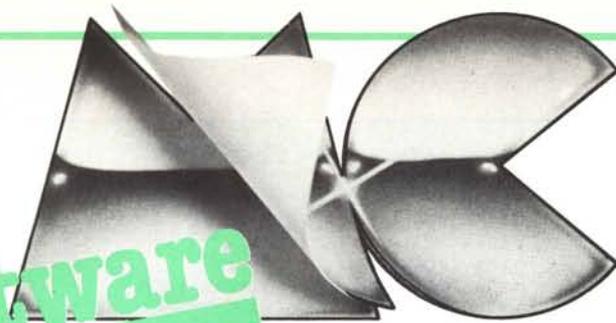


# software

## TI-99/4A



### Scrabble

di Pietro Vigone - Borgo Vercelli

Premettiamo innanzitutto che il computer non prende parte attiva nel gioco, ma si limita a svolgere la funzione di un tabellone elettronico che faciliti la composizione delle parole, e a calcolare e tenere il conto dei punteggi. Nonostante ciò il programma risulta assai interessante ed estremamente curato per quanto riguarda la grafica e la flessibilità e comodità d'uso; esso consente al possessore del TI 99/4A di poter giocare ad

uno dei più apprezzati e diffusi giochi da tavolo... senza dover comprare la relativa scatola, assai costosa e priva di alcune comodità come il calcolo automatico del punteggio.

Lo Scarabeo (Scrabble nei paesi anglosassoni) può essere giocato, in questa versione, da due fino a sei persone; consiste nel formare su un tavoliere parole leggibili in orizzontale o verticale, come negli schemi di parole crociate. Ogni giocatore è contraddistinto dal diverso colore del 'leggio', cioè di quella linea posta sotto il tavoliere dove compaiono i messaggi del calcolatore per l'utente.

I — Composizione della parola — Una volta apparse sul leggio le lettere a disposizione del giocatore, egli può provare a comporre parole mediante la tastiera. Ogni lettera viene stampata maiuscola se il giocatore la possiede, minuscola in caso contrario. Mediante i tasti posti in basso a destra, con i segni di 'minore' e 'maggiore', si può spostare il cursore per effettuare correzioni; con la barra spaziatrice si cancella completamente. Lo scarabeo ha le funzioni di una matta, cioè chi lo possiede lo può usare al posto in una qualsiasi altra lettera. Il calcolatore provvede automaticamente a usarlo quando necessario, tuttavia è possibile impostarlo da tastiera: una volta portato il cursore sotto la lettera da sostituire, che deve già essere stata impostata, va premuto il tasto posto vicino ad — ENTER —, che riporta il segno dei ':'. Se vi sono scarabei sul tavoliere, ogni giocatore al proprio turno può appropriarsene sostituendoli con la lettera di cui fanno le veci; ciò viene realizzato impostando la lettera e premendo ':'.  
Se tutte le lettere che appaiono sul leggio

Se tutte le lettere che appaiono sul leggio

```

100 REM SCRABBLE
110 REM PVigone fe84
120 CALL SCREEN(5)
130 DIM TBL(16,16),LETT$(5),PT(5),COL(5),VL(26),SCAR
(1,2),TV(1,7)
140 DATA 30,1,4,1,4,1,4,1,4,4,8,1,0,0,2,2
150 DATA 2,1,3,10,1,1,1,4,4,0,0,0,8
160 FOR L=0 TO 26
170 READ VL(L)
180 NEXT L
190 DATA 6,12,9,4,8,2
200 FOR L=0 TO 5
210 READ COL(L)
220 NEXT L
230 RESTORE 380
240 CALL CLEAR
250 CALL COLOR(1,2,5)
260 FOR R=2 TO 12
270 CALL COLOR(R,16,5)
280 NEXT R
290 PRINT " SCRABBLE": :
300 INPUT "QUANTI GIOCATORI ? (2/6) ":PLYRS
310 IF (PLYRS<2)+(PLYRS>6) THEN 300
320 CALL CLEAR
330 REM STAMPA SCHERMO
340 PRINT TAB(7);"S C R A B B L E"
350 CALL CHAR(33,"007C82828282827C")
360 CALL CHAR(62,"007C7C7C7C7C")
370 PRINT :
380 DATA "PUNTI",,"ACE 1","IOR 1","ST 1",,"LMN 2",
,"P 3",,"BDF 4","GUV 4",,"
HZ 8",,"Q 10"
390 FOR P=1 TO 17
400 READ ST$
410 PRINT " !!!!!!!!!!!!!!!!!!!!!!! ";ST$
420 NEXT P
430 PRINT : : :
440 CALL CHAR(136,"0048A828484888EE")
450 CALL CHAR(137,"0048A8284828A84E")
460 CALL CHAR(144,"00040AA244A4080E")
470 CALL CHAR(145,"00040AA244A20A04")
480 CALL COLOR(14,2,12)
490 CALL COLOR(15,2,10)
500 CALL CHAR(64,"0028927CBA7CBA92")
510 CALL CHAR(34,"0028927CBA7CBA92")
520 REM CASELLE SPEC.
530 CH=144
540 FOR P=1 TO 5
550 @=P
560 GOSUB 780
570 NEXT P
580 DATA 0,4,7,2,3,8,7,7,6,6,6,1
590 CH=136
600 FOR R=1 TO 4
610 READ P,@
620 GOSUB 780
630 NEXT R
640 CH=137
650 FOR R=1 TO 2
660 READ P,@
670 GOSUB 780
680 NEXT R
690 FOR P=0 TO 16 STEP 8
700 FOR @=0 TO 16 STEP 8
710 CALL HCHAR(P+3,@+6,145)
720 TBL(P,@)=145
730 NEXT @
740 NEXT P
750 TBL(8,8)=0
760 CALL HCHAR(11,14,34)
770 GOTO 950
780 REM ===== ==
790 REM SUB CALC. SIMMET.
800 GOSUB 880
810 P=16-P
820 GOSUB 880
830 @=16-@
840 GOSUB 880
850 P=16-P
860 GOSUB 880
870 RETURN
880 REM SUB POSIZ. CH
890 CALL HCHAR(P+3,@+6,CH)
900 CALL HCHAR(@+3,P+6,CH)
910 TBL(P,@)=CH
920 TBL(@,P)=CH
930 RETURN
940 REM ===== ==
950 REM INIZ. VARIABILI
960 CALL CHAR(61,"007FE")
970 CALL CHAR(129,"FF")

```

(continua a pag. 148)



```

2470 IF (C=32)+(C>64)*(C<91)THEN 3410
2480 FOR S=0 TO 1
2490 IF SCAR(S,0)=R THEN 2530
2500 NEXT S
2510 P=POS(RIM$,CHR$(D),1)
2520 IF P<>0 THEN 2640
2530 P=POS(RIM$,"S",1)
2540 IF P=0 THEN 3410
2550 FOR Q=0 TO 1
2560 IF (SCAR(Q,0)<>0)*(SCAR(Q,0)<>R)THEN 2620
2570 SCAR(Q,0)=D
2580 SCAR(Q,1)=Y
2590 SCAR(Q,2)=X
2600 SCR=1
2610 GOTO 2640
2620 NEXT Q
2630 GOTO 3410
2640 RIM$=SEG$(RIM$,1,P-1)&SEG$(RIM$,P+1,LEN(RIM$)-1)
2650 CALL HCHAR(Y,X,D)
2660 EN=EN+1
2670 TV(0,EN)=Y
2680 TV(1,EN)=X
2690 X=X+SX
2700 Y=Y+SY
2710 NEXT R
2720 REM ASSEGNAZ. PUNTI
2730 Y=TV(0,0)
2740 X=TV(1,0)
2750 PUNTI=0
2760 GOSUB 3040
2770 IF PUNTI=0 THEN 3410
2780 SY=1-SY
2790 SX=1-SX
2800 FOR D=0 TO EN
2810 Y=TV(0,D)
2820 X=TV(1,D)
2830 GOSUB 3040
2840 IF PUNTI=0 THEN 3410
2850 NEXT D
2860 LETT$(PL)=RIM$
2870 IF EN<5 THEN 2940
2880 PUN=(EN-4)*20+Q*SCR
2890 LEG$="ABBUONO DI "&STR$(PUN)
2900 GOSUB 3920
2910 CALL SOUND(400,500,3)
2920 CALL SOUND(500,700,3)
2930 PUNTI=PUNTI+PUN
2940 PT(PL)=PT(PL)+PUNTI
2950 LEG$="punteggio : "&STR$(PT(PL))
2960 GOSUB 3920
2970 FOR R=1 TO 50
2980 NEXT R
2990 IF (LETT$(PL)="")*(SAC$="")THEN 3560
3000 CALL HCHAR(22,1,32,96)
3010 FIRST=0
3020 NEXT PL
3030 GOTO 1030
3040 REM =====
3050 REM VALIDAZ. E P.TI
3060 LEG$=""
3070 PUN=0
3080 Q=1+FIRST
3090 Y=Y-SY
3100 X=X-SX
3110 CALL GCHAR(Y,X,C)
3120 IF (C>63)*(C<91)THEN 3090
3130 X=X+SX
3140 Y=Y+SY
3150 P=1
3160 CALL GCHAR(Y,X,C)
3170 IF (C<64)+(C>90)THEN 3250
3180 LEG$=LEG$&CHR$(C)
3190 IF (TBL(Y-3,X-6)<>136)*(TBL(Y-3,X-6)<>137)THEN
3210
3200 P=TBL(Y-3,X-6)-134
3210 IF TBL(Y-3,X-6)<144 THEN 3230
3220 Q=Q*(TBL(Y-3,X-6)-142)
3230 PUN=PUN+VL(C-64)*P
3240 GOTO 3130
3250 IF LEN(LEG$)<2 THEN 3370
3260 PUN=PUN*Q
3270 IF LEG$<>"SCARABEO" THEN 3290
3280 PUN=PUN+110-10*SCR
3290 LEG$=LEG$&" "&STR$(PUN)
3300 GOSUB 3920
3310 CALL SOUND(150,500,3)
3320 CALL KEY(3,R,N)
3330 IF R=78 THEN 3380
3340 IF R<>83 THEN 3320
3350 CALL SOUND(300,400,3)
3360 PUNTI=PUNTI+PUN
3370 RETURN
3380 PUNTI=0
3390 GOTO 3370
3400 REM =====
3410 REM CANCELLAZIONE
3420 FOR R=0 TO EN
3430 Y=TV(0,R)
3440 X=TV(1,R)
3450 P=TBL(Y-3,X-6)
3460 IF P<>0 THEN 3480
3470 P=33
3480 CALL HCHAR(Y,X,P)
3490 FOR Q=0 TO 1
3500 IF (SCAR(Q,1)<>Y)+(SCAR(Q,2)<>X)THEN 3520
3510 SCAR(Q,0)=0
3520 NEXT Q
3530 NEXT R
3540 LEG$=LETT$(PL)
3550 GOTO 1130
3560 REM FINE PARTITA
3570 FOR P=0 TO PLYRS-1
3580 IF P=PL THEN 3660
3590 PUN=0
3600 FOR R=1 TO LEN(LETT$(P))
3610 Q=ASC(SEG$(LETT$(P),R,1))
3620 PUN=PUN+VL(Q-64)
3630 NEXT R
3640 PT(P)=PT(P)-PUN
3650 PT(PL)=PT(PL)+PUN
3660 NEXT P
3670 CALL CLEAR
3680 FOR P=0 TO PLYRS-1
3690 CALL SOUND(500,200+P*200,3)
3700 CALL COLOR(10+P,COL(P),COL(P))
3710 PRINT : : :
3720 CALL HCHAR(23,6,P*8+110,22)
3730 PRINT TAB(10);PT(P);"PUNTI"
3740 CALL HCHAR(24,6,P*8+110,22)
3750 NEXT P
3760 PRINT : :
3770 FOR Q=1 TO 4000
3780 NEXT Q
3790 PRINT "UN'ALTRA PARTITA ? (S/N)"
3800 CALL KEY(3,R,N)
3810 IF R=83 THEN 3840
3820 IF R<>78 THEN 3800
3830 STOP
3840 FOR P=0 TO 5
3850 LETT$(P)=""
3860 PT(P)=0
3870 NEXT P
3880 SCAR(0,0)=0
3890 SCAR(1,0)=0
3900 GOTO 230
3910 REM =====
3920 REM SUB PRINT LEG$
3930 CALL COLOR(13,5,COL(PL))
3940 CALL HCHAR(21,7,129,20)
3950 FOR P=1 TO LEN(LEG$)
3960 CALL HCHAR(21,16-LEN(LEG$)/2+P,ASC(SEG$(LEG$,P,1)))
3970 NEXT P
3980 RETURN

```

sono solo vocali o solo consonanti, senza scarabei, il giocatore può farle sostituire, saltando però il turno; per farlo si imposta la parola-chiave 'CMB' e si preme — ENTER —. Mediante la parola-chiave 'PSS' si passa la mano. Impostata la propria parola, premendo — ENTER — si passa alla fase successiva.

2 — Posizionamento della parola — Una volta composta, la parola va disposta sul tavoliere. Il primo giocatore la dispone in orizzontale o in verticale, in modo che copra lo scarabeo centrale. Ogni successiva parola deve avere in comune almeno una lettera con le parole già esistenti, e anche questa viene conteggiata nel punteggio; ci si può addirittura limitare ad aggiungere qualche lettera a una parola già presente, per esempio trasformando CONTA in SCONTARE, totalizzando sempre il punteggio della parola intera. È fondamentale che ogni lettera che sia adiacente ad altre forme con esse parole ammesse; ad esempio, poniamo che trovandosi già sul tavoliere le parole SCONTARE e CANNA, venga composta STANCO:

S	C	O	N	T	A	R	E
T	A						
A	N						
N	N						
C	A						
O							

in questo caso devono essere valide, e lo sono infatti, le parole TA (Taranto), AN (Ancona), NN (cioè 'nessuno' e CA (Cagliari). Anche queste parole vengono conteggiate nel punteggio.

Quando si è premuto — ENTER —, compare al centro dello schermo un quadratino lampeggiante; esso deve essere portato mediante i quattro tasti con le frecce E, S, D e X nella casella in cui va posta la prima lettera della parola impostata. Premendo la lettera O, la parola viene scritta in orizzontale; premendo la V in verticale. Se ci si accorge di un errore, premendo lo — SPAZIO — si ritorna alla fase precedente. Il calcolatore non controlla che venga usata almeno una lettera preesistente, lasciando questo compito ai giocatori; è così possibile evitare di scrivere la parola completa, quando si vuole solo aggiungere un prefisso o suffisso a una già scritta.

Sul tavoliere esistono caselle speciali contrassegnate con 2L, 3L, x2 e x3. Le prime raddoppiano o triplicano il valore della lettera che vi si trova, le altre raddoppiano e triplicano il punteggio di tutta la parola. Se una parola copre due o più di queste ultime, il punteggio totalizzato viene moltiplicato rispettivamente per quattro e per nove. Viene raddoppiato anche il punteggio della prima parola posta sul tavoliere.

3 — Convalida e punteggio — Dopo che il calcolatore ha scritto la parola impostata sul tavoliere, presenta una alla volta le parole che si vengono a formare, scrivendone anche il valore. I giocatori devono control-

lare che esse siano ammesse, e in caso affermativo premere S. Premendo N invece le lettere appena disposte vengono cancellate e si torna alla prima fase.

Sono ammesse tutte le parole della lingua italiana contemplate in un qualsiasi dizionario, sia al maschile che al femminile, al singolare che al plurale; tutte le coniugazioni dei verbi, anche con prefissi e suffissi come PARTENDOSENE, PREAVVERTITECI, ecc., le sigle più comuni e quelle delle province italiane. Non sono ammesse voci arcaiche, dialettali, poetiche, diminutivi, vezzeggiativi ecc. né i nomi propri, né quelli storici o geografici e neppure le parole del gergo tecnico o scientifico.

Viene infine presentato il punteggio totale. Vi sono premi particolari se si usano in una sola volta 8, 7 o 6 lettere, rispettivamente di 50, 30 e 10 punti; chi compone poi la parola SCARABEO guadagna 100 punti. Tutti questi premi sono maggiorati di 10 se non è stato usato lo scarabeo.

La mano passa ora al giocatore successivo; le lettere non utilizzate rimangono e verranno reintegrate nella mano successiva fino al numero di otto.

4 — Fine del gioco — Quando sono esaurite tutte le lettere disponibili, si continua con quelle rimaste ai giocatori. La partita finisce quando uno di essi rimane senza più lettere. A questo punto questi totalizza un punteggio pari al valore delle lettere rimaste ai suoi avversari, mentre a questi viene sottratto l'equivalente delle lettere che ancora possiedono. Lo scarabeo viene conteggiato per trenta punti. I risultati vengono calcolati automaticamente e vengono stampati i vari punteggi, ognuno contornato da strisce del colore che contraddistingue il giocatore. Vince chi, indipendentemente dalla chiusura, totalizza il punteggio più alto.

Valore e numero delle lettere					
	punti	quantità		punti	quantità
A	1	12	N	2	6
B	4	4	O	1	12
C	1	7	P	3	4
D	4	4	Q	10	2
E	1	12	R	1	7
F	4	4	S	1	7
G	4	4	T	1	7
H	8	2	U	4	4
I	1	12	V	4	4
L	2	6	Z	8	2
M	2	6	scarabeo (30)		2

### Analisi del programma

Il programma è organizzato in blocchi che vengono eseguiti in sequenza; una REM posta all'inizio di ogni blocco indica la funzione che esso svolge.

Meglio che dall'analisi del listato, il funzionamento si comprende dalla spiegazione dell'uso delle variabili.

Le matrici LETT\$(5), PT(5), COL(5) contengono rispettivamente le lettere rimaste ad ogni giocatore dalla mano prece-

dente, il punteggio ed il codice del colore che contraddistingue i giocatori.

VL(26) contiene il valore delle singole lettere. VL e COL vengono inizializzate alle linee 140-220.

TBL(16,16) contiene alle locazioni corrispondenti alle caselle speciali il codice del carattere che le rappresenta sullo schermo. Viene sfruttata la simmetria della posizione di queste caselle per l'inizializzazione che avviene alle linee 520-940 contemporaneamente alla loro stampa sullo schermo.

SAC\$ rappresenta il 'sacchetto' dal quale vengono estratte le lettere. Ogni lettera vi compare un numero di volte pari alla quantità riportata nella tabellina. A causa della sua lunghezza questa stringa è inizializzata in due istruzioni, la 1000 e la 1010.

LEG\$ contiene la stringa da stampare sul leggìo, e funziona quindi da parametro per la subroutine di linea 3920. Nelle linee da 1150 a 1830 viene utilizzata per determinare le lettere di cui dispone il giocatore per comporre la sua parola.

PAR\$ è la parola composta dal giocatore, mentre RIMS contiene le lettere rimaste inutilizzate.

In SCAR(x,0) è memorizzato il codice ASCII della lettera sostituita dallo scarabeo, in SCAR(x,1) e SCAR(x,2) le coordinate della locazione dello schermo dove esso si trova. Quando però SCAR(x,0) contiene un numero minore di 20, questo indica la posizione della lettera da sostituire con lo scarabeo nella parola che è stata appena impostata. Lo scarabeo è rappresentato dal carattere di codice 64, cioè la 'chiocciolina'.

SCR è un flag, posto a 1 se è stato usato uno scarabeo nell'ultima parola impostata.

TV(1,7) contiene le coordinate delle lettere che sono state poste sul tavoliere nella mano corrente. EN è il numero di queste lettere meno uno.

X e Y sono le coordinate generiche di una locazione di schermo.

SX ed SY contengono l'incremento da dare ad esse per determinare la successiva locazione da prendere in esame.

Nella battitura del programma va posta attenzione alle linee 380 e 410. Nella prima, l'istruzione DATA deve essere scritta così com'è, senza saltare virgole; la stringa della seconda è formata da: 3 spazi, 17 punti interrogativi, 2 spazi. I colori che ho usato sono quelli che danno un'immagine migliore sul mio TV; su TV diversi, o particolarmente su monitor monocromatici, possono non essere tali, ma modificarli non dovrebbe dare alcuna difficoltà.

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 120.

## Usare le sprite col TI Basic!

di Riccardo Tesio - Cuneo

*Credevamo che oramai fosse stato detto tutto; che fosse già stato svelato, scritto e riscritto tutto quello che c'era di segreto, anormale, curioso, non-scritto-sui-manuali, ed in ogni caso utile, riguardo al sistema operativo ed al basic del TI 99 in versione non espansa. Eravamo giunti alla conclusione che con il TI 99 base non fosse proprio possibile ottenere di più di ciò che era previsto dal basic residente e descritto nel manuale. E invece le cose non stanno proprio così, e con grande piacere pubblichiamo, in parte in questo numero, in parte nel prossimo, la ricca documentazione inviataci da questo lettore di Cuneo; ma, saltando ogni altro convenevole, passiamo a raccontarvi le novità.*

Nei manuali forniti dalla Texas Instruments insieme al TI 99 ed ai suoi accessori viene più volte ripetuto che non è possibile utilizzare le sprite a meno di non avere a disposizione il modulo Extended Basic oppure quello Mini Memory, o peggio ancora sistema a dischi, espansione di memoria e assembler. Come potrete rendervi conto tra poco questa informazione è vera solo in parte. Vedrete come, ingannando il sistema operativo e l'interprete Basic, è possibile ottenere e manipolare le sprite anche con il TI Basic residente e con la macchina in configurazione minima avendo come unica espansione il registratore a cassette. Vi avvertiamo subito a scanso di equivoci che le sprite così ottenute non hanno la stessa versatilità di quelle producibili con l'extended Basic il quale fornisce una gamma veramente completa di istruzioni per muoverle in ogni direzione in maniera indipendente dal programma, ingrandirle, verificarne la posizione o la sovrapposizione. Nonostante ciò queste sprite ottenibili con il solo TI Basic godono di alcune notevoli proprietà: non vengono influenzate dagli

altri caratteri i quali vi scorrono sotto senza cancellarle, possono essere poste in un qualsiasi punto dello schermo all'interno di un sistema di coordinate con  $192 \times 256$  posizioni, permettono di realizzare, sebbene attraverso un apposito programma, il movimento di oggetti "pixel per pixel", evitando il fastidioso effetto dello sfarfallio che si ha spostando una figura all'interno del reticolo  $24 \times 32$  delle posizioni di stampa.

Ci sembra che ciò ampli di molto le possibilità del programmatore.

È possibile per esempio sfruttando il fatto che sprite e caratteri normali si sovrappongono senza cancellarsi tra loro, poter disegnare scenari e fondali per i vostri giochi senza che questi vengano cancellati durante l'azione.

Prima di cominciare a trattare l'argomento vero e proprio, consigliamo a coloro che posseggono qualche espansione del computer e che vogliono sperimentare quanto detto nell'articolo, di disconnettere ogni dispositivo esterno alla macchina, tranne il registratore a cassette. Questo in quanto un dispositivo esterno può alterare la mappa della memoria del computer.

In questo numero ci soffermeremo principalmente sul come creare le sprite, sul prossimo daremo una ampia spiegazione sul perché è possibile mediante i sotterfugi appresso spiegati ottenere l'effetto desiderato. Se quindi qualcuno troverà a tratti oscuro l'argomento di questo articolo, abbia la pazienza di aspettare un mese e tutto sarà spiegato in dettaglio.

È possibile utilizzare le sprite spostando la loro mappa di definizione (la cosiddetta SPRITE ATTRIBUTE LIST) nella memoria, sovrapponendola a quella dei caratteri definibili dall'utente. Per fare questo dobbiamo agire sui VDP write only registers. Il TI Basic non ci fornisce nessun mezzo per farlo (ciò è permesso solo a chi possiede il modulo Mini Memory median-

te l'istruzione CALL PEEKV come vedremo tra poco). Esiste però una maniera per arrivare lo stesso ai VDP register, essa sfrutta la particolare maniera con cui vengono registrati i programmi dal TI 99. Quando un programma viene riversato su cassetta esso è preceduto da una sorta di "header" di alcuni byte che contengono alcune informazioni sulle dimensioni del programma. Manipolando queste informazioni è possibile oltrepassare il limite inferiore della memoria ed andare a scrivere nei VDP register.

Poiché le modalità con cui vengono registrati i file su cassetta sono analoghe a quelle per i programmi, noi ci serviremo di un record di un file per scrivere un falso header. Ci occorrerà dunque un programma generatore di file per poter produrre un record su cassetta da poter poi usare come falso programma. Tale programma generatore è mostrato in figura 1. Nel prossimo numero ne analizzeremo in dettaglio il funzionamento. Grazie all'input di linea 10 si potrebbe virtualmente scrivere in ognuno degli 8 registri del VDP; in pratica solo il 5 è utilizzabile perché gli altri o vengono riportati al valore standard dal sistema operativo (1,2,3,4,7), oppure sono comunque inutilizzabili per deficienze del TI Basic (0,6). Il registro 5 indica il punto d'inizio della mappa delle sprite. Il valore di default è 6 e quest'ultimo viene sempre ripristinato premendo QUIT; inserendo 14 la mappa viene posta in corrispondenza ai caratteri 128...143, inserendo 15 in corrispondenza ai caratteri 144...159. Per i nostri scopi i valori che inseriremo in input sono 5,15.

Per poter utilizzare le sprite bisogna:

1) registrare su cassetta il finto programma ottenuto mediante il programma generatore di record sopra descritto.

2) resettare memoria ram e puntatori mediante QUIT o BYE (indispensabile)

```

10 REM* *
20 REM* GENERATORE FILES *
30 REM* *
40 REM* PER MODIFICARE I *
50 REM* *
60 REM* REGISTRI DEL VDP *
70 REM* *
80 REM* (C) BY TESIOWARE *
90 REM* *
100 CALL CLEAR
110 INPUT "REGISTRO (0-7), DATO (0-255)?  ":
R,D
120 A=18429-(256*R+D)
130 X#=CHR$(0)
140 OPEN #1:"CS1",OUTPUT,FIXED
150 PRINT #1:X$&X$&X$&X$&X$&X$&X$&CHR$(INT(A
/256))&CHR$(A)

```

Figura 1

```

1 REM CONVERSIONE RAPIDA
2 REM DECIMALE-ESADECIMALE
10 DIM A$(15),H$(255)
20 FOR I=0 TO 15
30 A$(I)=SEG$( "0123456789ABCDEF",I+1,1)
40 NEXT I
50 FOR I=0 TO 15
60 K=16*I
70 FOR J=0 TO 15
80 H$(K+J)=A$(I)&A$(J)
90 NEXT J
100 NEXT I

```

Figura 2

3) ricaricare da cassetta il finto programma

4) attendere alcuni istanti; quando lo schermo diventa nero a righe, premere una lettera ed ENTER (se non dà effetti riprovare); a questo punto dovrebbe tornare tutto quasi normale e dovrebbe comparire il messaggio \* MEMORY FULL

5) cancellare il programma con NEW.

Notate che il punto uno non deve essere ripetuto ogni volta, è sufficiente registrare lo pseudoprogramma generatore di sprite una volta per tutte.

Chi possiede il modulo Mini Memory può risparmiarsi tutta questa trafila ed inserire in testa al programma la linea

```
CALL PEEKV(-753,X)
```

Ora la mappa delle sprite è spostata in corrispondenza ai caratteri 144...159. Sono accessibili fino a 32 sprite (2 per ogni carattere); la chiamata è

```
CALL CHAR(144, "YIXIFIC1Y2X2F2C2")
```

```
CALL CHAR(145, "Y3X3F3C3Y4X4F4C4") ecc.
```

dove Y è la riga (0-191), X è la colonna (0-255), F è la forma (codice ASCII+96) C è il colore (codice Basic meno 1). Tutti i valori vanno espressi in esadecimale. Se Y viene posto uguale a 208 (esadecimale D0) quella e tutte le sprite che seguono non vengono visualizzate.

Esempi:

```
CALL CHAR(144, "D") cancella tutte le sprite;
```

```
CALL CHAR(144, "6080A10FD") posiziona una A bianca in mezzo allo schermo;
```

```
CALL CHAR(144, "0080AD800088A306");
```

```
CALL CHAR(145, "D") scrive MC rosso in alto.
```

Scrivendo qualcosa e premendo ripetutamente il tasto ENTER noterete come le scritte scorreranno regolarmente dal basso verso l'alto ma le sprite rimarranno ferme.

Se queste sprite devono muoversi può essere conveniente creare un vettore di 256 elementi che esegua una rapida conversione decimale esadecimale; provate ad esempio il programma di figura 2. Ora è più semplice usare le sprite, basta inserire tale routine di conversione nel programma. Ad esempio la routine di figura 3, abbinata a quella di figura 2, muove verticalmente pixel per pixel una chiocciolina viola.

Come si può vedere la velocità non è eccessiva, ma i pregi di queste sprite sono altri, quali la trasparenza, l'assenza di sfarfallio, i colori indipendenti, la possibilità di una regolazione fine della velocità (provate a considerare come migliorerebbero giochi come "Pecore nere" o "atterraggio" già presentati per il TI 99 su questa rivista). Le altre peculiarità delle TI sprite, quali il movimento automatico e le possibilità di ingrandimento non sono purtroppo accessibili. Come vedete i nuovi orizzonti aperti da queste interessanti "scoperte" sono notevoli e tutto è aperto ad ulteriori sviluppi. A questo punto diviene importante il vostro contributo. Vi invitiamo quindi a provare, a sperimentare, a trovare utilizzazioni per quanto vi abbiamo proposto. Scriveteci le vostre idee e noi pubblicheremo ogni nuovo contributo. Vi preannunciamo che già nel prossimo numero andremo più a fondo nell'argomento e pubblicheremo ulteriori "novità" riguardanti il TI 99 ed inoltre un programma che utilizza le sprite nelle modalità sopra esposte.

```
110 F$=H$(128)&H$(64+96)&H$(14-1)&H$(208)
120 FOR Y=0 TO 191
130 CALL CHAR(144,H$(Y)&F$)
140 NEXT Y
```

Figura 3

## Extended Basic

### Scritte in movimento

La breve routine che vi presentiamo permette di far scorrere sullo schermo da destra verso sinistra una qualunque sequenza di caratteri. Un simile effetto viene spesso usato nei programmi televisivi per realizzare i titoli di coda durante la sigla finale o per ricordare il titolo del film che state vedendo dopo un intermezzo pubblicitario. Anche se supponiamo che ben pochi di voi abbiano a che fare con la televisione, vi presentiamo questa routine, sorprendentemente semplice e breve visto l'effetto che produce, con la quale potrete rendere un po' più sofisticata la presentazione dei vostri programmi.

La routine è realizzata in forma di sottoprogramma (linee 1000-1080) per facilitarne al massimo la "trasportabilità" ma nulla vi vieta di prendere il blocco di istruzioni compreso tra le linee 1010-1070 e inserirlo direttamente nell'ambito di un programma. I due parametri VEL e MSG\$ devono contenere uno la velocità di scorrimento della scritta, l'altro la stringa che deve scorrere sullo schermo. Per la velocità bisogna impostare un valore compreso tra 0 (più lento) e 10 (più veloce); la stringa deve essere lunga un massimo di 227 caratteri. Il

nucleo del programma è costituito dalle linee 1030 e 1040. Come si vede l'effetto di scorrimento è ottenuto mostrando in rapida sequenza porzioni della stringa MSG\$ ottenute traslando di carattere in carattere all'interno della stringa stessa. La linea

1010, anche essa fondamentale, serve a far entrare inizialmente la scritta dall'estremità destra dello schermo; la linea 1040 regola la velocità mentre le linee 1050 e 1060 servono ad uscire dalla routine non appena venga premuto un tasto qualunque. **MC**

```
1 REM MESSAGGI CHE SCORRONO
2 REM SULLO SCHERMO IN
3 REM ORIZZONTALE
5 CALL CLEAR
10 INPUT "VELOCITA' (0-10): ":SPEED
20 FRA$="ATTENZIONE! GUARDARE SCORRERE QU
ESTO MESSAGGIO PER PIU' DI DUE MINUTI P
UO' FARVI MALE AGLI OCCHI --PREMERE UN TA
STO QUALUNQUE--"
30 CALL MESSAGGIO(SPEED,FRA$)
1000 SUB MESSAGGIO(VEL,MSG$)
1010 MSG$=RPT$(" ",28)&MSG$
1020 FOR CICLO=1 TO LEN(MSG$)
1030 DISPLAY AT(24,1):SEG$(MSG$,CICLO,28)
1040 FOR RIT=0 TO 10-VEL :: NEXT RIT
1050 CALL KEY(0,KEY,STATUS)
1060 IF STATUS<>0 THEN 1080
1070 NEXT CICLO :: GOTO 1020
1080 SUBEND
```

# Sin da adesso. Un sistema già completo.

*Spectravideo ha tanta potenza in più*

perchè ha un Basic super esteso che consente di programmare più rapidamente e con più facilità, occupando meno memoria.

*Con Spectravideo disegni i tuoi giochi*

hai a disposizione 32 sprites, 16 colori contemporanei, una serie di comandi specifici ed un macro linguaggio grafico con altri 14 comandi aggiuntivi.

*Spectravideo utilizza tutta la biblioteca CP/M*

è perfettamente compatibile con tutti i suoi programmi senza dover aggiungere costose interfacce.

*Spectravideo è sistema completo*

tutte le periferiche necessarie sono utilizzabili sin dal primo giorno e a basso costo. Tanti "computers system" in realtà non sono un sistema, non comunicano con qualunque altro elaboratore e si completano solo nel tempo.

*Tanta potenza ed espandibilità sanno starti dietro quando crescono le tue esigenze!*

# SVI<sup>TM</sup>

## SPECTRAVIDEO



Distributore per l'Italia COMTRAD Divisione Computers Tel. (0586) 424348 TLX 623481 COMTRAD I