



单片机系统设计

DANPIANJI XITONG SHEJI

主 编 罗德雄 黄应强
副主编 孙 鸿



单片机系统设计

主 编 罗德雄 黄应强
副主编 孙 鸿
参 编 张 艳 彭永杰

机械工业出版社

本书基于电子信息工程技术专业进行的“EP-CDIO”教学模式，从实际应用入手，以项目为载体，以“做中学”形式为基础，教师的主导为辅，循序渐进地介绍 51 单片机 C 语言编程方法及 51 单片机的硬件结构和功能应用，以及单片机系统软硬件的设计、仿真和调试方法，提升读者软硬件系统的设计能力。

全书共 6 个项目，内容丰富，实用性强，书中大部分内容均来自企业及教学实践，许多 C 语言代码可以直接应用到项目实践中，可作为高等专科和高职院校相关专业单片机课程教材，适合于 51 单片机的初学者，可供从事自动控制、机电一体化、电力电子、智能仪器仪表等专业的技术人员参考。

图书在版编目 (CIP) 数据

单片机系统设计/罗德雄，黄应强主编. —北京：机械工业出版社，2015.6

ISBN 978-7-111-50804-5

I. ①单… II. ①罗… ②黄… III. ①单片微型计算机-系统设计-高等教育-教材 IV. ①TP368.1

中国版本图书馆 CIP 数据核字 (2015) 第 154532 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑：舒雯 责任编辑：舒雯 版式设计：霍永明

责任校对：黄兴伟 封面设计：鞠杨 责任印制：康朝琦

北京京丰印刷厂印刷

2015 年 9 月第 1 版第 1 次印刷

184mm×260mm·13.5 印张·329 千字

0 001—1 000 册

标准书号：ISBN 978-7-111-50804-5

定价：48.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

网络服务

服务咨询热线：010-88379833 机工官网：www.cmpbook.com

读者购书热线：010-88379649 机工官博：weibo.com/cmp1952

教育服务网：www.cmpedu.com

封面无防伪标均为盗版

金书网：www.golden-book.com

前 言

单片机技术迅速发展,采用单片机开发的智能化测控设备和产品广泛应用到各个领域,单片机技术产品和设备促进了生产技术水平的提高。企业迫切需要大量熟练掌握单片机技术,并能开发、应用和维护管理这些智能化产品的高级工程技术人才。

单片机以体积小、功能强、可靠性好、性价比高等特点,已成为实现工业生产技术进步和开发机电一体化和智能化测控产品的重要器件。

本书是在电子信息工程技术专业进行的“EP-CDIO”教学模式下,从实际应用入手,以项目为载体,以“做中学”的形式为基础、教师的主导为辅,循序渐进地介绍51单片机C语言编程方法及其硬件结构和功能应用,以及单片机系统软硬件的设计、仿真和调试方法,提升读者软硬件系统设计能力。

本书共6个项目,内容丰富,实用性强,书中大部分内容均来自企业及教学实践,许多C语言代码可以直接应用到项目实践中,可作为高等专科和高职院校相关专业单片机课程教材,适合于51单片机的初学者,也可供从事自动控制、机电一体化、电力电子、智能仪器仪表等专业的技术人员学习参考。

本书项目1和项目2为C语言程序设计基础部分,首先介绍C语言的发展及特点,并通过实例重点介绍C语言程序的基本结构和使用VC++6.0开发C语言的过程;然后通过“简单功能计算器”和“学生成绩统计系统”两个项目介绍C语言的语法和程序结构、函数、数组等相关知识,培养读者C程序编写的基本能力。项目3为小车状态指示灯控制器,实现了C语言在单片机控制中的应用,达到了C语言与单片机知识衔接的目的,增强读者对C程序作用的认识,进一步提高读者C程序的编程能力。

项目4~项目6为51单片机应用系统设计部分,书中通过对智能小车寻迹、循线和控制系统的设计,介绍51系列单片机基础知识、端口应用、内部资源的使用、传感器和外围电路的识别与应用的知识,同时提高读者单片机系统开发中的软硬件设计能力。

本书的项目1由黄应强编写,项目2由张艳编写,项目3由彭永杰编写,项目4~6由罗德雄编写,在编写过程中得到宜宾机电一体化研究所所长孙鸿和学院各级领导的指导和大力支持,在此一并表示感谢。

编 者

目 录

前言

项目 1 简易功能计算器	1
【学习目标】	1
【项目描述】	1
【项目实施】	1
任务 1 设计方案	1
任务 2 软件设计	2
任务 3 系统仿真与调试	4
【相关知识】	7
1.1 C 语言基础	7
1.1.1 C 语言的发展及特点	7
1.1.2 简单 C 程序介绍	8
1.1.3 C 程序的开发过程	10
1.1.4 VC++6.0 集成开发环境	10
1.1.5 数据类型和数据运算	14
1.2 流程图	21
1.2.1 流程图的作用	21
1.2.2 流程图的组成	21
1.2.3 流程图的绘制方法	22
1.3 C 语言程序语句	23
1.3.1 控制语句	23
1.3.2 表达式语句	23
1.3.3 函数调用语句	23
1.3.4 空语句	23
1.3.5 复合语句	23
1.3.6 格式化输入/输出函数	23
1.4 算法与程序的三个基本结构	29
1.4.1 算法的概念	29
1.4.2 算法的特性	29
1.4.3 程序的三种基本控制结构	29
【能力拓展】	48
【项目考核】	48
项目 2 学生成绩统计系统	50
【学习目标】	50
【项目描述】	50
【项目实施】	50
任务 1 设计方案	50

任务 2 软件设计	51
任务 3 系统仿真与调试	56
【相关知识】	58
2.1 数组	58
2.1.1 一维数组的定义和引用	58
2.1.2 二维数组的定义和引用	62
2.1.3 字符数组	64
2.2 函数	68
2.2.1 概述	68
2.2.2 函数定义的一般形式	70
2.2.3 函数的参数和函数的值	71
2.2.4 函数的调用	73
2.2.5 函数的嵌套调用	74
2.2.6 函数的递归调用	75
2.2.7 数组作为函数参数	77
2.2.8 局部变量和全局变量	79
2.3 指针	82
2.3.1 地址指针的基本概念	82
2.3.2 变量的指针和指向变量的 指针变量	83
2.3.3 数组指针和指向数组的指针 变量	86
【能力拓展】	87
【项目考核】	88
项目 3 小车状态指示灯控制器	89
【学习目标】	89
【项目描述】	89
【主要的仪器设备和元器件】	89
【项目实施】	90
任务 1 总体设计	90
任务 2 硬件设计	90
任务 3 软件设计	91
任务 4 系统仿真与调试	92
【相关知识】	93
3.1 单片机基础	93
3.1.1 单片机的发展概况	93
3.1.2 单片机的应用	94

3.2 单片机的内部结构	95	任务2 硬件设计	144
3.2.1 CPU 系统	95	任务3 软件设计	145
3.2.2 存储器系统	100	任务4 系统仿真与调试	149
3.2.3 并行 I/O 端口系统	107	【相关知识】	150
3.2.4 其他功能单元	110	5.1 数码管及其显示控制	150
3.3 单片机的外部结构	110	5.1.1 数码管的结构	151
3.3.1 单片机的引脚功能	110	5.1.2 数码管的工作原理	151
3.3.2 复位电路	112	5.1.3 数码管的字形编码	151
3.4 单片机执行指令过程	113	5.1.4 LED 数码管的显示方式	152
3.5 软件 Proteus 和 Keil C 的使用	114	5.2 定时/计数器	153
3.5.1 软件 Proteus 的使用	114	5.2.1 定时器/计数器	153
3.5.2 Keil 软件的使用	115	5.2.2 定时/计数器方式控制寄存器 TMOD	154
3.6 独立式键盘	126	5.2.3 定时器/计数器的控制寄存器 TCON	154
3.6.1 键盘接口技术	126	5.2.4 定时/计数器的初始化	155
3.6.2 独立式键盘	126	5.2.5 定时器/计数器的工作方式	155
3.6.3 键盘程序设计流程	127	5.3 中断系统	159
【能力拓展】	128	5.3.1 中断系统的结构	160
【项目考核】	128	5.3.2 中断系统控制	161
项目4 智能小车寻迹控制系统	130	5.3.3 中断系统响应	163
【学习目标】	130	5.3.4 中断系统应用	164
【项目描述】	130	【能力拓展】	165
【主要的仪器设备和元器件】	130	【项目考核】	166
【项目实施】	131	项目6 智能小车控制系统	167
任务1 总体设计	131	【学习目标】	167
任务2 硬件设计	132	【项目描述】	167
任务3 软件设计	133	【主要的仪器设备和元器件】	167
任务4 系统仿真与调试	137	【项目实施】	168
【相关知识】	137	任务1 总体设计	168
4.1 直流电动机及其控制	137	任务2 硬件设计	168
4.1.1 H 桥驱动电路	137	任务3 软件设计	169
4.1.2 使能控制和方向逻辑	138	任务4 系统仿真与调试	175
4.2 寻迹传感器	140	【相关知识】	176
4.2.1 光电传感器构成的循迹检测 电路	140	6.1 矩阵式键盘	176
4.2.2 黑线检测原理	140	6.1.1 扫描方式键盘的程序设计 流程	176
【能力拓展】	141	6.1.2 键盘扫描程序	177
【项目考核】	141	6.1.3 键盘扫描的中断控制方式	178
项目5 智能小车循线控制系统	143	6.2 D-A 转换器	178
【学习目标】	143	6.2.1 D-A 转换器的基本原理和 主要技术指标	178
【项目描述】	143	6.2.2 集成 D-A 转换器——	
【主要的仪器设备和元器件】	143		
【项目实施】	144		
任务1 总体设计	144		

DAC0832	179
6.2.3 DAC0832 和 MCS-51 的接口 ...	180
6.3 液晶显示器及其工作原理	182
6.3.1 液晶显示器的特点	182
6.3.2 液晶显示器的分类	182
6.3.3 液晶显示器的工作原理	183
6.3.4 液晶显示器的技术参数	183
6.4 LCD1602 的使用	184
6.4.1 点阵字符型 LCD 的接口 特性	184
6.4.2 点阵字符型液晶显示模块的 基本特点	184
6.4.3 HD44780 的引脚与时序	185
6.4.4 HD44780 的编程结构	186
6.4.5 1602 字符型 LCD 指令集	187
6.4.6 LCD1602 与单片机的接口	

设计	188
6.4.7 程序设计	189
6.5 测距传感器	191
6.5.1 测距工作原理	191
6.5.2 超声波发射电路	192
6.5.3 超声波接收电路	192
【能力拓展】	193
【项目考核】	193
附录	195
附录 A MCS-51 单片机指令系统	195
附录 B 单片机的伪指令	201
附录 C 特殊功能寄存器	202
附录 D ASCII 码	203
附录 E Easy 51pro V2.0 的使用	203
参考文献	207

项目 1 简易功能计算器

【学习目标】

- 1) 了解 C 语言的发展及其特点。
- 2) 掌握 C 语言的基本数据类型和常量及变量的概念及使用方法。
- 3) 会各种运算符的使用方法和不同数据类型间的转换方法。
- 4) 会编写基本顺序结构、分支结构和循环结构的应用程序。
- 5) 具有使用 VC++6.0 开发 C 语言程序的能力。
- 6) 具有 C 程序的设计、仿真和基本调试能力。
- 7) 会撰写标准的项目报告，并能将学习结果进行交流汇报。

【项目描述】

编程实现简易功能计算器的功能：在提示下选择运算类型加、减、乘、除和乘方、立方、开方 7 种计算功能，当选择 0 时退出；同时当进行除法运算时，如输入除数为 0，则要求重新输入除数再进行运算。

【项目实施】

任务 1 设计方案

通过输出函数 `printf()` 建立简易功能计算器的功能界面，再通过输入函数 `scanf()` 输入运算的类型，并根据输入的运算类型提示需要输入的数值个数并输入对应的数据。当输入“1”时，执行加法运算；当输入“2”时，执行减法运算；以此类推，当输入“7”时，执行开方运算；当输入“0”时，结束运算；输入其他符号或正常运算并显示出运算结果后，均返回运算类型的输入，接受下一次的运算。除法运算的程序流程图如图 1-1 所示。

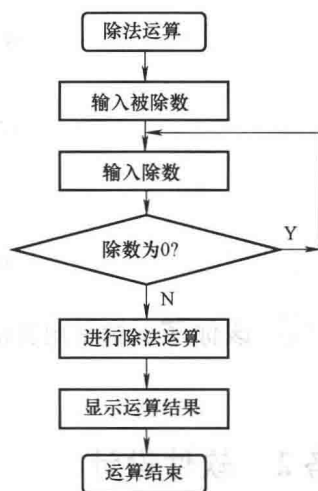


图 1-1 除法运算的程序流程图

 **该你了：**请分别绘制出加法、减法、乘法、除法、乘方、立方、开方运算的程序流程图。

在实现运算过程中注意运算的操作数个数的要求，除了加、减、乘、除四种运算需要两个操作数外，另外的乘方、立方、开方三种运算均只需一个操作数；另外在执行除法运算操作时，应考虑除数不能为 0 的特殊情况，应要求重新输入除数。其简易功能计算器的程序流程图如图 1-2 所示。

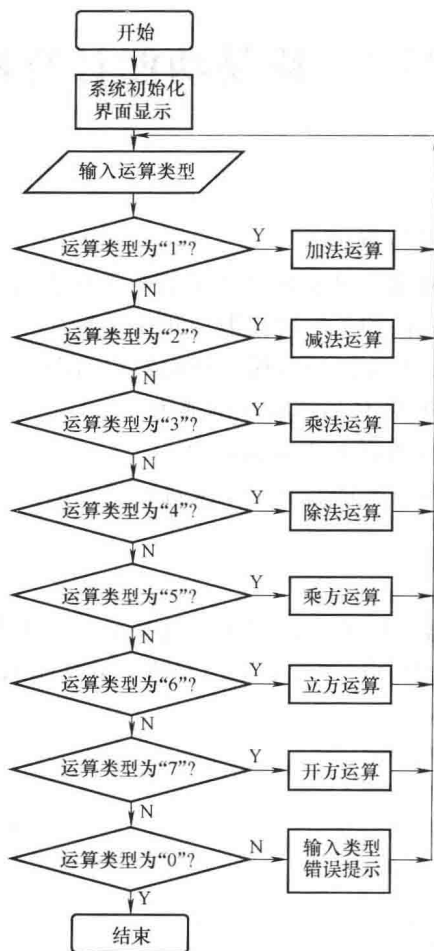


图 1-2 简易功能计算器的程序流程图



该你了：请试用其他结构形式绘制出简易功能计算器的程序流程图。

任务 2 软件设计

根据图 1-1 和图 1-2 的程序流程图，可以编写出如下的程序。

```

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
void main( )
{
    int fun;
    float num1,num2;
    system("cls");
    printf("\n\n");

```

```

printf( " \t\t | ----- | \n" );
printf( " \t\t |          宜宾职业技术学院          | \n" );
printf( " \t\t |          简易功能计算器          | \n" );
printf( " \t\t | ----- | \n" );
printf( " \t\t |          1-----加法          | \n" );
printf( " \t\t |          2-----减法          | \n" );
printf( " \t\t |          3-----乘法          | \n" );
printf( " \t\t |          4-----除法          | \n" );
printf( " \t\t |          5-----乘方          | \n" );
printf( " \t\t |          6-----立方          | \n" );
printf( " \t\t |          7-----开方          | \n" );
printf( " \t\t |          0-----退出          | \n" );
printf( " \t\t | ----- | \n" );
printf( " \t\t |          设计者:电子****   第 X 组   X X X          | \n" );
printf( " \t\t | ----- | \n" );
printf( " \t\t 请选择运算类型(0--7): ");
scanf( " %d" ,&fun);
while( fun)
{
    if( fun < = 4)
    {
        printf( " \t\t 请输入第一运算数: ");
        scanf( " %f" ,&num1);
        printf( " \t\t 请输入第二运算数: ");
        tt2:scanf( " %f" ,&num2);
        switch( fun)
        {
            case 1:printf( " \t\t 运算结果为:");
                    printf( " \n\t\t\t%f + %f = %f\n" , num1 , num2 , num1 + num2 );
                    break;
            case 2:printf( " \t\t 运算结果为:");
                    printf( " \n\t\t\t%f - %f = %f\n" , num1 , num2 , num1 - num2 );
                    break;
            case 3:printf( " \t\t 运算结果为:");
                    printf( " \n\t\t\t%f × %f = %f\n" , num1 , num2 , num1 * num2 );
                    break;
            case 4:if( num2 == 0)
                    {
                        printf( " \t\t 注意:除数不能为 0! 请重新输入一个数:");
                        goto tt2;
                    }
                    else
                    {
                        printf( " \t\t 运算结果为:");

```

```

        printf(" \n\t\t\t%f ÷ %f = %f\n", num1, num2, num1/num2);
        break;
    }

    case 0: exit(0);
        break;
    default: printf(" 输入选项错误! \n");
}

else if( fun > 7)
{
    printf(" \n\t\t 输入选项错误! \n");
    goto ddl;
}

else
{
    printf(" \t\t 请输入运算数: ");
    scanf("%f", &num1);
    printf(" \t\t 运算结果为: ");
    switch( fun)
    {
        case 5: printf(" \n\t\t\t%f 的平方 = %f\n", num1, num1 * num1); break;
        case 6: printf(" \n\t\t\t%f 的立方 = %f\n", num1, num1 * num1 * num1); break;
        case 7: printf(" \n\t\t\t%f 开方 = %f\n", num1, sqrt(num1)); break;
        default: printf(" \n\t\t 输入选项错误! \n");
    }
}

ddl: printf(" \n\t ***** \n");
    printf(" \t\t 请选择运算类型(0--7): ");
    scanf("%d", &fun);
}
}

```

任务3 系统仿真与调试

在 VC6 软件中, 将编写好的程序输入并以 *.C 的文件格式保存, 通过编译连接后生成可执行文件 (*.exe), 再运行对应生成的可执行文件, 即可出现简易功能计算器系统界面, 如图 1-3 所示。当选择运算类型为“1”时, 出现如图 1-4 所示的加法运算界面, 减法和乘方与加法运算类似; 当选择运算类型为“4”时, 出现如图 1-5 所示的除法运算界面; 当选择运算类型为“6”时, 出现如图 1-6 所示的立方运算界面, 开方和乘方运算也与立方运算类似; 当选择运算类型为“9”时, 由于输入类型错误出现如图 1-7 所示的结果; 当选择运算类型为“0”时, 出现如图 1-8 所示退出系统界面。

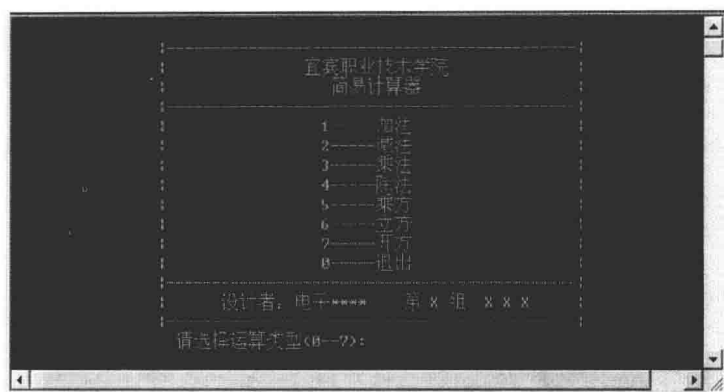


图 1-3 简易功能计算器系统界面



图 1-4 加法运算界面



图 1-5 除法运算界面



图 1-6 立方运算界面



图 1-7 输入类型错误结果

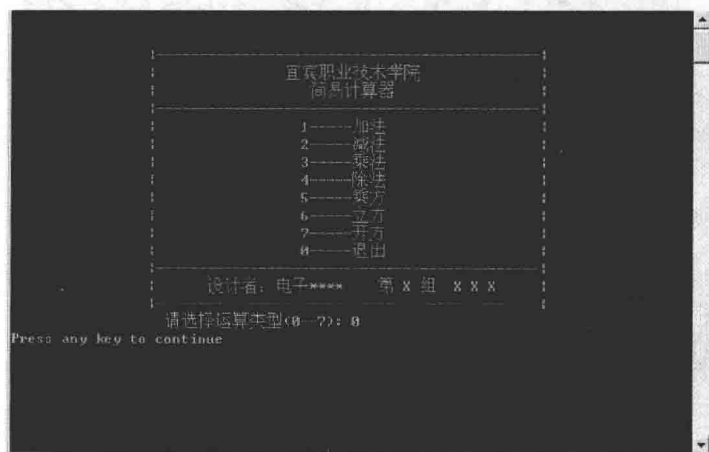


图 1-8 退出系统界面

【相关知识】

1.1 C 语言基础

在编写操作系统和系统使用程序或者需要对硬件进行操作的场合，用 C 语言明显优于其他高级语言。许多大型应用软件都是用 C 语言编写的。C 语言具有绘图能力强、可移植性好等特点，并具备很强的数据处理能力，因此适于编写系统软件。

1.1.1 C 语言的发展及特点

1. C 语言的发展

1963 年，剑桥大学将 ALGOL 60 语言发展成为 CPL 语言。

1967 年，剑桥大学的 Martin Richards 对 CPL 语言进行了简化，于是产生了 BCPL 语言。

1970 年，美国贝尔实验室的 Ken Thompson 将 BCPL 进行了修改，并为它起了一个有趣的名字“B 语言”。意思是将 CPL 语言煮干，提炼出它的精华。并且他用 B 语言写了第一个 UNIX 操作系统。

而在 1973 年，B 语言也被人“煮”了一下，美国贝尔实验室的 D. M. RITCHIE 在 B 语言的基础上最终设计出了一种新的语言，他取了 BCPL 的第二个字母作为这种语言的名字，这就是 C 语言。

为了推广 UNIX 操作系统，1977 年 Dennis M. Ritchie 发表了不依赖于具体机器系统的 C 语言编译文本——可移植的 C 语言编译程序。

1978 年 Brian W. Kernighan 和 Dennis M. Ritchie 出版了名著《The C Programming Language》，从而使 C 语言成为目前世界上流行最广泛的高级程序设计语言。

1988 年，随着微型计算机的日益普及出现了许多 C 语言版本。由于没有统一的标准，使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况，美国国家标准研究所 (ANSI) 为 C 语言制定了一套 ANSI 标准。C 语言发展迅速，而且成为最受欢迎的语言之一，主要因为它具有强大的功能。许多著名的系统软件，如 DBASE III PLUS、DBASE IV 都是用 C 语言编写的。用 C 语言加上一些汇编语言子程序，就更能显示 C 语言的优势了，像 PC-DOS、WORDSTAR 等就是用这种方法编写的。

2. C 语言的特点

1) 简洁紧凑、灵活方便。C 语言一共只有 32 个关键字、9 种控制语句，程序书写自由，主要用小写字母表示。它把高级语言的基本结构和语句与低级语言的实用性结合起来。C 语言可以像汇编语言一样对位、字节和地址进行操作，而这三者是计算机最基本的工作单元。

2) 运算符丰富。C 的运算符包含的范围很广泛，共有 34 个运算符。C 语言把括号、赋值、强制类型转换等都作为运算符处理，从而使 C 的运算类型极其丰富，表达式类型多种多样。灵活使用各种运算符可以实现在其他高级语言中难以实现的运算。

3) 数据结构丰富。C 的数据类型有：整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等，能用来实现各种复杂的数据类型的运算。并引入了指针概念，使程序效率更高。另外，C 语言具有强大的图形功能，支持多种显示器和驱动器，且计算功能、逻辑判断功能强大。

4) C 是结构式语言。结构式语言的显著特点是代码及数据的分隔化,即程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰,便于使用、维护及调试。C 语言是以函数形式提供给用户的,这些函数可方便地调用,并具有多种循环语句、条件语句控制程序流向,从而使程序完全结构化。

5) C 语法限制不太严格、程序设计自由度大。一般的高级语言语法检查比较严,能够检查出几乎所有的语法错误,而 C 语言允许程序编写者有较大的自由度。

6) C 语言允许直接访问物理地址,可以直接对硬件进行操作。因此 C 语言既具有高级语言的功能,又具有低级语言的许多功能,能够像汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元,所以 C 语言可以用来编写系统软件。

7) C 语言程序生成代码质量高,程序执行效率高,一般只比汇编程序生成的目标代码效率低 10% ~ 20%。

8) C 语言适用范围大,可移植性好。

C 语言有一个突出的优点就是适合于多种操作系统,如 DOS、UNIX,也适用于多种机型。

1.1.2 简单 C 程序介绍

【例 1.1】

```
void main ( )
```

```
{  
    printf ("This is a C program. \n");  
}
```

本程序的作用是在显示器显示以下一行信息:

This is a c program.

说明:

main 表示“主函数”,每一个 C 程序都必须有一个 main 函数;函数体由大括弧 { } 括起来;printf 是 C 语言中的输出函数;语句最后有一分号“;”。

【例 1.2】求 a、b 两数之和。

```
main ( ) /* 求 a、b 两数之和 */
```

```
{  
    int a,b,sum;  
    a = 123; b = 456;  
    sum = a + b;  
    printf ("sum is %d\n", sum);  
}
```

执行结果: sum is 579

本程序的作用是求两个整数 a 和 b 之和,结果在 sum。

/* */ 表示注释部分,可以用汉字或用英语作注释。

第 2 行是声明部分,定义变量 a 和 b 为整型 (int) 变量。

第 3 行是两个赋值语句,使 a 为 123, b 的值为 456。

第 4 行使 sum 的值为 a + b;

第 5 行 printf 函数中是输出。

【例 1.3】 从键盘输入两个整数，计算并输出它们的和。

```
#include <stdio.h>

/* add 函数用于求两数之和 */
int add(int x,int y)      /* 函数定义部分,add 为函数名,x,y 为形参 */
{
    int z;
    z = x + y;
    return(z);            //将两数之和返回到主调函数中
}

/* main( )函数完成两个整数的输入,并输出两数之和 */
main( )
{
    int a,b,sum;
    printf(" please input two number:");
    scanf("%d,%d",&a,&b);    //输入两个整数,分别放入变量 a,b 中
    sum = add(a,b);          //调用 add( ) 函数,将返回值赋给变量 sum
    printf(" sum = %d\n",sum);
}
```

程序运行结果:

please input two number:4,8

sum = 12

程序说明:

该程序由主函数 main()和被调函数 add()组成，它们各有自己的功能。main()函数中的 scanf()是系统提供的标准输入函数，其功能是输入 a 和 b 的值。

通过以上三个例子的分析，可以看出 C 语言源程序的基本结构有以下几个特点：

1) C 语言程序是由函数组成的，每个函数完成相对独立的功能，函数是 C 语言程序的基本模块单元。每个程序必须有一个且只有一个主函数 main()，除主函数之外，可以没有其他函数（如【例 1.1】和【例 1.2】），也可以有一个或多个其他函数（如【例 1.3】），被调用的函数可以是系统提供的函数（如 printf()和 scanf()），也可以是用户根据需要自己编写的函数（如 add()）。

2) 主函数的位置是任意的，可以在源程序的开头、两个函数之间或源程序的结尾。程序的执行总是从主函数开始，并在主函数结束。

3) C 语言源程序一般用小写字母书写，只有符号常量或其他特殊用途的符号才使用大写字母。

4) C 程序的书写格式自由，允许一行写多个语句，也允许一个语句写在多行，但所有的语句都必须以分号结束。如果某条语句很长，一般需要将其分成多行书写。

5) 可以用“/*……*/”对 C 程序的任何部分作注释，以增强源程序的可读性。VC++ 中还可以用“//”给程序加注释，两者的区别在于“/*……*/”可以对多行进行注释，而“//”只能对单行进行注释。源程序编译时，不对注释作任何处理。注释通常放在一段程序的开头用以说明该段程序的功能；或者放在某个语句的后面，对该语句进行说明。在使用“/*……*/”加注释时，需要注意“/*”和“*/”必须成对使用，其“/”和“*”

以及“*”和“/”之间不能有空格,否则程序会出错。为了避免错误,常用“//”进行注释。

1.1.3 C 程序的开发过程

开发一个 C 语言程序,一般要在一个集成开发环境中进行程序的编辑、编译、连接和运行四个步骤,C 语言程序开发过程如图 1-9 所示。

(1) 源文件的编辑 用户通过编辑器,将自己开发的 C 语言程序输入计算机的过程称为 C 程序源文件的编辑。编辑生成的文件以文本形式存储,扩展名为“.C”,也称为 C 的源程序。

源程序文件以 ASCII 码形式存储,不含任何格式和样式,计算机不能直接执行。

(2) 编译 计算机把 C 的源程序翻译成计算机可以识别的二进制形式的目标代码文件,这个过程称为编译,由 C 的编译程序完成。

C 的编译程序在编译的同时,还对源程序的语法和程序的逻辑结构等进行检查。当发现错误时,将会列出错误的位置和种类,此时需要重新编辑修改源程序。如果编辑成功,则生成目标文件,文件名同源程序文件名,扩展名为“.obj”。

编译生成的目标文件,不包含程序运行所需要的库函数等,计算机仍然不能直接执行。

(3) 连接 将目标程序和其他目标程序模块,以及系统提供的 C 库函数等进行连接生成可执行文件的过程,称为“连接”。连接生成的可执行文件的文件名同源程序文件名,扩展名为“.exe”。连接生成的可执行文件,计算机可以直接执行。

(4) 运行 在 DOS 环境中直接键入 C 程序的可执行文件名,或者在 C 的集成环境下选择“RUN”命令,以及在 Windows 的资源管理器内双击该可执行文件名,都可以获得运行结果。如果运行结果有误,需要重新编辑源程序,再进行编译、连接、运行,直到得到满意的运行结果。

1.1.4 VC++6.0 集成开发环境

集成开发环境是一个综合性的工具软件,它把程序设计全过程所需的各项功能有机地结合起来,统一在一个图形化操作界面下,为程序设计人员提供尽可能高效、便利的服务。

VC++6.0 就是一个功能齐全的集成开发环境,虽然它常常用来编写 C++ 源程序,但

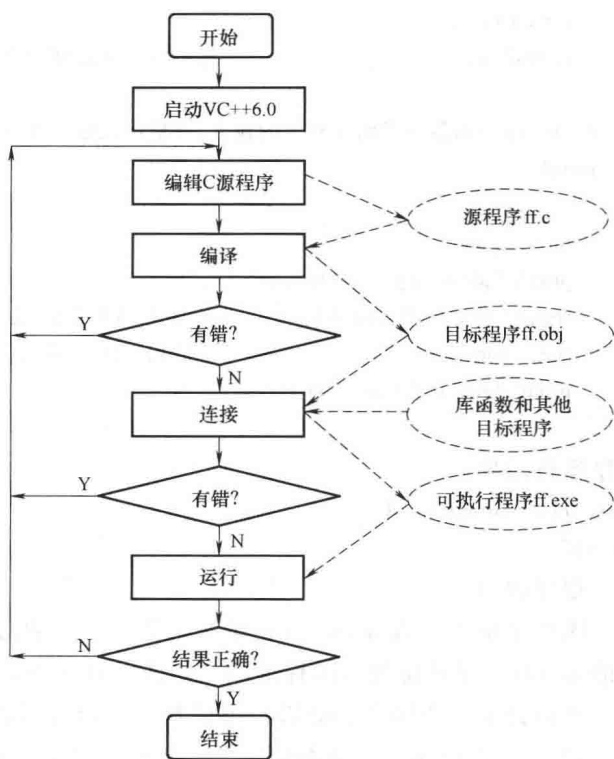


图 1-9 C 语言程序开发过程