

Proteus 教程

——电子线路设计、制版与仿真

(修订版)

朱清慧 张凤蕊 编著
翟天嵩 王志奎

清华大学出版社

北 京

内 容 简 介

本书详细介绍了 Proteus 软件在电子线路设计中的具体应用,可划分为四大部分:Proteus 8.7 软件导入及环境介绍,电子线路仿真与设计,微处理器系统仿真及 PCB 设计。第 1 章和第 2 章循序渐进地介绍了 Proteus ISIS 的具体功能;第 3 章和第 4 章介绍了基于 Proteus ISIS 的模拟电子技术、数字电子技术实验和综合设计与仿真;第 5 章和第 6 章对 51 系列单片机电路的设计和仿真介绍了大量的实例,并且对源程序与硬件电路的交互仿真做了重点介绍;第 7 章对其他单片机系列控制电路的设计和仿真进行了实例讲解;第 8 章讲述了 Proteus ARES 的 PCB 设计过程。

本书所引实例是作者多年教学和实际工作中的典型实例的总结和积累,经过充分的仿真验证和实际应用,读者在学习时很容易上手。本书的特色是通过实例学习软件,不用层层叠叠的菜单命令来困扰读者;内容编排上由浅入深,循序渐进,引领读者逐步深入 Proteus 的学习和应用。

本书结构清晰,语言通俗易懂,可作为高等院校电路设计与仿真类课程的教材及电子技术和单片机教学课程设计与实验教材,也可作为广大电子技术爱好者、在校电类工科大学生以及单片机系统开发者的自学用书。

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。举报:010-62782989, beiqinquan@tup.tsinghua.edu.cn。

图书在版编目(CIP)数据

Proteus 教程:电子线路设计、制版与仿真 / 朱清慧等编著. —修订版. —北京:清华大学出版社, 2023.6

ISBN 978-7-302-63824-7

I. ①P… II. ①朱… III. ①电子电路—计算机辅助设计—应用软件—高等学校—教材
IV. ①TN702.2

中国国家版本馆 CIP 数据核字(2023)第 104382 号

责任编辑:刘金喜

封面设计:范惠英

版式设计:妙思品位

责任校对:成凤进

责任印制:曹婉颖

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:010-83470000 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 装 者:三河市人民印务有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:21 字 数:485 千字

版 次:2023 年 8 月第 1 版 印 次:2023 年 8 月第 1 次印刷

定 价:78.00 元

产品编号:101494-01

序

作为 Proteus “大学计划”的一部分，Labcenter 和风标电子(Proteus 产品中国区总代理)一直鼓励和支持有经验的教师基于其挂牌 Proteus 实验室平台对 Proteus 应用于教学与科研的研究成果形成著作出版发行。Labcenter 和风标电子支持的有关 Proteus 的书籍出版了十多种，各书都有不同的侧重点，而且 Labcenter 的宗旨是持续不断地开发和升级，保持技术一流。Proteus 的升级非常频繁，目前又增加了很多新的功能、新的模型，市场上也客观地需要一种更新、更全的参考书。

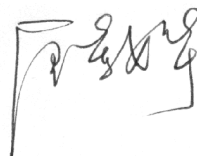
本书是一部介绍 Proteus 的全面翔实的教材，对 Proteus 各部分的功能都有详细的阐述和实例讲解，充分展示了 Proteus 从概念到产品的整个过程，包括智能原理布图、基本电路仿真、模型库的介绍、基于微控制器的协同仿真到 PCB 布板等各个环节，体现了作者坚实的专业功力和驾驭 Proteus 的能力。

在使用本书的过程中，如果需要对书中的实例或 Proteus 本身有更深入的了解，可通过清华大学出版社将您的意见或建议转发给我们。

电话：010-62776969

邮箱：c-service@tup.tsinghua.edu.cn

广州市风标电子技术有限公司



前 言

Proteus 嵌入式系统仿真与开发平台由英国 Labcenter 公司开发,是目前世界上比较先进的嵌入式系统设计与仿真平台。它是一种可视化的支持多种型号单片机(如 51、PIC、AVR、Motorola hcll 等),并且能够与当前流行的单片机开发环境(Keil、MPLAB、IAR)进行连接调试的软硬件仿真系统。Proteus 除了具有和其他 EDA 工具一样的原理图、PCB 自动或人工布线及电路仿真功能外,还对微控制系统与外设的混合电路的电路仿真、软件仿真、系统协同仿真做到了一体化和互动效果。

Proteus 软件已有 30 多年的历史,在全球拥有庞大的企业用户群,是目前唯一能够对各种处理器进行实时仿真、调试与测试的 EDA 工具,真正实现了在没有目标原型时就可对系统进行设计、测试与验证。由于 Proteus 软件包括逼真的协同仿真功能,因此得到了包括剑桥大学在内的众多大学用户的青睐,并作为电子学或嵌入式系统的课程教学、实验和水平考试平台。目前,Proteus 在国内单片机开发者及单片机爱好者中已开始普及,有很多开发者已经开始用此开发环境进行仿真。

Proteus 软件经过不断升级之后,功能越来越强大,性能也更加卓越,快速设计是 Proteus 软件升级的永恒主题。Proteus 8 Professional 具有三个典型特点,一是高度集成的工作视窗界面,二是公共的数据库,三是具有一个实时的网络表。Proteus 8 与 Proteus 7 相比,有很多新的变化,对于习惯了 Proteus 7 的老用户来说可能一时还不能快速适应,但 Proteus 8 真的非常优秀,一旦走入这个名副其实的虚拟实验室,将会受益匪浅、乐而忘返。在这次修订时,我们把 Proteus 8.7 版本软件的主要功能和优越性能推介给广大读者,希望大家能够快速上手。

本书对 Proteus 软件功能的介绍,与一般软件教程的明显区别在于,一开始并不是罗列大量的菜单,而是以简单的电路仿真实例逐步激发初学者的兴趣,在实例中学会关键菜单和主要工具命令,逐步加深,最后对命令和主工具进行系统的介绍。虽然本书的重点是单片机系统的设计和仿真,但为了使具有电子学基础知识的读者也能使用该软件,仍列举了电路分析、模拟电路和数字电路等基础学科的电路设计与仿真实例,同时对 PCB 设计也做了详细介绍。丰富的实例教学,使学习变得轻松而愉快。

本书共分四大部分:Proteus 8.7 软件导入及环境介绍,电子线路仿真与设计(电路分析、模拟电子技术、数字电子技术等电路设计仿真),微处理器系统仿真及 PCB 设计。每部分

都有相应的实例，例子选用每个学科中具有代表性的电路，给出 Proteus 元件清单、完整原理图及程序、仿真工具及仪器设置、仿真步骤及电路功能。内容安排由浅入深，由易到难，适合不同层次的电子学爱好者，既照顾到了初学 Proteus 和用习惯了 Proteus 7 的读者，也对想要快速掌握 Proteus 8 的读者有一定的帮助。

本书的读者对象是在校电类工科大学生、广大电子技术爱好者及单片机系统开发者，同时本书也可作为高等院校电路设计与仿真类课程的教材及电子技术和单片机教学课程设计与实验教材。

本书共 8 章，由南阳理工学院的朱清慧、张凤蕊、翟天嵩、王志奎教授编写完成。全书由朱清慧统稿、审定。

由于编者水平有限，书中难免存在不足之处，还望广大读者批评指正。

本书教学课件和实例源文件可通过扫描下方二维码下载。

服务邮箱：476371891@qq.com。



教学资源

编 者
2023 年 1 月

目 录

第 1 章 Proteus 8 快速入门	1
1.1 Proteus 8 整体功能预览	1
1.1.1 集成化的多视窗工作界面	1
1.1.2 Proteus 8 的虚拟仿真模式	3
1.2 Proteus 8 集成环境认识	6
1.2.1 Proteus 8 Professional 的安装与运行	6
1.2.2 Proteus 8 Professional 的快速设计向导	8
1.2.3 一阶动态电路的设计与仿真	21
1.2.4 异步四位二进制计数器的设计与仿真	29
1.2.5 AT89C51 与 8255A 接口电路的调试及仿真	34
第 2 章 原理图设计工作视窗	37
2.1 Schematic Capture 视窗环境及命令	37
2.1.1 Schematic Capture 视窗环境简介	37
2.1.2 Schematic Capture 视窗主菜单命令	38
2.2 绘图工具图标	46
2.2.1 选择模式	46
2.2.2 元件模式	46
2.2.3 结点模式	47
2.2.4 总线模式	47
2.2.5 导线标号模式	48

2.2.6 文本脚本模式	48
2.2.7 放置终端模式	51
2.2.8 元件引脚模式	52
2.2.9 子电路模式	52
2.2.10 二维图形模式	55
2.3 仿真工具图标	57
2.3.1 激励源模式	57
2.3.2 虚拟仪器模式	70
2.3.3 图表仿真	85
2.4 Proteus ISIS 的元件库	91
2.4.1 库元件的分类	92
2.4.2 各子类介绍	93
2.4.3 常用元件	104
第 3 章 电子技术实验	108
3.1 模拟电子技术实验	108
3.1.1 晶体管共射极单管放大器	108
3.1.2 差动放大器	111
3.1.3 低频功率放大器(OTL)	114
3.1.4 比例运算放大器	117
3.2 数字电子技术实验	119
3.2.1 门电路逻辑功能及测试	119
3.2.2 译码器和数据选择器	122
3.2.3 移位寄存器的功能测试	123
3.2.4 时序电路	125
3.2.5 集成计数器	127
3.2.6 投票表决电路设计与仿真	129
3.2.7 ADC0808 和 DAC0832 的应用设计与仿真	131

3.2.8 显示译码器和数码管的应用 设计与仿真	133	5.2.4 观察窗口	180
第4章 电子技术综合设计	137	5.3 应用I/O口输入/输出	181
4.1 直流可调稳压电源	137	5.3.1 Proteus电路设计	182
4.2 四路彩灯	141	5.3.2 源程序设计	182
4.2.1 核心器件74LS194简介	141	5.3.3 Proteus调试与仿真	183
4.2.2 题目分析与设计	142	5.3.4 总结与提示	184
4.2.3 仿真	144	5.4 4×4矩阵式键盘识别技术	184
4.2.4 扩展电路	145	5.4.1 Proteus电路设计	185
4.3 八路抢答器	146	5.4.2 源程序设计	185
4.3.1 核心器件74LS148简介	146	5.4.3 Proteus调试与仿真	187
4.3.2 题目分析与设计	147	5.4.4 总结与提示	187
4.4 数字钟	150	5.5 动态扫描显示	187
4.4.1 核心器件74LS90简介	150	5.5.1 Proteus电路设计	188
4.4.2 分步设计与仿真	151	5.5.2 源程序设计	188
4.5 音乐教室控制台	157	5.5.3 Proteus调试与仿真	190
4.5.1 核心器件74LS190简介	158	5.5.4 总结与提示	190
4.5.2 题目分析与设计	158	5.6 8×8点阵LED显示	190
4.6 直流数字电压表	163	5.6.1 Proteus电路设计	190
4.6.1 系统功能模块组成	163	5.6.2 源程序设计	192
4.6.2 A/D转换模块和时钟模块	163	5.6.3 Proteus调试与仿真	193
4.6.3 二/十进制转换电路	165	5.6.4 总结与提示	193
4.6.4 显示电路	166	5.7 I/O口的扩展	193
4.6.5 焊接与调试	167	5.7.1 Proteus电路设计	193
第5章 MCS-51 单片机接口基础	170	5.7.2 源程序设计	194
5.1 汇编源程序的建立与编译	170	5.7.3 Proteus调试与仿真	195
5.1.1 Proteus中的源程序设计与 编译	170	5.7.4 总结与提示	195
5.1.2 Keil μ Vision中的源程序设计与 编译	172	5.8 定时器/计数器实验	196
5.2 Proteus与单片机电路的交互式 仿真与调试	177	5.8.1 Proteus电路设计	196
5.2.1 加载目标代码	177	5.8.2 源程序设计	197
5.2.2 单片机系统的Proteus 交互仿真	178	5.8.3 Proteus调试与仿真	197
5.2.3 调试菜单与调试窗口	179	5.8.4 总结与提示	197
		5.9 外部数据存储器扩展	198
		5.9.1 Proteus电路设计	198
		5.9.2 源程序设计	199
		5.9.3 Proteus调试与仿真	199
		5.9.4 总结与提示	199
		5.10 外部中断实验	200

5.10.1	Proteus电路设计	201	6.1.3	Proteus调试与仿真	226
5.10.2	源程序设计	202	6.1.4	总结与提示	228
5.10.3	Proteus调试与仿真	202	6.2	I ² C总线应用技术	228
5.10.4	总结与提示	203	6.2.1	Proteus电路设计	228
5.11	单片机与PC机间的串行通信	203	6.2.2	源程序设计	229
5.11.1	Proteus电路设计	203	6.2.3	Proteus调试与仿真	233
5.11.2	源程序设计	205	6.2.4	用I ² C调试器监视I ² C总线	234
5.11.3	Proteus调试与仿真	206	6.2.5	总结与提示	234
5.11.4	总结与提示	207	6.3	基于单片机控制的电子万年历	234
5.12	单片机与步进电机的 接口技术	207	6.3.1	设计任务及要求	234
5.12.1	Proteus电路设计	207	6.3.2	设计背景	235
5.12.2	源程序设计	208	6.3.3	电路设计	235
5.12.3	Proteus调试与仿真	209	6.3.4	系统硬件实现	242
5.12.4	总结与提示	209	6.3.5	系统软件实现	244
5.13	单片机与直流电动机的 接口技术	210	6.3.6	Proteus调试与仿真	250
5.13.1	Proteus电路设计	210	6.4	基于DS18B20的水温控制系统	250
5.13.2	源程序设计	211	6.4.1	Proteus电路设计	251
5.13.3	Proteus调试与仿真	212	6.4.2	源程序清单	252
5.13.4	总结与提示	212	6.4.3	Proteus调试与仿真	256
5.14	基于DAC0832数模转换器的 数控电源	213	6.5	基于单片机的24×24点阵LED汉字 显示	257
5.14.1	Proteus电路设计	213	6.5.1	设计任务及要求	257
5.14.2	源程序设计	214	6.5.2	设计背景简介	257
5.14.3	Proteus调试与仿真	215	6.5.3	电路设计	257
5.14.4	总结与提示	216	6.5.4	系统硬件实现	258
5.15	基于ADC0808模数转换器的 数字电压表	216	6.5.5	系统软件实现	261
5.15.1	Proteus电路设计	216	6.5.6	系统仿真	265
5.15.2	源程序设计	217	第7章	其他类型单片机系统的 Proteus设计与仿真	267
5.15.3	Proteus调试与仿真	221	7.1	PIC单片机与字符液晶显示器的 接口	267
5.15.4	总结与提示	221	7.1.1	Proteus电路设计	267
第6章	AT89C51单片机综合设计	222	7.1.2	源程序清单	269
6.1	单片机间的多机通信	222	7.1.3	Proteus调试与仿真	271
6.1.1	Proteus电路设计	222	7.2	PIC单片机间的串口通信	272
6.1.2	源程序设计	223	7.2.1	Proteus电路设计	272
			7.2.2	源程序清单	273

7.2.3 Proteus调试与仿真·····	275	8.5 网络表的生成·····	303
7.3 AVR单片机AD转换·····	276	8.6 网络表的导入·····	304
7.3.1 Proteus电路设计·····	276	8.7 系统参数设置·····	306
7.3.2 源程序清单·····	278	8.7.1 设置电路板的工作层·····	306
7.3.3 Proteus调试与仿真·····	280	8.7.2 环境设置·····	307
7.4 基于AVR单片机的直流电机		8.7.3 栅格设置·····	308
控制电路·····	281	8.7.4 路径设置·····	308
7.4.1 Proteus电路设计·····	281	8.8 编辑界面设置·····	308
7.4.2 源程序清单·····	282	8.9 布局与调整·····	310
7.4.3 Proteus调试与仿真·····	289	8.9.1 自动布局·····	310
7.5 ARM入门·····	290	8.9.2 手工布局·····	311
7.5.1 Proteus电路设计·····	290	8.9.3 调整元件标注·····	313
7.5.2 源程序清单·····	291	8.10 设计规则的设置·····	313
7.5.3 Proteus调试与仿真·····	293	8.10.1 设置设计规则·····	313
第8章 Proteus ARES 的PCB设计···	295	8.10.2 设置默认设计规则·····	314
8.1 Proteus ARES编辑环境·····	295	8.11 布线·····	315
8.1.1 Proteus ARES工具箱		8.11.1 手工布线·····	315
图标按钮·····	296	8.11.2 自动布线·····	317
8.1.2 Proteus ARES主菜单栏·····	297	8.11.3 自动整理·····	318
8.2 印制电路板(PCB)设计流程·····	298	8.12 设计规则检测·····	319
8.3 为元件指定封装·····	299	8.13 后期处理及输出·····	320
8.4 元件封装的创建·····	300	8.13.1 PCB敷铜·····	320
8.4.1 放置焊盘·····	300	8.13.2 PCB的三维显示·····	321
8.4.2 分配引脚编号·····	302	8.13.3 PCB的输出·····	321
8.4.3 添加元件边框·····	302	8.14 多层PCB设计·····	323
8.4.4 元件封装保存·····	303	参考文献·····	326

Proteus 8 快速入门

Proteus 是由英国 Labcenter Electronics 公司开发的电子设计辅助工具软件, 已有 30 多年的历史, 在全球得到了广泛应用。Proteus 软件功能强大, 它集电路原理图设计、程序设计与编译、虚拟仿真、印制电路板(PCB)布线等多种功能于一体, 不仅能够对基本的电工电路、模拟和数字电路进行设计、仿真与分析, 还能够对微处理器系统进行原理图设计、程序设计和仿真, 支持层次电路图设计和复杂工程的团队协作开发。

1.1 Proteus 8 整体功能预览

Proteus 7 及以下版本主要基于两个独立界面 ISIS(Intelligent Schematic and Interactive Simulation)和 ARES(Automatic Routing and Editing Software)分别进行原理图设计和 PCB 设计。而升级版的 Proteus 8 为用户带来了高度集成化的工作视窗和更加快捷的设计体验, 其丰富完善的元件库(原理图仿真器件库和 PCB 布线零件库)和虚拟仪器工具使用户爱不释手。本书以 Proteus 8.7 版本为基础进行电路设计与仿真实践。

1.1.1 集成化的多视窗工作界面

Proteus 8 版本的主题是集成和快速设计, 集成化的工作视窗和提前植入的多项功能使其更人性化、更高效。软件开发的重点是将电子设计的各个独立部分结合在一起, 以实现更好的工作流程和更快速的项目设计。为了实现这一点, 以下三个主要的架构变化是必要的:

- 一个统一的应用程序框架。
- 一个公共数据库。
- 一个实时网络表。

在统一的应用程序框架下, 单个应用程序以工作视窗的形式集成在一个工作界面中, 通过主菜单栏下面的工作视窗选项卡图标选择当前视窗。这些工作视窗主要有:

- 原理图设计(Schematic Capture)——智能原理图输入、系统设计与仿真。
- 印制电路板设计(PCB Layout)——高级 PCB 布线编辑。
- 程序设计(Source Code)——程序代码编写、编译和仿真。
- 三维可视化(3D Visualizer)——实际电路板的三维效果生成展示。
- 设计资源管理器(Design Explorer)——原理图设计中的元件信息生成列表。
- 物料清单(Bill of Materials)——工程中所包含的物料清单生成显示。

- Gerber 查看器(Gerber Viewer)——PCB 板层设计实际效果查看。
- 项目备注(Project Notes)——工程文件下的项目备注文档编辑。
- 主页面(Home Page)——工程创新向导及学习教程等。

这些不同工作视窗下设计的文件统一保存在 Proteus 8 的一个工程文件中,共享一个数据库。运行 Proteus 8 后,先打开主界面,通过选择创建工程向导等打开集成工作视窗界面。根据需要,读者可同时打开多个工作视窗,单屏工作时可通过工作视窗选项卡来切换当前视窗,双屏或多屏工作时可同时显示多个当前视窗。

在 Proteus 中,从原理图设计、程序设计、系统仿真到 PCB 设计一气呵成,真正实现了从概念到产品的完整设计。Proteus 从原理图设计到 PCB 设计,再到实际电路板完成的流程如图 1-1 所示。

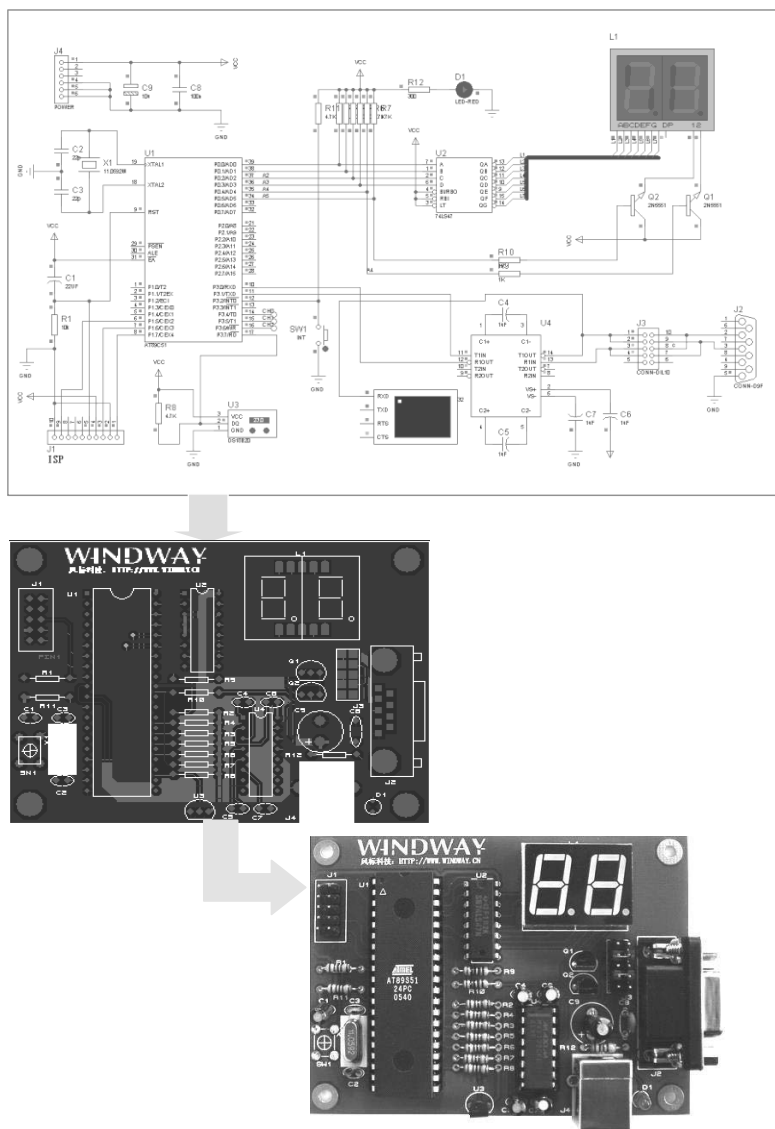


图 1-1 Proteus 设计流程

注意:

Proteus 软件中的电气符号及标识非我国国标, 本书未对截屏图中的电气符号进行修改, 正文中涉及的电气符号标识也力求和截图一致, 以增加本教程的可读性及 Proteus 软件的操作性。

在图 1-1 中, 最上面是一个基于单片机的应用电路原理图, 显示的画面正处于仿真运行状态。设计者可以从 Schematic Capture 视窗的原理图库中调用所需电路元件, 然后通过适当连线即可完成; 在 Source Code 视窗中进行程序设计和编译(也可从外部直接把编译后的程序文件导入单片机中), 然后返回 Schematic Capture 视窗运行交互仿真(无须向单片机元件导入程序), 即可查看电路功能效果。中间图片是 PCB Layout 视窗中设计的 PCB, 通过实时网络表建立原理图与 PCB 之间的对应连线关系。最下面的图为最终完成的实际电路板。可见, 整个电路从设计到实际电路制作完成, 通过一款 Proteus 软件即可完美实现。并且, 它的仿真结果与实际误差很小, 非常适合电子设计爱好者和高校学生自学使用, 能够缩短项目设计周期, 提高设计成功率。

1.1.2 Proteus 8 的虚拟仿真模式

Proteus 还支持虚拟仿真模式(Virtual Simulation Mode, VSM)。Proteus ISIS 的元件库类别丰富、功能强大, 库元件具有各自的电气特性, 可供进行虚拟仿真。当电路连接完成无误或程序编译成功后, 直接运行交互仿真按钮, 即可实现数据显示和声、光、色、动等逼真的效果, 还可以通过放置静态图表进行输出波形自动生成, 非常方便直观。

1. Proteus ISIS 的虚拟仿真模式

Proteus 8 的 VSM 有以下几种不同的仿真方式, 通称交互式仿真。

- 动态仿真——通过元件特性和仪器测试, 实时直观地反映电路设计的仿真结果(原理图视窗: 仿真元件+激励源+虚拟仪器)。
- 静态仿真——通过放置在电路中的图表生成所要测量的波形, 用来精确分析电路的各种性能, 如频率特性、噪声特性等(原理图视窗: 探针+图表)。
- 程序仿真——(程序代码视窗: 程序代码+原理图活动窗口)。

通过不同模式的仿真, 可实现对所绘制或设计的电路及程序进行正确性验证、功能模拟、性能分析等。功能强大、丰富逼真的仿真元件库、激励源、虚拟仪器、编译软件及各种辅助工具, 把 Proteus 8 建成了一个名副其实的电子设计虚拟实验室。

2. 交互式仿真实例(741 放大电路)

(1) 放大电路分析

图 1-2 给出了由运算放大器 741 组成的放大电路原理图。这不仅仅是一张简单的原理图, 图中除了基本仿真电子元件之外, 还有用于仿真的激励源、虚拟仪器和仿真工具。其中, 运放 741、电阻 R1~R3、 $\pm 12V$ 电源为仿真器件和电源, 即这些器件和电源不仅仅是一个原理符号, 还都有电气特性, 在运行仿真时会显现出和实际元器件一样的电气特性; 在 R1 的左端和 741

的输出端分别接有正弦信号和输出电压探针，这是仿真工具，一个是激励(即输入信号)，一个是响应测试(即输出电压指示)；另外在 741 输出端还接有一个示波器，可以在运行仿真时查看动态波形。

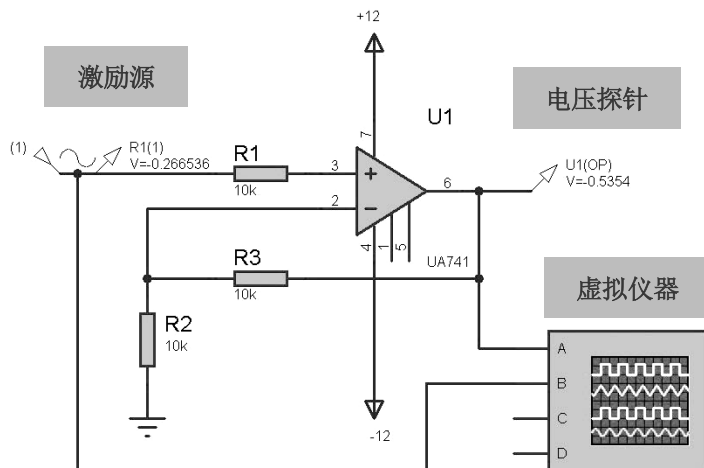


图 1-2 741 放大电路分析

同时，在原理图的空白处还添加并自动生成了两个静态仿真图表，分别用于电路的噪声分析和失真分析。

- 噪声分析：显示随频率变化的输出噪声和等效输入噪声电压，并列出电路各部分所产生的噪声电压清单。741 放大电路的噪声分析如图 1-3 所示。
- 失真分析：用于确定由测试电路所引起的电平失真的程度，失真分析图表用于显示随频率变化的二次和三次谐波失真电平。741 放大电路的失真分析如图 1-4 所示。

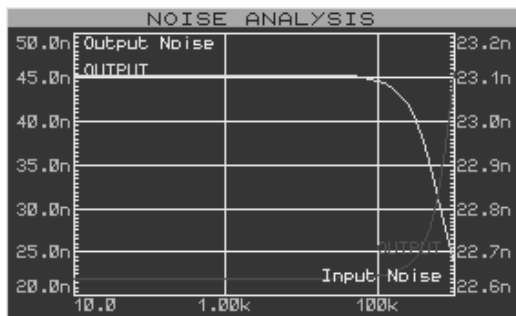


图 1-3 741 放大电路的噪声分析

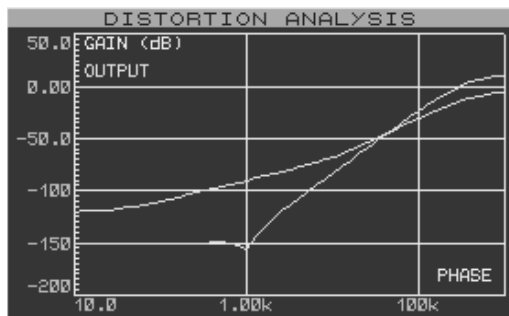


图 1-4 741 放大电路的失真分析

3. 微处理器系统仿真

单片机系统的仿真是 Proteus VSM 的主要特色。用户可在 Proteus 中直接编辑、编译、调试代码，并直观地看到仿真结果。

Proteus 8 元件库中的 CPU 模型有 8051、ARM、AVR、PIC、TMS320 等多种系列及常用外设。同时，原理图元件库中包含了 LED/LCD 显示、键盘、开关、喇叭及常用电机等完整

的仿真器件。VSM 甚至能仿真多个 CPU，它能轻松处理含有两个或两个以上微控制器的系统设计。

下面来看一个微处理器系统仿真与分析实例——交互式仿真显示系统输出结果，如图 1-5 所示。

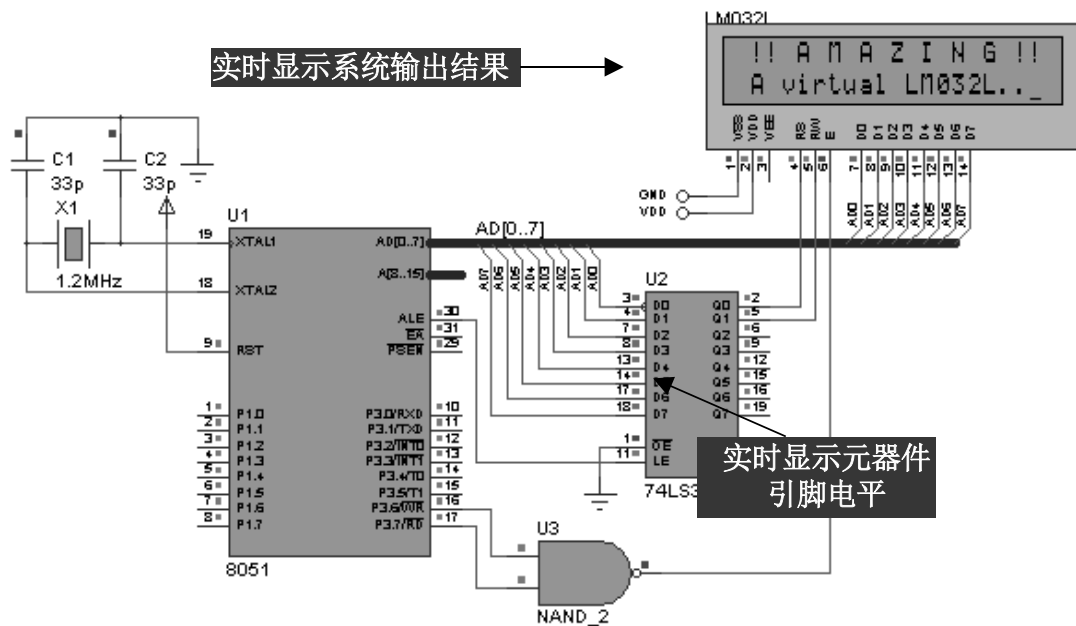


图 1-5 微处理器交互仿真实例

单片机 8051 通过锁存器驱动液晶显示屏。对单片机进行程序设计，在主界面建立工程向导时选择创建固件(Firmware)，则程序代码(Source Code)工作视窗就会建立，可以进行系统程序编写、编译和仿真，编译后的文件通过内部数据库自动导入到原理图的单片机属性中。

当然，如果是 Proteus 7 版本，可通过 Keil 软件(支持汇编和 C 格式)编写程序，编译后生成相应文件，用鼠标双击原理图中的单片机，把编译后的文件载入“Program File”栏即可。

无论采用哪种编程方式，如果程序无误，而且硬件电路也连接正确，则单击原理图设计视窗左下方的仿真控制按钮  可进行电路交互仿真，出现如图 1-6 所示的仿真结果。其中，每个芯片引脚还会通过红蓝两色的方点来表示引脚电平的高低，红色表示高电平，蓝色表示低电平。

另外，通过 COMPIM 串口仿真模型，可以实现虚拟仿真电路与外部实际电路的双向通信，如图 1-6 所示。

原理图中的 P1 为虚拟串口，通过适当设置和引用，可以直接实现模拟与实际电路一样的串行通信效果，避免了涉及外围及与 PC 之间的通信无法实现的情况。

在 Proteus 中，除了虚拟仿真元件库，还有很多虚拟仪器可供使用，因此，单片机与上位机之间的通信，以及两个单片机之间的通信仿真都可以圆满解决。

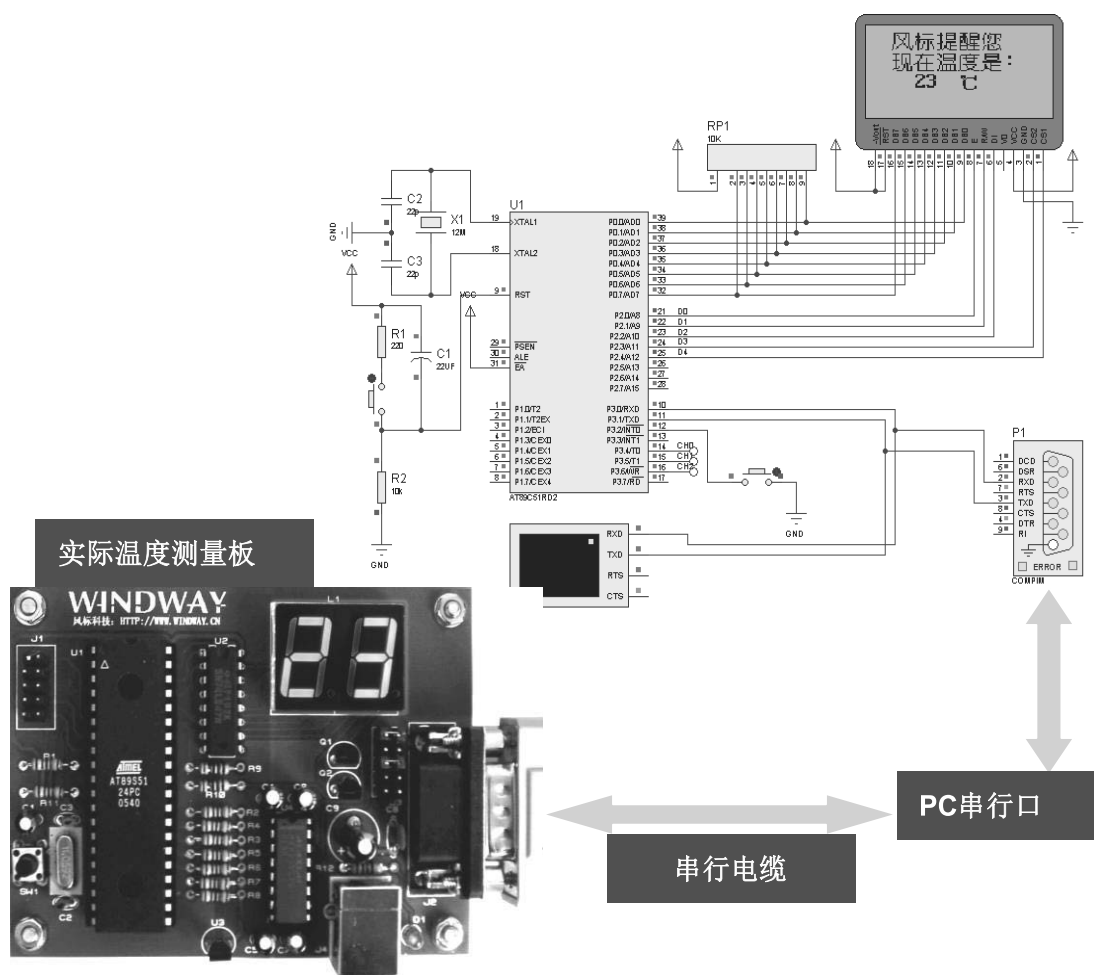


图 1-6 虚拟仿真电路与外部实际电路的双向通信

1.2 Proteus 8 集成环境认识

1.2.1 Proteus 8 Professional 的安装与运行

先按要求把 Proteus 8 Professional 软件安装到计算机上(支持 Windows XP 以上操作系统, Proteus 8.4 开始支持 Windows 10 系统), 安装结束后, 在桌面的“开始”程序菜单中找到“Proteus 8 Professional”选项并单击(或双击桌面快捷键)运行。Proteus 8 Professional 在 Windows 10 程序中的位置如图 1-7 所示。

图 1-8 所示为 Proteus 8 Professional 运行时闪过的软件版本信息显示界面。

接下来打开的是 Proteus 8 Professional 的主界面, 如图 1-9 所示。

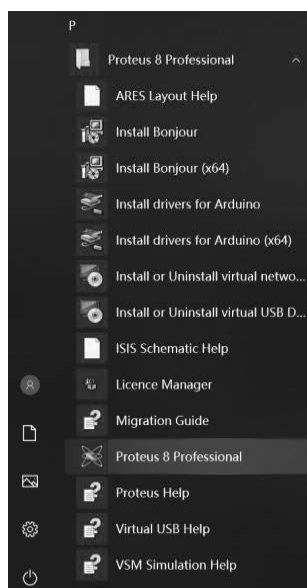


图 1-7 Proteus 8 Professional
在 Windows 10 程序中的位置

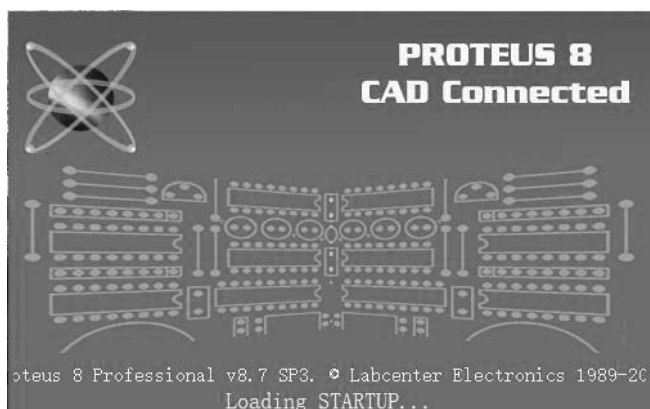


图 1-8 Proteus 8 Professional
运行时的版本信息显示

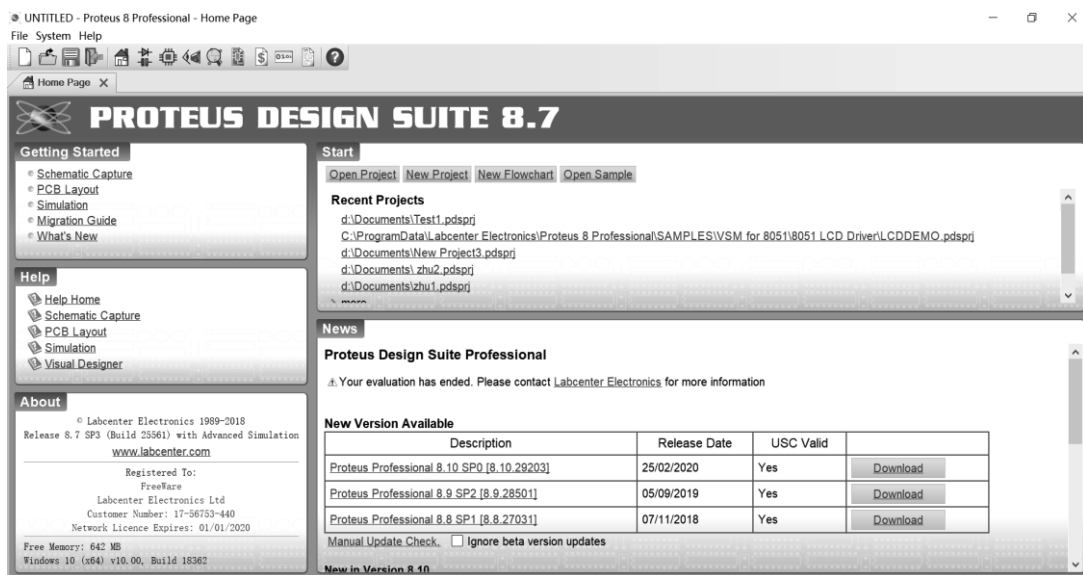


图 1-9 Proteus 8 Professional 的主界面

Proteus 8 Professional 的主界面可分为三个区域。

第一区域：左上角信息及图标栏，这是集成工作视窗的公共部分，分别为工程文件名称、主菜单栏(不同的工作视窗会有变化)、文件操作及工作视窗选项卡、工作视窗标签(首次运行时只有一个主界面视窗标签)。

第二区域：位于左列的三个区域。

- **Getting Started:** 学习教程，图文并茂、非常详细，分别介绍原理图绘制(Schematic Capture)、PCB 设计(PCB Layout)、Proteus 功能概览(Migration Guide)、仿真(Simulation)、目前版本及以前版本软件新增功能(What's New)。如果是初次接触 Proteus 8 Professional，可以先从“Migration Guide”开始学习，借助在线翻译软件阅读非常方便。

- **Help:** 帮助文档，与学习教程相辅相成。

- **About:** 版本信息。

第三区域：位于右列的两个区域。

- **Start:** 建立工程文件向导，可分别选择打开工程(Open Project)、新建工程(New Project)、新建流程图(New Flowchart)、打开样例(Open Sample)。这是我们开始 Proteus 8 设计工作的入口，也是主界面最核心的部分。

- **News:** 最新版本软件下载及新增功能，提供高于所用版本的其他各个版本软件的下载链接，可以看到图 1-9 中有 8.8、8.9、8.10 三个版本。介绍了各个版本的主要新增功能，最下方还提供了在线视频介绍。

1.2.2 Proteus 8 Professional 的快速设计向导

运行 Proteus 8 Professional 后，出现如图 1-9 所示的主界面中的快速设计向导，这是 Proteus 8 快速设计的一个重要组成元素。

1. 查看样例

初学者可以先单击“Start”下方的“Open Sample”按钮，打开 Proteus 8 自带的工程设计样例，从最简单的电容充放电电路到运放和译码器电路，以及各种型号的单片控制电路，非常丰富，便于读者了解和学习。查看样例后千万别保存，直接退出放弃保存，以防止样例更改后被覆盖。

图 1-10 是选择“Open Sample”后打开的样例查看窗口，这里选择“VSM for 8051”类别下的“8051 LCD Display Driver”(8051 液晶显示驱动)样例。

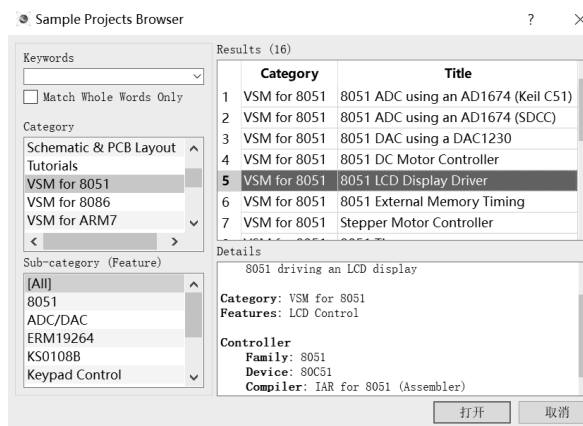


图 1-10 样例工程选择窗口

单击图 1-10 下方的“打开”按钮，出现“8051 LCD Display Driver”工程样例的集成工作视窗界面，当前视窗为原理图设计视窗，其上方对应的视窗标签为  Schematic Capture ，如图 1-11 所示。右边相邻的视窗标签为  Source Code ，即源程序代码工作视窗标签，单击后可把程序代码视窗切换为当前视窗，如图 1-12 所示。单击各工作视窗标签右边的“×”按钮可关闭该工作视窗。

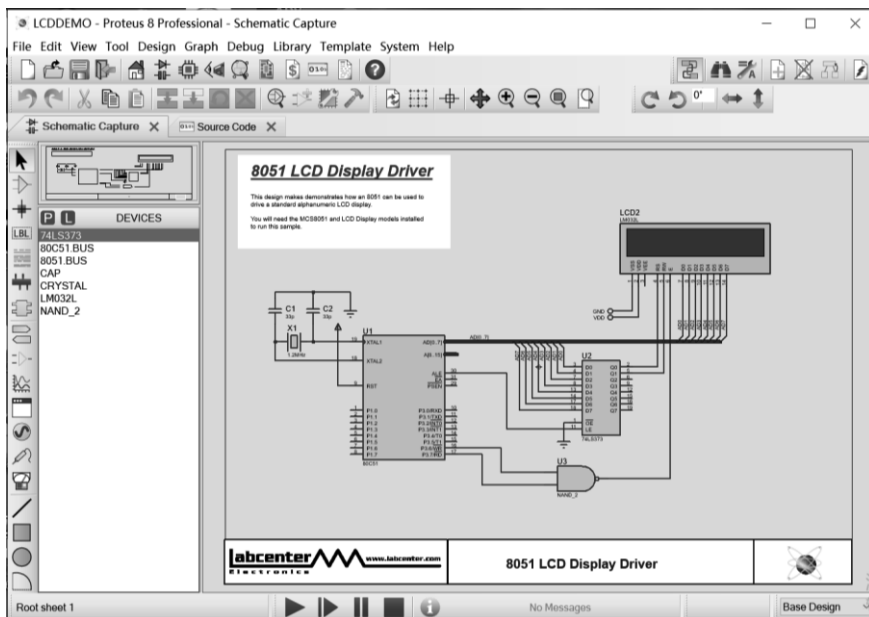


图 1-11 Schematic Capture 工作视窗环境

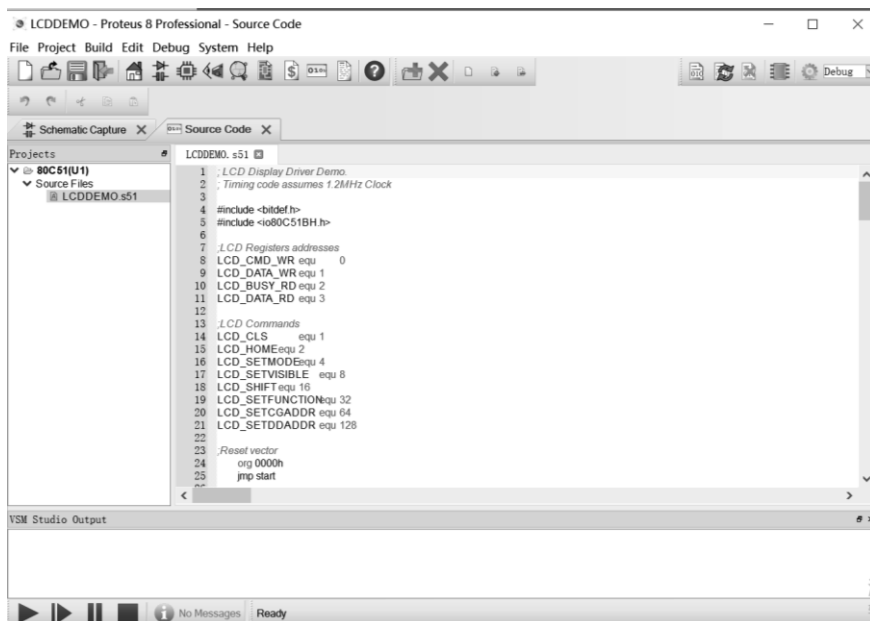


图 1-12 Source Code 工作视窗环境

2. 工作视窗简介

下面就以打开的样例对 Proteus 8 集成工作视窗环境进行简要介绍。

(1) 主菜单命令

主菜单命令与当前工作视窗相关, 选择不同的工作视窗, 主菜单命令也随之变化, 如原理图视窗的主菜单命令为 File Edit View Tool Design Graph Debug Library Template System Help, 而源程序代码视窗的主菜单命令为 File Project Build Edit Debug System Help, 以上两个视窗中的主菜单命令会在第 2 章详细介绍, PCB 视窗的主菜单命令为 File Output View Edit Library Tools Technology System Help(第 8 章详细介绍), 也不一样。

单击主菜单栏中的命令, 出现下拉菜单, 通过选择不同的子命令, 实现对工程或项目设计的存储、编辑、查看、编译、仿真等功能。重要的主菜单命令在不同的工作视窗下会以操作图标的形式放置在视窗上方。

(2) 系统主操作图标

主菜单栏下的一行固定图标, 是 Proteus 8 系统的主要操作图标, 不因当前工作视窗的切换而改变, 分别为工程文件操作类、视窗开启类和帮助。

从左到右前四个图标  为文件操作类, 单击后分别进行新建工程、打开工程(包括以前版本的非工程类设计文件)、存储工程和关闭工程的操作。

接下来的九个图标  为开启工作视窗选项卡, 其中第一个图标为切换至主页面(软件运行时首先开启主页), 后面依次为开启原理图设计工作视窗(创建向导中必创建此视窗, 因此也已经自动打开了)、PCB 设计视窗(如在创建向导中创建则自动打开)、三维可视化视窗(PCB 的 3D 效果生成)、Gerber 查看器(查看 PCB 各层设计效果)视窗、设计资源管理器视窗、物料清单视窗、程序代码视窗(如在创建向导中创建则自动打开, 显示或编写单片机程序及仿真模型代码)和工程备注视窗(文字输入)。工作视窗选项卡和工作视窗标签有明显区别, 前者主要为开启一个新的工作视窗, 后者为自动标记已打开的工作视窗、切换当前工作视窗、关闭对应的工作视窗。

最后一个是  图标, 单击可弹出帮助界面。

(3) 主命令图标和主工具图标

主命令图标是主菜单命令中的主要及常用命令以图标的形式放置在了工作视窗的上方。不同的当前工作视窗中, 主命令图标也不尽相同。主工具图标是不同的工作视窗下的主要工具, 通常指的是原理图设计工作视窗和 PCB 设计工作视窗, 这些工具图标位于工作视窗的左侧。

下面, 我们开启样例中主操作图标中的其余工作视窗选项卡, 再分别切换至当前视窗, 熟悉不同的工作视窗环境, 观察主命令图标、主工具图标等的变化, 如图 1-13~图 1-17 所示(图标栏的位置可以通过鼠标将其拖动到其他位置)。

另外, 还有 3D Visualizer 三维可视化工作视窗, 由于该样例工程中没有创建 PCB 项目, 因此是空的。

在图 1-11~图 1-17 的任何一个工作视窗环境下,单击主操作图标栏中的图标,都可返回如图 1-18 所示的 Proteus 8 主界面,以查找帮助、进行软件教程学习,同时还可以在主界面中单击其他工作视窗标签,再次回到任一已打开的工作视窗界面,如图 1-19 所示。单击主操作图标栏中的关闭工程图标,则关闭除主界面之外的所有已打开视窗,并返回如图 1-9 所示的主界面。

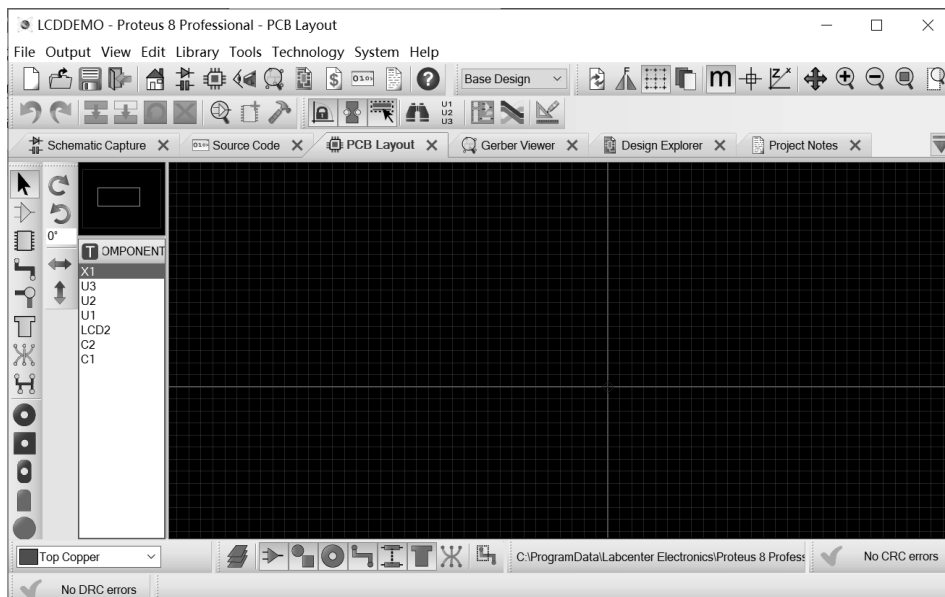


图 1-13 PCB Layout 工作视窗环境

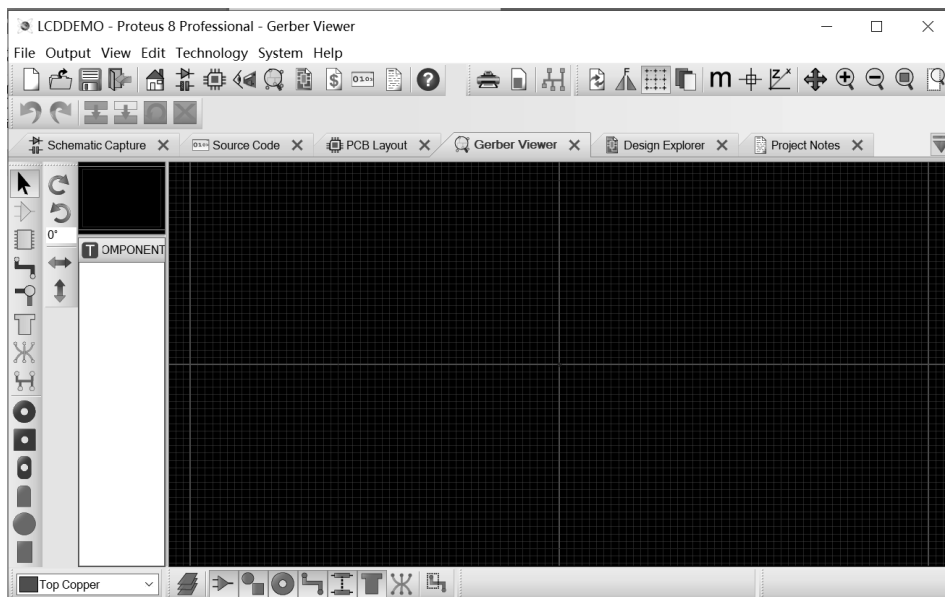


图 1-14 Gerber Viewer 工作视窗环境

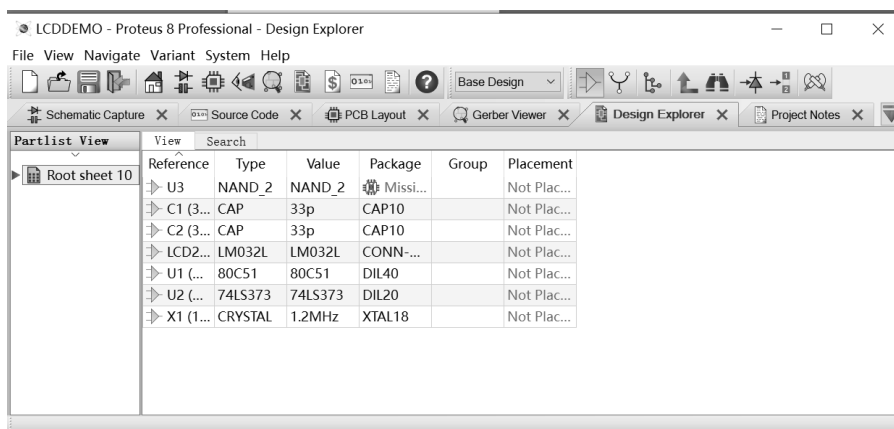


图 1-15 Design Explorer 工作视窗环境

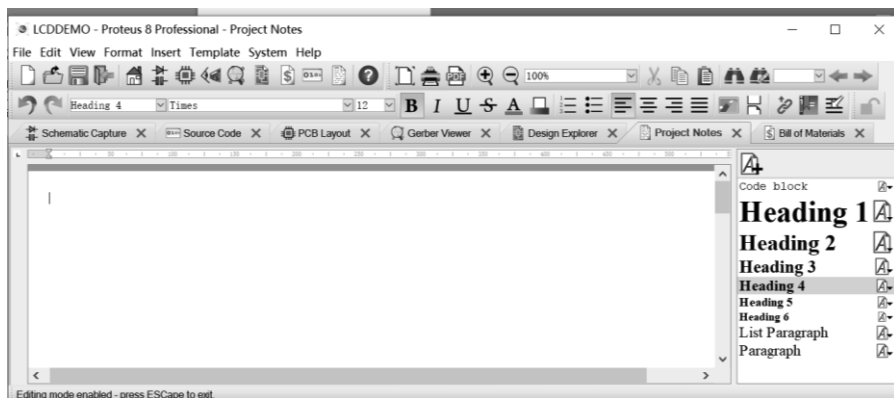


图 1-16 Project Notes 工作视窗环境

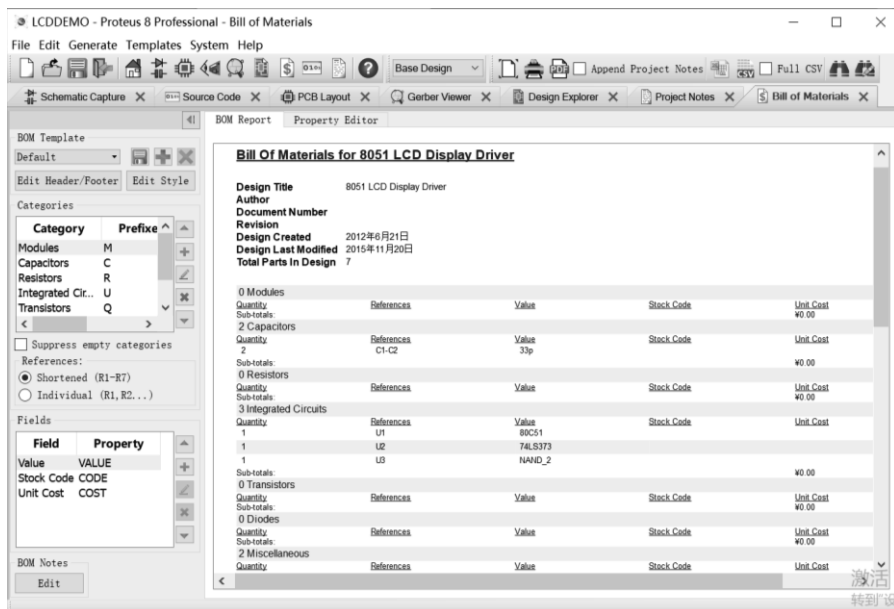


图 1-17 Bill of Materials 工作视窗环境

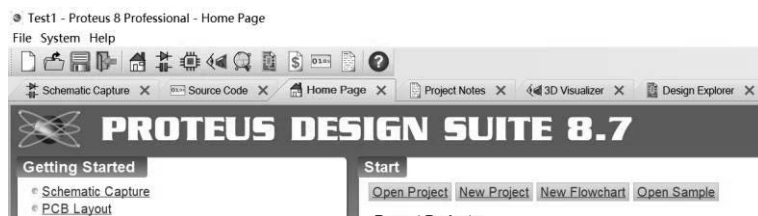


图 1-18 返回主界面

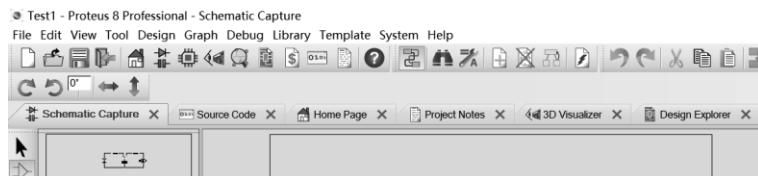


图 1-19 切换至原理图工作视窗

3. 打开工程项目

在图 1-9 的创建向导“Start”中，单击“Open Project”选项打开一个存在的文件，弹出的对话框如图 1-20 所示。可从已知目录中查找选择工程项目文件，默认文件类型为 Project Files(个人工程文件的存放路径，是在创建工程文件时指定的，请读者记好)，也可以通过下面的文件类型选择 Proteus 7 及早期版本中的原理图设计文件(Design Files)、PCB 设计文件(Layout Files)，以及备份的工程文件(Backup Project Files)。读者可以尝试操作一下，此处不再详述。

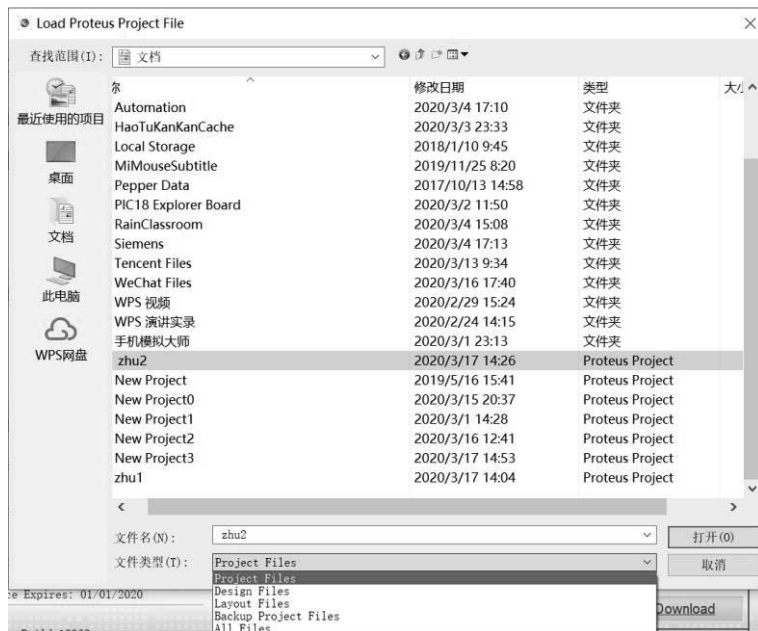
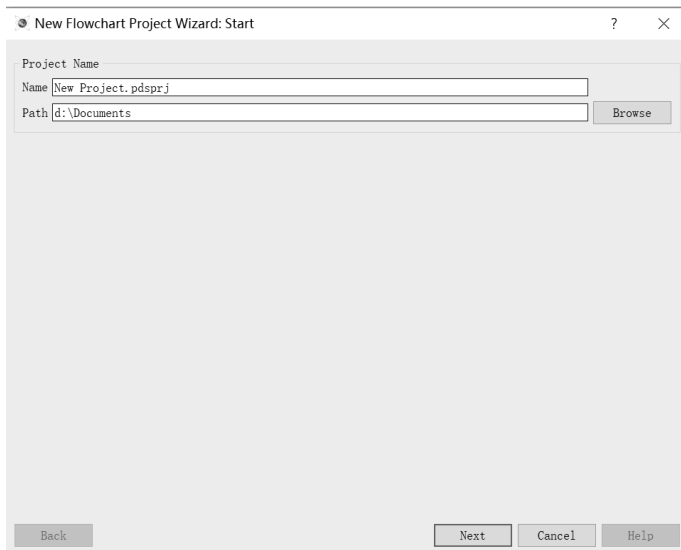


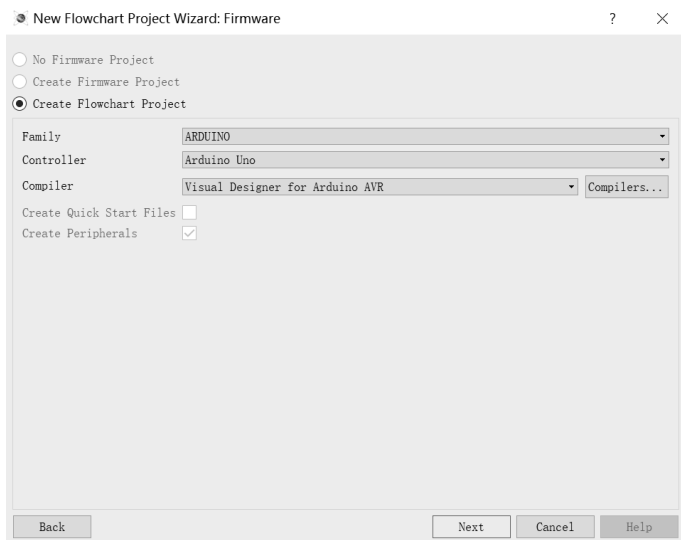
图 1-20 打开工程项目对话框

4. 创建流程图工程

先通过创建向导来了解一下流程图的含义。在图 1-9 的创建向导“Start”中，单击“New Flowchart”选项，新建流程图工程，弹出的对话框如图 1-21(a)和(b)所示。图 1-21(a)中，工程文件名称和存放路径可以更改和指定，但工程文件的扩展名“.pdsprj”不能更改；图 1-21(b)中，为工程文件指定固件(用户程序)对应的控制器类型和编译器，通过下拉菜单可以看到，Flowchart 设计主要针对 ARDUINO 控制器，编译器也是默认的。



(a) 输入工程名称



(b) 指定固件相关参数

图 1-21 新建流程图工程对话框

单击图 1-21(b)中的“Next”按钮，出现如图 1-22 所示的流程图工程创建内容总结(Summary)。可以看到，工程中含有原理图(Schematic)和固件(Firmware)项目设计，并没有 PCB(Layout)项目设计。

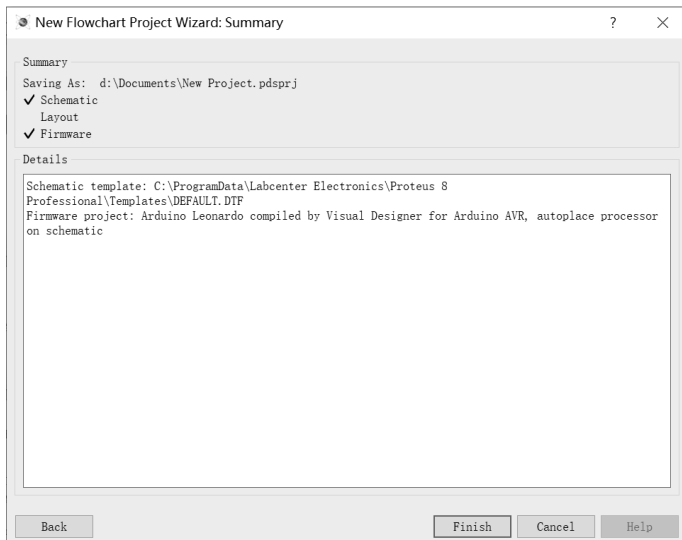


图 1-22 流程图工程创建内容总结

单击“Finish”按钮打开如图 1-23 所示的集成工作视窗界面，当前工作视窗标签为“Visual Designer”，即可可视化设计。而随着后面的学习会发现，同一个程序设计视窗选项卡，打开的工作视窗及标签都不尽相同，这与编程方式、编译器和处理对象有关。

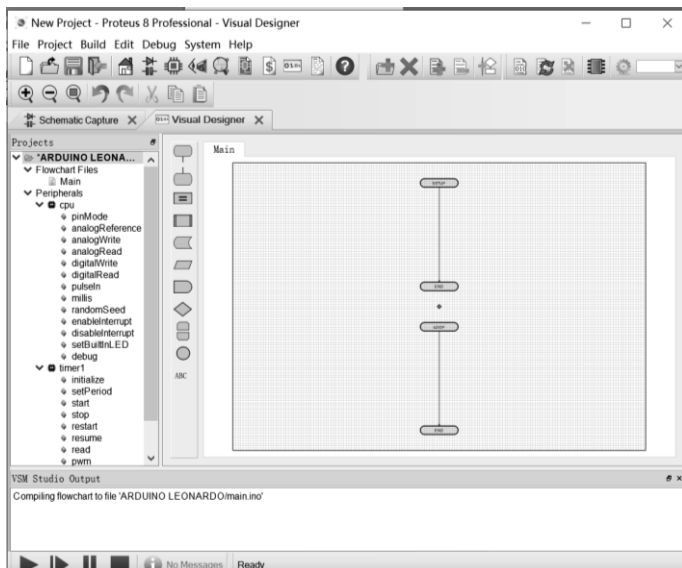


图 1-23 流程图工程的程序设计视窗

该视窗由三块区域组成，左边是程序目录结构，右边是程序流程图编辑窗口。程序目录和框架已经构建好了，读者可以通过左侧的工具来编写流程图程序，这是 Proteus 8 程序可以

快速设计的又一个组成元素。下边区域是程序编译和仿真信息栏。

因此, Proteus 8 的流程图是针对 ARDUINO 控制器的一种程序编写方式, 更详细的内容可查看其他相关资料。

单击图 1-23 中的“Schematic Capture”视窗标签, 可打开如图 1-24 所示的原理图设计工作视窗。可以看到, 在原理图设计视窗下已经加载了创建流程图工程时指定的单片机最小系统, 这是 Proteus 8 ISIS 硬件快速设计的一个组成元素。

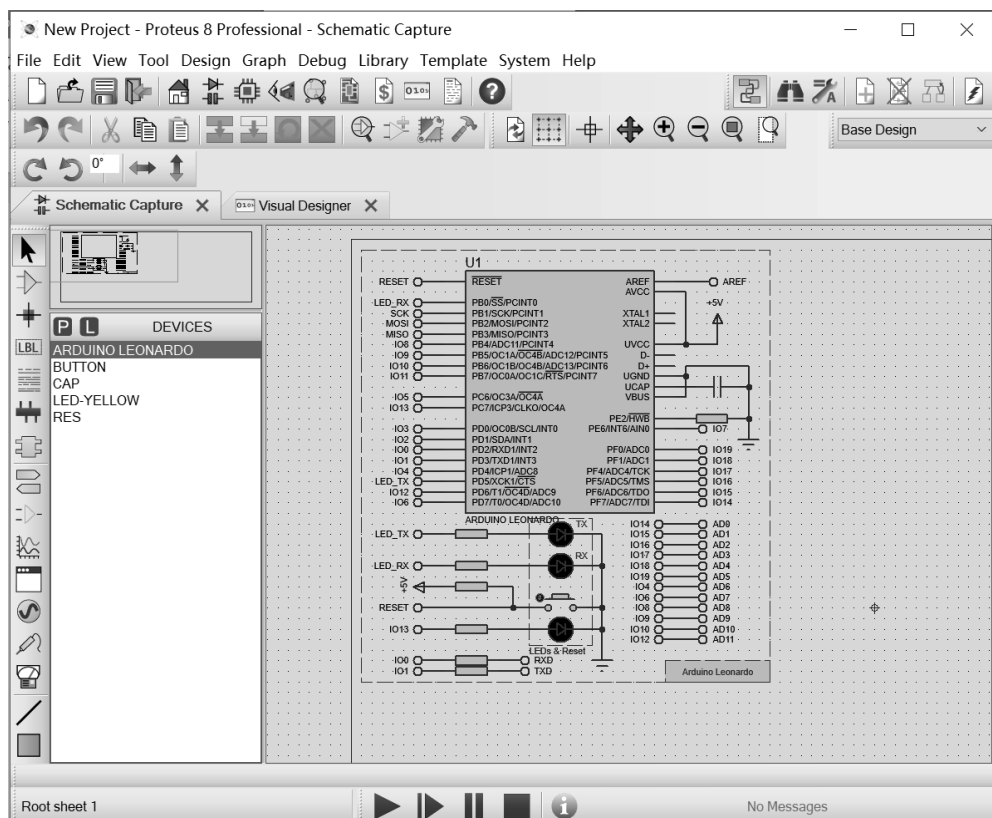


图 1-24 流程图工程原理图设计视窗

5. 创建新的工程

接下来讲重点, 即创建新的工程, 步骤如下。

(1) 返回 Proteus 8 主界面, 选择“New Project”, 打开如图 1-25 所示的创建向导对话框。该图和图 1-21(a)看似一样, 但其实路径下面多了三个选项: 新工程(New Project, 默认)、来自开发板(From Development Board)、空的工程(Blank Project, 不可选), 一般情况下默认为新工程。

(2) 把工程名称改为“Test1”(扩展名.pdsprj 不变), 文档存放路径先不变, 如图 1-26 所示。读者也可以浏览选择其他目录存放用户工程文件, 但一定要记清楚, 以便以后浏览。在主界面创建向导下面, 会显示最近打开过的工程文件。

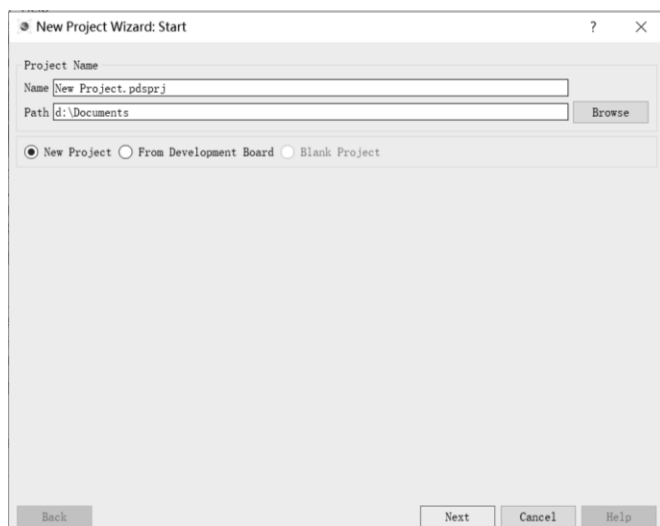


图 1-25 新工程创建向导对话框

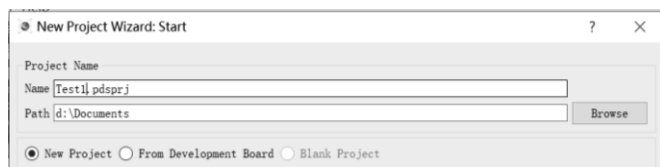


图 1-26 修改工程名称

(3) 单击图 1-25 中的“Next”按钮，出现如图 1-27 所示的创建原理图设计对话框。有两个选项：不创建原理图、创建原理图(默认)。原理图设计是一个工程的最基础工作，因此必须选择此项。接下来是原理图设计模板选择，默认为“DEFAULT”。

(4) 单击“Next”按钮，出现如图 1-28 所示的创建 PCB 布局对话框，默认为不创建。如果选择创建 PCB 布局，后面还有一系列选项，读者如果想看一下创建后的集成视窗效果，一路单击“Next”按钮即可。

(5) 单击图 1-28 中的“Next”按钮，出现如图 1-29 所示的创建固件对话框。有三个选项，分别是“No Firmware Project”(不创建固件)、“Create Firmware Project”(创建固件)和“Create Flowchart Project”(创建流程图)。如果绘制一个不含控制器的电子线路且不需要进行复杂模型创建，选择默认不创建固件项目；如果想要设计一个含有控制器的系统或需要进行复杂模型创建，即有编程需求，则选择后两项；而第三项流程图创建前面已有说明，这里选中间项“Create Firmware Project”。

(6) 选择“Create Firmware Project”项后，在图 1-29 下方选择需编程的控制器系列、型号、对应的程序编译软件，以及是否建立快速开始文件等。在“Family”中保留 8051 选项(通过右边的下拉箭头可选择其他 CPU 系列)，在“Controller”中选择 80C51 单片机，在“Compiler”中保留默认项“ASEM-51 (Proteus)”编译软件。如果选择的是其他控制器，要在图 1-30 所示的“Compiler”对话框中选择与之对应的编译软件；如果系统没有自带，可以直接在线单

击“Download”安装。在图 1-30 中可以看到，已经手动安装了一个“MASM32”8086 系列编译软件。

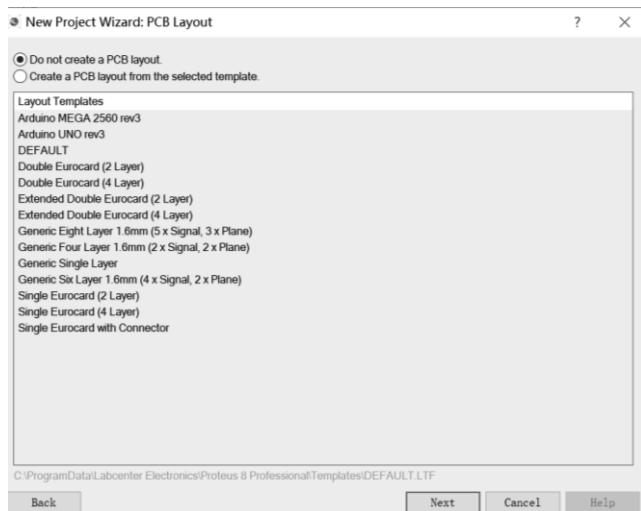


图 1-27 创建原理图设计对话框

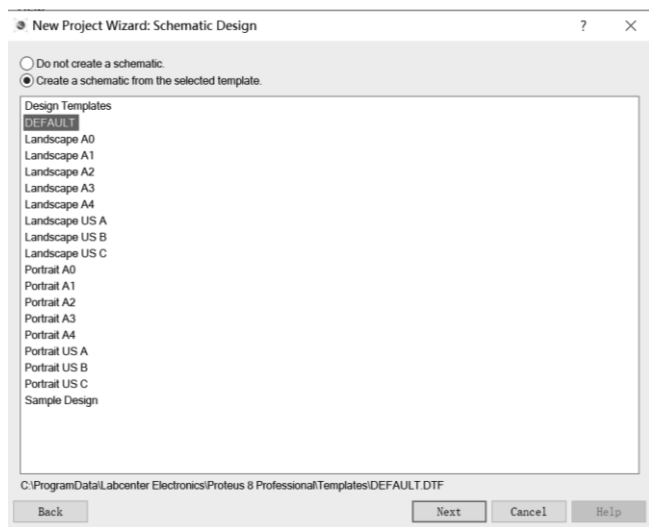


图 1-28 创建 PCB 布局对话框

(7) 选择图 1-29 中的“Create Quick Start Files”项并进行相应设置后，出现如图 1-31 所示的创建固件选择结果。

(8) 选择“Create Firmware Project”项后，单击对话框下方的“Next”按钮，出现 Test1 工程各项目创建概要，详细信息如图 1-32 所示。

(9) 单击图 1-32 下方的“Finish”按钮，完成 Test1 工程创建向导，打开集成工作视窗界面，如图 1-33 所示。当前工作视窗为“Source Code”，这是前面选择固件的结果，左边是程序文件目录结构，右边是程序编辑区，下面是仿真和编译信息显示区。可以看到程序设计视窗中已有相关程序体语句，这是我们前面勾选“Create Quick Start Files”的结果，否则是

空白。因此 Proteus 8 不再依赖外部程序编译器，这种集成化的设计为程序调试和硬件仿真带来了非常大的便利，后面例子中会详细说明活动窗口的作用。

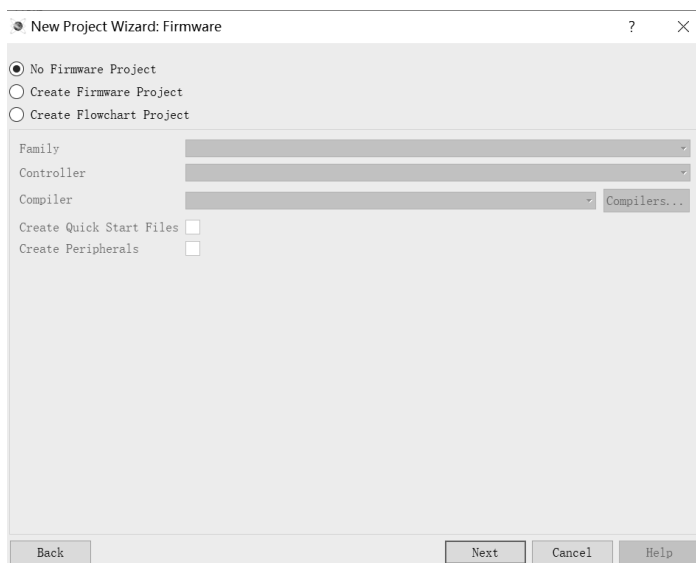


图 1-29 创建固件对话框

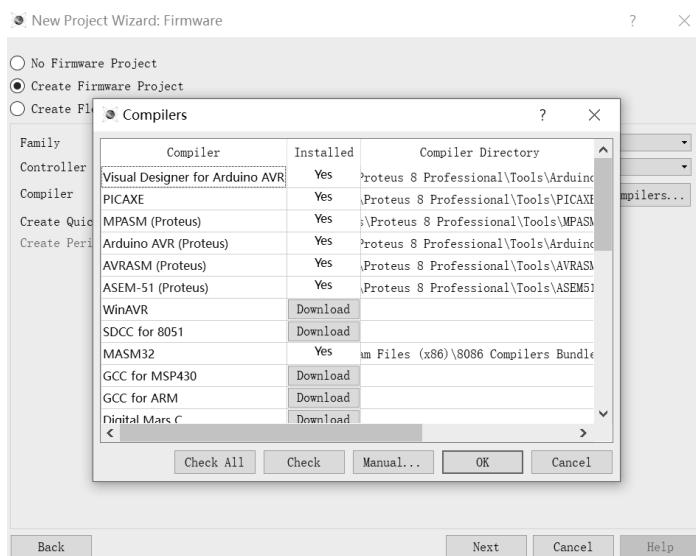


图 1-30 编译器选择对话框

(10) 切换图 1-33 中的当前视窗为原理图设计视窗，可看到单片机 80C51 元件已经自动加载到了绘图区，如图 1-34 所示。

至此，工程文件创建向导已经完成，我们已经来到了集成工作视窗界面，先单击左上方的存盘图标保存一下已建工程。接下来，我们一起动手设计三个简单工程，使读者通过练习，能够快速上手，开启直通 Proteus 8 Professional 的大门。

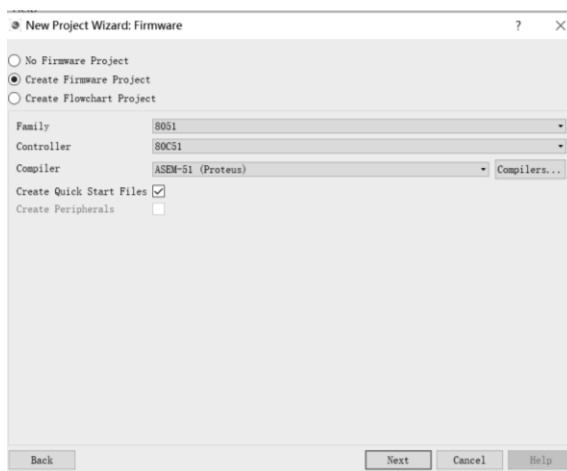


图 1-31 创建固件选择结果

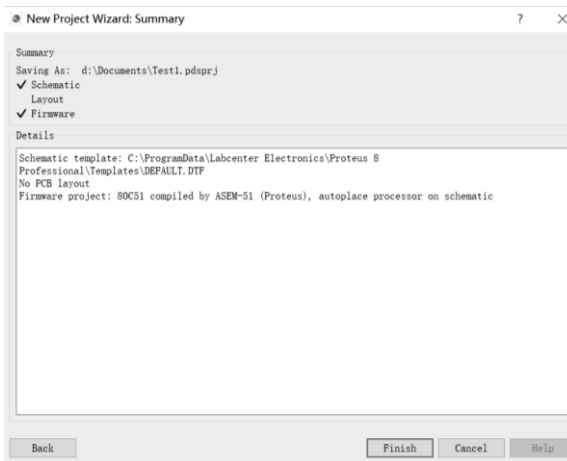


图 1-32 Test1 工程创建概要

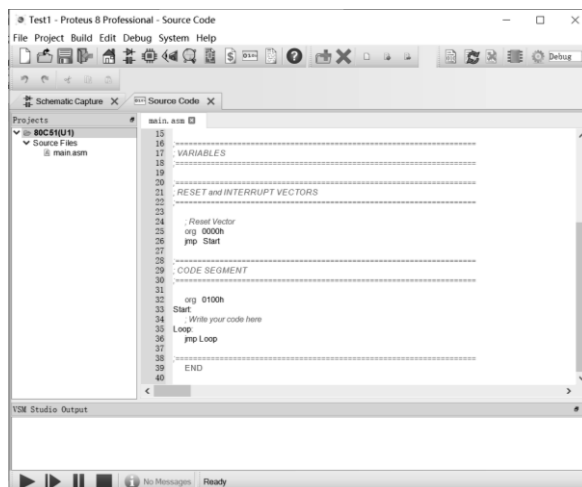


图 1-33 程序设计工作视窗

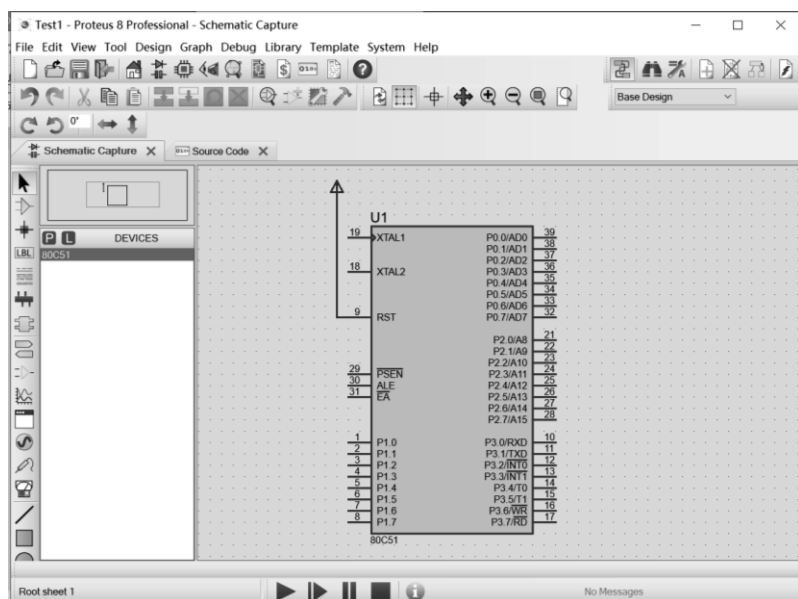


图 1-34 原理图设计工作视窗

1.2.3 一阶动态电路的设计与仿真

先从最简单的工作入手，来设计一个一阶动态电路——电容充放电电路，并通过仿真，观察电流流向和灯泡的亮灭。为了方便，直接在刚刚创建的工程项目 Test1 的原理图设计工作视窗中进行。首先做一些最基础的准备工作。

1. 认识原理图设计工作视窗

图 1-35 所示为 Proteus ISIS 原理图设计工作视窗。最左侧是绘图工具图标栏；右边为绘图区或原理图编辑区；左上方为对象列表区中选中对象形状的预览窗口，或绘图区视野平移，单击此区域后拖动，可平移绘图区的图纸，移动的蓝框为出现在绘图区的视野，绿框为绘图区的图纸边界，蓝框移动时观察右边绘图区的视野变化，合适时再次单击；左下方为对象列表区，选择不同的绘图工具图标，这里放置的对象类别会不同。例如，当选中元件模式  时，对象列表区出现的是元件名称；当选择总线模式  时，对象列表区出现的是放置总线相关的部件名称。绘图时，可把不同类别的对象，从对象列表区的名称符号中复制到绘图区变成图形符号，而且可以复制多个。

2. 元件的删除

先把图 1-35 中不需要的单片机元件 80C51 删除。右键快速单击 80C51 元件两次即可删除，会发现相关连线也没了，不过还剩下一个电源符号——箭头；还可用另外一种方法删除，将鼠标指针对准箭头单击鼠标右键，在打开的菜单中选择“Delete”，元件删除后，导线自动消失。



图 1-35 原理图设计视窗分区

现在练习块删除。单击图 1-34 中左上角的撤消按钮两次，恢复单片机、电源及连线；单击单片机图形左上方，拖曳出一个包含整个电路图的矩形，松开左键，在红色选区域内单击右键，出现右键菜单，选择“Block Delete”即“块删除”，则选中区域内的电路被全部删除，如图 1-36 所示。更方便的是，选中要删除的区域后，直接单击上方的块删除工具图标即可快速删除整个块。

强调一下，右键菜单不是恒定不变的，即使在同一个工作视窗下，也会根据鼠标右键点选的不同对象而不同。

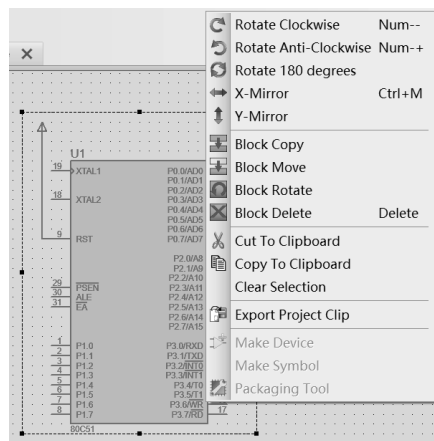


图 1-36 右键块删除指令

3. 元件的拾取

本例所用到的电路元件清单如表 1-1 所示。

表 1-1 一阶动态电路的元件清单

元件名	所在类	子类	备注	数量	参数
CAPACITOR	Capacitors	Animated	电容，可动态显示电荷	1	1000 μ F
RES	Resistors	Generic	电阻	2	1k Ω , 100 Ω
LAMP	Optoelectronics	Lamps	灯泡，会发光、灯丝会断	1	12V
SW-SPDT	Switches and Relays	Switches	两位开关，可单击操作	1	
BATTERY	Simulator Primitives	Sources	电池组	1	12V

首要任务是把表 1-1 中的所有元件从 Proteus ISIS 的元件库中拾取出来，放置到当前原理图设计视窗中。首先选中图 1-35 左侧主工具栏中的元件模式图标，再单击拾取元件(Pick Devices)图标或直接在键盘上按快捷键“P”，打开如图 1-37 所示的元件拾取对话框。

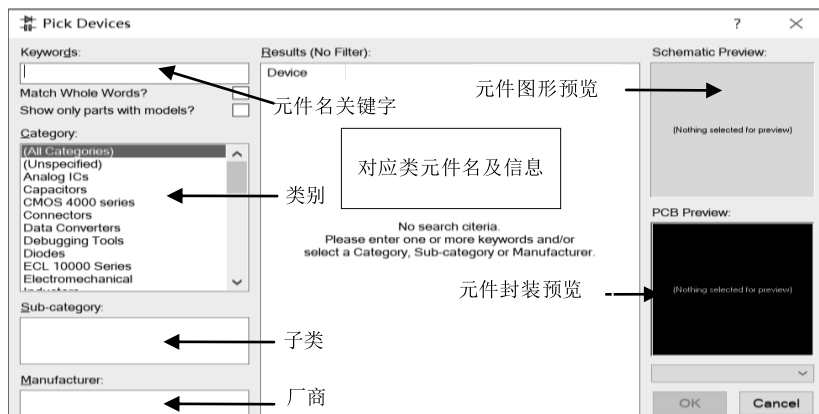


图 1-37 元件拾取对话框

元件拾取就是把所有用到的元件，通过元件拾取对话框一一从元件库的分类中找出来，放置到原理图设计视窗的对象列表区(DEVICES)中。这就像我们在实际连接电路时，先从元件柜的分类抽屉中把各种元件找出来，放到一个盒子里，然后在连接电路时直接从盒子里取出要用的元件，而不必每连接一个元件都要走到柜子边去找一次。

拾取元件主要有两种途径。

(1) 按类别查找拾取元件

Proteus ISIS 的元件是以其英文名称或元件代号在库中存放的。如果想快速拾取一个元件，首先要清楚它属于哪一个大类，然后还要知道它归属于哪一个子类，如果知道生产厂家就更好了，这样就缩小了查找范围。

按照表 1-1 中的顺序来依次拾取元件。首先是充电电容 CAPACITOR，在图 1-37 中打开的元件拾取对话框中，在“Category”类中选中“Capacitors”电容类，在下方的“Sub-category”子类中选中“Animated”(可动画演示)，中间的查询结果列表中只有一个元件，即要找的 CAPACITOR。双击元件名，该元件便已经拾取到对象列表区中了，在图 1-38 的 DEVICES 区可以看到多了一个 CAPACITOR。选择一个元件后单击右下角的“OK”按钮，元件拾取对话框关闭。连续拾取元件时不要单击“OK”按钮，直接双击元件名可继续。

拾取元件对话框共分四部分，左侧从上到下分别为直接查找时的名称输入和分类查找时的大类列表、子类列表、生产厂家列表；中间为查到的元件信息列表；右侧自上而下分别为元件的原理图和封装，图 1-38 中的元件没有显示封装。

(2) 直接查找拾取元件

把元件名的全称或部分输入到“Pick Devices”(元件拾取)对话框的“Keywords”栏中，在中间的查找结果“Results”中显示所有符合条件的元件列表，用鼠标拖动右边的滚动条，出现灰色标示的元件即为找到的匹配元件。图 1-39 所示为电阻元件 RES 的直接查找结果。

有时需要在元件列表区上下滚动, 找到灰色标记的元件后双击。

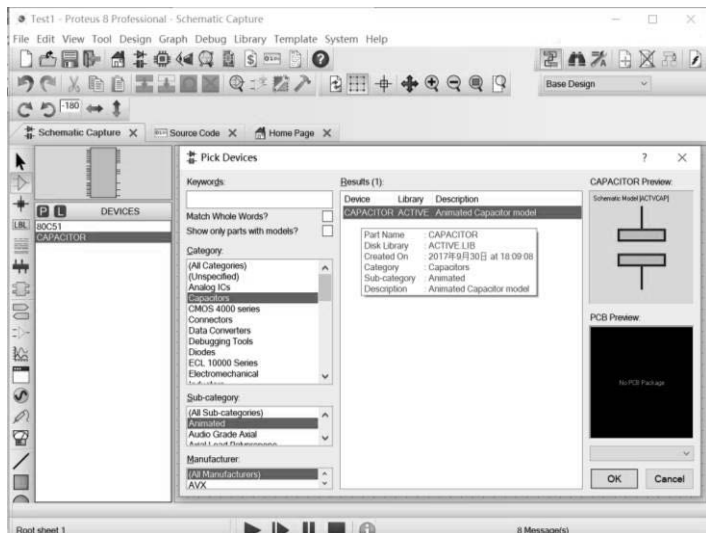


图 1-38 分类拾取元件示意图

这种方法主要用于对元件名熟悉之后, 为节约时间而直接查找。对于初学者来说, 还是分类查找比较好, 一是不用记太多的元件名, 二是对元件的分类有一个清楚的概念, 利于以后对大量元件的拾取。

按照以上两种方法中的任一方法, 把其余三个元件依次拾取到对象列表区中, 然后关闭元件拾取对话框。元件拾取后的界面如图 1-40 所示(在对象列表区中可通过右键把 80C51 删除)。

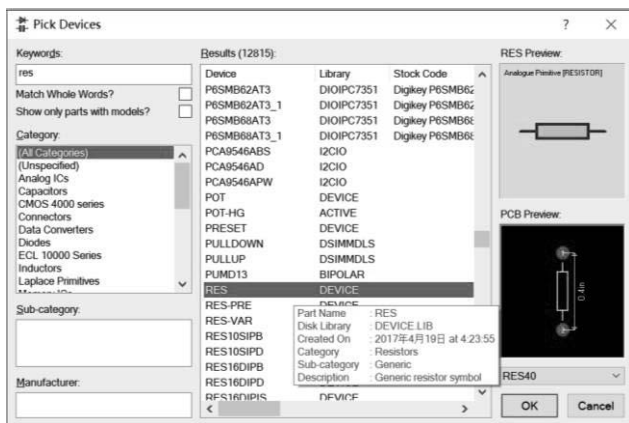


图 1-39 直接查找电阻 RES 结果

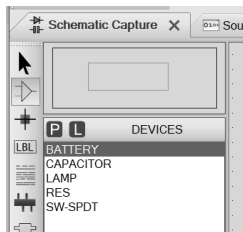


图 1-40 元件拾取后的界面

下面从对象列表区中选取元件放置到绘图区中。单击对象列表区中的某一元件名, 把鼠标指针移动到图形编辑区, 双击鼠标左键, 元件即被放置到编辑区中; 也可单击后拖动到合适的地方再次单击。电阻要放置两次, 因为本例中用到两个电阻。放置后的界面如图 1-41 所示。

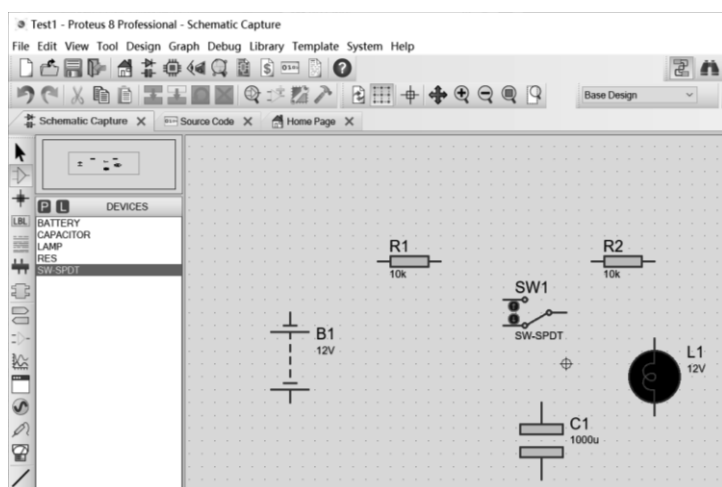


图 1-41 元件放置后的界面

元件拾取及布局结束后，记着单击主工具栏中的选择模式图标，取消元件模式，为下面的工作做好准备，这一点是初学者应该牢记的。

4. 绘图区视野控制

学会合理控制绘图区的视野是元件编辑和电路连接进行前的首要工作。

绘图区的视野平移可用以下方法：

- 在原理图编辑区的蓝色方框内，把鼠标指针放置在某处后，按下“F5”键，则以鼠标指针为中心显示图形。
- 当图形不能全部显示出来时，按住“Shift”键，移动鼠标指针到上、下、左、右边界，则图形自动平移。
- 快速显示想要显示的图形部分时，把鼠标指针指向左上预览窗口中的某处，并单击鼠标左键，则绘图区内的图形自动移动到指定位置。

绘图区的视野缩放用以下方法：

- 先把鼠标指针放置到绘图区内的蓝色框内，上下滚动鼠标滚轮即可缩放视野。如果没有鼠标滚轮，可使用图标来放大、缩小、查看整个图纸，以及选定绘图区域查看。
- 放置鼠标指针到绘图区内想要放大或缩小的地方，按“F6”（放大）或“F7”（缩小）键放大或缩小图形，按“F8”键显示整个图形。
- 按住“Shift”键，在绘图区内单击鼠标左键，拖出一个欲显示的窗口。

5. 元件位置调整和参数修改

在绘图区的元件上单击鼠标左键选中元件(为红色)，在选中的元件上再次单击鼠标右键则删除该元件，而在元件以外的区域内单击右键则取消选择。元件误删除后可用图标找回。单个元件选中后，单击鼠标左键不松可以拖动该元件；群选则可使用鼠标左键拖出一个选择区域，使用图标来整体移动。使用图标可整体复制，图标用来刷新图面。

按图 1-42 所示元件位置布置好元件。使用界面左下方的四个图标及中间的角度输入框 ，可改变元件的方向及对称性。把两位开关调整成图示的方位。

先单击一下左上方的存盘图标 ，在设计过程中应养成随时存盘的好习惯，以免文件丢失。下面来试着改变元件参数。

分别双击绘图区中的电阻 R1 下面的“10k”字样和 R2 电阻符号，可用不同的方式对电阻值进行修改。把 R1 的 Resistance(阻值)由 10kΩ 改为 1kΩ，把 R2 的阻值由 10kΩ 改为 100Ω(默认单位为 Ω)，如图 1-43 所示。

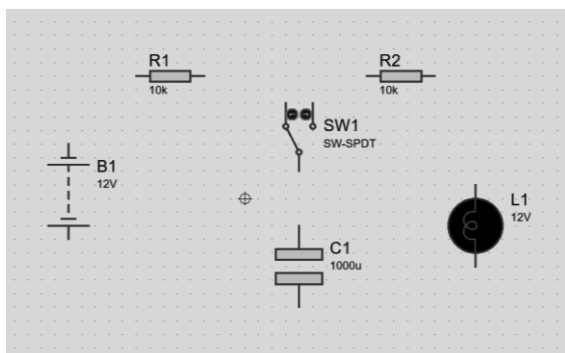
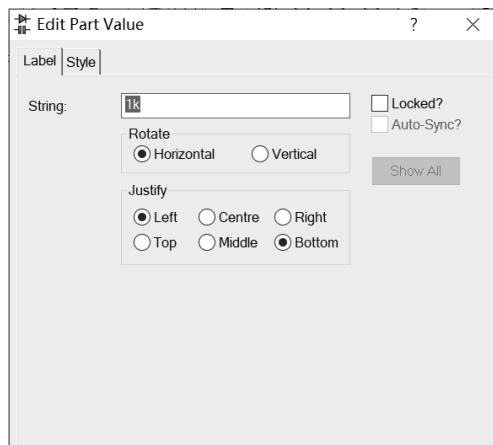
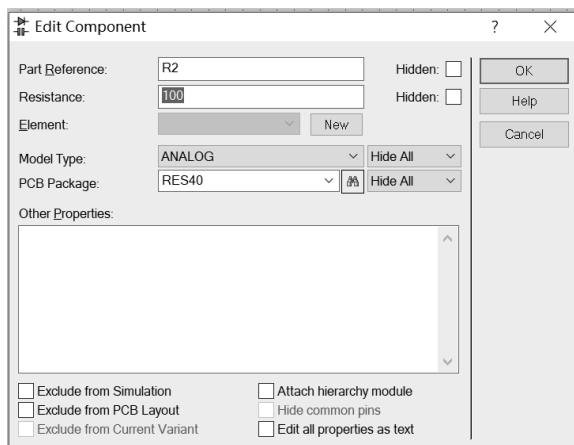


图 1-42 元件布置



(a) 双击电阻值进行修改



(b) 双击电阻进行修改

图 1-43 元件属性设置对话框

6. 电路连线

Proteus ISIS 的连线非常智能化，采用按背景格点捕捉和自动连线的形式，无须选择任何连线工具图标，更不要试图去选左侧工具栏中的  图标来连线(它没有元件引脚捕捉功能，重要的是这个画线工具画出的线不是导线，它和下方的其他二维绘图工具一起，用于在电路中画二维平面图)。只要把鼠标指针指向任何一个元件引脚，该引脚上立即出现一个小的红色捕捉光标，它会判断你下一步的操作是否想连线从而自动连线，而不需要选择连线的操作，此时只需用鼠标左键单击编辑区元件的一个端点并拖动到要连接的另外一个元件的端点，松开左键后再单击鼠标左键，即完成一根连线。如果要删除一根连线，右键双击连线即可。如果对连接好的线不满意，可以单击选中后拖动。

工具图标  是自动布线选择，默认为打开，即按 90° 拐角布线；如果需要折线，则单击该工具图标取消垂直布线即可。单击  图标可切换绘图区背景为点阵、方格或无背景，连续

单击两次则取消背景格点。完成后的电路如图 1-44 所示。

连线完成后，如果还想回到拾取元件状态，单击左侧工具栏中的“元件拾取”图标即可。记着再次存盘。

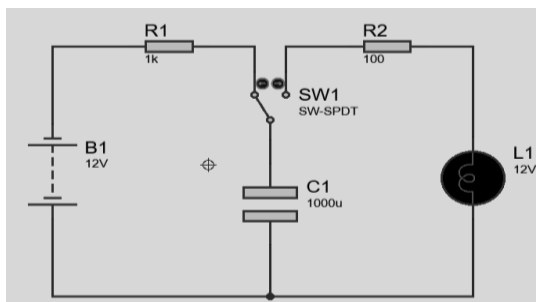


图 1-44 连接好的电路原理图

7. 电路的动态仿真

前面已经完成了电路原理图的设计和连接，下面来看看电路的仿真效果。

首先在主菜单中选择【System】→【Set Animation Options】菜单项，设置仿真时电压、电流的颜色仿真及电流的方向显示，如图 1-45 所示。在随后打开的对话框中勾选“Show Wire Voltage by Colour”和“Show Wire Current with Arrows”两项，即选择导线以红、蓝两色来表示电压的高低，以箭头标示来表示电流的流向。

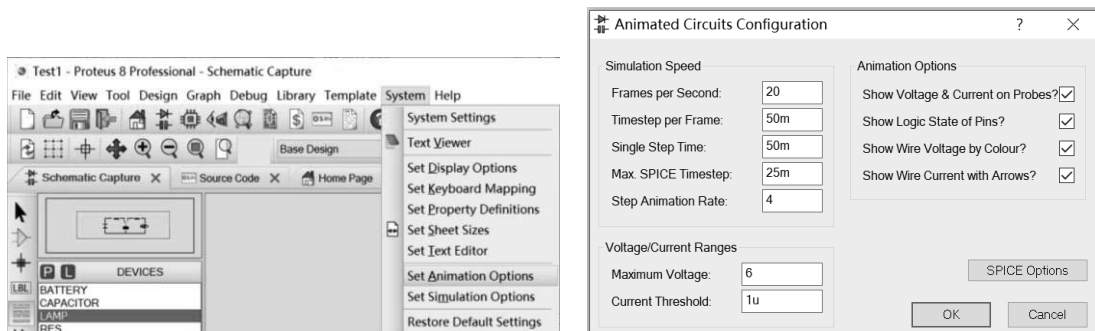


图 1-45 System 主菜单及 Animated Circuits Configuration 对话框

单击左下方的仿真控制按钮中的运行按钮，开始仿真。仿真开始后，单击图中的开关，使其先把电容与电源接通，能清楚地看到电容充电的效果，如图 1-46 所示。

在仿真状态下，接着单击开关，把电容与灯泡连通，看到灯泡闪了一下，如图 1-47 所示。由于充电时间常数为 1 秒，放电时间常数小一些，瞬间放电，所以灯亮的时间很短。如果放电时间常数再大，则不易观察到灯亮的效果。在仿真状态下，可以来回拨动开关，反复观察充放电过程。单击仿真控制按钮中的停止按钮，仿真结束。

8. 变式演练

下面来尝试自己动手绘制一个电容充放电电路，如图 1-48 所示。与刚才的电路不同的是，

这次选用了两个一位开关代替原来的一个两位开关；在充放电回路中分别串入了直流数字电流表，在电容两端并接了一个直流电压表，用于观察充放电过程中的电流及电压的变化；另外，放电回路中取消了放电电阻，充电电阻值和电容值也都有变化。

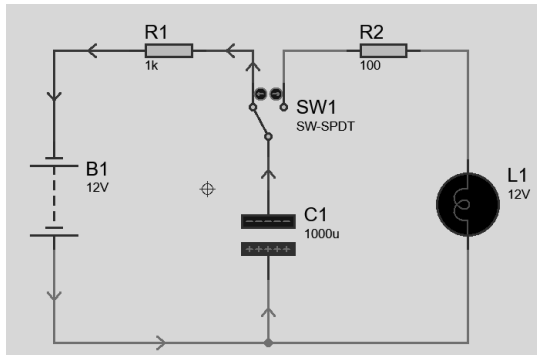


图 1-46 电容充电过程的仿真

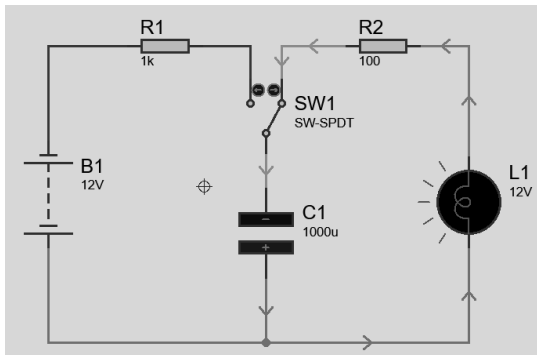


图 1-47 电容放电过程的仿真

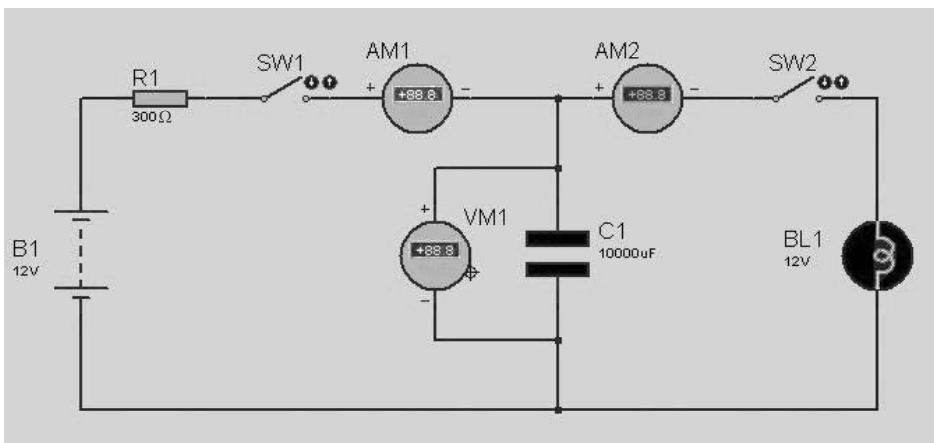


图 1-48 变式演练电路图

所用元件清单如表 1-2 所示。

表 1-2 变式演练电路的元件清单

元件名	所在类	子类	参数	备注
BATTERY	Miscellaneous		12V	电池组
SWITCH	Switches and Relays	Switches		一位开关
CAPACITOR	Capacitors	Animated	10000μF	电容
LAMP	Optoelectronics	Lamps	12V	灯
RES	Resistors	Generic	300Ω	电阻

先按表 1-2 所示拾取元件，并把元件放置到绘图区合适位置后连线。

选取虚拟仪器图标来获取直流电压表和电流表，如图 1-49 所示。

改变电表量程及名称。两个电流表设置为毫安表，分别取名为“AM1”和“AM2”；电压表取名为“VM1”。双击电流表，出现如图 1-50 所示的“Edit Component”(属性设置)对话框，照图完成设置。

运行仿真，拨动开关观察电容充放电效果、灯泡发光、电压表和电流表的变化趋势。反复操作，注意观察电流表和电压表数值的变化，回忆一阶动态电路性能。

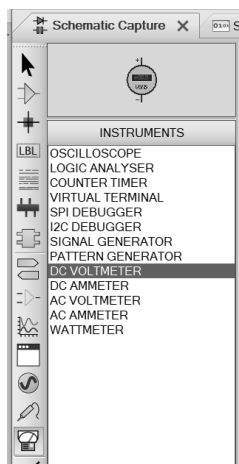


图 1-49 虚拟仪器菜单

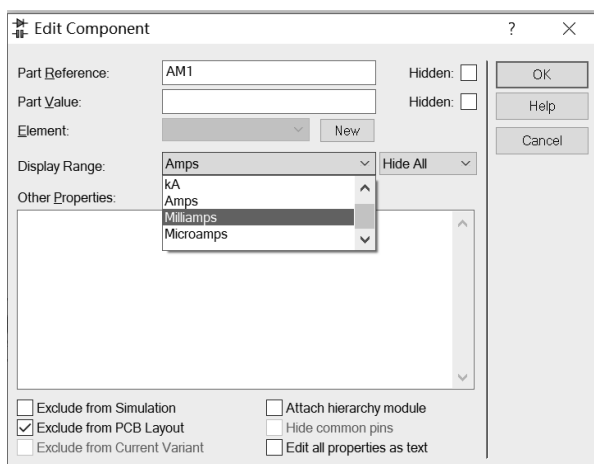


图 1-50 毫安表的设置

1.2.4 异步四位二进制计数器的设计及仿真

从上一节一阶动态电路的例子，我们已经对 Proteus 8 的基本功能有了初步的认识和了解。但是，还有一些疑问和不解没有解决。比如，怎么能知道 Proteus ISIS 中哪些元件属于哪些大类或子类、怎么能知道所用元件的名称等。通过下面这个数字电路的设计实例，会有进一步的体会。

如果已经具备了数字电子技术知识，就会知道，这是一个异步时序逻辑电路，四位二进制计数器或十六进制计数器，由四个触发器构成，这里选用 JK 触发器，把 JK 端全接高电平置 1，前一级的输出作为后一级的时钟信号。观察四个触发器输出端所组成的二进制数的变化是否为 0~F(即 0000~1111)。

在主界面工程创建向导中新建一个只有原理图设计的工程 Test2。

1. 元件的拾取

打开原理图设计工作视窗，先从元件拾取开始。如果不知道所用元件的确切名称及所在的类，可以用查询的方法在所有库里海选。选择主菜单中的【Library】→【Pick Parts From Libraries】菜单项，或直接单击左侧工具箱中的图标后再单击图标或按快捷键 P，打开如图 1-51 所示的对话框。

采用部分查找法，在所查找的元件名关键词中填写“JK”，所有 JK 触发器元件都被找出，显示在图 1-51 的中间部分查询结果中。选中“JKFF(ACTIVE)”，双击蓝色区域拾取元件到对象列表区。

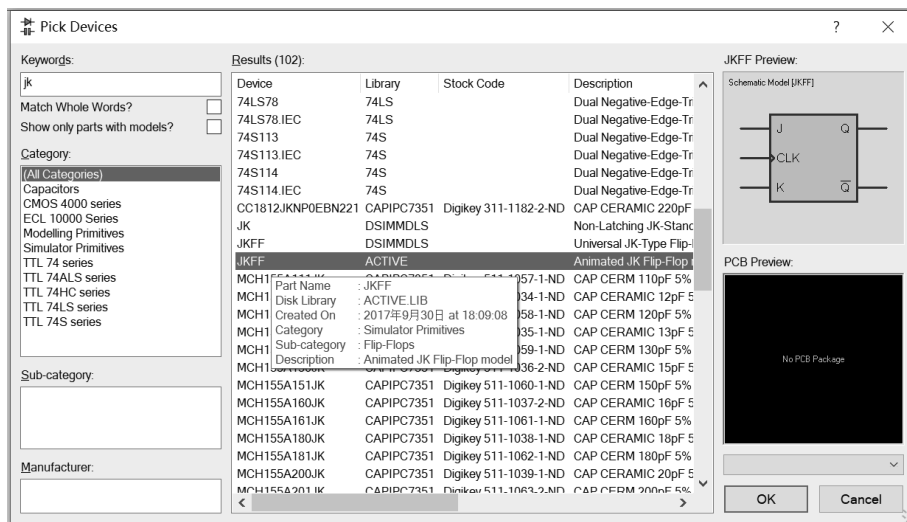


图 1-51 元件拾取对话框

采用不同的方法，依次按表 1-3 拾取所有元件后，单击“OK”按钮。

表 1-3 异步四位二进制计数器元件清单

元件名	所在类	子类	含义
JKFF	Modelling Primitives	Digital Sequential	触发器
CLOCK	Simulator Primitives	Sources	时钟
7SEG-BCD-GRN	Optoelectronics	7-segement Displays	含 BCD 译码驱动的七段数显
LOGICBPROBE[BIG]	Debugging Tools	Logic Probes	逻辑电平显示

2. 元件连线

在左上预览窗口中移动鼠标指针可平移绘图区视窗，在绘图区内滚动滚轮可以放大或缩小背景格点，放置元件图标和选择箭头图标或其他图标要来回切换。先放置一个 JK 触发器到绘图区。选中左侧电源和接地图标，单击“POWER”选项，一个箭头形状的标准数字直流电源(即高电平)出现在元件预览区，拖出后与触发器的 JK 端接上，如图 1-52 所示。

因为四个触发器的 JK 端接法都一样，故采取块复制法画其他三个触发器。用左键选中刚才所画的全部图形，选取上方的块复制图标，选择等间距在绘图区连续单击鼠标 3 次。如果想要调整复制块的位置，框选后选取上方的块移动图标重新布置即可，调整成如图 1-53 所示的图形。

接下来，把前一个触发器的输出端 $\bar{Q}$ 连接到下一个触发器的时钟 CLK 端上；再把时钟“CLOCK”从对象列表区拖出，连在第一个触发器的 CLK 输入端上。如果使用的是下降沿触发的触发器，则把前一个触发器的 Q 端连到下一个触发器的时钟 CLK 端上。

为了观察计数器计数的动态过程，在每个触发器的输出端 $\bar{Q}$ 上连接一个逻辑电平探测器，能够显示 0 和 1。把 LOGICPROBE(BIG)元件拖到图形编辑区内，连续双击鼠标 3 次，

得到四个逻辑探测器。分别接到每个 JK 触发器的输出端 $\bar{Q}$ 上。最右边一个触发器为二进制计数器的最高位(MSB)。

同时，把七段数码显示拖入图形编辑区。为了让 LED 连接更美观，可把背景栅格调小一档，选择主菜单中的【View】→【Snap 50th】菜单项，或直接按“F2”键。数码管的最左端是高位，分别和对应触发器的输出端 $\bar{Q}$ 相连，如图 1-54 所示。

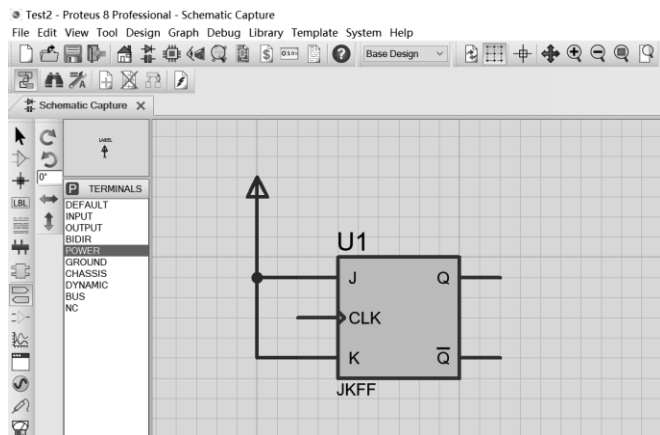


图 1-52 与触发器的 JK 端连接

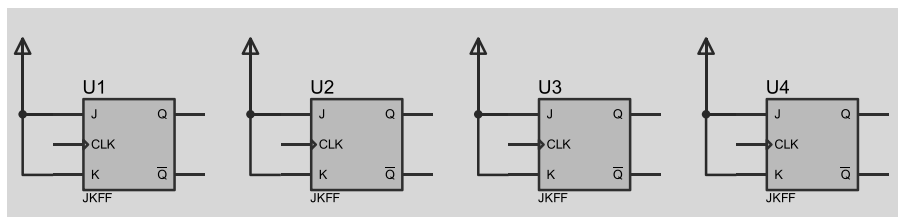


图 1-53 复制后的四个触发器

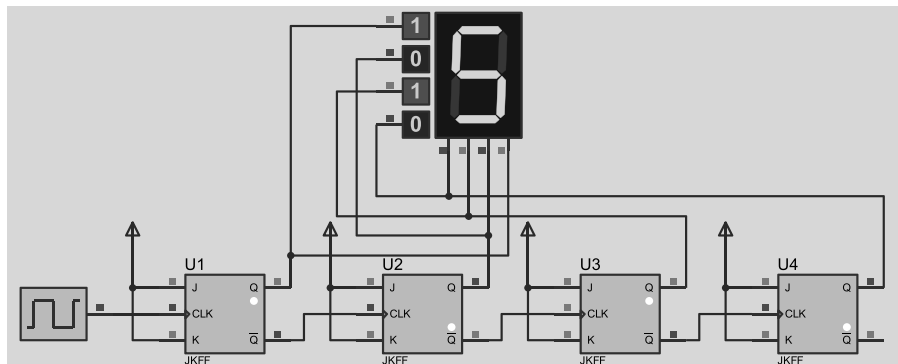


图 1-54 加上逻辑电平探测器和七段数码管后的电路

3. 电路的动态仿真

按下仿真运行按钮，在 1 秒时钟作用下，四个逻辑探测器组成的四位二进制数从 0000~1111 变化，而七段数码管则对应显示 0~F。另外，还观察到每个器件的连线端都有红蓝两

色小方块来显示该端的电平变化,红色为高电平,蓝色为低电平。虽然七段数显名称显示为BCD译码驱动,仿真可以看出十六进制数也可以显示。

4. 图表仿真

有时为了方便分析,还会在已绘制好的图中加上波形图,即图表仿真。在一个图表中,可以插入多个观测点的波形,既可以是数字波形,又可以是模拟波形或其他形式的波形。在绘图区自动生成仿真波形需要几个步骤:一是在图中放置探针,二是在图中空白区插入图表区,三是把探针名拖入图表区,四是进行波形图坐标及名称等参数修改,五是按下空格键进行图表仿真,即生成波形。

本例生成一个数字时序波形图,把上图中四个触发器输出端 $\bar{Q}$ 的波形显示出来。具体步骤如下。

(1) 把电压探针接在被测点。选择左侧工具栏的探针(Probe Mode)图标,在对象列表区显示有三种工具可选,前两个分别为电压(VOTAGE)和电流(CURRENT)探针;选中电压探针,在绘图区分别连接到四个触发器的输出 $\bar{Q}$ 端,把鼠标指针移动到要连接的线上,自动捕捉连线后单击即连接上(可通过右键菜单水平镜像),电压探针分别命名为Q0、Q1、Q2和Q3(双击电压探针可更名),如图1-55所示。电流探针是为了测电线上的电流值,用于模拟电路,圆圈内的箭头要求必须与导线平行,本例暂不用。

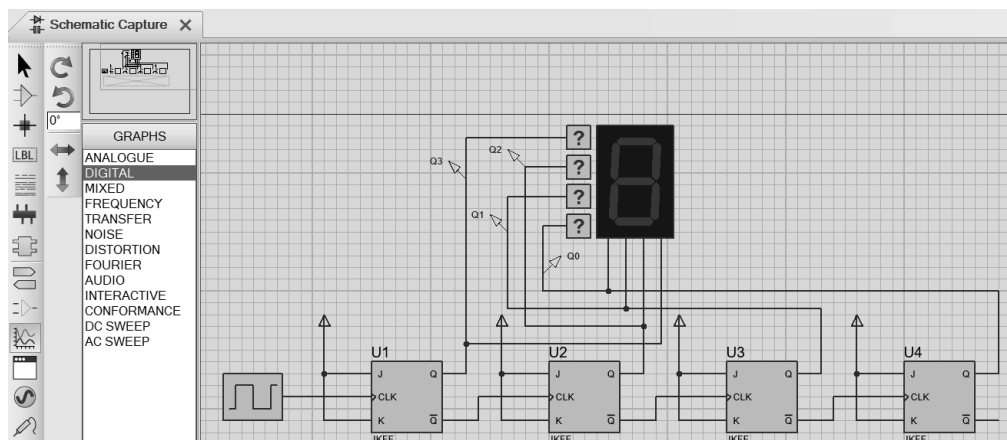


图 1-55 加上逻辑探针后的图形

(2) 单击左侧图标,选择图表仿真的种类。选择数字波形(DIGITAL)或数字模拟混合波形(MIXED)。在绘图区合适的空白位置单击鼠标左键拖出一个长方形图表区域,再次单击左键确定。在绘图区分别选中四个电压探针名字,再次单击把它拖曳到图表中,发现图表中多了四条白色的横线,此时可观察纵横向坐标。如果想删除图表中多余的电压探针,可选中探针名称,从右键菜单中选择删除。

(3) 双击图表中最上边的红色区域,把标题更改为“Timing Diagram”(时序图),把停止时间改为16(秒)。因为时钟CLOCK默认为1秒周期,所以计满一个循环(16个周期)共16秒,如图1-56所示。

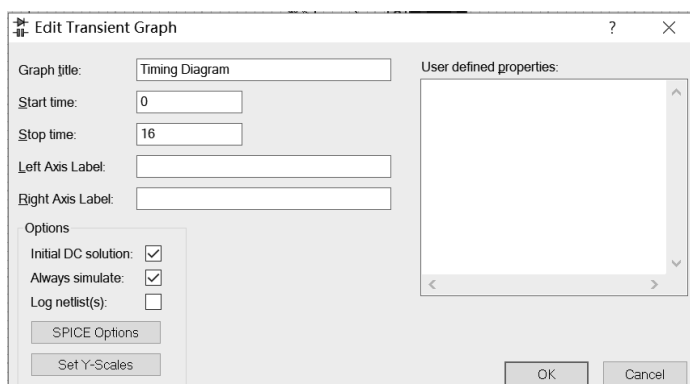


图 1-56 修改图表参数

(4) 按空格键，自动仿真图表，即计数器时序图如图 1-57 所示。

这种图表仿真生成的波形不同于示波器显示的波形，它能够静态地保留在原理图中，供读者分析或随图形一起输出。当修改电路时，按下空格键后，图表可以再次刷新生成。图表边框可以通过拖动进行调整。单击波形图标标题栏部分，可全屏显示波形，并可根据全屏显示的菜单更改波形及背景的颜色，拖动竖线，在左侧可查看各时刻、各观测点电平的高低等。用鼠标右键双击图表，则可将其删除。

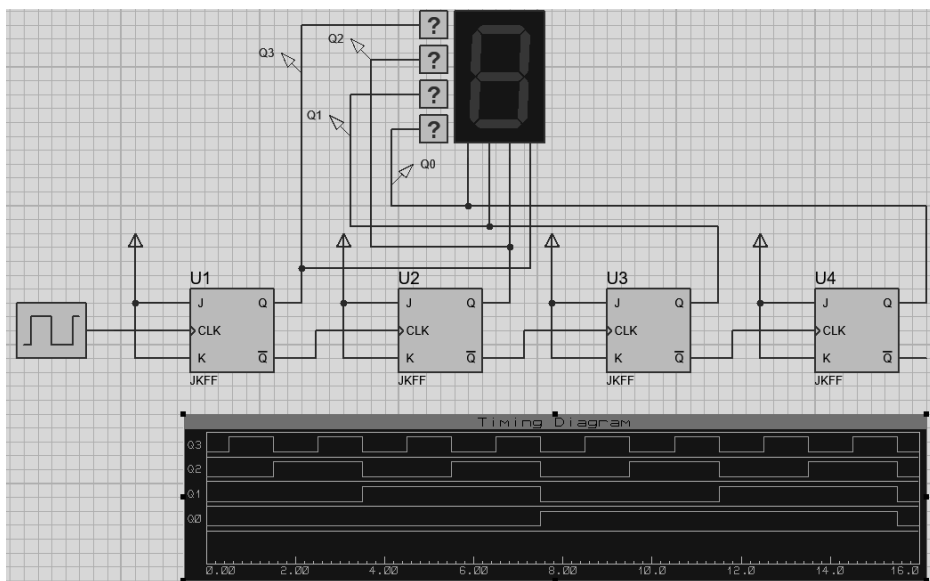


图 1-57 生成计数器时序图

5. 变式演练

设计一个 555 多谐振荡器，输出一个 10Hz 的方波。根据计算公式 $f = \frac{1}{T} = \frac{1.4}{(R_1 + 2R_2)C_1}$ ，取 C_1 为 473(0.047μF)，设 R_1 与 R_2 相等，则 $R_1=R_2=100\text{k}\Omega$ 。元件清单如表 1-4 所示。

表 1-4 多谐振荡器元件清单

元件名	所在类	子类	参数	备注
BATTERY	Miscellaneous		12V	电池组
NE555	Analog ICs	Timers		555 定时器
CAPACITOR	Capacitors	Animited	0.047μF	可显示电荷的电容
CAP	Capacitors	Generic	0.01μF	无极性电容
LOGICSPROBE(BIG)	Debugging Tools	Logic probes		逻辑电平探针

在电容 C_1 一端和 555 振荡器的输出端 3 分别放置电压探针，以生成这两点的波形。其中 V_C 选择为模拟波形(Analog)， V_O 为数字波形(Digital)。波形的停止时间不要选得太大，以免生成时间太长，因为周期为 10ms，停止时间选为 40ms 即可。

另外把这两点的波形接到示波器上观察。从虚拟仪器中拖出示波器，按图 1-58 所示接好。

单击交互仿真控制按钮中的运行按钮，自动弹出示波器界面。通过选择和调整相应按钮和旋钮，把电容的充放电波形图和输出方波显示出来，如图 1-59 所示。单击仿真停止按钮，示波器自动关闭，最好不要在示波器界面上关闭示波器，否则下次仿真运行时示波器将不会再出现，需要从主菜单的【Debug】→【Digital Oscilloscope】中调出。

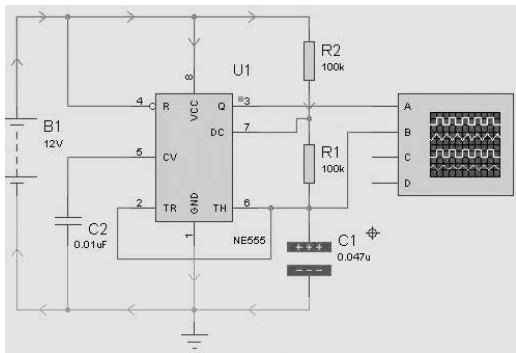


图 1-58 555 多谐振荡器

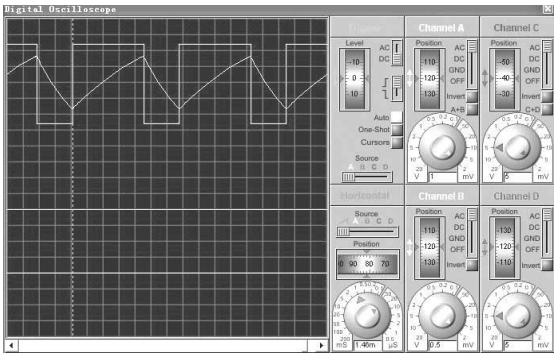


图 1-59 示波器界面

1.2.5 AT89C51 与 8255A 接口电路的调试及仿真

以上两个例子是基于电工电子技术的电路设计与仿真。其实，Proteus ISIS 的真正超群之处在于它对单片机电路的设计与仿真。接下来设计一个简单的 51 单片机接口扩展电路并仿真，表 1-5 是原理图绘制所需的元件。

表 1-5 AT89C51 与 8255A 接口电路元件清单

元件名	所在类	子类	备注
AT89C51	Microprocessor ICs	8051 Family	单片机
8255A	Microprocessor ICs		扩展 I/O
74LS373	TTL 74LS series	Flip-Flops & Latches	锁存器
LOGICSPROBE	Debugging Tools	Logic probes	逻辑电平探针

先在 Proteus 8 工程创建向导中新建一个 Test3 的工程项目，选择原理图、固件项目，固件选项选择 AT89C51 单片机和默认的编译器。在打开的集成工作视窗下，先把当前工作视窗转换为原理图设计视窗，按表 1-5 所示拾取元件，按图 1-60 所示绘图。

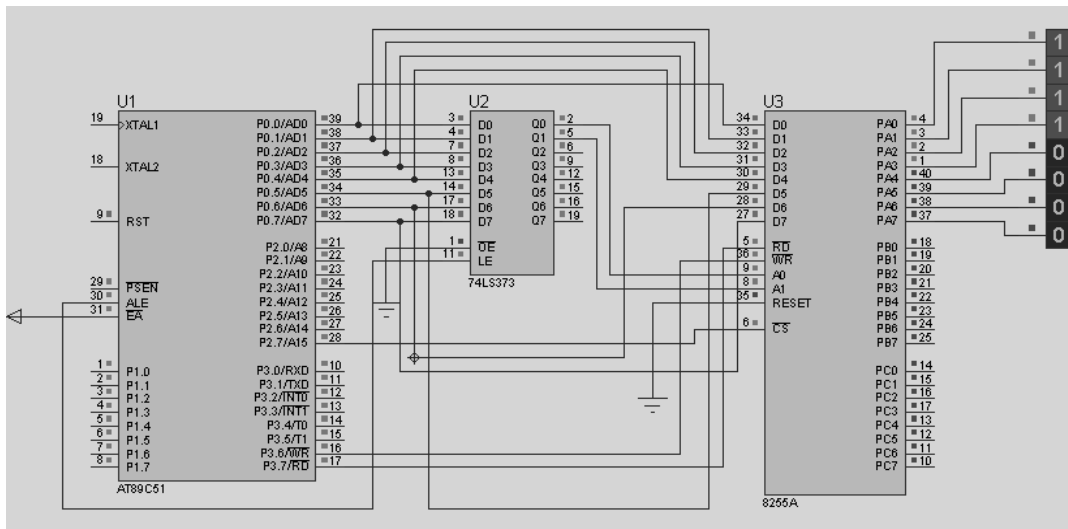


图 1-60 仿真运行中的电路

原理图绘制完成且无误后，存盘，并切换当前工作视窗为程序代码视窗。在程序编辑区的“Start”下面输入或从其他文本复制过来用户程序，对已建程序体结构不做任何修改，当然也可以把空的循环体及主程序中的跳转语句删除。

```
MOV    DPTR,#7FFFH    ;设 8255A 的控制地址为 7FFFH
MOV    A,#80H          ;将控制字 80H 写到 8255A 的 7FFFCH 地址
MOVX   @DPTR,A
MOV    DPTR,#7FFCH
MOV    A,#0FH          ;把十六进制数 0FH 送到 8255A 的 PA 口
MOVX   @DPTR,A
```

程序输入完成后，选择主菜单中的【Build】→【Build Project】菜单项或按“Ctrl+F7”键，或单击上方工具图标进行项目程序编译，如果一切正确，在左下方会显示成功编译，结果如图 1-61 所示。

可以看到，程序设计视窗非常便捷化，程序结构已经建好，只需填空即可。

切换至原理图设计工作视窗，单击仿真运行按钮，出现图 1-60 所示的仿真结果。编译后的程序自动嵌入单片机中，无须手动导入。从仿真结果看，8255A 的 PA 口输出的逻辑电平显示与程序写入的十六进制字一致。

以上 3 个例子分别为电路分析、电子技术和单片机技术中的三个电路，可以照顾到初学者的不同知识层次。这里，只介绍了 Proteus 8 的部分功能，即电路设计与仿真，因为仿真效果的展示可以提高初学者的学习兴趣，使初学者对软件有一个总体的把握。

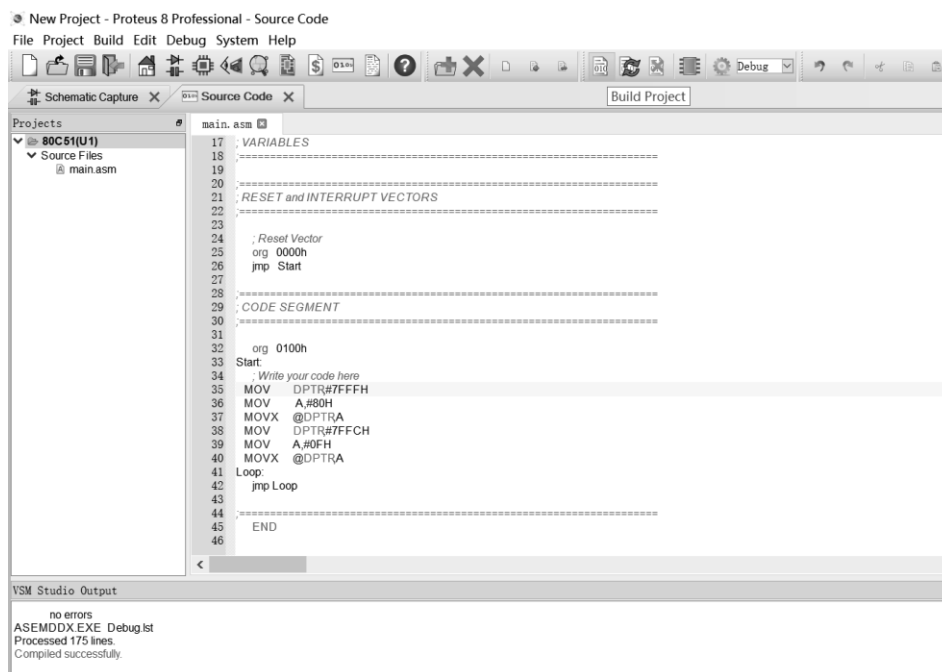


图 1-61 程序编写及编译