



国家示范性(骨干)高职院校建设项目成果  
高等职业教育教学改革系列规划教材

# C语言程序设计项目教程 (基于Keil C)

李建兰◎编著

- 以Keil软件作为开发平台,突出C语言在工程控制中的重要性
- 注重培养学生的编程思路和调试能力,弱化语法和算法要求
- 通过41个小任务和11个工程应用,介绍C语言在工程实践中的应用
- 配有免费的电子教学课件和源程序,以方便教师教学



中国工信出版集团



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

国家示范性（骨干）高职院校建设项目成果  
高等职业教育教学改革系列规划教材

# C 语言程序设计项目教程 (基于 Keil C)

李建兰 编著

邵建龙 主审

电子工业出版社  
Publishing House of Electronics Industry

北京·BEIJING

## 内容简介

本书以 Keil 软件为编译平台,以 C 语言在工程实践中的具体应用为主线,采用项目驱动形式,将 C 语言语法和编程技巧等知识通过项目任务和工程应用的方式传授给学生,打破了传统的教学方法和教学体系结构,解决了 C 语言程序设计这门课程抽象与枯燥难学的问题。

全书共分 8 个项目:项目 1 认识 C 语言、项目 2 顺序结构程序设计、项目 3 选择结构程序设计、项目 4 循环结构程序设计、项目 5 数组及应用、项目 6 函数及应用、项目 7 指针及应用、项目 8 C 语言综合程序设计。

本书可作为高职高专学校电气、电子、信息类相关专业的教材,也可供 C 语言初学者、电子爱好者和大专及职业高中等相关专业师生学习和实践参考。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有,侵权必究。

### 图书在版编目(CIP)数据

C 语言程序设计项目教程:基于 Keil C/ 李建兰编著. —北京:电子工业出版社, 2017.5  
ISBN 978-7-121-31526-8

I. ①C… II. ①李… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 108516 号

策划编辑:王艳萍

责任编辑:王艳萍

印刷:北京嘉恒彩色印刷有限责任公司

装订:北京嘉恒彩色印刷有限责任公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开本:787×1092 1/16 印张:11.75 字数:300.8 千字

版次:2017 年 5 月第 1 版

印次:2017 年 5 月第 1 次印刷

印数:3 000 册 定价:31.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888, 88258888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式:wangyp@phei.com.cn。

# 前 言

C 语言是一门工具语言，是目前国内外广泛使用的一种计算机高级语言，是现代大学生步入智能化、信息化社会，迈向电子工程师和程序员成功之路的一块敲门砖。

本书根据教育部高等教育人才培养的指导思想，结合高职高专院校学生及计算机高级语言特点，采用项目驱动形式，循序渐进地讲述 C 语言的语法知识、编程技巧和调试方法，适合电气、电子、信息类应用型人才的相关专业的学生学习。

本书以 C 语言在工程中的实际应用为主线，围绕项目任务展开教学，是编者多年来 C 语言课程教学改革的成果与经验总结。书中精选大量具有代表性的项目任务和工程应用实例，使读者既能掌握 C 语言的基本概念、基本知识和调试技能，又能拓宽 C 语言的编程思路和应用领域。突出培养学生运用所学知识和技能解决实际问题的能力，为其后续课程（如“单片机技术”）的学习和职业生涯打下良好的基础。该书具有如下特点。

(1) 用 Keil 软件作为开发平台，用仿真实验板作为调试工具，具有独创性，更加突出 C 语言在工程控制中的重要性。

Keil 软件是目前单片机工程应用中广泛使用的集成开发软件，它提供丰富的库函数和功能强大的集成开发调试工具，同时具有灵活多样的仿真功能。在 Keil 软件和仿真实验板上调试和仿真 C 程序能缩短计算机语言与工程实际问题间的距离，更加突出 C 语言在工程控制中的重要性。

(2) 以 C 语言在工程实践中的具体应用为主线，采用项目任务驱动法编写。

全书分为 8 个项目，包括 41 个具体任务和 11 个工程应用。每个任务都按照“项目任务—相关知识—任务实现—工程应用—归纳与总结”进行，每个工程应用都围绕“任务描述—编写 C 程序—上机调试与仿真”展开，教学安排符合程序设计类课程教学规律。不是照本宣科地去讲授知识，而是通过上机调试与相关知识相结合的方式，将 C 语言的语法知识、编程技巧与调试方法穿插在各个项目任务中进行传授，融“教、学、做”于一体，这样的教学有趣而生动。各任务完成后使学生知道这些知识具体用在哪、如何用，真正提高了学生动手能力和解决实际问题的能力。

(3) 打破传统的 C 语言知识体系结构，强调建立工程控制观点，从工程控制的角度重构课程内容，突出了知识在工程中的实效性。

不在语法和算法上对学生提出过高要求，而是注重编程思路和调试能力，注重如何开展实际应用，如何解决实际问题。内容的重点放在工程实践中的循环控制、位运算、逻辑关系等共性的知识上，将这些知识以不同的形式穿插在多个项目的任务中反复训练，逐步提高学生的编程技能和调试能力。同时增强了知识的融合性和灵活性，更突出了这些知识在工程中的实效性。

(4) 在工程项目调试过程中，引入仿真技术，直观、生动、灵活、有趣，增强了学生的求知欲，同时也激发了学生的学习热情。

在工程项目应用中，引入实验仿真板，程序运行结果直观、形象、生动、有趣，使学生懂得 C 语言在工程实践中的应用领域和广阔的发展前景，拓展学生的知识面，激发学生的求知欲和学习热情。

(5) 书中所有练习题都是在各个项目任务的基础上进行的进一步的拓展功能，教师容易掌控，学生容易上手。真正达到练习的目的，也以此加深和巩固所学知识。

全书由云南机电职业技术学院李建兰编著，昆明理工大学信息工程与自动化学院邵建龙主审。

为方便教学，本书配有免费的电子教学课件和程序源代码，相关教学资源请登录华信教育资源网（[www.hxedu.com.cn](http://www.hxedu.com.cn)）免费注册后下载。希望这本书能对读者学习和掌握 C 语言有所帮助。由于编者水平有限，书中难免有错误和不妥之处，敬请广大读者批评指正。

编者：李建兰  
主审：邵建龙  
编者：李建兰  
主审：邵建龙

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

本书在编写过程中，参考了以下文献资料，在此致以诚挚的谢意：  
① 清华大学出版社，2008 年。  
② 清华大学出版社，2008 年。  
③ 清华大学出版社，2008 年。  
④ 清华大学出版社，2008 年。  
⑤ 清华大学出版社，2008 年。  
⑥ 清华大学出版社，2008 年。  
⑦ 清华大学出版社，2008 年。  
⑧ 清华大学出版社，2008 年。  
⑨ 清华大学出版社，2008 年。  
⑩ 清华大学出版社，2008 年。

# 目 录

|                            |      |
|----------------------------|------|
| 项目 1 认识 C 语言               | (1)  |
| 项目任务                       | (1)  |
| 相关知识                       | (1)  |
| 1.1 C 语言的发展                | (1)  |
| 1.2 C 语言的特点                | (2)  |
| 1.3 C 语言的程序结构              | (3)  |
| 1.3.1 一个简单的 C 语言程序         | (3)  |
| 1.3.2 源程序的另外一种书写形式         | (8)  |
| 1.3.3 C 语言的组成部分            | (8)  |
| 1.4 Keil 开发软件              | (9)  |
| 1.5 任务实现                   | (15) |
| 任务 1.1: 认识第一个 C 程序         | (15) |
| 任务 1.2: 用 Keil 软件搭建第一个工程项目 | (16) |
| 归纳与总结                      | (17) |
| 练习题                        | (17) |
| 项目 2 顺序结构程序设计              | (18) |
| 项目任务                       | (18) |
| 相关知识                       | (18) |
| 2.1 标识符与关键字                | (19) |
| 2.1.1 标识符                  | (19) |
| 2.1.2 关键字                  | (19) |
| 2.2 常量与变量                  | (21) |
| 2.2.1 常量                   | (21) |
| 2.2.2 变量                   | (22) |
| 2.3 数据类型                   | (23) |
| 2.3.1 常见数据类型               | (23) |
| 2.3.2 用 typedef 重新定义数据类型   | (25) |
| 2.4 运算符与表达式                | (25) |
| 2.4.1 赋值运算符及表达式            | (25) |
| 2.4.2 算术运算符及表达式            | (26) |
| 2.4.3 关系运算符及表达式            | (27) |
| 2.4.4 逻辑运算符及表达式            | (28) |
| 2.4.5 条件运算符                | (30) |
| 2.4.6 位运算符                 | (30) |

|             |                   |      |
|-------------|-------------------|------|
| 2.4.7       | 复合赋值运算符           | (30) |
| 2.4.8       | 逗号运算符             | (31) |
| 2.4.9       | 求字节数运算符 sizeof    | (31) |
| 2.5         | 复合语句              | (32) |
| 2.6         | 数据的输入/输出语句        | (32) |
| 2.6.1       | printf()函数及用法     | (32) |
| 2.6.2       | scanf()函数及用法      | (33) |
| 2.6.3       | putchar()函数及用法    | (34) |
| 2.6.4       | getchar()函数及用法    | (34) |
| 2.7         | 预处理命令             | (35) |
| 2.7.1       | 宏定义               | (35) |
| 2.7.2       | 文件包含处理            | (35) |
| 2.7.3       | 条件编译              | (36) |
| 2.8         | 任务实现              | (37) |
| 任务 2.1:     | 求两整数之和            | (37) |
| 任务 2.2:     | 给定一个大写字母,用小写字母输出  | (39) |
| 任务 2.3:     | 在电脑屏上输出图案         | (40) |
| 2.9         | 工程应用——点亮一个发光二极管   | (41) |
|             | 归纳与总结             | (43) |
|             | 练习题               | (43) |
| <b>项目 3</b> | <b>选择结构程序设计</b>   | (44) |
| 项目任务        |                   | (44) |
| 相关知识        |                   | (44) |
| 3.1         | if 语句及用法          | (45) |
| 3.2         | switch 语句及用法      | (47) |
| 3.3         | 任务实现              | (48) |
| 任务 3.1:     | 比较大小              | (48) |
| 任务 3.2:     | 两个数排序             | (49) |
| 任务 3.3:     | 三个数排序             | (50) |
| 任务 3.4:     | 大、小写字母转换          | (51) |
| 任务 3.5:     | 比较大小并进行算数运算       | (52) |
| 任务 3.6:     | 判断星期并显示           | (53) |
| 任务 3.7:     | 百分制分数转换为成绩等级      | (54) |
| 任务 3.8:     | 输入字符并进行算数运算       | (55) |
| 任务 3.9:     | 闰年判断              | (56) |
| 3.4         | 工程应用——开关控制发光二极管点亮 | (58) |
|             | 归纳与总结             | (60) |
|             | 练习题               | (60) |

|                          |      |
|--------------------------|------|
| 项目 4 循环结构程序设计            | (61) |
| 项目任务                     | (61) |
| 相关知识                     | (61) |
| 4.1 while 语句及用法          | (62) |
| 4.2 do-while 语句          | (63) |
| 4.3 for 语句及用法            | (63) |
| 4.4 break、continue 语句及用法 | (65) |
| 4.5 循环的嵌套                | (65) |
| 4.6 任务实现                 | (66) |
| 任务 4.1: 用循环语句求和          | (66) |
| 任务 4.2: 用循环语句排序          | (68) |
| 任务 4.3: 用循环语句进行大、小写字母转换  | (69) |
| 任务 4.4: 输出被 5 整除的数       | (70) |
| 任务 4.5: 素数判断             | (71) |
| 任务 4.6: 用循环语句进行闰年判断      | (72) |
| 任务 4.7: 输出矩阵             | (73) |
| 4.7 工程应用——LED 灯闪烁        | (74) |
| 归纳与总结                    | (76) |
| 练习题                      | (76) |
| 项目 5 数组及应用               | (77) |
| 项目任务                     | (77) |
| 相关知识                     | (77) |
| 5.1 一维数组及应用              | (77) |
| 5.2 二维数组及应用              | (78) |
| 5.3 字符数组及应用              | (79) |
| 5.4 任务实现                 | (80) |
| 任务 5.1: 冒泡法排序            | (80) |
| 任务 5.2: 定义矩阵找最大数         | (82) |
| 任务 5.3: 输出字符串            | (84) |
| 任务 5.4: 输出菱形图案           | (86) |
| 5.5 工程应用——霓虹灯控制 (1)      | (88) |
| 归纳与总结                    | (90) |
| 练习题                      | (91) |
| 项目 6 函数及应用               | (92) |
| 项目任务                     | (92) |
| 相关知识                     | (93) |
| 6.1 函数定义、调用及声明           | (93) |
| 6.1.1 函数分类               | (93) |
| 6.1.2 函数定义               | (93) |



|             |                        |       |
|-------------|------------------------|-------|
| 6.1.3       | 函数调用                   | (95)  |
| 6.1.4       | 函数声明                   | (95)  |
| 6.2         | 函数的参数和函数的值             | (96)  |
| 6.2.1       | 形式参数和实际参数              | (96)  |
| 6.2.2       | 函数的返回值                 | (96)  |
| 6.3         | 函数的嵌套调用                | (97)  |
| 6.3.1       | 函数的嵌套调用                | (97)  |
| 6.3.2       | 数组名作为函数参数              | (97)  |
| 6.4         | 局部变量和全局变量              | (98)  |
| 6.4.1       | 局部变量                   | (98)  |
| 6.4.2       | 全局变量                   | (98)  |
| 6.5         | 变量的存储类型                | (98)  |
| 6.6         | 任务实现                   | (100) |
| 任务 6.1:     | 用函数调用实现信息显示            | (100) |
| 任务 6.2:     | 用函数调用实现比较两数大小          | (102) |
| 任务 6.3:     | 用函数调用实现求两整数和           | (103) |
| 任务 6.4:     | 用函数嵌套调用实现找五个整数中的最大和最小数 | (104) |
| 任务 6.5:     | 用函数调用实现闰年判断            | (107) |
| 任务 6.6:     | 用函数调用实现排序              | (108) |
| 任务 6.7:     | 用函数调用实现求 10 个整数平均值     | (110) |
| 任务 6.8:     | 用函数调用实现素数判断            | (111) |
| 任务 6.9:     | 用函数调用实现输出九九乘法表         | (113) |
| 任务 6.10:    | 输出 1 到 5 的阶乘值          | (115) |
| 任务 6.11:    | 多文件形式实现学生分数等级判断        | (116) |
| 6.7         | 工程应用——霓虹灯控制 (2)        | (118) |
|             | 归纳与总结                  | (120) |
|             | 练习题                    | (120) |
| <b>项目 7</b> | <b>指针及应用</b>           | (122) |
|             | 项目任务                   | (122) |
|             | 相关知识                   | (122) |
| 7.1         | 指针及指针变量                | (122) |
| 7.2         | 指针运算符                  | (123) |
| 7.3         | 变量的访问方式                | (124) |
| 7.4         | 指针运算                   | (124) |
| 7.5         | 任务实现                   | (125) |
| 任务 7.1:     | 用指针法实现大、小写字母转换         | (125) |
| 任务 7.2:     | 用指针法实现闰年判断             | (126) |
| 任务 7.3:     | 用指针法实现求两个整数中的较大数       | (127) |
| 任务 7.4:     | 用指针法实现两个整数按由小到大的顺序排序   | (128) |

|  |              |
|--|--------------|
| 任务 7.5: 用指针法实现 10 个整数按由小到大的顺序排序 .....  | (129)        |
| 7.6 工程应用——霓虹灯控制 (3) .....              | (130)        |
| 归纳与总结 .....                            | (132)        |
| 练习题 .....                              | (132)        |
| <b>项目 8 C 语言综合程序设计 .....</b>           | <b>(133)</b> |
| 8.1 图形输出速度控制 .....                     | (133)        |
| 8.2 模拟汽车转弯灯控制 .....                    | (136)        |
| 8.3 一键多功能控制 .....                      | (139)        |
| 8.4 模拟交通灯 10 秒倒计时控制 .....              | (142)        |
| 8.5 LED 电子广告牌控制 .....                  | (145)        |
| 归纳与总结 .....                            | (148)        |
| 练习题 .....                              | (148)        |
| <b>附录 A 仿真实验板简介 .....</b>              | <b>(150)</b> |
| <b>附录 B 多文件 C 程序使用方法 .....</b>         | <b>(155)</b> |
| <b>附录 C C 库函数 .....</b>                | <b>(159)</b> |
| C.1 寄存器库函数 REG $\times\times$ .H ..... | (159)        |
| C.2 字符函数 CTYPE.H .....                 | (159)        |
| C.3 一般输入/输出函数 STDIO.H .....            | (161)        |
| C.4 内部函数 INTRINS.H .....               | (162)        |
| C.5 标准函数 STDLIB.H .....                | (163)        |
| C.6 字符串函数 STRING.H .....               | (164)        |
| C.7 数学函数 MATH.H .....                  | (166)        |
| C.8 绝对地址访问函数 A8SACC.H .....            | (167)        |
| <b>附录 D ASCII 码表 .....</b>             | <b>(169)</b> |
| <b>附录 E C 语言经典错误 .....</b>             | <b>(170)</b> |
| <b>参考文献 .....</b>                      | <b>(175)</b> |

## 1.1 C 语言的发展

C 语言诞生于 20 世纪 70 年代初, 1978 年由美国电话电报公司 (AT&T) 贝尔实验室正式发布, 同时由 B.W.Kernighan 和 D.M.Ritchie 合著了著名的《THE C PROGRAMMING LANGUAGE》一书, 通常简称为《K&R》, 也有人称之为《K&R》标准。但是, 在《K&R》中并没有定义一个完整的标准 C 语言, 后来由美国国家标准协会 (American National Standards Institute) 在此基础上制定了一个 C 语言标准, 于 1983 年发布, 通常称之为 ANSI C。

早期的 C 语言主要用于 UNIX 系统, 由于 C 语言的强大功能和各方面的优点逐渐为人所认识, 到了 20 世纪 80 年代, C 语言开始进入其他操作系统, 并很快在各类大、中、小型计算机上得到广泛的使用, 成为当代最优秀的程序设计语言之一。



# 项目 1 认识 C 语言

## 教学目的

- 了解 C 语言的发展历史及特点；
- 初步认识和掌握 C 语言的程序结构；
- 了解 C 语言程序开发软件；
- 熟练掌握 Keil 软件的使用。

## 重点和难点

- C 语言的特点；
- C 语言的程序结构；
- Keil 软件的使用；
- 搭建第一个工程项目。



## 项目任务

任务 1.1: 认识第一个 C 程序

任务 1.2: 用 Keil 软件搭建第一个工程项目



## 相关知识

要使用 C 语言编程，必须先了解 C 语言，并能熟练掌握 C 语言语法知识和 C 程序调试方法。本书将通过项目任务形式，由浅入深地介绍与 C 语言相关的语法知识和调试方法，使读者能熟练掌握如何用 C 语言编写程序以及如何上机调试 C 程序。

### 1.1 C 语言的发展

C 语言问世于 20 世纪 70 年代初，1978 年由美国电话电报公司（AT&T）贝尔实验室正式发布。同时由 B.W.Kernighan 和 D.M.Ritchie 合著了著名的《THE C PROGRAMMING LANGUAGE》一书，通常简称为《K&R》，也有人称之为《K&R》标准。但是，在《K&R》中并没有定义一个完整的标准 C 语言，后来由美国国家标准协会（American National Standards Institute）在此基础上制定了一个 C 语言标准，于 1983 年发布，通常称之为 ANSI C。

早期的 C 语言主要用于 UNIX 系统。由于 C 语言的强大功能和各方面的优点逐渐为人们认识，到了 20 世纪 80 年代，C 语言开始进入其他操作系统，并很快在各类大、中、小和微型计算机上得到广泛的使用，成为当代最优秀的程序设计语言之一。



C 语言功能强,使用灵活,编写自由、可移植性好,兼有高级语言和低级语言的优点,既适合编写应用程序,又适合开发系统软件,是国内外广泛流行的计算机高级语言。其实就 C 语言的特点来看,C 语言更适合于解决某些小型程序的编程。C 语言作为传统的面向过程的程序设计语言,在编写底层的设备驱动程序和内嵌应用程序时,往往是更好的选择。因此,C 语言在单片机嵌入式技术领域得到了广泛的应用,现在越来越多的工程技术开发人员使用 C 语言,用于 51 单片机的 C 语言就称为 C51 语言。目前,单片机应用系统开发中广泛使用 C 语言,在大型、复杂的单片机应用系统开发中都通过 C 语言来设计程序。

## 1.2 C 语言的特点

与其他计算机高级语言相比,C 语言具有它自身的特点。我们可以用 C 语言来编写科学计算或其他应用程序,但 C 语言更适合于编写计算机的操作系统程序以及其他一些需要对机器硬件进行操作的场合,有的大型应用软件也采用 C 语言进行编写。这主要是因为 C 语言具有很好的可移植性和硬件控制能力,表达和运算能力也较强。许多以前只能用汇编语言来解决的问题现在可以改用 C 语言来解决。概括起来说 C 语言具有以下一些特点。

### 1. 语言简洁、紧凑,使用方便、灵活

C 语言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,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。

C 语言只有 32 个关键字,9 种控制语句,程序书写形式自由,源程序短。

### 2. 运算符丰富

C 语言的运算符包含的范围很广泛,共有 34 种运算符。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 语言的运算类型极其丰富,表达式类型多样化。

### 3. 数据结构丰富

C 语言具有现代化语言的各种数据结构,包括整型、浮点型、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结构的运算,并引入了指针概念,使程序效率更高。

### 4. 具有结构化的控制语句,可进行结构化程序设计

C 语言是完全模块化和结构化的语言,具有结构化的控制语句,如 if-else 语句、while 语句、do-while 语句、switch 语句、for 语句等。C 语言用函数作为程序的模块单位,便于实现程序的模块化。

### 5. 允许直接访问物理地址,对硬件进行操作

C 语言允许直接访问物理地址,能进行位操作,可以直接对硬件进行操作,因此 C 语言具有高级语言的功能和低级语言的许多功能,可用来编写系统软件。这种双重性,使它既是成功的系统描述语言,又是通用的程序设计语言。



## 6. 生成的目标代码质量高，程序执行效率高

C 语言描述问题比汇编语言迅速，工作量小、可读性好，易于调试、修改和移植，而代码质量与汇编语言相当。

## 7. 可移植性好，易于实现模块化设计

C 语言在不同机器上的编译程序，大多数代码是公共的，所以 C 语言的编译程序便于移植。在一个环境上用 C 语言编写的程序，不改动或稍加改动，就可移植到另一个完全不同的环境中运行。

## 8. 语法限制不严格，程序编写自由

一般的高级语言语法检查比较严格，能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，因此放宽了语法检查。如用户编写的用户自定义函数，位置可以放在主调函数的前面，也可以放在主调函数的后面，同时函数与函数间是平行的，互相独立的，可以放在程序的任意位置；又如整型数据与字符型数据以及逻辑型数据可以通用。这些灵活性必然给程序的编写带来极大的方便，使程序的编写自由空间大，且快速高效。

# 1.3 C 语言的程序结构

## 1.3.1 一个简单的 C 语言程序

C 语言是一种结构化程序设计语言，程序采用函数结构。

每个 C 语言程序由一个或多个函数组成，在这些函数中至少应包含一个主函数 `main()`，也可以包含一个 `main()` 函数和若干个其他的功能函数。不管 `main()` 函数放于何处，程序总是从 `main()` 函数开始执行，再到 `main()` 函数结束的。在 `main()` 函数中调用其他函数，其他函数也可以相互调用，但 `main()` 函数只能调用其他的功能函数，而不能被其他的函数所调用。功能函数可以是 C 语言编译器提供的库函数，也可以是由用户编写的自定义函数。

在编写 C 程序时，程序的开始部分一般是预处理命令，接下来是函数说明和变量定义，再往下就是对函数的定义等。一个简单 C 程序的流程结构框图如图 1-1 所示。

为了让读者能更好地理解简单的 C 程序，现在对任务 1.1 源程序代码做一些程序说明。

### 参考程序段 1:

```

/*****
    *@File      chapter 1-1.c
    *@Function  在电脑屏幕上显示一行信息
    *****/
#include <reg51.h>                                     //预处理命令

```



图 1-1 C 程序流程结构框图



```

#include <stdio.h>
void main()
{
    SCON=0x52;
    TMOD=0x20;
    TH1=0xf3;
    TR1=1;
    printf("Hello! My name is C program.\n");
    while(1);
}

```

//定义主函数  
//函数开始的标志  
//串口初始化, 打开显示窗口  
//输出所指定的信息  
//空循环, 程序暂停  
//函数结束的标志

### (1) #include <stdio.h>的作用

C 语言本身不提供输入和输出语句, 输入和输出操作是由函数来实现的。在 C 语言的标准函数库中提供了一个名为“stdio.h”的头文件, 它当中定义了 C 语言中的输入和输出函数, 如 printf() 函数。当使用输入和输出函数时, 需要先用预处理命令将该函数库头文件包含到源文件中。

在代码中加入头文件有两种书写方法, 分别为“#include <stdio.h>”和“#include "stdio.h"”。

### (2) printf()函数的作用

printf() 函数称为格式输出函数, 其功能是按用户指定的格式, 把指定的数据或字符显示输出。该函数是 C 语言提供的标准输出函数, 定义在 C 语言的标准函数库 (stdio.h) 中。

#### printf()函数的一般形式为

```
printf("格式控制字符串", 输出列表);
```

格式控制字符串可由格式字符串和非格式字符串组成。

格式控制字符串是以“%”开头的字符串, 输出列表在格式输出时才用到, 它给出了各个输出项, 要求与格式字符串在数量和类型上一一对应。相关知识将在后续内容里介绍。

非格式字符串在输出时原样输出, 在显示中起提示作用。本例中用到的就是非格式字符串。“\n”是一个转义字符, 其作用是换行。

### (3) 串口初始化作用

C 语言的一般 I/O 函数库中定义的 I/O 函数都是通过串行接口实现的, 串行口的波特率由 MCS-51 单片机中的定时器/计数器 1 溢出率决定。在使用 I/O 函数之前, 应先对 MCS-51 单片机的串行接口和定时器/计数器 1 进行初始化。串行口工作于方式 1, 定时器/计数器 1 工作于方式 2 (8 位自动重载方式), 设系统时钟为 12MHz, 波特率为 2400bps, 则初始化程序如下:

```

SCON=0x52;
TMOD=0x20;
TH1=0xf3;
TR1=1;

```

有关串行口的详细内容将在后续课程中再做介绍, 读者在这里只要会用串行口初始化程序即可, 其目的是打开串行口窗口, 提供程序输出显示界面。



#### (4) #include <reg51.h>作用及内容

由于在串行口初始化中用到了 SCON、TMOD 等几个特殊功能寄存器，使用它们时必须事先加以定义。Keil 编译器对这些特殊功能寄存器的定义都是放在一个名为 reg51.h 或 reg52.h 头文件里的，所以程序中需要先用预处理命令“#include <reg51.h>”将 51 单片机的特殊功能寄存器定义包含进来，这样使用 51 单片机的特殊功能寄存器才是合法的，否则编译器会报错。

打开 reg51.h 头文件可以看到这样一些内容：

```
/*-----  
REG51.H  
  
Header file for generic 80C 语言 and 80C31 microcontroller.  
Copyright (c) 1988-2002 Keil Elektronik GmbH and Keil Software, Inc.  
All rights reserved.  
-----*/  
  
#ifndef __REG51_H__  
#define __REG51_H__  
  
/* BYTE Register */  
sfr P0    = 0x80;  
sfr P1    = 0x90;  
sfr P2    = 0xA0;  
sfr P3    = 0xB0;  
sfr PSW   = 0xD0;  
sfr ACC   = 0xE0;  
sfr B     = 0xF0;  
sfr SP    = 0x81;  
sfr DPL   = 0x82;  
sfr DPH   = 0x83;  
sfr PCON = 0x87;  
sfr TCON = 0x88;  
sfr TMOD = 0x89;  
sfr TL0   = 0x8A;  
sfr TL1   = 0x8B;  
sfr TH0   = 0x8C;  
sfr TH1   = 0x8D;  
sfr IE    = 0xA8;  
sfr IP    = 0xB8;  
sfr SCON = 0x98;  
sfr SBUF = 0x99;  
  
/* BIT Register */  
/* PSW */  
sbit CY   = 0xD7;  
sbit AC   = 0xD6;  
sbit F0   = 0xD5;
```



```
sbit RS1 = 0xD4;
sbit RS0 = 0xD3;
sbit OV = 0xD2;
sbit P = 0xD0;

/* TCON */
sbit TF1 = 0x8F;
sbit TR1 = 0x8E;
sbit TF0 = 0x8D;
sbit TR0 = 0x8C;
sbit IE1 = 0x8B;
sbit IT1 = 0x8A;
sbit IE0 = 0x89;
sbit IT0 = 0x88;

/* IE */
sbit EA = 0xAF;
sbit ES = 0xAC;
sbit ET1 = 0xAB;
sbit EX1 = 0xAA;
sbit ET0 = 0xA9;
sbit EX0 = 0xA8;

/* IP */
sbit PS = 0xBC;
sbit PT1 = 0xBB;
sbit PX1 = 0xBA;
sbit PT0 = 0xB9;
sbit PX0 = 0xB8;

/* P3 */
sbit RD = 0xB7;
sbit WR = 0xB6;
sbit T1 = 0xB5;
sbit T0 = 0xB4;
sbit INT1 = 0xB3;
sbit INT0 = 0xB2;
sbit TXD = 0xB1;
sbit RXD = 0xB0;

/* SCON */
sbit SM0 = 0x9F;
sbit SM1 = 0x9E;
sbit SM2 = 0x9D;
sbit REN = 0x9C;
sbit TB8 = 0x9B;
```





```
sbit RB8 = 0x9A;
sbit TI = 0x99;
sbit RI = 0x98;

#endif
```

以上内容涉及 51 单片机的内部结构，现在读者还不熟悉，没有关系。其实这里都是一些符号的定义，即规定符号名与地址的对应关系。

### ① 特殊功能寄存器定义。

例如：

```
sfr P1 = 0x90;
```

定义 P1 与地址 0x90 对应，P1 口的地址就是 0x90 (0x90 是 C 语言中十六进制数的写法，相当于汇编语言中的 90H)。

从上面的头文件中读者可以看到一个频繁出现的词：**sfr**。sfr 并不是标准 C 语言的关键字，而是 Keil 软件里为能直接访问单片机中的 SFR 而提供的一个新的关键词，其用法如下：

```
sfr 变量名=地址值;
```

### ② 特殊位定义。

例如，符号 P1\_0 用来表示 P1.0 引脚。

```
sbit P1_0=P1^0;
```

在 C 语言里，如果直接写 P1.0，C 编译器并不能识别，而且 P1.0 也不是一个合法的 C 语言变量名，所以得给它另起一个名字，这里起的名为 P1\_0，可是 P1\_0 是不是就是 P1.0 呢？C 编译器并不这么认为，所以必须给它们建立联系。这里使用了 Keil C 的关键字 **sbit** 来定义，sbit 的用法有三种：

第一种方法：sbit 位变量名=地址值；

第二种方法：sbit 位变量名=sfr 名称^变量位地址值；

第三种方法：sbit 位变量名=sfr 地址值^变量位地址值；

如定义 PSW 中的 OV 可以用以下三种方法：

sbit OV=0xd2;      说明：0xd2 是 OV 的位地址值

sbit OV=PSW^2;    说明：其中 PSW 必须先用 sfr 定义好

sbit OV=0xD0^2;    说明：0xD0 就是 PSW 的地址值

因此这里用符号 P1\_0 来表示 P1.0 引脚，也可以起类似 P10 的名字，只要下面程序中也随之更改就行了。

### (5) “while(1);”的作用

while() 是 C 语言里的循环控制语句，它的具体用法将在项目 5 里介绍，这里讲解为什么要加上这个循环语句。

当程序执行完 printf() 函数后，它还将向下执行，但后面的空间并没有存放程序代码，这时程序会乱运行，也就是说出现了跑飞现象。加上 “while(1);” 语句，是让程序一直停止在这里不再往下运行，即防止程序跑飞。