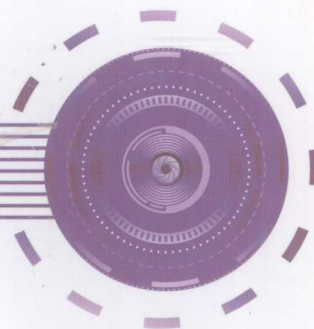




教育部高职高专规划教材



单片机原理与应用

第二版

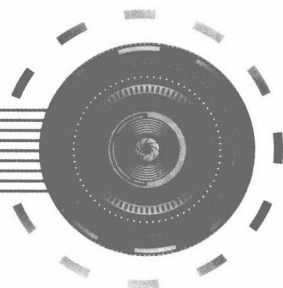
樊明龙 任丽静 主编



化学工业出版社



教育部高职高专规划教材



单片机原理与应用

第二版

樊明龙 任丽静 主编



化学工业出版社

· 北 京 ·

本书采用项目化方式编写,以 AT89C51 为样机,结合调试工具软件 KEIL 和仿真工具软件 PRO-TUES,从实用的角度出发,全面、详细地介绍了 C51 系列单片机的硬件知识、汇编语言程序设计方法和单片机产品的开发流程。

全书由八个项目组成,每个项目又有若干个学习任务。这些任务主要是单片机硬件、软件基础知识的学习,开发工具的学习和常见单片机产品的开发。通过完成相关学习任务,读者将会学习到 AT89C51 单片机的管脚功能、存储器组织、中断系统、定时器、串行口等内部资源的原理和应用方法,同时掌握单片机键盘接口技术、显示接口技术、单总线器件接口技术、现场实时控制技术和通信技术等单片机应用技术。

本书可作为高职高专电子信息类、自动化类、机电类、计算机类相关专业的单片机应用技术课程教材,也可作为相关工程技术人员的参考书。

图书在版编目 (CIP) 数据

单片机原理与应用/樊明龙,任丽静主编. —2 版.
北京:化学工业出版社,2013.11

教育部高职高专规划教材
ISBN 978-7-122-18716-1

I. ①单… II. ①樊…②任… III. ①单片微型计算机-
高等职业教育-教材 IV. ①TP368.1

中国版本图书馆 CIP 数据核字 (2013) 第 246046 号

责任编辑:张建茹
责任校对:宋 夏

文字编辑:云 雷
装帧设计:尹琳琳

出版发行:化学工业出版社(北京市东城区青年湖南街 13 号 邮政编码 100011)

印 刷:北京市振南印刷有限责任公司

装 订:三河市宇新装订厂

787mm×1092mm 1/16 印张 14 字数 348 千字 2014 年 2 月北京第 2 版第 1 次印刷

购书咨询:010-64518888 (传真:010-64519686) 售后服务:010-64518899

网 址: <http://www.cip.com.cn>

凡购买本书,如有缺损质量问题,本社销售中心负责调换。

定 价:29.00 元

版权所有 违者必究

前 言

高等职业教育的教学,以培养学生综合职业能力为目标。当前的高职高专教学改革,打破了传统的学科体系课程结构,建立基于工作过程的课程体系。本书以项目为载体,采用任务驱动方式编写。以 AT89C51 单片机为背景机,结合 Keil、Proteus 等开发工具,系统介绍了 MCS-51 系列单片机硬件结构、汇编语言程序设计和应用系统开发方法等内容。本书的主要特点有以下几个方面。

(1) 采用项目引领为主体结构,以任务驱动教与学。按照实际工作任务、工作过程和工作情境组织课程。从岗位需求出发,让学生进入工作实践。以工作任务为中心来整合相应的知识、技能,实现理论与实践的统一。为学生提供体验完整工作过程的学习机会。

(2) 适应理实一体化教学要求。理论实践一体化教学突破以往理论与实践相脱节的现象,强调充分发挥教师的主导作用,通过设定教学任务和教学目标,让师生双方边教、边学、边做,全程构建素质和技能培养框架,丰富课堂教学和实践教学环节,提高教学质量。在整个教学环节中,理论和实践交替进行,直观和抽象交错出现,没有固定的先实后理或先理后实,而理中有实,实中有理。突出学生动手能力和专业技能的培养,充分调动和激发学生学习兴趣。

(3) 注重对学科综合能力的培养。学科综合能力,主要指学生对学科内不同要素、部分环节之间内在联系的掌握程度,以及运用学科知识和方法分析、解决实际问题的能力。

(4) Keil 和 Proteus 开发平台的应用。Keil μ Vision4 引入灵活的窗口管理系统,使开发人员能够使用多台监视器,并提供了视觉上的表面对窗口位置的完全控制的任何地方。Proteus 软件不仅具有其他 EDA 工具软件的仿真功能,还能仿真单片机及外围器件。它是目前最好的仿真单片机及外围器件的工具。

本书内容由 8 个项目组成,每个项目又分解为若干个任务。每个任务的完成都有 Keil 项目程序和 Proteus 仿真电路。

参与本书编写的工作人员主要由高职高专院校从事单片机教学研究的骨干教师和企业从事单片机系统应用开发的技术人员组成。樊明龙、任丽静担任主编,参与编写的有马梅、张娟、戎朝余等。其中马梅编写了项目 1、项目 2,任丽静编写了项目 3,张娟编写了项目 4、项目 5、项目 6、项目 7,樊明龙编写了项目 8,戎朝余对全书的项目选择与确定做了大量的调研工作。

本书可作为高职高专电仪类、自动化类、电子信息类、计算机类、机电类相关专业教材,也可供单片机爱好者、企业工作人员参考使用。

在本书的编写过程中,参考了目前国内比较优秀的单片机书籍,在此谨向有关作者表示感谢。

限于作者的水平,书中难免有不足之处,恳请读者批评指正。

编者

2013 年 7 月

目 录

项目 1 基础知识和工具软件的使用	1
任务 1 熟悉计算机中的数制与编码	1
任务 2 学会使用硬件开发平台 KEIL	5
任务 3 学会使用硬件仿真平台 PROTEUS	9
思考题与习题	12
项目 2 单片机的硬件资源及应用	14
任务 1 学习单片机相关概念	14
任务 2 从里到外熟悉 AT89C51 单片机	17
任务 3 学习 AT89C51 主要硬件资源	20
任务 4 单片机必需的三个外部电路	31
任务 5 AT89C51 最小应用系统	35
思考题与习题	38
项目 3 汇编语言程序设计及调试	40
任务 1 AT89C51 指令格式和寻址方式	40
任务 2 熟悉 AT89C51 指令系统	43
任务 3 应用伪指令控制汇编过程	60
任务 4 汇编语言程序设计及调试	62
思考题与习题	71
项目 4 电子音乐发生器设计与仿真	74
任务 1 熟悉 AT89C51 中断系统	74
任务 2 定时/计数器及应用方法	82
任务 3 电子音乐发生器的设计与仿真	92
思考题与习题	100
项目 5 温度测量显示系统的设计与仿真	102
任务 1 学习键盘接口技术	102
任务 2 学习 LED 数码显示管接口技术	106
任务 3 温度测量和显示控制系统设计与仿真	113
任务 4 图像显示系统的设计与仿真	130
思考题与习题	134

项目 6 实现单片机与 PC 机通信	135
任务 1 了解串行通信基本知识	135
任务 2 AT89C51 串行口	137
任务 3 串行口扩展并行 I/O 口的设计与仿真	143
任务 4 实现单片机双机通信	145
思考题与习题	156
项目 7 数字电压表的设计与仿真	158
任务 1 学习 AT89C51 系统功能扩展	158
任务 2 A/D 转换器与 AT89C51 的接口技术	172
任务 3 D/A 转换器与 AT89C51 的接口技术	176
任务 4 数字电压表的设计与仿真	180
思考题与习题	185
项目 8 单片机典型应用系统开发	187
任务 1 单片机应用开发方法	187
任务 2 典型单片机应用系统开发实例	192
附录	214
附录 A ASCII 表	214
附录 B AT89C51 单片机指令表	215
参考文献	218

项目 1 基础知识和工具软件的使用

【学习目标】

- ① 掌握单片机中常用的数制和码制；
- ② 掌握单片机软件开发平台 KEIL 软件应用方法；
- ③ 掌握单片机硬件仿真平台 PROTEUS 软件应用方法。

任务 1 熟悉计算机中的数制与编码

计算机处理的各式各样的信息，本质上可归为两类：一类是数码，另一类是代码。无论是数码还是代码，均以二进制数的形式表示。下面主要介绍数制与码制的基本知识。

1.1.1 数制

计数体制简称为数制，是人们对数量计数的一种统计规律。按进位的原则进行计数的方法称为进位计数制，日常生活中最常用的是十进制数，计算机中采用的是二进制数，为了书写和阅读方便，还使用十六进制数。

数制包含多位数中每一位数的构成方法以及进位规则两个内容，其中涉及“基”和“权”这两个概念。基是某种数制所使用的数码的个数，例如十进制数使用的数码有 0~9 十个数字，显然十进制数的基是 10。权则表示多位数中每一位所具有的值的大小，例如在 666.6 这个数中，从左到右每一位的值大小是： 6×10^2 、 6×10 、6、 6×10^{-1} ，这说明每一位的权是不一样的。

(1) 十进制数 (Decimal Number)

十进制数的特点有：基为 10；各位的权是以 10 为底的幂次 10^i ($i=0,1,2,\dots$)；遵守“逢十进一”的进位规则。

一个任意的十进制数 D (n 位整数) 可以表示为：

$$D = D_{n-1} \times 10^{n-1} + D_{n-2} \times 10^{n-2} + \dots + D_1 \times 10^1 + D_0 \times 10^0 = \sum_{i=0}^{n-1} D_i \times 10^i$$

这个式子称为十进制数的按权展开式。例如，数 4321 可以展开为：

$$4321 = 4 \times 10^3 + 3 \times 10^2 + 2 \times 10^1 + 1 \times 10^0$$

(2) 二进制数 (Binary Number)

二进制数每位上只有 0 和 1 两个数字，基是 2，使用“逢二进一”的计数规律，二进制数从小到大的计数顺序为：0, 1, 10, 11, 100, ...

二进制数的位权是 2^i ，一个任意二进制数 B (n 位整数) 的按权展开式是：

$$B = B_{n-1} \times 2^{n-1} + B_{n-2} \times 2^{n-2} + \dots + B_1 \times 2^1 + B_0 \times 2^0 = \sum_{i=0}^{n-1} B_i \times 2^i$$

例如，数 100011.11B 可以展开为：

$$100011B = 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 35D$$

常把 8 位二进制数称为字节 (Byte)，16 位二进制数称为字 (Word)，32 位二进制数称为双字 (Double Word)。

(3) 十六进制数 (Hexadecimal Number)

二进制的缺点是书写长，不便记忆和阅读。为此通常采用十六进制数弥补二进制数的不足。十六进制数不仅书写方便，便于阅读，而且非常容易和二进制数进行转换。因此，在编写汇编语言源程序时，人们都习惯使用十六进制数。

十六进制数有 16 个计数数字，它们分别是：0~9、A、B、C、D、E 和 F。这里的 A~F 对应着十进制中的 10~15。表 1-1 列出了和十进制数 0~15 相应的二进制数和十六进制数。

表 1-1 十进制、二进制和十六进制对应表

十进制 D	二进制 B	十六进制 H	十进制 D	二进制 B	十六进制 H
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

通常在一个数的后面加上一个英文字母来表明它的数制形式，二进制数后跟 B，十进制数后跟 D，十六进制数后跟 H。十进制数的标志经常省略。十六进制数如果是字母打头，则需在前面加一个 0，如 0A4H 是一个十六进制数，它的按权展开式是：

$$0A4H = 10 \times 16^1 + 4 \times 16^0 = 164D$$

1.1.2 数制的转换

各种数制之间都存在着一定的转换关系，使用计算机时正确掌握和灵活运用这些关系是十分必要的。

(1) 二进制与十进制的相互转换

① 二进制数转换为十进制数 将二进制数按权展开相加之和即是等值的十进制数。例如：

$$1101101B = 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 = 109D$$

因此，1101101B=109D。

② 十进制数转换为二进制数 把十进制整数转换为二进制整数，采用“除 2 取余”法。例如，把 21 转化为二进制数的过程如下：

2	21	余数	低位
2	10	1	
2	5	0	
2	2	1	
2	1	0	
0	0	1	高位

所以，21D=10101B。

(2) 二进制与十六进制的相互转换

转换的基本准则是 4 位二进制数对应着 1 位十六进制数。由右向左每 4 位分为一组，不

足4位时,在前面补0。然后将每一组用对应的十六进制数代替即可。

【例 1-1】 将二进制数 110111000100010B 转换成十六进制数。

解 按每4位进行分组

$$\begin{array}{cccc} 0110 & 1110 & 0010 & 0010 \\ \hline 6 & E & 2 & 2 \end{array}$$

结果 0110 1110 0010 0010B=6E22H

(3) 十六进制数转换为二进制数

将十六进制数转换为对应的二进制数时,只要把十六进制的每一位用相应的4位二进制数代替即可。例如 2CH=0010 1100B。

1.1.3 原码、反码和补码

计算机中的二进制数根据其是否有符号位可以分为无符号数和有符号数。无符号数所有的位都是数值位,每一位都有权值,8位二进制无符号数的数值范围是0~255。有符号数是用符号位和数值部分一起编码来表示,有符号数的最高位是符号位,通常正数的符号位用“0”表示,负数的符号位用“1”表示。

在计算机中,有符号数的表示方法有原码、反码和补码三种形式。

(1) 原码

最高位是符号位,数值部分表示这个数的绝对值,这种表示方法称为原码。例如

$$[+105]_{\text{原码}} = 0\ 1101001B$$

$$[-105]_{\text{原码}} = 1\ 1101001B$$

两个数的数值位相同,而符号位相反。

(2) 反码

正数的反码和原码相同。负数的反码符号位不变,数值位为它的原码按位取反。例如

$$[+105]_{\text{反码}} = 0\ 1101001B$$

$$[-105]_{\text{反码}} = 1\ 0010110B$$

(3) 补码

正数的补码和原码相同。负数的补码符号位不变,数值位为它的补码加1。例如

$$[+105]_{\text{补码}} = 0\ 1101001B$$

$$[-105]_{\text{补码}} = 1\ 0010111B$$

8位二进制数的模是 $2^8 = 256 = 1\ 000\ 0000B$,如果两个8位数的和等于模 2^8 ,称这两个数互为补数。上述的两个数 01101001B 和 10010111B 就是一对补数。

引入补码的概念是为了将减法运算变成加法运算。根据两数互补的原理,减去一个数相当于加上这个数的补数。例如对于一个减法算式

$$X = 78 - 45$$

可以转换成加法运算。因为

$$[78]_{\text{补码}} = 01111000B, [-45]_{\text{补码}} = 10111011B, \text{所以}$$

$$[X]_{\text{补码}} = [78]_{\text{补码}} + [-45]_{\text{补码}} = 01111000B + 10111011B = 00110011B$$

结果是 $[X]_{\text{补码}} = 00110011B$,即 $X = 33$ 。

8位二进制有符号数的原码与补码的对应关系如表 1-2 所示。由表中可知用补码表示时 8位有符号数的范围是-128~127。

表 1-2 原码与补码对照表

二进制数	十六进制数	十进制	
		原码表示	反码表示
00000000B	00H	+0	+0
00000001B	01H	+1	+1
⋮	⋮	⋮	⋮
01111111B	7FH	+127	+127
10000000B	80H	-0	-128
10000001B	81H	-1	-127
⋮	⋮	⋮	⋮
11111110B	FEH	-126	-2
11111111B	FFH	-127	-1

1.1.4 常用编码

在计算机中要用二进制代码来表示各种信息（如数字、字母和标点符号等），把将这些信息转换成二进制代码的过程叫做编码。编码有多种不同的方案，即有多种码制，下面介绍常用的 8421BCD 码和 ASCII 码。

（1）8421BCD 码

计算机只能识别二进制数，但是人们却习惯用十进制数，为便于人机联系，通常采用具有二进制形式的代码来表示十进制数。这种编码方法就是将十进制数的每一位用四位二进制代码表示，通常称为二-十进制编码，简称 BCD（Binary Coded Decimal）码。

BCD 码有多种编码方案，最常用的是 8421BCD 码。表 1-3 列出了 8421BCD 码与十进制数 0~9 之间的对应关系。BCD 码是比较直观的，只要熟悉了它的 10 个编码，可以很方便地实现十进制数与 BCD 码之间的转换。

表 1-3 8421BCD 编码表

十进制数	8421BCD 码	十进制数	8421BCD 码
0	0000	5	0101
1	0001	6	0110
2	0010	7	0111
3	0011	8	1000
4	0100	9	1001

例如：

$$13 = (0001\ 0011)_{8421}$$

$$0100\ 1001\ 0110\ 0111)_{8421} = 4967$$

（2）字符的编码——ASCII 码

ASCII（American Standard Code for Information Interchange）码，美国标准信息交换码的简称，是计算机中应用最广泛的一种字符编码，用来识别数字、字母、通用符号、控制符等信息。ASCII 码字符表参见附录 A。

ASCII 码用 7 位二进制编码，能表示 128 个字符，其中包括 0~9 共 10 个十进制数码，26 个英文字母的大、小写形式，一些专用的可打印字符以及非打印字符等。在微型机中，

通常用1个字节（8位）来表示一个ASCII字符，其中，低7位为字符的ASCII码值，最高位或者为0，或者用于其他的功能，比如在通信系统中可用作奇偶校验位。

任务2 学会使用硬件开发平台 KEIL

Keil 公司是一家业界领先的微控制器（MCU）软件开发工具的独立供应商。Keil μ Vision4 是德国 KeilSoftware 公司出品的 51 系列兼容单片机软件开发系统。集成开发环境包含：编译器、汇编器、实时操作系统、项目管理器、调试器。2009 年发布的 Keil μ Vision4 引入灵活的窗口管理系统，使开发人员能够使用多台监视器，并提供了视觉上的表面对窗口位置的完全控制的任何地方。新的用户界面可以更好地利用屏幕空间和更有效地组织多个窗口，提供一个整洁、高效的环境来开发应用程序。

在介绍 Keil 软件操作方法之前，先新建一个文件夹“D:\mytest”，用来保存相关操作所产生的文件。这是一个良好的习惯，这样做可以避免和别的文件相混淆。

（1）启动 Keil μ Vision4

启动 Keil μ Vision4 后，进入到软件的应用界面，如图 1-1 所示。可以看到三个主要窗口：工程窗口、源程序编辑窗口（缺省文件名是 Text1）和编译输出窗口。

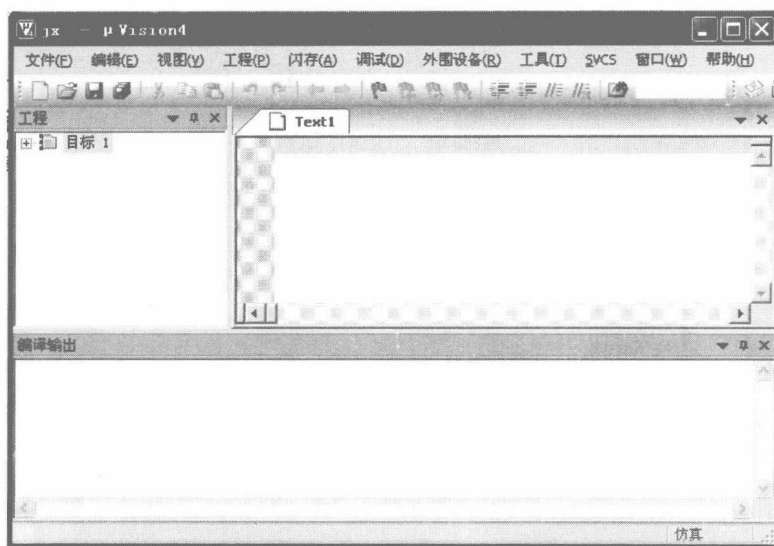


图 1-1 Keil μ Vision4 工作界面

（2）新建工程项目及项目设置

① 单击菜单命令“工程”→“新建 μ Vision4 工程”，弹出“Create New Project”对话框。

② 在“Create New Project”对话框中，文件保存位置选择“D:\mytest”，输入文件名“test”，如图 1-2 所示。单击保存，则新建的工程 test.uvproj 保存在“D:\mytest”文件夹中。

③ 保存工程后，会弹出单片机型号选择对话框（图 1-3）。选择 Atmel 公司的 AT89C51 单片机，单击确定。

④ 项目文件建立好之后，通常要对项目进行设置。单击菜单命令“工程”→“为目标



图 1-2 新建工程对话框

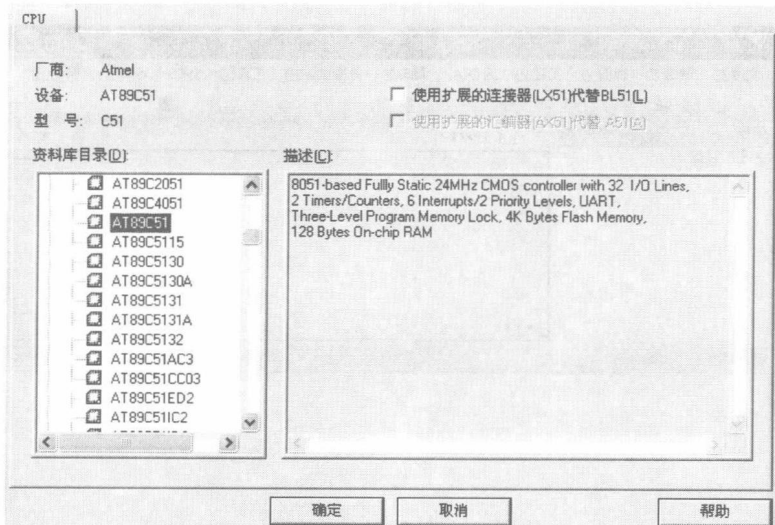


图 1-3 选择单片机型号对话框

‘目标 1’ 设置选项”，弹出设置选项对话框，选项界面有多个标签，大多数均使用默认设置。单击“输出”标签，界面如图 1-4 所示，在产生 HEX 文件复选框内打√，HEX 格式选择 HEX-80，其他信息保持原来状态。经过此操作后，对源程序进行编译时就会形式目标代码，并以 test.hex 文件保存在“D:\mytest”文件夹中。

(3) 源程序文件的编译

① 新建文件。单击菜单命令“文件”→“新建”，建立一个新文件。

② 保存文件。单击菜单命令“文件”→“保存”，弹出保存文件对话框，如图 1-5 所示。文件保存位置选择“D:\mytest”文件夹，文件名输入“myfile.asm”。注意文件扩展名必须输入，用汇编语言编写的源程序，扩展名是“.asm”，用 C51 编写的源程序，扩展名是“.c”。

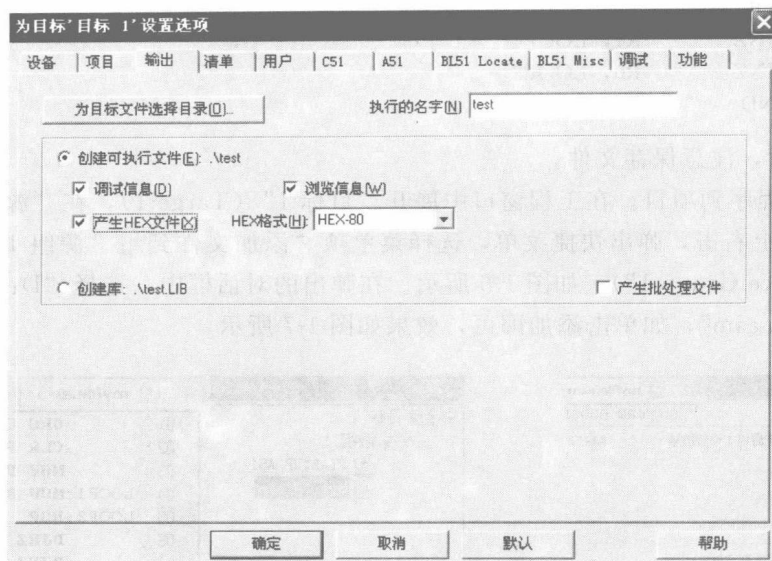


图 1-4 目标设置选项

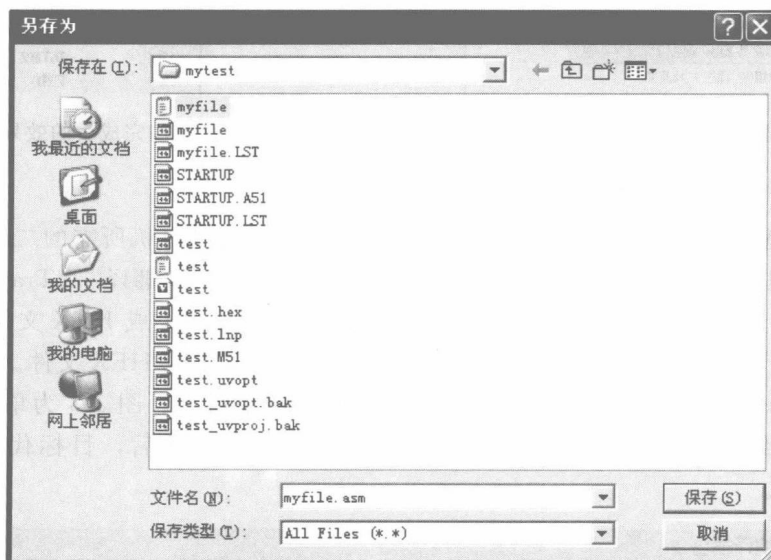


图 1-5 保存文件对话框

③ 编辑源程序。在文件窗口中编辑下面一段源程序。

```

ORG      0000H
CLR      P1.0
MOV      R0, #00H
LOOP1:  MOV      R1, #00H
LOOP2:  NOP
        DJNZ     R1, LOOP2
        DJNZ     R0, LOOP1
        SETB     P1.0
        MOV      R0, #00H
LOOP3:  MOV      R1, #00H

```

```

LOOP4:  NOP
        DJNZ     R1, LOOP4
        DJNZ     R0, LOOP3
        END

```

编辑完成后，注意保存文件。

④ 添加源程序到项目。在工程窗口中展开“目标 1”（Target1），在“源组 1”（Source Group 1）目录上右击，弹出快捷菜单，选择菜单项“添加文件到组‘源组 1’”（Add Files to Group “Source Group 1”），如图 1-6 所示。在弹出的对话框中，选择“D:\mytest”文件夹中的“myfile.asm”，如单击添加即可，效果如图 1-7 所示。



图 1-6 添加文件到项目

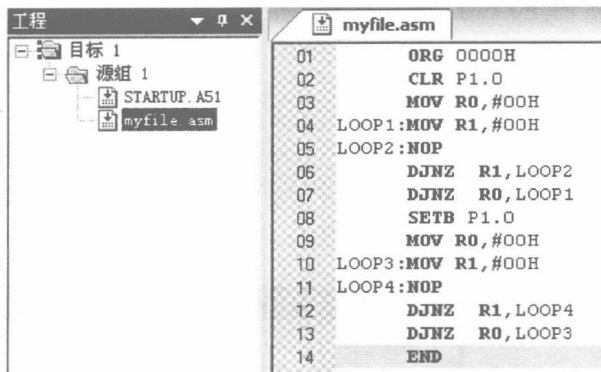


图 1-7 文件添加完成后的效果

(4) 编译、调试运行

① 编译源程序。这个过程是将用户编写的源程序转换成计算机所需的二进制目标程序。单击菜单“工程”，子菜单中与编译源程序有关的菜单有 3 个：“翻译”（Translate）命令将编译当前文件；“编译”（Build target）命令将编译当前文件并生成 HEX 文件；“编译全部文件”（Rebuild all target files）命令将重新编译所有文件并生成 HEX 文件。如果源程序有语法错误，编译将不能成功，系统在编译输出窗口中提示错误。图 1-8 为单击“工程”→“编译”命令，编译成功后编译窗口所显示的状态。编译成功后，目标代码保存在 myfile.hex 文件中。

编译输出

```

*** WARNING L2: REFERENCE MADE TO UNRESOLVED EXTERNAL
SYMBOL:  ?C_START
MODULE:  STARTUP.obj (?C_STARTUP)
ADDRESS: 0020H
Program Size: data=9.0 xdata=0 code=37
creating hex file from "test"...
"test" - 0 Error(s), 3 Warning(s).

```

图 1-8 编译并生成 HEX 文件

② 调试运行。单击菜单命令“调试”（Debug）→“启动/停止仿真调试”（Start/Stop Debug session）进入调试模式。这时单击“调试”菜单下的“单步步入”（Step）命令将进入单步运行调试；单击“运行”（Go）命令，将连续运行直到一个断点。

③ 内部资源窗口。在调试模式下，可以打开各类内部资源窗口，观察程序运行对存储

器、寄存器、I/O端口的影响。在“视图”(View)主菜单的下拉菜单中,可以打开寄存器、存储器、堆栈等窗口。在“外围设备”(Peripherals)主菜单的下拉菜单中,可以打开中断、并行I/O接口、定时器等窗口。

图1-9所示是存储器窗口,用于观察各类存储器的状态。在地址栏中输入类型和地址就可显示相应的存储单元状态。常用的类型是I(内部RAM)、C(ROM)、X(外部RAM)。当地址栏中输入“C:0000H”时,显示的是从0000H开始的ROM区存储单元的内容,一般是用户程序代码。

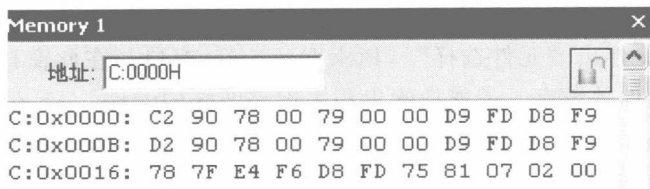


图 1-9 存储器观察窗口

任务3 学会使用硬件仿真平台 PROTEUS

PROTEUS 软件是英国 Labcenter electronics 公司出版的 EDA 工具软件(该软件中国总代理为广州风标电子技术有限公司)。它不仅具有其他 EDA 工具软件的仿真功能,还能仿真单片机及外围器件。它是目前最好的仿真单片机及外围器件的工具。

(1) 启动 PROTEUS ISIS

PROTEUS 软件启动后工作界面如图 1-10 所示,主要由菜单栏、工具栏、仿真原理图

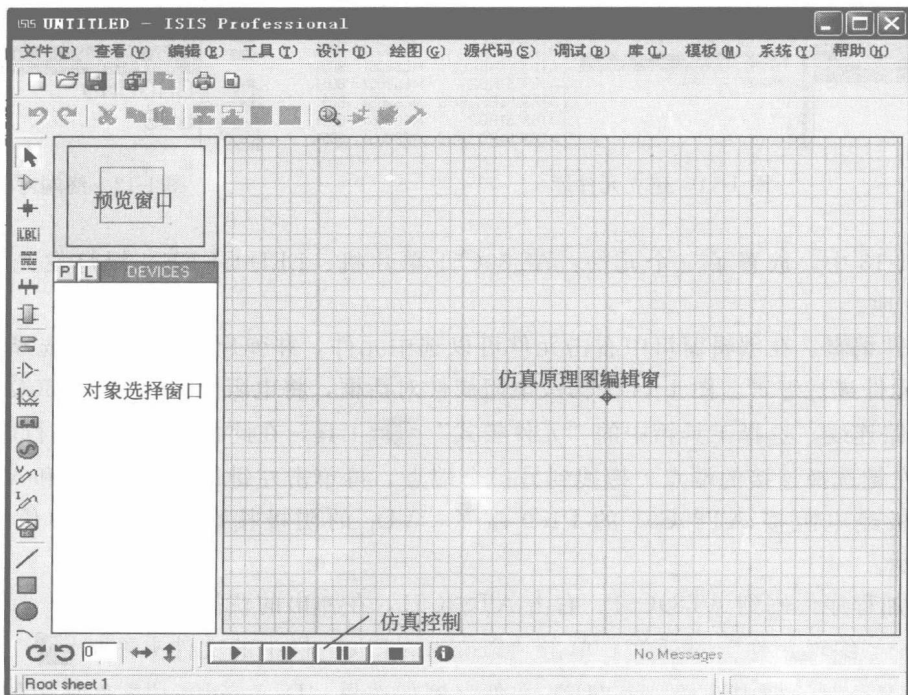


图 1-10 PROTEUS 工作界面

编辑窗口、预览窗口、对象选择窗口等部分组成。

(2) 文件管理

在主菜单“文件”(File)的下拉菜单中有“新建设计”(New Design)、“打开设计”(Load Design)、“保存设计”(Save Design)等命令。单击“文件”→“新建设计”菜单命令就可以新建一个设计,再单击“文件”→“保存设计”菜单命令将文件保存起来。例如将设计保存在文件夹“D:\mytest”中,并命名为 mydesign. DSN。

(3) 建立仿真模型

① 建立元件库。选择工具箱中的“元件模式”工具,单击对象选择窗口上方的“从库中选择”按钮,弹出“元件选择”(Pick Devices)窗口。在此窗口的“关键字”文本框中输入要添加的元件关键字,就能搜索出与关键字匹配的元件。

例如输入“AT89C51”,在出现的元件中双击所需的器件,该器件就被添加到设计文档的元件库中,如图 1-11 所示。

② 放置元件。将所需的元件添加到元件库后,就可从元件库中选择元件放置到编辑窗口。操作方法是在元件库中选中要放置的元件,然后在编辑窗口单击鼠标左键。

选择工具箱中的“终端模式”,常用的终端就显示在文件库中,如图 1-12 所示,这时可以放置电源、地等终端对象。



图 1-11 建立元件库

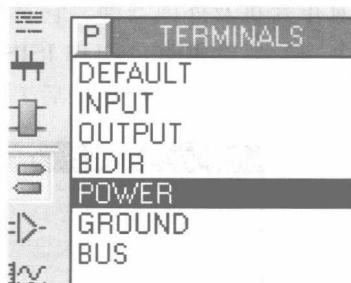


图 1-12 终端对象

在图 1-13 中,放置了三个元件:AT89C51 单片机、LED-GREEN 绿灯发光二极管和 GROUND 地。

③ 元件编辑。在编辑窗口中右击元件可以选中元件,并弹出快捷菜单,选择相关的菜单项可对元件进行编辑。图 1-14 为修改电阻属性对话框,将电阻阻值设置为 330Ω 。

④ 电路连线。选择工具箱中的“元件模式”工具,在此模式下可以进行电路连接。单击鼠标左键选择连接的端点,拖拽到另一个端点,再单击左键即可。将 R_1 和 VD_1 串联, VD_1 的阳极经电阻接 AT89C51 的 P1.0 端子, VD_1 的阴极按地,连接好的仿真电路如图 1-15 所示。

⑤ 添加程序代码到 AT89C51。右击 AT89C51,在弹出的快捷菜单中选择“编辑属性”命令,打开“编辑元件”对话框。单击“Program File”栏中的文件夹图标,在弹出的“选择文件名”对话框中选择一个 HEX 文件(例如选择“D:\mytest”文件夹中的 test. hex 文件),单击“打开”,返回到编辑元件对话框,如图 1-16 所示。

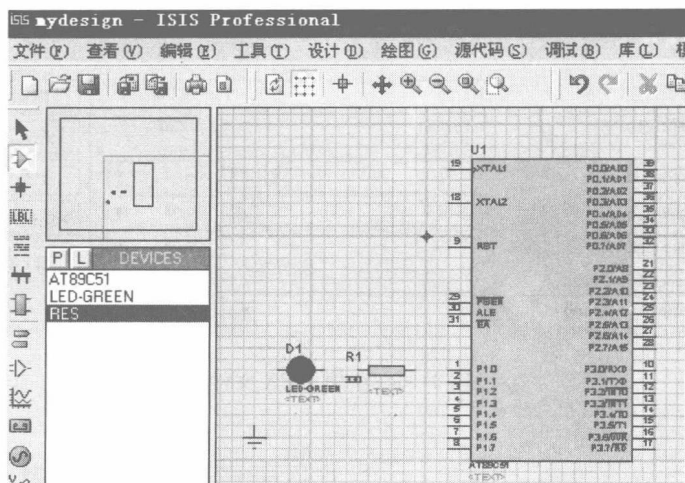


图 1-13 放置元件

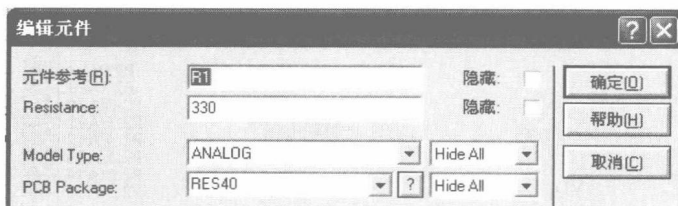


图 1-14 设置电阻属性

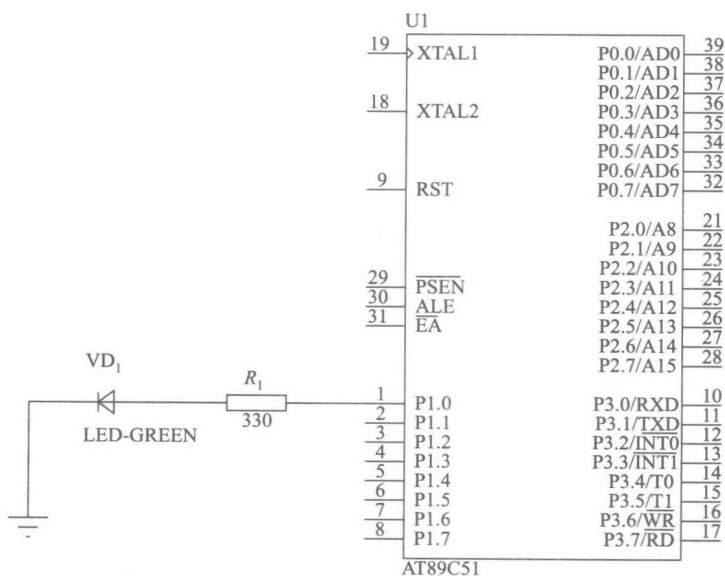


图 1-15 连接电路

⑥ 启动仿真。点击仿真控制工具箱  中的运行、暂停、停止工具，可以对仿真进行控制，观察仿真效果，如图 1-17 所示。仿真电路中，红色小方块表示该点的电平是高电平，蓝色小方块表示该点的电平是低电平，灰色小方块表示该点的电平不能确定。