

嵌入式系统课程教学实施方案项目规划教材
高等学校计算机专业特色教材

嵌入式 微控制器及其应用

*Applications of
Embedded Microcontroller*

薛丽萍 编著

高等教育出版社

嵌入式系统课程教学实施方案项目规划教材

高等学校计算机专业特色教材

嵌入式微控制器及其应用

Qianrushì Weikongzhìqì Jìqì Yìngyòng

薛丽萍 编著

高等教育出版社·北京

内容提要

本书以 Cortex-M3 内核的微控制器为例,系统介绍微控制器的体系结构、程序设计基础及驱动程序库、中断系统、定时器、常用接口设计、串行通信接口及网络接口应用编程、微控制器应用系统设计等内容。

本书取材于最新资料,提供较多应用实例,将抽象的知识与生动的实例相结合,详细讲解微控制器内部的主要功能模块,不面向寄存器编程,而是通过与各功能模块相关的驱动库函数、应用编程方法的介绍,结合典型应用实例,使读者快速掌握微控制器的原理与应用设计,具有很强的实用性。本书中的应用实例程序只需修改很少的部分或不用修改,即可在其他型号(除 LM3S9B92 外)的微控制器上运行。

本书内容丰富、实用,层次清晰,叙述详尽,方便教学与自学,可作为高等学校计算机类相关专业嵌入式系统课程的教材,也可作为相关工程技术人员参考书。

图书在版编目(CIP)数据

嵌入式微控制器及其应用/薛丽萍编著. -- 北京:高等教育出版社,2015.11
ISBN 978-7-04-043930-4

I. ①嵌… II. ①薛… III. ①微控制器 IV. ①TP332.3

中国版本图书馆CIP数据核字(2015)第223452号

策划编辑 时 阳	责任编辑 时 阳	封面设计 杨立新	版式设计 杜微言
插图绘制 邓 超	责任校对 杨凤玲	责任印制 田 甜	

出版发行	高等教育出版社	咨询电话	400-810-0598
社 址	北京市西城区德外大街4号	网 址	http://www.hep.edu.cn
邮政编码	100120		http://www.hep.com.cn
印 刷	北京市昌平百善印刷厂	网上订购	http://www.landaco.com
开 本	787mm×1092mm 1/16		http://www.landaco.com.cn
印 张	21	版 次	2015年11月第1版
字 数	490千字	印 次	2015年11月第1次印刷
购书热线	010-58581118	定 价	32.80元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换
版权所有 侵权必究
物料号 43930-00

出版说明

为推动嵌入式系统专业方向建设和课堂教学,建设嵌入式系统优质教学资源,解决嵌入式系统课程建设和教学中存在的问题,教育部高等学校计算机科学与技术专业教学指导分委员会、中国计算机学会教育专业委员会和高等教育出版社组织来自电子科技大学、浙江大学、国防科学技术大学、南京航空航天大学、上海交通大学、杭州电子科技大学、深圳大学、电子科技大学中山学院等在嵌入式系统专业方向建设和相关课程建设方面具有一定基础的高校的专家,开展了“嵌入式系统课程教学实施方案”项目研究。

项目研究工作自2010—2012年,历时两年,正式出版了《高等学校嵌入式专业方向核心课程教学实施方案》。通过将嵌入式系统人才培养目标,特别是能力培养目标分解、细化到不同课程,同时考虑到课程之间的前后衔接、课程与培养目标实现的支撑,《高等学校嵌入式专业方向核心课程教学实施方案》确定了该专业方向的5门核心课程:嵌入式系统概论、嵌入式微控制器及其应用、嵌入式操作系统、嵌入式硬件设计和嵌入式软件开发,并编写相应教材。这套教材旨在反映项目最新的研究成果,体现教学实施方案对于相应课程的定位和要求,以循序渐进的方式安排教学内容,讲解嵌入式系统平台设计的基本原理和基本方法,并给出典型性的示例。为保证教材质量,项目组确定由具有相应课程丰富教学和开发经验,并参与了课程教学实施方案研究的专家担任主编,同时邀请相关领域的权威专家对书稿进行了认真审读,数次修改,才最终交付出版。

这套教材具有如下特点:

- 内容全面,结构新颖。在内容上,理论与实践并重,涵盖课程教学实施方案中涉及的重点内容,并提供大量实例;在结构安排上,从基础知识入手,循序渐进;在叙述上,力求简明扼要,由浅入深。

- 原理部分通用性好。将教材的理论部分与实验部分分开,尽量减少理论内容对实验平台的依赖,以方便实际教学。

- 实验内容丰富、翔实。针对理论部分设计相应的实验,这些实验均系作者亲自设计、验证,力求展现细节,让读者易于上手,可以边干边学,具有较大的参考价值。

- 在拓展性上有突破。教材每章后都有相应的实验思考题和进一步探索的题目,供读者进一步深入研究。

- 面向实践,适用面广。教材兼顾教学、科研和工程开发的需要,对于广大高校本科生和研究生而言,是学习嵌入式系统的教科书;对于从事嵌入式系统开发的工程人员来说,则是实用的参考书。

我们希望这套教材的出版能够为国内高校嵌入式系统相关课程教学的开展提供有益的参

考和幫助，提升高校嵌入式系統的教学水平和開發水平，培養更多適應社會需求的嵌入式技術人才。我們將為此不懈努力，也希望得到各位讀者的熱情幫助，使這套教材能夠不斷完善和提高。

教育部高等學校計算機类专业教學指導委員會

中國計算機學會教育專業委員會

高等教育出版社

2015年1月

前 言

本书是为嵌入式系统专业方向 5 门核心课程中的“嵌入式微控制器及其应用”编写的教材，主要讲解微控制器的基本原理和应用设计方法，使读者了解微控制器的结构、功能部件及其应用编程，熟练使用相关微控制器开发工具，掌握微控制器应用开发的过程和技巧。本书包含微控制器的体系结构、程序设计基础及驱动程序库、中断系统、定时器、常用接口设计、串行通信接口及网络接口应用编程、微控制器应用系统设计等基本内容，使读者理解和掌握微控制器系统及其功能部件的结构特点、应用设计方法、编程技巧、调试过程等知识。在能力培养方面，使读者掌握微控制器的基本组成、工作原理和应用设计，获得微控制器应用系统设计的基本技能。在系统分析、设计、编程和调试的过程中，培养读者抽象问题、设计与选择解决方案的能力，具备微控制器应用系统软、硬件的设计能力。

全书共分 10 章。

第 1 章“微控制器概述”。首先介绍嵌入式微控制器的特点和发展概况，接着介绍微控制器的组成概貌和应用，并在此基础上介绍微控制器编程语言及开发环境。

第 2 章“Cortex-M3 体系结构与指令集”。以 32 位微控制器内核 ARM Cortex-M3 为例，介绍 Cortex-M3 的体系结构与编程模型、存储器系统、总线接口与总线矩阵，分类列出 Cortex-M3 的指令集。

第 3 章“微控制器的程序设计基础”。首先简要介绍汇编语言程序和 C 语言程序设计，接着以基于 Cortex-M3 处理器的 32 位 Stellaris 系列微控制器的驱动程序库为例，介绍 C 语言外设驱动程序库和 StellarisWare 图形库。同时以 IAR Embedded Workbench 集成开发环境为例，详细介绍微控制器软件开发环境的建立、仿真驱动程序的安装和应用程序的基本调试方法。

第 4 章“Cortex-M3 微控制器”。首先介绍 Cortex-M3 Stellaris 系列微控制器的组成，接着介绍复位控制、时钟控制、功率控制与系统控制以及微控制器最小系统组成，最后详细介绍通用输入输出端口（GPIO）及其应用编程。

第 5 章“中断/异常系统”。首先介绍中断系统的基础知识；接着介绍 Cortex-M3 处理器的中断/异常处理机制，包括嵌套向量中断控制器（NVIC）、中断/异常的类型、向量表、优先级、异常占先嵌套、末尾连锁、迟来中断处理等，并介绍中断响应的过程、复位和系统异常；最后以通用输入输出端口为例，详细介绍 GPIO 中断控制、中断应用程序编写方法和 GPIO 外部中断应用编程实例。

第 6 章“定时器”。首先介绍定时器的基础知识；接着详细介绍 Stellaris 系列微控制器的通用定时器模块（GPTM）、看门狗定时器（WDT）和脉宽调制模块（PWM），包括这些模块的结构与功能、寄存器、驱动库函数，同时给出这些模块的应用编程方法和应用实例。

第7章“微控制器的常用接口设计”。介绍基于 Stellaris 系列微控制器的键盘输入接口设计、LED 数码管接口设计、LCD/触摸屏接口设计、数/模转换接口设计，同时提供这些接口设计的编程实例。介绍模拟比较器和模/数转换器，包括它们的结构与功能、驱动库函数，同时给出相应的应用编程方法和应用实例。

第8章“串行总线接口及应用”。简要介绍通用异步收发器 (UART)、同步串行接口 (SSI)、I²C 总线接口、集成音频接口 I²S、通用串行总线 (USB) 等总线接口知识，详细介绍 Stellaris 系列微控制器所集成的 UART、SSI、I²C、I²S 和 USB 总线接口模块，包括这些接口模块的结构与功能、驱动库函数。同时给出这些接口的应用编程方法和应用实例。

第9章“网络接口及应用”。简要介绍控制器局域网 (CAN) 和以太网接口的相关知识，详细介绍 Stellaris 系列微控制器所集成的 CAN、以太网接口模块，包括这些接口的结构与功能、驱动库函数。同时给出这些网络接口的接口电路、编程方法、应用程序的关键结构与函数及应用编程实例。

第10章“微控制器应用系统设计”。介绍微控制器应用系统的设计与调试方法，从设计流程、需求分析、软件和硬件划分、硬件设计、软件设计、系统调试等6个方面介绍微控制器应用系统的开发过程。最后介绍环境温度检测设计实例。

全书配有20个实例，可以通过中国高校计算机课程网 (<http://computer.cncourse.com>) 下载相关实例代码 (带有详细注释和说明)。20个实例几乎涵盖了32位微控制器的所有内部资源，每个实例都在 LM3S9B92 开发板上调试过。对于没有该开发板的读者，可以使用自己的一套开发板，程序代码一般是通用的，只需对底层驱动程序稍作修改即可。

本书在《高等学校嵌入式专业方向核心课程教学实施方案》的指导下进行，得到了高等教育出版社、深圳大学计算机与软件学院及许多专家和学者的大力支持，并听取了多方面的意见和建议。深圳大学纪震教授做了很多组织和指导工作，研究生刘媛、陈泽楷、姚应龙参与了大量工作。电子科技大学中山学院李文生教授审阅了本书大纲，山东大学陈桂友教授详细审阅了全稿。本书在编写过程中参考了大量的国内外著作和资料，得到了 TI 公司的真诚帮助。深圳大学计算机与软件学院的领导和同事们也给予了真切的关心、指导和热情帮助。在此向所有关心、支持本书出版的朋友一并表示衷心的感谢！

本书在编写过程中已尽全力保证内容特别是实例的正确性，但由于时间匆忙，且作者自身水平有限，仍难免存在错误。请各位读者不吝赐教，以便我们在改版或再版时及时纠正、补充。

再次感谢所有关心嵌入式事业、嵌入式教学以及本书出版工作的朋友们！

作 者

2015年4月

目 录

第 1 章 微控制器概述	1	2.5.1 指令格式	27
1.1 微控制器和嵌入式系统	2	2.5.2 指令分类	28
1.2 微控制器的特点和发展	2	本章小结	37
1.2.1 微控制器的特点	2	习题与思考题	37
1.2.2 微控制器的发展历程	3	第 3 章 微控制器的程序设计基础	39
1.2.3 微控制器的发展趋势	4	3.1 汇编语言程序	40
1.3 微控制器的组成概貌	5	3.1.1 汇编语言格式	40
1.4 微控制器的应用	8	3.1.2 汇编程序构成	40
1.5 微控制器编程语言和开发环境概述	9	3.2 C 语言程序设计	41
1.5.1 微控制器编程语言概述	9	3.2.1 C 语言程序设计的特点	41
1.5.2 微控制器开发工具概述	9	3.2.2 C 语言程序设计概述	41
本章小结	10	3.3 驱动程序库	43
习题与思考题	10	3.3.1 Stellaris 驱动程序库概述	43
第 2 章 Cortex-M3 体系结构与指令集	11	3.3.2 引导代码概述	44
2.1 Cortex-M3 处理器结构	12	3.3.3 利用驱动程序库的编程方法	48
2.2 Cortex-M3 编程模型	14	3.3.4 StellarisWare 图形库	49
2.2.1 工作模式和工作状态	14	3.3.5 基本图形绘制与应用	51
2.2.2 数据类型和存储器格式	14	3.3.6 应用控件实现用户输入和显示	53
2.2.3 寄存器组	15	3.4 微控制器软件开发环境	56
2.2.4 特权级访问和用户级访问	19	3.4.1 IAR Embedded Workbench 微控制器	
2.2.5 Cortex-M3 堆栈	21	集成开发环境的建立	56
2.3 存储器系统	22	3.4.2 IAR Embedded Workbench 集成开发	
2.3.1 存储器映射	22	环境的使用	58
2.3.2 存储器接口	24	3.4.3 IAR Embedded Workbench 软件仿真	65
2.3.3 位寻址操作	24	本章小结	68
2.4 总线接口和总线矩阵	25	习题与思考题	68
2.4.1 AHB-Lite 和 APB 总线概述	25	第 4 章 Cortex-M3 微控制器	70
2.4.2 总线接口	26	4.1 32 位微控制器组成	71
2.4.3 总线矩阵	26	4.2 32 位微控制器内部结构	72
2.5 指令集	27	4.3 系统控制单元	76

4.3.1 复位控制	76	5.12.2 GPIO 外部中断应用编程	115
4.3.2 时钟控制	77	本章小结	117
4.3.3 功率控制	80	习题与思考题	117
4.3.4 系统控制	80	第 6 章 定时器	119
4.4 微控制器最小系统	82	6.1 定时器概述	120
4.5 通用输入输出端口	84	6.2 通用定时器模块结构与功能	121
4.5.1 GPIO 的基本特性	84	6.2.1 GPTM 内部结构	121
4.5.2 GPIO 寄存器组	85	6.2.2 GPTM 功能	123
4.5.3 GPIO 基本操作及相关函数	86	6.3 GPTM 的寄存器及库函数	123
4.5.4 GPIO 模式控制及相关函数	88	6.3.1 定时器的配置与运行控制	124
4.5.5 GPIO 引脚配置及相关函数	89	6.3.2 定时器中断	127
4.5.6 GPIO 应用编程实例	90	6.3.3 定时器内容的设置与获取	128
本章小结	92	6.4 定时器工作模式	131
习题与思考题	92	6.4.1 32 位定时器工作模式	131
第 5 章 中断/异常系统	94	6.4.2 16 位定时器/计数器工作模式	132
5.1 中断系统概述	95	6.5 通用定时器应用实例	136
5.2 嵌套中断向量控制器	96	6.6 看门狗定时器	140
5.3 中断/异常类型	98	6.6.1 看门狗定时器的结构与功能	140
5.4 向量表	101	6.6.2 看门狗定时器的寄存器与库函数	141
5.5 中断/异常优先级	102	6.6.3 看门狗定时器应用编程实例	143
5.5.1 优先级	102	6.7 脉宽调制器	146
5.5.2 优先级分组	102	6.7.1 PWM 内部结构	146
5.6 中断/异常嵌套	103	6.7.2 PWM 寄存器映射	148
5.6.1 异常占先	103	6.7.3 PWM 功能描述	149
5.6.2 末尾连锁	104	6.7.4 PWM 库函数	152
5.6.3 迟来中断处理	104	6.7.5 PWM 应用编程实例	154
5.6.4 中断嵌套及优先级处理原则	104	本章小结	156
5.7 中断/异常的响应过程	105	习题与思考题	157
5.7.1 中断/异常的响应序列	105	第 7 章 微控制器的常用接口设计	158
5.7.2 中断/异常返回序列	106	7.1 键盘和 LED 数码管接口设计	159
5.8 复位	107	7.1.1 矩阵式键盘接口设计	159
5.9 其他系统异常	108	7.1.2 键盘接口应用编程	160
5.10 外部中断	110	7.1.3 数码管	164
5.11 GPIO 中断控制	112	7.1.4 数码管接口设计实例	164
5.12 中断应用编程	113	7.2 LCD/触摸屏接口设计	169
5.12.1 中断应用程序的基本编写方法	113	7.2.1 LCD/触摸屏概述	169

7.2.2	QVGA 彩色 LCD 触摸屏接口设计	170	8.3.4	SSI 的 FIFO 操作及相关函数	217
7.2.3	LCD/触摸屏应用编程实例	172	8.3.5	SSI 中断及相关函数	218
7.3	模拟比较器	174	8.3.6	SSI 应用实例——Flash 存储器 扩展和 SD 卡	219
7.3.1	模拟比较器结构和功能	174	8.4	I ² C 接口	225
7.3.2	模拟比较器的常见用法	175	8.4.1	I ² C 总线概述	225
7.3.3	模拟比较器寄存器映射	176	8.4.2	I ² C 的接口功能与内部结构	227
7.3.4	内部参考电压编程	176	8.4.3	I ² C 中断和主机收发形式	229
7.3.5	比较器的配置及库函数	178	8.4.4	I ² C 库函数	230
7.3.6	模拟比较器应用实例	180	8.4.5	I ² C 应用编程	233
7.4	模/数转换器	186	8.5	I ² S 接口	234
7.4.1	ADC 结构与功能	186	8.5.1	I ² S 总线概述	234
7.4.2	采样序列及相关函数	187	8.5.2	I ² S 的特性与内部结构	236
7.4.3	ADC 模块控制及相关函数	188	8.5.3	I ² S 发送	237
7.4.4	硬件采样平均电路	191	8.5.4	I ² S 接收	238
7.4.5	内部温度传感器	191	8.5.5	I ² S 库函数	239
7.4.6	ADC 应用编程实例	192	8.5.6	I ² S 音频芯片及应用实例	241
7.5	数/模转换器	195	8.6	USB 接口	246
7.5.1	数/模转换器概述	195	8.6.1	USB 概述	246
7.5.2	数/模转换器 TLV5616	196	8.6.2	USB 模块内部结构与特性	249
7.5.3	DAC 接口设计与应用实例	196	8.6.3	用作 USB 设备	250
本章小结		199	8.6.4	用作 USB 主机	252
习题与思考题		200	8.6.5	OTG 模式	254
第 8 章 串行总线接口及应用		201	8.6.6	USB 初始化和配置	255
8.1	串行通信基础	202	8.6.7	USB 库函数	256
8.1.1	串行通信基本方式	202	8.6.8	USB OTG 应用实例	261
8.1.2	串行通信制式和多机通信	203	本章小结		274
8.2	通用异步串行通信	203	习题与思考题		275
8.2.1	UART 的内部结构与特性	203	第 9 章 网络接口及应用		276
8.2.2	UART 的控制及相关函数	204	9.1	CAN 总线通信接口	277
8.2.3	UART 数据收发	206	9.1.1	CAN 总线概述	277
8.2.4	UART 中断及相关函数	208	9.1.2	CAN 模块的特性与内部结构	278
8.2.5	UART 应用编程实例	209	9.1.3	CAN 初始化	279
8.3	同步串行接口	212	9.1.4	CAN 操作	280
8.3.1	同步串行接口概述	212	9.1.5	CAN 发送	281
8.3.2	SSI 的内部结构与特性	212	9.1.6	CAN 接收	281
8.3.3	SSI 控制及相关函数	213			

9.1.7 CAN 中断处理	283	10.1.2 需求分析	306
9.1.8 CAN 位处理	283	10.1.3 软件和硬件划分	307
9.1.9 CAN 库函数	284	10.1.4 硬件设计	307
9.1.10 CAN-bus 接口电路与应用编程	286	10.1.5 软件设计	309
9.2 以太网接口	292	10.2 微控制器应用系统调试	311
9.2.1 以太网接口概述	292	10.2.1 调试工具	311
9.2.2 以太网控制器特性与内部结构	292	10.2.2 调试方法	312
9.2.3 功能描述	293	10.3 环境温度检测设计实例	313
9.2.4 以太网库函数	296	10.3.1 设计要求	313
9.2.5 以太网接口电路与应用编程	297	10.3.2 硬件设计	313
本章小结	304	10.3.3 软件设计	314
习题与思考题	304	10.3.4 程序运行实时追踪调试	319
第 10 章 微控制器应用系统设计	305	本章小结	322
10.1 微控制器应用系统设计方法	306	习题与思考题	322
10.1.1 设计流程	306	参考文献	324

第 1 章 微控制器概述

嵌入式微控制器发展非常迅速，从最初的 4 位、8 位、16 位到现在的 32 位微控制器，产品多达数百种。本章首先讲述嵌入式微控制器的特点、发展概况和组成概貌，然后简要讲述微控制器的应用、编程语言及开发环境。

1.1 微控制器和嵌入式系统

微控制器（MicroController Unit, MCU）是将计算机所有基本功能部件（如运算器、控制器、存储器等）及其外部设备（以下简称外设）集成在一个芯片上构成的一个微型计算机系统。微控制器通常以某种微处理器内核为核心，在芯片上集成存储器（RAM、Flash）、总线逻辑、定时器/计数器、脉宽调制输出、各种输入输出（Input/Output, I/O）接口、串行通信接口、模/数转换器（ADC）、数/模（D/A）转换等各种功能模块和外围部件。微控制器的最大特点是单片化（也称之为单片机）、体积小，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比较丰富，适合用于控制，因此称为微控制器。

嵌入式系统是用来控制或监视机器、装置或工厂等并嵌入应用系统中的计算机系统，实时性是其主要特征。嵌入式系统以应用为中心，以计算机技术为基础，软硬件可裁减，从而能够适应实际应用中对功能、可靠性、成本、物理尺寸和功耗等的严格要求，因此比通常的计算机应用的设计更为复杂，涉及面也更为广泛。嵌入式系统在实际应用数量上远远超过了各种通用计算机。

微控制器是典型的嵌入式系统。它应嵌入式系统应用的要求而生，并以嵌入式应用为唯一目的。微控制器从体系结构到指令集都是按照嵌入式应用的特点而专门设计的，能最大限度地满足面向控制对象、应用系统的嵌入、现场的可靠运行及控制品质的各种要求，成为发展最快、品种最多、数量最大的嵌入式系统。

微控制器根据存储器结构可分为哈佛结构和冯·诺依曼结构，根据指令结构又可分为复杂指令集计算机（Complex Instruction Set Computer, CISC）和精简指令集计算机（Reduced Instruction Set Computer, RISC）微控制器。从微控制器诞生至今，随着技术的发展和制造工艺的成熟，其成本越来越低，性能却越来越强大，这使其应用已经无处不在，遍及各个领域。

与微处理器相比，微控制器具有成本低的优势。采用微控制器开发应用系统可使系统中芯片的数量减少，系统发生故障的概率降低，可靠性提高，系统的体积缩小。这些优势对于需要嵌入其他设备或装置中的系统具有重要意义。微控制器是目前嵌入式系统工业的主流，占嵌入式系统市场的份额高达 70%。微控制器在品种数量上远远超过微处理器，微控制器的生产厂商也远远超过微处理器的生产厂商。

1.2 微控制器的特点和发展

1.2.1 微控制器的特点

微控制器随着大规模集成电路技术的发展，功能越来越强，具有性能高、速度快、体积小、价格低、稳定可靠、应用广泛、通用性强等突出优点，因此在各个领域都得到了迅猛发展。微控制器主要有如下特点。

1. 高集成度、体积小和高可靠性

微控制器将各种功能部件集成在一块芯片上，集成度很高，同时却具有较小的体积。芯片本身是按照工业测控环境要求设计的，内部布线很短，其抗工业噪声性能优于一般通用的 CPU。微控制器的程序指令、常数及表格等固化在 ROM 中，不易被破坏；许多信号通道均在一块芯片内，可靠性较高。使用低功耗、低电压的微控制器还可以简化电源电路，降低对散热的要求，提高系统的可靠性，也便于生产便携式产品。

2. 控制功能强

为了满足对对象的控制要求，微控制器的指令集均有极丰富的条件/分支转移能力、I/O 接口的逻辑操作及位处理能力，非常适合用于专门的控制功能。

3. 易扩展

微控制器的芯片内具有计算机正常运行所必需的部件，芯片外部有许多可供扩展的三总线及并行、串行 I/O 引脚，很容易构造各种规模的计算机应用系统。

4. 可供选择的种类较多

根据实际系统的功能要求，可以选择合适类型的微控制器。例如，通用型微控制器一般控制功能强，指令集具有极丰富的转移指令、I/O 端口的逻辑操作及位处理功能，可满足工业控制的要求；而专用型微控制器则可满足一些特殊要求。

5. 优异的性能价格比

为了提高速度和运行效率，微控制器已开始使用 RISC 流水线和数字信号处理（Digital Signal Processing, DSP）等技术。微控制器的广泛使用降低了系统实现的成本，如开发成本、生产管理成本、产品维护成本，各大公司的商业竞争更使其价格十分低廉，性能价格比优异。

1.2.2 微控制器的发展历程

嵌入式微控制器发展非常迅速，从最初的 4 位、8 位、16 位到现在的 32 位微控制器，产品多达数百种。主流和应用最多的是 8 位、16 位和 32 位的微控制器。

1. 4 位微控制器

4 位微控制器的控制功能较弱，一次只能处理 4 位二进制数。4 位微控制器重点用于电子计算器、各种形态的智能单元、遥控器和家用电器中。典型产品有 NEC 公司的 μ PD75 系列、NS 公司的 COP400 系列、松下公司的 MN1400 系列、Rockwell 公司的 PPS/I 系列、Fujitsu 公司的 MB88 系列、SHARP 公司的 SMxx 系列、TOSHIBA 公司的 TMP47 系列等。这类微控制器的典型应用领域有：微型计算机的输入设备（鼠标、游戏杆）、电池充电器（镍镉电池、锂电池）、运动器材、一般家用电器的控制及遥控器、玩具控制、计时器、时钟、计算器和多功能电话等。

2. 8 位微控制器

最早推出 8 位微控制器的是 Intel 公司。继 1976 年推出 MCS-48 后，Intel 公司又于 1980 年推出了 MCS-51，为发展兼容性好的新一代微控制器奠定了良好的基础。在 8051 技术实现开放后，中国宏晶、Philips、Atmel、Dallas 和 Siemens 等公司纷纷推出了基于 80C51 内核的微控制器。与 4 位微控制器相比，8 位微控制器不仅具有较大的存储容量和寻址范围，而且中断源、

并行 I/O 接口和定时器/计数器个数都有了不同程度的增加,并集成有全双工串行通信接口。在指令集方面,普遍增设了乘除指令和比较指令。在中国应用较广的 8 位微控制器还有 Motorola 公司的 68HC 系列。Microchip 公司率先推出了 RISC、哈佛结构和 2 级流水线的 PIC 系列。Atmel 公司推出了 AVR 系列。ST 公司推出了高性能 STM8 系列。

8 位微控制器应用于家用电器、汽车电子设备、计算机外设、智能仪器仪表、过程控制、通信、办公自动化系统等许多领域。

3. 16 位微控制器

在 8 位微控制器的基础上,16 位微控制器于 1983 年诞生,其运行和数据吞吐能力比 8 位微控制器有了较大提高。比较常见的 16 位微控制器有 Intel 公司的 MCS-96/196 系列、德州仪器(Texas Instruments, TI)公司的 MSP430 系列、飞思卡尔(Freescale)公司的 MC9S12 系列等。

16 位微控制器主要应用于工业控制、智能仪器仪表、便携式设备、家用电器及作为计算机外部设备等。

4. 32 位微控制器

32 位微控制器是近几年推出的新型微控制器,运算速度和精度较高,完全支持高级编程语言,片上通常集成微处理器内核、定时器、A/D、D/A、脉宽调制器、中断控制器、串行接口、JTAG 测试接口和外围接口适配器等,成为嵌入式系统开发的理想硬件平台。近年来,随着家用电子设备的发展,32 位微控制器的应用越来越广泛。

目前,众多厂商相继推出了 32 位微控制器产品。ARM 公司设计的 Cortex-M3 处理器为高性能、低成本的微控制器提供了一个能够满足小存储、少引脚数以及低功耗三方面要求的内核。与此同时,它还提供出色的计算性能和优越的系统中断响应能力。基于 Cortex-M3 内核的 32 位微控制器有恩智浦(NXP)公司的 LPC17xx 系列、TI 公司的 Stellaris 系列、意法半导体的 STM32 系列、Atmel 公司的 SAM3 系列等。这些 32 位微控制器具有通用的架构、简易的开发流程、丰富的模拟外设及通信接口、丰富的设计资源和低廉的价格。基于 Cortex-M3 内核的 32 位微控制器正在成为微控制器的主流,因此,本书基于 Cortex-M3 内核的 32 位微控制器介绍相关内容。

1.2.3 微控制器的发展趋势

随着科学技术的不断进步,制造工艺和集成度不断提高,微控制器正朝着多功能、高性能的方向发展,包括与 DSP 融合,向片上系统(System on Chip, SOC)方向发展,采用 RISC 微处理器结构等。

1. 微型单片化和 SOC

现在的产品普遍要求体积小、重量轻,这就要求微控制器除了功能强、功耗低外,还要求其体积小。SOC 概念和应用技术的出现为各种应用提供了新的实现技术,标志着现代电子技术应用进入 SOC 阶段。SOC 也为微控制器技术提供了更广阔的应用范围。

2. 高整合度

一般来说,微控制器多半会配备模拟/数字转换器、数字/模拟转换器、内建 NAND 型快闪存储器、内建 ROM(只读存储器)、特定的通信接口(和 USB、以太网等外设通信)、DRAM

控制接口。另外，更多以前放在微控制器外部的混合信号元件也逐步被加入微控制器中，如比较器、放大器、滤波器和内部振荡器等。

3. 结合部分 DSP 功能

随着技术的进步，微控制器开始针对数字信号处理设计专用线路。数字信号处理通常包括相当多的卷积运算、矩阵运算等，用来处理数学上的拉普拉斯运算或傅里叶变换，广泛地应用于多媒体领域，这些运算若以专用的硬件实现，就是一系列的乘法器（Multiplier）与桶形移位器（Barrel Shifter）。目前比较高端的微控制器会内建专用的 DSP 计算线路，以加强数字信号处理的能力。

4. 广泛采用 Flash 存储技术

Flash 存储技术的应用彻底改变了微控制器系统的实现、调试和维护方法。采用 Flash 实现的程序存储器，使得微控制器提供在线编程能力成为可能。目前，Flash 形式的微控制器日渐增多，已经成为市场的主流。采用 Flash 的微控制器可以多次读/写，方便需要重复加载程序的应用场合，同时可以减少软件出错、改版或售后召回时的维修成本。

5. RISC 将成为主流架构

RISC 处理器采用专用的硬件线路来处理各个不同的指令，因此 RISC 架构的微处理器可以达到极高的运行效率与较低的功耗，缺点则是当指令集中的指令数目增多时，必须准备更多的硬件线路来处理指令，会造成芯片面积增加，同时增加芯片成本。由于微控制器的应用场合通常用到的指令数目较少，所用到的程序也必须较为精简，因此 RISC 架构特别适用于微控制器，有可能成为微控制器应用领域的主流架构。

1.3 微控制器的组成概貌

微控制器的组成与普通计算机很相似。图 1-1 所示为微控制器的典型组成概貌。微控制器基本由 CPU、程序存储器、数据存储器、总线、各种 I/O 部件、基本功能单元（定时器/计数器等）等组成。各主要部件通过内部总线连成一体，所有部件集成在一片集成电路内。

1. CPU

CPU 是整个微控制器的核心，通常是某一种微处理器内核，有些内核中包含中断控制系统，如 Cortex-M3 内核。有些微控制器的 CPU 中不包含中断控制系统，如 80C51 微控制器。

2. 系统控制单元

系统控制单元配置微控制器的整个操作，它不但提供器件信息，还处理复位控制、功率控制、时钟控制和低功耗模式等。

复位控制包括：①上电复位（POR）；②外部复位输入引脚（RST）有效；③内部掉电（BOR）检测器；④软件启动的复位；⑤看门狗定时

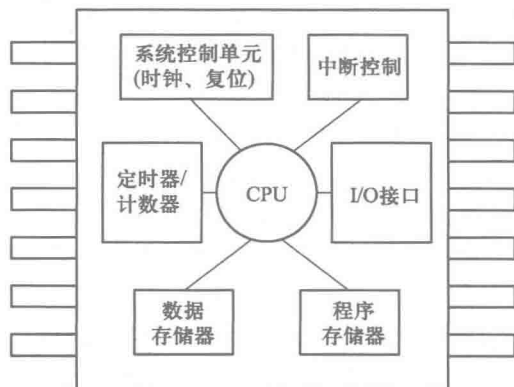


图 1-1 微控制器的典型组成概貌

器复位；⑥主振荡器检验失败。

时钟控制时钟源和时钟配置。时钟源主要由振荡器实现，振荡器产生整个系统所需的时钟。通常，振荡器是内部振荡器，需要外接一个晶体。内部振荡器还可外接其他元件（如 RC），或直接由外部振荡源驱动。高性能的微控制器可提供多种不同频率的内部振荡器。

高性能的微控制器还提供功率控制，例如一个集成的 LDO 稳压器用来对微控制器的大多数内部逻辑电路供电。为降低功耗，可以用一个不可编程的 LDO 将微控制器的 3.3 V 输入电压变为 1.2 V。

高性能的微控制器还提供低功耗模式，可控制微控制器在运行模式、睡眠模式和深度睡眠模式时系统每个外设或模块的时钟门控逻辑。

3. 程序存储器

程序存储器用来固化微控制器的应用程序和一些表格常数。按照不同结构，微控制器内部程序存储器可分为 Mask ROM 型、EPROM 型、ROM Less 型、OTP ROM 型和 Flash Memory 型。前三种程序存储器的微控制器是早期产品，目前已较少使用，现在的趋势是 Flash Memory 型程序存储器。

4. 数据存储器

随机存储器（RAM）是一种可读/写的存储器。微控制器内部的 RAM 除了作为工作寄存器、位标志和堆栈区以外，其他单元都可以作为数据存储器，存放输入的数据或运算的结果。

5. 总线

总线包括地址总线（Address Bus, AB）、数据总线（Data Bus, DB）和控制总线（Control Bus, CB）。其中，地址总线为进行数据交换提供地址，将地址传送至存储器或 I/O 接口；数据总线在 CPU 与存储器或 I/O 接口之间或者存储器与外设之间交换数据；控制总线用于传送控制信号、应答信号和状态信号等。

在微控制器中，总线通常包括片内总线和片外总线。片内总线用于连接微控制器内部的器件，而片外总线则提供用于扩展外部器件的总线接口。目前，有些微控制器不再提供片外总线，扩展外部器件可采用串行总线接口，如 UART、I²C、SPI、USB 等。

6. 中断控制

有些微控制器的 CPU 中不包含中断控制系统。通常，32 位微控制器的 CPU 内核中包含中断控制系统。

7. 定时器/计数器

定时器/计数器在嵌入式应用中是必不可少的，当采用系统时钟作为输入时称为定时器，而将外部信号作为输入时则称为计数器。定时器/计数器已成为微控制器的标准部件，几乎所有的微控制器都集成了定时器/计数器。定时器的个数取决于微控制器的种类及型号。有些微控制器还集成了可编程计数器阵列（Programmable Counter Array, PCA），与标准的定时器/计数器相比，它需要较少的处理器干预，用于提供增强的定时/计数功能。

8. 各种 I/O 接口

I/O 接口是计算机的输入输出接口。微控制器所提供的 I/O 接口种类繁多，与微控制器的型