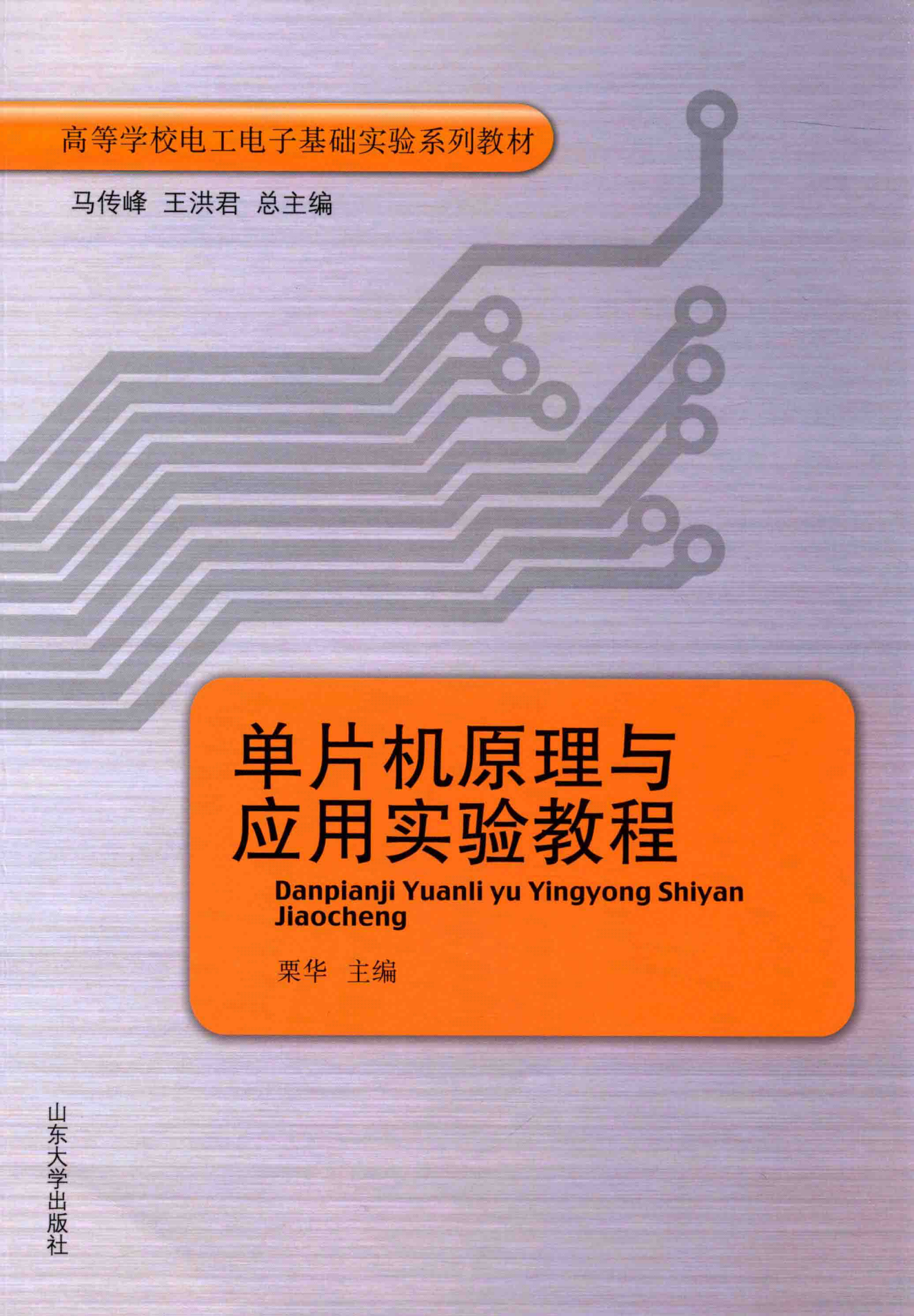


A decorative graphic of a circuit board pattern, consisting of multiple parallel lines that zig-zag across the page, ending in small circles, resembling a stylized 'S' or a series of connected paths.

高等学校电工电子基础实验系列教材

马传峰 王洪君 总主编

单片机原理与 应用实验教程

Danpianji Yuanli yu Yingyong Shiyan
Jiaocheng

栗华 主编

山东大学出版社

高等学校电工电子基础实验系列教材
国家“十一五”规划教材配套实验教材

单片机原理与应用实验教程

主 编	栗 华			
副主编	王晶晶	李庆华	陈江华	田存伟
审 阅	王洪君			

山东大学出版社

图书在版编目(CIP)数据

单片机原理与应用实验教程/栗华主编. —济南:山东大学出版社, 2015. 4

高等学校电工电子基础实验系列教材/马传峰, 王洪君总主编

ISBN 978-7-5607-5262-4

I. ①单… II. ①栗… III. ①单片微型计算机—高等学校—教材 IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2015)第 073535 号

责任策划:刘旭东

责任编辑:宋亚卿

封面设计:张 荔

出版发行:山东大学出版社

社 址:山东省济南市山大南路 20 号

邮 编:250100

电 话:市场部(0531)88364466

经 销:山东省新华书店

印 刷:泰安金彩印务有限公司

规 格:787 毫米×1092 毫米 1/16

20.75 印张 477 千字

版 次:2015 年 4 月第 1 版

印 次:2015 年 4 月第 1 次印刷

定 价:35.00 元

版权所有,盗印必究

凡购本书,如有缺页、倒页、脱页,由本社营销部负责调换

《高等学校电工电子基础实验系列教材》

编委会

主 任 马传峰 王洪君

副主任 郁 鹏 邢建平

委 员 (按姓氏笔画排序)

于欣蕾 万桂怡 王丰晓 王春兴 朱瑞富

孙国霞 孙梅玉 杨霓清 李 蕾 李德春

邱书波 郑丽娜 赵振卫 姚福安 栗 华

高 瑞 高洪霞 韩学山

内容简介

本书是国家“十一五”规划教材《单片机原理与应用》的配套实验教材。它是根据《单片机原理与应用》的实验教学需要,结合作者多年来的实验教学经验而编写的。本书设计了软件仿真实验、硬件接口实验和系统综合实验三个模块。实验内容涵盖了 MCS-51 单片机教学大纲中的汇编语言及 C 语言程序设计、片内外设及中断系统应用、片外存储器扩展、I/O 扩展、人机交互(包括键盘输入、LED 数码管显示、LCD 显示以及 LED 点阵显示)扩展、信号的输入/输出通道(包括常用的传感器、A/D、D/A 和开关量的输入/输出)扩展,以及综合性系统设计等。内容全面,贴近工程实际,注重能力培养。

本书既可作为电子信息、通信工程、计算机、自动化等专业专科生、本科生的单片机原理与应用实验教材,也可作为大学生电子设计大赛的培训教材,还可作为从事单片机开发应用工作的工程技术人员的自学实训书籍。

前 言

单片机这种 20 世纪 70 年代诞生的专用于小型智能控制领域的计算机是嵌入式计算机的一种,也是到目前为止应用最广泛的一种专用计算机。MCS-51 单片机以其集成度高、体积小、可靠性高、抗干扰能力强、控制功能灵活、可扩展性好、性价比高等特点,不仅成为了嵌入式计算机发展历史上的里程碑,而且直到现在仍然是嵌入式计算机的典型代表。学习单片机是电气、电子、自动化领域的学生学习智能控制、智能仪器仪表设计的入门基础。很多高水平的电子设计工程师都是从学习单片机尤其是 MCS-51 单片机开始的。

目前,各种以 MCS-51 单片机为原型的单片机书籍非常多,且基本上分成两类。第一类主要侧重 MCS-51 单片机的原理,把主要精力都放在了单片机芯片内部的介绍,而对于片外扩展的介绍却不够详细,对于现在工程设计中常用的一些集成度更高、采用串行总线接口的接口芯片几乎很少介绍。第二类则主要侧重 MCS-51 单片机的接口技术,把主要精力都放在了单片机的外部接口上,接口芯片介绍得比较多,种类也比较丰富,而对于单片机内部的知识却一带而过,对于第一类书籍,比较适合单片机的初学者学习,但是,所学的知识又很难应用到工程实践中。大多数的单片机的应用工程师最初从课本上所获得的知识非常有限,很多知识都是靠自己多年的工程实践慢慢积累的,而在此过程中也难免走了很多弯路。对于第二类书籍,一般要求读者已经具有单片机的基本知识,甚至已有一定的应用经验,书中的知识只是工程应用中的一种参考资料,不太适合初学者学习。

为此,我们于 2008 年编写了《单片机原理与应用》一书,该书被评为国家级“十一五”规划教材。本书将单片机的原理和接口知识以及工程应用知识很好地结合在了一起。在原理部分,把概念阐述清楚,把知识体系提取出来,使读者在最短的时间内掌握最核心、最主要的系统理论。在接口部分,密切结合实际,介绍了工程实践中常用的接口芯片及其在单片机应用系统中的应用方法,给出了连接示意电路图和主要接口程序。在系统设计部分,立足工程实践,把常遇到的各种因素详细地介绍给读者。精简而系统的理论知识、实用的接口技术和工程中常用的实战经验是本书相辅相成、缺一不可的三个组成部分。

自 2009 年 1 月以来,《单片机原理与应用》已经过两次印刷,销量在 1 万册以上,

被山东大学、齐鲁工业大学、山东师范大学、山东电子职业技术学院等多家院校选定为专业必修课“单片机原理与应用”的指定教材,使用效果良好。

在本书的使用过程中,用户感到不方便的是现有的实验教学仪器和实验教学计划与本教材无法配套,因而影响了本教材的实际使用效果。导致出现这一现象的原因是现有的实验教学仪器一般都是 2000 年左右配备的,这些实验教学仪器以及当时的实验教学计划侧重的是 MCS-51 单片机内部的原理性验证,接口实验较少,并且大部分的接口实验都是针对传统的并行接口芯片,这些芯片在现实工程中已很少使用。现实工程中常用的一些串行接口芯片在教材中已有大量篇幅介绍,但由于传统的实验教学仪器不支持,所以无法进行实验验证。此外,现实工程中用于单片机前向通道的传感器和用于后向通道的驱动器,传统的实验教学仪器支持的也很少。对于系统性的综合实验,虽然传统的教学仪器与教学计划也有所涉及,但一般能够开展的创新性、综合性实验也相对比较简单,一些类似于近年来在全国大学生电子设计大赛中出现的设计题目很难基于传统的实验教学仪器进行。

为此,我们设计了新的“SDU_MCU 系列单片机教学实验箱”,并编写了本实验教材。

本实验教材设计了软件仿真实验、硬件接口实验、系统综合实验三个模块。内容涵盖了整个 MCS-51 单片机教学计划中的汇编语言及 C 语言程序设计、片内外设及中断系统应用、片外存储器扩展、I/O 扩展、人机交互(包括键盘输入、LED 数码管显示、LCD 显示以及 LED 点阵显示)扩展、信号的输入/输出通道(包括常用的传感器、A/D、D/A 和开关量的输入/输出)扩展,以及综合性系统设计等。整个实验教学计划按 32~48 学时设计,具体的实验内容如下:

(1)纯软件仿真实验(4 个):单片机软件开发环境 Keil μ Vision2 的使用以及以此为基础的纯软件仿真实验共有 4 个,其中,2 个汇编语言程序设计实验,1 个 C 语言程序设计实验,1 个 C 语言与汇编语言混合编程实验。这部分实验的目的是让学生初步熟悉开发环境,熟悉汇编语言与 C 语言单片程序设计的基本语法规则,为后面的实验内容打下基础。

(2)Proteus 软件仿真实验 12 个,主要目的是训练学生掌握 Proteus 这种仿真工具软件的使用方法,并利用其进行初级系统设计与仿真验证。这种仿真设计方法可以提高系统设计效率,缩短设计周期,降低设计成本,在现代单片机应用系统设计中是非常实用的。

(3)硬件接口实验(26 个):使用硬件仿真调试器,基于“SDU_MCU 系列单片机教学实验箱”实现的硬件接口实验共有 26 个,分别是:按键声光报警实验 1 个,并口扩展实验 1 个,串并转换实验 1 个,并串转换实验 1 个,8255 接口实验 2 个,7279 键盘/动态 LED 显示实验 1 个,LCD 1602 显示实验 1 个,LCD 12864 显示实验 1 个,16 $\times$ 16 LED 点阵显示实验 1 个,I²C 扩展实验 1 个,RTC 实验 2 个,存储器扩展实验 1 个,A/D 实验 2 个,D/A 实验 2 个,温度测量实验 1 个,温湿度测量实验 1 个,红外对管发射接收与红外遥控实验 1 个,红外热释电实验 1 个,超声波测距实验 1 个,电机驱

动实验 3 个。这部分实验的目的是训练学生设计单片机硬件系统,并掌握实际硬件电路的调试方法。

(4)系统综合实验 4 个,分别是:电梯控制系统实验,自动国旗升降系统实验,基于单片机控制的函数发生器系统实验和分布式远程抄表系统实验。这部分实验属于综合创新实验,教材中只给出功能要求,没有过多的指导,主要靠学生根据前期的训练,自己设计系统软硬件结构,并实现其功能。

一般情况下,要完成这些实验至少需要 48 学时,其中,软件仿真实验、大部分硬件接口实验 1 个学时基本上可以完成 1 个实验,而部分较复杂的硬件接口实验可能需要安排 2 个或 2 个以上学时来完成,系统综合实验中的每个实验至少需要 3 个或 4 个学时。由于本实验箱采用模块化结构,便于拆解和携带,对于较为复杂的硬件实验,可以将所需要的模块带离实验室进行。当实验学时较少时,也可以根据学时安排选做其中的部分实验。

本书由栗华任主编并统稿,王晶晶、李庆华、陈江华、田存伟任副主编,王洪君进行了审阅。山东大学信息科学与工程学院硕士研究生王增迪、马良、宋玉川、薛树滨同学参与了本书实验箱硬件电路的制板与调测,研究生李峰、于涛、陈召村等同学参与了本书实验程序的验证和插图整理等工作;山东师范大学物理与电子科学学院、齐鲁工业大学电气工程与自动化学院聊城大学物理科学与信息工程学院在本书编写过程中给予了许多良好的建议和指导;山东大学资产与实验室管理部及山东大学国家级物联信息技术与系统工程实验教学示范中心有关领导在本书编写过程中给予了大量的帮助和鼓励,在此一并表示感谢。

感谢山东大学出版社对本书出版给予的支持和帮助。

由于作者水平有限,书中如有错误和不当之处,恳请读者批评指正。

编 者

2014 年 10 月

目 录

第 1 章 集成开发环境 Keil μ Vision2 及 Proteus 仿真软件使用指南	(1)
1.1 集成开发环境 Keil μ Vision2 应用指南	(1)
1.2 系统仿真软件 Proteus 应用指南	(25)
第 2 章 软件实验	(54)
2.1 汇编程序实验 1	(54)
2.2 汇编程序实验 2	(57)
2.3 C 语言程序实验	(60)
2.4 C 语言与汇编语言混合编程实验	(62)
第 3 章 Proteus 系统仿真实验	(66)
3.1 基本并行 I/O 口实验	(66)
3.2 扩展并行 I/O 口实验	(71)
3.3 静态 LED 显示实验	(77)
3.4 矩阵键盘扫描实验	(84)
3.5 定时/计数器实验	(96)
3.6 串口通信实验	(104)
3.7 LCD 1602 显示实验	(114)
3.8 ADC0808/9 信号采集实验	(130)
3.9 DAC0832 应用实验	(144)
3.10 I ² C 扩展实验	(158)
3.11 SPI 扩展实验	(180)
3.12 LCD 12864 显示实验	(190)
第 4 章 硬件接口实验	(205)
4.1 按键声光报警实验	(205)
4.2 74LS373 并口扩展实验	(208)

4.3	74LS165 并/串行 I/O 口转换实验	(211)
4.4	74LS164 串/并行 I/O 口转换实验	(215)
4.5	8255 并行 I/O 扩展及交通信号灯控制实验	(218)
4.6	8255 动态 LED 显示实验	(222)
4.7	7279 键盘扫描及动态 LED 显示实验	(225)
4.8	7279 键盘扫描及 LCD 1602 显示实验	(230)
4.9	LCD 12864 显示实验	(234)
4.10	16×16 LED 点阵显示实验	(236)
4.11	ADC0809 并行接口 A/D 转换实验	(243)
4.12	DAC0832 并行接口 D/A 转换实验	(247)
4.13	DS12887 并行接口 RTC 实验	(252)
4.14	I ² C 串行 E ² PROM 24C02 读写实验	(258)
4.15	I ² C 接口芯片 PCF8574 扩展并口实验	(263)
4.16	I ² C 接口芯片 PCF8563 扩展 RTC 实验	(266)
4.17	I ² C 接口芯片 AD7810 扩展 A/D 实验	(271)
4.18	I ² C 接口芯片 TLC5615 扩展 D/A 实验	(275)
4.19	DS18B20 温度测量实验	(280)
4.20	DHT11 温湿度测量实验	(284)
4.21	红外对管障碍物检测实验	(289)
4.22	红外热释电实验	(293)
4.23	超声波测距实验	(296)
4.24	步进电机驱动实验	(301)
4.25	直流电机驱动及转速测量实验	(304)
4.26	电机 PWM 调速实验	(307)
第 5 章	系统综合实验	(312)
5.1	单片机应用系统设计流程简介	(312)
5.2	电梯控制系统实验	(317)
5.3	程控函数发生器系统实验	(318)
5.4	自动国旗升降系统实验	(319)
5.5	自动远程抄表系统实验	(320)
参考文献		(321)

第 1 章 集成开发环境 Keil μ Vision2 及 Proteus 仿真软件使用指南

1.1 集成开发环境 Keil μ Vision2 应用指南

Keil 软件是众多单片机应用开发软件中的优秀软件之一,它集编辑、编译、仿真于一体,支持汇编语言、PLM 语言和 C 语言的程序设计,界面友好,易学易用。

下面通过图解的方式来介绍 Keil μ Vision2 软件的使用,学习如何通过输入源程序→新建工程→工程详细设置→源程序编译得到目标代码文件。

1.1.1 Keil μ Vision2 软件的启动

首先双击 Keil 的桌面快捷方式(见图 1.1),启动 Keil 集成开发软件。



图 1.1 Keil μ Vision2 的桌面图标

启动 Keil μ Vision2 后,将出现如图 1.2 所示界面。



图 1.2 Keil μ Vision2 的启动界面

然后,将出现如图 1.3 所示主界面。

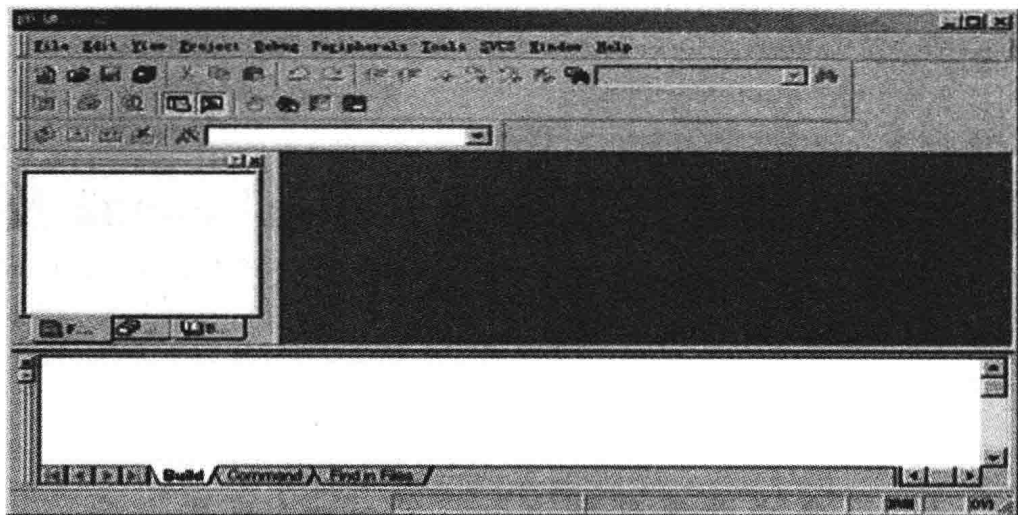


图 1.3 Keil μ Vision2 的主界面

1.1.2 Keil μ Vision2 主界面简介

Keil μ Vision2 的主界面窗口(见图 1.4)主要包括项目窗口(用于管理项目各组成文件)、文本编辑窗口(用于输入、编辑和查看文件代码)和输出窗口(用于显示文件编译结果),还有一些调试窗口。

Keil μ Vision2 的主界面窗口还有标题栏、菜单栏和工具栏,且大部分的菜单操作和工具栏操作都有对应的快捷键。表 1.1 和表 1.2 分别列出了 μ Vision2 的主要菜单命令、工具栏图标、默认的快捷键以及对它们的描述。

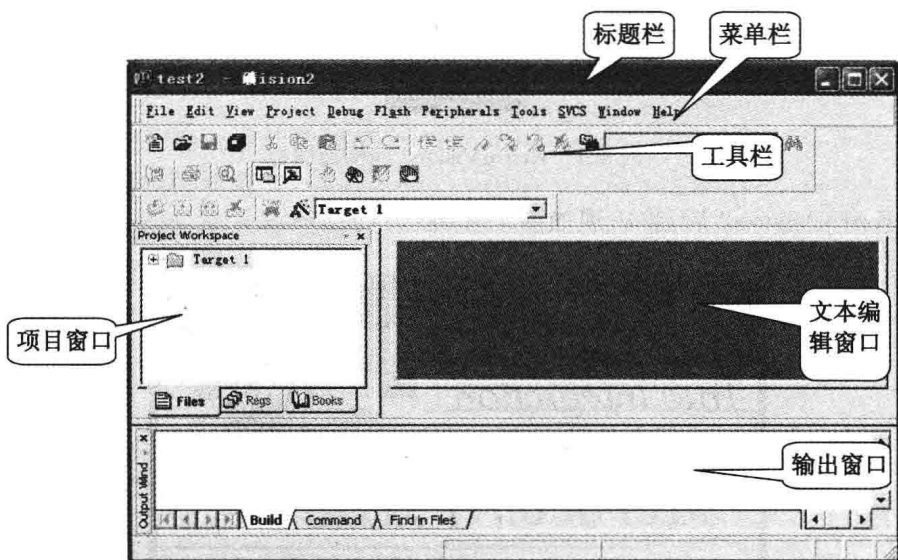


图 1.4 Keil μ Vision2 的主界面窗口

(1) 项目菜单和项目命令(见表 1.1)

表 1.1 项目菜单和项目命令

菜单命令	工具栏图标	快捷键	描述
New Project...			创建新项目
Open Project...			打开一个已经存在的项目
Close Project			关闭当前的项目
Select Device for Target			选择对象的 CPU
Remove Item			从项目中移走一个组或文件
Options ...		Alt+F7	设置对象、组或文件的工具选项
File Extensions, Book and Environment			选择不同文件类型的扩展名
Build Target		F7	编译修改过的文件并生成应用
Rebuild All Target Files			重新编译所有的文件并生成应用
Translate ...		Ctrl+F7	编译当前文件
Stop Build			停止生成应用的过程

(2) 调试菜单和调试命令(见表 1.2)

表 1.2 调试菜单和调试命令

菜单命令	工具栏图标	快捷键	描述
Start/Stop Debugging		Ctrl+F5	开始/停止调试模式
Go		F5	运行程序,直到遇到一个中断
Step		F11	单步执行程序,遇到子程序则进入
Step over		F10	单步执行程序,跳过子程序
Step out of Current Function		Ctrl+F11	执行到当前函数的结束
Stop Runing		Esc	停止程序运行
Breakpoints...			打开断点对话框
Insert/Remove Breakpoint			设置/取消当前行的断点
Enable/Disable Breakpoint			使能/禁止当前行的断点

续表

菜单命令	工具栏图标	快捷键	描述
Disable All Breakpoints			禁止所有的断点
Kill All Breakpoints			取消所有的断点
Memory Map...			打开存储器空间设置对话框
Performance Analyzer...			打开设置性能分析的窗口

1.1.3 Keil μ Vision2 创建项目实例

μ Vision2 包括一个项目管理器,它可以使 MCS-51 应用系统的设计变得简单。要创建一个应用,需要按下列步骤进行操作:

- ①新建一个项目文件并从器件库中选择一个器件。
- ②新建一个源文件并把它加入到项目中。
- ③针对目标硬件进行选项设置。
- ④编译项目并生成可编程 PROM 的 HEX 文件。

下面将逐步地进行描述,从而指引读者创建一个简单的 μ Vision2 项目。

第一步:双击 Keil μ Vision2 的桌面快捷方式,启动 Keil 集成开发软件。

第二步:新建文本编辑窗。单击工具栏上的“新建文件”按钮  ,或者单击“File”→“New...”菜单命令,即可在项目窗口的右侧打开一个新的文本编辑窗,如图 1.5 所示。

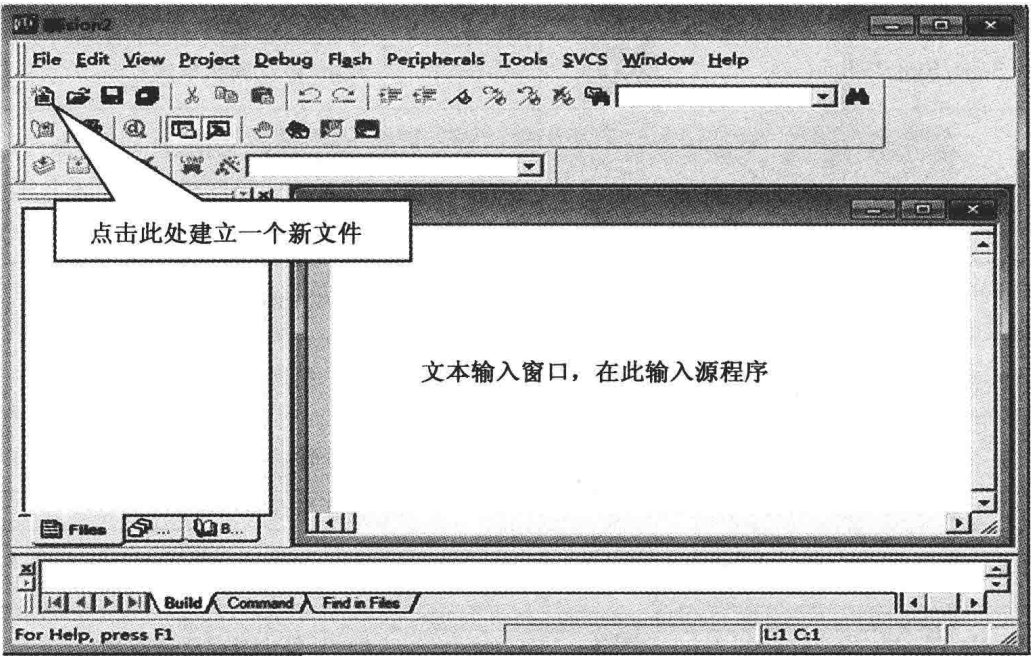


图 1.5 新建文本编辑窗

第三步:输入源程序。在新的文本编辑窗中输入源程序,可以输入 C 语言程序,也可以输入汇编语言程序,如图 1.6 所示。

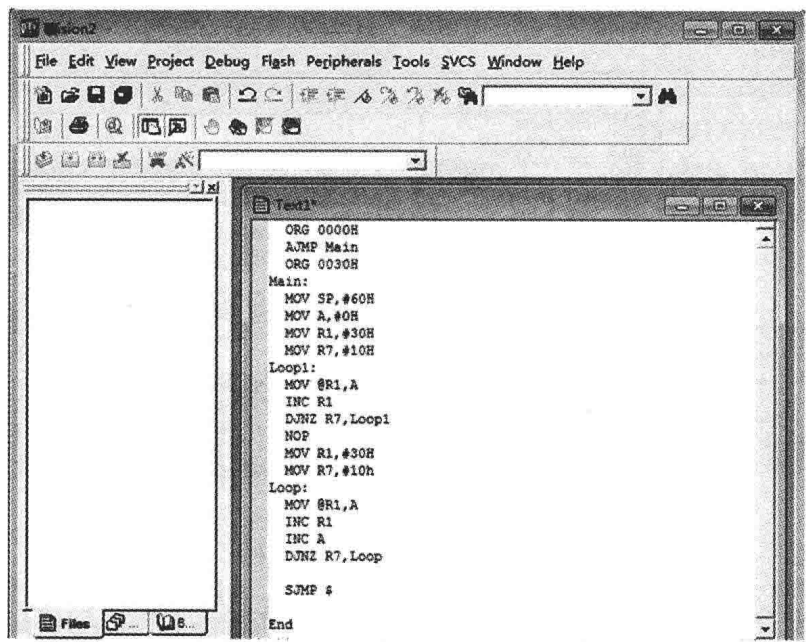


图 1.6 输入源程序

第四步:保存源程序。单击工具栏上的“保存文件”按钮,或单击“File”→“Save”菜单命令保存源程序(见图 1.7),会弹出一个对话框,如图 1.8 所示。应在对话框中的“文件名”栏输入要保存的文件名。注意:保存文件时,必须加上文件的扩展名。如果你使用汇编语言编程,那么保存时文件的扩展名为“.asm”;如果是 C 语言程序,文件的扩展名使用“.c”。

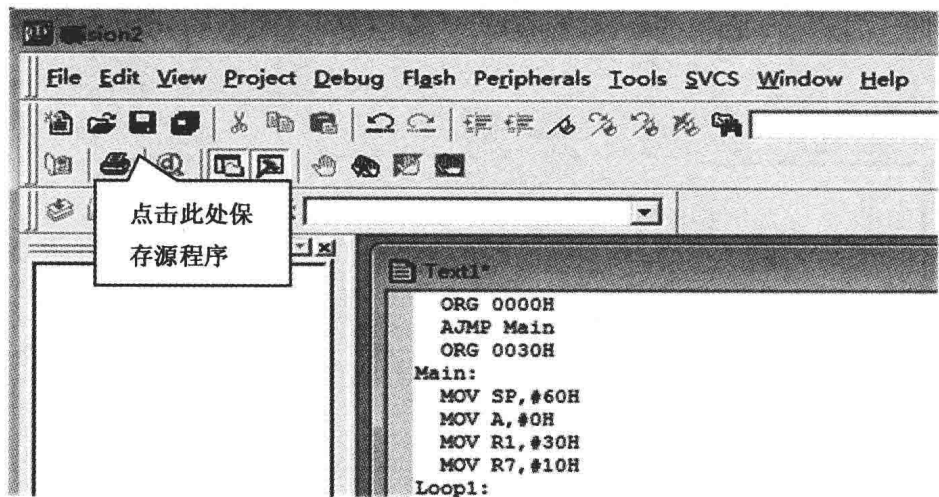


图 1.7 保存源程序

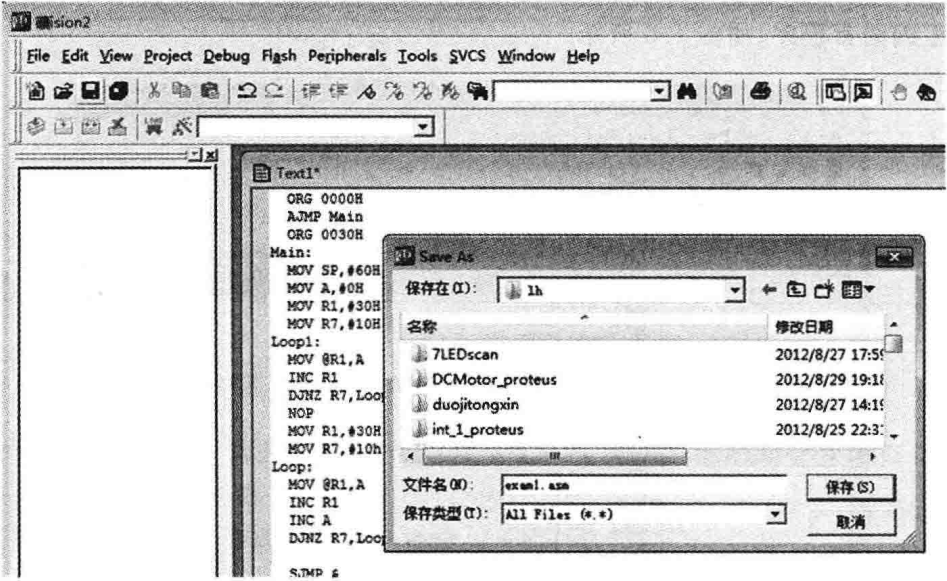


图 1.8 “保存源程序”对话框

注:第三步和第四步之间的顺序可以互换,即可以先输入源程序后保存,也可以先保存后输入源程序。

第五步:新建 Keil 工程。如图 1.9 所示,单击“Project”→“New Project ...”菜单命令,将出现如图 1.10 所示对话框。

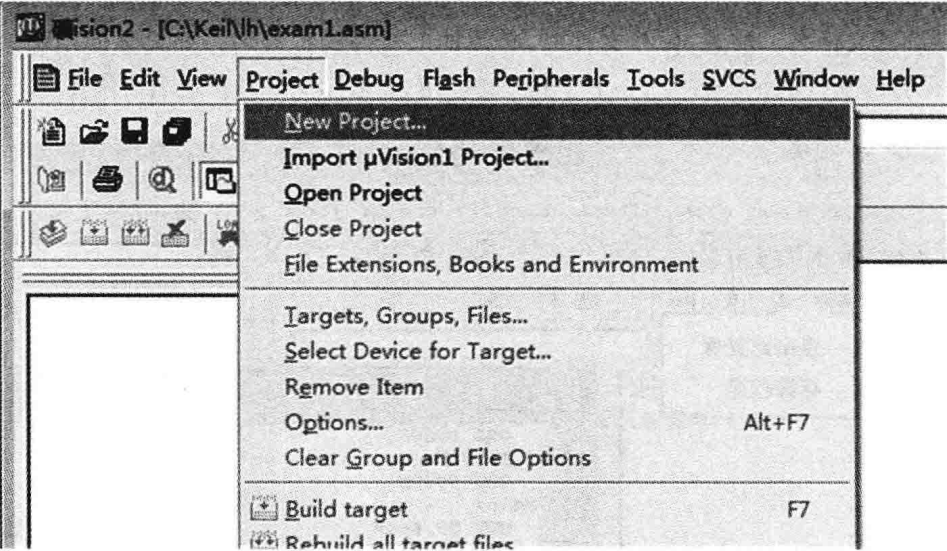


图 1.9 新建工程

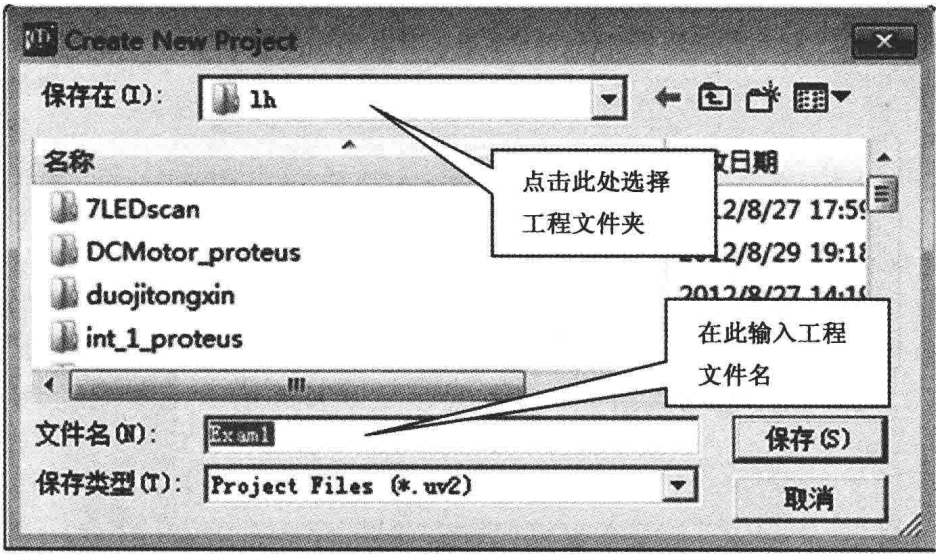


图 1.10 “新建工程”对话框

在“保存工程”对话框中输入工程文件名,Keil 工程默认扩展名为“.uv2”。工程名称不用输入扩展名(见图 1.10)。输入名称后保存,将出现“选择设备”对话框。

第六步:选择 CPU 型号。如图 1.11 所示,为工程选择 CPU 型号。本新建工程选择了 Atmel 公司的 AT89C51 单片机。

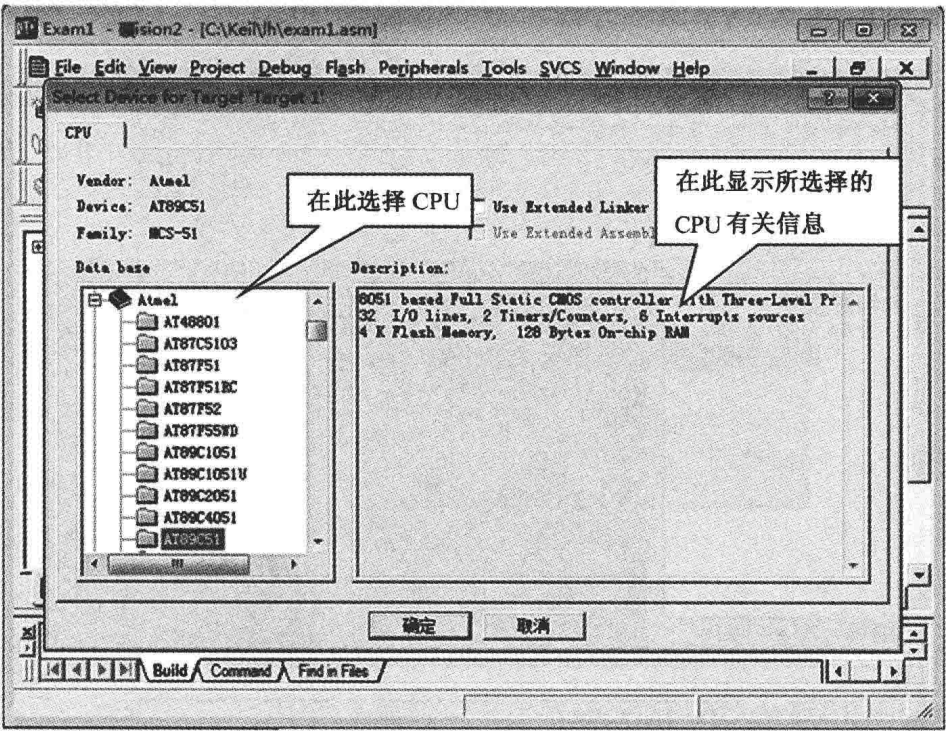


图 1.11 “选择 CPU 型号”对话框