

基于 Android 系统振动测试仪研究

Research on Android System Based on Vibration Test Instrument

吴立勇 (杭州雷威电器有限公司, 浙江 杭州 311300)

丁作文 (河南神火煤电股份有限公司, 河南 商丘 476600)

摘 要

主要研究搭载 Android 操作系统的智能手机软件的开发,介绍 Android 开发平台,针对振动测试仪的设计,介绍了实现的具体方法和实际步骤,并且在实际设备上进行了测试。

关键词:Android,Java,手机,振动测试仪

Abstract

The development of research with the Android operating system of intelligent mobile phone software, introduced the Android development platform, the design of vibration test instrument, introduces the steps and practical realization method, are tested and in the actual equipment.

Keywords:Android,Java,mobile phone,vibration tester

基于 Android 系统的开源优势,使得 Android 软件开发也变得更加容易,随着智能手机系统的不断普及,手机已经不仅仅是为了满足人们日常作为打电话的功能,而且所开发的软件设计出的产品已经能代替其他实物的功能,本文设计的就是一个款基于 Android 系统的手机振动仪,它能准确地测试出物体上下振动次数。

1 Android 开发简介

Android 是由 Google 公司推出的,以 Linux 为核心的开放源代码操作系统^[1],可以运用在电脑、手机、电视机、洗衣机等各种设备上。随着 Android 版本的不断更新,已经有越来越多的制造商都投入到 Android 系统开放的研究中。

Android 开发环境一般是基于 JAVA 语言进行开发的,开发平台是基于 Eclipse 软件开发的,并且可以下载到模拟器中查看软件实际的运行效果,最终再下载到整机中进行最终的测试^[2]。本文所开发出是软件就是在以上环境下进行开发的,并且在真机的环境下测试成功,实际使用效果也是非常满意。

2 振动仪的原理

Android 平台支持多种传感器,如:温度传感器、加速度传感器、加速度传感器、方向传感器等。表 1 为 Android 支持的传感器。

表 1 Android 支持的传感器

说明	传感器名称	本地接口名称	数值
加速度	TYPE_ACCELEROMETER	SENSOR_TYPE_ACCELEROMETER	1
磁力域	TYPE_MAGNETIC_FIFLD	SENSOR_TYPE_MAGNETIC_FIELD	2
方向	TYPE_ORIENTATION	SENSOR_TYPE_ORIENTATION	3
陀螺	TYPE_GYROSCOPE	SENSOR_TYPE_GYROSCOPE	4
光感	TYPE_LIGHT	SENSOR_TYPE_LIGHT	5
压力	TYPE_PROSSURE	SENSOR_TYPE_PROSSURE	6
温度	TYPE_TEMPERATURE	SENSOR_TYPE_TEMPERATURE	7
距离	TYPE_PROXIMITY	SENSOR_TYPE_PROXIMITY	8

本文就是加速度传感器检测被测设备的各个方向的加速度变化,通过加速度传感器的使用实现一个测试设备振动的功能,这样用户就不需要专门携带一个振动测试仪^[3],只要在移动设备上开启这个功能就可以方便地使用了。

3 系统设计与实现

3.1 系统控件的设计

论文在界面中两个 TextView 控件和一个 Button 控件,其中一个 TextView 控件用于显示设备上下振动的次数,Button 控件用控制是否需要检测物体振动次数的检测。

```
<LinearLayout
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <TextView
        . . . . .
        android:text="统计上下振动次数"
    />
    <TextView
        android:id="@+id/step"
        . . . . .
        android:text="0"
    />
    <Button
        android:id="@+id/controlBtn"
        . . . . .
        android:text="开始"
    />
</LinearLayout>
```

3.2 系统实现设计

在主程序中首先对每个控件进行绑定,绑定后的控件用于显示传感器检测到的读数。同时需要为传感器创建 SensorManger 对象,在 onCreate()方法中给 SensorManager 对象初始化^[4-6]。还需要重写 onResume()方法给传感器对象注册侦听器,给传感器对象指定一个类型,本文为 SensorManager.SENSOR_ACCELEROMETER,最后在 onSensorChanged()方法中获取传感器的读书并且显示在相应的 TextView 控件上。最后还要重写 onPause()方法在程序暂停的时候注销掉所有的 SensorManager 对象的侦听器。

考虑到设备上下振动的时候各个方向的加速度会变化,在 onSensorChanged() 方法中,把三个方向的加速度作为一个整体,用一个三维空间向量表示。加速度的变化量的模作为指标,在程序中自定义 magnitude() 方法就是求取加速度向量的模。

程序的具体实现代码如下所示:

```
.....
public class MainActivity extends Activity {
    private SensorManager _accSensorManager;
    private int _step=0;
    private double _oriValue=0;
    private double _lstValue=0;
    private double _curValue=0;
    private Boolean _motiveState=true;
    private Boolean _processState=false;
    private TextView _stepText=null;
    private Button _ctlBtn=null;
    protected void onCreate(Bundle savedInstanceState) {
        .....
        _accSensorManager =(SensorManager) getSystemService
        (SENSOR_SERVICE);
        _stepText=(TextView)findViewById(R.id.step);
        _ctlBtn=(Button)findViewById(R.id.controlBtn);
        _ctlBtn.setOnClickListener(new OnClickListener()
        {
            .....
        });
        @SuppressWarnings("deprecation")
        private SensorListener mySensorListener=new SensorListener()
        {
            public void onAccuracyChanged(int sensor,int accuracy){}
            public void onSensorChanged(int sensor,float[] values)
            {
                .....
            }
        };
        protected void onResume()
        {
            _accSensorManager.registerListener(mySensorListener,
            SensorManager.SENSOR_ACCELEROMETER,SensorManager.SEN-
            SOR_DELAY_UI);
            super.onResume();
        }
        protected void onPause()
        {
            _accSensorManager.unregisterListener(mySensorListener);
            super.onPause();
        }
        private double magnitude(float x, float y, float z) {
            // TODO Auto-generated method stub
            double magnitude=0;
            magnitude=Math.sqrt(x*x+y*y+z*z);
            return magnitude;
        }
    }
}
```

(上接第9页)

3 结束语

本项目利用 LabVIEW 提供的 Touch Panel 模块完成基于 WinCE 操作系统的仿真应用软件的开发,在实际开发过程中开发人员可不必具体了解 UDP 通讯协议,语音编码的细节,即可完成上层软件的构架,提高了软件的开发效率。

在软件的开发过程中还有几点注意:①如果在程序中使用到全局变量,一定要注意它的读写顺序,否则程序在执行时会出现异常状况;②在利用 Draw Text at Point 子 VI 向图片控件写字符串时,user-specified font 中的字体大小只是纵向像素,而

```
.....
magnitude=Math.sqrt(x*x+y*y+z*z);
return magnitude;
}
.....
}
```

4 测试结论

将编写调试后的程序 SensorTest 文件安装到华为 C8812 手机上运行结果如图 1 所示。当我们上下移动手机的时候,图 2 的界面上的数值会发生变化。界面显示的数值就为被测设备的上下振动次数。

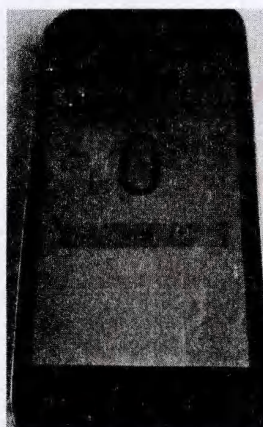


图 1 开始检测

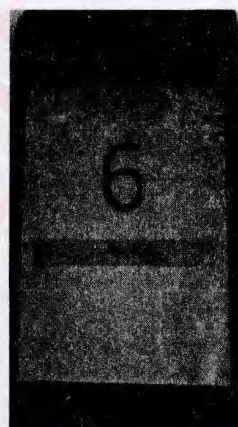


图 2 测试结果

本文是基于 Android 系统开发的手机振动仪的测试工具,可以运行在 Android 通信终端上,测试表明:本振动仪测试准确、方便。

参考文献

- [1]童承凤,胡庆.基于 Android 平台的双网双待的研究与设计[J].计算机应用与软件,2012,29(2):250-253
- [2]徐文权.基于 Symbian OS 系统的垃圾短信过滤器设计与实现[J].安庆师范学院学报,2012(6)
- [3]姚昱旻,刘卫国.Android 与 J2ME 平台间即时通信的研究与实现[J].计算机系统应用,2008,17(12):118-120
- [4]叶达峰.Eclipse 编程技术与实例[M].北京:人民邮电出版社,2006:67-68
- [5]刘向辉.基于 Java 的移动终端中间件的研究与设计[D].上海:华东师范大学软件工程学院,2005
- [6]公磊,周聪.基于 Android 的移动终端应用程序开发与研究[J].计算机与现代化,2008(8):86-89

[收稿日期:2013.7.16]

横向像素要根据字符类型确定。

参考文献

- [1]雷振山:LabVIEW 7 Express 实用技术教程[M].北京:中国铁道出版社
- [2]TPC_Getting_Started. National Instruments Corporation. 2011
- [3]黄永红,王恒海,陈照章,等.基于 LabVIEW 的温度监控与介电谱测试系统研制[J].仪器仪表学报,2008,29(7):1544-1547
- [4]赵奇峰,闵涛,等.基于 LabVIEW 串口数据采集系统设计[J].计算机技术与发展,2011,21(11):224-226

[收稿日期:2013.6.20]

嵌入式资源免费下载

总线协议:

1. [基于 PCIe 驱动程序的数据传输卡 DMA 传输](#)
2. [基于 PCIe 总线协议的设备驱动开发](#)
3. [CANopen 协议介绍](#)
4. [基于 PXI 总线 RS422 数据通信卡 WDM 驱动程序设计](#)
5. [FPGA 实现 PCIe 总线 DMA 设计](#)
6. [PCI Express 协议实现与验证](#)
7. [VPX 总线技术及其实现](#)
8. [基于 Xilinx FPGA 的 PCIE 接口实现](#)
9. [基于 PCI 总线的 GPS 授时卡设计](#)
10. [基于 CPCI 标准的 6U 信号处理平台的设计](#)
11. [USB30 电路保护](#)
12. [USB30 协议分析与框架设计](#)
13. [USB 30 中的 CRC 校验原理及实现](#)
14. [基于 CPLD 的 UART 设计](#)
15. [IPMI 在 VPX 系统中的应用与设计](#)
16. [基于 CPCI 总线的 PMC 载板设计](#)
17. [基于 VPX 总线的工件台运动控制系统研究与开发](#)
18. [PCI Express 流控机制的研究与实现](#)
19. [UART16C554 的设计](#)
20. [基于 VPX 的高性能计算机设计](#)
21. [基于 CAN 总线技术的嵌入式网关设计](#)
22. [Visual C 串行通讯控件使用方法与技巧的研究](#)
23. [IEEE1588 精密时钟同步关键技术研究](#)
24. [GPS 信号发生器射频模块的一种实现方案](#)
25. [基于 CPCI 接口的视频采集卡的设计](#)
26. [基于 VPX 的 3U 信号处理平台的设计](#)
27. [基于 PCI Express 总线 1394b 网络传输系统 WDM 驱动设计](#)
28. [AT89C52 单片机与 ARINC429 航空总线接口设计](#)
29. [基于 CPCI 总线多 DSP 系统的高速主机接口设计](#)
30. [总线协议中的 CRC 及其在 SATA 通信技术中的应用](#)
31. [基于 FPGA 的 SATA 硬盘加解密控制器设计](#)
32. [Modbus 协议在串口通讯中的研究及应用](#)
33. [高可用的磁盘阵列 Cache 的设计和实现](#)
34. [RAID 阵列中高速 Cache 管理的优化](#)

35. [一种新的基于 RAID 的 CACHE 技术研究与实现](#)
36. [基于 PCIE-104 总线的高速数据接口设计](#)
37. [基于 VPX 标准的 RapidIO 交换和 Flash 存储模块设计](#)
38. [北斗卫星系统在海洋工程中的应用](#)
39. [北斗卫星系统在远洋船舶上应用的研究](#)
40. [基于 CPCI 总线的红外实时信号处理系统](#)
41. [硬件实现 RAID 与软件实现 RAID 的比较](#)
42. [基于 PCI Express 总线系统的热插拔设计](#)
43. [基于 RAID5 的磁盘阵列 Cache 的研究与实现](#)
44. [基于 PCI 总线的 MPEG2 码流播放卡驱动程序开发](#)
45. [基于磁盘阵列引擎的 RAID5 小写性能优化](#)
46. [基于 IEEE1588 的时钟同步技术研究](#)
47. [基于 Davinci 平台的 SD 卡读写优化](#)
48. [基于 PCI 总线的图像处理及传输系统的设计](#)
49. [串口和以太网通信技术在油液在线监测系统中的应用](#)
50. [USB30 数据传输协议分析及实现](#)
51. [IEEE 1588 协议在工业以太网中的实现](#)
52. [基于 USB30 的设备自定义请求实现方法](#)
53. [IEEE1588 协议在网络测控系统中的应用](#)
54. [USB30 物理层中弹性缓冲的设计与实现](#)
55. [USB30 的高速信息传输瓶颈研究](#)
56. [基于 IPv6 的 UDP 通信的实现](#)
57. [一种基于 IPv6 的流媒体传送方案研究与实现](#)
58. [基于 IPv4-IPv6 双栈的 MODBUS-TCP 协议实现](#)
59. [RS485CAN 网关设计与实现](#)
60. [MVB 周期信息的实时调度](#)
61. [RS485 和 PROFINET 网关设计](#)
62. [基于 IPv6 的 Socket 通信的实现](#)

VxWorks:

1. [基于 VxWorks 的多任务程序设计](#)
2. [基于 VxWorks 的数据采集存储装置设计](#)
3. [Flash 文件系统分析及其在 VxWorks 中的实现](#)
4. [VxWorks 多任务编程中的异常研究](#)
5. [VxWorks 应用技巧两例](#)
6. [一种基于 VxWorks 的飞行仿真实时管理系统](#)
7. [在 VxWorks 系统中使用 TrueType 字库](#)
8. [基于 FreeType 的 VxWorks 中文显示方案](#)

9. [基于 Tilcon 的 VxWorks 简单动画开发](#)
10. [基于 Tilcon 的某武器显控系统界面设计](#)
11. [基于 Tilcon 的综合导航信息处理装置界面设计](#)
12. [VxWorks 的内存配置和管理](#)
13. [基于 VxWorks 系统的 PCI 配置与应用](#)
14. [基于 MPC8270 的 VxWorks BSP 的移植](#)
15. [Bootrom 功能改进经验谈](#)
16. [基于 VxWorks 嵌入式系统的中文平台研究与实现](#)
17. [VxBus 的 A429 接口驱动](#)
18. [基于 VxBus 和 MPC8569E 千兆网驱动开发和实现](#)
19. [一种基于 vxBus 的 PPC 与 FPGA 高速互联的驱动设计方法](#)
20. [基于 VxBus 的设备驱动开发](#)
21. [基于 VxBus 的驱动程序架构分析](#)
22. [基于 VxBus 的高速数据采集卡驱动程序开发](#)
23. [Vxworks 下的冗余 CAN 通讯模块设计](#)
24. [WindML 工业平台下开发 S1d13506 驱动及显示功能的实现](#)
25. [WindML 中 Mesa 的应用](#)
26. [VxWorks 下图形用户界面开发中双缓冲技术应用](#)
27. [VxWorks 上的一种 GUI 系统的设计与实现](#)
28. [VxWorks 环境下 socket 的实现](#)
29. [VxWorks 的 WindML 图形界面程序的框架分析](#)
30. [VxWorks 实时操作系统及其在 PC104 下以太网编程的应用](#)
31. [实时操作系统任务调度策略的研究与设计](#)
32. [军事指挥系统中 VxWorks 下汉字显示技术](#)
33. [基于 VxWorks 实时控制系统中文交互界面开发平台](#)
34. [基于 VxWorks 操作系统的 WindML 图形操控界面实现方法](#)
35. [基于 GPU FPGA 芯片原型的 VxWorks 下驱动软件开发](#)
36. [VxWorks 下的多串口卡设计](#)
37. [VxWorks 内存管理机制的研究](#)
38. [T9 输入法在 Tilcon 下的实现](#)
39. [基于 VxWorks 的 WindML 图形界面开发方法](#)
40. [基于 Tilcon 的 IO 控制板可视化测试软件的设计和实现](#)
41. [基于 VxWorks 的通信服务器实时多任务软件设计](#)

Linux:

1. [Linux 程序设计第三版及源代码](#)
2. [NAND FLASH 文件系统的设计与实现](#)
3. [多通道串行通信设备的 Linux 驱动程序实现](#)

4. [Zsh 开发指南-数组](#)
5. [常用 GDB 命令中文速览](#)
6. [嵌入式 C 进阶之道](#)
7. [Linux 串口编程实例](#)
8. [基于 Yocto Project 的嵌入式应用设计](#)
9. [Android 应用的反编译](#)
10. [基于 Android 行为的加密应用系统研究](#)
11. [嵌入式 Linux 系统移植步步通](#)
12. [嵌入式 C++ 语言精华文章集锦](#)
13. [基于 Linux 的高性能服务器端的设计与研究](#)
14. [S3C6410 移植 Android 内核](#)
15. [Android 开发指南中文版](#)
16. [图解 Linux 操作系统架构设计与实现原理（第二版）](#)
17. [如何在 Ubuntu 和 Linux Mint 下轻松升级 Linux 内核](#)
18. [Android 简单 mp3 播放器源码](#)
19. [嵌入式 Linux 系统实时性的研究](#)
20. [Android 嵌入式系统架构及内核浅析](#)
21. [基于嵌入式 Linux 操作系统内核实时性的改进方法研究](#)
22. [Linux TCP IP 协议详解](#)
23. [Linux 桌面环境下内存去重技术的研究与实现](#)
24. [掌握 Android 7.0 新增特性 Quick Settings](#)
25. [Android 应用逆向分析方法研究](#)
26. [Android 操作系统的课程教学](#)
27. [Android 智能手机操作系统的研究](#)
28. [Android 英文朗读功能的实现](#)
29. [基于 Yocto 订制嵌入式 Linux 发行版](#)
30. [基于嵌入式 Linux 的网络设备驱动设计与实现](#)
31. [如何高效学习嵌入式](#)
32. [基于 Android 平台的 GPS 定位系统的设计与实现](#)
33. [Linux ARM 下的 USB 驱动开发](#)
34. [Linux 下基于 I2C 协议的 RTC 驱动开发](#)
35. [嵌入式下 Linux 系统设备驱动程序的开发](#)
36. [基于嵌入式 Linux 的 SD 卡驱动程序的设计与实现](#)
37. [Linux 系统中进程调度策略](#)
38. [嵌入式 Linux 实时性方法](#)
39. [基于实时 Linux 计算机联锁系统实时性分析与改进](#)
40. [基于嵌入式 Linux 下的 USB30 驱动程序开发方法研究](#)
41. [Android 手机应用开发之音乐资源播放器](#)
42. [Linux 下以太网的 IPv6 隧道技术的实现](#)
43. [Research and design of mobile learning platform based on Android](#)
44. [基于 linux 和 Qt 的串口通信调试器调的设计及应用](#)
45. [在 Linux 平台上基于 QT 的动态图像采集系统的设计](#)

46. [基于 Android 平台的医护查房系统的研究与设计](#)
47. [基于 Android 平台的软件自动化监控工具的设计开发](#)
48. [基于 Android 的视频软硬解码及渲染的对比研究与实现](#)
49. [基于 Android 移动设备的加速度传感器技术研究](#)

Windows CE:

1. [Windows CE.NET 下 YAFFS 文件系统 NAND Flash 驱动程序设计](#)
2. [Windows CE 的 CAN 总线驱动程序设计](#)
3. [基于 Windows CE.NET 的 ADC 驱动程序实现与应用的研究](#)
4. [基于 Windows CE.NET 平台的串行通信实现](#)
5. [基于 Windows CE.NET 下的 GPRS 模块的研究与开发](#)
6. [win2k 下 NTFS 分区用 ntldr 加载进 dos 源代码](#)
7. [Windows 下的 USB 设备驱动程序开发](#)
8. [WinCE 的大容量程控数据传输解决方案设计](#)
9. [WinCE6.0 安装开发详解](#)
10. [DOS 下仿 Windows 的自带计算器程序 C 源码](#)
11. [G726 局域网语音通话程序和源代码](#)
12. [WinCE 主板加载第三方驱动程序的方法](#)
13. [WinCE 下的注册表编辑程序和源代码](#)
14. [WinCE 串口通信源代码](#)
15. [WINCE 的 SD 卡程序\[可实现读写的源码\]](#)
16. [基于 WinCE 的 BootLoader 研究](#)
17. [Windows CE 环境下无线网卡的自动安装](#)
18. [基于 Windows CE 的可视电话的研究与实现](#)
19. [基于 WinCE 的嵌入式图像采集系统设计](#)
20. [基于 ARM 与 WinCE 的掌纹鉴别系统](#)
21. [DCOM 协议在网络冗余环境下的应用](#)
22. [Windows XP Embedded 在变电站通信管理机中的应用](#)
23. [XPE 在多功能显控台上的开发与应用](#)
24. [基于 Windows XP Embedded 的 LKJ2000 仿真系统设计与实现](#)
25. [虚拟仪器的 Windows XP Embedded 操作系统开发](#)
26. [基于 EVC 的嵌入式导航电子地图设计](#)

PowerPC:

1. [Freescale MPC8536 开发板原理图](#)
2. [基于 MPC8548E 的固件设计](#)
3. [基于 MPC8548E 的嵌入式数据处理系统设计](#)
4. [基于 PowerPC 嵌入式网络通信平台的实现](#)
5. [PowerPC 在车辆显控系统中的应用](#)
6. [基于 PowerPC 的单板计算机的设计](#)
7. [用 PowerPC860 实现 FPGA 配置](#)
8. [基于 MPC8247 嵌入式电力交换系统的设计与实现](#)
9. [基于设备树的 MPC8247 嵌入式 Linux 系统开发](#)
10. [基于 MPC8313E 嵌入式系统 UBoot 的移植](#)
11. [基于 PowerPC 处理器 SMP 系统的 UBoot 移植](#)
12. [基于 PowerPC 双核处理器嵌入式系统 UBoot 移植](#)
13. [基于 PowerPC 的雷达通用处理机设计](#)
14. [PowerPC 平台引导加载程序的移植](#)
15. [基于 PowerPC 嵌入式内核的多串口通信扩展设计](#)
16. [基于 PowerPC 的多网口系统抗干扰设计](#)
17. [基于 MPC860T 与 VxWorks 的图形界面设计](#)
18. [基于 MPC8260 处理器的 PPMC 系统](#)
19. [基于 PowerPC 的控制器研究与设计](#)
20. [基于 PowerPC 的模拟量输入接口扩展](#)
21. [基于 PowerPC 的车载通信系统设计](#)
22. [基于 PowerPC 的嵌入式系统中通用 I/O 口的扩展方法](#)
23. [基于 PowerPC440GP 型微控制器的嵌入式系统设计与研究](#)
24. [基于双 PowerPC 7447A 处理器的嵌入式系统硬件设计](#)

ARM:

1. [基于 DiskOnChip 2000 的驱动程序设计及应用](#)
2. [基于 ARM 体系的 PC-104 总线设计](#)
3. [基于 ARM 的嵌入式系统中断处理机制研究](#)
4. [设计 ARM 的中断处理](#)
5. [基于 ARM 的数据采集系统并行总线的驱动设计](#)
6. [S3C2410 下的 TFT LCD 驱动源码](#)
7. [STM32 SD 卡移植 FATFS 文件系统源码](#)
8. [STM32 ADC 多通道源码](#)
9. [ARM Linux 在 EP7312 上的移植](#)
10. [ARM 经典 300 问](#)
11. [基于 S5PV210 的频谱监测设备嵌入式系统设计与实现](#)

12. [Uboot 中 start.S 源码的指令级的详尽解析](#)
13. [基于 ARM9 的嵌入式 Zigbee 网关设计与实现](#)
14. [基于 S3C6410 处理器的嵌入式 Linux 系统移植](#)
15. [CortexA8 平台的 \$\mu\$ C-OS II 及 LwIP 协议栈的移植与实现](#)
16. [基于 ARM 的嵌入式 Linux 无线网卡设备驱动设计](#)
17. [ARM S3C2440 Linux ADC 驱动](#)
18. [ARM S3C2440 Linux 触摸屏驱动](#)
19. [Linux 和 Cortex-A8 的视频处理及数字微波传输系统设计](#)
20. [Nand Flash 启动模式下的 Uboot 移植](#)
21. [基于 ARM 处理器的 UART 设计](#)
22. [ARM CortexM3 处理器故障的分析与处理](#)
23. [ARM 微处理器启动和调试浅析](#)
24. [基于 ARM 系统下映像文件的执行与中断运行机制的实现](#)
25. [中断调用方式的 ARM 二次开发接口设计](#)
26. [ARM11 嵌入式系统 Linux 下 LCD 的驱动设计](#)
27. [Uboot 在 S3C2440 上的移植](#)
28. [基于 ARM11 的嵌入式无线视频终端的设计](#)
29. [基于 S3C6410 的 Uboot 分析与移植](#)
30. [基于 ARM 嵌入式系统的高保真无损音乐播放器设计](#)
31. [UBoot 在 Mini6410 上的移植](#)
32. [基于 ARM11 的嵌入式 Linux NAND FLASH 模拟 U 盘挂载分析与实现](#)
33. [基于 ARM11 的电源完整性分析](#)

Hardware:

1. [DSP 电源的典型设计](#)
2. [高频脉冲电源设计](#)
3. [电源的综合保护设计](#)
4. [任意波形电源的设计](#)
5. [高速 PCB 信号完整性分析及应用](#)
6. [DM642 高速图像采集系统的电磁干扰设计](#)
7. [使用 COMExpress Nano 工控板实现 IP 调度设备](#)
8. [基于 COM Express 架构的数据记录仪的设计与实现](#)
9. [基于 COM Express 的信号系统逻辑运算单元设计](#)
10. [基于 COM Express 的回波预处理模块设计](#)
11. [基于 X86 平台的简单多任务内核的分析与实现](#)
12. [基于 UEFI Shell 的 PreOS Application 的开发与研究](#)
13. [基于 UEFI 固件的恶意代码防范技术研究](#)
14. [MIPS 架构计算机平台的支持固件研究](#)

15. [基于 UEFI 固件的攻击验证技术研究](#)
16. [基于 UEFI 的 Application 和 Driver 的分析与开发](#)
17. [基于 UEFI 的可信 BIOS 研究与实现](#)
18. [基于 UEFI 的国产计算机平台 BIOS 研究](#)
19. [基于 UEFI 的安全模块设计分析](#)
20. [基于 FPGA Nios II 的等精度频率计设计](#)
21. [基于 FPGA 的 SOPC 设计](#)
22. [基于 SOPC 基本信号产生器的设计与实现](#)
23. [基于龙芯平台的 PMON 研究与开发](#)
24. [基于 X86 平台的嵌入式 BIOS 可配置设计](#)
25. [基于龙芯 2F 架构的 PMON 分析与优化](#)
26. [CPU 与 GPU 之间接口电路的设计与实现](#)
27. [基于龙芯 1A 平台的 PMON 源码编译和启动分析](#)
28. [基于 PC104 工控机的嵌入式直流监控装置的设计](#)
29. [GPGPU 技术研究与发展](#)
30. [GPU 实现的高速 FIR 数字滤波算法](#)
31. [一种基于 CPUGPU 异构计算的混合编程模型](#)
32. [面向 OpenCL 模型的 GPU 性能优化](#)

Programming:

1. [计算机软件基础数据结构 - 算法](#)
2. [高级数据结构对算法的优化](#)
3. [零基础学算法](#)
4. [Linux 环境下基于 TCP 的 Socket 编程浅析](#)
5. [Linux 环境下基于 UDP 的 socket 编程浅析](#)
6. [基于 Socket 的网络编程技术及其实现](#)
7. [数据结构考题 - 第 1 章 绪论](#)
8. [数据结构考题 - 第 2 章 线性表](#)
9. [数据结构考题 - 第 2 章 线性表 - 答案](#)