

基于 Android 平台的 GPS 定位系统的设计与实现

郭蕊

北京工业职业技术学院 信息工程系 北京 100042)

摘要 Android 是目前发展最快、最具市场前景的智能操作系统。从 GPS 应用的系统需求着手,建立基于 Android 平台的 GPS 系统,对各主要功能模块进行了详细的研究和分析。结果表明,系统设计合理,便于在不同平台间移植和升级,具有广阔的应用前景。

关键词 Android 全球定位系统 网络协议

中图分类号 P228.4

文献标识码 A

文章编号 1671-6558 2012 02-54-05

Design and Implementation of GPS Positioning System Base on Android Platform

Guo Rui

Department of Informationg Engineering , Beijing Polytechnic College ,Beijing 100042 , China)

Abstract Android is the fastest growing and mostmarket prospects intelligent operating system. Begin on system requirements of GPS application ,the GPS system base on android is designed. Main modules of this system are analyzed in detail. The result shows that the is proved to be designed fairy ,and easy to port and upgrade in different platforms . It will be widely used in the future.

Key words android global positioning system netwok protocol

0 引言

随着全球定位系统 (GPS Global Positioning System) 的迅速发展,越来越多的研究机构致力于 GPS 的应用研究,将其广泛用于各个方面。当前的 GPS 定位系统主要采用单片机, windows mobile (CE) 和 Linux 的方式。但是,单片机功能单一,几乎无扩展性,无法支撑第三方软件,而 windows mobile (CE) 操作性、交互性不高且系统成本较高, Linux 则开发难度较大。Android 操作系统是目前最为火热的手持

设备操作系统,它不仅仅是一个手机系统,更是一整套包含硬件的解决方案,选择该系统,也就意味着选择了丰富的硬件产品及丰富的研发资源^[1]。鉴于 Android 的优势以及 GPS 的广泛应用,众多研究者把目光投向了 Android 和 GPS 的结合,力求使 GPS 以其更简单地方式应用于人们的日常生活,因此如何在 Android 平台中开发 GPS 定位系统成为了目前的一个研究热点^[2]。

本文从 GPS 定位系统的实用化角度出发,在

Android 平台下以一种简单快捷的方式获取位置信息。在系统中,采用自上而下的模块设计,通过 GPS 模块获取 GPS 信号,并通过数据采集处理模块对接收到的 GPS 信号进行封装处理,实现了数据传输的适应性。通信模块能够准确地将数据发送传输,并在后台服务器显示,为今后的监控应用提供了扩展功能。

1 GPS 原理

GPS 定位的基本原理是根据高速运动的卫星瞬间位置作为已知的起算数据,采用空间距离后方交会的方法,确定待测点的位置。

GPS 系统包含 3 大部分:空间星座部分、地面监控部分和用户设备部分。空间星座部分由 24 颗卫星组成,均匀分布在 6 个轨道面上,每个轨道面有 4 颗卫星,同时位于地平线以上的卫星数目随时间和地点而异,至少为 4 颗,最多为 11 颗。GPS 卫星的空间配置,保证了任何时刻,任何地方均可以同时观测到至少 4 颗卫星。每颗卫星装有 2 台铷钟和 2 台铯钟(1 台工作,3 台备用)为 GPS 卫星提供高精度的时间标准^[3]。

地面监控部分包括 1 个主控站、3 个注入站和 5 个监测站,设有 GPS 用户接收机、原子钟、进行数据初步处理的计算机和收集当地气象数据的传感器,而监测站获取卫星观测数据并将这些数据传送至主控站,并由主控站对地面实行全面控制,推算和编制

卫星星历、钟差、导航电文和其它控制指令等,依靠注入站注入到相应卫星的存储系统,并检测注入信息的正确性^[4]。

用户设备部分是直接面向用户的部分,主要是接收 GPS 卫星发射的信号,以获得必要的定位信息和观测量,并对数据处理、解算以完成定位工作^[5]。

GPS 具有全天候、全球覆盖、定位精度高、快速高效等特点,对人类活动的影响极大,应用价值极高。它可以从根本上解决在地球上的导航定位问题,以其简便的观测、精密的全球定位、优异的实时性、丰富的功能、良好的抗干扰性能、极强的保密性等特点,满足了不同用户的需求,使其获得了广泛的应用^[6]。

2 Android 平台下 GPS 定位系统的设计方案

2.1 整体设计方案

该系统应适用于所有采用 Android 系统的移动终端,系统可靠并且易于改进及升级,具备一定的可扩展性。在设计上采用客户端+服务器的模块化方法,利用 JAVA 编程在 Android 开发平台中运行程序,创建跟踪读取 GPS 信息获取用户的位置信息,将其存入后台服务器,并显示在 map 中。重点解决 GPS 信号的数据采集和信息处理,以及如何和通信模块建立连接发送位置信息。系统结构如图 1 所示,主要包含前端系统和后端服务系统,监控系统可做扩展使用,本文不做详细说明。

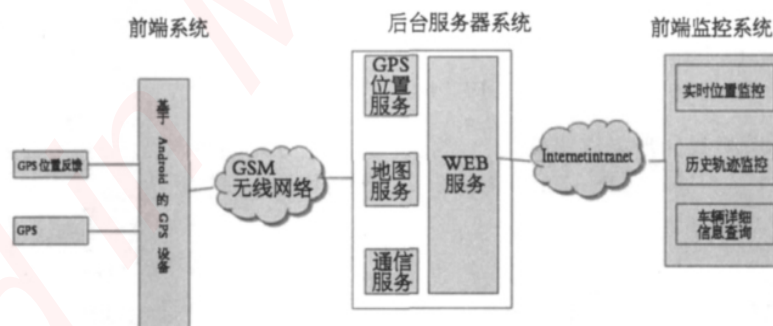


图 1 系统结构图

前端系统主要是客户端,包含 GPS 模块,数据采集处理模块以及通信模块,如图 2 所示。其中,数据采集模块集中采用 Android 的 API 模块,获取 GPS

模块中的 GPS 信号,通过数据处理模块对获取的 GPS 信号进行数据封装使其适应于网络传输协议;通信模块最终将已处理完成的数据传输至服务器。

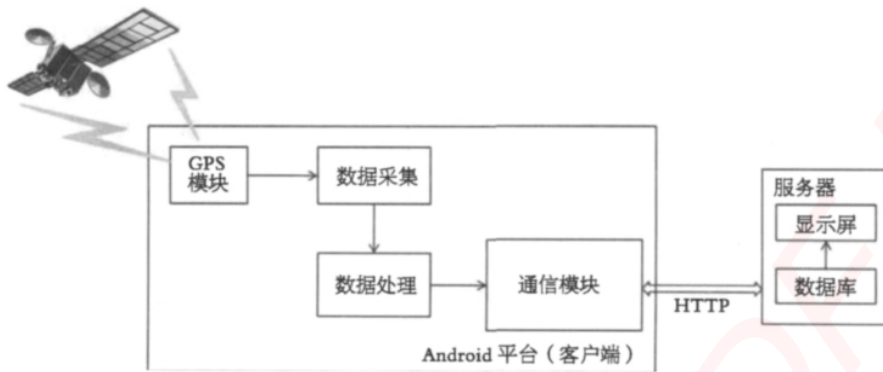


图2 前端系统组成

后端服务器系统将数据存储记录,以供GPS位置服务、通信服务、地图服务、WEB服务等。

2.2 软件实现

整体软件实现使用 Android 2.3.3 操作系统。由于希望更多的利用 Google 提供的服务,开发环境 SDK 选择的是与 Android 2.3.3 对应的 Google APIs。

代码编写环境选择了 JAVA 开发中广泛使用的开源 IDE Eclipse。通过界面调用 GPS 设置类的方法,将 GPS 监听器注入。当 GPS 坐标满足设定情况下数据反馈到界面进行显示,同时数据监听调用通信类的方法将坐标值通过无线网络发送,如图3所示。

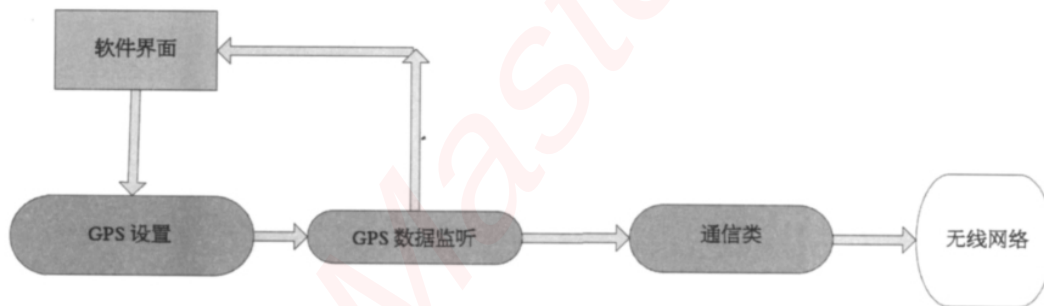


图3 系统编程实现

编程中将GPS模块和数据采集模块相结合,利用Android提供的API从平台的GPS模块,获取设备当前位置。主要包括3个功能函数和一个监听器:

- 1(设置GPS参数 openGPSSettings () ;
- 2(获取最新的GPS坐标位置 getLasGpsModel () ;
- 3(updateWithNewLocation Ilocation location) ;
- 4(GPS监听器 LocationListenerll = new LocationListener (。)

工作流程是首先打开系统的实用GPS设置;然后在获取最新坐标的时候,将重新构建过的GPS坐标监听器注入到软件中。当GPS坐标发生变化时候,监听器即可获取该坐标数值,在 public void onLocationChanged Ilocation arg0 方法中进行处理。

这里的处理包括两部分:一个是更新界面上的坐标数值和地理位置数值,另一个是调用通信模块的处理,将坐标值发送到服务器端。

数据处理模块将数据封装后送至通信模块传输,因此编程中将数据处理和通信模块结合实现为软件提供与服务器之间的通信服务。其中,通信协议采用超文本传输协议(HTTP HyperText Transfer Protocol)主要采用2种方法:

- 1(构建通信消息内容模型 CreateMsg Ilocation location) ;
- 2(发送消息 SendMsg (。)

当GPS定位模块获取到新的坐标时,即调用本模块的构建消息模型方法,将GPS数据构成本软件所需要的格式,然后在调用消息发送方法来发送数据到远程的服务器。

3 实验结果及效果分析

3.1 界面设计分析

本系统的界面本着简单明灵活的设计理念,如图4所示,上方文本框显示本系统的经纬度坐标值和本地地址(街道和城市地标名称等)下方的开始按钮为开始GPS定位,退出按钮为退出当前应用。

同时,Android界面中使用XML语言来描述界面的布局和构成,使用的字符串值均在xml文件中定义的,如果需要修改界面某些控件的名称,仅仅修改此文件就可以达到目的,创造了良好的修改和管理环境,使系统在应用方面更便捷。

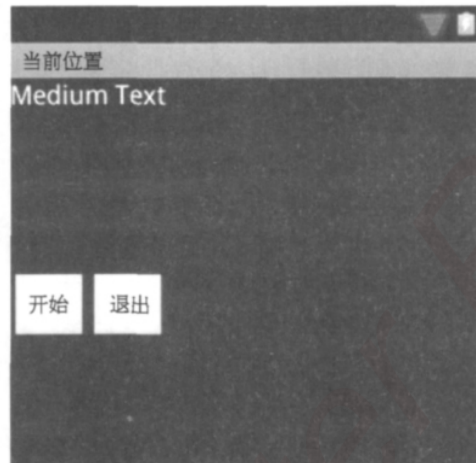


图4 系统界面

3.2 软件测试结果及分析

测试中,采用索爱Z1i Android智能手机加载本系统程序,通过HTTP协议将位置信息反馈给dell服务器,在某单位车辆实验场实际环境中使用车载

设备和定位系统联合调试,测试结果显示,可以在服务器的地图上监控到车辆实际的位置,能够较好地实现GPS定位功能。系统实际显示效果如图5所示。



图5 系统测试结果

其中,背景地图使用了卫星影像,上方为地图操作工具,左上方为地图放大级别标尺。地图上的圆形图标为当前车辆位置。车辆位置会随着车辆的运动实时变化图标后面的蓝色线条为车辆行驶的轨

迹。

在本系统的设计中,充分考虑了Android系统易扩展性和高效率等优点,集中使用了Android系统各接口API模块,使系统从开发效率和运行效率

上都有较大的提升,并能够稳定运行于所有 Android 系统,具有很强的适应性,在一定程度上满足了人们对位置的需求。同时,本系统可有效地运用于车辆监控,并为其它应用提供了相应的接口,具有可扩展性和可维护性。

4 结论

GPS 全球定位系统现在正被应用于全世界范围的工具导航、地理定位、时间精调以及武器精密制导等各领域。随着 GPS 技术的进一步发展, GPS 的应用对我们日常生活的影响越来越大,甚至会改变我们的生活方式。Android 操作系统以其强大的功能及开发资源赢得了越来越多人的亲睐。因此,本文采用 google 的 Android 平台,依据网络传输协议 HTTP 将实时获取的位置信息通过数据的处理发送至服务器,并存入数据库用以记录用户去过的位置,可用于监控及调度等。测试分析结果显示,本文的设计方案简单有效,可适用于 Android 系统的移动终端,使成果具有很大的适应性,并且为服务器向移动

终端发送命令也预留了相应的衔接端口,具有一定的扩展性。该实现方案完全能够满足 GPS 定位的需求,系统运行良好,对 GPS 应用的扩大及普及提供了一定的参考价值。

参考文献

- [1]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()5 ÷ 89.
- [2]刘正,陈强.基于 Android 的 GPS 应用中的研究与开发[J].中国西部科技 2010 1()15 ÷ 16.
- [3]邱致和.GPS 原理与应用[M].王万义,译.北京 电子工业出版社 2002.
- [4]公磊,周聪.基于 Android 的 GPS 测量系统开发[D].南昌大学 2008.
- [5]鲍雍荣.GPS 软件接收机的 C_Matlab 实现[D].上海交通大学 2007.
- [6]宋延昭.嵌入式平台开发及在 GPS 中的应用[D].北京邮电大学 2006.

责任编辑 刘莉宏)

上接第 53 页)

3.7 本安电源设计

多接口通讯管理器本安电源主要为各本安通讯接口提供电源,从而使各通讯接口实现本安特性。根据实际测量,各通讯接口在 5V 供电时,总的工作电流在 110 ~ 157mA 左右,电源功耗不大。因此多接口通讯管理器的本安电源不需要采用截流型保护电路,只需要采用限流型保护就可以了,本文采用了 DC/DC 隔离电源 + 安全栅电路来实现的,本安电源开路电压 5V,短路电流 500mA。

4 结论

本方案可实现 RS485、CANBUS、调制解调器、以太网接口之间的互联,实现本安信号和非本安信号的隔离,适用于煤矿井下具有上述接口的电气设备的互联,扩大了接口的应用范围。具有稳定、可靠、便于维护的特点。

参考文献

- [1]刘耀东.我国煤炭工业的现状、问题及其调整对策[J].中国能源 2008 30 1()5 - 13.
- [2]陈铁军.基于 RS485 的总线集散式测控系统的研究[J].微计算机信息 2008 24 1()72 ÷ 73.
- [3]谭鹏辉,白宏峰.井下辅助运输机械化探讨[J].煤炭工程 2006 1()77 ÷ 78.
- [4]许焰,汪劫,朱善安.基于 CAN 总线的数据采集和控制系统[J].机电工程 2007 2 2()59 ÷ 61.
- [5]单亚娟.TMS320F240DSP 处理器在电动机微机保护装置设计中的应用[J].电力系统及其自动化学报 2003 15 1()67 ÷ 69.
- [6]阳宪惠.现场总线技术及其应用[M].北京 清华大学出版社 2008.

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](#)

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](#)

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 定制嵌入式 Linux 发行版](#)

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](#)

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](#)

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](#)

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](#)

Programming:

1. [计算机软件基础数据结构 - 算法](#)