



VIC

da zero

di Tommaso Pantuso



Uno sguardo al video

Oggi parleremo un po' della Ram video del Vic e del C 64. Benchè quanto andremo ad illustrare non rappresenti in generale un fatto nuovo, siamo sicuri di far cosa gradita a quanti ci scrivono chiedendo qualcosa in più su un argomento che sui manuali è trattato molto sinteticamente.

schermo. Tale spazio che nel C 64 è fisso, sul Vic 20 varia a seconda della configurazione di memoria in cui ci troviamo. Il punto di inizio di tale zona in memoria può essere comunque facilmente identificato leggendo il contenuto di una specifica locazione, la 648 (decimale), e moltiplicandolo per 256.

Ad esempio nel Vic base il contenuto di

tale locazione è 30; moltiplicando tale numero per 256 otterremo:

$$30 \times 256 = 7680$$

che è appunto l'inizio della Ram Video la quale si estende da quel punto in su per 512 locazioni, di cui noi ne utilizzeremo solo 506 essendo questa la quantità totale di caratteri che possiamo posizionare sullo schermo.

Ricapitolando, in una certa area che si estende da 7680 a 8191, troveremo della memoria Ram rappresentabile con la griglia della figura 2 e che praticamente è l'immagine di quanto accade sullo schermo. A questo punto viene spontaneo chiedersi il significato della frase 'immagine dello schermo'. Lo spiegheremo con un esempio.

Premete il tasto <Home>: il cursore si porterà nella locazione in alto all'estrema sinistra dello schermo; a questo punto digitate la lettera 'A'; se esiste effettivamente una corrispondenza tra ciò che vediamo sullo schermo Tv e la corrispondente locazione nella Memoria Video di cui abbiamo parlato, con l'operazione eseguita dovremo ritrovare nella locazione 7680 qualcosa che rappresenti in qualche modo la nostra 'A'. Per verificare ciò, spostate il cursore una linea più in basso con il tasto <Crsr> verticale e scrivete:

PRINT PEEK(7680) <Return>
se agite su un Vic in configurazione base.

Vedrete comparire sullo schermo un numero, e non potevamo aspettarci altro visto che la memoria contiene solo rappresentazioni di numeri. Il numero in questo

Le griglie di schermo

Iniziamo il discorso partendo da un Vic 20 in configurazione base ed estendendolo poi al C 64.

Sullo schermo del Vic abbiamo a disposizione 23 righe e 22 colonne e su di esso possiamo imprimere 506 caratteri. Infatti su ciascuna linea potremo scrivere un numero di caratteri pari al numero di colonne e moltiplicando tale numero per il totale delle linee, che è 23, otterremo appunto come risultato 506.

Possiamo allora schematizzare lo schermo del computer come una griglia composta da 506 quadratini (figura 1), ciascuno dei quali rappresenta una possibile posizione di un carattere.

Ci chiediamo ora come poter accedere ad ogni singolo quadratino posizionandovi dentro un ben determinato carattere in maniera diversa da quella consueta.

Esiste in memoria uno spazio che consta di un certo numero di locazioni il quale gestisce tutto quello che viene posto sullo

*Figura 1
Possiamo schematizzare lo schermo del nostro computer come una griglia nella quale ciascun carattere occupa una ben precisa posizione.*

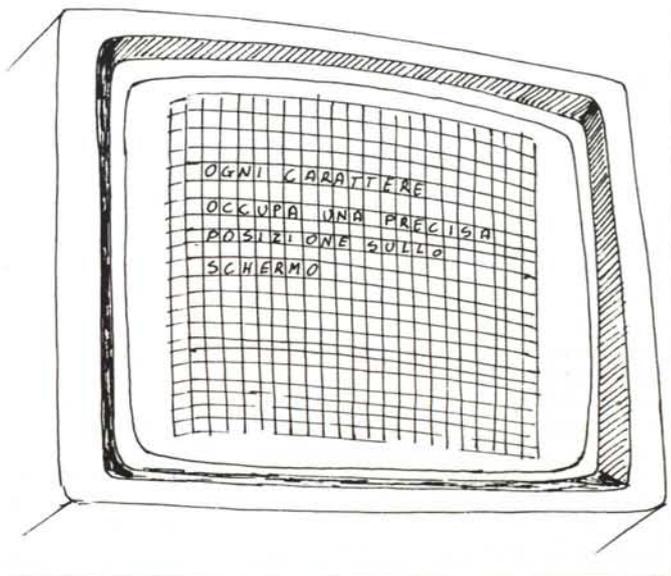




Figura 2 **Mapa di memoria dello schermo**

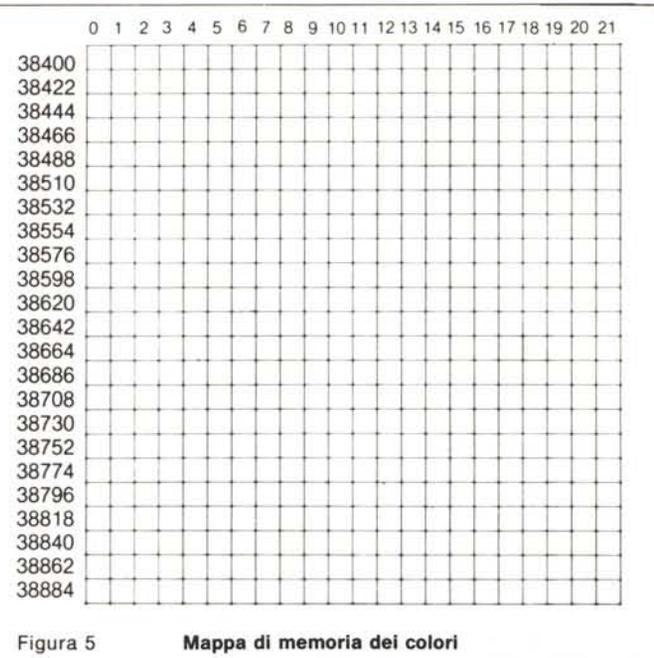


Figura 5 **Mapa di memoria dei colori**

Figura 2 - Alla griglia sullo schermo ne corrisponde un'altra in una certa area di memoria Ram. Se ad esempio scriviamo nella prima posizione di schermo in alto a sinistra, troveremo nella locazione corrispondente della Ram (quella rappresentata in questa figura è del Vic) cioè in 7680 il numero uno che in codice (vedi articolo) rappresenta una 'A'.
 Figura 5 - Mapa della Memoria del Colore per il Vic.

ne è '1' che rappresenta appunto la 'A' in un codice particolare detto 'Codice di Schermo'. Se ripetete l'esperimento con la lettera 'B' vi verrà restituito il numero 2 e così via numeri diversi per ogni carattere impresso in quella locazione. Il codice in questione non è, come è facile osservare, un codice ASCII, ma tra quest'ultimo ed il primo è possibile porre una corrispondenza (figura 2). Se infatti 'S' è il Codice di Schermo ed 'A' è il codice ASCII, tale corrispondenza può essere facilmente espressa con la seguente tabella:

Se A è compr. tra	allora S è compr. tra	rel.
32 e 63	32 e 63	A = S
64 e 95	0 e 31	A = S + 64
96 e 127	64 e 95	A = S + 32
160 e 191	96 e 127	A = S + 64

Quindi, per una verifica, considerando ad esempio il carattere 'B', esso è espresso in Codice di Schermo dal numero 2; essendo tale numero compreso tra 0 e 31, potremo ottenere il codice ASCII di 'B' con la seconda delle relazioni indicate nella tabella, $A = S + 64$, che nel nostro caso fornisce 66, rappresentazione ASCII di 'B'.

Per il 64 le cose si svolgono in maniera assai simile con la differenza che, avendo tale macchina la possibilità di imprimere sullo schermo 25 righe ciascuna di 40 caratteri, avrà un'area rappresentativa in memoria formata da $25 \times 40 = 1000$ locazioni (figura 4). Tale area parte da 1024 e giunge fino alla locazione 2023. Anche con il 64 potete ripetere il semplice esperimento proposto cambiando, naturalmente, 7680 in 1024. Ne proponiamo comunque per essa un altro che consiste nel cercare una

parola sullo schermo ed inviare un messaggio quando viene trovata.

Supponiamo di voler cercare la parola 'END'. Le tre lettere che la compongono vengono codificate (in Codice di Schermo) dai tre numeri 5, 14 e 4. Allora, quando sullo schermo scriveremo la parola END, in tre locazioni adiacenti della Memoria di Schermo a partire da un certo punto, dipendente dalla prima posizione impegnata sullo schermo per la scrittura, verranno posti rispettivamente i numeri 5, 14 e 4. Per raggiungere il nostro scopo, dovremo allora programmare il computer in modo che esso analizzi sequenzialmente tutte le locazioni del video a partire dalla 1024 ed invii

il messaggio quando trova la sequenza esatta. Il programma che svolge questa funzione è riportato nel Listato 1; esso ci indica inoltre, una volta trovata la parola cercata, a partire da quale punto nella Memoria di Schermo essa è stata impressa. Per una verifica, battete in macchina il programma in questione e dopo scrivete in una qualsiasi posizione sullo schermo la parola 'END'. Dato il Run, vedrete, dopo qualche secondo, comparire l'indicazione che la macchina ha identificato la parola e la locazione da cui essa parte. Naturalmente un programma del genere funziona anche sul Vic dopo l'opportuna modifica del ciclo FOR ... NEXT.

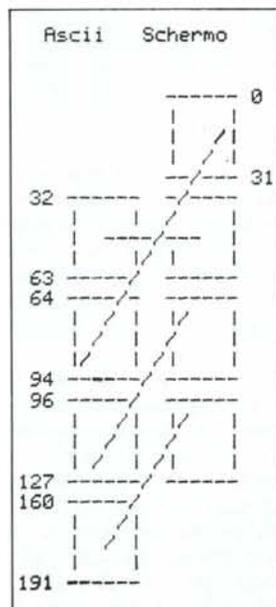


Figura 3
Corrispondenza tra il codice ASCII e quello di schermo.

Scriviamo sullo schermo

I ragionamenti fatti possono essere invertiti: ovvero, così come noi possiamo leggere nella zona video il codice di un carattere posto in una certa posizione dello schermo, se noi memorizziamo in una determinata locazione della Ram video il Codice di Schermo relativo ad un carattere, il carattere rappresentato comparirà nel punto corrispondente della griglia di schermo. In altre parole, se proviamo a scrivere nella prima locazione della Memoria di Schermo il numero decimale 2 con POKE 1024,2 o con POKE 7680,2 a seconda che si tratti del C 64 o del Vic, nella prima colonna della prima linea dello schermo dovrebbe comparire il carattere 'B'. Provate a farlo e non vedrete niente! Non vi stiamo prendendo in giro, solo che tale operazione non è sufficiente per la visualizzazione della lettera. In effetti il carattere è presente sullo schermo e per verificarlo portate il cursore nella prima posi-

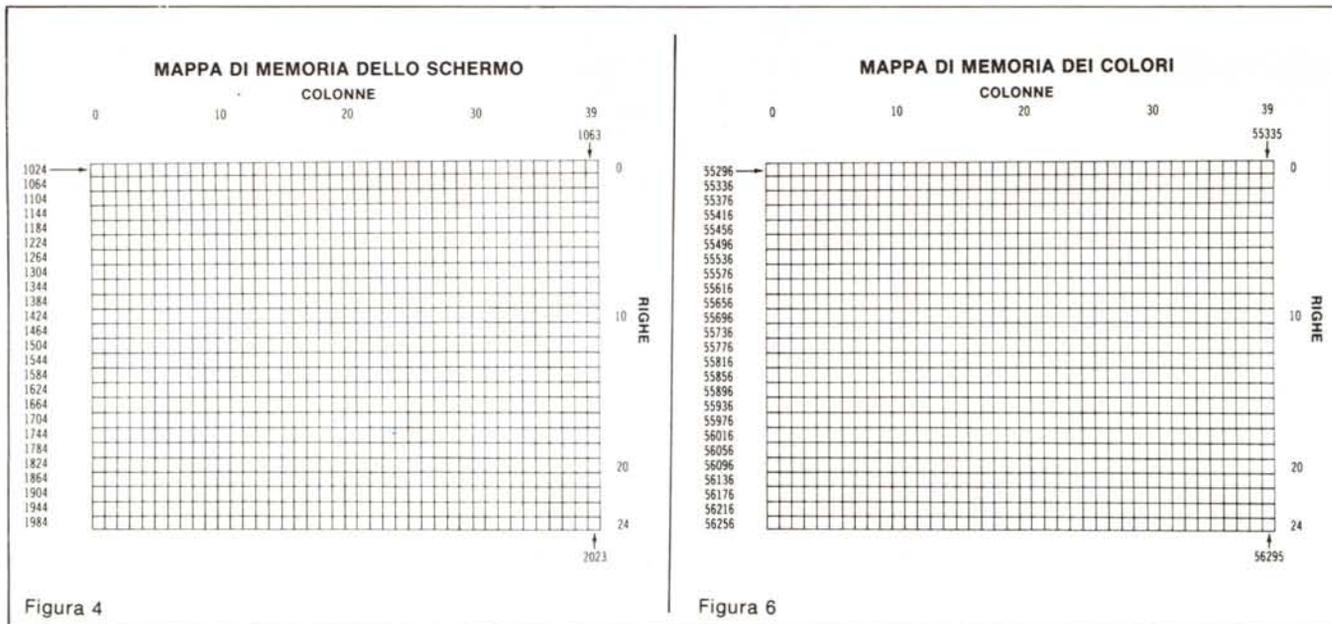


Figura 4 - Mappa della memoria di schermo del Commodore 64.

Figura 6 - Mappa della Memoria del Colore del Commodore 64.

zione premendo il tasto <Home>: vedrete il carattere comparire sotto il cursore, durante il lampeggio, ogni volta che questo è acceso e non lo vedrete più negli istanti in cui il cursore è spento, cioè quando non è presente sullo schermo. Dobbiamo infatti definire un altro parametro fondamentale e cioè il colore con cui vogliamo che venga visualizzato il carattere. Alla Ram video corrisponde infatti una seconda 'griglia', quella di colore, composta dallo stesso numero di locazioni accessibili sullo schermo. Tale griglia, che chiameremo Memoria del Colore, inizia nel Vic base a partire da 38400 mentre nel C 64 è posta a partire dalla locazione 55296 (figure 5 e 6).

Allora, ragionando sul 64 per fissare le idee, volendo scrivere una 'B' nella prima locazione del video, cioè quella in alto a sinistra sullo schermo del televisore, dovremo per prima cosa memorizzare nella Ram di Schermo il codice 2 relativo alla lettera 'B' e poi nella corrispondente locazione della Memoria del Colore il numero di codice del colore (da 0 a 15) con cui vogliamo che venga impresso il carattere (figura 7). Dei colori ottenibili, i primi 8, validi sia per il Vic che per il C 64, sono riportati nella tabella seguente:

Codice	Colore
0	Nero
1	Bianco
2	Rosso
3	Celeste
4	Viola
5	Verde
6	Blu
7	Giallo

Il tutto si risolve quindi con l'istruzione:
POKE 1024,2: POKE 55296,0
che scriverà il carattere nella corretta posizione in colore nero.

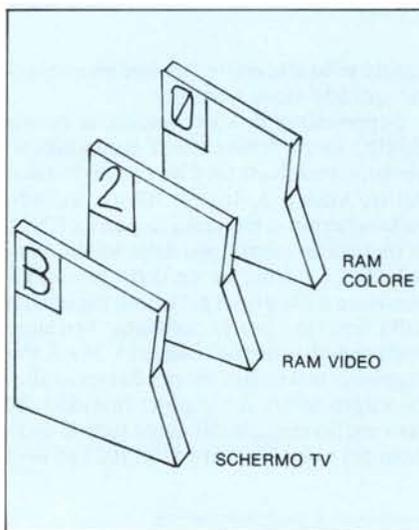


Figura 7 - Per imprimere una B in colore nero nella prima locazione di schermo, bisogna depositare uno '0' (codice del colore nero) nella prima locazione della Ram del colore e un 2 (codice della B) nella prima locazione della Ram di Schermo.

È bene ricordare che esistono delle macchine di nuova serie che all'accensione permettono di scrivere sullo schermo con delle Poke senza che venga definito il colore. Naturalmente il discorso fatto vale in generale se si ha la necessità di visualizzare con un colore diverso da quello presente all'accensione.

A volte si ha la necessità, durante lo svolgimento di un programma, di individuare una ben precisa locazione nella Memoria video, per effettuare una lettura o una scrittura, partendo dalle coordinate, riga e colonna, di una certa locazione sullo schermo. In altre parole può capitare ad esempio di dover imprimere un carattere in una certa posizione e di volerlo fare dando alla macchina, in un programma, solo la

riga e la colonna in cui vogliamo avvenga l'impressione. Per facilitare questo compito, di seguito diamo delle semplici relazioni che permettono di ottenere l'effetto descritto.

Se stiamo lavorando su di un Vic, possiamo aggiungere al programma queste linee:

$$\begin{aligned} \text{VIDEO} &= 7680 + X + 22 \cdot Y \\ \text{COLORE} &= 38400 + X + 22 \cdot Y \end{aligned}$$

dove X rappresenta la colonna in cui vogliamo scrivere o leggere e Y la riga. Verifichiamo per sicurezza l'ultima relazione data. Per accedere al punto di coordinate 10,10 nella mappa video scriveremo:

$$\text{VIDEO} = 7680 + 10 + 22 \cdot 10 = 7910$$

vogliamo cioè impegnare la casella a partire dalla 7680 che è la prima. Segnate sulla griglia data nella figura il punto 10,10 (n.b. il conteggio inizia da 0) e contando partendo dalla prima locazione, in alto a sinistra vi accorgete che si andrà a cadere proprio nella casella 7910.

Nello stesso modo, la formula da usare per il 64 è:

$$\begin{aligned} \text{VIDEO} &= 1024 + X + 40 \cdot Y \\ \text{COLORE} &= 55296 + X + 40 \cdot Y \end{aligned}$$

Come ulteriori esempi provate i programmi riportati nei listati 2, 3 e 4 che tracciano il grafico della retta $Y=X$ con risoluzione 40×25 utilizzando i metodi esposti. Nel programma 2, il tracciamento inizierà dal punto più in alto all'estrema sinistra dello schermo. Nel segmento 3 l'origine degli assi viene invece traslata in basso a sinistra mentre con il programma del listato 4 vengono tracciati gli assi e quotati, sia sulle ordinate che sulle ascisse, ottenendo un quadrante di 35×22 punti. Come si può notare dall'osservazione delle linee 190 e 200, l'origine del quadrante viene spostata in basso di 22 locazioni e a destra di altre 5

```

10 FORI=1024TO2023
20 IFPEEK(I)=5ANDPEEK(I+1)=14ANDPEEK(I+2)=4THEN50
30 NEXT
50 PRINT"HO TROVATO END A PARTIRE"
60 PRINT"DALLA LOCAZIONE";I:END
    
```

Listato 1 - Questo segmento cerca la parola "end" sullo schermo del 64 ed invia un messaggio quando la trova. Per il Vic in versione base gli estremi del ciclo For ... Next sono 7680 e 8185.

```

100 PRINT"X"
110 POKE53280,0:POKE53281,0:PRINT" ";
130 REM FORI=0TO38:PRINT"-";:NEXT:PRINT" "
140 PRINT" ":FORI=0TO21:PRINT"  I":NEXT
145 PRINT" ":FORI=21TO0STEP-1:PRINTI:NEXT
147 FORI=0TO38:PRINT"-";:NEXT
150 PRINT"      0      5      10      15";
160 PRINT"    20     25     30";
170 FORX=0TO24
180 Y=X
190 POKE1024+X+5+40*(22-Y),46
200 POKE55296+X+5+40*(22-Y),1
210 NEXT
220 PRINT" "
230 GOTO230
    
```

Listato 4 - Questo programma traccia la retta Y = X con origine in basso a sinistra e gli assi cartesiani quotati.

affinché il grafico parta dallo 0 degli assi tracciati.

Naturalmente non pensate di poter disegnare chissà quali grafici perché la risoluzione disponibile è molto bassa; inoltre ricordate che gli incrementi di X e Y devono essere degli interi. Comunque pensiamo che i programmi dati possano essere considerati un utile esempio che aiuta a comprendere il modo in cui accedere direttamente allo schermo con delle Poke.

Conserviamo una schermata

Un'altra cosa che ci chiediamo è se sia possibile memorizzare ciò che appare sullo schermo e richiamarlo quando ne abbiamo bisogno. Naturalmente ciò è possibile grazie al fatto che un'immagine codificata di ciò che noi vediamo è presente in quella zona della memoria che abbiamo chiamato 'Memoria di Schermo' o 'Memoria video'. Basterà allora far sì, con un piccolo artificio, che con un semplice comando "Save" venga conservato, su nastro o disco, il contenuto della suddetta zona per poi poterlo richiamare con un "Load" in qualsiasi momento.

Se ben ricordate (ne abbiamo già parlato) esistono sia sul Vic che sul C 64 delle locazioni (puntatori) il cui contenuto punta all'inizio e alla fine di una certa zona di memoria, tipicamente inizio e fine programma Basic, e quando diamo il comando "Save" viene automaticamente conservato il contenuto della zona di memoria compresa tra questi due puntatori. In altre parole, se un puntatore indica 2048 e l'altro 3048, un "Save" conserverà i 1000 byte tra questi due numeri.

Il puntatore d'inizio è contenuto nelle locazioni 43 e 44 e quello di fine nelle 45 e 46. Il punto della memoria da essi individuato è facilmente calcolabile con PEEK(43) + PEEK(44)*256 (puntatore d'inizio) e con PEEK(45) + PEEK(46)*256 (puntatore di fine). Dato che conosciamo esattamente il punto d'inizio e quello di fine della 'Memoria di Schermo', sposteremo i puntatori in questione all'inizio ed alla fine di tale zona prima di dare il comando "Save" per salvare tutta la Ram video.

Come prima cosa dobbiamo calcolare i valori da memorizzare nelle locazioni 43-

44 e 45-46 come indicato dalla tabella seguente:

P	Vic base	C 64
	A = 7680 B = 8186	A = 1024 B = 2023
44	P1 = INT(A)/256	
43	P2 = A-INT(A)/256	
46	P3 = INT(B)/256	
45	P4 = B-INT(B)/256	

Fatto questo, digiteremo in maniera diretta:

```

POKE 44,P1: POKE 43,P2
POKE 46,P3: POKE 45,P4
    
```

dopo di che se daremo il comando:

```

SAVE "nome" (cassetta) o SAVE "nome", 8 (disco)
    
```

la schermata sarà memorizzata.

Per ricaricarla di nuovo sullo schermo, basterà dare invece il comando

```

LOAD"nome", 1,1 (cassetta) o LOAD"nome", 8,1 (disco).
    
```

Ricordiamo che il secondo 1 nel comando di caricamento permette di posizionare il contenuto del supporto magnetico esattamente nel punto in cui esso è stato prelevato durante il salvataggio.

Tanto per fare un esempio pratico, memorizziamo una certa porzione di schermo di uno dei due computer di cui stiamo parlando, ad esempio di un 64. Supponiamo

```

10 FORX=0TO24
20 Y=X
30 POKE1024+X+40*Y,81
40 POKE55296+X+40*Y,1
50 NEXT
    
```

Listato 2 - Questo programma traccia la retta Y = X con origine in alto a sinistra.

```

10 FORX=0TO24
20 Y=X
30 POKE1024+X+40*(24-Y),81
40 POKE55296+X+40*(24-Y),1
50 NEXT
    
```

Listato 3 - Questo programma traccia ancora la retta Y = X ma con origine in basso a sinistra.

di voler conservare su disco le prime 512 locazioni dello schermo su cui abbiamo impresso delle parole o delle scritte particolari formate utilizzando i caratteri grafici Commodore.

Ricaviamo per prima cosa il valore dei puntatori alla porzione desiderata compresa tra 1024 e 1536 (512 byte):

```

1024/256 = 4      resto = 0
1536/256 = 6      resto = 0
    
```

I valori da posizionare nelle locazioni 43-44 e 45-46 sono quindi:

Loc	Val
43	0
44	4
45	0
46	6

e potremo farlo con:

```

POKE 43,0: POKE 44,4
    
```

e

```

POKE 45,0: POKE 46,6
    
```

Fatto ciò non dobbiamo fare altro che dare il comando SAVE "nome" ecc. per salvare il pezzo di Memoria video su supporto magnetico.

Come già accennato, se vogliamo richiamare la schermata conservata, possiamo farlo con

```

Load "nome", 1,1 (cassetta)
o con
    
```

```

Load "nome", 8,1 (disco)
    
```

dopo aver naturalmente riempito la zona Ram di Colore corrispondente con il codice del colore con cui vogliamo che i vari caratteri siano impressi sullo schermo. Ricordate come ultima cosa che il caricare con un 1 come secondo numero in un comando di Load modifica il valore dei puntatori di inizio e fine memoria portandoli allo stesso valore a cui erano stati posti per il salvataggio. Per riassettarli al valore iniziale basta effettuare:

```

POKE 43,1:POKE 44,8: NEW
per il 64 oppure
    
```

```

POKE 43,1:POKE 44,16:NEW
per il Vic in configurazione base.
    
```

Per oggi ci fermiamo qui ma ci proponiamo di riprendere l'argomento per illustrare altre cose interessanti che riguardano lo schermo.