

# 应用于 SPCE061A 的 GB2312 汉字语音库的制作

林洪艺, 黄丽敏, 林聪仁

(厦门大学信息科学与技术学院, 厦门 361005)

**摘要:** 汉语语音播放是许多嵌入式系统的重要功能, 语音库的制作又是实现这一功能的重要基础。阐述了制作 SPCE061A 压缩格式 GB2312 汉字语音库的方法。此方法分生成语音库文件及汉字读音序号索引表、裁剪语音库文件、压缩语音库文件等几个步骤。

**关键词:** 汉字语音库; 语音播放; 嵌入式; SPCE061A

## Producing GB2312 Chinese speech library for SPCE061A

LIN Hong-yi, HUANG Li-min, LIN Cong-ren

(School of Information Science and Technology, Xiamen University, Xiamen 361005, China)

**Abstract:** Chinese speech broadcasting is an important function in embedded systems, and one of its important foundations is producing speech library. This article introduces a method of producing GB2312 Chinese characters' speech library of SPCE061A compressed format. The major steps of the method are as follows: generating GB2312 speech library and index table, tailoring speech library files, and compressing speech library files.

**Key words:** Chinese characters' speech library; speech broadcasting; embedded; SPCE061A

### 0 引言

近年来,随着嵌入式系统技术的不断发展,汉语语音播放应用越来越广泛。汉语语音播放主要有录音芯片法、语音合成芯片法及汉字语音库法等方法。语音库法将不同读音的单字读音文件存储在系统空余的Flash空间或外扩Flash芯片上,降低了系统成本。在PC机上用微软的Visual C++ 6.0及Speech SDK开发环境可以得到每个汉字语音的WAV文件。本文阐述了GB2312汉字标准语音库的生成、汉字读音序号索引表的生成、汉字语音库的裁剪及压缩。该汉字语音库由1220个不同读音文件组成,可实现不同汉字内容的组合播放。将此汉字语音库应用于SPCE061A硬件平台,可开发出多种通用汉语语音播放系统。

SPCE061A是凌阳公司的16位单片机,具有DSP指令、可编程音频处理功能,配合凌阳压缩算法及音频播放函数,在进行其它操作的同时可很好地完成语音播放功能。限于文章篇幅,本文仅介绍语音库的制作。

### 1 汉字语音库及索引表生成

GB2312一、二级汉字有6763个,但只有1306个读音,对少数在特定词组中有特殊发音的单字,Speech SDK只支持整个词组的发音,所以实际提取1220个不同读音文件,可以满足应用需要。汉字语音库若由6763个汉字读音文件组成,容量较大,但汉字与读音一一对应,易于查找;若由1220个不同单字读音文件组成,容量较小,需建立一个汉字与语音对应的汉字读音序号索引表。为节省存储芯片成本,本文采用第二种方法。

本文以微软的Visual C++ 6.0及Speech SDK 5.1为开发环境,按GB2312汉字的内码顺序,生成汉字的读音文件,并对读音文件的内容进行比较,提取不同单字读音文件,与此同时生成6763个汉字与1220读音文件对应的汉字读音序号索引表。

微软的Speech SDK 5.1是基于COM组件的

收稿日期: 2007-09-14

基金项目: 厦门大学985二期信息创新平台项目资助

作者简介: 林洪艺(1982-),男,硕士研究生,研究方向为水声通信技术。

Windows 软件开发包, 供桌面语音程序开发者进行二次开发。本文通过调用 Speech SDK 中 SAPI(语音应用程序接口) 的两个 COM 接口: ISpVoice 和 ISpStream, 间接调用 Speech SDK 的 TTS 发音引擎, 提取汉字的读音。ISpVoice 和 ISpStream 两个接口函数的详细说明请参阅 Speech SDK 5.1 的帮助文档 sapi.chm。

### 1.1 主程序流程图

生成汉字语音库的主程序流程如图 1 所示, 首先初始化对话框的属性, 然后准备调用 SAPI。由于 SAPI 是以 COM 为基础, 所以调用前须初始化 COM, 代码如下:

```
if( FAILED( CoInitialize( NULL) ) ) return FALSE;
```

同样, 在程序结束前, 亦须释放 COM, 代码如下:

```
CoUninitialize();
```

程序的两个主要按钮为 IDC\_GenChars 及 IDC\_Standardize。初始化 COM 后, 程序等待按钮单击事件。若单击按钮为 IDC\_GenChars, 则进入语音库素材提取模块, 根据汉字内码顺序, 生成 GB2312 汉字 1220 个读音文件及汉字读音序号索引表; 若单击按钮为 IDC\_Standardize, 则进入语音库裁剪模块, 将 1220 个不同单字读音文件统一裁剪为 6.5kB; 若单击按钮为 IDCANCEL, 则释放主程序所用资源, 退出程序。

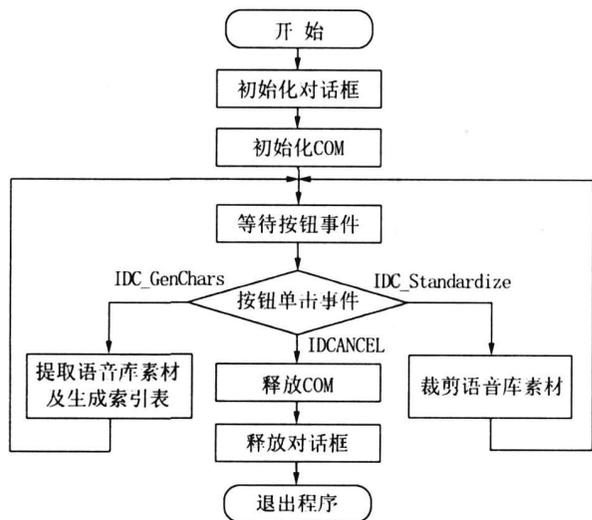


图 1 主程序流程图

### 1.2 IDC\_GenChars 按钮处理函数

IDC\_GenChars 按钮处理函数完成汉字语音库及汉字读音序号索引表生成功能。将 6763 个汉字的读音文件存到“TTS”文件夹; 比较文件内容, 选出不同的读音文件, 复制到“TTSD”文件夹; 将 6763 个汉字在 TTSD 下读音文件列表中的序号写入汉字读

音序号索引表 table.asm 文件。索引表中每行的格式为“.dw xxxx”, 图 2 列出部分内容。

IDC\_GenChars 按钮单击处理函数流程如图 3 所示。首先初始化函数中的变量, 然后进入提取汉字语音的循环。以 index 为循环计数, 次数为 6768, 按汉字内码顺序, 每次生成一个汉字的读音文件, 同时将汉字读音序号写入索引表。汉字内码的高、低字节与区位码对应, 关系为: 内码高字节= 区码+ A0H, 内码低字节= 位码+ A0H, 且每区有 94 个汉字。因此, 每次循环开始时, 若 index 为 94 的整数倍, 即待处理汉字为新一区的首字, 则将汉字内码高字节加 1, 低字节恢复为 0xA1。内码计算后, 创建发音辅助对象, 调用 Speech SDK 的发音引擎, 生成当前汉字的读音文件(这里用 a.wav 表示), 并存至 TTS 文件夹。然后将 a.wav 与 TTSD 文件夹内的读音文件一一进行对比, 若找到相同文件, 则将序号写入 table.asm; 若未找到相同文件, 则说明 a.wav 为新读音, 将其复制到 TTSD 文件夹, 将序号写入 table.asm。最后, 汉字内码低字节加 1, 释放发音辅助对象, 若 index 未达到 6768, 则继续下一次循环; 否则, 结束提取汉字语音的循环。读音相同的汉字在 Speech SDK 发音引擎生成的读音文件大小相同, 即文件大小不同则读音必不同, 所以读音文件比较时, 先比较大小, 若大小相同再比较内容, 这样可以节省程序运行时间。

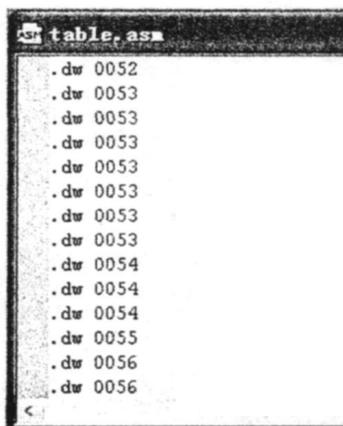


图 2 汉字读音序号索引表

## 2 汉字语音库的裁剪

用本文方法生成 1220 个语音库读音文件, 音频格式为 PCM 8kHz 16Bit Mono。用音频编辑软件 Cool Edit 对读音文件波形进行分析时, 发现文件长度都在 1.3 秒左右, 而实际发音部分约 0.35 秒, 余下部分为静音。为减小语音库的大小, 将每个汉字的读音文件统一裁减为 6.5kB, 实验表明, 这个长度可以

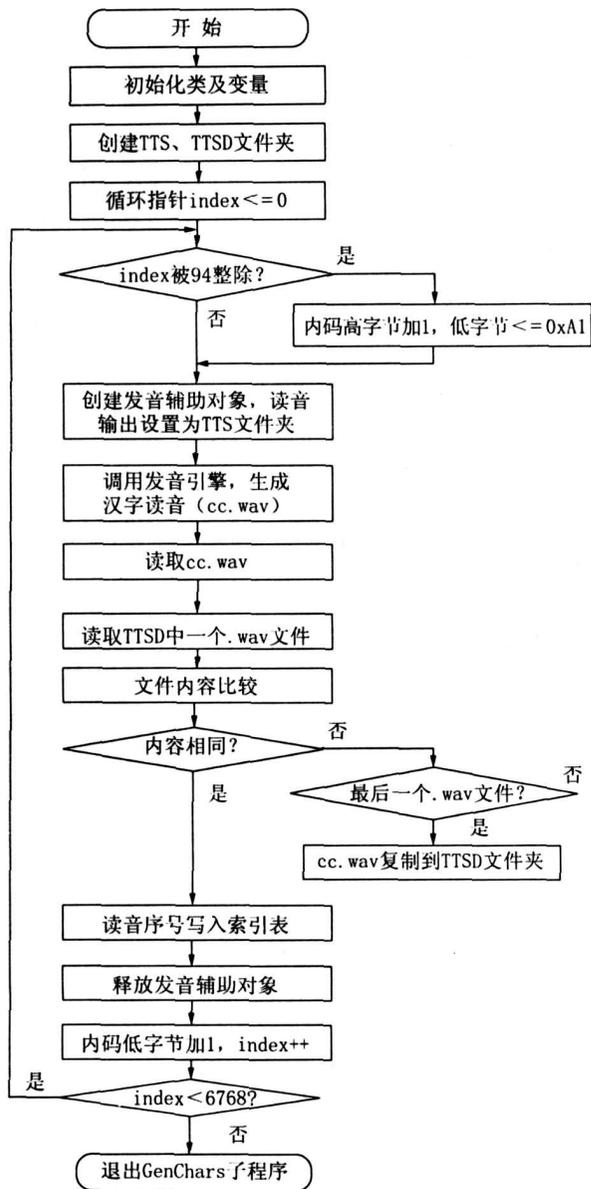


图3 IDC\_GenChars按钮处理函数流程图

基本保留完整发音波形。

在程序中用 IDC\_Standardize 按钮的单击处理函数来实现这个裁剪功能。由于裁剪改变了 WAV 文件大小, 故须对 WAV 文件中文件大小信息进行修改。WAV 读音文件由文件头和数据体两部分组成, 前 38 字节为文件头, 其中第 5 至 8 字节内容为文件长度减 8, 数据体前 8 字节为数据头, 其中第 5 至 8 字节为实际语音数据长度。对 6.5kB 的 WAV 文件, 这两个大小信息分别为  $6656 - 8 = 6648$  和  $6656 - 38 - 8 = 6610$ , 按 Intel 小端模式存储为 F8, 19, 00, 00 和 D2, 19, 00, 00。

相同长度的 WAV 文件, 前 46 字节内容相同, 程序中先将前 46 字节内容存至字节数组 buffer1, 然后进入 while 循环, 每次打开 TTSD 文件夹下一个读音

文件, 将该文件第 47 至 6656 字节的语音数据存至字节数组 buffer2, 用 buffer1、buffer2 内容创建一个新的 WAV 读音文件。裁剪处理后的语音库总大小从 24.1MB 缩小为 7.74MB, 为原来的 32.1%。

### 3 应用于 SPCE061A 汉字语音库的压缩

裁剪后的 7.74M 语音文件对嵌入式应用来说还略显过大, 用凌阳提供的音频压缩算法, 可对语音文件进一步压缩, 节省存储空间和成本。凌阳音频压缩软件 Compress Tool 的压缩比范围为 8:1 到 80:1.5。采用压缩比为 80:3 的 SACM\_S480 算法对 WAV 文件进行压缩, 每个文件由 WAV 格式的 6.5kB 压缩为大小 306 字节。48k 凌阳音频文件, 缩减为原来的 4.6%。

1220 个凌阳压缩音频格式的语音文件总大小为 364.6kB, 用 ResWriter 工具将其烧写进 Flash 芯片后, 会自动生成一个语音资源索引表, 大小为 14.3 kB, 每个文件在表中占 12 字节: 起始地址 4 字节、文件长度 4 字节、结束地址 4 字节。语音库文件与语音资源索引表总大小为 378.9kB, 若用 512kB 的凌阳 SPR4096 Flash 芯片存储, 则烧写后仍有 133.1kB 的剩余空间。汉字读音序号索引表编译后大小为 13.21kB, 可存至同一芯片, 节省系统的资源与成本。实验表明, 用此压缩的语音库文件播放, 语音基本清晰自然, 能适用于大部分嵌入式语音播放系统。若对音质要求较高, 可采用较低压缩比的算法, 当然要用更大容量的 FLASH。

### 4 结束语

本文所开发的软件在 PC 上制作了凌阳压缩音频格式的汉字语音库。实验表明, 用此语音库播放的语音具有较好的清晰度与可懂度, 可以构建基于 SPCE061A 的廉价实用的汉语语音播放系统。当然系统应用中有一些问题待解决, 如汉语中的多音字在不同词组中的辨别处理, 多个汉字读音在拼接中的平滑过渡等。

#### 参考文献:

- [1] Microsoft Corporation. TextToSpeech Tutorial[Z]. Speech SDK5.1 sapi.chm.
- [2] Microsoft Corporation. Simple TTS Application[EB/OL]. MSDN <http://www.msdn.microsoft.com>.
- [3] 李晶皎. 嵌入式语音技术及凌阳 16 位单片机应用[M]. 北京: 北京航空航天大学出版社, 2003.
- [4] 肖玮. 使用 SAPI 实现语音识别与合成[J]. 现代计算机, 2005(2) (总第 205 期): 91-94.
- [5] 袁嵩. 一个 TTS 系统的实现方案[J]. 计算机工程与应用, 2004 (21): 121-122.

责任编辑: 么丽羊