基于ARM7的MVB_CAN网关设计

憨光明，王玉松
(西南交通大学智能控制与仿真工程研究中心，四川　成都　610031)

摘　要：介绍了用于机车内部数据通讯的MVB网络和CAN总线网络的报文结构，给出了MVB-CAN总线网关的硬件和软件实现方法。

关键词：MVB；CAN；MVB-CAN网关

0　引言

随着网络技术和控制技术的发展，机车信息化控制技术向着更高的方向发展。为此，IEC制定出新的列车通信网路国际标准TCN（机车通信网络），以满足列车微机控制系统中对车载数据通信网提出的新要求。

按照TCN标准，列车通信网可分为两级：第一级是WTB绞线式列车总线（Wired Train Bus），可实现车辆间的数据通信；第二级是MVB（Multifunction Vehicle bus）多功能车辆总线，主要实现同一个车辆内各个功能控制单元之间的数据通信。

多功能车辆总线（MVB）是主要用于有互操作性和互换性要求的互连设备之间进行串行数据通信的一种总线，MVB以其高实时性、高可靠性及可管理性等多方面的优势而广泛的应用在列车总线控制当中。

控制器局部网（CAN-CONTROLLER AREA NETWORK）是BOSCH公司为现代汽车应用推出的一种多主机局部网。由于该网的卓越性能，现已广泛应用于工业自动化、多种控制设备、交通工具、医疗仪器以及建筑、环境控制等众多领域。

对于现在的机车总线通信，MVB总线主要应用在单台机车内的车载设备间的通讯，而CAN总线则用于车载设备内部模块间的数据通讯。因此，实现CAN总线与MVB总线之间的通信就成为工程实践中一个必须解决的问题。

1　报文

1.1　CAN报文

CAN报文主要包括数据帧、远程帧、出错帧、超载帧等四种帧。

在总线中传送的报文，每帧一般由7部分组成。CAN协议支持两种报文格式，它们之间唯一的不同是标识符（ID）长度不同。它们的标准格式为11位，扩展格式为29位。

在标准格式中，报文的起始位通常也称为帧起始（SOF），之后，一般是由11位标识符和远程发送请求位（RTR）组成的仲裁场。RTR位用于标明是数据帧还是请求帧，请求帧中没有数据字节。控制场的包括的标识符扩展位（IDE）用于指出是标准格式还是扩展格式。此外，它还包括一个保留位，以便为将来扩展使用。它的最后四个字节用来指明数据场中数据的长度（DLC）。数据场范围为0~8个字节，其后有一个检测数据错误的循环冗余检查（CRC）。CAN的标准帧格式如图1所示。

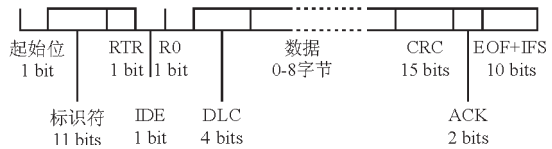

图1　CAN的标准帧格式

应答场通常（ACK）包括应答位和应答分隔符。发送站发送的这两位均为隐性电平（逻辑1），这时，正确接收报文的接收站会发送主控电平（逻辑0）覆盖它。用这种方法，发送站可以保证网络中至少有一个站能正确接收到报文。报文的

尾部应由帧结束标出。在相邻的两条报文间有一很短的间隔位，如果这时没有站进行总线存取，则总线将处于空闲状态。

1.2 MVB报文

MVB报文主要包括两种帧：主帧和从帧。其中主帧只能由总线主设备发送，而从帧则为响应主帧而由从设备发送。

主帧应传送16位数据字，发送的数据字的第一位应为它的最高有效位，用bit0来表示，其内容为16位字中的最高前四位，应为FCODE。次有效的后12位用于表示Focde所指定的地址或参数。因此，主帧有固定的格式，为34 bit (9 bit主帧分界符+16 bit数据位+8 bit校验位+1 bit终止分界符)。一个主帧应以主起始分界符开始，其后为16位帧数据，接着为8位校验序列。MVB的主帧格式如图2所示。

一个从帧应以从帧起始分界符开始，接着为16、32、64、128或256位帧数据，在每64个数据位后还包含一个8位的校验序列或当帧数据只有16或32位时将一个8位的校验序列附在其后，其MVB的从帧格式如图3所示。

2 CAN总线与MVB总线协议转换

CAN总线和MVB总线都属于现场总线范畴，因此，它们都有自己的网络分层结构。对于大多数现场总线，它们都包括物理层，数据链路层和应用层。

MVB有过程数据和消息数据之分。基于过程数据的网络分层是三层 (物理层、数据链路层、应用层)，而基于消息数据的网络分层则有7层。由于消息数据比较复杂且应用场合较少，因此，本文只讨论过程数据。但实际上，有时也可以将MVB视为一个三层协议的网络。

CAN网络分层结构按照CAN总线协议来说，就只有两层即物理层和数据链路层。但是涉及到具体的应用，它实际上也包括应用层。

为了实现两种总线间的协议转换，通常需要首先对他们的帧格式进行转换，其主要原因是帧字节数大小不同，而帧的编码解码部分是由网卡硬件完成的；其次是对应用层的转换，即通过一定的形式将CAN总线中表示过11位标志符的数据含义和MVB总线中过程变量的数据集对应起来，以实现数据之间的传输。

3 MVB–CAN网关的硬件实现

为了设计方便，本设计的CAN侧采用三星公司生产的S3C44B0微控制器，MVB侧则采用MVBC01 ASIC专用芯片。

MVB通信控制器选用的MVBC01 ASIC专用

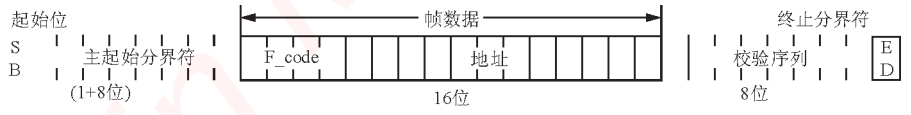

图2 MVB的主帧格式

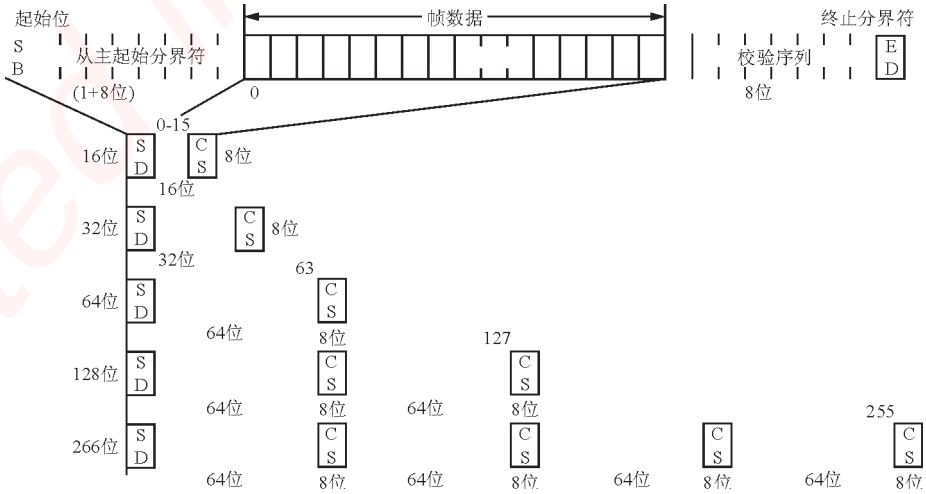

图3 MVB的从帧格式

芯片采用的是16 bit数据总线，该芯片同时可提供丰富的接口控制信号（如RDY、TMRDY等待信号），因此，可用于简化与各种宿主CPU以及通信存储器的接口设计。此外，MVBC01还内嵌有完整的MVB1类设备协议，其中，MVB物理层接口采用电气短距离介质ESD+接口，系统信号通道则使用光耦来实现主系统与外界的电隔离，从而提高了系统的可靠性。另外，系统还使用RS 485芯片作为收发器，并使用过压保护模块来防止瞬间过压对器件造成的损坏。MVB侧的接口单元如图4所示。

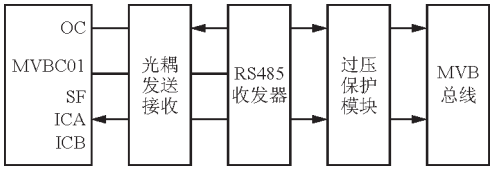

图4　MVBC01接口单元

MVBC01端口ICA（MVB Input Data Channel A）和ICB（MVB Input Data Channel B）分别为MVB输入数据通道A和输入数据通道B，来自物理层收发器的MVB信号可由此端口送入MVBC01中。MVBC01的端口OC（MVB Output Data Channel）是MVB的数据输出端口，数据经此端口可发送至物理层收发器。MVBC端口SF（Send Frame）的输出信号可作为物理层的使能信号，该信号有效时，表示一个报文正在通过MVBC端口OC（MVB Output Data Channel）输出。

本系统CAN侧采用的是三星公司生产的S3C44B0X微控制器，S3C44B0X是韩国三星公司推出的一款基于ARM7TDMI核的16/32位RISC高性价比、低功耗的嵌入式微处理器。该芯片的片上资源非常丰富，因为它本身带有可满足CAN2.0B协议的CAN控制器，故可省去CAN控制器和单片机之间的硬件连接。

CAN收发器采用PHILIPS公司的PCA82C250。主要提供对总线的差动发送能力和对CAN控制器的差动接收能力。当微处理器对CAN控制器进行相应配置后，收发器即可自动完成相应的CAN总线动作。

MVBC01的引脚$\overline{INT}$除在MVBC01复位时作为复位完成信号外，还可作为MVBC01接收到网络数据时的通知信号。它的状态通过s3c44b0x的$\overline{INT1}$传送给S3C44B0X。S3C44B0X轮询检测$\overline{INT1}$，若检测到跳变发生且MVBC01判断此信号为网络控制数据接收事件，则通过I^2C接口读取MVBC01的数据。MVBC01的复位可通过微控制器的PA0端口触发。当MVBC01复位完成后，再通过/INT通知微控制器。MVBC01的控制口配置成串行方式的I^2C接口，S3C44B0X通过该接口对MVBC01的寄存器进行初始化和其它控制操作。其网关硬件连接图如图5所示。

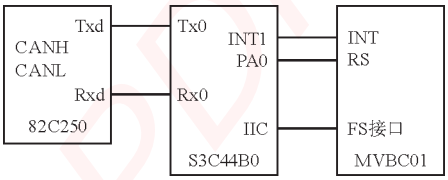

图5　网关硬件连接

4　软件实现

由于MVB-CAN网关两侧连接的是两个现场总线，因此，它们实际的工作就是通过对信息的重新封装以使MVB总线和CAN总线能够读取和传输。

由于CAN数据帧包含8个字节的数据，而MVB从帧最多包含256个字节的数据，因此，从CAN侧接收到的数据要通过MVB传输，就需要按照一定的格式把CAN帧中的数据组合成MVB帧的数据格式；同样，从MVB侧接收到的数据要通过CAN传输，也需要按照一定的格式对MVB帧中的数据进行分帧处理。另外，为了实现透明传输，还需要把MVB侧的过程数据的数据集与CAN侧11位标志符对应起来。

本文采用的方案是建立一种规范的方式来定义MVB端的MVB各种变量的含义，如所在端的端口地址、长度、以及变量在数据集中的偏移量等。而后再将这些信息按照一定的规则和CAN总线侧的帧的标识符等信息对应成一个表格，这样，在CAN侧的数据接收就可以借助这个表格来完成。这样做的优点在于把一些必要的信息（位偏移、长度等）按照约定的形式来实现共享，因而不再需要在数据集中分出空间来传输一些（包括长度和偏移等）信息。本文把表格定义为应用层转换表（inbus-to-outbus或者out-to-inbus）。这里的端口指的是一种既可以被总线访问，又可以

被应用处理器访问的存储器，它在MVB中是用来寻址的，数据的发送是通过应用将数据放入端口，数据的接收则是从端口中取数据放到应用中去。至于端口与端口间的数据拷贝，则是通过主轮询方式来实现的。当主节点轮询到需要发送数据的端口（源端口）时，便会产生从源端口到宿端口（数据接收端口）的数据拷贝。数据集在MVB中其实就是一般的纯数据，之所以称它为集，主要是因为MVB中一帧的数据较长，通常我们把多个变量的信息放在一个集中（一个帧中）来传输，这样做较为经济。

本系统的软件模块主要有CAN模块、MVB控制模块、总任务处理模块、以及数据缓冲模块。其中CAN控制模块主要负责从CAN数据包中解析出完整的CAN协议报文，并存入数据缓冲区2，同时将数据缓冲区A中的CAN数据封装成完整的CAN协议报文后发送到CAN总线上；MVB控制模块主要负责从MVB数据包中解析出完整的MVB协议报文，并将其存入数据缓冲区1，同时再将数据缓冲区B中的MVB数据封装成完整的MVB协议报文后发送到MVB上；总任务处理模块主要起综合调度和监控作用，同时，它还用于整个传输过程中的中断响应；缓冲区模块主要负责将CAN侧和MVB侧接收的数据存储到缓冲区中。

网关程序的整体设计思路是当CAN侧数据发往MVB侧时，首先使CAN侧检测到有数据传过来，总处理模块通过调用CAN模块处理函数来接收数据，并通过对CAN帧的解码来获取CAN标志符，然后调用应用层转换表（inbus-to-outbus）以查询出相应的MVB端口的相关变量，接着再把相应的数据发送到数据缓冲区2中，最后由总任务处理模块通知MVB控制模块将缓冲区2中数据取走，并对数据进行MVB帧格式的编码，再发送到MVB总线上后释放缓冲区。

同样，当MVB侧数据发往CAN侧时，首先将数据发送到MVB上，然后由总任务处理模块检测MVB上是否有数据传输，并通知MVB控制模块接收数据，同时对信息进行解码分析，从中获取端口相关变量，依据端口相关变量查询应用层转换表（out-to-inbus），找出对应CAN标识符，同时将数据发送到数据缓冲区1。此时，总任务处理模块可通过CAN控制模块从缓冲区1中提取数据并进行完整的CAN报文封装，再发送到CAN总线上并释放缓冲区A。其中，MVB侧发往CAN侧的主程序流程如图6所示。

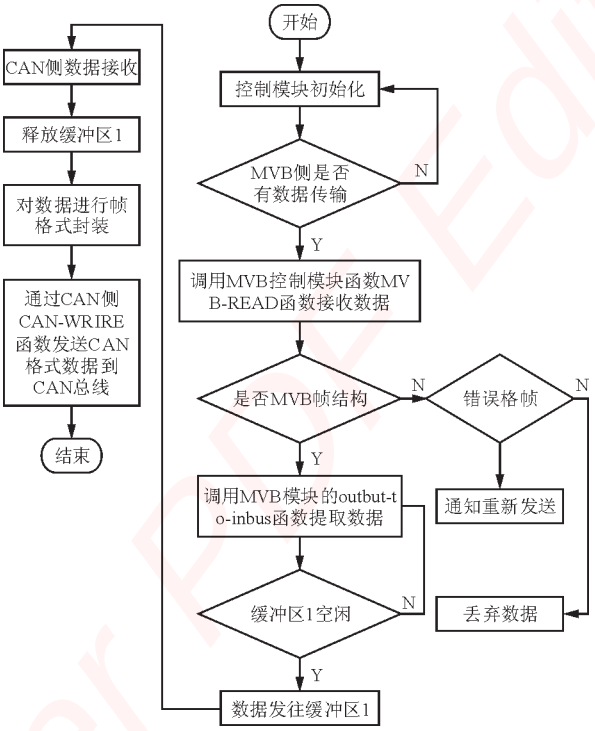

图6　MVB-CAN主程序流程

5　结束语

本文主要介绍了MVB-CAN网关的硬件和软件实现方法。其中，应用层协议的转换是整个网关的主要工作所在，所以，设计时必须建立尽量完善的关系对应列表，同时应提供友好的用户接口函数。

通过MVB-CAN网关的设计可以实现机车内车载设备间不同接口单元之间的通讯，从而为列车通信网络的多元化发展提供有力的支持。

参考文献

[1] 李元熙. 基于ARM7的CAN总线到以太网网关研究与实现[D].南京：南京理工大学，2006.

[2] 陈特放.基于MVB的机车逻辑控制单元[J].计算机测量与控制，2008，16(7)：XX

[3] 阳宪惠.现场总线技术及其应用[M].北京：清华大学出版社，1999.

[4] 张毅坤.单片微型计算机原理及应用[M].西安：西安电子科技大学出版社，1998.

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. 基于磁盘异或引擎的 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

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

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 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

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μC-OSⅡ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：

WeChat ID: kontronn

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 NiosⅡ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

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