

消除语音系统噪声的软件方法

郑 宇, 王 波

(上海理工大学管理学院, 上海 200093)

摘 要: 在 MP3、MP4、PDA、手机等智能音频设备中, 音质是衡量其性能最重要的标准之一, 但在播放或录音时的噪音会严重影响其音质效果。该文分析以硬件方法消除“POP”噪声的弊端, 研究使用软件来降低、避免和消除噪声的方法, 介绍软件的设计方案, 结合 Windows CE 平台给出程序的实现。该设计方案经过大量项目的实践和改进, 具有设计灵活、节省成本和应用效果良好等优点。

关键词: 噪声; 线程; 偏置电压

Software Method of Noise Cancellation in Voice Systems

ZHENG Yu, WANG Bo

(Business School, University of Shanghai for Science and Technology, Shanghai 200093)

【Abstract】 The quality of voice is an important design request of embedded voice device, especially in MP3, MP4, PDA and mobile telephone. If there is noise it will influence their timbre. This paper analyzes the defects when using the hardware methods to eliminate noise, researches how to use software methods to reduce, prevent and eliminate noise, and gives the design of the software program. The essential functions that should be involved in noise cancellation driver are analyzed. The design is proved to be correct and effective by the successful result.

【Key words】 noise; thread; bias voltage

1 音频系统中噪声产生的原因

“POP”噪声是指音频器件在上电、断电瞬间及上电稳定后各种操作带来的瞬态冲击所产生的爆破声^[1]。本文提到的音频系统是指音频半导体器件, 包括音频数模转换器、模数转换器、音频放大器等应用系统。产生“POP”噪声的瞬态冲击通常是一种很窄的尖脉冲^[1-2], 如图 1 的第 3 个波形。在音频芯片或音频功率放大器上电、断电瞬间及在待机切换时, 电源对输入电容、偏置电容或输出电容等充电时, 难免会产生尖脉冲信号, 即“POP”爆破声^[3]。大量的实验表明, 大部分“POP”噪声是由芯片上电、掉电时偏置电压的瞬间跳变引起的。

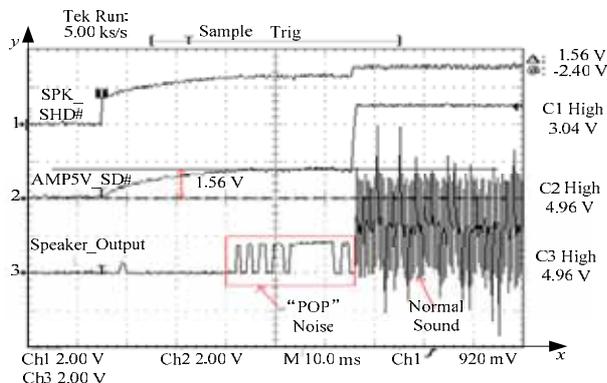


图 1 “POP”噪声的测量波形

硬件电路设计中消除“POP”噪声的常用方法如下:

(1)桥式结构抑制共模噪声, 桥式模式的噪声明显小于单端模式的噪声。

(2)无输出耦合电容(OCL)结构, 该结构与桥式结构非常类似, 在输出端将直流共模电压抵消掉, 只有交流信号对负

载做功^[4]。

(3)使用外部的静音(MUTE)电路。

实验表明, 上述硬件设计方案可以在一定程度上降低“POP”噪声, 但是使用硬件解决方案存在如下几个缺点: (1)增加硬件设计的复杂性和开发周期。(2)增加硬件设计成本和最终产品的成本。(3)导致语音信号的延时或失真, 上述设计方案往往采用滤波电容过滤噪声, 如果滤波电容过大会使芯片的建立时间变长^[5-6], 使人感觉声音“久久”没有输出。电容过大还会使音频系统的重要指标——总谐波失真+噪声(THD+N)变差。(4)降低电池的续航能力。

2 消除噪声的软件方法

不管采用上述哪种硬件方法消除“POP”噪声都会增加产品的成本, 而且使用单一的解决方法并不能完全消除“POP”噪声, 只能在一定程度上减少和降低“POP”噪声, 笔者根据大量的项目实践, 总结并改进了几种切实有效的软件方法来降低和消除嵌入式音频系统中的噪声, 不但可以节省产品成本, 而且增加了灵活性。因篇幅所限, 本文重点研究 2 种软件消除噪声的方法, 并给出架构设计和驱动程序实现方案。

在给出软件方案之前, 需要重点关注音频器件上电时序的问题, 根据大量的项目实践, 总结出应该遵守如下的时序:

(1)初始化音频编解码器(audio codec), 给偏置电压(bias voltage)上电时需要增加一定的延时。

(2)初始化外部功率放大器(amplifier)或者麦克风芯片, 需

基金项目: 上海市重点学科建设基金资助项目(T0502)

作者简介: 郑宇(1979-), 男, 工程师、硕士, 主研方向: 信息管理与信息系统; 王波, 教授

收稿日期: 2008-07-26 **E-mail:** aqbc@163.com

要尽可能等待偏置电压稳定后再给功放或者麦克风上电。

(3)初始化扬声器或者耳机芯片。

遵循上述的音频芯片上电时序可以尽可能降低“POP”噪声，上述3个时序之间需要增加一定的延时，以便前面的电压稳定，不影响后面的初始化进程，具体的延时时间需要根据不同厂商的芯片特性决定。

下面重点介绍2种使用软件消除“POP”噪声的方法。在嵌入式音频设备中，常见的噪声存在于2个方面：扬声器和耳机的输出端和麦克风的输入端，下述2种设计方案都适用于这2种情况，为了节省篇幅，分别针对输出和输入端的噪声消除进行了分析，并给出实现方案。

(1)使用独立的线程初始化音频芯片

本节设计了输出端的噪声消除方案，该方案也适用于录音时的状况。使用单独的线程控制音频芯片有如下优点：

- 1)可以方便地控制该线程的运行时序。
- 2)音频芯片和DMA控制器^[7-8]的初始化是同步运行的，降低了音频芯片和DMA控制器等器件的初始化时延，避免声音输出的延时。
- 3)便于软件开发人员维护程序代码。

图2示出了使用独立线程初始化音频芯片的架构。

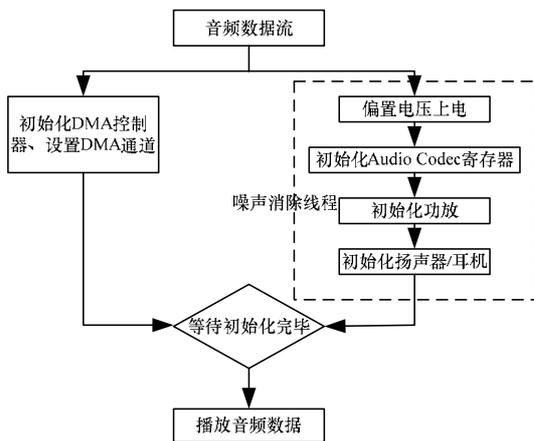


图2 使用独立线程初始化音频芯片的架构设计

参考图2,给出在Windows CE平台下输出端噪声消除驱动程序的实现方案。

1)在Windows CE的WAV_Init^[9]中创建噪声消除线程和消息事件

```

g_hNoiseCancelThread=CreateThread(NULL, 0, (LPTHREAD_
START_ROUTINE) NoiseCancelThread, NULL, 0, NULL);
if(!g_hNoiseCancelThread)
//如果噪声消除线程创建失败，则返回错误信息
{RETAILMSG(1, (TEXT("WAVEDEV.DLL- Failed to create
NoiseCancelThread.\r\n")));
goto Exit; }
g_hNoiseCancelEvent=CreateEvent( NULL, FALSE, FALSE,
NULL); //创建噪声消除通知事件
if(!g_hNoiseCancelEvent)
{ RETAILMSG(1, (TEXT("WAVEDEV.DLL- Failed to create
Noise Cancel Event.\r\n")));
goto Exit;}
  
```

2)实现噪声消除线程

```

LPTHREAD_START_ROUTINE NoiseCancelThread ( volatile
unsigned char * myParam )
{ while(1)
  
```

```

{ WaitForSingleObject(g_hNoiseCancelEvent,INFINITE);
//等待噪声消除事件被触发
InitAudioGPIO();//初始化音频设备 GPIO,给偏置电压上电
Sleep();
//延时,等待电压稳定,具体的值需要参考芯片的规格说明书
InitAudioCodec();//初始化音频编解码芯片
Sleep();//延时,等待音频编解码芯片上电稳定,具体的值
//需要参考芯片的规格说明书
InitAudioAMP();//初始化功率放大器,给其上电
Sleep();//延时,等待功放芯片上电稳定,具体的值需要参
//考芯片的规格说明书
InitAudioDevice();//初始化其他音频相关的芯片
}}
  
```

3)当检测到音频数据时激活噪声消除线程

```

BOOL HardwareContext::StartOutputDMA()//当有音频数据需要
//播放时,上层应用程序会调用此函数
{RETAILMSG(1,(TEXT("StartOutputDMA\r\n")));
waveopen_count++;
//计数,判断是否是第1次有音频数据调用该函数
if(waveopen_count==1)
//如果是第1次,则初始化相关音频芯片
{SetEvent(g_hNoiseCancelEvent);
//触发噪声消除事件,以便激活噪声消除线程
InitOutputDMA();
//初始化DMA控制器,并设置输出DMA通道
...//完成其他的事情 }
...//完成其他的事情 }
  
```

上述设计方案能比较有效地消除和避免嵌入式音频系统中的“POP”噪声，但是有一点需要注意，开发人员应根据芯片规格说明书上的上电时序要求初始化音频芯片，而具体延时时间需开发人员根据具体的项目和经验进行调整。

(2)使用清除DMA缓存的方法消除噪声音频数据

噪声也是一种信号，会经过DMA通道之后由扬声器或者耳机输出，由于噪声的音频数据通常都出现在播放的最初时候，因此本文的设计方案是在噪声音频数据进入DMA通道之后强行清空前端DMA通道的数据。这样设计的优点是能从音频数据的源头上清除噪声。

使用清除DMA缓存方法消除噪声的架构设计如图3所示。

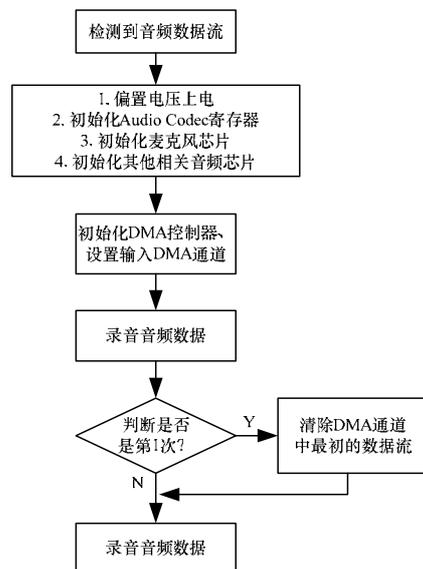


图3 通过清除DMA缓存消除噪声的架构设计

参考图 3,给出了在 Windows CE 平台下输入端噪声消除的驱动程序实现方案。

1)按照图 3 初始化相关的音频芯片

```

BOOL HardwareContext::StartInputDMA()
//当有音频数据需要录音时,上层应用程序会调用此函数
{
    RETAILMSG(1,(TEXT("StartInputDMA\r\n")));
    waveopen_count++;
//设置计数器,用来判断是否是第 1 次调用
    if(waveopen_count==1)
    {
        InitAudioGPIO();
//初始化音频设备 GPIO,给偏置电压上电
        Sleep();
//延时,等待电压稳定,具体的值需要参考芯片的规格说明书
        InitAudioCodec();
//初始化音频编解码芯片
        Sleep();
//延时,等待音频编解码芯片上电稳定,具体的值需要参考芯片
//的规格说明书
        InitMicrophone();
//初始化功率放大器,给其上电
        Sleep();
//延时,等待功放芯片上电稳定,具体的值需要参考芯片的规格
//说明书
        InitAudioDevice();
//初始化其他音频相关的芯片
        InitInputDMA();
//初始化 DMA 控制器,并设置输入 DMA 通道
        ...//完成其他的事情
    }
    ...//完成其他的事情
}

```

2)清除 DMA 前端一定字节的噪声音频数据

```

static USHORT InputCount;
//设置清除计数器
BOOL ClearDMABuffer()
//清除 DMA 缓存的函数
{if(InputCount<12)
//设置需要清除的数量,需要根据具体的项目设定
{
    memset(pBufferStart,0x0, AUDIO_DMA_PAGE_SIZE);
//清除 DMA 缓存,memset()参考 MSDN[9]
    InputCount++;
//计数,当下次录音的时候需要重新清 0
}}

```

上述方法能够有效地消除和减少“POP”噪声,但是该方案对开发人员提出了比较高的要求,需要把握好需要清空 DMA 字节的大小,如果设置得不好,就会清除正常的语音数据,因此,需要根据具体的项目和经验进行调整。另外,如果没有噪声,则不需要使用该方案。

(3)使用软件方法消除噪声方案的实现效果

经过大量的项目实践,发现使用上述 2 种软件消除噪声

的方案基本能达到消除和降低“POP”噪声的要求,播放和录音的音质效果得到了改善,如图 4 所示,以输出端音频数据的测量结果为例,可以看到最终输出的音频数据质量得到了较大的提升。

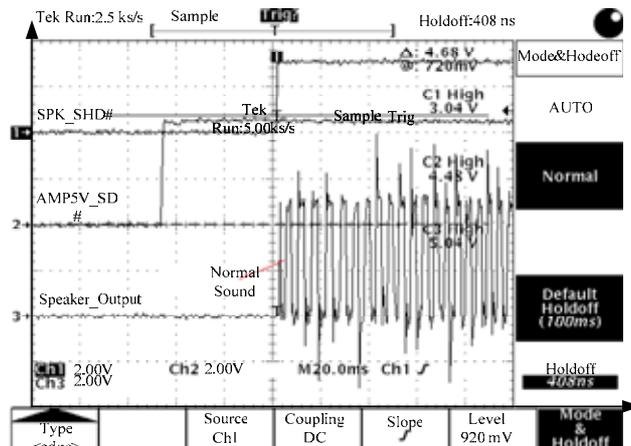


图 4 使用软件消除“POP”噪声的效果图

3 结束语

本文分析了使用硬件消除“POP”噪声的弊端和不足,对使用软件降低、避免和消除噪声的方法进行了研究,给出了软件的设计方案,并结合 Windows CE 平台给出了驱动程序的实现。

本文提出了 2 种利用软件方法消除“POP”噪声的新方案:(1)使用线程来控制芯片的初始化时序,方便控制而又灵活;(2)使用清除 DMA 缓存的方法从根源上清除噪声数据。方案的不足之处是需要开发人员具备较多的开发经验。该解决方案已经在基于 Windows CE 操作系统平台上得以实现,在实际项目开发中得到广泛采用,并获得了良好的应用效果。

参考文献

- [1] 王 玉. 音频系统应用中的“POP”噪声及其常用解决方法[Z]. (2006-06-21). <http://www.eetchina.com/>.
- [2] 刘 鹏. 一种脉冲噪声图像复原算法[J]. 计算机研究与发展, 2006, 43(11): 1939-1946.
- [3] 郑 双, 朱名日. 嵌入式系统电源的设计与调试[J]. 单片机与嵌入式系统应用, 2007, 11(10): 67-69.
- [4] 姬建伟, 宋家友, 王晓东. 一种低功耗高线性度低噪声放大器设计[J]. 微计算机信息, 2007, 13(5): 280-282.
- [5] 刘 虹, 黄 涛, 徐 成. 基于 FPGA 的音频处理芯片的设计[J]. 微计算机信息, 2005, 11(1): 91-92.
- [6] 曹仰杰. 一种自适应抗噪声的 PPM 预测模型[J]. 计算机工程, 2007, 33(4): 64-66.
- [7] Microsoft Corporation. Introducing Microsoft Windows CE 3.0[Z]. Microsoft Corporation, 2001-01.
- [8] 於少峰. 基于 AC97 标准的嵌入式音频系统设计与实现[J]. 电子器件, 2004, 27(4): 193-196.
- [9] Microsoft. MSDN Library[Z]. (2006-02-01). <http://msdn.microsoft.com/library>.