



国外经典教材·计算机科学与技术

PEARSON

Prentice  
Hall

Microcontroller Technology:  
The 68HC11 and 68HC12,  
Fifth Edition

# 微控制器原理与 应用 (第5版)

(加) Peter Spasov 著  
李小洪 译



清华大学出版社

国外经典教材·计算机科学与技术

# 微控制器原理与应用

(第5版)

(加) Peter Spasov 著  
李小洪 译



清华大学出版社

北京

## 内 容 简 介

本书着重介绍微控制器的原理及其应用。全书共 5 部分。第 1 部分概述微控制器技术, 第 2 部分介绍如何从编程者的角度进行开发。第 3 部分着重介绍总线概念和操作模式、硬件和时钟操作。第 4 部分侧重于几个主题: 接口连接的概念、并行输入/输出、串行子系统、可编程定时器操作和模拟转换子系统。第 5 部分概述微控制器的应用控制软件、具体应用以及 8 位微控制器系列产品。最后提供 8 个很有参考价值的附录。

本书是为高等院校本科生、研究生编写的教材, 同时也可供从事微控制器产品开发的工程技术人员和业余爱好者自学参考。

Simplified Chinese edition copyright © 2005 by **PEARSON EDUCATION ASIA LIMITED and TSINGHUA UNIVERSITY PRESS.**

Original English language title from Proprietor's edition of the Work.

Original English language title: **Microcontroller Technology: The 68HC11 and 68HC12, Fifth Edition** by Peter Spasov Copyright © 2005, 2001

EISBN: 0-13-112984-8

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Pearson Education.

This edition is authorized for sale only in the People's Republic of China (excluding the Special Administrative Region of Hong Kong and Macao).

本书中文简体翻译版由 Pearson Education 授权给清华大学出版社在中国境内(不包括中国香港、澳门特别行政区)出版发行。

北京市版权局著作权合同登记号 图字: 01-2004-6253

版权所有, 翻印必究。举报电话: **010-62782989 13501256678 13801310933**

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

### 图书在版编目(CIP)数据

微控制器原理与应用(第 5 版)/(加)斯帕索夫(Spasov, P); 李小洪译. —北京: 清华大学出版社, 2006.1 (国外经典教材·计算机科学与技术)

书名原文: **Microcontroller Technology: The 68HC11 and 68HC12, Fifth Edition**

ISBN 7-302-11800-0

I. 微… II. ①斯…②李… III. ①微控制器—教材 IV. TP332.3

中国版本图书馆 CIP 数据核字(2005)第 104523 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客 户 服 务: 010-62776969

文稿编辑: 文开棋

封面设计: 久久度文化

印 刷 者: 清华大学印刷厂

装 订 者: 三河市金元装订厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印 张: 37.25 字 数: 900 千字

版 次: 2006 年 1 月第 1 版 2006 年 1 月第 1 次印刷

书 号: ISBN 7-302-11800-0/TP·7675

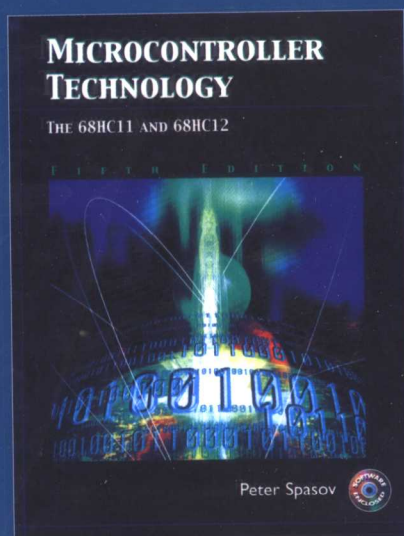
印 数: 1~4000

定 价: 68.00 元

## (加) Peter Spasov 著

### 作者简介

Peter Spasov, 加拿大安大略省弗来明大学应用计算和工程科学系教授。他具有丰富的教学和实践经验。多年来一直担任该校Applied Projects的协调人。他主持并领导着CFI/OIT Applied Technologies for Healthy Aging Research和CITO Ubiquitous Location Tracking项目。著作有*Microcontroller Technology: The 68HC11*和*Programming for Technology Students Using Visual Basic*。在进入大学任职之前,他是Gulf Canada公司的仪表工程师。Spasov教授喜欢徒步旅行、越野滑雪和划皮划艇。



## 李小洪 译

### 译者简介

李小洪，毕业于西安电子科技大学电子工程系自动控制专业，1996年获学士学位，同年分配到电子工业部36所工作，主要从事单片机软件硬件的开发、设计和研究工作。熟悉51系列和PIC等单片机。代表译著有《Keil Software\_Cx51 编译器用户手册》和《微处理器与微型计算机》（第6版）。

# 译者序

自 20 世纪 80 年代后期以来,摩托罗拉的 68HC11 单片机在诸多领域(汽车电子、通信、计算机外设、工业控制以及消费类电子产品)中得到了广泛的应用。十余年来,已发展到十余个系列,几十个品种,生产量高达上亿片。摩托罗拉单片机被应用到汽车的门控系统、遥控系统和压力检测系统等诸多系统中。这些集中体现了摩托罗拉单片机的高可靠性。但与此形成鲜明对比的是,可以查阅的中文资料却乏善可陈。在需要的时候,往往只能啃英文。考虑到这一点,我们就想到可以翻译一些资料,以便大家在使用时有更好的资料可以参考。

不久前,应清华大学出版社的邀请,与几个志同道合的好友一起参与了本书的翻译。本书给读者提供了理解和使用微控制器(特别是摩托罗拉的 68HC11 和 68HC12)所需要的背景知识,这两个微控制器可以很容易地应用在一个完整的控制系统或嵌入式设备中。本书首先解释了微控制器的应用和基础原理,然后从一个程序员的角度对设备的开发进行了说明,最后说明的是硬件,以及控制应用需要如何与外界进行连接。

本书可以作为学习自动化控制的学生、感兴趣的业余爱好者和有经验的微控制器用户的入门教材和参考书。书中用具体的例子对各个主题进行了详细的说明。而且还提供了许多实际的问题,给出了问题的答案。还有一点是,本书还可以引导读者如何深入理解摩托罗拉的数据手册。附录中还附了数据手册中的一些重要部分,并且用实际的问题引导读者如何从这些部分中发现相关信息。

本书中文版是我们集体智慧的结晶。在 5 个月时间内,我们齐心协力完成了本书的翻译。其中,第 1 章、第 2 章和第 3 章由李健平翻译。第 4 章、第 5 章和第 6 章由李小洪翻译。第 7 章、第 8 章和第 9 章由曾翠翻译。第 10 章、第 11 章和第 12 章由郑国贤翻译。其余部分由曾琴完成。全书由李小洪统一协调和统稿。在此期间,得到了文开棋老师的大力协助和支持,在此表示感谢。要感谢合作伙伴们,是大家的共同努力才使得本书能够如期与读者见面。

由于译者水平有限,书中难免会有不当和错误之处,希望读者能够批评指正。同时,也希望本书能对读者有所帮助。

# 前 言

微控制器在工业中对许多类设备(从消费品到专用设备)进行控制。它们已经替代了老的控制器,包括微处理器,而且,支持脱机的计算机主处理器的需求正在增长,随着越来越多的设备需要更多的智能特征,需求也随之增长,其应用范围从现在的汽车发动机控制到激光打印机控制,以及其他的计算机外围设备。消费品的应用之一是洗衣机控制器,它可以根据负载大小、衣物类型和脏的程度来调整洗涤循环。当前技术已经发展到将洗衣机连接到因特网上。在原理上,它还可以在需要时购买或订购洗涤剂。可以预见,在不久的将来,手表之类的可穿戴计算机就可以通过无线网络和洗衣机进行通信,并控制洗衣机了。

本书旨在说明如何使用一个通用的微控制器:摩托罗拉的 68HC11,并进而扩展到一个更大的范围,68HC12 也在讨论之列。68HC11 和 68HC12 微控制器的使用比较简单,而且还有一个完整的控制系统所必需的大部分特性。因此,自动控制专业的学生就可以把它们作为一个元件用在控制系统中。有兴趣的爱好者也可以通过它们理解并实现计算机和数据通信系统之间的实验。虽然以后会使用更新的处理器,但由于 68HC11 使用简单,因此它是学习嵌入式控制基础知识的一个很好的工具。

第 5 版的目标仍旧和前面几版一样,提供理解和使用微控制器(特别是 68HC11/12)所需要的背景知识。它先以介绍的性质解释了微控制器的起源和应用。然后从编程者的视角说明了一个机器的开发。最后说明应用控制中使用的机器硬件,以及应用是如何与外界连接的。

读者会发现,本书有助于学习如何使用 68HC11/12。原始生产商的数据手册没有提供初学者所需要的背景知识说明。有经验的用户则会发现本书可以和生产商的原始文档相得益彰。

本书可以作为介绍接口连接和工业控制课程的教材。要使用本书,读者需要了解数字逻辑和编码系统。有一些计算机编程知识当然好,但不强求。虽然本书是为大学课程编写的,但也供感兴趣的电脑黑客或爱好者自学参考。其他接受正式教育的人也能找到有用的信息。本书还可以提供利用微控制器训练工具或仿真软件实现的动手练习。练习一般是针对原理的,有助于读者理解其他的微控制器和相关设备。

为了可以灵活地使用本书,本书分成了 5 部分:

第 1 部分:第 1 章介绍了微控制器技术是什么,如何发展以及如何使用。接着介绍了微控制器术语和元件,最后回顾了必需的存储器概念。

第 2 部分:第 2 章到第 4 章讨论了编程的有关概念、发送指令给微控制器的语言和如何使用寄存器和存储器,还讨论了程序如何生成、使用和生成文档。

第 3 部分:第 5 章到第 7 章讨论了芯片操作,主题包括系统总线、操作模式、时钟操作和存储器技术。

第 4 部分:第 8 章到第 12 章说明并行、串行、可编程定时器和模拟接口子系统,并给出了使用子系统的基本软件。同时也介绍了微控制器和传感器以及传动装置连接的一些通用

硬件设计。第8章概述了所有子系统,但没有说明和特定处理器相关的细节。

第5部分:第13章说明了控制方法,揭示了程序是如何在控制应用中使用子系统的。第14章和15章讨论了目前的工业,提供了一个典型应用的调查,其中包括选择微控制器和其他微控制器的特性。

介绍课程通常应该包括第1部分和第2部分,但本书认为还应该包括第8章。提高或更深入的课程应该包括第3部分和第4部分;但这两种课程都可以使用第5部分。

虽然本书包含了一些数据手册,但不完整。完整的数据,请参考附录C中所列出的文献。

## 第5版中的修改

由于越来越多的嵌入式设备可以连接到因特网,第10章就包含了更多的网络通信的信息,其中描述了以太网和因特网接口。第15章去掉了Intel 8051的内核部分,取而代之的是68HC11产品系列中的下一代HCS12系列的特性。

## 如何使用本书

使用本书时,最重要的一点是对计算机和相关技术的热爱。如果读者已经熟悉汇编语言编程,就可以粗略地读一下第1章和第2章,看看有没有新的东西。不需要仔细阅读。如果读者不熟悉数字逻辑和编码系统,就应该阅读附录F。

只要有可能,本书就会使用普通的原理。在学完本书后,读者应该能够使用任何一种微控制器,但要全面了解一个特定的微控制器,就需要深入下去。本书选择的是68HC11,但它有许多变种。在解释特定的细节时,本书通常指的是68HC11E9,但有特别说明的除外。

要理解微控制器,还需要知道许多术语,不知道术语就不能理解概念——不知道概念也不能理解术语,本书采用的对策是逐步对细节进行说明。

附录对本书所用的术语提供了参考和进一步说明。附录A详细列出了本书所用的68HC11指令,这些指令日后会经常用到。附录B是快速参考部分,它可以作为一个交叉参考,以查找其他需要的68HC11细节。如果没有68HC11数据手册或手册,则必须查看附录A和B。

大部分术语在正文中都有说明。文中用黑体表示新术语的首次出现。术语表提供了一个补充的参考。

技术类书籍往往会使用许多缩写。不幸的是,这可能会为许多读者造成困扰。实际上,大多数技术数据手册在使用缩写时通常没有解释。在本书中,在第一次使用每个缩写时都进行了定义,有许多缩写的定义在后文会反复出现,读者还可以在术语表中找到一个缩写列表。

附录C列出了有些主题其他资料的出处,同时还列出了本书提到的一些可以提供商业产品的供应商。作者的网站是查阅资料的另一个好地方:[flaming0.flemingc.on.cd/~pspasov/MCU/mcu.htm](http://flaming0.flemingc.on.cd/~pspasov/MCU/mcu.htm)。

附录D定义了一些惯例。由于同样的事情可以用许多方法实现,因此本书认为需要定义一个标准的方法。惯例已在工业上得到广泛使用。例如,本书用摩托罗拉的美元符号(\$ )惯例表示数字编码的一种方法(这时符号不是表示美元),这样就可以保证符号含义的



一致性。

一些程序例子所用的头文件包含在附录 E 中。注意,附录 E 对一个头文件进行了说明。

附录 F 介绍了数字逻辑和数字转换基础。如果读者不熟悉这些主题,就需要先读这部分内容。

如果需要了解波形的基础知识,可以参考附录 G。

最后,附录 H 介绍了因特网上的资源和配套网站所包含的内容。

**68HC12 兼容性** 读者可能还在使用 68HC12 的书,本书将省略那些 68HC11 没有的特性。第 15 章对 68HC12 独有的特性进行了粗略的说明。还要注意的,两个处理器的机器代码是不同的,但为 68HC11 写的源代码可以用在 68HC12 上。换句话说,可以把 68HC11 的汇编源代码和 C 源代码编译成 68HC11 或 68HC12 的机器代码。第 2 章所介绍的专用机器语言的细节不能应用到 68HC12,但原理是一样的。读者可以从因特网上查找到 68HC12 的开发板和其他工具。

## 补充

教师可以联系培生北京代表处,获得本书所附的教师习题答案(PowerPoint 版)。

## 感谢

本人要感谢加拿大摩托罗拉教育训练协会的 Hugh K. McGugan,他提供了技术文档,审查了一些章节,并经常主动积极地帮助我,没有他的帮助就不会有本书。还要感谢美国密歇根州苏圣玛丽苏必利尔湖州立大学的 Alan D. Niemi,他对原稿给出了有价值的意见和建议。感谢德州圣安东尼奥德州西南州立大学的 V. S. Anandu 和宾夕法尼亚技术大学的 Jeffrey Rankinen 对原稿提出了有益的建议。而且,我还要感谢我的学生和同事们,他们的经验帮助我改进了本书,任何不足的地方都是由于我没有听从他们的建议而造成的。

我很感激日本佳能社团的 Hitoshi Doi 和 Chuck Westfall,他们提供了有价值的 EOS 照相机资料。另外,我还要感谢美国 ImageCraft 的 Richard Man,他提供了 ICC11 C 交叉编译器的资料和演示版。同时,荷兰雷伊斯维克技术学院的 Harry Broeder 提供了 THRSim11 模拟程序的一个特殊版本。我要衷心感谢维吉尼亚州黑堡弗吉尼亚技术学院的 Charles Edward Nunally、加州洛杉矶加利福尼亚州立大学的 Jack Levine、德国 Ober-Ramstadt edv-beratung und schulung 的 Kurt Schneider、伊利诺州芝加哥德锐技术学院的 Patrick J. O'Connor 以及伊利诺州芝加哥伊利诺大学芝加哥校区的 Trung Do,他们提供了有价值的意见、建议和修正意见。我特别要感谢 Kurt Schneider 提供的许多建议。澳大利亚科廷科技大学的 Clive Maynard 和马来西亚 KDU college penang 的 Cheah Soon Hooi 对第 5 版提出了改进和修改意见。我要感谢编辑和制作人员,感谢他们杰出和专业的支持,他们是 Prentice Hall 的 Charles Stewart、Mayda Bosco、Maria Rego 和 Alex Wolf,以及 Carlisle Publishers Services 的 Kelli Jauron。

最后,要特别感谢我的妻子 Renate,她在情感上给我的支持是无价之宝。我还要感谢我 11 岁的女儿 Emillie、8 岁的女儿 Hannah 和 5 岁的女儿 Anika,她们给我带来了很多的欢乐。

Peter Spasov  
于弗莱明大学

# 目 录

## 第 1 部分 介绍微控制器技术

第 1 章 微控制器原理 .....	1	1.2.4 微控制器 .....	5
1.1 微控制器是什么,有什么用处 .....	1	1.2.5 摩托罗拉 MC68HC11 .....	5
1.1.1 概况 .....	2	1.3 自上而下看微控制器系统 .....	7
1.1.2 术语和惯例 .....	3	1.3.1 微控制器系统 .....	7
1.1.3 技术视角 .....	3	1.3.2 微控制器单元(MCU) .....	8
1.1.4 特定视角 .....	4	1.3.3 中央处理单元(CPU) .....	10
1.2 历史 .....	4	1.4 存储器概念 .....	11
1.2.1 早期的计算机 .....	4	1.4.1 结构 .....	11
1.2.2 数字计算机 .....	4	1.4.2 半导体存储器 .....	13
1.2.3 微处理器 .....	5	1.5 微控制器存储器分配图 .....	14

## 第 2 部分 软 件

第 2 章 编程 .....	16	2.5.9 直接寻址和扩展寻址模式 .....	39
2.1 汇编语言和其他编程语言 .....	16	2.5.10 变址寻址模式 .....	44
2.2 源代码、目标代码和汇编编译器 .....	18	2.5.11 存储器堆惯例 .....	51
2.2.1 机器语言 .....	18	2.5.12 寻址模式复习 .....	52
2.2.2 汇编语言 .....	20	2.6 基本操作 .....	53
2.2.3 例子 .....	21	2.6.1 数据处理 .....	53
2.2.4 手工汇编 .....	22	2.6.2 算术 .....	55
2.2.5 仿真器 .....	23	2.6.3 逻辑 .....	56
2.3 使用高级语言 .....	23	2.6.4 其他操作 .....	60
2.3.1 一般概念 .....	23	2.6.5 C 中的基本算术和逻辑操作 .....	60
2.3.2 微控制器的 C 语言 .....	24	2.7 微控制器算术和条件代码寄存器 .....	61
2.4 中央处理单元(CPU)的读取/执行 操作 .....	26	2.7.1 2 的补码和符号位 .....	61
2.5 指令集和寻址模式 .....	29	2.7.2 进位、溢出、零和半进位 .....	63
2.5.1 指令集参考 .....	30	2.7.3 二进制编码的十进制数 (BCD)算术 .....	69
2.5.2 指令的类型 .....	31	2.7.4 乘法 .....	70
2.5.3 寻址模式 .....	31	2.7.5 整数除法 .....	71
2.5.4 前缀字节 .....	31	2.7.6 小数除法 .....	71
2.5.5 固有寻址模式 .....	32	2.7.7 浮点数 .....	72
2.5.6 列表和执行惯例 .....	34	2.8 用循环和分支控制程序流程 .....	74
2.5.7 停止程序的执行 .....	36	2.8.1 流程控制 .....	74
2.5.8 立即寻址模式 .....	36	2.8.2 条件分支 .....	76

2.8.3	相对寻址	81	3.9.1	加电和外部复位	120
2.8.4	二级存储器访问指令	82	3.9.2	其他处理器复位	121
2.8.5	跳转指令	84	3.9.3	非法操作码陷阱	122
2.8.6	浮动程序	85	3.9.4	不可屏蔽中断(XIRQ)	122
2.9	总结	85	3.9.5	中断请求(IRQ)	126
	练习题	85	3.9.6	中断设置	126
<b>第3章</b>	<b>堆栈、子程序、中断和复位</b>	<b>92</b>	3.9.7	中断查询	127
3.1	堆栈介绍	92	3.10	软件和CPU控制中断	127
3.2	使用堆栈保存数据	93	3.10.1	软件中断(SWI)	127
3.3	使用子程序	96	3.10.2	CPU控制和监视	128
3.4	利用子程序进行模块化编程	101	3.11	死神之吻:堆栈溢出	128
3.4.1	自上而下解决问题	101	3.12	总结	129
3.4.2	子程序库	105		练习题	129
3.4.3	文件和使用子程序	105	<b>第4章</b>	<b>交叉汇编和程序开发</b>	<b>133</b>
3.4.4	简单的参数传递	108	4.1	程序开发介绍	133
3.5	子程序操作	109	4.2	源代码的格式	134
3.5.1	子程序对堆栈的使用	109	4.3	代码和数据段	137
3.5.2	嵌套子程序说明	111	4.4	伪操作	138
3.6	中断概念	112	4.5	分两步汇编的过程	144
3.6.1	复位	112	4.6	汇编选项和预处理命令	148
3.6.2	中断	114	4.7	十六进制和二进制文件	150
3.7	中断矢量	116	4.8	文档文件	153
3.8	中断操作	118	4.9	模拟	156
3.8.1	中断屏蔽和使能	118	4.10	仿真板和仿真	156
3.8.2	寄存器入栈	119	4.11	总结	158
3.8.3	中断优先级和中断嵌套	120		练习题	159
3.9	硬件中断和复位	120			

## 第3部分 硬 件

<b>第5章</b>	<b>总线概念和操作模式</b>	<b>161</b>	5.10	系统电路	173
5.1	引言	161	5.11	总结	175
5.2	总线	161		练习题	176
5.3	三态	163	<b>第6章</b>	<b>硬件</b>	<b>177</b>
5.4	地址译码	166	6.1	半导体技术	177
5.4.1	地址译码电路	166	6.1.1	集成电路	177
5.4.2	一个地址译码设计的例子	167	6.1.2	晶体管的原理	179
5.4.3	局部和全部译码系统	169	6.1.3	生产和设计硬件	180
5.5	操作模式	169	6.1.4	HCMOS 和其他的逻辑系列	180
5.6	单芯片操作模式	170	6.1.5	CMOS 和 HCMOS 防护	181
5.7	扩展复用操作模式	171	6.2	芯片指标	183
5.8	特殊启动模式	172	6.2.1	数据手册	183
5.9	特殊测试操作模式	173	6.2.2	最大额定值	184

6.2.3 建议的工作条件 .....	184	6.6.4 修改 EEPROM 的编程向导 ...	192
6.2.4 直流(dc)电特性 .....	184	6.7 配置控制寄存器(CONFIG) .....	196
6.2.5 交流(ac)电特性 .....	186	6.8 RAM 和 I/O 映射寄存器(INIT) .....	196
6.3 存储器技术 .....	187	6.9 总结 .....	197
6.3.1 回顾和概览 .....	187	练习题 .....	197
6.3.2 只读存储器(ROM) .....	187	<b>第7章 时钟操作</b> .....	199
6.3.3 随机访问存储器(RAM) .....	188	7.1 时序图 .....	199
6.4 可擦除可编程只读存储器(EPROM)		7.2 系统时钟 .....	200
.....	188	7.3 总线操作 .....	203
6.5 闪存 .....	191	7.4 周期操作 .....	207
6.6 电可擦除可编程只读擦除器		7.5 示波器测试循环 .....	209
(EEPROM) .....	191	7.6 逻辑分析仪 .....	211
6.6.1 起源和应用 .....	191	7.7 传输线的影响 .....	213
6.6.2 寄存器块 .....	191	7.8 总结 .....	215
6.6.3 EEPROM 操作和指标 .....	192	练习题 .....	216

## 第 4 部分 接口连接

<b>第8章 接口连接的概念</b> .....	218	<b>第9章 并行输入/输出</b> .....	240
8.1 介绍 .....	218	9.1 子系统的介绍 .....	240
8.2 输入/输出子系统和寄存器 .....	219	9.2 7段(LED)显示输出 .....	243
8.2.1 不同的子系统 .....	219	9.2.1 7段显示 .....	243
8.2.2 I/O 寄存器 .....	220	9.2.2 7段代码 .....	244
8.2.3 I/O 端口 .....	220	9.2.3 驱动7段显示器 .....	245
8.2.4 端口替换单元 .....	221	9.2.4 软件查找7段代码 .....	247
8.3 存储器或输入/输出映射 .....	222	9.2.5 硬件解码器 .....	247
8.4 用轮询或中断接口 .....	225	9.2.6 多路复用显示器 .....	247
8.4.1 轮询 I/O .....	225	9.3 液晶显示器(LCD) .....	249
8.4.2 中断 I/O .....	226	9.3.1 操作 .....	249
8.5 并行 I/O 子系统 .....	226	9.3.2 7段 LCD .....	250
8.6 串行系统 .....	227	9.3.3 点阵 LCD .....	250
8.6.1 同步串行 I/O 系统 .....	228	9.3.4 LCD 激励波形 .....	254
8.6.2 异步串行 I/O 子系统 .....	229	9.4 键盘接口 .....	255
8.7 可编程计时器 I/O 子系统 .....	230	9.4.1 矩阵键盘 .....	255
8.8 模拟数字 I/O 子系统 .....	231	9.4.2 7选2 键盘解码 .....	256
8.9 I/O 子系统寄存器 .....	233	9.4.3 去抖动 .....	259
8.9.1 控制寄存器和系统配置 .....	234	9.4.4 矩阵键盘的软件驱动程序 ...	260
8.9.2 状态寄存器 .....	235	9.4.5 使用硬件解码芯片 .....	265
8.9.3 数据寄存器 .....	236	9.5 其他使用输入/输出的设备 .....	266
8.10 接口标准 .....	237	9.6 选通输入/输出 .....	266
8.11 总结 .....	238	9.6.1 输入选通 .....	267
练习题 .....	239	9.6.2 输出选通 .....	268
		9.6.3 配置 .....	269

9.7 完整的握手输入/输出 .....	271	10.6.5 SPI 数据传输操作总结 .....	327
9.7.1 输入握手 .....	271	10.7 SPI 软件 .....	328
9.7.2 输出握手 .....	274	10.7.1 配置和寄存器 .....	328
9.8 并行接口标准 .....	278	10.7.2 主机 SPI 操作 .....	331
9.8.1 Centronics 并行接口 .....	278	10.7.3 从机 SPI 操作 .....	332
9.8.2 IEEE-488 通用仪器总线 .....	284	10.7.4 采样驱动程序 .....	333
9.8.3 小型计算机系统接口(SCSI) .....	287	10.7.5 把字长度为奇数的数据 传输到外围设备 .....	335
9.9 总结 .....	289	10.8 SPI 错误处理 .....	336
9.9.1 并行 I/O 子系统总结 .....	289	10.9 网络通信 .....	336
9.9.2 并行 I/O 应用总结 .....	291	10.9.1 物理层和数据链路层 .....	337
练习题 .....	292	10.9.2 互连网络和以太网 .....	338
<b>第 10 章 串行子系统</b> .....	295	10.9.3 68HC11 以太网接口 .....	341
10.1 异步通信系统 .....	295	10.10 总结 .....	343
10.1.1 串行通信的第一步 .....	295	10.10.1 异步通信 .....	343
10.1.2 协议和流控制 .....	301	10.10.2 串行通信接口(SCI) .....	343
10.1.3 通信通道操作 .....	301	10.10.3 同步串行系统 .....	344
10.1.4 modem .....	302	10.10.4 串行外围接口(SPI) .....	344
10.2 串行通信接口(SCI) .....	304	练习题 .....	344
10.2.1 68HC11 中的 SCI 子系统 .....	304	<b>第 11 章 可编程定时器操作</b> .....	348
10.2.2 发送操作 .....	305	11.1 概述 .....	348
10.2.3 接收操作 .....	306	11.2 输出比较器 .....	352
10.2.4 SCI 软件 .....	306	11.2.1 操作顺序 .....	353
10.3 SCI 寄存器 .....	312	11.2.2 单个脉冲的例子 .....	355
10.3.1 串行通信数据寄存器 (SCDR) .....	312	11.2.3 方波和脉宽调制(PWM) 输出 .....	358
10.3.2 波特率寄存器(BAUD) .....	313	11.2.4 步进马达输出 .....	363
10.3.3 串行通信控制寄存器 1 (SCCR1) .....	313	11.3 输入捕捉 .....	367
10.3.4 串行通信控制寄存器 2 (SCCR2) .....	314	11.3.1 操作顺序 .....	367
10.3.5 串行通信状态寄存器 (SCSR) .....	317	11.3.2 脉冲宽度的例子 .....	368
10.3.6 接收错误 .....	318	11.3.3 周期(频率、速度)测量 .....	370
10.4 同步串行输入/输出 .....	321	11.3.4 长周期测量 .....	374
10.5 串行外围接口(SPI) .....	322	11.4 脉冲累加器 .....	376
10.6 SPI 拓扑结构和应用 .....	323	11.4.1 功能描述 .....	376
10.6.1 总线拓扑 .....	323	11.4.2 事件计数 .....	377
10.6.2 使用总线拓扑的微控制器 I/O .....	324	11.4.3 门控时间累加器 .....	380
10.6.3 级联拓扑 .....	325	11.5 实时中断 .....	383
10.6.4 使用级联拓扑的微控制器 I/O .....	326	11.6 总结 .....	386
		练习题 .....	387
		<b>第 12 章 模拟转换器子系统</b> .....	389
		12.1 概念和术语 .....	389
		12.2 变换器 .....	392
		12.2.1 传感器 .....	392

12.2.2 信号调节 .....	396	12.5 内嵌 A/D 子系统 .....	407
12.2.3 激励器 .....	397	12.6 总结 .....	412
12.3 数模转换器 .....	399	练习题 .....	413
12.4 模数转换器 .....	404		

## 第 5 部分 微控制器世界

<b>第 13 章 应用控制软件</b> .....	415	练习题 .....	476
13.1 布尔逻辑控制 .....	415	<b>第 14 章 应用</b> .....	479
13.2 序列发生器 .....	417	14.1 汽车发动机控制 .....	479
13.3 序列机 .....	420	14.2 消费类电子:自动照相机 .....	485
13.4 模拟反馈控制 .....	426	14.3 计算机外围设备:点阵打印机 .....	491
13.5 数字信号处理器 .....	431	14.4 社会和微控制器技术 .....	499
13.6 使用 C 语言 .....	436	<b>第 15 章 8 位微控制器系列</b> .....	501
13.6.1 为什么使用 C .....	437	15.1 微控制器和半导体行业 .....	501
13.6.2 控制例子 .....	438	15.2 摩托罗拉系列 .....	503
13.6.3 使用 C 语言的 PID 控制器 .....	442	15.2.1 68HC11 系列 .....	503
13.6.4 C 函数库 .....	444	15.2.2 68HC05 系列 .....	504
13.6.5 用户自定义函数 .....	445	15.2.3 68HC16 系列 .....	506
13.6.6 幕后:使用堆栈 .....	447	15.2.4 68HC08 系列 .....	508
13.6.7 指针 .....	450	15.2.5 68HC12 系列 .....	509
13.6.8 中断编程 .....	451	15.3 摩托罗拉的 HCS12 系列 .....	511
13.6.9 其他变量类型和特性 .....	455	15.4 其他系列的说明 .....	512
13.7 多任务 .....	455	15.5 选择微控制器 .....	513
13.8 模糊逻辑 .....	461	15.6 总结 .....	516
13.8.1 为什么要用它,它是什么 .....	461	<b>附录 A 指令集摘要</b> .....	517
13.8.2 概述 .....	462	<b>附录 B 快速参考</b> .....	530
13.8.3 建立一个知识库 .....	463	<b>附录 C 更多的信息</b> .....	538
13.8.4 模糊逻辑推理 .....	469	<b>附录 D 惯例</b> .....	541
13.8.5 开发以及其他 .....	472	<b>附录 E 头和库文件源程序清单</b> .....	546
13.8.6 MC68HC12 支持的模糊逻辑 .....	473	<b>附录 F 数字逻辑和二进制代码</b> .....	549
13.9 总结 .....	475	<b>附录 G 基础波形</b> .....	561
		<b>附录 H 因特网和网站资源</b> .....	563
		术语表 .....	569
		缩写词 .....	575

# 第 1 部分 介绍微控制器技术

## 第 1 章 微控制器原理

### 目标

在学完本章后,应达到以下目标:

- 能大致说明微控制器是什么,如何使用
- 学会使用微控制器开发中应用的一些术语和惯例
- 能够画出包含中央处理单元(CPU)、存储器、寄存器和总线的框图
- 能够识别半导体存储器的基本类型及其一般应用
- 能确定地址和地址的范围
- 能解释存储器分配图

### 1.1 微控制器是什么,有什么用处

**瑞士军刀** “做出选择”,有一个声音说。你会选择什么呢?在旅途中你只能带一种工具,而旅途中要经过森林、平原、山地和沙漠,你需要的工具应该是小巧、轻便和有用的,于是你选择了瑞士军刀。“声音”赞许地点了下头。这是一个好的选择,根据型号的不同,瑞士军刀可能配备了各种刀片、锯条、剪刀、开罐器、螺丝起子(最重要)、镊子、牙签和其他的东西。

“微控制器是什么?”“声音”问到。你回答:“是一个(计算机化的)瑞士军刀。”“声音”满意了。确实,微控制器很像瑞士军刀,也是一个单个的设备,这里指的是一个芯片。微控制器通过指令,可以读出信息、保存信息、通信、计时和开关所连接设备,而且根据型号的不同,还可以做其他的事情。但是,尽管概念强大,用途很多,但你可能还是没有听说过很多有关微控制器的知识。大多数和计算机有关的著作都描述了计算机所用的处理器芯片。但大部分单片可编程设备都是微控制器。

如果你是一个细心的人,就会发现到处都有微控制器。最常见的就是在 1985 年以后生产的几乎所有的汽车中。消费类产品有电视机、CD 机、洗衣机、电话和微波炉。办公用计算机除了使用主处理器外,还使用微控制器来控制外围设备,如键盘和打印机。在自动化制造系统中,如机器人和流水线等生产设备上都使用微控制器。

本书将从 3 个方面来说明微控制器:概况、技术和特殊性,先说明总的方面,然后从技术视角和特殊性方面进行说明。

### 1.1.1 概况

可以把微控制器看成是一个可编程的单片集成电路(integrated circuit, IC),用来控制一个系统的操作的,被控制的系统通常是一个机器。最典型的应用就是汽车的控制,包括发动机喷油控制、传输控制、悬吊和驾驶控制、显示板和制动系统。实际上,某些使用68HC11微控制器控制发动机功能的汽车曾经连续5年赢得了Indianapolis 500比赛的胜利。

图1.1是汽车发动机中的空气/燃料混合的控制部分,空气和燃料的数量以及点火的时机决定了燃烧的效率和排出废气的数量。

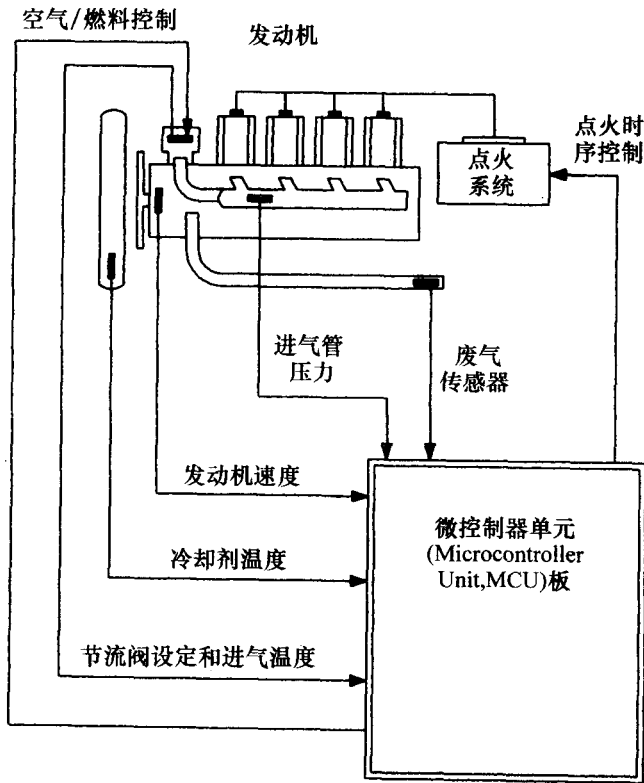


图 1.1 在汽车发动机中用微控制器控制空气/燃料的混合

微控制器单元利用传感器的输入来控制点火时机,以获得最大的燃烧效率,排出最少的废气,这和输入变化的快慢也有关系,废气传感器检测氧气的浓度,这样系统就可以对出现的问题进行纠正。通过编程,微控制器可以控制发动机什么时候制冷、什么时候加热、什么时候加速和什么时候巡航。同时,如果需要,驾驶员可以指示机器对速度和加速性能进行优化。如果有其他的输入,微控制器单元可以保持常速,例如巡航控制。

本书想要通过本例子的一些细节说明在控制系统中用一个微控制器可以做什么。有关空气/燃料比率控制的更详细的资料,可以参考附录C的参考1。

自动照相机也使用了微控制器,用来控制曝光和聚焦。微控制器的另一个典型应用是计算机鼠标,微控制器被用来读取鼠标球的运动、检测按下的按键以及处理与计算机的



通信。

### 1.1.2 术语和惯例

本书中将使用计算机和微控制器工业中所使用的特定的词、缩写和首字母缩写,虽然本书尽量不使用技术术语,但有时候仍不可避免。当本书首次使用一个术语时,将用黑体表示。在术语表中,你可以找到术语的定义。在你第一次遇到某个术语的时候,有必要去理解它。

在下面的对技术的说明中(1.1.3节),本书使用了许多技术术语。由于计算机和微控制器领域瞬息万变,因此不是所有的人都以同样的方式使用所有的术语,但之间的差异是很小的,通常很容易理解。

本书不能预知每位读者的背景,因此,为了方便起见,在附录F中包含了对数字逻辑的介绍。你可以根据自己的背景来确定是否需要和什么时候阅读附录F。如果对这些知识很熟悉,就可以跳过。如果以前没有接触过,就应该多花点时间看一下。如果遇到不熟悉的术语,也可能需要查阅附录F,如二进制、位、字节、字和十六进制等。

微控制器用二进制数的形式使用和保存信息,本书将使用二进制、十六进制和十进制数,并在需要的时候对它们的变体进行说明。现在对ASCII码<sup>①</sup>进行说明。ASCII码是一种二进制代码,每个数字代表一个可打印字符,如“A”、“a”、“#”和特殊控制功能,如换行(LF)和退格(BS)。“快速参考”(附录B)包含了一个ASCII码表,其中显示了每个(标准)字符的十六进制代码。

---

### 练习

找出ACK,FS,L和j的ASCII码。

### 答案

\$06, \$1C, \$4C, \$6A

注意,美元符号(\$)代表后面的数字是十六进制。欲进一步了解十六进制,请参考附录F。

---

微控制器包含了很多位操作。当本书提到置位一个位的时候,就表示要设置这个位为逻辑高(1)。清除或复位一个位表示设置这个位为逻辑低(0)。其他的惯例将在需要的时候做介绍。附录D列出了本书所使用的一些惯例。

### 1.1.3 技术视角

微控制器是一个单芯片器件,其中包含了存储程序和数据信息的存储器,以及用来控制数据输入、处理和输出的可编程逻辑电路。换句话说,除了中央处理单元(CPU),微控制器

---

<sup>①</sup> ASCII的全称为American Standard Code for Information Interchange,即美国信息交换标准码。