

I «device driver»

settima parte

Terminiamo in questa puntata l'analisi del device driver chiamato ANSI.SYS, analisi che avevamo interrotto prima di parlare della routine che gestisce l'OUTPUT verso lo schermo video, routine che ci interessa particolarmente e che è in un certo senso la principale in quanto ANSI.SYS gestisce proprio l'output formattato verso il video. Per comodità di analisi, riportiamo ancora una volta in figura 1 una tabella sinottica indicante tutte le sequenze ANSI gestite dal PC (o meglio dall'MS-DOS) con a fianco il significato della sigla

Le routine di OUTPUT

Parliamo in questo caso di routine «al plurale» in quanto alla stessa routine fanno capo sia la funzione di OUTPUT (pura e semplice) che la funzione OUTPUT WITH VERIFY, segno questo che l'output con verifica viene trattato alla stessa stregua dell'output semplice, come dire che non viene gestito l'output con verifica...

Nella figura 2 vediamo il disassemblato della routine principale che implementa la funzione OUTPUT: dal momento che in CX c'è il numero di caratteri da inviare in output e che in DI c'è l'indirizzo corrente del buffer contenente tali caratteri, si capisce subito che verrà chiamata per CX volte la subroutine 02FBH ogni

volta avendo in AL il carattere da visualizzare oppure (è proprio il compito della funzione OUTPUT) interpretare una sequenza ANSI, che come sappiamo deve iniziare con la coppia di caratteri «ESC» e «[».

Nella figura 3 invece è disassemblata la routine 02F8H: tra un salvataggio nello stack all'inizio ed il corrispondente ripristino dei registri al termine della routine, c'è il corpo vero e proprio della subroutine.

Dal momento che si tratta di output su video, deve innanzitutto essere testato il cosiddetto «modo video» nonché il numero di colonne rappresentabili sullo schermo (è la word posta all'indirizzo 0040H:0049H) e successivamente viene letto il valore della pagina attuale

nome	sequenza	significato
CUP	[#];[#]H	"CURsor Position" : posizionamento del cursore
HVP	[#];[#]f	"Horizontal and Vertical Position" : posizionamento del cursore
CUU	[#]A	"CURsor Up" : muove il cursore in alto
CUD	[#]B	"CURsor Down" : muove il cursore in basso
CUF	[#]C	"CURsor Forward" : muove il cursore a destra
CUB	[#]D	"CURsor Backward" : muove il cursore a sinistra
DSR	6n	"Device Status Report" : segnala la posizione del cursore
SCP	s	"Save Cursor Position" : salva la posizione del cursore
RCP	u	"Restore Cursor Position" : ripristina la posizione del cursore
ED	2J	"Erase display" : cancella lo schermo (equivale al CLS)
EL	K	"Erase Line" : cancella dal cursore alla fine linea
SGR	#[#]m	"Set Graphic Rendition" : setta caratteristiche dell'output
SM	#h	"Set Mode" : setta il modo video
RM	#l	"Reset Mode" : resetta il modo video
KKR	#[#]p	"Keyboard Key Reassignment" : associa la stringa "s" ad un tasto
KKR	"c";"s"p	"Keyboard Key Reassignment" : associa la stringa "s" al tasto "c"

Figura 1 - In questa che è l'ultima puntata relativa al driver ANSI.SYS, riportiamo l'elenco delle sequenze ANSI gestibili da MS-DOS: ripetiamo ancora una volta che indistintamente tutte le sequenze riportate devono essere precedute dalla coppia di caratteri «escape» e «[», che dunque permettono di contraddistinguere le sequenze ANSI dai normali caratteri inviati al video.

video (valore posto all'indirizzo 0040H:0062H) e la posizione corrente del cursore all'interno della pagina stessa (valore posto a partire dall'indirizzo 0040H:0050H, una word per ognuna delle 8 pagine video): tutti questi valori vengono salvati in opportune locazioni di memoria. Dal momento che queste operazioni vengono effettuate per ogni singolo carattere da inviare, si comincia già a capire il perché della lentezza intrinseca dell'ANSI.SYS: tutto sommato questa parte di inizializzazione poteva utilmente essere eseguita «una tantum» all'inizio della routine 02E2H.

Comunque al termine delle inizializzazioni viste, legate dunque al particolare modo video in atto, viene chiamata la subroutine 034CH, che è, scendendo dunque sempre più nei dettagli, il «cuore» della routine OUTPUT: si tratta infatti di una routine particolare che effettua un salto ad una locazione di volta in volta differente, a seconda dello «stato» in cui ci si trova.

Tale subroutine 034CH è la seguente

```
034C BEFD00      MOV SI,00FD
```

```
034F FF24      JMP [SI]
```

ed in essa si vede che l'indirizzo di salto è posto nella locazione di memoria di indirizzo 00FDH (nel Code Segment, in quanto in DS è stato posto il valore contenuto in CS, subito prima della CALL 034CH): a seconda di come si evolve la situazione e cioè a seconda di come precede l'analisi sintattica della (eventuale) sequenza ANSI in tale locazione verrà posto l'indirizzo della routine che dovrà processare il «prossimo carattere atteso».

Dal momento che una sequenza ANSI inizia per «ESC» e «[», innanzitutto verrà testata la presenza di un «ESC» (routine all'indirizzo 03E3H, figura 4) ed in caso positivo la presenza del carattere «[» (routine all'indirizzo 0351H, figura 5).

Ricordiamo che tutte le routine citate vengono attivate una per volta durante l'analisi di ogni singolo carattere e non in sequenza, questo per permettere al DOS di compiere eventualmente altre funzioni tra l'invio di un carattere ed il successivo.

Siamo arrivati dunque ad aver ricono-

sciuto la coppia «ESC» e «[» e perciò arriviamo alla routine posta all'indirizzo 0364H (fig. 6): ora ci possiamo aspettare un gran numero di caratteri differenti a seconda dei quali si passerà a routine differenti. Siamo di fronte ad una struttura ad albero in cui a seconda del carattere viene percorso un ramo particolare per arrivare ad un nuovo nodo dal quale si dipartono più rami, scelti in base al carattere ancora successivo.

Tutto questo perché l'ANSI ha scelto per le sequenze omonime una struttura del tipo:

— caratteri di inizio sequenza («ESC» e «[»);

— caratteri indicanti la posizione sul video oppure altri valori parametrici, i quali caratteri possono mancare del tutto o in parte, ed in genere sono separati da un «;»

— carattere finale che definisce il tipo di sequenza ANSI.

In tal modo si è costretti a leggere «tutta» la sequenza per capire di che tipo si tratta e perciò quali sono le funzioni da svolgere: se si fosse adattata la struttura più razionale da:

```
02E2 E3DC      JCXZ 02C0      ;si torna al DOS
02E4 268A05    MOV AL,ES:[DI]
02E7 47        INC DI
02EB EB0D00    CALL 02FB
02EB E2F7      LOOP 02E4
02ED EBD1      JMP 02C0      ;si torna al DOS
```

Figura 2 - Disassemblato del ciclo principale della routine OUTPUT.

```
03E3 3C1B      CMP AL,1B      ;"ESC"
03E5 7505      JNZ 03EC
03E7 C7045103  MOV WORD PTR [SI],0351
03EB C3        RET
03EC EB2CFD    CALL 011B
03EF C706FD00E303  MOV WORD PTR [00FD],03E3
03F5 C3        RET
```

Figura 4 - Disassemblato della routine posta all'indirizzo 03E3H.

```
0351 3C5B      CMP AL,5B      ;"["
0353 7403      JZ 035B
0355 E9BB00    JMP 03E3
035B C7046403  MOV WORD PTR [SI],0364
035C 33DB      XOR BX,BX
035E 891E0501  MOV [0105],BX
0362 E00B      JMP 036C
```

Figura 5 - Disassemblato della routine posta all'indirizzo 0351H.

```
02FB 50        PUSH AX
02F9 53        PUSH BX
02FA 51        PUSH CX
02FB 52        PUSH DX
02FC 56        PUSH SI
02FD 57        PUSH DI
02FE 06        PUSH ES
02FF 55        PUSH BP
0300 C706170100BB  MOV WORD PTR [0117],BB00 ;CGA
0306 96        XCHG SI,AX
0307 BB4000    MOV AX,0040
030A BEDB    MOV DS,AX
030C A14900    MOV AX,[0049] ;modo video + colonne
030F FECC    DEC AH
0311 2EA3FF00  MOV CS:[00FF],AX
0315 3C07      CMP AL,07
0317 7507      JNZ 0320
0319 2EC706170100B0  MOV WORD PTR CS:[0117],B000
0320 A06200    MOV AL,[0062] ;pagina video
0323 2EA21601  MOV CS:[0116],AL
0327 9B        CBW
032B 03C0    ADD AX,AX
032A BBD8    MOV BX,AX
032C BB4750    MOV AX,[BX+50] ;posiz. cursore
032F 2EA30101  MOV CS:[0101],AX
0333 A14E00    MOV AX,[004E] ;offset pag. video
0336 B104      MOV CL,04
033B D3EB      SHR AX,CL
033A 0E        PUSH CS
033B 1F        POP DS
033C A31901    MOV [0119],AX
033F 96        XCHG SI,AX
0340 EB0900    CALL 034C
0343 5D        POP BP
0344 07        POP ES
0345 5F        POP DI
0346 5E        POP SI
0347 5A        POP DX
0348 59        POP CX
0349 5B        POP BX
034A 5B        POP AX
034B C3        RET
```

Figura 3 - Disassemblato della routine posta all'indirizzo 02FBH.

- caratteri di inizio;
- carattere che indica di quale sequenza si tratta;
- eventuali parametri richiesti dalla sequenza;
- carattere o caratteri terminatori.

Allora si poteva sapere subito quale tipo di funzione deve essere effettuata e la sua effettiva gestione poteva avvenire all'arrivo del carattere «terminatore».

Ecco che dunque al punto in cui siamo

```

03BC BB2700      MOV BX,0027 ;indirizzo tabella
03BF B3C303      ADD BX,+03
03C2 B03F00      CMP BYTE PTR [BX],00
03C5 7425        JZ 03EC
03C7 3807        CMP [BX],AL
03C9 75F4        JNZ 03BF
03CB BB4701      MOV AX,[BX+01]
03CE BB5F05      MOV BX,055F
03D1 43          INC BX
03D2 031EFB00    ADD BX,[00FB]
03D6 BA17        MOV DL,[BX]
03D8 32F6        XOR DH,DH
03DA 8BCA        MOV CX,DX
03DC 0BC9        OR CX,CX
03DE 7501        JNZ 03E1
03E0 41          INC CX
03E1 FFEO        JMP AX

```

Figura 7
Disassemblato della
routine posta
all'indirizzo 03BCH.

```

0364 3C3B        CMP AL,3B ; ";"
0366 750C        JNZ 0374
0368 FE060601    INC BYTE PTR [0106]
036C EBD601      CALL 0545
036F 33C0        XOR AX,AX
0371 8907        MOV [BX],AX
0373 C3          RET
0374 3C30        CMP AL,30 ; "0"
0376 7212        JB 038A
0378 3C39        CMP AL,39 ; "9"
037A 770E        JA 038A
037C EBC601      CALL 0545
037F 2C30        SUB AL,30
0381 8607        XCHG AL,[BX]
0383 B40A        MOV AH,0A
0385 F6E4        MUL AH
0387 0007        ADD [BX],AL
0389 C3          RET
038A 3C3D        CMP AL,3D ; "="
038C 7413        JZ 03A1
038E 3C3F        CMP AL,3F ; "?"
0390 740F        JZ 03A1
0392 3C22        CMP AL,22 ; virgolette
0394 7404        JZ 039A
0396 3C27        CMP AL,27 ; apice
0398 7522        JNZ 03BC
039A C704A203     MOV WORD PTR [SI],03A2
039E A20501      MOV [0105],AL
03A1 C3          RET

```

Figura 6
Disassemblato della
routine posta
all'indirizzo 0364H.

La routine relativa alla lettera «n» deve fornire il cosiddetto DSR («Device Status Report») e cioè una stringa di identificazione del terminale usato: nel nostro caso non ha senso parlare di terminale ed allora tale routine fornisce la posizione del cursore come coppia di numeri separati da un «;», rispettivamente la riga e la colonna.

Il fatto strano è che tale stringa è fornita non in output su video direttamente, ma bensì viene posta nel buffer di tastiera, pronta per essere processata come se fosse stata digitata dall'operatore: questo fatto è alquanto scomodo in quanto in generale tale stringa viene successivamente interpretata come un comando, ovviamente sconosciuto, il che comporta una segnalazione d'errore. Domandarsi il perché di questa scelta è come chiedersi perché alcune tastiere hanno i tasti funzione posti a sinistra in verticale (comodissimi) e viceversa altre tastiere ce li hanno in alto in fila orizzontale (scomodissimi) da usare, perché si rischia sempre di premere un altro tasto «di sotto»...

La routine relativa alla lettera «p» è invece utile in quei casi in cui si voglia associare ad un certo tasto (qualsiasi e non necessariamente quelli di funzione) una sequenza di caratteri, ad esempio un comando frequentemente usato; ma non solo: ad esempio si può cambiare il carattere visualizzato premendo un certo tasto.

Ad esempio, possedendo una tastiera italiana e volendo scrivere programmi, allora può essere più utile scambiare le lettere accentate (inutili nei programmi, ma nemmeno buone per i word processor in quanto aventi un codice maggiore di 128, fatto che il più delle volte è deleterio) con le più proficue parentesi quadre oppure quelle «graffe», viceversa indispensabili se si scrivono programmi in Pascal oppure in C.

Con questo terminiamo la nostra analisi del device driver ANSI.SYS e rimaniamo alla prossima puntata, dove parleremo di un argomento completamente nuovo, sul quale non anticipiamo nulla...

arrivati, devono essere letti alcuni parametri, tutti contraddistinti dalla presenza di un carattere numerico oppure da una stringa racchiusa tra virgolette, in tal modo perfettamente distinguibili dai caratteri specifici delle sequenze ANSI che nel nostro caso possono essere:

A B C D H J K R
(rigorosamente maiuscoli)

nonché

f h l m n p s u
(quest'altri rigorosamente minuscoli)

A seconda di quale di questi viene incontrato (siamo all'indirizzo 03BCH, figura 7) si salta ad uno dei 16 indirizzi, secondo una tabella posta a partire dall'indirizzo 002AH che riporta, tre byte alla volta, il carattere della sequenza ANSI e l'indirizzo della routine, come possiamo vedere in figura 8.

Si può dunque vedere (ma non riportiamo i disassemblati in questione, per motivi di spazio), che in pratica le routine relative a tutte quante le lettere, ad

eccezione di quelle pertinenti alla «n» ed alla «p», effettuano le loro operazioni chiamando l'INT 10H e cioè il gestore del video, chi in un modo, chi in un altro.

lettera	codice	entry point
A	41	043B
B	42	0443
C	43	042B
D	44	0436
H	4B	040A
J	4A	048B
K	4B	04A0
R	52	04D1
f	66	040A
h	6B	04DB
l	6C	04D4
m	6D	045B
n	6E	04B4
p	70	04EE
s	73	044B
u	75	0450

Figura 8 - «Entry point» delle routine che gestiscono le sequenze ANSI in base alla lettera indicata.

Le pubblicazioni Technimedia



AUDIOREVIEW

La più qualificata rivista italiana di elettroacustica ed alta fedeltà

MCMICROCOMPUTER

La più diffusa e più autorevole rivista italiana di informatica

OROLOGILE MISURE DEL TEMPO

La prima rivista per tutti gli appassionati di orologi

Technimedia

Via Carlo Perrier, 9 - 00157 Roma - Tel. 06/4180300 (12 linee ric. aut.)