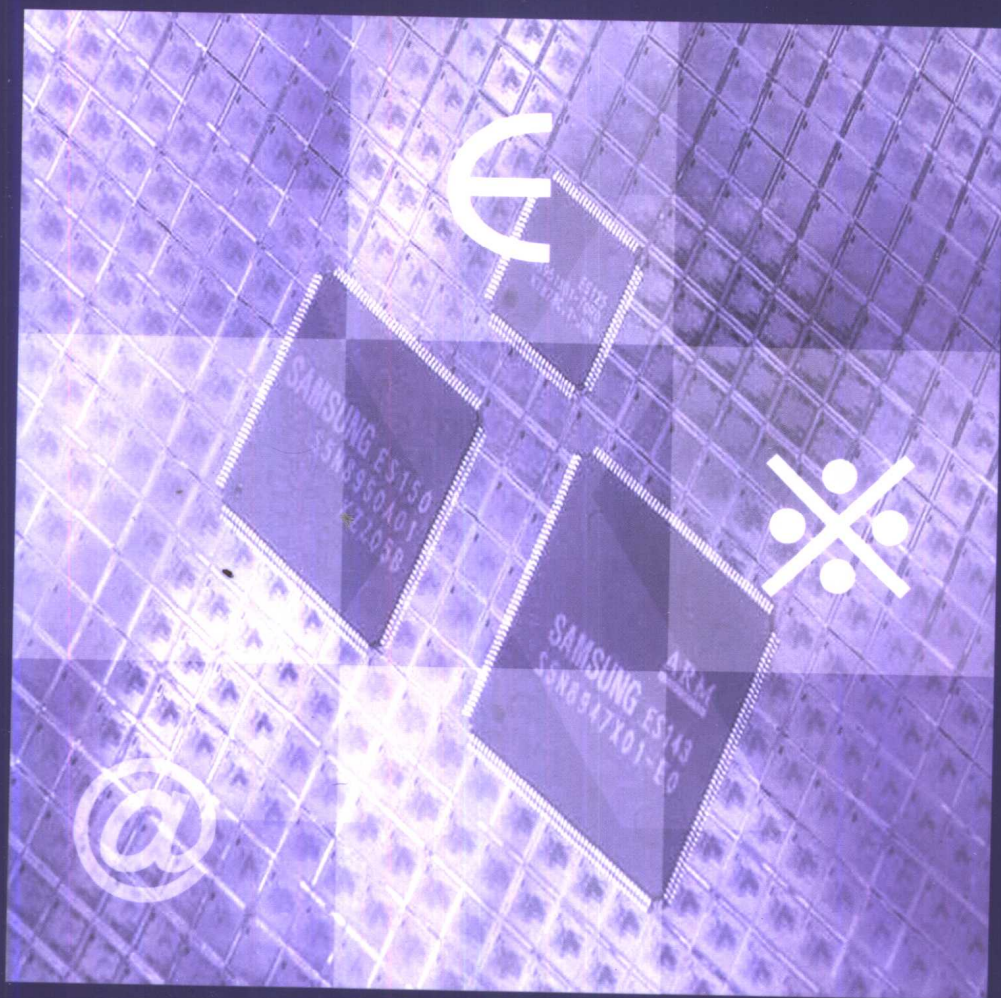


嵌入式系统设计 与实例开发

——基于 ARM 微处理器与 $\mu\text{C}/\text{OS}-\text{II}$ 实时操作系统



王田苗 主编



清华大学出版社

嵌入式系统设计与实例开发

——基于 ARM 微处理器与 μ C/OS-II 实时操作系统

王田苗 主编

清华大学出版社

(京) 新登字 158 号

内 容 简 介

本书以目前流行的基于 ARM 架构的嵌入式微处理器及易于学习开发的 $\mu\text{C}/\text{OS}$ 嵌入式操作系统为核心,结合作者的教学与科研工作,介绍了嵌入式系统的原理、设计方法及实例编程开发。

本书共分 11 章,前 3 章介绍了嵌入式系统的基本概念及嵌入式系统设计的一般方法。第 4 章介绍了嵌入式系统的开发工具——ARM 公司的仿真开发环境 SDT 的使用方法。第 5~9 章是本书的核心内容,介绍了如何在 $\mu\text{C}/\text{OS}$ -II 实时内核基础上扩展一个实用嵌入式操作系统的设计过程,包括 $\mu\text{C}/\text{OS}$ -II 在 ARM7 微处理器上移植、文件系统、外设驱动、Unicode 汉字库、图形用户接口(GUI)和其他应用程序接口(API)的设计,并介绍了在扩展的嵌入式系统开发平台上进行编程的方法。最后 2 章介绍了二次开发的实例——嵌入式机器人控制器和嵌入式工程机械智能监控器的设计。

本书可以作为机器人技术、机电控制系统、信息家电、工业控制、手持设备、智能玩具、医疗仪器等方面嵌入式系统开发与应用的参考书,也可作为高等学校有关嵌入式系统教学的本科生或研究生的教材。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名:嵌入式系统设计与实例开发

——基于 ARM 微处理器与 $\mu\text{C}/\text{OS}$ -II 实时操作系统

作 者:王田苗 主编

责任编辑:钟志芳

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印 刷 者:清华大学印刷厂

发 行 者:新华书店总店北京发行所

开 本:787 $\times$ 1092 1/16 印张:17.25 字数:391 千字

版 次:2002 年 9 月第 1 版 2002 年 9 月第 1 次印刷

书 号:ISBN 7-302-05859-8/TP $\cdot$ 3467

印 数:0001~5000

定 价:28.00 元

前 言

嵌入式系统是继 IT 网络技术之后, 又一个新的技术发展方向。由于嵌入式系统具有体积小、性能强、功耗低、可靠性高以及面向行业应用的突出特征, 目前已经广泛地应用于军事国防、消费电子、网络通信、工业控制等各个领域。

在以往先进机器人研究与实际机电控制系统应用过程中, 常规方式是采用上位机/下位机的控制方式, 上位机一般为工控机或 PC 机, 下位机一般为单片机或 DSP 等微控制器。然而, 近年来随着微电子与嵌入式软件技术的发展, 32/64 位微处理器的性能得到了很大的提升, 我们发现这种常规的上位机/下位机的控制方式已失去了原有的优势, 特别是针对高性能、低功耗的实际现场应用, 只需要采用基于微处理器的嵌入式系统就可以实现所需要的功能, 而且性能价格比高。因此, 我们认为嵌入式技术是未来一个重要的技术发展方向。

中国是世界未来制造业及加工业的重要基地之一, 也可以说是机械与电子产品制造的重要基地之一。面对加入 WTO 之后的机遇与挑战, 我们隐隐约约感觉到这种巨大的变革, 将引起对嵌入式技术人才的大量迫切需求。为此, 作者及其课题组成员很想通过承担与参与国家 863 项目“机群智能化工程机械”嵌入式智能监控器的研制、“机器人辅助微创外科手术”控制系统与遥操作的研究、“智能仿生机器鱼”推进控制技术等课题的研究, 结合培养博士后、博士生和硕士生所取得的有关科研成果, 在分析研究美国科罗拉多州立大学和华盛顿大学有关嵌入式系统教学课程体系结构的基础上, 试图从嵌入式系统设计、嵌入式系统编程及工程实践训练三个方面, 编写一本嵌入式系统设计与实例开发的教材, 进行一种大胆有益的尝试。

本书特色是以目前流行的基于 ARM 架构的嵌入式微处理器及易于学习开发的 $\mu\text{C}/\text{OS}$ 嵌入式操作系统为核心, 结合作者的教学与科研工作, 介绍了嵌入式系统的原理、设计方法及实例编程开发。对本书研究开发的基于 ARM 的嵌入式硬件平台与实时操作系统的扩展程序(包括文件系统、存储器系统、各种接口程序、GUI、API、Unicode 汉字库、LCD 和键盘等驱动程序等)有兴趣的读者, 可以与北京航空航天大学逸夫科学馆北航机器人研究所 110 室联系。电话: (010) 82318003/82313564, 传真: (010) 82313564。

全书共分 11 章, 参与本书编写的有刘淼、陶伟、王晓君、魏洪兴、胡耀光、王田苗。全书由王田苗统稿。

感谢北京航空航天大学机器人研究所智能机电技术与系统研究组全体教师及研究生的大力帮助与支持, 感谢清华大学出版社编辑人员, 正是由于他们高效辛勤的工作, 才使本

书能够在这么短的时间内与读者见面。

由于作者知识所限，书中不足之处在所难免，恳请各位专家和读者赐正。

作 者

于北京航空航天大学逸夫科学馆

2002.7.31

目 录

第 1 章 嵌入式系统概述	1
1.1 嵌入式系统简介	1
1.1.1 什么是嵌入式系统	1
1.1.2 嵌入式系统的特点	5
1.1.3 嵌入式系统的分类	5
1.2 嵌入式系统的应用领域	8
1.3 嵌入式系统在机电产品方面的应用	10
1.4 嵌入式系统的现状和发展趋势	13
1.4.1 嵌入式系统的现状	13
1.4.2 未来嵌入式系统的发展趋势	14
第 2 章 嵌入式系统的基本知识	16
2.1 嵌入式微处理器和嵌入式操作系统	16
2.1.1 嵌入式微处理器	16
2.1.2 嵌入式操作系统	21
2.2 嵌入式系统的选型原则	26
2.2.1 硬件平台的选择	26
2.2.2 嵌入式操作系统的选型原则	27
2.3 嵌入式系统中的一些重要概念	30
2.3.1 基本概念	30
2.3.2 关于实时系统的概念	31
2.4 基于 ARM 和 μ C/OS-II 的嵌入式开发平台	32
2.4.1 为什么选择 ARM 和 μ C/OS-II	33
2.4.2 UP-NetARM200 嵌入式开发平台简介	33
2.4.3 UP-NetARM200 嵌入式开发平台结构	34
第 3 章 嵌入式系统的设计准则及开发方法	37
3.1 嵌入式系统设计的一般方法	37
3.2 嵌入式系统的硬件/软件协同设计技术	39
3.3 嵌入式系统的测试技术	41
3.3.1 软件分析和仿真技术	41

3.3.2	软件测试技术	41
3.3.3	嵌入式软件的测试工具	43
3.4	嵌入式系统的可靠性 ^[6]	44
3.4.1	嵌入式系统失效特征	44
3.4.2	常用的可靠性参数	45
第 4 章	ARM SDT 仿真环境	48
4.1	SDT 仿真环境概述	48
4.1.1	ARM 仿真器的流程简介	48
4.1.2	重要概念	49
4.2	APM 开发工具	50
4.2.1	建立项目文件	50
4.2.2	配置项目工具	52
4.3	ADW 开发工具	55
4.3.1	环境简介	55
4.3.2	工具配置和调试	57
第 5 章	μ C/OS-II 在 ARM 微处理器上的移植	60
5.1	移植——将 μ C/OS-II 移植到自己的硬件平台	60
5.2	μ C/OS-II 在 S3C44B0X 上的移植	61
第 6 章	基于 μ C/OS-II 建立自己的 RTOS	66
6.1	基于 μ C/OS-II 扩展 RTOS 的体系结构	66
6.2	建立文件系统	68
6.2.1	文件系统简介	68
6.2.2	文件系统的实现过程	69
6.2.3	文件系统相关的 API 函数功能详解	72
6.3	外设及驱动程序	73
6.3.1	串行口	73
6.3.2	液晶显示驱动程序	74
6.3.3	键盘驱动程序	75
6.3.4	USB 接口	76
6.3.5	网络相关组件	77
6.4	图形用户接口 (GUI) 函数	77
6.4.1	基于 Unicode 的汉字字库	77
6.4.2	Unicode 字库的显示及相关函数	83
6.5	基本绘图函数	83
6.5.1	相关绘图函数	84

6.5.2 典型的控件	91
6.6 系统的消息队列	95
6.6.1 系统消息	95
6.6.2 消息相关函数	95
6.7 其他实用的应用程序接口 (API) 函数	96
第 7 章 ARM7 的硬件开发平台介绍	100
7.1 ARM7 微处理器简介	100
7.1.1 ARM 系列芯片简介	100
7.1.2 ARM7TDMI 简介	100
7.1.3 Samsung S3C44B0X 介绍	101
7.2 基于 S3C44B0X 微处理器的嵌入式系统	101
7.2.1 Samsung S3C44B0X 简介	101
7.2.2 基于 S3C44B0X 微处理器的嵌入式系统体系结构	105
7.3 存储器系统介绍	106
7.4 接口介绍	109
7.4.1 通用异步收发器 (UART)	109
7.4.2 USB 接口	116
7.4.3 JTAG 调试接口	123
7.5 人机交互接口	123
7.5.1 LCD 显示模块	124
7.5.2 键盘模块	126
第 8 章 用户应用程序的建立与调试	131
8.1 在嵌入式系统上运行的应用程序的结构	131
8.2 建立应用程序	136
8.2.1 系统的消息循环	136
8.2.2 创建新任务以及任务之间的同步	137
8.2.3 使用绘图函数	139
8.2.4 系统的控件	141
8.3 用户应用程序综合举例	142
8.3.1 配置 ARM SDT 2.5 开发环境	142
8.3.2 开发基本的嵌入式应用程序	148
第 9 章 嵌入式应用程序举例	154
9.1 绘图 API 函数	154
9.1.1 绘图的 API 函数应用举例	154
9.1.2 绘图的 API 函数应用举例的源代码	156

9.2 系统的消息循环	161
9.2.1 使用系统的消息循环	161
9.2.2 系统的消息循环使用的源代码	163
9.3 USB 联机通信	168
9.3.1 USB 联机通信程序的实现过程	168
9.3.2 USB 联机通信程序实现的源代码	169
9.4 文件的使用	170
9.4.1 文件的读取应用举例	170
9.4.1 文件的读取实现的源代码	172
9.5 列表框控件的使用	173
9.5.1 列表框控件的使用举例	173
9.5.2 列表框控件举例的源代码	174
9.6 文本框控件的使用	176
9.6.1 文本框控件的使用举例	176
9.6.2 文本框控件的举例源代码	178
9.7 系统的多任务和系统时钟	181
9.7.1 系统的多任务和系统时钟应用举例	181
9.7.2 系统的多任务和系统时钟举例源代码	183
9.8 综合举例	187
9.8.1 综合举例的设计思路	187
9.8.2 综合举例的源代码	190
第 10 章 嵌入式机器人控制器的设计	196
10.1 基于 PC 的机器人控制系统	197
10.1.1 基于 DSP 的嵌入式网络直流伺服驱动器	197
10.1.2 PC 机与网络直流伺服驱动器的接口关系	198
10.2 两自由度机器人控制软件结构设计	199
10.2.1 G 代码编辑环境	200
10.2.2 G 代码解析	202
10.2.3 轨迹插补与运动学逆解	203
10.3 机器人控制	203
10.3.1 两自由度机器人控制软件的实现	204
10.3.2 运动轨迹仿真及机器人运动学逆解的实现	205
10.3.3 机器人控制的实现	206
10.4 嵌入式机器人控制器设计	207
10.4.1 嵌入式控制器可以取代 PC 机吗	207
10.4.2 嵌入式机器人控制器的结构	208
10.4.3 嵌入式机器人控制器应用程序设计	208

10.4.4	菜单式应用程序的结构分析	209
10.4.4	用嵌入式控制器实现机器人控制	211
10.4.5	嵌入式控制与网络直流伺服驱动器的通信接口	214
第 11 章	嵌入式工程机械智能监控器	217
11.1	工程机械简介	217
11.1.1	中国的工程机械	217
11.1.2	工程机械的监控器	218
11.2	基于 ARM 的工程机械监控器的设计	220
11.2.1	监控器的硬件结构	220
11.2.2	监控器的功能及相关函数	222
附录 1	基于 μC/OS-II 和 μC/OS 的一些产品	244
附录 2	国外高校嵌入式系统课程的体系结构	256
参考文献		261

第 1 章 嵌入式系统概述

随着社会的日益信息化，计算机和网络已经全面渗透到日常生活的每一个角落。对于我们每个人来说，需要的已经不再仅仅是那种放在桌上处理文档、进行工作管理和生产控制的计算机“机器”。任何一个普通人都可能拥有从小到大的各种使用嵌入式技术的电子产品，小到 MP3、PDA 等微型数字化产品，大到网络家电、智能家电、车载电子设备等。目前，各种各样的新型嵌入式系统设备在应用数量上已经远远超过了通用计算机。在工业和服务领域中，使用嵌入式技术的数字机床、智能工具、工业机器人、服务机器人正在逐渐改变着传统的工业生产和服务方式。

嵌入式系统（Embedded System）是当今最热门的概念之一，然而到底什么是嵌入式系统呢？什么样的技术可以称之为嵌入式系统技术呢？通过本章的学习，不仅可以回答以上问题，同时还能够对嵌入式系统及其技术和应用有一个全面的了解。

通过本章的学习，读者将了解以下内容：

- ✧ 什么是嵌入式系统
- ✧ 嵌入式系统的特点及分类
- ✧ 嵌入式系统的主要应用领域
- ✧ 嵌入式系统的发展趋势

1.1 嵌入式系统简介

1.1.1 什么是嵌入式系统

在讨论嵌入式系统定义之前，先来看一看图 1-1 所示的几个嵌入式系统的典型应用。

嵌入式系统本身是一个相对模糊的定义。由于目前嵌入式系统已经渗透到日常生活中的各个方面，在工业、服务业、消费电子等领域的应用范围不断扩大，使得难以给出“嵌入式系统”一个明确的定义。

举个简单例子：一个手持的 MP3 是否可以称为嵌入式系统呢？答案是肯定的。那么一个 PC104 的微型工业控制计算机是嵌入式系统吗？当然也是，工业控制是嵌入式系统技术的一个典型应用领域。然而比较两者，会发现除了其中都嵌入有微处理器，二者几乎完全不同。那是否可以说嵌入有微处理器的设备就是嵌入式系统呢？



图 1-1 使用嵌入式技术的各种设备

那么到底什么是嵌入式系统呢？

1. 嵌入式系统的历史

虽然嵌入式系统是近几年才风靡起来的，但是这个概念并非新近才出现。从 20 世纪 70 年代单片机的出现到今天各式各样的嵌入式微处理器、微控制器的大规模应用，嵌入式系统已经有了近 30 年的发展历史。

作为一个系统，往往是在硬件和软件双螺旋式交替发展的支撑下逐渐趋于稳定和成熟，嵌入式系统也不例外。

嵌入式系统最初的应用是基于单片机的。20 世纪 70 年代单片机的出现，使得汽车、家电、工业机器、通信装置以及成千上万种产品可以通过内嵌电子装置来获得更佳的使用性能，更容易使用，更快、更便宜。这些装置已经初步具备了嵌入式的应用特点，但是这时的应用只是使用 8 位的芯片，执行一些单线程的程序，还谈不上“系统”的概念。

提示：最早的 8 位单片机是 Intel 公司的 8048，它出现在 1976 年。Motorola 同时推出了 68HC05，Zilog 公司推出了 Z80 系列，这些早期的单片机均含有 256 字节的 RAM、4KB 的 ROM、4 个 8 位并口、1 个全双工串行口、两个 16 位定时器。之后在 80 年代初，Intel 又进一步完善了 8048，在它的基础上研制成功了 8051，这在单片机的历史上是值得纪念的一页，迄今为止，51 系列的单片机仍然是最为成功的单片机芯片，在各种产品中有着非常广泛的应用。

从 80 年代早期开始，嵌入式系统的程序员开始用商业级的“操作系统”编写嵌入式应用软件，这使得开发人员可以进一步缩短开发周期，降低开发成本并提高开发效率。1981 年，Ready System 开发出世界上第 1 个商业嵌入式实时内核（VTRX32）。这个实时内核包含了许多传统操作系统的特征，包括任务管理、任务间通信、同步与相互排斥、中断支持、内存管理等功能。此后一些公司也纷纷推出了自己的嵌入式操作系统，如 Integrated

System Incorporation (ISI) 的 pSOS 和 WindRiver 的 VxWorks、QNX 公司的 QNX 等。这些嵌入式操作系统都具有嵌入式的典型特点：它们均采用占先式的调度，响应的时间很短，任务执行的时间可以确定；系统内核很小，具有可裁剪性、可扩充性和可移植性，可以移植到各种处理器上；较强的实时性和可靠性，适合嵌入式应用。这些嵌入式实时多任务操作系统的出现，使得应用开发人员从小范围的开发中解放出来，同时也促使嵌入式有了更为广阔的应用空间。

90 年代以后，随着对实时性要求的提高，软件规模不断上升，实时内核逐渐发展为实时多任务操作系统 (RTOS)，并作为一种软件平台逐步成为目前国际嵌入式系统的主流。这时候更多的公司看到了嵌入式系统的广阔发展前景，开始大力发展自己的嵌入式操作系统。除了上面的几家老牌公司以外，还出现了 Palm OS、Windows CE、嵌入式 Linux、Lynx、Nucleus 以及国内的 Hopen、Delta OS 等嵌入式操作系统。随着嵌入式技术的发展前景日益广阔，相信会有更多的嵌入式操作系统软件出现。图 1-2 给出了比较有代表性的嵌入式操作系统。



风河的 Tornado/VxWorks



Palm 公司的 Palm OS

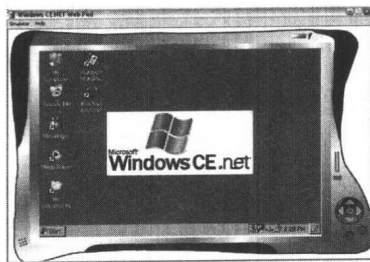
微软的 Windows CE (引自 www.pocketpcpower.net)

图 1-2 各种嵌入式操作系统

今天 RTOS 已经在全球形成了 1 个产业，根据美国 EMF (电子市场分析) 报告，1999 年全球 RTOS 市场产值达 3.6 亿美元，而相关的整个嵌入式开发工具 (包括仿真器、逻辑分析仪、软件编译器和调试器) 则高达 9 亿美元。

2. 嵌入式系统的定义

根据 IEEE (国际电气和电子工程师协会) 的定义，嵌入式系统是“控制、监视或者辅助设备、机器和车间运行的装置” (原文为 *devices used to control, monitor, or assist the operation of equipment, machinery or plants*)。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，还可以涵盖机械等附属装置。

不过,上述定义并不能充分体现出嵌入式系统的精髓。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。

可以从以下几个方面来理解国内对嵌入式系统的定义:

- ✦ 嵌入式系统是面向用户、面向产品、面向应用的,它必须与具体应用相结合才会具有生命力、才更具有优势。可以这样理解上述三个方面的含义,即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减利用。
- ✦ 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以及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。所以,介入嵌入式系统行业,必须有一个正确的定位。例如 Palm OS 之所以在 PDA 领域占有 70% 以上的市场,就是因为其立足于个人电子消费品,着重发展图形界面和多任务管理;而风河的 VxWorks 之所以在火星车上得以应用,则是因为其高实时性和高可靠性。
- ✦ 嵌入式系统必须根据应用需求可对软硬件进行裁剪,满足应用系统的功能、可靠性、成本、体积等要求。所以,如果能建立相对通用的软硬件基础,然后在其上开发出适应各种需要的系统,是一个比较好的发展模式。目前的嵌入式系统的核心往往是一个只有几 KB 到几十 KB 的微内核,需要根据实际的使用进行功能扩展或者裁减,但是由于微内核的存在,使得这种扩展能够非常顺利地进行。

同时还应该看到,嵌入式系统本身还是一个外延极广的名词。凡是与产品结合在一起的具有嵌入式特点的控制系统都可以叫嵌入式系统,而且有时很难给它下一个准确的定义。现在人们讲嵌入式系统时,某种程度上是指近些年来比较热的具有操作系统的嵌入式系统,本文在进行分析和展望时,也沿用这一观点。

一般而言,嵌入式系统的构架可以分成四个部分(如图 1-3 所示):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,在这里对其不加区分,这也是嵌入式系统和 Windows 系统的最大区别)。

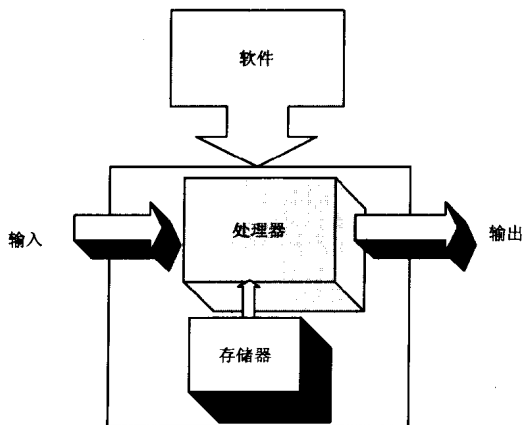


图 1-3 嵌入式系统的组成

1.1.2 嵌入式系统的特点

从前面对嵌入式系统所作的定义可以看出嵌入式系统的几个重要特征:

1. 系统内核小。由于嵌入式系统一般是应用于小型电子装置,系统资源相对有限,所以内核较之传统的操作系统要小得多。比如 ENEA 公司的 OSE 分布式系统,内核只有 5KB,而 Windows 的内核则要大得多。

2. 专用性强。嵌入式系统的个性化很强,其中的软件系统和硬件的结合非常紧密,一般要针对硬件进行系统的移植,即使在同一品牌、同一系列的产品中也需要根据系统硬件的变化和增减不断进行修改。同时针对不同的任务,往往需要对系统进行较大更改,程序的编译下载要和系统相结合,这种修改和通用软件的“升级”是完全不同的概念。

3. 系统精简。嵌入式系统一般没有系统软件和应用软件的明显区分,不要求其功能设计及实现上过于复杂,这样一方面利于控制系统成本,同时也利于实现系统安全。

4. 高实时性的操作系统软件是嵌入式软件的基本要求。而且软件要求固化存储,以提高速度。软件代码要求高质量和高可靠性。

5. 嵌入式软件开发要想走向标准化,就必须使用多任务的操作系统。嵌入式系统的应用程序可以没有操作系统而直接在芯片上运行;但是为了合理地调度多任务,利用系统资源、系统函数以及专家库函数接口,用户必须自行选配 RTOS (Real-Time Operating System) 开发平台,这样才能保证程序执行的实时性、可靠性,并减少开发时间,保障软件质量。

6. 嵌入式系统开发需要专门的开发工具和环境。由于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不能对其中的程序功能进行修改,必须有一套开发工具和环境才能进行开发,这些工具和环境一般是基于通用计算机上的软硬件设备以及各种逻辑分析仪、混合信号示波器等。开发时往往有主机和目标机的概念,主机用于程序的开发,目标机作为最后的执行机,开发时需要交替结合进行。

1.1.3 嵌入式系统的分类

由于嵌入式系统由硬件和软件两大部分组成,所以其分类也可以从硬件和软件进行划分。

1. 嵌入式系统的硬件

从硬件方面来讲,各式各样的嵌入式处理器是嵌入式系统硬件中的最核心的部分。目前,世界上具有嵌入式功能特点的处理器已经超过 1000 种,流行体系结构包括 MCU、MPU 等 30 多个系列。鉴于嵌入式系统广阔的发展前景,很多半导体制造商都开始大规模生产嵌入式处理器,并且公司自主设计处理器也已经成了未来嵌入式领域的一大趋势,其中从单片机、DSP 到 FPGA,品种越来越多,速度越来越快,性能越来越强,价格也越来越低。目前嵌入式处理器的寻址空间可以从 64KB 到 16MB,处理速度最快可以达到 2000 MIPS,封装从几个引脚到几百个引脚不等。

根据其现状, 嵌入式处理器可以分成下面几类^[1] (如图 1-4 所示):

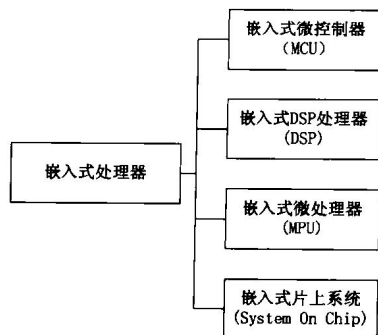


图 1-4 嵌入式系统分类

◆ 嵌入式微控制器 (MicroController Unit, MCU)

嵌入式微控制器 (如图 1-5 所示) 的典型代表是单片机。从 20 世纪 70 年代末单片机出现到今天, 虽然已经经过了 20 多年的历史, 但这种 8 位的电子器件目前在嵌入式设备中仍然有着极其广泛的应用。单片机芯片内部集成 ROM/EPROM、RAM、总线、总线逻辑、定时/计数器、看门狗、I/O、串行口、脉宽调制输出、A/D、D/A、Flash、EEPROM 等各种必要功能和外设。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, 因此称为微控制器。

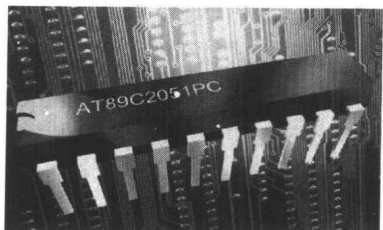


图 1-5 MCU

由于 MCU 低廉的价格、优良的功能, 所以拥有的品种和数量最多, 比较有代表性的包括 8051、MCS-251、MCS-96/196/296、P51XA、C166/167、68K 系列以及 MCU 8XC930/931、C540、C541, 并且有支持 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。近来 Atmel 推出的 AVR 单片机由于集成了 FPGA 等器件, 所以具有很高的性价比, 势必将推动单片机获得更高的发展。

◆ 嵌入式 DSP 处理器 (Digital Signal Processor, DSP)

DSP 处理器 (如图 1-6 所示) 是专门用于信号处理方面的处理器, 其在系统结构和指令算法方面进行了特殊设计, 具有很高的编译效率和指令执行速度。在数字滤波、FFT、频谱分析等各种仪器上 DSP 获得了大规模的应用。

DSP 的理论算法在 20 世纪 70 年代就已经出现, 但是由于专门的 DSP 处理器还未出现, 所以这种理论算法只能通过 MPU 等由分立元件实现。MPU 较低的处理速度无法满足 DSP 的算法要求, 其应用领域仅仅局限于一些尖端的高科技领域。随着大规模集成电路技术的发展, 1982 年世界上诞生了首枚 DSP 芯片。其运算速度比 MPU 快了几十倍, 在语音合成和编码解码器中得到了广泛应用。至 80 年代中期, 随着



图 1-6 DSP

CMOS 技术的进步与发展,第二代基于 CMOS 工艺的 DSP 芯片应运而生,其存储容量和运算速度都得到了成倍提高,成为语音处理、图像硬件处理技术的基础。到 80 年代后期,DSP 的运算速度进一步提高,应用领域也从上述范围扩大到了通信和计算机方面。90 年代后,DSP 发展到了第五代产品,集成度更高,使用范围也更加广阔。

目前最为广泛应用的是 TI 的 TMS320C2000/C5000 系列,另外如 Intel 的 MCS-296 和 Siemens 的 TriCore 也有各自的应用范围。

◆ 嵌入式微处理器 (MicroProcessor Unit, MPU)

嵌入式微处理器(如图 1-7 所示)是由通用计算机中的 CPU 演变而来的。它的特征是具有 32 位以上的处理器,具有较高的性能,当然其价格也相应较高。但与计算机处理器不同的是,在实际嵌入式应用中,只保留和嵌入式应用紧密相关的功能硬件,去除其他的冗余功能部分,这样就以最低的功耗和资源实现嵌入式应用的特殊要求。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。目前主要的嵌入式处理器类型有 Am186/88、386EX、SC-400、Power PC、68000、MIPS、ARM/ StrongARM 系列等。

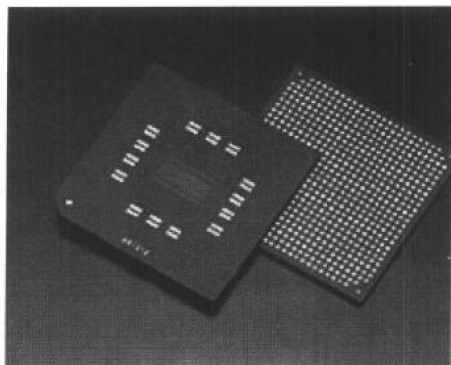


图 1-7 MPU

其中 ARM / StrongARM 是专为手持设备开发的嵌入式微处理器,属于中档的价位。

◆ 嵌入式片上系统 (System On Chip, SOC)

片上系统 SOC (如图 1-8 所示)是追求产品系统最大包容的集成器件,是目前嵌入式应用领域的热门话题之一。SOC 最大的特点是成功实现了软硬件无缝结合,直接在处理器片内嵌入操作系统的代码模块。而且 SOC 具有极高的综合性,在一个硅片内部运用 VHDL 等硬件描述语言,实现一个复杂的系统。用户不需要再像传统的系统设计一样,绘制庞大复杂的电路板,一点点地连接焊制,只需要使用精确的语言,综合时序设计直接在器件库中调用各种通用处理器的标准,然后通过仿真之后就可以直接交付芯片厂商进行生产。由于绝大部分系统构件都是在系统内部,整个系统就特别简洁,不仅减小了系统的体积和功耗,而且提高了系统的可靠性,提高了设计生产效率。

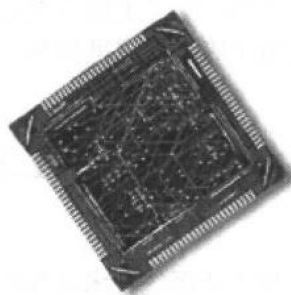


图 1-8 SOC

由于 SOC 往往是专用的,所以大部分都不为用户所知,比较典型的 SOC 产品是 Philips 的 Smart XA。少数通用系列如 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。

预计不久的将来,一些大的芯片公司将通过推出成熟的、能占领多数市场的 SOC 芯片,一举击退竞争者。SOC 芯片也将在声音、图像、影视、网络及系统逻辑等应用领域中发挥重要作用。