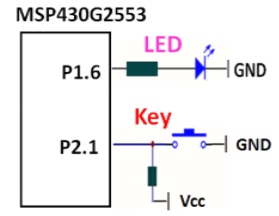
**课前练习：**

接线如右图，采用下面**两种编程方法均能实现**：

每按下一次P2.1连的按键，

就把连接在P1.6上LED灯的状态求反一次。

**请比较两个程序有哪些不同之处？写出你认为的两个主要不同。**

**（注意：请不要写什么是中断，也不要写查询和中断的区别，**

**而是写这两个程序上的不同）**

**//方法2**

#include "msp430.h"

int main( void )

{ WDTCTL = WDTPW + WDTHOLD; //关闭看门狗

P1DIR |=BIT6; //置P1.6为输出

P2DIR &=~BIT1 ; //置P2.1为输入

P2IES |=BIT1 ;

P2IFG &=~BIT1 ;

P2IE |=BIT1;

\_EINT( );

for(;;) { }; //主循环

}

#pragma vector=PORT2\_VECTOR

\_\_interrupt void blink( )

{ P1OUT^=BIT6;

P2IFG &=~BIT1;

}

**//方法1:**

#include "msp430.h"

void blink( );

int main( void )

{ WDTCTL = WDTPW + WDTHOLD;

P1DIR |=BIT6; //置P1.6为输出

P2DIR &=~BIT1 ; //置P2.1为输入

for(;;) //主循环

{ while((P2IN&BIT1)!=0){}; //查询按键按下

while((P2IN&BIT1)==0){}; //查询按键释放

blink( ); //调用LED求反函数

};

}

void blink( ) //LED求反函数

{

P1OUT ^=BIT6; //对P1.6求反

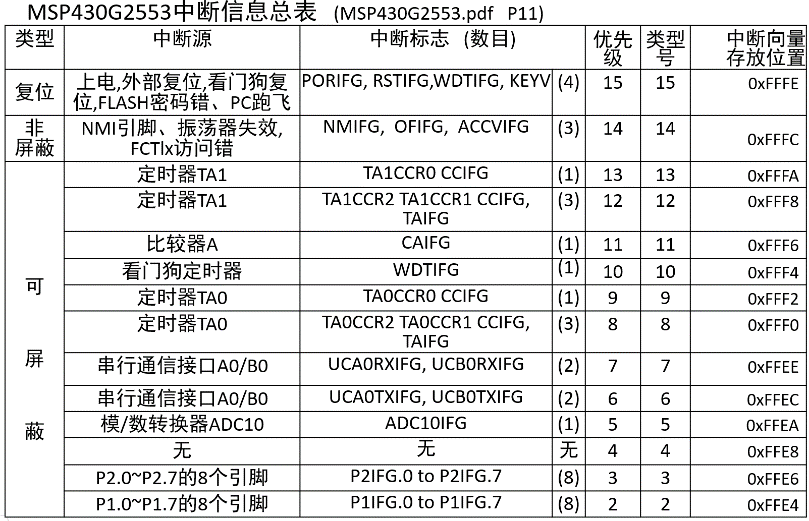
}

**一、**根据**学堂上提供的“实验4-1-中断预习练习讲解”视频**，写出上题中的**两个主要不同**：

1.方法2中断源的引脚信号不需要CPU通过执行程序的方式进行检测。

2.方法1需要查询P2IN寄存器，而方法2不用。

**二、边做实验任务，边回答下面的5个问题**：



1. 在msp430g2553.h头文件中，

符号PORT1\_VECTOR对应的类型号值是多少？

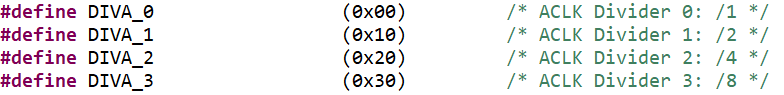
(任务1，学习查看msp430g553.h头文件的定义，

与中断信息总表对应)

符号PORT1\_VECTOR对应的类型号值是2

1. 在msp430g2553.h头文件中查找符号DIVA0、DIAV1、DIVA\_0、DIVA\_1、DIVA\_2、DIVA\_3的定义，比较DIVA0和DIVA\_0不同、DIVA1和DIVA\_1不同。(任务1，学习查看msp430g553.h头文件的定义，与明天实验5相关)



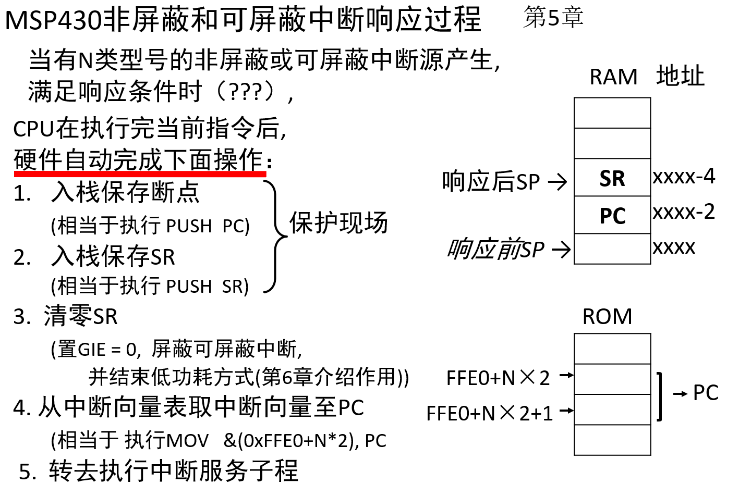


DIVA0和DIVA\_0不同:

DIVA0的值为 0x10; DIVA\_0的值为0x00

DIVA1和DIVA\_1不同:

DIVA1的值为 0x20; DIVA\_1的值为0x10



1. 什么是程序跑飞？为什么会造成跑飞？

(与实验4任务1的第5小问相关)

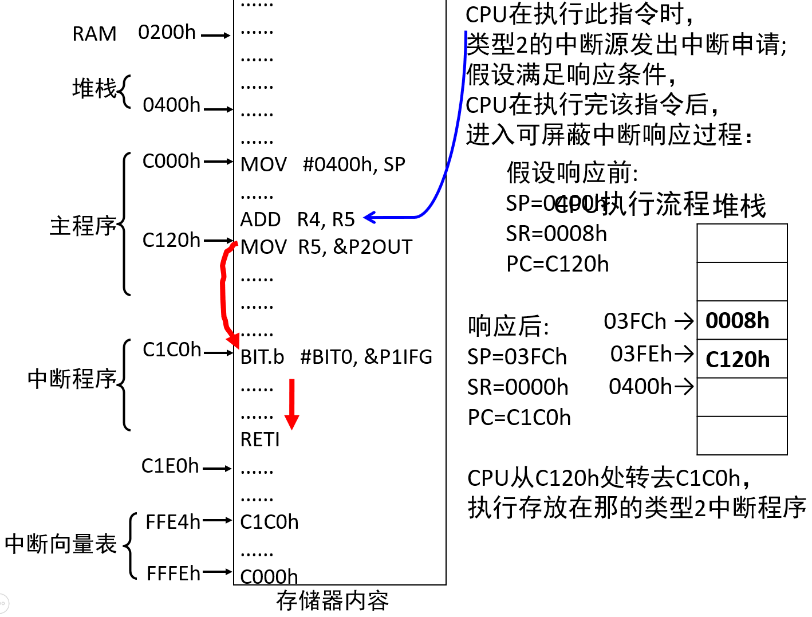
程序跑飞运行时偏离正常的运行路径，无法正常运行。

中断向量的端口位置与实际端口位置不一样，PC指针

不指向自己编写程序造成程序跑飞。

1. 引脚P1.2、P1.4、P1.5和P2.3都作为中断申请引脚，

请问需要写几个中断函数? 是哪几个？

(任务2，掌握中断源与其对应的中断函数的关系)

需要写2个中断函数，分别是:

**#pragma** vector1=PORT1\_VECTOR

**#pragma** vector2=PORT2\_VECTOR

**\_\_interrupt** **void** **pin\_ISR1**( )

**\_\_interrupt** **void** **pin\_ISR2**( )

5．在中断函数入口加断点，对调试中断实验有什么益处? **?(参看中断调试经验视频，并在实验中应用)**

可以帮助分析中断函数的功能为何没有实现