

# 嵌入式语音信号处理实验系统的设计与实现

梁瑞宇<sup>1,2</sup>, 王青云<sup>1</sup>, 赵力<sup>2</sup>

(1. 南京工程学院 通信工程学院, 南京 211167; 2. 东南大学 信息科学与工程学院, 南京 210096)



**摘要:**为体现教学与科研相互促进的教学理念,面向语音信号处理教学和科研需要,以 Cortex-A8 微处理器和专用音频编解码芯片 WM8960 为硬件核心,构建一款嵌入式语音信号处理综合实验平台。基于该实验平台,利用 QT 开发软件,设计一款语音信号处理综合实验教学与开发软件。文中完整地讲述了系统的硬件构成,驱动程序的移植,并结合基于谱减法的语音增强算法,介绍了基于 QT 的语音开发流程。基于该实验平台,学生不仅可以掌握语音信号处理的基本知识,还可以熟悉面向语音应用的嵌入式软件开发流程。系统界面友好,功能强大,可有效用于高校语音及信号处理教学工作,也可用于相关科研人员进行语音信号处理相关课题的研发工作。

**关键词:**语音信号处理; 语音增强; 嵌入式; 实验教学

中图分类号: TN 912.3; G 642.0 文献标志码: A 文章编号: 1006-7167(2017)05-0126-05

## Design and Implementation of an Experimental System for Embedded Speech Signal Processing

LIANG Ruiyu<sup>1,2</sup>, WANG Qingyun<sup>1</sup>, ZHAO Li<sup>2</sup>

(1. School of Communication Engineering, Nanjing Institute of Technology, Nanjing 211167, China;  
2. School of Information Science and Engineering, Southeast University, Nanjing, 210096, China)

**Abstract:** To embody the idea of mutual promotion between teaching and scientific research, and to meet the needs of teaching and research for speech signal processing, an integrated experimental platform for embedded speech signal processing is design by using Cortex-A8 microprocessor and WM8960 audio coding and decoding chip as core hardware. Moreover, an integrated experimental teaching and development software is designed on this platform by using QT development software. Then, the system hardware structure and transplant of driver are described. By taking an example of the speech enhancement algorithm based on spectral subtraction, the speech development process based on QT is introduced. Based on this experimental platform, students can master not only the basic knowledge of speech signal processing, but also the development process of embedded software for the speech application. The system had friendly interface and strong functions, can be used for the teaching of speech and signal processing in colleges and universities, and for the research and development of related topics on speech signal processing.

**Key words:** speech signal processing; speech enhancement; embedded; experimental teaching

收稿日期: 2016-09-15

**基金项目:**江苏省“六大人才高峰”项目(2016-DZXX-023); 江苏省青蓝工程, 江苏省高等教育教学改革研究课题(2015JSJG123); 南京工程学院高等教育研究课题重点课题(2015ZD02)

**作者简介:**梁瑞宇(1978-), 男, 江苏徐州人, 博士, 副教授, 主要研究方向为语音信号处理。Tel.: 15380920360; E-mail: lly1711@163.com

## 0 引言

随着人机交互和智能机器人技术的发展, 语音信号处理已成为信息科学研究领域中发展最为迅速的的一个分支。语音信号处理是用数字信号处理技术和语音学知识对语音信号进行处理的一门新兴学科, 是一门

理论性强、实用面广、内容新、难度大的交叉学科<sup>[1-2]</sup>。语音信号处理作为信息处理专业的一门重要的专业课,所涉及的理论知识较多。语音信号处理包括语音增强<sup>[3]</sup>、语音编码、语音合成、语音识别<sup>[4]</sup>、说话人识别<sup>[5]</sup>、情感识别<sup>[6]</sup>、语音隐藏、声源定位<sup>[7]</sup>等。围绕着这些应用,该课程还会涉及一些相关理论,如矢量量化、隐马尔科夫模型、高斯混合模型,支撑向量机等。因此,如何激发学生的学习兴趣,从枯燥的理论中升华出来,是提升课堂教学质量的关键。

目前,很多高校的语音信号处理课程的授课时间在32~48学时。在短暂的课堂授课时间内,使学生对各种应用有所了解,并从中选择一到两种进行深入研究,是非常困难的一件事。为此,很多高校都会增加实验环节使学生对课堂内容进行巩固和吸收。但是,目前语音信号处理实验主要以MATLAB仿真为主<sup>[8]</sup>,虽然有助于学生快速理解理论知识,但是缺乏对语音如何采集、处理和播放整个流程的深入了解。因此,开发出一款功能强大且实用的语音信号处理综合实验平台,为学生和老师从事教学科研工作提供方便,已经成为语音信号处理教学工作的重中之重。一个有效的实验平台不但可以使学生掌握理论知识,还可以使学生了解语音信号处理的整个开发流程,为学生后期从事语音信号处理科创、竞赛或工作积累经验。针对目前语音教学科研平台较少的现状<sup>[9-10]</sup>,本文按照主流的嵌入式系统开发设计理念,研发了一款语音信号处理综合实验平台,介绍了平台的硬件构成和软件实现,并通过语音增强实验验证了平台的有效性。

## 1 系统构成

嵌入式语音实验平台的硬件结构如图1所示,实验平台由S5PV210核心处理器、WM8960语音编解码芯片、存储电路、扬声器、麦克风、外设接口、电源和时钟电路构成。其中,电源和时钟电路提供系统工作必需的电源和时钟。外设接口包括串口、USB接口、网络接口,用来与电脑通信,实现人机交互等功能。

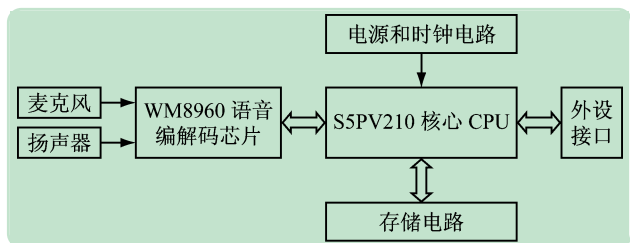


图1 嵌入式语音实验平台硬件结构图

实验平台的核心处理器选用Samsung公司高端的ARM Cortex-A8微处理器S5PV210芯片。S5PV210基于ARM v7指令架构,适用于复杂操作系统及用户应

用,芯片功耗在300 mW以下,而性能却高达2 000 MIPS<sup>[11-12]</sup>。S5PV210微处理器具有复杂的流水线架构,带有先进的动态分支预测,可实现2.0 DMIPS/MHz。

语音编解码芯片选用欧胜微电子公司的低功耗、高质量的立体编解码器芯片WM8960。该芯片内置有麦克风接口和立体声耳机驱动器以及立体声24 bit sigma-delta模数转换器和数模转换器。其运行的模拟电源电压低至2.7 V,数字内核运行电压可低至1.7 V。其高级的片上数字信号处理能够实现麦克风输入的自动电平控制,并支持多种采样率。该芯片与麦克风和扬声器进行配合,可实现语音的采集和播放。

存储电路由DDR2 SDRAM芯片K4T1G164QQ和NAND Flash芯片K9F2G08UOD构成。系统采用4片16位128MB的SDRM芯片来构成32位的数据接口,总大小为512MB。DDR2 SDRAM相比于老式的DDR SDRAM具有更高的运行速率与更低的功耗。SDRAM主要用来存储临时数据,而NAND Flash主要用来存储根文件系统、boot代码、系统内核代码和简单的数据。NAND Flash是连续存储介质,适合存放大的数据量,其共用地址和数据总线,对一个固定大小的区域进行操作,且使用寿命长。

## 2 WM8960 驱动移植

因为嵌入式Linux的开源性、可裁剪性和可移植性,所以其成为嵌入系统开发最常用的操作系统之一。在Linux操作系统中,内核主要用来管理软硬件资源,并提供运行环境。要想通过Linux系统来对硬件进行操作,必须在硬件和内核之间架设一个通道,即Linux设备驱动。平台以WM8960作为语音采集和播放芯片,因此下面简单介绍WM8960芯片的驱动移植过程。

由于WM8960语音芯片是采用模块的方式编译进Linux内核系统的,故其驱动程序的入口和出口函数分别为WM8960\_modinit, WM8960\_exit。所谓的入口和出口函数指的是:当系统上电时,内核会将所有的驱动代码挂载到系统里,而加载驱动首先执行的就是驱动的入口函数。而当卸载不想要的驱动时,执行的是对应驱动的出口函数。

入口函数只是将WM8960设备对应的数据结构注册到I<sup>2</sup>C总线中。采用的方式就是调用I<sup>2</sup>C总线的接口函数i2c\_add\_driver()。为增强驱动的移植性,内核采用了驱动-设备分离的方式,即与硬件相关的信息放在一个结构体里注册进内核,而与硬件无关的信息放在另一个结构体中注册进内核。驱动和设备二者通过匹配注册的name来进行识别,此处的注册name为WM8960。

当驱动层和设备层通过name匹配成功之后,第

一个会执行的函数就是 probe 函数。probe 函数会判断是否是负责驱动的设备,然后完成该设备的初始化操作。代码如下:

```
static __devinit int WM8960_i2c_probe(struct i2c_client *i2c,
    const struct i2c_device_id *id) {
    struct WM8960_priv *WM8960; //WM8960 的私有数据结构体
    struct snd_soc_codec *codec; //Codec 结构体
    WM8960 = kzalloc( sizeof( struct WM8960_priv ), GFP_KERNEL); //分配空间
    if ( WM8960 == NULL )
        return -ENOMEM;
    codec = &WM8960->codec;
    i2c_set_clientdata( i2c, WM8960 );
    codec->control_data = i2c;
    codec->dev = &i2c->dev;
    return WM8960_register( WM8960, SND_SOC_I2C ); //注册
}
```

probe 函数首先定义了两个非常复杂数据结构:一个 WM8960 的私有数据结构体 WM8960,另一个是片上 Linux 音频架构中的 Codec 结构体 codec。然后,程序调用 Linux 内核内存分配接口 kzalloc 函数分配了一个空间,并且将空间的首地址交给了 WM8960 私有数据结构体。为了让其他函数也能使用该内存,此处通过 i2c\_set\_client 函数将地址保存到 i2c\_client 结构体下面的一个 void \* 指针里。这样,其他函数可以通过 i2c\_get\_client 函数接口来得到保存的数据。probe 函数最后调用 WM8960\_register 函数来完成 WM8960 的注册。

完成上述驱动添加和注册工作后,音频文件的采集和播放就可以通过调用 Linux 系统的声卡驱动,再调用刚注册的 WM8960 音频编解码芯片来完成。通常,在进行语音信号采集与处理前,主程序需要对音频采集参数进行设置,主要包括采样率的设置、单双通道的配置、录音与播音缓冲区大小的设置、采样位数的设置等。配置完成后,算法便可以从录音缓冲区读取采集的语音数据,处理完成后,送入语音播放缓冲区。

录音程序的核心代码如下:

```
.....
int SIZE = 8;
int CHANNELS = 1;
int RATE = 8000;
int LENGTH = 3;
if ( snd_open( SND_PCM_STREAM_CAPTURE, CHANNELS,
    RATE, SIZE ) < 0 ) {
    fprintf( stderr, "设备打开失败 = \n" );
    return; }
size_t c, frame_size;
c = ( off64_t ) capture.chunk_bytes; //定义块的大小
```

```
int count = ( LENGTH * RATE ) / c; //计算块的数量
unsigned char data_buff[ count * c ]; //定义数据缓存大小
frame_size = c * 8 / capture.bits_per_frame; //求帧的数量
for( int i = 0; i < count; i ++ ) {
    if ( SND_readPcm( &capture, frame_size ) != frame_size ) {
        fprintf( stderr, "读取错误\n" );
        return; }
    memcpy( data_buff + i * c, capture.data_buf, c ); //从数据缓冲区读出数据
}
QVector<double> Y1( count * c );
for( int i = 0; i < Y1.size(); i ++ )
    Y1[ i ] = ( data_buff[ i ] - 128 ) / 128.0; //预处理
snd_close( SND_PCM_STREAM_CAPTURE ); //关闭设备
.....
```

### 3 语音信号处理实验设计

#### 3.1 实验设计

本文设计的语音信号处理综合实验平台功能比较强大,可以面向不同层次的学生进行实验。教师可以根据学生基础,为学生量身定制实验套餐。如表 1 所示,实验主要分为 3 类。

表 1 实验套餐及适用学生

套餐类型	提供资料*	学生条件*	适用实践课类型
1	I	A, B, C	课程设计或课外实验
2	I, II	B, C	课外实验
3	I, II, III	C	课堂实验

注:字母说明——学生条件:A. 嵌入式底层程序开发基础;B. QT 编程基础;C. C 或 C++ 语言基础。提供资料——I. 实验原理及要求;II. 编译好内核;III. QT 界面及函数定义

由表可知,根据学生的学习基础,老师可以选择不同的套餐进行实验设置,而学生可以跨过不同阶段进行超前学习。不同于 MATLAB 的教学方法和实验设置,该实验设置并不固定在一中单一模式下,可进行选择性扩展和加深。通过分层次的设置,教师可以通过不同层次的实例演示来激发学生的学习兴趣,加深学生对知识的理解,也为学生从事语音信号处理相关的实际课题或竞赛提供一种切实可行的实现方案和思路。对于电子类学生来说,C 语言或 C++ 是重要和普遍的基础课,因此学生进行实验困难不大,还可以锻炼学生 C 语言的编程能力。下面以语音增强算法为例,介绍面向套餐 3 的学生实验安排情况。

#### 3.2 语音增强算法原理

对于任何实验来说,学生都必须首先了解实验的基本原理。因此,简单介绍基于谱减法的语音增强算法的基本原理。

语音增强主要研究如何利用信号处理技术消除信号中的强噪声干扰,从而提高输出信噪比以提取出有



用信号的技术。语音增强是语音算法中最基本,最常用的算法之一,其主要目的:①改进语音质量,消除背景噪音,使听者乐于接受,不感觉疲劳,这是一种主观度量;②提高语音可懂度,这是一种客观度量。这两个目的往往不能兼得,所以实际应用中总是视具体情况而有所侧重。根据语音和噪声的特点,常用的语音增强算法包括谱减法、维纳滤波法、卡尔曼滤波法、自适应滤波法等。考虑到学生的理解程度,本文以谱减法<sup>[13]</sup>为例来进行说明。整个算法的原理如图2所示。

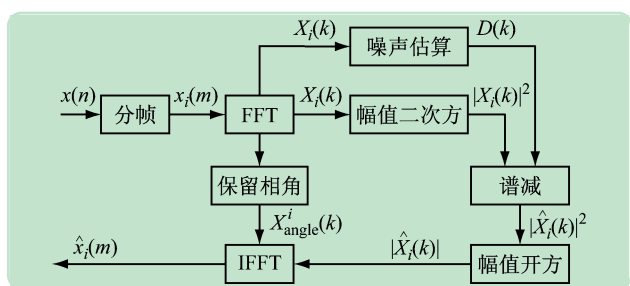


图2 谱减法原理图

设语音信号的时间序列为  $x(n)$ , 加窗分帧处理后得到第  $i$  帧语音信号为  $x_i(m)$ , 帧长为  $N$ 。任何一帧语音信号  $x_i(m)$  做 FFT 后为

$$X_i(k) = \sum_{m=0}^{N-1} x_i(m) \exp\left(j \frac{2\pi mk}{N}\right) \quad (1)$$

$$k = 0, 1, \dots, N-1$$

对  $X_i(k)$  求出每个分量的幅值和相角, 幅值是  $|X_i(k)|$ , 相角为

$$X_{\text{angle}}^i(k) = \arctan\left[\frac{\text{Im}(X_i(k))}{\text{Re}(X_i(k))}\right] \quad (2)$$

已知前导无话段(噪声段)时长为  $IS$ , 对应的帧数为  $NIS$ , 可以求出该噪声段的平均能量为

$$D(k) = \frac{1}{NIS} \sum_{i=1}^{NIS} |X_i(k)|^2 \quad (3)$$

谱减公式为

$$| \hat{X}_i(k) |^2 = \begin{cases} |X_i(k)|^2 - a \times D(k), & |X_i(k)|^2 \geq a \times D(k) \\ b \times D(k), & |X_i(k)|^2 < a \times D(k) \end{cases} \quad (4)$$

式中:  $a$  和  $b$  是两个常数,  $a$  称为过减因子,  $b$  称为增益补偿因子。

谱减后的幅值为  $| \hat{X}_i(k) |$ , 结合原先的相角  $X_{\text{angle}}^i(k)$ , 就可利用快速傅里叶逆变换求出增强后的语音序列  $\hat{x}(m)$ 。

### 3.3 基于 QT 的算法实现

此处选择 QT 作为编程载体目的:①为了便于学生直观观测结果;②QT 是优秀的跨平台 C++ 图形界面应用程序开发框架。基于 QT 进行 Linux 下的图形

界面开发,可以大大提高 Linux 下的编程效率<sup>[14-16]</sup>。

在没有 QT 编程的基础下,教师可以选择套餐 3 开展实验。此时,套餐 3 提供的基本资料包括:添加好语音采集设备的 Linux 内核,构建好的 QT 执行界面,QT 的运行说明文档以及针对该实验的函数定义:  $\text{QVector}\langle \text{double} \rangle \text{ssubspec}(\text{QVector}\langle \text{double} \rangle \&\text{signal}, \text{int wlen}, \text{int inc}, \text{int NIS}, \text{double } a, \text{double } b)$ 。这里,谱减法的实现函数为  $\text{ssubspec}$ , 其输入变量包括  $\text{signal}$  (输入语音信号),  $\text{wlen}$  (帧长),  $\text{inc}$  (帧移),  $\text{NIS}$  (前导帧数),  $a$  (过减因子),  $b$  (增益补偿因子)。基于谱减法原理设计的语音增强函数如下:

```
QVector<double> ssubspec(QVector<double>&signal, int wlen, int inc, int NIS, double a, double b)
{
    QVector<double> wnd = hamming(wlen); //设置窗函数
    QVector<double> output, X; //谱减后的语音序列和幅值
    QVector<double> Yphase, Yeng; //谱减前相位和能量
    int N = signal.size(); //信号长度
    int nf = (N-wlen)/inc + 1; //帧数
    QVector<double> Y = enframe(signal, wnd, wlen, nf, inc); //信号分帧
    //噪声段平均能量,参考式(3)
    QVector<double> D = noiseeng(Y, NIS, wlen);
    //计算信号相位
    for(int i = 0; i < nf; i++) {
        std::complex<double> f[wlen];
        for(int j = 0; j < wlen; j++) {
            f[j].real() = Y[i * wlen + j];
            f[j].imag() = 0;
        }
        //求信号的傅里叶变换,参考式(1)
        fft(f, wlen);
        for(int j = 0; j < wlen/2 + 1; j++) {
            //求信号相位,参考式(2)
            Yphase << angle(f[j]);
        }
        for(int j = 0; j < wlen; j++) {
            Yeng << abs(f[j]) * abs(f[j]);
        }
        //谱减法降噪,参考式(4)
        for(int j = 0; j < wlen/2 + 1; j++) {
            if(Yeng[j] >= a * D[j]) {
                X << sqrt(Yeng[j] - a * D[j]);
            }
            else {
                X << sqrt(b * Yeng[j]);
            }
        }
        output = OverlapAdd2(X, Yphase, wlen, inc, nf); //合成谱减后的语音
    }
    return output;
}
```

当输入为一段信噪比为 5 dB 语音信号时,谱减法语音增强后的效果如图 3 所示。显示的软件为语音信号处理平台的综合性实验软件(此处为实验平台的完

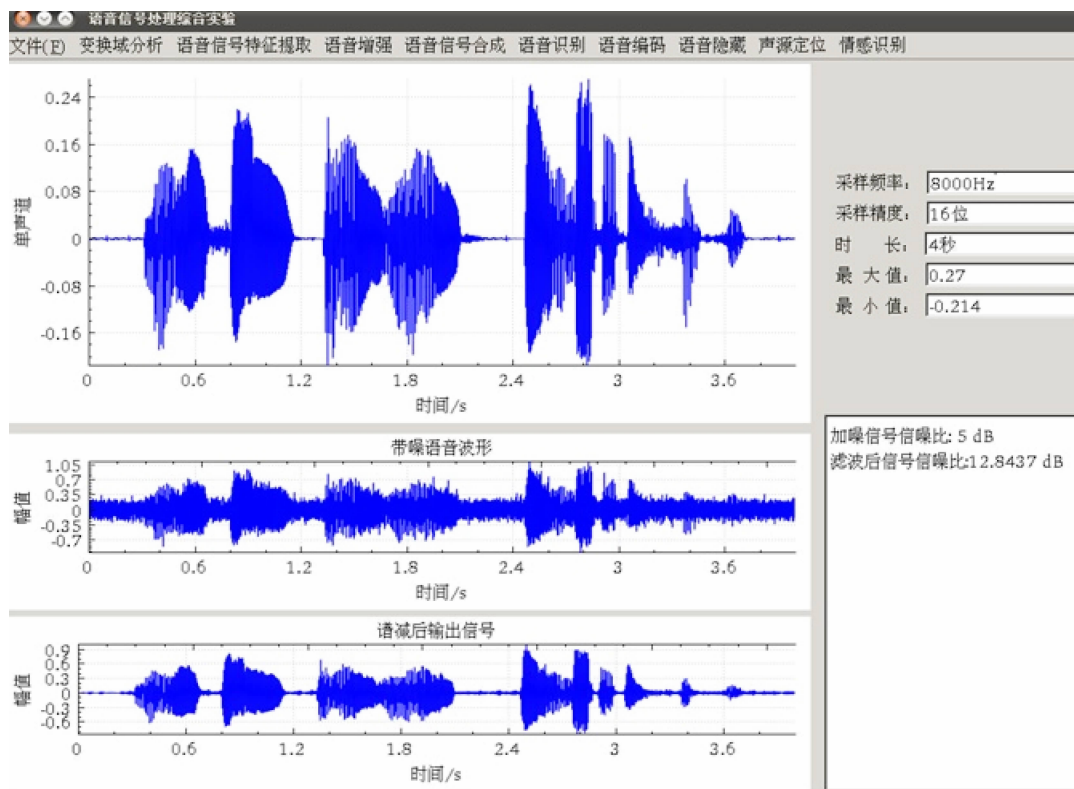


图 3 谱减法语音增强效果图

整界面,在单独实验时可根据需要进行裁剪)。菜单栏显示了平台可以完成的实验,中心区为波形显示区域,右边为参数显示区域。中心区的上部为原始语音信号,下部分两段,分别为带噪的语音波形和谱减后的输出信号。由图可知,增强后的信号信噪比由 5 dB 变为 12.843 7 dB,提高了 7.843 7 dB,改善效果比较明显。从信号波形也可看出,谱减后的波形较带噪语音有了明显改善。

#### 4 结 论

针对目前语音信号处理课程偏重于理论教学的现状,本文秉承学以致用原则,开发了一款基于嵌入式系统的语音信号处理综合实验平台。本文重点介绍了系统的硬件构成和软件设计架构,并通过基于谱减法的语音增强实验,验证了平台的性能有效性和界面友好性。此外,该平台涉及了语音信号处理目前的主流研究方向,因此既可用于高等学院语音信号处理课程的教学工作,也可以满足从事语音信号处理相关科研人员学习和研发的需要。

#### 参考文献 (References):

- [1] 刘卫东,孟晓静,王艳芬. 语音信号处理实验教学研究探索[J]. 实验室研究与探索, 2008, 27(4): 72-74.
- [2] 赵 力,梁瑞宇,魏 昕,等. 语音信号处理[M]. 3 版. 北京: 机械工业出版社, 2016.
- [3] Deng F, Bao C, Kleijn W B. Sparse hidden markov models for speech enhancement in non-stationary noise environments [J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2015, 23(11): 1973-1987.
- [4] 周 健,赵 力,梁瑞宇,等. 基于噪声鲁棒性特征和 SVM 的耳语音可懂度增强[J]. 东南大学学报(英文版), 2012, 28(3): 261-265.
- [5] Zhang X, Zhang Y, Shi Y, et al. Power control algorithm in cognitive radio system based on modified Shuffled Frog Leaping Algorithm [J]. International Journal of Electronics and Communications, 2012, 66(6): 448-454.
- [6] 王 薇,杨丽萍,魏 丽,等. 语音情感特征的提取与分析[J]. 实验室研究与探索, 2013, 32(7): 91-94,191.
- [7] 张潇丹,胡 峰,赵 力. 基于改进的蛙跳算法与支持向量机的实用语音情感识别[J]. 信号处理, 2011, 27(5): 678-689.
- [8] 杨 毅,李泽伟,邓北星,等. 语音信号处理实验的改革与实践[J]. 实验室研究与探索, 2014, 33(4): 123-126.
- [9] 伍技祥,仲元昌,李彩玲,等. 嵌入式数字语音处理实验系统的设计与实现[J]. 实验室研究与探索, 2012, 31(10): 46-49.
- [10] 宋亚男,林锡海,徐荣华,等. 机器人语音识别实验设计与实现[J]. 实验技术与管理, 2013, 30(2): 36-38.
- [11] 杨继森,张 静,朱 革,等. 基于 ARM Cortex 平台的嵌入式系统课程虚拟实验平台设计[J]. 实验技术与管理, 2014, 31(7): 97-101.
- [12] 王青云,梁瑞宇,冯月芹. ARM Cortex-A8 嵌入式原理与系统设计[M]. 北京: 机械工业出版社, 2014.
- [13] Boll S F. Suppression of acoustic noise in speech using spectral subtraction[J]. IEEE Transactions on Acoustics, Speech and Signal Processing, 1979, 27(2): 113-120.
- [14] 房汉雄,穆伟斌,齐 迹,等. 移动终端实验教学平台设计与应用[J]. 实验技术与管理, 2014, 31(8): 103-105.
- [15] 田 磊. 嵌入式 Linux 系统中基于 QT 库的应用程序设计[J]. 实验室研究与探索, 2014, 33(5): 84-86,115.
- [16] 赵明波,谢 楠,王亚军,等. 基于 RFID 和 QT/E 车辆出入监控系统的设计[J]. 实验室研究与探索, 2014, 33(11): 95-99.