

嵌入式微机 MPC555 驻留片内监控器的开发与实现

徐 遑 贾克斌

(北京工业大学电子信息与控制工程学院, 北京 100022)

摘 要 主要讨论了如何在摩托罗拉嵌入式 RISC (精简指令集计算机) 微型计算机 (PowerPC 系列 MPC555) 上进行有效的软件开发. 结合驻留片内监控器程序的开发实例, 对开发的方法、过程进行了简要阐述, 并着重对这种控制应用程序模块运行的监控器程序的设计及特点进行了详细分析.

关键词 摩托罗拉嵌入系统, PowerPC, 精简指令集计算机 (RISC), 驻留片内监控器, 嵌入式系统

分类号 TP 392

0 引言

嵌入式 RISC 微型计算机, 是微电子技术发展到现阶段的产物. 1976 年, 美国 Intel 公司的 MCS-48 单片微型机的问世使其成为单片微型机划时代的里程碑, 标志着大规模集成技术和计算机技术的伟大成就. 随着技术的进步, 如 CMOS 技术的出现和生产工艺的提高, 各公司不断推出更高性能的单片微型机. 1991 年, 由 3 家实力雄厚的计算机及半导体公司——苹果、IBM 和摩托罗拉结盟, 共同努力将 IBM 的 POWER RISC 微处理器体系结构融合在一个 RISC 芯片中, 导致了一个全新设计的微处理器——PowerPC 的出现. 本文所讨论的硬件平台——MPC555, 正是建立在 PowerPC 体系结构上, 采用 RISC 技术产生的. 和以往的单片微型机相比, MPC555 具有明显的特点和优势. 其超强的寻址能力, 32 位的数据宽度, 虚拟支持, 流水线式指令处理方式^[1,2], 都令其他同类产品相形见绌, 而与此同时, 其体积仅有 2.5 cm×2.5 cm×0.5 cm, 所有这些特性都使其特别适用于极其复杂的嵌入式控制系统, 如汽车发动机、办公设备、通信及网络设备、视频装置和家用电器等.

在当前阶段, 嵌入式微型计算机仍然向着“一片就是一台计算机”的方向进一步发展, 把所有的功能都集成在片内. 这样可以简化计算机部分硬件设计, 降低成本, 提高可靠性, 缩小电路占用空间, 这方面美国摩托罗拉公司做的最为突出.

作者设计并编制了驻留片内监控器程序, 并通过监控器实现了一个包括上位机和下位机的小系统的构建. 通过这个系统, 可实现对 MPC555 的资源控制、程序执行控制, 及上下位机的通信.

1 Motorola 的嵌入式微型计算机——MPC555 的概述

摩托罗拉 MPC555 隶属于 Motorola PowerPC RISC 微控制器家族, 是一款高档, 适用于精密控制需求的微控制器 (Micro Controller). 其增加了浮点单元的 PowerPC 强劲动力核心、26 k 的快速静态内存、448 k 片内闪存 EEPROM、队列串行多信道模块 (QSMCM)、双 CAN 总线 2.0B 控制器模块 (TouCAN TM)、双重时间处理器单元 (TPU3)、模块化 I/O 系统 (MIOS1)、32 通道模拟输入、双队列模数转换器 (QADC64)、HCMOS (CDR1) 技术^[3]、高达 40 MHz 的工作频率等特性使得 MPC555 可以完成复杂、精密、实时的控制任务, 在我国有着广泛的用途. 其模块框图见图 1^[3].

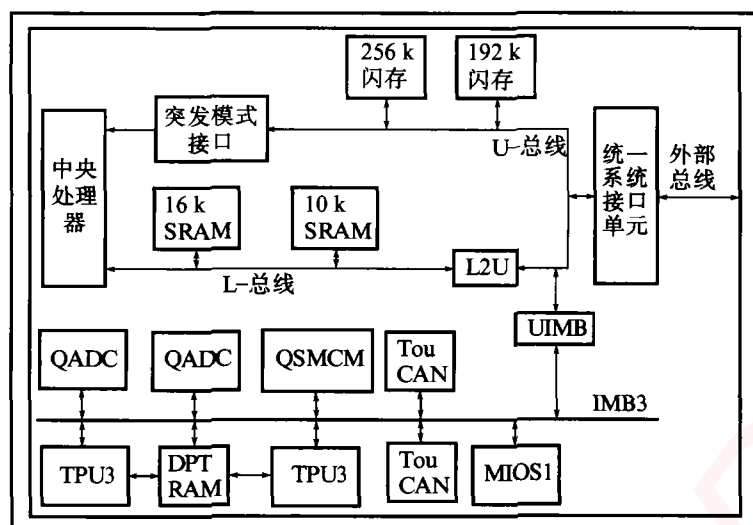


图1 MPC555块图

2 驻留片内监控器的设计和开发

2.1 什么是监控器程序

监控器程序实际上是一种控制机制，它控制着片外消息的接收，消息的片内发送与程序的运行，同时对 MPC555 的各种异常进行监控及管理。简单说，就是一个简单的系统管理程序。值得注意的是，监控器程序并不是一个应用程序，而是系统程序。

监控器在应用前景上是很广阔的。现在绝大多数系统都不是一个单功能系统，它们包含了很多的功能模块。引入一套好的机制将这些模块联系起来，以发挥其最大功效，对于开发者是一个需要解决的问题。对使用者来说，这样一套好的机制将带来性能上的益处，而监控器也正是这样的一个机制。

作者编制的监控器程序由于增加了串口通信模块，因此可以与微机通信，很方便地搭建起一套小系统，更增强了它的实用性。在工业控制、数据采集、多层分布式集散系统网络，如包括上位机 (PC) 和下位机 (嵌入式系统) 的应用场合，乃至日常家电上都有它的用武之地。

2.2 监控器的工作原理

由于监控器是系统程序，因此它驻留于内部闪存中。当机器启动后，监控器立即接管控制权，同时进行必要的自身环境设置。然后，进入消息循环，等待消息。一旦接到消息，就进入消息派送进而转入应用程序分支。也就是说，一旦监控器开始工作，它就将是系统的全权控制者，由于监控器接管了所有的异常处理 (所谓异常处理就是对由程序运行引起的系统不正常状态进行处理)，所以不会因应用程序运行异常而死机。这样除非关电，否则它是不会停止运行的，见图2 所示。

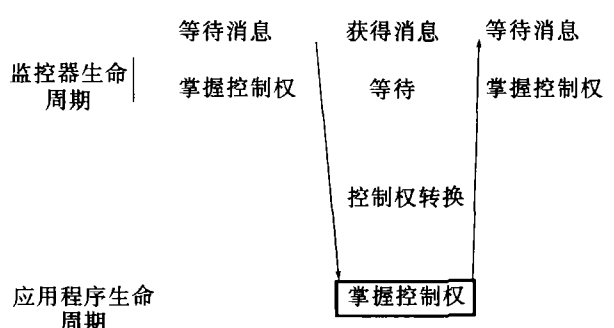


图2 生命周期图

2.3 监控器程序的消息接收及处理

本研究的消息来源是串口，因此只需不断的查询 MPC555 串口状态寄存器 (QSCI1SR) 的对应位就可以判断出是否有信息传来^[3]。这样，通过读取接收缓冲区就可以接收信息了。如果要加上对其他消息来源的支持，如键盘或其他外设，只要在查询程序中加入一个查询循环对多个消息源循环查询就可以了。

在获得消息后将消息放入一指定缓冲区,其中命令标识以供消息处理程序提取,运行参数供下面的应用程序提取。剩下的工作是等待系统的消息处理程序处理。这正是以消息为基础,以事件为驱动的模式,包括了消息循环(利用一个 WHILE 循环)、翻译、派送。这样做,可以将不同功能模块的执行控制起来,实时据外部事件驱动应用程序,比以往一个芯片只被一个应用程序占据有了很大改进。

2.4 异常处理

MPC555 的所有 14 种异常处理都有固定的内存偏移地址^[3],全在一张向量表中,以异常类型、十六进制异常代码及异常说明 3 部分形式给出。只需对不同的异常编写处理代码,然后将各自的处理代码在编译时定位到对应内存中的向量表中,监控器就可以在发生异常时执行处理代码,接管异常了。请注意,由于在应用程序发生异常时,系统控制权无法正常返还监控器,因此在异常处理代码中加入了强制返还控制权的代码(包括恢复机器状态和设置返回地址),确保从异常处理返回时可以准确的跳回监控器程序。

但要注意的是,在强制将控制权返还给监控器的过程中,对发生异常前的机器状态应予以恢复,处理完异常后该从何处继续执行指令也应该明确指出,这主要通过操作机器状态存储/恢复寄存器(SRR0/SRR1)^[3~5]来完成。SRR0 指明了处理完异常返回后该从何处继续执行指令,SRR1 存储了异常发生前的机器状态,只要在异常返回指令 rfi 执行前设置 SRR0 就可以控制异常返回后该从何处继续执行指令,一般 SRR1 不用设置。作者设计的监控器的异常处理程序均会向串口发出异常的类型,这样就提供了一条通道,使外部用户可以知道具体的问题所在。

2.5 通讯模块

MPC555 本身提供了队列多通道串行模块,每个通道提供了一组控制寄存器用于如波特率等的设置,一组状态寄存器,用于获取串口状态。由于在硬件上的很好支持,只需设置一下寄存器就可以通信了,当然,与外部通信时还应规定一套协议,以保证通信的可靠性。图 3 为帧结构。

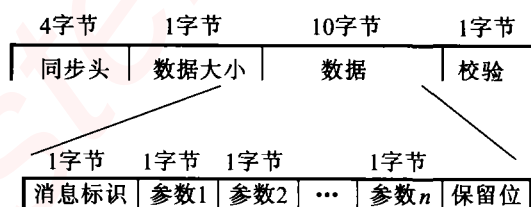


图3 帧结构

注意:4 个参数组成了 32 位地址。如参数 1 是

0x00,参数 2 是 0x3f,参数 3 是 0x98,参数 4 是 0x1c,则所代表地址为 0x003f981c。

上面的帧是一较简单的格式结构。对 PC 和 MPC555 都在通信时先判断同步标志握手,再进行校验,若二者有一个不符合则请求重发。这里假设传输过程中线路情况较好,半双工,通信时一呼叫一应答。

2.6 其他

嵌入式系统上的编程采用 Diab C^[6],这是一种在标准 C 基础上针对嵌入式硬件的特点加以扩展的 C 语言。必须编写 make 文件(描述了处理所有软件资源的原则和资源之间的依赖关系)和link 文件(用于程序代码的定位,描述了程序执行时该将各部分调入到什么地址段执行)。Diab C 提供软件包完成编译链接。

将此程序与应用功能模块一起编译并下载到芯片上就可以检验其运行效果了。调试可以用 Single Debugger 调试器。

3 结论

监控器作为系统程序很好地实现了它的功能,有效地控制了应用功能模块的执行。作者编制的驻留监控器程序已经应用在了摩托罗拉 PowerPC 系列 MPC555 单片微型计算机上。另外,值得提出的是,了解嵌入式硬件的内部特性,对有效编程有着重要的意义,可以使开发者更好地将软件与硬件配合起来,开发出好的程序。作者编制的监控器程序,如果与实时多任务操作系统核心一起编译就可以实现多任务操作。由于实时多任务操作系统可以进行任务管理、资源管理、事件处理、队列管理、存储管理、循环定时器,

这样就允许许多进程、线程的实现,从而将使整个监控器功能更加强大。

参 考 文 献

- 1 Chakravarty Dipto. PowerPC 体系结构. 北京: 电子工业出版社, 1994
- 2 沈绪榜. Risc 及后编译技术. 北京: 清华大学出版社, 1994
- 3 MOTOROLA. MPC555 user manual. 1999
- 4 MOTOROLA. RCPU RISC CENTRAL PROCESSING UNIT REFERENCE MANUAL. 1999
- 5 MOTOROLA. CDR MoneT FLASH EEPROM Spec. 1998
- 6 Diab Data, Inc. Diab C USER MANUAL. 1998

Development and Implement of the Monitor on Chip of Embedded Computer MPC555

Xu Chuan Jia Kebin

(College of Electronic Information & Control Engineering, Beijing Polytechnic University, Beijing 100022)

Abstract It is discussed how to effectively develop software on Motorola Embedded RISC microcomputer (MPC555 of Power PC system). With the development example of the monitor on chip its development method and development process are briefly elaborated, and at the same time the design and characteristics of the monitor program are analyzed detailedly.

Keywords Motorola embedded system, Power PC, RISC, monitor on chip, Embedded system

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](#)
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. [USB3.0 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 USB3.0 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. [USB3.0 物理层中弹性缓冲的设计与实现](#)
55. [USB3.0 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](#)
63. [MVB 网络重复器的设计](#)
64. [一种新型 MVB 通信板的探究](#)

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](#)
42. [基于 VXWORKS 的 RS485MVB 网关的设计与实现](#)

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](#)
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](#)
50. [基于 Android 系统振动测试仪研究](#)
51. [基于缓存竞争优化的 Linux 进程调度策略](#)

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](#)
25. [基于 PowerPC603e 通用处理模块的设计与实现](#)

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](#)
34. [基于 ARM S3C6410 的 uboot 分析与移植](#)

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](#)

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 技术研究与实现](#)
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](#)

