

21世纪高职高专规划教材

电子信息
工学结合模式
系列教材

单片机 应用系统设计 与实现

王先彪 主编

清华大学出版社

电子信息
工学结合模式
系列教材

21世纪高职高专规划教材

单片机 应用系统设计 与实现

王先彪 主编

清华大学出版社
北京

内 容 简 介

本书以项目为导向,采用工程项目开发模式,通过流水灯、8 路抢答器、篮球记分牌、万年历和门禁系统 5 个典型案例,由浅入深、循序渐进,逐步介绍软件 Proteus 和 Keil μ Vision 的使用、单片机结构、LED 显示技术、汇编程序设计、中断系统、键盘接口技术、定时器/计数器、LCD 显示技术、存储器扩展技术以及串行通信技术。

本书可作为高职院校电子信息类专业的教材,也可供单片机应用系统设计和开发人员以及其他院校电子类专业师生参考、学习使用。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

单片机应用系统设计与实现/王先彪主编. —北京:清华大学出版社,2014

21 世纪高职高专规划教材. 电子信息工学结合模式系列教材

ISBN 978-7-302-33259-6

I. ①单… II. ①王… 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 166030 号

责任编辑:王剑乔

封面设计:傅瑞学

责任校对:刘 静

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:13

字 数:298 千字

版 次:2014 年 1 月第 1 版

印 次:2014 年 1 月第 1 次印刷

印 数:1~2700

定 价:28.00 元

产品编号:041922-01

前言

发现问题、解决问题是当今企业对高素质实用型技术人才的基本需求。打破传统的学科体系课程结构,建立基于项目导向、任务驱动、教学做合一的教学方法,是当前高职院校教学改革的方向。培养善于学习、发现问题、解决问题,动手能力强,能很快适应未来工作岗位的技术人才,是编写本教材的目的所在。

与同类教材相比,本教材具有以下特点:

(1) 以项目为导向,从需求分析、硬件电路设计、软件代码编写到产品调试与完善,完全符合企业工程项目开发流程。

(2) 每个项目既相互独立,形成完整产品,又相互关联,由浅入深,逐步延伸、拓展。

(3) 以 Proteus 作为应用系统设计与仿真平台,在做中学、学中做,实现了从产品概念到设计完成的全过程训练,一气呵成,大大降低了产品的开发成本,提高了学生的学习积极性。

(4) 高级语言与汇编语言并重。前三个项目既给出了汇编程序,也给出了 C 语言程序,在尽量理解汇编程序的基础上,注重突出 C 语言的优越性。

(5) 所有项目的程序均已通过测试,可直接运行,也可在此基础上加以完善和创新。

本书由王先彪主编,唐扬波任副主编,刘冬香主审。参编者分工如下:王先彪编写项目一与项目二;唐扬波编写项目五的关联知识以及项目三;刘继增(罗定职业技术学院)编写项目四的关联知识;陈伟杭、朱然辉测试并编写项目四的程序部分;梁水林测试并编写项目五的程序部分;朱会东(广东德生科技有限公司技术总监)负责全书策划,并对如何培养行业实用型人才提供了宝贵的经验。

本书提供配套电子课件及程序源代码,登录 <http://www.cedu.cc> 可免费下载使用。

由于编者水平所限,书中难免存在错误与不足,恳请各位读者批评指正。编者邮箱: wangxb64@163.com 或 electybo@163.com。

编者
2013 年 8 月

目 录

项目一 流水灯	1
1.1 项目设计	1
1.1.1 项目要求	1
1.1.2 电路设计	1
1.1.3 程序设计	2
1.1.4 项目实施	4
1.2 关联知识	5
1.2.1 仿真工具 Proteus	5
1.2.2 仿真工具 Keil μ Vision	11
1.2.3 Proteus 与 Keil μ Vision 的联合仿真	19
1.3 项目拓展	22
1.3.1 单片机最小系统的开发过程	22
1.3.2 流水灯的功能拓展	23
项目二 抢答器	24
2.1 项目设计	24
2.1.1 项目要求	24
2.1.2 电路设计	24
2.1.3 程序设计	26
2.1.4 项目实施	29
2.2 关联知识	30
2.2.1 单片机结构	30
2.2.2 LED(Light Emitting Diode)技术	41
2.2.3 汇编程序设计	46
2.3 项目拓展	49
2.3.1 单片机应用系统常见故障与调试	49
2.3.2 查表程序的编写	50
2.3.3 抢答器的功能拓展	51

项目三 篮球记分牌	52
3.1 项目设计	52
3.1.1 项目要求	52
3.1.2 电路设计	52
3.1.3 程序设计	53
3.1.4 项目实施	62
3.2 关联知识	64
3.2.1 中断系统	64
3.2.2 键盘接口技术	70
3.3 项目拓展	76
3.3.1 按键处理的常用技巧	76
3.3.2 篮球记分牌的功能拓展	78
项目四 万年历	79
4.1 项目设计	79
4.1.1 项目要求	79
4.1.2 电路设计	79
4.1.3 程序设计	81
4.1.4 项目实施	91
4.2 关联知识	93
4.2.1 定时器/计数器	93
4.2.2 LCD(Liquid Crystal Display)技术	98
4.2.3 数字温度传感器 DS18B20	104
4.3 项目拓展	107
4.3.1 模/数与数/模转换技术	107
4.3.2 万年历的功能拓展	117
项目五 门禁系统	118
5.1 项目设计	118
5.1.1 项目要求	118
5.1.2 电路设计	118
5.1.3 程序设计	119
5.1.4 项目实施	143
5.2 关联知识	145
5.2.1 存储器扩展技术	145
5.2.2 串行通信技术	156
5.3 项目拓展	170

5.3.1 单片机和 RS-485 通信接口	170
5.3.2 单片机的 I/O 口模拟串口	173
5.3.3 门禁系统的功能拓展	174
附录 A 特殊功能寄存器	175
附录 B 单片机伪指令	176
附录 C 单片机指令系统	178
C.1 单片机指令格式	178
C.2 寻址方式	179
C.2.1 立即数寻址	179
C.2.2 直接寻址	179
C.2.3 寄存器寻址	180
C.2.4 寄存器间接寻址方式	180
C.2.5 位寻址	181
C.2.6 基址寄存器加变址寄存器间接寻址	181
C.2.7 相对寻址	181
C.3 指令系统	182
C.3.1 指令分类	182
C.3.2 数据传送类指令	183
C.3.3 算术运算指令	186
C.3.4 逻辑运算指令	189
C.3.5 控制转移指令	191
C.3.6 位操作指令	195
参考文献	199

流水灯

1.1 项目设计

1.1.1 项目要求

设计一个流水灯控制系统,符合以下要求:①采用单片机 AT89C51 进行控制;②利用 8 个发光二极管完成显示;③能够控制二极管按顺序亮和灭,产生“流水”移动效果。

通过本项目的设计开发,学生应学会利用 Keil μ Vision 创建项目、编写源程序、调试程序并生成 HEX 文件;学会利用 Proteus 创建设计文件、构建电路原理图、加载 HEX 文件并仿真。

1.1.2 电路设计

1. 设计思路

流水灯控制系统主要包括单片机控制和发光二极管显示两个模块,如图 1-1 所示。

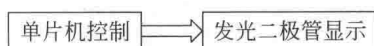


图 1-1 流水灯控制系统设计思路

2. 器件清单

流水灯控制系统器件清单如表 1-1 所示。

表 1-1 流水灯控制系统器件清单

功能块	器件标号	器件名称	KEYWORDS	参数值	数量
主控	U1	微处理器	AT89C51	—	1
发光	D1~D8	发光二极管	LED	LED-YELLOW	8
	$R_1 \sim R_8$	电阻	RES	270 Ω	8
时钟	X1	晶振	CRYSTAL	12MHz	1
	$C_1 \sim C_2$	电容	CAP	30pF	2
复位	R_9	电阻	RES	10k Ω	1
	C_3	电容	PCELECT10U50V	10 μ F	1

注:“KEYWORDS”项中的关键词可用来在构建电路原理图时快速提取所需要的器件。

3. 电路原理

单片机将数据传送到输出口 P2,以电平信号的形式输出。P2 引脚输出“0”时,输出引脚为低电平,由于发光二极管阳极接正电源端,对应的发光二极管有电流流过,二极管发光;引脚输出“1”时,输出引脚为高电平,对应的发光二极管没有电流流过,二极管不发光。只要对输出数据进行移位,对应的发光二极管便会相应地移动、变化,在人眼中形成灯在“流动”的感觉,成为流水灯。

流水灯硬件电路图如图 1-2 所示。

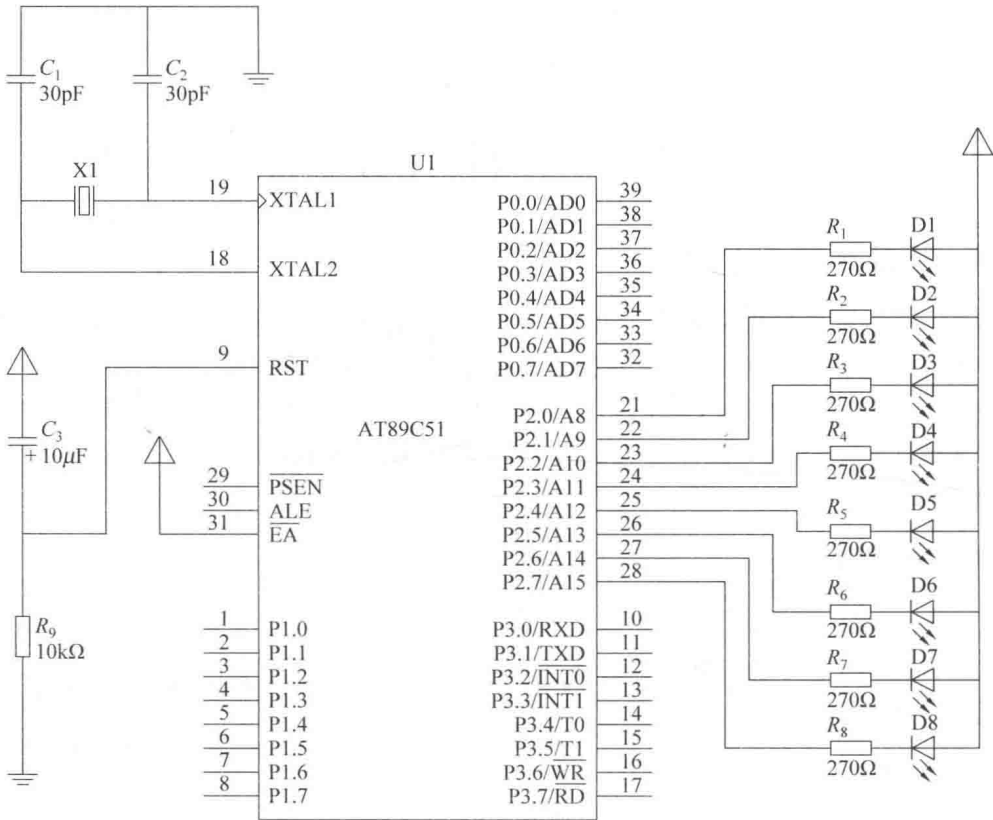


图 1-2 流水灯硬件电路图

1.1.3 程序设计

1. 程序流程

流水灯控制系统程序流程图如图 1-3 所示。

2. 程序代码

汇编语言是一种用文字助记符表示机器指令的符号语言,是最接近机器的一种语言。其主要优点是占用资源少、程序执行效率高。对于不同的 CPU,其汇编语言可能有所差异,所以相对于 C 语言来说,汇编程序的移植性较差。换句话说,基于某种型号的单片机

设计的汇编程序,当用于另外一种型号的单片机时,可能需要修改汇编程序。

C 语言是一种结构化的高级语言,其优点是程序简洁(譬如,从下面的利用汇编代码和 C 代码实现延时功能的程序段就可以看出),可读性好,移植容易,是使用较为普遍的一种语言;缺点是占用资源较多,执行效率没有汇编程序高。

1) 汇编代码

为了让单片机输出 P2 口控制发光二极管的数据保留一段时间,需要在每次更新 P2 口的数据(即执行 MOV P2, A)之后,调用延时子程序(即 ACALL DELAY)。延时子程序的执行时间大约为 0.5s。因此,流水灯每 0.5s 移动一次。

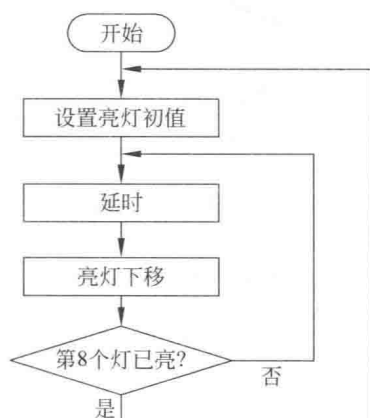


图 1-3 程序流程图

```

;*****
;主程序
;*****
        ORG      0000H                ;程序在 ROM 中的起始存放位置
BEGIN:
        MOV      A, #0FEH             ;给寄存器 A 赋初值 0FEH(=11111110B)
LOOP1:
        MOV      P2, A                ;将寄存器 A 的值发送至 P2 口
        ACALL    DELAY               ;调用软件延时子程序
        RL       A                   ;将 A 的值循环左移, 对应亮灯自上而下移动
        LJMP     LOOP1               ;反复执行
;*****
;延时子程序, 晶体振荡器为 12MHz, 约延时 0.5s
;*****
        ORG      0100H                ;延时子程序在 ROM 中的起始存放位置
DELAY:
        MOV      R5, #5               ;(R5)=5, 将 5 赋给寄存器 R5
DEL01:
        MOV      R6, #200             ;(R6)=200, 将 200 赋给寄存器 R6
DEL02:
        MOV      R7, #250             ;(R7)=250, 将 250 赋给寄存器 R7
DEL03:
        DJNZ     R7, DEL03             ;R7-1≠0 时继续执行本行指令
        DJNZ     R6, DEL02             ;R6-1≠0 时转至 DEL02
        DJNZ     R5, DEL01             ;R5-1≠0 时转至 DEL01
        RET                          ;返回主程序
  
```

提示:

- (1) 可以通过改变赋给寄存器 R5、R6 和 R7 的值来调整延时时间;
- (2) 存入寄存器的值不能超过 255, 即 0FFH;
- (3) 若十六进制立即数以字母开头, 需要在前面加上“0”, 如 0FFH。

2) C 代码

```

#include "reg51.h"           //包含头文件
//-----
//延时子函数
//-----
delay()
{
    int i, j;                //定义变量
    for(i=0;i<255;i++)        //第 1 层循环, 循环变量的值不能超过 255
        for(j=0;j<150;j++)    //第 2 层循环
            ;                  //空操作
}
//-----
//主函数
//-----
main()
{
    int i;
    while(1)                  //反复运行
    {
        P2=0xFE;              //赋初值, 即低位灯亮
        for(i=0;i<8;i++)      //循环点亮 8 个灯
        {
            delay();           //调用延时
            P2= (P2<<1) | 0x01; //位左移, 对应亮灯下移。注意与汇编代码中对应语句的区别
        }                     //C 语言移位后无法在最低位自动补“1”, 需要 0x01
    }
}

```

1.1.4 项目实施

项目实施步骤如下所述。

第 1 步: 在 Keil 环境中创建项目文件, 保存为 lsd. uv2(后缀. uv2 可由系统自动生成)。创建新的程序文件, 输入 1.1.3 小节中所列流水灯的源程序, 保存为 lsd. asm 或 lsd. c, 后缀视汇编代码还是 C 代码而定。

注意: 后缀. asm 或. c 必须由设计者输入, 系统不会自动生成。

第 2 步: 在 Keil 环境中编译源程序 lsd. asm 或 lsd. c, 生成单片机可执行目标文件 lsd. hex。

第 3 步: 利用 Proteus 仿真软件绘制 1.1.2 小节所示的仿真电路图 1-2, 保存为 lsd. dsn。后缀. dsn 由系统自动生成。

第 4 步: 在 Proteus 仿真软件中选择单片机芯片, 并加载可执行文件 lsd. hex。

第 5 步: 在 Proteus 仿真软件中单击仿真进程控制按钮中的执行按钮 , 进行仿真, 效果如图 1-4 所示。

第 6 步: 如果仿真不符合要求, 检查电路设计和程序代码, 直到符合要求为止。

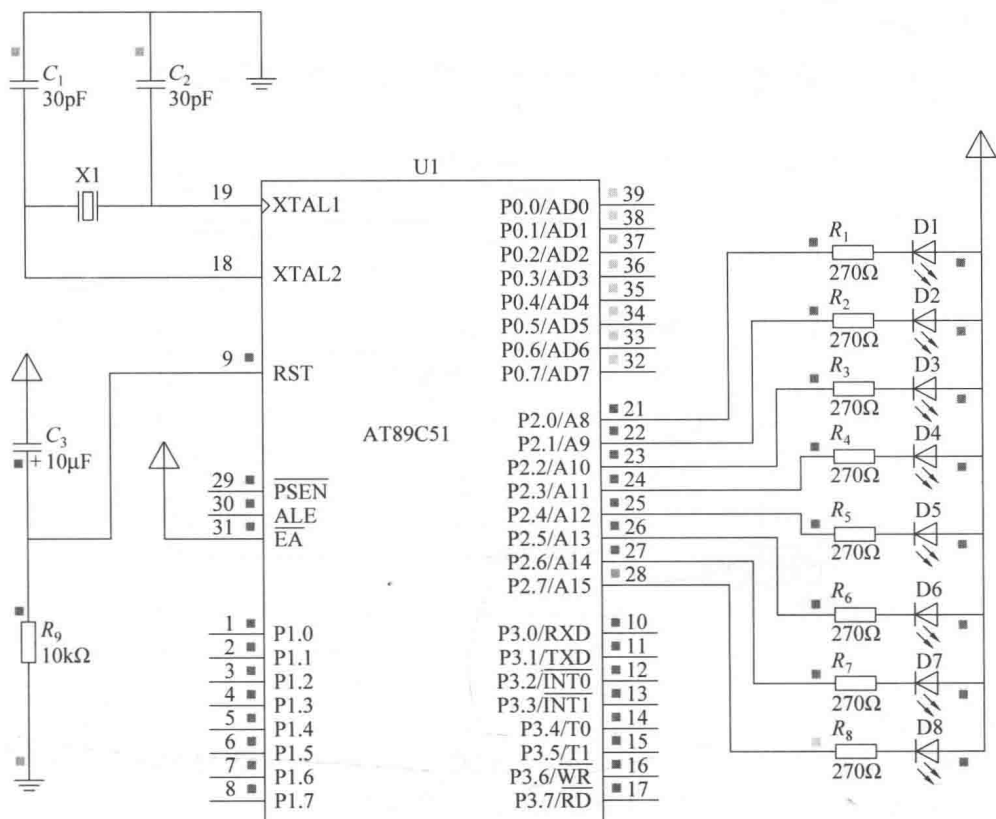


图 1-4 流水灯仿真效果图

1.2 关联知识

1.2.1 仿真工具 Proteus

Proteus 软件是英国 Labcenter Electronics 公司开发的电路分析与仿真软件。它除了具有原理图编辑、印制电路板自动或人工布线及电路仿真功能外,最大的特色是其电路仿真是交互的、可视化的。

Proteus 由 ISIS 和 ARES 两个软件构成,其中 ISIS 是电子系统仿真平台,ARES 是布线编辑软件。在不需要硬件设备投入的情况下,Proteus 可以建立完整的模拟电子、数字电子及单片机应用的学习设计开发环境。这里主要介绍 Proteus ISIS 在仿真单片机应用系统中的应用,包括原理图的编辑和整个单片机应用系统的仿真调试。其他与模拟电子、数字电子设计有关的功能可以查阅相关资料。

1. 创建设计文件

1) 启动 Proteus

双击桌面上的 ISIS 7 Professional 图标或者单击屏幕左下方【开始】→【程序】→【Proteus 7 Professional】→【ISIS 7 Professional】, 出现如图 1-5 所示屏幕, 表明进入

Proteus ISIS 集成环境。



图 1-5 仿真软件 Proteus 人—机界面

2) 创建新设计

启动 Proteus 软件进入 Proteus ISIS 编辑环境后,选择【File】→【New Design】命令,弹出如图 1-6 所示的新设计模板选择对话框。通常选择默认选项,然后单击“OK”按钮。

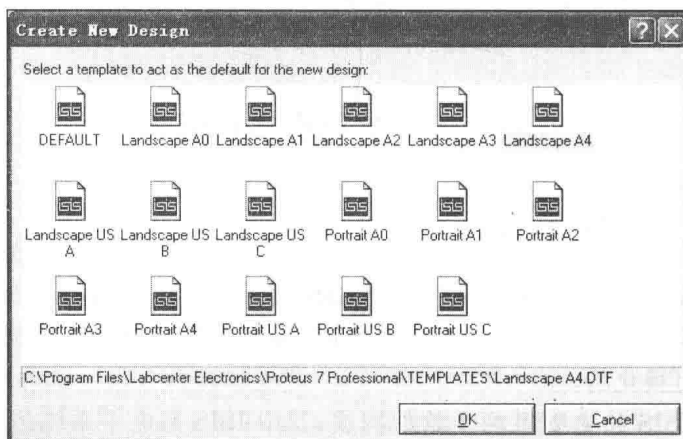


图 1-6 新设计模板选择

3) 保存新设计

选择【File】→【Save Design】命令或直接单击快捷按钮,再选择合适的文件夹和文件名(例如“lsd”),保存新设计。保存后,新设计文件名将显示在 Proteus 窗口的标题栏。也可以先画原理图,再保存新设计文件。

注意: 设计文件的后缀.dsn 不需要用户自己录入,系统会自动生成。

2. 构建电路图

Proteus 编辑原理图的界面分为 3 个区域,如图 1-5 所示,分别为浏览区、原理图编辑区和器件列表区。浏览区用于器件形状浏览或整体原理图浏览,原理图编辑区用于编辑原理图,器件列表区用于显示原理图中要用到的器件。在画原理图之前,需要通过搜索查找并提取所需要的器件。被提取到的器件将在器件列表区显示出来。以下所示是部分常用快捷图标。

-  Component Mode: 器件选择;
-  Junction Dot Mode: 在原理图中标注连接点;
-  Wire Label Mode: 标注网络标号;
-  Text Script: 在电路中输入说明文本;
-  Buses Mode: 绘制总线;
-  Subcircuit Mode: 绘制子电路块;
-  Terminals Mode: 对象选择器列出输入/输出、电源、地等终端;
-  Device Pin Mode: 对象选择器将列出普通引脚、时钟引脚、反电压引脚和短接引脚等;
-  Graph Mode: 对象选择器列出各种仿真分析所需的图表;
-  Tape Recorder Mode: 当对设计电路分割仿真时采用此模式;
-  Generator Mode: 对象选择器列出各种激励源;
-  Voltage Probe Mode: 电压探针,电路进入仿真模式时可显示各探针处的电压值;
-  Current Probe Mode: 电流探针,电路进入仿真模式时可显示各探针处的电流值;
-  Virtual Instrument Mode: 对象选择器列出各种虚拟仪器。
-  2D Graphics Line Mode: 绘制直线(用于创建器件或表示图表时绘制线);
-  2D Graphics Box Mode: 绘制方框;
-  2D Graphics Circle Mode: 绘制圆;
-  2D Graphics Arc Mode: 绘制弧;
-  2D Graphics Closed Path Mode: 绘制任意形状图形;
-  2D Graphics Text Mode: 文本编辑,用于插入说明;
-  2D Graphics Symbol Mode: 用于选择各种符号器件;
-  2D Graphics Makers Mode: 用于产生各种标记图标;
-  Rotate Clockwise: 顺时针旋转 90° ;
-  Rotate Anti-Clockwise: 逆时针旋转 90° ;
-  X-Mirror: 水平镜像;
-  Y-Mirror: 垂直镜像。

1) 提取器件

Proteus ISIS 提供了强大的搜索器件的功能。选择【Library】→【Pick Device/Symbol】命令,或者单击  后再单击 ,出现如图 1-7 所示的器件选择对话框。

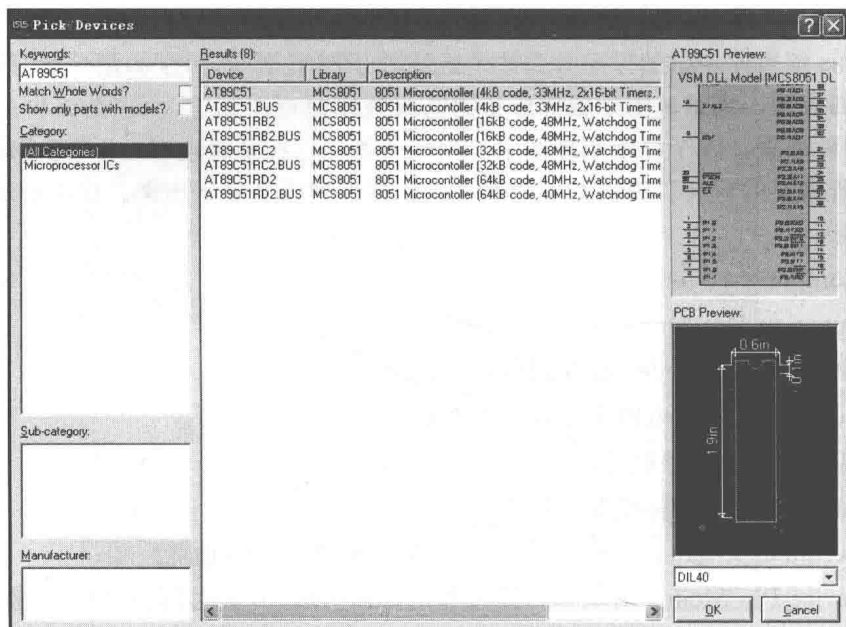


图 1-7 器件选择对话框

在左上角的搜索“Keywords”文本框中输入要查找的器件,例如“AT89C51”,在搜索结果“Results”区域就会显示相关的资料。双击需要的器件,则在原理图编辑环境中的器件列表区出现所选器件,如图 1-8 所示。

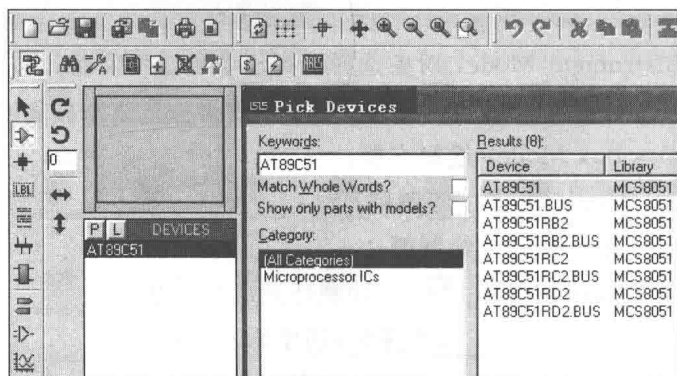


图 1-8 提取器件

用同样的方法找出原理图中需要的所有器件。

提示:

(1) 也可以单击需要的器件,然后单击右下角的“OK”按钮。采用这种办法,将关闭器件选择对话框图 1-7;若要选择下一个器件,需要重新打开此对话框。

(2) 在本书每一个项目的电路设计中,要用到的器件都在器件清单中的“KEYWORDS”栏目以大写加粗字母标出,在“Keywords”文本框中直接输入这些型号,就能很方便地找出需要的器件。

2) 放置器件

在器件列表区中选中器件,选中的器件在浏览区中将显示具体的形状和方位。最后在原理图编辑区中单击放置器件。

3) 删除器件

右击可以删除器件,同时删除该器件的所有连线。如果错删了器件,可以使用 Undo 命令恢复。

4) 拖动器件

用鼠标指向选中的器件并按住左键拖曳可以拖动器件。该方式不仅对整个器件有效,而且对器件中关联的 labels(指器件名称、参数)也有效。

5) 拖动器件标签

许多类型的器件附着有一个或多个属性标签。例如,每个器件有一个 reference 标签和一个 value 标签,可以很容易地移动这些标签使电路图看起来更美观。移动标签的步骤如下:

(1) 选中器件;

(2) 用鼠标指向标签并按住鼠标左键;

(3) 拖动标签到需要的位置。如果想要定位得更精确,可以在拖动时改变捕捉的精度(使用 F4、F3、F2 或 Ctrl+F1 键);

(4) 释放鼠标。

6) 调整器件大小

子电路、图表、线、框和圆可以调整大小。调整器件大小的步骤如下:

(1) 选中器件;

(2) 如果器件可以调整大小,器件周围会出现黑色小方块,称为“手柄”;

(3) 按住鼠标左键拖动这些“手柄”到新的位置,器件的大小即改变。

7) 调整器件的朝向

许多类型的器件可以调整朝向为 0° 、 90° 、 270° 、 360° 或通过 x 轴、 y 轴镜像。调整器件朝向的步骤如下:

(1) 选中器件;

(2) 单击  图标,器件逆时针旋转;右击  图标,器件顺时针旋转;

(3) 单击  图标,器件按 x 轴镜像;右击  图标,器件按 y 轴镜像。

8) 编辑器件属性

编辑器件属性的步骤是:用鼠标左键双击需要编辑属性的器件,将弹出器件编辑窗口,如图 1-9 所示。在图中,修改所需要的器件属性值。

9) 复制器件

复制器件的方法如下:

(1) 选中需要的器件;

(2) 单击图标 ;

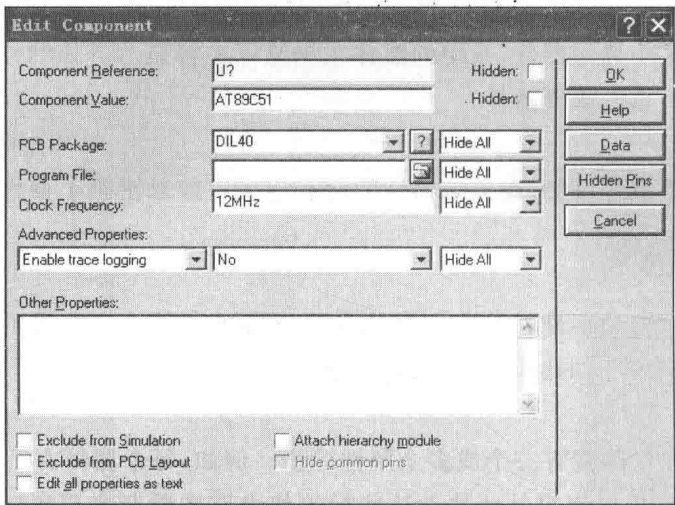


图 1-9 器件属性编辑对话框

- (3) 把复制的器件轮廓拖到需要的位置后，单击鼠标左键；
- (4) 右击完成复制。

10) 连线、绘制原理图

要在两个器件间连线，先单击第一个器件连接点，再单击另一个连接点，ISIS 自动将两个点连上。如果用户想自己决定走线路径，只需在想要的拐点处单击即可。线路路径器用来设置走线方法，单击【Tools】→【Wire Auto-Router(WAR)】命令，实现对 WAR 的设置。该功能默认是打开的。打开 WAR 是折线连线，关闭 WAR 是两点直接连线。

对具有相同特性的画线，可采用重复布线的方法：先画一条，再在器件引脚双击。假设要连接一个 8 字节 ROM 的数据线到单片机 P0 口，只要画出某一条从 ROM 数据线到单片机 P0 口的线，对于其余的连线，单击 ROM 器件的引脚即可。

提示：在“直线模式”下，按住 Ctrl 键，也可以放置斜线。

3. 加载程序

单击原理图中的单片机芯片，出现对话框如图 1-10 所示。在“Program File”文本框右侧，单击文件夹图标打开文件夹(此时会弹出一个文件夹对话框，其操作与所有应用软件打开文件夹对话框一样)，浏览用 Keil μVision 编译生成的 HEX 目标文件，例如本项目编译好的 LSD.HEX。单击“OK”按钮即可完成程序的加载。

4. 仿真

将目标程序加载到单片机芯片后，选择【Debug】→【Execute】命令对整个系统进行软、硬件全面仿真，也可通过快捷键中的键进行快捷仿真。

通过观察系统的运行情况是否符合预期指标，找出问题，然后修改软、硬件，直到应用系统完全符合要求为止。