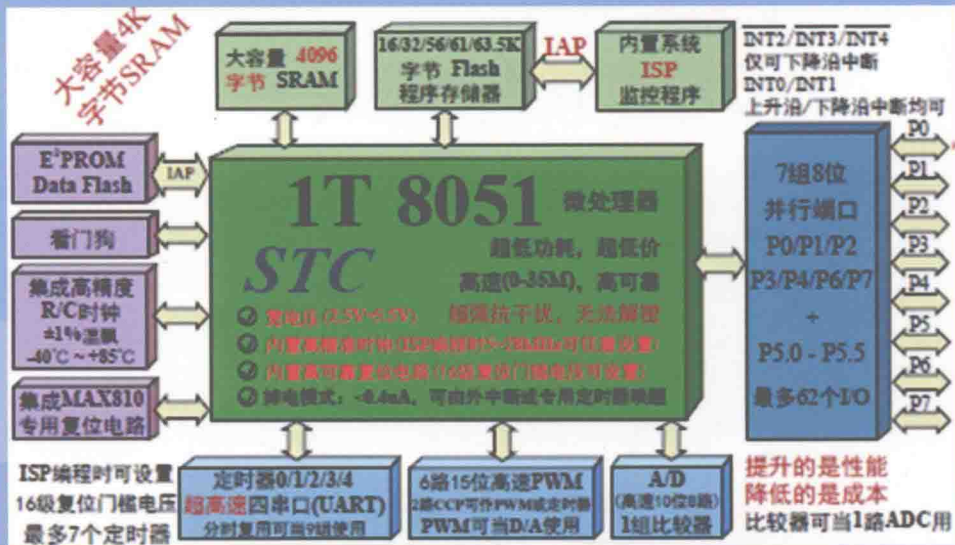


嵌入式C语言程序设计

——基于STC15W4K32S4系列单片机

丁向荣 编 著
姚永平 主 审



教材内容及特色:

- ◆ C语言程序设计与嵌入式系统有机融合, 满足当前C语言与单片机课程改革的需要
- ◆ 基于STC官方IAP15W4K58S4单片机开发板开发教学例程, 直接体现C语言在嵌入式系统中的应用
- ◆ 理实一体, 每章都安排有相应的实验项目, 体现教材的应用性与工程性
- ◆ STC单片机大学推广计划的合作教材
- ◆ 免费提供教学课件与教学例程程序代码



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>

新编高等教育电子信息类规划教材

嵌入式 C 语言程序设计

——基于 STC15W4K32S4 系列单片机

丁向荣 编著

姚永平 主审

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

C 语言是目前最为基础、最为流行的程序设计语言,具有简洁、紧凑、灵活、实用、高效、可移植性好等优点。C 语言的数据类型丰富,可直接面向机器,既用来编写系统程序,又用来编写应用程序。目前,C 语言已是嵌入式系统应用编程中最为重要的编程语言,是嵌入式应用学子不可或缺的编程语言。

本教材在编写策略上,相比传统 C 语言程序设计教材有较大的突破。将 C 语言与嵌入式应用领域紧密结合,将 C 语言课程与单片机课程有机融合,突显 C 语言学习的应用性,做到有的放矢,又保证单片机学习的系统性,满足目前大学“C 语言课程与单片机课程融合”教学改革的需求。

本教材适合电子信息类专业 C 语言课程或单片机课程的教学用书,亦适合作为无 C 语言基础单片机培训的培训教材以及无 C 语言基础单片机爱好者的自学用书。

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

嵌入式 C 语言程序设计:基于 STC15W4K32S4 系列单片机/丁向荣编著.

—北京:电子工业出版社,2016.4

ISBN 978-7-121-27814-3

I. ①嵌… II. ①丁… III. ①单片微型计算机—C 语言—程序设计—高等学校—教材

IV. ①TP368.1②TP312

中国版本图书馆 CIP 数据核字(2015)第 299825 号

策 划:郭乃明

责任编辑:郭乃明 特约编辑:范 丽

印 刷:涿州市京南印刷厂

装 订:涿州市京南印刷厂

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 印张:18.25 字数:467 千字

版 次:2016 年 4 月第 1 版

印 次:2016 年 4 月第 1 次印刷

印 数:3 000 册 定价:42.00 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:(QQ):34825072。

前 言

C 语言是目前最为基础、最为流行的程序设计语言，具有简洁、紧凑、灵活、实用、高效、可移植性好等优点。C 语言的数据类型丰富，可直接面向机器，既可用于编写系统程序，又可用于编写应用程序。目前，C 语言已是嵌入式系统应用编程中最为重要的编程语言，是嵌入式应用学子不可或缺的编程语言。

本教材在编写策略上，相比传统 C 语言程序设计教材有较大的突破。将 C 语言与嵌入式应用领域紧密结合，将 C 语言课程与单片机课程有机融合，突显 C 语言学习的应用性，做到有的放矢，又保证单片机学习的系统性，充分体现 C 语言与单片机在电子产品设计中的应用性。本教材是在大学“C 语言课程”与“单片机课程”融合的背景下，选用当前市场占有率最高的 STC 单片机为教学平台，以代表 STC 最新、最高水平的 STC15W4K32S4 系列单片机为教学机型，围绕 STC 大学合作计划赠送 STC15 实验箱开发的。本书是 STC 大学计划推荐教材，STC 高性能单片机联合实验室上机实践指导教材，STC 杯单片机系统设计大赛参考教材，是 STC 推荐的全国大学生电子设计竞赛 STC 单片机参考教材。使用本书作为教材的院校将优先免费获得 STC 可仿真的 STC15 系列实验箱的支持（主控芯片为 STC 可仿真的 IAP15W4K58S4）。

本书注重实践性、应用性与工程性，理论联系实际，书中大部分例程具有一定的单片机应用系统功能，实验、实践操作性强。全书共 13 章，包括 C 语言概述、STC 单片机开发环境的建立、IAP15W4K58S4 单片机的存储器与 I/O 口、函数、顺序程序结构与应用编程、选择程序结构与应用编程、循环程序结构与应用编程、数组与指针、IAP15W4K58S4 单片机的定时/计数器、IAP15W4K58S4 单片机中断系统、IAP15W4K58S4 单片机的串行口、C 语言构造数据类型、IAP15W4K58S4 单片机应用系统设计等。

为便于教学，向采用本教材教学的教师免费提供电子课件与例程的源程序。

本书由丁向荣编著，由 STC 单片机创始人姚永平先生主审。

由于编者水平有限。书中定有疏漏和不妥之处，敬请读者不吝指正！书中相关勘误或信息也会动态地公布在 STC 官网上：www.stcmcu.com。您有什么建议，可发电子邮件到：dingxiangrong65@163.com，与作者进一步沟通与交流

作 者

2015 年 10 月于广州

序

21 世纪全球全面进入了计算机智能控制/计算时代，而其中的一个重要方向就是以单片机为代表的嵌入式计算机控制/计算。由于最适合中国工程师/学生入门的 8051 单片机有 30 多年的应用历史，绝大部分工科院校均有此必修课，有几十万名对该单片机十分熟悉的工程师可以相互交流开发/学习心得，有大量的经典程序和电路可以直接套用，从而大幅降低了开发风险，极大地提高了开发效率，这也是 STC.宏晶科技/南通国芯微电子有限公司基于 8051 系列单片机产品的巨大优势。

Intel 8051 技术诞生于 20 世纪 70 年代，不可避免地面临着落伍的危险，如果不对其进行大规模创新，我国的单片机教学与应用就会陷入被动局面。为此，STC.宏晶科技对 8051 单片机进行了全面的技术升级与创新，经历了 STC89/90、STC10/11、STC12、STC15 系列，累计上百种产品：全部采用 Flash 技术（可反复编程 10 万次以上）和 ISP/IAP（在系统可编程/在应用可编程）技术；针对抗干扰进行了专门设计，超强抗干扰；进行了特别加密设计，如 STC15 系列现无法解密；对传统 8051 进行了全面提速，指令速度最快提高了 24 倍；大幅提高了集成度，如集成了 A/D、CCP/PCA/PWM（PWM 还可当 D/A 使用）、高速同步串行通信端口 SPI、高速异步串行通信端口 UART、定时器、看门狗、内部高精度时钟（ $\pm 1\%$ 温漂， $-40^{\circ}\text{C} \sim +85^{\circ}\text{C}$ 之间，可彻底省掉外部昂贵的晶振）、内部高可靠复位电路（可彻底省掉外部复位电路）、大容量 SRAM、大容量 EEPROM、大容量 Flash 程序存储器等。针对大学教学，现 STC15 系列一个单芯片就是一个仿真器，定时器改造为支持 16 位自动重载（学生只须学一种模式），串行口通信波特率计算改造为 $[\text{系统时钟}/4/(\text{65536}-\text{重装数})]$ ，极大地简化了教学，针对实时操作系统 RTOS 推出了不可屏蔽的 16 位自动重载定时器，并且在最新的 STC-ISP 烧录软件中提供了大量的贴心工具，如范例程序/定时器计算器/软件延时计算器/波特率计算器/头文件/指令表/Keil 仿真设置等。封装也从传统的 PDIP40 发展到 DIP8/DIP16/DIP20/SKDIP28，SOP8/SOP16/SOP20/SOP28，TSSOP20/TSSOP28，DFN8/QFN28/QFN32/QFN48/QFN64，LQFP32/LQFP48/LQFP64S/LQFP64L，每个芯片的 I/O 口从 6 个到 62 个不等，价格从 0.89 元到 5.9 元不等，极大地方便了客户选型和设计。

2014 年 4 月，STC 宏晶科技重磅推出了 STC15W4K32S4 系列单片机，宽电压工作范围，不需要任何转换芯片，STC15W4K32S4 系列单片机可直接通过电脑 USB 接口进行 ISP 下载编程，集成了更多的 SRAM（4K 字节）、定时器 7 个（5 个普通定时器+CCP 定时器 2）、串口（4 个），集成了更多的高性能部件（如比较器、带死区控制的 6 路 15 位专用 PWM 等）；开发了功能强大的 STC-ISP 在线编程软件，包含了项目发布、脱机下载、RS-485 下载、程序加密后传输下载、下载需口令等功能，并已申请专利。IAP15W4K58S4 一个芯片就是一个仿真器(OCD, ICE)，人民币 5.6 元，是全球第一个实现一个芯片就可以仿真的，再也不需要啥 J-Link/D-Link 了。

在中国民间草根企业掌握了 Intel 8051 单片机技术，以“初生牛犊不怕虎”的精神，击

溃了欧美竞争对手之后，站在 8051 单片机发展的前沿，正在向 32 位前进的途中，STC 感恩社会，回馈社会，全力支持我国的单片机/嵌入式系统教育事业，STC 大学推广计划正在如火如荼地进行中，免费向一本、二本大学赠送可仿真的 STC15 系列实验箱（仿真芯片 IAP15W4K58S4），共建 STC 高性能单片机联合实验室。本教材为 STC 大学推广计划的合作教材，也是 STC 杯单片机系统设计大赛的推荐教材。

对大学计划与单片机教学的看法

STC 大学计划在如火如荼的进行中，第九届“STC 杯单片机系统设计大赛”刚成功落幕，全国数百所高校，1100 支队伍参赛；在国内多所大学建立了 STC 高性能单片机联合实验室，已建和在建的如上海交通大学、复旦大学、同济大学、浙江大学、南京大学、东南大学、吉林大学、哈尔滨工业大学、哈尔滨工业大学(威海)、东北大学、兰州大学、西安交通大学、西北工业大学、西北农林科技大学、南开大学、天津大学、中山大学、厦门大学、山东大学、四川大学、成都电子科技大学、中南大学、湖南大学、中国农业大学、中国海洋大学、中央民族大学、北京师范大学、北京航空航天大学、南京航空航天大学、沈阳航空航天大学、南昌航空大学、北京理工大学、大连理工大学、华南理工大学、南京理工大学、武汉理工大学、华东理工大学、太原理工大学、上海理工大学、浙江理工大学、河南理工大学、东华理工大学、兰州理工大学、天津理工大学、天津工业大学、哈尔滨理工大学、哈尔滨工程大学、合肥工业大学、北京工业大学、南京工业大学、浙江工业大学、广东工业大学、沈阳工业大学、河南工业大学、北京化工大学、北京工商大学、华北电力大学(北京)、华北电力大学(保定)、长安大学、西南大学、西南交通大学、福州大学、南昌大学、东华大学、上海大学、苏州大学、江南大学、河海大学、江苏大学、安徽大学、新疆大学、石河子大学、齐齐哈尔大学、中北大学、河北大学、河南大学、黑龙江大学、扬州大学、南通大学、宁波大学、深圳大学、北京林业大学、南京林业大学、东北林业大学、南京农业大学、大连海事大学、西安电子科技大学、杭州电子科技大学、桂林电子科技大学、南京邮电大学、西安邮电大学、西安科技大学、河南科技大学、天津财经大学、南京财经大学、首都师范大学、华南师范大学、上海师范大学、沈阳师范大学、河南师范大学、中国计量学院、中国石油大学、中国矿业大学等国内著名高校，以及广东轻工职业技术学院、深圳信息职业技术学院、深圳职业技术学院等著名的职业高校

上海交通大学/西安交通大学/浙江大学/山东大学/成都电子科技大学等著名高校的多位知名教授使用 STC 1T 8051 创作的全新教材也在陆续推出中。多所高校每年都有用 STC 单片机进行的全校创新竞赛，如杭州电子科技大学/南通大学/湖南大学/哈尔滨工业大学(威海)/山东大学等。

现在学校的学生单片机入门到底应该先学 32 位好还是先学 8 位的 8051 好？我觉得还是 8 位的 8051 单片机好。因为现在大学嵌入式只有 64 个学时，甚至只有 48 个学时，学生能把 8 位的 8051 单片机学懂做出产品，今后只要给他时间，他就能触类旁通了。但如果也只给 48 个学时去学 ARM，学生没有学懂，最多只能搞些函数调用，没有意义，培养不出真正的人才。所以大家反思说，还是应该先以 8 位单片机入门。C 语言要与 8051 单片机融合教学，大一第一学期就要开始学（现在有些中学的课外兴趣小组都在学 STC 的 8051 + C 语言），大三学有余力的学生再选修 32 位嵌入式单片机课程。

对大学工科非计算机专业 C 语言教学的想法

再讲讲 C 语言，现在工科非计算机专业讲 C 语言的书多是空中飘着，落不着地，学完之后不知道干什么。以前我们学 BASIC/C，学完用 DOS 系统，也在 DOS 下开发软件。现在学生学完 C，要从 Windows 去返回 DOS 运行，学的 C 也不能在 8051 上运行。嵌入式 C 语言有多个版本，国内 Keil C 流行，现我们也在开发我们中国人自己的 C 编译器。现在学标准 C 语言，没办法落地了，学完了，PC 上干不了事，单片机上也动不了。我们现在推教学改革将单片机和 C 语言（嵌入式 C，面向控制的 C）放在一门课中讲，在大一的第一学期就讲，学生学完后就知道他将来能干啥了，大二的第二学期再开一门 Windows 下的 C++ 开发，正好我们的单片机 C 语言给它打基础。学生学完模电/数电（FPGA）/数据结构/RTOS（实时操作系统）/自动控制原理/数字信号处理等后，在大三再开一门综合电子系统设计，这样人才就诞生了。我们现在主要的工作是在推进中国的工科非计算机专业高校教学改革，研究成果的具体化，就是大量高校教学改革教材的推出。丁向荣老师编著的这本书，就是我们的研究成果的杰出代表。希望能在我们这一代人的努力下，让我们中国的嵌入式单片机系统设计全球领先。

对全国大学生电子设计竞赛的支持

2015 年/2017 年全国大学生电子设计竞赛，采用可仿真的超高速 STC15 系列 1T 8051 单片机为主控芯片设计（不需外部晶振，不需外部复位，宽电压，一片芯片就是一台仿真器）。

获得最高奖的参赛队伍（限一支），STC 特别奖励其 10 万元（其全体指导老师分享 7 万元，其全体参赛学生分享 3 万元）。

获得一等奖的参赛队伍（限 300 支以内）STC 特别奖励每队 5000 元（参赛学生分享 3000 元，老师分享 2000 元）。

获奖单位需将获奖作品软/硬件全部提供给 STC 开源，其知识产权归提供高额奖金的 STC 所有，获奖单位需在大赛结束后 2 个月内提供大赛组委会和学校提供的相应证明来申请。

举行全国大学生电子设计竞赛校内选拔赛的 211 高校，可获得 1 万元的赞助，条件是组织 100 个以上的队伍参赛（每队 3 人），并有 50 个以上的队伍采用 STC 可仿真的超高速 STC15 系列 1T 8051 单片机为主控芯片设计参赛作品。

在校内举办 STC 杯单片机系统设计大赛的 211 高校，可获得 1 万元的赞助，条件是组织 100 个以上的队伍参赛（每队 3 人），采用 STC 可仿真的超高速 STC15 系列 1T 8051 单片机为主控芯片设计参赛作品。

在校内举办 STC 杯单片机系统设计大赛的普通一本高校，可获得 8000 元的赞助，条件是组织 100 个以上的队伍参赛（每队 3 人），采用 STC 可仿真的超高速 STC15 系列 1T 8051 单片机为主控芯片设计参赛作品。

在校内举办 STC 杯单片机系统设计大赛的普通二本高校，可获得 5000 元的赞助，条件是组织 100 个以上的队伍参赛（每队 3 人），采用 STC 可仿真的超高速 STC15 系列 1T 8051 单片机为主控芯片设计参赛作品。

在校内举办 STC 杯单片机系统设计大赛的普通三本高校，可获得 4000 元的赞助，条件是组织 100 个以上的队伍参赛（每队 3 人），采用 STC 可仿真的超高速 STC15 系列 1T 8051

单片机为主控芯片设计参赛作品。

在校内举办 STC 杯单片机系统设计大赛的高职高专高校，可获得 3000 元的赞助，条件是组织 100 个以上的队伍参赛（每队 3 人），采用 STC 可仿真的超高速 STC15 系列 1T 8051 单片机为主控芯片设计参赛作品。

感谢 Intel 公司发明了经久不衰的 8051 体系结构，感谢丁向荣老师的这本新书，保证了中国 30 年来的单片机教学与世界同步。本书是 STC 大学计划推荐教材，STC 高性能单片机联合实验室上机实践指导教材，STC 杯单片机系统设计大赛参考教材，是 STC 推荐的全国大学生电子设计竞赛 STC 单片机参考教材，采用本书作为教材的院校将优先免费获得我们可仿真的 STC15 系列实验箱的支持（主控芯片为 STC 可仿真的 IAP15W4K58S4）。

明知山有虎，偏向虎山行！

STC MCU Limited: Andy.姚

www.STCMCU.com www.GXWMCU.com

2015/6/5

目 录

第 1 章 C 语言概述	(1)
1.1 计算机程序与计算机语言	(1)
1.2 C 语言的发展与主要特点	(2)
1.3 程序的算法	(3)
1.3.1 算法的概念	(4)
1.3.2 算法的描述	(5)
1.4 Keil C51	(7)
习题 1	(8)
第 2 章 STC 单片机开发环境的建立	(9)
2.1 Keil C 集成开发环境	(9)
2.1.1 Keil μ Vision4 集成开发环境概述	(10)
2.1.2 应用 Keil μ Vision4 开发工具编辑、编译用户程序, 生成机器代码	(11)
2.1.3 应用 Keil μ Vision4 集成开发环境调试用户程序	(16)
2.2 STC15W4K32S4 系列单片机硬件实验平台的建立	(20)
2.2.1 单片机概念	(20)
2.2.2 STC15W4K32S4 系列单片机	(21)
2.2.3 STC15-IV 版实验箱简介	(23)
2.2.4 STC 系列单片机的在线编程	(24)
2.2.5 单片机应用程序的下载与运行	(26)
2.2.6 Keil μ Vision4 与 STC 仿真器的在线仿真	(28)
2.2.7 STC-ISP 在线编程软件的其他功能	(30)
实验 2.1 Keil μ Vision4 集成开发环境的操作使用	(31)
实验 2.2 STC 单片机用户程序的在线编程与在线仿真	(32)
习题 2	(35)
第 3 章 IAP15W4K58S4 单片机的存储器与 I/O 口	(36)
3.1 IAP15W4K58S4 单片机的引脚功能	(36)
3.2 IAP15W4K58S4 单片机的内部结构	(41)
3.3 IAP15W4K58S4 单片机的存储结构	(42)
3.4 IAP15W4K58S4 单片机的并行 I/O 口	(46)
3.4.1 IAP15W4K58S4 单片机的并行 I/O 口与工作模式	(46)
3.4.2 IAP15W4K58S4 单片机的并行 I/O 口的结构	(47)
3.5 IAP15W4K58S4 单片机的时钟与复位	(49)
3.5.1 IAP15W4K58S4 单片机的时钟	(49)
3.5.2 IAP15W4K58S4 单片机的复位	(51)

习题 3	(53)
第 4 章 函数	(54)
4.1 C 语言程序的结构	(54)
4.2 预处理命令	(55)
4.3 函数的定义	(61)
4.3.1 函数的分类	(61)
4.3.2 函数的定义	(61)
4.4 函数的调用	(63)
4.5 函数的递归调用与再入函数	(64)
*4.6 中断函数	(65)
习题 4	(66)
第 5 章 顺序程序结构与应用编程	(67)
5.1 C 语言的标识符与关键字	(67)
5.2 常量	(69)
5.3 变量	(71)
5.3.1 区别变量名与变量值	(71)
5.3.2 变量的定义和取名规则	(72)
5.3.3 8051 单片机特殊功能寄存器的访问	(75)
5.3.4 8051 单片机位寻址区 (20H~2FH) 位变量的定义	(76)
5.4 C 语言运算符与表达式	(77)
5.4.1 赋值运算符与赋值表达式	(77)
5.4.2 算术运算符与算术表达式	(77)
5.4.3 逗号运算符与表达式	(78)
5.4.4 位运算符与表达式	(78)
5.4.5 复合运算符与表达式	(78)
5.5 C 语言的顺序程序结构	(79)
5.5.1 表达式语句与复合语句	(79)
5.5.2 顺序程序结构程序实例	(80)
实验 5.1 IAP15W4K58S4 单片机 I/O 口的基本输入/输出操作	(81)
实验 5.2 IAP15W4K58S4 单片机 I/O 口的位输入/输出操作	(82)
实验 5.3 IAP15W4K58S4 单片机 I/O 口的位逻辑运算操作	(83)
习题 5	(84)
第 6 章 选择程序结构与应用编程	(86)
6.1 数值关系运算符与表达式	(86)
6.2 条件运算符与表达式	(86)
6.3 逻辑关系运算符与表达式	(87)
6.4 条件分支语句与应用编程	(87)
6.5 开关语句与应用编程	(88)
实验 6.1 IAP15W4K58S4 单片机 I/O 口的逻辑控制 (if 语句) 操作	(89)

实验 6.2 IAP15W4K58S4 单片机 I/O 口的逻辑控制 (switch/case 语句) 操作	(90)
习题 6	(91)
第 7 章 循环程序结构与应用编程	(93)
7.1 while 语句与 do-while 语句	(93)
7.2 for 语句	(94)
7.3 goto 语句、break 语句与 continue 语句	(96)
实验 7.1 IAP15W4K58S4 单片机控制 LED 灯闪烁——while 语句的应用	(96)
实验 7.2 IAP15W4K58S4 单片机控制 LED 灯闪烁——for 语句的应用	(97)
习题 7	(98)
第 8 章 数组与指针	(99)
8.1 数组	(99)
8.1.1 一维数组	(99)
8.1.2 二维数组	(103)
8.1.3 字符数组	(104)
8.2 指针	(106)
8.2.1 地址和指针	(106)
8.2.2 一维数组和指针	(107)
8.2.3 二维数组和指针	(110)
8.2.4 字符数组和指针	(111)
实验 8.1 IAP15W4K58S4 单片机程序存储器的应用——数组的应用编程	(113)
实验 8.2 IAP15W4K58S4 单片机片内扩展 RAM 的测试	(114)
实验 8.3 IAP15W4K58S4 单片机程序存储器的应用——指针的应用编程	(114)
习题 8	(116)
第 9 章 IAP15W4K58S4 单片机的定时/计数器	(117)
9.1 IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的结构和工作原理	(117)
9.2 IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的控制	(118)
9.3 IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的工作方式	(120)
9.4 IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的应用举例	(122)
9.4.1 IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的定时应用	(123)
9.4.2 IAP15W4K58S4 单片机定时/计数器 (T0/T1) 的计数应用	(124)
9.4.3 T0、T1 的综合应用	(125)
9.5 IAP15W4K58S4 单片机的定时/计数器 T2	(128)
9.6 IAP15W4K58S4 单片机的定时器 T3、T4	(129)
9.7 IAP15W4K58S4 单片机的可编程时钟输出功能	(131)
实验 9.1 IAP15W4K58S4 单片机定时/计数器的应用——秒表	(134)
实验 9.2 IAP15W4K58S4 单片机定时/计数器的应用——频率计	(134)
习题 9	(135)
第 10 章 IAP15W4K58S4 单片机中断系统	(137)
10.1 中断系统概述	(137)

10.1.1	中断系统的几个概念	(137)
10.1.2	中断的技术优势	(138)
10.1.3	中断系统需要解决的问题	(138)
10.2	IAP15W4K58S4 单片机的中断系统	(138)
10.2.1	IAP15W4K58S4 单片机的中断请求	(139)
10.2.2	IAP15W4K58S4 单片机的中断响应	(145)
10.2.3	IAP15W4K58S4 单片机中断应用举例	(147)
10.3	IAP15W4K58S4 单片机外部中断的扩展	(151)
实验 10.1	IAP15W4K58S4 单片机定时器中断的应用编程	(152)
实验 10.2	IAP15W4K58S4 单片机外部中断的应用编程	(153)
习题 10		(154)
第 11 章	IAP15W4K58S4 单片机的串行口	(155)
11.1	串行通信基础	(155)
11.2	IAP15W4K58S4 单片机的串行口 1	(158)
11.2.1	串行口 1 的控制寄存器	(158)
11.2.2	串行口 1 的工作方式	(160)
11.2.3	串行口 1 的波特率	(164)
11.2.4	串行口 1 的应用举例	(165)
11.3	IAP15W4K58S4 单片机与 PC 机的通信	(173)
11.3.1	单片机与 PC 机 RS-232 串行通信的接口设计	(173)
11.3.2	IAP15W4K58S4 单片机与 PC 机 USB 总线通信的接口设计	(176)
11.3.3	IAP15W4K58S4 单片机与 PC 机串行通信的程序设计	(177)
11.4	IAP15W4K58S4 单片机串行口 1 的中继广播方式	(178)
11.5	IAP15W4K58S4 单片机串行口 2	(179)
11.6	IAP15W4K58S4 单片机串行口 3	(180)
11.7	IAP15W4K58S4 单片机串行口 4	(182)
11.8	IAP15W4K58S4 单片机串行口硬件引脚的切换	(183)
实验 11.1	IAP15W4K58S4 单片机的双机通信	(185)
实验 11.2	IAP15W4K58S4 单片机与 PC 机间的串行通信	(186)
习题 11		(188)
第 12 章	C 语言构造数据类型	(190)
12.1	结构体类型	(190)
12.1.1	结构体类型的定义	(190)
12.1.2	结构体类型的应用	(191)
12.1.3	结构体数组的应用	(197)
12.1.4	结构体变量和结构体指针变量在函数中的应用	(199)
12.2	共用体	(200)
12.3	枚举类型	(203)
实验 12.1	结构体类型的应用编程——时钟	(205)

实验 12.2 共用体类型的应用编程——方波发生器	(206)
实验 12.3 枚举类型的应用编程——季节的显示	(207)
习题 12	(208)
第 13 章 IAP15W4K58S4 单片机应用系统设计	(209)
13.1 LED 数码管	(209)
13.1.1 LED 显示原理	(209)
13.1.2 74HC595 驱动芯片	(212)
13.2 键盘	(215)
13.2.1 键盘工作原理	(215)
13.2.2 独立式按键	(217)
13.2.3 矩阵键盘的结构与原理	(219)
13.2.4 键盘的工作方式	(220)
13.3 电子时钟的设计	(223)
13.4 IAP15W4K58S4 单片机应用系统的低功耗设计与可靠性设计	(228)
13.4.1 IAP15W4K58S4 单片机应用系统的低功耗设计	(228)
13.4.2 IAP15W4K58S4 单片机应用系统的可靠性设计	(237)
实验 13.1 LED 数码管的显示	(239)
实验 13.2 简单键盘的应用编程	(240)
实验 13.3 矩阵键盘的应用编程	(241)
实验 13.4 电子时钟的调试	(243)
习题 13	(244)
附录 A ASCII 码表 (见附表 A)	(245)
附录 B ANSI C 关键字与 Keil C51 编译器扩展的关键字	(246)
附录 C C 语言的运算符种类、优先级与结合性 (见附表 C)	(248)
附录 D C51 常用头文件与库函数	(250)
附录 E STC15-IV 版实验箱各功能模块电路介绍	(257)
附录 F 常用模块文件 (stc15.h、gpio.h、595hc.h)	(265)
附录 G STC15 系列单片机选型表	(272)

第 1 章 C 语言概述

内容提要

本章主要学习计算机中指令的作用、指令与程序的关系，以及计算机编程语言。理解程序设计者所必备知识的四要素：算法、数据结构、程序设计方法与计算机编程语言。着重学习 C 语言的特点，以及程序算法的描述方法，点明了 C51 的特点与应用范畴。

1.1 计算机程序与计算机语言

一个完整的计算机是由硬件和软件两部分组成的，缺一不可。看得到、摸得着的实体部分是计算机的硬件部分，计算机硬件只有在软件的指挥下，才能发挥其效能。计算机采取“存储程序”的工作方式，即事先把程序加载到计算机的存储器中，当启动运行后，计算机便自动地按照程序进行工作。

1. 指令与程序

计算机在人们眼中是“万能”的，能自动完成各种各样的工作。但究其本质，计算机只能完成一些简单的操作，计算机的每一次操作都是根据人们事先指定的指令进行的，通过简单操作的不同组合以及快速运行，计算机就能按照人们的意志完成各种各样的复杂工作。

(1) 指令。指令是规定计算机完成特定任务的命令，微处理器就是根据指令指挥与控制计算机各部分协调地工作。

(2) 程序。程序是指令的集合，是解决某个具体任务的一组指令。在用计算机完成某个工作任务之前，人们必须事先将计算方法和步骤编制成由逐条指令组成的程序，并预先将它以二进制代码（机器代码）的形式存放在程序存储器中。

2. 编程语言

编程语言分为机器语言、汇编语言和高级语言。

(1) 机器语言。机器语言是用二进制代码表示的，是机器能直接识别的语言，因此机器语言程序又称为目标程序。早期的计算机编程就是用二进制代码进行编程的，但机器语言与人们习惯的语言差别太大，难学、难写、难记忆、难阅读、难修改、难推广，当时只有极少数计算机专业人员会用机器语言编程。

(2) 汇编语言。汇编语言是用英文助记符来描述指令的，如用 ADD 表示“加”，SUB 表示“减”等等，记忆、阅读、书写远胜于机器语言，但计算机并不能直接识别与执行汇编语言指令，需要用一种称为汇编程序的软件，将汇编语言指令转换为机器语言指令代码，才可被计算机识别与执行。助记符指令与机器代码指令有着——对应的关系，与机器语言指令一样，

直接面向机器操作，依赖于具体机器的特性，机器语言与汇编语言都称为计算机的低级语言。

(3) 高级语言。高级语言是一种接近于人们习惯使用的自然语言与数学语言的编程语言。20 世纪 50 年代开发出了第一种计算机高级语言——FORTRAN 语言。数十年来，全世界涌现了 2500 种以上高级语言，每种高级语言都有其特定的用途，影响最大的有 FORTRAN 语言和 ALGOL（适合数值计算）、BASIC/QBASIC（适合初学者的小型会话语言）、COBOL（适合商业管理）、PROLOG（人工智能语言）、C 语言（系统描述语言）、C++ 语言（支持面向对象程序设计的大型语言）、Visual Basic（支持面向对象程序设计的语言）等。

高级语言经历了如下几个不同的发展阶段。

① 非结构化语言。初期的高级语言都属于非结构化设计语言，编程风格比较随意，只要符合语法规则即可，程序中的流程可随意跳转，使程序变得难以阅读与维护。早期的 BASIC、FORTRAN 等都属于非结构化设计语言。

② 结构化语言。规定程序必须由顺序结构、选择（分支）结构、循环结构等基本模块构成，程序中流程不允许随意跳转，程序总是由上而下顺序执行各个基本模块。这种程序结构具有结构清晰，易于编写、阅读和维护。QBASIC、FORTRAN77 和 C 语言都属于结构化程序设计语言。

③ 面向对象的语言。非结构化语言、结构化语言都属于基于工作过程语言，编写程序时需要具体指定每一个过程的细节，适用于编写较小规模的程序。在实践应用的发展中，人们又提出了面向对象的程序设计方法。程序面对的不是过程的细节，而是一个个对象，对象是由数据以及对数据进行的操作组成。C++、C#、Java 等语言是支持面向对象程序设计的语言。

1.2 C 语言的发展与主要特点

C 语言是目前国际广泛使用的高级语言。

1. C 语言的发展历程

C 语言的祖先是 BCPL 语言。BCPL 语言如何演化为 C 语言以及 C 语言的发展历程见表 1.1 所示。

表 1.1 C 语言发展历程表

时 间	C 语言发展概况
1967	英国剑桥大学的 Martin Richards 提出了没有类型的 BCPL (Basic Combined Programming Language) 语言
1970	美国 AT&T 贝尔实验室的 Ken Thompson 以 BCPL 语言为基础，设计出了很简单且很接近硬件的 B 语言。但 B 语言过于简单，功能有限
1972-1973	美国 AT&T 贝尔实验室的 D.M.Ritchie 在 B 语言基础上设计出了 C 语言。C 语言既保持了 BCPL 和 B 语言的优点（精炼，接近硬件），又克服了它们的缺点（过于简单，无数据类型）。开发 C 语言的目的是尽可能地降低编程对硬件平台的依赖性，使之具有移植性。C 语言新特点主要体现在具有多种数据类型（如字符、数值、数组、指针等）
1973	最初的 C 语言是为描述和实现 UNIX 操作系统提供一种工作语言，Ken Thompson 和 D.M.Ritchie 合作把 UNIX 的 90% 以上用 C 语言改写。随着 UNIX 的日益广泛使用，C 语言迅速得到推广
1978	Brain W.Kernighan 和 Dennis M.Ritchie 合著了影响深远的名著 The C Programming Language，这本书介绍的 C 语言实际成为了第一个 C 语言标准。1978 年以后，C 语言先后移植到大、中、小和微型计算机上，C 语言很快风靡全世界，成为世界上应用最广泛的程序设计语言
1983	美国国家标准协会 (ANSI) 根据 C 语言问世以来各种版本对 C 语言的发展和扩充，制定了第一个 C 语言标准草案 ('83 ANSI C)

时 间	C 语言发展概况
1989	美国国家标准协会 (ANSI) 公布了一个完整的 C 语言标准——ANSI X3.159-1989 (常称 ANSI C 或 C89)
1990	国际标准化组织 ISO 接受 C89 作为国际标准 ISO/IEC 9899:1990 (简称 C90), 它和 ANSI 的 C89 基本上是相同的
1999	1995 年, ISO 对 C90 做了一些修订, 1999 年又对 C 语言标准进行修订, 在基本保留原来 C 语言特性的基础上, 针对应用的需要, 增加了一些功能, 尤其是 C++ 中的一些功能, 命名为 ISO/IEC 9899:1999, 2001 年与 2004 年先后进行了两次技术修正。ISO/IEC 9899:1999 及其技术修正被称为 C99 标准

2. C 语言的特点

C 语言既可以编写系统软件, 又可以编写应用软件, 主要具有以下特点:

(1) 语言简洁、紧凑, 使用方便、灵活。只有 37 个关键字、9 种控制语句, 程序书写形式自由, 一行中可书写多条语句, 一个语句可分散在多行。

说明: 虽然 C 语言书写形式自由, 为了便于阅读、维护, 建议在学习与应用编程中, 养成良好的书写习惯。

(2) 运算符丰富。有 34 种运算符, 把括号、赋值、强制类型转换等都作为运算符处理, 表达式类型多样化。

(3) 数据类型丰富。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, C99 又扩充了复数浮点类型、超长整型(long long)、布尔类型(bool)。

(4) 模块化结构。具有结构化的控制语句, 如 if/else 语句、while 语句、do/while 语句、switch/case 语句、for 语句等, 用函数作为程序的基本模块单位, 便于实现程序的模块化。

(5) 语法限制不太严格, 程序设计自由度大。如对数组下标越界不做检查, 对变量的类型使用比较灵活, 因此, 不能完全依赖编译查错, 程序员更要养成严谨的工作习惯, 仔细检查, 确保自己的程序正确。

(6) 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。C 语言具有高级语言的功能和低级语言的许多功能, 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。

(7) 用 C 语言编写的程序可移植性好。C 的编译系统简洁, 很容易移植到新系统, 在新系统上运行时, 可直接编译“标准链接库”中的大部分功能, 不须修改源代码。几乎所有计算机系统都可以使用 C 语言。

(8) 生成目标代码质量高, 程序执行效率高。

C 语言既可以编写系统软件, 又可以编写应用软件。许多以前只能用汇编语言处理的问题, 现在都可以改为 C 语言来编程了。如各种单片机、嵌入式系统应用编程都采用 C 语言编程了, Keil C 就是专门针对 8051 单片机或 ARM 系统进行访问扩充的 C 语言。

1.3 程序的算法

一个程序主要包括以下两方面的信息:

(1) 对数据的描述。在程序中要指定用到哪些数据以及这些数据的数据类型和组织形式,

这也就是数据结构。

(2) 对操作的描述。在程序中指定计算机操作的步骤，也就是算法。

数据是操作对象，操作的目的是对数据进行加工处理。作为程序设计人员，必须认真考虑和设计数据结构和操作步骤（即算法）。著名计算机科学家沃思（Nikiklaus Wirth）提出一个公式：

$$\text{算法} + \text{数据结构} = \text{程序}$$

实际上，一个过程化的程序除了以上 2 个主要因素之外，还应当采用结构化程序设计方法来进行程序设计，并且用一种计算机语言来描述。因此，算法、数据结构、程序设计方法和计算机语言等四个方面是一个程序设计人员所应具备的知识。

算法是解决“做什么”和“怎么做”的问题。程序中的操作语句，实际上就是算法的体现。

1.3.1 算法的概念

广义地说，为解决一个问题而采取的方法和步骤，就称为算法。

计算机程序的算法可分为两大类：数值运算算法和非数值运算算法。数值运算的目的是求数值解，例如求平方根、求圆柱体的体积等，都属于数值运算范畴。非数值运算包括的面非常广，最常见的是用于事务管理领域，例如学生档案管理、职工工资管理、图书管理等等。

数值运算往往有现成的模型，可采用数值分析的方法，因此对数值运算算法的研究比较深入。算法比较成熟，对各种数值运算都有比较成熟的算法可供选用，如计算机程序系统中的“数学程序库”，C 语言编译系统中的头文件 `math.h` 中就包含了许多数学运算算法。

非数值运算的种类繁多，要求各异，难以做到全部都有现成的答案，只有一些典型的非数值运算算法（如排序、查找搜索算法）有现成、成熟的算法可供选用。大多问题需要程序设计者参照已有的类似算法思路，自行设计相关问题算法。

【例 1.1】 求 $1+2+3+4+5$ ，试编制求解算法。

解：(1) 用最原始的方法实现。

步骤 1：先求 $1+2$ ，得到结果 3

步骤 2：将步骤 1 得到结果 3，与 3 相加，得到结果 6

步骤 3：将步骤 2 得到结果 6，与 4 相加，得到结果 10

步骤 4：将步骤 3 得到结果 10，与 5 相加，得到结果 15

试想用这种方法求解 $1+2+\dots+100$ ，需要编写多少个步骤？需要 99 个步骤，显然是不可取的。

(2) 寻找一种通用的运算算法。设置 2 个变量， i 为被加数， j 为加数。此外，每次运算的和直接存回被加数变量。用循环算法来求解结果。可将上述算法做如下修改。

步骤 1： $i=1$

步骤 2： $j=2$

步骤 3： $i+j \rightarrow i$

步骤 4： $j+1 \rightarrow j$