

基于 DSP 的 CANopen 通讯协议的实现

An Implementation of CANopen Protocol Based on DSP

(沈阳中国科学院沈阳自动化研究所 ;北京中国科学院研究生院 马艳歌^{1,2} 贾凯¹ 徐方¹
Ma,Yange Jia,Kai Xu,Fang

摘要 : CANopen 是一种开放的应用层协议 其应用可以进一步提高系统的可靠性、通讯效率及灵活性 而且可以使产品具有很好的兼容性。本文采用 CANopen 通讯协议实现了 CAN 总线 DSP 系统与上位机 CAN 卡之间的通讯 并通过测试实验验证了信息传递的可靠性 保证了全数字网络化伺服驱动系统中对电机控制的准确性和实时性。

关键词 : DSP; CAN 总线 ; CANopen 协议 伺服驱动

中图分类号 : TP273 文献标识码 : A

Abstract: CANopen protocol is an open application protocol, the use of it enhances the agility and efficiency in communication and improves interoperability and interchangeability among products from different manufacturers. This paper describes its implementation on the communication system between CAN-bus DSP system and CAN card in PC system. The reliability of communication is verified by testing experiment and the results illustrate that this solution can ensure the veracity and real time feature of motion control system.

Key Words: DSP; CAN-bus; CANopen protocol; servo driver

引言

CANopen 是一个开放的、标准化的应用层协议 在各种控制系统中得到了广泛的应用。依靠 CANopen 协议的支持 可以对不同 CAN 商的设备通过总线进行实时通讯。

本文针对自主研发的全数字电机伺服驱动系统, 使用 CANopen 通讯协议实现了 CAN 总线数字信号处理器 (DSP) 系统与上位机 CAN 卡之间的通讯 并通过测试实验验证了信息传递的可靠性 保证了全数字网络化伺服驱动系统中对电机控制的快速性、准确性和实时性。

2 通讯系统实现的基本原理

2.1 LF240 系列 DSP 的 CA 控制器模块

LF240 系列 DSP 的 CA 控制器模块 是一个完全的 CA 控制器。具有以下特性 :

(1) 完全支持 CAN2.0 协议 ;

(2) 对象有 4 个邮箱 用于接收和发送信息 其数据长度为 16 字节 ;

(3) 可编程的中断配置 位定时器及 CAN 总线唤醒功能 ;

(4) 当发送出现错误或仲裁丢失数据时 , CA 控制器有自动重发功能 ;

(5) 总线错误自诊断功能 ;

(6) 支持自测试模式。

2.2 CANopen 协议

CANopen 协议以 CAN 芯片为硬件基础 , 有效利用 CAN 芯片所提供的简单通信功能区实现工业控制网络的复杂应用层协议要求。应用中应该注意以下几个方面的内容 :

2.2.1 对象字典 OD (Object Dictionary)

OD 是 CANopen 协议的核心概念 , 它是一个有序的对象组 每个对象用一个 16 位的索引值和一个 8 位的子索引寻址。CANopen 网络中的每个节点都有一个 OD, 包含了描述这个设备和它的网络行为的所有参数。

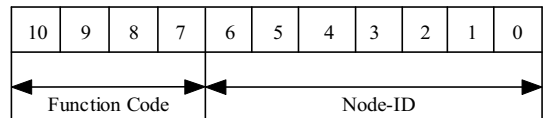


图 1 CANopen 标识符

2.2.2 CANopen 标识符

CANopen 定义了强制性的缺省标识符 (COB-ID) 分配表。它是基于 16 位的 CAN-ID 包括 16 位功能段 (Function Code) 和 16 位地址段 (Node-ID) 如图 所示。文献中给出了 CANopen 预定义主从连接集 CAN 标识符分配情况。

表 1 NM 消息格式

COB-ID	Byte0	Byte1
0x000	命令字	Node-ID

2.2.3 CANopen 通讯机制

马艳歌 硕士研究生

中国科学院知识创新工程重大项目 (KGCX-SW-15)

CANopen通讯模型定义了四种报文 (通讯对象) : 图如图 所示。

1管理报文 NMT(Network Management)

用于网络中主节点对从节点监控和管理。此消息不需要应答。 NMT消息格式如表 所示。

服务数据对象 SDO(Service Data Object)

SDO通过使用索引和子索引来访问一个设备的 QD 协议属于确认服务类型。根据传送数据的长度不同, 有两种传送机制 (加速传送 最多 字节 和分段传送 大于 字节。) 发送和应答报文总是包含 字节信息 基本结构如表 所示 :

表 2a Client->Serve/Serve->Client (加速传送)

Byte0	Byte1~2	Byte3	Byte4-7
SDO 命令字	对象索引	对象子索引	...

表 2b Client->Serve/Serve->Client (分段传送)

Byte0	Byte1...7
SDO 命令字	最大 7 字节数据

表 3 PDO 映射参数

对象 0x1A01:第二个发送 PDO 映射		
子索引	值	意义
0	2	2 个对象映射到 PDO 中
1	0x60000208	对象 0x6000, 子索引 0x02, 由 8 位组成
2	0x64010110	对象 0x6401, 子索引 0x01, 由 16 位组成

表 4 PDO 报文

COB-ID	Byte0	Byte1	Byte2
0x280+ Node-ID	对象 1 的 8 位数据	对象 2 的 数据的低 8 位	对象 2 的 数据的高 8 位

3过程数据对象 PDO(Process Data Object)

用于传输正常网络操作中的实时数据。每个 PDO 在 ODB 用 个对象描述 : PD通讯参数和 PD映射参数。在此举例说明 PD的使用方法。假设第二个发送 PD映射如表 所示 如果这个报文被发送 则由 字节组成 报文结构如表 所示。

定义报文或特殊功能报文

定义了三个用于同步 应急指示和时间标记的特定对象。用于配合以上三种通讯机制。

系统通讯的软、硬件设计

3.1 CAN总线硬件设计

由于 TMS320LF2407芯片内自带了 CAN总线控制器模块 使得 CAN总线的电路设计变得非常简单。将 TMS320LF2407的 CANTX CANRX引脚和 CAN收发器 PCA82C25的 TXD RXD引脚相连即可。 CAN总线之间需要使用一个合适的终端电阻连接。为了进一步提高系统的抗干扰能力 可以在控制器和传输介质之间加接光电隔离器件 6N137电路原

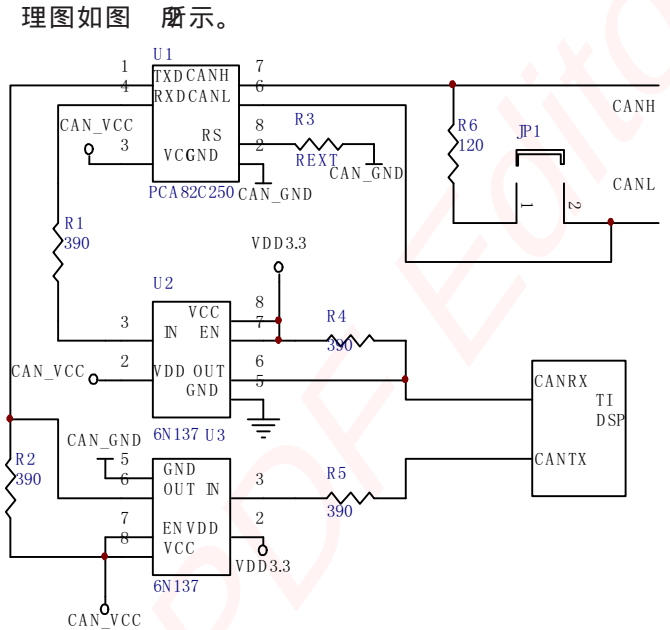


图 2 CAN总线收发电路

3.2 CANopen协议对象字典 ODB的建立

本伺服驱动器通讯支持 CANopen通讯子协议 (DS301和驱动设备子协议及运动控制命令 (DSP-402) 在开发的过程中 根据应用的具体要求 将数字输入 输入模块、电机及驱动设备、控制器、编码器等设备类型及其相应的功能在设备描述的协议中进行了描述。建立的 ODB为三个部分 :

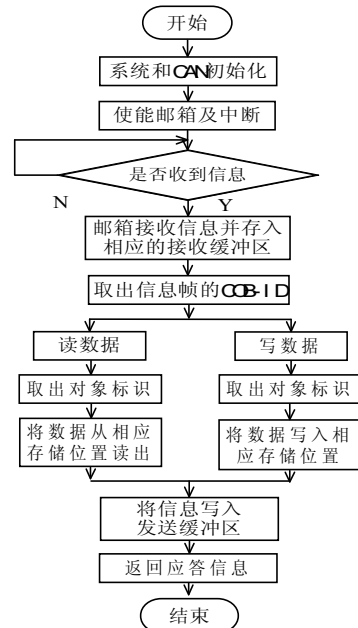


图 3 通讯程序流程图

1通讯协议对象 (Index 0x1000- 0x1AFF)

定义设备类型 错误寄存器 厂商指定状态寄存器, COB-ID同步信息 通讯循环周期 同步 CA信息窗口 以及设备名 软硬件版本号等。另外 SDO的通讯参数以及 PDO的通讯和映射参数也在该部分定义。本

设计中采用固定 PD映射 不允许动态映射。

2标准设备子协议对象 (Index 0x6000- 0x65FF)

定义三类信息 第一类是电机及驱动器的相关设备参数 如类型 厂商信息 分类号等 第二类是命令字和状态字 第三类是位置、速度的相关参数。

3厂商指定协议对象 (0x2000- 0x20FF)

根据厂商需求 灵活定义所使用的变量。此部分的使用使 CANopen协议既保证了设备的通用性 又保证了系统良好的扩展性。

3通讯程序的设计及实现

针对带 CA控制器的 TMS320LF2407和 PC CAN之间的通讯进行了软件设计。

TMS320LF2407的 CA控制器的邮箱 配置为接收方式 邮箱 配置为发送方式。都采用标准信息帧格式 (11Bit ID) 其中发送信息采用查询方式 接收信息采用中断方式 邮箱 属低中断优先级。信息传输波特率设置为 500Kbps

采用 CANopen协议对设备进行控制的过程可以理解为用 SD数据字典的读写过程。检测设备对象的状态或查询设备当前信息时 可用 SD读取数据字典中的对象 要对设备发布控制命令可利用 SD相应的数据字典对象写入数据。接收到信息帧时首先要将其命令字节 (byte)读出 以确定下一步的读写命令处理。

信息帧的接收和发送均采用缓冲区的方式。读操作时 发送信息帧长度为 字节 应答信息帧约定为 字节 写操作时 发送信息帧长度约定为 字节 应答信息帧为 字节。

采用不同的发送邮箱发送数据长度不同的信息帧。本程序设计中 采用邮箱 发送 字节长度的信息帧 采用邮箱 发送 字节长度的信息帧。这种方式可以提高信息传输的效率。软件设计流程图如图 所示。

4测试结果

为了测试 CAN总线 DS系统与 PC CAN之间的通讯情况 采用 VC编写了友好的人机界面 见图 4)进行通讯测试时 首先要选择通讯端口 设置通讯波特率, ID屏蔽位, 信息帧格式及传输数据的格式 (十进制或十六进制。)特别需要注意 PC端设置的通讯波特率要和 DS端的通讯波特率保持一致。然后打开 CAN总线端口 选定发送 SD的功能段及地址段, 填写信息帧长度 以字节为单位 及内容 即可进行通讯测试。

例如, 要将 号地址的电机轮速期望值设为 0x3E电机轮速期望值在 0中的索引为 0x606使用启动域下载协议 加速传输。则设置地址段为 信息帧长度为 8,信息帧内容为 2B 6B 60 00 E8 03 00 00发送成功后返回应答信息帧内容为 : ID=581

LENTH=4, DATA=60 6B 60 0 0 0 0 0说明写入操作正确 见图 4)

要将 号地址的电机轮速期望值读出, 即读出 0中的对象 0x606使用启动域上传协议 服务器使用加速传输方式应答。则设置地址段为 信息帧长度为 信息帧内容为 40 6B 60 00 00 00 00 00发送成功后返回应答信息帧内容为 : ID=581, LENTH=8, DATA=4B 6B 60 0 E8 03 0 说明读取操作正确 图 4)

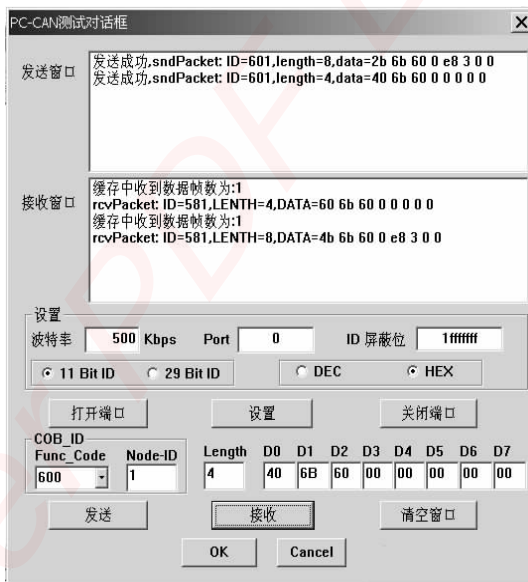


图 4 测试结果图

结束语

本文用 CANopen协议实现了 CAN总线 DS系统与上位机 CAN的通讯 实现了控制系统的数字化、模块化 并保持了系统的可扩展性 为提高伺服驱动系统的智能化水平奠定了坚实的基础。

参考文献

- [1]H. Boterenbrood. CANopen: high-level protocol for CAN-bus.
- [2]Elmo Motion Control. CANopen DS 301 Implementation Guide. Elmo Motion Control. 2003.9
- [3]Elmo Motion Control. CANopen DSP 402 Implementation Guide. Elmo Motion Control. 2003.9

陶一波等 大规模视频会议中的多人桌面共郭系统的设计与实现 [微计算机信息, 2005, 8- 3: 139- 141

作者简介 马艳歌 (1981- 女 汉 硕士研究生。研究方向为伺服电机驱动、智能控制 ; E-mail: yangezi@sia.

贾凯 (1964- 男 汉 副研究员 研究方向是 伺服电机驱动、智能控制、智能机械、机器人 徐方 (1962-), 男 汉 研究员 研究方向 智能控制 机器人技术。

Author Introduction: Ma Yange (1981-) Female, Han Nationality, Master, Research Field: Servo Driver, Intelligent System; Jia Kai (1964-) Male, Han nationality, Associate Professor, Research Field: Servo Driver, Intelligent Control, Intelligent Mechanism, Robotics; Xu Fang (1962-) Male, Han Nationality, Professor, Research Field: Intelligent Control, Robotics (23页)

测试传进扫描链里 所以对于结果的观察就没有问题。但是在改进后的方法中 输出结果要怎样观察将是一个问题。

如果输出的向量只有一位就不会有这样的问 题。因为我们现在要产生一个新的测试向量所需要的时间只有一个时钟周期 所以时间刚刚好。但是一般而言 待测电路的输出向量都会超过一位 所以我们在观察电路输出结果这个步骤上就会有问 题。为了解决这个问题 需要对传统的基于扫描的测试电路做些修改。在待测电路的输出部分 可以用 MISR 来取代原来的扫描链电路 也就是将输出送到 MISR 做特征分析 如图 所示。

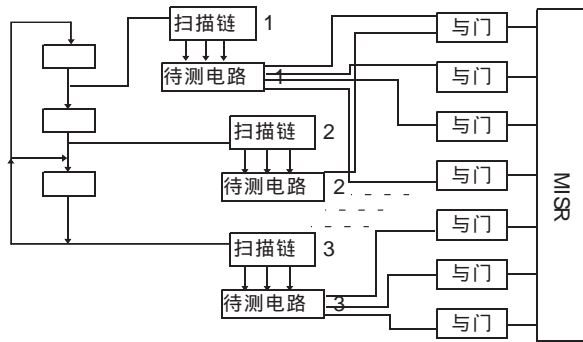


图 6 改进后测试结构

但是由于电路越来越复杂 电路的主要输入与输出的个数就会很多 如果对这样的待测电路加测试电路 将会出现外加电路面积过大的问题。所以我们针对 MISR 部分提出一个方法 使得因测试而增加得 MISR 可以尽量地缩小。

缩小 MISR 大小最直接的想法是将主要输出经组合电路压缩变小 但是这样会使错误覆盖率降低 使测试向量变多 这些问题在文献中谈到了。造成错误覆盖率降低的原因是如果我们将有共同输入向量的两个主要输出通过组合电路合并的话 将可能会造成原本可以测试到的错误 在合并后就测试不到了 所以造成了错误覆盖率的降低。

为了避免这个情况的发生 我们在向量压缩的过程中 只有将没有共同输入向量的主要输出合并在一起 就不会有问题出现了。如图 所示 两个不同的待测电路的主要输出经过合并以后输出到 MISR 这样用一个与门代替了一个寄存器以便减少面积 而错误覆盖率则因为两两合并的主要输出端都没有相同的主要输出资料 所以就没有错误覆盖率降低的问题存在 即原来可以被侦测到的错误 在经过主要输出端合并之后仍然可以被侦测到。

4 结论

经过改进后的基于扫描的测试架构 用向量压缩来减少测试时间 并且将 Test-Per-Scan 的测试方式改成了 Test-Per-Clock 的测试方式 加快了测试的速度 在不影响故障覆盖率的前提下将 MISR 缩小 减少了

硬件消耗 满足了对大型电路进行测试的需要。

参考文献

- [1]Nan_Chen Lai "A Reseeding Technique for LFSR-Based BIST Applications", in Dissertation And Thesis Abstract System, JULY 1999, pp.6- 9
- [2] I.Hanzoglu, J.H.Patel, Reducing test application time for built-in self-test test pattern generator's, in Proc.18th IEEE VLSI Test Symposium, 2000 ,pp.369- 375.
- [3]Sheng_Nan Chiou "Generating Efficient Tests for Continuous Scan", in Dissertation And Thesis Abstract System, AUGUST 1999,pp.24- 46
- [4]陶锴 丁,国清 轮胎智能监测系统的研究 [微计算机信息 , 2005, 4:82- 84
- [5]Chi-Min Chen, S.J.Wang, "Reducing Testing for Logic- BIST", in Dissertation and Thesis System, 2001,p4- 27.
- [6]S.Jacob, Reducing the MISR Size, in Proc IEEE Trans. Compute, Vol.45, no.8, Aug. 1996, pp.930- 938.

作者简介 唐玉兰 (1981- 女 汉族 江苏省丹阳市 , 江南大学信息学院在读硕士研究生 主要研究方向为大规模集成电路的可测性设计。 E-mail:shaoy-i323@126.com; 宗光 男 教授 博士 博士生导师 , 58 研究所副总工程师 中国电子学会理事、学术委员会委员 主要研究方向为大规模集成电路设计。

(214000)江南大学信息工程学院 唐玉兰 于宗光 李天阳 黄越

中国电子科技集团公司第五十八研究所 于宗光 (School of Information Technology, Southern Yangtze University, 214000) Tang,Yulan Yu,Zong-guang Li,Tianyang Huang,Yue

(China Electronics Technology Group Corporation No.58 Research Institute) Yu,Zongguang

通信地址 : (214000)江苏省无锡市滨湖区震泽三村 80 号 40信箱 唐玉兰

(School of Information Technology, Southern Yangtze University, Wuxi Jiangsu, 214000) Tang,Yulan

投稿日期 :2005.6.26) 修稿日期 :2005.7.6)

接第 14 版)

(110016)沈阳中国科学院沈阳自动化研究所 马艳歌 贾凯 徐方

(100039)北京中国科学院研究生院 马艳歌 (Shenyang Institute of Automation , Chinese Academy of Sciences, Liaoning Shenyang, 110016,China) Ma,Yange Jia,Kai Xu,Fang

(Graduate School of the Chinese Academy of Sciences, Beijing, 100039, China) Ma,Yange

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](#)
73. [基于 CANopen 的列车通信网络实现研究](#)
74. [基于 SJA1000 的 CAN 总线智能控制系统设计](#)
75. [基于 CANopen 的运动控制单元的设计](#)
76. [基于 STM32F107VC 的 IEEE 1588 精密时钟同步分析与实现](#)

77. [分布式控制系统精确时钟同步技术](#)
78. [基于 IEEE 1588 的时钟同步技术在分布式系统中应用](#)
79. [基于 SJA1000 的 CAN 总线通讯模块的实现](#)
80. [嵌入式设备的精确时钟同步技术的研究与实现](#)
81. [基于 SJA1000 的 CAN 网桥设计](#)
82. [基于 CAN 总线分布式温室监控系统的设计与实现](#)

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](#)
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 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](#)
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 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](#)
- [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 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](#)
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](#)
14. [数据结构考题 - 第 4 章 串](#)
15. [数据结构考题 - 第 4 章 串答案](#)
- 16.

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. [可编程 FIR 滤波器的 FPGA 实现](#)
7. [基于 FPGA 的 AES 加密算法的高速实现](#)
8. [基于 FPGA 的精确时钟同步方法](#)
9. [应用分布式算法在 FPGA 平台实现 FIR 低通滤波器](#)
10. [流水线技术在用 FPGA 实现高速 DSP 运算中的应用](#)
- 11.