

基于Proteus的单片机 系统设计与仿真实例

蒋辉平 周国雄 主编

Proteus



机械工业出版社
CHINA MACHINE PRESS



附赠1CD

基于 Proteus 的单片机系统设计 设计与仿真实例

主 编 蒋辉平 周国雄
副主编 刘志壮 谭永宏
参 编 龙晓辉 黎 焕 廖小华
赵进辉



机械工业出版社

本书以 AT89C51 在 Proteus 平台下的仿真应用为核心, 使用汇编和 C 语言进行编程, 对单片机的 Proteus 仿真做了较为详细的介绍。包括简单的软件仿真与硬件仿真, 单片机的外围扩展与接口技术, 并给出了仿真实例。书中对发光二极管、数码管显示、键盘、简单 I/O 口、点阵 LED 显示、LCD 显示、可编程 I/O 口 8255、A/D 及 D/A 转换、步进电动机、I²C 总线、SPI 总线及单总线等做了相关介绍与实例仿真。另外, 本书的实例对于 AT89S51 系列单片机也全部适用。

本书附带光盘 1 张, 包括本书中的全部实例源代码及仿真电路原理图。

本书实例丰富, 有较强的实用性, 既可作为从事单片机设计的技术开发人员、高校师生及广大电子爱好者的参考资料, 也可作为单片机技术培训、电子信息工程等相关专业学生实验及电子设计的配套教材。

图书在版编目(CIP)数据

基于 Proteus 的单片机系统设计与仿真实例/蒋辉平, 周国雄
主编. —北京: 机械工业出版社, 2009. 4

ISBN 978-7-111-26650-1

I. 基… II. ①蒋…②周… III. ①单片微型计算机—系统设计—应用软件, Proteus②单片微型计算机—系统仿真—应用软件, Proteus IV. TP368

中国版本图书馆 CIP 数据核字(2009)第 042242 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 王保家 责任编辑: 卢若薇 版式设计: 张世琴

责任校对: 陈立辉 封面设计: 王伟光 责任印制: 李妍

北京铭成印刷有限公司印刷

2009 年 7 月第 1 版第 1 次印刷

184mm × 260mm · 19.5 印张 · 479 千字

标准书号: ISBN 978-7-111-26650-1

ISBN 978-7-89451-044-0(光盘)

定价: 39.00 元(含 1CD)

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

销售服务热线电话: (010)68326294

购书热线电话: (010)88379639 88379641 88379643

编辑热线电话: (010)88379727

封面无防伪标均为盗版

前 言

Proteus 是英国 Labcenter Electronics 公司开发的 EDA 工具软件。该软件具有原理布图、PCB 设计及自动布线和电路的分析与仿真功能，可以对基于微控制器的设计连同所有的周围电子器件一起仿真。用户甚至可以实时采用诸如 LED/LCD、键盘、RS232 终端等动态外设模型来对设计进行交互式仿真。Proteus 具有功能很强的 ISIS 智能原理图输入系统，有丰富的操作菜单与工具。在 ISIS 编辑区中，能方便地完成单片机系统的硬件设计、软件设计、单片机源代码级调试与仿真。Proteus 有三十多个元器件库，数千种元器件仿真模型，十余种信号激励源，十余种虚拟仪器仪表。特别是有从 8 位单片机 8051 系列直至 32 位单片机 ARM7 系列的多种单片机类型库。

本书以 AT89C51 在 Proteus 平台下仿真为核心，使用了汇编和 C 语言进行编程。内容包括有软件仿真、硬件仿真、各种外围扩展应用技术、单片机的串行通信技术及单片机应用系统程序设计，并且对单片机部分外围芯片和各种应用技术及原理做了详细的介绍。本书共分 6 章，每一章节的内容都是根据作者多年的教学经验做出的合理安排。内容由易到难，深入浅出，注解详细，汇编、C 语言同时并进。从第 2 章到第 6 章的每一节都有完整的仿真实例及仿真效果图，让学生感觉学有所成，可大大提高学习兴趣。本书为初学者对单片机学习和提高提供了很好的资料。另外，本书的实例对于 AT89S51 系列单片机也全部适用。

本书第 1 章介绍了 Proteus 仿真平台，Proteus 与 Keil 的联合仿真。第 2 章介绍了 Proteus 的软件仿真。第 3 章介绍了 Proteus 的硬件仿真。第 4 章对 AT89C51 单片机的外围扩展与接口技术仿真进行了介绍。第 5 章对 AT89C51 单片机的串行通信与接口技术进行了介绍。第 6 章对单片机应用系统设计进行了介绍。

在本书的编写过程中，作者参阅了大量的书籍及参考文献，并引用了部分参考文献和书籍。主要来源见附录。在此对有关作者表示衷心的感谢和谢意，本书亦是作者多年教学和科研经验的总结。

在本书的编写过程中，刘旭辉老师提出了宝贵的意见。邓利芝、朱磊、陈爱松、李花、刘俊、皮暑利、姚军、张体强、蒋婵等同学做了部分工作。在此也一并表示衷心的感谢和谢意。同时也感谢关心和支持我们的朋友。

由于编写时间仓促和水平有限，书中难免会有错误和不足之处，诚恳请求各位读者批评和指正。

作者 E-mail: hn_yz_laojiang@163.com

作 者

目 录

前言

第 1 章 Proteus 仿真基础 1

- 1.1 Proteus 仿真平台简介 1
- 1.2 Proteus 的基本操作 4
 - 1.2.1 对象的基本操作 4
 - 1.2.2 导线的基本操作 7
- 1.3 Proteus 与第三方软件接口 9
- 1.4 Proteus 与 Keil 联合仿真 10
 - 1.4.1 在 Keil 中编辑程序 10
 - 1.4.2 在 Keil 中调试 13
 - 1.4.3 在 Proteus ISIS 中调试 14
- 1.5 Proteus 简单仿真实例 16

第 2 章 AT89C51 单片机的软件仿真 18

- 2.1 片内数据存储器的读写 18
 - 2.1.1 程序流程图 18
 - 2.1.2 汇编语言源程序 18
 - 2.1.3 C 语言源程序 19
 - 2.1.4 片内数据存储器读写的仿真结果 20
- 2.2 片外数据存储器的读写 21
 - 2.2.1 程序流程图 21
 - 2.2.2 汇编语言源程序 21
 - 2.2.3 C 语言源程序 21
 - 2.2.4 片外数据存储器读写的仿真结果 22
- 2.3 数据传送 23
 - 2.3.1 程序流程图 23
 - 2.3.2 汇编语言源程序 23
 - 2.3.3 数据传送的仿真结果 23
- 2.4 数据排序 24
 - 2.4.1 程序流程图 24
 - 2.4.2 汇编语言源程序 24
 - 2.4.3 C 语言源程序 26

- 2.4.4 数据排序的仿真结果 26
- 2.5 双字节加法程序 27
 - 2.5.1 程序流程图 27
 - 2.5.2 汇编语言源程序 27
 - 2.5.3 双字节加法程序的仿真结果 28
- 2.6 双字节乘法程序 28
 - 2.6.1 程序流程图 28
 - 2.6.2 汇编语言源程序 28
 - 2.6.3 双字节乘法程序的仿真结果 29
- 2.7 拼字、拆字程序 30
 - 2.7.1 程序流程图 30
 - 2.7.2 汇编语言源程序 30
 - 2.7.3 拼字、拆字程序的仿真结果 31
- 2.8 转移程序的设计 32
 - 2.8.1 程序流程图 32
 - 2.8.2 汇编语言源程序 32
 - 2.8.3 C 语言源程序 33
 - 2.8.4 转移程序设计的仿真结果 34
- 2.9 位指令程序的设计 35
 - 2.9.1 程序流程图 36
 - 2.9.2 汇编语言源程序 36
 - 2.9.3 位指令程序设计的仿真结果 36

第 3 章 AT89C51 单片机的硬件仿真 38

- 3.1 发光二极管的应用 38
 - 3.1.1 程序流程图 38
 - 3.1.2 汇编语言源程序 38
 - 3.1.3 C 语言源程序 39
 - 3.1.4 发光二极管的应用仿真效果图 40
- 3.2 数码管显示的应用 40
 - 3.2.1 静态扫描显示 40
 - 3.2.2 动态扫描显示 43
- 3.3 键盘的应用 46
 - 3.3.1 独立式键盘的应用 46
 - 3.3.2 行列式键盘的应用 48

3.4 简单 I/O 接口的应用	53	4.4.3 C 语言源程序	104
3.4.1 程序流程图	53	4.4.4 ADC0808 A/D 转换器的应用 仿真效果图	106
3.4.2 汇编语言源程序	53	4.5 DAC0832 D/A 转换器的应用	106
3.4.3 C 语言源程序	55	4.5.1 DAC0832 简介	106
3.4.4 简单 I/O 接口的应用仿真 效果图	56	4.5.2 汇编语言源程序	109
3.5 定时器的应用	56	4.5.3 C 语言源程序	109
3.5.1 程序流程图	57	4.5.4 DAC0832 D/A 转换器的应用 仿真效果图	111
3.5.2 汇编语言源程序	57	4.6 步进电动机的控制	111
3.5.3 C 语言源程序	60	4.6.1 步进电动机简介	111
3.5.4 定时器的应用仿真效果图	62	4.6.2 汇编语言源程序	112
3.6 串行口的应用	62	4.6.3 C 语言源程序	117
3.6.1 串行口的基本应用	63	4.6.4 步进电动机的控制仿真效果图	122
3.6.2 采用串行口扩展并行输入口	66	4.7 PWM 脉冲控制信号的产生	123
3.7 中断的应用	68	4.7.1 PWM 简介	123
3.7.1 程序流程图	69	4.7.2 汇编语言源程序	123
3.7.2 汇编语言源程序	69	4.7.3 C 语言源程序	125
3.7.3 C 语言源程序	71	4.7.4 PWM 脉冲控制信号的产生软件 模拟仿真效果图	127
3.7.4 中断的应用仿真效果图	72	4.8 直流电动机的控制	128
第 4 章 AT89C51 单片机的外围 扩展与接口技术仿真	74	4.8.1 直流电动机简介	128
4.1 点阵 LED 显示的应用	74	4.8.2 汇编语言源程序	128
4.1.1 点阵 LED 简介	74	4.8.3 C 语言源程序	131
4.1.2 汇编语言源程序	75	4.8.4 直流电动机控制的仿真效果图	133
4.1.3 C 语言源程序	76	第 5 章 AT89C51 单片机的串行 通信与接口技术	134
4.1.4 点阵 LED 显示的应用仿真 效果图	77	5.1 AT24CXX 系列 EEPROM 的应用	134
4.2 LCD1602 的应用	78	5.1.1 I ² C 总线简介	134
4.2.1 LCD1602 简介	78	5.1.2 AT24C1024 简介	135
4.2.2 汇编语言源程序	80	5.1.3 AT24C1024 汇编语言源程序	139
4.2.3 C 语言源程序	83	5.1.4 AT24C1024 C 语言源程序	144
4.2.4 LCD1602 的应用仿真效果图	88	5.1.5 AT24C1024 仿真效果图	151
4.3 8255A 可编程并行 I/O 口的应用	88	5.2 采用 DS1302 时钟芯片制作电子时钟	153
4.3.1 8255A 简介	88	5.2.1 DS1302 时钟芯片简介	153
4.3.2 8255A 输入口的应用	91	5.2.2 DS1302 汇编语言源程序	156
4.3.3 8255A 输出口的应用	95	5.2.3 DS1302 C 语言源程序	162
4.4 ADC0808 A/D 转换器的应用	100	5.2.4 DS1302 仿真效果图	169
4.4.1 ADC0808 简介	100	5.3 DS18B20 温度传感器的应用	170
4.4.2 汇编语言源程序	102		

5.3.1	1-Wire 总线简介	170	6.1.2	汇编语言源程序	212
5.3.2	DS18B20 简介	171	6.1.3	C 语言源程序	217
5.3.3	DS18B20 汇编语言源程序	175	6.1.4	模拟交通灯仿真效果图	223
5.3.4	DS18B20 C 语言源程序	181	6.2	数字电压表的设计	224
5.3.5	DS18B20 仿真效果图	188	6.2.1	方案实现	224
5.4	TLC2543 A/D 转换器的应用	188	6.2.2	汇编语言源程序	224
5.4.1	SPI 总线简介	188	6.2.3	C 语言源程序	228
5.4.2	TLC2543 简介	190	6.2.4	数字电压表仿真效果图	230
5.4.3	TLC2543 汇编语言源程序	193	6.3	多功能电子万年历的设计	230
5.4.4	TLC2543 C 语言源程序	197	6.3.1	方案实现	230
5.4.5	TLC2543 仿真效果图	199	6.3.2	C 语言源程序	234
5.5	DS2430 EEPROM 芯片的应用	199	6.3.3	电子万年历仿真效果图	279
5.5.1	DS2430 简介	199	6.4	基于 DS18B20 的多点温度测量 系统的设计	279
5.5.2	汇编语言源程序	202	6.4.1	方案实现	279
5.5.3	C 语言源程序	206	6.4.2	汇编语言源程序	279
5.5.4	DS2430 仿真效果图	210	6.4.3	C 语言源程序	292
第 6 章	单片机应用系统设计	212	6.4.4	DS18B20 多点测温仿真效果图	301
6.1	模拟交通灯的设计	212	参考文献		302
6.1.1	方案实现	212			

第 1 章 Proteus 仿真基础

本章简单介绍了 Proteus 仿真平台常用的基本操作和一些基本应用，并结合一个简单实例对 Proteus 与 Keil 联合仿真调试进行了说明。

1.1 Proteus 仿真平台简介

Proteus 是英国 Labcenter Electronics 公司开发的 EDA 工具软件。该软件具有原理布图、PCB 设计及自动布线和电路的分析与仿真功能，可以对基于微控制器的设计连同所有的周围电子器件一起仿真。用户甚至可以实时采用诸如 LED/LCD、键盘、RS232 终端等动态外设模型来对设计进行交互式仿真。Proteus 具有功能很强的 ISIS 智能原理图输入系统，有丰富的操作菜单与工具。在 ISIS 编辑区中，能方便地完成单片机系统的硬件设计，软件设计，单片机源代码级调试与仿真。Proteus 有三十多个元器件库，数千种元器件仿真模型，十余种信号激励源，十余种虚拟仪器仪表。特别是有从 8 位单片机 8051 系列直至 32 位单片机 ARM7 系列的多种单片机类型库。Proteus 由 ISIS 和 ARES 两部分构成，其中 ISIS 是电子系统仿真平台，ARES 是布线编辑软件。本书只介绍有关 ISIS 的仿真应用。

安装完 Proteus 后，运行 ISIS 7 Professional，主窗口界面如图 1-1 所示。下面分别对窗口内各部分进行简单介绍。

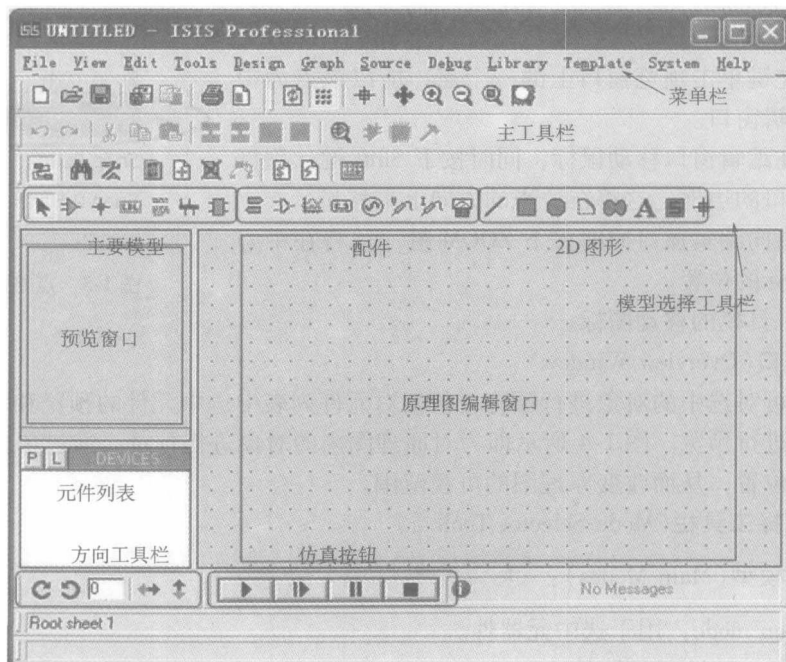


图 1-1 Proteus ISIS 主窗口界面

1. Proteus ISIS 菜单栏

菜单栏包括文件(File)、视图(View)、编辑(Edit)、工具(Tools)、设计(Design)、图形(Graph)、源(Source)、调试(Debug)、库(Library)、模板(Template)、系统(System)和帮助(Help),如图 1-2 所示。Proteus ISIS 菜单栏完全符合 Windows 风格。

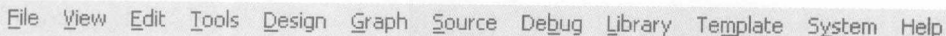


图 1-2 菜单栏

2. Proteus ISIS 主工具栏

主工具栏包括 File 工具栏、View 工具栏、Edit 工具栏和 Design 工具栏等。每个工具栏的打开与关闭,可通过 View | Toolbars... 命令进行设置。

3. 原理图编辑窗口(Editing Window)

原理图编辑窗口用于显示当前编辑的原理图的一部分。通过使用 View 菜单的 Redraw 命令可以使原理图编辑窗口的内容重绘,这也会重绘预览窗口。在任何其他的命令使得显示有些凌乱的时候也可使用这个功能。蓝色方框内为可编辑区,元件放置在编辑窗口里面。原理图编辑窗口没有滚动条,可用预览窗口来改变原理图的可视范围。ISIS 中坐标系统的基本单位是 10nm,主要是为了和 ARES 保持一致,但坐标系统的识别单位被限制在 1th。坐标原点默认在图形编辑区的中间,图形的坐标值显示在屏幕右下角的状态栏中,如: +200.0 +300.0 th。

编辑窗口内有点状的栅格,可以通过 View 菜单的 Grid 命令进行切换。可以通过 View | Snap 命令设置点与点之间的捕捉间距。如图 1-3 所示,通过 View 菜单选中 Snap 50th。通过使用 View 菜单的 X Cursor 命令,选中后将会在捕捉点显示一个交叉十字。

可以使用多种方法重新定位编辑窗口中设计的不同部分:

1) 通过左键单击预览窗口上的一个点,能以该点为中心重新定位编辑窗口。

2) 通过在编辑窗口移动鼠标,同时按下 Shift 键,用鼠标轻触编辑窗口的边框。这将在适当的方向上平移显示。

3) 通过指向编辑窗口同时按下 ZOOM 键,这将在中心重新定位显示光标的位置。

4) 使用工具栏的移动图标。

4. 预览窗口(Overview Window)

预览窗口可对选中的对象进行预览,包括对元件列表中选中元件的预览和对原理图编辑窗口内的对象进行预览。图 1-4 所示即为对原理图内的对象进行预览。通过单击鼠标可以改变绿色方框的位置,从而改变原理图的可视范围。

5. 模型选择工具栏(Mode Selector Toolbar)

(1) 主要模型(Main Modes):

① Selection mode: 用于选中元器件。

② Component: 选择元器件。

③ Junction Dot: 放置连接点。

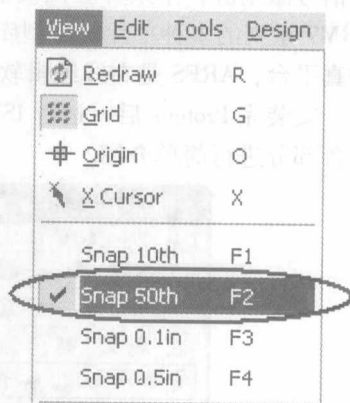


图 1-3 点格的设置

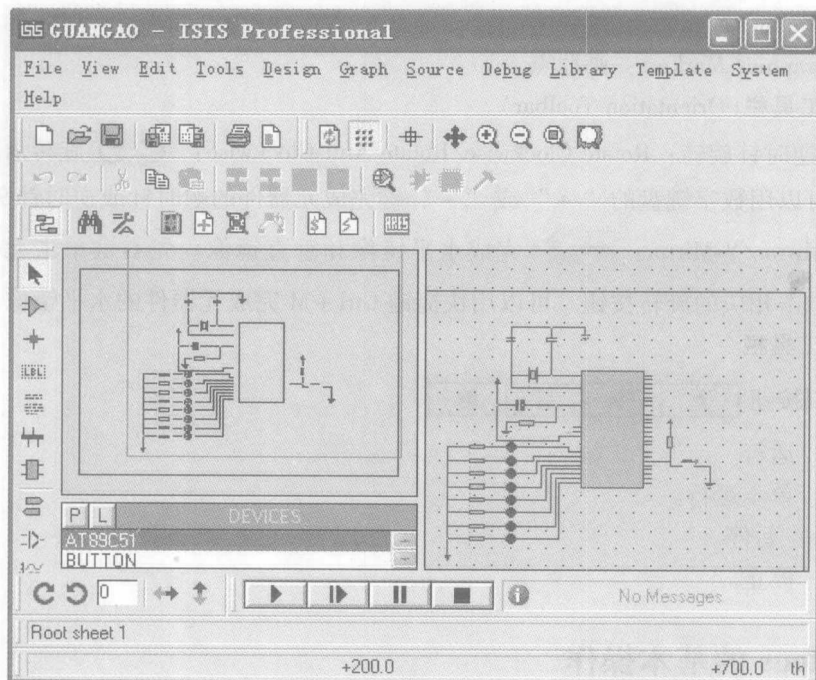


图 1-4 预览窗口的位置

- ④ Wire Label: 放置标签。
- ⑤ Text Script: 放置脚本。
- ⑥ Buses: 用于绘制总线。
- ⑦ Subcircuit: 绘制子电路块。

(2) 配件(Gadgets): 

- ① Terminals: 终端, 对象选择列出各种终端。
- ② Device Pins: 器件引脚, 对象选择列出各种引脚。
- ③ Graph: 图表, 对象选择列出各种仿真分析所需的图表。
- ④ Tape Recorder: 录音机, 对设计电路分割仿真时采用此模式。
- ⑤ Generator: 信号发生器, 对象选择列出各种激励源。
- ⑥ Voltage Probe: 电压探针, 可显示各探针处的电压值。
- ⑦ Current Probe: 电流探针, 可显示各探针处的电流值。
- ⑧ Virtual Instruments: 虚拟仪器, 对象选择列出各种虚拟仪器。


(3) 2D 图形(2D Graphics): 


- ① 2D Graphics Line: 画各种直线。
- ② 2D Graphics Box: 画各种方框。
- ③ 2D Graphics Circle: 画各种圆。
- ④ 2D Graphics Arc: 画各种圆弧。
- ⑤ 2D Graphics Closed Path: 画各种多边形。
- ⑥ 2D Graphics Text: 画各种文本。

⑦ 2D Graphics Symbols: 画符号。

⑧ 2D Graphics Markers: 画原点。

6. 方向工具栏 (Orientation Toolbar)

顺时针/逆时针旋转: Rotate Clockwise/Rotate Anti-Clockwise:  旋转角度只能是 90 的整数倍。可以用数字键盘的“+”或“-”键完成元器件的顺时针或逆时针 90°。

镜像 X-Mirror/Y-Mirror:  完成水平镜像和垂直镜像。先右键单击元件选中元器件, 再左键单击相应的旋转按钮。可以用快捷键 Ctrl + M 完成元器件的水平镜像。

7. 仿真工具栏

仿真控制按钮 

① Play: 运行。

② Step: 单步运行。


③ Pause: 暂停。

④ Stop: 停止。

1.2 Proteus 的基本操作

1.2.1 对象的基本操作

1. 选取对象

下面以选取电阻 RES 为例进行说明。单击元件列表之上的按钮 , 或直接按“P”键。此时弹出元件库选择对话框, 如图 1-5 所示。在 Keywords 编辑框中输入“RES”, 则 Results

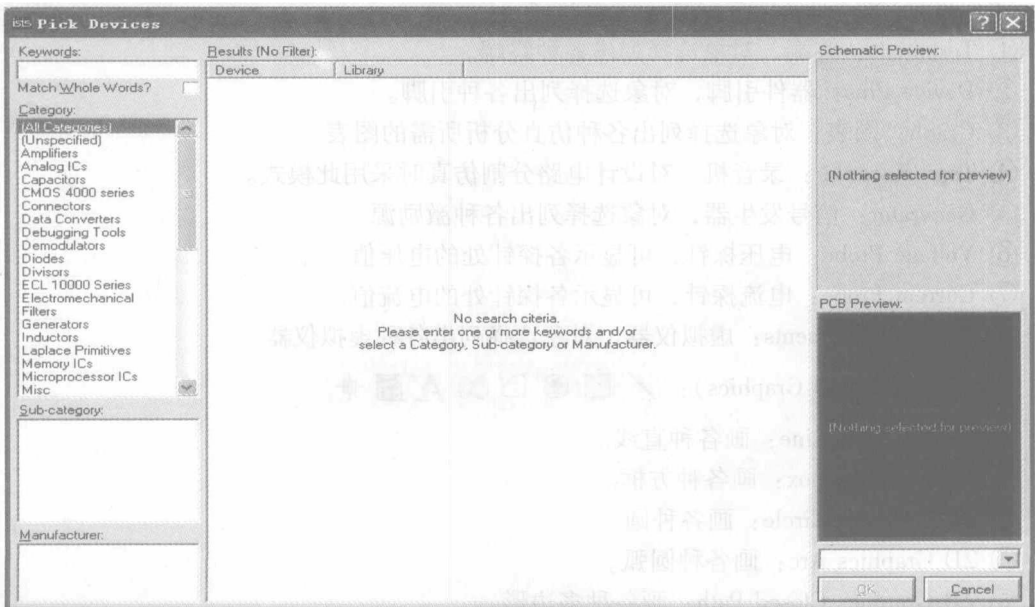


图 1-5 元器件选取对话框

列表区域列出所有与关键字相匹配的元器件，并在 Category 列表框中列出元器件所属的类。当选择不同的类、子类时，对应的元器件列表区域将列出对应类别的元器件，同时在 Results 列表区域给出元器件所在的库文件名及元器件的基本信息。在 Schematic Preview 预览区域，可以看到 RES 元器件的外形；而在 PCB Preview 区域，可以看到此元器件 PCB 预览图。当选中“Match Whole Words?”复选框时，元器件列表区域只列出完全与关键字相匹配的元器件。当对元器件的制造商有要求时，在 Manufacturer 区域选择期望的厂商，即可在元器件列表区域得到相应的元器件。

单击 OK 按钮，或在元器件列表区域双击元器件名称，即可完成对该元器件的添加。添加的元器件将出现在对象选择器列表中，如图 1-6 所示。

2. 放置对象

在对象选择器中，单击要放置的元器件，在编辑窗口中单击一下，此时鼠标处有一个红色的元器件，如图 1-7 所示。在合适的位置再单击一下左键，就放置一个元器件。也可单击右键或按 Esc 键取消元件的放置。

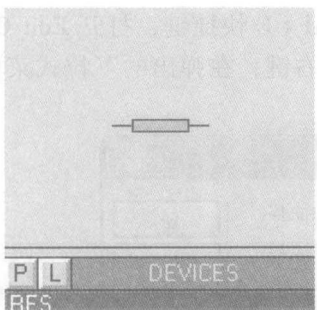


图 1-6 元器件的添加

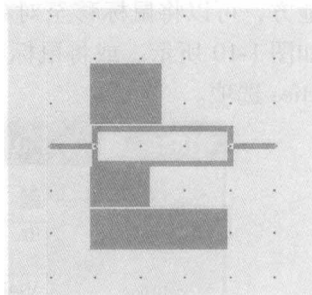


图 1-7 元器件的放置

3. 选中对象

用鼠标左键单击对象可以选中对象。该操作将使对象呈高亮显示。选中对象后可以对其进行编辑等操作。选中对象时该对象上所有连线全部被选中。

对于活动的器件，如开关 Button 等，如要选中可单击鼠标左键，拖出一个框，并将该元器件全部框住，便可选中，如图 1-8 所示。

要选中一组对象，可以通过按下 Ctrl 键，然后依次选中要选择的对象。也可以通过左键拖出一个框的方式，但只有完全位于框内的对象才能被选中，如图 1-9 所示。在空白处单击左键可以取消所有对象的选择。

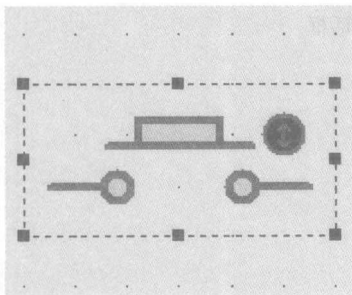


图 1-8 对活动器件的选择

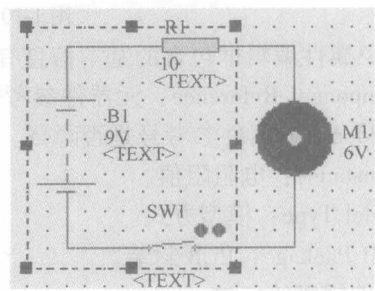





图 1-9 对一组对象的选择



4. 删除对象

用鼠标选中对象，然后按 Delete 键删除对象。或者直接在要删除的对象上单击右键，弹出的下拉式菜单中，选择 Delete Object 选项。

如果有误删除操作，可以按 Ctrl + Z 快捷键，或点击图标  取消。也可以在菜单栏 Edit 中选择 Undo 取消操作。

5. 调整对象的方向

选中对象，单击  按钮可以使对象顺时针方向旋转 90°。单击  按钮，可以使对象逆时针方向旋转 90°。

选中对象，单击  可以使对象按 X 轴镜像。或者使用快捷键 Ctrl + M。单击  可以使对象按 Y 轴镜像。

6. 编辑对象

用鼠标双击对象，弹出编辑对话框，可以通过对话框对对象属性进行编辑。在元器件比较密集的地方，可以将鼠标移至对象的上方，然后按 Ctrl + E 快捷键，打开 Edit Component 对话框，如图 1-10 所示。或将鼠标移到对象上方，单击右键，在弹出的下拉式菜单中选择 Edit Properties 选项。

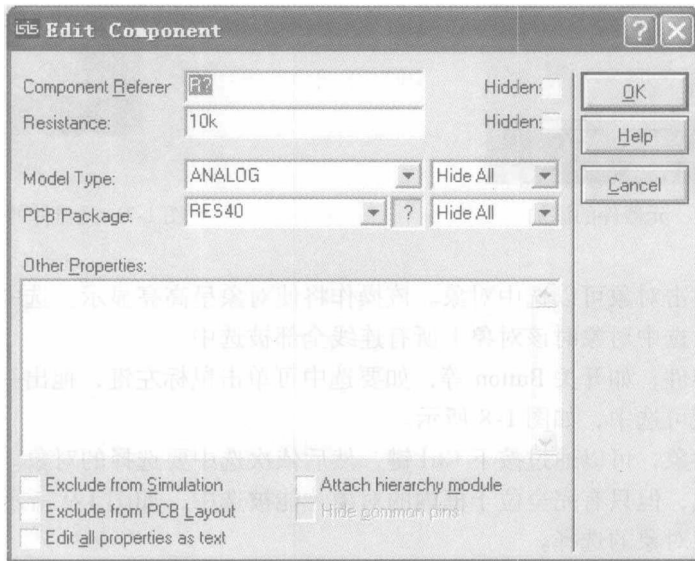


图 1-10 Edit Component 对话框

对象属性编辑框包含的基本信息有(以电阻为例)：

Component Reference：元器件参考号。

Hidden：元器件参考号是否隐藏。


Resistance：电阻阻值。

Model Type：模型方式。

PCB Package：PCB 封装。

7. 编辑网络标签

元件、端点和线都可以像元件一样编辑。使用网络标号，对应的网络标号之间是相互连

接的。单击网络标号 ，将鼠标移到元件的端点或线上，此时光标会变成十字形。然后单击左键，此时会弹出 Edit Wire Label 对话框，如图1-11 所示。在 String 对应的编辑栏中输入网络标号。

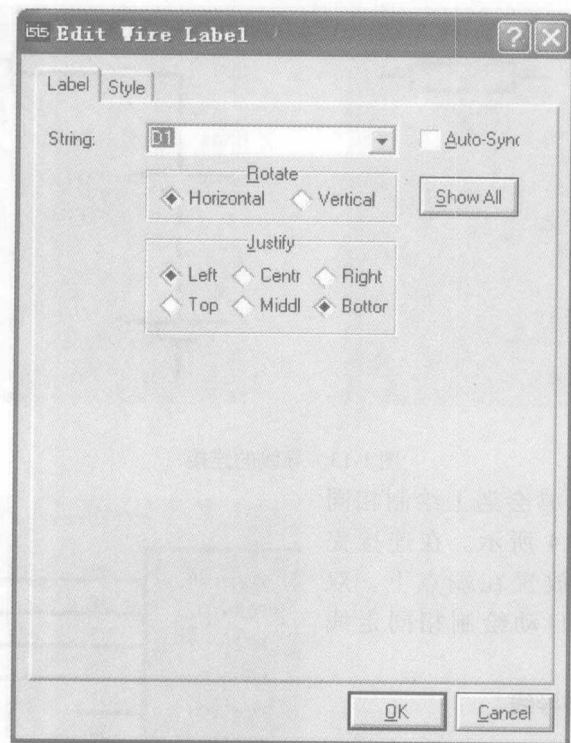




图 1-11 Edit Wire Label 对话框

8. 复制所有选中的对象

以复制四个电阻为例。选中需要的对象，单击 Copy 按钮 。把复制的轮廓拖到需要的位置。单击左键放置复制的对象，如图 1-12 所示。

9. 移动所有选中的对象

选中需要的对象。将所选的对象拖到需要的位置。或者按 Move 按钮 ，单击左键放置。

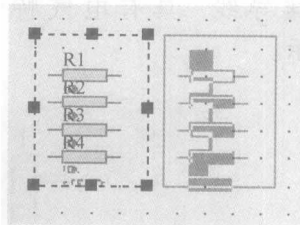


图 1-12 复制所有选中的对象

1.2.2 导线的基本操作

1. 绘制导线

原理图中的导线具有电气连接意义。在产生网络表时，就是根据导线或网络标号完成的。原理图中的总线不具有电气特性。总线的作用在于指引读者快速找到导线中相应网络标号的位置。

两个元器件间连接导线的步骤如下：

- 1) 单击第一个元器件的连接点，移动鼠标，此时会在连接点引出一根导线。

2) 如果想要 Proteus ISIS 自动写出直线路径, 只需单击另一个连接点, 如图 1-13 所示。如果用户想自己决定走线路径, 只需在拐点处单击, 在走线的过程中按下 Ctrl 键就会改变走线方式。

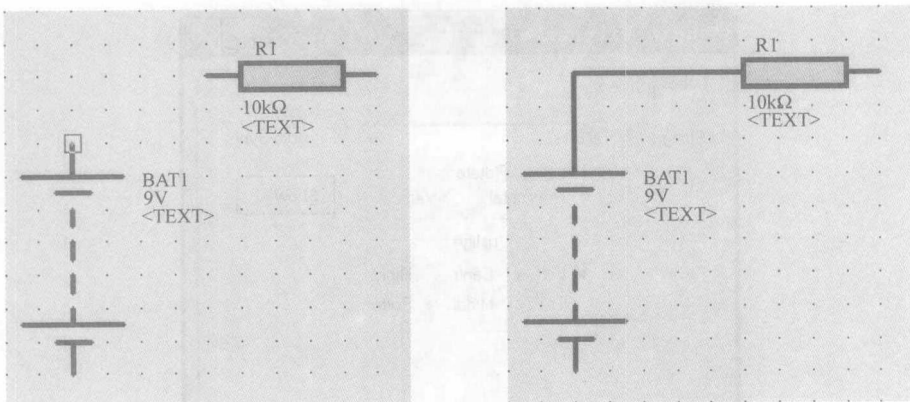



图 1-13 导线的连接

在绘制原理图中, 常会遇上绘制相同走线的导线, 如图 1-14 所示。在连接完第一根线后, 将鼠标放置在端点上, 双击鼠标左键, 软件会自动绘制相同走线的导线。

2. 使用连接点连接导线

单击工具箱中 Junction Dot 按钮 , 可在电路中添加圆点。连接点可以精确的连接导线, 具有电气特性。如图 1-15 所示。

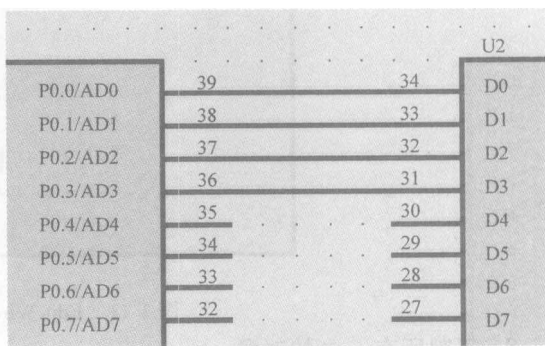


图 1-14 绘制相同走线的导线

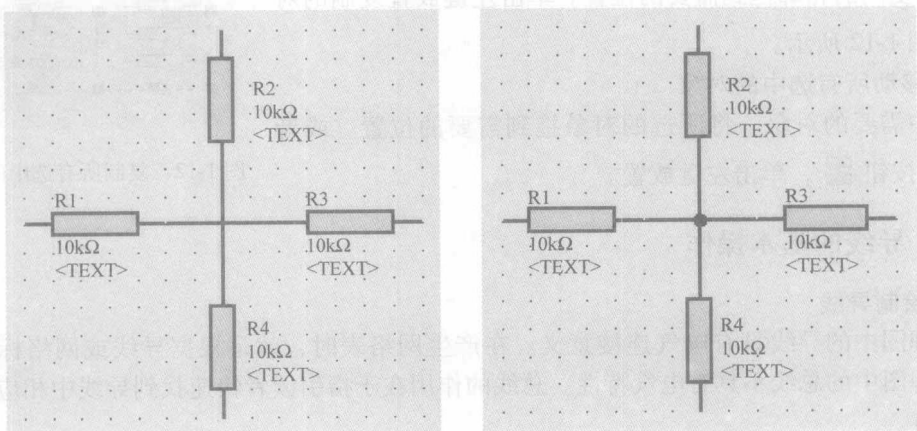




图 1-15 连接点的应用

3. 线路自动路径器


线路自动路径器  为用户省去了必须标明每根线的具体路径的麻烦。自动接线功能默认是打开的。

如果用户只在两个连接点单击，自动接线将选择一个合适的走线方式。但是如果已选择了一个连接点，用户在走线过程中，单击了一个或是多个非连接点后，Proteus ISIS 会认为用户是在手工定线。

4. 移动导线

选择  按钮，然后将鼠标移至导线的上方，单击左键，光标会变成一个双向箭头，然后移动鼠标。如图 1-16 所示。

5. 绘制总线

单击  按钮，移动鼠标到需要绘制总线的位置，单击左键，移动鼠标，便可绘制出一条总线，如图 1-17 所示。

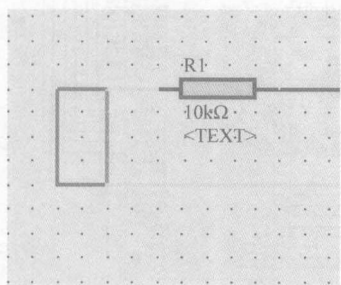


图 1-16 移动导线

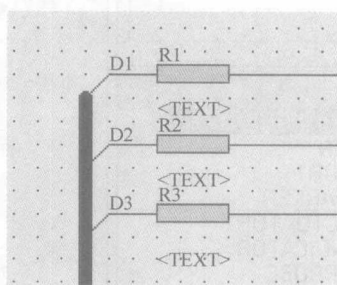


图 1-17 绘制总线

1.3 Proteus 与第三方软件接口

Keil μ Vision2(简称 Keil2)是德国 Keil 公司出品的 51 系列兼容单片机软件开发系统。该系统支持 C 语言和汇编语言。Keil2 界面友好，操作简单。下面简单介绍 Proteus 与 Keil2 的连接设置。

- 1) 安装 Keil2 和 Proteus 7.1。
- 2) 将 Proteus 7 Professional\Models\VDM51.DLL 复制到 Keil2\C51\bin 目录下。
- 3) 在 Keil2\TOOLS.INI 文件中的[C51]字段下添加：TDRV4 = BIN\VDM51.DLL(“Proteus VSM Simulator”)，并保存。其中 TDRV4 中的数字“4”可以任意。
- 4) 在 Proteus 中绘制原理图后，选取 Debug | Use Remote Debug Monitor 选项。
- 5) 在 Keil2 中编制程序完成后，选取 Project | Options for Target 'Target1'选项，选择 Debug 选项卡，选中 Proteus VSM Simulator 选项。单击 Setting 按钮，设置 Host 为 127.0.0.1，Port 为 8000。
- 6) 在 Keil2 中进行 Debug，同时在 Proteus 中查看结果。如图 1-18 所示。

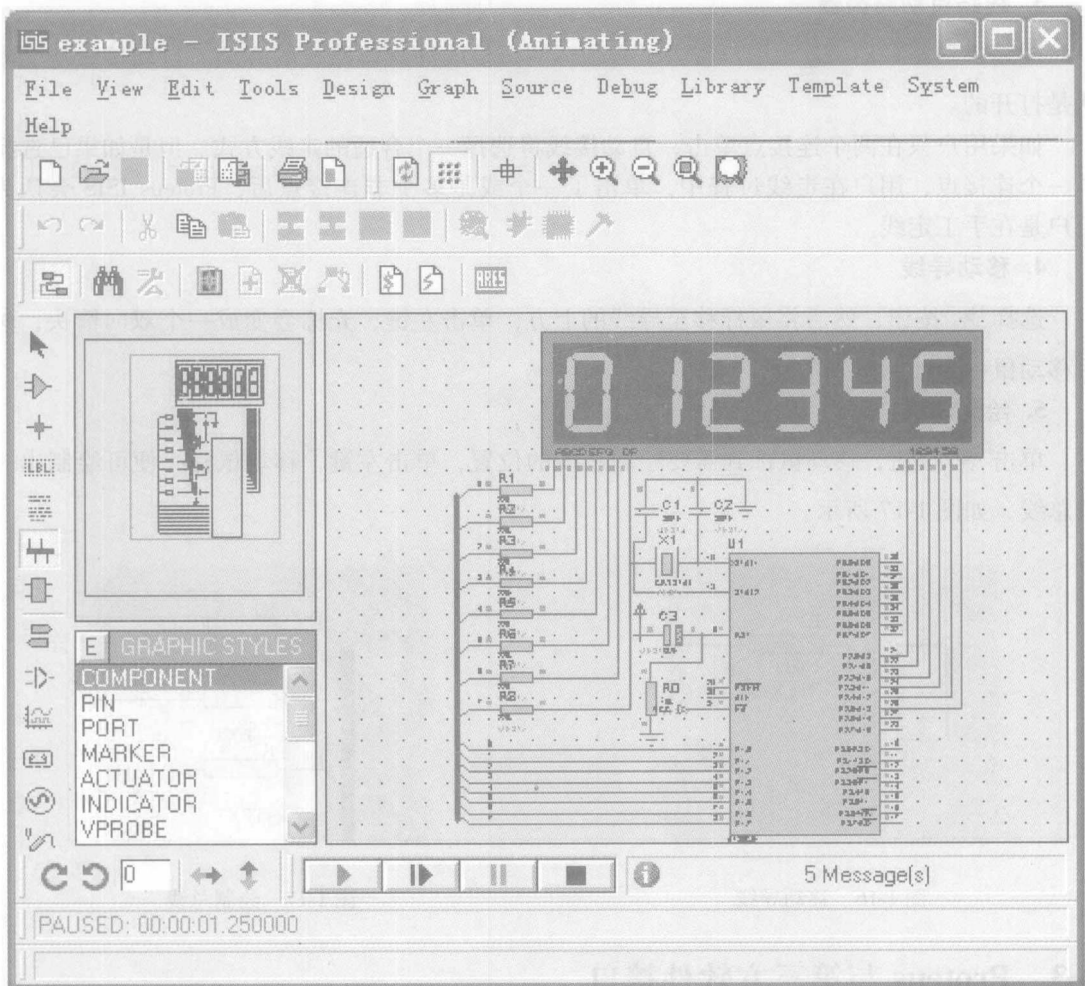


图 1-18 Proteus 与 Keil2 联机调试

1.4 Proteus 与 Keil 联合仿真

下面通过一个数据存储置数程序完成 Proteus ISIS 和 Keil 的联合仿真。该实例将单片机片内 RAM 的 30H-3FH 存储空间分别置为 00-0FH。

1.4.1 在 Keil 中编辑程序

打开 Keil2，选择 Project | New Project 命令，在弹出的 Create New Project 对话框中选择目标保存路径，在“文件名”编辑框中输入工程名称，如图 1-19 所示。

单击“保存”，在弹出的 Select Device for Target 'Target 1'对话框中双击 Data base 框中的 Atmel 选项，选择 AT89C51 单片机，如图 1-20 所示，在这里是设置我们的 MCU 为 AT89C51 单片机。按“确定”后，在随后弹出的 μ Vision2 的对话框中选择“是”。