

基于 Tilcon 的指控系统多任务人机交互软件设计

朱 伟, 许春雷, 孔 军

(江苏自动化研究所, 江苏 连云港 222006)

摘 要: Tilcon是军事特别是指挥控制领域广泛使用的人机交互界面开发工具。简单介绍了Tilcon的组成和人机交互处理流程, 分析了应用Tilcon处理多任务人机交互事件的不足, 设计了一个基于Tilcon的指控系统多任务人机交互软件架构, 并给出了具体实现方法。工程应用表明, 该人机交互软件体系结构清晰, 具有较好的健壮性、开放性和兼容性, 可显著提高指控系统人机交互处理的实时性。

关键词: 指挥控制系统; 人机交互; Tilcon; 体系结构

中图分类号: TP39

文献标识码: A

DOI: 10.3969/j.issn.1673-3819.2011.04.026

Design of Software of Command and Control System with Multitask and Man-machine Interaction Based on Tilcon

ZHU Wei, XU Chu-lei, KONG Jun

(Jiangsu Automation Research Institute, Lianyungang 222006, China)

Abstract: Tilcon is the main development tool of man-machine interaction of command and control system. Tilcon's composition and method of Man-machine Interaction are simply introduced. After analyzing the shortage of handling multitask and man-machine interaction with Tilcon, a multitask man-machine interaction software architecture of command and control system and a realizing method based on Tilcon are lay out. According to practical application, the architecture is legible, with preferable robustness, open and compatibility. It can obviously improve real-time performance of command and control system's Man-machine Interaction.

Key words: command and control system; man-machine interaction; Tilcon; architecture

指挥控制系统作为作战系统的核心子系统, 是作战信息处理中心和指挥员实施指挥的直接部位, 通常需要同时完成情报综合处理、作战辅助决策、武器综合控制等多个独立功能^[1]。实时多任务 VxWorks 操作系统以其卓越的多任务事务处理能力已成为军事指控控制领域中使用最广泛的系统。近年来, 由于指挥控制系统功能日趋复杂, 对人机交互界面的实时性、灵活性、扩展性等提出了更高的要求。另一方面, 所见即所得的 Tilcon 人机交互界面开发工具为嵌入式系统下的人机交互界面开发提供了简便快捷的界面设计方式^[2], 但也存在着人机交互处理集中、任务之间耦合性强, 无法兼容其他人机交互方式、实时性难以保证等不足。如何应用 Tilcon 设计多任务人机交互软件, 对于确保指控系统的实时性、扩展性, 提高任务的可靠性、可移植性, 促进指控系统下功能任务的构件化设计均具有重要意义。

1 Tilcon人机交互界面开发工具

1.1 Tilcon

Tilcon是wind River推出的多平台用户界面开发工具, 具有功能齐全、界面美观、使用方便的特点,

支持Windows、VxWorks、Linux、QNX 等多种平台。Tilcon人机交互界面开发环境提供了常用界面控件, 用户可以直接拖放控件到窗口上构造自己的图形、表页面界面, 实现了所见即所得的人机界面开发方式^[3], 已成为嵌入式系统下主流的人机交互界面开发工具。

1.2 Tilcon 组成

Tilcon 主要由三个组件构成^[4], 即人机交互界面运行引擎 EVE、应用程序 API 接口函数和可视化人机交互界面开发环境, 其组成及关系如图 1 所示^{[5][6]}。

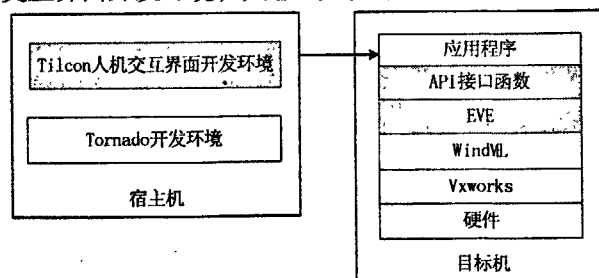


图1 Tilcon构成及关系图

运行在嵌入式系统之上的EVE 是一个独立于用户应用程序的微内核, 以两种方式管理所有的用户界面显示及人机交互事件: 1) 执行应用程序的API 命令进行屏幕绘制; 2) 以事件、回调函数等形式处理操控消息, 驱动应用程序运行。

API 是供用户应用程序调用的一组基本函数, 为EVE和用户程序之间提供交互接口, 用户可以通过它动态地创建对象、修改对象属性。

Tilcon 可视化界面开发环境集成了大量常用控

收稿日期: 2011-04-28

修回日期: 2011-05-10

作者简介: 朱 伟 (1973 -), 男, 安徽淮北人, 高级工程师, 研究方向为作战指挥控制系统软件设计。
许春雷 (1964 -), 男, 研究员, 硕士生导师。
孔 军 (1968 -), 男, 研究员, 硕士生导师。

件, 用户通过拖拉控件来设计界面, tilcon界面采用平台无关的二进制资源形式存储, 具有较好的移植性。

1.3 Tilcon 人机交互事件处理

Tilcon 通过一个循环任务来处理人机交互事件。任务初始化需要先后完成: 1) 调用 TRT_Start 接口函数来启动 tilcon 图形引擎; 2) 调用 TRT_WindowLoad 函数来加载 twd 界面资源文件; 3) 调用 TRT_WindowDisplay 函数人机交互图形界面。初始化完成后调用 TRT_GetInput 函数建立主框架循环。TRT_GetInput 函数处于等待状态并实时接收、处理来自于 Tilcon 引擎所传递的界面操控事件。Tilcon 人机交互事件处理流程如图 2 所示^[7]。

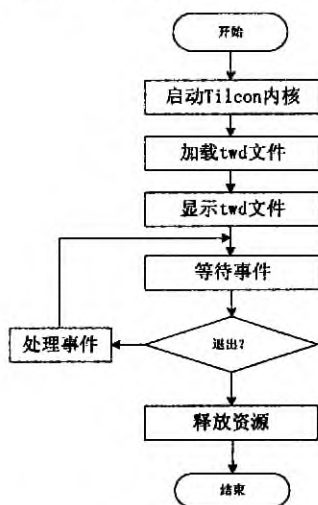


图2 Tilcon人机交互事件处理流程

1.4 Tilcon 实现多任务人机交互处理的局限性

从Tilcon的人机交互事件处理方式来看, 所有Tilcon的控件交互事件在一个循环任务中来完成, 无论系统中有多少个任务、多少窗口和控件。Tilcon人机交互处理方式如图3所示。如果基于Tilcon建立多任务系统, 每个任务多个窗口, 每个窗口包含多个控件的话, 具有以下明显不足:

1) 体系结构不好, 由于所有任务的人机交互处理在Tilcon主任务中完成, 功能任务之间耦合紧密, 不利于任务的独立设计、维护和移植。

2) Tilcon人机交互处理任务软件规模将过于庞大, 人机交互事件的处理过于复杂。

3) 由于Tilcon采用循环方式来处理人机交互事件, 所以从本质上来说Tilcon串行处理所有的人机交互事件。如果前面的人机交互处理时间较长, 则直接影响到后续人机交互响应的实时性。

4) 在Tilcon人机交互处理任务中完成各个任务的人机交互处理, 如图3所示。人机交互处理在Tilcon任务中, 其他功能模块在任务中, 不利于任务的模块化、构件化设计, 不便于系统的功能扩展和伸缩配置。

指控系统显控设备一般都配置专用功能键、非标准数字小键盘、专用触摸屏/触摸板等人机交互设备, 作为窗口控件操作的快捷或备用输入手段。但由于Tilcon主任务无法处理这些设备操控事件, 需重复配置这些操控事件的处理模块, 带来了系统功能代码的重复。

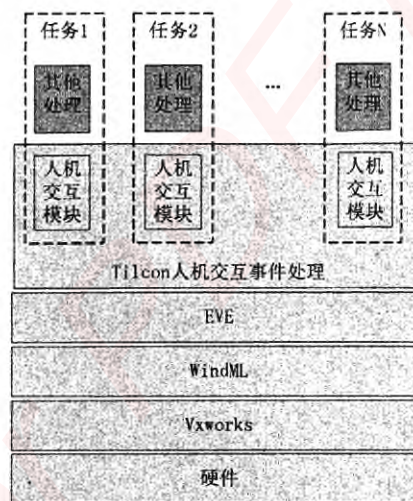


图3 Tilcon多任务人机交互处理示意图

2 通用的多任务人机交互软件架构设计

软件架构反映了软件系统构成要素、构成要素之间的相互关系。好的软件架构应用以下几个基本特征^[8]:

1) 结构清晰、易于理解。软件架构具有层次性, 不同层次组件具有依赖关系, 同层组件关系简单, 耦合松散。

2) 易于维护、升级。同层组件之间的松耦合特性使得应用需求发生变化、需要升级或增加新的组件时, 不会对整个架构和其他组件带来影响。

3) 具有较好的移植性。层次化软件架构带来了软件较好的适应性。软件从一个平台移植到另一个平台时, 只需把平台紧密相关的层次升级或替换即可, 不会影响业务处理层次的组件。

在对指控系统任务和人机交互事件进行深入分析基础上, 采用层次化、组件化设计思想, 建立了基于Tilcon的多输入、多任务人机交互处理模型, 如图4所示。

该模型最上层为功能任务层。任务功能完整、可独立设计、开发和维护, 便于采用构件化思想进行开发。各任务可通过信号量、事件或消息通信等方式响应人机交互通用中间件的人机交互输入命令。

人机交互通用中间件位于功能任务和交互设备预处理中间层, 响应Tilcon、数字小键盘、触摸屏等人机交互输入事件, 并规范化交互方式, 匹配人机交互

互输入和相应的任务，启动任务进行处理人机交互事件。中间件和功能任务之间采用规范化交互方式，可以屏蔽底层人机交互输入的差异性，方便系统功能的扩展和功能任务的移植，具有较好的开放性和兼容性。



图4 基于Tilcon的多输入、多任务人机交互处理模型

层次化的人机交互模型把人机交互处理功能模块配置在对应的任务中，输入设备预处理和人机交互中间件只是简单处理、分发人机交互信息，整个系统结构清晰，人机交互效率较高，可有效保证系统人机交互的实时性。

3 基于 Tilcon 的人机交互设计实例

不失一般性，利用信号量同步机制^[9]，给出了基于Tilcon的人机交互设计实例。信号量机制不具备参数传递能力，所以设计了一个规范化的信息交互数据结构，实现人机交互中间件和功能任务之间的事件信息的交互。信息交互数据结构C语言定义示例如下：

```
typedef struct
{
    unsigned short    code ; // 控件类型
    unsigned short    state ; // 状态或值
    unsigned short    event ; // 事件类型
    unsigned char     *id ; // 控件ID
    unsigned char     *txt ; // 输入字符串
    unsigned char     *frame_window ; // 控件所属窗口id
}I_Rec_Data_Str;
```

为规范信息交互结构中各参数含义，可以使用宏方式定义，在人机交互中间件和各服务模块中包含此宏定义。如果参数含义改变，只需要直接改变宏定义即可。

Tilcon任务（或其他输入设备）预处理完毕后，

启动人机交互中间件，人机交互中间件完成交互信息的规范化后，释放信号量启动功能任务处理模块：

```
For(;;)
{
    等待信息预处理结果；
    信息交互结构赋值；
    释放输入信号量Input_SEM_ID;
}
```

功能任务人机交互处理模块中应用接收信号量，读入人机交互信息，处理人机交互信息的流程：

```
For(;;)
{
    等待输入信号量Input_SEM_ID;
    读入人机交互信息；
    处理人机交互信息；
}
```

4 结束语

Tilcon 是嵌入式系统下主流的人机交互界面设计工具，由于 Tilcon 在主任务中完成人机交互事件的处理，不适于实现多输入、多任务人机交互界面的设计。通用多输入、多任务人机交互处理模型降低了功能任务和 Tilcon 处理任务的耦合度，提高了系统的结构化水平和人机交互实时性，有利于实现功能任务的构件化设计，下一步有待开展指控系统下功能任务的构件化实现研究。

参考文献：

- [1] 赵祖铭.潜艇指控系统技术和发展分析[J].指挥控制与仿真,2007,29(2):1-8.
- [2] 侯小鹏,谭显春,秦克.基于Tilcon 的交互式多页面的设计[J].微计算机信息,2010,26(52):84-85.
- [3] 张加林,李松,汪高武,等.基于Tilcon的VxWorks图形界面开发技术[J].指挥控制与仿真,2007,29(5):100-103.
- [4] 刘炳锋,李江红,等.基于Tilcon的某武器显控系统界面设计[J].计算机测量与控制,2008,16(8):1153-1154.
- [5] Wind River Systems Inc. WindML DDK 3.0 PROGRAMMER'S GUIDE[R].2002.
- [6] Tilcon SoftWare White Paper[Z]. Canada: Tilcon Software Ltd,2003.
- [7] 韩改宁,李永锋.基于嵌入式的Tilcon 用户图形界面设计与开发[J].现代电子技术,2009,16(3):51-53.
- [8] 祖兆研.基于插件技术的软件架构设计及应用[D].南京:河海大学硕士学位论文,2007.
- [9] 孔祥营,柏桂枝.嵌入式实时操作系统 VxWork 及其开发环境[M].北京:中国电力出版社,2002.

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](#)

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](#)

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 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](#)

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.

