

国外计算机科学经典教材

THOMSON

Embedded C Programming and the Microchip PIC

嵌入式 C 语言编程与 Microchip PIC

(美) Richard Barnett 等著
Larry O'Cull
薛晓东 秦新建 译



清华大学出版社

国外计算机科学经典教材

嵌入式 C 语言编程与 Microchip PIC

(美) Richard Barnett 等著
Larry O'Cull
薛晓东 秦新建 译



清华大学出版社

北 京

Richard Barnett Larry O'Cull et al
Embedded C Programming and the Microchip PIC
EISBN: 1-4018-3748-4

Copyright © 2004 by Delmar Learning, a division of Thomson Learning.

Original language published by Thomson Learning (a division of Thomson Learning Asia Pte Ltd).
All Rights reserved.

本书原版由汤姆森学习出版集团出版。版权所有，盗印必究。

Tsinghua University Press is authorized by Thomson Learning to publish and distribute exclusively this Simplified Chinese edition. This edition is authorized for sale in the People's Republic of China only (excluding Hong Kong, Macao SAR and Taiwan). Unauthorized export of this edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

本中文简体字翻译版由汤姆森学习出版集团授权清华大学出版社独家出版发行。此版本仅限在中华人民共和国境内(不包括中国香港、澳门特别行政区及中国台湾地区)销售。未经授权的本书出口将被视为违反版权法的行为。未经出版者预先书面许可，不得以任何方式复制或发行本书的任何部分。

981-265-586-7

北京市版权局著作权合同登记号 图字：01-2004-1055

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

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。

本书防伪标签采用特殊防伪技术，用户可通过在图案表面涂抹清水，图案消失，水干后图案复现；或将表面膜揭下，放在白纸上用彩笔涂抹，图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

嵌入式C语言编程与Microchip PIC/(美)巴内特(Barnett, R.)等著;薛晓东,秦新建译. —北京:清华大学出版社, 2005.7

书名原文: Embedded C Programming and the Microchip PIC
(国外计算机科学经典教材)

ISBN 7-302-10021-7

I. 嵌… II. ①巴…②薛…②秦… III. ①C语言—程序设计—教材②微控制器, PIC—教材
IV. ①TP312 ②TP332.3

中国版本图书馆CIP数据核字(2004)第124670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

组稿编辑: 曹 康

文稿编辑: 于 平

封面设计: 康 博

版式设计: 康 博

印刷者: 清华大学印刷厂

装订者: 三河市新茂装订有限公司

发行者: 新华书店总店北京发行所

开 本: 185×260 印张: 26.5 字数: 678千字

版 次: 2005年7月第1版 2005年7月第1次印刷

书 号: ISBN 7-302-10021-7/TP·6879

印 数: 1~4000

定 价: 49.00元

出版说明

近年来,我国高等学校的计算机学科教育进行了较大的改革,急需一批门类齐全、具有国际水平的计算机经典教材,以适应当前的教学需要。引进国外经典教材,可以了解并吸收国际先进的教学思想和教学方法,使我国的计算机学科教育能够与国际接轨,从而培育更多具有国际水准的计算机专业人才,增强我国信息产业的核心竞争力。Pearson、Thomson、McGraw-Hill、Springer、John Wiley 等出版集团都是全球最有影响的图书出版机构,它们在高等教育领域也都有着不凡的表现,为全世界的高等学校计算机教学提供了大量的优秀教材。为了满足我国高等学校计算机学科的教学需要,我社计划从这些知名的国外出版集团引进计算机学科经典教材。

为了保证引进版教材的质量,我们在全中国范围内组织并成立了“清华大学计算机外版教材编审委员会”(以下简称“编委会”),旨在对引进教材进行审定、对教材翻译质量进行评审。

“编委会”成员皆为全国各类重点院校教学与科研第一线的知名教授,其中许多教授为各校相关院、系的院长或系主任。“编委会”一致认为,引进版教材要能够满足国内各高校计算机教学与国际接轨的需要,要有特色风格,有创新性、先进性、示范性和一定的前瞻性,是真正的经典教材。为了保证外版教材的翻译质量,我们聘请了高校计算机相关专业教学与科研第一线的教师及相关领域的专家担纲译者,其中许多译者为海外留学回国人员。为了尽可能地保留与发扬教材原著的精华,在经过翻译和编辑加工之后,由“编委会”成员对文稿进行审定,以最大程度地弥补和修正在前面一系列加工过程中对教材造成的误差和瑕疵。

由于时间紧迫和能力所限,本套外版教材在出版过程中还可能存在一些不足和遗憾,欢迎广大师生批评指正。同时,也欢迎读者朋友积极向我们推荐各类优秀的国外计算机教材,共同为我国高等学校的计算机教育事业贡献力量。

清华大学出版社

国外计算机科学经典教材

编审委员会

主任委员:

孙家广 清华大学教授

副主任委员:

周立柱 清华大学教授

委员 (按姓氏笔画排序):

王成山	天津大学教授
王 珊	中国人民大学教授
冯少荣	厦门大学教授
冯全源	西南交通大学教授
刘乐善	华中科技大学教授
刘腾红	中南财经政法大学教授
吉根林	南京师范大学教授
孙吉贵	吉林大学教授
阮秋琦	北京交通大学教授
何 晨	上海交通大学教授
吴百锋	复旦大学教授
李 彤	云南大学教授
沈钧毅	西安交通大学教授
邵志清	华东理工大学教授
陈 纯	浙江大学教授
陈 钟	北京大学教授
陈道蓄	南京大学教授
周伯生	北京航空航天大学教授
孟祥旭	山东大学教授
姚淑珍	北京航空航天大学教授
徐佩霞	中国科学技术大学教授
徐晓飞	哈尔滨工业大学教授
秦小麟	南京航空航天大学教授
钱培德	苏州大学教授
曹元大	北京理工大学教授
龚声蓉	苏州大学教授
谢希仁	中国人民解放军理工大学教授

简 介

嵌入式微控制器就是一台微型计算机，它将其大部分的外设和所需要的存储器连同 CPU 一起包含在一个单独的集成电路中。它是真正意义上的“集成在一块芯片上的微型计算机”。嵌入式微控制器的应用时间已经超过 30 年之久。Intel 公司的 8051 系列微控制器是最早将存储器、I/O 设备、算术逻辑单元(ALU)、程序只读存储器(ROM)以及一些其他的外设统统集成在一个非常小的封装中的微控制器之一。直到现在这些处理器仍然用来设计新的产品。其他在 Intel 的引领下进入嵌入式微控制器领域的公司有 General Instruments 公司、National Semiconductor 公司、Motorola 公司、Philips/Signetics 公司、Zilog 公司、AMD 公司、Hitachi 公司、Toshiba 公司、Microchip Technology 公司、以及 Atmel 等公司。

在过去 10 年中，Microchip Technology 公司在精简指令集计算(RISC)的核心架构的研发上成为业界的领导者，这种架构可以提供成本非常低却非常优秀的解决方案。PIC 处理器家族是基于 EEPROM 存储器的，直到最近才开始转为基于 FLASH 存储器技术。FLASH 技术是永久性且可以重复编程的存储器技术，被广泛地应用于数码照相机、便携式音响设备和个人电脑主板等产品上面。这项存储技术使得 Microchip 公司可以通过提供一套在系统内可编程的解决方案来推动微控制器工业的发展。

在这个高科技发展过程中接下来的重要一步就是，实现专用于这些新的微处理器的高级语言编译器。这些编译器的代码生成和代码优化的功能是强大的。C 语言具有灵活的编程风格和可以自定义结构等优点，由于它既能够适用于某个特定目标系统，同时其代码仍然能够被移植到其他系统中，因此 C 语言被引入单片机编程领域。这类语言的关键优势是，它建立了一些知识库，这些库文件可以被反复应用。在此基础上，各项后续项目的开发周期就被缩短了，从而开发成本也就降低了。

迄今为止专门为 Microchip 公司的 PIC 微处理器家族设计的最好的 C 语言工具之一是 PIC-C 语言。它由 Custom Computer Services 公司开发研制，这个完整的集成开发环境(IDE)使得文本编辑、编译、部分编程和调试都可以由一个 PC Windows 应用程序完成。作者编写本书的动力在于 PIC 和其他 RISC 微控制器的应用日益普及、集成度不断提高(芯片内功能的高度集成以及电路板上的芯片数的不断减少)和利用这种技术开发产品的时候对“兼容性思想”的需要。尽管您可能有为 PC 编写 C 程序或者为一个微控制器编写汇编语言程序的经验，但是当为嵌入式微控制器编写 C 程序的时候，必须调整思路来获得理想的最终结果：精简、有效、可靠、可重用代码。本书对初学者来说是一本很好的基础教科书，同时也是对有经验的嵌入式微控制器设计人员有帮助的一本参考工具书。

前 言

《嵌入式 C 语言编程与 Microchip PIC》一书的宗旨在于讲授 C 语言编程及其 C 语言编程在嵌入式微控制器中的应用,同时本书还讲述了一些 Microchip 公司的 PIC RISC 微控制器家族的应用实例。

读者对象

本书适用于以下两类读者对象:

- 电机和计算机工程、电子工程、电机工程技术和计算机工程技术专业的学生。本书非常适合下列两类情况的学生:
 - 没有学习过 C 语言编程课程的初学者:本书可以作为两个学期或者四个季度的 C 语言编程课程的教材,该课程将教授学生 C 语言编程方法以及如何把 C 语言应用于嵌入式微控制器设计中。在此基础上,学生可以深入到更复杂的嵌入式应用中,这些应用程序都可以在嵌入式微控制器上运行,并且学生只需要很少的硬件知识。在第 1 章嵌入式 C 语言指南结束之后,本书就可以作为该课程的其他部分的编程参考书。
 - 已经学习过 C 语言编程课程的学生可以将本书作为一个学期或者两个季度的嵌入式微控制器的课程。在这种情况下,学生只需要学习第 1 章中那些与嵌入式环境有关的部分,就可以尽快转向高级硬件概念。第 1 章的结构(本书其他部分的结构也一样)给其他课程所需的信息提供了一个有用的参考。
- 想熟练应用一种新的微控制器的实习工程师、工艺人员和技术人员:根据实际需要,第 1 章可以用于学习必需的概念,也可以用作参考资料(这取决于用户的编程经验)。第 2 章中讨论了 Microchip PIC 微控制器硬件,它可以引导用户逐步学习一种新的微控制器并能够作为以后项目的参考。

预备知识

需要一些数字系统、记数法和逻辑设计的知识。第 1 章嵌入式 C 语言指南的最初版本已经成功地用于一门基础微控制器课程(大学二年级、没有预先必要的编程经验),该课程是在两个学期的基本数字逻辑课程结束之后开设的。本书也是一本优秀的高级微控制器选修课程教材。在很多情况下,许多学生会选择保存本书,将它作为高级项目设计课程的参考书,并带到了实际工作中,作为有用的参考书。

本书结构

本书按照逻辑主题单元组织。教师可以按照本书的顺序,从 C 语言开始,逐渐到 PIC 硬件

再到更加高级的主题,也可以选择自行安排主题的顺序来满足特定的要求。各主题都是独立的,非常便于选择。章节练习题和上机练习也是根据主题分开,使得读者可以很容易地选择适用于特定实例的内容。

章节内容概述

第1章 嵌入式C语言指南

本章逐步详尽地介绍了嵌入式微控制器的C语言编程方法。每个编程的概念都配有一个或多个示例程序来解释它的使用方法。学习完本章,学生将能够编写C语言程序解决问题。本章将CCS-PICC C编译器作为应用Microchip PIC的C编译的模型,上述基本概念也可以应用到其他PIC C编译器中。

第2章 PIC微控制器硬件

PIC RISC 处理器部分讲述了从基本结构到微控制器包含的所有标准外设的使用方法。每个外设都有说明其常规用法的示例程序。学习完第1章、第2章之后,学生将能够应用PIC RISC处理器解决实际问题。

第3章 标准I/O和处理器指令

本章介绍了C语言中的内嵌函数及其用法。同样有示例程序介绍使用内嵌函数的方法。学习完第3章之后,学生在解决实际问题时通过应用内嵌函数就可以加快编程速度、提高效率。

第4章 CCS-PICC C编译器和IDE

本章可以作为CCS-PICC C编译器及其相应的集成开发环境(IDE)的使用手册。Microchip公司的MPLAB调试环境可以从Microchip公司的网站<http://www.microchip.com>获得。

第5章 项目开发

本章着重讲述了一个使用微控制器的项目的开发步骤。作者在本章描述了电动脚踏车的完整开发步骤以解释这个过程。学生可以学会如何高效地开发项目才能够获得最大的成功。

附录

附录A, 库函数参考。本附录对在本书出版时可用的内嵌库函数做了详尽介绍。

附录B, PIC微控制器的编程。本附录是对PIC设备的FLASH存储区域进行的实际编程的指南,学生通过对它的学习可以加深对编程函数的理解。

附录C, CCS ICD-S 串行系统内编程器/调试器。本附录是对CCS ICD-S编程和调试工具的简介。

附录D, Microchip ICD-II 串行系统内编程器/调试器。本附录是对Microchip ICD-II编程和调试工具的介绍。

附录E, “FlashPIC-Dev”开发板。本附录介绍了Progressive Resource LLC公司开发的“FlashPIC-Dev”开发板。

附录F, ASCII表。

附录G, PIC16F877指令集概要。本附录使用PIC16F877汇编代码编程示例来介绍汇编

代码指令集概要。

附录 H, PIC18F458 指令集概要。汇编代码指令集概述的参考材料。

附录 I, 精选问题的答案(按章节排序)。

基本原理

不断提高的微控制器技术为人们持续提供了大量令人惊叹的功能和极高的运行速度。这些改进推进了诸如 C 语言的高级语言的全面运用, 包括以往都是由汇编语言程序完成的、对时间要求苛刻的任务。同时, 微处理器变得越来越容易使用, 并成为进行教学的完美工具。许多学校将微控制器设备作为课程的目标设备。微控制器开发板的价格下降到如此程度, 使得许多学校指定学生购买此板作为“部件工具箱”的一部分, 因此, 所有的学生拥有自己的开发板。这样的课程要求 C 语言编程作为预备知识, 另外一些课程则综合地讲述 C 语言编程以及嵌入式微控制器的应用。

本书是一本顺应时代需要的教科书, 它可作为需要有 C 语言编程基础或者不需要该基础的课程的教材, 也可以作为其他后续课程的有用的参考书。

使用的硬件

本书中的绝大多数应用程序都是使用“FlashPIC-Dev”评估板开发的, 该评估板是由 Progressive Resources, LLC 公司提供的(详细信息参见附录 E)。这种评估板非常适合教学使用, 也是一个很好的通用目标开发板。然而, Microchip PIC 微控制器很容易使用, 只要将它们插入原型板, 加上晶振, 附加两个电容器, 连接四条编程线, 它就能很好地工作。学生对每种方法的应用都会很成功。

PIC16F877 和 PIC18F458 微控制器用来实现本书中的例子。PIC 微控制器的一个主要优点是它们的体系结构以及对设备的编程方式均相似。这意味着本书所有的例子能够运行于任何实际的 PIC 微控制器, 只要它包含了完成工作所需要的外设和其他资源, 代码无需改动就可以用在 PIC 家族的其他成员上。总的来说, 本书也适合于 PIC 家族的其他成员。

本书还介绍了更多的通用外设, 代码可作为模板应用于 PIC 家族某些其他成员中的非常规外设。

目 录

第 1 章 嵌入式 C 语言指南	1
1.1 目标	1
1.2 介绍	1
1.3 基础概念	1
1.4 变量和常量	3
1.4.1 变量类型	3
1.4.2 变量使用范围	5
1.4.3 常量	5
1.4.4 枚举和定义	7
1.4.5 存储分类	8
1.4.6 类型转换	8
1.5 I/O(输入输出)操作	9
1.6 运算符和表达式	10
1.6.1 赋值和算术运算符	10
1.6.2 逻辑和关系运算符	13
1.6.3 增量、减量和复合赋值运算	14
1.6.4 条件表达式	15
1.6.5 运算符的优先级	16
1.7 控制语句	17
1.7.1 while 循环	17
1.7.2 do/while 循环	19
1.7.3 for 循环	20
1.7.4 if/else	21
1.7.5 switch/case	24
1.7.6 break、continue 和 goto	26
1.8 函数	30
1.8.1 原型和函数组织	31
1.8.2 有返回值的函数	33
1.8.3 递归	34
1.9 指针和数组	37
1.9.1 指针	38
1.9.2 数组	41
1.9.3 多维数组	43
1.9.4 函数的指针	45
1.10 结构体和共用体	50

1.10.1	结构体	50
1.10.2	结构体数组	52
1.10.3	结构体的指针	52
1.10.4	共用体	54
1.10.5	typedef 操作符	56
1.10.6	位和位段	56
1.10.7	sizeof 操作符	58
1.11	存储器类型	59
1.11.1	常量和变量	59
1.11.2	寄存器变量	62
1.12	实时方法	65
1.12.1	使用中断	65
1.12.2	状态机	68
1.13	本章小结	73
1.14	练习题	74
1.15	上机练习	75
第 2 章	PIC 微控制器的硬件	77
2.1	目标	77
2.2	简介	77
2.3	体系结构总览	77
2.4	存储器的组织方式	79
2.4.1	数据存储器	79
2.4.2	FLASH 存储器	80
2.4.3	返回地址堆栈	80
2.5	中断和重置	81
2.6	I/O 端口	84
2.7	定时器	89
2.7.1	一般功能概要	89
2.7.2	定时器 0	93
2.7.3	定时器 1	95
2.7.4	定时器 2	103
2.7.5	看门狗定时器	108
2.8	串行 I/O	109
2.8.1	异步串行端口	109
2.8.2	CAN 总线模块	116
2.8.3	同步串行端口	119
2.9	模拟到数字转换的 I/O	125
2.9.1	模拟到数字转换的背景	125
2.9.2	模拟到数字转换的模块	126

2.10	断电(睡眠)模式	131
2.11	汇编语言	132
2.12	本章小结	134
2.13	练习题	137
2.14	上机练习	138
第3章	标准 I/O 和处理器指令	140
3.1	目标	140
3.2	简介	140
3.3	字符输入/输出函数—— <code>getchar()</code> 和 <code>putchar()</code>	140
3.4	标准输出函数	145
3.4.1	输出字符串 <code>puts()</code> 和“文件”输出字符串 <code>fputs()</code>	145
3.4.2	格式化输出 <code>printf()</code> 和格式化文件输出 <code>fprintf()</code>	146
3.5	标准输入函数	149
3.5.1	输入字符串函数—— <code>gets()</code> 和 <code>fgets()</code> 函数	149
3.5.2	输入字符串函数—— <code>get_string()</code>	150
3.6	标准预处理器指令	151
3.6.1	<code>#include</code> 指令	151
3.6.2	<code>#define</code> 指令	152
3.6.3	<code>#ifdef</code> 、 <code>#ifndef</code> 、 <code>#else</code> 和 <code>#endif</code> 指令	154
3.6.4	<code>#error</code> 指令	159
3.6.5	<code>#pragma</code> 指令	159
3.7	CCS-PICC 函数限定指令	160
3.7.1	<code>#inline</code> 和 <code>#separate</code> 指令	160
3.7.2	<code>#int_default</code> 、 <code>#int_global</code> 和 <code>#int_xxx</code> 指令	160
3.8	CCS-PICC 预定义标识符	162
3.9	CCS-PICC 设备相关的特定指令	162
3.9.1	<code>#device</code> 指令	162
3.9.2	<code>#fuse</code> 指令	163
3.9.3	<code>#id</code> 指令	164
3.10	CCS-PICC 内部库预处理器指令	165
3.10.1	<code>#use delay</code> 指令	165
3.10.2	<code>#use fast_io</code> 、 <code>#use fixed_io</code> 和 <code>#use standard_io</code> 指令	165
3.10.3	<code>#use i2c</code> 指令	167
3.10.4	<code>#use rs232</code> 指令	167
3.11	CCS-PICC 存储控制预处理器指令	169
3.11.1	<code>#type</code> 指令	169
3.11.2	<code>#bit</code> 指令	169
3.11.3	<code>#byte</code> 指令	170
3.11.4	<code>#locate</code> 指令	170

3.11.5	#reserve 指令	170
3.11.6	#zero_ram 指令	171
3.11.7	#rom 指令	171
3.11.8	#org 指令	171
3.11.9	#asm 和#endasm 指令	172
3.12	CCS-PICC 编译器控制预处理器指令	173
3.12.1	#case 指令	173
3.12.2	#OPT 指令	174
3.12.3	#priority 指令	174
3.13	本章小结	174
3.14	练习题	175
3.15	上机练习	176
第 4 章	CCS-PICC C 编译器和 IDE	177
4.1	目标	177
4.2	简介	177
4.3	集成开发环境	177
4.4	项目	178
4.4.1	打开已有的项目	178
4.4.2	创建新的项目	179
4.4.3	为一个项目设定包含目录	180
4.4.4	编译项目	180
4.4.5	关闭项目	181
4.5	PIC Wizard 代码生成器	181
4.5.1	General 选项卡	183
4.5.2	通信选项卡	183
4.5.3	“SPI and LCD” 选项卡	184
4.5.4	Timers 选项卡	184
4.5.5	Analog 选项卡	185
4.5.6	Interrupts 选项卡	185
4.5.7	Drivers 选项卡	186
4.5.8	I/O Pins 选项卡	187
4.5.9	已生成的项目	187
4.6	源文件	188
4.6.1	打开一个现存的源文件	188
4.6.2	生成一个新的源文件	188
4.6.3	改变项目的主源文件	189
4.7	编辑器操作	189
4.7.1	书签	189
4.7.2	缩进和 Tab 键	189

4.7.3	括号匹配	190
4.7.4	语法高亮显示	190
4.7.5	其他编辑器选项	190
4.8	View 菜单	191
4.8.1	C/ASM List 选项	191
4.8.2	Symbol Map 选项	191
4.8.3	Call Tree 选项	192
4.8.4	Statistics 选项	193
4.8.5	Compiler Messages 选项	193
4.8.6	Data Sheet 选项	193
4.8.7	Valid Fuses 选项	193
4.8.8	Valid Interrupts 选项	193
4.8.9	Binary File 选项	194
4.8.10	COD Debug File 选项	194
4.9	目标设备编程	194
4.10	Tool 菜单	195
4.10.1	Device Editor 选项	195
4.10.2	Device Selector 选项	195
4.10.3	File Compare 选项	195
4.10.4	Numeric Converter 选项	196
4.10.5	Serial Port Monitor 选项	196
4.11	Microchip MPLAB	197
4.11.1	从 CCS-PICC 启动 MPLAB	198
4.11.2	MPLAB 工作空间和项目	198
4.11.3	模拟器开发模式	199
4.11.4	在 MPLAB 中进行编译	199
4.11.5	源文件和程序存储器窗口	199
4.11.6	执行速度	199
4.11.7	调试命令	200
4.11.8	设置和取消断点	200
4.11.9	Run to Cursor	201
4.11.10	Watch	201
4.11.11	File Registers(RAM)窗口	201
4.11.12	修改存储器	202
4.11.13	查看和修改状态机	202
4.12	本章小结	203
4.13	练习题	203
4.14	上机练习	204

第 5 章 项目开发	205
5.1 目标	205
5.2 绪论	205
5.3 构思开发阶段	205
5.4 项目开发过程步骤	205
5.4.1 定义阶段	205
5.4.2 设计阶段	207
5.4.3 测试定义阶段	208
5.4.4 构造和测试硬件原型的阶段	208
5.4.5 系统整合和软件开发阶段	209
5.4.6 系统测试阶段	209
5.4.7 庆功阶段	209
5.5 项目开发过程总结	209
5.6 示例项目：电动脚踏车	210
5.6.1 概念阶段	210
5.6.2 定义阶段	210
5.6.3 设计的系统考虑	214
5.6.4 硬件设计-驱动单元	222
5.6.5 软件设计-驱动单元	226
5.6.6 硬件设计-显示单元	229
5.6.7 软件设计-显示单元	232
5.6.8 测试定义阶段	234
5.6.9 构造和测试硬件原型阶段	234
5.6.10 系统整合和软件开发阶段——驱动单元	239
5.6.11 系统整合和软件开发阶段——显示单元	263
5.6.12 系统测试阶段	271
5.7 改进	275
5.8 本章小结	275
5.9 练习题	275
5.10 上机练习	276
附录 A 库函数参考	277
附录 B PIC 微控制器编程	363
B.1 同步端口编程	363
B.2 商用编程器	364
B.3 Boot Loader 编程	365
附录 C CCS ICD-S 串行系统内编程器/调试器	366
附录 D Microchip ICD 2 串行系统内编程器/调试器	367
D.1 用于 PIC FLASH 产品的调试器解决方案	367
D.2 MPLAB ICD 2 电路内调试器设置	367

D.3 支持的 PIC FLASH 产品	368
附录 E “FlashPIC-DEV” 开发板	370
E.1 规格	371
E.2 应用笔记	371
E.2.1 电源	371
E.2.2 串行连接	372
E.2.3 SPI 连接	373
E.2.4 并行端口	373
E.2.5 系统时钟	373
E.2.6 CAN 接口	373
附录 F ASCII 表	374
附录 G PIC16F877 指令集摘要	378
附录 H PIC18F458 指令集摘要	390
H.1 指令集摘要	390
H.2 READ-MODIFY-WRITE 操作	391
附录 I 精选问题的答案(按章排列)	396

第 1 章 嵌入式 C 语言指南

1.1 目标

学习本章应该掌握以下几点内容：

- 定义、描述、区分变量和常量的类型、使用范围和用法。
- 创建各种比特数的数值类型和字符串类型的变量和常量的声明。
- 对变量声明应用枚举。
- 通过赋值运算符给变量和常量赋值。
- 估计 C 语言中各类运算符的运算结果。
- 解释程序执行流程中每条控制语句的执行结果。
- 创建由变量、运算符和控制语句构成的函数完成任务。
- 使用指针、数组、结构体和共用体作为函数变量。
- 使用本章中的概念创建 C 语言程序来完成任务。

1.2 介绍

本章是 C 语言应用于嵌入式微控制器应用程序的一个基础教程。本章包括了 C 语言的扩展内容，那就是 CCS-PICC C 语言的一部分。读者将从基础概念开始学习，直到能够编写完整的程序。本章还列举了很多应用于微控制器的实例来进一步充实内容。

本章的知识是按照程序员需要的顺序组织的：

- 声明变量和常量
- 简单的输入输出操作，使得程序可以利用微控制器的并行端口
- 给变量和常量赋值，并使用变量做算术运算
- 使用 C 结构体和控制语句来组成完整的 C 程序

本章的最后部分涵盖了一些更高级的主题，例如指针、数组、结构体和共用体，以及它们在使用。本章的结尾是一些高级概念，例如实时编程和中断等。

1.3 基础概念

编写 C 程序从某种意义上说就像是盖砖房：先要打好地基，再用沙子和水泥做好砖头，然后将砖头按行排列砌好一层，最后一层层垒起来就盖好了一幢房子。在嵌入式 C 程序中，一组的指令放在一起构成了函数，这些函数可以被视为更高级的操作，然后函数组合起来就构成了一个程序。

每个 C 语言程序至少要有有一个函数，也就是 `main()` 函数。`main()` 函数是整个 C 语言程序的