

基于 UEFI 的可信 Tiano 设计与研究

曾颖明, 谢小权

(中国航天科工集团第二研究院 706 所, 北京 100854)

摘要:对 UEFI BIOS 典型的实现-Tiano 系统进行了分阶段剖析,针对其所存在的安全缺陷及不足,结合可信计算的技术和思想,提出了可信 Tiano 这一新概念。介绍了可信 Tiano 系统的总体设计和工作流程,并对可信 Tiano 系统关键技术做详细研究,包括信任根的实现机制、可信度量组件和 EFI 映像签名,最后给出可信 Tiano 系统与标准 Tiano 系统的对比分析。通过分析容易看出,改进后的 Tiano 系统能较好地满足新一代 BIOS 在可信及安全方面的需求。

关键词:统一可扩展固件接口; Tiano; 可信计算; 可信平台模块; EFI 映像

中图分类号:TP393.08 **文献标识码**:A **文章编号**:1000-7024(2009)11-2645-04

Design and study of trusted Tiano based on UEFI

ZENG Ying-ming, XIE Xiao-quan

(Institute 706, Second Academy of China Aerospace Science and Industry Corporation, Beijing 100854, China)

Abstract: Security defects and insufficiency of current Tiano are pointed out by analysis the typical realization of UEFI BIOS, a new concept of trusted Tiano combined with trusted computing technology and idea is introduced. First the overall design and workflow of the system is introduced. Key technology research of the system is presented as well including the realization mechanism of the root of trust, trust measurement components and EFI signature image. Finally, comparative analysis is given between trust Tiano and standard Tiano. It is easy to see that this design can meet the needs of trusted and security of a new generation BIOS.

Key words: UEFI; Tiano; trusted computing; TPM; EFI image

0 引言

随着传统 BIOS 的局限越来越明显,统一可扩展固件接口(unified extensible firmware interface, UEFI)规范给 BIOS 的研发带来了开放性和标准性,其易于扩展和移植,必将成为新一代 BIOS 标准。为了更好地推广和促进 UEFI 技术发展,Intel 给出了一个符合 UEFI 接口规范的参考实现,即 Tiano 系统。BIOS 是连接软件程序和硬件设备之间的枢纽,在计算机系统中担负着重要的作用,负责计算机启动过程中系统硬件的检测初始化及操作系统引导操作,其可信安全直接影响整个计算机系统的安全度。BIOS 中存在的安全隐患,相对于运行于操作系统中的病毒、安全漏洞来讲,具有不易清除,不易监控及隐蔽性强的特点。由于 BIOS 处于计算机系统底层,当 BIOS 发生不可预知的故障时,会造成计算机系统的崩溃;当恶意者对 BIOS 进行攻击时,计算机系统从底层被恶意控制。

Intel 公司在最初构建和设计 Tiano 系统时,并没有考虑安全因素,因此,其相比传统 BIOS 的优点,如模块化设计、接口设计、高级语言编写、组件加载模式、硬盘存储方式等都使其比传统 BIOS 有更多的安全问题。正是考虑到 Tiano 系统的这些安全缺陷,本文结合可信计算的思想,并根据可信计算机对 BIOS 的安全需求来对可信 Tiano 系统进行设计和研究。

1 Tiano 系统

1.1 介绍

与传统 BIOS 实现机制有所不同,Tiano 系统为了模块化,将启动过程分成了多个阶段,每个阶段由不同的模块来实现,各阶段之间都有定义好的接口。从平台启动到平台初始化,再到操作系统启动,最后到关机共有 7 个阶段,每个阶段由不同的模块来实现,各阶段之间都有定义好的接口。在操作系统启动前有 4 个阶段:安全检测(SEC(security))、EFI 初始化准备(Pre EFI initialization environment, PEI)、驱动执行环境(driver execution environment, DXE)、启动设备选择(boot device selection, BDS)。操作系统启动后有 3 个阶段:瞬时系统加载(transient system load, TSL)、系统运行时(run time, RT)、生命后(after life, AL)。

1.2 Tiano 系统的安全问题

下面开始对 Tiano 系统所存在的安全问题进行分析。

(1) 存储安全隐患 EFI 将 Tiano 系统的 SEC 和 PEI 模块保存在主板上的 BIOS FLASH 芯片中,其它模块的程序和数据保存在硬盘某段保护扇区内。如果闪存出问题,Tiano 系统将无法正常启动;如果硬盘出问题,容易使 Tiano 系统某些功能模块完整性受到破坏。由于用户的误操作或硬盘攻击,存储在硬盘中的部分 Tiano 模块很容易遭受破坏或更改。

(2)DXE映像加载漏洞 DXE阶段需要确保驱动程序的加载过程是完整可信的。过程的完整性包括:单个组件的完整,防止组件被破坏,感染病毒,植入木马;该过程所需要的组件不能缺失,不能被伪造或替换,并保证各个组件在过程中的顺序,如果顺序发生变化,可能导致计算机系统进入某种不安全状态。因此 Tiano 系统在加载驱动过程中需要对驱动程序的完整性进行验证,并能准确而及时地判断出映像文件是否被非法篡改,并保证以正确的顺序进行加载。

(3)BDS阶段的安全问题:在该阶段对系统资源和启动项进行管理,以防出现不安全访问,应该在该阶段增加对用户身份的合法性认证。

(4)缓冲区溢出:在 Tiano 系统中,绝大部分代码都是用 C 语言来编写,缓冲区溢出是一种高级语言编程错误,黑客利用缓冲区溢出攻击,可以导致程序运行失败、系统宕机、重新启动等后果。更为严重的是,可以利用它执行非授权指令,甚至可以取得系统特权,进而进行各种非法操作。缓冲区溢出是 Tiano 系统安全防护的新难题。

(5)网络通信问题:在连接通信前,服务器和客户端有必要对对方进行身份鉴别和完整性检测,来保证通信及自身系统平台的安全性。EFI 具有硬件驱动,可以独立地控制计算机,能远程修改配置、更换驱动、载入木马模块等,所以在使用底层的 EFI 网络协议实现通信时,服务器和客户端都有必要在建立通信前,与通信方进行身份鉴别和完整性检测,否则会对自身系统平台的安全构成威胁。

通过以上分析可以看出,Tiano 系统在安全性和稳定性方面还需要做很多工作。

2 可信 Tiano

由于 Tiano 本身是分阶段的模块化的 BIOS 系统,所以可以而且应该利用可信计算的思想来对 Tiano 系统的各个模块分别进行设计,并在模块之间加入可信控制,再结合 TPM 安全芯片,构建一条 Tiano 内部信任链,保障 Tiano 系统的可信安全,即构建一个可信 Tiano 系统,从而扩展到整个计算机的可信安全。

2.1 可信 Tiano 的总体设计

可信 Tiano 系统的设计思想是:平台上电后即建立一个可信的度量根(CRTM 部分由 Tiano 的 SEC 构成,存储在 FLASH 芯片中),然后初始化硬件安全芯片 TPM 芯片作为信任链的可信存储根和可信报告根,接着在 Tiano 子模块控制权转移过程中,上一级模块对下一级执行模块加以认证,从而建立一条 Tiano 系统内的信任链。

由图 1 可以看出,在设计可信 Tiano 中,增加了两个模块,分别是 TPM 安全芯片和作为 UEFI API 一部分的 Trust API。

可信 Tiano 需要完成对 TPM 芯片的驱动及设计一个专门负责在 Tiano 中进行可信度量的组件。由于 UEFI 的特殊机制,该组件不仅可以完成对 Tiano 各模块的度量,还可以使可信度量组件作为 UEFI 运行时服务,即可以在操作系统下继续进行度量工作。

2.2 可信 Tiano 的工作流程

程序模块在执行之前需要进行完整性度量,这就形成了

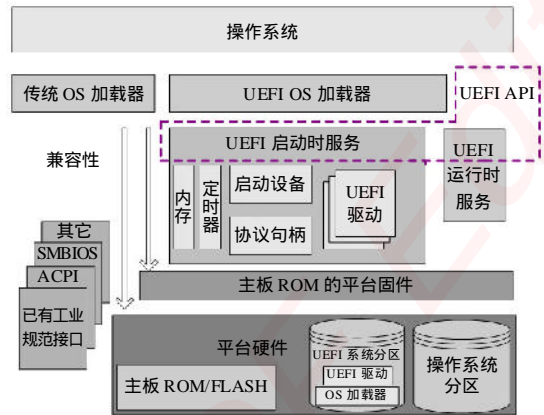


图 1 可信 Tiano 总体架构

平台的执行流和度量流两条流程,执行流是模块之间控制权的传递,度量流是完整性度量的过程。如图 2 所示,图中序号代表先后顺序。可以看出,下层先对上层要执行的代码作完整性度量,度量通过后,把控制权传递给上层,开始执行流。即一层度量一层,一层信任一层,从而把这种信任扩展到整个平台系统。

(1)Tiano 的安全检测阶段 SEC 作为 CRTM,并结合 TPM 安全芯片共同作为可信 Tiano 最初的信任根,即先初始化 TPM 芯片;

(2)做 CPU、芯片组、主板、内存工作,建立 C 代码执行环境;对 PEI 代码进行可信度量,保存度量日志 SML 到内存。将度量的哈希值结果保存到 TPM 芯片中对应的 PCR 寄存器,并与存放于 TPM 芯片安全存储区域中的哈希值比较,若相同,表示可信度量通过,则加载执行 PEI,否则进入可信恢复模式;

(3)PEI 对 DXE 代码进行可信度量,保存度量日志 SML 到内存。将度量的哈希值结果保存到 TPM 芯片中对应的 PCR 寄存器,并与存放于 TPM 芯片安全存储区域中的哈希值比较,若相同,表示可信度量通过,则加载执行 DXE,否则进入可信恢复模式;

(4)DXE 在加载 EFI 映像之前,对被加载映像进行可信度量(引入 EFI 映像签名机制),如可信度量成功,则加载映像,否则不加载执行;

(5)DXE 对 BDS 代码进行可信度量,保存度量日志 SML 到内存。将度量的哈希值结果保存到 TPM 芯片中对应的 PCR 寄存器,并与存放于 TPM 芯片安全存储区域中的哈希值比较,若相同,表示可信度量通过,则加载执行 BDS,否则进入可信

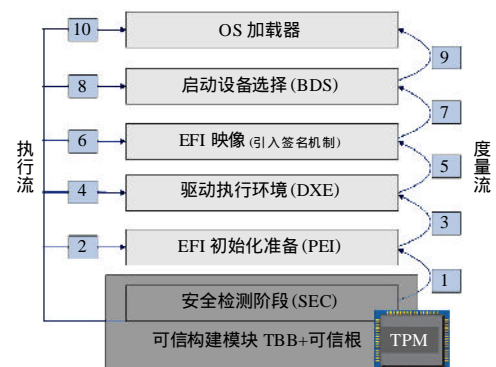


图 2 可信 Tiano 的工作流程

恢复模式；

(6) BDS 对 OS 加载器进行可信度量,保存度量日志 SML 到内存。将度量的哈希值结果保存到 TPM 芯片中对应的 PCR 寄存器,并与存放于 TPM 芯片安全存储区域中的哈希值比较,若相同,表示可信度量通过,则加载执行 OS 加载器,否则进入可信恢复模式；

(7) OS 加载器开始执行后,操作系统执行前的可信链建立完毕。

至此,经过以上(1)~(7)过程,完成可信 Tiano 中信任关系的逐级建立和传递,可信 Tiano 可以基于这种信任链传递的机制将信任根的可信性扩展到了计算机的其它部分,从而构造了计算机的完整信任链。

2.3 可信 Tiano 关键技术分析

2.3.1 信任根的实现机制

在设计可信 Tiano 中,如何保证 TPM 芯片初始化是平台最先执行的模块是个待解决的难题。通过深入研究 UEFI 规范,本文采用优先级描述文件机制来保证可信 Tiano 中具有 TBB 功能的模块最先被加载,成为第一个执行模块。

2.3.2 可信度量组件

可信度量组件是度量请求方与可信 Tiano 的度量交互模块,属于 Trust API 的一部分。它必须能够同时运行在 BIOS 和操作系统环境下,因此,可信度量组件在可信 Tiano 中应该以 EFI 运行时服务的形式而存在,因为只有 EFI 运行时服务可以在 BIOS 启动后继续运行。这样一来,它的度量对象可以包括 EFI 映像、EFI 数据、操作系统加载器、操作系统核心文件、操作系统环境下的应用程序及数据等。另一方面,可信度量组件作为信任链传递的惟一度量组件,利于更好地保证信任链的传递及整个计算机平台的安全。

2.3.3 EFI 映像签名

EFI 映像可能位于不安全的介质上,也可能通过不安全的传输方式传输,也可能通过一个不安全的端口传入。针对这些情况,平台在加载 EFI 映像时,可信度量组件要认证 EFI 映像的来源及完整性。由于在平台加载过程中,会加载很多 EFI 映像,这样一来,如果可信度量组件对每个 EFI 映像都进行完整性度量的话,显然会严重影响平台启动性能和效率。

因此,本文引入一种对 EFI 映像进行数字签名的方法来取代对被加载 EFI 映像的可信度量。同时,UEFI 规范的启动完整性服务(boot integrity services, BIS)协议提供了校验 EFI 映像完整性的有效策略机制,即启动授权密钥(boot authentication key, BA Key)。平台系统的合法所有者拥有一对 BA 公/私钥,本文就是用这一对公私钥来实施 EFI 映像的签名。对 EFI 映像签名和嵌入的方式可以用在平台获取新的 EFI 映像时,先对映像进行签名验证,判断其是否得到合法授权,从而保证平台的安全性,而不必推迟到执行过程再验证;如果平台在获取新 EFI 映像时已经对其嵌入签名并做了验证,在映像的完整性和发布方均可信的前提下,就不必在加载时再对此映像作完整性度量。

数字签名必须由两块内容组成:一是数据(也常称作映像),二是一对公/私钥。为了创建数字签名,先选用某一种哈希算法对映像做处理,得到一个固定长度的 Hash 值。该 Hash

值和 BA 私钥通过一个签名算法加密后成为一个数字签名。原映像和经过加密的数字签名封装在一起成为签名后的映像,如图 3 所示。

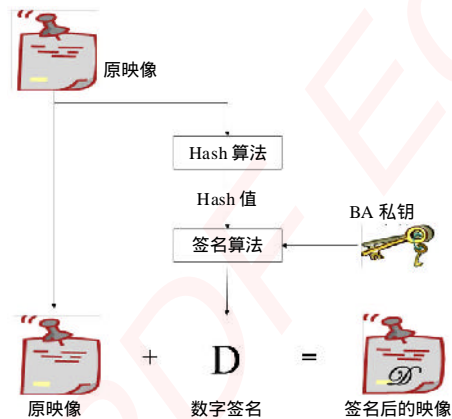


图 3 创建数字签名

为了验证一个签名,需要用到原始映像和公钥。首先,把签名后的映像分解成原映像和数字签名。用与 BA 私钥对应的 BA 公钥对签名值解密,得到定长的哈希值。同时对分解出的原映像进行哈希运算也得到相同长度的 Hash 值。然后对这两个 Hash 值进行比较,如果一致,则说明映像未被篡改,验证的过程如图 4 所示。

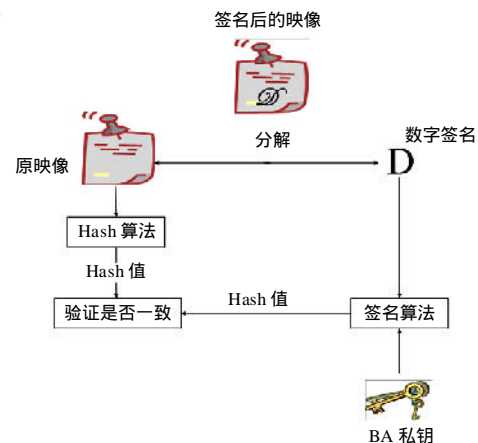


图 4 验证数字签名

3 对比分析

本文提出的可信 Tiano 与标准 Tiano 的不同之处,主要体现在以下几方面:

容易集成到整个计算机架构:可信 Tiano 与安全增强型 CPU、安全增强型系统芯片组、TPM 芯片、可信操作系统、可信计算软件栈、可信网络连接软件等的有机结合而构建成的可信计算环境,使应用程序进程能够在相互隔离的内存空间中运行,从而有效地防止各种恶意代码和缓冲区溢出攻击,确保应用程序运行过程中的可信赖性和数据安全性。

更完整的信任链:可信 Tiano 中,以 TPM 芯片和 Tiano 的 SEC 为信任源,建立了一个完整的从 TBB 到 DXE、EFI 映像、启动设备选择、操作系统加载器、操作系统核心模块、应用程

序直到网络通信的信任链传递模型,从而把信任扩展到整个平台系统。与标准 Tiano 相比,可信 Tiano 构造了一条更完整的信任链,能有效地解决了 EFI 启动问题、加载过程漏洞和网络通信问题。

更好的扩展性:由于 UEFI 的特殊机制,使可信 Tiano 系统具有良好的扩展性。比如可信度量组件不仅可以运行在可信 Tiano 下,而且在操作系统下仍然可以运行。可信 Tiano 不受操作系统的限制,只需移植安全服务接口的软件工程,可以在不同操作系统下的运行使用,维护量小。另一方面,在操作系统环境下提供应用时,安全模块的安全服务接口集成在可信 Tiano 的内部,并且不需要操作系统级的 TPM 驱动。

更主动的 BIOS 级安全防护:可信 Tiano 不受操作系统自身的安全隐患限制,结合可信计算,安全性相对主动,可以提供一定意义上主动的 BIOS 级安全防护。可以看出,可信 Tiano 为保证计算机终端的可信安全发挥了重要的作用。

4 结束语

随着 UEFI 日益得到工业界的认可和发展,其将给我国 BIOS 的发展和自主创新带来极大的机遇,因此研究 Tiano 安全问题并构建可信 Tiano 是十分必要的。可信 Tiano 还处于初步设计阶段,各个子模块的详细设计还有待进一步的完善,同时这些技术也许还不能够满足特定主机的安全防护需

求,应当结合传统 BIOS 的安全增强防护来共同构建可信安全的 Tiano 系统。由于 Tiano 是由国外公司提出和设计实现的,在其基础上构建可信 Tiano 还不能保证我国特殊应用领域中计算机的安全,因为核心技术在国外厂商手里。如果我国能够根据 UEFI 接口规范,完全自主设计一个 UEFI 实现系统,而在该系统下去构建可信 UEFI BIOS 将会有更加积极的意义。

参考文献:

- [1] Intel.Unified extensible firmware interface specification 2.1[Z].2007.
- [2] TCG.TCG EFI platform specification 1.2[Z].2006.
- [3] TCG.TCG EFI protocol 1.2[Z].2006.
- [4] TCG.TCG specification architecture overview 1.4[Z].2007.
- [5] 徐娜.基于可信计算平台的可信执行环境研究与实现[D].北京:中国科学院计算技术研究所,2006.
- [6] 周伟东,池亚平.一种基于信任根加强 EFI BIOS 自身安全的方案[J].学术研究,2007,7: 87-91.
- [7] 章睿,刘吉强,彭双和.基于 EFI 的信任链传递研究及实现[J].计算机应用,2007,27(9):2174-2176.
- [8] 王江少,余综,李光.可信计算之信任链技术研究[J].计算机工程与设计,2008,29(5):2195-2198.

(上接第 2644 页)

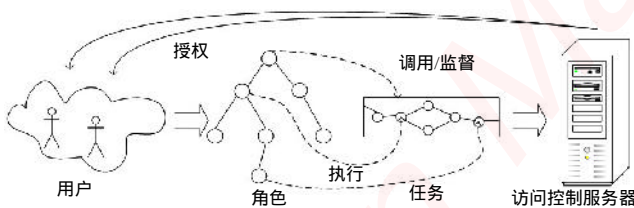


图 7 采用新方法后的访问控制过程

“需做工作”原则,完全支持职责分离原则和最小权限原则,支持任务代理,且更符合商业组织中多层角色关系的实际;任务组合原语从任务之间的包含关系和工作流基本形式出发,实现了访问控制系统对任务执行顺序的描述,破除了固定工作流机对商业应用和访问控制扩展性的约束,并根据面向协作的角色继承规范了主体自定义 workflows 任务的能力范围。

面向服务的体系结构是当前开发应用的主流,分布式协作系统也是在这个环境下诞生的。如今的访问控制中对服务的管理大都要吸收任务的思想,本文在基于任务-角色的访问控制中用原子任务模式化操作,用组合任务来创建和发布服务,提高了服务构成的灵活性。

限于篇幅,文章只针对两个问题进行了研究,没有最终形成模型,在提出任务组合原语思想时没有考虑工作流程中数

据流动,以及多个安全域间的服务交互问题,是下一步的主要研究方向。

参考文献:

- [1] Eldon Y Li, Timon C Du, Jacqueline W Wong. Access control in collaborative commerce[J]. Decision Support Systems, 2007, 43: 675-685.
- [2] Sandhu R S, Ferraiolo D F, Kuhn R. The NIST model for role based access control: Towards a unified standard[C]. Proceeding of the 5th ACM Workshop on Role-Based Access Control. Berlin: ACM Press, 2000: 47-63.
- [3] Sejong Oh, Seog Park. Task-role-based access control model[J]. Information Systems, 2003, 28: 533-562.
- [4] 陈娟娟. 基于角色-任务的工作流系统存取控制模型[D]. 武汉: 华中师范大学, 2003.
- [5] 付松龄, 谭庆平. 基于任务和角色的分布式工作流安全模型[J]. 国防科技大学学报, 2004, 26(3): 57-62.
- [6] YU Min, YU Wen-sen. An improved role hierarchy model for role based access control[J]. Journal of Jiangxi Normal University, 2004, 28(1): 11-14.
- [7] 邓集波, 洪帆. 基于任务的访问控制模型[J]. 软件学报, 2003, 14(1): 76-82.

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](#)

邀请注册码



关注论坛公众号

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](#)

邀请注册码



关注论坛公众号

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](#)
86. [MVB1 类设备控制器的 FPGA 设计](#)
87. [MVB 接口彩色液晶显示诊断单元的显示应用软件设计](#)
88. [IPv6 新型套接字的网络编程剖析](#)
89. [基于规则的 IPv4 源程序到 IPv6 源程序的移植方法](#)
90. [MVB 网络接口单元的 SOC 解决方案](#)
91. [基于 IPSec 协议的 IPv6 安全研究](#)
92. [具有 VME 总线的车载安全计算机 MVB 通信板卡](#)
93. [SD 卡的传输协议和读写程序](#)
94. [基于 SCTP 的 TLS 应用](#)
95. [基于 IPv6 的静态路由实验设计](#)
96. [基于 MVB 的地铁列车司机显示系统研究](#)
97. [基于参数优化批处理的 TLS 协议](#)
98. [SSD 数据结构与算法综述](#)
99. [大容量 NAND Flash 文件系统中的地址映射算法研究](#)
100. [基于 MVB 总线的动车组门控系统的设计与仿真研究](#)
101. [多功能车辆总线 MVB 控制](#)

邀请注册码



关注论坛公众号

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](#)

邀请注册码



关注论坛公众号

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. [基于 VxWorks 系统的 MiniGUI 图形界面开发](#)
60. [VxWorks BSP 开发中的 PCI 配置方法](#)
61. [VxWorks 在 S3C2410 上的 BSP 设计](#)
62. [VxWorks 操作系统中 PCI 总线驱动程序的设计与实现](#)
63. [VxWorks 概述](#)
64. [基于 AT91RM9200 的 VxWorks END 网络驱动开发](#)
65. [基于 EBD9200 的 VxWorks BSP 设计和实现](#)
66. [基于 VxWorks 的 BSP 技术分析](#)
67. [ARM LPC2210 的 VxWorks BSP 源码](#)
68. [基于 LPC2210 的 VxWorks BSP 移植](#)
69. [基于 VxWorks 平台的 SCTP 协议软件设计实现](#)
70. [VxWorks 快速启动的实现方法\[上电到应用程序 1 秒\]](#)

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](#)

邀请注册码



关注论坛公众号

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](#)
53. [基于 AT91RM9200 的嵌入式 Linux 系统的移植与实现](#)
54. [路由信息协议在 Linux 平台上的实现](#)
55. [Linux 下 IPv6 高级路由器的实现](#)
56. [基于 Android 平台的嵌入式视频监控系统设计](#)

邀请注册码



关注论坛公众号

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](#)
35. [NAND FLASH 在 WINCENET 系统中的应用设计](#)

邀请注册码



关注论坛公众号

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](#)
31. [基于 MPC850 的多功能通信管理器](#)
32. [基于 MPC8640D 处理系统的技术研究](#)
33. [基于双核 MPC8641D 处理器的计算机模块设计](#)
34. [基于 MPC8641D 处理器的对称多处理技术研究](#)

邀请注册码



关注论坛公众号

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](#)
36. [UBoot 在 AT91RM9200 上的移植简析](#)
37. [基于工控级 AT91RM9200 开发板的 UBoot 移植分析](#)
38. [基于 ARM11 和 Zigbee 的人员定位防丢器](#)

邀请注册码



关注论坛公众号

39. [基于 NAND FLASH 的嵌入式系统启动速度的研究](#)
40. [μ COS II 在 ARM7 上的移植](#)

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 技术研究与实现](#)
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](#)

邀请注册码



关注论坛公众号

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](#)
41. [EFI-Tiano 环境下的 AES 算法应用模型](#)
42. [EFI 及其安全性研究](#)
43. [基于 UEFI Shell 的 PreOS Application 的开发与研究](#)
44. [UEFI Bootkit 模型与分析](#)
45. [UEFI 计算机系统快速调试方法的实现](#)
46. [基于 EFI 系统的多文件系统解决方案](#)

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. [用 IPv6 编程接口实现有连接通信的方法](#)
17. [一种战棋游戏的 AI 算法设计与实现浅析](#)
18. [基于 TLS 协议的 ECC 扩展研究](#)

邀请注册码



关注论坛公众号

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](#)
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. [FPGA 的 VHDL 设计策略](#)
16. [用 FPGA 实现串口通信的设计](#)
17. [GPIB 接口的 FPGA 实现](#)
18. [一种基于 FPGA 的 FFT 阵列处理器](#)
19. [基于 FPGA 的 FFT 信号处理器的硬件实现](#)

邀请注册码



关注论坛公众号