

# 基于 VB 实现的的汉字点阵字模的提取方法

范志敏, 刘洁莉, 补家武

(中国地质大学(武汉) 机电学院, 湖北 武汉 430074)

[摘要] 具体介绍点阵字库中字模的存储结构和存储方式, 给出了一种简便、实用的汉字点阵字模的提取方法以及其他字模信息的处理方法。

[关键词] 汉字字库; 字模提取

## 0 前言

如今在工业生产和日常生活中, LED 点阵屏和 LCD 点阵式液晶显示屏的应用越来越广泛, 如设备的操作显示屏、手机的显示屏以及大型广告显示屏等等。然而无论是 LED 点阵屏还是 LCD 点阵式液晶显示屏, 若要显示汉字, 都需要获得待显示汉字的点阵字模, 再由单片机控制系统进行处理。获取字模的最原始的方法是手工方法。这种方法不仅工作量大、效率低, 而且容易出错, 而使用昂贵的通用汉字库芯片, 对于一般的使用者来说又很不经济。为此, 本文介绍了一种简便、实用的提取汉字字模的方法, 利用该方法可以从汉字库中得到所有一级和二级汉字的点阵, 方便地制作合乎自己要求的专用小型点阵汉字库。

## 1 汉字的编码及汉字库简介

根据 GB2312-80, 每个汉字都有唯一的位置编码, 该编码就是字符的区号和行号的二进制代码, 称作该汉字的区位码<sup>[1]</sup>。计算机中的汉字与字符是混合在一起的, 区分汉字信息与 ASC II 码采取的方法是使表示汉字的两个字节的最高位等于 1。这种高位为 1 的双字节汉字编码就称为汉字的机内码, 又称内码<sup>[1]</sup>。机内码与区位码之间的变换关系如下:

$$\text{内码高位字节} = \text{区码} + 160; \text{内码低位字节} = \text{位码} + 160 \quad (1)$$

汉字库根据其中存储的汉字信息的不同可以分为点阵字库、矢量字库、曲线轮廓字库。根据汉字字体的不同, 每一种字库又可分为楷书、隶书、黑体、宋体等等。

点阵字库中存放着汉字的点阵字模信息。它根据点阵的大小, 又可分为 16 点阵字库、24 点阵字库等。当汉字的点阵数目增加时, 其显示质量越高, 但所占用的存储空间也越大。在同一点阵字库中, 各汉字字模信息所占的存储空间相同, 因而便于查询和提取。在矢量字库和曲线轮廓汉字库中则存放各汉字的索引表及其笔划或字形信息。不管是点阵汉字、矢量汉字还是曲线轮廓汉字, 要在 LED 点阵屏或 LCD 屏中显示它们, 最终还是要提取它们的点阵字模信息, 所以这里主要介绍点阵字库和点阵汉字。

## 2 汉字点阵库结构

不同点阵的汉字库中字模的存储方式有所不同。在 16 点阵字库中字模的存储方式是从左到右, 由上到下, 如图 1 所示。而在 24 点阵字库中则是从上到下, 从左到右, 如图 2 所示。

Font(0)	Font(1)
Font(2)	Font(3)
...	...
Font(30)	Font(31)

图 1 16 点阵字库中字模的存储方式

Font(0)	Font(3)	...	Font(69)
Font(1)	Font(4)	...	Font(70)
Font(2)	Font(5)	...	Font(71)

图 2 24 点阵字库中字模的存储方式

由汉字或符号的区位码可以唯一确定其在点阵字库中的起始位置, 计算方法如下:

[收稿日期] 2003-12-23; [修改稿收到日期] 2004-02-20

[作者简介] 范志敏(1979-), 男, 湖北武汉人, 硕士研究生, 研究方向为机械与电子工程。

$$\text{起始位置} = 94 \times (\text{区码} - 1) + (\text{位码} - 1) \quad (2)$$

这里仅介绍最常用的 16 点阵和 24 点阵汉字字模的提取,其它小点阵字模的提取方法类似。

对于 16 点阵字库,每个字模信息占用 32 个字节,根据式(1)和(2)可知,一个汉字的字模在 16 点阵字库中的起始位置可由下式求得:

$$\text{起始位置} = 32 \times [(\text{内码高字节} - 161) \times 94 + (\text{内码低字节} - 161)] \quad (3)$$

对于 24 点阵字库,每个字模占用 72 个字节,因而计算一个汉字字模在 24 点阵字库中的起始位置的公式与(3)式类似,为:

$$\text{起始位置} = 72 \times [(\text{内码高字节} - 161) \times 94 + (\text{内码低字节} - 161)] \quad (4)$$

在不同的 24 点阵字库中,各种非汉字符号的存放方式不一样。有的省去了前 15 个区,并将这 15 个区的符号单独存放,有的没有省去。如果省去了前 15 个区(如 UC DOS 中),则其起始位置的计算公式应修改为:

$$\text{起始位置} = 72 \times [(\text{内码高字节} - 176) \times 94 + (\text{内码低字节} - 161)] \quad (5)$$

知道了汉字字模在字库中的起始位置,进而就可以将该汉字的字模从相应的字库中取出。

### 3 具体应用实例

下面以 16×16 点阵字库为例,给出利用 Visual Basic 语言提取字模的实现方法。

首先,求得汉字的机内码:

```
intEachWord = Asc(strEachWord)
intNeiMa = 65536 + intEachWord      '得到机内码
```

然后,将得到的机内码转化为十六进制,再分别取得其高位和低位字节:

```
strQuMa = Mid(Hex(intNeiMa), 1, 2)
intQuMa = "&H" & strQuMa           '得到机内码的高位字节
strWeiMa = Mid(Hex(intNeiMa), 3, 2)
intWeiMa = "&H" & strWeiMa         '得到机内码的低位字节
```

接下来,根据公式(1)求出该汉字的区位码:

```
strQuMa = (intQuMa - 160) And 127   '得到区码
strWeiMa = (intWeiMa - 160) And 127 '得到位码
```

再由公式(3)计算出该汉字字模在字库中的起始位置:

```
intStartPosition = 32 * ((intQuMa - 1) * 94 + intWeiMa - 1)
```

那么,从这个起始位置顺序读取 32 次,就获得了该汉字的 16 点阵字模了。

提取 24 或 32 点阵字模的方法与提取 16 点阵字模类似,就不再重述。这里应注意的是:要获得汉字正确的点阵字模信息,必须要有相应的汉字点阵库文件(如 HZK16 等)。

如果要获得汉字的大点阵字模信息,就可以通过由图到点阵的方法。在 VB 中,可以方便地利用 API 函数来获取图像点阵,有关 API 函数描述为:

Getobject(hobject, cCount, lPobject):取得对指定对象进行说明的一个结构

GetBitmaPBits(hBitmap, dwCount, lPBits):将来自位图的二进制位复制到一个缓冲区用<sup>[3]</sup>。

在 API 浏览器加入以上两个 API 函数及 BITMAP 结构后,就可在程序中引用这两个函数了。

先将待取的汉字打印在一适当大小的图片框(其坐标设为象素 Pixels)中,然后再进行逐点扫描,从而获得其点阵信息。

实现代码如下:

```
Dim bmp As BITMAP, byteArray() As Byte
Dim hBitmap, res, totbyte As long
HBitmaP = Picture1.Picture.Handle
res = Getobject(hBitmap, len(bmp), bmp)
totbyte = bmp.bmWidthBytes * bmp.bmHeight
ReDim byteArray(totbyte - 1)
res = GetBitmapBits(hBitmap, totbyte, byteArray(0))
```

这样打印在图片框中汉字的点阵信息就存储在数组 ByteArray 中了。当然,这种方法也可以直接提取

所需图片的点阵信息。

将汉字字模提取出来后,就可以根据需要进行相关的处理,如:

(1)验证字模的正确性,即将字模代码转换为二进制的字符,在为“1”的地方填充上与底色不同的颜色,在为“0”的地方不填色(与底色相同),在图片框中将该汉字显示出来;

(2)将汉字或字符放大,可采取面积放大法或无级平滑放大法<sup>[2]</sup>;

(3)由于在 LED 点阵屏中显示汉字,根据显示方法可分为行扫描和列扫描,因此,可先将字模代码转换为二进制的字符,然后根据字模的存储方式和提取的需要进行组合而获得行扫描或列扫描的代码<sup>[3]</sup>;

(4)保存为 TXT 文件, TXT 文本是较简单的众所周知纯文本格式,可以方便查看相应的字模信息;

(5)保存为 ASM 文件,按照 ASM 文件的格式将字模代码进行相应的处理<sup>[4]</sup>;

(6)保存为 HEX 文件,将字模代码转换为二进制的字符,再根据 intel 的 HEX 文件格式要求进行排列组合。

#### 4 结束语

笔者在液晶显示屏上编写用户使用手册时,就是利用上述的方法提取汉字的点阵字模,取得了良好的效果。

#### [参 考 文 献]

[1]教育部考试中心. 全国计算机等级考试三级 A 类教程[M]. 北京:高等教育出版社,1998.

[2]蔡士杰,徐朝晖. 点阵汉字无级平滑放大的一种方法[J]. 计算机学报, 1993,16(1):77-80.

[3]Microsoft Corporation. Basic 6.0 中文版语言参考手册[M]. 微软(中国)有限公司译. 北京:希望电子出版社,1998.

[4]李 华,孙晓民. MCS-51 系列单片机实用接口技术[M]. 北京:航空航天大学出版社,1993.

[编辑:夏 宁]