

单片机 应用技术

主 编 覃奈新
副主编 欧福强
参 编 谢 托



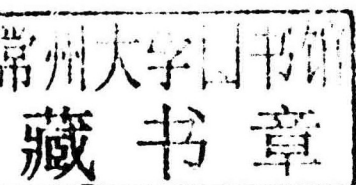
广西人民出版社

单片机应用技术

主 编 覃奈新

副主编 欧福强

参 编 谢 托



 广西人民出版社

图书在版编目 (C I P) 数据

单片机应用技术 / 覃奈新主编. —南宁: 广西人民出版社, 2013.9

ISBN 978- 7- 219- 08527- 1

I. ①单… II. ①覃… III. ①单片微型计算机 IV. ①TP368.1

中国版本图书馆 CIP 数据核字 (2013) 第 206252 号

出版发行 广西人民出版社
社 址 广西南宁市桂春路 6 号
邮 编 530028
责任编辑 李带舅 廖集玲
封面设计 陆文渲

印 制 南宁市开源彩色印刷有限公司
开 本 787mm× 1092mm 1/16
印 张 3.25
字 数 77 千字
版 次 2013 年 9 月第 1 版
印 次 2013 年 9 月第 1 次印刷

书 号 ISBN 978- 7- 219- 08527- 1/ T·40
定 价 9.80 元

版权所有 翻版必究

前 言

本课程的设计突破了学科体系模式，打破了原来各学科体系的框架，将各学科的内容按“项目”进行整合。本课程的“项目”以职业实践活动为主线，因而，它是跨学科的，且理论与实践一体化。强调学生个人适应劳动力市场变化的需要。因而，本课程的设计兼顾了企业和个人两者的需求，着眼于人的全面发展，以培养全面素质为基础，以提高综合职业能力为核心。

本课程包含了单片机应用技术的五个项目，每个项目均由若干个具体的典型工作任务组成，每个任务均将相关知识和实践（含实验）过程有机结合，力求体现“做中学”、“学中做”的教学理念；本课程内容的选择上降低理论重心，突出实际应用，注重培养学生的应用能力和解决问题的实际工作能力。

本教材建议课时数为 72 学时，安排建议见下表(仅供参考)。

项 目	教学内容	学 时
1	项目一 制作闪光灯	12
2	项目二 制作流水灯	14
3	项目三 制作数计时器	16
4	项目四 制作电子时钟	20
5	项目五 制作智能小车	20
合 计		72

本教材由柳州市第一职业技术学校覃奈新老师任主编、柳州市第一职业技术学校欧福强老师任副主编，参与编写人员全部为柳州市第一职业技术学校老师。具体编写分工为：覃奈新（项目一、项目二）、欧福强（项目三、项目四）、谢托（项目五）。

编者水平有限，书中难免存在不足之处甚至错误，我们恳请使用本教材的师生对教材中的问题提出批评、建议和意见，以便进一步完善本教材。

编 者
2012 年 5 月

目 录

项目一 制作闪光灯	1
项目二 制作流水灯	18
项目三 制作计时器	21
项目四 制作电子时钟	27
项目五 制作智能小车	41

项目一 制作闪光灯

一、学习目标:

- 1.了解 AT89C51 系列单片机基本知识。
- 2.掌握单片机最小化系统的构成条件及电路。
- 3.掌握目标代码的下载方法。

二、知识点介绍

单片微型计算机简称单片机，是典型的嵌入式微控制器（Microcontroller Unit），常用英文字母的缩写 MCU 表示单片机，单片机又称单片微控制器，它不是完成某一个逻辑功能的芯片，而是把一个计算机系统集成到一个芯片上。单片机由运算器，控制器，存储器，输入输出设备构成，相当于一个微型的计算机（最小系统），和计算机相比，单片机缺少了外围设备等。概括的讲：一块芯片就成了一台计算机。它的体积小、质量轻、价格便宜、为学习、应用和开发提供了便利条件。同时，学习使用单片机是了解计算机原理与结构的最佳选择。它最早是被用在工业控制领域。

由于单片机在工业控制领域的广泛应用，单片机由仅有 CPU 的专用处理器芯片发展而来。最早的设计理念是通过将大量外围设备和 CPU 集成在一个芯片中，使计算机系统更小，更容易集成进复杂的而对体积要求严格的控制设备当中。

INTEL 的 8080 是最早按照这种思想设计出的处理器，当时的单片机都是 8 位或 4 位的。其中最成功的是 INTEL 的 8031，此后在 8031 上发展出了 MCS51 系列单片机系统。因为简单可靠而性能不错获得了很大的好评。尽管 2000 年以后 ARM 已经发展出了 32 位的主频超过 300M 的高端单片机，直到现在基于 8031 的单片机还在广泛的使用。在很多方面单片机比专用处理器更适合应用于嵌入式系统，因此它得到了广泛的应用。事实上单片机是世界上数量最多处理器，随着单片机家族的发展壮大，单片机和专用处理器的发展便分道扬镳。

单片机诞生于 1971 年，经历了 SCM、MCU、SoC 三大阶段，早期的 SCM 单片机都是 8 位或 4 位的。其中最成功的是 INTEL 的 8031，此后在 8031 上发展出了 MCS51 系列 MCU 系统。基于这一系统的单片机系统直到现在还在广泛使用。随着工业控制领域要求的提高，开始出现了 16 位单片机，但因为性价比不理想并未得到很广泛的应用。90 年代后随着消费电子产品大发展，单片机技术得到了巨大提高。随着 INTEL i960 系列特别是后来的 ARM 系列的广泛应用，32 位单片机迅速取代 16 位单片机的高端地位，并且进入主流市场。

而传统的 8 位单片机的性能也得到了飞速提高，处理能力比起 80 年代提高了数百倍。高端的 32 位 Soc 单片机主频已经超过 300MHz，性能直追 90 年代中期的专用处理器，而普通的型号出厂价格跌落至 1 美元，最高端的型号也只有 10 美元。

当代单片机系统已经不再只在裸机环境下开发和使用，大量专用的嵌入式操作系统被广泛应用在全系列的单片机上。而在作为掌上电脑和手机核心处理的高端单片机甚至可以直接使用专用的 Windows 和 Linux 操作系统。

现代人类生活中所用的几乎每件有电子器件的产品中都会集成有单片机。手机、电话、计算器、家用电器、电子玩具、掌上电脑以及鼠标等电子产品中都含有单片机。汽车上一般配备 40 多片单片机，复杂的工业控制系统上甚至可能有数百片单片机在同时工作！单片机的数量不仅远超过 PC 机和其他计算的总和，甚至比人类的数量还要多。

单片机渗透到我们生活的各个领域，几乎很难找到哪个领域没有单片机的踪迹。导弹的导航装置，飞机上各种仪表的控制，计算机的网络通讯与数据传输，工业自动化过程的实时控制和数据处理，广泛使用的各种智能 IC 卡，民用豪华轿车的安全保障系统，录像机、摄像机、全自动洗衣机的控制，以及程控玩具、电子宠物等等，这些都离不开单片机。更不用说自动控制领域的机器人、智能仪表、医疗器械以及各种智能机械了。因此，单片机的学习、开发与应用将造就一批计算机应用与智能化控制的科学家、工程师。

单片机广泛应用于仪器仪表、家用电器、医用设备、航空航天、专用设备的智能化管理及过程控制等领域，大致可分如下几个范畴：

智能仪器

单片机具有体积小、功耗低、控制功能强、扩展灵活、微型化和使用方便等优点，广泛应用于仪器仪表中，结合不同类型的传感器，可实现诸如电压、电流、功率、频率、湿度、温度、流量、速度、厚度、角度、长度、硬度、元素、压力等物理量的测量。采用单片机控制使得仪器仪表数字化、智能化、微型化，且功能比起采用电子或数字电路更加强大。例如精密的测量设备（电压表、功率计，示波器，各种分析仪）。

工业控制

单片机具有体积小、控制功能强、功耗低、环境适应能力强、扩展灵活和使用方便等优点，用单片机可以构成形式多样的控制系统、数据采集系统、通信系统、信号检测系统、无线感知系统、测控系统、机器人等应用控制系统。例如工厂流水线的智能化管理，电梯智能化控制、各种报警系统，与计算机联网构成二级控制系统等。

家用电器

家用电器广泛采用了单片机控制，从电饭煲、洗衣机、电冰箱、空调机、彩电、其他音响视频器材、再到电子秤量设备和白色家电等。

网络和通信

现代的单片机普遍具备通信接口，可以很方便地与计算机进行数据通信，为在计算机网络和通信设备间的应用提供了极好的物质条件，通信设备基本上都实现了单片机智能控制，从手机，电话机、小型程控交换机、楼宇自动通信呼叫系统、列车无线通信、再到日常工作中随处可见的移动电话，集群移动通信，无线电对讲机等。

设备领域

单片机在医用设备中的用途亦相当广泛，例如医用呼吸机，各种分析仪，监护仪，超声诊断设备及病床呼叫系统等等。

模块化系统

某些专用单片机设计用于实现特定功能，从而在各种电路中进行模块化应用，而不要求使用人员了解其内部结构。如音乐集成单片机，看似简单的功能，微缩在纯电子芯片中（有别于磁带机的原理），就需要复杂的类似于计算机的原理。如：音乐信号以数字的形式存于存储器中（类似于 ROM，由微控制器读出，转化为模拟音乐电信号（类似于声卡）。

在大型电路中，这种模块化应用极大地缩小了体积，简化了电路，降低了损坏、错误率，也方便于更换。

汽车电子

单片机在汽车电子中的应用非常广泛，例如汽车中的发动机控制器，基于 CAN 总线的汽车发动机智能电子控制器、GPS 导航系统、abs 防抱死系统、制动系统、胎压检测等。

此外，单片机在工商、金融、科研、教育、电力、通信、物流和国防航空航天等领域都有着十分广泛的用途。

Mcs-51 单片机简介

由于在 8 位单片机中性能突出，8051 单片机已成为主流单片机。日前它仍是我国单片机应用的主流。标准的 51 单片机一般都有 4 个 8 位的并行 I/O 口、1 个全双工的串行接口、2 个 16 位的定时器 / 计数器、5 个中断源、2 个中断优先级、21 个特殊功能寄存器、1 个数据指针和 1 个程序指针等资源。

（一）认识单片机的引脚：

AT89S51 有 PDIP、PLCC、TQFP 三种封装方式，其中最常见的是采用 40 封装的双列直插 PDIP 封装，外形结构下图。



图 1.1 单片机芯片

芯片共有 40 个引脚，引脚的排列顺序为从靠芯片的缺口左边那列引脚逆时针数起，依次为 1、2、3、4...40，其中芯片的 1 脚顶上有个凹点。在单片机的 40 个引脚中，电源引脚 2 根，外接晶体振荡器引脚 2 根，控制引脚 4 根以及 4 组 8 位可编程 I/O 引脚 32 根。

1. 主电源引脚（2 根）

V_{CC} （40）：电源输入，接 +5V 电源

G_{ND} （20）：接地线

2. 外接晶振引脚 (2 根)

XTAL1 (19) : 片内振荡电路的输入端

XTAL2 (20) : 片内振荡电路的输出端

3. 控制引脚 (4 根)

RST (9) : 复位引脚, 引脚上出现 2 个机器周期的高电平将使单片机复位。

ALE/ $\overline{\text{PROG}}$ (30) : 地址锁存允许信号

$\overline{\text{PSEN}}$ (29) : 外部存储器读选通信号

$\overline{\text{EA/VPP}}$ (31) : 程序存储器的内外选通, 接低电平从外部程序存储器读指令, 如果接高电平则从内部程序存储器读指令。

4. 可编程输入 / 输出引脚 (32 根)

AT89S51 单片机有 4 组 8 位的可编程 I/O 口, 分别位 P0、P1、P2、P3 口, 每个口有 8 位 (8 根引脚), 共 32 根。每一根引脚都可以编程, 比如用来控制电机、交通灯、霓虹灯等, 开发产品时就是利用这些可编程引脚来实现我们想要的功能。

P0 口 (39~32) : 8 位双向 I/O 口线, 名称为 P0.0~P0.7

P1 口 (1~8) : 8 位准双向 I/O 口线, 名称为 P1.0~P1.7

P2 口 (21~28) : 8 位准双向 I/O 口线, 名称为 P2.0~P2.7

P3 口 (10~17) : 8 位准双向 I/O 口线, 名称为 P3.0~P3.7

(二) 单片机最小系统的组成:

单片机要正常运行, 必须具备一定的硬件条件, 其中最主要的就是三个基本条件:

①电源正常; ②时钟正常; ③复位正常。下面就以我们实验用的 51 单片机 AT89S51 为例介绍其硬件连接方法。在 AT89S51 单片机的 40 个引脚中: 电源引脚 2 根, 晶振引脚 2 根, 控制引脚 4 根, 可编程输入输出引脚 32 根。下面图 1.2 就是 AT89S51 最小化应用 (仅驱动 1 个发光二极管) 的接线方法:

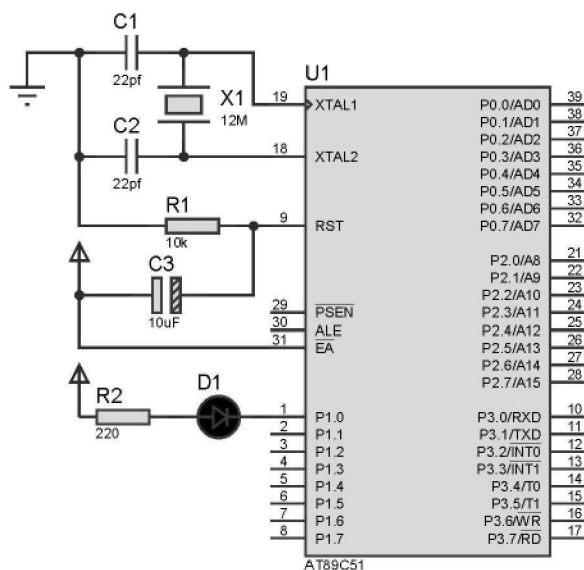


图 1.2 单片机最小系统

① 工作电源：电源是单片机工作的动力源泉，没有电源当然不能工作啦，对应的接线方法为：40 脚（VCC 电源引脚，工作时接 +5V 电源，20 脚（GND 为接地线。

② 时钟电路：时钟电路为单片机产生时序脉冲，单片机所有运算与控制过程都是在统一的时序脉冲的驱动下进行的，时钟电路就好比人的心脏，如果人的心跳停止了，人的生命就停止，同样，如果单片机的时钟电路停止工作（晶振停振），那么单片机也就停止运行了。

③ 复位电路：在复位引脚（9 脚）脚持续出现 24 个振荡器脉冲周期（即 2 个机器周期）的高电平信号将使单片机复位。如下图所示电容 C 和电阻 R 构成了单片机上电自动复位电路，复位后，单片机从 0000H 单元开始执行程序，并初始化一些专用寄存器为复位状态值，受影响的专用寄存器如下表所示：

表 1.1 单片机寄存器状态

寄存器	状态	寄存器	状态
PC	0000H	TCON	00H
ACC	00H	TL0	00H
PSW	00H	TH0	00H
SP	07H	TL1	00H
DPTR	0000H	TH1	00H
P0P3	FFH	SCON	00H
IP	xxx00000B	SBUF	不确定
IE	0xx00000B	PCON	0xxx0000B
TMOD	00H		

（4）控制引脚 EA 接法。 $\overline{\text{EA}}$ / VPP（31 脚）为内外程序存储器选择控制引脚，当 $\overline{\text{EA}}$ 为低电位时，单片机从外部程序存储器取指令；当 $\overline{\text{EA}}$ 接高电平时，单片机从内部程序存储器取指令……AT89S51 单片机内部有 4KB 可反复擦写 1000 次以上的程序存储器，因此我们把 $\overline{\text{EA}}$ 接到 +5V 高电平，让单片机运行内部的程序，我们就可以通过反复烧写来验证我们的程序了。

（5）接着我们给单片机加一个指示灯。我们在单片机 P1.7（8 脚）上接一个发光二极管，这样就可以用来做单片机驱动发光二极管了，图中发光二极管负极对单片机引脚 P1.7 之间串接了一个 560 欧姆的限流电阻，防止发光二极管和单片机的引脚 P1.7 因为电流过大烧坏，使发光二极管和单片机都工作在安全状态。

至此，我们完成了 AT89S51 单片机最小化系统的连接，只要把你编写的程序烧写到单片机内部，并接上 5V 电源就可以正常运行了，单片机剩下的其他可以不接线，就这样空着。赶快动手编一个驱动发光二极管闪烁的程序来验证一下吧。

（三）Keil 软件使用：

Keil 单片机集成开发软件是目前最流行的 MCS-51 单片机开发软件，Keil 提供了包括 C 编译器、宏汇编、连接器、库管理及一个功能强大的仿真调试器在内的完整开发方案，通过一个集成开发环境（uVision2）将这些部分组合在一起。Keil 单片机集成开发软件可以运行在 Win98、NT、Win2000、WinXP 等操作系统，掌握这一软件的使用对于使用 51 系列单片机的爱好者来说是十分必要的，如果你使用 C 语言编程，那么 Keil 是你的不二之选，即使不使用 C 语言而仅用汇编语言编程，其方便易用的集成环境、强大的软件仿真调试工具也会令你事半功倍。下面我们通过图解的方式来学习 Keil 软件的使用，学习如何输入源程序 → 新建工程 → 工程详细设置 → 源程序编译得到目标代码文件。

第一步：双击 Keil。uVision2 的桌面快捷方式（图 1.3），启动 Keil 集成开发开发软件。软件启动后的界面如图 2 所示。

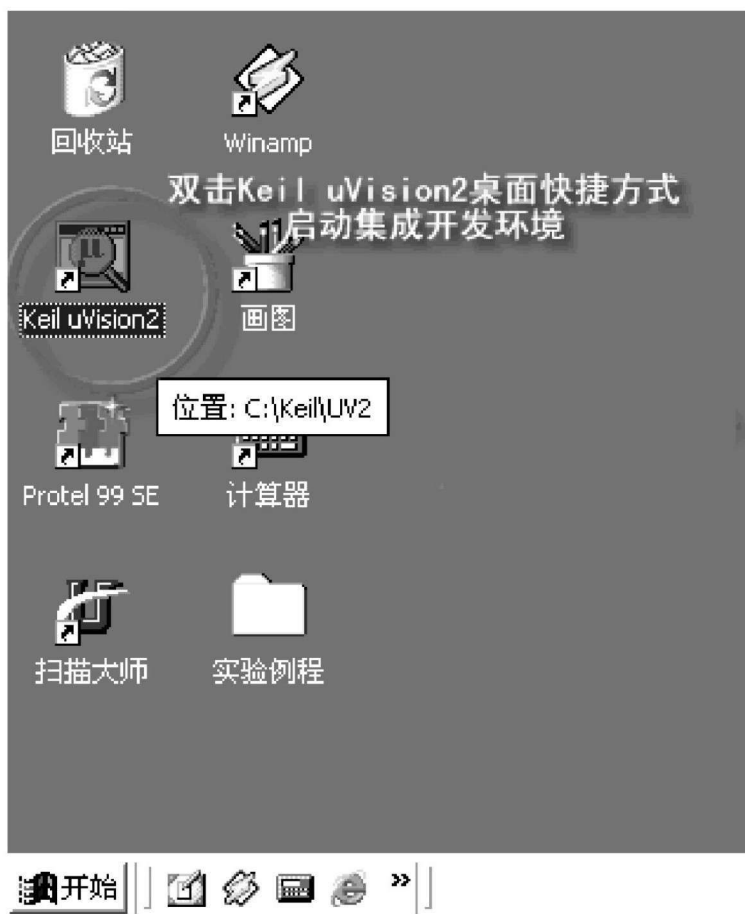


图 1.3：启动 Keil uVision2 单片机集成开发环境

第二步：新建文本编辑窗。点击工具栏上的新建文件快捷按键，即可在项目窗口的右侧打开一个新的文本编辑窗。

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，见图 1.4。

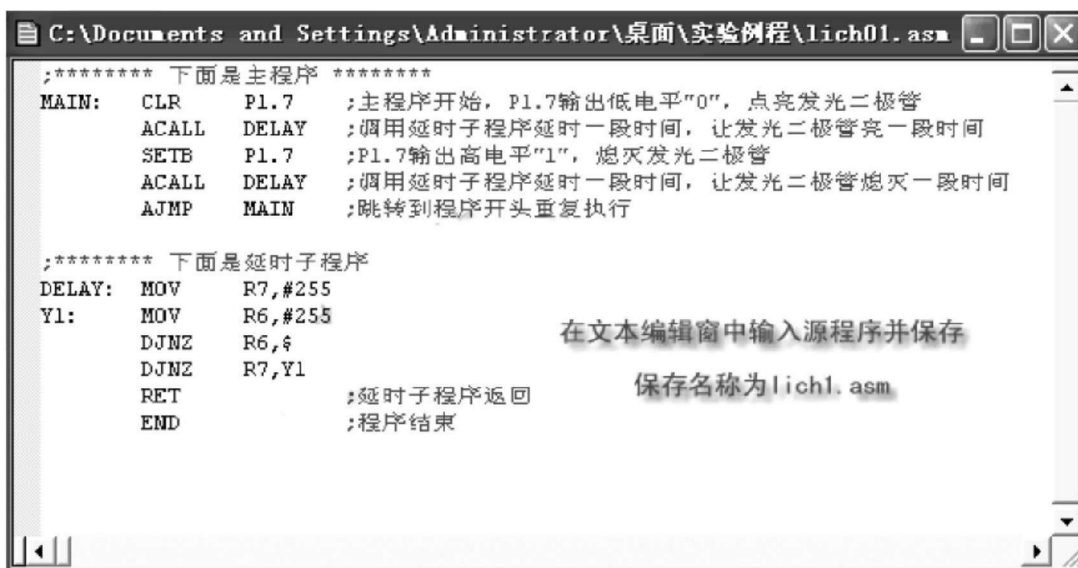


图 1.4 输入源程序

第四步：保存源程序。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”。

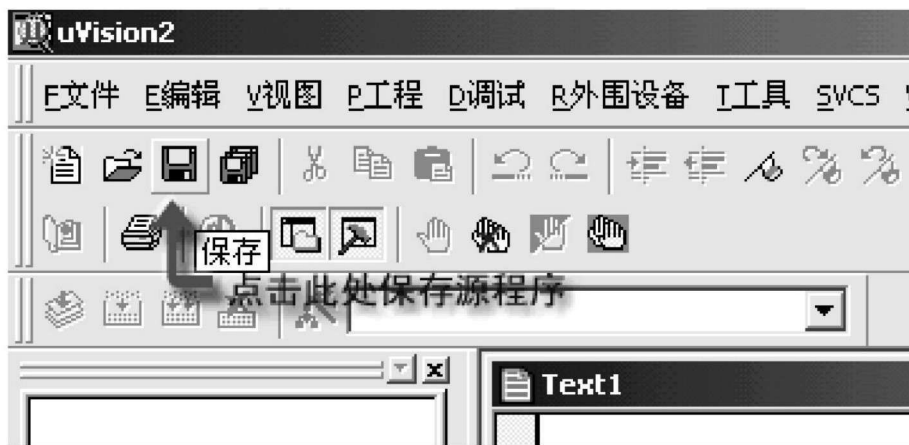


图 1.5: 保存源程序

注：第 3 步和第 4 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：新建立 Keil 工程。如图 1.6 所示，点击“工程”→“新建工程”命令，将出现保存对话框（图 1.7）。

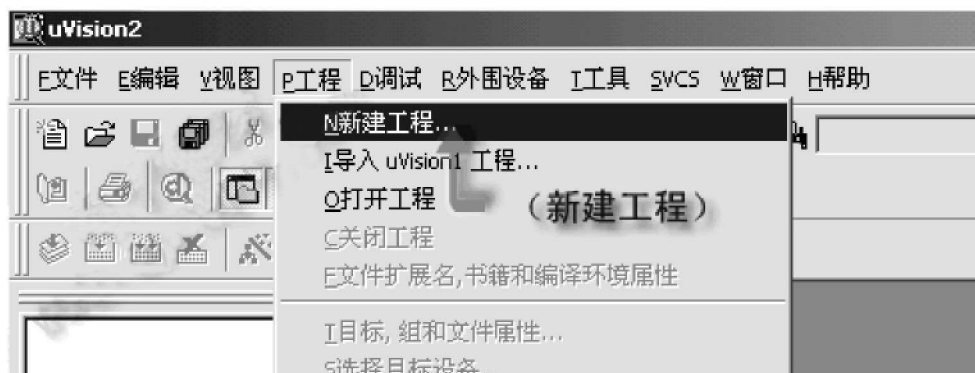


图 1.6: 新建工程



图 1.7: 保存工程对话框

在保存工程对话框中输入你的工程的文件名，Keil 工程默认扩展名为“.uv2”，工程名称不用输入扩展名（见图 1.9），一般情况下使工程文件名称和源文件名称相同即可，输入名称后保存，将出现“选择设备”对话框（图 1.8），在对话框中选择 CPU 的型号。

第六步：选择 CPU 型号。如图 1.8 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。

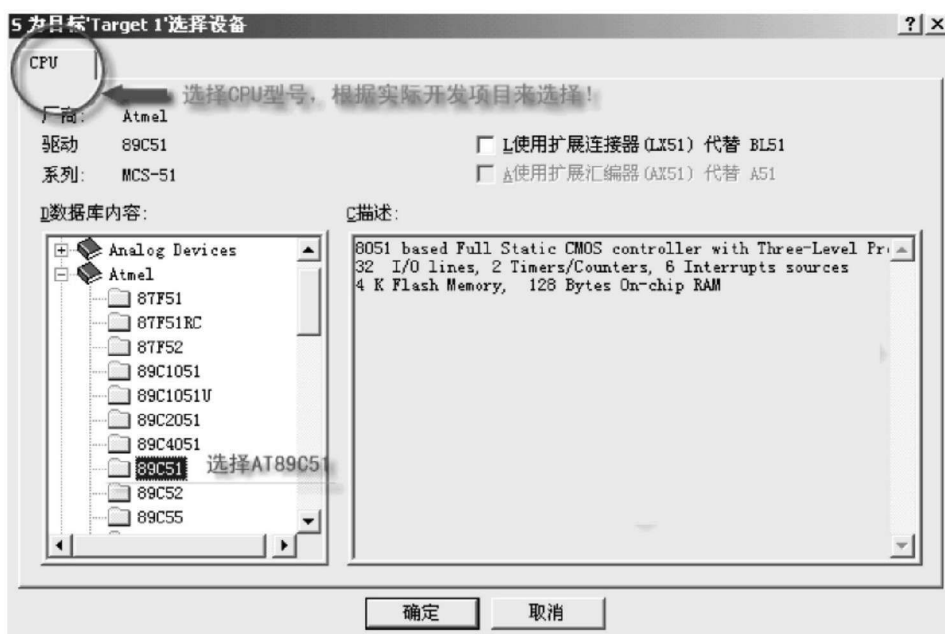


图 1.8: 选择 CPU 型号对话框

第七步：加入源程序到工程中。在选择好 CPU 型号后，点击“确定”按钮返回主界面，此时可见到工程管理窗中出现“Target 1”，点击“Target 1”前面的“+”号展开下一层的“Source Group 1”文件夹，此时的新工程是空的，“Source Group1”文件夹中什么文件都没有，必须把刚才输入的源程序加入到工程当中。如图 1.11 所示，右击工程管理窗中的“Source Group 1”，出现下拉菜单，点击“增加文件到组‘Source Group 1’”命令，将出现添加文件对话框（图 1.9）。

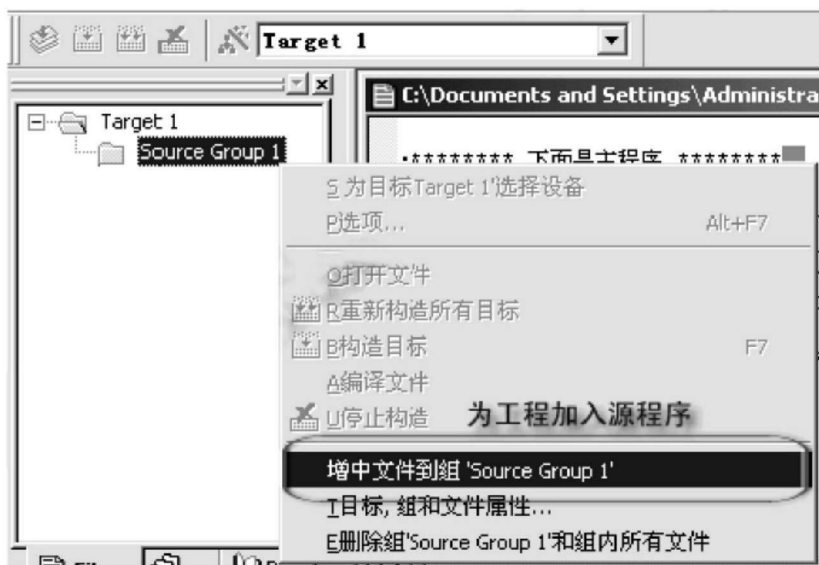


图 1.9: 添加文件到工程命令

在添加文件对话框（图 1.9）中，找到要添加到工程中的源程序文件。注意：在对话框中的文件类型默认为“C 源文件 (*.c)”，如果你要添加到工程中的是汇编语言程序，则在文件类型中必须选中“Asm 源文件 (*.asm; *.src)”，以 *.asm 为扩展名的汇编源程序才会出现在文件列表框中。

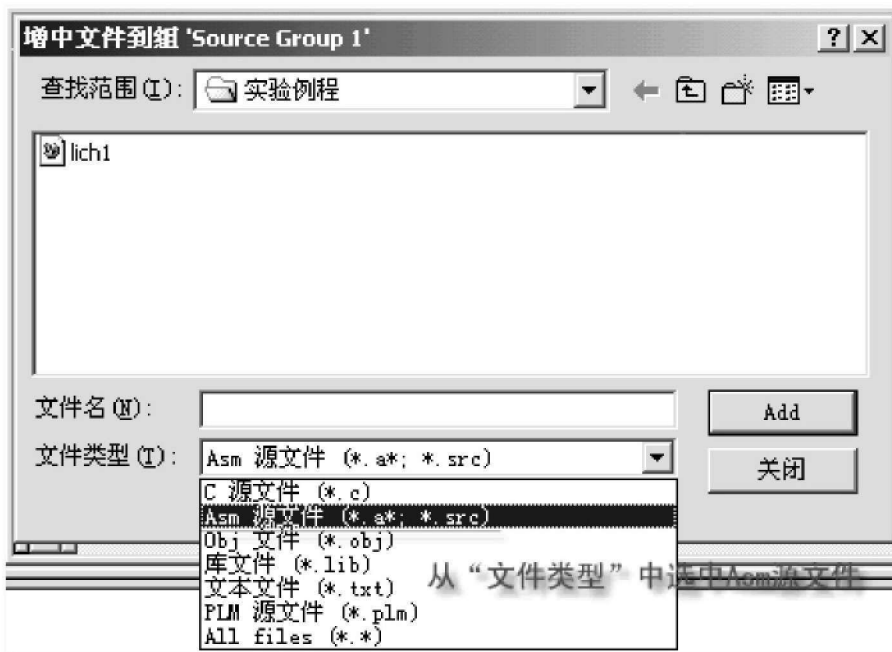


图 1.10: 浏览选择 ASM 源程序

双击该文件 lich1.asm，即可将该文件添加到工程当中，另外也可以单击 lich1.asm 选中该文件，再点击“Add”按钮，也可以把文件加入工程中（见图 1.11）。



图 1.11: 选中 ASM 源程序，加入到工程中

工程目标“Target 1”属性设置对话框（图 1.14）中有 8 个页面，设置的项目繁多复杂，大部分使用默认设置即可，我们主要设置其中的“目标”、“输出”、“调试”三个页面，下面对这三个页面的设置进行详细介绍。

① 工程目标属性设置。该页面单片机的晶振频率、存储器等，我们把晶振的频率改为 11.0592，频率设置和你实验板上的实际晶振频率相同即可（图 1.17）。



图 1.14 晶振频率设置

② 工程输出设置。该页面设置如下图 1.15 所示。注意：如果要进行单片机写片实验，则一定要把“E 生成 HEX 文件”选项选中，程序编译后才能生成我们写单片机需要的 HEX 格式目标文件。

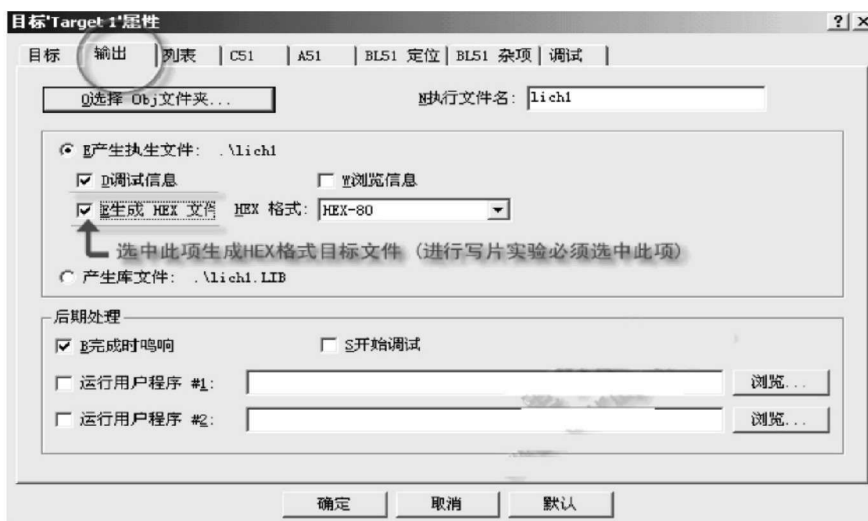


图 1.15: 工程输出设置