

CCDOS(CCBIOS)2·10

程序清单注释

中科院计算所八室

一九八七·四

说 明

1. CCDOS (CCBIOS) 是用 8088 汇编语言编写的, 而本书中的程序清单是在 IBM PC 机上用 DEBUG 反汇编指令得到的。

2. 参照汇编语言的格式, 对程序清单进行了整理。非指令区 (数据区、地址表、字符串等) 按原意还原为 DW、DB 等伪指令。

3. 为了方便阅读, 对原程序清单作了较详细的程序段落划分, 并加有简明的程序说明。

4. DEBUG 反汇编得到的程序指令并不是严格的汇编语言原程序。例如, 转移指令、调子程序指令中的地址参数不是标号地址而是绝对地址。所以程序清单中指令前的地址仍然保留。

目 录

一、FILE1. EXE文件.....	0 0 1
二、CCCC. EXE文件.....	0 0 3
10H中断 显示器管理模块	0 0 3
数据区及工作区	0 0 3
指令区	0 0 6
字符字模库	0 6 0
16H中断 键盘管理模块	0 6 2
数据区及工作区	0 6 2
指令区	0 6 3
CCCC. EXE文件自身执行指令.....	1 1 9
三、ALI: 24P. EXE文件	1 2 1
数据区及工作区	1 2 1
17H中断 打印机管理模块入口	1 2 2
05H中断 打印屏幕管理模块入口	1 1 0
ALL24P. EXE文件自身执行指令.....	1 4 1

一. FILE1.EXE文件

FILE1.EXE文件是系统起动后, CCDOS调入内存执行的第一个文件, 这个文件的功能是检查系统盘上汉字字模库文件(CCLIB)的完好性。如果字模库完好性, 则为字模库申请内存。字模库首址暂时存在1FH中断向量指针中, 待CCCC.EXE文件使用。若读字模库出错, 则显示出错信息, 并要求重新引导系统。

```
0000 JMP    0009      ; 转0 0 0 9执行。
0002 NOP                ;
;
;
0003 DB    "cclib", 00 ; 汉字字模库文件名数据。
;
;
0009 PUSH  DS          ; 保存数据段指针。
000A PUSH  CS          ; 数据段指向代码段。
000B POP   DS          ;
000C LEA  DX, [0003]   ; 取汉字字模库文件名首址。
0010 MOV  AX, 3D00     ; 3DH号功能调用, 打开字模库文件供读。
0013 INT  21           ;
0015 JB   008A        ; 若打开文件失败, 转0 0 8 A出错处理。
0017 MOV  BX, AX       ; 字模库文件把柄→BX。
0019 MOV  AX, 4202     ; 4 2 H号功能调用, 移动读写指针到库文件尾部,
001C INT  21           ; DX(高位) AX(低位)返回文件尺寸。
001E JB   008A        ; 若出错, 转0 0 8 A出错处理。
0020 TEST DL, 0F       ; 测试字模库文件尺寸。
0023 JNZ  002A        ;
0025 CMP  AX, 0000     ;
0028 JZ   008A        ; 若文件尺寸为0, 转0 0 8 A, 出错处理。
002A PUSH  DX          ; 保存库文件尺寸。
002B PUSH  AX          ;
002C LEA  DX, [0000]   ; DS: DX指向当前代码段首址。(当前代码段首址为汉
; 字字模库首址)
0030 MOV  AX, 251F     ; 2 5号功能调用, 1FH中断指针指向当前代码
0033 INT  21           ; 首址, 即汉字字模库首址。(CCCC.EXE文件执行后
; 1FH中断指针还要修改)
0035 MOV  AH, 3E       ; 3 EH号功能调用关闭汉字字母库文件。
0037 INT  21           ;
0039 POP  AX           ; 恢复汉字字模库文件尺寸。
003A POP  DX           ;
003B MOV  CX, 000C     ; 计算以"节"为单位的字模库在内存所需空间。
003E SHL  DX, CL       ;
0040 MOV  CX, 0004     ;
```

```

0043 SHR    AX, CL    ;
0045 ADD    DX, AX    ;
0047 ADD    DL, 20    ;
004A ADD    DH, 10    ;
004D POP    DS        ; 恢复数据段。
004E MOV    AX, 3103  ; 8 1 H号功能调用, 为字模库申请内存。
0051 INT    21        ;
;
; 出错信息数据。
0053 DB    07, 07
0055 ^FILE^CLIB^ERROR^,0A,0D
0068 ^HIT A ANY KEY TO BOOT DOS ACAIN! ^, ^Y^
;
; 汉字字模库出错处理。
008A LEA    DX, [0053] ; 出错信息首址->DX
008E MOV    AH, 09    ; 9 H号功能调用, 显示出错信息。
0090 INT    21        ;
0092 MOV    AH, 07    ; 7 号功能调用, 等待键盘输入。
0094 INT    21        ;
0096 INT    19        ; 19H中断, 重新引导系统。

```


； 不允许处理光标也表示屏幕向上没有光标， FF允许处理光标也表示屏幕上有光标。

；
0056 DW 0001 ; 3 页面光标。图形方式下是提示行光标位置单元。
；
0058 DW 0001 ; 4 页面光标。
005A DW 0001 ; 5 页面光标。
005C DW 0001 ; 6 页面光标。
005E DW 0001 ; 7 页面光标。
0060 DW 0000 ; 光标起始结束扫描线号单元。它是 6845 的第 10, 11号内部寄存器的一个拷贝。
；
0062 DB 00 ; 当前显示页号。
；
0063 DW 0009 ; 显示器插件板端口地址。
；
0065 DB 00 ; 操作模式单元。它是插件板操作模式寄存器当前设置值的一个拷贝。
；
0066 DB 00 ; 颜色寄存器单元。它是插件板颜色寄存器当前值的一个拷贝。
；
0067 DB 00, 00, 00, 00, 00 ;
006C DB 00, 00, 00, 00 ; 提示行汉字内码暂存区。
0070 DB 00 ;
0071 DW 0000 ; 写操作时的内光标。写操作时它保存汉字内码第一字节对应的内光标值。也称作写汉字光标单元。
；
；
0073 DB 00 ; 提示行操作时的内码标志，当处理的是第一字节内码时设置为1。
；
0074 DB 00 ;
0075 DW 0000 ; 用于存放汉字字模库起始段段码。
；
0077 DB 27H ; 640×200图形方式颜色寄存器控制字。
；
0078 DW 18 DUP (0) ; 汉字字模加工区。它分左右 2 个半区，每个半区 18 个字节，分别用于存放左右 16 个字节的汉字字模。每半区最后 2 个字节用于存放空白，实现汉字行之间的间隔。ASCII码只用左半区。
；
009C DB 00, 00, 00, 00, 00 ;
00A1 DB 00, 00, 00, 00 ; 提示行写汉字时的汉字内码缓冲区。
；
； 图形方式下当前屏幕不计提示行只可显示 10 行汉字，而一般西文软件是按 25 行设计的。为实现兼容本系统有一个 25 行区，当前屏幕只是 25 行区的一个可浮动的窗口。当前屏幕在 25 行区的位置由下 2 个单元给出。
00A5 DB 00 ; 当前屏幕首行对应的 25 行区的行号，简称汉字
；
； 屏起始行。
00A6 DB 00 ; 当前屏幕（不包括提示行）尾行对应的 25 行区

; 的行号, 简称汉字屏结束行。
 ;
 00A7 DW 0000 ; 外光标单元。即用户程序定位的25行区光标, 也称
 ; 作25行区光标位置。
 00A9 DW 00C0 ; 写操作时的外光标, 用以判断高层软件的意图。
 ;
 00AB DW 0000 ; 读操作时的外光标, 用以判断高层软件的意图。
 ;
 00AD DW 0000 ; 读操作时的内光标, 读操作时它保存汉字内码第
 ; 一字节对应的内光标。
 00AF DB 00 ; 光标标志2, 取消自动光标方式下使用的光标。

; TTY方式写时高7位为1允许设置光标。对于扩展字符码(80H~A0H)
 ; 该标志第0位为0, TTY方式下也不设置光标。

;
 ; 以下三个数据区简称25区, 是一个虚屏幕区, 以字符计可存放80×25个字符。在图型方式
 ; 下当前屏幕是25行区的一个窗口, 只有10行。当前屏幕可以在25行区上下浮动, 实现中西
 ; 文软件兼容。25行区最后一行(由0开始计数是第24行)始终对应屏幕的提示行。

; 25行字符区

00B0 DB 00, 00, 00, 00, 00, 00, 00, 00, 25行字符区, 用于存放80×25行字符
 DB 00, 00, 00, 00, 00, 00, 00, 00; (每字符1字节)。ASCII码占一字节
 DB 00, 00, 00, 00, 00, 00, 00, 00; 位置, 汉字内码占2字节位置。本区的
 DB 00, 00, 00, 00, 00, 00, 00, 00; 最后一行(最后80个字节)始终对应当
 DB 00, 00, 00, 00, 00, 00, 00, 00; 前屏幕的提示行。
 DB 00, 00, 00, 00, 00, 00, 00, 00;

为节省篇幅, 以下数据略去。25行字符区共80×25个字节。

; 25行属性区

0880 DB 00, 00, 00, 00, 00, 00, 00, 00; 25行属性区, 每一字符位置与25行字符
 DB 00, 00, 00, 00, 00, 00, 00, 00; 区相应位对应。写入的数据是相应字符
 DB 00, 00, 00, 00, 00, 00, 00, 00; 的显示属性, 或颜色码。
 DB 00, 00, 00, 00, 00, 00, 00, 00;
 DB 00, 00, 00, 00, 00, 00, 00, 00;

为节省篇幅, 以下数据略去。25行属性区共80×25个字节。

; 25行标志区

1050 DB 00, 00, 00, 00, 00, 00, 00, 00; 25行标志区, 每1字符位与25行字符区
 DB 00, 00, 00, 00, 00, 00, 00, 00; 相应位对应。
 DB 00, 00, 00, 00, 00, 00, 00, 00; 80H 没有处理完的汉字内码
 DB 00, 00, 00, 00, 00, 00, 00, 00; (没有处理完的第一字节汉字内码)
 DB 00, 00, 00, 00, 00, 00, 00, 00; 01H 处理完的第一字节汉字内码
 DB 00, 00, 00, 00, 00, 00, 00, 00; 82H 处理完的第二字节汉字内码

为节省篇幅，以下数据略去。25行标志区共80×25个字节。

```

;
;
;
;
; 转移地址数据表
;
1820 DW 18E0 ; 0号功能模块入口地址
1822 DW 1A43 ; 1号功能模块入口地址
1824 DW 1A69 ; 2号功能模块入口地址
1826 DW 1C65 ; 3号功能模块入口地址
1828 DW 24B3 ; 4号功能模块入口地址
182A DW 1C9E ; 5号功能模块入口地址
182C DW 1D49 ; 6号功能模块入口地址
182E DW 1DF1 ; 7号功能模块入口地址
1830 DW 1E30 ; 8号功能模块入口地址
1832 DW 1E67 ; 9号功能模块入口地址
1834 DW 1EA9 ; 10(0AH)号功能模块入口地址
1836 DW 1CC2 ; 11(0BH)号功能模块入口地址
1838 DW 1EEB ; 12(0CH)号功能模块入口地址
183A DW 1EDA ; 13(0DH)号功能模块入口地址
183C DW 23FC ; 14(0EH)号功能模块入口地址
183E DW 1CE8 ; 15(0FH)号功能模块入口地址
1840 DW 24B8 ; 16(10H)号功能模块入口地址
1842 DW 25AD ; 17(11H)号功能模块入口地址
1844 DW 25C0 ; 18(12H)号功能模块入口地址
1846 DW 25D3 ; 19(13H)号功能模块入口地址
;
;
;
; ××××××××××××××××××
; ×10H中断 显示器管理模块指令区×
; ××××××××××××××××××
;
; CCCC。EXE文件调入内存后，将显示器模块调入内存并修改10H中断指针指向
; CCBIOS 10H中断入口地址，取代ROMBIOS 10H中断程序实现对显示器的管理。
;
;
; CCBIOS 10H中断入口
;
1848 STI ; 关中断。
1849 CLD ; 清方向标志。
184A PUSH ES ; 保存现场。

```

```

184B PUSH DS ;
184C PUSH DX ;
184D PUSH CX ;
184E PUSH BX ;
184F PUSH SI ;
1850 PUSH DI ;
1851 PUSH BP ;
1852 PUSH AX ;
185 MOV AL,AH ; 功能调用号→AL
1855 XOR AH,AH ; 0→AH
1857 SHL AX,1 ; 为转相应地址,乘2。
1859 MOV SI,AX ; 存入SI用以转移相应功能号地址。
185B CMP AX,0028 ; 功能号超过范围?
185E JB 1864 ;
1860 POP AX ; 超过,恢复恢复AX。
1861 JMP 1A3F ; 转1A3F中断返回。
1864 MOV AX,0923 ; 代码段段码→AX。
1867 MOV DS,AX ; 数据段指向代码段。
1869 MOV AX,B800 ; 彩色板CRT刷新区地址段码→AX。
186C MOV DI,[0010] ; 取设备置参数→DI。
1870 AND DI,0030 ;
1874 CMP DI,+30 ; 单色板?
1877 JNZ 187C ; 彩色板,接187C执行。
1879 MOV AX,B000 ; 单色板,单色板CRT刷新区地址段码→AX。
187C MOV ES,AX ; 附加段指向CRT刷新区地址段码。
187E POP AX ; 恢复功能调用号。
187F MOV AH,[0049] ; CRT工作模式→AH。
1883 2E CS ;
1884 JMP [SI+1820] ; 根据调用功能号转相应功能模块。
;
;
;
; 6845内部寄存器初始化数据表
;
1888 DB 38,28,2D,0A,1F,06,19,1C ; 40×25字符方式
1890 DB 02,07,06,07,00,00,00,00;
1898 DB 71,50,5A,0A,1F,06,19,1C; 80×25字符方式
18A0 DB 02,07,06,07,00,00,00,00;
18A8 DB 38,28,2D,0A,07,06,64,70; 图形方式
18B0 DB 02,01,06,07,00,00,00,00;
18B8 DB 61,50,52,0F,19,06,19,19; 80×25单色板 字符方式
18C0 DB 02,0D,0B,0C,00,00,00,00;
;
; CRT刷新区长度(容量)数据
;
18C8 DW 0800

```

```

18CA DW 1000
18CC DW 4000
18CE DW 4000
;
;
; 屏幕列数控制数据
18D0 DB 28,28,50,50,28,28,50,50; 分别对应 0~7 工作方式
;
; 操作模式寄存器控制字
18D8 DB 2C,28,2D,29,2A,2E,1E,29; 分别对应 0~7 工作方式
;
;
; 10H中断 0号功能模块
;
; 设置显示方式
;
; 入口: (AL) = 显示方式(0~7)
; (AL) = 0 40x25黑白字符
; (AL) = 1 40x25彩色字符
; (AL) = 2 80x25黑白字符
; (AL) = 3 80x25彩色字符
; (AL) = 4 320x200彩色图形
; (AL) = 5 320x200黑白图形
; (AL) = 6 640x200黑白图形
; (AL) = 7 单色板 黑白字符
;
18E0 MOV DX,03D4 ; 彩色板端口地址→DX。
18E3 MOV BL,00 ; 00→BL(彩色方式控制字)
18E5 CMP DI,+30 ; 单色板?
18E8 JNZ 18F1 ;
18EA MOV AL,07 ; 单色板,07→AL(CRT模式)
18EC MOV DX,03B4 ; 单色板端口地址→DX
18EF INC BL ; 0.1→BL(黑白方式控制字)
18F1 MOV AH,AL ; CRT模式→AH
18F3 MOV [0049],AL ; CRT模式→CRT工作模式单元。
18F6 MOV [0083],DX ; 插件板端口地址→显示器插件板地址单元。
18FA PUSH DS ; 保存数据段指针。
18FB MOV SI,0040 ; 数据段指向DOS数据区。
18FF MOV DS,SI ;
1900 MOV SI,0049 ; 取DOS数据区CRT工作模式单元地址。
1903 MOV [SI],AL ; 当前模式→DOS数据区CRT工作模式单元。
1905 MOV SI,0063 ; 取DOS数据区显示器插件板地址单元地址。
1908 MOV [SI],DX ; 插件板地址→DOS数据区插件板地址单元。
199A PUSH AX ; 保存功能号及CRT模式。
199B PUSH DX ; 保存插件板地址。

```

190C	ADD	DX,+04	;	取插件板操作模式寄存器端口地址。
190F	MOV	AL,BL	;	控制字→操作模式寄存器, 关闭显示。
1911	OUT	DX,AL	;	
1912	POP	DX	;	恢复插件板地址。
1913	SUB	AX,AX	;	数据段指向中断向量地址段码。
1915	MOV	DS,AX	;	
1917	LEA	AX,[2798]	;	取ASCII I码字符字模库地址。
191D	MOV	SI,007C	;	取1FH中断向量地址。
191E	MOV	[SI],AX	;	设置1FH中断向量指针, 指向ASCII I码字
1920	MOV	AX,CS	;	符字模库。
1922	MOV	[SI+02],AX	;	
1925	LDS	BX,[0074]	;	取1DH中断向量指针→DS: BX (DS恢复指向
			;	代码段, BX指向CS: 1888)
1929	POP	AX	;	恢复功能号及CRT模式。
192A	MOV	CX,0010	;	16→CX (CRT初始化表每个表项字节数。
192D	CMP	AH,02	;	哪种工作模式?
1930	JB	1942	;	模式0、1, BX指向第一个表项, 转19
			;	42执行。
1932	ADD	BX,CX	;	BX指向下一个表项。
1934	CMP	AH,04	;	哪种工作模式?
1937	JB	1942	;	模式2、3, BX指向第二个表项, 转19
			;	42执行。
1939	ADD	BX,CX	;	BX指向第三个表项。
193B	CMP	AH,07	;	哪种工作模式?
193E	JB	1942	;	模式4、5、6 BX指向第三个表项, 转19
			;	42执行。
1940	ADD	BX,CX	;	模式7, BX指向第四个表项。
1942	PUSH	AX	;	保存功能号及CRT模式。
1943	XOR	AH,AH	;	0→AH(6845内部寄存器号)
1945	MOV	AL,AH	;	内部寄存器号→AL。
1947	OUT	DX,AL	;	内部寄存器号→6845索引寄存器。
1948	INC	DX	;	DX指向6845内部寄存器。
1949	INC	AH	;	内部寄存器号+1。
194B	MOV	AL,[BX]	;	BX指针下的表项数据→AL。
194D	OUT	DX,AL	;	写6845内部寄存器。
194E	INC	BX	;	BX指向下一个表项数据。
194F	DEC	DX	;	DX恢复指向6845索引寄存器。
1950	LOOP	1945	;	CX-1, 转回1945继续写内部寄存器, 直到16个
			;	内部寄存器写完。
1952	POP	AX	;	恢复CRT工作模式。
1953	POP	DS	;	恢复数据段指针。
1954	XOR	DI,DI	;	0→当前页刷新区的首址单元。
1956	MOV	[004E],DI	;	
195A	MOV	BYTEPTR [0062],00	;	0→当前显示页单元。
195F	MOV	CX,2000	;	2000H→CX (CRT刷新区长度)
1962	CMP	AH,04	;	哪种工作模式?

1965	JB	1973	; 模式1、2、3转1973执行。
1967	CMP	AH,07	; 单色板?
196A	JZ	1970	; 单色板, 转1970执行
196C	XOR	AX,AX	; 图形方式, 0000→AX(空白)。
196E	JMP	1976	; 转1976清CRT刷新区为空白。
1970	MOV	CX,0800	; 0800H→CX(单色板刷新区长度)。
1973	MOV	AX,0720	; 空格及属性→AX。
1976	REPZ		; 用AX填充ES: DI~DI+CX的刷新区。
1977	STOSW		;
1978	MOV	WORD PTR [0060],0607, 0607	(光标起始结束扫描行)→光标起始结束扫描线单元。
197E	MOV	AL,[0049]	; 当前CRT模式→AL。
1981	XOR	AH,AH	; 0→AH
1983	MOV	SI,AX	; 当前CRT模式→SI。
1985	MOV	DX,[0063]	; 插件板地址→DX。
1989	ADD	DX,+04	; DX指向操作模式寄存器。
198C	CS,		;
198D	MOV	AL,[SI+18D8]	; 取当前模式下的控制字。
1991	OUT	DX,AL	; 控制字送操作模式寄存器。
1992	MOV	[0065],AL	; 控制字→显示器操作模式单元。
1995	PUSH	DS	; 保存数据段。
1996	MOV	DI,0040	; 数据段指向DOS数据区。
1999	MOV	DS,DI	;
199B	MOV	DI,0065	; 取DOS数据区操作模式单元地址。
199E	MOV	[DI],AL	; 控制字→DOS数据区操作模式单元。
19A0	PCP	DS	; 恢复数据段。
19A1	CS		;
19A2	MOV	AL,[SI+18D0]	; 取当前模式下的列数(每行字符数)控制字。
19A6	XOR	AH,AH	;
19A8	MOV	[004A],AX	; 列数控制字→屏幕列数单元。
19AB	AND	SI,000E	; 为查表, 变换当前模式数字。
19AF	CS,		;
19B0	MOV	CX,[SI+18C8]	; 取当前模式下的CRT刷新区长度。
19B4	MOV	[004C],CX	; 存刷新区容量单元。
19B8	MOV	DI,0050	; DI指向0页面光标位置单元。
19BB	XOR	AX,AX	; 0→AX
19BD	MOV	[DI],AX	; 0→0页面光标位置单元。
19BF	MOV	[DI+05],AL	; 0→光标标志1。
19C2	PUSH	DS	; 保存数据段。
19C3	MOV	AX,0040	; 数据段指向DOS数据区段码。
19C6	MOV	DS,AX	;
19C8	MOV	WORD PTR [DI],0000,	0→DOS数据区光标位置单元。
19CC	POP	DS	; 恢复数据段。
19CD	INC	DX	; DX指向插件板颜色寄存器。
19CE	MOV	AL,30	; 30(颜色寄存器控制字)→AL
19D0	CMP	BYTE PTR [0049],06,	640×200图形方式?

19D5	JNZ	19DA	;	不是, 接19DA执行。
19D7	MOV	AL,[0077]	;	是, 640×200方式控制字→AL。
19DA	OUT	DX,AL	;	写颜色寄存器。
19DB	MOV	[0066],AL	;	存颜色寄存器单元。
19DE	CMP	DYTE FTR [0049],04;		哪种工作模式?
19E3	JB	1A3F	;	字符方式, 转1A3F中断返回。
19E5	CMP	BYTE PTR [0049],07;		单色板?
19EA	JZ	1A3F	;	单色板, 转1A3F中断返回。
19EC	XOR	AX,AX	;	图形方式, 0→AX。
19EE	NOT	AX	;	FFFFH→AX
19F0	MOV	[0071], AX	;	FFFFH→写汉字光标单元(写操作时的内光标)
19F3	MOV	[00AD],AX	;	FFFFH→读操作时的内光标单元
19F6	NOT	AX	;	0→AX
19F8	MOV	[00A7],AX	;	0→25行区光标(外光标)单元
19FB	AND	BYTE PTR [0073],30;		初始化提示行汉字内码标志单元。
1A00	MOV	CX,0012	;	18→CX
1A03	PUSH	DS	;	附加段指向数据段。
1A04	POP	ES	;	
1A05	MOV	DI,0078	;	取字模加工区首址。
1A08	REPZ		;	字模加工区18个字节清0。
1A09	STOSW		;	
1A0A	MOV	[009C],AL	;	0→009C单元
1A0D	MOV	CX,0004	;	0→CX(清提示行内码缓冲区计数)
1A10	MOV	DI,00A1	;	取提示行内码缓冲区首址。
1A13	REPZ		;	清提示行内码缓冲区。
1A14	STOSB		;	
1A15	MOV	BYTE PTR [00A5],00;		0→汉字屏起始行单元。
1A1A	MOV	BYTE PTR [00A6],09;		9→汉字库结束行单元。
1A1F	MOV	[00AB],AX	;	0→读操作时的外光标单元
1A22	MOV	[00A9],AX	;	0→写操作时的外光标单元
1A25	MOV	CX,07D0	;	2000→CX(清25行区计数)
1A28	MOV	DI,00B0	;	取25行字符区首址。
1A2B	REPZ		;	清25行字符区。
1A2C	STOSB		;	
1A2D	MOV	CX,07D0	;	2000→CX
1A30	MOV	DI,1050	;	取25行标志区首址。
1A33	REPZ		;	清25行标志区。
1A34	STOSB		;	
1A35	MOV	CX,07D0	;	2000→CX
1A38	MOV	DI,0880	;	取25行属性区首址。
1A3B	MOV	AL,03	;	03→AL
1A3D	REPZ		;	初始化25行属性区。
1A3E	STOSB		;	
1A3F	POP	BP	;	恢复现场。
1A40	POP	DI	;	
1A41	POP	SI	;	

```

1A42 POP BX ;
1A43 POP CX ;
1A44 POP DX ;
1A45 POP DS ;
1A46 POP ES ;
1A47 IRET ; 中断返回。
;
;
; 10H中断 1号功能模块
;
; 建立光标类型用于光标大小
;
; 入口: (CH) = 光标起始扫描线
; (CL) = 光标结束扫描线
;
1A48 MOV AH,0A ; 设置6845内部寄存器号10。
1A4A MOV [0060],CX ; 光标起始结束扫描线→光标起始结束扫描线单元。
;
1A4E CALL 1A53 ; 调子程序,写10号(光标起始线)、11号(光标结束线)内部寄存器。
;
1A51 JMP 1A3F ; 转1A3F,中断返回。
;
; 子程序,写6845的2个连续的内部寄存器。进入时AH中有第1个内部寄存器号,CH、CL中有写2个寄存器的数据。
1A53 MOV DX,[0063] ; 取插件板地址(6845索引寄存器端口)。
1A57 MOV AL,AH ; 取第1个内部寄存器的号→AL。
1A59 OUT DX,AL ; 6845索引寄存器指向AH号内部寄存器。
1A5A INC DX ; DX指向6845内部寄存器端口。
1A5B MOV AL,CH ; 取第1个数据。
1A5D OUT DX,AL ; 写AH号内部寄存器。
1A5E DEC DX ; DX指向索引寄存器端口。
1A5F MOV AL,AH ; 取内部寄存器号。
1A61 INC AL ; 指向第2个内部寄存器。
1A63 OUT DX,AL ; 6845索引寄存器指向AH+1号内部寄存器。
1A64 INC DX ; DX指向内部寄存器端口。
1A65 MOV AL,CL ; 取第2个数据。
1A67 OUT DX,AL ; 写AH+1号内部寄存器。
1A68 RET ; 子程序返回。
;
;
; 10H中断 2号功能模块
;
; 设置光标位置
;
; 入口: (DH) = 新光标行值
; (DL) = 新光标列值

```

```

;      ( BX ) 新光标所在的页
;
1A69  CMP  DH,19      ; 设置光标行号超过范围?
1A6C  JB   1A70      ;
1A6E  JMP  1A3F      ; 超过, 什么也不作, 转1A3F中断返回。
1A70  CMP  DL,[004A] ; 设置光标列号超过范围?
1A74  JB   1A78      ;
1A76  JMP  1A3F      ; 超过, 什么也不作, 转1A3F中断返回。
1A78  MOV  CX,0040    ; 0040H→CX。
1A7B  PUSH DS        ; 保存数据段。
1A7C  MOV  DS,CX      ; 数据段指向DOS数据区。
1A7E  MOV  [0050],DX  ; 设置的光标位置→DOS数据区光标位置单元。
1A82  POP  DS        ; 恢复数据段。
1A83  CMP  BYTE PTR [0049],04 ; 哪种工作模式?
1A88  JB   1A94      ; 字符方式, 转1A94执行。
1A8A  CMP  BYTE PTR [0049],07 ; 单色板?
1A8F  JZ   1A94      ; 单色板, 转1A94执行。
1A91  JMP  1B33      ; 图形方式, 转1B33执行。
;
; 字符方式设置光标
1A94  MOV  CL,BH      ; 页号→CL
1A96  XOR  CH,CH      ; 0→CH
1A98  SHL  CX,1      ; 为寻址,  $CX \times 2 \rightarrow CX$ 。
1A9A  MOV  SI,CX      ;
1A9C  MOV  [SI+0050],DX ; 新光标位置→设置页面光标位置单元。
1AA0  CMP  [0062],BH  ; 设置页面是当前显示页?
1AA4  JNZ  1AAB      ; 不是, 经1AAB中断返回。
1AA6  MOV  AX,DX      ; 当前页, 光标位置→AX。
1AA8  CALL 1C54      ; 调子程序, 将新光标写入光标寄存器。
1AAB  JMP  1A3F      ; 转1A3F中断返回。
;
; 子程序, 图形方式下光标处理。抹原光标, 如果BP = 0, 设置新光标。
1AAD  TEST BYTE PTR [0055],FF ; 光标标志1 = FFH ( 屏幕有光标吗 ) ?
1AB2  JZ   1B32      ; = 0, 转1B32 子程序返回。
1AB4  TEST BYTE PTR [0054],FF ; 测试自动光标标志。
1AB9  JNZ  1AD1      ; 建立自动光标状态, 转1AD1处理光标。
1ABB  TEST BYSE PTR [00AF],FF ; 取消自动光标状态, 测试光标标志2, TTY,
; 方式?
1AC0  JZ   1B32      ; 不是, 转1B32, 子程序返回。
1AC2  TEST BYTE PTR [00AF],01 ; TTY方式, 光标标志2第0位 = 1 ?
1AC7  JNZ  1AD1      ; = 1, 转1A1D处理光标。
1AC9  OR   BYTE PTR [00AF],01 ; = 0, 第0位置1。
1ACE  JMP  1B32      ; 转1B32, 子程序返回。
1AD0  NOP            ;
1AD1  MOV  AX [0050]  ; 当前内光标位置→AX。
1AD4  PUSH AX        ; 保存内光标位置。

```


1AD5	MOV	AX,[0052]	;	原内光标位置→AX。
1AD8	MOV	[0050],AX	;	原内光标位置暂存当前行内光标位置单元。
1ADB	CALL	23BA	;	调子程序,AX返回原光标位置在刷新区的地址。
1ADE	CMP	BYTE PTR [0049],06	;	640×200图形方式?
1AE3	JZ	1AE7	;	是,转1AE7。
1AE5	SHL	AX,1	;	不是,AX×2→AX。
1AE7	ADD	AX,0230	;	AX+640(8个扫描行)=原内光标在刷新区的实际地址。
1AEA	MOV	SI,AX	;	
1AEC	ES,		;	
1AED	TEST	BYTE PTR [SI],FF	;	原内光标位置有光标吗?
1AF0	JZ	1B02	;	没有,转1B02执行。
1AF2	ES,		;	
1AF3	MOV	BYTE PTR [SI],00	;	抹去原光标。
1AF6	CMP	BYTE PTR [0049],06	;	640×200图形方式?
1AFB	JZ	1B02	;	
1AFD	ES		;	
1AFE	MOV	BYTE PTR [SI+01],00	;	彩色图形方式,再抹一个字节。
1B02	POP	[0050]	;	当前内新光标位置→当前内光标位置单元。
1B06	CMP	BP,+01	;	BP=1(设置新光标吗)?
1B09	JZ	1B2C	;	BP=1,不设置新光标,转1B2C返回。
1B0B	CALL	23BA	;	调子程序,AX返回新光标在刷新区的地址。
1B0E	CMP	BYTE PTR [0049],06	;	640×200图形方式?
1B13	JZ	1B17	;	是,转1B17。
1B15	SHL	AX,1	;	彩色图形方式,AX×2→AX。
1B17	ADD	AX,0280	;	AX+640(8个扫描)=新光标在刷新区的实际地址。
1B1A	MOV	SI,AX	;	
1B1C	ES,		;	
1B1D	MOV	BYTE PTR [SI],FF	;	在新光标位置设置光标。
1B20	CMP	BYTE PTR [0049],06	;	640×200图形方式?
1B25	JZ	1B2C	;	
1B27	ES,		;	
1B28	MOV	BYTE PTR [SI+01],FF	;	彩色图形方式,再设置1个字节。
1B2C	MOV	AX,[0050]	;	新当前内光标位置→原内光标位置单元。
1B2F	MOV	[0052],AX	;	
1B32	REF		;	子程序返回。
;				
; 图形方式设置光标,进入时DX中是光标在25行区的行列值、即外光标。				
1B33	MOV	BP,0001	;	设置参数。
1B36	CALL	1AAD	;	调子程序,BP=1只抹原光标。
1B39	CMP	DH,18	;	新外光标行是25行区的提示行?
1B3C	JNZ	1B41	;	
1B3E	JMP	1B55	;	是,转1B55执行。
1B40	NOP		;	
1B41	XOR	BH,BH	;	0→BH(图形方式,只有1个页面)
1B43	MOV	AL,[00A5]	;	汉字屏起始行→AL。