



卓越工程师培养计划

▪ 单片机 ▪

◎ 侯玉宝 陈忠平 邬书跃 编著



51单片机

C语言程序设计 经典实例 (第2版)



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>

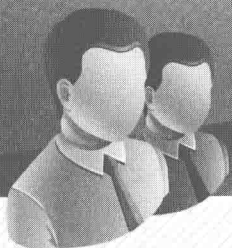


卓越工程师培养计划

■ 单片机 ■

<http://www.phei.com.cn>

◎ 侯玉宝 陈忠平 邬书跃 编著



51单片机

C语言程序设计 经典实例 (第2版)

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书是“以项目为载体，采用任务驱动方式”编写而成的。本书以 STC89C52RC 单片机为蓝本，结合 Keil C51、Proteus 虚拟仿真软件，从实验、实践、实用的角度出发，通过丰富的实例详细介绍了 51 系列单片机 C 语言程序设计和单片机控制系统的应用技术。本书的主要内容包括单片机系统的软/硬件开发环境、C51 程序设计语言基础、LED 灯光设计、按键控制与数码管显示、中断控制应用设计、定时器/计数器控制应用设计、单片机串行通信设计、LED 点阵显示器设计、LCD 液晶显示设计、A/D 与 D/A 转换、串行总线扩展及应用设计、综合应用设计实例共 12 个项目 67 个任务。

本书内容全面，取材新颖，叙述清楚，理论联系实际，突出实用特色。本书适合单片机爱好者自学使用，还可作为高等院校 51 单片机课程“教、学、做”一体化的教学用书，也是 51 系列单片机应用开发人员的实用参考书。

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

51 单片机 C 语言程序设计经典实例/侯玉宝，陈忠平，邬书跃编著. —2 版. —北京：电子工业出版社，2016.5

(卓越工程师培养计划)

ISBN 978-7-121-28777-0

I. ①5… II. ①侯… ②陈… ③邬… III. ①单片微型计算机—C 语言—程序设计 IV. ①TP368.1
②TP312

中国版本图书馆 CIP 数据核字 (2016) 第 098982 号

责任编辑：张 剑 (zhang@phei.com.cn)

印 刷：三河市华成印务有限公司

装 订：三河市华成印务有限公司

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：27.25 字数：698 千字

版 次：2012 年 5 月第 1 版

2016 年 5 月第 2 版

印 次：2016 年 5 月第 1 次印刷

印 数：3 000 册 定价：69.00 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, 88258888。

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

本书咨询联系方式：zhang@phei.com.cn。

第2版前言

单片机是芯片级的微型计算机系统，可以嵌入到各种应用系统中，以实现智能化控制。近30年来，8位单片机以其性价比高、功耗低、易于开发等优点，加上嵌入式C语言的推广普及，片载Flash程序存储器及其在系统内可编程（ISP）和在应用中编程（IAP）技术的广泛应用，使其越来越受到广大电子工程师的喜爱。

本书第1版自2012年5月出版以来，已被许多学校或培训机构作为单片机课程的实践教材来使用，受到众多教师、学生和读者的认可，在此我们表示衷心的感谢。该书以国内最流行的80C51系列单片机的硬件和软件的设计为背景，以C语言为基础，以项目为载体，采用任务驱动方式的教学方法，通过丰富的C程序实例，由浅入深地介绍了80C51系列单片机的基础知识及各种应用开发技术。

鉴于单片机及嵌入式系统技术发展迅速，决定对本书进行修订。第2版坚持原版“项目为载体，任务带动教学”、“软硬结合，虚拟仿真”、“C语言编程，增强可读性”、“兼顾原理，注重实用”的编写原则，并在此基础上，根据读者的建议对原版进行修订与补充。

与第1版相比，本书第2版主要在以下3个方面进行了修订。

- ① 将第1版的项目一和项目二整合为一个项目，并精简为3个任务，使用的编译软件Keil和仿真软件Proteus均为最新版本。
- ② 由于篇幅原因，仅第1版项目三中的内容全部保留，而第1版其余项目中均删除了部分任务。
- ③ 为了增强读者的综合实践能力，本书第2版新增加了综合应用设计实例项目。该项目中包含4个不同的综合实践任务，以进一步加强、巩固读者对定时器控制、中断控制、矩阵键盘控制、数码管动态显示控制、LCD液晶显示控制等知识的综合应用和实际设计能力。

本书由湖南涉外经济学院侯玉宝、湖南工程职业技术学院陈忠平和湖南涉外经济学院邬书跃编著。参加本书编写的还有湖南工程职业技术学院陈建忠、李锐敏、龚亮、龙晓庆、周少华，湖南航天诚远精密机械有限公司刘琼，湖南涉外经济学院高金定，湖南科技职业技术学院高见芳，湖南三一重工集团王汉其，湖南航天局7801研究所武娟梅、袁芳和葛建。全书由湖南工程职业技术学院徐刚强教授主审，在编写过程中还得到了湖南工程职业技术学院许睿等诸多高工和老师的大力支持及帮助，在此向他们表示衷心的感谢。同时，对在编写过程中参考的多部51单片机原理及相关著作的作者表示深深的谢意！由于编者知识水平和经验有限，书中难免存在缺点和错误，恳请广大读者给予批评指正。

编著者

第 1 版前言

单片机是芯片级的微型计算机系统，具有性价比高、功耗低、易于开发等优点，可以嵌入各种应用系统中，以实现智能化控制。近 20 年来，嵌入式 C 语言的推广普及，片载 Flash 程序存储器及其在系统内可编程（ISP）和在应用中编程（IAP）技术的广泛采用，使得单片机越来越受到广大电子工程师的欢迎。

本书以国内最流行的 80C51 系列单片机的硬件和软件设计为背景，以 C 语言为基础，以项目为载体，采用任务驱动方式的教学方法，通过丰富的 C 语言程序实例，由浅入深地介绍了 80C51 系列单片机的基础知识及各种应用开发技术。在编写过程中，作者注重题材的取舍，使本书具有以下 4 个特点。

1. 项目为载体，任务带动教学

本书是“以项目为载体，采用任务驱动方式”编写而成的，强调“教、学、做”一体化，坚持理论知识够用的原则，并将知识点分散到多个任务中，使读者能够边学边做，轻松完成单片机学习之旅。

2. 软硬结合，虚拟仿真

沿用传统单片机学习与开发经验，通过相关编译软件（如 Keil）编写程序并生成 *.Hex 文件，然后在 Proteus 中绘制硬件电路图（这一过程相当于硬件电路的焊接），调用 *.Hex 文件进行虚拟仿真（这一过程相当于硬件调试）。对于单片机初学者来讲，这样可节约学习成本，提高学习积极性；对于单片机系统开发人员来讲，可缩短开发时间，提高设计效率，降低开发成本。

3. C 语言编程，增强可读性

C 语言是一种编译型程序设计语言，它兼顾了多种高级语言的特点，并具备汇编语言的功能。用 C 语言来编写程序会大大缩短开发周期，可以明显地增加程序的可读性，便于改进和扩充。采用 C 语言进行单片机程序设计是单片机开发与应用的必然趋势。许多人员在学习 MCS—51 单片机时，均先学习了汇编语言，然后再学习用 C 语言编写 MCS—51 程序代码，通过这种历程他们深深地感悟：汇编指令太枯燥，学习起来费时费力，用汇编语言编写一个程序或读懂程序不是一件容易的事情；使用 C 语言进行编程时，不必对单片机的硬件结构有很深入的了解，编写程序相对简单，且程序的可读性和可移植性均很强。

4. 兼顾原理，注重实用

基本原理、基本实例一直是学习和掌握单片机应用技术的基本要求，本书侧重于实际应用，因此很少讲解相关理论知识，这样避免了知识的重复讲解。为紧随技术的发展，在编写过程中还注重知识的新颖性和实用性，因此本书介绍了 SPI 总线、I²C 总线、1 - Wire 总线芯片的使用方法，使读者学习的知识能够紧随时代发展的步伐。

参加本书编写的有湖南工程职业技术学院陈忠平、徐刚强、李锐敏，湖南航天局 7801 研究所刘琼，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陈建忠教授主审，在编写过程中还得到了湖南工程职业技术学院龚亮、龙晓庆、许睿等众位高工和老师的大力支持及帮助，在此向他们表示衷心的感谢。同时，对在编写过程中参考的多部 51 单片机原理及相关著作的作者表示深深的谢意！由于编者知识水平和经验的局限性，书中难免存在缺点和错误，敬请广大读者给予批评指正。

编著者

目 录

| | |
|--------------------------|-----|
| 项目一 单片机系统的软、硬件开发环境 | 1 |
| 任务 1 单片机最小应用系统的组成 | 1 |
| 任务 2 Keil C51 编译软件的使用 | 5 |
| 任务 3 Proteus 8.0 仿真软件的使用 | 11 |
| 项目二 C51 程序设计语言基础 | 22 |
| 任务 1 C51 程序组成的识读 | 22 |
| 任务 2 运算符和表达式 | 28 |
| 任务 3 程序结构及流程控制 | 34 |
| 任务 4 数组与指针 | 40 |
| 任务 5 函数与编译预处理 | 46 |
| 项目三 LED 灯光设计 | 51 |
| 任务 1 LED 控制原理 | 51 |
| 任务 2 流水灯的设计 | 54 |
| 任务 3 拉幕式与闭幕式广告灯的设计 | 67 |
| 项目四 按键控制与数码管显示 | 74 |
| 任务 1 键盘控制原理 | 74 |
| 任务 2 LED 数码管显示原理 | 76 |
| 任务 3 查询式按键设计 | 79 |
| 任务 4 8 位数码管动态显示设计 | 82 |
| 任务 5 数码管显示矩阵键盘按键的设计 | 87 |
| 任务 6 简单拨号键盘的设计 | 93 |
| 项目五 中断控制应用设计 | 101 |
| 任务 1 中断控制原理 | 101 |
| 任务 2 采用外中断控制的条形 LED 彩灯设计 | 105 |
| 任务 3 采用两个外中断实现 LED 键控移位 | 109 |
| 任务 4 采用外中断实现计数 | 112 |
| 任务 5 采用外中断进行计数和清零控制 | 115 |
| 项目六 定时器/计数器控制应用设计 | 119 |
| 任务 1 定时器/计数器控制原理 | 119 |

| | |
|----------------------------------|------------|
| 任务 2 简单计数器的设计 | 124 |
| 任务 3 方波信号发生器 | 128 |
| 任务 4 用 T2 实现流水灯控制 | 132 |
| 任务 5 59s 计时器的设计 | 135 |
| 任务 6 简单门铃设计 | 140 |
| 任务 7 速度可调流水灯控制 | 143 |
| 任务 8 简单电子频率计 | 146 |
| 项目七 单片机串行通信设计 | 153 |
| 任务 1 单片机串行通信原理 | 153 |
| 任务 2 甲机通过串口控制乙机 LED 显示状态 | 160 |
| 任务 3 甲机通过串口控制乙机计时 | 164 |
| 任务 4 单片机双机通信 | 170 |
| 任务 5 单片机向主机发送字符串 | 176 |
| 任务 6 单片机与 PC 通信 | 180 |
| 项目八 LED 点阵显示器设计 | 187 |
| 任务 1 LED 点阵显示器控制原理 | 187 |
| 任务 2 一个 8×8 LED 点阵字符串显示 | 189 |
| 任务 3 两个 8×8 LED 点阵字符串显示 | 195 |
| 任务 4 两个 8×8 LED 点阵滚动显示 | 201 |
| 任务 5 一个 16×16 LED 点阵汉字显示 | 204 |
| 任务 6 一个 16×16 LED 点阵汉字移位显示 | 209 |
| 任务 7 两个 16×16 LED 点阵汉字显示 | 212 |
| 任务 8 两个 16×16 LED 点阵汉字移位显示 | 218 |
| 项目九 LCD 液晶显示设计 | 222 |
| 任务 1 LCD 液晶显示原理 | 222 |
| 任务 2 字符式 LCD 静态显示 | 233 |
| 任务 3 字符式 LCD 移位显示 | 236 |
| 任务 4 汉字式 LCD 静态显示 | 241 |
| 任务 5 汉字式 LCD 移位显示 | 250 |
| 任务 6 汉字式 LCD 滚动显示 | 259 |
| 任务 7 汉字式 LCD 图片显示 | 268 |
| 项目十 A/D 与 D/A 转换 | 281 |
| 任务 1 ADC0809 模/数转换 | 281 |
| 任务 2 ADC0832 模/数转换 | 287 |
| 任务 3 TLC549 模/数转换 | 293 |
| 任务 4 TLC2543 模/数转换 | 299 |

| | | |
|-------------|------------------------------|------------|
| 任务 5 | DAC0832 数/模转换 | 306 |
| 任务 6 | TLC5615 数/模转换 | 311 |
| 任务 7 | PCF8591 的 A/D 及 D/A 转换 | 318 |
| 项目十一 | 串行总线扩展及应用设计 | 332 |
| 任务 1 | 24C04 开启次数统计 | 332 |
| 任务 2 | PCF8574 串行总线扩展 | 342 |
| 任务 3 | MAX7219 控制数码管动态显示 | 349 |
| 任务 4 | DS18B20 测温 | 357 |
| 任务 5 | DS1302 可调日历时钟 | 365 |
| 项目十二 | 综合应用设计实例 | 385 |
| 任务 1 | LCD 数字钟的设计 | 385 |
| 任务 2 | 篮球比赛计分器的设计 | 391 |
| 任务 3 | 电子音乐播放器的设计 | 400 |
| 任务 4 | 电子密码锁的设计 | 411 |
| 附录 A | Proteus 常用快捷键 | 418 |
| 附录 B | C51 库函数 | 419 |
| 参考文献 | | 425 |

项目一 单片机系统的软、硬件开发环境

【知识目标】

- ☺ 掌握 STC89C51 单片机最小系统的组成及相关电路的工作原理。
- ☺ 掌握单片机编译软件 Keil μ Vision 5 的使用方法。
- ☺ 单片机虚拟仿真软件 Proteus 8 Professional 的使用方法。

【能力目标】

- ☺ 学会搭建单片机最小应用系统电路。
- ☺ 学会利用 Keil μ Vision 5 软件对单片机 C 程序进行编译与调试。
- ☺ 学会利用 Proteus 8 Professional 软件进行单片机程序的硬件仿真。



任务 1 单片机最小应用系统的组成

单片机最小应用系统又称为单片机基本系统，是指用最少的元器件能使单片机工作起来的一个最基本的应用系统。在这种系统中，使用 STC89 系列单片机的一些内部资源就能够满足硬件设计需求，不需扩展外部的存储器或 I/O 接口等器件，通过用户编写的程序，单片机就能够达到控制的要求。

单片机的最小应用系统结构只能使用在控制较简单的场合，该系统包括单片机、时钟电路、复位电路等部分。同时，单片机要正常运行，还必须具备电源正常、时钟正常、复位正常 3 个基本条件。STC89C51 单片机组成的最小应用系统电路原理图如图 1-1 所示。

从图 1-1 中可以看出，电路以 STC89C51 单片机为核心，STC89C51 的第 18 脚、第 19 脚外接由 C1、C2 和 X1 构成的石英晶体振荡器电路；STC89C51 的第 9 脚外接由 K1、R1、R2 和 C3 构成的按钮复位电路；STC89C51 的第 31 脚外接电源，以进行片内和片外程序存储器的选择控制。当然，STC89C51 单片机要正常工作，还需提供电源，因此在实际电路中，STC89C51 的第 20 脚应该接地，而第 40 脚应该接电源 +5V。

图 1-1 所示的单片机最小应用系统通上电时，单片机就开始工作，4 组 P 端口处于高电平的状态。在这种情况下，是否说明单片机正常工作呢？单片机要完成相应的任务操作，还需要程序来进行控制，没有固化程序的单片机系统不能完成任何实质上的工作，所以图 1-4 所示的电路在通电后，单片机进入工作准备就绪状态。

1. 电源电路

电源电路是单片机工作的动力源泉。对应的接线方法为：单片机的第 40 引脚 (V_{CC}) 为电源引脚，工作时接 +5V 电源；第 20 引脚 (V_{SS}) 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

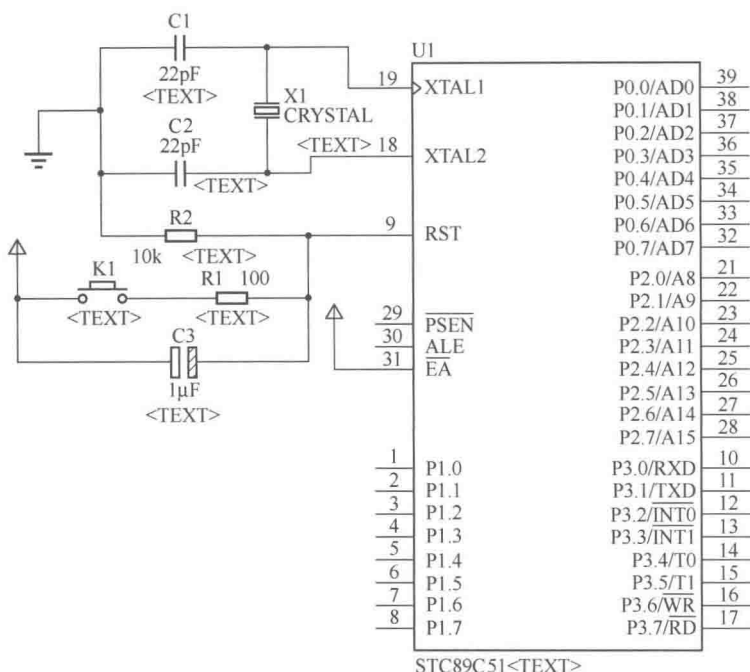


图 1-1 STC89C51 单片机最小应用系统电路原理图

动下进行的。时钟电路就好像人的心脏，如果人的心跳停止了，人就没有生命。同样，如果单片机的时钟电路停止工作，那么单片机也就停止运行了。单片机的时钟具有两种工作模式，即片内时钟和片外时钟模式。

1) 片内时钟模式 STC89 系列单片机的内部也有 1 个高增益单级反相放大器，XTAL1 为反相放大器的输入端，XTAL2 为反相放大器的输出端。单片机的这个反相放大器与作为反馈元件的片外晶体或陶瓷谐振器和电容一起构成了稳定的自激振荡器，发出的脉冲直接送入内部的时钟电路，作为单片机 CPU 的时钟。图 1-2 所示为片内时钟模式电路的连接方法。

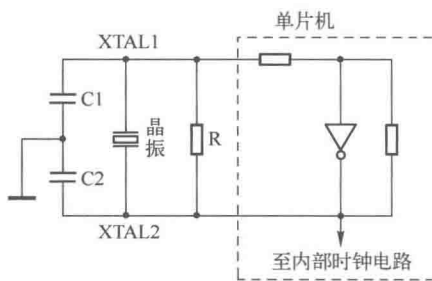


图 1-2 片内时钟模式电路

当外接晶振时，电容 C1 和 C2 容量值通常选择 30pF；外接陶瓷谐振器时，C1 和 C2 的典型值约为 47pF。在设计印制电路板 (PCB) 时，晶体或陶瓷谐振器和电容应尽可能安装在单片机芯片附近，以减少寄生电容，保证振荡器稳定和可靠工作。为了提高温度稳定性，应采用 NPO 电容（具有温度补偿特性的单片陶瓷电容器）。C1、C2 对频率有微调作用，晶振频率越高，系统时钟频率也越高，单片机的运行也就越快。运行速度越快，对存储器的速度要求就越高，对 PCB 的工艺要求也越高。

使用片内时钟模式时，如何判断单片机的外接晶振是否工作呢？可以使用电压法来进行检测。具体操作是：使用高阻电压表分别测量 XTAL1 和 XTAL2 两引脚的对地电压，在正常情况下，两引脚的对地电压应该是电源电压的 1/2 或者更低一些，且 XTAL2 引脚上的电压要高于 XTAL1 引脚上的电压。

注意：使用此方法时，会对频率有一点影响，严重的会导致晶振停振，这是因为电压表一加上相当于在振荡电路上又并上或串上了分布电容、电阻和电感等，这样就影响了原来电路的状态。

2) 片外时钟模式 在系统中，若有多片单片机时，为了使各单片机之间时钟信号同步，应当引入唯一的公用外部脉冲信号作为各单片机的振荡脉冲。这个外时钟信号由外部振荡器产生，可以为有源晶振或其他的时钟芯片。但是，对于不同工艺类型的单片机，外部时钟的输入引脚不同。

对于普通的 8051 单片机，外部时钟信号由 XTAL2 引脚接入后，直接送到单片机内部的时钟发生器，而引脚 XTAL1 则应直接接地，如图 1-3 (a) 所示。

注意：由于 XTAL2 引脚的逻辑电平不是 TTL 信号，因此需要外接一个上拉电阻。

对于 CMOS 型的单片机，和普通的 8051 不同的是其内部的时钟发生器的信号取自于反相放大器的输入端。因此，外部的时钟信号应该从单片机的 XTAL1 引脚输入，而 XTAL2 引脚则需要悬空，如图 1-3 (b) 所示。这里单片机有 80C51、80C52、AT89S52、STC89C51 等。



图 1-3 片外时钟模式电路

外部脉冲信号通过一个二分频的触发器而成为内部时钟信号，故对外部信号的占空比没有什么要求，但最小的高电平和低电平持续时间应符合产品技术的要求。如果 STC89 系列单片机的时钟频率超过 33MHz 时，应直接使用外部有源晶振。

STC89 系列单片机时钟电路中 R 的阻值、C1 和 C2 容量的大小与单片机的时钟模式、晶振频率有关，见表 1-1。对于 STC89C5xD 和 STC89LV5xD 单片机而言，外接晶振的频率范围则与单片机的工作电压有关，见表 1-2。

表 1-1 R、C1 和 C2 的规格选择

| 时钟模式 | 12 时钟模式 | | | | | | |
|------------|---------------|------|-------|-------|-------|-------|-------|
| | X1 (晶振) /MHz | 2~25 | 26~30 | 31~35 | 36~39 | 40~43 | 44~48 |
| C1, C2/ pF | ≤47 | ≤10 | ≤10 | ≤10 | ≤10 | ≤10 | |
| R/ kΩ | 不用 | 6.8 | 5.1 | 4.7 | 3.3 | 3.3 | |
| 时钟模式 | 6 时钟模式 | | | | | | |
| | X1 (晶振) / MHz | 2~25 | 26~30 | 31~35 | 36~39 | 40~43 | 44~48 |
| | C1, C2/ pF | ≤47 | ≤5 | 不用 | 不用 | 不用 | 不用 |
| | R/ kΩ | 不用 | 6.8 | 5.1 | 4.7 | 3.3 | 3.3 |

表 1-2 STC89C5xD + 和 STC89LV5xD + 单片机时钟频率范围

| 单片机工作电压 | 内部时钟 | | 外部时钟 | |
|---------|---------|---------|---------|---------|
| | 12 时钟模式 | 6 时钟模式 | 12 时钟模式 | 6 时钟模式 |
| 5.0V | 2~48MHz | 2~36MHz | 2~48MHz | 2~36MHz |
| 3.3V | 2~48MHz | 2~32MHz | 2~36MHz | 2~18MHz |

3. 复位电路

单片机的复位操作,使 CPU 和系统中的其他部件都处于一个确定的初始状态,并从这个初始状态开始工作。只要单片机的复位端 RST 保持高电平,单片机便保持复位状态。复位后,除 SP 值为 0x07、P1 ~ P3 口为 0xFF 外,其他所有特殊功能寄存器 SFR 的复位值均为 0x00。

单片机通常采用上电复位和按钮复位两种方式。图 1-4 (a) 所示为上电复位电路,图 1-4 (b) 和 (c) 所示为按钮复位电路。

上电复位是利用电容的充放电来实现的,上电瞬间,RST 端的电位与 V_{CC} 相同,RC 电路充电,随着充电电流的减少,RST 端的电位逐渐下降。只要 V_{CC} 的上升时间不超过 1ms,振荡器的建立时间不超过 10ms,该时间就足以保证完成复位操作。上电复位所需的最短时间是振荡周期建立时间加上 24 个时间周期,在这个时间内,RST 端的电平就维持高于施密特触发器 (Schmidt Trigger) 的下阈值。

按钮复位有按钮脉冲复位和按钮电平复位两种方法,如图 1-4 (b) 和 (c) 所示。按钮脉冲复位是由单片机外部提供一个复位脉冲,此脉冲保持宽于 24 个时钟周期;复位脉冲过后,由内部下拉电阻保证 RST 端为低电平。按钮电平复位是上电复位和手动复位相结合的方案,上电复位的工作过程与图 1-4 (a) 相同,在手动复位时,按下复位按钮 RESET,电容对 R1 迅速放电,RST 端变为高电平,RESET 松开后,电容通过电阻 R2 进行充电,使 RST 端恢复为低电平。

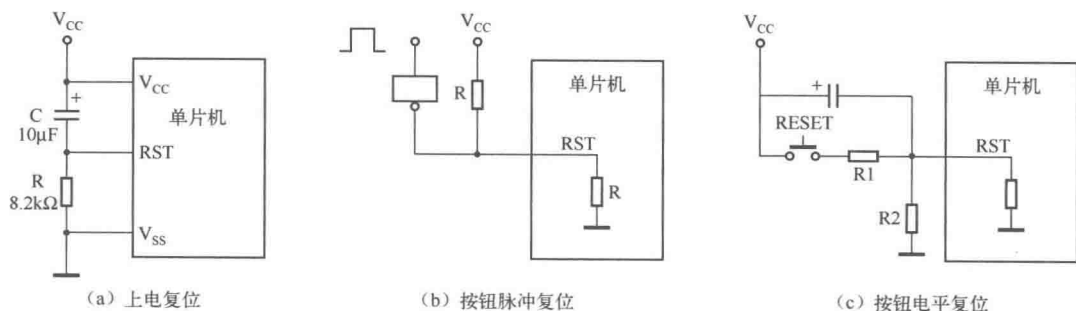


图 1-4 单片机复位电路

4. 控制引脚EA的接法

控制引脚EA为片内、片外程序存储器的选择控制引脚,当EA为低电平时,单片机从外部程序存储器取指令;当EA接高电平时,单片机从内部程序存储器中取指令。

5. 四组 I/O 端口 P0、P1、P2 和 P3

☺ P0 (P0.0 ~ P0.7): 8 位三态双向 I/O 端口。在访问外部存储器时,它是分时作为低 8 位地址线和 8

位双向数据总线用的。在不访问外部存储器时，作为通用 I/O 端口用于传送 CPU 的 I/O 数据。P0 端口能以吸收电流的方式驱动 8 个 LSTTL 负载，一般作为扩展时地址/数据总线使用。

☺ P1 (P1.0~P1.7)：带内部上拉电阻的 8 位准双向 I/O 端口（作为输入时，端口锁存器置 1）。对 P1 端口写 1 时，P1 端口被内部的上拉电阻拉为高电平，这时可作为输入口。当 P1 端口作为输入端口时，因为有内部上拉电阻，那些被外部信号拉低的引脚会输出一个电流。P1 端口能驱动（吸收或输出电流）4 个 TTL 负载，它的每个引脚都可定义为输入或输出口，其中 P1.0、P1.1 兼有特殊的功能。

◇ T2/P1.0：定时器/计数器 2 的外部计数输入/时钟输出。

◇ T2EX/P1.1：定时器/计数器 2 重载/捕捉/方向控制。

☺ P2 (P2.0~P2.7)：带内部上拉电阻的 8 位准双向 I/O 端口。当外部无扩展或扩展存储器容量小于 256B 时，P2 端口可作为一般 I/O 端口使用，扩充容量在 64KB 范围时，P2 口为高 8 位地址输出端口。当作为一般 I/O 口使用时，可直接连接外部 I/O 设备，能驱动 4 个 LSTTL 负载。

☺ P3 (P3.0~P3.7)：带内部上拉电阻的 8 位准双向 I/O 端口。向 P3 端口写入 1 时，P3 端口被内部的上拉电阻上拉为高电平，可用做输入口。当作为输入时，被外部拉低的 P3 端口会因为内部上拉而输出电流。第一功能作为通用 I/O 端口，第二功能作为控制口，见表 1-3。

表 1-3 P3 端口引脚的第二功能

| P3 引脚 | 第二功能 | 功能描述 | P3 引脚 | 第二功能 | 功能描述 |
|-------|--------------------------|-----------|-------|------------------------|-------------|
| P3.0 | RXD | 串行口输入 | P3.4 | T0 | 定时器 0 的外部输入 |
| P3.1 | TXD | 串行口输出 | P3.5 | T1 | 定时器 1 的外部输入 |
| P3.2 | $\overline{\text{INT0}}$ | 外部中断 0 输入 | P3.6 | $\overline{\text{WR}}$ | 片外数据存储器写信号 |
| P3.3 | $\overline{\text{TNT1}}$ | 外部中断 1 输入 | P3.7 | $\overline{\text{RD}}$ | 片外数据存储器读信号 |



任务 2 Keil C51 编译软件的使用

Keil C51 标准 C 编译器是众多单片机应用开发优秀软件之一，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。本任务通过 P1 端口输入、P0 端口输出实验为例，简单介绍 Keil C51 软件的基本操作方法。

1. Keil C51 软件基本操作

1) 启动 Keil C51 软件 双击桌面 Keil μ Vision5 快捷图标，弹出图 1-5 所示的画面。之后，进入 μ Vision5 集成开发环境，如图 1-6 所示。

2) 创建一个新的工程项目 在图 1-6 的界面中，执行菜单命令“Project”→“Close Project”，关闭已打开的项目。执行菜单命令“Project”→“New μ Vision Project”，弹出“Create New Project”对话框，在此对话框中选择保存路径，并输入项目名，如图 1-7 所示。

输入项目名后，单击“保存”按钮，弹出“Select Device for Target 'Target 1'...”对话框，如图 1-8 所示。因 Keil μ Vision5 中没有 STC 单片

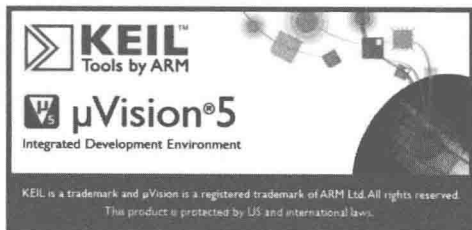


图 1-5 启动 Keil 时的画面

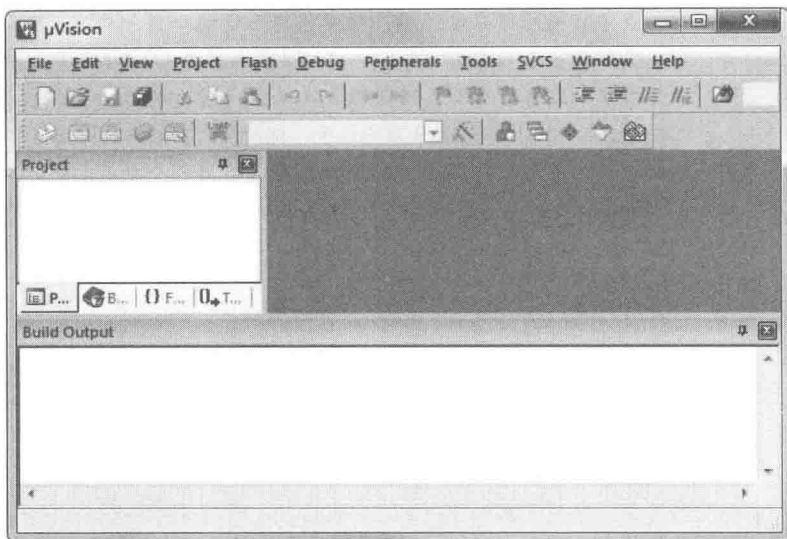


图 1-6 µVision5 集成开发环境

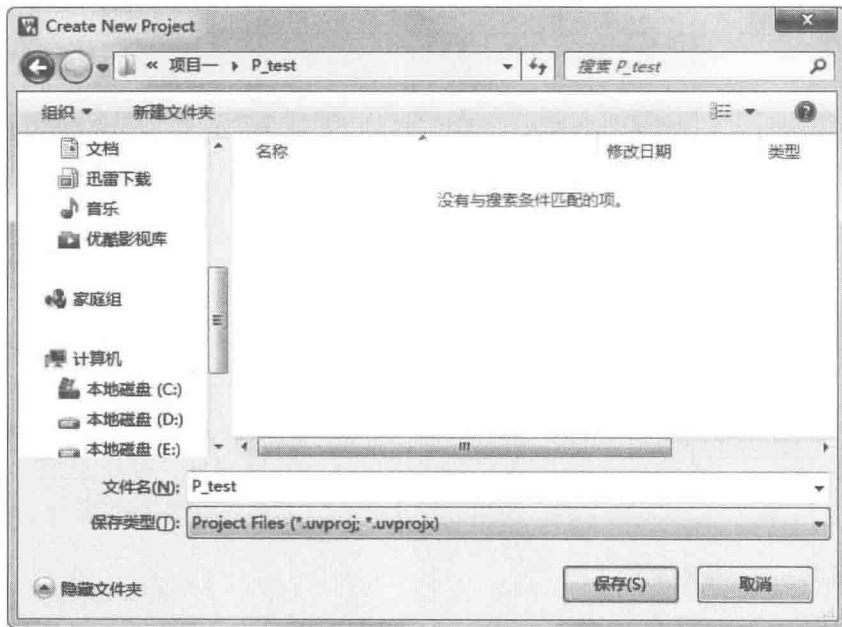


图 1-7 “Create New Project”对话框

机型号，可以用 Intel 公司的 8052/87C52/87C54/87C58、Atmel 公司的 AT89C/5152/55/55WD 或 NXP 公司的 P87C52/P87C54/P87C58/P87C51RD + 等替代。在图 1-8 中将其当做 Atmel 公司的 AT89C51RC。

选择目标芯片后，单击“OK”按钮，弹出如图 1-9 所示的对话框，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。在此单击“否”按钮，项目窗口中将不添加启动代码；单击“是”按钮，项目窗口中将添加启动代码。这二者的区别如图 1-10 所示。

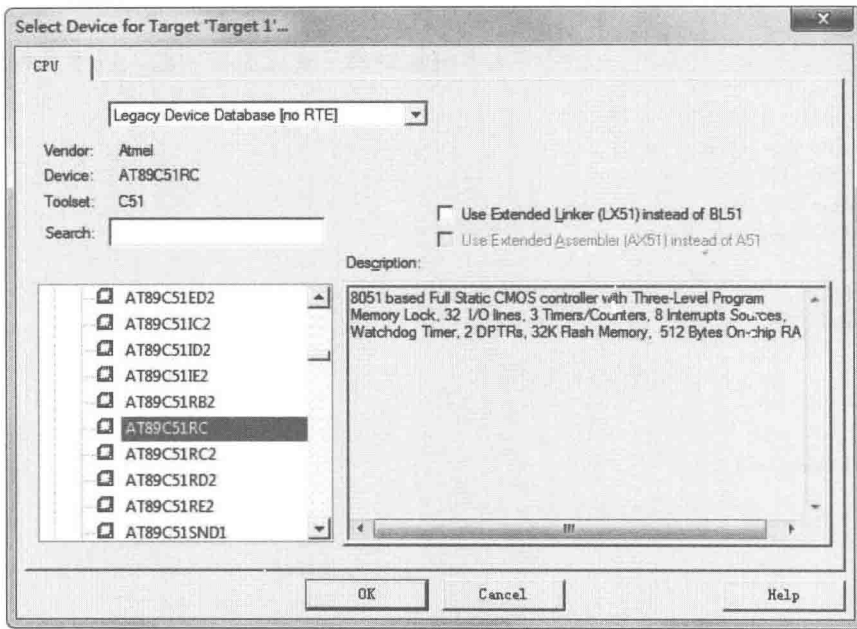


图 1-8 “Select Device for Target 'Target 1'...” 对话框

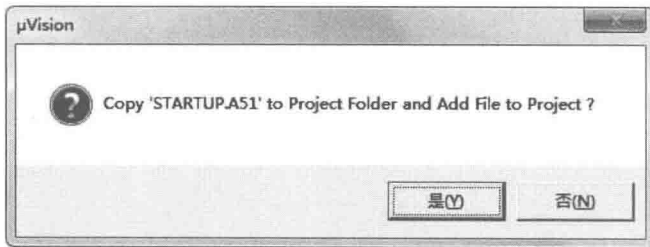
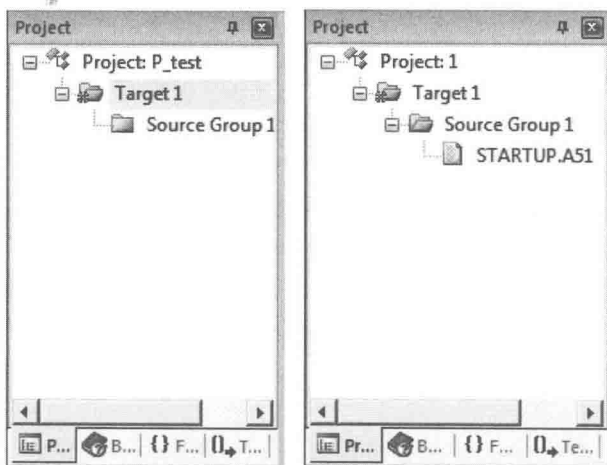



图 1-9 询问是否添加启动代码对话框



(a) 未添加启动代码


(b) 添加启动代码


图 1-10 是否添加启动代码的区别

3) **新建 C51 源程序文件** 创建新的项目后, 执行菜单命令“File”→“New”, 或者在工具栏中单击图标, 将打开一个空的文本编辑窗口。在此窗口中输入以下源程序代码:

```
#include "reg52. h"
#define uint unsigned int
sbit BZ = P3^7;
sbit key = P1^0;
void delays(uint ms)
{
    uint i;
    while(ms --)
    {
        for(i = 0; i < 120; i ++);
    }
}
void main(void)
{
    while(1)
    {
        if(key == 0)
        {
            BZ = 0x0;
            delays(10);
            BZ = 0x1;
            delays(50);
            P0 = 0xFF;
        }
        else
        {
            P0 = ~ P0;
            delays(500);
        }
    }
}
```

源程序输入好后, 执行菜单命令“File”→“Save”, 弹出“保存”对话框。在此对话框中输入保存的文件名称, 保存时后缀名为 .c。在项目窗口的“Target1”→“Source Group 1”上单击鼠标右键, 在弹出的菜单中选择“Add Existing Files to Group 'Source Group 1'”, 然后选择刚才所保存的源程序代码文件, 并单击“ADD”按钮, 即可将其添加到项目中, 如图 1-11 所示。

4) **编译文件** 添加源程序文件后, 执行菜单命令“Project”→“Build target”, 或者在工具栏中单击图标, 进行源程序的编译。编译完成后, μ Vision5 将会在输出窗口 (Output Window) 的编译页 (Build) 中显示相关信息。如果编译的程序有语法错误, 双击错误信息, 光标将会停留在 μ Vision5 文本编辑窗口中出现该错误或警告的源程序位置上。修改好源程序代码后, 再次执行菜单命令“Project”→“Build target”, 或者在工具栏中单击图标, 对源程序重新进行编译。

5) **HEX 文件的生成** 写入 51 系列单片机中的文件一般为 .HEX 文件。要得到 .HEX 文件, 在 Keil 中需进行相关设置。执行菜单命令“Project”→“Options for Target 'Target 1'”, 或者在工具栏中单击图标, 然后在弹出的“Options for Target 'Target 1'”对话框中选择“Output”选项卡, 选中“Create HEX File”选项即可, 如图 1-12 所示。