

基于 SJA1000 的 CAN 总线智能控制系统设计

SJA1000 - Based CAN - Bus Intelligent Control System Design

武汉理工大学系统仿真与控制中心 陆前锋 刘波 陈明昭

Lu Qianfeng, Liu Bo, Chen Mingzhao

摘要:本文介绍了一种新型的现场总线控制器 SJA1000 的基本原理结构及功能特点,重点叙述了基于 CAN 控制器 SJA1000 的智能控制系统硬件电路及软件设计,给出了初始化程序和部分通信程序。

关键词:CAN 总线 SJA1000 智能控制 系统设计

Abstract: This paper introduces the elementary configuration and functional characteristics of SJA1000, a hard circuit and soft configuration of the intelligent controller system based on CAN controller SJA1000 is presented in the paper. At last, the initialization and communication program are given.

Keywords: CAN bus SJA1000 Intelligent controller System design

中图分类号: TP571 文献标识码: A 文章编号: 1003-7241(2003)01-0061-04

1 引言

CAN 总线是德国 Bosch 公司从 80 年代初为解放汽车中众多的控制与测试仪器之间的数据交换而开发的一种串行数据通信协议,它是一种多主总线,通信介质可以是双绞线、同轴电缆或光导纤维。通信速率可以 1 Mbps,通信距离可达 10KM。CAN 总线通信接口中集成了 CAN 协议的物理层和数据链路层功能,可完成对通信数据的成帧处理,包括位填充、数据块编码、循环冗余检验、优先级判别等工作。CAN 协议的一个最大的特点是废除了传统的站地址编参与,而之以对通信数据进行编码,这使得网络内的节点个数理论上不受限制。CAN 协议采用 CRC 校验,并可以提供相应的错误处理功能,具有极高的可靠性和独特的设计,特别适用工业过程监控设备的互连,在工业现场应用越来越广泛。CAN 总线具有下列主要特性:(1)多主站依据优先权进行总线访问;(2)非破坏性的基于优先权的总线仲裁;(3)借助接收滤波的多地址帧传送;(4)远程数据请求;(5)配置灵活;(6)全系统的数据相容性;(7)错误检测和出错信令;(8)发送期

间若丢失仲裁或由于出错而遭破坏的帧可自动重发送;(9)暂时错误和永久性故障节点的判别以及故障节点的自动脱离 CAN 总线。

2 SJA1000 的特点及功能说明

PHILIPS 公司的 PCA82C200 是符合 CAN2.0C 协议的总线控制器,SJA1000 是它的替代产品,它是应用于汽车和一般工业环境的独立 CAN 总线控制器。具有完成 CAN 通信协议所要求的全部特性。经过简单总线连接的 SJA1000 可完成 CAN 总线的物理层和数据链路层的所有功能。其硬件与软件设计和 PCA82C200 的基本 CAN 模式 BasicCAN 兼容。同时,新增加的增强 CAN 模式 Pelican 还可支持 CAN2.0B 协议。SJA1000 的主要特性如下:(1)管脚及电气特性与独立 CAN 总线控制器 PCA82C200 兼容;(2)软件与 PCA82C200 兼容(缺省为基本 CAN 模式);(3)扩展接收缓冲器(64 节 FIFO);(4)支持 CAN2.0B 时支持 11 和 29 位标识符;(5)位通讯速率为 1Mbits/s;(6)增强 CAN

模式 (PeliCAN); (8) 采用 24MHz 时钟频率; (8) 支持多种微处理器接口; (9) 可编程 CAN 输出驱动配置; (10) 工作温度范围为 -40 ~ +125。

2.1 CAN 控制模块的说明

2.1.1 接口管理逻辑

接口管理逻辑负责解释来自 CPU 的命令, 控制 CAN 寄存器的寻址, 向主控制器提供中断信息和状态信息。

2.1.2 发送缓冲器

发送缓冲器是 CPU 与 BSP (位流处理器) 之间的接口, 能够存储发送到 CAN 网络上的完整信息。缓冲器长 13 个字节, 由 CPU 写入, BSP 读出。

2.1.3 接收缓冲器 (RXFIFO)

接收缓冲器是接收过滤器和 CPU 之间的接口, 用来存储从 CAN 总线上接收的信息。接收缓冲器 (RXB, 13 个字节) 的一个窗口, 可被 CPU 访问。CPU 在此 FIFO 的支持下, 可以在处理信息的时候接收其他信息。

2.1.4 接收过滤器

接收过滤器把它的数据和接收的识别码的内容相比较, 决定是否接收信息。在纯粹的接收测试中, 所有信息都保存在 RXFIFO 中。

2.1.5 位流处理器 (BSP)

位流处理器是在发送缓冲器、RXFIFO 和 CAN 总线之间控制数据流的程序装置。它还执行错误检测、仲裁、总线填充和错误处理。

2.1.6 位时序逻辑 (BTL)

位时序逻辑监视串口的 CAN 总线和处理与总线有关的位时序。它在信息开关“弱势—支配”的总线传输时同步 CAN 总线位流 (硬同步), 接收信息时再次同步下一次传送 (软同步)。BTL 提供了可编程的时间段来补偿传播延迟时间、相位转换 (如由于振荡漂移) 和定义采样点及每一位的采样次数。

2.1.7 错误管理逻辑 (EML)

EML 负责传送层中调制解调器的错误管理。它接收 BSP 的出错报告, 使 BSP 和 IML 进行错误统计。通常情况下, 错误计数器计数达到 255 时, 将关闭总线, 并自动转入复位状态, 并产生错误中断。经过 128 个周期的总线空闲后, 重新开启总线。当错误计数器计数达到 96 时 (警告限制), 将产生错误中断。

2.2 新增功能

前已述及 SJA1000 同时支持 BasicCAN 模式和 PeliCAN 模式。

PeliCAN 模式, 将识别码从 11 位扩展到 29 位使 CAN 总线的节点数大大地扩大, 从而使得 CAN 总线的适用范围更加广泛。除了扩展了识别码, PeliCAN 模式还有以下扩展功能: (1) 可读/写访问的错误计数器; (2) 可编程的错误计数器; (3) 最近一次错误代码寄存器; (4) 对每一个 CAN 总线错误的中断; (5) 具体控制为控制的仲裁丢失中断; (6) 单次发送 (无重发); (7) 只听模式 (无确认, 无活动的出错标志); (8) 支持热插拔 (软件位速率检测); (9) 接受过滤器扩展 (4 字节代码, 4 字节屏蔽); (10) 自身信息接收 (自接收请求)。以上扩展功能为用户更好地识别和处理发送和接收错误、更好地应用和调试 CAN 控制器提供了更多方便。

3 智能控制系统硬件电路设计

根据 SJA1000 的功能特点以及课题的实际需要, 我们选用 SJA1000 作为 CAN 控制器, 并且使用了 CAN 控制器接口芯片 PCA82C250, PCA82C250 是 CAN 协议控制器和物理总线的接口。此器件对总线提供差动发送能力, 对 CAN 控制器提供差动接受能力。整个系统以 Intel 16 位单片机 80C196KC 为核心。SJA1000 在智能控制系统中的硬件配置与 CAN 总线接口的电路如图 1 所示。硬件电路中使用 PCA82C250 的目的是为了增大通信距离, 提高系统的瞬间抗干扰能力, 保护总线, 降低射频干扰, 实现热防护等。

SJA1000 在电路中是一个总线接口芯片, 实现从上位机 PC - CAN 接口到现场微处理器之间的数据通信, 对于微处理器而言, SJA1000 是一个总线接口, SJA1000 片内的存储单元相对 80C196KC 来说是片外的数据存储单元。因此, 可以按照扩展片外数据存储器的形式来访问 SJA1000 的寄存器地址, 80C196KC 是 CAN 控制器的微处理器, 把 80C196KC 的 ALE、RD、WR 和 SJA1000 的 ALE、RD、WR 相连就构成一个最小系统节点。80C196KC 通过地址总线经 GAL 译码来选通 SJA1000, 通过读、写外部数据存储器的形式来访问 SJA1000。该电路的主要功能就是通过 CAN 总线接收来自上位机的数据进行分析组态然后下传给下位机的控制电路实现控制功能, 当智能控制系统接收到下位机的上传数据, SJA1000 就会产生一个中断, 引发微处理器产生中断, 通过中断处理程序接收每一帧信息并通过 CAN 总线上传给上位机进行分析, 以便及时纠正误码、错码。

在进行电路设计时应当注意: 总线两端那两个 124 的电阻, 对匹配总线阻抗起着相当重要的作用。忽略掉它们, 会使数据通信的抗干扰性和可靠性大大降低, 甚至无法通信; 82C250

第 8 脚与地之间的电阻 R_S 称为斜率电阻,它的取值决定了系统处于高速工作方式还是斜率控制方式。把该引脚直接与地相连,系统将处于高速工作方式。在这种方式下,为避免射频干扰,建议使用屏蔽电缆作总线;而在波特率较低、总线较短时,一

般采用斜率控制方式,上升和下降的斜率取决于 R_S 的阻值。通常情况下,15K ~ 200K 为 R_S 较理想的取值范围。在这一种方式下,可以使用平行线或双绞线作总线。

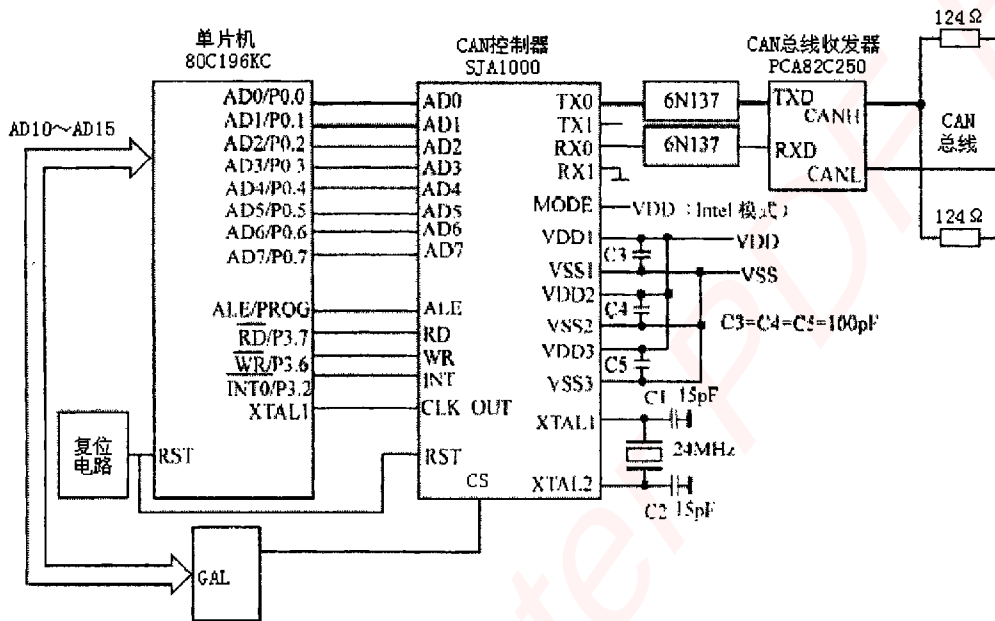


图 1 智能控制系统硬件原理图

4 系统软件设计

系统软件设计的指导思想是系统上电后首先对 80C196KC 和 SJ1A1000 进行初始化,以确定工作主频、波特率、输出特性等,然后通过查询方式获取模数转换采样值,并把该值通过 SJ1A1000 传送到 CAN 总线上由上位 PC 机进行显示控制,而对 CAN 总线上的信息则采用中断方式,系统每接收到一帧信息,便产生一次中断以触发微处理器进入中断,在中断服务程序中读取该帧信息并传送到现场。为防止出现死机和干扰,程序中还采用看门狗技术进行定时监控。对 CAN 总线的初始化设计,对于编程人员来说是重难点,必需对写入的每一个寄存器进行仔细和全面的考虑。如果 CAN 总线的初始化工作没有做好,系统将无法正常工作。假设 SJ1A1000 的首址是 A000H,用汇编语言编制的 SJ1A1000 的初始化程序如下:

```
CR EQU 1000H;控制寄存器
CMR EQU A001H;命令寄存器
SR EQU A002H;状态寄存器
IR EQU A003H;中断寄存器
```

```
ACR EQU A004H;验收码寄存器
AMR EQU A005H;验收屏蔽寄存器
BTR0 EQU A006H;总线定时寄存器 0
BTR1 EQU A007H;总线定时寄存器 1
OCR EQU A008H;输出控制寄存器
DI
LDB CL, # 03H
STB CL, CR;开放接收中断,复位请求位置 1,以开始初始化。
LDB CL, # 01H
STB CL, ACR;将节点 1 标识符送 ACR
LDB CL, # 0FFH
STB CL, AMR;验收滤波
LDB CL, # 00H
STB CL, BTR0;传送波特率为 250kbps
LDB CL, # 14H
STB CL, BTR1;定义位周期宽度,采样点位置及采样次数选
LDB CL, # 0AAH
STB CL, OCH
```

```

SIB CL ,OCH;选择正确输出方式,建立输出驱动器的配置。
LDB CL , # 1AH
SIB CL ,CR;复位请求位置 0,初始化结束
EI
    
```

5 结论

实践表明,SJA1000 是一种较好的 CAN 总线控制器件,它的抗干扰性能优良,通信速率较高,电路也比较简单。在 PeliCAN 模式下,其识别码达 29 位,因而可满足各种应用场合。在自动化控制系统中,合理安排这 29 位识别码可以使许多问题得以简化。PeliCAN 模式增加的各种功能,如:各种错误处理功能大大地提高了用户程序处理各种通信错误的能力;支持热插拔和自身信息接收功能,使用户不必事先知识预设的波特率,从而大大地方便了通信的调试。本文是作者在阅读了大量的有关 CAN 总线资料后进行实际 CAN 总线的应用开发研究工作时的一些

(上接第 57 页)

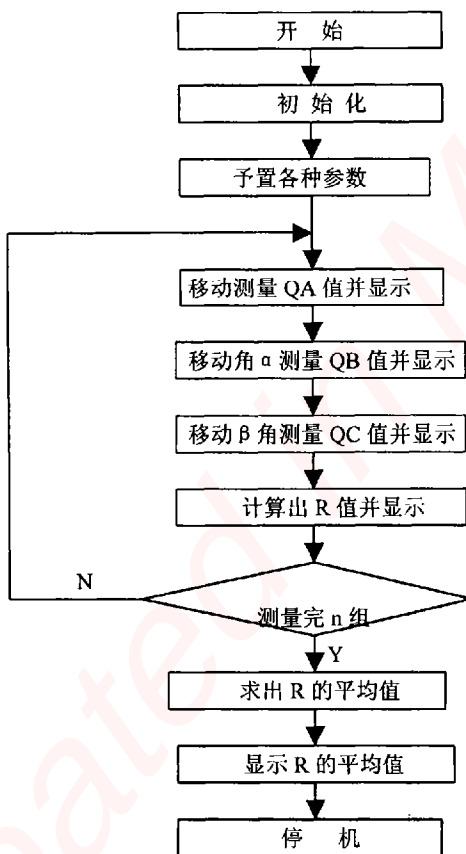


图 5 程序框图

心得,同时希望现场总线技术能够在自动化领域中应用的更加广泛。

6 参考文献

- [1] PHILIP Semiconductors 公司 . DATA HANDBOOK 80C51 - based 8 - bit Microcontrollers[S]. 1995
- [2] 肖海荣,周风余 . 基于 SJA1000 的 CAN 总线系统智能节点设计[J]. 计算机自动测量与控制,2001,9(2)
- [3] 邬宽明 . CAN 总线原理和应用系统设计[M]. 北京航空航天大学出版社
- [4] SJA1000 独立 CAN 控制器介绍[S]. 广州周力功单片机发展有限公司
- [5] 独立 CAN 器件 SJA1000 的应用[J]. 南京工程学院学报,2002 年 3 月

陆前锋(1978 -),男,硕士研究生,主要从事系统仿真与控制技术方面的研究。

5 测量仪程序框图

测量系统的程序框图如图 5 所示,测量前根据被测工件的尺寸予置步进电机 A、B、C 的移动距离,位移传感器测头绕旋转中心旋转的角度 α_i ($i = 1, 2, 3 \dots, n$), n - 测量组数,参数予置完成后系统开始测量,测完一组(3 次)后,计算出 R 值存贮并显示,当 n 组数据全部测完后,求出 R 的平均值。

6 结束语

本文详细介绍了模具圆弧内径自动测量仪的整体结构、测量原理、硬件组成、软件框图,从而证明了模具圆弧内径测量的可行性。本自动测量仪还可以通过改变滑座和位移传感器的安装方式可适用于对圆弧小半径内圆和外圆弧外径进行测量。虽然开发此测量仪投资较大,但可以省去做大量样板的时间和费用,对于提高劳动生产率和产品质量都具有十分重要意义,特别适用于大批量模具生产的测量。

7 参考文献

- [1] 陈建国,朱正德 . 工件非整圆部分内径的快速测量[J]. 机械设计与制造,1999,5
- [2] 余永权,李小青,陈林康 . 单片机应用系统的功率接口技术[M]. 北京航空航天大学出版社,1993 年

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](#)
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](#)
65. [具有 MVB 接口的输入输出设备的分析](#)
66. [基于 STM32 的 GSM 模块综合应用](#)
67. [基于 ARM7 的 MVB CAN 网关设计](#)
68. [机车车辆的 MVB CAN 总线网关设计](#)
69. [智能变电站冗余网络中 IEEE1588 协议的应用](#)
70. [CAN 总线的浅析 CANopen 协议](#)
71. [基于 CANopen 协议实现多电机系统实时控制](#)
72. [以太网时钟同步协议的研究](#)

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](#)
43. [实时操作系统 VxWorks 在微机保护中的应用](#)
44. [基于 VxWorks 的多任务程序设计及通信管理](#)
45. [基于 Tilcon 的 VxWorks 图形界面开发技术](#)
46. [嵌入式图形系统 Tilcon 及应用研究](#)
47. [基于 VxWorks 的数据采集与重演软件的图形界面的设计与实现](#)
48. [基于嵌入式的 Tilcon 用户图形界面设计与开发](#)
49. [基于 Tilcon 的交互式多页面的设计](#)
50. [基于 Tilcon 的嵌入式系统人机界面开发技术](#)
51. [基于 Tilcon 的指控系统多任务人机交互软件设计](#)
52. [基于 Tilcon 航海标绘台界面设计](#)
53. [基于 Tornado 和 Tilcon 的嵌入式 GIS 图形编辑软件的开发](#)

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 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](#)
27. [基于 XPEmbedded 的警务区 SMS 指挥平台的设计与实现](#)
28. [基于 XPE 的数字残币兑换工具开发](#)

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](#)
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](#)
26. [嵌入式微机 MPC555 驻留片内监控器的开发与实现](#)
27. [基于 PowerPC 和 DSP 的电能质量在线监测装置的研制](#)
28. [基于 PowerPC 架构多核处理器嵌入式系统硬件设计](#)
29. [基于 PowerPC 的多屏系统设计](#)
30. [基于 PowerPC 的嵌入式 SMP 系统设计](#)

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](#)
35. [基于 S5PC100 移动视频监控终端的设计与实现](#)

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 COM Express 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](#)
32. [面向 OpenCL 模型的 GPU 性能优化](#)
33. [基于 GPU 的 FDTD 算法](#)
34. [基于 GPU 的瑕疵检测](#)
35. [基于 GPU 通用计算的分析与研究](#)
36. [面向 OpenCL 架构的 GPGPU 量化性能模型](#)
37. [基于 OpenCL 的图像积分图算法优化研究](#)
38. [基于 OpenCL 的均值平移算法在多个众核平台的性能优化研究](#)
39. [基于 OpenCL 的异构系统并行编程](#)
40. [嵌入式系统中热备份双机切换技术研究](#)

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](#)
10. [基于小波变换与偏微分方程的图像分解及边缘检测](#)
11. [基于图像能量的布匹瑕疵检测方法](#)
12. [基于 OpenCL 的拉普拉斯图像增强算法优化研究](#)
13. [异构平台上基于 OpenCL 的 FFT 实现与优化](#)

FPGA / CPLD:

1. [一种基于并行处理器的快速车道线检测系统及 FPGA 实现](#)
2. [基于 FPGA 和 DSP 的 DBF 实现](#)
3. [高速浮点运算单元的 FPGA 实现](#)
4. [DLMS 算法的脉动阵结构设计及 FPGA 实现](#)
5. [一种基于 FPGA 的 3DES 加密算法实现](#)
- 6.