

基于 VxWorks系统的 MiniGUI 图形界面开发

明芳¹⁾ 徐凌²⁾ 王刚¹⁾

(中船重工第 709 研究所系统软件部¹⁾ 武汉 430074) (武汉理工大学²⁾ 武汉 430070)

摘要 MiniGUI 是一个非常适合于实时嵌入式产品的高效、可靠、可定制、小巧灵活的图形用户界面支持系统,能支持多种不同的嵌入式操作系统,包括 Linux、VxWorks 以及 Win32 等平台,概述 MiniGUI 的技术特点和优势,介绍基于 MiniGUI 嵌入式系统的软件体系结构,MiniGUI 在 VxWorks 下的程序开发方法。

关键词 MiniGUI I 图像用户界面 VxWorks

中图分类号 TP391.41

1 引言

MiniGUI 是根据嵌入式系统应用特点量身定做的图形支持系统。它源自一个自由软件项目,现由北京飞漫软件技术有限公司维护并开展后续开发。MiniGUI 项目的最初目标是为基于 Linux 的实时嵌入式系统提供一个轻量级的图形用户界面。经过近 8 年的开发和应用,已非常成熟和稳定,广泛应用于手持信息终端、机顶盒、工业控制系统及工业仪表、彩票机、金融终端等产品和领域。

2 MiniGUI I 的特点及应用

MiniGUI I 为实时嵌入式操作系统提供了非常完善的图形及图形用户界面支持。MiniGUI I 本身的移植性设计,使得不论在哪个硬件平台、哪种操作系统上运行,MiniGUI I 均能为上层应用程序提供一致的应用程序编程接口 (API)。

2.1 MiniGUI I 的功能特色

作为操作系统和应用程序之间的中间件,MiniGUI I 将底层操作系统及硬件平台差别隐藏起来,并对上层应用程序提供一致的功能特性,这些功能特性包括:

(1) 完备的多窗口机制和消息传递机制。

(2) 常用的控件类,包括静态文本框、按钮、单行和多行编辑框、列表框、组合框、进度条、属性页、工具栏、拖动条、树型控件、月历控件等。

(3) 对话框和消息框支持以及其它 GUI 元素,包括菜单、加速键、插入符、定时器等。

界面皮肤支持。用户可通过皮肤支持获得外

观非常华丽的图形界面。

(4) 各种流行图像文件的支持,包括 JPEG、GIF、PNG、TGA、BMP 等等。

(5) 多字符集、多字体和简体中文 (GB2312) 输入法支持。

2.2 MiniGUI I 的技术优势

和其它针对嵌入式产品的图形系统相比,MiniGUI I 具有如下几大技术优势:

1.2.1 轻型、占用资源少

MiniGUI I 本身的占用空间非常小。MiniGUI I 的典型存储空间占用情况如下:

· MiniGUI I 支持库: 500K ~ 700K (由编译选项确定)

· MiniGUI I 字体、位图等资源: 400K (由应用程序确定,可缩小到 200K 以内)

· GB2312 输入法码表: 200K (不是必需的,由应用程序确定)

· 应用程序: 1M ~ 2M (由应用程序决定)

1.2.2 高性能、高可靠性

MiniGUI I 良好的体系结构及优化的图形接口,可确保最快的图形绘制速度。针对多窗口环境下的图形绘制开展了大量的研究及开发,优化了 MiniGUI I 的图形绘制性能及资源占有。MiniGUI I 在工业控制系统的大量应用,证明 MiniGUI I 具有非常好的性能。

1.2.3 可配置

为满足嵌入式系统千变万化的需求,必须要求

GUI系统是可配置的。MiniGUI I实现了大量的编译配置选项,通过这些选项可指定 MiniGUI I库中包括哪些功能而同时不包括哪些功能。大体说来,可以在如下几个方面对 MiniGUI I进行定制配置:

- MiniGUI I要运行的操作系统,是普通嵌入式 Linux、eCos还是 VxWorks
- 生成基于线程的 MiniGUI I- Threads运行模式还是基于进程的 MiniGUI I- Lite运行模式,或者只是最简单的 MiniGUI I- Standalone运行模式。
- 需要支持的 GAL(显示)引擎和 IAL(输入)引擎,以及引擎相关选项。
- 需要支持的字体类型、字符集、图像文件格式、控件类、控件的整体风格等。

1.2.4 跨操作系统支持

理论上,MiniGUI I可支持任意一个多任务嵌入式操作系统;实际已支持 Linux/uClinux、eCos、uC/OS-II、VxWorks等嵌入式操作系统。同时,在不同操作系统上的 MiniGUI I,提供完全兼容的 API接口。

3 基于 MiniGUI I的嵌入式系统软件结构

MiniGUI I具有良好的软件架构,通过抽象层将 MiniGUI I上层和底层操作系统隔离开来。如图 1 所示,基于 MiniGUI I的应用程序一般通过 ANSIC 库以及 MiniGUI I自身提供的 API来实现自己的功能;MiniGUI I中的“可移植层”可将特定操作系统及底层硬件的细节隐藏起来,而上层应用程序则无需关心底层的硬件平台输出和输入设备。

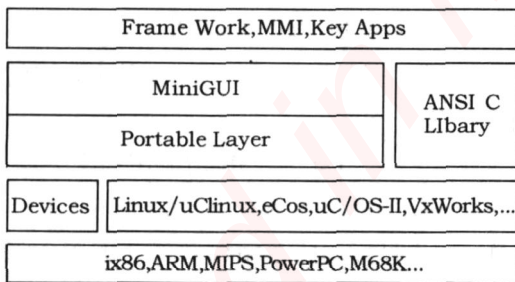


图 1 MiniGUI I和嵌入式操作系统的关系

4 基于 MiniGUI I的程序开发方法

下面结合一个简单实例介绍 MiniGUI I的基本编程方法。程序功能是在窗口中打印字符串“Hello World”。为了介绍方便,程序在一个源文件 helloworld.c里实现。

4.1 包含必要的头文件

helloworld.c的开始所包括的四个头文件 <minigui/common.h>、<minigui/minigui.h>、<

minigui/gdi.h>和 <minigui/window.h>是所有的 MiniGUI I应用程序都必须包括的头文件:

4.2 程序入口点

一个 C 程序的入口点为 main 函数,而一个 MiniGUI I程序的入口点为 MiniGU Main,该函数原型如下: int MiniGU Main (int argc, const char * argv[])。下面是 MiniGU Main函数的实现。

```
int MiniGU Main (int argc, const char * argv[ ] )
{
    MSG Msg;
    HWND hMainW nd;
    MA NW NCREATE Create Info;
    Create Info dwStyle = WS_V ISBLE | WS_BORDER |
    WS_CAPTION;
    Create Info dwExStyle = WS_EX_NONE;
    Create Info spCaption = "Hello, world!";
    Create Info hMenu = 0;
    Create Info hCursor = GetSystemCursor(0);
    Create Info hIcon = 0;
    Create Info MainW indowProc = HelloW inProc;
    Create Info lx = 0;
    Create Info ty = 0;
    Create Info rx = 320;
    Create Info by = 240;
    Create Info BkColor = COLOR_lightwhite;
    Create Info dwAddData = 0;
    Create Info hHosting = HWND_DESKTOP;
    hMainW nd = CreateMainW indow (&Create Info);
    if (hMainW nd == HWND_NVAL D)
        return - 1;
    ShowW indow (hMainW nd, SW_SHOWNORMAL);
    while (GeMessage (&Msg, hMainW nd) ) {
        TranslateMessage (&Msg);
        DispatchMessage (&Msg);
    }
    MainW indowThreadCleanup (hMainW nd);
    return 0;
}
```

(1)创建和显示窗口

hMainW nd = CreateMainW indow (&Create Info); 每个 MiniGUI I应用程序的初始界面一般都是一个主窗口,你可以通过调用 CreateMainW indow 函数来创建一个主窗口,其参数是一个指向 MA NW NCREATE结构的指针,本例中就是 Create Info,返回值为所创建主窗口的句柄。MA NW NCREATE结构描述一个主窗口的属性,在使用 Create Info 创建主窗口之前,需要设置它的各项属性。

创建完主窗口之后,还需要调用 ShowW indow

函数才能把所创建的窗口显示在屏幕上。ShowWindow 的第一个参数为所要显示的窗口句柄,第二个参数指明显示窗口的方式(显示还是隐藏),SW_SHOWNORMAL 说明要显示主窗口,并把它置为顶层窗口。

(2)消息循环

MiniGUI 为每一个 MiniGUI 程序维护一个消息队列。在发生事件之后,MiniGUI 将事件转换为一个消息,并将消息放入目标程序的消息队列之中。应用程序现在的任务就是执行如下的消息循环代码,不断地从消息队列中取出消息,进行处理:

```
while ( GetMessage( &M sg, hMainW nd) ) {  
    TranslateMessage( &M sg );  
    DispatchMessage( &M sg );  
}
```

GetMessage 函数从应用程序的消息队列中取出一个消息,TranslateMessage 函数把击键消息转换为 MSG_CHAR 消息,DispatchMessage 函数最终把消息发往该消息的目标窗口的过程,让其进行处理,处理完后返回到 DispatchMessage,在从 DispatchMessage 中返回到应用程序中,应用程序又从下一个 GetMessage 函数调用开始消息循环。

4.3 窗口过程函数

窗口过程函数是 MiniGUI 程序的主体部分,应用程序实际所做的工作大部分都发生在窗口过程函数中,因为 GUI 程序的主要任务就是接收和处理窗口收到的各种消息。在 helloworld.c 程序中,窗口过程是名为 HelloWinProc 的函数。CreateMainWindow 函数根据 MAINWINDOW_NCREATE 结构类型的参数中指定的窗口过程创建主窗口。窗口过程函数总是定义为如下形式:

```
static int HelloWinProc (HWND hW nd, int message,  
WPARAM wParam, LPARAM lParam) {  
    HDC hdc;  
    switch (message) {  
        case MSG_PAINT:  
            hdc = BeginPaint (hW nd);  
            TextOut(hdc, 100, 100, "Hello World!");  
            EndPaint(hW nd, hdc);  
            return 0;  
        case MSG_CLOSE:  
            DestroyMainWindow (hW nd);  
            PostQuitMessage (hW nd);  
            return 0; }  
    return DefaulMainWinProc (hW nd, message, wParam,  
lParam);  
}
```

程序通常不直接调用窗口过程函数,而是由 MiniGUI 进行调用;也就是说,它是一个回调函数。窗口过程函数不予处理的消息应该传给 DefaulMainWinProc 函数进行缺省处理。

4.4 在 Tomado 下编译 MiniGUI 程序

首先从飞漫公司的网站上下载运行于 PC 平台的 VxWorks 操作系统的 MiniGUI V1.6.x 开发包——“minigui-dev-1.6.9-vxworks.zip”,解压后可以得到 libmgext.a libminigui.a 等文件,然后按照以下步骤编写测试程序:

(1)在 Tomado 新建工作空间 test.wsp,在工作空间 test.wsp 下建立工程 project.wpj,并在 project.wpj 下建立 bootable.vxWorks.image,选择 PII 环境的 BSP。

(2)添加库 libmgext.a libminigui.a

(3)将 MiniGUI 下的 include 文件夹拷贝到 Tomado2.2\target\h 下。

(4)将 helloworld.c 添加到 project.wpj 中。

(5)将 helloworld.c 中的入口函数 MiniGU_Main 添加到 usrAppInit.c 中,然后进行编译。将生成的 vxWorks(应用程序)下载到目标机中,可看到实验结果如图 2 所示。

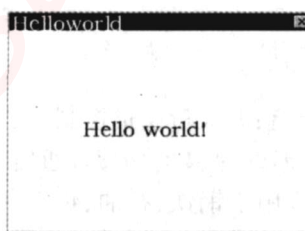


图 2 实验结果截图

4 结语

MiniGUI 现在已发展成为一个比较成熟的图形用户界面支持系统,它可以运行于 i386、ARM、MIPS、PowerPC、Cold-fire 等多种硬件平台上。能较好的支持包括 VxWorks、linux 在内的多种嵌入式操作系统。在 VxWorks 系统下采用 MiniGUI,可以比较快捷的完成相应的图形用户信息交互界面的开发,是 VxWorks 下一种较好的图形用户界面解决方案。

参考文献

- [1] 北京飞漫技术有限公司. MiniGUI 用户手册 V2.0.3/1.6.9 [EB/OL]. [2006-06]. <http://www.minigui.com/download/cindex.shtml>
- [2] 北京飞漫技术有限公司. MiniGUI 技术白皮书 for V2.0.3/1.6.9 [EB/OL]. [2005-09-9]. <http://www.minigui.com/download/cindex.shtml>
- [3] 北京飞漫技术有限公司. Datasheet for MiniGUI V2.0.3/1.6.9 [EB/OL]. [2006-06]. <http://www.minigui.com/download/cindex.shtml>
- [4] 北京飞漫技术有限公司. MiniGUI 编程指南 V2.0.3/1.6.9 [EB/OL]. [2006-06]. <http://www.minigui.com/download/cindex.shtml>

studied, the hardware interface is designed and a communication program between PC and DSP is developed. Through a concrete instance, it indicates that the method presented in this paper is simple, effective, and satisfy the demand of this control system.

Key words micro - displacement actuator, control system, DSP, serial communication (Page: 172)

Design & Implement of Physically Based Smoke Animation

by *Zi Lingling*

Abstract Fluid Simulation such as smock, water, fire, etc. is not only difficult but also attractive in computer graphics. This paper realizes smock simulation and animation using computational fluid dynamics and computer graphics. It gives smock physical model set and settle equations by way of Navier - Stokes equation and semi-lagrangian method. Finally smock density field data will be done and it will be changed screen image using volume rendering.

Key words physical model, smock, volume rendering (Page: 175)

Search Engine of Power System Documents

by *Liang Dichen*

Abstract Full-text retrieval system is a sort of computer information retrieval tool. In this paper, it propose a text retrieval system of design proposal, and introduce index database and full text database foundation, initialization problem, problem of design search function, design of resultant page, and so on.

Key words recall rate, precision rate, search function (Page: 177)

Application for Two Work Place Qian Portions Electronic Control System

by *Chen Xiaozhu*

Abstract This article mainly introduces the design thought of Mitsubishi FX2N PLC applying for two work place qian portion of the electronic control system. It gives a detailed statement on the components of the system hardware and design of software. Especially for bilateral automatic welding methods made more detailed description of programming ideas. For debugging showed: the electrical control system to completely satisfy the requirements of control. This has greatly enhanced the efficiency of production machine tools.

Key words PLC, two work place qian portions, STL, programming ideas (Page: 180)

Simple Analyzed Content - based Multimedia Retrieval Technology

by *Chen Fang*

Abstract Content - based multimedia retrieval technology manage multimedia information in a new way. It in-

cludes structured organizing and browsing the video media and processing of image and audio signal. By using this technology, widely access of information is possible. This paper analysis the technology, characteristics and methods of content - based multimedia retrieval. At the end, the paper points out some problems which needs mending, such as feature extraction, semantic knowledge acquisition, content - based coding and so on.

Key words multimedia, content - based, retrieval (Page: 183)

Create and Extraction of the Block's Attribute of AutoCAD

by *Huang Wei*

Abstract The block's attribute of AutoCAD, by which word information or data information can be added on block, is a variable character area. This method can avoid overabundance of inefficient information and complexity in the process of extracting and treating. The method can assure that information it extracted is needed list information. The list establishing by this method has uniform mode, is easy to be managed in AutoCAD and can meet the requirement of file standardization in enterprise.

Key words computer aided design, block's attribute, item list, data extraction (Page: 187)

Development of MiniGUI Graphical Interface Based on VxWorks

by *Ming Fang*

Abstract MiniGUI is an embedded graphics support system for real time embedded products with high efficiency, reliability, scalability, and configurability, can support for many different real time embedded operating systems, including Linux, VxWorks, Win32 platform and so on. This paper describes briefly the technical features of MiniGUI and its advantages, then discusses the software architecture of embedded systems based on MiniGUI. Finally introduces programming methods of MiniGUI based on VxWorks.

Key words MiniGUI, GUI, VxWorks (Page: 189)

Brief introduction on anti-interference in PLC's application

by *Ma Jieyu*

Abstract this paper mainly introduces the application of programmable controller in industrial control field and a number of problems on ensuring its normal operation during application process of PLC, at the same time, it gives out some wise proposals.

Key words PLC, industrial control, anti-interference, wiring, grounding, proposal (Page: 192)

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](#)
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](#)
83. [基于 DSP 的 CANopen 通讯协议的实现](#)
84. [基于 PCI9656 控制芯片的高速网卡 DMA 设计](#)
85. [基于以太网及串口的数据采集模块设计](#)

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](#)
54. [VxWorks 环境下内存文件系统的应用](#)
55. [VxWorks 下的多重定时器设计](#)
56. [Freescale 的 MPC8641D 的 VxWorks BSP](#)
57. [VxWorks 实验五\[时间片轮转调度\]](#)
58. [解决 VmWare 下下载大型工程.out 出现 WTX Error 0x100de 的问题](#)
- 59.

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](#)
52. [Linux 基于 W83697 和 W83977 的 UART 串口驱动开发文档](#)

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](#)
29. [Windows CENET 下 ADC 驱动开发设计](#)

30. [Windows CE 下 USB 设备流驱动开发与设计](#)
31. [Windows 驱动程序设计](#)
32. [基于 Windows CE 的 GPS 应用](#)
33. [基于 Windows CE 下大像素图像分块显示算法的研究](#)
34. [基于 Windows CE 的数控软件开发与实现](#)

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 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](#)
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. [基于 FPGA 的 CAN 总线通信节点设计](#)

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

12. [基于 FPGA 的高速时钟数据恢复电路的实现](#)
13. [基于 FPGA 的高阶高速 FIR 滤波器设计与实现](#)
14. [基于 FPGA 高效实现 FIR 滤波器的研究](#)
- 15.