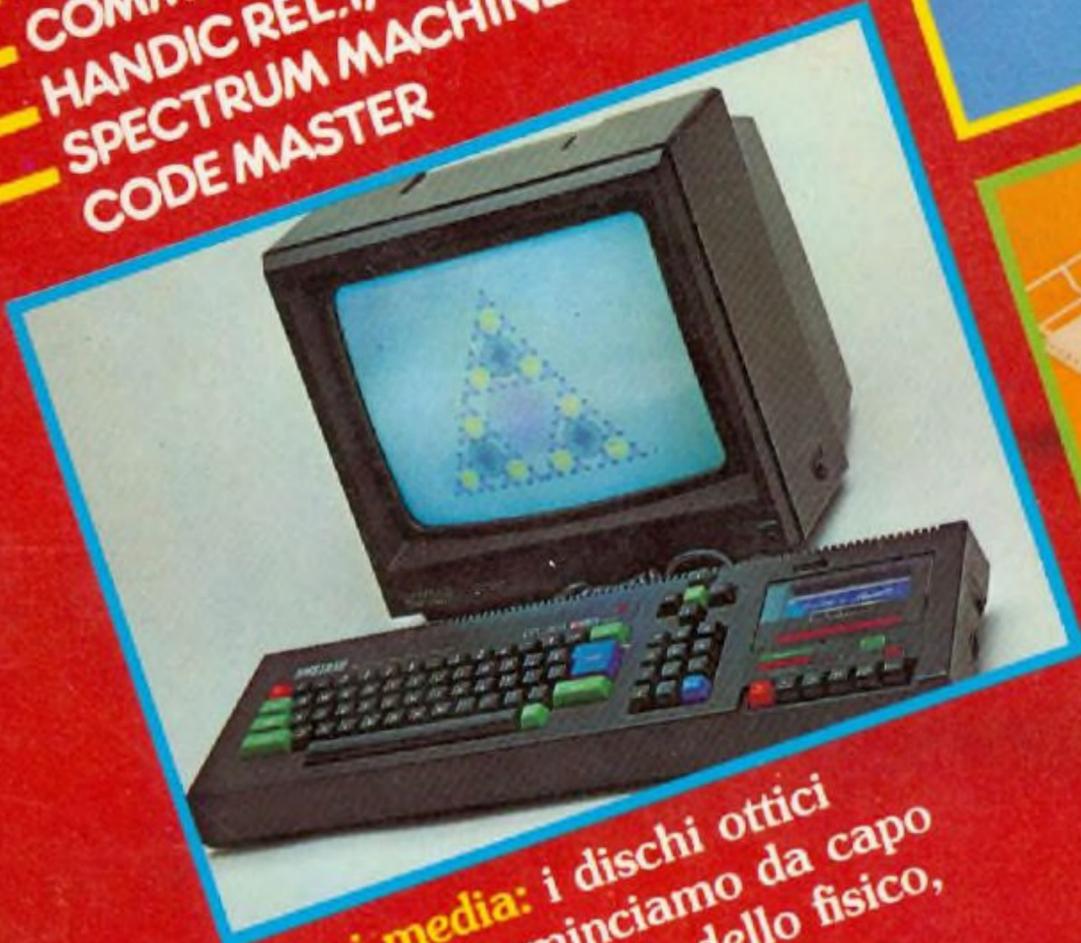


microcomputer[®]

HARDWARE & SOFTWARE
DEI SISTEMI PERSONALI

IN PROVA:

- AMSTRAD CPC 464
- MEMOTECH MTX 512
- COMMODORE MPS 802
- HANDIC REL. I/O PER VICE 64
- SPECTRUM MACHINE
- CODE MASTER



Nuovi media: i dischi ottici
Grafica: ricominciamo da capo
La simulazione: modello fisico,
 matematico, statistico
MBASIC: la gestione dei file
Indice analitico 1984

HANDICAPPATI MOTORI:
 un computer per la vita



natale un tv color
 per il tuo
 computer



microcomputer[®]

49

I dischi ottici



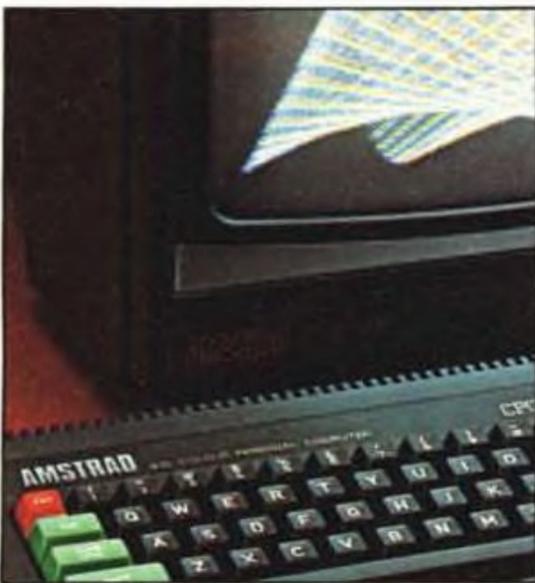
54

La presa SCART dei TV color



70

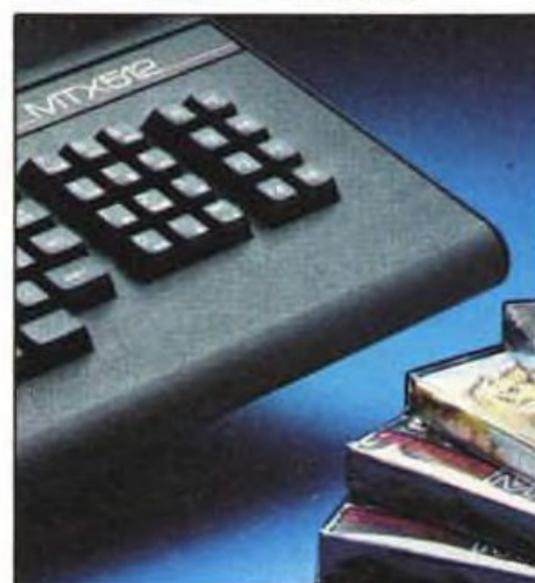
Amstrad CPC 464



- 4 Indice degli inserzionisti
- 7 Indice analitico '84
- 10 Il futuro dell'Home Computer
Paolo Nuti
- 12 Posta
- 27 News
- 36 Libri
- 43 Handicap e computer
Un'interfaccia per vivere
Marco Marinacci
- 49 I dischi ottici - *Paolo Nuti*
- 54 La presa SCART dei TV color
Valter Di Dio
- 58 MCgiochi - I giochi del computer
Corrado Giustozzi
- 60 MCgiochi: recensioni
- 70 Amstrad CPC 464 - *Leo Sorge*
- 78 Memotech MTX 512
Maurizio Bergami
- 84 Commodore MPS 802
Andrea De Prisco
- 90 Handic Rel, I/O per Vic e 64
Tommaso Pantuso
- 94 Dream Software: Machine code for beginners - *Maurizio Bergami*
- 98 Grafica - *Francesco Petroni*
- 104 La simulazione - *Valter Di Dio*
- 110 Parla più Forth - *Raffaello De Masi*
- 115 MCmicrofacile - *Tommaso Pantuso*
- 121 Il Data Base - *Andrea de Prisco*
- 127 Vic da zero + 64
Tommaso Pantuso
- 133 Commodore 64 - L'ADP Basic
Andrea de Prisco
- 141 L'Assembler dello Z80
Pierluigi Panunzi
- 144 Software MBasic - *Pierluigi Panunzi*
- 148 Software RPN - *Paolo Galassetti*
- 151 Software Apple - *Valter Di Dio*
- 155 Software Commodore 64
Leo Sorge
- 164 Software Vic 20
Tommaso Pantuso
- 170 Software Spectrum
Maurizio Bergami
- 180 Software TI-99/4A
Fabio Schiattarella
- 184 Software TI-99/4A Ext. Basic
Fabio Schiattarella
- 188 Software Sharp PC-1500
Fabio Marzocca
- 192 I trucchi del CP/M
Pierluigi Panunzi
- 194 Elenco del software disponibile su cassetta o minifloppy
- 195 Guida computer
- 216 Micromarket-micromeeting
- 223 Microtrade
- 225 Campagna abbonamenti Servizio arretrati

78

Memotech MTX 512



90

Handic Rel I/O per Vic e 64



94

Machine code for beginners





Nonostante nel settore degli home computer lo strapotere mondiale sia indiscutibilmente della Commodore, in Europa esiste anche la Sinclair, in attesa dell'annunciata offensiva MSX sul Vecchio Continente, e in Gran Bretagna esistono tanti altri marchi che producono modelli validissimi. Il numero uno è attualmente l'Amstrad CPC 464, che nella natia Albione è stato addirittura chiamato 'The Spectrum Beater', ovvero quello che sconfiggerà lo Spectrum: l'hardware è stato studiato a metà tra quello Sinclair e le nuove frontiere MSX; il software è incredibilmente evoluto, dato che, per dirne qualcuna, gestisce finestre, interrupt e periferiche direttamente da Basic! Per di più l'estetica è al contempo seria ma non distaccata, ottimizzata dal registratore integrato nel computer e dall'alimentazione localizzata nel monitor. Le 80 colonne, il CP/M, il Locomotive Basic, il Logo sono tutti motivi sufficienti a motivare un acquisto del genere, per di più ad un prezzo assolutamente basso: 998.000 lire più IVA per la versione con monitor a colori.

L'esterno

Il primo impatto con il CPC 464 è decisamente positivo. Una tastiera seria, con

Amstrad CPC 464

di Leo Sorge

paddle numerico e gestione cursori (per l'editing) separati; registratore a cassette con contagiri incorporato; cavo di alimentazione e connettore video provenienti dal monitor (quindi niente scatolette sparse!); monitor a colori da 14", con mobile dell'identica tonalità del computer: indubbiamente andiamo verso l'integrazione delle varie unità funzionali.

Andando a vedere le varie connessioni scopriamo l'interfaccia per stampante compatibile Centronics, l'uscita per disco, quella per joystick e una audio con jack stereo da 3,5 mm che consente di collegarsi ad impianti di riproduzione esterni (l'uscita è di 50 mV); comunque nel computer, accanto alla meccanica del registratore, è posto un altoparlantino da circa 3 cm, il cui volume è controllato dal potenziometro sul lato destro del computer.

La tastiera, completamente nera, è movimentata da una serie di tasti speciali colorati in tinte vive: tolto l'Escape, che è rosso (in alto a sinistra), l'Enter è un grosso tasto blu sulla destra (duplicato sul tastierino numerico), mentre tutti gli altri tasti speciali sono in verde. Il registratore è completamente controllato da software, con l'aiuto dei messaggi sullo schermo; i dati possono essere scritti a due diverse velocità, agendo sul comando SPEED TAPE che porta a 2400 baud un valore solitamente pari alla metà; in lettura la selezione della velocità è automatica.

Veramente ottima, sia per i costi che per l'estetica, la soluzione di integrare l'alimentatore nel monitor: ciò obbliga l'utente ad acquistare quello che offre la casa, ma abbiamo trovato un'ottima qualità sia nella versione a colori, in media risoluzione (le



80 colonne non sono chiare con tutte le combinazioni di colori) che in quella monocromatica ad alta risoluzione (fatta apposta per l'uso professionale), e in entrambi i casi il prezzo è così basso che pensiamo lo prendereste comunque!

Vogliamo ora accennare ad una caratteristica del sistema: in qualsiasi momento durante l'esecuzione di un programma si può avere una pausa, basta premere Escape una volta; per ricominciare bisognerà premere un qualsiasi altro tasto, poiché premendo ancora Escape si fermerebbe il processo in via definitiva (ma proseguibile con Cont). Inoltre se il computer sta lavorando si sente un buzz nell'altoparlante incorporato.

L'interno

Nonostante si tratti di un prodotto britannico, l'architettura hardware segue decisamente gli schemi statunitensi (e giapponesi): molte delle funzioni del computer, l'interfaccia d'uscita per video e stampante, nonché il generatore di suono, sono realizzate tramite circuiti specializzati, e non con un unico circuito integrato su misura (la ULA ben nota agli 'ZXers'); d'altronde la ULA c'è anche qui, ma svolge solo un numero limitato di compiti. Andiamo a vedere le singole parti.

Il microprocessore non ha bisogno di commenti: è lo Z80, attualmente il più impiegato per applicazioni di home e personal della fascia bassa, qui nella versione A con clock a 4 MHz ricavato per divisione di frequenza dal quarzo del colore (16 MHz; lo vedete in alto a destra, affiancato all'ULA). La Rom, da 32K, comprende il sistema operativo e il Basic, estesissimo: la

Costruttore:

Amstrad Consumer Electronics
Brentwood House, 169 Kings Road
Brentwood, Essex CM14 4EF

Distributore per l'Italia:

Microstar, Via Cagliari, 17 - 20125 Milano

Prezzi:

CPC 464 + monitor monocrom. L. 698.000
CPC 464 + monitor a colori L. 998.000

N.B.:

I prezzi si intendono IVA 18% esclusa.

vedete a sinistra, montata su zocchetto e contraddistinta dalla sigla Amstrad 40009. La Ram è organizzata in un banco da 64 Kbyte tramite otto chip 4164 da 64 Kbit, tutti montati su zocchetti: è veramente da encomiare la pulizia dell'intera piastra madre. Altrove nell'articolo riportiamo i sommi capi della mappa di memoria.

Veniamo adesso all'interfacciamento con il mondo esterno, ovvero alle connessioni con tastiera, video, stampante e joystick. I segnali della tastiera, ovvero i contatti provenienti da righe e colonne della matrice, vengono affidati a due diverse soluzioni, di cui una è un 74LS245, mentre per l'altro si sfrutta la porta integrata nel

chip sonoro 8912, seguendo un progetto tradizionale; di qui vengono affidati al Pio 8255, che ne ricava un codice numerico facilmente manipolabile, tanto che ogni tasto è completamente ridefinibile, e gli si può assegnare una stringa contenente più istruzioni.

L'8255 gestisce anche l'uscita per stampante, ovviamente parallela e centronics compatibile, nel senso che agli 8 bit di dati vanno aggiunti solo una massa, lo strobe e il busy: proprio quest'ultima scelta va fatta notare, in quanto è più frequente l'uso dell'acknowledge (piedino 10 del connettore Amphenol), il che potrebbe portare perdite di tempo. Tra le altre funzioni di questo integrato c'è il controllo completo del registratore a cassette, sia del suo stato che del motore.

Il segnale video è generato da un circuito integrato classico e di ottima qualità, il 6845, che seppure progettato per usi basati sul microprocessore 6502 assolve perfettamente i suoi compiti anche con la famiglia Z-80: sulla piastra si trova sotto al Pio. I segnali del CRT controller (CRT = tubo a raggi catodici = televisore) sono mostrati anche a pag. V3 del manuale, e mostrano che l'uscita prevede i tre colori fonamen-



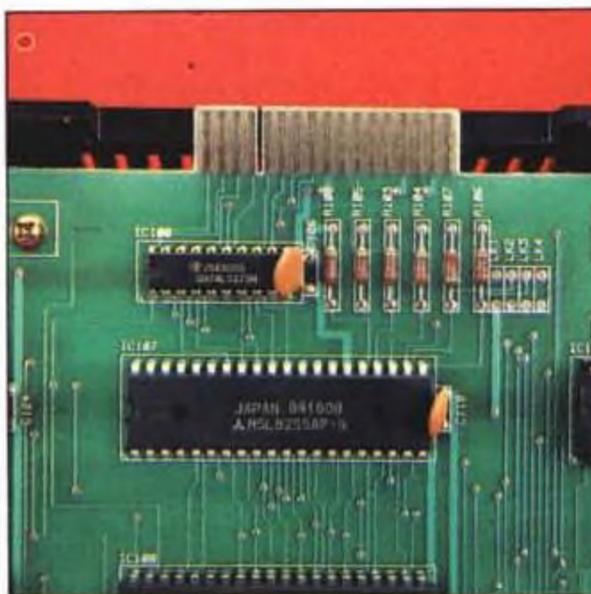
Le connessioni posteriori: da sinistra verso destra le prese sono di video, alimentazione, disco, stampante, joystick e audio.



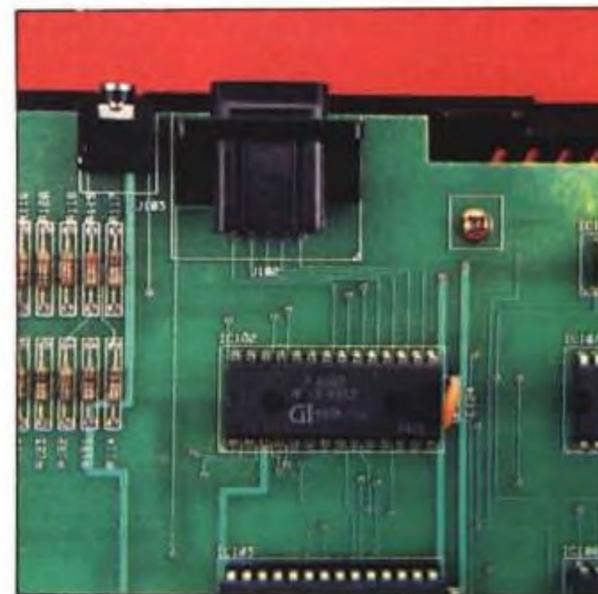
La tastiera è completa di paddle numerico e controllo separato del cursore. Finalmente il registratore è incorporato.

tali rosso, giallo e blu (RGB), più la luminosità e i sincronismi, per un totale di 5 connessioni più la massa; per il collegamento con il monitor la casa usa un DIN a 6 poli, mentre per l'alimentazione, che avviene tramite il monitor venduto insieme al computer, si usa un comune jack coassiale. Per la gestione della pagina video, che è comunque in alta risoluzione, il chip video accede a 16K di Ram — la zona più elevata, da \$C000 a \$FFFF — ed è interessante notare che il rinfresco di tutta la Ram disponibile avviene tramite questi accessi alla memoria video. Il 6845 gestisce anche una light pen, ma questo accessorio non è ancora pronto. A ben pensarci, però, il 6845 mostra almeno un punto debole: essendo della generazione precedente, non gestisce sprite via hardware, ma i progettisti hanno parzialmente colmato questa lacuna con una sapiente gestione software delle finestre (lo vedremo nel paragrafo della grafica).

Per quanto riguarda il joystick bisogna spendere due parole. I controlli vengono ricavati mettendosi in parallelo ad alcuni tasti, e il collegamento avviene tramite un connettore a vaschetta da 9 contatti, perfettamente compatibile con i soliti modelli. La Amstrad ha però previsto l'uso di due joystick tramite un unico connettore: il secondo va collegato direttamente ad uno



L'8255, una porta programmabile parallela, serve principalmente la stampante.

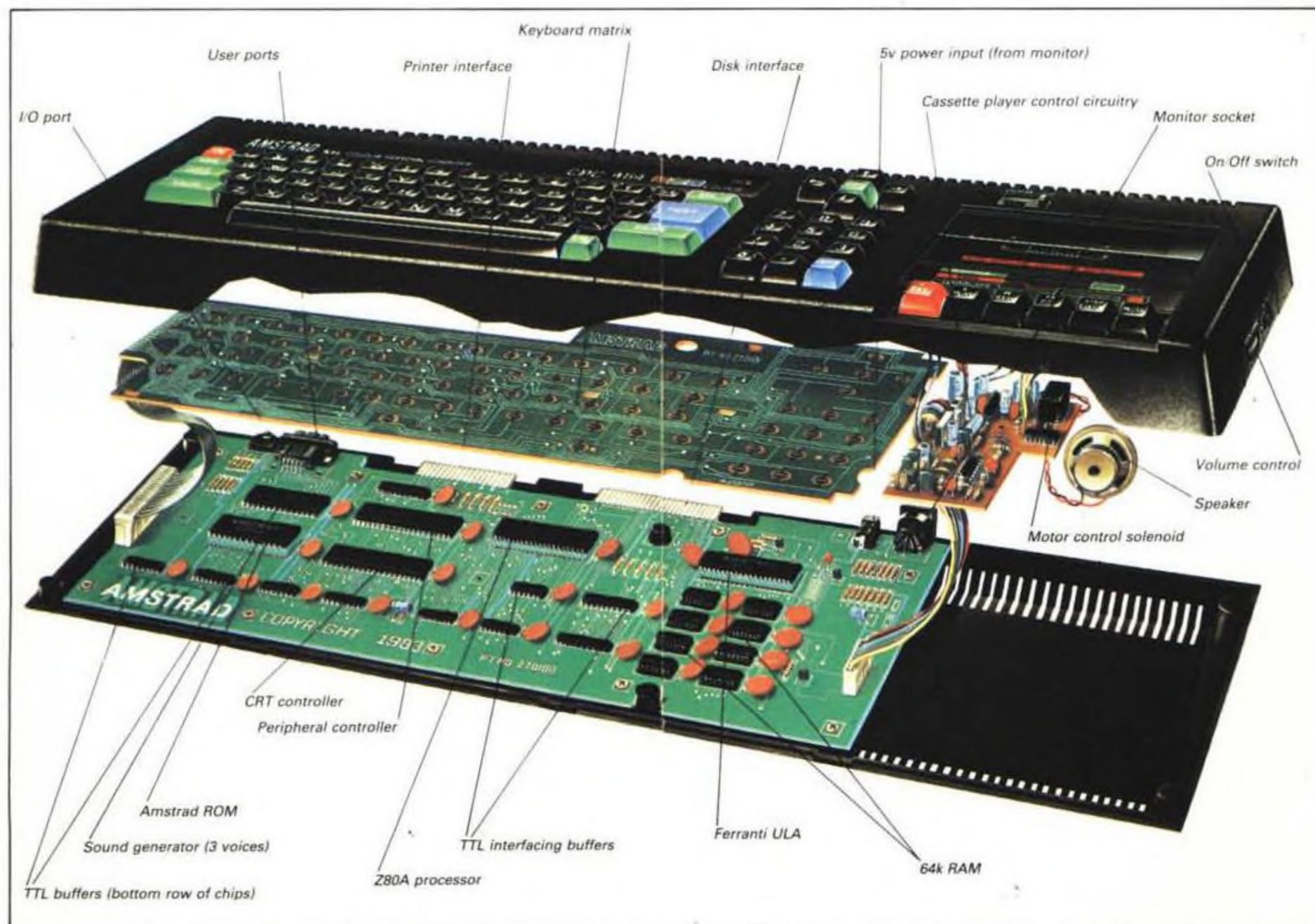


Il chip sonoro 8912 è connesso ad un jack stereo da 3.5 mm.

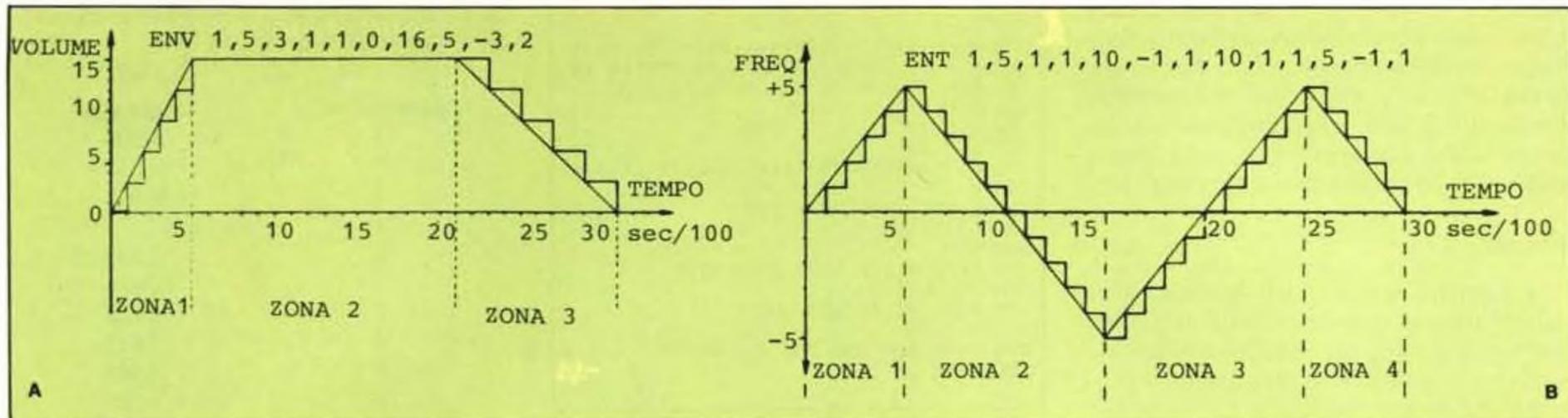
speciale della casa, che riporta una vaschetta posteriore. I segnali sono ottenuti utilizzando per entrambi gli stessi contatti relativi alle direzioni, ma cambiando sia il fuoco che la massa, per un totale di 8 piedini impiegati.

Passiamo adesso al suono, affidato all'ottimo General Instruments 8912. Si tratta di un integrato versatissimo, in grado di fornire fino a tre canali musicali (eventualmente sincronizzati, anche solo in parte) ma anche uno di rumore: nel caso del-

l'Amstrad ci troviamo davanti ad un suo intenso sfruttamento: la forma d'onda viene infatti divisa in 5 segmenti, eventualmente non tutti specificabili, il che consente un dettaglio superiore al tipico ADSR in 4 divisioni (anche se l'eventuale quinta sarà scarsamente utilizzata in problemi di sintesi musicale). Il suono viene riprodotto da un altoparlantino di piccole dimensioni e media qualità seminascolato nel vano del registratore; il volume è regolabile da un potenziometro posto lateralmente, sulla



Tratto da Personal Computer World, maggio '84.



A) Un esempio di costruzione di un inviluppo in ampiezza — comando ENV — a tre sezioni: per ognuna va specificato il numero di gradini (tutti uguali nella stessa sezione) e le loro dimensioni orizzontali e verticali.

B) Un esempio di costruzione di un inviluppo in frequenza — comando ENT — a 4 sezioni. La specificazione dei parametri avviene nello stesso modo dell'analogo ENV. Entrambi i comandi vanno iterati, per ottenere un effetto costante nel tempo.

destra. C'è anche un jack da 3,5 mm, che offre in uscita il suono dei tre canali secondo la regola:

canale sinistro = canale A + metà del canale C;

canale destro = canale B + metà del canale C;

La grafica

La prima cosa che si nota leggendo il manuale nel capitolo 'Graphic Primer' è l'elevatissima versatilità offerta dal Locomotive Basic, sia nell'indirizzamento dei punti che nei comandi messi a disposizione.

Il 6845 gestisce direttamente 16K di memoria, corrispondenti a 128K bit; la scelta di colori dipende dalla risoluzione, secondo la formuletta:

$128K / \text{memoria occupata} = n'$ bit disponibili per i colori,
da cui si ottiene il numero di colori, pari a $2^{n'}$.

A proposito dei colori va anche detto che abbiamo a disposizione una vastissima gamma di tonalità: ben 27, delle quali, a scelta — come vedremo — 16 accessibili in multicolor, 4 nel modo normale e solo due in alta risoluzione. Le diverse gestioni del video sono tre, che la casa chiama normale, multicolore e in alta risoluzione. In tutti e tre i casi lo schermo è gestito direttamente in alta risoluzione, il che, se rallenta leggermente il listing, comunque rapido, consente di mischiare a piacere grafici e scritte, cosa questa veramente utile all'hobbista. Il modo normale, o modo 1, è il valore di default che troviamo in funzione all'accensione, e consiste in una pagina testo di 25 righe da 40 colonne l'una, ogni carattere essendo iscritto in una matrice di 8×8 punti: ciò consente una risoluzione punto per punto di 320 punti orizzontali per 200 verticali, per un totale di circa 64K bit. Con una semplice divisione veniamo a sapere che disponiamo di 4 colori:

$$128K / 64K = 2 \text{ bit};$$

$$2^2 = 4 \text{ tonalità},$$

sempre da scegliere tra le 27 possibili.

Il multicolor vede una pagina grafica di

soli 160×200 punti, con un testo ridotto a 20 righe di 25 colonne, con caratteri veramente grandi!, ma offre una risoluzione cromatica di ben 16 colori: infatti

$$128K / 32K = 4 \text{ bit}$$

$$\text{da cui } 2^4 = 16.$$

L'ultimo modo è quello in alta risoluzione: 640 punti orizzontali per i soliti 200 verticali, con un display da 80 colonne per applicazioni di tipo personal e business, ovviamente in sole due tonalità. A proposito del modo 80 colonne va ricordato che

```
100 REM *****
110 REM ** questo gestisce le **
120 REM ** finestre come sprite **
130 REM *****
140 :
150 MODE 2
160 CLS
170 FOR a=2 TO 30
180 WINDOW #2, a-1,a,15,15
190 PRINT #2, " "
200 WINDOW #2,a,a+1,15,15
210 PRINT #2,CHR$(229)
220 FOR t=0 TO 99 : NEXT t
230 NEXT
```

```
100 REM *****
110 REM ** questo modula un suono **
120 REM ** sia in ampiezza che in **
130 REM ** frequenza **
140 REM *****
150 :
160 EVERY 10 GOSUB 180
170 GOTO 170
180 ENV 1,5,3,1,1,0,16,5,-3,2
190 ENT 1,15,1,1,15,-1,5
200 SOUND 1,124,30,,1,1
210 RETURN
```

dei due monitor forniti dalla casa, uno a fosfori verdi ma ad alta definizione e l'altro a colori in media definizione, chi dovesse lavorare a lungo con programmi applicativi dovrebbe provvedersi del primo, che è più nitido ed affatica molto meno la vista; per usi diversi, invece, va bene anche quello a colori — indispensabile nelle applicazioni da home: giochi, Basic, apprendimento... — ma con particolari combinazioni dei colori di sfondo e di bordo.

Una delle peculiarità più interessanti dell'Amstrad è la possibilità di aprire fino a 8 finestre diverse, anche sovrapponibili,

sullo schermo, che possono essere usate indipendentemente l'una dall'altra. L'istruzione che abilita questa funzione è

WINDOW # <numero>, si, de, su, giù
ove ovviamente si = estremo limite sinistro (minima ascissa), de = estremo limite destro (massima ascissa), su = estremo limite superiore (massima ordinata), giù = estremo limite inferiore (minima ordinata). Per poter scrivere, listare o disegnare in una di queste finestre basterà specificare il numero d'ordine (da 0 a 7) dopo l'istruzione generale; per listare nella quarta finestra basterà dunque una

LIST #3

e analogamente avverrà per le altre istruzioni.

Probabilmente l'unico grosso problema del 6845 è la mancanza di una gestione hardware delle sprite: quasi tutti i migliori computer, e tutti quelli della nuova generazione, ne hanno un certo numero, variabile in relazione anche alle loro dimensioni. A ben vedere, però, l'Amstrad ha un software in parte superiore: le otto finestre, infatti, possono essere usate come altrettante sprite di dimensioni variabili e mobili nello schermo di opportuna definizione: in tal caso si tratterà di ridefinire un certo numero di caratteri che vadano a formare la nostra immagine, quindi dimensionare una finestra opportuna, scriverci dentro i caratteri e poi spostarla per lo schermo con un ciclo for-next (o meglio while-wend con verifica dei confini dello schermo), non dimenticandoci di pulire le scritte precedenti. Un esempio di quest'idea, sfruttando una finestra da 1 solo carattere non ridefinito, viene dato altrove nel testo, in uno dei listati acclusi alla prova. Certo non abbiamo a disposizione il controllo automatico di eventuali collisioni tra sprite, come invece accade in progetti più recenti, ma gran parte del vantaggio dei folletti (sprite) è conservata.

Tra le istruzioni interessanti della grafica notiamo la

ORIGIN x,y

che ridefinisce il vertice in basso; essendoci le finestre non poteva mancare. È poi da segnalare che le varie istruzioni di Draw,

Move, Plot e Test (per i significati vedere l'elenco delle parole) vengono fornite nella doppia versione normale con r finale, che sta per 'relativo': in altre parole è possibile non far riferimento alle coordinate assolute, ma solo a quelle correnti, cosa questa molto utile in programmi di grafica.

Il suono

Le capacità musicali dell'Amstrad sono indubbiamente notevoli, tali da non poter essere gustate appieno dall'altoparlantino inserito nel mobile: consci di questo fatto, i progettisti hanno realizzato un'uscita stereofonica jack da 3,5 mm (tipo auricolare delle radio, per intenderci). Il chip dedicato è il General Instruments 8912, inserito in moltissimi home computer, anche gli MSX usano un circuito integrato analogo che si basa su tre canali di uscita sinusoidale a cui può essere sommata una certa quantità di rumore. La caratteristica più interessante è ovviamente la possibilità di costruirsi l'inviluppo delle frequenze in uscita, usando il comando ENV (ENvelope Volume) che di fatto controlla l'ampiezza dell'uscita in modo assai più preciso di quanto non possa fare la generica indicazione del volume. Non solo: dell'inviluppo si può controllare anche la frequenza, tramite l'istruzione ENT (ENvelope Tone), che segue pedissequamente le suddivisioni eventualmente date da un precedente comando ENV.

Entriamo un po' più nei particolari. I tre canali, denominati A, B e C, sono mandati al jack stereo secondo la regola

canale destro = A + metà C;
canale sinistro = B + metà C.

con un livello d'uscita decisamente costante, pari a circa 50 mV su tutta la banda.

La parola fondamentale è ovviamente Sound, seguita da ben 7 parametri. Di questi il primo identifica lo stato del chip, ovvero quali canali stanno suonando e in che condizioni. Le cose più importanti riguardano la selezione e il sincronismo degli oscillatori, che funzionano secondo la seguente regola:

| | |
|----|--------------------------|
| 1 | azione il canale A |
| 2 | azione il canale B |
| 4 | azione il canale C |
| 8 | sincronizza tutto con A |
| 16 | sincronizza tutto con B |
| 32 | sincronizza tutto con C. |

Il secondo parametro riguarda la frequenza. Per motivi hardware il valore qui impostato non è la frequenza vera e propria, bensì quella ottenuta dall'operazione

$$\text{frequenza} = 125000 / (2 \times \text{parametro});$$

a tale proposito bisogna specificare che il manuale in inglese non riporta il fattore 2 a denominatore. Poiché il parametro può variare tra 1 e 4096 si ha che il campo di frequenza varia tra 62500 e circa 15 Hz (in pratica 16), ovviamente in modo non continuo: se alle basse frequenze — diciamo minori di 1000 Hz — il salto tra due valori contigui è minimo, ed è quindi massima l'accuratezza, per valori alti la precisione decade velocemente, come si può vedere

```

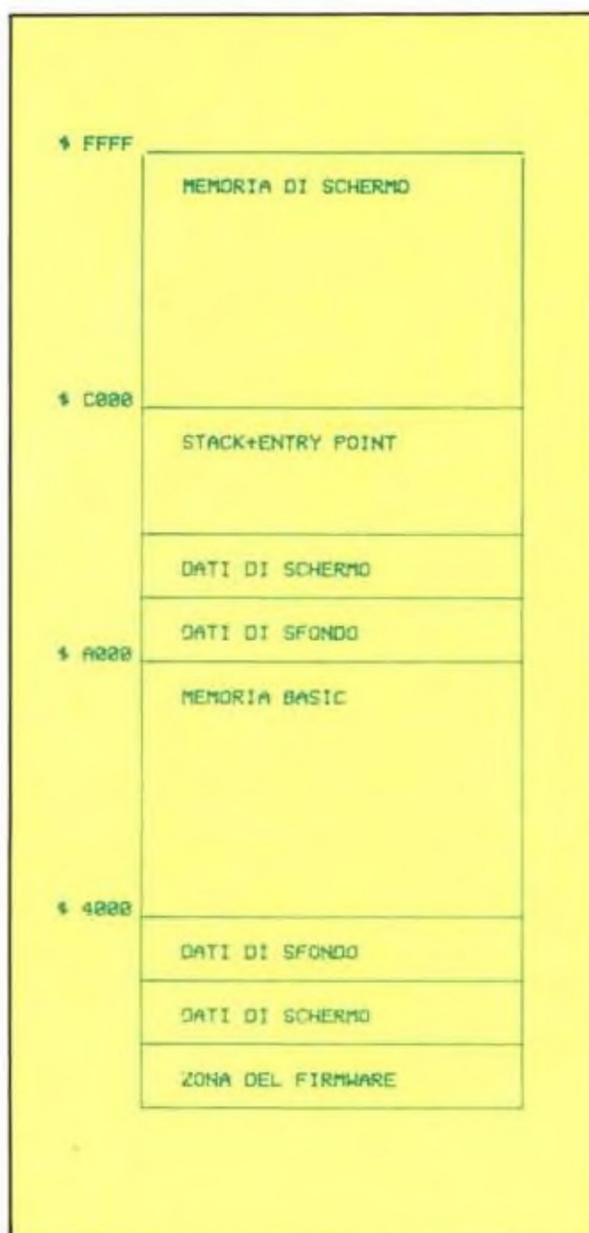
100 REM *****
101 REM ## QUESTO PROGRAMMA MOSTRA ##
102 REM ## L'USO DELL' ISTRUZIONE ##
103 REM ## ##
104 REM ## EVERY ##
105 REM ## ##
106 REM *****
110 :
120 EVERY 100 GOSUB 150
130 GOTO 130
140 END
150 READ A: IF A=99 THEN STOP
160 SOUND 1,A
170 FOR T=0 TO 200: NEXT
180 RETURN
190 DATA 100,200,300,400,500
200 DATA 99

```

```

100 REM *****
110 REM ## Questo mostra le varie ##
120 REM ## gestioni dell' errore ##
130 REM *****
140 :
150 CLS
160 ON ERROR GOTO 220
170 ON BREAK GOSUB 260
180 DATA A,B,C,D,E
190 READ A$: PRINT A$
200 FOR T=0 TO 999: NEXT
210 GOTO 190
220 REM -- ERRORE ---
230 IF ERR=4 THEN PRINT "SONO FINITI
I DATI !"
240 IF ERL<210 AND ERL>180 THEN PRINT
"... nelle linee centrali !"
250 END
260 REM -- BREAK ---
270 SOUND 1,100
280 SOUND 1,200
290 RETURN

```



La mappa di memoria, limitata ai 64K RAM; le ROM sono agli stessi indirizzi della RAM, nel primo e nel quarto blocco da 16K.

dalle frequenze che si ottengono impostando i valori da 1 a 10:

| parametro | frequenza |
|-----------|-----------|
| 1 | 62500 |
| 2 | 31250 |
| 3 | 20833 |
| 4 | 15625 |
| 5 | 12500 |
| 6 | 10416 |
| 7 | 8928 |
| 8 | 7812 |
| 9 | 6944 |
| 10 | 6250 |

e per motivi musicali risultano meglio utilizzabili le frequenze inferiori ai 2000 Hz. Impostando il valore 0 otteniamo una situazione non musicale, che ci serve per realizzare rumori puri ed effetti speciali.

La possibilità di definire l'inviluppo dell'onda è eccellente, dato che arriviamo ad una segmentazione in 5 parti. In generale quelle utili sono 4 (talvolta 3), dato che i suoni reali sono approssimabili con gran facilità e sufficiente precisione ad una spezzata di 4 lati. Per ognuno dei 5 segmenti, comunque, vanno specificati tre parametri, che identificano in modo assoluto il tipo di forma che intendiamo modellare. Poiché si tratta di far variare una grandezza (l'ampiezza) tra un valore iniziale ed uno finale, considerando che il passaggio avviene per gradini (aiutatevi con le figure) noi dovremo specificare:

- il numero di gradini;
- la loro altezza (è un'ampiezza);
- la loro lunghezza (è un tempo).

La sintassi del comando ENV è quindi ENV N, A1,B1,C1 A2,B2,C2 ... A5,B5,C5

dove i vari gruppi A,B,C possono anche non essere 5, ed N è un numero d'identificazione del comando, e va specificato sempre (è a piacere): permette la sincronizzazione con il comando ENT. Così come il precedente ENV varia l'ampiezza dei segnali in uscita (dei quali rimane fissata la frequenza tramite il comando Sound), così ENT varia la frequenza, rimanendo prefissata l'ampiezza stabilita da un precedente Sound (nel qual caso è costante) o da un ENV (nel qual caso c'è variazione nel tempo), cosa questa utile per tremoli e vibrato. Per meglio comprendere conviene studiare qualche minuto le figure che illustrano queste istruzioni e i loro campi di variazione.

Né ENV né ENT, una volta specificati, funzionano poi per sempre, ma solo una volta: per la loro corretta gestione bisognerà quindi manipolare l'interrupt con la solita Every, come mostrato in uno dei listini che presentiamo nel corso della prova.

Il resto del Basic

Prima di andare a vedere le parole vere e proprie riteniamo interessante una breve digressione. Ogni microprocessore ad 8 bit (almeno di quelli esistenti) può gestire fino

Le parole più significative del Locomotive Basic vers. 1.0

| | | | | | |
|----------|--|---------------|---|--------------|---|
| After | interrupt | Instr | cerca una stringa dentro un'altra | Round | da Randomize |
| Atn | arcotangente | Joy | legge lo stato dei joystick | Sound | arrotonda un numero |
| Auto | autonumeraz. linee | Key | assegna una stringa ai tasti | | gestisce il suono (canale, tempo, volume, involuppo) |
| Bin\$ | stringhe binarie | Key Def | ridefinisce la tastiera | Space\$ | genera spazi |
| Border | colore del bordo | Line Input | legge un'intera linea | Speed | accelera alcuni processi (lampeggio, registrazione,...) |
| Call | salto al LM | List | lista il programma | Speed Ink | accelera il lampeggio sfondo-bordo |
| Cat | cataloga (cassetta) | Locate | posiziona il cursore | Speed Key | modifica la velocità di autorepeat dei tasti |
| Chain | riunisce programmi (cassetta) | Log | logaritmo naturale | Speed Write | controlla la velocità di registrazione |
| Cint | arrotonda | Log10 | log. in base 10 | Sq | determina lo stato del processore audio |
| Clear | azzerà variabili | Lower\$ | converte al lower case | String\$ | definisce stringhe ripetute |
| Clg | pulisce la grafica | Max | numero più alto di una sequenza anche letterale | Symbol | ridefinisce i caratteri |
| Closein | chiude il file in ingresso su cassetta | Memory | ridefinisce il più alto byte del Basic | Symbol After | stabilisce il numero di caratteri ridefiniti |
| Closeout | chiude il file in uscita su cassetta | Merge | fa il merge tra il programma in memoria e uno sul nastro | Tag | mischia testo e grafica |
| Cls | pulisce le finestre | Min | numero più basso di una sequenza (anche letterale) | Tagoff | disabilita Tag |
| Creal | | Mod | resto della divisione | Test | restituisce il colore del punto specificato |
| Data | dati nel listato | Mode | selezione dell'output su schermo | Testr | come Test ma in coordinate relative |
| Defn | definisce funzione | Move | muove il cursore grafico senza disegnare | Time | tiene il tempo di accensione |
| Defint | definisce intero | Mover | come Move ma in coordinate relative | Troff | disabilita il Trace |
| Defreal | definisce reale | On | identifica diramazioni multiple | Tron | abilita la funzione di Trace |
| Defstr | definisce stringa | On Break | gestione Basic del Break | Upper\$ | converte una stringa nel suo equivalente in upper case |
| Def | definisce intero | On Error Goto | gestione Basic dell'errore | Using | formatta lo schermo |
| Deg | converte in gradi | On Sq | gestione Basic del suono | Vpos | restituisce la posizione verticale del cursore |
| Delete | cancella linee | Openin | apre su cassetta un file in ingresso | Wait | aspetta che su una porta di I/O si verifichino specificate condizioni |
| Di | disabilita gli interrupt | Openout | apre su cassetta un file in uscita | Wend | segnala la fine del ciclo di While |
| Draw | disegna linee | Origin | ridefinisce l'origine delle coordinate | While | determina la condizione di ciclo |
| Drawr | dis. linee in coordinate relative | Out | manda un valore sulla porta d'uscita | Width | stabilisce il numero di colonne su cui scrive la stampante |
| Edit | edita le linee | Paper | seleziona il colore di sfondo del carattere | Window | gestisce le finestre testo-grafiche |
| Ei | abilita interrupt | Pen | sceglie i colori | Window Swap | scambia i numeri d'ordine delle finestre |
| Ei | abilita interrupt | Plot | disegna un punto | Write | scrive in un certo formato |
| Ei | abilita interrupt | Plotr | come Plot ma in coordinate relative | Xor | Or esclusivo logico |
| Ei | abilita interrupt | Pos | ritorna la posizione del cursore (su schermo o stampante) | Xpos | restituisce la posizione orizzontale del cursore |
| Ei | abilita interrupt | Rad | abilita la misurazione di angoli in radianti | Ypos | restituisce la posizione verticale del cursore |
| Ei | abilita interrupt | Randomize | genera numeri casuali | Zone | gestisce la formattazione del video |
| Ei | abilita interrupt | Release | rilascia i canali del suono | | |
| Ei | abilita interrupt | Remain | disabilita i timer | | |
| Ei | abilita interrupt | Renum | rinumeri le linee di programma | | |
| Ei | abilita interrupt | Restore | ripristina i Data | | |
| Ei | abilita interrupt | Resum | assegna la prossima linea da eseguire dopo un On Error Goto | | |
| Ei | abilita interrupt | Rnd | gestisce i numeri casuali generati | | |

a 64K di memoria, tra Ram e Rom. Orbene, il CPC 464 usa contemporaneamente 64K di Ram e 32K di Rom, per un totale di 96K: come fa? È presto detto: la Rom è mascherata all'utente, che non può accedervi, ma non al sistema operativo che la controlla. Possiamo però sfruttare diverse delle routine del S.O. anche da Basic, poiché gli indirizzi d'ingresso sono messi in Ram, come indicato sul manuale (vedi anche la mappa di memoria pubblicata nella pagina a fronte).

Per quel che concerne il linguaggio, siamo davanti ad un eccellente insieme di istruzioni: circa 160 le parole riservate, un vero record per una versione 1.0! A parte la grafica e il suono, che vedremo in maggior dettaglio, abbiamo la programmazione strutturata, la presenza di moltissimi operatori, l'uso esteso delle parole ON (per verificare situazioni non solo algoritmiche, ma anche logiche) e DEF (per vari tipi di numero, di grandezze, etc), un ottimo tool di debugging per ricerca e correzione degli

errori di sintassi, un esteso controllo degli errori di programma (ON ERROR, ON BREAK, ERR, ERL e la definizione dei messaggi d'errore da parte dell'utente), una valida gestione dell'uscita da Print (sia su schermo che su stampante!), ma la ciliegina sulla torta è senz'altro rappresentata dalla manipolazione degli interrupt direttamente da Basic.

Per ovvi motivi, noi non vedremo in dettaglio tutte le caratteristiche del linguaggio implementato sul CPC 464, altrimenti andremmo a scrivere un libro: potrete comunque intuirne gran parte degli usi, sia dalla tabella esplicativa delle parole del Basic — che per semplicità non comprende quelle più comuni (tutti sanno cosa vogliono dire Print, Rem, List...) — che dalla buona quantità di listatini commentati, che riteniamo sufficientemente significativi.

L'aritmetica può contare su un gran numero di istruzioni: tra gli operatori aritmetici troviamo infatti anche il MODULO, che

restituisce il resto di una divisione, per cui PRINT 25 MOD 3 scriverà sullo schermo il valore 1.

Tra i vari usi del Def ce ne sono alcuni tipici del Fortran: Def Int definisce un numero intero, mentre Def Real gestisce le quantità come reali.

Proseguendo il raffronto con gli altri linguaggi, la programmazione strutturata viene senz'altro dal Pascal, e si articola in due modi: nell'ormai standard

ON <condizione a più possibilità> GOTO (oppure GOSUB) oppure nel più recente

WHILE <condizione> ... WEND ove Wend sta per While END, che spesso semplifica enormemente la verifica di condizioni di ciclo.

Passando alle funzioni trigonometriche, le solite seno, coseno, tangente, ne possiamo impostare l'argomento sia in radianti che in gradi, tramite la parola riservata Deg. È possibile richiamare il valore del pi greco, usando l'espressione PI che offre

otto decimali. A proposito di cifre decimali, la precisione è proprio di 8 cifre. Va rilevato che il calcolo degli esponenziali pare avere la stessa accuratezza di quello dei decimali, per cui digitando

```
PRINT 2 ^ 6-2 * 2 * 2 * 2 * 2
```

si ottiene il valore corretto, cioè 0, cosa questa non sempre valida negli home computer che calcolano le potenze con un algoritmo diverso da quello della moltiplicazione, presentando quindi un errore diverso.

Per quel che concerne le funzioni matematiche c'è anche da rilevare la LOG10, che restituisce il logaritmo in base 10 (anziché in base e) dell'argomento, il che in calcoli matematici è piuttosto utile.



Schermata del programma Typing Tutor, per familiarizzare con la tastiera, inserito nella cassetta in dotazione. Il nostro record è nessun errore con una media di 0.98 secondi per tasto ...

Tra le funzioni strane segnaliamo almeno le due complementari MIN e MAX, che accettano una sequenza numerica e ne restituiscono l'elemento rispettivamente minore e maggiore: provate ad esempio un miniprogramma del tipo

```
10 FOR t = 0 to 99
20 c = MIN (30,t)
30 NEXT t
```

e vedrete che fino a 30 il minimo è t, poi rimane sempre 30.

Anche l'editor e il debugger, che consentono rispettivamente di scrivere e di correggere il listato, sono validi. Il primo agisce su tutto lo schermo: avendo individuato un errore basta tornare sulla sua linea, adoperando i tasti cursore, ribatterla automaticamente tenendo pigiato il tasto Copy ed infine effettuare la correzione. La presenza del RENUMber e del DELETE semplificano di molto la vita dei programmatori, senza contare la TRace ON (che però sfrutta l'intero schermo, e va quindi accuratamente gestita).

Vediamo ora come si definiscono i caratteri utente. All'accensione abbiamo a disposizione i 16 codici dal CHR\$(240) al CHR\$(255), ma possiamo ridefinire questo limite con una

```
SYMBOL AFTER <numero>
```

che rende disponibili tutti i codici da quello specificato in poi. Per definire il carattere bisogna avere a disposizione la stringa di 8

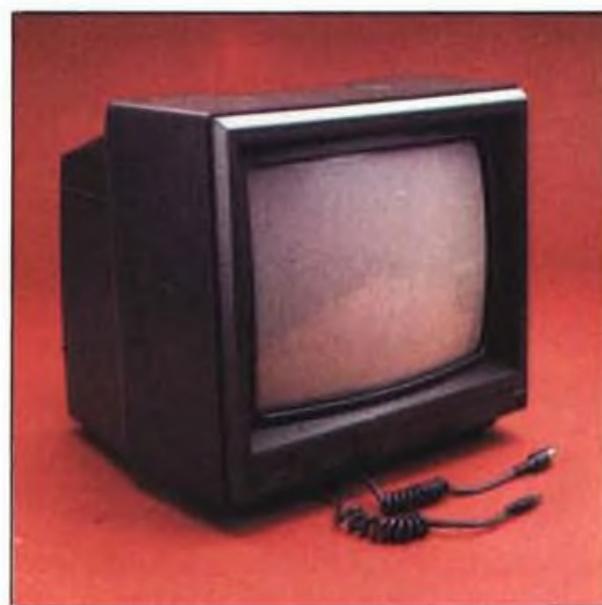
numeri decimali che lo rappresenta, ed usare una

```
SYMBOL <n' codice> , <1' numero> , ..., <8' numero>
```

È evidente che per ripristinare la situazione standard basta digitare un

```
SYMBOL AFTER 240
```

L'ultimo argomento che tratteremo a proposito del Basic è la gestione degli interrupt. La possibilità più appariscente è senz'altro l'inserimento di un vostro listatino Basic nella routine generale. Per capirci richiamiamo brevemente il funzionamento del computer. Il microprocessore gestisce almeno due tipi di situazioni, quelle statiche (es.: il programma) e quelle dinamiche (es.: le memorie, lo schermo, la lettura della tastiera, etc.); le prime rimangono im-



mutate per (almeno) alcuni secondi, laddove per lo stesso tempo le seconde variano migliaia di volte (es.: ogni secondo lo schermo TV varia 30 volte, e ogni schermo è composto di alcune decine di migliaia di punti). Per gestire le situazioni dinamiche si richiede un breve ma frequente intervento del microprocessore, che quindi generalmente serve le situazioni statiche, ma ogni tanto, a cadenza fissa, va a servire le altre.

Una tipica frequenza con cui vengono servite queste richieste è di 50 volte al secondo: ciò vuol dire che ogni cinquantesimo di secondo il processore interrompe il suo lavoro principale. In inglese interruzione si dice interrupt.

Inserire una propria routine Basic dentro la routine di interrupt del sistema operativo vuol dire che 50 volte al secondo oltre alle normali funzioni il computer farà quello che voi gli chiedete, tipicamente suonare una delle note specificate in una serie di Data, o aggiornare delle coordinate, o leggere il joystick, e così via: è evidente che questo programma dovrà essere breve, altrimenti il processore perderà troppo tempo ad eseguire la vostra routine e gliene rimarrà troppo poco per il programma principale, che diventerà lentissimo.

L'istruzione che manipola questa opzione si chiama Every (=ogni), e ha la seguente sintassi:

```
EVERY <secondi * 50> GOSUB <n' linea>
```

ovvero ogni tot secondi unisci alla routine di interrupt anche le istruzioni comprese tra <n' linea> e il successivo Return. Per un esempio vedete l'opportuno listatino pubblicato nell'articolo.

Se volete che la vostra routine Basic venga eseguita una sola volta, ma ad un istante specificato, usate la parola After (=dopo), che ha la stessa sintassi di Every:

```
AFTER <secondi * 50> GOSUB <n' linea>
```

Le periferiche

La Amstrad ha in catalogo due periferiche dedicate, una stampante ed un'interfaccia per un disco da 3" (cui eventualmente se ne può attaccare un secondo). Della stampante, che si chiama DMP 1, c'è ben poco da dire: si tratta di una 80 colonne a punti, con ogni carattere inscritto in una matrice 5 x 7, che in modo grafico fornisce 480 pixel per linea. Nel modo testo da 80 colonne la velocità di stampa è di 50 caratteri per secondo.

Più interessante è il disco. Lo standard adottato, è quello da 3" Hitachi-Panasonic, nella configurazione singola faccia - doppia densità. Per usare l'unità bisogna provvedersi dell'interfaccia DDI-1, che mette a disposizione dell'utente non solo l'ovvia estensione del Basic per la gestione della periferica aggiunta, ma anche due sistemi operativi: AMSDOS, quello della casa, e CP/M, della Digital Research, e con tre diverse possibili formattazioni, di tipo System (programmi e dati, 169K utente), Data (solo dati, 178K) ed IBM compatibili in CP/M (154K).

L'interfaccia contiene dell'altro software, consistente in Mr. Logo, un linguaggio particolarmente adatto per insegnare la programmazione ai bambini (ma non solo a loro!) che sfrutta tutte le caratteristiche del suo standard, più quelle hardware di suono e grafica dell'Amstrad.

Va infine citata la possibilità di connettere un secondo drive senza dover acquistare altre interfacce.

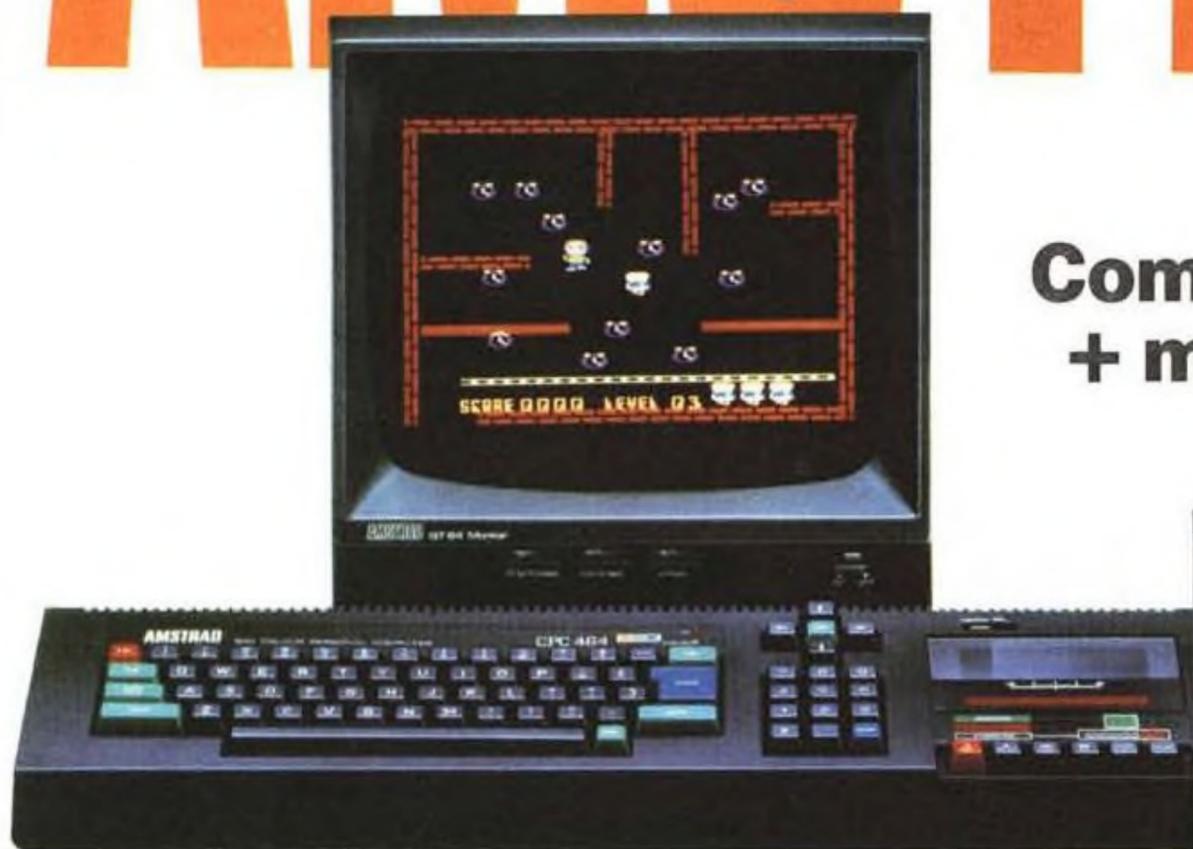
Conclusioni

L'Amstrad CPC 464 si presenta con un'estetica curata, un'architettura integrata e completa, un hardware di classe, un firmware su ROM che ridefinisce gli attuali standard del Basic ed una disponibilità di software già estesa alle normali applicazioni (wordprocessor, spreadsheet, assembler) ma con enormi possibilità di sviluppo. Tutto ciò viene offerto a prezzi veramente accessibili. Le scelte hardware (Z 80; 8255; 8912) ne fanno un computer appetibile dagli utenti smaliziati. È inoltre il primo di questa categoria che offra reali caratteristiche di piccolo personal che lo rendono adatto a professionisti e commercianti, oltre che ai loro figli.

Il nostro parere è che, dal punto di vista delle prestazioni offerte, si tratti forse del migliore acquisto attualmente possibile nella sua categoria di prezzo. **MC**

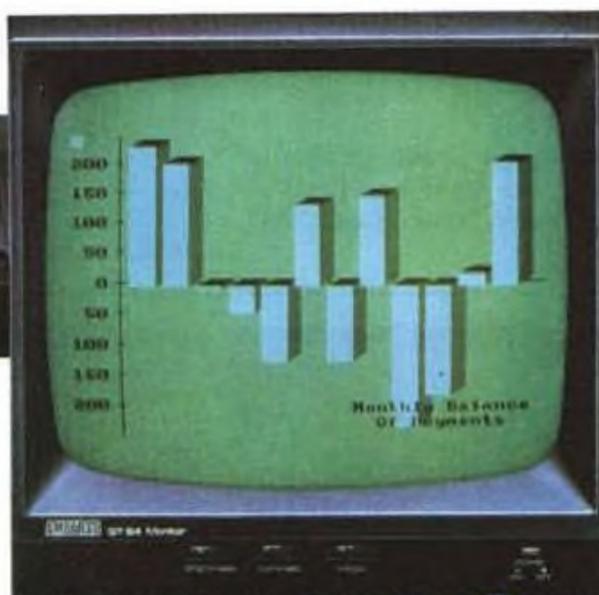
AMSTRAD CPC 464

Arnold è unico



**Computer + registratore
+ monitor fosfori verdi
L. 698.000***

**Computer + registratore
+ monitor a colori
L. 998.000***



**Questo è il nuovo, straordinario home computer AMSTRAD CPC 464,
Arnold per gli amici.**

* IVA Esclusa

Grande memoria

64K RAM + 32K ROM

Basic

il più veloce e completo, senza confronto a questi livelli di prezzo. Un assaggio? AFTER e EVERY per la gestione degli interrupt, WINDOW per aprire fino ad 8 finestre su schermo

Suono

3 voci estese su 7 ottave e, soprattutto, comandi come SOUND, SQ, ENV, ENT, invece di ostiche PEEK e POKE

Schermo

20, 40, 80 colonne a scelta

Colore

tavolozza di 27 colori utilizzabili

Alta risoluzione

640 x 200 punti singolarmente indirizzabili

Software

basta un catalogo che si arricchisce al ritmo di due titoli per settimana e che già comprende decine di giochi, word processor, spreadsheet, PASCAL, linguaggio C. ecc...?

Esclusività per l'Italia:

MICROSTAR srl

Via Cagliero, 17 - 20125 Milano
Tel. 02/688.76.04