Routine grafiche estese

di Lorenzo Succi - Lugo (RA)

Le routine che vi invio permettono di ottenere nuove modalità grafiche con l'Apple; questo risultato si ottiene usando sulla Language Card un Applesoft modificato. Per caricare la Language Card ho usato una routine proposta da Valter Di Dio alcuni mesi fa e che mi ha stimolato a "pasticciare" dentro l'interprete. Ecco cosa sono riuscito a fare:

Modo xdraw per hplot e hplot to

Spesso si sente la mancanza, nel tracciamento di linee, di poter avere un comando xplot equivalente al comando xdraw per il tracciamento di profili; in questo modo le linee vengono tracciate con i punti di colore opposto di quello che trovano, e tracciando una seconda volta la stessa linea, questa scompare. Normalmente l'Applesoft quando esegue un comando hplot o hplot to, una volta calcolato l'indirizzo del byte da modificare, va a leggere il byte, gli sovrappone la maschera del colore e del bit e lo riscrive. Per ottenere l'xplot, cioè l'inverso di ciò che viene trovato, bisogna saltare l'uso della maschera del colore ($001C) e la prima lettura del byte; ho scelto di neutralizzare il bit del colore per semplicità e perché perferisco il monitor monocromatico e senza lo shift del mezzo bit.

Copia della pagina 1 sui punti disegnati in pagina 2

Questa modalità è valida per hplot, hplot to, draw e xdraw. In questo caso bisogna estrarre il bit dalla pagina 1, salvarlo, spiazzare di $2000 l'indirizzo a cui si è letto questo bit, scriverlo e riportare il puntatore sulla pagina 1. Poiché bisogna aggiungere istruzioni dove non possono stare, ho messo un JSR al blocco di istruzioni eccedenti, localizzabile in qualsiasi punto sia comodo. Volendo si può usare la zona $F775-$F7D8, quasi in fondo all'Applesoft, in cui si trova il comando shload.

Le routine dell'Applesoft

L'accumulatore floating point ($9D ... $A2)

Prima di andare avanti con le routine del Basic dobbiamo soffermarci un attimo su un elemento fondamentale dell'interprete: il FAC, Floating point ACcumulator. Come visto nelle puntate precedenti in queste sei locazioni vengono messe tutte le informazioni necessarie all'uso delle variabili, siano queste stringhe o numeri interi o numeri reali.

A seconda del contenuto della variabile il FAC o più esattamente le sue locazioni, assumono significati diversi.

Cominciamo dal più semplice.

Numeri Interi

Vanno da -32768 a +32767 e corrispondono ad un numero di 16 bit in complemento a due (il bit più alto vale -32768). In FAC viene posto nelle locazioni $A0 e $A1.

Numeri in virgola mobile

Qui le cose si complicano un pochino, per semplificare i calcoli i computer usano per tutti i numeri la notazione scientifica, ma con esponente 2.

Il numero viene quindi diviso in due parti: la mantissa e l'esponente.

La mantissa comprende gli "uni" significativi del numero, l'esponente indica di quanti posti occorre spostare il punto decimale (o si dovrebbe dire binale?) per avere il valore corretto. Esempio!

5.5 è in binario 101.1, la mantissa è perciò .1011 e l'esponente 3 ovvero 11.

Proprio per come è costruita la mantissa, il primo bit dopo la virgola sarà sempre uno, perché se fosse uno zero si potrebbe far scorrere ancora di un posto la mantissa ed aumentare di uno l'esponente. Nell'Apple allora il primo bit dopo la virgola viene sostituito dal bit di segno; per cui 5.5 diventa .0011 E 11.

Per quanto riguarda la precisione l'Applesoft usa cinque byte per ciascun numero in virgola mobile. Il primo byte è l'esponente messo però in una forma particolare: il valore del byte è uguale a $80 + l'esponente. Quindi $82 rappresenta +2 mentre $7E sarà -2. Il massimo esponente possibile sarà perciò $FF = +127; il minimo dovrebbe essere $00 = -128, ma tutti i numeri che hanno per esponente zero vengono considerati dall'Applesoft nulli, quindi il minor esponente possibile è $01 = -127.

Per la mantissa restano quattro byte meno un bit, il primo, che rappresenta il segno della mantissa (ricordate però che il primo bit dopo la virgola è sottinteso).

Il numero 5.5 sarà rappresentato nella memoria dell'Apple come:

83 30 00 00 00 (HEX)

Quando l'interprete deve utilizzare un numero FP per delle operazioni trasferisce il contenuto della variabile in uno dei suoi due accumulatori FP, appunto FAC1 e FAC2. Negli accumulatori il formato del numero è leggermente diverso: il primo bit della mantissa viene ripristinato e il segno diventa un byte intero che vale $FF se il numero è negativo o $00 se positivo. Quindi

+5.5 → 83 B0 00 00 00 00
−5.5 → 83 B0 00 00 00 FF

Le locazioni dei FAC vanno da $9D a $A2 per il FAC1 e da $A5 a $AA per il FAC2.

Le Stringhe

Nel caso che le operazioni coinvolgano delle stringhe, i byte dell'accumulatore assumono ovviamente significati diversi. Il fatto che il contenuto del FAC sia un numero o una stringa viene indicato dal valore della locazione $11 che vale 0 per i numeri e $FF per le stringhe.

Per le stringhe i parametri sono solo tre: la lunghezza, il byte basso e il byte alto dell'indirizzo di memoria da cui inizia il contenuto della stringa. In FAC1 i tre parametri occupano rispettivamente le locazioni $9D, $9E e $9F.

Conoscendo la funzione dell'accumulatore è possibile utilizzare le routine matematiche dell'interprete anche dai nostri programmi in linguaggio macchina, risparmiando così tempo di sviluppo dei programmi e soprattutto spazio in memoria per il codice.

Altri comandi poco usati, e quindi sacrificabili, sono wait ($E784/28 byte liberi), recall ($F3BC/28 byte liberi), save+load ($D8B0/50 byte liberi, da verificare), store ($F39F/29 byte liberi, già usati per la prossima routine).

Suggerisco di provare a cambiare i byte $600C e $6114 con i codici di and oppure di or. Per portare i bit dalla pagina 2 alla pagina 1 bisogna invertire tra loro le istruzioni ADC #$20 e SBC#$20, come pure le istruzioni CLC e SEC.

L'ultima raffinatezza, a cui sto pensando in un contesto più ampio, sarebbe usare parole chiave diverse per avere contemporaneamente disponibili l'hplot normale e quello modificato senza dover commutare continuamente tra Language Card e ROM come si deve fare adesso.

Nuovo comando:
PIXEL aexpr1,aexpr2;avar

Questa terza routine è un nuovo comando, creato modificando la parola chiave ($D185-$D189), e lo spazio riservato a "store" ($F39F-$F3BB). Avar può essere un numero intero (consigliabile) o un numero reale; al posto del ";" accetta anche il ",".

Questo comando inserisce nella variabile a cui fa riferimento un numero che è funzione dello stato del pixel (bit) le cui coordinate sono "aexpr1", "aexpr2". Se il pixel è spento (=0), "avar"=0; se è acceso (=1), "avar"<>0, con valore che dipende dalla posizione del bit nel byte cui appartiene secondo la formula avar=2^pos×256.

Avvertenze per l'uso: tutte queste routine, come spiegato precedentemente, sono fatte per funzionare su video monocromatico; inoltre compilare programmi Basic che usino queste routine può dare risultati imprevedibili, bisogna provare.

Tracciamento superveloce di linee precalcolate

Questa quarta routine può essere indipendente dalle routine precedenti, non richiede l'uso della Language Card, essendo chiamata con una "call" dopo aver caricato in memoria una tabella contenente le coordinate dei punti estremi di massimo 256 linee, ed è comunque usabile solo in certe condizioni.

Il risultato pratico di questa routine è di "capitalizzare" parte del lavoro fatto dall'interprete e salvarlo in una tabella. In questo modo si risparmia il 75% del tempo. Il vincolo maggiore è dato dal fatto che la tabella è valida solo per quel singolo disegno e in quella specifica parte di schermo in cui è stata disegnata dal programma Basic che l'ha generata. Ogni tabella occupa 6×256 byte, cioè 1536 byte. Per creare la tabella bisogna usare un programma Basic che tracci le linee secondo le proprie esigenze e che dopo ogni punto e ogni linea tracciata vada a leggere in pagina zero il risultato dell'interpretazione e lo salvi in una zona di memoria tampone. Ho scelto un blocco diverso per ogni locazione di pagina zero per poter sfruttare la velocità dell'indirizzamento indicizzato assoluto, che fa risparmiare molte istruzioni. Prima di chiamare la routine bisogna mettere in $FC il numero di linee da tracciare (1..255).

Un ultimo trucchetto grafico, ispiratomi da Apple Mechanic, sostituisce, in modo testo, gli spazi vuoti con un carattere prescelto, stendendo un reticolo sullo schermo non occupato da testo. La locazione da variare è la 64671 ($FC9F) e il valore normale è 160($A0), cioè il blank. Il carattere impostato è alterabile solo con una poke, ricommutando sulle ROM oppure spegnendo.

Questo programma è disponibile su disco presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 184.

```
1  REM  *** SUL //C  TOGLIERE LE RIGHE ***
2  REM  ***  15 - 20 - 35 - 740 - 750  ***
3 :
10 D$ =  CHR$ (13) +  CHR$ (4)
15  POKE 768,0
20  PRINT D$"BRUN SLOT FINDER.OBJ0": IF  PEEK (768) <  > 1 THEN
     HOME : END
25  PRINT D$"BRUN ROM>L.CARD.OBJ0"
30  REM CAMBIA CARATTERE BLANK
35  POKE 64671,174: HOME
36  PRINT
40  HTAB 14: VTAB 7: PRINT "NUOVE ROUTINE"
45  HTAB 14: VTAB 8: PRINT "              "
50  HTAB 14: VTAB 9: PRINT "GRAFICHE  PER"
55  HTAB 14: VTAB 10: PRINT "             "
60  HTAB 14: VTAB 11: PRINT " APPLE "; CHR$ (221); CHR$ (219);"
     BY "
65  HTAB 14: VTAB 12: PRINT "             "
70  HTAB 14: VTAB 13: PRINT "LORENZO SUCCI"
75  HTAB 14: VTAB 14: PRINT "             "
80  HTAB 14: VTAB 15: PRINT "D'OBBLIGO  LA"
85  HTAB 14: VTAB 16: PRINT "             "
90  HTAB 14: VTAB 17: PRINT "LANGUAGE CARD"
95  FOR I = 1 TO 5000: NEXT
140  HGR
150  PRINT D$"BLOAD PIC.UNO,A$2000"
155  VTAB (22): PRINT "ESEMPI DI HPLOT 'XOR', USA IL JOYSTICK"
159  REM MODIFICA HPLOT TO(HLINE)
160  POKE 62861,165: POKE 62862,48: POKE 62863,234: POKE 62864,2
     34: POKE 62865,41: POKE 62866,127
169  REM MODIFICA HPLOT
170  POKE 62554,165: POKE 62555,48: POKE 62556,234: POKE 62557,2
     34: POKE 62558,41: POKE 62559,127
250  VTAB (23): PRINT "PREMI UN TASTO PER CONTINUARE"
300 X =  PDL (0) / 255 * 278:Y =  PDL (1) / 255 * 190
309  REM TRACCIA
310  HPLOT 1,Y TO X,Y: HPLOT 278,Y TO X,Y: HPLOT X,1 TO X,Y: HPLOT
     X,190 TO X,Y
319  REM RITRACCIA
320  HPLOT 1,Y TO X,Y: HPLOT 278,Y TO X,Y: HPLOT X,1 TO X,Y: HPLOT
     X,190 TO X,Y
329  REM PREMUTO TASTO?/AZZERA TASTIERA
330  IF  PEEK (49152) > 127 THEN A =  PEEK (49168): GOTO 400
340  GOTO 300
399  REM COMMUTA FULLSCREEN
400 A =  PEEK (49234)
410  FOR I = 1 TO 190: HPLOT 0,I TO I,I: HPLOT 279,191 - I TO 27
     9 - I,191 - I: NEXT
420  FOR I = 1 TO 190: HPLOT 0,I TO I,I: HPLOT 279,191 - I TO 27
     9 - I,191 - I: NEXT
425  FOR I = 1 TO 1000: NEXT : REM PAUSA
```

```
50  HOME
100  HGR2 : HGR : HCOLOR= 3
110  PRINT  CHR$ (4)"BLOAD PIC.DUE,A$2000"
120 A =  PEEK (49234):A =  PEEK (49237)
150 DX = 175:DY = 51
160 I = 0
200  FOR S = 0 TO 3.14 STEP 0.02:X = 50 *  COS (S):Y = 50 *  SIN
     (S)
220  HPLOT DX + X,DY + Y
230 A =  PEEK (224):B =  PEEK (225):C =  PEEK (226)
240  POKE 28672 + I,A: POKE 28928 + I,B: POKE 29184 + I,C
250  HPLOT  TO DX - X,DY - Y
260 A =  PEEK (224):B =  PEEK (225):C =  PEEK (226)
270  POKE 29440 + I,A: POKE 29696 + I,B: POKE 29952 + I,C
280 I = I + 1: NEXT
290  PRINT  CHR$ (4)"BSAVETABELLA2,A$7000,L$600
300  TEXT : PRINT I
```

Listato 2 - Programma che genera la tabella necessaria al plottaggio veloce dei disegni.

```
430  HOME : TEXT : VTAB (22): PRINT "ESEMPI DI HPLOT DA PAG.1 A
     PAG.2"
440  REM PAUSA
450  FOR I = 1 TO 5000: NEXT
500  REM MODIFICA HPLOT
510  POKE 62554,169: POKE 62555,0: POKE 62556,81: POKE 62557,38:
      POKE 62558,61: POKE 62559,185: POKE 62560,244: POKE 62561,
     32: POKE 62562,0: POKE 62563,96
520  REM MODIFICA HPLOT TO(HLINE)
530  POKE 62867,32: POKE 62868,0: POKE 62869,97: POKE 62870,234
540  REM MODIFICA DRAW/XDRAW
550  POKE 62660,32: POKE 62661,0: POKE 62662,97: POKE 62663,234
560  HOME : HGR2 : HGR : HCOLOR= 3: VTAB (22): PRINT "QUESTA E'
     LA PAG.1, LA PROSSIMA E' LA 2"
570  PRINT D$"BLOAD PIC.DUE,A$2000"
580  PRINT D$"BLOAD EFFETTO.OBJ3": PRINT D$"BLOAD EFFETTO.OBJ6"
589  REM COMMUTA FULLSCREEN/PAG.2
590  PRINT D$"BLOAD PIC.UNO,A$4000":A =  PEEK (49234):A =  PEEK
     (49237)
595 X = 115:Y = 110
599  REM DISEGNA ROMBO
600  FOR I = 0 TO 60: HPLOT X + I,Y - I TO X + I,Y + I: HPLOT 23
     5 - I,Y - I TO 235 - I,Y + I: NEXT
605 DX = 175:DY = 51
609  REM DISEGNA CERCHIO
610  PRINT  CHR$ (7): FOR I = 0 TO 3.14 STEP 0.02:X = 50 *  COS
     (I):Y = 50 *  SIN (I): HPLOT DX + X,DY + Y TO DX - X,DY - Y
     : NEXT : PRINT  CHR$ (7)
619  REM DISEGNA RETTANGOLO
620  FOR I = 0 TO 30: HPLOT 200,I TO 275,I: NEXT
629  REM RIEMPI TUTTO LO SCHERMO
630  FOR I = 0 TO 191: HPLOT 0,0 TO 279,I: NEXT
640  FOR I = 278 TO 0 STEP  - 1: HPLOT 0,0 TO I,191: NEXT
649  REM PAUSA
650  FOR I = 1 TO 1000: NEXT : TEXT : HOME
659  REM TRACCIA LINEE PRECALCOLATE
660  VTAB (22): PRINT "SE VUOI TRACCIARE PIU' VELOCEMENTE .."
670  PRINT D$"BLOADTABELLA2": PRINT D$"BLOAD DRAWLINE.OBJ0"
680  POKE 252,157: REM $FC=N.LINEE
690  HGR2 : HGR : PRINT D$"BLOAD  PIC.DUE,A$2000"
699  REM  COMMUTA FULLSCREEN/PAG.2
700 A =  PEEK (49234):A =  PEEK (49237)
704  REM CHIAMA DRAWLINE AT $8000
720  PRINT  CHR$ (7): CALL 32768: PRINT  CHR$ (7)
729  REM PAUSA
730  FOR I = 1 TO 10000: NEXT
740  TEXT : HOME : PRINT "POKE 64671,160 PER SCHERMO NORMALE": POKE
     34,2
750  PRINT D$"CATALOG": END
```

Listato 1 - Programma che implementa e prova le nuove routine grafiche. Per funzionare devono essere presenti sullo stesso disco due disegni qualsiasi del nome PIC.UNO e PIC.DUE. Occorre inoltre aver già fatto girare il programma che genera TABELLA 2 (listato 2).

```
5  HOME : VTAB 24
10  PRINT  CHR$ (4)"BLOAD PIXEL.OBJ2"
20  POKE 53637,80: POKE 53638,73: POKE 53639,88: POKE 53640,69: POKE
     53641,204
100  HGR
110  HCOLOR= 3
120  HPLOT 0,0: CALL 62454
130  FOR I = 10 TO 20
140  STORE I,10:PX%
150  PRINT PX%
155  IF PX% THEN  PRINT  CHR$ (7):
160  NEXT
170  TEXT
180  LIST 130 - 160
```

Listato 3 - Programma che prova il nuovo comando PIXEL; prima di lanciarlo occorre far girare ROM > L.CARD.OBJ0.

Commenti

Per commutare tra la ROM e la Language Card basta effettuare le seguenti operazioni

ROM → LC POKE 49280,0
LC → ROM POKE 49282,0

Attenzione al fatto che sull'Apple IIc la locazione 64671 non contiene il Blank per cui effettuando la modifica suggerita nell'ultima parte dell'articolo il computer va in blocco.

Per chi fosse interessato ad approfondire il discorso sul tracciamento veloce di disegni predefiniti consigliamo di rileggersi gli articoli di Roberto Angeletti (MC nn. 35 e 37) sul suo ANNA animation language. MC

```
ROM>L.CARD.OBJ0,A$300,L$21

0300-  AD B1 C0   LDA   $C0B1
0303-  AD B1 C0   LDA   $C0B1
0306-  A2 00      LDX   #$00
0308-  86 3C      STX   $3C
030A-  A9 D0      LDA   #$D0
030C-  85 3D      STA   $3D
030E-  A9 FF      LDA   #$FF
0310-  85 3E      STA   $3E
0312-  85 3F      STA   $3F
0314-  A1 3C      LDA   ($3C,X)
0316-  81 3C      STA   ($3C,X)
0318-  20 BA FC   JSR   $FCBA
031B-  90 F7      BCC   $0314
031D-  AD B1 C0   LDA   $C0B1
0320-  60         RTS

DRAWLINE.OBJ0,A$8000,L$28

8000- A5 FC F0 26 AA BD 00 72
8008- 48 BC 00 71 BD 00 70 AA
8010- 68 20 57 F4 A6 FC BD 00
8018- 73 48 BC 00 75 BD 00 74
8020- AA 68 20 3A F5 C6 FC 4C
8028- 00 80 60

EFFETTO3.OBJ0,A$6000,L$24

6000-  85 FF      STA   $FF
6002-  A5 27      LDA   $27
6004-  18         CLC
6005-  69 20      ADC   #$20
6007-  85 27      STA   $27
6009-  BD B9 F4   LDA   $F4B9,X
600C-  49 FF      EOR   #$FF
600E-  85 FE      STA   $FE
6010-  A9 00      LDA   #$00
6012-  51 26      EOR   ($26),Y
6014-  25 FE      AND   $FE
6016-  05 FF      ORA   $FF
6018-  29 7F      AND   #$7F
601A-  91 26      STA   ($26),Y
601C-  38         SEC
601D-  A5 27      LDA   $27
601F-  E9 20      SBC   #$20
6021-  85 27      STA   $27
6023-  60         RTS
6024-  00         BRK

EFFETTO6.OBJ0,A$6100,L$2D

6100-  B1 26      LDA   ($26),Y
6102-  85 FD      STA   $FD
6104-  08         PHP
6105-  A5 27      LDA   $27
6107-  18         CLC
6108-  69 20      ADC   #$20
610A-  85 27      STA   $27
610C-  A5 FD      LDA   $FD
610E-  25 30      AND   $30
6110-  85 FE      STA   $FE
6112-  A5 30      LDA   $30
6114-  49 FF      EOR   #$FF
6116-  85 FF      STA   $FF
6118-  A9 00      LDA   #$00
611A-  51 26      EOR   ($26),Y
611C-  25 FF      AND   $FF
611E-  05 FE      ORA   $FE
6120-  29 7F      AND   #$7F
6122-  91 26      STA   ($26),Y
6124-  38         SEC
6125-  A5 27      LDA   $27
6127-  E9 20      SBC   #$20
6129-  85 27      STA   $27
612B-  28         PLP
612C-  60         RTS

PIXEL.OBJ2,A$F39F,L$1D

F39F-  20 B9 F4   JSR   $F4B9
F3A2-  20 11 F4   JSR   $F411
F3A5-  B1 26      LDA   ($26),Y
F3A7-  25 30      AND   $30
F3A9-  29 7F      AND   #$7F
F3AB-  48         PHA
F3AC-  20 B1 00   JSR   $00B1
F3AF-  20 E7 DF   JSR   $DFE7
F3B2-  A9 00      LDA   #$00
F3B4-  48         TAY
F3B5-  68         PLA
F3B6-  91 26      STA   ($26),Y
F3B8-  60         RTS
F3B9-  EA         NOP
F3BA-  EA         NOP
F3BB-  EA         NOP
```

Figura 4 - Routine di L.M. delle modifiche all'interprete Applesoft. Per salvare la routine PIXEL occorre far girare ROM > L.CARD.OBJ0, poi battere F39F <839F.83BBM e quindi la BSAVE.

Errata corrige

Nel programma Adventure (MC n. 38) c'è un errore. Per rimediare ecco il consiglio dell'autore.

Caricare ADVENTURE, e in modo diretto scrivere...

```
1460 RUN
DEL 1470,1490
DEL 1740,1770
```

Poi salvarlo con lo stesso nome.

In questo modo ogni volta che si rigioca un'avventura bisogna fare la "fatica" di ricaricarla, ma in compenso si ha a disposizione della memoria in più.

Smau: il giro del mondo in 91.000 metri quadrati

Smau: chi lo visita farà un entusiasmante giro del mondo in 91.000 mq. Qui infatti troverà tutte le novità dei più importanti produttori mondiali. Qui troverà esperti capaci di consigliare le soluzioni più aderenti al futuro dell'azienda e dell'organizzazione del lavoro. Troverà la 18ª edizione del Premio Smau Industrial Design; troverà Convegni e Seminari; troverà lo Spazio Giovani. Troverà il mondo intero: tutto racchiuso in 91.000 metri quadrati.

+ × = smau

22º Salone Internazionale per l'Ufficio: sistemi per l'informatica, la telematica, le comunicazioni, macchine, arredamento per l'ufficio

ENTE GESTIONE MOSTRE COMUFFICIO

Quartiere Fiera Milano
19-24 Settembre 1985

Contemporaneamente, 3ª EIMU, Esposizione Internazionale Mobili Ufficio