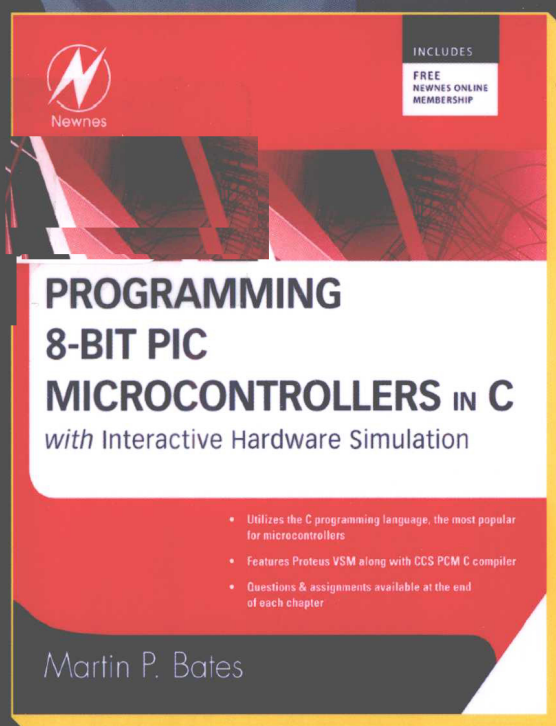


8位单片机C语言编程 基于PIC16

Programming 8-bit PIC Microcontrollers in C
with Interactive Hardware Simulation

[英] Martin P. Bates 著
李中华 张雨浓 译



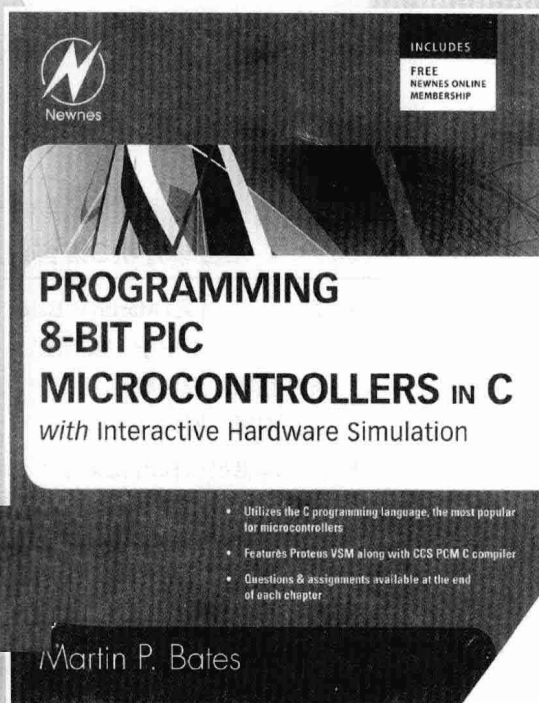


图灵电子与电气工程丛书

8位单片机C语言编程 基于PIC16

Programming 8-bit PIC Microcontrollers in C
with Interactive Hardware Simulation

[英] Martin P. Bates 著
李中华 张雨浓 译



人民邮电出版社
北京

图书在版编目 (C I P) 数据

8位单片机C语言编程 : 基于PIC16 / (英) 贝茨
(Bates, M. P.) 著 ; 李中华, 张雨浓译. -- 北京 : 人民
邮电出版社, 2010. 8

(图灵电子与电气工程丛书)

书名原文: Programming 8-bit PIC
Microcontrollers in C:with Interactive Hardware
Simulation

ISBN 978-7-115-23166-6

I. ①8… II. ①贝… ②李… ③张… III. ①单片微
型计算机—C语言—程序设计 IV. ①TP368.1②TP312

中国版本图书馆CIP数据核字(2010)第118834号

内 容 提 要

本书是用C语言对PIC16微控制器进行编程的实用指导。书中结合实例简明扼要地描述了如何基于PIC进行C语言编程,并着重介绍了Microchip MPLAB IDE、CCS PCM C编译器、Proteus VSM等常用软件。

本书适合刚从事微处理器应用设计的技术人员、大学生以及无PIC微控制器编程经验的爱好者使用。

图灵电子与电气工程丛书

8位单片机C语言编程: 基于PIC16

- ◆ 著 [英] Martin P. Bates
译 李中华 张雨浓
责任编辑 朱 巍
执行编辑 罗词亮
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市海波印务有限公司印刷
- ◆ 开本: 787×1092 1/16
印张: 11.25
字数: 303千字 2010年8月第1版
印数: 1-3 000册 2010年8月河北第1次印刷

著作权合同登记号 图字: 01-2009-6907号

ISBN 978-7-115-23166-6

定价: 39.00元

读者服务热线: (010)51095186 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Programming 8-bit PIC Microcontrollers in C with Interactive Hardware Simulation by Martin P. Bates, ISBN: 978-0-7506-8960-1.

Copyright © 2008 by Elsevier Inc. All rights reserved.

Authorized Simplified Chinese translation edition published by the Proprietor.

ISBN: 978-981-272-326-0.

Copyright © 2010 by Elsevier (Singapore) Pte Ltd. All rights reserved.

Elsevier (Singapore) Pte Ltd.

3 Killiney Road

#08-01 Winsland House I

Singapore 239519

Tel: (65)6349-0200

Fax: (65)6733-1817

First Published 2010

2010年初版

Printed in China by POSTS & TELECOM PRESS under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in China only, excluding Hong Kong SAR and Taiwan. Unauthorized export of this edition is a violation of the Copyright Act. Violation of this Law is subject to Civil and Criminal Penalties.

本书简体中文版由Elsevier (Singapore) Pte Ltd.授权人民邮电出版社在中华人民共和国境内（不包括香港特别行政区和台湾地区）出版与销售。未经许可之出口，视为违反著作权法，将受法律之制裁。

序

现代生活中，嵌入式微控制器无处不在。在普通家庭就到处可见嵌入式微控制器的应用，比如手机、计算器和MP3播放器。这些嵌入式产品都无一例外地需要至少一个控制器，有些可能需要几个控制器：一个用于控制用户界面（按键和显示），另一个用于控制电动机，甚至可能控制整个系统管理。同样，嵌入式微控制器也可以应用于洗衣机、车库电动门、卷发器或者电动牙刷。如果嵌入式产品使用可充电电池供电，那么现代高密度化学电池就需要智能充电器。

10年前，学习如何使用微控制器对于人们来说还是相当困难的。当时最便宜的编程器也要100美元左右，产品开发不仅需要可擦除的有窗存储部件（其成本是一次性可编程存储部件的10倍左右），还需要一个UV擦除器，用于擦除有窗存储部件。调试工具仅仅在专业领域才使用。现在，大部分微控制器已使用电可擦除的内存作为程序存储器了。这意味着器件可以在电路中被重新编程，不需要UV擦除器，也不需要特别的用于开发的程序包。对于初学者来说，只需花大约25美元购买PICkit 2 Starter Kit开发包，就能对Microchip公司的很多微控制器（MCU）进行编程和调试了。Microchip公司一般都会提供免费的集成开发环境（IDE），包括汇编器和仿真器。如此低廉的学习使用嵌入式微控制器的成本前所未有。

尽管MPLAB包含免费的汇编器，但是编写汇编代码十分繁琐，而且维护起来也很困难。使用C语言进行开发则将程序员从多字节数学和分页控制等细节中解放出来，而且能提高代码的可读性和可维护性。CCS和Hi-Tech编译器都提供免费的学生版本。如果将开发过程所节省的时间折算进来，完整版的编译器也是相当便宜的。

虽然使用C语言就不必再学习PIC16汇编语言了，并且用户也不必再操心所有的细节了，但还是有必要理解这种体系结构。时钟的选择、外围设备的设置以及引脚的复用问题都是需要了解的。本书将引导读者从了解控制器开始，最终学会如何完成一个应用。本书将采用十分有特色的PIC16F877A单片机来练习，介绍它的体系结构和器件配置。这是一款很好的入门产品，它和其他的PIC16单片机的体系结构很相似，只是在I/O线路、内存、外围设备方面有所不同。使用PIC16F877A单片机开发的产品可以很容易移植到更小、更便宜的中等系列的PIC微控制器芯片上。本书还会介绍单片机的外围设备，并且会介绍如何设置这些外围设备，让其工作，从而简化固件。

本书还介绍了Microchip的集成开发环境，即MPLAB软件。MPLAB软件包含一个编辑器和一个仿真器，并提供了很多编译器的接口，包括本书使用的CCS编译器。最后，本书还介绍了Proteus仿真器，它能帮助读者完成系统的仿真。制作PCB原型时，使用该仿真器能节省很多时间和金钱。

Dan Butler

Microchip公司首席应用工程师

前 言

本书是系列丛书的最后一本，这一丛书介绍了如何使用美国Microchip公司的PIC系列微控制器芯片进行嵌入式应用设计。该系列的PIC微控制器芯片是用于教学和培训的最流行的微控制器系列，同时，它也正在逐渐应用于工业和商业领域。*Interfacing PIC Microcontrollers*和*Programming PIC Microcontrollers*这两本书展示了使用主流的设计和仿真软件，即电器实验中心（Labcenter Electronics）的Proteus VSM软件开发的应用样例，这款软件是专门用于基于微控制器的电路设计的。应用的演示文件可以从作者的技术支持网站（后面会有详细介绍）下载，然后在屏幕上运行，这样读者就可以详细地学习每个程序的操作。

本书的目标是：

- 用简单的步骤介绍微控制器的C语言编程；
- 演示使用Microchip MPLAB IDE软件创建C语言项目；
- 指导初学者使用PIC的16系列单片机的CCS PCM C编译器；
- 解释如何使用Proteus VSM软件在仿真硬件时测试C语言应用；
- 描述Microchip PICDEM机械电子电路板的应用；
- 概述嵌入式系统设计和项目开发的原则。

随着微控制器芯片内存容量的不断增加，C语言逐渐成为人们开发嵌入式系统的首选语言。Microchip公司提供了专为C语言编程而设计的18和24系列芯片。然而，C语言编程有时也可以使用比较简单的16系列PIC芯片，但前提是应用程序足够简单，且不出16系列芯片限定的内存容量。

本书使用的参考器件是PIC16F877A微处理器，因为这款微处理器包含了全部的外围设备，并且带有适当的内存容量。上一本有关接口技术的书也用它来讲，所以本系列书若都用于培训，会有良好的延续性。

微控制器传统上使用汇编语言编程，每一种类型的处理器都有自己的语法规则，这些规则能直接翻译成机器代码。然而，有一些学生、教师或者业余编程爱好者或许希望跳过学习琐细的汇编编码，而直接学习相对简单、功能更强大的C语言。因此，一本不需要太多汇编知识基础就能读懂的、尽可能通俗易懂地介绍C语言编程的书籍就显得十分及时了。虽然市面上已有几本介绍微控制器C语言编程的书，但是这些书对初学者来说太深、太难理解，而且还介绍了太多的初学阶段还用不上的细节，分散了初学者的注意力。

本书正文用尽可能简单的程序介绍了嵌入式编程技巧，也有通过屏幕完全交互式的电路仿真来演示这一系列的基本技巧，这些编程技巧可以很容易地应用到读者自己的项目中。本书的重点是编写关于每一主题的简单程序，正文还附有硬件模块示意图来说明系统的运作、完整的电路原理图、仿真结果图以及源代码清单，同时读者也可以从网上下载全部例子。大学学生或者设计工程师可以使用这些技巧高标准地编辑他们的项目文件。本书的每一章结束时，都会有一些自我检测和专门为学习而设计的配套作业。

本书的另一个特点就是使用Proteus VSM（virtual system modeling，虚拟系统建模）软件。

原理图捕捉组件 (ISIS) 允许使用扩展有源元件库创建电路图。程序最终要固化在微控制器中, 而形象的原理图则使得应用程序在下载硬件前, 被全面调试。这不仅为专业工程师节省了开发时间, 而且为学生和业余编程爱好者提供了一款优秀的学习工具。

链接、资源和致谢

Microchip 公司

Microchip公司 (www.microchip.com) 是PIC微控制器及相关产品的制造商。我十分感谢Microchip公司在我写这本书的过程中所提供的支持和帮助, 也很感谢Microchip公司允许我使用它们的商标和知识产权。在这里要特别感谢Microchip英国公司的约翰·罗伯兹先生, 感谢他的建议和帮助。Microchip公司的网站上有所有微控制器芯片的硬件、软件以及开发系统的详细信息和说明。为了使用本书所提到的工具进行应用程序开发, 读者必须下载和安装MPLAB IDE。也可以从网上下载PIC16F877A微控制器的数据手册作为参考手册。

PIC、PICmicro、MPLAB、MPASM、PICkit、dsPIC和PICDEM都是Microchip公司的商标。

电气实验中心公司

电气实验中心公司 (www.labcenter.co.uk) 是Proteus VSM软件 (virtual system modeling) 的开发商。Proteus VSM是嵌入式应用开发领域最强大的联合仿真系统。我很感谢电气实验中心的团队在我写这套丛书的过程中所提供的帮助, 尤其是John Jameson先生。这款仿真软件的学生版/评估版可以从www.proteuslite.com上下载到。ISIS Lite、ProSPICE Lite和16F877A仿真器模型可以在www.proteuslite.com/register/ipmbundle.htm找到。

Proteus VSM、ISIS和ARES是电气实验中心公司的商标。

Custom 计算机服务公司

Custom 计算机服务公司 (www.ccsinfo.com) 主要从事PIC微控制器的编译器开发。公司的主要产品包括应用于12位PIC芯片的PCB编译器、用于16位芯片的PCM和用于18系列芯片的PCH。我十分感谢Custom计算机服务公司 (CCS) 的James Merriman所提供的支持。CCS公司的编译器的手册可以从其网站上下载 (本书使用的是第4版)。现在还可以下载到16F877A芯片编译器的30天试用版。

作者的网站

本书有一个专门的网站www.picmicros.org.uk提供技术支持。本书的所有应用例子都可以从网站上免费下载, 并且可以使用评估版的Proteus VSM进行测试。假如不购买相应的VSM许可证, 那么项目的设计文件将会被锁定, 硬件配置就无法改变。类似地, 如果没有相应的编译器许可证 (这个许可证可以从CCS公司的网站获取), 那么固化的程序将不能被修改或重新编译。通过作者网站www.larrytech.com上的链接可以找到生产商提供的其他资料。同时要特别感谢Larrytech因特网服务公司的Gabe Hudson进行网站维护和技术支持。

有关PIC系列丛书的任何疑问和意见, 读者都可以通过电子邮件联系我, 邮箱是 martin@picmicros.org.uk。

最后, 感谢我的妻子Julia, 她包揽了繁琐的家务, 这样我才能完成本书。

绪 言

本书分为5章。第1章总体介绍了PIC微控制器的内部结构，具体描述了16F877A的特性。因为该芯片拥有所有的外围接口，因此它被作为16系列微控制器的代表。所有的16系列芯片都有一个共同的程序执行核心，而它们的主要区别在于程序和数据存储单元的大小。在编辑的时候，某些可操作特性是可以被配置的，如时钟电路的类型、看门狗定时器的开启、复位机制等。内部特性包括文件寄存器系统，其中包括控制寄存器、RAM模块和非易失性的EEPROM模块。并行端口为MCU提供了默认的I/O，但是大多数的引脚都有不止一个功能。8个模拟输入端口和串行接口（UART、SPI和I²C）被引出至特定的引脚。这些硬件特性要点都会在第1章描述，这样后面要介绍的I/O编程就会更容易理解。同时在初始阶段还会（并且只会）使用MPLAB IDE来描述应用开发过程。一个C语言程序的例子将被编辑、编译、下载和测试来演示基本的开发过程和分析所生成的文件集。对MPLAB的调试特性也会进行概要性的描述：如运行、单步运行、断点、监视窗口等。目标代码的反编译允许对C语言源程序的中间汇编语言版本进行分析。

第2章用一些简单的程序介绍了C语言编程。这一章一开始就介绍输入和输出，因为这是嵌入式系统的主要特性。接着，用一个完整的示例程序简单介绍变量、条件模块（IF）、循环模块（WHILE、FOR）。变量和顺序控制的介绍更加具体，同时也会引入函数的概念。因此，这会引申出用于操作定时器和端口的库函数。在简单的计数程序中还会介绍小键盘及可显示文字和数字的LCD。然后会介绍更多的数据类型（长整型、浮点型、数组等）以及汇编语言指令和使用头文件的目的。最后，这一章还会描述汇编程序是如何插入到C程序当中的。

第3章着重介绍使用CCS C库函数对输入和输出操作的编程。这里使用少量的函数就能提供所有的初始化和必需的操作顺序，简化了编程过程。我们给出了模拟输入、使用中断和定时器的示例程序，还用示例应用展示了串行端口的功能。然后对各种类型的串行总线的优点进行比较，并提供一些例子演示用于数据存储的外部串行EEPROM和数模转换器输出之间的连接。这些应用程序可以在VSM中进行测试，但这不是必需的：因为读者可以自由选择是否使用VSM。

第4章着重讨论Microchip公司的PICDEM机械电子电路板。大多数的演示应用都会选择此款芯片，因为它相对便宜，同时拥有一系列用以检测常用的机械电子电路属性的特性，如输入传感器（温度、光、位置）和输出致动器（DC和步进电机）。在对这些性能分别测试后，将对温度控制器的要求进行描述。这一章将详细介绍3.5位7段LCD的运用（这些内容不会在本书其他部分提及）。为辅助进一步的应用设计和实现，这一章还提供了一个电路板的仿真版。

第5章描述了软件和硬件的设计原则，并提供了更多的例子。一个简单的温度控制器为基于机械电子电路板的设计提供了另外一种选择，而数据记录器的设计是基于另外一个标准的硬件系统，可以适用于一系列应用的基板（BASE板）。此外，还提供一个完全的仿真版用于测试和进一步的开发工作。然后，描述操作系统，并对轮询循环操作、中断驱动的系统 and 实时操作系统3种程序设计方案进行比较。接下来是针对给出的应用要求和通用的设计要求确定选择MCU的最终标准。

3个附录（附录A、附录B和附录C）分别介绍了使用ISIS原理图捕捉进行硬件设计、使用CCS

C进行软件设计和使用Proteus VSM进行系统测试。将这些主题从本书的主体中分离出来，主要是因为它们更多的是针对具体的产品。总体来说，MPLAB、CSS C和Proteus VSM组成了一个完整的学习/设计工具包，但要有效地应用这些工具仍需要对产品说明书进行仔细的研究。特别地，VSM带有设计得很好的、比较全面的帮助文件，因此本书不需要重复介绍那些内容。另外，就像别的优秀设计工具一样，VSM发展得非常快，所以再详细的指导都可能很快过时。

附录D对不同的编译器进行了比较，并指明了它们适用的应用场合。附录E总结了CSS C的语法要求。附录F列出了编译器提供的全部CCS C库函数，为便于读者参考，将它们按功能分组。这些附录都是为了给读者在使用CSS C编程时提供一个方便的参考，而辅助完整的CSS C编译器参考手册使用的。

本书的每一章都尽量自成一体，读者可根据自己的兴趣和先前掌握的知识进行取舍，某些部分跳过不看，某些部分仔细地研究。但全书总体上比较连贯地描述了嵌入式系统应用的C语言编程，以保证读者可以打下坚实的基础。

目 录

第 1 章 PIC 微控制器系统	1	2.2 PIC16 C程序基础	25
1.1 PIC 16 微控制器	1	2.2.1 变量	25
1.1.1 MCU特性	2	2.2.2 循环	26
1.1.2 程序执行	3	2.2.3 决策	27
1.1.3 RAM文件寄存器	4	2.2.4 循环控制	28
1.1.4 其他PIC芯片	5	2.2.5 FOR循环	28
1.2 PIC MCU配置	5	2.2.6 SIREN程序	29
1.2.1 时钟选择	5	2.2.7 空程序	30
1.2.2 配置选项	6	2.3 PIC16 数据操作	31
1.2.3 在C语言中的配置	7	2.3.1 变量类型	31
1.3 PIC16 MCU 外围设备	7	2.3.2 赋值运算	34
1.3.1 定时器	8	2.3.3 条件运算	35
1.3.2 A/D转换器	9	2.4 PIC16 C顺序控制	35
1.3.3 比较器	10	2.4.1 while循环	36
1.3.4 并行从端口	10	2.4.2 break、continue和goto	37
1.3.5 中断	11	2.4.3 if..else和switch..case	38
1.4 PIC16 串行接口	12	2.5 PIC16 C函数和结构	39
1.4.1 USART	12	2.5.1 基本函数	40
1.4.2 SPI 总线	13	2.5.2 全局变量和局部变量	41
1.4.3 I ² C总线	14	2.6 PIC16 C输入和输出	42
1.5 PIC16 MPLAB 项目	15	2.6.1 串行LCD	42
1.5.1 MPLAB C 项目	15	2.6.2 键盘和计算器	45
1.5.2 项目文件	16	2.7 PIC16 C更多的数据类型	48
1.6 PIC16 编程与调试	17	2.7.1 数组	48
1.6.1 编程	18	2.7.2 间接寻址操作符	49
1.6.2 调试	18	2.7.3 枚举类型	50
1.6.3 设计包	20	2.8 PIC16 C编译器伪指令	51
练习	20	2.8.1 程序伪指令	51
作业	21	2.8.2 头文件	51
第 2 章 C 编程精华	22	2.9 PIC16 C汇编子程序	56
2.1 PIC16 C入门	22	2.9.1 程序编译	57
2.1.1 简单程序	22	2.9.2 汇编程序块	57
2.1.2 程序创建	22	2.9.3 PIC汇编语言	60
2.1.3 程序测试	23	练习	61
2.1.4 程序分析	24	作业	63

第3章 C 外围接口编程	64	4.6 PICDEM 温度控制器	107
3.1 PIC16 C模拟输入	64	4.6.1 规格	108
3.1.1 模拟信号设置	64	4.6.2 I/O分配	108
3.1.2 电压测量	65	4.6.3 实现	108
3.2 PIC16 C中断	67	4.7 PICDEM板仿真	110
3.2.1 C中断	67	4.7.1 电路描述	112
3.2.2 中断举例	68	4.7.2 演示应用	113
3.2.3 中断语句	69	练习	113
3.3 PIC16 C硬件定时器	70	作业	114
3.3.1 计数器/定时器操作	70	第5章 PIC16 C 应用与系统	116
3.3.2 PWM模式	70	5.1 PIC16 C应用设计	116
3.3.3 比较模式	71	5.1.1 硬件设计	116
3.3.4 捕捉模式	72	5.1.2 软件设计	116
3.4 PIC16 C URAT串行链路	73	5.1.3 应用调试和测试	117
3.5 PIC16 C SPI 串行总线	74	5.2 PIC16 C温度控制器	118
3.6 PIC16 C I ² C串行总线	77	5.2.1 系统操作	118
3.7 PIC16 C并行与串行接口	78	5.2.2 软件设计和实现	119
3.7.1 并行从端口	78	5.3 PIC16 C数据记录系统	120
3.7.2 通信链路比较	80	5.3.1 BASE板	120
3.8 PIC16 C EEPROM接口	81	5.3.2 程序纲要	122
3.9 PIC16 C模拟输出	82	5.4 PIC16 C操作系统	123
练习	85	5.4.1 轮询I/O口	123
作业	86	5.4.2 中断	123
第4章 C语言在机械电子电路板中的应用	88	5.4.3 PC 操作系统	124
4.1 PICDEM机械电子板概述	88	5.4.4 实时操作系统	125
4.1.1 PICDEM 硬件	88	5.5 PIC16 C系统设计	126
4.1.2 电机驱动	91	5.5.1 硬件选择	126
4.1.3 测试程序	92	5.5.2 微控制器	127
4.1.4 调试	93	5.5.3 硬件设计	128
4.2 PICDEM 液晶显示器	94	5.5.4 软件设计	129
4.2.1 LCD连接	94	练习	129
4.2.2 LCD测试程序	96	作业	130
4.2.3 BCD 计数程序	98	附录A 使用 ISIS Schematic Capture	
4.3 PICDEM 直流电机测试程序	98	进行硬件设计	131
4.3.1 基本控制	99	附录B 使用 CCS C 进行软件设计	134
4.3.2 转数计数器	99	附录C 使用 Proteus VSM 进行系统	
4.4 PICDEM 步进电机控制	101	测试	141
4.4.1 结构	101	附录D C 编译器比较	145
4.4.2 步进电机的测试	102	附录E CCS C 编程语法小结	153
4.4.3 方向控制	103	附录F CCS C 编程函数参考	156
4.5 PICDEM 模拟传感器	104	答案	163
4.5.1 光传感器	105		
4.5.2 温度测量	106		

第 1 章 PIC 微控制器系统

1.1 PIC16 微控制器

- MCU 特性
- 程序执行
- RAM 文件寄存器
- 其他PIC芯片

目前在电子元件领域里，微控制器单元（MCU）可以做得相当细小，这是电子硬件持续小型化过程中的重要进步之一。如今即使音乐生日卡和电子价格标签这样小型的产品都带有一个MCU。MCU是模拟系统（如音频系统和电视）数字化的一个重要因素。另外，MCU更是汽车、机器人和工业系统等大型系统中不可或缺的部件。微控制器的应用无处不在，因此了解它的工作原理是十分有用的。

计算机或者数字控制器主要包括3个组件：用于与外界通信的输入/输出设备、用于计算和处理数据操作的处理器以及用于存放程序 and 数据的存储器。图1-1给出了较为详细的描述。与常规的微处理器系统（如PC）在印制电路板上连接多个独立芯片不同，微控制器在一个芯片上集成了所有的组件。MCU实际上是一个片上计算机，只不过它仍然需要诸如键盘和显示屏之类的输入/输出设备，才可以构成一个完整的工作系统。

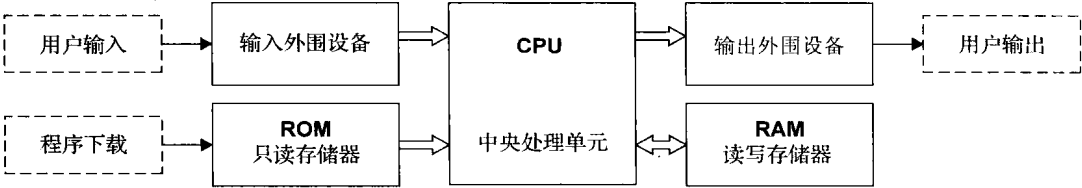


图1-1 数字控制器的组件

微控制器把程序存储在ROM（只读存储器）上。以前，UV（紫外线）可擦除编程ROM（EPROM）用于原型产品或小批量的产品，一次性可编程ROM用于使用周期更长的产品。可编程ROM芯片的编程在制造的最后阶段进行，而EPROM则由用户编程。

闪存ROM通常用于原型和低容量的产品。在设计好电路后，用户可以在电路内对闪存ROM进行编程。这样加快了原型开发周期，并且更容易适应软件的变化。现在，闪存ROM很常见，它们常被用作USB存储器、数码相机存储器等，一般容量都在吉字节级（GB）。

可用微控制器的产品系列正在快速地发展。最早广泛使用的Intel 8051是从早期的Intel PC处理器（如8086）发展而来的。这种微控制器在相当长的时间内一直独占市场，而其他的微控制器发展比较缓慢，主要以复杂处理器的形式出现，用于诸如发动机管理系统等应用中。这些微控制器比较昂贵，因此它们主要应用于高附加值的产品中。人们慢慢地意识到了微控制器的

应用潜力。

闪存ROM的发展促进了市场的扩张，Microchip公司首先受益。PIC16F84因其价格便宜并且支持多次编程而广为人知，迅速成为学生群体和业余爱好者的首选。在这次成功之后，Microchip产品线得到快速发展，并呈现多元化趋势。MPLAB是一款免费提供的开发支持系统，帮助PIC占领了低端市场。

闪存ROM是使得微控制器系统的学习更为容易和更加有趣的技术进步之一。而交互式电路设计软件则是又一种技术进步。整个设计过程更为透明，因此，初学者就可以更快地实现整个工作系统的构建。此外，低成本内电路调试是又一种技术，只需要在开发工具上投入适度的花费，就可以快速地构建硬件和运行系统。

1.1.1 MCU 特性

今天，微控制器得到了很好的发展，因为任何特定电路板上使用的MCU特性都必须尽可能与应用的实际需求相匹配。需要考虑的主要特性如下：

- 输入/输出数量；
- 程序存储器的容量；
- 数据RAM的容量；
- 非易失性数据存储器；
- 最大时钟速率；
- 接口数量；
- 开发系统支持；
- 成本与可用性。

PIC16F877A是一种很有用的参考器件，因为它只有一个极小的指令集，却具有全部的外围设备特性。这里介绍的微控制器应用设计的通用方法是，使用有多余容量的芯片开发设计，然后选择一种与应用需求最匹配的相关器件。尽可能选用较低级的芯片（如PIC10/12系列），如果

确实需要有更强大的功能，则可以选择更高指标的芯片（如PIC18/24系列）。由于所有的设备都有相同的核心体系结构和兼容的指令集，这是可行的。

PIC芯片之间最重要的区别在于指令集大小，这可以分为12位、14位和16位。后缀A用来指出芯片最大时钟速率是20 MHz，这主要是从最初的16F877器件升级而来。除此以外，这些芯片可以被视为相同的，后缀对于大多数用途是可选的。图1-2给出了16F877A的引脚图，而图1-3给出了芯片的内部体系结构，后者就是数据表中所定义方框图的简化版。

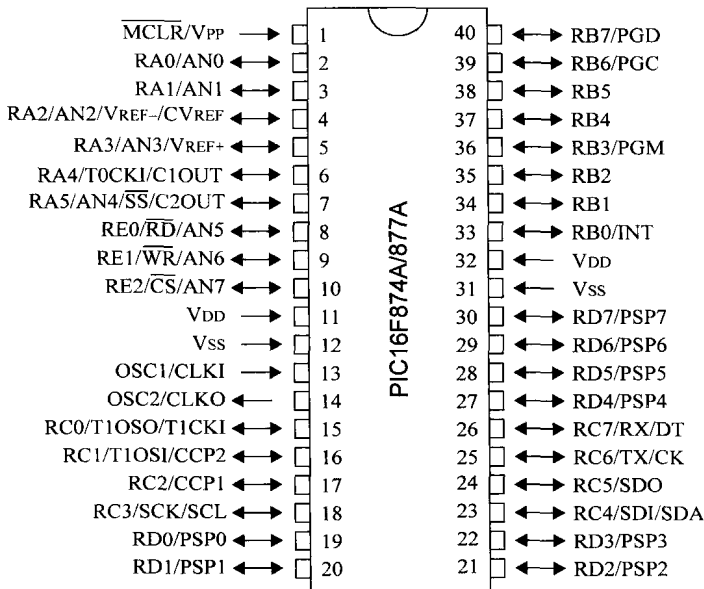


图1-2 16F877引脚（经Microchip公司许可复制）

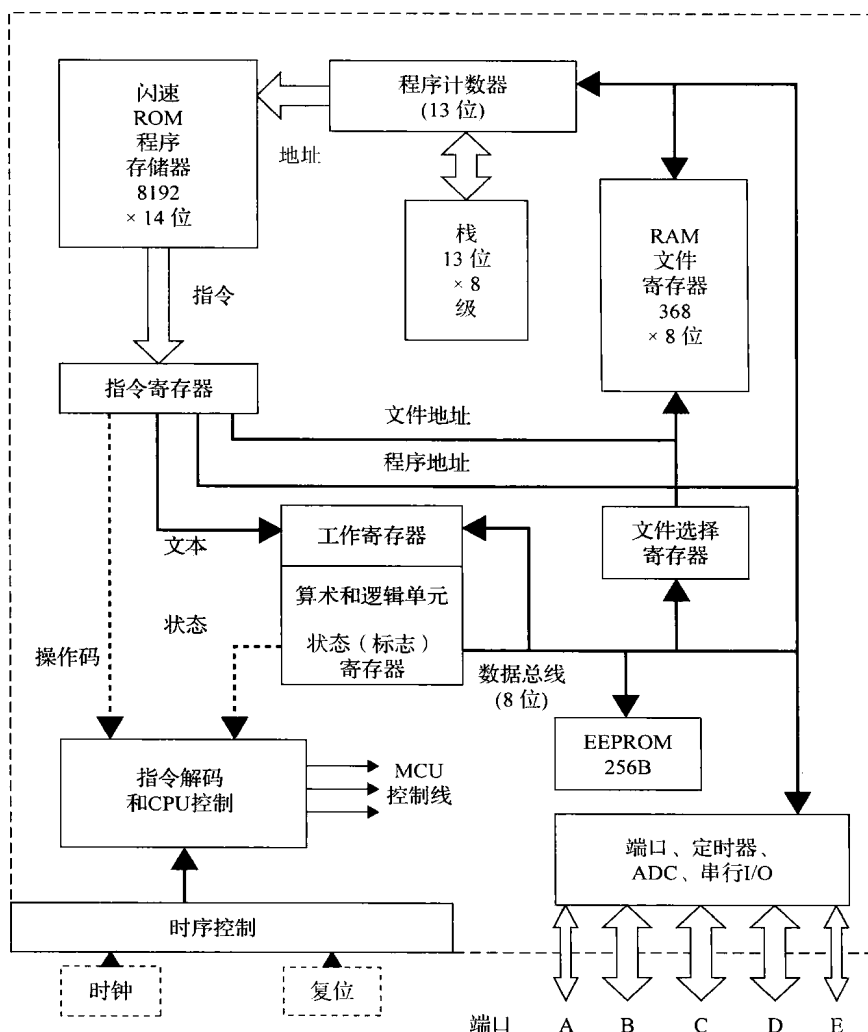


图1-3 PIC16F877 MCU 方框图

1.1.2 程序执行

芯片有8k (8096 × 14 bit) 的闪存ROM程序存储器，该存储器必须通过串行编程引脚PGM、PGC和PGD进行编程。定长的指令包含了操作码和操作数（立即数据、寄存器地址或跳转地址）。中等级别PIC的指令数量有限（只有35个），因此被归类为RISC（reduced instruction set compute，精简指令集）处理器。

观察内部体系结构图，可以确定程序执行中所用到的模块。程序存储器ROM包含了机器代码，地址为0000h~1FFFh (8 kB)。程序计数器用来保持当前指令的地址，每运行一步，值就增加或被修改。每次复位或上电时，它都被复位到0，地址0000的第一条指令被加载到指令寄存器，然后进行解码，执行。接下来，程序按指令顺序运行，操作文件寄存器（000~1FFh）的内容，执行数据传输指令，在端口与文件寄存器之间传输数据或者执行算术和逻辑指令来处理数据。CPU有一个主工作寄存器（W），所有数据必须通过它进行传递。

当一个分支指令（条件跳转）被解码后，就执行一个位测试。如果结果为真，则指令中包含的目标地址将被装载到程序计数器中，以强制跳转；如果结果为假，则执行顺序不变。在汇编语言中，使用CALL和RETURN来调用子程序时，会有相似的处理过程。栈用来存放返回地址，以便程序可以自动跳转回原来的程序位置。然而，CCS C编译器不支持这种机制，因为它将子程序（或C函数）的层数限制为8，即栈的深度。作为代替，使用一条简单的GOTO指令来进行函数的调用和返回，编译器自动计算出返回地址。

1.1.3 RAM 文件寄存器

如表1-1所示，主要的RAM模块是368个8位文件寄存器的集合，包括特殊功能寄存器（SFR），以及通用寄存器（GPR）。当在C程序中创建变量时，将变量存放到通用寄存器，起始地址为0020h。文件寄存器分为4个部分，即寄存器区0到寄存器区3。在每个RAM存储器区中，SFR位于低位的地址。

表1-1 PIC16F877简明的文件寄存器图

存储器区0 (000~07F)		存储器区1 (080~0FF)		存储器区2 (100~180)		存储器区3 (180~1FF)	
地址	寄存器	地址	寄存器	地址	寄存器	地址	寄存器
000h	间接	080h	间接	100h	间接	180h	间接
001h	定时器0	081h	选项	101h	定时器0	181h	选项
002h	程序计数器低位字节	082h	程序计数器低位字节	102h	程序计数器低位字节	182h	程序计数器低位字节
003h	状态寄存器	083h	状态寄存器	103h	状态寄存器	183h	状态寄存器
004h	文件选择	084h	文件选择	104h	文件选择	184h	文件选择
005h	端口A数据	085h	端口A方向	105h	—	185h	—
006h	端口B数据	086h	端口B方向	106h	端口B数据	186h	端口B方向
007h	端口C数据	087h	端口C方向	107h	—	187h	—
008h	端口D数据	088h	端口D方向	108h	—	188h	—
009h	端口E数据	089h	端口E方向	109h	—	189h	—
00Ah	程序计数器高位字节	08Ah	程序计数器高位字节	10Ah	程序计数器高位字节	18Ah	程序计数器高位字节
00Bh	中断控制	08Bh	中断控制	10Bh	中断控制	18Bh	中断控制
00Ch~01Fh	20个外围设备控制寄存器	08Ch~09Fh	20个外围设备控制寄存器	10Ch~10Fh	4个外围设备控制寄存器	18Ch~18Fh	4个外围设备控制寄存器
				110h~11Fh	16个通用寄存器	190h~19Fh	16个通用寄存器
020h~06Fh	80个通用寄存器	0A0h~0EFh	80个通用寄存器	120h~16Fh	80个通用寄存器	1A0h~1EFh	80个通用寄存器
070h~07Fh	16个公共访问GPR	0F0h~0FFh	访问070h~07Fh	170h~17Fh	访问070h~07Fh	1F0h~1FFh	访问070h~07Fh

有些寄存器的访问是可以跨存储器区边界的。例如，在每个存储器区的相应地址处，可以访问所有模块的状态寄存器。其他的寄存器则只能在特定的页面中进行访问，如端口A数据寄存器。有些寄存器地址在物理上没有实现。由于有些寄存器在多个存储器区中都是可以访问的，所以在编译机器码时，编译器可以将存储器区的切换次数降到最低，从而节省程序代码空间和执行时间。文件寄存器集的更多细节请参阅MCU数据表。

程序计数器使用两个8位寄存器来存储13位的程序存储器地址。只有地址002h的低位字节是直接寻址的。状态寄存器003h记录了ALU（算术逻辑单元）运算的结果，如结果为零、进位或借位。间接和文件选择寄存器用来对GPR地址进行索引寻址。在所有PIC MCU中，定时器0都是可用的定时器/计数器寄存器，而定时器1和定时器2寄存器则位于外围设备模块。端口寄存器位于存储器区0，地址从05h（端口A）到09h（端口E），每个端口的方向寄存器则位于存储器区1的相应位置。可以看到，共有 $80+16+80+96+96=368$ 个通用寄存器（GPR）可以用作数据RAM。注意，用于每个变量的寄存器数量取决于变量类型，可以从1位到32位不等（1~4个GPR）。

1.1.4 其他 PIC 芯片

在任意的嵌入式设计中，MCU特性都应该与实际应用需求相匹配。制造商需要确保随着应用需求的提高，持续提供功能更强大的常见类型的器件。我们可以从Microchip公司身上看到这一点。Microchip公司开始的时候生产基本的芯片（如16C84），后来为了满足市场需求扩大了产品规模。PIC微控制器目前分为10、12、16、18和24系列。它们的主要特性如表1-2所示。

表1-2 PIC微控制器类型

MCU	引脚数	数据字长（位）	程序存储器（字节）	典型指令集	速度（MIPS）	描 述
10FXXX	=6	8	≤512	33×12位	≤2	引脚数少，规格小，便宜，无EEPROM，无低功耗功能，汇编程序
12FXXX	=8	8	≤2 kB	12/14位	≤0.5	引脚数少，规格小，便宜，EEPROM，10位ADC，部分低功耗功能，汇编程序
16FXXX	≤64	8	≤14 kB	35×14位	≤5	引脚数中等，UART，I ² C，SPI，大多数低功耗，C或汇编程序
18FXXXX	≤100	8	≤128 kB	75×16位	≤16	引脚数多，CAN，USB J系列3 V电源供电，C程序
24FXXXX	≤100	16	≤128 kB	76×24位	=16	引脚数很多，3 V电源供电，无EEPROM，数据RAM≤8 kB，C程序

最初的16系列CMOS器件被命名为16CXX。闪存存储区问世后，最初的16系列器件就变成了16FXXX。目前，低引脚数（LPC）的芯片（如10/12系列）很少，而高引脚数芯片发展很快。此外，还有表中未列出的运行速度可达40 MIPS（百万条指令每秒）的24HXXXX系列芯片和高指标范围的dsPIC（数字信号处理器）。

1.2 PIC MCU 配置

- 时钟振荡器类型
- 看门狗、上电、掉电定时器
- 低电压编程
- 代码保护
- 内电路调试模式

在对PIC微控制器进行编程时，必须在下载主程序之前设置好操作模式。这是由独立于主程序存储器块的配置寄存器中的某些特定位来控制的。几种主要的选择如下。

1.2.1 时钟选择

‘877芯片有两种主要的时钟模式，即CR和XT。CR模式需要在CLKIN引脚连接一个简单的

电阻和电容电路,时钟周期由时间常数($C \times R$)决定。 R 的取值应在3 k Ω 到100 k Ω 之间, C 的值应大于20 pF。例如,如果 $R=10$ k Ω , $C=10$ nF,则时钟周期大约为 $2 \times C \times R = 200$ μ s (从CR电路的电压上升/下降时间计算得来),时钟频率大约为5 kHz。当对程序时序要求不高时,这种选择是可以接受的。

XT模式最为常用,这是因为与芯片本身的成本相比,其他外部元器件的成本较低,而且准确的时序通常很有必要。在CLKIN和CLKOUT引脚之间跨接一个外部晶振和两个电容。在该模式下,晶振频率可以在200 kHz到4 MHz之间,通常准确率高于50 ppm (parts per million, 百万分率),即0.005%。一般设置为4 MHz,因为这是一个标准晶体可能具有的最大振荡频率。这时,指令周期为1.000 μ s (每秒一百万条指令,即1MIP)。

低速率的晶振可以减少功率消耗,因为CMOS器件的功率消耗与时钟速率成正比。LP (低功耗) 模式支持32 kHz~200 kHz的时钟频率。如果要达到20 MHz的时钟速率,则需要使用HS (高速) 晶振,功率消耗也会相应增加。

在对芯片编程时,必须将MCU配置熔丝设置成需要的时钟模式。现在,许多PIC芯片都内置了振荡器,而不需要增加外部元件。内置振荡器比RC时钟的精度更高,但比晶振的要略低一些。它的常用运行频率是8 MHz,在芯片配置阶段可以进行校准,以得到一个更精确的时钟源。

1.2.2 配置选项

除了时钟外,还必须对几个其他的硬件选项进行配置选择。

1. 看门狗定时器

当启用看门狗定时器(WDT)时,WDT在给定的时间内(默认值为18 ms)自动地复位处理器。这样,程序可以跳出那些因软件设计人员无法预料的由程序错误或者运行时状态引起的死循环。为了维持正常的运行,必需禁用WDT或者在事先设定时间之内在程序循环内复位WDT。因此,很重要的一点是,如果不想使用WDT特性,则可以通过设置MCU配置位来禁用WDT。否则,程序会因为MCU的随机复位而出错。

2. 上电定时器

上电定时器(PuT)在电源电压到达操作电压和开始执行程序之间提供了标准的72 ms延迟。这样确保了在时钟启动前电源电压是稳定的。因为该功能对正常的程序执行没有任何负面影响,所以推荐使用。

3. 振荡器启动定时器

当上电定时器过期后,在开始执行程序之前还有进一步的延迟,以等待时钟稳定。当选定了晶振时钟模式后,CPU在被启用之前会等待1 024个周期。

4. 掉电复位(BoR)

瞬时的电压降低或电源掉电可能会中断MCU程序的执行。在启用掉电复位功能后,当电源电压低于指定的电压阈值时,掉电检测电路将保持MCU为复位状态;当电源电压恢复正常后才会释放复位。在CCS C里,低电压检测函数触发中断,让程序有序地重新执行。

5. 代码保护(CP)

在编程期间,可以将芯片配置成避免从芯片读取机器代码,从而保护商业利益和代码安全。可选择只让被选中的代码得到写入保护(参见稍后讲到的WRT_X%)。

6. 内电路编程和调试

现在,大部分PIC芯片都支持内电路编程和调试(ICPD)。这样,在主机系统的控制下,可以在目标硬件上下载和测试程序代码。在使用软件仿真消除大部分的程序错误后,可以使用这