

# 第2章 第一个单片机系统

欢迎访问 电路飞翔网

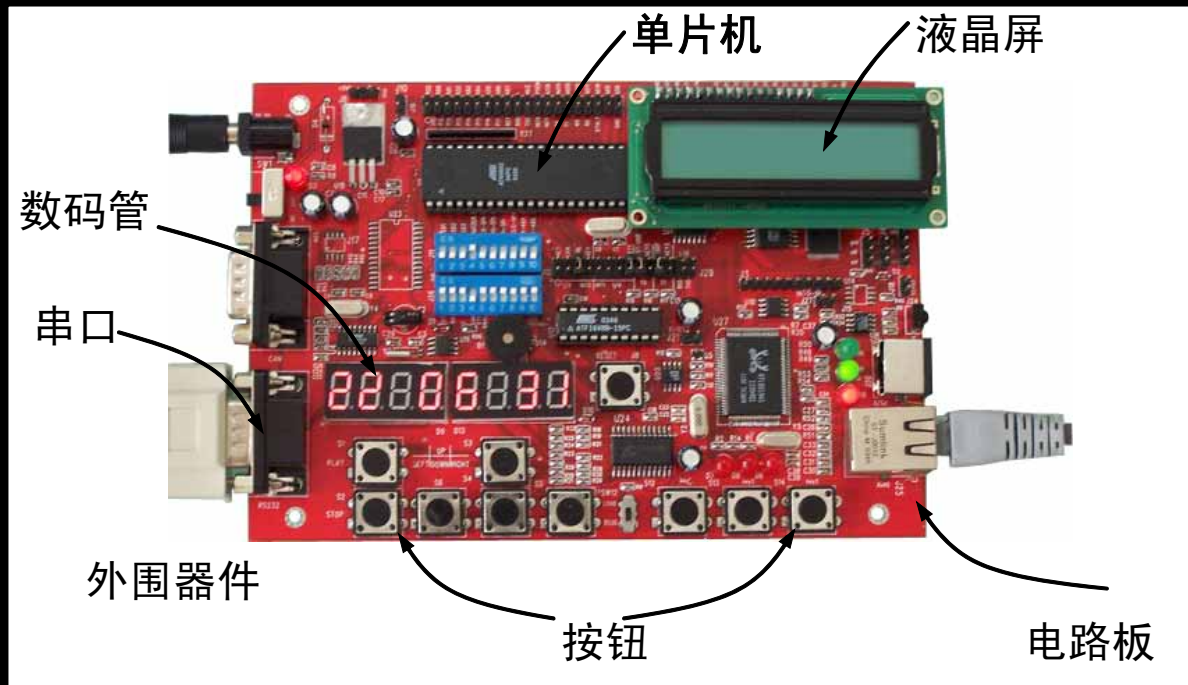
<http://www.circuitfly.com> 获取更多信息

- 2.1 什么是单片机系统
- 2.2 如何控制一个发光二极管
- 2.3 单片机系统开发过程
- 2.4 实例点拨——闪烁5次的发光二极管

## 2.1 什么是单片机系统

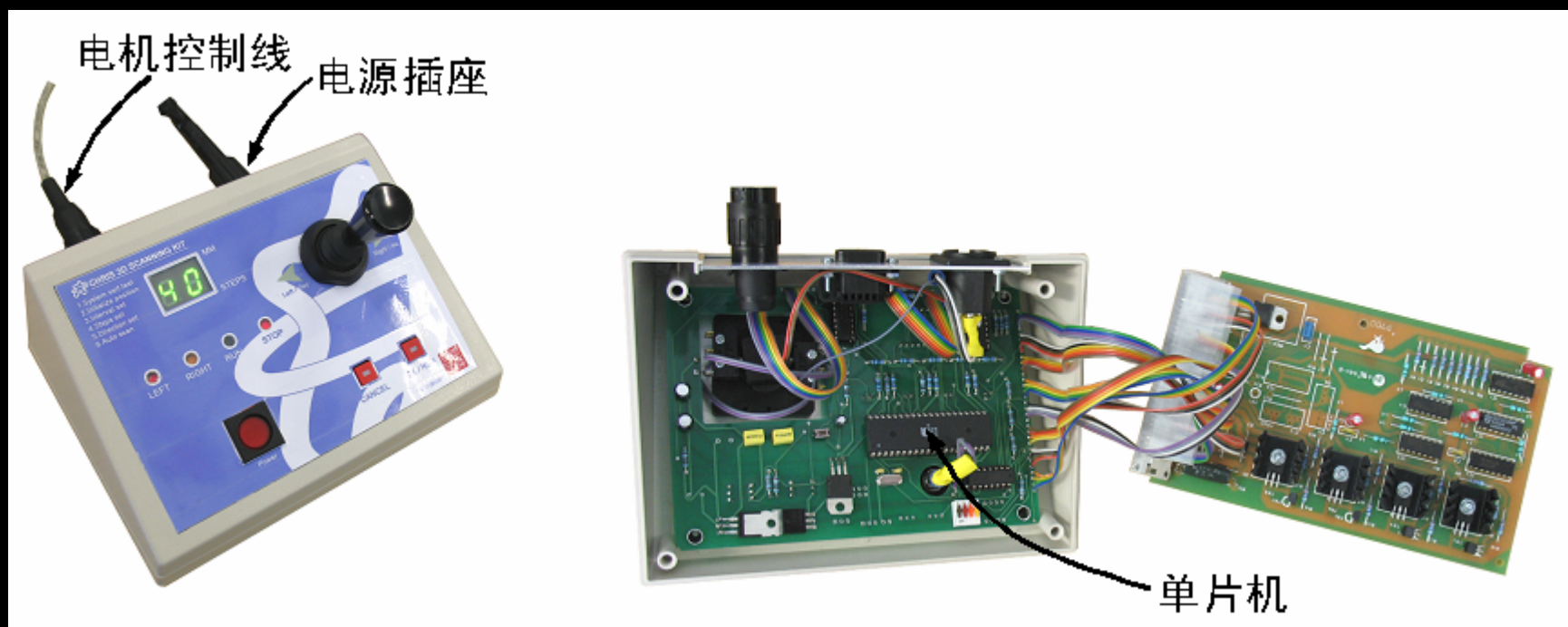
### • 2.1.1 单片机本身

单片机本身就是一个电子器件，可以根据指令的要求控制外围器件，它的功能可以通过指令进行控制。



## 2.1 什么是单片机系统

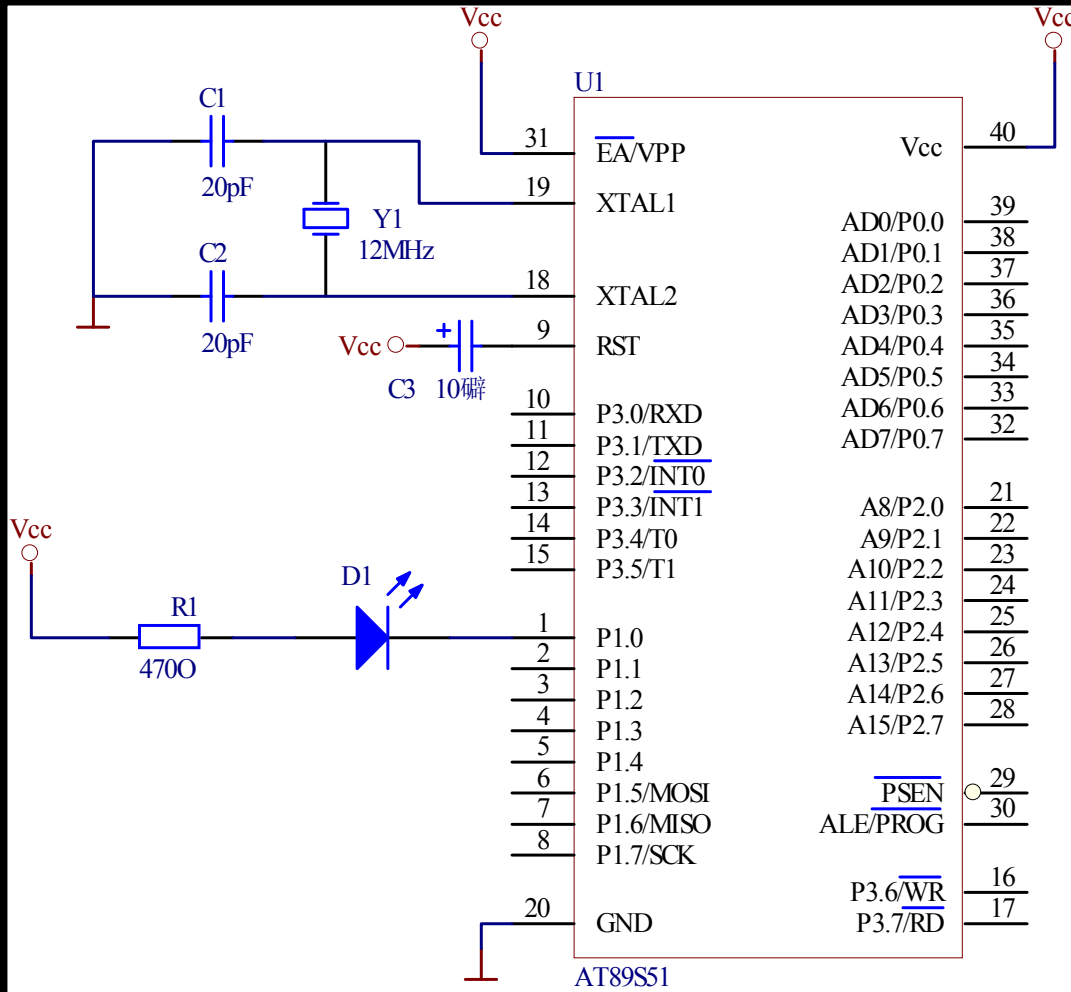
### • 2.1.2 构成单片机系统——单片机+外围器件



### 电机控制盒实例

## 2.2 如何控制一个发光二极管

### • 2.2.1 硬件设计（系统电路图）



电容 C1



电容 C2



电容 C3



电阻 R1



发光二极



管 D1

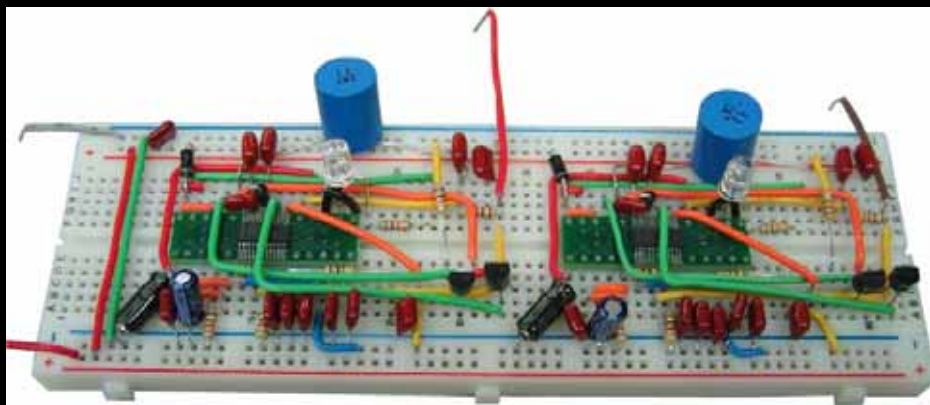


晶振 Y1

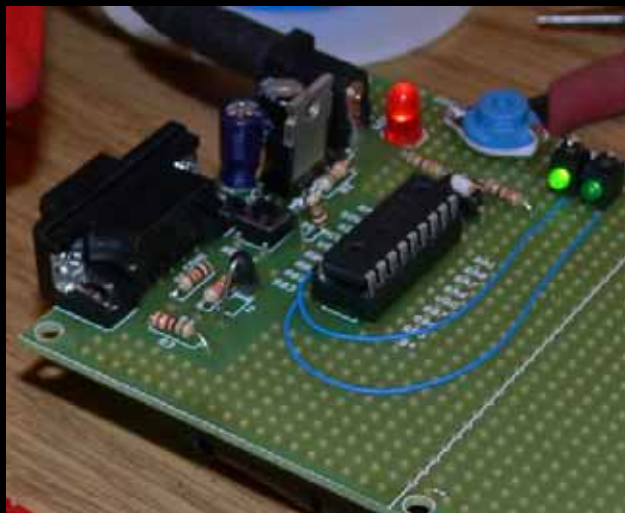


## 2.2 如何控制一个发光二极管

### • 2.2.1 硬件设计（搭建硬件电路的器材）



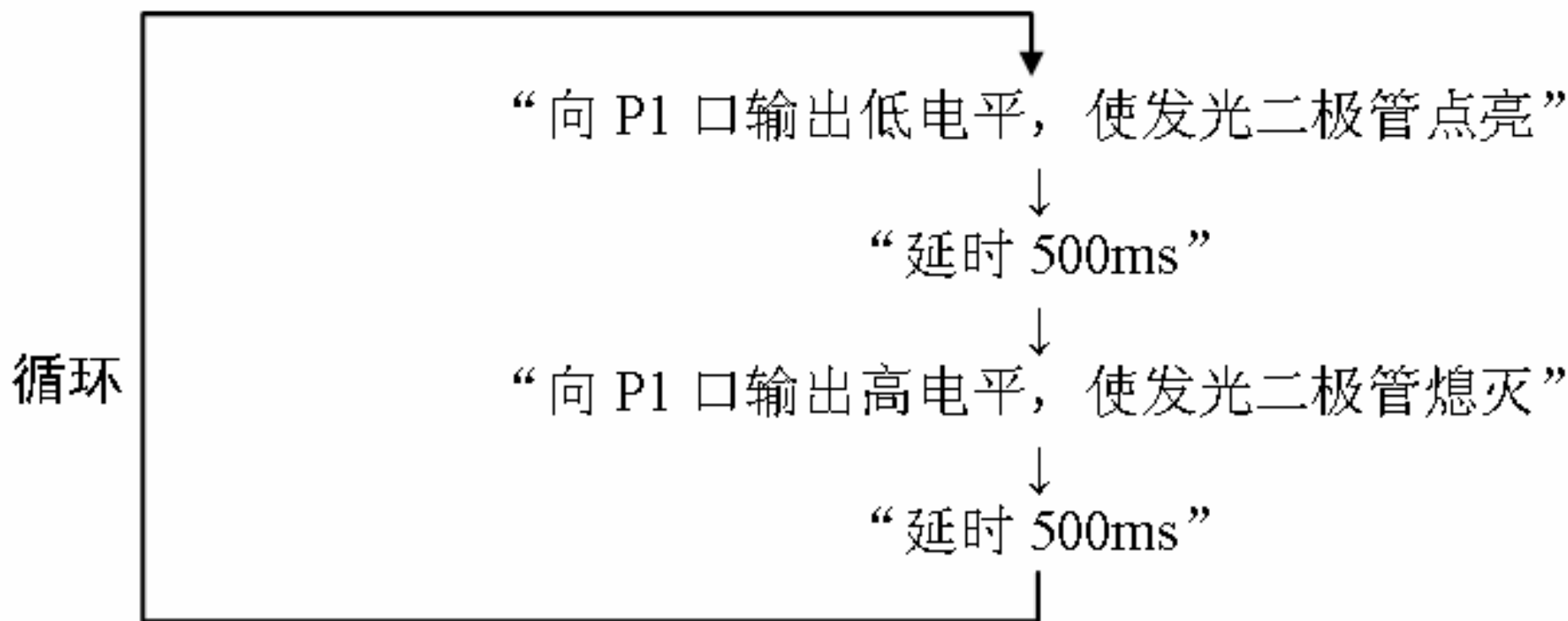
◁ 面包板用于电子技术实验



◁ 万用板用于电子技术实验

## 2.2 如何控制一个发光二极管

### • 2.2.2 软件设计（中文描述的程序设计思路）



## 2.2 如何控制一个发光二极管

- 2.2.2 软件设计（翻译成单片机的语言）

START:

```
MOV    P1, #00H      ; 向P1口输出低电平
CALL   DELAY          ; 调用延时子程序
MOV    P1, #0FFH     ; 向P1口输出高电平
CALL   DELAY          ; 调用延时子程序
JMP     START         ; 跳回START, 循环执行
```

## 2.2 如何控制一个发光二极管

### • 2.2.2 软件设计（指令解析）

- 标号——可以是以英文字母开头的字母、数字或某些特殊符号的序列。
- 助记符——用来表达指令的操作功能。
- 目的操作数——指令最终作用的对象。
- 源操作数——参与指令的操作。
- 注释——分号后面的文字描述

START:	MOV	P1,	#00H	;向P1口输出低电平
↑	↑	↑	↑	↑
标号	:	助记符	目的操作数,	源操作数 ;注释



## 2.2 如何控制一个发光二极管

- 2.2.2 软件设计（完整的程序1/2）

```
ORG    00H                ; 设置起始地址  
START:                ; 标号  
MOV     P1, #00H          ; 向P1口输出低电平，点亮二极管  
CALL    DELAY             ; 调用延时子程序  
MOV     P1, #0FFH         ; 向P1口输出高电平，熄灭二极管  
CALL    DELAY             ; 调用延时子程序  
JMP     START             ; 跳回START，循环执行
```

## 2.2 如何控制一个发光二极管

### 2.2.2 软件设计（完整的程序2/2）

DELAY:

```
        MOV    R3, #50        ; 延时子程序（500ms）  
D1:     MOV    R4, #20  
D2:     MOV    R5, #248  
        DJNZ   R5, $  
        DJNZ   R4, D2  
        DJNZ   R3, D1  
        RET                      ; 返回主程序  
        END                      ; 汇编程序结束
```

## 2.2 如何控制一个发光二极管

### 2.2.2 软件设计（指令简析）

ORG 00H —— 伪指令,其功能是设定经汇编后的执行代码存储在单片机中的起始地址为00H。

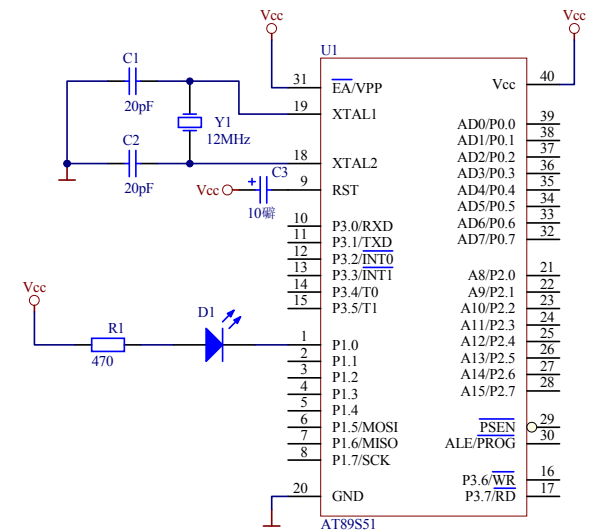
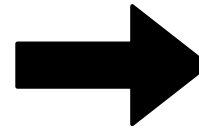
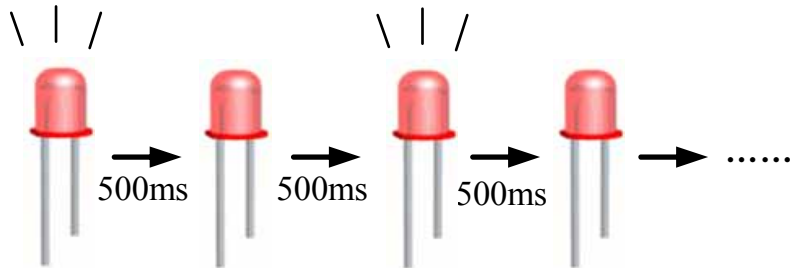
END —— 伪指令,提示汇编器程序结束于此,当汇编遇到END后,就不会再继续进行。

伪指令 (pseudo opcode) —— 汇编程序所提供的帮助汇编器进行汇编的指令,并非单片机指令的一部分,因此伪指令不占存储器空间,只是协助程序的汇编工作。

## 2.3 单片机系统开发过程

1. 明确系统功能:  
闪烁, 间隔500ms

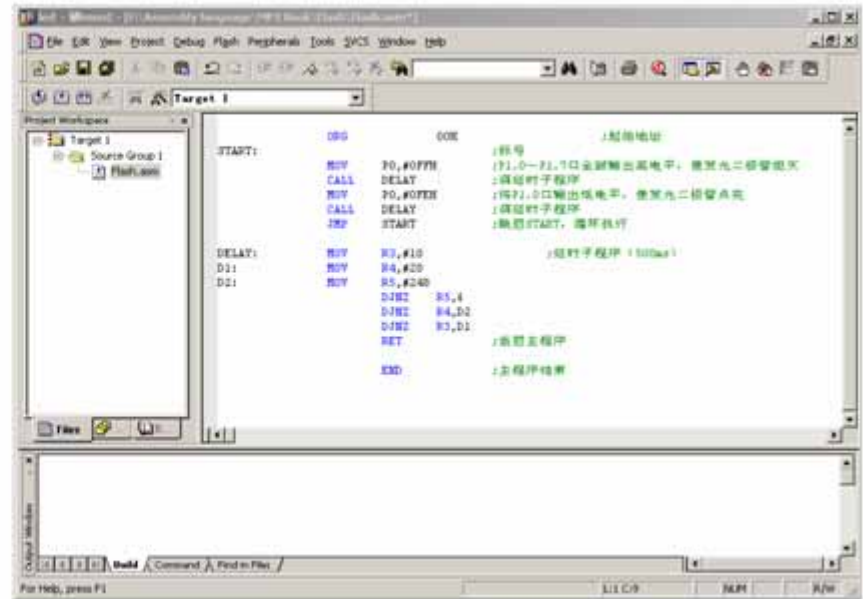
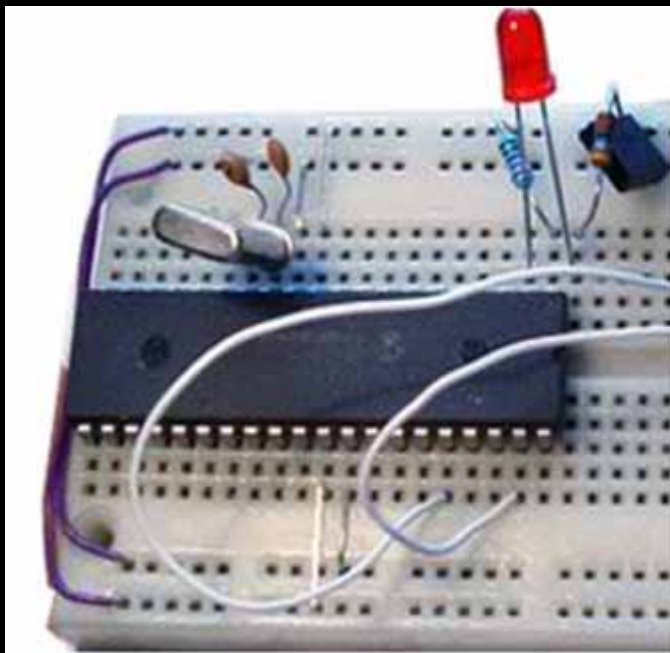
2. 硬件设计:  
系统电路图



## 2.3 单片机系统开发过程

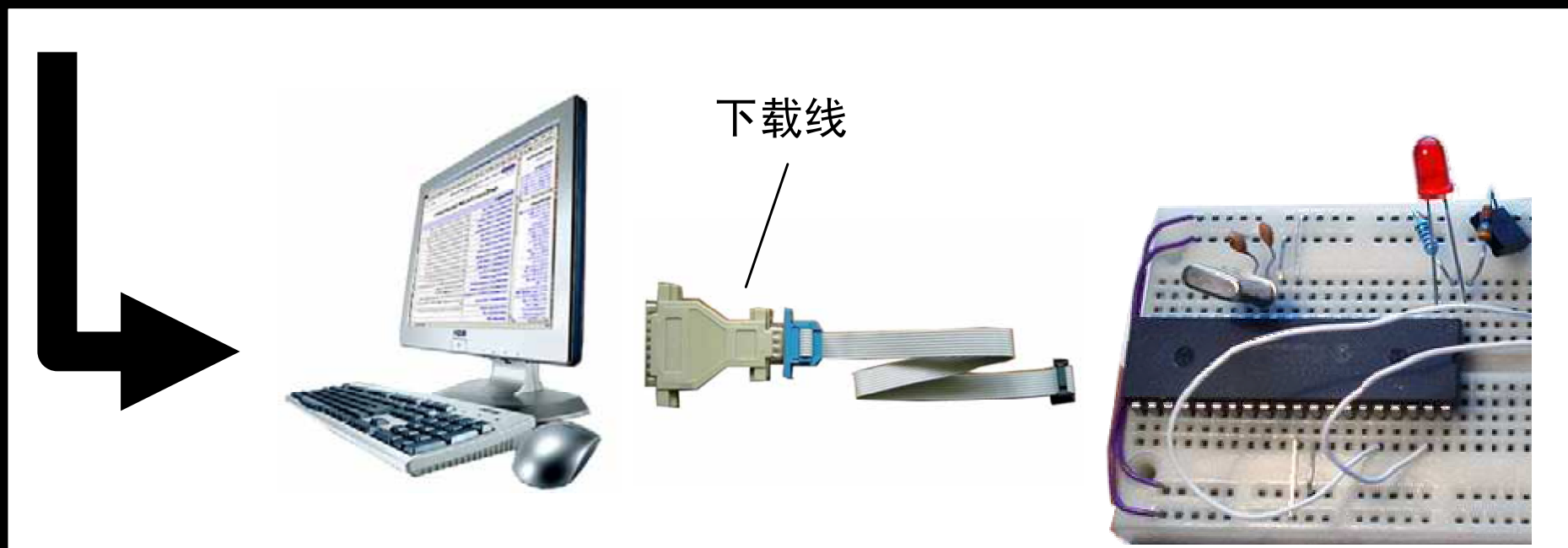
3. 搭建硬件平台：  
在面包板、万用板  
或实验板上

4. 软件设计：  
在汇编语言开发环境中编  
写汇编程序



## 2.3 单片机系统开发过程

### 5. 下载程序到单片机并调试



## 2.4 实例点拨——闪烁5次的发光二极管

- 2.4.1 明确系统功能

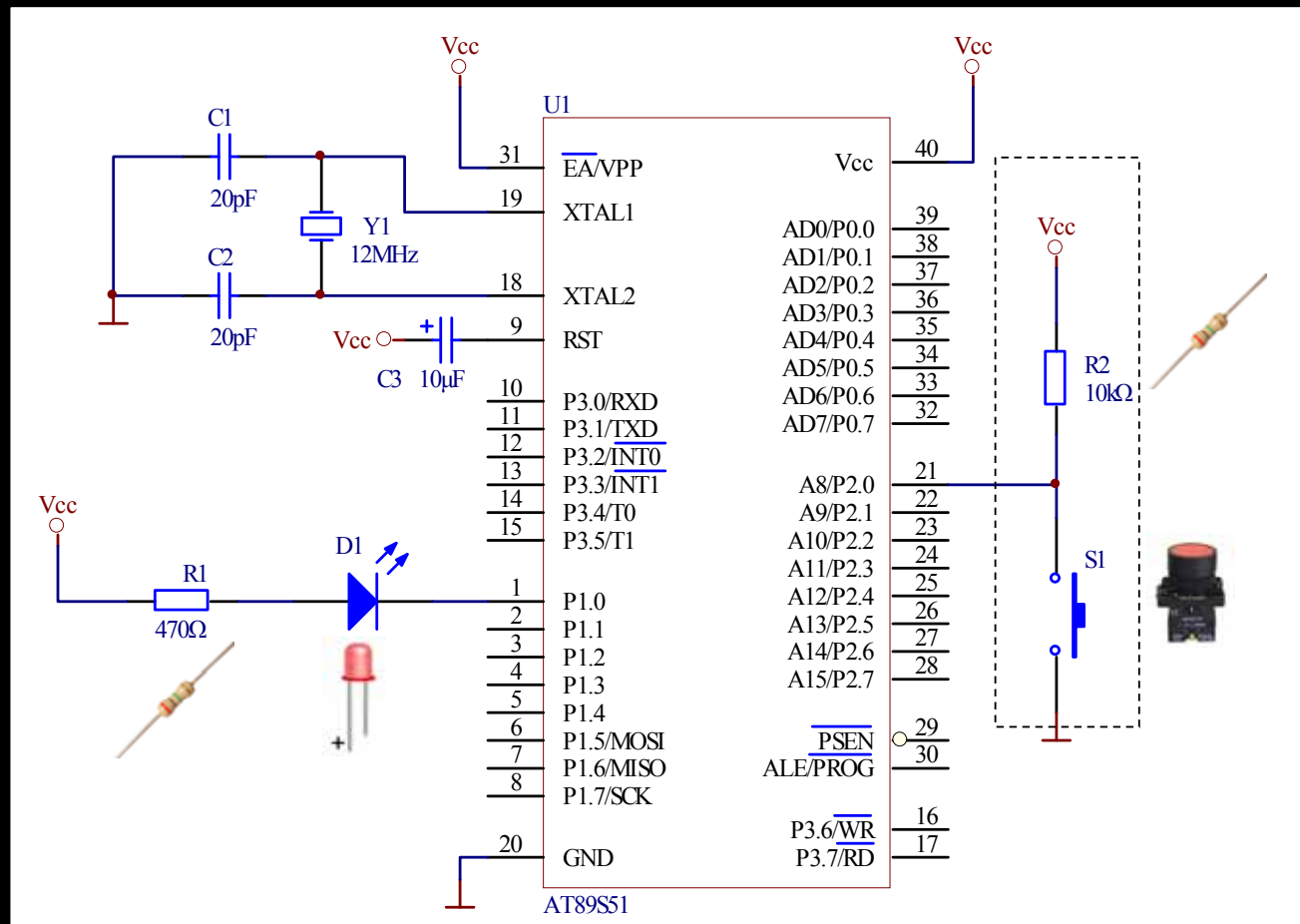
系统启动后，一支发光二极管闪烁5次后熄灭，

当按下一个按钮开关后再闪烁5次后熄灭。

以后每次按下按钮开关，发光二极管都会闪烁5次。

## 2.4 实例点拨——闪烁5次的发光二极管

### • 2.4.2 硬件设计(单片机+外围器件)



外围器件：

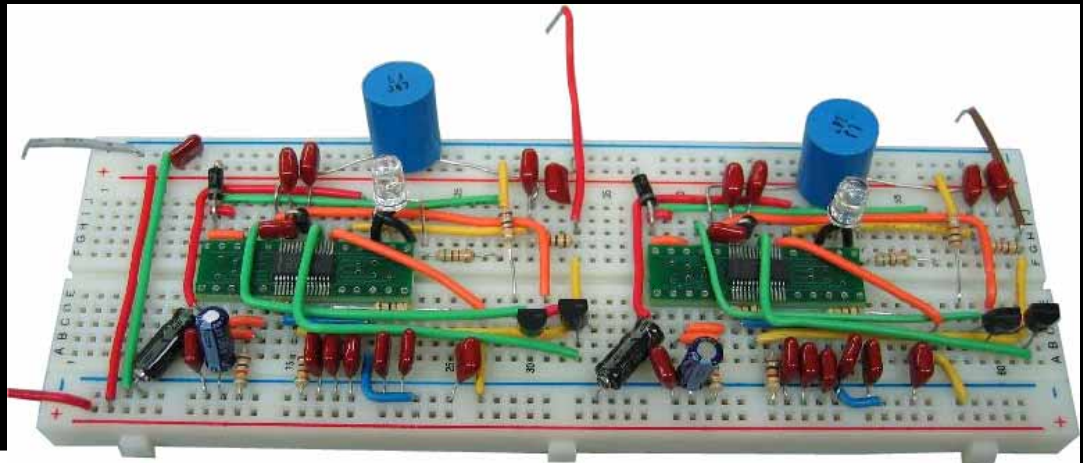
一个发光二  
极管、

一个按钮开  
关



## 2.4 实例点拨——闪烁5次的发光二极管

### • 2.4.3 搭建硬件平台



在面包板或万用板上插入器件，用导线完成电气连接

## 2.4 实例点拨——闪烁5次的发光二极管

### 2.4.4 软件设计

```
ORG    00H           ; 设置起始地址
START:                ; 标号
MOV     R0, #5        ; 循环闪烁5次, R0为计数器
LOOP:                ; 标号
MOV     P1, #00H      ; 向P1口输出低电平, 点亮二极管
CALL    DELAY         ; 调用延时子程序
MOV     P1, #0FFH     ; 向P1口输出高电平, 熄灭二极管
CALL    DELAY         ; 调用延时子程序
DJNZ    R0, LOOP      ; R0减1, 如果不等于0跳到LOOP标号处
```

## 2.4 实例点拨——闪烁5次的发光二极管

### 2.4.4 软件设计

```
JB    P2.0, $    ; 如果P2.0为高电平则循环执行本行
JMP   START      ; 跳到START标号处重新开始执行
DELAY:
.....          ; 延时子程序（500ms）
RET              ; 返回主程序
END              ; 汇编程序结束
```

## 2.4 实例点拨——闪烁5次的发光二极管

### • 2.4.4 软件设计（指令解析）

✓ MOV R0, #5

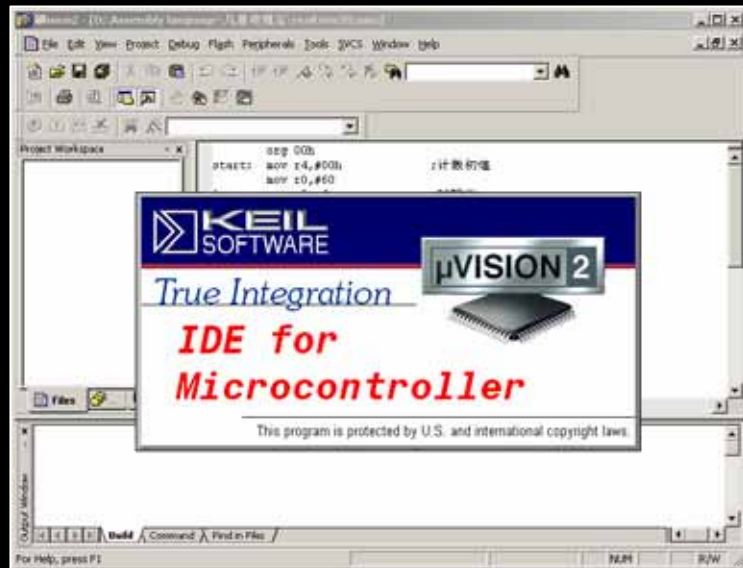
✓ DJNZ R0, LOOP

✓ JB P2.0, \$

✓ JMP START

## 2.4 实例点拨——闪烁5次的发光二极管

### • 2.4.5 下载程序到单片机并调试

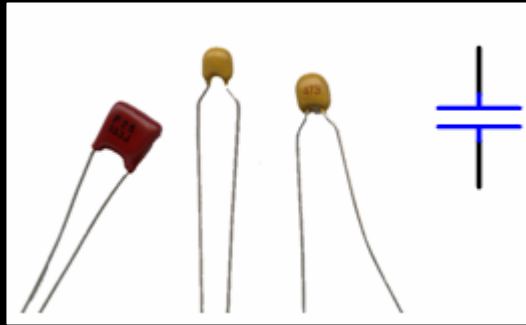


- 在 $\mu$ Vision中把目标程序编辑好,
- 生成单片机可执行的代码,
- 通过下载线下载到单片机系统板中,
- 然后运行该程序

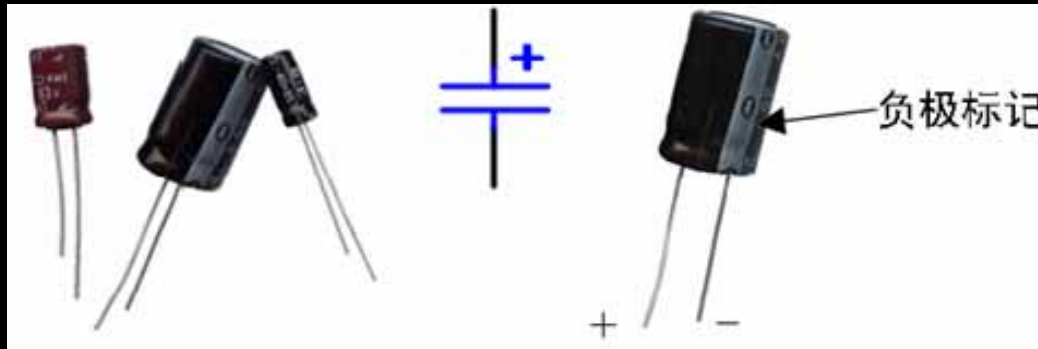
开发环境 $\mu$ Vision

# 器件介绍

## 1. 电容器

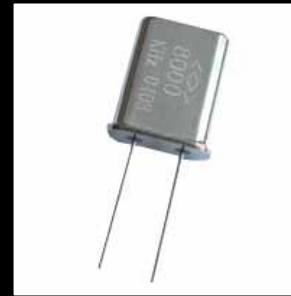


△ 独石电容



△ 电解电容

## 2. 晶体振荡器



◁ 无源晶振



◁ 有源晶振

晶振的电路符号 ▷

