

# 基于FPGA的高速时钟数据恢复电路的实现

李湘琼 黄启俊 常胜

(武汉大学, 物理科学与技术学院电子科技系)

**摘要:** 介绍了一种利用输出时钟在具有不同相位的时钟信号之间进行切换实现高速时钟恢复电路的方法。利用Altera公司Quartus软件提供的修改逻辑单元和逻辑块锁定及插入buffer的方法, 消除了时钟切换产生的毛刺, 弥补了不同相位时钟由于不同的传输延迟而造成的相位偏移。设计的电路实现了数字光端机要求的204.8MHz的工作频率。同时, 分析了决定该电路工作频率的主要因素, 通过仿真验证使用EP3C10E144C7芯片最高工作频率可以达到400MHz。

**关键词:** 时钟恢复; Logiclock; 超前滞后鉴相器; 现场可编程逻辑门阵列

## Design of high-speed clock and data recovery circuit Based on FPGA

Li Xiangqiong Huang Qijun Chang Sheng

(Department of Electronics Science and Technology, College of Physics Science and Technology, Wuhan University)

**Abstract:** The paper presents a method to realize high-speed clock and data recovery circuit, which is based on the idea of utilizing output clock to make switching among the clocks whose phase are different. By using the modified logic elements, locking logic region provided by Altera's Quartus and inserting buffers, the burrs appearing in clock switching is eliminated and the phase offset generated from different transmitting delays of the clocks with different phases is compensated. The designed circuit achieves the operation frequency, which meet the requirement of our project. The main factor which affects the operation frequency of the presented circuits is also analyzed in this paper. The result of simulation based on the chip of EP3C10E144C7 shows that a highest operation frequency of 400 MHz can be achieved.

**Key words:** clock recovery; logiclock ; the early-later phase detector; FPGA

### 0 引言

时钟数据恢复电路是高速收发器的核心模块, 而高速收发器是通信系统中的关键部分。随着光纤在通信中的应用, 信道可以承载的通信速率已经可以达到GHz, 从而使得接收端的接收速率成为限制通信速率的主要瓶颈。因此高速时钟数据恢复电路的研究是目前通信领域的研究热点。目前时钟数据恢复电路主要是模拟IC和数字IC, 其频率已经可以达到几十GHz。而由于FPGA器件的可编程性、低成本、短的设计周期以及越来越大的容量和速度, 在数字领域的应用逐渐有替代数字IC的趋势, 已经广泛作为数字系统的控制核心。但利用中低端FPGA还没有可以达到100MHz以上的时钟数据恢复电路。由于上面的原因, 许多利用FPGA实现的高速通信系统中必须使用额外的专用时钟数据恢复IC, 这样不仅增加了成本, 而且裸露在外的高速PCB布线使还会带来串扰、信号完整性等非常严重的问题。如果可以在中低端FPGA上实现高速时钟数据恢复电路, 则可降低成本且提高整个电路系统的性能。

目前利用FPGA实现时钟恢复电路的方法, 基本都是首先利用FPGA内部的锁相环产生 $N \cdot f$ 的高频时钟, 然后再根据输入信号控制对高速时钟的分频, 从而产生与输入信号同步的时钟信号[1~3], 其中 $N$ 决定了恢复时钟信号的相位精度, 通常 $N$ 等于8。因此如果输入信号的频率为100MHz, 则系统的工作频率就必须达到800MHz, 对于中低端FPGA, 如此高的工作频率显然无法承受。虽然高端FPGA可以达到GHz的工作频率, 但其高昂的价格不适合用于普通用户。而其它基于中低端FPGA实现高速时钟恢复电路的方法, 要么需要外部VCO模块[4], 要么只能恢复数据而无法得到同步的时钟信号[5]。针对这种情况, 本文提出了一种利用Altera FPGA中的锁相环及Logiclock等技术, 实现高速时钟恢复电路的方法。电路是在Altera的EP2C5T144C6芯片上实现的, 用于数字光端机的接收端

从100路2.048MHz压缩视频码流合成的串行码流中正确提取100路视频码流, 其工作频率为204.8MHz, 通过硬件验证电路可以正确工作。

### 1 时钟恢复电路原理及环路结构

时钟恢复电路的目的是从输入的数据流中, 提取出与其同步的时钟信号。时钟信号不可能凭空产生, 因此该电路本身必须有一个时钟信号产生机制, 除此之外还必须有一个判断控制机制—能够判断并且调整该时钟信号与输入数据之间的相位关系, 使其同步。

传统的基于FPGA的时钟恢复电路的结构如图1所示。如前所述, 这种结构的电路用中低端FPGA, 工作频率不可能达到100MHz以上。本文采用的方法是利用锁相环产生不同相位的时钟信号, 然后再根据控制信号控制输出时钟在这些时钟之间进行切换, 从而使时钟与输入数据同步。具体结构如图2所示。下面详细介绍各个模块的工作原理及电路实现。

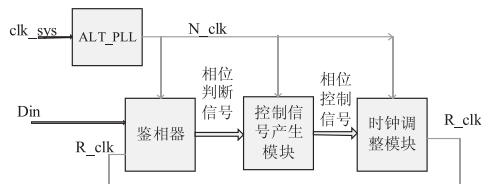


图1 基于高频时钟分频的时钟恢复电路结构图

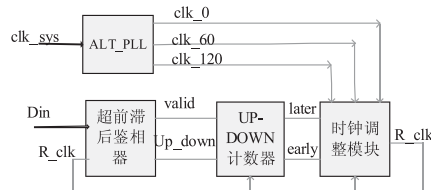


图2 基于时钟切换的时钟恢复电路结构图

## 2 模块电路设计

电路由三个模块构成，鉴相器模块和计数器模块通过判断时钟信号和输入信号的相位关系，产生相位调整的控制信号，时钟调整模块根据送来的控制信号对输出时钟进行相位调整。

### 2.1 超前滞后鉴相器

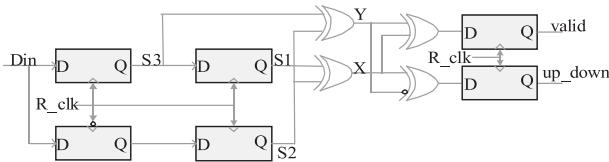


图3 超前滞后鉴相器电路

鉴相器使用可以识别输入信号连0和连1的超前滞后鉴相器[6]，具体电路如图3所示。利用四个D触发器对输入信号Din采样产生s1、s2、s3三个信号，如果s2=s3≠s1，则表示时钟超前，up-down为高电平；如果s1=s2≠s3，则表示时钟滞后，up-down为低电平；当输入信号连0或连1时s1=s2=s3，valid为低电平，此时up-down无效。这两个信号都必须用时钟信号进行同步产生，否则电路可能由于s1、s2、s3的延迟差而进入一种死锁状态。

这种鉴相器只有超前或滞后两种状态，如果直接将其输出用作控制时钟相位的调整，则时钟切换会过于频繁，而且输入信号中的毛刺会引起误操作。所以将其输出信号送给up-down计数模块，进行一段时间的积累后，再产生用于控制时钟相位的信号。

### 2.2 up-down计数器模块

计数器模块[2]的功能是在鉴相器送来的信号控制下进行计数，产生相位调整的控制信号。计数器的初始值为M，当输入信号valid为高电平时，判断up-down信号。如果该信号为1，则计数器加计数，否则减计数。当计数器的值为2M时，early产生高电平脉冲；当为0时，later产生高电平脉冲。

### 2.3 时钟产生调整模块

#### 2.3.1 电路结构

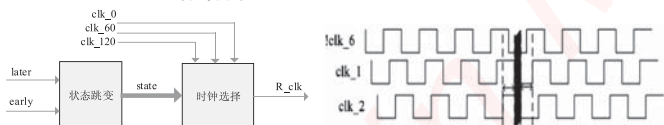


图4 时钟产生调整模块结构图

图5 时钟正确切换的时序

时钟产生调整模块的主要功能是产生和输入信号频率相同的时钟信号，并根据相位判断模块发送过来的控制信号，不断地调整输出时钟相位，使得输出时钟的上升沿维持在输入信号中间位置，其结构如图4所示。不同相位的同频时钟是利用EP2C5T144C6中的锁相环产生的，由于课题项目的需要，1个片子内部必须含有2个时钟恢复电路，受到全局时钟数目的限制，采用6个时钟信号进行切换。在该模块电路设计设置6个状态，每一状态对应某一相位的时钟信号。当检测到early信号为高电平时，状态就跳变到比当前时钟信号相位提前1个相位的状态上（若相位超前则再继续超前），而当later信号为高电平时跳变到比当前时钟信号滞后1个相位的状态上去（相位滞后则继续滞后），然后再根据当前的状态选择相应的时钟信号，作为当前工作时钟即输出时钟信号R\_clk。

#### 2.3.2 时钟切换

采用时钟切换实现该相位调整，首要问题就是必须消除时钟切换时产生的毛刺，因此所有子电路都采用恢复

出来的时钟信号作为其工作时钟，这样所有的时钟切换情况都可用图5表示。假定此时的工作时钟为clk\_1，只要能够保证时钟切换避开上面的阴影区域，而是处在上图中的红色区域，则无论是时钟向前还是向后切换，都不会出现毛刺。以工作频率为200MHz计算，每段红色区域大约为1.6ns，而Altera中的LE单元延迟大约为0.23ns，所以采用buffer弥补延迟完全可以使时钟切换发生在合适的位置。

#### 2.3.3 时钟相位偏移的补偿

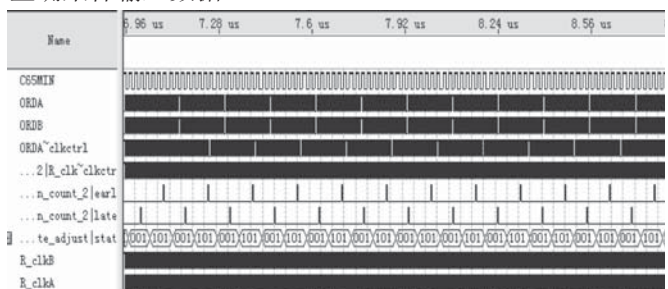
该电路结构的另一关键是必须保证不同相位时钟信号经过时钟判断调整模块后，它们之间的相位关系不会因延迟的不同而失效而必须仍然成立，否则就会导致相位调整过大或过小，电路会因此变得不稳定。电路延迟由门延迟和连线延迟组成。对于连线延迟在Altera Cyclone II中只要两个LE之间的连线类型相同，延迟也就相同，而每个LAB中有16个LE也有16个LOCAL\_LINE布线资源，因此同一个LAB中LE之间的互连线延迟都是相同的。对于门延迟，在Cyclone II中每个LE有A、B、C、D四个输入端，各端口延迟不同。首先通过在某些路径上插入buffer，使得所有时钟信号经过的LE的数目相同。然后再在3个时钟信号中各插入一个buffer，再根据最终的布线情况，调整这3个buffer的端口连接，就可以使得各时钟信号经过的门延迟基本相同。除了上述两种方法，在应用中使用了Quartus II的logiclock功能，按照设计的需要将同一功能模块的逻辑放在相同的LAB中，这样可以使连线延迟对电路的影响降到最小。同时利用该功能也可以直接将调整好的时钟恢复电路应用于整个通信系统中，使得其他电路的布局布线不会对该电路产生影响[7]。

### 2.4 数据恢复

由于时钟数据恢复电路的目的就是得到能够正确采样输入数据的时钟信号。为了保证恢复出的时钟和输入数据的相位关系在整个芯片内都能够成立，在系统应用中输入信号也走全局时钟网络。经过这样处理后，只要在需要恢复数据的地方加一个D触发器即可得到正确的数据信号。

## 3 环路仿真结果及讨论

采用EP2C5T144C6器件的环路的仿真结果如图6所示，其中工作频率为204.8MHz。由于整个电路用的是经过全局时钟控制模块后的时钟和输入数据，所以在图6中给出的是这两个信号时序关系：ORDA~clkctrl和R\_clk~clkctrl。由图(a)及局部放大图(b)可以看出，电路稳定时钟信号在输入数据的中间位置左右摆动，可以正确采样输入数据。

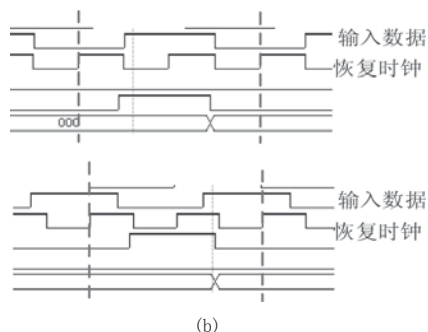


(a)

(b) (参见下页)

图6 (a)整体电路电路仿真结果 (b)局部放大仿真波形





(b)

本文提出的电路结构不需要高频时钟信号，因此只要相位调整过程中时钟信号的脉冲宽度大于器件要求的最小值，且满足整个电路满足建立保持时间就可以正常工作。因此最小的时钟周期 $T = \max(3 \cdot T_{\min}, T_{\text{per}})$ ，其中 $T_{\min}$ 是所用器件时钟信号脉冲宽度的最小值， $T_{\text{per}}$ 是满足建立保持时间的最小时钟周期。例如EP2C35F672C6芯片工作时钟高电平脉冲最小值为1ns，采用该芯片电路的工作频率可以达到300MHz，这个工作频率已经通过了Altera DE2板的硬件验证。如果采用更快的器件如Cyclone III EP3C10T144C7，时钟脉冲宽度最小值0.625ns，经过仿真验证其工作频率可以达到400MHz。在时钟调整模块后加一个简单的二分频电路，就可以实现12个时钟相位的调整精度，根据不同器件的性能很容易进行扩展，达到所需要的设计要求。

#### 4 结论

本文利用时钟切换的方法，在低端Altera EP2C5T144C6上实现了204.8MHz的时钟数据恢复电路，并通过了硬件验证。通过理论分析给出了决定该电路工作频率的主要因素，同时对该电路稍加改动就可以实现更高精度的时钟数据恢复电路，具有很好的扩展性，为利用中低端FPGA实现高速通信系统提供了一种可参考的解决方案。

#### 参考文献:

- [1] Best R E. 锁相环设计、仿真与应用[M]. 北京: 清华大学出版社, 2003.
- [2] 李新昌, 吴嗣亮, 王旭. 数字锁相环技术在恢复位时钟中的应用[J]. 军民两用技术与产品. 2004, (2): 39-41.
- [3] 张文, 姚福安, 候磊. 基于FPGA实现的一种新型数字锁相[J]. 现代电子技术. 2007, (4): 183-185.
- [4] LatticeSC MACO Core LSCDR1X18 Low-Speed Clock and Data Recovery User's Guide[R].
- [5] Sawyer N. Data Recovery[M]. XAPP224 (v2.5) July 11, 2005.
- [6] Razavi B. Challenges in the design of high-speed clock and data recovery circuits[J]. IEEE Communications Magazine. 2002, (8): 94-101.
- [7] Altera. Quartus II Version 7.1 Handbook[R].

(上接58页)

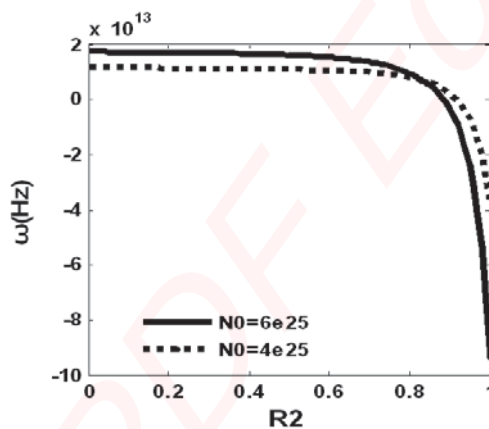


图3 光纤本身固有振荡频率 $\omega_0$ 随 $R_2$ 的变化

了腔内激光强度的平均，使得输出激光自脉冲减弱。同时也可以抑制光纤本身固有的振荡。实验中如果采用高掺杂浓度的光纤，也有利于抑制自脉冲行为。

#### 参考文献

- [1] Glas P, Naumann I, Cher A, et al. Self pulsing versus self locking in cw pumped neodymium doped double clad fiber laser[J]. Opt. Commun., 1999, 161(3): 345-358.
- [2] Ortac B, Hideur A, Chartier T, et al. Influence of cavity losses on stimulated Brillouin scattering in a self-pulsing side-pumped ytterbium-doped double-clad fiber laser [J]. Opt Commun, 2003, 215(4-6): 389-395.
- [3] LeBoudec P, Francois P L, Delevaque E, et al. Influence of ions pairs on the dynamical behavior of Er<sup>3+</sup>-doped fiber laser[J]. Opt. Quantum Electron., 1993, 25(8): 501-507.
- [4] Rangel-Rojo R, Mohebi M. Study of the onset of self-pulsing behavior in an Er-doped fiber laser[J]. Opt Commun, 1997, 137(1): 98-102.
- [5] 陈默. 双包层掺镱光纤激光器及固体激光器弛豫振荡特性研究[D]. 清华大学, 2003.
- [6] El-Sherif A F, King T A. Dynamics and self-pulsing effects in Tm<sup>3+</sup>-doped silica fibre lasers[J]. Opt Commun, 2002, 208(4): 381-389

# 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](#)
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](#)

## 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](#)

## 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 CPU/GPU 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.