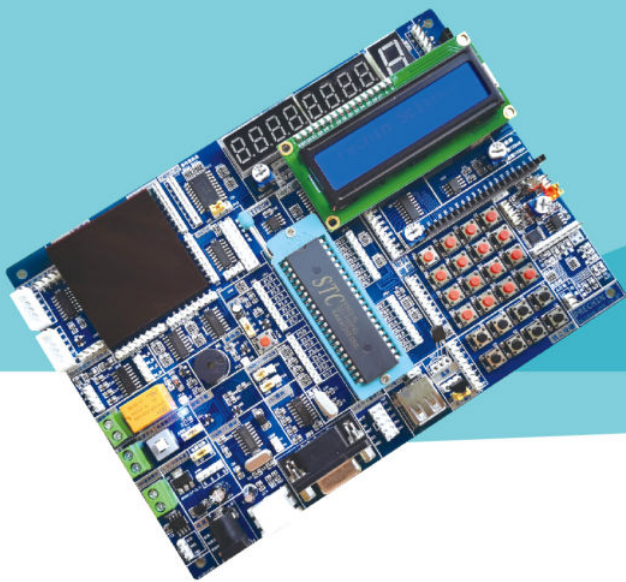


杨春丽◎著

单片机原理与应用

实验指导

(C语言版)



云南大学出版社
YUNNAN UNIVERSITY PRESS

单片机原理与应用

实验指导

(C语言版)

杨春丽◎著



云南大学出版社
YUNNAN UNIVERSITY PRESS

图书在版编目(CIP)数据

单片机原理与应用实验指导: C语言版 / 杨春丽著

— 昆明: 云南大学出版社, 2020

ISBN 978-7-5482-4020-4

I. ①单… II. ①杨… III. ①单片微型计算机—C语言—程序设计—高等学校—教学参考资料 IV.

①TP368.1②TP312.8

中国版本图书馆CIP数据核字(2020)第099655号

策划编辑: 赵红梅

责任编辑: 石可

封面设计: 王嫣一

DANPIANJI YUANLI YU YINGYONG SHIYAN ZHIDAO (C YUYANBAN)

单片机原理与应用

实验指导

(C语言版)

杨春丽◎著

出版发行: 云南大学出版社

印 装: 云南金伦云印实业股份有限公司

开 本: 787mm×1092mm 1/16

印 张: 9.25

字 数: 220千

版 次: 2020年8月第1版

印 次: 2020年8月第1次印刷

书 号: ISBN 978-7-5482-4020-4

定 价: 28.00元

社 址: 昆明市一二一大街182号(云南大学东陆校区英华园内)

邮 编: 650091

发行电话: 0871-65033244 65031071

网 址: <http://www.ynup.com>

E-mail: market@ynup.com

若发现本书有印装质量问题, 请与印厂联系调换, 联系电话: 0871-64120080。

前 言

单片机技术在家用电器、汽车电子、智能仪器、工业控制仪表等各个领域都有极为广泛的应用。单片机技术是电子工程师必须掌握的一门技术。近年来，作者一直致力于单片机课程的教学改革，通过精简和优化教学内容，在单片机课程教学中引入项目驱动教学法，构建理论与实践一体化的教学模式。该模式从学生实际出发，充分利用理论、实验、开放实验等有限的学时，注重单片机课程的实践性，建立了典型项目库，通过充分的实践引导学生循序渐进地掌握单片机的软硬件资源及工作原理。

在总结教学改革成效的基础上，作者根据课程教学要求，以提高学生的实践动手能力和工程设计能力为目的，从应用的角度出发，采用 C 语言编程编写了本书。本书包括 3 篇，依次为“准备篇”“基础篇”和“提升篇”。

第 1 篇为准备篇，主要讲解单片机应用系统的开发流程和 51 单片机的主要开发工具及其使用方法，包括 ARM 公司的集成开发环境 Keil uVision4，宏晶公司的 STC-ISP 的使用方法以及一款硬件学习载体 51 单片机学习板。通过学习本篇内容，读者将熟悉 51 单片机开发工具的使用，为后续学习做好准备。

第 2 篇为基础篇，主要是基本功能模块实验，介绍了 51 单片机的主要基本功能模块，包括 C51 语言、51 单片机 I/O 口基本功能、外部中断、定时器/计数器、串口通信、数码管显示控制以及独立按键和矩阵键盘的控制等，针对主要知识点设计有相应的实验内容。实验内容可采用软件仿真实现或硬件实现。通过本篇内容的学习，读者将掌握 51 单片机主要基本功能模块的应用，这是后续完成完整系统设计的基础。

第 3 篇为提升篇，主要是单片机的综合应用，其中包括五个综合实训项目，选取了基于 51 单片机的数字频率计设计、直流电机 PWM 脉宽调速、基于 LCD1602 液晶显示屏的矩阵键盘电子密码锁设计、基于 DS18B20 的温度报警器设计几个应用的设计实例，具有一定的代表意义和实用价值。

本书的特色在于所有实验项目的安排都站在初学者的角度，遵循由易到难、循序渐进的原则，每个程序都有对应的硬件电路连接图，并对涉及的知识点进行了适当介绍，使读者更加容易理解，进而达到举一反三的目的。由于作者水平有限，书中难免存在纰漏，敬请读者批评指正。如有意见请发至作者邮箱：yangchunli@dali.edu.cn。

作 者

2019 年 11 月于大理

目 录

Part 1 准备篇——51 单片机开发工具介绍及使用

1.1 单片机应用系统开发流程简介	2
1.2 Keil 软件使用说明	3
1.3 STC - ISP 的使用说明	12
1.4 硬件载体——51 单片机学习板	13

Part 2 基础篇——51 单片机基本功能模块实验

2.1 I/O 口的基本功能	16
2.1.1 实验目的	16
2.1.2 知识准备	16
实验 1 点亮一个 LED 灯	20
实验 2 点亮多个 LED 灯	23
实验 3 单灯闪烁实验	24
实验 4 流水灯实验	27
2.2 数码管显示	31
2.2.1 实验目的	31
2.2.2 知识准备	31
实验 5 单个数码管显示	33
实验 6 LED 数码管的静态显示	34
实验 7 LED 数码管的动态显示	36
实验 8 LED 数码管的自减	38
2.3 按键控制	41
2.3.1 实验目的	41
2.3.2 知识准备	41
实验 9 独立按键控制数码管加减	43
实验 10 4 路抢答器的设计	47
实验 11 矩阵键盘控制数码管显示	51
2.4 外部中断	54
2.4.1 实验目的	54
2.4.2 知识准备	54
实验 12 外部中断 0 的低电平触发方式	57

实验 13 外部中断 0、外部中断 1 混合使用	59
2.5 定时器/计数器及定时器中断	64
2.5.1 实验目的	64
2.5.2 知识准备	64
实验 14 定时器/计数器产生方波	67
实验 15 定时器/计数器 T0、T1 控制数码管加减	68
实验 16 交通灯的设计实验	71
2.6 单片机串口通信	75
2.5.1 实验目的	75
2.5.2 知识准备	75
实验 17 串行口的工作方式 0	77
实验 18 串行口的工作方式 1 向电脑发送数据	79
实验 19 串行口的工作方式 1 从电脑接收数据	81
2.7 LCD1602 字符型液晶显示	85
2.7.1 实验目的	85
2.7.2 知识准备	85
实验 20 LCD1602 液晶显示屏的静态显示	89

Part 3 提升篇——51 单片机系统综合运用实验

实训 1 基于 51 单片机的数字频率计设计	96
实训 2 直流电机 PWM 脉宽调速实验	100
实训 3 基于 LCD1602 液晶显示屏的矩阵键盘电子密码锁设计	105
实训 4 基于 DS18B20 的温度报警器设计	113

附 录

附录 A 51 单片机学习板原理图	128
附录 B <reg52.h> 头文件	129
附录 C <intrins.h> 头文件	132
附录 D C51 库函数	133
参考文献	141

Part 1 准备篇

—— 51 单片机开发工具介绍及使用

本篇是 51 单片机开发的准备篇，主要讲解单片机应用系统的开发流程和 51 单片机的主要开发工具，包括 ARM 公司的集成开发环境 Keil μ Vision4、宏晶公司的 STC - ISP 的使用方法以及一款硬件学习载体——51 单片机学习板。通过学习本篇内容，读者将熟悉 51 单片机开发工具的使用，为后续学习做好准备。

1.1 单片机应用系统开发流程简介

单片机应用系统是指以单片机为控制核心，配以一定的外围电路，能够实现一定功能的系统。

一旦单片机系统产品研发完成，软件就固化在硬件环境中。单片机软件是针对相应的单片机硬件系统开发的，是专用的。

任何一个单片机系统产品都是软件和硬件的逻辑结合体，用户必须根据产品对系统功能、性能参数的要求，对软件和硬件统一考虑进行设计开发，才能达到最佳效果。

开发设计单片机应用系统主要包括以下几个步骤（如图 1.1.1 所示）：

第一，项目分析、拟订设计方案；

第二，根据拟订的设计方案进行软件、硬件设计；

第三，系统联调、测试，结合项目要求修改软件、硬件设计，直至完全符合要求。

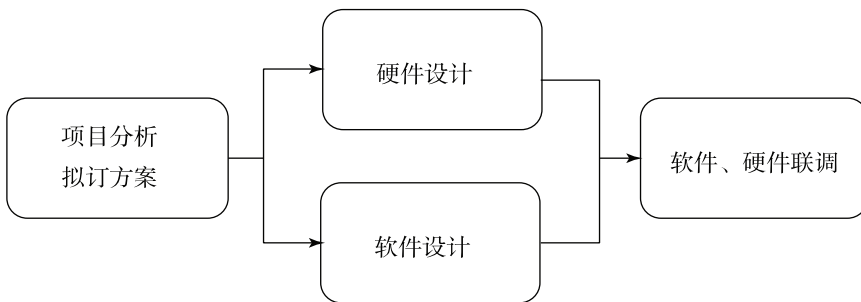


图 1.1.1 单片机应用系统开发流程

1.2 Keil 软件使用说明

Keil 是美国 Keil Software 公司出品的一套非常好用的 8051 系列单片机软件开发系统。Keil 提供了包括 C 编译器、宏汇编、连接器、项目管理和一个功能强大的仿真调试器等在内的完整开发方案，通过一个集成开发环境（uVision）将这些部分组合在一起。如果使用 C 语言编程，那么 Keil 几乎就是不二之选，即使不使用 C 语言而使用汇编语言编程，Keil 方便易用的集成环境、强大的软件仿真调试工具也会令用户事半功倍。在后面的学习中，我们对 Keil 和 μ Vision4 两个术语不做严格的区分，一般多称呼为 Keil μ Vision4，包含 μ Vision4 集成开发环境之意。Keil 软件启动后，呈现的编辑状态的操作界面，如图 1.2.1 所示。



图 1.2.1 Keil μ Vision4 软件编辑状态下的操作界面

Keil μ Vision4 的项目开发流程主要包含以下步骤：运行 Keil μ Vision4；创建项目文件；创建程序文件并保存；将源文件添加到项目文件中；编写程序代码；创建 Hex 文件；编译项目。

下面以利用 Keil μ Vision4 创建一个 LED 程序为例，介绍使用 Keil μ Vision4 的项目开发流程。


①运行 Keil μ Vision4。首先双击 Keil μ Vision4 文件图标 ，会弹出 Keil 界面，如图 1.2.2 所示。



图 1.2.2 Keil 界面

接着会自动进入 Keil μVision4 的开发环境，如图 1.2.3 所示。

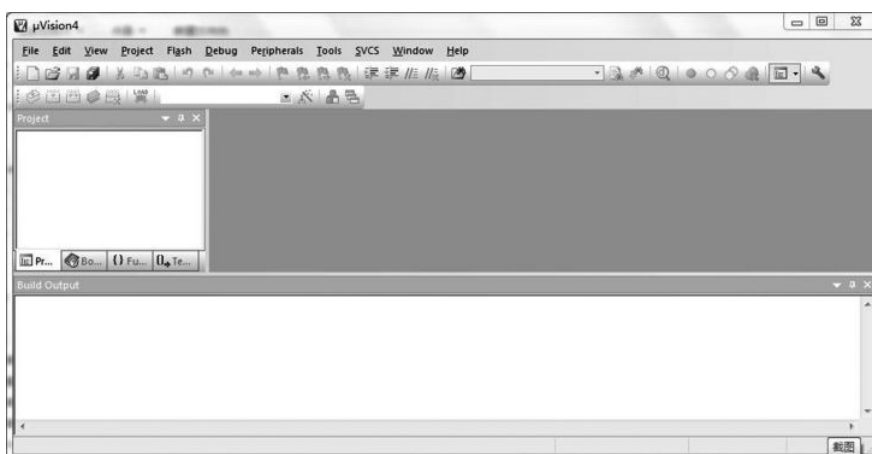


图 1.2.3 Keil μVision4 开发环境

②创建项目文件。运行【Project】菜单中的【New μVision Project...】，接着会弹出【Create New Project】对话框，如图 1.2.4 所示。我们可以选择工程需要保存的路径，输入工程文件名，然后点击【保存】按钮。

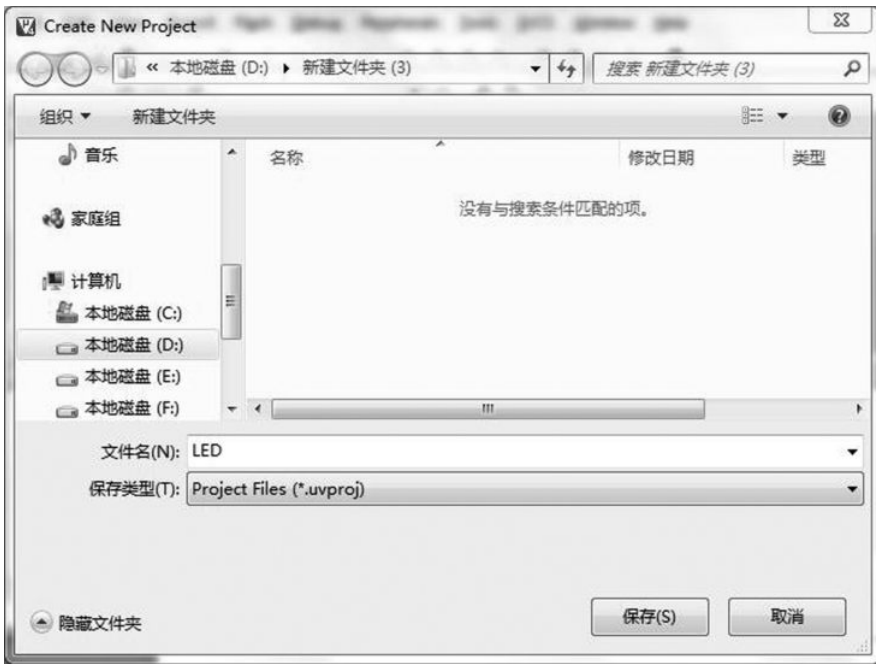


图 1.2.4 新建工程

接下来会弹出如图 1.2.5 所示对话框，选择单片机的生产厂家以及型号。

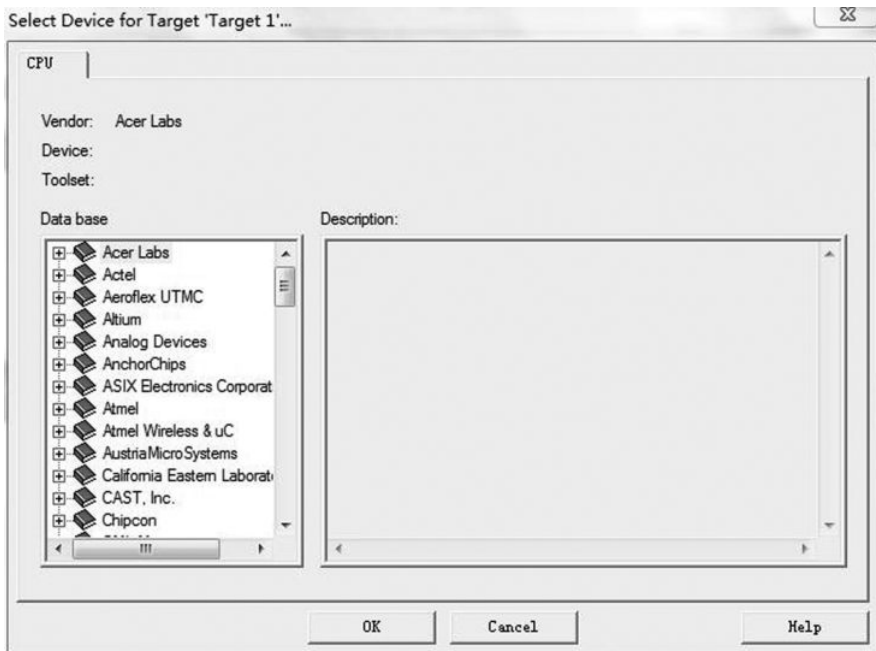


图 1.2.5 选择单片机厂家以及型号

我们这里选择【Atmel】公司的【AT89C52】芯片，在如图 1.2.6 所示对话框的【Description】中可以看到该单片机资源的简单介绍，然后点【OK】即可。

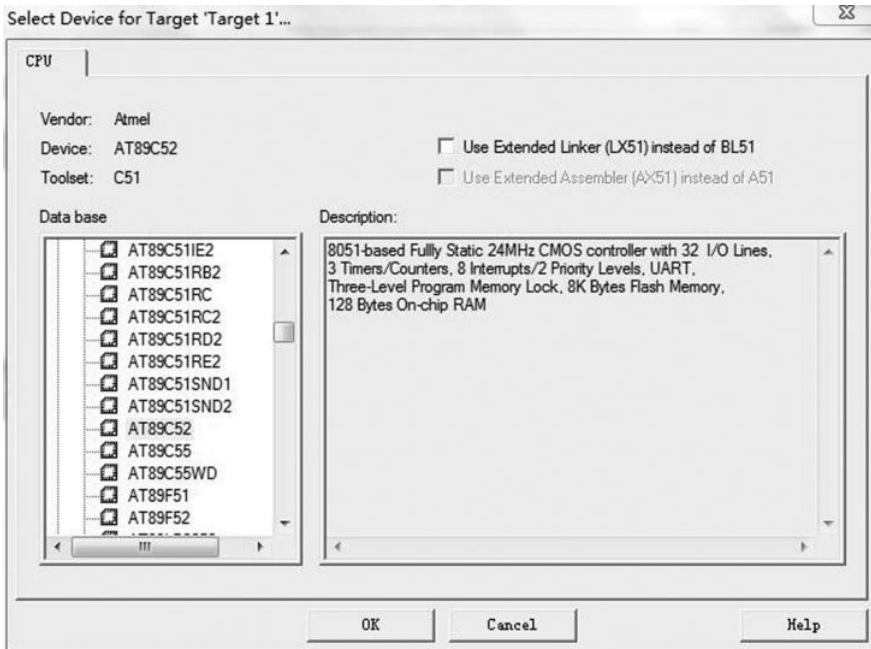


图 1.2.6 选择相应的单片机

接下来会弹出一个是否需要添加启动代码的对话框，如图 1.2.7 所示。我们这里选择【是】即可。添加进入工程的启动代码为“STARTUP. A51”。

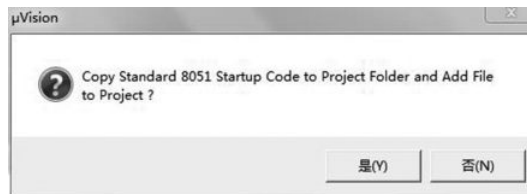


图 1.2.7 添加启动代码

如图 1.2.8 所示，我们的工程已经创建成功了。

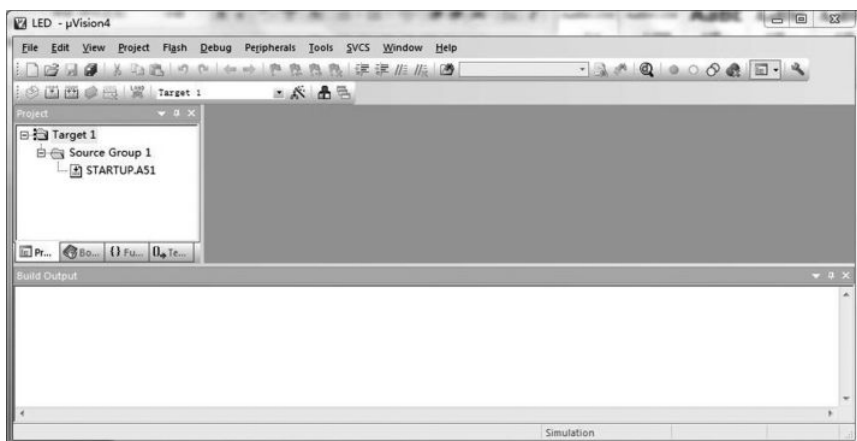



图 1.2.8 工程创建成功

③创建程序文件并保存。工程创建完成后，接着我们来创建程序文件。单击【File】菜单中的【New】选项，或者单击界面上的快捷图标，然后会弹出如图 1.2.9 所示的界面。

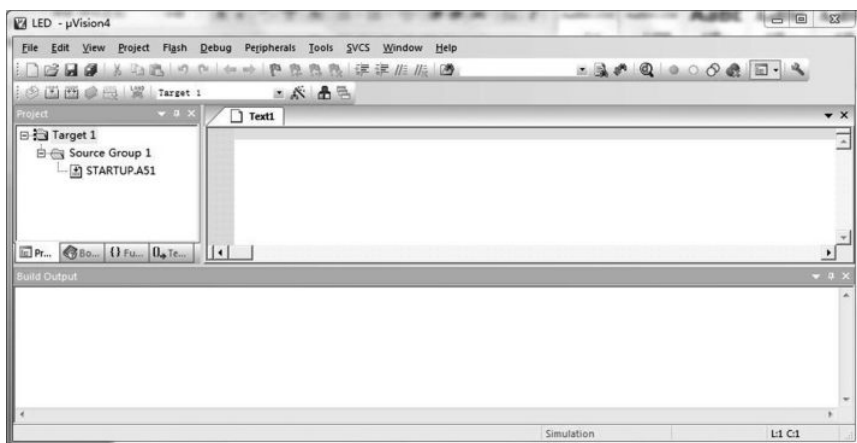


图 1.2.9 创建新文件


单击【File】菜单中的【Save】选项，或者单击界面上的快捷图标，会弹出如图 1.2.10 所示的对话框。在【文件名】中，我们输入相应的文件名（建议文件名和工程名起为相同的主名）。如果用 C 语言编写代码，则扩展名必须为 .c，如图 1.2.10 所示；如果用汇编语言编写代码，则扩展名为 .asm。最后单击【保存】按钮。



图 1.2.10 保存文件

④将源文件添加到项目文件中。在左边工程窗口中用右键点击【Source Group 1】，在下拉菜单中选择【Add Files to Group 'Source Group 1'】，如图 1.2.11 所示。

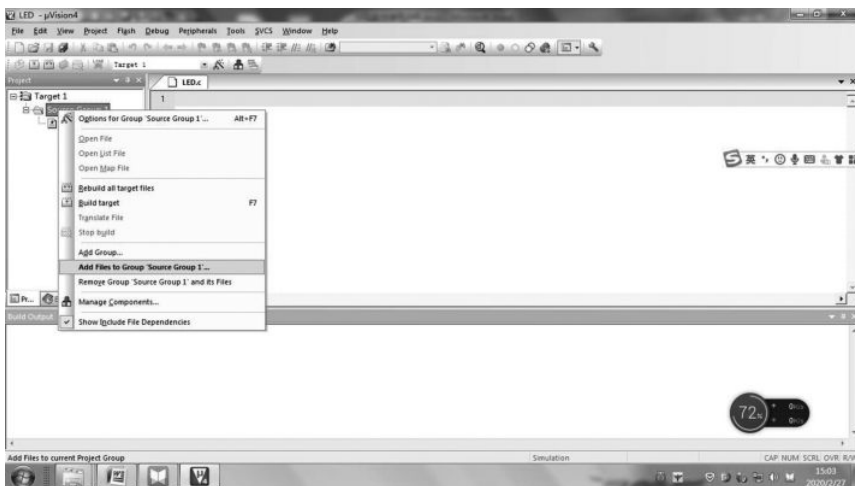


图 1.2.11 将文件添加到工程菜单

选择在如图 1.2.10 所示界面中保存的相应的 C 文件，单击【Add】按钮，最后单击【Close】按钮，如图 1.2.12 所示。




图 1.2.12 选择要添加的文件

⑤编写程序代码。在文件编辑窗口中，编写程序代码，如图 1.2.13 所示。



图 1.2.13 编写程序代码

⑥创建 HEX 文件。用右键单击工程窗口【Target1】下拉菜单中的【Options for Target 'Target 1'】或者单击界面上的 ，如图 1.2.14 所示，在弹出的菜单中，选择【Output】选项，然后勾选【Create HEX File】，最后点【OK】，如图 1.2.15 所示。

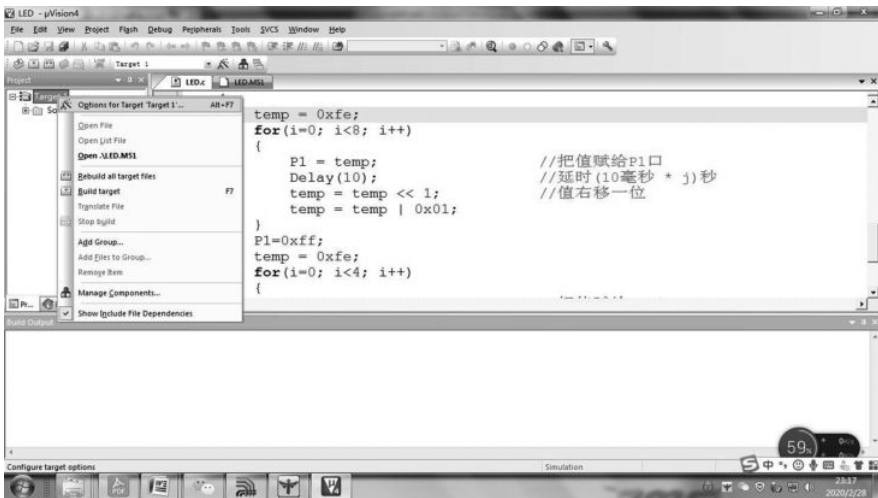


图 1.2.14 进入设置设备选项

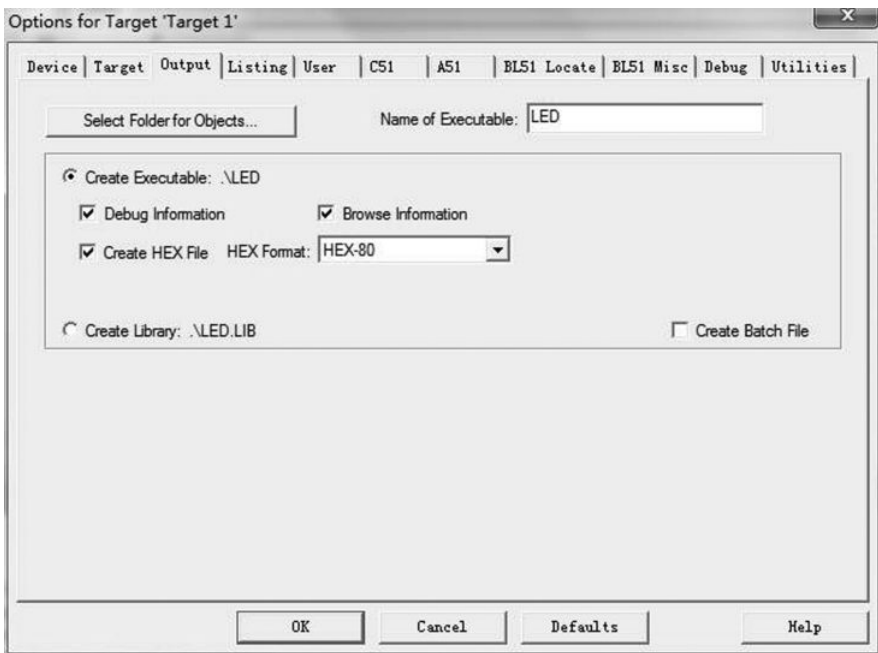


图 1.2.15 勾选“Create HEX file”选项




⑦编译项目。单击如图 1.2.16 所示的工具栏上的【Build】按钮  或【Rebuild】按钮 ，如果系统提示如图 1.2.17 所示的“0 Error (s), 0 Warning (s)”，则表示系统编译成功（程序消除了语法错误，但不能保证没有逻辑错误）；如果系统提示如图 1.2.18 所示的“Target not created”，则提示仍有错误，需根据错误指示修正源代码。代码修改后再再次单击【Rebuild】按钮 重新编译，直至编译成功。



图 1.2.16 编译文件

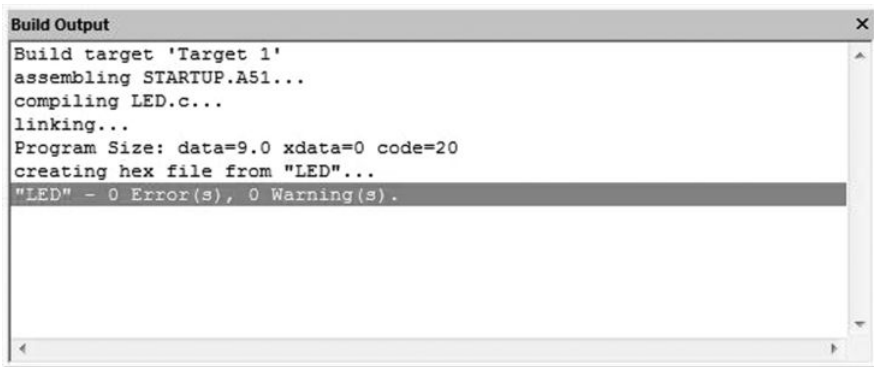


图 1.2.17 编译成功

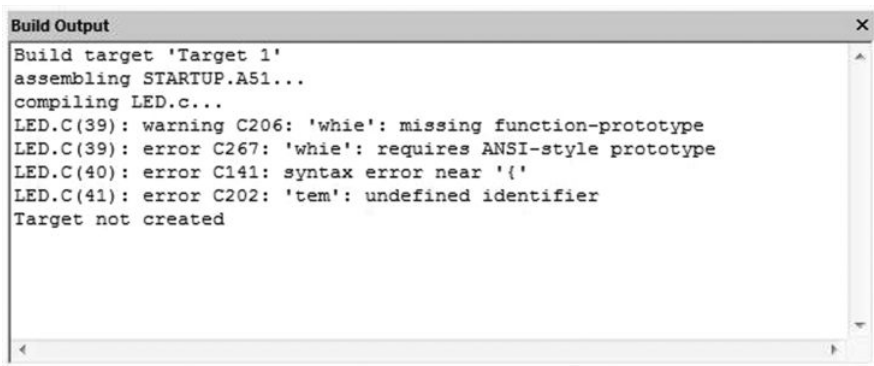


图 1.2.18 程序有错误

系统编译成功后，会自动生成扩展名为 .hex 的文件。HEX 文件（全称 Intel HEX）是由一行行符合 Intel HEX 文件格式的文本所构成的 ASCII 文本文件。Intel HEX 文件中，每一行包含一个 HEX 记录。这些记录由对应机器语言码和常量数据的十六进制编码数字组成。Intel HEX 文件通常用于传输将被存于 ROM 或者 EPROM 中的程序和数据。大多数 EPROM 编程器或模拟器使用 Intel HEX 文件。

我们可以通过相应的烧录软件（如 PROGISP 或 STC - ISP 等）把 HEX 文件烧写到单片机的 ROM 中，让单片机按照我们编写的代码进行运行。