

A TUTTO

commodore

I.P.

Direttore responsabile Ruben Castelfranchi
Aut. alla pubblic. Trib. di Monza n. 258 del 28.11.74
Stampa: Elcograf - Beverate di Brivio (Co)

APRILE 1985

4

| | |
|--|----|
| POSTA | 1 |
| NOTIZIE | 4 |
| SPECIALE | 8 |
| “Commodore C-16” | |
| UTILITY | 12 |
| HARDWARE | 16 |
| “Niente paura davanti alla porta d’utente” | |
| LISTATI | 20 |
| “Disegnare col joystick” | |
| SOFTWARE | 26 |
| “Caverns of Khafka” | |
| “H.E.R.O.” | |
| “Space Rescue” | |
| “The spider and fly” | |
| “Acos +” | |
| “Terrorist” | |
| “Beamrider” | |
| “Money Manager” | |



POSTA

LA FUNZIONE TAB

Spettabile Redazione di Sperimentare, sono un appassionato di computer e possiedo da circa 5 mesi un Commodore 64. Vi scrivo per porvi una domanda riguardo un fatto che mi ha incuriosito non poco. Desiderando stampare sullo schermo le lettere dell'alfabeto ai due estremi, cioè alla colonna 0 e alla colonna 39, ho battuto le seguenti linee di programma:

```
10 PRINT " A "; TAB (39) "A"  
CRSR ↑ 20 PRINT " B"; TAB (39) "B"  
210 PRINT " Z"; TAB (39) "Z"
```

Dato il RUN, è cominciato il fatto strano: ho visto che solo la prima linea di programma viene eseguita correttamente, cioè viene stampato il carattere A nella colonna 0 e nella colonna 39.

Tutte le righe successive vengono invece stampate con due caratteri nelle colonne 0 e 1: viene cioè ignorata la funzione TAB (39). Ho notato che sopprimendo lo shift del cursore verso l'alto (CRSR ↑), ciò non si verifica più, ma d'altronde questo è indispensabile per non creare una riga stampata e una no. Infatti, "printando" qualsiasi carattere alla colonna 39, dopo la stampa il cursore si posiziona automaticamente alla colonna 0 della riga successiva e quindi, alla successiva PRINT, passando ancora alla riga successiva, ne lascerebbe una vuota. Ho anche notato che si può evitare l'anomalia facendo "printare" tutto alla colonna 38, cioè alla penultima anziché all'ultima.

Io vorrei quindi sapere come mai si verifica tutto ciò: come mai cioè, ordinando la stampa di un carattere all'ultima colonna (39) dello schermo, questo viene invece stampato di seguito a quello o a quelli

esistenti sulla stessa riga (ma solo se nella PRINT relativa c'è il simbolo CRSR ↑ e se nella linea precedente di istruzioni c'è un'altra TAB (39).

Per maggiore chiarimento vi prego di battere le righe 10 e 20 dell'esempio che riporto, e vi renderete conto di come la linea 10 provochi la stampa della lettera A alla colonna 0 e 39, mentre la linea 20 e le successive danno come risultato la stampa delle lettere alle colonne 0 e 1 (anziché 0 e 39).

So bene che esistono altri modi per ottenere ciò che voglio, ma vi sarei davvero grato se mi rispondeste sulla rivista; spiandomi come mai ciò avviene.

Vi ringrazio fin d'ora, salutandovi cordialmente e facendovi i più vivi complimenti per l'ottima rivista e soprattutto per l'inserito "A Tutto Commodore".

Giovanni Midali - Forlì

Caro Giovanni, come tu hai specificato, esistono molti modi per ottenere quello che desideri, e noi te ne indichiamo uno dei tanti. Il motivo vero e proprio per cui non riesci a fare quello che desideri con la funzione TAB non si conosce e può avere molte spiegazioni: probabilmente il "CRSR ↑" prima del TAB non fa riconoscere al sistema la linea di stampa corrente per cui non viene svolta la funzione. Comunque questo è uno dei piccoli difetti del sistema operativo, forse per questo è stata creata una funzione SPC. Se proverai il piccolo esempio che pubblichiamo, vedrai che funziona perfettamente quando la zona di stampa sul video è pulita, ma se riscrivi la stessa cosa in una zona già usata noterai degli strani effetti, anche questo è un piccolo difetto di sistema. Speriamo di aver soddisfatto la tua curiosità e ti invitiamo a riscriverci per qualsiasi altro problema o difetto che riscontrerai durante l'uso del calcolatore.

```
10 PRINT "A"; SPC(38) "A"
20 PRINT "B"; SPC(38) "B"
30 PRINT "C"; SPC(38) "C"
40 PRINT "D"; SPC(38) "D"
```

QUALE COMMODORE COMPRARE

Spettabile Redazione di A Tutto Commodore, sono da due mesi un vostro soddisfatto lettore, mi complimento quindi per l'ottima riuscita della vostra rivista, ma veniamo ai fatti; ho deciso di acquistare un computer, ma la scelta tra i due big, ovvero Commodore e Sinclair è abbastanza difficile ed oltre ad essi ci sono altri personal che io ritengo validi, per chiarirmi un po' le idee insomma vi chiedo per favore di rispondere a queste mie domande.

1) Vorrei avere informazioni sul Commodore PLUS 4, di cui ho sentito parlare, vorrei sapere inoltre se i programmi del C-64 sono compatibili con quelli del PLUS 4.

2) Per quanto tempo verrà ancora prodotto il C 64? Ho letto che è in vendita per il C-64 un'interfaccia denominata IEEE 488, per quale periferica occorre tale accessorio?

3) Vorrei avere informazioni sui Commodore serie 600, 700, 8000 e 8296.

4) So che è difficile fare un confronto del genere perché ciò creerebbe le immancabili parzialità ma vorrei sapere nei limiti del possibile i pregi e i difetti dei due più famosi personal ovvero Commodore 64 e Sinclair Spectrum.

Aspetto con fiducia le vostre risposte ringraziando anticipatamente e scusandomi per i numerosi errori.

Monnicchi Damiano - Pistoia

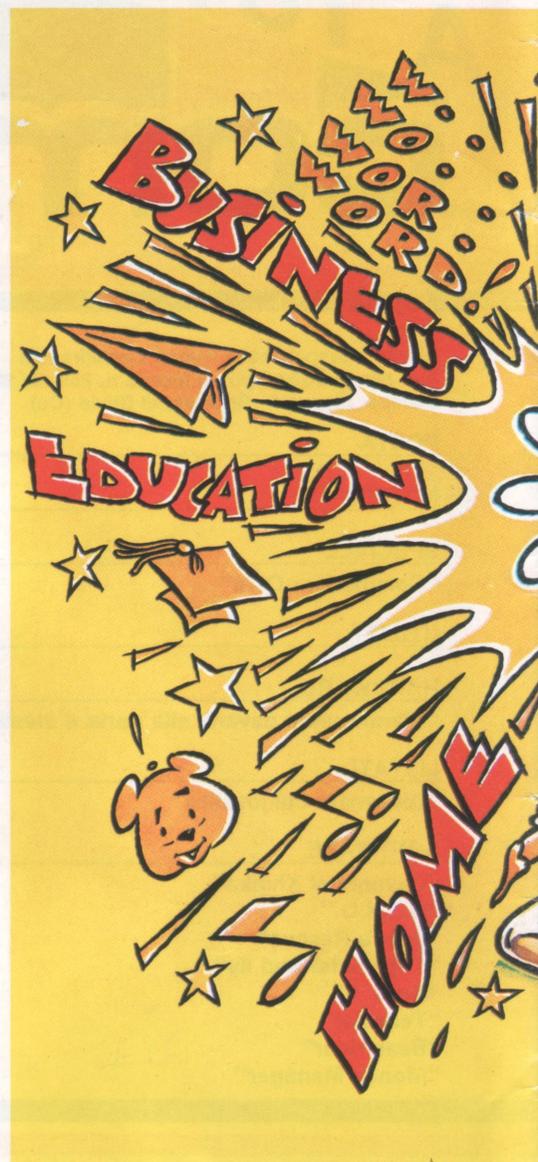
Caro Damiano, grazie per i complimenti alla rivista e speriamo di meritargli anche per le risposte alle tue interessanti domande.

Per prima cosa avrai sicuramente visto l'articolo sul numero scorso in cui parliamo ampiamente del PLUS 4 che speriamo abbia esaudito la prima parte della tua richiesta, per la seconda parte i programmi per il C-64 non girano sul PLUS 4 ma i programmi per questo nuovo nato Commodore non mancheranno e ne sono già in vendita numerosi. Venendo al secondo gruppo di domande sul C-64 siamo sicuri che la Commodore non ha ancora pensato di cessarne la produzione visto l'enorme successo e la grande richiesta che continua ad avere questo computer. L'interfaccia IEEE 488 è una interfaccia particolare Commodore per collegare al C-64 periferiche dedicate di tipo professionale come stampanti o disk drive da 1 Megabyte ecc.. Per quanto riguarda la terza domanda, vale a dire la serie professionale Commodore, ti consigliamo di rivolgerti direttamente alla Commodore Italiana in Via F.lli Gracchi 48 a Cinisello Balsamo (MI). La risposta all'ultima domanda pensiamo possa essere tranquillamente contenuta in un libro ma cercheremo di risponderti in maniera concisa. Innanzi tutto l'operatività: lo Spectrum produce parole chiave con la pressione di un solo tasto, risparmiando così tempo. Nel Commodore c'è una netta divisione fra pagina grafica e testo, cosa che non succede nello Spectrum.

Altra pecca del 64 è che le linee di programma non vengono controllate al momento dell'inserimento, ma danno errore solo quando il programma viene fatto girare, nello Spectrum l'errore viene segnalato subito. La produzione dei suoni è nettamente superiore nel 64 dato che contiene un Chip capace di produrre contemporaneamente 3 voci, nulla a che fare con il "povero beep" dello Spectrum. Altro difetto sul Commodore è al momento del caricamento in quanto non si può capire se il computer sta caricando superato però ora dal TurboTape. Ci sono poi numerose altre differenze nel Basic che però non abbiamo spazio per esaminare. Un saluto infine da parte di tutta la redazione.

HELP!!!

Spettabile Redazione, sono un vostro fedele lettore dal numero di Agosto del 1982. Vi lancio ora il mio HELP! Posseggo un ZX Spectrum che sto per cambiare con un Commodore 64. Ecco il problema: sono insospettito dall'improvviso calo del prezzo del C 64, non vorrei che ciò fosse dovuto all'imminente ritiro di tale macchina dal mercato. Attendo vostre delucidazioni, vi ringrazio e vi prego di rispondere



in fretta, in quanto dalla vostra risposta dipenderà la mia scelta. È così impossibile l'uso della grafica sul C-64? Grazie ed ancora complimenti.

Stefano Gambaro - Roma

Caro Stefano, ti ringraziamo prima di tutto per la fedeltà con cui segui la nostra rivista. L'improvviso calo di prezzo del C-64 ha destato in effetti molti sospetti ma è dipeso esclusivamente da politiche commerciali e di mercato in quanto il successo del Commodore 64 è indubbio e la sua produzione è ben lungi dall'esaurirsi data la continua richiesta.

Effettivamente l'uso della grafica sul Commodore 64 è abbastanza complesso dato che non esistono istruzioni per l'uso diretto di comandi grafici.

Nonostante queste limitazioni operative, le capacità grafiche del 64 sono eccellenti.



ti. In ogni caso ti consigliamo l'acquisto di software specifico o di tavolette grafiche. Per un approfondimento maggiore dell'argomento ti consigliamo di acquistare dei libri specifici, facilmente reperibili sul mercato; ad ogni modo sulla nostra rivista, nei numeri di Giugno e Luglio/Agosto, sono apparsi articoli interessanti sulla grafica e anche nei prossimi numeri non mancheranno articoli sull'argomento. Un saluto infine dalla redazione.

MENO SOFTWARE

Spettabile Sperimentare A Tutto Commodore.

Sono un vostro abbonato, e ho letto l'inserito A tutto Commodore.

Personalmente, preferirei che queste pagine rimangano nella rivista, a fianco del "Sinclair Club".

Unico appunto che vi faccio, è l'eccessivo

spazio utilizzato per la spiegazione del soft commerciale o giochi.

Potreste ridurre la spiegazione al necessario ed eliminare gran parte delle foto, utilizzando lo spazio risparmiato per la pubblicazione dei "list".

Ovviamente, quanto sopra scritto, è solo la mia opinione che nulla toglie alla "bontà" della rivista che apprezzo molto, altrimenti non avrei rinnovato l'abbonamento. Nella speranza che la mia lettera vi serva a qualcosa, vi saluto e vi porgo i migliori auguri.

Walter

Caro Walter, grazie della fiducia che ci hai concesso rinnovando l'abbonamento e dei tuoi consigli. In poche parole cercheremo di convincerti dell'utilità di separare l'inserito Commodore dal resto della rivista; per prima cosa questa operazione ha

permesso agli utenti Commodore di avere più spazio, un maggior numero di pagine, e la possibilità di ingrandirsi notevolmente senza rubare spazio al Sinclub e alle altre rubriche di Sperimentare. Non ultima la possibilità non remota di creare proprio una rivista autonoma per tutti i Commodoriani. Per quanto riguarda il Software, la rubrica è molto seguita e sono numerosissime le richieste di ampliarla e di arricchirla con più illustrazioni visto il numero incredibile di software presente sul mercato, quindi come vedi, ci troviamo in una posizione delicata ma pensiamo di aggirare l'ostacolo ampliando sicuramente in un prossimo futuro la parte riservata ai listati e ai consigli di programmazione. Soddisfatto Walter. Noi speriamo di sì e ti invitiamo a scriverci di nuovo. Ciao.

LISTATI

Spettabile rubrica "A Tutto Commodore", in primo luogo un complimento a tutti voi per la scelta fatta nei confronti della rubrica in oggetto, scelta sicuramente accettata con molto entusiasmo da tutti i "Commodoriani". Il mio augurio è che "A Tutto Commodore" cresca continuamente, sviluppandosi e trattando tutti gli argomenti, e sono tanti, che il computer Commodore coinvolge. Se mi è consentito vorrei aggiungere una osservazione e una domanda o forse una proposta. Perché sui listati dei programmi invece di stampare i caratteri grafici non usate altre notazioni, per altro molto diffuse, affinché la comprensione del testo risulti semplificata. Faccio un esempio i vari PRINT " " in PRINT { CLR/HOME } ecc..

Personalmente incontro molte difficoltà, a volte insuperabili, ad interpretare alcuni simboli grafici. Ringrazio molto e saluto cordialmente.

Morandotti Ferruccio - Pavia

Caro Ferruccio, ti ringraziamo innanzitutto per i complimenti che fai alla nostra rivista; il suggerimento che ci dai non possiamo metterlo in pratica, in quanto ci porterebbe via del tempo prezioso. Vedi Ferruccio, i listati che vengono pubblicati, debbono essere prima provati e con l'inserimento delle notazioni che ci suggerisci, questo non sarebbe possibile.

Logicamente, tu dirai: "perché non le inserite dopo aver provato il programma?". Perché in questo modo non saremmo mai sicuri della correttezza del listato. Pertanto continuiamo con il metodo usato sinora, cercando di ottenere stampe di listati più chiare possibili, magari potremmo aggiungere nel commento al programma, notazioni che spieghino le linee in cui compaiono un gran numero di simboli grafici.

TRASMISSIONE DATI SUL COMMODORE

La telematica è entrata a far parte anche del mondo degli hobbysti, possessori di macchina come lo Spectrum, il QL, o come il diffusissimo C-64. Per questa macchina sono state prodotte numerose periferiche ed interfacce.

La PRISM, che è la produttrice del VTX 5000, un modem utilizzato con lo Spectrum, ha realizzato il Modem 1000.

Senza dilungarci molto su questa interfaccia, diciamo che essa svolge le identiche funzioni della già citata VTX 5000, permette cioè agli utenti del C-64 di comunicare fra di loro, e di utilizzare il Micronet 800, nonché con l'aggiunta di software adatto di utilizzare i sistemi di posta elettronica.

Prodotto dalla Commodore Communications Modem, un altro apparecchio, pubblicizzato in una forma alquanto originale: "Per il prezzo di dieci alieni potete comunicare con gli amici più lontani", permette di utilizzare i servizi del Micronet e del Prestel. In aggiunta a questi servizi, se ne viene a creare però uno nuovo: il Compunet.

L'utente può comunicare con altri utenti, con la sicurezza che la comunicazione resta riservata.

È possibile creare così dei club, scambiarsi informazioni, notizie e fare degli annunci di qualsiasi tipo grazie a questo servizio. A differenza quindi dei sistemi fino ad ora analizzati, nei quali l'utente in genere può usufruire di una banca dati e utilizzare servizi



tipo posta elettronica, con il Compunet si viene a creare un modo nuovo e diverso di comunicazione fra gli utenti dello stesso servizio.

In altre parole, alcune delle informazioni trasmesse dal sistema vengono fornite dagli stessi utenti.

Ci sembra che questo modo di fare, sia effettivamente il migliore, e che la via indicata debba essere seguita anche dagli altri produttori e per le altre macchine.

Esistono oltre quelli citati, altri ser-

vizi, quali il CompuServe, che permette agli utenti di C-64, di comunicare fra di loro, tramite modem, con lo scopo principale di scambiarsi informazioni, relativamente ai problemi concernenti la programmazione, o indicazioni su giochi e via dicendo.

Il colloquio con gli altri utenti del sistema, può effettuarsi sia tramite una conversazione pubblica, alla quale cioè tutti possono accedere, che con conversazioni private, fra due utenti.

A SCUOLA DI VIDEOGAME

Ti sei stancato dei soliti videogiochi da bar? Ti piacerebbe saper programmare un videogioco di tua invenzione? Vuoi imparare il Basic senza annoiarti, esplorando le capacità del computer in maniera divertente? A Milano è nata la Video Design School, una scuola in grado di insegnarti teoria e pratica della progettazione dei giochi elettronici ed introdurli nel fantastico mondo della computer

graphic.

Alla Video Design School troverai molti corsi di diverso livello, tutti entusiasmanti, che ti avvicineranno al mondo del computer in modo stimolante e se sei già esperto, ti potrai tuffare nelle incredibili possibilità creative della grafica computerizzata: generazione di immagine, animazione, percezione visiva e teoria del colore ti attendono, nell'elaborazione di progetti concreti. Sarai tu l'inventore del nuovo PACMAN?

La scuola mette a disposizione degli allievi una sala computer attrezza-

ta con i favolosi Commodore 64 ed un team di insegnanti professionisti, i quali, meglio di qualsiasi programmatore, potranno aiutarti a superare le difficoltà.

Già nel corso base sarai in grado di sviluppare un videogioco di tua invenzione, imparando il linguaggio BASIC ed animando così gli sprite del tuo computer ... alla fine del corso sarai in grado di affrontare progetti non solo "videogiocosi"!

Tutte le informazioni si possono richiedere alla Softgraf, telefono 02/6899912.

102 PROGRAMMI PER COMMODORE 64

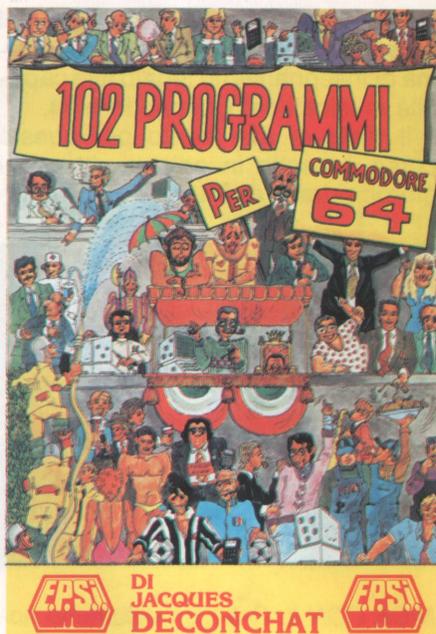
Autore:
Jacques Deconchat

Casa editrice: E.P.S.I.
Prezzo L. 20.000

Attraverso la programmazione di oltre 100 giochi potrete arrivare ad imparare il linguaggio Basic divertendovi.

Questo è in breve lo scopo che si prefigge "102 programmi per Commodore 64" egregiamente scritto da Jacques Deconchat.

I giochi presentati sono stati intelligentemente inseriti in ordine di difficoltà crescente in modo tale da permettere a chiunque di capire quello che si sta trattando. Si parte dal primo livello dove vengono affrontate le istruzioni più elementari del Basic (PRINT, INPUT, FOR ... NEXT, IF ... THEN, GOTO, GOSUB, ecc.). Si passa poi alle istruzioni necessarie alla con-



fezione ed alla manipolazione degli elementi di un array e della ricerca di un elemento di una stringa (DIM, R (I), MID\$, DATA, READ). Al quarto livello si analizza l'istruzione GET (basilare

per la costruzione di giochi interattivi). Ed infine si arriva alle istruzioni che stanno al limite del Basic ed alla soglia del linguaggio macchina: PEEK e POKE, con le quali si può avere una manipolazione diretta nelle locazioni di memoria del computer.

Il libro non si limita a riportare crudamente una serie di listati ma aggiunge molti commenti grazie ai quali la programmazione risulterà piacevole e veloce. Innanzi tutto si fanno alcuni riferimenti sulla difficoltà di programmazione, sulla categoria e sulla difficoltà del gioco.

Dopo la descrizione del gioco e delle sue regole, si passa all'analisi completa del programma.

Il tutto si conclude dando all'utente alcuni suggerimenti sulle possibili estensioni del programma in base alle quali ci si potrà sbizzarrire nella programmazione.

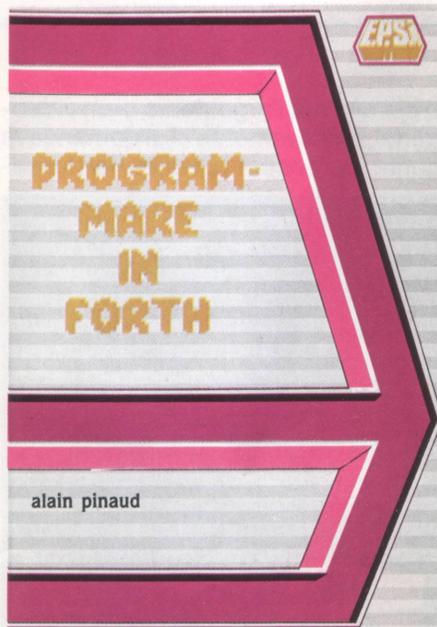
Oltre 230 pagine con più di 102 programmi riportati integralmente sono il biglietto da visita di questo interessante "102 programmi per Commodore 64".

PROGRAMMARE IN FORTH

Autore:
Alain Pinaud

Casa editrice: E.P.S.I.
Prezzo: L. 14.000

Per chi vuole allargare le proprie conoscenze nel campo dei linguaggi informatici ecco disponibile un nuovo manuale: "PROGRAMMARE IN FORTH" di Alain Pinaud edito da E.P.S.I.. Così come l'autore stesso dichiara il Forth risulta essere un linguaggio totalmente diverso da quelli abitualmente utilizzati, un'alternativa, ma non per questo la programmazione risulta meno interessante e soddisfacente. Specialmente per questa diversa concezione il Forth sembrerà forse leggermente difficoltoso. È consigliabile, quindi, avvicinarsi alla lettura di questo manuale già con una discreta conoscenza dei diversi linguaggi informatici e, possibilmente, con una infarinatura generale su ciò



che riguarda il linguaggio macchina. Tutto questo servirà per poter affrontare la lettura del volume con maggior facilità e comprensione e, quindi, per

ottenere in seguito migliori risultati pratici.

Questo manuale è concepito in modo tale da rendere chiara e concisa la spiegazione del linguaggio Forth.

È praticamente strutturato in due parti distinte: i primi capitoli espongono essenzialmente la filosofia Forth e le nozioni basilari, mentre nei capitoli successivi viene esplicitata tutta la parte riguardante la programmazione vera e propria.

In ogni capitolo di questo volume si trovano chiare presentazioni e spiegazioni, esercitazioni pratiche con relative soluzioni, applicazioni dei concetti spiegati ed uno specchio ri-epilogativo nel quale vengono riportati tutti i nuovi termini che sono stati spiegati nel capitolo e le loro funzioni.

Il testo è anche fornito di un glossario contenente tutta la terminologia e di due Programmi.

"Programmare in Forth" pur essendo solo un'opera di iniziazione darà la possibilità con adeguata applicazione, di programmare in modo nuovo e alternativo.

MUSICA CON L'HOME COMPUTER

Si può fare musica con un home computer? Ancora un paio d'anni fa la domanda suonava provocatoria, come se si volesse sostenere che i "bip" e i motivetti facili facili dei videogiochi potessero essere scambiati per "vera musica".

È difficile negare che l'avvenimento dell'anno 1984, sotto il profilo del connubio musica-elettronica, sia stato la diffusione commerciale dell'interfaccia MIDI, e della possibilità di controllare sintetizzatori ed altri strumenti attraverso computer come Commodore 64, Apple II, Spectrum, BBC.

Così il personal computer si trova al centro forse non della "vera" musica elettronica, ma della musica diffusa nel mondo contemporaneo.

È possibile imparare a servirsi di questi nuovi strumenti musicali? Le esperienze didattiche già esistono. Franco Fabbri, che ha tenuto il primo corso su questo argomento al CEPAM di Reggio Emilia nel 1983/84, e che scrive mensilmente di queste co-

se su "Fare Musica", svolgerà nei prossimi mesi a Milano un programma di insegnamento legato alle capacità musicali del Commodore 64.

Il testo di riferimento per questo corso è uscito in edicola da pochi giorni: si tratta di *MUSICA ELETTRONICA CON IL C-64*, un volumetto accompagnato da una cassetta contenente una serie di programmi dimostrativi; l'autore è lo stesso Franco Fabbri.



Il corso è indirizzato a tutti coloro che vogliono iniziare a familiarizzarsi con i problemi del controllo numerico del suono, e con i concetti fondamentali dell'informatica musicale.

Non è richiesta la conoscenza preliminare di linguaggi di programmazione; peraltro, gli allievi che porteranno a termine il corso avranno raggiunto una buona conoscenza del Basic ed avranno una certa confidenza con il linguaggio macchina, oltre a conoscere le caratteristiche del SID, il generatore dei suoni incorporato nel COMMODORE 64.

CORSI

DATA INIZIO: 18/1/1985

ORARIO: dalle h. 18 alle h. 20,30 il venerdì

COSTI: L. 200.000 + iscrizione.

Le iscrizioni si raccolgono c/o LABORATORIO DI INFORMATICA MUSICALE tutti i pomeriggi escluso il sabato in Via Sottocorno, 46 - Milano - Tel. 02/7385622.

PROSSIMAMENTE QUI

La popolarità dell'ex Beatle non ha confini.

Il suo ultimo film "Broad Street" prodotto dall'americana 20th Century Fox, in Inghilterra ha riscosso un clamoroso successo.

Ma questo era prevedibile; infatti gli inglesi hanno un sentimento di venerazione nei confronti dei Beatles e di Paul Mc Cartney in particolare.

Le sue canzoni, anche le più vecchie, sono quotidianamente trasmesse alla radio e ciò si ripete così di frequente, che sembra quasi che il tempo si sia fermato agli anni sessanta quando i quattro di Liverpool erano in testa a tutte le classifiche come oggi è per i Duran Duran o i Bronski Beat.

Mc Cartney superstar, mito eterno. In Inghilterra, paese in cui tutto prima

o poi diventa l'argomento di un videogame, effettivamente ci si domandava come mai ancora i Beatles non fosse-

ro stati affidati al joystick. Ma adesso Mc Cartney in game è una realtà. Il gioco si intitola Broad Street, come il film.

È per Commodore 64, edito dalla Mind Games già produttrice di altri giochi nati dagli schermi cinematografici (Alien tra i games della Mind è ancora un best seller).

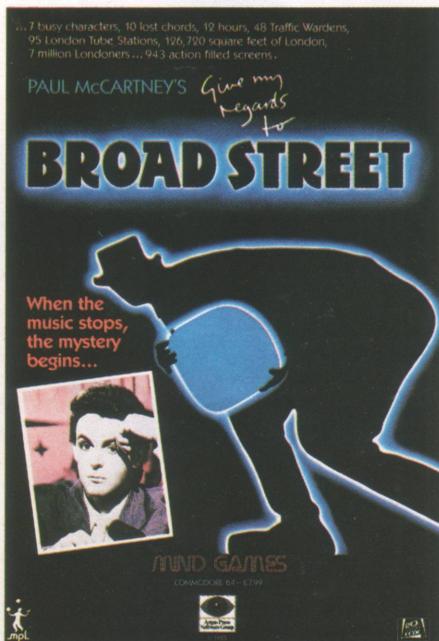
Il gioco è un'adventure con ben 943 schermi che riproducono schematicamente Londra.

I sette personaggi del gioco si muovono per la città attraverso 95 stazioni della metropolitana, 48 incroci stradali, su una superficie che propone in scala i 126.720 "square feet" (40.000 Km quadrati) di Londra.

Ovviamente la musica non manca: è quella della colonna sonora del film di cui in questi giorni anche in Italia è disponibile il disco.

Per il gioco da noi forse è prematuro accennare qualcosa.

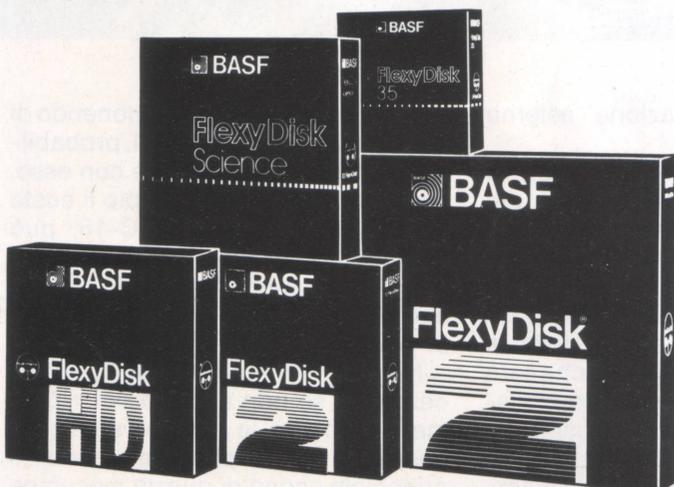
Comunque è già importante sapere che c'è.



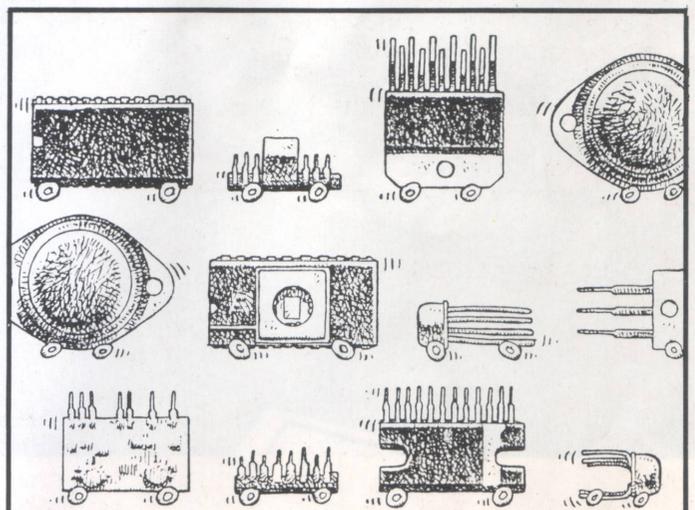
**A tutto Commodore...
 pubblica i vostri
 migliori programmi.
 Mandateceli!**

**Go to
 BASF FlexyDisk®**

*La nuova linea di
 dischetti BASF.
 Sicurezza dei dati
 grazie ad una
 tecnologia d'élite.*



DATA BASE S.p.A.
 V.le Legioni Romane, 5
 20147 MILANO
 Tel. (02) 40 303
 Telex 31 52 06



**....NOI POSSIAMO ARRIVARE
 DOVUNQUE....**

...CERTO! BRUCIA I TEMPI TELEFONANDO
 AL N. 081/657365.
 EVADEREMO I VOSTRI ORDINI IN 24 ORE.
 SPEDIZIONI TUTTI I GIORNI IN TUTTA ITALIA ED ISOLE,
 DI COMPONENTI ELETTRONICI E RICAMBI.
 FORNITRICE ENTI E SCUOLE
 RICHIEDETE CATALOGHI E DOCUMENTAZIONI GRATIS,
 SPECIFICANDO LA VS. ATTIVITÀ



VICO ACITILLO, 69-71/85
 80128 NAPOLI
 TEL. 081/657365

COMMODORE C-16



Se avete seguito con attenzione questa rivista, non vi sarà sfuggita la prova del PLUS 4. Nel caso vi foste dimenticati di che cosa si tratta, tornate a sfogliare le pagine del numero di marzo.

Il C-16, come il PLUS 4, sono due delle nuove macchine prodotte dalla Commodore per far fronte all'offensiva scatenata dai vari altri produttori di micro. Il C-16, è molto simile al PLUS 4, in quanto utilizza lo stesso linguaggio di quest'ultimo, e cioè il BASIC 3.5 della Commodore. Le differenze dal PLUS 4, riguardano soprattutto l'espandibilità del sistema, e la capacità di base della macchina,

nonché la configurazione esterna della stessa.

Il C-16 è infatti realizzato con lo stesso contenitore del VIC20 e del C64. Conserva di questi ultimi anche la tastiera, che è decisamente uno degli elementi meglio riusciti di queste macchine. Diversa risulta però la disposizione dei tasti, che a detta di molti è più logica. È evidente che sarà avvantaggiato da questa nuova disposizione, chi non ha mai usato un Commodore, mentre i vecchi vichiani e simili, dovranno farci un po' la mano.

Dubitiamo comunque che i vecchi commodoriani, lascino le loro macchine per passare al C-16, che è più

potente del VIC, ma pur disponendo di molti più comandi del C-64, probabilmente non può competere con esso. Ne è una riprova ad esempio il costo delle due macchine. Il C-16, può quindi considerarsi come il sostituto del VIC 20, mentre il C-64, non ha ancora trovato una valida macchina in grado di sostituirlo.

Non avendo quindi prestazioni eccezionali, se non a livello potenziale (nel senso che le avrebbe se disponesse di un po' più di memoria), il principale scopo di questa macchina è quello didattico, sia nel senso di avvicinare nuovi utenti al computer, che in quello dell'uso della macchina co-

me strumento educativo, ad esempio nelle scuole.

Fra i programmi di cui la macchina è fornita, troviamo infatti, sia un'introduzione al BASIC, adatta a neoprogrammatori di tutte le età, che un altro programma più specifico, denominato Zortek.

Questa idea, di computer per la didattica, è quella che viene più di ogni altra messa in evidenza dal materiale illustrativo distribuito dalla Commodore per la presentazione della sua nuova macchina.

Oltre ai due già citati programmi, che permettono al neofita di scoprire con divertimento il linguaggio BASIC e le possibilità del suo nuovo giocattolo, un altro programma, permette al futuro game designer, di imparare l'Assembler.

Analizziamo ora le caratteristiche della macchina, che per molti aspetti sono pressoché identiche a quelle del PLUS 4.

NELL'INTERNO DEL C-16

Il microprocessore è un 7501, che combina insieme un 6510, con il generatore del suono, il sistema di controllo dell'INPUT/OUTPUT, del banco di memoria e del generatore di grafici.

Il generatore di suono, permette di avere solo due voci, oppure una voce ed un generatore di rumore bianco, con nove livelli di volume per le due voci di suono.

Notevole in questa macchina è la capacità grafica, che si avvale di una risoluzione di 320x200 pixel e di ben 121 colori, che sono ottenuti grazie a 8 diverse sfumature dei quindici colori di base, ai quali si aggiunge il nero.

I comandi grafici, sono direttamente gestibili in BASIC, e questo contrariamente alle abitudini della Commodore.

In questo modo, l'utente, pur disponendo di una risoluzione simile a quella del C-64, non dovrà ricorrere ad un'assurda serie di POKE, per vedere apparire qualcosa sullo schermo, ne tanto meno, ricorrere a linguaggi aggiuntivi, quali il SIMON, che sebbene siano adattissimi allo scopo, riducono ulteriormente la memoria del C-64 e, per contro, aumentano il costo del sistema.

Visto che la Commodore ha fatto finalmente lo sforzo di sistemare in

ROM questi comandi grafici, era anche logico pensare che utilizzasse una serie di comandi piuttosto potenti.

Infatti le varie istruzioni, pur nella loro apparente semplicità, permettono di realizzare diverse cose e di questo ci si accorge analizzando istruzioni come la CIRCLE, dove il numero dei parametri da utilizzare si spreca.

Forse sarebbe stato più comodo realizzare più istruzioni con meno parametri, in luogo di un'unica istruzione omnicomprensiva, ma questa tendenza sembra si stia diffondendo fra i costruttori.

A dire il vero alcune di queste istruzioni, ricordano le istruzioni del SIMON'S BASIC. In quel caso una delle istruzioni più articolate è l'ARC, con la quale è possibile tracciare un arco di circonferenza o ellisse, che abbia il centro nel punto x, y specificato, a partire dall'angolo z, fino all'angolo t.

Una cosa complicatissima anch'essa, che forse si poteva risolvere con qualche parametro in meno, pur conservando la stessa flessibilità, ed evitando i numerosi syntax error che, immancabilmente compaiono, nel corso dell'esecuzione di programmi contenenti tali istruzioni, in quanto sovente ci si dimentica questo o quel-

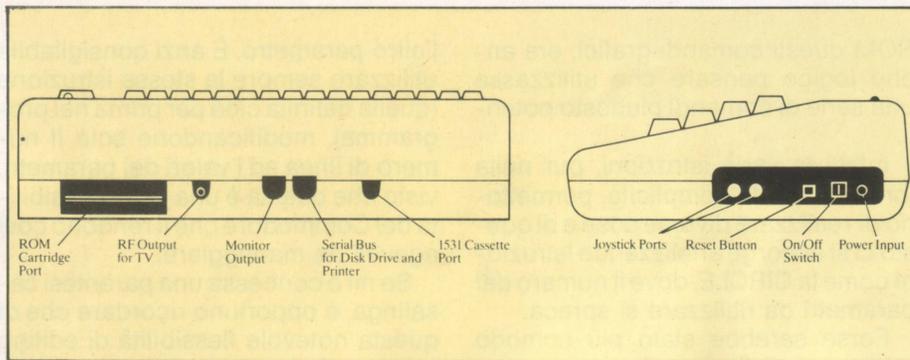
l'altro parametro. È anzi consigliabile utilizzare sempre la stessa istruzione (quella definita cioè per prima nel programma), modificandone solo il numero di linea ed i valori dei parametri, visto che questa è una delle possibilità dei Commodore che li rendono così agevoli da maneggiare.

Se mi è concessa una parentesi casalinga, è opportuno ricordare che di questa notevole flessibilità di editing sono stati dotati ad esempio l'M-24 dell'Olivetti, gli MSX, e via discorrendo.

Torna comunque alla mente, considerando questo articolatissimo comando, l'analoga funzione presente nel BASIC Spectrum, che a dire il vero, è anche più flessibile, considerando che con la stessa funzione è possibile disegnare linee, archi, cerchi ... in altre parole con una DRAW, seguita da due o tre parametri, si riesce a far fare al simpatico computer nero, un'infinità di cose, comprese delle stelle o dei poligoni, con un numero indeterminato di punte (sempre con una sola istruzione).

Una delle caratteristiche Sinclair, che in un certo qual modo è stata trasposta sul C-16 e sul PLUS 4, è la segnalazione della locazione di un errore di sintassi. Nei Sinclair, che





hanno un editing del tutto particolare, la segnalazione avviene mediante il posizionamento di un cursore lampeggiante vicino alla zona di errore, prima dell'accertazione della linea stessa, (che non diventa quindi linea di programma se non è sintatticamente corretta), mentre nel C-16, dopo la segnalazione dell'errore, tramite la pressione del tasto HELP, che è posizionato sulla tastiera in corrispondenza del tasto funzione F7/F8, si ha l'esecuzione automatica del flag (acceso-spento), sulla parola incriminata.

Finalmente non dovremo passare notti insonni per scoprire dove abbiamo sbagliato. Non è tutto qui però.

La macchina, o meglio queste due nuove macchine, sono state realizzate finalmente, pensando a chi poi le dovrà programmare.

Una delle caratteristiche negative dei Commodore, è infatti stata fino ad ora, la loro difficile programmazione, a cui faceva da contrappunto una eccezionale potenza latente, difficile però da utilizzare per l'utente medio, che si doveva accontentare di apprezzare le doti della sua macchina, solo utilizzando programmi realizzati da professionisti.

Questa macchina, quindi, oltre al tasto HELP, dispone di una serie di funzioni, veramente molto interessanti.

Parliamo ad esempio delle funzioni di utility. Tra di esse annoveriamo l'AUTO o il RENUMBER, con le quali rispettivamente si ottengono le numerazioni automatiche delle linee dei programmi e la rinumerazione delle stesse.

Molto più importanti sono però le altre funzioni, utili soprattutto in fase di debugging. Tra di esse troviamo la TRON, la TROF, e persino la TRACE. Con queste funzioni, è possibile visualizzare la linea che si sta eseguendo

in un determinato momento, ponendo così controllare i risultati parziali, per verificare, dove un programma sbaglia, e dove invece esegue correttamente le varie operazioni. Altre funzioni, permettono infine la gestione degli errori.

Fra le funzioni insistenti sugli altri Commodore, che meritano particolare attenzione, segnaliamo RESTORE

numero di linea, funzione che come è noto, permette di spostare il puntatore dei DATA, su di una specifica linea, cosa che fino ad ora era realizzabile sui Commodore, solo grazie all'uso di routine particolari, (salvo disporre ovviamente dell'onnipotente Simon).

Altre istruzioni degne di nota sono la PRINT USING, che permette all'utente di scegliersi un proprio formato di stampa, o la INSTR che permette la ricerca di una sottostringa all'interno di una stringa data.

Ovviamente come nel PLUS 4, sono disponibili quei comandi che permettono la realizzazione di "strutture".

Da notare che comandi di questo tipo, si trovano anche nel Simon.

Quindi possiamo supporre, che l'esperienza derivata dalla realizzazione e diffusione di questo linguaggio, che è un optional sui C-64, abbia convinto la Commodore, a realizzare delle

CARATTERISTICHE TECNICHE

Memoria

RAM 16K standard di cui 12K utilizzabili in Basic dall'utente. ROM 32K con sistema operativo e interprete Basic incorporati.

Microprocessore

7501 con clock da 0.89 a 1.76 MHz.

Video

40 colonne per 25 righe di testo.

Colori

121 colori (15 colori di base con 8 gradazioni più il nero).

Set di caratteri

Lettere maiuscole e minuscole, numeri e simboli.

Caratteri in negativo e lampeggianti.

Set di caratteri grafici Pet Commodore.

Display mode

Caratteri di testo.

Grafici in alta risoluzione e in multicolor.

Testo in schermo diviso/grafici in alta risoluzione o in multicolor.

Risoluzione grafica

320 X 200 pixel.

Suono

2 generatori di suono oppure 1 generatore di suono e un generatore di rumore bianco. 9 livelli di volume per le due voci.

Tastiera

Standard qwerty con 66 tasti.

4 tasti di controllo del cursore.

4 tasti di funzione programmati, che possono essere ridefiniti dall'utente per supportare fino a 8 funzioni prescelte. Tasti di controllo del colore.

Tasto di HELP.

Set di caratteri maiuscolo e minuscolo.

Set di caratteri grafici.

Colore

Grigio antracite.

macchine, che di sistema disponessero di quei comandi, scelta, che era opportuno fare.

Uno dei grossi svantaggi dei C-16, è la limitata quantità di memoria disponibile, che permette, o meglio non permette, di sfruttare in modo adeguato le risorse della macchina.

Si spera che in futuro espansioni di memoria vengano realizzate, se non dalla Commodore, perlomeno dalle aziende specializzate in questo settore.

Fra gli altri, difetti, segnaliamo la mancanza degli sprites, che caratterizzavano il C-64. È vero che possono essere sostituiti grazie ad una particolare gestione del video, simile a quella a finestre, ma quanto questa sia pratica rispetto agli sprites, solo gli utenti potranno poi dirlo. Gli altri difetti riguardano l'adozione di periferiche appositamente realizzate, quali il re-



gistratore e addirittura i joystick, come sul PLUS 4.

I dispositivi di I/O, comprendono

Input/Output

User port C16.

Porta seriale Commodore.

Porta per cartuccia Rom e per unità disco parallela.

2 porte per joystick.

Porta di collegamento per unità registratore C1531

Uscita monitor con segnale composito/croma/luma.

Input/output audio

Ingresso alimentazione di rete.

Caratteristiche principali

Basic esteso versione 3.5 incorporato, con oltre 75 comandi in grado di pilotare grafica e suono.

Monitor in linguaggio macchina incorporato, con 12 comandi disponibili.

Possibilità di creazione finestre su video.

Periferiche opzionali

SFS 481 unità a disco veloce

C 1542 unità a disco

C 1531 registratore a cassette

MCS 801 stampante a matrice a 7 colori

MPS 802 stampante a matrice

DPS 1101 stampante a margherita "letter quality"

C 1520 stampante/plotter a 4 colori

C 1703 monitor a colori

Altre periferiche

Commodore 16 funziona anche in collegamento con l'unità a disco C 1541, con la stampante a matrice C 1526 e con il monitor a colori C 1701.

Dimensioni

Altezza: 76.2 mm

Larghezza: 406.4 mm

Profondità: 203.2 mm

Alimentazione di rete

240 volt AC, 50 Hz con alimentatore esterno.

Consumo

8.5 watt massimo

uno slot per l'inserimento di cartidge, diverso da quello delle macchine precedenti (per evitare rischiosi tentativi di usare programmi delle altre macchine sul C 16), un'uscita per il tv domestico, un'uscita monitor, l'interfaccia per la stampante o il drive e quella per il registratore. Questi dispositivi si trovano sul retro della macchina; sulla destra troviamo invece l'ingresso per l'alimentazione, il tasto acceso/spento, un tasto di RESET, e le due porte per i joystick.

Le periferiche, come già accennato sono realizzate appositamente, salvo stampanti e drive.

SOFTWARE

Per quanto concerne la disponibilità di software, non ci resta che sfogliare il catalogo della Commodore Italiana.

Infatti considerando la scarsa diffusione della macchina, pur essendo disponibili programmi all'estero, non esiste ancora quel mercato sotterraneo di software, che permette ad alcuni utenti ad esempio del C-64, di disporre di migliaia di titoli.

I programmi che abbiamo reperito, riguardano innanzi tutto software educativo, con 11 titoli, che a giudicare dagli stessi devono riguardare soprattutto la matematica. Vi è poi un programma applicativo, un word processor per la precisione chiamato "SCRIPT PLUS" su cartidge, e 4 videogiochi.

Giancarlo Butti

NEI MEANDRI DEL COMMODORE 64

a cura di Franco Francia

LA GRAFICA IN BASSA RISOLUZIONE

Il mese scorso, parlando del modo testo, abbiamo visto le due principali tecniche di gestione dello schermo in bassa risoluzione.

La prima di esse consiste nello stampare l'insieme di caratteri desiderati attraverso l'istruzione PRINT; la seconda, leggermente più complessa ma decisamente più versatile, si basa invece su assegnamenti diretti nella memoria di schermo (ed eventualmente anche nella memoria colore, per chi possedesse un vecchio modello di COMMODORE 64) per mezzo di un ciclo di comandi POKE.

Un sistema per creare nella grafica tra-

mite queste tecniche consiste nell'utilizzare il set di caratteri implementato nella macchina. Avrete senza dubbio notato come la maggior parte dei tasti possieda in aggiunta alla funzione principale anche due riquadri visibili sulla loro parte frontale; tutti i simboli di sinistra sono ottenibili premendo il logo COMMODORE in contemporanea, mentre quelli di destra si richiamano per mezzo dello SHIFT.

Fra quanto visto in precedenza, la mia personale preferenza circa la tecnica da impiegare per le nostre applicazioni pratiche va senz'altro all'assegnamento diretto in memoria di schermo, in virtù della sua maggiore versatilità soprattutto nel campo dell'animazione.

Il programma seguente vi mostrerà come simulare il lampeggio di un oggetto, cambiandogli semplicemente il suo colore:

```
100 PRINT CHR$(147)
150 POKE 1388, 42
200 FOR C = 1 TO 20
250 POKE 55660, 6
300 FOR D = 1 TO 100
350 NEXT D
400 POKE 55660, 1
450 FOR D = 1 TO 100
500 NEXT D
550 NEXT C
```

Come potete notare, tutto questo è molto semplice da comprendere.

La linea 100 provvede a pulire lo schermo.

La linea 150 posiziona un asterisco nella locazione di schermo posta alla decima riga e quarta colonna.

Le linee 250 e 400 assegnano rispettivamente il colore blu e bianco all'asterisco.

Le linee 300/350 e 450/500 costituiscono semplicemente un breve ciclo di ritardo durante il quale il colore dell'asterisco viene mantenuto costante.

Le linee 200/550 definiscono il ciclo principale di cambiamento del colore dell'asterisco, eseguito ricorsivamente venti volte.

Proviamo adesso a creare un piccolo disegno, e quindi memorizzarlo sullo schermo:

```
10 clr:Poke59468,12
15 Poke53280,2:Poke53281,2:Print"
  2"
20 e=59466:g=e-2:Pokee+1,16:t=14:
  goto105~
25 Poke9,85:goto65
30 Pokee,15:Poke9,10:gosub70:ifbt
  henreturn
35 gosub95:goto95
40 Poke9,90:goto65
45 Poke9,95:gosub65:goto95
50 Poke9,105:goto65
55 Pokee,50:Poke9,200:gosub70:got
  o35
60 Pokee,15:fori=200to50step-2:Po
  ke9,i:next
65 Pokee,51
70 fori=1to15:next:Poke9,0:return
75 Pokee,85:Poke9,150:return
80 fori=1to500:next
85 fori=1to1250:next
90 fori=1to200:next
95 fori=1to50:next:return
100 Pokee+1,0:forx=1to30:Pokee+2,1
  94:Pokee+2,12:next:Pokee+1,16:
  return
105 b$="#####":c$="#####
  #####"
```

```
110 d$="RRR202##### W EW":e$="###
  #####  #####  #####":g$=
  "####"
115 f$="RQR202##### W EW":h$="##
  PRRR##### WE W ":i$="###P#RR
  R##### WE W "
120 Print"#####
  . NPM .#
125 Print" .sPc(9)"_ .sPc(8)
  ". FPOL .#
130 Print" .sPc(9)"NMM . ."
  sPc(7)"NMM MM
135 Print" . POLL _ _
  _ . N M MM
140 Print" * _ _ _NMM MNM MNM
  M _ _ _ _#
145 Print" ###NM MNM N _ _ MN _ MN _
  M NM ### _ _#
150 Print"##### MN _ M ### _ _ _ _
  IN _ M### _ _#
155 Print" | |# _ _ _ _ _ _ _ _ _ _
  |# _ _ _ _ _ _ _ _ _ _#
160 Print" | | | | | | | | | | | |
  | | | | | | | | | | | |
165 Print" +"):forx=1to34:Print"~
  ";next:Print"+N#
170 forx=1to37:Print"~":next:Print"
  t"N#
```

```

10 POKE 53281, 1
15 PRINT CHR$(147)
20 C = 54272
25 POKE 1106, 87: POKE 1106 + C, 2
30 POKE 1146, 102: POKE 1146 + C, 6
35 POKE 1186, 102: POKE 1186 + C, 6
40 POKE 1145, 64: POKE 1145 + C, 6
45 POKE 1147, 64: POKE 1147 + C, 6
50 POKE 1225, 78: POKE 1225 + C, 6
55 POKE 1227, 77: POKE 1227 + C, 6
60 GOTO 60
    
```

Per chi non avesse sufficiente voglia di digitare questo programma, sveliamo che apparirà un omino in alto a sinistra.

L'animazione sfrutta una tecnica affine a quella impiegata per il lampeggio dell'asterisco nell'esempio precedente; la differenza è data che l'alternanza non riguarda più il colore di un carattere, ma il carattere stesso.

Infatti, commutando rapidamente due o più caratteri grafici nella stessa locazione di schermo, si creeranno semplici forme di movimento costituenti la base di una ancora grezza animazione.

Per vedere il nostro omino tentare inutilmente di volare, aggiungete al programma qui sopra le seguenti nuove linee:

```

60 FOR R = 1 TO 100
65 POKE 1145, 99
70 POKE 1147, 99
75 POKE 1145, 64
    
```

```

80 POKE 1147, 64
85 NEXT R
90 GOTO 90
    
```

Due differenti sets di caratteri sono disponibili per i nostri grafici, ma soltanto uno di essi alla volta può essere presente sullo schermo.

Avrete già avuto modo di notare come premendo contemporaneamente i tasti SHIFT e COMMODORE, eventuali lettere maiuscole presenti sullo schermo diventassero minuscole e viceversa; ebbene, il primo caso rappresenta il set numero uno, mentre il secondo rappresenta invece il set numero due.

Per scegliere il set da impiegare nei vostri programmi, potete utilizzare le seguenti istruzioni:

```

POKE 53272, 21 (set numero uno)
POKE 53272, 23 (set numero due)
    
```

Anche caratteri in negativo (in inglese: reverse) sono facilmente ottenibili aggiungendo ad ogni rispettivo codice di schermo il valore 128.

Ritorniamo adesso alla nostra cara ed un po' trascurata istruzione PRINT. Per farci perdonare da parte dei suoi sostenitori, diremo che le sue migliori qualità sono rappresentate da una maggiore velocità di esecuzione rispetto ad un'istruzione POKE, nonché dalla possibilità di combinare fra loro più caratteri per crearne uno complesso (mentre POKE ne può indiriz-

zare soltanto uno alla volta).

Per comparare le rispettive velocità di POKE e PRINT, provate a digitare il seguente programma:

```

10 PRINT CHR$(147)
15 FOR R = 1024 TO 2023
20 POKE R, 42
25 POKE R + 54272, 14
30 NEXT R
35 FOR K = 0 TO 500
40 NEXT K
45 PRINT CHR$(5)
50 PRINT CHR$(147)
55 FOR R = 1 TO 999
60 PRINT "*";
65 NEXT R
70 PRINT CHR$(19)
    
```

Si noterà come PRINT sia circa due volte più rapida di POKE; PRINT può essere inoltre maggiormente velocizzata facendogli stampare più di un asterisco per volta, sfruttando quindi la sua seconda caratteristica positiva.

In conclusione di puntata, eccovi un listino dimostrativo di tutto quanto visto finora; si tratta di un breve disegno animato raffigurante un'invasione dallo spazio ... ma non vi diremo di più. È interessante notare come, senza barare, sia interamente basato su grafica in bassa risoluzione, utilizzando i caratteri disponibili in dotazione.

Buon divertimento, ed arrivederci al mese prossimo.

```

175 Printtab(36)"N":Printtab(35)"N
   ":Printtab(34)"N N":Printtab(3
   4)"N N N
180 Printtab(35)"I":Printtab(35)"I
   "/L":Printtab(34)"NI":Print
   tab(33)"NNI
185 Printtab(32)"NN I":Printtab
   (31)"NN I":Printtab(30)"NN
   I"
190 Print:Print" ":gosub485
195 Print:Print"          "tab(18)"
   _":Printtab(17)"N_____M":Print
   tab(17)"I_____I"
200 fori=1to39:Print" ":next:Prin
   t
205 Printtab(19)"NP":forx=1to5:P
   rinttab(19)"I"
210 next:Printtab(t)"_____L_____
215 fori=1to7:Printtab(t)"L_____L
   _____":next
220 Printtab(19)"[W]":Printtab(19)
   "_____I"
225 ifc=0thengosub80:goto695
230 Printtab(24)"          "":gosub750:
   gosub455
235 forx=1to20:Print"#####!/?":gosu
   b45:Print"#####":gosub30:nex
   t:c=1
    
```

```

240 Print"#####
245 Print"#####tab(15)"  "sPc(6)"r
   "":Printtab(16)"I_____I"
250 Printtab(15)"#####I#####":Pri
   nttab(t)"#####RR#####
255 Printtab(t)"<QM NQ>":Pr
   inttab(t)"M N"
260 Printtab(t)"FOI":Pr
   inttab(t)"MN"
265 Printtab(15)"_____I":Prin
   ttab(15)"#####EE#####
270 Printtab(12)"_R#####N#####
   _":Printtab(11)"#####
   #####
275 Printtab(10)"#####N#####M
   #####":tab(51)"I#####
   #####I"
280 Printtab(11)"I#####N#####
   I":Printtab(11)"I_____I"
285 Printtab(10)"NNNN"sPc(12)"MMMM
290 ifcthenforx=1to10:Print" "
   ":next:gosub85:gosub100
295 gosub85
300 ifcthenPrint:Print"#####ta
   b(15)"#####N#####
   I"
305 ifcthengosub85:c=0:a=1:goto245
    
```

```

310 gosub100:if a then Pokee+1,0:end
315 Print"┌───┐
320 Print"U───┐I"SPc(9)" n r
   O───┘ n d
325 k$="┌───┐
330 Print"II"SPc(9)"II"SPc(9)"I───┘
   ;k$;"I d
335 Print"II"SPc(9)"II"SPc(9)"IO";
   k$;"PI d
340 Print"IJ───┐KI   r   II
   ┌──────────┐II d
345 if d=1 then Print tab(18)"┌───┐
350 Print"l.... W W I ┌───┐
   II ┌──────────┐II d
355 j$="   II ┌──────────┐II d
360 Print"┌───┐┌───┐┌───┐";j$
365 Print k$;"┌───┐┌───┐┌───┐";j$
370 Print tab(13)"┌───┐┌───┐"SPc(6)"I───┘
   ;k$;"I d
375 Print tab(12)"N┌───┐"SPc(7)"IO";k$
   ;"PI d
380 Print"┌───┐"SPc(7)"┌───┐┌───┐I"SPc(7)
   )"II n n n n II d
385 Print"┌───┐II"SPc(7)"┌───┐┌───┐I"SPc(7)
   )"II l l l l l l P I d
390 Print"┌───┐II"SPc(6)"O II"SPc(7)
   )"II ┌──────────┐P┌───┐II d
395 if d=1 then Print tab(8)"┌───┐┌───┐┌───┐
400 Print" UKJI P┌───┐IL"SPc(7)"I
   l l l l l l l l I d II d
405 Print"┌───┐┌───┐M┌───┐UIM"SPc(6)"
   II ┌──────────┐II d
410 Print"J┌───┐K   N JK M   II
   lJKJKJKI l l l l d
415 Print"O";for x=1to37:Print"┌───┐";
   :next:Print"PI d
420 Print"┌───┐OPP";for x=1to31:Print"
   ┌───┐";next:Print"OOP┌───┐
425 Print"┌───┐II"SPc(t)"U───┐I"SPc(7)"II d
430 Print"┌───┐II"SPc(t)"J┌───┐┌───┐K"SPc(7)"II d
435 for i=1to2:Print"┌───┐II"SPc(15)"I
   ┌───┐II"SPc(8)"II";next
440 Print" LvN"SPc(15)"II"SPc(15)"II"SPc(8)"MLv";Print tab(19)"┌───┐┌───┐
   ┌───┐
445 if b then 835
450 gosub790:c=1:9oto195
455 Print"┌───┐┌───┐┌───┐"SPc(17)"
   ┌───┐"
460 for x=1to8:Print tab(11)"I"SPc(15)"I";next
465 Print tab(11)"M"SPc(15)"N";Print tab(12)"M"SPc(13)"N"
470 Print tab(13)"M"SPc(11)"N";Print tab(t)"M"SPc(9)"N"
475 Print tab(15)"M"SPc(7)"N";Print tab(16)"M Q N d
480 Print tab(17)"┌───┐┌───┐";return
485 Print tab(39);for x=1to28:Print"┌───┐┌───┐";9osub25:9osub90
490 next:for x=1to4:Print"┌───┐┌───┐";9osub40:9osub35:next

```

```

495 for x=1to2:Print"┌───┐┌───┐";9osub45
   :next:for x=1to2:Print"┌───┐┌───┐";9
   osub45:next
500 Print"┌───┐┌───┐";9osub45:Print"┌───┐
   ┌───┐";9osub45:Print"┌───┐┌───┐";9osub
   45
505 Print"┌───┐┌───┐";9osub45:for x=1to2
   :Print"┌───┐┌───┐";9osub45:next:for
   x=1to5
510 Print"┌───┐┌───┐";9osub50:next:Pr
   int"┌───┐┌───┐┌───┐";9osub100:Pr
   int"┌───┐┌───┐";9osub80:9osub80
515 Print:Print b$tab(34)"┌───┐┌───┐┌───┐
   ┌───┐";
520 9osub30:for x=1to2:Print"┌───┐┌───┐
   ┌───┐";9osub30:next
525 9osub80:for x=1to7:Print"┌───┐┌───┐
   ┌───┐";9osub30:9osub35:next:9osub80
   :Print
530 Print b$tab(34)"┌───┐┌───┐┌───┐┌───┐
   ┌───┐";9os
   ub30:for x=1to2:Print"┌───┐┌───┐";
   9osub30:next
535 9osub35:for x=1to6:Print"┌───┐┌───┐
   ┌───┐";9osub30:9osub90:next:Print
540 9osub80:Print b$tab(34)"┌───┐┌───┐
   ┌───┐";9osub
   30:Print"┌───┐┌───┐";9osub30:Print"
   ┌───┐┌───┐";9osub30
545 9osub85:for x=1to6:Print"┌───┐┌───┐
   ┌───┐";9osub30:next:Print"┌───┐";9osub3
   0:9osub80
550 9osub85:Print b$tab(22)"┌───┐┌───┐
   ┌───┐";9o
   sub30:Print"┌───┐┌───┐┌───┐";9osub30
555 Print"┌───┐┌───┐┌───┐";
560 for x=1to7:Print"┌───┐┌───┐┌───┐";9osub
   30:next:9osub85
565 Print:Print c$tab(23)"┌───┐";for x=
   1to7:Print"┌───┐┌───┐┌───┐";9osub30:n
   ext:9osub80
570 Print:Print b$tab(13)"┌───┐┌───┐";9osu
   b30:Print"┌───┐┌───┐┌───┐";9osub30:Pr
   int"┌───┐┌───┐┌───┐";
575 for x=1to5:Print"┌───┐┌───┐┌───┐";9osub
   30:next:9osub85
580 for x=1to2:Print"┌───┐┌───┐";9osub30
   :next:Print"┌───┐┌───┐";9osub30:Prin
   t"┌───┐";
585 9osub30:Print"┌───┐┌───┐┌───┐";9osub3
   0:Print"┌───┐┌───┐";9osub30:Print"┌───┐
   ┌───┐";9osub30
590 for x=1to3:Print"┌───┐┌───┐";9osub3
   0:next:9osub85:Print"┌───┐┌───┐┌───┐
   ┌───┐";for x=1to3
595 Print"┌───┐┌───┐┌───┐";9osub30:next:9
   osub80:Print"┌───┐┌───┐┌───┐┌───┐click!";
   :9osub40
600 9osub90:Print"┌───┐┌───┐┌───┐┌───┐";9o
   sub85:9osub100:9osub85
605 Print c$tab(34)"┌───┐┌───┐┌───┐┌───┐
   ┌───┐";for x=1to10
   :Print"┌───┐┌───┐┌───┐┌───┐";b=1:9os
   ub30:next
610 Print"┌───┐┌───┐┌───┐┌───┐";9osub30:Pr
   int"┌───┐┌───┐";9osub30:Print"┌───┐
   ┌───┐";
615 Print tab(t)"┌───┐┌───┐┌───┐┌───┐";
   9osub30:for x=1to3:Print"┌───┐┌───┐

```

```

%J!!!";:gosub30
620 next:forx=1to11:Print"!!.. !!!!"
;:gosub30:next:Print" . "::gosu
b30:Print"!!! "
625 gosub80:gosub85:b=0:Printb$;"20
20000";
630 gosub55:Print"202-2000EW0000";:g
osub55
635 Print"R202-2000EW0000";:gosub
55:Print"RR202-200000 EW0000"
;:gosub55
640 Printd$;:gosub55:forx=1to17:Pr
inte$;d$;:gosub55:next:gosub80
645 Print"!!!. !!!!";:gosub30:Print
" .!!";:gosub30:gosub80:forx=1to
6
650 Print" !!! 00000000 !!! 0000";:gos
ub30:next:Print"!! 000000";
655 gosub55:Print" 0000";:gos
ub85:forx=1to5:Print" 0000.
0. !!!!";:gosub30
660 next:gosub85:Print" 0000 00";:
gosub30:Print" 00000000 0";:gos
ub30
665 Print" 00";:gosub30:forx=1to2:
Print" .!!";:gosub30:next
670 Print" 0. !!";:gosub30:Print" 0!"
;:gosub30:Print"E";:gosub30:gos
ub80
675 forx=1tot:Printe$;f$;:gosub55:
next:Printe$;f$;"!N";:gosub55:
Printe$;
680 Print"RQR202N000000W000";:gosub
55:Printe$;
685 Print"RQRN000000W000";:gosub55:P
rinte$;"RQN000000";:gosub55
690 Printe$;"RN0000";:gosub55:Prin
t"00 N";return
695 Print9$;"20R0";:gosub55:Print"0
R000000";:gosub55
700 Print"RQR000000 W000";:gosub55:P
rint"2P2RQR000000E W000";:gosu
b55:Printe$;
705 gosub55:forx=1to12:Printe$;h$;
:gosub55:next:gosub80:Print"!!!
!! !!";:gosub30
710 Print" 0000";:gosub30:gosub85:P
rint"R0000";:gosub30
715 Print" !! !!";:gosub30:Print" 20!
00";:gosub30:gosub80:gosub85
720 forx=1to13:Print" 0000 00";:gos
ub40:gosub45:gosub30:next:gosu
b80
725 Print" 2 2 !!";:gosub30:forx=1to
2:Print" _ 0. !!!!";:gosub30:next
730 gosub80:forx=1to2:Print" _!!! !!"
;:gosub30:next:Print" L!!!";:gos
ub30
735 forx=1to13:Print" 0000 00";:gos
ub40:gosub45:gosub30:next:Prin
t" 0!";
740 gosub80:Print"00000. 0!";:gosub3
0:Print" 0000 !!";:gosub30
745 Print" 00!";:gosub30:gosub80:fo

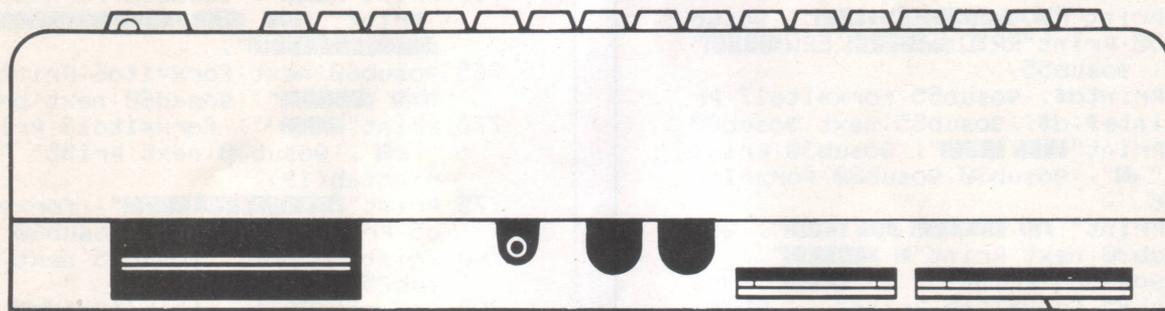
```

```

rx=1to12:Printe$;i$;:gosub55:n
ext:goto315
750 Print"0000 0.00000";:gosub60:Pri
nt" _ 0000000000";:gosub60:Print
90sub60:Printb$tab(20)"0000";:f
orx=1to13:Print" 0000 00";:gosu
b60:next
760 Print" 0000";:gosub60:forx=1to7
:Print" 0000 2000000000000000000000
0000000000";
765 gosub60:next:forx=1to6:Print"
0000 200000";:gosub60:next:b=1
770 Print"0000!";:forx=1to13:Print
" .!!";:gosub30:next:Print" " :P
rinttab(19);
775 Print"00000000000000000000";:forx=1t
o5:Print" [200]0000";:gosub60
780 Print" 0000";:gosub45:next:go
sub75:Print"0000 "
785 forx=1to99:Printtab(12)"0000
":next:Poke9,0:r
eturn
790 Print"000";:forx=1to17:Print" 0
00";:gosub30:next:gosub80:b=1
795 forx=1to5:Print" 0000 00";:next
:Print" 0 00";:gosub30
800 Print"U0 00";:gosub30:Print"-0
00";:gosub30:gosub80
805 Print"-000 00";:forx=1to2:Prin
t" 0000 00";:next:gosub30
810 Print" 0000 00";:gosub30:gosub8
0:forx=1to11:Print" 0000 00";n
ext
815 gosub30:Print" 0000 0";:gosub30:
fori=1to4000:next:Print"000000
0000";
820 Print"click!20U0000000";:gosub
40:gosub90:Print" " :Print
9$tab(17);
825 gosub85:gosub100:forx=1to20:Pr
int" M!N0000";:gosub30
830 Print" 0000";:gosub30:next:d=
1:goto240
835 gosub100:forx=1to6:Print"0";Pr
int"20000";:Print" 000000
00";:gosub60
840 Print" " :gosub60:next:Pr
inttab(18)" 0 00";:gosub30:for
x=1to11
845 Print" 0000 00";:ifx=7orx=9orx=
10thenPrintspc(30)"000000000000
";
850 ifx=8thenPrintspc(31)"000000000000
0000";
855 next:gosub30:Print" 0 00";:gos
ub30:Print" 0000 00";:forx=1to2
860 Print" 0000 00";:next:gosub30:P
rint"-0000 00";:gosub30:Print"-
0000 00";:gosub30
865 Print"U0000 00";:gosub30:forx=1
to5:Print" 0000 00";:next:gosub
30
870 forx=1to17:Print" 0000 00";:gos
ub30:next:c=1:goto195

```

COME UTILIZZARE AL MEGLIO LA USER PORT DEL VIC 20 E DEL C64



User Port

Nel VIC 20 e nel C 64 è montata di serie un'interfaccia parallela ad 8 bit (figura 1). Questa viene chiamata, in gergo, "porta d'utente". Si tratta proprio del portone di casa del computer, una via di comunicazione con il mondo circostante, in grado di emettere e ricevere dati.

Questa funzione viene pilotata da un programma, e perciò possono essere svolti diversi compiti, che altrimenti dovrebbero essere sbrigati ricorrendo ad un inimmaginabile guazzabuglio di hardware. Tramite la porta d'utente è possibile ad esempio effettuare l'elaborazione e visualizzazione di dati di misura, la decodifica di segnali Morse, oppure il pilotaggio di apparecchiature od impianti (modelli ferroviari, selettori telefonici, effetti di luce, riscaldamento, proiettori di diapositive, impianti protezione della casa, eccetera). In questo settore non esistono limiti alla fantasia.

IL CIRCUITO PUO' ESSERE COSTRUITO IN UN ATTIMO

Gli esempi applicativi descritti finora necessitano evidentemente di una certa quantità di hardware, che però, grazie al valido sussidio del computer, è molto ridotta. Gli elettronici dilettanti non saranno perciò messi di fronte a problemi insolubili. Le conoscenze attuali consentono di progettare programmi che permettono di indirizzare nella direzione voluta il flusso dei dati.

Il nucleo di tali programmi è costituito da routine che permettono il corretto

"utilizzo" della porta di utente mediante comandi di pilotaggio. Questo articolo tratta proprio tale argomento, in quanto esso viene accennato in maniera veramente superficiale nei manuali Commodore.

Compito dell'"addestratore all'uso

della porta" (figura 2) è il rendere visibili i livelli logici ("0" - "1", oppure "basso" - "alto") presenti alla porta di utente. Se il livello presente sul conduttore di porta è "alto" (+ 5 V), si accende il relativo LED, mentre se il livello è "basso" il LED rimane spento.

È utile, ma non assolutamente necessario, effettuare il collegamento tra porta e strumento mediante un connettore a presa e spina, al quale potranno essere anche collegati; circuiti (pilotati dalla porta). Tramite questo connettore potranno inoltre essere immessi dati nel computer.

Contemporaneamente, i LED indicheranno lo stato dei rispettivi segnali, per esempio per scopi di controllo.

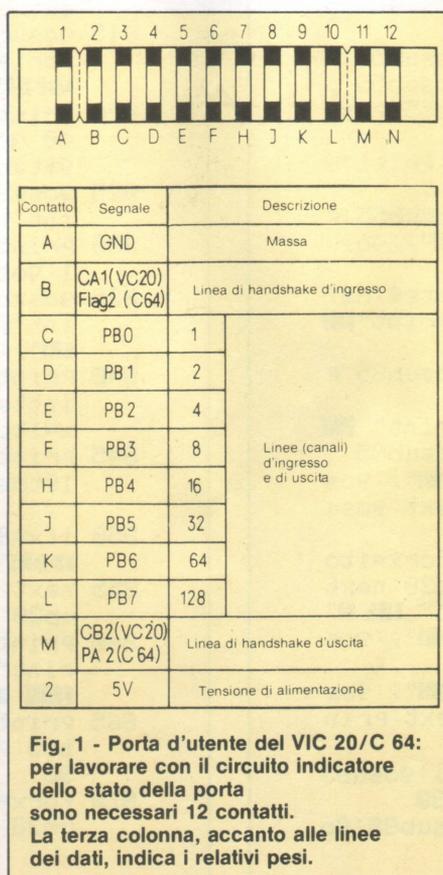
Anche l'alimentazione di questo apparecchio viene prelevata dalla porta d'utente: infatti, anche con i circuiti supplementari, la corrente assorbita rimane inferiore a 100 mA.

Il carico dei restanti conduttori di porta, tramite gli ingressi dei due circuiti integrati piloti 74LS04, non è invece critico.

DUE CONDUTTORI PER UNA STRETTA DI MANO

I collegamenti ai piedini della porta d'utente (figura 1) potranno lasciare perplessi in due soli punti, quelli cioè che riguardano i conduttori di handshake per l'ingresso e l'uscita.

Questi conduttori sono in grado di pilotare il traffico dei dati attraverso la porta d'utente mediante riconoscimen-



to del messaggio. Per esempio, il conduttore di handshake in emissione può far sì che un circuito collegato alla porta possa raccogliere dati solo entro un terminato intervallo di tempo.

Ma di questo parleremo più avanti.

Le otto linee di dati della porta sono bidirezionali. Ciò vuol dire che ciascuna linea di dati (detta pure "canale") può essere predisposta come uscita o come ingresso, mediante un'opportuna istruzione del programma. Una modifica della disposizione scelta in precedenza è sempre possibile, ancora mediante istruzioni nel programma.

All'interno dei computer, i conduttori di porta vanno ad un componente universale di interfaccia, che si chiama VIA (Versatile Interface Adapter = adattatore di interfaccia versatile), ma questo non deve interessarci ulteriormente.

Le istruzioni di pilotaggio per la porta d'utente vengono impartite tramite il comando BASIC "POKE". In questo modo è possibile raggiungere le variabili del sistema esecutivo relative alla porta di utente ed alimentarle con gli opportuni valori. A seconda di questi valori, il sistema operativo del computer mette in atto l'istruzione di pilotaggio.

SCelta DELLA DIREZIONE DI TRASFERIMENTO DEI DATI

Dopo l'accensione del computer, verranno collegate automaticamente per l'immissione tutte le linee dei dati. Il circuito didattico per l'uso della porta visualizza questa condizione, in quanto tutti i LED sono accesi. Tramite il "registro della direzione dati", che è una variabile del sistema, è possibile modificare la condizione di ciascun canale, per farlo diventare un'uscita e, naturalmente, farlo ridiventare un ingresso. Nel VIC 20, il registro della direzione dati può essere raggiunto all'indirizzo DDRB = 37138, mentre nel C 64 questo indirizzo è DDRB = 56579. La relativa istruzione di controllo è:

POKE DDRB, X

dove X può essere un numero decimale qualsiasi, compreso tra 0 e 255. Quando viene impartita l'istruzione di controllo, il numero decimale scelto carica il corrispondente numero binario sugli otto conduttori dei dati della porta (i relativi pesi sono elencati in figura 1). Ciascun LED spento segnala ora che il relativo

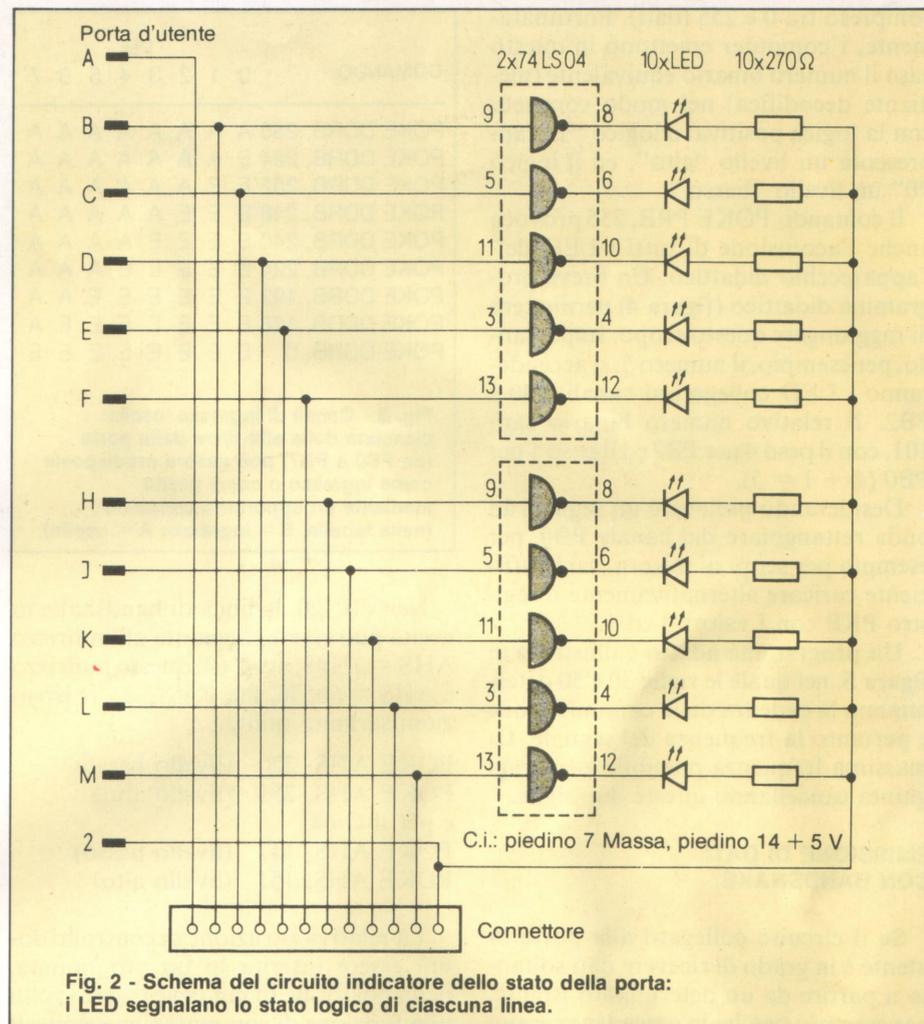


Fig. 2 - Schema del circuito indicatore dello stato della porta: i LED segnalano lo stato logico di ciascuna linea.

conduttore di porta è predisposto come uscita.

L'istruzione POKE DDRB, 255 predisporre, per esempio, tutti i canali come uscite. Questo è a prima vista sconcertante, in quanto, essendo il corrispondente numero binario 1111 1111, dovrebbero risultare accesi tutti i LED, e questa condizione segnala che tutti i canali sono predisposti per funzionare da ingressi.

In realtà, i livelli logici "0" ed "1" sono correlati alla "logica positiva". Gli home computer Commodore funzionano però in "logica negativa", ed in questo caso le attribuzioni dei livelli logici sono invertite: il logico "0" corrisponde ad un livello "alto" ed il logico "1" ad un livello "basso". Per questo motivo, tutti i LED degli otto canali di questo circuito rimangono spenti.

Se, per esempio, l'istruzione fosse POKE DDRB, 240, il relativo numero binario avrebbe la seguente forma: 1111 0000.

Ne consegue che ora i canali da PB0 a PB3 saranno predisposti come ingressi ed i canali da PB4 a PB7 saranno predisposti come uscite. In questo modo, sarà possibile scegliere qualunque combinazione di ingressi ed uscite.

La figura 3 elenca le istruzioni di pilotaggio per alcune di queste combinazioni. Per l'emissione o per la lettura di parole di dati complete da 8 bit, tutti i canali dovranno naturalmente essere predisposti come uscite oppure come ingressi.

Se il registro della direzione dati è stato caricato con il valore 255, potranno essere emessi dati tramite il "registro dei dati". Nel VIC 20, il registro dei dati ha l'indirizzo PRB = 37136, mentre nel C 64 questo indirizzo è PRB = 56577. È sufficiente trasferire i dati in questo registro, mediante una POKE. La relativa istruzione è la seguente:

POKE PRB, X

dove X è ancora un numero decimale

compreso tra 0 e 255 (dati). Fortunatamente, i computer emettono in questo caso il numero binario equivalente (mediante decodifica) nel modo consueto con la logica positiva: il logico "1" rappresenta un livello "alto", ed il logico "0" un livello "basso".

Il comando POKE PRB, 255 provoca anche l'accensione di tutti i LED dell'apparecchio didattico. Un breve programma didattico (figura 4) permetterà di raggiungere questo scopo. Impostando, per esempio, il numero 5, si accenderanno i LED collegati ai canali PB0 e PB2. Il relativo numero binario sarà 101, con il peso 4 per PB2 ed il peso 1 per PB0 (4 + 1 = 5).

Desiderando prelevare un segnale ad onda rettangolare dal canale PB0, per esempio per scopi di misura, sarà sufficiente caricare alternativamente il registro PRB con i valori 0 ed 1.

Un programma adatto è illustrato in figura 5, nel quale le righe 30 e 50 determinano la cadenza della commutazione e pertanto la frequenza del segnale. La massima frequenza possibile viene raggiunta cancellando queste due righe.

EMISSIONE DI DATI CON HANDSHAKE

Se il circuito collegato alla porta di utente è in grado di ricevere dati soltanto a partire da un determinato istante, per esempio perché in precedenza erano presenti alla porta d'utente dati non validi, ecco che entra in gioco il conduttore di handshake di emissione. Anche questa linea può essere predisposta, mediante un'istruzione di controllo, a livello alto oppure basso, a seconda di quanto richiede il circuito per effettuare il trasferimento dei dati.

| COMANDO | PB | | | | | | | |
|----------------|----|---|---|---|---|---|---|---|
| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| POKE DDRB, 255 | A | A | A | A | A | A | A | A |
| POKE DDRB, 254 | E | A | A | A | A | A | A | A |
| POKE DDRB, 252 | E | E | A | A | A | A | A | A |
| POKE DDRB, 248 | E | E | E | A | A | A | A | A |
| POKE DDRB, 240 | E | E | E | E | A | A | A | A |
| POKE DDRB, 224 | E | E | E | E | E | A | A | A |
| POKE DDRB, 192 | E | E | E | E | E | E | A | A |
| POKE DDRB, 128 | E | E | E | E | E | E | E | A |
| POKE DDRB, 0 | E | E | E | E | E | E | E | E |

Fig. 3 - Canali di ingresso/uscita: ciascuna delle otto linee della porta (da PB0 a PB7) può essere predisposta come ingresso o come uscita mediante un opportuno comando (nella tabella, E = ingresso; A = uscita).

Nel VIC 20, la linea di handshake in uscita può essere raggiunta all'indirizzo AHS = 37148; nel C 64, questo indirizzo è AHS = 56576. Per il VIC 20, le istruzioni saranno quindi:

```
POKE AHS, 220 (livello basso)
POKE AHS, 255 (livello alto)
e per il C 64:
POKE AHS, 147 (livello basso)
POKE AHS, 151 (livello alto)
```

La relativa istruzione di controllo dovrà essere inserita in un programma, nella posizione in cui deve essere svolta una funzione di commutazione, come il trasferimento di dati. La reazione a questo segnale è demandata al circuito esterno. La figura 6 mostra un programma dimostrativo di gestione dell'handshake per il VIC 20.

Per esempio, un circuito esterno potrà ricevere dati dalla porta d'utente quando la linea di handshake sarà a

livello basso. In caso contrario, i dati non sono validi, perché sono destinati ad un altro circuito.

Per prima cosa, la riga 20 del programma emette la parola di dati "0". Per il circuito esterno, questa non è valida, poiché la linea di handshake è ancora a livello alto.

Soltanto dopo che alla porta sarà stato applicato il dato valido 127, esso verrà trasferito. Affinché, nel ciclo successivo, la parola "0" venga nuovamente dichiarata non valida, avviene nella riga 70 un reset del segnale di handshake. Il processo può essere facilmente seguito mediante questo circuito didattico.

INSERIMENTO DATI TRAMITE LA PORTA

Il primo passo per l'immissione di dati tramite la porta d'utente, è quello di stabilire la direzione del flusso di dati nel senso dell'ingresso. L'attribuzione di controllo POKE DDRB, 0 svolge questa funzione. D'ora in poi, tutti i canali saranno a livello alto, cosicché i LED del circuito didattico saranno accesi.

Ora, ciascun canale potrà essere portato a livello basso, collegando per esempio i relativi terminali a massa. In questo modo, è possibile predisporre un numero binario qualsiasi, visualizzandolo sul circuito didattico. Commutando, per esempio, i canali PB1 e PB6 a livello basso, i LED indicheranno il numero binario 1011 1101, che corrisponde al numero decimale 189. Per trasferire i dati nel computer, viene interrogata la porta d'utente, per esempio con il comando PRINT PEEK (PRB), dove PRB è ancora l'indirizzo del "registro dati". Poiché l'interrogazione è abbinata

```
10 POKE DDRB, 255
20 INPUT "NUMERO"; X
30 POKE PRB, X
40 GOTO 20
```

Fig. 4 - Emissione dati.
Nel VIC 20 si ha DDRB = 37138 (C 64: 56579) e PRB = 37136 (C 64: 56577).

```
10 POKE DDRB, 255
20 POKE PRB, 1
30 FOR I=1 TO 10: NEXT I
40 POKE PRB, 0
50 FOR I=1 TO 10: NEXT I
60 GOTO 20
```

Fig. 5 - Emissione segnali:
questo programma produce un segnale ad onda rettangolare in corrispondenza alla linea di porta PBO.

```
10 POKE 37138, 255
20 POKE 37136, 0
30 FOR I=1 TO 500: NEXT I
40 POKE 37136, 127
50 POKE 37148, 220
60 FOR I=1 TO 500: NEXT I
70 POKE 37148, 255
80 GOTO 20
```

Fig. 6 - Uscita con handshake:
il livello basso della linea di handshake in uscita convalida la parola di dati 127.

```
10 POKE DDRB, 0
20 PRINT "VALORE INGRESSO="; PEEK (PRB)
30 GOTO 20
```

Fig. 7 - Immissione dati:
è sufficiente interrogare lo stato del registro dati PRB (per gli indirizzi, vedi figura 4).

```

10 POKE 37138,0
20 PRINT"ATTENDERE LO SBLOCCO"
30 WAIT 37149,16
40 PRINT"PAROLA DI DATI ATTUALE";PEEK(37136)
50 GOTO 20
    
```

Fig. 8 - Ingresso con handshake. Lo stato del registro dati viene interrogato solo dopo una commutazione di livello sulla linea di handshake d'ingresso.

```

10 POKE 37138,240
20 POKE 37136,0
30 PRINT"ATTENDERE LO SBLOCCO"
40 WAIT 37149,16
50 IF PEEK(37136)=11 THEN POKE 37136,128
60 FOR I=1 TO 200:NEXT I
70 GOTO 20
    
```

Fig. 9 - Sistema misto di ingressi/uscite. Un livello basso sulla linea di porta PB2 ha come conseguenza un livello alto sulla linea di porta PB7.

```

10 POKE 37138,255
20 FOR X=0 TO 7
30 Y=2^X
40 POKE 37136,Y
50 FOR I=1 TO 100:NEXT I
60 NEXT X
70 POKE 37136,0
80 POKE 37148,255
90 FOR I=1 TO 100:NEXT I
100 POKE 37148,220
110 GOTO 20
    
```

Fig. 10 - Punto luminoso mobile: il circuito visualizza un punto luminoso in movimento

```

10 POKE 37138,255
20 DATA 1,3,7,15,31,63,127,255,255,127,63,
31,15,7,3,1,0
30 POKE 37148,225
40 READ A
50 POKE 37136,A
60 FOR I=1 TO 50:NEXT I
70 IF A=0 THEN RESTORE
80 POKE 37148,220
90 FOR I=1 TO 50:NEXT I
100 GOTO 30
    
```

Fig. 11 - Banda luminosa: a causa dei valori contenuti nella riga DATA, i LED mostrano una banda luminosa che si allunga e si accorcia.

ta con un PRINT, il computer visualizzerà sullo schermo il numero decimale che corrisponde al numero binario impostato sulla porta d'utente. In figura 7 è listato un breve programma con il quale sarà possibile fare esercizi.

INGRESSO DATI CON HANDSHAKE

Come per l'emissione dei dati, anche i dati validi per l'ingresso sono contrassegnati mediante un segnale di riconoscimento. Questo segnale potrà, per esempio, indurre il computer ad interrompere l'elaboratore di un programma fino a quando non incontrerà un segnale di

riconoscimento. Il segnale di riconoscimento appare sulla linea di handshake d'ingresso.

Quando il livello logico su questa linea commuterà da alto a basso, potrà essere proseguita l'elaborazione del programma.

L'istruzione di controllo per il VIC 20 sarà in questo caso:

WAIT 37149, 16

e per il C 64

WAIT 56589, 16

Con il programma di figura 8, il circuito didattico evidenzierà questa con-

nessione. Nella riga 10, i canali vengono predisposti come ingressi. A causa della riga 30, il VIC 20 attenderà soltanto che al contatto M della porta d'utente non apparirà un fronte di commutazione da livello alto a livello basso. Successivamente, con la riga 40, la parola di dati presente in questo istante sulla porta d'utente verrà trasferita e visualizzata.

Con WAIT 37149, 16 viene interrogato lo stato del "flag di interruzione", tramite il componente VIA contenuto nel VIC 20: il valore 16 indica che si tratta del bit 4 (peso uguale a 16).

Le funzioni di pilotaggio e regolazione verranno opportunamente svolte mediante una combinazione di emissione/immissione dei dati. Allo scopo è possibile rilevare una parola di dati attualmente presente nella porta d'utente, elaborarla nel computer e trasferirla nuovamente alla porta d'utente. In figura 9 è listato un opportuno programma.

Come spiegato all'inizio, con la riga 10 vengono predisposti come ingressi, e rispettivamente come uscite, gruppi di 4 canali alla volta. Il comando della riga 20 garantisce che non siano presenti dati nei quattro canali d'uscita (i relativi LED risultano spenti). Successivamente il computer attende che il segnale di riconoscimento arrivi all'ingresso dei dati.

Non appena questo segnale viene rilevato, l'istruzione della riga 50 verifica se nella porta d'ingresso a quattro canali è presente il valore 11. Se non è stato inserito questo numero, il gioco ricomincia dall'inizio.

Se però il canale d'ingresso PB2 viene portato a livello basso 1011 = decimale 11), verrà eseguito il comando POKE della riga 50. Poiché questo comando carica il valore 128 nel registro dati, si accenderà il LED del canale d'uscita PB7 che ha il peso uguale a 128. Il valore d'ingresso 11 ha avuto perciò come conseguenza l'emissione del valore 128. In corrispondenza a tutti gli altri valori d'ingresso (valori ammissibili: da 0 a 16, in quanto solo quattro canali sono disponibili come ingressi) non si hanno reazioni da parte dei canali d'uscita.

I più gravi ostacoli che si opponevano all'utilizzo della porta d'utente dovrebbero ora essere eliminati, cosicché la porta risulta disponibile per i propri esperimenti. I programmi listati nelle figure 10 ed 11 illustrano due esempi di gestione del circuito didattico che indica lo stato della porta.

DISEGNARE COL JOYSTICK

di Alessandro Barattini

Il Commodore 64 ha capacità grafiche molto interessanti e nel modo multicolore si possono avere fino a 4 differenti colori in una matrice di 8X8 Pixel. Sfortunatamente, per poter sfruttare queste caratteristiche, bisogna usare numerose PEEK e POKE che sono molto noiose da utilizzare. Questo programma dà una pratica soluzione al suddetto problema perché permette di disegnare con colori diversi muovendo il cursore sullo schermo con un joystick; in più, le figure create possono essere memorizzate sul disco o sul registratore.

Dopo aver dato il run del programma, il colore del bordo diventa blu e il colore di sfondo diventa nero. I colori possono essere cambiati rispettivamente con i tasti funzione F5 e F7.

Il joystick deve essere messo nella porta 2 per poter muovere il cursore che si trova al centro dello schermo; per disegnare premere il tasto di sparo. Nell'angolo in alto a sinistra viene indicato lo stato di lavoro: il quadratino di sinistra visualizza infatti un 1 o un 2 o un 3 per indicare il modo con cui si sta disegnando, oppure una E per indicare il modo di cancellazione. Il quadratino sulla destra indica il colore con cui il cursore sta disegnando. Fra i due quadratini vengono visualizzati uno o due punti che stanno ad indicare se il cursore disegna a singola o doppia ampiezza.

In una matrice di 8X8 Pixel si può avere un solo colore per ciascun modo di disegno; per ottenere quattro differenti colori (compreso il colore di sfondo) bisogna usare tutti e tre i modi di disegnare, che si selezionano con il tasto funzione F3. Il cursore può disegnare con i 16 colori disponibili, che possono essere selezionati con il tasto funzione F1.

Il cursore, come abbiamo già accennato, può disegnare in modo singolo con un'area di 1X2 Pixel oppure in modo doppio con un'area 2X2 Pixel; il tasto (*) (asterisco) cambia il cursore da singolo a doppio e viceversa.

I numeri da 1 a 9 controllano la velocità di spostamento del cursore: 1 è la velocità più alta, 9 la più bassa. La velocità di spostamento diminuisce quando il cursore disegna o cancella (per trovare la velocità ideale bisogna

fare alcune prove). Il tasto SHIFT + CLR HOME cancella lo schermo.

Il colore di sfondo e del bordo possono essere memorizzati con il tasto SHIFT + (£) (sterlina) e richiamati in qualsiasi momento con il tasto (£) (sterlina). Per avere una visione migliore del disegno fatto premere il tasto (-) (meno) che cancella lo status di lavoro dello schermo; per ripristinarlo premere il tasto (+) (più).

Quando un disegno è finito può essere memorizzato sul disco con il tasto funzione F2 e poi ricaricato in memoria con il tasto funzione F4; se invece si vuole impiegare il registratore si usa il tasto funzione F6 per memorizzare ed il tasto funzione F8 per caricare in memoria. Qualunque sia il mezzo usato, il programma richiede un nome per identificare il disegno; dato il nome premere il tasto RETURN. Per uscire dalle funzioni di memorizzazione e caricamento, prima che queste vengano effettuate, premere il tasto RUN/STOP e si tornerà al normale modo di disegno. Se si desidera rimpiazzare un disegno già esistente mettere il prefisso " 0."

prima del nome del disegno. In tabella 1 sono elencati i comandi disponibili. Il CARICATORE BASIC - Una volta che il programma è stato scritto deve essere memorizzato prima di dare il RUN; infatti il programma si autocancella dopo il RUN. Dopo il RUN il programma carica il linguaggio macchina in alte locazioni della RAM e precisamente fra gli indirizzi C000 e CFFF in una zona che non può essere modificata dal BASIC e sempre presente in memoria. Se si vuole uscire dal programma basta premere RUN/STOP e RESTORE. Naturalmente è possibile usare qualsiasi programma BASIC, ma che ovviamente non effettui delle POKE nella zona di memoria riservata al programma grafico. Un programma BASIC con queste caratteristiche può essere usato insieme col nostro programma grafico. Una precauzione importante è quella di memorizzare il programma BASIC prima di usarlo con il programma grafico perché quest'ultimo può soprascriverlo. Per riattivare il programma grafico fare SYS52992 e premere RETURN. Prima di uscire dal programma memorizzate

| TASTI | COMANDI |
|----------|--|
| F1 | CAMBIA IL COLORE DEL CURSORE |
| F3 | CAMBIA IL MODO DI DISEGNARE (1, 2, 3 o E) |
| F5 | CAMBIA IL COLORE DEL BORDO |
| F7 | CAMBIA IL COLORE DI SFONDO |
| F2 | MEMORIZZA IL DISEGNO SUL DISCO |
| F4 | CARICA IN MEMORIA UN DISEGNO DAL DISCO |
| F6 | MEMORIZZA IL DISEGNO SUL REGISTRATORE |
| F8 | CARICA UN DISEGNO DAL REGISTRATORE |
| 1..9 | CAMBIA LA VELOCITA' DEL CURSORE |
| - | SPEGNE GLI INDICATORI DI STATO DI LAVORO |
| + | ATTIVA GLI INDICATORI DI STATO DI LAVORO |
| CLR HOME | PULISCE LO SCHERMO |
| SHIFT | MEMORIZZA IL COLORE DI SFONDO E DEL BORDO RICHIAMA I COLORI ORIGINALI DI SFONDO E DEL BORDO |
| * | SELEZIONA IL CURSORE SINGOLO O DOPIO |

Tabella 1: Sommario dei tasti comando.

| COPPIE DI BIT | INFORMAZIONI SUL COLORE |
|---------------|--|
| 00 | COLORE DI SFONDO \$D021 |
| 01 | 4 BIT SUPERIORI DELLA MEMORIA DI SCHERMO \$0400-07FF |
| 10 | 4 BIT INFERIORI DELLA MEMORIA DI SCHERMO \$0400-07FF |
| 11 | COLORE DEI 4 BIT \$D800-DBFF |

Tabella 2

qualsiasi figura che avete creato (naturalmente se vi interessa) perché altrimenti viene perduta.

IL TEST SULLA SOMMATORIA - Le linee da 20 a 50 effettuano un test sulla sommatoria dei numeri letti nelle istruzioni DATA e segnalano se vi sono stati errori di trascrizione. Una volta che il programma è stato controllato e risulta corretto, queste linee possono essere cancellate.

MULTICOLOR BIT-MAPPING - Con questo metodo una sezione di 8k di memoria viene visualizzata sullo schermo e nel nostro caso occupa le locazioni da \$2000 a \$3F3F. Due bit (una coppia) controllano un punto sullo schermo e in totale ci sono 32000 punti (160X200). Le combinazioni di coppie di bit determinano dove devono essere lette le informazioni sui colori riguardanti ciascun punto. In due bit sono possibili 4 combinazioni: entrambi i bit disattivati (00), entrambi i bit attivati (11) o un bit attivato e uno no (01 o 10) (vedi tabella 2).

LA SUBROUTINE CHE ATTIVA I PIXEL - Il cuore di questo programma è una subroutine che si trova agli indirizzi \$C000 e \$C130; essa può essere trasferita in programmi che necessitano di operare in alta risoluzione. Prima però di usarla bisogna chiamare la routine di inizializzazione che parte all'indirizzo \$C0E7; quest'ultima attiva un pixel o una coppia di pixel con coordinate X e Y.

Il range delle X varia fra 0 e 319 (orizzontale) e il range delle Y fra 0 e 199 (verticale); la posizione 0,0 è l'angolo sinistro in alto sullo schermo. Prima di usare la routine le coordinate della X vengono memorizzate alle locazioni \$0340 e \$0341, e le coordinate della Y alla locazione \$0342. Per la X occorrono due byte perché il range supera il numero 255, massimo memorizzabile in un singolo byte.

COME LAVORA IL PROGRAMMA - Per attivare un pixel è necessario conoscere il byte corretto della sezione di 8K; di questo byte bisogna poi determinare quale degli 8 bit deve esse-

| | |
|-------------|---|
| \$00FD-00FE | Byte in cui e' attivato il bit |
| \$0022-0023 | Temp. mem |
| \$0024 | Velocita' del cursore |
| \$0025 | Nastro o disco |
| \$0026 | Cursore singolo o doppio |
| \$0027 | Temp. mem |
| \$003F-0041 | Temp. mem |
| \$00A5 | Modo di disegno |
| \$00A6-00A7 | Colore del byte |
| \$00A8 | Colore del cursore |
| \$033C-033F | Temp. mem |
| \$0340-0341 | Coordinata X |
| \$0342 | Coordinata Y |
| \$CFFD-CFFF | Temp. mem |
| \$2000-3F3F | Area mappa bit |
| \$3F45-3F46 | Area di mem. del colore dello sfondo e del bordo |
| \$4000-43FF | Area di mem. mappa colore |
| \$4400-47FF | Area di mem. mappa schermo |
| \$C000-C0E6 | Routine Principale |
| \$C0E7-C130 | Inizializza mappa bit |
| \$C150-C1D3 | Muove il cursore |
| \$C300-C34C | Disegna |
| \$C400-C4FC | Legge i tasti comando |
| \$C500-C584 | Mette i byte nella mappa colore |
| \$C600-C676 | Visualizza il modo di disegnare |
| \$C700-C742 | Legge i tasti Per doppio o singolo cursore |
| \$C900-C96F | LOAD |
| \$CA00-CA7A | SAVE |
| \$CB00-CB6E | Inizializzazione |
| \$CC00-CC4F | Dati Per il messaggio Per SAVE e LOAD |
| \$CC50-CCCF | Dati Per lo sSprite (il segnalatore di stato) |
| \$CD00-CD8B | Nome Per SAVE e LOAD |
| \$CE00-CE37 | Trasferisce la memoria all'area di memorizzazione |
| \$CE50-CE87 | Trasferisce la memoria dall'area di memorizzazione |
| \$CF00-CF17 | Loop di chiamata delle subroutine |
| \$CFA0-CFAB | Trasferisce i dati dello sSprite al buffer del registratore |

Tabella 3: Locazioni di memoria usate.

re attivato. La formula che determina il byte ed il bit ci viene data dal manuale:

$$\text{BYTE} = \text{base} + (\text{INT}(\text{Y}/8) * 320) + (\text{INT}(\text{X}/8) * 8) + (\text{Y AND } 7) \\ \text{BIT} = 7 - (\text{X AND } 7)$$

Nel caso nostro base è uguale a 8192. Qui la formula è stata semplificata in:

$\text{BYTE} = \text{base} + ((\text{Y AND } 248) * 40) + (\text{X AND } 248) + (\text{Y AND } 7)$ tradotta e trasferita nel codice macchina localizzato fra le locazioni \$C065 e \$C0C0. Ma invece di attivare il bit appropriato con la formula predetta, viene usato un indice di una tabella di predefiniti valori maschera (una maschera è una serie di bit che isola uno o più bit da un gruppo di bit). Se si vogliono variare i valori della maschera e attivare bit singolo o cop-

pie per lo schermo multicolore, basta levare l'indice dal suo posto in tabella. LE ROUTINE DI SAVE E LOAD DEL KERNAL - I dati della figura sono memorizzati in un blocco continuo di memoria fra gli indirizzi \$2000 e \$47FF. Prima di una memorizzazione i dati riguardanti il colore per la coppia di bit 11 (memoria colore) sono trasferiti alle locazioni \$4000 - \$43FF. I dati riguardanti il colore per le coppie di bit 01 e 10 (memoria di schermo) sono trasferiti alle locazioni \$4400 - \$47FF. Quando si carica una figura i dati sono ritrasferiti automaticamente alle proprie locazioni. Per fare le operazioni di SAVE e LOAD vengono usate le subroutine del Kernal: SETLFS, SETNAM, Save e Load.

```
10 Print"Maspetta sto caricando i
  1 codice macchina."
15 rem testa la sommatoria
20 P=0:s=206373:for k= 0 to 1809:
  read a:P=P+a:next
30 if s=P then Print"tutto o.k.":
  goto50
```

```
40 if s<P then Print"errore nei
  data":end
50 restore
95 rem routine Principale
100 for k= 49152 to 49456:read b:P
  oke k,b:next
101 data 101,0,32,16,8,4,2,1,192,1
```

```

92,48,48
102 data 12,12,3,3,128,128,32,32,8
,8,2,2
103 data 64,64,16,16,4,4,1,1,63,63
,207,207
104 data 243,243,252,252,169,8,133
,34,56,176,29,169
105 data 16,133,34,56,176,22,169,2
4,133,34,56,176
106 data 15,169,32,133,34,56,176,8
,169,0,133,34
107 data 56,176,1,96,173,65,3,201,
2,176,248,201
108 data 1,208,7,173,64,3,201,64,1
76,237,173,66
109 data 3,201,200,176,230,169,0,1
41,63,3,133,254
110 data 173,64,3,41,248,141,60,3,
173,66,3,41
111 data 7,24,109,60,3,141,60,3,17
3,65,3,105
112 data 0,141,61,3,173,66,3,41,24
0,133,253,160
113 data 5,10,46,63,3,136,208,249,
141,62,3,165
114 data 253,160,3,10,38,254,136,2
08,250,24,109,62
115 data 3,133,253,165,254,109,63,
3,133,254,24,165
116 data 253,109,60,3,133,253,165,
254,109,61,3,24
117 data 105,32,56,133,254,173,64,
3,41,7,24,101
118 data 34,170,189,0,192,133,35,1
60,0,177,253,166
119 data 34,224,32,240,5,5,35,145,
253,96,37,35
120 data 145,253,96,162,0,169,38,1
57,0,4,157,0
121 data 5,157,0,6,157,0,7,169,1,1
57,0,216
122 data 157,0,217,157,0,218,157,2
33,218,232,208,225
123 data 169,63,133,167,169,0,133,
166,168,145,166,136
124 data 192,0,208,249,198,167,166
,167,224,31,208,241
125 data 173,17,208,9,32,141,17,20
8,173,24,208,9
126 data 8,141,24,208,96
195 rem routine sPosta cursore
200 for k= 49488 to 49619:read b:P
oke k,b:next
210 data 230,63,165,63,197,36,208,
123,169,0,133,63
211 data 173,0,220,41,1,208,10,173
,1,208,201,40
212 data 240,3,206,1,208,173,0,220
,41,2,208,10
213 data 173,1,208,201,239,240,3,2
38,1,208,173,0
214 data 220,41,8,208,32,173,16,20

```

```

8,201,1,208,7
215 data 173,0,208,201,62,240,18,2
38,0,208,238,0
216 data 208,173,0,208,201,0,208,5
,169,1,141,16
217 data 208,173,0,220,41,4,208,20
,173,16,208,201
218 data 0,208,7,173,0,208,201,0,2
40,6,206,0
219 data 208,206,0,208,173,16,208,
201,1,208,12,173
220 data 0,208,201,254,208,5,169,0
,141,16,208,96
295 rem routine Per disegnarne
300 for k= 49920 to 49996:read b:P
oke k,b:next
310 data 173,16,208,141,65,3,173,0
,208,141,64,3
311 data 173,1,208,56,233,40,141,6
6,3,173,0,220
312 data 41,16,208,48,165,165,201,
0,208,9,32,61
313 data 192,206,66,3,32,61,192,16
5,165,201,1,208
314 data 3,32,40,192,165,165,201,2
,208,6,32,61
315 data 192,32,47,192,165,165,201
,3,208,6,32,61
316 data 192,32,54,192,96
395 rem routine Per leggere i com
andi e Per Preparare la mappa
colore
400 for k= 50176 to 50564:read b:P
oke k,b:next
410 data 169,8,133,37,32,159,255,3
2,228,255,201,133
411 data 208,12,230,168,166,168,22
4,16,208,4,162,0
412 data 134,168,166,168,142,37,20
8,201,134,208,12,198
413 data 165,166,165,224,255,208,4
,162,3,134,165,201
414 data 135,208,15,238,32,208,174
,32,208,224,16,208
415 data 5,162,0,142,32,208,201,13
6,208,15,238,33
416 data 208,174,33,208,224,16,208
,5,162,0,142,33
417 data 208,201,49,208,4,162,5,13
4,36,201,50,208
418 data 4,162,10,134,36,201,51,20
8,4,162,16,134
419 data 36,201,52,208,4,162,21,13
4,36,201,53,208
420 data 4,162,32,134,36,201,54,20
8,4,162,48,134
421 data 36,201,55,208,4,162,64,13
4,36,201,56,208
422 data 4,162,112,134,36,201,57,2
08,4,162,255,134
423 data 36,201,147,208,3,32,8,193
,201,137,208,3

```

```

424 data 32,0,202,201,138,208,3,32
,0,201,201,45
425 data 208,5,162,0,142,21,208,20
1,43,208,5,162
426 data 3,142,21,208,201,139,208,
7,162,1,134,37
427 data 32,0,202,201,140,208,7,16
2,1,134,37,32
428 data 0,201,201,169,208,12,174,
33,208,142,254,207
429 data 174,32,208,142,255,207,20
1,92,208,12,174,254
430 data 207,142,33,208,174,255,20
7,142,32,208,133,39
431 data 96,2,0,0,173,0,220,41,16,
208,125,32
432 data 21,195,165,254,41,7,170,1
65,254,74,74,74
433 data 133,167,165,253,74,74,74,
133,166,224,0,240
434 data 15,202,24,165,166,105,32,
133,166,144,242,230
435 data 167,56,176,237,160,0,165,
165,201,3,208,16
436 data 177,166,41,15,106,106,106
,106,5,168,42,42
437 data 42,42,145,166,165,165,201
,2,208,8,177,166
438 data 41,240,5,168,145,166,165,
165,201,1,208,40
439 data 24,165,167,105,212,133,16
7,165,253,41,1,176
440 data 11,177,166,41,240,5,168,1
45,166,56,176,16
441 data 177,166,41,15,106,106,106
,106,5,168,42,42
442 data 42,42,145,166,96
495 rem routine che abilita lo sch
ermo Per disegnare
500 for k= 50698 to 50806:read b:p
oke k,b:next
510 data 165,165,201,3,208,25,169,
171,141,201,3,141
511 data 204,3,141,210,3,141,213,3
,169,191,141,216
512 data 3,141,198,3,141,207,3,201
,2,208,27,169
513 data 191,141,198,3,141,207,3,1
41,216,3,169,171
514 data 141,201,3,141,204,3,169,1
86,141,210,3,141
515 data 213,3,201,1,208,23,169,17
1,141,198,3,141
516 data 201,3,141,204,3,141,207,3
,141,210,3,141
517 data 213,3,141,216,3,201,0,208
,25,169,191,141
518 data 198,3,141,207,3,141,216,3
,169,186,141,201
519 data 3,141,210,3,141,204,3,141
,213,3,96
595 rem routine cursore doppio o s

```

```

ingolo
600 for k= 50944 to 51010:read b:p
oke k,b:next
610 data 165,39,201,42,208,15,166,
38,240,7,162,0
611 data 134,38,56,176,4,162,1,134
,38,166,38,224
612 data 1,208,25,206,66,3,32,0,19
7,162,140,142
613 data 211,3,142,214,3,162,136,1
42,217,3,142,220
614 data 3,56,176,14,162,128,142,2
11,3,142,214,3
615 data 142,217,3,142,220,3,96
695 rem routine load
700 for k= 51456 to 51567:read b:p
oke k,b:next
710 data 32,0,206,165,165,141,253,
207,169,27,141,17
711 data 208,169,21,141,24,208,173
,33,208,141,69,63
712 data 173,32,208,141,70,63,169,
0,141,21,208,169
713 data 6,141,33,208,160,0,185,0,
204,32,210,255
714 data 200,192,40,208,245,165,37
,201,1,208,13,160
715 data 0,185,40,204,32,210,255,2
00,192,20,208,245
716 data 32,0,205,169,0,32,213,255
,173,69,63,141
717 data 33,208,173,70,63,141,32,2
08,169,3,141,21
718 data 208,32,32,193,32,80,206,3
2,160,207,173,253
719 data 207,133,165,96
795 rem routine save
800 for k= 51712 to 51834:read b:p
oke k,b:next
810 data 32,0,206,165,165,141,253,
207,169,27,141,17
811 data 208,169,21,141,24,208,173
,33,208,141,69,63
812 data 173,32,208,141,70,63,169,
0,141,21,208,169
813 data 6,141,33,208,160,0,185,0,
204,32,210,255
814 data 200,192,20,208,245,165,37
,201,1,208,13,160
815 data 0,185,40,204,32,210,255,2
00,192,20,208,245
816 data 32,0,205,169,17,32,210,25
5,169,0,133,61
817 data 169,32,133,62,162,0,160,7
2,169,61,32,216
818 data 255,173,69,63,141,33,208,
169,3,141,21,208
819 data 32,32,193,32,80,206,32,16
0,207,173,253,207
820 data 133,165,96
895 rem inizializzazione
900 for k= 51968 to 52078:read b:p

```

```

oke k,b:next
910 data 32,231,192,169,3,141,21,2
    08,169,1,141,39
911 data 208,169,160,141,0,208,141
    ,1,208,169,0,141
912 data 33,208,169,1,141,29,208,1
    73,22,208,9,16
913 data 141,22,208,169,3,133,165,
    169,3,141,28,208
914 data 169,32,133,36,141,2,208,1
    69,58,141,3,208
915 data 169,6,133,167,169,108,133
    ,166,169,38,133,170
916 data 169,1,133,171,169,1,133,1
    68,169,1,141,40
917 data 208,169,0,133,38,141,37,2
    08,169,6,141,32
918 data 208,32,160,207,169,14,141
    ,248,7,169,15,141
919 data 249,7,96
995 rem messa9910 Per save e load
1000 for k= 52224 to 52431:Poke k,0
    :next
1010 for k= 52224 to 52283:read a:P
    oke k,a:next
1011 data 147,5,17,17,18,83,65,86,6
    9,32,83,85
1012 data 32,68,73,83,67,79,146,58,
    32,13,145,18
1013 data 76,79,65,68,32,68,65,29,2
    9,29,29,29
1014 data 29,29,29,146,19,17,17,18,
    29,29,29,29
1015 data 29,29,29,29,78,65,83,84,8
    2,79,29,146
1020 Poke 52329,12:Poke 52332,8:Pok
    e 52338,8:Poke 52341,12
1095 rem dati Per lo sSprite
1100 for k= 52371 to 52397:read a:P
    oke k,a:next
1110 data 170,140,85,191,140,85,171
    ,136,85,171,136,85
1111 data 191,128,85,171,128,85,171
    ,128,85,191,128,85
1112 data 170,128,85
1195 rem routine Per il nome in sav
    e e load
1200 for k= 52480 to 52619:read a:P
    oke k,a:next
1210 data 162,0,160,0,232,224,255,2
    08,251,200,192,16
1211 data 208,246,32,159,255,32,228
    ,255,201,32,144,7
1212 data 201,96,176,3,32,210,255,2
    01,20,208,3,32
1213 data 210,255,166,211,224,31,20
    8,4,160,30,132,211
1214 data 224,14,208,4,160,15,132,2
    11,201,13,208,3
1215 data 56,176,3,56,176,190,169,1
    ,166,37,160,1
1216 data 32,186,255,162,112,160,25

```

```

    4,202,200,189,0,4
1217 data 201,32,240,247,132,65,169
    ,16,56,229,65,168
1218 data 162,0,24,189,94,4,201,47,
    176,10,201,32
1219 data 176,6,24,105,64,157,94,4,
    232,224,17,208
1220 data 233,152,162,95,160,4,32,1
    89,255,169,13,32
1221 data 210,255,169,192,32,144,25
    5,96
1295 rem routine Per trasferire la
    memoria all'area da memorizzar
    e
1300 for k= 52736 to 52791:read a:P
    oke k,a:next
1310 data 162,0,189,0,216,157,0,64,
    189,0,217,157
1311 data 0,65,189,0,218,157,0,66,1
    89,0,219,157
1312 data 0,67,189,0,4,157,0,68,189
    ,0,5,157
1313 data 0,69,189,0,6,157,0,70,189
    ,0,7,157
1314 data 0,71,232,224,0,208,203,96
1395 rem routine Per trasferire la
    memoria dall'area di memorizza
    zione
1400 for k= 52816 to 52871:read a:P
    oke k,a:next
1410 data 162,0,189,0,64,157,0,216,
    189,0,65,157
1411 data 0,217,189,0,66,157,0,218,
    189,0,67,157
1412 data 0,219,189,0,68,157,0,4,18
    9,0,69,157
1413 data 0,5,189,0,70,157,0,6,189,
    0,71,157
1414 data 0,7,232,224,0,208,203,96
1495 rem loop di chiamata delle sub
    routine
1500 for k= 52992 to 53015:read a:P
    oke k,a:next
1510 data 32,0,203,32,80,193,32,0,1
    95,32,0,196
1511 data 32,0,197,32,0,198,32,0,19
    9,76,3,207
1595 rem routine Per trasferire i d
    ati dello sSprite al buffer del
    registratore
1600 for k= 53152 to 53163:read a:
    Poke k,a:next
1610 data 160,128,185,79,204,153,12
    7,3,136,208,247,96
1620 Poke 2049,0:Poke 2050,0:sys 52
    992

```

OFFERTISSIMA -- EXELCO --



SOLO METÀ PREZZO

La divis. EXELCO vi propone
una vantaggiosissima
combinazione **COMMODORE**

| | |
|---|-----------------------|
| n. 1 COMMODORE C 64 | L. 699.500 |
| n. 1 Registratore per C 64 | L. 82.000 |
| n. 1 Libro "C64 EXPOSED" | L. 24.000 |
| n. 1 Libro "Impariamo il computer" | L. 25.000 |
| n. 1 Cassetta Software "Garden Wars" | L. 20.000 |
| n. 1 Libro "Svegliamo il Commodore 64" | L. 9.900 |
| n. 1 Libro "Il libro dei giochi per C64" | |
| n. 1 Libro "Il Commodore C64 al lavoro" | |
| Totale | L. 860.400 |

| Descrizione | Q.tà | Prezzo unitario | Prezzo Totale |
|--|------|-------------------|---------------|
| n. 1 COMMODORE C64 | | | |
| n. 1 Registratore per C64 | | | |
| n. 1 Libro "C64 EXPOSED" | | | |
| n. 1 Libro "Impariamo il computer" | | | |
| n. 1 Cassetta Software "Garden Wars" | | | |
| n. 1 Libro "Svegliamo il C64" | | | |
| n. 1 Libro "Il libro dei giochi per C64" | | | |
| n. 1 Libro "Il Commodore C64 al lavoro" | | | |
| | | L. 425.000 | |

Desidero ricevere il materiale indicato nella tabella, a mezzo pacco postale contro assegno, al seguente indirizzo:

Nome

Cognome

Via

Città

Data C.A.P.

SPAZIO RISERVATO ALLE AZIENDE - SI RICHIEDE L'EMISSIONE DI FATTURA
Partita I.V.A.

PAGAMENTO:

- A) Anticipato, mediante assegno circolare o vaglia postale per l'importo totale dell'ordinazione.
B) Contro assegno, in questo caso, è indispensabile versare l'acconto di Lire 50.000 mediante assegno circolare o vaglia postale. Il saldo sarà regolato contro assegno.
AGGIUNGERE: L. 5.000 per contributo fisso. I prezzi sono comprensivi di I.V.A.

SOLO METÀ PREZZO

EXELCO

Via G. Verdi, 23/25
20095 - CUSANO MILANINO - Milano

NON PERDETE QUESTA OCCASIONE!

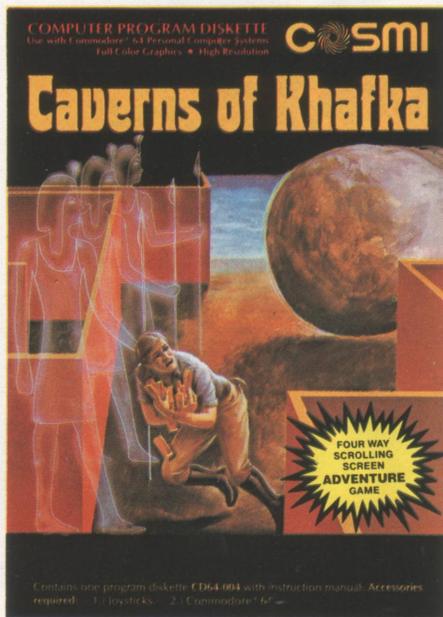
CAVERNS OF KHAFKA

Casa produttrice
AUDIOGENIC

Configurazione
COMMODORE 64

Supporto
DISCO

Prezzo
L. 27.000



Ecco un'affascinante avventura che vi terrà ore e ore attaccati al vostro video nell'intento di trovare a portare a casa il tesoro reale che si trova all'interno della tomba del faraone Khafka.

Attenzione però, non è così facile come sembra, in quanto i pericoli sono innumerevoli e ti saranno d'intralcio durante tutta l'avventura.

Al principio ricevi cinque omini o "lives" (vite) con i quali dovrai cercare nelle profonde viscere della terra i cinque Sacri Sigilli che verranno da te posti uno per volta sulla porta della tomba del faraone e, fatto questo potrai così recuperare il tesoro nascosto che da secoli aspetta di essere svelato.

I Sacri Sigilli sono nascosti (a caso naturalmente) nell'immensa caverna e ogni volta che ci si avventura in essa il luogo in cui vengono nascosti i sigilli varia, in modo che non si possa trovare il cosiddetto "giro perfetto" che renderebbe così una volta trovato, il gioco monotono e quindi non più divertente.

La tomba del faraone è posta alla base dell'elevatore e i "calibri sagomati" dentro i quali verranno posti i Sacri Sigilli sono tra le due statue del Dio RA.

Una volta aperta la porta si raggiunge la Grande Galleria che porta alla Sala del Re dove (guarda il caso) si trova il prezioso tesoro.

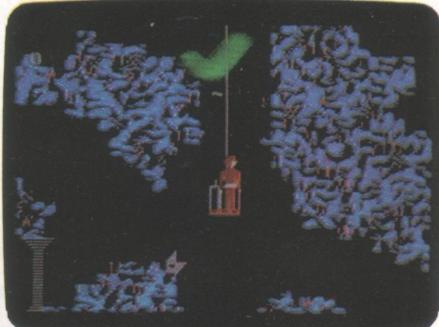
I pericoli si fanno sempre più minacciosi man mano che ci si avvicina al tesoro ed essi vengono procurati da sinistri animali, quali i pippistrelli, gli scarabei, oppure da cause naturali come massi cadenti e rotolanti, pozze di acido e magma vulcanico.

Tu perderai la vita ogni volta che verrai colpito da una di queste minacce o quando cadrai per due volte consecutive nel tentativo di arrampicarti sulla corda che possiedi o nel tentativo di scalare montagne sotterranee troppo ripide.

Dopo tante disgrazie ecco una buona notizia: ogni volta che entrerai nella sala del re recupererai cinque vite e potrai così quindi rimanere in gioco più del previsto.



Per completare l'avventura nel migliore dei modi, oltre che stare molto attenti alle minacce che incombono sul proprio capo, bisogna essere veramente dei maghi in fatto di "joystick", elemento indispensabile in questo gioco. Esso servirà infatti a muovere l'elevatore in basso ed in alto (anche nel caso in cui non ci si trova sopra), a camminare o correre ai lati, a lanciare la corda su cui arrampicarti, ed infine a camminare carponi in presenza di pericolo.



In generale possiamo dire che "caverns of khafka" è una delle migliori avventure oggi in commercio per il Commodore 64 sia per quanto riguarda la grafica e sia per quanto riguarda la versatilità nei movimenti.

Il programma viene venduto su disco o su cassetta a vostra scelta e il joystick come detto sopra è indispensabile. Forza quindi.

H.E.R.O.

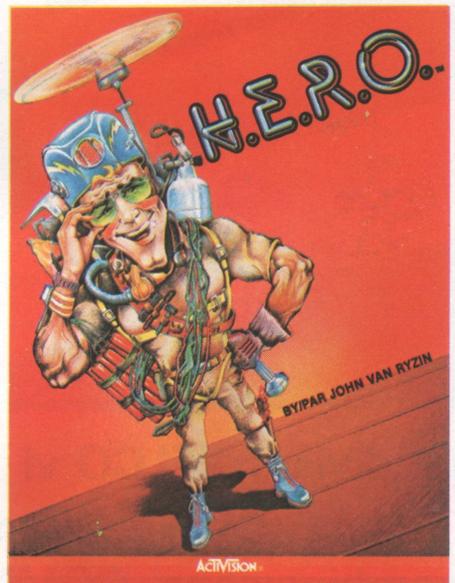
Casa produttrice
ACTIVISION

Autore
JOHN VAN RYZIN

Configurazione
COMMODORE 64

Supporto
CASSETTA

Prezzo
L. 32.000



Un'eruzione vulcanica ha imprigionato i minatori nelle caverne di Monte Leone.

Il nostro eroe, Roderick Hero avrà bisogno della nostra assistenza e di tutte le attrezzature speciali che gli metteremo a disposizione, per salvare tutti i minatori.

All'inizio del gioco si hanno a disposizione cinque vite, una in azione e quattro di riserva. Ogni ventimila punti si vince una vita supplementare fino ad un massimo di sei.

Esiste un indicatore di energia che si riempie completamente all'inizio di un nuovo schermo. Il livello energetico comincia a diminuire non appena si muove la leva del joystick; se resti a corto di energia ma ti rimane tuttavia ancora una vita, si riceverà una nuova riserva all'inizio del livello successivo.

Hero possiede un cappello con un propulsore ad elica incorporato, il quale consente di volare in qualunque direzione.

Per volare in alto, bisogna spingere la leva in avanti; per correre o volare a destra

o a sinistra, la leva va invece spinta nella direzione corrispondente; per volare in una miniera, è invece sufficiente saltare dalla sporgenza e lasciarsi trascinare dalla forza di gravità.

L'elmetto è provvisto di un raggio microlaser che consente di distruggere gli animali delle caverne.

Il raggio può essere usato anche per perforare le pareti delle miniere, ma questo richiede molto tempo e soprattutto spreca tantissima energia.

Per sparare, è sufficiente premere il solito pulsante del joystick, mentre le raffiche sono ottenibili mantenendo premuto quest'ultimo.

All'inizio del gioco, e prima di entrare in ogni miniera si dispone di sei candelotti di dinamite, ognuno dei quali è in grado di demolire una piccola parete.

Per fare ciò è necessario scendere a terra, quindi appoggiare il candelotto contro la parete da abbattere spingendo la leva di comando all'indietro, ed infine scappare via il più velocemente possibile in modo tale da evitare le conseguenze dell'esplosione.

Una zattera resistente al calore consente di galleggiare sui fiumi di lava; attenzione però perché non è per niente facile scendervi.

Le miniere si estendono verticalmente ed orizzontalmente all'interno di Monte Leone. Alcune di esse sbucano in un vicolo cieco, mentre altre sono bloccate da pareti.

Le pareti e le superfici luminose rosse contengono depositi di lava vulcanica incandescente; entrare in collisione con queste ultime provoca la perdita di una vita.

Le miniere sono illuminate da numerose lanterne che bisogna evitare di toccare se non si vuole oscurare completamente il percorso, e restare così in completa balia delle animali assassini.



Le miniere brulicano infatti di pericolose bestiacce: ragni, pipistrelli, farfalle notturne e serpenti. È necessario cercare di evitarli, oppure distruggerli con il microlaser a disposizione.

Ogni movimento lungo il fiume di lava viene seguito da un tentacolo indistruttibile il cui contatto provoca la perdita di una

delle vite.

Per passare ad un livello superiore è indispensabile trovare e portare in salvo almeno un minatore. Più in basso si scende, più rapido diventa il gioco, più pericolosi e numerosi gli ostacoli da superare, e soprattutto più difficile ed estenuante la ricerca del minatore intrappolato.

Si hanno a disposizione cinque difficoltà di gioco, ognuna delle quali prende inizio ad un preciso livello:

| Gioco | Livello iniziale |
|-------|------------------|
| 1 | 1 |
| 2 | 5 |
| 3 | 9 |
| 4 | 13 |
| 5 | 17 |

Il gioco cinque si differenzia dagli altri per la casualità dei livelli successivi al diciassettesimo iniziale.



I punti vengono così distribuiti:

Animale colpito: 50 punti
Candelotti di dinamite risparmiati: 50 punti
Distruzione di una parete: 75 punti
Salvataggio di un minatore: 1000 punti.

In aggiunta, viene addizionato un punteggio bonus proporzionale all'energia risparmiata al momento del salvataggio del minatore intrappolato.

Per i giocatori più esperti, consigliamo di osservare attentamente la dimostrazione iniziale del gioco, in modo da familiarizzarsi con il terreno ed i pericoli da affrontare.

Qualora Hero restasse a corto di dinamite, può comunque utilizzare il raggio microlaser per fare un foro nella parete; più sta vicino ad essa, più elevata è la velocità di perforazione. Se una lanterna si spegne, l'unica tecnica di orientamento possibile si basa nel cercare le tipiche irregolarità nella parte bassa dello schermo.

In conclusione, un altro cavallo di battaglia Activation, con grafica stupenda ed effetti sonori sufficienti; peccato che manchi una musicchetta di sottofondo, altrimenti sarebbe davvero qualcosa di eccezionale.

SPACE RESCUE

Casa produttrice
HES

Configurazione
COMMODORE 64

Supporto
FLOPPY DISK

Prezzo
L. 24.000

Space Rescue™

Welcome aboard the Star Wolf! You and Benji must save the universe from imminent invasion by rescuing kidnapped scientists within our solar system. Space Rescue can be successfully completed with sharp reflexes and a quick mind.

Joystick optional
Disk drive required
Eight unique levels of play
Ages 8 and over

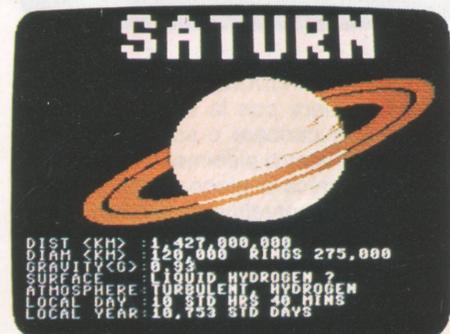


Created by
EA

Diskette for
Commodore 64

Il cagnolino Benjamin, amico di tante avventure trasmesse circa due anni fa in TV, questa volta si ripropone sul nostro video in versione spaziale, grazie al videogioco SPACE RESCUE della Human Engineered Software (HES).

Scopo del gioco è aiutare Benji a trovare alcuni scienziati catturati dalle forze aliene e imprigionati sui vari pianeti del



sistema solare.

In questo gioco i livelli di difficoltà sono otto e vanno dall' "Ufficiale Pilota" al "Maresciallo di flotta", permettendo così ai più giovani ed inesperti di poter concludere felicemente insieme a Benji il salvataggio



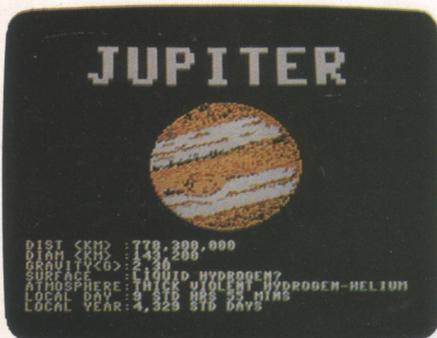
del gruppo di scienziati a discapito delle creature provenienti da altri universi.

Per concludere l'avventura vi è un limite di tempo ovviamente proporzionale ai livelli di difficoltà.

Così, se nel livello "Ufficiale Pilota" vi è un limite di 60 giorni stellari, nel "Maresciallo di flotta" il limite è di soli 10 giorni stellari.

La vostra astronave da cui emerge la testa di Benji è equipaggiata nel migliore dei modi con gli strumenti indispensabili.

Come vedete nella foto, in basso a sinistra del video vi sono otto lettere dell'alfabeto, di cui le prime tre indicano il reparto con cui si vuole comunicare, e sono: Transport room (Sala trasporto) - Engine room (Sala motori) - Weapons room (Sala armamenti).



Portando quindi il cursore sulla lettera T si comunicherà con la sala trasporto la quale potrà caricare o scaricare dall'astronave, a vostro piacimento e secondo la vostra tecnica di gioco, carburante, cibo, armi ed eventualmente scienziati liberati, nel caso ce ne fossero.

Comunicando con la sala motori invece, potrete cambiare la vostra destinazione sui vari pianeti, tenendo conto del carburante e della potenza dei motori.

Infine con la sala armamenti potrete cambiare il tipo di difesa e il tipo di armi da usare contro le astronavi aliene durante l'operazione salvataggio.

È bene si sappia che questo gioco vi sarà anche didatticamente utile in quanto quando lo vorrete saranno riportate sullo schermo le caratteristiche di tutti i pianeti del sistema solare con i vari parametri reali riguardanti la distanza dalla terra, il loro diametro, l'atmosfera e la durata del giorno e dell'anno.

Prima di concludere vi darò qualche consiglio e truccetto.

Gli scienziati sono nascosti solo dentro delle cupole che si trovano sulla superficie dei vari pianeti, e quindi non rastrellate inutilmente l'intera superficie; inoltre, al primo livello il 95% delle volte tutti e sei gli scienziati sono nascosti su Plutone!

Nei livelli più alti comunque sarà bene tornare ogni tanto sulla terra per effettuare eventuali riparazioni o per rifornirsi di cibo e carburante.

Il gioco viene caricato da floppy disk ed è completo di manuale molto chiaro ma purtroppo in inglese.

Il joystick pur non essendo indispensabile è molto utile.

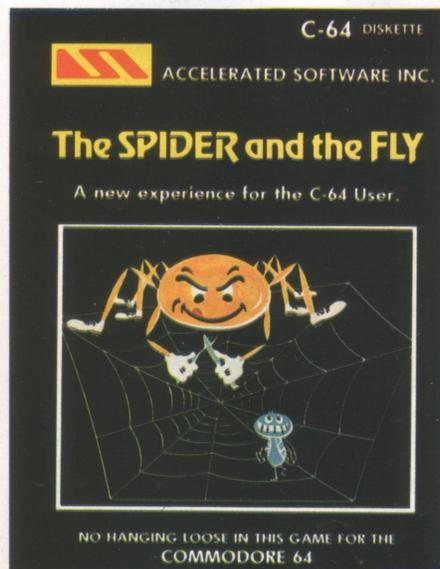
THE SPIDER AND THE FLY

Casa produttrice
ACCELERATED SOFTWARE INC.

Configurazione
COMMODORE 64

Supporto
DISCO

Prezzo
NON IMPORTATO



Il ragno e la mosca è il titolo di questo gioco su dischetto della "Accelerated Software Inc." che si carica con LOAD "ASI", 8 o LOAD "*", 8 e RUN; durante il caricamento vi è una schermata che presenta il gioco con titolo, autori e casa produttrice. Dopo alcuni secondi, schiacciando il "fire" del joystick piazzato nella "control port" numero due, si può scegliere prima il numero dei giocatori (da uno a quattro) e poi il livello di difficoltà da cui partire (nove in tutto). Quindi ha inizio il gioco il cui protagonista è un ragno che deve cercare di intrappolare un determi-



nato numero di mosche e farfalle tessendo la propria ragnatela.

Al primo livello le mosche e le farfalle da catturare sono trenta, ed esse appaiono poco alla volta sul campo di battaglia e si muovono leggermente più lentamente di quanto si muove il ragno; per riuscire a far suoi gli insetti deve velocemente circondarli, poiché la ragnatela dopo alcuni secondi comincia a disfarsi.

Lungo il percorso ci sono poi diversi ostacoli che rendono duro e pericoloso il lavoro del ragno, essi sono una mano e delle scatole di insetticida: la mano con l'indice in fuori si aggira casualmente e lentamente in lungo e largo per lo schermo con aria del tutto innocua, ma se per caso tocca la ragnatela che il ragno sta tessendo lo fa cadere in terra e deve ricominciare da capo o se, ancora peggio, tocca il ragno stesso, allora perdi una delle tre vite che hai a disposizione. Il secondo pericolo è costituito da scatole di insetticida che improvvisamente compaiono qua e là e che se vengono urtate fanno perdere un ragno. Catturati i primi trenta insetti si passa al secondo livello e questa volta le mosche e le farfalle da catturare sono quaranta e si muovono un po' più velocemente, e le scatole di insetticida sono più numerose; e così si va avanti fino al nono livello, con sempre più insetti e insetticida.

Nel complesso un gioco non troppo originale con una grafica e una struttura semplici ma efficaci, per venire al capo del quale non è necessaria una particolare tattica ma piuttosto una certa abilità e tanta pazienza, il che lo rende particolarmente adatto ai più piccoli.

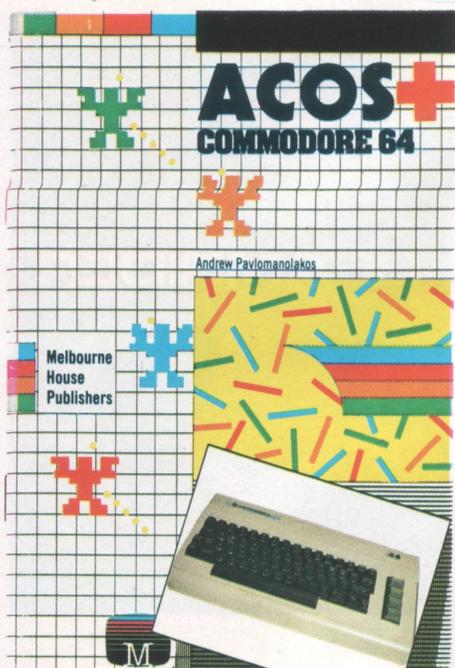
ACOS +

Casa produttrice
MELBOURNE HOUSE

Configurazione
COMMODORE 64

Supporto
CASSETTA

Prezzo
NON IMPORTATO



È stata una notevole sorpresa dover recensire per questa nostra rivista il programma Acos +, proprio perché ho avuto la possibilità di venire in contatto con un prodotto che si distingue per la sua originalità in un settore, quello dei programmi di utilità, che, normalmente, non è molto prodigo di fantasia.

Il package comprende tre programmi: Acos +, Format e Zorn.

Il primo è il programma vero e proprio; si tratta di un "tool" che aggiunge ben 36 comandi al BASIC del Commodore, in maniera specifica per la gestione della cassetta.

È proprio in questa direzione, che è indirizzato l'Acos +: permettere al gestore di un Datasette (il registratore del Commodore) di poter fruire dei "servizi" che normalmente sono riservati a chi possiede un drive.

L'unica differenza che si riscontra nella

gestione così organizzata della cassetta, oltre a una questione di tempo, consiste negli spostamenti in avanti ed indietro del nastro che sono comandati a mano dietro istruzione specifica sullo schermo. Viene cioè invitato l'utente, nel momento e nei modi scelti dal programma, a premere i tasti FWD, REWIND, PLAY, RECORD oppure STOP.

Praticamente il programma opera tramite dei LOAD e dei SAVE che vengono studiati dall'Acos + in maniera da collocare sulla cassetta i vari files in modo ordinato senza il rischio di cancellature od errori. Questo viene attuato mediante una DIRECTORY che, in fase di formattazione, viene creata all'inizio della cassetta e che viene costantemente aggiornata coi nomi dei programmi e la loro posizione relativa sul nastro.

Il programma FORMAT è proprio indicato a questa funzione: misurare il nastro, prepararne una DIRECTORY in testa, e prepararne i riferimenti relativi.

Lo scopo degli autori dovrebbe essere quello di facilitare le operazioni soprattutto ai neofiti, ma, a nostro parere, esso è forse più indicato per quelli che hanno già un po' di pratica con registratori ed affini, perché la serie di operazioni da svolgere nella preparazione della cassetta non è del tutto semplice e immediata.

Non è da sottovalutare un'altra possibilità che viene offerta al programmatore, quella cioè di aggiungere una serie non indifferente di comandi al BASIC, tale da potere evitare di dover utilizzare quelle POKE e PEEK che sono tanto ostiche ai programmatori non troppo esperti.

I risultati di questo maquillage si possono subito riscontrare caricando e facendo girare il programma dimostrativo ZORN che è inserito subito dopo al FORMAT.

Si tratta dell'ennesima versione di Space Invader, ma non può sfuggire la differente velocità di esecuzione del programma, che è nettamente superiore a quella dei giochi in BASIC.

Nel manuale in inglese, peraltro di semplicissima consultazione, si trova tutto ciò che necessita per un uso appropriato del programma. In particolare le funzioni vengono divise secondo questo principio: utilità generali, ausili per il suono, per la grafica, per la gestione degli sprite.

Ne risulta l'impressione di un pacchetto abbastanza utile in generale, ma sicuramente molto fantasioso, che, se utilizzato con estro può fornire dei risultati senza dubbio superiori alla norma.

Una nota di merito va sicuramente fatta per la casa produttrice di Acos +; non è infatti il primo prodotto di una certa qualità che viene messo sul mercato da questi della Melbourne House, in particolare, infatti, ci risulta abbiano lavorato benissimo anche su altri famosissimi microcomputer.

TERRORIST

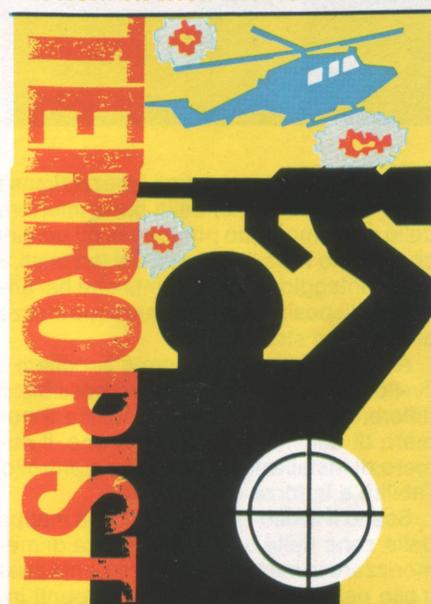
Casa produttrice
VIRGINGAMES

Configurazione
COMMODORE 64

Supporto
CASSETTA

Prezzo
L. 32.000

ACTION...ATTACK...ACTION...ATTACK



COMMODORE 64...JOYSTICK CONTROL

La città è stata attaccata dai terroristi, e tu, in qualità di capo della squadra speciale antiterrorismo, devi, con l'aiuto dei tuoi uomini, riprenderne il controllo. Per riuscire in questa difficile impresa devi fare appello a tutta la tua abilità ed avere una fervida memoria. Prima di addentrarti nella città ti viene mostrata una pianta dell'area dove dovrai intervenire: ficcatela bene in testa più velocemente che puoi! Adesso la tua missione dipende dall'abilità con cui sceglierai i mezzi a tua disposizione: l'automobile, il motoscafo e l'elicottero da combattimento; la salvezza della città è nelle tue mani e dipende dal modo con cui proteggerai l'area di tua competenza. I terroristi attaccheranno la città cercando di occupare i punti chiave della stessa; hai un solo sistema per evitarlo: volare sopra l'intera zona con il tuo elicottero da combattimento e ingaggiare battaglia con i terroristi, prima che questi abbiano occu-

pato l'edificio preso di mira.

Sul lato sinistro dello schermo vengono mostrate le installazioni o punti chiave che possono essere attaccati dai terroristi. Un segnalatore a forma di diamante, di fianco al simbolo dell'installazione, indica che questa è stata presa di mira (le installazioni attaccate possono essere più di una).



Vengono inoltre mostrati i mezzi rimasti a tua disposizione e, nella parte inferiore dello schermo, il tuo punteggio, gli uomini che ti sono rimasti, la forza dei terroristi, il loro punteggio e il carburante che hai ancora a disposizione per qualunque mezzo di trasporto stai usando.

All'inizio del gioco bisogna scegliere il livello di difficoltà fra i quattro possibili; il differente livello di gioco modifica il numero di uomini a tua disposizione, il numero di installazioni sotto il tuo controllo, l'abilità e la forza dei terroristi.

Scelto il livello viene mostrata la mappa della zona delle operazioni; cerca di memorizzare la mappa più velocemente possibile per evitare di perdere dei punti inutilmente.

Dopo queste operazioni preliminari inizia il gioco vero e proprio, la mappa viene cancellata e lo schermo si presenta senza più alcun punto di riferimento, senza strade e fiumi.

Con i mezzi a tua disposizione e fidando nella tua eccezionale memoria dovrai ricostruire la mappa della città; in questo modo potrai prevenire gli attacchi dei terroristi predisponendo degli uomini nei punti strategici ed evitando di ricorrere ad interventi di emergenza.

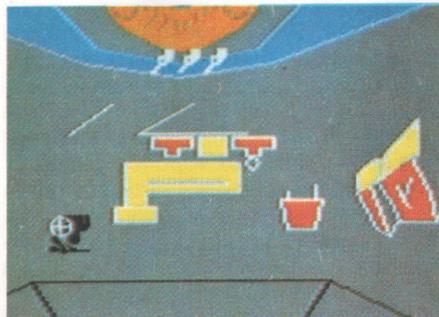
Alla partenza sei in automobile ma puoi subito cambiare mezzo.

Per spostare i mezzi sulla mappa puoi servirti dell'elicottero (in questa fase il consumo di carburante è notevole); anche l'esplorazione del terreno comporta notevole dispendio di carburante, tant'è che ogni tanto il mezzo che stai usando si blocca perché è finito il combustibile. Per rifornire i mezzi puoi passare il carburante dai serbatoi di riserva dell'elicottero, che a sua volta si rifornisce volando al centro della mappa e atterrando sul deposito di carburante della città. Cerca però di evita-

re gli sprechi perché, come tutte le riserve, anche quella di carburante è destinata a finire prima o poi, il che comporta la perdita di uomini e mezzi. Le installazioni che non hai ancora scoperto sono soggette all'attacco dei terroristi e vengono visualizzate, indipendentemente dall'esito dell'attacco stesso, quando esso è finito.

Il primo segnale di imminenza dell'attacco è un "beep" ripetuto più volte; l'installazione attaccata viene segnalata sulla parte sinistra dello schermo, che comincia a lampeggiare. Un suono specifico (nel caso del teatro, il Can can) indica che l'attacco all'installazione è cominciato. A questo punto inizia il conto alla rovescia ed hai a disposizione 10 secondi per prepararti.

L'ATTACCO - Una stella lampeggiante indica sulla mappa il punto che è stato attaccato. Voli con l'elicottero sopra la stella (ora lo schermo cambia mostrando l'interno dell'installazione attaccata) e datti da fare, più velocemente che puoi, per sconfiggere i terroristi. Ad intervalli molto rapidi compaiono le sagome degli attaccanti: inquadrate nel tuo mirino e spara subito perché se non sei abbastanza rapido, loro colpiranno i tuoi uomini. Alla fine della battaglia vengono calcolati e sommati nelle rispettive tabelle i punti a tuo favore e quelli a favore dei terroristi, ed in base alla differenza fra i due punteggi parziali l'installazione viene dichiarata libera o caduta in mani nemiche.



Il gioco finisce quando tutti i terroristi sono stati sconfitti (o hanno sconfitto te) oppure quando tutta la mappa è stata riscoperta. Se il punteggio ottenuto è abbastanza alto viene inserito nella tabella degli HIGH SCORE insieme agli altri cinque più alti punteggi.

Il gioco è molto divertente, originale e corredato da dettagliate istruzioni in inglese che spiegano i punteggi ottenibili, i punteggi di partenza per i vari livelli, l'elenco delle installazioni che possono essere soggette ad attacco, l'elenco dei tasti funzione da usare, nonché un sacco di consigli per un migliore svolgimento del gioco. Per giocare è necessario il joystick; il sistema di caricamento è realizzato in modo ultraveloce.

BEAMRIDER

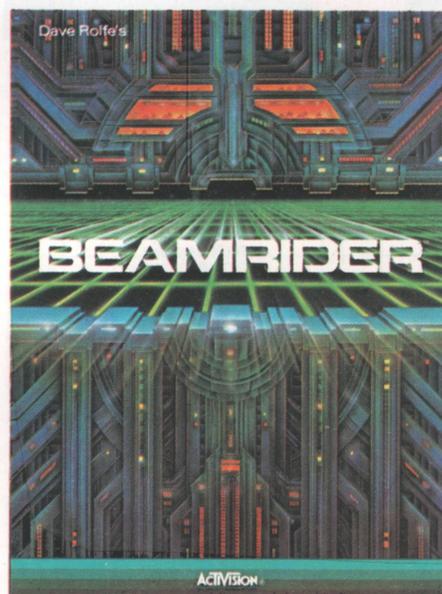
Casa produttrice
ACTIVISION

Autore
DAVE ROLFE

Configurazione
COMMODORE 64

Supporto
CASSETTA

Prezzo
L. 32.000



Niente da dire: quando si tratta di produrre videogames, Activision non sbaglia un colpo!

Anche nel caso di questo BeamRider, per prima cosa che ci colpisce al termine del suo purtroppo lento caricamento, è la grafica introduttiva; infatti, per quanto semplice, è curata fin nei minimi particolari, come è sempre stata consuetudine e dimostrazione di estrema professionalità da parte di questa ormai conosciutissima software house.

Vediamo adesso più da vicino questo nuovo videogame.

Tu sei il BeamRider, e la tua missione consiste nell'eliminare lo scudo restrittivo che circonda la Terra.

Attenzione, però, perché le sentinelle nemiche ne sorvegliano ogni suo settore.

Ti troverai subito nella stazione spaziale, il cui pannello di comando visualizza il tuo punteggio, due navi di riserva, il settore, il numero di dischi volanti nemici, ed infine tre siluri da usare solamente in caso di estrema necessità.

La tua nave-luce si può spostare a destra o a sinistra, in base alla direzione in cui muovi il joystick.

Per lanciare i raggi laser, è necessario premere il pulsante di sparo. Ogni settore contiene quindici dischi nemici che devono essere distrutti per poter passare al settore successivo.

Quando tutti e quindici i dischi bianchi di un dato settore sono stati distrutti, la loro nave sentinella resta indifesa e comincia a volare in alto fra i raggi.

Può essere distrutta solo da un siluro.

I raggi laser hanno effetto solo su alcuni invasori. I siluri distruggono invece il primo oggetto che incontrano sul raggio. Poiché si hanno a disposizione solo tre siluri, è necessario utilizzarli in modo intelligente.

Ogni settore presenta un nuovo pericolo rispetto a quello precedente; via via, si dovrà fare i conti con dischi nemici bianchi, frammenti spaziali marroni, navi stridenti gialle, navi di bloccaggio verdi, invasori blu, inseguitori arancioni, bombe a zig zag rosse, e mine magnetiche.

Soltanto alcuni di questi elementi, cosiddetti vulnerabili, possono essere distrutti con il laser; tutti gli altri devono essere evitati o silurati.

I premi bonus di settore aumentano con ogni nuovo settore superato; più elevato è quest'ultimo, tanto maggiore sarà il bonus.

I punti vengono così distribuiti:

40 punti ogni disco nemico bianco,
200 punti ogni nave stridente gialla,
300 punti ogni sentinella di settore.

Se riesci a distruggere la sentinella di settore, si riceverà in aggiunta ai punti sopra riportati, punti bonus per ogni nave della tua flotta (ogni nave si illuminerà e al tuo punteggio saranno aggiunti 100 punti più il bonus).

Se si riesce a raggiungere il settore 14 con un punteggio di trentamila punti o più, si sarà in grado di iscriversi al club dei Beamriders inviando al distributore Activision di zona una fotografia del teleschermo, a testimonianza del risultato raggiunto.

Alcuni trucchi per i giocatori più esperti e smalzati consistono innanzitutto nell'imparare a manovrare il joystick dolcemente, in modo da spostarsi agevolmente da un raggio all'altro; è inoltre pericoloso allontanarsi troppo dai raggi centrali, per non correre il rischio di ritrovarsi intrappolato in un angolo.

Inoltre, quando la sentinella si avvicina rapidamente, è meglio abbandonare il raggio da cui si intende sparare prima che le navi di bloccaggio vi atterrino, e quindi restare in attesa su un altro. Non appena le navi di bloccaggio hanno bloccato il raggio in questione, ci si sposta su un altro raggio non bloccato e si colpisce la nave con un siluro.



È infine estremamente importante osservare attentamente i movimenti dei nemici in questo tipo di videogioco, perché in genere hanno una forma ripetitiva, per cui è possibile anticiparli.

Per quel che riguarda considerazioni tecniche su grafica ed effetti speciali, possiamo senza dubbio affermare di trovarci ben al di sopra della media, soprattutto tenendo conto della relativa occupazione di memoria di questo videogame; solamente sedici Kbytes.

THE MONEY MANAGER

| |
|---------------------------------------|
| Casa produttrice TIMEWORKS |
| Configurazione COMMODORE 64 |
| Supporto CASSETTA |
| Prezzo NON IMPORTATO |

CAT. NO. 6301
64K

TIMEWORKS

Personal Computer Software

The Money Manager™

Home and business budget and cash flow system

| ITEM | BUDGET | % ACTUAL | % DIFF |
|------------|--------|----------|--------|
| PRESS C | ***** | JAN 1983 | ***** |
| INCOME | 2000 | 1870 | -110 |
| RENT | 300 | 26 | 18 |
| UTILS | 100 | 8 | 200 |
| CAR PMT | 200 | 17 | 200 |
| GROCERIES | 150 | 12 | 180 |
| AUTO/TRANS | 33 | 2 | 29 |
| MAINT/REP | 44 | 3 | 20 |
| MED/DENT | 50 | 4 | 20 |
| CLOTHING | 55 | 4 | 200 |
| EDUCATION | 0 | 0 | 0 |
| RECREATION | 79 | 6 | 39 |
| SAVINGS | 200 | 17 | 170 |
| VISA | 22 | 1 | 200 |
| TOTAL EXP | 1233 | 1558 | -345 |
| NET INCOME | 767 | 312 | -455 |

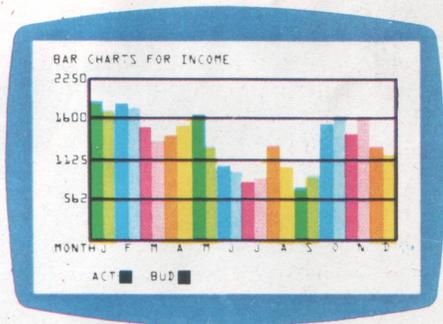
The Money Manager è un programma di utilità che interesserà certamente chi ha bisogno di tenere sotto controllo le spese di casa, ma vista la sua flessibilità può essere impiegato per moltissimi scopi.

Dopo aver risposto alla domanda di inzializzazione del sistema o di caricamento di dati da registratore potete scegliere il colore dello sfondo e scritte tra nero su bianco o viceversa; ora il computer vi chiede se deve caricare dei dati salvati con un altro programma: - The Electronic Checkbook -, anche qui si risponde Y/N.

Dopo aver inserito l'anno corrente vi comparirà davanti il menu principale che consta di 9 opzioni: la prima, ENTER BUDGET, vi chiederà il mese in cui intendete inserire i dati, e poi, i valori per tutte le sedici categorie; la stessa cosa fa ENTER ACTUALS.

Se, per caso, vi accorgete di avere sbagliato ad inserire dati, o volete aggiornare le spese, con le opzioni 3 e 4 potete rivedere e modificare budget e actuals.

Le opzioni 5, 6, 7 servono per consultare le vostre tabelle di dati: la 5 vi permette di consultare, per ogni mese, tutti i dati inseriti, con relative medie, percentuali, e totali.



Le 6 non vi dà più i dati di tutti i campi in un mese, ma i dati relativi ad un determinato campo, in tutti i mesi dell'anno.

La settima invece disegna un grafico a barre per un campo, in tutti i mesi, distinguendo tra i valori di budget e di actuals.

Se i campi non vi soddisfano, o ne avete bisogno altri, con l'opzione 8 potete cambiare il nome a ciascuno di essi. L'ultima opzione è di fine programma ma serve anche per salvare i vostri dati per utilizzarli un'altra volta.

In conclusione, è un buon programma, molto utile in vari casi, non è facile usarlo subito, ma dopo un po' di familiarità il tuo utilizzo diventa semplice e veloce.

Il poterlo interfacciare con un altro programma, l'ELECTRONIC CHECKBOOK, lo rende più sofisticato e più comodo.

La nota negativa è il non potere scegliere se salvare i dati su disco o su cassetta (che rallenta, a volte, il caricamento e il salvataggio) e una documentazione dati assai scarsa e non sempre chiara.

EGG COMPUTER

Copia riservata agli abbonati

Elettronica Giovane



4



**ANTEPRIMA
LAS VEGAS**
COMMODORE PC-128
ATARI 130XE

**LO
SPECIALE
MSX**

MSX