

AKA 自由、协作、创造

源码开放的 嵌入式系统软件分析与实践

——基于 **SkyEye** 和 **ARM** 开发平台

陈渝 李明 杨晔 等编著



北京航空航天大学出版社

源码开放的
嵌入式系统软件分析与实践
——基于 SkyEye 和 ARM 开发平台

陈 渝 李 明 杨 晔 等编著

北京航空航天大学出版社

内 容 简 介

本书详细介绍了如何使用各种基于 GPL 协议和 Open Source 协议的开发工具和软件,并通过这些软件来研究、移植并开发 Linux、 μ Clinux 和 μ C/OS-II 操作系统以及相关系统软件的过程,同时对多种典型嵌入式开发板上的操作系统和系统软件的移植进行了详细分析和描述。通过介绍 SkyEye 硬件模拟平台的内部实现和具体使用,让读者可以在没有实际嵌入式硬件开发板的情况下,学习、研究和开发嵌入式操作系统和系统软件。

书中所有例子都在相关的嵌入式硬件开发板上或 SkyEye 硬件模拟平台上调试过。书中附光盘 1 张,包含与本书相关的硬件文档、软件工具、操作系统和系统软件,可作为读者进行具体实践的补充。本书面向嵌入式系统软件开发初学者和中高级开发人员,具有较强的系统性和实用性,可作为高等院校相关专业的教学参考书籍以及嵌入式系统开发人员的参考资料,也可以作为嵌入式系统开发的培训教材。

图书在版编目(CIP)数据

源码开放的嵌入式系统软件分析与实践:基于 SkyEye

和 ARM 开发平台/陈渝等编著. —北京:北京航空航天大学出版社,2004.9

ISBN 7-81077-514-6

I. 源… II. 陈… III. 微处理器—系统开发

IV. TP332

中国版本图书馆 CIP 数据核字(2004)第 088667 号

源码开放的嵌入式系统软件分析与实践

——基于 SkyEye 和 ARM 开发平台

陈 渝 李 明 杨 晔 等编著

责任编辑 朱伟锋

*

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:787×1092 1/16 印张:33 字数:845 千字

2004 年 9 月第 1 版 2004 年 9 月第 1 次印刷 印数:4 000 册

ISBN 7-81077-514-6 定价:48.00 元(含光盘 1 张)

前 言

本书特点

嵌入式系统软件开发是目前信息技术领域非常具有发展前景的方向之一。随着微电子技术、计算机技术等飞速发展,目前基于 32 位嵌入式微处理器的计算机应用正处于迅猛发展的时期。然而,相关的嵌入式软件的开发却远远滞后于硬件的发展,尤其是精通嵌入式系统软件开发的人才在世界范围内,尤其是在国内还很稀少。这主要是由以下一些原因造成的:第一,开发嵌入式系统软件一般需要嵌入式硬件设备的支持,而对于初学者(特别是学生)而言,购置嵌入式硬件设备的开销较难以承受。第二,目前在学习嵌入式系统开发的过程中,大多缺少比较全面的实践过程,大部分读者只停留在对书本所介绍的知识的了解阶段,导致实际动手能力较差和缺乏实践经验。第三,在台式机和服务器领域,主流软件开发基本上围绕着 Intel x86 CPU,以及 Microsoft 的 Windows 平台或 Linux 平台,而目前在嵌入式领域,各种嵌入式系统千差万别,没有一个占绝大多数的开发平台,这导致初学者很难选择一个利于入门且资源丰富的开发环境。

针对上述问题,我们通过嵌入式系统开发的实践过程,总结出嵌入式系统开发需要掌握的知识和经验,编写了这本书。与以往单纯介绍嵌入式系统知识的丛书相比,本书更注重实践性,它具有如下一些特点:

第一,本书提供了模拟嵌入式环境的仿真开发平台——SkyEye,使得读者在没有嵌入式硬件的情况下,只要有一台 PC 机,就可以通过 SkyEye 硬件模拟平台进行嵌入式软件的学习、调试与开发,从而可以真正从实践中了解和掌握嵌入式系统的开发知识。

第二,在各式各样的 32 位嵌入式 CPU 中,基于 ARM CPU 内核的嵌入式微处理器所占的比例是最高的,而且它基于 RISC 架构,利于学习和掌握。为此,本书选择基于 ARM CPU 内核的嵌入式系统作为我们研究、移植和开发的 CPU 核心,提供了多达 6 种基于 ARM CPU 内核的常见开发板的操作系统和相关系统软件的分析 and 移植过程,包含了许多开发者宝贵的移植方法和经验。

第三,本书选择 $\mu\text{C}/\text{OS}-\text{II}$ 和 Linux 作为我们研究的嵌入式操作系统。前者是一个简单易懂的实时嵌入式操作系统,目前被国内多所大学用于教学和科研。尤其是在清华大学的邵贝贝教授翻译了 Jean J. Labrosse 所著的《 $\mu\text{C}/\text{OS}-\text{II}$: the Real Time Kernel》一书之后,更是推动了 $\mu\text{C}/\text{OS}-\text{II}$ 和嵌入式操作系统学习在中国的普及。至于 Linux,它是功能强大的操作系统,横跨了服务器、台式机和嵌入

式系统 3 个硬件领域,是全世界的电脑爱好者共同的智慧结晶。它们都是开放源代码的,因此读者可以深入地分析和了解它们的实现细节,从而可以更深入地了解嵌入式操作系统的核心技术,而这是其他一些商业操作系统无法比拟的。

第四,本书选择基于 GPL 协议和 Open Source 协议的高质量软件——GNU Tools 作为我们的嵌入式开发工具,本书介绍的操作系统和系统软件实例都是用 GNU Tools 开发的。目前许多人认为基于命令行的 GNU Tools 软件不好用,质量没有保证,愿意使用一些商业开发工具软件,例如 Microsoft 的 Visual Studio 等。而我们认为,如果要开发面向嵌入式领域的系统级软件或操作系统,选择 GNU Tools 有很多优势。目前 GNU Tools 支持的嵌入式 CPU 种类是最多的,远远超过其他的嵌入式开发工具。而且目前还有多种高质量的 IDE 集成开发环境都支持 GNU Tools,这样开发人员也不需要直接使用基于命令行的 GNU Tools。掌握了 GNU Tools,就可以以不变应万变。

第五,本书配有光盘,光盘上有本书介绍到的各种硬件文档、嵌入式软件开发工具、嵌入式操作系统和本书作者实现的软件实例。读者在阅读本书的时候,可以参考相关的硬件文档,并通过光盘提供的软件进行实践。

内容组织

本书由基础篇和实践篇构成,第 1 章~第 7 章为基础篇,主要以我们在嵌入式领域的开发经验为指导,介绍了在嵌入式系统开发过程中需要使用的各种开发工具和开发技巧,为下一步在具体的硬件板上进行开发打下基础。第 8 章~第 12 章为实践篇,主要介绍了在各种具体的开发板和 SkyEye 上移植、运行和调试操作系统和系统软件的过程。具体内容如下。

基础篇

1. 第 1 章介绍嵌入式系统的基本知识和嵌入式开发的简要过程。
2. 第 2 章讲述 GNU Tools 的使用,特别是 ld、gcc、gdb、make 的使用细节。
3. 第 3 章讲述 SkyEye 嵌入式系统模拟环境的具体实现以及扩展和使用 SkyEye 的方法。
4. 第 4 章介绍 Boot Loader 的基本概念和框架结构,特别对 Armboot 和 RedBoot 进行了深入的分析。
5. 第 5、6 章讲述 $\mu\text{C}/\text{OS}-\text{II}$ 和 Linux 的基础知识、移植这两种操作系统的基本方法、文件系统的基本知识和安装方法以及对相关应用的移植分析。并讲解了如何使用 SkyEye 运行面向具体开发板的操作系统和应用。
6. 第 7 章讲述嵌入式系统的硬件调试、软件调试等,包括基于 JTAG 方式的 JTAGGER 调试、GDB 远程调试、KGDB 远程内核调试、KDB 内核调试、printk 调试方式等多种技术。

实践篇

7. 第8章以基于 Intel StrongARM CPU 的开发板为例,全面详细地讲解对 $\mu\text{C}/\text{OS}-\text{II}$ 和 ARM Linux 的移植过程和方法。从而让读者对嵌入式系统软件开发过程有一个全面和比较细致的了解。
8. 第9、10、11、12章分别以基于 Samsung s3c4510、Samsung 44B0、Atmel AT91X40、Intel XScale PXA 255、Cirrus Logic EP7312 等 CPU 的开发板为例,详细讲解 $\mu\text{C}/\text{OS}-\text{II}$ 、 μClinux 或 ARM Linux 的移植过程和方法。部分章节还讲解如何使用 SkyEye 来模拟相关的硬件,并在其上运行和调试操作系统。

阅读建议

为了更好地阅读本书,我们希望读者熟悉 C 语言,对计算机体系结构、ARM 体系结构和汇编指令有一定的了解,能够掌握 Linux 的基本使用方法。由于本书中部分内容与硬件相关,为此我们在提到特定硬件时,给出了必要的相关信息。同时随本书提供的光盘中有更详细的硬件相关文档,有助于读者更深入地分析相关硬件。

本书的大多数实验都是在 Linux 上运行的,所以熟悉使用 Linux 是能够实践本书讲解内容的基础。如果您是嵌入式系统的初学者,建议先看看附录,学习 Linux 的常用命令。如果读者从未使用过 Linux,建议先利用 VMware for Windows 软件在 Windows 操作系统上安装一个 Linux 操作系统,然后在 VMware 环境下的 Linux 操作系统中学习基本的 Linux 使用方法。建议读者一边看本书,一边参照本书的内容在 Linux 系统上进行各种实践。如果没有真实的开发板,则可以用 SkyEye 代替开发板进行实践。

如果你已经熟悉了 Linux 环境,建议的阅读顺序为:先读第1章,了解嵌入式开发的基本知识,然后阅读第2、3章了解嵌入式开发的一些常用软件工具,之后阅读第5章和第9章中有关 $\mu\text{C}/\text{OS}-\text{II}$ 的部分,这样就可以用 GNU Tools 软件和 SkyEye 软件来编译、调试和运行第5章和第9章中介绍的 $\mu\text{C}/\text{OS}-\text{II}$ 。

如果你对嵌入式系统有一定的了解,可能就不需要读第1章。对于适合初学者阅读的其他部分,可根据具体情况选择阅读和实践。然后,读者可阅读第6章,掌握移植 Linux 的基本方法。接下来,阅读第4章可以了解 BootLoader 的基础和典型 BootLoader 的具体实现。阅读第7章可以学习到许多比较有深度的调试技术,对于操作系统级或系统软件级开发很有帮助。第8章对如何在具体开发板上移植操作系统和应用进行了详细讲解,然后读者可以根据具体的兴趣和要求,参看从第8章~第12章介绍的在各种具体开发板上移植操作系统和应用。注意,在阅读本书的同时,可通过 GNU Tools 软件和 SkyEye 软件进行实践。

作者与论坛

参与本书写作的有陈渝(SkyEye 项目发起人)、李明(清华 BBS 嵌入式系统版版主)、杨晔(Intel 中国上海研发中心工程师)、詹荣开(LinuxForum 论坛 Linux Driver 版版主)、尹首一(清华 BBS Xwindows 版版主)、刘宇洪、王利明(大连理工大学 BBS 技术主管)、谢军、焦振强、宋磊、苏杭、许拥军、高靖。冯刚、殷庆轩、尹文超、康烁、王华勇、布和等也参与了本书的完善工作。全书由陈渝统稿。

虽然本书作者认真地对本书进行了检查,但本书中难免还会有错误或笔误。我们欢迎讨论和指正,大家可以给 yuchen@tsinghua.edu.cn 发 email 告诉我们书中还存在错误。我们在 SkyEye 网站 <http://www.skyeye.org> 上设置了错误纠正网页,会即时公布错误纠正信息。同时我们在 SkyEye 网站上设立了一个论坛,欢迎大家就书中的问题或嵌入式开发方面的情况进行交流。

致 谢

下面是参与 SkyEye 开发的人员, SkyEye 的发展和壮大有他们的汗水和心血,本书的内容中也有他们的成果。目前参与开发 SkyEye 的人员有:陈渝、李明、杨晔、王利明、尹首一、刘宇洪、康烁、尹文超、曾益、谢军、焦振强、王宇轩、王曦、肖贺、Simone Zinanni、Trilok Soni 等。

下面是帮助过 SkyEye 开发的单位和个人:AKA embedded 兴趣小组提供了嵌入式学习的场所和工具;北京理工大学的马忠梅老师和她的学生叶楠、MiniGUI 项目的开发者魏永明对 SkyEye 的 LCD 模拟开发提供了大量的帮助;gro.clinux.org 网站为 SkyEye 提供了类 SourceForge 分布式协同开发环境;www.linuxfans.org 网站提供了 SkyEye 的论坛;中国科技大学的 qsluo 设计了 SkyEye 的网站;Intel 中国公司、ARM 中国公司、周立功单片机发展公司、思创嵌入式开发小组提供了多种嵌入式开发板给 SkyEye 开发小组。

感谢北京理工大学的马忠梅老师和清华大学的邵贝贝教授,正是他们的建议和帮助,促成了本书的面世。同时感谢北京航空航天大学出版社的编辑人员,由于他们高效的工作,才使得本书能够及时与读者见面。

陈 渝

2004 年 5 月

目 录

基础篇

第 1 章 嵌入式系统	3
1.1 嵌入式硬件系统介绍	3
1.2 嵌入式软件系统	6
1.3 嵌入式系统学习开发入门	11
第 2 章 嵌入式开发工具	13
2.1 GNU Tools 开发工具	13
2.2 binutils 开发工具	15
2.3 ld 软件	31
2.4 gcc 编译工具	39
2.5 gdb 调试工具	48
2.6 GNU make	51
2.7 其他嵌入式开发工具	68
2.8 小 结	71
第 3 章 SkyEye 硬件模拟平台	72
3.1 SkyEye 介绍	72
3.2 SkyEye 模拟硬件介绍	74
3.3 SkyEye 的设计实现	75
3.4 安装使用 SkyEye	128
3.5 改动和扩展 SkyEye	133
3.6 小 结	135
第 4 章 嵌入式系统的 Boot Loader 技术	136
4.1 基本的 Boot Loader 程序概念	136
4.2 典型的 Boot Loader 程序框架结构	138
4.3 ARMboot 的分析和使用	148
4.4 RedBoot 在嵌入式系统中的应用	192
4.5 其他的开源 Boot Loader 简介	200
4.6 小 结	201

第 5 章 $\mu\text{C}/\text{OS}-\text{II}$ 实时内核及其应用	202
5.1 $\mu\text{C}/\text{OS}-\text{II}$ 简介	202
5.2 $\mu\text{C}/\text{OS}-\text{II}$ 在 ARM 处理器上的移植分析	204
5.3 $\mu\text{C}/\text{OS}-\text{II}$ 上的 TCP/IP 协议栈——lwIP	212
5.4 $\mu\text{C}/\text{OS}-\text{II}$ 的简单应用——Genie-shell 程序	222
5.5 基于 SkyEye 的 $\mu\text{C}/\text{OS}-\text{II}$ 的实时性能测试分析	224
5.6 小 结	232
第 6 章 Linux 内核和文件系统配置	233
6.1 Linux 内核配置与编译	233
6.2 嵌入式 Linux 常用文件系统简介	252
6.3 根文件系统	255
6.4 小 结	268
第 7 章 嵌入式系统软硬件调试技术	269
7.1 硬件级调试工具简介	269
7.2 JTAG 标准	275
7.3 ARM 核的 JTAG 片上调试接口	277
7.4 简易的嵌入式系统调试器 JTAGER	283
7.5 GDB 远程调试概述	291
7.6 使用 gdbserver 进行远程调试	295
7.7 内核级软件调试技术	300
7.8 内核消息的获取与记录	301
7.9 远程内核调试工具 KGDB	302
7.10 Linux 内核调试器 KDB	310
7.11 小 结	317

实 践 篇

第 8 章 嵌入式系统软件开发基本分析	321
8.1 JFlash 的工作原理及程序分析	322
8.2 Boot Loader 功能介绍及程序分析	331
8.3 $\mu\text{C}/\text{OS}-\text{II}$ 实时内核移植分析	338
8.4 Linux 操作系统移植分析	342
8.5 RTLinux 的移植	357
8.6 嵌入式 Linux 的应用程序开发	359
8.7 小 结	361

第 9 章 Samsung 系列开发板	362
9.1 Evaluator-7T s3c4510 评估板简介	362
9.2 通过 BSL 方式使用 Ev-7T 评估版	363
9.3 使用 BSL 改变系统配置	365
9.4 使用 BSL 读/写 Flash	367
9.5 使用 RedBoot 下载和执行程序	370
9.6 相关程序代码分析	373
9.7 μ C/OS-II 在 Evaluator-7T 上的移植分析	383
9.8 μ Clinux 在 Evaluator-7T 上的裁减与移植	383
9.9 思创 S3C44B0 黄金开发板简介	388
9.10 44B0 开发板硬件配置	389
9.11 44B0 开发板软件配置	390
9.12 思创开发板电路原理详介	391
9.13 44B0 开发板使用工具介绍	395
9.14 44B0 开发板烧写指南	399
9.15 μ Clinux 在思创 S3C44B0 上的移植	403
9.16 小 结	406
第 10 章 AT91EV40 开发板	407
10.1 AT91EV40 开发板简介	407
10.2 μ Clinux 的配置、编译和运行	416
10.3 8019AS 网络控制芯片驱动分析	425
10.4 μ C/OS-II for SkyEye 的编译和运行	429
10.5 μ C/OS-II for SkyEye 分析	430
10.6 TCP/IP 协议栈—lwIP 移植分析	441
10.7 小 结	442
第 11 章 Intel PXA25xLubbock 开发板	443
11.1 基于 XScale CPU 的 Intel PXA250 Lubbock 开发板简介	443
11.2 Linux Host 主机环境配置	445
11.3 XScale 交叉编译环境的建立	452
11.4 Lubbock 开发板的 Linux 内核配置	458
11.5 在 SkyEye 上安装调试 Linux 内核	465
11.6 基于 Lubbock 开发板的 Linux 内核启动分析	467
11.7 小 结	468
第 12 章 EVB EP7312 开发板	469
12.1 EP7312CPU 简介	469

12.2	EVB7312 开发套件简介	470
12.3	在 EP7312 裸板上烧写 BootLoader 的说明	472
12.4	在 EVB7312 开发板上运行嵌入式 Linux	475
12.5	Linux 2.4.13 在 EVB7312 开发板上的 .config 文件	486
12.6	用 SkyEye 模拟 EP7312 开发系统	490
12.7	ARM Linux for EP7312 的启动分析	493
12.8	小 结	494
附录 A	ARM 指令集	495
附录 B	Linux 常见命令	498
附录 C	光盘内容	505
附录 D	嵌入式开发的资源	507
附录 E	GPL 协议	510
参考文献		515

基础篇

- ◆ 嵌入式系统
- ◆ 嵌入式开发工具
- ◆ SkyEye 硬件模拟平台
- ◆ 嵌入式系统的 Boot Loader 技术
- ◆ $\mu\text{C}/\text{OS} - \text{II}$ 实时内核及其应用
- ◆ Linux 内核和文件系统配置
- ◆ 嵌入式系统软硬件调试技术

第1章 嵌入式系统

在阅读这一章之前,请一定要先仔细阅读本书前言,以确定是否有必要阅读本章。本书的对象是嵌入式操作系统和系统软件开发的初学者、中级嵌入式开发人员、在校学习嵌入式操作系统和系统软件的大学生等。本书最大的特点是让读者不需要在真实硬件的环境下就可以学习和开发嵌入式系统软件。我们认为嵌入式系统开发强调实践,没有实践的磨炼,很难深入掌握嵌入式系统开发的精髓。嵌入式系统是一个软硬件结合的综合性系统,所以虽然本书侧重于嵌入式系统软件的学习和开发,但对相关的硬件和系统结构也有比较深入的分析,提供了大量的实践内容。

1.1 嵌入式硬件系统介绍

嵌入式系统的核心部件是各种类型的嵌入式处理器,按嵌入式处理器的寄存器位数来分,从8位、16位、32位到64位,各种类型层出不穷。据目前不完全统计,全世界嵌入式处理器的品种总量已经超过1000种,流行体系结构有三十几个系列。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为64KB~16MB,处理速度为0.1~2000MIPS。从功能上看,用于嵌入式系统的嵌入式处理器主要分为3类:嵌入式微处理器、嵌入式微控制器、嵌入式DSP。下面介绍用于嵌入式系统中的一些关键功能部件。

1.1.1 嵌入式微处理器

嵌入式微处理器(Embedded Microprocessor Unit,简称EMPU)在本质上与通用计算机中的CPU是一样的,只是在具体的实现细节和功能上有所不同。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统的体积和功耗。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,在功能上去掉一些在嵌入式领域不常用的功能单元。

和工业控制计算机相比,嵌入式微处理器具有体积小、质量轻、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,从而降低了系统的可靠性,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。嵌入式处理器目前主要有x86、Power PC、68K/Cold-fire、MIPS、ARM、SuperH等系列。

1.1.2 嵌入式微控制器

嵌入式微控制器(Microcontroller Unit,简称MCU)又称单片机,即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、Flash RAM、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基本都是一样的,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。微控制器的片上外设资源一般比较丰富,适合于比较简单的控制过程,因此称为微控制器。

嵌入式微控制器是目前嵌入式系统的主流之一,品种和数量相当多,较有代表性的通用系列包括 8051、P51XA、MCS - 251、MCS - 96/196/296、C166/167、MC68HC05/11/12/16、68300 等。另外还有许多半通用系列,如支持 USB 接口的 MCU 8XC930/931、C540、C541,支持 I²C、CAN - Bus、LCD 的 MCU 及众多专用 MCU 和兼容系列。

1.1.3 嵌入式 DSP 处理器

嵌入式 DSP 处理器(Embedded Digital Signal Processor,简称 EDSP)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 处理器的阶段。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 公司的 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 公司的 MCS - 296 和 Infineon(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、实时语音压解系统、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 Texas Instruments 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 DSP56300 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嵌入式 DSP 结构的 DSP 处理器,其特点是具备双 Harvard 结构和双乘/累加单元。

1.1.4 嵌入式片上系统

随着 EDI 技术和 VLSI 设计的迅速发展,在一个硅片上可以包括 CPU、各种外设控制器(如网络控制器、LCD 控制器等)等多个功能单元,构成一个完整的复杂硬件系统,这就是嵌入式片上系统(System On Chip,简称 SOC)。这样,各种嵌入式处理器内核将作为 SOC 的一种标准库,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。用户只需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。这样除了一些个别无法集成的器件外,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,这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 Infineon 公司的 TriCore、Motorola 公司的 M-Core、某些基于 ARM CPU 核的应用处理器、Echelon 公司和 Motorola 公司联合研制的 Neuron 芯片等。专用 SOC 一般专用于特定系统中,例如 Philips 公司的 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,形成了一个可加载 JAVA 或 C 语言的专用 SOC。

1.1.5 嵌入式外设

嵌入式系统中的外设千差万别,这里只简要介绍存储器和通信接口设备。

1. 存储器

在嵌入式系统中存储数据和程序的功能部件称为存储器。按存储器所处的位置,可把存储器分为内部存储器(位于嵌入式处理器内部)和外部存储器(位于嵌入式处理器外部)。内部存储器位于嵌入式处理器所在的同一个芯片中,这样处理器不需要多余的访问电路就可快速地访问内部存储器。外部存储器与嵌入式处理器分别处于不同的芯片中,可能需要在处理器和存储器之间增加附加电路,且访问速度一般比内部存储器要慢。目前常见的存储设备有 RAM、ROM、Flash、硬盘、软盘、CD-ROM 等,它们有各自的特征,我们应该根据具体的应用需求,选择最合适的存储器。

2. 通信接口设备

目前有多种通信接口设备,用于不同设备之间的数据交换。如常见的串行接口(UART)、USB 接口、以太网络接口、IEEE1394 接口、蓝牙接口、红外线接口(Infra Red Data Association,简称 IrDA)等。最常用的是串行接口,它以串行方式一位一位地发送和接收数据,UART 硬件接收到这些位之后,会把这些位转换成字节,并存储到缓冲区中,以便进一步处理。在进行嵌入式开发的初期,软件一般都通过 UART 来进行各种输入输出操作。以太网接口是最普遍的计算机网络接口,目前 100 Mbps 的以太网已经非常便宜,常用于嵌入式系统中的高速数据传输。USB 接口和 IEEE1394 接口常用于连接各种数字设备,如数字摄像机、数字照相机、移动 U 盘、移动硬盘等。在无线数据传输中,常见的有 IEEE802.11 系列无线网络传输接口、蓝牙接口以及红外线接口。

1.1.6 典型嵌入式处理器介绍

这里主要介绍一些目前典型的嵌入式处理器 ARM 和 MIPS 的基本情况。有关其他嵌入式处理器(如 PowerPC、68K/ColdFire 等)的信息请参考相应的资料。

1. ARM

ARM(Advanced RISC Machines)CPU 是 ARM 公司的高性能、廉价、低功耗的 RISC 处理器,适用于多种领域,例如嵌入控制、消费/教育类多媒体、DSP 和移动式应用等。ARM 公司将其技术授权给世界上许多著名的半导体、软件和 OEM 厂商,每个厂商得到 ARM CPU 相关技术及服务。目前,总共有 30 家半导体公司与 ARM 签订了硬件技术使用许可协议,其中包括 Intel、IBM、LG 半导体、NEC、SONY、Philips 这样的大公司。至于软件系统的合伙人,则

包括 Microsoft、SUN 和 MRI 等一系列知名公司。

ARM 公司提供一系列 CPU 内核、体系扩展、微处理器和系统芯片方案。由于所有产品均采用一个通用的基本体系架构,所以相同的软件理论上可在所有产品中运行。目前 ARM 公司常见的 CPU 内核有:

ARM7: 小型、快速、低功耗、集成式 RISC 内核,用于移动通信。

ARM7TDMI: 这是 ARM 公司目前授权用户最多的一项产品,其特点是将 ARM7 指令集同 Thumb 扩展组合在一起,以减少内存容量和系统成本。同时它还利用嵌入式 ICE 调试技术来简化系统设计,并用一个 DSP 增强扩展来改进性能。该产品的典型用途是数字蜂窝电话和硬盘驱动器。

ARM9TDMI: 基于 5 阶段流水线的 ARM9 内核,同时配备 Thumb 扩展、ICE 调试和 Harvard 结构。在生产工艺相同的情况下,性能可达 ARM7TDMI 的两倍之多。常用于联网和机顶盒。

目前高端的 ARM CPU 核有 ARM10 和 ARM11 系列,主要用于高端的个人手持设备和通信设备中。

2. MIPS

MIPS 是世界上很流行的一种 RISC 处理器。MIPS 的意思是“无内部互锁流水级的微处理器”(Microprocessor without Interlocked Piped Stages),其机制是尽量利用软件办法避免流水线中的数据相关问题。它最早是在 20 世纪 80 年代初期由美国斯坦福大学 Hennessy 教授领导的研究小组研制出来的。MIPS 公司的 R 系列就是在此基础上开发的 RISC 工业产品的微处理器。在嵌入式方面,MIPS 系列微处理器是目前仅次于 ARM 的用得最多的处理器之一(1999 年以前 MIPS 是世界上用得最多的处理器),其应用领域覆盖游戏机、路由器、激光打印机、掌上电脑等各个方面。MIPS 的系统结构及设计理念比较先进,其指令系统经过通用处理器指令体系 MIPS I、MIPS II、MIPS III、MIPS IV 到 MIPS V,嵌入式指令体系 MIPS16、MIPS32 到 MIPS64 的发展已经十分成熟。在设计理念上 MIPS 强调软硬件协同提高性能,同时简化硬件设计。

1.2 嵌入式软件系统

如果嵌入式系统的功能(如电动玩具、空调、冰箱等的简单功能需求)比较简单,则嵌入式软件只需要完成比较简单的功能,就不需要嵌入式操作系统的支持。如果嵌入式系统的功能比较复杂,需要网络功能、存储器管理、进程/线程管理等,则通过嵌入式操作系统的帮助,可加快嵌入式系统软件的开发进度和可靠性。

1.2.1 嵌入式操作系统概述

简单地讲嵌入式操作系统就是支持嵌入式系统工作的操作系统。它在本质上与通用操作系统没有太大的区别,它一般用于比较复杂的嵌入式系统中。嵌入式系统一般具有实时特点。所谓实时系统,是指在这种系统中,一个优先等级高的任务能够获得立即的、没有延迟的服务,不需要等候任何其他优先级低的任务,而且在它得到 CPU 的使用权后,它可以一直执行到工作结束或是有更高等级的进程出现为止。在 Borko Furht、David Gluch 等著的《Real Time