



高职高专教育“十二五”规划教材

# 单片机应用技术项目教程 (C语言版)

主 编 郭志勇  
副主编 王伟伟



中国水利水电出版社  
[www.waterpub.com.cn](http://www.waterpub.com.cn)

高职高专教育“十二五”规划教材

# 单片机应用技术项目教程

## (C语言版)

主 编 郭志勇

副主编 王伟伟



中国水利水电出版社  
www.waterpub.com.cn

## 内 容 提 要

本书由学校骨干教师、项目研发人员和高新企业的工程师共同编写。本书采用“项目驱动”的编写思路,从职业岗位技能出发,分为技能基础篇和技能提高篇,共有11个项目28个工作模块以及两个课程设计范例。

在技能基础篇中,注重职业岗位的基本技能训练,主要介绍单片机硬件系统、单片机开发系统、单片机并行端口应用、定时与中断系统、显示与键盘接口技术、A/D与D/A转换接口、串行接口通信技术以及单片机应用系统设计方法等内容;在技能提高篇中,注重职业岗位的开发技能训练,主要介绍键盘控制电机方向和转速、多路温度采集监控系统、按键设置液晶电子钟、16×32LED点阵显示设计与实现等单片机产品开发方法、关键知识以及这些项目的设计与实现,可以作为学生的课程设计。其中技能提高篇中的部分项目,可作为“中级嵌入式系统硬件设计工程师(单片机)”认证教材。两个课程设计范例分别是双向四车道交通灯和出租车计价器。

本书适用作为高职高专院校计算机应用技术、电子信息、机电等相关专业单片机技术课程的教材,也可作为广大电子制作爱好者的自学用书。

本书配有电子教学课件、习题参考答案、C语言源程序代码和Proteus仿真电路等。读者也可以通过该课程教学网站(<http://www1.avceit.cn/jisuanji/ktzi.html>)下载相关资源,还可以与作者联系(adzgyz@163.com),获取更多教学服务支持。

### 图书在版编目(CIP)数据

单片机应用技术项目教程: C语言版 / 郭志勇主编  
— 北京: 中国水利水电出版社, 2011.1  
高职高专教育“十二五”规划教材  
ISBN 978-7-5084-8051-0

I. ①单… II. ①郭… III. ①单片微型计算机—C语言—程序设计—高等学校: 技术学校—教材 IV. ①TP368.1②TP312

中国版本图书馆CIP数据核字(2010)第219616号

策划编辑: 雷顺加

责任编辑: 宋俊娥

封面设计: 李 佳

|      |   |
|------|---|
| 书 名  | 高职高专教育“十二五”规划教材<br>单片机应用技术项目教程(C语言版)  |
| 作 者  | 主 编 郭志勇<br>副主编 王伟伟  |
| 出版发行 | 中国水利水电出版社<br>(北京市海淀区玉渊潭南路1号D座 100038)<br>网址: <a href="http://www.waterpub.com.cn">www.waterpub.com.cn</a><br>E-mail: <a href="mailto:mchannel@263.net">mchannel@263.net</a> (万水)<br><a href="mailto:sales@waterpub.com.cn">sales@waterpub.com.cn</a><br>电话: (010) 68367658 (营销中心)、82562819 (万水) |
| 经 售  | 全国各地新华书店和相关出版物销售网点  |
| 排 版  | 北京万水电子信息有限公司  |
| 印 刷  | 北京市天竺颖华印刷厂  |
| 规 格  | 184mm×260mm 16开本 17.75印张 432千字  |
| 版 次  | 2011年1月第1版 2011年1月第1次印刷   |
| 印 数  | 0001—4000册  |
| 定 价  | 30.00元  |

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

## 高职高专教育“十二五”规划教材 编委会

主任委员 孙敬华 刘甫迎

副主任委员 刘晶璘 李 雪 胡学钢 丁亚明 孙 湧  
王路群 蒋川群 丁桂芝 宋汉珍 安志远

委 员 (按姓氏笔画排序)

|     |     |     |     |     |
|-----|-----|-----|-----|-----|
| 卜锡滨 | 方少卿 | 王伟伟 | 邓春红 | 冯 毅 |
| 刘 力 | 华文立 | 孙街亭 | 朱晓彦 | 余 东 |
| 吴 玉 | 吴 锐 | 吴昌雨 | 张兴元 | 张成叔 |
| 张振龙 | 李 胜 | 李 锐 | 李京文 | 李明才 |
| 李春杨 | 李家兵 | 杨圣春 | 杨克玉 | 苏传芳 |
| 金 艺 | 姚 成 | 官纪明 | 徐启明 | 郭 敏 |
| 钱 峰 | 钱 锋 | 高良诚 | 梁金柱 | 梅灿华 |
| 章炳林 | 黄存东 | 傅建民 | 喻 洁 | 程道凤 |

项目总策划 雷顺加

# 前 言

《单片机应用技术项目教程（C语言版）》顺应现代高等教育指导思想的变革，突出技能培养在课程中的主体地位，用工作任务来引领理论，使理论从属于技能实践。本书可作为高职高专院校计算机应用技术、电子信息、机电等相关专业单片机技术课程的教材，也可作为广大电子制作爱好者的自学用书。

本书主要特色如下：

## 1. 采用“项目驱动”的编写思路，突出技能培养在课程中的主体地位

本书以解决实际项目的思路和操作为编写主线，贯穿多个知识点，用工作模块来引领理论，使理论从属于技能培养。教会学生如何完成工作任务，关注学生能做什么，而不是知道什么，知识、技能的学习结合工作任务的完成过程来进行。

## 2. 与职业岗位标准接轨，已获得行业协会认可，作为**认证教材**，具有独特的“双证书”特色

本书内容与职业岗位标准密切接轨，部分项目由中国软件行业协会嵌入式系统分会的工程师编写，根据中级嵌入式系统硬件设计工程师（单片机）认证的要求，涉及必须掌握的关键知识点、基本技能以及单片机产品制作与调试等方面的内容，本书已获得中国软件行业协会嵌入式系统分会认可，作为“中级嵌入式系统硬件设计工程师（单片机）”认证教材。

学习本书后，由中国软件行业协会嵌入式系统分会进行考核（或经中国软件行业协会嵌入式系统分会认可，由学校进行考核）。考核合格的，可以申请中国软件行业协会嵌入式系统分会的“中级嵌入式系统硬件设计工程师（单片机）”证书。该证书可以作为专业技术人员职业能力考核的证明，以及岗位聘用的重要依据。

## 3. 既适合教学，又符合企业实际工作需要

注重采用企业真实工作任务、贴近企业职业岗位实际需求。本书在拉近单片机教学与职业岗位需求距离的同时，兼顾知识的系统性和完整性。本书既适合教学，又符合企业实际工作需要。

## 4. 全新的**仿真教学**模式，C语言编程

打破了传统教材原有界限，与职业岗位基本技能融合在一起，引入 Proteus 仿真软件，采用 C 语言编程，将学生从单片机复杂的硬件结构中解放出来，侧重高职院校学生技能和动手能力的锻炼，实现了在计算机上完成单片机电路设计、软件设计及调试与仿真。真正实现了从概念到产品的完整设计，使学生理解和掌握从概念到产品的完整过程。

## 5. **教学资源丰富**，提供教学支持及服务

课程教学网站提供支持，包括电子教案、实训项目、源代码和仿真电路、技能大赛作品、学生作品、课程设计、校企合作资源及相关其他素材等。

本书由学校骨干教师、项目研发人员和高新企业的工程师共同编写。采用“项目驱动”的编写思路，从职业岗位技能出发，分为技能基础篇和技能提高篇，共有 11 个项目 28 个工作模块以及两个课程设计范例。

在技能基础篇中，注重职业岗位的基本技能训练。主要介绍单片机硬件系统、单片机开发系统、单片机并行端口应用、定时与中断系统、显示与键盘接口技术、A/D 与 D/A 转换接口、串行接口通信技术以及单片机应用系统设计方法等内容；在技能提高篇中，注重职业岗位的开发技能训练，主要介绍键盘控制电机方向和转速、多路温度采集监控系统、按键设置液晶电子钟、16×32 LED 点阵显示设计与实现等单片机产品开发方法、关键知识以及这些项目的设计与实现，可以作为学生的课程设计。其中技能提高篇中的部分项目，可作为“中级嵌入式系统硬件设计工程师（单片机）”认证教材。两个课程设计范例分别是双向四车道交通灯和出租车计价器。

本书由郭志勇任主编，王伟伟任副主编。郭志勇对本书的编写思路与大纲进行了总体规划，指导全书的编写，承担全书的各个项目连贯性及统稿。项目一和项目六由郭志勇编写，项目二和项目十由王伟伟编写，项目三和项目八由巩雪洁编写，项目五和项目九由李健编写，项目七和课程设计范例一由李自成编写，项目四和课程设计范例二由程淑玉编写，项目十一由林艺春编写。参加本书电路调试、程序调试、素材收集、校对等工作的还有张长井、杨振宇、郭雨、王大灵、郭丽、赵露、蔡瑞瑞等。

为了方便教师教学，本书配有电子教学课件、习题参考答案、C 语言源程序代码和 Proteus 仿真电路。读者也可以通过该课程教学网站（<http://www1.avceit.cn/jisuanji/ktzi.html>）获得教材的技能训练、技能拓展、问题与讨论、所有源代码和仿真电路，以及学生的实训项目、课程设计项目作品和技能大赛作品。还可以与作者联系（[adzgzy@163.com](mailto:adzgzy@163.com)），获取更多教学服务支持。

本课程采用“教、学、做”一体化，参考学时如下表所示，读者可根据实际教学情况增减。

本课程参考学时

| 项目                 |                  | 参考学时 |
|--------------------|------------------|------|
| 技能基础篇<br>(共 64 学时) | 项目一 发光二极管 LED 控制 | 8    |
|                    | 项目二 LED 循环点亮控制   | 10   |
|                    | 项目三 数码管显示控制      | 10   |
|                    | 项目四 键盘的设计与实现     | 10   |
|                    | 项目五 定时/计数器综合应用   | 8    |
|                    | 项目六 模拟量输入输出设计与实现 | 8    |
|                    | 项目七 单片机串行通信设计与实现 | 10   |
| 技能提高篇<br>(任选)      | 项目八 键盘控制电机方向和转速  | 6    |
|                    | 项目九 按键设置液晶电子钟    | 8    |
|                    | 项目十 8 路温度采集监控系统  | 8    |
|                    | 项目十一 点阵显示设计与实现   | 12   |

由于时间紧迫和编者水平有限，书中难免会有错误和不妥之处，敬请广大读者和专家批评指正。

编者  
2010 年 10 月

# 目 录

前言

|   |    |                                   |    |
|---|----|-----------------------------------|----|
| 项目一 发光二极管 LED 控制 .....                        | 1  | 2.2.3 “头文件包含”处理 .....             | 38 |
| 1.1 工作模块 1 点亮一个 LED .....                     | 1  | 2.2.4 程序存储器 .....                 | 39 |
| 1.1.1 用 Proteus 设计第一个 LED 控制电路 ·1             |    | 2.3 工作模块 4 开关控制 LED 循环点亮 .....    | 41 |
| 1.1.2 用 Keil C51 设计第一个 C 语言<br>LED 控制程序 ..... | 5  | 2.3.1 开关控制 LED 循环点亮电路设计 .....     | 41 |
| 1.1.3 用 Proteus 仿真运行调试 .....                  | 8  | 2.3.2 开关控制 LED 循环点亮程序设计 .....     | 41 |
| 1.2 认识单片机 .....                               | 9  | 2.3.3 C51 数据类型 .....              | 42 |
| 1.2.1 单片机概述 .....                             | 9  | 2.3.4 C 语言常量与变量 .....             | 45 |
| 1.2.2 AT89S52 单片机结构 .....                     | 13 | 2.4 工作模块 5 步进电机控制 .....           | 47 |
| 1.2.3 AT89S52 单片机引脚功能 .....                   | 14 | 2.4.1 认识步进电机 .....                | 47 |
| 1.2.4 AT89S52 单片机最小系统 .....                   | 15 | 2.4.2 步进电机控制电路设计 .....            | 48 |
| 【技能训练 1-1】单片机最小系统应用——<br>开关控制 LED 点亮 .....    | 17 | 2.4.3 步进电机控制程序设计 .....            | 49 |
| 1.3 工作模块 2 LED 闪烁控制 .....                     | 18 | 2.5 技能拓展 ULN2003A 驱动器应用 .....     | 50 |
| 1.3.1 LED 闪烁控制设计与实现 .....                     | 18 | 2.5.1 ULN2003A 的特点 .....          | 50 |
| 1.3.2 C 语言程序的基本构成 .....                       | 20 | 2.5.2 ULN2003A 的引脚功能 .....        | 51 |
| 1.3.3 C 语言基本语句 .....                          | 22 | 【技能训练 2-2】单片机驱动继电器<br>电路设计 .....  | 51 |
| 1.4 技能拓展 音频控制应用 .....                         | 23 | 关键知识点小结 .....                     | 52 |
| 1.4.1 音频控制电路 .....                            | 23 | 问题与讨论 .....                       | 53 |
| 1.4.2 音频控制程序 .....                            | 24 | 项目三 数码管显示控制 .....                 | 54 |
| 【技能训练 1-2】报警产生器 .....                         | 24 | 3.1 工作模块 6 数码管循环显示 0~9 .....      | 54 |
| 关键知识点小结 .....                                 | 25 | 3.1.1 认识数码管 .....                 | 54 |
| 问题与讨论 .....                                   | 26 | 3.1.2 数码管循环显示 0~9 电路设计 .....      | 56 |
| 项目二 LED 循环点亮控制 .....                          | 27 | 3.1.3 数码管显示程序设计 .....             | 57 |
| 2.1 工作模块 3 LED 循环点亮控制 .....                   | 27 | 【技能训练 3-1】共阳极 LED 数码管应用 .....     | 58 |
| 2.1.1 LED 循环点亮电路设计 .....                      | 27 | 3.2 C 语言语句结构 .....                | 59 |
| 2.1.2 LED 循环点亮程序设计 .....                      | 28 | 3.2.1 关系运算符和关系表达式 .....           | 59 |
| 2.1.3 并行 I/O 端口电路 .....                       | 29 | 3.2.2 逻辑运算符和逻辑表达式 .....           | 59 |
| 【技能训练 2-1】P0 口外接上拉电阻 .....                    | 32 | 3.2.3 if 语句 .....                 | 60 |
| 2.2 MCS-51 单片机内存空间 .....                      | 33 | 3.2.4 switch 语句 .....             | 62 |
| 2.2.1 数据存储器 .....                             | 34 | 3.2.5 循环结构控制语句 .....              | 63 |
| 2.2.2 特殊功能寄存器 .....                           | 35 | 3.2.6 break 语句和 continue 语句 ..... | 65 |
|   |    | 3.3 工作模块 7 多个数码管动态扫描显示 .....      | 66 |

|                             |                         |    |                              |            |    |
|-----------------------------|-------------------------|----|------------------------------|------------|----|
| 3.3.1                       | 多个数码管动态显示电路设计           | 66 | 4.5.2                        | 中断加查询的扩展方法 | 98 |
| 3.3.2                       | 多个数码管动态显示程序设计           | 67 | 【技能训练 4-2】利用外部中断 0 扩展        |            |    |
| 3.3.3                       | C 语言数组                  | 68 | 4 个外部中断                      | 99         |    |
| 3.4                         | 工作模块 8 0~99 计数显示        |    | 关键知识点小结                      | 100        |    |
| (静态显示)                      |                         | 70 | 问题与讨论                        | 101        |    |
| 3.4.1                       | 0~99 计数显示电路设计           | 70 | 项目五 定时/计数器综合应用               | 102        |    |
| 3.4.2                       | 0~99 计数显示程序设计           | 71 | 5.1 工作模块 12 生产线自动打包系统        | 102        |    |
| 3.4.3                       | C 语言函数                  | 72 | 5.1.1 认识定时器/计数器              | 102        |    |
| 3.5                         | 技能拓展 7 段字型译码器 74LS47 应用 | 74 | 5.1.2 生产线自动打包控制电路            | 104        |    |
| 3.5.1                       | 认识 74LS47               | 74 | 5.1.3 生产线自动打包控制程序            | 105        |    |
| 3.5.2                       | 74LS47 引脚功能             | 75 | 5.2 工作模块 13 霓虹灯控制系统          | 106        |    |
| 【技能训练 3-2】使用 74LS47 实现 0~99 |                         |    | 5.2.1 定时器/计数器工作方式            | 106        |    |
| 计数显示                        | 75                      |    | 5.2.2 定时器/计数器有关寄存器           | 108        |    |
| 关键知识点小结                     | 76                      |    | 5.2.3 霓虹灯控制系统设计与实现           | 109        |    |
| 问题与讨论                       | 77                      |    | 【技能训练 5-1】0~59 秒数码显示         |            |    |
| 项目四 键盘的设计与实现                | 78                      |    | (定时器)                        | 111        |    |
| 4.1 工作模块 9 独立式键盘设计与实现       | 78                      |    | 5.3 工作模块 14 电子钟设计与实现         | 113        |    |
| 4.1.1 认识键盘                  | 78                      |    | 5.3.1 电子钟电路设计                | 113        |    |
| 4.1.2 独立式键盘电路设计             | 79                      |    | 5.3.2 电子钟程序设计                | 115        |    |
| 4.1.3 独立式键盘程序设计             | 79                      |    | 5.4 技能拓展 INTRINS.H 头文件的应用    | 116        |    |
| 4.1.4 键盘防抖动措施               | 81                      |    | 5.4.1 _crol_、_irol_和_lrol_函数 | 116        |    |
| 【技能训练 4-1】一键多功能按键识别         |                         |    | 5.4.2 _cror_、_iror_和_lror_函数 | 117        |    |
| 设计与实现                       | 82                      |    | 5.4.3 _nop_函数                | 117        |    |
| 4.2 工作模块 10 矩阵式键盘设计与实现      | 84                      |    | 5.4.4 _testbit_函数            | 117        |    |
| 4.2.1 矩阵式键盘的结构与原理           | 84                      |    | 【技能训练 5-2】用_crol_函数实现 LED    |            |    |
| 4.2.2 矩阵式键盘电路设计             | 85                      |    | 循环点亮                         | 118        |    |
| 4.2.3 矩阵式键盘程序设计             | 86                      |    | 关键知识点小结                      | 118        |    |
| 4.3 工作模块 11 中断方式矩阵键盘        | 88                      |    | 问题与讨论                        | 119        |    |
| 4.3.1 中断概念                  | 88                      |    | 项目六 模拟量输入输出设计与实现             | 120        |    |
| 4.3.2 中断方式矩阵键盘电路设计          | 89                      |    | 6.1 工作模块 15 模数转换 LED 显示      | 120        |    |
| 4.3.3 中断方式矩阵键盘程序设计          | 89                      |    | 6.1.1 模拟量输入输出概述              | 120        |    |
| 4.4 MCS-51 单片机中断系统          | 90                      |    | 6.1.2 模数转换 LED 显示电路          | 121        |    |
| 4.4.1 单片机中断源                | 90                      |    | 6.1.3 模数转换 LED 显示程序          | 121        |    |
| 4.4.2 中断系统结构                | 91                      |    | 6.2 ADC0808/ADC0809 模数转换器    | 122        |    |
| 4.4.3 中断有关的 4 个特殊功能寄存器      | 92                      |    | 6.2.1 认识 ADC0808/ADC0809     |            |    |
| 4.4.4 中断处理过程                | 94                      |    | 模数转换器                        | 122        |    |
| 4.4.5 C 语言中断服务函数            | 97                      |    | 6.2.2 ADC0808/ADC0809 结构及引脚  | 123        |    |
| 4.5 技能拓展 外部中断源的扩展           | 98                      |    | 6.2.3 ADC0808/ADC0809 工作过程及  |            |    |
| 4.5.1 用定时器扩展外部中断源           | 98                      |    | 编程方法                         | 125        |    |



|   |     |                                 |     |
|---|-----|---------------------------------|-----|
| 【技能训练 6-1】采用中断方式完成模数转换 LED 显示 .....       | 125 | 关键知识点小结 .....                   | 163 |
| 6.3 工作模块 16 数字电压表设计与实现 .....              | 127 | 问题与讨论 .....                     | 164 |
| 6.3.1 数字电压表电路设计 .....                     | 127 | 项目八 键盘控制电机方向和转速 .....           | 166 |
| 6.3.2 数字电压表程序设计 .....                     | 128 | 8.1 单片机产品开发 .....               | 166 |
| 6.3.3 C 语言结构体类型 .....                     | 129 | 8.1.1 单片机的结构 .....              | 166 |
| 6.3.4 结构体数组 .....                         | 131 | 8.1.2 单片机产品开发流程 .....           | 167 |
| 【技能训练 6-2】结构体数组的应用 .....                  | 132 | 8.2 工作模块 21 步进电机控制系统设计与实现 ..... | 169 |
| 6.4 工作模块 17 信号发生器设计与实现 .....              | 132 | 8.2.1 步进电机控制技术 .....            | 170 |
| 6.4.1 认识 DAC0832 数模转换器 .....              | 133 | 8.2.2 步进电机控制系统电路设计 .....        | 170 |
| 6.4.2 信号发生器电路设计 .....                     | 135 | 8.2.3 步进电机控制系统程序设计 .....        | 171 |
| 6.4.3 信号发生器程序设计 .....                     | 136 | 8.3 工作模块 22 直流电机控制系统设计与实现 ..... | 174 |
| 6.5 技能拓展 直流电机转速控制 .....                   | 137 | 8.3.1 直流电机控制技术 .....            | 174 |
| 6.5.1 什么是 PWM .....                       | 138 | 8.3.2 直流电机控制系统电路设计 .....        | 175 |
| 6.5.2 占空比 .....                           | 138 | 8.3.3 直流电机控制系统程序设计 .....        | 176 |
| 6.5.3 直流电机转速控制 .....                      | 138 | 8.4 技能拓展 步进电机智能控制 .....         | 179 |
| 【技能训练 6-3】基于 ADC0808/0809 的直流电机转速控制 ..... | 138 | 8.4.1 步进电机智能控制思路 .....          | 179 |
| 关键知识点小结 .....                             | 139 | 8.4.2 步进电机智能控制组成 .....          | 180 |
| 问题与讨论 .....                               | 140 | 【技能训练 8-1】步进电机智能控制设计 .....      | 180 |
| 项目七 单片机串行通信设计与实现 .....                    | 142 | 关键知识点小结 .....                   | 181 |
| 7.1 工作模块 18 单片机点对点数据传输 .....              | 142 | 问题与讨论 .....                     | 182 |
| 7.1.1 串行通信基本知识 .....                      | 142 | 项目九 按键设置液晶电子钟 .....             | 183 |
| 7.1.2 单片机点对点数据传输电路设计 .....                | 145 | 9.1 RT12864 阵型液晶显示模块 .....      | 183 |
| 7.1.3 单片机扩展 I/O 口程序设计 .....               | 146 | 9.1.1 认识 RT12864 液晶屏 .....      | 183 |
| 7.2 工作模块 19 水塔水位单片机远程监控系统 .....           | 147 | 9.1.2 RT12864 液晶显示模块内部结构 .....  | 185 |
| 7.2.1 单片机串行口结构 .....                      | 147 | 9.1.3 控制指令及相应代码 .....           | 186 |
| 7.2.2 串行通信设置 .....                        | 149 | 9.2 工作模块 23 液晶电子钟电路设计与实现 .....  | 188 |
| 7.2.3 水塔水位单片机远程监控电路设计 .....               | 151 | 9.2.1 按键设置电路设计 .....            | 189 |
| 7.3 工作模块 20 单片机点对多数据传输 .....              | 155 | 9.2.2 液晶显示电路设计 .....            | 189 |
| 7.3.1 MCS-51 多机通信技术 .....                 | 155 | 9.2.3 按键设置液晶电子钟电路设计 .....       | 189 |
| 7.3.2 单片机点对多数据传输电路设计 .....                | 156 | 9.3 工作模块 24 液晶电子钟程序设计 .....     | 191 |
| 7.3.3 单片机点对多数据传输程序设计 .....                | 157 | 9.3.1 按键设置程序设计 .....            | 191 |
| 【技能训练 7-1】单片机串行口扩展 I/O 口电路设计 .....        | 160 | 9.3.2 时间程序设计 .....              | 194 |
| 7.4 技能拓展 RS-485 串行接口应用 .....              | 162 | 9.3.3 液晶显示程序设计 .....            | 195 |
| 【技能训练 7-2】全双工 RS-485 连接电路设计与实现 .....      | 163 | 9.4 技能拓展 DS1302 时钟芯片的应用 .....   | 201 |
|   |     | 9.4.1 认识 DS1302 .....           | 201 |

|   |     |   |     |
|---|-----|---|-----|
| 9.4.2 DS1302 寄存器 .....                      | 201 | 项目十一 点阵显示设计与实现 .....                      | 224 |
| 【技能训练 9-1】使用 DS1302 芯片实现液晶<br>电子钟电路设计 ..... | 202 | 11.1 工作模块 27 16×32 LED 点阵汉字<br>显示设计 ..... | 224 |
| 关键知识点小结 .....                               | 203 | 11.1.1 LED 点阵显示系统结构与原理 .....              | 224 |
| 问题与讨论 .....                                 | 204 | 11.1.2 16×32 LED 点阵汉字显示电路<br>设计 .....     | 226 |
| 项目十 8 路温度采集监控系统 .....                       | 205 | 11.1.3 16×32 LED 点阵汉字显示程序<br>设计 .....     | 231 |
| 10.1 DS18B20 温度传感器 .....                    | 205 | 11.2 工作模块 28 点阵屏焊接与调试 .....               | 243 |
| 10.1.1 认识 DS18B20 .....                     | 205 | 11.2.1 MCU 最小系统和串口焊接与调试 .....             | 244 |
| 10.1.2 DS18B20 内部结构及功能 .....                | 207 | 11.2.2 显示功能板焊接和调试 .....                   | 248 |
| 10.1.3 DS18B20 通信协议 .....                   | 209 | 11.2.3 外扩 RAM 及 Flash 的焊接和调试 .....        | 249 |
| 10.2 工作模块 25 8 路温度采集监控<br>电路设计 .....        | 212 | 11.2.4 外扩 KEY 和 EEPROM 焊接与<br>调试 .....    | 249 |
| 10.2.1 温度采集电路设计 .....                       | 212 | 11.2.5 系统整体测试 .....                       | 249 |
| 10.2.2 监控电路设计 .....                         | 212 | 11.3 技能拓展 并行 I/O 接口扩展 .....               | 250 |
| 10.2.3 8 路温度采集监控电路设计与实现 .....               | 213 | 11.3.1 并行 I/O 口扩展特性 .....                 | 250 |
| 10.3 工作模块 26 8 路温度采集监控<br>程序设计 .....        | 214 | 11.3.2 I/O 接口扩展常用芯片 .....                 | 250 |
| 10.3.1 键盘处理程序设计 .....                       | 214 | 【技能训练 11-1】P0 口扩展 I/O 接口设计<br>与实现 .....   | 252 |
| 10.3.2 8 路温度采集程序设计 .....                    | 216 | 关键知识点小结 .....                             | 253 |
| 10.3.3 8 路温度显示程序设计 .....                    | 217 | 问题与讨论 .....                               | 254 |
| 10.4 技能拓展 I <sup>2</sup> C 串行扩展 .....       | 219 | 附表 1 LED 各模块焊接步骤对应元件清单 .....              | 255 |
| 10.4.1 I <sup>2</sup> C 总线接口电路 .....        | 219 | 附表 2 LED 显示模块焊接元件清单 .....                 | 256 |
| 10.4.2 I <sup>2</sup> C 总线数据传送操作 .....      | 219 | 课程设计范例一 双向四车道交通灯控制 .....                  | 257 |
| 【技能训练 10-1】基于 AT24C1024 片外程序<br>存储器扩展 ..... | 221 | 课程设计范例二 出租车计价器 .....                      | 266 |
| 关键知识点小结 .....                               | 222 | 参考文献 .....                                | 272 |
| 问题与讨论 .....                                 | 223 |   |     |

# 项目一 发光二极管 LED 控制



## 终极目标

能完成单片机最小系统和输出电路设计,能应用 C 语言程序完成单片机输入输出控制,实现对 LED 控制的设计、运行及调试。

## 促成目标

1. 了解 AT89S52 单片机结构;
2. 掌握 AT89S52 单片机的引脚功能;
3. 掌握 AT89S52 单片机最小系统电路设计;
4. 掌握 C 语言基本构成和基本语句;
5. 会利用单片机 I/O 口实现点亮一个 LED 和控制 LED 闪烁。

## 1.1 工作模块 1 点亮一个 LED



使用 AT89S52 单片机, P1.0 引脚接发光二极管(LED)的阴极,通过 C 语言程序控制,从 P1.0 引脚输出低电平,使发光二极管点亮。

### 1.1.1 用 Proteus 设计第一个 LED 控制电路

Proteus 是英国 Labcenter Electronics 公司开发的多功能 EDA 软件。Proteus 不仅是模拟电路、数字电路、模/数混合电路的设计与仿真平台,也是目前较先进的单片机和嵌入式系统的设计与仿真平台。它实现了在计算机上完成从原理图与电路设计、电路分析与仿真、单片机代码级调试与仿真、系统测试与功能验证到形成 PCB 的完整的电子设计、研发过程。

按照工作任务要求,点亮一个 LED 电路由 AT89S52 单片机和一个 LED 电路构成。AT89S52 单片机是美国 ATMEL 公司生产的低电压、高性能 8 位单片机,具有丰富的内部资源,使用 AT89S52 单片机无须外部存储器。

LED 加正向电压发光,反之不发光。一般接法是阳极接高电平,阴极管单片机的某一输出口线,当该输出口线为低电平时,LED 亮,该输出口线为高电平时,LED 不亮。这样只要

编程控制单片机的该输出口, 就可控制指示灯亮或灭。

在本工作模块中, LED 的阳极通过  $220\Omega$  限流电阻后连接到 5V 电源上, 限流电阻在这里起到了限流的作用, 使通过 LED 的电流被限制在十几毫安左右。P1.0 引脚接 LED 的阴极, P1.0 引脚输出低电平时对应的 LED 点亮, 输出高电平时对应的 LED 熄灭。LED 点亮电路设计如图 1-1 所示。

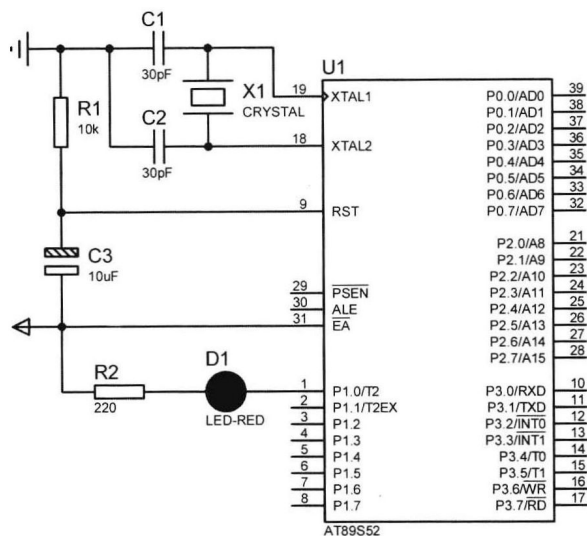


图 1-1 LED 点亮电路

本书使用 Proteus 7.5SP3 Professional 中文版。双击桌面上的 ISIS 7 Professional 图标或者单击屏幕左下方的“开始”→“程序”→Proteus 7 Professional→ISIS 7 Professional, 进入 Proteus ISIS 集成环境, 如图 1-2 所示。

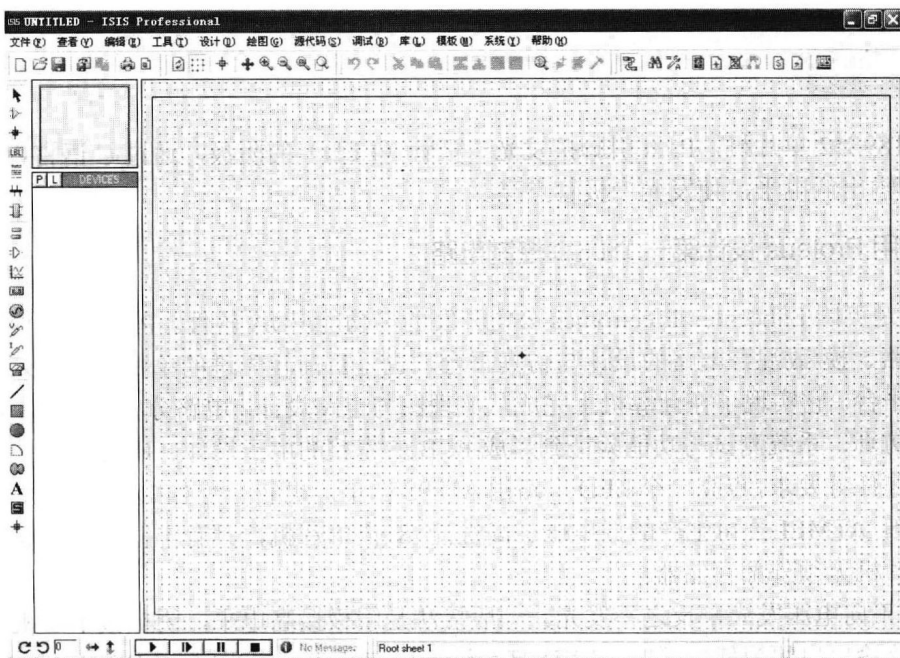


图 1-2 ISIS 集成环境

(1) 新建设计文件。单击“文件”→“新建设计”命令，在弹出的“新建设计”对话框中选择 DEFAULT 模板后单击“确定”按钮，如图 1-3 所示。

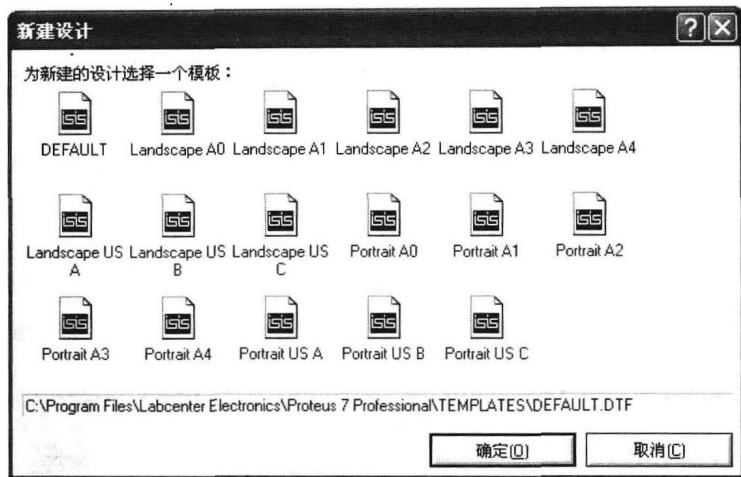


图 1-3 “新建设计”对话框

(2) 设置图纸尺寸。单击“系统”→“设置图纸大小”命令，在弹出的 Sheet Size Configuration 对话框中选择 A4 图纸尺寸或自定义尺寸后单击“确定”按钮。

(3) 设置网格。单击“查看”→“网格”命令，显示网格（再次单击，不显示网格）。单击“查看”→Snap xxth（或 Snap x.xin），可改变网格单位，默认为 Snap 0.1in。

(4) 保存设计文件。单击“文件”→“保存设计”命令，在弹出的“保存 ISIS 设计文件”对话框中指定文件夹，输入文件名“点亮一个 LED”，并选择保存类型为“设计文件 (\*.DSN)”单击“保存”按钮。

(5) 选取元器件。从 Proteus 元器件库中选取元器件 AT89C52（单片机），AT89S52 也可用 AT89C52 代替。单击模式选择工具栏的“元件”按钮，单击“器件选择”按钮，在弹出的 Pick Devices（选取元器件）对话框的“关键字”栏中输入元器件名称 AT89C52（也可以是分类、小类、属性值），与关键字匹配的元器件 AT89C52 显示在元器件列表（结果）中。双击选中的元器件 AT89C52，便将所选元器件 AT89C52 加入到对象选择器窗口，单击“确定”按钮完成元器件选取，如图 1-4 所示。

(6) 用同样方法选取其他元器件。从 Proteus 元器件库中选取元器件 CRYSTAL（晶振）、CAP（电容）、CAP-ELEC（电解电容）、RES（电阻）、LED-RED（红色发光二极管）。

(7) 放置元器件。单击对象选择器窗口的元器件 AT89C52，元器件名 AT89C52 变为蓝底白字，预览窗口显示 AT89C52 元器件；单击方向工具栏按钮可实现元器件的左旋、右旋、水平和垂直翻转，以调整元器件的摆放方向；将鼠标指针移到编辑区某一位置，单击一次就可放置元器件 AT89C52。用同样方法放置其他元器件。按图 1-1 所示放置元器件。

(8) 编辑元器件。单击模式选择工具栏“编辑”按钮，进入编辑状态。右击（或单击）元器件，该元器件变为红色表明被选中，鼠标指针放到被选中的元器件上，按住左键拖动，将鼠标移到编辑区某一位置松开，即完成元器件的移动。鼠标指针放到被选中的元器件上右击，单击弹出的快捷菜单中的方向工具栏按钮可实现元器件的旋转和翻转。右击被选中的元器件，可删除该元器件。在被选中的元器件外单击，可撤销选中。按图 1-1 所示编辑元器件。

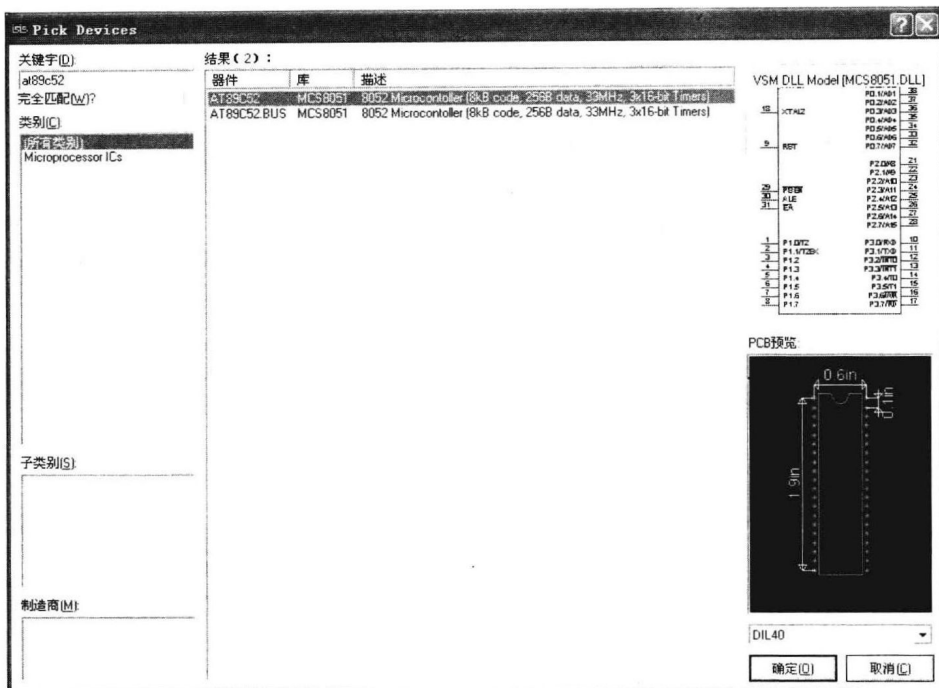





图 1-4 Pick Devices 对话框

(9) 放置终端。单击模式选择工具栏“终端”按钮，单击对象选择器窗口的电源终端 POWER，该终端名背景变为蓝色，预览窗口显示该终端；单击方向工具栏“左旋转”按钮，电源终端逆时针旋转 90°；将鼠标指针移到编辑区某一位置，单击一次就可放置一个终端。用同样方法放置接地终端 GROUND。

(10) 连线。单击命令工具栏“实时 Snap(捕捉)”按钮，实时捕捉有效（再次单击，实时捕捉无效），当鼠标指针接近引脚末端时，该处会自动出现一个小方框“□”，表明可以自动连接到该点。按图 1-1 所示单击要连线的元器件起点和终点，完成连线。

(11) 属性设置。先右击后单击元器件电容 C1，弹出“编辑元件”对话框，如图 1-5 所示。将电容量改为 30pF，单击“确定”按钮完成元器件电容 C1 的属性编辑。用同样方法编辑其他元器件属性。

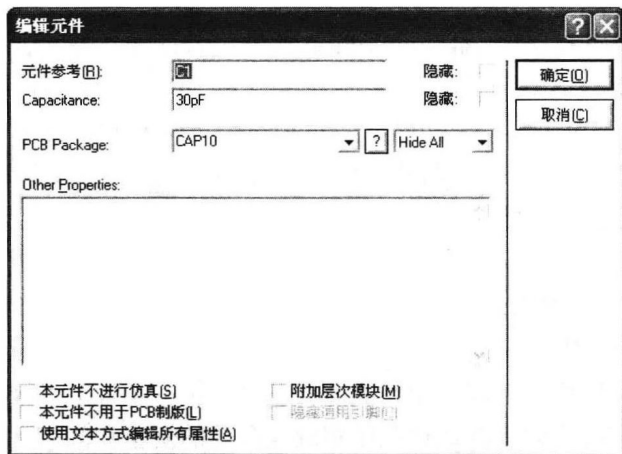


图 1-5 “编辑元件”对话框

(12) 电气规则检测。单击“工具”→“电气规则检查”命令，弹出检查结果窗口，完成电气检测。若检测出错，根据提示修改电路图并保存，直至检测成功。电气规则检查窗口如图 1-6 所示。

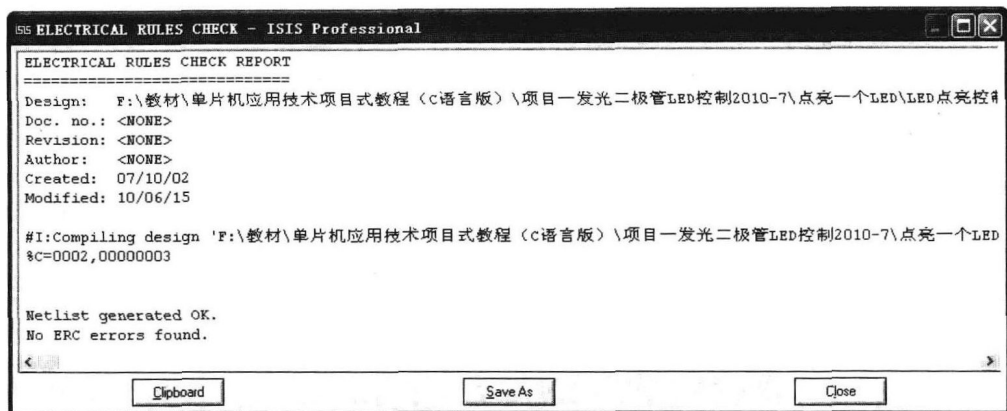


图 1-6 电气规则检查窗口

### 1.1.2 用 Keil C51 设计第一个 C 语言 LED 控制程序

Keil C51 是德国 Keil 软件公司开发的基于 8051 内核的微控制器软件开发平台，是 51 系列单片机 C 语言软件开发系统，是目前开发 8051 内核单片机的主流工具。Keil C51 软件提供了丰富的库函数和功能强大的集成开发调试工具，全 Windows 界面。μVision2 集成开发环境可以完成工程建立和管理、编译、连接、目标代码的生成、软件仿真和硬件仿真等完整的开发流程。

#### 1. 编写点亮一个 LED 程序

由于 P1.0 引脚接 LED 的阴极，LED 的阳极通过 220Ω 限流电阻后连接到 5V 电源上，所以从 P1.0 引脚输出低电平就可以点亮 LED。“点亮一个 LED”的 C 语言程序如下：

```
#include <AT89X52.H>    //包含 AT89X52.H 头文件
sbit LED=P1^0;         //定义 LED 为 P1.0 引脚
void main (void)
{
    LED=0;              //P1.0 引脚输出低电平点亮 LED
    while(1);
}
```

程序编程说明：

(1) “#include <AT89X52.H>”语句是一个“文件包含”处理，是将 AT89X52.H 头文件的内容全部包含进来。这里程序中包含 AT89X52.H 头文件的目的是为了使用 P1^0 这个符号，即通知 C 编译器，程序中所写的 P1^0 是指 AT89S52 单片机的 P1.0 引脚。

(2) P1.0 不能直接使用，这里用“sbit LED=P1^0;”就是定义用符号 LED 来表示 P1.0 引脚，也可以起 P1\_0 或 P10 一类的名字。

(3) “LED=0;”语句是使 P1.0 引脚输出低电平，点亮发光二极管 LED。

(4) “while(1);”语句的表达式是 1，也就是说 while 语句的表达式始终为真，进入死循环，LED 始终点亮。

(5) Keil C 支持 C++ 风格的注释，可以用“//”进行注释，也可以用 /\*.....\*/ 进行注释。

## 2. 建立第一个 C 程序项目

双击桌面上的 Keil  $\mu$ Vision2 图标或者单击屏幕左下方的“开始”→“程序”→Keil  $\mu$ Vision2，进入 Keil  $\mu$ Vision2 集成开发环境，如图 1-7 所示。

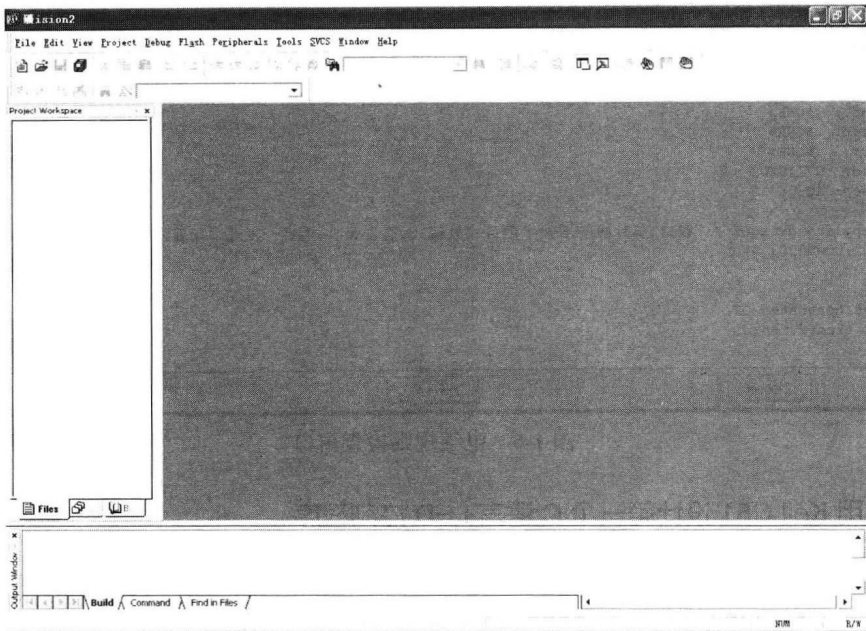


图 1-7 Keil  $\mu$ Vision2 集成开发环境

(1) 建立工程文件，选择单片机。单击“工程”→“新建工程”命令，在弹出的“新建工程”对话框中指定文件夹，输入文件名“点亮一个 LED”，单击“保存”按钮，在弹出的为目标 Target 1 选择设备对话框中选择单片机型号（Atmel 的 89S52），单击“确定”按钮。选择单片机对话框如图 1-8 所示。

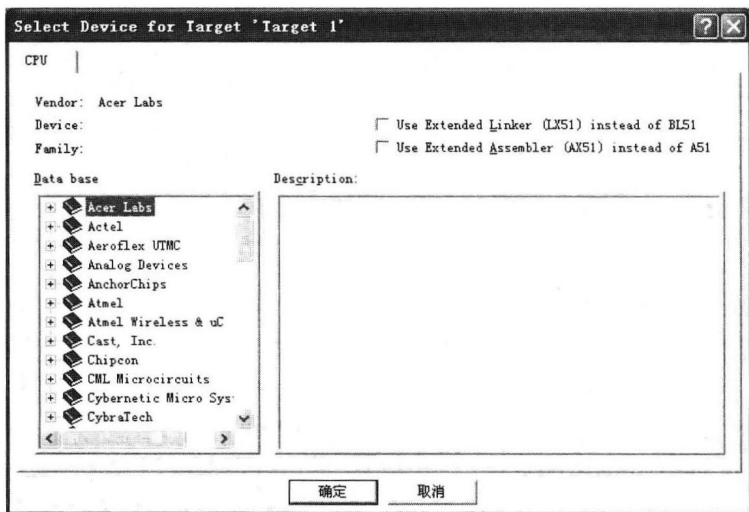


图 1-8 选择单片机对话框

(2) 建立源文件并加载。单击“文件”→“新建”命令，在文件编辑窗口输入“点亮一个 LED”的 C 语言源程序，单击“文件”→“保存”命令，在弹出的“另存”对话框中指



定文件夹（一般与工程文件放在同一文件夹中），输入文件名“点亮一个 LED.c”（c 为 C 语言源程序的后缀），单击“保存”按钮，完成源文件的建立。

在工程窗口中右击 Target 1 文件夹下的 Source Group 1 文件夹后，单击弹出菜单的 Add Files to Group ‘Source Group 1’，在弹出的 Add Files to Group ‘Source Group 1’ 对话框中将文件类型设为 C Source file (\*.c)，单击刚才保存的源文件名“点亮一个 LED.c”，单击 Add 按钮再单击“关闭”按钮，完成源文件加载。源文件加载对话框如图 1-9 所示。

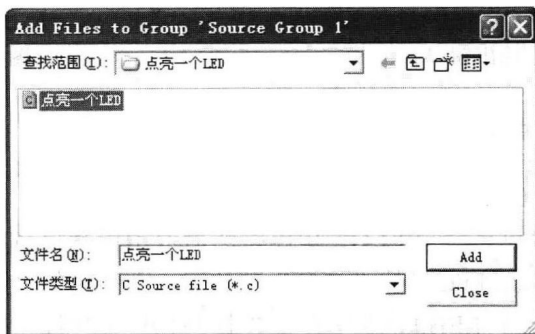


图 1-9 源文件加载对话框

(3) 设置工程的配置参数。在工程窗口中右击 Target 1 文件夹，单击弹出快捷菜单中的 Options for Target ‘Target 1’，参数设置窗口如图 1-10 所示。

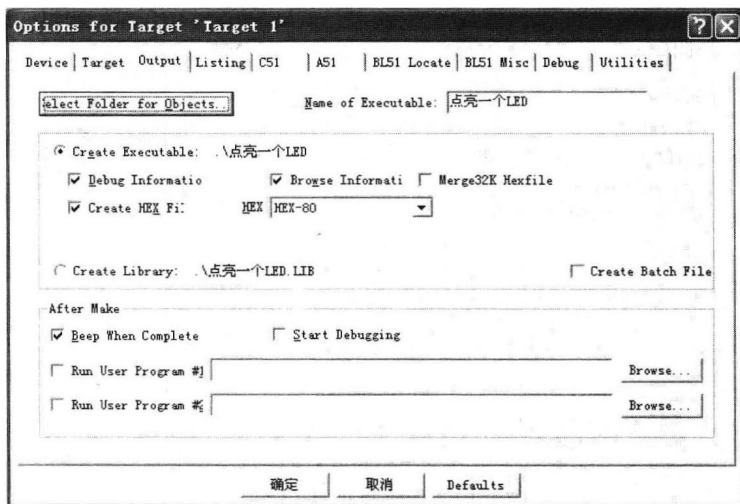


图 1-10 参数设置窗口

在弹出的 Options for Target ‘Target 1’ 对话框中做以下设置：Target 标签页的晶振频率栏设为 12MHz，Output 标签页选中 Create Hex Files 复选框，其余采用默认设置。单击“确定”按钮，完成配置参数设置。

(4) 进行编译和连接。单击“工程”→“构造目标”，完成编译，生成名为“点亮一个 LED.hex”的 HEX 文件。通过输出窗口查看编译信息，若提示出错，双击输出窗口的出错信息行，文件编辑窗口中出错指令所在行左侧会有箭头提示，逐个排除错误后重新编译。输出窗口如图 1-11 所示。

(5) 进入调试模式，打开 P1 口对话框。在调试模式中，单击“外围设备”→I/O-Ports →Port 1，打开 P1 口对话框。