

基于 PowerPC 嵌入式内核的多串口通信扩展设计

居晓¹, 李志宇², 肖莉萍²

(1. 南京航空航天大学 自动化学院, 江苏 南京 210016; 2. 南京航空航天大学 无人机研究院, 江苏 南京 210016)

摘要: 本文设计了一种基于 PowerPC 嵌入式内核 MPC565 和串口模块 EMM-8M-XT 的多串口通信系统, 给出了系统总体的硬件架构和软件设计流程, 详述了串口卡跳线及寄存器的设置方法, 实现了 8 个 RS232/422/485 工作模式可任意配置的全双工通信串口。MPC565 通过 PC/104 总线对串口卡进行读写控制, 体积小, 使用方便。经大量长时间的通信测试和考机验证, 串口系统工作稳定, 实时性好, 抗干扰性强, 在不同应用环境中采用特定通信模式能有效保证信息交换的可靠性, 适用于具有多路串行数据传输需求的复杂控制场合。

关键词: PowerPC; 嵌入式系统; 多串口通信; PC/104 总线

中图分类号: TN923

文献标识码: A

文章编号: 1674-6236(2014)07-0177-04

Design of the expansion of multi-serial ports communication based on PowerPC embedded core

JU Xiao¹, LI Zhi-yu², XIAO Li-ping²

(1. College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China;
2. Research Institute of Unmanned Aerial Vehicle, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

Abstract: This article has completed the design and realization of the audio decoding system's hardware plane mainly referring to the all-purpose DSP chip of TI Company's C5000 series-TMS320VC5509, researching on MP3's encoding-decoding algorithm and the DSP realization on its hardware plane. The whole system's plane includes minimum system of DSP, FLASH storage, audio CODEC and power, etc. The system downloads the data stream of MP3 format through the communication between USB interface and PC, stores them in FLASH and then reads the MP3 data stream in Flash by DSP, finishes decoding and plays with CODEC. Moreover, the system can also be programmed for other encoding-decoding algorithms for DSP, realizes the audio encoding-decoding for various format of data stream and has a very good flexibility for updating soft, and at the same time solves the problem of lower power consumption.

Key words: MPEG; audio decoding; DSP; low power

在一些复杂的控制领域中, 控制计算机往往需要与多个终端进行信息的交互通信。但由于工程现场环境的限制和应用设备的日渐增多, 近距离集中式的通信管理已不再具有普遍适用性。

随着远程集散控制理念的出现, 串行数据通信因其结构简单、设计方便、价格低廉、占用资源少^[1]、传输距离远等优势而得到了广泛普及。而近年来串口通信正朝着带宽、速度、稳定性逐渐增加的方向发展^[2], 利用多个串口进行信号采集或数据传输为多设备分布式布局提供了可能。但一般的处理器板卡只有一至两个 RS232 串口, 难以满足控制系统对串口数量和数据传输过程中抗干扰性能的要求, 因此需要对串口进行扩展设计。

传统的串口扩展设计多以微控制器与串口芯片在板集成的方式实现, 需单独设计其控制电路和外围接口电路, 周期

较长, 灵活性低, 且稳定性和抗干扰性仍有待验证。MPC565 与 EMM 串口卡技术相对更加成熟, 各性能指标已经过长期的实践验证, 同样具有的 PC/104 总线结构使其能够以简单的插拔方式实现串口功能的扩展, 使用灵活、方便。因此本文结合以 PowerPC 嵌入式处理器 MPC565 为核心的飞控计算机开发平台, 介绍一种多串口通信在实际工程应用中的扩展方法。利用 MPC565 与 EMM-8M-XT 组成串口扩展系统, 通过简单的搭积木方式实现 8 个通信方式可配置的串口扩展。

1 系统硬件结构设计

本文涉及的内容已成功应用于某型无人机飞控系统的预研开发。众所周知, 无人机飞行控制系统是个典型的多输入、多输出系统, 其多外设平台大多采用分布式布局^[3], 串行通信逐渐成为了各传感器及任务设备与飞控计算机内核之间的主要通信方式, 高质量的数据传输尤其重要。

收稿日期: 2013-08-30

稿件编号: 201308200

作者简介: 居晓 (1983—), 男, 河北昌黎人, 硕士研究生。研究方向: 先进飞行控制技术、无人机嵌入式控制系统。

本文中飞控计算机内核采用具有 PC/104 结构的 MPC565 板卡。PowerPC 系列的 MPC565 处理器具有较高的硬件稳定性和抗干扰性,工作频率可达 50 MHz 以上,在无操作系统的情况下也能可靠、灵活的直接操作硬件,运算处理能力强大,可靠性高,工作温度范围 -40℃ ~+85℃,同时应用 PC/104 总线结构还可以外接串口、A/D、D/A、I/O 等功能组件,为其提供了方便的资源扩展能力,且有利于功能的模块化集成。MPC565 可通过简单的插拔方式与同样具有 PC/104 结构的 EMM-8M-XT 实现串口功能的扩展。EMM-8M-XT 串口板提供 8 个通信模式可配置的串行接口,分别连接磁罗盘、大气机、GPS、惯导等设备,完成不同的任务功能。扩展串口通信后飞控系统结构如图 1 所示。

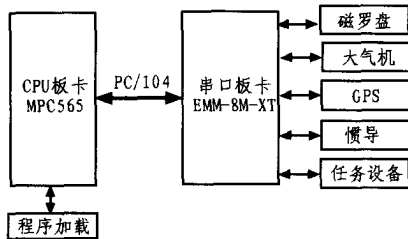


图 1 多串口通信系统结构图
Fig. 1 Schematic diagram of multi-serial ports communication system

本文应用的 MPC565 板卡设有 3 个 RS232 通信串口,主要用于程序装订和飞控参数的测试,因此需额外的串口模块来满足飞控计算机的基本功能。这里选用 DIAMOND 公司的 EMM-8M-XT 串口模块实现串口功能的扩展。

EMM-8M-XT 串口模块集成了两个高性能的 16C654 控制芯片,8 个独立的串口通道各有 64 字节的接收缓冲器和发送缓冲器,自动软硬件流程控制,最高配置波特率可达 1.5 Mb/s,有 10 个不同的中断优先级可选。每个通道的状态可以通过 CPU 的操作读取寄存器。三态输出为双向数据总线和控制总线提供 TTL 驱动能力、优先级中断系统控制、可编程的串行接口特性^[4-5]。串口基地址、中断号和通信模式(RS232/422/485 可选)亦可通过硬件跳线灵活配置,如图 2 和图 3 所示。

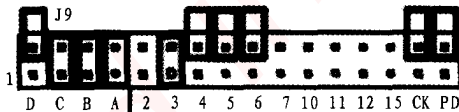


图 2 基地址和中断号跳线设置示意图
Fig. 2 Sketch map of jumper configuration for base address and interrupt number

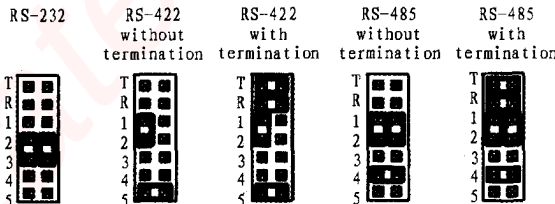


图 3 通信模式跳线设置示意图
Fig. 3 Sketch map of jumper configuration for communication mode

的基地址设为 200H, port1~port8 共用 3 号中断。基地址跳线 J9 的 A、B、C、D 定义如图 4 所示,“In”表示放置跳线,“Out”表示不放置跳线。每个串行口的基地址和中断号信息均保存在 EEPROM 中,上电后板卡从中读取这些数据,自动完成基地址和中断号的初始配置。

				Addr. Pointer Register
D	C	B	A	
In	Out	In	In	100
In	Out	In	Out	140
In	Out	Out	In	180
In	Out	Out	Out	1C0
Out	In	In	In	200

图 4 基地址跳线定义图
Fig. 4 Definition chart of jumper for base address

2 应用软件设计

应用软件采用 C 语言编程,可读性好,主要分为串口板初始化软件和串口中断服务软件。EMM-8M-XT 的 8 个串口相对独立。除共用中断级外, MPC565 可对每个串口的特性进行单独控制,因此波特率、数据格式、缓冲区深度等需通过初始化软件依次设置。8 串口初始化流程和中断处理流程如图 5 所示。

用中断方式实现芯片功能比查询方式能更有效地利用 CPU,具有更好的实时性,大批量数据传输时中断方式更能显示出其优势。

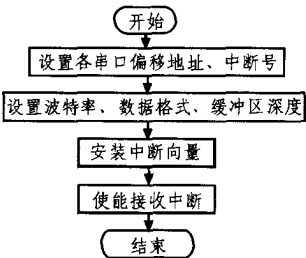


图 5 串口初始化流程图
Fig. 5 Flow chart of serial port initialization

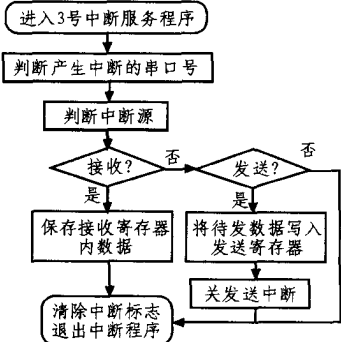


图 6 中断处理函数流程图
Fig. 6 Flow chart of interrupt service routine

本例中 8 个串口共用 3 号中断,由内部硬件机制完成板级中断状态寄存器相应位的置位与清零。在中断处理子程序中,首先要关闭 CPU 响应其他中断的能力,以免造成多层中断嵌套可能导致的程序死机^[6-7]。这样即使中断同时到来各串口也能正常工作,不会出现抢占现象,亦不会产生中断阻塞

或丢失情况。8个串口共用3号中断，具体哪一个通道产生中断由位于202H的中断状态寄存器指示（bit0~bit7分别代表1~8号串口）。中断发生后，CPU将“一视同仁”，按照3号中断处理函数内设置的中断状态查询次序，对到来的中断依次响应。数据接收或发送时，各串口采用相同的中断处理流程，下面以串口中断服务程序为例，对软件设计思路进行描述。

串口中断处理程序 EMMComInterrupt:

```
void EMMComInterrupt(int port)
```

```
{
```

```
    unsigned char status,temp;
```

```
    int base;
```

```
    base=Com[port].base; // 取该串口地址
```

```
    status = OutByte(base+ISR)&0x07; // 读取中断标志寄存器
```

```
    if((status&0x1) == 0) //bit0 为 0 时有中断产生
```

```
    {
```

```
        switch(status)
```

```
        // 判断中断源
```

```
        {
```

```
            case 4:
```

```
                // 接收到数据
```

```
                if(OutByte(base+LSR)&0x01)
```

```
                // 接收到数据并已保存到接收缓冲寄存器
```

```
                Com[port].RxBuf[Com[port].Rxhead++]=OutByte(base+RHR);
```

```
                // 从接收缓冲寄存器取出接收到的数据
```

```
                break;
```

```
            case 2: // 发送寄存器为空
```

```
                if(OutByte(base+LSR)&0x20) // 可以向发送缓冲寄存器写入数据
```

```
                {
```

```
                    if(!ComTxLen(port)) // 没有要发送的数据，关闭本串口中断
```

```
                    {
```

```
                        temp= OutByte(base+IER);
```

```
                        InByte(base+IER,temp&~BIT1);
```

```
                        break;
```

```
                    }
```

```
                    else // 将待发送数据写入发送缓冲寄存器
```

```
                    {
```

```
                        InByte(base+THR,Com[port].TxBuf[Com[port].Txtail++]);
```

```
                    }
```

```
                    }
```

```
                break;
```

```
            case 0: OutByte(base+MSR);
```

```
                break;
```

```
        }
```

```
    }
```

3 性能验证

多串口通信扩展系统实物图如图7所示。实际应用中分别对8个串口进行了局部测试和系统测试，如图8所示。在115200bps的高波特率下，长时间的通信验证未出现帧格式错误和丢帧现象。中断的收发方式亦同时保证了较高的传输速度和对其它任务的实时响应。

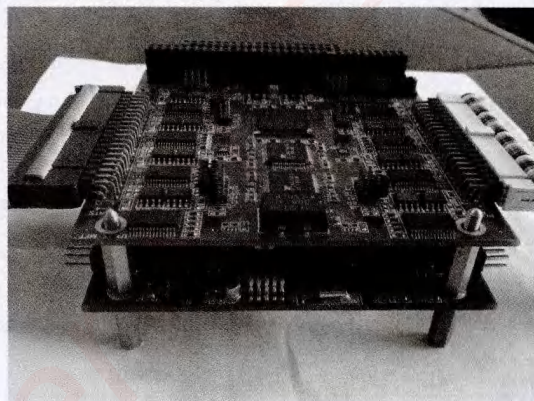


图7 多串口通信系统实物图

Fig. 7 Real picture of multi-serial ports communication system



图8 多串口通信系统实物图

Fig. 8 Real picture of multi-serial ports communication system

4 结论

文中利用PC/104总线结构的PowerPC嵌入式处理器MPC565和串口模块EMM-8M-XT成功构建了多串口通信扩展系统。各串口在带有校验机制的长时间连续串行通信测试中，RS232、RS422、RS485 3种工作模式下均收发正常，帧格式正确且误码率极低。系统测试中，8个串口工作于无人机电载复杂电磁环境和温变循环（-40℃、+85℃）等恶劣条件下，能够长时间连续稳定的进行数据传输，具有较强的抗干扰能力和良好的实时性能。

该多串口通信系统结构简单, 插拔式的 PC/104 总线结构使其具备了丰富的资源扩展能力。该设计方法已成功应用于某型无人机相似冗余度结构飞控计算机的原理样机开发中, 对工业控制领域的其它类似应用可提供参考和借鉴。

参考文献:

- [1] 兰立荣, 闫继英, 林海土, 等. 基于 ARM 处理器设备的多串口设计与实现[J]. 仪表技术, 2011(5):19-21.
LAN Li-rong, YAN Ji-ying, LIN Hai-tu, et al. Design and implementation of multi-channel UART based on ARM apparatus[J]. Instrumentation Technology, 2011(5):19-21.
- [2] Chen Z H, Shi M, Yi Q. A method for DSP asynchronous serial port expansion based on TL16C752B[C]//Business Management and Electronic Information (BMEI), 2011 International Conference on. IEEE, 2011(1):844-847.
- [3] 李志宇, 肖前贵. 应用 PC/104 模块实现多串口通讯扩展设计[J]. 工业控制计算机, 2004, 17(9): 43-44.
LI Zhi-yu, XIAO Qian-gui. Design of extending multi-serial ports communication by applying PC/104 modules[J]. Industrial Control Computer, 2004, 17(9):43-44.

- [4] 林为传, 韩晓晨, 陈觉之. 基于 MPC860 与 ST16C654 的串行通信设计[J]. 电子设计工程, 2010, 18(7):158-160.
LIN Wei-chuan, HAN Xiao-chen, CHEN Jue-zhi. Design of serial communications based on MPC860 and ST16C654[J]. Electronic Design Engineering, 2010, 18(7): 158-160.
- [5] 宋宇, 于耀, 周余, 等. 智能配电数字终端的转串口芯片驱动设计[J]. 电子测量技术, 2013, 36(2): 68-71.
SONG Yu, YU Yao, ZHOU Yu, etc. Serial chip driver design of digital terminal for intelligent power distribution[J]. Electronic Measurement Technology, 2013, 36(2):68-71.
- [6] 李伟. 基于中断方式的 TL16C754B 的串口通信驱动程序的开发[J]. 测控技术, 2012, 31(4): 93-95.
LI Wei. Driver program development of TL16C754B based on interrupt[J]. Measurement & Control Technology, 2012, 31(4):93-95.
- [7] 吴春瑜, 李璟, 王相森. 通用异步收发器 TL16C554 分析[J]. 辽宁大学学报: 自然科学版, 2007, 34(4):315-318.
WU Chun-yu, LI Jing, WANG Xiang-sen. Analysis of UART TL16C554[J]. Journal of Liaoning University: Natural Sciences Edition, 2007, 34(4):315-318.

吉时利仪器推出 2450 型数字源表 * 的固件升级服务包

先进电气测试仪器与系统的世界级领导者——吉时利仪器日前宣布, 推出新的固件升级服务包, 用于更加完善其非常成功的 2450 型数字源表 * 源测量一体单元 (SMU) 仪器——全球首款电容式触摸屏的交互式 SMU 台式仪器。

此服务包满足了客户提出的众多要求, 包含了对若干 bug 的修复, 并增加了额外的功能。一些新功能包括:

- ☐ 增强的图形操控功能, 包括触摸屏指间控制缩放 (pinch and zoom) 功能的显著改善。
- ☐ 拷贝屏幕, 并直接将其保存到 USB 闪存驱动器的能力。
- ☐ 使用 TSP 脚本时, 前面板用户界面的可编程用户提示。
- ☐ 前面板用户界面上采用新的密码保护方法。
- ☐ 一个新的虚拟前面板 Web 服务应用程序。
- ☐ 所有 2450 手册的更新, 简体中文版本和日语版本的更新。

除了 2450 的固件升级服务包外, 还发布了一个可用于 2450 的非编程启动软件 KickStart 的升级版本, 该更新版本改进了软件, 所采取的方法是:

☐ 通过 TSPLink 系统支持高达 4 台 2450 源表间的互连。客户现在可以控制多达 4 台 2450 并绘制结果。一个简单的例子是构建一台与 2 通道低功耗曲线绘图仪相当的装置。它具备了绘制器件曲线族的能力。

- ☐ 多 SMU 识别。
- ☐ 手动控制和调节图表中的 x- 和 y- 轴。

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](#)

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](#)

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](#)

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](#)
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](#)

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](#)

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](#)

Programming:

1. [计算机软件基础数据结构 - 算法](#)