



基于 Proteus 的 51系列单片机 设计与仿真

///

(第3版)

陈忠平 编著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



基于

Proteus 的

51 系列 单片机

设计与仿真



(第3版)

陈忠平 © 编著

电子工业出版社

Publishing House of Electronics Industry

内 容 简 介

本书以目前流行的软硬件仿真软件 Proteus 为核心,采用现代教学方法,从实验、实践、实用的角度出发,通过丰富的实例详细讲述了 Proteus 软件在 51 单片机课程教学和单片机应用产品开发过程中的应用。

本书以夯实基础,面向应用,理论与实践紧密结合为原则,采用汇编及 C 语言作为系统软件开发平台。全书共 9 章,主要包括 80C51 单片机系统设计相关软件的使用、Proteus8.0 入门、51 系列单片机软件程序设计、51 系列单片机通用 I/O 端口控制、LED 数码管与键盘的应用、D/A 转换器和 A/D 转换器的应用、显示器的应用、电动机控制和综合应用设计。

本书适合从事单片机应用研发的工程技术人员自学使用,也可作为高等学校相关专业单片机课程的教学用书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

基于 Proteus 的 51 系列单片机设计与仿真/陈忠平编著. —3 版. —北京:电子工业出版社,2015.9
ISBN 978-7-121-27041-3

I. ①基… II. ①陈… III. ①单片微型计算机-系统设计-应用软件②单片微型计算机-系统仿真-应用软件 IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2015)第 203233 号

责任编辑:张 剑(zhang@phei.com.cn)

印 刷:北京京科印刷有限公司

装 订:北京京科印刷有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:25 字数:640 千字

版 次:2008 年 9 月第 1 版

2015 年 9 月第 3 版

印 次:2015 年 9 月第 1 次印刷

印 数:3 000 册 定价:59.90 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

第3版前言

单片机又称为单片微处理器，其实质是将一个计算机系统集成到一个芯片上。它具有体积小、质量轻、价格便宜、控制功能强等特点，在工业控制、智能仪表、家用电器及军事装置等方面都得到了广泛的应用。

本书第1版于2008年9月出版，现已被许多学校或培训机构作为单片机课程的实践教材，得到众多教师、学生和读者的认可，在此我们表示衷心的感谢。

鉴于单片机及嵌入式系统技术发展迅速，决定对本书再次进行修订。本次修订在保留第2版体例结构及编写特色的基础上，主要进行了以下修改：

- ☺ 第1章删除了Keil软件窗口介绍部分的内容。
- ☺ 第2章以Proteus8.0版本为例，讲述Proteus ISIS的使用方法，并删除Proteus ARES的PCB设计部分内容。
- ☺ 第4章的内容重新整合为声光控制、定时/计数器控制、外部中断控制、串行通信控制。
- ☺ 第5章删除了串行口驱动一位共阴极LED数码管显示、矩阵式键盘的识别（二）的内容，增加了LED数码管的识别与检测方法。
- ☺ 第6章将原6.3节和原6.4节的内容整合为A/D转换器的应用。
- ☺ 第7章增加了字模PCtoLCD2002的使用，并针对每个点阵显示程序讲解了字体的取模方式。
- ☺ 第8章删除了步进电动机正转控制部分的内容，将其知识点在步进电动机正、反转控制中进行体现。
- ☺ 第9章增加了矩阵键盘键值显示的内容，进一步加强读者的综合应用设计能力。

本书由湖南工程职业技术学院陈忠平编著。参加本书编写的还有湖南工程职业技术学院徐刚强、龚亮、陈建忠、龙晓庆，湖南航天诚远精密机械有限公司刘琼，湖南涉外经济学院侯玉宝、高金定，湖南科技职业技术学院高见芳，湖南三一重工集团王汉其等。全书由湖南工程职业技术学院李锐敏教授主审。

由于作者水平有限，书中难免有疏漏之处，恳请读者予以指正或提出修改意见。

编著者

目 录

第 1 章 80C51 单片机系统设计相关软件的使用	1
1.1 Keil C51 的使用	1
1.2 仿真器	9
1.3 编程器	10
1.4 ISP 下载	12
1.5 串行调试软件	14
第 2 章 Proteus 8.0 快速入门	15
2.1 Proteus ISIS 的操作及电路原理图设计	15
2.1.1 Proteus ISIS 编辑环境及参数设置	15
2.1.2 Proteus ISIS 原理图设计	23
2.2 Proteus VSM 虚拟系统模型	32
2.2.1 激励源	32
2.2.2 Proteus VSM 虚拟仪器的使用	33
2.2.3 Keil C51 与 Proteus 的联机调试	44
第 3 章 51 系列单片机程序设计	47
3.1 清零与置数程序的设计	47
3.1.1 片内清零程序的设计	47
3.1.2 置数程序的设计	50
3.2 拼字与拆字程序的设计	52
3.2.1 片内拼字程序的设计	52
3.2.2 片内拆字程序的设计	55
3.3 数据块传送与排序程序的设计	57
3.3.1 数据块传送程序的设计	57
3.3.2 数据排序程序的设计	60
第 4 章 51 系列单片机通用 I/O 端口控制	65
4.1 声光控制	65
4.1.1 闪烁灯控制	65
4.1.2 声光报警控制	70
4.1.3 流水灯控制	74
4.1.4 花样灯控制	77
4.1.5 模拟交通灯控制	81
4.2 定时器/计数器控制	84
4.2.1 T/C 延时控制	84
4.2.2 T/C 计数控制	87

4.3	外部中断控制	90
4.3.1	单个外部中断控制	90
4.3.2	外部中断优先控制	94
4.4	串行通信控制	100
4.4.1	两个单片机之间的串行通信控制	100
4.4.2	单片机与 PC 之间的串行通信控制	105
第 5 章	LED 数码管与键盘的应用	112
5.1	LED 数码管应用	112
5.1.1	LED 数码管的显示原理	112
5.1.2	一位共阳极 LED 数码管的静态显示	115
5.1.3	0 ~ 99 计数器的设计	118
5.1.4	59s 计时器的设计	123
5.1.5	8 位 LED 数码管移位显示	127
5.1.6	MAX7219 串行驱动 LED 数码管	133
5.2	键盘的应用	145
5.2.1	键盘工作原理	146
5.2.2	查询式键盘设计	147
5.2.3	矩阵式键盘的识别	151
第 6 章	D/A 转换器和 A/D 转换器的应用	160
6.1	DAC0832 转换器的应用	160
6.1.1	DAC0832 输出方波	161
6.1.2	DAC0832 输出三角波	164
6.1.3	DAC0832 输出正弦波	167
6.2	TLC5615 转换器的应用	170
6.2.1	TLC5615 输出锯齿波	172
6.2.2	TLC5615 输出阶梯波	177
6.3	A/D 转换器的应用	181
6.3.1	ADC0808 数字电压表的设计	182
6.3.2	ADC0832 数字电压表的设计	190
第 7 章	显示器的应用	200
7.1	点阵 LED 的应用	200
7.1.1	一个 5 × 7 点阵字符显示	202
7.1.2	一个 8 × 8 点阵字符显示	205
7.1.3	两个 8 × 8 点阵字符显示	209
7.1.4	16 × 16 点阵汉字显示	216
7.2	液晶显示器的应用	223
7.2.1	字符式 LCD 字符串显示	229
7.2.2	字符式 LCD 静态显示	235
7.2.3	汉字式 LCD 静态显示	241
7.2.4	汉字式 LCD 滚动显示	262

第 8 章 电动机控制	276
8.1 步进电动机控制	276
8.1.1 步进电动机的起动、停止控制	277
8.1.2 步进电动机正、反转控制	281
8.1.3 步进电动机转速控制	285
8.2 直流电动机控制	288
8.2.1 直流电动机的起动、停止控制	290
8.2.2 直流电动机正、反转控制	294
8.2.3 直流电动机转速控制	298
8.2.4 直流电动机多地控制	301
第 9 章 综合应用设计	305
9.1 数字电子钟的设计	305
9.2 篮球比赛计分器的设计	310
9.3 DS1302 可调时钟的设计	319
9.4 24C04 开启次数统计	343
9.5 DS18B20 测量温度	355
9.6 按键选播电子音乐	365
9.7 矩阵键盘键值显示	373
附录 A 汇编指令速查表	380
附录 B C51 库函数	385
附录 C Proteus 常用快捷键	391
参考文献	392

第 1 章 80C51 单片机系统设计

相关软件的使用


单片机应用系统是以单片机为核心，同时配以相应的外围电路及软件来完成某些功能的系统。它包括硬件和软件两部分，硬件是系统的“躯体”，软件是系统的“灵魂”。本章主要讲述相关软件的使用。



1.1 Keil C51 的使用

单片机的源程序是在哪里进行编写的？是在哪里将其调试并生成 .HEX 文件的？其实这些工作在单片机的一些编译软件中就可以完成。单片机程序的编译调试软件比较多，如 51 汇编集成开发环境、伟福仿真软件、Keil 单片机开发系统等。

Keil C51 是当前使用最广泛的基于 80C51 单片机内核的软件开发平台之一，由德国 Keil Software 公司推出。 μ Vision4 是 Keil Software 公司推出的关于 51 系列单片机的开发工具。 μ Vision4 集成开发环境 IDE 是一个基于 Windows 的软件开发平台，集编辑、编译、仿真于一体，支持汇编语言和 C 语言的程序设计。一般来说，Keil C51 和 μ Vision4 指的是 μ Vision4 集成开发环境。

Keil C51 可以从相关网站下载并安装。安装好后，双击桌面上的快捷图标，或者在“开始”菜单中选择“Keil μ Vision4”，即可启动 Keil μ Vision4 集成开发环境，如图 1-1 所示。

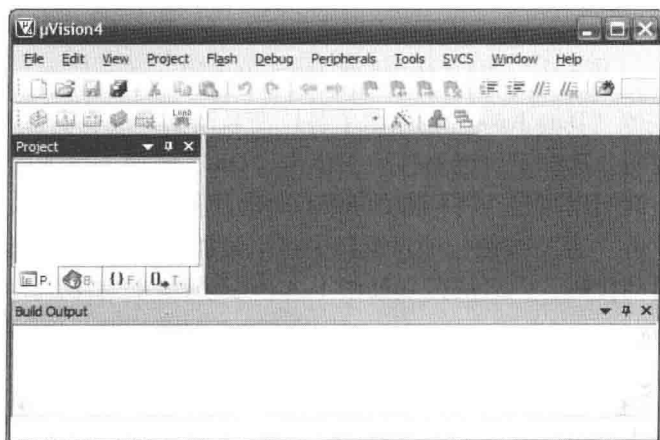



图 1-1 启动 Keil μ Vision4 后的界面

1. 创建项目

Keil μ Vision4 中有一个项目管理器，它包含了程序的环境变量和与编辑有关的全部信息，为单片机程序的管理带来了很大的方便。

【创建新项目的操作步骤】

- (1) 启动 μ Vision4, 创建一个项目文件, 并从元器件数据库中选择一款合适的 CPU。
- (2) 创建一个新的源程序文件, 并把这个源程序文件添加到项目中。
- (3) 设置工具选项, 使之适合目标硬件。
- (4) 编译项目, 并生成一个可供 PROM 编程的 .HEX 文件。

1) 启动 μ Vision4 并创建一个项目文件 μ Vision4 是一个标准的 Windows 应用程序, 直接在桌面上双击图标  就可启动它。在 μ Vision4 中执行菜单命令 “Project” → “New Project”, 弹出 “Create New Project” 对话框, 在此可以输入项目名称 (建议每个项目使用一个独立的文件夹)。输入新建项目名后, 单击 “确定” 按钮, 弹出如图 1-2 所示的 “Select Device for Target 'Target 1'” 对话框。在此对话框中, 根据需要选择合适的单片机型号。执行菜单命令 “Project” → “Select Device for Target” 也会弹出图 1-2 所示的对话框。

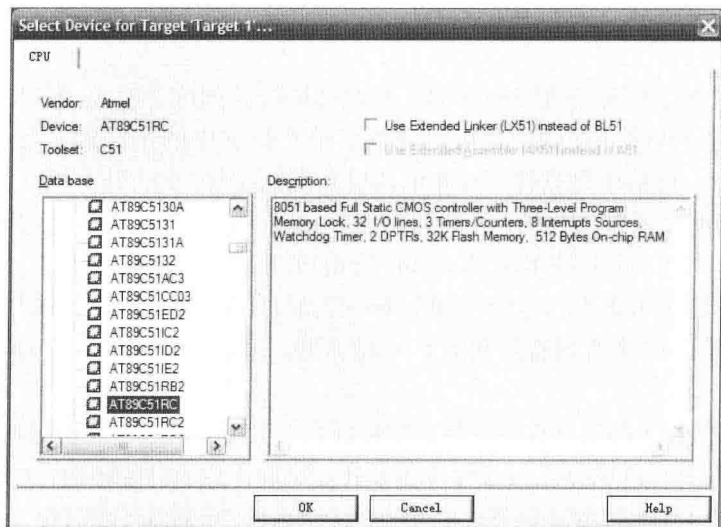


图 1-2 “Select Device for Target 'Target 1'” 对话框

在图 1-2 中, 左侧 “Data base” 栏列出了各厂商名及其产品, 右侧 “Description” 栏是对该选中单片机的说明。选择了目标器件后, 单击 “OK” 按钮, 将弹出图 1-3 所示的对话框。在此对话框中, 询问用户是否将标准的 8051 启动代码复制到项目文件夹并将该文件添加到项目中。如果单击 “否” 按钮, 项目窗口中将不添加启动代码; 如果单击 “是” 按钮, 项目窗口中将添加启动代码。二者的区别如图 1-4 所示。

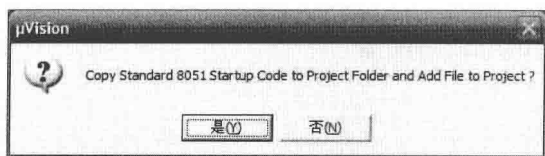
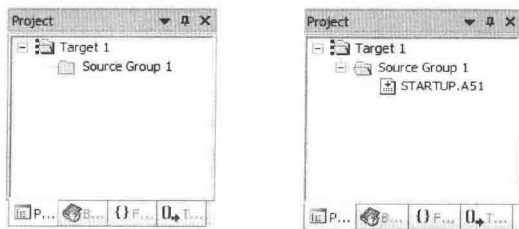


图 1-3 询问是否添加启动代码对话框




(a) 未添加启动代码

(b) 添加启动代码

图 1-4 是否添加启动代码的区别

startup. a51 文件是大部分 8051CPU 及其派生产品的启动程序, 启动程序的操作包括清除数据存储内容、初始化硬件及可重入堆栈指针。一些 8051 派生的 CPU 需要初始化代码以使配置符合硬件上的设计。例如, Philips 的 8051RD + 片内 xdata RAM 需通过在启动程序中的设置才能使用。应按照目标硬件的要求来创建相应的 startup. a51 文件, 或者直接将它从安装路径的 \C51\LIB 文件夹中复制到项目文件中, 并根据需要进行更改。

2) 创建新的源程序文件 单击图标或执行菜单命令“File”→“NEW”, 即可创建一个源程序文件。该命令会打开一个空的编辑器窗口, 在编辑窗口中输入源代码, 如图 1-5 所示。源代码可以用汇编语言或单片机 C 语言进行书写, 源代码输入完成后, 执行菜单命令“File”→“Save as...”或“Save”, 即可对源程序进行保存。在保存时, 文件名只能由字符、字母或数字组成, 并且一定要带扩展名(使用汇编语言编写的源程序的扩展名为 .A51 或 .ASM; 使用单片机 C 语言编写的源程序的扩展名为 .C)。源程序保存好后, 源程序窗口中的关键字呈彩色高亮度显示。

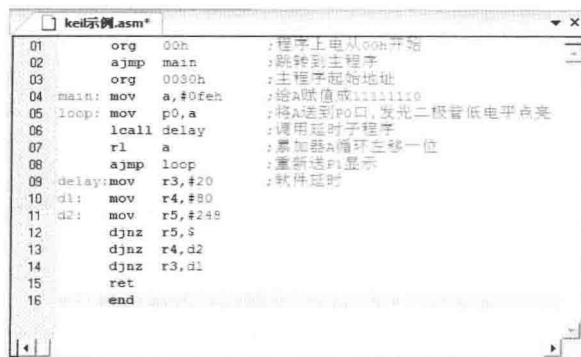


图 1-5 源程序编辑窗口

源程序文件创建好后, 可以把这个文件添加到项目中。在 μ Vision4 中, 添加的方法有多种。如图 1-6 所示, 在“Source Group 1”上单击鼠标右键, 在弹出的菜单中选择“Add Files to Group 'Source Group 1'”, 在弹出的“Add Files to Group 'Source Group 1'”对话框中选择刚才创建的源程序文件即可将其添加到项目中。

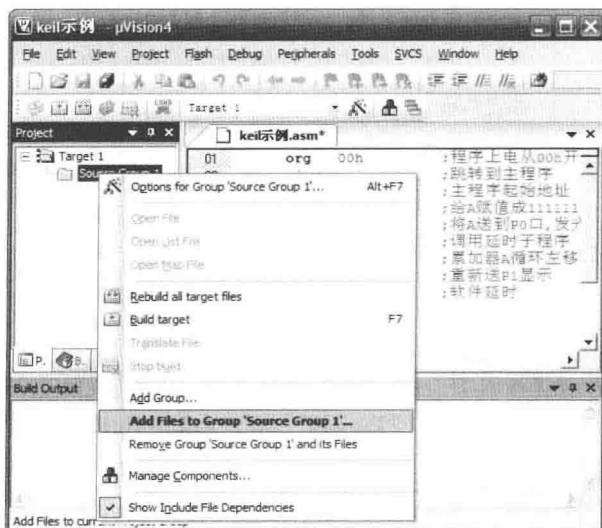



图 1-6 在项目中添加源程序文件

3) 为目标设定工具选项 单击图标  或执行菜单命令“Project”→“Options for Target”，将会出现“Options for Target 'Target 1'”对话框，如图 1-7 所示。在“Target”选项卡中可以对目标硬件及所选器件片内部件进行参数设定。表 1-1 描述了“Target”选项卡的选项说明。

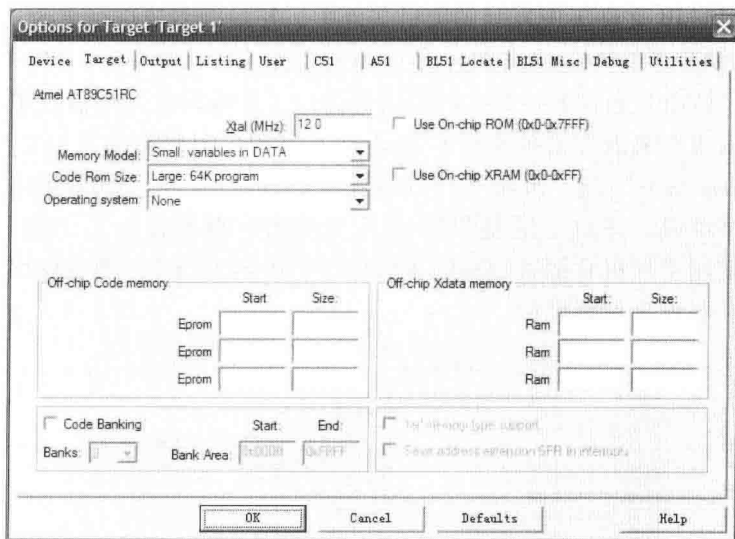


图 1-7 “Options for Target 'Target 1'”对话框

表 1-1 “Target”选项卡的选项说明

选 项	说 明
Xtal	指定器件的 CPU 时钟频率，多数情况下，其值与 XTAL 的频率相同
Use On - chip ROM	使用片上自带的 ROM 作为程序存储器
Memory Model	指定 C51 编译器的存储模式，在开始编辑新应用时，默认为 Small
Code Rom Size	指定 ROM 存储器的大小
Operating system	操作系统的选择
Off - chip Code memory	指定目标硬件上所有外部地址存储器的地址范围
Off - chip Xdata memory	指定目标硬件上所有外部数据存储器的地址范围
Code Banking	指定 Code Banking 块数

标准的 80C51 的程序存储器空间为 64KB，若程序存储器空间超过 64KB 时，可在“Target”选项卡中对“Code Banking”栏进行设置。Code Banking 为地址复用，可以扩展现有的 CPU 程序存储器寻址空间。选中“Code Banking”栏后，用户根据需求在“Banks”栏中选择合适的块数。在 Keil C51 中，用户最多可以使用 32 块 64KB 的程序存储空间，即 2MB 的空间。

4) 编译项目并创建 HEX 文件 在“Target”选项卡中设置好参数后，即可对源程序进行编译。单击图标  或执行菜单命令“Project”→“Build Target”，可以编译源程序并生成应用程序。当所编译的程序有语法错误时， μ Vision4 将会在“Build Output”窗口中显示错误和警告信息，如图 1-8 所示。双击某一条信息，光标将会停留在 μ Vision4 文本编辑窗口中出现该错误或警告的源程序位置上。

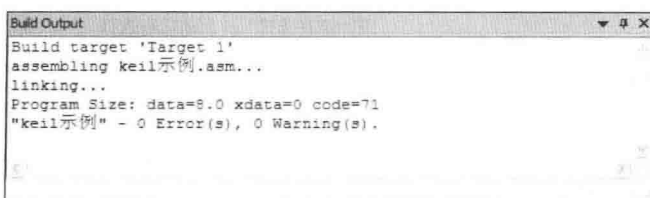


图 1-8 错误和警告信息

若成功创建并编译了应用程序，就可以开始调试。当程序调试好后，要求创建一个 HEX 文件，生成的 .HEX 文件可以下载到 EPROM 编程器或模拟器中。

若要创建 HEX 文件，必须将“Options for Target 'Target 1'”对话框的“Output”选项卡中的“Create HEX File”复选框选中，如图 1-9 所示。

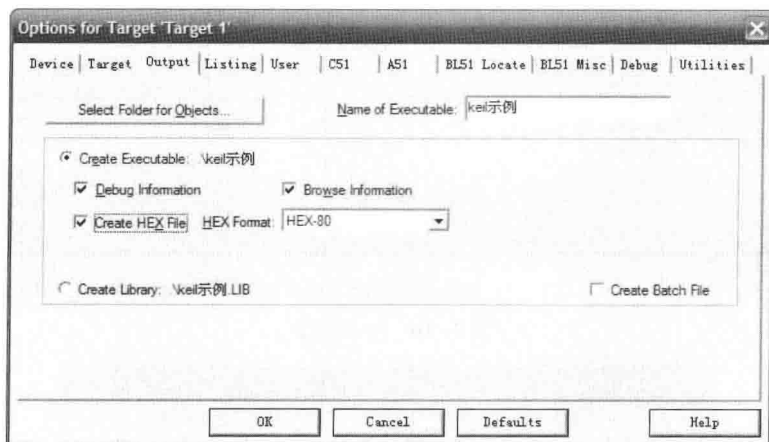


图 1-9 选中“Create HEX File”


2. 仿真设置

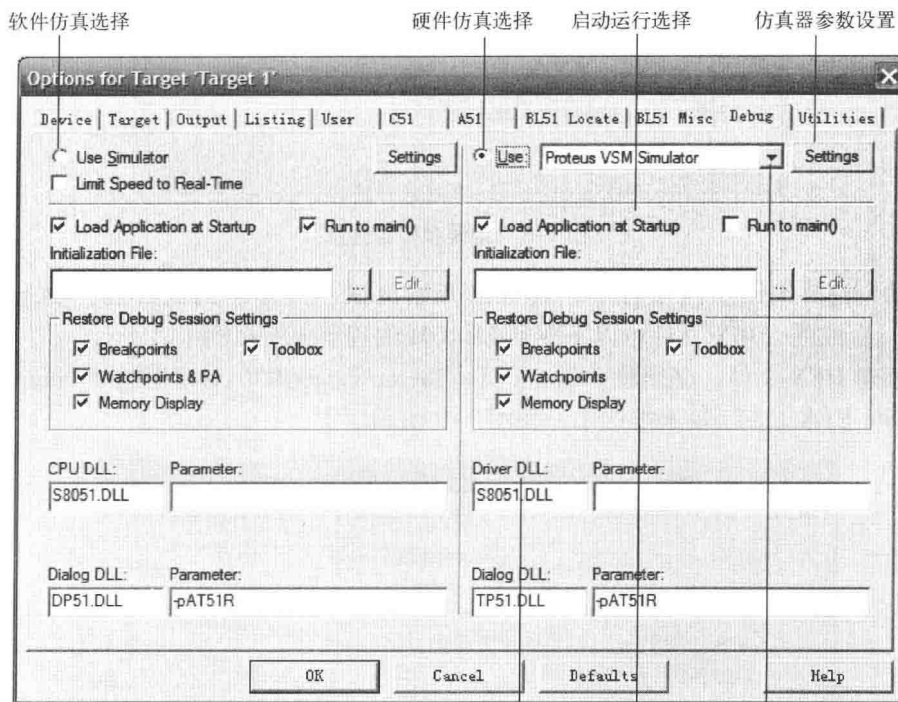
使用 μ Vision4 调试器可对源程序进行测试， μ Vision4 提供了两种工作模式，这两种模式可以在“Option for Target 'Target 1'”对话框的“Debug”选项卡中选择，如图 1-10 所示。

☺ Use Simulator: 软件仿真模式，将 μ Vision4 调试器配置成纯软件产品，能够仿真 8051 系列产品的绝大多数功能而不需要任何硬件目标板，如串行口、外部 I/O 和定时器等，这些外围部件设置是在从元器件数据库选择 CPU 时选定的。

☺ Use: 硬件仿真，如 TKS Debugger，用户可以直接把这个环境与仿真程序或 Keil 监控程序相连。

(1) CPU 仿真： μ Vision4 仿真器可以模拟 16MB 的存储器，该存储器被映射为读、写或代码执行访问区域。除将存储器映射外，仿真器还支持各种 80C51 派生产品的集成外围器件。在“Debug”选项卡中可以选择和显示片内外围器件，也可通过设置其内容来改变各种外设的值。

(2) 启动调试：源程序编译好后，选择相应的仿真操作模式，可启动源程序的调试。单击图标或执行菜单命令“Debug”→“Star/Stop Debug Session”，即可启动 μ Vision4 的调试模式，如图 1-11 所示。



仿真目标器件驱动选择 仿真配置记忆选择 仿真器类型选择

图 1-10 仿真设置

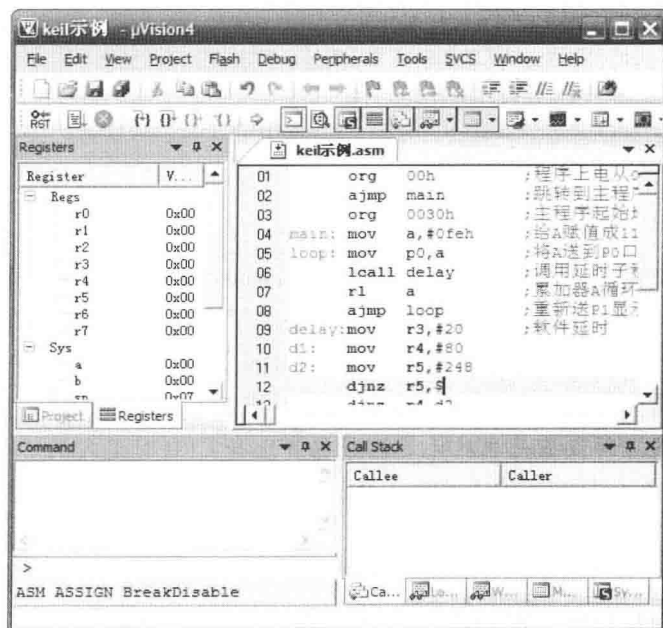




图 1-11 调试界面



(3) 断点的设定：在编辑源程序过程中，或者在程序尚未编译前，可以设置执行断点。 μ Vision4 中可用不同的方法来定义断点。

☺ 在文本编辑窗口或反汇编窗口中选定所在行，然后单击“File Toolbar”按钮或图标.

- ⑤ 在文本编辑窗口或反汇编窗口本地菜单上单击鼠标右键，打开快捷菜单，进行断点设置。
 - ⑥ 执行菜单命令“Debug”→“Breakpoint”，打开“Breakpoint”对话框，在此对话框中可以查看、定义或更改断点的设置。
 - ⑦ 在“Command”窗口中可以使用 BreakSet、BreakKill、BreakList、BreakEnable 和 BreakDisable 等命令。
- (4) 目标程序的执行：目标程序的执行可使用以下方法进行操作。
- ⑧ 执行菜单命令“Debug”→“Run”，或者直接单击图标。
 - ⑨ 在文本编辑窗口或反汇编窗口中单击鼠标右键，在弹出的快捷菜单上选择“Run till Cursor line”命令。
 - ⑩ 在“Command”窗口中可以使用 Go、Ostep、Pstep、Tsetp 命令。

3. Keil 程序调试与分析

前面讲述了如何在 Keil 中建立、编译、链接项目，并获得目标代码，但是做到这一步仅代表源程序没有语法错误，至于源程序中存在的其他错误，必须通过调试才能发现并解决。事实上，除极简单的程序外，绝大多数的程序都要通过反复调试才能得到正确的结果。因此，调试是软件开发中的一个重要环节。

1) 寄存器和存储器窗口分析 进入调试状态后，执行菜单命令“Debug”→“Run”，或者单击图标，全部运行源程序；执行菜单命令“Debug”→“Step”，或者单击图标，单步运行源程序。在源程序运行过程中，项目工作区（Project Workspace）“Registers”选项卡中显示相关寄存器当前的内容。若在调试状态下未显示此窗口，可执行菜单命令“View”→“Project Window”将其打开。

在源程序运行过程中，可以通过存储器窗口（Memory Window）来查看存储区中的数据。在存储器窗口中有供用户输入存储器类型起始地址的文本输入栏，用于设置关注对象所在的存储区域和起始地址，如“D:0x30”。其中，前缀表示存储区域，冒号后为要观察的存储单元的起始地址。常用的存储区前缀有“d”或“D”（表示内部 RAM 的直接寻址区）、“i”或“I”（表示内部 RAM 的间接寻址区）、“x”或“X”（表示外部 RAM 区）、“c”或“C”（表示 ROM 区）。由于 P0 端口属于特殊功能寄存器（SFR），片内 RAM 字节地址为 80H，所以在存储器窗口的上部输入“d:80h”时，可查看 P0 端口的当前运行状态为 FE，如图 1-12 所示。

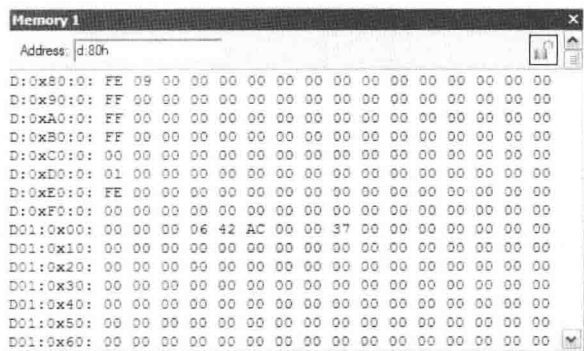







图 1-12 存储器窗口

2) 延时子程序的调试与分析 在源程序编辑状态下，执行菜单命令“Project”→“Options for Target 'Target 1'”，或者在工具栏中单击图标，再在弹出的对话框中选择“Target”选项卡。在“Target”选项卡的“Xtal (MHz):”栏中输入 12，即设置单片机的晶振频率为 12MHz。然后在工具栏中单击图标，对源程序再次进行编译。

执行菜单命令“Debug”→“Start/Stop Debug Session”，或者在工具栏中单击图标，

进入调试状态。在调试状态下,单击图标,使光标首次指向 LCALL DELAY 后,项目工作区“Registers”选项卡的 Sys 项中 sec 为 0.00000400,表示进入首次运行到 LCALL DELAY 时花费了 0.00000400s,如图 1-13 所示。再次单击图标,光标指向“RL A”,Sys 项的 sec 为 0.79846900,如图 1-14 所示。因此,DELAY 的延时时间为二者之差,即 0.79846500s,也就是说延时约为 0.8s。

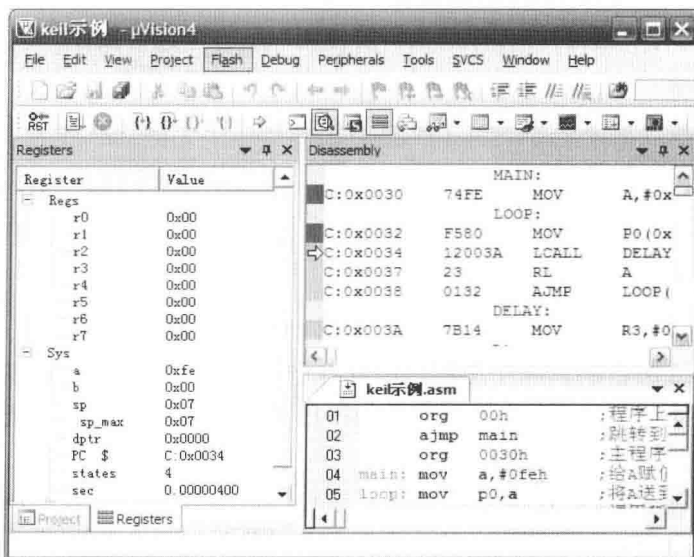


图 1-13 光标首次指向 LCALL DELAY

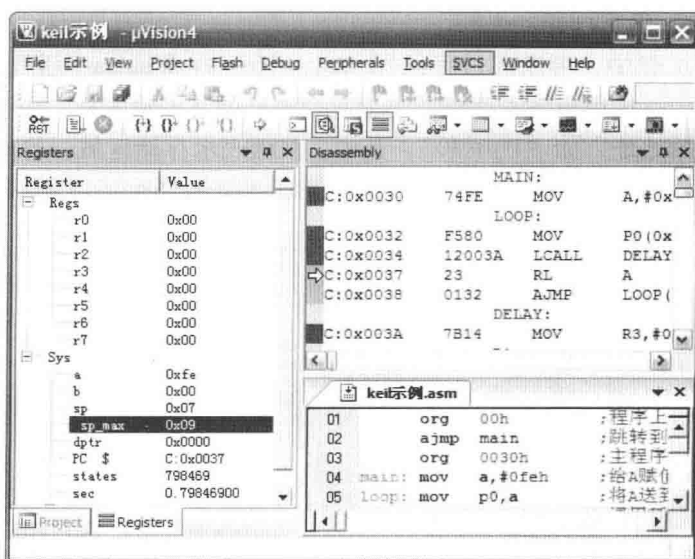





图 1-14 光标首次指向“RL A”

3) P0 端口运行模拟分析 执行菜单命令“Debug”→“Start/Stop Debug Session”,或者在工具栏中单击图标,进入调试状态。

执行菜单命令“Peripherals”→“L/O Ports”→“Port 0”,将弹出“Parallel Port 0”窗

口。“Parallel Port 0”窗口的最初状态如图 1-15 (a) 所示,表示 P0 端口的初始值为 0xFF,即 FFH。单击图标或多次单击图标后,“Parallel Port 0”窗口的状态将会发生变化,如图 1-15 (b) 所示,表示 P0 端口当前为 0xFB,即 FBH。

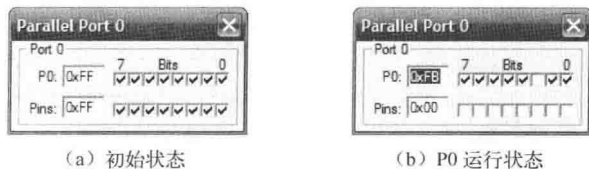


图 1-15 P0 端口状态



1.2 仿真器

编写好源程序后,进行程序的调试时,需对其进行仿真。仿真就是使用可控的手段来模仿佛真实的情况。在单片机嵌入式系统设计中,仿真主要是指对程序的仿真。仿真分为两类,即软件仿真和硬件仿真。

软件仿真就是使用计算机软件来模拟单片机的实际运行情况,而用户不需要搭建硬件电路即可对程序进行验证,但是软件仿真无法完全仿真与硬件相关的部分。硬件仿真就是使用附加的硬件来替代用户系统的单片机,并完成单片机全部或大部分的功能,它能直接反映单片机的全部或部分实际运行控制功能,在开发过程中必须进行硬件仿真。

能进行硬件仿真的设备称为仿真器,常用的仿真器有南京伟福公司生产的伟福仿真器和广州周立功公司生产的 TKS 系列仿真器。TKS 系列仿真器在硬件上采用了 HOOKS/Bond-out 仿真技术,可以实时在线仿真 Philips 公司生产的 80C51 系列单片机及 Atmel、Winbond、HY 等公司生产的兼容 MCS-51 内核的标准 80C51 单片机。TKS 系列仿真器如图 1-16 所示。



图 1-16 TKS 系列仿真器

TKS 系列仿真器除可以使用本身自带的仿真软件外,还可以嵌入到 Keil C51 中进行硬件调试仿真,其加载方法是:①将随机提供的 TKS 仿真器驱动文件 TKS_DEB.DLL 复制到 Keil 的安装目录 C51\Bin 下;②打开 Keil 安装目录下的 Took.ini 文件,在多个分类中找到 [C51],并加入语句“TDRV3=C:\Keil\C51\bin\TKS_DEB.dll (“TKS Debugger”)”。

TDRV3 是驱动 DLL 的序号,其数值可改变。C:\Keil\C51\bin 为安装目录,在此假设 Keil 软件安装在 C 盘根目录下。

将 TKS_DEB.DLL 加载到 Keil C51 中后,在图 1-10 所示的“Options for Target 'Target 1'”对话框的“Device”选项卡中先选择硬件仿真,并选中合适的仿真器类型,然后单击硬件仿真器参数设置按钮,将弹出图 1-17 所示的“Target Setup”窗口,在该窗口中进行相应的仿真硬件配置。

仿真硬件配置好后,就可以在 Keil C51 编译环境中使用 TKS 仿真器进行硬件仿真的了,其硬件仿真调试方法与软件仿真调试方法类似。

伟福仿真器使用自带的 WAVE 仿真软件,进行硬件仿真时,也需进行相应的设置。

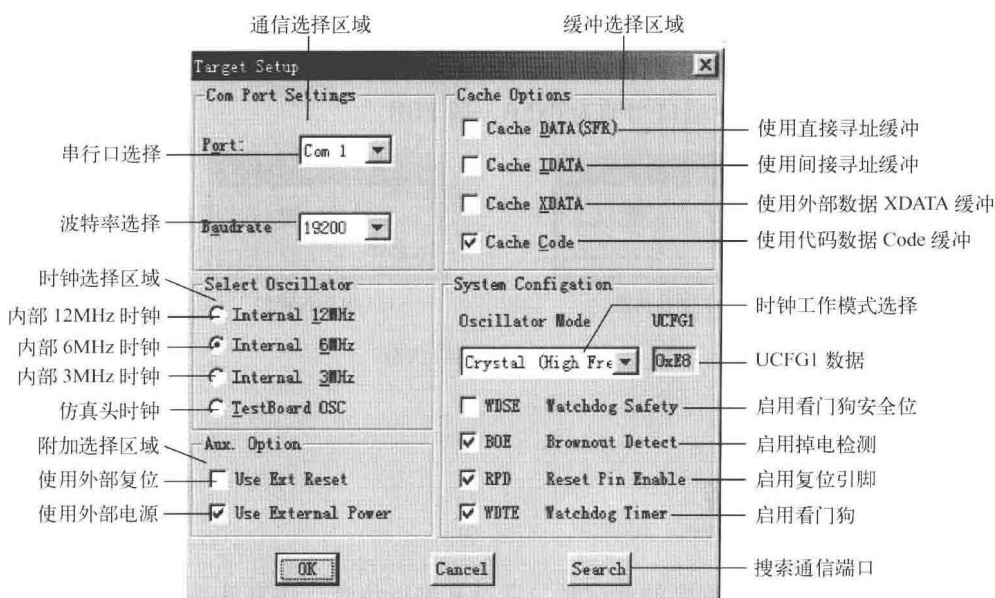


图 1-17 仿真硬件配置

因 80C51 系列的单片机型号较多,且存储器容量大小、功能等方面有所不同,因此需要选择合适的仿真头对单片机进行硬件仿真。



1.3 编程器

编程器又称程序固化器,是将调试生成的 .BIN 或 .HEX 文件固化到存储器中的设备。不同型号的单片机或存储器,厂家都要为其提供配套的编程器进行程序固化。由于生产厂家众多,芯片型号繁多,不可能每一种芯片都由一个专用的编程器对其进行程序固化,因此有些公司研究出通用编程器,通用编程器可以支持多种型号芯片程序的读/写操作。常见的通用编程器有南京西尔特电子有限公司的 SUPERPRO 通用编程器和周立功公司生产的 EasyPRO 系列通用编程器。EasyPRO 系列编程器如图 1-18 所示。

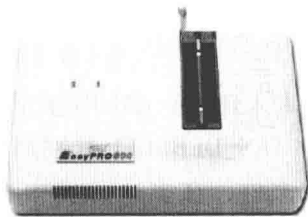



图 1-18 EasyPRO 系列编程器

南京西尔特公司生产的 SUPERPRO 是一种可靠性高、速度快、性价比较高的通用编程器,能够直接与计算机的并行打印口或 USB 口相连,对数十个厂家生产的 PLD、EPROM、Flash、BPROM、MCU/MPU、DRAM/SRAM 等数千种芯片进行可编程操作。

SUPERPRO 软件可选择中文或英文两种语言进行安装。软件安装好后,打开软件时,将弹出计算机与编程器的连接信息。

使用编程器时,首先将芯片放在锁紧座中(注意,芯片的方向不要弄错),然后将芯片锁紧。打开编程器电源,与计算机进行连接。

执行菜单命令“器件”→“选择器件”,或者直接单击工具栏的图标,弹出“选择