

Uno schermo di trecento righe

Utilizzando un semplice programma in Linguaggio Macchina è possibile fare in modo che lo schermo video standard dell'Apple si trasformi in una finestra di 24 righe prelevate da una pagina di oltre trecento. Per contenere trecento righe da 40 caratteri ciascuna occorrono 12000 locazioni di memoria: proprio quante ne contiene la Language Card.

Infatti a parte quattro Kappa gestiti a banchi l'espansione language card aggiunge 12 Kappa alla memoria dell'Apple. Questi dodici Kappa non sono però di solito usati perché nascosti sotto la ROM che, prima di poter utilizzare la RAM, deve essere disabilitata. Dal momento che la ROM contiene sia il Basic che il sistema operativo,

ogni programma che utilizza l'espansione di memoria deve essere «auto-sufficiente». O quanto meno lo devono essere le subroutine di scrittura e lettura.

Come si possono utilizzare i dodici Kappa di RAM in più come memoria di schermo?

In pratica non potendo cambiare i puntatori di inizio e fine RAM-VIDEO, come avviene in altre macchine, dobbiamo ricorrere ad un piccolo trucco: si intercetta la routine di SCROLLING, che è quella che ci fa perdere la prima riga in alto, e prima che questa venga eseguita si salva la prima riga nella Language Card.

Purtroppo però la routine di scrolling nell'Apple è interna a quella di gestione della stampa (COUT: \$FDFO) e per intercettarla è necessario riscrivere una parte di questa, al-

meno fino alla chiamata della routine di SCROLL.

Fatto questo si deve aggiungere una routine di salvataggio da video a Language Card e una di richiamo da Language Card a video e una di scroll verso il basso. Per poter rivedere poi quanto salvato sulla Language Card in modo semplice, si sono attivati tre nuovi tasti: CTRL W, CTRL Z e CTRL Q. I primi due permettono di srotolare lo schermo in alto e in basso facendolo scorrere dietro la finestra video, il terzo esegue un HOME e riporta all'inizio i puntatori della Language Card.

Per i possessori di un Apple IIc i tre tasti sono stati trasformati in MELA-VUOTA freccia su, MELA-VUOTA freccia giù e MELA-VUOTA freccia a sinistra, inoltre si sono utilizzate alcune istruzioni nuove del 65C02 e una routine speciale del Monitor IIc.

L'uso è semplicissimo: caricate in memoria con il solito CALL -151 il dump di figura 4 per l'Apple II+ o il IIc oppure quello di figura 3 per l'Apple IIc o il IIe con le nuove ROM, poi salvate il tutto con BSAVE ROLL,AS9480,LS17F. Quando avete necessità di eseguire un listato molto lungo, cercate un file in un CATALOG o aspettate delle stampe da un

Questo programma è disponibile su disco presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 151.

```

1 : ROLL up e down
2 :
3 : *** Versione IIc ***
4 :
5 :
6 ROM EQU $C0B2
7 LCON EQU $C0B3
8 :
9 WNDL EPZ $20
10 WNDW EPZ $21
11 WNDT EPZ $22
12 WNDB EPZ $23
13 CH EPZ $24
14 CV EPZ $25
15 BREL EPZ $2B
16 BASH EPZ $29
17 BAS2L EPZ $2A
18 BAS2H EPZ $2B
19 INVF EPZ $22
20 YSAV EPZ $34
21 CSW EPZ $36
22 CSWH EPZ $37
23 HIL EPZ $73
24 HIH EPZ $74
25 AUXL EPZ $CE
26 AUXH EPZ $CF
27 :
28 KBRD EQU $C000
29 KSTR EQU $C010
30 MELAV EQU $C061
31 MELAP EQU $C062
32 :
33 DOSTN EQU $3EA
34 :
35 SCROLL EQU $FC70
36 BELL EQU $FB09
37 SCR1 EQU $FC95
38 VTABZ EQU $FC24
39 BS EQU $FC10
40 CLRRLN EQU $FCA0
41 HOME EQU $FC5B
42 :
43 DRG $9480
44 OBJ $B00
45 :
46 : Intercetta la COUT
47 :
48 START LDA #$94
49 STA CSWH
50 STA HIH
51 STA CSW
52 :
53 LDA #$1F
54 STA HIL
55 JSR CLRRAUX
56 JMP DOSTN
57 :
58 NOP
59 :
60 : Vecchio inizio dell COUT
61 : copiato dalla ROM
62 :
63 COUT1 CMP #$A0
64 BCC COUT2
65 AND INVF
66 COUT2 STY YSAV+1
67 PHA
68 JSR VIDW
69 PLA
70 LDY YSAV+1
71 RTS
72 :
73 VIDW CMP #$BD
74 BNE NWAIT
75 LDY KBRD
76 BPL NWAIT
77 CPY #$93
78 BNE NWAIT
79 BIT KSTR
80 KWAIT LDY KBRD
81 BPL KWAIT
82 CPY #$93
83 BEQ NWAIT
84 BIT KSTR
85 NWAIT BRA VOUT :BRA #$B0
86 :
87 STOR LDY CH
88 STA (BASL),Y
89 INC CH
90 ADV LDA CH
91 CMP WNDW
92 BCS CR
93 RTS
94 VOUT CMP #$A0
95 BCS STOR
96 TAY
97 BPL STOR
98 CMP #$BD
99 BEQ CR
100 CMP #$BA
101 BEQ LF1
102 :
103 BNE UP1
104 BS1 BIT MELAV
105 BMI CLRHOM
106 JMP BS
107 :
108 UP1 BIT MELAV
109 BMI ROLLUP
110 JMP BELL
111 LF1 BIT MELAV
112 BPL LF
113 BMI ROLLDWN
114 :
115 CR HEX 6424 :STZ CH
116 LF INC CV
117 LDA CV
118 CMP WNDB
119 BCS SCRL
120 JMP VTABZ
121 SCRL DEC CV
122 :
123 : nuova scrolling
124 :
125 NSCROL JSR SCROLL1
126 BCC RTS2
127 AUXSCR HEX 642A :STZ BAS2L
128 LDA #$D0
129 STA BAS2H
130 LDA LCON
131 LDA LCON
132 LDY #$2B
133 LOOP1 LDA (BAS2L),Y
134 HEX 922A :STA (BAS2L)
135 INC BAS2L
136 BNE LOOP1
137 INC BAS2H
138 BNE LOOP1
139 LDA ROM
140 RTS2 RTS
141 :
142 :vecchio scrolli modificato
143 :
144 SCROLL1 JSR SAVET
145 JSR SCROLL
146 INCAUX CLC
147 LDA AUXL
148 ADC #$2B
149 STA AUXL

```

(continua a pag. 130)

(segue da pag. 129)

```

150      BCC OUT
151      INC AUXH
152 OUT   CMP #8D1
153      LDA AUXH
154      SBC #FF
155      BCC RTS1          ;C=1 >=FFD1
156      LDA #8D0
157      STA AUXL
158 RTS1   RTS
159 ;
160 ;Clear aux + Home
161 ;
162 CLRHM  JSR HOME
163 CLRAX  LDA #8D0
164      STA AUXH
165      HEX 64CE
166      RTS          ;STZ AUXL
167 ;
168 ;roll UP
169 ;
170 ROLUP  JSR SCROLL1
171      BCC LOADL
172      JMP SCRL3
173 ;
174 ;roll DOWN
175 ;
176 ROLDWN LDA AUXL
177      JSR OUTL
178      BCC RTS2
179      JSR SAVEL
180      JSR SCRLDW
181      JSR DECAUX
182      BCS LOADT

183 CLRTOP LDA WNDT
184      JSR VTABZ
185      JMP CLRLIN
186 ;
187 ; decrementa puntatore RAM
188 ;
189 DECAUX SEC
190      LDA AUXL
191      SBC #28
192      STA AUXL
193      BCS OUTL
194      DEC AUXH
195 OUTL   CMP #0
196      LDA AUXH
197      SBC #8D0
198      BCC CLRAUX
199 RTS6   RTS
200 ;
201 ; salva l'ultima riga
202 ;
203 SAVEL  LDA WNDB
204      HEX 3A          ;DEC acc.
205      JSR VTABZ
206      BRA MOVETO    ;BRA->#80
207 ;
208 ; salva la prima riga
209 ;
210 SAVET  LDA WNDB
211      JSR VTABZ
212 ;
213 ; muove da video a LC
214 ;
215 MOVETO LDA LCON

216      LDA LCON
217      LDY WNDW
218      DEY
219 MOVI   LDA (BASL),Y
220      STA (AUXL),Y
221      DEY
222      BPL MOVI
223      LDA ROM
224      RTS
225 ;
226 ; recupera la prima riga
227 ;
228 LOADT  LDA AUXL
229      BNE DK1
230      LDA AUXH
231      CMP #8D0
232      BEQ RTS4
233 DK1    LDM WNDT
234      JSR VTABZ
235      BRA MOVEDA    ;BRA->#80
236 ;
237 ;recupera ultima riga
238 ;
239 LOADL  LDA WNDB
240      HEX 3A          ;DEC acc.
241      JSR VTABZ
242 ;
243 ; muove da LC a video
244 ;
245 MOVEDA LDA LCON
246      LDA LCON
247      LDY WNDW
248      DEY

```

Figura 1 - Sorgente assembler della routine di roll up e down per Apple IIc o IIe con le nuove ROM. Il programma non va bene per i modelli senza 65C02 perché usa alcuni dei nuovi codici e una nuova routine del monitor.

```

1 ; ROLL up e down
2 ;
3 ; *** Versione II+, IIe **
4 ;
5 ;
6 ROM    EQU %C0B2
7 LCON   EQU %C0B3
8 ;
9 WNDL   EP2 %20
10 WNDW  EP2 %21
11 WNDR  EP2 %22
12 WNDB  EP2 %23
13 CH    EP2 %24
14 CV    EP2 %25
15 BASL  EP2 %28
16 BASH  EP2 %29
17 BAS2L EP2 %2A
18 BAS2H EP2 %2B
19 INVFL EP2 %32
20 YSAV  EP2 %34
21 CSW   EP2 %36
22 CSWH  EP2 %37
23 HIL   EP2 %73
24 HIH   EP2 %74
25 AUXL  EP2 %CE
26 AUXH  EP2 %CF
27 ;
28 KBRD  EQU %C000
29 KSTR   EQU %C010
30 ;
31 ;
32 ;
33 DOSIN EQU %3EA
34 ;
35 SCROLL EQU %FC70
36 BELL   EQU %FB09
37 SCRL3  EQU %FC95
38 VTABZ  EQU %FC24
39 BS     EQU %FC10
40 CLREOL EQU %FC9C
41 HOME   EQU %FC5B
42 ;
43      ORG %9480
44      OBJ %800
45 ;
46 ;
47 ; Intercetta la COUT
48 ;
49 START  LDA #594
50      STA CSWH
51      STA HIH
52      STA CSW
53      LDA #57F
54      STA HIL
55      JSR CLRAUX
56      JMP DOSIN
57 ;
58      NOP
59      NOP
60 ;
61 ;Vecchio inizio della COUT
62 ;copiato dalla ROM.
63 ;
64 COUT1  CMP #8A0
65      BCC COUTZ
66      AND INVFL
67 COUTZ  STY YSAV+1
68      PHA
69      JSR VIDW
70      PLA
71      LDY YSAV+1
72      RTS
73 ;

74 VIDW  CMP #8D0
75      BNE NWAIT
76      LDY KBRD
77      BPL NWAIT
78      CPY #93
79      BNE NWAIT
80      BIT KSTR
81 KWAIT  LDY KBRD
82      BPL KWAIT
83      CPY #8B3
84      BEQ NWAIT
85      BIT KSTR
86 NWAIT  JMP VOUT
87 ;
88 STOR  LDY CH
89      STA (BASL),Y
90      INC CH
91 ADV   LDA CH
92      CMP WNDW
93      BCS CR
94 RTS3  RTS
95 VOUT  CMP #8A0
96      BCS STOR
97      TAY
98      BPL STOR
99      CMP #8B0
100     BEQ CR
101     CMP #8BA
102     BEQ LF
103     CMP #8B8
104     BNE NEWCOM
105     BACK JMP BS
106 ;
107 ; nuovi comandi
108 ;
109 NEWCOM CMP #597          ;'W
110     BEQ ROLUP
111     CMP #59A          ;'Z
112     BEQ ROLDWN
113     CMP #591          ;'D
114     BEQ CLRHM
115     JMP BELL
116 ;
117 CR    LDA #500
118     STA CH
119 LF    INC CV
120     LDA CV
121     CMP WNDB
122     BCS SCRL
123     JMP VTABZ
124 SCRL  DEC CV
125 ;
126 ; nuovo scrolling
127 ;
128 NSCROL JSR SCROLL1
129     BCC RTS2
130 AUXSCR LDA #0
131     STA BAS2L
132     LDA #8D0
133     STA BAS2H
134     STA LCON
135     LDA LCON
136 LOOP1 LDY #28
137     LDA (BAS2L),Y
138     LDY #0
139     STA (BAS2L),Y
140     INC BAS2L
141     BNE LOOP1
142     INC BAS2H
143     BNE LOOP1
144     LDA ROM

145 RTS2   RTS
146 ;
147 ;vecchio scroll modificato
148 ;
149 SCROLL1 JSR SAVET
150     JSR SCROLL
151     INCAUX CLC
152     LDA AUXL
153     ADC #28
154     STA AUXL
155     BCC OUT
156     INC AUXH
157 OUT    CMP #8D1
158     LDA AUXH
159     SBC #FF          ;C=1 >=FFD1
160     BCC RTS1
161     LDA #8D0
162     STA AUXL
163 RTS1   RTS
164 ;
165 ; Clear AUX + HOME
166 ;
167 CLRHM  JSR HOME
168 CLRAX  LDA #8D0
169     STA AUXH
170     LDA #0
171     STA AUXL
172     RTS
173 ;
174 ; roll UP
175 ;
176 ROLUP  JSR SCROLL1
177     BCC LOADL
178     JMP SCRL3
179 ;
180 ; roll DOWN
181 ;
182 ROLDWN LDA AUXL
183     JSR OUTL
184     BCC RTS2
185     JSR SAVEL
186     JSR SCRLDW
187     JSR DECAUX
188     BCS LOADT
189 CLRTOP LDA WNDR
190     JSR VTABZ
191     JMP CLREOL
192 ;
193 ; decrementa puntatore RAM
194 ;
195 DECAUX SEC
196     LDA AUXL
197     SBC #28
198     STA AUXL
199     BCS OUTL
200     DEC AUXH
201 OUTL   CMP #0
202     LDA AUXH
203     SBC #8D0
204     BCC CLRAUX
205 RTS6   RTS
206 ;
207 ; salva l'ultima riga
208 ;
209 SAVEL  LDY WNDB
210     DEY
211     TAY
212     JSR VTABZ
213     JMP MOVETO
214 ;
215 ; salva la prima riga
216 ;

```

Figura 2 - Sorgente assembler della routine di roll up e down per Apple II+ e vecchi IIe.

APPLE

```

249 MOV2 LDA (AUXL),Y
250 STA (BASL),Y
251 DEY
252 BPL MOV2
253 LDA ROM
254 RTS4 RTS
255 :
256 : scrolling verso il basso
257 :
258 SCRLDW LDX WNDB
259 DEX
260 TXA
261 JSR VTABZ
262 NEXTL LDA BASL
263 STA BAS2L
264 LDA BASH
265 STA BAS2H
266 DEX
267 TXA
268 JSR VTABZ
269 LDY WNDW
270 DEY
271 NEXTC LDA (BASL),Y
272 STA (BAS2L),Y
273 DEY
274 BPL NEXTC
275 CPX WNDT
276 BNE NEXTL
277 RTS
278 :
279 :
280 :
281 END

```

```

217 SAVET LDA WNDT
218 JSR VTABZ
219 :
220 : muove da video a LC
221 :
222 MOVETO LDA LCON
223 LDA LCON
224 LDY WNDW
225 DEY
226 MOV1 LDA (BASL),Y
227 STA (AUXL),Y
228 DEY
229 BPL MOV1
230 LDA ROM
231 RTS
232 :
233 : recupera la prima riga
234 :
235 LOADT LDA AUXL
236 BNE OK1
237 LDA AUXH
238 CMP #80
239 BEQ RTS4
240 OK1 LDA WNDT
241 JSR VTABZ
242 JMP MOVEVA
243 :
244 : recupera ultima riga
245 :
246 LOADL LDY WNDB
247 DEY
248 TYA
249 JSR VTABZ
250 :
251 : muove da LC a video
252 :
253 MOVEVA LDA LCON
254 LDA LCON
255 LDY WNDW
256 DEY
257 MOV2 LDA (AUXL),Y
258 STA (BASL),Y
259 DEY
260 BPL MOV2
261 LDA ROM
262 RTS4 RTS
263 :
264 : scrolling verso il basso
265 :
266 SCRLDW LDX WNDB
267 DEX
268 TXA
269 JSR VTABZ
270 NEXTL LDA BASL
271 STA BAS2L
272 LDA BASH
273 STA BAS2H
274 DEX
275 TXA
276 JSR VTABZ
277 LDY WNDW
278 DEY
279 NEXTC LDA (BASL),Y
280 STA (BAS2L),Y
281 DEY
282 BPL NEXTC
283 CPX WNDT
284 BNE NEXTL
285 RTS
286 :
287 :
288 :
289 END

```

```

9480- A9 94 85 37 85 74 85 36
9488- A9 7F 85 73 20 4C 95 4C
9490- EA 03 EA EA C9 A0 90 02
9498- 25 32 84 35 48 20 A4 94
94A0- 68 A4 35 60 C9 8D D0 18
94A8- AC 00 C0 10 13 C0 93 D0
94B0- 0F 2C 10 C0 AC 00 C0 10
94B8- FB C0 83 F0 03 2C 10 C0
94C0- 80 0D A4 24 91 28 E6 24
94C8- A5 24 C5 21 80 28 60 C9
94D0- A0 B0 EF AB 10 EC C9 8D
94D8- F0 1F C9 8A F0 14 C9 88
94E0- D0 08 2C 61 C0 30 62 4C
94E8- 10 FC 2C 61 C0 30 64 4C
94F0- D9 FB 2C 61 C0 10 04 30
94F8- 62 64 24 E6 25 A5 25 C5
9500- 23 B0 03 4C 24 FC C6 25
9508- 20 28 95 90 1D 64 2A A9
9510- D0 85 2B AD 83 C0 AD 83
9518- C0 A0 28 B1 2A 92 2A E6
9520- 2A D0 F8 E6 2B D0 F4 AD
9528- 82 C0 60 20 91 95 20 70
9530- FC 18 A5 CE 69 28 85 CE
9538- 90 02 E6 CF C9 D1 A5 CF
9540- E9 FF 90 04 A9 D0 85 CE
9548- 60 20 58 FC A9 D0 85 CF
9550- 64 CE 60 20 2B 95 90 63
9558- 4C 95 FC A5 CE 20 80 95
9560- 90 C8 20 89 95 20 D5 95
9568- 20 75 95 B0 3D A5 22 20
9570- 24 FC 4C A0 FC 38 A5 CE
9578- E9 28 85 CE B0 02 C6 CF
9580- C9 00 A5 CF E9 D0 90 C4
9588- 60 A5 23 3A 20 24 FC 80
9590- 05 A5 22 20 24 FC AD 83
9598- C0 AD 83 C0 A4 21 88 B1
95A0- 28 91 CE 88 10 F9 AD 82
95A8- C0 60 A5 CE D0 06 A5 CF
95B0- C9 D0 F0 20 A5 22 20 24
95B8- FC 80 06 A5 23 3A 20 24
95C0- FC AD 83 C0 AD 83 C0 A4
95C8- 21 88 B1 CE 91 28 88 10
95D0- F9 AD 82 C0 60 A6 23 CA
95D8- BA 20 24 FC A5 28 85 2A
95E0- A5 29 85 2B CA 8A 20 24
95E8- FC A4 21 88 B1 28 91 2A
95F0- 88 10 F9 E4 22 D0 E5 60

```

Figura 3 - Dump di memoria dalla versione IIc del programma ROLL.

```

9480- A9 94 85 37 85 74 85 36
9488- A9 7F 85 73 20 70 95 4C
9490- EA 03 EA EA C9 A0 90 02
9498- 25 32 84 35 48 20 A4 94
94A0- 68 A4 35 60 C9 8D D0 18
94A8- AC 00 C0 10 13 C0 93 D0
94B0- 0F 2C 10 C0 AC 00 C0 10
94B8- FB C0 83 F0 03 2C 10 C0
94C0- 4C D0 94 A4 24 91 28 E6
94C8- 24 A5 24 C5 21 B0 26 60
94D0- C9 A0 B0 EF AB 10 EC C9
94D8- 80 F0 1A C9 8A F0 1A C9
94E0- 88 D0 03 C0 10 FC C9 97
94E8- F0 61 C9 9A F0 65 C9 91
94F0- F0 7B 4C D9 FB A9 00 85
94F8- 24 E6 25 A5 25 C5 23 80
9500- 03 4C 24 FC C6 25 20 2D
9508- 95 90 21 A9 00 85 2A A9
9510- D0 85 2B AD 83 C0 AD 83
9518- C0 A0 28 B1 2A A0 00 91
9520- 2A E6 2A D0 F4 E6 2B D0
9528- F0 AD 82 C0 60 20 97 95
9530- 20 70 FC 18 A5 CE 69 28
9538- 85 CE 90 02 E6 CF C9 D1
9540- A5 CF E9 FF 90 04 A9 D0
9548- 85 CE 60 20 2D 95 90 72
9550- 4C 95 FC A5 CE 20 84 95
9558- 90 D2 20 8D 95 20 D5 95
9560- 20 79 95 B0 4B A5 22 20
9568- 24 FC 4C 9C FC 20 58 FC
9570- A9 D0 85 CF A9 00 85 CE
9578- 60 38 A5 CE E9 28 85 CE
9580- 80 02 C6 CF C9 00 A5 CF
9588- E9 D0 90 E4 60 A4 23 88
9590- 98 20 24 FC 4C 9C 95 A5
9598- 22 20 24 FC AD 83 C0 AD
95A0- 83 C0 A4 21 88 B1 28 91
95A8- CE 88 10 F9 AD 82 C0 60
95B0- A5 CE D0 06 A5 CF C9 D0
95B8- F0 22 A5 22 20 24 FC 4C
95C0- C9 95 A4 23 88 98 20 24
95C8- FC AD 83 C0 AD 83 C0 A4
95D0- 21 88 B1 CE 91 28 88 10
95D8- F9 AD 82 C0 60 A6 23 CA
95E0- BA 20 24 FC A5 28 85 2A
95E8- A5 29 85 2B CA 8A 20 24
95F0- FC A4 21 88 B1 28 91 2A
95F8- 88 10 F9 E4 22 D0 E5 60

```

Figura 4 - Dump di memoria dalla versione II+ del programma ROLL.

programma e avete paura di perderne una parte (magari dopo aver atteso sei ore il risultato), lanciate BRUN ROLL e premete CTRL Q o MELA-VUOTA←. Provate poi a listare un programma molto lungo e godetevi la soddisfazione di scorrere in su e giù il listato con la semplice pressione di due tasti.

Avvertenze

Dopo un reset la routine si sgancia, per riattivarla occorre effettuare una CALL 38016 da Basic o un 9480G dal monitor. La routine di ROLL non funziona sotto proDOS in quanto il proDOS risiede anche lui nella Language Card e se qualcuno ci scrive sopra si arrabbia non poco.

Ci sono poi delle difficoltà nel caso siano installate le espansioni di memoria a 128K (tipo RAMEX, SATURN ecc.) che creano conflitti con la Language Card. Un altro piccolo bug del programma (risolvibile solo andando

a modificare anche la routine di GET-LINE) è il fatto che l'interprete del Basic non è abituato a ignorare i nuovi comandi e premendo il RETURN dopo uno o più di questi si ottiene il messaggio di SYNTAX ERROR; se proprio dà fastidio si può uscire con il CTRL X. Lo stesso avviene dopo 240 comandi ROLL consecutivi, allorché, riempito il buffer di riga, il computer comincia a suonare il campanello e poi esce automaticamente con il CTRL X, ma questo accade molto raramente, e comunque nessuno dei due procura danni a quello che si sta facendo. Nel sorgente Lisa per Apple IIc compaiono alcune istruzioni strane (HEX), servono a creare i rispettivi codici esadecimali delle corrispondenti istruzioni del 65C02 che il LISA non conosce, l'uso dei nuovi codici ha permesso di mantenere invariata la lunghezza del programma nonostante la subroutine di gestione del tasto MELA-VUOTA.

Le routine dell'Applesoft

Questa volta vediamo il gruppo di routine grafiche dell'interprete che ci permetteranno di ottenere dai nostri programmi in linguaggio macchina lo stesso tipo di istruzioni e controlli grafici che sono possibili in ambiente Basic. La velocità di esecuzione di queste routine, che in fondo sono le stesse del Basic è circa raddoppiata, in quanto manca tutta la parte dell'interpretazione del programma.

Le routine grafiche dell'Applesoft non sono certamente le più veloci possibili, infatti in genere si deve scegliere tra una routine veloce, ma lunga in termini di occupazione di memoria e poco versatile, oppure una più lenta, ma completa e molto breve. Dal momento che le routine grafiche devono risiedere in ROM, e che lo spazio a loro disposizione non era poi molto, i progettisti della Microsoft hanno ovviamente optato per delle routine lente, ma che occupassero poco spazio.

Inoltre il modo in cui i punti sullo schermo corrispondono ai bit della memoria presenta delle particolarità tutte Apple che rendono la vita difficile a chi voglia scrivere delle routine efficienti. Una riga della pagina grafica è composta da 280 punti che corrispondono ai sette bit meno significativi di 40 byte di RAM consecutivi. Ma un punto che si trovi nella riga sottostante corrisponde ad un byte che dista 1024 byte da quello soprastante. Dopo otto righe ritroviamo invece la prosecuzione della prima riga con i byte da 41 ad 80 e così via per 64 righe; dopodiché si saltano 64 byte e si ricomincia.

HGR2	F3D8	Seleziona e pulisce la pagina 2 in alta risoluzione
HGR	F3E2	Seleziona e pulisce la pagina 1 in alta risoluzione
CLEAR	F3F2	Pulisce la pagina corrente (nero)
BKGND	F3F6	Colora la pagina corrente con l'ultimo colore utilizzato che si trova in 1C
HPOSN	F411	Posiziona il cursore dell'alta risoluzione senza disegnare nulla; l'Accumulatore contiene la coordinata y e i registri Y e X la parte alta e bassa della coordinata x. Deposita nelle locazioni 26, 27, 30 l'indirizzo di memoria del Bit corrispondente.
HPLLOT	F457	Richiama HPOSN e tenta di plottare un punto alle coordinate del cursore. Se le coordinate sono fuori dello schermo torna con un errore, se il punto è di un colore non compatibile con la posizione sullo schermo non plotta niente.
HPLOTTO	F53A	Disegna una linea dall'ultimo punto plottato alle coordinate: AX=x e Y=y.
FIND	F5CB	Converte la posizione del cursore (locazioni 26, 27 e 30) nelle coordinate x e y; deposita il risultato nelle locazioni E0-E1 ed E2.
DRAW	F601	Disegna una SHAPE nella posizione attuale del cursore, cerca in X e Y l'indirizzo della Shapetable, usa A come ROT = ed E7 come SCALE =.
XDRAW	F65D	Identica alla DRAW tranne che usa il colore complementare a quello del punto da plottare.
SETCOL	F6EC	Setta il colore al valore del registro X. Se questo contiene un numero minore di 0 o maggiore di 7 viene ignorato.

Tabella 1 - Elenco delle routine grafiche dell'interprete Applesoft e loro punti di entrata. Per utilizzarle occorre caricare nei registri del microprocessore i valori desiderati ed effettuare il JSR. Attenzione alle HPLLOT che in caso di coordinate errate disegnano in posizioni casuali.

1A, 1B	Punta alla SHAPE usata da DRAW
1C	Ultimo colore usato
26, 27	Indirizzo del BYTE che contiene il punto x, y
30	Maschera del punto in detto BYTE
E0, E1	x-coord. (0,279)
E2	y-coord. (0,191)
E4	Colore
E6	Pagina (32 = pag. 1, 64 = pag. 2)
E7	SCALE =
E8, E9	Indirizzo delle SHAPE TABLE
EA	Collision counter (usato dalla DRAW)
F9	ROT =

Tabella 2 - Locazioni in pagina zero utilizzate dalle routine grafiche del Basic e loro significato. Il collision counter (\$EA) contiene il numero di punti di una shape che hanno generato collisione durante una DRAW o il numero di punti della shape meno quelli della collisione in caso di XDRAW.

In questo modo avanzano oltretutto 512 byte inutilizzati per ciascuna pagina grafica.

Per quanto riguarda il colore i punti possono essere di quattro colori (più ovviamente il bianco e il nero), che dipendono dalla posizione del punto sullo schermo e dentro al byte: i punti delle colonne dispari (x = 1,3,5,..., 279) possono essere solo Verdi o Arancio, quelli delle colonne pari solo Viola o Blu. Se ciascun punto debba essere Verde o Viola piuttosto che Arancio o Blu dipende dal valore del bit sette del byte di appartenenza. Quindi i punti di uno stesso byte possono essere solo di due dei quattro colori. Per ottenere un punto Bianco si devono accendere due punti accostati.

A causa del fatto che i punti validi per byte sono dispari (7) uno stesso valore (ad esempio \$55) assume un certo colore nei byte pari e il colore opposto nei byte dispari.

Esempio

```

0300- 20 D8 F3 JSR $F3D8
0303- A2 03 LDX #$03
0305- 20 EC F6 JSR $F6EC
0308- A9 B0 LDA #$B0
030A- 95 FF STA $FF
030C- 85 FD STA $FD
030E- A2 D0 LDX #$D0
0310- 86 FE STX $FE
0312- A0 00 LDY #$00
0314- 20 57 F4 JSR $F457
0317- 38 SEC
0318- A5 FF LDA $FF
031A- E5 FD SBC $FD
031C- 85 FE STA $FE
031E- 20 42 03 JSR $0342
0321- 38 SEC
0322- A5 FE LDA $FE
0324- E5 FD SBC $FD
0326- 85 FE STA $FE
0328- 20 42 03 JSR $0342
032B- 18 CLC
032C- A5 FF LDA $FF
032E- 65 FD ADC $FD
0330- 85 FE STA $FE
0332- 20 42 03 JSR $0342
0335- 18 CLC
0336- A5 FE LDA $FE
0338- 65 FD ADC $FD
033A- 85 FE STA $FE
033C- 20 42 03 JSR $0342
033F- 4C 17 03 JMP $0317
0342- C6 FD DEC $FD
0344- F0 09 BEQ $034F
0346- A5 FE LDA $FE
0348- A4 FF LDY $FF
034A- A2 00 LDX #$00
034C- 4C 3A F5 JMP $F53A
034F- 68 PLA
0350- 68 PLA
0351- 60 RTS

```

```

10 HGR2 : HCOLOR= 3
20 X = 208:Y = 170:A = 176
30 HPLLOT X,Y
40 X = X - A:A = A - 1: HPLLOT TO X,Y
50 Y = Y - A:A = A - 1: HPLLOT TO X,Y
60 X = X + A:A = A + 1: HPLLOT TO X,Y
70 Y = Y + A:A = A + 1: HPLLOT TO X,Y
80 IF A > 0 THEN 40

```

Programma per il disegno di una spirale quadrata in alta risoluzione nelle due versioni in Basic e in linguaggio macchina.

Provate a battere:
HGR
CALL-151
*2000:55
*2003:55

i primi punti verranno Viola, i secondi Verdi; battete ancora:

*2006:D5
*2009:D5

i punti saranno ora Blu e Arancio.

Dovrebbe essere chiaro ormai che per utilizzare la grafica dal linguaggio macchina le routine della ROM sono oltremodo comode perché permettono di ignorare tutti questi problemi.

In tabella 1 trovate l'elenco delle routine grafiche e relativi punti di entrata, mentre in tabella 2 ci sono le locazioni in pagina zero utilizzate da queste routine. Come vedete è anche possibile utilizzare dal linguaggio macchina le SHAPE TABLE e le relative istruzioni di DRAW, XDRAW, SCALE e ROT molto comode per eseguire delle scritte in alta risoluzione e con varie inclinazioni e dimensioni. Il programma di esempio disegna una spirale quadrata ed è l'esatta traduzione in linguaggio macchina del listato Basic che gli sta a fianco. Il programma in Basic occupa 152 byte ed impiega tre secondi e un decimo, mentre il programma in linguaggio macchina occupa 82 byte ed impiega per eseguire il disegno 1 secondo e cinque decimi!



DISEGNA E SCRIVE



GRAF

MEGLIO

DI UNA STAMPANTE,
COSTA POCO, LA
TECNOLOGIA È
GIAPPONESE, MA VIENE
DA SAN MARINO:
PS-80, PLOTTER PER
HOME A 80 COLONNE!



IL PLOTTER DI FAMIGLIA.

- Disegna e scrive (caratteri da mm. 1,2 a mm. 75) a 4 colori su carta, cartoncino fino a 21 cm. di larghezza (formato A4) e carta a rullo.

- Il PS-80 è munito di una porta parallela Centronics e di porta seriale RS232 (optional). Quindi è compatibile con la grande maggioranza degli Home e Personal disponibili sul mercato. Sono inoltre disponibili interfacce per i seguenti modelli:

APPLE II/II+/IIe/IIc
COMMODORE 64/16/PLUS4
SINCLAIR ZX/QL

MINICAD

Un programma di Grafica in Alta Risoluzione capace di generare solidi tridimensionali e di manipolarli con varie funzioni tra i quali: Ingrandimento, Riduzione, Rotazione, Duplicazione, ecc. Tutti i disegni ottenuti con MINICAD possono essere memorizzati su disco e/o stampati con il Plotter Stampante PS-80. Con le dovute interfacce possono essere utilizzati il C 64, C 16, PLUS/4, C 128.

Realizzato dalla
LEONI INFORMATICA MILANO

SOFT DUMP

È il programma che consente di disegnare con il PS-80 le immagini del video.

L'Hard-copy del video viene eseguito nei 4 colori disponibili sul PS-80.

- Programma su disco o nastro
- Istruzioni in italiano
Realizzato da COMPUTRON-SHOP - ROMA.

- Garanzia di un anno
- Assistenza e ricambi garantiti dalla ACS della Repubblica di San Marino.

Agenti ACS

Modena/Reggio E./Parma
Computer House
Via Secchi 28/B
42100 Reggio Emilia
Tel. 0522/35890-35872

Marche
Ing. Boria Luciano
Via Manzoni, 44
60131 Ancona
Tel. 071/890292

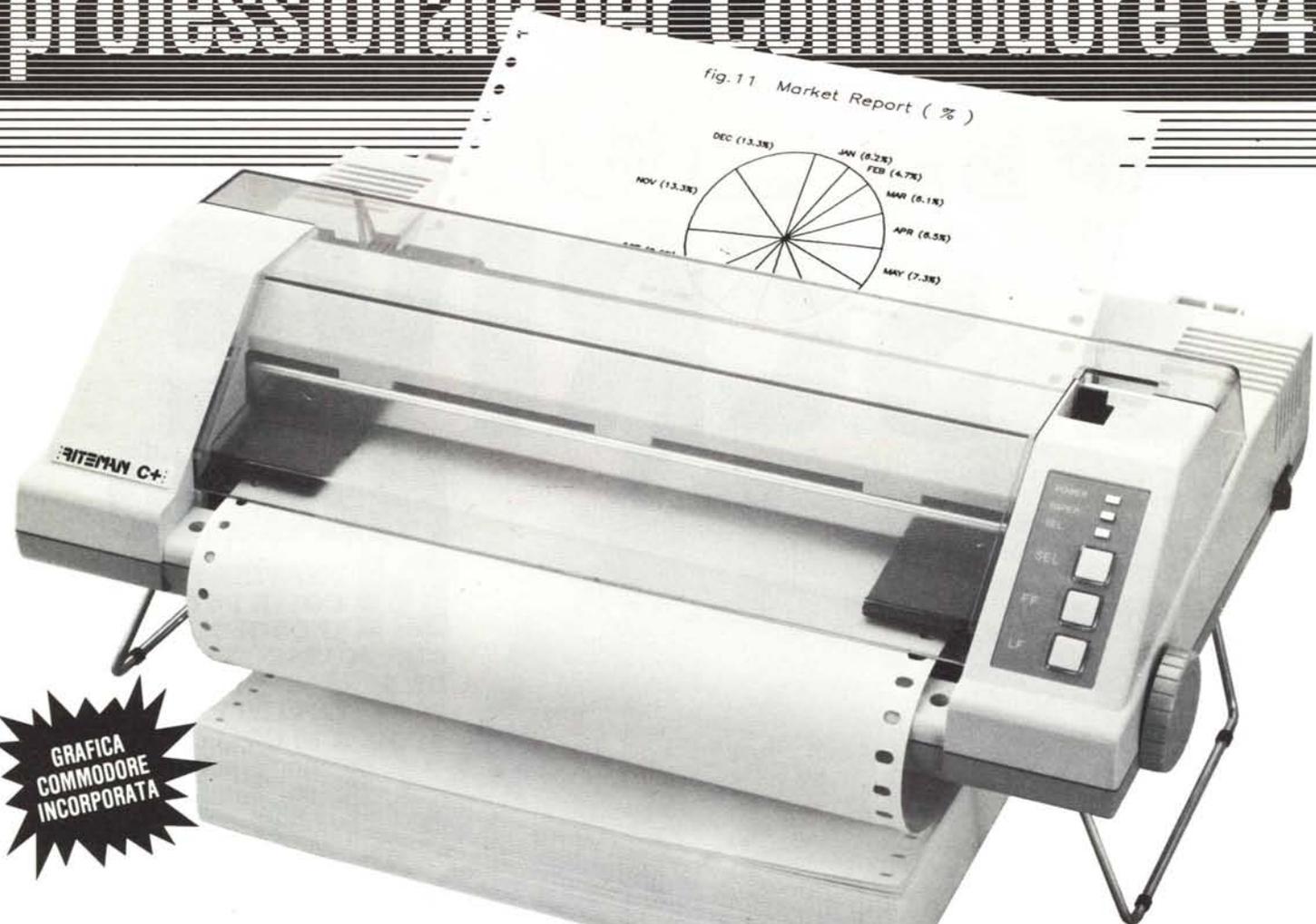
Roma Città
Computron
Largo Forano, 7/8
00199 Roma
Tel. 06/8391556

ACS

ADVANCED COMPUTER SYSTEMS

Via Nonagaldaria, 24
47031 Cailungo Repubblica di San Marino
Tel. 0541/902459

RITEMAN C+ : una stampante professionale per Commodore 64



La straordinaria qualità di stampa e il funzionamento silenzioso vi convinceranno che la **Riteman C+** è veramente speciale.

Dopo aver provato la grande comodità dell'inserimento della carta frontalmente vi renderete conto che questa stampante è proprio quella che fa al caso vostro. La carta viene inserita in continuo mediante il trattore regolabile, senza ulteriori perdite di tempo per il posizionamento e l'allineamento. È possibile utilizzare pure fogli

singoli di carta di qualsiasi spessore, incluso il cartoncino ed anche la stampa di etichette autoadesive senza inconvenienti di sorta! Ciò è reso possibile dalle caratteristiche costruttive del piano di caricamento e dalla linea di alimentazione orizzontale con la testina stampante posta al di sopra della carta.

Notate inoltre i cavalletti di sollevamento incorporati che consentono di inserire il pacco della carta, sotto alla macchina, per una migliore efficienza e per la massima

riduzione dello spazio occupato. Il metodo di stampa, bidirezionale, consente una velocità fino a 105 cps. La qualità delle lettere, unita alla spaziatura proporzionale, consente di produrre documenti quasi «tipografici».

La **Riteman C+**, con le sue prestazioni ed il suo basso costo Vi sorprenderanno veramente!

Se volete ottenere ancor di più dal vostro Commodore 64 è giunta l'ora di dotarlo della stampante Riteman C+.

* Commodore è marchio registrato della Commodore International

METODO DI SCRITTURA: Matrice di punti ● **TESTINA DI SCRITTURA:** 9 punti. Durata della testina 50.000.000 battute di caratteri circa. ● **VELOCITÀ DI STAMPA:** In funzione del tipo di stampa 105 cps, uni o bidirezionali con controllo logico ● **CARICAMENTO CARTA FRONTALE CON TRATTORE** ● **INTERLINEA:** 1/6", 1/8", 7,72", programmabile n/72" e n/216" ● **FORMATO CARTA:** Con trattore da 4" a 9 1/2" - con frizione da 4" a 10" ● **SETS CARATTERI:** 82 segni grafici del Commodore - 96 caratteri ASCII (maiuscole/minuscole) ● **CARATTERI INTERNAZIONALI:** 4 lingue: Italiano, Inglese, Francese, Tedesco, USA. Opzione: Spagnolo, Scandinavo ● **IMMAGINE GRAFICA:** Densità orizzontale di 480 o 960 punti per linea ● **TIPI DI SCRITTURA:** Standard - grassetto - espanso - compresso - com-

presso espanso - enfattizzato - doppio enfattizzato - corsivo - esponenti/deponenti e in negativo ● **FORMATO PAGINA:** Standard a 66 righe con spazio e formato programmabili, con salto sulla perforazione ● **FORMATO BUFFER (una riga standard):** In «standard» 80 caratteri - In «compresso» 132 caratteri - In «espanso» 40 caratteri - In grafica: 480 o 960 bit ● **INTERFACCIA:** Seriale Bus Commodore ● **CONTROLLO SPESSORE CARTA:** 2 fogli (incluso l'originale) - pressione di trascinamento 40 gr. m² ● **CARTUCCIA NASTRO REINCHIOSTRANTE:** Vita media della cartuccia 1 milione di caratteri circa ● **MISURE D'INGOMBRO:** 395x270x105 mm ● **PESO:** Kg. 4,800 ● **ALIMENTAZIONE:** AC 120, 220, 240 volt, 50/60 Hz ● **CONSUMO:** 30 W ● **RUMORE:** < 60 dB

Distributore per l'Italia dei prodotti «Riteman» della C. Itoh. Electronics-Japan

METROMARKET

s.p.a. - import/export - capitale sociale 500.000.000 interamente versato - 34123 TRIESTE - VIA ECONOMO, 5/a - TEL. (040) 730281/2 - TELEX 460252