

基于 CPCI 总线的 PMC 载板设计

贾亮¹, 王浩¹, 叶青林²

(1. 沈阳航空航天大学 电子信息工程学院, 辽宁 沈阳 110136;

2. 北京方天科技科技有限公司, 北京 100084)

摘要: 设计了一种基于 CPCI 总线标准的 PMC 接口载板。载板以 FPGA 为核心, 集成了 CPCI 接口模块和 DPRAM (双口 RAM) 模块, CPCI 接口模块采用 FPGA+PCI IP 核 (软核) 解决了系统集成的问题, DPRAM 模块为系统提供了数据缓存功能。环回测试和中断测试解决了在没有子卡 PMC 模块的情况下, 读写、验证 DPRAM 空间数据和测试中断响应的问题。此外, 还支持子卡 PMC 模块后出线到 CPCI 总线。调试结果表明, 该载板在嵌入式实时操作系统 VxWorks 下可以稳定运行, 正确地读写 DPRAM 空间的数据, 及时地响应中断, 满足了对载板的性能需求。

关键词: CPCI; FPGA; PMC; 载板

中图分类号: TP302.1

文献标识码: A

文章编号: 1674-7720(2013)21-0019-04

Design of PMC carrier board based on CPCI bus

Jia Liang¹, Wang Hao¹, Ye Qinglin²

(1. Electronic and Information Engineering Institute, Shenyang Aerospace University, Shenyang 110136, China;

2. Beijing Fountain Microsystems Co., Ltd, Beijing 100084, China)

Abstract: This paper designs a PMC interface carrier board based on CPCI bus standard. The carrier board uses FPGA as the core design, and integrates CPCI interface module and DPRAM (double RAM) module. CPCI interface module uses FPGA+PCI IP Core (soft) to solve the problem of system integration, the design of DPRAM module provides data cache function for the system, the design of loopback and interrupt test solve the problem of read and write and verification of DPRAM spatial data and test the interrupt response in the case that without PMC card module. In addition, it supports PMC card module to CPCI bus line. The test results show that the carrier board under VxWorks embedded real-time operating system can operate stably, read and write the correct DPRAM spatial data, interrupt timely response, meet the performance requirements of a carrier board.

Key words: CPCI; FPGA; PMC; carrier board

在工业和嵌入式领域实际工程应用中,模块化的设计思想已深入人心。针对不同使用场合的具体应用以及为方便后续的维护和升级,将其中具有通用性和可持续利用的部分单独设计成载板,载板配合不同的应用模块就可以组合成具有不同功能的专业 I/O 模块。

Compact PCI (Compact Peripheral Component Interconnect) 简称 CPCI, 中文又称紧凑型 PCI, 是国际工业计算机制造商联合会 PICMG (PCI Industrial Computer Manufacturer's Group) 于 1994 提出来的一种总线接口标准, 是以 PCI 电气规范为标准的高性能工业用总线^[1]。CPCI 规范改进自 PCI 电气规范 2.1, 应用于工业和嵌入式领域。当前最新

的 CPCI 规范是 PICMG 3.0。PICMG 3.0 主要将应用在高带宽电信传输上, 以适应未来电信的发展, PICMG 2.x 则仍是目前 CPCI 的主流, 并将在很长时间内主宰 CPCI 的应用。

PCI 夹层卡 PMC (PCI Mezzanine Cards) 是 IEEE P1386.1 的标准, 作为一个 IEEE 标准, PMC 确保了任何符合该标准的主板或者模块能够与其他按照该标准设计的主板或者模块兼容^[2]。这样就为用户提供了很大的柔性, 用户可以任意组合和搭配不同的主卡和模块。PMC 是个开放标准, 它为 Multibus II、VME 和 Compact PCI 带来了前所未有的大量 I/O 产品和高性能。

本文结合实际的工程项目,设计实现一个基于 CPCI 总线并符合 PICMG 2.0 规范的 6U (233.35 mm×160 mm) 通用型 PMC 载板 FTC-C920 并开发其在 VxWorks 5.5 下的驱动软件。

1 载板总体结构设计

本着简洁和通用的前提,此载板主要由电源模块 (Power)、PMC 接口模块 (J11~J14)、CPCI 总线接口模块 (J1、J4、J5) 和 FPGA 组成,总体结构如图 1 所示。

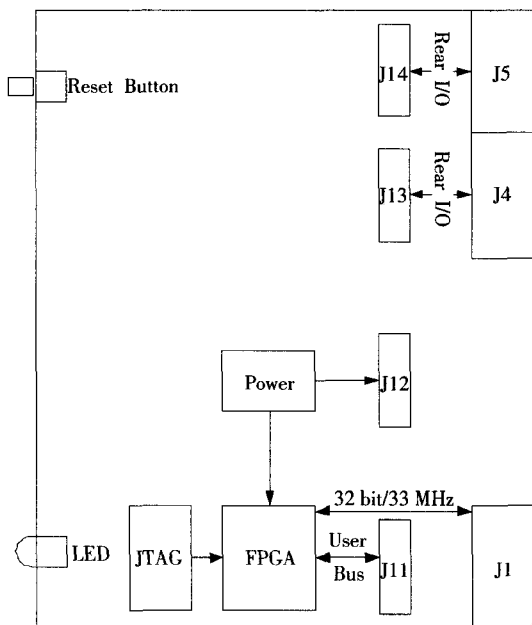


图 1 FTC-C920 总体结构示意图

电源模块主要用于给 FPGA 和用户 PMC 子卡供电 (提供 +5 V、+3.3 V、+12 V 和 -12 V 电源),载板功耗控制在 3 W 以内 (包括 3 W);CPCI 总线接口模块主要用于插入对应的 CPCI 插槽 (背板) 与 CPCI 主控板 (系统板) 进行通信;PMC 接口模块主要用于 PMC 子卡和载板的连接与通信;FPGA 芯片主要用于支持 32 bit/33 MHz PCI Slave 接口,支持 8K×16 bit 双口 RAM,支持双口 RAM R 端口环回自检 (图 2 中虚线所示),其功能框图如图 2 所示;JTAG 主要用于调试 FPGA 程序。

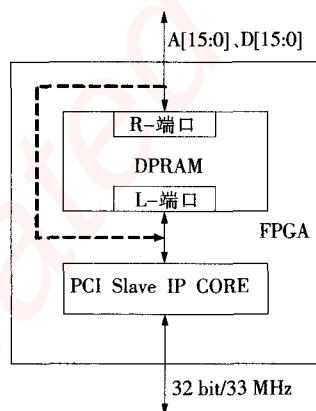


图 2 FPGA 内部功能框图

2 PMC 接口和 CPCI 总线接口模块设计

PMC 接口的特性为:前两个 (“P11”和 “P12”) 用于 32 位 PCI 信号,第 3 个 (“P13”) 是 64 位 PCI 信号需要。一个额外的总线连接器 (“P14”),可用于非指定的 I/O 信号^[2]。本设计中,PMC 接口模块采用 4 个 (J11~J14) PMC (PCI Mezzanine Cards) 8 mm 高标准连接器,支持用户子卡 PMC 后出线到 CPCI 总线接口 J4 和 J5。图 1 给出了板卡的 CPCI 连接器情况,J1 用作 32 位 PCI,J4 和 J5 用作后面板 I/O,可自定义^[1]。

3 FPGA 设计

从图 2 中可以清楚地看到,FPGA 主要被设计集成了 PCI 接口模块 (PCI Slave IP CORE) 和 DPRAM 模块。

3.1 PCI 接口模块设计

3.1.1 几种 PCI 接口设计方案及选择

常见的设计方案有:(1) 专用的 PCI 接口芯片 + CPLD;(2) ASIC+CORE;(3) 专用的 FPGA+PCI IP CORE (硬核);(4) 专用的 FPGA+PCI IP CORE (软核);(5) 带 PCI 接口的 DSP^[3]。

几种方案各有利弊,本文结合项目的实际应用需求选择了第 4 种方案,这个方案满足了系统集成需求,并可自由添加接口逻辑,降低了成本,减小了设计风险。

3.1.2 具体设计

此模块主要实现 33 MHz 工作时钟、32 bit 总线宽度的 PCI 接口功能,支持内存空间及配置空间的读写和 PCI 中断功能。

PCI 配置空间分配^[4]:Vendor ID (厂商号) 1206,Device ID (设备号) 0920,BAR0 (基地址) 映射为 1 MB 内存空间,其中 0x0000~0x3FFF 为 DPRAM 空间,0x4000~0x7FFF 为 DPRAM 环回测试空间,0x8000 以后为控制寄存器。

正常情况下 (配合用户 PMC 子卡使用时),CPCI 通过 DPRAM 左 (L) 端口对 DPRAM 空间进行读写,用户子卡则通过 DPRAM 右 (R) 端口对 DPRAM 空间进行读写;环回测试情况下 (即没有用户 PMC 子卡的情况下),CPCI 可以通过环回测试空间对 DPRAM 进行读写,即模拟用户子卡对 DPRAM 空间进行读写。

需要注意的是,0x0000~0x3FFF 为 DPRAM 16 KB 空间,对应 DPRAM 空间大小为 8K×16 bit,而 0x4000~0x7FFF 为 DPRAM 环回测试空间,不是 FPGA 另外开辟的 RAM 空间,只是软件通过这个空间地址在环回测试下读写 DPRAM 空间 (8K×16 bit)。示意图如图 3 所示。

PCI 控制寄存器的分配如表 1 所示。

3.2 DPRAM 模块设计

双口 RAM 模块是 FPGA 调用内部的 IP 核实现的,其为数据处理提供缓存功能,总线宽度为 16 bit,容量为 8 K×16 bit。在没有用户 PMC 子卡的情况下,为了验证对数据处理的可用性及准确性,通过 FPGA 逻辑设计支持双口 RAM R 端口环回自检,如图 3 (b) 中虚线所示,

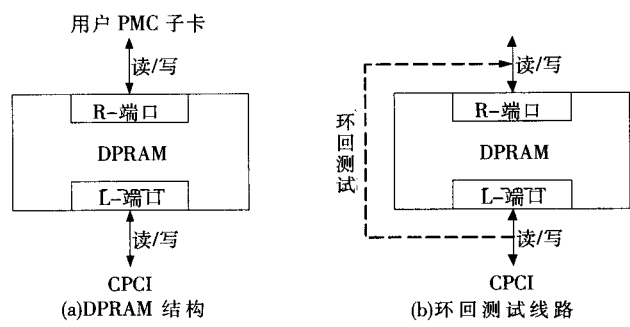


图3 读写 DPRAM

表 1 PCI 控制寄存器的分配

PCI 端地址 addr[15:0]	名称	属性	描述	默认值
0x8000	FPGA_VER_REG	R	FPGA 版本寄存器	0x0011
0x8002	RST_CTRL_REG	W/R	子卡复位控制寄存器	0x0001
0x8004	INT_EN_REG	W/R	中断使能寄存器	0x0000
0x8006	INT_CLR_REG	W/R	清中断寄存器	0x0000
0x8008	BIST_CTRL_REG	W/R	自检控制寄存器	0x0000
0x800a	INT_STA_REG	R	中断状态寄存器	0x0000
0x800c	INT_TEST_REG	W/R	中断测试寄存器	0x0000
0x800e	CPCI2PMC_INT_REG	W/R	CPCI 产生 PMC 子卡中断寄存器	0x0000
0x8010	PMC_INT_STA_REG	R	CPCI 产生 PMC 子卡中断状态寄存器	0x0000

注:R 的意思是只读,W/R 的意思是可读可写。

即模拟用户子卡对双口 RAM 进行读写数据的操作。
CPCI 和用户 PMC 子卡通信采用双口 RAM 方式,双口 RAM 用户 PMC 子卡侧总线定义如表 2 所示。

表 2 子卡双口 RAM 接口信号定义
(均为 LVTTTL 电平)

信号名称	信号方向 (载板到子卡为 OUT, 反之为 IN)	信号说明
CS_n	IN	片选信号,低电平有效
RD_n	IN	读使能信号,低电平有效
WR_n	IN	写使能信号,低电平有效
RESET_n	OUT	子卡复位信号,低电平有效
INT_n	OUT	中断信号,低电平有效
DATA[15:0]	INOUT	16 位数据总线
ADDR[15:0]	IN	16 位地址总线

4 VxWorks 下设备驱动程序设计

开发此 PMC 载板 Vxworks 5.5 下的驱动,硬件选用了 Motorola 公司的实时主控单板机 MCP-750、工控机箱 CPX2000 series 和此 PMC 载板 FTC-C920,操作系统则选择了美国风河公司的 VxWorks 嵌入式实时操作系统,使用普通的 PC,在 Tornado IDE(集成开发环境)下进行驱动的开发。以良好的可靠性和卓越的实时性著称的 VxWorks 可以满足系统对于高实时性和高可靠性的要求[5]。

驱动开发软硬件环境简图如图 4 所示。

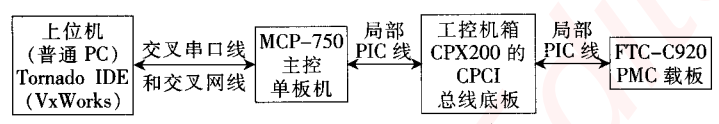


图 4 驱动开发软硬件环境简图

对驱动程序的设计主要有 3 个部分:载板自检测试模块、FTC-C920 和系统控制器互通测试模块以及 CPCI 中断测试模块。

4.1 载板自检测试模块

载板自检测试主要是载板检测自身器件是否正常工作,功能是否可以实现。

4.1.1 查找 C920 载板

对于 Vxworks 下 PCI 设备调试来说,其有专门的函数可供调用,查找 FTC-C920 载板可以通过调用函数 pciFindDevice()来实现,它包含在 pciConfigShow 库函数中,此库函数专门用来帮助显示一些 PCI 设备的信息。此函数可以根据 PCI 设备给定的供应商标识(VENDOR ID)和设备标识(DEVICE ID)找到相应的设备,并以此返回设备的总线号(busNO)、设备号(deviceNO)和功能号(funcNO)。

4.1.2 读取 FTC-C920 上的配置资源信息

根据获得的设备的总线号(busNO)、设备号(deviceNO)和功能号(funcNO),通过调用操作系统库函数 pciConfigLib 中的函数 pciConfigInLong 和 pciConfigInByte 访问 C920 的配置空间,获得设备映射的内存基地址(memBase)和 I/O 基地址(ioBase)以及中断号 irq,分辨对基地址空间是 I/O 操作还是内存操作,获得的基地址要分别与存储器屏蔽位 PCI_MEMBASE_MASK 和 I/O 屏蔽位 PCI_IOBASE_MASK 相“与”,才能得到真正板卡的内存基地址和 I/O 基地址。

4.2 FTC-C920 和系统控制器互通测试模块

互通测试主要是:

- (1)系统控制器(MCP-750)读取用户 PMC 子卡数据:用户 PMC 子卡向双口 RAM 中写入数据,系统控制器(MCP750)从双口 RAM 中读取数据,并验证数据是否正确。
- (2)用户 PMC 子卡读取系统控制器(MCP-750)数据:系统控制器(MCP-750)向双口 RAM 中写入数据;用户 PMC 子卡从双口 RAM 中读取数据,并验证数据是否正确。

由于 FTC-C920 是通用型 PMC 载板,为了在没有用户 PMC 子卡的情况下验证互通测试,在 FPGA 的设计上进行了支持 DPRAM R 端口环回自检的设计,如图 2 中虚线所示。

正常情况下(配合用户 PMC 子卡使用时),环回测试空间不可使用,此时 FPGA 内部环回自检控制寄存器的相应位为默认值 0。如进行环回自检(没有用户 PMC

子卡时), 需往环回自检控制寄存器的相应位写 1, 此时, 系统控制器可对环回测试空间进行读写。

4.3 CPCI 中断测试模块

CPCI 中断测试主要是用户 PMC 子卡通过写寄存器的方式产生 CPCI 中断; 系统控制器响应中断, 读取数据并验证数据是否正确。

在没有用户 PMC 子卡的情况下, 可以通过产生内部中断的方式进行测试, 具体的方法是在 FPGA 中设置中断使能寄存器, 中断测试寄存器, 中断状态寄存器和清中断寄存器。程序流程图如图 5 所示。

本文详细描述了某项目中 PMC 载板 FTC-C920 的设计方法及其在 VxWorks 操作系统下驱动程序的开发流程。在没有用户 PMC 子卡的情况下, 通过环回测试空间的设计, 解决了互通测试的要求, 通过中断测试寄存器的设计, 解决了中断测试的要求。调试结果表明, 此 PMC 载板在 VxWorks 5.5 下可以稳定地运行。本文的设计方法具有一定的通用

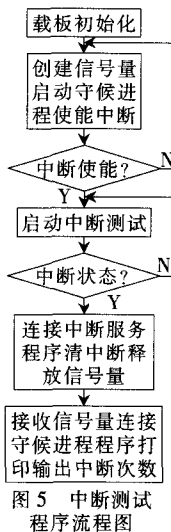


图 5 中断测试程序流程图

性, 可为相关工程技术人员提供可以参考的设计经验。

参考文献

- [1] PICMG.PICMG2.0 Rev.3.0. PICMG2.0 D3.0 Compact PCI Specification[S].
- [2] IEEE IEEE P1386.1.Draft Standard Physical and Environmental Layers for PCI Mezzanine Cards: PMC[S].
- [3] 郑毅.基于 PMC (PCI 背卡) 接口的实时数据采集卡[D].成都: 电子科技大学, 2003.
- [4] 李贵山, 陈金鹏.PCI 局部总线及其应用[M].西安: 西安电子科技大学出版社, 2003.
- [5] 唐晓平, 何峰, 梁甸农.基于 VxWorks 的 PCI 总线驱动设计[J].嵌入式操作系统应用, 2008, 11(2): 79-81.

(收稿日期: 2013-06-13)

作者简介:

贾亮, 男, 1971 年生, 硕士, 副教授, 主要研究方向: 信号与信息处理。

王浩, 男, 1988 年生, 硕士研究生, 主要研究方向: 信号与信息处理。

叶青林, 男, 1978 年生, 本科, 软件工程师, 主要研究方向: 嵌入式软件开发。

(上接第 15 页)

本文讨论了电力行业 95598 综合信息系统的研究与设计, 从电力行业 95598 综合信息系统建设的重要性和意义入手, 结合电力行业系统设计原则, 对 95598 综合信息系统的业务功能模块进行详细的划分, 同时根据该系统的功能, 提出了结合呼叫平台和客户服务业务子系统两部分的系统体系架构, 并对其中主要模块的技术方案进行了阐述。

参考文献

- [1] 孟凡博, 刘进江.基于客服系统知识库的设计与实现[J].微计算信息, 2007(2): 235-237.
- [2] ANDERSON E W, FORNELL C, DONALD R. Customer satisfaction, marketshareandprofitability[J]. JournalofMarketing,

2004;53-66.

- [3] 周兵.VoIP 呼叫中心运营商的新一代增值利刃[J].CTI 世界: 计算机与电信集成世界, 2002(9): 421-426.
- [4] 叶飞, 傅海阳.呼叫中心的技术研究和方案实现[J].电力系统通信, 2001(6): 37-42.
- [5] 吴瑞云.我国呼叫中心运营市场的现状与发展[J].电信软科学研究, 2004(7): 211-213.

(收稿日期: 2013-07-09)

作者简介:

杜一龙, 男, 1984 年生, 本科, 助理工程师, 主要研究方向: 信息管理。

刘谊, 男, 1974 年生, 博士后, 副教授, 主要研究方向: 信息管理。

(上接第 18 页)

计数器的最大值和最小值。随机函数在最小值和最大值之间任意取值, 取值概率相等。

本文对无线燃气集抄系统的燃气表终端、集抄器及 GPRS 通信模块进行电路和软件设计, 提出了主动随机退避算法。但由于时间和现有条件的限制, 系统还有很多不足之处有待改进: (1) 网络多节点中继延时性增加了系统不稳定性, 而且每一次中继都会增加网络延时, 有效地优化无线网络延时问题, 缩短中继时间需更深入研究; (2) 集抄器路由协议有待更深入研究。

参考文献

- [1] 曹磊.MSP430 单片机 C 程序设计与实践[M].北京: 北京航空航天大学出版社, 2007.

- [2] 陈荣.GPRS 无线网络数据业务优化研究[D].北京: 北京邮电大学, 2012.

- [3] 王桂军.微功率无线集抄系统的设计与应用实现[D].北京: 北京邮电大学, 2012.

- [4] 戴立渊.Ad Hoc 无线网络路由协议的设计与实现[D].成都: 电子科技大学, 2012.

- [5] 郑少仁, 王海涛, 赵志峰.Ad Hoc 网络技术[M].北京: 人民邮电出版社, 2005.

(收稿日期: 2013-08-02)

作者简介:

汪宋良, 男, 1982 年生, 硕士, 讲师, 主要研究方向: 无线低功耗与无线网络协议技术应用。

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](#)

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](#)

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](#)

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\[可实现读写的源码\]](#)

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](#)

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](#)

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](#)