

## 8.4 BIOS 中断调用

### 8.4.1 BIOS 中断调用概述

BIOS (BasicInput/OutputSystem) 是 IBM-PC 机的监控程序, 它固化在微型机主板的 ROM 中, 其内容主要有系统测试程序 (Post)、初始化引导程序(Boot)、I/O 设备的基本驱动程序和许多常用程序模块, 它们一般以中断服务程序的形式存在。例如, 负责显示输出的显示 I/O 程序为 10H 号中断服务程序; 负责打印输出的打印 I/O 程序为 17H 中断服务程序等。

图 8.5 是用户程序和操作系统关系示意图, 由图可见 BIOS 程序直接建立在硬件基础上, 磁盘操作系统 (DOS) 和其他操作系统建立在 BIOS 基础上, 各种高级语言 则建立在操作系统基础上。用户程序可以使用高级语言, 也可以调用 DOS 或其他操作系统, 还可以调用 BIOS, 甚至直接指挥硬件设备。

通常应用程序调用 DOS 提供的系统功能, 完成输入/输出或其他操作, 这样做用户可以少考虑硬件, 实现起来容易。

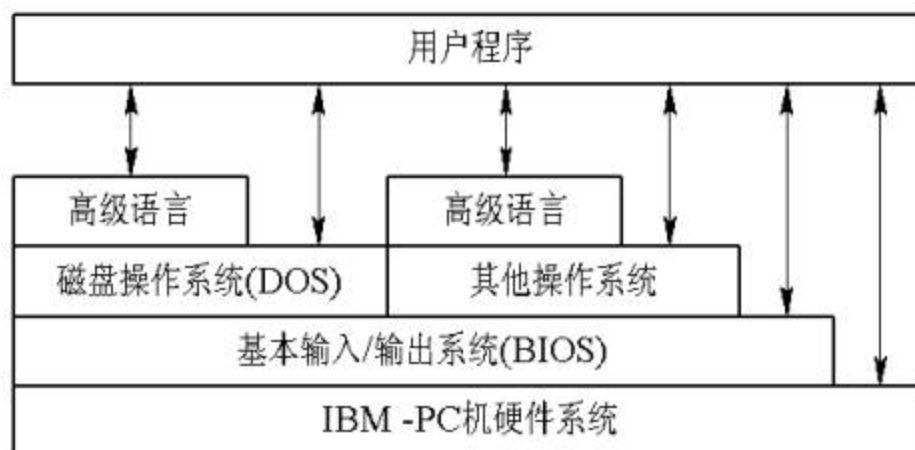


图 8.5 用户程序和操作系统关系示意图

应用程序直接对硬件编程的优点是程序的效率高,缺点是需要程序员对硬件性能有较深的了解。总的来说,编程复杂,所以一般不直接对硬件编程。

BIOS 中断程序处于 DOS 功能调用和硬件环境之间。和 DOS 功能调用相比其优点是效率高,缺点是编程相对复杂;和直接对硬件编程相比,优点是实现相对容易,缺点是效率相对低。

在下列情况下可考虑使用 BIOS 中断:

- (1)有些功能 DOS 没有提供,但 BIOS 提供了;
- (2)有些场合无法使用 DOS 功能调用;
- (3)其他原因。

#### 8.4.2 BIOS 中断调用方法

BIOS 的调用实际上是利用每一台计算机中 BIOS 固有的 I/O 操作程序来方便地解决自己的问题,由于这些已经在计算机中了,故使用时不必要再把它写入自己的程序,只要指明它的操作位置就可以了。

##### 1. BIOS 调用的基本操作

由于 BIOS 中的每一种功能调用往往包含不同的几个操作细节,因此调用时需要说明三部分,基本步骤为:

- (1)设置分功能号。按实现的操作功能的要求,给指定寄存器(通常为 AH)送入分功能号。
- (2)设置入口参数。按操作要求,给寄存器填写相应参数的内容(某些调用无参数)。
- (3)使用中断语句 INTn。执行调用的功能,其中 n 为中断号。



(4) 分析出口参数。具体步骤如图 8.6 所示。

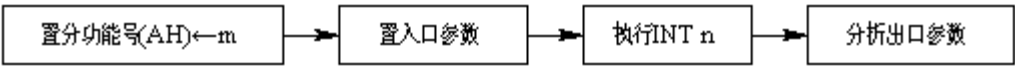


图 8.6 分析出口参数具体步骤

例如：

MOV AH, 0; 分功能号为 0

MOV AL, 10H; 置入口参数

INT 1AH

; 1AH 为中断号，功能为读时间计数器的值

## 2. 键盘输入功能调用

BIOS 中断 16H 为键盘输入。16H 号中断有三个功能，功能号要求放在 AH 中，

具体功能见表 8-3。

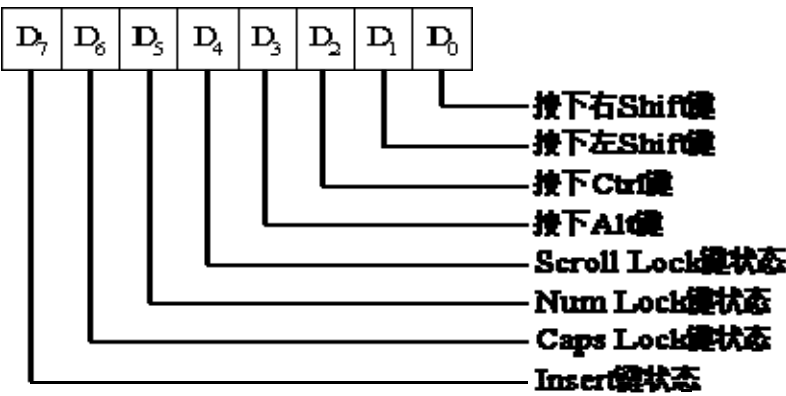


图 8.7 功能键状态

(AH)=0 时，读键盘键入的一个 ASCII 码字符，返回字符码在 AL 中，扫描码在 AH; (AH)=1 时，读键盘键入的一个 ASCII 码字符，并设置 ZF 标志位：

若按过一键（键盘缓冲区不空），ZF=0，否则 ZF=1；(AH)=2 时，读特殊功能键的状态送 AL，AL 各位含义见图 8.7

表 8-3BIOS 键盘中断（INT16H）

在内存 0040: 0017 中记录了双态键和组合键的状态，该字节称为键盘状态字节 KB-FLAG，其中，高 4 位表示键 Insert、CapsLock、NumLock 和 ScrollLock 是 ON(=1)还是 OFF(=0)，低 4 位表示键 Alt、Ctrl、左 Shift 和右 Shift 是否按动，若按动，则置 1。这 8 个键 有时又称为变换键

通过 INT16H 的功能 2 可以查看上述 8 个键的状态，有关程序段如下：

```
MOV AH, 2
```

```
INT 16H; 取键盘状态送入 AL 中
```

```
TEST AL, 10000000B
```

```
; 测试键盘状态字节第 7 位
```

```
JE INSOFF
```

```
; 如果是 0，则转至 INS 处于 OFF 的程序段
```

```
INSOFF:
```

用户通过程序改变键盘状态字节的内容，等效于敲入了对应的键，下面 4 句汇编语句使 Insert 键置于 ON 状态。

```
MOV AX, 0
```

```
MOV ES, AX
```

```
MOV AL, 10000000B
```

```
OR ES:[417H], AL
```

【例 8-6】从键盘读入 0~9 中任一数，根据不同数显示不同的字符串。要求用 BIOS 中断来接收键盘输入。

程序如下：

```
STACK SEGMENT PARA STACK
```

```
；堆栈段定义
```

```
DB 256 DUP ( ? )
```

```
STACK ENDS
```

```
DATA SEGMENT PARA PUBLIC 'DATA'；数据段定义，设每字符
```

```
THIRTY DB 30；串长 30
```

```
PARAM DB 128 DUP ( 0 )
```

```
MSG0DB' ILIKEIBMPC...'
```

```
MSG1 DB ' 8080PROGRAMMINGISRUN...'
```

```
MSG2 DB ' TIMETOBUYMANYDISKETTES...'
```

```
MSG3 DB ' THISPROGRAMWORKS...'
```

```
MSG4 DB ' TURNOFFTHATPRINTER! ...'
```

```
MSG5 DB ' IHAVEMANY...'
```

```
MSG6 DB ' THEPSPCANBEUSEFUL...'
```

```
MSG7 DB ' BASICWASEASIERTHENTHIS...'
```

```
MSG8 DB ' DOSISINDISPESABLE...'
```

```
MSG9DB' LASTMESSAGEOFTHEDAY..'
```

```
ERRMSG DB ' ERROR! INVALIDPARAMETER!!! '
```

```
DATAENDS
```

CODE SEGMENT PARA PUBLIC 'CODE'

; 代码段

START PROC FAR

ASSUME CS:CODE

PUSH DS

MOV AX, 0

PUSH AX

MOV AX, DATA

MOV ES, AX

ASSUME ES:DATA

MOV DX, AX

ASSUME DS:DATA

MOV AH, 0; 功能号为 0

INT 16H; 键盘输入, 字符在 AL 中

SUB AL, '0'; 减去字符 '0' 的 ASCII 码

JC ERROR; 键入字符编码小于 30H 转

CMP AL, 9; 是 9 吗

JAE RROR; 大于 9 转, 否则顺序执行

MOV BX, OFFSETMSG0; MSG0 位移送 BX

MUL THIRTY; (AL) × 30

ADD BX, AX

; 计算输入信息在信息表中的位移

CALLDISPLAY; 转子显示信息

RET; 返回

ERROR: MOV BX, OFFSET ERRMSG

; 出错显示信息位移送 BX

CALLDISPLAY; 转子显示信息

RET; 返回

DISPLAY PROC NEAR

MOV CX, 30; 显示字符个数

DISP1: MOV AL, [BX]

CALL DISPCHAR; 专显示字符子程序

INC BX

LOOP DISP1

MOVDL, 0DH; 显示回车符

CALL DISPCHAR

MOV DL, 0AH; 显示换行符

CALL DISPCHAR

RET

DISPLAY ENDP

DISPCHAR RPOC NEAR

; 显示 AL 中字符子程序

PUSH BX

MOV BX, 0

MOV AH, 14; 14 号功能调用为输出字符

INT 10H

POP BX

RET

DISPCHAR ENDP

CODE ENDS

END START

### 3. BIOS 显示功能调用

显示器是微型机系统重要的输出设备。显示器与微型机之间的接口电路又称做显示适配器(显示卡)。显示卡的种类很多,有 MDA 卡、HGC 卡、CGA 卡、EGA 卡和 VGA 卡等。IBM-PC 和 PC/XT 单色显示器使用单色显示器适配器 MDA (MonochromeDisplayAdapter) 卡,它不支持图形方式,只支持 80 列和 25 行的文本方式。文本方式又称为字符方式、字符/数字方式。IBM-PC 和 PC/XT 彩色显示器使用彩色/图形适配器 CGA(ColorGraphicsAdapter)卡,支持文本和图形两种方式且有单色、彩色两种显示。

有关显示输出的 DOS 功能调用不多,而 BIOS 调用(INT10H)的功能很强,主要包括设置显示方式,设置光标大小和位置,设置调色板号,显示字符和显示图形等,见表 8-4。下面做简单介绍。

表 8-4 中断类型 10H 的功能调用操作



6	屏幕初始化或上卷	AL=上卷行数(AL=0时全屏幕为空白) BH=卷入行属性 CH/CL=左上角行号/列号 DH/DL=右上角行号/列号	
7	屏幕初始化或下卷	AL=下卷行数(AL=0时全屏幕为空白) BH=卷入行属性 CH/CL=左上角行号/列号 DH/DL=右上角行号/列号	
8	读光标位置的属性和字符		AH/AL=属性/字符
9	在光标位置显示字符和属性	BH=显示页号；AL=字符 BL=属性；CX=字符重复次数	

#### 1) 显示方式的设置

INT10H 的 0H 功能用来设置显示方式，详见表 8-4。方式 0~6 是 CGA 卡的工作方式，其中，0~3 是文本方式，4~6 是图形方式。方式 7 是 MDA 卡的惟一工作方式。

文本方式在屏幕上显示字符，字符在屏幕上的位置用行、列坐标表示，如图 8.8 所示。25×80 文本方式下的行号为 0~24，列号为 0~79，则屏幕左上角为第 0 行、第 0 列，右下角为第 24 行、第 79 列。

也就是说，文本方式下屏幕上显示的字符位置是固定的。屏幕各行列号所确定的位置的显示内容，由显示卡上的显示存储器的对应位置的存储单元提供，这就是屏幕的“存储器映像”。即如果向某显示存储器写入字符的 ASCII 码，字符就会显示在屏幕的相应位置上。

对应的显示在屏幕上的每个字符在存储器中由连续的两个字节表示，一个字节保存字符 ASCII 码，另一个字节保存字符属性。

在屏幕上仅显示字母、数字和一些字符图形的方式就是文本方式。在文本方式下字符的属性字节既适用于单色显示也适用于彩色显示。

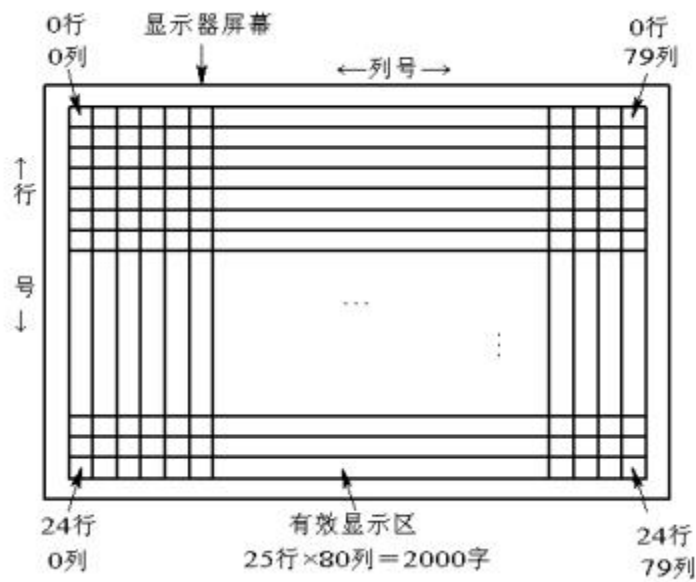


图 8.8 显示器屏幕字符位置图

(1) 单色字符显示。该显示方式定义了字符显示的特性，包括字符是否闪烁，是否加亮，是否反显（白底黑字）。单色字符显示属性字节的各位功能如图 8.9 所示，各位的不同组合将使字符呈现不同的属性，如表 8-5 所示。

(2) 彩色字符显示。在显示彩色文本时，字符的属性字节用于定义前景(显示的字符)和背景的颜色，每个字符可以选择 16 种颜色中的一种，而背景只能选择 8 种颜色 之一。图 8.10 给出了以 16 色文本方式显示的属性字节的各位内容。前景的 16 种颜色由 D0~D3 位组合，R、G 和 B 分别代表红(Red)、绿 (Green)和蓝(Blue)，BL 表示闪烁，I 为亮度。闪烁和亮度仅适用于前景。表 8-6 列出了 16 色字符方式颜色的组合。

图 8.10 16 色文本方式属性字节内容

表 8-6 16 种颜色的组合

(3) 显示存储器。对所有的显示适配器，文本方式下显示字符的原理都是一样的，差别仅是不同适配器的视频显示存储器的起始地址不同：MDA 的显存起始地址为 B000:0000；CGA、EGA 和 VGA 的显存起始地址为 B800:0000。每个字符的 ASCII 码和属性码存放于连续的两个字节中。显示存储单元与屏幕上字符的对应关系如图 8.11 所示。

在 25 × 80 的文本显示方式下，屏幕有 2000 个字符位置，由于每个字符用两个字节表示，所以需要显存容量为 4KB (4000B)。如果有 16KB 显存，则可以保存 4 屏显示字符的数据，通常称为 4 页数据。对 CGA、EGA 和 VGA 的 80 列显示方式，0 页在显存中的起始地址为 B800:0000，1 页为 B800:1000，2 页为 B800:2000，3 页为 B800:3000。

图 8.11 显示存储单元与屏幕上字符的对应关系

屏幕上某一字符位置在显存中的偏移地址可用下列公式计算：

字符偏址

= 页偏址 + ((行数 × 行宽) + 列数) × 字节数

图形方式下，在屏幕上可以显示“点”，点也称为像素、像元。像素在屏幕上的位置也用行列坐标值表示。在分辨率为  $320 \times 200$  的图形方式下，行号为  $0 \sim 199$ ，列号为  $0 \sim 319$ 。限于篇幅图形方式将在第 9 章讨论。

## 2) 光标的设置

IBM-PC 和 PC/XT 单色显示器上每个字符由  $7 \times 9$  点阵组成，字符框点阵为  $9 \times 14$ 。彩色显示器字符点阵为  $5 \times 7$ ，字符框点阵为  $8 \times 8$ ，光标的宽度为一个字符的宽度，长度最大可充满字符框。

光标在屏幕上指示字符的显示位置，它不是 ASCII 字符表中的字符。计算机有专门的硬件来控制光标，我们熟悉的光标符一般是一个下划线或方块符。

(1) INT10H 的 01H 功能是设置光标的大小和状态。光标起始行值放在 CH 的低 4 位，结束行值放在 CL 的低 4 位。CH 的 D4 位为光标禁止位，用来表示光标的状态。当该位为 1 时，光标不显示。单色显示器的光标大小的范围从  $0 \sim 13$ 。光标的默认方式是宽度为 2 列的闪烁的下划线。

(2) INT10H 的 02H 功能是设置光标位置。光标位置的行号设在 DH 寄存器中，列号设在 DL 中。在  $25 \times 80$  的显示方式中，坐标设在  $(0, 0)$  是屏幕的左上角， $(24, 79)$  是屏幕的右下角。BH 中必须包含被输出的页号，对单色显示器来说，页号总是 0。

**【例 8-7】**置光标开始行为 5，，并把它设置到第 5 行第 6 列。

MOV CH, 5; 开始行为 5

MOV CL, 7; 结束行为 7

MOV AH, 1; 定义行

INT 10H; 调用 BIOS

MOV BH, 4; 第 5 行

MOV DL, 5; 第 6 列

MOV BH, 0; 0 页

MOV AH, 2; 设置行

INT 10H; 调用 BIOS

### 3) 读光标位置

INT 10H 功能 3 是读光标位置，页号必须在 BH 中指定。此功能把光标位置的行号回送给 DH，列号回送给 DL。光标大小的参数填入 CH 和 CL 中，也就是说，在 CH 和 CL 中回送的是用功能 1 设置的光标参数。

**【例 8-8】**读 0 页的当前光标位置。

MOV AH, 3

MOV BH, 0

INT 10H

### 4) 选择显示页

INT10H 的功能 5 可由程序确定显存中的显示区域。ROMBIOS 将 CGA 的显存分为 4 页，每页  $25 \times 80$  个字符，或分为 8 页，每页  $25 \times 40$  个字符。每一页的起始地址在 1KB 的边界。这 4 页的起始地址分别为 B800:0000、B800:1000、B800:2000 和 B800:3000。

**【例 8-9】**选择显示页。

MOV AL, VPAGE; AL=显示页数

MOV AH, 5; AH=功能号

INT 10H

## 5)清屏和卷屏

INT 10H 功能 6 能使屏幕内容上卷指定的行，这个功能需要设置 7 个参数。如果屏幕的起始行、列不为 (0, 0)，结束的行、列不为 (24, 79)，则屏幕只有指定的一部分区域具有上卷的功能，这个屏幕上的部分区域叫做窗口(Window)，像这样的窗口可以在屏幕上设置多个，这些窗口都可独立使用。如果上卷超过指定窗口的顶部，这些行的内容就消失，出现在窗口底部的新行被填为空格，其属性由 BH 寄存器决定。

如果 AL=0，则实际完成的工作是清除屏幕的功能，它将按 AL 中的 BLANK 字符(0) 使指定的窗口为空白。程序 CLEAR:

```
CLEARPROC  
  
PUSH AX  
  
PUSH BX  
  
PUSH CX  
  
PUSH DX  
  
MOV AH, 6; 屏幕上卷功能号  
  
MOV AL, 0; AL=0 时，为清屏  
  
MOV CH, 0; CH: 屏幕左上角坐标的行号  
  
MOV CL, 0; CL: 屏幕左上角坐标的列号  
  
MOV DH, 24; DH: 屏幕右下角坐标的行号  
  
MOV DL, 79; DL: 屏幕右下角坐标的列号  
  
MOV BH, 7; 上卷行属性正常（即黑底白字）
```

INT10H

POP DX

POP CX

POP BX

POPAX

RET

CLEAR ENDP

10H 的功能 7 和功能 6 类似,也能使屏幕(或窗口)初始化或使屏幕(或窗口)的内容下卷指定的行,其他参数的设置与功能 6 一样。请看下面的例 8-10。

**【例 8-10】**清除左上角为(0, 0), 右下角为(24, 39)的窗口,初始化为反相显示,该窗口相当于全屏幕的左半部分。

MOV AH, 7; 选择屏幕下卷功能

MOV AL, 0;

MOV BH, 70H; 显示属性

MOV CH, 0; 左上行值

MOV CL, 0; 左上列值

MOV DH, 24; 右下行值

MOV DL, 39; 右下列值

INT 10H; 调用 BIOS

下面编写一个完整的程序(例 8-11)在 PC 机上运行。此程序在屏幕的中间建立一个 20 列宽和 9 行高的窗口,然后把键盘输入的内容在这个窗口上显示出来。

键入的字符将被显示在窗口的最下面一行，每当输入 20 个字符，该行就向上滚动，9 行字符输入完后，顶端行的内容消失。

【例 8-11】在屏幕中心的小窗口显示字符。

； 功能：显示窗口字符，按 Esc 结束程序

.MODE LSMALL

.STACK

.CODE

ESCKEYEQU1BH; ASCII 的 Esc 键

WINULC EQU 30; 窗口的左下列数

WINULR EQU 8; 窗口的左上行数

WINLRC EQU 50; 窗口的右下列数

WINLRR EQU 16; 窗口的右下行数

WINWIDTH EQU 20; 窗宽

； 主程序

MAINPROCFAR

LOCATE: MOV AH, 2; 定位光标

MOV DH, WIN\_LRR; DX←光标

MOV DL, WIN\_ULC

MOV BH, 0; 0 页

INT 10H; 调用 BIOS

MOV CX, WIN\_WIDTH; CX←窗宽

GET\_CHAR: MOV AH, 1; 键入一个字符



```
INT 21H; 调用 DOS

CMPAL, Esc_KEY; Esc 键处理

JZEXIT

LOOPGET_CHAR; 获取下一个字符

; 设置显示屏幕

MOV AH, 6

MOV AL, 1

MOV CH, WIN_ULR

MOV CL, WIN_ULC

MOV DH, WIN_LRR

MOV DL, WIN_LRC

MOV BH, 7

INT 10H; 调用 BIOS

JMP LOCATE

EXIT: MOVAX, 4C00H

INT 21H

MAIN ENDP

END MAIN
```

例 8-11 的程序中使用了几种 ROM 显示中断服务程序：清除屏幕、光标定位和上卷。如果在屏幕上同时有几个窗口工作，就要分别清除它们，这可通过设置不同的左上角坐标和右下角坐标来完成。

(1) 单色显示器字符的显示。该显示方式使用 INT10H 的 9、10 两功能，这两个功能的共同点是：都能在光标处显示一个字符，且显示后，光标不动。这就是说，当显示下一个字符前，一定要用 2 号功能将光标移至下一个字符位置。这两种功能的区别在于用 9 号功能显示的字符的属性是用户规定的，用 10 号功能显示的字符的属性是该位置上已有的属性(以前规定的)。

**【例 8-12】**从屏幕(40,50)位置开始,用反显示法连续显示 5 个闪烁的“\*”。

```
MOV AH, 2; 设置光标位置

MOV BH, 0; 单色显示器, 显示页永远为 0

MOV DH, 40; 行号

MOV DL, 50; 列号

INT 10H; 将光标设置在 (40, 50) 处

MOV AH, 9; 显示字符属性及属性功能

MOV AL, '*' ; 欲显示 '*'

MOV BH, 0; 页号=0

MOV BL, 0F0H; 属性 11110000

MOV CX, 5; 重复显示字符的个数

INT 10H
```

(2)彩色字符的显示。将上例中的 BL 赋值 11001110=0CEH, 就可从彩色屏幕(40, 50)处开始连续显示 5 个红底闪烁的黄色“\*”。由于彩色显示器工作方式有多种, 因此在设置光标, 显示字符前, 要先设置彩色显示器的工作方式。

**【例 8-13】**在品红背景下, 显示 5 个浅绿色闪烁的星号。

```
MOV AH, 09; 显示字符及属性
```

MOV AL, '\*' ; 字符送入

MOV BH, 0; 选 0 页

MOV BL, 0DAH; 设置颜色属性

MOV CX, 05; 显示 5 次

INT 10H; 调用 BIOS

(3)彩色文本方式下外框颜色的设置。彩色显示器工作在彩色文本方式时，屏幕分为两部分，中心为字符显示区，其余为外边框。外边框的颜色可通过 INT10H 的 0BH 号功能设置。

MOV AH, 11

MOV BH, 0; 入口参数

MOV BL, 1; 边框色值 1（蓝色）

INT 10H;

通过上述各例可以看出，利用 BIOS 中断编程可以不涉及硬件特征，编程相对比较容易，不同的 BIOS 中断具体参数不同，使用 BIOS 中断编程需要查询有关手册。

#### 4. BIOS 打印功能

BIOS17H 中断指令提供了由 AH 寄存器指定的三种不同的操作。

(1)BIOS17H 中断指令的功能 0 是打印一个字符的功能。要打印输出的字符放在 AL 中，打印机号放在 DX 中，BIOS 最多允许连接三台打印机，机号分别为 0，1 和 2。如果只有一台打印机，那么就是 0 号打印机，打印机的状态信息被回送到 AH 寄存器。

MOV AH, 0; 请求打印

MOV AL, CHAR; 写入打印字符

MOV DX, 0; 设置 0# 打印口

INT 17H; 调用 BIOS

(2)BIOS17H 中断指令的功能 1 是初始化打印机功能，用于初始化打印机并回送打印机状态到 AH 寄存器。如果把打印机开关关上然后又打开，打印机各部分就复位到初始值。此功能和打开打印机时的作用一样。在每个程序的初始化部分可以用 17H 的功能 1 来初始化打印机。

MOV AH, 01; 初始化打印机

MOV DX, 0; 设置 0# 打印口

INT 17H; 调用 BIOS

这个操作要发送一个换页符，因此这个操作能把打印机头设置在一页的顶部。对于大多数打印机，只要一接通电源，就会自动地初始化打印机。

(3)BIOS17H 中断指令的功能 2 是把状态字节读入 AH 寄存器。

打印机的状态字节如图 8.12 所示。

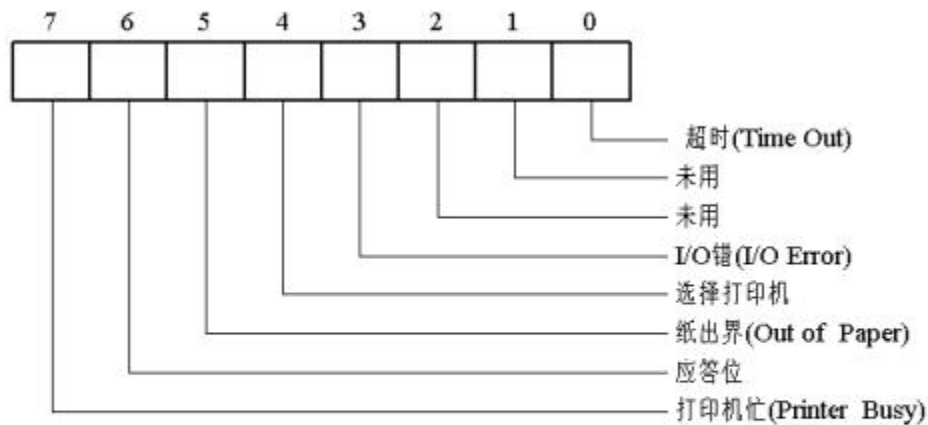


图 8.12 打印机的状态字节

打印机忙(PrinterBusy) 表示打印机正在接收数据，或正在打印，或处于脱机状态。应答位(Acknowledge)表示打印机已发出一个表明它已经接收到数据的信号。选择位 (Select)表示打印机是联机的。超时位(TimeOut)表示打印机发出忙信号很长一段时间了，系统将不再给它传送数据。表示打印出错的是第 5 位(纸出界)或第 3 位(I/O 错) 为 1。如果打印机没有接上电源，没有装上纸或没有联机，而打印程序已开始运行，这时显示器的指示光标会不停地闪烁，当接通 打印机的电源后，某些输出数据就会丢失。

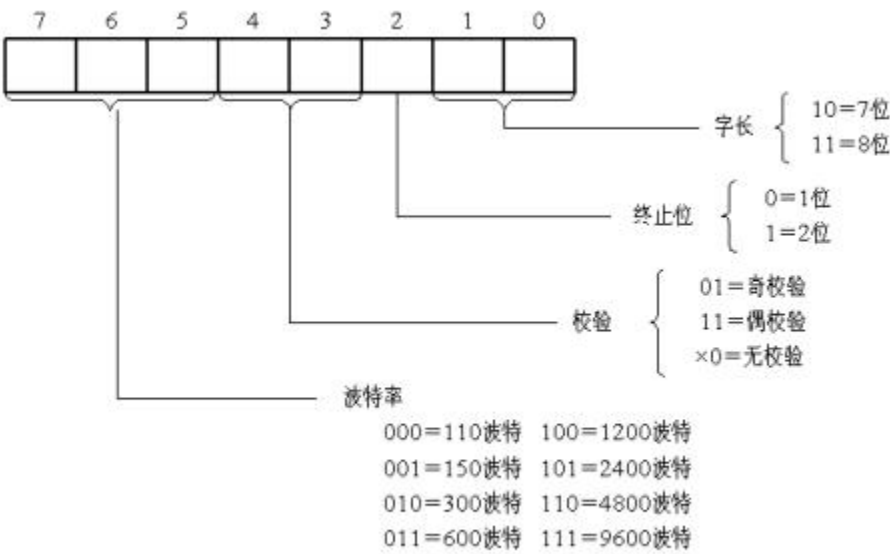
如果在打印程序中先安排指令测试打印机的状态，则 BIOS 操作就会返送回状态码，DOS 打印操作是自动进行测试的，但对各种情况都显示一个“纸出界”的信息。当打印机接通电源后，即开始正常打印，而且不丢失任何数据。

## 5. BIOS 串行通信口功能

IBM - PC 及其兼容机提供了一种有较强的硬件依赖性，但却比较灵活的串行口 I/O 的方法，即通过 INT14H 调用 ROMBIOS 串行通信口中断服务程序。具体功

能 见 表 8-7。该中断服务程序包括将串行口初始化为指定的字节结构和传输速率，检查控制器的状态，读写字符等功能。

表 8-7BIOS 串行通信口功能（INT14H）



【例 8-14】编写一程序，要求 0 号通信口的传输率为 2400 波特，字长为 8 位，1 位终止位，无奇偶校验。

MOV AH, 0; 串行通信口初始化

MOV AL, 0A3H; 0A3H=10100011B

MOV DX, 0; 指向 COM1

INT 14H; 调用 BIOS

返回参数中通信口状态字节各位置 1 的含义如图 8.14 所示

图 8.13 串行通信口初始化参数

在接收和发送过程中，错误状态位(1, 2, 3, 4 位)一旦被置为 1，则读入的接收数据已不是有效数据，所以在串行通信应用程序中，应检测数据传输是否出错。

奇偶错。通信线上(尤其是用电话线传输时)的噪音引起某些数据位的改变，产生奇偶错。通常检测出奇偶错时，要求正在接收的数据至少应重新发送一段。

超越错。在上一个字符还未被处理机取走，又有字符要传送到数据寄存器里时，就会引起超越错。如果处理机处理字符的速度小于单行通信口的波特率，则会产生这种错误。

帧格式错。如果接收/发送器未接收到一个字符数据的停止位，则会引起帧格式错。这种错误可能是由于通信线上的噪音导致停止位的丢失，或者是由于接收方和发送方初始化不匹配而引起的。

间断。间断有时候并不能算是一个错误，而是为某些特殊的通信环境设置的“空格”状态。当间断位为 1 时，说明接收的“空格”状态超过了一个完整的数据字传输时间。

PS/2 以及所有的 PC 机，AH=04 功能允许程序员将波特率设置为 19200，数据位的长度可以设置为 5, 6, 7 或已经不像 AH0 功能那样只能设置成 7 或 8 位。

中断向量地址一览表

一.8088 中断向量		
0-3	0	除以零
4-7	1	单步(用于 DEBUG)
8-B	2	非屏蔽中断
C-F	3	断点指令
10-13	4	溢出
14-17	5	打印屏幕
18-1F	6,7	保留
二.8259 中断向量		
20-23	8	定时器
24-27	9	键盘
28-2B	A	彩色/图形
2C-2F	B	异步通讯(secondary)
30-33	C	异步通讯(primary)
34-37	D	硬磁盘
38-3B	E	软磁盘
3C-3F	F	并行打印机



三.BIOS 中断		
40-43	10	屏幕显示
44-47	11	设备检验
48-4B	12	测定存储器容量
4C-4F	13	磁盘 I/O
50-53	14	串行通讯口 I/O
54-57	15	盒式磁带 I/O
58-5B	16	键盘输入
5C-5F	17	打印机输出
60-63	18	BASIC 入口代码
64-67	19	引导装入程序
68-6B	1A	日时钟
四.提供给用户的中断		
6C-6F	1B	Ctrl-Break 控制的软中断
70-73	1C	定时器控制的软中断
五.数据表指针		

74-77	1D	显示器参量表
78-7B	1E	软盘参数表
7C-7F	1F	图形表
六.DOS 中断		
80-83	20	程序结束
		系统功能调用
84-87	21	结束退出
88-8B	22	Ctrl-Break 退出
8C-8F	23	
90-93	24	严重错误处理
		绝对磁盘读功能
94-97	25	绝对磁盘写
98-9B	26	驻留退出
		DOS 保留
9C-9F	27	打印机
		DOS 保留
A0-BB	28-2E	
BC-BF	2F	
C0-FF	30-3F	
七.BASIC 中断		

100-17F	40-5F	保留
180-19F	60-67	用户软中断
1A0-1FF	68-7F	保留
200-217	80-85	由 BASIC 保留
218-3C3	86-F0	BASIC 中断
3C4-3FF	F1-FF	保留

DOS 功能调用

AH	功 能	调 用 参 数	返 回 参 数
00	程序终止  (同 INT 20H)	CS=程序段前缀	
01	键盘输入并回显		AL=输入字符
02	显示输出	DL=输出字符	
03	异步通讯输入		AL=输入数据
04	异步通讯输出	DL=输出数据	

05	打印机输出	DL=输出字符	
06	直接控制台 I/O	DL=FF (输入) DL=字符 (输入)	
07	键盘输入 (无回显)		AL=输入字符
08	键盘输入 (无回显) 检测 C trl-break		AL=输入字符
09	显示字符串	DS:DX=串地址“&”结束字符串	
0A	键盘输入到缓冲区	DS:DX=缓冲区首地址 (DS:DX)=缓冲 区最大 (DS:DX+1)=实字符数 际输入的字符数	
0B	检验键盘状态		AL=00 有输入, AL=FF 无输入
0C	清除输入缓冲区并请求指 定的输入功能	AL=输入功能号 (1, 6, 7, 8, A)	
0D	磁盘复位		清除文件缓冲区
0E	指定当前缺省的磁盘驱动 器	DL=驱动器号 AL=驱动器数 0=A, 1= B, ...	
0F	打开文件	DS:DX=FCB 首地址	AL=00 文件找到  AL=FF 文件未找到

10	关闭文件	DS:DX=FCB 首地址	AL=00 目录修改成功  AL=FF 目录中未找到文件
11	查找第一个目录项	DS:DX=FCB 首地址	AL=00 找到, AL=FF 未找到
12	查找下一个目录项	DS:DX=FCB 首地址 (文件名中带*或?)	AL=00 找到, AL=FF 未找到
13	删除文件	DS:DX=FCB 首地址	AL=00 删除成功, AL=FF 未找到
14	顺序读	DS:DX=FCB 首地址	AL=00 读成功  =01 文件结束, 记录中无数据  =02 DTA 空间不够  =03 文件结束, 记录不完整
15	顺序写	DS:DX=FCB 首地址	AL=00 写成功  =01 盘满  =02DTA 空间不够
16	建文件	DS:DX=FCB 首地址	AL=00 建立成功  =FF 无磁盘空间
17	文件改名	DS:DX=FCB  首地址 (DS:DX+1)=旧文件名	AL= 00 成功  = FF 未成功

		(DS:DX+17)=新文件名	
19	取当前缺省磁盘		AL=缺省的驱动器  驱动器 号 0=A, 1=B, 2=C, .
1A	置 DTA 地址	DS:DX=DTA 地址	
1B	取缺省驱动器 FAT 信息		AL=每簇的扇区数  DS:BX=FTA 标识字节  CX=物理扇区的大小  DX=缺省驱动器的簇数
1C	取任一驱动器 FAT 信息	DL=驱动器号	同上
21	随机读	DS:DX=FCB 首地址	AL=00 读成功  =01 文件结束  =缓冲区溢出  =03 缓冲区不满
22	随机写	DS:DX=FCB 首地址	AL=00 成功  =01 盘满  =02 缓冲区溢出

23	测定大小	DS:DX=FCB 首地址	AL=00 成功  文件长度填入 FCB  AL=FF 未找到
24	设置随机记录号	DS:DX=FCB 首地址	
25	设置中断向量	DS:DX=中断向量  AL=中断类型号	
26	建立程序段前缀	DX=新的程序段的段前缀	
27	随机分块读	DS:DX=FCB 首地址  CX=记录数	AL= 00 读成功  = 01 文件结束  = 02 缓冲区太小, 传输结束  = 03 缓冲区不满  CX=读取的记录数
28	随机分块写	DS:DX=FCB 首地址  CX=记录数	AL=00 写成功  AL=01 盘满  =02 缓冲区溢出
29	分析文件名	ES:DI=FCB 首地址	AL=00 标准文件

		DS:SI=ASCII 串  AL=控制分析标志	=01 多义文件  =FF 非法盘符
2A	取日期		CX=年 DH:DL=月:日(二进制)
2B	设置日期	CX:DH:DL=年:月:日	AL=00 成功, AL=FF 无效
2C	取时间		CH:CL=时:分  DH:DL=秒:1/100 秒
2D	设置时间	CH:CL=时:分  DH:DL=秒:1/100 秒	AL=00 成功  AL=FF 无效
2E	置磁盘自动读写标志	AL=00 关闭标志  AL=01 打开标志	
2F	取磁盘缓冲区的首址		ES:BX=缓冲区首址
30	取 DOS 版本号		AH=发行号, AL=版本号
31	结束并驻留	AL=返回码, DX=驻留区大小	
33	Ctrl-Break 检测	AL=00 取状态  AL=01 置状态(DL)	DL=00 关闭 Ctrl-Break 检测  =01 打开 Ctrl-Break 检测



		DL=00 关闭检测  =01 打开检测	
35	取中断向量	AL=中断类型	ES:BX=中断向量
36	取空闲磁盘空间	DL=驱动器号  0=缺省, 1=A, 2=B...	成功:AX=每簇扇区数, BX=有效簇数, CX =每扇区字节数, DX=总簇数  失败:AX=FFFF
38	置/取国家信息	DS:DX=信息区首地址	BX=国家码(国际电话前缀码)  AX=错误码
39	建立子目录(MKDIR)	DS:DX=ASCII 串地址	AX=错误码
3A	删除子目录(RMDIR)	DS:DX=ASCII 串地址	AX=错误码
3B	改变当前目录(CHDIR)	DS:DX=ASCII 串地址	AX=错误码
3C	建立文件	DS:DX=ASCII 串地址  CX=文件属性	成功:AX=文件代号  失败:AX=错误码
3D	打开文件	DS:DX=ASCII 串地址  AL=0 读, =1 写, =2 读/写	成功:AX=文件代号  失败:AX=错误码
3E	关闭文件	BX=文件号	失败:AX=错误码

3F	读文件或设备	DS:DX=数据缓冲区地址  BX=文件代号  CX=读取的字节数	读成功:  AX=实际读入的字节数 AX=0 已到文件尾  读出错:AX=错误码
40	写文件或设备	DS:DX=数据缓冲区地址  BX=文件代号  CX=写入的字节数	写成功:  AX=实际写入的字节数  写出错: AX=错误码
41	删除文件	DS:DX=ASCIIZ 串地址	成功:AX=00, 出错:AX=错误码(2, 5)
42	移动文件指针	BX=文件代号  CX:DX=位移量  AL=移动方式(0, 1, 2)	成功:DX:AX=新指针位置  出错:AX=错误码
43	置/取文件属性	DS:DX=ASCII 串地址  AL=0 取文件属性  AL=1 置文件属性  CX=文件属性	成功: CX=文件属性  失败:AX=错误码
44	设备文件 I/O 控制	BX=文件代号  AL=0 取状态	DX=设备信息

		=1 置状态 DX  =2 读数据  =3 写数据  =6 取输入状态  =7 取输出状态	
45	复制文件代号	BX=文件代号 1	成功:AX=文件代号 2  失败:AX=错误码
46	人工复制文件代号	BX=文件代号 1  CX=文件代号 2	失败:AX=错误码
47	取当前目录路径名	DL=驱动器号  DS:SI=ASCII 串地址	(DS:SI=ASCII 串)  失败:AX=错误码
48	分配内存空间	BX=申请内存容量	成功:AX=分配内存首址  失败:BX=最大可用空间
49	释放内存空间	ES=内存起始段地址	失败:AX=错误码
4A	调整已分配的存储块	ES=原内存起始地址  BX=再申请的容量	失败:BX=最大可用空间  AX=错误码
4B	装配/执行程序	DS:DX=ASCII 串地址	失败:AX=错误码

		ES:BX=参数区首地址  AL=0 装入执行  AL=3 装入不执行	
4C	带返回码结束	AL=返回码	
4D	取返回码		AX=返回代码
4E	查找第一个匹配文件	DS:DX=ASCIIZ 串地址  CX=属性	AX=出错代码 (02, 18)
4F	查找下一个匹配文件	DS:DX=ASCII 串地址  (文件名中带?或*)	AX=出错代码 (18)
54	取盘自动读写标志		AL=当前标志值
56	文件改名	DS:DX=ASCII 串 (旧)  ES:DI=ASCIIZ 串 (新)	AX=出错码 (03, 05, 17)
57	置/取文件日期和时间	BX=文件代号  AL=0 读取  AL=1 设置 (DX: CX)	DX: CX=日期和时间  失败: AX=错误码
58	取/置分配策略码	AL=0 取码	成功: AX=策略码

		=1 置码 (BX)  BX=策略码	失败:AX=错误码
59	取扩充错误码		AX=扩充错误码  BH=错误类型  BL=建议的操作  CH=错误场所
5A	建立临时文件	CX=文件属性  DS:DX=ASCII 串地址	成功:AX=文件代号  失败:AX=错误码
5B	建立新文件	CX=文件属性  DS:DX=ASCII 串地址	成功:AX=文件代号  失败:AX=错误码
5C	控制文件存取	AL=00 封锁, =01 开启  BX=文件代号  CX:DX=文件位移  SI:DI=文件长度	失败:AX=错误码
62	取程序段前缀地址		BX=PSP 地址

\* AH=0-2E 适用 DOS1.0 以上版本;

AH=2F-57 适用 DOS2.0 以上版本;

AH=58-62 适用 DOS3.0 以上版本.

BIOS 中断

INT	AH	功能	调 用 参 数	返 回 参 数
10	0	设置显示方式	AL=00 40*25 黑白方式  =01 40*25 彩色方式  =02 80*25 黑白方式  =03 80*25 彩色方式  =04 320*200 彩色图形方式  =05 320*200 黑白图形方式  =06 640*200 黑白图形方式  =07 80*25 单色文本方式  =08 160*200 16 色图形  =09 320*200 16 色图形  =0A 640*200 16 色图形  =0B 保留 (EGA)  =0C 保留 (EGA)  =0D 320*200 彩色图形 (EGA)	

			=0E 640*200 彩色图形 (EGA)  =0F 640*350 黑白图形 (EGA)  =10 640*350 彩色图形 (EGA)  =11 640*480 单色图形 (EGA)  =12 640*480 16 色图形 (EGA)  =13 320*200 256 色图形 (EGA)  =40 80*30 彩色文本 (CGE400)  =41 80*50 彩色文本 (CGE400)  =42 640*400 彩色文本 (CGE400)	
10	1	置光标类型	(CH) 0-3=光标起始行  (CL) 0-3=光标结束行	
10	2	置光标位置	BH=页号, DH, DL=行, 列	
10	3	读光标位置	BH=页号	CH=光标起始行, DH, DL=行, 列
10	4	读光笔位置		AH=0 光笔未触发, =1 光笔触发 CH=象 素行, BX=象素列, DH=字符行, DL=字 符列
10	5	置显示页	AL=页号	

10	6	屏幕初始化或上卷	AL=上卷行数, AL=0 整个窗口空白 BH=卷入行属性, CH=左上角行号, CL=左上角列号, DH=右下角行号, DL=右下角列号	
10	7	屏幕初始化或下卷	AL=下卷行数, AL=0 整个窗口空白 BH=卷入行属性, CH=左上角行号, CL=左上角列号, DH=右下角行号, DL=右下角列号	
10	8	读光标位置的字符和属性	BH=显示页	AH=属性, AL=字符
10	9	在光标位置显示字符及其属性	BH=显示页, AL=字符, BL=属性  CX=字符重复次数	
10	A	在光标位置显示字符	BH=显示页, AL=字符  CX=字符重复次数	
10	B	置彩色调板(320*200 图形)	BH=彩色调板 ID  BL=和 ID 配套使用的颜色	
10	C	写像素	DX=行 (0-199) CX=列 (0-639)  AL=像素值	
10	D	读像素	DX=行 (0-199) CX=列 (0-639)	AL=像素值
10	E	显示字符(光标前	AL=字符, BL=前景色	



		移)		
10	F	取当前显示方式		AH=字符列数, AL=显示方式
10	13	显示字符串(适用 AT)	ES:BP=串地址, CX=串长度  DH, DL=起始行, 列, BH=页号  AL=0, BL=属性  串:char, char...  AL=1, BL=属性  串:char, char...  AL=2  串:char, attr, char, attr,...  AL=3  串:char, attr, char, attr,...	光标返回起始位置          光标跟随移动          光标返回起始位置          光标跟随移动
11		设备检验		AX=返回值    bit0=1, 配有磁盘    bit1=1, 80287 协处理器    bit4, 5=01, 40*25BW(彩色板)    =10, 80*25BW(彩色板)

				=11, 80*25BW (黑白板)  bit6, 7=软盘驱动器号  bit9, 10, 11= RS-232 板号  bit12=游戏适配器  bit13=串行打印机  bit14, 15=打印机号
12		测定存储器容量		AX=字节数 (KB)
13	0	软盘系统复位		
13	1	读软盘状态		AL=状态字节
13	2	读磁盘	AL=扇区数  CH, CL=磁道号, 扇区号  DH, DL=磁头号, 驱动器号  ES:BX=数据缓冲区地址	读成功:AH=0  AL=读取的扇区数  读失败:  AH=出错代码
13	3	写磁盘	同上	写成功:AH=0 AL=写入的扇区数  写失败: AH=出错代码
13	4	检验磁盘扇区	同上 (ES:BX 不设置)	成功:AH=0 AL=检验的扇区数  失败:AH=出错代码

13	5	格式化盘磁道	ES:BX=磁道地址	成功:AH=0 失败:AH=出错代码
14	0	初始化串行通讯口	AL= 初始化参数  DX=通讯口号 (0, 1)	AH = 通讯口状态  AL = 调制解调器状态
14	1	向串行通讯口写字符	AL=字符, DX=通讯口号 (0, 1)	写成功: (AH) 7=0  写失败: (AH) 7=1  (AH) 0-5=通讯口状态
14	2	从串行通讯口读字符	DX=通讯口号 (0, 1)	读成功: (AH) 7=0, (AL)=字符  读失败: (AH) 7=1, (AH) 0-6=通讯口状态
14	3	取通讯口状态	DX=通讯口号 (0, 1)	AH=通讯口状态  AL=调制解调器状态
15	0	启动盒式磁带马达		
15	1	停止盒式磁带马达		
15	2	磁带分块读	ES:BX=数据转输区地址  CX=字节数	AH=状态字节  AH=00 读成功, =01 冗余检验错  =02 无数据传输, =04 无引导, =80 非法命令

15	3	磁带分块写	DS:BX=数据传输区地址  CX=字节数	AH=状态字节（同上）
16	0	从键盘读字符		AL=字符码，AH=扫描码
16	1	读键盘缓冲区字符		ZF=0 AL=字符码，AH=扫描码  ZF=1 缓冲区空
16	2	取键盘状态字节		AL=键盘状态字节
17	0	打印字符，回送状态字节	AL=字符，DX=打印机号	AH=打印机状态字节
17	1	初始化打印机，回送状态字节	DX=打印机号	AH=打印机状态字节
17	2	取状态字节	DX=打印机号	AH=打印机状态字节
1A	0	读时钟		CH:CL=时:分  DH:DL=秒:1/100 秒
1A	1	置时钟	CH:CL=时:分，DH:DL=秒:1/100 秒	
1A	2	读实时钟		CH:CL=时:分(BCD)  DH:DL=秒:1/100 秒(BCD)

1A	6	置报警时间	CH:CL=时:分(BCD)  DH:DL=秒:1/100 秒(BCD)	
1A	7	清除报警		