

# 外部中断



专 业：生物医学工程

年 级：2022级

班 级：1班

姓 名：张台忍

学 号：3022202299

邮 箱：ztr8526@gmail.com

**2024年4月13日**

# **一 实验目的**

熟悉集成开发环境的使用方法.

掌握编写中断函数的方法.

掌握单片机外部中断的使用方法.

# **二 实验设备**

微机一台, 集成开发环境一套, 单片机开发仪

# **三 实验内容**

实验流程如图所示

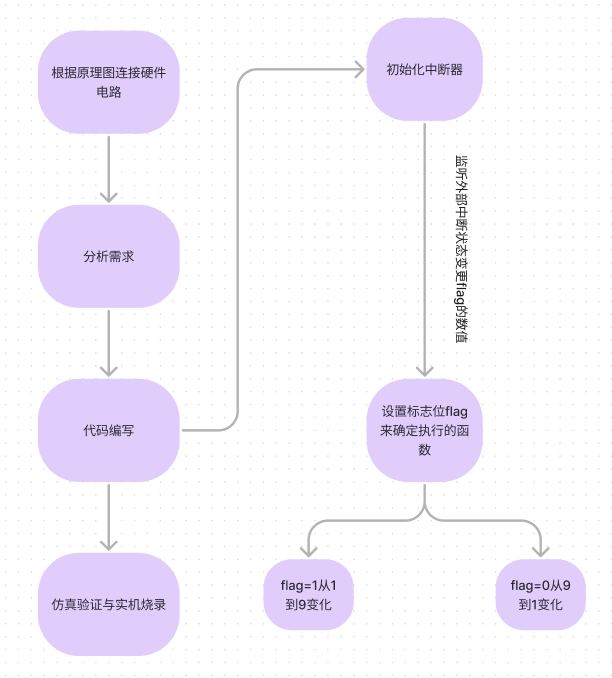


图1 实验流程图

## 1 建立工程

在中, 参照实验0的方法, 建立工程, 然后新建语言程序. 编译通过后进行接下来的操作.

## 2 需求分析

本次实验想要实现的功能是实现两个外部中断, 分别为外部中断和外部中断; 外部中断要实现在数码管上显示数字, 数字从开始, 以的周期递增显示到数字, 并且停留在上; 外部中断要实现的功能是在静态数码管上显示数字, 数字从开始, 以的周期递减显示到数字, 并停留在上. 要实现这些功能, 我们需要用到外部中断, 首先观察原理图, 完成外部中断的分配和硬件电路的连接.

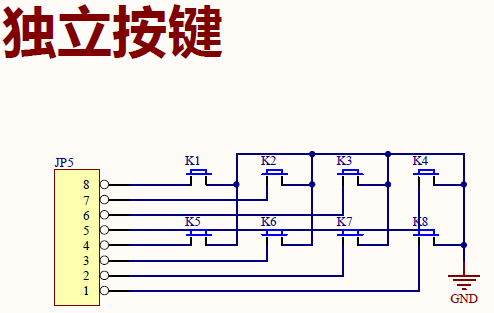


图2 独立按键原理图

本实验中, 按键控制外部中断, 按键控制外部中断. 查阅单片机手册得知, 引脚对应外部中断, 引脚对应外部中断. 因此硬件电路连接时, 分别将和接到单片机的对应引脚上, 使得开关的状态称为外部中断的触发条件.

数码管的显示方式在上次的实验中已得到充分论述, 再此便不在论述.

## 3 仿真电路搭建

通过上一节的分析, 可在中搭建仿真电路, 仿真电路图如下所示.

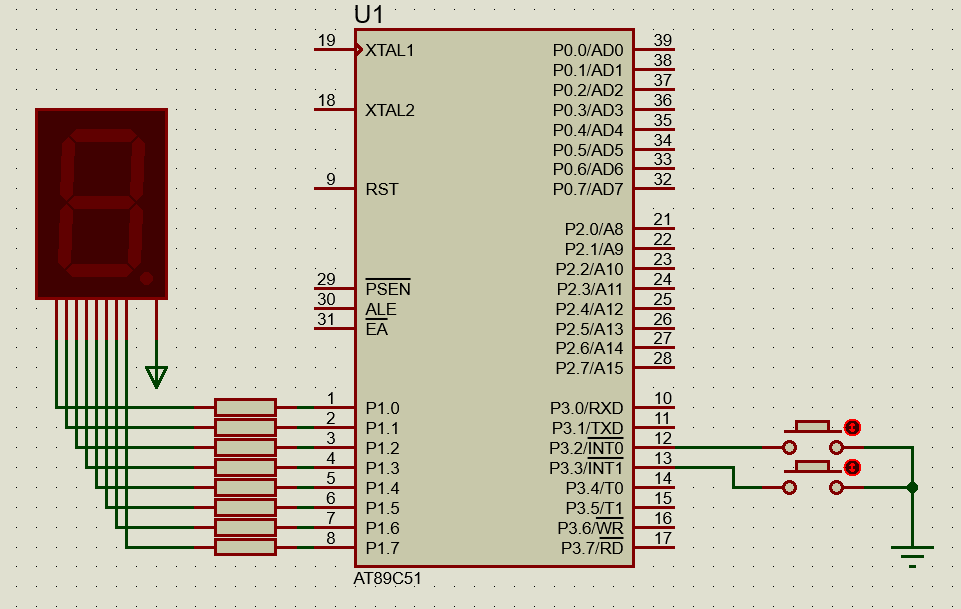


图3 仿真电路搭建

电路图中按键接入外部中断控制端口, 数码管通过端口连接, 输出高低电平控制显示的数字.

## 4 代码编写

连接完硬件电路之后, 开始进行代码的编写. 此实验设计按键, 因此需要在代码中加入按键消抖的代码. 以下代码是按键消抖代码.

1. **if**(P3\_2==0)
2. {
3. DelayMs(10);
4. **while**(P3\_2==0);
5. //要执行的代码
6. }

代码中使用语句判断按键是否按下, 如果按下将执行内部的代码, 按下按键后, 执行一段时间的延时, 消除由按键物理抖动引起的电平变化, 再进入的循环, 此句的功能是等按键松开时候再执行操作, 若一直不松开按键, 代码将一直停留在中, 无法进一步执行, 此操作保证安全性.

由于本实验设计外部中断, 因此再开始编写功能代码之前, 应该先初始化中断器, 使能要使用的外部中断, 查阅数据手册可知, 需要先将置, 打开总中断开关, 再将, , , 全部置, 打开外部中断和外部中断. 通过这种方式即可实现中断器的初始化. 具体代码如下所示.

1. **void** Init()
2. {
3. IT0=1;
4. EX0=1;
5. IT1=1;
6. EX1=1;
7. EA=1;
8. }

完成这些基本的设置后, 即可开始编写功能代码. 完整代码如下.

1. #include <regx51.h>
2. #include <intrins.h>
3. #include "Delay.h"
5. **char** a[]={0xF9,0xA4,0xB0,0x99,0x92,0x82,0xF8,0x80,0x98};
6. **int** i=0;
7. **int** j=8;
8. flag=0;
9. **void** Init()
10. {
11. IT0=1;
12. EX0=1;
13. IT1=1;
14. EX1=1;
15. EA=1;
16. }
18. **void** k1() interrupt 0
19. {
20. **if**(P3\_2==0)
21. {
22. DelayMs(10);
23. **while**(P3\_2==0);
24. flag=1;
25. i=0;
26. }
27. }
29. **void** k2() interrupt 2
30. {
31. **if**(P3\_3==0)
32. {
33. DelayMs(10);
34. **while**(P3\_3==0);
35. flag=2;
36. j=8;
37. }
38. }
40. **void** goUp()
41. {
42. **while**(flag==1)
43. {
44. P1=a[i];
45. i++;
46. **if**(i>8)
47. i=8;
48. DelayMs(300);
49. }
50. }
52. **void** goLow()
53. {
55. **while**(flag==2)
56. {
57. P1=a[j];
58. j--;
59. **if**(j<0)
60. j=0;
61. DelayMs(300);
62. }
63. }
65. **void** main()
66. {
67. Init();
68. **while**(1)
69. {
70. **if**(flag==1)
71. goUp();
72. **if**(flag==2)
73. goLow();
74. }
75. }

代码中函数实现检测按键按下并触发外部中断, 函数实现检测按键按下并触发外部中断. 函数实现数码管数字显示从数字以的周期递减至数字并停在数字, 函数实现数码管数字显示从数字以的周期递增至数字并停在数字. 中断函数中通过对于标志位的改变, 实现状态的切换.

# 四 结果与讨论

## 1 仿真测试

代码编写完成后, 将生成的文件烧录至仿真软件中的单片机中, 测试功能是否实现.

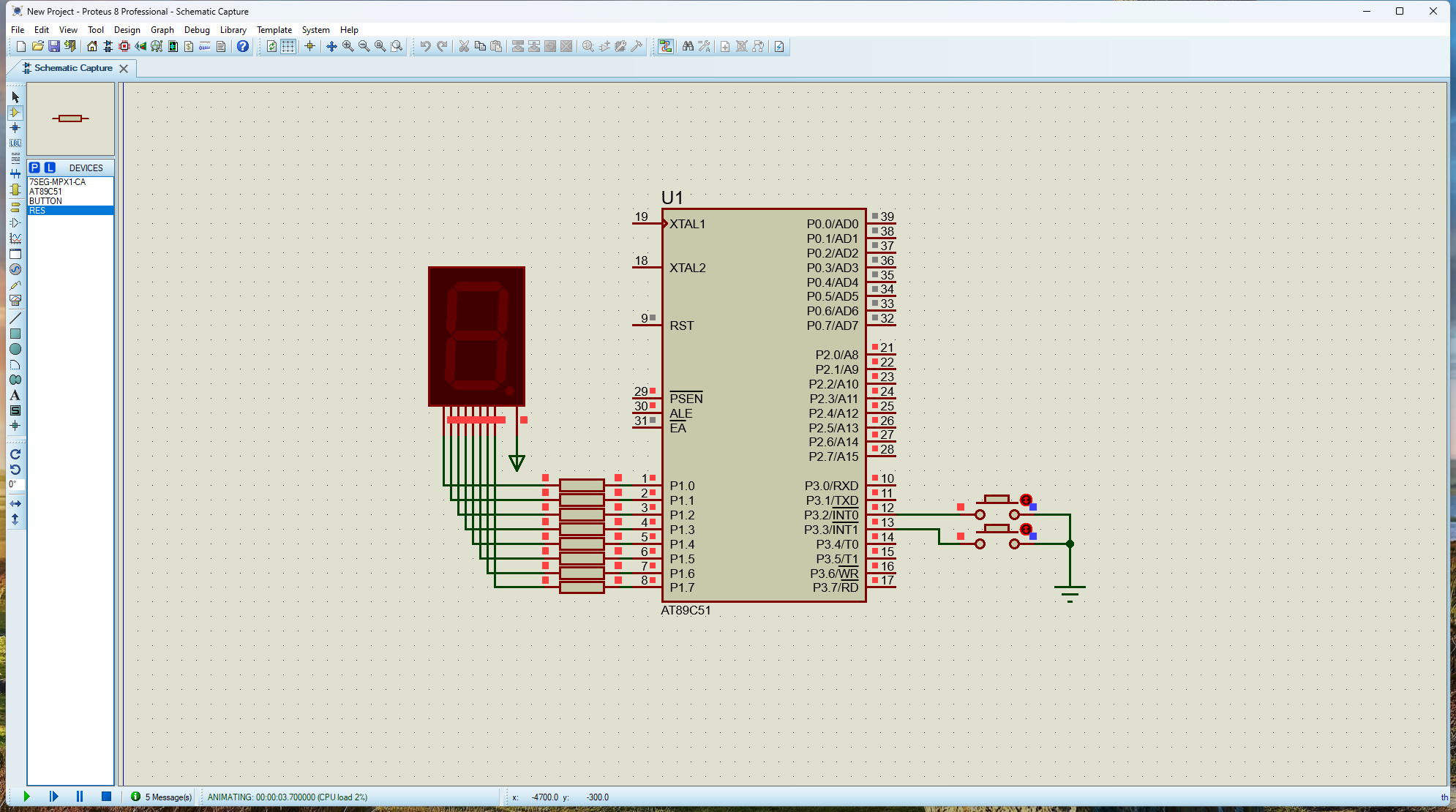


图4 仿真测试

当按下按键时, 观察到数码管从开始显示, 并递增到数字, 并且再递增过程中再次按下按键, 数字会回到并开始重新递增. 同样, 当按下按键时, 观察到数码管从开始显示, 并递减到数字, 并且再递减过程中再次按下按键, 数字会回到并开始重新递减. 从仿真结果看, 已实现了要求的功能.

## 2 实机烧录

把生成的文件烧录至单片机中, 观察实际的现象.

## 3 分析讨论

本次实验涉及到外部中断的原理和应用, 并涉及到实际操作中物理按键消抖的代码实现. 通过此实验加深了对单片机中断的理解.