



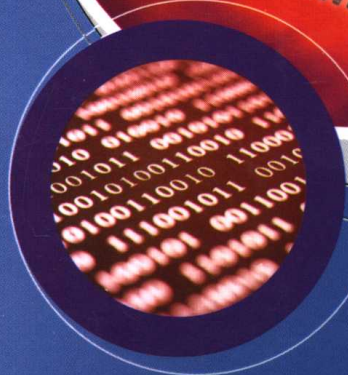
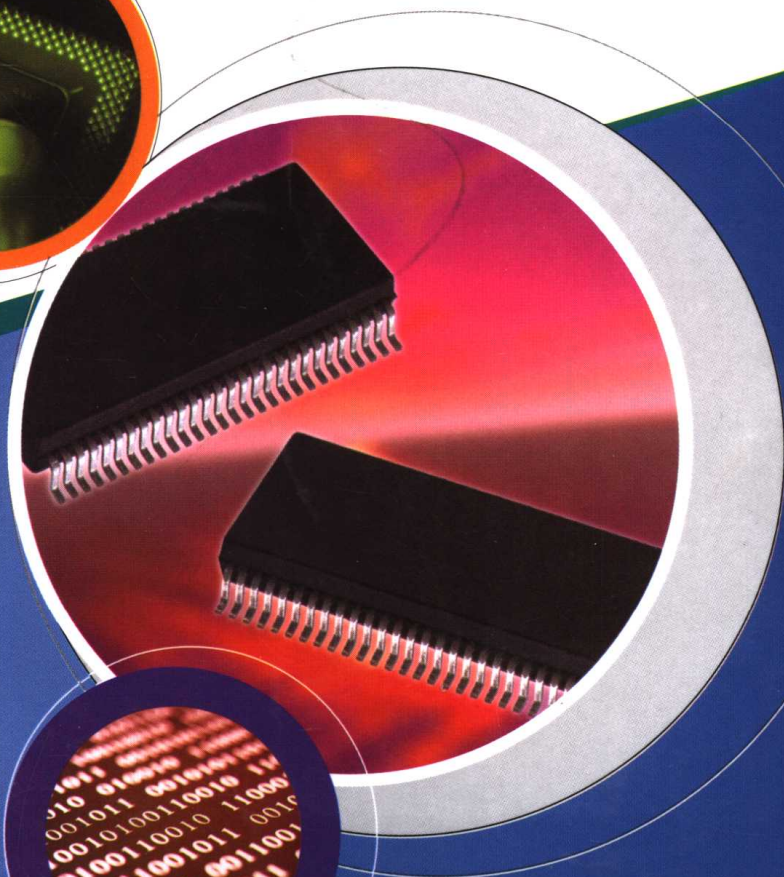
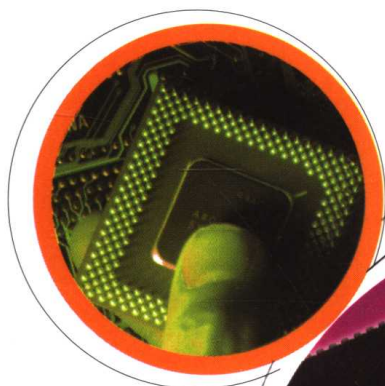
信息产业部中国电子企业协会
全国单片机应用设计师考试认证委员会

职业认证培训教材

单片机应用设计培训教程

——实践篇

夏继强 邢春香 编著



北京航空航天大学出版社

TP368.1/398

:2

2008



信息产业部中国电子企业协会 职业认证培训教材
全国单片机应用设计师考试认证委员会

单片机应用设计培训教程

—— 实践篇

夏继强 邢春香 编著

北京航空航天大学出版社

内 容 简 介

本书是信息产业部中国电子企业协会、全国单片机应用设计师考试认证委员会指定的职业认证培训教材——《单片机应用设计培训教程》套书中的实践篇。

该教程以 80C51 系列单片机为基础,分为入门训练、基础训练、应用设计、综合设计,共 4 章。主要内容包括单片机汇编语言和 C 语言的软件开发过程,集成开发环境的使用;80C51 的内部运算功能单元实验,定时器/计数器实验、中断实验及串行口实验;单片机系统的并行扩展实验,I²C 总线、One-wire、SPI 等多种串行扩展实验;综合应用设计。

本书还配备了按教材全部实验内容统一规划的实验板电路,以供学员在设计和使用参考。

本书既可作为单片机应用设计的培训教材,也可作为大专院校单片机课程的实验教材,或作为广大工程技术人员学习开发单片机的参考用书。

图书在版编目(CIP)数据

单片机应用设计培训教程. 实践篇/夏继强,邢春香
编著. —北京:北京航空航天大学出版社,2008. 1
ISBN 978-7-81124-197-6

I. 单… II. ①夏…②邢… III. 单片微型计算机—技术
培训—教材 IV. TP368. 1

中国版本图书馆 CIP 数据核字(2007)第 195469 号

© 2008, 北京航空航天大学出版社, 版权所有。

未经本书出版者书面许可,任何单位和个人不得以任何形式或手段复制或传播本书内容。
侵权必究。

单片机应用设计培训教程——实践篇

夏继强 邢春香 编著

责任编辑 李春风

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:010-82317024 传真:010-82328026

<http://www.buaapress.com.cn> E-mail: bhp@263.net

北京时代华都印刷有限公司印装 各地书店经销

*

开本:787×960 1/16 印张:11.5 字数:258 千字

2008 年 1 月第 1 版 2008 年 1 月第 1 次印刷 印数:5 000 册

ISBN 978-7-81124-197-6 定价:22.00 元

序

伴随着 20 世纪 90 年代末计算机网络的成熟发展,到 21 世纪人类进入了后 PC 时代。在这一阶段,人们开始考虑如何将客户终端设备变得更加智能化、数字化,从而使得改进后的客户终端设备轻巧便利、易于控制或具有某些特定的功能。单片机技术提供了一种灵活、高效并具有高性价比的产品解决方案,在 market 需求的推动下,单片机目前正处于一种高速发展的阶段。单片机具有体积小、可靠性高、价格低、功能多、使用方便和系统设计灵活等特点,故在民用和工业测控领域得到了广泛的应用。彩电、冰箱、空调、录像机、VCD、摇控器、游戏机、电子玩具、电饭煲等随处可见单片机的影子。现在,单片机已经渗透到我们生活的各个领域,并改变着我们的生活。

在 IT 革命的背景下,单片机产业已成为 21 世纪现代经济舞台上的重要角色,单片机控制实现的产品随处可见。随之而来的问题是相关技术人员的严重短缺。而现在单片机应用开发方面的人才培养远远不能满足行业的需求。据最新统计数据显示,单片机应用设计开发人才的需求高居 IT 行业的首位,但我国的单片机专业人才却处于极度匮乏的状态,缺口在 20 万人以上。

中国电子企业协会(CEEAA)是国家一级协会,隶属信息产业部领导。中国电子企业协会负责单片机应用设计师职业认证工作。根据《国务院关于大力推进职业教育改革与发展的决定》(国发[2002]16 号)、《鼓励软件产业和集成电路产业发展的若干政策》(国务院[2000]18 号)以及信息产业部电子行业职业技能鉴定指导中心印发的《关于授权中国电子企业协会在全国 IC 设计从业人员中开展 IC 设计师、单片机应用设计师技术培训的批复》(信电职监字[2006]041 号)的文件精神,为贯彻落实中共中央、国务院全国人才工作会议精神,强化单片机应用设计人才队伍建设,推动单片机应用设计职业培训事业社会化发展进程,根据企业人才需求的实际情况及单片机应用设计的特点,加快培养一批具有一定单片机应用设计专业技能水平的专门人才,中国电子企业在全国范围内寻找有条件的高校建立联合培训认证中心,共同推广单片机应用设计师职业认证工作。为了配合单片机联合培训认证中心的授课,提高课程质量,中国电子企业与全国单片机应用设计师考试认证委员会组织国内单片机教学、科研方面的教授专家,精心打造了本套单片机应用设计师认证培训教程。

单片机应用设计师职业资格认证是面向广大有志于从事电子信息应用技术、计算机应用



技术方向的相关专业本科生及工程技术人员的。该课程体系坚持“以职业为主线,以就业为指导”的课程设计原则,为提高我国 IT 从业人员硬件技术水平和职业能力提供了完整的教育培训解决方案,以满足企业大量相应岗位的需求。通过对申请人的专业知识、技术、能力的培训和职业化的认证体系,充分体现企业用人需求和岗位技能要求,证书将作为求职、任职、单位录用的依据之一。该培训体系突破传统的教学理念和教学手段,使学员的实践工作能力在短期内得到一个跨跃式的提升,达到独立进行单片机应用开发软/硬件设计的水平,以满足现代企业人才招聘的要求。

毋庸置疑,单片机应用设计人才是现在及未来很长一段时间内的职场热点,广阔的就业前景以及高额的薪资报酬将吸引更多的 IT 人才投身到单片机应用开发领域。我们相信,中国电子企业协会进行的单片机应用设计师职业资格认证工作,将会培养出大量优秀的单片机专业人才满足各个行业的需求,从而更好地促进国内单片机的应用开发。

中国电子企业协会会长:

序 言

自 Intel 公司 1977 年推出 MCS-48 单片机以来,单片机经历了 30 年的发展历程。在我国,单片机应用也有 20 多年的历史。

单片机诞生于微型计算机时代的嵌入式计算机系统(简称嵌入式系统)。嵌入式计算机系统的出现,是现代计算机发展史上的里程碑。嵌入式计算机系统与通用计算机的发展道路完全不同,形成了独立的单片机技术发展道路。

单片机早期的概念是“单片微型计算机”(Single Chip MicroComputer, SCMP),即集成在一个芯片上的微型计算机。由于单片机是用来嵌入到对象体系中实现智能化控制的专用计算机,必须充分满足对象的控制要求,故其后的发展方向与通用计算机大相径庭。“单片微型计算机”很难描述单片机的本质与技术特点,单片机便正名为“微控制器”(MicroController Unit, MCU)。目前单片机正朝着“片上系统”(System on Chip, SoC)发展。

单片机是一个智能化电子系统,应用在人类可以预见的任何领域,以至于人们不再讨论单片机的应用领域。单片机应用应该是电子技术工程师、计算机应用工程师的一门基础性技术。在传统电子系统向智能化电子系统转型时期,普及单片机应用技术具有重要意义。

Intel 公司 30 年前推出的 MCS-48 系列单片机,是完全按照对象领域嵌入式应用要求的嵌入式计算机。在迅速完善后,于 1980 年推出的 MCS-51 系列单片机,被认为是单片机的经典结构。其后 Intel 公司专门致力于发展通用计算机,将 MCS-51 技术开放给客户使用。Philips 公司及其他一些半导体公司便在 MCS-51 的基础上推出了众多 MCS-51 系列兼容单片机,经过多年的发展形成了带有 MCU 特色的单片机系列。此后人们便把所有厂家在 MCS-51 系列单片机基础上发展起来、与 MCS-51 系列的 CPU 兼容,但在工艺上均采用了 CHMOS 技术的单片机,通称为 80C51 系列单片机(或简称 51 系列单片机)。

80C51 系列单片机由于具有经典的体系结构、良好的兼容性,因而成为单片机教育中最普及、最流行的教学内容。加之 20 多年来的不断发展、提高,80C51 系列单片机一直是 8 位单片机中的主流系列,在嵌入式系统技术发展前沿都可以看到 80C51 的身影。因此,把 80C51 系列单片机作为“单片机应用设计师”的培训教材内容是十分妥当的。

本教材包括理论篇与实践篇两部分,在内容安排上突出入门的基础训练。无论是理论篇还是实践篇,都是为了使学员更好地掌握单片机应用的基础知识与基本技能,为学员今后在嵌入式系统应用领域的发展打下良好的基础。

单片机是一个智能化控制内核,有无限广阔的应用领域,也是发展最迅速的技术领域。早



期,单片机应用系统的开发大多是基于单片机及外围芯片、开发装置、汇编语言的软/硬件系统设计;如今,已进入到基于厂家开发平台、集成开发环境、高级语言、操作系统管理的全方位系统设计时代。但从基本理论与基本技术的训练出发,“单片机应用设计师”的培训仍然侧重基础训练,理论教学中仍然以汇编语言为主,没有操作系统内容,实验与实训课是基于实验板与开发装置的应用环境。这不会妨碍基本理论与技能的训练。

与早期单片机应用相比,单片机本身的综合性能、集成度、产品开发环境、编程语言、外围芯片技术、操作系统支持及嵌入式系统周边技术都有了极大的发展。对于未来从事嵌入式系统应用事业的学员,在学好本课程的同时,还应建立以下的单片机应用设计新观念:

(1) 全方位的应用设计。从早期单片机应用系统以功能性设计为主,转变到功能性设计、可靠性、功耗管理等的全方位应用设计。与传统电子系统相比,单片机应用系统的可靠性观念有了很大变化:芯片的可靠性有了极大的提高,单片机应用系统可靠性设计的重点将转向极限应用环境中的可靠性保证;单片机及其外围芯片的普遍 CMOS 化,使单片机应用系统的运行功耗可控,从而使功耗管理成为应用系统的常规设计内容。

(2) 集成开发环境下的产品开发。集成开发环境是半导体厂家为用户提供的—个产品开发环境,它包括单片机的软、硬件平台,程序设计的编辑、汇编、仿真调试工具、操作系统支持、函数库及数据库等。最大程度地缩短了从厂家单片机到用户产品系统开发的距离。在一个良好集成开发环境下可以实现产品开发最大限度的“多、快、好、省”。

(3) 最大化的系统集成。30 年来单片机发展的总趋势,是从单片微型计算机(SCMP)向微控制器(MCU)、片上系统(SoC)发展。随着集成度的不断提高,各种类型的单片机层出不穷。因此,在设计应用系统时,应选择合适的单片机,以使应用系统的外围器件减到最少,实现最大化的单片系统集成。

(4) 操作系统的普遍使用。在以往的单片机应用系统中,很少使用操作系统,随着单片机内部资源的不断增加,实时多任务操作系统的普及,单片机应用系统设计中应尽量使用操作系统。在操作系统的支持下,基于任务级的编程方法既方便、迅速、可靠,又利于软件的移植、修改与升级。

(5) 与周边技术的交叉融合。近年来与单片机相配套的外围器件有了很大的发展,不仅出现了许多具有串行接口、电源管理功能的外围电路芯片,也出现了各种类型的可编程逻辑阵列(FPGA/CPLD)。单片机应用设计师在从事产品开发时,应通晓这些技术领域,并把它们引入到单片机产品研发中来。

(6) 普遍的高级语言程序设计。在本教材中以汇编语言训练为主,是考虑到汇编语言的基础作用。学习、掌握汇编语言有利于对单片机的体系结构、硬件环境的深层理解。但在单片机实际产品开发中,应以 C 语言为主。

“单片机应用设计师”培训是单片机应用的入门基础。有了这个基础,加上建立起产品开发的新观念,就能成为一个优秀的单片机应用工程师。

何立民

2007 年 12 月

前言

职业技能教育是提高我国从业人员技能水平的重要措施。《单片机应用设计培训教程》套书是为配合单片机职业培训的需要而编写的,是信息产业部中国电子企业协会、全国单片机应用设计师考试认证委员会指定的职业认证培训系列教材,本书是《单片机应用设计培训教程》套书中的实践篇。

本书既可作为单片机应用设计的培训教材,也可作为大专院校单片机课程的实验教材,以及广大工程技术人员学习开发单片机的参考用书。

本教程分为入门训练、基础训练、应用设计、综合设计 4 部分,共 4 章。

第 1 章是单片机应用开发的入门训练,介绍汇编语言和 C 语言的软件开发过程,集成开发环境的使用。

第 2 章是单片机应用开发的基础训练,根据 80C51 的内部功能,着重练习定时器/计数器、中断及串行口的使用。

第 3 章是单片机系统的串并行扩展内容,练习单片机系统各个典型接口的应用设计,针对串行外围器件的扩展越来越多的实际情况,设置了串行口、I²C 总线、One-wire、SPI 等多种串行扩展芯片的实训内容。

第 4 章是综合设计,给出 1 个应用前 3 章内容的综合应用实验,学员可以根据兴趣,设计更多的实践内容。在这一章中,还介绍了嵌入式实时多任务操作系统 $\mu\text{C}/\text{OS}-\text{II}$ 的开发方法,但这部分不作为大纲中的考试内容来要求。

在编写过程中,考虑到学员学历及专业背景不同,并希望在有限的集中培训时间内达到夯实基础、快速提高、靠近实用的目的,主要突出以下几个特点:

1. 循序渐进

本教程在内容编排上采用由浅入深、由易到难的原则,从最初的熟悉单片机开发环境,到内部单元功能实验,再到单片机系统扩展实验,直至最后的综合实验,体现了从了解单片机基本原理,到能够根据需要设计出单片机系统的不同阶段的训练内容。不同层次的读者可以根据自己的情况选择练习起点。

2. 突出串行扩展

目前的单片机系统设计中,除了传统的并行外围器件扩展外,串行外围器件扩展也很活



跃,各大芯片制造商都推出这类产品,这也为系统设计尤其是硬件设计带来了方便。本教程力求能够跟上这种发展趋势,除并行外围器件扩展实验外,还介绍了目前使用较多的 UART、I²C 总线、One-wire、SPI 等多种串行扩展芯片的使用方法。

3. 虚拟接口与虚拟器件思想

虚拟接口与虚拟器件是目前单片机系统设计的又一大特色,采用这一思想,可以充分利用单片机的软件资源,实现一些接口和器件的功能,给应用系统设计带来了灵活性。本教程在编写中也反映了这一特色,特别是在串行扩展实验中,给出了多种串行通信的模拟软件包,应用这些软件包可以灵活地扩展出各种串行接口,希望能通过这些实验在虚拟设计上起到抛砖引玉的作用。

4. 高级语言与汇编语言并重

为简化系统编程,增加单片机应用系统程序的可读性和可移植性,单片机编程也从传统的汇编语言编程转向 C 语言编程。为适应这一形势需要,本教程的所有程序都提供了 C 语言代码。同时,为满足不同层次读者的需要,书中也给出了相应的汇编语言程序,可以对照学习。

本书配备了按教材全部实验内容统一规划的实验板电路,以供学员在设计和使用参考。由于书中的 11 个实验均为单片机应用设计的基本实验内容,各培训单位可以根据已有条件参考应用。

本套教材由何立民教授统一筹划,在编写过程中得到了中国电子企业协会培训中心领导的关心指导,也得到了北京航空航天大学出版社的大力支持。

本教程由夏继强、邢春香负责内容规划、修改、定稿。研究生尚真真、钱问发、刘媛媛、习亮、刘铮、曹磊、张小武、陈亮亮参与了程序的编制和软件的调试。何立民教授、张迎新教授对本教程提出了宝贵意见,在此表示衷心的感谢。

由于编者的水平所限,书中难免存在缺点和错误,请广大读者给予批评指正。

作者的电子邮箱:xiajiqiang@buaa.edu.cn。

作 者

2007 年 12 月

目 录

第 1 章 入门训练

1.1 汇编语言及 C51 语言编程方法介绍	1
1.1.1 汇编语言	1
1.1.2 C51 语言	3
1.2 集成开发环境	4
1.2.1 Keil C51 集成开发环境介绍	4
1.2.2 程序固化	10

第 2 章 基础训练

2.1 单片机内部功能单元简介	13
2.1.1 80C51 单片机的内部结构	13
2.1.2 80C51 的内部存储器组织	14
2.2 实验 1——运算功能实验	15
2.3 实验 2——单片机 I/O 口实验	17
2.4 实验 3——定时器/计数器实验	20
2.5 实验 4——中断实验	25
2.6 实验 5——串行扩展实验	29

第 3 章 应用设计

3.1 并行扩展与串行扩展	34
3.1.1 并行扩展	34
3.1.2 I ² C 总线简介	34
3.1.3 One-wire 协议简介	47
3.1.4 SPI 协议简介	55
3.2 实验 6——并行扩展液晶显示器实验	57
3.3 实验 7——ZLG7290 扩展键盘显示实验	61



3.4	实验 8——日历时钟 PCF8563 实验	66
3.5	实验 9——A/D、D/A 转换器 PCF8591 实验	77
3.6	实验 10——DS18B20 数字温度采集实验	80
3.7	实验 11——X5045 看门狗实验	83
 第 4 章 综合设计		
4.1	多功能时钟及温度采集实验	95
4.2	$\mu\text{C}/\text{OS}-\text{II}$ 嵌入式实时多任务操作系统实验	114
4.2.1	嵌入式实时多任务操作系统	114
4.2.2	$\mu\text{C}/\text{OS}-\text{II}$ 实时操作系统	114
4.2.3	$\mu\text{C}/\text{OS}-\text{II}$ 的移植	119
4.2.4	$\mu\text{C}/\text{OS}-\text{II}$ 下的多功能时钟及温度采集实验	121
附录 A 实验板电路图		158
附录 B 实验使用的芯片引脚图		162
附录 C MCS-51 指令系统特点及速查表		163
C.1	指令系统格式及标识	163
C.2	指令系统分类	164
C.3	MCS-51 指令速查表	168
参考文献		170

第 1 章

入门训练

通过对本章的学习和实验,了解程序的编制过程、仿真运行、程序写入及系统脱机运行等内容。

1.1 汇编语言及 C51 语言编程方法介绍

80C51 单片机支持汇编、PL/M、C 和 BASIC 共 4 种语言编程,本教程中的应用程序以 80C51 支持的汇编和 C51 两种语言编写。

1.1.1 汇编语言

1. 80C51 汇编语言简介

汇编语言是机器语言的助记符,80C51 单片机汇编语言格式如下:

[标号]: [操作码] [操作数] ;[注释]

其中:

- ▶ 标号: 用户定义的符号,代表该条指令所在的地址。标号一般以英文字母开始,以“:”结束。目前,一些编译器可以编译较长的标号,但不能超过 31 个字符,一般用 1~8 个字母或数字,看起来清晰方便即可。在汇编语言中,并不是每条语句都要加标号,如果是在子程序调用及转移目标地址时,则必须加上标号,且每行语句只能有一个标号。
- ▶ 操作码: 指令操作码助记符由 2~5 个英文字母组成,它指出 CPU 的工作内容。例如,数据传送指令用 MOV 表示,加 1 指令用 INC 表示等。
- ▶ 操作数: 与操作码之间必须以 1 个或几个空格分隔,根据不同指令,操作数可以有 3 个、2 个、1 个或没有。操作数是正整数,负数用补码表示,不允许有小数或分数。操作数之间用逗号分开。
- ▶ 注释: 用户对该条指令功能的说明。注释可以是汉字,也可以是英文,不参与汇编。注释前必须加分号。

除了指令外,51 汇编语言还定义了一些伪指令,它本身不产生机器指令,是对汇编起某种



控制作用的特殊命令。常用的伪指令有：

- ORG 设置源程序地址值；
- END 结束源程序；
- EQU 定义常数和寄存器名；
- DB 定义常数表；
- DW 定义地址表。

在本教程中，80C51 系列单片机简称为 51 单片机。51 单片机指令集见附录 C。

汇编语言的源代码文件称为源程序文件，以 .ASM 为扩展名。汇编生成 HEX 文件及 LST 文件为输出文件。HEX 文件称为执行文件，以 .HEX 为扩展名；LST 文件是汇编生成的第 2 个文件，可作为文档保存。

2. 采用汇编语言的编程过程

- ① 根据应用要求，在 PC 机上用文本编辑软件编写汇编语言应用程序，即编写源程序。
- ② 利用汇编工具软件对应用程序（源程序）进行汇编，生成目标程序（.OBJ 文件）。在某些情况下，还需要经过链接生成二进制代码程序（.BIN 文件）或十六进制代码程序（.HEX 文件）。
- ③ 将 .OBJ 文件、.BIN 文件或 .HEX 文件调入开发装置仿真运行。

在某些集成开发环境下，上述过程可通过菜单选项依次完成或批处理完成，而不必在命令行方式下进行，为使用者带来极大方便。

3. 采用汇编语言应注意的问题

① 在 51 单片机的 64 KB 程序存储器中，有 5 个特殊单元，对应 5 个中断源的中断入口地址，分别是：0003H、000BH、0013H、001BH 和 0023H。在系统使用中断的情况下，编程时应跳过这部分空间。也可在编程时养成首先跳过这部分空间的习惯，以备进一步开发时能满足有中断的要求。在片内的 128 个 RAM 单元中，20H~2FH 是可位寻址空间，在编程时应充分利用。在 51 单片机中，堆栈设在 RAM 空间（复位后栈顶自动指向 R7），在程序的开始处，应设合适的栈顶。

② 在编写程序时，每行中应尽量使用空格，最好使程序对齐，以便阅读，为便于理解，还要有适当的注释。编程时可以用如下格式：

```
ORG      0000H
LJMP     MAIN           ;跳到主程序
ORG      0003H
LJMP     INTO           ;外部中断 0 入口
ORG      000BH
LJMP     T0             ;定时器/计数器 0 中断入口
:
ORG      0100H
MAIN:    MOV     A, #05H
:
```

③ 习惯使用伪指令,记忆和理解一个变量名比记忆一个地址要容易得多。另外,80C51具有丰富的指令集,有些指令的用法非常巧妙,使用这些指令编程比用常用的指令组合完成同样功能更节省运算时间和代码量。

④ 一些开发环境对编程有特殊要求。例如,程序中不能含有一些特殊的保留字,程序最后要加上 END 等。在使用时应先仔细阅读有关材料,避免编程时出现不必要的麻烦。

1.1.2 C51 语言

1. C51 语言简介

C 语言是一种结构化的编程语言,本书中把开发 80C51 使用的 C 语言简称为 C51 语言。采用 C51 语言编写的应用程序结构清楚、模块化程度高、可读性强且容易移植。应用 C51 语言进行软件开发,用户不必具体考虑寄存器和存储器的分配等工作,而把这部分工作交给编译、链接软件;用户只要了解 80C51 的存储器结构,甚至不必去了解 51 单片机的指令系统。C51 语言集成开发环境一般都提供了数学计算等子程序库,为程序开发带来了方便。虽然采用 C51 语言编程得到的源代码不如有经验人员编写的汇编语言精练,而且不易控制运行时间,但对于相对复杂的系统开发,采用比用汇编语言开发容易得多,可以节省开发时间,并有利于系统的维护和升级。另外,在实时性要求较高的场合,可以采用 C51 语言和汇编语言混合编程。

很多公司提供了 C51 编译器,本书的 C51 程序都是针对 Keil C51 编译器编写的。

2. C51 程序的编写过程

① 根据应用要求,在 PC 机上用文本编辑软件编写 C 语言源程序。

② 利用 C51 编译工具软件对源程序进行编译,生成目标程序(.OBJ 文件)。

③ 利用 C51 链接工具软件对目标程序进行链接定位,生成绝对地址程序后,即可调入开发装置仿真运行。在某种情况下,可将绝对地址程序转化为十六进制代码程序(.HEX 文件)。

有些 C51 开发软件还提供了集成开发环境,上述过程可通过菜单选项依次完成或批处理完成,而不必在命令行方式进行,使用极其方便。

3. 采用 C51 语言应注意的问题

编写 C51 语言程序时,除了上面编写汇编程序提到的书写整齐、有适当的注释外,还要注意不同结构层次的程序应该从不同的起始位置开始,以便层次清楚。另外,还要注意以下几点:

① 与微机上运行的标准 C 程序不同,C51 程序是用于单片机应用系统的,因此,要考虑单片机的资源,例如存储器空间和寻址方式等。虽然编写 C51 程序时不必完全了解 51 的指令系统,但了解指令系统有助于编出精练、实用的程序。

② C51 程序是针对具体应用系统编写的,所以软件的编写一定要在系统硬件的基础上完成,也就是软件编写一定要考虑系统硬件环境。



③ 在编译和链接时,要根据需要合理选择编译控制指令和链接控制指令。大多数集成开发环境会自动完成一些编译控制和链接控制工作,有些选项与用户的具体使用有关,需要手动输入编译环境。例如,选定单片机后,片内 RAM 大小即确定,在链接时编译器会自动完成该工作;XDATA()对应系统硬件的片外 RAM 空间,需要用户根据硬件的实际链接情况给定。

C51 涉及的内容较多,详细内容建议阅读参考文献[1]。

1.2 集成开发环境

用汇编语言或 C 语言编写好程序后,可以通过编译器、链接器直接生成可执行文件,将可执行文件通过编程器或者 ISP 下载写入单片机,将单片机放入目标板即可运行。在实际开发过程中一次成功的可能性很小,往往要花费大量的时间进行调试,仿真器就是用来调试的手段之一。使用仿真器可以实时观察代码运行是否符合用户的要求,并可以在仿真运行环境中进行代码的修改。支持 51 单片机的仿真器很多。如果用户使用仿真器,其提供商会提供与仿真器配套的编译和仿真环境,并有详细说明;如果采取直接下载程序到单片机中,则可使用 Keil C51 集成开发环境编制程序并生成下载文件。本节将介绍使用 Keil C51 集成开发环境时如何调试程序。

为了便于学习,附录 A 给出了与本书配套使用实验板电路图,该实验板通过简单的跳线就可以完成本书的全部实验以及综合设计。读者还可以组合其他的综合设计。

1.2.1 Keil C51 集成开发环境介绍

读者可以从 <http://www.zlgmcu.com>(周立功单片机)网站上免费下载 Keil C51 集成开发软件的测试版。测试版软件在功能、创建应用程序和程序代码的长度方面都有限制,使用者可根据需要购买完全版的 Keil C51 软件。Keil C51 μ Vision2 集成开发环境是基于 80C51 内核的软件开发平台,支持工程的建立、程序的编译与链接、软件仿真、硬件仿真、目标代码的生成等功能。Keil C51 编译器在产生代码的准确性和效率方面达到了较高的水平。

Keil C51 V7.0 的安装与其他软件类似,在此不作详细介绍。Keil C51 集成开发环境的工具栏、菜单栏等的使用,可参考集成开发环境的帮助文件。

与大多数集成开发环境类似,Keil C51 集成开发环境也使用工程的方法来管理文件,在一个工程文件中源程序(C51 程序、汇编程序)、头文件等都可以进行统一管理。

下面通过一个 P2 口输出实验实例,对 Keil C51 集成开发环境进行说明。实验电路如图 1-1 所示(假设要求蜂鸣器发出 4 种不同长短的声音)。

实验步骤如下:

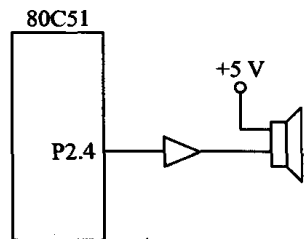


图 1-1 P2 口输出实验电路

① 双击桌面 Keil μ Vision2 快捷图标,进入 Keil C51 集成开发环境,如图 1-2 所示。在菜单栏中选择 Project→New Project,建立一个新工程 P2test. μ v2,并根据提示进行工程的保存等操作。

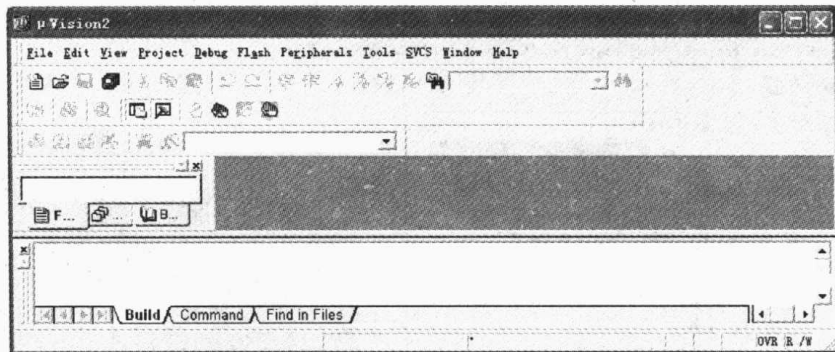


图 1-2 Keil C51 集成开发环境主界面

② 工程保存后,Keil C51 μ Vision2 将立即弹出如图 1-3 所示的器件选择对话框。不同型号的 51 单片机其内部资源不同,Keil C51 μ Vision2 可以根据所选定的芯片进行 SFR 预定义,在软硬件仿真时提供观察窗口等。这里假定选用了 Atmel 公司的 AT89S51 单片机。

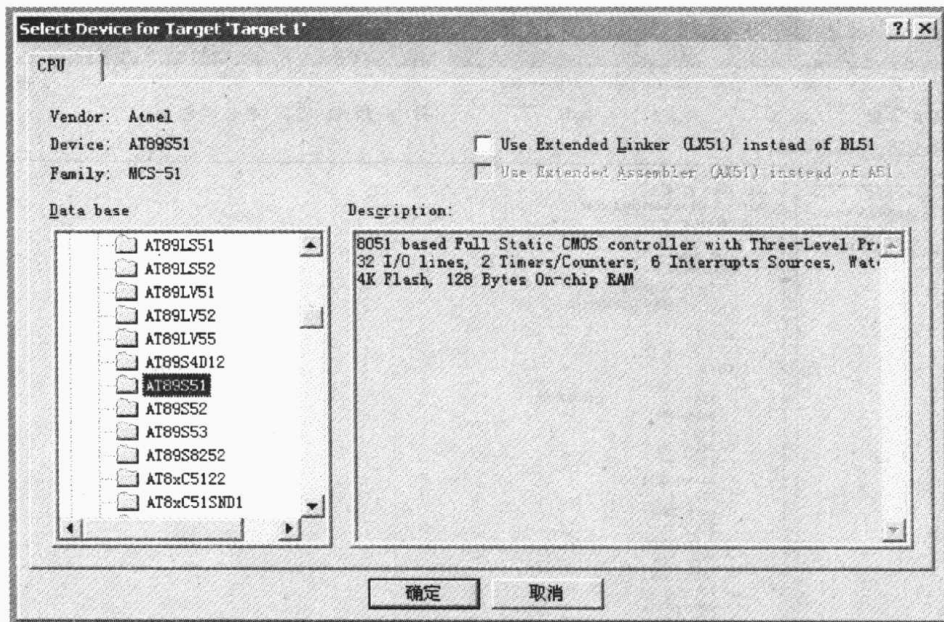


图 1-3 器件选择对话框

③ 新建.C 或.ASM 文件,并保存代码。这里假设编写的汇编程序和 C51 程序命名为 P2test.ASM 和 P2test.C。P2test.ASM 和 P2test.C 文件也可用其他编辑软件预先写好,然



后在工程中添加源文件,如图 1-4 所示,将 P2test.C 添加到工程中,这样即可完成一个完整工程的建立过程。添加 P2test.C 后的窗口如图 1-5 所示。至此,工程文件已经做好了进行编译链接的准备。

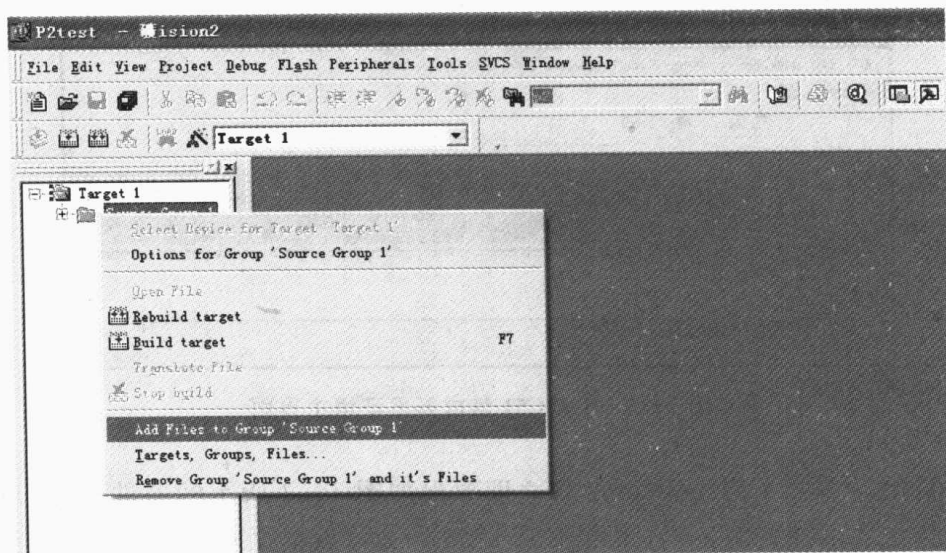


图 1-4 在工程中添加一个源文件

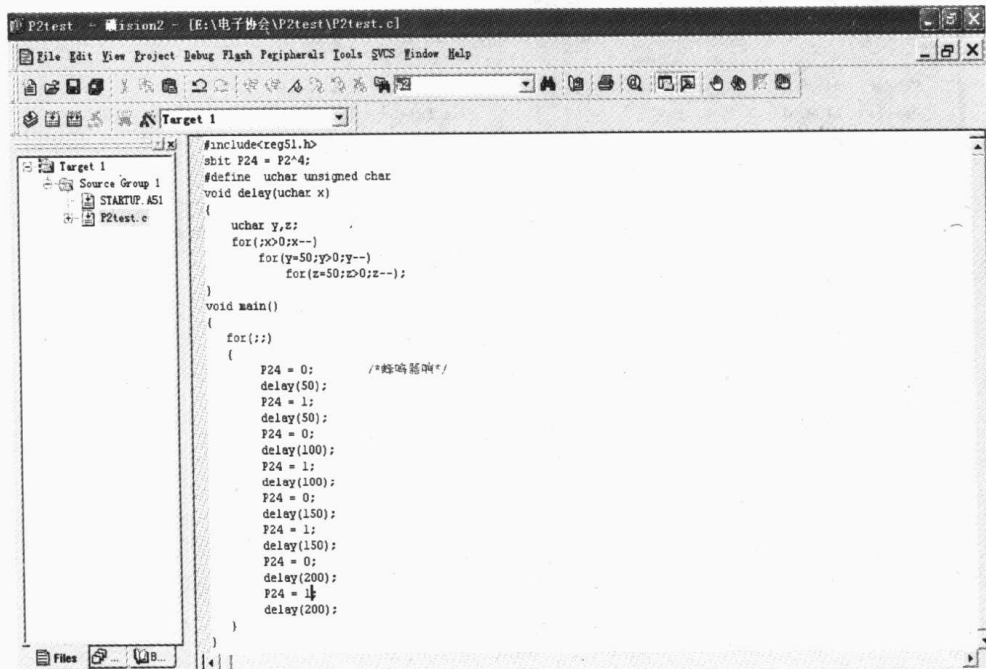


图 1-5 添加 C 源文件后的窗口