono nuove funzioni: sono utili per ripassare gli aspetti principali del BASIC finora trattati.

* Si scriva un programma che richieda l'immissione di due numeri, li sommi e ne visualizzi il risultato.

* Si assegnino due parole (stringhe) a due variabili alfanumeriche. Si faccia in modo da "concatenare" le due parole, assegnando il risultato ad una terza variabile, anch'essa alfanumerica, e la si visualizzi.

* Si scriva un programma che conti il numero di caratteri di una parola digitata da tastiera e visualizzi un messaggio del tipo "LA PAROLA DIGITATA CONTIENE * CARATTERI" (* indica il numero di caratteri).

* Si scriva un programma che, digitato un carattere da tastiera, visualizzi il codice ASCII decimale ad esso equivalente.

* Si scriva un programma che, sollecitata l'immissione di una parola da tastiera, visualizzi il seguente messaggio "L'ULTIMO CARATTERE DELLA PAROLA È *" (dove * rappresenta, appunto, l'ultimo carattere).

* Si scriva un programma che, dopo aver richiesto l'immissione di un nominativo, trovi lo spazio separatore tra nome e cognome e visualizzi un messaggio del tipo: "LO SPAZIO SEPARATORE È ALLA * POSIZIONE" (* rappresenta il numero risultante).

* Si scriva un programma che conti il numero di parole in una frase immessa da tastiera e visualizzi il risultato.

* Si provi a visualizzare i caratteri di codice ASCII compresi tra 128 e 255 (decimali), impiegando la funzione CHR$( ) all'interno di un ciclo.

* Si scriva un programma che, dopo aver richiesto l'immissione di un numero, visualizzi il carattere ASCII corrispondente. Se il numero corrisponde ad un carattere non visualizzabile (ad esempio i valori compresi tra 0 e 32), dovrà apparire un messaggio del tipo "CARATTERE NON VISUALIZZABILE!".

Basic & Basic

LYNX

Il Lynx ha una speciale funzione BASIC, la UPC$(A$), che converte la stringa A$ in caratteri maiuscoli, così il programma si riduce a:

```
1010 INPUT "DIGITA UN NOMINATIVO";N$
1020 LET N$ = UPC$(N$)
2000 PRINT N$
2010 REM INSERIRE QUI LA "RETURN"
```

SPECTRUM

Il programma, per lo Spectrum, risulta:

```
1010 INPUT "DIGITA UN NOMINATIVO";N$
1020 LET P$ = ""
1030 FOR L = 1 TO LEN N$
1040 LET T$ = N$(L)
1050 LET T = CODE T$
1060 IF T > 97 THEN LET T = T — 32
1070 LET T$ = CHR$ T
1080 LET P$ = P$ + T$
1090 NEXT L
2000 LET N$ = P$
2010 PRINT N$
2020 REM INSERIRE QUI LA "RETURN"
```

DRAGON 32

Il programma funziona, ma i caratteri minuscoli sono presenti soltanto sulla stampante, quindi il problema della conversione, in realtà, non sussiste

BBC Micro

Sostituire la linea 1010 con:

```
1010 INPUT "DIGITARE UN NOMINATIVO", N$
```

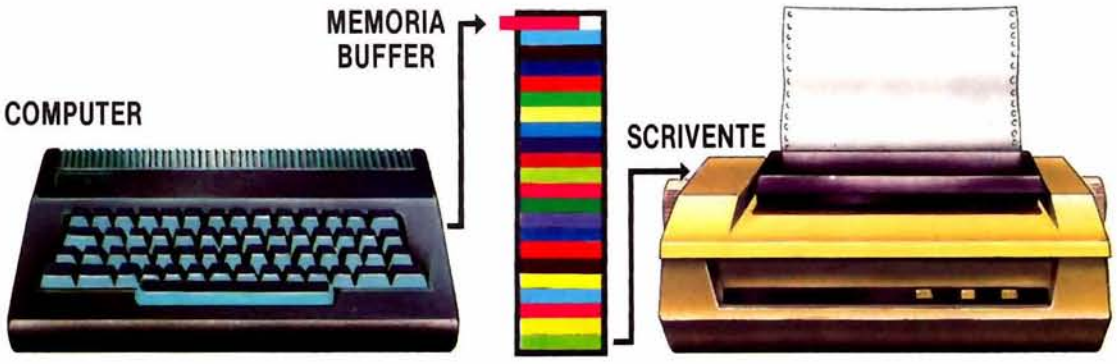
Cosa sono i "buffer"?

Il computer, per comunicare con le periferiche più lente, spesso impiega un sistema di compensazione

I respingenti (in inglese "buffer") che si trovano sui treni servono per "ammorbidire" gli impatti, assorbendo l'energia prodotta grazie a molle o a pistoni. Anche il computer ha dei buffer, che, per alcuni aspetti, funzionano come quelli di un treno, aiutando le varie parti del sistema a "convivere" senza bruschi urti.
Il termine buffer viene adoperato nell'informatica per indicare due cose completamente diverse fra loro. Per il programmatore la parola buffer significa un particolare uso di una parte della memoria, mentre per il progettista significa un tipo di amplificazione del segnale elettrico. Il secondo caso verrà trattato a parte.

I buffer di memoria

Prendiamo come esempio un programma di word processing nel quale sia possibile, fra l'altro, spostare un brano di testo, conservato in una parte della memoria, ad un'altra parte. Il testo è composto di caratteri dell'alfabeto, di spazi e di certi caratteri "non-stampabili" o "di controllo" (come il RETURN o <CR>). Tutto il testo viene rappresentato, in memoria, secondo il codice ASCII, sotto forma binaria. Per ogni carattere è necessario un byte di memoria. Quando si spostano i caratteri del brano di testo in questione dalla posizione originale nella memoria a quella nuova, si deve poter disporre di un deposito temporaneo (in memoria), ove appoggiare il brano. La parte di memoria riservata a questo compito di deposito temporaneo viene chiamata buffer.
Facciamo un altro esempio: la stampa di un documento editato usando il word processor. Il documento potrebbe consistere di 15.000 o più diversi caratteri, ma, logicamente, non è concepibile trasmetterli tutti in una sola volta alla stampante, poiché la maggior parte delle stampanti è in grado di stampare soltanto 80 caratteri al secondo. Per risolvere questo problema, una parte della memoria del computer è riservata come "buffer di stampa", sotto il controllo del software del word processing. Il software provvede a riempire questo buffer con i caratteri da stampare per poi trasmetterli alla stampante, approssimativamente alla stessa velocità con cui opera la stampante, consentendo però, nel frattempo, di lavorare ad un altro testo.
La capacità del "buffer di stampa" può variare da pochi byte (a volte appena 128 o 256 byte), fino a diversi Kbyte, ma il principio di funzionamento non cambia: un gruppo di caratteri ASCII viene depositato nel buffer e, successivamente, alla stampante viene inviato un carattere alla volta.
Ovviamente, il primo byte introdotto nel buffer sarà anche il primo ad essere stampato e questo tipo di buffer viene definito FIFO (First In First Out = Il primo ad arrivare il primo ad uscire). Quando tutti i caratteri sono stati estratti dal buffer, il software preleva un successivo blocco di caratteri destinati alla stampante. I buffer FIFO sono una caratteristica comune a quasi tutto il software per computer.
Vengono usati ogni volta che esiste un'incompatibilità di velocità, non solo fra il computer e la stampante ma anche fra il computer e l'unità a disco o fra il computer e la tastiera. La velocità fenomenale con cui opera il computer di solito significa che, normalmente, esso è in grado di identificare quali tasti siano stati digitati in meno tempo di quanto occorra per digitarli, ma può accadere che il computer, momentaneamente impegnato in una lettura o scrittura su disco, non sia in grado di occuparsi dei tasti e visualizzarne il corrispondente carattere sullo schermo.
In questo caso si ricorre all'impiego di un "buffer di tastiera", incorporato nel sistema



Un deposito temporaneo

I buffer vengono frequentemente impiegati frapponendoli tra il computer e la stampante, per ovviare al fatto che quest'ultima non è in grado di stampare i caratteri alla stessa velocità con la quale vengono inviati dal computer. Pertanto, intere sequenze di caratteri vengono velocemente scaricate nel buffer, e solo ad un suo eventuale riempimento il computer viene avvertito di sospendere il trasferimento, altrimenti esso è libero di svolgere altre operazioni. I circuiti collegati al buffer provvedono ad inoltrare i caratteri accumulati alla stampante. Questo sistema permette di evitare lunghe ed inattive pause di lavoro, in attesa che vengano stampati testi o tabulati



La quantità d'acqua che, dal secchio, viene versata nel serbatoio, è molto più grande di quella che fuoriesce dal rubinetto: in modo analogo, il computer "versa" i dati dentro il buffer che li inoltra ad una velocità pari a quella con cui opera la stampante

operativo del computer, nel quale vengono depositati i codici dei caratteri digitati: il computer li presenterà sul display non appena è in grado di farlo. Normalmente, l'utente non si accorge minimamente di questa operazione, ma con alcuni sistemi operativi o con certi tipi di software applicativo, nel quale l'elaborazione dei dati è considerevole, ci può essere un leggero ritardo fra la pressione di un tasto e l'apparizione del carattere digitato sullo schermo.

Buffer di memoria aggiuntivi

Chi possiede una stampante avrà probabilmente notato quanto sia lenta la stampa di un testo, specialmente se questo è molto lungo. Con la maggioranza dei sistemi operativi in uso oggi, il computer non può svolgere altre operazioni contemporaneamente alla fase di stampa, quindi anche l'operatore è costretto a lunghe pause di snervante attesa finché la stampa non è conclusa. Molte ditte hanno messo in commercio dei "buffer aggiuntivi", ossia delle "scatole nere" da collegare tra il computer e la stampante. Anche se la velocità di stampa resta inalterata, il computer, grazie a questo accessorio, scarica i testi da stampare molto velocemente, e può essere impiegato per altre operazioni senza costringere a lunghe attese. Il principio di funzionamento è molto semplice: nella scatola è contenuta una memoria aggiuntiva (a volte anche 16 Kbyte) gestita da un programma incorporato. Spesso la stampante è dotata di un buffer incorporato, ma raramente questo supera una capacità di 2 Kbyte, quindi si riempie piuttosto velocemente. Il buffer aggiuntivo, avendo una maggiore capacità, immagazzina più caratteri. Se, addirittura, il testo da stampare può essere contenuto per intero in questo buffer, il computer è quasi immediatamente libero di dedicarsi ad altre operazioni. È il buffer aggiuntivo che provvederà ad inviare, ad opportuni intervalli, i caratteri alla stampante.

La memoria viene spesso adoperata come se fosse un contenitore o uno scaffale per la conservazione dei programmi e dati, ma è anche possibile organizzarla, invece, in "pile" (stack) oppure in buffer. Uno stack è una struttura LIFO (Last In First Out = ultimo ad arrivare primo ad uscire) mentre il buffer è una struttura FIFO.

L'esempio, al quale spesso si ricorre per descrivere lo stack, è quello di paragonarlo ad una pila di piatti in un ristorante self-service. Ogni tanto vengono aggiunti nuovi piatti alla pila, ma è sempre l'ultimo arrivato (quello posto in cima alla pila) ad essere portato via per primo. Lo stack assomiglia al buffer in quanto è anch'esso una porzione di memoria usata in modo temporaneo: l'unica differenza sta nell'ordine con cui i dati sono inseriti e prelevati. Gli stack vengono usati moltissimo, internamente, dai linguaggi ad alto livello (negli interpreti BASIC, per esempio) per depositarvi temporaneamente dei dati da riutilizzare in seguito.

Si consideri questo breve frammento di programma BASIC:

```
FOR X = 1 TO 10
PRINT "X = ";X
FOR Y = 1 TO 10
GOSUB 7000
NEXT Y
PRINT "C$ = "; C$
NEXT X
```

Questo è un esempio di cicli FOR-NEXT nidificati. Quando il BASIC giunge alla seconda istruzione FOR, deve "ricordarsi" il valore dell'indice del primo ciclo, cosicché lo deposita (in inglese viene usato il termine "push", letteralmente "pigiare") nella pila. Terminata l'esecuzione del ciclo interno, il valore viene "ripreso" (in inglese il termine è "pop") dalla cima della pila e l'esecuzione prosegue. Poiché si possono nidificare i cicli anche a molti più livelli, sarà necessaria una locazione di stack per ciascun ciclo e le informazioni, così richiamate, devono risultare nel corretto ordine.

I buffer, invece, organizzano la memoria in modo che il primo dato entrato è il primo ad uscire e questo metodo è molto utile, ad esempio, nelle operazioni di input e di output, dove spesso occorre interfacciare apparecchi funzionanti a velocità diverse tra loro. Per esempio, la funzione di input del BASIC opera, generalmente, accettando intere righe di caratteri, terminate da un carattere RETURN (o <CR>), mentre l'interprete, spesso, elabora la riga immessa ad un carattere per volta. Per impiegare i buffer è necessario l'uso di un puntatore (pointer), che tenga conto della posizione, all'interno del buffer, dove verrà depositato il successivo carattere. Un puntatore consiste solitamente in uno o più byte ed il valore in esso contenuto incrementa ogni volta che si deposita un nuovo carattere nel buffer.

I buffer per segnali

I circuiti logici del computer operano a livelli TTL (Transistor-Transistor-Logic), ossia il livello 1 corrisponde ad una tensione di 5 volt, mentre il livello 0 a 0 volt. Tuttavia, la CPU non è in grado di pilotare direttamente dei circuiti nei quali circolino correnti molto alte. Si ricorre quindi all'impiego di speciali buffer, paragonabili idealmente a dei piccoli trasformatori, che convertono appropriatamente i segnali. I "signal buffer" sono anch'essi dei minuscoli chip, in grado di controllare, generalmente, fino a sei diversi segnali

I computer analogici

Oltre ai computer digitali, esistono altri tipi di apparecchi, non meno utili, sviluppati per applicazioni particolari



per concessione della PHYSICAL & ELECTRONICS LABS. LTD.

Il computer analogico

Il pannello di un computer analogico assomiglia ad un vecchio centralino telefonico. La "programmazione" non avviene mediante linguaggi simili al BASIC: innumerevoli cavetti collegano le varie prese poste su di esso, permettendo di "costruire" i più disparati circuiti elettronici

Esistono due distinti tipi di computer: fino ad ora, abbiamo trattato, principalmente, solo i computer digitali, così chiamati perché elaborano dati, appunto, in forma digitale. L'altro tipo di computer, molto più anziano, comprende gli apparecchi analogici.

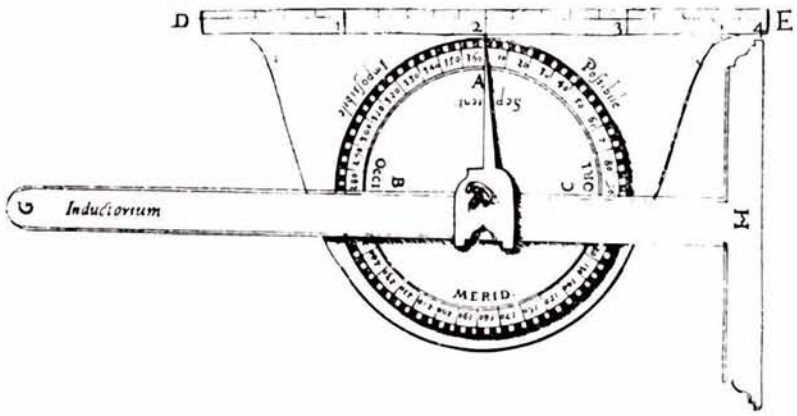
Un semplice esempio di dispositivo analogico è il tachimetro, nel quale lo spostamento di una lancetta è proporzionale (con continuità e quindi in forma analogica) alla velocità di un veicolo. Nei moderni computer analogici si impiegano componenti comuni a molti elettrodomestici: transistor, condensatori, resistenze, bobine, ecc.

Agli albori dell'elettronica, gli studiosi si accorsero che il comportamento di alcuni componenti elettrici assomigliava, in certi casi, a quello di certe strutture meccaniche. Un esempio di tale analogia è costituito da un circuito elettronico a risonanza, composto da un'induttanza magnetica e da un condensatore: il comportamento della corrente che scorre in esso è del tutto simile a quello di un peso sospeso ad una molla. Matematicamente, i due "modelli" sono identici.

Alcuni dispositivi analogici rassomigliano molto al modello reale che essi intendono simulare: nelle gallerie del vento si adoperano riproduzioni in scala di aerei o di automobili. In altri casi, invece, il modello analogico è del tutto diverso: la realtà può spesso essere rappresentata con una serie di formule matematiche o un particolare fenomeno fisico esser riprodotto con un circuito elettrico.

Le macchine calcolatrici analogiche hanno assunto un certo rilievo a seguito dell'invenzione del regolo calcolatore, dovuta nel 1630 a William Oughtred. Lo scorrimento di due righelli affiancati, sui quali erano riportate due scale numeriche, graduate logaritmicamente, permise di eseguire moltiplicazioni con grande facilità.

Alla fine del XIX secolo, Lord Kelvin inventò un ingegnoso strumento meccanico per calcolare l'entità delle maree nei porti.



Il planimetro

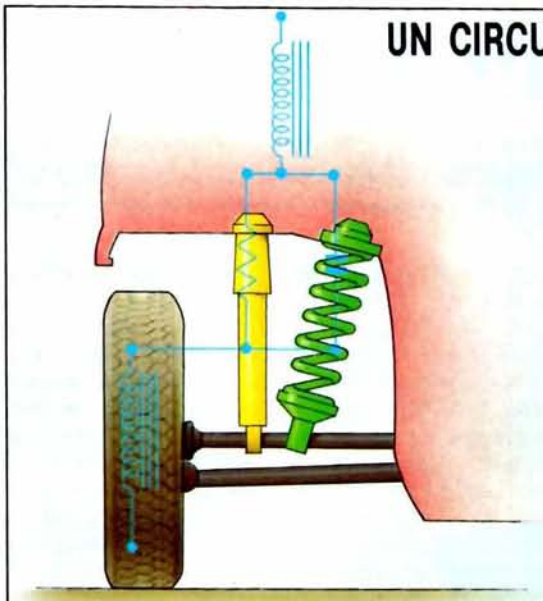
Strumenti analogici, come questo planimetro, hanno rivoluzionato il lavoro dei geometri: non più complicate misurazioni e difficili calcoli matematici, ma un semplice allineamento di qualche indice permette di dedurre i dati di una "triangolazione". Ancora oggi, apparecchiature molto simili vengono impiegate per calcolare l'area di superfici irregolari

Successivamente, nel 1930, venne costruita in America una macchina elettromeccanica per la soluzione di equazioni differenziali (il tipo di equazioni con le quali ci si scontra ogniqualvolta si rappresenta matematicamente il mondo reale).

I primi computer analogici

Il primo computer analogico totalmente elettronico entrò in funzione nel 1947, poco dopo la nascita dei primi computer digitali.
Abbiamo già spiegato (a pag. 68), come vengano eseguite le operazioni aritmetiche sui computer digitali, sfruttando porte logiche. Gli apparecchi analogici sfruttano le caratteristiche della corrente per l'esecuzione dei calcoli. Facciamo un esempio: se lungo un filo elettrico scorre una corrente di 4 ampère e lungo un altro filo una corrente di 5 ampère, unendo i due, otteniamo una corrente di 9 ampère, ossia la somma delle due correnti; ecco un semplice "sommatore" elettrico.
Anche complicate funzioni matematiche, spesso, trovano una semplice rappresentazione ricorrendo ad elementari circuiti elettrici.
I computer analogici non vengono programmati con le stesse modalità di quelli digitali: occorre, infatti, costruire un circuito specifico per ogni modello da rappresentare. Tutti i componenti di un computer analogico sono montati su di un pannello, sul quale si trova una quantità incredibile di manopole e di spinotti, che permettono di combinare i vari componenti nelle strutture più disparate. Questi pannelli ricordano vagamente quelli impiegati dalle centraliniste agli inizi del secolo.
Nei computer analogici, le quantità fisiche, quali la velocità o la forza, vengono rappresentate mediante correnti elettriche, mentre altri elementi (la massa di un'auto o la potenza di un ammortizzatore, ad esempio) vengono simulati mediante componenti elettronici.
Risulta evidente che la principale differenza tra i due generi di computer consiste nel tipo delle informazioni in gioco: negli analogici, le misure possono variare con continuità, mentre in quelli digitali le misure variano sempre a "gradini", ossia in modo cosiddetto "discreto".
Affinché un computer digitale possa interpretare grandezze analogiche, è necessario l'impiego di speciali convertitori.
L'enorme vantaggio dei computer digitali, rispetto a quelli analogici, è quello di poter elaborare o trasmettere dati senza pericolo di perderne la qualità: avendo, infatti, soltanto due livelli (0 od 1) è difficile che un particolare dato venga interpretato male. Con grandezze analogiche, invece, il rischio è di gran lunga superiore: basta una lieve distorsione nel segnale e la misura non è più attendibile.
Di contro, i computer analogici hanno una "risposta" più immediata alle variazioni nei segnali immessi: non viene perso tempo a convertirli in formato binario ed, eventualmente, a riconvertirli in un momento successivo. In alcune applicazioni questa caratteristica è di estrema importanza, come nel caso dei piloti automatici negli aerei: sebbene i computer digitali siano incredibilmente veloci, non lo sono sufficientemente per compensare le improvvise variazioni che influiscono sul volo di un aereo.
I computer analogici sono largamente impiegati nell'industria, in tutte quelle applicazioni ove è necessario operare regolazioni accurate di meccanismi. Nell'industria chimica, per esempio, vengono spesso adoperati per dosare con precisione i vari componenti dei prodotti. I computer analogici sono destinati ad aver una minore diffusione dei loro "cugini" digitali: non si prevedono, almeno per il momento, impieghi tali da giustificare la produzione di home computer analogici su larga scala. Trent'anni fa, quando i computer digitali erano enormi e molto costosi, si ragionava diversamente, ma l'avvento dei microprocessori ha decisamente portato ad una situazione ben diversa.

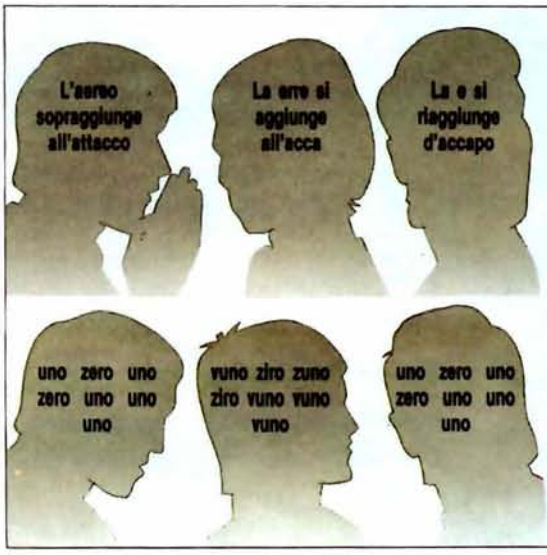
UN CIRCUITO ANALOGICO



Il sistema di sospensioni, nelle automobili, consiste in un ammortizzatore e in una robusta molla. Un sobbalzo improvviso viene assorbito dalla molla e le oscillazioni successive vengono attutite dall'ammortizzatore. Non è facile, per gli ingegneri, individuare, meccanicamente, le caratteristiche di questi due elementi per ottenere le condizioni ottimali di guida. Si è ricorsi, quindi, alla simulazione con modelli elettronici: un voltaggio fluttuante viene applicato ad un circuito elettrico (vedere figura) ed i valori dei vari componenti vengono variati fino ad ottenere, in uscita, lo smorzamento delle fluttuazioni

KEVIN JONES

PASSA LA PAROLA!



Le distorsioni dei segnali analogici attraverso un circuito elettrico si accumulano, come accade nel gioco del "telegrafo senza fili": la frase risultante risulta ben diversa dall'originale!

Nei circuiti digitali, essendoci soltanto i due valori (0 ed 1), eventuali distorsioni possono essere facilmente individuate ed eliminate nei seguenti passaggi

Herman Hollerith

Dall'Ufficio del Censimento alla IBM: un inventore col senso degli affari

Herman Hollerith nacque in America nel 1860. Dopo essersi laureato alla Columbia University, venne assunto dall'Ufficio del Censimento Nazionale degli Stati Uniti, con il compito di compilare le statistiche derivanti dal Censimento del 1880. Il lavoro veniva eseguito manualmente e progredì così lentamente che, 10 anni più tardi, quando stava per essere fatto un nuovo censimento, l'ufficio stava ancora tabulando i risultati di quello precedente. Hollerith intuì che vi erano grosse possibilità per individui con un poco d'inventiva: per potersi meglio aggiornare, e sviluppare le proprie capacità creative lasciò l'Ufficio del Censimento per entrare nell'Ufficio Brevetti a Washington.

La sua prima invenzione fu la codificazione dei dati del censimento su nastro di carta: la striscia di carta veniva suddivisa in sezioni, una per ogni specifico dato. Ad esempio: un foro praticato nella sezione relativa al sesso dell'individuo, indicava "sesso maschile", mentre l'assenza di tale perforazione indicava "sesso femminile". Questi fori venivano letti ed interpretati, successivamente, da una speciale macchina.

I primi brevetti di Hollerith uscirono nel 1884 e durante gli anni che seguirono egli migliorò sensibilmente il sistema, che impiegò nell'elaborazione di statistiche relative alla salute degli abitanti delle città americane ed in applicazioni richieste dall'amministrazione dell'esercito.

Cinque anni più tardi perfezionò il suo sistema, sostituendo al nastro originale una scheda individuale per ciascuna persona. Le schede erano della stessa misura di una banconota da un dollaro (probabilmente perché i macchinari già esistenti per la manipolazione di banconote si prestavano assai bene anche a questa nuova applicazione). I fori, inizialmente, erano tondi e venivano praticati con normali obliteratori, del tutto simili a quelli impiegati dai controllori sugli autobus. Più avanti, comunque, per poter includere un maggior numero di dati su ciascuna scheda, si adottarono fori quadrati, di 6 mm. per lato. Il principale vantaggio delle schede individuali, rispetto al nastro continuo, stava nel fatto che i dati potevano essere anche selezionati e non solo totalizzati. Per sapere, ad esempio, quante donne di 80 anni d'età vivessero a New York, bastava far estrarre, dalla macchina, le schede che contenevano una particolare combinazione di fori, corrispondente ai tre requisiti (sesso femminile, età 80 anni, abitante a New York). Le prime macchine erano in grado di produrre soltanto un semplice conteggio, ma più tardi, con l'intenzione di rendere più efficiente il sistema, Hollerith introdusse l'addizione e altre semplici operazioni aritmetiche, dimostrando così che la macchina era in grado di sostituire, con grande vantaggio, l'uomo.

Il successo, sul piano commerciale, arrivò nel 1889, quando il Dipartimento dei Censimenti indisse una gara d'appalto per la fornitura di macchinari da impiegare per l'elaborazione dei dati relativi al successivo censimento. I vari sistemi vennero sottoposti ad una verifica, utilizzando i dati del precedente censimento ed il sistema inventato da Hollerith vinse la gara. Le sue macchine erano protette da brevetto ed egli sfruttò questa situazione di "monopolio", chiedendo al Governo 65 cents per ogni 1000 schede elaborate. Sebbene il lavoro implicasse l'assegnazione di una scheda individuale a ciascun abitante degli Stati Uniti, grazie alle macchine di Hollerith esso venne portato a termine in soli 2 anni. Hollerith poté annunciare che la popolazione americana era composta da 56 milioni di individui, regolandosi di conseguenza quando inviò la sua fattura al Governo!

Quando giunse il momento di elaborare i dati del censimento del 1900, i suoi macchinari erano molto più efficienti, ed egli mantenne invariata la tariffa. Quando, infine, i suoi brevetti scaddero il Governo si rivolse ad altre ditte, ma Hollerith ebbe la meglio sulla concorrenza fondando una società, che divenne poi la International Business Machines (IBM). Attualmente la IBM è una delle maggiori produttrici di elaboratori elettronici del mondo, con un volume d'affari annuo che si aggira sui 20 miliardi di dollari.

1860
Herman Hollerith nasce a Buffalo nello Stato di New York

1879
Si laurea all'Università di Columbia e diventa assistente nell'Ufficio di Censimento Nazionale degli Stati Uniti

1882
Si iscrive all'Istituto di Tecnologia del Massachusetts per eseguire ulteriori ricerche

1883
Svolge la sua opera nell'Ufficio Brevetti a Washington

1884
Chiede il suo primo brevetto per la rappresentazione di dati su nastro perforato

1887
Il suo sistema viene adoperato per l'elaborazione delle statistiche riguardanti la mortalità nella città di Baltimora

1889
Registra un brevetto per la scheda perforata individuale

1890
Vince la gara d'appalto per la fornitura di macchine per l'elaborazione del censimento del 1890. Gli viene conferita una laurea "honoris causa" dall'Università di Columbia per il suo operato nell'elaborazione delle statistiche

1900
Inventa una nuova generazione di macchine perfezionate

1901
Il nuovo macchinario viene adoperato per il Censimento dell'Agricoltura

1911
Costituisce una società per azioni, la Tabulating Recording Company

1924
La Società cambia la ragione sociale e diventa la International Business Machines (IBM)

1929
Hollerith muore a Washington

	Errata	Corrige
pag. 4 (3° riquadro da sinistra)	Scheda 1/0 L'unità di 1/0...	Scheda I/0 L'unità di I/0...
pag. 38 (1° colonna, riga 22)	35 IF A> THEN GOTO 80<CR>	35 IF A> 999 THEN GOTO 80< CR>
pag. 136 (programma)	... 1060 RETURN 2000 REM VISUALIZZAZIONE NOMINATIVI ...	... 1060 RETURN 1100 REM SUBROUTINE DI SOSTITUZIONE (SWAP) 1110 LET T$=A$(P) 1120 LET A$(P)=A$(P+1) 1130 LET A$(P+1)=T$ 1140 LET S=1 1150 RETURN 2000 REM VISUALIZZAZIONE NOMINATIVI ...