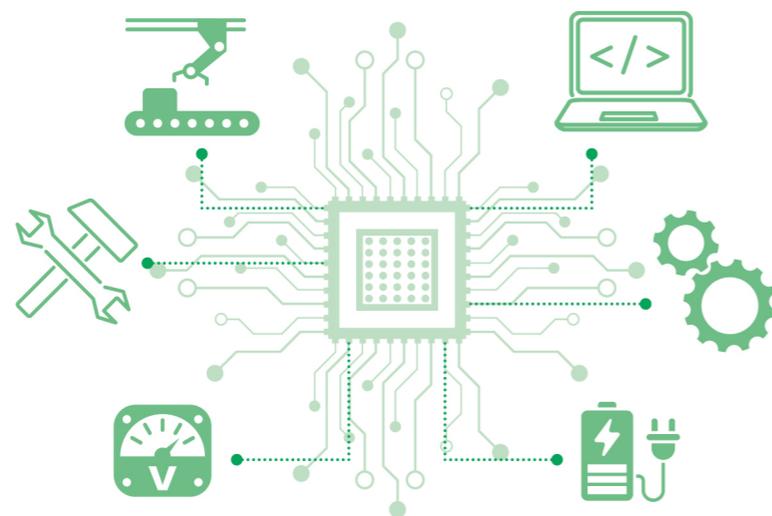


巍巍交大 百年书香
www.jiaodapress.com.cn
bookinfo@sjtu.edu.cn



策划编辑 刘子嘉
责任编辑 胡思佳
封面设计 碧 君



单片机技术及应用



扫描二维码
关注上海交通大学出版社
官方微信

ISBN 978-7-313-32040-7



9 787313 320407 >

定价: 48.00元

高等职业教育机电系列精品教材
河北省职业院校在线精品课程配套教材

单片机技术及应用

主编 曹月真



上海交通大学出版社

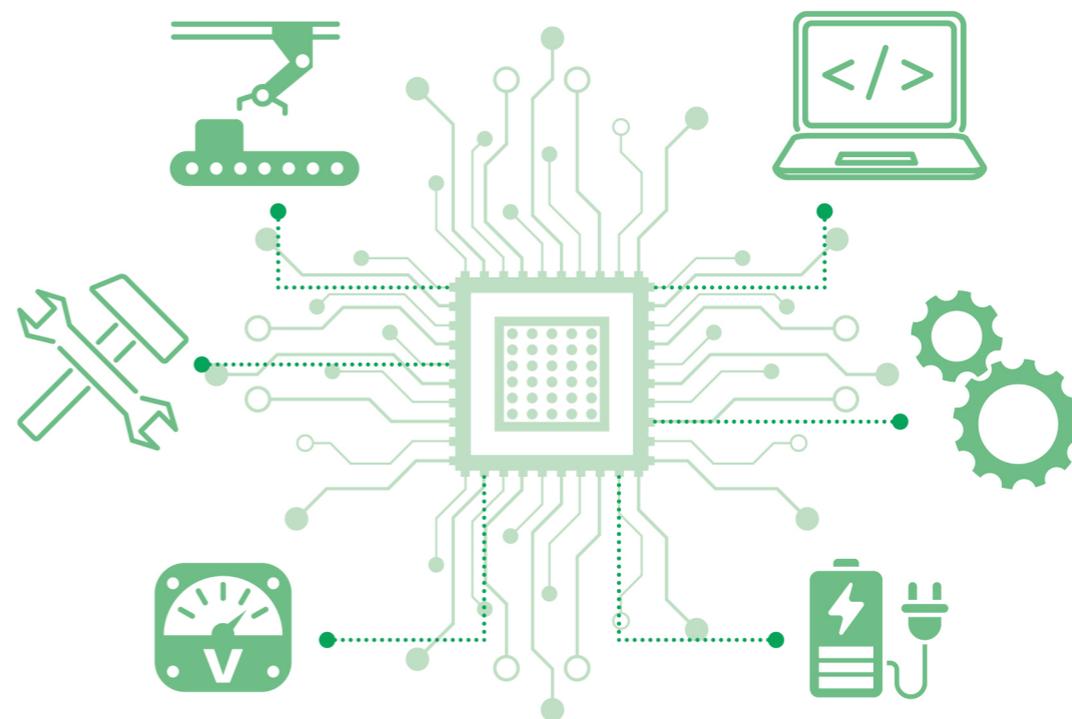
高等职业教育机电系列精品教材

河北省职业院校在线精品课程配套教材

单片机技术及应用

主编 曹月真

主审 李 敏



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

高等职业教育机电系列精品教材

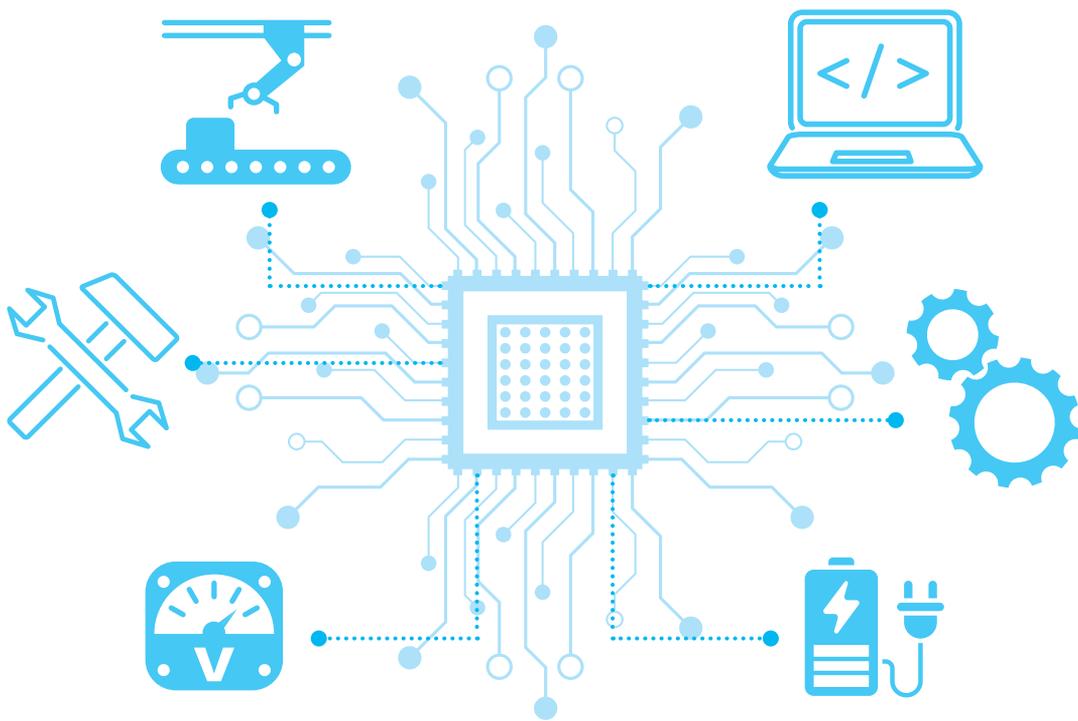
河北省职业院校在线精品课程配套教材

单片机技术及应用

主 编 曹月真

副主编 薛鹤娟 王素芹

主 审 李 敏



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

内容提要

本书共包含九个课题,分别为单片机的初步认知、交通信号灯的设计与制作、按键与数码管的应用设计、中断系统的应用、LED点阵显示设计、你来我往双机通信、液晶显示系统设计、可控的电机、模数和数模转换。

本书既可作为高等职业院校的应用电子技术、电子信息工程技术、机电一体化技术、工业机器人技术、无人机应用技术等专业相关课程的教学用书,也可作为对单片机技术感兴趣的自学者和工程技术人员的参考书。

图书在版编目(CIP)数据

单片机技术及应用 / 曹月真主编. -- 上海: 上海交通大学出版社, 2025. 7. -- ISBN 978-7-313-32040-7
I. TP368. 1
中国国家版本馆 CIP 数据核字第 2024J5N273 号

单片机技术及应用

DANPIANJI JISHU JI YINGYONG

主 编: 曹月真

出版发行: 上海交通大学出版社

邮政编码: 200030

印 制: 大厂回族自治县聚鑫印刷有限责任公司

开 本: 787 mm×1 092 mm 1/16

字 数: 328 千字

版 次: 2025 年 7 月第 1 版

书 号: ISBN 978-7-313-32040-7

定 价: 48.00 元

地 址: 上海市番禺路 951 号

电 话: 021-64071208

经 销: 全国新华书店

印 张: 13.5

印 次: 2025 年 7 月第 1 次印刷

电子书号: ISBN 978-7-89564-370-3

版权所有 侵权必究

告读者: 如您发现本书有印装质量问题请与印刷厂质量科联系

联系电话: 0316-8836866

单片机技术及应用是高等职业院校电类专业的一门实用性很强的技能实践课程。本书结合高等职业院校对学生的技能要求,采用任务导入、任务分析、任务实施的模式组织内容,将知识点和技能点融为一体,采用教、学、做一体化的教学模式,充分体现了理实结合的职业教育特色。

衡水职业技术学院“单片机技术及应用”课程组在河北省在线精品课程建设项目的基礎上,依据《国家职业教育改革实施方案》和《职业院校教材管理办法》等文件精神,结合多年教学改革的成果与经验,编写了在线课程“单片机技术及应用”的配套教材。本书的编写特色如下。

1. 教学内容紧密对接专业标准

编者对教学内容进行了重构,保证教学内容适度、够用,避免了难度过大、内容冗杂。

2. 教学设计充分体现职教特点

本书采用了大量实际案例,做到步步有任务,节节有训练。每个知识点都设有相应的实操,做到教、学、做有机统一,非常适合学生学习和实践。

3. 教学案例紧密联系生活实际

本书选取的案例贴近生产生活实际,任务仿真和实操也比较容易上手,使枯燥的理论学习变得生动形象,利于学生理解和掌握。

4. 任务评价注重学生职业素养

本书采用对学生完成的任务进行评价的方式,并利用详细的学生任务评价表将知识、技能、职业素养、合作精神都融入评价范围,旨在促进学生职业能力和创新精神的全面发展,体现了高等职业院校育人的特点。

5. 将 C 语言学习有机融入教学案例

本书采用 C 语言编程,将 C 语言学习融入教学案例中,以单片机为平台进行 C 语言教学,使两者相得益彰。

6. 任务化设计突出实用性

本书中引用了很多单片机应用的小实例,并以任务引入、任务目标、任务分析、知识链接、任务实施、任务评价和任务拓展的形式展现。书中任务涵盖了 C51 程序设计的基本语法、语句和应用方法,涉及的知识点由易到难、循序渐进,突出可行性和实用性。

7. 数字教学资源助力课堂教学

本书依托“智慧职教”教学平台,形成了立体化的数字教学资源,包括微课视频、PPT、练

习题库、配套软件等。学生可登录平台进行在线学习,也可通过扫描书中的二维码观看微课视频学习。本书配套的教学资源可在平台的课程页面在线浏览或下载,既方便学生进行自学,也方便教师采用“线上线下”融合的教学模式,提高信息化教学水平。

本书参考学时为 64 学时,具体安排见下表。实际使用时,不同学校可根据具体教学情况增减学时。

课 题	课 程 内 容	理 论 学 时	实 践 学 时
一	单片机的初步认知	2	4
二	交通信号灯的设计与制作	2	6
三	按键与数码管的应用设计	2	8
四	中断系统的应用	2	8
五	LED点阵显示设计	2	4
六	你来我往双机通信	2	4
七	液晶显示系统设计	2	4
八	可控的电机	2	4
九	模数和数模转换	2	4
总计		64	

本书由衡水职业技术学院曹月真担任主编,衡水职业技术学院薛鹤娟和王素芹担任副主编。曹月真对本书的编写思路与大纲进行了总体策划,对全书统稿,并编写了课题一至课题八。薛鹤娟和王素芹协助完成了统稿工作,薛鹤娟编写了课题九,王素芹编写了本书内容提要、前言及思考与练习题。微课视频教学课程由曹月真、薛鹤娟、王素芹主讲。本书由衡水职业技术学院李敏教授主审。李敏教授在审稿过程中提出了许多宝贵的意见和建议,在此深表感谢!

由于编者水平有限,书中难免存在不足之处,敬请读者批评指正。

编 者

目录

课题一	单片机的初步认知	1
	任务一 搭建单片机最小系统	1
	任务二 单片机控制 LED 点亮	13
	思考与练习题	24
课题二	交通信号灯的设计与制作	26
	任务一 单片机控制 LED 闪烁	26
	任务二 设计流水灯	33
	任务三 设计 LED 简易交通信号灯	42
	思考与练习题	48
课题三	按键与数码管的应用设计	49
	任务一 独立按键控制 LED 变化	49
	任务二 数码管静态显示	55
	任务三 数码管显示矩阵键盘键值	61
	任务四 数码管动态显示	67
	思考与练习题	75
课题四	中断系统的应用	77
	任务一 外部中断的应用	77
	任务二 设计简易频率计	87
	任务三 设计声光报警器	98
	思考与练习题	105

课题五	LED 点阵显示设计	108
	任务一 8×8 点阵引脚测试	108
	任务二 8×8 点阵显示静态图形	112
	任务三 8×8 点阵循环显示数字	119
	任务四 16×16 点阵显示汉字	124
	思考与练习题	130
课题六	你来我往双机通信	132
	任务一 PC 与单片机通信	132
	任务二 扩展 I/O 口	142
	任务三 双机通信	150
	思考与练习题	156
课题七	液晶显示系统设计	158
	任务一 设计 LCD1602 显示系统	158
	任务二 设计 LCD12864 显示系统	166
	思考与练习题	179
课题八	可控的电机	180
	任务一 可控的直流电机	180
	任务二 可控的步进电机	185
	思考与练习题	192
课题九	模数和数模转换	193
	任务一 设计呼吸灯	193
	任务二 设计数字电压表	200
	思考与练习题	209
	参考文献	210



课题一 单片机的初步认知

课题描述

单片机具有性能高、速度快、体积小、价格低、可重复编程等优点,被广泛应用于多个领域,如智能仪器仪表、工业监控、计算机网络和通信、家用电器、医疗设备、汽车设备、军用机械等。单片机系统包括硬件电路设计和软件设计两部分,采用 C 语言编程,其硬件仿真平台为 Proteus 软件。

任务一 搭建单片机最小系统

任务引入

单片机最小系统也称单片机最小应用系统,是指采用最少数元器件组成的能够使单片机工作的系统。

任务目标

知识目标

- (1)了解单片机的硬件结构。
- (2)认识单片机的引脚。
- (3)学会搭建单片机最小系统的硬件电路。

技能目标

- (1)能熟练查阅常用电子元器件和芯片的规格、型号、使用方法等技术资料。
- (2)能用 Proteus 仿真软件对单片机最小系统的硬件电路进行仿真。

素养目标

- (1)培养严谨的工程思维与问题解决能力。
- (2)提升自主学习与持续学习的能力。

任务分析

单片机最小系统主要由电源电路、复位电路和时钟电路组成。在搭建单片机最小系统的硬件电路之前,需要先认识单片机的内部结构以及引脚图。

一、认识单片机

1. 什么是单片机

单片机是使用超大规模集成电路技术把具有数据处理能力的中央处理器、随机存储器(数据存储器)、只读存储器(程序存储器)、I/O口、中断系统和定时/计数器等集成到一块硅片上的微型计算机系统,在工业控制、航空航天等领域被广泛应用。根据中央处理器的处理能力,单片机可分为4位、8位、16位、32位单片机。本书以8位基本型单片机 AT89C51 为研究对象来详细介绍单片机的内部结构、工作原理及应用。AT89C51 单片机的外形如图 1-1 所示。



微课堂

单片机的认知



图 1-1 AT89C51 单片机的外形

2. 单片机的内部结构

AT89C51 单片机的内部结构如图 1-2 所示。

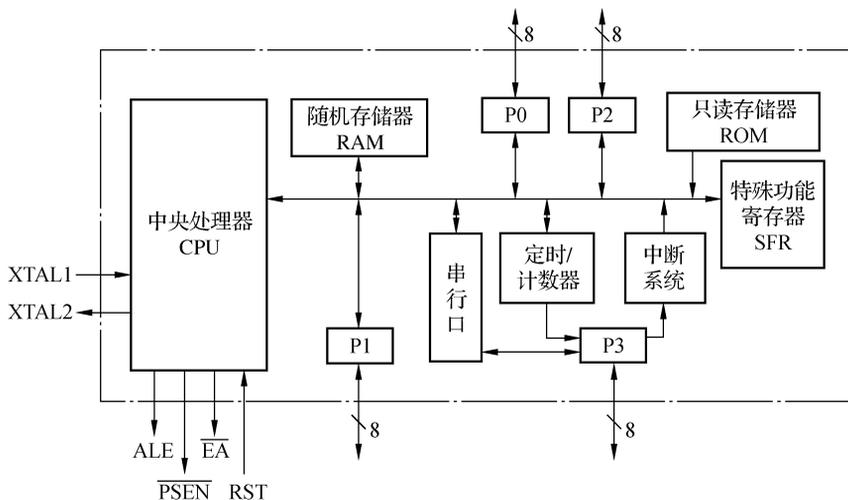


图 1-2 AT89C51 单片机的内部结构

1) 中央处理器

中央处理单元又称 CPU, 是单片机的核心, 它由运算器和控制器等部件组成。

2) 存储器

AT89C51 单片机的程序存储器和数据存储器有各自的寻址系统。程序存储器用来存放程序和表格常数, 通过 16 位地址总线进行寻址, 可寻址的地址空间为 64 KB。数据存储器常用来存放程序运行中所需要的常数或变量, 分为内部 RAM 和外部 RAM, 最大可扩展的地址空间同样为 64 KB。

3) 定时/计数器

AT89C51 单片机内部有两个 16 位可编程的定时/计数器。

4) 中断系统

AT89C51 单片机的中断系统共有五个中断源, 包括两个外部中断、两个定时/计数器中断和一个全双工的异步串行口中断。

5) 时钟电路

AT89C51 单片机内部的时钟电路为单片机提供时钟脉冲。单片机工作时的晶体振荡器(晶振)频率通常为 11.059 2 MHz 或 12 MHz。

6) I/O 口

AT89C51 单片机有四个并行 I/O 口, 分别为 P0、P1、P2、P3, 每个 I/O 口有八个引脚, 能够实现数据的并行输入和输出。

3. 单片机的引脚

AT89C51 单片机采用 40 引脚的双列直插封装方式。图 1-3 所示为 AT89C51 单片机的引脚图。

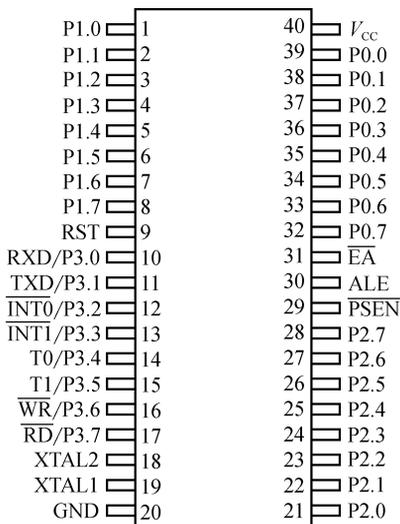


图 1-3 AT89C51 单片机的引脚图

AT89C51 单片机各引脚功能如下。

1) 电源引脚

GND(某些品牌单片机的此引脚名称为 V_{SS}): 接地。

V_{CC} :接+5 V 电源。

2) 时钟引脚

XTAL1:内部振荡电路反相放大器的输入端。当单片机使用内部振荡器时,此引脚外接晶振的一个引脚;当单片机使用外部振荡器时,此引脚接外部振荡源。

XTAL2:内部振荡电路反相放大器的输出端。当单片机使用内部振荡器时,此引脚外接晶振的另一个引脚;当单片机使用外部振荡器时,此引脚悬空。

3) 控制引脚

RST:当单片机运行时,此引脚上出现持续时间大于两个机器周期的高电平(由低到高跳变),将使单片机复位。

ALE:单片机正常运行时,此引脚用于将地址的低字节锁存到外部地址锁存器中,ALE 引脚以不变的频率(时钟振荡频率的 1/6)周期性地发出正脉冲。因此,它可用于对外输出时钟,也可用于定时目的。但要注意,每当访问外部数据存储器时,系统将跳过一个 ALE 脉冲。

对于可擦可编程只读存储器(EPROM)型单片机,在对 EPROM 编程期间,此引脚接收编程脉冲。

\overline{PSEN} :外部程序存储器读选通信号输出端。当系统从外部程序存储器读取指令(或数据)时, \overline{PSEN} 接外部程序存储器的输出允许端。

\overline{EA} : \overline{EA} 为内部程序存储器和外部程序存储器选择引脚。当 \overline{EA} 为高电平时,系统访问内部程序存储器,如果 PC 值大于 0FFFH,系统将自动转向地址以 1000H 开始的外部程序存储器。当 \overline{EA} 为低电平时,系统直接访问地址为 0000H~FFFFH 的外部程序存储器。

4) 输入/输出引脚

P0 口(P0.0~P0.7):一个八位漏极开路型双向 I/O 口。当系统访问外部存储器时,P0 口是分时传送数据的低八位地址总线和数据总线。P0 口作为普通 I/O 口使用时需外接上拉电阻,能驱动八个 LS 型 TTL 负载。

P1 口(P1.0~P1.7):一个带有内部上拉电阻的八位准双向 I/O 口,能驱动四个 LS 型 TTL 负载。

P2 口(P2.0~P2.7):一个带有内部上拉电阻的八位准双向 I/O 口。当系统访问外部存储器时,P2 口输出高八位地址。P2 口可以驱动四个 LS 型 TTL 负载。

P3 口(P3.0~P3.7):一个带有内部上拉电阻的八位准双向 I/O 口,能驱动四个 LS 型 TTL 负载。P3 口还具有第二功能,P3 口各引脚第二功能的定义见表 1-1。

表 1-1 P3 口各引脚第二功能的定义

P3 口引脚	第二功能定义	功能说明
P3.0	RXD	串行输入口
P3.1	TXD	串行输出口
P3.2	$\overline{INT0}$	外部中断 0 输入端
P3.3	$\overline{INT1}$	外部中断 1 输入端

续表

P3 口引脚	第二功能定义	功能说明
P3.4	T0	T0 外部计数脉冲输入端
P3.5	T1	T1 外部计数脉冲输入端
P3.6	$\overline{\text{WR}}$	外部 RAM 写选通脉冲输出端
P3.7	$\overline{\text{RD}}$	外部 RAM 读选通脉冲输出端

二、时钟与时序

单片机系统中的各个部件在一个统一的时钟脉冲控制下有序地进行工作,时钟电路是单片机系统最基本、最重要的电路。

1. 时钟

AT89C51 单片机内部有一个高增益反相放大器,引脚 XTAL1 和 XTAL2 分别是该放大器的输入端和输出端。在引脚 XTAL1 和 XTAL2 两端跨接晶振或陶瓷振荡器可构成稳定的自激振荡电路,该振荡电路的输出信号可直接送入内部时序电路。AT89C51 单片机的时钟信号可通过两种方式产生,即内部时钟方式和外部时钟方式。

1) 内部时钟方式

内部时钟方式是指由单片机内部的高增益反相放大器和外部跨接的晶振(或陶瓷振荡器)、微调电容构成的时钟电路产生时钟信号,其工作原理如图 1-4 所示。

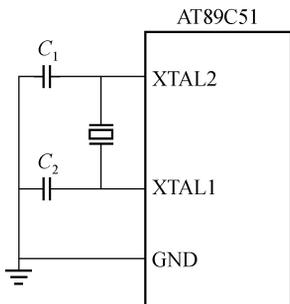


图 1-4 内部时钟方式

当 XTAL1、XTAL2 外接晶振时, C_1 、 C_2 的值通常为 30 pF 左右; C_1 、 C_2 对晶振的频率有微调作用,晶振的频率可在 1.2~12 MHz 范围选择。由于内部时钟方式的时钟电路接线简单,因此单片机系统大多采用这种方式。

内部时钟方式产生的时钟信号的频率就是晶振的固有频率,常用 f_{osc} 表示。如果选择 12 MHz 晶振,则 $f_{\text{osc}} = 12 \times 10^6$ Hz。

2) 外部时钟方式

外部时钟方式是指利用单片机外部时钟电路产生时钟信号。外部时钟方式产生的时钟

信号直接由 XTAL1 引脚输入单片机,此时 XTAL2 引脚悬空,如图 1-5 所示。

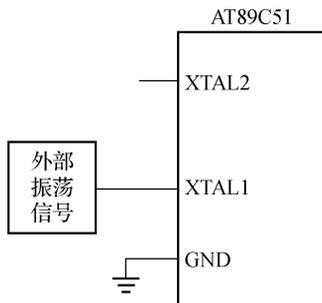


图 1-5 外部时钟方式

2. 时序

时序是用定时单位来描述的,AT89C51 单片机的时序单位有四个,分别是节拍、状态、机器周期和指令周期。

当执行指令时,CPU 都是按照一定顺序进行的。由于指令的字节数不同,因此 CPU 执行指令所需要的时间也就不同。即使是字节数相同的指令,CPU 执行指令所需要的时间也会因执行操作的不同而有很大差别。

1) 节拍与状态

振荡脉冲的周期定义为节拍,为方便描述,一般用 P 表示。在 AT89C51 单片机中,一个状态等于两个节拍,用 S 表示。前半周期相应的节拍定义为 P1,后半周期对应的节拍定义为 P2。

2) 机器周期

完成一个最基本操作(读或写)所需要的时间称为机器周期。AT89C51 单片机的机器周期是固定的,一个机器周期有 6 个状态,表示为 S1~S6,而一个状态包含两个节拍,那么一个机器周期就有 12 个节拍,可以记为 S1P1,S1P2,⋯,S6P1,S6P2。一个机器周期共包含 12 个振荡脉冲,即机器周期就是振荡脉冲的 12 分频。若单片机采用 6 MHz 的时钟频率,则一个机器周期为 2 μs;若单片机采用 12 MHz 的时钟频率,则一个机器周期为 1 μs。

3) 指令周期

指令周期是执行一条指令所需要的时间,一般由若干个机器周期组成。指令不同,所需要的机器周期数也不同。对于一些简单的单字节指令,指令周期可能和机器周期时间相同;而对于一些比较复杂的指令,如乘、除运算等则需要 4 个机器周期才能完成,这时指令周期大于机器周期。

通常,一个机器周期即可完成的指令称为单周期指令,两个机器周期才能完成的指令称为双周期指令。AT89C51 单片机中的大多数指令都是单周期或双周期指令,只有乘、除运算为四周期指令。AT89C51 单片机的指令时序图如图 1-6 所示。

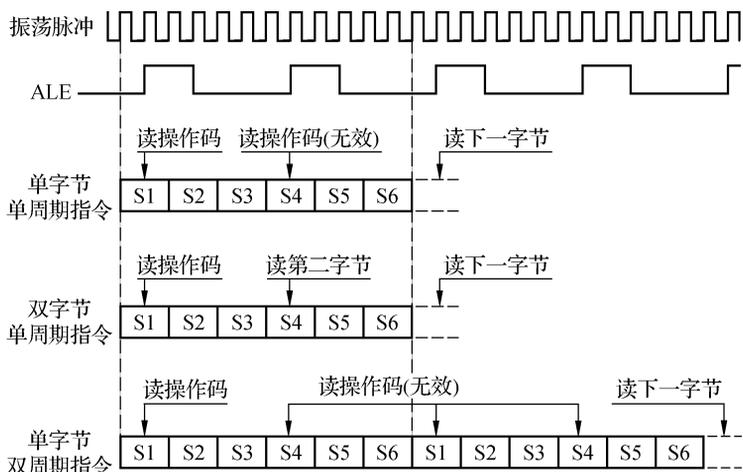


图 1-6 AT89C51 单片机的指令时序图

三、复位系统

单片机在上电时一般都需要进行一次复位操作,以便使芯片内的一些部件处于一个确定的初始状态。复位是一种很重要的操作,一般部件本身不具有自动上电复位功能,需要借助外部复位电路提供的复位信号才能进行复位操作。除正常初始化之外,当程序运行出错或操作错误使系统处于锁死状态时,也需要通过复位按键使 AT89C51 单片机摆脱“跑飞”或锁死状态而重新启动。

AT89C51 单片机的第 9 脚(RST)为复位引脚。单片机上电后,时钟电路开始工作,只要 RST 引脚上持续出现大于两个机器周期的高电平即可引起单片机执行复位操作。单片机复位后,PC=0000H,CPU 从程序存储器的 0000H 开始执行指令。单片机中的特殊功能寄存器(SFR)的复位状态表如表 1-2 所示。

表 1-2 特殊功能寄存器复位状态表

特殊功能寄存器	复位状态	特殊功能寄存器	复位状态
PC	0000H	TMOD	00H
ACC	00H	TCON	00H
B	00H	TH0	00H
PSW	00H	TL0	00H
SP	07H	TH1	00H
DPTR	0000H	TL1	00H
P0~P3	FFH	SCON	00H
IP	XXX0000B	SBUF	不定
IE	0XX0000B	PCON	0XXX000B

单片机的外部复位电路有上电复位和按键复位两种。

1) 上电复位电路

最简单的上电复位电路由电容和电阻串联构成,如图 1-7 所示。图中标示的参数为晶振为 6 MHz 时各元器件的参数。

2) 按键复位电路

图 1-8 所示为按键复位电路,图中标示的参数是晶振为 6 MHz 时各元器件的参数。当按下复位按键后,电容迅速通过 R_2 放电,放电结束时的 $V_R = (R_1 \times V_{CC}) / (R_1 + R_2)$ 。由于 R_1 远大于 R_2 ,所以 V_R 非常接近 V_{CC} ,这使 RST 引脚处于高电平状态。松开复位按键后,过程与上电复位电路相同。

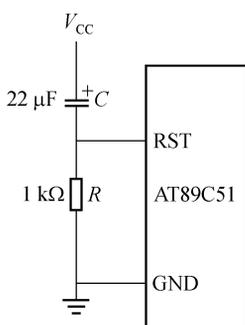


图 1-7 上电复位电路

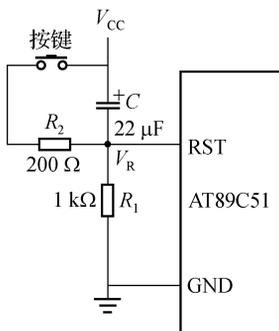


图 1-8 按键复位电路

任务实施

参照图 1-9 搭建单片机最小系统的硬件电路回答问题。

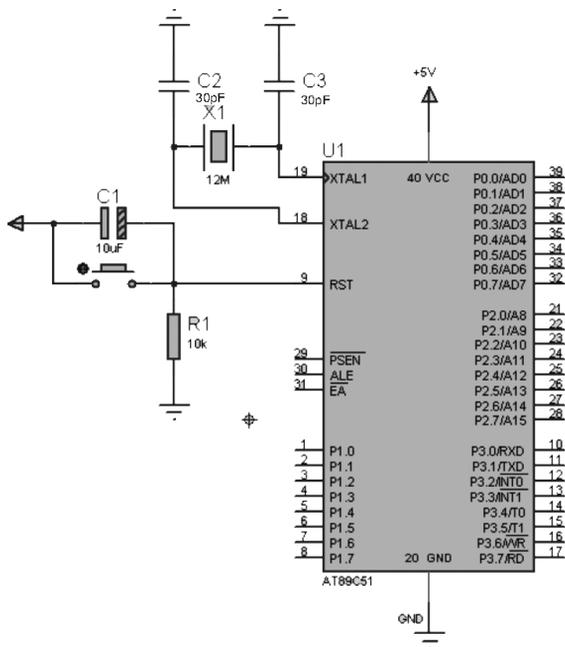


图 1-9 单片机最小系统的硬件电路

- (1)通常情况下单片机的工作电压是多少?
- (2)9号引脚为高电平还是低电平?系统可以实现复位吗?
- (3)组成单片机最小系统的元器件有哪些?

任务评价

学生任务评价表如表 1-3 所示。

表 1-3 学生任务评价表

任务时间				
任务名称	评分内容及标准	满分	得分	评价
搭建单片机 最小系统	1. 课堂状态(非常好 10分,一般 5分,不好 3分) 2. 任务完成情况(非常好 10分,一般 5分,不好 3分) 3. 任务完成的主动性(积极 10分,不积极 5分)	30分		自我评价:
	1. 准时参加任务实施(准时 10分,不准时 5分) 2. 积极参与任务交流讨论(积极 10分,不积极 5分) 3. 准确说出任务设计流程与方法(准确 10分,不准确 5分) 4. 团队合作情况(非常好 10分,一般 5分,不好 3分)	40分		小组评价:
	1. 电路设计与绘制情况(非常好 10分,一般 5分,不好 3分) 2. 遵守安全操作规程(遵守 10分,不遵守 5分) 3. 工作台及场地卫生情况(非常好 10分,一般 5分,不好 3分)	30分		教师评价:
总计				

任务拓展

Proteus 软件的使用步骤和方法如下。

1. 启用 Proteus 编辑环境

在计算机中安装好 Proteus 软件后,然后执行“开始菜单”→“所有程序”→“Proteus 7 Professional”命令,单击图标即可启动程序,启动界面如图 1-10 所示。



图 1-10 Proteus 启动界面

系统进入启动界面 2~3 s 后,弹出编辑操作 ISIS Professional 窗口,如图 1-11 所示。

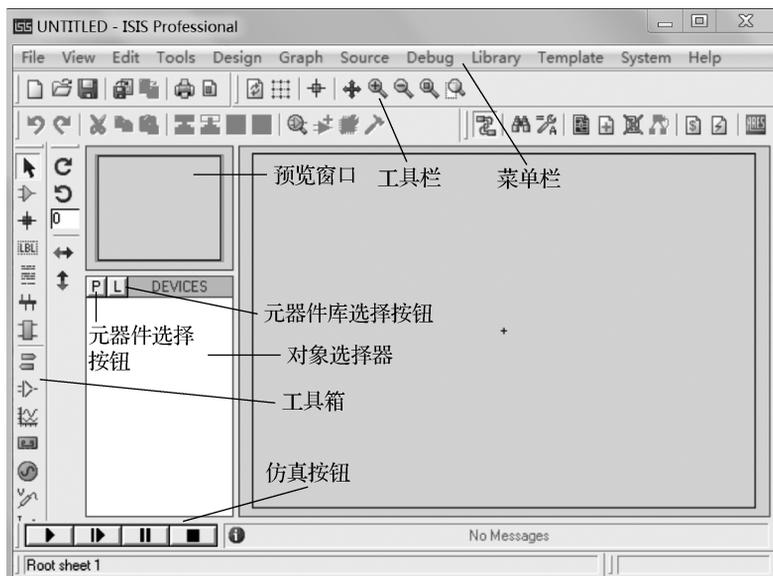


图 1-11 ISIS Professional 窗口

2. 选择元器件

在工具箱中单击元器件按钮图标后,系统才会弹出元器件及元器件库选择按钮,如图 1-12 所示。其中“P”为元器件选择按钮,“L”为元器件库选择按钮。

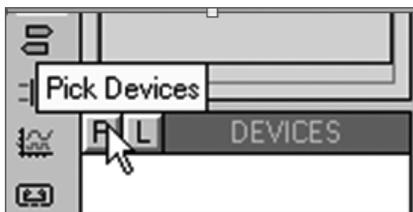


图 1-12 元器件及元器件库选择按钮

单击“P”按钮,弹出一个对象选择对话框,在此对话框的“Keywords”文本框中输入元器件名称,单击“OK”按钮就可以从库中选择元器件,并且所选元器件名称将在对象选择器中显示。

例如,在“Keywords”文本框中输入“at89C”,“Results”选项区将显示出若干个匹配的 AT89C51,如图 1-13 所示。然后双击需要的元器件就可以将其添加到对象选择器中,在其列表中可以看到选中元器件的名称。

3. 放置、移动、旋转、删除元器件

1) 放置元器件

经过前面两步将所需要的元器件添加到对象选择器后,单击要放置的元器件,如 LED-RED,此时蓝色条(见图 1-14 框线所示)出现在该元器件名字上,再在原理图编辑窗口任意位置单击,就将 LED-RED 放置到原理图编辑窗口中了。

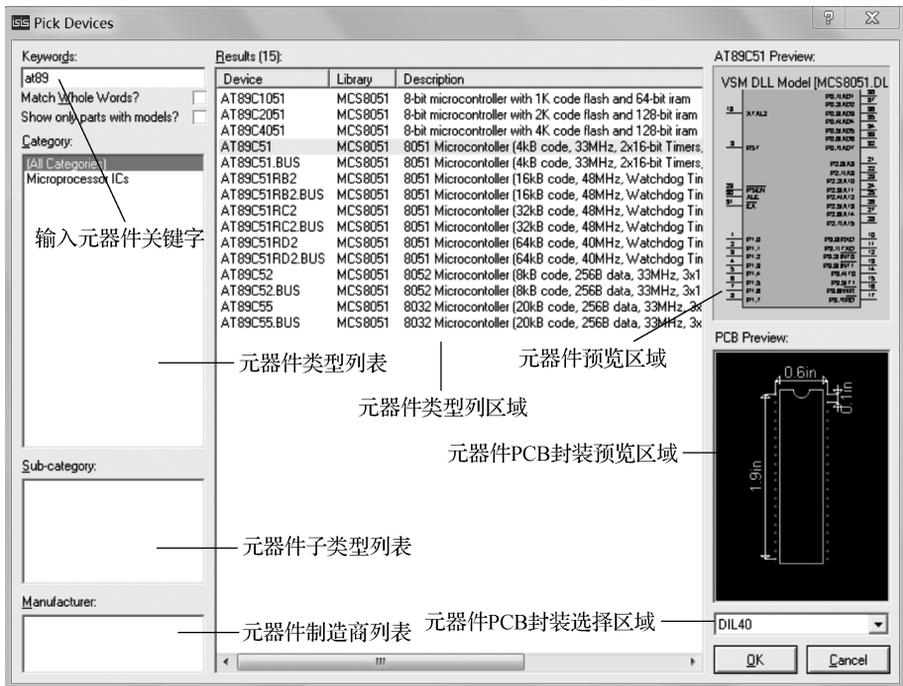


图 1-13 元器件浏览对话框

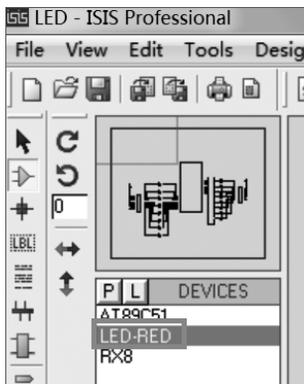


图 1-14 元器件选中状态

2) 移动元器件

将元器件放置到原理图编辑窗口中后,若要移动元器件,应先单击元器件,使元器件处于选中状态(默认情况下为红色),再按住鼠标左键拖动,元器件就会跟随鼠标移动,到达合适位置时,松开鼠标左键即可。

3) 旋转元器件

放置元器件前,先单击要放置的元器件,再单击方向工具栏中相应的转向按钮即可旋转元器件,然后在原理图编辑窗口中单击,就将一个已经更改方向的元器件放置到原理图编辑窗口中了。

若要在原理图编辑窗口中更改元器件方向,则应右击选中该元器件,在弹出的对话框中

输入要旋转的角度即可。

4) 删除元器件

若要在原理图编辑窗口中删除元器件,则右键双击该元器件就可删除元器件,或者先单击选中该元器件,再按下 Delete 键也可删除元器件。

通过放置、移动、旋转、删除操作,可将各元器件放置在原理图编辑窗口的合适位置。

4. 放置电源、地

1) 放置电源

单击工具箱中的元器件终端图标,在对象选择器中单击“POWER”,其选中状态如图 1-15 所示,再在原理图编辑窗口的合适位置单击就将电源放置在原理图编辑窗口中了。

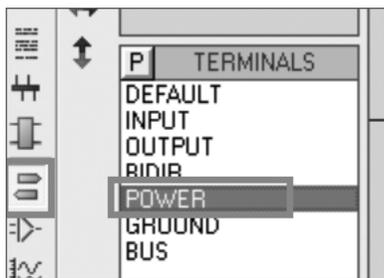


图 1-15 电源选中状态

2) 放置地

单击工具箱中的元器件终端图标,在对象选择器中单击“GROUND”,再在原理图编辑窗口的合适位置单击就可将地放置在原理图编辑窗口中了。

5. 布线

原理图编辑窗口中没有专门的布线按钮,但系统默认自动布线按钮有效,因此可直接画线。

1) 在两个对象之间连线

使光标靠近一个对象的引脚末端,在该处单击,移动鼠标使光标靠近另一个对象的引脚末端,在该处再次单击,就可以画一条连线。如果想手动设定走线路径,应拖动鼠标在想要拐弯处单击,到达画线端的另一端再次单击,就可画好一条连线。在拖动鼠标的过程中,按住 Ctrl 键,在画线端的另一端单击,即可手动画一条任意角度的连线。

2) 移动画线

右击画线,弹出菜单,选择“Drag Wire”选项,此时画线变成黄色。拖动鼠标,该线就跟随鼠标移动。若要同时移动多条画线,应先框选这些画线,再单击移动按钮,将画线拖动到合适的位置处单击就可以改变画线的位置。

3) 总线及分支线的画法

画总线:可以把已经画好的单线设置为总线,选中该线,右击,在弹出的菜单中选择“Edit Wire Style”选项,在“Global Style”下拉列表中选择“WIRE”,然后单击“OK”按钮即可,如图 1-16 所示。

画分支线:使光标靠近一个对象的引脚末端,在该处单击,然后拖动鼠标,在总线上单击即可画好一条分支线。若要使分支线与总线成任意角度,要同时按住 Ctrl 键,再在总线上单击。



图 1-16 编辑画线样式对话框

6. 设置、修改元器件属性

在需要修改的元器件上右击,然后在弹出的菜单中选择“Edit Properties”选项或按快捷键 Ctrl+E,弹出“Edit Component”对话框,在此对话框中可设置元器件属性。如果需要成组设置元器件属性,可以使用属性分配功能,用鼠标左键框选所需要设置的所有的元器件,执行“Tools”→“Property Assignment Tool”命令,或者按快捷键 A,弹出图 1-17 所示的属性分配对话框,然后在此对话框中可设置所选元器件的属性。

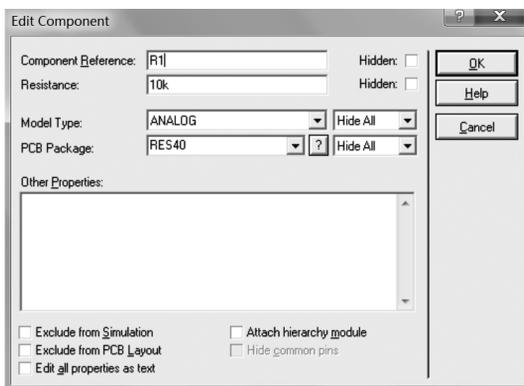


图 1-17 属性分配对话框

任务二 单片机控制 LED 点亮

任务引入

单片机最小系统硬件电路搭建好之后就可以利用单片机控制输入输出设备了,最简单的输出设备是 LED。本任务的目标就是利用单片机控制 LED 点亮。

任务目标

知识目标

- (1) 学会分析单片机 I/O 口高低电平状态对应的 LED 亮灭情况。

- (2) 学会搭建单片机控制 LED 点亮的硬件电路。
- (3) 学会使用单片机任意 I/O 口控制 LED 亮灭。

技能目标

- (1) 能用 Keil 软件编写程序并成功编译生成目标文件。
- (2) 能用 Proteus 仿真软件对单片机控制 LED 点亮的硬件电路进行仿真。

素养目标

- (1) 参与小组讨论, 树立团队合作意识, 提高沟通技巧。
- (2) 培养自主学习能力和持续学习的能力。

任务分析

本任务的仿真环节在 Keil 和 Proteus 软件上完成, 使用 Keil 软件编写控制程序, 将编译好的目标文件烧录到单片机中, 实现点亮 LED 的功能。实物的搭建可在 AT89C51 单片机实训板上完成。

知识链接

一、Keil 软件的使用方法

Keil 软件是众多单片机应用开发软件之一, 它集编辑、编译、仿真于一体, 支持汇编语言和 C 语言的程序设计, 界面友好, 易学易用。下面以“点亮一盏灯”为例, 说明 Keil 软件的基本使用方法。

1. 创建一个工程

- (1) 进入 Keil 软件的编辑界面后, 执行“Project”→“New Project”命令, 如图 1-18 所示。

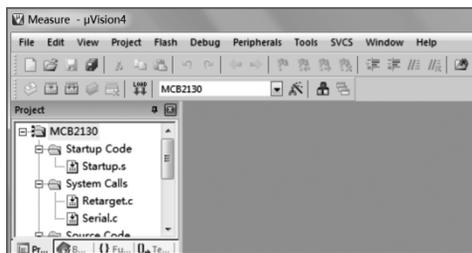


图 1-18 新建工程

(2) 在弹出的“Create New Project”对话框中选择要保存的路径, 输入工程文件的名称。例如, 将工程文件保存到 C51 目录并命名为“点亮一盏灯”, 如图 1-19 所示, 然后单击“保存”按钮。

(3) 保存工程文件后会弹出单片机型号选择对话框。Keil 软件几乎支持所有 AT89C51 内核的单片机, 用户可以根据需要选择不同型号的单片机。对话框左侧的“Data base”下拉列表里会显示出 AT89C51 系列单片机的生产厂商, 展开各生产厂商对应的选项, 即可看到该生产厂商生产的单片机型号。这里以 Atmel 公司的 AT89C51 为例来说明如何选择单片机型号。首先选择生产厂商“Atmel”, 展开“Atmel”选项后, 列表框显示 80C32E、AT48801

等多种单片机型号。在列表框中选择“AT89C51”选项,右边的“Description”文本框中就会出现对这款单片机的功能描述,如图 1-20 所示,然后单击“OK”按钮。

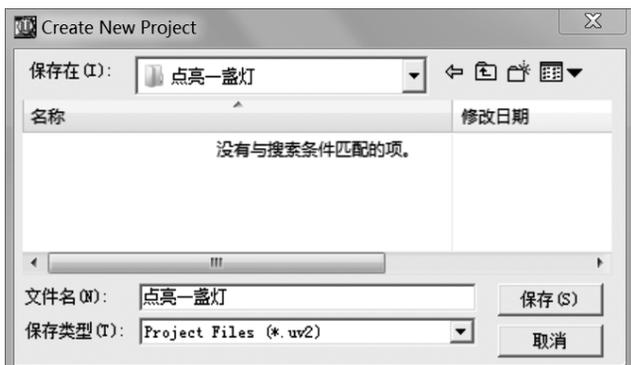


图 1-19 “Create New Project”对话框

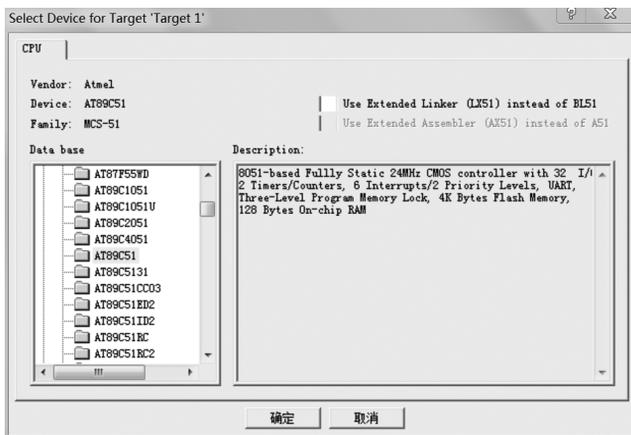


图 1-20 单片机型号选择对话框

(4)完成单片机型号选择后,系统进入图 1-21 所示工程创建完成界面。此时标题栏上显示的名称为“点亮一盏灯”,即创建的工程名称。

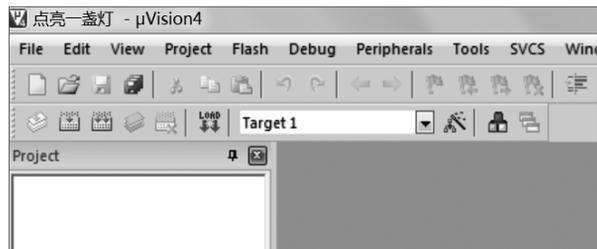


图 1-21 工程创建完成界面

2. 向工程中添加文件

(1)执行“File”→“New”命令,新建文件,如图 1-22 所示。新建文件后系统进入程序编写界面,如图 1-23 所示。

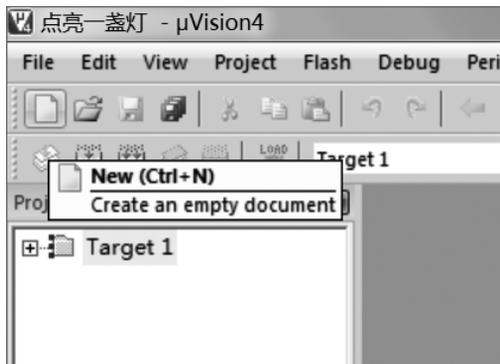


图 1-22 新建文件

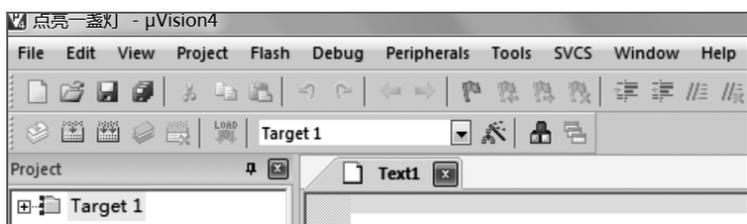


图 1-23 程序编写界面

(2)保存文件时,可执行“File”→“Save As”命令,系统会弹出“Save As”对话框,如图 1-24 所示。在“文件名”文本框中输入文件名,同时,必须输入正确的扩展名。

注意:如果选用 C 语言编写程序,则扩展名为“.c”;如果选用汇编语言编写程序,则扩展名为“.asm”。输入文件名后,单击“保存”按钮。



图 1-24 文件保存对话框

(3)回到程序编写界面后,展开“Target 1”选项,会出现“Source Group 1”选项,右击该选项,在图 1-25 所示快捷菜单中选择“Add Files to Group 'Source Group 1'...”选项,弹出“Add Files to Group 'Source Group 1'...”对话框,选择“text”文件,然后单击“Add”按钮将文件添加到工程中,如图 1-26 所示。

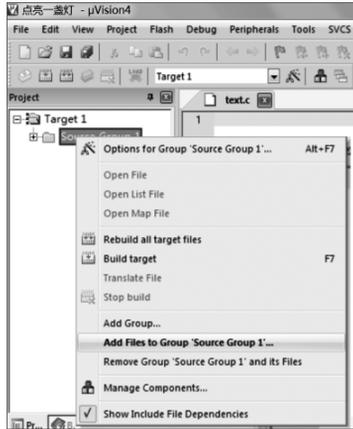


图 1-25 添加文件界面



图 1-26 添加文件对话框

这时在“Source Group 1”选项下多出一个子项“text.c”，如图 1-27 所示。子项的多少与所增加的程序的多少保持一致。

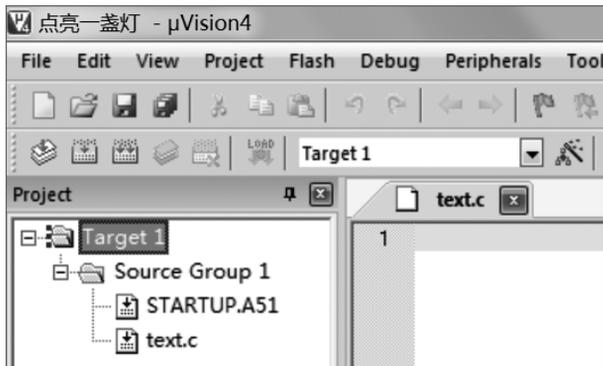


图 1-27 文件添加完成后的界面

3. 项目输出设置

右击图 1-28 所示项目文件夹“Target1”，弹出项目功能菜单，选择“Options for Target

'Target1'...'选项,弹出项目选项设置窗口。选择“Output”选项卡,如图 1-29 所示,选中“Create HEX File”复选框,就可以输出 HEX 文件到指定的路径中。

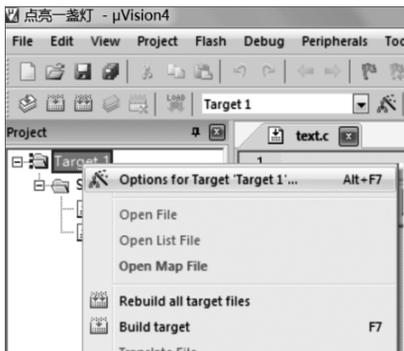


图 1-28 项目选项设置窗口

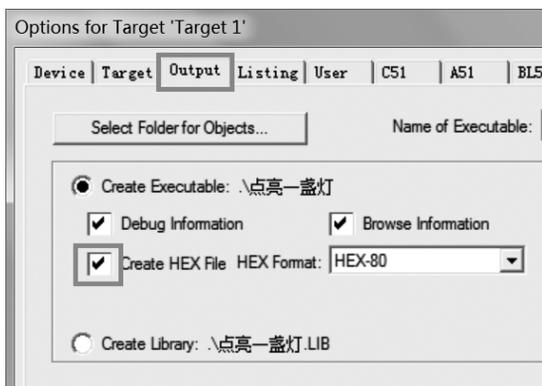


图 1-29 输出 HEX 文件界面

4. 编写 C 语言程序

在程序编辑文本框中编写 C 语言程序,如图 1-30 所示。在编写程序时,Keil 软件会自动识别关键字,并用不同的颜色提示用户加以注意,这样会减少编写错误,有利于提高编程效率。

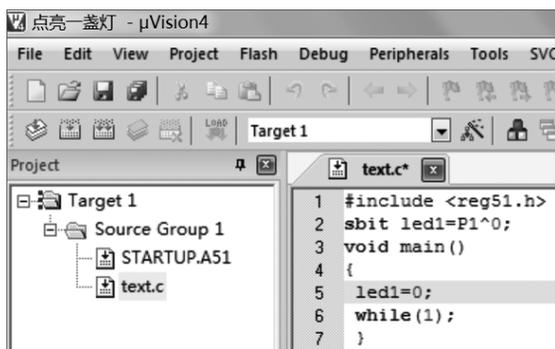


图 1-30 编写程序

5. 编译程序

执行“Project”→“Built Target”命令(使用快捷键 F7)或单击图 1-31 框线所示图标编译程序。当界面显示“creating hex file from “点亮一盏灯”...”时说明程序没有语法错误,成功生成了目标文件。程序编译成功界面如图 1-32 所示。

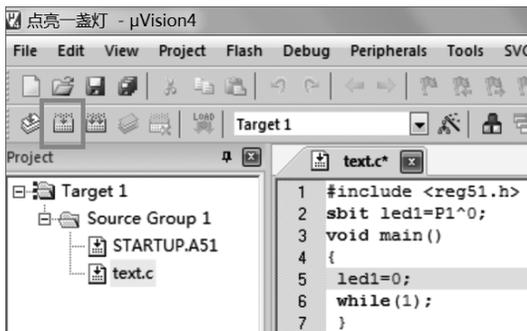


图 1-31 程序编译

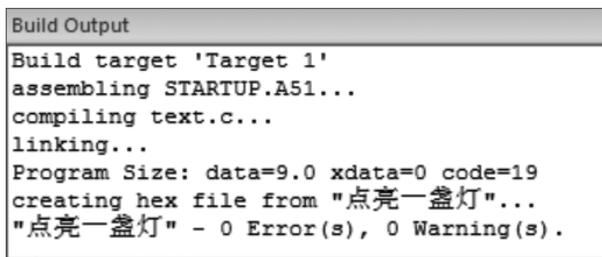


图 1-32 程序编译成功界面

以上操作是在 Keil 软件环境下创建一个完整工程的全过程,但这只是单片机系统的软件开发过程,接下来利用 Proteus 软件与 Keil 软件进行联调仿真。

二、Proteus 与 Keil 软件的联调仿真

在 Proteus 元件库中找到需要的元器件,连接好硬件电路,双击芯片弹出图 1-33 所示对话框。此时“Program File”一栏为空,单击文件夹图标,找到需要烧录的目标文件,如图 1-34 所示。双击文件,“Program File”一栏显示即将烧录的目标文件,单击“OK”按钮即可完成程序烧录,如图 1-35 所示。之后就可以进行两种软件的联调仿真了。



图 1-33 程序烧录设置

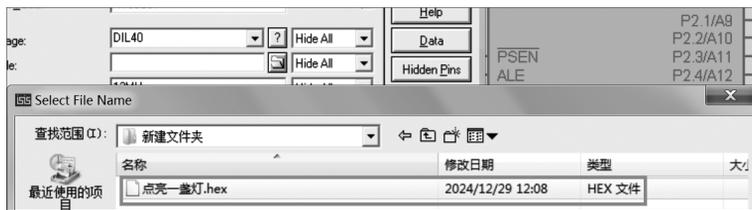


图 1-34 查找目标文件



图 1-35 选中目标文件

1. 常用元器件名

RES:电阻。

CAP:电容。

CRYSTAL:晶振。

RESPACK:电阻排。

LED:发光二极管。

SWITCH:刀开关。

BUTTON:按钮。

DIODE:二极管。

NPN、PNP:三极管。

BUZZER:蜂鸣器。

ELECTRO:电解电容。

Matrix:8×8 点阵。

2. 常用快捷键

F8:当前工作区全部显示。

F6:放大,以鼠标为中心放大。

F7:缩小,以鼠标为中心缩小。

G:栅格开关。

O:重新设置原点,将鼠标指向的点设为原点。

U:撤销键。

任务实施

一、编写程序,并在 Keil 中生成目标文件

单片机控制 LED 点亮的程序如下。

```

#include <reg51.h>
sbit led1 = P1^0;
void main()
{
    led1 = 0;
    while(1);
}
    
```

(1)程序第一行为头文件包含,reg51.h 头文件中使用了 sbit 关键字。程序第一行也可写成#include“reg51.h”。

(2)sbit 关键字用于定义特殊功能寄存器中可以位寻址的相应位,在这里将 P1 口的 P1.0 命名为“led1”。

(3)一个 C51 程序有且只有一个 main 函数,其函数体用{}括起来。

(4)给 led1 赋值 0,即给 P1.0 赋值 0,单片机输出低电平,LED 点亮。

(5)while(1)为条件循环语句,1 代表条件为真,即永真循环,也就是无限循环,语句后面加了“;”,表示语句结束,即原地踏步。

程序编写完成后,单击 Keil 工具栏中的“Built”图标,当软件下面的状态栏显示 creating hex file 时,表示目标文件已经生成。

二、在 Proteus 中搭建硬件电路,实现软件仿真

单片机控制 LED 点亮的硬件电路如图 1-36 所示。该硬电路需要用到的元器件有单片机、晶振、电容、电解电容、电阻、发光二极管。连接好硬件电路,烧录 Keil 软件生成的目标文件。

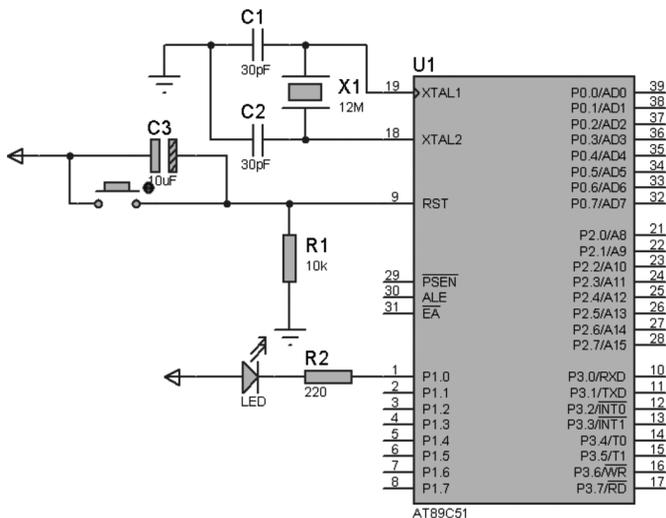


图 1-36 单片机控制 LED 点亮的硬件电路

将目标文件成功烧录至单片机中,然后单击 Proteus 软件左下角的“播放”按钮,即可看到仿真效果,如图 1-37 所示(此图省略单片机最小系统电路)。