

ARM嵌入式系统设计与实践

于红旗 徐欣 刁节涛 孙兆林 ● 编著



清华大学出版社

ARM嵌入式系统设计与实践

于红旗 徐欣 刁节涛 孙兆林 编著

清华大学出版社
北京

内 容 简 介

NXF 公司推出的 LPC 系列 ARM 内核微处理器在工业控制等领域得到了广泛的应用, LPC ARM 是嵌入式系统中常用的较容易入门的 ARM 嵌入式处理器, 掌握 LPC ARM 微处理器的软硬件开发技术是嵌入式软硬件开发人员应掌握的一种基本技能。

本书结合 Proteus 仿真工具、Keil MDK 集成开发环境, 以 Keil C 为开发语言, 介绍了嵌入式系统的基本概念、ARM 内核的分类、ARM 内核开发流程、定时器、中断、GPIO、常见外设、操作系统、RLARM 嵌入式实时操作系统等基本概念、基本应用, 另外还描述了如何将 $\mu\text{C}/\text{OS-II}$ 移植到 LPC ARM 上, 用 Keil MDK 进行编译, 通过 Proteus 进行仿真, 讲述了 $\mu\text{C}/\text{GUI}$ 的基础知识, 以及如何将 $\mu\text{C}/\text{GUI}$ 移植到 Keil MDK 平台, 使用 LPC ARM 控制 12864 液晶显示的内容。

本书结构合理、文字流畅、通俗易懂, 通过仿真软件, 不需要任何硬件即可进行 ARM 嵌入式系统设计实验, 是学习 ARM 的理想入门教材, 具有计算机初步知识的读者都可快速入门。

本书可作为大学研究生、本科生、专科生 ARM 课程教材或自学材料, 也可供嵌入式系统相关技术人员参考。

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目 (CIP) 数据

ARM 嵌入式系统设计与实践/于红旗等编著. —北京: 清华大学出版社, 2015
ISBN 978-7-302-38587-5

I. ①A… II. ①于… III. ①微处理器-系统设计 IV. ①TP332

中国版本图书馆 CIP 数据核字 (2014) 第 273639 号

责任编辑: 贾小红 朱英彪

封面设计: 姜 姗

版式设计: 魏 远

责任校对: 王 云

责任印制: 宋 林

出版发行: 清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址: 北京清华大学学研大厦 A 座 邮 编: 100084

社 总 机: 010-62770175 邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 北京市人民文学印刷厂

装 订 者: 三河市漂源装订厂

经 销: 全国新华书店

开 本: 185mm×260mm 印 张: 17.75 字 数: 421 千字

版 次: 2015 年 6 月第 1 版 印 次: 2015 年 6 月第 1 次印刷

印 数: 1~3000

定 价: 38.00 元

前 言

NXP 公司推出的 LPC 系列 ARM 内核微处理器,在工业控制等领域得到了广泛的应用,LPC ARM 是嵌入式系统中常用的较容易入门的 ARM 嵌入式处理器,掌握 LPC ARM 微处理器的软硬件开发技术是嵌入式软硬件开发人员应掌握的一种基本技能。

本书以 Keil C 为基础,将 ARM 理论知识与实际仿真演示相结合,使读者更容易理解 ARM 的基本概念、基本原理。同时,通过本书的学习,读者会逐步掌握 Proteus 仿真环境、Keil MDK 集成开发环境的基本使用,可以在自己的计算机上通过 Proteus 来搭建自己的创新设计平台。

本书是根据作者 4 年来讲授的研究生课程“嵌入式电子系统”讲稿整理而成的,为方便教师授课和学生学习,每一章可以安排 2~3 个学时,一共 12 章,适合 24~36 学时。

本书共分 12 章,各章主要内容如下。

第 1 章:嵌入式系统概述,主要讲述了嵌入式系统的基本概念、嵌入式系统的基本组成等。

第 2 章:ARM 微处理器内核分类及开发流程,主要介绍 ARM 内核分类、ARM 开发流程及开发环境,重点介绍了 ARM 初始化过程、RealView MDK 软件及其开发流程。

第 3 章:ARM 嵌入式微处理器体系结构及最小系统开发,主要介绍 ARM 微处理器的体系结构,使读者对 ARM 处理器工作模式、寄存器等有一个基本的了解,最后以 Philips 的 LPC21XX 系列为例,对 LPC ARM 最小系统的软硬件开发过程给以简要的描述。

第 4 章:LPC21XX 系列 ARM 芯片 GPIO 及其应用,主要介绍了 LPC ARM 的 GPIO 及其基本应用。

第 5 章:LPC213X 系列 ARM 芯片中断及应用,主要介绍 LPC ARM 中断的基本概念及相关的寄存器,针对外部中断给出应用。

第 6 章:LPC21XX 控制 1602 液晶应用初步,主要讲述 LPC ARM 控制 1602 液晶模块的方法。

第 7 章:LPC21XX 控制 12864 液晶应用初步,主要讲述 LPC ARM 控制 12864 液晶模块的基本应用,也是第 8 章的基础。

第 8 章: μ C/GUI 及其在 LPC21XX 上的基本应用,主要讲述 μ C/GUI 基础知识,并将其应用到 LPC21XX 上,对 12864 液晶模块的驱动进行了编写,并在 Proteus 中进行仿真。

第 9 章:LPC21XX 系列 ARM 芯片时钟管理与定时器应用,讲述了 LPC ARM 内部的时钟管理,讲述如何通过寄存器进行时钟的设置,介绍了 LPC 内部的定时器,给出了定时器/计数器的简单应用。

第 10 章：LPC 串行口通信及应用，介绍了 LPC ARM 内部串行口的基本结构、基本寄存器，给出了查询方式和中断方式两种工作方法的仿真实例。

第 11 章：RTX-51 实时多任务操作系统，主要介绍常见嵌入式实时操作系统的基本概念、嵌入式操作系统中的进程管理及进程通信等，还介绍了支持 LPC ARM 的 RL-ARM 操作系统并给出了简单应用仿真。

第 12 章： $\mu\text{C}/\text{OS-II}$ 嵌入式实时操作系统及基本应用仿真，对 $\mu\text{C}/\text{OS-II}$ 进行了初步的介绍，重点放在如何将 $\mu\text{C}/\text{OS-II}$ 移植到 LPC ARM 上，最后给出了简单的例子对其应用做了简单说明。

本书结构合理，文字流畅，通俗易懂，是学习 LPC ARM 的理想入门教材，具有计算机初步知识的读者都可快速入门。本书可作为研究生、本科生、专科生 ARM 课程教材或自学材料，也可供嵌入式系统相关技术人员参考。

本书的编写人员主要有于红旗、徐欣、刁节涛、孙兆林等人。其中，于红旗负责全书的内容组织和目录规划及编写，徐欣教授、孙兆林副教授提供了部分课件素材，刁节涛教授完成了全书的审校，田苗苗、张琨完成了书稿的整理及实验验证工作。书稿在整理过程中，也得到了徐晖、王玺、刘海军、聂洪山、李楠、张玉梅等老师的热情帮助，在此表示感谢。限于篇幅，本书没有对 ARM 处理器内部的各个组成部分进行全面的介绍，仅列举出有利于教学和实验的部分内容，加上嵌入式系统本身也在不断地发展变化之中，一些有益的科研项目和科研经验尚未在本书中体现，一些更深入的应用也未加入。限于作者水平，本书难免有疏漏之处，欢迎广大读者批评指正。同时，为采用本书作为教材的老师提供部分章节的 PPT 供参考。作者的联系方式是 dr.yhq@163.com。

本书在编写的过程中参阅了不少同行有益的教材，在此深表感谢！

编 者

目 录

第 1 章 嵌入式系统概述.....	1
1.1 身边的嵌入式系统.....	1
1.2 嵌入式系统概述.....	2
1.2.1 嵌入式系统的定义.....	2
1.2.2 嵌入式系统的特点.....	2
1.2.3 嵌入式系统与通用计算机系统的区别.....	3
1.3 嵌入式系统的基本结构.....	4
1.3.1 嵌入式系统的硬件组成.....	4
1.3.2 嵌入式系统的软件.....	7
1.3.3 嵌入式软件系统的设计方法.....	8
1.3.4 嵌入式系统的开发模式.....	11
1.3.5 嵌入式项目开发流程.....	11
1.4 嵌入式系统的发展.....	12
1.4.1 嵌入式系统的历史.....	12
1.4.2 嵌入式系统的发展现状.....	13
1.4.3 嵌入式系统的发展趋势.....	13
1.5 小结.....	14
第 2 章 ARM 微处理器内核分类及开发流程.....	15
2.1 ARM 处理器简介.....	15
2.1.1 ARM 处理器概述.....	15
2.1.2 ARM 微处理器的应用领域.....	15
2.1.3 ARM 微处理器的特点.....	16
2.2 ARM 内核分类.....	17
2.2.1 ARM7.....	18
2.2.2 ARM9.....	20
2.2.3 SecurCore 微处理器系列.....	21
2.2.4 ARM10.....	22
2.2.5 ARM11.....	22
2.2.6 StrongARM 微处理器系列.....	23
2.2.7 Xscale 处理器.....	23
2.2.8 ARM Cortex.....	23

2.2.9	ARM 流水线	28
2.3	ARM 微处理器的应用选型	30
2.3.1	ARM 微处理器内核的选择	31
2.3.2	系统的工作频率	31
2.3.3	芯片内存存储器的容量	31
2.3.4	片内外围电路的选择	31
2.3.5	多芯核 ARM 系列的选择	31
2.3.6	根据应用选型	31
2.4	ARM 程序设计基础	32
2.4.1	系统的初始化过程	32
2.4.2	定义程序入口点	33
2.4.3	设置异常向量	33
2.4.4	初始化存储器系统	33
2.4.5	初始化堆栈指针寄存器	34
2.4.6	初始化临界 I/O 设备	36
2.4.7	初始化 C 代码的运行环境	36
2.4.8	改变处理器的运行模式和状态	36
2.4.9	使能中断	36
2.4.10	进入 C 代码运行	37
2.5	RealView MDK 简介	37
2.5.1	RealView MDK 概述	37
2.5.2	产品模块介绍	38
2.5.3	ULINK2 仿真器	40
2.6	LPC2000 系列 ARM 芯片 GPIO 简介	42
2.6.1	GPIO (通用 I/O 口) 简介	42
2.6.2	I/O 口连接模块简介	43
2.6.3	GPIO 基本控制寄存器	44
2.6.4	GPIO 应用示例	45
2.6.5	Proteus 原理图	46
2.6.6	如何控制 LED	46
2.6.7	如何编程	47
2.7	RealView MDK 开发流程	47
2.8	其他常见的 ARM 开发环境及工具	49
2.9	小结	50
第 3 章	ARM 嵌入式微处理器体系结构及最小系统开发	51
3.1	ARM 微处理器体系结构	51
3.1.1	RISC 体系结构	51
3.1.2	ARM 微处理器工作模式	53
3.1.3	ARM 微处理器的寄存器结构	53
3.1.4	ARM 微处理器工作状态	58

3.1.5	ARM 微处理器的异常处理.....	58
3.1.6	ARM 处理器存储结构.....	59
3.1.7	ARM 处理器的存储映射 I/O	61
3.1.8	ARM 处理器的内部总线.....	61
3.2	LPC214X 系列 ARM 芯片应用开发.....	62
3.2.1	LPC214X 系列 ARM 芯片简介	62
3.2.2	LPC2148 管脚描述	64
3.2.3	LPC2148 最小系统设计	69
3.2.4	LPC2148 内置 Flash 的烧写.....	73
3.3	小结.....	77
第 4 章	LPC21XX 系列 ARM 芯片 GPIO 及其应用	78
4.1	LPC21XX 管脚功能设置.....	78
4.2	LPC21XX GPIO 相关寄存器	80
4.3	LPC21XX GPIO 基本应用	82
4.3.1	GPIO 控制 LED 闪烁.....	82
4.3.2	通过 GPIO 由按键控制 LED.....	84
4.3.3	软件译码点亮单个 7 段数码管	85
4.3.4	硬件译码点亮单个 7 段数码管	87
4.3.5	LPC 控制多个 7 段数码管	88
4.4	小结.....	91
第 5 章	LPC213X 系列 ARM 芯片中断及应用.....	92
5.1	什么是中断.....	92
5.2	为什么需要中断.....	92
5.3	与中断有关的几个概念.....	93
5.4	从一个中断例子讲起.....	94
5.4.1	管脚功能设置.....	94
5.4.2	中断初始化.....	95
5.4.3	主程序	95
5.4.4	中断服务程序.....	95
5.5	LPC213X 中断及相关寄存器.....	96
5.5.1	中断结构	96
5.5.2	中断源及其 VIC 中断号.....	96
5.5.3	外部中断相关寄存器.....	98
5.5.4	中断处理流程.....	105
5.5.5	中断使用注意事项.....	107
5.6	中断基本应用.....	107
5.6.1	实例一：两个外部中断控制两个 LED 闪烁	107
5.6.2	实例二：3 个外部中断控制数码管加减、复位	109
5.7	小结.....	110

第 6 章	LPC21XX 控制 1602 液晶应用初步	111
6.1	液晶显示器概述	111
6.2	1602 液晶显示模块概述	112
6.3	1602 液晶显示模块管脚定义	114
6.4	1602 液晶显示模块和 LPC 接口	115
6.5	LPC 对 1602 的 4 种操作模式	116
6.5.1	读操作时序	116
6.5.2	写操作时序	119
6.6	1602 液晶内部显示存储器	122
6.7	1602 液晶指令	123
6.7.1	显示功能设置	124
6.7.2	显示开关及光标设置	124
6.7.3	清屏	125
6.7.4	设定显示屏或光标移动方向指令	126
6.7.5	AC 值控制及屏幕移动设置	127
6.7.6	光标归位指令	128
6.7.7	指令小结及编程技巧	128
6.8	1602 液晶的初始化	128
6.9	1602 液晶综合应用	129
6.9.1	LCD 显示字符串	129
6.9.2	按键控制显示内容	131
6.10	小结	132
第 7 章	LPC21XX 控制 12864 液晶应用初步	133
7.1	12864 液晶模块概述	133
7.1.1	HD61202 的特点	134
7.1.2	HD61202 的引脚功能	135
7.1.3	HD61202 显示 RAM 地址	136
7.2	12864 液晶模块管脚定义	137
7.3	LPC 对 12864 液晶模块的操作时序	138
7.3.1	写操作时序	138
7.3.2	读操作时序	139
7.4	12864 液晶模块的指令	140
7.4.1	读状态指令	140
7.4.2	行设置指令	141
7.4.3	页设置指令	142
7.4.4	列地址设置指令	142
7.4.5	显示开/关指令	145
7.5	12864 液晶模块的初始化	145
7.6	12864 液晶模块综合应用	145
7.6.1	字模软件	145

7.6.2	12864 液晶模块显示数字.....	146
7.6.3	12864 液晶模块显示汉字.....	148
7.6.4	12864 显示图片.....	148
7.6.5	12864 液晶模块的绘图.....	149
7.7	小结.....	152
第 8 章	μ C/GUI 及其在 LPC21XX 上的基本应用.....	153
8.1	μ C/GUI 基础知识.....	153
8.1.1	μ C/GUI 简介.....	153
8.1.2	μ C/GUI 目标硬件.....	153
8.1.3	开发环境（编译程序）.....	154
8.1.4	μ C/GUI 的特点.....	154
8.1.5	屏幕和坐标.....	156
8.1.6	数据类型.....	156
8.2	μ C/GUI 入门指南.....	156
8.2.1	μ C/GUI 文件结构.....	156
8.2.2	向目标程序加入 μ C/GUI.....	157
8.2.3	将 μ C/GUI 的“C”文件加入工程中.....	157
8.2.4	配置 μ C/GUI.....	157
8.2.5	初始化 μ C/GUI.....	158
8.2.6	在目标硬件上使用 μ C/GUI.....	158
8.2.7	“Hello World”范例程序.....	159
8.3	μ C/GUI 在 LPC 单片机上的应用.....	160
8.3.1	获取 μ C/GUI3.98 源码.....	160
8.3.2	新建工程.....	160
8.3.3	修改配置文件.....	160
8.3.4	复制操作系统支持文件.....	161
8.3.5	代码移植.....	161
8.3.6	设置 include 路径.....	168
8.3.7	其他修改.....	169
8.3.8	Proteus 仿真.....	170
8.3.9	显示其他内容.....	171
8.3.10	文本显示相关函数.....	171
8.3.11	数值显示函数.....	175
8.3.12	绘图函数.....	177
8.3.13	字体相关函数.....	180
8.3.14	颜色相关函数.....	183
8.4	Keil 编译库的使用方法.....	183
8.4.1	去掉主函数文件.....	184

8.4.2	Lib 文件的使用.....	184
8.5	小结.....	185
第 9 章	LPC21XX 系列 ARM 芯片时钟管理与定时器应用	186
9.1	LPC21XX 系列 ARM 处理器外部时钟连接.....	186
9.1.1	ARM 处理器内部时钟结构.....	186
9.1.2	ARM 处理器内部时钟及相互关系.....	187
9.1.3	外设时钟相关寄存器.....	188
9.1.4	PLL 相关寄存器	188
9.1.5	PLL 设置过程	190
9.1.6	PLL 操作方法	191
9.1.7	PLL 使用注意事项.....	191
9.1.8	PLL 设定实例	191
9.2	LPC21XX ARM 定时器.....	192
9.2.1	定时器/计数器概述.....	192
9.2.2	相关寄存器.....	194
9.2.3	定时器工作流程.....	199
9.2.4	定时器操作方法.....	200
9.3	定时器综合应用.....	200
9.3.1	使用查询方式使 LED 闪烁.....	200
9.3.2	使用中断方式使 LED 闪烁.....	202
9.3.3	使用中断方式使数码管显示时间.....	203
9.3.4	使用中断刷新方式使数码管显示时间.....	206
9.4	小结.....	207
第 10 章	LPC 串行口通信及应用	208
10.1	串行通信基本概念.....	208
10.1.1	同步通信方式.....	208
10.1.2	异步通信方式.....	209
10.1.3	串行通信的数据传送方式.....	209
10.1.4	串行通信的奇偶校验.....	210
10.1.5	异步串行通信的波特率.....	210
10.1.6	串行接口标准.....	210
10.2	RS-232C 接口.....	210
10.2.1	接口信号.....	210
10.2.2	技术指标.....	212
10.2.3	RS-232 的帧结构	212
10.2.4	RS-232 的编程和使用	213
10.3	LPCARM 串口的编程与应用	213
10.3.1	LPC21XX 的串口管脚.....	213
10.3.2	串口基本结构.....	213

10.3.3	相关的寄存器及其作用.....	215
10.4	LPC 串口通信应用举例.....	219
10.4.1	实例一：串口向 PC 发送字符串.....	219
10.4.2	实例二：串口将接收到的数据发送给 PC（查询方式）.....	223
10.5	LPC21XX ARM 处理器串口通信（中断方式）.....	225
10.5.1	相关向量中断控制寄存器设置.....	225
10.5.2	串口相关寄存器.....	225
10.5.3	串口通信中断方式示例.....	229
10.5.4	实例三：串口将接收到的数据发给 PC（中断方式）.....	229
10.6	小结.....	230
第 11 章	RTX-51 实时多任务操作系统.....	231
11.1	操作系统基础知识.....	231
11.1.1	什么是计算机操作系统.....	231
11.1.2	操作系统的功能.....	231
11.1.3	进程和线程的基本概念.....	232
11.2	实时操作系统基础知识.....	235
11.2.1	实时系统及其特点.....	235
11.2.2	计算机实时操作系统.....	236
11.2.3	嵌入式实时操作系统.....	240
11.2.4	常见的嵌入式实时操作系统.....	244
11.3	RL-ARM 嵌入式实时操作系统及基本应用.....	246
11.3.1	多任务系统简介.....	246
11.3.2	RL-ARM 实时多任务系统任务状态及任务间通信.....	248
11.3.3	如何使用 RTX 内核.....	249
11.3.4	RTX 的配置.....	249
11.3.5	RTX 应用入门.....	250
11.4	小结.....	254
第 12 章	μ C/OS-II 嵌入式实时操作系统及基本应用仿真.....	255
12.1	μ C/OS-II 文件结构.....	255
12.2	μ C/OS-II 任务状态及其转换关系.....	257
12.3	μ C/OS-II 任务控制块.....	258
12.4	任务调度器.....	259
12.4.1	任务切换.....	259
12.4.2	就绪表.....	260
12.4.3	中断服务.....	260
12.4.4	时钟节拍.....	260
12.5	μ C/OS-II 的初始化和启动.....	260
12.6	任务管理.....	261
12.7	μ C/OS-II 任务间通信方式.....	263

12.7.1	信号量	263
12.7.2	互斥型信号量.....	264
12.7.3	事件标志组.....	265
12.7.4	消息邮箱.....	265
12.7.5	消息队列.....	266
12.8	μ C/OS-II 在 LPC 上的移植.....	267
12.9	小结.....	271
参考文献.....		272

1.1 身边的嵌入式系统

现代社会已经步入信息时代，我们周围处处都有嵌入式系统的踪影。据统计，欧美国家平均每人拥有的嵌入式处理器数量超过 10 个，而福特公司所出售的汽车所含的处理器计算能力，大于 IBM 公司出售的计算机的计算能力。试想一下我们周围含有嵌入式处理器的设备，以遥控器为例，里面就有一个单片机，能完成发送遥控指令的这一特定功能。汽车里面的嵌入式处理器数量更多，如图 1-1 所示为 VOLVO S80 汽车的 CAN 总线网络，每个功能模块都需要一个嵌入式处理器来完成通信、控制等特定的功能。

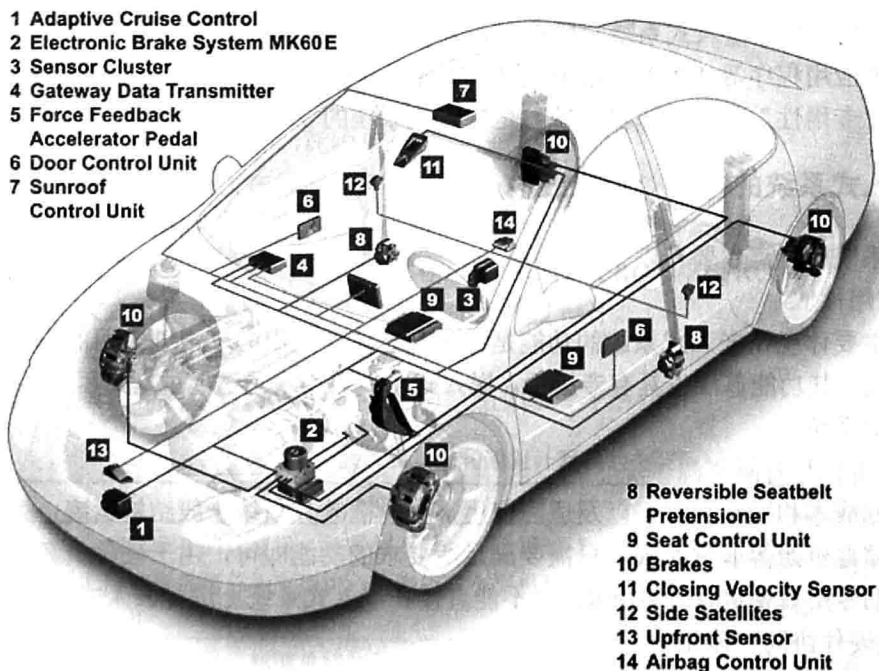


图 1-1 VOLVO S80 汽车的 CAN 总线网络

可以说，在我们的生活中，随处都有嵌入式系统的身影，从数字电视、信息家电、智

能玩具、手持通信设备到社区建筑的水、电、煤气表的远程自动抄表、防火防盗系统等，嵌入式系统无处不在，对我们生活的影响也日渐重要。其应用的广度和数量已远远超过通用计算机系统，计算机应用技术也开始进入一个被称为后 PC 技术的春天。掌握嵌入式系统的设计与实现技术，有利于激发并实现我们的创造力。

1.2 嵌入式系统概述

1.2.1 嵌入式系统的定义

嵌入式系统目前主流的定义方法有两种：一种是 IEEE（国际电气和电子工程师协会）给出的定义：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”（devices used to control, monitor, or assist the operation of equipment, machinery or plants）。可以看出，此定义是从应用上考虑的，嵌入式系统是软件和硬件的综合体，还可以涵盖机电等附属装置。

另一种定义是国内普遍认可的定义：“以应用为中心，以计算机技术为基础，软件、硬件可裁减，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。”根据该定义，嵌入式系统本身是一个可大可小的专用计算机系统，一个手持的 MP3、MP4 和一个 PC104 小型工控机都可以认为是嵌入式系统。

不管根据哪种定义方法，嵌入式系统都是采用“量体裁衣”的方式把所需的功能嵌入到各种应用系统中。嵌入式系统是一种计算机软件与硬件的综合体，特别强调“量身定做”的原则，针对某一种特殊用途开发出的截然不同的一项系统，这就是所谓的客定制化（Customized）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的应用程序等 4 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，“嵌入性”“专用性”与“计算机系统”是嵌入式系统的 3 个基本要素。

1.2.2 嵌入式系统的特点

嵌入式系统的特点主要有以下 4 种。

（1）专用性

专用性主要体现在嵌入式系统的功能是针对某一特定应用的，内部采用的也是专用的嵌入式处理器，其功能算法也具有专用性，嵌入式系统对用户是透明的，是用户“看不见”的专用计算机系统。

（2）小型化与有限资源

为了降低成本和减小功耗，以及适应应用需求，嵌入式系统一般结构紧凑，处理器运算能力、存储器资源等非常有限，只需要满足所要求的功能即可，由于硬件资源非常少，软件需要借助专用设备进行开发和更新，不能直接在嵌入式系统上进行嵌入式软件的开发。

（3）软硬件协同一体化

软件硬件协同设计（Co-design）的目标是将任务分解到硬件还是软件来实现。硬件实现时处理速度最快，但成本最高；软件实现时处理速度较慢，但成本最低。因此，需要在两者之间寻求一个平衡点。嵌入式系统由于有成本和功能的要求，需要在方案设计时划分

好软硬件各自实现的功能及软硬件实现方案，这可以通过软硬件协同软件来进行划分，由于这种软件价格非常昂贵，一般采用嵌入式可编程逻辑器件，在开发过程中，可以随时调整软硬件所做的任务。另外，由于嵌入式系统资源有限，其上运行的操作系统也经过了较大幅度的精简，一般将应用软件与操作系统进行一体化设计。

(4) 软件开发需要交叉开发环境

由于嵌入式系统本身资源有限，一般的开发方法是在 PC 机上编写程序，然后在 PC 机上编译、链接，最后生成在嵌入式系统上可执行的程序，通过烧写器或 JTAG 接口将程序下载到嵌入式系统中。这种软件开发方法就是交叉开发。这里采用的 PC 机为开发平台，也称为宿主机 (Host)，执行程序的嵌入式系统为执行机，也称为目标机 (Target)。宿主机和目标机之间一般通过 RJ45、RS-232、USB 等接口相连，以方便程序的下载和调试。

1.2.3 嵌入式系统与通用计算机系统的区别

嵌入式系统与通用计算机系统有着完全不同的技术要求和技术发展方向。通用计算机系统的技术要求是高速、海量的数值计算，其技术发展方向是总线速度的无限提升、存储容量的无限扩大；而嵌入式计算机系统的技术要求则是智能化控制，技术发展方向是与对象密切相关的嵌入性能、控制能力与控制可靠性的不断提高。

两者之间的区别主要体现在以下几个方面。

- 嵌入式系统一般专用于特定的任务，PC 是一个通用计算机。
- 嵌入式系统使用多种类型的处理器，PC 采用的处理器类型较少。
- 嵌入式系统极关注成本。
- 嵌入式系统有实时约束，即任务的执行时间是可以预测且满足功能要求的。
- 嵌入式系统使用实时多任务操作系统。
- 嵌入式系统软件故障造成的后果比 PC 系统更严重。
- 嵌入式系统大多有功耗约束。
- 嵌入式系统经常在极端的环境下运行。
- 嵌入式系统的系统资源比 PC 系统少得多。
- 嵌入式系统通常所有目标代码存放在 ROM/EPROM/EEPROM 中。
- 嵌入式系统需要专用工具和方法进行开发设计。
- 嵌入式系统的数量远远超过 PC。

嵌入式系统和通用计算机的主要区别总结如表 1-1 所示。

表 1-1 嵌入式系统和通用计算机的主要区别

	通用计算机	嵌入式系统
形式与类型	实实在在的计算机。按其体系结构、运算速度和规模可分为大型机、中型机、小型机和微机	“看不见”的计算机，形式多样，应用领域广泛，按应用进行分类
组成	通用处理器、标准总线 and 外设、软硬件相对独立	面向特定应用的微处理器，总线和外设一般集成在处理器内部，软硬件紧密结合
系统资源	系统资源充足，有丰富的编译器、集成开发环境、调试器等	系统资源紧缺，系统内没有编译器等相关开发工具

	通用计算机	嵌入式系统
开发方式	开发平台和运行平台都是通用计算机	采用交叉编译方式，开发平台一般是通用计算机，运行平台是嵌入式系统
二次开发性	应用程序可重新编程	一般不能重新编程开发
发展目标	编程功能电脑，普遍进入社会	变为专用电脑，实现“普及计算”

1.3 嵌入式系统的基本结构

从构成上来讲，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（可选），以及用户的应用软件系统等 4 个部分组成。也可以将这些部分抽象成硬件层、中间层、操作系统层和应用软件层，如图 1-2 所示。

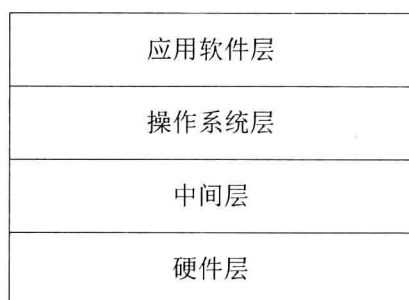


图 1-2 嵌入式系统的基本结构

1.3.1 嵌入式系统的硬件组成

嵌入式系统的硬件以嵌入式处理器为核心，配置必要的外围接口、外设部件。一般包括嵌入式处理器、存储器、I/O 系统和外设。如图 1-3 给出了典型的嵌入式系统的硬件组成图，以嵌入式处理器为核心，在处理器芯片内部一般集成了时钟管理、定时器、DMA、中断管理、I/O 端口、A/D 等功能模块，在芯片外围一般需要电源管理模块、时钟单元、复位电路、外部扩展端口、调试端口以及 SRAM、DRAM、Flash、ROM 等存储单元，人机之间主要通过外部设备进行互通，如键盘、鼠标、触摸屏、红外、LCD 液晶屏等。

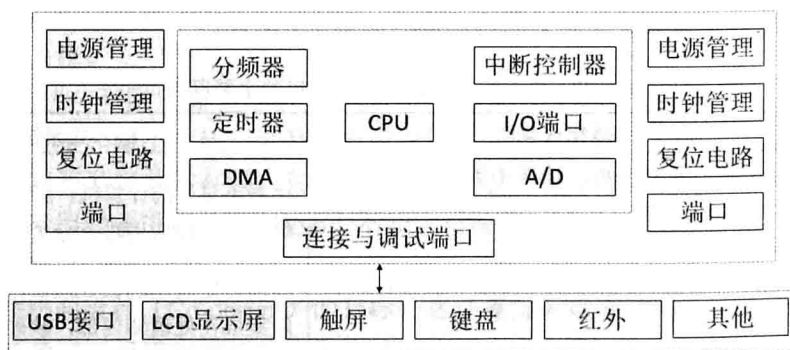


图 1-3 典型嵌入式系统硬件结构