21 面向CENTURY 高等学校计算机类专业规划教材

ARM-Linux嵌入式系统开发基础

主　编　孙　弋
副主编　朱周华　刘新良

西安电子科技大学出版社
http://www.xduph.com

面向21世纪高等学校计算机类专业规划教材

ARM-Linux嵌入式系统开发基础

主　编　孙　弋

副主编　朱周华　刘新良

西安电子科技大学出版社

2008

内 容 简 介

本书针对 ARM-Linux 嵌入式系统开发进行了系统的介绍，并从产品开发的角度详细介绍了嵌入式产品开发的流程及相关基础知识。内容主要包括嵌入式产品开发流程、嵌入式开发环境、编程原理、Bootloader、内核配置及启动流程、文件系统等，并在最后一章以串口和 A/D 接口应用为例简述了数据传送设备的开发过程。

本书内容翔实丰富、结构合理，可作为高等院校嵌入式系统相关课程用书，同时也可以作为嵌入式认证的培训教材。

★本书配有电子教案，需要的老师可与出版社联系，免费提供。

图书在版编目(CIP)数据

ARM-Linux 嵌入式系统开发基础 / 孙弋主编. —西安：西安电子科技大学出版社，2008.8

面向 21 世纪高等学校计算机类专业规划教材

ISBN 978-7-5606-2072-5

Ⅰ. A… Ⅱ. 孙… ① 微处理器，ARM—高等学校—教材 ② Linux 操作系统—高等学校—教材

Ⅳ. TP332 TP316.89

中国版本图书馆 CIP 数据核字(2008)第 109121 号

策　　划　戚文艳

责任编辑　张　玮　戚文艳

出版发行　西安电子科技大学出版社(西安市太白南路 2 号)

电　　话　(029)88242885　88201467　邮　编　710071

http: //www.xduph.com　　E-mail: xdupfxb001@163.com

经　　销　新华书店

印刷单位　陕西天意印务有限责任公司

版　　次　2008 年 8 月第 1 版　2008 年 8 月第 1 次印刷

开　　本　787 毫米×1092 毫米　1/16　印 张 20.5

字　　数　485 千字

印　　数　1～4000 册

定　　价　29.00 元

ISBN 978 - 7 - 5606 - 2072 - 5/TP • 1067

XDUP 2364001-1

前　言

随着电子技术的发展和SOC片上系统应用领域的不断扩展，嵌入式系统已经在消费、电子、军事和医疗卫生中得到了广泛应用，并且将来会在更广泛的领域中占据更多的市场份额。

本书从嵌入式产品的基本开发流程、产品项目流程控制入手，对嵌入式系统的产品开发基础知识展开描述，期望通过本书使读者不仅了解嵌入式系统的基本概念，还能掌握一些产品的开发方法。

本书共分为8章，各章的主要内容如下：

第1章介绍了嵌入式系统的现状和主流的嵌入式操作系统，详细描述了嵌入式系统产品开发的特点和设计流程、嵌入式产品的软/硬件划分、硬件详细设计及软件设计，最后对嵌入式开发电路基础、电子电路抗干扰设计基础和嵌入式系统电源管理技术这三个方面进行了介绍。

第2章介绍了嵌入式微处理器的结构和类型，并以本书所用到的ARM9 S3C2410X微处理器为例，详细描述其各寄存器的特性，并以ViVi中的代码为例介绍其寄存器的设置。

第3章介绍了Linux C编译调试的基础知识，包括Vi和Emacs编辑器的使用、GNU GCC编译器的主要选项、GNU make的基本概念以及GDB调试工具使用的基本概念及技巧。

第4章主要针对Linux C的多线程编程的基本概念，具体包括线程、进程、进程间的通信机制及互斥锁、信号灯等进行了详细的描述。通过本章的学习，读者可基本了解Linux操作系统的多线程编程概念。

第5章描述了嵌入式系统开发中十分重要的Bootloader软件，介绍了Bootloader软件的基本原理和工作流程，并以ViVi和U-Boot为例详细介绍了相关Bootloader软件在ARM9 S3C2410X上的开发应用。

第6章和第7章是本书的核心部分，介绍了嵌入式Linux内核和嵌入式文件系统的基本概念和相关开发技术。第6章主要包括嵌入式Linux的版本控制和代码结构、ARM-Linux系统的内存管理、进程管理与调度、中断响应与处理机制、模块化机制、内核的配置、内核启动分析，并对内核移植作了简单描述。

针对嵌入式文件系统，第7章从文件系统的基本概念、嵌入式Linux的MTD驱动技术、虚拟文件系统等方面系统地描述了嵌入式文件系统，并详细描述了基于Flash存

储器的嵌入式文件系统 JFFS2、YAFFS、CramFS 的基本概念，最后着重描述了 Busybox 软件在嵌入式文件系统建立中的应用。

第 8 章针对目前应用中最常见的串口通信和 A/D 接口应用开发，以短距离无线传输具体的应用为例详细介绍了串口及 A/D 接口开发过程，并提供了具体代码和详细的开发流程，供读者参考。

本书由孙弋博士主编，朱周华和刘新良任副主编，廖晓群编写了本书部分内容，研究生刘南、任博涵、李剑桥、赵立军、孙媛媛等参与了编写并完成了部分文字录入及图表绘制。同时本书也参考了国内外众多学者的研究成果，在此表示感谢。

由于本书编者水平有限，疏漏在所难免，希望读者批评指正。

编　者

2008 年 5 月于西安

目　录

第 1 章　嵌入式系统开发基础

根据电气工程师协会的定义，嵌入式系统(Embedded System)是用来控制或者监视机器、装置、工厂等大规模系统的设备。一般认为嵌入式系统是以应用为中心，以计算机技术为基础，其软/硬件可裁剪，可满足应用系统对功能、可靠性、成本、体积、功耗的严格要求的专用计算机系统。嵌入式系统一般由嵌入式微处理器、外围硬件设备、嵌入式操作系统以及用户应用程序四个部分组成，用于实现对其他设备的控制、监视或管理等功能。

作为专用的计算机系统，嵌入式系统同 PC 系统相比具有以下特点：

(1) 嵌入式系统功耗低、体积小、专用性强。嵌入式系统与 PC 系统的最大不同就是嵌入式 CPU 大多工作在为特定用户群设计的系统中，能够把 PC 系统中许多由板卡完成的任务集成在芯片内部，从而使系统设计趋于小型化。

(2) 嵌入式系统中的软件一般都固化在存储器芯片或单片机芯片中，以提高执行速度和系统可靠性。

(3) 嵌入式系统的硬件和软件都经过精心设计，系统精简，其操作系统一般和应用软件集成在一起。

(4) 软件代码质量要求高。

(5) 嵌入式系统开发需要专门的开发工具和开发环境。

近年来微电子技术迅猛发展，嵌入式处理器的性能速度也随之有很大的提高，嵌入式系统领域发生了翻天覆地的变化。特别是网络的普及，嵌入式与互联网成为热门的应用领域。技术的进步可以使嵌入式系统具备网络功能，并将它们与 Internet 或企业内联网连接起来。这种特性增强了嵌入式系统多方面的实用性，也进一步扩展了嵌入式系统的应用领域。

美国著名的未来学家尼葛洛庞帝曾预言，嵌入式系统是继 PC 和 Internet 之后最伟大的发明。如今该领域的发展验证了这个预言的正确性。现在嵌入式系统正处于高速发展阶段，未来几年，这种发展和竞争将愈演愈烈。

将嵌入式系统的应用按照市场领域划分，可以分为以下几类：

- 消费类电子产品。
- 控制系统和工业自动化。
- 机器人领域。
- 数据/无线通信。

在企业专用解决方案方面，如物流管理、条码扫描、移动信息采集等领域中，小型手

持嵌入式系统也将发挥巨大的作用。在自动控制领域，嵌入式系统不仅可以用于ATM机、自动售货机、工业控制等专用设备，并且和移动通信设备、GPS、娱乐相结合后可以发挥更大的作用。

嵌入式系统由于硬件的限制，其硬件资源较为紧张，如CPU主频较低、内存较小、以小容量的Flash存储器替代磁盘等。在使用电池的系统中，嵌入式系统还要实现低功耗，因而具备较长时间的续航能力。

1.1 嵌入式系统概述

1.1.1 典型的嵌入式操作系统

目前国际上使用的嵌入式操作系统主要可分为实时操作系统和非实时操作系统两类。对于时间要求严格的系统，可称之为实时系统。实时系统的一个重要特点就是对时间要求非常严格。如果实时系统不能在某个预定的时间内响应某个事件，系统将会出错。特别是航天、军工领域必须使用实时操作系统。

从20世纪80年代开始，市场上出现各种各样的商用嵌入式操作系统，这些操作系统大部分都是为专有系统开发的，从而逐步演化成了现在多种形式的商用嵌入式操作系统百家争鸣的局面。这些操作系统有Linux、μC/OS、Windows CE、VxWorks、Palm OS和QNX等。

1. Linux

在所有的操作系统中，Linux是发展最快、应用最广泛的系统之一。Linux本身的种种特性使其成为嵌入式开发的首选。在进入市场的前两年中，嵌入式Linux的设计通过广泛应用而获得巨大的成功。随着嵌入式Linux技术的成熟，以其按应用要求可定制系统、支持多数硬件平台等特性，已由早期的试用阶段迈进到逐渐成为嵌入式市场的主流。

根据IDC的报告，Linux已经成为全球第二大操作系统。Linux发展如此之快的另一个主要原因是产品的成本。在激烈的市场竞争中，只拥有先进的技术是远远不够的，如何减少产品的投入也是需要重点考虑的问题。免费的Linux为厂商节约了一大笔开支，特别是对于经济实力不强的公司来说。

目前Linux内核的最新版本已经达到2.6.xx。

2. μC/OS

μC/OS是一个典型的实时操作系统。该系统从1992年开始发展，目前流行的是第二个版本，即μC/OS Ⅱ。其特点可以概括为以下几个方面：公开源代码，代码结构清晰、明了，注释详细，组织有条理，可移植性好，可裁剪，可固化，内核属于抢占式，最多可以管理60个任务。自从清华大学邵贝贝教授将Jean J.Labrosse的《μC/OS: the Real Time Kernel》一书翻译后，在国内掀起μC/OS Ⅱ的学习热潮，特别是在教育研究领域(μC/OS系统在教育研究领域是免费的)。该系统短小精悍，是研究和学习实时操作系统的首选。

3. Windows CE

Windows CE 是微软公司的产品，是从整体上为有限资源的平台设计的多线程、完整优先权、多任务的操作系统。Windows CE 采用模块化设计，并对于从掌上电脑到专用的工控电子设备进行定制。此操作系统的基本内核需要至少 200 KB ROM 存储器。从游戏机到现在大部分的高价掌上电脑都采用了 Windows CE 作为操作系统，其缺点是系统软件价格过高，影响整个产品的成本控制。

4. VxWorks

VxWorks 是 WindRiver(风河)公司专门为实时嵌入式系统设计开发的操作系统软件，为程序员开发提供了高效的实时任务调度、中断管理、实时的系统资源以及实时的任务间通信。应用程序员可以将尽可能多的精力放在应用程序本身，而不必再去关心系统资源的管理。该系统主要应用在单板机、数据网络(以太网交换机、路由器)和通信等多方面。

5. Palm OS

Palm OS 是一种 32 位的嵌入式操作系统，用于掌上电脑。此系统是 3Com 公司的 PalmComputing 部开发的(Palm Computing 目前已经独立成为一家公司)，它运行在一个抢占式的多任务内核之上，同一时刻用户界面仅仅允许一个应用程序被打开，与同步软件 Hotsync 结合可以使掌上电脑与 PC 上的信息实现同步，把台式机的功能扩展到了手掌上。同其他嵌入式操作系统相比，Palm OS 具有更大的灵活性和移动性，是一款非常流行的掌上电脑操作系统。

6. QNX

QNX 是一款实时操作系统，由加拿大 QNX 软件系统有限公司开发，广泛应用于自动化、控制、机器人科学、电信、数据通信、航空航天、计算机网络系统、医疗仪器设备、交通运输、安全防卫系统、POS 机、零售机等任务关键型应用领域。20 世纪 90 年代后期，QNX 系统在高速增长的 Internet 终端设备、信息家电及掌上电脑等领域也得到了广泛应用。

1.1.2　Linux 系统

Linux 在 1991 年诞生于芬兰。大学生 Linus Torvalds 由于没有足够的钱购买昂贵的商用操作系统，于是自己编写了一个小的操作系统内核，这就是 Linux 的前身。Linus Torvalds 将操作系统的源代码在 Internet 上公布，受到了计算机爱好者的热烈欢迎。各种各样的计算机高手不断地为它添加新的特性，并不断地提高它的稳定性。1994 年，Linux 1.0 正式发布。现在，Linux 已经成为一个功能强劲的 32 位操作系统。

1. 嵌入式 Linux 的特点

嵌入式系统以应用为中心，以计算机为基础，软/硬件可裁剪，适用于对功能、可靠性、成本、功耗严格要求的专用计算机系统。实时性是嵌入式系统的基本要求，嵌入式 Linux 是指将 Linux 经过裁剪小型化后固化在存储器中，应用于特定嵌入式场合的专用 Linux 操作系统。嵌入式 Linux 的开发和研究已经成为目前操作系统领域的一个热点。专用嵌入式操作系统与嵌入式 Linux 操作系统的比较见表 1-1。

表 1-1 专用嵌入式实时操作系统与嵌入式 Linux 操作系统的比较

	专用嵌入式实时操作系统	嵌入式 Linux 操作系统
版权费	每生产一件产品需交纳一份版权费	免费
购买费用	数十万元(RMB)	免费
技术支持	由开发商独家提供有限的技术支持	全世界的自由软件开发者提供支持
网络特性	另加数十万元(RMB)购买	免费且性能优异
软件移植	难(因为是封闭系统)	易，代码开放(有许多应用软件支持)
应用产品开发周期	长(因为可参考的代码有限)	短，新产品上市迅速(因为有许多公开的代码可以参考和移植)
实时性能	好	须改进，可用 PT_Linux 等模块加以弥补
稳定性	较好	较好，但在高性能系统中须改进

Linux 的特点如下：

第一，Linux 系统具有层次结构且内核完全开放。Linux 是由很多体积小且性能高的微内核系统组成的。在内核代码完全开放的前提下，不同领域和不同层次的用户可以根据应用需求方便地对内核进行改造，低成本地设计和开发出满足需要的嵌入式系统。

第二，强大的网络支持功能。Linux 诞生于 Internet 时代并具有 Unix 的特性，保证了它支持所有标准 Internet 协议，并且可以利用 Linux 的网络协议栈将其开发成为嵌入式的 TCP/IP 网络协议栈。此外，Linux 还支持 Ext2、FAT16、FAT32、Romfs 等文件系统，为开发嵌入式系统应用打下了很好的基础。

第三，Linux 具备一整套工具链，自行建立嵌入式系统的开发环境和交叉运行环境，可以跨越嵌入式系统开发中仿真工具的障碍。Linux 也符合 IEEE POSIX.1 标准，使应用程序具有较好的可移植性。

传统的嵌入式开发的程序调试和调试工具是用在线仿真器(ICE)实现的。它通过取代目标板的微处理器，给目标程序提供一个完整的仿真环境，完成监视和调试程序；但一般价格比较昂贵，只适合做非常底层的调试。使用嵌入式 Linux，一旦软/硬件能够支持正常的串口功能，即使不用仿真器，也可以通过串口很好地进行开发和调试工作，从而节省一笔不小的开发费用。嵌入式 Linux 为开发者提供了一套完整的工具链(Tool Chain)。它利用 GNU 的 GCC 做编译器，用 GDB、KGDB、XGDB 做调试工具，能够很方便地实现从操作系统到应用软件各个级别的调试。

第四，Linux 具有广泛的硬件支持特性。无论是 RISC 还是 CISC，32 位还是 64 位等各种处理器平台，Linux 都能很好地运行。Linux 通常使用的微处理器是 Intel X86 芯片家族，但它同样能运行于 Motorola 公司的 68K 系列 CPU 和 IBM、Apple、Motorola 公司的 PowerPC CPU，以及 Intel 公司的 StrongARM CPU 等处理器。Linux 支持各种主流硬件设备和最新硬件技术，甚至可以在没有存储管理单元(MMU)的处理器上运行。这意味着嵌入式 Linux 将具有更广泛的应用前景。

2. 系统软件操作平台

Linux 作为嵌入式操作系统是完全可行的，因为 Linux 提供了完成嵌入功能的基本内核和所需要的所有用户界面，能处理嵌入式任务和用户界面。Linux 作为嵌入式系统，具有易

移植、内核稳定、功能强大、易于开发等特点。

嵌入式 Linux 系统需要三个基本元素：系统引导(用于机器加电后的系统定位引导)、Linux 微内核(内存管理、程序管理)、初始化进程。但假如要成为完整的操作系统并且保持小型化，还必须加上硬件驱动程序、硬件接口程序和应用程序组。

Linux 系统开发一般基于 GNU 的 GCC 编译器，作为 GNU 工具链的一部分，与 GDB 源调试器一起工作。它提供了开发嵌入式 Linux 系统的所有软件工具。

3. 嵌入式 Linux 系统开发模式

嵌入式系统通常是一个资源受限的系统，直接在嵌入式系统的硬件平台上编写软件比较困难，有时甚至是不可能的。目前，一般采用的办法是，先在通用计算机上编写程序，然后通过交叉编译，生成目标平台上可运行的二进制代码格式，最后下载到目标平台上的特定位置上运行，具体步骤如下：

(1) 建立嵌入式 Linux 交叉开发环境。目前，常用的交叉开发环境主要有开放和商业两种类型。开放的交叉开发环境的典型代表是 GNU 工具链，目前已经能够支持 X86、ARM、MIPS、PowerPC 等多种处理器。商业的交叉开发环境主要有 Metrowerks CodeWarrior、ARM Software Development Toolkit、SDS Cross Compiler、WindRiver Tornado、Microsoft Embedded Visual C 等。交叉开发环境是指编译、链接和调试嵌入式应用软件的环境，它与运行嵌入式应用软件的环境有所不同，通常采用宿主机/目标机模式。

(2) 交叉编译和链接。在完成嵌入式软件的编码之后，就要进行编译和链接，以生成可执行代码。由于开发过程大多是在 Intel 公司 X86 系列 CPU 的通用计算机上进行的，而目标环境的处理器芯片却大多为 ARM、MIPS、PowerPC、DragonBall 等系列的微处理器，这就要求在建立好的交叉开发环境中进行交叉编译和链接。

例如，在基于 ARM 体系结构的 GCC 交叉开发环境中，arm-linux-gcc 为交叉编译器，arm-linux-ld 为交叉链接器。通常情况下，并不是每一种体系结构的嵌入式微处理器都只对应于一种交叉编译器和交叉链接器。如对于 M68K 体系结构的 GCC 交叉开发环境而言，就对应于多种不同的编译器和链接器。假如使用的是 COFF 格式的可执行文件，那么在编译 Linux 内核时，需要使用 m68k-coff-gcc 和 m68k-coff-ld，而在编译应用程序时则需要使用 m68k-coff-pic-gcc 和 m68k-coff-pic-ld。编写好的嵌入式软件经过交叉编译和交叉链接后，通常会生成两种类型的可执行文件：用于调试的可执行文件和用于固化的可执行文件。

(3) 交叉调试。

① 硬件调试。假如不采用在线仿真器，可以让 CPU 直接在其内部实现调试功能，并通过在开发板上引出的调试端口发送调试命令和接收调试信息，完成调试过程。目前，Motorola 公司提供的开发板上使用的是 DBM 调试端口，而 ARM 公司提供的开发板上使用的则是 JTAG 调试端口。使用合适的软件工具与这些调试端口进行连接，可以获得与 ICE 类似的调试效果。

② 软件调试。在嵌入式 Linux 系统中对 Linux 系统内核进行调试时，可以先在 Linux 内核中设置一个调试桩(Debug Stub)，用作调试过程中和宿主机之间的通信服务器。然后，可以在宿主机中通过调试器的串口与调试桩进行通信，并通过调试器控制目标机上 Linux 内核的运行。

嵌入式上层应用软件的调试可以使用本地调试和远程调试两种方法。假如采用的是本

地调试，首先要将所需的调试器移植到目标系统中，然后就可以直接在目标机上运行调试器来调试应用程序了；假如采用的是远程调试，则需要移植一个调试服务器到目标系统中，并通过它与宿主机上的调试器共同完成应用程序的调试。在嵌入式 Linux 系统的开发中，远程调试时目标机上使用的调试服务器通常是 GDB Server，而宿主机上使用的调试器则是 GDB，两者相互配合共同完成调试过程。

(4) 系统测试。在整个软件系统编译过程中，嵌入式系统的硬件一般采用专门的测试仪器进行测试，而软件则需要有相关的测试技术和测试工具支持，并要采用特定的测试策略。测试技术指的是软件测试的专门途径，以及能够更加有效地运用这些途径的特定方法。在嵌入式软件测试中，经常要在基于目标机的测试和基于宿主机的测试之间做出折中。基于目标机的测试需要消耗较多的开发成本，而基于宿主机的测试虽然代价较小，但是在仿真环境中进行的，难以完全反映软件运行时的实际情况。从这两种环境下的测试可以发现不同的软件缺陷，从而可对目标机环境和宿主机环境下的测试内容进行合理取舍。嵌入式软件测试中经常用到的测试工具主要有内存分析工具、性能分析工具、覆盖分析工具、缺陷跟踪工具等，在这里不加详述。

以下即为一个典型开发工具的使用流程：

① 写入或植入引导代码。

② 编写串口打印字符串代码。

③ 将 GDB 目标码移植至工作串口，可与另一台运行 GDB 程序的 Linux 主机系统对话。

④ 利用 GDB 使硬件和软件初始化代码在 Linux 内核启动时工作。

⑤ Linux 内核启动，串口成为 Linux 控制口并用于后续开发。

⑥ 假如在目标硬件上已运行了完整的 Linux 内核，即可开始调试用户的应用进程。

1.2 嵌入式系统设计基础

1.2.1 嵌入式系统设计的特点

嵌入式系统设计的主要任务是定义系统的功能、决定系统的架构，并将功能映射到架构。这里的架构既包括软件系统架构也包括硬件系统架构。嵌入式系统的设计方法跟一般的硬件设计、软件开发的方法不同，是采用硬件和软件协同设计的方法，开发过程涉及软件领域的知识、硬件领域的综合知识，甚至还涉及机械等方面的知识，设计者必须熟悉并能自如地运用这些领域的各种技术，才能使设计的系统达到最优。

与通常的系统设计相比，嵌入式系统的设计具有以下几个特点：

(1) 软/硬件协同并行开发。软/硬件协同并行开发就是在整个设计的生命周期内，软件和硬件的设计一直保持并行，在设计过程中两者交织在一起，互相支持，互相提供开发的平台。传统方法中则将软/硬件的设计分开独立进行，然后独立进行硬件和软件设计，最后才进行软/硬件的集成。系统是否满足用户需求只有等到软/硬件集成之后进行测试才能知道。

(2) 嵌入式系统是面向特定应用的系统。嵌入式 CPU 与通用型 CPU 最大的不同在于，嵌入式 CPU 大多工作在为特定用户群设计的系统中，它通常都具有低功耗、小体积、高集

成度等特点，能够把通用 PC 中许多由板卡完成的任务集成在芯片内部，从而有利于嵌入式系统设计趋于小型化，同时跟网络的耦合也越来越紧密。

(3) 嵌入式系统设计采用交叉开发环境。嵌入式系统的开发通常采用“宿主机/目标机”交叉开发方式(如图 1-1 所示)。首先，利用宿主机上丰富的资源以及良好的开发环境来开发和仿真调试目标机上的软件，生成编译好的目标文件，再通过 JTAG 口、UTAR 接口或者以太网接口将交叉编译生成的目标代码传输并下载到目标机上，并用交叉调试器在实时内核/操作系统或监控程序的支持下进行实时分析和调试，最后，目标机在目标环境下运行。

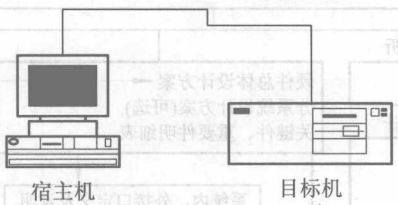


图 1-1　宿主机/目标机开发方式

宿主机(Host)是一台通用计算机，一般由 PC 机承担。它通过串行接口或网络连接与目标机进行通信。宿主机上的软/硬件资源比较丰富，具有各种各样的开发调试工具，如 GNU 的嵌入式开发工具套件等。这些辅助的开发工具和丰富的硬件资源能够保证系统的开发效率。

目标机(Target)常在嵌入式系统的开发过程期间使用。目标机可以是嵌入式系统的实际运行环境，也可以是能替代实际环境的仿真系统。通常目标机的体积较小，集成度高，外围设备丰富，备有键盘、触摸屏等，以及 LCD、LED 等输入/输出设备。由于目标机的硬件资源有限，在嵌入式系统目标机上运行的软件通常需要根据具体应用进行裁减和配置。

1.2.2　嵌入式系统的设计流程

嵌入式系统的设计和开发流程一般分为以下几个阶段：产品定义(即系统需求分析阶段、规格说明阶段)、硬件和软件划分、迭代与实现、详细的硬件与软件设计、硬件与软件集成、系统测试、系统维护与升级。各个阶段通常需要不断的反复和修改，直到完成最终设计目标。

电子信息产品系统的设计流程如图 1-2(a)所示。

对于嵌入式产品的开发与研制可以完全按照电子产品的开发流程进行，通过图 1-2(b)所示的文档管理体系实现产品开发的嵌入式系统的设计过程，在各阶段内部及各阶段之间会发生大量的迭代与优化，前一步的设计缺陷会直接导致后面阶段的设计任务无法完成，从而必须重新进行产品定义设计，这就会造成生产成本的提高，增加产品的开发时间。主要原因在于前期的需求分析工作一定要精细，确保准确无误，尽量减少开发过程中的需求变更。下面对各阶段给出具体的描述。

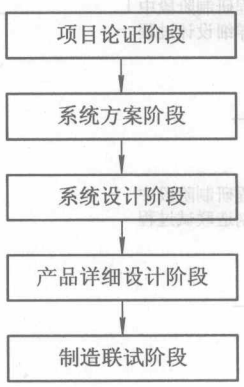
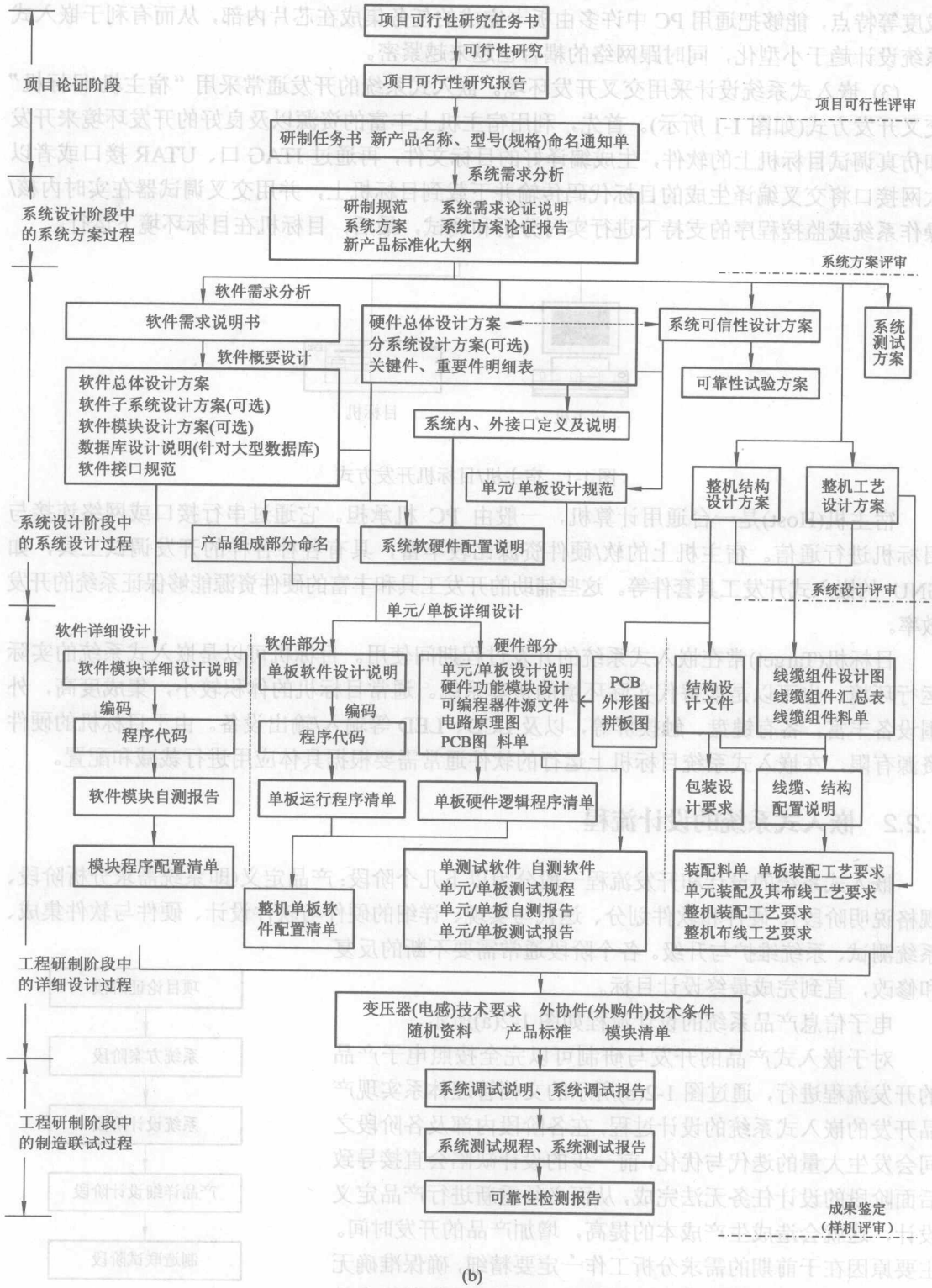


图 1-2(a)　嵌入式系统的设计流程



(b)

图 1-2(b)　电子信息产品研发过程文档体系

1. 项目论证阶段

项目论证阶段需要完成项目的可行性分析并形成可行性研究报告。

2. 系统方案阶段

项目论证评审后进入系统方案阶段，本阶段对产品需求加以分析、细化，并抽象出需要完成的功能列表，明确定义所要完成的任务。由于嵌入式设计分为硬件与软件设计两个部分，设计人员必须确定系统的哪些功能由硬件完成，哪些功能由软件完成，这种选择称为“划分决策”，即软件与硬件的划分。本阶段最终形成产品研发的系统方案、研制规范、需求分析说明和系统方案论证报告。

产品功能包括系统的基本功能，如输入/输出、操作方式等；产品的性能相当于一般软件工程的非功能性需求，它包括系统性能、成本、功耗、体积、重量等因素。为了使产品定义更加清晰明确，可以使用一个产品需求分析表格(如表 1-2 所示)，它将系统的功能和性能需求综合起来。

表 1-2 需求分析表格

项 目	描 述
名 称	对于所完成项目的总体描述
目 的	满足系统要求的描述
输入/输出	系统输入/输出的数据类型、数据特征和输入/输出的设备类型
功 能	详细描述系统功能、系统数据的流向等
性 能	系统所要求处理的速度、实时性和实用性
生产成本	主要是硬件构件和人员人力成本
功 耗	依靠电池供电的嵌入式系统设备必须考虑功耗问题
结 构	最终的产品因为使用的领域不同而存在很大差异，对系统的结构设计有一定的了解，有助于对系统体系结构的设计

本阶段的需求分析工作对于产品的开发成败至关重要，因此必须严格控制产品需求变更，任何的需求变更都可能导致研发周期和研发成本的巨大增加。

3. 系统设计阶段

在通过系统方案评审后，进入系统设计阶段，软件开发部分完成软件需求分析，形成软件需求分析说明书。经过软件概要设计后，形成软件总体设计方案、软件开发接口规范等文档。硬件部分完成硬件总体设计方案(包括产品可信性设计)、系统内外接口定义及说明、单板设计规范(复杂产品)等。

4. 产品详细设计阶段

产品设计阶段主要是完成软/硬件的详细设计，编制代码，形成软件各模块的设计说明，完成硬件部分各单板的原理图、PCB 和料单，同时还要完成产品的结构设计。

5. 制造联试阶段

完成产品详细设计后，进入产品制造联试阶段，本阶段完成产品的系统调试和可靠性测试，并形成相应的系统调试报告和可靠性测试报告。

经过以上阶段，产品样机的研制即完成。

此后将进入产品的小批量生产和中试阶段，以及产品的生产流程，这不再是本书的研