

CEGA 卡直接存取显示缓冲区技术

李晓华 (云南省军区自动化站)

摘要: 本文介绍了 CEGA 卡的体系结构、存储映象图、VRAM 组织结构等。说明了 CEGA 卡与 014 卡之间的区别。对如何直接操纵字符、图形显示缓冲区作了详细的讨论。

目前,我国大部分微机都配制了 CEGA 卡(即中文增强型图形适配器),其中长城 286、386 机型基本上全部采用了 CEGA 卡。这与原长城 0520-CH 机的 014 卡的显示方式有些不同。有些软件不能在 CEGA 卡上运行。这些软件大部分都是对视频缓冲区进行了直接操作。由于 CEGA 卡兼容了原 014 卡的汉字显示方式,所以仍然有些软件能在 CEGA 卡上运行。其中包括 BIOS 的 INT10。为了帮助计算机软、硬件人员更好地了解 CEGA 卡,本文介绍对 CEGA 卡显示缓冲区的直接存取技术。

一、CEGA 的体系结构

CEGA 卡做到了兼容 IBM 标准的 EGA 模式,又做到了兼容 014 卡(长城 CH 高分图形适配器)的汉字显示方式。所以 CEGA 卡的体系结构由两部份组成:一部分是 EGA(增加型图形单元),它是标准的 EGA 操作模式(通常又称为 IBM 工作方式)。另一部分由 GW014(字符方式)单元组成(通常称为 GW 工作方式)。这两部分电路独立工作互不影响。如图 1 所示:

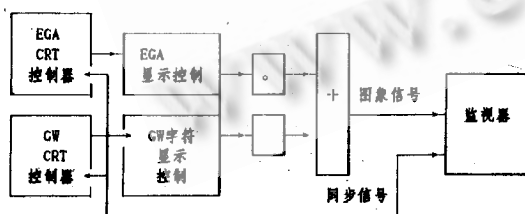


图 1 CEGA 卡体系结构框图

二、CEGA 存储映象及 VRAM 组织结构

1.CEGA 存储映象

CEGA 卡由于要兼顾 EGA、GW 两种工作方式,所以其存储映象应根据操作模式的不同而各异。如表 1 所示:

表 1

存储映象	模 式
B0000H-B7FFFH	7
B8000H-BFFFFH	0、1、2、3、4、5、6
A0000H-AFFFFH	D、E、F、10、11、12

其中模式 7、F 是为单色显示器而配置的。而 EGA 的模式(0-6)与 CGA(0-6)的模式完全一样。D、E、10、11、12 是 EGA 特有的图模式。

2.CEGA 卡的 VRAM 组织结构

CEGA 卡上的 BIOS 固化在地址空间 C0000H--C7FFFH 段,共 32KB。卡上存储器分三部分:字符发生器(32KB)、汉字显示存储器(64KB)、西文字符图形显示存储器(256KB)。

这三部分存储器不能同时被 CPU 存取,需要通过 I/O 寄存器 2DDH 来选择其中之一进行读写操作。另外还有一个控制器 2DEH。

显然,只要把上述两个控制器的最低位均置为 1,则就实现了 EGA 方式和 GW 方式的叠加显示。即实现

了在图形方式下显示汉字的目的。

三、字符显示缓冲区的结构

字符显示缓冲区共 64K 分成两区:基本区和扩展区,基本区与扩展区是一一对应的,基本区的起始地址为 B800:0000h,扩展区的起始地址为 B000:0000h(014 卡为 C000:0000h),每区 32K 字节,每页 4K,前 7 页为正常信息用,最后一页供系统作为提示区用。其对应关系如表 2 所示:

表 2

B000:0000 0001	ASCII 扩展码
	第二属性 不用
B800:0000 0001	ASCII 显示码
	第 1 属性

ASCII 码与显示缓冲区的对应关系

B000:0000 0001	汉字代码高 8 位
	第二属性
	\$ 1F
0002 0003	第二属性
B800:0000 0001	汉字代码低 8 位
	第一属性
0002 0003	汉字内码高 8 位
	第一属性

汉字代码与显示缓冲区的对应关系

字符显示缓冲区的字符并不是字符的机内码,而是字符的显示码。无论是 ASCII 字符,还是汉字字符,其显示码的长度都是 15 位。作为字符来说,通常又分为全角和半角字符。一般全角字符要占用 3 个标准字符的位置,所以汉字字符和全角字符就要占用 32 字节的位置。而半角字符只占用 16 字节的点阵字模。因此只要掌握了上述对应关系后,就不难实现对 CEGA 显示缓冲区直

接存取了。

四、字符显示缓冲区直接存取举例

在对字符显示缓冲区直接操作之前,先看看怎样从字符发生器中取字模点阵:在 CEGA 卡上字符发生器 CG 是由两片 ROM 和两片 RAM 组成,ROM 中固化有 7445 个 16 * 16 点阵的国家一、二级汉字和 256 个 8 * 16 个点阵的 ASEII 字符,RAM 和 4K 字节的存储器,用来存放用户自行定义的 128 个字符(16 * 16 点阵汉字或图形符号),当要进行汉字、ASCII 码的处理时,可以从 CG 中读取字符点阵。当然从其它字库(软件字库)中读取也是可以的,只是速度较慢。

从 ROM 中取字模点阵时,其计算公式为:

ASCII 字符:地址码 = ASCII 码

国际汉字:地址码 = 256 + (区码 - 1 - 6) * 94 + (位码 - 1)

汉字机内码 = 汉字区位码和 A0A0H 按位加

这样根据上述转换规则计算出相应的字符发生器的地址码,再并上全角,ROM 标志后,就得到完整的汉字显示码。再根据显示缓冲区的对应关系,把汉字显示码填到显示缓冲区中去。这样在 CRT 屏幕上就能显出相应的字符(全角字符和半角字符)。下面是把不字的显示码送到显缓冲区中去的程序片断。

```
:file mane CHARVIDE.ASM
```

```
mov ax,0B800h ;把示缓冲区的段地址送到 es 中
```

```
mov es,ax
```

```
mov di,0
```

```
mov al,charcode ;取 ascii 码或汉字机内码的第一字节到 al
```

```
cmp al,0a1h ;判断是否为汉字
```

```
jae hz ;若是转到 hz 处理部分
```

```
;不是执行 ascii 填充
```

```
stosb ;把 al->[di]
```

```
mov al,ascii-attrib ;ascii 的属性字
```

```
mov [di],al
```

```
push es ;保存 0b800h
```

```
mov ax,0b000h ;指向扩展区
```

```

mov es, ax
mov byte ptr[di-1],0 ;ascii 的扩展属性->[di-1]
pop es
inc di ;指向 ascii 码的下一位置
mov hzs,0 ;清汉字标志位
ret
hz: ;汉字机内码的第一字节处理;
cmp hzs,0
jnz writehz ;如不为 0,则是一个完整的汉字,
;转到汉字显示码计算.
mov hzs,al ;暂存机内码第一字节
ret

```

;汉字显示码计算:

```

writehz:
mov ah,hzs ;把机内码第一字节送 ah
and ax,7f7fh
sub ax,2121h
cmp ah,15
jb writevide
sub ah,6
writevide:
mov cx,ax
mov al,94
mul ch
oxr ch,ch
add ax,cx
add ax,256
or ah,40h ;此时在 AX 中得到完整的汉字显示码
;并加上了全角,rom 标志.
;以下是把汉字的显示码填到显示缓冲
;区中:
stosb ;填入汉字显示码的低字节
mov al,attrib1 ;汉字字符的第 1 属性
mov [di],al
push es
mov ax,0b000h ;指向扩展区
mov es, ax
mov byte ptr[di-1],ah ;填入高字节

```

```

mov al,attrib2 ;汉字字符的第 2 属性
mov [di],al
inc di
mov al,$ 1fh
mov [di],al
inc di
pop es
inc di ;指向下一个汉字位置
mov hzs,0 ;汉字标志位清 0
ret

```

五、图形显示缓冲区及有关寄存器编程

CEGA 卡图形显示缓冲区是一个 256K 字节的动态随机存取存储器(DRAM)的存储区,被分成 4 个彩色一面,每个页为 64KB,用来存储屏幕上的显示数据,所以它能够显示 16 种模拟彩色,由于它具有多种显示方式,使得各显示方式下区域划分不尽相同,其地址空间可以从 A0000H, B0000H, B8000H 开始。4 个彩色一面相同位置(它们的任意组合)的值决定了屏幕上的一个象素。

为了能对图形缓冲区进行读/写操作,需要掌握 EGA 中的各种控制器和寄存器。有关寄存器的操作,请参考其它书籍。

举例:读 0 方式时寄存器的设置

```

mov dx,3ceh ;图形控制器
mov al,5 ;模式寄存器
out dx,al
mov dx,3ceh
mov al,4
out dx,al ;读页面选择寄存器
mov dx,3cfh
mov al,colorpage;读指定为 colorpage 的页面
out dx,al
mov ax,0a000h ;图形显示缓冲区段地址
mov ds,ax
;送显示缓冲区数据

```