

## 基于STM32的GSM模块综合应用

宋维, 崔晓坤

(广东科学技术职业学院, 广东 广州 510640)

**摘要:** GSM模块短信收发系统广泛应用于实时性要求不高的远程监控和测控系统当中。介绍一种GSM模块短信收发采集控制系统,特别是对STM32微处理器的端口配置和GSM模块短信收发系统的实现进行了详细讨论。

**关键词:** STM32; GSM; SMS; 双串口

**中图分类号:** TP311 **文献标识码:** A **文章编号:** 1009-3044(2012)18-4536-02

### Comprehensive Application of STM32 Based on GSM module

SONG Wei, CUI Xiao-kun

(Guangdong Institute of Science And Technology, Guangzhou 510640, China)

**Abstract:** GSM module SMS sending and receiving system applied to remote monitoring and control system where real time performance is not highly required. This paper puts emphasis on a data acquisition and control system which send and receive text messages. And particularly, the realization of STM32 microprocessor port configuration and GSM module SMS sending and receiving mechanism were discussed.

**Key words:** STM32; GSM; SMS; Dual serial communication

GSM模块短信收发系统广泛应用于实时性要求不高的远程监控和测控系统当中。移动通信技术和各种无线局域网技术的发展,给计算机远程数据采集和控制系统提供了更丰富、更合理的数据传输手段,这些手段也逐步渗入到各类系统当中形成各种跨视界跨地域的无线远程采集和控制系统。计算机监控系统的信息传输、远程抄表、电力和油田监测,往往由于环境条件恶劣、距离遥远,不适合进行数据的有线传输。GSM(全球移动通信系统)作为现代无线远程通讯技术的代表已经发展得十分成熟完善,采用GSM网络传输信息价格便宜,技术可行<sup>[1]</sup>。基于GSM的无线远程采集和控制系统比传统有线控制系统具有如下优点:通讯范围广泛、系统经济性较好、总体结构简单等,作为一种廉价、便捷的数据采集和控制方案受到人们的青睐<sup>[2]</sup>。

该文讨论一种GSM模块短信收发采集控制系统,重点探讨微处理器的端口配置和GSM模块短信收发系统的实现。该系统采用基于Cortex M3内核的STM32作为现场监测和控制核心,利用Siemens公司的TC35i模块,通过短消息来实现数据传输。STM32作为32位微处理器,具有主频高,速度快,功耗低等特点,由于价格低廉,代替8位单片机是大势所趋,而且接口丰富,具有良好的通用性和针对各种控制和监测系统的可扩展性。

### 1 系统的功能与结构

GSM模块短信收发采集控制系统是一个可以由用户通过手机短信来进行远程控制的终端模块。用户可以通过中文和英文短信来控制开关量的接通与断开、或查询开关量(I/O口)的开关状态,及其它复杂控制,也可以外接AD口或数字传感器采集系统完成测量、数据采集。操作者可以通过短信指令,按照事先定义好的内容对设备进行控制,也可由现场设备返回监控状态,或发出各类告警信息。除此之外,还可以经过串口连接上位机,通过配套的上位机软件实时监测GSM模块短信收发采集控制系统的状态。

系统由TC35i模块和以STM32为核心的短信收发系统监控单元两部分组成。TC35i模块是由Siemens公司生产的GSM模块,符合ETSI标准GSM0707和GSM0705<sup>[3]</sup>。TC35i有40个引脚,通过一个ZIF连接器引出。这40个引脚按功能可以划分为5类,即电源、数据传输接口、SIM卡接口、音频接口和控制信号部分。用户通过RS232接口,用AT指令对其进行控制<sup>[4]</sup>。

AT指令是主要移动电话生产商共同为GSM研制的一整套指令,其中包括对SMS的控制<sup>[5]</sup>。AT指令是以"AT"为前缀,以回车符<CR>为后缀(即ASCII字符\r\n,对某些模块的操作序列可以只\r为后缀)的ASCII指令序列,用于和GSM模块通讯。

TC35i负责短信的收发。以STM32为核心的短信收发系统则用来读取TC35i收到的短信,并执行相应的测量、数据采集、控制、无线通信功能。两者之间通过RS232串口实现数据交换。GSM无线通信控制终端通过GSM网络和监控中心进行双向的信息传输。系统硬件结构如图1所示。

### 2 系统软件设计

TC35i集成了标准的RS232接口,通过MAX232芯片做电平转换,TC35i即可通过串行口与STM32芯片进行通信。需要注意的是TC35i模块作为数据通信设备(DCE)连接,RxD0引脚为输出脚,TxD0为输入脚。系统与GPRS模块采用串口连接。模块要正常

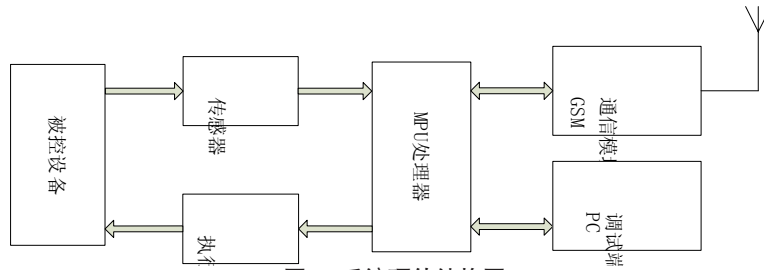


图1 系统硬件结构图

工作,首先要启动模块并对串口进行配置,然后通过串口发 AT 命令联结 GSM 无线网络,并利用 AT 指令集使用 SMS 服务。这些工作通过主控芯片 STM32 完成。

STM32 是意法半导体公司出产的基于专为嵌入式应用设计的 ARM Cortex-M3 内核的嵌入式 MPU,具有高性能、低成本、低功耗的特点。增强型系列时钟频率达到 72MHz,是同类产品中性能最高的产品,功耗达到 36mA,是 32 位市场上功耗最低的产品<sup>[6]</sup>。系统以 STM32F103RBT6 为主控芯片。STM32F103RB 带有 3 个异步/同步串行收发端口,系统中通过 usart2 完成与 GSM 模块的交互,通过 usart1 来与 PC 机串口相连,以使用户对控制终端进行自定义或通过 PC 机实时监测 GSM 模块短信收发运行的状态。在调试升级阶段可以利用双串口完成主程序的调试。

STM32 串行通信处理有轮询和中断两种方式。端口可以重映射,这里不进行重映射,即设置 PA9,PA10 为 USART1 的 Tx/Rx 管脚,PA2,PA3 为 USART2 的 Tx/Rx 管脚。系统的串口初始化过程为:

1. 首先初始化串行通信端口 GPIO
2. 串口外设时钟使能
3. 填充 USART\_InitStructure,设置波特率,数据位,校验位,停止位,初始化串口。
4. 串口中断使能。
5. 串口 1 使能
6. 串口 2 使能
7. 串口中断 NVIC 优先级设定。

GSM 模块短信收发和运行状态的监测通过 usart 的中断处理程序实现。

在 USART1\_IRQHandler()中,PC 机软件每发送一条指令,将同时发给 uart1 和 uart2,发给 uart1 用于回显,发给 uart2 用来指令操控,如下所示:

```

USART_SendData(USART1,chRecv);
while(USART_GetFlagStatus(USART1,
USART_FLAG_TXE) == RESET);
USART_SendData(USART2,chRecv);
while(USART_GetFlagStatus(USART2,
USART_FLAG_TXE) == RESET);
    
```

同样,在 USART1\_IRQHandler()中,从 uart2 接收到 GSM 模块的应答,保存为返回结果字符串的同时,直接转发到 usart1 上,以便从 PC 机监视模块返回值。

### 3 AT 指令的处理

STM32 通过串口收发 AT 指令和获取 PDU 协议单元字段来完成与 GSM 模块交互。GSM 模块负责短信与服务中心间的收发。

如果 AT 被正确执行,则 GSM 模块返回以 OK 结尾的字符串,如果未被正确执行,返回以 ERROR 结尾的字符串。程序中根据这点来判断 GSM 模块有无正确返回。除此之外,还有第三种情况,即 GSM 模块无响应,这时没有任何返回值,对于这种情况可以通过设置超时来进行重发处理。这要求以上的串口初始化过程后,同时也要初始化定时器或 RTC 实时时钟,以便产生一个定时的基准。

用于短信收发和相关程序设计的常用 AT 指令有:

AT+IPR? <CR>;用于查询模块与主控 MPU 通讯的波特率,同样该指令也可完成波特率的设定。

AT+CMGF=[<mode>] <CR>;用于设置短信的模式:< mode >为 1,短信为文本模式;< mode >为 0,短信为 PDU 模式。文本模式用来发送英文和数字,PDU 模式则还可以用来发送中文。本系统收发的是中文短信,因此采用 PDU 模式。

AT+CMGR=<seiral><CR>;读取短信,< seiral >为短信的位置序号。

AT+CMGD=< seiral ><CR>;删除短信,< seiral >为短信的位置序号。

AT+CMGS=<LENGTH><CR><发送内容><ctrl-Z>;采用 PDU 模式时发送中文内容短信<sup>[7]</sup>。

系统对于中文短信的发送是通过 AT+CMGS 指令完成的。在发送之前先要按照 PDU 编码的格式构造中文短信。由于 STM32 编译器本身对中文处理为 gb2312 代码,而 PDU 协议编码对中文是采用 UTF-8 的编码,因此中文短信要通过 GSM 网络传输,传输之前应该将 GB2312 转换为 unicode 编码。这可以通过查表方式完成,即设置一个 gb2312 编码和相应 UTF-8 编码的结构体数组,形成汉字编码表。结构体的第一个元素为汉字的 gb2312 编码,这样可以用二分法快速数据查找算法查找到该汉字在 gb2312 编码表数组中的位置,该位置对应的结构体第二个元素就是对应的 unicode 编码。如果要将 PDU 编码中的汉字读出,则需要另外一个 UTF-8 到 gb2312 的编码表,同样可以通过二分法快速得到 GSM 短信内容的 gb2312 编码。

云计算平台中的资源利用率可达80%以上,与传统数据中心的资源利用率相比有大幅提升。

用户不再需要进行巨大的一次性IT投资,彻底省去了购置、安装、管理软硬件的费用,因为他们可以从云计算提供商那里租用这些IT基础设施;使用这些IT资源时,可以按照自己的实际使用量付费。

对应用管理的动态、高效率、自动化是云计算的核心。它要保证用户在创建一个服务的时候,能够用最少的操作和极短的时间就完成资源分配、服务配置、服务上线和服务激活等一系列操作。于此类似,当用户需要停用一个服务的时候,云计算能够自动完成服务停止、服务下线、删除服务配置和资源回收等操作。在虚拟化技术的支持下,Web应用可以被做成虚拟器件,当需要启动服务的时候,被快速部署到云计算环境中;当服务不再需要时,可以取消部署以释放占用的资源。可见云计算可以在软件和解决方案等不同层次提供极大的灵活性与自动化。

作为一种革命性的信息产业浪潮,云计算能够形成新的业务价值链,促进跨领域的创新协作,从而产生更高的价值。在以云计算推动的新价值链产生过程中,能够创造更多的就业机会,产生更多新兴服务,建立新兴产业。

### 3 云计算产生的原动力

云计算带来的是IT产业的转型和升级。不仅各个微观经济实体成为了云计算产业链中的参与者,各国政府也同样重视这一产业的重要变革。毕竟,就如同制造业的变革导致了全球范围的重新分工,云计算的出现也将引发IT产业在世界范围内的再分工。世界各国,尤其是新兴发展中国家不应错过这个难得的机遇以实现自己产业结构的升级。各国政府对于高科技产业的重视程度和投入力度是推动云计算向前发展的重要动力。

在计算层面。云计算之所以在今天产生,是六方面原动力共同作用的结果。

- 1)芯片和硬件技术的飞速发展,使得硬件能力激增、成本大幅下降。让独立运作的公司集中可观的硬件能力实现规模效益成为可能。
- 2)虚拟化技术的成熟,使得这些硬件资源可以被有效地细粒度分割和管理,以服务的形式提供硬件和软件资源成为可能。
- 3)面向服务架构的广泛应用,使得开放式的数据模型和通信标准越来越广泛地为人们使用,为云中资源与服务的组织方式提供了可行的方案。
- 4)软件即服务模式流行,云计算以服务的形式向最终用户交付应用的模式被越来越多的用户所接受。
- 5)互联网技术的发展,让网络的带宽和可靠性都有了质的提高,使得云计算通过互联网为用户提供服务成为可能。
- 6)Web2.0技术的流行和广泛接受,改变了人们使用互联网的方式,通过创新的用户体验为云计算培养了使用群。

### 4 云计算带来的变革

反观近百年来计算技术的发展,我们不难从中发现与社会经济相似的规律。在早期的大型机时代,公司或机构自己购买、营运并使用计算设备。随着半导体、网络和软件技术的综合发展,计算技术进入个人电脑时代。互联网上出现了多种多样的服务提供商,崭新的业务模式层出不穷。在这个过程中,提供商通过制定标准来巩固自己的专业地位与业务规模。这样的业务化与规模化不断深入,伴随着技术的成熟,IT服务将逐渐变得如水和电一样,可以通过无处不在的互联网随处获得。这就是云计算,顺应着历史发展的脉络在我们这个时代诞生。当下,仅是一个开始,随着IT生态系统的进一步精细分工,云计算将孕育出新兴的产业链。新的参与者将加入,旧的参与者必须相机而变。

云计算作为一种新兴的IT运用模式,带来了IT产业调整和升级,同时也催生了一条全新的产业链。这条产业链中主要由硬件供应商、基础软件提供商、云提供商、云服务提供商、应用提供商、企业机构用户和个人用户等不同角色组成。

#### 参考文献:

- [1] 杨正洪,周发武.云计算和物联网[M].北京:清华大学出版社,2011.
- [2] 蔺华,杨东日,刘龙庚.大师访谈-云计算[M].北京:电子工业出版社,2011.
- [3] 『虚拟化与云计算』小组.云计算宝典[M].北京:电子工业出版社,2011.

---

### 4 结束语

基于STM32的GSM模块短信收发采集控制系统,适用范围广,适合在野外不方便到达的地方或车间等不易操作的地方使用;维护费用极低;可靠性好。它的双串口功能使特定时段用户通过PC上位机调试和监控操作变得更加直观、方便。不足的地方是它对用户的短信指令响应时间确定性不高,不适用于高实时性的场合。

#### 参考文献:

- [1] 刘盛平,韦云隆,杨飞.基于移动短信技术的无线传感器网络系统[J].微计算机信息,2007,23(2-1):158-160.
- [2] 闫正龙,陈正江,黄强,等.基于GIS/GPS/GSM/GPRS技术的车辆监控系统的设计与实现[J].西北大学学报:自然科学版,2008,38(1):127-130.
- [3] 吕芳,胡林静,臧琛.基于TC35 GSM的汽车防盗报警系统的软件设计[J].测控技术,2009,28(z1):125-127.
- [6] STM32F103RB Datasheet[EB/OL].<http://www.st.com/internet/mcu/product/164487>,2012.
- [7] 周明刚,王元芳,宋庆美,周敬东,等.用AT指令实现的用电欠费短信通知系统[J].现代计算机,2009(6).

# 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](#)

## 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](#)
27. [基于 XPEmbedded 的警务区 SMS 指挥平台的设计与实现](#)



## 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](#)

## 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](#)

## 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](#)

RT Embedded <http://www.kontronn.com>

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](#)