

基于 PowerPC440G型微控制器的嵌入式系统设计与研究

杨凡

福建实达电脑设备公司 福建福州 350002)

摘要 介绍一种新型 PowerPC440G型 32位高性能微控制器 提出一种基于 PowerPC440G的网络服务器嵌入式系统硬件设计方案并给出部分设计细节 然后讨论基于此硬件平台的嵌入式操作系统 VxWorks的开发、移植及 BSP引导程序的开发流程。

关键词 精简指令集计算机 (RISC); PowerPC440GP, VxWorks BSP, BootROM

中图分类号 : TN915.05 文献标识码 : A 文章编号 : 1006- 6977(2006) 07- 0017- 05

Design and research of embedded system based on PowerPC440GP microcontroller

YANG Fan

(Fujian Start Computer Equipment Company, Fuzhou 350002,China)

Abstract:A new style 32-bit high performance microcontroller PowerPC440GP is introduced.The typical embedded system hardware design project about network server product based on PowerPC440GP, and a part of design are given in detail.The development, porting and development flow of BSP bootstrap program of embedded operating system VxWorks based on this hardware platform are discussed.

Key words:reduced instruction set computer(RISC); PowerPC440GP; VxWorks BSP; BootROM

1 引言

PowerPC440G是 IBM公司推出的高性能 32位 RISC嵌入式处理器 片上集成了 PowerPC440核及各种丰富的外围设备接口资源 最高速度可达 500 MHz适合应用在嵌入式产品诸如交换机、路由器、网络服务器等台式设备中。本文针对该处理器的特点提出一种针对网络服务器的较为通用的应用系统设计方案 并对此方案的关键软硬件技术进行讨论。

2 PowerPC440G的特点与性能

12位处理器局部总线 (PLB)提供峰值为 4.2 GB/s的读写能力 用来访问 PCI, DDR SDRAM等高性能设备 采用 32位地址路径 能提供 64 GB内存寻址能力。

32位片上外围设备总线 (OPB)接串口、以太

网口、外部总线控制器、I/O等低性能设备 以提高PLB总线利用率。

64位寻址的 DC总线访问 PLB总线上各种主从设备的状态与控制寄存器。

拥有 64位片上双倍速率 DDR SDRAM控制器 最多可提供 4个设备片选信号。

拥有最高频率 133 MHz的 64位 PC接口。

可支持 4个外设的 32位外部总线控制器。

4个 10 MB/s/100 MB/s以太网控制器。

4个串口和 4个 I/O。

32个通用输入输出 GPIO

1个外部中断和 4个内部中断资源。

CPU工作频率可在 400 MHz 466 MHz 500 MHz频率下通过初始化配置选择 ,一般功耗小于 4 W

电源 逻辑电压为 1.8 V, DDR SDRAM接口电压为 2.5 V, I/O电压为 3.3V

3应用系统设计方案

该网络服务器平台以 PowerPC440G为核心，进行了必要的扩展 系统结构框图如图 所示。

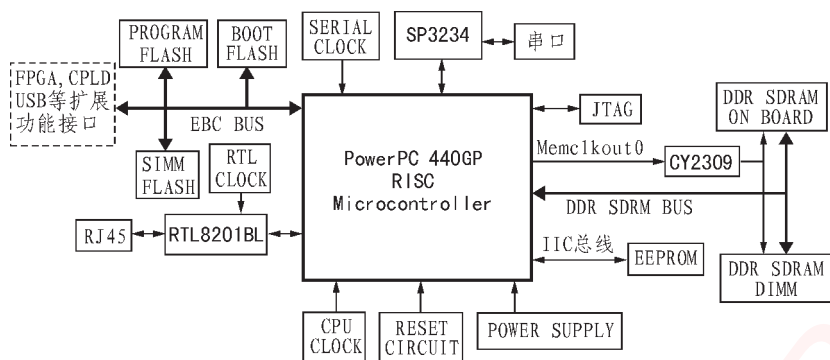


图 1 PowerPC440G网络服务器系统结构框图

系统包含 PowerPC440G处理器及其上电复位电路、电源电路、系统时钟电路、DDR内存及扩展电路、程序存储及启动调试 Flash电路，个连接在 1C总线上提供启动配置的 EEPROM用于 IC调试的 JTAG 以及以太网口、串口等通信接口。

3.电源电路设计

本系统共需 1.8 和 2.5、V 3.3 V种电源 其中由开关电源为系统提供统一的 3.3 电源和地平面 选用 LP3963ES-2型 LDO提供 2.5 电压 该电路的最大负载电流可达 3 能满足系统需要。以 2.5 为输入 选用的 TPS775型 DC/D转换器提供 1.8 电压 可有效减少二次电源的功率损耗。

3.2 CP时钟电路设计

本系统采用一个 33.33 MHz的外部晶体振荡器 连接到 CP的 Syscl脚作为外部输入的低频时钟源 然后通过初始化配置片内锁相环（PL将外部输入的低频时钟源倍频 为系统提供高频系统时钟。

3.3 DDR SDRAM电路设计

由于 PowerPC440G的 DDR SDRAM接口具有最高 6位数据总线和 位 ECC错位 所以选用 个 HY5DU281622ETP- 作为板上内存模块 该电路结构为 8 M 16 bits, 个内存电路共用 个 bankSel选空间 其中 个内存电路构成数据总线为 6位的数据存储区 内存容量为 64 字节，另外 个内存电路的低 位用作 ECC还可扩展 1 个 18的 DIMM(必须采用 7位的 ECC内存条。)

内存模块各电路及 DIMM时钟由 PowerPC440GP的 Memclkout脚提供 其频率等于 PL总线时钟 一般为 100 MHz或 133 MHz)为了保证各器件时钟同步 引入 CY230时钟匹配器件 该器件拥有内部时钟锁相环 可以将引脚输出的时钟信号分成 条相位频率完全相同的时钟信号 条供给芯片 条供给 DIMM并且可以避免任一时钟信号反射对其他时钟造成的影响。DDR

SDRAM模块采用 SSTL-信号标准 工作电压为 2.5.V进行 布板设计时要重点考虑 Data/DM/DQ等信号线的走线及长度、信号线的端接方法及符合

SSTL-信号要求的端接电压 V_{ref} 和输入参考电压信号 V_{ref} 的电路设计。Data/DM/DQ等信号线应尽量采用相同的走线结构并严格保持等长。信号线末端的端接推荐使用串并联方式 如图 所示。这里串联端接电阻器的阻值一般为 22 并联端接电阻器阻值一般为 25 位置应尽量靠近信号接收端引脚 可有效减少信号反射和电磁干扰 适应更高的时钟率。对于端接电压 V_{ref} 和参考电压 V_{ref} 采用内存厂商推荐的 ML655来提供。另外 由于 DDR SDRAM电路时序操作的复杂性 结合控制器内部时序寄存器的配置对这部分电路进行板级仿真是非常重要的。

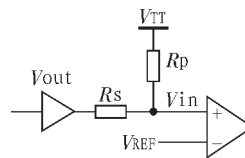


图 2 DDR SDRAM信号线串并联端接

3.4程序存储电路设计

由于 PowerPC440G的 EBC总线具有 3位数据线 选择 个 MX29LV160BT型 Flash器件用以存放 BSP实时操作系统 VxWorks和用户应用程序。将 Flash的 BYT脚设置为高电平 使 Flash工作于x 模式。这样 个 Flash共用 个 CS选空间 构成 EBC数据总线同步访问的高 1位和低 1位 容量为 4 字节。由于 PowerPC440G采用 PowerPC内核 即 440G的 A3是 LSB, A是

MSB数据总线亦然，而Flash的A和D端是LSB连线时要注意引脚次序。另外由于Flash是4字节同步读取操作，连线时应将CP的地址线左移位，即最低位A和A不用。

为了便于系统调试，可增加一个512K的SST39SF040用于BootROM启动，通过跳线与Flash互选CS0或CS2。这样系统启动时会自动从CS0的设备上读取启动代码。程序存储模块如图所示。

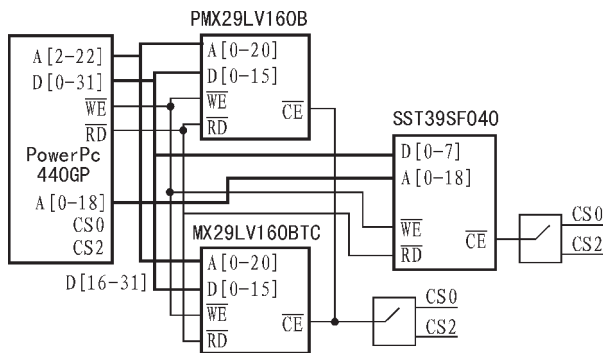


图 3 Flash程序存储电路

3.5以太网接口电路设计

用RTL8201B型网口电路实现PowerPC440GP的100MHz以太网的接口，使用外部25MHz晶体振荡器，通过16PT85网络滤波器连接以太网，如图所示。由于PowerPC440GP带有100MHz以太网MAC，所以能够实现和RTL8201B的无缝连接。RTL8201B有两种接口：S和M。本系统采用MII接口的选择可以通过设置MII/SNIBI脚为高电平，正确设置ANE、SPEED和DUPLEX脚来实现。M能够运行在25MHz和2.5MHz频率，分别为100MHz以太网和10MHz以太网支持。数据传输时，MA首先判定TXEN信号并将TXEN数据改为TXEN数据，再通过TXD[0:3]传到物理层上，在TXEN信号有效期间，

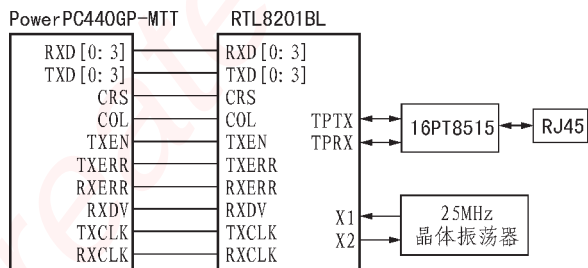


图 4 PowerPC440GP与RTL8201B接口框图

PH通过传输时钟信号TXCLK对TXD[0:3]的数据进行同步采样。接收数据时，PH判定接收使能信号来接收RXD[0:3]的数据。

3.系统启动配置

PowerPC440GP有一个1总线启动配置控制器。当引脚UART0-DCD被置为高电平时使能该控制器。选用一个保存有启动配置信息的AT24C32连接到1C接口作为从设备。当系统上电或者复位时，该控制器能够从位于1C接口的从设备中连续读取16字节。这16字节数据被保存在本上电配置寄存器CPC0-STRP0:3用来初始化PLL设置、片内各总线时钟率、启动位置、启动宽度及一些用户定义的配置等。如果读取数据失败或者引脚UART0-DCD被置于低电平，则这些配置将全部使用默认值。另一配置引脚UART0-DSR置高电平或低电平决定了1总线从设备位地址是0xA还是0xA8。

4 VxWorks的开发与移植

本系统选用Windriver公司推出的嵌入式实时操作系统VxWorks及其集成开发工具Tornado系统开发调试工具采用Windriver公司生产的Vision-16仿真器。仿真器一端连接PC接口，另一端连接PowerPC440GP的JTAG接口。开发时首先调试PowerPC内核和外部DDR SDRAM，一旦工作正常，就可以通过仿真器下载RTOS内存来辅助硬件调试。然后调试网口，如果网口工作正常，就可以脱离仿真器，利用Tornado提供的工具软件（例如WDB）通过网口线建立电路板与PC的通信机制，调试其他模块和开发应用程序。开发程序完成后，将正确的启动代码烧制到BootROM中，由BootROM引导程序通过f将VxWorks内核与应用程序下载烧制到Flash中。

VxWorks广泛流行的重要原因是它的可移植性。通过板级支持包BSP，VxWorks操作系统的应用代码可独立于硬件。系统移植时只需根据硬件平台修改BSP，不需要改变操作系统和应用程序。

5 VxWorksBSP文件结构与开发流程

在VxWorks中，BSP由一组与特定目标系统相关的文件组成。这些文件包括编译文件makefile

ROM初始化文件 romInit,系统初始化文件 sysLib,配置文件 config,目标板定义文件 bspname,网络配置文件 configNet,系统库文件 sysLib,用户配置文件 usrConfig,引导配置文件 bootConfig,引导初始化文件 bootInit,串口文件 sysSerial,以太网接口文件 ibmEmacEnd等,BS启动流程如图 5 所示。

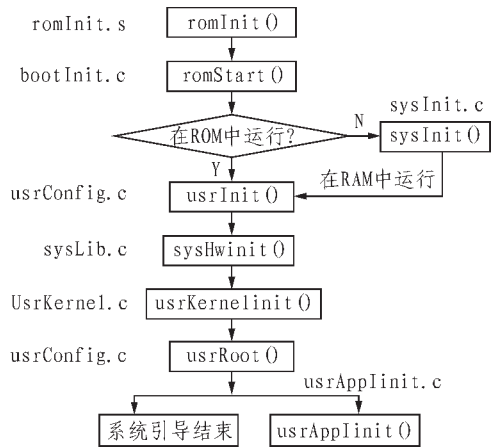


图 5 Vxworks BS启动流程

5.1 建立开发环境

主要以开发板 CP的 BS文件为模板在 \tornado\target\config目录下创建用户的 BS目录 bspname,把 \tornado\target\config下的文件和 BS模板文件拷贝到该目录下。

5.2 修改模板文件

5.2.1 makefile

该文件通过命令行方式对创建映像文件进行控制 必须定义下列宏 :

- CPU: PowerPC440GP;
- TOOL: GNU;
- TGT_DIR目标板目录的路径 使用默认 ;
- TARGET_DIR: BS目录名 自定义 ;
- VENDOR目标生产商名 , IBM;
- BOARD目标板名 自定义 ;
- ROM_TEXT_ADR启动 ROM的入口地址 本系统设为 0xffff80100;
- ROM_SIZE: ROM大小 本系统为 512 KB;
- RAM_LOW_ADRS加载 VxWork的目标地址 本系统设为 0x0010000;
- RAM_HIGH_ADRS将引导 ROM映像复制到RAM的目标地址 本系统设为 0x00c0000

5.2.2 bspname.h

该文件根据 PowerPC440G设置串行接口、时钟及 I/设备等 必须包含以下内容 :
 中断向量 优先级 ;
 I/设备地址 ;
 设备寄存器各位的含义 ;
 系统和附加时钟参数 最大和最小速率)

5.2.3 config.h

该文件包含定义所有与 PowerPC440G目标板相关的组件。

- 设置缺省启动行 使用网络启动 ;
- RA地址和大小 地址从 开始 , 64 M大小
- 设置 ECC使能 ;
- 设置是否支持 MM和 Cache支持 Cache;
- 定时器是否使用外部时钟 否 ;
- 串口时钟定义 使用外部时钟 ;
- 串口默认通道 使用通道 1;
- 包含网口 支持以太网接口 ;
- WD默认通信方式 网口。

另外 注意 ROM_TEXT_ADRS, ROM_SIZE, RAM_LOW_ADRS, RAM_HIGH_ADRS必须与 makefile的定义保持一致。

5.2.4 romInit.s

该文件是一端汇编语言的初始化代码 其中的 romInit函数是 BootROM和基于 ROM的 Vx-Work映像的入口。需要做的工作有 :

- 将相关寄存器清零和置位 ;
- 屏蔽中断 通过设置 MS寄存器中的 C位与 E位 ;
- 关闭数据、指令 Cache;
- 初始化 EBC总线寄存器 通过对 bar寄存器的设置为 EBC总线上的各种设备分配片选信号与地址空间 ;
- 初始化内存 对 DDR SDRAM接口各寄存器进行配置 为板载内存与 DIMM分配片选信号与地址空间。注意时序寄存器 SDRAM0_TR1, SDRAM0_WRDT等的配置应与内存模块的布线仿真结合起来 ;
- 初始化堆栈指针 ;
- 计算 romStart函数地址 然后跳转到该函数执行 语言代码 并不再返回。

5.2.5 用户配置文件 usrConfig.c

其中的代码是 VxWork映像的主要初始化代码,而引导配置文件 bootConfig中的代码是 bootrom映像的主要初始化代码。 bootInit是 ROM初始化的第 阶段, RomInit完成后跳转到此文件中的 romStart(该函数完成 ROM映像所需要的解压缩和重定位工作。以上 个文件一般无须用户直接修改。

5.2.6系统库文件 syslib.c

该文件包含与特定目标系统相关的库函数 这些函数为操作系统和应用程序提供板级接口 使这些程序与硬件系统无关。该文件至少包含下列函数 : sysModel, sysBspRev, sysHwlnit, sysHwlnit和 sys MemTc等 其中 sysHwlnit是本文件的核心 ,大部分硬件初始化工作都在这部分完成 本系统也可选择将包括串口 ,网口等硬件设备的初始化入口程序添加到该函数中 ,相关设备的驱动程序可作为子文件引入 syslib.c

5.3创建 VxWork映像

根据具体需要在命令行环境下利用 makefile创建各种镜像 ,也可以在 Tornado集成环境下在 Build菜单里选择 Build Boot Rom来创建各种类型

的 BootROM

结束语

介绍了一种新型高性能嵌入式处理器 并对以这种处理器为核心构建的网络服务器的硬件设计和软件开发流程做了详细说明。鉴于不同处理器的嵌入式系统设计具有一定的共性 本文可为其他类型的嵌入式系统开发提供一定的参考。

参考文献 :

- [1] IBM Corporate.PowerPC440GP Embedded Processor User's Manual[Z]. 2002.
- [2] WindRiver. VxWorks Reference Manual[Z]. 1999.
- [3] WindRiver. VxWorks BSP开发人员指南 [M] 金剛 苏琪 杨锡励 译 北京 清华大学出版社 , 2003.

作者简介 杨凡 (1977-男 硕士毕业于华中科技大学自动控制系 现任福建实达电脑设备公司研发工程师。

收稿日期 : 2005- 12- 31

咨询编号 : 060707

上接第 页 用外部中断 服务程序实现动感的实时控制 这样 ,MCU在 D变成高电平后即可控制动感平台动作 提高了实效性。程序流程如图 7所示。

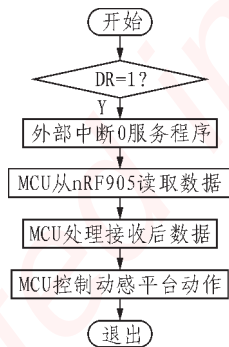


图 动感控制编程流程

5结束语

据考证 国内现有的车辆模拟驾驶训练系统均

未安装动感系统。车辆模拟驾驶动感系统具有广阔的发展前景。本文为该系统的设计提供了切实可行、安全可靠的参考设计方案。

参考文献 :

- [1] Honeywell.Digital Compass Module HMR3000 User's Guide[Z].
- [2] Nordic Semiconductor ASA. Single Chip 433/868/915 MHz Transceiver nRF905[Z].
- [3] 求是科技 单片机典型模块设计实例导航 [M] 北京 人民邮电出版社 , 2004.

作者简介 崔加前 (1979-男 山东滕州人 硕士研究生 主要研究方向是交通智能控制和机电一体化。

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](#)

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](#)

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 CC++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](#)

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](#)

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 IO 口的扩展方法](#)

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](#)

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](#)

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](#)