

嵌入式系统开发与应用系列教程

# ARM9 嵌入式Linux 开发实验与实践

田 泽 编著



嵌入式Linux

嵌入式Linux



北京航空航天大学出版社

TP332

80D

2006

嵌入式系统开发与应用系列教程

# ARM9 嵌入式 Linux 开发实验与实践

田 泽 编著

北京航空航天大学出版社

## 内 容 简 介

以 ARM9 核的 Embest EDUKIT - II/III 多核实验教学系统为硬件开发平台,以嵌入式 Linux 操作系统为软件开发平台,设计开发了 21 个实验,内容包括:Linux 启动代码编程实验,Linux 内核基础实验,Linux 驱动程序实验,Linux 文件系统及应用程序添加、调试实验,Linux 应用程序设计实验,Linux 的 GUI 及综合应用实验。内容安排上从易到难,相对完整,密切结合嵌入式系统教学实际,贴近嵌入式 Linux 开发的实际工程应用,并提供了丰富的从入门到系统开发的实验例程。使读者不仅能够系统地掌握嵌入式 Linux 开发的基本知识和技能,而且会为基于嵌入式 Linux 的嵌入式系统开发打下坚实的基础。书中附英蓓特公司免费提供的光盘 1 张,内含 Embest IDE Pro 校园版安装软件及其他相关资料。

本书可作为高等院校电子、通信、自动化、计算机等专业嵌入式系统课程实验教学的教材,也可作为相关工程技术人员嵌入式培训教材及参考用书。

### 图书在版编目(CIP)数据

ARM9 嵌入式 Linux 开发实验与实践/田泽编著. —北京:北京航空航天大学出版社,2006. 11

ISBN 7 - 81077 - 813 - 7

I. A… II. 田… III. ①微处理器,ARM9②Linux 操作系统 IV. ①TP332②TP316. 89

中国版本图书馆 CIP 数据核字(2006)第 122914 号

© 2006,北京航空航天大学出版社,版权所有。

未经本书出版者书面许可,任何单位和个人不得以任何形式或手段复制或传播本书及其所附光盘内容。

侵权必究。

### ARM9 嵌入式 Linux 开发实验与实践

田 泽 编 著

责任编辑 王 鹏

\*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:010-82317024 传真:010-82328026

http://www.buaapress.com.cn E-mail:bhpress@263.net

涿州市新华印刷有限公司印装 各地书店经销

\*

开本:787×960 1/16 印张:22.5 字数:504 千字

2006 年 11 月第 1 版 2006 年 11 月第 1 次印刷 印数:5 000 册

ISBN 7 - 81077 - 813 - 7 定价:29.50 元(含光盘 1 张)

## 嵌入式技术的发展

以 8 位单片机(或微控制器 MCU)为核心的嵌入式系统早已广泛应用在各个领域,但这些应用大多数还处于单机使用的嵌入式低层次阶段。其特点是以 MCU 为核心,与一些简单的传感器、监测设备、伺服控制、指示、显示设备等配合,实现一定的测量、显示、信息处理及控制等功能。即便是在一些工业控制和汽车电子、智能家居的多机应用中,为了实现多个 MCU 构成系统间的信息交流,通常是利用 CAN、RS-232、RS-485 等总线将 MCU 组网。但这种网络的应用空间有限,相关的通信协议也比较单一,并且一般孤立于广泛应用的 Internet 之外。目前,Internet 已成为社会重要的基础信息设施,是信息流通的重要渠道。嵌入式系统必将与 Internet 完美融合,方便、低廉地将信息传送到世界上任何一个地方。

随着嵌入式设备与 Internet 的广泛结合,手机、PDA、路由器、调制解调器等复杂的高端应用对嵌入式处理器的性能提出了更高的要求。虽然以 8 位为核心的嵌入式技术不断发展,性能也不断提高,但由于其性能的局限性,已无法满足未来高性能嵌入式技术的发展需求。市场的竞争、技术的竞争要求不断降低嵌入式系统开发的成本,提高嵌入式系统的性价比;同时,也要求缩短嵌入式系统的开发周期。自从 20 世纪 70 年代初出现了嵌入式系统的概念以来,嵌入式系统以其高性能、低功耗、低成本等特点高速发展。当时的嵌入式系统很多都不采用操作系统,它们只是为了实现某个控制功能,使用一个简单的循环控制对外界的控制请求进行处理。随着嵌入式系统的发展,当应用系统越来越复杂、使用范围越来越广泛时,每增加一项新的应用功能,都可能需要从头开始设计系统软件,没有操作系统已成为其最大的缺陷。8 位处理器运行嵌入式操作系统也是有困难的,因此,以 32 位处理器作为高性能嵌入式系统开发的核心已是嵌入式技术发展的必然趋势。

20 世纪 90 年代后,嵌入式系统设计从以嵌入式微处理器/DSP 为核心的“集成电路”级设计逐渐转向“集成系统”级设计,提出了片上系统 SoC(System on a Chip)的基本概念。目前,嵌入式系统已进入以 SoC 为核心的设计阶段,并开始逐步实用化、规范化。SoC 为高性能嵌入式系统开发提供了功能丰富的硬件平台,也为实时嵌入式操作系统的广泛使用提供了硬件基础。从 20 世纪 80 年代开始就陆续出现了一些嵌入式操作系统,比较著名的有 VxWorks、Windows CE、Palm、 $\mu$ Clinux、pSOS 和  $\mu$ C/OS-II 等。但真正广泛使用却是近年来的事情,一方面,嵌入式系统软件开发复杂度的增加要求嵌入式操作系统的支持;另一方面,大量高性能、面向实际应用、集成多种系统功能的 SoC 芯片成为高端嵌入式应用的硬件核心,为可靠、高效、低成本地运行嵌入式操作系统提供了硬件平台。大部分嵌入式操作系统价格昂贵,而源代码开放的  $\mu$ C/OS-II、 $\mu$ Clinux、Linux 是大家比较看好的,也比较适用于教学。有关嵌入式操作系统的知识,一些出版社已经出版了多种相关专业书籍,请参阅相关资料。

嵌入式微处理器具有体积小、重量轻、成本低、可靠性高等优点,是嵌入式系统的核心。目



前,比较有影响的 32 位嵌入式处理器有 ARM 公司的 ARM 系列、Compaq 公司的 Alpha、HP 公司的 PA-RISC、IBM 公司的 Power PC、MIPS 公司的 MIPS 和 Sun 公司的 Sparc。而 ARM 处理器具有高性能、低功耗、低成本等显著优点,已成为高性能、低功耗嵌入式处理器的代名词,是目前 32 位、64 位嵌入式处理器中应用最为广泛的一个系列。

20 世纪 90 年代初,半导体行业产业链形成设计业、制造业、封装测试业三业分离的产业分工。台积电、联电等半导体代工厂崛起,一些 fabless、chipless 公司如雨后春笋般出来,英国先进 RISC 机器公司(Advanced RISC Machines,简称 ARM 公司)就是 fabless、chipless 这一生产模式最为成功的典范。ARM 公司既不生产芯片,也不销售芯片,而是设计出高效的 IP 内核,授权给各个半导体公司使用。半导体公司在 ARM 技术的基础上,根据自己公司的产品定位,添加自己的设计并推出芯片产品,最后由 OEM 客户采用这些芯片来构建基于 ARM 技术的最终应用系统产品。经过 10 多年的发展,ARM 公司已成为业界领先的 IP 核供应商。

回顾 ARM 技术的发展历程,在 ARM7 体系结构(体系结构版本 v3)刚被广泛接受和使用时,嵌入式微处理器的市场仍然大都由 8 位、16 位处理器占领。然而,这些产品却不能满足当时移动电话、调制解调器等高端应用对处理器性能的要求。这些高端产品需要 32 位 RISC 处理器的性能和更优于 16 位 CISC 处理器的代码密度,这就要求要以更低的成本取得更好的性能和更优的代码密度。为了满足这种发展需求,ARM 体系结构增加了 T 变种,开发了一种新的指令体系,即 16 位 Thumb 指令集。Thumb 技术是 ARM 技术的一大特色。ARM 发布的 ARM7TDMI 体系结构(体系结构版本 v4T)的处理器,是第一款支持 Thumb 指令集的微处理器,其工作模式可以方便地切换到 Thumb 状态。在该状态下运行的指令集是 16 位 Thumb 指令集,这在当时的 16 位系统与现在需要的 32 位系统之间搭起了一座桥梁。性能更优而不需要付出额外代价的 ARM,对那些在当时使用 8 或 16 位处理器,却一直在寻找更优性能的用户来说,提供了解决方案,也极大地推动了 ARM 技术的发展。16 位处理器一直没有像人们预期的那样发展起来的原因很复杂,由于 32 位 ARM 处理器提供了低于一般 16 位微处理器的价格而高于一般 16 位微处理器的性能,使得高端的嵌入式应用可以完全跨越 16 位而直接进入 32 位应用领域,这也许是原因之一。

ARM 处理器得到了众多半导体厂家和整机厂商的大力支持,全球已有 103 家巨型 IT 公司在采用 ARM 技术,20 家最大的半导体厂商中有 19 家是 ARM 的用户,包括 TI、Philips 和 Intel 等公司。优良的性能和准确的市场定位极大地丰富了 ARM 资源,加速了基于 ARM 核的、面向各种应用系统芯片的开发应用,使得 ARM 获得了更广泛的应用,确立了 ARM 技术的市场领先地位。ARM 在高性能嵌入式应用领域获得了巨大的成功,已在 32 位嵌入式应用中稳居世界第一。在 2002 年,基于 ARM 核的芯片占据了整个 32、64 位嵌入式微处理器市场的 79.5%,全世界已使用了 20 多亿个 ARM 核。如今,ARM 公司已经成为业界的龙头老大,“每个人口袋中都装着 ARM”是毫不夸张的,因为几乎所有的手机、移动设备、PDA 都是使用基于 ARM 核的系统芯片开发的。为了顺应当今世界技术革新的潮流,了解、学习和掌握高性能嵌入式技术,就必然要学习以 32 位 ARM 微处理器为核心的嵌入式开发技术以及其应用开发环境和平台。

对 IT 产业发展规律进行总结发现:如果说过去 20 年 PC 机的广泛应用是集成电路和 IT 相关技术发展的驱动器,并且极大地促进了 IT 相关技术发展,那么未来几十年除了 PC 技术要继续高速发展之外,主要驱动器应该是与 Internet 相结合的、可移动的(Mobile)、便携的

(Portable)、实时嵌入式 Internet 的信息处理设备,即进入了后 PC(Post-PC)时代。目前,嵌入式 Internet 还仅局限于移动商务、智能家居(家电上网)、工业控制和智能设备的应用等方面。随着相关应用技术的发展,嵌入式技术必将与许多实际应用领域相结合,以难以想像的速度发展。这必然会极大地拓展嵌入式应用的广度和深度,体现嵌入式系统与实际应用密切结合的价值。

当代通用计算机工业是 Wintel(Microsoft 和 Intel 公司 20 世纪 90 年代初建立的联盟)垄断的工业。随着信息技术和网络技术的高速发展,嵌入式技术的广泛应用使得这种垄断在后 PC 时代不再存在。嵌入式系统正是非 PC 设备的主体。互联网技术在世界范围的扩展以及中国通信事业的高速发展,为我国开发嵌入式产品造就了巨大市场。与实际应用密切结合且技术高度分散的嵌入式技术,为我国 IT 技术的发展提供了难得的机遇。加快嵌入式技术的开发应用,掌握嵌入式开发核心技术,对于 IT 技术并不发达的中国是非常重要的。

## 嵌入式教学现状和 32 位嵌入式系统教学体系的建设

发展嵌入式技术,人才是关键。培养和培训相关人才,一方面要在学生中加强嵌入式系统教学,这是人才的源头;另一方面,有一定实际项目开发经验的工程技术人员也需要加强嵌入式开发技术的学习。目前,大学生就业难,而许多单位又招聘不到合适的开发人员。国内教育与人才需求之间存在着不相适应的严重矛盾。虽然产生这一矛盾的原因很多,但就目前的工科教育而言,存在着重知识传授而轻能力与素质培养、重理论讲述而轻实践环节训练、重共性教育而轻个性发展、重对传统的继承而轻对现状的突破和创新等众多问题。反映在学生的相关课程教学体系设置中,就是可以提高自主开发能力的训练条件不具备;没有规范性的工程实际训练课程;已开设的相关课程教学内容陈旧,无法满足日新月异的高新技术变革,而且广度有限,深度不够。反映在实践教学过程中,就是没有一流的、与实际工程应用密切结合的实验教学设施,以及相关实践性课程内容设计,使高校培养的人才创新意识薄弱,实践能力不强,与实际工程应用需求严重脱节。这些原因可能是当前工科人才培养不能满足实际工程需求的一个重要因素。因此,用人单位不得不承担繁重的工程应用再教育的任务,这严重地制约了我国相关技术的发展。

就我国大部分高校目前的嵌入式教学而言,整体仍然是停留在 20 世纪 80 年代初发展起来的以 8 位 51 单片机为核心的单片机教学水平上。总结近 20 多年来我国以单片机为核心的嵌入式教学情况,虽然它极大地促进了中国 IT 技术的发展,但是由于受传统教学体系的影响,以及我国高等教育条件的限制,目前教学内容设置、教学方法、教学手段、教材编写体系与这门课程以实际应用为主的基本特征严重脱节。学生学完这门课程后,只能进行一般的、基于指令基础上的简单编程,而不能进行基本的系统应用设计。在此基础上要想完成具有较大规模的应用程序设计是有一定难度的,要独立进行一般的开发工作就更加困难。一般学生进入技术开发岗位,基本上要进行相当长一段时间的实际培训,才能基本进入开发流程。如果要独立承担项目,还需要更长一段时间。

以 32 位 ARM 为核心的嵌入式技术日益成为高性能嵌入式应用的基础,面向实际工程应用、以 ARM 为核心的嵌入式技术培训课程也越来越受到工程技术人员的欢迎,这正说明了在高校建立一个新的、基于 ARM 技术的嵌入式系统教学课程体系的必要性和迫切性,这可以从源头上解决嵌入式技术发展对人才的需求。

当然,与传统的以 8 位 51 单片机开发应用相比,ARM 的性能和处理能力是遥遥领先的,



应用也日益广泛,但与之相应地,基于 ARM 的嵌入式系统软、硬件开发的复杂度和难度也急剧加大。要建立一个新的、以 ARM 技术为核心的嵌入式系统开发与应用的课程体系时,就需要总结以往以 51 单片机为核心的嵌入式系统教学经验,同时也要与 ARM 技术的自身特点及国内实际的教育基础和产业发展的具体情况进行密切地结合。既要考虑大大提高目前我国嵌入式系统教学的水平,又不能脱离现在的教学实际;既要强调嵌入式基础教育,又要面向实际工程应用,提升嵌入式系统教学的实用性和工程性,将比较复杂的嵌入式系统各个模块的综合开发的例程、嵌入式操作系统移植等内容写入教材。

作者早期翻译了 ARM bible 级的《ARM SoC 体系结构》一书,并长期从事基于 ARM 的相关教学、科研工作。近些年,作者多次应邀参加嵌入式教学体系建设的研讨会,并在国内 10 多个大中城市开设了师资培训班,同 ARM 公司、众多的基于 ARM 核的芯片厂商、相关课程的老师,以及对 ARM 感兴趣的读者建立了广泛的联系。通过与他们的交流,深感建立 32 位嵌入式教学体系,科学、规范地编写一套基于 ARM 的嵌入式系统开发与应用系列教材很有必要,这使我们萌发了编写该系列教材的念头并付诸于实践中,并于 2004 年 6 月开始陆续出版,为嵌入式系统的教学起到了一定的促进作用。

通过与教师们的交流,以及本人在本科生、研究生教学和科研工作中的实践,作者体会到,不同层次的高校及不同专业,对嵌入式教学的需求不一样,深度也不一样,有的偏向硬件接口及应用功能开发的学习,而有的注重操作系统的学习。就处理器选择而言,有的希望以 ARM7 开课,有的希望以 ARM9 开课,而有的希望以 Intel 公司的 XScale 开课。对操作系统的学习,有的希望讲授 WinCE 或 VxWorks,而有的希望以开源的  $\mu\text{C}/\text{OS}-\text{II}$ 、 $\mu\text{Clinux}$  或嵌入式 Linux 为主。这反映了嵌入式应用的多样性和复杂性,同时也给教材的编写增加了难度,通过一本理论教材或实验教材全面反映这些内容是不现实的。因此本系列教程配套了多本实验教程,以满足不同高校不同专业的需求。

### 本系列教程的组成

本系列教程将理论教学与实践教学融为一体。实验教程都是以 Embest ARM 实验教学系统为实验平台而开发的。本系列教程的组成如下:

#### 理论教程

《嵌入式系统开发与应用教程》(配套多媒体教学课件)

#### 实验教程

《嵌入式系统开发与应用实验教程(第 2 版)》

——配套:Embest S3CEV40 实验教学系统,多媒体实验教学课件

《ARM7 嵌入式开发实验与实践》

《ARM9 嵌入式开发实验与实践》

《ARM7  $\mu\text{Clinux}$  开发实验与实践》

《ARM9 嵌入式 Linux 开发实验与实践》

《ARM9 WinCE 开发实验与实践》

——配套:Embest EDUKIT-II/III 多核实验教学系统,多媒体实验教学课件





## 本系列教程相关实验教学平台介绍

进行基于 ARM 的嵌入式系统开发时,选择合适的开发工具可以加快开发进度,节省开发成本。目前,世界上有几十家公司提供不同类别的 ARM 开发工具和产品。进行 ARM 应用的开发工具主要包括集成开发环境 IDE、JTAG 仿真器、评估板等。

一般来说,一套具备最基本功能的嵌入式集成开发环境 IDE 是嵌入式系统开发所必不可少的。目前,国内外主要使用的 IDE 环境有 ARM 公司的 SDT 和 ADS、美国 Green Hills 软件公司的 Multi 2000、台湾的 Hitool for ARM,以及英蓓特公司的 Embest IDE for ARM 等,而国内使用较多的 IDE 为 ARM 公司的 SDT、ADS,以及英蓓特公司的 Embest IDE for ARM。Embest IDE 是英蓓特公司在国内较早开发的集成了 GNU 的汇编器 as、编译器 gcc、链接器 ld 等工具的集成开发环境,同时公司也提供了兼容 ADS/SDT 的开发工程、支持 SDT 编译器等版本的 Embest IDE。

SDT 和 ADS 是 ARM 公司早期的集成开发环境(IDE),其用户界面不太符合中国人使用的习惯,而且软件本身价格很高,仿真器也是价格不菲,国外软件对用户的售后服务以及技术支持都是有限的。SDT 和 ADS 因为盗版严重,早已停止升级,无法满足未来 ARM 嵌入式技术的发展。为了满足未来复杂 ARM SoC 调试的挑战及防止盗版,实现快速、高效的软件开发,ARM 公司开发了 RealView 系列 ARM 调试解决开发工具,可以实现 SoC、基于 SoC 的产品开发以及基于 SoC 应用软件的开发调试,它提供的高性能和高灵活性能够满足甚至超越复杂片上系统(SoC)工程调试的要求,但价格昂贵。

仿真器有 ARM 公司的 Muti-ICE 和英蓓特公司的 Embest 系列 ARM JTAG 仿真器。

英蓓特公司的 ARM 集成开发环境套件(包括 Embest IDE、指令集模拟器、仿真器及 Flash 编程器等相关开发工具)是国内最早的 ARM 开发工具之一,完全自主开发,本土技术支持实力雄厚,且价格合理。同时,能为嵌入式系统开发人员提供软硬件开发工具和嵌入式系统完整解决方案及技术信息服务,且可以满足未来 ARM 嵌入式技术发展的需要。因此,本套教程选用英蓓特公司的集成开发环境套件作为 ARM 应用开发的教学工具。

Samsung 公司 S3C 系列芯片是国内市场占有率最高的 ARM 核处理器芯片,基于 ARM7 核的 S3C44B0X 芯片、基于 ARM9 核的 S3C2410X 芯片应用广泛,其接口模块丰富,适用面广,比较适合教学实验。英蓓特公司开发的 Embest EDUKIT - II/III 多核实验教学系统包括基于 S3C44B0X(ARM7)和 S3C2410X(ARM9)的 CPU 子板,以较低的价格为用户提供了全面的嵌入式学习与开发的平台。EDUKIT - II/III 多核实验教学系统集成了几乎所有的嵌入式硬件接口,并提供了可以选配的 GPRS 模块、GPS 模块,系统配置了  $\mu$ C/OS - II、 $\mu$ Clinux、WinCE、嵌入式 Linux、VxWorks 等操作系统。该实验系统标准配置提供的 ARM7 和 ARM9 CPU 子板都可自由插拔,方便用户根据需求自由配置和更换。ARM7 实验系统可以实现基础的 ARM7 嵌入式教学,主要包括指令实验、基础接口实验、 $\mu$ C/OS - II 操作系统实验和  $\mu$ Clinux 操作系统实验;ARM9 实验系统可以实现高端的 ARM9 嵌入式教学,主要包括扩展接口实验、嵌入式 Linux 操作系统实验和 WinCE 操作系统实验。

除 ARM 核 CPU 子板外,EDUKIT - II/III 多核实验教学系统还支持 Intel 公司 XScale270、ADI 公司 DSP Blackfin533 作为 CPU 子板,可作为系统的选择配置模块。EDUKIT - II/III 多核实验教学系统可以满足嵌入式各种层次的教学需求,也为老师根据学生和专业特





点以及未来发展提供了灵活选择。本系列教程中的实验与实践教程以 EDUKIT - II/III 教学系统作为实验平台正是基于这种考虑的。

### 本系列教程的教学前期知识储备

嵌入式开发与应用的内容繁杂,涉及到基本的硬件知识(如嵌入式处理器及基本的接口知识、扩展的人机接口、网络通信接口等)、操作系统(应该至少了解一种操作系统的中断、优先级、任务间通信、同步等知识)、程序设计知识(C、C++、汇编语言程序设计,至少要熟练 C 语言);同时,还要具备一定的数字电路知识以及使用示波器、逻辑分析仪等基本技能。因此,在系统地学习本门课程之前,必须先学习微机原理与接口、C 语言程序设计课程,并了解一些计算机操作系统原理、体系结构和系统结构的基本概念,同时对于网络协议也要有一定的了解。为了兼顾各层次学生的基本需求,我们在理论、实验教材中,对于涉及到的相关内容,加了一些必要的背景知识,如网络协议的基本介绍、触摸屏基本知识、键盘接口程序设计的基本知识等,使学生既能专心于嵌入式知识的学习,也能使他们在较短的时间内对相关知识有所了解。

### 致 谢

在本书的编写过程中,得到了北京航空航天大学出版社、ARM 中国总裁谭军博士、北京航空航天大学何立民教授的关心和指导,北京大学信息科学技术学院盛世敏教授、于敦山博士和蒋安平博士也给予了大量的支持和帮助。感谢我的博士导师中国科学院微电子研究所的仇玉林研究员、北京大学大盛世敏教授,感谢我的硕士导师西安交通大学邱祖廉教授,感谢他们对我一如既往的关心、理解和帮助。他们将我带入科学研究的殿堂,并给予我各种帮助,是我无穷的精神源泉,在此表示深深感谢!

在本系列教程的编写过程中,研究生万永波、闫效莺、杨峰、车晓萍、陈群英、李攀、王进军、余兆安、黄信兵、倪祯浩、刘平、黄鹏、强亚东、马永刚、夏大鹏、淮治华等同学,验证了部分程序并添加了详细的注解,使之更贴近于面向基础的课程教育,在此表示感谢!

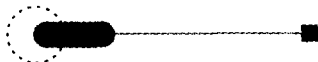
深圳市英蓓特信息技术有限公司的徐光峰、张国瑞、黄日新、唐晖等工程师也提供了更加详细的 Embest ARM 实验教学系统的技术资料和各种帮助,并对本套教材以及实验体系的形成、内容设计及完善提出了大量意见和建议,在此表示感谢!

感谢我的爱人王永红给予我的理解和支持,是她在家庭中默默地劳作和操持,使我可以安心于工作。正是由于她给予我最及时、最需要的关心和照顾,使我在单调的工作之余,生活总是绚丽、多彩。感谢我的儿子田祎琨,我很少在生活和学习上给予他照顾,希望他幼小的心灵能够理解我!

感谢所有帮助过我的人们,有了他们的理解、帮助和支持,我才能完成我的写作。

由于时间仓促等众多客观条件的制约,书中难免存在各种错误和不足,敬请读者谅解,并真诚地欢迎读者提出宝贵的意见和建议。希望教育界、科研界、产业界携手并进,抓住嵌入式为我国 IT 发展提供的难得机遇,促进我国嵌入式技术快速、稳定、健康的发展。

田 泽  
2006 年 7 月



# 前 言

---

本书是《嵌入式系统开发与应用系列教程》中嵌入式操作系统实验教程之一,是构建一个新的以 ARM 技术为核心的嵌入式系统开发与应用教学课程体系的重要部分。本书以基于 ARM9 核(S3C2410X)的 Embest EDUKIT - II/III 实验教学系统为硬件开发平台,以基于 ARM9 广泛使用的嵌入式 Linux 操作系统为软件开发平台,系统建立了基于 ARM9 相对完整的嵌入式 Linux 开发实验与实践教学与培训体系。

第 1 章是关于嵌入式系统开发的基本情况介绍,主要是基于 ARM 的嵌入式开发环境、工具和基于 ARM 的学习情况介绍。

第 2 章是关于 Embest EDUKIT - II/III 实验教学系统的组成和各组成部分的使用介绍。

通过以上两章使读者对嵌入式开发与应用的基本情况、嵌入式开发工具和实验教材所对应的 Embest EDUKIT - II/III 实验教学系统的基本构成有个全面了解。

第 3 章是 Linux 开发环境的建立,包括 Cygwin 开发环境、Linux 开发环境、编译工具、源代码文件、实验例程说明、环境变量及编译、烧写映像文件,以及本书的相关文档约定。

第 4 章是 Linux 启动代码编程实验,包括 VIVI 实验、VIVI 编译及使用实验。

第 5 章是 Linux 内核基础实验,包括 Linux 编译运行实验和 Linux 内核调试实验。

第 6 章是 Linux 驱动程序实验,包括 LED 字符设备驱动程序实验、LCD 显示驱动程序实验和动态加载模块实验。

第 7 章是 Linux 文件系统及应用程序添加、调试实验,包括 cramfs 文件系统实验、FTP 应用程序调试实验设计 NFS 方式调试应用程序实验。

第 8 章是 Linux 应用程序设计实验,包括串口通信实验、I<sup>2</sup>C 总线接口实验、多进程与进程间通信实验、网络应用程序实验、文件和目录操作实验、实时时钟实验、A/D 转换实验、键盘和 LED 控制实验、EEPROM 读/写实验。

第 9 章是 Linux 的 GUI 及综合应用实验,包括 Qt 实验和 Web 服务器实验。

附录的主要内容是指令集速查表、指令集编码表及随书光盘的内容介绍。

本书从教学和实验角度出发,将基于 ARM9 的 Linux 开发的复杂工程例程按照学习的自然过程,分类写入到实验教程中,在加强基于 ARM9 的嵌入式 Linux 操作系统基础教学的同时,希望在一个较为真实的实际工程平台上进行实验教学,形成从易到难、相对完整、贴近实际工程应用的嵌入式 Linux 操作系统实验教学体系。本书包含大量软件和硬件资源,也可以作为基于 ARM9 的 Linux 开发的技术参考手册。

嵌入式实验教学体系涉及到计算机接口技术、计算机应用软件开发技术、计算机操作系统



技术、应用电子技术、网络通信技术等内容繁杂。在本书的内容设计以及教学实验系统开发过程中,密切结合了基于 ARM9 的 Linux 开发的实际工程应用。通过本书的学习,使得读者能够系统掌握基于 ARM9 的 Linux 嵌入式开发的基本知识和技能,为基于 ARM9 的 Linux 嵌入式开发打下坚实的基础。

基于 ARM 的嵌入式教学体系建设刚刚开始,而嵌入式应用开发又涉及到软、硬件及操作系统等复杂的知识。因此它的教学体系、尤其是面向实际开发应用的教学体系的建立,是一项非常复杂的系统工程。书中难免存在一些错误和不足,敬请读者谅解,并欢迎读者提出宝贵的意见和建议。

田 泽

2006 年 8 月

# 目 录

## 第 1 章 嵌入式系统开发与应用基础

1.1 基于 ARM 的嵌入式开发环境概述 .....	2
1.1.1 交叉开发环境 .....	2
1.1.2 模拟开发环境 .....	3
1.1.3 评估电路板 .....	4
1.1.4 嵌入式操作系统 .....	4
1.2 ARM 开发工具简介 .....	5
1.2.1 ARM SDT .....	5
1.2.2 ARM ADS .....	7
1.2.3 Multi 2000 .....	10
1.2.4 Embest IDE for ARM .....	14
1.2.5 OPENice32 - A900 仿真器 .....	15
1.2.6 Multi - ICE 仿真器 .....	16
1.2.7 ARM RealView 开发工具 .....	17
1.3 基于 ARM 的嵌入式系统开发学习 .....	18

## 第 2 章 Embest EDUKIT - II/III 实验教学系统

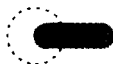
2.1 实验教学系统简介 .....	19
2.1.1 基本组成 .....	19
2.1.2 主要特点 .....	20
2.1.3 软、硬件配置 .....	21
2.1.4 Embest IDE for ARM 集成开发环境 .....	23
2.1.5 Embest Emulator for ARM JTAG 仿真器 .....	25
2.1.6 Flash 编程器 .....	26
2.1.7 各种连接线与电源适配器 .....	27
2.2 Embest EDUKIT - II/III 实验教学系统的安装 .....	27



2.2.1	Embest IDE for ARM 集成开发环境的安装	27
2.2.2	Flash 编程器的安装	30
2.2.3	实验软件平台和硬件平台的连接	31
2.3	Embest EDUKIT - II/III 实验教学系统硬件电路	31
2.3.1	功能模块介绍	32
2.3.2	硬件设计原理说明	33
2.3.3	硬件结构	52
2.3.4	硬件资源分配	60
2.4	Embest IDE for ARM 集成开发环境使用说明	66
2.4.1	Embest IDE for ARM 主窗口	66
2.4.2	工程的管理	67
2.4.3	工程的基本设置	69
2.4.4	工程的编译、链接	88
2.4.5	加载调试	88
2.4.6	Flash 编程工具	96

### 第3章 Linux 开发环境建立

3.1	建立 Linux 开发环境	98
3.1.1	Cygwin 开发环境	98
3.1.2	Linux 开发环境	102
3.1.3	Embest IDE 开发环境	102
3.1.4	文件传输服务器	102
3.1.5	文件准备	103
3.2	编译工具	103
3.2.1	交叉编译工具安装	103
3.2.2	环境变量及编译开关设置	104
3.3	源代码文件	106
3.3.1	VIVI 源代码安装	106
3.3.2	Linux 源代码安装	107
3.3.3	文件系统安装	107
3.3.4	带 Qt 的文件系统安装	107
3.4	实验例程	108
3.4.1	VIVI 实验	108
3.4.2	Linux 实验	108



3.4.3 文件系统实验 .....	108
3.4.4 Qt 实验 .....	109
3.5 环境设置及编译 .....	110
3.5.1 测试程序 .....	110
3.5.2 驱动程序 .....	112
3.5.3 busybox 工具编译 .....	113
3.5.4 VIVI 编译 .....	113
3.5.5 Linux 编译 .....	113
3.5.6 root 文件系统编译 .....	115
3.5.7 QTOPIA 编译 .....	115
3.6 烧写映像文件 .....	116
3.6.1 烧写 Nor Flash .....	118
3.6.2 烧写 Nand Flash .....	119
3.7 文档约定 .....	121
<b>第 4 章 Linux 启动代码编程实验</b>	
4.1 VIVI 介绍 .....	122
4.2 VIVI 编译及使用实验 .....	130
<b>第 5 章 Linux 内核基础实验</b>	
5.1 Linux 编译运行实验 .....	143
5.2 Linux 内核调试实验 .....	154
<b>第 6 章 Linux 驱动程序实验</b>	
6.1 LED 字符设备驱动程序实验 .....	167
6.2 LCD 显示驱动程序实验 .....	186
6.3 动态加载模块实验 .....	205
<b>第 7 章 Linux 文件系统及应用程序添加、调试实验</b>	
7.1 cramfs 文件系统 .....	213
7.2 FTP 应用程序调试实验 .....	221
7.3 NFS 方式调试应用程序实验 .....	225

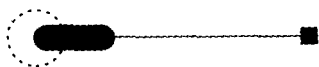


**第 8 章 Linux 应用程序设计实验**

8.1 串口通信实验 .....	230
8.2 I <sup>2</sup> C 总线接口实验 .....	235
8.3 多进程与进程间通信实验 .....	254
8.4 网络应用程序实验 .....	261
8.5 文件和目录操作实验 .....	268
8.6 实时时钟实验 .....	278
8.7 A/D 转换实验 .....	283
8.8 键盘和 LED 控制实验 .....	289
8.9 EEPROM 读/写实验 .....	300

**第 9 章 Linux 的 GUI 及综合应用实验**

9.1 Qt 实验 .....	310
9.2 Web 服务器实验 .....	325
附录 A ARM 指令、ARM 寻址方式和 Thumb 指令速查表 .....	330
附录 B ARM 和 Thumb 指令集编码 .....	337
附录 C 随书附带光盘内容说明 .....	340
参考文献 .....	343
温馨提示 .....	344





# 第 1 章

## 嵌入式系统开发与应用基础

在过去的几年里,嵌入式系统技术的推广和应用得到了飞速发展,目前它已经成为电子产品设计领域最为热门的实用技术之一,已经被广泛应用于消费电子、无线网络、工业控制、移动通信和军事国防等各个领域。

2004 年我国嵌入式系统应用产品生产总量估计超过 10 000 亿元。2004 年我国嵌入式微处理器销售总量大约为 13 亿片(与中国人口相当),据 WSTS 预测,2005 年全球嵌入式微处理器市场的产量将达 125 亿美元,包括 DVD 播放器、机顶盒、音视频接收设备、MP3 播放器、数码相机和汽车电子等。

ARM 处理器已经稳居世界 32/64 位处理器首位,是高端嵌入式系统开发的工业标准和首选。基于 ARM 的嵌入式系统及开发环境已经成为嵌入式系统技术最成功的代表,也是最具增长和潜力的领域,近年来受到广泛关注。由于产业的推动,基于 ARM 的嵌入式教学体系在国内比较容易实施,国内已经建立了相当数量的基于 ARM 的嵌入式实验室。

进入 21 世纪,集成电路和软件技术已经成为社会经济发展的重要支柱产业,嵌入式系统作为集成电路应用和软件的集成体,其内涵和外延也更加广泛。如何把这些新技术用于教学与科研实践,取得创新的研究成果,培养高水平的应用人才,是高等院校和科研单位始终关注的话题。

高校作为培养嵌入式技术人才的摇篮,虽然很多高校已经开设或准备开设嵌入式系统相关课程,但由于嵌入式系统设计是一门综合性非常强的学科,涉及微电子技术、计算机技术、智能控制技术和行业应用技术等诸多学科和相关课程内容,目前国内还没有建立一套完整的,以 ARM 技术为核心的嵌入式系统开发与应用的教学课程体系,各高校也普遍缺乏具有 32 位嵌入式系统教学及科研经验的教师和完善的嵌入式教学实验设备。

因此,根据目前国内的嵌入式教学现状,联合相关教学设备的研发和生产厂商,共同编写了基于 ARM 的嵌入式系列教材。其目的就是为了牢牢抓住嵌入式发展的难得机遇,加强产、学、研的联合、协作与互动,充实嵌入式人才发展的师资,完善设备,迎接未来嵌入式技术发展的挑战。

本章主要讲述如何构建基于 ARM 的嵌入式开发环境,各种基于 ARM 的嵌入式开发工具以及基于 ARM 嵌入式系统开发学习,这些都是后续内容学习的基础。



## 1.1 基于 ARM 的嵌入式开发环境概述

ARM 技术是高性能、低功耗嵌入式芯片的代名词,尤其是在基于嵌入式 Internet 方面应用广泛。因此,学习嵌入式系统的开发应用技术,应该在基于某种 ARM 核系统芯片应用平台的基础上进行。在介绍嵌入式系统开发应用之前,应该对基于 ARM 的嵌入式开发环境进行了解。本节主要对如何构造 ARM 嵌入式开发环境等基本情况介绍。

### 1.1.1 交叉开发环境

作为嵌入式系统应用的 ARM 处理器,其应用软件的开发属于跨平台开发,因此,需要一个交叉开发环境。交叉开发是指在一台通用计算机上进行软件的编辑编译,然后下载到嵌入式设备中运行调试的开发方式。用来开发的通用计算机可以选用比较常见的 PC、工作站等,运行通用的 Windows 或 Unix 操作系统。开发计算机一般称“宿主机”,嵌入式设备称为“目标机”。在主机上编译好的程序,下载到目标机上运行,交叉开发环境提供调试工具对目标机上运行的程序进行调试。

交叉开发环境一般由运行于宿主机上的交叉开发软件和宿主机到目标机的调试通道组成。

运行于宿主机上的交叉开发软件至少必须包含编译调试模块,其编译器为交叉编译器。宿主机一般为基于 x86 体系的台式计算机,而编译出的代码必须在 ARM 体系结构的目标机上运行,这就是所谓的交叉编译。在主机上编译好目标代码后,通过宿主机到目标机的调试通道将代码下载到目标机,然后由运行于宿主机的调试软件控制代码在目标机上运行调试。为了方便调试开发,交叉开发软件一般为一个整合编辑、编译汇编链接、调试、工程管理及函数库等功能模块的集成开发环境(Integrated Development Environment, IDE)。

组成 ARM 交叉开发环境的宿主机到目标机的调试通道一般有以下 3 种:

#### 1. 基于 JTAG 的 ICD

JTAG 的 ICD(In-Circuit Debugger)也称为 JTAG 仿真器,是通过 ARM 芯片的 JTAG 边界扫描口进行调试的设备。JTAG 仿真器通过 ARM 处理器的 JTAG 调试接口与目标机通信,通过并口/串口、网口、USB 口与宿主机通信。JTAG 仿真器价格比较便宜,连接比较方便。通过现有的 JTAG 边界扫描口与 ARM CPU 核通信,属于完全非插入式(即不使用片上资源)调试。它无需目标存储器,不占用目标系统的任何应用端口。通过 JTAG 方式可以完成:

- 读/写 CPU 的寄存器,访问控制 ARM 处理器内核;
- 读/写内存,访问系统中的存储器;
- 访问 ASIC 系统;