

文章编号:1001-9081(2008)02-0491-03

基于 DSP 的实时语音检测的设计与实现

杨永超,付中华,蒋冬梅

(西北工业大学 计算机学院,西安 710072)

(yyc821103@sohu.com)

摘 要:提出了一种基于 DSP 的实时语音检测的方法,通过设置中断服务程序实现 DSP 与 DMA 的并行处理,采用双缓冲加一缓冲的方法保持语音连续性,利用短时能量状态转换图的方法对语音进行检测及存储,最后利用自制控制板实现程序控制和状态显示。实验结果表明,检测的语音数据和标注的语音数据相比较平均正确率可达 94.98%,有效地实现了语音的实时处理。

关键词:数字信号处理器;中断服务程序;语音检测;短时能量

中图分类号: TP912.11 **文献标志码:** A

Design and implementation of real-time voice activity detection based on DSP

YANG Yong-chao, FU Zhong-hua, JIANG Dong-mei

(School of Computer Science, Northwestern Polytechnical University, Xi'an Shanxi 710072, China)

Abstract: In this paper, a method of Real-time Voice Activity Detection (VAD) based on DSP was proposed. After configuring Interrupt Service Routine(ISR), the audio data was saved through parallel processing of DSP and DMA based on the method of Short-term energy state-switch process. A control board was designed to implement program control function. Experimental result shows that the data of storage account for 94.98% of the marked in the DSP system, so as to realize the real time processing of voice.

Key words: DSP; Interrupt Service Routine(ISR); Voice Activity Detection (VAD); short-term energy

0 引言

语音检测是语音分析、语音合成和语音识别中的一个重要环节。在背景噪声或一定的干扰下,正确识别语音信号的起止点对语音信号的后续处理十分重要。语音检测可以提高实时系统运行的效率,节约 DSP 的计算量和存储空间,有利于在存储器容量相对较小的 DSP 平台上实现。

许多语音短时分析都有帧间重叠,为了在这种条件下保持连续性,本文采用双缓冲加一缓冲的方法来保持语音连续及实时。DSP 计算检测特征和进行检测分析时需要占用指令周期,为了避免中断信号采集过程,所以采用了 DMA 中断方式和双缓冲技术。本文通过配置 DSP 的相关寄存器,编写 DMA 的中断服务程序,实现了 DSP 与 DMA 的并行处理,同时利用基于短时能量的状态转换图方法进行语音检测,将结果保存到片外存储器中,当片外存储器存储的数据到达预置的长度后,停止处理,将语音数据回放出来。整个过程用自制的控制板来实现程序的开关功能和状态显示功能。

本文所做的工作都是在北京合众达公司的 SEED-DEC5502 平台上完成的,此平台采用美国 TI 公司的 TMS320VC5502,相关信息可查询^[1]。

1 DSP 与 DMA 的并行处理

1.1 DMA 中断处理方式

在这种方式下,DSP 不再被动等待,而是可以执行其他程序,一旦 DMA 传输数据完毕,可以向 DSP 提出服务请求,DSP

如果响应该请求,便暂时停止当前程序的执行,转去执行与该请求对应的中断服务程序,完成后,再继续执行原来被中断的程序。

DMA 中断处理方式的优点是显而易见的,它不但为 DSP 省去了查询外设状态和等待外设就绪所花费的时间,提高了 DSP 的工作效率,还满足了外设的实时要求。但需要为每个 I/O 设备分配一个中断请求号和相应的中断服务程序。

DMA 最明显的一个特点是它不是用软件而是采用一个专门的硬件电路——DMA 控制器来控制内存与外设之间的数据交流,无须 DSP 介入,较大提高 DSP 的工作效率。

在进行 DMA 数据传送之前,DMA 控制器会向 DSP 申请总线控制权,DSP 如果允许,则将控制权交出。因此,在数据交换时,总线控制权由 DMA 控制器掌握,在传输结束后,DMA 控制器将总线控制权交还给 DSP。

1.2 DSP 与 DMA 的并行处理原理

DMA 通道启动之前首先要对几个关键的 DMA 寄存器进行配置(如表 1 所示),这几个关键寄存器包括:DMA 源和目的参数寄存器 DMACSDP(Source and destination parameters register)、DMA 通道控制寄存器 DMACCR(Channel control register)、DMA 中断控制寄存器 DMACICR(Interrupt control register)、DMA 源开始地址和目的开始地址寄存器 DMACSSAL、DMACSSAU、DMACDSAL、DMACDSAU(Source/Destination start address (lower/upper part) register)、帧长度寄存器(Frame number register)和元素长度寄存器(Element number register)等。其他寄存器以及引脚配置请查询文献

收稿日期:2007-08-24;修回日期:2007-10-15。 基金项目:陕西省科学技术研究发展计划项目(2005k04-G23)。

作者简介:杨永超(1982-),男,陕西西安人,硕士研究生,主要研究方向:DSP 及数字语音处理;付中华(1977-),男,湖北十堰人,副教授,主要研究方向:语音信号处理、数字音频信号处理、多媒体信号压缩与编解码;蒋冬梅(1973-),女,河南永城人,副教授,博士,主要研究方向:语音识别、语音压缩、模式识别。

[2]和[3]。

表 1 DMA 控制寄存器配置

DMA 控制寄存器	配置信息
DMACSDP	12-9: DST-DARAM via internal memory port 1
	5-2: SRC-Peripherals
	1-0: DATATYPE-16-bit
DMACCR	15-14: DSTAMODE-Automatic post increment
	13-12: SRCAMODE-Constant address
	8: AUTOINIT-enabled
	7: EN-disabled
DMACICR	4-0: SYNC-McBSP 1 receive event
	3: FRAMEIE-Send interrupt request
DMACSSAL	(DMA_AdrPtr) (MCBSP_ADDR(DRR11))
DMACSSAU	0
DMACDSAL	(DMA_AdrPtr) & DaramBuf1
DMACDSAU	0
DMACFN	1024
DMACEN	1

注:表 1 中的配置信息栏中,前面的数字表示引脚号,“:”后边的信息表示所选择的配置选项,在文献[3]中以数字形式给出。

配置完以上几个关键的 DMA 寄存器后,启动 DMA 通道,让 DMA 通道开始传输数据到存储器 DaramBuf1 中(如图 1 所示),此时,DSP 处于等待状态,当存储器 DaramBuf1 满,触发中断,这时,DSP 开始处理存储器 DaramBuf1 中的数据,而 DMA 继续传输数据到另一个存储器 DaramBuf2,也就是说,DMA 传输数据到存储器 DaramBuf2 的同时,DSP 正在处理先前传输到存储器 DaramBuf1 中的数据。等存储器 DaramBuf2 传满,则中断服务程序跳转到存储器 DaramBuf2 进行数据处理,DMA 传输数据到刚才处理过的存储器 DaramBuf1,这样,实现一个循环并行的数据传输和处理过程。

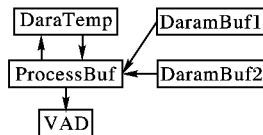


图 1 实时语音检测原理

1.3 DMA 中断程序设计

中断程序用来响应 DMA 通道接收事件,进入中断时,首先对下一次 DMA 工作寄存器进行配置,以便当中断服务程序运行时,DMA 也在传输数据。配置完成后,中断服务程序开始处理已经传满的存储器,进行相应的程序处理,处理完之后,等待下一个存储器传输结束,DMA 产生中断给 DSP,DSP 响应中断,进行下一次中断服务程序的响应。这样循环处理,直到程序退出。

2 语音检测的 DSP 实现

2.1 基于短时能量的语音检测

短时能量是一种常用的语音检测特征,其定义如式(1):

$$E = \sum_{n=1}^N x^2(n) \quad (1)$$

式(1)中 E 表示一帧内信号的能量值, N 为帧长, $x(n)$ 是帧加窗后的语音幅度。语音可以分为清音、浊音两大类。相对类似于白噪声、强度不太大的背景噪声,浊音的短时能量要比噪声的高很多,可以用来区分噪声^[4]。本文采用短时能量的方法来实现实时的语音检测。

如图 2 所示,分为五个状态来实现短时能量的语音检测:开始静音状态、语音可能开始状态、语音状态、语音可能结束

状态、结束静音状态。

首先计算一帧语音能量,和所设置的门限进行比较,如果小于门限,则认为是开始静音状态,当大于该门限时,语音处于语音可能开始状态,当处于语音可能开始状态时,计算一帧语音能量,如果仍然大于门限,计数器加一,当连续三次大于该门限时,认为进入语音状态,计数器清零;如果小于门限值时,返回到开始静音状态。当处于语音状态时,对语音进行存储,同时对能量进行判断处理,如果语音能量大于门限时,存储语音数据,如果小于门限时,依然存储数据,但是语音状态转到语音可能结束状态。如果处于语音可能结束状态,语音能量大于门限时,转到语音状态,语音能量小于门限时,另一个计数器加一,如果连续十次小于能量门限时,则状态转到结束静音状态。当处于语音可能结束状态时,认为语音数据是有用的,并且进行存储。也就是说只在两个状态进行数据的存储,一个是语音状态,一个是语音可能结束状态。以上就是基于短时能量的语音检测的 DSP 实现流程。

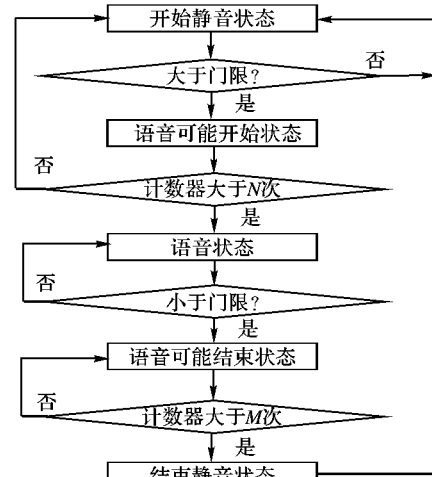


图 2 短时能量语音检测流程

这里首先需要说明的是设置 TLV320AIC23B(CODEC 芯片)的采样率控制寄存器为 8 kHz(其他 CODEC 寄存器配置可参考文献[5]),程序中设置 N 为 3, M 为 10,用 1024 大小的缓冲区来存储 DMA 传来的数据,帧长 256,帧移 64,程序中所选择 N 、 M 都是经验值,实验结果表明,上述配置符合设计要求。能量门限是用以下方法计算得到的,开始处理时,首先对 1024 个输入语音数据进行分帧,计算每帧能量,取其平均值,实验表明这个数据作为门限是不符合设计要求的,Matlab 分析图显示所存储的静音数据太多,为此,我们将该值乘以一个经验系数 4,再一次进行检测,满足设计要求。

2.2 双缓冲加一缓冲实现连续实时语音检测

片内存储器在(DARAM)中申请两个大小为 1024 的缓冲区 DaramBuf1 和 DaramBuf2(如图 1),申请了一个大小为 192 的临时缓冲区 DaramTemp 和一个大小为 256 的处理缓冲区 ProcessBuf,用 ProcessBuf 缓冲区来保持语音的连续性。在 SDRAM(片外存储器)还申请了一个大小为 32 000 的 SdramBuf 缓冲区,用来存储经过语音检测后的数据。1024 大小的两个缓冲区用来存储 DMA 传送过来的数据。192 大小的缓冲区用来存储 ProcessBuf 中的后 192 个数据,而 ProcessBuf 中的前 192 个数据来自于 DaramTemp,后 64 个数据来自于 DaramBuf1 或 DaramBuf2。把 DaramBuf1 或 DaramBuf2 分为 16 帧,每帧 64 个数据,这样相当于每次帧移 64 个数据,和以前的 192 个数据一起构成 ProcessBuf 里的 256 个一帧数据。

程序开始时,DaramTemp 初始化数据为 192 个 0,

ProcessBuf 缓冲区里的 256 个数据初始化也全为 0, 当 DaramBuf1 或者 DaramBuf2 满时, 启动 DMA 中断服务程序, 此时 DaramTemp 做的工作如下: 将自身 192 个点的绝对值进行相加, 并且把自己复制给 ProcessBuf 的前 192 个值, 随后, ProcessBuf 从 DaramBuf1 或 DaramBuf2 中取出 64 个值复制到 ProcessBuf 的后 64 个值, 再把这 64 个点的绝对值和前 192 个点的绝对值之和相加, 得到 256 个点的绝对值之和, 构成一帧数据的能量。用计算出的这个能量值和能量门限进行比较, 此步是将能量值带入到语音检测程序中实现的, 语音检测程序用来对状态位进行标注, 如果符合前述的能够存储语音的状态要求, 则程序对语音数据进行存储, 否则进行下一帧操作。

当一帧处理完之后, 到达下一帧数据的处理阶段, 程序所做的工作如下: 将 ProcessBuf 的后 192 个点的值移动到前 192 个点的位置, 同时复制给 DaramTemp, 并对 DaramTemp 求其绝对值之和。此时, ProcessBuf 所接收到的数据是从第 193 个数据开始, DaramBuf1 或 DaramBuf2 将第二个 64 点数据传送给 ProcessBuf 的后 64 点的位置。再求 ProcessBuf 里的 256 个点的绝对值之和, 之后, 语音检测程序判断, 并重复下去。

当到达 DaramBuf1 或 DaramBuf2 的第 16 帧时, 会有下面的情况发生: DaramTemp 存储了 DaramBuf1 或 DaramBuf2 的后 192 个点的值, 而 DaramBuf1 或 DaramBuf2 后面已经没有数据了, 这样中断服务程序就结束了, 而保留 DaramBuf1 或 DaramBuf2 的后 192 个点的值用来和后面处理的 1024 缓冲区在数据上衔接, 应该说如果只做有用语音的存储, 这样一个 192 缓冲在效果上是没有必要的, 但是如果做 MFCC 等后端处理, 则 192 个大小的缓冲是需要的, 它能够保证语音信号的连贯性。

3 控制板的设计及实现

3.1 控制板原理

按照设计思路, 焊接了一个按键控制板来对程序进行控制, 当按键按下时, 启动程序, 用若干发光二极管来显示程序运行状态。默认电平为高电平, 当按键事件发生时, 则为低电平, 此事件被程序检测到, 于是启动程序运行。为了避免抖动, 做了适当的按键检测延时, 从而实现了程序的控制功能。用不同的发光二极管来显示不同的程序运行状态, 包括如下状态: 门限值计算完毕状态、语音准备播放状态、语音播放结束状态。这些状态可以根据需要在程序中设置, 也可以再增加若干状态。

3.2 按键及控制灯的实现过程

通过对 SEED-DEC5502 外设扩展接口上的 GPIO(通用输入输出)的应用, 合理的对程序进行了控制。

首先定义 GPIO 方向寄存器 IODIR 和 GPIO 数据寄存器 IODATA 接口信息:

```
#define GPIODIR ( *(volatile ioport Uint16 *) (0x3400) )
#define GPIODATA ( *(volatile ioport Uint16 *) (0x3401) )
```

随后配置 GPIO 方向寄存器的引脚方向:

```
GPIO_FSET( IODIR, IOODIR, 0 );
```

注: GPIO_FSET() 为 CSL(运行支持库)已经定义好的函数。

配置哪个引脚作为输出引脚, 哪个引脚作为输入引脚, 这是可选择的。用引脚 GPIO0 作为输入引脚, 用来响应按键事件, 用引脚 GPIO1、GPIO3、GPIO4 作为输出引脚, 用来进行状态显示。最后, 对 GPIO 的数据寄存器 IODATA 相应的域值进行初始化。

程序启动后, 用一个 while 循环来等待按键事件的发生。当按下键后, 程序开始运行, 此时灯 1 亮 (GPIO1), 表示门限值已经计算完毕。对着麦克风说话, 有语音被语音检测程序

检测到, 则进行语音的存储。

需要说明的是, 为了验证语音检测的鲁棒性, 做了三个实验, 实验结果如表 2 所示, 第一个实验通过 PC 机录音“1、2、3、4、5、6”, 字间停顿大概 1 秒, 正确率可达到 96.23%。第二个实验录音“西北工业大学”, 字间停顿大概 1 秒, 正确率可达到 90.40%。第三个实验录音“A、B、C、D、E、F”, 字间停顿大概 1 秒, 正确率可达到 98.31%。然后, 分别存储成 .wav (8 kHz, 16 位、单通道) 文件, 通过主机 Line out 口来接口 SEED-DEC5502 的 Lint in 口, 实现音频输入。实验结果表明 (如表 2 所示), 总的平均正确率可达到 94.98%, 程序能够准确判断出合适的语音起止点以及语音段, 并且将其存储, 当 SdramBuf 满时, 此处, 程序有意做了一个较长时间的延时, 用来提示观察者语音准备播放, 此时灯 2 亮 (GPIO3), 经过一段时间的延时后, 连贯的语音被播放出来, 灯 3 亮 (GPIO4), 表示语音结束, 退出程序。回放出的语音人耳主观听觉良好 (由于 SdramBuf 的大小有限, 表 2 只给出了三个实验的前四个数据的信息)。

表 2 实验结果

样本 1	切分间隔 /ms	检测间隔 /ms	正确率 /%	平均 正确率/%
“1”	327	319	97.55	96.23
“2”	355	347	97.75	
“3”	344	319	92.73	
“4”	323	313	96.90	
样本 2	切分间隔 /ms	检测间隔 /ms	正确率 /%	平均 正确率/%
“西”	519	372	71.68	90.40
“北”	462	455	98.48	
“工”	436	408	93.58	
“业”	467	457	97.86	
样本 3	切分间隔 /ms	检测间隔 /ms	正确率 /%	平均 正确率/%
“A”	480	479	99.79	98.31
“B”	422	417	98.82	
“C”	419	413	98.57	
“D”	457	439	96.06	

注: 总正确率为 94.98%。

4 结果分析

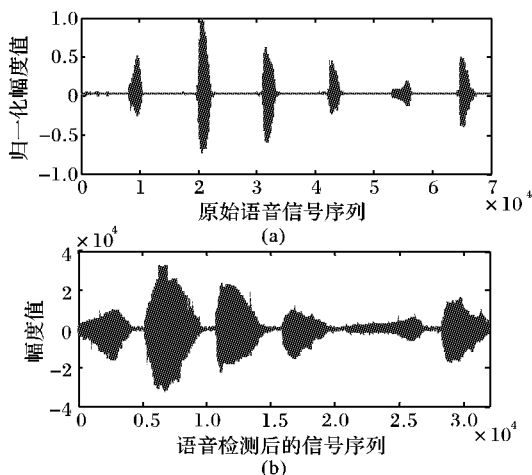


图 3 原始语音及检测后的语音比较

图3为第一个实验的语音比较图, 图3(a)为发音“1、2、

(下转第 498 页)

的零件成形质量优于常压边力作用方案得到的零件成形质量。由此可见,只要冲压回弹仿真计算准确,利用上述反演方法可有效地提高冲压件的成形质量,节省大量反复试模修模过程所耗费的人力和物力。

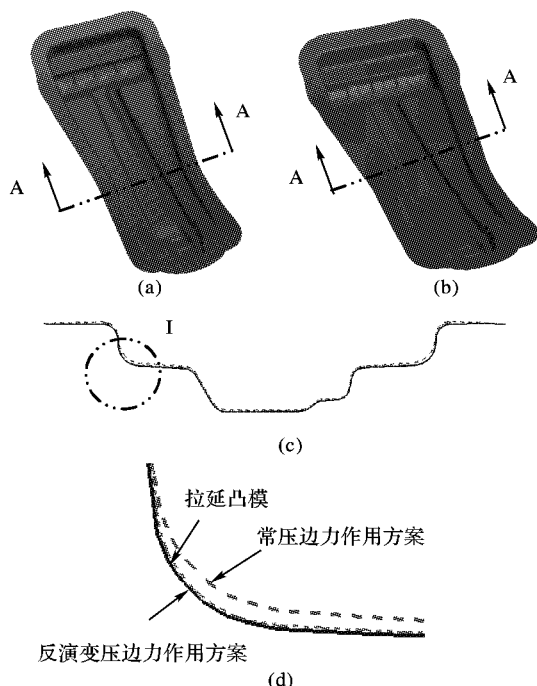


图 6 压边力反演前后零件成形情况对照

3 结语

在冲压件成型中,变压边力策略是一种控制回弹的有效措施,而小压边力 F_{BH1} 、大压边力 F_{BH2} 以及压边力变化时刻 t_1 的确定是变压边力策略中的关键。本文综合运用 RBF 神经网络和三维回弹模拟技术由冲压件成形质量反求变压边力。隐节点数、各基函数的数据中心以及扩展常数的确定是 RBF 神经网络学习中应解决的关键问题。针对此问题,本文提出了一种基于泛化的径向基函数神经网络的动态结构设计方法 DYNDRBF,编制了相应的计算程序。该方法将 RBF 神经网络学习问题分粗调阶段和精调阶段为两个阶段。在粗调阶段,只按正交最小平方方法动态增加隐节点的数目及选取相应的样本输入作为数据中心,直至隐节点数满足一定要求;在精调阶段,用改进 BP 算法对粗调得到的 RBF 神经网络进行学习,其中包括数据中心、扩展常数和连接权值的动态调整等。调整过程中,综合运用神经网络泛化能力的多种改进方法来

提高网络的泛化能力和鲁棒性。在以上两个阶段中所涉及的扩展常数的初值、给定的最多隐节点数和正则化系数则运用 IP- μ GA 算法来确定。DYNDRBF 方法在变压边力反演神经网络设计中的应用结果表明,运用该方法设计的神经网络不仅具有较好的训练精度,而且具有较强的泛化能力和鲁棒性。变压边力反演研究结果表明,只要冲压回弹仿真计算准确,利用上述求解方法可有效地提高冲压件的成形质量,节省大量反复试模修模过程所耗费的人力和物力。

参考文献:

- [1] LIU Y C. The effect of restraining force on shape deviation in flanged channels [J]. Journal Engineering Materials Technology, 1988, 110: 389-394.
- [2] SCHMOECKEL D, BEIH M. Springback reduction in draw-bending process of sheet metals [J]. Annals of the CIRP, 1993, 42(1): 339-342.
- [3] SCHMOECKEL D, BEIH M, SPRINGBACK A P. Accommodation of springback error in channel forming using active binder force control: numerical simulations and experiments [J]. Journal of Engineering Materials and Technology, Transaction of the ASME, 1996, 118(3): 426-435.
- [4] 林忠钦, 刘昱, 李淑慧, 等. 应用正交试验设计提高 U 形件的成形精度 [J]. 机械工程学报, 2002, 38(3): 83-89.
- [5] 刘昱. 基于回弹控制的提高轿车冲压件成形精度方法研究 [D]. 上海: 上海交通大学, 2001: 1-20.
- [6] 刘昱, 林忠钦, 李淑慧. 变压边力优化设计提高仪表板支架冲压成形精度 [J]. 塑性工程学报, 2002, 9(2): 56-59.
- [7] 阎平凡, 张长水. 人工神经网络与模拟进化计算 [M]. 北京: 清华大学出版社, 2000.
- [8] INAMDAR M V, DATE P P, DESAI U B. Development of an artificial neural network to predict springback in air vee bending [J]. International Journal of Advanced Manufacturing Technology, 2000, 16(5): 376-381.
- [9] LIU G R, HAN X. Computational inverse techniques in nondestructive evaluation [M]. Boca Raton: CRC PRESS, 2003: 1-222.
- [10] 张智星, 孙春在, 水谷英二. 神经——模糊和软计算 [M]. 西安: 西安交通大学出版社, 2000: 308-309.
- [11] 徐秉铮, 张百灵, 韦岗. 神经网络理论与应用 [M]. 广州: 华南理工大学出版社, 1996.
- [12] VOGL T P, MANGIS J K. Accelerated the convergence of the back-propagation method [J]. Biological Cybern, 1988, 59(9): 256-264.

(上接第 493 页)

3、4、5、6”的归一化的波形图,图 3(b)为经 DSP 处理后 SDRAM(片外存储器)中存储的未归一化的语音数据,从图(a)可以看出,每个数据语音中间做了较长时间的停顿,以便更好地验证语音检测程序的准确性。语音间隔中的长时间静音数据是所不希望存储的语音信息,经过语音检测程序的处理,存储了有用语音信息,图 3(a)为数字“1、2、3、4、5、6”,由于所申请的 SDRAM(片外存储器)容量有限,所以存储的数据也是有限的,但还是利用了有限的空间存储了较多的语音数据。有一点不足,在做第二个实验“西北工业大学”时,发现“西”的正确率很低,只有 71.68%(如表 2 所示),原因在于“西”的发音第一个字母为清音“x”,在不影响实验结果的情况下,只用了短时能量的方法来实现语音检测,要解决清音正确率低的问题,还要结合其他方法,短时过零率是目前用到的比较多的方法。

5 结语

本文提出了一种基于 DSP 的实时语音检测的方法,通过 DSP 和 DMA 的并行工作,对语音进行检测和存储。并利用自制控制板,对程序进行控制和状态显示。运用本文的方法完成了三个实验,实验结果表明,本文的方法实现了基于 DSP 的实时语音检测。

参考文献:

- [1] 北京合众达公司. SEED-DEC5502 用户指南 (Rev. B) [K], 2005: 14-17.
- [2] TI. TMS320C55x Chip Support Library API Reference Guide (Rev. I) [K], 2004: 62-82.
- [3] TI. TMS320VC5501/5502 DSP Direct Memory Access (DMA) Controller Reference Guide (Rev. G) [K], 2005: 38-67.
- [4] 赵力. 语音信号处理 [M]. 北京: 机械工业出版社, 2003: 35-36.
- [5] TI. TLV320AIC23B Data Manual (Rev. G) [K], 2003: 27-28.