



# Video Display Controller 8563

Seconda parte

■ Questo mese vedremo una dozzina di registri interni all'8563 che ci permetteranno di effettuare lo scroll fine orizzontale e verticale della pagina video, di cambiare il colore di fondo di questa e come sia possibile rilocare la mappa video e/o quella colore in un altro punto della video-ram nascosta dentro a tale integrato video. Infine un paio di programmi di linguaggio macchina che, «infiltrati» nelle routine di interrupt (mascherabile e non) del 128, permettono di cambiare colore di schermo o l'intero contenuto di questo con la pressione di un solo tasto. ■

## 12 registri

La tabella 1, che a dire il vero sarebbe dovuta apparire sul numero scorso, mostra i registri già visti nella puntata precedente. In tabella 2, questa volta puntualmente, sono mostrati i 12 registri che andiamo a descrivere questo mese.

I registri 2 e 3 servono per centrare lo schermo in senso orizzontale: il primo spostandolo ad intervalli di un carattere (ovvero otto pixel) il secondo mezzo carattere alla volta (quattro pixel). Per effettuare tali «tarature» ci avvarremo del programma di figura 1, col quale potremo leggere o scrivere in qualunque dei registri dell'8563.

Il suo funzionamento è assai semplice: dapprima si indica a quale registro siamo interessati, successivamente alla domanda «value» (valore) se indichiamo un numero si effettuerà un'operazione di scrittura se invece battiamo [return] avremo una lettura e il valore del registro sarà «stampato» sullo schermo.

Tornando ai nostri registri 2 e 3, ba-

sterà variare i loro contenuti per vedere tutto lo schermo spostarsi a destra o a sinistra di un carattere o di mezzo, a seconda del registro interessato. Per l'esattezza, immetteremo un valore compreso tra 0 e 120 nel registro 2 (di default è 102) o un valore compreso tra 66 e 79 nel registro 3 (default 73). Si badi bene che molti dei registri dell'8563, se manipolati a casaccio possono provocare la scomparsa del video a 80 colonne sino al ripristino degli stessi o, meglio, al prossimo reset del sistema. A tal proposito, se qualche lettore intende esplorare l'8563 con la routine di figura 1, prima di modificare il contenuto di ogni registro sarebbe opportuno che prendesse nota del suo contenuto iniziale, in modo da poterlo ripristinare anche alla cieca, in caso di perdita del video.

Degli otto bit di cui è formato il registro 5, i tre meno significativi servono per centrare lo schermo in senso verticale, un pixel alla volta. Avendo a disposizione soli tre bit, potremo scegliere tra 8 posizioni diverse, pari dunque all'altezza di un carattere. I valori che possono essere immessi in tale re-

gistro sono compresi (da una analisi abbastanza sommaria... chi sa di più si faccia vivo) tra 224 e 231, e di default è presente il valore 224.

## Scroll fine e Reverse Video

Più che un vero e proprio scroll fine, come già successo su altri Commodore, il 128 dispone solo di due registri per far scrollare di pochi pixel la pagina video nelle quattro direzioni, ferma restando una cornice dello stesso colore del fondo tipo il bordo delle 40 colonne. Ciò significa che per far scrollare più righe di testo occorre usare una routine in linguaggio macchina che terminato lo scroll, ad esempio, di 8 pixel in su, velocemente resettata il registro, sposta tutto il testo una riga più in alto e riempie la nuova riga formata in basso col nuovo testo da far apparire lentamente.

Il difficile sta nel fatto che la routine in linguaggio macchina deve essere o così veloce da non farcene accorgere (di solito si ha uno sfarfallio di scher-

mo) oppure sincronizzata col quadro in modo da effettuare tutte le operazioni quando il «pennello» elettronico è fuori campo (sta ad esempio costruendo il bordo superiore o inferiore). Considerato poi che una routine in linguaggio macchina, nel 128, per veloce che sia dovrà in ogni caso combattere coll'indirizzamento indiretto dei registri che indirizzano indirettamente la memoria (vedi puntata precedente) non resta che concludere che lo scroll fine, inteso come lo era nel 64, non è (praticamente) possibile. A meno che qualcuno non ci riesca e ce lo segnali. Grazie.

Ad ogni modo, i registri interessati allo scroll sono per il verso verticale il 24 e per quello orizzontale il 25. In figura 2 è mostrato un semplice programmino che permette di muovere lo schermo video sotto la cornice esterna, nelle quattro direzioni, usando i quattro tasti cursore.

Colpo di scena: il bit 6 del registro 24 permette di ottenere lo schermo in reverse: caratteri colore di fondo su fondo colore dei caratteri. Settandolo si ha l'effetto di veder apparire anche per le 80 colonne un bordo attorno allo schermo: il reverse video agisce solo sulle posizioni dei caratteri.

## Pagina video e Pagina colore

La mappa di memoria dell'8563 descritta il mese scorso e mostrata in figura 3, può essere in parte riconfigurata. Possiamo ad esempio spostare la pagina video da tutt'altra parte così come per la pagina colore, meglio detta degli attributi. I registri interessati sono: 12, 13, 20 e 21. I primi due (12 parte alta e 13 parte bassa) permettono di spostare la pagina video, gli altri due, con lo stesso formato, la pagina degli attributi. Possiamo ad esempio spostare una di queste due mappe (o ambedue) nella zona libera della memoria video, a partire cioè dall'indirizzo

4096. Se ad esempio vogliamo spostare la mappa video a tale indirizzo e la mappa degli attributi a 6144 (4096+2048), basterà immettere nel registro 12 il valore 16 (4096/256) e nel registro 20 il valore 24 (6144/256) le rispettive «parti-basse» resteranno a zero.

Oltre a questo, occorre informare il sistema operativo di tale cambiamento, in modo da ottenere letture e scritture effettivamente nello schermo desiderato. Per fare questo sarà sufficiente ricopiare il valore «pok-ato» nel registro 12, nella locazione Ram 2606 del 128, il valore «pok-ato» nel registro 20 nella locazione 2607. Più avanti mostreremo una applicazione di tale possibilità, comandata dal tasto restore.

## Colore di schermo

Il registro 26 del Video Display Controller serve per cambiare colore. I 4 bit meno significativi sono riferiti al colore di fondo, i 4 più significativi il colore della «penna» in alta risoluzione. 4 bit equivalgono a 16 possibili combinazioni diverse, il numero dei colori di cui il 128 dispone.

Per colorare diversamente lo schermo, come per le celle della mappa colore vista lo scorso mese, seguendo le specifiche RGBI, indicheremo per il colore che vogliamo usare di quali componenti cromatiche esso è composto e se desideriamo o meno l'intensità doppia. Un colore è dunque codificato con la sequenza di bit:

Bit 0: intensità  
Bit 1: blu  
Bit 2: verde  
Bit 3: rosso

Se ad esempio desideriamo un colore di schermo blu chiaro, setteremo il bit 0 e il bit 1 del registro colore (blu chiaro = blu + intensità doppia). Il bianco è composto da tutte e tre le componenti cromatiche più l'intensità

doppia (quindi setteremo tutt'e quattro i bit), mentre per le altre combinazioni di colori abbiamo che:

Blu + Verde = Cyan  
Verde + Rosso = Giallo  
Rosso + Blu = Magenta

Cyan, per la cronaca è un quasi-azzurro mentre magenta assomiglia molto a un rosa intenso.

Se settiamo solo il bit intensità, il «nero-chiaro» che dovremmo ottenere è un grigio scuro, mentre settando solo le componenti cromatiche e non l'intensità, il «bianco-scuro» così ottenuto è il grigio chiaro.

## Posizione cursore

Terminiamo la nostra carrellata sui registri dell'8563 (l'alta risoluzione sul prossimo numero, promesso) con i registri 14 e 15 nei quali possiamo leggere la posizione del cursore. Tanto per cambiare la parte alta nel primo dei due, la parte bassa nel secondo. Si noti che per posizione cursore non si intende riga-colonna, ma l'effettiva locazione di video ram dove si trova il carattere sotto il cursore che vediamo sullo schermo. Tant'è che se spostiamo la mappa video col procedimento visto prima, anche i registri 14 e 15 si aggiorneranno automaticamente al nuovo range di valori.

Figura 1

```
3 FAST
5 REM *****
6 REM * LETTURA - SCRITTURA NEI *
8 REM * REGISTRI DELL'8563 *
9 REM *****
10 INPUT"REGITER ":R:V=-1
20 INPUT"VALUE ":V:IFV=-1THEN60
30 POKEDEC("D600"),R
40 POKEDEC("D601"),V
50 GOTO10
60 POKEDEC("D600"),R
70 PRINTPEEK(DEC("D601"))
80 GOTO10
```

Tabella 1

Registro	Funzione svolta
0	Read/Write altri registri
1	Caratteri per linea
4	Sincronismo verticale
6	Linee per pagina
7	Shift verticale 8 pixel
9	Matrice caratteri (vert.)
11	Linee cursore
18	Indirizzo Video Ram Hi
19	Indirizzo Video Ram Lo
31	Registro canale
34	Inizio rappresentazione
35	Fine rappresentazione

Tabella 2

Registro	Funzione svolta
2	Shift orizzontale 8 pixel
3	Shift orizzontale 4 pixel
5	Shift verticale 1 pixel
12	Inizio pagina video Hi
13	Inizio pagina video Lo
14	Posizione cursore Hi
15	Posizione cursore Lo
20	Inizio pagina attributi Hi
21	Inizio pagina attributi Lo
24	Scroll fine verticale (bit 6 = reverse screen)
25	Scroll fine orizzontale
26	Registro colore

Figura 2

```

10 REM .....
20 REM *
30 REM *      S C R O L L   F I N E
40 REM *
50 REM *      U S A N D O
60 REM *
70 REM *   I 4 TASTI CURSORE
80 REM *
90 REM .....
100 POKEDEC("D600"),.6:POKEDEC("D601"),.24
110 X=7:FAST
120 GETAS:IFAS=""THEN120
130 A=ASC(AS+CHRS(0))
140 IF (A AND 127)<>17 AND (A AND 127)<>29 THEN 120
150 IF A=145 AND Y<7 THEN Y=Y+1
160 IF A=29 AND X<7 THEN X=X+1
170 IF A=17 AND Y>0 THEN Y=Y-1
180 IF A=157 AND X>0 THEN X=X-1
190 V=64+X
200 POKEDEC("D600"),.25
210 POKEDEC("D601"),.V
220 V=32+Y
230 POKEDEC("D600"),.24
240 POKEDEC("D601"),.V
250 GOTO120
    
```

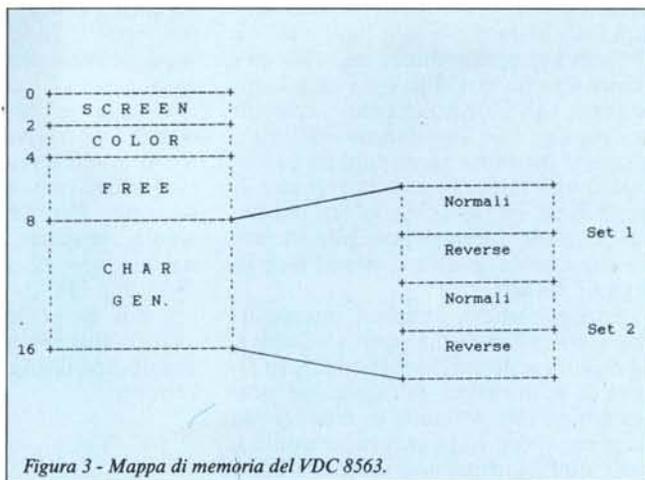


Figura 3 - Mappa di memoria del VDC 8563.

## Figura 5 e seguenti

Il listato di figura 5 implementa un rapido metodo per cambiare colore di schermo: è sufficiente infatti la pressione del tasto ALT per vedere cambiare ciclicamente tutti i colori possibili (16). Oltre a questo, con la sequenza CONTROL SHIFT è possibile effettuare il reverse di schermo che, co-

me detto, avviene settando o resettando il bit 6 del registro 24. Come visibile in figura 5, il listato è composto da alcune istruzioni e da un po' di linee data contenenti la routine in linguaggio macchina disassemblata in figura 6. Apriamo a questo punto una piccola parentesi per tutti quelli che non hanno già militato precedentemente nella fila dei Vic-hinghi o dei post-vichinghi (classe millenovecento...64).

Il centoventotto, come i suoi nobili predecessori, relega la gestione della tastiera ad una apposita routine che parallelamente a quanto avviene nel computer (es. l'esecuzione di un programma) provvede a scandire i tasti per vedere quale di questi è stato premuto. Parallelamente nel senso simulato del termine, per la precisione col meccanismo delle interruzioni: qualsiasi cosa stia facendo il processore, ogni 60-esimo di secondo, viene man-

dato a questo un interrupt mascherabile. A questo punto, se il processore non ha settato la mascherazione delle interruzioni, molla tutto ed esegue la scansione della tastiera.

Detto ciò, per implementare qualcosa che alla pressione di un determinato tasto deve essere eseguita, l'unico modo è di «incunearsi» nella gestione della tastiera, cambiando il puntatore alla routine che la gestisce. Ovvero, scriveremo da qualche parte in memoria il nostro bravo programmino in linguaggio macchina, il puntatore alla gestione della tastiera viene fatto puntare alla nostra routine, la quale a sua volta terminerà con un salto alla normale routine di manipolazione delle interruzioni che, come detto, controlla quale tasto è stato premuto.

La nostra routine per vedere se il tasto interessato è stato premuto farà sicuramente riferimento ad alcune celle

- BIT 0 Luminosità
- BIT 1 Blu
- BIT 2 Verde
- BIT 3 Rosso
- BIT 4 Flash
- BIT 5 Sottolineato
- BIT 6 Reverse
- BIT 7 Secondo set

Figura 4 - Contenuto di una cella della pagina colore del VDC.

Figura 5

```

10 REM .....
20 REM *
30 REM *   C A M B I A   I L   C O L O R E
40 REM *
50 REM *   D I   S C H E R M O   A D   O G N I
60 REM *
70 REM *   P R E S S I O N E   D I   A L T
80 REM *
81 REM *   E   A B I L I T A   I L
82 REM *
83 REM *   R E V E R S E   D I   S C H E R M O
84 REM *
85 REM *   C O N   C O N T R O L - S H I F T
86 REM *
90 REM .....
100 FAST
110 FOR I=4864 TO 4962
120 READ II
130 POKE I,II
140 NEXT
150 SYS4864
1000 DATA120.169.19.141.21.3.169.17.141.20.3.169.0.133.254
1010 DATA88.96.72.138.72.152.72.165.211.197.250.240.30.133
1020 DATA250.201.8.208.32.166.254.169.26.141.0.214.44.0.214
1030 DATA16.248.142.1.214.232.201.16.208.2.166.0.134.254
1040 DATA104.168.104.170.104.76.101.250.201.5.208.244.162
1050 DATA24.142.0.214.44.0.214.16.254.173.1.214.73.64.142
1060 DATA0.214.44.0.214.16.251.141.1.214.76.58.19
    
```

Figura 7

```

100 FORI=5120TO5184
110 READI II
120 POKE I,II
130 NEXT
140 POKE792,59
150 POKE793,20
160 END
210 REM .....
220 REM *
230 REM *   D O P P I O   S C H E R M O
240 REM *
250 REM *
260 REM *   (C) 1986 ADP-SOFTWARE
270 REM *
280 REM *
290 REM .....
1000 DATA162.12.142.0.214.44.0.214.16.251.173.1.214.73.16
1010 DATA142.0.214.44.0.214.16.251.141.1.214.141.46.10.24
1020 DATA105.8.141.47.10.162.20.142.0.214.44.0.214.16.251
1030 DATA141.1.214.169.17.32.210.255.169.145.32.210.255.96
1040 DATA32.0.20.76.64.250
    
```

di memoria come avviene nel programma di figura 6 che andiamo a commentare.

Le prime 9 istruzioni («linee» 1300-1310) servono per inizializzare il programma: viene cambiato l'indirizzo della routine di manipolazione delle interruzioni (agendo sulle celle \$314 e \$315) e posto uguale a zero la cella \$FE che ci serve per mantenere il colore di schermo attuale. Come si vede, l'inizio della manipolazione delle interruzioni è stata portata all'indirizzo 1311, dove effettivamente ci occuperemo di cambiare colore ad ogni ALT e di invertire il video se si digita CONTROL SHIFT. La routine di \$1311 per prima cosa salva i registri A, X, Y nello stack dopodiché accede alla cella di

Figura 6

```

MONITOR
PC SR AC XR YR SP
;
B000 00 00 00 00 F8

01300 78 SEI
01301 A9 13 LDA #S13
01303 8D 15 03 STA $0315
01306 A9 11 LDA #S11
01308 8D 14 03 STA $0314
0130B A9 00 LDA #S00
0130D 85 FE STA $FE
0130F 58 CLI
01310 60 RTS
01311 48 PHA
01312 8A TXA
01313 48 PHA
01314 98 TYA
01315 48 PHA
01316 A5 D3 LDA $D3
01318 C5 FA CMP $FA
0131A F0 1E BEQ $133A
0131C 85 FA STA $FA
0131E C9 08 CMP #S08
01320 D0 20 BNE $1342
01322 A6 FE LDX $FE
01324 A9 1A LDA #S1A
01326 8D 00 D6 STA $D600
01329 2C 00 D6 BIT $D600
0132C 10 F8 BPL $1326
0132E 8E 01 D6 STX $D601
01331 E8 INX
01332 C9 10 CMP #S10
01334 D0 02 BNE $1338
01336 A6 00 LDX $00
01338 86 FE STX $FE
0133A 68 PLA
0133B A8 TAY
0133C 68 PLA
0133D AA TAX
0133E 68 PLA
0133F 4C 65 FA JMP $FA65
01342 C9 05 CMP #S05
01344 D0 F4 BNE $133A
01346 A2 18 LDX #S18
01348 8E 00 D6 STX $D600
0134B 2C 00 D6 BIT $D600
0134E 10 FB BPL $134B
01350 AD 01 D6 LDA $D601
01353 49 40 EOR #S40
01355 8E 00 D6 STX $D600
01358 2C 00 D6 BIT $D600
0135B 10 FB BPL $1358
0135D 8D 01 D6 STA $D601
01360 4C 3A 13 JMP $133A

```

memoria \$D3 che al termine di ogni scansione indica quale o quali tasti-prefisso sono stati premuti. La corrispondenza bit-tasto è mostrata qui di seguito:

```

bit 0 SHIFT
bit 1 COMMODORE
bit 2 CONTROL
bit 3 ALT
bit 4 CAPS LOCK

```

Tornando al disassemblato, prima di procedere si controlla se tale cella è uguale al contenuto di \$FA in modo da non avere repeat su questi tasti che ci creerebbe non pochi problemi per azzeccare il colore giusto. Dopo di ciò si controlla se il valore di \$D3 è uguale a 8 (è stato premuto l'ALT) o uguale a 5 (sono stati premuti CONTROL e SHIFT). A seconda di questi due casi, come prevedibile, si provvede a mandare il contenuto di \$FE al registro 26 dell'8563 e a incrementare \$FE oppure si complementa il bit 6 del registro 24, che fa passare da schermo normale a reverse e viceversa. Al termine, «linee» 133A-133F, si ripristinano i registri Y, X, A e si salta a \$FA65 per effettuare la normale scansione della tastiera.

In figura 7 è listato il programma che implementa due schermi a 80 colonne, switch-abili tramite il tasto Restore. Attenzione: il tasto restore non è un tasto normale. Esso infatti non è, come gli altri, scandito dal processore, ma è collegato a questo direttamente (a meno di qualche integratuccio di comodo) tramite il suo piedino NMI. L'acronimo sta per interrupt non mascherabile e ciò sta a significare che il processore non può fare a meno di ascoltarlo. Come per gli interrupt normali, anche quello mascherabile ha la sua routine di gestione in memoria, la quale non fa altro che controllare se è stato premuto anche il run/stop nel qualcaso viene eseguita la ben nota routine di re-inizializzazione del sistema. Analogamente al caso precedente, per implementare l'algoritmo di scambio schermo ci «incuneeremo» nella routine di NMI, cambiando il puntatore a questa per farlo puntare al nostro programmino in linguaggio macchina e facendo terminare quest'ultimo con salto alla normale manipolazione dell'NMI.

In figura 8 è disassemblato il programma in linguaggio macchina presente in figura 7 sotto forma di linee data.

Tale programma è dunque attivato ad ogni pressione del tasto restore o, alternativamente, da programma tramite una SYS 5120. Le linee 140 e 150 di figura 7 provvedono a spostare il

Figura 8

```

MONITOR
PC SR AC XR YR SP
;
B000 00 00 00 00 F8

01400 A2 0C LDX #S0C
01402 8E 00 D6 STX $D600
01405 2C 00 D6 BIT $D600
01408 10 FB BPL $1405
0140A AD 01 D6 LDA $D601
0140D 49 10 EOR #S10
0140F 8E 00 D6 STX $D600
01412 2C 00 D6 BIT $D600
01415 10 FB BPL $1412
01417 8D 01 D6 STA $D601
0141A 8D 2E 0A STA $0A2E
0141D 18 CLC
0141E 69 08 ADC #S08
01420 8D 2F 0A STA $0A2F
01423 A2 14 LDX #S14
01425 8E 00 D6 STX $D600
01428 2C 00 D6 BIT $D600
0142B 10 FB BPL $1428
0142D 8D 01 D6 STA $D601
01430 A9 11 LDA #S11
01432 20 D2 FF JSR $FFD2
01435 A9 91 LDA #S91
01437 20 D2 FF JSR $FFD2
0143A 60 RTS
0143B 20 00 14 JSR $1400
0143E 4C 40 FA JMP $FA40

```

puntatore alla routine di NMI. Quello che fa il programma di figura 8 è assai semplice, a seconda di quale schermo è attivo, sposta i puntatori 12, 13 20 e 21 dell'8563 e cambia il contenuto delle celle 2606, 2607 della ram in modo da passare allo schermo, in quel momento, nascosto. Infine un Cursor-Down e un Cursor-Up sono «sparati» sul video in modo da avere il cursore sempre sullo schermo selezionato. Fine.

## Inviare i vostri contributi

Chiunque voglia contribuire alla rubrica «128 da zero» proponendo argomentazioni, trucchetti o semplicemente notizie riguardanti il Commodore 128 non deve far altro che inviare il proprio contributo in redazione, possibilmente indicando sulla busta «128 da zero».

I lavori più interessanti saranno ospitati su queste pagine ovviamente ricompensando gli autori a seconda sia delle dimensioni che dell'importanza degli elaborati stessi. Non dimenticate (a tal proposito) di indicare sempre chiaro il vostro nome, indirizzo, codice fiscale. Anche il vostro numero di telefono potrebbe esserci utile per potervi eventualmente contattare prima della pubblicazione per chiarimenti. Buon lavoro.