

T9 输入法在 Tilcon 下的实现

毛晓梅 邹璞玉 周 群 (江苏自动化研究所计算机事业部技术开发部,江苏 连云港 222006)

摘 要

T9 输入法是小型掌上设备普遍采用的输入法,Tilcon 是一种嵌入式系统下图形界面开发工具。简要介绍了 T9 输入法和 Tilcon 的基本原理和特点,分析了 T9 在 Tilcon 下的设计原理和实现方法,结果表明该方法能满足实际的应用需求。

关键词:T9,Tilcon,嵌入式系统

Abstract

T9 is the input method which is used generally for the small handheld wquipment.Tilcon is a graphical interface development solution that interactive user interfaces foe embedded systems.This paper introduces the basic principles and characteristics of T9 and Tilcon firstly,and analyzes the design theory and realization of T9 under Tilcon.

Keywords:T9,Tilcon,embedded systems

Tilcon 是一个支持多种操作系统平台的图形界面开发工具。T9 输入法,几乎所有的手机都是使用该输入法,它解决了小型掌上设备的文字输入问题,已经成为全球手机文字输入的标准之一。在某项目上,有一个特定应用,需要使用 T9 输入法,由于我们使用的操作系统是 VxWorks5.5,在 VxWorks 下使用原有的工具进行图形界面开发比较困难,因此就利用 Tilcon 方便、快捷地进行图形界面设计,从而实现 T9 输入法。

1 Tilcon 概述

Tilcon 是目前先进的实时操作系统下图形开发工具。同步支持最新版本的 Tornado/VxWorks 以及 WindML 多媒体库。适用于 VxWorks 实时环境下构建虚拟仪表,实时控制,分布式控制等高级图形应用。Tilcon 本身已经集成大量成熟的控件,用户无需再调用低级的图形函数去画线填充,可以像 Windows 下 VB 可视化编程一样用拖动控件的方式构造自己的图形应用。最大的简化了用户的开发过程。Tilcon 采用了先进的图形技术,用户开发的图形界面不会有任何用户自己编写的程序与图形系统混合。图形界面的开发质量,绝不会影响系统的安全性和稳定性。彻底解决了困扰嵌入式实时图形领域应用的难题,具有极高的可靠性和可维护性。Tilcon 开发的图形界面同时支持本地和分布式显示。图形界面开发完成后就可以自动的显示在本地或通过 TCP/IP 网络显示到其他网络平台上。

Tilcon 的体系结构如图 1 所示,主要包括三个组成部分:Tilcon EVE(Embedded Vector Engine),独立的 API 开发平台和可视化设计工具,Tilcon 建立界面。EVE 是一个比较小的,

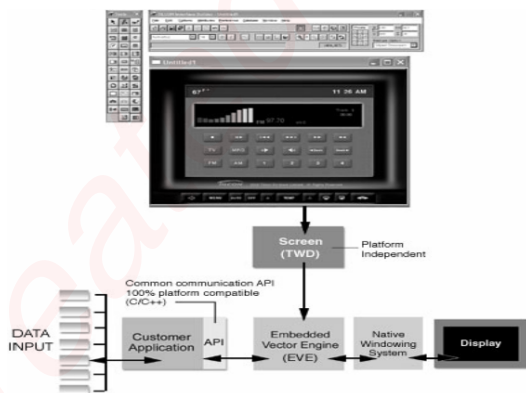


图 1 Tilcon 体系结构示意图

可以呈现用户图形界面的事件驱动型的内核,它是一个与用户事件、用户编写的程序分开的进程,这种分离的特性使得用户的应用程序便于移植、修改和维护。第二个组成部分是一个独立的 API 调用平台,为应用程序和 EVE 提供通讯、动态创建对象功能以及对象属性的总体控制功能,但仍比较简单并易与应用。Tilcon 建立界面部件可以让你只需要通过鼠标托拽即可完成复杂的界面设计,界面上的各个对象的外观和属性都可以根据用户的需求来设定。

2 T9 输入法

现在手机最常用的输入法就是 T9 输入法,T9 可以以直观、简捷的方法快速输入简短文字。通常键盘上有九个常用的数字键,每个数字按键代表多个字母,如图 2 所示,例如数字键“5”可以代表 J、K 或 L 三个字母。一般的输入法可能按数字键 5,一次选择的是 J,两次是 K,三次是 L,在拼音输入的方式下,需要选择需要的字母来组成拼音,则就需要按键很多次才能得到所需要的拼音,而 T9 输入法只需要输入该字母所在的按键一次,程序按算法自动组成合理的拼音,这样就大大减少了输入时按键的次数,这就是 T9 能够大大提高输入效率的根本原因。T9 的中文输入包括简体中文和繁体中文,输入方式包括笔划、拼音以及注音。在 T9 中使用频率最高的就是拼音和笔划输入了。拼音输入是用户们最为熟悉的方式,T9 会从内置的语言数据库中选出匹配这些拼音的汉字。



图 2 T9 键盘示意图

3 T9 在 Tilcon 下的实现

在 Tilcon 下使用 T9 输入法,可分为两部分来设计,一是 T9 算法,二是 Tilcon 下显示 T9 输入法的结果,也就是相当于手机屏幕的效用。

3.1 T9 算法

本文的 T9 算法以拼音、小写英文、大写英文以及数字这四种方式为例,这四种方式之间可相互切换。算法的主体结构如图 3 所示,这个算法的结构是一个循环的结构,首先获得一个输入字符,然后判断是在哪种输入方式下,再分别调用相应的输入法。

在上述四种方式中,拼音的方式也就是中文汉字输入方式比较复杂,在输入过程中先得到能组合的拼音,再根据该拼音在汉字库中找到对应的汉字。在算法中用到的相关数据结构如下:
`unsigned char PY_mb_a[255]=`

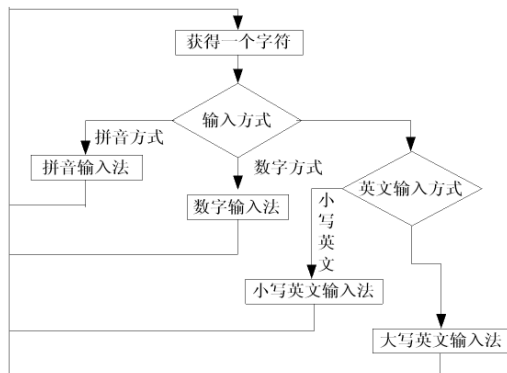


图3 T9 算法主体结构示意图

```

{{"啊啊吡铜嘎呵腌"},
{"哎哀埃埃埃皓矮矮蒿艾爱隘碍"},
...}/* 汉字库 */

unsigned char PY_index_a[][13] = { {0x20,0x20,0x20,0x20,
0x20,0x20,0x00,0x00,0x00,0x00,0x01,0x00,0x00},
{'i',0x20,0x20,0x20,0x20,0x20,0x01,0x00,0x00,0x00,0x02,0x01,
0x00},
{'n',0x20,0x20,0x20,0x20,0x20,0x02,0x00,0x00,0x00,0x02,0x02,
0x00},
{'n','g',0x20,0x20,0x20,0x20,0x01,0x01,0x00,0x00,0x03,0x03,0x00},
{'o',0x20,0x20,0x20,0x20,0x20,0x01,0x00,0x00,0x00,0x02,0x04,
0x00}
};/* 字母 a 能组合的拼音 */
    
```

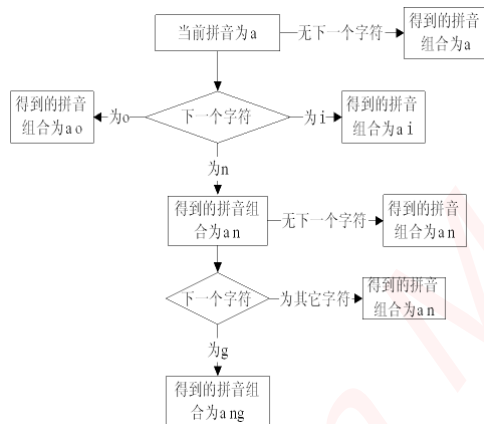


图4 拼音方式下查找过程示意图

如果第一个输入字符是字母 a，根据下一个字符，在 PY_index_a[][13]中查找是否有与该字母的组合，查找过程如图 4 所示。

3.2 Tilcon 下的实现

3.2.1 Tilcon 与 T9 的交互机制

T9 输入法的

结果要用 Tilcon 的图形界面呈现出来，就需要一种机制在 T9 和 Tilcon 之间进行消息交互。Tilcon 的 API 函数提供了一种消息通道的交互方式，如图 5 所示，在 T9 输入法与 Tilcon

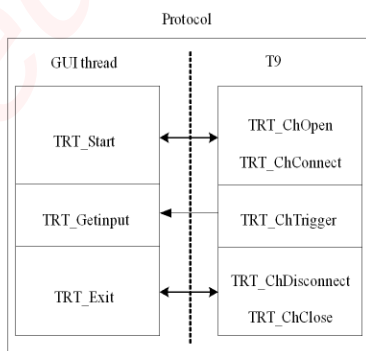


图5 Tilcon 与 T9 的交互过程示意图

之间建立了一个消息通道。首先在 Tilcon 进程中 TRT_ChOpen() 建立一个通道，然后 TRT_ChConnect() 为这个通道建立连接，再在 T9 中 TRT_ChTrigger() 发消息给 Tilcon 通知它接收输入的结果，并处理相应的事件，最后 TRT_ChDisconnect() 断开连接，TRT_ChClose() 关闭该通道。

3.2.2 Tilcon 下的界面设计

一个简单的输入事件整个过程就是先输入一个字符，然后在相应的文本框中显示汉字、大小写英文或数字。因此在界面上就应有一个标识是处在何种方式下进行的输入事件的一个对话框，还应有一些可以输入字符的文本框，如图 6 所示就是 Tilcon 下所设计的拼音输入方式下的对话框的界面。



图6 拼音输入法示意图

启动 T9 任务后，当从 T9 键盘上敲入数字键“1”，设定输入方式为拼音方式，获取 T9 算法的结果，doCNStrDraw() 首先在 Listbox0 中显示得到的拼音组合，即拼音 a、b、c，选中拼音 a 后，在 Listbox1 中显示选中的拼音 a 下的汉字，也就是“啊啊吡...”，选中汉字“阿”后，Accert_text() 可将该选中的汉字写入文本框；Move_fun_l_r() 可以在 Listbox1 中左右移动来选择想要的汉字；Move_ud() 可以在 Listbox1 中上下翻页；Changedisplay() 切换输入方式，即把拼音方式切换为小写英文、大写英文或数字方式；Cancel_text() 可删除所得到拼音组合和文本框中的内容。Tilcon 下显示界面的主要代码结构如下：

```

void EventProcess()
{
    while( EventGet( ))/* 获取事件 */
    {
        switch(tfFlag)
        {
            case 0 : doCNStrDraw();
            /* 在界面上显示汉字 */
            case 1 : doEnglishStrDraw();
            /* 在界面上显示大小写英文 */
            case 2: doDigitalStrDraw();
            /* 在界面上显示数字 */
            case 3: Changedisplay();
            /* 输入方式切换 */
            case 4: Accert_text();/* 确认内容 */
            case 5: Move_fun_l_r();/* 左右键 */
            case 6: Move_ud();/* 上下页 */
            case 7: Cancel_text();/* 删除键 */
        }
    }
}
    
```

Tilcon 在实现的过程中截获了所有的输入事件，在实现的 T9 算法的过程中，先于 Tilcon 获取了键盘的输入事件，因此在 T9 运行的过程中，Tilcon 中的键盘输入就会暂时失去效用，所以需要把截获的输入事件还给 Tilcon，我们利用一个热键在 T9 与 Tilcon 之间进行切换，当敲入该热键后就会从 T9 中回到 Tilcon 中，或从 Tilcon 到 T9 中，也就在 Tilcon 下顺利实现了 T9 输入法。

参考文献

- [1]Tilcon Software Ltd,Tilcon Software White Paper[M],2005
- [2]Tilcon Software Ltd,Tilcon Evaluation Guide[M],2005
- [3]Tilcon Software Ltd,Tilcon User Guide[M],2004

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](#)

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](#)

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](#)

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](#)

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](#)

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](#)

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](#)

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](#)