

# PERSONAL SOFTWARE



ANNO 4 N. 29  
GIUGNO 1985 - L. 4.000

LA PRIMA RIVISTA EUROPEA DI SOFTWARE PER PERSONAL COMPUTER

Spedizione in abbonamento postale gruppo III/70

Copie riservate agli abbonati



**Musica  
con un MSX**

**Monitor  
per C 64**

**Progetto  
aeromodello  
con lo  
Spectrum**

**Super Spreadsheet  
per C 16**

**Torneo di Bridge  
con Apple**



# Novità Jackson



John Scriven, Patrick Hall

## COMMODORE 64 A SCUOLA

Imparare è bello se lo si fa divertendosi; è il messaggio di questo libro scolastico ma divertente, sul Commodore 64.

Cod. 574D Pag. 182 Lire 18.000

Umberto Barzagli

## STATISTICA A UNA DIMENSIONE CON IL C64

Il primo manuale didattico espressamente studiato per l'apprendimento dei principi fondamentali della statistica tramite calcolatore.

Cod. 570A Pag. 172 Lire 17.000

Rita Bonelli

## COMMODORE 16 PER TE BASIC 3.5

Il primo libro-cassetta sul Commodore 16 per imparare il BASIC sul video.

Con cassetta.  
Cod. 413B Pag. 294 Lire 35.000

Czes Kosniowski

## MATEMATICA E COMMODORE 64

Un libro per chi vuol saperne di più sulle applicazioni matematiche del C64, per studiare ma anche per divertirsi.

Con cassetta.  
Cod. 570D Pag. 158 Lire 24.000



GRUPPO  
EDITORIALE  
JACKSON

## La biblioteca che fa testo

Totomac

## IL SISTEMA TOTOMAC la nuova frontiera del totocalcio per C64

Uomo del nostro tempo, Totomac si affida all'ausilio del computer per individuare le probabilità meno utilizzate dalla massa dei giocatori, e, avendo ben presente i principi della selezione naturale, cerca di batterli con tecniche intelligenti.

Con cassetta.  
Cod. 576D Pag. 128 Lire 24.000

Michael Browne

## UNITÀ A DISCHI per Personal Computer

Il libro, destinato a lettori con una buona conoscenza di base dell'uso di un calcolatore e del linguaggio BASIC, descrive il funzionamento dell'unità a dischi di un personal computer e il significato dei comandi relativi, con particolare attenzione per le diverse tecniche di gestione dei file su disco.

Cod. 300P Pag. 158 Lire 15.000

ritagliare (o fotocopiare) e spedire in busta chiusa a:  
GRUPPO EDITORIALE JACKSON - Divisione Libri - Via Rosellini, 12 - 20124 Milano

### CEDOLA DI COMMISSIONE LIBRARIA

#### VOGLIATE SPEDIRMI

n° copie	codice	Titolo	Prezzo unitario	Prezzo totale

Totale

Pagherò contrassegno al postino il prezzo indicato più L. 3.000 per contributo fisso spese di spedizione.

Condizioni di pagamento con esecuzione del contributo spese di spedizione:

Allego assegno della Banca  Allego fotocopia del versamento sul c/c n. 11666203 a voi intestato

N° \_\_\_\_\_  Allego fotocopia di versamento su vaglia postale e voi intestato

Nome e Cognome \_\_\_\_\_

Via \_\_\_\_\_

Cap \_\_\_\_\_

Città \_\_\_\_\_

Prov. \_\_\_\_\_

Data \_\_\_\_\_

Firma \_\_\_\_\_

Spazio riservato alle Aziende Si richiede l'emissione di fattura

ORDINE  
MINIMO  
L. 50.000

Partita I.V.A. \_\_\_\_\_

# SOMMARIO

<b>RAM TEST</b> <i>di Paolo Agostini</i>	COMMODORE 64	<b>12</b>
<b>SUPER SPREADSHEET</b> <i>di Angelo Motta</i>	COMMODORE 16	<b>17</b>
<b>I CHING: IL LIBRO DEI MUTAMENTI</b> <i>di Daniela Cerù</i>	TEXAS TI99/4A	<b>28</b>
<b>PROGETTO AEROMODELLO "U-CONTROL"</b> <i>di Paolo Ferrami</i>	ZX SPECTRUM	<b>34</b>
<b>MUSICA CON UN MSX</b> <i>di Paolo Ferrami</i>	MSX	<b>49</b>
<b>TASTI FUNZIONE PER SPECTRUM 48 KBYTE</b> <i>di Massimo Modelli</i>	ZX SPECTRUM	<b>53</b>
<b>NUOVO TORNEO DI BRIDGE</b> <i>di Vincenzo Delle Cave</i>	APPLE II	<b>58</b>
<b>MONITOR 64</b> <i>di Gianluca Puccio</i>	COMMODORE 64	<b>68</b>
<b>QUOZIENTE INTELLIGENZA</b> <i>di Ivano Parbuono</i>	ZX SPECTRUM	<b>96</b>

## RUBRICHE

<b>EDITORIALE</b> <i>di Riccardo Paolillo</i>		<b>5</b>
<b>POSTA</b>		<b>6</b>
<b>PERSONAL NEWS</b> <i>a cura di Marco Giacobazzi</i>		<b>8</b>
<b>I SEGRETI DEI PERSONAL:</b>		
<b>Palla che rimbalza</b> <i>di Sergio Borsani</i>	TEXAS TI99/4A	<b>102</b>
<b>File Examiner</b> <i>di Paolo Agostini</i>	COMMODORE 64	<b>104</b>
<b>L'architettura del sistema</b> <i>di Mauro Lenzi</i>	SHARP PC-1251	<b>109</b>
<b>PERSONAL MARKET</b>		<b>112</b>



Questo mese: Progetto aeromodello "U-Control", uno splendido programma per calcolare le strutture degli aeromodelli.

**ANNO 4**  
**N. 29**  
**GIUGNO 1985**

# SERVIZIO SOFTWARE

# Bit

Bit e Personal Software propongono ai propri lettori i dischi o le cassette dei programmi pubblicati. Uso e descrizione dei programmi si trovano sui rispettivi numeri delle riviste.



# PERSONAL SOFTWARE

Bit n°	Programma	Sistema	Codice	Supporto	P.S. n°	Programma	Sistema	Codice	Supporto
38	Gioco della scimmia	VIC 20	VI381A	Cassetta	3	La carta del cielo	Apple II	AP032A	Disco
	Spaccamattoni					Collisione			
38	Planet	Apple II	AP382B	Disco	4	Interi in precisione multipla	Apple II	AP042A	Disco
42	Apple-Chef	Apple II	AP422A	Disco		Grafica 30			
42	Provariflessi	VIC 20	VI421B	Cassetta	5	Pretty printer	Apple II	AP052A	Disco
45	Tiny FORTH	Apple II	AP452A	Disco		Shape table			
45	All Babà	ZX Spectrum	SP451B	Cassetta	7	Data base modulare	Apple II	AP072A	Disco
46	Forzaquattro	Apple II	AP462A	Disco	14	Tool-Kit	C 64	C6141A	Cassetta
48	Simulavolo	ZX Spectrum	SP481A	Cassetta	19	Type-Writer	VIC 20	VI192A	Disco
48	Memory Alfa IV	C 64	C6481B	Cassetta	20	Scopa	C 64	C6201A	Cassetta
49	Scorpion	Apple II	AP492A	Disco					
50	Fp-Plot	Apple II	AP502A	Disco					
50	Prima e Terza	ZX Spectrum	SP501B	Cassetta					
51	Magicalatog	Apple II	AP512A	Disco					
53	Partita a golf	VIC 20	VI531A	Cassetta					
53	Analisi numerica	C 64	C6531B	Cassetta					
53	PL/Bit: il compilatore	Apple II	AP532C	Disco					
54	Costellations	Apple II	AP542A	Disco					
54	Come polarizzare i transistor col C 64	C 64	C6541B	Cassetta					
58	Memory Omega I	C 64	C6582A	Disco					
58	Copy disk per C 64	C 64	C6582B	Disco					
59	Checksum 64	C 64	C6592A	Disco					
59	Checksum 64	C 64	C6591B	Cassetta					
59	Data-Bank	ZX Spectrum	SP591C	Cassetta					
60	Life HGR	Apple II	AP602A	Disco					
60	Tutti pittori	C 64	C6601B	Cassetta					
60	Difesa della Terra	C 16	C1601C	Cassetta					
60	Lost on the pack	Sega	SE601D	Cassetta					
60	Seidata e Wordproc	ZX Spectrum	SP601E	Cassetta					
60	Il Barone Rosso	TI99/4A	TI601F	Cassetta					
60	Word processor	C 64	C6602G	Disco					
60	Othello	VIC 20	VI601H	Cassetta					
61	Calcolo enigmatico in Pascal	Apple II	AP612A	Disco					
61	Disegno di mobili componibili	C 16	C1611B	Cassetta					
61	Esperimento di Millikan	C 64	C6611C	Cassetta					
61	Esperimento di Millikan	C 64	C6612D	Disco					
61	Disegnare in alta risoluzione	ZX Spectrum	SP611E	Cassetta					
61	Printing music	TI99/4A	TI611F	Cassetta					
61	Musica facile	Sega	SE611G	Cassetta					
61	VIC-Calc	VIC 20	VI612H	Disco					
62	Gestione conto/corrente	C 64	C6622A	Disco					
62	Gioco della tombola	TI99/4A	TI621B	Cassetta					
62	Aspc: lo Spectrum contro la carie	ZX Spectrum	SP621C	Cassetta					

Tutti i dischi e le cassette dei programmi sono in vendita a L. 15.000 ciascuno.

Per richiedere i programmi in contrassegno, pagando direttamente al postino la cifra indicata, inviare il seguente tagliando  
**Spedire in busta chiusa a Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano**

Inviatemi i seguenti nastri e/o dischi con i programmi pubblicati su Bit - Personal Software

Cod.  a L.  **15000**

Cod.  a L.

Cod.  a L.

Cod.  a L.

**+ SPESE POSTALI (contributo fisso)**  **3000**

**TOTALE L.**

che pagherò al postino alla consegna del pacco



**GRUPPO EDITORIALE JACKSON**

Cognome.....

Nome .....

Indirizzo .....

CAP.....

Città .....

Firma.....

## Oltre il personal

**L**a spettacolare Esposizione mondiale di Tsukuba (Giappone), attualmente in pieno svolgimento, ci offre l'occasione di parlare un po' del nostro futuro.

Le splendide vetrine presentate alla rassegna vogliono celebrare il trionfo della scienza al servizio dell'uomo. In quest'ottica, i numerosissimi robot impiegati nelle attività più disparate, lungi dal voler simboleggiare una ipotesi di automizzazione del genere umano significano l'esatto contrario: delegando a delle macchine attività ripetitive, noiose e pericolose saremo in grado di riconquistare una ulteriore fetta di vita da vivere umanamente. E perché questo concetto non rimanga fine a sé stesso, ecco che viene applicato all'agricoltura. Sono così presentati i frutti dalle dimensioni spropositate (ma ottimi come sapore e potere nutritivo), provenienti da culture idroponiche controllate da computer.

Lo scopo dell'operazione è comunque oltre che scientifico anche propagandistico. Infatti, l'idea di base è quella di convincere il mondo di una supremazia nipponica in campo tecnologico.

La realtà, invece come ben sappiamo, è alquanto differente, e probabilmente continuerà ad esserlo per parecchio tempo soprattutto nei campi chiave. A parte il mercato dei componenti e quello dei robot in cui i giapponesi recitano un ruolo di primo piano, in altri campi fondamentali quali l'informatica, la supremazia statunitense non viene messa in discussione da nessuno.

Motivi storici e geografici (in Giappone non esiste una Silicon Valley abitata da una vasta serie di geni del software) fanno sì che soprattutto il mercato del personal computer sia saldamente in mani americane. Bisogna comunque riconoscere che l'idea che qualcuno ancora ha che i giapponesi siano solo degli abili copiatori, si sta dimostrando sbagliata: nel 1984 l'industria giapponese ha registrato un numero di brevetti maggiore di quella americana. La maggior parte di questi successi sono stati ottenuti nel campo dell'elettronica di consumo; ecco quindi far bella di mostra di sé il televisore più grande del mondo o il cinema in cui si proietta il film che può, entro certi limiti, essere condizionato dagli stessi spettatori.

Ma in definitiva come sarà il nostro futuro? Sicuramente a misura d'uomo e questo è quello che conta.

Ma sarà anche spettacolare e ben curato artisticamente perché, come è noto, anche l'occhio vuole la sua parte.

Avremo quindi una tecnologia avanzatissima, ma dal volto umano. Se tutti questi discorsi siano solo demagogia lo sapremo prestissimo, anzi subito: il futuro tecnologico, infatti, è già iniziato.

## Caratteri misteriosi

Abito a Lanzo in provincia di Torino. Frequento la terza media presso il collegio Salesiano di Lanzo dove tantissimi miei amici e compagni come me, possiedono da Natale, un C 64 e pure loro come me possiedono tantissime riviste e trovano le mie stesse difficoltà nell'eseguire la battitura dei programmi di queste ultime. Io, a nome dei miei compagni, mi sono preso la responsabilità di questa lettera. A noi interesserebbe sapere quale rivista insegna come battere i caratteri presenti in Personal Software.

Space Traveller e Frogger per C 64 n. 23.

[<1 CLR>]	[<1 R.C.>]
[<1 WHT>]	[<1 G.2>]
[<1 RED>]	[<1 V.C>]
[<1 CYN>]	[<1 AZZ>]
[<1 PUR>]	[<3 CHR\$(166)>]
[<1 GRN>]	[<1 HOME>]
[<1 BLU>]	[<4 CRSR D>]
[<1 YEL>]	[<4 CRSR R>]
[<1 ARA>]	[<4 CRSR L>]
[<1 MAR>]	

Fabio Cominato  
Lanzo (TO)

Il C 64 e i personal Commodore in genere, ci hanno sempre dato qualche problema per la stampa dei listati. I problemi sono dovuti ai tasti grafici e speciali che spesso in stampa sono di difficile interpretazione. L'esperienza (cioè il numero di lettere ricevute) ci insegna che pubblicare listati prodotti direttamente dalla stampante Commodore può causare grosse difficoltà a chi intende ricopiarli; così sono stati ideati vari sistemi per la stampa di listati decodificati.

Space traveller e Frogger, ad esempio, sono stati listati con uno di questi programmi. Il significato della simbologia utilizzata è stato più volte pubblicato in numeri precedenti della rivista, ma lo ripetiamo volentieri per chi non lo conosce.

Tutti i tasti grafici e speciali sono stati racchiusi entro le parentesi quadre e tra i simboli di minore e maggiore.

Inoltre un numero, precisa quante volte

occorre digitare il tasto indicato. Ad esempio [<1 Wht>] significa che occorre digitare una volta il tasto 2 contemporaneamente al tasto Ctrl (il cursore diventa bianco), [<4 Crsr D>] = 4 volte il tasto di spostamento verso il basso (Down) del cursore e così via. Per i caratteri grafici viene presentato il codice ASCII corrispondente: occorre consultare il manuale del C 64 per sapere a quale tasto corrisponde. Ad esempio [<3 Chr\$(166)>] significa che occorre digitare 3 volte il simbolo grafico che si ottiene premendo contemporaneamente il tasto + e il tasto Commodore. Da questo numero di Personal Software adottiamo un nuovo sistema di codifica ideato dai nostri colleghi di Super-Commodore. La legenda è consultabile in una tabellina che verrà pubblicata in ogni numero della rivista. Inoltre la stampa verrà realizzata mediante una stampante a margherita che consentirà di migliorare notevolmente la qualità. Questo sistema sarà utilizzato per produrre i listati di tutti i Commodore, compreso il C 16.

## Programmi Mio cercasi

Una domanda da girare alla rivista Personal Software.

Quando vi ricorderete che in Italia (e forse anche fuori) vi sono dei possessori di Mio che aspettano famelici ed impazienti di vedere (e magari con relative spiegazioni) qualche programma frutto di altrui fatiche?

Grazie per l'ospitalità. W l'Mio e l'Olivetti. P.S.: scherzi a parte, continuate pure così che va bene lo stesso.

Caiella Marco  
Ponte S. Giovanni (PG)

Appelli di questo tipo ne arrivano parecchi: questo è uno dei più simpatici e lo pubblichiamo volentieri.

In passato, per altre macchine, i risultati erano stati incoraggianti.

Forza allora utilizzatori di Olivetti Mio; inviateci i vostri programmi! Anche se, ci scusi Signor Caiella, va bene aspettare il frutto delle altrui fatiche, ma qualcosina si potrebbe anche farla in proprio. O no?

# Ecco l'elenco completo dei 100 Jacksoniani, fortunati vincitori del concorso abbonamenti Jackson '85.

## CONGRATULAZIONI!!!



**Airoldi Marco**  
Via R. Felisatti, 5  
44100 Ferrara

**Aliboni Enrico**  
P.zza d. Repubblica, 19  
27026 Garlasco (PV)

**Amabili Stefania**  
Via M. Bartolotti, 8  
48023 Marina di Ravenna (RA)

**Alessandrelli Fabio**  
Via Fano, 35  
60128 Ancona

**Antolini Mariano**  
Via Isnardi, 32/19  
16016 Cogoleto (GE)

**Antonelli Antonio**  
Via Adriatico Vico X, 8  
66036 Orsogna (CH)

**Bandello Melide**  
Via Pasubio, 4  
37045 Legnago (VR)

**Barzaghi Giovanni**  
Via S. Stefano, 83  
20057 Veduggio al Lambro (MI)

**Baschiera Bruno**  
Via Rizzos  
33091 Castelnuovo Del Friuli (UD)

**Basile Modesto**  
Via L. Sturzo, 39  
67100 L'Aquila (AQ)

**Beccari Alberto**  
Via XXV Aprile  
46010 Breda Cisoni (MN)

**Benedetti Ario**  
Via Pigafetta, 32  
00154 Roma

**Biancheri Giovanni**  
Via S. Rufino, 16  
16040 Leivi (GE)

**Brizzi Andrea**  
V.le Europa, 18  
39100 Bolzano

**Burzagli Laura**  
Via Traversi, 7  
50127 Firenze

**C.B.L. Computers s.n.c.**  
Via S. Carlo, 13  
98060 S. Angelo di Brolo (ME)

**Cacudi Alessandro**  
Via Benanducci, 6  
72021 Francavilla Fontana (BR)

**Careddu Mauro**  
Via Is Mirrioni, 95  
09100 Cagliari

**Carettoni Roberto**  
Via Europa, 21  
20010 Pogliano Mil. (MI)

**Caporale Enzo**  
Via Ten. Vitti, 93  
70043 Monopoli (BA)

**Casali Aldo**  
P.zza Caiazzo, 3  
20124 Milano

**Cassai Fulvio**  
Via Papa Giovanni XXIII, 36  
20097 Melegnano (MI)

**Ceccato Giovanni**  
Via Pacinotti, 4  
36060 Romano D'Ezzelino (VI)

**Chiavieri Renato**  
P.zza Mazzini, 20  
44030 Serravalle (FE)

**Chillà Aldo**  
Via Verdi, 6  
50055 Lastra a Signa (FI)

**Cogliati Francesco**  
Via Friuli, 22  
20092 Cinisello B. (MI)

**Cursaro Paolo**  
Via Nazionale, 100  
84040 Capaccio Scalo (SA)

**D'Amico Michele**  
Via Giotto-P.co Gabriella  
81100 Caserta

**De Carlo Franco**  
Via Bertola, 9  
10121 Torino

**Di Stefano Gaetano**  
Via Scarcella, 67  
95030 Trappeto (CT)

**De Marzo Silvio**  
Via Napoli, 5  
89024 Polistena (RC)

**Fabbretti Giuseppe**  
Via Delle Baleniere, 92  
00121 Ostia Lido (Roma)

**Fabbri Fabio**  
Via Cellini, 16/C  
57023 Cecina (LI)

**Fantechi Renzo**  
Via Forlivese, 29  
50065 Pontassieve (FI)

**Farroni Pierluigi**  
Via Borgo S. Lorenzo, 2  
62020 Loro Piceno (MC)

**Fogliano Pierluigi**  
Via Capeceolatro, 22  
20148 Milano

**Galliano Bruno**  
Via G. Ferraris, 39  
15100 Alessandria

**Gavinelli Sandro**  
Via Libertà, 52  
28043 Bellinzago (NO)

**Ghezzi Guglielmo**  
Via dei Pini, 58  
45010 Rosolina (RO)

**Giordano Giancamillo**  
Via M. Ausiliatrice, 36  
10152 Torino

**Grilli Giuliano**  
Via Di Mezzo, 33  
41032 Cavezzo

**Guerrini Roberto**  
Via Pascoli, 3  
53027 S. Quirico D'Orcia (SI)

**Jannoni Sebastianini Giulio**  
Via Livorno, 89  
00162 Roma

**I.T.I.S. "A. Volta"**  
P.zza S.M. della Fede, 16  
80141 Napoli

**I.T.I. di Castelfranco Veneto**  
Via dei Carpani, 19  
31033 Castelfranco V. (TV)

**I.P.S.I.A. "Giorgi"**  
Via Tenaglio, 53  
31100 Treviso

**Lenza Dino**  
Via Virgilliana, 3  
46100 Mantova

**Lenza Pietro**  
Via degli Eucalipti, 14  
84100 Salerno

**Levi Moise**  
Via Sammartini, 37  
20125 Milano

**Lignocchi Roberto**  
Via B. Bonini, 11  
25127 Milano

**Loretelli Albertino**  
Via dei Villini, 23  
04011 Aprilia (LT)

**Lunghi Giuseppe**  
Via Gramsci, 5  
26013 Crema (MI)

**Luparia Renato**  
Via Vallescura, 185  
15030 Conzano (AL)

**Mangiaracina Antonio**  
Via Vill. Messina, 5  
92018 S. Margherita di Belice (AG)

**Marcelli Rossella**  
Via G. Nadi, 14  
40139 Bologna

**Marchi Raffaele**  
Via Metauro, 38  
65010 S. Teresa di Spoltore (PE)

**Martini Francesco**  
Via Passo del Turchino, 60  
00139 Roma

**Micro Control snc**  
Via Claudia, 4557  
41056 Savignano S/Panaro (MO)

**Montaperto Giuseppe**  
Via della Liberazione, 43  
20090 S. Maurizio al L. (MI)

**Morgano Oreste**  
Via Nazionale, 127  
94010 Calascibetta (EN)

**Musacchia Benedetto**  
Via Terra Santa, 92  
90141 Palermo

**Natalini Simone**  
Via Ciurini, 25  
56013 Marina di Pisa (PI)

**Nocera Gaspare**  
Via Edif. A 36  
21020 Ispra (VA)

**Oberino Silvio**  
Via Valle Soana, 59  
10085 Pont Canavese (TO)

**Olivari Claudio**  
Via Piave, 6  
46034 Governolo (MN)

**Lenza Monica**  
Via Pietro Donati  
26013 S.M. della Croce - Crema (CR)

**Pagnini Marco**  
Via Partigiani, 15  
61100 Pesaro

**Pala Bruno**  
Via Matteotti, 33  
09026 San Sperante (CA)

**Pennestre Claudio**  
Via C. Cavour, 21  
34077 Ronchi dei Legionari (GO)

**Perpignani Andrea**  
Via Tuberose, 8  
20146 Milano

**Piana Guido**  
Via Petrarca, 39  
07021 Arzachena (SS)

**Pizzirani Luigi**  
Via Pessina, 34  
70125 Bari

**Poli Carlo**  
Via Stradella, 1  
20129 Milano

**Pool Informatica**  
Via Emilia S. Stefano, 9/c  
42100 Reggio Emilia

**Priotti Gabriele**  
Via Piazani, 2  
28025 Gravellona Toce (NO)

**Rapaccini Silvio**  
Viale R. Margherita, 84  
57025 Piombino (LI)

**Ritarossi Mario**  
Via Cerreto di Spoleto, 48  
00181 Roma

**Rossi Alessandro**  
Via Yuri Gagarin, 5/2  
40044 Pontecchio Marconi (BO)

**Rossi Guido**  
P.zza Vittorio Emanuele II, 5  
50065 Pontassieve (FI)

**Russo Enrico**  
Via Cacciapuoti, 58  
84014 Giuliano (NA)

**Sammarco Giovanni**  
Via del Faro, 72  
74020 San Vito (TA)

**Sardano Filippo**  
L.go Veneziani, 16  
70043 Monopoli (BA)

**Satema di G.&L. F.lli Grillo**  
Via Milano, 473  
13069 Vigliano Biellese (VC)

**Schiassaro Roberto**  
Via Porta Po, 47  
45100 Rovigo

**Scala Livio**  
Villaggio Riviera, 35  
32010 Farra D'Alpago (BL)

**Scotta Eligio**  
Via Castello, 65  
12039 Verzuolo (CN)

**Sessa Sergio**  
Via Gramsci, 42  
20099 Sesto S. Giovanni (MI)

**Sestito Aurelio**  
Via Martelli, 14  
88064 Chiaravalle Centrale (CZ)

**Sichel Teresa**  
Via Volontari del Sangue, 1  
29010 Pontenure (PC)

**Sparaco Lucio**  
Via Aldo Moro, 169  
81055 S. Maria Capua Vetere (CE)

**Talabbe Silvano**  
Via Purocelo, 21  
40026 Imola (BO)

**Tarallo Vincenzo**  
Via G. D'Agostino, 18  
89029 Taurianova (RC)

**Tarzia Simonetta**  
Via Terpi, 17/6  
16141 Genova

**Tincani Daniele**  
Via Trieste, 6  
55040 Retignano di Stazzema (LU)

**Tomasella Miguel Angel**  
Via S. Tiziano, 5  
31020 Zoppè di S. Vendemiano (VI)

**Tuccella Silvano**  
Via Madonna d. Misericordia, 12  
66100 Chieti

**Turci Andrea**  
Via Dormelletto, 84  
28041 Arona (NO)

**Turnu Efisio**  
Via Cesaro Balbo, 16  
09170 Oristano (OR)

**Zattoni Raffaele**  
Via Palazzolo, 8  
48020 S. Romualdo (RA)

**Zollo Sandro**  
Via Monastero, 127/1  
17026 Noli (SV)



## Facciamoli parlare

**L**a Welwyn Systems sta producendo in grossi volumi per il mercato europeo i sintetizzatori vocali Currah nelle versioni adatte per il Commodore 64 e lo ZX Spectrum. L'esperienza dell'azienda inglese e la buona disponibilità a magazzino dei nuovi prodotti, permettono di prevedere una discreta diffusione degli accessori, lanciati inizialmente con una garanzia piena di 12 mesi.

*Welwyn Electronics  
Via F. Albani, 3  
20148 Milano  
Tel. 02-490788*

## Musica e Commodore

**D**alle colonne delle riviste del Gruppo Editoriale Jackson si parla spesso dell'integrazione tra musica e informatica, presentando e commentando prodotti, che realizzano una sintesi tra due mondi molto vicini. Uno scoglio da superare è però spesso rappresentato dal costo di tastiere aggiuntive o software specializzato, non sempre abbordabile dall'appassionato. La Commodore sembra venire incontro alle esigenze dei giovani musicofili presentando due semplici ed economici prodotti che consentono di sfruttare il SID del C 64 per prendere confidenza con la musica o suonare con il computer. Si tratta di Musico, un programma interattivo per imparare a leggere e scrivere la musica, sviluppato da musicisti del CEPAM (Centro Permanente Attività Musicali) di Reggio Emilia, e di Music Maker, package che trasforma il C64 in una tastiera. Musico è un insieme di programmi su floppy che in dieci "sedute", sette lezioni e tre unità di verifica e d'esercitazione, introduce l'allievo ai concetti di base dell'acustica e della grammatica musicale. L'interattività del sistema consente di scegliere il modo più personale di muover-

si tra lezioni ed esercizi: per evitare noiose ripetizioni questi ultimi sono legati all'impiego delle funzioni casuali, presentandosi così come nuovi ogni volta che si usa il programma. Altri aspetti particolarmente curati di Musico sono la grafica (per la rappresentazione di note e pause), il suono e la concezione didattica. Questo programma, di produzione totalmente italiana, è distribuito dalla Commodore per 75.000 lire, più IVA. Music Maker offre invece la possibilità di suonare direttamente il C 64, senza necessità di utilizzare tastiere esterne. Oltre al supporto contenente il programma, viene infatti offerta una "tastiera" musicale da applicare su quella alfanumerica della macchina e una serie di adesivi per consentire anche al principiante di riconoscere le note. Il controllo delle possibilità offerte da Music Maker avviene attraverso i quattro tasti funzione del Commodore 64, secondo modalità che il programma stesso si incarica di comunicare. E' possibile inoltre memorizzare sequenze di note su cassetta o disco, scriven-





do separatamente la melodia ed il ritmo. Music Maker è in vendita a 55.000 lire più IVA.

Commodore Italiana S.p.A.  
Via F.lli Gracchi, 48  
20092 Cinisello Balsamo (MI)  
Tel. 02-618321



abrasività della dispersione magnetica consente una maggior durata effettiva delle testine di lettura/scrittura. Capacità massima 500 Kbyte, garanzia illimitata come per tutti i dischi venduti dalla casa americana.

3M Italia S.p.A.  
20090 Milano S. Felice - Segrate (MI)  
Tel. 02-75452595

## Hit parade Mastertronic

**L**a società che distribuisce i giochi prodotti dalla casa di oltremarica ha inserito nel proprio catalogo una serie di titoli realizzati dalla British Telecom. Questi game, della serie Firebird, hanno spopolato in Inghilterra e pare che siano stati accolti da un buon successo anche nel nostro paese, stando almeno alle classifiche diffuse dalla Mastertronic, che vedono per il mese di Marzo il nuovo entrato Booty al terzo posto. La prima posizione viene lasciata da *Chiller* che la cede a 1985 *The day after*, un gioco che richiede l'attraversamento di 12 territori ostili per recuperare l'energia necessaria alla sopravvivenza della Terra distrutta.

Persona  
Via A. Mario, 11  
37100 Verona  
Tel. 045-592960

interfaccia seriale RS-232C o parallela a 8 bit. Il supporto per simboli e comandi grafici è opzionale.

Vianello S.p.A.  
Via Tommaso da Caszaniga, 9/6  
20121 Milano  
Tel. 02-6596171

## Microdischi garantiti a vita

**L**a 3M ha avviato la distribuzione dei propri modelli di microdischi da 3,5", compatibili con tutti i drive che utilizzano questo formato (Apple, HP, D.G., alcuni MSX). La levigatura superficiale del rivestimento magnetico garantisce un contatto "ravvicinato" tra testina e disco, mentre la bassa



## Plotter a quattro colori

**E'**giunto in Italia il plotter X-Y PL-1000 della YEW, distribuito dalla Vianello S.p.A.. L'area coperta è di 10 per 15 pollici su carta o, opzionalmente, su lucidi. Sono previste fino a quattro penne con diversi colori ed il firmware mette a disposizione del programmatore una serie di funzioni intelligenti quali cerchi, archi, coordinate, caratteri opzionali, simboli e vari tipi di linee. Il PL-1000 può funzionare anche come stampante ed è dotato di



## Stampante "compatibile" dal Giappone

**L**a iDP 560 Citizen è un'interessante stampante economica che si può collegare, oltre che a tutti i computer dotati di uscita seriale (RS-232) o parallela (Centronics), anche alle macchine Commodore (VIC 20 e C-64). Stampa su 40 colonne in due colori, utilizzando carta da 69 mm, alla velocità di 65 caratteri al secondo. La testina è a matrice di punti, 5 per 7, e può operare anche in modo grafico. Pesa 2 kg e ha un ingombro

di 24 per 17,6 per 8,2 cm. La versione con interfaccia Centronics o attacco Commodore costa 270.000 lire più IVA, mentre il modello con interfaccia seriale RS-232C ha un prezzo di 320.000 lire, IVA esclusa.

*Fanton S.r.l.*  
Via Cimabue, 5  
10137 Torino  
Tel. 011-3097347

no, edito dalla JCE. Disponibili anche una serie di accessori prodotti da costruttori inglesi indipendenti, reperibili in Italia presso gli importatori autorizzati, alcuni dei quali sono stati da noi già citati in queste pagine.

*Rebit Computer*  
Viale Matteotti, 66  
20092 Cinisello Balsamo (MI)  
Tel. 02-6181801

altri accessori quali le batterie ricaricabili. Le caratteristiche tecniche sono queste: 80 colonne per 24 righe, risoluzione di 560 per 192 pixel, peso 1,2 kg, controlli di contrasto e luminosità.

*Apple Computer*  
Palazzo Q8 - Milanofiori  
20089 Rozzano (MI)  
Tel. 02-8242156

### Didattica e computer

**I**l Cineca, centro di calcolo interuniversitario di Bologna, ha organizzato una serie di corsi e di seminari sulla progettazione, realizzazione e revisione critica di software didattico. Dopo le giornate di Marzo, il corso viene replicato a Settembre, in due parti: la prima dal 9 al 13, l'altra dal 23 al 27. Oltre agli argomenti previsti, verrà dato ampio spazio alle esercitazioni pratiche, ed ogni partecipante verrà dotato di un proprio strumento. Il costo complessivo è di 1.200.000 lire, che viene ridotto del 60% ai docenti delle università e delle scuole pubbliche.

*Cineca*  
Via Magnanelli, 6/3  
40033 Casalecchio di Reno (BO)  
Tel. 051-576541 (Dott. Ivan Grossi)



### L'Apple IIc va da solo

**A** quasi un anno dalla sua diffusione sul nostro mercato l'Apple IIc viene completato dall'annuncio (con disponibilità immediata) del visore a cristalli liquidi, che permette di utilizzare il compatto computer Apple senza necessità di monitor o video casalinghi. Il Flat Panel Display è un po'



caruccio, 1.290.000 lire più IVA, ma è molto interessante per usi professionali e didattici che possono richiedere una certa autonomia d'uso, da realizzare anche con

### Ti leggo la mano...

**F**orse sta per finire l'era dei badge, quelle tesserine plastificate che consentono l'accesso ad alcuni luoghi riservati, permettendo l'apertura di certe porte automatiche al solo



personale autorizzato. Dagli USA arriva la notizia dell'installazione di un sistema basato sul riconoscimento delle impronte digitali, che permette un livello di sicurezza non raggiungibile con i tradizionali mezzi in uso. Realizzato dalla Identix di Palo Alto, il sistema di sicurezza elettronico che riconosce le impronte digitali è stato installato ... Provate a indovinare? Ma sì, proprio al quartier generale dell'F.B.I. al decimo piano del J. Edgar Hoover Building!

*Identix Incorporated*  
2452 Watson Court  
Palo Alto - CA 94303

### Il QL avanza

**I**n vendita da Gennaio in versione originale inglese, al prezzo di 1.149.000 lire più IVA, il Sinclair QL è fornito insieme a quattro programmi applicativi che consentono di sfruttare al meglio le potenzialità della macchina costruita attorno al Motorola 68008. Corredato da certificato di garanzia italiano valido in tutti i centri autorizzati, viene commercializzato dalla Rebit Computer, una divisione della GBC. E' pronto anche il manuale in italia-

# Novità Jackson.

David Lawrence

## LINGUAGGIO MACCHINA DEL COMMODORE 64

Il libro apre nuovi orizzonti a tutti coloro che sono interessati alla programmazione in linguaggio macchina del COMMODORE 64.

Con cassetta  
Cod. 572D Pag. 208 Lire 29.000

Clive Prigmore

## IL BASIC IN 30 ORE PER SPECTRUM

Questo semplice corso di autoistruzione insegna a programmare, e un programma ha sempre bisogno di due ingredienti, un linguaggio e una struttura: dunque questo libro non insegna solo il BASIC, ma anche come si organizza correttamente un buon programma.

Cod. 501B Pag. 360 Lire 40.000

Rodnay Zaks

## IL TUO PRIMO PROGRAMMA IN BASIC

La diffusione del BASIC per la sua semplicità e quasi "naturalità" di programmazione fa sì che una cultura generale sull'informatica e la sua applicazione non può prescindere da una conoscenza di base di questo linguaggio. Questo lo scopo del libro: permettere anche a chi ha soltanto una cultura di base, di capire che cos'è il BASIC e come si usa.

Cod. 507B Pag. 216 Lire 19.500

Czes Kosniowski

## MATEMATICA E COMMODORE 64

Tutte le funzioni matematiche disponibili sul C64 sono qui descritte, ed il loro uso è illustrato con programmi che possono essere utilizzati dal lettore all'interno dei suoi, per particolari applicazioni.

Il libro contiene anche informazioni e programmi su altri argomenti, come i codici e la crittografia, i numeri casuali, le serie, la trigonometria, i numeri primi e l'analisi statistica dei dati.

Con cassetta  
Cod. 570D Pag. 160 Lire 24.000

F. Williams

## AI CONFINI DELLO SPECTRUM Applicazioni avanzate

Un esame attento dei listati consentirà al lettore di apprendere i "segreti" della programmazione strutturata e migliorare notevolmente le proprie capacità di programmatore.

I programmi presentati vanno dagli arcade più famosi, tra cui il celebre "Spectrum Invaders" ai programmi di utilità più interessanti, dai giochi d'azzardo ai programmi didattici, dai programmi funzionali a quelli di matematica e di giochi di strategia.

Con cassetta  
Cod. 414B Pag. 180 Lire 28.000

Mike Grace

## ADVENTURE E COMMODORE 64

Un manuale per ideare e utilizzare programmi di Adventure basati esclusivamente sul testo.

Una progettazione modulare del programma rende più facile la comprensione della struttura. L'abilità nella programmazione avrà modo di emergere durante la costruzione graduale dell'Adventure, per modificare il programma dimostrativo, o addirittura per costruirne uno nuovo.

Con cassetta  
Cod. 571D Pag. 240 Lire 35.000

Rita Bonelli

## COMMODORE 16 PER TE: BASIC 3.5

È un libro di introduzione al BASIC C16 con il classico taglio didattico Bonelli. Adatto per la Scuola media inferiore e per chi non conosce l'informatica.

La cassetta allegata al libro contiene diverse lezioni, una per ogni capitolo, che devono essere lette prima del capitolo relativo.

Con cassetta  
Cod. 413B Pag. 296 Lire 35.000



**GRUPPO EDITORIALE JACKSON**

## La biblioteca che fa testo.

ritagliare (o fotocopiare) e spedire in busta chiusa a:  
GRUPPO EDITORIALE JACKSON - Divisione Libri - Via Rosellini, 12 - 20124 Milano

### CEDOLA DI COMMISSIONE LIBRARIA

#### VOGHIATE SPEDIRMI

n° copie	codice	Prezzo unitario	Prezzo totale
Totale			

Pagherò contrassegno al postino il prezzo indicato più L. 3.000 per contributo fisso spese di spedizione.

#### Condizioni di pagamento con esenzione del contributo spese di spedizione:

- Allego assegno della Banca  Allego fotocopia del versamento su c/c n. 11666203 a voi intestato
- Allego fotocopia di versamento su vaglia postale a voi intestato

n° \_\_\_\_\_  
 Nome \_\_\_\_\_  
 Cognome \_\_\_\_\_  
 Via \_\_\_\_\_  
 Cap \_\_\_\_\_ Città \_\_\_\_\_ Prov. \_\_\_\_\_  
 Data \_\_\_\_\_ Firma \_\_\_\_\_

Spazio riservato alle Aziende. Si richiede l'emissione di fattura

**ORDINE MINIMO**  
L. 50.000

Partita I.V.A. \_\_\_\_\_

**C**apita a tutti di pensare prima o poi: "Ma funziona davvero bene la memoria del mio computer?". In genere ci si accorge del malfunzionamento della memoria quando accade di inserire una stringa, per esempio: A\$ = "abcde" e alla richiesta di visualizzazione della stringa si ottiene qualcosa come "a & c)e" o una serie di segni senza senso. Ma com'è possibile controllare la presenza o meno di difetti? Col BASIC si tratta di un'impresa ardua e lunga, in quanto si deve effettuare un test sulla memoria stessa che contiene il programma in linguaggio BASIC. Per far questo sarebbe necessario modificare i puntatori della memoria BASIC spostandoli più in alto e si dovrebbe far funzionare il computer ininterrottamente perlomeno un giorno intero per permettergli di completare il programma.

# RAM test

## Controlliamo se ci sono difetti nella RAM del nostro Commodore

di Paolo Agostini

Per questa ragione si deve necessariamente ricorrere al codice macchina, veloce e di facile implementazione, e che ha inoltre il pregio di poter essere memorizzato in una locazione di memoria al di fuori della memoria RAM che dovrà essere sottoposta al test.

Il programma in linguaggio macchina

qui presentato effettua una serie di test sulle singole locazioni di memoria RAM, a partire dalla locazione 2048 (esadecimale \$0800) fino alla 40959 (esadecimale \$9FFF). Ciò permette di controllare che i chip costituenti la memoria del Commodore 64 esplicino la loro importantissima funzione senza difetti che

### Listato 1 - Il programma sorgente in BASIC.

```

100 REM RAM-TEST DI P. AGOSTINI/PADOVA 19
85
110 PRINTCHR$(147);"[<1CRSR D>]ATTENDERE
PREGO"
120 DATA 169,147,032,210,255,169,000,141,
116,194,153
130 DATA 141,117,194,032,119,194,013,032,
032,032,138
140 DATA 032,032,032,032,032,032,032,032,
042,042,084
150 DATA 032,082,065,077,032,084,069,083,
084,032,128
160 DATA 042,042,013,032,032,032,032,032,
032,032,065
170 DATA 032,032,032,032,068,073,032,080,
046,032,203
180 DATA 065,071,079,083,084,073,078,073,
013,013,120
190 DATA 013,032,081,085,069,083,084,079,
032,080,126
200 DATA 082,079,071,082,065,077,077,065,
032,067,185
210 DATA 079,078,084,082,079,076,076,065,
032,067,206
220 DATA 072,069,032,076,065,032,013,032,
077,069,025
230 DATA 077,079,082,073,065,032,082,065,
077,032,152
240 DATA 070,085,078,090,073,079,078,073,
032,083,229
250 DATA 069,078,090,065,032,069,082,082,
079,082,216
260 DATA 073,046,013,032,068,065,080,080,
082,073,100
270 DATA 077,065,032,076,069,032,076,079,
067,065,126

```

```

280 DATA 090,073,079,078,073,032,068,073,
032,077,163
290 DATA 069,077,079,082,073,065,032,013,
032,068,078
300 DATA 065,032,050,048,052,056,032,065,
032,052,228
310 DATA 048,057,053,057,032,086,069,078,
071,079,118
320 DATA 078,079,032,082,073,069,077,080,
073,084,215
330 DATA 069,032,013,032,067,079,078,032,
085,078,053
340 DATA 032,086,065,076,079,082,069,032,
078,085,172
350 DATA 077,069,082,073,067,079,044,032,
080,079,170
360 DATA 073,032,083,073,032,013,032,067,
079,078,050
370 DATA 084,082,079,076,076,065,032,067,
072,069,190
380 DATA 032,084,065,076,069,032,086,065,
076,079,152
390 DATA 082,069,032,083,073,065,032,013,
032,069,038
400 DATA 070,070,069,084,084,073,086,065,
077,069,235
410 DATA 078,084,069,032,080,082,069,083,
069,078,212
420 DATA 084,069,032,078,069,076,076,065,
032,077,146
430 DATA 069,077,079,082,073,065,046,013,
032,065,089
440 DATA 084,084,069,078,068,069,082,069,
044,032,167
450 DATA 080,082,069,071,079,046,013,141,
169,032,014
460 DATA 133,251,160,255,162,255,202,208,
253,136,223

```

## COMMODORE 64

## Seguito listato 1.

470 DATA 208,248,198,251,165,251,208,240,  
160,004,141  
480 DATA 169,000,153,251,000,200,192,005,  
208,248,146  
490 DATA 169,147,032,210,255,032,119,194,  
032,032,198  
500 DATA 032,032,032,032,032,032,040,095,  
032,076,179  
510 DATA 079,067,065,090,073,079,078,069,  
032,083,203  
520 DATA 079,084,084,079,032,084,069,083,  
084,041,207  
530 DATA 128,169,255,141,116,194,032,154,  
194,169,016  
540 DATA 170,141,116,194,032,154,194,169,  
085,141,116  
550 DATA 116,194,032,154,194,169,000,141,  
116,194,030  
560 DATA 032,154,194,169,147,032,210,255,  
173,117,203  
570 DATA 194,208,117,032,119,194,013,013,  
032,084,238  
580 DATA 069,083,084,032,084,069,082,077,  
073,078,219  
590 DATA 065,084,079,032,083,069,078,090,  
065,032,165  
600 DATA 069,082,082,079,082,073,032,013,  
013,032,045  
610 DATA 083,080,069,071,078,069,082,069,  
032,073,194  
620 DATA 076,032,067,079,077,080,085,084,  
069,082,219  
630 DATA 032,069,032,032,082,073,065,067,  
067,069,076  
640 DATA 078,068,069,082,076,079,032,013,  
032,080,097  
650 DATA 069,082,032,067,079,078,084,073,  
078,085,215  
660 DATA 065,082,069,032,073,076,032,076,  
065,086,144  
670 DATA 079,082,079,046,013,141,160,000,  
162,040,034  
680 DATA 169,042,032,210,255,202,208,248,  
096,000,182  
690 DATA 032,119,194,084,069,083,084,032,  
084,069,082  
700 DATA 082,077,073,078,065,084,079,046,  
013,013,098  
710 DATA 076,065,032,077,069,077,079,082,  
073,065,183  
720 DATA 032,082,065,077,032,082,073,083,  
085,076,175  
730 DATA 084,065,032,068,073,070,069,084,  
084,079,196  
740 DATA 083,065,046,013,141,076,044,194,  
000,000,150  
750 DATA 000,104,133,002,104,133,003,162,  
000,230,103  
760 DATA 002,208,002,230,003,161,002,041,  
127,032,040  
770 DATA 210,255,162,000,161,002,016,237,  
165,003,187  
780 DATA 072,165,002,072,096,000,234,032,  
175,194,018

790 DATA 032,098,195,032,185,194,032,175,  
194,032,145  
800 DATA 169,195,032,216,194,096,000,169,  
008,133,188  
810 DATA 252,169,000,133,251,096,000,160,  
000,173,210  
820 DATA 116,194,145,251,132,253,165,252,  
133,254,103  
830 DATA 032,249,194,164,253,200,016,237,  
230,252,035  
840 DATA 165,252,201,160,208,227,096,000,  
160,000,189  
850 DATA 132,253,165,252,133,254,177,251,  
205,116,146  
860 DATA 194,208,071,032,249,194,164,253,  
200,016,045  
870 DATA 235,230,252,165,252,201,160,208,  
225,096,232  
880 DATA 000,169,019,032,210,255,169,032,  
032,210,104  
890 DATA 255,169,036,032,210,255,165,254,  
032,019,147  
900 DATA 195,165,253,032,019,195,096,072,  
074,074,151  
910 DATA 074,074,032,030,195,104,041,015,  
201,010,008  
920 DATA 176,004,009,048,208,002,105,054,  
076,210,124  
930 DATA 255,096,000,000,032,119,194,019,  
017,017,237  
940 DATA 017,017,017,017,017,017,017,017,  
032,069,237  
950 DATA 082,082,079,082,069,032,078,069,  
076,076,213  
960 DATA 065,032,076,079,067,065,090,073,  
079,078,192  
970 DATA 069,032,032,160,169,255,141,117,  
194,032,177  
980 DATA 003,195,076,234,194,000,169,019,  
032,210,108  
990 DATA 255,162,005,169,017,032,210,255,  
202,208,235  
1000 DATA 248,032,119,194,032,082,073,069,  
077,080,238  
1010 DATA 073,077,069,078,084,079,032,077,  
069,077,203  
1020 DATA 079,082,073,065,032,082,065,077,  
032,067,142  
1030 DATA 079,076,032,086,065,076,079,082,  
069,032,164  
1040 DATA 068,073,032,013,160,169,036,032,  
210,255,024  
1050 DATA 173,116,194,032,019,195,096,169,  
019,032,021  
1060 DATA 210,255,162,005,169,017,032,210,  
255,202,237  
1070 DATA 208,248,032,119,194,032,067,079,  
078,070,103  
1080 DATA 082,079,078,084,079,032,077,069,  
077,079,224  
1090 DATA 082,073,065,032,082,065,077,032,  
067,079,142  
1100 DATA 076,032,086,065,076,079,082,069,  
032,068,153  
1110 DATA 073,032,032,032,013,160,169,036,  
032,210,021

potrebbero pregiudicare il corretto funzionamento dei programmi.

Come si effettua il test della memoria RAM? Semplicemente memorizzando nelle singole celle di memoria un valore numerico e accertandosi poi che tale valore numerico sia effettivamente presente in esse. Se una delle celle presenta un valore diverso, essa è chiaramente difettosa. Il programma inserisce nei byte componenti la memoria RAM, dei pattern di byte ben precisi che consentono di effettuare un test di ogni bit sia allo stato logico 0 che allo stato logico 1.

Generalmente i programmi che effet-

Numero esadecimale	Numero binario
\$ FF	11111111
\$ AA	10101010
\$ 55	01010101
\$ 00	00000000

tuano il test della RAM hanno il difetto di "scrivere" il valore nella cella di memoria e di "leggerlo" poi a pochi millisecondi di distanza: con tale metodo si possono accertare solamente gli errori più grossolani del sistema. Il programma presentato in queste pagine, invece riempie dapprima tutta la memoria RAM disponibile per poi passare a controllare le singole celle di memoria a parecchi secondi di distanza, per cui se un difetto

esiste, questo ha maggiori possibilità di manifestarsi sulla lunga distanza. Per "ritardare" l'azione di scrittura e successiva lettura del velocissimo linguaggio macchina è stata inserita una routine che stampa sullo schermo la locazione di memoria sotto test in codice esadecimale (che ha un'aria più "professionale")

Se il programma trova locazioni di memoria difettose, ne stampa il numero sullo schermo in codice esadecimale e alla fine del test stampa sullo schermo un messaggio indicante che la memoria RAM non ha superato il test.

Probabilmente il vostro computer lo supererà con facilità ma se qualcuno è curioso di vedere cosa succede qualora il computer abbia qualche difetto nella memoria RAM, si posizioni il programma in linguaggio macchina nella corretta zona di memoria e si diano i seguenti comandi in modo diretto:

POKE 49875,161: POKE 49908,161: SYS 49152 (RETURN)

Così facendo il programma tenta di "scrivere" anche nelle locazioni di memoria dell'interprete BASIC che sono notoriamente "read-only", per cui quando andrà a rileggerle troverà un valore differente da quello che ha tentato di inserire e darà l'avviso di errore.

Il listato presentato in queste pagine è un cosiddetto "BASIC Loader", vale a dire un programma sorgente in lin-

guaggio BASIC che ha tre funzioni ben definite. Il compito principale che tale programma si propone è quello di "pokare" nella porzione di memoria adatta i singoli valori costituenti le istruzioni in linguaggio macchina per il microprocessore 6510. La seconda funzione del programma in BASIC è quella dell'autocontrollo e - se trova un errore in una riga di dati ne informa immediatamente l'operatore - indicandogli quale è la riga che necessita di correzione. La terza e ultima funzione che adempie il programma in BASIC è quella di salvare sul supporto magnetico prescelto (nastro o disco) il solo programma in linguaggio macchina (mentre il BASIC Loader - se non viene espressamente salvato a cura dell'operatore - viene distrutto dallo stesso programma in linguaggio macchina che esso ha generato).

Una volta che il programma sia stato salvato su nastro o disco, potrà essere caricato ogni qualvolta lo si vorrà col comando Load "RAM test", 8,1 per il disco o Load "RAM test",1,1 per il nastro e verrà posto in funzione col comando Sys 49152 seguito da Return. ■

#### Seguito listato 1.

```

1120 DATA 255,173,116,194,032,019,195,096
,000,000,056
1130 FORI=49152TO50161STEP10:X=0
1140 FORJ=0TO9:READA:X=X+A:X=X AND 255
1150 POKEI+J,A:NEXTJ
1160 RI=PEEK(63)+PEEK(64)*256
1170 READZ:IFZ<>XTHENPRINT"ERRORE DATI RI
GA #";RI:STOP
1180 NEXTI
1190 PRINT"[<1CLR>][<2CRSR D>]I DATI SONO
CORRETTI."
1200 PRINT"[<1CRSR D>]SALVO SU NASTRO O D
ISCO (N/D)? ";
1210 GETA$:IFA$<>"N"ANDA$<>"D"THEN1210
1220 IFA$="N"THENA$="NASTRO":ND=1
1230 IFA$="D"THENA$="DISCO ":ND=8
1240 PRINTCHR$(18);A$
1250 REM ROUTINE DI SALVATAGGIO
1260 POKE 781,ND : SYS(65466)

```

```

1270 :
1280 NM$="RAMTEST":L=LEN(NM$)
1290 FORC=1TOL:POKE680+C,ASC(MID$(NM$,C))
:NEXT
1300 POKE 780,L
1310 POKE 781,681 AND 255
1320 POKE 782,681/256
1330 SYS(65469)ZZZCLOSECLOSECLOSECLOSE
CLOSECLOSECLOSECLOSECLOSECLOSECLOSE
LOSE

```

## COMMODORE 64

Listato 2 - Il programma in Assembly.

```

LINE# LOC CODE LINE
00001 0000 ;RAMTEST
00002 0000 JSR $OUT=$FFD2
00003 0000 AUX=$FB
00004 0000 AUX1=$02
00005 0000 #=$C000
00006 0000 ;PULISCE LO SCHERMO
00007 0000 LDA #93
00008 0002 20 D2 FF
00009 0005 20 D2 FF
00010 0007 8D 74 C2
00011 000A 8D 74 C2
00012 000D ;STAMPA UN MESSAGGIO
00013 000D JSR PRINT
00014 0010 .BYTE $0D
00015 0011 20 20
00016 0011 20 20
00017 0017 20 20
00018 0018 20 20
00019 0018 20 20
00020 0019 20 51
00021 0019 20 4D
00022 0019 20 44
00023 0019 20 44
00024 0019 20 44
00025 0019 20 44
00026 0019 20 43
00027 0019 20 43
00028 0019 20 43
00029 0019 20 43
00030 0019 20 45
00031 0019 20 45
00032 0019 20 41
00033 0019 20 41
00034 0019 20 41
00035 0019 20 41
00036 0019 20 41
00037 0019 20 41
00038 0019 20 41
00039 0019 20 41
00040 0019 20 41
00041 0019 20 41
00042 0019 20 41
00043 0019 20 41
00044 0019 20 41
00045 0019 20 41
00046 0019 20 41
00047 0019 20 41
00048 0019 20 41
00049 0019 20 41
00050 0019 20 41
00051 0019 20 41
00052 0019 20 41
00053 0019 20 41
00054 0019 20 41
00055 0019 20 41
00056 0019 20 41
00057 0019 20 41
00058 0019 20 41
00059 0019 20 41
00060 0019 20 41
00061 0019 20 41
00062 0019 20 41
00063 0019 20 41
00064 0019 20 41
00065 0019 20 41
00066 0019 20 41
00067 0019 20 41
00068 0019 20 41
00069 0019 20 41
00070 0019 20 41
00071 0019 20 41
00072 0019 20 41
00073 0019 20 41
00074 0019 20 41
00075 0019 20 41
00076 0019 20 41
00077 0019 20 41
00078 0019 20 41
00079 0019 20 41
00080 0019 20 41
00081 0019 20 41
00082 0019 20 41
00083 0019 20 41
00084 0019 20 41
00085 0019 20 41
00086 0019 20 41
00087 0019 20 41
00088 0019 20 41
00089 0019 20 41
00090 0019 20 41
00091 0019 20 41
00092 0019 20 41
00093 0019 20 41
00094 0019 20 41
00095 0019 20 41
00096 0019 20 41
00097 0019 20 41
00098 0019 20 41
00099 0019 20 41
00100 0019 20 41
00101 0019 20 41
00102 0019 20 41
00103 0019 20 41
00104 0019 20 41
00105 0019 20 41
00106 0019 20 41
00107 0019 20 41
00108 0019 20 41
00109 0019 20 41
00110 0019 20 41
00111 0019 20 41
00112 0019 20 41
00113 0019 20 41
00114 0019 20 41
00115 0019 20 41
00116 0019 20 41
00117 0019 20 41
00118 0019 20 41
00119 0019 20 41
00120 0019 20 41
00121 0019 20 41
00122 0019 20 41
00123 0019 20 41
00124 0019 20 41
00125 0019 20 41
00126 0019 20 41
00127 0019 20 41
00128 0019 20 41
00129 0019 20 41
00130 0019 20 41
00131 0019 20 41
00132 0019 20 41
00133 0019 20 41
00134 0019 20 41
00135 0019 20 41
00136 0019 20 41
00137 0019 20 41
00138 0019 20 41
00139 0019 20 41

```





**È** il classico tabellone elettronico, simile all'ormai famoso VisiCalc o all'omonimo programma su firmware del fratello maggiore Plus/4.

Questo programma vi consente di avere a disposizione una matrice su cui inserire dei dati, numerici ed alfanumerici (label), ed elaborare delle formule.

Il video è una finestra del tabellone nella quale vengono visualizzate 17 righe per 3 colonne numeriche ed una quarta colonna sul lato sinistro con tre righe sul lato superiore contenenti le label alfanumeriche.

La prima casella in alto a sinistra viene invece utilizzata per ricordare con quale comando state operando; mentre le ultime quattro righe dello schermo sono adibite ai messaggi del sistema ed all'input dei dati.

### Come utilizzare il programma

Dopo aver dato il Run, vi verrà chiesto se dovete caricare dei dati in precedenza elaborati e salvati su un file, altrimenti occorrerà dimensionare il tabellone: al massimo 50 righe per 15 colonne con un minimo di 17 righe per 3 colonne. Le misure massime sono dettate dalle limitazioni imposte dai 12 Kbyte di memoria a disposizione, mentre il formato minimo è quello relativo alla finestra del tabellone sul video.

Dopo il suddetto dimensionamento, il C 16 provvederà a visualizzare la griglia con le prime 17 righe e le prime 3 colonne a "0". Dopo di che appariranno sul lato inferiore dello schermo: "Spreadsheet Analysis - System Ready" e "Premi: I-D-C-G-H-S-B-E- (Muovi Coi Cursori)" ed il C 16 sarà pronto ad elaborare i vostri calcoli.

Per accedere ai comandi disponibili, basterà, a questo punto, premere il tasto corrispondente alla iniziale del comando desiderato, per passare alla relativa routine.

I comandi sono i seguenti.

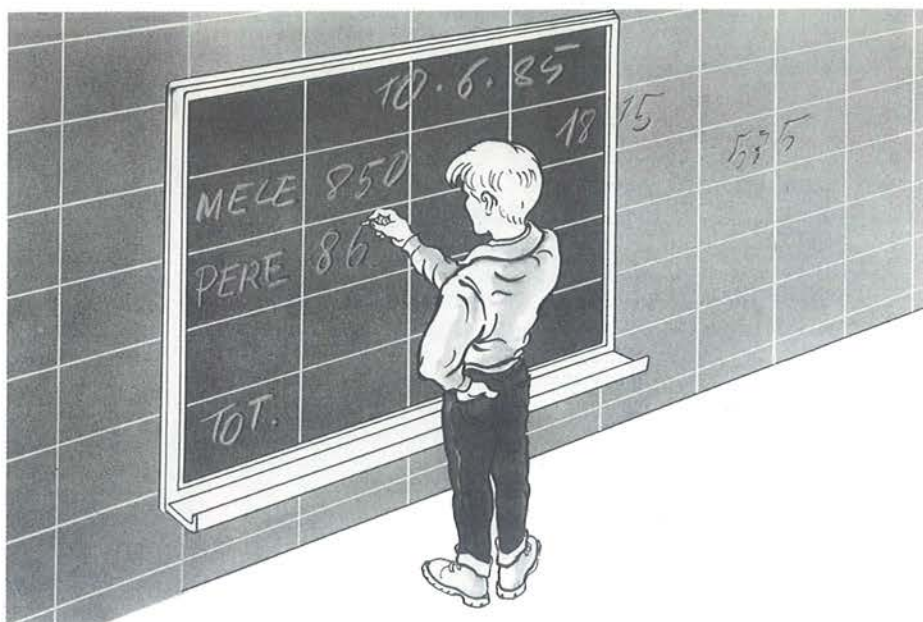
**Help** - Visualizza, per intero, sulle ultime quattro righe del video, tutti i comandi disponibili. Premere un tasto qualsiasi per ritornare al ciclo principale.

**Blank** - Consente di vuotare una casella numerica del tabellone, lasciandola completamente bianca (senza che appaia alcun dato, e nemmeno lo "zero"). E' possibile utilizzare il comando anche

# Super Spreadsheet

Implementiamo anche sul nostro C 16 un classico tabellone elettronico.

di Angelo Motta



per caselle non visualizzate sulla finestra del video.

**Insert** - Serve ad inserire i dati numerici o alfanumerici, e le formule per i calcoli. Dopo aver premuto il tasto I, apparirà in fondo allo schermo la scritta "Numero-Label-Formula". Premere l'iniziale di ciò che si intende inserire e procedere secondo quanto successivamente richiesto dal programma.

Bisogna tenere ben presente il modo di inserimento delle formule poiché per l'elaborazione di queste ultime, è stato utilizzato un metodo del tutto originale, più avanti illustrato nel comando Compute. Prima di spiegare il metodo da utilizzare per l'inserimento delle formule, occorre

fare una breve premessa su come vengono memorizzati i dati numerici del tabellone. All'inizio del programma è stata dimensionata la variabile A (numero righe, numero colonne) che rappresenta il formato scelto (si veda anche l'elenco delle variabili utilizzate).

Pertanto, la casella 00A è rappresentata dalla variabile A(00,0); la 01A dalla A(01,0); la 00B dalla A(00,1) e così di seguito sino alla A(numero righe-1, numero colonne-1) (numero righe e numero colonne -1 in quanto il dimensionamento inizia dallo zero). Le formule vanno inserite utilizzando, quale identificatore della casella, la rispettiva variabile della matrice, sostituendo alla vir-

gola l'apostrofo, poiché, in fase di Input, tale carattere non viene riconosciuto se non come separatore di dati.

Esempio: se vogliamo che la casella 03A sia uguale alla somma della 00A più la 01A, dovremo inserire la seguente formula: "A(03'0)=A(00'0)+A(01'0)".

Altro esempio: se la casella 00B deve contenere la percentuale della 00A rispetto alla 04A, dovremo inserire: "A(00'1)=A(00'0)/A(04'0)\*100".

Si ricorda che la lettera A della prima colonna corrisponde allo 0, la B all'1 e via di seguito.

Il metodo è abbastanza semplice: dopo aver inserito alcune formule ci si accorgerà quanto sia facile utilizzare il pro-

gramma.

Possono essere utilizzate le quattro operazioni (+ - \* /) e l'elevazione di potenza "↑".

La lunghezza massima accettata di una formula è di 60 caratteri; nel caso in cui ve ne occorresse una più lunga, la stessa dovrà essere divisa in due. Come per il comando Blank è possibile eseguire inserimenti anche per caselle non visualizzate sulla finestra.

**Delete** - Consente di azzerare una cella del tabellone e, in analogia ai comandi precedentemente visti, è utilizzabile anche per caselle non presenti sul video.

**Goto** - Muove la finestra e fa apparire nella prima casella numerica la cella

indicata nella richiesta. Il sistema effettua comunque un controllo e se le rimanenti righe e/o colonne del tabellone non sono sufficienti a riempire totalmente la finestra, sposta la casella richiesta di tante righe e/o colonne in modo da poter riempire totalmente il video.

**Compute** - Manda in esecuzione le formule contenute nella matrice R\$(R) ed aggiorna il tabellone.

Come precedentemente annunciato, è stato utilizzato il seguente metodo che consente di rendere operative le formule contenute in stringhe e nello stesso tempo avere un notevole risparmio di memoria ed una buona velocità di esecu-

#### Listato 1 - Il programma Super Spreadsheet.

##### TABELLA DI CONVERSIONE

```
{HOME}.....HOME
{CLR}.....PULIZIA SCHERMO
{CD}.....CURSORE IN BASSO
{CR}.....CURSORE A DESTRA
{CU}.....CURSORE IN ALTO
{CL}.....CURSORE A SINISTRA
{SPC}.....SPAZIO
{RVS ON}.....REVERSE ON
{RVS OFF}.....REVERSE OFF
{FLASH ON}.....FLASH ON
{FLASH OFF}.....FLASH OFF
{INST}.....INSERT
{WHITE}.....COLORE BIANCO
{RED}.....COLORE ROSSO
{GREEN}.....COLORE VERDE
{BLUE}.....COLORE BLU
{ORANGE}.....COLORE ARANCIO
{BLACK}.....COLORE NERO
{BROWN}.....COLORE MARRONE
{LT.GREEN}.....COLORE VERDE CHIARO
{LT.BLUE}.....COLORE BLU CHIARO
{PURPLE}.....COLORE PORPORA
{YELLOW}.....COLORE GIALLO
{CYAN}.....COLORE CIANO
{YL-GRN}.....COLORE GIALLO-VERDE
{PINK}.....COLORE ROSA
{BL-GRN}.....COLORE BLU-VERDE
{D.BLUE}.....COLORE BLU SCURO
```

I CARATTERI GRAFICI, OTTENUTI CON LA PRESSIONE DEI TASTI 'SHIFT' E 'CBM', SONO CODIFICATI IN MODO DA INDICARE IL TASTO DA PREMERE ASSIEME A 'SHIFT' O

'CBM'. ES. IL CUORICINO E' CODIFICATO CON {SH S}.

IL NUMERO DENTRO LE PARENTESI INDICA LE VOLTE CHE IL TASTO VA PREMUTO.

```
100 GOTO1000
500 ::::::::::::::::::::::::::::::::::::
      ::::::::::::::::::::::::::::::::::::RETURN
1000 COLOR0,13,5:COLOR4,13,5:DIMR$(200)
1010 PRINT"{CLR}":CHAR,9,1,"{BLACK}SPREADSHEET ANALYSIS"
1020 PRINT:INPUT"{CD}{D.BLUE}DEVI CARICARE UN FILE (S/N)";Q$
1030 IFQ$="S"THEN10000
1040 INPUT"{2*CD}{BROWN}NUMERO COLONNE (DA 3 A 15)";NC
1050 IFNC<3ORNC>15THEN1040
1060 INPUT"{CD}NUMERO RIGHE (DA 17 A 50)";NR
1070 IFNR<17ORNR>50THEN1060
1080 NR=NR-1:NC=NC-1:R=0
1090 DIMA(NR,NC),X$(NR),Y$(NC)
1110 FORI=0TONR:X$(I)="":FORJ=0TONC:A(I,J)=1E36:NEXTJ,I
1120 FORI=0TONC:Y$(I)="":NEXT
2000 COLOR1,1:PRINT"{CLR}":FORI=1TO38:POKE3072+I,67:POKE3152+I,67:POKE3872+I,67:NEXT
2010 FORI=40TO760STEP40:POKE3072+I,66:POKE3081+I,66:POKE3084+I,66
2020 POKE3093+I,66:POKE3102+I,66:POKE3111+I,66:NEXT
2030 POKE3072,112:POKE3081,114:POKE3084,114:POKE3093,114:POKE3102,114:POKE3111,110
2040 POKE3152,107:POKE3161,91:POKE3164,91:POKE3173,91:POKE3182,91:POKE3191,115
```

## COMMODORE 16

zione. All'inizio del programma è stata inserita, dopo la linea 100 Goto 1000, la linea 500 composta da 62 caratteri ":" (due punti) e da un Return finale (ricordiamo che per il sistema operativo il ":" è il carattere utilizzato per separare più istruzioni su un'unica linea e non ha nessun altro effetto ai fini dell'esecuzione di un programma).

Il trucco sta nel sostituire ai caratteri ":" i caratteri della formula, contenuta nella stringa R, che deve essere elaborata (si tenga presente che gli operatori matematici non hanno il codice ASCII dei caratteri che li rappresentano - da ciò le istruzioni If contenute nel ciclo di conversione nella subroutine da 16000 in

poi). A questo punto basta mandare in esecuzione la linea 500, che di fatto è diventata una subroutine, per avere eseguito la formula inserita. Come sopra evidenziato, i caratteri ":" in più nella riga 500, non hanno nessun effetto e vengono saltati sino al raggiungimento della istruzione Return.

Con questo sistema si utilizza un'unica linea che conterrà, di volta in volta, tutte le formule presenti nel programma.

**Save** - Consente di salvare i dati del tabellone per poterli riutilizzare successivamente. Il caricamento dei dati viene eseguito nella fase iniziale del programma previa conferma nella richiesta del Load.

**End** - Fine lavoro.

**Tasti cursore** - Muovono la finestra/video di una casella nella direzione del tasto cursore premuto. Analogamente al comando Goto il sistema controlla se sono stati raggiunti i bordi del tabellone, nel qual caso annulla l'effetto del comando.

## Avvertenze

Le caselle della finestra/video (sia numeriche che alfanumeriche) sono formate da otto caratteri e, pertanto, sia in fase di inserimento (Label) che in quella di stampa (Label e Numeri), il sistema

```

2050 POKE3872,109:POKE3881,113:POKE3884,
    113:POKE3893,113:POKE3902,113:POKE3
    911,125
2060 CHAR,10,1,"{RVS ON}LB":CHAR,2,2,"LA
    BEL{RVS OFF}"
2070 GOSUB4000
3000 M$="IDCHGBS{CL}{CR}{CU}{CD}":A$="":
    DUNTILA$="E"
3010 CHAR,2,22,"{YELLOW}SPREADSHEET ANAL
    YSIS - SYSTEM READY"
3020 CHAR,1,23,"PREMI:I-D-C-H-G-B-S-E-(T
    ASTI CURSORE)"
3030 TRAP3800:GETKEYA$:Z=INSTR(M$,A$):GO
    SUB5000
3040 IFZ<8THEN3080
3050 X=X-(A$="{CU}"ANDX+16<=NR)+(A$="{CD
    }"ANDX>0)
3060 Y=Y-(A$="{CL}"ANDY+2<=NC)+(A$="{CR}
    "ANDY>0)
3070 GOSUB4000:GOTO3010
3080 ONZGOSUB8000,9000,16000,6000,14000,
    15000,7000
3090 LOOP
3100 COLOR0,1:COLOR4,1:COLOR1,3:PRINT"{C
    LR}"
3110 CHAR,9,10,"{RED}SPREADSHEET ANALYSI
    S"
3120 CHAR,12,12,"SYSTEM CLOSED"
3130 END
3800 GOSUB5000:CHAR,0,22,"":PRINTERR$(ER
    ),EL
3810 GOSUB12010:RESUME3000
4000 PRINT"{BLACK}":CHAR,13,1,Y$(Y):CHAR
    ,22,1,Y$(Y+1):CHAR,31,1,Y$(Y+2)
4010 PRINT"{BLACK}":CHAR,13,1,Y$(Y):CHAR
    ,22,1,Y$(Y+1):CHAR,31,1,Y$(Y+2)
4020 POKE3172,(Y+129):POKE3181,(Y+130):P

```

```

    OKE3190,(Y+131)
4030 FORI=0TO16:CHAR,1,3+I,"{BLACK}"+X$(
    X+I)
4040 Z$=STR$(X+I):IFLEN(Z$)=2THENZ$="0"+
    RIGHT$(Z$,1)
4050 CHAR,10,3+I,"{RVS ON}"+RIGHT$(Z$,2)
    +"{RVS OFF}":W=Y:GOSUB5500
4060 CHAR,13,3+I,Z$:W=Y+1:GOSUB5500
4070 CHAR,22,3+I,Z$:W=Y+2:GOSUB5500
4080 CHAR,31,3+I,Z$:NEXT:RETURN
5000 CHAR,1,1," "
5020 FORN=3912TO4072:POKEN,32:NEXT:RETUR
    N
5500 Z$=STR$(ABS(INT(A(X+I,W))))
5510 IFLEN(Z$)<8THENZ$=" "+Z$:GOTO5510
5520 Z$=RIGHT$(Z$,8)
5530 IFA(X+I,W)>1E35THENZ$=" "
5540 IFA(X+I,W)<0THENZ$="{RED}"+Z$:ELSEZ
    $="{BLACK}"+Z$
5550 RETURN
6000 CHAR,1,1,"{YELLOW}{RVS ON}**HELP**{
    RVS OFF}"
6010 CHAR,0,21,"{RVS ON}COMANDI{RVS OFF}
    : {RVS ON}I{RVS OFF}NSERT-{RVS ON}D
    {RVS OFF}ELETE-{RVS ON}C{RVS OFF}OM
    PUTE-{RVS ON}H{RVS OFF}ELP-{RVS ON}
    S{RVS OFF}AVE"
6020 CHAR,0,22,"{RVS ON}G{RVS OFF}OTO-{R
    VS ON}B{RVS OFF}LANK-{RVS ON}E{RVS
    OFF}ND-({RVS ON}MUOVI COI TASTI CUR
    SORE{RVS OFF})"
6030 CHAR,0,24,"PREMI UN TASTO PER CONTI
    NUARE"
6040 GETKEYA$:GOTO5000
7000 INPUT"{CLR}{2*CD}{BLACK}NOME DEL FI
    LE DA REGISTRARE:";F$
7010 INPUT"{CD}IL NOME E' ESATTO (S/N)";

```

provvede automaticamente a controllare la lunghezza.

Le label superiori agli otto caratteri vengono ridotte a tal misura, mentre per quanto riguarda i dati numerici, pur mantenendo inalterato il valore degli stessi, viene visualizzata soltanto la parte intera sino ad un massimo di otto cifre (dall'unità alle decine di milioni - max 99.999.999). Esempio: se inserite il numero 1,35 il sistema visualizzerà solamente l'1; se invece inserite 123456789, vedrete solamente le ultime otto cifre e precisamente 23456789.

Nel primo caso basterà inserire una formula che moltiplichi per 100 il valore della casella per vederne anche la parte decimale; nel secondo caso dovrete dividere il valore della casella per 10. Va comunque segnalato che, a parte la limitazione per la parte decimale dei numeri, valori superiori ai 99.999.999 non sono poi di così facile utilizzo.

- I numeri sono visualizzati in nero se positivi e rosso se negativi; pertanto, a meno che non abbiate una stampante a colori, dovrete effettuare una routine di stampa che provveda a far comparire il segno "-" (meno) davanti ai numeri negativi.

- Per ritornare al menu da un qualsiasi comando chiamato, basta inserire XXX alla richiesta posta dal comando stesso.

- Le formule vengono elaborate in ordine al loro inserimento; pertanto ricordarsi di evitare di inserire formule il cui calcolo dovrà essere svolto dopo l'elaborazione di eventuali formule inserite successivamente. A tal riguardo si provveda con il comando Insert - opzione Modifica Formula, a variarne l'ordine di esecuzione.

- Nel ciclo principale è stata inserita l'istruzione Trap 3800 che provvede ad intercettare eventuali errori rimandando l'esecuzione del programma alle linee 3800-3810 che segnalano, nelle ultime righe dello schermo, il tipo di errore commesso e la linea del programma dove si è verificato. Nel caso in cui il programma dovesse bloccarsi (ad esempio basta premere il tasto Stop due volte consecutivamente) basterà digitare in modo diretto l'istruzione Goto 2000 per farlo ripartire mantenendo inalterati i dati sino a qual momento presenti.

Per risparmiare spazio in memoria e dar posto ad un numero maggiore di formule da inserire, sono state omesse tutte le

Rem descrittive dei vari blocchi componenti il programma. Ecco di seguito l'analisi dello stesso.

## REMARKS

**500** - Linea importantissima, composta da 62 caratteri ":" (duepunti) più un Return. Consente l'elaborazione delle formule presenti nel tabellone. Da copiare tassativamente così com'è composta, altrimenti potrebbero verificarsi malfunzionamenti nel programma.

**1000-1120** - Inizializzazione del tabellone: viene richiesto se devono essere caricati dati già precedentemente elaborati, nel qual caso rimanda alla routine di Load alle linee 10000, o altrimenti chiede le dimensioni del tabellone (min 3 x 17 e max 15 x 50). Le misure massime sono dettate dal limite dei 12 Kbyte di memoria disponibile; mentre le minime dal formato della finestra del video. Se in futuro verranno annunciate espansioni di memoria, tali limiti massimi potranno essere superati.

Vengono inoltre dimensionate le matrici contenenti i numeri del tabellone [A(righe/colonne)] e quelle contenenti le label alfanumeriche [X\$ (righe) e Y\$ (colonne)]. Nella prima viene inserito il numero 1E36 che produce il Blank in sede di stampa a video; nelle seconde vengono invece inseriti 8 spazi da utilizzarsi come maschera in fase di scrolling del tabellone.

Infine vengono posti a zero il contatore delle formule (R) ed i puntatori di stampa a video della finestra (X e Y).

La matrice R\$ (200), dimensionata nella linea 1000, conterrà le formule che nel corso del programma verranno inserite. Tale dimensionamento a 201 elementi è del tutto proforma e fatto esclusivamente per evitare errori nel corso del programma in quanto il numero delle formule che il programma può contenere, dipende sia dalla grandezza del tabellone che dalla lunghezza della formula stessa. Indicativamente, con un tabellone di 15 x 50 si possono inserire circa 100 formule.

**2000-2070** - Viene stampata la griglia della finestra e chiama la subroutine alla linee 4000-4080 che provvede a riempire la finestra con i valori indicati a partire dai puntatori (X e Y).

**3000-3138** - Ciclo principale. Viene creato un loop dal quale è possibile uscirne

solamente inserendo la "E" (End) quale comando. La linea 3040 provvede a ricercare all'interno della stringa M\$ l'iniziale del comando scelto e ne restituisce il numero corrispondente alla posizione, che viene assegnato alla variabile Z. In base al valore di questa variabile vengono chiamate le subroutine ai vari comandi disponibili (linee 3090) o, se premuto un tasto cursore, viene ruotata la finestra/video, previo controllo dei bordi: controllo effettuato con i soliti operatori relazionali alle linee 3060-3070.

**3800-3810** - Routine che viene eseguita in presenza di un errore nel corso del programma, intercettato dal ciclo principale con l'istruzione Trap inserita. Viene evidenziato sul video il tipo di errore commesso ed in quale linea si è verificato.

**4000-4080** - Subroutine di stampa dati nella finestra/video.

**5000-5020** - Subroutine che effettua la cancellazione a video della stringa contenente il comando appena utilizzato, e delle ultime quattro righe dello schermo.

**5500-5550** - Subroutine che tramuta un dato numerico della matrice del tabellone in stringa per consentire la stampa con incolonnamento sulla destra. La stringa viene posta ad otto caratteri (formato della casella) e viene stampata in nero se il numero è positivo o rosso se negativo.

Viene utilizzato questo sistema di incolonnamento, anziché l'istruzione Print Using, poiché il numero da visualizzare deve essere come massimo di otto lettere e, in caso di numero con lunghezza superiore, comparirebbero degli asterischi, usando il suddetto comando.

**6000-6040** - Subroutine che effettua il comando Help. Vengono visualizzate nelle ultime quattro righe dello schermo i comandi a disposizione.

**7000-7110** - Subroutine che effettua il Save dei dati presenti nel tabellone. Viene prima di tutto richiesto il nome del file da registrare e la conferma dello stesso; quindi viene mandato in esecuzione il comando e la richiesta, alla fine del salvataggio, se si vuol proseguire con il tabellone. La routine è stata predisposta per essere utilizzata con il registratore; chi disponesse del drive dovrà opportunamente variarla.

**8000-8520** - Subroutine di Insert. All'inizio viene richiesto se l'input deve essere un Numero, Label o Formula e quindi, in

## COMMODORE 16

base alla richiesta, si passa ad una delle tre sezioni che compongono la routine. Nella prima, riguardante i dati numerici, il sistema chiede la casella, ne controlla l'esistenza, e quindi viene richiesto il valore da inserire che viene immediatamente visualizzato se la casella è presente sulla finestra.

Nella seconda vengono invece inserite le label (sia di riga che di colonna) che il sistema provvede automaticamente a portare ad otto caratteri, inserendo degli spazi per i dati inferiori.

Nella terza vengono invece inserite le formule ed il programma chiede se l'inserimento riguarda una nuova formula o una modifica di una già esistente. Se si

tratta di una nuova, basta digitare la formula tenendo presente le modalità di inserimento precedentemente riportate nell'istruzione Insert: per quanto riguarda invece le modifiche è possibile cancellare una formula precedentemente inserita, o modificarla.

Con il comando Insert è possibile ottenere lo stesso effetto dei comandi Blank e Delete. Nel primo caso basta chiedere l'opzione Numero ed al valore della casella inserire "0"; nel secondo caso, sempre con l'opzione Numero, alla richiesta del valore dovrà essere inserito un numero superiore a 1E35. Sta di fatto, comunque, che i relativi comandi Blank e Delete sono più veloci da eseguirsi.

**9000-9030** - Subroutine che provvede ad effettuare il comando Delete.

**10000-10080** - Routine di Load di un file di dati precedentemente elaborati, al termine della quale il sistema salta alla istruzione 2000 provvedendo alla stampa della finestra/video. Come precedentemente evidenziato per il Save, la routine è predisposta per il registratore.

**11000-11020** - Subroutine chiamata dai comandi Insert-Delete-Goto-Blank che controlla la validità della casella scelta.

**12000-12010** - Subroutine che viene chiamata quando una casella scelta (si veda subroutine precedente) risulta non valida e fa apparire la scritta "Casella inesistente". Viene inoltre creato

## Seguito Ilistato Super Spreadsheet.

```

Q$
7020 IFQ$<>"S"THEN7010
7030 OPEN1,1,2,F$:PRINT#1,NC:PRINT#1,NR:
PRINT#1,R
7040 FORI=0TONR:FORJ=0TONC:PRINT#1,A(I,J
):NEXTJ,I
7050 FORI=0TONR:PRINT#1,X$(I):NEXT
7060 FORI=0TONC:PRINT#1,Y$(I):NEXT
7070 FORI=1TOR:PRINT#1,R$(I):NEXT
7080 PRINT#1:CLOSE1
7090 INPUT"{3*CD}{RED}CONTINUI (S/N):";Q
$
7100 IFQ$="S"THEN2000
7110 PRINT"{CLR}":GOTO3100
8000 CHAR,1,1,"{WHITE}{RVS ON}*INSERT*{R
VS OFF}"
8020 Q$="":DOUNTILQ$="XXX"
8030 CHAR,0,21,"{WHITE}{RVS ON}L{RVS OFF
}ABEL-{RVS ON}N{RVS OFF}UMERO-{RVS
ON}F{RVS OFF}ORMULA-(XXX=ESCE)":INP
UTQ$:GOSUB5020
8035 IFQ$<>"N"THEN8075
8040 CHAR,0,21,"CASELLA:":INPUTC$
8050 GOSUB11000:IFIN=1THENGOSUB12000:GOT
O8030
8060 CHAR,0,22,"VALORE:":INPUTC:A(X1,Y1)
=C
8070 GOSUB13000
8075 IFQ$<>"L"THEN8200
8080 CHAR,0,21,"CASELLA DEL LABEL:":INPU
TP$
8090 IFASC(P$)>64THEN8150
8100 P=VAL(P$)
8110 IFP<0ORP>NRTHENGOSUB12000:GOTO8030
8120 GOSUB8500

```

```

8130 X$(P)=LEFT$(P$,8):IFP>=XANDP<X+17TH
ENCHAR,1,3+P-X,"{BLACK}"+X$(P)
8140 GOTO8200
8150 P=ASC(P$)-65:IFP<0ORP>NCTHENGOSUB12
000:GOTO8030
8160 GOSUB8500:Y$(P)=LEFT$(P$,8)
8170 IFP>=YANDP<Y+3THENCHAR,13+9*(P-Y),1
,"{BLACK}"+Y$(P)
8200 DOWHILEQ$="F":IFERE(0)>150THEN8240
8210 CHAR7,22,"{FLASH ON}MANCA SPAZIO IN
MEMORIA{FLASH OFF}"
8220 PRINT"{3*CR}MODIFICA UNA FORMULA GI
A' INSERITA":GOSUB12010:Q$="XXX":EX
IT
8240 CHAR,0,21,"{RVS ON}M{RVS OFF}ODIFICA
A O {RVS ON}N{RVS OFF}UOVA FORMULA:
":INPUTP$:IFP$="M"THEN8280
8250 GOSUB5020:CHAR,0,21,"NUOVA FORMULA:
":INPUTZ$
8260 IFLEN(Z$)>60THENGOSUB5020:PRINT"{FL
ASH ON}TROPPO LUNGA!{FLASH OFF}":GO
SUB12010:GOTO8250
8270 R=R+1:R$(R)=Z$:EXIT
8280 IFR=0THENEXIT
8285 FORK=1TOR
8290 GOSUB5020:CHAR,0,23,R$(K):CHAR,0,21
,"{RVS ON}D{RVS OFF}ELETE-{RVS ON}P
{RVS OFF}ROSSIMA-{RVS ON}M{RVS OFF}
ODIFICA"
8295 GETKEYP$:IFP$="P"THEN8360
8300 IFP$<>"D"THEN8330
8310 FORJ=KTOR-1:R$(J)=R$(J+1):NEXT
8320 R$(R)="":R=R-1:GOTO8370
8330 INPUTZ$
8340 IFLEN(Z$)>60THENGOSUB5020:PRINT"{FL
ASH ON}TROPPO LUNGA!{FLASH OFF}":GOS
UB12010:GOTO8290"

```

# Ritorna in edicola

# VIDEO BASIC

Il corso più entusiasmante su cassetta  
del Gruppo Editoriale Jackson per Commodore 64,  
VIC 20 e Spectrum

## 200.000 copie vendute

del 1° fascicolo della prima edizione

Ogni lezione  
uno spettacolo

Con la 1ª lezione  
una cassetta giochi  
compresa nel prezzo



Il corso è composto da:  
**20 fascicoli** + (Quattordicinali)  
**20 cassette** +  
**5 splendidi raccoglitori**

Oggi è davvero facile imparare il Basic. Con Video Basic il corso su cassetta che ti permette di programmare subito il tuo computer. È facile: tu chiedi, lui risponde, tu impari. Passo dopo passo. Sul tuo schermo appaiono le domande, le risposte, gli esercizi e

tu, senza fatica, presto e bene, impari a conoscere e programmare il tuo computer, sia esso un VIC 20, un Commodore 64 o un Sinclair. Video Basic è in edicola. Provalo subito. Ogni lezione è uno spettacolo.

Oggi il Basic si impara così. Video Basic, il corso su cassetta per parlare subito col tuo computer.

Video Basic  
per imparare non solo il Basic.



Un'altra grande idea firmata  
**GRUPPO EDITORIALE JACKSON**

Milano • San Francisco • Londra • Madrid

# S T O P !



**AI REGISTRATORI  
CHE NON CARICANO !!  
ora c'è:**

**AZIMUTH  
CONTROLLER**



**PERMETTE A CHIUNQUE  
IN BREVE TEMPO, DI TARARE  
PERFETTAMENTE LE TESTINE  
DEL PROPRIO REGISTRATORE**

Distribuito da **SOFTY** via Longhi 10 20137 Milano

**ATTENZIONE!**

L'AZIMUTH CONTROLLER può essere acquistato anche per corrispondenza. Inviando L. 30.000 in contanti o in assegno, in lettera raccomandata munita del vostro indirizzo a: SOFTY Casella Postale 10 20090 Rodano Mi.  
N.B. Per pagamenti in contrassegno vi saranno addebitate L. 5000 per spese postali.

Cercasi Distributori per zone libere Tel. (02) 7383251 (4 linee ricerca automatica)

## COMMODORE 16

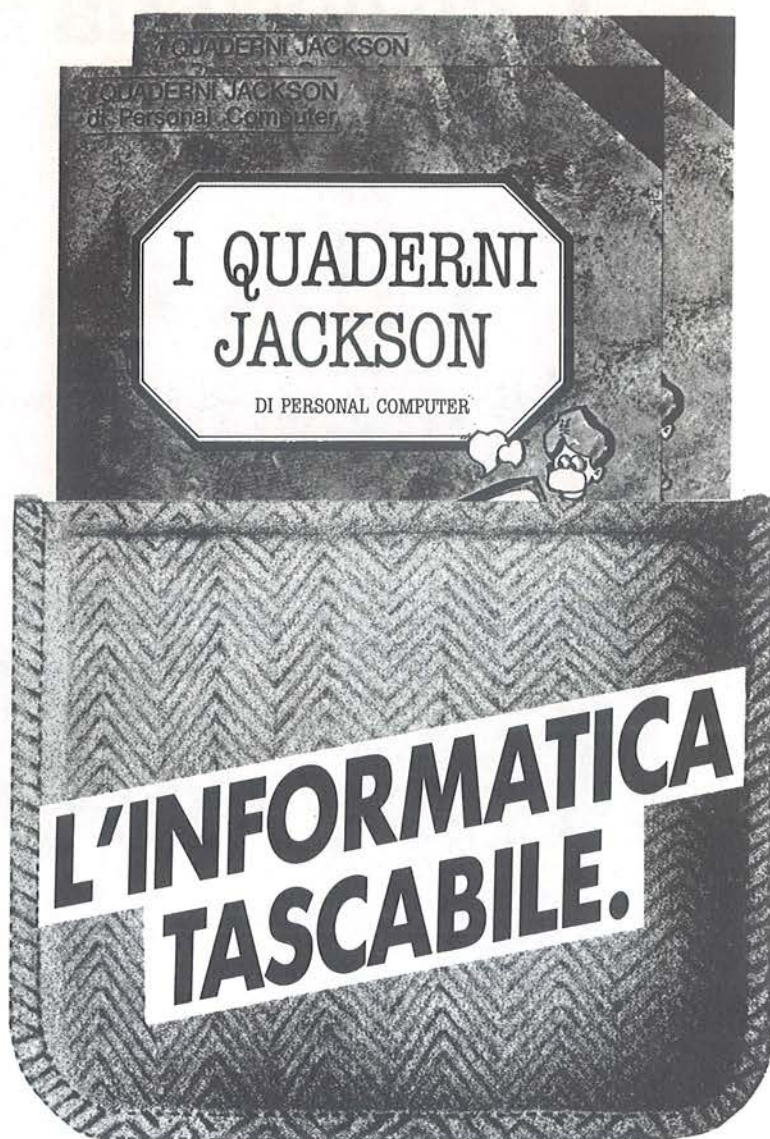
## Seguito Ilistato Super Spreadsheet.

```

8350 R$(K)=Z$
8360 NEXTK
8370 GOSUB5020:EXIT:LOOP:GOSUB5020:LOOP:
      GOTO5000
8500 INPUT"LABEL:";P$
8510 IFLEN(P$)<8THENP$=P$+" ":GOTO8510
8520 RETURN
9000 CHAR,1,1,"D.BLUE}{RVS ON}*DELETE*{
      RVS OFF}":CHAR,0,21,"CASELLA DA ELI
      MINARE(RIGA/COL-XXX=FINE)"
9010 INPUTC$:IFC$="XXX"THEN5000
9020 GOSUB11000:IFIN=1THENGOSUB12000:GOT
      O9000
9030 A(X1,Y1)=0:GOSUB5020:GOSUB13000:GOT
      O9000
10000 INPUT"{CD}NOME DEL FILE DA CARICAR
      E:";F$
10010 INPUT"{CD}IL NOME E' ESATTO (S/N)"
      ;Q$
10020 IFQ$<>"S"THEN10010
10030 OPEN1,1,0,F$:INPUT#1,NC:INPUT#1,NR
      :INPUT#1,R:DIMA(NR,NC),X$(NR),Y$(N
      C)
10040 FORI=0TONR:FORJ=0TONC:INPUT#1,A(I,
      J):NEXTJ,I
10050 FORI=0TONR:INPUT#1,X$(I):IFX$(I)="
      "THENX$(I)="      "
10055 NEXT
10060 FORI=0TONC:INPUT#1,Y$(I):IFY$(I)="
      "THENY$(I)="      "
10065 NEXT
10070 FORI=1TOR:INPUT#1,R$(I):NEXT
10080 CLOSE1:GOTO2000
11000 X1=VAL(LEFT$(C$,2)):Y1=ASC(RIGHT$(
      C$,1))-65:IN=0
11010 IFX1<0ORX1>NRORY1<0ORY1>NCTHENIN=1
11020 RETURN
12000 CHAR,0,24,"{BLACK}CASELLA INESISTE
      NTE!!!"
12010 FORI=1TO1000:NEXT:GOTO5020
13000 IFX1<XORX1>X+16ORY1<YORY1>Y+2THENR
      ETURN
13020 Z$=STR$(ABS(INT(A(X1,Y1))))
13030 IFLEN(Z$)<8THENZ$=" "+Z$:GOTO13030
13035 Z$=RIGHT$(Z$,8)
13040 IFA(X1,Y1)<0THENZ$="{RED}"+Z$:ELSE
      Z$="{BLACK}"+Z$
13050 IFA(X1,Y1)>1E35THENZ$="      "
13060 CHAR,13+9*(Y1-Y),3+X1-X,LEFT$(Z$,9
      )
13070 RETURN
14000 CHAR,1,1,"{YELLOW}{RVS ON}**GOTO**
      {RVS OFF}"
14010 CHAR,0,21,"NUOVA POSIZIONE(RIGA/CO
      L-XXX=ESCE):"
14020 INPUTC$:IFC$="XXX"THEN5000
14040 GOSUB11000:IFIN=1THENGOSUB12000:GO
      TO14000
14050 IFX1+16>NRTHENX1=NR^16
14060 IFY1+2>NCTHENY1=NC^2
14070 X=X1:Y=Y1:GOSUB5000:GOTO4000
15000 CHAR,1,1,"{GREEN}{RVS ON}**BLANK*{
      RVS OFF}"
15020 CHAR,0,21,"CASELLA DA VUOTARE (RIG
      A/COL-XXX=ESCE):"
15030 INPUTC$:IFC$="XXX"THEN5000
15040 GOSUB11000:IFIN=1THENGOSUB12000:GO
      TO15000
15050 A(X1,Y1)=1E36:GOSUB5020:GOSUB13000
      :GOTO15000
16000 CHAR,1,1,"{RED}{RVS ON}COMPUTE*{RV
      S OFF}"
16010 TRAP16500
16020 IFR=0THENCHAR,7,22,"{FLASH ON}NESS
      UNA FORMULA PRESENTE!{FLASH OFF}":
      GOTO16300
16030 FORI=1TO R:CHAR,5,22,"ATTENDI STO
      ELABORANDO I DATI"
16040 FORJ=1TOLEN(R$(I))
16050 K=ASC(MID$(R$(I),J))
16060 IFK=61THENK=178
16070 IFK=43THENK=170
16080 IFK=45THENK=171
16090 IFK=42THENK=172
16100 IFK=47THENK=173
16110 IFK=94THENK=174
16120 IFK=39THENK=44
16130 POKE4110+J,K:NEXT
16140 FORJ=LEN(R$(I))+1TO61:POKE4110+J,5
      8:NEXT
16170 GOSUB500
16180 NEXTI
16190 GOSUB4000:GOTO5000
16300 FORI=0TO1000:NEXT:GOTO5000
16500 GOSUB5020:CHAR,5,22,"{FLASH ON}FOR
      MULA N.":PRINTR"ERRATA-CONTROLLA"
16510 GOSUB12010:RESUME16190

```





Arrivano i Quaderni Jackson, tanti volumi monografici per conoscere bene il personal computer e l'informatica.

Nei quaderni Jackson c'è tutto quello che è importante sapere sui computer, la programmazione, i linguaggi, il software, le applicazioni e i nuovi sviluppi dell'informatica.

Quaderni Jackson: l'informatica a tutti i livelli, in una collana aperta, pratica, essenziale, aggiornata.

L'informatica tascabile per chi vuole saperne di più e compiere così un salto di qualità nel mondo di oggi e di domani.

Ogni mese, 2 volumi.

#### Volumi già pubblicati:

*Gianni Giaccagnini*

"Vivere col Personal Computer"

*Paolo Bozzola*

"Dentro e fuori la scatola"

*Enrico Odetti*

"Ed è subito BASIC Vol. I"

"Ed è subito BASIC Vol. II"

*Paolo Capobussi*

e *Marco Giacobazzi*

"A ciascuno il suo Personal"

*Fulvio Francesconi*

e *Fernando Paterlini*

"To do or not to do"

**In edicola,  
a sole lire 6.000.**



**GRUPPO  
EDITORIALE  
JACKSON**

SAN FRANCISCO - LONDRA - MILANO

**Leoni informatica**



### LISTINO PROGRAMMI COMMODORE 64

Programmi in configurazione base (\*)  
IVA esclusa

LINEA TOP	Prezzo
Contabilità Generale	300.000
Contabilità Forfettaria Visentini	300.000
Distinta Base	350.000
Condominio New	350.000
Gestione Farmacie	300.000
Multisystem - Sviluppo sistemi	
Totocalcio - Enalotto - Totip	350.000

PERSONALI	
Totocalcio a sviluppo colonnare T	80.000
Calcolo dell'Equo Canone T	80.000
Modello 740 T	100.000
Conto corrente	100.000
Gestione Conti Casa T	100.000
Totoplus T	100.000
Impariamo il Basic T	100.000
100 Programmi Basic	80.000
Rubrica telefonica T	100.000

GESTIONALI	
Anagrafiche	150.000
Appuntamenti	100.000
Mailing List	100.000
Scadenario effetti	100.000
Ordini (Cli/For)	100.000
Super Mail	180.000
Bolle e Fatture	210.000
Contabilità Fatture (Cli/For/Gen)	100.000
Magazzino e Fatturazione agganciati	200.000
Magazzino alfanumerico (600 art.)	200.000
Magazzino Grossisti (2500 art.)	280.000
Magazzino Dettaglio (2500 art.)	280.000
Magazzino Taglia/col. (2500 art.)	280.000

GESTIONI SPECIFICHE	
Agenti e Fatture Cli/For	150.000
Agenzie immobiliari	150.000
Studi Ottici	200.000
Studi Dentistici	200.000
Hotels e Pensioni	280.000
Gommisti	280.000
Officine	280.000
Tavole Calde	280.000
Scadenario premi e polizze	150.000
Librerie e Biblioteche	120.000
Studi Medici	200.000
Parrucchieri	280.000
Clubs Nautici	280.000
Ristoranti	280.000
Lavanderie	280.000
Campeggi	280.000

GESTIONE TESTI	
Word Processor III	100.000
Easy Script T (Commodore orig.)	75.000

TECNICI/SERVIZI	
Legge 373 (calcolo isolamenti termici)	100.000
Ingegneria civ. II (travi intelaiate)	100.000
Copia Disco Singolo	50.000
Ingegneria civ. I (calcoli strutt.)	100.000
Computi metrici	200.000
Kit allineamento testine	45.000

GESTIONE BANCHE DATI	
The Manager 64 (Commodore orig.)	100.000
Super Base (Commodore orig.)	175.000

(\*) Per configurazione base si intende: Commodore 64, floppy 1541, stampante Commodore a 80 colonne. I programmi segnati con la "T" sono disponibili anche su cassetta.

## COMMODORE 16

un ciclo di temporizzazione nella linea 12010 più volte chiamato nel corso del programma.

**13000-13070** - Controlla se la casella con la quale si sta operando con i comandi Insert-Blank o Delete, è presente sul video, nel qual caso provvede a far apparire il valore relativo al tipo di operazione effettuata.

**14000-14070** - Subroutine che effettua l'istruzione Goto.

**15000-15050** - Subroutine con il comando Blank.

**16000-16190** - Subroutine che provvede ad eseguire il comando Compute.

Viene dapprima creato un ciclo che provvede ad inserire le stringhe nella linea 500, rendendo tale riga operativa; a chiamarla come subroutine ed infine ad effettuare il calcolo. Al termine del ciclo viene chiamata la subroutine alle linee 4000 che provvede all'aggiornamento della finestra con i nuovi risultati. All'inizio di questa routine è stata posta l'istruzione Trap 16500 che intercetta qualsiasi errore nelle formule e lo segnala fra le linee 16500-16510 in modo che l'operatore possa chiamare l'istruzione Insert e controllare tale formula.

### Elenco delle variabili utilizzate

**NR** - Numero delle righe (da 17 a 50).

**NC** - Numero delle colonne (da 3 a 15).

**A(NR,NC)** - Matrice contenente il valore delle caselle numeriche del tabellone.

**XS(NR)** - Matrice contenente le label alfanumeriche delle righe.

**YS(NC)** - Matrice contenente le label alfanumeriche delle colonne.

**X** - Numero della riga da stampare alla prima posizione numerica della finestra.

**Y** - Numero della colonna da stampare alla prima posizione numerica della finestra.

**R** - Contatore delle formule.

**RS(R)** - Matrice contenente le formule per i calcoli.

**MS** - Stringa contenente le iniziali dei comandi a disposizione.

**X1** e **Y1** - Coordinate della casella in fase di elaborazione dati e comandi.

**IN** - Identificatore di errore nella scelta della casella. ■

# Che cosa ha in

## Libreria Pr Personal Con

# più il Personal Computer IBM?

ogrammi  
nputer **IBM**

## La Libreria Programmi Personal Computer IBM, per esempio.

Il tuo Concessionario IBM per il Personal Computer ha una novità per te: la Libreria Programmi Personal Computer IBM.

La Libreria Programmi Personal Computer IBM è uno strumento utilissimo, che ti permette di trovare nel modo più immediato i programmi firmati da IBM. Puoi guardarli e confrontarli per vedere subito, fra quelli di uno stesso settore applicativo, quale sia il programma che risponde meglio alle tue esigenze di lavoro.

Ma la Libreria Programmi Personal Computer IBM ti dà anche un aiuto in più: con ogni programma troverai la brochure che lo riguarda. Ce ne sono anche molte altre, che ti illustreranno tutto quello che il tuo Personal Computer IBM può fare e le sue applicazioni particolari, come il Videotel.

Puoi prenderle e portarle a casa, per leggerle in tutta tranquillità e pensare bene alla scelta che devi fare.

E non dimenticare che il Concessionario IBM (gli indirizzi sono sulle Pagine Gialle) è un vero esperto, che conosce perfettamente il Personal Computer IBM ed i problemi della tua attività. Potrà consigliarti nel modo migliore nella scelta dei programmi e ti illustrerà tutte le possibilità che il Personal Computer IBM ti offre, oltre alla grande versatilità delle sue prestazioni.

Per acquisto, consulenza e servizi, puoi anche rivolgerti al Negozio IBM Centromilano.

Allora, non ti pare che il Personal Computer IBM abbia veramente qualcosa in più?



Desidero ricevere:

- Informazioni sul Personal Computer IBM e i suoi programmi.
- Gli indirizzi dei Concessionari IBM Personal Computer della mia regione
- Una visita o dimostrazione pratica di un concessionario

Nome e Cognome .....

Azienda .....

La mia attività è .....

Indirizzo .....

Spedisci questo tagliando a: IBM Italia  
Direzione Entry Systems  
Casella Post. 137 - 20090 Segrate Milano

**IBM**

## Un po' di storia

**L'** I Ching o Libro dei Mutamenti è un sistema di oracoli che ha origini antichissime: come il nome fa intuire esso è nato in Cina. La tradizione attribuisce la base dell'I Ching ad un saggio, tal Pao-Hsi, vissuto la bellezza di 53 secoli fa. Durante la dinastia degli Yin, circa nell'undicesimo secolo avanti Cristo, il principe Wen e suo figlio Tan inserirono dei loro commenti nel Libro dei Mutamenti. Lo stesso Confucio, in seguito, studiò ed apprezzò l'I Ching. Ma vediamo che cosa ha a che fare tutto ciò con le pagine di una rivista d'informatica. L'I Ching, come detto, è una sorta d'oracolo che risponde alle domande di chi lo consulta: esso è formato dall'unione di due simboli, "--" e

# I Ching: il Libro dei Mutamenti

Interrogate il vostro Texas TI99/4A  
ed ascoltate i suoi consigli

di Daniela Cerù

### Listato 1 - Il programma I Ching.

```
5 DATA 3,4,9,10,13,14,16,17,19,2
0,23,24,26,27
10 DATA 13,14,19,20
15 DIM IC$(64)
20 CALL CLEAR
25 RANDOMIZE
30 C=INT(10*RND)+5
40 CALL COLOR(2,C,C)
50 FOR I=16 TO 17
60 CALL VCHAR(3,I,42,8)
70 NEXT I
80 FOR I=1 TO 14
85 READ R
90 CALL VCHAR(13,R,42,8)
100 NEXT I
110 FOR I=1 TO 4
115 READ R
120 CALL HCHAR(R,5,42,3)
125 CALL HCHAR(R,28,42,3)
130 NEXT I
135 CALL HCHAR(16,11,42,2)
140 CALL HCHAR(17,11,42,2)
145 CALL HCHAR(18,29,42,2)
150 CALL VCHAR(14,21,42,2)
155 CALL VCHAR(15,22,42,2)
160 INPUT " VUOI QUALCHE SPIEGAZ
IONE ? ":R$
170 IF R$="S" THEN 1000
175 IF R$="SI" THEN 1000
```

```
180 INPUT "COME TI CHIAMI ?":R$
185 CALL SCREEN(6)
187 CALL COLOR(2,2,6)
190 CALL CLEAR
200 PRINT "BENE ";R$
205 PRINT "Esponimi il tuo probl
ema"
210 PRINT
215 INPUT " ":P$
220 IF LEN(P$)<30 THEN 900
230 IF P$<>" " THEN 215
240 PRINT
245 PRINT "MMM...HO CAPITO"
250 PRINT "ORA ATTENDI UN ATTIMO
E..."
254 PRINT
255 PRINT
256 PRINT
270 FOR I=0 TO 5
275 RANDOMIZE
280 A(I)=INT(2*RND)
290 IF A(I)=0 THEN 300 ELSE 320

300 PRINT TAB(12); "*** **"
310 GOTO 330
320 PRINT TAB(12); "*****"
330 PRINT
340 FOR J=I*10 TO I*10+9
350 READ IC$(J)
370 NEXT J
380 NEXT I
```

## TEXAS TI99/4A

"-" combinati sei volte tra di loro. In tutto ci saranno 64 diverse combinazioni ( $2^6 = 64$ ) ed una di queste sarà la risposta alla domanda formulata.

### La sapienza cinese unita alla tecnologia del duemila

Il metodo di consultazione degli antichi cinesi era piuttosto complesso (una scaglia di tartaruga, parecchi steli d'achillea e così via) perciò noi abbiamo pensato di semplificarlo usando un computer.

Già, sarà proprio una di queste fredde macchine, aggregati di diodi e resistenze, di integrati e condensatori, di fili elettrici e di tasti, a dare la risposta ai quesiti che verranno formulati. L'home computer adottato è il TI99/4A della Texas Instruments ma ogni altro personal va be-

ne, sia per la poca memoria di cui c'è bisogno, sia per la facile traduzione che può essere fatta dal BASIC di questa macchina. Come potete vedere dal listato 1 il programma non è affatto lungo; esso è scritto in BASIC normale (non esteso) e le istruzioni grafiche sono quanto mai ridotte. Crediamo però che il bello di questo "pozzo di saggezza" stia soprattutto nelle risposte che vengono date, soprattutto se viene fatta una consultazione collettiva, tra amici.

### Il programma

Diamo, dunque, una rapida occhiata al listato: le istruzioni dalla 5 alla 155 servono per scrivere il titolo del programma e dimensionare il vettore IC\$ in cui saranno memorizzati i responsi dell'oracolo. Le righe 25-40 scelgono ca-

sualmente un colore tra quelli con codice 5 e 14.

Quindi può essere fornita una breve spiegazione sul Libro dei Mutamenti e sul suo uso per coloro che s'avvicinano per la prima volta a questa fonte di sapienza orientale. Dopo di ciò viene chiesto il nome e terminata questa operazione l'oracolo si mette a vostra disposizione chiedendovi il motivo per cui lo interpellate (180-230); l'esposizione del vostro problema può essere lunga quanto vi pare.

Terminata la vostra richiesta, c'è un attimo di raccoglimento e dopo di ciò l'I Ching darà il suo responso. La pausa è stata ottenuta di proposito (270-380) allo scopo di visualizzare un simbolo alla volta e nel contempo approfittare per leggere una decina dei Data delle righe 1400-1555 che contengono tutti i 64 re-

```

390 FOR I=60 TO 63
400 READ IC$(I)
410 NEXT I
420 PRINT
430 PRINT "ECCO IL MIO RESPONSO
      "
435 N=0
440 FOR I=0 TO 5
450 N=N+A(I)*2^I
455 CALL SOUND(99,110+I*11,1)
460 NEXT I
465 FOR I=0 TO 30
470 CALL SOUND(99,110+I*11,1)
475 NEXT I
480 PRINT IC$(N)
485 I=440
490 CALL SOUND(150,I,1)
495 I=I-10
500 IF I=120 THEN 485
505 CALL KEY(0,X,Z)
510 IF Z=0 THEN 490
515 PRINT TAB(60)
520 PRINT "BENE !Spero che quant
o dettoti sia di aiuto.
      Arrivederci a presto
      "
530 END
900 PRINT
910 INPUT "BASTA COSI' ? ":P$
920 IF P$="N" THEN 215
925 IF P$="NO" THEN 215

```

```

930 GOTO 240
1000 CALL CLEAR
1005 CALL SCREEN(C)
1007 CALL COLOR(2,2,C)
1010 PRINT "L'I CHING e'un antic
o testo "
1015 PRINT "cinese,una sorta di
oracolo "
1020 PRINT "che risponde alle do
mande di "
1025 PRINT "chi lo consulta.
      "
1030 PRINT "Non bisogna pero'chi
edergli "
1035 PRINT "previsioni metereolo
giche o "
1040 PRINT "come vincere al toto
calcio o "
1045 PRINT "cose simili.Esso puo
'essere "
1050 PRINT "utile nel fare una s
celta o "
1055 PRINT "nel risolvere un'ind
ecisione "
1060 PRINT "I responsi sono sibi
llini ma "
1065 PRINT "degni di un vero ora
colo !!"
1070 CALL KEY(0,X,Z)
1075 IF Z=0 THEN 1070
1100 GOTO 180

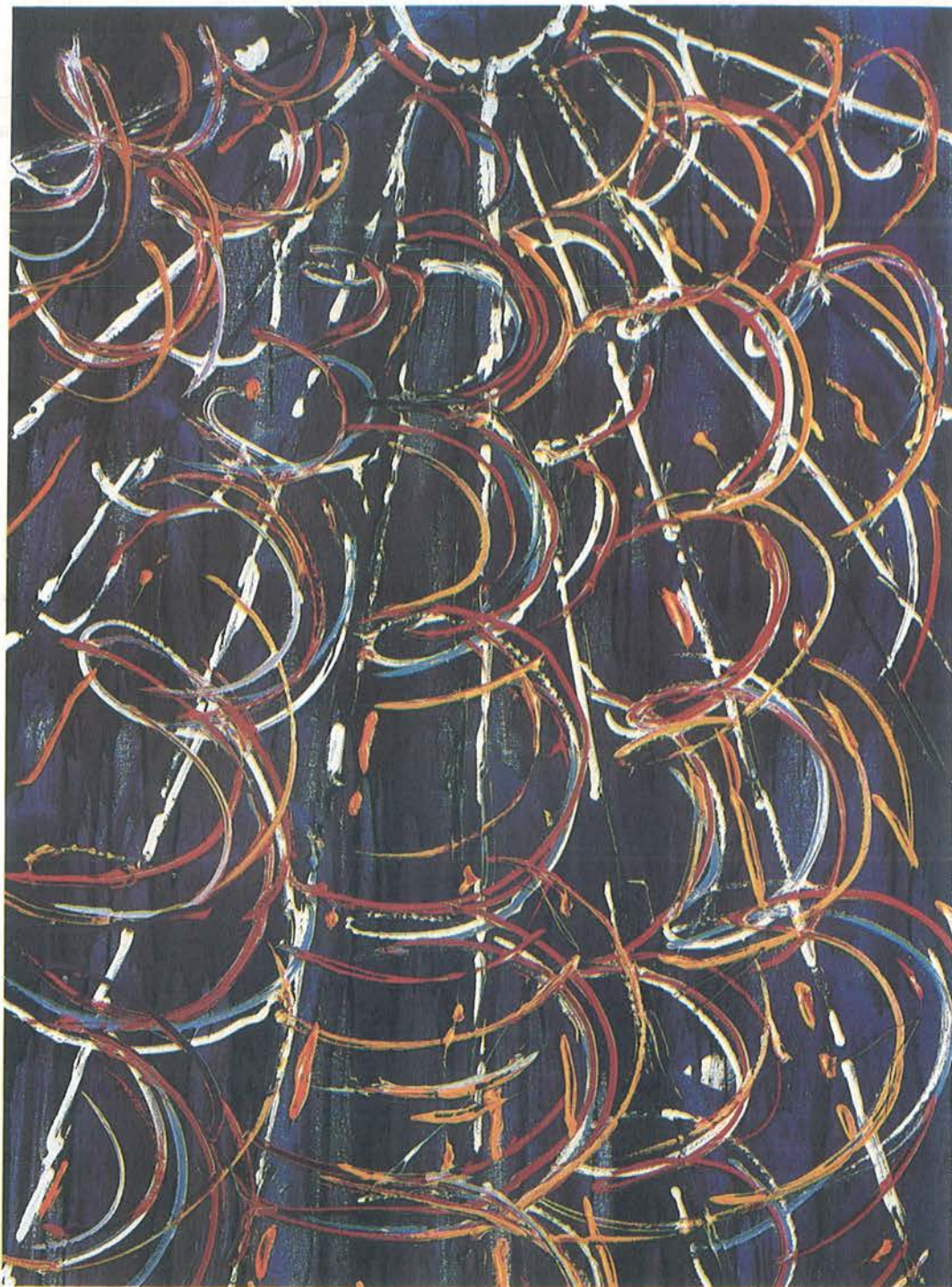
```

# Primo Festival Eur Ustica 29 Giugno /6 Lu isola per fare gr

stC

## Mario Schifano

È nato in Libia a Homs nel 1934. Autodidatta ha esordito nell'ambito dell'informale materico con riferimenti alla pittura di Foutrier. Dal 1962 le prime opere pop con l'utilizzazione di immagini prese dai mass media. Ha fatto parte del gruppo romano della pop-art italiana. In questi ultimi anni opere come inventari della produzione precedente e rivisitazioni di De Chirico.



L'opera di Mario Schifano che sintetizza mare, cielo, sole, computers, che verrà data in omaggio a tutti i partecipanti al festival.

# oporto del Software glio: ci vuole una piccola ande un festival.

**A** Ustica, promosso da Fininvest e Italturist, è nato il Centro Archimede, un'associazione culturale che si propone come luogo permanente di ricerca e confronto sui temi delle nuove tecnologie e dell'informatica.

Quest'anno dal 29 Giugno al 6 Luglio, proprio nell'isola di Ustica, organizzato dal Centro Archimede, si terrà il Primo Festival Europeo del Software al quale è possibile partecipare sia attivamente - inviando uno o più programmi inediti su vari argomenti già precedentemente segnalati - o semplicemente in veste di osservatori dei più recenti sviluppi nel mondo dell'hardware e del software.

Partecipare al Festival di Ustica significa quindi vivere da protagonisti il proprio tempo, aprirsi verso le nuove frontiere dell'intelligenza nell'immediato futuro (si calcola che nel 1990 in Italia una famiglia su tre avrà il suo personal computer), prepararsi ai rapidi imprevedibili sviluppi tecnologici di domani.

## Il Programma del Primo Festival Europeo del Software

**29 giugno - sabato** / Arrivo ad Ustica dei partecipanti. Pomeriggio: cerimonia inaugurale e convegno su "Mezzogiorno innovazione e nuove tecnologie". In serata spettacolo di apertura.

**30 giugno - domenica** / Convegno in collaborazione con la FAST sull'intelligenza artificiale e sistemi esperti. In serata tavola rotonda sul software italiano.

**1° luglio - lunedì** / Convegno e presentazione del progetto "Esprit" della Comunità Europea in collaborazione con il Ministero della Ricerca Scientifica.

**2 luglio - martedì** / Convegno sulle prospettive dell'informatica nel settore politico e rapporti sociali in collaborazione con il Centro Einaudi di Torino.

**3 luglio - mercoledì** / Convegno su "Informatica, territorio e casa" in collaborazione con il Censis. In serata dibattito sulla difesa ambientale.

**4 luglio - giovedì** / Convegno sulla "Moneta elettronica" in collaborazione con il Banco di Sicilia. In serata spettacolo folcloristico.

**5 luglio - venerdì** / Convegno sulle "Tecnologie per il turismo" in collaborazione con Assicurazioni Ticino.

Nel pomeriggio convegno su "Informatica per il manager", in collaborazione con Italturist. In serata spettacolo con premiazione dei vincitori del concorso sul software e assegnazione dei floppy d'oro, d'argento e di bronzo.

**6 luglio - sabato** / In mattinata convegno sulla "Difesa giuridica del software".

**Ogni sera** / Musica in piazza, rassegne cinematografiche e di video-clip.

Si svolgeranno inoltre le finali del campionato di videatletica, realizzate in collaborazione con A.I.V.A.

### QUOTE DI PARTECIPAZIONE PER L'INTERA SETTIMANA

SISTEMAZIONE	DA MILANO aereo MI/PA e viceversa - aliscafo	DA ROMA aereo Roma/PA e viceversa - aliscafo	DA NAPOLI aliscafo	DA PALERMO aliscafo
VILLAGGIO PUNTA SPALMATORE SEDE CONGRESSI	850.000	750.000	680.000	630.000
ALBERGO (tipo A)	740.000	650.000	550.000	470.000
ALBERGO (tipo B)	640.000	550.000	500.000	370.000
ALBERGO (tipo C)	590.000	500.000	400.000	330.000
CASA PRIVATA	480.000	380.000	270.000	210.000
APPARTAMENTO	580.000	490.000	380.000	310.000

#### La quota comprende:

I trasferimenti con accompagnatori dall'aeroporto al porto di Palermo e viceversa.

L'aliscafo da Palermo o da Napoli ad Ustica e viceversa. Trattamento di pernottamento e mezza pensione negli alberghi e di pensione completa nel villaggio Punta Spalmatore.

Trattamento di solo pernottamento nelle case private e in appartamento.

#### Supplementi:

Pensione completa negli alberghi: L. 50.000

Trasferimento in pullman da Roma a Napoli in coincidenza della partenza dell'aliscafo per Ustica: L. 40.000.

**Quota di iscrizione:** L. 100.000 da pagare contemporaneamente alla quota di partecipazione.

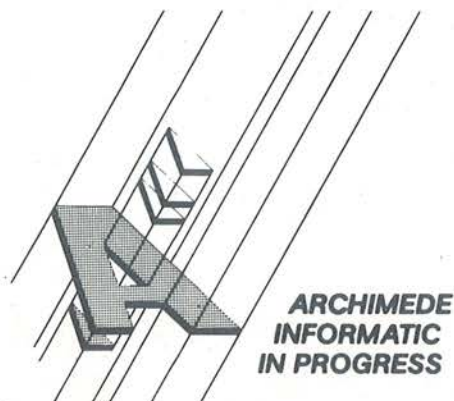
La quota di iscrizione consente la partecipazione a tutte le iniziative in programma.

Per iscrizioni rivolgersi a: ITALTURIST - Via V. Pisani, 16 Milano - Tel. 02/6704505 - 6704510

Oppure a queste Sedi locali:

BOLOGNA 051/271203 - 234984  
ROMA 06/6797077 - 6785465

FIRENZE 055/282680 - 283436  
PALERMO 091/297314 - 261052



**ARCHIMEDE  
INFORMATIC  
IN PROGRESS**

Seguito listato I Ching.

1400 DATA "RICETTIVITA'.SUCCESSO CON L'OBEDIENZA", "SGRETOLAMENTO.CERCA DI NON MUOVERTI"

1405 DATA "UNIONE.SEI VERAMENTE SICURO DI TE STESSO ?","CONTEMPLAZIONE.LO SCOPO NON E'RAGGIUNTO"

1410 DATA "ARMONIA.IL RISPETTO GENERA IL RISPETTO","PROGRESSO.E SIBITE LE VOSTRE QUALITA'"

1415 DATA "ASSEMBRAMENTO.IL PICCOLO E' AIUTATO DAL GRANDE","STASI.MANCA LA COMUNICAZIONE"

1420 DATA "MODESTIA.BRILLARE MA SENZA ABBAGLIARE","LA MONTAGNA.CONCENTRAZIONE ERIPOSO"

1425 DATA "ARRESTO RAPIDO.COLTIVARE UN TALENTO","CRESCITA.OCCORRE FERMEZZA E CORREZIONE"

1430 DATA "PICCOLI ECCESSI.SUCCESSO,MA NELLE PICCOLE COSE","IL VIAGGIATORE.RICERCA DI UNACCORDO"

1435 DATA "INFLUENZA.SAGACITA'IN MOLTE OPINIONI","RITIRATA.LA DIGNITA'PRIMA DITUTTO"

1440 DATA "COORDINAZIONE.SAGGEZZA DELLAESPERIENZA","INESPERIENZA.INUTILE FARE,SIDEVE RIPETERE"

1445 DATA "L'ABISSO.LA SINCERITA'E'DA PREFERIRSI","DISPERSIONE.COORDINAZIONE DELLE IDEE"

1450 DATA "VIA LIBERA.TORNATE AI MODI ANTICHI","VERSO LA FINE.I GUAI SONO FINITI"

1455 DATA "OPPRESSIONE.PERCHE TANTE PA-ROLE ?","CONFLITTO.BISOGNA SAPERSI FERMARE IN TEMPO"

1460 DATA "ASCESA.IL FURBO COGLI



## TROVERETE NEL PROSSIMO NUMERO

IN  
EDICOLA  
DAL  
26  
GIUGNO

**D**ISEGNATORE:  
UN PROGRAMMA  
PER MSX

**E**ASYVIDEO  
PER C 64

**A**GENDA  
TELEFONICA  
CON IL CASIO

**H**I-RES PER TI99

**R**OUTINE 2  
PER SPECTRUM

**S**IMULAZIONE  
DINAMICA  
DI FENOMENI  
CONTINUATIVI  
CON IL C 64

**C**UBO MAGICO  
CON SPECTRUM

**S**TATISTICA  
A PIÙ  
DIMENSIONI  
PER APPLE

**D**ATA BASE  
PER C 16



## TEXAS TI99/4A

## Seguito Ilistato I Ching.

E L'OC-CASIONE", "PURIFICAZIONE. E SAMINATE BENELE CONSEGUENZE"

1465 DATA "IL POZZO. AIUTO RECIPROCO E COMPRESIONE", "INSERIMENTO. BISOGNA TENTARE RIPETUTAMENTE"

1470 DATA "PERSEVERANZA. TUTTE LE STRADESONO APERTE", "FERMEZZA. TENERE GLI OCCHI E LE ORECCHIE BEN APERTI"

1475 DATA "GRANDI ECCESSI. INDIFFERENZA AI COMMENTI", "INCONTRO. NON FIDATEVI DEI PIU' FORTI"

1480 DATA "RITORNO. L'OSTACOLO NON E' POICOSI' IMPORTANTE", "ALIMENTO. IL PENSIERO ACCRESCER L'AZIONE"

1485 DATA "INIZIO DIFFICILE. TROPPO SPERANZA INGANNA", "ACCRESCIMENTO. AVANZATE SENZAESITARE"

1490 DATA "IL TUONO. NON METTETEVI IN APPRENSIONE", "LA PENETRAZIONE. IL SAGGIO E' PROTETTO DALLA LEGGE"

1495 DATA "DISCIPLINA. UN GRANDE PROTETTO DAI PICCOLI", "INNOCENZA. LA SCORRETTEZZA PORTA AL DISASTRO"

1500 DATA "OSCURAMENTO. PERCHE' AT TIRARE L'ATTENZIONE?", "AVVENENZA. A VOLTE CI SI DEVE FIDARE DELLE APPARENZE"

1505 DATA "SUCCESSO. FINIRE E' LA PARTE ARDUA", "LA FAMIGLIA. OGNI COSA AL SUOPOSTO"

1510 DATA "ABBONDANZA. IL MEZZO SUCCESSO NON BASTA", "CHIAREZZA. BISOGNA FARE IL NECESSARIO"

1515 DATA "RIVOLUZIONE. PRIMA AGIRE, DOPO PENSARE", "COMPAGNIA. NON MESCOLATE CIO' CHE NON SI MESCOLA"

1520 DATA "AVVICINAMENTO. IL SUCCESSO PUO' ESSERE PROVVISORIO", "MINORAZIONE. SI STABILIRA' UN CERTO EQUILIBRIO"

1525 DATA "RESTRIZIONE. FRENO ALL'ENTRATA PRESTO O TARDI", "SINCERITA'. IL SAGGIO CREDE A COLUI CHE CREDEREA"

1530 DATA "LA FIDANZATA (O LA MOGLIE). VIRTU' DELL'ASTINENZA", "SEPARAZIONE. LA PREVISIONE CATTIVA S'INGANNA"

1535 DATA "GIOIA. NON DIMENTICARE ANCHE IL PIACERE", "PRUDENZA. MA FORSE LA TIGRE NON MORDE"

1540 DATA "PACE. FINE DELLE PICCOLEZZE EINIZIO DELLE GRANDEZZE", "DOMARE. ISPIRARSI AGLI ALTRI"

1545 DATA "ATTESA. NON FARSITRASCIANARE", "RINSAVIMENTO. FORSE LA NUBE NON SCOPPIERA"

1550 DATA "POTENZA. LA FERMEZZA TRIUNFA", "POSSESSO. TENETE BENE I FILI IN MANO"

1555 DATA "APERTURA. SPARPAGLIATE E NON AMMASSATE", "IL CREATIVO. UN CAMBIAMENTO E' SALUTARE"

sponsi.

A volte può non apparire molto chiaro quello che dice l'I Ching, ma dobbiamo sempre rammentare che ci troviamo di fronte ad un oracolo, il cui scopo è anche indurre coloro che lo consultano ad esaminare sé stessi e le cose, partendo magari dalla riflessione sui suoi responsi (volutamente stringati e sibillini).

La riga 450 calcola in che posizione del vettore IC\$ si trova la risposta da dare; quindi verrà prodotta una serie di suoni

che durerà finché l'utente non digita un qualsiasi carattere che decreterà la fine del programma. Come potete vedere è tutto molto semplice e lineare e le uniche istruzioni che i "non texani" dovranno modificare sono le varie Call che agiscono sullo schermo, sul sonoro o sul carattere ricevuto o meno.

Come dettovi in precedenza, noi consigliamo di consultare l'I Ching tra amici perché possono verificarsi responsi molto interessanti come quanto è suc-

cesso ad una coppia fresca fresca di matrimonio, di nostra conoscenza, ritornati dal loro viaggio di nozze. A "lei", che chiedeva informazioni sulla fedeltà del marito l'I Ching ha risposto con "Progresso. Esibite le vostre qualità", mentre per "lui" che domandava come sarebbero stati i primi mesi di matrimonio il responso è stato "La fidanzata (o la moglie). Virtù dell'astinenza". Che altro dire?!

## Alcune note tecniche

**L'**aeromodello a volo vincolato circolare da allenamento (U-Control) che per brevità chiameremo semplicemente V.V.C., è ancora oggi il modello ideale per il principiante che desidera avvicinarsi all'aeromodellismo a motore con poca spesa e con buone possibilità di pieno successo. Vediamo come sono fatti, in linea generale questi modelli.

I V.V.C. sono aeromodelli con motore a scoppio che vengono comandati per mezzo di due cavi di filo d'acciaio che muovono, tramite alcuni semplici leveraggi, il timone di profondità dell'aereo: la traiettoria è quindi circolare, con possibilità di variare la quota di volo e di eseguire alcune manovre acrobatiche (compatibilmente alle caratteristiche dell'aereo e, soprattutto, alla bravura del pilota).

Il pilota guida l'aeromodello per mezzo di una manopola, alla quale sono fissati i cavi di comando che si innestano in una squadretta imperniata alla fusoliera o all'ala del modello (figura 1).

Quando il pilota piega verso di sé l'estremità superiore della manopola, eleva verso l'alto il timone di profondità, causando una deportanza sul piano di coda e, di conseguenza, fa salire l'aereo. Ovviamente, il movimento contrario della manopola abbassa la profondità, causando un incremento di portanza sul piano di coda che fa picchiare il velivolo (figura 2).

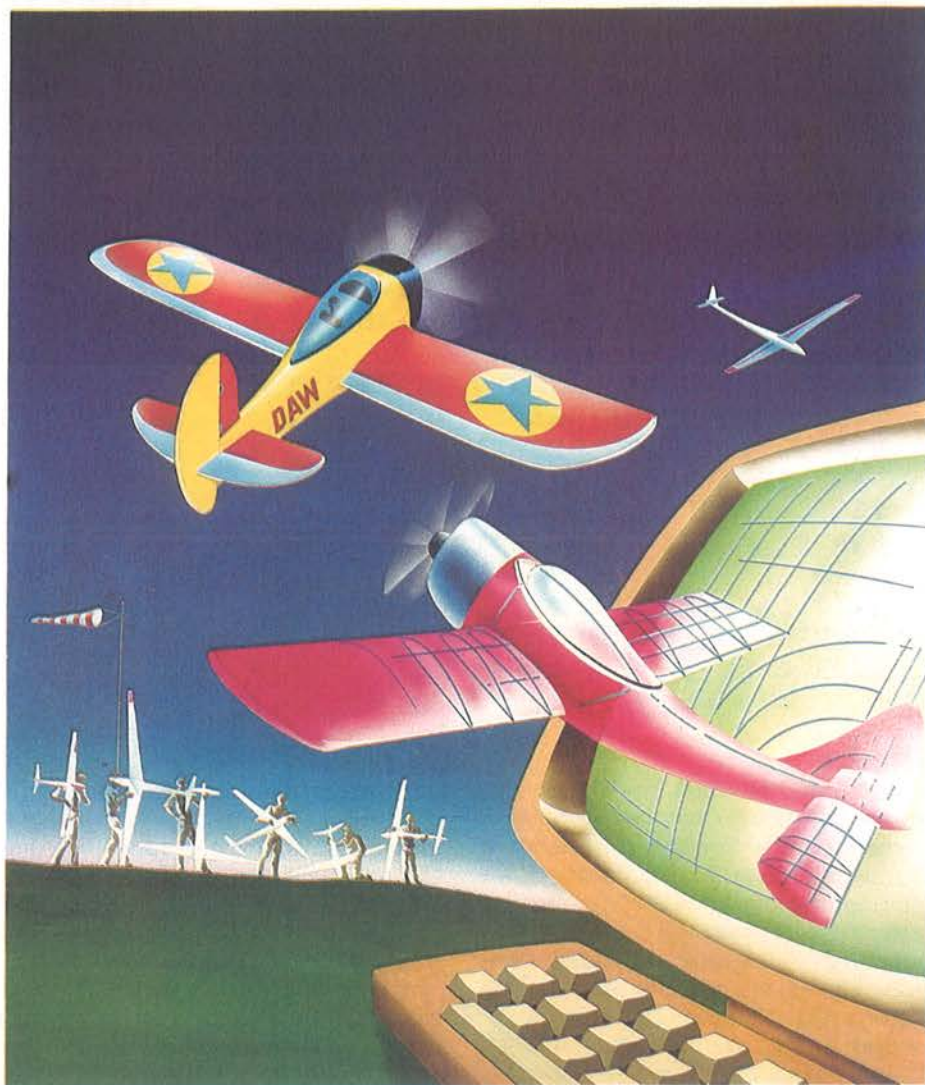
Tanto maggiore è la sensibilità ai comandi dell'aereo, tanto più pronta sarà la risposta ai minimi movimenti del polso del pilota: ciò comporta la possibilità di eseguire manovre rapide e complesse, a scapito della facilità di pilotaggio, quindi anche un piccolo errore di manovra può causare la distruzione del modello. In questo programma verrà elaborato il progetto di un V.V.C. da addestramento (trainer), con comandi non troppo sensibili che "perdonino" eventuali errori di pilotaggio grazie a dei tempi di reazione più lunghi rispetto ai modelli acrobatici.

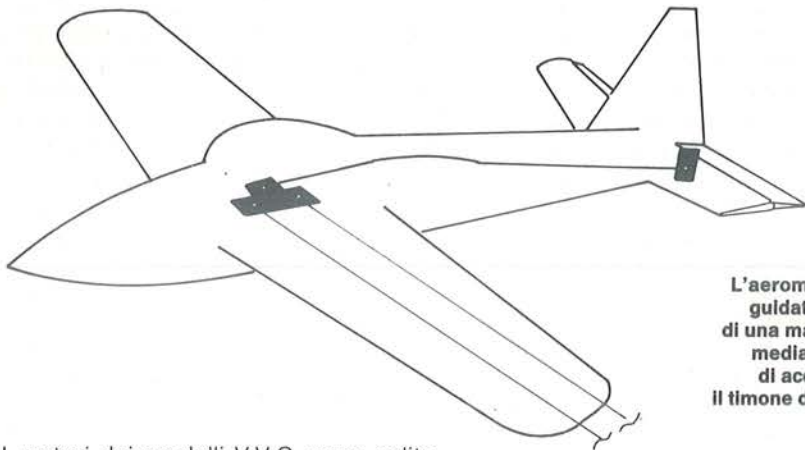
In un V.V.C. il complesso dei dispositivi di comando sono in definitiva la manopola, i cavi, la squadretta di comando e la barretta d'acciaio che collega quest'ultima col timone orizzontale.

# Progetto aeromodello "U-Control"

Volate con il vostro Spectrum 48 Kbyte

di Paolo Ferrami





**Figura 1 -**  
L'aeromodello viene guidato per mezzo di una manopola che, mediante due cavi di acciaio, muove il timone di profondità.

I motori dei modelli V.V.C. sono solitamente di piccola cilindrata e non troppo spinti come prestazioni, risultando quindi anche più economici.

La fusoliera può essere di tipo a cassone oppure, più semplicemente, ricavata da una tavoletta di balsa dura, di 12 millimetri circa o di altro legno più resistente come obeche o pioppo, naturalmente di minor spessore. Questo tipo di fusoliera anche se esteticamente meno valida del tipo cassonato, riduce di gran lunga le difficoltà di costruzione e installazione sia dei comandi, sia del gruppo motore-serbatoio.

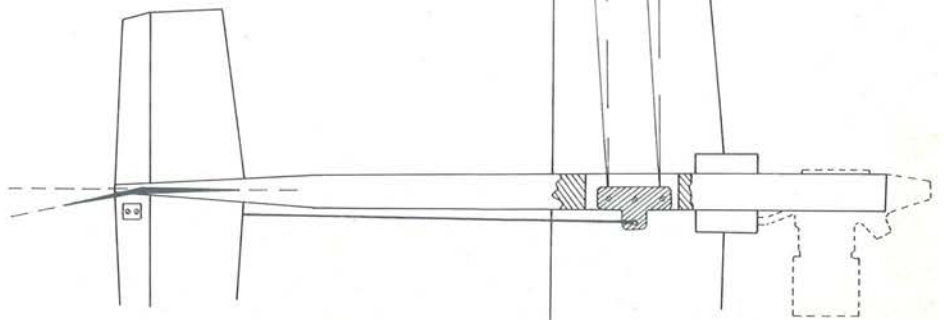
Analogo discorso si può fare per l'ala: di solito per piccole cilindrate è possibile ricavarla da un foglio di balsa sagomato secondo il profilo alare, anche se è sempre preferibile l'ala costruita con centine di balsa con profilo piano-convesso, con bordo d'attacco rinforzato e rivestite con carta-seta pesante o altro materiale equivalente.

Il gruppo di superfici di coda viene ricavato da una tavoletta di balsa di 3-5 millimetri di spessore, ricordando che il piano di coda e la deriva sono formate da due parti distinte: nel piano di coda si distinguono una parte fissa (stabilizzatore) e una parte mobile (timone orizzontale) incernierate mediante cerniere di nylon che si possono trovare in qualsiasi negozio di modellismo o, al limite, mediante cerniere di tessuto incollate ai due piani, in modo che il timone di profondità sia libero di ruotare; nell'impenningo verticale il timone verrà invece incollato alla deriva, inclinato di qualche grado, verso l'esterno della circonferenza di volo, come verrà indicato dal programma stesso.

Alla fusoliera potrete dare la forma che

preferite, vincolati solamente dalla lunghezza del braccio di leva, dalla posizione del centro di gravità e dalla facilità di raggiungere gli "organi vitali" dell'aereo per la manutenzione; queste misure, cui abbiamo accennato prima, vi verranno date dal computer dopo averle elaborate dai dati in proprio possesso. Ricordatevi che il serbatoio del carburante deve essere del tipo adatto ai V.V.C. e deve essere posto alla stessa altezza del carburatore del motore, per una corretta alimentazione durante il volo.

Come forse avrete già intuito, prima di accingervi a progettare e a costruire un aeromodello anche semplice come un V.V.C., è necessario un minimo di conoscenze tecniche e di esperienza o, per lo meno, che un amico più esperto di voi, vi aiuti nei primi tentativi di volo, assistendovi direttamente "sul campo".



**Figura 2 -**  
Ecco come il movimento dei cavi di acciaio fa variare l'inclinazione del timone di profondità dell'aereo.

Infatti costruire un aeromodello non comporta grosse difficoltà, specialmente se questo modello è stato acquistato, in scatola di montaggio, in un negozio di modellismo (vi sono molti modelli, di diverse marche, corredati di tutti gli accessori e, soprattutto, di dettagliatissime istruzioni per il montaggio), ma, una volta giunti sul campo di volo, normalmente capita che, dopo non pochi tentativi, il modello non ne voglia sapere di partire o peggio ancora, una volta decollato ritorni a terra a una velocità e a un assetto più vicini a quelli di un sasso lanciato in aria, che a quelli di un velivolo, trasformando l'aeromodello in una scultura astratta di balsa.

Vi consiglio quindi, prima di intraprendere la progettazione di un V.V.C. con questo programma, di leggere qualche testo dedicato al modellismo aereo o, meglio ancora, di prendere dimestichezza con questo hobby costruendovi dapprima un aeromodello ricavato da una scatola di montaggio acquistata in un negozio (tanto per "farvi le ossa") e poi, in un secondo tempo, prendervi la grande soddisfazione di progettare e costruire un modello tutto frutto della vostra fantasia e ingegno, aiutati nel dimensionamento e nel progetto dalla "scatola nera" di Sir Clive, vero gioiello nel firmamento personal.

## Il programma

Il programma "Progetto e dimensionamento di un V.V.C." vi permetterà di progettare un aeromodello a volo vincolato da allenamento di cilindrata compresa tra 1 e 3,5 centimetri cubi inserendo, come uniche variabili, la cilindrata e l'allungamento delle superfici (ricordo che l'allungamento è il rapporto tra l'apertura alare e il quadrato delle superfici). Questa semplificazione è stata possibile perchè possiamo considerare la cilindrata di un piccolo motore da V.V.C. proporzionale alla potenza erogata, approssimazione accettabile solo nell'arco di utilizzazione di cilindrata sopra menzionata.

Questa limitazione non dà nessun problema, perchè è sconsigliabile costruire modelli da addestramento con cilindrata superiori ai 3,5 c.c.

Impostata la cilindrata, il programma ricava il rapporto peso-potenza ottimale, quindi il peso totale e la superficie alare dell'aeromodello, seguendo alcune norme generali dettate anche dall'esperienza. È chiaro che, noti questi

dati, basta impostare l'allungamento delle superfici per avere tutte le dimensioni necessarie per il disegno dell'aereo. Il programma è stato strutturato a pagine; ognuna delle quali tratta un particolare aspetto del progetto: vediamo con un esempio come si deve utilizzare il programma.

All'inizio una schermata introduttiva illustra le quattro opzioni che si hanno a disposizione durante il programma:

**P** - pagina precedente: serve per saltare a una pagina ennesima del programma. Ovviamente la pagina deve essere una pagina precedente rispetto a quella in cui ci si trova al momento della richiesta;

**S** - pagina seguente: serve per passare alla pagina successiva;

**C** - copia su carta: serve per avere l'hard copy della schermata sulla stampante, comando estremamente utile se si vuole disporre dei disegni quotati dell'aereo;

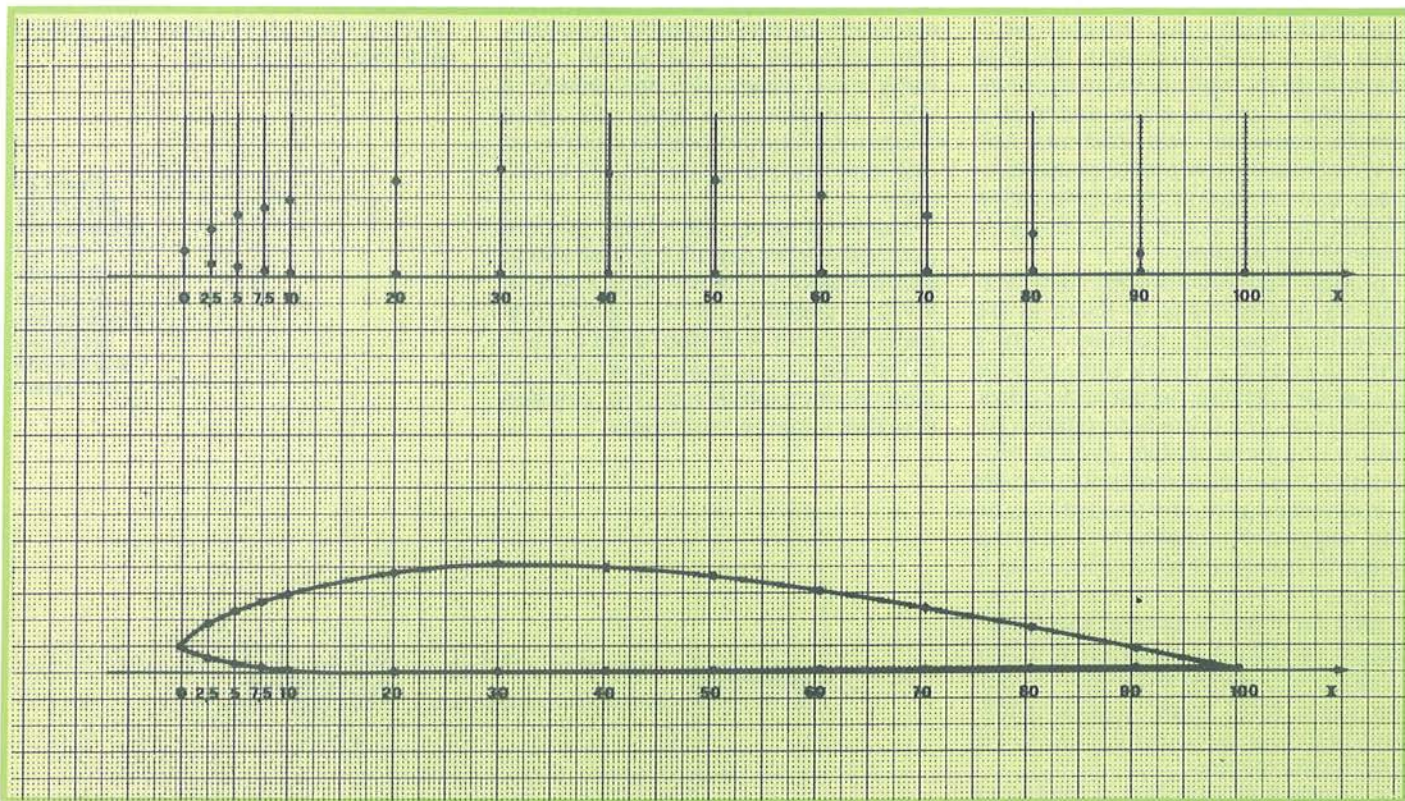
**A** - abbandono: questo comando blocca il programma senza cancellarlo. Per evitare di fermarlo accidentalmente, viene richiesta una conferma prima di eseguire lo Stop.

Ogni altro tasto viene disattivato.

Dopo questa schermata introduttiva segue la pagina 1 che, dopo avere eseguito il disegno di un motore, chiede la cilindrata usata dal modello (che ricordo deve essere compresa tra 1 e 3,5 c.c.). Eseguito l'input del dato, compare la scritta delle quattro opzioni disponibili; premiamo S e passeremo alla pagina 2. In questa pagina il programma eseguirà un primo dimensionamento di massima del modello; compariranno, lampeggiando per qualche secondo, la cilindrata, il peso totale e la superficie alare del modello. Una scritta, vicino al disegno in pianta dell'ala, ci avvertirà che verrà presa come campione un'ala a pianta rettangolare.

Passiamo alla pagina 3. Qui ci viene richiesto l'allungamento; maggiore è l'allungamento, maggiore sarà l'apertura alare a scapito della lunghezza della corda media (come mostrato dai disegni sul video). Per motivi di esperienza, abbiamo limitato la scelta a due soli allungamenti. Fatta la scelta comparirà, di fianco al disegno corrispondente, il valore dell'allungamento scritto con caratteri inversi.

Figura 3 - Lo schema del profilo Saint Cyr 52: è uno dei tre profili alari piano-convessi proposti dal programma.



Seguono le quattro opzioni. La pagina è formata da più sottopagine, che vedremo in dettaglio: dopo avere visualizzato tutti i dati necessari per il dimensionamento dell'ala, compare il disegno di un profilo alare piano-convesso seguito dai nomi di tre dei profili più usati: Clark X, Clark Y e Saint Cyr 52 (figura 3) preceduto ciascuno da un numero. Se desideriamo utilizzare uno di questi profili, dobbiamo semplicemente digitare il numero corrispondente e, dopo aver eventualmente copiato sulla stampante la pagina, passeremo a una delle sottopagine 4bis con tabulate le coordinate del profilo scelto.

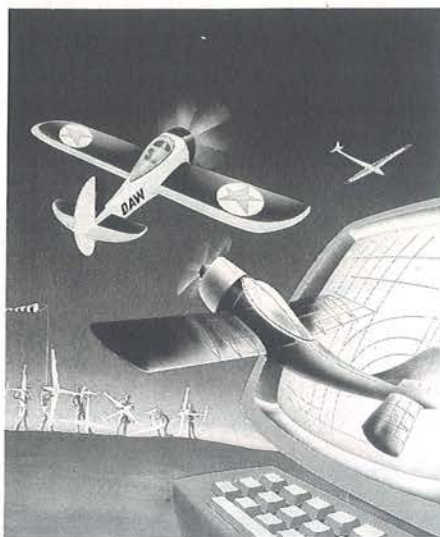
Per disegnare il profilo desiderato usando i dati contenuti nella tabella, bisogna sapere che i simboli X, Ys e Yi rappresentano rispettivamente le ascisse, le ordinate superiori e le ordinate inferiori del profilo che abbiamo scelto. Le quote dei profili sono in millimetri e riferite a un profilo di lunghezza standard di 100 mm. Una volta fatto il debito rapporto tra profilo standard e quello del nostro modello, si riportano su carta millimetrata sopra una retta orizzontale i valori di X e, per ciascuna ascissa, le coppie di valori delle ordinate; unendo i punti così ottenuti con un curvilineo, otterremo il disegno del profilo desiderato (disegno non ottenibile direttamente dallo Spectrum a causa della troppa alta definizione dei punti rispetto alla risoluzione del computer).

Stampate le coordinate, passiamo alla pagina 5, nella quale lo Spectrum ci darà il disegno quotato dell'ala vista dall'alto e di fronte. A questo punto conosciamo tutto dell'ala, per cui possiamo passare a occuparci del piano di coda: la pagina 6 ci mostra per prima cosa la superficie totale del piano di coda e dello stabilizzatore per poi chiederci, con le stesse modalità dell'ala, l'allungamento da adattare per gli impennaggi, scegliendo il valore tra i due consigliati. Anche qui il valore scelto verrà stampato in caratteri inversi di fianco al disegno della superficie interessata. Passando alla pagina 7, il programma calcolerà tutti i dati dell'impennaggio orizzontale: superficie, allungamento, apertura e corda media.

Per quanto riguarda il profilo da adottare, è buona norma utilizzare lo stesso profilo impiegato per l'ala (come ci viene ricordato dallo stesso programma dopo

avere disegnato un profilo alare piano-convesso). Stampati i dati (o trascritti su un foglio se non si possiede la stampante), passiamo alla pagina 8 che ci disegnerà le tre viste del piano di coda, evidenziando nel disegno e nelle quotature le due superfici distinte dello stabilizzatore e del timone di profondità; anche per questo motivo si è reso necessario il tracciamento della vista laterale, con quotata la frazione di corda spettante al timone di profondità.

Il passo successivo riguarda il dimensionamento degli impennaggi verticali, cioè della deriva e del timone verticale. La forma degli impennaggi verticali riveste poca importanza nei modelli a volo vincolato essendo la stabilità laterale assicurata dal vincolo dei cavi. Nel pro-



gramma è stata presa come campione una diffusissima configurazione di impennaggi verticali: quella a trapezio-rettangolo. La pagina 9 ci dà dunque tutti i dati per il dimensionamento di deriva e timone: superficie e apertura, corda minima e massima nonché la superficie della parte mobile. Il computer ci mostra anche il disegno di un profilo biconvesso-simmetrico senza consigliarne nessun tipo: questo perché solitamente gli impennaggi verticali sono ricavati da tavolette di balsa semplicemente arrotondate sul bordo d'entrata e appuntite su quello d'uscita.

Le tre viste quotato degli impennaggi a pagina 10 del programma mostrano come il timone verticale vada fissato (incolato!) alla deriva con una inclinazio-

ne di 8°-10° verso l'esterno della circonferenza di volo, in modo che l'aeromodello tenda a virare verso l'esterno della circonferenza, tendendo così i cavi di comando per avere sempre il controllo del velivolo anche se questo assume assetti pericolosi. La pagina 11 ci dà le posizioni relative (calettamenti) tra ala e piano di coda e tra asse motore e asse del velivolo. Inoltre viene consigliata l'escursione massima del timone di profondità a cabrare e a picchiare, per avere un modello relativamente lento alle sollecitazioni del pilota al fine di facilitarne il controllo.

Passando alla pagina 12, vedremo comparire sullo schermo due classici tipi di squadretta (a T e trapezoidale) la scelta delle quali sarà dettata per lo più da motivi di ingombro. Comunque, posto di scegliere il modello A, vedremo lo schermo pulirsi parzialmente, per poi disegnare al centro la squadretta scelta, quotata nei tre fori di attacco della barra e dei cavi. Vengono segnalati due valori per la distanza dei cavi: uno per avere un pilotaggio facile, l'altro per comandi più sensibili.

Premiamo il tasto S per passare alla pagina 13, che ci indicherà i valori da rispettare per il disegno della fusoliera, affinché il modello risulti centrato (cioè il baricentro cada nel punto corretto per un volo equilibrato del modello). Il primo dato che viene calcolato è il braccio di leva, che risulta essere la distanza tra il centro di gravità (o baricentro) e il centro aerodinamico del piano di coda (che possiamo con buona approssimazione considerare posizionato al 25 per cento della corda dello stabilizzatore). Il braccio di leva quindi determina la lunghezza della parte di fusoliera compresa tra le due superfici orizzontali (ala e piano di coda). Questa grandezza è intimamente legata alla superficie dell'impennaggio orizzontale, per cui una variazione arbitraria di questo valore potrebbe portare grossi problemi di stabilità al modello. La seconda grandezza è il baricentro, calcolato in percentuale della corda media e misurato in centimetri a partire dal bordo d'attacco.

Per far sì che il centro di gravità cada nel punto indicato dal computer, è necessario che il peso del muso del modello, equipaggiato di motore, serbatoio e accessori, eguagli quello della parte rimanente, cioè quella compresa tra il bari-

## VARIABILI NUMERICHE

**Q** - Contiene il valore della nota del brano musicale.  
**Z** - Contiene il valore della nota del brano musicale.  
**A** - Idem (viene usata anche come variabile di controllo).  
**L** - Variabile di controllo.  
**O** - Contiene i valori numerici inseriti nei Data.  
**CMC** - Contiene la cilindrata in centimetri cubici del motore.  
**PARA** - Parametro per il dimensionamento del peso e della superficie alare.  
**PESO** - Contiene il peso del modello.  
**SUPALA** - Contiene il valore della superficie alare in centimetri quadrati.  
**LAM** - Contiene il valore dell'allungamento alare.  
**APALA** - Contiene il valore dell'apertura alare.  
**CORALA** - Contiene il valore della corda media alare con tutti i decimali (il valore verrà poi ridotto a una precisione di due sole cifre decimali).  
**CORAL** - Contiene la parte decimale del valore iniziale di Corala.  
**CORIN** - Contiene le prime due cifre decimali di Corala.  
**CORIND** - Contiene le prime due cifre decimali di Coral.  
**PRO** - Contiene il tipo di profilo scelto.  
**SUPIA** - Contiene il valore della superficie del piano di coda.  
**PARMO** - Contiene il valore della superficie della parte mobile del piano di coda.  
**LAMP** - Contiene l'allungamento del piano di coda.  
**CORPIA** - Contiene il valore della lunghezza della corda media del piano di coda con tutti i decimali

(conterrà poi solo il valore ridotto a due decimali).

**CORPI** - Contiene la parte decimale di Corpia.  
**CORPN** - Contiene le prime due cifre decimali di Corpia.  
**CORPIND** - Contiene il valore di Corpn, diviso per cento.  
**STAB** - Contiene il valore della frazione di corda della parte mobile (il valore verrà poi ridotto a una precisione di due sole cifre decimali).  
**STA** - Contiene le sole cifre decimali di Stab.  
**STAD** - Contiene le prime due cifre decimali di Stab.  
**SUDERI** - Contiene il valore della superficie della deriva.  
**APERI** - Contiene il valore dell'apertura della deriva.  
**CORDERI** - Contiene il valore della corda media della deriva.  
**CORDEMI** - Contiene il valore della corda minima della deriva.  
**CORDEMA** - Contiene il valore della corda massima della deriva.  
**BRAL** - Contiene il valore della lunghezza del braccio di leva.  
**CG** - Contiene il valore della posizione del centro di gravità rispetto al bordo di entrata dell'ala.  
**SQ** - Contiene il valore della posizione della squadretta di comando rispetto il bordo di entrata dell'ala.  
**PA** - Contiene il numero della pagina a cui si vuole saltare.

## VARIABILI ALFANUMERICHE

**KS** - Contiene sotto forma di stringa il valore di Corpia  
**AS** - Contiene il tipo di squadretta scelta  
**QS** - Stringa contenente ventisette spazi

Tabella 1 - Elenco delle variabili numeriche e alfanumeriche usate.

centro e la coda. Un modo semplicissimo per verificare il centraggio è quello di sollevare l'aeromodello con due dita sull'asse passante per il baricentro e verificare che il modello stia in equilibrio: se tende a pendere in avanti, si deve alleggerire il muso o, se siamo ancora sotto il limite di peso, appesantire la coda (per esempio incollando degli strati di balsa): se invece è il muso a essere troppo leggero, si deve appesantirlo o allungarlo (è, come vedete, il principio della bilancia a due bracci).

Ottima cosa è effettuare questi tentativi riportando su carta il disegno di massima della fusoliera e, una volta segnato il punto in cui cade il C.G. del velivolo, trascrivere le coordinate cartesiane dei vari componenti del modello (motore, serbatoio, carrello ecc.) e, per ciascun componente, riportare su una tabella il peso, l'ascissa e l'ordinata.

Utilizzando le formule  $X = (P_n \star X_n) / P_{tot}$ ;  $Y = (P_n \star Y_n) / P_{tot}$ ; dove  $P_n$  è il peso di ciascuna componente,  $X_n$  e  $Y_n$  le relative coordinate,  $P_{tot}$  il peso totale del modello, otterremo le coordinate  $X$  e  $Y$  del C.G.. Se queste non corri-

spondono a quelle richieste, bisogna modificare la disposizione dei pesi (l'ordinata del C.G. non ha molta importanza in un V.V.C.). Vediamo un esempio:

	P <sub>n</sub>	X <sub>n</sub>	Y <sub>n</sub>
Motore con elica	150	25	30
Serbatoio	50	32	20
Ala	80	45	25
Fusoliera	90	50	30
Impennaggi	40	95	35

Peso totale 410 gr.

$$X_{cg} = (150 \star 25 + 50 \star 32 + 80 \star 45 + 90 \star 50 + 40 \star 95) / 410 = 42,07$$

$$Y_{cg} = (150 \star 30 + 50 \star 20 + 80 \star 25 + 90 \star 30 + 40 \star 35) / 410 = 28,29$$

I valori usati, ovviamente, sono puramente indicativi.

L'ultimo valore visualizzato dal computer è la posizione della squadretta di comando, sia in percentuale della corda, sia in centimetri dal bordo d'attacco.

Vi viene consigliato anche la misura e l'inclinazione dei cavi di comando. La pagina 14 visualizza un disegno di una semivista dall'alto del modello con indicate le posizioni degli organi di comando.

In basso a destra sullo schermo verrà disegnato il profilo del piano di coda con indicata l'altezza dell'attacco della barra con il timone di profondità.

Anche l'attacco, come la squadretta, potete autocostruirla o acquistarla in un qualsiasi negozio di modellismo. Nel disegno che compare in questa pagina sono evidenziati i cavi inclinati rispetto all'asse trasversale dell'ala che, insieme all'inclinazione del timone verticale, contribuiscono a far sì che il modello non tenda a entrare all'interno della circonferenza di volo (un altro metodo è quello di inclinare di pochi gradi il motore verso l'esterno rispetto all'asse della fusoliera, ma i primi due accorgimenti adottati sono senz'altro sufficienti e più semplici da realizzare). La pagina 15 è l'ultima pagina del programma e contiene alcuni consigli per la messa a punto del modello. A questo punto, se lo desi-

derate potrete saltare alla pagina 3 e modificare l'allungamento dell'ala o alla pagina 6 per modificare quello del piano di coda per vedere la differenza e scegliere poi tra i diversi disegni, quello più adatto alla linea del vostro modello. Per eseguire tutto il progetto il programma impiega solo qualche minuto, ma è chiaro che il tempo aumenta considerevolmente se desiderate effettuare l'hard copy di tutte le schermate tramite la stampante.

Da quanto risulta da questa descrizione, il programma richiede un intervento limitatissimo da parte dell'utente per quanto riguarda il dimensionamento delle superfici, mentre lascia molta libertà sulla forma da adottare per la fusoliera e le superfici (il progetto lavora infatti su una corda di riferimento di forma rettangolare). È evidente che quello dato dal computer è uno dei possibili modi di progettare un aeromodello da addestramento V.V.C. e le formule adottate per il dimensionamento potranno fare arricciare il naso a qualche modellista, ma nel campo di cilindrata adottato, i risultati sono sicuramente validi e sicuri, anche perché i criteri per arrivare ai

risultati finali sono estremamente rigorosi. Se avrete la pazienza di digitare il listato, avrete per le mani un programma che vi potrà dare parecchie soddisfazioni (tra le quali quella di avere una risposta per il vostro amico che non perde occasione per spararvi a bruciapelo la classica domanda "...ma oltre ai videogiochi, che cosa sa fare il tuo computer?") La descrizione dettagliata del programma la potrete trovare nei REMarks che seguono, anch'essi strutturati a pagine e che corrispondono ai Rem che compaiono sul listato. Desideriamo comunque farvi notare subito le istruzioni contenute nelle ultime linee del listato, evidenziate dai Rem registrazione e verifica automatica: caricate per prima cosa queste linee di programma che, una volta lanciate con un Goto 9000, caricheranno la parte di listato da voi digitato fino a quel momento per poi preparare lo Spectrum alla verifica automatica. Una volta riavvolto il nastro, premete un tasto qualsiasi e il computer effettuerà la verifica del nastro: se il programma risulterà registrato correttamente, apparirà sullo schermo una segnalazione di verifica corretta e potrete andare a

mangiarvi un gelato, con l'animo sollevato dalla preoccupazione di avere buttato via delle ore per digitare un programma che il computer si rifiuta categoricamente di caricare ed eseguire. Se invece apparirà la scritta "Tape loading error" dovrete solamente limitarvi a ridare il Goto 9000 e ripetere l'operazione. A proposito dei Rem che compaiono sul listato, vi consiglio di non saltarli a piè pari quando copierete il listato perché il tempo perso per digitare i Rem vi verrà ampiamente ripagato quando dovrete cercare il solito errore di battitura che vi manda a pallino l'esecuzione del programma (errori inevitabili vista la lunghezza del listato).

Cominciate a scaldarvi le mani perché si avvicina il momento di digitare il programma: coraggio e in bocca al lupo!

## REMARKS

**1-199** - Presentazione del programma dopo aver dato un Poke 23609,100 per aumentare il volume del segnale sonoro dei tasti, si passa alla subroutine U.D.G. (Gosub 8000) per creare i simboli esponenziali e le quattro frecce necessarie

Listato 1 - Il programma Progetto aeromodello "U-Control".

```

1 REM
2 REM
3 REM
4 REM
5 REM
6 REM
7 REM
8 REM
9 REM
10 PAPER 1: INK 6: BORDER 1: B
RIGHT 1: CLS
45 POKE 23609,100
46 GO SUB 8000
47 LET a=-3: LET q=9: LET z=14
120 PRINT AT 0,0: INK 2: "##": I
NK 4: " PROGETTO E DIMENSIONAMENT
0 " : INK 2: "##"
121 PRINT AT 1,1: INK 4: "DI UN
AEROMODELLO "U-CONTROL"
125 PRINT INK 5: "Questo pro
gramma consente di realizzare
un aeromodello a volovincolato d
a addestramento partendo dalla
potenza installata, nel nostro
caso dalla cilindrata del moto
re che si vuole installare "

```

```

127 PRINT INK 7: "(di solito tra
128 3.5 cm)"
130 PRINT " NOTE AL PROCED
AMMA"
135 PRINT " P PAGINE PRECEDEN
T"
137 PRINT " A ABBANDONO"
138 PRINT " C COPY"
139 PRINT " S PER PROSEGUIRE"
148 FOR L=1 TO 216: READ D: IF
INKEY$="" THEN BEEP 1,0: NEXT L
149 FOR A=1 TO 4: PAUSE 2: BEEP
09,10: NEXT A: BEEP 2,10: RES
TORE 4000: LET A=-3: LET Z=9: LE
T B=14
200 REM
201 REM
PAGINA 1: SCELTA
DELLA CILINDRATA
202 REM
208 BORDER 0: CLS
210 PLOT 0,0: DRAW 0,175: DRAW
255,0: DRAW 0,-175
215 PLOT 100,120: DRAW -3,0: D
RAW 0,35: DRAW 30,0: DRAW 0,-35:
DRAW -3,0: DRAW 0,-3: DRAW 5,0:
DRAW 8,5: DRAW 10,0
220 DRAW -4,-8: DRAW 25,-3: DRA
W 0,-8: DRAW -25,-3: DRAW 0,-6:
DRAW -45,0: DRAW 0,20: DRAW 3,0:
DRAW 0,5
225 PRINT AT 0,1: INVERSE 1: "Pa
g. 1"
230 FOR A=130 TO 155 STEP 5: PL

```

Seguito programma Progetto aeromodello "U-Control"

```

OT 94, A: DRAW 36, 0: NEXT A
235 FOR A=116 TO 120: PLOT 109,
A: DRAW 10, 0: NEXT A
240 PRINT INK 7; AT 14, 3; "CARATT
ERISTICHE MOTORE"
245 PRINT AT 16, 1; " CILINDRATA
cm="
250 INPUT "cm="; CMC
260 PRINT CMC
270 BEEP .1, .22: PRINT AT 21, 10;
INK 5; " P A C S"
280 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1, .33: GO TO 7500
290 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1, .33: GO SUB 7000
295 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1, .33: COPY
297 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 300
298 GO TO 280
300 CLS
301 REM
302 REM PAGINA 2
303 REM PRIMO DIMENSIONAMENTO
310 PLOT 0, 0: DRAW 0, 175: DRAW
255, 0: DRAW 0, -175
315 PRINT AT 0, 1; INVERSE 1; "Pa
g. 2"
320 PRINT INK 7; AT 3, 7; "DIMENSI
ONAMENTO"
325 PRINT AT 6, 2; "MOTORE cm="; C
MC
330 LET PARA=15*CMC*CMC
331 LET PESO=INT (PARA+250)
335 LET SUPALA=INT (730+PARA)
340 BEEP .1, .22: FOR A=1 TO 50:
PRINT FLASH 1; AT 8, 2; "PESO TOTAL
E ...cm"; PESO: NEXT A
345 PRINT AT 8, 2; "PESO TOTALE g
...PESO
350 BEEP .1, .22: FOR A=1 TO 50:
PRINT FLASH 1; AT 10, 2; "SUPERFICI
E ALARE cm²"; SUPALA: NEXT A
355 PRINT AT 10, 2; "SUPERFICIE A
LARE cm²"; SUPALA
360 PRINT AT 12, 1; "Verra' press
come campione "; AT 13, 1; "un'ala
a pianta rettangolare"
361 INK 7; PLOT 58, 48: DRAW 120
0: DRAW 0, -25: DRAW -120, 0: DRA
W 0, 25: INK 6
370 BEEP .1, .22: PRINT AT 21, 10;
INK 5; " P A C S"
380 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1, .33: GO TO 7500
385 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1, .33: GO SUB 7000
390 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1, .33: COPY
395 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 400
397 GO TO 380
400 CLS
401 REM
402 REM PAGINA 3
403 REM SCELTA ALLUNGAMENTO
404 REM DELL'ALA
405 REM
410 PLOT 0, 0: DRAW 0, 175: DRAW

```

```

255, 0: DRAW 0, -175
415 PRINT AT 0, 1; INVERSE 1; "Pa
g. 3"
420 PRINT AT 2, 5; INK 7; "SCELTA
ALLUNGAMENTO"; AT 4, 1; INK 4; "L=
(Apertura alare)"; Superficie"
435 PRINT AT 10, 1; "L=5"
440 PLOT 60, 100: DRAW 0, -25: DR
AW 130, 0: DRAW 0, 25: DRAW -130, 0
445 PRINT AT 15, 1; "L=6"
450 PLOT 60, 60: DRAW 0, -25: DRA
W 155, 0: DRAW 0, 25: DRAW -155, 0
455 INPUT "L="; LAM
457 BEEP .1, 0
465 IF LAM=5 THEN PRINT INVERSE
1; AT 10, 1; "L=5"
468 IF LAM=6 THEN PRINT INVERSE
1; AT 15, 1; "L=6"
470 BEEP .1, .22: PRINT AT 21, 10;
INK 5; " P A C S"
480 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1, .33: GO TO 7500
485 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1, .33: GO SUB 7000
490 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1, .33: COPY
495 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 500
497 GO TO 480
500 CLS
501 REM
502 REM PAGINA 4
503 REM DIMENSIONI FINALI ALA
504 REM
510 PLOT 0, 0: DRAW 0, 175: DRAW
255, 0: DRAW 0, -175
515 PRINT AT 0, 1; INVERSE 1; "Pa
g. 4"
517 PRINT AT 2, 4; INK 7; "DIMENS
IONAMENTO ALA"
520 PRINT AT 4, 2; "Superficie...
...cm²"; SUPALA
523 PRINT AT 5, 2; "Allungamento.
...L"; LAM
525 LET APALA=INT (SQR (LAM*SUP
ALA))
527 BEEP .1, 0: FOR a=1 TO 50: P
RINT FLASH 1; AT 6, 2; "Apertura al
are...cm"; APALA: NEXT A
530 PRINT AT 6, 2; "Apertura alar
e...cm"; APALA
532 LET CORALA=SUPALA/APALA
533 LET CORAL=INT (CORAL*(CORAL
A): LET CORIN=INT (CORAL*100): L
ET CORIND=CORIN/100: LET CORALA=
(INT CORALA)+CORIND
536 BEEP .1, 0: FOR a=1 TO 50: P
RINT FLASH 1; AT 7, 2; "Corda media
...cm"; CORALA: NEXT A
537 PRINT AT 7, 2; "Corda media..
...cm"; CORALA
540 PRINT AT 9, 2; "Profilo: Pian
o convesso"
543 PLOT 48, 83: DRAW 20, 5, -.2:
DRAW 15, 1: DRAW 70, -13, -.2: DRAW
-100, 0: DRAW -5, 2: DRAW -1, 3
545 PRINT AT 13, 1; INK 4; " Tip
o:
1 CLARK Y"
546 PRINT AT 14, 1; INK 4; "
2 CLARK X"
547 PRINT INK 4; AT 15, 1; "
3 SAINT CYR 52"
548 LET B$="

```



nella quotatura del disegno. Nella riga 198 si trovano le istruzioni di lettura e di esecuzione del brano musicale, letto nei Data digitati a partire dalla linea 4000. Durante l'esecuzione l'istruzione Inkey\$ controlla se un tasto è stato premuto per passare immediatamente alla pagina seguente, prima della fine del brano.

**200-298** - Pagina 1: scelta della cilindrata. Dopo il tracciamento del disegno del motore (linee 215-235), viene richiesta la cilindrata del motore (CMC). A partire dalla linea 280 possiamo trovare le istruzioni che eseguono le quattro opzioni (Copy, Pagina precedente, Pagina seguente, Abbandono). Ritroveremo questo gruppo di istruzioni alla fine di ogni blocco di programma corrispondente a una pagina video.

**300-397** - Pagina 2: primo dimensionamento. Nella linea 330 viene calcolato il parametro che servirà al dimensionamento del peso e della superficie alare, parametro ottenuto moltiplicando per 15 il quadrato della cilindrata. Nella linea 361 viene disegnato un rettangolo che rappresenta la vista in pianta dell'ala presa come campione. Dalla linea 370 seguono le opzioni.

**400-497** - Pagina 3: scelta dell'allungamento dell'ala. Dopo la visualizzazione delle due possibili configurazioni di allungamento e le conseguenze sul disegno dell'ala (linee 440-450) viene chiesto l'Input dell'allungamento lambda (Lam). Le linee 465-468 evidenziano la scelta fatta scrivendo, con caratteri inversi, il valore di lambda di fianco al disegno dell'ala precedentemente eseguito. Seguono le quattro opzioni (linee 480-497).

**500-597** - Pagina 4: dimensioni dell'ala. Dopo essere stati calcolati vengono visualizzati i valori delle dimensioni dell'ala. Nelle linee 532-533 - il valore della corda alare Corala viene prima calcolato con tutti i decimali, poi, questi ultimi, vengono ridotti a due soli.

Per fare ciò si è isolata la parte decimale di Corala, moltiplicata per cento e isolata di nuovo la parte intera ottenuta da quest'ultima operazione (ottenendo così i due primi decimali come interi); questi due numeri, divisi nuovamente per cento (e pertanto ritornati numeri decimali), sono stati aggiunti alla parte intera di Corala (a qualcuno potrà sembrare poco chiaro: ha ragione. In prati-

ca però la cosa è molto semplice: fate un bel respiro profondo e rileggete di nuovo il tutto con calma). Ora che sapete tutto dell'ala, non resta che scegliere uno dei tre profili alari (linee 550-555): a seconda della scelta, il programma andrà a una delle subroutine di visualizzazione e stampa delle coordinate del profilo, collocate a partire dalla linea 3000. La linea 557 cancella con due stringhe di 32 spazi la richiesta di opzioni, prima di stampare la schermata sulla printer.

**600-697** - Pagina 5: disegno quotato dell'ala. Con un ciclo For-Next situato alla linea 610 viene disegnata la vista in pianta dell'ala rettangolare, dopo di che vengono eseguite le quotature mediante i caratteri U.D.G. caricati all'inizio del programma.

**700-797** - Pagina 6: dimensionamento del piano di coda e scelta del suo allungamento. Dopo aver calcolato la superficie dello stabilizzatore e del timone orizzontale, il programma stampa i risultati e richiede l'allungamento del piano di coda, con le stesse modalità adottate per l'ala. Anche qui, come nell'ala, a fianco del disegno in pianta dello stabilizzatore, apparirà il valore, scritto con

```

550 INPUT "VUOI IL PROFILO (1/2
3) O NESSUNO DI QUESTI (0) ";PA
551 IF PRO=3 THEN PRINT FLASH 1
; AT 15,8; "-"
552 IF PRO=2 THEN PRINT AT 14,8
; FLASH 1; "-"
553 IF PRO=1 THEN PRINT FLASH 1
; AT 13,8; "-"
554 IF PRO=0 THEN GO TO 570
555 IF PRO=3 OR PRO=2 THEN FOR
A=1 TO 5: BEEP .03,22: NEXT A: G
O TO 550
556 PRINT AT 20,1; "PREMI UN TAS
TO PER PROSEGUIRE"; AT 21,1; "OPP
URE C PER COPIARE"
557 IF INKEY$="0" OR INKEY$="C"
THEN BEEP .1,33: PRINT AT 20,0;
0$:0$: COPY
558 IF INKEY$="" THEN GO TO 556
559 IF PRO=3 THEN GO SUB 3000
560 IF PRO=2 THEN GO SUB 3200
561 IF PRO=1 THEN GO SUB 3100
570 BEEP .1,22: PRINT AT 21,10;
INR 5; "P A C S"
580 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33: GO TO 7500
585 IF INKEY$="A" OR INKEY$="a"
THEN BEEP .1,33: GO SUB 7000
590 IF INKEY$="0" OR INKEY$="C"
THEN BEEP .1,33: COPY
595 IF INKEY$="S" OR INKEY$="s"
THEN GO TO 600

```

```

597 GO TO 580
600 CLG : PLOT 0,0: DRAW 0,175:
DRAW 255,0: DRAW 0,-175
601 REM
602 REM
603 REM
        PAGINA 5
        DISEGNO QUOTATO
        DELL'ALA
604 REM
605 PRINT AT 0,1; INVERSE 1;"Pa
9,5"
610 FOR A=144 TO 112 STEP -1: P
LOT 56,A: DRAW 175,0: NEXT A
612 PLOT 55,144: DRAW -12,0: DR
AW 0,-32: DRAW 12,0: PRINT OVER
1; AT 3,5; "Y": PRINT AT 8,5; "A"
PLOT 43,160: DRAW 0,-62: PRINT A
T 5,0: OVER 1; INK 7; CORALA
615 PRINT AT 9,7; "4": AT 9,28; "4"
PLOT 56,112: DRAW 0,-13: DRAW
175,0: DRAW 0,13: PRINT AT 8,15
; INK 7; APALA
620 FOR A=56 TO 51 STEP -1: PLO
T 56,A: DRAW 175,0: NEXT A
620 BEEP .1,22: PRINT AT 21,10;
INR 5; "P A C S"
680 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33: GO TO 7500
685 IF INKEY$="A" OR INKEY$="a"
THEN BEEP .1,33: GO SUB 7000
690 IF INKEY$="0" OR INKEY$="C"
THEN BEEP .1,33: COPY
695 IF INKEY$="S" OR INKEY$="s"
THEN GO TO 700

```

Seguito programma Progetto aeromodello "U-Control"

```

697 GO TO 680
700 CLS : PLOT 0,0 : DRAW 0,175 :
DRAW 255,0 : DRAW 0,-175
701 REM
702 REM PAGINA 6
703 REM DIMENSIONAMENTO DEL
PIANO DI CODA
704 REM
705 PRINT AT 0,1 : INVERSE 1 ; "Pa
s.b"
710 PRINT AT 2,1 : INK 7 ; "DIMENS
IONAMENTO PIANO DI CODA"
715 LET SUPIA=SUPALA*.25
716 LET PARMO=INT (SUPIA*.33)
717 BEEP .1,0 : FOR a=1 TO 50 : P
RINT FLASH 1 : AT 4,2 : "Superf. pia
no coda cm" : SUPIA : NEXT a
720 PRINT AT 4,2 : "Superf. piano
coda cm" : SUPIA
721 FOR a=0 TO 50 : PRINT FLASH
1 : AT 5,2 : "Sup. stabilizzatore cm" :
PARMO : NEXT a
722 PRINT AT 5,2 : "Sup. stabiliz
zatore cm" : PARMO
723 PRINT AT 7,7 : "ALLUNGAMENTO"
724 PRINT AT 11,1 : "L=4"
730 PLOT 60,100 : DRAW 0,-27 : DR
AW 100,0 : DRAW 0,27 : DRAW -100,0
735 PRINT AT 15,1 : "L=5"
740 PLOT 60,60 : DRAW 0,-25 : DR
AW 130,0 : DRAW 0,25 : DRAW -130,0
745 INPUT "ALLUNGAMENTO L=" : LAM
P
750 IF LAMP=4 THEN PRINT INVERS
E 1 : AT 11,1 : "L=4"
753 IF LAMP=5 THEN PRINT INVERS
E 1 : AT 15,1 : "L=5"
770 BEEP .1,22 : PRINT AT 21,10 :
INK 5 : " P A C S"
780 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33 : GO TO 7500
785 IF INKEY$="A" OR INKEY$="a"
THEN BEEP .1,33 : GO SUB 7000
790 IF INKEY$="C" OR INKEY$="c"
THEN BEEP .1,33 : COPY
795 IF INKEY$="S" OR INKEY$="s"
THEN GO TO 800
797 GO TO 780
800 CLS : PLOT 0,0 : DRAW 0,175 :
DRAW 255,0 : DRAW 0,-175
801 REM
802 REM PAGINA 7
803 REM DIMENSIONI FINALI DEL
PIANO DI CODA
804 REM
805 PRINT AT 0,1 : INVERSE 1 ; "Pa
s.7"
810 PRINT AT 2,3 : INK 7 ; "DIMENS
IONI PIANO DI CODA"
820 PRINT AT 5,2 : "Superf. piano
coda cm" : SUPIA
825 LET APIA=INT (SQR (LAMP*SUP
IA))
826 PRINT AT 6,2 : "Allungamento.
L" : LAMP
827 BEEP .1,0 : FOR a=1 TO 50 : P
RINT FLASH 1 : AT 7,2 : "Apertura pi
ano coda cm" : APIA : NEXT a
830 PRINT AT 7,2 : "Apertura pian
o coda cm" : APIA

```

```

832 LET COAPIA=SUPIA/APIA
833 LET COAPI=COAPIA*INT (COAPI
A) : LET COAPN=INT (COAPI*100) : L
ET COAPINO=COAPN/100 : LET COAPIA
=(INT COAPIA)+COAPINO
839 BEEP .1,0 : FOR A=1 TO 50 : P
RINT AT 8,2 : FLASH 1 : "Corda medi
a.....cm" : COAPIA : NEXT A
840 PRINT AT 8,2 : "Corda media..
.....cm" : COAPIA
855 PRINT AT 13,1 : "Profilo: Pia
no-convesso (usate) : AT 14,1 : "lo
stesso tipo usato per l'ala" : OV
ER 1 : " )"
860 PLOT 48,40 : DRAW 20,5,-.2 :
DRAW 15,1 : DRAW 70,-13,-.2 : DRAW
-100,0 : DRAW -4,2 : DRAW -1,3
870 BEEP .1,22 : PRINT AT 21,10 :
INK 5 : " P A C S"
880 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33 : GO TO 7500
885 IF INKEY$="A" OR INKEY$="a"
THEN BEEP .1,33 : GO SUB 7000
890 IF INKEY$="C" OR INKEY$="c"
THEN BEEP .1,33 : COPY
895 IF INKEY$="S" OR INKEY$="s"
THEN GO TO 900
897 GO TO 880
900 CLS : PLOT 0,0 : DRAW 0,175 :
DRAW 255,0 : DRAW 0,-175
901 REM
902 REM PAGINA 8
903 REM DISEGNO QUOTATO DEL
PIANO DI CODA
904 REM
905 PRINT AT 0,1 : INVERSE 1 ; "Pa
s.8"
915 FOR A=144 TO 120 STEP -1 : P
LOT 32,A : DRAW 135,0 : NEXT A
916 FOR A=118 TO 107 STEP -1 : P
LOT 32,A : DRAW 135,0 : NEXT A
917 PLOT 32,144 : DRAW 0,11 : DR
AW 135,0 : DRAW 0,-11 : PRINT AT 2,
4 : "A" : PRINT AT 2,20 : "Y" : PRINT
AT 1,10 : INK 7 : APIA
930 PLOT 32,144 : DRAW -13,0 : DR
AW 0,-37 : DRAW 13,0 : PRINT OVER
1 : AT 3,2 : "Y" : PRINT AT 9,2 : "A" :
PLOT 19,160 : DRAW 0,69 : LET H$=
STR$ COAPIA : LET H=LEN H$
931 FOR A=1 TO H : PRINT INK 7 : A
T A+3,1 : H$(A TO A) : NEXT A
935 FOR A=70 TO 65 STEP -1 : PLO
T 32,A : DRAW 135,0 : NEXT A
940 PLOT 205,142 : DRAW 0,-35 : P
LOT 204,143 : DRAW 0,-30 : PLOT 20
3,144 : DRAW 0,-25
941 PLOT 202,143 : DRAW 0,-20 : P
LOT 201,140 : DRAW 0,-12
944 PLOT OVER 1 : 203,119 : DRAW 0
VER 1 : 16,0 : DRAW 0,-12 : DRAW -14
0 : PRINT AT 9,27 : "A" : PLOT 219,
40 : DRAW 0,43
945 PRINT AT 6,27 : "Y"
950 LET STAB=(COAPIA/3) : LET ST
A=STAB*INT (STAB) : LET STAB=INT
(STA*100) : LET STAB=INT (STAB)+(
STAB/100)
960 PRINT OVER 1 : INK 7 : AT 7,28
: STAB
970 BEEP .1,22 : PRINT AT 21,10 :
INK 5 : " P A C S"
980 IF INKEY$="P" OR INKEY$="p"

```

```

THEN BEEP .1,33: GO TO 7500
985 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33: GO SUB 7000
990 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33: COPY
995 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1000
997 GO TO 980
1000 CLS: PLOT 0,0: DRAW 0,175:
DRAW 255,0: DRAW 0,-175
1001 REM
1002 REM PAGINA 9
1003 REM DIMENSIONI DERIVA
1004 REM
1005 PRINT AT 0,1: INVERSE 1;"Pa
9.9"
1010 PRINT AT 2,1: INK 7;" 0
DIMENSIONI DERIVA"
1013 LET SUDERI=INT (SUPA/3.5+.
5)
1015 LET APERI=INT (SQR (.8*SUPI
A))
1017 LET CORDERI=INT ((2*SUDERI/
APERI)+.5)
1019 LET CORDEMI=.5+INT (CORDERI
*.4)
1021 LET CORDEMA=1+INT (CORDERI*
.6)
1022 PRINT INK 4;AT 3,1:"La form
a della deriva non ha";AT 4,1:"m
olla importanza nel V.V.C.";AT 5
,1:"Diamo qui una possibile";AT
6,1:"configurazione:"
1023 LET SUDERIC=.5+INT ((CORDEM
I+CORDEMA)*APERI/2)
1025 BEEP .1,0: FOR A=1 TO 50: P
RINT AT 8,1: FLASH 1;"Superficie
deriva...cm²";SUDERIC: NEXT A
1027 PRINT AT 8,1:"Superficie de
riva...cm²";SUDERIC
1028 BEEP .1,0: FOR A=1 TO 50: P
RINT AT 9,1: FLASH 1;"Apertura d
eriva...cm²";APERI: NEXT A
1029 PRINT AT 9,1:"Apertura deri
va...cm²";APERI
1030 BEEP .1,0: FOR A=1 TO 50: P

```

```

RINT AT 10,1: FLASH 1;"Corda min
ima deriva...cm";CORDEMI: NEXT A
1031 PRINT AT 10,1:"Corda minima
deriva...cm";CORDEMI
1035 BEEP .1,0: FOR A=1 TO 50: P
RINT AT 11,1: FLASH 1;"Corda mas
sima deriva...cm";CORDEMA: NEXT A
1036 PRINT AT 11,1:"Corda massim
a deriva...cm";CORDEMA
1040 LET PARTEMO=SUDERI/3.5
1041 BEEP .1,0: FOR A=1 TO 50: P
RINT AT 12,1: FLASH 1;"Sup. parte
e mobile...cm²";INT (PARTEMO+1
): NEXT A
1042 PRINT AT 12,1;"Sup. parte m
obile...cm²";INT (PARTEMO+1)
1050 PRINT AT 15,1:"Profilo: Bic
onvesso simmetrico"
1055 PLOT 60,30: DRAW 20,5,-.2:
DRAW 15,1: DRAW 70,-.8,-.2: DRAW
-70,-.8,-.17: DRAW -15,1: DRAW -2
0,5,-.2: DRAW 0,3,-3
1070 BEEP .1,22: PRINT AT 21,10:
INK 5;" P A C S"
1080 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33: GO TO 7500
1085 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33: GO SUB 7000
1090 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33: COPY
1095 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1100
1097 GO TO 1080
1100 CLS: PLOT 0,0: DRAW 0,175:
DRAW 255,0: DRAW 0,-175
1101 REM
1102 REM PAGINA 10
1103 REM DISEGNO QUOTATO DELLA
DERIVA
1104 REM
1105 PRINT AT 0,24: INVERSE 1;"P
ag. 10"
1110 LET COR=22: FOR A=144 TO 58
STEP -1: LET COR=COR+.4: PLOT 5
0,A: DRAW COR,0: PLOT 200,A: DRA
W 2,0: NEXT A

```

caratteri inversi, dell'allungamento scelto. Seguono le istruzioni per le quattro opzioni.

**800-897** - Pagina 7: dimensioni finali del piano di coda. Questo blocco di programma stampa tutti i dati necessari per il disegno del piano di coda. Anche in questo caso per avere il valore della corda media con una precisione di solo due decimali, si è usato lo stesso algoritmo impiegato per l'ala.

**900-997** - Pagina 8: disegno quotato del piano di coda. Poiché il piano di coda è formato da due superfici distinte, per la vista dall'alto si sono usati due diversi cicli For-Next per il tracciamento del disegno, in modo che fosse evidenziata la fessura esistente tra i due piani (915-916). Nelle linee 930-931 per motivi di

spazio sul disegno, viene fatto stampare verticalmente il valore della corda media: nella variante K si inserisce il valore di Corpia con l'istruzione  $K\$ = \text{Str}\$ \text{Copia}$ , poi si calcola la lunghezza della stringa con l'istruzione  $K = \text{Len } K\$$ ; la linea 931 ha poi stampato il valore dall'alto verso il basso prelevando un carattere alla volta dalla variabile  $K\$$  mediante l'istruzione  $K\$ = K\$ (a \text{ To } a)$ . Nella linea 950 viene calcolato il valore della lunghezza del segmento di due cifre decimali.

**1000-1097** - Pagina 9: dimensioni dell'impennaggio verticale. Vengono calcolati tutti i dati necessari per il dimensionamento di un classico tipo di impennaggio verticale: quello trapezio-rettangolo. Nelle linee 1013-1015 sono calco-

late l'apertura e la superficie in funzione della superficie del piano di coda, dopo di che viene calcolata prima la corda media (1017) e, successivamente, la corda minima e massima della deriva. Nella linea 1040 viene calcolata la superficie del timone verticale, valore che viene aumentato di uno in fase di visualizzazione. Nella linea 1055 viene disegnato un profilo biconvesso-simmetrico. Seguono le solite quattro opzioni del programma.

**1100-1197** - Pagina 10: disegno quotato dell'impennaggio verticale. Nella linea 1114 viene disegnato un trapezio rettangolo plottando con il solito ciclo For-Next, un punto dalle coordinate (50,144) alle coordinate (50,58); per ogni punto viene tracciato un segmento orizzontale

Seguito programma Progetto aeromodello "U-Control"

```

1118 PLOT 199,58: DRAW 0,32: PLO
T 203,58: DRAW 0,32
1120 PRINT AT 4,4: "A": AT 13,4: "Y
": PLOT 35,58: DRAW 0,86: DRAW 2
4,0: PLOT 35,58: DRAW 24,0: PRIN
T AT 10,2: INK 7: APERI
1127 PRINT AT 2,5: "←": PLOT
45,155: DRAW 27,0: DRAW 0,-12: P
LOT 30,155: DRAW 57,0: DRAW -37,
0: DRAW 0,-12: PRINT AT 1,6: INK
7: CODEM1
1129 PRINT AT 16,7: "←": PLO
T 50,43: DRAW 57,0: DRAW 0,15: P
LOT OVER 1,50,43: DRAW OVER 1,0,
15: PLOT OVER 1,50,44: DRAW OVER
1,-2,0
1135 PRINT AT 15,9: INK 7: CODEM
A
1140 PLOT 62,144: DRAW OVER 1,0,
-86
1155 PLOT 64,20: DRAW 44,0
1156 PLOT 65,20: DRAW 39,0: PLOT
65,18: DRAW 39,0
1157 PLOT 64,21: DRAW 43,0: PLOT
64,19: DRAW 43,0
1158 PLOT 70,23: DRAW 30,0: PLOT
70,17: DRAW 30,0
1160 PLOT 64,20: DRAW -35,-13
1161 PLOT 64,21: DRAW -13,-5: PL
OT 64,19: DRAW -11,-4
1165 PLOT 64,20: DRAW -35,0: PLO
T 46,24: DRAW 0,-16,3
1167 PRINT INK 7: AT 20,2: "9."
1170 BEEP .1,22: PRINT AT 21,10:
INK 5: "P A C S"
1180 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33: GO TO 7500
1185 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33: GO SUB 7000
1190 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33: COPY
1195 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1200
1197 GO TO 1180
1200 CLS: PLOT 0,0: DRAW 0,175:
DRAW 255,0: DRAW 0,-175
1201 REM
1202 REM PAGINA 11
1203 REM POSIZIONE SUPERFICI E
MOTORE
1204 REM
1205 PRINT AT 0,1: INVERSE 1: "Pa
g. 11"
1210 PRINT AT 3,1: INK 7: "NOTE P
OSIZIONAMENTO SUPERFICI"
1220 PRINT AT 5,1: OVER 1: "Calet
tamento ala-piano coda: 0°"
1222 PRINT AT 6,1: OVER 1: "Calet
tamento motore-asse: 0°"
1224 PRINT AT 8,1: "Il timone del
la deriva deve": AT 9,1: OVER 1:
"essere fissato inclinato di
8-10°": INK 7: "all'esterno dell
a circonferenza di volo": IN
K 4: "(vedi Pag. 10)"
1230 PRINT AT 13,1: "ESCURSIONE M
AX STABILIZZATORE"
1235 PRINT AT 16,6: "10°-15° a ca
brare"
1237 PRINT AT 18,6: "5° a pi

```

```

cchiare"
1270 BEEP .1,22: PRINT AT 21,10:
INK 5: "P A C S"
1280 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33: GO TO 7500
1285 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33: GO SUB 7000
1290 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33: COPY
1295 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1300
1297 GO TO 1280
1300 CLS: PLOT 0,0: DRAW 0,175:
DRAW 255,0: DRAW 0,-175
1301 REM
1302 REM PAGINA 12
1303 REM SCELTA E DIMENSIONA
MENTO DELLA SQUADRETTA
DI COMANDO
1304 REM
1305 PRINT AT 0,1: INVERSE 1: "Pa
g. 12"
1310 PRINT AT 2,3: INK 7: "SCELTA
E DIMENSIONAMENTO": AT 3,4: "SQUA
DRETTA DI COMANDO"
1315 PLOT 40,120: DRAW 24,0: DRA
W 0,16: DRAW 9,0: DRAW 0,-16: DR
AW 24,0: DRAW 0,-9: DRAW -57,0:
DRAW 0,9: PRINT AT 6,15: "TIPO A"
1317 PLOT 40,70: DRAW 0,9: DRAW
28,16: DRAW 1,0: DRAW 28,-16: DR
AW 0,-9: DRAW -57,0
1318 PRINT AT 11,15: "TIPO B"
1320 INPUT "TIPO A o B?": A$
1325 IF A$="A" OR A$="a" THEN GO
TO 3500
1330 IF A$="B" OR A$="b" THEN GO
TO 3600
1333 IF A$<>"B" OR A$<>"b" OR A$
<>"a" OR A$<>"A" THEN GO TO 1325
1370 BEEP .1,22: PRINT AT 21,10:
INK 5: "P A C S"
1380 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33: GO TO 7500
1385 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33: GO SUB 7000
1390 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33: COPY
1395 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1400
1397 GO TO 1380
1400 CLS: PLOT 0,0: DRAW 0,175:
DRAW 255,0: DRAW 0,-175
1401 REM
1402 REM PAGINA 13
1403 REM DIMENSIONI FISSE DELLA
FUSOLIERA
1404 REM
1405 PRINT AT 0,1: INVERSE 1: "Pa
g. 13"
1410 PRINT AT 1,2: INK 7: "NOTE P
ER IL DISEGNO DELLA": AT 2,2: "FUS
OLIERA E IL CENTRAGGIO"
1415 PRINT INK 4: AT 4,1: "Per il
disegno della fusoliera": AT 5,1:
"attenzione SCRUPOLOSAMENTE a"
1417 PRINT INK 4: AT 6,1: "queste
indicazioni:"
1420 LET BAAL=INT (APALA/2+4)
1430 BEEP .1,22: FOR A=1 TO 50:
PRINT FLASH 1: AT 10,2: "Braccio d
i leva min. cm": BAAL: NEXT A:
1432 PRINT AT 10,2: "Braccio di l

```

la cui lunghezza  $Cor = 22$  è incrementata volta per volta di 0,4 pixel ( $Cor = Cor + 0,4$ ).

Le linee 1140-1167 disegnano la vista dall'alto degli impennaggi, evidenziando l'angolo formato dal timone verticale con la deriva.

**1200-1297** - Pagina 11: calettamento ala-piano di coda e calettamento motore-asse dell'aereo. Questo blocco di programma dà alcuni dati fissi per il funzionamento e la messa a punto delle parti fin qui elaborate. Vi consigliamo di inserire il numero esatto di spazi per ogni stringa, per un corretto centraggio delle informazioni sulla pagina video.

**1300-1397** - Pagina 12: scelta e dimensionamento della squadretta di comando. Dopo il titolo, vengono disegnati i

due tipi più diffusi di squadrette di comando (linee 1315-1318) seguite dall'Input per la scelta del modello da quotare. A seconda della scelta, le linee 1325-1330 indirizzeranno alla routine di pulizia dello schermo e tracciamento del disegno quotato, situate a partire dalla linea 3500. La linea 1333 ha il solo compito di controllare che non venga dato un Input errato.

**1400-1497** - Pagina 13: dimensioni a cui attenersi durante il disegno della fusoliera. Questo blocco di programma, dopo aver stampato le spiegazioni necessarie, calcola il braccio di leva Bral (linea 1420) e la posizione, in percentuale, della corda e in centimetri del centro di gravità CG e della posizione della squadretta CS (1442-1443).

Nelle linee 1460 e 1461 vengono riportate alcune indicazioni sulla posizione dei cavi che verranno poi visualizzate sotto forma di disegno nella pagina seguente.

**1500-1597** - Pagina 14: vista in pianta dell'aeromodello per la posizione dei cavi e della squadretta di comando. La linea 1525 traccia la vista dall'alto della semifusoliera sinistra mentre la linea seguente ha il compito di tracciare l'asse della fusoliera come una linea tratteggiata. La linea 1530 traccia una possibile configurazione dell'ala (trapezia), la linea 1535 traccia gli impennaggi orizzontali. I caratteri grafici che compaiono nelle linee 1550 e 1551 sono stati ottenuti stampando con caratteri inversi il simbolo del punto (.) e lo spazio. Le linee 1557-1558 tracciano i cavi di co-

```

eva min. cm":BRAL
1440 BEEP .1,22:FOR A=1 TO 50:
PRINT FLASH 1:AT 11,1:"Centro d
l gravità CG 25% Corda":NEXT a
1441 PRINT AT 11,1:"Centro di gr
avità CG 25% Corda"
1442 LET CG=CORALA*.25
1443 LET SQ=CORALA*.35
1445 PRINT AT 12,1:("):CG;"cm da
l bordo d'attacco)
1450 BEEP .1,22:FOR A=1 TO 50:
PRINT FLASH 1:AT 13,1:"Squadret
ta al 35% Corda":NEXT a
1451 PRINT AT 13,1:"Squadretta a
l 35% Corda"
1455 PRINT AT 14,1:("):SQ;"cm da
l bordo d'attacco)
1460 PRINT AT 17,1:INK 4:"Lungh
ezza cavi m.13/16"
1461 PRINT AT 18,1:INK 4:"Cavi
inclinati all'indietro":AT 19,1:
"d, 8°/10°"
1470 BEEP .1,22:PRINT AT 21,10;
INK 5:"P A C S"
1480 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33:GO TO 7500
1485 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33:GO SUB 7000
1490 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33:COPY
1495 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1500
1497 GO TO 1480
1500 CLS:PLOT 0,0:DRAW 0,175:
DRAW 255,0:DRAW 0,-175
1501 REM
1502 REM PAGINA 14
1503 REM VISTA DI PIANTA DELLO
AEREO CON POSIZIONE
SQUADRETTA E CAVI
1504 REM
1505 PRINT AT 0,1:INVERSE 1:"Pa
g. 14"
1525 PLOT 7,150:DRAW 10,-5,1:D
RAW 200,5,31
1527 PLOT 5,150:FOR Q=5 TO 215

```

```

STEP 10:PLOT 0,150:DRAW 5,0:P
LOT INVERSE 1,0+5,150:DRAW INVE
RSE 1,5,0:NEXT Q
1530 PLOT 59,142:DRAW 3,-120:D
RAW 37,0,5:DRAW 7,119
1535 PLOT 207,146:DRAW 3,-40:D
RAW 17,0,5:DRAW 7,45:PLOT 220
149:DRAW 0,-43
1550 PRINT AT 3,9:INK 7:OVER 1
1551 PRINT INK 7:AT 2,10:OVER 1
1555 PLOT INK 7:82,154:DRAW INK
7,50,0:DRAW INK 7,0,-2:DRAW I
NK 7,100,0:DRAW INK 7,0,1:DRAW
INK 7,-5,0
1557 PLOT 74,141:DRAW 6,-135
1558 PLOT 91,141:DRAW 6,-135
1565 PLOT 148,53:DRAW 20,5,-2:
DRAW 15,1:DRAW 70,-13,-2:DA
W -100,0:DRAW -4,2:DRAW -1,3
1566 PLOT 220,48:DRAW 0,5:DRAW
-1,0:DRAW 0,-5
1567 PLOT 222,46:DRAW 0,-15:DA
RW 0,0:DRAW 0,15
1568 PRINT AT 19,14:"h SQUADRETT
A ↑":AT 20,21:"mm12,13"
1570 BEEP .1,22:PRINT AT 21,10;
INK 5:"P A C S"
1580 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33:GO TO 7500
1585 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33:GO SUB 7000
1590 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33:COPY
1595 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1500
1597 GO TO 1580
1600 CLS:PLOT 0,0:DRAW 0,175:
DRAW 255,0:DRAW 0,-175
1601 REM
1602 REM PAGINA 15
1603 REM NOTE FINALI PER LA
COSTRUZIONE
1604 REM
1605 PRINT AT 0,1:INVERSE 1:"Pa

```

Seguito programma Progetto aeromodello "U-Control"

```

5.15"
1609 PRINT AT 2,9; FLASH 1;"NOTE
FINALI"
1615 PRINT INK 4;AT 4,1;"Usate u
n'elica con passo corto"
1617 PRINT INK 4;AT 5,1;"Il serb
atoio deve contenere ";AT 6,1;
"miscela per 2/3 min. di volo"
1618 PRINT INK 4;AT 7,1;"Per la
barra di comando usate";AT 8,1;"
filo di acciaio armonico di ";AT
9,1;"1.5/2 mm. Se la velocita'
di ";AT 10,1;"volo fosse eccessi
va, non";AT 11,1;"aumentate il p
eso del modello";AT 12,1;" ma ri
ducete il passo della"
1620 PRINT INK 4;AT 13,1;"elica
o aumentate la superficie";AT 14,
1;"alare"
1670 BEEP .1,22; PRINT AT 21,10;
INK 5;" P A C S"
1680 IF INKEY$="P" OR INKEY$="p"
THEN BEEP .1,33; GO TO 7500
1685 IF INKEY$="a" OR INKEY$="A"
THEN BEEP .1,33; GO SUB 7000
1690 IF INKEY$="c" OR INKEY$="C"
THEN BEEP .1,33; COPY
1695 IF INKEY$="s" OR INKEY$="S"
THEN GO TO 1700
1697 GO TO 1680
1700 CLS; PLOT 0,0; DRAW 0,175;
DRAW 255,0; DRAW 0,-175
1701 REM
1702 REM ULTIMA PAGINA
1703 REM
1710 PRINT FLASH 1; INK 7; PAPER
2; BRIGHT 1;AT 10,10;"BUON LAVORO"
1720 RESTORE 4000
1735 LET A=-3; LET Z=14; LET Q=9
FOR L=1 TO 216; READ Q; IF INK
EY$="" THEN BEEP .1,0; NEXT L
1740 FOR A=1 TO 4; PAUSE 2; BEEP
.09,10; NEXT A; BEEP .2,10; RES
TORE 4000; LET A=-3; LET Z=9; LE
T Q=14
1770 STOP
1790 REM
1791 REM INIZIO SOTTOPAGINE
1795 REM
2880 STOP
2900 REM
2901 REM COORDINATE PROFILI
2902 REM
2904 REM PROFILO SAINT CYR 52
2905 REM
3000 CLS; PRINT AT 0,1; INVERSE
1;"Pas 4 bis"; PRINT AT 1,5;"PA
OFILO SAINT CYR 52"
3002 PRINT AT 3,1;"X 0 2.5
5 7.5 10
3005 PRINT AT 4,1; INK 4;"Ys 2.5
4.24 5.5 6.47 6.8"
3010 PRINT AT 5,1; INK 5;"Yi 2.5
1.66 1.3 0.87 0.8"
3020 PRINT AT 7,1;"X 10 20
30 40 50
3025 PRINT AT 8,1; INK 4;"Ys 6.8
9.10 10 9.8 8.8"
3030 PRINT AT 9,1; INK 5;"Yi 0.8

```

```

0.02 0 0 0 "
3035 PRINT AT 11,1;"X 60 70
80 90 100
3040 PRINT AT 12,1; INK 4;"Ys 7.
5 5.5 3.9 2.1 0
3045 PRINT AT 13,1; INK 5;"Yi 0
0 0 0 0
3050 PRINT AT 18,1;"UNENDO I PUN
TI INDICATI DALLA TABELLA SI O
TTIENE IL PROFILO"
3090 RETURN
3095 REM
3096 REM PROFILO CLARK Y
3097 REM
3100 CLS; PRINT AT 0,1; INVERSE
1;"Pas 4 bis"; PRINT AT 1,5;"PA
OFILO CLARK Y"
3102 PRINT AT 3,1;"X 0 2.5
5 7.5 10
3105 PRINT AT 4,1; INK 4;"Ys 3.6
6.43 7.8 8.79 9.56"
3110 PRINT AT 5,1; INK 5;"Yi 3.6
1.42 0.9 0.59 0.39"
3120 PRINT AT 7,1;"X 10 20
30 40 50
3125 PRINT AT 8,1; INK 4;"Ys 9.5
11.3 11.7 11.3 10.49"
3130 PRINT AT 9,1; INK 5;"Yi 0.3
0.01 0 0 0
3135 PRINT AT 11,1;"X 60 70
80 90 100
3140 PRINT AT 12,1; INK 4;"Ys 9.
1 7.43 5.2 2.8 0
3145 PRINT AT 13,1; INK 5;"Yi 0
0 0 0 0
3150 PRINT AT 18,1;"UNENDO I PUN
TI INDICATI DALLA TABELLA SI O
TTIENE IL PROFILO"
3190 RETURN
3195 REM
3196 REM PROFILO CLARK X
3197 REM
3200 CLS; PRINT AT 0,1; INVERSE
1;"Pas 4 bis"; PRINT AT 1,5;"PA
OFILO CLARK X"
3202 PRINT AT 3,1;"X 0 2.5
5 7.5 10
3205 PRINT AT 4,1; INK 4;"Ys 4
6.64 7.95 8.92 9.68"
3210 PRINT AT 5,1; INK 5;"Yi 4
1.48 1.14 0.88 0.5"
3220 PRINT AT 7,1;"X 10 20
30 40 50
3225 PRINT AT 8,1; INK 4;"Ys 9.6
11.3 11.7 11.4 10.51"
3230 PRINT AT 9,1; INK 5;"Yi 0.5
0.03 0 0 0
3235 PRINT AT 11,1;"X 60 70
80 90 100
3240 PRINT AT 12,1; INK 4;"Ys 9.
15 7.95 5.22 2.8 0.12"
3245 PRINT AT 13,1; INK 5;"Yi 0
0 0 0 0
3250 PRINT AT 18,1;"UNENDO I PUN
TI INDICATI DALLA TABELLA SI O
TTIENE IL PROFILO"
3290 RETURN
3295 REM
3296 REM SOTTOPROGRAMMA DISEGNO
DELLE SQUADRETTE
3297 REM
3298 REM
3299 REM
3300 REM
3301 REM
3302 REM
3303 REM
3304 REM
3305 REM
3306 REM
3307 REM
3308 REM
3309 REM
3310 REM
3311 REM
3312 REM
3313 REM
3314 REM
3315 REM
3316 REM
3317 REM
3318 REM
3319 REM
3320 REM
3321 REM
3322 REM
3323 REM
3324 REM
3325 REM
3326 REM
3327 REM
3328 REM
3329 REM
3330 REM
3331 REM
3332 REM
3333 REM
3334 REM
3335 REM
3336 REM
3337 REM
3338 REM
3339 REM
3340 REM
3341 REM
3342 REM
3343 REM
3344 REM
3345 REM
3346 REM
3347 REM
3348 REM
3349 REM
3350 LET Q$="
"; FOR A=4 TO 21; BEEP

```

## ZX SPECTRUM

```

3508 1: PRINT AT A,1;Q$: NEXT A
3515 PLOT 80,90: BEEP .01,1: DRA
3524 0: BEEP .01,1: DRAW 0,16: B
3531 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3540 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3549 0: BEEP .01,1: DRAW 0,-9: BEEP
3558 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3567 0: BEEP .01,1: DRAW 0,-9
3576 0: BEEP .01,1: DRAW 0,16: B
3585 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3594 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3603 0: BEEP .01,1: DRAW 0,-9: BEEP
3612 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3621 0: BEEP .01,1: DRAW 0,-9
3630 0: BEEP .01,1: DRAW 0,16: B
3639 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3648 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3657 0: BEEP .01,1: DRAW 0,-9: BEEP
3666 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3675 0: BEEP .01,1: DRAW 0,-9
3684 0: BEEP .01,1: DRAW 0,16: B
3693 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3702 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3711 0: BEEP .01,1: DRAW 0,-9: BEEP
3720 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3729 0: BEEP .01,1: DRAW 0,-9
3738 0: BEEP .01,1: DRAW 0,16: B
3747 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3756 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3765 0: BEEP .01,1: DRAW 0,-9: BEEP
3774 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3783 0: BEEP .01,1: DRAW 0,-9
3792 0: BEEP .01,1: DRAW 0,16: B
3801 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3810 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3819 0: BEEP .01,1: DRAW 0,-9: BEEP
3828 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3837 0: BEEP .01,1: DRAW 0,-9
3846 0: BEEP .01,1: DRAW 0,16: B
3855 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3864 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3873 0: BEEP .01,1: DRAW 0,-9: BEEP
3882 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3891 0: BEEP .01,1: DRAW 0,-9
3900 0: BEEP .01,1: DRAW 0,16: B
3909 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3918 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3927 0: BEEP .01,1: DRAW 0,-9: BEEP
3936 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3945 0: BEEP .01,1: DRAW 0,-9
3954 0: BEEP .01,1: DRAW 0,16: B
3963 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3972 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
3981 0: BEEP .01,1: DRAW 0,-9: BEEP
3990 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
3999 0: BEEP .01,1: DRAW 0,-9
4008 0: BEEP .01,1: DRAW 0,16: B
4017 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4026 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4035 0: BEEP .01,1: DRAW 0,-9: BEEP
4044 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4053 0: BEEP .01,1: DRAW 0,-9
4062 0: BEEP .01,1: DRAW 0,16: B
4071 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4080 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4089 0: BEEP .01,1: DRAW 0,-9: BEEP
4098 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4107 0: BEEP .01,1: DRAW 0,-9
4116 0: BEEP .01,1: DRAW 0,16: B
4125 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4134 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4143 0: BEEP .01,1: DRAW 0,-9: BEEP
4152 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4161 0: BEEP .01,1: DRAW 0,-9
4170 0: BEEP .01,1: DRAW 0,16: B
4179 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4188 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4197 0: BEEP .01,1: DRAW 0,-9: BEEP
4206 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4215 0: BEEP .01,1: DRAW 0,-9
4224 0: BEEP .01,1: DRAW 0,16: B
4233 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4242 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4251 0: BEEP .01,1: DRAW 0,-9: BEEP
4260 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4269 0: BEEP .01,1: DRAW 0,-9
4278 0: BEEP .01,1: DRAW 0,16: B
4287 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4296 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4305 0: BEEP .01,1: DRAW 0,-9: BEEP
4314 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4323 0: BEEP .01,1: DRAW 0,-9
4332 0: BEEP .01,1: DRAW 0,16: B
4341 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4350 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4359 0: BEEP .01,1: DRAW 0,-9: BEEP
4368 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4377 0: BEEP .01,1: DRAW 0,-9
4386 0: BEEP .01,1: DRAW 0,16: B
4395 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4404 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4413 0: BEEP .01,1: DRAW 0,-9: BEEP
4422 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4431 0: BEEP .01,1: DRAW 0,-9
4440 0: BEEP .01,1: DRAW 0,16: B
4449 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4458 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4467 0: BEEP .01,1: DRAW 0,-9: BEEP
4476 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4485 0: BEEP .01,1: DRAW 0,-9
4494 0: BEEP .01,1: DRAW 0,16: B
4503 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4512 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4521 0: BEEP .01,1: DRAW 0,-9: BEEP
4530 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4539 0: BEEP .01,1: DRAW 0,-9
4548 0: BEEP .01,1: DRAW 0,16: B
4557 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4566 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4575 0: BEEP .01,1: DRAW 0,-9: BEEP
4584 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4593 0: BEEP .01,1: DRAW 0,-9
4602 0: BEEP .01,1: DRAW 0,16: B
4611 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4620 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4629 0: BEEP .01,1: DRAW 0,-9: BEEP
4638 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4647 0: BEEP .01,1: DRAW 0,-9
4656 0: BEEP .01,1: DRAW 0,16: B
4665 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4674 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4683 0: BEEP .01,1: DRAW 0,-9: BEEP
4692 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4701 0: BEEP .01,1: DRAW 0,-9
4710 0: BEEP .01,1: DRAW 0,16: B
4719 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4728 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4737 0: BEEP .01,1: DRAW 0,-9: BEEP
4746 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4755 0: BEEP .01,1: DRAW 0,-9
4764 0: BEEP .01,1: DRAW 0,16: B
4773 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4782 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4791 0: BEEP .01,1: DRAW 0,-9: BEEP
4800 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4809 0: BEEP .01,1: DRAW 0,-9
4818 0: BEEP .01,1: DRAW 0,16: B
4827 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4836 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4845 0: BEEP .01,1: DRAW 0,-9: BEEP
4854 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4863 0: BEEP .01,1: DRAW 0,-9
4872 0: BEEP .01,1: DRAW 0,16: B
4881 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4890 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4899 0: BEEP .01,1: DRAW 0,-9: BEEP
4908 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4917 0: BEEP .01,1: DRAW 0,-9
4926 0: BEEP .01,1: DRAW 0,16: B
4935 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4944 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4953 0: BEEP .01,1: DRAW 0,-9: BEEP
4962 0: BEEP .01,1: DRAW -57,0: BEEP .01,1: D
4971 0: BEEP .01,1: DRAW 0,-9
4980 0: BEEP .01,1: DRAW 0,16: B
4989 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:
4998 0: BEEP .01,1: DRAW 4,0: BEEP .01,1:

```

```

4010 DATA a,19,a,21,a,17,a,19,a,
16,a,17,a,14,a,19,a,16,a,17,a,14
,a,16,a,13,a,14
4020 DATA a,10,a,7,a,9,a,5,a,7,a
,a,8,a,2,a,7,a,4,a,5,a,2,a,4,a
,a,8,a,2,a,4,a,8,a,4,a,8,a
4030 DATA a,7,a,5,a,4,a,2,a,1,a,
,a,1,a,5,a,9,-3,a,-1,a,1,a,2,a,1,
,a,1,a,5,a,9,-3,a,-1,a,1,a,2,a,1,
,a,1,a,5,a,9,-3,a,-1,a,1,a,2,a,1,
4040 DATA a,12,a,10,a,9,a,7,a,6,
,a,7,a,9,a,10,a,2,a,4,a,6,a,7,a,6
,a,7,a,9,a,10,a,2,a,4,a,6,a,7,a,6
,a,7,a,9,a,10,a,2,a,4,a,6,a,7,a,6
4050 DATA a,7,a,5,a,4,a,2,a,1,a,
,a,1,a,5,a,9,-3,a,-1,a,1,a,2,a,1,
,a,1,a,5,a,9,-3,a,-1,a,1,a,2,a,1,
,a,1,a,5,a,9,-3,a,-1,a,1,a,2,a,1,
4060 DATA a,12,a,10,a,9,a,7,a,6,
,a,7,a,9,a,10,a,2,a,4,a,6,a,7,a,6
,a,7,a,9,a,10,a,2,a,4,a,6,a,7,a,6
,a,7,a,9,a,10,a,2,a,4,a,6,a,7,a,6
5990 STOP
6950 REM
6960 REM ABBANDONO
6970 REM
6980 REM
7000 INPUT "confermi? (s/n) ";A$
7010 IF A$="S" OR A$="s" THEN ST
OP
7020 RETURN
7400 REM
7405 REM INDIRIZZAMENTO PAGINE
7410 REM
7500 BEEP .1,4: INPUT "PAGINA N"
:PA: LET PA=(PA+1)*100
7501 IF PA=100 THEN GO TO 200
7502 GO TO PA
7600 STOP
7900 REM
7905 REM CARATTERI U.D.G.
7909 REM
8000 POKE USA "a", 0
8001 POKE USA "a", +1, BIN 01110000
8002 POKE USA "a", +2, BIN 01001000
8003 POKE USA "a", +3, BIN 00010000
8004 POKE USA "a", +4, BIN 01111000
8005 POKE USA "a", +5, BIN 00000000
8006 POKE USA "a", +6, BIN 00000000
8007 POKE USA "a", +7, BIN 00000000
8010 POKE USA "a", 0

```

mando, inclinati rispetto l'asse trasversale dell'ala. Infine le linee da 1567 a 1568 disegnano il profilo piano-convesso del piano di coda, evidenziando la levetta sul timone orizzontale che verrà collegata alla squadretta per mezzo della barra di comando. Seguono le opzioni.

**1600-1697** - Pagina 15: note finali per la costruzione. Verranno visualizzati alcuni consigli per la messa a punto del modello e per il volo. È praticamente l'ultima pagina del programma.

**1700-1795** - Schermata finale. Nella linea 1720 viene dato un Restore per poter rileggere il brano musicale "immagazzinato" nella linea 4000. La linea 1735 esegue la melodia letta tramite l'istruzione Read O; prima però vengono

reinserti nelle variabili A,Z,O i valori delle note, poiché queste variabili erano state riutilizzate nel corso del programma per altri scopi (cicli For-Next ecc.). A questo punto iniziano le sottopagine e le subroutine.

**2900-3290** - Sottopagine contenenti le tabelle dei profili alari. Non c'è molto da spiegare: l'unica cosa su cui bisogna porre una certa attenzione è il numero degli spazi da inserire tra un numero e l'altro per una corretta impaginazione (ricordiamo che per ogni X abbiamo due valore di Y).

**3490-3700** - Sottoprogramma per la pulizia parziale dello schermo e per il disegno quotato delle squadrette. Nella linea 3500 viene caricata la variabile Q con 26 spazi, poi, con un ciclo da 4 a 21,

viene pulito lo schermo della pagina 12, tranne le prime quattro righe. La linea 3515 traccia il disegno della squadretta, associando ad ogni linea un brevissimo segnale sonoro. La linea 3520 disegna i tre fori nei quali verranno agganciati i cavi e la barra di comando dell'elevatore. Le linee 3530-3590 quotano il disegno e, con il Goto 1370, si torna alla pagina 12 per le opzioni. Con la stessa tecnica le linee 3600-3613 puliscono lo schermo e disegnano il secondo tipo di squadretta, poi, visto che la quotatura è la medesima, con un Goto si torna alla linea 3520, risparmiando così un po' di memoria (e di tempo).

**3995-5990** - Data del brano musicale. Dalla linea 4000 alla 4050 sono contenute le note del brano musicale suonato





**S**e trovate frustrante avere per le mani un computer MSX e riuscire a fargli emettere solo una serie di note stonate, nonostante la sua grande potenzialità nel campo musicale, allora questo programma fa al caso vostro. Esso infatti esegue cinque brani musicali, rigorosamente scelti col metodo della casualità illogica, sfruttando il macrolinguaggio musicale sviluppato dall'istruzione Play. Non solo, ma una volta caricato questo programma, potrete avere la sadica soddisfazione di ascoltarvi la Bourré di Bach a "78 giri" o trasformare l'Inverno di Vivaldi in un "Tip-Tap". Grazie alle istruzioni Delete e Renumber potrete isolare un motivo e inserirlo in un vostro programma (ricordo che l'esecuzione dell'istruzione Play non blocca l'elaborazione, poichè i suoni sono gestiti autonomamente da un generatore di suoni a tre canali AY-3-8910). Ritornando al programma così come è listato, per utilizzarlo correttamente do-

# Musica con un MSX

Sfruttate le notevoli possibilità sonore del vostro computer

di Paolo Ferrami

vrete semplicemente inserire il numero del brano scelto così come appare sul menu, ricordandovi che potete selezionare più brani senza attendere la fine dell'esecuzione del primo: in questo modo i brani scelti verranno suonati in suc-

cessione senza soluzione di continuità. I brani programmati sono la Bourré in Mi minore di Bach, la Romanza op. 50 di Beethoven, l'Inverno (dalle "Quattro Stagioni") di Vivaldi, l'Allegretto dalla sinfonia n. 3 di Brahms e la Sonatina in

## Listato 1 - Il programma di sintesi musicale.

```

10 REM JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
20 REM J J
30 REM J MUSICA CON UN M.S.X. J
40 REM J J
50 REM J di Paolo Ferrami J
60 REM J J
70 REM J 1985 J
80 REM J J
90 REM JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
100 REM
110 REM
120 GOSUB 1660:REM -> Al titolo
130 REM
140 REM
150 REM MENU
160 REM
170 SCREEN0:COLOR15,13,13
180 KEY OFF
190 LOCATE0,0:PRINT" _____"
200 LOCATE1,2:PRINT"1] J.S.BACH: BOURRE'
in MI minore"
210 LOCATE1,4:PRINT"2] L.V.BEETHOVEN: RO
MANZA op.50"
220 LOCATE1,6:PRINT"3] A.VIVALDI: INVERN
O (le 4 stagioni)"
230 LOCATE1,8:PRINT"4] J.BRAHMS: ALLEGRE
TTO (Sinf. n.3)"
240 LOCATE1,10:PRINT"5] A.DIABELLI: SONA

```

```

TINA in RE maggiore
250 CLEAR:LOCATE1,18:PRINT" ":INPUT "n°
motivo:";A
260 ON A GOSUB 310,630,840,1200,1490
270 GOTO250
280 REM
290 REM BACH:BOURRE' in mi minore
300 REM
310 FORN=1T02
320 PLAY"V7T15006L8EF+","V7T15004L80E","
V7T150"
330 PLAY"06L46","04L4E"
340 PLAY"06L8F+E","04L4A"
350 PLAY"06L4D+","04L4B"
360 PLAY"06L8EF+","04L4A"
370 PLAY"05L4B","04L40"
380 PLAY"06L8C+D+","04L4F+"
390 PLAY"06L4E","04L4E"
400 PLAY"06L8DC","04L4G"
410 PLAY"05L4B","04L40"
420 PLAY"05L8AG","04L4A"
430 PLAY"05L4F+","04L4B"
440 PLAY"05L86ABAGF+","04L4AGB"
450 PLAY"05L4E","04L8EF+"
460 PLAY"06L8EF+","04L8GF+"
470 PLAY"06L46","04L4E"
480 PLAY"06L8F+E","04L4A"
490 PLAY"06L4D+","04L4B"
500 PLAY"06L8EF+","04L4A"
510 PLAY"04L4B","04L40"
520 PLAY"05L8C+D+","04L5F+"

```

Seguito programma Musica con un MSX.

530 PLAY"05L4E", "04L4E"  
 540 PLAY"05L8DC", "03L4F+"  
 550 PLAY"04L4B", "03L4G"  
 560 PLAY"04L8AG", "04L4C"  
 570 PLAY"04L8GL4F+L8G", "04L4DD"  
 580 PLAY"0403L1G", "03L1B", "03L1G"  
 590 NEXT:RETURN  
 600 REM  
 610 REM BEETHOVEN: ROMANZA Op.50  
 620 REM  
 630 PLAY"V7T20004L4G.", "V7T20003L4G.", "V7T200"  
 640 PLAY"05L5C05L2C.", "04L1C"  
 650 PLAY"L4EL4DL4FL4D", "03L1F"  
 660 PLAY"05L2CR804L8B05L8D04L8B", "03L1G"  
 670 PLAY"04L26R4L4G", "03L1G"  
 680 PLAY"05L2CL8DL8C+L8DL8D+", "04L2C03L2F"  
 690 PLAY"05L2EL8FL8EL8FL8A", "04L2C03L2F"  
 700 PLAY"05L2ER8L8FL8F+L8G", "04L1C"  
 710 PLAY"05L2D.04L4G", "04L1G"  
 720 PLAY"05L2F.L4D", "04L1D", "03L1G"  
 730 PLAY"04L4B05L4C04L4B05L4C", "03L4604L4EL4DL4E"  
 740 PLAY"05L2F.L4D", "04L1D", "03L1G"  
 750 PLAY"04L4B05L4GL4F+L4G", "03L4604L4EL4D+L4E"  
 760 PLAY"L2A.L8B06L8C", "04L2FL2F+"  
 770 PLAY"05L2C.L8L8EL8D", "04L1G"  
 780 PLAY"05L2DL4G04L4B", "03L1G"  
 790 PLAY"05L1C", "04L1E", "03L1C"  
 800 RETURN  
 810 REM  
 820 REM VIVALDI: INVERNO  
 830 REM  
 840 PLAY"V7T14506L4CL86FL4EL8DC", "V5T14505L4CCCC", "V5T14504L4666G"  
 850 PLAY"06L4D05L4GR4L4G", "04L4BBBB", "04L4666G"  
 860 PLAY"06L8FEDC05L4B06L4F", "04L4BBBB", "04L4666G"  
 870 PLAY"06L4FER4L4E", "05L4CCCC", "04L466G"  
 880 PLAY"06L4DL8EFL46L8AB", "04L4BBBB", "04L4666G"  
 890 PLAY"06L4CL8L8DEL4FL8GA", "05L4CCCC", "04L4AAAA"  
 900 PLAY"05L4B06L8CDL4EL8FG", "04L4BBBB", "04L4666G"  
 910 PLAY"05L4AL8B06L8CL4DL8EC", "04L4AAAA", "04L4FFF+F"  
 920 PLAY"05L2BL8BGf+G", "04L4BBBB", "04L466G"  
 930 PLAY"06L2DL8D05GF+G", "05L4CCCC", "04L4AAAA"  
 940 PLAY"06L2EL8ED05L8GF+G", "05L4CCCC", "04L4666G"  
 950 PLAY"06L2F+L8F+DCD", "05L4CCCC", "04L4AAAA"

960 PLAY"06L4605L4GR406L4G", "04L4BBBB", "04L4666G"  
 970 PLAY"06L86F+EDC05L8BAG", "05L4CCCC", "04L4AAAA"  
 980 PLAY"05L2A.L4G", "05L4DDDD", "04L4F+F+F+F+"  
 990 PLAY"05L26R2", "04L4BBBB", "04L4666G"  
 1000 PLAY"05L4G06L8DC05L4BL8AG", "04L4BBBB", "04L4666G"  
 1010 PLAY"05L4ADR4D", "04L4AAAA", "04L4F+F+F+F+"  
 1020 PLAY"06L8C05L8BAGL4F+06L4C", "05L4DDDD", "04L4F+F+F+F+"  
 1030 PLAY"06L4C05L4BR4L4G", "05L4DDDD", "04L4666G"  
 1040 PLAY"06L8FEDC05L4B06L4F", "05L4BBBB", "04L4666G"  
 1050 PLAY"06L4FER4E", "05L4CCCC", "04L466G"  
 1060 PLAY"05L4AL8B06L8CL4DL8EF", "05L4CC04L4AA", "04L4FFFF"  
 1070 PLAY"05L4B06L8CDL4EL8FG", "04L4BBBB", "04L4666G"  
 1080 PLAY"06L4CL8DEL4FL8GA", "05L4CCCC", "04L4666G"  
 1090 PLAY"05L2BR4L8B06L8C", "05L4DDDD", "04L4666G"  
 1100 PLAY"L806D05L8BAG06L8EFGE", "04L4BB05L4CC", "04L4666G"  
 1110 PLAY"06L4D05L4GR4L8B06L8C", "04L4BBBB", "04L4666G"  
 1120 PLAY"L806D05L8BAG06L8EFGE", "04L4BB05L4CC", "04L4666G"  
 1130 PLAY"06L4D05L4GR406L8D6", "04L4BBBB", "04L4666G"  
 1140 PLAY"06L4EL8DC05L4B.06L8C", "05L4CCDD", "04L4666G"  
 1150 PLAY"L106C.", "05L1E.", "05L1C."  
 1160 RETURN  
 1170 REM  
 1180 REM BRAHMS: ALLEGRETTO  
 1190 REM  
 1200 PLAY"T15005L4C.L16D", "T150", "T150"  
 1210 PLAY"05L2D+", "04L8CD+605L8C04L86D+ "  
 1220 PLAY"05L8G.L16FL4DL8C.L16D", "04L4FL4G+R4", "05L2C"  
 1230 PLAY"05L2D+", "04L8CD+605L8C04L86D+ "  
 1240 PLAY"05L8A+L16G+L4DL8D.L16D+", "04L4FG+R4", "05L2CR4"  
 1250 PLAY"05L2FL8G+.L16G", "03L8B04L8DFAR4", "R205L4D"  
 1260 PLAY"05L2D+L8G.L16F", "04L8CD+605C"  
 1270 PLAY"05L4CL8CL16DC04L8B05L8C", "03L8G+04DL4FF+ "  
 1280 PLAY"05L4DD+E", "04L4GC03L4A+", "04L4B05L4CC"  
 1290 PLAY"05L4FA+.L8G+", "03L4A+04L4FG+", "05L4CDC"  
 1300 PLAY"05L4DG.L16F", "04L4A+D+F"

## Seguito programma Musica con un MSX.

```

1310 PLAY"05L16C0C04L16B05L16CL4D+D", "04
L4G+L46F+"
1320 PLAY"05L2G06L8C.L16D", "04L4F+D+D", "
05R4R4L4G"
1330 PLAY"06L2D+", "04L8CD+G05L8C04L86D+"
, "05L2G."
1340 PLAY"06L8G.L16FL4DL8C.L16D", "04L4G
L4G+R4", "06L2C05L4G+"
1350 PLAY"06L2D+", "04L8CD+G05L8C04L86D+"
, "05L2G."
1360 PLAY"06L8A+.L16G+L4DL8D.L16D+.", "04
L4FG+R4", "06L2C"
1370 PLAY"06L2FL8G+.L16G", "03L8B04L8DFAR
4", "R206L4D"
1380 PLAY"06L2D+L8G.L16F", "04L8CD+G05C",
"r2o6l4d"
1390 PLAY"06L4CL8CL16D05L8B06L8C", "03LA
G+04DL4FF+", "r2o6l4c"
1400 PLAY"06L4DD+E", "05L4B06L4CC", "05L4G
6G"
1410 PLAY"06L4FA+.L8G+", "06L4CDC", "05L4G
+04L4FG+"
1420 PLAY"06L4DG.L8F", "05L2A+L4G+", "04L4
A+D+F"
1430 PLAY"06L16C0C05L16B06L16CL4D+D", "05
L4CC04L4B", "04L4G+6G"
1440 PLAY"06L2CL2C", "05L4FEL2E", "04L8C06
5L4CL2C"
1450 RETURN
1460 REM
1470 REM DIABELLI: SONATINA
1480 REM
1490 PLAY"T15006L2DL4F+D", "T15004L8DAF+A
DAF+A"
1500 PLAY"05L4A.L8BL4AR4", "0418DAF+ADAF+
A"
1510 PLAY"06L2EL4GE", "04L8C+AGAC+AGA"
1520 PLAY"05L4A.L8BL4AR4", "04L8C+AGAC+AG
A"
1530 PLAY"06L2DL8F+EDC+", "04L8DAF+ADAF+A

```

```

"
1540 PLAY"05L8B06L8C+05L8B06L8C+05L8B06L
8GF+E", "04L2GE"
1550 PLAY"06L8DF+ED05L8C+06L8ED06L8C+", "
0412A03L4A"
1560 PLAY"06L2DR2", "04L8DGf+6L4DR4"
1570 PLAY"T15006L2DL4F+D", "T15004L8DAF+A
DAF+A"
1580 PLAY"05L4A.L8BL4AR4", "0418DAF+ADAF+
A"
1590 PLAY"06L2EL4GE", "04L8C+AGAC+AGA"
1600 PLAY"05L4A.L8BL4AR4", "04L8C+AGAC+AG
A"
1610 PLAY"06L2DL8F+EDC+", "04L8DAF+ADAF+A
"
1620 PLAY"05L8B06L8C+05L8B06L8C+05L8B06L
8GF+E", "04L2GE"
1630 PLAY"06L8DF+ED05L8C+06L8ED06L8C+", "
0412A03L4A"
1640 PLAY"06L2DR2", "04L8DGf+6L4DR4"
1650 RETURN
1660 REM
1670 REM titolo
1680 REM
1690 OPEN"GRP:"AS1
1700 SCREEN2:COLOR8,1,1:CLS
1710 CIRCLE(125,85),80,2,,.8:BEEP
1720 CIRCLE(125,85),100,2,,.8:BEEP
1730 PAINT(32,85),2:BEEP
1740 PSET(70,50),1:PRINT#1,"JJJJJJJJJJJJJJ
JJ":BEEP:COLOR11
1750 PSET(70,63),1:PRINT#1,"MUSICA CON
UN":BEEP:COLOR4
1760 PSET(73,84),1:PRINT#1," M.S.X.
":BEEP:COLOR15
1770 PLAY"t150o5l4or6cr6l2c", "t150o5l4er
6fr6l2e", "t150o5l4gr6ar6l2g"
1780 PLAY"o4l1c", "o4l1e", "o4l1g"
1790 PSET(70,104),1:PRINT#1," di P.FERRA
MI "
1800 FORA=1T03900:NEXT:RETURN

```

Re maggiore di Diabelli.

## Il programma

Dato per scontato che abbiate letto almeno una volta il manuale di istruzioni allegato al vostro computer, per poter programmare facilmente dei brani musicali con un MSX è necessario saper leggere le note da un pentagramma: questo non significa assolutamente che dobbiate saper suonare uno strumento, ma è sufficiente che riusciate a riconoscere il nome e il valore di ogni nota senza preoccuparvi se questo avviene con una certa lentezza. Usando un

qualsiasi metodo in vendita nei negozi di dischi e strumenti musicali, dopo un paio di ore di esercizio avrete raggiunto l'abilità necessaria per programmare il sintetizzatore musicale di un MSX, almeno per quanto riguarda l'istruzione Play.

Un altro problema che si incontra è che l'istruzione Play legge le note in notazione anglosassone: problema che riguarda soprattutto l'ottava di riferimento su cui si sta lavorando. Al nostro Do corrisponde la lettera C, al Re la D e così via fino al Sol, a cui corrisponde la G, poi con il La si passa alla S, al Si corrisponde la B e, a questo punto, tra la B e

la C, passiamo all'ottava superiore.

Esempio:

*alla scala:*

(ottava 1) Do Re Mi Fa Sol La Si

(ottava 2) Do Re...

*corrisponde:*

(ottava 1) C D E F G A B

(ottava 2) C D...

Nulla cambia invece per quanto riguarda il valore delle note.

Vi ricordo infine che l'ottava viene cambiata usando la lettera O seguita immediatamente dal numero corrispondente all'ottava desiderata, per gli intervalli si usa la lettera R seguita dal valore del tempo, per il volume usiamo V (n) dove

n è un numero compreso tra 0 e 7 e, infine, ci sono i comandi S e M che cambiano la forma e l'involuppo dell'onda e, di conseguenza, il timbro. Per la velocità di esecuzione usiamo all'inizio del brano la lettera T (n) con n compreso tra 32 e 255.

Due sono i modi di utilizzare l'istruzione Play: il primo è quello di inserire tutte le istruzioni in una variabile stringa, da far leggere poi a un'unica istruzione Play, l'altro consiste nel ripetere l'istruzione, contenente ciascuna poche note relative alle tre voci, per esempio le note (e le pause) di una musica.

Per questo programma ho adottato il secondo metodo, inserendo nella prima

relativi alle due mani). L'unica limitazione è quella che non devono esserci più di tre note suonate contemporaneamente, poichè ciascuna nota va assegnata a una voce. Se un brano richiede degli accordi a tre note di accompagnamento, è possibile rimediare sostituendo a quest'ultimi degli accordi a duine. Vi consiglio infine di non alterare i valori della forma d'onda e dell'involuppo quando trascrivete per le prime volte un brano dallo spartito al computer.

Tornando ai brani contenuti nel programma, vediamo come possiamo "giocare" sulle istruzioni per modificarne l'esecuzione. Un primo esperimento è quello di inserire all'inizio di un brano,

guire la Sonatina del Diabelli a un coro di grilli. Se odiate i grilli inserite S2M3000, allora la melodia sarà eseguita da una spinetta (strumento adattissimo al brano in questione); ricordo che S2 è una forma d'onda triangolare-retta. Un ultimo effetto che vi consiglio di provare sempre sulla linea 1490, è l'onda a "dente di sega": digitate questa linea:

```
1490 PLAY
"S8M1900T200O6L2DL4F+D",
"S8M1900T200O4L8DAF+A"
```

poi date il Run e ascoltate l'effetto. Per fare in modo che l'Inverno di Vivaldi sia suonato con un'ocarina da un gruppetto di soldati in marcia (sic!) riscrivete la linea 840 in questo modo:

```
840 PLAY "TI4506L4CL8GFL4EL8DC",
"S5M30T1450514CCCC",
"S5M50T14504L4GGGG"
```

Sostituendo alla linea così modificata il valore 255 in luogo del 145 dopo le lettere T, i soldati si metteranno a correre. Poichè non c'è limite al cattivo gusto, ecco come fare per trasformare una Bourrée di Bach in un "Tip-Tap":

```
320 PLAY "T250O6L8EF+",
"S1M20T250O4L8GE",
"S1M20T250"
```

Ora provate la stessa forma d'onda, ma con un periodo di involuppo pari a 900 (M900): la differenza è notevole.

Vedete che, una volta memorizzato un brano è facile e divertente cambiare i parametri per ottenere effetti particolari o strumenti diversi senza dover ricorrere all'istruzione Sound, il cui utilizzo è estremamente più complesso e meno immediato.

Sul programma non c'è molto da dire essendo formato semplicemente da un menu che tramite l'istruzione On-Gosub manda al sottoprogramma contenente il brano scelto. I REMarks contenuti nel listato vi aiuteranno a trovare i vari brani. Vorrei soffermarmi sulla routine Titolo (nel quale troviamo il testo scritto nella pagina grafica Screen 2). Per ottenere questo risultato basta procedere all'apertura di un file con l'istruzione:

```
OPEN "GRP:" AS1
```

e stampare la stringa mediante l'istruzione Print # 1, "[testo]"; ovviamente al posto dell'istruzione Locate useremo Pset.

PLAY "o5L4Do4L8GABo5C", "o3L2GL4A", "o3L2B"

Figura 1 - Ecco come si rappresenta una parte di spartito con l'istruzione Play.

istruzione Play il tempo T, cioè il numero delle semiminime eseguite in un minuto, e, quando era il caso, il volume da assegnare a ciascuna delle tre voci; poi, per ogni istruzione seguente l'ottava, il nome e il valore della nota corrispondente a una misura (figura 1). Questo metodo allunga un po' il listato, ma in compenso permette di poter controllare in ogni momento la situazione del programma e confrontarlo con lo spartito, rendendo immediato il ritrovamento di una eventuale "stecca". Così facendo, inoltre, è possibile programmare brani presi da spartiti per strumenti a tastiera (tanto per intenderci, quelli con i pentagrammi in Chiave di Sol e in Chiave di Violino

alla prima stringa del Play, la lettera V0 a una o due voci, con il risultato di riuscire a togliere il volume e ascoltare solo la melodia o l'accompagnamento (questo è anche un ottimo metodo per ricercare errori che fanno stonare il brano). Provate ad ascoltare la Sonatina del Diabelli poi, dato il Break, editare la linea 1490 e inserire V0 subito dopo gli apici della prima stringa: il risultato sarà quello di far eseguire al computer solo l'accompagnamento.

Altro esperimento è quello di inserire, sempre alla linea 1490, al posto di V0 le lettere S14M100, cioè una forma d'onda triangolare con periodo di involuppo 100: il risultato sarà quello di fare ese-

# Tasti funzione per Spectrum 48 Kbyte

## Come facilitare l'input dei programmi

di Massimo Modelli

Una delle particolarità dello Spectrum che da principio lascia perplesso, ma che si rivela poi molto comoda, è la disponibilità di tutte le parole chiave del BASIC con la pressione di un paio di tasti, cosa che permette di introdurre programmi in BASIC molto rapidamente. Tutto il funzionamento della tastiera per ottenere le parole chiave, i caratteri grafici, i vari simboli, ecc. è basato sull'uso dei due tasti di Shift per modificare lo stato del cursore (L, K, E, G); quasi tutte le combinazioni dei due tasti di Shift con i diversi stati del cursore producono risultati diversi, moltiplicando virtualmente il numero delle tastiere disponibili.

Vi è però un caso in cui l'azione dei due tasti di Shift è la stessa per tutto un gruppo di tasti; se infatti si porta il cursore nello stato "E", l'azione del Caps Shift e del Symbol Shift sui tasti delle prime tre file è la stessa e riporta la parola o il simbolo scritto in rosso sotto il tasto stesso.

Vi sono quindi 28 tasti (escludendo i due Shift) ai quali si potrebbe utilmente assegnare il compito di fornire dei testi particolari, realizzando così un'utile opzione assente sullo Spectrum: i tasti fun-

zione.

Il programma che segue permette appunto di realizzare questa opzione generando un codice macchina del tutto autonomo e integrato nel funzionamento del sistema operativo.

Le comodità offerte da questa opzione sono varie: è possibile ad esempio memorizzare le chiamate a particolari

routine (come la renumber) nella forma Randomize Usr XXX per averle disponibili con la pressione di un tasto, ma forse l'uso più comodo, grazie alla sua compatibilità con l'Interfaccia 1, consiste nel poter ottenere velocemente i comandi diretti ai microdrive, ad esempio ottenere con un tasto le nove battute necessarie per "Save★" m"; "1;""".

Listato 1 - Il listato Assembly.

```

00010
00020 ;TASTIERA PROGRAMMABILE
00030 ;*****
00040 ORG 0FAFFH
00050 DEFU START ;indirizzo
00060 ;della routine
00070
00080 ;Sottoprogrammi di attivazione e
00090 ;spegnimento
00100
00110 ST1 LD A,0FAH ;rout on
00120 LD I,A
00130 IM 2
00140 RET
00150
00160 ST2 IM 1 ;rout off
00170 RET
00180
00190 ;aggiornamento del contatempo e
00200 ; test sulla ROM
00210 START PUSH AF
00220 PUSH HL
00230 PUSH BC
00240 PUSH DE
00250 TEST LD A,243 ;test ROM
00260 LD HL,0
00270 CP (HL)
00280 JR Z,CONT
00290 POP DE
00300 POP BC

```

```

00310 POP HL
00320 POP AF
00330 EI
00340 RET
00350 ;AGGIORNA OROLOGIO
00360 CONT LD HL,(23672);FRAMES
00370 INC HL
00380 LD (23672),HL
00390 LD A,H
00400 OR L
00410 JR NZ,KEYINT
00420 INC (IY+64) ;FRAMES3
00430 ;INIZIO operazioni tastiera
00440
00450 ;Verifica stato del programma
00460 KEYINT LD HL,40H
00470 PUSH HL ;PRE LOAD
00480 ; RETURN ADRES
00490 LD A,(RIT)
00500 AND A
00510 JR Z,TP1
00520 DEC A
00530 LD (RIT),A
00540 RET
00550 TP1 LD A,(COUNT)
00560 AND A
00570 JR Z,LETKEY
00580
00590 ;Routine di trasferimento
00600 OUT LD HL,23611
00610 BIT 5,(HL) ;FLAGS(5)
00620 RET NZ
00630 SET 5,(HL) ;FLAGS(5)

```

## Il programma

Il programma BASIC (listato 2) di cui è fornito il listato svolge due compiti: nella sua prima sezione si trovano le routine che permettono di editare, con possibilità di modifica, i testi che costituiscono le funzioni; la seconda parte del programma produce invece la routine in linguaggio macchina che realizza i tasti funzione, preparandola per lavorare all'indirizzo di memoria fornito dall'utente.

Tutte le fasi del lavoro sono dirette dal computer in maniera interattiva, in modo da non necessitare di un particolare apprendimento per poter usare il programma. Vediamo comunque di descrivere, anche con esempi, le fasi principali del lavoro.

Dando il Run al programma viene visualizzato sullo schermo un fac-simile della tastiera dello Spectrum, con un cursore lampeggiante sul tasto corrispondente alla cifra 1; per mezzo dei tasti con le frecce è possibile spostare il cursore sul tasto a cui si desidera assegnare una particolare funzione, e con la pressione del tasto Space si può passare a definirlo. Supponiamo di voler assegnare al

tasto J la funzione:

LOAD "m"; 1;"

Poiché non è possibile fornire in un'input la parola chiave Load si dovrà utilizzare la funzione BASIC Chr\$ con argomento 239, che è il codice numerico del "carattere" Load (provate a digitare come comando diretto Print Chr\$ 239: sullo schermo compare appunto Load). Per fare ciò dovrete spostare il cursore prima delle virgolette, battere Chr\$ 239 + e riportare il cursore all'interno delle virgolette. Successivamente digitate il resto dell'istruzione, facendo attenzione, ogni volta che dovrete inserire le virgolette, a batterne due per far capire al BASIC che non intendete finire la stringa, ma inserire delle virgolette. La linea che inserirete sarà quindi la seguente:

CHR\$ 239 + """"m""";1;"

Rispondendo negativamente alla domanda "Premi F per finire", ritornerete alla fase iniziale: portate di nuovo il cursore sul tasto J e premete lo Space, dovrebbe apparire sullo schermo la frase:

LOAD "m"; 1;"

che vi mostra la funzione che avete associato a quel tasto; se vorrete modificarla, premete il tasto m, altrimenti il solo Enter. Se ridigiterete la istruzione precedentemente vista, aggiungendo in fondo dopo tutte le virgolette i seguenti caratteri:

+ CHR\$ 8

andandola a controllare la frase sembrerà la stessa, ma al momento dell'uso avrete una sorpresa, in quanto il cursore non si posizionerà dopo le virgolette, ma si inserirà tra di esse, dove dovrete solo digitare il nome del programma; il carattere Chr\$ 8 corrisponde infatti al comando "cursore a sinistra".

La corrispondenza tra i caratteri e i codici numerici da associare alla funzione Chr\$ si trova nella tabella in Appendice A del manuale italiano. Al momento dell'input il programma fornisce comunque i codici delle parole chiave più importanti che non si possono ottenere quando si è in input.

Una volta terminata la definizione delle funzioni si può passare alla seconda fase del lavoro e cioè alla preparazione e memorizzazione della routine in linguaggio macchina. Il programma vi-

### Seguito listato Assembly.

```

00640      LD  HL, (IND)
00650      LD  A, (HL)
00660      LD  (IY-50), A ;LAST-K
00670      INC  HL
00680      LD  (IND), HL
00690      LD  A, (COUNT)
00700      DEC  A
00710      LD  (COUNT), A
00720      RET
00730 ;LETTURA TASTIERA
00740 LETKEY CALL 028EH ;KEY-SCAN
00750      RET  NZ
00760      LD  A, (IY+7) ;MODE
00770      CP  1
00780      JR  NZ, TER
00790      LD  A, D
00800      CP  27H
00810      JR  NZ, TER
00820      LD  A, E
00830      CP  0FFH
00840      JR  Z, TER
00850      AND  07H
00860      CP  3H
00870      JR  Z, TER
00880      CP  4H
00890      JR  Z, TER
00900      LD  A, E ;ESISTE
00910      LD  B, 0 ;LA
00920      LD  HL, TAB ;STRINGA?
00930      LOOP LD  C, (HL)

```

```

00940      INC  HL
00950      AND  A
00960      JR  Z, FINE
00970      DEC  A
00980      ADD  HL, BC
00990      JR  LOOP
01000      CP  C
01010      JR  Z, TER
01020      LD  A, 1
01030      LD  (NOREP), A
01040      LD  A, C
01050      LD  (COUNT), A
01060      LD  (IND), HL
01070      JR  OUT
01080      LD  A, (NOREP)
01090      AND  A
01100      JR  Z, BOARD
01110      XOR  A
01120      LD  (NOREP), A
01130      LD  A, 6
01140      LD  (RIT), A
01150      RET
01160      BOARD XOR  A
01170      JP  2C3H
01180
01190 ;Sezione dati
01200 RIT  DEFB 0 ;RITARDO
01210 COUNT DEFB 0 ;CARATTERI
01220 NOREP DEFB 0 ;FLAG RIP
01230 IND  DEFW 0
01240 TAB  END ;inizio dei dati
01250

```

## ZX SPECTRUM



## Libri firmati JACKSON



Nicole Bréaud-Pouliquen

**LA PRATICA DELL'APPLE**

"Il Sistema APPLE II", il "BASIC Applesoft" il disegno e la grafica: arricchiti da esempi e esercizi. 130 pagine L. 10.000  
Codice 341D

F. Franceschini - F. Paterlini

**Voi e il vostro Commodore 64**

Uno strumento fondamentale per la comprensione e programmazione del Commodore 64. Con consigli, programmi testati, glossario e utili accenni di BASIC. 256 pagine L. 22.000 Codice 347D

Alan Miller

**PROGRAMMI SCIENTIFICI IN PASCAL**

Un'opera base per chi desidera costruirsi una "libreria" di programmi in grado di risolvere i più frequenti problemi scientifici e ingegneristici. 372 pagine L. 25.000  
Codice 554P

Carmine Elefante

**L'home computer TI/99-4A**

Il BASIC, il BASIC Esteso e il microprocessore dell'home computer della T.I. Con programmi di utilità e svago. 192 pagine L. 15.000  
Codice 343B

Giacomino Baisini - Giò Federico Baglioni

**IL FORTH PER VIC 20 E CBM 64**

La programmazione in FORTH e la sua implementazione sul Commodore VIC 20 e CBM 64. 150 pagine L. 11.000  
Codice 527B

Franco Filippazzi - Giulio Occhini

**VOI E L'INFORMATICA**

L'opera che il manager moderno non può ignorare. In 100 tavole: gli strumenti dell'Informatica, l'Informatica e l'Azienda, realtà e prospettive tecnologiche... 116 pagine L. 15.000  
Codice 526A

Roland Dubois

**CAPIRE I MICROPROCESSORI**

Un fantastico viaggio alla scoperta del "cervello" elettronico: la funzione del microprocessore, delle memorie ROM e RAM, delle interfacce... 126 pagine L. 10.000  
Codice 342A

Gaetano Marano

**77 PROGRAMMI PER SPECTRUM**

Dalla Grafica alla Business Grafica, dalla musica alle animazioni, dai giochi all'elettronica... tutte le possibilità offerte dal più piccolo dei computer. 150 pagine a colori L. 16.000  
Codice 555A

Rita Bonelli-Daria Gianni

**ALLA SCOPERTA DEL VIC 20**

Un testo chiave per imparare a conoscere e usare uno dei Personal del momento. 308 pagine L. 22.000  
Codice 338D  
Cassetta Programmi L. 15.000  
Floppy Programmi L. 25.000

## La Biblioteca che fa testo

In busta chiusa, e senza impegno, inviate questo coupon a:

Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano

Desidero ricevere gratuitamente il Catalogo Generale della Biblioteca Jackson e informazioni sulle 10 Riviste specialistiche da voi pubblicate. (allego L. 1.000 in francobolli per contributo spese di spedizione)

Desidero ricevere contrassegno il/i volume/i più L. 3.000 per contributo spese di spedizione

(pagherò al ricevimento L. ....)

Nome ..... Cognome .....

Via .....

CAP ..... Città .....

sualizza la lunghezza complessiva della routine (dati compresi) e si pone in attesa che l'utente determini l'indirizzo di memoria per il quale la routine dovrà essere rilocata; quello che viene mostrato (a fianco della scritta Inizio) è il più alto disponibile e l'utente può modificarlo utilizzando i tasti 6 e 7, che lo diminuiscono o lo aumentano a passi di 256 byte (non è possibile una regolazione dell'origine più fine per le caratteristiche della routine).

Una volta determinata l'origine della routine il programma fornisce gli indirizzi delle due entry point utili: Start e Stop, rispettivamente per attivarla e disattivarla. L'ultima operazione da svolgere è chiaramente la memorizzazione su nastro o cartuccia della routine così preparata, e per fare ciò il programma deve passarla in memoria, ponendola a partire da un indirizzo arbitrario scelto dall'utente; se si utilizza l'indirizzo di origine, la routine può subito funzionare e si possono saltare per il momento le operazioni di salvataggio su nastro. Un consiglio: prima di provare la routine è bene salvare il programma BASIC appena digitato, per evitare di perderlo se, in seguito ad un errore di battitura dei codici contenuti nelle linee Data, il sistema si bloccasse.

### La routine

Supponiamo di aver preparato la routine con tutte le funzioni che desideravamo e di averla salvata su cassetta o cartuccia; a questo punto il programma BASIC non ci serve più e, dopo averlo eventualmente salvato per poter in seguito definire nuove funzioni, lo possiamo cancellare. A questo punto ogni volta che lo desidereremo potremo disporre dei tasti funzione definiti seguendo questo procedimento:

- modificare il valore di RAMtop con il comando Clear in modo da fare spazio alla routine;
- caricare la routine, ricordando di specificare l'indirizzo di origine, se essa è stata inizialmente salvata da un indirizzo diverso da origine;
- attivarla con il comando Randomize Usr Start (gli indirizzi Origine, Start e Stop sono forniti dal programma BASIC, ed è sempre bene annotarsi, in quanto sono gli unici dati che servono per controllare la routine).

Da ora in avanti qualunque lavoro si svolga al computer questa routine farà parte integrante del sistema operativo, più precisamente delle routine di interrogazione della tastiera. Ogni qualvolta si premano contemporaneamente e in modo esteso (cursore "E") un tasto e il Caps Shift verrà resa, se esiste, la corrispondente funzione: se nessuna funzione è definita per quel tasto, verrà fornita la parola chiave o il simbolo scritto sotto al tasto.

E' importante ricordarsi di disattivare la routine con il comando Randomize Usr Stop prima di riscrivere o caricare qualcosa nella zona di memoria occupata

da questa: infatti, il programma residente in memoria in quella zona viene eseguito automaticamente (50 volte al secondo) e se non vi risiede un programma coerente, la cosa più facile che succeda è che la macchina si blocchi e necessiti quindi di un Reset.

Come probabilmente avrete capito, il funzionamento della routine è basato sul "dirottamento del controllo dopo un interrupt"; in altre parole, in seguito ad un segnale di interrupt generato dalla ULA dello Spectrum il controllo non passa, come avviene normalmente, alla routine di lettura della tastiera, ma alla nostra routine dei tasti funzione; questa

utilizza varie parti della routine per leggere la tastiera e al termine le ripassa il controllo. Poiché le routine della ROM "fantasma" sono diverse, sia come contenuti che come indirizzi, la routine esegue un test preliminare sulla prima locazione di memoria per accertarsi di essere eseguita con la ROM giusta.

Un consiglio: sullo Spectrum il comando New è particolarmente drastico, in quanto, oltre a cancellare i programmi e le variabili, reinizializza le variabili di sistema e, cosa per noi rilevante, reinizializza pure il valore del registro I e riporta la CPU nel modo di lavoro 1 (IM 1). Ai nostri fini è come se si fosse eseguito il

#### Listato 2 - Il programma BASIC.

```

1 REM
*****
T A S T I   F U N Z I O N E
*****
4 REM La FN x converte le coordinate della tastiera (riga e colonna) in coordinate dello schermo
5 DEF FN x(x,y)=(x-1)*3+y+(y=2 OR y=3)-(y=4)
10 PAPER 1: INK 7: BORDER 1: CLS
14 REM valore della Ramtop
15 LET x=60000: INPUT "RAMTOP a 60000 ? (y,n)",a$: IF a$="y" OR a$="Y" THEN GO TO 18
16 IF a$<>"n" AND a$<>"N" THEN BEEP .5,5: GO TO 15
17 INPUT "Valore Ramtop ",x
18 CLEAR x
90 GO TO 1000
99 REM *****
100 FOR i=0 TO 8: PRINT AT i,0: PAPER 0,,: NEXT i: PRINT AT 1,0: LET x=0: LET y=168
110 FOR i=1 TO 40: PRINT PAPER 0;"1234567890QWERTYUIOPASDFGHJKL ecZXCVBNMsb"(i): PAPER 4;" ": PRINT PAPER 0;" ": PLOT x,y: DRAW 16,0: DRAW 0,-9: DRAW -16,0: DRAW 0,9: LET x=x+24
120 IF i=10 THEN PRINT PAPER 0'TAB 2: LET x=16: LET y=152
130 IF i=20 THEN PRINT PAPER 0'TAB 3: LET x=23: LET y=136
140 IF i=30 THEN PRINT PAPER 0'TAB 2: LET x=16: LET y=120
150 NEXT i: PRINT AT 9,0: INVERSE 1: INK 5;" T A S T I   F U N Z I O N E ": RETURN
199 REM *****
200 PAPER 8: OVER 1: PRINT #0: AT 0,0;"Posizionati con le frecce e premi SPACE"
210 LET x=1: LET y=x: LET x1=x: LET y1=x: PRINT INK 9;AT x,x;"*
220 FOR i=0 TO 0 STEP 0: PRINT

```

```

AT y*2-1, FN x(x,y);"*"; AT y1*2-1, FN x(x1,y1): INK 9;"*": LET x=x1: LET y=y1
230 IF INKEY$="" THEN OVER 0: GO TO 260
240 LET y1=y+(INKEY$="6" AND y<4)-(INKEY$="7" AND y>1): LET x1=x+(INKEY$="8" AND x<10)-(INKEY$="5" AND x>1): NEXT i
250 REM valore della mezzafila [caps s.-U]=7; [A-G]=6;...; [1-5]=4;...; [space-B]=0
260 LET mf=7-((y+3)*(x>5)+(-y+4)*(x<6))
265 REM valori delle colonne; i tasti piu' esterni delle mezzefile (caps s.,A,0,1,0,P...) valgono 4; i piu' interni (U,G,T...) valgono 0
270 LET col=4-((x-1)*(x<6)+(10-x)*(x>5))
272 REM valori dei tasti; B=1,H=2...A=39,caps shift=40
275 LET key=col*8+mf+1
290 RETURN
1000 REM *****PROGRAMM
1010 LET b$="": FOR i=1 TO 40: LET b$=b$+CHR$(i): NEXT i: GO SUB 100
1020 GO SUB 200: IF mf=3 OR mf=4 OR key=25 OR key=40 THEN PRINT AT 10,0;"TASTO NON RIDEFINIBILE": BEEP .2,30: LET pep=4: GO TO 1100
1025 LET c=1: FOR i=1 TO key-1: LET c=c+1+CODE b$(c): NEXT i
1030 IF NOT CODE b$(c) THEN GO TO 1070
1035 PRINT AT 10,0;,,,,,
1050 PRINT AT 10,0;b$(c+1 TO c+CODE b$(c));AT 0,0: INPUT "Premi m per modificare ";a$: IF a$="m" OR a$="M" THEN LET b$=b$(TO c)+b$(c+1+CODE b$(c) TO ): GO TO 1070
1060 LET pep=2: GO TO 1100
1070 REM *****
1080 INPUT AT 0,0;"PLOT 246 DRAW 252 RAND 249PRINT 245 SAVE 248 LOAD 239ENTER 13 CLS 251 BORDER 231Funzione corr

```



## ZX SPECTRUM

sottoprogramma di Stop della routine: pertanto dopo il New sarà necessario riattivare la routine al solito modo. E' possibile evitare ciò utilizzando al posto del New il seguente comando: Randomize Usr 4660. In questo modo del comando New viene eseguita solo la seconda parte e cioè la cancellazione del programma e delle variabili BASIC: non c'è bisogno di dire che questo comando può comodamente venire definito con i tasti funzione sotto la lettera N (o A, ma attenti a non sbagliare).

Nel listato 1 è riportato il programma Assembly dei tasti funzione, per permettere a chi lo volesse di operare delle

modifiche, ad esempio unire la routine ad altre operanti automaticamente, per mezzo del "dirottamento del controllo dopo un interrupt". Grosso modo le sezioni della routine sono 6:

- ST1 e ST2: sottoprogrammi di attivazione e di spegnimento (Start e Stop);
- aggiornamento del contatempo e test sulla ROM;
- verifica dello stato del programma; può essere in attesa di ricevere il comando per l'emissione di una funzione (lettura della tastiera) o in fase di trasferimento dei caratteri del testo di una funzione;
- Out: routine di servizio per il modo

trasferimento;

- Letkey: routine di servizio per il modo lettura;
- sezione dati: finisce con Tab, che è la copia identica della variabile stringa BS del programma BASIC. Il formato dei dati è il seguente: per ogni tasto il primo valore riporta la lunghezza del testo associato e i seguenti riportano il testo stesso; segue il dato per il tasto successivo, ecc..

La tabella vuota (tutti i testi con lunghezza 0) occupa quindi 40 byte. ■

## Seguito listato BASIC.

```

ispondente =>";a$: IF LEN a$>255
THEN PRINT AT 10,0;"TESTO TROPP
O LUNGO",: BEEP .2,30: GO TO 108
0
1085 LET pep=2: IF LEN a$=0 THEN
LET pep=4
1090 LET b$=b$( TO c)+a$+b$(c+1
TO ): LET b$(c)=CHR$ LEN a$
1100 PRINT AT y*2-1,FN x(x,y); 0
VER 1; PAPER pep;"*": INPUT "pre
mi f per finire ";a$: IF a$="f"
THEN GO TO 2000
1110 GO TO 1020
2000 REM 2000
2010 FOR i=10 TO 21: PRINT AT i,
0,: NEXT i: PRINT AT 11,0;
2020 LET len=180+LEN b$: LET org
=255*256+255
2030 LET org=org-256: IF org+len
>65535 THEN GO TO 2030
2040 PRINT "Lunghezza ";len,"Ini
zio ";org
2050 PRINT #0;"Usa le frecce (6
e 7) per cam- biare l'indirizzo
di inizio, premi 0 per finir
e"
2060 IF INKEY$="0" THEN GO TO 21
00
2070 LET org=org+256*((INKEY$="7
")-(INKEY$="6")): IF org+len>655
35 THEN LET org=org-256: BEEP .3
,20
2080 IF org<32768 THEN LET org=0
rg+256: BEEP .3,20
2090 PRINT AT 11,27;org: PAUSE 8
GO TO 2060
2100 PRINT "Start at ";org+2,"S
top at ";org+9
2110 LET org1=INT (org/256)+1
2120 INPUT "Forniscimi l'indiriz
zo da cui caricare la routine
";ind: PRINT "Indirizzo di caric
amento attuale=>";ind
2121 INPUT "Con quale nome salvo
la routine?";a$: IF LEN a$>10 O
R LEN a$<1 THEN GO TO 2121
2122 PRINT "Per caricare la rout
ine usare; LOAD """;a$;"CODE
";: IF ind<>org THEN PRINT org

```

```

2123 PRINT "Per attivarla e dis
attivarla RANDOMIZE USR Start
o Stop"
2130 FOR i=ind TO ind+179: READ
a: POKE i,a: NEXT i: FOR i=1 TO
LEN b$: POKE i+ind+179,CODE b$(i
): NEXT i
2140 SAVE a$CODE ind,len: PRINT
#0;"Riavvolgi per la verifica":
VERIFY a$CODE ind,len: STOP
3000 DATA 11,org1,62,org1-1,237,
71,237,94,201,237,86,201,245,229
,197,213,62,243,33,0,0,190,40,6,
209,193,225,241,251,201,42,120,9
2,35,34,120,92,124
3010 DATA 181,32,3,253,52,64,33,
77,0,229,58,174,org1,167,40,5,61
,50,174,org1,201,58
3020 DATA 175,org1,167,40,27,33,
59,92,203,110,192,203,238,42,177
,org1,126,253,119,206,35,34
3030 DATA 177,org1,58,175,org1,6
1,50,175,org1,201,205,142,2,192,
253,126,7,254,1,32,52,122
3040 DATA 254,39,32,47,123,254,2
55,40,42,230,7,254,3,40,36,254,4
,40,32,123,6,0
3050 DATA 33,179,org1,78,35,167,
40,4,61,9,24,247,185,40,14,62,1,
50,176,org1,121,50
3060 DATA 175,org1,34,177,org1,2
4,166,58,176,org1,167,40,10,175,
50,176,org1,62,6,50,174,org1
3070 DATA 201,175,195,195,2,0,0,
0,0,0
9000 LET c1=64255: LET c2=63231:
FOR i=0 TO 300: LET n1=PEEK (c1
+i): LET n2=PEEK (c2+i): PRINT i
;TAB 8; INVERSE (n1<>n2);n1;TAB
18;n2: NEXT i
9999 PRINT 65536-USR 7962

```

**N**ei numeri 23 (Dicembre '84) e 24 (Gennaio '85) di **Personal Software** ho presentato un programma che elaborava i conti di un torneo di Bridge a coppie. Sollecitato da un amico che gestisce un simpatico circolo di Bridge a Roma, ho apportato a tale programma alcune modifiche che lo rendono più completo e più utile, soprattutto per quei lettori che organizzano in casa dei piccoli tornei.

In questo numero pubblichiamo per comodità il listato di tutto il programma, ma ci limiteremo a commentare esclusivamente le novità apportate al vecchio listato. Il limite del programma precedente era quello di elaborare la classifica di un singolo torneo a coppie senza però prevedere la possibilità di elaborare una classifica individuale tra tutti i giocatori partecipanti ad una serie di tornei, con l'eventualità di scartare i peggiori risultati. Supponiamo che un gruppo di amici decida di organizzare una serie di 12 tornei (anche piccoli, 4-5 tavoli), di non giocare ogni torneo con lo stesso partner e infine di voler, in qualunque momento, una classifica individuale in cui vengano scartati i tre peggiori risultati di ogni giocatore. Ebbene, mentre per fare i conti di un torneo così piccolo non è certamente necessario il computer, il suo contributo però sarà senza dubbio notevole per attribuire volta per volta un punteggio a tutti i giocatori partecipanti al torneo; scartare i risultati peggiori e fornire la classifica individuale aggiornata dopo ogni turno di gioco.

### Modifiche nella struttura dati

Si sono resi necessari tre ulteriori file di testo in aggiunta a quelli già esistenti (Torneo e Dati). Tali file sono chiamati Anagrafica, Classifica, Turno. Il primo è un file random che contiene i nomi dei giocatori interessati alla classifica finale (max 100). Ogni giocatore è contraddistinto da un codice progressivo a partire da 1. Il secondo file, Classifica, è anch'esso di tipo random e contiene il punteggio che il programma attribuisce ad ogni giocatore in funzione del risultato conseguito nel singolo torneo (max 30 tornei). Il terzo file, Turno, è invece di tipo sequenziale e contiene, a regime, il numero dei tornei giocati.

# Nuovo torneo di Bridge

Rivisitiamo il nostro programma per Apple II

*di Vincenzo Delle Cave*

### Come usare il programma

Una volta digitato il listato 1 e salvato col nome di Bridge-Men-Class e il listato 2 col nome di Bridge-Cont-Class, si può lanciare il programma Bridge-Mem-Class che presenterà il seguente menu:

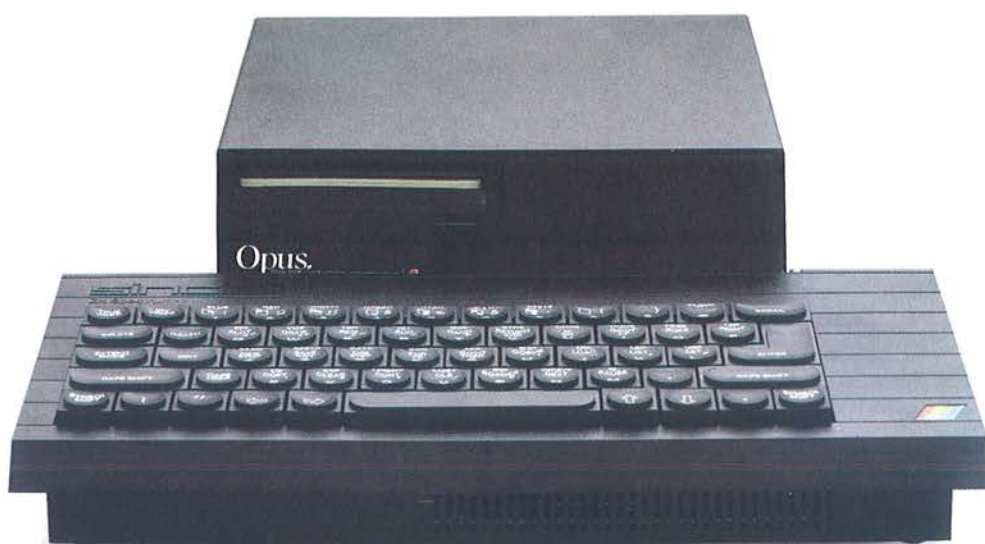
- 2) visualizza mani;
- 3) fine;
- 4) classifica individuale;
- 5) inizializza classifica;
- 6) variazione punteggi.

La prima cosa da fare è digitare l'opzione 5. A questo punto compare un sottomenu del tipo 1) inserimento nomi; 2)





**Vi informiamo  
che il vostro piccolo Spectrum  
è molto più grande.**



**Opus.**  
**DISCOVERY**

**BIT SHOP**  
**primavera**



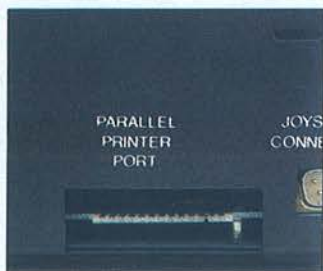
Discovery, è l'unità a dischi più avanzata e più completa presentata fino ad ora per potenziare lo Spectrum; sfrutta i più moderni ritrovati della tecnologia nel campo dei dischetti.

Il modello più economico, Discovery 1, comprende: • Unità a dischetti standard da 3,5 pollici. • Connettore passante per periferiche. • Interfaccia joystick. • Porta per stampante parallela. • Uscita per monitor monocromatico.

Discovery 1 può essere potenziato successivamente aggiungendo un secondo disc drive per mezzo del kit di espansione "Discovery +"; in alternativa, è disponibile il "Discovery 2" già dotato di due drive.

**Opus.**  
**DISCOVERY**

# Questo è l'unico che da solo trasforma un computer in un sistema completo.



### CARATTERISTICHE TECNICHE

**Unità a dischetti:**  
 Tipo del disco . . . . . standard da 3,5in  
 Modello . . . . . 35401  
 Numero delle tracce . . . . . 40  
 Numero delle facce . . . . . 1  
 Capacità totale . . . . . 250 Kbyte  
 Capacità formattata . . . . . 180 Kbyte  
 Accesso traccia-traccia . . . . . 3 ms  
 Alimentazione . . . . . interna

# Discovery: il vostro Spectrum stema.

Discovery è stato concepito in modo da funzionare con gli stessi comandi della "Interface 1" per permettervi l'uso, senza bisogno di modificare nulla, dei numerosi programmi disponibili su cassette o su cartucce per microdrive.

Discovery non occupa "RAM utente" dello Spectrum, per cui potete eventualmente utilizzare programmi su cassetta senza scollegarlo dal computer. Superfluo diventa anche l'alimentatore Sinclair in quanto è la stessa unità (dotata di interruttore generale) ad alimentare lo Spectrum.



Spectrum e Sinclair sono marchi registrati della Sinclair Research Ltd.

**BIT SHOP**  
primavera



#### Interfacce:

- Connettore passante per allacciamento delle periferiche, da 56 poli a circuito stampato.
- Porta per stampante parallela, compatibile Centronics.
- Interfaccia joystick: connettore a 9 poli tipo "D" (compatibile Kempston)
- Uscita monitor: plug audio

#### Alimentazione:

- Tensione di rete: 220 Vac, 50 Hz.





# Distribuito e garantito in Italia da Bit Shop Primavera.

## ABRUZZI

● **Lanciano** - DIGIT TEAM, Via Mario Bianco 2, Tel. (0872)37266 ● **Pescara** - CEP MICRO SYSTEM, Via Conte di Ruvo 134, Tel. (085)693750 - COMPUTER MARKET, Via Trieste 73, Tel. (085)26007

## CALABRIA

● **Catanzaro** - VISICOM, Via XX Settembre 62/A, Tel. (0961)24181 ● **Cosenza** - DEFIM, Via Massaua 25, Tel. (0984)74214 ● **Reggio Calabria** - PROTEO, Via S. Marco 8/B, Tel. (0965)21685

## CAMPANIA

● **Benevento** - DE.VI COMPUTER, Via E. Goduti 62/64, Tel. (0824)54005 ● **Casapulla** - ANDROMEDA, Via Appia 128, Tel. (0823)460469 ● **Napoli** - COMPUTER FACTORY, Via L. Giordano 40/42, Tel. (081)241242 - COMPUTER FACTORY, Via G.B. Marino 11/13, Tel. (081)621379 ● **Quarto** - S.G. CRISTOFARO, Via Campana 255bis, Tel. (081)8763676 ● **Salerno** - GENERAL COMPUTER, C.so Garibaldi 56, Tel. (089)237835

## EMILIA-ROMAGNA

● **Bologna** - EDP SISTEMI BOLOGNA, V.le Pietramellara 61/F, Tel. (051)263032 ● **Fiorenzuola d'Arda** - CENTRO COMPUTER, C.so Garibaldi 125/A, Tel. (0523)983976 ● **Forlì** - HOME & PERSONAL COMPUTER, P.zza Melozzo 1, Tel. (0543)35209 ● **Lugo** - SELCO ELETTRONICA, Via Magnapassi 26, Tel. (0545)22601 ● **Parma** - BIT SHOW, Borgo Parente 14/E, Tel. (0521)25014 ● **Piacenza** - SO.V.E.R., Via IV Novembre 60, Tel. (0523)34388 ● **Reggio Emilia** - MICROINFORMATICA, Via S. Giuseppe 4/A, Tel. (0522)34716 ● **Rimini** - COMPUTER SHOP, Via Garibaldi 44, Tel. (0541)27691 ● **Sassuolo** - MICROINFORMATICA, P.zza Martiri Partigiani 31, Tel. (0536)802955

## FRIULI VENEZIA GIULIA

● **Trieste** - COMPUTER SHOP, Via P. Reti 6, Tel. (040)61602 ● **Udine** - P.S. ELETTRONICA, Via Tavagnacco 89/91, Tel. (0432)482086

## LAZIO

● **Rieti** - ESSEEMMECI, Via Cintia 70, Tel. (0746)44704 ● **Roma** - COMPUTER MARKET, P.zza S. Donà di Piave 14, Tel. (06)7945493 - EASY-BYTE, Via G. Villani 24/26, Tel. (06)7811519

## LIGURIA

● **Genova** - COMPUTER CENTER, Via S. Vincenzo 109/R, Tel. (010)581474 ● **Genova Sampierdarena** - COMPUTER CENTER, Via G.D. Storaice 4/R, Tel. (010)454107 ● **Imperia** - COMPUTER SHOP, Via A. Doria 45, Tel. (0183)275448 ● **La Spezia** - I.L. ELETTRONICA, Via Lunigiana 481, Tel. (0187)511739 ● **Recco** - DIGIT CENTER, Via B. Assereto 78, Tel. (0185)74252

## LOMBARDIA

● **Bergamo** - SANDIT, Via S. Francesco D'Assisi 5, Tel. (035)224130 ● **Brescia** - IL COMPUTER, Via Solferino 5, Tel. (030)42100 ● **Busto Arsizio** - BUSTO BIT, Via Gavinana 17, Tel. (0331)625034 ● **Cesano Maderno** - ELECTRONIC CENTER, Via Ferrini 6, Tel. (0362)520728 ● **Cinisello Balsamo** - GBC ITALIANA, V.le Matteotti 66, Tel. (02)6181801 ● **Crema** - EL.COM., Via IV Novembre 56/58, Tel. (0373)83393 ● **Gallarate** - S.E.D., Via Arnaldo da Brescia 2, Tel. (0331)795735 ● **Lecco** - S.G.A. INFORMATICA, Via Leonardo da Vinci 7, Tel. (0341)361264 ● **Luino** - MICRO COMPUTER, V.le Amendola 48, Tel. (0332)537536 ● **Mantova** - ANTEK COMPUTER, Via Cavour 69/71, Tel. (0376)329333 ● **Milano** - GBC ITALIANA, Via Petrella 6, Tel. (02)2041051 - GBC ITALIANA, Via Cantoni 7, Tel. (02)437478 - LAS VEGAS, Galleria Manzoni 40, Tel. (02)705055 ● **Monza** - EMI, Via Az-



zone Visconti 39, Tel. (039)388275 ● **Pavia** - M3 COMPUTERS, V.le C. Battisti 4/A, Tel. (0382)31087 ● **Sesto Calende** - J.A.C. NUOVE TECNOLOGIE, Via Matteotti 38, Tel. (0331)923134 ● **Varese** - SUPERGAMES COMPUTER, Via Carrobbio 13, Tel. (0332)241092 ● **Vigevano** - VISENTIN, C.so Vittorio Emanuele 82, Tel. (0381)83833 ● **Voghera** - BYTE ELETTRONICA, Via Matteotti 65, Tel. (0382)212280

## MARCHE

● **Ancona** - CESARI, Via De Gasperi 40, Tel. (071)85620 ● **Macerata** - CERQUETELLA, Via Spalato 126, Tel. (0733)35344 ● **Senigallia** - C.I.D.I., Via Maierini 10, Tel. (071)659131

## MOLISE

● **Campobasso** - SISTEMA, Via Monsignor Bologna 10, Tel. (0874)94795 ● **Termoli** - ROSATI COMPUTER, Via Martiri della Resistenza 88, Tel. (0875)82291

## PIEMONTE

● **Alba** - PERSONAL & HOME COMPUTER, Via Paruzza 2, Tel. (0173)35441 ● **Alessandria** - BIT SYSTEM, Via Savonarola 13, Tel. (0131)445692 ● **Biel-la** - NEGRINI, Via Tripoli 32/A, Tel. (015)402861 ● **Cuneo** - THEMA, Via Statuto 10, Tel. (0171)60983 ● **Favria Canavese** - MISTER PERSONAL, Via Caporal Cattaneo 52, Tel. (0124)428344 ● **Torino** - A. B.

# BIT SHOP primavera



**LA PRIMA RETE DI SPECIALISTI  
IN PERSONAL COMPUTER.**

**BIT SHOP PRIMAVERA S.p.A.  
VIA C. FARINI 82, 20159 MILANO  
TEL. (02)6880429/68882171**

COMPUTER, C.so Grosseto 209, Tel. (011)2163665 - COMINFOR SISTEMI, C.so B. Telesio 4/B, Tel. (011)793007 - COMPUTER SHOP, Via Nizza 91, Tel. (011)6509576 - DUEGLI, C.so Racconigi 26, Tel. (011)3358756 - ZUCCA COMPUTER, Via Tripoli 179, Tel. (011)352262 ● **Verbania-Intra** - ELLIOTT COMPUTER SHOP, P.zza Don Minzoni 32, Tel. (0323)43517 ● **Vercelli** - ANALOG, Via Dionisotti 18, Tel. (0161)61105

## PUGLIA

● **Bari** - NUOVA HALET ELECTRONICS, Via Estramurale Capruzzi 192, Tel. (080)228822 - SISMET, C.so Cavour 146, Tel. (080)540733 - TECNO-SISTEMI, P.zza Garibaldi 55, Tel. (080)210084 ● **Foggia** - SINFOR, V.le Europa 44/46, Tel. (0881)32579 ● **Lecce** - CAMEL ELETTRONICA E COMPUTER, Via Marinosci 1/3, Tel. (0832)592861 ● **Taranto** - SUD COMPUTER, Via Polibio 7/A, Tel. (099)338041

## SARDEGNA

● **Cagliari** - BIT SHOP, Via Zagabria 47, Tel. (070)490954

## SICILIA

● **Caltanissetta** - ELEONORI & AMICO, C.so Umberto I 254/256, Tel. (0934)25610 ● **Catania** - ARIA NUOVA, P.zza Lanza 16, Tel. (095)438573 ● **Messina** - TEMPO REALE, Via del Vespro 71, Tel. (090)773983 ● **Palermo** - INFORMATICA COMMERCIALE, Via Notarbartolo 23/B, Tel. (091)291500 ● **Siracusa** - LOGOL SYSTEM, V.le Scala Greca 339/9, Tel. (0931)53244

## TOSCANA

● **Firenze** - ANDREI CARLO & C., Via G. Milanese 28/30, Tel. (055)472810 - ELETTRONICA CENTOSTELLE, Via Centostelle 5/A, Tel. (055)610251 ● **Livorno** - C.P.E. ELETTRONICA, Via Paoli 32, Tel. (0586)27357 ● **Lucca** - LOGOS INFORMATICA, V.le S. Concordio 160, Tel. (0583)55519 ● **Pisa** - IT-LAB, Via XXIV Maggio 101, Tel. (050)552590 - TECNOVAVAS COMPUTER, Via Emilia 36, Tel. (050)502516 ● **Pistoia** - C.D.E., V.le Adua 350, Tel. (0573)400712 ● **Prato** - C. BARBAGLI ELETTRONICA, Via E. Boni 76/80, Tel. (0574)595001 ● **Viareggio** - C.D.E., Via. A. Volta 79, Tel. (0584)942244

## TRENTINO ALTO ADIGE

● **Bolzano** - BONTADI, P.zza Verdi 15/B, Tel. (0471)971619 ● **Rovereto** - S.E.D.A., Via Fontana 8/B, Tel. (0464)34506 ● **Trento** - S.E.D.A., Via Sighele 7/1, Tel. (0461)984564

## UMBRIA

● **Perugia** - STUDIO SYSTEM, Via D'Andreotto 49/55, Tel. (075)754964

## VALLE D'AOSTA

● **Aosta** - INFORMATIQUE, Av. Du Conseil des Commis 16, Tel. (0165)362242

## VENETO

● **Bassano del Grappa** - TODARO, Via Jacopo da Ponte 51, Tel. (0424)22810 ● **Castelfranco Veneto** - E.D.S., Via S. Pio X 154, Tel. (0423)497151 ● **Conegliano Veneto** - E.D.S., V.le Italia 132, Tel. (0438)62345 ● **Mirano** - SAVING ELETTRONICA, Via Gramsci 40, Tel. (041)432876 ● **Padova** - COMPUTER POINT, Via Roma 63, Tel. (049)22564 - S.I.C. ITALIA, Via Nicolò Tommaseo 13, Tel. (049)663133 ● **San Donà di Piave** - COMPUTIME, P.zza Rizzo 61, Tel. (0421)50474 ● **Venezia** - PERSONAL COMPUTER, Cannaregio 5894, Tel. (041)29040 ● **Verona** - CHIP COMPUTER, Stradone S. Fermo 7, Tel. (045)21255 ● **Vicenza** - A.T.R., Via del progresso 7/9, Tel. (0444)564611

## Listato 1 - Il programma Bridge-Mem-Class.

```

10 CLEAR
12 DIM PP(25,120)
13 DIM PA(41,3): REM      MATRIC
    EDEI PUNTEGGI E DELLE COPPIE
    DA INSERIRE
14 DIM CL%(100,2)
15 DIM REC%(100)
16 DIM RCE%(100)
17 DIM GG%(100)
30 REM
31 W = 0:ZZ$ = "S":D$ = CHR$(4)

33 MM = 0:MMEN1 = - 1
34 MP = 0
35 HOME
36 INVERSE
37 HTAB 10: PRINT "CIRCOLO I PU
    GLIESI "
38 NORMAL
39 PRINT : PRINT "PROGRAMMA PER
    IL CALCOLO DEGLI "
41 PRINT "SCORES IN UN TORNEO MI

```

```

TCELL "
44 VTAB 6: INVERSE : PRINT "VINC
    ENZO DELLE CAVE -1984 "
45 PRINT "VIETATE LE COPIE!!!
    ": NORMAL
50 VTAB 9: PRINT "1) GESTIONE TO
    RNEO"
53 PRINT : PRINT "2) VISUALIZZAZ
    IONE MANI"
55 PRINT : PRINT "3) FINE"
56 PRINT : PRINT "4) CLASSIFICA
    INDIVIDUALE "
57 PRINT : PRINT "5) INIZIALIZZA
    CLASSIFICA "
58 PRINT : PRINT "6) VARIAZIONE
    PUNTEGGI "
59 VTAB 23: INPUT "SCEGLI FRA LE
    SEGUENTI OPZIONI ";KK$
60 KK = VAL (KK$)
62 IF KK = 1 THEN 197
64 IF KK = 2 THEN 8500
66 IF KK = 3 THEN 9000
67 IF KK = 4 THEN 1000
68 IF KK = 5 THEN 1500
69 IF KK = 6 THEN 7000

```

modifica nomi. Digitare 1 e inserire l'elenco dei giocatori partecipanti alla classifica finale. Con l'opzione 2 si potranno eventualmente modificare tali nomi.

Il programma riempie il file Anagrafica attribuendo un codice progressivo in ordine di inserimento dei nomi, e inializza a -1 il file Classifica e a 0 il file Turno. (per i nomi usare max 30 caratteri)

L'opzione 4 del menu principale permette a questo punto di avere l'elenco dei giocatori col relativo punteggio (0 ovviamente) nell'ordine di inserimento. Digitare a questo punto 4; alla domanda numero turni da eliminare rispondere con -1 ed attendere la stampa; tale stampa tornerà utile in seguito. Ora si può iniziare ad eseguire i conti del primo torneo selezionando l'opzione 1.

Al termine dei conti viene stampato il prospetto relativo e la classifica del singolo torneo.

Quindi, e questa è la variante rispetto al programma Bridge-Conti presentato precedentemente, il programma chiede di inserire i codici di tutti i giocatori partecipanti al torneo, sempre che si desideri aggiornare la classifica dei giocatori.

Tali codici sono quelli che il program-

ma aveva attribuito ai giocatori durante la fase di inserimento nomi (inizializzazione).

Per inserire tali codici ci tornerà senz'altro utile la stampa ottenuta al termine della fase di inizializzazione.

Può capitare che partecipi a un torneo qualche giocatore che non faccia parte dell'archivio Anagrafica, cioè che non partecipa alla classifica finale: tali giocatori verranno identificati col codice 99.

Terminata questa fase viene stampata una classifica delle coppie Nord-Sud ed Est-Ovest con a fianco il punteggio che il programma attribuisce alle coppie, in funzione della classifica ottenuta. (Tale punteggio è ottenuto come se la classifica finale rappresentasse un ipotetico "score" del torneo, cioè la prima coppia classificata rende  $2 \star (x-1)$  punti, dove  $x$  è il numero di tavoli del torneo, e così via per gli altri piazzamenti).

A questo punto il programma provvede a riempire il file Classifica. Tale file non è altro che una matrice di 100 righe (numero max giocatori partecipanti alla classifica finale) e 30 colonne (numero max di tornei della classifica finale). In questo modo viene aggiornata la colonna di tale matrice relativa al torneo giocato.

### Classifica individuale (opzione 4)

Il programma chiede il numero di risultati che si desidera scartare (max 5, e comunque un numero inferiore al numero di tornei giocati) e il numero di giocatori interessati alla classifica. Il problema della eliminazione dei risultati è stato risolto molto semplicemente. In fase di inizializzazione, come già detto, tutti i record del file Classifica vengono inializzati a -1. Per cui, se per esempio dopo 10 tornei per il giocatore di codice 20 sono presenti 3 campi col valore -1, ciò sta a significare che tale giocatore non ha giocato 3 tornei; pertanto se si voleva una classifica che scartasse i quattro peggiori risultati, per tale giocatore si dovrà scartare un solo punteggio. Il numero di risultati utili per ogni giocatore è perciò  $RU = TU - NT$  (se  $Cont < NT$ ), oppure  $RU = TU - Cont$  (se  $Cont > NT$ ), dove  $TU$  è il numero di tornei giocati,  $NT$  è il numero di tornei da scartare,  $Cont$  è il numero di tornei non giocati.

Con tale procedimento è bastato eseguire un ordinamento decrescente dei primi  $TU$  elementi del vettore contenente i punteggi del giocatore ed eseguire poi la somma dei primi  $RU$  elementi.

Per il giocatore dell'esempio, verranno in tal caso scartati i tre -1 relativi ai tornei cui non ha partecipato e il peggior risultato conseguito.

Il programma a questo punto effettua una conversione codice-nome del giocatore accedendo al file Anagrafica e stampa la classifica aggiornata dei giocatori interessati. Successivamente viene stampato un prospetto, ordinato per codice, del giocatore in cui è evidenziato il nome del giocatore e i punteggi ottenuti in tutti i tornei disputati.

L'opzione 6 del menu principale consente di correggere eventuali errori dovuti all'inserimento di codici errati durante l'esecuzione del programma Brid-

ge-Cont-Class. Ciò infatti comporterebbe l'attribuzione di un punteggio ad un giocatore invece che ad un altro, pregiudicando, ovviamente, la classifica finale di tutti i giocatori. Se si dovesse verificare tale inconveniente è necessario selezionare tale opzione e successivamente digitare il numero del torneo dove si è verificato l'errore, il codice del giocatore interessato, e il punteggio esatto (inteso come punteggio-classifica calcolato col solito metodo).

Illustriamo ora brevemente le linee di programma relative alle modifiche apportate.

A) Per il programma Bridge-Mem-Class.

50-70 - Nuovo menu principale.

1300 - Stampa prospetto punteggio tornei.

1301-1340 - Stampa prospetto punteggio tornei.

1500-2110 - Routine di inizializzazione della classifica: fase di inserimento e modifica nomi.

3000-3070 - Semplice routine di sort (algoritmo Shell-Sort) molto veloce.

B) Per il programma Bridge-Cont-Class.

1900-3700 - Aggiornamento del file Classifica in base ai codici dei giocatori ed in funzione della posizione ottenuta: incremento del file Turno contenente il numero di tornei giocati e stampa dei punti classifica ottenuti dalle varie coppie.

#### Seguito programma Bridge-Mem-Class.

```

70 GOTO 35
197 REM
207 REM
217 REM
227 HOME
237 INPUT "N.RO MANI GIOCATE(BOA
RDS)";N
240 IF N > 40 THEN HOME : GOTO
237
247 REM
257 INPUT "DA QUANTE COPPIE E' S
TATA GIOCATO OGNI MANO ? ";X

260 IF X > 21 THEN HOME : GOTO
257
267 REM
270 RIC = X
277 PUNTMED = X - 1
287 XX = X + 5
297 INPUT "QUANTI TAVOLI SONO ?
";YY
300 IF YY > 40 THEN HOME : GOTO
297
307 NUM = INT (YY / 2)
317 FLAG = YY - INT (YY / 2) * 2
: REM FLAG=1 TAVOLI DISPAR
I,FLAG=0 TAVOLI PARI
327 REM
337 REM
347 REM
357 REM
367 REM
377 REM
387 REM
397 REM
407 REM
417 REM
427 REM

```

```

437 REM
467 REM
477 REM
487 REM
493 REM
494 REM
496 REM
499 REM INIZIO ROUTINE NUOVA

500 FOR NB = 1 TO N
510 HOME
515 INVERSE : HTAB 15: PRINT "BO
ARD N. ";NB: NORMAL : PRINT
: PRINT : PRINT
520 INPUT "NUMERO PRIMA COPPIA N
-S ";KK#
530 PA(1,1) = VAL (KK#)
540 VTAB 10: PRINT "OK? ": GET
KK#
550 IF KK# = "N" THEN 510
560 IF KK# < > "S" THEN 540
570 PA(1,3) = PA(1,1) + 100
580 GOSUB 6000
590 FOR I = 1 TO X
600 PP(I,NB * 3 - 2) = PA(I,1)
610 PP(I,NB * 3) = PA(I,3)
620 NEXT I
630 PRZ = 0
635 FOR I = 1 TO X:PA(I,2) = 888
8: NEXT I
640 GOSUB 5000
650 FOR I = 1 TO RIC
660 PP(I,NB * 3 - 1) = PA(I,2)
670 NEXT I
680 NEXT NB
690 GOSUB 8000
700 IF MP = 0 THEN 910
710 PRZ = RIC
720 FOR NB = 1 TO N
730 HOME

```



```

750 FOR I = 1 TO X
760 PA(I,1) = PP(I,NB * 3 - 2)
770 PA(I,3) = PP(I,NB * 3)
780 NEXT I
790 FOR I = 1 TO RIC
800 PA(I,2) = PP(I,NB * 3 - 1)
810 NEXT I
815 FOR I = RIC + 1 TO X:PA(I,2)
    = 8888: NEXT I
820 GOSUB 5000
830 FOR I = RIC + 1 TO X
840 PP(I,NB * 3 - 1) = PA(I,2)
850 NEXT I
860 NEXT NB
865 RIC = X
870 GOSUB 8000
910 GOTO 8705
1000 REM CLASSIFICA INDIVIDUALE

1001 CLEAR
1002 DIM PP%(100,31): DIM CL%(10
    0,2): DIM REC%(100): DIM RCE
    %(100): DIM GG%(100): DIM G1
    $(100)
1003 D$ = CHR$(4)
1005 CNT = 0
1010 MEN1 = -1: ZERO = 0: SUM% =
    0
1020 PRINT : PRINT D$;"OPEN TURN
    0"
1030 PRINT D$;"READ TURNO"
1040 INPUT TU:UT = TU
1050 PRINT D$;"CLOSE TURNO"
1060 HOME : INPUT "N.TURNI DA EL
    IMINARE (MAX. 5) ";NT$:NT =
    VAL (NT$)
1070 IF NT > 5 OR NT > TU - 1 THEN
    1060
1080 HOME : INPUT "N.RO GIOCATOR
    I INTERESSATI (MAX 100) ";NG
    $:NG = VAL (NG$)
1090 IF NG > 100 THEN 1080
1100 PRINT : PRINT D$;"OPEN CLAS
    SIFICA,L150"
1110 FOR J = 1 TO NG
1120 PRINT D$;"READ CLASSIFICA,R
    "J
1130 FOR I = 1 TO TU
1140 INPUT REC%(I): IF REC%(I) =
    -1 THEN CNT = CNT + 1
1150 NEXT I
1160 GOSUB 3000
1170 IF CNT > NT THEN RU = TU -
    CNT: GOTO 1190
1180 RU = TU - NT
1190 FOR I = 1 TO RU
1195 IF REC%(I) < 0 THEN 1210
1200 SUM% = SUM% + REC%(I)
1210 NEXT I
1220 CL%(J,1) = J:CL%(J,2) = SUM%

```

```

1221 SUM% = 0: CNT = 0
1225 PRINT J
1230 NEXT J
1240 PRINT D$;"CLOSE CLASSIFICA"

1242 UT = NG: FOR I = 1 TO NG:REC
    %(I) = CL%(I,2):RCE%(I) = CL
    %(I,1): NEXT I
1243 GOSUB 3000
1244 PRINT : PRINT D$;"OPEN ANAG
    RAFICA,L30"
1245 FOR I = 1 TO NG
1246 PRINT D$;"READ ANAGRAFICA,R
    "RCE%(I)
1247 INPUT GG%(I):G1$(RCE%(I)) =
    GG%(I)
1248 NEXT I
1250 PRINT : PRINT D$;"PR#1"
1260 PRINT "CLASSIFICA DOPO ";TU
    ;" TURNI DI GIOCO E ";NT;" R
    ISULTATI SCARTATI": PRINT : PRINT
    : PRINT
1270 FOR I = 1 TO NG
1280 PRINT GG%(I); SPC(30 - LEN
    (GG%(I)));REC%(I)
1290 NEXT I
1300 PRINT : PRINT D$;"OPEN CLAS
    SIFICA,L150"
1301 FOR I = 1 TO NG
1302 PRINT D$;"READ CLASSIFICA,R
    "I
1303 FOR J = 1 TO TU: INPUT PP%(
    I,J)
1304 NEXT J: NEXT I
1305 PRINT D$;"CLOSE CLASSIFICA"

1306 PRINT : PRINT : PRINT : PRINT
    : PRINT : PRINT : PRINT SPC(
    30);
1308 FOR I = 1 TO TU
1310 I$ = STR$(I)
1312 PRINT I$; SPC(4 - LEN (I$
    ));
1314 NEXT I
1316 PRINT : PRINT
1318 FOR I = 1 TO NG
1320 PRINT G1$(I); SPC(30 - LEN
    (G1$(I)));
1322 FOR J = 1 TO TU
1324 PP$ = STR$(PP%(I,J))
1326 IF PP$ = "-1" THEN PP$ = "-
    "
1328 PRINT PP$; SPC(4 - LEN (P
    P$));
1330 NEXT J
1334 PRINT
1336 NEXT I
1340 PRINT D$;"PR#0": GOTO 10
1500 REM INIZIALIZZA CLASSIFIC
    A

```

Seguito programma Bridge-Mem-Class.

```

1502 HOME : PRINT "INIZIALIZZAZI
ONE CLASSIFICA": PRINT "CONF
ERMATA? (S/N) ";: GET KK$
1504 IF KK$ < > "S" THEN 30
1505 HOME : INVERSE : HTAB 8: PRINT
"FASE DI INIZIALIZZAZIONE": NORMAL
1506 VTAB 10: PRINT "1) INSERIME
NTO NOMI": PRINT : PRINT "2)
MODIFICA NOMI ": VTAB 23: PRINT
"COSA SCEGLI? ";: INPUT KK$
1508 IF VAL (KK$) = 0 OR VAL (
KK$) > 2 THEN 1506
1510 IF VAL (KK$) = 2 THEN 1600

1512 HOME : INPUT "N. GIOCATORI
INTERESSATI ";NG$:NG = VAL
(NG$)
1514 HOME : PRINT "INSERISCI I N
OMI DEI GIOCATORI ": PRINT :
PRINT
1516 FOR I = 1 TO NG
1518 PRINT I;: HTAB 5: INPUT GG$
(I): NEXT I
1520 PRINT : PRINT D$;"OPEN ANAG
RAFICA,L30"
1522 FOR I = 1 TO NG
1524 PRINT D$;"WRITE ANAGRAFICA,
R"I
1526 PRINT GG$(I): NEXT I
1528 PRINT D$;"CLOSE ANAGRAFICA"

1530 GOTO 2010
1600 REM MODIFICA NOMI
1602 HOME : PRINT "INSERISCI COD
ICE DA MODIFICARE ": INPUT N
G$:NG = VAL (NG$)
1604 PRINT "INSERISCI NUOVO NOME
": INPUT GC$
1606 PRINT : PRINT D$;"OPEN ANAG
RAFICA,L30"
1608 PRINT D$;"WRITE ANAGRAFICA,
R"NG
1610 PRINT GC$
1612 PRINT D$;"CLOSE ANAGRAFICA"

1616 VTAB 22: PRINT "TERMINATO (
S/N) ";: GET KK$
1618 IF KK$ < > "N" THEN 10
1620 GOTO 1602
2010 PRINT : PRINT D$;"OPEN CLAS
SIFICA,L150"
2020 FOR I = 1 TO 100
2030 PRINT D$;"WRITE CLASSIFICA,
R"I
2040 FOR J = 1 TO 30: PRINT MENI
: NEXT J
2050 NEXT I
2060 PRINT D$;"CLOSE CLASSIFICA"

2070 PRINT : PRINT D$;"OPEN TURN
O"
2080 PRINT D$;"WRITE TURNO"
2090 PRINT ZERO
2100 PRINT D$;"CLOSE TURNO"
2110 GOTO 10
3000 REM GOSUB ORDINAMENTO
3010 FOR G = 1 TO UT:M = 1
3020 M = 2 * M: IF M < = UT THEN
3020
3030 M = INT (M / 2): IF M = 0 THEN
RETURN
3035 IF UT = M THEN 3030
3040 FOR G = 1 TO UT - M:C = G
3050 D = C + M: IF REC%(C) > = R
EC%(D) THEN 3070
3060 AA% = REC%(C):REC%(C) = REC%
(D):REC%(D) = AA%:AA% = RCE%
(C):RCE%(C) = RCE%(D):RCE%(D
) = AA%:C = C - M: IF C > 0 THEN
3050
3070 NEXT : GOTO 3030
5000 HOME
5010 ZZ$ = "N-S PUNTEGGIO E-O
BOARD N."
5030 INVERSE : PRINT ZZ$: HTAB 3
9: VTAB 1: PRINT NB: NORMAL
5040 FOR I = 1 TO X
5050 VTAB I + 1: HTAB 1: PRINT P
A(I,1)
5055 VTAB I + 1: HTAB 8: IF PA(I
,2) = - 9999 THEN PRINT "M
": GOTO 5070
5056 VTAB I + 1: HTAB 8: IF PA(I
,2) = 8888 THEN PRINT " ": GOTO
5070
5060 VTAB I + 1: HTAB 8: PRINT P
A(I,2)
5070 VTAB I + 1: HTAB 15: PRINT
PA(I,3)
5080 NEXT I
5090 HTAB 31: VTAB 23: PRINT "OK
?"
5110 FOR I = PRZ TO X - 1
5120 VTAB I + 2: HTAB 8
5130 INPUT " ";KK$
5131 HTAB 15: VTAB I + 2: PRINT
PA(I + 1,3)
5132 IF KK$ = "" AND PA(I + 1,2)
= - 9999 THEN HTAB 8: VTAB
I + 2: PRINT "M": HTAB 15: VTAB
I + 2: PRINT PA(I + 1,3): GOTO
5170
5133 IF KK$ = "" AND PA(I + 1,2)
= 8888 THEN HTAB 8: VTAB I
+ 2: PRINT " ": HTAB 15: VTAB
I + 2: PRINT PA(I + 1,3): GOTO
5170
5134 IF KK$ = "" THEN HTAB 8: VTAB
I + 2: PRINT PA(I + 1,2): HTAB

```

```

15: VTAB I + 2: PRINT PA(I +
1,3): GOTO 5170
5140 IF KK$ = CHR$(6) THEN MP =
1: RIC = I - PRZ: GOTO 5180
5150 IF KK$ = "M" THEN PA(I + 1,
2) = - 9999: GOTO 5170
5160 PA(I + 1,2) = VAL (KK$)
5170 NEXT I
5180 VTAB 23: HTAB 34
5190 GET KK$
5200 IF KK$ = "N" THEN 5110
5210 IF KK$ < > "S" THEN 5190
5220 RETURN
6000 REM CALCOLA COPPIE N-S
6020 FOR I = 1 TO X - 1
6030 IF PA(I,1) < > 1 THEN PA(I
+ 1,1) = PA(I,1) - 1: GOTO
6050
6040 PA(I + 1,1) = YY
6050 NEXT I
6060 REM CALCOLA COPPIE E-O
6070 IDEA = 0
6080 FOR I = 1 TO X - 1
6090 IF I = NUM THEN IDEA = 1: GOTO
6120
6100 IF FLAG = 1 THEN IDEA = 0
6110 GOTO 6140
6120 IF FLAG = 1 THEN IDEA = 0
6130 IF PA(I,3) = 103 GOTO 6220
6140 IF PA(I,3) = 101 GOTO 6180
6150 IF PA(I,3) = 102 GOTO 6200
6160 PA(I + 1,3) = PA(I,3) - 2 -
IDEA
6170 GOTO 6230
6180 PA(I + 1,3) = 100 + YY - 1 -
IDEA
6190 GOTO 6230
6200 PA(I + 1,3) = 100 + YY - IDE
A
6210 GOTO 6230
6220 PA(I + 1,3) = 100 + YY - FLA
G * (YY - 1)
6230 IDEA = 0
6240 NEXT I
6250 RETURN
7000 REM VARIAZIONE PUNTEGGI
7010 PRINT : PRINT D$;"OPEN TURN
O"
7020 PRINT D$;"READ TURNO"
7030 INPUT TU
7035 PRINT D$;"CLOSE TURNO"
7040 HOME : INPUT "N.TORNEO ";NT
: IF NT > TU THEN 7040
7050 INPUT "CODICE GIOCATORE ";C
G: INPUT "NUOVO PUNTEGGIO ";
NP%
7060 HOME : INVERSE : PRINT "ATT
ENDERE VARIAZIONE IN ATTO": NORMAL
7070 PRINT : PRINT D$;"OPEN CLASS
IFICA,L150"

```

```

7080 PRINT D$;"READ CLASSIFICA,R
"CG
7090 FOR I = 1 TO 30: INPUT REC%
(I): NEXT I
7100 REC%(NT) = NP%
7110 PRINT D$;"WRITE CLASSIFICA
,R"CG
7120 FOR I = 1 TO 30: PRINT REC%
(I): NEXT I
7130 PRINT D$;"CLOSE CLASSIFICA
"
7140 HOME : PRINT "ALTRE MODIFIC
HE? S/N ";: GET KK$
7150 IF KK$ < > "N" THEN 7040
7160 GOTO 10
8000 PRINT : PRINT D$;"OPEN TORN
EO"
8010 PRINT D$;"DELETE TORNEO"
8020 PRINT D$;"OPEN TORNEO"
8030 PRINT D$;"WRITE TORNEO"
8040 FOR J = 1 TO N * 3
8050 FOR I = 1 TO RIC
8060 PRINT PP(I,J)
8070 NEXT I: NEXT J
8090 PRINT D$;"CLOSE TORNEO"
8091 PRINT : PRINT D$;"OPEN DATI
"
8092 PRINT D$;"DELETE DATI"
8093 PRINT D$;"OPEN DATI"
8094 PRINT D$;"WRITE DATI"
8095 PRINT X: PRINT N: PRINT YY
8096 PRINT D$;"CLOSE DATI"
8100 RETURN
8500 REM
8501 PRINT : PRINT D$;"OPEN DATI"
8502 PRINT D$;"READ DATI"
8504 INPUT X: INPUT N: INPUT YY
8505 PRINT D$;"CLOSE DATI"
8520 HOME : VTAB 15: INPUT "BATT
I IL N.DELLA MANO DA VISUALI
ZZARE ";KK$
8525 KK = VAL (KK$)
8530 IF KK > N THEN PRINT "NON
C'E' !!": GET KK$: GOTO 8520
8540 PRINT : PRINT D$;"OPEN TORN
EO"
8550 PRINT D$;"READ TORNEO"
8560 FOR J = 1 TO N * 3
8570 FOR I = 1 TO X
8580 INPUT PP(I,J)
8590 NEXT I: NEXT J
8600 PRINT D$;"CLOSE TORNEO"
8601 REM
8610 IF KK$ = "" THEN 8800
8620 FOR I = 1 TO X
8630 PA(I,1) = PP(I, KK * 3 - 2)
8640 PA(I,2) = PP(I, KK * 3 - 1)
8650 PA(I,3) = PP(I, KK * 3)

```

Seguito programma Bridge-Mem-Class.

```

8660 NEXT I
8670 NB = KK
8680 GOSUB 5000
8690 RIC = X
8691 FOR I = 1 TO X
8692 PP(I, KK * 3 - 2) = PA(I, 1)
8693 PP(I, KK * 3 - 1) = PA(I, 2)
8694 PP(I, KK * 3) = PA(I, 3)
8695 NEXT I
8700 GOSUB 8000
8701 HOME : PRINT "VUOI I CONTI?"
      : GET KK$
8703 IF KK$ = "N" THEN 8710
8704 IF KK$ < > "S" THEN 8701
8705 HOME : INVERSE : PRINT "ATT
      ENDERE PER I CONTI": NORMAL
      : PRINT : PRINT D$;"RUN BRID
      GE-CONT-CLASS"
8710 GOTO 10
8800 FOR NB = 1 TO N
8810 FOR I = 1 TO X
8820 PA(I, 1) = PP(I, NB * 3 - 2)
8830 PA(I, 2) = PP(I, NB * 3 - 1)
8840 PA(I, 3) = PP(I, NB * 3)
8850 NEXT I
8860 GOSUB 5000
8861 FOR I = 1 TO X
8862 PP(I, NB * 3 - 2) = PA(I, 1)
8863 PP(I, NB * 3 - 1) = PA(I, 2)
8864 PP(I, NB * 3) = PA(I, 3)
8865 NEXT I
8870 NEXT NB
8880 RIC = X
8890 GOSUB 8000
8895 GOTO 8701
9000 HOME
10000 HOME

```

Listato 2 - Il programma Bridge-Cont-Class.

```

10 D$ = CHR$(4)
40 DIM PP$(25, 123)
70 DIM PA$(41, 3): REM      MATRI
      CE DEI PUNTEGGI E DELLE COPP
      IE DA INSERIRE
100 DIM MNS$(41, 41)
130 DIM MEO$(41, 41)
160 DIM A$(41, 2): REM      MATRIC
      EDEI PUNTEGGI N-S
190 DIM C$(41, 2): REM      MATRIC
      EDEI PUNTEGGI E-O
220 DIM VHR$(50)
250 DIM REC$(31)
280 DIM CNS$(51)
310 DIM CEO$(51)
340 DIM D$(51, 2)
370 DIM CSN$(51)
400 DIM COE$(51)
430 ZZ$ = "S"
460 W = 0
490 HOME : INVERSE : PRINT "ATTE
      NDERE PER I CONTI": NORMAL :
      PRINT : PRINT : PRINT
520 GOSUB 8620
550 GOSUB 5230
580 REM
610 GOSUB 6100
640 GOSUB 4000
670 REM
700 X = YY:LL = 1
730 FOR I = 1 TO X
760 PA$(I, 1) = MNS$(I, 0):PA$(I, 2)
      = MNS$(I, N + 1)
790 NEXT I
820 GOSUB 6280
850 CL = 1:MM = 0:X = YY:XX = YY +
      5
880 GOSUB 6940
910 FOR I = 1 TO YY
940 REM
970 D$(I, 1) = A$(I, 1):D$(I, 2) = A
      $(I, 2): NEXT I
1000 REM
1030 REM
1060 REM
1090 REM
1120 REM
1150 REM
1180 PRINT : PRINT D$;"PR#1"
1210 PRINT : PRINT "C L A S S I
      F I C A      N - S": PRINT : PRINT
      : PRINT : PRINT
1240 GOSUB 3850
1270 FOR I = 1 TO X
1300 PA$(I, 1) = MEO$(I, 0):PA$(I, 2)
      ) = MEO$(I, N + 1)
1330 NEXT I:LL = 1:GOSUB 6280
1360 CL = 1:MM = 0:X = YY:XX = YY

```

```

+ 5
1390 GOSUB 6940
1420 REM
1450 REM
1480 REM
1510 REM
1540 REM
1570 REM
1600 REM
1630 REM
1660 REM
1690 REM
1720 REM
1750 REM
1780 REM
1810 REM
1840 PRINT : PRINT : PRINT "C L
A S S I F I C A   E - 0": PRINT
: PRINT : PRINT : PRINT
1870 GOSUB 3850
1900 PRINT : PRINT D$;"PR#0": HOME
: PRINT "VUOI AGGIORNARE LA
CLASSIFICA? (S/N) ": GET KK$

1930 IF KK$ < > "S" THEN 3970
1960 HOME
1990 PRINT "DAMMI IL CODICE DELL
E COPPIE N-S "
2020 FOR I = 1 TO YY
2050 PRINT I;: PRINT " ";: INPUT
CNS%(I): HTAB 5: INPUT CSN%(
I)
2080 IF CNS%(I) > 100 OR CSN%(I)
> 100 THEN HOME : PRINT "E
R R O R E": GOTO 1990
2110 NEXT I
2140 PRINT "CONFERMATO (S/N)? ":
GET KK$
2170 IF KK$ < > "S" THEN 1960
2200 HOME
2230 PRINT "DAMMI IL CODICE DELL
E COPPIE E-0 "
2260 FOR I = 1 TO YY
2290 PRINT 100 + I;: PRINT " "
;: INPUT COE%(I): HTAB 7: INPUT
COE%(I)
2320 IF COE%(I) > 100 OR COE%(I)
> 100 THEN HOME : PRINT "E
R R O R E": GOTO 2230
2350 NEXT I
2380 PRINT "CONFERMATO (S/N)? ":
GET KK$
2410 IF KK$ < > "S" THEN 2200
2440 PRINT : PRINT D$;"OPEN TURN
0"
2470 PRINT D$;"READ TURNO"
2500 INPUT TU
2530 TU = TU + 1
2560 PRINT D$;"CLOSE TURNO"
2590 PRINT : PRINT D$;"OPEN TURN

```

```

0"
2620 PRINT D$;"WRITE TURNO"
2650 PRINT TU
2680 PRINT D$;"CLOSE TURNO"
2710 PRINT : PRINT D$;"OPEN CLAS
SIFICA,L150"
2740 FOR J = 1 TO YY
2770 PRINT D$;"READ CLASSIFICA,R
"CNS%(D%(J,1))
2800 FOR I = 1 TO 30
2830 INPUT REC%(I)
2860 NEXT I
2890 REC%(TU) = D%(J,2)
2920 PRINT D$;"WRITE CLASSIFICA,
R"CNS%(D%(J,1))
2950 FOR I = 1 TO 30
2980 PRINT REC%(I)
3010 NEXT I
3040 NEXT J
3044 FOR J = 1 TO YY
3046 PRINT D$;"READ CLASSIFICA,R
"CSN%(D%(J,1))
3048 FOR I = 1 TO 30
3050 INPUT REC%(I)
3052 NEXT I
3054 REC%(TU) = D%(J,2)
3056 PRINT D$;"WRITE CLASSIFICA,
R"CSN%(D%(J,1))
3058 FOR I = 1 TO 30
3060 PRINT REC%(I)
3062 NEXT I
3064 NEXT J
3070 FOR J = 1 TO YY
3100 PRINT D$;"READ CLASSIFICA,R
"CEO%(A%(J,1) - 100)
3130 FOR I = 1 TO 30
3160 INPUT REC%(I)
3190 NEXT I
3220 REC%(TU) = A%(J,2)
3250 PRINT D$;"WRITE CLASSIFICA,
R"CEO%(A%(J,1) - 100)
3280 FOR I = 1 TO 30
3310 PRINT REC%(I)
3340 NEXT I
3370 NEXT J
3372 FOR J = 1 TO YY
3374 PRINT D$;"READ CLASSIFICA,R
"COE%(A%(J,1) - 100)
3376 FOR I = 1 TO 30
3378 INPUT REC%(I)
3380 NEXT I
3382 REC%(TU) = A%(J,2)
3384 PRINT D$;"WRITE CLASSIFICA,
R"COE%(A%(J,1) - 100)
3386 FOR I = 1 TO 30
3388 PRINT REC%(I)
3390 NEXT I
3392 NEXT J
3400 PRINT D$;"CLOSE CLASSIFICA"
3430 PRINT : PRINT D$;"PR#1"

```

Seguito programma Bridge-Cont-Class.

```

3460 FOR I = 1 TO YY
3490 PRINT D%(I,1);" ";D%(I,2)

3520 NEXT I
3550 PRINT : PRINT : PRINT : PRINT

3580 FOR I = 1 TO YY
3610 PRINT A%(I,1);" ";A%(I,2)

3640 NEXT I
3670 PRINT D$;"PR#0"
3700 GOTO 3970
3730 REM
3760 REM
3790 REM
3820 REM
3850 FOR I = 1 TO X
3880 PRINT PA%(I,1);: PRINT SPC(
5 - LEN ( STR$ (PA%(I,1))))
;: PRINT PA%(I,2)
3910 NEXT I
3940 RETURN
3970 PRINT : PRINT D$;"RUN BRIDGE
-MEM-CLASS"
4000 PRINT : PRINT D$;"PR#1"
4030 PRINT CHR$(9);"132N"
4090 PRINT " ": PRINT " ": PRINT
" ": PRINT " "
4120 REM STAMPA
4150 S = N
4180 FOR I = 0 TO S
4210 PRINT SPC( 3 - LEN ( STR$
(MNS%(0,I))))MNS%(0,I);
4240 NEXT I
4270 PRINT " ";"TOT";
4300 PRINT " "
4330 FOR I = 1 TO YY
4360 PRINT " "
4390 FOR J = 0 TO S + 1
4420 IF MNS%(I,J) = 7777 THEN PRINT
" -";: GOTO 4480
4450 PRINT SPC( 3 - LEN ( STR$
(MNS%(I,J))))MNS%(I,J);
4480 IF J = S THEN PRINT " ";
4510 NEXT J
4540 PRINT " "
4570 NEXT I
4600 FOR I = 1 TO 25
4630 PRINT " "
4660 NEXT
4690 FOR I = 0 TO S
4720 PRINT SPC( 3 - LEN ( STR$
(MEO%(0,I))))MEO%(0,I);
4750 NEXT I
4780 PRINT " ";"TOT";
4810 PRINT " "
4840 FOR I = 1 TO YY
4870 PRINT " "

```

```

4900 FOR J = 0 TO S + 1
4930 IF MEO%(I,J) = 7777 THEN PRINT
" -";: GOTO 4990
4960 PRINT SPC( 3 - LEN ( STR$
(MEO%(I,J))))MEO%(I,J);
4990 IF J = S THEN PRINT " ";
5020 NEXT J: PRINT " "
5050 NEXT I
5080 FOR I = 1 TO 25
5110 PRINT " "
5140 NEXT
5170 PRINT D$;"PR#0"
5200 RETURN
5230 RIC = X
5260 PUNTMED = X - 1
5290 XX = X + 5
5320 NUM = INT (YY / 2)
5350 FLAG = YY - INT (YY / 2) *
2
5380 FOR I = 1 TO XX
5410 VHR%(I) = 0
5440 NEXT I
5470 REM
5500 FOR P = 0 TO YY
5530 MNS%(P,0) = P
5560 MEO%(P,0) = P + 100
5590 NEXT
5620 FOR P = 0 TO N
5650 MNS%(0,P) = P
5680 MEO%(0,P) = P
5710 NEXT
5740 FOR I = 1 TO YY
5770 FOR J = 1 TO N
5800 MNS%(I,J) = 7777
5830 MEO%(I,J) = 7777
5860 NEXT : NEXT
5890 RETURN
5920 W = W + 1
5950 FOR I = 1 TO X
5980 MNS%(A%(I,1),W) = A%(I,2)
6010 MEO%(C%(I,1) - 100,W) = (X -
1) * 2 - C%(I,2)
6040 NEXT I
6070 RETURN
6100 FOR NB = 1 TO N
6130 FOR J = 1 TO X
6160 PA%(J,1) = PP%(J,NB * 3 - 2)
6190 PA%(J,2) = PP%(J,NB * 3 - 1)
6220 PA%(J,3) = PP%(J,NB * 3)
6250 NEXT J
6280 REM
6310 FOR I = 1 TO X:M = 1
6340 M = 2 * M: IF M < = X THEN
6340
6370 M = INT (M / 2): IF M = 0 THEN
6820
6380 IF X = M THEN 6370
6400 FOR I = 1 TO X - M:C = I
6430 D = C + M: IF PA%(C,2) > =

```

```

        PA%(D,2) THEN 6790
6460 T = PA%(C,2)
6490 U = PA%(C,1)
6520 V = PA%(C,3)
6550 PA%(C,1) = PA%(D,1)
6580 PA%(C,3) = PA%(D,3)
6610 PA%(C,2) = PA%(D,2)
6640 PA%(D,2) = T
6670 PA%(D,1) = U
6700 PA%(D,3) = V
6730 C = C - M
6760 IF C > 0 THEN 6430
6790 NEXT : GOTO 6370
6820 IF LL = 1 THEN LL = 0: RETURN

6850 FOR I = 1 TO X
6880 IF PA%(I,2) = - 9999 THEN
    MM = MM + 1
6910 NEXT I
6940 TMR = X * 2 - 2 - MM
6970 FOR I = 0 TO X - 2
7000 VHR%(I + 1) = TMR - 2 * I
7030 IF VHR%(I + 1) < 0 THEN VHR
    %%(I + 1) = 0
7060 NEXT I
7090 FOR I = X TO XX
7120 VHR%(I) = 0
7150 NEXT I
7180 SUM = VHR%(1)
7210 L = 1
7240 FOR I = 1 TO X - 1
7270 IF PA%(I,2) = - 9999 THEN
    7840
7300 IF PA%(I,2) < > PA%(I + 1,
    2) GOTO 7420
7330 SUM = SUM + VHR%(I + 1)
7360 L = L + 1
7390 GOTO 7720
7420 FOR J = I TO I - L + 1 STEP
    - 1
7450 A%(J,2) = SUM / L
7480 A%(J,1) = PA%(J,1)
7510 C%(J,2) = SUM / L
7540 C%(J,1) = PA%(J,3)
7570 NEXT J
7600 L = 1
7630 SUM = VHR%(I + 1)
7660 IF I = X AND CL = 1 THEN CL
    = 0: RETURN
7690 IF I = X GOTO 8050
7720 NEXT I
7750 IF PA%(X,2) = - 9999 GOTO
    7840
7780 I = X
7810 GOTO 7420
7840 FOR AA = I TO I + MM - 1
7870 A%(AA,2) = PUNTMED
7900 A%(AA,1) = PA%(AA,1)
7930 C%(AA,2) = PUNTMED
7960 C%(AA,1) = PA%(AA,3)

```

```

7990 NEXT AA
8020 MM = 0
8050 GOSUB 5920: REM ASSEGNAZ
    IONE PUNTEGGI
8080 REM SOMMA PUNTEGGI NS
8110 PRINT "*";
8140 NEXT NB
8170 REM ASSEGNA TOTALI
8200 FOR I = 1 TO YY
8230 FOR J = 1 TO N
8260 IF MNS%(I,J) = 7777 THEN KR
    = 0: GOTO 8320
8290 KR = MNS%(I,J)
8320 IF MEO%(I,J) = 7777 THEN KZ
    = 0: GOTO 8380
8350 KZ = MEO%(I,J)
8380 SNS = SNS + KR
8410 SEO = SEO + KZ
8440 NEXT J
8470 MNS%(I,N + 1) = SNS
8500 MEO%(I,N + 1) = SEO
8530 SNS = 0: SEO = 0
8560 NEXT I
8590 RETURN
8620 REM
8650 PRINT : PRINT D$"OPEN DATI"

8680 PRINT D$"READ DATI"
8710 INPUT X: INPUT N: INPUT YY
8740 PRINT D$"CLOSE DATI"
8770 PRINT : PRINT D$;"OPEN TORN
    EO"
8800 PRINT D$;"READ TORNEO"
8830 FOR J = 1 TO N * 3
8860 FOR I = 1 TO X
8890 INPUT PP%(I,J)
8920 NEXT I: NEXT J
8950 PRINT D$;"CLOSE TORNEO"
8980 RETURN

```

## Introduzione

Il C 64, fra tanti aspetti positivi, ha forse il difetto di offrire poche facilitazioni a chi intende sviluppare dei programmi in linguaggio macchina. Il Monitor proposto intende colmare questa lacuna arricchendo il C 64 di una serie di nuove istruzioni.

Il programma, (vedere listato 1) come si può vedere, è composto esclusivamente da linee Data, eccettuata la prima parte che gestisce il trasferimento di tutti i valori in memoria e la ricerca di eventuali errori in fase di digitazione.

Per quanto riguarda gli errori, bisogna fare particolare attenzione alla correttezza del valore negativo presente nella linea successiva ad ogni gruppo di otto linee. Infatti esso rappresenta la somma (cambiata di segno) dei valori del blocco di Data precedente. Il programma

# Monitor 64

## Arricchite il set di di istruzioni del BASIC

di Gianluca Puccio

effettua un controllo confrontando la somma dei dati appena letti con la somma di verifica. Se i due numeri non coincidono il programma si arresta dando una segnalazione di errore, indicando anche il gruppo di linee tra le quali cercare il "bug".

Se di errori non ne sono stati commessi,

la Sys di linea 70 manda in esecuzione il Monitor 64: appare un messaggio di presentazione, vengono stampati i valori di alcuni registri, ed il cursore lampeggia aspettando il primo comando.

A questo punto vi daremo un piccolo consiglio. Il programma, quello con i Data, occupa circa 14 Kbyte, mentre il

### Listato 1 - Il programma Monitor 64.

```
FILE: MONITORLST PAG. 1
10 I=49152:FF=1
20 T=0:FORK=ITOI+127:READP:POKEK,P:T=T+P
30 IFK=52854THENFF=0:K=I+127
40 NEXT
50 READS:IFS+TTHEN80
60 IFFTHENI=I+128:GOTO20
70 SYS49152
80 L=1010+80*INT((I-49152)/128)
90 PRINT"ERRORE NEI DATA DEL GRUPPO DI LI
   NEE :{ 3 SPAZI}"L-"L+70:END
1000 :
1010 DATA32,68,229,169,9,141,32,208,141,3
   3,208,169,128,141,138,2
1020 DATA169,31,160,192,32,30,171,32,95,1
   95,76,116,192,234,234,158
1030 DATA32,32,32,32,32,32,32,32,32,32,32
   ,32,32,32,77,79
1040 DATA78,73,84,79,82,32,32,54,52,13,13
   ,32,32,32,32,32
1050 DATA32,32,32,32,32,32,32,32,32,32,32
   ,32,32,66,89,13
1060 DATA13,32,32,32,32,32,32,32,32,32,32
   ,32,32,71,73,65
1070 DATA78,76,85,67,65,32,80,85,67,67,73
   ,79,13,13,0,234
1080 DATA234,234,234,234,169,0,133,2,169,
   13,32,210,255,32,96,165
1083 :
1085 DATA-9837
1087 :
1090 DATA134,122,132,123,32,115,0,240,244
   ,208,5,234,234,234,234,234
1100 DATA133,251,162,0,189,186,192,208,3,
   76,163,192,197,251,240,3
1110 DATA232,208,241,138,10,170,189,208,1
   92,141,179,192,189,209,192,141
```

```
1120 DATA180,192,32,116,164,76,116,192,23
   4,234,77,94,64,82,46,70
1130 DATA84,67,72,71,33,83,76,86,36,35,61
   ,62,68,65,0,0
1140 DATA248,193,139,194,243,194,95,195,1
   53,195,218,195,2,196,194,196
1150 DATA50,196,60,198,116,164,131,197,19
   6,197,229,197,101,198,146,198
1160 DATA49,200,65,199,146,200,128,204,13
   3,193,234,234,234,234,234,72
1163 :
1165 DATA-18853
1167 :
1170 DATA74,74,74,74,170,189,22,193,32,21
   0,255,104,41,15,170,189
1180 DATA22,193,32,210,255,96,48,49,50,51
   ,52,53,54,55,56,57
1190 DATA65,66,67,68,69,70,234,234,234,23
   4,234,162,4,208,2,162
1200 DATA2,200,185,0,2,201,32,208,24,200,
   185,0,2,201,48,144
1210 DATA16,201,58,144,8,201,65,144,8,201
   ,71,176,4,202,208,233
1220 DATA96,198,2,208,251,234,234,234,234
   ,234,162,2,208,2,162,1
1230 DATA32,115,0,144,2,233,8,233,47,10,1
   0,10,10,149,250,32
1240 DATA115,0,144,2,233,8,233,47,21,250,
   149,250,202,208,225,96
1243 :
1245 DATA-14836
1247 :
1250 DATA234,234,234,234,234,169,143,160,
   193,32,30,171,96,234,234,42
1260 DATA32,69,82,82,79,82,69,32,42,0,160
   ,0,32,43,193,32
1270 DATA43,193,32,47,193,198,250,240,5,1
   62,2,32,57,193,185,1
```



## COMMODORE 64

## Seguito programma Monitor 64.

1280 DATA2,240,2,133,2,165,2,208,34,32,90,  
193,165,251,133,253  
1290 DATA165,252,133,254,32,90,193,165,25  
1,133,20,165,252,133,21,198  
1300 DATA250,208,5,32,90,193,240,3,32,94,  
193,96,234,234,234,234  
1310 DATA234,165,254,197,21,144,10,208,9,  
165,253,197,20,144,2,208  
1320 DATA1,202,96,234,234,234,234,234,160  
,0,32,43,193,200,185,0  
1323 :  
1325 DATA-16783  
1327 :  
1330 DATA2,208,4,165,2,240,3,76,133,193,3  
2,90,193,165,251,41  
1340 DATA248,133,251,165,252,32,255,192,1  
65,251,32,255,192,169,46,160  
1350 DATA8,153,122,194,136,208,250,169,32  
,32,210,255,160,0,169,32  
1360 DATA32,210,255,177,251,201,32,144,11  
,201,128,144,4,201,160,144  
1370 DATA3,153,123,194,32,255,192,200,192  
,8,208,226,169,32,32,210  
1380 DATA255,169,58,32,210,255,169,123,16  
0,194,32,30,171,169,13,32  
1390 DATA210,255,24,169,8,101,251,133,251  
,208,2,230,252,165,198,240  
1400 DATA252,32,228,255,201,13,240,155,96  
,234,234,234,234,234,46,255  
1403 :  
1405 DATA-18962  
1407 :  
1410 DATA255,46,46,0,234,234,234,234,234,  
234,234,160,0,32,43,193  
1420 DATA185,1,2,208,1,96,32,47,193,165,2  
,240,3,76,133,193  
1430 DATA185,1,2,208,241,32,90,193,165,25  
1,133,253,165,252,133,254  
1440 DATA160,0,32,94,193,165,251,145,253,  
230,253,208,2,230,254,200  
1450 DATA177,122,208,236,169,94,32,210,25  
5,169,32,32,210,255,165,254  
1460 DATA32,255,192,165,253,32,255,192,16  
9,32,32,210,255,169,145,141  
1470 DATA119,2,169,17,141,120,2,169,2,133  
,198,104,104,76,125,192  
1480 DATA234,234,234,160,0,32,43,193,165,  
2,208,8,200,185,0,2  
1483 :  
1485 DATA-17940  
1487 :  
1490 DATA201,32,240,3,76,133,193,200,185,  
0,2,201,39,208,245,32  
1500 DATA90,193,185,1,2,208,3,96,234,234,  
200,185,0,2,240,11  
1510 DATA129,251,230,251,208,244,230,252,  
76,26,195,169,64,32,210,255  
1520 DATA169,32,32,210,255,165,252,32,255  
,192,165,251,32,255,192,169  
1530 DATA32,32,210,255,169,39,32,210,255,  
169,145,141,119,2,169,17  
1540 DATA141,120,2,169,2,133,198,104,104,  
76,125,192,234,234,234,169  
1550 DATA133,160,195,32,30,171,169,46,32,  
210,255,160,0,169,32,32

1560 DATA210,255,185,12,3,32,255,192,200,  
192,4,208,240,169,13,32  
1563 :  
1565 DATA-17615  
1567 :  
1570 DATA210,255,96,234,234,32,32,65,67,3  
2,88,82,32,89,82,32  
1580 DATA83,82,13,0,234,234,234,234,1  
69,4,133,251,160,0,32  
1590 DATA47,193,198,251,208,249,185,1,2,2  
08,4,165,2,240,3,76  
1600 DATA133,193,160,0,32,94,193,165,251,  
153,12,3,200,192,4,208  
1610 DATA243,96,234,234,0,0,0,0,0,0,0,0,  
0,0,0  
1620 DATA0,0,0,0,0,0,0,0,0,0,169,1,133,25  
0,32,154  
1630 DATA193,165,2,240,3,76,133,193,32,22  
5,193,232,208,12,165,251  
1640 DATA129,253,230,253,208,242,230,254,  
208,238,96,234,234,234,234,234  
1643 :  
1645 DATA-15164  
1647 :  
1650 DATA234,234,169,2,133,250,32,154,193  
,165,2,240,3,76,133,193  
1660 DATA32,225,193,232,208,20,161,253,12  
9,251,230,253,208,4,230,254  
1670 DATA240,8,230,251,208,234,230,252,20  
8,230,96,234,234,234,234,234  
1680 DATA234,234,160,0,140,160,196,32,43,  
193,32,43,193,238,160,196  
1690 DATA32,47,193,185,1,2,208,245,165,2,  
240,3,76,133,193,133  
1700 DATA250,32,185,193,141,161,196,160,1  
,204,160,196,240,9,32,94  
1710 DATA193,153,161,196,200,208,242,32,2  
25,193,232,208,48,160,0,132  
1720 DATA250,177,253,217,161,196,240,2,19  
8,250,200,204,160,196,208,241  
1723 :  
1725 DATA-20555  
1727 :  
1730 DATA165,250,208,17,169,32,32,210,255  
,165,254,32,255,192,165,253  
1740 DATA32,255,192,162,0,230,253,208,206  
,230,254,208,202,96,234,234  
1750 DATA14,77,79,78,73,84,79,82,32,54,52  
,32,46,76,77,77  
1760 DATA66,76,0,0,0,0,0,0,0,0,0,0,234,  
234,234  
1770 DATA234,234,169,2,133,250,32,154,193  
,165,2,240,3,76,133,193  
1780 DATA32,225,193,232,208,39,161,253,19  
3,251,240,17,169,32,32,210  
1790 DATA255,165,254,32,255,192,165,253,3  
2,255,192,162,0,230,253,208  
1800 DATA4,230,254,240,8,230,251,208,215,  
230,252,208,211,96,234,234  
1803 :  
1805 DATA-18214  
1807 :  
1810 DATA234,234,234,234,234,173,1,2,201,  
32,208,66,173,2,2,201  
1820 DATA39,208,59,160,2,162,0,200,185,0,  
2,201,39,240,10,157

**Seguito programma Monitor 64.**

1830 DATA161,196,232,224,17,208,240,240,3  
7,142,160,196,200,185,0,2

1840 DATA201,32,208,21,200,185,0,2,201,48  
,208,13,200,185,0,2

1850 DATA201,49,240,9,201,56,208,6,174,16  
0,196,240,1,96,198,2

1860 DATA208,251,234,234,234,234,234,169,  
15,166,250,160,1,32,186,255

1870 DATA173,160,196,162,161,160,196,32,1  
89,255,165,10,48,8,162,0

1880 DATA160,0,32,213,255,96,166,251,164,  
252,32,216,255,96,234,234

1883 :  
1885 DATA-17674  
1887 :  
1890 DATA234,234,234,32,5,197,32,43,193,3  
2,43,193,185,1,2,208

1900 DATA4,165,2,240,3,76,133,193,32,115,  
0,32,115,0,201,39

1910 DATA208,249,32,94,193,133,250,32,90,  
193,165,251,133,253,165,252

1920 DATA133,254,32,90,193,169,253,133,10  
,32,87,197,96,234,234,234

1930 DATA234,234,234,234,32,5,197,56,233,  
48,133,250,185,1,2,208

1940 DATA4,165,2,240,3,76,133,193,133,10,  
32,87,197,96,234,234

1950 DATA234,234,234,234,234,32,5,197,56,  
233,48,133,250,185,1,2

1960 DATA208,4,165,2,240,3,76,133,193,169  
,1,133,10,32,87,197

1963 :  
1965 DATA-16382  
1967 :  
1970 DATA165,144,41,16,240,10,169,23,160,  
198,32,30,171,96,169,44

1980 DATA160,198,24,144,245,234,234,13,42  
42,32,86,69,82,73,70

1990 DATA89,32,69,82,82,79,82,32,42,42,13  
,0,13,42,42,32

2000 DATA79,75,32,42,42,13,0,234,234,234,  
234,234,160,0,32,43

2010 DATA193,185,1,2,208,4,165,2,240,3,76  
,133,193,32,90,193

2020 DATA165,251,141,91,198,165,252,141,9  
2,198,32,0,207,96,234,234

2030 DATA234,234,234,234,234,160,0,32,43,  
193,185,1,2,208,4,165

2040 DATA2,240,3,76,133,193,32,90,193,169  
,35,32,210,255,169,32

2043 :  
2045 DATA-14161  
2047 :  
2050 DATA32,210,255,165,252,166,251,32,20  
5,189,96,234,234,234,234,234

2060 DATA234,234,169,1,133,250,173,1,2,20  
1,32,240,3,76,133,193

2070 DATA160,0,185,2,2,201,48,144,244,201  
,58,176,240,234,166,250

2080 DATA240,12,234,217,22,199,144,4,208,  
227,240,2,198,250,200,192

2090 DATA5,208,223,185,2,2,208,213,133,25  
1,133,252,32,115,0,240

2100 DATA46,56,233,48,72,6,251,38,252,166  
,251,164,252,6,251,38

2110 DATA252,6,251,38,252,24,138,101,251,  
133,251,152,101,252,133,252

2120 DATA104,24,101,251,133,251,169,0,101  
,252,133,252,24,144,205,169

2123 :  
2125 DATA-19230  
2127 :  
2130 DATA36,32,210,255,169,32,32,210,255,  
165,252,32,255,192,165,251

2140 DATA32,255,192,96,234,234,54,53,53,5  
1,53,234,234,234,234,234

2150 DATA234,169,1,162,2,133,187,134,188,  
132,183,32,213,243,165,186

2160 DATA32,180,255,165,185,32,150,255,96  
,234,234,234,234,234,234,234

2170 DATA234,169,8,133,186,173,1,2,208,3,  
76,116,199,201,36,208

2180 DATA3,76,145,199,201,61,208,3,76,244  
,199,160,0,200,185,1

2190 DATA2,208,250,169,111,133,185,32,33,  
199,76,116,199,234,234,234

2200 DATA234,234,234,234,160,0,169,111,13  
3,185,32,33,199,32,165,255

2203 :  
2205 DATA-19111  
2207 :  
2210 DATA32,210,255,201,13,208,246,32,66,  
246,96,234,234,234,234,234

2220 DATA234,160,0,200,185,1,2,208,250,13  
3,144,169,96,133,185,32

2230 DATA33,199,160,3,132,250,32,165,255,  
170,32,165,255,164,144,208

2240 DATA52,198,250,208,241,32,205,189,16  
9,32,32,210,255,32,165,255

2250 DATA166,144,208,33,201,0,240,20,32,2  
10,255,32,225,255,240,21

2260 DATA32,228,255,240,232,32,228,255,24  
0,251,208,225,169,13,32,210

2270 DATA255,160,2,208,191,32,66,246,96,2  
34,234,234,234,234,234,234

2280 DATA234,234,234,234,160,255,200,185,  
2,2,208,250,133,144,152,208

2283 :  
2285 DATA-20660  
2287 :  
2290 DATA3,76,133,193,169,96,133,185,169,  
2,32,35,199,32,165,255

2300 DATA164,144,208,22,170,169,35,32,210  
,255,169,32,32,210,255,32

2310 DATA165,255,32,205,189,169,13,32,210  
,255,32,66,246,96,234,234

2320 DATA234,160,255,200,185,1,2,208,250,  
132,251,169,0,141,60,3

2330 DATA169,2,162,1,160,0,32,186,255,165  
,251,162,1,160,2,32

2340 DATA189,255,32,192,255,169,2,32,195,  
255,173,60,3,201,1,240

2350 DATA4,201,3,208,36,160,0,132,250,169  
,13,32,210,255,169,35

2360 DATA32,210,255,169,32,32,210,255,190  
,61,3,185,62,3,32,205

2363 :  
2365 DATA-16507

## COMMODORE 64

Monitor 64 vero e proprio occupa meno di 4 Kbyte (da 49.152 a 52.854).

Quindi perchè non registrare direttamente il secondo? Tanto più che oltre al tempo di caricamento, il primo ha bisogno di un altro minuto circa per la lettura dei Data.

Come registrare il Monitor 64? Facile, basta usare il suo stesso comando di registrazione!

Su disco: S "Monitor 64" 08 C000 CE77.

Su nastro: idem, sostituendo 08 con 01

Nel caso si usi il disco, ricordarsi in seguito di caricarlo con ",8,1".

### Spiegazione comandi

Tutti i numeri da digitare e tutti quelli visualizzati dal Monitor sono esadecimali. Gli indirizzi sono sempre formati da 4 caratteri, quindi bisogna scrivere 0400 e non 400.

Analogamente i valori da 1 byte sono sempre formati da 2 caratteri.

Tutti i comandi hanno un solo carattere (seguito eventualmente da dati). Attenzione a digitare correttamente anche gli spazi, come indicato dagli esempi. Comunque basta ricordarsi che dopo il carattere del comando segue sempre uno spazio, ed anche tutti i dati (se ce ne sono, ovviamente) sono separati tra loro con uno spazio.

Se si sbaglia qualcosa appare subito un messaggio d'errore.

*M (Memory) es.: M 0801*

Visualizza la memoria partendo dall'indirizzo digitato, o meglio dal multiplo di 8 più vicino ad esso. Nell'esempio partirebbe da 0800.

In una riga compaiono: un indirizzo, il suo contenuto e i contenuti di 7 locazioni successive, e una stringa composta

dal Chr\$ di questi otto valori. Se un carattere non è visualizzabile, ad esempio Home, al suo posto compare un punto.

Ogni volta che si preme Return viene visualizzata un'altra linea (altri 8 byte); tenendolo premuto si ottiene una visualizzazione rapidissima. Un qualsiasi altro tasto interrompe la lettura.

↑ (Poke) es.: ↑ D020 01 00

Effettua la Poke dei valori presenti, partendo dall'indirizzo digitato. Nell'esempio, porrebbe 01 in D020 e 00 in D021.

I valori possono essere tanti quanti ce ne stanno in due linee di schermo. Dopo la Poke appare automaticamente l'indirizzo successivo all'ultimo usato, e il cursore alla sua destra è pronto ad accettare il prossimo valore.

Si può anche tornare sulle linee precedenti e modificarle. Per terminare basta

#### Seguito programma Monitor 64.

```

2367 :
2370 DATA189,164,250,200,200,192,4,208,22
      2,169,13,32,210,255,96,234
2380 DATA234,234,160,0,32,43,193,185,1,2,
      208,4,165,2,240,3
2390 DATA76,133,193,32,90,193,165,252,32,
      255,192,165,251,32,255,192
2400 DATA169,32,32,210,255,169,32,32,210,
      255,160,0,177,251,133,253
2410 DATA32,255,192,230,251,208,2,230,252,
      166,253,189,202,202,133,250
2420 DATA240,42,169,32,32,210,255,177,251,
      133,254,32,255,192,230,251
2430 DATA208,2,230,252,165,250,201,8,144,
      18,169,32,32,210,255,177
2440 DATA251,133,255,32,255,192,230,251,2
      08,2,230,252,24,166,214,160
2443 :
2445 DATA-20232
2447 :
2450 DATA16,32,240,255,164,253,185,202,20
      1,168,162,3,185,202,203,32
2460 DATA210,255,200,202,208,246,234,234,
      234,234,234,234,234,234,169,32
2470 DATA32,210,255,164,250,240,97,192,2,
      240,115,192,6,144,9,192
2480 DATA9,176,5,169,40,32,210,255,192,1,
      208,5,169,35,32,210
2490 DATA255,192,8,144,5,165,255,32,255,1
      92,165,254,32,255,192,192
2500 DATA4,240,8,192,6,240,4,192,9,208,10
      ,169,44,32,210,255
2510 DATA169,88,32,210,255,192,6,144,9,19
      2,9,176,5,169,41,32
2520 DATA210,255,192,5,240,8,192,7,240,4,

```

```

      192,10,208,10,169,44
2523 :
2525 DATA-18116
2527 :
2530 DATA32,210,255,169,89,32,210,255,169
      ,13,32,210,255,165,198,240
2540 DATA252,32,228,255,201,13,208,3,76,1
      66,200,96,234,234,165,254
2550 DATA8,24,101,251,133,254,165,252,105
      ,0,133,255,40,16,150,198
2560 DATA255,24,144,145,234,234,0,0,0,0,0
      ,0,0,0,0,0
2570 DATA0,0,0,0,0,0,0,0,0,0,0,33,105,0,0,0
      ,105
2580 DATA9,0,111,105,9,0,0,105,9,0,30,105
      ,0,0,0,105
2590 DATA9,0,42,105,0,0,0,105,9,0,87,6,0,
      0,21,6
2600 DATA120,0,117,6,120,0,21,6,120,0,24,
      6,0,0,0,6
2603 :
2605 DATA-10039
2607 :
2610 DATA120,0,135,6,0,0,0,6,120,0,126,72
      ,0,0,0,72
2620 DATA99,0,108,72,99,0,84,72,99,0,36,7
      2,0,0,0,72
2630 DATA99,0,48,72,0,0,0,72,99,0,129,3,0
      ,0,0,3
2640 DATA123,0,114,3,123,0,84,3,123,0,39,
      3,0,0,0,3
2650 DATA123,0,141,3,0,0,0,3,123,0,0,144,
      0,0,150,144
2660 DATA147,0,69,0,162,0,150,144,147,0,1
      2,144,0,0,150,144
2670 DATA147,0,168,144,165,0,0,144,0,0,96
      ,90,93,0,96,90
2680 DATA93,0,156,90,153,0,96,90,93,0,15,

```

Seguito programma Monitor 64.

90,0,0,96,90  
 2683 :  
 2685 DATA-7008  
 2687 :  
 2690 DATA93,0,51,90,159,0,96,90,93,0,60,54,0,0,0,60,54  
 2700 DATA63,0,81,54,66,0,60,54,63,0,27,54,0,0,0,54  
 2710 DATA63,0,45,54,0,0,0,54,63,0,57,132,0,0,57,132  
 2720 DATA75,0,78,132,102,0,57,132,75,0,18,132,0,0,0,132  
 2730 DATA75,0,138,132,0,0,0,132,75,0,0,6,0,0,0,3  
 2740 DATA3,0,0,1,0,0,0,11,11,0,2,7,0,0,0,4  
 2750 DATA4,0,0,10,0,0,0,9,9,0,11,6,0,0,3,3  
 2760 DATA3,0,0,1,0,0,11,11,11,0,2,7,0,0,0,4  
 2763 :  
 2765 DATA-3771  
 2767 :  
 2770 DATA4,0,0,10,0,0,0,9,9,0,0,6,0,0,0,3  
 2780 DATA3,0,0,1,0,0,11,11,11,0,2,7,0,0,0,4  
 2790 DATA4,0,0,10,0,0,0,9,9,0,0,6,0,0,0,3  
 2800 DATA3,0,0,1,0,0,8,11,11,0,2,7,0,0,0,4  
 2810 DATA4,0,0,10,0,0,0,9,9,0,0,6,0,0,3,3  
 2820 DATA3,0,0,0,0,0,11,11,11,0,2,7,0,0,4,4  
 2830 DATA5,0,0,10,0,0,0,9,0,0,1,6,1,0,3,3  
 2840 DATA3,0,0,1,0,0,11,11,11,0,2,7,0,0,4,4  
 2843 :  
 2845 DATA-368  
 2847 :  
 2850 DATA5,0,0,10,0,0,9,9,10,0,1,6,0,0,3,3  
 2860 DATA3,0,0,1,0,0,11,11,11,0,2,7,0,0,0,4  
 2870 DATA4,0,0,10,0,0,0,9,9,0,1,6,0,0,3,3  
 2880 DATA3,0,0,1,0,0,11,11,11,0,2,7,0,0,0,4  
 2890 DATA4,0,0,10,0,0,0,9,9,0,45,45,45,65,68,67  
 2900 DATA65,78,68,65,83,76,66,67,67,66,67,83,66,69,81,66  
 2910 DATA73,84,66,77,73,66,78,69,66,80,76,66,82,75,66,86  
 2920 DATA67,66,86,83,67,76,67,67,76,68,67,76,73,67,76,86  
 2923 :  
 2925 DATA-4052  
 2927 :  
 2930 DATA67,77,80,67,80,88,67,80,89,68,69,67,68,69,88,68

2940 DATA69,89,69,79,82,73,78,67,73,78,88,73,78,89,74,77  
 2950 DATA80,74,83,82,76,68,65,76,68,88,76,68,89,76,83,82  
 2960 DATA78,79,80,79,82,65,80,72,65,80,72,80,80,76,65,80  
 2970 DATA76,80,82,79,76,82,79,82,82,84,73,82,84,83,83,66  
 2980 DATA67,83,69,67,83,69,68,83,69,73,83,84,65,83,84,88  
 2990 DATA83,84,89,84,65,88,84,65,89,84,83,88,84,88,65,84  
 3000 DATA88,83,84,89,65,0,0,0,0,0,0,0,0,0,0,0  
 3003 :  
 3005 DATA-9082  
 3007 :  
 3010 DATA160,0,32,43,193,165,2,208,98,185,1,2,208,1,96,201  
 3020 DATA32,208,88,32,90,193,165,251,133,253,165,252,133,254,162,3  
 3030 DATA185,2,2,221,202,203,208,20,185,3,2,221,203,203,208,12  
 3040 DATA185,4,2,221,204,203,208,4,134,20,240,9,232,232,232,224  
 3050 DATA171,208,221,240,38,185,5,2,208,3,76,213,205,201,32,208  
 3060 DATA26,152,24,105,6,168,185,0,2,201,35,208,22,162,2,32  
 3070 DATA57,193,185,1,2,208,4,165,2,240,3,76,133,193,169,1  
 3080 DATA76,213,205,201,40,208,95,162,2,3,2,57,193,165,2,208,235  
 3083 :  
 3085 DATA-15715  
 3087 :  
 3090 DATA200,185,0,2,201,44,208,24,185,1,2,201,88,208,220,185  
 3100 DATA2,2,201,41,208,213,185,3,2,208,2,08,169,6,76,213,205  
 3110 DATA201,41,208,24,185,1,2,201,44,208,192,185,2,2,201,89  
 3120 DATA208,185,185,3,2,208,180,169,7,76,213,205,162,2,32,58  
 3130 DATA193,165,2,208,166,185,1,2,201,41,208,159,185,2,2,208  
 3140 DATA154,169,8,76,213,205,162,2,32,58,193,165,2,208,113,185  
 3150 DATA1,2,208,5,169,3,76,213,205,201,4,4,208,31,185,2,2  
 3160 DATA201,88,208,10,185,3,2,208,87,169,4,76,213,205,201,89  
 3163 :  
 3165 DATA-14954  
 3167 :  
 3170 DATA208,78,185,3,2,208,73,169,5,76,2,13,205,162,2,32,57  
 3180 DATA193,165,2,208,59,185,1,2,208,24,173,7,2,201,66,208  
 3190 DATA12,173,8,2,201,73,240,5,169,2,76,213,205,169,11,76  
 3200 DATA213,205,201,44,208,26,185,2,2,201,88,208,10,185,3,2  
 3210 DATA208,14,169,9,76,213,205,201,89,2,08,5,185,3,2,240,3

## COMMODORE 64

dare Return quando ci si trova su una linea in cui è presente solo l'indirizzo. Oppure si può cambiare linea di schermo e digitare un altro comando.

@ (Poke ASCII) es.: @ 09F0' jolly

Scrivere in memoria, partendo dall'indirizzo digitato, i codici ASCII corrispondenti ai caratteri presenti dopo l'apice. Nell'esempio porrebbe 4A (il valore ASCII della J) in 09F0, 4F (codice della O) in 09F1, ecc. Finita l'operazione, il Monitor stampa l'indirizzo successivo all'ultimo utilizzato, stampa anche l'apice, ed attende altri caratteri. Il numero di caratteri digitabili è limitato alle solite due linee di schermo.

Per terminare o per modificare linee precedenti valgono le stesse considerazioni fatte in merito al comando Poke.

R (Reg)

Visualizza il valore dei seguenti registri: AC (Accumulatore), XR (X Register), YR (Y Register), SR (Status Register). Non è necessario alcun parametro.

(Poke reg) es.: . 00 07 05 00

Permette di modificare il contenuto dei registri.

F (Fill) es.: F 0400 07E7 51

Pone il valore digitato in tutte le locazioni che vanno dal 1° al 2° indirizzo compresi.

T (Transfer) es.: T 0400 07E7 1000

Copia il contenuto della zona di memoria che va dal 1° al 2° indirizzo, nella zona di memoria che inizia dal 3° indirizzo.

C (Compare) es.: C 1300 1C00 8F00

Confronta il contenuto di tutte le loca-

zioni che vanno dal 1° al 2° indirizzo, con le locazioni che partono dal 3° indirizzo. In caso il Monitor incontri delle discordanze, ne stampa le locazioni. Se le due zone di memoria sono identiche non stampa nulla.

H (Hunt) es.: H E000 FFFF 20 D2 FF

Cerca in memoria tra il 1° e il 2° indirizzo tutti i gruppi di valori uguali a quello digitato. Ogni volta che ne trova uno, ne stampa l'indirizzo. Nell'esempio verrebbe visualizzato, tra gli altri, l'indirizzo F5C9 perchè nelle locazioni F5C9, F5CA e F5CB sono contenuti rispettivamente i valori 20, D2 e FF.

Come per altri comandi precedenti, si possono digitare tanti valori quanti ce ne stanno su due linee di schermo.

G (Go) es.: G 1000

Esegue il codice macchina iniziando

#### Seguito programma Monitor 64.

```

3220 DATA76,133,193,169,10,133,21,240,83,
      32,115,0,32,115,0,32
3230 DATA115,0,173,11,2,201,48,16,3,32,11
      5,0,165,21,201,8
3240 DATA16,9,201,2,240,10,32,94,193,240,
      49,32,90,193,240,44
3243 :
3245 DATA-12810
3247 :
3250 DATA32,90,193,165,251,229,253,133,25
      1,165,252,229,254,133,252,56
3260 DATA165,251,233,2,133,251,176,2,198,
      252,230,252,208,6,165,251
3270 DATA16,174,48,8,198,252,208,168,165,
      251,48,164,162,0,165,20
3280 DATA221,202,201,208,9,165,21,221,202
      ,202,208,2,240,5,232,208
3290 DATA237,240,141,138,162,0,129,253,23
      0,253,208,2,230,254,165,21
3300 DATA240,26,165,251,129,253,230,253,2
      08,2,230,254,165,21,201,8
3310 DATA48,10,165,252,129,253,230,253,20
      8,2,230,254,169,65,76,198
3320 DATA194,76,201,194,234,234,234
3323 :
3325 DATA-19515
  
```



## Dalla grande edicola Jackson

### Tutto sul personal computer

#### COMPUSCUOLA

La rivista di informatica nella didattica per la scuola italiana.  
9 numeri all'anno: L. 2.000 a numero  
Abbonamento: solo L. 15.000

#### Personal

L'unica rivista indipendente per gli utenti dei personal computer Olivetti.  
10 numeri all'anno: L. 4.000 a numero  
Abbonamento: solo L. 35.000



L'unica rivista italiana dedicata ai sistemi MS-DOS, Personal computer IBM e compatibili.  
10 numeri all'anno: L. 5.000 a numero  
Abbonamento: solo L. 40.000

#### PERSONAL SOFTWARE

Aspetti e problemi del software per personal computer, programmi, giochi e sistemi operativi.  
11 numeri all'anno: L. 4.000 a numero  
Abbonamento: solo L. 34.000



La prima rivista europea di personal computer, software e accessori. Con test, novità, analisi del mercato...  
11 numeri all'anno: L. 5.000 a numero  
Abbonamento: solo L. 43.000

#### Quando l'informazione fa testo

In busta chiusa inviate questo coupon a:  
Gruppo Editoriale Jackson  
via Rosellini, 12 - 20124 MI

Desidero ricevere GRATIS un numero  
della Rivista \_\_\_\_\_

(allego L. 1.000 in francobolli per  
contributo spese di spedizione)

Inviatemi GRATIS il Catalogo della  
Biblioteca JACKSON (allego L. 1.000  
in francobolli per contributo spese di  
spedizione)

Nome \_\_\_\_\_

Cognome \_\_\_\_\_

via \_\_\_\_\_

CAP \_\_\_\_\_ Città \_\_\_\_\_

dall'indirizzo digitato. Il controllo torna al Monitor quando il computer incontra un'istruzione Rts.

*!* (Quit)

Termina l'esecuzione del Monitor e restituisce il controllo al BASIC. Per ritornare al Monitor battere: Sys 49152.

*S* (Save) es.: *S 'prova' 08 9000 915E*

Registra su disco (08) o su nastro (01) la routine in linguaggio macchina che va dal 1° indirizzo compreso al 2° indirizzo escluso. Nell'esempio si voleva salvare da 9000 a 9150 compresi. Quindi il 2° indirizzo deve essere sempre aumentato di 1.

Ovviamente il comando registra non solo programmi ma in generale porzioni di memoria qualsiasi, ad esempio la pagina video.

Nel caso si usi l'unità a nastri, può essere omesso il nome del programma ma va digitata comunque la coppia di apici. Esempio: *S " 01 0400 07E7*

*L* (Load) es.: *L 'prova' 08*

Carica un programma da disco (08) o da nastro (01), allocandolo nello stesso posto nel quale si trovava quando è stato registrato. Con il nastro si può omettere il nome.

*V* (Verify) es.: *V 'prova' 08*

Verifica un programma precedentemente registrato.

*\$* (Esa-Dec) es.: *\$ CE77*

Trasforma il numero esadecimale digitato in un numero decimale e lo stampa. Dopo il comando devono esserci sempre 4 cifre esadecimali. Quindi bisogna digitare 007A e non 7A.

*#* (Dec-Esa) es.: *# 32768*

Trasforma il numero decimale digitato nell'equivalente esadecimale. Il numero deve essere comunque composto da 5 cifre: va scritto 01024 e non 1024.

*=* (Tape range) es.: *= prog*

Permette di conoscere l'indirizzo iniziale e l'indirizzo finale della zona di memoria che occuperebbe un programma, attualmente su nastro.

Non va digitato alcuno spazio tra il comando ed il nome (a meno che il nome stesso non inizi con uno spazio).

*>* (Drive)

Comprende quattro comandi. Per tutti non va digitato lo spazio dopo il segno *>*.

Comando disco es.: *> S:game*

Manda al drive tutto quanto compare dopo il simbolo quindi tutti i comandi per il disco vengono accettati. Nell'esempio, il comando S (Scratch) cancellerebbe il programma "game".

Directory es.: *> \$*

Visualizza la directory del disco presente nel drive. Anche in questo caso viene "spedito" tutto quanto segue *>*, quindi è possibile la sintassi estesa, della forma: *> \$:progr★*.

Si può fermare temporaneamente la visualizzazione premendo un tasto qualsiasi, poi ripartire sempre con un tasto qualsiasi. Si può anche sfruttare il tasto Stop per interrompere definitivamente la lettura.

Stato

Il solo simbolo *>* permette di conoscere lo stato del drive.

Ind. d'inizio es.: *> = nome*

Visualizza l'indirizzo d'inizio di un programma residente su disco.

D (Disassemble) es.: *D C000*

Disassembla la memoria partendo dall'indirizzo digitato. In una linea compaiono: un indirizzo, poi 1,2 o 3 byte di codice macchina (secondo il tipo di indirizzamento), infine il disassemblato di quei byte.

Se il contenuto di una locazione di memoria non corrisponde a nessuna delle istruzioni del 6510, al posto dell'istruzione mnemonica apparirà la stringa "----". Ogni volta che si preme Return viene disassemblata un'altra istruzione. Qualsiasi tasto per terminare.

A (Assemble) es.: *A 9500 LDA # 51*

Assembla l'istruzione digitata dopo l'indirizzo. Appare poi automaticamente l'indirizzo per l'istruzione successiva. Digitare Return per interrompere. Si possono modificare le linee precedenti allo stesso modo che per il comando Poke.

Nelle istruzioni di salto relativo (Branch) deve comparire l'indirizzo dove salte-

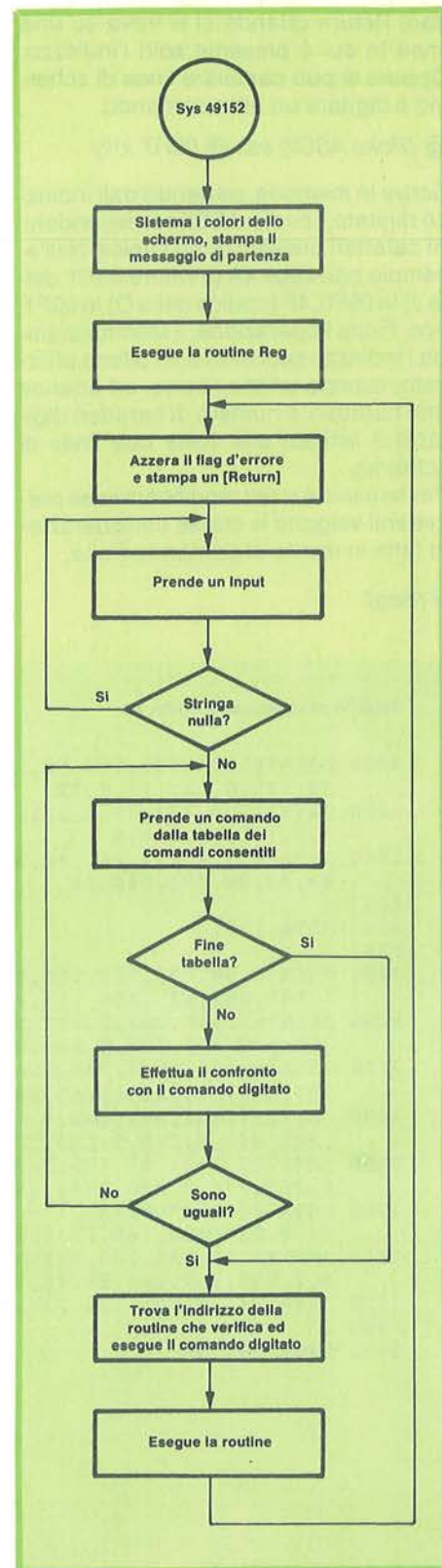


Figura 1 - Schema a blocchi del programma.

## COMMODORE 64

rebbe il programma in caso di branch verificato. Esempio BNE 9C4F. Attenzione a non superare i confini del range permesso, pena messaggio d'errore.

### Commento alle routine

Ecco lo schema di funzionamento di Monitor 64, come è riportato nello schema a blocchi di figura 1 e nel disassemblato di figura 2 riportato a fine articolo.

#### Inizializzazione (\$C0-\$C01A)

Esegue la subroutine \$E544 che pulisce lo schermo. Colora di verde il bordo e lo sfondo. Dà il Repeat automatico a tutti i tasti. Esegue la subroutine \$AB1E per stampare il messaggio di partenza. Prima del salto a tale subroutine, l'accumulatore e il registro X devono contenere rispettivamente la parte bassa e la parte alta dell'indirizzo d'inizio della stringa da stampare.

Alla fine della stringa deve esserci un byte uguale a zero. Poi esegue la subroutine Reg del Monitor per visualizzare il contenuto di alcuni registri. Infine salta a Start (\$C074) in attesa di un comando.

#### Start (\$C074-\$C089)

Azzerare la locazione 2 che rappresenta il flag d'errore; viene esaminato al ritorno di varie subroutine di verifica per i dati digitati.

Stampa un Return ed esegue la subroutine \$A560, la quale prende un Input e lo memorizza da \$0200 in poi, aggiungendo uno zero alla fine. Al ritorno i registri X e Y rappresentano rispettivamente il byte basso e il byte alto dell'indirizzo in cui è posto il primo dei caratteri digitati. Questo indirizzo viene memorizzato nelle locazioni \$7A e \$7B, sfruttate successivamente dalla subroutine \$0073 che preleva un carattere dalla memoria.

Se il primo carattere è un Chr\$(0), allora l'Input era una stringa nulla, e il controllo torna di nuovo all'istruzione Jsr \$A560. In caso contrario salta al controllo comandi.

#### Controllo comandi (\$C090-\$C0B5)

Prende un carattere dalla tabella dei comandi permessi (tabella 1), e lo confronta con il comando digitato. Se coincidono allora il comando è lecito e termina il ciclo per i confronti. L'indice alla tabella viene moltiplicato per 2 ed usato,

### COMANDI

M	Memory	C1F8
↑	Poke	C28B
@	Poke ASCII	C2F3
R	Reg	C35F
.	Poke reg	C399
F	Fill	C3DA
T	Transfer	C402
C	Compare	C4C2
H	Hunt	C432
G	Go	C63C
!	Quit	A474
S	Save	C583
L	Load	C5C4
V	Verify	C5E5
\$	Esa-Dec	C665
#	Dec-Esa	C692
=	Tape range	C831
>	Drive	C741
D	Disassemble	C892
A	Assemble	CC80
> com.	Comando	C75B
> \$	Directory	C791
>	Stato	C774
> =	Ind. inizio	C7F4

Tabella 1 - Indirizzi dei comandi riconosciuti dal programma.

ancora come indice, per prendere dall'apposita tabella l'indirizzo della routine per l'esecuzione del comando. Tale indirizzo viene memorizzato direttamente nel vettore di salto alla routine.

Se il comando non fosse stato riconosciuto, l'indice alla tabella degli indirizzi punterebbe all'ultima coppia di valori: l'indirizzo della routine di errore.

Per quanto riguarda il comando Quit (!), nella tabella compare l'indirizzo \$A474 dell'interprete BASIC. Si tratta dell'ultima parte della routine di gestione degli errori, ed il suo scopo è la stampa del consueto Ready: quanto basta per uscire dal Monitor.

Al termine della routine di controllo comandi troviamo una istruzione di salto a Start, dove tutto ricomincia con un nuovo Input.

#### Subroutine per la stampa di un byte (\$C0FF-\$C115)

Ottenere 2 cifre esadecimali partendo da un valore (0-255) è abbastanza semplice. Per la prima cifra basta prendere il nibble alto del valore (i suoi 4 bit più significativi) ed usarlo come indice alla

tabella che contiene il codice ASCII di tutte le cifre esadecimali. Abbiamo così ricavato il primo carattere. Utilizzando il nibble basso del valore iniziale (i 4 bit meno significativi) come indice alla tabella appena menzionata, si ricava il secondo carattere.

Facciamo un esempio: trasformiamo il valore 124, la cui codifica binaria è 0111 1100. Con 4 istruzioni Lsr otteniamo il valore del nibble alto: 0000 0111, cioè 7. Il settimo valore della tabella è 55, il codice ASCII del numero 7, il nostro primo carattere. Ora prendiamo, con una And #15, il nibble basso di 124: 0000 1100, cioè 12. Il dodicesimo valore della tabella è 67, il codice ASCII della lettera C, il nostro secondo carattere. Ecco come 124 è trasformato in \$7C

#### Subroutine per la verifica di: spazio + indirizzo esadecimale (\$C12B-\$C153)

Il programma verifica i dati digitati, esaminando l'Input Buffer, la zona di memoria nella quale viene memorizzato il contenuto di una linea di schermo quando si preme Return.

Il registro X contiene il numero di cifre esadecimali da verificare. Per qualsiasi errore di sintassi viene settato il flag d'errore (la locazione 2). Al ritorno dalla subroutine il programma controlla tale flag per conoscere il risultato della verifica.

#### Subroutine per trasformare 4 caratteri esadecimale in un indirizzo (\$C15A-\$C17F)

Il registro X contiene il numero di byte da ottenere: 4 caratteri danno 2 byte, 2 caratteri ne danno 1 solo.

Per trovare il valore del primo carattere basta sottrarre 48 se è una cifra oppure 56 se è una lettera. Poi si moltiplica questo valore per 16 con quattro istruzioni Asl, ricavando il nibble alto del byte cercato. Analogamente si sottrae 48 o 56 dal secondo carattere ricavando il nibble basso del byte.

Se si vogliono 2 byte si trasformano così altri due caratteri. Per prendere un carattere è stata usata la subroutine \$0073, che termina con il carry azzerato se il carattere è una cifra, settato in caso contrario.

#### Subroutine Err (\$C185-\$C18C)

Sfrutta la subroutine \$AB1E per stampare il messaggio d'errore del Monitor.

*Subroutine che prende 2 indirizzi e 1 valore oppure 3 indirizzi (\$C19A-\$C1DB)*

Prima di chiamare questa subroutine il flag \$FA è posto a 2 se il terzo parametro è un indirizzo, è posto a 1 se esso è un valore.

La subroutine verifica i 3 parametri e, se tutto a posto, memorizza il primo in (\$FD;\$FE) e il secondo in (\$14;\$15). Se il terzo è un indirizzo lo memorizza in (\$FB;\$FC), altrimenti memorizza il valore in \$FB.

*Subroutine che confronta due indirizzi (\$C1E1-\$C1F2)*

Confronta l'indirizzo in (\$FD;\$FE) con l'indirizzo in (\$14;\$15). Se il primo è minore o uguale al secondo setta il registro X, altrimenti lo lascia a 0 (come deve essere prima di chiamare questa subroutine).

*Peek (\$C1F8-\$C278)*

Per trovare il multiplo di 8 più vicino all'indirizzo dato (da cui far partire la lettura) basta una And #248 effettuata sul byte basso di tale indirizzo.

Durante la lettura della memoria i valori esadecimale vengono stampati subito, mentre i caratteri corrispondenti si aggiungono ad una stringa che viene stampata solo prima di passare alla riga successiva. I valori che non si possono visualizzare come caratteri (Home, ecc) sono: tutti quelli minori di 32 e tutti quelli tra 128 e 159 compresi.

In \$C271 viene chiamata la subroutine \$FFE4 che esegue una Get prendendo un carattere dal buffer di tastiera e ponendolo nell'accumulatore.

*Poke (\$C28B-\$C2ED)*

Scrivo in memoria i valori digitati usando la subroutine \$C15E per passare dalla forma esadecimale al numero.

Poi stampa l'indirizzo successivo all'ultimo utilizzato e pone un Crsr Up e un Crsr Down nel buffer da tastiera perché in questo modo la prossima Input considererà tutti i caratteri della linea. Normalmente invece vengono considerati soltanto i caratteri a partire dalla posizione che aveva il cursore all'inizio della Input.

Infine elimina l'indirizzo di ritorno della subroutine e salta direttamente a \$C07D per la prossima Input, evitando la stampa

pa del Return.

*Poke ASCII (\$C2F3-\$C359)*

Scrivo in memoria i codici ASCII dei caratteri digitati prelevandoli dall'Input buffer.

Alla fine pone Crsr Up e Crsr Down nel buffer di tastiera ed elimina l'indirizzo per Rts proprio come nella subroutine Poke.

*Reg (\$C35F-\$C382)*

Visualizza il contenuto dei registri, che non corrispondono però ai loro valori attuali bensì a quelli posseduti al momento dell'ultima Sys. Questi valori saranno restituiti ai registri tornando al BASIC.

*Poke reg (\$C399-\$C3C1)*

Modifica il valore dei registri.

*Fill (\$C3DA-\$C3FA)*

Pone a 1 il flag \$FA in modo che la subroutine \$C19A, chiamata in \$C3DE, verifichi e memorizzi 2 indirizzi e 1 valore. Poi inizia il ciclo nel quale pone in memoria il valore e incrementa il primo indirizzo. Il ciclo termina quando il primo indirizzo è maggiore del secondo oppure se il primo arriva a 0, cosa che succede se il secondo indirizzo è \$FFFF

*Transfer (\$C402-\$C42A)*

Durante il ciclo di trasferimento della memoria viene incrementato sia il primo che il terzo indirizzo. L'uscita dal ciclo avviene come per la subroutine Fill.

*Hunt (\$C432-\$C49D)*

Memorizza da \$C4A1 in poi tutti i byte da cercare e in \$C4A0 il loro numero. Effettua il ciclo di ricerca stampando l'attuale indirizzo solo se tutti i valori coincidono con quelli presenti in memoria.

*Compare (\$C4C2-\$C4FD)*

Effettua il ciclo di ricerca stampando l'attuale indirizzo se il contenuto del primo indirizzo è diverso da quello del terzo.

*Subroutine per la verifica di nome file e numero periferica (\$C505-\$C550)*

Memorizza da \$C4A1 in poi il nome file e in \$C4A0 la sua lunghezza, la quale non deve superare i 16 caratteri.

Poi controlla che il numero della periferica sia 1 o 8. Nel caso esso sia 8 control-

la anche che il nome file non sia una stringa nulla.

*Subroutine per l'esecuzione di Save, Load e Verify (\$C557-\$C57D)*

Chiama la subroutine \$FFBA per assegnare i parametri (canale, numero periferica e indirizzo secondario) e la subroutine \$FFBD per assegnare il nome file e la sua lunghezza.

Poi legge il flag 10. Se è maggiore o uguale a 128 chiama la subroutine \$FFD8 per eseguire una Save. Altrimenti chiama la subroutine \$FFD5 che eseguirà Load o Verify a seconda che il flag 10 sia, rispettivamente, 0 o 1.

*Save (\$C583-\$C5BC)*

Verifica la sintassi e memorizza tutti i parametri, poi setta il flag 10 per la Save e chiama la subroutine \$C557 per l'esecuzione vera e propria del comando.

*Load (\$C5C4-\$C5DD)*

Verifica e memorizza i parametri, poi azzerà il flag 10 e chiama la subroutine \$C557 per l'esecuzione.

*Verify (\$C5E5-\$C613)*

Verifica e memorizza i parametri, poi setta a 1 il flag 10 e chiama la subroutine \$C557 per l'esecuzione. Al ritorno va a leggere il contenuto della locazione 144 (lo Status). Se il bit 4 è settato vuol dire che l'esito della Verify è negativo.

Stampa di conseguenza il messaggio di OK o quello di errore.

*Go (\$C63C-\$C65D)*

Una volta ottenuti i 2 byte dell'indirizzo di partenza, essi vengono memorizzati direttamente nel vettore di salto, cioè alla locazione \$C65A

*Esa-Dec (\$C665-\$C68A)*

Dopo aver trasformato i caratteri esadecimali in un indirizzo, pone la parte bassa di quest'ultimo in X, la parte alta in Accumulatore e chiama la subroutine \$BDCD che stampa, naturalmente in decimale, il numero contenuto in (A;X).

*Dec-Esa (\$C692-\$C713)*

All'inizio c'è un ciclo per la verifica del numero decimale, carattere per carattere. Il flag \$FA informa se bisogna effettuare un controllo aggiuntivo sulla grandezza del numero, dato che il massimo decimale accettato è 65535.



## COMMODORE 64

Se per esempio la prima cifra è 2, le cifre successive possono anche essere tutti 9 e quindi il flag \$FA viene azzerato (nessun ulteriore controllo). Ma se la prima cifra è 2, la seconda non potrà essere maggiore di 5 e perchè ci sia questo controllo il flag \$FA rimane settato.

Superata la verifica si passa alla conversione. Prima pone a 0 quello che sarà il numero esadecimale risultante, chiamiamolo Esa. Poi inizia il ciclo nel quale moltiplica Esa per 10 e somma al risultato una cifra del numero decimale. Quando le cifre sono finite esce dal ciclo e stampa Esa.

#### Subroutine per eseguire una Open al Drive (\$C721-\$C738)

Pone in (\$BB,\$BC) l'indirizzo del nomefile, e in \$B7 la sua lunghezza, che deve essere nel registro Y prima di chiamare questa subroutine. Poi chiama la subroutine \$F3D5 per una Open al bus seriale, la \$FFB4 per mandare un Talk, e la \$FF96 per mandare l'indirizzo secondario.

#### Drive (\$C741-\$C76A)

Se il controllo è arrivato qui, vuol dire che il comando inizia con >. Ora bisogna appurare di quale comando si tratta. Se è stato digitato il solo simbolo >, allora salta alla routine Stato (\$C774). Se c'è anche \$ salta alla routine Directory (\$C791). Se è presente il segno = va alla routine Ind (\$C7F4).

Se non rientra in nessuno dei casi precedenti, allora il comando è da inviare così com'è direttamente al drive. Si tratta per esempio di S (Scratch), V (Validate), o un altro dei tanti comandi possibili. L'indirizzo secondario è posto a 15 (il

canale di comando del drive) e viene chiamata la subroutine \$C721 per la Open. Poi c'è il salto alla routine Stato, che visualizzerà il messaggio del caso.

#### Stato (\$C774-\$C78A)

Dopo aver aperto la comunicazione con il drive la routine prende un carattere sfruttando la subroutine \$FFA5 e lo stampa, fino a quando non incontra un Chr\$(13). A questo punto esegue la subroutine \$F642 per la Close.

#### Directory (\$C791-\$C7E8)

Vediamo prima di tutto qual è la struttura della directory quando viene letta usando \$ (anche in BASIC con Load "\$",8).

Analizziamo le informazioni relative ad un singolo file. All'inizio ci sono due byte da saltare (il loro contenuto non ci interessa). Seguono 2 byte che rappresentano il numero di blocchi occupati dal file sul disco. Poi c'è il nome del file racchiuso tra virgolette e un certo numero di spazi (codice 32) che dipende dal numero di caratteri non utilizzati dal nome (se questo cioè è più corto di 16 caratteri).

Segue un codice di 3 caratteri indicante il tipo di file: Prg (programma), Seq (sequenziale), Usr (User) o Rel (Relative). Infine un Chr\$(0) che separa un gruppo di informazioni dal gruppo successivo, relativo ad un altro file.

Ci sono due eccezioni alla struttura di dati vista prima: il primo e l'ultimo gruppo di byte.

Il primo porta le informazioni relative al disco nel suo complesso. Inizia con 4 byte da saltare invece di 2, poi ci sono 2 byte uguali a 0, il nome del disco e per

ultima, prima del Chr\$(0), la sua ID.

L'ultimo gruppo di byte contiene il numero totale di blocchi liberi sul disco e la stringa "Blocks Free".

Dopo l'apertura del canale di comunicazione c'è un piccolo ciclo durante il quale vengono saltati i primi byte (come si è detto precedentemente). La subroutine \$BDCD chiamata in \$C7B5 stampa il numero di blocchi occupati dal file. Segue il ciclo di lettura di tutti i byte rimanenti.

Se è stato premuto un qualsiasi tasto, il programma attende che ne venga premuto un altro (qualsiasi). La subroutine \$FFE1 chiamata in \$C7CB controlla se si è premuto il tasto Stop. Se è così salta tutto il resto e va direttamente alla Close finale. Se il carattere appena letto è un Chr\$(0) la linea viene terminata con la stampa di un Return e si ricomincia tutto da capo per la nuova linea, byte da saltare, ecc.

Durante la lettura viene continuamente controllato lo Status, e se questo è diverso da 0 si pone fine all'esecuzione. Questo succede normalmente quando la directory è terminata, ma accade anche in caso di errore: disco non inserito o mal formattato o altre cose del genere.

#### Indirizzo d'inizio programma (\$C7F4-\$C82D)

Per conoscere l'indirizzo d'inizio di un programma residente su disco occorre procedere alla sua lettura come se si trattasse di un file sequenziale. I primi due byte letti non sono altro che l'indirizzo voluto.

Dopo la Open il programma prende il byte basso poi controlla lo Status; se tutto OK prende anche il byte alto e stampa l'indirizzo. Segue l'immane Close.

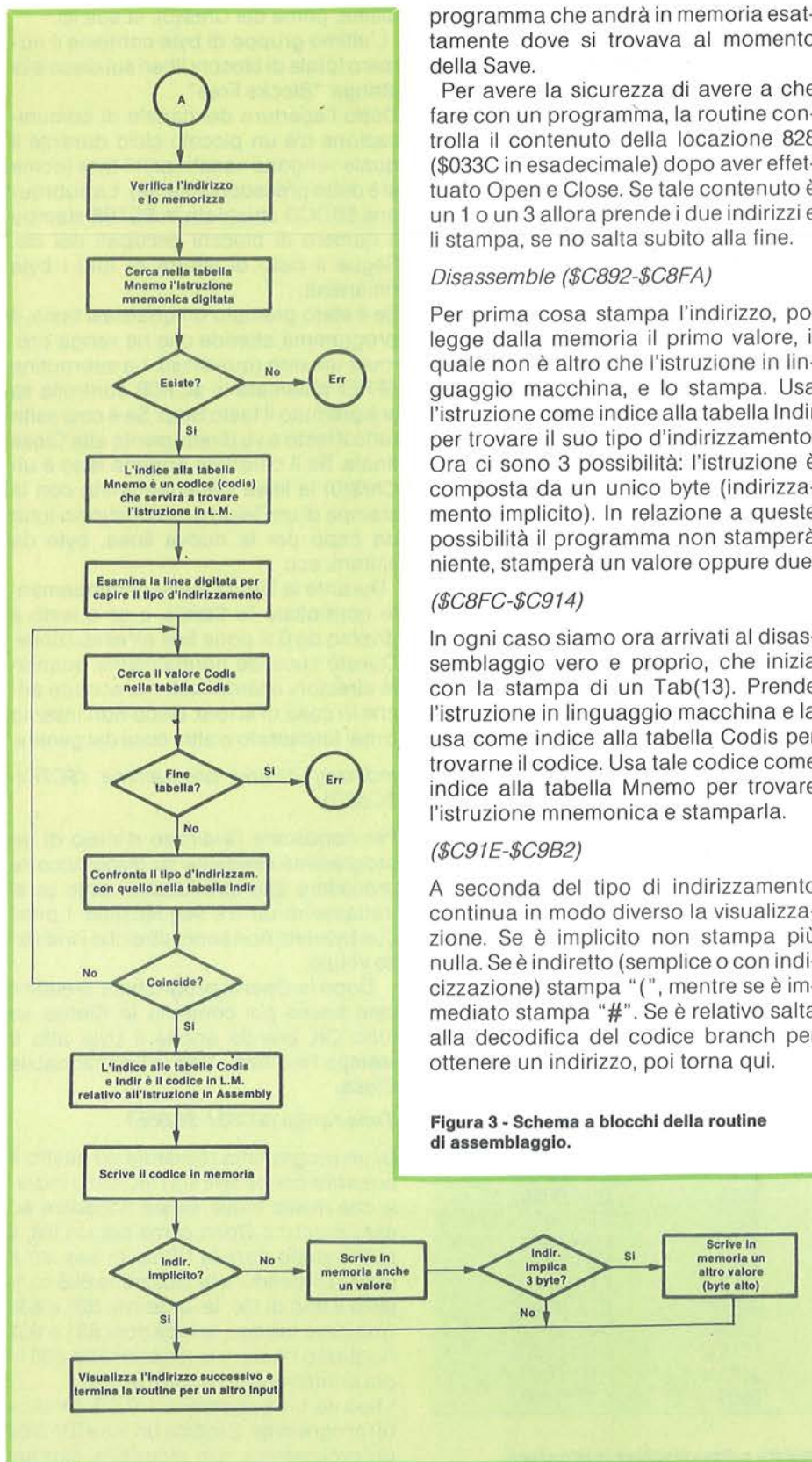
#### Tape range (\$C831-\$C88E)

Di un programma residente su nastro è possibile conoscere sia l'indirizzo iniziale che quello finale. Basta accedere ad esso con una Open come per un file, e subito dopo dare la Close. In seguito a questa procedura la locazione 828 conterrà il tipo di file, le locazioni 829 e 830 l'indirizzo iniziale, le locazioni 831 e 832 l'indirizzo finale, e le locazioni da 833 in poi conterranno il nome del file.

Il tipo del file può essere: 1, 2 o 3. 1 indica un programma, 2 indica un file e 3 indica un programma non rilocabile, cioè un

Codice	Nome	Sintassi	Esempio
0	Implicito		TXA
1	Immediato	###	LDA #07
2	Relativo	\$\$\$	BCC 9E56
3	Zero Page	\$\$	STA C6
4	Zero Page,X	\$\$,X	LDA FA,X
5	Zero Page,Y	\$\$,Y	LDX FA,Y
6	Indicizzato Indiretto	(\$,X)	CMP (FD,X)
7	Indiretto Indicizzato	(\$),Y	STA (FB),Y
8	Indiretto	(\$\$\$)	JMP (0300)
9	Assoluto,X	\$\$\$X	DEC 0402,X
10	Assoluto,Y	\$\$\$Y	AND D021,Y
11	Assoluto	\$\$\$	SBC D400

Tabella 2 - Tipi di indirizzamento (utilizzati per assemblare e disassemblare le istruzioni).



programma che andrà in memoria esattamente dove si trovava al momento della Save.

Per avere la sicurezza di avere a che fare con un programma, la routine controlla il contenuto della locazione 828 (\$033C in esadecimale) dopo aver effettuato Open e Close. Se tale contenuto è un 1 o un 3 allora prende i due indirizzi e li stampa, se no salta subito alla fine.

#### Disassemble (\$C892-\$C8FA)

Per prima cosa stampa l'indirizzo, poi legge dalla memoria il primo valore, il quale non è altro che l'istruzione in linguaggio macchina, e lo stampa. Usa l'istruzione come indice alla tabella Indir per trovare il suo tipo d'indirizzamento. Ora ci sono 3 possibilità: l'istruzione è composta da un unico byte (indirizzamento implicito). In relazione a queste possibilità il programma non stamperà niente, stamperà un valore oppure due.

#### (\$C8FC-\$C914)

In ogni caso siamo ora arrivati al disassemblaggio vero e proprio, che inizia con la stampa di un Tab(13). Prende l'istruzione in linguaggio macchina e la usa come indice alla tabella Codis per trovarne il codice. Usa tale codice come indice alla tabella Mnemo per trovare l'istruzione mnemonica e stamparla.

#### (\$C91E-\$C9B2)

A seconda del tipo di indirizzamento continua in modo diverso la visualizzazione. Se è implicito non stampa più nulla. Se è indiretto (semplice o con indicizzazione) stampa "(", mentre se è immediato stampa "#". Se è relativo salta alla decodifica del codice branch per ottenere un indirizzo, poi torna qui.

Figura 3 - Schema a blocchi della routine di assemblaggio.

Stampa un valore oppure un indirizzo e poi riprende in maniera diversa, sempre in relazione al tipo di indirizzamento, stampando ",X" o ")" o altro.

Fine della riga. Stampa un Return ed attende che venga premuto un tasto, per continuare oppure per interrompere il disassemblaggio.

#### Tabella Codis, Indir e Mnemo (\$C9CA-\$CC74)

Per comprendere come funzionano i comandi Disassemble e Assemble vediamo come sono strutturate queste tabelle.

Codis è un vettore di 256 elementi. Il codice che troviamo all'n-esimo posto rappresenta il punto della tabella Mnemo in cui c'è l'istruzione mnemonica relativa all'istruzione n in linguaggio macchina. Attenzione che l'espressione "n-esimo posto" indica un numero che va da 0 a 255 (e non da 1 a 256).

Facciamo un esempio chiarificatore: vogliamo conoscere l'istruzione mnemonica relativa al codice \$00 di linguaggio macchina. Allora n=0. Al posto 0 della tabella Codis troviamo il valore \$21. Questo valore dice di andare a guardare al posto \$21 della tabella Mnemo. \$21 corrisponde a 33 decimale. Se contiamo partendo da 0, al posto 33 della tabella Mnemo troviamo il valore ASCII della lettera B di Brk. Dal codice \$00 abbiamo quindi ottenuto la corretta istruzione Brk.

Analogamente al posto \$60 della tabella Codis c'è il valore \$81 che indica il punto della tabella Mnemo in cui c'è l'istruzione Rts.

Tutti i posti della tabella Codis che non sono codici di linguaggio macchina contengono il valore \$00 che porta al posto \$00 della tabella Mnemo, e qui troviamo i caratteri "---" ad indicare appunto un codice di linguaggio macchina non lecito.

La tabella Indir è molto più semplice. Anch'essa è un vettore di 256 elementi e contiene dei valori che indicano il tipo di indirizzamento (\$00 = implicito, \$01 = immediato, ecc.) secondo lo schema della tabella 2.

Al posto \$00 (istruzione \$00) c'è \$00 (indirizzamento implicito).

Al posto \$01 (istruzione \$01) c'è \$06 (indirizzamento indicizzato indiretto). E così via.

#### Assemble (figura 3) (\$CC80-\$CCC3)

## COMMODORE 64

Cerca nella tabella Mnemo l'istruzione mnemonica digitata. Se non esiste dà segnalazione d'errore. Altrimenti l'indice alla tabella è un codice che memorizza per dopo.

(\$CCC5-\$CE2A)

Esamina i caratteri digitati per trovare il tipo d'indirizzamento e memorizzare il valore o l'indirizzo presenti.

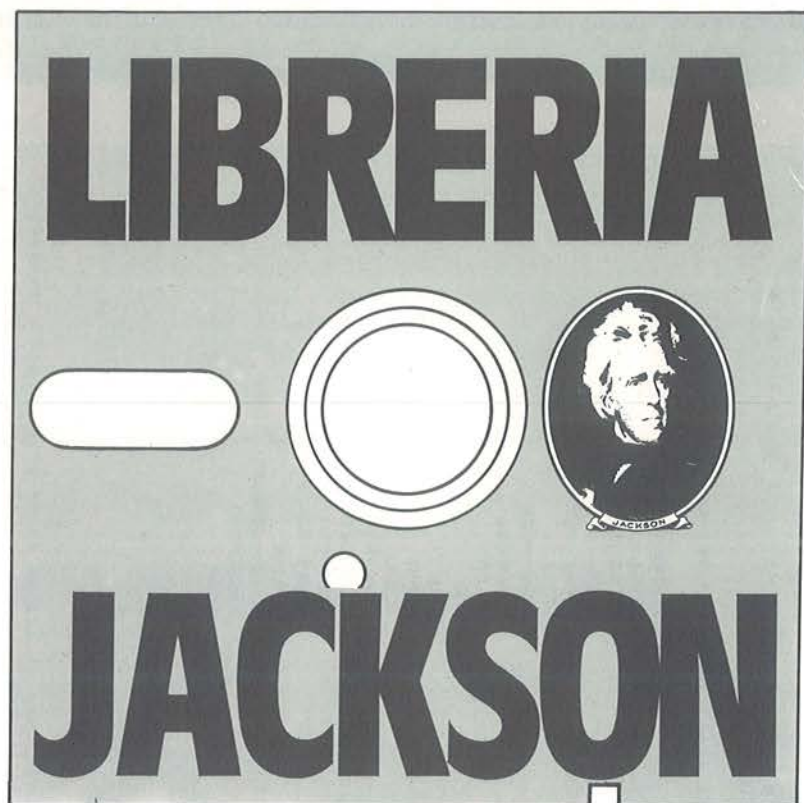
(\$E2C-\$E71)

Riprende il codice precedentemente memorizzato e lo cerca nella tabella Codis. Quando lo trova usa l'indice attuale alla tabella Codis (istruzione in linguaggio macchina) e lo utilizza per prendere un valore dalla tabella Indir. Confronta tale valore con il tipo d'indirizzamento trovato all'inizio. Se coincidono vuol dire che l'istruzione è stata digitata con un corretto tipo d'indirizzamento.

Se non coincidono incrementa l'indice alle tabelle Codis e Indir e continua il ciclo. Ma se arriva a fine tabella significa che il tipo d'indirizzamento digitato è illecito per quella istruzione.

All'uscita dal ciclo (se tutto OK) l'indice alle tabelle è proprio il codice linguaggio macchina cercato. Pone tale codice in memoria e, a seconda del tipo di indirizzamento pone in memoria un altro byte, altri due o nessuno. Poi salta alla fine della routine Poke per la stampa dell'indirizzo successivo all'ultimo usato e la preparazione ad un altro Input. ■

Continua



**A Milano,  
in via Mascheroni 14.  
La prima software  
libreria italiana**

Un tempo si andava in libreria per il gusto della scoperta, per il piacere di esser informati sulle novità. Per incontrarsi, discutere, chiedere un consiglio al libraio-amico. Tutto questo è ancora possibile, per un prodotto assolutamente nuovo: libri e riviste di informatica italiani ed esteri, software, giochi.

Dove? Alla **Libreria JACKSON.**

La prima software - libreria italiana.

**A Milano, in via Mascheroni 14.**

**Tel. 02-437385**

**Vieni a trovarci: ti aspettiamo.**



## COMMODORE 64

```

C0FF 48 PHA ;memorizza il valore da stampare
C100 4A LSR ;
C101 4A LSR ;
C102 4A LSR ;
C103 4A LSR ;Trova il 1° carattere e lo stampa
C104 AA TAX ;
C105 BD 16 C1 LDA C116,X ;
C106 20 D2 FF JSR FFD2 ;
C108 68 PLA ;
C10C 29 0F AND #0F ;
C10E AA TAX ;
C10F BD 16 C1 LDA C116,X ;Trova il 2° carattere e lo stampa
C112 20 D2 FF JSR FFD2 ;
C115 60 RTS ;
C116 ;
C116 ;Tabella dei codici ascii delle cifre esadecimali
C116 '0 '1 '2 '3 '4 '5 '6 '7 '8 '9 'A 'B 'C 'D 'E 'F
C126 ;

C126 EA NOP ;
C127 EA NOP ;
C128 EA NOP ;
C129 EA NOP ;
C12A EA NOP ;
C12B ;
C12B ;Sub. per la verifica di: spazio + indirizzo esadec.
C12B ;
C12B ;In X il num. di car. esad.
C12B A2 04 LDX #04 ;
C12D 00 02 BNE C131 ;
C12F A2 02 LDX #02 ;
C131 C8 INY ;
C132 B9 00 02 LDA 0200,Y ;Entry per verificare 2 car. esad.
C135 C9 20 CMP #20 ;Incrementa l'indice al car.
C137 00 18 BNE C151 ;Prende un car. dal comando
C139 C8 INY ;
C13A B9 00 02 LDA 0200,Y ;Se non e' uno spazio, ERR
C13D C9 30 CMP #30 ;Prende un altro car.
C13F 90 10 BCC C151 ;Se minore di '0', ERR
C141 C9 3A CMP #3A ;
C143 90 08 BCC C14D ;
C145 C9 41 CMP #41 ;
C147 90 08 BCC C151 ;Se magg. di '9 e min. di 'A', ERR
C149 C9 47 CMP #47 ;
C14B 80 04 BCS C151 ;Se maggiore di 'F', ERR
C14D CA DEX ;
C14E 00 E9 BNE C139 ;Se ce ne sono ancora, continua
C150 60 RTS ;
C151 C6 02 DEC 02 ;Setta il flag d'errore
C153 00 FB BNE C150 ;
C155 EA NOP ;
C156 EA NOP ;
C157 EA NOP ;
C158 EA NOP ;
C159 EA NOP ;
C15A ;

C15A ;Sub. per trasformare 4 car. esad. in un indirizzo
C15A ;
C15A A2 02 LDX #02 ;In X il num. di byte da ottenere
C15C 00 02 BNE C160 ;
C15E A2 01 LDX #01 ;Entry per trasf. 2 car. (1 byte)
C160 20 73 00 JSR 0073 ;Prende un car.
C163 90 02 BCC C167 ;
C165 E9 08 SBC #08 ;Ne trova il valore
C167 E9 2F SBC #2F ;
C169 0A ASL ;
C16A 0A ASL ;
C16B 0A ASL ;Lo memorizza come nibble alto
C16C 0A ASL ;
C16D 95 FA STA FA ;
C16F 20 73 00 JSR 0073 ;Prende un altro car.
C172 90 02 BCC C176 ;
C174 E9 08 SBC #08 ;
C176 E9 2F SBC #2F ;Ne trova il valore
C178 15 FA ORA FA,X ;
C17A 95 FA STA FA,X ;
C17C CA DEX ;Lo memorizza come nibble basso
C17D 00 E1 BNE C160 ;
C17F 60 RTS ;Se si tratta di un $$$$, continua
C180 EA NOP ;
C181 EA NOP ;
C182 EA NOP ;
C183 EA NOP ;
C184 EA NOP ;
C185 ;
C185 ;ERR (sub. chiamata per un errore dell'utente)
C185 ;
C185 A9 8F LDA #8F ;
C187 A0 C1 LDY #C1 ;
C189 20 1E AB JSR AB1E ;Stampa il messaggio d'errore
C18C 60 RTS ;
C18D EA NOP ;
C18E EA NOP ;
C18F ;
C18F ;Messaggio d'errore
C18F ;
C18F '* 'E 'R 'O 'R 'E ' '* 00
C19A ;
C19A ;Sub. che verifica: ' $$$ $ opp. ' $$$ $ $$$ $ $$$ $
; ne ricava i valori e li memorizza
C19A ;
C19A A0 00 LDY #00 ;
C19C 20 2B C1 JSR C12B ;Verifica ' $$$ $ $$$ $ $$$ $ $$$ $
C19F 20 2B C1 JSR C12B ;
C1A2 20 2F C1 JSR C12F ;
C1A5 C6 FA DEC FA ;
C1A7 F0 05 BEQ C1AE ;
C1A9 A2 02 LDX #02 ;
C1AB 20 39 C1 JSR C139 ;Se il flag FA era posto a 2,
; verifica '$$ (ci sono 3 indir.)
C1AE B9 01 02 LDA 0201,Y ;
C1B1 F0 02 BEQ C1B5 ;
C1B3 85 02 STA 02 ;Se dopo il comando c'e' altro...

```

Seguito figura 2.

```
C185 A5 02 LDA 02 ;e/o il flag 2 e' settato, salta
C187 D0 22 BNE C10B ;
C189 20 5A C1 JSR C15A ;
C19C A5 FB LDA FB ;Memorizza il 1' indir. in (FD;FE)
C1BE 95 FD STA FD ;
C1C0 A5 FC LDA FC ;
C1C2 95 FE STA FE ;
C1C4 20 5A C1 JSR C15A ;
C1C7 A5 FB LDA FB ;Memorizza il 2' indir. in (14;15)
C1C9 95 14 STA 14 ;
C1CB A5 FC LDA FC ;
C1CD 95 15 STA 15 ;
C1CF C6 FA DEC FA ;
C1D1 D0 05 BNE C108 ;
C1D3 20 5A C1 JSR C15A ;Memorizza il 3' indir. in (FB;FC)
C1D6 F0 03 BEQ C10B ;
C1D8 20 SE C1 JSR C15E ;...oppure il valore in FB
C1DB 60 RTS ;
C1DD EA NOP ;
C1DE EA NOP ;
C1DF EA NOP ;
C1E0 EA NOP ;
C1E1 ;
C1E1 ;Sub. che confronta l'ind. (FD;FE) con l'ind. (14;15)
C1E1 ;
C1E1 A5 FE LDA FE ;
C1E3 C5 15 CMP 15 ;Confronta le parti alte
C1E5 90 0A BCC C1F1 ;hi1 < hi2
C1E7 D0 09 BNE C1F2 ;hi1 > hi2
C1E9 A5 FD LDA FD ;
C1EB C5 14 CMP 14 ;Confronta le parti basse
C1ED 90 02 BCC C1F1 ;lo1 < lo2
C1EF D0 01 BNE C1F2 ;lo1 > lo2
C1F1 CA DEX ;Setta X: indica che l'operazione,
C1F2 60 RTS ; dopo il confronto, va eseguita
C1F3 EA NOP ;
C1F4 EA NOP ;
C1F5 EA NOP ;
C1F6 EA NOP ;
C1F7 EA NOP ;
C1F8 ;
C1F8 ;PEEK sintassi: M ind
C1F8 ;
C1F8 ; LDY #00 ;
C1FA 20 2B C1 JSR C12B ;Verifica l'indir.
C1FD C8 INY ;
C1FE B9 00 02 LDA 0200,Y ;
C201 D0 04 BNE C207 ;Se c'e' altro dopo il comando,
C203 A5 02 LDA 02 ;
C205 F0 03 BEQ C20A ;
C207 4C 85 C1 JMP C185 ;e/o il flag 2 e' settato, ERR
C20A 20 5A C1 JSR C15A ;Conv. l'ind. e lo pone in (FB;FC)
C20D A5 FB LDA FB ;
C20F 29 F8 AND #F8 ;Trova il multiplo di 8 piu'
C211 85 FB STA FB ; vicino all'ind. e lo memorizza

C213 A5 FC LDA FC ;
C215 20 FF C0 JSR FFC0 ;
C218 A5 FB LDA FB ;Stampa l'ind.
C21A 20 FF C0 JSR C0FF ;
C21D A9 2E LDA #2E ;
C21F A0 08 LDY #08 ;
C221 99 7A C2 STA C27A,Y ;inizializza la stringa CHR
C224 88 DEY ;
C225 D0 FA BNE C221 ;
C227 A9 20 LDA #20 ;
C229 20 D2 FF JSR FFD2 ;Stampa uno spazio
C22C A0 00 LDY #00 ;
C22E A9 20 LDA #20 ;
C230 20 D2 FF JSR FFD2 ;Stampa uno spazio
C233 B1 FB LDA (FB),Y ;Prende un valore dalla memoria
C235 C9 20 CMP #20 ;
C237 90 08 BCC C244 ;
C239 C9 80 CMP #80 ;
C23B 90 04 BCC C241 ;
C23D C9 A0 CMP #A0 ;
C23F 90 03 BCC C244 ;Se e' un carattere stampabile,
C241 99 7B C2 STA C27B,Y ; lo aggiunge alla stringa CHR
C244 20 FF C0 JSR C0FF ;Stampa il valore
C247 C8 INY ;
C248 C0 08 CPY #08 ;
C24A D0 E2 BNE C22E ;Se non ne ha stampati 8, continua
C24C A9 20 LDA #20 ;
C24E 20 D2 FF JSR FFD2 ;Stampa uno spazio
C251 A9 3A LDA #3A ;
C253 20 D2 FF JSR FFD2 ;Stampa i due punti
C256 A9 7B LDA #7B ;
C258 A0 C2 LDY #C2 ;
C25A 20 1E AB JSR AB1E ;Stampa la stringa CHR
C25D A9 0D LDA #0D ;
C25F 20 D2 FF JSR FFD2 ;Stampa un [RETURN]
C262 18 CLC ;
C263 A9 08 LDA #08 ;
C265 65 FB ADC FB ;
C267 95 FB STA FB ;
C269 D0 02 BNE C26D ;
C26B E6 FC INC FC ;
C26D A5 C6 LDA C6 ;
C26F F0 FC BEQ C26D ;Aspetta che sia premuto un tasto
C271 20 E4 FF JSR FFE4 ;Esegue una GET da tastiera
C274 C9 00 CMP #00 ;
C276 F0 9B BEQ C213 ;Se il car. e' [RETURN], continua
C278 60 RTS ;Altrimenti, fine
C279 EA NOP ;
C27A EA NOP ;
C27B ;
C27B ;CHR (stringa di 8 caratteri + 00)
C27B ;
C27B ;
C27B ;
C284 ;
C284 EA NOP ;
C285 EA NOP ;
```

## COMMODORE 64

```

C286 EA ;
C287 EA ;
C288 EA ;
C289 EA ;
C28A EA ;
C28B ;
C28C ;POKE
C28D ; ind val val ...
C28E ;
C28F ;
C290 LDY #00 ;
C291 JSR C12B ;Verifica l'indir.
C292 LDA #01 ;
C293 BEQ C296 ;Se non segue altro, fine
C294 RTS ;
C295 ;
C296 ;
C297 JSR C12F ;Verifica un valore
C298 LDA #02 ;
C299 BEQ C2A0 ;
C29A ;
C29B ;
C29C ;Se il flag 2 e' settato, ERR
C29D ;Se c'e' un altro valore, continua
C29E ;
C29F ;
C2A0 LDA #01 ;
C2A1 BEQ C296 ;
C2A2 ;
C2A3 ;
C2A4 JSR C15A ;
C2A5 LDA #02 ;
C2A6 STA #03 ;Conv. l'ind. e lo pone in (FD;FE)
C2A7 LDA #04 ;
C2A8 ;
C2A9 ;
C2AA ;
C2AB ;
C2AC ;
C2AD ;
C2AE ;
C2AF ;
C2B0 ;
C2B1 ;
C2B2 ;
C2B3 ;
C2B4 ;
C2B5 ;
C2B6 ;
C2B7 ;
C2B8 ;
C2B9 ;
C2BA ;
C2BB ;
C2BC ;
C2BD ;
C2BE ;
C2BF ;
C2C0 ;
C2C1 ;
C2C2 ;
C2C3 ;
C2C4 ;
C2C5 ;
C2C6 ;
C2C7 ;
C2C8 ;
C2C9 ;
C2CA ;
C2CB ;
C2CC ;
C2CD ;
C2CE ;
C2CF ;
C2D0 ;
C2D1 ;
C2D2 ;
C2D3 ;
C2D4 ;
C2D5 ;
C2D6 ;
C2D7 ;
C2D8 ;
C2D9 ;
C2DA ;
C2DB ;
C2DC ;
C2DD ;
C2DE ;
C2DF ;
C2E0 ;
C2E1 ;
C2E2 ;
C2E3 ;
C2E4 ;
C2E5 ;
C2E6 ;
C2E7 ;
C2E8 ;
C2E9 ;
C2EA ;
C2EB ;
C2EC ;
C2ED ;
C2EE ;
C2EF ;
C2F0 ;
C2F1 ;
C2F2 ;
C2F3 ;
C2F4 ;
C2F5 ;
C2F6 ;
C2F7 ;
C2F8 ;
C2F9 ;
C2FA ;
C2FB ;
C2FC ;
C2FD ;
C2FE ;
C2FF ;
C300 ;
C301 ;
C302 ;
C303 ;
C304 ;
C305 ;
C306 ;
C307 ;
C308 ;
C309 ;
C30A ;
C30B ;
C30C ;
C30D ;
C30E ;
C30F ;
C310 ;
C311 ;
C312 ;
C313 ;
C314 ;
C315 ;
C316 ;
C317 ;
C318 ;
C319 ;
C31A ;
C31B ;
C31C ;
C31D ;
C31E ;
C31F ;
C320 ;
C321 ;
C322 ;
C323 ;
C324 ;
C325 ;
C326 ;
C327 ;
C328 ;
C329 ;
C32A ;
C32B ;
C32C ;
C32D ;
C32E ;
C32F ;
C330 ;
C331 ;
C332 ;
C333 ;
C334 ;
C335 ;
C336 ;
C337 ;
C338 ;
C339 ;
C33A ;
C33B ;
C33C ;
C33D ;
C33E ;
C33F ;
C340 ;
C341 ;
C342 ;
C343 ;
C344 ;
C345 ;
C346 ;
C347 ;
C348 ;
C349 ;
C34A ;
C34B ;
C34C ;
C34D ;
C34E ;
C34F ;
C350 ;
C351 ;
C352 ;
C353 ;
C354 ;
C355 ;
C356 ;
C357 ;
C358 ;
C359 ;
C35A ;
C35B ;
C35C ;
C35D ;
C35E ;
C35F ;

```

Segue figura 2.

```
C35F ;REG          sintassi: R
C35F ;
C35F AS 85        ; LDA #85
C361 A0 C3        ; LDY #C3
C363 20 1E AB     ; JSR AB1E
C366 A9 2E        ; LDA #2E
C368 20 D2 FF     ; JSR FFD2
C36B A0 00        ; LDY #00
C36D A9 20        ; LDA #20
C36F 20 D2 FF     ; JSR FFD2
C372 B9 0C 03    ; LDA 030C,Y
C375 20 FF C0    ; JSR C0FF
C378 C8          ; INY
C379 C0 04        ; CPY #04
C37B D0 F0        ; BNE C36D
C37D A9 00        ; LDA #00
C37F 20 D2 FF     ; JSR FFD2
C382 60          ; RTS
C383 EA          ; NOP
C384 EA          ; NOP
C385 ;            ;Stringa contenente i nomi dei registri
C385 ; 'A 'C 'X 'R 'Y 'R 'S 'R 00 00
C385 ;
C385 ;
C385 ;
C394 ;
C394 EA          ; NOP
C395 EA          ; NOP
C396 EA          ; NOP
C397 EA          ; NOP
C398 EA          ; NOP
C399 ;
C399 ;POKE REG    sintassi: . val val val val
C399 ;
C399 AS 04        ; LDA #04
C39B 85 FB        ; STA FB
C39D A0 00        ; LDY #00
C39F 20 2F C1    ; JSR C12F
C3A2 C6 FB        ; DEC FB
C3A4 D0 F9        ; BNE C39F
C3A6 B9 01 02    ; LDA 0201,Y
C3A9 D0 04        ; BNE C3AF
C3AB A5 02        ; LDA 02
C3AD F0 03        ; BEQ C3B2
C3AF 4C 85 C1    ; JMP C185
C3B2 A0 00        ; LDY #00
C3B4 20 5E C1    ; JSR C15E
C3B7 A5 FB        ; LDA FB
C3B9 99 0C 03    ; STA 030C,Y
C3BC C8          ; INY
C3BD C0 04        ; CPY #04
C3BF D0 F3        ; BNE C3B4
C3C1 60          ; RTS
C3C2 EA          ; NOP
C3C3 EA          ; NOP
C3C4 ;
C3C4 ;Zona di memoria non utilizzata

C3C4 ;
C3C4 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
C3D4 00 00 00 00 00 00
C3DA ;
C3DA ;FILL
C3DA ; sintassi: F ind1 ind2 val
C3DA ;
C3DA AS 01        ; LDA #01
C3DA ;FA=1 indica alla sub. C19A che
C3DA STA FA      ; il comando termina con un valore
C3DA JSR C19A    ;Verifica e mem. i dati d. comando
C3DE 20 9A C1    ; LDA 02
C3E1 A5 02        ; BEQ C3E8
C3E3 F0 03        ; Se il flag 2 e' settato, ERR
C3E5 4C 85 C1    ; JMP C185
C3E8 20 E1 C1    ; JSR C1E1
C3EB E8          ; INX
C3EC D0 0C        ; BNE C3FA
C3EC ;Se il 1' e' maggiore del 2', fine
C3EE A5 FB        ; LDA FB
C3F0 81 FD        ; STA (FD,X)
C3F2 E8 FD        ; INC FD
C3F4 D0 F2        ; BNE C3E8
C3F6 E8 FE        ; INC FE
C3F8 D0 EE        ; BNE C3E8
C3FA 60          ; RTS
C3FA ;fine
C3FB EA          ; NOP
C3FC EA          ; NOP
C3FD EA          ; NOP
C3FE EA          ; NOP
C3FF EA          ; NOP
C400 EA          ; NOP
C401 EA          ; NOP
C402 ;
C402 ;TRANSFER
C402 ; sintassi: T ind1 ind2 ind3
C402 ;
C402 AS 02        ; LDA #02
C402 ;FA=2 indica alla sub. C19A, che
C404 85 FA        ; STA FA
C406 20 9A C1    ; JSR C19A
C409 A5 02        ; LDA 02
C40B F0 03        ; BEQ C410
C40D 4C 85 C1    ; JMP C185
C410 20 E1 C1    ; JSR C1E1
C413 E8          ; INX
C414 D0 14        ; BNE C42A
C414 ;Se ind1 > ind2, fine
C416 A1 FD        ; LDA (FD,X)
C418 81 FB        ; STA (FB,X)
C41A E8 FD        ; INC FD
C41C D0 04        ; BNE C422
C41E E8 FE        ; INC FE
C420 F0 08        ; BEQ C42A
C422 E8 FB        ; INC FB
C424 D0 EA        ; BNE C410
C426 E8 FC        ; INC FC
C428 D0 E6        ; BNE C410
C42A 60          ; RTS
C42B EA          ; NOP
C42C EA          ; NOP
C42D EA          ; NOP
C42E EA          ; NOP
```



## COMMODORE 64

```

C42F EA ;
C430 NOP ;
C431 NOP ;
C432 NOP ;
C432 ;
C432 ;HUNT ;
C432 ;
C432 LDY #00 ;
C434 STY C4A0 ;Inizializza a 0 il numero
C437 20 2B C1 JSR C12B ;dei byte da cercare
C43A 20 2B C1 JSR C12B ;
C43D EE A0 C4 INC C4A0 ;Verifica gli ind. e tutti i val.
C440 20 2F C1 JSR C12F ;presenti, e mem. il num. dei val.
C443 B9 01 02 LDA 0201,Y ;
C446 D0 F5 BNE C43D ;
C448 A5 02 LDA 02 ;
C44A F0 03 BEQ C44F ;
C44C 4C 85 C1 JMP C185 ;Se il flag 2 e' settato, ERR
C44F 85 FA STA FA ;FA=0 indica alla sub. C189 che
C451 20 B9 C1 JSR C189 ;sono presenti solo 2 ind.
C454 80 A1 C4 STA C4A1 ;Memorizza il primo val.
C457 A0 01 LDY #01 ;
C459 C0 A0 C4 CPY C4A0 ;
C45C F0 09 BEQ C467 ;
C45E 20 5E C1 JSR C15E ;Memorizza (se ce ne sono),
C461 99 A1 C4 STA C4A1,Y ;tutti gli altri val.
C464 C8 INY ;
C465 D0 F2 BNE C459 ;
C467 20 E1 C1 JSR C1E1 ;Confronta ind1 e ind2
C46A E8 INX ;
C46B D0 30 BNE C49D ;
C46D A0 00 LDY #00 ;
C46F 84 FA STY FA ;
C471 B1 FD LDA (FD),Y ;
C473 D9 A1 CMP C4A1,Y ;
C476 F0 02 BEQ C47A ;
C478 C6 FA DEC FA ;
C47A C8 INY ;Confronta i byte in memoria con
C47B CC A0 C4 CPY C4A0 ;quelli del comando. Se i o piu'
C47E D0 F1 BNE C471 ;non coincidono, setta FA
C480 A5 FA LDA FA ;
C482 D0 11 BNE C495 ;
C484 A9 20 LDA #20 ;
C486 20 D2 FF JSR FFD2 ;Se il flag FA e' settato, salta
C489 A5 FE LDA FE ;
C48B 20 FF C0 JSR C0FF ;Stampa uno spazio
C48E A5 FD LDA FD ;Stampa la locazione attuale
C490 20 FF C0 JSR C0FF ;
C493 A2 00 LDX #00 ;Azzera X per la prossima JSR C1E1
C495 E6 FD INC FD ;
C497 D0 CE BNE C467 ;Esegue: ind1 = ind1 + 1
C499 E6 FE INC FE ;
C49B D0 CA BNE C467 ;
C49D 60 RTS ;fine
C49E EA NOP ;
C49F EA NOP ;
C4A0 ;

```

```

C4A0 ;N. byte
C4A0 ;
C4A0 00
C4A1 ;
C4A1 ;Byte usati da HUNT e altre routine
C4A1 ;
C4A1 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
C4B1 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
C4BD ;
C4BD EA NOP ;
C4BD EA NOP ;
C4BE EA NOP ;
C4BF EA NOP ;
C4C0 EA NOP ;
C4C1 EA NOP ;
C4C2 ;
C4C2 ;COMPARE ; sintassi: C ind1 ind2 ind3
C4C2 ;
C4C2 A9 02 LDA #02 ;
C4C4 85 FA STA FA ;Verifica e memorizza i tre ind.
C4C6 20 9A C1 JSR C19A ;
C4C9 A5 02 LDA 02 ;
C4CB F0 03 BEQ C4D0 ;
C4CD 4C 85 C1 JMP C185 ;Se il flag 2 e' settato, ERR
C4D0 20 E1 C1 JSR C1E1 ;Confronta ind1 e ind2
C4D3 E8 INX ;
C4D4 D0 27 BNE C4FD ;Se ind1 > ind2, fine
C4D6 A1 FD LDA (FD,X) ;
C4D8 C1 FB CMP (FB,X) ;Confr. i contenuti di ind1 e ind3
C4DA F0 11 BEQ C4ED ;Se sono uguali, salta
C4DC A9 20 LDA #20 ;
C4DE 20 D2 FF JSR FFD2 ;Stampa uno spazio
C4E1 A5 FE LDA FE ;
C4E3 20 FF C0 JSR C0FF ;
C4E6 A5 FD LDA FD ;Stampa l'ind. attuale
C4E8 20 FF C0 JSR C0FF ;
C4EB A2 00 LDX #00 ;Azzera X per la prossima JSR C1E1
C4ED E6 FD INC FD ;
C4EF D0 04 BNE C4F5 ;
C4F1 E6 FE INC FE ;Esegue: ind1 = ind1 + 1
C4F3 F0 08 BEQ C4FD ;
C4F5 E6 FB INC FB ;
C4F7 D0 D7 BNE C4D0 ;
C4F9 E6 FC INC FC ;Esegue: ind3 = ind3 + 1
C4FB D0 D3 BNE C4D0 ;
C4FD 60 RTS ;
C4FE EA NOP ;
C4FF EA NOP ;
C500 EA NOP ;
C501 EA NOP ;
C502 EA NOP ;
C503 EA NOP ;
C504 EA NOP ;
C505 ;
C505 ;Sub. che verifica: [spazio]nomefile' device
C505 ; (usata da SAVE, LOAD e VERIFY)
C505 ;

```

## Seguito figura 2.

```

C505 A0 01 02 LDA 0201 ;
C508 C9 20 CMP #20 ;
C50A D0 42 BNE C54E ;Controlla che ci sia uno spazio
C50C A0 02 02 LDA 0202 ;e un apice
C50F C9 27 CMP #27 ;
C511 D0 38 BNE C54E ;
C513 A0 02 LDY #02 ;
C515 A2 00 LDX #00 ;
C517 C8 INY ;
C518 B9 00 02 LDA 0200,Y ;Prende un carattere
C51B C9 27 CMP #27 ;
C51D F0 0A BEQ C529 ;Se e' un apice,nomefile e' finito
C51F 9D A1 C4 STA C4A1,X ;Se no, memorizza il car.
C522 E8 INX ;
C523 E0 11 CPX #11 ;Se nomefile non e' troppo lungo,
C525 D0 F0 BNE C517 ;(max 16 car.) torna in ciclo
C527 F0 25 BEQ C54E ;Se lo e', ERR
C529 8E A0 C4 STX C4A0 ;Memorizza la lunghezza del nomefile
C52C C8 INY ;
C52D B9 00 02 LDA 0200,Y ;
C530 C9 20 CMP #20 ;Controlla che segua uno spazio...
C532 D0 15 BNE C549 ;
C534 C8 INY ;
C535 B9 00 02 LDA 0200,Y ;
C538 C9 30 CMP #30 ;e uno zero
C53A D0 0D BNE C549 ;
C53C C8 INY ;
C53D B9 00 02 LDA 0200,Y ;
C542 F0 09 BEQ C54D ;
C544 C9 38 CMP #38 ;
C548 AE A0 C4 LDX C4A0 ;Se non e' '1 e non e' '8', ERR
C54B F0 01 BEQ C54E ;
C54D 60 RTS ;Se e' '8, nomefile deve esserci
C54E C6 02 DEC 02 ;Setta il flag d'errore
C550 D0 FB BNE C54D ;
C552 EA NOP ;
C553 EA NOP ;
C554 EA NOP ;
C555 EA NOP ;
C556 EA NOP ;
C557 ;
C557 ;Sub. usata da SAVE, LOAD e VERIFY
C557 ; rimanda l'esecuzione alle routine del Kernal
C557 ;
C557 A9 0F LDA #0F ;
C559 A6 FA LDX FA ;
C55B A0 01 LDY #01 ;
C55D 20 BA FF JSR FFBA ;
C560 A0 A0 C4 LDA C4A0 ;
C563 A2 A1 LDX #A1 ;
C565 A0 C4 LDY #C4 ;
C567 20 8D FF JSR FF8D ;
C56A A5 0A LDA 0A ;
C56C 30 08 BMI C576 ;Se 0A >= 128, va a eseguire Save
C56E A2 00 LDX #00 ;
C570 A0 00 LDY #00 ;
C572 20 D5 FF JSR FF5D ;
C575 60 RTS ;
C576 A6 FB LDX FB ;
C578 A4 FC LDY FC ;
C57A 20 D8 FF JSR FF88 ;
C57D 60 RTS ;
C57E EA NOP ;
C57F EA NOP ;
C580 EA NOP ;
C581 EA NOP ;
C582 EA NOP ;
C583 ;
C583 ; sintassi: S 'nomefile' dev inizio fine
C583 ;
C583 20 05 C5 JSR C505 ;Verifica nomefile e n. periferica
C586 20 28 C1 JSR C128 ;
C589 20 28 C1 JSR C128 ;
C58C B9 01 02 LDA 0201,Y ;Verifica gli indirizzi e che
C58F D0 04 BNE C595 ;non segua nient'altro
C591 A5 02 LDA 02 ;
C593 F0 03 BEQ C598 ;
C595 4C 85 C1 JMP C185 ;Se il flag 2 e' settato, ERR
C598 20 73 00 JSR 0073 ;Prendi un car. (il 1' apice)
C59B 20 73 00 JSR 0073 ;
C59E C9 27 CMP #27 ;
C5A0 D0 F9 BNE C59B ;
C5A2 20 5E C1 JSR C15E ;Memorizza in FA il n. periferica
C5A5 85 FA STA FA ;
C5A7 20 5A C1 JSR C15A ;
C5A9 A5 FB LDA FB ;
C5AC 85 FD STA FD ;
C5AE A5 FC LDA FC ;
C5B0 85 FE STA FE ;
C5B2 20 5A C1 JSR C15A ;Memorizza in <FD>FE)
C5B5 A9 FD LDA #FD ;
C5B7 85 0A STA 0A ;
C5B9 20 57 C5 JSR C557 ;
C5BC 60 RTS ;
C5BD EA NOP ;
C5BE EA NOP ;
C5BF EA NOP ;
C5C0 EA NOP ;
C5C1 EA NOP ;
C5C2 EA NOP ;
C5C3 EA NOP ;
C5C4 ;
C5C4 ; sintassi: L 'nomefile' dev
C5C4 ;
C5C4 20 05 C5 JSR C505 ;Verifica nomefile e n. periferica
C5C7 38 SEC ;
C5C8 E9 30 SBC #30 ;
C5CA 85 FA STA FA ;Memorizza in FA il n. periferica
C5CC B9 01 02 LDA 0201,Y ;
C5CF D0 04 BNE C5D5 ;Se c'e' qualcos'altro dopo ...

```

## COMMODORE 64

```

C5D1 A5 02 LDA 02 ;
C5D3 F0 03 BEQ C5D8 ;
C5D5 4C 85 C1 JMP C185 ;e/o il flag 2 e' settato, ERR
C5D8 85 0A STA 0A ;0A=0 per la Load
C5DA 20 57 C5 JSR C557 ;ed esegue
C5DD 60 RTS ;
C5DE EA NOP ;
C5DF EA NOP ;
C5E0 EA NOP ;
C5E1 EA NOP ;
C5E2 EA NOP ;
C5E3 EA NOP ;
C5E4 EA NOP ;
C5E5 ;
C5E5 ;VERIFY ; sintassi: V 'nomefile' dev
C5E5 ;
C5E5 20 05 C5 JSR C505 ;Verifica nomefile e n. periferica
C5E8 38 SEC ;
C5E9 E9 30 SBC #30 ;
C5EB 85 FA STA FA ;Memorizza in FA il n. periferica
C5ED B9 01 02 LDA 0201,Y ;-----
C5F0 00 04 BNE C5F6 ;Se e' presente altro nel comando
C5F2 A5 02 LDA 02 ;
C5F4 F0 03 BEQ C5F9 ;
C5F6 4C 85 C1 JMP C185 ;e/o il flag 2 e' settato, ERR
C5F9 A9 01 LDA #01 ;-----
C5FB 85 0A STA 0A ;Setta a 1 il flag FA per Verify
C5FD 20 57 C5 JSR C557 ;ed esegue
C600 A5 90 LDA 90 ;-----
C602 29 10 AND #10 ;-----
C604 F0 0A BEQ C610 ;Controlla il risultato di Verify
C606 A9 17 LDA #17 ;(Status), e se OK, salta avanti
C608 A0 C6 LDY #C6 ;-----
C60A 20 1E AB JSR AB1E ;Prende il messaggio di errore
C60D 60 RTS ;Stampa il messaggio, e fine
C60E A8 2C LDA #2C ;-----
C610 A0 C6 LDY #C6 ;
C612 18 CLC ;
C613 90 F5 BCC C60A ;Prende il messaggio di OK
C615 EA NOP ;
C616 EA NOP ;
C617 ;
C617 ;Messaggio di errore per VERIFY
C617 ;
C617 00 '* '* 'V'E'R'I'F'Y' 'E'R'R'O'R'
C627 '* '* 00 00
C62C ;
C62C ;Messaggio di OK per VERIFY
C62C ;
C62C 00 '* '* 'O'K' '* '* 00 00
C637 ;
C637 EA NOP ;
C638 EA NOP ;
C639 EA NOP ;
C63A EA NOP ;
C63B EA NOP ;

C63C ; sintassi: G ind
C63C ;60
C63C ;
C63C LDA #00 ;
C63E 20 2B C1 JSR C12B ;Verifica l'ind.
C641 B9 01 02 LDA 0201,Y ;
C644 00 04 BNE C64A ;e che non ci sia nient'altro
C646 A5 02 LDA 02 ;
C648 F0 03 BEQ C64D ;
C64A 4C 85 C1 JMP C185 ;Se il flag 2 e' settato, ERR
C64D 20 5A C1 JSR C15A ;-----
C650 A5 FB LDA FB ;Converte l'ind. e lo memorizza
C652 8D 5B C6 STA C65B ;subito dopo l'istruzione JSR
C655 A5 FC LDA FC ;
C657 8D 5C C6 STA C65C ;-----
C65A 20 00 00 JSR 0000 ;Effettua il salto all'ind.
C65D 60 RTS ;
C65E EA NOP ;
C65F EA NOP ;
C660 EA NOP ;
C661 EA NOP ;
C662 EA NOP ;
C663 EA NOP ;
C664 EA NOP ;
C665 ;
C665 ;ESA-DEC ; sintassi: $ ind
C665 ;
C665 A0 00 LDA #00 ;
C667 20 2B C1 JSR C12B ;Verifica l'ind.
C66A B9 01 02 LDA 0201,Y ;
C66D 00 04 BNE C673 ;Se c'e' qualcos'altro dopo ...
C66F A5 02 LDA 02 ;
C671 F0 03 BEQ C676 ;
C673 4C 85 C1 JMP C185 ;e/o il flag 2 e' settato, ERR
C675 20 5A C1 JSR C15A ;Conv. l'ind. e lo mem. in (B;FC)
C679 A9 23 LDA #23 ;
C67B 20 D2 FF JSR FFD2 ;Stampa '#'
C67E A9 20 LDA #20 ;
C680 20 D2 FF JSR FFD2 ;Stampa uno spazio
C683 A5 FC LDA FC ;Mette l'ind. (hi;lo) in (A;X)
C685 A6 FB LDA FB ;saranno i parametri per la B0CD,
C687 20 CD B0 JSR B0CD ;che stampa un numero decimale
C68A 60 RTS ;
C68B EA NOP ;
C68C EA NOP ;
C68D EA NOP ;
C68E EA NOP ;
C68F EA NOP ;
C690 EA NOP ;
C691 EA NOP ;
C692 ;
C692 ;DEC-ESA ; sintassi: # ind.decim.
C692 ;
C692 A9 01 LDA #01 ;Setta il flag FA. Indica che le
C694 85 FA STA FA ;cifre gia' esaminate coincidono
C696 AD 01 02 LDA 0201 ; con quelle del max (85535)

```

Seguito figura 2.

```

C699 C9 20 CMP #20 ;
C69B F0 03 BEQ C6A0 ;
C69D 4C 85 C1 JMP C185 ;Se non c'e' uno spazio, ERR
C6A0 A0 00 LDY #00 ;
C6A2 B9 02 02 LDA 0202,Y ;Prendi il car. successivo
C6A5 C9 30 CMP #30 ;
C6A7 90 F4 BCC C69D ;
C6A9 C9 3A CMP #3A ;Se non e' una cifra, ERR
C6AB 80 F0 BCS C69D ;
C6AD EA ;
C6AE A6 FA LDX FA ;Se il flag FA non e' settato, va
C6B0 F0 0C BEQ C6BE ;alla fine del ciclo di verifica
C6B2 EA NOP ;
C6B3 08 16 C7 CMP C716,Y ;Se lo e', confronta la cifra con
C6B6 90 04 BCC C6BC ; con quella del max
C6B8 00 E3 BNE C69D ;Se e' maggiore, ERR
C6BA F0 02 BEQ C6BE ;
C6BC C6 FA DEC FA ;Se e' <, FA=0. Per gli altri car.
C6BE C8 INY ;esegue solo il controllo di cifra
C6BF C0 05 CPY #05 ;
C6C1 00 DF BNE C6A2 ;Se non ha esaminato 5 car., torna
C6C3 B9 02 02 LDA 0202,Y ;
C6C6 00 D5 BNE C69D ;Se c'e' qualcos'altro dopo, ERR
C6C8 85 FB STA FB ;
C6CA 85 FC STA FC ;
C6CA ;
C6CC 20 73 00 JSR 0073 ;
C6CC F0 2E BEQ C6FF ;
C6D1 38 SEC ;
C6D2 E9 30 SBC #30 ;
C6D4 48 PHA ;
C6D5 06 FB ASL FB ;Esegue: ESA = 2 * ESA
C6D7 26 FC ROL FC ;
C6D9 A6 FB LDX FB ;
C6DB A4 FC LDY FC ;
C6DD 06 FB ASL FB ;
C6DF 26 FC ROL FC ;
C6E1 06 FB ASL FB ;
C6E3 26 FC ROL FC ;
C6E5 18 CLC ;
C6E6 8A TXA ;
C6E7 65 FB ADC FB ;Esegue: ESA = ESA + (X)Y
C6E9 85 FB STA FB ;in totale e' stato eseguito:
C6EB 98 TYA ; ESA = 10 * ESA
C6EC 65 FC ADC FC ;
C6EE 85 FC STA FC ;
C6F0 68 PLA ;
C6F1 18 CLC ;
C6F2 65 FB ADC FB ;
C6F4 85 FB STA FB ;
C6F6 A9 00 LDA #00 ;
C6F8 65 FC ADC FC ;
C6FA 85 FC STA FC ;
C6FC 18 CLC ;
C6FD 90 C0 BCC C6AC ;Torna in ciclo
C6FF A9 24 LDA #24 ;
C701 20 D2 FF JSR FFD2 ;Stampa '$
C704 A9 20 LDA #20 ;
C706 20 D2 FF JSR FFD2 ;Stampa uno spazio
C709 A5 FC LDA FC ;
C70B 20 FF C0 JSR C0FF ;
C70E A5 FB LDA FB ;Stampa ESA
C710 20 FF C0 JSR C0FF ;
C713 60 RTS ;
C714 EA ;
C715 EA ;
C716 ;
C716 ;Max: stringa del piu' grande valore decimale permesso
C716 ;
C716 '6 '5 '5 '3 '5 ;
C71B EA ;
C71C EA ;
C71D EA ;
C71E EA ;
C71F EA ;
C720 EA ;
C721 ;
C721 ;Sub. per eseguire una OPEN al drive
C721 ;
C723 A9 01 LDA #01 ;
C723 A2 02 LDX #02 ;
C725 85 B8 STA BB ;
C727 86 BC STX BC ;
C729 84 B7 STY B7 ;Sistema il puntatore al nomefile
;e la lunghezza del nomefile
C72B 20 D5 F3 JSR F3D5 ;Esegue OPEN per il bus seriale
C72E A5 BA LDA BA ;
C730 20 B4 FF JSR FFB4 ;Manda un Talk al bus seriale
C733 A5 B9 LDA B9 ;
C735 20 96 FF JSR FF96 ;Manda l'indirizzo secondario
C738 60 RTS ;
C739 EA ;
C73A EA ;
C73B EA ;
C73C EA ;
C73D EA ;
C73E EA ;
C73F EA ;
C740 EA ;
C741 ;
C741 ;DRIVE: qui sono smistati i vari comandi per il drive
C741 ;
C741 A9 08 LDA #08 ;
C743 85 BA STA BA ;Pone a 8 il n. periferica attuale
C745 AD 01 02 LDA 0201 ;
C748 00 03 BNE C74D ;Se il comando era solo >,
C74A 4C 74 C7 JMP C784 ;salta alla routine STATO
C74D C8 24 CMP #24 ;
C74F D0 03 BNE C754 ;Se il comando e' >$, salta alla
;routine DIR(ectory)
C751 4C 91 C7 JMP C791 ;
C754 C9 30 CMP #30 ;
C756 D0 03 BNE C75B ;Se il comando e' >= , salta alla

```

## COMMODORE 64

```

C758 4C F4 C7 ;routine IND(iririzzo di partenza)
C75B A0 00 ;
C75B C8 ;Altrimenti e' un comando che va
C75D ; dato cosi' com'e' al drive
C75E B9 01 02 LDA 0201,Y ;Pone in Y la lunggh. del comando
C761 D0 FA BNE C75D ;
C763 A9 6F LDA #6F ;
C765 85 B9 STA B9 ;Indirizzo secondario = 0F
C767 20 21 C7 JSR C721 ;Esegue la sub. Open
C76A 4C 74 C7 JMP C774 ;Salta alla routine STATO
C76D EA NOP ;
C76E EA NOP ;
C76F EA NOP ;
C770 EA NOP ;
C771 EA NOP ;
C772 EA NOP ;
C773 EA NOP ;
C774 ;
C774 ;STATO
C774 ; sintassi: >
C774 A0 00 LDY #00 ;
C776 A9 6F LDA #6F ;
C778 85 B9 STA B9 ;indirizzo secondario = 0F
C77A 20 21 C7 JSR C721 ;Esegue Open
C77D 20 A5 FF JSR FFA5 ;Prende un car. dal drive
C780 20 D2 FF JSR FFD2 ;e lo stampa
C783 C9 0D CMP #0D ;fino a che non trova un chr$(13)
C785 D0 F6 BNE C77D ;
C787 20 42 F6 JSR F642 ;Esegue Close
C78A 80 RTS ;
C78B EA NOP ;
C78C EA NOP ;
C78D EA NOP ;
C78E EA NOP ;
C78F EA NOP ;
C790 EA NOP ;
C791 ;DIR
C791 ; sintassi: >#[:nomefile] [*]
C791 ;
C791 A0 00 LDY #00 ;
C793 C8 ;
C793 C8 INY ;
C794 B9 01 02 LDA 0201,Y ;Pone in Y la lunggh. del comando
C797 D0 FA BNE C793 ;
C799 85 90 STA 90 ;Azzera lo Status
C79B A9 60 LDA #60 ;Indirizzo secondario = 0
C79D 85 B9 STA B9 ;Esegue la sub. Open
C79F 20 21 C7 JSR C721 ;FA contiene il n. di coppie (+1)
C7A2 A0 03 LDY #03 ;di byte da saltare in inizio riga
C7A4 84 FA STY FA ;Prende una coppia di byte in A:X
C7A4 ;
C7A4 ;
C7A6 20 A5 FF JSR FFA5 ;
C7A9 AA TAX ;
C7AA 20 A5 FF JSR FFA5 ;
C7AD A4 90 LDY 90 ;
C7AF D0 34 BNE C7E5 ;Se ST <> 0 , fine
C7B1 CE FA DEC FA ;Se no, decrem. il contat. e cont.
-----
C7B3 D0 F1 BNE C7A6 ;
C7B5 20 CD 80 JSR B0CD ;Stampa il num. decim. in (A:X)
C7B8 A9 20 LDA #20 ;(blocchi occupati) e uno spazio
C7BA 20 D2 FF JSR FFD2 ;Prende un carattere
C7BD 20 A5 FF JSR FFA5 ;
C7C0 A6 90 LDX 90 ;
C7C2 D0 21 BNE C7E5 ;Se ST <> 0 , fine
C7C4 C9 00 CMP #00 ;
C7C6 F0 14 BEQ C7DC ;Se il car. e' un chr$(0), salta
C7C8 20 E1 FF JSR FFE1 ;Se no, lo stampa
C7CE F0 15 BEQ C7E5 ;Se e' stato premuto lo STOP, fine
C7D0 20 E4 FF JSR FFE4 ;
C7D3 F0 E8 BEQ C7BD ;Se si e' premuto un tasto,attende
C7D5 20 E4 FF JSR FFE4 ;finche' non se ne preme un altro
C7D8 F0 F8 BEQ C7D5 ;
C7DA D0 E1 BNE C7BD ;Torna in ciclo (altro car.)
C7DC A9 00 LDA #00 ;
C7DE 20 D2 FF JSR FFD2 ;Stampa un [RETURN]
C7E1 A0 02 LDY #02 ;e torna in ciclo (altra linea)
C7E3 D0 BF BNE C7A4 ;
C7E5 20 42 F6 JSR F642 ;Esegue Close
C7E8 80 RTS ;
C7E9 EA NOP ;
C7EA EA NOP ;
C7EB EA NOP ;
C7EC EA NOP ;
C7ED EA NOP ;
C7EE EA NOP ;
C7EF EA NOP ;
C7F0 EA NOP ;
C7F1 EA NOP ;
C7F2 EA NOP ;
C7F3 EA NOP ;
C7F4 ;
C7F4 ;IND
C7F4 ; sintassi: >=nomefile
C7F4 ;
C7F4 A0 FF LDY #FF ;
C7F6 C8 C8 INY ;
C7F7 B9 02 02 LDA 0202,Y ;Pone in Y la lunggh. del nomefile
C7FA D0 FA BNE C7F6 ;
C7FC 85 90 STA 90 ;Azzera lo Status
C7FE 98 TYA ;
C7FF D0 03 BNE C804 ;
C801 4C 85 C1 JMP C185 ;Se non c'e' nomefile, ERR
C804 A9 60 LDA #60 ;
C806 85 B9 STA B9 ;Indirizzo secondario = 0
C808 A9 02 LDA #02 ;
C80A 20 23 C7 JSR C723 ;Esegue la sub. Open
C80D 20 A5 FF JSR FFA5 ;Prende il byte basso dell'ind.
C810 A4 90 LDY 90 ;
C812 D0 16 BNE C82A ;Se ST <> 0 , fine
C814 AA TAX ;
C815 A9 23 LDA #23 ;
C817 20 D2 FF JSR FFD2 ;Stampa '#'
C81A A9 20 LDA #20 ;e uno spazio

```

## Seguito figura 2.

```

C81C 20 D2 FF JSR FFD2 ;
C81F 20 A5 FF JSR FFA5 ;Prende il byte alto dell'in.
C822 20 CD B0 JSR B0CD ;Stampa l'ind.
C825 A9 00 LDA #00 ;e un [RETURN]
C827 20 D2 FF JSR FFD2 ;
C82A 20 42 F6 JSR F642 ;Esegue la sub. Close
C82D 60 RTS ;
C82E EA NOP ;
C82F EA NOP ;
C830 EA NOP ;
C831 ;
C831 ;TAPE RANGE sintassi: =nomefile
C831 ;
C831 A0 FF LDY #FF ;
C833 C8 INY ;
C834 B9 01 02 LDA 0201,Y ;
C837 D0 FA BNE C833 ;
C839 84 FB STY FB ;
C83B A0 00 LDA #00 ;
C83D 80 3C 03 STA 033C ;
C840 A9 02 LDA #02 ;
C842 A2 01 LDX #01 ;
C844 A0 00 LDY #00 ;e secondario
C846 20 BA FF JSR FFBA ;
C848 A2 01 LDX #01 ;
C84D A0 02 LDY #02 ;
C84F 20 BD FF JSR FFBD ;
C852 20 C0 FF JSR FFC0 ;Esegue Open
C855 A9 02 LDA #02 ;
C857 20 C3 FF JSR FFC3 ;Esegue Close
C85A A0 3C 03 LDA 033C ;
C85D C9 01 CMP #01 ;
C85F F0 04 BEQ C865 ;Se la loc. del codice progr. non
C861 C9 03 CMP #03 ;contiene i opp. 3, e quindi
C863 D0 24 BNE C889 ;non c'e' stato un 'FOUND', fine
C865 A0 00 LDY #00 ;
C867 84 FA STY FA ;
C869 A9 00 LDA #00 ;
C86B 20 D2 FF JSR FFD2 ;Stampa un [RETURN]
C86E A9 23 LDA #23 ;
C870 20 D2 FF JSR FFD2 ;Stampa '#'
C873 A9 20 LDA #20 ;
C875 20 D2 FF JSR FFD2 ;Stampa uno spazio
C878 BE 30 03 LDX 0330,X ;
C87B B9 3E 03 LDA 033E,Y ;
C87E 20 CD B0 JSR B0CD ;Prende un ind. e lo stampa
C881 A4 FA LDY FA ;
C883 C8 INY ;
C884 C8 INY ;
C885 C0 04 CPY #04 ;Se non sono stati stampati
C887 D0 DE BNE C867 ;entrambi gli ind., torna in ciclo
C889 A9 00 LDA #00 ;
C88B 20 D2 FF JSR FFD2 ;Stampa un [RETURN]
C88E 60 RTS ;
C88F EA NOP ;

C890 EA NOP ;
C891 EA NOP ;
C892 ;
C892 ;DISASSEMBLY sintassi: D ind
C892 A0 00 LDY #00 ;
C894 20 28 C1 JSR C128 ;Verifica l'ind.
C897 B9 01 02 LDA 0201,Y ;
C89A D0 04 BNE C8A0 ;e che non ci sia niente dopo
C89C A5 02 LDA #2 ;
C89E F0 03 BEQ C8A3 ;
C8A0 4C 85 C1 JMP C185 ;
C8A3 20 5A C1 JSR C15A ;Memorizza l'ind in (FB/FC)
C8A6 A5 FC LDA FC ;
C8A8 20 FF C0 JSR C0FF ;
C8AB A5 FB LDA FB ;Stampa l'ind.
C8AD 20 FF C0 JSR C0FF ;
C8B0 A9 20 LDA #20 ;
C8B2 20 D2 FF JSR FFD2 ;
C8B5 A9 20 LDA #20 ;e due spazi
C8B7 20 D2 FF JSR FFD2 ;
C8BA A0 00 LDY #00 ;
C8BC B1 FB LDA (FB),Y ;Prende il 1' byte (l'istr. in LM)
C8BE 85 FD STA FD ;lo memorizza in FD
C8C0 20 FF C0 JSR C0FF ;e lo stampa
C8C3 E6 FB INC FB ;
C8C5 D0 02 BNE C8C9 ;Esegue: ind = ind + 1
C8C7 E6 FC INC FC ;
C8C9 A6 FD LDX #0 ;
C8CB B0 CA CA LDA CACA,X ;Usa l'istr. come indice per avere
C8CE 85 FA STA FA ;il tipo d'indirizzam. dalla
C8D0 F0 2A BEQ C8FC ;tabella INDIR, e lo mem. in FA
C8D2 A9 20 LDA #20 ;
C8D4 20 D2 FF JSR FFD2 ;Stampa uno spazio
C8D7 B1 FB LDA (FB),Y ;Prende il 2' byte
C8D9 85 FE STA FE ;lo mem. in FE
C8DB 20 FF C0 JSR C0FF ;e lo stampa
C8DE E6 FB INC FB ;
C8E0 D0 02 BNE C8E4 ;Esegue: ind = ind + 1
C8E2 E6 FC INC FC ;
C8E4 A5 FA LDA FA ;
C8E6 C9 08 CMP #8 ;
C8E8 90 12 BCC C8FC ;Se Indir < 8
C8EA A9 20 LDA #20 ;(l'istruz. e' composta da 3 byte)
C8EC 20 D2 FF JSR FFD2 ;Stampa uno spazio
C8EF B1 FB LDA (FB),Y ;Prende il 3' byte
C8F1 85 FF STA FF ;lo mem. in FF
C8F3 20 FF C0 JSR C0FF ;e lo stampa
C8F6 E6 FB INC FB ;
C8F8 D0 02 BNE C8FC ;Esegue: ind = ind + 1
C8FA E6 FC INC FC ;
C8FC 18 CLC ;
C8FD A6 D6 LDX #6 ;
C8FF A0 10 LDY #10 ;Effettua un TAB(16) sfruttando
C901 20 F0 FF JSR FFF0 ;la sub. Plot del Kernal
C904 A4 FD LDY #0 ;Usa l'istr. come indice per

```

# COMMODORE 64

```

C906 B9 CA C9 LDA C9CA,Y ;trovarne il codice nella tabella
C909 A8 TAY ;CODIS. Poi usa questo codice come
C90A A2 03 LDX #03 ;indice per trovare il codice
C90B CA CB LDA CBCA,Y ;mnemonico nella tabella MNEMO
C90F 20 D2 FF JSR FFD2 ;Stampa il cod. mnem. (3 car.)
C912 C8 INY ;
C913 CA DEX ;
C914 00 F6 BNE C90C ;
C916 EA NOP ;
C917 EA NOP ;
C918 EA NOP ;
C919 EA NOP ;
C91A EA NOP ;
C91B EA NOP ;
C91C EA NOP ;
C91D EA NOP ;
C91E A9 20 LDA #20 ;
C920 20 D2 FF JSR FFD2 ;
C923 A4 FA LDY FA ;
C925 F0 61 BEQ C988 ;
C927 C0 02 CPY #02 ;
C929 F0 73 BEQ C99E ;
C92B C0 06 CPY #06 ;
C92D 90 09 BCC C938 ;
C931 B0 05 BCS C938 ;
C933 A9 28 LDA #28 ;
C935 20 D2 FF JSR FFD2 ;
C938 C0 01 CPY #01 ;
C93A D0 05 BNE C941 ;
C93C A9 23 LDA #23 ;
C93E 20 D2 FF JSR FFD2 ;
C941 C0 08 CPY #08 ;
C943 90 05 BCC C94A ;
C945 A5 FF LDA FF ;
C947 20 FF C0 JSR C0FF ;
C94A A5 FE LDA FE ;
C94C 20 FF C0 JSR C0FF ;
C94F C0 04 CPY #04 ;
C951 F0 08 BEQ C95B ;
C953 C0 06 CPY #06 ;
C955 F0 04 BEQ C95B ;
C957 C0 09 CPY #09 ;
C959 D0 0A BNE C965 ;
C95B A9 2C LDA #2C ;
C95D 20 D2 FF JSR FFD2 ;
C950 A9 58 LDA #58 ;
C962 20 D2 FF JSR FFD2 ;
C965 C0 06 CPY #06 ;
C967 90 09 BCC C972 ;
C969 C0 05 CPY #05 ;
C96B 80 05 BCS C972 ;
C96D A9 29 LDA #29 ;
C96F 20 D2 FF JSR FFD2 ;
C972 C0 05 CPY #05 ;
C974 F0 08 BEQ C97E ;
C976 C0 07 CPY #07 ;
C978 F0 04 BEQ C97E ;
C97A C0 0A CPY #0A ;
C97C D0 0A BNE C988 ;
C97E A9 2C LDA #2C ;
C980 20 D2 FF JSR FFD2 ;
C983 A9 59 LDA #59 ;
C985 20 D2 FF JSR FFD2 ;
C988 A9 0D LDA #0D ;
C98A 20 D2 FF JSR FFD2 ;
C98D A5 C6 LDA C6 ;
C98F F0 FC BEQ C98D ;
C991 20 E4 FF JSR FFE4 ;
C994 C9 0D CMP #0D ;
C996 D0 03 BNE C998 ;
C998 4C A6 C8 JMP C8A6 ;
C99B 60 RTS ;
C99C EA NOP ;
C99D EA NOP ;
C99E A5 FE LDA FE ;
C9A0 08 PHP ;
C9A1 18 CLC ;
C9A2 65 FB ADC FB ;
C9A4 85 FE STA FE ;
C9A6 A5 FC LDA FC ;
C9A8 69 00 ADC #00 ;
C9AA 85 FF STA FF ;
C9AC 28 PLP ;
C9AD 10 BPL C945 ;
C9AF C6 FF DEC FF ;
C9B1 18 CLC ;
C9B2 90 91 BCC C945 ;
C9B4 EA NOP ;
C9B5 EA NOP ;
C9B6 ;
C9B6 ;zona di memoria non usata
C9B6 ;
C9B6 ;
C9B6 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
C9C6 00 00 00 00
C9CA ;
C9CA ;Tabella CODIS
C9CA ;contiene i codici che permettono di passare da
C9CA ;un'istr. L.M. alla relativa istr. mnemonica
C9CA ;
C9CA 21 69 00 00 00 69 09 00 6F 69 09 00 69 09 00
C9DA 1E 69 00 00 69 09 00 2A 69 00 00 69 09 00
C9EA 57 06 00 00 15 06 78 00 75 06 78 00 15 06 78 00
C9FA 18 06 00 00 06 78 00 87 06 00 00 06 78 00
CA0A 7E 48 00 00 48 63 00 6C 48 63 00 54 48 63 00
CA1A 24 48 00 00 48 63 00 30 48 00 00 48 63 00
CA2A 81 03 00 00 03 78 00 72 03 78 00 54 03 78 00
CA3A 27 03 00 00 03 78 00 80 03 00 00 03 78 00
CA4A 00 90 00 00 96 90 93 00 45 00 A2 00 96 90 93 00
CA5A 0C 90 00 00 96 90 93 00 A8 90 A5 00 90 93 00
CA6A 60 5A 50 00 60 5A 50 00 9C 5A 99 00 60 5A 50 00
CA7A 0F 5A 00 00 60 5A 50 00 33 5A 9F 00 60 5A 50 00

```

Seguito figura 2.

```

C98A 3C 36 00 00 3C 36 3F 00 51 36 42 00 3C 36 3F 00
C98B 1B 34 00 00 36 3F 00 2D 36 00 00 00 36 3F 00
C98C 39 34 00 00 39 34 4B 00 4E 34 66 00 39 34 4B 00
C98D 12 84 00 00 00 84 4B 00 8A 84 00 00 84 4B 00
CACA ;
CACA ;Tabella INDIR
CACA ;contiene il tipo di indirizzam. di tutte le istr. L.M.
CACA ;
CADA 00 06 00 00 03 03 00 00 01 00 00 00 00 0B 0B 00
CADA 02 07 00 00 04 04 00 00 0A 00 00 00 00 09 09 00
CAEA 0B 06 00 00 03 03 00 00 01 00 00 0B 0B 0B 00
CAFA 02 07 00 00 04 04 00 00 0A 00 00 00 09 09 00
CB0A 00 06 00 00 03 03 00 00 01 00 00 0B 0B 0B 00
CB1A 02 07 00 00 04 04 00 00 0A 00 00 00 09 09 00
CB2A 00 06 00 00 03 03 00 00 01 00 00 0B 0B 0B 00
CB3A 02 07 00 00 04 04 00 00 0A 00 00 00 09 09 00
CB4A 00 06 00 00 03 03 00 00 00 00 00 00 0B 0B 00
CB5A 02 07 00 00 04 04 05 00 00 0A 00 00 09 09 00
CB6A 01 06 01 00 03 03 00 00 01 00 00 0B 0B 0B 00
CB7A 02 07 00 00 04 04 05 00 00 0A 00 00 09 09 00
CB8A 01 06 00 00 03 03 00 00 01 00 00 0B 0B 0B 00
CB9A 02 07 00 00 04 04 00 00 0A 00 00 00 09 09 00
CBBA 01 06 00 00 03 03 00 00 01 00 00 0B 0B 0B 00
CBBA 02 07 00 00 04 04 00 00 0A 00 00 00 09 09 00
CBCA ;
CBCA ;Tabella MNEMO
CBCA ;contiene tutte le istr. mnemoniche
CBCA ;
CBCA '- - - 'A 'D 'C 'A 'N 'D 'A 'S 'L 'B 'C 'C 'B
CBDA 'C 'S 'B 'E 'Q 'B 'I 'T 'B 'M 'I 'B 'N 'E 'B 'P
CBEA 'L 'B 'R 'K 'B 'V 'C 'B 'V 'S 'C 'L 'C 'L 'D
CBFA 'C 'L 'C 'L 'V 'C 'M 'P 'C 'P 'C 'P 'Y 'D
CBGA 'E 'C 'D 'E 'X 'D 'E 'Y 'E 'O 'R 'I 'N 'C 'I 'N
CC1A 'X 'I 'N 'Y 'J 'M 'P 'J 'S 'R 'L 'D 'A 'L 'D 'X
CC2A 'L 'D 'Y 'L 'S 'R 'N 'O 'P 'O 'R 'A 'P 'H 'A 'P
CC3A 'H 'P 'L 'A 'P 'L 'P 'R 'O 'L 'R 'O 'R 'I 'T
CC4A 'I 'R 'T 'S 'B 'S 'E 'C 'S 'E 'D 'S 'E 'I
CC5A 'S 'T 'A 'S 'T 'X 'S 'T 'Y 'T 'A 'X 'T 'A 'Y 'T
CC6A 'S 'X 'T 'X 'A 'T 'X 'S 'T 'Y 'A
CC75 ;
CC75 ;zona di memoria non usata
CC75 ;
CC75 00 00 00 00 00 00 00 00 00 00 00 00 00 00
CC80 ;
CC80 ;ASSEMBLE
CC80 ;
CC80 00 00 LDY #00 ;
CC82 20 2B C1 JSR C12B ;Verifica l'ind.
CC85 A5 02 LDA 02 ;
CC87 00 62 BNE CCEB ;Se il flag 2 e' settato, ERR
CC88 B9 01 02 LDA 0201,Y ;
CC8C 00 01 BNE CC8F ;
CC8E 60 RTS ;Se non segue altro, fine
CC8F C9 20 CMP #20 ;
CC81 00 58 BNE CCEB ;Se non c'e' uno spazio, ERR
CC93 20 5A C1 JSR C15A ;Converte l'indirizz.
CC96 A5 FB LDA FB ;
CC98 85 FD STA FD ;e lo memorizza in (FD;FE)
CC9A A5 FC LDA FC ;
CC9C 85 FE STA FE ;
CC9E A2 03 LDX #03 ;
CCA0 B9 02 02 LDA 0202,Y ;
CCA3 00 CA CB CMP CBA,X ;Confronta il codice mnemonico
CCA6 00 14 BNE CCBC ;con quello (indicizzato da X)
CCA8 B9 03 02 LDA 0203,Y ;della tabella MNEMO
CCA9 00 0C CMP CCB,X ;
CCAE 0A 0C BNE CCBC ;
CCB0 B9 04 02 LDA 0204,Y ;
CCB3 00 CC CB CMP CBC,X ;
CCB6 0A 04 BNE CCBC ;
CCB8 26 14 STX 14 ;Se coincidono,mem. in 14 l'indice
CCBA F0 09 BEQ CCC5 ;(Codis), ed esce dal ciclo
CCBC E8 INX ;
CCBD E8 INX ;Se no, Pone X al prossimo codice
CCBE E8 INX ;mmem.
CCBF E0 CPX #AB ;
CCC1 00 DD BNE CCA0 ;Se ci sono ancora codici,continua
CCC3 F0 26 BEQ CCEB ;Se la tabella e' finita, ERR
CCC5 B9 05 02 LDA 0205,Y ;
CCC8 00 03 BNE CCCC ;Se l'istr. e' finita,
CCCA 4C 05 CD JMP C0D5 ;Indir = 0 (Implicito)
CCCD C9 20 CMP #20 ;
CCCF 00 1A BNE CCEB ;Se non segue uno spazio, ERR
CCD1 98 TYA ;
CCD2 18 CLC ;
CCD3 69 06 ADC #06 ;Pone l'indice di verifica dopo il
CCD5 A8 00 TAY ;codice mnem.
CCD6 B9 00 02 LDA 0200,Y ;
CCD9 C9 23 CMP #23 ;
CCDB 00 16 BNE CCF3 ;Se non segue '#', salta
CCDD A2 02 LDX #02 ;
CCDF 20 39 C1 JSR C139 ;Se segue '#', verifica il valore
CCE2 B9 01 02 LDA 0201,Y ;
CCE5 00 04 BNE CCEB ;Se c'e' qualcos'altro dopo ...
CCE7 A5 02 LDA 02 ;
CCE9 F0 03 BEQ CCEE ;
CCEB 4C 85 C1 JMP C185 ;e/o il flag 2 e' settato, ERR
CCEE A8 01 LDA #01 ;
CCF0 4C 05 CD JMP C0D5 ;Indir = 1 (immediato)
CCF3 C9 28 CMP #28 ;
CCF5 00 5F BNE CD56 ;Se non segue '(', salta
CCF7 A2 02 LDX #02 ;
CCF9 20 39 C1 JSR C139 ;Se segue '(', verifica un valore
CCFC A5 02 LDA 02 ;
CCFE 00 EB BNE CCEB ;Se il flag 2 e' settato, ERR
CD00 C8 INY ;
CD01 B9 00 02 LDA 0200,Y ;
CD04 C9 2C CMP #2C ;
CD06 00 18 BNE CD20 ;
CD08 B9 01 02 LDA 0201,Y ;Se segue una virgola ...
CD0B C9 58 CMP #58 ;
CD0D 00 DC BNE CCEB ;

```



# COMMODORE 64

```

C00F B9 02 02 LDA 0202,Y ;e una X ...
C012 C9 29 CMP #29 ;
C014 00 05 BNE CCEB ;
C016 B9 03 02 LDA 0203,Y ;e una ) ...
C019 00 00 BNE CCEB ;
C01B A9 06 LDA #06 ;e nient'altro,
C01D 4C 05 CD JMP CDD5 ;Indir = 6 (Indicizzato Indiretto)
C020 C9 29 CMP #29 ;
C022 00 18 BNE C03C ;
C024 B9 01 02 LDA 0201,Y ;Se segue una ) ...
C027 C9 2C CMP #2C ;
C029 00 C0 BNE CCEB ;
C02B B9 02 02 LDA 0202,Y ;e una virgola ...
C02E C9 59 CMP #59 ;
C030 00 B9 BNE CCEB ;
C032 B9 03 02 LDA 0203,Y ;e una Y ...
C035 00 B4 BNE CCEB ;
C037 A9 07 LDA #07 ;e nient'altro,
C039 4C 05 CD JMP CDD5 ;Indir = 7 (Indiretto Indicizzato)
C03C A2 02 LDX #02 ;
C03E 20 3A C1 JSR C13A ;Deve esserci il 2' byte di un ind
C041 A5 02 LDA #02 ;
C043 00 A6 BNE CCEB ;Se no, ERR
C045 B9 01 02 LDA 0201,Y ;
C048 C9 29 CMP #29 ;
C04A 00 9F LDA #9F ;
C04C B9 02 02 LDA 0202,Y ;Poi una ) ...
C04F 00 9A BNE CCEB ;
C051 A9 08 LDA #08 ;e nient'altro
C053 4C 05 CD JMP CDD5 ;Indir = 8 (Indiretto)
C056 A2 02 LDX #02 ;
C058 20 3A C1 JSR C13A ;Visto che non c'era # o ( ,
C05B A5 02 LDA #02 ;deve esserci un valore
C05D 00 71 BNE CDD0 ;Se non c'è, ERR
C05F B9 01 02 LDA 0201,Y ;
C062 00 05 BNE C069 ;
C064 A9 03 LDA #03 ;Se non c'è, nient'altro,
C066 4C 05 CD JMP CDD5 ;Indir = 3 (Zero Page)
C069 C9 2C CMP #2C ;
C06B 00 1F BNE C08C ;Se non segue una virgola, salta
C06D B9 02 02 LDA 0202,Y ;
C070 C9 58 CMP #58 ;Se c'è una virgola ...
C072 00 0A BNE C07E ;
C074 B9 03 02 LDA 0203,Y ;e una X ...
C077 00 57 BNE CDD0 ;
C079 A9 04 LDA #04 ;e nient'altro,
C07B 4C 05 CD JMP CDD5 ;Indir = 4 (Zero Page,X)
C07E C9 59 CMP #59 ;
C080 00 4E BNE CDD0 ;
C082 B9 03 02 LDA 0203,Y ;Se dopo la virgola c'è una Y ...
C085 00 48 BNE CDD0 ;
C087 A9 05 LDA #05 ;e nient'altro,
C089 4C 05 CD JMP CDD5 ;Indir = 5 (Zero Page,Y)
C08C A2 02 LDX #02 ;
C08E 20 39 C1 JSR C139 ;Deve esserci il 2' byte di un ind
C091 A5 02 LDA #02 ;
C093 00 38 BNE CDD0 ;Se no, ERR
C095 B9 01 02 LDA 0201,Y ;
C098 00 18 BNE CDB2 ;Se c'è dell'altro, salta
C09A AD 07 02 LDA 0207 ;
C09D C9 42 CMP #42 ;Se non c'è altro dopo l'ind ...
C09F 00 0C BNE C0AD ;
C0A1 AD 08 02 LDA 0208 ;
C0A4 C9 49 CMP #49 ;
C0A6 F0 05 BEQ C0AD ;e l'istr. era un branch (la 1'
C0A8 A9 02 LDA #02 ;lettera e 'B e la 2' non e' 'I),
C0AA 4C 05 CD JMP CDD5 ;Indir = 2 (Relativo)
C0AD A9 08 LDA #08 ;
C0AF 4C 05 CD JMP CDD5 ;Altrimenti, Indir = 11 (Assoluto)
C0B2 C9 2C CMP #2C ;
C0B4 00 1A BNE CDD0 ;
C0B6 B9 02 02 LDA 0202,Y ;Se dopo l'ind. c'è una virgola,
C0B9 C9 58 CMP #58 ;
C0BB 00 0A BNE CDC7 ;
C0BD B9 03 02 LDA 0203,Y ;e una X ...
C0C0 00 0E BNE CDD0 ;
C0C2 A9 09 LDA #09 ;e nient'altro,
C0C4 4C 05 CD JMP CDD5 ;Indir = 9 (Assoluto,X)
C0C7 C9 59 CMP #59 ;
C0C9 00 05 BNE CDD0 ;
C0CB B9 03 02 LDA 0203,Y ;Se dopo la virgola c'è una Y ...
C0CE F0 03 BEQ CDD3 ;
C0D0 4C 85 C1 JMP C185 ;
C0D3 A9 0A LDA #0A ;e basta, Indir = 10 (Assoluto,Y)
C0D5 85 15 STA 15 ;
C0D7 F0 53 BEQ CE2C ;Memorizza Indir in 15
C0D9 20 73 00 JSR 0073 ;Se Indir=0 (impl.), salta
C0DB 20 73 00 JSR 0073 ;Pone il car. corrente dopo il
C0DD 20 73 00 JSR 0073 ;codice mmem.
C0DE AD 08 02 LDA 020B ;
C0E5 C9 3A CMP #3A ;
C0E7 10 03 BPL CDEC ;
C0E9 20 73 00 JSR 0073 ;Se c'è # o ( , avanza di 1 car
C0EC A5 15 LDA 15 ;
C0EE C9 08 CMP #08 ;
C0F0 10 03 BPL CDFB ;
C0F2 C9 02 CMP #02 ;Se dopo l'istr. c'è solo un val.
C0F4 F0 0A BEQ CE00 ;e l'istr. non e' un branch,
C0F6 20 5E C1 JSR C15E ;mem. il valore in FB
C0F9 F0 31 BEQ CE2C ;Se dopo l'istr. c'è un ind.,
C0FB 20 5A C1 JSR C15A ;mem. l'ind. in (FB:FC)
C0FE F0 2C BEQ CE2C ;Salta avanti
C000 20 5A C1 JSR C15A ;
C003 A5 FB LDA FB ;*** codifica per il branch ***
C005 E5 FD SBC FD ;
C007 STA FB ;
C009 A5 FC LDA FC ;Esegue: ind.salto - ind.istr.
C00B E5 FE SBC FE ;
C00D 20 39 FC STA FC ;
C00F 38 SEC ;
C010 A5 FB LDA FB ;

```

## COMMODORE 64

Seguito figura 2.

```

CE12 E9 02 ;Alla differenza trovata,
CE14 85 FB ;viene sottratto ancora #2
CE16 B0 02 ;Il byte basso (FB) e' il codice
CE18 C6 FC ;
CE1A E6 FC ;
CE1C D0 06 ;
CE1E A5 FB ;
CE20 10 AE ;Se e' minore di -126, ERR
CE22 30 08 ;
CE24 C6 FC ;
CE26 D0 A8 ;
CE28 A5 FB ;
CE2A 30 A4 ;Se e' maggiore di +129, ERR
CE2C A2 00 ;
CE2E A5 14 ;Pone X al 1' dato d.tabella CODIS
CE30 D0 CA C9 ;Prende Codis
CE32 D0 09 ;Lo confronta con quello d.tabella
CE34 A5 15 ;Se coincidono, confronta anche
CE36 D0 CA CA ;Indir con quello d. tabella INDIR
CE38 D0 02 ;
CE3A F0 05 ;Se OK, l'istr. Codis con il tipo
CE3C E8 ; Indir, esiste: esce dal ciclo
CE3E D0 ED ;
CE40 F0 80 ;Se e' a fine tabella, ERR
CE42 8A ;
CE44 A2 00 ;L'indice alle tabelle e' l'istr.
CE46 81 FD ;L.M., e viene posto in memoria
CE48 E6 FD ;
CE4A 00 02 ;Esegue: ind = ind + 1
CE4C E6 FE ;
CE4E A5 15 ;
CE50 F0 1A ;Se Indir=0 (impl.), salta
CE52 A5 FB ;
CE54 81 FD ;Pone in memoria il valore da FB
CE56 E6 FD ;
CE58 D0 02 ;Esegue: ind = ind + 1
CE5A E6 FE ;
CE5C A5 15 ;
CE5E C9 08 ;
CE60 30 0A ;
CE62 A5 FC ;Se l'istr. e' composta da 3 byte,
CE64 81 FD ;e' posto in memoria il 3' byte
CE66 E6 FD ;
CE68 D0 02 ;Esegue: ind = ind + 1
CE6A E6 FE ;
CE6C A9 41 ;
CE6E 20 D2 FF ;Stampa una 'A'
CE70 4C C9 C2 ;Salta alla fine di POKE per la
CE72 EA ; stampa dell'ind. e la preparaz.
CE74 EA ; ad un nuovo Input
CE76 EA ;

```



## Dalla grande edicola Jackson Tutto sull'hobby e home computer

**hc**  
HOME COMPUTER

La rivista del computer in casa. "Prove su strada" di software e programmi per tutti i personal computer.

11 numeri all'anno: L. 3.500 a numero  
Abbonamento: solo L. 31.500

**strumenti  
MUSICALI**

Il mondo delle 7 note in versione...elettronica. Con test strumentali, novità e analisi del mercato, servizi speciali.  
10 numeri all'anno: L. 3.000 a numero  
Abbonamento: solo L. 24.000

**elektor**

Il mensile di elettronica venduto in mezzo milione di copie e redatto in 7 lingue. Con articoli su: applicazioni, progettazioni, sperimentazioni, invenzioni.

11 numeri all'anno: L. 3.000 a numero  
Abbonamento: solo L. 29.000

**VIDEO  
GIOCHI**

La guida indiscussa al fantastico mondo dei videogames. La più eccitante, divertente, istruttiva rassegna del settore.

11 numeri all'anno: L. 3.500 a numero  
Abbonamento: solo L. 31.000

### Quando l'informazione fa testo

In busta chiusa inviate questo coupon a:  
Gruppo Editoriale Jackson  
via Rosellini, 12 - 20124 Mi

Desidero ricevere GRATIS un numero  
della Rivista \_\_\_\_\_

(allego L. 1.000 in francobolli per  
contributo spese di spedizione)

Inviatemi GRATIS il Catalogo della  
Biblioteca JACKSON (allego L. 1.000  
in francobolli per contributo spese di  
spedizione)

Nome \_\_\_\_\_

Cognome \_\_\_\_\_

via \_\_\_\_\_

CAP \_\_\_\_\_ Città \_\_\_\_\_

**CON INSERTO SUPERBIT**  
64 PAGINE DI SOFTWARE PER IL TUO PERSONAL

# è in edicola il nuovo numero

*Bit, la prima  
e più diffusa rivista  
di personal computer  
e accessori*



**UNA  
PUBBLICAZIONE  
DEL  
GRUPPO  
EDITORIALE  
JACKSON**

SAN FRANCISCO-LONDRA-MILANO



**BITEST: HONEYWELL  
SUPERTEAM**



SALONE  
INFORMATICA

SPECIALE COMUNICAZIONE PC-PC

FORUM P.C.

SOFTTEST: ENERGRAPHIC

IN PROVA: ARISION XL80

**B**itest  
Honeywell  
Superteam

**S**ofttest  
Energraphics

**X**L 80

**A**rision

**C**ronometriamo  
i tempi  
dei personal

**U**tility Bitdisk  
per Apple

**S**cheda  
di comunicazione  
Apple-Spectrum

**M**SX Rally

**W**ord processor  
per Apple

**G**estione C.C.  
con il C 64

**S**inc Poker

**SPECIALE:  
COMUNICAZIONE  
PC-PC**

**I**l programma qui di seguito descritto non è uno dei soliti giochi, bensì una cosa molto simpatica e che incuriosirà moltissimo anche i vostri amici, oltre ad essere sotto un certo aspetto istruttivo per le nozioni (anche se in verità non molte) che si possono apprendere. Noi crediamo che, chi più chi meno, tutti abbiamo sentito parlare del famoso Quoziente Intelligenza: che poi per impostazione o per validità risponda esattamente al valore reale di una persona, questo a voi stabilirlo, o ai molteplici trattati che esistono in materia. Una cosa è certa: test di questo tipo sono sempre più utilizzati per stabilire la preparazione culturale, la prontezza di riflessi, la capacità di adattamento e il buon senso di una persona. Tutto ciò è avvalorato dal fatto che molte aziende e numerosi concorsi pubblici si affidano a test di questo tipo per la selezione del personale da assumere ed adibire a determinate mansioni.

Il programma qui realizzato, dietro consigli di esperti e dopo aver letto libri in materia (dai quali sono state tratte) ed avendo tratto da questi delle indicazioni utili, è stato così strutturato: dopo una breve presentazione si chiede se si vogliono le istruzioni, nelle quali è spiegato in maniera molto sommaria cosa sono l'intelligenza, il quoziente intelligenza e l'età mentale. Dopo ciò verranno le istruzioni vere e proprie relative al programma e la rappresentazione di

# Quoziente intelligenza

## Una serie di test per misurare la vostra intelligenza

di Ivano Parbuono

alcuni esempi. Al termine di tutto questo, vi verrà chiesto il nome e l'età, dopo di che inizieranno le domande, che sono 50 in totale e che spaziano in vari campi, dalla conoscenza di nozioni pratiche alla storia, dalla geografia alla comprensione di certe regole matematiche, dalla prontezza di riflessi alla capacità di adattamento. Al termine delle 50 domande verrà segnalata la quantità di risposte esatte che si sono date: in base a queste verranno assegnati dei coefficienti, che, in relazione all'età di chi si è sottoposto al test e ad alcune operazioni matematiche fatte dal computer, daranno il valore della vostra età mentale e il valore del vostro quoziente intelligenza. Quando il valore del vostro

Q.I. sarà stato visualizzato si potrà confrontare con la tabella generale dei valori Q.I. che apparirà di seguito sullo schermo: si potrà così stabilire la fascia alla quale si appartiene.

I valori sono riferiti naturalmente a questo test, ma chiaramente si possono trarre delle conclusioni ben più ampie, anche se naturalmente non tutti saranno d'accordo sulla loro validità.

Passiamo ora ad un'analisi più approfondita del programma stesso, in maniera da poter rendere più chiara ed accessibile l'interpretazione delle singole righe con la possibilità di eventuali miglioramenti o traduzioni su altri tipi di computer.

Inizieremo subito con le righe che parto-

Listato 1 - Il programma Quoziente Intelligenza.

```

-100 BORDER 6: FOR P=1 TO 9 STEP
1,4: INK (RND*6): PAUSE 5
110 PLOT 48+P,120-P: DRAW 0,32:
PLOT 32+P,152-P: DRAW 32,0
115 PLOT 112+P,120-P: DRAW -24,
0: DRAW 0,32: DRAW 24,0: PLOT 88
+P,136-P: DRAW 14,0
120 PLOT 136+P,120-P: DRAW 24,0
: DRAW 0,16: DRAW -24,0: DRAW 0,
16: DRAW 24,0
125 PLOT 200+P,120-P: DRAW 0,32
: PLOT 184+P,152-P: DRAW 32,0
130 PLOT 88+P,48-P: DRAW -24,0:
DRAW 0,32: DRAW 24,0: DRAW 0,-3
2: PLOT 80+P,56-P: DRAW 16,-14
135 PLOT 160+P,48-P: DRAW 0,32:
PLOT 152+P,48-P: DRAW 16,0: PLO
T 152+P,80-P: DRAW 16,0
140 PLOT 112+P,48-P: DRAW 4,0:
PLOT 184+P,48-P: DRAW 4,0
150 NEXT P
190 PRINT AT 10,3: "0": AT 10,20:
"0": AT 20,2: BRIGHT 1: "REALIZZAT
O DA IVANO PARBUONO": PAUSE 200:

```

```

CLS
200 BORDER 7: PAPER 7: INK 1: P
RINT AT 10,4: "VUOI LE ISTRUZIONI
S/N ?"
204 POKE 23600,100
205 INPUT R$
210 IF R$="S" THEN GO SUB 5000
215 IF R$="N" THEN GO SUB 5000
220 LET L=350: LET M=500: LET E
=0: LET N=0: LET A=4: LET B=2
245 CLS: PRINT AT 2,5: BRIGHT
1: "FRA POCO INIZIAMO": PAUSE 100
250 INPUT "COME TI CHIAMI?": P$
255 PRINT AT 10,8: BRIGHT 1: FL
ASH 1: "CIAO "; P$
260 PRINT AT 10,4: FLASH 1: "QUA
NTI ANNI HAI?": INPUT K: PRINT A
T 16,22: BRIGHT 1: K: PAUSE 200
270 CLS: PRINT AT 8,10: BRIGHT
1: "OK "; P$: AT 14,2: FLASH 1:
BRIGHT 1: "SE SEI PRONTO PREMI UN
TASTO "; AT 16,10: "PER INIZIARE
": PAUSE 0
310 GO TO 700
350 INPUT " "; R
370 READ X
380 IF X<>R THEN GO TO 400

```

Seguito Ilistato Quoziente Intelligenza.

```

390 LET E=E+1: PRINT AT 16,4: I
NVERSE 1: BRIGHT 1:"ESATTO": BEE
P .05,08: BEEP .02,4: BEEP .0,2:
PAUSE 20: RETURN
400 PRINT AT 16,20: INVERSE 1:
BRIGHT 1:"ERRATO": PAUSE 20: RET
URN
500 CLS : LET N=N+1: FOR I=PI-P
I TO VAL "248" STEP VAL "8": PLO
T I,VAL "162": DRAW VAL "7",VAL
"7": PLOT I,VAL "169": DRAW VAL
"7": VAL "-7": NEXT I
610 FOR I=PI-PI TO VAL "248" ST
EP VAL "6": PLOT I,VAL "2": DRAW
VAL "7" VAL "7": PLOT I,VAL "9"
: DRAW VAL "7" VAL "-7": NEXT I:
PRINT AT 2,15: BRIGHT 1:N
615 PLOT 0,2: DRAW 0,157: PLOT
255,2: DRAW 0,157: RETURN
700 BORDER 7: PAPER 7: INK 0
705 GO SUB M: PRINT AT A,B:"QUA
LI DELLE SEGUENTI PAROLE NON
E' UNA CONGIUNZIONE": "1 = PUR
E": "2 = DUNQUE": "3 = FUORCHE":
"4 = INFATTI": GO SUB L
710 GO SUB M: PRINT AT A,B:"QUA
LE GRANDE STATISTA ITALIANOSTIPU
LO LA TRIPlice ALLEANZA CONAUSTRI
A E GERMANIA?": "1 = CRISPI": "2
= DE PRETIS": "3 = RATAZZI": "4
= GREGORETTI": GO SUB L
720 GO SUB M: PRINT AT A,B-1:"C
HI SCRISSE LA FATIDICA FRASE "
LIBERI NON SAREM SE NON SIAM UNI
": "1 = MANZONI": "2 = MAZZINI":
"3 = MERCANTINI": "4 = DANTE": GO
SUB L
730 GO SUB M: PRINT AT A,B:"
QUALE MONETA CIRCOLA
IN BRASILE?": "1 = L'ESCUDO":
"2 = IL REIS": "3 = IL CRUZIERO":
"4 = LA PESETA": GO SUB L
740 GO SUB M: PRINT AT A,B:"INS
ERISCI IL NUMERO MANCANTE": "27
-38 -32 -35 -37 - - -42":
"1 = 38": "2 = 39": "3 = 40": "4 = 4
4": "5 = 41": GO SUB L
750 GO SUB M: PRINT AT A,B:"72
E' TANTO SUPERIORE A 51 QUANTO
INFERIORE A ....?": GO SUB L
    
```

```

760 GO SUB M: PRINT AT A,B:" I
L GRUPPO DI LETTERE RPI
TPSPTRP E' UGUALE A: " "1 = RP
TIPSPTRP": "2 = RPITSPTRP": "3 =
RPITPSTPRP": "4 = RPITPSPTAP": "5
= RPIPTSPTRP": GO SUB L
770 GO SUB M: PRINT AT A,B-1:"I
L GRUPPO DI NUMERI 8866886866
E' UGUALE A: " "1 = 6866886866"
"2 = 6866886866": "3 = 6866886866"
"4 = 6866886866": "5 = 6866886
866": GO SUB L
780 GO SUB M: PRINT AT A,B:"IL
GONFARRIO DI BIASIMO": "1 = PER
DONO": "2 = MAGNANIMO": "3 = GIUST
IZIA": "4 = LODE": "5 = COMPRESIO
NE": GO SUB L
790 GO SUB M: PRINT AT A,B-1:"S
CARTARE ANAGRAMMANDO LE PAROLE C
HE SEGUONO QUELLA CHE NON I
NDICA UN FIUME": "1 = COEN": "2
= CIMINO": "3 = BLENASO": "4 = NUB
IODA": "5 = ERVETE": GO SUB L
800 GO SUB M: PRINT AT A,B-1:"S
EGNARE LA LETTERA CHE COMPLETALA
    
```

Figura 1 - La presentazione



no dalla 100 e che, aprendo un loop chiuso poi alla linea 150 permettono la scritta iniziale di presentazione in rilievo per mezzo di Plot e Draw. Alla linea 200 il programma chiede se si vogliono le istruzioni in attesa dell'Input posto alla linea 205. Se si risponde con S, sia maiuscola che minuscola, il programma - con un Gosub ci manda alla linea 5000, dove inizia la routine delle istruzioni. Alla linea 204 è stata sistemata una istruzione Poke, che fa in modo di inserire sui tasti del computer un clic sonoro: da quel momento ogni volta che si preme un tasto si sentirà un beep e ciò verrà utile in seguito, al momento in cui si dovranno dare delle risposte. Riprendiamo ora, da dove avevamo interrotto, la descrizione inerente le istruzioni, e cioè dalla linea 5000, che, fino alla 5140

comprende una routine di scorrimento orizzontale, intervallata da una lettera all'altra da un pause 5. Dalla linea 5150 fino alla linea 5180 continuano sempre le istruzioni, ma presentate in maniera diversa e cioè proiettate sul video a blocchi intervallati da pause più o meno lunghe. terminate le istruzioni, alla linea 5185 è stato sistemato un comando che permette di scrivere l'istruzione "premi un tasto per continuare" al di sotto delle normali righe: così facendo appaiono in successione sul video, per mezzo delle linee che vanno dalla 5200 alla 5230, tre esempi con le relative risposte esatte; questo per dare un'idea di come si svolgerà il test Q.I., terminando con questi esempi la parte dedicata alle istruzioni. Premendo poi un tasto, la linea 5250 (

ove è stato sistemato un Return, ci rimanderà alla linea 220) ove sono stati collocati vari Let corrispondenti a linee e righe del programma: è stato fatto ciò per risparmiare memoria. Le istruzioni che vanno dalla linea 245 alla 270 avvertono dell'inizio del programma, chiedendo il nome e l'età. Nel momento in cui si premerà un tasto inizierà il programma con le relative domande, cioè entra in azione la linea 310, che, con un Goto manda il programma alla linea 700, ove viene ristabilito tutto lo schermo bianco, mentre poi la linea 705, con un Gosub M (che come abbiamo visto alla linea 220 corrisponde a 500), invierà appunto il programma alla linea 500, dove innanzitutto verrà pulito lo schermo. Il contatore delle domande si incrementerà ogni volta di 1, mentre il For l

Seguito Ilistato Quoziente Intelligenza.

```

SERIE A-Z B-U C-U G-...
1 = R 2 = E 3 = P 4 = Q
810 GO SUB M: PRINT AT A,B-1;"O
UALE MARE ERA DETTO DAI ROMANI
* MARE INFERUM? "1 = IL T
IRRENO" 2 = L'ADRIATICO" 3 = L
O IONIO" 4 = IL MEDITERRANEO":
GO SUB L
820 GO SUB M: PRINT AT A,B;"NEL
LO STEMMMA DI QUALE CITTA' ITALI
ANA E' RAFFIGURATO UN GRIFONE?"
1 = MILANO" 2 = PERUGIA" 3
= BARI" 4 = NOVARA": GO SUB L
830 GO SUB M: PRINT AT A,B;"QUA
LE NUMERO VENIVA INDICATO DAGLI
ANTICHI ROMANI CON LA LETTERA
U SORMONTATA DA UN TRATTO ORIZ
ZONTALE?" 1 = 5.000" 2 = 10
.000" 3 = 15.000" 4 = 100.000":
GO SUB L
840 GO SUB M: PRINT AT A,B;"LA
SICILIA HA UNO SVILUPPO COST
IERO MAGGIORE DI QUELLO DELL
A SARDEGNA?" 1 = SI" 2 = NO"
3 = UGUALE": GO SUB L
850 GO SUB M: PRINT AT A,B;"NON
RIENTRA NELLA SERIE" 1 = ROM
A" 2 = PARIGI" 3 = LONDRA" 4
= ZURIGO" 5 = MADRID": GO SUB L
860 GO SUB M: PRINT AT A,B;"INS
ERISCI IL NUMERO MANCANTE
1 = 11" 2 = 12" 3 = 13" 4 = 14" 5 = 15
": GO SUB L
870 GO SUB M: PRINT AT A,B;"IND
ICARE IL NUMERO MANCANTE ALL'
INTERNO DEL TRIANGOLO" PLOT 2
4,88: DRAW 20,32: DRAW 20,-32: D
RAW -40,0: PLOT 104,88: DRAW 20,
32: DRAW 20,-32: DRAW -40,0: PLO
T 104,88: DRAW 20,32: DRAW 20,-3
2: DRAW -40,0
875 PRINT AT 8,3;"5": AT 8,7;"3"
: AT 9,4;"14": AT 11,5;"1": AT 8,13
;"9": AT 8,17;"4": AT 9,14;"25": AT
11,15;"11": AT 8,23;"7": AT 8,27;
"3": AT 9,25;"?": AT 11,25;"5"
876 PRINT AT 13,0;"1 = 24" 2 =
16" 3 = 8" 4 = 20" 5 = 12":
GO SUB L
880 GO SUB M: PRINT AT A+5,B+5;
"IN QUALE ANNO CRISTOF
ORO COLOMBO" HA SCOPERTO L'
AMERICA?" GO SUB L
890 GO SUB M: PRINT AT A,B;"QUA
LE DELLE SEGUENTI ESPRESSIONI E'
ERRATA" 1 = IO BEVO TUTTO"
2 = VIVERE CANTANDO" 3 = IO MI
MANGIO UN PANINO" 4 = GUARDATE
LA TELEVISIONE" 5 = CORRERE NEI
PRAI": GO SUB L
900 GO SUB M: PRINT AT A,B+3;"O
UALE LETTERA MANCA?"
N" 3 = L" 4 = D" 5 = Q": GO S
UB L
910 GO SUB M: PRINT AT A,B;"IND
ICARE IL NUMERO COMUNE A
LLE 4 SERIE DI NUMERI
5-3-... 5-1-9-... 8-
7-8-... 3-8-4-... "1 =
16" 2 = 12" 3 = 20" 4 = 14":
GO SUB L
920 GO SUB M: PRINT AT A,B;"IL
CONTRARIO DEL NUMERO 3" 1 = 6
REMITO" 2 = VUOTO" 3 = PIENO"
4 = META" 5 = MINIMO": GO SUB
L
930 GO SUB M: PRINT AT A,B-1;"C
HI HA SCOLPITO LA STUPENDA PORTA

```

```

DEL PARADISO DEL BATTISTERO DI
FIRENZE?" 1 = DONATELLO" 2
= Ghiberti" 3 = IACOPO DELLA QU
ERCIA": GO SUB L
940 GO SUB M: PRINT AT A,B+6;"C
YDONIA VULGARIS CHE CO
SA SI CHIAMA COSI'?" 1 = IL C
OTOGNO" 2 = IL COTONE" 3 = LA
CICOGNA" 4 = L'AQUILA": GO SUB
L
950 GO SUB M: PRINT AT A,B;"QUA
NTI SONO I VERSI DI UN SONE
TTO?" 1 = 8" 2 = 12" 3 = 14
" 4 = 16": GO SUB L
960 GO SUB M: PRINT AT A,B;"QUA
LE E' IL PIU' VASTO STATO DELL'
AFRICA?" 1 = ALGERIA" 2 = EG
ITTO" 3 = SUDAN": GO SUB L
970 GO SUB M: PRINT AT A,B;"QUA
LE NON RIENTRA NELLA SERIE?"
1 = VERDI" 2 = MOZART" 3 = BOT
TICELLI" 4 = DONIZETTI" 5 = CH
OPIN": GO SUB L
980 GO SUB M: PRINT AT A,B;"QUA
LE DI QUESTI LAGHI SI TROVA IN U
MBRIA?" 1 = MAGGIORE" 2 = BR
ACCIANO" 3 = CHIUSI" 4 = MISUR
INA" 5 = TRASIMENO": GO SUB L
990 GO SUB M: PRINT AT A,B;"QUA
LE ELEMENTO CHIMICO HA PER SIMBO
LO Hg?" 1 = L'ARGENTO" 2 =
IL MERCURIO" 3 = L'IDROGENO" 4
= L'OSSIGENO" 5 = L'ARGON": GO
SUB L
1000 GO SUB M: PRINT AT A,B-1;"N
EL DISEGNARE LA CARTINA GEOGRAFI
CA DI UNA REGIONE,OGNI TRATTOLU
NGO UN CHILOMETRO VIENE RAPPRES
ENTATO CON UN TRATTO LUNGO 20 CE
NTIMETRI."
1005 PRINT "QUANTI MILIONI DI VO
LTE LA CARTINA E' PIU' PICCOLA D
ELLA REGIONE" 1 = 10" 2 = 15
" 3 = 25" 4 = 50" 5 = 200": G
O SUB L
1010 GO SUB M: PRINT AT A,B+3;"O
UALE LETTERA MANCA?"
U-S-P-M-"E" 1 = C" 2 =
G" 3 = P" 4 = F" 5 = H": GO
SUB L
1020 GO SUB M: PRINT AT A,B;"SE
A = 2 C = 6 Z = 42
F = 12 P = ... QUALE
E' IL NUMERO CORRISPONDENTE": GO
SUB L
1030 GO SUB M: PRINT AT A,B;"DOV
E SI TROVANO LE GROTTI DI ALTAM
IRA, FAMOSE PER LE IMPORTANTI PI
TTURE PREISTORICHE?" 1 = FRAN
CIA" 2 = ITALIA" 3 = GRECIA"
4 = SPAGNA": GO SUB L
1040 GO SUB M: PRINT AT A,B;"SAN
GENNARO ERA VESCOVO
DI NAPOLI?" 1 = SI" 2 =
NO": GO SUB L
1050 GO SUB M: PRINT AT A,B;"LA
PARZIEFAN E' UNA PARTE
DELL'ATMOSFERA?" 1 = SI" 2
= NO": GO SUB L
1060 GO SUB M: PRINT AT A,B;"CHE
COS'E' CHE VIENE CHIAMATO
BIOTINA?" 1 = UNA VELA
" 2 = UNA PIANTA" 3 = UNA VITA
MINA" 4 = UN VERME": GO SUB L
1070 GO SUB M: PRINT AT A,B;"QUA
NTE PROVINCE HA LA REGIONE
TOSCANA?" 1 = 6" 2 = 6
" 3 = 9" 4 = 10" 5 = 11": GO
SUB L
1080 GO SUB M: PRINT AT A,B;"ALE
SSANDRO MANZONI SI RISPOSE DO
PO ESSERE RIMASTO VEDUO?" 1
= SI" 2 = NO": GO SUB L
1090 GO SUB M: PRINT AT A+4,B;"C

```

## ZX SPECTRUM

Seguito listato Quoziente intelligenza.

```

CONTINUARE LA SEQUENZA CON
IL NUMERO SUCCESSIVO
1-3-9-27-...": GO SUB L
1100 GO SUB M: PRINT AT A+5,B;"
CONTINUARE LA SEQUENZA"
95-83-72-62-53-...": GO SUB L
1110 GO SUB M: PRINT AT A,B;"QUA
LE E' UGUALE AL NUMERO 1": CIRC
LE 30,100,12: CIRCLE 60,90,10: C
IRCLE 100,95,13: CIRCLE 140,102,
11: CIRCLE 185,88,12: CIRCLE 230
,101,13.3
1115 PRINT AT 9,3;"1";AT 10,7;"2
";AT 9,12;"3";AT 8,17;"4";AT 10,
23;"5";AT 9,28;"6": GO SUB L
1120 GO SUB M: PRINT AT A,B;"IL
RISULTATO DI 6*10*100*0
E' UGUALE A:""1 = 6000""2
= 60000""3 = 0""4 = 60""5 = 6
00000": GO SUB L
1130 GO SUB M: PRINT AT A+5,B;"
SE 28 DONNE SU 100 FUMANO,""G
UANTE DONNE SU 600 NON FUMANO?":
GO SUB L
1140 GO SUB M: PRINT AT A+5,B-1;
"INDICARE IL NUMERO CHE COMPLETA
LA SERIE 72-65-58-51-...
": GO SUB L
1150 GO SUB M: PRINT AT A,B;"LE
ARAGOSTE NON SI DEVONO MANGIARE
IN CASA PERCHE' BISOGNACUODE
RLE VIVE IL CHE E' COME TORTU
RARLE""1 = VERO""2 = FALSO":
GO SUB L
1160 GO SUB M: PRINT AT A,B;"QUA
NTE CONIUGAZIONI ESISTONO N
ELLA LINGUA ITALIANA?""1 = 5"
"2 = 3""3 = 4""4 = 2": GO SUB
L
1170 GO SUB M: PRINT AT A,B-1;"5
CANE = 12 FINESTRA = 24
SCARPA = 18 ROSSO = 15
4""2 = FARFALLA = ""1 = 2
= 18": GO SUB L
1180 GO SUB M: PRINT AT A,B;"COM
PLETARE LA SERIE""
3 25 7 6 2 3 10
3 7 3""1 = 12""2 = 5
""3 = 22""4 = 16""5 = 18": GO
SUB L
1190 GO SUB M: PRINT AT A,B-1;"G
LI ANIMALI CHE HANNO UN NUMERO P
ARI DI DITA SONO DETTI ""1
= ARTIODATTILI""2 = PERISSODAT
TILI""3 = PTERODATTILI""4 = IS
ODATTILI": GO SUB L
3000 DATA 3,2,1,3,3,93,4,5,4,3,4
,1,2,1,1,4,4,2,1492,3,3,1,2,2,1,
3,3,3,5,2
3010 DATA 3,5,2,4,2,2,3,4,1,61,4
5,5,0,360,44,2,2,1,4,1
3400 CLS : PAUSE 50
3500 BORDER 6: PRINT AT 10,5; IN
VERSE 1; BRIGHT 1;"IL TEST E' TE
RMINATO"
3550 PAUSE 100: CLS : PRINT AT 0
,2; BRIGHT 1;"LE TUE RISPOSTE ES
ATTE SONO": INVERSE 1: PRINT AT
16,10;E;" SU 50": INVERSE 0
3560 PAUSE 200: CLS : PRINT AT 4
,0; INK 2; FLASH 1;"VALORE MEDIO
ETA' MENTALE = 800""VALORE
MEDIO QUOZ.INTELL. = 72": FLAS
H 0; BRIGHT 1;"VUOI CONOS
CERE I TUOI VALORI?": BRIGHT 0;
(Y / N)"
3600 INPUT L$
3610 IF L$="N" THEN GO SUB 9000
3615 IF L$="n" THEN GO SUB 9000

```

```

3650 GO SUB 7000
5000 LET F=0
5030 CLS : BORDER 4: PRINT AT 4,
10; BRIGHT 1;"PREMESSA"
5110 INK 2: LET A$="L'INTELLIGEN
ZA NON E' L'INSIEME DELLE COGNIZ
IONI ACQUISITE,CIO' CHE SI SA, MA
E' LA CAPACITA' CHE SI HA DI CO
NOSCERE. IL Q.I. NON E' LA
MISURA DI CIO' CHE SI E' IMPARA
TO, MA PIUTTOSTO DELLA CAPACITA'
DI IMPARARE. L'ETA' MENTALE
DI UN INDIVIDUO E' CIO' CHE IN G
ENERE UN TEST DI INTELLIGENZA HA
LO SCOPO DI DEFINIRE"
5120 FOR I=1 TO LEN A$
5130 PRINT AT 10,(32-I AND I(32)
);A$(I-31 OR I(32) TO I): BEEP .
005,35
5140 PAUSE 5: NEXT I: PAUSE 50:
CLS
5150 PRINT AT 2,10; BRIGHT 1;"IS
TRUZIONI"
5155 PRINT "" Il computer ti p
orra' delle do-mande, esattamente
e 50 alle qualianche se senza l
imiti di tempo dovrai risponde
re abbastanza velocemente." : PA
USE 200
5160 PRINT "" Le domande spaz
ieranno su vari campi della co
noscenza umana,dalla matematica
alla geografia,dalle cose prat
iche alla pron-tezza di rifless
i e saranno piu'o meno difficili
"
5162>PAUSE 200: PRINT #0;"PREMI
UN TASTO PER CONTINUARE"
5165 PRINT "" Alle domande che
ti verranno presentate dovrai
rispondere semplicemente prem
endo il numero corrispondente a
quella che se-condo te e' la ris
posta esatta seguito da ENTER":
PAUSE 250
5170 PRINT "" O dare addirittura
la risposta completa sempre seg
uita da ENTER": PAUSE 200
5175 PRINT "" Le risposte ESATTE
verranno se-gnalate visivament
e ed accompa-gnate da un BEEP.
Mentre le risposte
ERRATE verranno segnalate solo v
isivamente." : PAUSE 200
5180 PRINT "" Alla fine del TEST
ti verra' assegnato un punteggi
o determinatodal computer in bas
e a dei coe-ficienti inseriti n
el programma stesso." : PAUSE 200
5185 PRINT #0;"PREMI UN TASTO PE
R CONTINUARE" : PAUSE 0
5200 CLS : PRINT AT 2,12; FLASH
1; BRIGHT 1;"ESEMPI"
5210 PRINT AT 4,2;"NON RIENTRA N
ELLA SERIE""1 = PO""2 = NIL
O""3 = ADIGE""4 = ADDA""5 = M
INCIO":AT 16,0;"LA RISPOSTA ESAT
TA E' LA NUMERO 2"
5215 PRINT #0;"PREMI UN TASTO PE
R CONTINUARE": PAUSE 0
5218 CLS : PRINT AT 2,12; FLASH
1; BRIGHT 1;"ESEMPI"
5220>PRINT AT 4,0;"IN CHE ANNO E
' ENTRATA IN VIGORE LA COSTITUZI
ONE ITALIANA?""1 = 1945""2
= 1944""3 = 1947""4 = 1948""5
= 1949""LA RISPOSTA ESATTA
E' LA NUMERO 4"
5225 PRINT #0;"PREMI UN TASTO PE
R CONTINUARE": PAUSE 0
5228 CLS : PRINT AT 2,12; FLASH
1; BRIGHT 1;"ESEMPI"

```

Seguito Ilistato Quoziente Intelligenza.

```

5230 PRINT AT 8,0;"INDICARE IL N
UMERO CHE COMPLETA""LA SERIE
65-55-60-50-55-45-...""
L'ARISPOSTA ESATTA E' 50"
5235 PRINT #0;"PREMI UN TASTO PE
R CONTINUARE": PAUSE 0
5250 RETURN
7020 IF E<=5 THEN LET W=20
7030 IF E>5 AND E<=10 THEN LET W
=22
7035 IF E>10 AND E<=20 THEN LET
W=25
7040 IF E>20 AND E<=30 THEN LET
W=30
7045 IF E>30 AND E<=40 THEN LET
W=35
7050 IF E>40 AND E<=48 THEN LET
W=43
7055 IF E>48 AND E<=50 THEN LET
W=50
7060 IF W=20 THEN LET C=8
7065 IF W=22 THEN LET C=9
7070 IF W=25 THEN LET C=10
7075 IF W=30 THEN LET C=11
7080 IF W=35 THEN LET C=12
7085 IF W=43 THEN LET C=13
7090 IF W=50 THEN LET C=14
7100 LET Z=C*(E+W)
7110 LET H=Z-K*2
7120 LET I=H/10
7150 CLS : PRINT AT 4,5; FLASH 1
;"IL VALORE DELLA TUA "; AT 7,6;"
ETA' MENTALE E'"; BRIGHT 1;Z
7150 PAUSE 200; PRINT AT 12,4; F
LASH 1; BRIGHT 1;"IL VALORE DEL
TUO QUOZ."; AT 14,6;"INTELLIGENZA
E'"; FLASH 0; I
7170 PRINT #0;"PREMI UN TASTO PE
R CONTINUARE": PAUSE 0
7200 CLS : PRINT AT 10,4; FLASH
1;"VUOI CONOSCERE LA SCALA"; AT 1
2,9;"DEI VALORI?"; AT 14,11;"( S
/ N )": INPUT S$
7210 IF S$="n" THEN GO SUB 9000
7220 IF S$="N" THEN GO SUB 9000
7300 CLS : PRINT AT 2,4; BRIGHT
1;"SCALA DEI VALORI 0.I."
7310 PRINT AT 5,1;"valore q.i.";
AT 5,16;"valutazione"; AT 5,12;"
"; AT 7,12;"|
7320 INK 3; PLOT 8,125; DRAW 85,
0; PLOT 125,125; DRAW 67,0; INK
0
7350 PRINT AT 8,6;"> 131 | 2% EC

```

Figura 3 - Rappresentazione di una delle cinquanta domande.

INDICARE IL NUMERO MANCANTE ALL'INTERNO DEL TRIANGOLO

1 = 24  
3 = 16  
5 = 8  
7 = 20  
9 = 12

```

CEZIONALE"; AT 9,2;"120 - 130 | 4
% OTTIMO"; AT 10,2;"100 - 119 | 9
% SUPERIORE"; AT 11,2;" 80 - 99
| 21% MEDIO SUPERIORE"; AT 12,2;"
65 - 79 | 35% MEDIO"; AT 13,2;" 5
0 - 64 | 17% MEDIO INFERIORE"
7355 PRINT AT 14,2;" 30 - 49 |
6% MEDIOCRE"; AT 15,2;" 14 - 29
| 3% PESSIMO"; AT 16,2;" 0 - 13
| 1% NON HA CERVELLO"
7360 PRINT #0;"PREMI UN TASTO PE
R CONTINUARE": PAUSE 0
7370 CLS : PRINT AT 9,4; FLASH 1
;"QUALCHE ALTRO VUOLE FARE"; AT 1
0,7;"IL TEST ( S / N )": INPUT
S$
7380 IF S$="n" THEN GO SUB 9000
7390 IF S$="N" THEN GO SUB 9000
7400 CLS : GO TO 1
9000 CLS : PRINT AT 10,14; FLASH
1; BRIGHT 1;"CIAO"

```

Figura 2 - La richiesta dei dati.

FRA POCO INIZIAMO

CIAO CARLA

QUANTI ANNI HAI? 33

Figura 4 - Esempio con relativa risposta esatta.

ESEMPI

INDICARE IL NUMERO CHE COMPLETA LA SERIE 65-55-60-50-55-45-...

L'ARISPOSTA ESATTA E' 50



# ZX SPECTRUM

successivo creerà un primo tratto relativo alla cornice delle domande e le istruzioni di linea 510 e 515 creeranno il resto della cornice, terminando con un Return che rimanderà il programma alla linea 705 per permettere l'apparizione della prima domanda sul video per mezzo del Print A,B, che usa come valori quelli riportati alla linea 220. Proseguendo sempre nell'analisi della linea 705, l'istruzione Gosub L invierà il programma alla linea 350, dove è stato collocato l'Input R per le risposte da dare, mentre alla linea 370 è stata collocata l'istruzione Read X, che leggerà in sequenza progressiva i dati di linea 3000 e 3010.

La linea 380 ha il compito di confrontare il valore che Read X ha trovato con la risposta R: se il valore di X è diverso da quello di R, il programma darà la segnalazione d'errore; se invece X risulta uguale ad R, il programma proseguirà con la linea 390, il contatore delle risposte giuste s'incrementerà ed appa-

rirà in caratteri luminosi la scritta "Esatto" seguita da alcuni "Beep": dopo una brevissima pausa il programma ritornerà alla domanda successiva, situata alla linea 710, per ripetere di nuovo tutto l'iter precedente. Tutto ciò si ripete fino alla linea 1190, dove è situata l'ultima domanda, la cinquantesima. La linea 3550 comunica il totale delle risposte esatte. La linea 3580 fa apparire sullo schermo il valore medio dell'età mentale e il valore medio del quoziente intelligenza (riferito a questo test specifico). Una volta che il computer ha comunicato i valori medi vi verrà chiesto se volete conoscere i vostri valori relativi a questo test.

Le linee comprese tra la 7020 e la 7055 hanno il compito di controllare la quantità delle risposte esatte ed assegnano loro un valore, mentre le linee comprese tra la 7060 e la 7090 assegnano un coefficiente al valore precedentemente assegnato.

La linea 7100 stabilisce la vostra età

mentale, la 7110 e la 7120 determinano invece il quoziente intelligenza (Z corrisponde al valore della vostra età mentale calcolato in precedenza, K corrisponde alla vostra età reale inserita all'inizio del programma). Le linee 7150 e 7160 hanno il compito di visualizzare sul video i valori ottenuti dalle varie operazioni; la 7200 chiede se volete conoscere la scala generale dei valori, in modo da confrontarli con il vostro. Rispondendo affermativamente, verrà stampata una scala di valori riferita al Q.I.. Premendo poi un tasto qualsiasi, il video si pulisce e se nessun altro vuole fare il test, il programma termina con un "Ciao" per mezzo della linea 9000. ■

Figura 5 - Rappresentazione dei valori medi età mentale e Q.I.

```

VALORE MEDIO ETA' MENTALE = 800
VALORE MEDIO QUOZ. INTELL. = 72

VUOI CONOSCERE I TUOI VALORI?
(Y / N)
    
```

Figura 7 - La risposta esatta è la 2: Il Nilo è l'unico fiume non italiano.

```

ESEMPI
NON RIENTRA NELLA SERIE

1 == PO
2 == NILO
3 == ADIGE
4 == ADDA
5 == MINCIO

LA RISPOSTA ESATTA E' LA NUMERO 2
    
```

Figura 6 - Valori relativi al test effettuato, in questo caso, da Carla.

```

IL VALORE DELLA TUA
ETA' MENTALE E' 804

IL VALORE DEL TUO QUOZ.
INTELLIGENZA E' 73.8
    
```

Figura 8 - Rappresentazione grafica della scala generale dei valori.

SCALA DEI VALORI Q.I.

valore q.i.	valutazione
> 131	2% ECCEZIONALE
120 - 130	4% OTTIMO
100 - 119	9% SUPERIORE
80 - 99	21% MEDIO SUPERIORE
65 - 79	35% MEDIO
50 - 64	17% MEDIO INFERIORE
30 - 49	8% MEDIOCRE
14 - 29	3% PESSIMO
0 - 13	1% NON HA CERVELLO

di Sergio Borsani

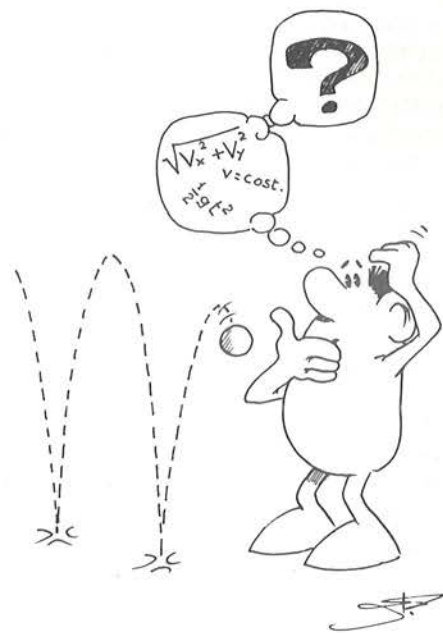
## Palla che rimbalza

**U**n giorno, a casa di un amico, ho potuto vedere alcuni giochi di provenienza USA girare su un TI 99/4A equipaggiato con armamento pesante, cioè espansione, floppy, editor/Assembler, speech synthesizer, ecc. In quell'occasione rimasi colpito da un programma che simulava il gioco del tennis. Una voce da Odissea nello Spazio scandiva i punteggi, "fifteen - thirty", "fifteen - forty", e la pallina rimbalzava in modo molto realistico. Potere degli sprite. Sono stati i protagonisti anche all'ultimo corso di BASIC da me organizzato. Non c'è niente che attragga l'attenzione e stimoli la fantasia di un ragazzo, più di uno spiritello che aleggia sullo schermo. Qualcuno è tentato di creare un gioco lì per lì, come se si trattasse di oggetti reali da manipolare con la tastiera invece che con le mani; ma le difficoltà non tardano a venire ed è bene non farsi troppe illusioni.

Il computer o il video che è il mezzo elettivo con il quale ci comunica, ci presentano un microcosmo bidimensionale dalle leggi fisiche alterate; non c'è gravità né attrito, i corpi, quando si urtano, invece di rimbalzare elasticamente si compenetrano senza deformarsi proseguendo nel loro moto inerziale. Ma che strano moto inerziale. Invece di allontanarsi indefinitamente dal punto di partenza, una volta varcato il bordo dello schermo, ricompaiono esattamente dal lato opposto. Distrarci tra i Print, gli Input e gli If-Then Else, poteva essere un'impresa ardua ma non impossibile, ma qui, per tutti i diavoli, per far saltare una pallina bisogna conoscere la geometria e la fisica, e ricreare a suon di formule il mondo che ci è familiare.

Il libro della Natura è scritto in lingua matematica, diceva Galileo Galilei, che non a caso viene qui citato per essere stato il precursore nell'enunciazione delle leggi che regolano la caduta dei gravi e che descrivono le traiettorie dei corpi che si muovono sotto l'azione delle forze di gravità.

"Proiectum, dum fertur motu composito ex horizontali aequabili et ex naturaliter accelerato deorsum, ecc." (Discorsi, IV).



Gli sprite, nel TI 99/4A, hanno la caratteristica, non comune a tutti gli altri computer, di mantenere la loro velocità automaticamente grazie ai parametri presenti nell'istruzione Call Sprite e nella Call Motion. Ad esempio, nel Commodore 64 il movimento avviene modificando continuamente i registri che controllano la posizione.

Una pallina che rimbalza si può immaginare animata contemporaneamente da due movimenti, uno uniforme lungo la direzione orizzontale ed uno uniformemente accelerato lungo la direzione verticale. Nel primo la velocità è costante, la qual cosa si ottiene facilmente nel TI 99/4A definendo una volta per tutte l'intensità della componente orizzontale della velocità. Nel listato 1 è stato posto  $VX=20$ , un valore che sposta lo sprite verso destra consentendo qualche rimbalzo prima che la pallina attraversi tutto lo schermo. La componente verticale deve variare secondo la legge  $Vy = Voy - gt$ ; se inizialmente la pallina viene lanciata verso l'alto, la sua velocità diminuisce di una quantità proporzionale al tempo. Quando, aumentando  $t$ , il prodotto  $gt$  diventa uguale a  $Voy$ , la velocità  $Vy$  risulta uguale a zero, la pallina pertanto smette di salire avendo raggiunto il punto più alto della sua traiettoria e poi comincia a scendere. Il nostro computer non possiede la funzione Time ed il passare del tempo dev'essere scandito incrementando una variabile:  $T = T + 1$ . In fisica se  $t$  è espresso in

## TEXAS TI99/4A

secondi e  $V$  in  $m/s$ , la costante  $g$  assume il valore  $9,7805$  o, come riportano la maggior parte dei testi,  $9,8$ . Noi, poiché aumentando  $T$  nel modo descritto in precedenza, non abbiamo un tempo espresso in secondi, non potremo usare questo valore di  $g$ . Nel programma 1, alla linea 150, abbiamo posto  $g=2$ . Perché abbiamo scritto  $VY+2\star T$  invece di  $VY-2\star T$  come suggerisce un confronto con la formula data in precedenza?

La ragione sta nel sistema di riferimento adottato. Normalmente l'asse verticale è orientato verso l'alto mentre il computer (non tutti) pone l'origine nell'angolo in alto a sinistra del video e pertanto orienta l'asse verticale verso il basso. A questo si rimedia attribuendo a  $VY$  valori negativi e a  $g$  valori positivi. Basta un meno o un più per farci andare a gambe all'aria.

Eseguito il programma vedrete la pallina rimbalzare da sinistra verso destra se-

guita dalla fedele ombra più in basso. A proposito, come si fa in modo che la pallina non scenda al di sotto della sua ombra? Se controlliamo la posizione dovremo farlo quando la pallina è ancora al di sopra dell'ombra, ma in tal caso il computer non distingue se lo sprite sta scendendo o se sta salendo. Conviene controllare la velocità. Se è negativa la pallina sta ancora salendo, se è positiva, allora sta scendendo. E quando sappiamo che la pallina è prossima al rimbalzo? Quando la componente verticale della velocità è vicina al valore iniziale  $VY$  con il segno cambiato; la pallina cioè cade con la stessa velocità con la quale era salita. Poiché inizialmente era  $VY=-50$ , se  $V$  diventa maggiore di  $45$  significa proprio che siamo quasi arrivati, poniamo lo sprite al centro dell'ombra con una provvidenziale Call Locate, invertiamo il segno a  $V$ , azzeriamo l'orologio e via per un altro rimbalzo.

Qualcuno osserverà giustamente che la pallina prima o dopo dovrà fermarsi; gli urti non sono perfettamente elastici, parte dell'energia viene assorbita e a ogni rimbalzo viene raggiunta una minore altezza mentre la traiettoria si fa meno ampia. La modifica del programma risulta abbastanza agevole, basta inserire un ciclo For-Next che faccia diminuire ad ogni rimbalzo la componente verticale della velocità. Nel programma Palla 2,  $VY$  viene fatta variare da  $-60$  a  $-5$  ottenendo così l'effetto desiderato senza ulteriori sostanziali cambiamenti (listato 2).

Che dire poi della palla da tennis dalla quale ha preso inizio il nostro discorso? Una pallina da tennis rimbalza continuamente da sinistra a destra e da destra a sinistra. Le modifiche rispetto al primo programma sono ancora più semplici ed invece di tante spiegazioni proponiamo una specie di quiz. Se qualcuno non trova una soluzione

## Listato 1 - Il programma Palla 1.

```
100 REM PALLA 1
110 CALL CLEAR :: CALL CHAR(128,"071F3F7
F7F"&RPT$("F",12)&"7F7F3F1F07E0F8FCFEFE"
&RPT$("F",12)&"FEFEFCF8E0")
120 CALL CHAR(132,RPT$("0",16)&"0F3F7FFF
7F3F0F"&RPT$("0",18)&"F0FCFEFFFEFCF000")
130 VX=20 :: VY=-50 :: T=1
140 CALL SPRITE(#1,128,12,150,10,VY,VX,#
2,132,2,158,10,0,VX):: CALL MAGNIFY(3)
150 T=T+1 :: V=VY+2*T :: IF V<45 THEN CA
LL MOTION(#1,V,VX):: GOTO 150
160 V=-50 :: T=1 :: CALL POSITION(#2,Y,X
):: CALL LOCATE(#1,150,X):: GOTO 150
```

## Listato 2 - Il programma Palla 2.

```
100 REM PALLA 2
110 CALL CLEAR :: CALL CHAR(128,"071F3F7
F7F"&RPT$("F",12)&"7F7F3F1F07E0F8FCFEFE"
&RPT$("F",12)&"FEFEFCF8E0")
120 CALL CHAR(132,RPT$("0",16)&"0F3F7FFF
7F3F0F"&RPT$("0",18)&"F0FCFEFFFEFCF000")
130 VX=10 :: VY=-60 :: T=1
140 CALL SPRITE(#1,128,12,150,10,VY,VX,#
2,132,2,158,10,0,VX):: CALL MAGNIFY(3)
150 FOR VY=-60 TO -5 STEP 5
160 T=T+1 :: V=VY+2*T :: IF V<ABS(VY+4) T
HEN CALL MOTION(#1,V,VX):: GOTO 160
170 V=VY :: T=1 :: CALL POSITION(#2,Y,X)
:: CALL LOCATE(#1,150,X)
180 NEXT VY :: T=1 :: GOTO 140
```

## Listato 3 - Il programma Palla 3.

```
100 REM PALLA 3
110 CALL CLEAR :: CALL CHAR(128,"071F3F7
F7F"&RPT$("F",12)&"7F7F3F1F07E0F8FCFEFE"
&RPT$("F",12)&"FEFEFCF8E0")
120 CALL CHAR(132,RPT$("0",16)&"0F3F7FFF
7F3F0F"&RPT$("0",18)&"F0FCFEFFFEFCF000")
130 VX=30 :: VY=-50 :: T=1
140 CALL SPRITE(#1,128,12,150,10,VY,VX,#
2,132,2,158,10,0,VX):: CALL MAGNIFY(3)
150 T=T+1 :: V=VY+2*T :: IF V<45 THEN 17
0
160 V=-50 :: T=1 :: CALL POSITION(#2,Y,X
):: CALL LOCATE(#1,150,X):: VX=-VX :: CA
LL MOTION(#2,0,VX)
170 CALL MOTION(#1,V,VX):: GOTO 150
```

## Listato 4 - Il programma Palla 4.

```
100 REM PALLA 4
110 CALL CLEAR :: CALL CHAR(128,"071F3F7
F7F"&RPT$("F",12)&"7F7F3F1F07E0F8FCFEFE"
&RPT$("F",12)&"FEFEFCF8E0")
120 CALL CHAR(132,RPT$("0",16)&"0F3F7FFF
7F3F0F"&RPT$("0",18)&"F0FCFEFFFEFCF000")
130 VX=60 :: VY=-50 :: T=1
140 CALL SPRITE(#1,128,12,150,10,VY,VX,#
2,132,2,158,10,0,VX):: CALL MAGNIFY(3)
150 T=T+1 :: V=VY+4*T :: IF V<45 THEN 17
0
160 V=-50 :: T=1 :: CALL POSITION(#2,Y,X
):: CALL LOCATE(#1,150,X):: VX=-VX :: CA
LL MOTION(#2,0,VX)
170 CALL MOTION(#1,V,VX):: GOTO 150
```

## TEXAS TI99/4A

soddisfacente può esaminare il listato 3, ma è invitato a farlo non prima di aver eseguito qualche tentativo.

Vogliamo infine aggiustare il tiro. Dopo tutto la risposta contenuta nel programma 3 sembra piuttosto un "pallonetto". Sarebbe ottenere un colpo teso e all'apparenza dotato di una maggiore energia? Un consiglio: aumentare VX e simulare un campo gravitazionale più intenso. Non arrendetevi subito, una possibile soluzione, è contenuta nel listato 4. ■

di Paolo Agostini

## File Examiner

**M**olto spesso succede di incontrare difficoltà nell'esaminare programmi in BASIC o in linguaggio macchina, a causa del fatto che hanno qualche sistema di protezione oppure perché nel corso del caricamento, vengono posizionati in locazioni di memoria difficilmente accessibili (Per esempio RAM posta sotto la memoria ROM), o altro.

Per ovviare a tale inconveniente, abbiamo progettato il programma Filex per il C 64 che ha le seguenti funzioni:

1) legge la directory del disco senza distur-

bare la memoria del computer;

2) legge da disco e visualizza i programmi in BASIC convertendone i "token" in parole-chiave;

3) legge da disco e disassembla i programmi in linguaggio macchina, visualizzandoli sullo schermo sotto forma di codice in linguaggio Assembly;

4) legge da disco e visualizza file numerici o di altro tipo, quali ad esempio i file di un word processor.

Il programma, proprio a causa della diversificazione dei compiti che si propone, è abbastanza lungo, e necessita della massima attenzione nella trascrizione dei dati relativi ai codici operativi. Noterete infatti la presenza, in coda ad ogni codice, di quella che viene chiamata dagli addetti ai lavori "opcode tag", cioè di una lettera che segue le tre lettere del codice operativo e che serve a specificarne il "modo":

1) codice di 3 lettere seguito da uno spazio

## Listato 1 - Il programma Filex.

```

5 POKE53280,4:POKE53281,7:PRINTCHR$(147);
  CHR$(31);"ATTENDERE PREGO"
7 FORX=1TO18:X$=X$+CHR$(32):NEXTX:GOSUB100
9 REM -----
10 REM *** MENU PRINCIPALE ***
11 REM -----
20 PRINTCHR$(147);TAB(7);CHR$(176);:FORI=
  1TO23:PRINTCHR$(192);:NEXT
30 PRINTCHR$(174)
40 PRINTTAB(7);CHR$(221);SPC(5);"FILE EXA
  MINER";SPC(5);CHR$(221)
50 PRINTTAB(7);CHR$(221);SPC(5);"DI P.AGO
  STINI";SPC(5);CHR$(221)
60 PRINTTAB(7);CHR$(173);:FORI=1TO23:PRIN
  TCHR$(192);:NEXT
70 PRINTCHR$(189)
80 PRINT:PRINT:PRINTTAB(17);CHR$(18);"MEN
  U";CHR$(146)
90 PRINT:PRINT:PRINT
100 PRINTTAB(6);CHR$(18);" (1) ";CHR$(146)
  );SPC(2);"LEGGI LA DIRECTORY"
110 PRINT:PRINTTAB(6);CHR$(18);" (2) ";CH
  R$(146);SPC(2);"LEGGI IL FILE"
130 PRINT:PRINTTAB(6);CHR$(18);" (3) ";CH
  R$(146);SPC(2);"FINE"
140 PRINT:PRINT:PRINTTAB(16);CHR$(18);"QU
  ALE?";CHR$(146)
150 GETA$:A=VAL(A$):IFA<1ORA>3THEN150
160 ONAGOTO 200,500,3200
170 END
179 REM -----
180 REM CHIUSURA CANALI APERTI
181 REM -----
190 PRINT#15,"I0":CLOSE1:CLOSE15:GOTO3100

```

```

199 REM -----
200 REM **** LEGGE LA DIRECTORY ****
201 REM -----
210 PRINTCHR$(147):PRINT"DISK NAME = ";
220 CLOSE1:CLOSE15:FL=0:L$="":B$=""
230 OPEN1,8,0,"$":OPEN15,8,15
240 INPUT#15,E1,E1$,E2,E3:IF E1 THENPRINT
  E1,E1$:GOTO190
250 FORI=1TO33:GET#1,A$
260 L$=L$+A$:NEXT:CLOSE1:PRINTL$
270 FORI=1TO40:PRINTCHR$(45);:NEXTI
280 PRINT"{RVS}BLOCKS{ 3 SPAZI}FILE NAME
  { 6 SPAZI}TIPO "
290 OPEN1,8,0,"$"
300 GET#1,A$:GET#1,A$:GOSUB350
310 GOSUB350:IF FL=1 THENPRINT:GOTO190
320 PRINTRIGHT$(X$+STR$(L),4);CHR$(32);CH
  R$(32);CHR$(32);
330 PRINTLEFT$(B$+X$,18);FT$:GETA1$:IFA1$
  <>" THEN3070
340 GOTO310
349 REM -----
350 REM SUBROUTINE LETTURA DIRECTORY
351 REM -----
360 B$="":GET#1,A$:GET#1,A$
370 GET#1,A$:L=ASC(A$+CHR$(0))
380 GET#1,A$:L=L+ASC(A$+CHR$(0))*256
390 GET#1,A$:INPUT#15,E1,E1$,E2,E3:IF E1
  THENPRINT E1,E1$:GOTO190
400 IFA$="" THENFL=1:RETURN
410 IFA$<>CHR$(34) THEN390
420 GET#1,A$
425 IFPEEK(653) THEN425
430 IFA$=CHR$(34) THEN450
440 B$=B$+A$:GOTO420
450 GET#1,A$:IFA$=CHR$(32) THEN450
460 FT$=A$

```

## COMMODORE 64

```

470 GET#1,A$:FT$=FT$+A$:GET#1,A$:FT$=FT$+
  A$
480 GET#1,A$:IFA$<>" THEN480
490 RETURN
499 REM -----
500 REM *** LETTURA DEL FILE ***
501 REM -----
510 PRINTCHR$(147);PRINTCHR$(18);"NOME DE
  L FILE (ANCHE CON *) ":PRINT
520 INPUTF$:PRINT:PRINT:IFF$=""ORF$=""TH
  EN500
530 PRINTCHR$(18);"TIPO DI FILE
  { 2 SPAZI}(PROG/SEQ/USR)":PRINT:INPUT
  T$
540 T$=LEFT$(T$,1):PRINT:PRINT:IFT$<>"S"
  HENIFT$<>"P"THEN IF T$<>"U"THEN500
545 PRINTCHR$(18);"VISUALIZZO COME NUMERI
  O LETTERE? (S/N)":PRINT:PRINT
550 GETA$:IFA$<>"S"ANDA$<>"N"THEN550
552 IFA$="S"THEN3300
555 PRINTCHR$(18);"PROGRAMMA IN {OFF} BAS
  IC {RVS} O IN {OFF} L.M {RVS} ":PRINT

560 GETA$:IFA$<>"B"ANDA$<>"L"THEN560
565 IFA$="L"THEN750
569 REM -----
570 REM ** LETTURA PROGRAMMI BASIC **
571 REM -----
580 PRINTCHR$(147);CHR$(18);"{ 2 SPAZI}**
  *{ 2 SPAZI}LETTURA PROGRAMMA IN BASIC
  { 2 SPAZI}***{ 2 SPAZI}"
585 CLOSE1:CLOSE15:OPEN15,8,15
590 OPEN1,8,5,"0:"+F$+" "+T$+"R"
600 INPUT#15,E1,E1$,E2,E3:IF E1 THENPRINT
  E1,E1$:GOTO190
610 GET#1,A$:L=ASC(A$+CHR$(0)):GET#1,A$:L
  =L+ASC(A$+CHR$(0))*256
620 PRINT"LOCAZ. DI INIZIO DEL PROGRAMMA:
  ";L
630 PRINT"(NORMALMENTE 2049 PER IL BASIC)
  ":PRINT:PRINT:PRINT
640 SL=0:GET#1,A$,A$:IFA$=""THENGOTO190
650 GET#1,A$,B$:GETA$:IFA$<>" THEN3070
655 IFPEEK(653)THEN655
660 N=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*256
  :PRINTN;
670 GET#1,A$:P=ASC(A$+CHR$(0)):IFP=0THENP
  RINT:GOTO640
680 IF(PEEK(212)<>0)OR(P<128)THENPRINTCHR
  $(P);:GOTO700
690 PRINTA$(P-128);
700 IF(A$="" OR A$="")AND(PEEK(211)>65)
  THEN730
710 IFPEEK(211)>75THEN730
720 GOTO670
730 PRINT:PRINTN;:SL=SL+1:GOTO670
740 END
749 REM -----
750 REM *** LETTURA PROGRAMMI L.M. ***
751 REM -----
760 PRINTCHR$(147);CHR$(18);"* LETTURA PR
  OGRAMMA IN CODICE MACCHINA *"
765 CLOSE1:CLOSE15:OPEN15,8,15
770 OPEN1,8,5,"0:"+F$+" "+T$+"R"

800 INPUT#15,E1,E1$,E2,E3:IF E1 THENPRINT
  E1,E1$:GOTO190
810 GET#1,A$:L=ASC(A$+CHR$(0)):GET#1,A$:L
  =L+ASC(A$+CHR$(0))*256
820 PRINTCHR$(18);"LOCAZ. DI INIZIO DEL PR
  OGRAMMA: ";L:PRINT:PRINT
830 GET#1,A$:IFA$=""THENA$=A$+CHR$(0)
835 IFPEEK(653)THEN835
840 OP=ASC(A$+CHR$(0))
850 GOSUB1500:GETA$:IFA$<>" THEN3070
860 IFST=0THEN830
870 INPUT#15,E1,E1$,E2,E3:IF E1 THENPRINT
  E1,E1$:GOTO190
880 IFST=64THEN190
890 GOTO830
900 :
910 :
920 :
930 REM *****
940 REM * FILEX *
950 REM * DI PAOLO AGOSTINI *
960 REM * V. PASCOLI, 5/B - PADOVA *
970 REM *****
980 END
989 REM -----
990 REM *** INIZIALIZZAZIONE ***
991 REM -----
1000 DIMA$(255):FORI=0TO75:READA$(I):NEXT
  I

1010 DATA END,FOR,NEXT,DATA,INPUT#,INPUT,
  DIM,READ,LET,GOTO,RUN,IF,RESTORE,GOS
  UB
1020 DATA RETURN,REM,STOP,ON,WAIT,LOAD,SA
  VE,VERIFY,DEF,POKE,PRINT#,PRINT,CONT
  ,
1030 DATA LIST,CLR,CMD,SYS,OPEN,CLOSE,GET
  ,NEW,TAB(,TO,FN,SPC(,THEN,NOT,STEP,+
  ,
1040 DATA *,/,↑,AND,OR,>,<,>,SGN,INT,ABS,
  USR,FRE,POS,SQR,RND,LOG,EXP,COS,SIN
1050 DATA TAN,ATN,PEEK,LEN,STR$,VAL,ASC,C
  HR$,LEFT$,RIGHT$,MID$,GO
1059 REM -----
1060 REM *** DATI CODICE ASSEMBLER ***
1061 REM *** ATTENZIONE AGLI SPAZI !!
1062 REM -----
1070 DATA BRK ,ORAF,?, ?, ?,ORAC,ASLC,?
1080 DATA PHP ,ORAB,ASL ,?, ?,ORAA,ASLA,?
1090 DATA BPLJ,ORAG,?, ?, ?,ORAH,ASLH,?
1100 DATA CLC ,ORAE,?, ?, ?,ORAD,ASLD,?
1110 DATA JSRA,ANDF,?, ?, ?,BITC,ANDC,ROLC,?
1120 DATA PLP ,ANDB,ROL ,?,BITA,ANDA,ROLA
  ,?
1130 DATA BMIJ,ANDG,?, ?, ?,ANDH,ROLH,?
1140 DATA SEC ,ANDE,?, ?, ?,ANDD,ROLD,?
1150 DATA RTI ,EORF,?, ?, ?,EORC,LSRC,?
1160 DATA PHA ,EORB,LSR ,?,JMPA,EORA,LSRA
  ,?
1170 DATA BVCJ,EORG,?, ?, ?,EORH,LSRH,?
1180 DATA CLI ,EORE,?, ?, ?,EORD,LSRD,?
1200 DATA RTS ,ADCF,?, ?, ?,ADCC,RORC,?
1210 DATA PLA ,ADCB,ROR ,?,JMPK,ADCA,RORA
  ,?
1220 DATA BVSJ,ADCG,?, ?, ?,ADCH,RORH,?

```

# I SEGRETI DEI PERSONAL

## Seguito listato Filex.

```

1230 DATA SEI ,ADCE,?,?,?,ADCD,RORD,?
1240 DATA ?,STAF,?,?,STYC,STAC,STXC,?
1250 DATA DEY ,?,TXA ,?,STYA,STAA,STXA,?
1260 DATA BCCJ,STAG,?,?,STYH,STAH,STXI,?
1270 DATA TYA ,STAE,TXS ,?,?,STAD,?,?,?
1280 DATA LDYB,LDAF,LDXB,?,LDYC,LDAC,LDXC
,?
1290 DATA TAY ,LDAB,TAX ,?,LDYA,LDAALDXA
,?
1300 DATA BCSJ,LDAG,?,?,LDYH,LDAH,LDXI,?
1310 DATA CLV ,LDAE,TSX ,?,LDYD,LDAD,LDXE
,?
1320 DATA CPYB,CMPE,?,?,CPYC,CMPC,DECC,?
1330 DATA INY ,CMPB,DEX ,?,CPYA,CMPA,DECA
,?
1340 DATA BNEJ,CMPG,?,?,?,CMPH,DECH,?
1350 DATA CLD ,CMPE,?,?,?,CMPD,DECD,?
1360 DATA CPXB,SBCF,?,?,CPXC,SBCC,INCC,?
1370 DATA INX ,SBCB,NOP ,?,CPXA,SBCA,INCA
,?
1380 DATA BEQJ,SBCG,?,?,?,SBCI,INCI,?
1390 DATA SED ,SBCE,?,?,?,SBCD,INCD,?
1395 DATA *
1400 DIMR$(255)
1410 FOR T = 0TO255: READ R$(T) :NEXTT
1420 READR$:IFR$(R$)<"*"THENPRINT"ERRORE NEG
LI OPCODE":END
1430 RETURN
1440 END
1499 REM -----
1500 REM STAMPA SULLO SCHERMO GLI
OPCODE ASSEMBLER
1501 REM -----
1510 OX=L:GOSUB3000:PRINT".,":OX$;
1520 PRINT"{ 2 SPAZI}";LEFT$(R$(OP),3);"
";
1530 U$=RIGHT$(R$(OP),1)
1540 IFU$=""?THENOX=OP:GOSUB3000:PRINTSPC
(5);CHR$(18);OX$:L=L+1:RETURN
1550 IFU$="" THENGOSUB1600:RETURN
1560 ONASC(U$)-64GOSUB1700,1800,1900,2000
,2100,2200,2300,2400,2500,2600,2700
1570 RETURN
1599 REM -----
1600 REM MODO IMPLICITO
1601 REM -----
1610 L=L+1:PRINT:RETURN
1690 :
1699 REM -----
1700 REM MODO ASSOLUTO
1701 REM -----
1710 PRINT" $";
1720 GET#1,A$:GET#1,B$
1730 OX=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*2
56:GOSUB3000
1740 L=L+3:PRINTOX$
1750 RETURN
1790 :
1799 REM -----
1800 REM MODO IMMEDIATO
1801 REM -----
1810 PRINT" #";
1820 GET#1,A$
1830 OX=ASC(A$+CHR$(0)):GOSUB3000
1840 L=L+2:PRINTOX$
1850 RETURN
1890 :
1899 REM -----
1900 REM MODO PAGINA ZERO
1901 REM -----
1910 PRINT" $";
1920 GET#1,A$
1930 OX=ASC(A$+CHR$(0)):GOSUB3000
1940 L=L+2:PRINTOX$
1950 RETURN
1990 :
1999 REM -----
2000 REM MODO ASSOLUTO,X
2001 REM -----
2010 PRINT" $";
2020 GET#1,A$:GET#1,B$
2030 OX=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*2
56:GOSUB3000
2040 L=L+3:PRINTOX$;" ,X"
2050 RETURN
2099 REM -----
2100 REM MODO ASSOLUTO,Y
2101 REM -----
2110 PRINT" $";
2120 GET#1,A$:GET#1,B$
2130 OX=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*2
56:GOSUB3000
2140 L=L+3:PRINTOX$;" ,Y"
2150 RETURN
2190 :
2199 REM -----
2200 REM MODO (INDIRETTO,X)
2201 REM -----
2210 PRINT" ($";
2220 GET#1,A$
2230 OX=ASC(A$+CHR$(0)):GOSUB3000
2240 L=L+2:PRINTOX$;" ,X"
2250 RETURN
2290 :
2299 REM -----
2300 REM MODO (INDIRETTO),Y
2301 REM -----
2310 PRINT" ($";
2320 GET#1,A$
2330 OX=ASC(A$+CHR$(0)):GOSUB3000
2340 L=L+2:PRINTOX$;" ,Y"
2350 RETURN
2390 :
2399 REM -----
2400 REM MODO PAG. ZERO,X
2401 REM -----
2410 PRINT" $";
2420 GET#1,A$
2430 OX=ASC(A$+CHR$(0)):GOSUB3000
2440 L=L+2:PRINTOX$;" ,X"
2450 RETURN
2490 :
2499 REM -----
2500 REM MODO PAG. ZERO,Y
2501 REM -----
2510 PRINT" $";
2520 GET#1,A$
2530 OX=ASC(A$+CHR$(0)):GOSUB3000

```

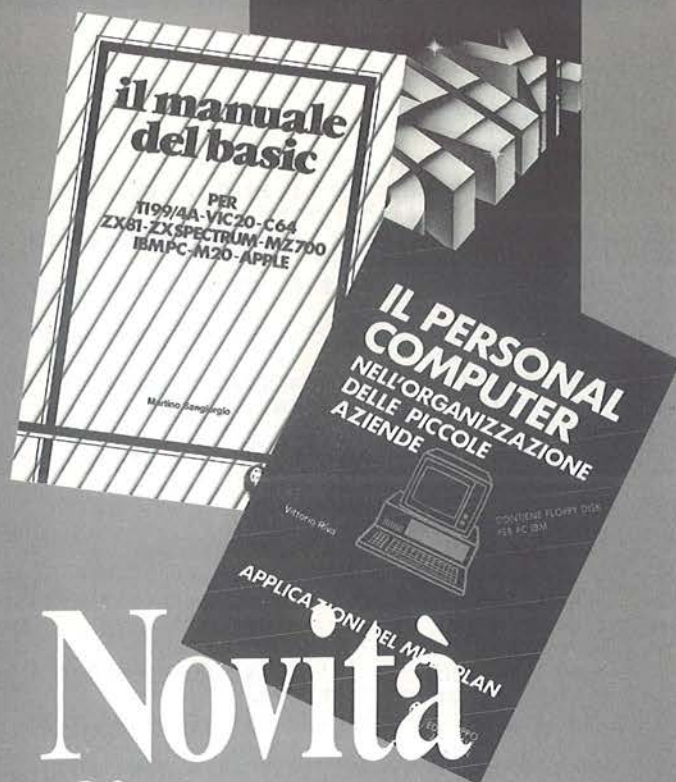
## COMMODORE 64

```

2540 L=L+2:PRINTOX$;" ,Y"
2550 RETURN
2590 :
2599 REM -----
2600 REM          SALTO RELATIVO
2601 REM -----
2610 PRINT" $";
2620 GET#1,A$
2630 OX=ASC(A$+CHR$(0))
2640 OX=OX+(OX>127)*256:OX=OX+L:GOSUB3000

2650 L=L+2:PRINTOX$
2660 RETURN
2690 :
2699 REM -----
2700 REM          SALTO INDIRETTO
2701 REM -----
2710 PRINT" ($";
2720 GET#1,A$:GET#1,B$
2730 OX=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*2
56:GOSUB3000
2740 L=L+3:PRINTOX$;" )"
2750 RETURN
2790 :
2999 REM -----
3000 REM          TRASFORMAZIONE DEC/HEX
3001 REM -----
3010 OX$="":H$="0123456789ABCDEF"
3020 IFOXTHENA=INT(OX/16):OX$=MID$(H$,1+O
X-A*16,1)+OX$:OX=A:GOTO3020
3030 IFLEN(OX$)<2THENOX$=RIGHT$("00"+OX$,
2)
3040 IF(LEN(OX$)<4)AND(LEN(OX$)>2)THENOX$
=RIGHT$("0000"+OX$,4)
3050 RETURN
3060 :
3069 REM -----
3070 REM INTERRUZIONE ROUTINE LETTURA
3071 REM -----
3080 CLOSE1:CLOSE15
3090 PRINT:PRINTCHR$(18);"*INTERRUZIONE D
ELLA{ 2 SPAZI}ROUTINE DI LETTURA*"
3099 REM -----
3100 REM          ROUTINE DI ATTESA
3101 REM -----
3110 PRINT:PRINTTAB(10);CHR$(18);" PREMER
E UN TASTO "
3120 GETA$:IFA$=""THEN3120
3130 GOTO10
3199 REM -----
3200 REM          FINE PROGRAMMA
3201 REM -----
3210 CLOSE1:CLOSE15:PRINTCHR$(147):END
3220 END
3299 REM -----
3300 REM          LETTURA NUMERICA FILE
3301 REM -----
3310 PRINTCHR$(147);CHR$(18);"LETTURA DEL
FILE IN FORMA NUMERI/LETTERE"
3320 CLOSE1:CLOSE15:OPEN15,8,15
3330 OPEN1,8,5,"0:"+F$+" ,"+T$+" ,R"
3340 INPUT#15,E1,E1$,E2,E3:IF E1 THENPRIN
T E1,E1$:GOTO190
3350 GET#1,A$:L=ASC(A$+CHR$(0)):GET#1,A$:
L=L+ASC(A$+CHR$(0))*256

```

Ambiente  
Unix

# Novità firmate Jackson.

IL PERSONAL COMPUTER  
NELL'ORGANIZZAZIONE DELLE PICCOLE  
AZIENDE: APPLICAZIONI DEL MULTIPLAN  
COD. 578P L. 29.000

IL MANUALE DEL BASIC  
COD. 534A L. 45.000

AMBIENTE UNIX  
COD. 543P L. 19.000



GRUPPO  
EDITORIALE  
JACKSON

La biblioteca che fa testo.

**Seguito listato Filex.**

```

3360 PRINT"LOCAZIONE DI INIZIO DEL PROGRA
MMA:";L:A1$="":PRINT
3370 INPUT#15,E1,E1$,E2,E3:IF E1 THENPRIN
T E1,E1$:GOTO190
3373 PRINT"SE E' UN FILE DI WORD PROCESSO
R PREMI"
3374 PRINT"IL TASTO F1, IN CASO CONTRARIO
QUALUN-"
3375 PRINT"QUE ALTRO TASTO.":PRINT:PRINT
3376 GETA1$:IFA1$=""THEN3376
3377 IFA1$=CHR$(133)THEN3550
3380 GET#1,A$:GETA1$:IFA1$<>"THEN3070
3385 IFPEEK(653)THEN3385
3390 IFST=00THEN3410
3400 IFST=64THEN190
3410 X=ASC(A$+CHR$(0)):X0$=STR$(X)
3420 X0$=RIGHT$(X0$,LEN(X0$)-1)
3430 X0$=" "+RIGHT$("000"+X0$,3)+" ":PRIN
TX0$;
3440 IFX<35ORX>127THENPRINT"{GIU' }
{ 4 SIN}{RVS}{ 3 SPAZI}{OFF}{SU} ";
3450 IFX>34ANDX<128THENPRINT"{GIU' }
{ 4 SIN}{RVS} ";CHR$(X);" {OFF}{SU}
";
3455 IFX=34THENPRINT"{GIU' }{ 4 SIN}{RVS}
' {OFF}{SU} ";

```

```

3460 IFPOS(0)>38THENPRINTCHR$(13)
3470 GOTO3380
3475 :
3480 REM LA STRINGA ALLA LINEA 3440 E' CO
MPOSTA DA: 1 CRSR DOWN + 4 CRSR LEFT
+
3490 REM 1 REVERSE ON + 3 SPAZI + 1 REVER
SE OFF+ 1 CURSOR UP + 1 SPAZIO
3500 REM LA STRINGA ALLA LINEA 3450 E' CO
MPOSTA DA: 1 CRSR DOWN + 4 CRSR LEFT
+
3510 REM 1 REVERSE ON + 1 SPAZIO;CHR$(X);
+1 SPAZIO+RVS OFF + 1 CRSR UP+1 SPAZ
IO
3520 REM LA STRINGA ALLA LINEA 3455 E' CO
ME LA RIGA ALLA LINEA 3440 CON UN
3530 REM APOSTROFO IN LUOGO DELLO SPAZIO
CENTRALE
3539 REM -----
3540 REM TRASCRIZIONE FILE IN LETTERE
3541 REM -----
3550 GET#1,A$:GETA1$:IFA1$<>"THEN3070
3560 IFPEEK(653)THEN3560
3570 IFST=00THEN3590
3580 IFST=64THEN190
3590 IFA$=CHR$(34)THENPRINT""";:GOTO3550
3600 IFA$=""THENGOTO3550
3610 IFASC(A$)<32THEN3550
3620 PRINTA$;:GOTO3550

```

vuoto: Modo implicito (operatori del tipo Brk, Cli, Rts, Sci, Dey, Txa, Tya, Txs, Tay, Tax, ecc.);

2) codice di 3 lettere seguito dalla lettera A : Modo assoluto (operatori del tipo Jmp \$C000, Lda \$C000, ecc.);

3) codice di 3 lettere seguito dalla lettera B: Modo immediato (operatori del tipo Ora #\$40, Lda #\$7F, ecc.);

4) codice di 3 lettere seguito dalla lettera C: Modo pagina zero (operatori del tipo Lda

\$FB, And \$oA, ecc.);

5) codice di 3 lettere seguito dalla lettera D : Modo assoluto, X (detto in inglese Absolute X-Indexed Addressing; operatori del tipo Sta \$Co10, X ecc.);

6) codice di 3 lettere seguito dalla lettera E: Modo assoluto, Y (detto in inglese Absolute Y-Indexed Addressing; operatori del tipo Lda \$Co10, Y ecc.);

7) codice di 3 lettere seguito dalla lettera F: Modo (indiretto, X) (operatori del tipo

Lda (\$C000, X) ecc.);

8) codice di 3 lettere seguito dalla lettera G : Modo (indiretto), Y (operatori del tipo Lda (\$C000), Y ecc.);

9) codice di 3 lettere seguito dalla lettera H : Modo pagina zero, X (operatori del tipo Lda \$40, X ecc.);

10) codice di 3 lettere seguito dalla lettera I : Modo pagina zero, Y (operatori del tipo Sta \$40, Y ecc.).

È possibile arrestare a piacimento la visua-



## Dalla grande edicola Jackson

Tutte le applicazioni  
professionali

### AUTOMAZIONE

Un'aggiornatissima panoramica delle nuove tecnologie microelettroniche e informatiche applicate all'automazione industriale.  
11 numeri all'anno: L. 3.500 a numero  
Abbonamento: solo L. 30.500

### elettronica

Il punto di riferimento più qualificato per chi voglia aggiornarsi su prodotti, applicazioni, tecnologie elettroniche, in Italia e all'estero.  
11 numeri all'anno: L. 3.500 a numero  
Abbonamento: solo L. 31.000

### telecomunicazioni

Le frontiere aperte dalla telematica, le telecomunicazioni professionali in tutti i loro settori.  
10 numeri all'anno: L. 3.500 a numero  
Abbonamento: solo L. 28.000

### INFORMATICA

La rivista professionale per chi si occupa di sistemi: dai microcomputer ai mini, ai supermini, ai mainframe. Con notizie in anteprima dall'America.  
11 numeri all'anno: L. 3.500 a numero  
Abbonamento: solo L. 31.000

### l'Elettronica

Quindicinale di politica industriale, componentistica, informatica e telecomunicazioni per uomini di marketing, responsabili acquisti, manager di settore.  
22 numeri all'anno: L. 2.500 a numero  
Abbonamento: solo L. 44.000

## Quando l'informazione fa testo

In busta chiusa inviate questo coupon a:  
Gruppo Editoriale Jackson  
via Rosellini, 12 - 20124 Mi

Desidero ricevere GRATIS un numero  
della Rivista \_\_\_\_\_

(allego L. 1.000 in francobolli per  
contributo spese di spedizione)

Inviatemi GRATIS il Catalogo della  
Biblioteca JACKSON (allego L. 1.000  
in francobolli per contributo spese di  
spedizione)

Nome \_\_\_\_\_

Cognome \_\_\_\_\_

via \_\_\_\_\_

CAP \_\_\_\_\_ Città \_\_\_\_\_



## COMMODORE 64

lizzazione dei dati sullo schermo, premendo il tasto Shift. Ciò viene ottenuto mediante un Peek alla locazione di memoria 653 che contiene il flag dei tasti Shift e Commodore. Per soste prolungate si potrà usare il tasto Shift Lock.

Si può interrompere in qualunque momento la visualizzazione della directory o di qualsiasi programma, mediante la pressione di un tasto. I programmi in BASIC si fermeranno dopo aver completato la linea in corso di visualizzazione, gli altri file invece, saranno interrotti istantaneamente. Il menu principale prevede espressamente la funzione di fine lavoro: si consiglia caldamente di utilizzare questa e non il tasto Run/Stop; in caso contrario rimarranno aperti i canali di comunicazione tra drive e computer.

Si sconsiglia altresì di leggere file che non siano originati da word processor con la funzione "lettere", in caso contrario si potrebbero produrre degli errori che condurebbero al malfunzionamento della macchina.

Per facilitare il compito a chi ricopia il programma (listato 1), consigliamo vivamente l'uso della cross-reference di figura 1, che riporta tutte le variabili utilizzate nel programma e la riga in cui appaiono.

Un ultimo appunto: quando il programma legge una riga BASIC, la cui lunghezza sia superiore alla lunghezza fisica della riga sullo schermo (cioè maggiore di 80 caratteri), esso va a capo riportando di nuovo il numero di linea all'inizio per poi continuare col resto della riga. ■

Figura 1 - Lista delle variabili usate dal programma.

VAR.	LINEA DEL PROGRAMMA									
A	150	160	3020							
A\$	150	250	260	300	360	370	380	390	400	410
	420	430	440	450	460	470	480	550	552	560
	565	610	640	650	660	670	700	810	830	840
	900	1720	1730	1820	1830	1920	1930	2020	2030	2120
	2130	2220	2230	2320	2330	2420	2430	2520	2530	2620
	2630	2720	2730	3120	3350	3380	3410	3550	3590	3600
	3610									
A\$(	690	1000								
A1\$	330	650	850	3360	3376	3377	3380	3550		
B\$	220	330	360	440	650	660	1720	1730	2020	2030
	2120	2130	2720	2730						
E1	240	390	600	800	870	3340	3370			
E1\$	240	390	600	800	870	3340	3370			
E2	240	390	600	800	870	3340	3370			
E3	240	390	600	800	870	3340	3370			
F\$	520	590	770	3330						
FL	220	310	400							
FT\$	330	460	470							
H\$	3010	3020								
I	20	60	250	270	1000					
L	320	370	380	610	620	810	820	1510	1540	1610
	1740	1840	1940	2040	2140	2240	2340	2440	2540	2640
	2650	2740	3350	3360						
L\$	220	260								
N	660	730								
OP	840	1520	1530	1540						
OX	1510	1540	1730	1830	1930	2030	2130	2230	2330	2430
	2530	2630	2640	2730	3020					
OX\$	1510	1540	1740	1840	1940	2040	2140	2240	2340	2440
	2540	2650	2740	3010	3020	3030	3040			
P	670	680	690							
R\$	1420									
R\$(	1400	1410	1520	1530						
SL	640	730								
ST	860	880	3390	3400	3570	3580				
T	1410									
T\$	530	540	590	770	3330					
U\$	1530	1540	1550	1560						
X	7	3410	3440	3450	3455					
X\$	7	320	330							
X0\$	3410	3420	3430							

## SHARP PC-1251

di Mauro Lenzi

## L'architettura del sistema

Finalmente, dopo lunghe peripezie, siamo riusciti ad entrare in possesso del manuale tecnico dello Sharp, che la ditta importatrice di questo pocket computer ci ha gentilmente fatto pervenire; presto sarà disponibile in tutti i negozi, ma per il momento esistono in circolazione pochissime copie, e crediamo siano in molti ad attenderlo ansiosamente.

Diremo innanzitutto, con un pizzico di orgoglio, che tutti i segreti che abbiamo "carpito" hanno trovato nel manuale una conferma, anzi, noi abbiamo anche trovato delle cose in più.

Il manuale infatti, da una parte soddisfa pienamente tutti coloro che desiderano dedicarsi al linguaggio macchina, descrivendo con grande precisione le varie istruzioni di disponibili, dall'altra è molto stringato nelle spiegazioni relative al firmware: manca completamente un'utilissimo disassemblato della ROM.

Comunque, se dovessimo dire che ci dispiace mentiremmo, infatti il libro contiene tutte quelle utilissime notizie che ci permetteranno di proseguire ad analizzare, pezzetto per pezzetto, il computer, ma non svela proprio tutti i segreti, altrimenti noi cosa scriveremo?

Vediamo come è strutturata la CPU.

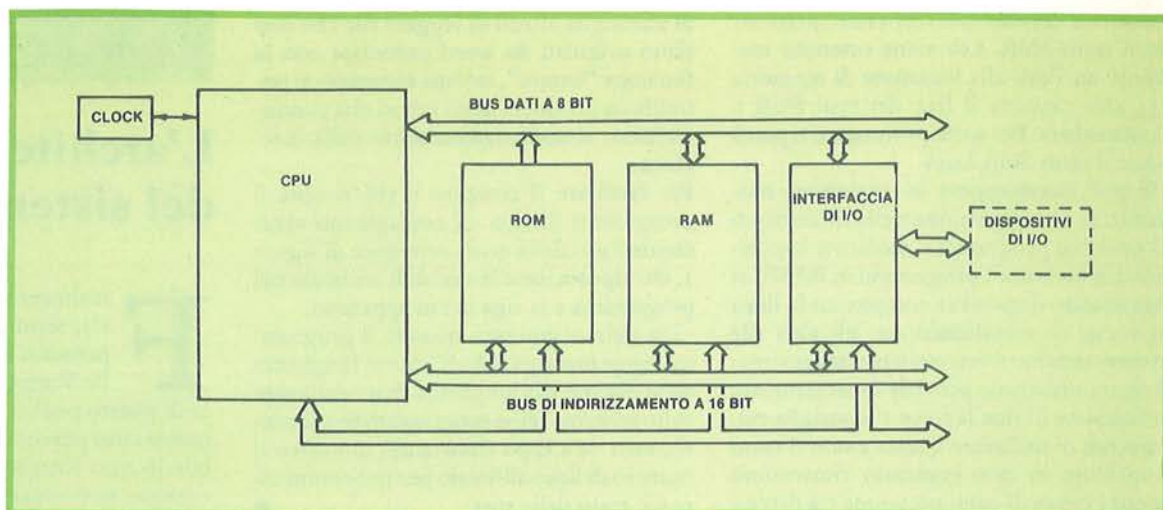
Consideriamo lo schema generale mostrato in figura 1. La CPU (Central Processing Unit) rappresenta il cuore del computer. Essa include alcuni registri interni, la ALU (Arithmetic Logic Unit), una unità di controllo che decodifica ed esegue i programmi in linguaggio macchina, una piccola RAM usata per i registri ed un sistema di stack ed I/O.

Esternamente la CPU è collegata ad un clock a cristallo di quarzo ed a tre bus per la gestione dei dati: un bus ad otto bit per la trasmissione dei dati, uno a 16 bit per l'indirizzamento ed un bus di controllo.

I dati sono mossi dalla sorgente alla destinazione col bus dei dati, byte per byte; i dati possono essere spostati dalla CPU alla memoria e viceversa, inoltre possono esse-

# I SEGRETI DEI PERSONAL

Figura 1 - Architettura di un sistema a microprocessore di tipo convenzionale.



re mandati o ricevuti all'esterno (display, stampante, microcassette, ecc.) tramite la I/O interfaccia.

Il bus a 16 bit viene usato per indirizzare i dati e il control bus coordina le varie attività del sistema.

La CPU dirige l'esecuzione di ogni coman-

do: se un dato deve essere spostato o modificato occorre prima che passi per la CPU. Vediamo finalmente come è strutturata la CPU del PC-1251.

Facciamo riferimento allo schema in figura 2. Si possono distinguere sia il bus dei dati che quello per l'indirizzamento a 16 bit,

invece non è stato rappresentato il control bus, che per altro svolge un ruolo importante solo per l'hardware del sistema.

Notiamo che all'interno del microprocessore esiste un altro bus di indirizzamento dati, che però è a sette bit: serve per indirizzare una piccola RAM interna di 96 byte.

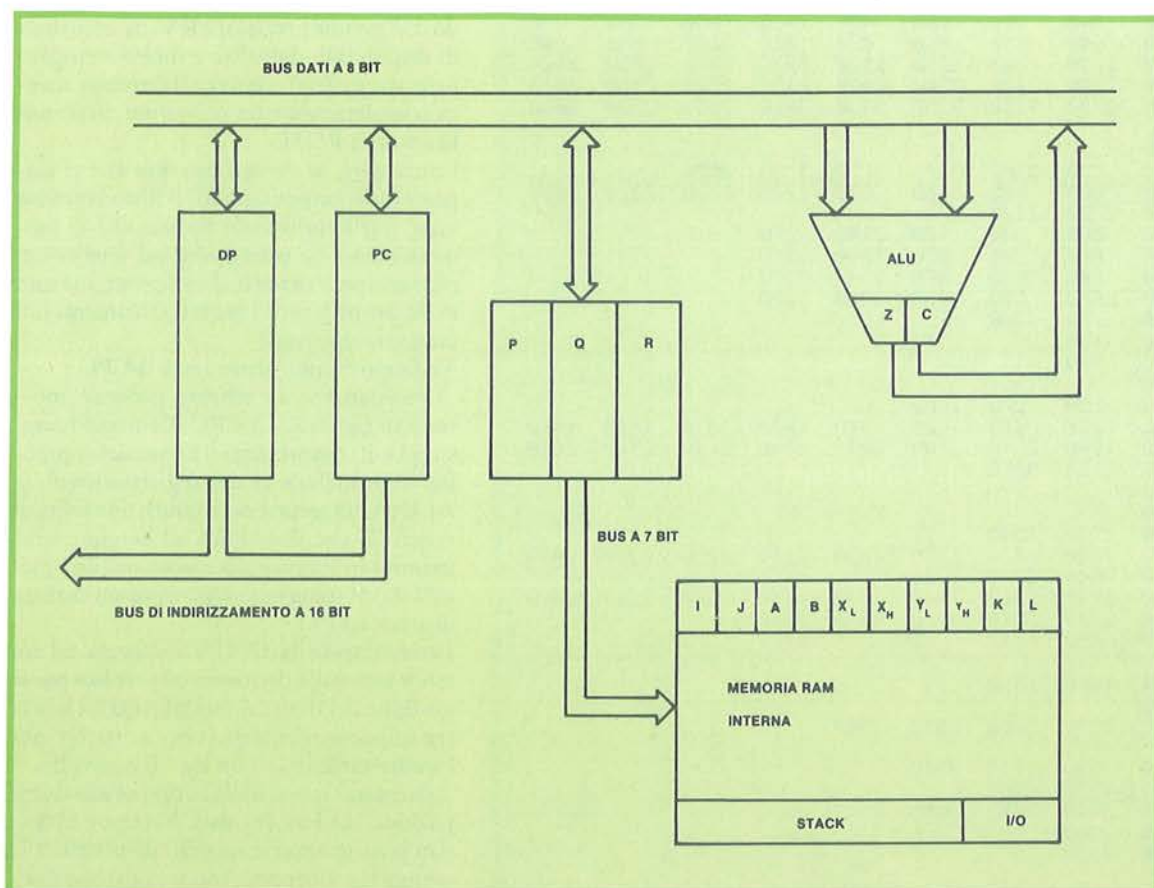


Figura 2 - Una schematizzazione della CPU del PC-1251, tratta dal Reference Manual. Anche gli Sharp-1250/1250A usano una CPU simile e sono strutturati in maniera analoga.

## SHARP PC-1251

Questa RAM non esiste in altri microprocessori come l'8080 ed il 6502, ma può ricordare vagamente la zero page di quest'ultimo.

Possiamo inoltre vedere rappresentata la ALU, che si occupa delle operazioni matematiche e logiche (And e Or), a cui sono collegati due flag, cioè registri ad un bit: il carry (C) e lo zero (Z).

Se il risultato di una operazione della ALU è zero, il flag Z è posto uguale ad uno e viceversa; il carry invece è posto ad uno se il risultato di un'operazione causa un riporto. I flag possono essere settati, cioè posti uguali ad uno o resettati, cioè azzerati anche da programma e molti salti condizionati dipendono dalla condizione in cui si trovano questi registri.

Connesso al bus a sedici bit si nota il registro DP (Data Pointer).

Si tratta di un registro estremamente utile e comodo perché permette un indirizzamento indiretto dei dati molto semplice. È un registro a due byte usato per indirizzare i dati nella memoria esterna: ogni operazione di lettura o scrittura di un dato nella memoria esterna usa questo registro. Il DP può essere incrementato, decrementato, caricato con un valore diretto oppure con i registri X e Y.

A fianco del registro DP c'è il PC (Program Counter). Anche questo è un registro a 16 bit e contiene l'indirizzo della successiva istruzione che deve essere eseguita: l'esecuzione di un programma in linguaggio macchina è ovviamente sequenziale, ma può essere modificata con istruzioni di salto del tipo Goto e Gosub che modificano il contenuto di questo registro.

Tre registri a sette bit vengono invece utilizzati per la gestione della memoria interna: P, Q e R; mentre P e Q sono normalmente usati per indirizzare i 96 byte di questa memoria, il registro R viene utilizzato prevalentemente per gestire lo stack del sistema, che è posto anch'esso nella memoria interna.

I registri I e J sono ad un byte e vengono usati come puntatori per istruzioni riguardanti il movimento di blocchi di dati: con questo microprocessore è infatti possibile spostare con un'unica istruzione un intero blocco di dati, cosa che invece richiede numerose linee di programmazione in altre note CPU come il già citato 6502!

Il registro A è l'accumulatore, usato da quasi tutte le più importanti operazioni della ALU.

Il registro B non ha nessuna particolare

predisposizione e viene di solito usato come riserva.

Esistono poi altri due registri a sedici bit, X e Y, usati comunemente come puntatori per scaricare o caricare un dato dall'accumulatore ad un indirizzo della memoria esterna.

Infine i registri K e L ad otto bit sono un po' per tutti gli usi.

Oltre a tutti questi registri esiste un registro specifico per l'I/O, posto agli indirizzi della memoria interna 5C, 5D, 5E e 5F.

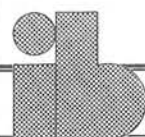
Anche lo stack è posto nella memoria interna, a partire dalla locazione 5B in giù. Il registro R punta sempre all'indirizzo più alto dello stack.

I dati possono essere immessi (Pushed) nello stack, oppure estratti (Pulled). Il primo elemento immesso nello stack è sempre nell'indirizzo più basso di questa fila di dati, l'elemento inserito più recentemente è invece in cima, ed è anche il primo ad essere tolto. Lo stack può essere usato come una sorta di schedario temporaneo di dati o per memorizzare gli indirizzi di ritorno da subroutine, cosa che per altro viene automaticamente fatta dalle istruzioni Call e Rtn (Return).

Siamo così giunti alla fine di questa rapida galoppata all'interno della CPU del PC-1251: coloro che non si intendono di linguaggio macchina non devono spaventarsi anche se hanno afferrato solo una minima parte di questo discorso, perché nei prossimi articoli avremo modo di tornare spesso su questi argomenti che finiranno per divenire familiari.

Prima di congedarci da voi qualche piccolo commento.

L'impressione che si ha trovandosi di fronte ad un microprocessore così complesso e sofisticato è estremamente favorevole: esso dispone di ben 14 registri, di cui 3 a sedici bit ed anche il set di istruzioni è ricchissimo. Anche programmatori ormai incalliti come noi non possono lasciarsi sfuggire un'espressione di meraviglia di fronte ai prodigi della tecnica: in una scatola così piccola c'è dentro tanta roba da impressionare. Ci chiediamo solo come mai sia stata prevista una RAM così piccola, mentre le potenzialità della macchina sono ben superiori. ■



INFORMATICA  
BIELLA

RIVENDITORE AUTORIZZATO

apple computer inc.



### Software

Contabilità generale 80CL Prodos  
Contabilità semplificata  
multiaziendale  
Gestione Parrocchie  
Gestione Alberghi  
Parcellazione studi legali  
Fatturazione su MAC

### Hardware

Interfacce per Olivetti  
ET 121 / 201 / 221 / 111  
Interfacce per Adler  
G 8008 SE / 1005 / 1010 / 1030



INFORMATICA  
BIELLA

VIA ROMA 11  
13051 BIELLA  
TEL. 015 - 29.875  
24.181

## Apple

Vendo **Apple II europlus** in perfette condizioni a L. 600.000. Telefonare ore pasti. Ugo Rossini - Via Cadore, 31 - 20135 Milano - Tel. 02/4553588

## Commodore

Vendo **espansione 16 Kbyte e registratore dedicato** per VIC 20. Telefonare a: Danilo Bartoli - Via Amendola, 7 - 25086 Rezzato (BS) - Tel. 2791586 (ore 19.00 - 20.00)

Vendo per Commodore 64 **programma di contabilità ordinaria**: prima nota, libro giornale, schede clienti/fornitori e conti di reddito; stampa bilancio e situazione clienti/fornitori, il tutto automaticamente. Rag. Luciano Usueli - Via Isonzo, 31 - 21100 Varese - Tel. 0332/242596

Vendo **VIC 20, datassette, 30 giochi**, alcune riviste e libri, tutto in ottime condizioni (esiste ancora la garanzia). Tutto a L. 180.000. Marilena Fontanesi - Via C. Tenca, 5 - 20124 Milano - Tel. 02/6594598

Si è aperto in Campania un **club riservato ai soli utilizzatori di C64**. L'iscrizione è gratuita. Tutti gli iscritti riceveranno ogni mese un bollettino ricco di programmi e di trucchi. Ezio Esposito - Via Bonea, 1 - 80069 Vico Equense (NA) - Tel. 081/8798584

Vendo per VIC 20 **espansione 8 Kbyte RAM, cassetta game "Sirpent" originale**, cassetta con 5 game più libro "Alla scoperta del VIC 20" e libro con 20 programmi. Il tutto a sole L. 100.000. Stefano Bonelli - Via G. Di Vittorio, 7 - 53014 Monteroni D'Arabia (SI)

■ Per CBM 64 dispongo di libri, manuali, utility su cartridge (Turbo, Fast Copy, Turbo Disco, ecc.). Marcello Cesi - Via Magliana Nuova, 178 - 00146 Roma - Tel. 06/5266009

Vendo **VIC 20 (1 anno di vita) più espansione 16 Kbyte** modificata per duplicare cartucce su nastro, 50 programmi con utility e games, super expander a L. 250.000. Tratto solo in Calabria. Angelo Lacco - Via Degli Stadi, 80 - 87100 Cosenza - Tel. 0984/31871

Vendo **VIC 20 più joystick**, "Introduzione al BASIC" parti 1 e 2, 3 cartridge, registratore a L. 300.000. Roberto Carotti - Via Benedetto Croce, 142 - 52100 Arezzo - Tel. 0575/352982

Vendo **VIC 20 più registratore**, joystick, 3 cartucce, molti programmi in cassetta, 2 libri sul VIC 20, moltissimi listati. Antonio Pavan - Via Jacopo Della Quercia, 48 - 35134 Padova - Tel. 049/612447

■ Vendo **VIC 20 più 16 Kbyte, registratore originale**, centinaia di programmi, trislot autocostruito a L. 250.000. Telefonare ora di cena. Alessandro Giolitti - Via Fabroni, 45 - 50134 Firenze - Tel. 055/473810

Vendo **VIC 20 più espansione 16 Kbyte**, super expander, cartuccia machine code language, cartucce di giochi e nastro di programmi a L. 350.000. Riccardo Bozzi - Via Risorgimento, 90 - 20092 Cinisello Balsamo (MI) - Tel. 6128851

Compro per C64 **schede aggluntive di ogni tipo**. Ferruccio Cantone - Via Genova, 18 - 10076 Nole (TO) - Tel. 011/9297125

Cambio **più di 1.500 programmi** per C64. Rispondo solo a chi vuole scambiare. Scrivi anche se non possiedi molti programmi, potresti avere quello che fa per me. Bruno Castelnuovo - Via S. Antonio, 3 - 23020 Montagna in Valtellina (SO) - Tel. 0342/380234

■ Vendo **Commodore 64 più floppy disk**, registratore a nastro, stampante, vari testi, linguaggi, diversi programmi gestionali e giochi. Telefonare ore ufficio. Mauri Renato - Via Manzoni, 1 - 20080 Zibido S. Giacomo (MI) - Tel. 9003105

Cambio **programmi per C64, C16 e Plus/4**. Dispongo di numerosi manuali. Cerco anche routines grafiche per MPS-802. Richiedere liste a: Gerardo Marra - Via P. Baratta, 91/A - 84091 Battipaglia (SA) - Tel. 0828/23130

Cerco **programma Fortran** per CBM 64. Cambio con programma a scelta tra The Last One, Logo ed altri. Telefonare ore pasti (solo zona Milano). Luca Candaten - Via Senigallia, 11 - 20161 Milano - Tel. 02/6468851

Vendo **CBM 64 più floppy, datassette, monitor colore, 4 libri**, tutto il miglior software esistente su dischetti; tutto come nuovo, usato pochissimo, disponibile per dimostrazioni a L. 1.600.000. Luigi Callegari - Via De Gasperi, 47 - 21040 Sumirago (VA) - Tel. 0331/909183

Attenzione! Causa passaggio a CBM 64, vendo **VIC 20 più registratore**, libro "Alla scoperta del VIC 20", due cassette gioco a sole L. 200.000 trattabili. Tutto in ottimo stato (comprato nel Gennaio 1985). Contatto tutta Italia. Per informazioni scrivere a: Gennaro Pinto - Via Seripando, 28 - 64100 Salerno

Vendo al miglior offerente **Commodore CBM 4032 più dual drive 4040**, il tutto in perfette condizioni e con molti programmi. Dott. Lucio Crippa - Via F. Nullo, 5 - 20035 Lissone (MI) - Tel. 039/481485

Cambio **30 videogames per C64** (One On One, Fort Apocalypse, Wimbledon, Mundial Soccer, ecc.) per Tot 13 o Magic Desk più Easy Script. Per contatti scrivere o telefonare. Salvatore Argonazza - Via S. Francesco, 45 - 87060 S. Giorgio Albanese (CS) - Tel. 0983/86112

Vendo **Commodore 64 più registratore, drive 1541, stampante MPS802**, 180 magnifici programmi, manuali vari, tutto come nuovo con imballi originali a L. 1.600.000 o separatamente. Massimo Proia - Via Pubblico Passeggio, 16 - 29100 Piacenza (PC) - Tel. 0523/32417

Vendo **VIC 20 accessoriatissimo**: Eprom, Motherboard, 3 Kbyte S.E., registratore, light pen, libri riviste, software LM; il tutto a sole L. 450.000 trattabili. Telefonare ore pasti. Massima serietà. Silvio Cappelli - Via Milano, 19 - 19100 La Spezia - Tel. 0187/36600

Vendo **VIC 20 più espansione 16 Kbyte** più 3 cartucce (VIC-Avenger, Dragonfire, Sargon II Chess), tante cassette; il tutto a L. 180.000. Rivolgersi ore serali a: Marco Della Sala - Via Francesco Ferraironi, 41/7 - 00172 Roma - Tel. 06/288472

Sono da poco un possessore di CBM 64 e sono molto interessato all'acquisto di programmi su nastro: utility, giochi e programmi vari. Prego inviarmi vostre liste. Pasquale Martire - Via Emanuele Carnevale, 1 - 90145 Palermo - Tel. 091/56393

## Sinclair

Cerco **Sinclair ZX80 con vecchia o nuova ROM** e schema elettrico dello ZX80. Telefonare solo tra le 12.30 e le 13.00 o tra le 19.30 e 20.30. Aldo V�dramin - Via Chiaradia, 3 - 33077 Sacile (PN) - Tel. 0434/733139

Disponendo di 1.200 programmi per ZX Spectrum **cerco utenti disposti allo scambio Spectrum - Commodore 64**. Annuncio sempre valido. Alfredo Trifiletti - Via Fiume, 20/A - 71100 Foggia - Tel. 0881/75385

Vendo **ZX Spectrum 48 Kbyte più 100 programmi**, interfaccia joystick programmabile a L. 350.000 trattabili. Inoltre vendo programmi per Apple IIe - Il plus. Stefano Sparvoli - Via Passo Rolle, 53 - 20134 Milano - Tel. 2151496

Causa passaggio a sistema superiore vendo agli "apprendisti" di BASIC **Sinclair ZX81 più alimentatore**, manuale, espansione 16 Kbyte, programmi su cassetta a sole L. 70.000! Scrivere o telefonare. Antonello Carbone - Via Adone, 5 - 95040 S.G. Galermo (CT) - Tel. 095/393046

Vendo **Spectrum 48 Kbyte** a L. 300.000; 350 programmi L. 150.000; interfaccia joystick L. 50.000; registratore L. 50.000. In blocco o separatamente; telefonare ore pasti, solo zona Roma. Giuseppe Fasulo - Via F. Berni, 5 - 00185 Roma - Tel. 7577351

Vendo **Spectrum 48 Kbyte con manuali**, registratore, libro pr LM, 2 libri di grafica, light pen, cassette giochi, giornali. Acquisito nel Dicembre 1984, tutto in ottime condizioni a L. 400.000. Mauro Pavone - Via Capocelatro, 30 - 20148 Milano - Tel. 02/4075792

Vendo **Sinclair ZX80** a L. 100.000. Andrea Roidi - Viale Augusto, 105 - 80125 Napoli - Tel. 081/616837

**Quantum Leap User Club cerca nuovi soci in tutta Italia**. Iscrizione gratuita. Abbiamo già a disposizione libri, software e tanti consigli e idee. Roberto Ghezzi - Via Volontari Del Sangue, 202 - 20099 Sesto S. Giovanni (MI) - Tel. 02/2485511

Desidero comunicare con possessori di Sinclair QL per scambio informazioni, chiarimenti, ecc. Ulderico Guadagno - Via Tito Livio, 95 - 00136 Roma - Tel. 06/3492694

Vendo ZX Spectrum 48 Kbyte più alimentatore stabilizzato, stampante Alphacom 32, numerosi programmi, tutto a L. 450.000. Telefonare ore pasti. Giuseppe Pingitore - Corso G. Nicotera, 137 - 88046 Lamezia Terme (CZ) - Tel. 0968/23816

Compro, cambio software per QL. Richiedere e/o inviare lista. Andrea Galli - Via Palagetta, 212 - 50017 S. Piero a Ponti (FI)

## Texas

Vendo SSS Extended BASIC per TI 99/4A più manuale, corso su cassetta a L. 170.000; 3 cassette TI 99 Newssoft a L. 20.000; 4 cassette originali Texas Instruments a L. 35.000 (valore L. 70.000); cavetto registratore a L. 10.000. Giampaolo Badiali - Via C. Pisacane, 12 - 60033 Chiaravalle (AN) - Tel. 071/946311

Compro Minimemory SSS più cassetta Lines, Sintetizzatore Vocale, Terminal Emulator solo se totalmente funzionanti e convenienti. Tratto con le zone di Torino e Milano. Telefonare ore serali, dopo le 19.00. Nicola Passalacqua - Via Onorato Vigliani, 15/4 - 10135 Torino - Tel. 011/6192252

Vendo TI 99/4A, box disk controller TI, Extended BASIC, TI Logo; SSS dati pers., statistica, scacchi, aiuto programmazione; SSS didattici ed aritmetici; manuali, accessori a L. 500.000 trattabili oppure separatamente. Agostino Micheloni - Via A. Provolo, 20 - 37123 Verona - Tel. 045/21419

Cerco modulo Extended BASIC per TI 99/4A. Possibilmente con manuale d'uso. Giovanni Arcadi - Via Enrico Fermi, 7 - 89048 Siderno (RC) - Tel. 0964/341461

Compro per TI 99/4A modulo SSS Soccer ad un prezzo trattabile ed inoltre cambio un sacco di giochi, veramente belli, sia su nastro che listato. Scrivetemi presto e a tutta forza! Claudio Faneli - Via Leopoldo Bignone, 27/5 - 16157 Ge-Pra (GE) - Tel. 010/663998

Compro per TI 99/4A coppia di joystick. Inoltre possiedo un TI 99/4A non funzionante. Cerco laboratorio per eventuale riparazione. Chi può fornirmi indirizzi utili mi scriva. Luigi Garofoli - Via Ospedale, 29 - 60011 Arcevia (AN)

Per TI 99/4A compro interfaccia RS232 perfettamente funzionante. Scrivere o telefonare ore serali. Alessandro Garoli - Corso Dante, 35 - 12100 Cuneo - Tel. 65265

Compro manuale per Extended BASIC scritto in italiano. Massimiliano Bruno Ventre - Via De Galboli, 7 - 28100 Novara - Tel. 0321/478356

## Varie

Per Olivetti M20 vendo programma di word processing con molte funzioni fatto da me. Dispone anche dell'incollamento dei testi automatico. Telefonare ore pomeridiane a: Antonio Moro - Via A. De Gasperi, 5G - 36022 Cassola (VI) - Tel. 0424/83027

Vendo MPF II 64 Kbyte RAM, Applesoft compatibile, imballo originale con vari programmi, linguaggi Applesoft e M. BASIC, manuali, a L. 300.000. Umberto Torrini - Via Bolognese, 57 - 50139 Firenze - Tel. 055/474836

Vendo console Intellevison, computer Lucky completo di adattatore e trasformatore, 12 cassette fra cui Nova Blast, Soccer, Scooby Doo's a L. 600.000 trattabili. Luca Bruscoli - Via Arno, 9 - 30026 Portogruaro (VE) - Tel. 0421/72956

Vendo box, disk drive, disk controller, disk manager, il tutto garantito, per L. 1.500.000 non trattabili. Solo zone di Milano, Bergamo, Lecco. Telefonare di domenica. Carlo Aceti - Via I Maggio, 4 - 92050 Verderio Inferiore (CO) - Tel. 039/510715

Vendo Sega SC-3000 16 Kbyte espandibili, alimentatore stabilizzato, cavetti, manuale d'istruzioni, cartuccia BASIC, Star Jaker, Congo Bongo, 2 joystick semiprofessionali a L. 350.000 (valore effettivo L. 750.000). Marco Moniga - Via Collecchio, 16/8 - 20148 Milano - Tel. 02/395140

Vendo per N.E. Z80 o Micro Design configurazione CP/M le schede aggiuntive "Computer Parlante" e "Grafica Sovrapposta" con relativo software. Eseguo schede personalizzate. Marco Sudetti - Via Martiri della Libertà, 52 - 10040 Borgaretto (TO) - Tel. 3580200

Vendo Sega SC-3000 32 Kbyte ancora in garanzia, perfetto, con circa 150 programmi a L. 35.000 non trattabili. Guglielmo Nervegna - Via Risorgimento, 273 - 47100 Forlì - Tel. 0543/84006

Compro hardware rotti, danneggiati, inutilizzabili, dei più conosciuti home computer. Annuncio sempre valido. Gian Luigi Calzolaro - Via M. della Benedicta, 3/12 - 16010 Rossiglione (GE) - Tel. 010/925447

Vendo Apple compatibile orientato alla grafica, completo di monitor, drive, video digitizer con telecamera, stampante e moltissimo software. Occasione d'oro. Tri - S. Polo 1674 30125 Venezia - Tel. 041/32382

Per Sharp MZ-731 i seguenti programmi: Matematica 1, 2, 3 - Spese Casa - Master Mind - Tris - Tombola - Totocalcio - Atterraggio - Attacco Aereo - Il Guardiano del Te. Scrivere per informazioni. Nicola Bellettieri - Via A. Righi, 34 - 50047 Prato (FI) - Tel. 0574/583039

Vendo Atari 2600, 2 joystick, 7 cartucce: Space War, Slot Racers (Labirinto), Outlaw (Cow Boy), Pole Position, Enduro. Tutto in buone condizioni a L. 500.000. Telefonare ore 14-15/20-21). Santo Torrini - Vicolo Castro, 15 - 95100 Catania - Tel. 095/347703

## Guida per l'input dei programmi versioni VIC 20 e C 64

Notate che i listati contengono "parole" racchiuse tra parentesi graffe { }. Tali parole rappresentano caratteri di controllo come mostrato nel sottostante riquadro. Se sono precedute da un numero, questo indica il numero di volte che quel tasto deve essere premuto. Se il simbolo è sottolineato deve essere premuto contemporaneamente a SHIFT mentre se è racchiuso da [ < > ] deve essere premuto contemporaneamente al tasto COMODORE. Inoltre, se tra parentesi si trova un carattere alfabetico "solitario", questo dovrà essere premuto contemporaneamente al tasto CONTROL.

Con questo sistema di codifica, sarà molto più agevole copiare i listati senza faticose e dubbie interpretazioni di caratteri grafici e di controllo di cursore o dei colori.

Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:
{CLR}	SHIFT CLR/HOME		{CYN}	CTRL 4		[<7>]	7	
{HOME}	CLR/HOME		{PUR}	CTRL 5		[<8>]	8	
{SU}	SHIFT ↑ CRSR ↓		{GRN}	CTRL 6		{F1}	F1	
{GIU'}	↑ CRSR ↓		{BLU}	CTRL 7		{F2}	F2	
{SIN}	SHIFT ← CRSR →		{YEL}	CTRL 8		{F3}	F3	
{DES}	← CRSR →		[<1>]	7		{F4}	F4	
{RVS}	CTRL 9		[<2>]	8		{F5}	F5	
{OFF}	CTRL 0		[<3>]	9		{F6}	F6	
{BLK}	CTRL 1		[<4>]	0		{F7}	F7	
{WHT}	CTRL 2		[<5>]	1		{F8}	F8	
{RED}	CTRL 3		[<6>]	2				

# PERSONAL PICCOLI SOFTWARE ANNUNCI

Sei un lettore di Personal Software e vuoi entrare in contatto con tutti gli altri lettori per comprare, cambiare o vendere il tuo sistema o qualche particolare accessorio? Spedisci questo tagliando a:  
Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano

COMPRO  CAMBIO  VENDO

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

**Attenzione: Piccoli Annunci è aperta solamente alla vendita, l'acquisto o lo scambio di materiale fra privati, in un unico esemplare. Annunci di diversa natura verranno cestinati.**

Annuncio gratuito  Annuncio evidenziato (allegare L. 5.000). Non si accettano fotocopie o fogli volanti: allegati al coupon.

Nome ..... Cognome .....

Via ..... N. .... C.A.P. ....

Città ..... Prov. .... Tel. ....

# PERSONAL SOFTWARE LIBRI

Gruppo Editoriale Jackson

Vuoi ordinare dei libri? Spedisci questo tagliando a:  
Gruppo Editoriale Jackson  
Via Rosellini, 12 - 20124 Milano

Nome Cognome

\_\_\_\_\_

Indirizzo

\_\_\_\_\_ N. \_\_\_\_\_

Cap.

\_\_\_\_\_ Città \_\_\_\_\_ Provincia \_\_\_\_\_

Partita I.V.A. (indispensabile per le aziende)

\_\_\_\_\_

Si richiede l'emissione della fattura

Inviatemi i seguenti libri:

Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità	Codice Libro	Quantità

Pagherò al postino il prezzo indicato + L. 3.000 per contributo fisso spese di spedizione.

Allego assegno n° ..... di L. ....

Non abbonato  Abbonato sconto  Elettronica Oggi  Elettronica Oggi  Elektor  Informatica Oggi  Computerworld  Bit  Personal Software  Strumenti Musicali  Video Giochi  Home computer  Soft Dealer  Telecomunicazioni Oggi  PC Magazine  Personal O  Compu scuola  Super VIC  Super Commodore

Data .....

Firma .....

UNA PUBBLICAZIONE DEL GRUPPO EDITORIALE JACKSON

# PERSONAL SOFTWARE

ANNO 4 - N. 29 GIUGNO 1985

**DIRETTORE RESPONSABILE:** Giampietro Zanga

**CAPO REDATTORE:** Lorenzo Barrile

**DIRETTORE TECNICO:** Riccardo Paolillo

**COORDINAMENTO TECNICO:** Luca Zaninello

**GRAFICA E IMPAGINAZIONE:** Ugo Luzzati

**SEGRETARIA DI REDAZIONE:** Silvia Borgonovo

**REDAZIONE USA:** Gabriella Martino, Sergio Mello-Grand  
G.E.J. Publishing Group Inc.

**UFFICIO ABBONAMENTI:** Tel. 6880951/2/3/4/5

**HANNO COLLABORATO A QUESTO NUMERO:**

P. Agostini, A. Motta, D. Cerù, P. Ferrami, M. Modelli, V. Delle Cave, G. Puccio, I. Parbuono, M. Giacobazzi, S. Borsani, M. Lenzi

## CONSOciate ESTERE:

**USA:** G.E.J. PUBLISHING Group, Inc. a Subsidiary of -  
Gruppo Editoriale Jackson 1307 S. Mary Avenue -  
94087 Sunnyvale (CA) Tel. (408) 7730103 - Telex: 49959972

**GRAN BRETAGNA:** G.E.J. Publishing L.t.d. 187 Oxford Street - London W1R1AJ  
Tel. (01) 4392931 - Telex: (051) 21248

**PUBBLICITÀ:** Concessionario per l'Italia e l'Estero  
J. Advertising S.r.l. - Via Restelli, 5 - 20124 Milano  
Tel. (02) 6882895-6882458-6880606 Telex: 316213 REINA I

**PUBBLICITÀ PER USA E CANADA:**  
International Media Marketing 16704 Marquardt Avenue,  
P.O. Box 1217 Cerritos (CA) 90701 - (213) 926-9552

**FOTOCOPOSIZIONE:** Lineacomp S.r.l. Via Rosellini, 12 - 20124 Milano

**STAMPA:** Litografica S.r.l. (Busto Arsizio - MI)

**CONCESSIONARIO ESCLUSIVO PER LA DIFFUSIONE IN ITALIA E ALL'ESTERO:** SODIP - Via Zuretti, 25 - 20125 Milano

**AUTORIZZAZIONE ALLA PUBBLICAZIONE:**  
Tribunale di Milano n. 69 del 20/2/1982

Spedizione in abbonamento postale Gruppo III/70  
Prezzo della rivista L. 4.000. Numero arretrato L. 8.000  
Abbonamento annuo (11 numeri) L. 34.000; per l'Estero L. 51.000  
I versamenti vanno indirizzati a: Gruppo Editoriale Jackson - Via Rosellini, 12  
20124 Milano - mediante emissione di assegno bancario, cartolina vaglia  
o utilizzando il C/C postale numero 11666203.  
Per i cambi di indirizzo, indicare, oltre naturalmente al nuovo, anche l'indirizzo  
precedente, ed allegare alla comunicazione l'importo di L. 500, anche in francobolli.

© TUTTI I DIRITTI DI RIPRODUZIONE O TRADUZIONE  
DEGLI ARTICOLI PUBBLICATI SONO RISERVATI

Il Gruppo Editoriale Jackson è iscritto nel Registro nazionale della Stampa  
al n. 117 - Vol. 2 - Foglio 129 in data 17-8-1982

## GRUPPO EDITORIALE JACKSON S.r.l.

**DIREZIONE, REDAZIONE, AMMINISTRAZIONE:**  
Via Rosellini, 12 - 20124 Milano - Telefoni: 68.80.951/2/3/4/5

**SEDE LEGALE:** Via G. Pozzone, 5 - 20121 Milano

**DIREZIONE EDITORIALE:** Daniele Comboni

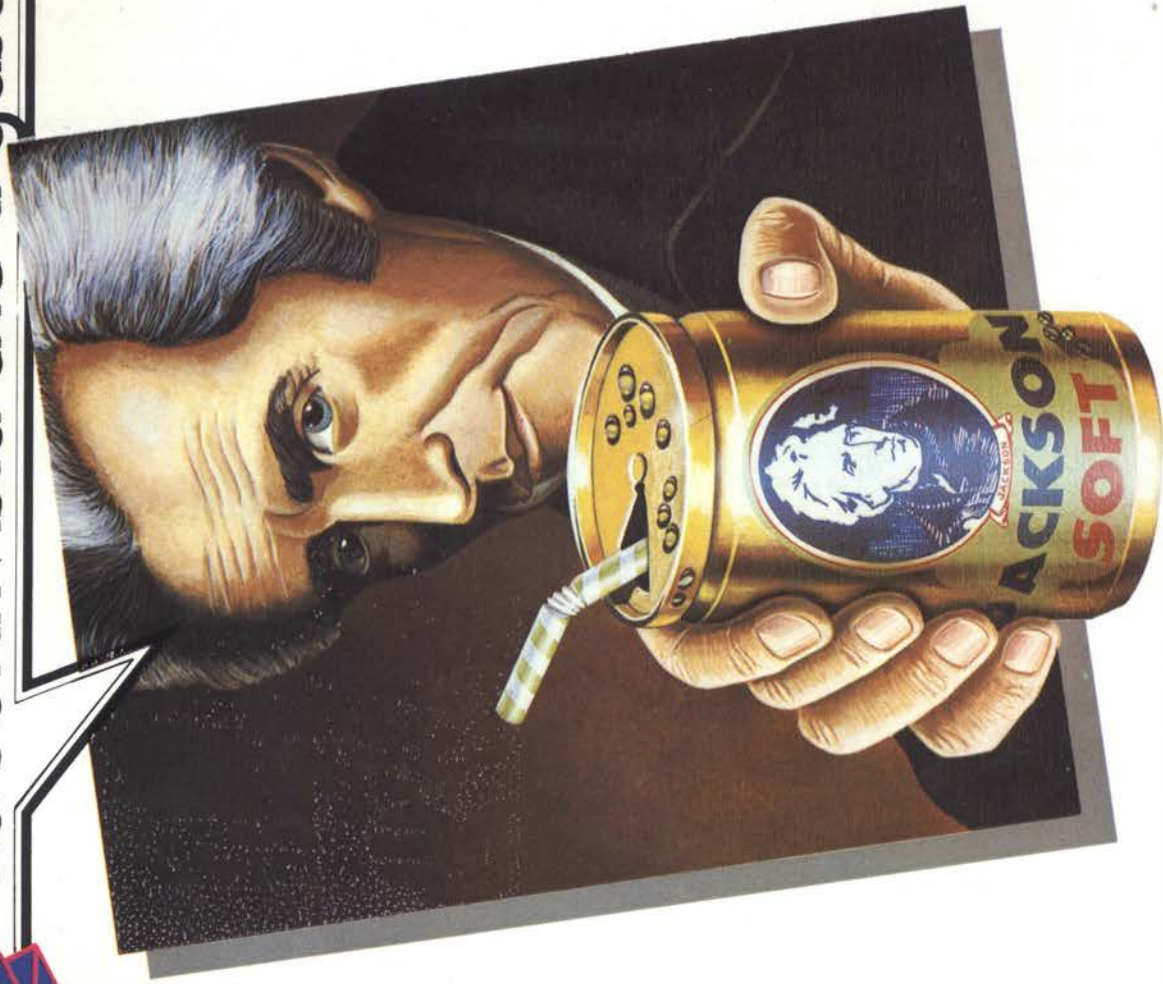
**DIREZIONE AMMINISTRATIVA:** Giuliano Di Chiano

**DIREZIONE DIVISIONE PERIODICI:** Dario Tiengo

**DIREZIONE DIVISIONE LIBRI E GRANDI OPERE:** Roberto Pancaldi

**FINALMENTE!**

**La Softrivista che ti gasa!**



**QUALCOSA DI SUPER, DI INEDITO,  
DI IRRESISTIBILE**

**IL VERO GIOCO**

**COMINCIA ADESSO**

**IN EDICOLA  
JACKSON SOFT  
SERIE ORO**

I giochi esclusivi per Commodore 64 e Spectrum 48 K importati dall'Inghilterra, mai presentati in Italia. Una sfida Jackson al già visto, al già fatto, al... già registrato.



La prima puntata del fantastico, inedito **PYJAMARAMA**



Corri in edicola, il vero gioco comincia solo adesso e se sei davvero bravo partecipa alla "sfida al campione", utilizzando il tagliando che troverai sull'ultima pagina di copertina di ogni numero.

GRUPPO EDITORIALE JACKSON

