

## 摘要

本文论述了基于 DSP 的 MP3 随身听实时编码适配器系统的设计与开发技术。重点研究了 MPEG 音频第 III 层编码 的算法标准。详细分析了压缩算法中的四个主要功能模块：子带滤波器组编码，心理声学模型，比特流量化与霍夫曼编码，帧数据流格式化。比较了音频编码和语音编码之间的异同。在此基础上，结合实际应用，设计并调试了实现此功能的硬件系统。重点讨论了实现压缩算法的 DSP 芯片、完成控制功能的单片机、数据存储、音频 A/D 变换、LCD 显示、按键响应等系统外围设备的设计。实测结果表明，该系统达到设计要求，具有较好的性能价格比。

关键词： DSP MP3 MPEG 音频编码

## **ABSTRACT**

This paper deals with the design and development of a real-time encoding system for MP3 player based on DSP. The focus is placed on the investigation of the standard of the encoding algorithm for MPEG audio layer III, and the analysis of the major four modules in the compression algorithm, including encoding of subband filter bank, psychoacoustics model, quantification and Huffman coding, frame packing. In addition, a comparison is made between music coding and voice coding. On the basis of this, according to the actual application requirements, the hardware system implementing the encoding algorithm is designed and developed. Some key problems in the system development are discussed in detail, such as the DSP chips, the microcontroller, the scheme of data storage, the A/D converter, the LCD and the peripherals of system. The testing results show that the system agrees with the requirements and has a higher ratio of performance to cost.

**Keyword :** DSP MP3 MPEG audio coding

## 第一章 绪论

### 1.1 MPEG 音频编码发展状况

MPEG (Moving Pictures Experts Group) 是动态图象专家组织的英文缩写, 这个组织始建于 1988 年, 是致力于建立和制定视频和音频压缩标准的国际组织。该组织于 1992 年正式推出了具有 1.5Mbit/s 数据传输率的数字存储媒体运动图像及其伴音的 MPEG-I 的标准草案<sup>[1]</sup>, 即 ISO/IEC11172, 于 1993 年正式通过。该标准规定了数字音视频编码的国际标准, 主要包括三个方面: 系统、视频和音频。视频压缩仅仅包含画面信息, 音频压缩包含声音信息, 系统实现音频和视频的同步。所有播放 MPEG 视频压缩数据和音频压缩数据所需的时钟信息都包含在系统流中, 其中音频编码可以独立使用。

MPEG-I 音频编码标准支持单声道、双声道、立体声或联合立体声的两个声音通道的编码格式。由于该压缩算法可以把 CD 音质的两个通道共包含 1.4Mbit/s 的数据流压缩到 128kbit/s, 且仍然保持高保真的声音, 使其很快得到国际认可。随着多媒体技术和网络通信技术的进一步发展, ISO/IEC 在 1994 年又推出了 ISO/IEC13818, MPEG-II 运动图像及其伴音通用压缩标准<sup>[2]</sup>。该标准最初的目的是在 MPEG-I 兼容的基础上实现低比特率压缩和多声道扩展, 后来为了适应演播电视的要求, 开始致力于定义一个可以获得更高质量的多声道音频标准, 它可提供左、右、中及两个环绕声道、一个加重低音声道, 和多达 7 个伴音声道。同时可以将单声道编码比特率降低到 8kbit/s。

MPEG-I 和 MPEG-II 音频编码都提供了 3 个压缩层。层 I 是简单压缩, 它是一种听觉心理声学模型下的亚抽样编码。层 II 加入了更高的精度。层 III, 是现在流行的 MP3 音乐格式, 加入了非线性量化、霍夫曼编码和其它实现低速率高保真音质的先进技术, 它可以把一个 1.4Mbit/s 的立体声双通道数据流压缩为 32Kbit/s~384Kbit/s 且保持高保真的音质。依次下去的等级提供更高的质量和越来越高的压缩率, 但要求计算机有越来越强的压缩计算能力。典型数据为, 层 I 的目标是每个通道 192Kbit/s, 层 II 的目标是每个通道 128Kbit/s, 层 III 的目标是每个通道 64Kbit/s。层 II 要达到 64Kbit/s 时不如层 III 效果好, 而在 128Kbit/s, 层 II 和层 III 的效果接近, 而且都比层 I 效果好。

MP3 这种以压缩比和保真度出众的音频文件格式让许多个人用户青睐有加, 同时也让 IT 厂商看到了潜在的市场。作为 MP3 文件格式最大优势—文件小的体现, MP3 随身听已经作为一种时尚的标志出现在街头人们的挎包里、腰带上或胸前。作为全新的休闲娱乐产品, MP3 随身听生产厂商们也在揣摩着用户的心理,

## 第一章 绪论

### 1.1 MPEG 音频编码发展状况

MPEG (Moving Pictures Experts Group) 是动态图象专家组织的英文缩写, 这个组织始建于 1988 年, 是致力于建立和制定视频和音频压缩标准的国际组织。该组织于 1992 年正式推出了具有 1.5Mbit/s 数据传输率的数字存储媒体运动图像及其伴音的 MPEG-I 的标准草案<sup>[1]</sup>, 即 ISO/IEC11172, 于 1993 年正式通过。该标准规定了数字音视频编码的国际标准, 主要包括三个方面: 系统、视频和音频。视频压缩仅仅包含画面信息, 音频压缩包含声音信息, 系统实现音频和视频的同步。所有播放 MPEG 视频压缩数据和音频压缩数据所需的时钟信息都包含在系统流中, 其中音频编码可以独立使用。

MPEG-I 音频编码标准支持单声道、双声道、立体声或联合立体声的两个声音通道的编码格式。由于该压缩算法可以把 CD 音质的两个通道共包含 1.4Mbit/s 的数据流压缩到 128kbit/s, 且仍然保持高保真的声音, 使其很快得到国际认可。随着多媒体技术和网络通信技术的进一步发展, ISO/IEC 在 1994 年又推出了 ISO/IEC13818, MPEG-II 运动图像及其伴音通用压缩标准<sup>[2]</sup>。该标准最初的目的是在 MPEG-I 兼容的基础上实现低比特率压缩和多声道扩展, 后来为了适应演播电视的要求, 开始致力于定义一个可以获得更高质量的多声道音频标准, 它可提供左、右、中及两个环绕声道、一个加重低音声道, 和多达 7 个伴音声道。同时可以将单声道编码比特率降低到 8kbit/s。

MPEG-I 和 MPEG-II 音频编码都提供了 3 个压缩层。层 I 是简单压缩, 它是一种听觉心理声学模型下的亚抽样编码。层 II 加入了更高的精度。层 III, 是现在流行的 MP3 音乐格式, 加入了非线性量化、霍夫曼编码和其它实现低速率高保真音质的先进技术, 它可以把一个 1.4Mbit/s 的立体声双通道数据流压缩为 32Kbit/s~384Kbit/s 且保持高保真的音质。依次下去的等级提供更高的质量和越来越高的压缩率, 但要求计算机有越来越强的压缩计算能力。典型数据为, 层 I 的目标是每个通道 192Kbit/s, 层 II 的目标是每个通道 128Kbit/s, 层 III 的目标是每个通道 64Kbit/s。层 II 要达到 64Kbit/s 时不如层 III 效果好, 而在 128Kbit/s, 层 II 和层 III 的效果接近, 而且都比层 I 效果好。

MP3 这种以压缩比和保真度出众的音频文件格式让许多个人用户青睐有加, 同时也让 IT 厂商看到了潜在的市场。作为 MP3 文件格式最大优势—文件小的体现, MP3 随身听已经作为一种时尚的标志出现在街头人们的挎包里、腰带上或胸前。作为全新的休闲娱乐产品, MP3 随身听生产厂商们也在揣摩着用户的心理,

到底什么才是 MP3 随身听的卖点? 产品具有什么样的特色才能抓住用户的心?

目前, 世界上已经推出了几种 MP3 解码芯片, 如: ST 公司的 STA015、Micronas 公司的 MAS3507 等等。所以目前大多数型号的 MP3 随身听仅有播放功能, 但众多消费者希望能够一机多用, 比如可以实时记录语音数据。在某些 MP3 播放器整机产品中已经使用 G.726 标准的 ADPCM 语音编码算法来实现实时语音录音。它能够满足一般记录语音的质量要求, 但是对于要求质量较高的音乐信号就有些无能为力了。而使用 MP3 格式压缩编码技术除了可以获得较大的音乐数据压缩比, 还可以得到较高的音乐回放质量。但一直没有出现 MP3 硬件实时压缩器件, 同时 MP3 的硬件压缩又有较大市场需求量。正是基于此, 本文首先研究 MP3 编码算法, 然后利用 DSP 处理器开发 MP3 硬件编码适配器, 将模拟音频信号压缩为 MP3 格式, 可以在 MP3 机上播放。它不仅适用于对 CD 音质的音频编码, 同时通过降低采样频率, 可采用更低的 16Kbit/s 的编码比特率录制现场语音。

## 1.2 本文的主要工作和内容安排

本文的研究工作主要包含两部分内容。

第一部分是对 MPEG 音频第三层算法标准的研究。重点研究了编码的算法实现, 在实现过程中对音频编码 (44.1KHz 采样频率, 压缩到 64Kbit/s 每通道的编码速率) 主要采用 MPEG-I 的标准, 对于更低采样频率 (16KHz) 的语音编码则采用了 MPEG-II<sup>[3]</sup> 的标准。为了获得较高的压缩效果和高质量的音乐回放质量, 该标准采用了复杂的数学和心理学技术, 同时利用了对人耳灵敏度的研究结果。本文详细讨论了音频第 III 层编码算法的四个模块, 即子带滤波器组编码; 心理声学模型; 量化与编码和帧数据流格式化。

第二部分是基于 MPEG 音频第 III 层压缩算法的 MP3 实时编码适配器硬件系统实现。首先介绍了硬件系统的组成, 包括各功能模块的元器件的选择, 各功能模块之间的接口设计, 硬件电路板的设计等等。然后讨论了软件控制, 包括单片机对于按键的响应, 控制 LCD 的显示和对电源系统的管理等, 特别是单片机从编码器接收被压缩的数据流控制, 必须优化接收数据时序的算法, 以满足实时压缩的要求。

本文共分四章。第一章是绪论, 介绍了 MPEG 音频编码标准的发展状况以及本文的主要工作及内容。第二章详细探讨了 MPEG 音频第 III 层编码算法标准。第三章对实现其算法标准的硬件系统设计进行了具体分析与讨论。第四章对全文工作进行了总结。



到底什么才是 MP3 随身听的卖点? 产品具有什么样的特色才能抓住用户的心?

目前, 世界上已经推出了几种 MP3 解码芯片, 如: ST 公司的 STA015、Micronas 公司的 MAS3507 等等。所以目前大多数型号的 MP3 随身听仅有播放功能, 但众多消费者希望能够一机多用, 比如可以实时记录语音数据。在某些 MP3 播放器整机产品中已经使用 G.726 标准的 ADPCM 语音编码算法来实现实时语音录音。它能够满足一般记录语音的质量要求, 但是对于要求质量较高的音乐信号就有些无能为力了。而使用 MP3 格式压缩编码技术除了可以获得较大的音乐数据压缩比, 还可以得到较高的音乐回放质量。但一直没有出现 MP3 硬件实时压缩器件, 同时 MP3 的硬件压缩又有较大市场需求量。正是基于此, 本文首先研究 MP3 编码算法, 然后利用 DSP 处理器开发 MP3 硬件编码适配器, 将模拟音频信号压缩为 MP3 格式, 可以在 MP3 机上播放。它不仅适用于对 CD 音质的音频编码, 同时通过降低采样频率, 可采用更低的 16Kbit/s 的编码比特率录制现场语音。

## 1.2 本文的主要工作和内容安排

本文的研究工作主要包含两部分内容。

第一部分是对 MPEG 音频第三层算法标准的研究。重点研究了编码的算法实现, 在实现过程中对音频编码 (44.1KHz 采样频率, 压缩到 64Kbit/s 每通道的编码速率) 主要采用 MPEG-I 的标准, 对于更低采样频率 (16KHz) 的语音编码则采用了 MPEG-II<sup>[3]</sup> 的标准。为了获得较高的压缩效果和高质量的音乐回放质量, 该标准采用了复杂的数学和心理学技术, 同时利用了对人耳灵敏度的研究结果。本文详细讨论了音频第 III 层编码算法的四个模块, 即子带滤波器组编码; 心理声学模型; 量化与编码和帧数据流格式化。

第二部分是基于 MPEG 音频第 III 层压缩算法的 MP3 实时编码适配器硬件系统实现。首先介绍了硬件系统的组成, 包括各功能模块的元器件的选择, 各功能模块之间的接口设计, 硬件电路板的设计等等。然后讨论了软件控制, 包括单片机对于按键的响应, 控制 LCD 的显示和对电源系统的管理等, 特别是单片机从编码器接收被压缩的数据流控制, 必须优化接收数据时序的算法, 以满足实时压缩的要求。

本文共分四章。第一章是绪论, 介绍了 MPEG 音频编码标准的发展状况以及本文的主要工作及内容。第二章详细探讨了 MPEG 音频第 III 层编码算法标准。第三章对实现其算法标准的硬件系统设计进行了具体分析与讨论。第四章对全文工作进行了总结。

## 第二章 MPEG 音频编码算法标准

### 2.1 MPEG 音频编码算法标准概述

对于 MPEG-I, 音频压缩算法标准提供了以下模式:

- 1) 音频信号采样率可以是 32KHz, 44.1KHz 或 48KHz。
- 2) 压缩后的比特流可以按照单声道、独立的两个单声道、立体声、联合立体声等 4 种模式之一支持单声道或双声道编码。其中, 联合立体声模式是利用了立体声通道之间的关联和通道之间相位差的无关性得到的。
- 3) 压缩后的比特率可以从 32Kbit/s 到 224Kbit/s (每通道), 也可以使用用户自定义的比特率。
- 4) MPEG 音频编码标准提供了三个独立的压缩层次, 使用户可以在复杂性和压缩质量之间权衡选择。层 I 最简单, 最适于使用的比特率为 128Kbit/s (每通道) 以上。层 II 的复杂度中等, 使用的比特率为 128Kbit/s (每通道) 左右, 其应用包括: 数字广播, CD-ROM 上的声音信号以及 CD-I 和 VCD。层 III 即我们通常所说的 MP3 (MPEG-I (or II) Audio Layer III), 它最复杂, 但音质最佳, 比特率为 64Kbit/s (每通道) 左右。尤其适用 ISDN 上的声音传输, 本文将主要讨论这一层的编码算法。
- 5) 编码后的比特流支持 CRC 校验。
- 6) MPEG 音频编码标准还支持比特流中附带附加信息。

对于 MPEG-II, 增加了低采样频率的编码标准, 采样频率可以扩展到 16KHz、22.05KHz、24KHz。同时压缩后的比特率可以低到 8Kbit/s(每通道), 这有利于实现音质要求不高的现场语音录音。在本文所述的系统对于语音编码的实现, 采用 16KHz 的采样频率, 压缩到 16Kbit/s 的比特率。

#### 2.1.1 MPEG 编码器的基本结构

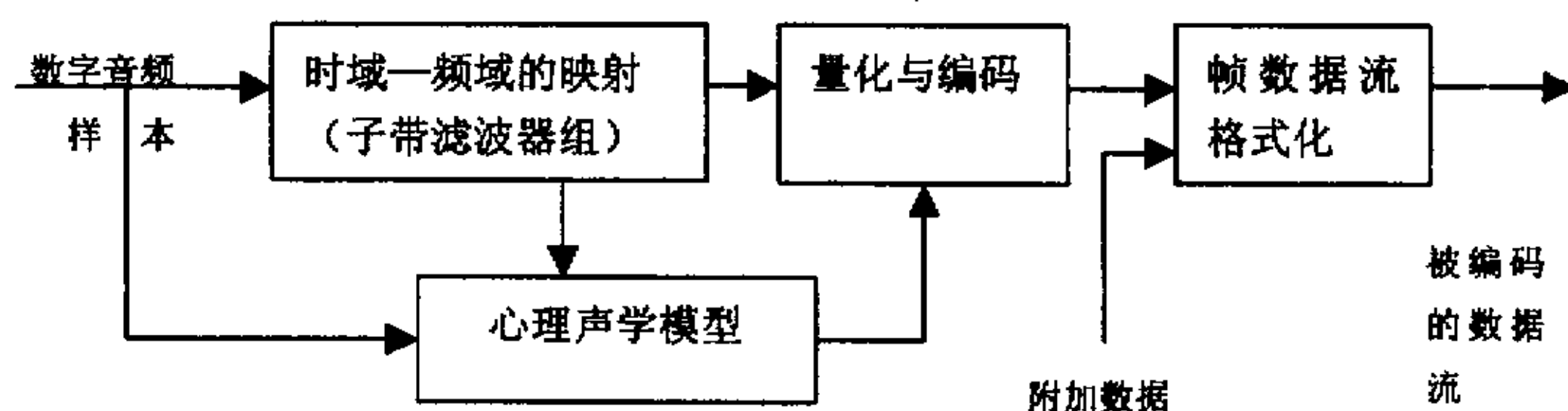


图 2.1 MPEG-I 音频编码器结构框图

## 第二章 MPEG 音频编码算法标准

### 2.1 MPEG 音频编码算法标准概述

对于 MPEG-I, 音频压缩算法标准提供了以下模式:

- 1) 音频信号采样率可以是 32KHz, 44.1KHz 或 48KHz。
- 2) 压缩后的比特流可以按照单声道、独立的两个单声道、立体声、联合立体声等 4 种模式之一支持单声道或双声道编码。其中, 联合立体声模式是利用了立体声通道之间的关联和通道之间相位差的无关性得到的。
- 3) 压缩后的比特率可以从 32Kbit/s 到 224Kbit/s (每通道), 也可以使用用户自定义的比特率。
- 4) MPEG 音频编码标准提供了三个独立的压缩层次, 使用户可以在复杂性和压缩质量之间权衡选择。层 I 最简单, 最适于使用的比特率为 128Kbit/s (每通道) 以上。层 II 的复杂度中等, 使用的比特率为 128Kbit/s (每通道) 左右, 其应用包括: 数字广播, CD-ROM 上的声音信号以及 CD-I 和 VCD。层 III 即我们通常所说的 MP3 (MPEG-I (or II) Audio Layer III), 它最复杂, 但音质最佳, 比特率为 64Kbit/s (每通道) 左右。尤其适用 ISDN 上的声音传输, 本文将主要讨论这一层的编码算法。
- 5) 编码后的比特流支持 CRC 校验。
- 6) MPEG 音频编码标准还支持比特流中附带附加信息。

对于 MPEG-II, 增加了低采样频率的编码标准, 采样频率可以扩展到 16KHz、22.05KHz、24KHz。同时压缩后的比特率可以低到 8Kbit/s(每通道), 这有利于实现音质要求不高的现场语音录音。在本文所述的系统对于语音编码的实现, 采用 16KHz 的采样频率, 压缩到 16Kbit/s 的比特率。

#### 2.1.1 MPEG 编码器的基本结构

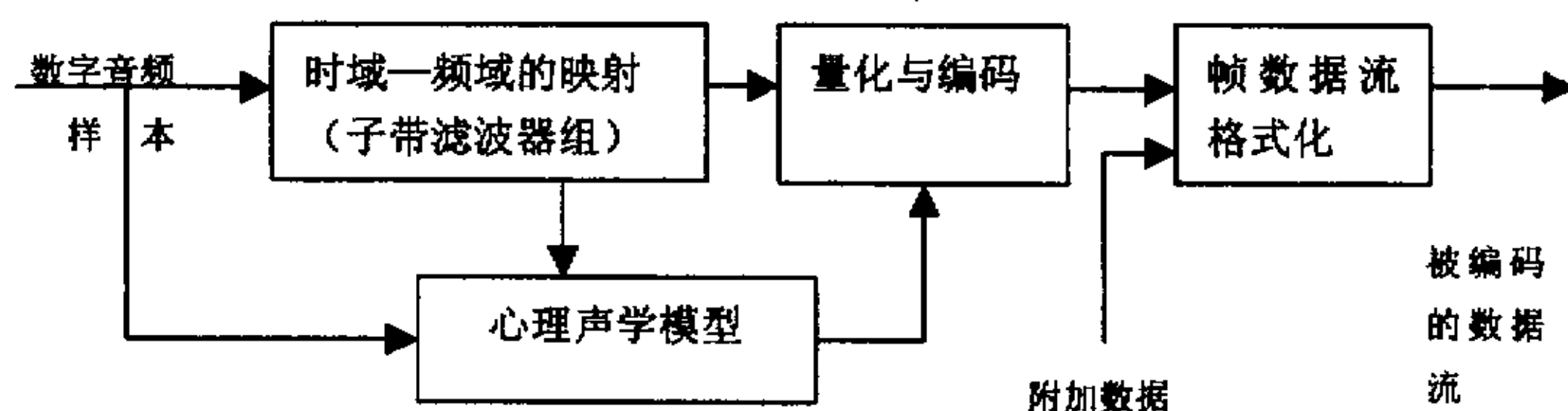


图 2.1 MPEG-I 音频编码器结构框图



以下将主要介绍 MPEG-I 编码器算法标准。如图 2.1 所示, 编码时输入的数字声音脉冲信号经过一个滤波器组 (Filter Bank) 被分到频域里的多个子带里, 输入声音信号同时要经过一个心理声学模型, 以确定每个子带里信号能量与掩蔽阈值的比率, 即信掩比  $SMR(\text{Signal-to-Mask Ratio})^{[4]}$ 。量化与编码部分用信掩比  $SMR$  来决定分配给子带信号的量化位数, 在每个子带中, 超过信掩比率的量化噪声将不被编码。最后将量化的子带样本和格式标记及其它数据封装成比特流存储。解码器对该比特流解码, 恢复被量化的子带样本值以重建声音信号。

### 2.1.2 MPEG 音频压缩标准定义的 3 个层次

#### 1) 层 I

层 I 按帧进行编码, 除了声音数据以外, 每帧还包括有一个头部, 一个可选的 CRC 校验码以及附加数据, 表 2.1 示意了层 I 比特流数据的安排。一帧数据有 384 个样本组成, 经过子带滤波器产生 32 个子带, 每个子带 12 个样本, 如图 2.2 所示。每 12 个子带样本组成一组, 进行一次比特分配并纪录一个缩放因子。缩放因子的计算过程如下: 找到 12 个子带样本的最大值, 在缩放因子表中从最小的值开始, 第一个大于这个最大值的数就作为该子带的缩放因子。比特分配信息告诉解码器每个样本由几位表示。缩放因子索引由 6 比特组成, 解码器使用这个 6 比特的缩放因子索引找到相应的缩放因子, 用它乘以解码出来的样本值, 以恢复被量化的子带值。缩放因子充分利用了量化器的量化范围, 通过比特分配和缩放因子相配合, 可以表示动态范围超过 120dB 的样本。

表 2-1 层 I 比特流数据构成

头部	CRC	比特分配信息	缩放因子索引	样本	附加数据
----	-----	--------	--------	----	------

#### 2) 层 II

层 II 对层 I 的算法做了直观的改进。它编码每帧的样本数更大 (1152 个/每通道), 并且在中高频区对比特分配做了一些限制。它对比特分配、缩放因子和量化样本值的编码也更紧凑。因为层 II 在比特分配、缩放因子上需要的比特数少了, 所以可以有更多的比特来对声音数据进行编码, 因此音质较层 I 有所提高。与层 I 对每个子带由 12 个样本组成一组的编码不同, 层 II 对一个子带分成三个组进行编码, 其中每组 12 个样本, 如图 2.2 所示。当有可能出现声音失真时, 编码器对这 3 组样本使用 3 个不同的缩放因子。在其它情况下, 层 II 编码器让各个组共享一个缩放因子。层 II 比特流中有一个缩放因子选择域 SCFSI, 解码器根据这个域的信息判断是否需要以及如何共享缩放因子。表 2.2 列出了层 II 的数据流格式。

以下将主要介绍 MPEG-I 编码器算法标准。如图 2.1 所示, 编码时输入的数字声音脉冲信号经过一个滤波器组 (Filter Bank) 被分到频域里的多个子带里, 输入声音信号同时要经过一个心理声学模型, 以确定每个子带里信号能量与掩蔽阈值的比率, 即信掩比  $SMR(\text{Signal-to-Mask Ratio})^{[4]}$ 。量化与编码部分用信掩比  $SMR$  来决定分配给子带信号的量化位数, 在每个子带中, 超过信掩比率的量化噪声将不被编码。最后将量化的子带样本和格式标记及其它数据封装成比特流存储。解码器对该比特流解码, 恢复被量化的子带样本值以重建声音信号。

### 2.1.2 MPEG 音频压缩标准定义的 3 个层次

#### 1) 层 I

层 I 按帧进行编码, 除了声音数据以外, 每帧还包括有一个头部, 一个可选的 CRC 校验码以及附加数据, 表 2.1 示意了层 I 比特流数据的安排。一帧数据有 384 个样本组成, 经过子带滤波器产生 32 个子带, 每个子带 12 个样本, 如图 2.2 所示。每 12 个子带样本组成一组, 进行一次比特分配并纪录一个缩放因子。缩放因子的计算过程如下: 找到 12 个子带样本的最大值, 在缩放因子表中从最小的值开始, 第一个大于这个最大值的数就作为该子带的缩放因子。比特分配信息告诉解码器每个样本由几位表示。缩放因子索引由 6 比特组成, 解码器使用这个 6 比特的缩放因子索引找到相应的缩放因子, 用它乘以解码出来的样本值, 以恢复被量化的子带值。缩放因子充分利用了量化器的量化范围, 通过比特分配和缩放因子相配合, 可以表示动态范围超过 120dB 的样本。

表 2-1 层 I 比特流数据构成

头部	CRC	比特分配信息	缩放因子索引	样本	附加数据
----	-----	--------	--------	----	------

#### 2) 层 II

层 II 对层 I 的算法做了直观的改进。它编码每帧的样本数更大 (1152 个/每通道), 并且在中高频区对比特分配做了一些限制。它对比特分配、缩放因子和量化样本值的编码也更紧凑。因为层 II 在比特分配、缩放因子上需要的比特数少了, 所以可以有更多的比特来对声音数据进行编码, 因此音质较层 I 有所提高。与层 I 对每个子带由 12 个样本组成一组的编码不同, 层 II 对一个子带分成三个组进行编码, 其中每组 12 个样本, 如图 2.2 所示。当有可能出现声音失真时, 编码器对这 3 组样本使用 3 个不同的缩放因子。在其它情况下, 层 II 编码器让各个组共享一个缩放因子。层 II 比特流中有一个缩放因子选择域 SCFSI, 解码器根据这个域的信息判断是否需要以及如何共享缩放因子。表 2.2 列出了层 II 的数据流格式。

表 2.2 层 II 比特流数据构成

头部	CRC	比特分配信息	SCFSI	缩放因子信息	样本	附加数据
----	-----	--------	-------	--------	----	------

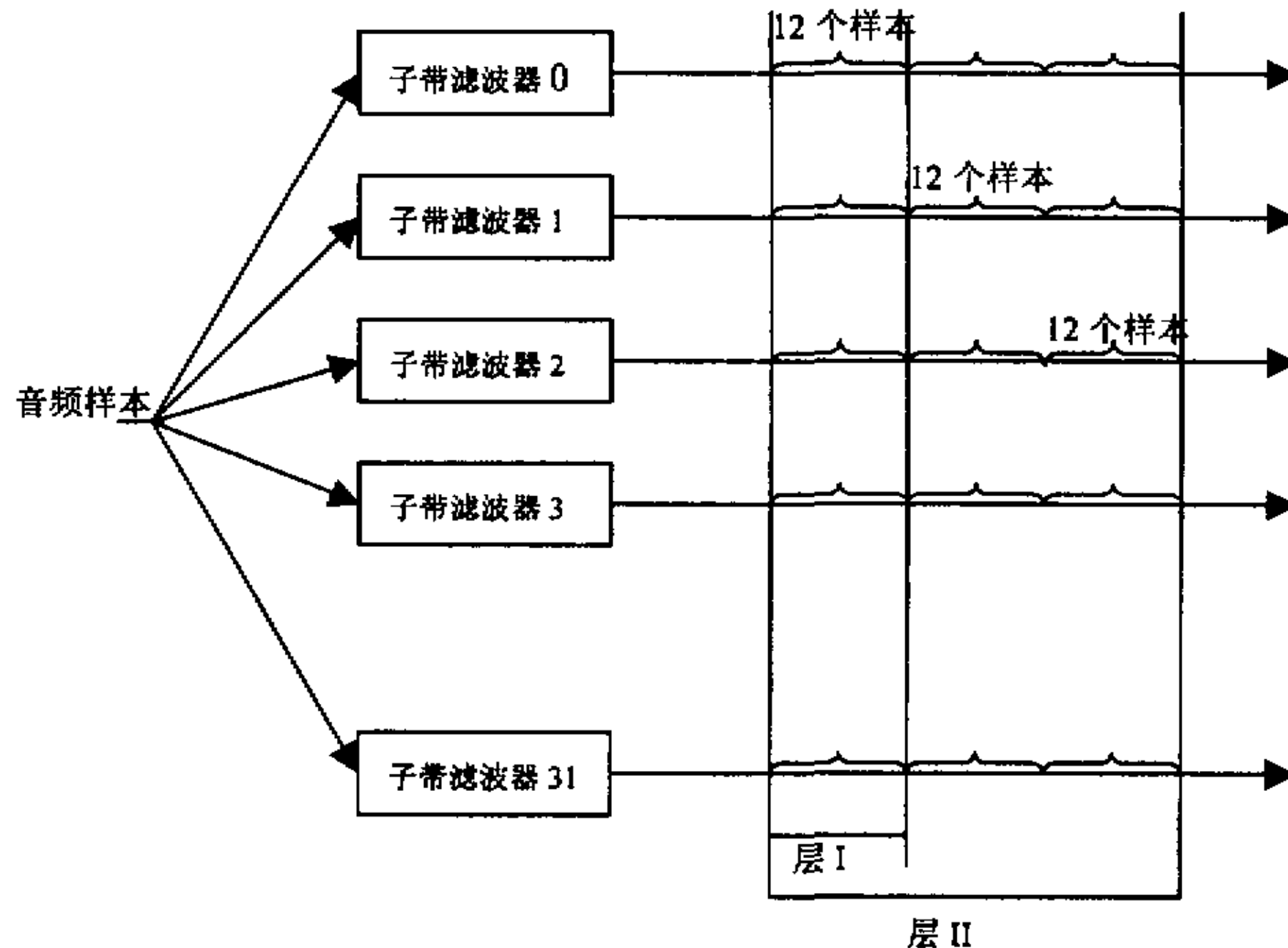


图 2.2 层 I 和层 II 子带样本的形成示意图

### 3) 层 III

层 III 使用了更为精细的算法，它是从 ASPEC(Audio Spectral Perceptual Entropy Encoding)和 OCF(Optimal Coding in the Frequency Domain)<sup>[6]</sup>演变而来的。虽然层 III 所用的滤波器组与层 II 所用的相同，但是层 III 使用改进离散余弦变换 MDCT 对滤波器组的不足做了一些补偿。与子带滤波器组不同，MDCT 是没有失真的。MDCT 把子带的输出在频域里进一步地细分。它指定了两种 MDCT 块长：长块的长度为 18 个样本，短块的长度为 6 个样本。相邻变换窗之间有 50% 的重叠，所以窗口的大小分别为 36 个样本和 12 个样本。长块对于平稳声音信号可以得到更高的频域分辨率，而短块的切换可以得到更高的时域分辨率。短块的长度是长块的 1/3，三个短块可以代替一个长块，所以 MDCT 的样本数不受分块长度的影响。对于给定的一帧信号，MDCT 可以全部使用长块或全部使用短块，也可以长短块混合使用。由于低频区的频域分辨率对音质有重大的影响，所以在混合块长的模式下，MDCT 对最低频的 2 个子带使用长块，而对其余的 30 个子带使用短块。这样，既能保证低频区的频域分辨率，又不会牺牲高频区的时域分辨率。层 III 编码器已经部分消除了子带滤波器组引入的样本混叠。除了使用 MDCT 外，层 III 还采用了许多其它改进措施来提高压缩比而不降低音质。而且，层 III 的计

算量并没有比层 II 增加许多, 增加的主要是编码器的复杂度和编码器所需要的存储容量。下面将详细讨论第 III 层音频编码的四个主要组成部分: 子带滤波器组编码、心理声学模型、量化与编码和帧数据流格式化。

## 2.2 子带滤波器组编码

子带滤波器组编码完成样本信号从时域到频域的映射, 首先介绍有关变换域编码的一些基本概念。变换域编码主要由映射、量化和编码组成。映射是将原始信号中的各个样值从一个域变换到另外一个域, 然后对映射后的数据进行量化与编码。映射的关键在于能够产生一系列更加有效的系数, 对这些系数进行编码所需的总比特数要比对原始数据进行编码所需的总比特数少的多, 从而使数据率得以压缩。映射变换的方法很多, 常用的是正交变换法来去除样本的相关性, 其实现数据压缩的本质在于经过多维坐标系中适当的旋转和变换, 能够把散布在各个坐标轴上的原始数据在新的、适当的坐标系中集中到少数坐标轴上, 因此可能用较少的编码维数来表示一组信号样本, 实现高效率的压缩编码。

另外, 在变换域中选择哪些系数进行量化编码, 对压缩性能有很大的影响。原则上应保留能量集中、方差大的系数, 主要分为区域取样和阈值取样两种方式。后一种方式即事先设定一阈值, 如果某系数方差或幅值在该阈值允许的范围内, 则保留并编码传输, 否则就舍弃, MPEG 标准就是采用了这种方式。

### 2.2.1 子带滤波器组编码的定义与特点

所谓子带编码(SBC, subband coding)就是利用带通滤波器(BPF)组把信号频带分割成若干个子频带, 通过等效于单边带调幅的调制过程, 将各子带搬移到零频率附近以得到低通表示, 然后再以奈奎斯特频率对各子带输出取样, 并对取样值进行通常的数字编码。

把音频信号分成子带后进行编码的特点:

- 1) 由于声音频谱的非平坦性, 如果对不同子带合理地分配比特数, 就有可能分别控制各子带的量化电平数目以及相应的重建误差的方差, 使码流更精确地与各子带的信源统计特性相匹配。例如语音信号, 低频带的基音与共振峰要求保存较高的精度, 可以用较高的比特数来表示样值, 而发生在高频带的摩擦音及类噪声样值可以分配较少的编码比特。

- 2) 各子带的量化噪声都束缚在本子带内, 这样就能避免能量较小频带内的输入信号被其它频段的量化噪声所掩盖。



### 2.2.2 子带滤波器组算法

实质上,该算法是将规定格式的数字音响信号(若其采样频率是 44.1KHz,每样本量化为 16 位,则其编码比特率为  $16 \times 44100 \text{ bit/s}$ ,若是立体声,则码速率应加倍,其值为 1.411Mbps,即 CD 格式音乐的比特率)分解成 32 个子带输出。由 32 个 FIR 型 BPF 组成,每个 BPF 的长度为 512 点。因此,实际上是用一个  $32 \times 512$  的变换矩阵  $C$  将输入序列  $\{X_i\}$  中的连续 512 个数据点所构成的输入矢量  $X$ ,变换成一个具有 32 个分量的输出矢量  $S$ 。其中,  $S$  的 32 个数据点  $S_k$  按子带频率从低到高的顺序排列。子带分析滤波器组算法包括以下步骤:(流程图见附图 A.1)

步骤 1: 输入 32 个音频样本;

步骤 2: 构造一个具有 512 个元素的矢量空间  $X$ , 将这 32 个样本移至矢量下表 0—31 处,最新进来的在位置 0,最先进来的 32 个元素移出去;

步骤 3: 用矢量  $C$  对  $X$  加窗,得到积矢量  $Z$ :

$$Z_i = C_i * X_i, \quad i = 0, 1, \dots, 511 \quad \text{式 (2-1)}$$

其中,  $C_i$  的值在专门的表<sup>[1]</sup>中给出;

步骤 4: 计算:

$$Y_i = \sum_{j=0}^7 Z_{i+64j}, \quad i = 0, 1, \dots, 63 \quad \text{式 (2-2)}$$

步骤 5: 通过下面矩阵运算得到 32 个子带的输出样本:

$$S_i = \sum M_{ik} \cdot Y_k, \quad i = 0, 1, \dots, 31 \quad \text{式 (2-3)}$$

其中,  $M_{ik} = \cos [(2i+1)(k-16)\pi/64]$   $i = 0, 1, \dots, 31; k = 0, 1, \dots, 63$ 。

滤波器组完成了输入样本从时域到频域的映射,由于标准提供了特定的时频映射变换矩阵,所以计算得到的频域样本数与时域样本数相同。与层 I、层 II 不同的是,在第 III 层编码中采用了分节处理的方法,即将每一帧数据分为两节来处理,每节的样本数为 576,两节数据的处理方法基本相同,都是将他们分成 32 个子带分别量化。而且,在 ISO/IEC11172-3 标准中,第 III 层编码滤波器组的频域分辨率与信号有关,可以有  $6 \times 32$  或  $18 \times 32$  个频率样本。如果采用  $6 \times 32$  个样本,则每一子带分为三组且样本被分别量化。这样,总的样本数仍是 576/节,一帧样本数仍是 1152/通道。



## 2.3 心理声学模型

### 2.3.1 心理声学模型基本概念

#### 1) 掩蔽效应

一个声音的听觉感受受到另一个声音影响的现象叫做“掩蔽效应”。前者称为被掩蔽音，后者称为掩蔽音。被掩蔽音单独存在时的听阈分贝值称为绝对听阈。在掩蔽情况下，必须加大被掩蔽音的强度才能被人耳听见，此时的听阈称为掩蔽听阈。

#### 2) 绝对听阈

图 2.3 绘出了人耳的绝对听阈曲线，人耳对不同频率的敏感程度差别很大。其中，对 2KHz 到 4KHz 范围内的信号最为敏感，幅度很低的信号都能被人耳听到。而在低频区和高频区，能被人耳听到的信号幅度要高的多。

掩蔽阈值 (dB)

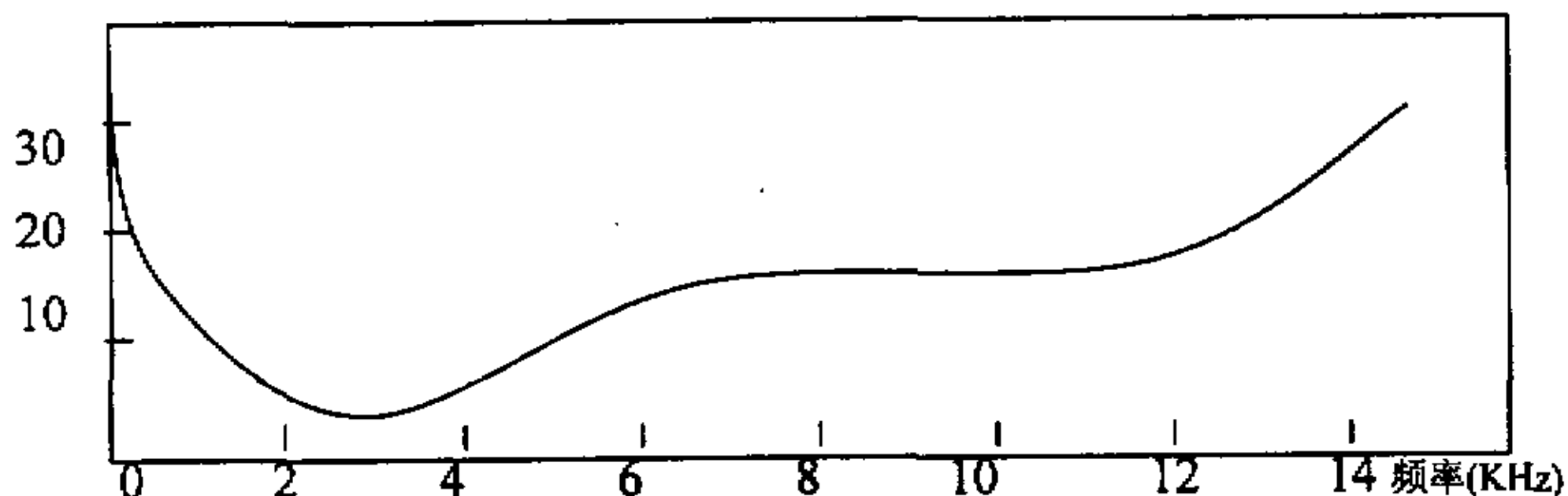


图 2.3 人耳的绝对听阈曲线

#### 3) 掩蔽听阈

掩蔽听阈分为频域掩蔽和时域掩蔽。

频域掩蔽：频域中的一个强音会掩蔽与之同时发声的附近的弱音。它有如下两个规律：一，低频纯音可以有效地掩蔽高频纯音，但高频纯音对低频纯音的掩蔽作用则很小。二，在距离强音较远处，绝对听阈比该强音所引起的掩蔽阈值更高。此时，噪声的掩蔽阈值应取绝对听阈。

例如，由低到高排列的三个连续频段的量化值分别是 10dB、60dB 和 35dB，其中，中间频段对两边频段的掩蔽效应是 12dB 和 15dB，由于低频段的信号量化值 10dB 小于 12dB，被掩盖，所以无需编码；对于高频区的量化值大于 12dB，所以必须编码。

时域掩蔽：除了同时发出的声音之间有掩蔽现象以外，在时间上相邻的音之间也有掩蔽现象。我们称之为时域掩蔽。产生时域掩蔽的原因在于人的大脑处理信息需要花费一定的时间。

## 2.3 心理声学模型

### 2.3.1 心理声学模型基本概念

#### 1) 掩蔽效应

一个声音的听觉感受受到另一个声音影响的现象叫做“掩蔽效应”。前者称为被掩蔽音，后者称为掩蔽音。被掩蔽音单独存在时的听阈分贝值称为绝对听阈。在掩蔽情况下，必须加大被掩蔽音的强度才能被人耳听见，此时的听阈称为掩蔽听阈。

#### 2) 绝对听阈

图 2.3 绘出了人耳的绝对听阈曲线，人耳对不同频率的敏感程度差别很大。其中，对 2KHz 到 4KHz 范围内的信号最为敏感，幅度很低的信号都能被人耳听到。而在低频区和高频区，能被人耳听到的信号幅度要高的多。

掩蔽阈值 (dB)

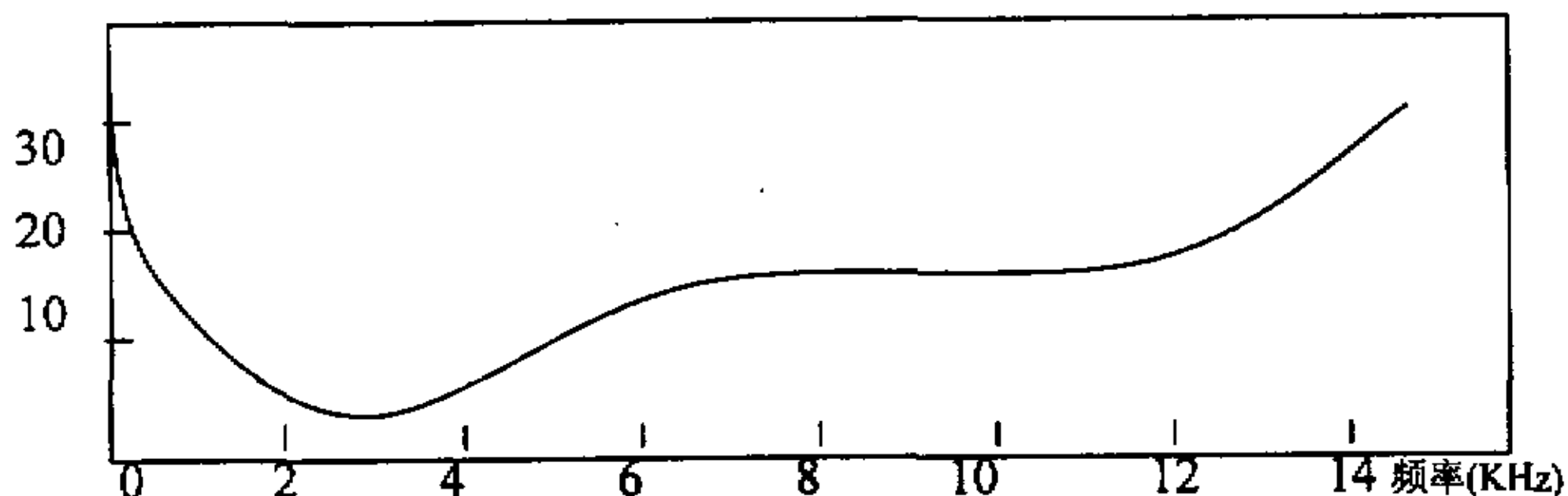


图 2.3 人耳的绝对听阈曲线

#### 3) 掩蔽听阈

掩蔽听阈分为频域掩蔽和时域掩蔽。

频域掩蔽：频域中的一个强音会掩蔽与之同时发声的附近的弱音。它有如下两个规律：一，低频纯音可以有效地掩蔽高频纯音，但高频纯音对低频纯音的掩蔽作用则很小。二，在距离强音较远处，绝对听阈比该强音所引起的掩蔽阈值更高。此时，噪声的掩蔽阈值应取绝对听阈。

例如，由低到高排列的三个连续频段的量化值分别是 10dB、60dB 和 35dB，其中，中间频段对两边频段的掩蔽效应是 12dB 和 15dB，由于低频段的信号量化值 10dB 小于 12dB，被掩盖，所以无需编码；对于高频区的量化值大于 12dB，所以必须编码。

时域掩蔽：除了同时发出的声音之间有掩蔽现象以外，在时间上相邻的音之间也有掩蔽现象。我们称之为时域掩蔽。产生时域掩蔽的原因在于人的大脑处理信息需要花费一定的时间。

4) 巴克 (*bark*)

频域计量单位。是音频范围内频率尺度的非线性映像, 基本符合人耳的频率选择性。

$$1 \text{ bark} \approx \begin{cases} frequency/100 & frequency < 500 \text{ Hz} \\ 9 + 4\log(frequency/1000) & frequency > 500 \text{ Hz} \end{cases} \quad \text{式 (2.4)}$$

心理声学模型是为了获得滤波器组每一子带内刚好可察觉的噪声级, 即它的掩蔽阈值。对于子带滤波器组输出的谱值量化后产生的噪声, 如果能够被控制在掩蔽阈值以下, 则最终的压缩数据被解码后的结果与原始信号可以不加区分。一个给定信号的掩蔽能力取决于它的频率和响度, 所以心理声学模型的最终输出是信掩比 SMR(signal-to-mask ratio), 即信号强度与掩蔽阈值的比率。其中, 阈值是以频率为自变量计算得到的。MPEG 音频标准提供了两种心理声学模型, 模型 1 比模型 2 简单, 两个模型对所有层次都适用。实际上, 对于这两种可供选择的模型, MPEG-I 建议第 I、II 层采用模型 1 计算, 而第 III 层则用模型 2。只是第 I、II 层输出的是每一子带的信掩比, 而第 III 层输出并不是面向 32 个子带的信掩比, 它是基于对子带滤波器组输出的频谱值重新分割后的信掩比, 即将时频映射后输出的谱值重新分割, 把它们分为若干个阈值分区, 从而得到每个分区的信掩比。这里只介绍模型 2 的基本算法流程。

阈值分区的信掩比计算需要同时考虑以下三个输入量:

- 1) 信掩比计算过程中 FFT 的移位长度 *iblen*, 其值  $384 < iblen < 640$ , 在整个计算过程中, *iblen* 必须保持恒量。在第 III 层中, 计算分区阈值需要两个不同的移位长度, 但每一次必须以恒定的移位长度计算。
- 2) 最新的 *iblen* 样本信号。
- 3) 采样率。必须保持恒量, 对于音频编码, 可选项有 48kHz, 44.1kHz, 32kHz。

### 2.3.2 心理声学模型 2 的标准算法流程

- 1) 重构 1024 个样本作为 FFT 的输入信号。

上文提到第 III 层编码中, 将一帧的数据分为两节来处理, 对于每节样本阈值分区的信掩比的计算, 都需要添加 *iblen* 新样本, 且必须存储 1024-*iblen* 旧样本。该 1024 个连续样本构成输入信号  $S_i$ , 其中,  $1 \leq i \leq 1024$ 。

- 2) 计算输入信号的复频谱。

首先给输入信号  $S_i$  加 1024 点的汉宁窗:

$$SW_i = S_i * (0.5 - 0.5 \cos(2\pi * (i - 0.5) / 1024)) \quad 1 \leq i \leq 1024 \quad \text{式 (2-5)}$$

然后对  $SW_i$  进行 FFT 计算, FFT 与子带滤波器组操作并行进行, 以弥补子带

4) 巴克 (*bark*)

频域计量单位。是音频范围内频率尺度的非线性映像,基本符合人耳的频率选择性。

$$1 \text{ bark} \approx \begin{cases} frequency/100 & frequency < 500 \text{ Hz} \\ 9 + 4\log(frequency/1000) & frequency > 500 \text{ Hz} \end{cases} \quad \text{式 (2.4)}$$

心理声学模型是为了获得滤波器组每一子带内刚好可察觉的噪声级,即它的掩蔽阈值。对于子带滤波器组输出的谱值量化后产生的噪声,如果能够被控制在掩蔽阈值以下,则最终的压缩数据被解码后的结果与原始信号可以不加区分。一个给定信号的掩蔽能力取决于它的频率和响度,所以心理声学模型的最终输出是信掩比 SMR(signal-to-mask ratio),即信号强度与掩蔽阈值的比率。其中,阈值是以频率为自变量计算得到的。MPEG 音频标准提供了两种心理声学模型,模型 1 比模型 2 简单,两个模型对所有层次都适用。实际上,对于这两种可供选择的模型,MPEG-I 建议第 I、II 层采用模型 1 计算,而第 III 层则用模型 2。只是第 I、II 层输出的是每一子带的信掩比,而第 III 层输出并不是面向 32 个子带的信掩比,它是基于对子带滤波器组输出的频谱值重新分割后的信掩比,即将时频映射后输出的谱值重新分割,把它们分为若干个阈值分区,从而得到每个分区的信掩比。这里只介绍模型 2 的基本算法流程。

阈值分区的信掩比计算需要同时考虑以下三个输入量:

- 1) 信掩比计算过程中 FFT 的移位长度 *iblen*, 其值  $384 < iblen < 640$ , 在整个计算过程中, *iblen* 必须保持恒量。在第 III 层中,计算分区阈值需要两个不同的移位长度,但每一次必须以恒定的移位长度计算。
- 2) 最新的 *iblen* 样本信号。
- 3) 采样率。必须保持恒量,对于音频编码,可选项有 48kHz, 44.1kHz, 32kHz。

## 2.3.2 心理声学模型 2 的标准算法流程

- 1) 重构 1024 个样本作为 FFT 的输入信号。

上文提到第 III 层编码中,将一帧的数据分为两节来处理,对于每节样本阈值分区的信掩比的计算,都需要添加 *iblen* 新样本,且必须存储 1024-*iblen* 旧样本。该 1024 个连续样本构成输入信号  $S_i$ , 其中,  $1 \leq i \leq 1024$ 。

- 2) 计算输入信号的复频谱。

首先给输入信号  $S_i$  加 1024 点的汉宁窗:

$$SW_i = S_i * (0.5 - 0.5 \cos(2\pi * (i - 0.5) / 1024)) \quad 1 \leq i \leq 1024 \quad \text{式 (2-5)}$$

然后对  $SW_i$  进行 FFT 计算,FFT 与子带滤波器组操作并行进行,以弥补子带

滤波器组缺乏低频选择性的不足。因为子带滤波器组具有最小预回声优化窗, 所以它可对音频编码提供足够的时域分辨率。同时, FFT 可以提供信掩比计算所需的谱分辨率。 $R_\omega$ 、 $F_\omega$  分别代表  $SWI$  变换后的幅度和相位, 由于考虑到时域掩蔽效应的影响, 它们的值是由前两节样本的频谱矢量迭代计算得到的, 计算过程如下:

$$\hat{R}_\omega = 2.0R_\omega(t-1) - R_\omega(t-2) \quad \text{式 (2-6)}$$

$$\hat{F}_\omega = 2.0F_\omega(t-1) - F_\omega(t-2) \quad \text{式 (2-7)}$$

其中,  $\hat{R}_\omega$ 、 $\hat{F}_\omega$  表示当前节的数值。

### 3) 计算预测性参数 $C_\omega$

掩蔽阈值的计算不仅与频域掩蔽有关, 而且受到时域掩蔽的影响, 所以需要由其它节的频域样本预测下一节样本数据的频谱分布, 从而降低时域掩蔽的影响。通过计算  $C_\omega$  来完成上述要求。

$$C_\omega = \frac{((R_\omega \cos F_\omega - \hat{R}_\omega \cos \hat{F}_\omega)^2 + (R_\omega \sin F_\omega - \hat{R}_\omega \sin \hat{F}_\omega)^2)^{0.5}}{R_\omega + \text{abs}(\hat{R}_\omega)} \quad \text{式 (2-8)}$$

### 4) 计算每个阈值分区的能量 $e_b$ :

$$e_b = \sum_{\omega=\omega_{lowb}}^{\omega_{highb}} R_\omega^2 \quad \text{式 (2-9)}$$

然后根据预测性参数  $C_\omega$  的值, 计算每个阈值分区的加权能量:

$$c_b = \sum_{\omega=\omega_{lowb}}^{\omega_{highb}} R_\omega^2 * C_\omega \quad \text{式 (2-10)}$$

在低频区, 每条 FFT 谱线为一个阈值分区, 在高频区, 每个阈值分区则由多条谱线组成, 式 2-9、2-10 中的  $b$  表示分区的序列号,  $\omega_{lowb}$  表示每一分区的最低谱线,  $\omega_{highb}$  表示每一分区的最高谱线。

5) 在整个频域内, 分别对阈值分区的能量和加权能量通过频率展开函数  $\text{sprdang}(x, y)$  卷积求和:

$$ecb_b = \sum_{bb=1}^{b_{\max}} e_{bb} * \text{sprdang}(bval_{bb}, bval_b) \quad \text{式 (2-11)}$$



$$ct_b = \sum_{bb=1}^{b_{\max}} c_{bb} * sprdang(bval_{bb}, bval_b) \quad \text{式 (2-12)}$$

MPEG 标准规定, 频率展开函数的计算如下:

$$tmp1 = 1.05(y - x) \quad \text{式 (2-13)}$$

$$tmp2 = 8 \min((tmp1 - 0.5)^2 - 2(tmp1 - 0.5), 0) \quad \text{式 (2-14)}$$

$$sprdngf(x, y) = \begin{cases} 0 & tmp1 < -100 \\ 10^{\frac{tmp1 + tmp2}{10}} & \text{其它} \end{cases} \quad \text{式 (2-15)}$$

其中,  $tmp1$ 、 $tmp2$  是临时变量,  $x$  ( $bval_{bb}$ )、 $y$  ( $bval_b$ ) 是不同阈值分区信号的临界巴克值, 它表示每个分区的平均谱值, MPEG 标准规定了所有阈值分区的临界巴克值。

由于  $ct_b$  是用信号能量加权, 所以必须被重新归一化到  $cb_b$ :

$$cb_b = \frac{ct_b}{ecb_b} \quad \text{式 (2-16)}$$

同时, 由于频率展开函数的非归一化性, 所以  $ecb_b$  也应当重新归一化, 且归一化能量  $en_b$  为:

$$en_b = ecb_b * rnorm_b \quad \text{式 (2-17)}$$

归一化系数  $rnorm_b$ :

$$rnorm_b = 1 / \sum_{bb=1}^{b_{\max}} sprdang(bval_{bb}, bval_b) \quad \text{式 (2-18)}$$

6) 将  $cb_b$  转换为音调指数  $tb_b$ :

$$tb_b = 0.299 - 0.43 \log_e cb_b \quad 0 < tb_b < 1 \quad \text{式 (2-19)}$$

7) 计算每个分区所要求的信噪比  $SNR_b$ :

$$SNR_b = \max(\minval_b, tb_b * TMN_b + (1 - tb_b) * NMT_b) \quad \text{式 (2-20)}$$

信号与噪声在不同频率段内相互影响的程度不同, 所以把每个阈值分区允许的最小巴克值与之进行比较, 从而得到每个分区的信噪比。其中,  $\minval_b$  是每个阈值分区最小的巴克值,  $TMN_b$  是信号掩盖噪声的巴克值,  $NMT_b$  是噪声掩盖信号的巴克值, 它们的值在专门的表<sup>[1]</sup>中给出。

8) 计算功率的信噪比:

$$bc_b = 10^{\frac{-SNR_b}{10}} \quad \text{式 (2-21)}$$

9) 实际的能量阈值计算公式:

$$nb_b = en_b * bc_b \quad \text{式 (2-22)}$$

把阈值能量分配到所有 FFT 谱线上, 得到每一条谱线的阈值能量:

$$nb_\omega = nb_b / (\omega_{highb} - \omega_{lowb} + 1) \quad \text{式 (2-23)}$$

加上绝对阈值 (在专门的表中给出) 就可以得到最终的音频能量阈值  $thr_\omega$ :

$$thr_\omega = \max(nb_\omega, absthr_\omega) \quad \text{式 (2-24)}$$

10) 计算  $SMR_n$ :

最后将所有的 FFT 谱线重新分割成  $n$  个区域, 计算每个区域的信掩比。参数  $width_n = 1$  时表示心理模型的窄带型区域,  $width_n = 0$  表示宽带型区域。对于每个区域的能量  $epart_n$  计算如下:

$$epart_n = \sum_{\omega=\omega_{low_n}}^{\omega_{high_n}} R_\omega^2 \quad \text{式 (2-25)}$$

如果  $width_n = 1$ , 则每个区域的阈值  $npart_n$  为:

$$npart_n = \sum_{\omega=\omega_{low_n}}^{\omega_{high_n}} thr_\omega \quad \text{式 (2-26)}$$

如果  $width_n = 0$ , 则:

$$npart_n = \min(thr_{\omega_{lown}} \cdots thr_{\omega_{highn}}) * (\omega_{highn} - \omega_{lown} + 1) \quad \text{式(2-27)}$$

$$SMR_n = 10 \log_{10} \left( \frac{epart_n}{npart_n} \right) \quad \text{式(2-28)}$$

### 2.3.3 MPEG 音频第 III 层编码心理声学算法流程

由于心理声学模型 2 的实现有很大的自由度, 所以在实际的参数计算中, 对前文所给的心理声学模型的标准算法步骤进行了相应地修改, 下面以 44.1KHz 采样频率时单通道编码为例具体说明。如果是双通道编码, 算法流程基本一致, 只是左右声道按顺序依次编码。(流程图见附图 A.2)

8) 计算功率的信噪比:

$$bc_b = 10^{\frac{-SNR_b}{10}} \quad \text{式 (2-21)}$$

9) 实际的能量阈值计算公式:

$$nb_b = en_b * bc_b \quad \text{式 (2-22)}$$

把阈值能量分配到所有 FFT 谱线上, 得到每一条谱线的阈值能量:

$$nb_\omega = nb_b / (\omega_{highb} - \omega_{lowb} + 1) \quad \text{式 (2-23)}$$

加上绝对阈值 (在专门的表中给出) 就可以得到最终的音频能量阈值  $thr_\omega$ :

$$thr_\omega = \max(nb_\omega, absthr_\omega) \quad \text{式 (2-24)}$$

10) 计算  $SMR_n$ :

最后将所有的 FFT 谱线重新分割成  $n$  个区域, 计算每个区域的信掩比。参数  $width_n = 1$  时表示心理模型的窄带型区域,  $width_n = 0$  表示宽带型区域。对于每个区域的能量  $epart_n$  计算如下:

$$epart_n = \sum_{\omega=\omega_{low_n}}^{\omega_{high_n}} R_\omega^2 \quad \text{式 (2-25)}$$

如果  $width_n = 1$ , 则每个区域的阈值  $npart_n$  为:

$$npart_n = \sum_{\omega=\omega_{low_n}}^{\omega_{high_n}} thr_\omega \quad \text{式 (2-26)}$$

如果  $width_n = 0$ , 则:

$$npart_n = \min(thr_{\omega_{lown}} \cdots thr_{\omega_{highn}}) * (\omega_{highn} - \omega_{lown} + 1) \quad \text{式(2-27)}$$

$$SMR_n = 10 \log_{10} \left( \frac{epart_n}{npart_n} \right) \quad \text{式(2-28)}$$

### 2.3.3 MPEG 音频第 III 层编码心理声学算法流程

由于心理声学模型 2 的实现有很大的自由度, 所以在实际的参数计算中, 对前文所给的心理声学模型的标准算法步骤进行了相应地修改, 下面以 44.1KHz 采样频率时单通道编码为例具体说明。如果是双通道编码, 算法流程基本一致, 只是左右声道按顺序依次编码。(流程图见附图 A.2)

1) 分区阈值的计算。由于心理模型算法将经 FFT 变换后的谱线分为若干个阈值分区, 所以首先需计算每个分区的阈值。因为第 III 层编码是以节为单位计算数据的, 所以算法首先对每节的 576 个样本进行两次 FFT, 第一次进行 1024 点 FFT, 其数据的移位长度是 576。第二次将每节的样本数分成三个短块, 对每个短块的 192 个数据进行 256 点 FFT, 其移位长度是 192。据此将进行预测性参数  $C_{\omega}(\omega)$  的计算, 计算矩阵做以下调整, 对经 FFT 后得到的谱值, 只计算前 206 条谱线的  $C_{\omega}(\omega)$ , 其余的值都被设定为 0.4。其中前 6 条谱线的  $C_{\omega}(\omega)$  是由长块的 FFT(窗口长度是 1024, 移位长度是 576)求得。对 6~205 条谱线则由短块的 FFT(窗口长度是 256, 移位长度是 192)求得, 并且它的值是采用三个短块 FFT 中的第二个短块的变换值求得。式 (2-29) 是其计算公式。

$$C_{\omega}(\omega) = \begin{cases} C_{\omega\_l}(\omega) & 0 \leq \omega < 6 \\ C_{\omega\_s}((\omega + 2) \text{DIV } 4) & 6 \leq \omega < 206 \\ 0.4 & \omega \geq 206 \end{cases} \quad \text{式(2-29)}$$

其中  $C_{\omega\_l}(\omega)$  表示长块的变换值, 同样  $C_{\omega\_s}(\omega)$  表示短块的变换值, DIV 表示整除, 结果负向取整。

根据  $C_{\omega}(\omega)$  可以得到每个分区的阈值, 流程图见附图 A.3。在 44.1KHz 的采样频率编码时, 对于长块, 算法将所有的谱线分为 63 个阈值分区, 且每个分区的频率线宽度是不同的, 而每个短块被分为 39 个分区, 一共 3 个短块。

2) 心里声学熵的计算。由于第 III 层编码中引入了混合块<sup>[6]</sup>编码的概念, 即一节数据是用长块编码还是用短块编码, 是根据心里声学熵的值来确定的, 因此需要计算心理声学熵的值, 心理声学的熵值  $pe$  定义如下:

$$pe = -\sum (cbwidth_k * \log(thr_k / (eb_k + 1))) \quad \text{式(2-30)}$$

其中,  $thr$  是分区阈值,  $eb$  是分区能量,  $k$  是阈值计算分区的序列号,  $cbwidth_k$  是每个阈值分区的宽度。

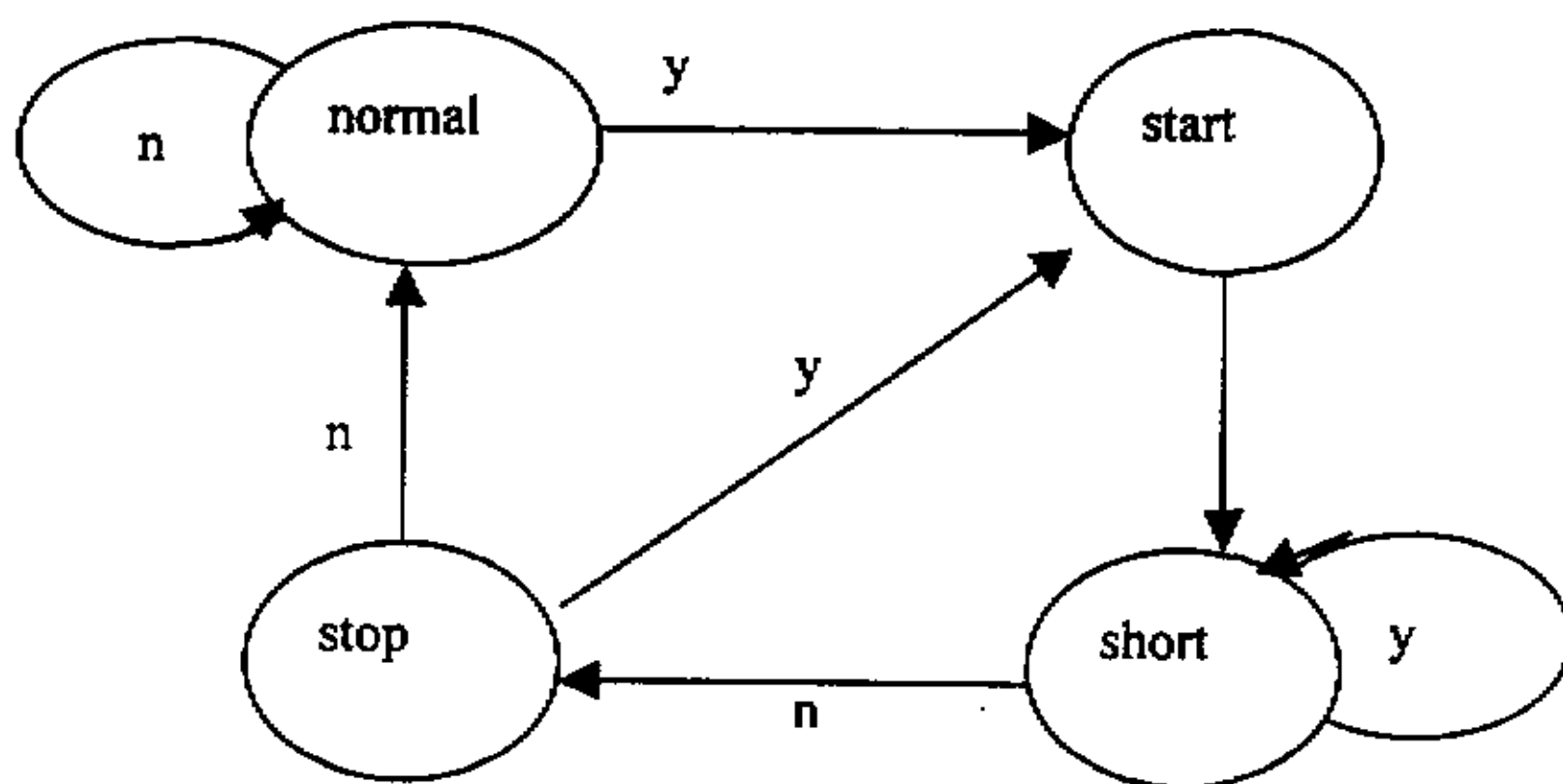


图 2.4 编码类型状态图

每节样本数据按照 normal、start、stop、short 4 种类型编码,其中前三种属于长块编码,short 类型属于短块编码。具体使用哪个类型编码取决于前一节样本数据的编码类型和当前节的  $pe$  值。根据上页图 2.4 所示流程图,选择当前节的编码类型。当  $pe > switch\_pe = 1800$ , 表示条件成立。

3) 比例因子频段阈值的计算。在实际应用的计算中,并没有像标准模型中规定的那样将得到的分区阈值分配到 FFT 的谱线上,而是将每个分区中的阈值直接转换到比例因子频段上。按照规定,对于长块,算法将 63 个阈值分区转换成 21 个比例因子频段。需要计算每个比例因子频段的信掩比率,流程图见附图 A.4。而算法将每个短块的 39 个阈值分区转换为 12 个比例因子频段。对于每节数据的三个短块一共要求出 36 个比例因子频段的信掩比率,流程图见附图 A.5。

总之,心理声学模型的输出包含:编码类型、每个比例因子频段的信掩比率以及心理声学熵值。

### 2.3.4 MDCT

根据心理声学模型的输出,对子带滤波器组的输出部分进行处理,首先要经过 MDCT<sup>[7]</sup>。

#### 1) MDCT 定义

正交变换一般是分组进行的,而对每一组系数的编码一般也是独立进行,因此量化误差对于相继各分块的影响也是不同的。由于正交变换在边界处存在着固有的不连续性,因此这些分组边界处就可能产生很大的噪声。为了削减这种噪声,通常采用将相邻分块的数据点进行部分重叠后再做变换:首先用本组的  $M$  个取样和两个相邻组的各  $K/2$  个取样组成  $(M+K)$  个样本,加窗后做  $M+K$  点离散余弦变换(DCT),得到  $M+K$  个独立的变换系数。为了减少各组间的失真,必须把  $K$  个样本重叠。由于对  $K$  个重叠点变换了两次,因而导致了 DCT 编码效率的降低。为此,Prencen 和 Bradly 提出了一种改进型 DCT,简称 MDCT。

MDCT 利用时间域的重叠对消(aliasing cancellation)技术来降低边界效应,首先,对输入序列  $x(m)$ ,用本组的  $M$  个取样和前后两个相邻组的各一半取样重叠构成  $2M$  个样本,再取满足如下的对称性  $2M$  点长的窗函数  $h(m)$ :

$$h(i)h(i) + h(i+M)h(i+M) \quad \text{式(2-31)}$$

使各组窗口中的数据间有 50%重叠,  $x(m)$  的 MDCT 系数  $Y(k)$  及其反变换的时域取样  $y(m)$  分别定义为:

$$Y(k) = \sum_{m=0}^{2M-1} x(m) \cdot h(m) \cdot \cos\left\{\frac{(2k+1)\pi}{2M}(m+m_0)\right\} \quad \text{式(2-32)}$$



每节样本数据按照 normal、start、stop、short 4 种类型编码, 其中前三种属于长块编码, short 类型属于短块编码。具体使用哪个类型编码取决于前一节样本数据的编码类型和当前节的  $pe$  值。根据上页图 2.4 所示流程图, 选择当前节的编码类型。当  $pe > switch\_pe = 1800$ , 表示条件成立。

3) 比例因子频段阈值的计算。在实际应用的计算中, 并没有像标准模型中规定的那样将得到的分区阈值分配到 FFT 的谱线上, 而是将每个分区中的阈值直接转换到比例因子频段上。按照规定, 对于长块, 算法将 63 个阈值分区转换成 21 个比例因子频段。需要计算每个比例因子频段的信掩比率, 流程图见附图 A.4。而算法将每个短块的 39 个阈值分区转换为 12 个比例因子频段。对于每节数据的三个短块一共要求出 36 个比例因子频段的信掩比率, 流程图见附图 A.5。

总之, 心理声学模型的输出包含: 编码类型、每个比例因子频段的信掩比率以及心理声学熵值。

### 2.3.4 MDCT

根据心理声学模型的输出, 对子带滤波器组的输出部分进行处理, 首先要经过 MDCT<sup>[7]</sup>。

#### 1) MDCT 定义

正交变换一般是分组进行的, 而对每一组系数的编码一般也是独立进行, 因此量化误差对于相继各分块的影响也是不同的。由于正交变换在边界处存在着固有的不连续性, 因此这些分组边界处就可能产生很大的噪声。为了削减这种噪声, 通常采用将相邻分块的数据点进行部分重叠后再做变换: 首先用本组的  $M$  个取样和两个相邻组的各  $K/2$  个取样组成  $(M+K)$  个样本, 加窗后做  $M+K$  点离散余弦变换 (DCT), 得到  $M+K$  个独立的变换系数。为了减少各组间的失真, 必须把  $K$  个样本重叠。由于对  $K$  个重叠点变换了两次, 因而导致了 DCT 编码效率的降低。为此, Prencen 和 Bradly 提出了一种改进型 DCT, 简称 MDCT。

MDCT 利用时间域的重叠对消 (aliasing cancellation) 技术来降低边界效应, 首先, 对输入序列  $x(m)$ , 用本组的  $M$  个取样和前后两个相邻组的各一半取样重叠构成  $2M$  个样本, 再取满足如下的对称性  $2M$  点长的窗函数  $h(m)$ :

$$h(i)h(i) + h(i+M)h(i+M) \quad \text{式(2-31)}$$

使各组窗口中的数据间有 50% 重叠,  $x(m)$  的 MDCT 系数  $Y(k)$  及其反变换的时域取样  $y(m)$  分别定义为:

$$Y(k) = \sum_{m=0}^{2M-1} x(m) \cdot h(m) \cdot \cos\left\{\frac{(2k+1)\pi}{2M}(m+m_0)\right\} \quad \text{式(2-32)}$$

$$y(m) = \frac{2}{M} \cdot h(m) \cdot \sum_{k=0}^{M-1} x(m) \cdot h(m) \cdot \cos\left\{\frac{(2k+1)\pi}{2M}(m+m_0)\right\} \quad \text{式(2-33)}$$

其中,  $k = 0, 1, \dots, M-1$ ;  $m = 0, 1, \dots, 2M-1$ ;  $m_0 = M/2 + 1/2$

重建信号  $x'(m)$  可以表示为:

$$x'(m) = y'(m+M) + y(m) \quad \text{式(2-34)}$$

其中,  $y'(m)$  是前一个分块样本的反变换。在编码时, 利用这 50%重叠的窗口对每一输入样本进行了两次变换, 如果  $M$  是偶数, 可得变换系数  $Y(k)$  具有如下的对称性:

$$Y(k) = -Y(2M-1-k) \quad \text{式(2-35)}$$

因此, 仍然只有  $M$  个独立的变换系数需要传输, 50%重叠变换的编码性能并没有降低。

## 2) MPEG 音频第 III 层编码中的 MDCT 算法

从心理声学模型得到每节数据的编码块类型, 子带滤波器组对不同块类型的输出有不同的计算矩阵。对于每个子带, 每节输出 18 个样本值, 再加上前一节同一子带的 18 个样本值, 一共 36 个样本作为一次 MDCT 的输入:

### I “normal”型

$$Z_i = X_i \sin\left(\frac{\pi}{36}\left(i + \frac{1}{2}\right)\right) \quad 0 \leq i < 36 \quad \text{式(2-36)}$$

### II “start”型

$$Z_i = \begin{cases} X_i \sin\left(\frac{\pi}{36}\left(i + \frac{1}{2}\right)\right) & 0 \leq i < 18 \\ X_i & 18 \leq i < 24 \\ X_i \sin\left(\frac{\pi}{12}\left(i - 18 + \frac{1}{2}\right)\right) & 24 \leq i < 30 \\ 0 & 30 \leq i < 36 \end{cases} \quad \text{式(2-37)}$$

### III “stop”型

$$Z_i = \begin{cases} 0 & 0 \leq i < 6 \\ X_i \sin\left(\frac{\pi}{12}\left(i - 6 + \frac{1}{2}\right)\right) & 6 \leq i < 12 \\ X_i & 12 \leq i < 18 \\ X_i \sin\left(\frac{\pi}{36}\left(i + \frac{1}{2}\right)\right) & 18 \leq i < 36 \end{cases} \quad \text{式(2-38)}$$

## IV “short” 型

对于 short 型, 36 个样本值被重叠分成三块:

$$Y_i^{(0)} = X_{i+6} \quad 0 \leq i \leq 12 \quad \text{式(2-39)}$$

$$Y_i^{(1)} = X_{i+12} \quad 0 \leq i \leq 12 \quad \text{式(2-40)}$$

$$Y_i^{(2)} = X_{i+18} \quad 0 \leq i \leq 12 \quad \text{式(2-41)}$$

每块分别计算:

$$Z_i^{(k)} = Y_i^{(k)} \sin\left(\frac{\pi}{12}\left(i + \frac{1}{2}\right)\right) \quad 0 \leq i < 12, 0 \leq k < 2 \quad \text{式(2-42)}$$

对 Z 取 MDCT:

$$x_i = \sum_{k=0}^{n-1} Z_k \cos\left(\frac{\pi}{2n}\left(2k+1+\frac{n}{2}\right)(2i+1)\right) \quad 0 \leq i < n/2 \quad \text{式(2-43)}$$

其中, 长块计算时  $n=36$ , 短块计算时  $n=12$ 。对于长块计算, 还需要进行蝶形对消来减少混叠。得到的每组  $x_i$  的值将被量化后进行编码。

## 2.4 量化与编码

根据心理声学模型的输出, 对经过 MDCT 后输出的样本以节为单位进行比特位的量化与分配。MPEG-I 使用了一个三级迭代的循环模型, 见图 2.5。最高一级称为帧循环程序, 帧循环程序调用一个称作外层迭代循环的子程序, 外层迭代循环调用它的子程序, 内层迭代循环。

### 2.4.1 三级迭代循环模型

首先, 帧循环对此模型所用到的迭代变量复位, 计算压缩每节数据所能提供的最大比特数。然后调用外层迭代模型, 这层循环首先调用内层迭代循环, 内层迭代循环对输入矢量进行量化, 通过递增量化步长使量化输出能够在一定的比特位数限制之内被编码。

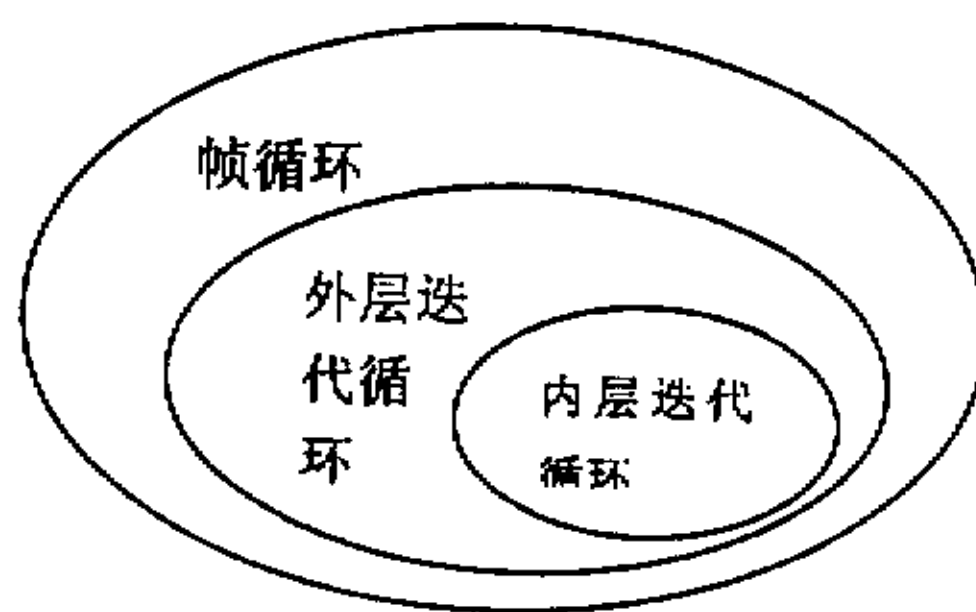


图 2.5 三层迭代模型

然后确定霍夫曼编码表的选择, 并对量化频谱进行比特编码。如果经过量化产生的噪声过大, 有可能导致样本失真, 无法被正确解码, 所以外层迭代循环根据内层输出来检测每个比例因子频段的量化噪声, 如果超过了允许的最低掩蔽阈值, 则以比例因子为迭代变量, 不断递增

## IV “short” 型

对于 short 型, 36 个样本值被重叠分成三块:

$$Y_i^{(0)} = X_{i+6} \quad 0 \leq i \leq 12 \quad \text{式(2-39)}$$

$$Y_i^{(1)} = X_{i+12} \quad 0 \leq i \leq 12 \quad \text{式(2-40)}$$

$$Y_i^{(2)} = X_{i+18} \quad 0 \leq i \leq 12 \quad \text{式(2-41)}$$

每块分别计算:

$$Z_i^{(k)} = Y_i^{(k)} \sin\left(\frac{\pi}{12}\left(i + \frac{1}{2}\right)\right) \quad 0 \leq i < 12, 0 \leq k < 2 \quad \text{式(2-42)}$$

对 Z 取 MDCT:

$$x_i = \sum_{k=0}^{n-1} Z_k \cos\left(\frac{\pi}{2n}\left(2k+1+\frac{n}{2}\right)(2i+1)\right) \quad 0 \leq i < n/2 \quad \text{式(2-43)}$$

其中, 长块计算时  $n=36$ , 短块计算时  $n=12$ 。对于长块计算, 还需要进行蝶形对消来减少混叠。得到的每组  $x_i$  的值将被量化后进行编码。

## 2.4 量化与编码

根据心理声学模型的输出, 对经过 MDCT 后输出的样本以节为单位进行比特位的量化与分配。MPEG-I 使用了一个三级迭代的循环模型, 见图 2.5。最高一级称为帧循环程序, 帧循环程序调用一个称作外层迭代循环的子程序, 外层迭代循环调用它的子程序, 内层迭代循环。

### 2.4.1 三级迭代循环模型

首先, 帧循环对此模型所用到的迭代变量复位, 计算压缩每节数据所能提供的最大比特数。然后调用外层迭代模型, 这层循环首先调用内层迭代循环, 内层迭代循环对输入矢量进行量化, 通过递增量化步长使量化输出能够在一定的比特位数限制之内被编码。

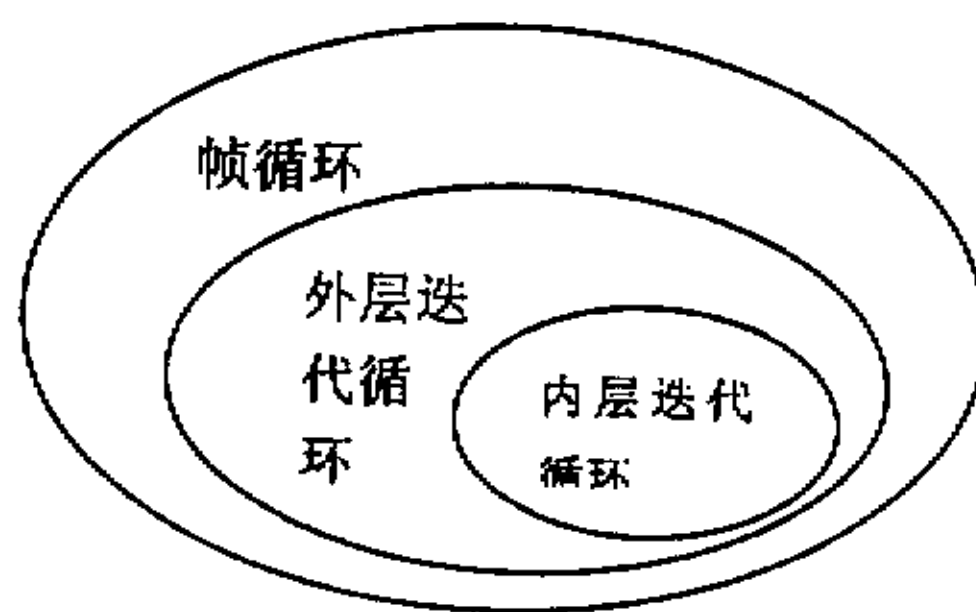


图 2.5 三层迭代模型

然后确定霍夫曼编码表的选择, 并对量化频谱进行比特编码。如果经过量化产生的噪声过大, 有可能导致样本失真, 无法被正确解码, 所以外层迭代循环根据内层输出来检测每个比例因子频段的量化噪声, 如果超过了允许的最低掩蔽阈值, 则以比例因子为迭代变量, 不断递增

## IV “short” 型

对于 short 型, 36 个样本值被重叠分成三块:

$$Y_i^{(0)} = X_{i+6} \quad 0 \leq i \leq 12 \quad \text{式(2-39)}$$

$$Y_i^{(1)} = X_{i+12} \quad 0 \leq i \leq 12 \quad \text{式(2-40)}$$

$$Y_i^{(2)} = X_{i+18} \quad 0 \leq i \leq 12 \quad \text{式(2-41)}$$

每块分别计算:

$$Z_i^{(k)} = Y_i^{(k)} \sin\left(\frac{\pi}{12}\left(i + \frac{1}{2}\right)\right) \quad 0 \leq i < 12, 0 \leq k < 2 \quad \text{式(2-42)}$$

对 Z 取 MDCT:

$$x_i = \sum_{k=0}^{n-1} Z_k \cos\left(\frac{\pi}{2n}\left(2k+1+\frac{n}{2}\right)(2i+1)\right) \quad 0 \leq i < n/2 \quad \text{式(2-43)}$$

其中, 长块计算时  $n=36$ , 短块计算时  $n=12$ 。对于长块计算, 还需要进行蝶形对消来减少混叠。得到的每组  $x_i$  的值将被量化后进行编码。

## 2.4 量化与编码

根据心理声学模型的输出, 对经过 MDCT 后输出的样本以节为单位进行比特位的量化与分配。MPEG-I 使用了一个三级迭代的循环模型, 见图 2.5。最高一级称为帧循环程序, 帧循环程序调用一个称作外层迭代循环的子程序, 外层迭代循环调用它的子程序, 内层迭代循环。

### 2.4.1 三级迭代循环模型

首先, 帧循环对此模型所用到的迭代变量复位, 计算压缩每节数据所能提供的最大比特数。然后调用外层迭代模型, 这层循环首先调用内层迭代循环, 内层迭代循环对输入矢量进行量化, 通过递增量化步长使量化输出能够在一定的比特位数限制之内被编码。

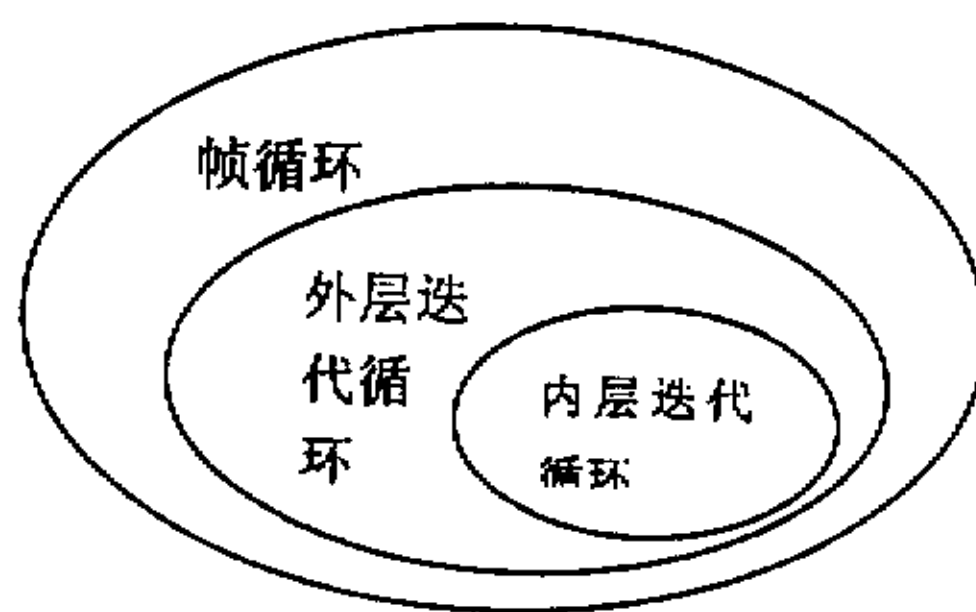


图 2.5 三层迭代模型

然后确定霍夫曼编码表的选择, 并对量化频谱进行比特编码。如果经过量化产生的噪声过大, 有可能导致样本失真, 无法被正确解码, 所以外层迭代循环根据内层输出来检测每个比例因子频段的量化噪声, 如果超过了允许的最低掩蔽阈值, 则以比例因子为迭代变量, 不断递增



其值, 重新调用内层迭代循环, 使量化满足要求。一旦满足要求, 存储每个比例因子频段最终的比例因子数值, 跳出外层迭代循环。在帧循环程序中计算存储每节数据所用的比特位数。

#### 1) 三级循环模型的输入

I: 经过 MDCT 后的频谱矢量  $x(i)$ ,  $0 \leq i \leq 575$ ;

II: 每个比例因子频段允许的最大量化噪声  $x_{\min}(sb)$  (由心理声学模型得到):

$$x_{\min}(sb) = \text{ratio}(sb) * \text{en}(sb) / b\omega(sb) \quad \text{式(2-44)}$$

III: 长短块选择标志、块类型、混合块标志。它们共同决定比例因子频段的数量;

IV: 平均比特数  $\text{mean\_bits}$ , 在比例因子频段进行霍夫曼编码时一节样本可用的平均比特数。

#### 2) 三级迭代循环模型的输出

I: 经过量化后的矢量值  $ix(0 \dots 575)$ ;

II: 比例因子  $\text{scalefac\_l}(sb)$  或  $\text{scalefac\_s}(sb)$ ; 取决于块类型, 混合块标志。 $\text{scalefac\_l}(sb)$  表示长块类型的比例因子,  $\text{scalefac\_s}(sb)$  表示短块类型的比例因子;

III: 全局增益, 包含量化步长信息;

IV: 每节数据所占的比特数;

V: 预加重标志  $\text{preflag}$ 。它的选用依赖于具体情况, 当预测到需要进行若干次量化步长时, 则启动预加重, 对比例因子频段高频段的量化步长加权量化, 从而减少迭代次数, 降低程序运行时间。

### 2.4.2 三级迭代循环模型算法流程

#### 1) 帧循环 (流程图见附 A.6)

##### I: 计算平均比特数

以 44.1KHz 的采样频率, 128Kbit/s(双通道)的编码速率为例, 一帧数据被压缩所占的比特数是 3343 比特, 去掉每帧数据的头信息 (32 比特) 和附加信息, 得到每节数据在双通道压缩时的平均比特数是 1527 比特。

##### II: 复位所有的迭代变量

在比例因子编码部分,  $\text{scalefac\_l}[sb]$  或  $\text{scalefac\_s}[sb]$  被相应地设置为 0, 量化步长的计数器  $\text{qquant}$  复位为 0, 预加重标志  $\text{preflag}$  设置为 0。

##### III: 计算比例因子选择信息( $\text{scfsi}$ )

MPEG 第 III 层编码将一帧数据分为两节来处理, 其中第一节数据的比例因子频段的比例因子必须编码传送, 它的比例因子能否被第二节利用, 即第二节数

其值, 重新调用内层迭代循环, 使量化满足要求。一旦满足要求, 存储每个比例因子频段最终的比例因子数值, 跳出外层迭代循环。在帧循环程序中计算存储每节数据所用的比特位数。

#### 1) 三级循环模型的输入

I: 经过 MDCT 后的频谱矢量  $x(i)$ ,  $0 \leq i \leq 575$ ;

II: 每个比例因子频段允许的最大量化噪声  $x_{\min}(sb)$  (由心理声学模型得到):

$$x_{\min}(sb) = \text{ratio}(sb) * \text{en}(sb) / b\omega(sb) \quad \text{式(2-44)}$$

III: 长短块选择标志、块类型、混合块标志。它们共同决定比例因子频段的数量;

IV: 平均比特数  $\text{mean\_bits}$ , 在比例因子频段进行霍夫曼编码时一节样本可用的平均比特数。

#### 2) 三级迭代循环模型的输出

I: 经过量化后的矢量值  $ix(0 \dots 575)$ ;

II: 比例因子  $\text{scalefac\_l}(sb)$  或  $\text{scalefac\_s}(sb)$ ; 取决于块类型, 混合块标志。 $\text{scalefac\_l}(sb)$  表示长块类型的比例因子,  $\text{scalefac\_s}(sb)$  表示短块类型的比例因子;

III: 全局增益, 包含量化步长信息;

IV: 每节数据所占的比特数;

V: 预加重标志  $\text{preflag}$ 。它的选用依赖于具体情况, 当预测到需要进行若干次量化步长时, 则启动预加重, 对比例因子频段高频段的量化步长加权量化, 从而减少迭代次数, 降低程序运行时间。

### 2.4.2 三级迭代循环模型算法流程

#### 1) 帧循环 (流程图见附 A.6)

##### I: 计算平均比特数

以 44.1KHz 的采样频率, 128Kbit/s(双通道)的编码速率为例, 一帧数据被压缩所占的比特数是 3343 比特, 去掉每帧数据的头信息 (32 比特) 和附加信息, 得到每节数据在双通道压缩时的平均比特数是 1527 比特。

##### II: 复位所有的迭代变量

在比例因子编码部分,  $\text{scalefac\_l}[sb]$  或  $\text{scalefac\_s}[sb]$  被相应地设置为 0, 量化步长的计数器  $\text{qquant}$  复位为 0, 预加重标志  $\text{preflag}$  设置为 0。

##### III: 计算比例因子选择信息( $\text{scfsi}$ )

MPEG 第 III 层编码将一帧数据分为两节来处理, 其中第一节数据的比例因子频段的比例因子必须编码传送, 它的比例因子能否被第二节利用, 即第二节数

据的比例因子是否传送需要经过计算。算法提供了四个标准<sup>[1]</sup>来判断,如果标准有一个不符合,则所有比例因子频段的  $scfsi=0$ ,如果  $scfsi=0$ ,则第二节的每个频段的比例因子需要重新计算并且传送,否则不传送该节各频段的比例因子,它的值按照第一节的比例因子系数计算。

## 2) 内层迭代循环 (流程图见图 A.8)

内层循环通过式 2-45 对 MDCT 输出的频谱值进行量化。

$$ix(i) = NINT\left(\left(\frac{|x(i)|}{(\sqrt[4]{2})^{qquant + quantan f}}\right)^{0.75} - 0.0946\right) \quad \text{式(2-45)}$$

其中, NINT 返回最接近的整数值, qquant 是量化步长。quantanf 的初始值由帧循环设置如下:

$$quantan f = system\_const * \log_e(sfm) \quad \text{式(2-46)}$$

其中  $system\_const=210$ , 谱平坦分量  $sfm$  的值依赖于输入样本频谱矢量的计算:

$$sfm = \frac{e^{\frac{1/n \sum_{i=0}^{n-1} \log x(i)^2}{1/n \sum_{i=0}^{n-1} x(i)^2}}}{1/n \sum_{i=0}^{n-1} x(i)^2} \quad \text{式(2-47)}$$

式 2-45 的计算得到所有频谱矢量的量化值, MPEG-I 要求对量化值进行霍夫曼编码。霍夫曼编码对量化的最大值有限制, MPEG-I 规定最大值不能超过 8191。所以首先判断所有的量化值是否超过限定, 如果超过最大限制, 内层迭代循环递增量化步长 qquant, 重新量化。然后确定其霍夫曼编码位数, 使其所占的比特数小于每节编码所能提供的最大比特数 (由帧循环计算得到), 否则也要递增量化步长重新量化。

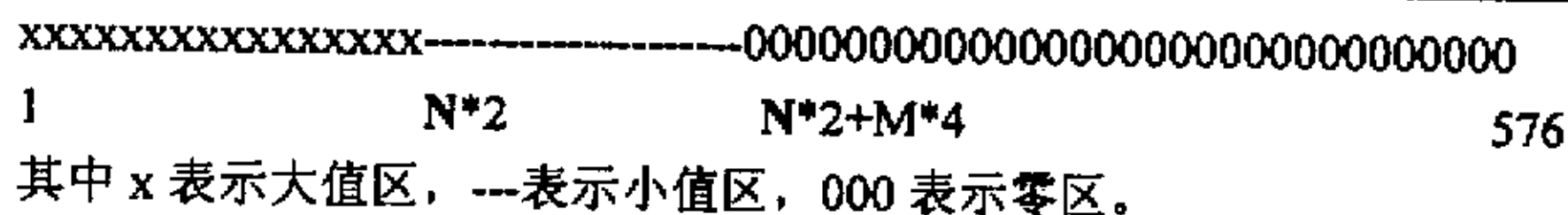


图 2.6 霍夫曼编码规则示意图

其中, 对量化值的霍夫曼编码规则如图 2.6 所示。对于高频区值是 0 的量化值不编码, 这部分被称为“零区”。接着第一个不是零的数值开始计算量化值的绝对值小于等于 1 (1, 0, -1) 的值的长度, 这一部分称为小值区。小值区的量化

值是以 4 个为一组进行霍夫曼编码的, 一共有 M 组, MPEG-I 标准为这一部分提供了 2 个霍夫曼编码表以供选择。

剩下的部分是大值区。霍夫曼编码对这一部分是成对编码的, 共有 N 对。这一部分又分成 2 个或 3 个区域, 是对大值区的进一步细分, 从而获得更高的错误恢复能力和鲁棒性。在非短块编码时, 大值区被分为三个区域 (0 区, 1 区, 2 区), 具体分割可灵活选定。标准规定在长块编码时每节包含的比例因子频段总数是 21, 在本系统设计中, 将比例因子频段的前 1/3 作为 0 区, 后 1/4 作为 2 区。在短块编码时, 大值区被分为两个区域 (I 区、II 区), 此时 I 区、II 区包含的频段都是 MPEG-I 规定的默认值。标准提供了 32 个不同的霍夫曼编码表, 每个分区分别应用不同的表进行编码。

霍夫曼表选择是根据量化值的最大值以及信号的统计特性来决定的。大值区采用线性编码和霍夫曼码表结合的编码方式。如果量化值小于 15, 直接利用码表编码, 如果大于等于 15, 则将值分为两部分进行编码。首先, 量化值减去 15 所得到的数值利用一个线性编码整数进行二进制编码, 二进制的所占的比特数用 linbits 来表示。15 再用对应的霍夫曼码表编码, 从而完成一个量化值的编码。不同数值的 linbits 对应不同的霍夫曼码表, 因为  $0 \leq \text{linbits} \leq 13$ , 所以最大的量化值是 8191。例如一个分区的最大量化值是 300, 则其需要 9 位二进制数来表示, 所以选用 linbits 为 9 的霍夫曼表, 此时对于一量化值是 20 的数, 把它分解为 5+15, 5 用 9 位二进制数来表示, 15 选择 linbits 为 9 的霍夫曼表对应的值编码。

### 3) 外层迭代循环 (流程图见附图 A.7)

外层迭代循环也称失真控制循环, 这级循环控制在内层迭代循环过程中由于谱线的量化而产生的量化噪声。通过给比例因子频段分配比例因子从而对量化噪声进行整形, 每次比例因子都是以 1 开始。若发现某个频段内的量化噪声超过了允许的阈值 (由心理声学模型给出), 则通过递增比例因子来减小量化噪声。因为越小的量化噪声, 就需要越多的量化次数才能被掩盖。一旦所有频段的量化噪声都在允许的范围之内, 则外层迭代循环结束。

外层迭代循环的步骤:

步骤 1: 调用内层迭代循环;

步骤 2: 计算每一个比例因子频段的量化噪声  $\text{xfsf}(sb)$ :

$$\text{xfsf}(sb) = \frac{\sum_{i=b(sb)}^{b(sb)+bw(sb)-1} (|x(i)| - ix(i)^{4/3} * \sqrt[4]{2}^{qq_{quant} + q_{quant} \tan f})^2}{\text{bandwidth}(sb)} \quad \text{式(2-48)}$$

步骤 3: 保存比例因子频段信息;

包括比例因子和量化步长等, 万一外层循环未能计算出合适的结果就被取消

时, 这些被保存的值将会作为一个近似值同量化的频谱值一起被传送。

步骤 4: 预加重;

当预加重标志  $\text{preflag}=1$  时, 此项功能有效。它通过所给列表可以放大高频部分的量化值,  $\text{preflag}$  是否有效取决于具体实现, 例如, 当首次内层循环调用之后, 在高频段的四个比例因子频段都超出了掩蔽阈值, 就可以设置  $\text{preflag}=1$ 。

步骤 5: 放大超出掩蔽阈值的比例因子频段;

如果一个比例因子频段超过了允许的最小掩蔽阈值, 则此频段内的所有量化频谱值都会被加权一个比例因子, 重新量化。

步骤 6: 循环过程结束的条件;

通常情况下, 如果没有比例因子频段超过允许的掩蔽阈值, 则循环结束。同时规定在下面两种情况下循环结束。一是所有频段的比例因子都被放大, 二是有比例因子的值超过了规定的上限。在长块编码中, 第 1~11 的比例因子频段的比例因子不能超过 15, 剩下的 10 个频段的比例因子不能超过 10。对于短块编码, 第 1~18 的频段的最大值是 15, 其余的最大值是 7。

## 2.5 MPEG 音频第 III 层数据流格式

为了使编码数据能够被正确解码, 必须按照 MPEG 的标准对其进行格式化处理。通常一帧数据有帧头信息、附带信息、以及主数据组成, 对于 44.1KHz 的采样频率, 128Kbit/s (双通道) 的压缩比特率时的编码来说, 一帧的数据量是 417 字节。如图 2.7 所示。

### 1) 帧头信息

任何一帧被编码的数据都是由帧头开始的, 帧头信息由 32 位比特数组成, 以每帧数据的帧同步字 (FFF, 十六进制) 为起始标志, 紧接帧同步字后的第 13 个比特位如果是 1 表示遵循 MPEG-I 的标准, 是 0 则是按照 MPEG-II 的标准编码。接下来的编码数据指明了编码层、码速率、采样频率、以及编码模式等信息。对于音频编码, 采用了 44.1KHz 的采样频率, 128Kbit/s 的比特率 (双通道), 联合立体声编码, 其对应的 32 位比特数为: FF FB 90 60, 如图 2.7 所示。

同时, 可以在帧头之后传送 16 位的冗余循环校验码, 式 2-49 给出了它的校验多项式。

$$G(x) = x^{16} + x^{15} + x^2 + 1 \quad \text{式(2-49)}$$

对本系统经过反复验证, 可以认为编码出错的概率很小, 所以在本系统中将此校验步骤省去。



时, 这些被保存的值将会作为一个近似值同量化的频谱值一起被传送。

步骤 4: 预加重;

当预加重标志  $\text{preflag}=1$  时, 此项功能有效。它通过所给列表可以放大高频部分的量化值,  $\text{preflag}$  是否有效取决于具体实现, 例如, 当首次内层循环调用之后, 在高频段的四个比例因子频段都超出了掩蔽阈值, 就可以设置  $\text{preflag}=1$ 。

步骤 5: 放大超出掩蔽阈值的比例因子频段;

如果一个比例因子频段超过了允许的最小掩蔽阈值, 则此频段内的所有量化频谱值都会被加权一个比例因子, 重新量化。

步骤 6: 循环过程结束的条件;

通常情况下, 如果没有比例因子频段超过允许的掩蔽阈值, 则循环结束。同时规定在下面两种情况下循环结束。一是所有频段的比例因子都被放大, 二是有比例因子的值超过了规定的上限。在长块编码中, 第 1~11 的比例因子频段的比例因子不能超过 15, 剩下的 10 个频段的比例因子不能超过 10。对于短块编码, 第 1~18 的频段的最大值是 15, 其余的最大值是 7。

## 2.5 MPEG 音频第 III 层数据流格式

为了使编码数据能够被正确解码, 必须按照 MPEG 的标准对其进行格式化处理。通常一帧数据有帧头信息、附带信息、以及主数据组成, 对于 44.1KHz 的采样频率, 128Kbit/s (双通道) 的压缩比特率时的编码来说, 一帧的数据量是 417 字节。如图 2.7 所示。

### 1) 帧头信息

任何一帧被编码的数据都是由帧头开始的, 帧头信息由 32 位比特数组成, 以每帧数据的帧同步字 (FFF, 十六进制) 为起始标志, 紧接帧同步字后的第 13 个比特位如果是 1 表示遵循 MPEG-I 的标准, 是 0 则是按照 MPEG-II 的标准编码。接下来的编码数据指明了编码层、码速率、采样频率、以及编码模式等信息。对于音频编码, 采用了 44.1KHz 的采样频率, 128Kbit/s 的比特率 (双通道), 联合立体声编码, 其对应的 32 位比特数为: FF FB 90 60, 如图 2.7 所示。

同时, 可以在帧头之后传送 16 位的冗余循环校验码, 式 2-49 给出了它的校验多项式。

$$G(x) = x^{16} + x^{15} + x^2 + 1 \quad \text{式(2-49)}$$

对本系统经过反复验证, 可以认为编码出错的概率很小, 所以在本系统中将此校验步骤省去。

```

00000000h: FF FB 90 60 00 00 02 FA 49 D9 03 0C 3A F0 5F 49 ;
00000010h: 3A B0 3D ED 5E 0B E9 25 62 0C 10 EB C1 7D 22 64 ;
00000020h: 60 F7 A9 78 15 3A 7C C2 D2 49 CB 8A BE F7 AC EA ;
00000030h: 7B D7 84 E3 E9 DB CF 4E 1B BB A4 23 7C DD 99 BF ;
00000040h: 5E 3F 9E E7 20 C8 A5 FE 89 79 65 14 6D DE 29 9B ;
00000050h: C5 8C 3E 33 70 D4 90 58 62 56 4F E3 61 8A DC 89 ;
00000060h: EC 65 0F B3 9F DB 2A 37 A5 CD 7C D3 6E AA CD A2 ;
00000070h: CE 7F 7F ED DD A3 FE EF E2 0F FF 5F FF FF FF ;
00000080h: FF FF FF CB 2D B3 30 28 46 5B 83 12 F2 BA FE F0 ;
00000090h: 14 53 F6 B3 EA 5C 89 B1 D6 D4 1F A7 3A CD 94 34 ;
000000a0h: 8E E6 B9 5A 66 FA EE 6A 86 A4 31 6A 07 02 47 47 ;
000000b0h: CB DA 97 FF 8F FE E5 56 5C 8E 2A 48 E5 76 EA 96 ;
000000c0h: 2E 79 72 36 78 E4 B7 2D 5D 97 3C 72 D9 FC 8F 9F ;
000000d0h: 5D 8E CF 2A C7 33 93 DF EF 9E EB CA EA B1 CB FF ;
000000e0h: FF FF FF 5E 4B 5F 5E 5F 5C FE 0F 2F ED 4D 7E 32 ;
000000f0h: 05 EB 7B D9 D6 ED 95 47 27 0B 7A 7F 7E DA B4 ;
00000100h: 0E EB 7E 3F 07 76 BC BA 4A 7F 0D 7C 7E 8E 5E ;
00000110h: 40 7E 5E 9A 4E D0 5E 5F 5C FE 0F 2F ED 4D 7E 32 ;
00000120h: 2E 79 72 36 78 E4 B7 2D 5D 97 3C 72 D9 FC 8F 9F ;
00000130h: 5D 8E CF 2A C7 33 93 DF EF 9E EB CA EA B1 CB FF ;
00000140h: FF FF FF 5E 4B 5F 5E 5F 5C FE 0F 2F ED 4D 7E 32 ;
00000150h: 05 EB 7B D9 D6 ED 95 47 27 0B 7A 7F 7E DA B4 ;
00000160h: 0E EB 7E 3F 07 76 BC BA 4A 7F 0D 7C 7E 8E 5E ;
00000170h: 40 7E 5E 9A 4E D0 5E 5F 5C FE 0F 2F ED 4D 7E 32 ;
00000180h: 2E 79 72 36 78 E4 B7 2D 5D 97 3C 72 D9 FC 8F 9F ;
00000190h: 5D 8E CF 2A C7 33 93 DF EF 9E EB CA EA B1 CB FF ;
000001a0h: FF FF FB 90 60 00 00 02 FA 3B D8 B3 09 3A E0 5F ;

```

图 2.7 一帧 MP3 格式音频双通道编码十六进制数据图

## 2) 附带信息

因为一帧数据分为两节来编码, 所以附带信息中包含了每节中的每个声道(双声道或单声道)的控制信息, 包括每个声道的数据所占的比特数、小值区、大值区量化值的个数、还有编码数据的块类型信息、频段的比例因子长度及其量化步长信息和霍夫曼码表的选择信息等等。具体内容请参阅 ISO/IEC11172-3。对于单声道音频编码, 附带信息包含 17 字节的数据, 双声道包含 32 字节的数据, 如图 2.7 方框中的阴影部分。在此部分中, 包含以下一些主要编码信息:

I: 主数据开始位置。紧接着帧头信息, 用一个 9 比特的无符号二进制数表示主数据的开始位置。它的值用来确定一帧中主数据的第一个比特位相对于这一帧的同步字的负相偏移距离, 因为主数据并不一定直接从附加信息后开始。只有此二进制数为 0, 主数据才直接从附加信息后开始。

II: 比例因子传送信息。在音频第 III 层编码中, 将一帧的数据分为两节, 每节数据又被分为若干个比例因子频段, 编码时要对每个比例因子频段分配比例因子, 本章第四节中提到, 在满足一定标准后, 后一节数据的比例因子与前一节共享, 无需传送, 所以算法用一个比特位  $scfsi$  来表示此信息。当  $scfsi=0$  时, 表示两节都必须传送相应频段的比例因子, 当  $scfsi=1$  时, 表示第一节比例因子同样适用于第二节的相应频段, 此时无需传送第二节的比例因子。当块类型为短块时,

scfsi=0。

III: 主数据的比特数。用一个 12 比特的无符号二进制数表示每节中每个通道用来传送比例因子及霍夫曼编码数据的比特位数。由于附加信息占用的比特位数是确定的(对于单通道包含 17 字节, 双通道包含 32 字节), 所以可以用它的值计算下一个通道或下一节编码数据开始的位置。

IV: 比例因子占用的比特数。算法利用两个两比特的二进制数  $slen1$ 、 $slen2$  可以求得比例因子所占的比特数。对于长块编码, 即编码类型是 normal, stop, start。  $slen1$  表示比例因子频段 1~11 的比例因子所占的比特位数,  $slen2$  表示频段 12~21 的比例因子所占的比特位数。对于短块编码, 即编码类型是 short。  $slen1$  表示频段 1~6 的比例因子所占的比特位数,  $slen2$  表示频段 7~12 的比例因子所占的比特位数。对于混合块编码, 即表示低频段的两个子带滤波器输出值利用长块计算, 其余的 30 个子带用短块计算。  $slen1$  表示比例因子频段 0~7(长块)的比例因子所占的比特位数及比例因子频段 3~5(短块)的比例因子所占的比特数,  $slen2$  表示频段 6~11(短块)的比例因子所占的比特数。

由此得到每节样本编码比例因子所占得比特数  $factor\_len$ :

$$factor\_len = \begin{cases} 11 * slen1 + 10 * slen2 & \text{长块编码} \\ 3 * (6 * slen1 + 6 * slen2) & \text{短块编码} \\ slen1 * 8 + 3 * (3 * slen1 + 6 * slen2) & \text{混合块编码} \end{cases} \quad \text{式(2-50)}$$

VI: 霍夫曼码表选择信息。对长块编码, 算法将比例因子频段分为三个区域, 每个区域的霍夫曼码表选择信息占用 5 个比特。另外, 算法利用 7 个比特位表示了三个区域的边界信息, 可以得出每个区域包含的比例因子频段的数目。

VII: 预加重标志。附加的对高频区频段的量化值步长放大的快速算法, 若它的值是 1, 对比例因子加权, 与再量化比例因子用表值倍乘的效果相同。

3) 主数据。包括每个比例因子频段的比例因子值和霍夫曼编码数据, 解码器将通过帧头信息与附带信息对主数据解码。如图 2.7 所示, 附加信息后的主数据分为两节, 图中用阴影加以区别。每节数据都包含左右两个声道的编码数据, 其中每个声道的数据平均是 95 字节。包括每个频段的比例因子数值和霍夫曼编码数据。比例因子所占的比特数在附加信息中给出, 这样就可以计算霍夫曼编码数据包含的比特数。

霍夫曼编码按以下规则进行: 对大值区采用成对编码, 本章第四节中提到, 对绝对值小于 15 的量化值直接进行编码, 如果量化值大于等于 15, 将会被分为两部分来编码。如果一对值的一个或两个值不是 0, 则在码流中需要增加一个或两个符号位。霍夫曼编码的顺序是由编码块类型决定的。若是长块编码, 则编码数据按频率增加顺序排序。若是短块编码, 按比例因子频段的顺序, 从比例因子

频段 0 开始。在每个比例因子频段内, 从块 0 开始到块 2 结束, 在每个块内, 量化值按频率增加的顺序排序。

## 2.6 低采样率语音编码特性

为了保证其算法的普遍性, MPEG-II 标准算法又进行了低采样频率的算法扩展。标准补充了 16kHz, 22.05kHz, 24kHz 三个低采样频率的编码算法, 其压缩比特率从 8Kbit/s~160Kbit/s。这样对于音质要求不是很高的语音编码, 可采用此扩展算法, 从而节省数据的存储空间。我们在硬件系统实现中, 对于音频录音采用了 44.1kHz 的采样频率和 128kbps 的码速率 (双通道)。而对语音录音则采用了低采样频率扩展的 16kHz 的采样频率和 16kbps 的码速率 (单通道), 这样对于一片 32M 的 Flash 存储器可以录制大约 4.5 小时的语音节目。

```
00000000h: FF F3 2B C0 00 0E C0 6E 98 00 D2 4C 48 39 00 5A ;
00000010h: 6C 10 D0 9A 73 B4 FF 4C C2 96 22 44 1A 49 6B BA ;
00000020h: 60 C9 2F 7A 32 BC 9E 2A 0A B6 AD A8 69 C2 20 09 ;
00000030h: B5 08 B3 90 30 51 BC 06 99 09 0A 50 25 D0 EB 9D ;
00000040h: 92 74 53 EA 53 2A FF FF FF F3 2B C0 00 0E C0 7A ;
```

图 2.8 一帧 MP3 格式语音单通编码十六进制数据图

对于低采样率的单通道语音编码算法与上述 MPEG-I 标准有一些差别。首先, 一帧的样本数是 576, 在算法中也没有分节处理, 就以一帧为单位编码。由于去掉了与节相关的一些编码信息, 所以对于帧码流来说, 其附加信息仅包含 9 字节, 如图 2.8 中的阴影部分。并且由图可以得到 16KHz 的采样频率时, 被编码后一帧的数据是 72 字节。

频段 0 开始。在每个比例因子频段内, 从块 0 开始到块 2 结束, 在每个块内, 量化值按频率增加的顺序排序。

## 2.6 低采样率语音编码特性

为了保证其算法的普遍性, MPEG-II 标准算法又进行了低采样频率的算法扩展。标准补充了 16kHz, 22.05kHz, 24kHz 三个低采样频率的编码算法, 其压缩比特率从 8Kbit/s~160Kbit/s。这样对于音质要求不是很高的语音编码, 可采用此扩展算法, 从而节省数据的存储空间。我们在硬件系统实现中, 对于音频录音采用了 44.1kHz 的采样频率和 128kbits/s 的码速率 (双通道)。而对语音录音则采用了低采样频率扩展的 16kHz 的采样频率和 16kbits/s 的码速率 (单通道), 这样对于一片 32M 的 Flash 存储器可以录制大约 4.5 小时的语音节目。

```
00000000h: FF F3 2B C0 00 0E C0 6E 98 00 D2 4C 48 39 00 5A ;
00000010h: 6C 10 D0 9A 73 B4 FF 4C C2 96 22 44 1A 49 6B BA ;
00000020h: 60 C9 2F 7A 32 BC 9E 2A 0A B6 AD A8 69 C2 20 09 ;
00000030h: B5 08 B3 90 30 51 BC 06 99 09 0A 50 25 D0 EB 9D ;
00000040h: 92 74 53 EA 53 2A FF FF FF F3 2B C0 00 0E C0 7A ;
```

图 2.8 一帧 MP3 格式语音单通编码十六进制数据图

对于低采样率的单通道语音编码算法与上述 MPEG-I 标准有一些差别。首先, 一帧的样本数是 576, 在算法中也没有分节处理, 就以一帧为单位编码。由于去掉了与节相关的一些编码信息, 所以对于帧码流来说, 其附加信息仅包含 9 字节, 如图 2.8 中的阴影部分。并且由图可以得到 16KHz 的采样频率时, 被编码后一帧的数据是 72 字节。



### 第三章 MPEG 音频第 III 层编码硬件系统实现

因为上述压缩算法将被用来实现数码 MP3 实时编码适配器,所以必须通过硬件实现其算法。本系统是以两片 DSP 作为编码器的主要部分,还包括音频 A/D 转换,单片机的控制模块,电池供电系统, LCD 显示部分, 按键输入部分, 存储器模块等组成, 图 3.1 给出了系统框图。

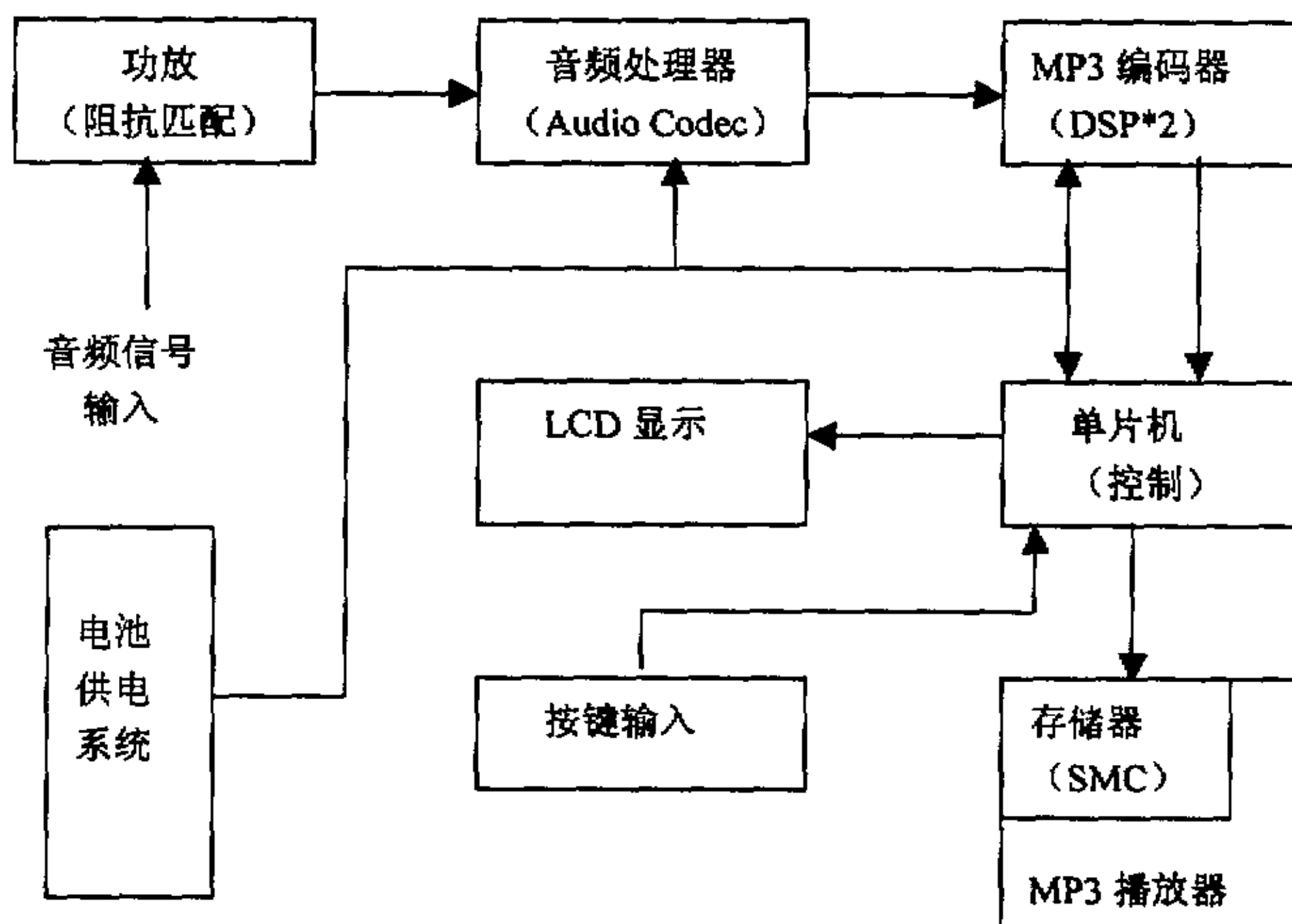


图 3.1 MP3 编码适配器系统框图

首先,对输入的模拟音频信号放大,然后由音频处理器对信号进行 A/D 转换,使其变为 PCM (脉冲编码调制) 格式的数字音频信号。MP3 编码器对输入的数字信号进行编码,把数据压缩成为具有 MP3 格式的码流,单片机接收压缩后的码流,并经过一定的处理之后,将其存储在 MP3 播放器的存储器中。下面对各模块分节讨论。

#### 3.1 MP3 编码器

MP3 编码器是整个适配器系统的核心部分,它用来实时完成在第二章中描述的 MPEG 标准算法。它是由两片型号为 TMS320VC5402 的 DSP 芯片串行组成。

### 3.1.1 DSP 芯片介绍

DSP 芯片<sup>[8]</sup>, 也称数字信号处理器, 是一种具有特殊结构的微处理器, 其内部采用哈佛结构。哈佛结构的主要特点是程序和数据存储器是两个相互独立的存储器, 具有独立的地址、数据和控制总线, 从而使数据的吞吐率大大提高。由于程序和数据存储器在两个独立的空间, 因此取指和执行能完全并行。同时 DSP 芯片配有专用的硬件乘法器, 广泛采用流水线操作<sup>[9]</sup>, 以减少指令执行的时间, 从而增强了处理器的处理能力。

在设计 DSP 应用系统中, 选择 DSP 芯片是非常重要的一个环节, 应根据实际的具体应用系统需要来确定。对于本文的 MP3 编码器, 需要考虑以下因素。

1) 运算速度。运算速度是 DSP 芯片的一个最重要的性能指标, 也是选择 DSP 芯片时所需要考虑的一个主要因素。由于 MPEG 算法是以帧为单位处理数据的, 所以选择 DSP 芯片应该比较一帧内 DSP 芯片的处理能力和 DSP 算法的运算量。

2) 价格。由于 MP3 编码器属消费类电子, 所以选定一个价格合适的 DSP 芯片是比较重要的。

3) 开发工具和硬件资源。

4) 功耗: 这一点对消费者来说, 非常关键, 由于 DSP 全速工作时的功耗很大, 所以要结合速度与功耗综合考虑。

5) 其它因素: 如封装的形式、质量标准、生命周期等。

基于此, 我们选用两片 TMS320VC5402<sup>[10]</sup> (以下简称 C5402) 来实现方案要求。C5402 是德州仪器公司 (TI) 1999 年 10 月推出的性价比较高 (目前价格仅为 5 美元) 的定点数字信号处理器 (DSP)。C5402 主要特点如下:

- 运算速率达 100MIPS;
- 具有先进的多总线结构, 三条 16 位数据存储器总线和一条程序存储器总线;
- 40 位算术逻辑单元 (ALU), 包括一个 40 位循环移位器和两个 40 位累加器;
- 一个 17bit×17bit 乘法器和一个 40 位专用加法器, 允许 16 位带 / 不带符号的乘法;
- 数据 / 程序寻址空间 1M×16bit, 内置 4K×16bit ROM 和 16K×16bit RAM;
- 内置可编程等待状态发生器、锁相环 (PLL) 时钟产生器、两个多通道缓存串行口、一个 8 位并行与外部处理器通信的 HPI 口、2 个 16 位定时器以及 6 通道 DMA 控制器;
- 低功耗, 工作电源有 3V 和 1.8V (内核), 特别适合电池供电设备。

由此可见, 与 TMS32054X 系列的其它芯片相比, C5402 以其独有的高性能、低功耗和低成本特性, 使得一推出就受到业内用户的欢迎。特别是 C5402 有两个

MCBPS<sup>[11]</sup> (多通道缓存串行口), 它提供了全双工的通信机制以及双缓存的发送寄存器和三缓存的接收寄存器, 允许连续的数据流传送, 数据长度可以为 8、16、20、24、32 比特。MCBSP 通过 7 个引脚使数据通路和控制通路与外部设备相连。数据经 MCBSP 与外设的通信通过 DX 和 DR 引脚传输, 控制信号由 CLKX、CLKR、FSX、FSR (R 表示接收, X 表示发送) 等四条引脚实现。

CPU 和 DMA 控制器可以读取寄存器 DRR[1,2] 的数据实现接收, 并且可以对 DXR[1, 2] 写入数据实现发送。接收和发送帧同步时钟既可以由内部采样速率产生器产生, 也可以由外部时钟源驱动。当 FSR 和 FSX 都为输入时 (FSXM=FSRM=0, 外部时钟源驱动), MCBSP 分别在 CLKR 和 CLKX 的下降沿检测, 且 DR 的数据也在 CLKR 的下降沿进行采样。当 FSR 和 FSX 都为输出时, 则在 CLKX 和 CLKR 的上升沿进行检测。

16 位的串行口控制器 SPCR[1,2] 和引脚控制寄存器 PCR 用来配置串行口; 接收控制寄存器 RCR[1,2] 和发送控制寄存器 XCR[1,2] 分别设置接收和发送的不同参数。图 3.2 给出了 FSX 和 FSR 为输出时 MCBSP 的发送和接收时序图, 另外, MCBSP 还可以通过 (R/X)DATDLY 寄存器设置接收和发送数据延迟, 通过 (R/X) PHASE 设置接收和发送的多阶段。MP3 编码器正是利用了 DSP 的缓存串口进行数据间的传送与接收。

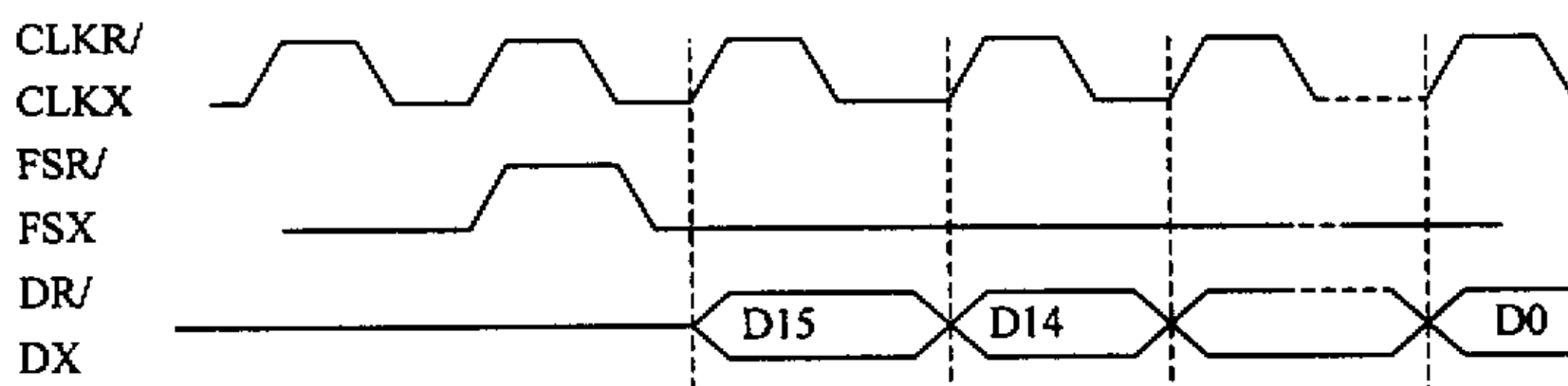


图 3.2 MCBSP 的接收发送时序图

### 3.1.2 MP3 编码器硬件连接

本系统中利用 C5402 来实现两种模式 (44.1KHz, 16KHz 的采样率) 的 MPEG 音频第 III 层编码算法。因为对于音频帧编码来说, 一帧的延时是 26ms。如果想用一片 C5402 实现 MPEG 压缩算法, 无论从硬件资源, 指令周期, 还是功耗等方面考虑, 都是不可行的。由于 C5402 有两个通道的缓存串口, 所以考虑用两片 C5402 串行<sup>[15]</sup>工作来实现, 其原理框图如图 3.3 所示。这样, 可以增加一倍的指令延时, 从而在完成算法功能的基础上, 降低指令运算速度以减小功耗。同时充分利用了芯片的硬件资源, 使算法实现的实时性得以满足, 而且降低编程难度, 缩短开发周期。

MCBPS<sup>[11]</sup> (多通道缓存串行口), 它提供了全双工的通信机制以及双缓存的发送寄存器和三缓存的接收寄存器, 允许连续的数据流传送, 数据长度可以为 8、16、20、24、32 比特。MCBSP 通过 7 个引脚使数据通路和控制通路与外部设备相连。数据经 MCBSP 与外设的通信通过 DX 和 DR 引脚传输, 控制信号由 CLKX、CLKR、FSX、FSR (R 表示接收, X 表示发送) 等四条引脚实现。

CPU 和 DMA 控制器可以读取寄存器 DRR[1,2] 的数据实现接收, 并且可以对 DXR[1, 2] 写入数据实现发送。接收和发送帧同步时钟既可以由内部采样速率产生器产生, 也可以由外部时钟源驱动。当 FSR 和 FSX 都为输入时 (FSXM=FSRM=0, 外部时钟源驱动), MCBSP 分别在 CLKR 和 CLKX 的下降沿检测, 且 DR 的数据也在 CLKR 的下降沿进行采样。当 FSR 和 FSX 都为输出时, 则在 CLKX 和 CLKR 的上升沿进行检测。

16 位的串行口控制器 SPCR[1,2] 和引脚控制寄存器 PCR 用来配置串行口; 接收控制寄存器 RCR[1,2] 和发送控制寄存器 XCR[1,2] 分别设置接收和发送的不同参数。图 3.2 给出了 FSX 和 FSR 为输出时 MCBSP 的发送和接收时序图, 另外, MCBSP 还可以通过 (R/X)DATDLY 寄存器设置接收和发送数据延迟, 通过 (R/X) PHASE 设置接收和发送的多阶段。MP3 编码器正是利用了 DSP 的缓存串口进行数据间的传送与接收。

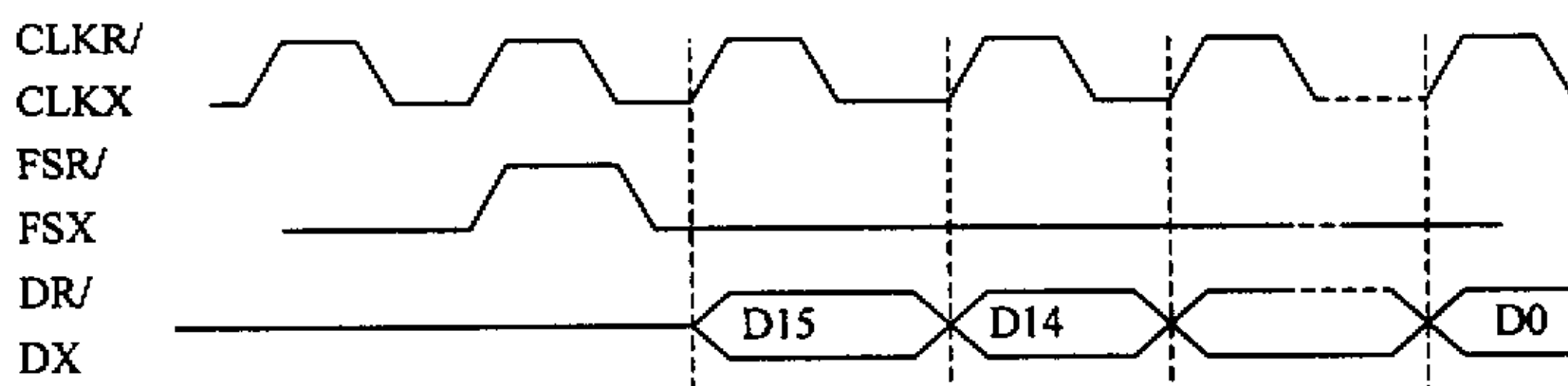


图 3.2 MCBSP 的接收发送时序图

### 3.1.2 MP3 编码器硬件连接

本系统中利用 C5402 来实现两种模式 (44.1KHz, 16KHz 的采样率) 的 MPEG 音频第 III 层编码算法。因为对于音频帧编码来说, 一帧的延时是 26ms。如果想用一片 C5402 实现 MPEG 压缩算法, 无论从硬件资源, 指令周期, 还是功耗等方面考虑, 都是不可行的。由于 C5402 有两个通道的缓存串口, 所以考虑用两片 C5402 串行<sup>[15]</sup>工作来实现, 其原理框图如图 3.3 所示。这样, 可以增加一倍的指令延时, 从而在完成算法功能的基础上, 降低指令运算速度以减小功耗。同时充分利用了芯片的硬件资源, 使算法实现的实时性得以满足, 而且降低编程难度, 缩短开发周期。

由图 3.3 可见,充分利用了 C5402 提供的两个缓存串口进行数据通信。首先,模拟的音频信号通过一个音频的 A/D 处理器 (Philips 的 UDA1341TS),它的系统时钟 SYSCLK、样本采样时钟 WS、位采样时钟 BCK 都由 DSP 提供。不同的是,对音频编码,时钟是由 DSP1 提供的。它提供的时钟频率是 11.2896MHz,即 DSP1 的输入晶振频率,如图 3.3 虚线所示。对语音编码,则由 DSP2 提供,其频率也是

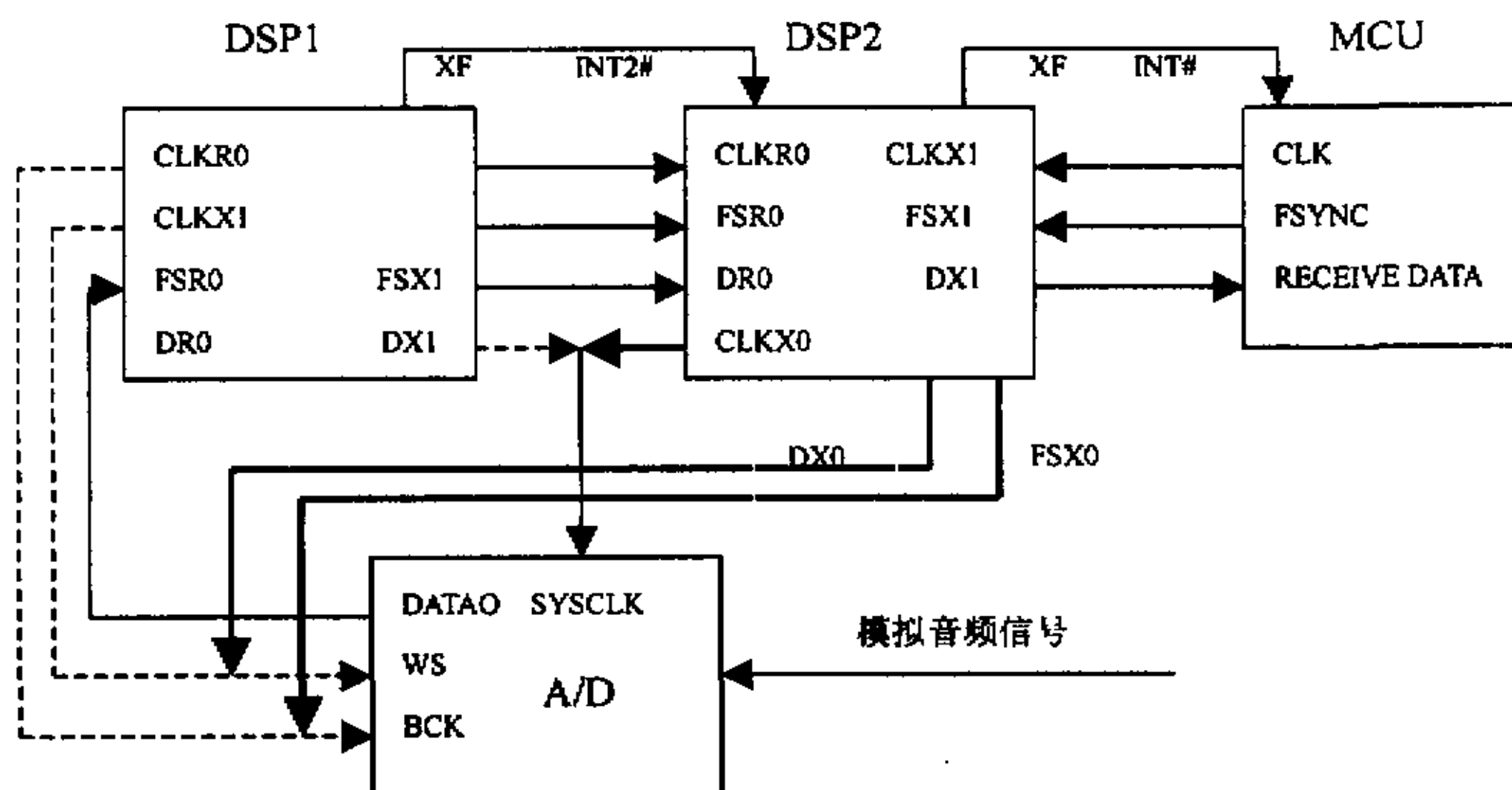


图 3.3 基于 DSP 的系统实现原理框图

DSP2 的晶振输入频率 12MHz,如图 3.3 粗线所示。经过 A/D 转换后的数据通过缓存串口 1 进入 DSP1, DSP1 完成了 MPEG 算法时域到频域的映射和心理声学模型计算。当处理完一帧中每节的数据后就利用 XF 信号产生一个中断给 DSP2,并且从缓存串口 2 输出数据。而 DSP2 通过缓存串口 1 接收数据, DSP2 完成算法中的量化与编码和帧数据流格式化计算。当一帧数据形成后就产生一个中断通知单片机接收数据, DSP2 通过缓存串口 2 输出数据。

### 3.1.3 DSP 引导装载程序

在 DSP 程序调试过程中,可以利用 DSP 提供的 JTAG 接口,它可使 PC 机中的 DSP 软件开发环境与 DSP 芯片通信。对多片 DSP 系统的 JTAG 接口需要以菊花链的形式连接。由于 MP3 编码器属于嵌入式系统,必须脱机运行,所以需要采用引导装载程序<sup>[16]</sup> (Bootloader) 在系统上电复位时把用户程序从外部 ROM 引导到高速 RAM 中,以保证其全速运行。C5402 提供了四种 Bootloader 方式: 8 位或 16 位并行 EPROM 方式、8 位或 16 位并行 I/O 方式、8 位或 16 位串行口方式和处理器通信口 (主端口) HPI 方式。

上电复位后, C5402 检测其 MP / MC 引脚, 如果 MP / MC = 0 则进入 Bootloader 程序。进入 Bootloader 后, 如有 INT2 请求中断, 则进入 HPI 引导方式,



由图 3.3 可见,充分利用了 C5402 提供的两个缓存串口进行数据通信。首先,模拟的音频信号通过一个音频的 A/D 处理器 (Philips 的 UDA1341TS),它的系统时钟 SYSCLK、样本采样时钟 WS、位采样时钟 BCK 都由 DSP 提供。不同的是,对音频编码,时钟是由 DSP1 提供的。它提供的时钟频率是 11.2896MHz,即 DSP1 的输入晶振频率,如图 3.3 虚线所示。对语音编码,则由 DSP2 提供,其频率也是

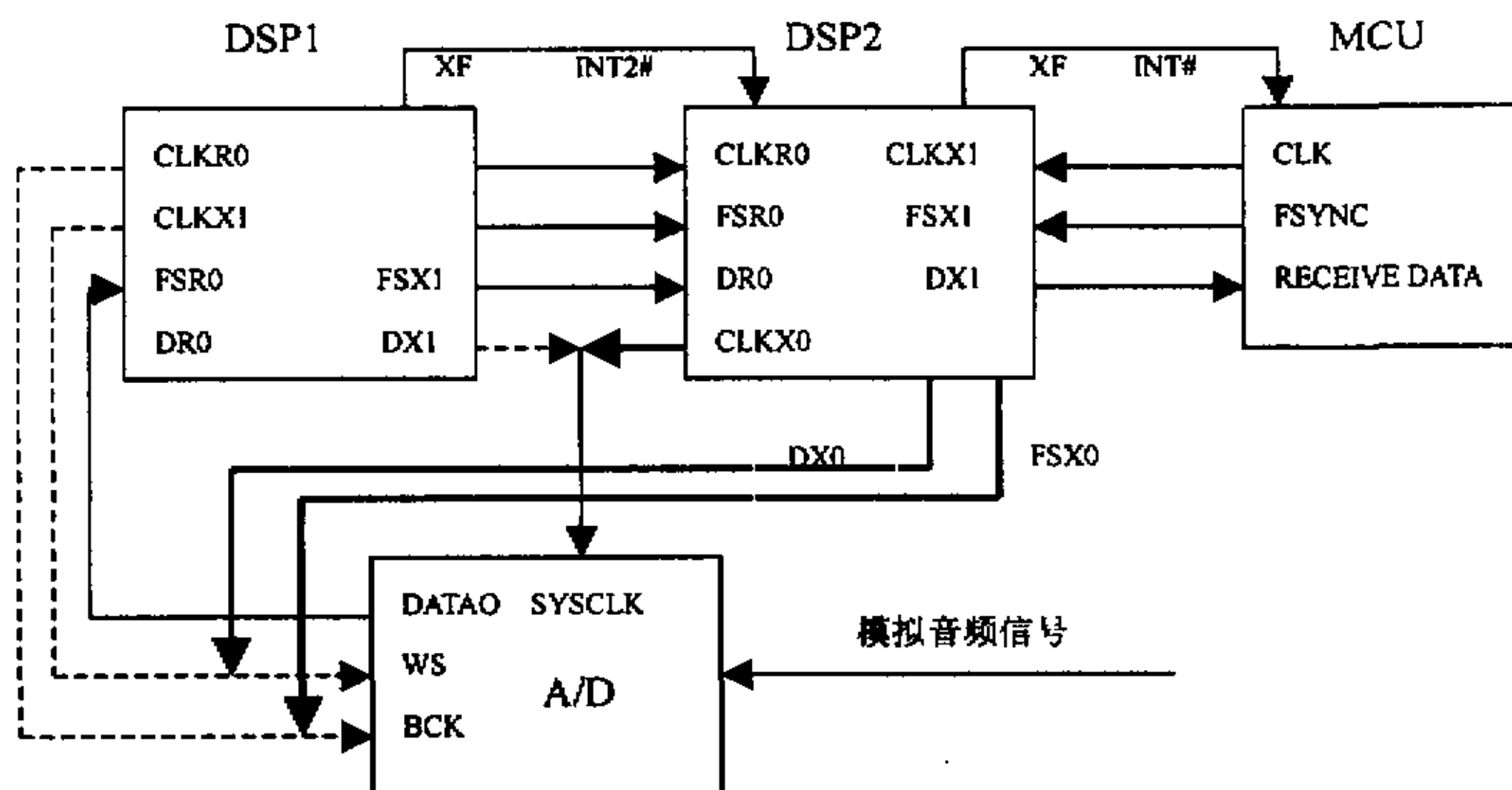


图 3.3 基于 DSP 的系统实现原理框图

DSP2 的晶振输入频率 12MHz,如图 3.3 粗线所示。经过 A/D 转换后的数据通过缓存串口 1 进入 DSP1, DSP1 完成了 MPEG 算法时域到频域的映射和心理声学模型计算。当处理完一帧中每节的数据后就利用 XF 信号产生一个中断给 DSP2,并且从缓存串口 2 输出数据。而 DSP2 通过缓存串口 1 接收数据, DSP2 完成算法中的量化与编码和帧数据流格式化计算。当一帧数据形成后就产生一个中断通知单片机接收数据, DSP2 通过缓存串口 2 输出数据。

### 3.1.3 DSP 引导装载程序

在 DSP 程序调试过程中,可以利用 DSP 提供的 JTAG 接口,它可使 PC 机中的 DSP 软件开发环境与 DSP 芯片通信。对多片 DSP 系统的 JTAG 接口需要以菊花链的形式连接。由于 MP3 编码器属于嵌入式系统,必须脱机运行,所以需要采用引导装载程序<sup>[16]</sup> (Bootloader) 在系统上电复位时把用户程序从外部 ROM 引导到高速 RAM 中,以保证其全速运行。C5402 提供了四种 Bootloader 方式: 8 位或 16 位并行 EPROM 方式、8 位或 16 位并行 I/O 方式、8 位或 16 位串行口方式和处理器通信口 (主端口) HPI 方式。

上电复位后, C5402 检测其 MP / MC 引脚, 如果 MP / MC = 0 则进入 Bootloader 程序。进入 Bootloader 后, 如有 INT2 请求中断, 则进入 HPI 引导方式,

否则就读入地址为 0FFFFH 的引导方式选择字 BRS (Boot Routine Selection) 的内容。BRS=XXXX00 时, 对应 I/O 或串行口引导方式。BRS=XXXX01 时, 进入 8 位 EPROM 引导方式。BRS=XXXX10 时, 进入 16 位 EPROM 引导方式。对于 EPROM 引导方式, 由于 BOOT 程序已经设定好相应的数据存放格式, 因此在 EPROM 中的数据组织就成为关键。下面具体说明 EPROM 的数据组织方法: 假

表 3.1 EPROM 数据组织

数据区地址 (Hex)	内容 (Hex)	含义
8000	08	8 位的 BOOT 标识 (H)
8001	AA	8 位的 BOOT 标识 (L)
8002	7F	SWWSR(H)
8003	FF	SWWSR(L)
8004	F8	BSCR(H)
8005	00	BSCR(L)
8006	00	程序入口 XPC(H)
8007	00	程序入口 XPC(L)
8008	20	程序入口地址 (H)
8009	00	程序入口地址 (L)
800A	04	程序块长度 (H)
800B	00	程序块长度 (L)
800C	00	存放目标 XPC(H)
800D	00	存放目标 XPC(L)
800E	20	存放目标地址 (H)
800F	00	存放目标地址 (L)
8010	XX	程序代码 1 (H)
8011	XX	程序代码 2 (L)
...	...	...
880E	XX	程序代码 N(H)
880F	XX	程序代码 N(L)
8810	00	块结束标志 (H)
8811	00	块结束标志 (L)
...	...	...
FFFE	80	代码存放首地址 (H)
FFFF	00	代码存放首地址 (L)

设存放在 EPROM 中的控制代码和用户代码的首地址为外部数据区的 8000H, 等待状态数为 7 个, 程序执行的入口点地址 2000H, 程序代码长度为 400H, 用户代码存放在片内程序区的首地址为 2000H, 则 EPROM 中的数据组织如表 3.1 所示。表中括号中的 H 表示高 8 位, L 表示低 8 位。

在本系统的实际应用中采用了并行 8 位 Flash ROM 引导方式, 硬件连接框图如图 3.4 所示。

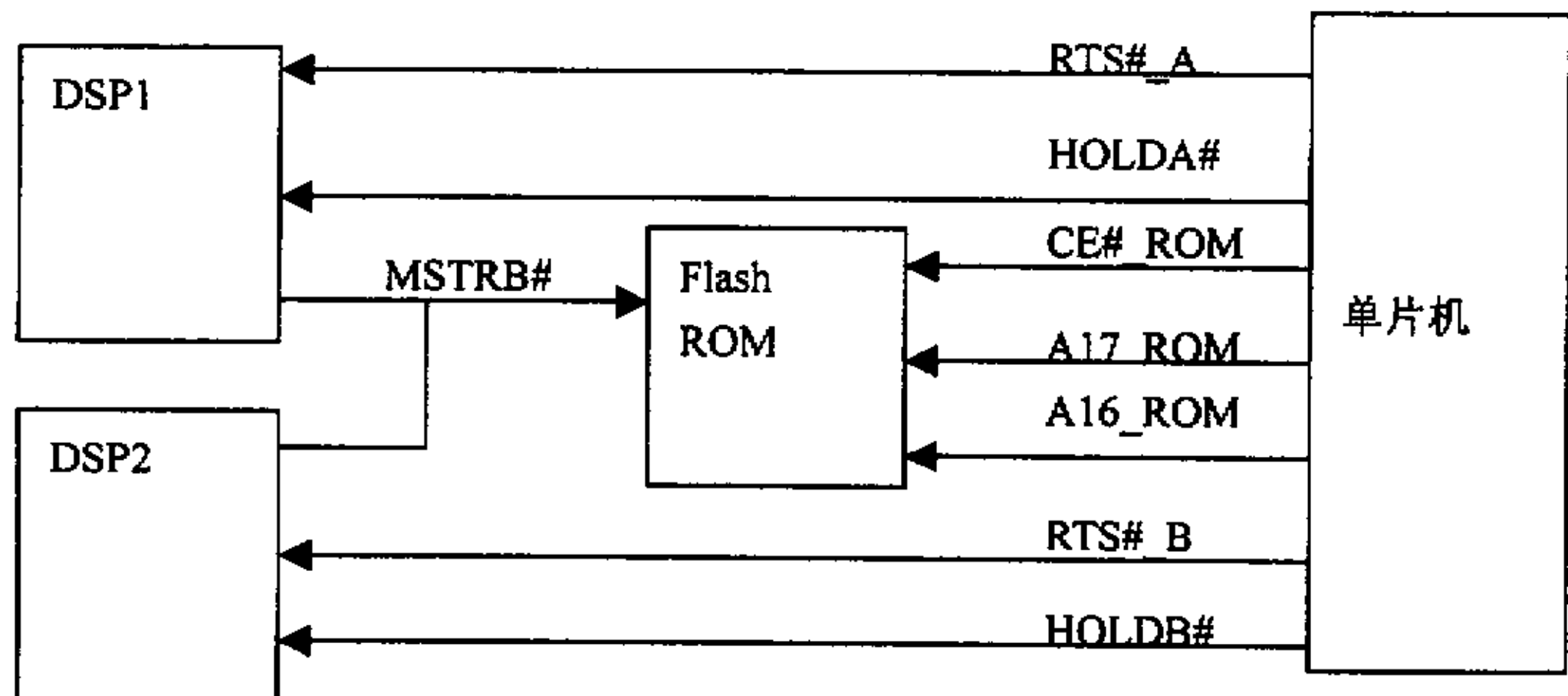


图 3.4 并行 8 位 BOOT 结构硬件框图

利用单片机控制两片 DSP 分别装载引导程序。Flash ROM 选用的是 SST 公司的 2M 比特存储容量的芯片。如表 3-1 所示, DSP 通过访问外部存储器的 8000H 地址来装载用户程序代码, 所以我们可以利用单片机控制 Flash ROM 的最高的两位地址线来选择语音编码和音频编码两种情况下两片 DSP 的不同存放地址, 如表 3.2 所示。

表 3.2 DSP 代码存放地址

		CE#_ROM	A17_ROM	A16_ROM
未执行装载程序时		1	X	X
语音编码	DSP2	0	1	1
	DSP1	0	0	1
音频编码	DSP2	0	1	0
	DSP1	0	0	0

可知, 语音编码时, DSP1 的代码存放首地址是 38000H, DSP2 是 18000H, 音频编码时 DSP1 的代码存放首地址是 8000H, DSP2 是 28000H。对应于表 3-1 的数据组织, 可以得到 Flash ROM 中的程序代码内容, 如表 3.3。

表 3.3 Flash ROM 中的程序代码内容

语音编码	DSP1	08AA 7FFF F800 0000 2400 0A41 0000 2400 ...
	DSP2	08AA 7FFF F800 0000 2F00 0065 0000 0186 ...
音频编码	DSP1	08AA 7FFF F800 0000 3700 01F8 0000 00BC ...
	DSP2	08AA 7FFF F800 0000 3280 0404 0000 03F8 ...

上面主要介绍了存储 DSP 程序的 Flash ROM 在不同编码情况下的数据安排。由于采用两片 DSP 串行工作, 具体如何将代码程序从 Flash ROM 装载到正确的 DSP 芯片中, 将是下面要讨论的问题。

由于引导装载程序是在芯片上电复位后开始的, 所以充分利用 DSP 芯片提供的复位引脚#RS 和保持引脚#HOLD。在复位过程中如果有#HOLD 信号, 在内部仍然进行正常的复位操作, 但是总线和控制信号处于高阻状态。正是利用 DSP 芯片的这一特性, 通过单片机控制 DSP 的这两个信号引脚来完成引导装载程序, 硬件连线如图 3.4。下面将具体描述单片机控制 DSP 的 BOOT 程序<sup>[17]</sup>。当 DSP 上电以后, 首先对 DSP2 复位, 并且使 DSP1 的#HOLD 信号有效, 这样就可以装载 DSP2 的代码程序, 结束以后, 使 DSP2 的#HOLD 信号有效, 同时给 DSP1 一个复位信号, 同时使它的#HOLD 信号无效, 这样就可以装载 DSP1 的代码程序。图 3.5 示出了装载 DSP 程序的流程图,

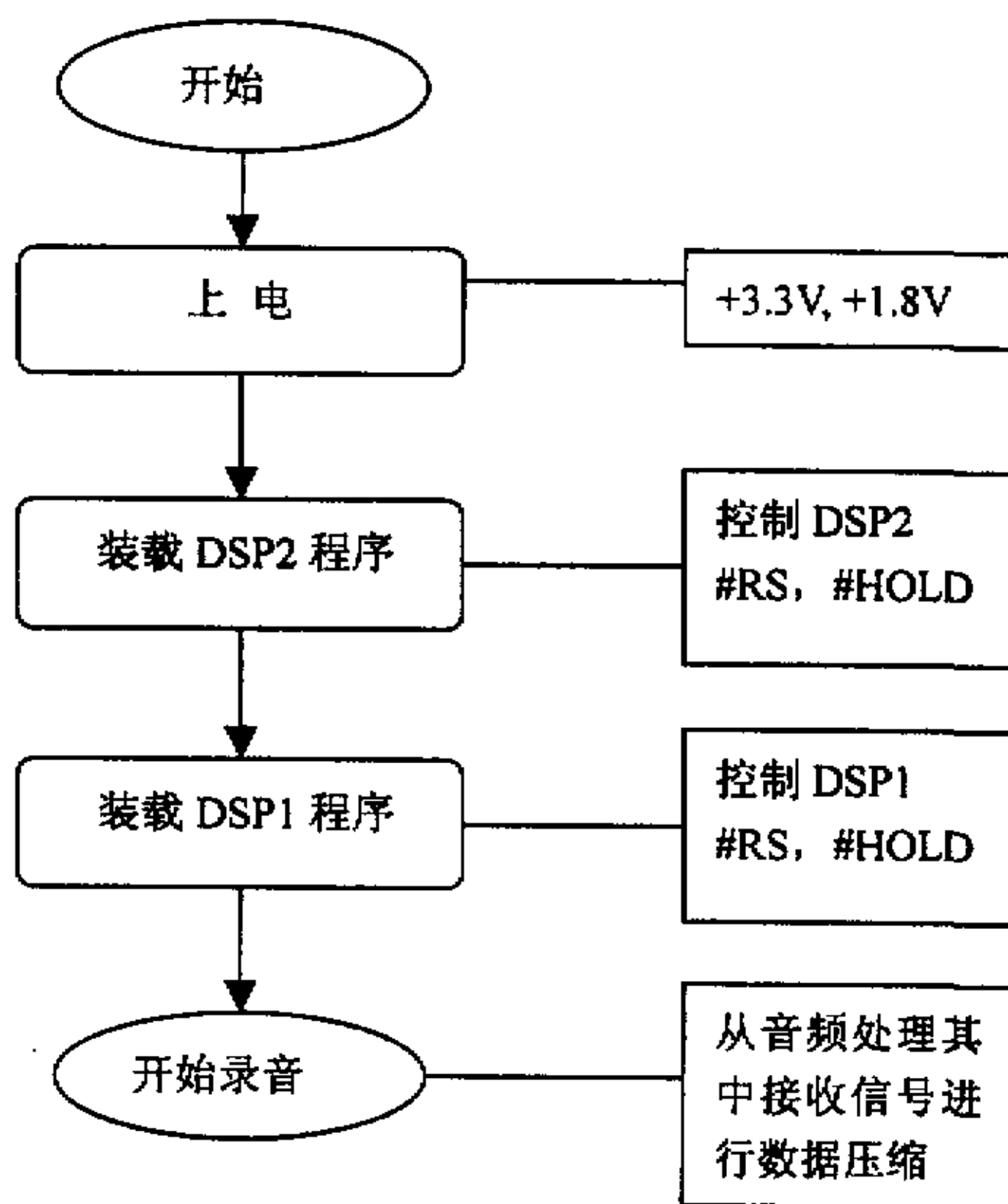
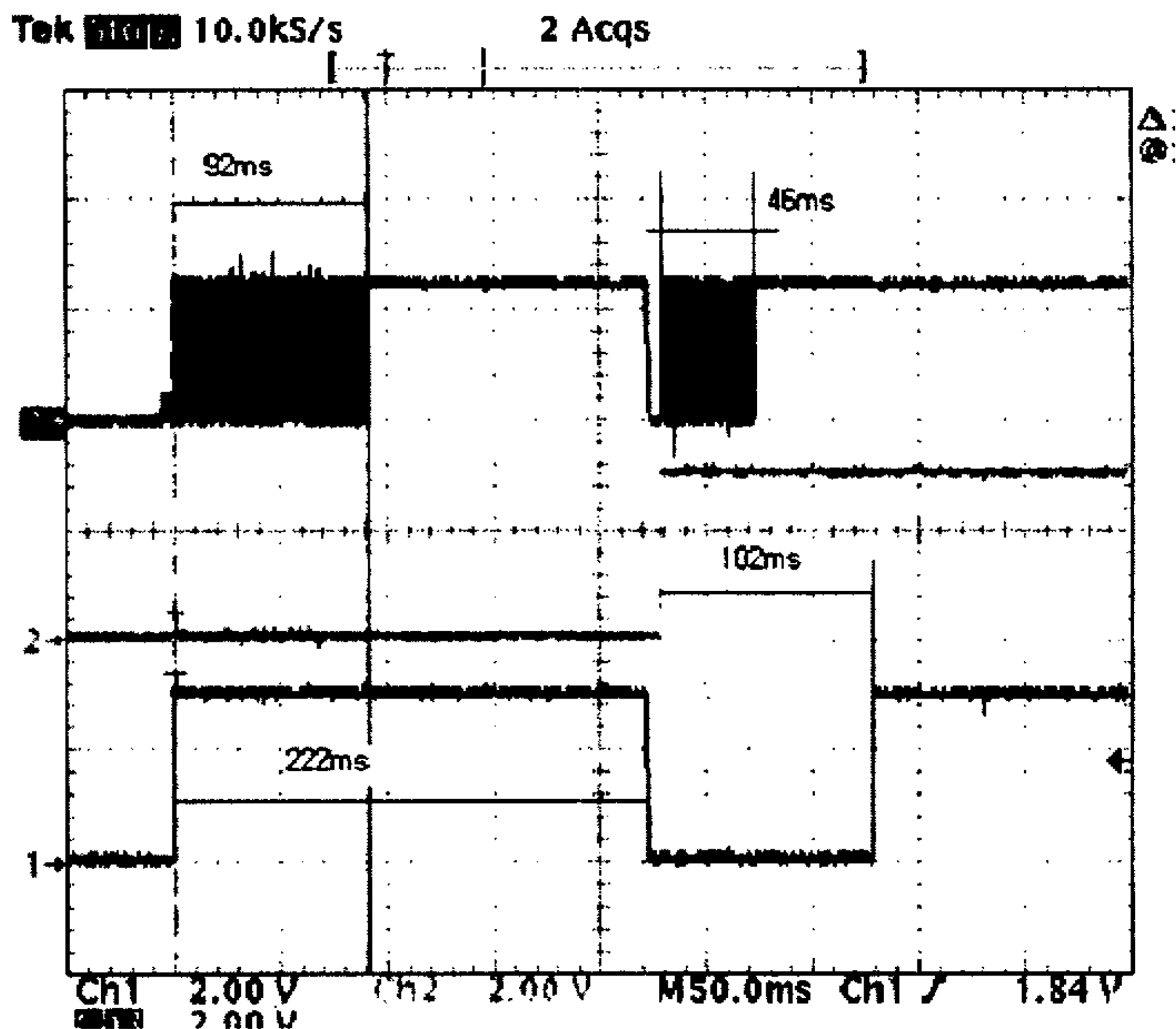


图 3.5 装载 DSP 程序流程图



1: DSP2 #HOLD 2: DSP1 #HOLD 3: Flash #OE

图 3.6 示波器观察到 DSP 的 #HOLD 信号波形

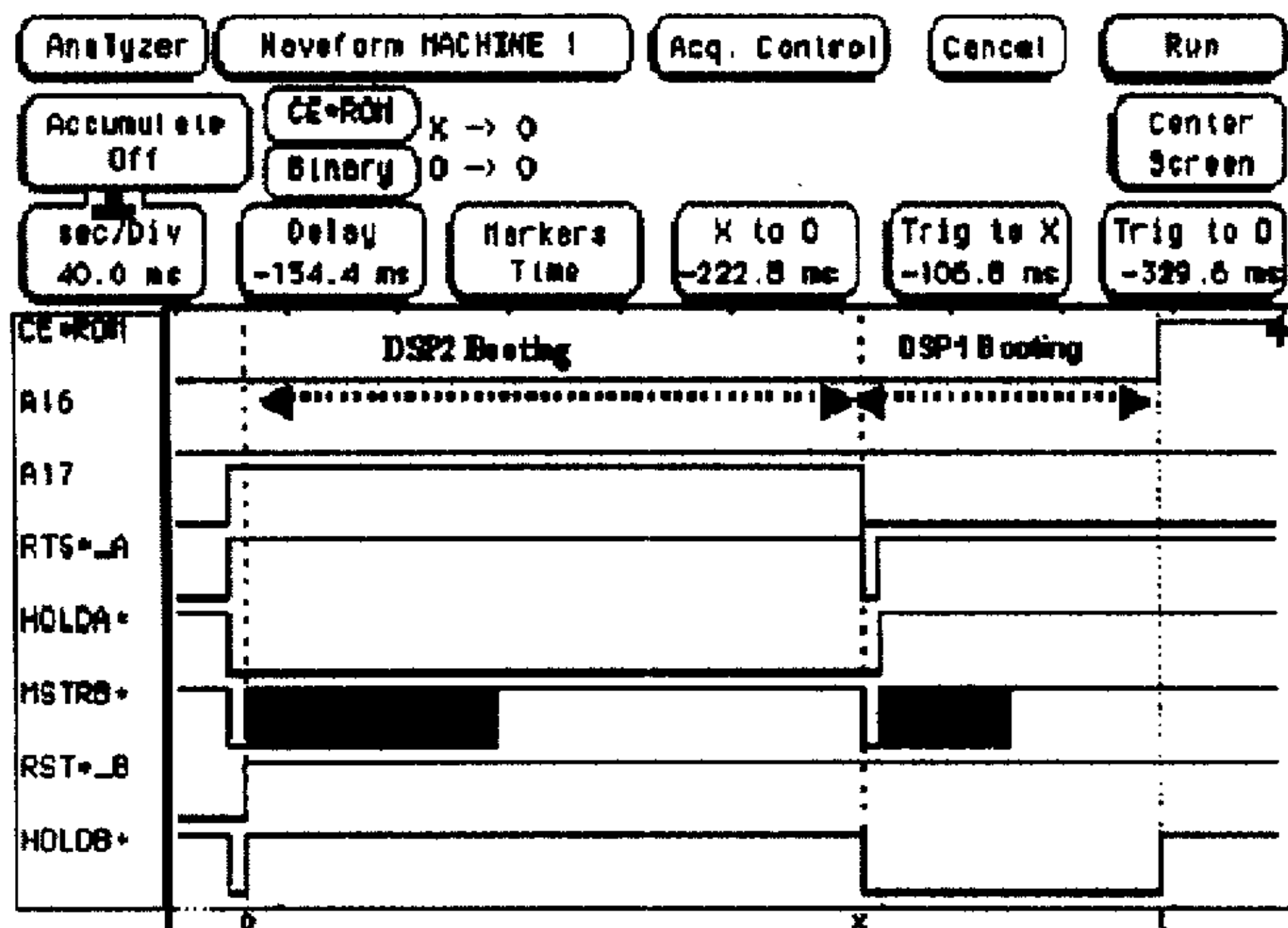


图 3.7 逻辑分析仪观察到的控制 DSP 程序装载的各信号时序波形



两片 DSP 从 Flash ROM 中读取程序总共约需 324ms, 随单片机控制#HOLD 信号的有效时间不同, 可能会略有差别。利用示波器观察其波形, 如图 3.6 所示。可见对于 DSP2, 约需 92ms 装载完毕, 对于 DSP1, 仅需 46ms。通过逻辑分析仪观察, 更清楚地得到 DSP 装载程序时各个信号的时序波形, 如图 3.7 所示, 给出了音频编码时单片机控制 Flash ROM 的信号时序图, 其中包括最高两位的地址线。

## 3.2 接口电路设计

整个编码适配器系统可以分成几个功能模块, 每个模块采用不同的芯片。所以模块之间的接口电路设计是非常关键的, 而正确合适的接口协议是一个系统稳定性的根本保障。本节讨论各接口电路的设计与实现, 包括: 音频处理器和 DSP 之间的接口电路、两片 DSP 之间的接口电路、DSP 与单片机之间的接口电路以及单片机与存储器之间的接口电路。

### 3.2.1 音频处理器与 DSP 之间的接口电路

音频处理器主要是对输入的模拟音频信号进行 A/D 转换, 转成 PCM 格式的数字信号<sup>[18]</sup>送入 DSP1 进行处理。采用 Philips 公司的一款音频处理芯片 UDA1341TS<sup>[19]</sup>, 它的主要特点有: (1) 同时具有 A/D、D/A 功能。(2) 具有双通道录音输入功能。其中一个通道可直接用于 MIC 的输入, 另外一个通道可用于线入录音。(3) 系统时钟有 256Fs, 384Fs, 512Fs 之分, 以便根据具体需要选择。Fs 表示样本的采样频率。(4) 低功耗设计, 输入电压为 3V。(5) 提供 L3 端口, 单片机可通过此端口设置芯片内部寄存器。(6) 多种格式的数据接口, 兼容 I2S 数据格式、高位数据对齐格式(MSB)、低位数据对齐格式(LSB)等。(7) 数据分辨率可以有 16, 18, 20 比特。

二者之间通过串口通信, 可参阅图 3.3。DSP1 通过一个缓存串口接收数据。在音频编码时, 它提供音频处理器所需的系统时钟, 位采样时钟, 样本采样时钟, 数值见表 3.4。其中, 系统时钟等于 256Fs (Fs 指采样频率)。在语音编码时, 最大的区别就是选用 DSP2 提供系统时钟, 系统时钟是 512Fs。

表 3.4 音频处理器信号时钟选择

模式	采样频率	SYSCLK	WS	BCK
音频编码	44.1KHz	11.2896MHz	44.1KHz	1.4112KHz
语音编码	16KHz	8.192MHz	16KHz	512KHz

另外, 由于 DSP 是 16 位定点型, 所以输入数据格式采用 16 位精度, 高位对

两片 DSP 从 Flash ROM 中读取程序总共约需 324ms, 随单片机控制#HOLD 信号的有效时间不同, 可能会略有差别。利用示波器观察其波形, 如图 3.6 所示。可见对于 DSP2, 约需 92ms 装载完毕, 对于 DSP1, 仅需 46ms。通过逻辑分析仪观察, 更清楚地得到 DSP 装载程序时各个信号的时序波形, 如图 3.7 所示, 给出了音频编码时单片机控制 Flash ROM 的信号时序图, 其中包括最高两位的地址线。

## 3.2 接口电路设计

整个编码适配器系统可以分成几个功能模块, 每个模块采用不同的芯片。所以模块之间的接口电路设计是非常关键的, 而正确合适的接口协议是一个系统稳定性的根本保障。本节讨论各接口电路的设计与实现, 包括: 音频处理器和 DSP 之间的接口电路、两片 DSP 之间的接口电路、DSP 与单片机之间的接口电路以及单片机与存储器之间的接口电路。

### 3.2.1 音频处理器与 DSP 之间的接口电路

音频处理器主要是对输入的模拟音频信号进行 A/D 转换, 转成 PCM 格式的数字信号<sup>[18]</sup>送入 DSP1 进行处理。采用 Philips 公司的一款音频处理芯片 UDA1341TS<sup>[19]</sup>, 它的主要特点有: (1) 同时具有 A/D、D/A 功能。(2) 具有双通道录音输入功能。其中一个通道可直接用于 MIC 的输入, 另外一个通道可用于线入录音。(3) 系统时钟有 256Fs, 384Fs, 512Fs 之分, 以便根据具体需要选择。Fs 表示样本的采样频率。(4) 低功耗设计, 输入电压为 3V。(5) 提供 L3 端口, 单片机可通过此端口设置芯片内部寄存器。(6) 多种格式的数据接口, 兼容 I2S 数据格式、高位数据对齐格式(MSB)、低位数据对齐格式(LSB)等。(7) 数据分辨率可以有 16, 18, 20 比特。

二者之间通过串口通信, 可参阅图 3.3。DSP1 通过一个缓存串口接收数据。在音频编码时, 它提供音频处理器所需的系统时钟, 位采样时钟, 样本采样时钟, 数值见表 3.4。其中, 系统时钟等于 256Fs (Fs 指采样频率)。在语音编码时, 最大的区别就是选用 DSP2 提供系统时钟, 系统时钟是 512Fs。

表 3.4 音频处理器信号时钟选择

模式	采样频率	SYSCLK	WS	BCK
音频编码	44.1KHz	11.2896MHz	44.1KHz	1.4112KHz
语音编码	16KHz	8.192MHz	16KHz	512KHz

另外, 由于 DSP 是 16 位定点型, 所以输入数据格式采用 16 位精度, 高位对

两片 DSP 从 Flash ROM 中读取程序总共约需 324ms, 随单片机控制#HOLD 信号的有效时间不同, 可能会略有差别。利用示波器观察其波形, 如图 3.6 所示。可见对于 DSP2, 约需 92ms 装载完毕, 对于 DSP1, 仅需 46ms。通过逻辑分析仪观察, 更清楚地得到 DSP 装载程序时各个信号的时序波形, 如图 3.7 所示, 给出了音频编码时单片机控制 Flash ROM 的信号时序图, 其中包括最高两位的地址线。

## 3.2 接口电路设计

整个编码适配器系统可以分成几个功能模块, 每个模块采用不同的芯片。所以模块之间的接口电路设计是非常关键的, 而正确合适的接口协议是一个系统稳定性的根本保障。本节讨论各接口电路的设计与实现, 包括: 音频处理器和 DSP 之间的接口电路、两片 DSP 之间的接口电路、DSP 与单片机之间的接口电路以及单片机与存储器之间的接口电路。

### 3.2.1 音频处理器与 DSP 之间的接口电路

音频处理器主要是对输入的模拟音频信号进行 A/D 转换, 转成 PCM 格式的数字信号<sup>[18]</sup>送入 DSP1 进行处理。采用 Philips 公司的一款音频处理芯片 UDA1341TS<sup>[19]</sup>, 它的主要特点有: (1) 同时具有 A/D、D/A 功能。(2) 具有双通道录音输入功能。其中一个通道可直接用于 MIC 的输入, 另外一个通道可用于线入录音。(3) 系统时钟有 256Fs, 384Fs, 512Fs 之分, 以便根据具体需要选择。Fs 表示样本的采样频率。(4) 低功耗设计, 输入电压为 3V。(5) 提供 L3 端口, 单片机可通过此端口设置芯片内部寄存器。(6) 多种格式的数据接口, 兼容 I2S 数据格式、高位数据对齐格式(MSB)、低位数据对齐格式(LSB)等。(7) 数据分辨率可以有 16, 18, 20 比特。

二者之间通过串口通信, 可参阅图 3.3。DSP1 通过一个缓存串口接收数据。在音频编码时, 它提供音频处理器所需的系统时钟, 位采样时钟, 样本采样时钟, 数值见表 3.4。其中, 系统时钟等于 256Fs (Fs 指采样频率)。在语音编码时, 最大的区别就是选用 DSP2 提供系统时钟, 系统时钟是 512Fs。

表 3.4 音频处理器信号时钟选择

模式	采样频率	SYSCLK	WS	BCK
音频编码	44.1KHz	11.2896MHz	44.1KHz	1.4112KHz
语音编码	16KHz	8.192MHz	16KHz	512KHz

另外, 由于 DSP 是 16 位定点型, 所以输入数据格式采用 16 位精度, 高位对

齐。图 3.8 表示音频编码时的时钟、数据波形，对于双声道编码时需要同时输入左右两声道的数据。具体工作过程如下：芯片上电以后，单片机通过 L3 端口设置音频处理器内部的寄存器，主要包括系统时钟频率、工作模式（A/D 或 D/A）、录音模式（线入或 MIC 录音）、数据输入格式，分辨率、是否直流滤波以及输入增益等设置。然后启动 DSP 工作，它提供音频处理器所需要的时钟信号，从而使芯片正常工作。

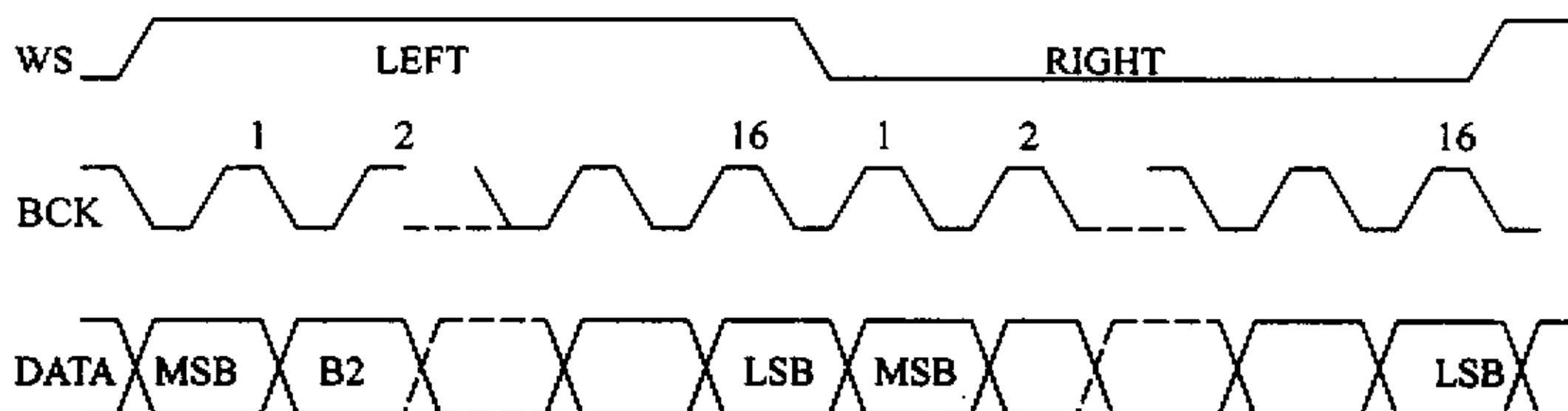


图 3.8 音频编码时音频处理器的时序波形

### 3.2.2 DSP 之间的接口电路

由于使用了两片 DSP 串行工作，所以它们之间的数据通信协议的规定是非常关键的。两片 DSP 分别利用各自的缓存串口进行通信。对于音频编码，DSP1 主要的工作是对接收到的音频数据进行子带编码，同时完成有关计算心理声学模型算法的程序。音频编码的采样率为 44.1KHz，一帧数据包含 1152 个样本数据，所以帧延时约为 26ms。因此，DSP1 需要在 26ms 内完成算法。它的输出包括：（1）经 MDCT 后的频谱矢量；（2）每个比例因子频段允许的最小失真度；（3）长短块选择标志、块类型、混合块标志；（4）平均比特数。其数据流格式如表 3.5 所示。

表 3.5 双通道编码 DSP1 输出一个区组的数据流格式（单通道）

4 比特	4 比特	2 字节	1 字节	1 字	n 字	576 字
------	------	------	------	-----	-----	-------

因为音频编码属于双通道编码，所以算法的实现比较复杂。对于一帧数据的每一节，数据头是 4 比特，如果是左声道，其值是全 1。如果是右声道，其值是全 0。接下来的 4 比特数据分别表示长短块选择标志（1 比特），块类型（2 比特），混合块标志（1 比特）。后面的两个字节表示本区组的长度，以字为单位。紧接着用一个字节的数据表示比例因子段数。接下来一个字表示平均比特位的数值。后面的 n 个字分别表示每个比例因子频段的最小失真度。最后的 576 个字则是一个通道经过 MDCT 变换后的频谱矢量。

当 DSP1 有一节的数据形成后，通过 XF 产生一个中断信号通知 DSP2（见图



齐。图 3.8 表示音频编码时的时钟、数据波形，对于双声道编码时需要同时输入左右两声道的数据。具体工作过程如下：芯片上电以后，单片机通过 L3 端口设置音频处理器内部的寄存器，主要包括系统时钟频率、工作模式（A/D 或 D/A）、录音模式（线入或 MIC 录音）、数据输入格式，分辨率、是否直流滤波以及输入增益等设置。然后启动 DSP 工作，它提供音频处理器所需要的时钟信号，从而使芯片正常工作。

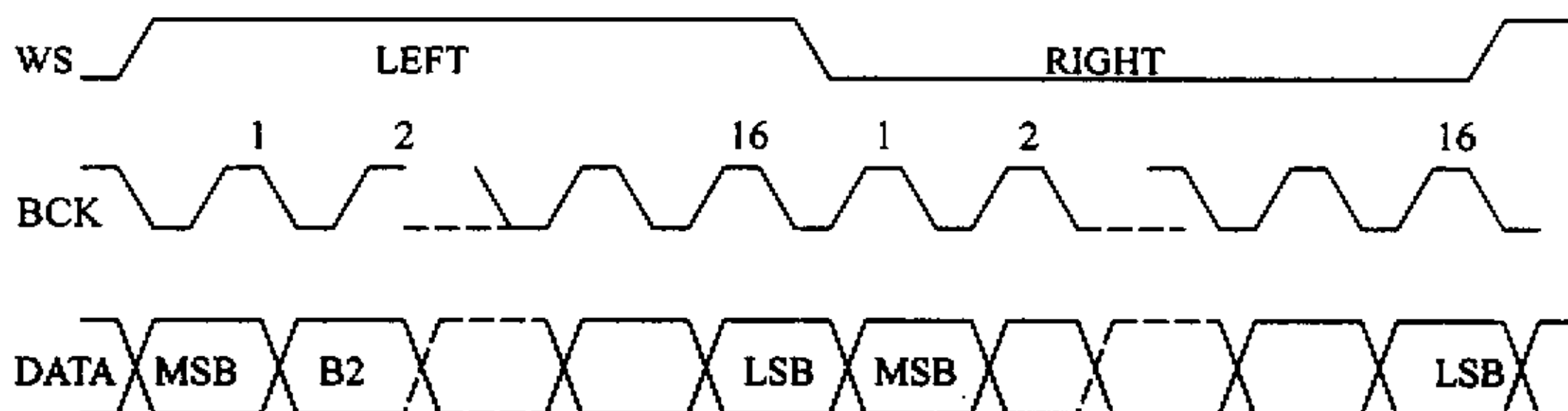


图 3.8 音频编码时音频处理器的时序波形

### 3.2.2 DSP 之间的接口电路

由于使用了两片 DSP 串行工作，所以它们之间的数据通信协议的规定是非常关键的。两片 DSP 分别利用各自的缓存串口进行通信。对于音频编码，DSP1 主要的工作是对接收到的音频数据进行子带编码，同时完成有关计算心理声学模型算法的程序。音频编码的采样率为 44.1KHz，一帧数据包含 1152 个样本数据，所以帧延时约为 26ms。因此，DSP1 需要在 26ms 内完成算法。它的输出包括：（1）经 MDCT 后的频谱矢量；（2）每个比例因子频段允许的最小失真度；（3）长短块选择标志、块类型、混合块标志；（4）平均比特数。其数据流格式如表 3.5 所示。

表 3.5 双通道编码 DSP1 输出一个区组的数据流格式（单通道）

4 比特	4 比特	2 字节	1 字节	1 字	n 字	576 字
------	------	------	------	-----	-----	-------

因为音频编码属于双通道编码，所以算法的实现比较复杂。对于一帧数据的每一节，数据头是 4 比特，如果是左声道，其值是全 1。如果是右声道，其值是全 0。接下来的 4 比特数据分别表示长短块选择标志（1 比特），块类型（2 比特），混合块标志（1 比特）。后面的两个字节表示本区组的长度，以字为单位。紧接着用一个字节的数据表示比例因子段数。接下来一个字表示平均比特位的数值。后面的 n 个字分别表示每个比例因子频段的最小失真度。最后的 576 个字则是一个通道经过 MDCT 变换后的频谱矢量。

当 DSP1 有一节的数据形成后，通过 XF 产生一个中断信号通知 DSP2（见图



3.3), 并产生发送数据的时钟信号和帧同步信号。DSP2 接收到中断信号后, 通过缓存串口的时钟接收信号和帧同步接收信号控制所需的数据信息。其中, DSP1 的发送时钟频率是 DSP1 的晶振频率经过四分频后得到的。DSP 的外部标志输出引脚 XF 可以用于与外部接口器件的握手信号, 可以通过对 ST1 中的 XF 位置位或清除来控制。

另外, 对低采样率语音编码<sup>[20]</sup>, 没有分节处理, 所以是以帧为编码单位的。每当 DSP1 有一帧数据形成以后, 就产生一个中断通知 DSP2 接收数据。由于语音编码的采样率和编码比特率都是比较低的, 所以对于编程实现来说, 它的实时性要求不是很高, 容易实现。

### 3.2.3 DSP 与单片机之间的接口电路

DSP2 的主要任务是完成算法中关于三层迭代模型计算以及码流格式的规范化, 并把标准 MP3 格式的数据以帧为单位通过缓存串口的数据输出端传送给单片机。其中, 利用单片机的 SPI(Serial Peripheral Interface)来接收数据, SPI 有三条信号线, 即时钟信号、数据输入信号和数据输出信号。利用时钟信号线提供 DSP2 的串口时钟, 同时通过软件模拟一个可编程端口来提供 DSP2 的帧同步信号, 利用数据输入信号来接收数据, 如图 3.9 所示。

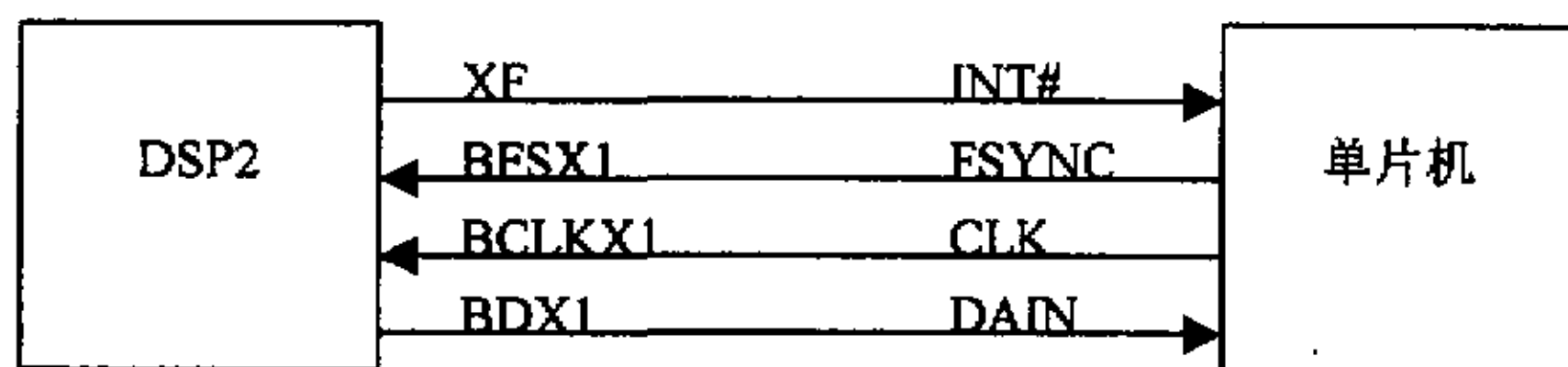


图 3.8 DSP2 与单片机的接口电路

首先介绍工作流程, 当 DSP2 有一帧数据产生以后, 它通过设置 XF 信号为低对单片机产生一个中断。单片机接收到中断信号以后, 将同时发送时钟信号和帧同步信号, DSP2 则在每个时钟的下降沿传送数据 (以字节位单位)。在接收一个字节之前, 首先设置帧同步信号为低电平, 然后时钟信号连续发出 8 个时钟脉冲, 从而接收一个字节的的数据, 完成以后, 帧同步信号在设置为高电平。图 3.9 是用示波器观察到的单片机接收一个字节数据的信号波形图。

DSP2 的数据发送时钟由单片机提供。由于音频编码的一帧数据是 417 字节, 再加上一些控制信息数据, 则每接收一帧数据, 单片机都需要连续发送 431 字节的时钟信号以接收数据。待一帧数据接收完后, 帧同步信号将会设置为高电平, 当 DSP2 接收到这一信息后, 它将准备下一帧数据的发送。图 3.10 给出了单片机接收一帧数据的信号波形图。

从图 3.10 可以看出, 传送两帧数据的间隔大约是 26ms, 即一帧数据的时延。当帧同步信号变为恒定的高电平时, 表示从 DSP2 中接收一帧数据完毕, 单片机要在下一个中断到来之前, 将这帧数据经过一定处理后存放到存储器中。由于实时性要求比较高, 单片机提供给 DSP2 的串口时钟频率大于 2.5MHz, 这样才能保证在下一帧数据到来之前, 当前帧的数据被正确存入到存储器中。

为了保证从 DSP2 接收数据的正确率, 接收到的数据要按照一定格式排列。对于上面提到一帧 431 个字节的数据如图 3.11 所示, 最先接收到的两个字节属无效字节, 一般情况下它的值是 00H、A1H, 这样有利于单片机系统的调试。接下来的两个字节表明了此帧序列的长度。对于双通道编码来说, 其值一般是 A1H(低位字节)、01H(高位字节), 即 417 字节。同时为了避免错误, 单片机判断接收到的这两个字节数据, 如果值大于 430 或小于 16, 都将被认为是无效帧, 从而舍去此帧数据, 重新接收下一帧数据。在 16KHz 采样频率, 16Kbit/s 码速率的语音编码时, 它的值通常是 72 字节。单片机接收到的第五个字节相当于帧序列号, 第一帧为 00H, 接下来依次类推, 如 01H、02H、...FFH、00H、01H 等。这一字节对于编码数据无任何影响, 可以不考虑。真正的 MP3 数据流是从第 6 个字节开始, 在调试过程中, 根据这些协议可以很方便地判断接收到的数据是否有错。

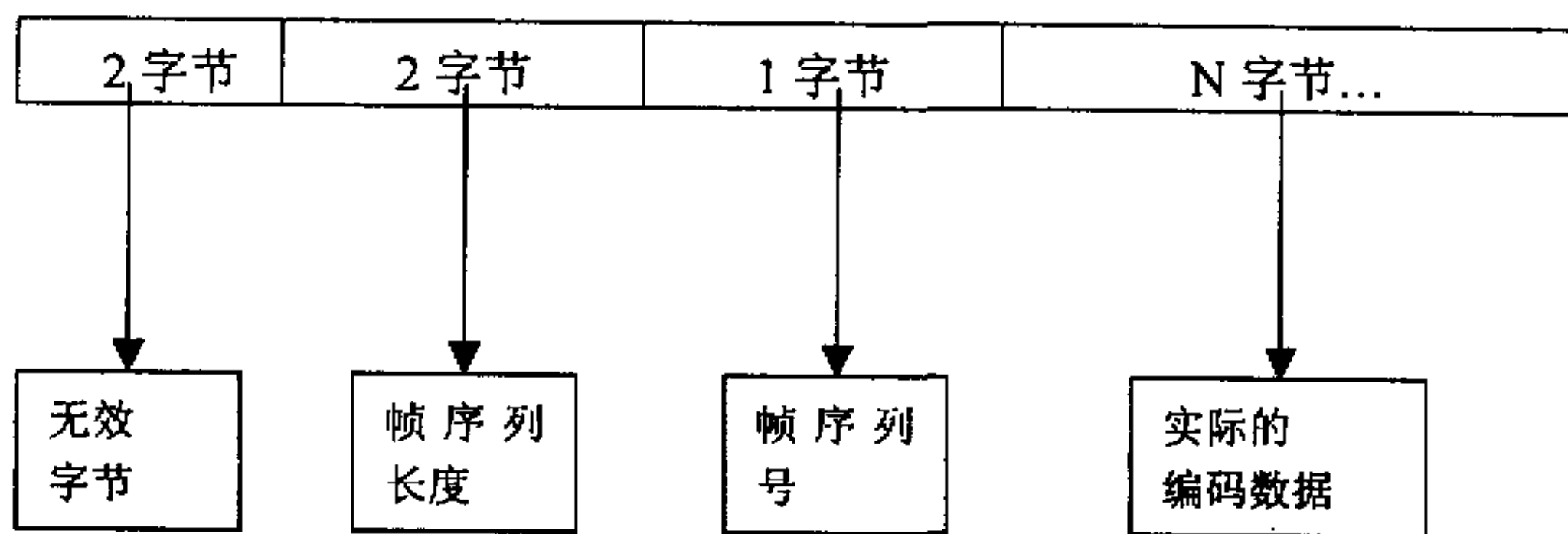


图 3.11 单片机接收帧数据序列格式

#### 3.2.4 单片机与存储器之间的接口电路

被编码的数据必然要存储到相应的存储设备中, 当前针对 MP3 歌曲的存储器很多, 但主要是使用外扩的 Flash 卡, 主要产品是 SMC(SmartMediaCard)和 MMC(MultiMediaCard)。但从各个商家使用的 Flash 卡总体情况及设计的方便性和可扩充能力来看, SMC 最有希望成为未来的主流。目前东芝, 三星等公司生产的 SMC<sup>[21]</sup>容量最高已经可以达到 128M 字节。本系统采用三星生产的 SMC 芯片作为存储设备。

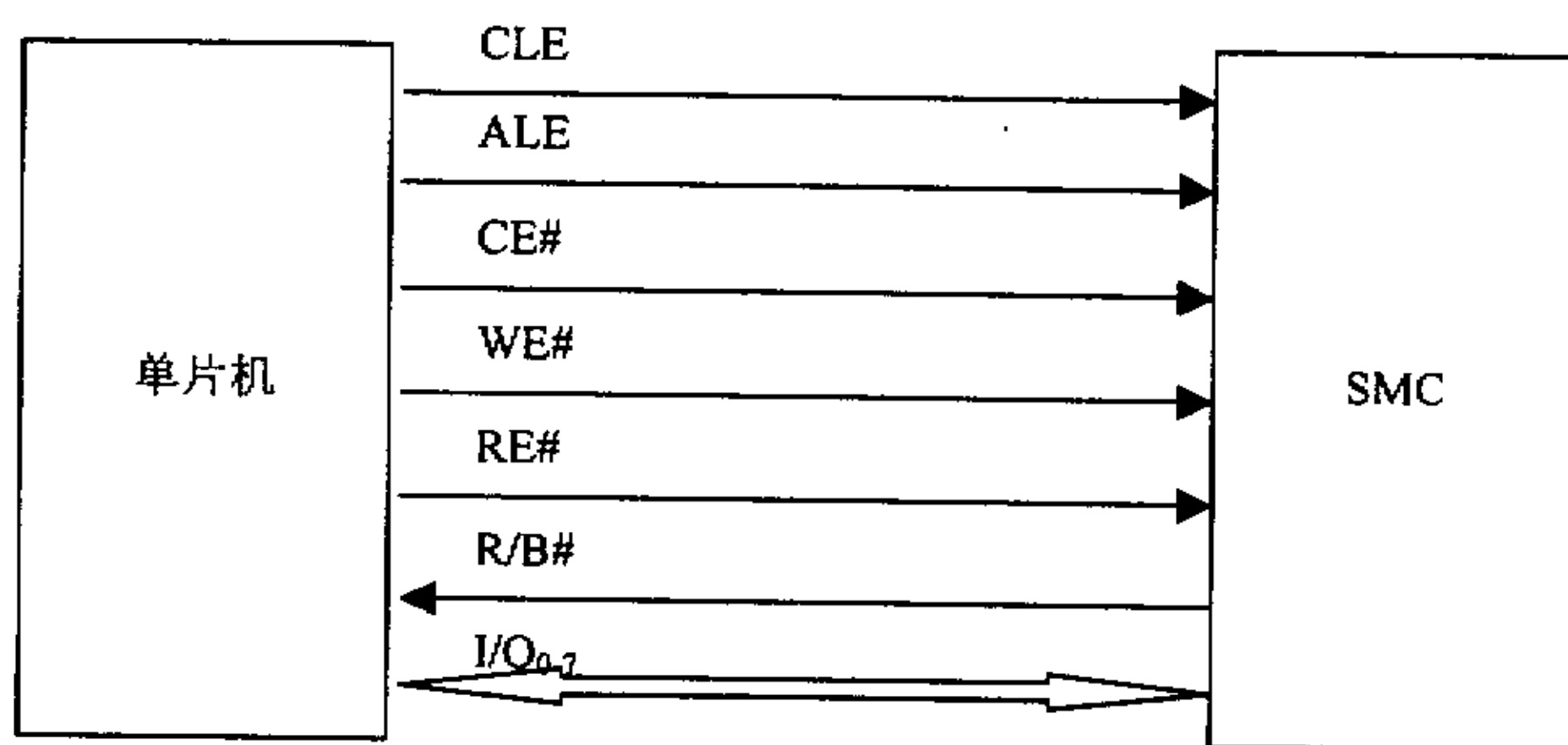


图 3.12 单片机与存储器 SMC 之间的接口电路

图 3.12 所示为单片机与 SMC 之间的接口电路。其中，信号 CLE 有效时表示此时数据线输出的是命令字，例如读、写、擦除等命令。信号 ALE 有效时表示此时数据线输出的是 SMC 的地址，通过三个连续的脉冲信号输出 24 位地址。CE# 是芯片始能信号。WE# 表示写有效，RE# 表示读有效，R/B# 提供 SMC 所处的状态。SMC 要求写数据时必须以块为单位写入，它提供的每块的大小是 528 字节。为了使数据有较高的正确率，通常给每块写入的数据是 512 字节。

下面主要介绍 SMC 存储文件的格式。因为 Flash 卡的文件存储格式没有统一的标准，所以我们自己定义了 SMC 内部的文件存储格式。MP3 录放机中，SMC 中的文件必须按照一定的格式存放，系统才能对之进行快速的查找和读取。存放系统数据和数据文件格式的区域，被称为文件分配表 FAT(File Allocation Table)。

MP3 录放机中的 FAT 使用类似 ISO9660 标准的方式。文件数据采用扇区数据链的方式存放，NAND<sup>[21]</sup>型 FLASH 的每个扇区为 16K 字节，每个文件所占据的扇区在文件系统区标出。以 32M 字节容量的 SMC 为例，整个存储空间可以被分为 2K 扇区。其中扇区 0 是系统数据区域，存储有关存储器的总容量、生产商、生产日期、是否被初始化、内存总扇区数、坏扇区标志区域等信息。扇区 1 是文件系统数据区域，包含已存文件的总数目、总的播放时间、剩余内存扇区数等信息。扇区 2 和扇区 3 被定义为数据扇区标志区域，它用两个字节来标志一个扇区的存储状况。如果它的值是 FFFFH，则表示此扇区为空，否则它的值代表下一个链节扇区的位置。如果它的值是 0000H，则表示它是一个文件的最后一个区域，即文件结束标志。扇区 4 和扇区 5 是空闲扇区数据标志区域，每次录音文件前，单片机将空闲扇区号顺序标注在空闲扇区区域，每两个字节标注一个空闲扇区的绝对位置，然后根据其位置将录入的数据写入空扇区中。扇区 6 到扇区 13 属于文件头信息区域，表示每一首歌曲的头信息，例如一个文件的文件名、文件长度、

播放时间等, 见表 3.6。标明一个文件的头信息需要 512 字节的空间, 所以对于 32K 大小的文件头信息区域来说, 一共可以记录 256 个文件的信息。扇区 14 和扇区 15 用来做数据备份区, 供单片机转移、改写数据时使用。文件的数据区域是从扇区 16 开始纪录的。

表 3.6 文件头信息区域记录格式

扇区号	起始地址	数据长度 (字节)	数据内容	数据存放 格式	说明
6	18000H	256	文件名 1	Char	文件信息 区域 512 字节
	18100H	4	文件长度	DWord	
	18104H	2	该文件所占据 扇区数目	Word	
	18106H	2	该文件所在第一个扇区的 序号	Word	
	18108H	2	该文件所在第一个扇区的 起始地址(L→H)	Word	
	1810AH	2	歌词语种: 0->英文, 1->汉文 2->汉文, 3->日文	Word	
	1810CH	4	歌词文件长度(L→H) (为 0 表示无歌词文件)	DWord	
	18110H	2	歌词文件开始扇区号	Word	
	18112H	4	播放时间 (hour min sec rev)	Byte	
	18116H	2	比特率(L→H) (Byte/s)	Word	
	18118H	1	文件类型: 00H—MP3 01H—ADPCM FFH—普通文件	char	
	18119H	8	录制日期 (年 月 日)	Char	
	18121H	223	保留		
6~13	18200H	256	文件名 2	Char	文件信息 区域
	18300H	4	文件长度	DWord	
	18304H	2	该文件所占据 扇区数目	Word	
	...	...	...	...	
	37FFFH	省 略 (总共可以存放 256 个文件的头信息, 文件信息区域共占据 128K 字节)			

在录音过程中, 单片机首先通过读取剩余扇区数, 确定录制时间。然后通过从数据扇区标志区域读取所有的空闲扇区, 并将它们的序号由前到后逐个写入临

时数据扇区标志区域。完成此项工作后,将文件名和开始录音的首扇区号写入文件信息区。之后,单片机将首扇区号和偏移地址 0x0000 写入相应的寄存器当中,并且启动 DSP 开始录音,随后将得到的数据根据首扇区号写入。每当一个扇区写完后,同时查找空扇区地址,并且将此扇区号写入当前标志扇区区域中的相应位置,以便构成文件的链表索引。如果录音结束,则在标志扇区中的当前扇区标志中写入 0000H,表示文件结束。录音结束后,单片机可以计算出当前录音文件的长度,并根据该长度修改文件信息(可直接写入)和系统信息(利用数据备份区改写)。

### 3.3 单片机控制模块

单片机是整个适配器的控制部分,系统其它功能模块都是在它的指挥协调下完成。

#### 3.3.1 单片机的性能特点

选用一款合适的单片机作为整个嵌入式系统的控制部分,需要考虑多方面因素。例如它的指令运行周期、存储器大小、接口电路、功耗和价格等。本系统选择了富士通公司的 MB89560<sup>[22]</sup>,它是用先进的 COMS 工艺制造的高性能 8 位单片机,该公司的 COMS 技术把 HMOS 的高速、高密度特性和 CMOS 的低功耗的特点结合在一起,使器件具有优越的性能,其主要特点如下:

- F<sup>2</sup>MC-8L CPU 核;
- 低工作电压: 2.2~3.6V;
- 主工作频率 10MHz,指令运行周期 0.4us,另外还有一个频率为 32.768KHz 的从工作时钟;
- 内部 ROM 32K\*8 位,内部 RAM 1K\*8 位;
- 50 个可编程 I/O 端口;
- 多种类型的定时器:双通道的 8 位计数定时器,单通道的 16 位计数定时器,单通道的 8 位 PWM 定时器;
- 12 个外部中断源,3 个中断优先级;
- 8 个通道的 10 位 A/D 转换功能;
- 可编程时钟输出;
- 具有 I2C 接口功能;
- UART/SPI 串行接口;
- 具有段式 LCD 驱动控制功能,最大可编程 96 段 LCD;
- 可分为掩膜,OTP 两种类型,80 管脚 LQFP 封装。



时数据扇区标志区域。完成此项工作后,将文件名和开始录音的首扇区号写入文件信息区。之后,单片机将首扇区号和偏移地址 0x0000 写入相应的寄存器当中,并且启动 DSP 开始录音,随后将得到的数据根据首扇区号写入。每当一个扇区写完后,同时查找空扇区地址,并且将此扇区号写入当前标志扇区区域中的相应位置,以便构成文件的链表索引。如果录音结束,则在标志扇区中的当前扇区标志中写入 0000H,表示文件结束。录音结束后,单片机可以计算出当前录音文件的长度,并根据该长度修改文件信息(可直接写入)和系统信息(利用数据备份区改写)。

### 3.3 单片机控制模块

单片机是整个适配器的控制部分,系统其它功能模块都是在它的指挥协调下完成。

#### 3.3.1 单片机的性能特点

选用一款合适的单片机作为整个嵌入式系统的控制部分,需要考虑多方面因素。例如它的指令运行周期、存储器大小、接口电路、功耗和价格等。本系统选择了富士通公司的 MB89560<sup>[22]</sup>,它是用先进的 COMS 工艺制造的高性能 8 位单片机,该公司的 COMS 技术把 HMOS 的高速、高密度特性和 CMOS 的低功耗的特点结合在一起,使器件具有优越的性能,其主要特点如下:

- F<sup>2</sup>MC-8L CPU 核;
- 低工作电压: 2.2~3.6V;
- 主工作频率 10MHz,指令运行周期 0.4us,另外还有一个频率为 32.768KHz 的从工作时钟;
- 内部 ROM 32K\*8 位,内部 RAM 1K\*8 位;
- 50 个可编程 I/O 端口;
- 多种类型的定时器:双通道的 8 位计数定时器,单通道的 16 位计数定时器,单通道的 8 位 PWM 定时器;
- 12 个外部中断源,3 个中断优先级;
- 8 个通道的 10 位 A/D 转换功能;
- 可编程时钟输出;
- 具有 I2C 接口功能;
- UART/SPI 串行接口;
- 具有段式 LCD 驱动控制功能,最大可编程 96 段 LCD;
- 可分为掩膜,OTP 两种类型,80 管脚 LQFP 封装。

时数据扇区标志区域。完成此项工作后,将文件名和开始录音的首扇区号写入文件信息区。之后,单片机将首扇区号和偏移地址 0x0000 写入相应的寄存器当中,并且启动 DSP 开始录音,随后将得到的数据根据首扇区号写入。每当一个扇区写完后,同时查找空扇区地址,并且将此扇区号写入当前标志扇区区域中的相应位置,以便构成文件的链表索引。如果录音结束,则在标志扇区中的当前扇区标志中写入 0000H,表示文件结束。录音结束后,单片机可以计算出当前录音文件的长度,并根据该长度修改文件信息(可直接写入)和系统信息(利用数据备份区改写)。

### 3.3 单片机控制模块

单片机是整个适配器的控制部分,系统其它功能模块都是在它的指挥协调下完成。

#### 3.3.1 单片机的性能特点

选用一款合适的单片机作为整个嵌入式系统的控制部分,需要考虑多方面因素。例如它的指令运行周期、存储器大小、接口电路、功耗和价格等。本系统选择了富士通公司的 MB89560<sup>[22]</sup>,它是用先进的 COMS 工艺制造的高性能 8 位单片机,该公司的 COMS 技术把 HMOS 的高速、高密度特性和 CMOS 的低功耗的特点结合在一起,使器件具有优越的性能,其主要特点如下:

- F<sup>2</sup>MC-8L CPU 核;
- 低工作电压: 2.2~3.6V;
- 主工作频率 10MHz,指令运行周期 0.4us,另外还有一个频率为 32.768KHz 的从工作时钟;
- 内部 ROM 32K\*8 位,内部 RAM 1K\*8 位;
- 50 个可编程 I/O 端口;
- 多种类型的定时器:双通道的 8 位计数定时器,单通道的 16 位计数定时器,单通道的 8 位 PWM 定时器;
- 12 个外部中断源,3 个中断优先级;
- 8 个通道的 10 位 A/D 转换功能;
- 可编程时钟输出;
- 具有 I2C 接口功能;
- UART/SPI 串行接口;
- 具有段式 LCD 驱动控制功能,最大可编程 96 段 LCD;
- 可分为掩膜,OTP 两种类型,80 管脚 LQFP 封装。

由于 MB89560 具有上述特点, 所以可以很方便地将它用于此适配器系统中, 并且充分利用它的各种硬件资源。而且富士通公司提供了此单片机系列的仿真系统和 PC 仿真开发环境, 这样, 仅仅利用 C 编程语言就可完成整个单片机的控制功能程序的编写, 不仅缩短开发周期, 提高开发效率, 而且可以保证系统很好的稳定性。

### 3.3.2 单片机的控制功能

适配器系统中的其它模块都是在单片机的驱动或控制下运行的, 下面介绍单片机在各模块中完成的功能。

#### (1) 单片机控制音频处理器

本章第二节提到, 音频处理器 UDA1341TS 有一个 L3 控制接口。它有三条控制线: MODE、CLOCK、DATA, 单片机通过这个接口可以读写音频处理器的内部寄存器, 从而设置它的各项功能。

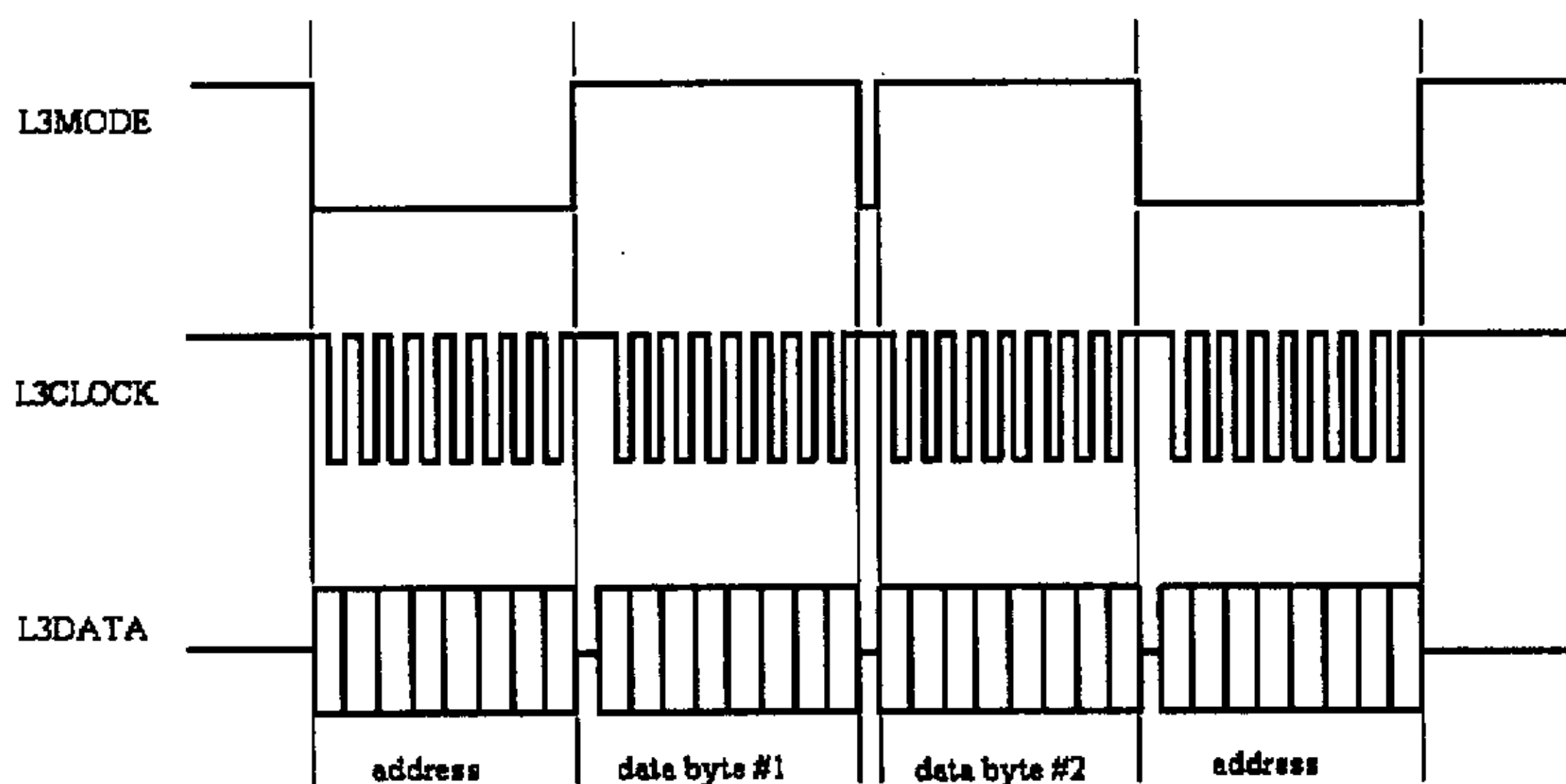


图 3.14 音频处理器 L3 接口信号时序图

如图 3.14, 当 MODE 是低电平时, 输入一个 8 字节的地址数据, 不同的地址控制不同的功能设置。当地址是 DATA0 时, 紧接着当 MODE 变高后, 写入的一个或两个字节的的数据设置 A/D 或 D/A 的增益、输出音量、静音等功能。当地址是 DATA1 时, 单片机此时可以读出当前寄存器的值, 从而判断音频处理器的当前状态。当地址是 STATUS 时, 写入的数据设置系统采样时钟、数据输入格式、分辨率、工作模式 (A/D 或 D/A 转换) 以及音频格式的各项指标等。

#### (2) 单片机控制 LCD 显示

由于单片机具有段式 LCD 的驱动功能, 所以很方便地设计出如图 3.15 所示的 LCD 显示图案。单片机内部 RAM 中, 有 12 字节专用于表示 96 段 LCD 显示,

由于 MB89560 具有上述特点, 所以可以很方便地将它用于此适配器系统中, 并且充分利用它的各种硬件资源。而且富士通公司提供了此单片机系列的仿真系统和 PC 仿真开发环境, 这样, 仅仅利用 C 编程语言就可完成整个单片机的控制功能程序的编写, 不仅缩短开发周期, 提高开发效率, 而且可以保证系统很好的稳定性。

### 3.3.2 单片机的控制功能

适配器系统中的其它模块都是在单片机的驱动或控制下运行的, 下面介绍单片机在各模块中完成的功能。

#### (1) 单片机控制音频处理器

本章第二节提到, 音频处理器 UDA1341TS 有一个 L3 控制接口。它有三条控制线: MODE、CLOCK、DATA, 单片机通过这个接口可以读写音频处理器的内部寄存器, 从而设置它的各项功能。

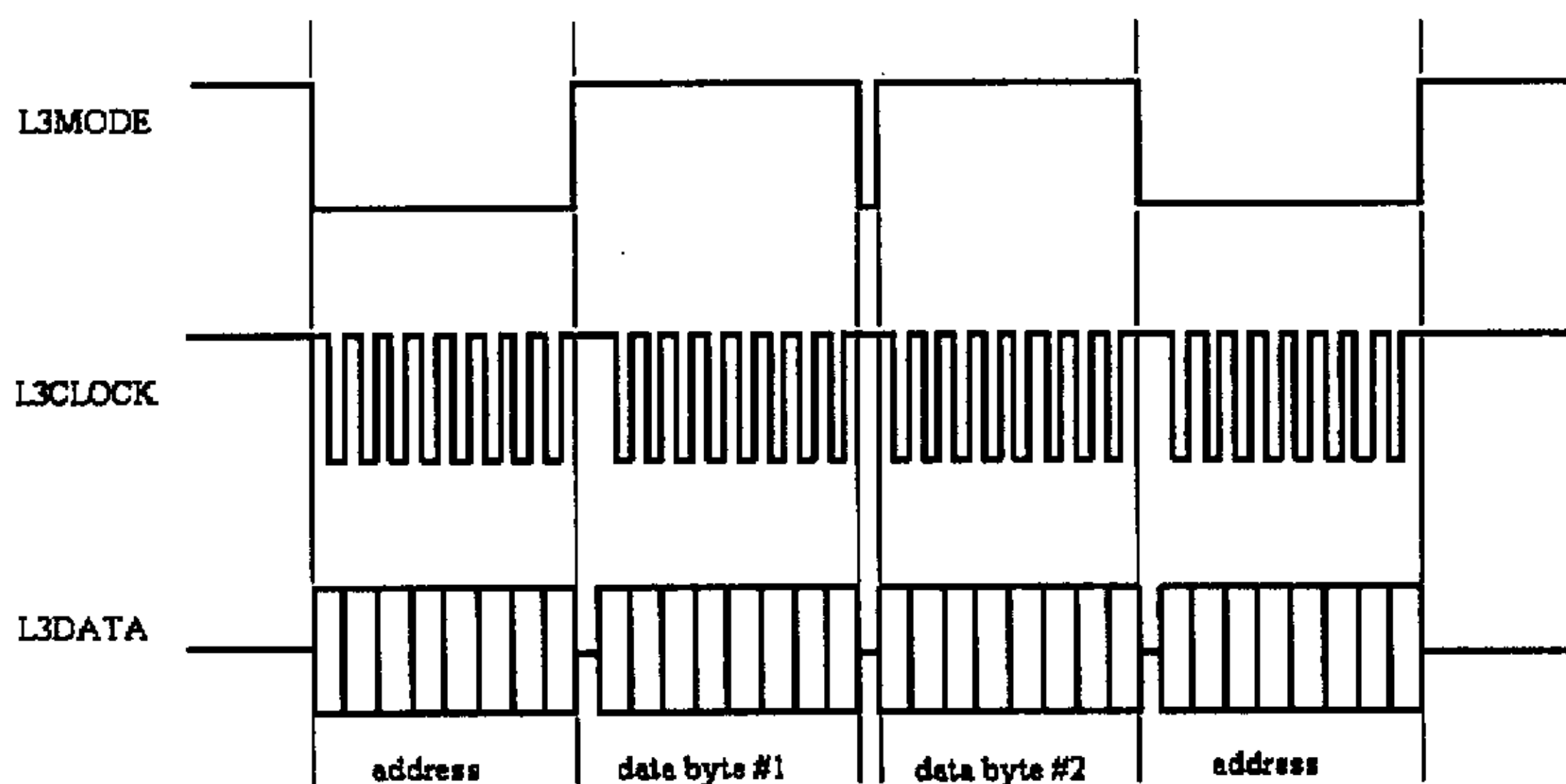


图 3.14 音频处理器 L3 接口信号时序图

如图 3.14, 当 MODE 是低电平时, 输入一个 8 字节的地址数据, 不同的地址控制不同的功能设置。当地址是 DATA0 时, 紧接着当 MODE 变高后, 写入的一个或两个字节的的数据设置 A/D 或 D/A 的增益、输出音量、静音等功能。当地址是 DATA1 时, 单片机此时可以读出当前寄存器的值, 从而判断音频处理器的当前状态。当地址是 STATUS 时, 写入的数据设置系统采样时钟、数据输入格式、分辨率、工作模式 (A/D 或 D/A 转换) 以及音频格式的各项指标等。

#### (2) 单片机控制 LCD 显示

由于单片机具有段式 LCD 的驱动功能, 所以很方便地设计出如图 3.15 所示的 LCD 显示图案。单片机内部 RAM 中, 有 12 字节专用于表示 96 段 LCD 显示,

所以可以根据事先列好的真值表来具体编程。LCD 在录音时需要同步显示录音时间, 通过 6 个 7 段式数字格式分别显示录音时、分、秒。同时需要显示音频录音还是语音录音, 图中五线谱代表音频录音, 人头代表语音录音。音频录音时的码速率是 128kbps, 语音录音时的码速率是 16kbps。另外, 可以显示当前录音数据是存放在 MP3 播放器内置的 Flash 存储器中, 还是外插的 SMC 中。同时利用单片机的 A/D 转换通道间隔采样电源电压, 来及时显示电池耗电状况。

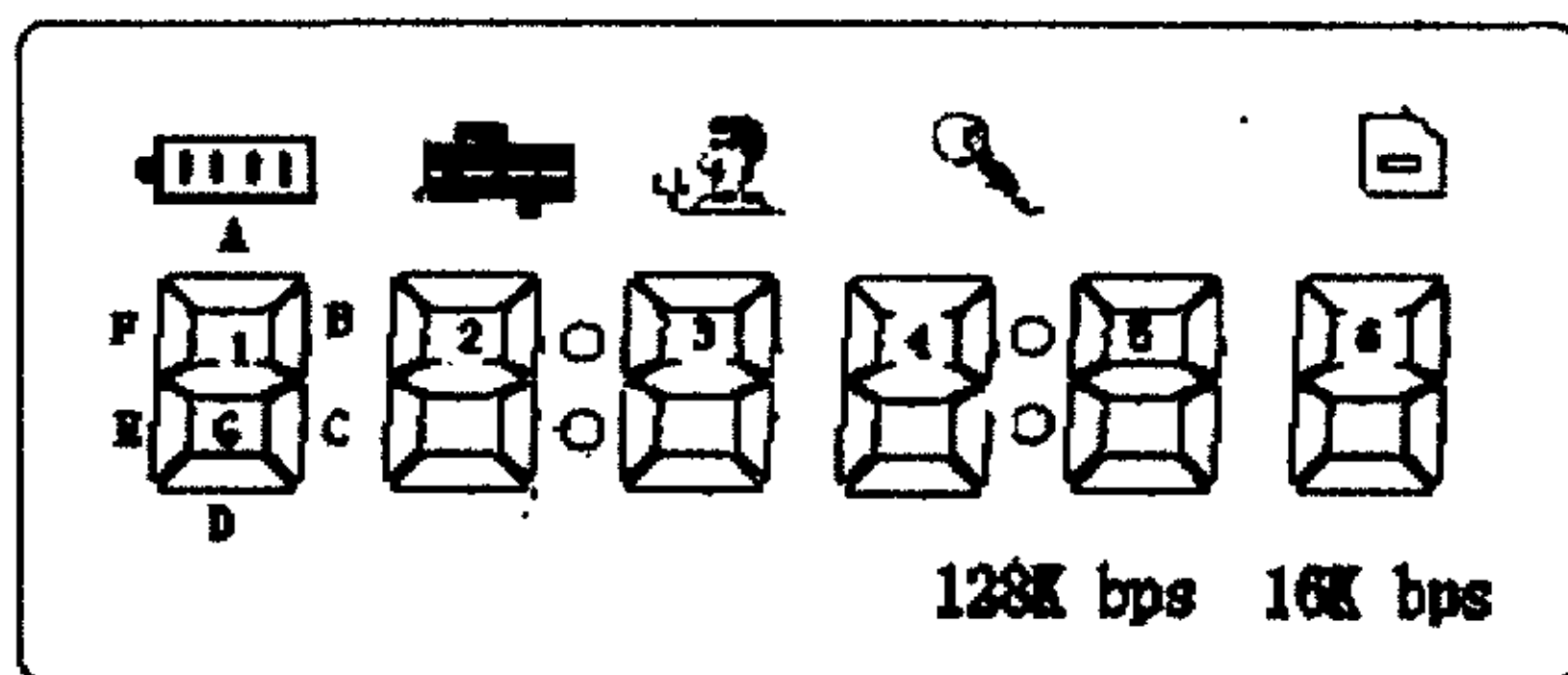


图 3.15 LCD 显示图案

### (3) 单片机响应按键输入

录音模式中共有 5 个按键, 开始/暂停、停止、录音模式选择、存储器类型选择、格式化存储器按键。开始/暂停和停止这两个按键采用中断类型, 一旦有按键输入, 就进入相应的中断子程序进行处理。其余两个按键采用查询类型, 在录音之前, 不断扫描端口从而得到相应的录音模式和存储器类别。

### (4) 单片机控制电源模块

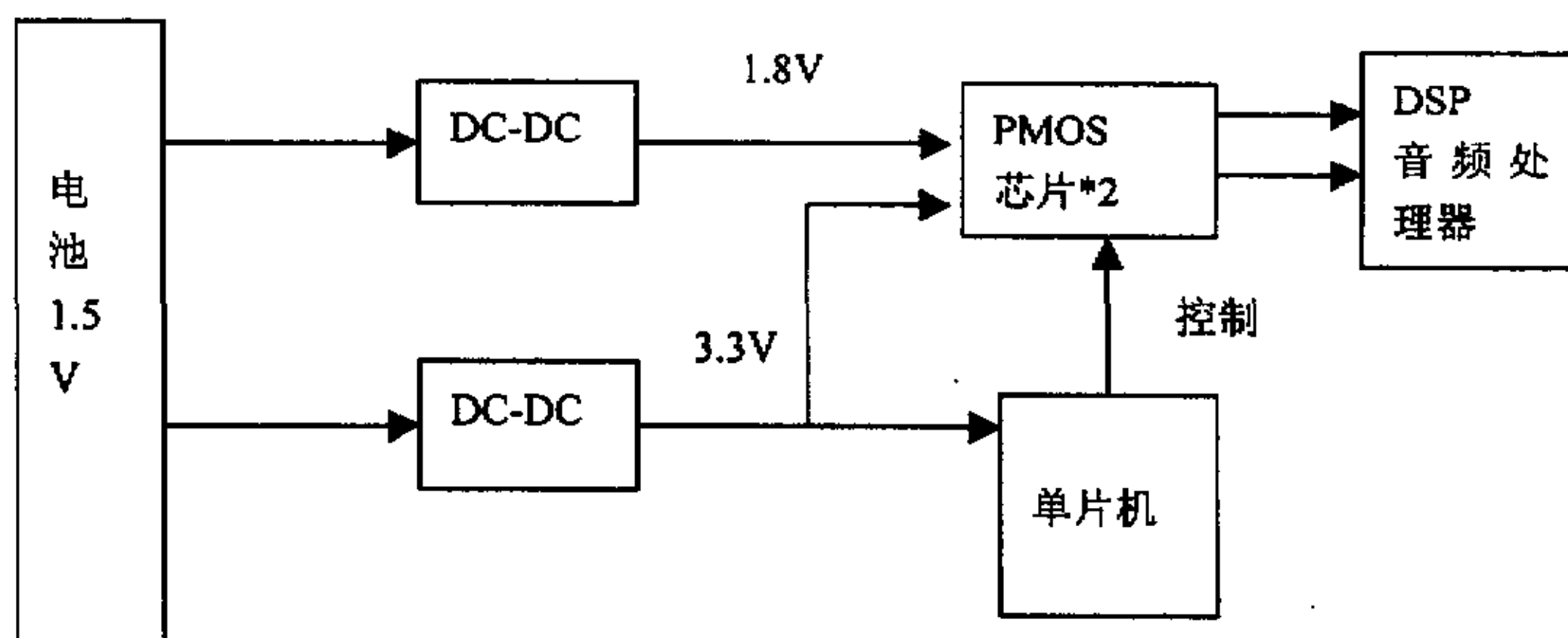


图 3.16 电源供电系统



适配器系统采用 5 号单电池供电, 因为系统中的芯片供电电压各不相同, 所以需要采用 DC-DC<sup>[23]</sup> 转换器输出所需的电压, 因为 DSP 需要 1.8V 和 3.3V 两路供电电压, 而其它芯片在 3.3V 电压下均可正常工作, 所以采用了两片升压型的 DC-DC 转换芯片, 一路输出 1.8V, 另外一路输出 3.3V。电源供电系统模块的设计电路框图如图 3.16, 对于消费类电子, 节省耗电量是必须重点考虑的, 所以, 对非录音状态, 单片机通过控制两个 PMOS<sup>[24]</sup> 管处于截止状态, 不对其它芯片供电,。只有在录音状态时, 单片机才使 PMOS 管导通。

另外, 单片机对 DSP 的控制, 以及单片机接收传送数据部分在本章第二节中已有详细介绍, 这里不再赘述。

### 3.3.3 单片机控制下的系统工作流程

本节将对适配器系统在录音模式下的工作流程作详细描述。在录音开始之前, 首先通过按键输入选择录音模式(音频录音或语音录音)和存储器类型(内置 Flash 存储器或外插 SMC)。然后按录音开始键, 单片机响应按键并做中断处理。首先与所要求的存储器取得通信, 读取 FAT 中的系统信息区域数据。如果存储器没有被格式化, LCD 将会显示要求操作者格式化存储器。否则单片机将读取 FAT 的文件系统信息区域, 获得当前存储器的存储状况, 将当前空闲扇区的序号标注在空闲扇区标志区域中, 同时在 LCD 上显示剩余录音时间。然后单片机写入当前录音文件的部分文件信息, 如当前录音文件的文件名、该文件所占第一个扇区序号及其起始地址、文件类型和比特率等。接着向存储器发出写数据指令, 此时存储器将等待接收数据。到此单片机结束了录音前的准备工作。

接下来, 单片机将会控制一片 PMOS 管导通, 给音频处理器供电, 同时利用它的 L3 接口控制音频处理器处于要求的工作状态。紧接着使两片 DSP 供电, 同时控制 DSP 引导装载 Flash ROM 中的代码程序, 至此录音的启动工作结束。单片机将等待 DSP 的中断信号从而接收被压缩的数据, 并且将数据传送到存储器当中。单片机同时要记录录音时间, 录音长度, 并且使 LCD 有相应的显示。

对于录音的结束有三种情况。第一, 接收到录音结束键的信息, 单片机首先关掉 DSP 和音频处理器的供电电压, 然后填充 FAT 的文件信息区域, 包括当前录音文件的文件长度, 播放时间等, 还要修改文件系统区域中相关的数据, 例如剩余扇区数, 所存文件个数等, 从而结束录音。第二, 如果当前存储器已满, 则自动结束录音, 并且修改 FAT 的相关数据, 与第一种情况相同, 同时 LCD 显示存储器已满。第三种情况属于意外断电的考虑。录音过程中, 如系统中途断电, 则会造成录音数据丢失, 已写入数据的存储器空间成为无法寻址空间。为避免上述情况发生, 在录音前, 单片机首先写入文件信息(其中, 文件长度、所占扇区

适配器系统采用 5 号单电池供电, 因为系统中的芯片供电电压各不相同, 所以需要采用 DC-DC<sup>[23]</sup>转换器输出所需的电压, 因为 DSP 需要 1.8V 和 3.3V 两路供电电压, 而其它芯片在 3.3V 电压下均可正常工作, 所以采用了两片升压型的 DC-DC 转换芯片, 一路输出 1.8V, 另外一路输出 3.3V。电源供电系统模块的设计电路框图如图 3.16, 对于消费类电子, 节省耗电量是必须重点考虑的, 所以, 对非录音状态, 单片机通过控制两个 PMOS<sup>[24]</sup>管处于截止状态, 不对其它芯片供电,。只有在录音状态时, 单片机才使 PMOS 管导通。

另外, 单片机对 DSP 的控制, 以及单片机接收传送数据部分在本章第二节中已有详细介绍, 这里不再赘述。

### 3.3.3 单片机控制下的系统工作流程

本节将对适配器系统在录音模式下的工作流程作详细描述。在录音开始之前, 首先通过按键输入选择录音模式(音频录音或语音录音)和存储器类型(内置 Flash 存储器或外插 SMC)。然后按录音开始键, 单片机响应按键并做中断处理。首先与所要求的存储器取得通信, 读取 FAT 中的系统信息区域数据。如果存储器没有被格式化, LCD 将会显示要求操作者格式化存储器。否则单片机将读取 FAT 的文件系统信息区域, 获得当前存储器的存储状况, 将当前空闲扇区的序号标注在空闲扇区标志区域中, 同时在 LCD 上显示剩余录音时间。然后单片机写入当前录音文件的部分文件信息, 如当前录音文件的文件名、该文件所占第一个扇区序号及其起始地址、文件类型和比特率等。接着向存储器发出写数据指令, 此时存储器将等待接收数据。到此单片机结束了录音前的准备工作。

接下来, 单片机将会控制一片 PMOS 管导通, 给音频处理器供电, 同时利用它的 L3 接口控制音频处理器处于要求的工作状态。紧接着使两片 DSP 供电, 同时控制 DSP 引导装载 Flash ROM 中的代码程序, 至此录音的启动工作结束。单片机将等待 DSP 的中断信号从而接收被压缩的数据, 并且将数据传送到存储器当中。单片机同时要记录录音时间, 录音长度, 并且使 LCD 有相应的显示。

对于录音的结束有三种情况。第一, 接收到录音结束键的信息, 单片机首先关掉 DSP 和音频处理器的供电电压, 然后填充 FAT 的文件信息区域, 包括当前录音文件的文件长度, 播放时间等, 还要修改文件系统区域中相关的数据, 例如剩余扇区数, 所存文件个数等, 从而结束录音。第二, 如果当前存储器已满, 则自动结束录音, 并且修改 FAT 的相关数据, 与第一种情况相同, 同时 LCD 显示存储器已满。第三种情况属于意外断电的考虑。录音过程中, 如系统中途断电, 则会造成录音数据丢失, 已写入数据的存储器空间成为无法寻址空间。为避免上述情况发生, 在录音前, 单片机首先写入文件信息(其中, 文件长度、所占扇区

数和播放时间暂时不写), 并修改文件系统区的总文件数 (总文件数加 1)。在录音完成后, 单片机再将文件信息中的文件长度、所占扇区数和播放时间写入, 并修改文件系统区的其它各项。这时, 如系统在录音过程中断电, 再次上电后, 单片机根据文件系统区的总文件数可检索到被中断的下载或录音数据的文件信息。在读该文件信息时发现文件长度为空, 则根据文件信息中的首扇区号检查该链表, 将链表的末扇区标志改为 0x0000。如该文件是录音文件, 则单片机根据链表将文件信息中的文件长度、所占扇区数和播放时间填入, 并修改文件系统区 (总文件数不变)。如该文件不是录音文件, 则单片机删除该链表, 删除该文件信息, 并将文件系统区的总文件数减 1。

### 3.4 系统改进

由于考虑到一片 C5402 芯片中硬件资源的不足, 研制成功的 MP3 实时编码器系统中采用了两片 DSP, 这也给算法的实现带来了一定的难度, 硬件调试的难度也相应增加。例如, 两片 DSP 的 BOOT 程序, 需要通过单片机对 DSP 的 #HOLD, #RS 的时序控制来完成。还有 DSP 之间数据通信的时序控制要求都是很高的, 所以, 可考虑采用单片的 TMSC5416 实现 MP3 实时编码算法。C5416 是 TI 公司采用  $0.15\mu\text{m}$  工艺生产的第一款 DSP 器件, 速度为 160MIPS, 有三个多通道缓存串行口, 有 128K 片内 RAM, 能够满足 MP3 实时编码器的要求。

另外, 本系统是作为 MP3 播放器<sup>[25]</sup>的适配器来开发的, 所以它只能作为现有 MP3 播放器的一个附件来使用, 在应用上受到很大的限制。直接将适配器设计为可实时录音同时可播放的 MP3 播放器, 是此类产品的最终目的。其原理框图如图 3.17, 由图可以看出, 我们可以直接用两片 DSP 芯片解码, 从而实现播放功能。

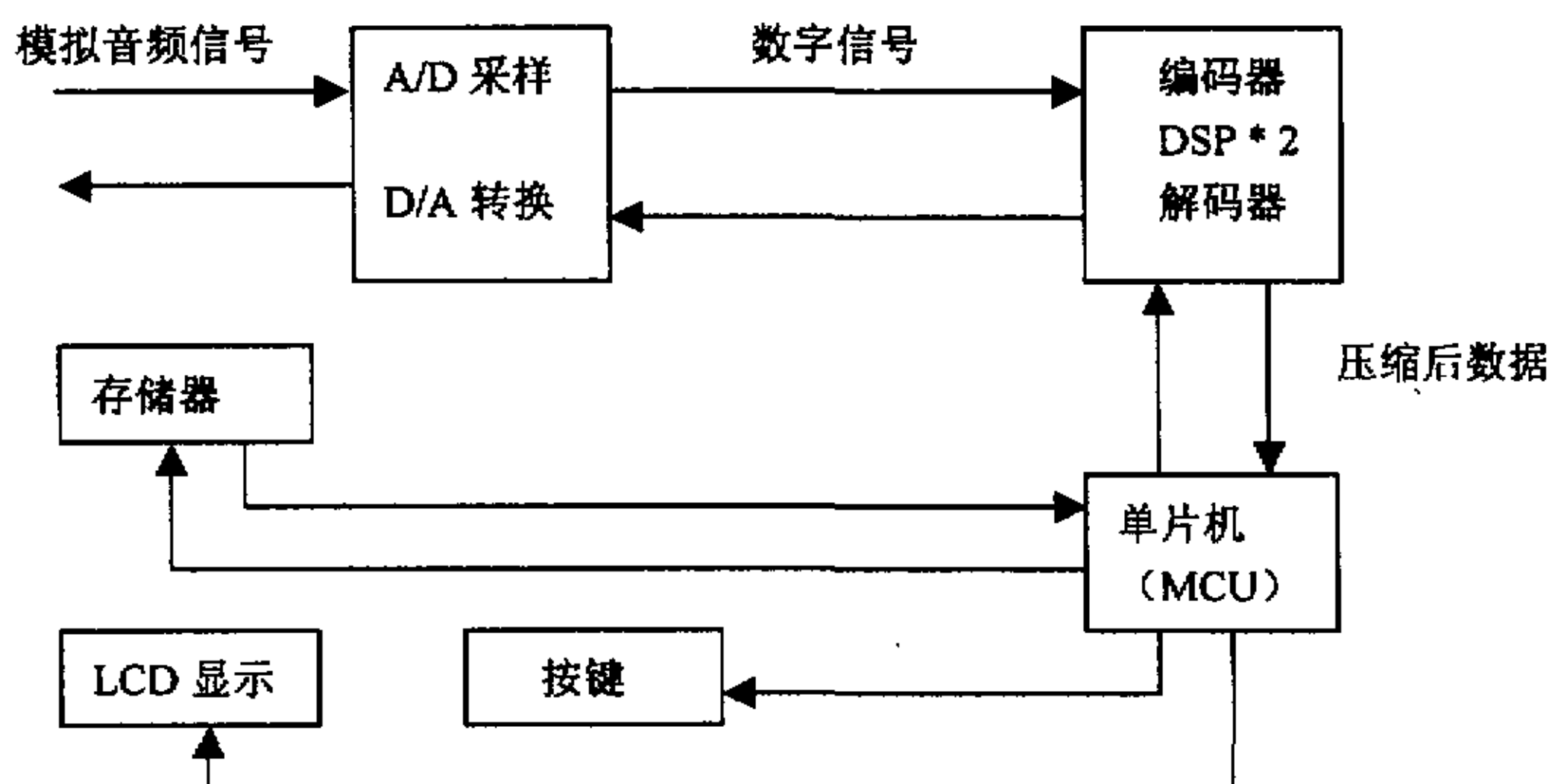


图 3.17 可实时录音 MP3 播放器原理框图

数和播放时间暂时不写), 并修改文件系统区的总文件数 (总文件数加 1)。在录音完成后, 单片机再将文件信息中的文件长度、所占扇区数和播放时间写入, 并修改文件系统区的其它各项。这时, 如系统在录音过程中断电, 再次上电后, 单片机根据文件系统区的总文件数可检索到被中断的下载或录音数据的文件信息。在读该文件信息时发现文件长度为空, 则根据文件信息中的首扇区号检查该链表, 将链表的末扇区标志改为 0x0000。如该文件是录音文件, 则单片机根据链表将文件信息中的文件长度、所占扇区数和播放时间填入, 并修改文件系统区 (总文件数不变)。如该文件不是录音文件, 则单片机删除该链表, 删除该文件信息, 并将文件系统区的总文件数减 1。

### 3.4 系统改进

由于考虑到一片 C5402 芯片中硬件资源的不足, 研制成功的 MP3 实时编码器系统中采用了两片 DSP, 这也给算法的实现带来了一定的难度, 硬件调试的难度也相应增加。例如, 两片 DSP 的 BOOT 程序, 需要通过单片机对 DSP 的 #HOLD, #RS 的时序控制来完成。还有 DSP 之间数据通信的时序控制要求都是很高的, 所以, 可考虑采用单片的 TMSC5416 实现 MP3 实时编码算法。C5416 是 TI 公司采用  $0.15\mu\text{m}$  工艺生产的第一款 DSP 器件, 速度为 160MIPS, 有三个多通道缓存串行口, 有 128K 片内 RAM, 能够满足 MP3 实时编码器的要求。

另外, 本系统是作为 MP3 播放器<sup>[25]</sup>的适配器来开发的, 所以它只能作为现有 MP3 播放器的一个附件来使用, 在应用上受到很大的限制。直接将适配器设计为可实时录音同时可播放的 MP3 播放器, 是此类产品的最终目的。其原理框图如图 3.17, 由图可以看出, 我们可以直接用两片 DSP 芯片解码, 从而实现播放功能。

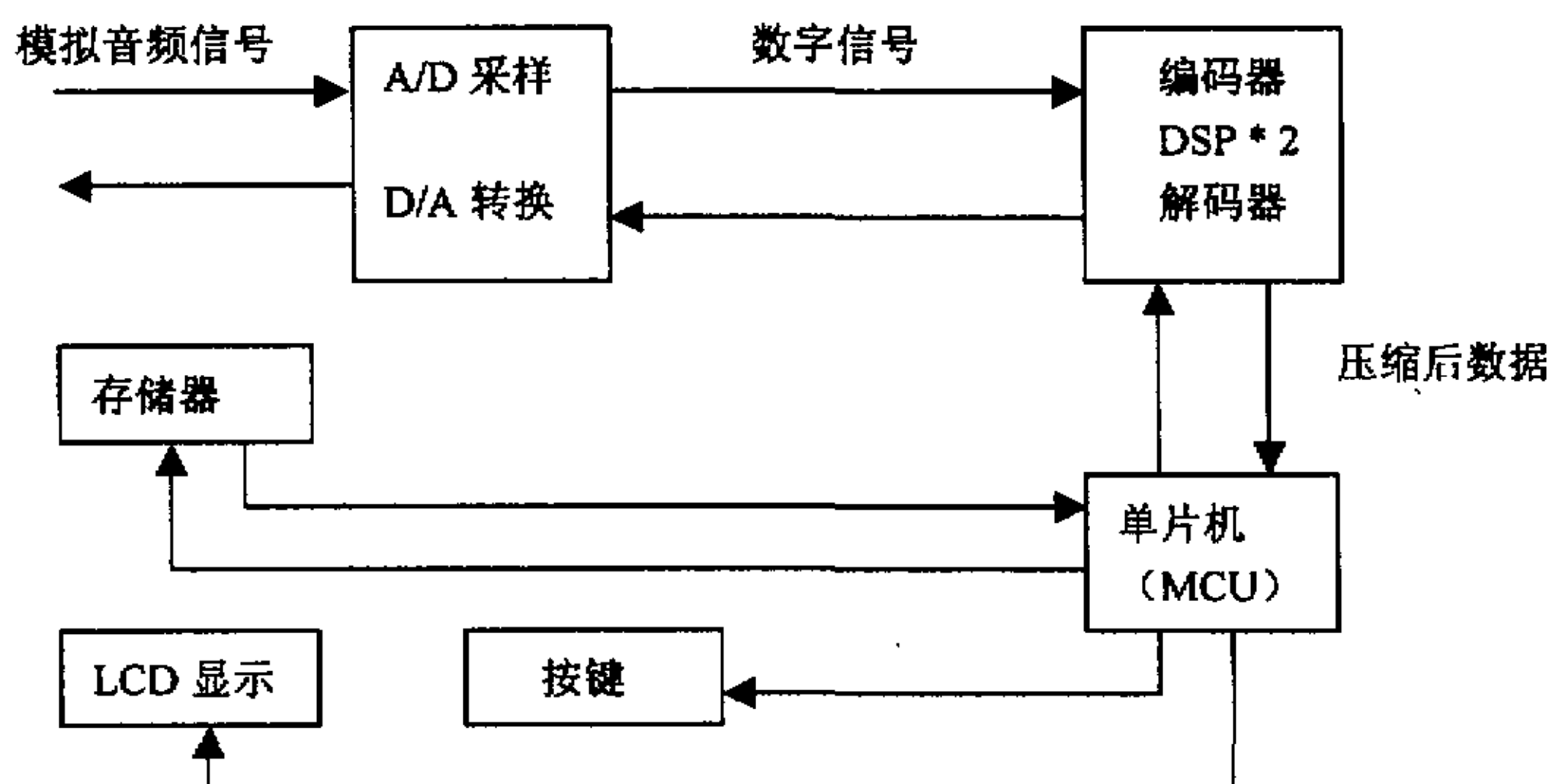


图 3.17 可实时录音 MP3 播放器原理框图



## 第四章 结束语

本文首先对 MPEG 音频第 III 层编码算法的标准进行了研究。对其编码压缩原理进行了详细的探讨。该标准采用了复杂的数学和心理学技术,同时利用了对人耳灵敏度的研究结果,实现了音频数据的高质量压缩。本文将其编码算法分成四个模块分别讨论:1,子带滤波器组编码;2,心理声学模型;3,量化与编码;4,帧数据流格式化。

在此基础上,通过对 TI 公司制造的 DSP 芯片 TMS320VC5402 性能的研究,利用其较高的性能价格比和较低的功耗等特点,采用了两片 DSP 芯片完成此算法的硬件实现。在实现过程中利用单片机来控制两片 DSP 依次引导装载程序,并充分利用了 DSP 提供的存冲串口使串行的 DSP 之间能够互相通信。在算法实现的基础上,通过附加一些外围系统模块最终成功地开发了具有实时录音功能的 MP3 硬件编码适配器。它主要有两方面的功能。第一,音频录音功能,用对录线将各种音频信号(CD、Walkman、MD 等)以 128Kbps 的码速率录制成为 MP3 格式。第二,语音录音功能,通过采用更低的 16KHz 的采样频率,可以录制现场语音,并以 16Kbps 的码速率压缩成具有 MP3 格式的数据形式。

目前该产品(见附录 B)已经进入了小批量生产阶段,并对最初生产的 50 套产品进行了综合测试。结果表明,用 128Kbps 的码速率录制 CD 或 Walkman 中的音乐节目,用 MP3 播放器回放,获得非常好的音乐效果。没有发现可识别的噪声,低音饱满,高音丰富,完全达到了 MP3 播放器的音乐质量要求<sup>[26]</sup>。用 16Kbps 码速率录制的现场语音非常的清晰,引入的噪声非常小,其语音回放质量超过了 32Kbps 的 ADPCM 编码语音回放质量,且存储能量比 ADPCM 格式编码提高一倍。同时,由于系统的硬件配置合理,软件编程严密,所以系统稳定性比较高,产品死机的概率非常小。另外,采用多种措施来降低系统功耗,使得一节 5 号电池可以连续录音达到 6 个小时以上,达到同类产品先进水平,增强了产品的市场竞争力。

总之,MP3 格式压缩编码技术除了可以获得较大的音乐数据压缩比外,还可以得到较高的音乐和语音回放质量。其硬件实时压缩系统的成功实现是对现有 MP3 播放器功能的有益补充,使其可以脱离 PC 机单独使用,大大地扩展了 MP3 播放器的应用领域。



## 附录 A

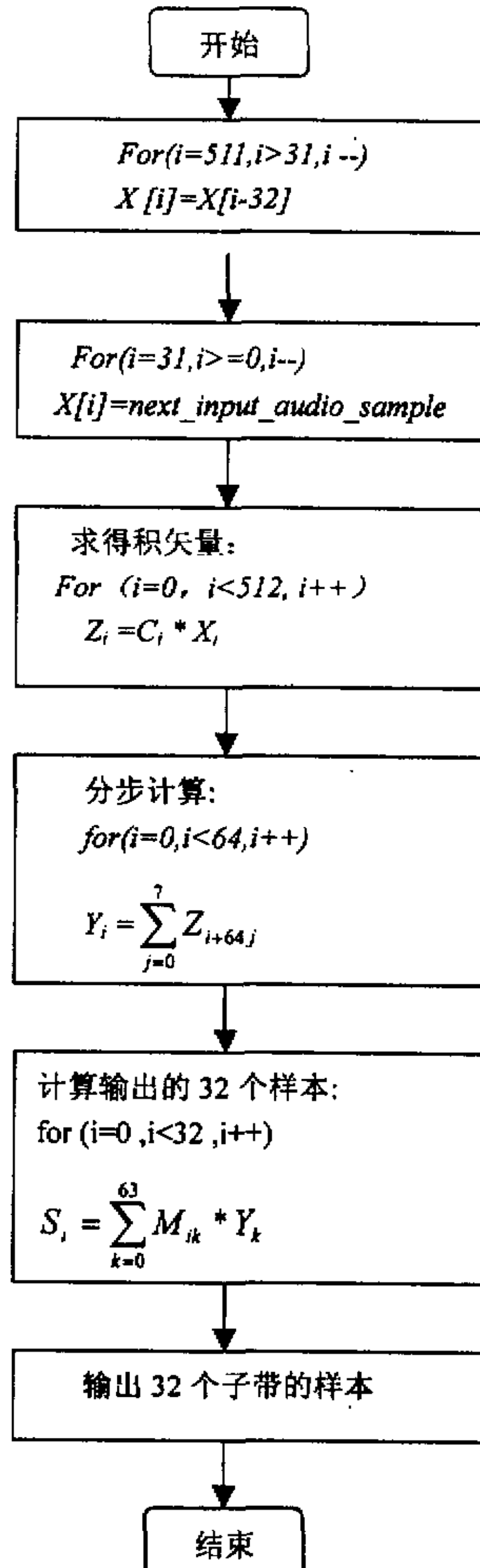


图 A.1 子带滤波器组算法流程图

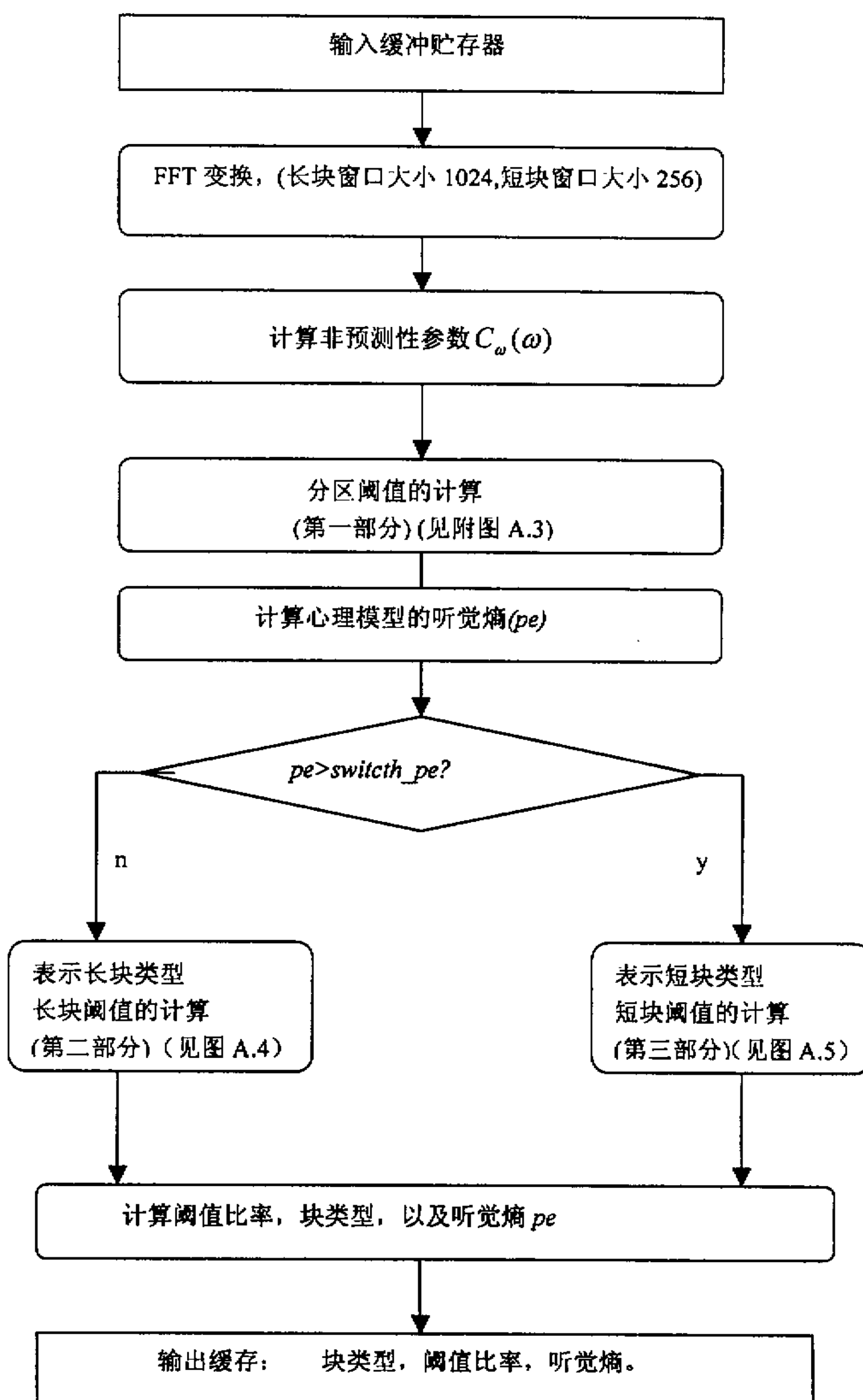


图 A.2 实际应用中的心理声学模型算法流程图

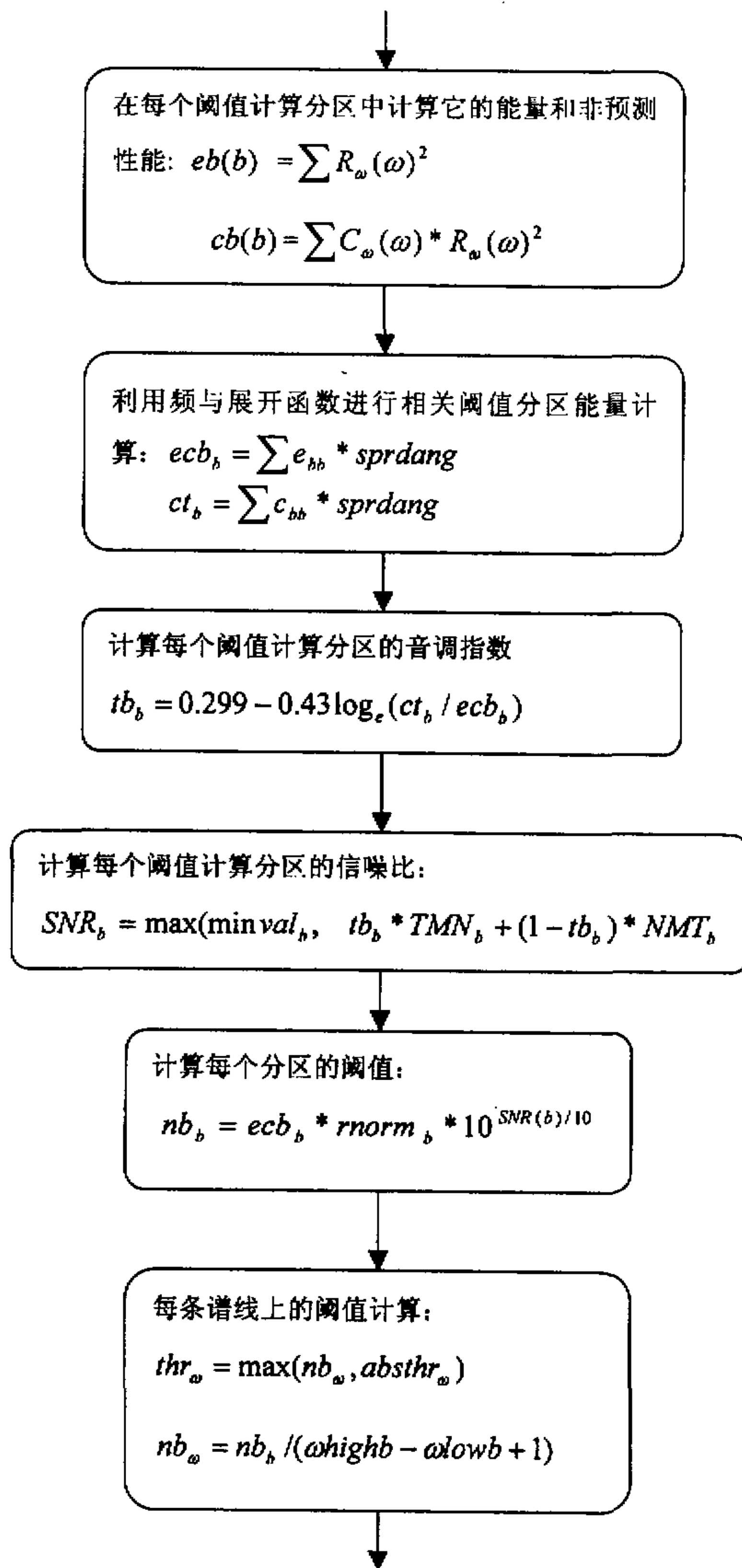


图 A.3 心理声学模型分区阈值的计算流程图

将阈值计算分区转换为比例因子波段,

每个比例因子波段的能量:  $en(sb) = \omega_1 * eb(bu) + \sum_{b=bu+1}^{b=bo-1} eb(b) + \omega_2 * eb(bo)$

每个比例因子波段的阈值:  $thrn(sb) = \omega_1 * thr(bu) + \sum_{b=bu+1}^{b=bo-1} thr(b) + \omega_2 * thr(bo)$

其中  $\omega_1, \omega_2$  的值参见参考文献[1]

计算每个比例因子波段的阈值比率:  $ratio(sb) = \frac{thrn(sb)}{en(sb)}$

图 A.4 心理声学模型长块阈值计算流程图

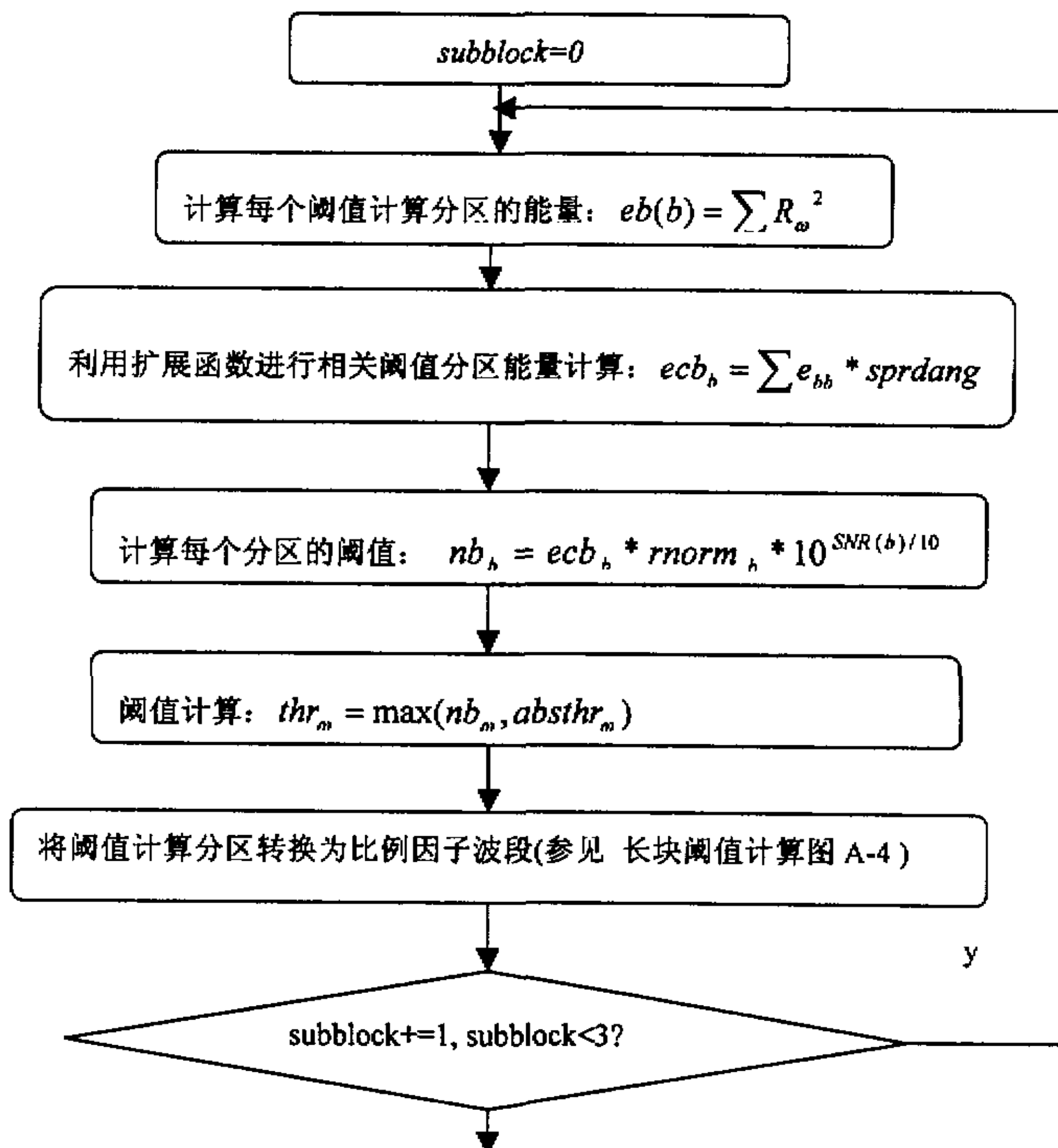


图 A.5 心理声学模型短块阈值计算流程图

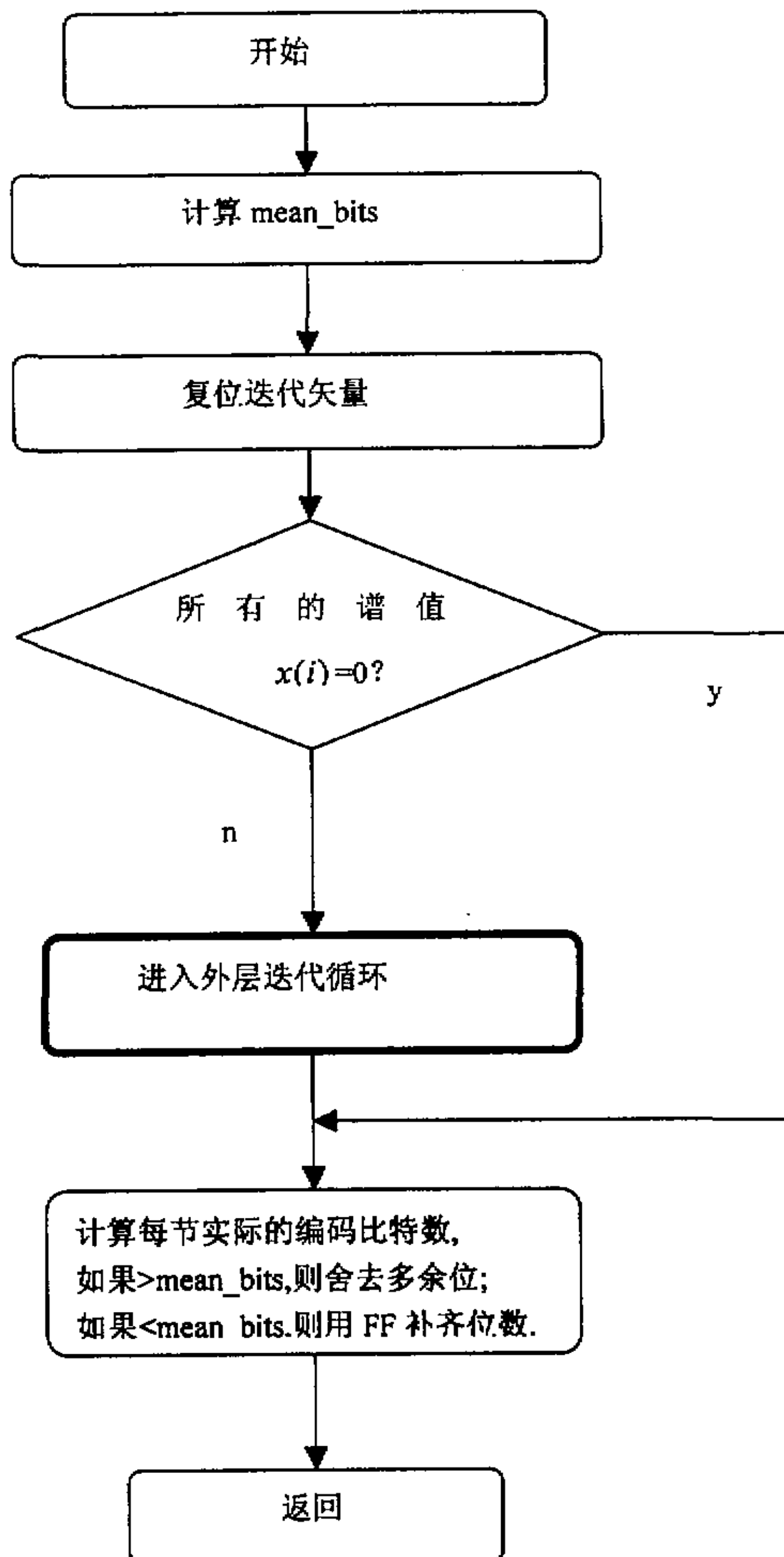


图 A.6 三层迭代循环模型算法流程图



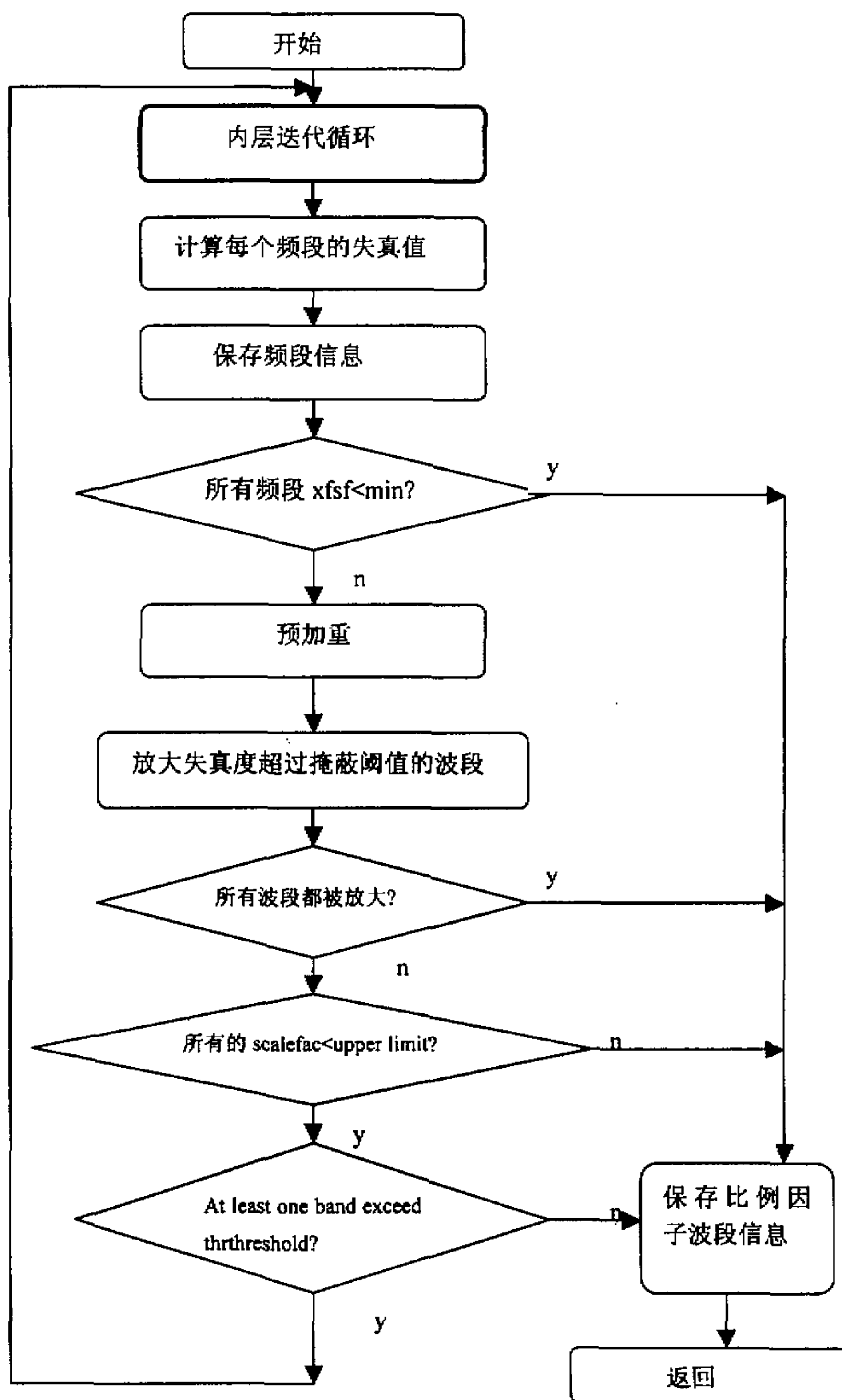


图 A.7 外层迭代循环算法流程图

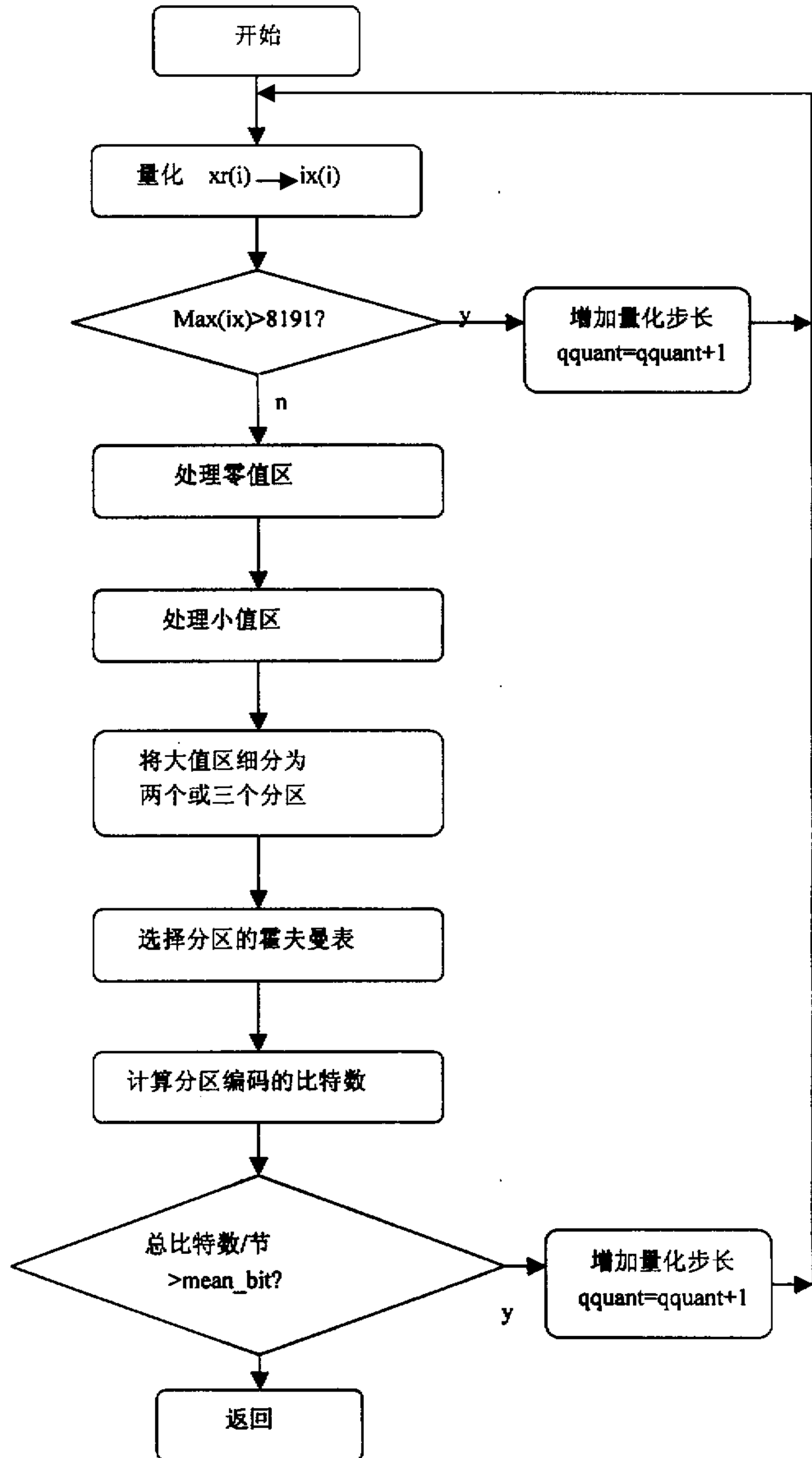
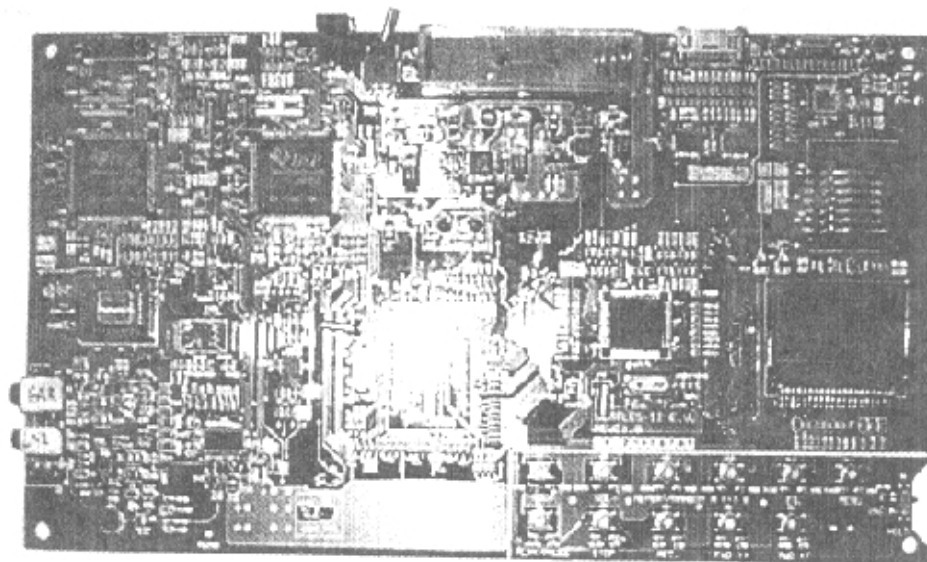
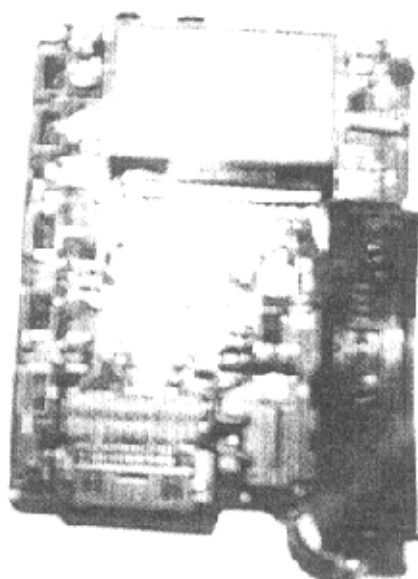


图 A.8 内层迭代循环流程图

## 附录 B



附录 B.1 MP3 编码适配器演示电路板



附录 B.1 MP3 编码适配器产品样机

## 致谢

首先我要诚挚的感谢我的导师姬红兵教授和武岳山副教授。他们科学严谨的治学态度和认真负责的精神时时鞭策着我发奋努力, 不断提高, 并且对我的论文工作提出了许多宝贵的意见与建议, 使我受益匪浅。在生活方面也给了我很大的鼓励、指导和帮助。

我还要特别感谢中国华大集成电路设计中心 ASIC 部总工程师唐斌博士, 他在论文完成过程中给予了热情的帮助和支持。

感谢中国华大集成电路设计中心 ASIC 部的江川、张牧童对我的关心与帮助, 他们总能热情耐心的解答我的问题。

感谢 202 教研室硕士生李一雷同学对我的关心与帮助。

感谢潘焱, 蒙卡娜, 周燕, 周永红, 王晓明, 鱼青峰, 吕楠, 刘曼等同事, 在我做论文课题期间给与的巨大支持与帮助。

最后, 感谢我的父母, 我将永远记住他们对我的祝愿与期望。

## 参考文献

- [1] ISO/IEC 11172-3. Coding of moving pictures and associated audio for digital storage media at up to about 1.5Mbit/s-Part 3: Audio [S]. 1993.
- [2] ISO/IEC 13813-3. Coding of moving pictures and associated audio for digital storage media at up to about 1.5Mbit/s-Part 3:Audio [S]. 1994.
- [3] 钟玉琢, 乔秉新, 祁卫. 运动图像及其伴音通用编码国际标准—MPEG-2 [M]. 北京: 清华大学出版社, 1997.
- [4] D.Pan. A tutorial on MPEG/Audio Compression[J]. IEEE Trans. On multimedia. 1995, Vol. 2, pp. 60-74.
- [5] Peter Noll. MPEG Audio Digital Coding. IEEE Signal Processing Magazine. 1997, 1053-5888.
- [6] P.Noll, D.Pan. ISO/MPEG Audio Coding[J]. International of the Audio Engineering Society(AES). 1994, 11, Vol.42, NO.10.
- [7] H.G.Musmann. The ISO Audio Coding Standard. Proc.IEEE Globecom. 1990, 12.
- [8] 高梅国, 何佩琨, 王飞. 数字信号处理器 DSPs 的发展[M]. 电子技术应用. 2000, 5.
- [9] 李刚. 数字信号微处理器的原理及其开发应[M]. 天津: 天津大学出版社, 2000.
- [10] TMS320VC5402 FIXED-POINT DIGITAL SIGNAL PROCESSOR. Texas Instruments Incorporated, 1999.
- [11] TMS320C54X User's Guide. Texas Instruments Incorporated, 1999.
- [12] TMS320C54X Peripherals Reference Guide. Texas Instruments Incorporated, 1999.
- [13] TMS320C54X Optimizing C Compiler User's Guide. Texas Instruments Incorporated, 1999.
- [14] Telecommucations Applications With the TMS320C54X DSPs Application Book. Texas Instruments Incorporated, 1999.



- 
- [15] TMS320C54X C Source Debugger User's Guide . Texas Instruments Incorporated, 1999.
- [16] 张雄伟, 曹铁勇. DSP 芯片的原理与开发应用[M]. 北京: 电子工业出版社, 2000.
- [17] 韦江维, 廖义奎, 农建波等. TMS320VC5402 DSP 的并行 I/O 引导装载方法研究. 电子技术应用. 2000, 8.
- [18] 程佩青. 数字信号处理[M]. 北京: 清华大学出版社, 1995.
- [19] UDA1341TS Economy audio CODEC for MiniDisc(MD) home stereo and portable applications. PHILIPS, 1998, 12.
- [20] H.S.Malvar. Signal Processing with Lapped Transforms. Artech House Inc., 1992
- [21] Nand Flash Memory & SmartMedia Databook. SAMSUNG, 1999.
- [22] 8-BIT MICROCONTROLLER MB89560 SERIES HARDWARE MANUAL. Fujitsu Semiconductor, 1995.
- [23] 1-to-Cell, High-Current, Low\_Noise, Step-Up DC-DC Converters with Linear Regulator. MAXIM, 1997, 4.
- [24] Si2305Ds. VISHAY, 1998, 12.
- [25] 王宏. 基于 TMS320C50 的通用语音信号处理系统. 电子技术应用. 2001, 3.
- [26] 唐斌. 时尚娇点: MP3 数字音乐播放器. 电子产品世界. 2001, 3.

## 在学期间研究成果

### 一. 发表论文:

1、武岳山, 马延飞. 由 PowerBuilder 生成动态链接库及其应用. 西安电子科技大学学报. 2000, 6. 第 27 卷 增刊. pp. 128~130. EI page one 数据库收录.

2、马延飞. 基于 DSP 的 MPEG 音频第 III 层编码算法的硬件实现. 单片机与嵌入式系统应用. 已被录用.

### 二. 科研成果:

科研项目课题: MP3 硬件播放器系统解决方案。

作者主要负责编码功能的硬件实现。

该项目通过了信息产业部部级优秀成果鉴定。