



嵌入式系统原理实验教程

—— ARM体系结构

QIANRUSHI XITONG YUANLI SHIYAN JIAOCHENG
ARM TIXI JIEGOU

卢冶 编著

南开大学出版社

嵌入式系统原理实验教程

——ARM 体系结构

卢冶 编著

常州大学图书馆
藏书章

南开大学出版社

天津

嵌入式系统原理实验教程：ARM 体系结构 / 卢冶编

ARM 体系结构

图书在版编目 (CIP) 数据

嵌入式系统原理实验教程：ARM 体系结构 / 卢冶编
著. — 天津：南开大学出版社，2019.5
ISBN 978-7-310-05791-7

I . ①嵌… II . ①卢… III . ①微处理器—系统设计—
教材 IV . ① TP332.3

中国版本图书馆 CIP 数据核字 (2019) 第 081879 号

版权所有 侵权必究

南开大学出版社出版发行

出版人：刘运峰

地址：天津市南开区卫津路 94 号 邮政编码：300071

营销部电话：(022)23508339 23500755

营销部传真：(022)23508542 邮购部电话：(022)23502200

*

天津午阳印刷股份有限公司印刷

全国各地新华书店经销

*

2019 年 5 月第 1 版 2019 年 5 月第 1 次印刷

185×260 毫米 16 开本 14.5 印张 312 千字

定价：36.00 元

如遇图书印装质量问题，请与本社营销部联系调换，电话：(022)23507125

目 录

第 1 章 引言	1
1.1 编写目的	1
1.2 本书特色	1
第 2 章 嵌入式系统概论	2
2.1 嵌入式系统的概念	2
2.2 嵌入式系统的组成及处理器介绍	3
2.3 嵌入式操作系统	5
2.4 嵌入式 Linux 操作系统	9
第 3 章 嵌入式 ARM 技术概论	19
3.1 ARM 体系结构的技术特征及发展	19
3.2 ARM 微处理器结构特征与数据类型	20
3.3 存储系统	24
3.4 协处理器(CP15)	25
3.5 存储管理单元(MMU)	26
3.6 高速缓冲存储器(Cache)	26
3.7 流水线	27
3.8 寄存器组织	29
第 4 章 微处理器指令系统与混合编程	35
4.1 ARM 处理器的寻址方式	35
4.2 ARM 汇编语言的程序结构	39
4.3 混合编程	43
第 5 章 ARM 硬件开发平台概述	49
5.1 Cortex-A53 处理器概述	49
5.2 S5P6818 应用处理器	50
5.3 OURS-S5P6818 实验平台简介	52
第 6 章 ARM 裸机系统汇编实验	64
6.1 ARM 开发环境搭建	64
6.2 S5P6818 启动分析	64
6.3 通过 TF 卡运行程序	76
6.4 ARM 汇编控制蜂鸣器实验	83

6.5	ARM 汇编控制 LED 闪烁	104
6.6	ARM 汇编按键控制蜂鸣器	109
6.7	其他实验	115
第 7 章	ARM 裸机系统 C 语言实验	116
7.1	C 程序 LED 流水灯	116
7.2	C 程序蜂鸣器控制	124
7.3	C 程序复位控制	126
7.4	C 程序按键控制 LED	128
7.5	C 程序按键控制 LED 和蜂鸣器	129
7.6	C 程序 LED 模拟心脏跳动	131
7.7	C 程序按键中断	132
7.8	C 程序串口 shell	133
7.9	C 程序串口输入实验	134
7.10	C 程序移植 printf 函数实验	135
第 8 章	嵌入式 Linux 实验	137
8.1	编译 U-Boot 实验	137
8.2	编译 Kernel 实验	146
8.3	其他实验	164
第 9 章	嵌入式 Linux Qt 编程	165
9.1	Linux 系统 tslib 库移植实验	165
9.2	其他实验	197
第 10 章	嵌入式 Android 开发	198
10.1	继电器控制实验	198
10.2	字模表	206
10.3	综合实验	223
参考文献	224
致谢	225

第 1 章 引言

1.1 编写目的

本书实验平台采用北京奥尔斯教育科技有限公司的 S5P6818 嵌入式原理教学实验系统,采用编程与移植裁剪等方法,对嵌入式系统各个层次和完整系统开发进行实验,使学生加深对嵌入式系统原理和应用开发的理解,提高学生动手实践能力。

本书从嵌入式 ARM 体系结构的基础知识、系统环境构建到系统分层应用,分为三个阶段深入浅出地为读者指明嵌入式 ARM 体系结构及其实验应用的方法,引领读者进行嵌入式系统开发实践。书中内容涉及 ARM 裸机操作、汇编与 C 语言、Linux 系统裁剪与移植、驱动开发、QT 移植、Android 系统移植与应用等。书中对实验原理、实验环境、实验步骤、实验结果做出了详尽的讲解和说明,以期学生能够更好地了解并掌握嵌入式系统设计与开发的方法。

1.2 本书特色

重基础,适合教学,讲解全面。本书在一般性教材的基础上,对嵌入式系统的软硬件开发环境、系统搭建、功能开发等进行了大量的讲解,可以让读者更进一步、更全面地了解嵌入式的开发过程。

重实践,与实际项目相结合。本书按照嵌入式系统层次结构,对应提供了丰富的实验内容,并且实验原理和实验内容一一对应,每个知识点在深入理解后可运用于实践,本书附带参考设计代码和说明文档,访问网页(<http://ics.nankai.edu.cn/lemledded/Ev1.0.pdf>)可下载。

重应用,书中的实例针对常用、普遍的设备、软件环境、工具等进行了详细的讲解和说明,力求教程所涉及的内容能够紧密结合行业实际应用的需要。

第 2 章 嵌入式系统概论

电子数字计算机诞生于 1946 年,在其后漫长的历史进程中,计算机始终被供养在特殊的机房中,是实现数值计算的大型昂贵设备。直到 20 世纪 70 年代,随着微处理器的出现,计算机才出现了历史性的变化。以微处理器为核心的微型计算机以其小型、价廉、高可靠性特点,迅速走出机房;基于高速数值解算能力的微型机,以其智能化水平引起了控制专业人士的兴趣,被嵌入到一个个对象体系中,实现对象体系的智能化控制。例如,将微型计算机经电气加固、机械加固,并配置各种外围接口电路,安装到大型舰船中,构成自动驾驶仪或轮机状态监测系统。这样一来,计算机便失去了原来的形态与通用的计算机功能。为了区别于原有的通用计算机系统,把嵌入到对象体系中,实现对象体系智能化控制的计算机,称作嵌入式计算机系统。

2.1 嵌入式系统的概念

根据国际电气与电子工程师学会(IEEE)的定义,嵌入式系统是“控制、监视或者辅助装置、机器和设备运行的装置”。这主要是从应用上加以定义的,从中可以看出嵌入式系统是软件和硬件的综合体。不过上述定义并不能充分体现嵌入式系统的精髓,目前国内一个普遍被认同的定义是:以应用为中心,以计算机技术为基础,软件硬件可裁剪,适应应用系统对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。简单地说,嵌入式系统集成系统的应用软件与硬件于一体,类似于 PC 中 BIOS 的工作方式,具有软件代码小、高度自动化、响应速度快等特点,特别适合于要求实时和多任务的体系。嵌入式系统主要由嵌入式处理器、相关支撑硬件、嵌入式操作系统及应用软件系统等组成,它是可独立工作的“器件”。

在明确了嵌入式系统定义基础上,我们可从以下几方面来理解嵌入式系统。

①嵌入式系统是面向用户、面向产品、面向应用的,嵌入式系统是与应用紧密结合的,它具有很强的专用性,必须结合实际系统需求进行合理的裁剪利用。嵌入式系统和具体应用有机地结合在一起,它的升级换代也是和具体产品同步进行,因此嵌入式系统产品一旦进入市场,具有较长的生命周期。

②嵌入式系统是将先进的计算机技术、半导体技术和电子技术与各个行业的具体应用相结合后的产物。这一点就决定了它必然是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。

③嵌入式系统必须根据应用需求对软硬件进行裁剪,满足应用系统的功能、可靠性、成本、体积等要求。为了提高执行速度和系统可靠性,嵌入式系统中的软件一般都固化在存储器芯片或单片机本身中,而不是存贮于磁盘等载体中。

④嵌入式系统本身不具备自主开发能力,即使设计完成以后用户通常也是不能对其中

的程序功能进行修改的,必须有一套开发工具和环境才能进行开发。

实际上,凡是与产品结合在一起的具有嵌入式特点的控制系統都可以叫嵌入式系統。现在人们讲嵌入式系統时,某种程度上指近些年比较热门的具有操作系统的嵌入式系統。

2.2 嵌入式系统的组成及处理器介绍

嵌入式系统是计算机软件和硬件的综合体,可涵盖机械或其他附属装置。所以嵌入式系統可以笼统地分为硬件和软件两部分。嵌入式系統的构架可以分成四个部分:处理器、存储器、输入输出(I/O)和软件(由于多数嵌入式设备的应用软件和操作系统都是紧密结合的,在这里我们对其不加区分,这也是嵌入式系統和通用PC系統的最大区别)。嵌入式系統的组成如图2-1所示。

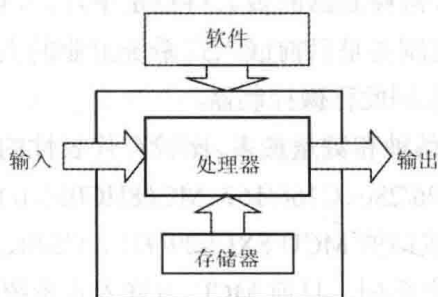


图 2-1 嵌入式系统组成

嵌入式系統的硬件部分,包括处理器/微处理器、存储器及外设器件和 I/O 端口、图形控制器等。嵌入式系統有别于一般的计算机处理系統,它不具备像硬盘那样大容量的存储介质,而大多使用 EPROM、EEPROM 或闪存(Flash Memory)作为存储介质。

嵌入式系統的核心部件是各种类型的嵌入式处理器,目前据不完全统计,全世界嵌入式处理器的品种总量已经超过 1000 多种,流行体系结构有二三十个系列。现在几乎每个半导体制造商都生产嵌入式处理器,越来越多的公司有自己的处理器设计部门。嵌入式处理器的寻址空间一般从 64 KB 到 16 MB~32 MB,处理速度从 0.1 MIPS 到 2000 MIPS。根据其现状,嵌入式计算机可以分成下面几类。

1. 嵌入式微处理器(Embedded Microprocessor Unit, EMPU)

嵌入式微处理器的基础是通用计算机中的 CPU。在应用中,将微处理器装配在专门设计的电路板上,只保留和嵌入式应用有关的母板功能,这样可以大幅度减小系統体积和功耗。为了满足嵌入式应用的特殊要求,嵌入式微处理器虽然在功能上和标准微处理器基本是一样的,但在工作温度、抗电磁干扰、可靠性等方面一般都做了各种增强。

和工业控制计算机相比,嵌入式微处理器具有体积小、重量轻、成本低、可靠性高的优点,但是在电路板上必须包括 ROM、RAM、总线接口、各种外设等器件,从而降低了系統的可靠性,技术保密性也较差。嵌入式微处理器及其存储器、总线、外设等安装在一块电路板上,称为单板计算机,如 STD-BUS、PC104 等。近年来,德国、日本的一些公司又开发出了类

似“火柴盒”式名片大小的嵌入式计算机系列 OEM 产品。

嵌入式处理器目前主要有 AML86/88、386EX、SC-400、Power PC、68000、MIPS、ARM 等系列。

2. 嵌入式微控制器(Microcontroller Unit, MCU)

嵌入式微控制器又称单片机,顾名思义,就是将整个计算机系统集成到一块芯片中。嵌入式微控制器一般以某一种微处理器内核为核心,芯片内部集成 ROM/EPROM、RAM、总线、总线逻辑、定时 / 计数器、WatchDog、I/O、串行口、脉宽调制输出、A/D、D/A、FlashRAM、EEPROM 等各种必要功能和外设。为适应不同的应用需求,一般一个系列的单片机具有多种衍生产品,每种衍生产品的处理器内核都是一样的,不同的是存储器和外设的配置及封装。这样可以使单片机最大限度地和应用需求相匹配,功能不多不少,从而减少功耗和成本。

和嵌入式微处理器相比,微控制器的最大特点是单片化,体积大大减小,从而使功耗和成本下降、可靠性提高。微控制器是目前嵌入式系统工业的主流。微控制器的片上外设资源一般比较丰富,适合于控制,因此称微控制器。

嵌入式微控制器目前的品种和数量最多,比较有代表性的通用系列包括 ARM、8051、P51XA、MCS-251、MCS-96/196/286、C166/167、MC68HC05/11/12/16、68290 等。另外还有许多半通用系列,如支持 USB 接口的 MCU 8XC929/931、C540、C541,支持 I2C、CAN-Bus、LCD 及众多专用 MCU 和兼容系列。目前 MCU 占嵌入式系统约 70% 的市场份额。

特别值得注意的是近年来提供 X86 微处理器的著名厂商 AMD 公司,将 AML86CC/CH/CU 等嵌入式处理器称之为 Microcontroller, MOTOROLA 公司把以 Power PC 为基础的 PPC505 和 PPC555 也列入单片机行列。TI 公司也将其 TMS320C2XXX 系列 DSP 作为 MCU 进行推广。

3. 嵌入式 DSP 处理器(Embedded Digital Signal Processor, EDSP)

DSP 处理器对系统结构和指令进行了特殊设计,使其适合于执行 DSP 算法,编译效率较高,指令执行速度也较高。在数字滤波、FFT、谱分析等方面 DSP 算法正在大量进入嵌入式领域, DSP 应用正从在通用单片机中以普通指令实现 DSP 功能,过渡到采用嵌入式 DSP 处理器。嵌入式 DSP 处理器有两个发展来源:一是 DSP 处理器经过单片化、EMC 改造、增加片上外设成为嵌入式 DSP 处理器, TI 的 TMS320C2000/C5000 等属于此范畴;二是在通用单片机或 SOC 中增加 DSP 协处理器,例如 INTEL 的 MCS-286 和 SIEMENS 的 TriCore。

推动嵌入式 DSP 处理器发展的另一个因素是嵌入式系统的智能化,例如各种带有智能逻辑的消费类产品,生物信息识别终端,带有加解密算法的键盘, ADSL 接入、实时语音解压系统,虚拟现实显示等。这类智能化算法一般都是运算量较大,特别是向量运算、指针线性寻址等较多,而这些正是 DSP 处理器的长处所在。

嵌入式 DSP 处理器比较有代表性的产品是 TI 公司的 TMS320 系列和 MOTOROLA 公司的 DSP56000 系列。TMS320 系列处理器包括用于控制的 C2000 系列,移动通信的 C5000 系列,以及性能更高的 C6000 和 C8000 系列。DSP56000 目前已经发展成为

DSP56000、DSP56100、DSP56200 和 DSP56290 等几个不同系列的处理器。另外 PHILIPS 公司今年也推出了基于可重置 SP 结构低成本、低功耗技术上制造的 REAL DSP 处理器,特点是具备双 Harvard 结构和双乘 / 累加单元,应用目标是大批量消费类产品。

4. 嵌入式片上系统(System On Chip, SOC)

随着 EDA 的推广和 VLSI 设计的普及化,及半导体工艺的迅速发展,在一个硅片上实现一个更为复杂的系统的时代已来临,这就是 SOC。各种通用处理器内核将作为 SOC 设计公司的标准库,和许多其他嵌入式系统外设一样,成为 VLSI 设计中一种标准的器件,用标准的 VHDL 等语言描述,存储在器件库中。用户只须定义出其整个应用系统,仿真通过后就可以将设计图交给半导体工厂制作样品。这样除个别无法集成的器件以外,整个嵌入式系统大部分均可集成到一块或几块芯片中去,应用系统电路板将变得很简洁,对于减小体积和功耗、提高可靠性非常有利。

SOC 可以分为通用和专用两类。通用系列包括 SIEMENS 的 TriCore、MOTOROLA 的 M-Core、某些 ARM 系列器件、ECHELON 和 MOTOROLA 联合研制的 Neuron 芯片等。专用 SOC 一般专用于某个或某类系统中,不为一般用户所知。一个有代表性的产品是 PHILIPS 的 Smart XA,它将 XA 单片机内核和支持超过 2048 位复杂 RSA 算法的 CCU 单元制作在一块硅片上,形成一个可加载 JAVA 或 C 语言的专用的 SOC,可用于公众互联网如 Internet 安全方面。

2.3 嵌入式操作系统

嵌入式操作系统(Embedded Operating System, EOS)是指用于嵌入式系统应用的操作软件。嵌入式操作系统是一种用途广泛的系统软件,是嵌入式系统(包括硬、软件系统)极为重要的组成部分,通常包括与硬件相关的底层驱动软件、系统内核、设备驱动接口、通信协议、图形界面、标准化浏览器等。嵌入式操作系统具有通用操作系统的基本特点,如负责嵌入式系统的全部软、硬件资源的分配、任务调度,控制、协调并发活动。与通用操作系统相比较,嵌入式操作系统在系统实时高效性、硬件的相关依赖性、软件固态化以及应用的专用性等方面具有较为突出的特点。它必须体现其所在系统的特征,能够通过装卸某些模块来达到系统所要求的功能。目前在嵌入式领域广泛使用的操作系统:嵌入式实时操作系统 μ C/OS-II、嵌入式 Linux、Windows Embedded、VxWorks 等,以及应用在智能手机和平板电脑的 Android、iOS 等。

2.3.1 嵌入式操作系统的种类

一般情况下,嵌入式操作系统可以分为两类:一类是面向控制、通信等领域的实时操作系统,如 WindRiver 公司的 VxWorks、ISI 的 pSOS、QNX 系统软件公司的 QNX、ATI 的 Nucleus 等;另一类是面向消费电子产品的非实时操作系统,这类产品包括个人数字助理(PDA)、移动电话、机顶盒、电子书、WebPhone 等,如 Apple 公司的 iOS、Google 公司的

Android 等。

1. 非实时操作系统(如图 2-2 所示)

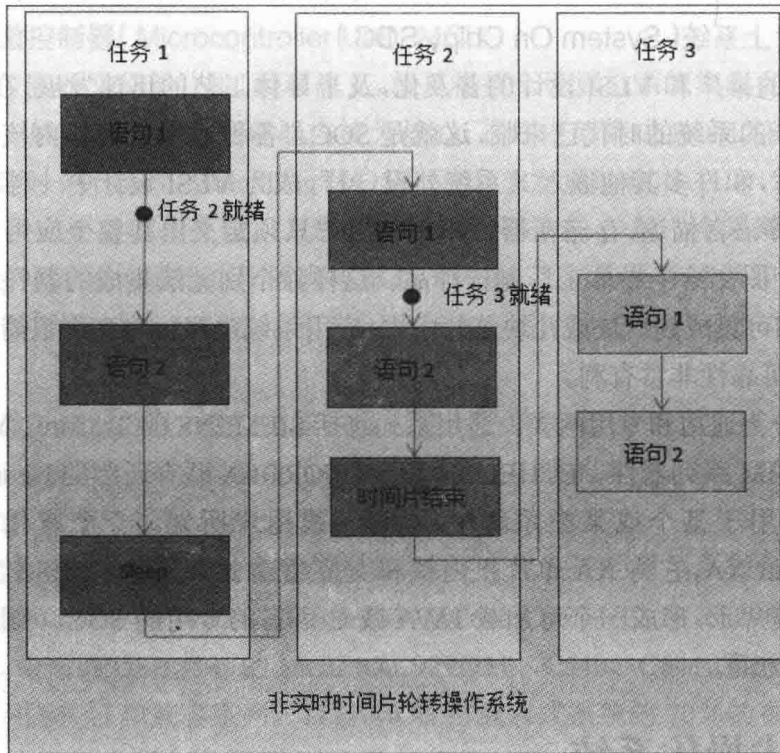


图 2-2 非实时操作系统任务调度

早期的嵌入式系统中没有操作系统的概念,程序员编写嵌入式程序通常直接面对裸机及裸设备。在这种情况下,通常把嵌入式程序分成两部分,即前台程序和后台程序。前台程序通过中断来处理事件,其结构一般为无限循环;后台程序则掌管整个嵌入式系统软、硬件资源的分配、管理以及任务的调度,是一个系统管理调度程序。这就是通常所说的前后台系统。

一般情况下,后台程序也叫任务级程序,前台程序也叫事件处理级程序。在程序运行时,后台程序检查每个任务是否具备运行条件,通过一定的调度算法来完成相应的操作。对于实时性要求特别严格的操作通常由中断来完成,仅在中断服务程序中标记事件的发生,不再做任何工作就退出中断,经过后台程序的调度,转由前台程序完成事件的处理,这样就不会造成在中断服务程序中处理费时的事件而影响后续和其他中断。

实际上,前后台系统的实时性比预计的要差。这是因为前后台系统认为所有的任务具有相同的优先级别,即是平等的,而且任务的执行又是通过 FIFO 队列排队,因而对那些实时性要求高的任务不可能立刻得到处理。另外,由于前台程序是一个无限循环的结构,一旦在这个循环体中正在处理的任務崩溃,使得整个任务队列中的其他任务也得不到机会被处理,从而造成整个系统的崩溃。由于这类系统结构简单,几乎不需要 RAM/ROM 的额外开销,因而使简单的嵌入式应用被广泛使用。

2. 实时操作系统(如图 2-3 所示)

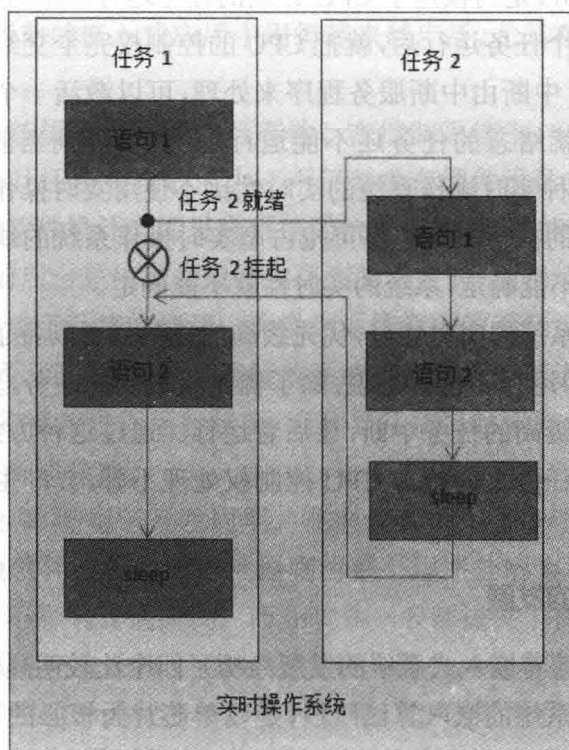


图 2-3 实时操作系统任务调度

实时操作系统是指能在确定的时间内执行其功能并对外部的异步事件做出响应的计算机系统。其操作的正确性不仅依赖于逻辑设计的正确程度,而且与这些操作进行的时间有关。“在确定的时间内”是该定义的核心。也就是说,实时系统是对响应时间有严格要求的。

实时操作系统对逻辑和时序的要求非常严格,如果逻辑和时序出现偏差将会引起严重后果。实时系统有两种类型:软实时系统和硬实时系统。软实时系统仅要求事件响应是实时的,并不要求限定某一任务必须在多长时间内完成;而在硬实时系统中,不仅要求任务响应要实时,而且要求在规定的时间内完成事件的处理。通常,大多数实时系统是两者的结合。实时应用软件的设计一般比非实时应用软件更困难。实时系统的技术关键是如何保证系统的实时性。

实时多任务操作系统是指具有实时性、能支持实时控制系统工作的操作系统。其首要任务是调度一切可利用的资源完成实时控制任务,其次才着眼于提高计算机系统的使用效率,重要特点是要满足对时间的限制和要求。实时操作系统具有如下功能:任务管理(多任务和基于优先级的任务调度)、任务间同步和通信(信号量和邮箱等)、存储器优化管理(含 ROM 的管理)、实时时钟服务、中断管理服务。实时操作系统具有如下特点:规模小,中断被屏蔽的时间很短,中断处理时间短,任务切换很快。

实时操作系统可分为可抢占型和不可抢占型两类。对于基于优先级的系统而言,可抢

占型实时操作系统是指内核可以抢占正在运行任务的 CPU 使用权并将使用权交给进入就绪态的优先级更高的任务,是内核抢了 CPU 让别的任务运行。不可抢占型实时操作系统使用某种算法并决定让某个任务运行后,就把 CPU 的控制权完全交给了该任务,直到它主动将 CPU 控制权还回来。中断由中断服务程序来处理,可以激活一个休眠态的任务,使之进入就绪态;而这个进入就绪态的任务还不能运行,一直要等到当前运行的任务主动交出 CPU 的控制权。使用这种实时操作系统的实时性比不使用实时操作系统的系统性能好,其实时性取决于最长任务的执行时间。不可抢占型实时操作系统的缺点也恰恰是这一点,如果最长任务的执行时间不能确定,系统的实时性就不能确定。

可抢占型实时操作系统的实时性好,优先级高的任务只要具备了运行的条件,或者说进入了就绪态,就可以立即运行。也就是说,除了优先级最高的任务,其他任务在运行过程中都可能随时被比它优先级高的任务中断,让后者运行。通过这种方式的任务调度保证了系统的实时性,但是,如果任务之间抢占 CPU 控制权处理不好,会产生系统崩溃、死机等严重后果。

2.3.2 嵌入式操作系统的发展

嵌入式操作系统伴随着嵌入式系统的发展经历了四个比较明显的阶段。

第一阶段是无操作系统的嵌入算法阶段,是以单芯片为核心的可编程控制器形式的系统,同时具有与监测、伺服、指示设备相配合的功能。这种系统大部分应用于一些专业性极强的工业控制系统中,一般没有操作系统的支持,通过汇编语言编程对系统进行直接控制,运行结束后清除内存。这一阶段系统的主要特点是:系统结构和功能都相对单一,处理效率较低,存储容量较小,几乎没有用户接口。由于这种嵌入式系统使用简便、价格很低,以前在国内工业领域应用较为普遍,但是已经远远不能适应高效的、需要大容量存储介质的现代化工业控制和新兴的信息家电等领域的需求。

第二阶段是以嵌入式 CPU 为基础、以简单操作系统为核心的嵌入式系统。这一阶段系统的主要特点是:CPU 种类繁多,通用性比较差;系统开销小,效率高;一般配备系统仿真器,操作系统具有一定的兼容性和扩展性;应用软件较专业,用户界面不够友好;系统主要用来控制系统负载以及监控应用程序运行。

第三阶段是通用的嵌入式操作系统阶段,是以嵌入式操作系统为核心的嵌入式系统。这一阶段系统的主要特点是:嵌入式操作系统能运行于各种不同类型的微处理器上,兼容性好;操作系统内核精小、效率高,并且具有高度的模块化和扩展性;具备文件和目录管理、设备支持、多任务、网络支持、图形窗口以及用户界面等功能;具有大量的应用程序接口(API),开发应用程序简单;嵌入式应用软件丰富。

第四阶段是以基于 Internet 为标志的嵌入式系统,这是一个正在迅速发展的阶段。目前大多数嵌入式系统还孤立于 Internet 之外,但随着 Internet 的发展以及 Internet 技术与信息家电、工业控制技术相结合日益密切,嵌入式设备与 Internet 的结合将代表着嵌入式技术的真正未来。

2.3.3 使用嵌入式操作系统的必要性

嵌入式操作系统在目前的嵌入式应用中用得越来越广泛,尤其在功能复杂、系统庞大的应用中显得愈来愈重要。

一是,嵌入式操作系统提高了系统的可靠性。在控制系统中,出于安全方面的考虑,要求系统起码不能崩溃,而且还要有自愈能力。不仅要求在硬件设计方面提高系统的可靠性和抗干扰性,而且也应在软件设计方面提高系统的抗干扰性,尽可能地减少安全漏洞和不可靠的隐患。长期以来的前后台系统软件设计在遇到强干扰时,使得运行的程序产生异常、出错、跑飞,甚至死循环,造成了系统的崩溃。而操作系统管理的系统,这种干扰可能只是引起若干进程中的一个被破坏,可以通过系统运行的系统监控进程对其进行修复。通常情况下,这个系统监视进程用来监视各进程运行状况,遇到异常情况时采取一些利于系统稳定可靠的措施,如把有问题的任务清除掉。

二是,提高了开发效率,缩短了开发周期。在嵌入式操作系统环境下,开发一个复杂的应用程序,通常可以按照软件工程中的解耦原则将整个程序分解为多个任务模块。每个任务模块的调试、修改几乎不影响其他模块。商业软件一般都提供了良好的多任务调试环境。

三是,嵌入式操作系统充分发挥了 CPU 的多任务潜力。它本来是为运行多用户、多任务操作系统而设计的,特别适于运行多任务实时系统。CPU 采用利于提高系统可靠性和稳定性的设计,使其更容易做到不崩溃。例如,CPU 运行状态分为系统态和用户态。将系统堆栈和用户堆栈分开,以及实时地给出 CPU 的运行状态等,允许用户在系统设计中从硬件和软件两方面对实时内核的运行实施保护。如果还是采用以前的前后台方式,则无法发挥 CPU 的优势。

从某种意义上说,没有操作系统的计算机(裸机)是没有用的。在嵌入式应用中,只有把 CPU 嵌入到系统中,同时又把操作系统嵌入进去,才是真正的计算机嵌入式应用。

2.3.4 嵌入式操作系统的优缺点

在嵌入式操作系统环境下开发应用程序使程序的设计和扩展变得容易,不需要大的改动就可以增加新的功能。通过将应用程序分割成若干独立的任务模块,使应用程序的设计过程大为简化;而且对实时性要求苛刻的事件都得到了快速、可靠的处理。通过有效的系统服务,嵌入式操作系统使得系统资源得到更好的利用。

但是,使用嵌入式实时操作系统还需要额外的 ROM/RAM 开销,2%~5% 的 CPU 额外负荷,以及内核的费用。

2.4 嵌入式 Linux 操作系统

Linux 的出现,最早开始于一位名叫莱纳斯·托瓦兹(Linus Torvalds)的计算机业余爱好者,当时他是芬兰赫尔辛基大学的学生。他的目的是想设计一个代替 Minix(是由 Andrew Tannebaum 的计算机教授编写的一个操作系统示教程)的操作系统,这个操作系统可用于

386、486 或奔腾处理器的个人计算机上,并且具有 Unix 操作系统的全部功能,因而开始了 Linux 雏形的设计。

嵌入式 Linux 操作系统是嵌入式操作系统的一个成员,其最大的特点是源代码公开并且遵循 GPL 协议,近几年来已成为研究热点。目前正在开发的嵌入式系统中,有近 50% 的项目选择 Linux 作为嵌入式操作系统。

嵌入式 Linux 操作系统是将日益流行的 Linux 操作系统进行裁剪修改,使之能在嵌入式计算机系统上运行的一种操作系统。嵌入式 Linux 既继承了 Internet 上无限的开放源代码资源,又具有嵌入式操作系统的特性。

嵌入式 Linux 的特点是版权免费,购买费用仅仅是媒介成本,技术支持由全世界的自由软件开发者提供,支持网络特性,免费而且性能优异,软件移植容易,代码开放,有许多应用软件支持,应用产品开发周期短,新产品上市迅速,因为有许多公开的代码可以参考和移植,实时性能由 RT Linux、Hardhat Linux 等嵌入式 Linux 支持,实时性能稳定性好,安全性好。

嵌入式 Linux 的应用领域非常广泛,主要的应用领域有信息家电、PDA、机顶盒、Digital Telephone、Answering Machine、Screen Phone、数据网络、Ethernet Switches、Router、Bridge、Hub、Remote access servers、ATM、Frame relay、远程通信、医疗电子、交通运输计算机外设、工业控制、航空航天领域等。

Linux 做嵌入式的优势:第一, Linux 是开放源代码的,不存在黑箱技术,遍布全球的众多 Linux 爱好者又是 Linux 开发者的强大技术支持。第二, Linux 的内核小、效率高,内核的更新速度很快, Linux 是可以定制的,其系统内核最小只有约 134KB。第三, Linux 是免费的操作系统,在价格上极具竞争力。Linux 还有着嵌入式操作系统所需要的很多特色,突出的就是 Linux 适应于多种 CPU 和多种硬件平台,是一个跨平台的系统。到目前为止,它可以支持二三十种 CPU。而且性能稳定,裁剪性很好,开发和使用都很容易。很多 CPU 包括家电业芯片,都开始做 Linux 的平台移植工作。移植的速度远远超过 Java 的开发环境。也就是说,如果今天用 Linux 环境开发产品,那么将来换 CPU 就不会遇到困扰。同时, Linux 内核的结构在网络方面是非常完整的, Linux 对网络中最常用的 TCP/IP 协议有最完备的支持。提供了包括十兆、百兆、千兆的以太网,以及无线网络, Token ring (令牌环网)、光纤甚至卫星的支持。所以 Linux 很适于做信息家电的开发。

还有使用 Linux 的原因是无线连接产品的开发者越来越多。Linux 在快速增长的无线连接应用主场中有一个非常重要的优势,就是有足够快的开发速度。这是因为 Linux 有很多工具,并且 Linux 为众多程序员所熟悉。因此,我们要在嵌入式系统中使用 Linux 操作系统。

Linux 的大小适合嵌入式操作系统——Linux 固有的模块性、适应性和可配置性,使得这很容易做到。另外, Linux 源码的实用性使得成千上万的程序员热切期望它用于无数的嵌入式应用软件中,从而导致很多嵌入式 Linux 的出现,包括 Embedix、ETLinux、LEM Linux Router Project、LOAF μ CLinux、muLinux、ThinLinux、FirePlug Linux 和 PizzaBox Linux。

相比微软, Linux 的图形界面发展很快,像 GNOME、KDE、UTITY 等都是很优秀的桌面

管理器,针对嵌入式应用, Linux 有 QT 桌面管理器,并且其背后有着众多的社团支持,可定制性强,已经在 Unix 和 Linux 世界普及开来。

传统的嵌入式系统厂商也采用了 Linux 策略,如 Lynxworks Windriver QNX 等,还有 Internet 上的大量嵌入式 Linux 爱好者的支持。嵌入式 Linux 支持几乎所有的嵌入式 CPU 和被移植到几乎所有的嵌入式 OEM 板。

2.4.1 Linux 介绍

从应用上讲, Linux 一般有四个主要部分:内核、Shell、文件结构和实用工具。

1. Linux 内核

内核是系统的核心,是运行程序和管理像磁盘和打印机等硬件设备的核心程序。它从用户那里接受命令并把命令送给内核去执行。

2. Linux Shell

Shell 是系统的用户界面,提供了用户与内核进行交互操作的一种接口。它接收用户输入的命令并把它送入内核去执行。

实际上 Shell 是一个命令解释器,它解释由用户输入的命令并且把它们送到内核。不仅如此, Shell 有自己的编程语言用于对命令的编辑,它允许用户编写由 Shell 命令组成的程序。Shell 编程语言具有普通编程语言的很多特点,比如它也有循环结构和分支控制结构等,用这种编程语言编写的 Shell 程序与其他应用程序具有同样的效果。

Linux 提供了像 Microsoft Windows 那样的可视的命令输入界面——X Window 的图形用户界面(GUI)。它提供了很多窗口管理器,其操作就像 Windows 一样,有窗口、图标和菜单,所有的管理都是通过鼠标控制。现在比较流行的窗口管理器是 KDE 和 GNOME。

每个 Linux 系统的用户可以拥有他自己的用户界面或 Shell,用以满足他们自己专门的 Shell 需要。同 Linux 本身一样,Shell 也有多种不同的版本。目前主要有下列版本的 Shell。

- ① Bourne Shell: 是贝尔实验室开发的。
- ② BASH: 是 GNU 的 Bourne Again Shell,是 GNU 操作系统上默认的 Shell。
- ③ Korn Shell: 是对 Bourne Shell 的发展,在大部分内容上与 Bourne Shell 兼容。
- ④ C Shell: 是 SUN 公司 Shell 的 BSD 版本。

3. Linux 文件结构

文件结构是文件存放在磁盘等存储设备上的组织方法,主要体现在对文件和目录的组织上。目录提供了管理文件的一个方便而有效的途径。我们能够从一个目录切换到另一个目录,而且可以设置目录和文件的权限,设置文件的共享程度。

使用 Linux,用户可以设置目录和文件的权限,以便允许或拒绝其他人对其进行访问。Linux 目录采用多级树形结构,图 2-4 表示了这种树形等级结构。用户可以浏览整个系统,可以进入任何一个已授权进入的目录,访问那里的文件。

文件结构的相互关联性使共享数据变得容易,几个用户可以访问同一个文件。Linux 是一个多用户系统,操作系统本身的驻留程序存放在以根目录开始的专用目录中,有时被指

定为系统目录。图 2-4 中那些根目录下的目录就是系统目录。

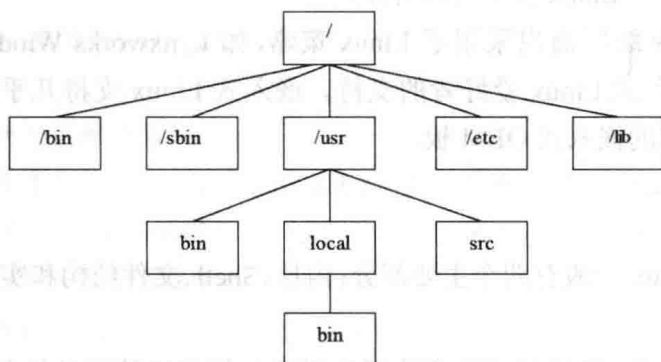


图 2-4 Linux 的目录结构

内核、Shell 和文件结构一起形成了基本的操作系统结构。它们使得用户可以运行程序、管理文件以及使用系统。此外，Linux 操作系统还有许多被称为实用工具的程序，辅助用户完成一些特定的任务。

4. Linux 实用工具

标准的 Linux 系统都有一套叫作实用工具的程序，它们是专门的程序，例如编辑器、执行标准的计算操作等。用户也可以产生自己的工具。

实用工具可分三类。

①编辑器：用于编辑文件。

②过滤器：用于接收数据并过滤数据。

③交互程序：允许用户发送信息或接收来自其他用户的信息。

Linux 的编辑器主要有 Ed、Ex、Vi 和 Emacs。Ed 和 Ex 是行编辑器，Vi 和 Emacs 是全屏幕编辑器。

Linux 的过滤器(Filter)读取从用户文件或其他地方的输入，检查和处理数据，然后输出结果。从这个意义上说，它们过滤了经过它们的数据。Linux 有不同类型的过滤器，一些过滤器用行编辑命令输出一个被编辑的文件。另外一些过滤器是按模式寻找文件并以这种模式输出部分数据。还有一些执行字处理操作，检测一个文件中的格式，输出一个格式化的文件。过滤器的输入可以是一个文件，也可以是用户从键盘键入的数据，还可以是另一个过滤器的输出。过滤器可以相互连接，因此，一个过滤器的输出可能是另一个过滤器的输入。在有些情况下，用户可以编写自己的过滤器程序。

交互程序是用户与机器的信息接口。Linux 是一个多用户系统，它必须和所有用户保持联系。信息可以由系统上的不同用户发送或接收。信息的发送有两种方式：一种方式是与其他用户一对一地链接进行对话；另一种是一个用户对多个用户同时链接进行通信，即所谓广播式通信。