

微机原理实验报告

实验八 8086 中断实验

[REDACTED]

[REDACTED]

[REDACTED]

2021/12/20


一、实验目的

1. 了解8086内部响应中断的机制；掌握中断向量的作用。
2. 利用实验仪上单脉冲、74HC244电路，不使用8259，实现一个中断实例。
3. 复习本节实验内容，可尝试自行编写程序，做好实验准备工作，填写实验报告。

二、实验设备

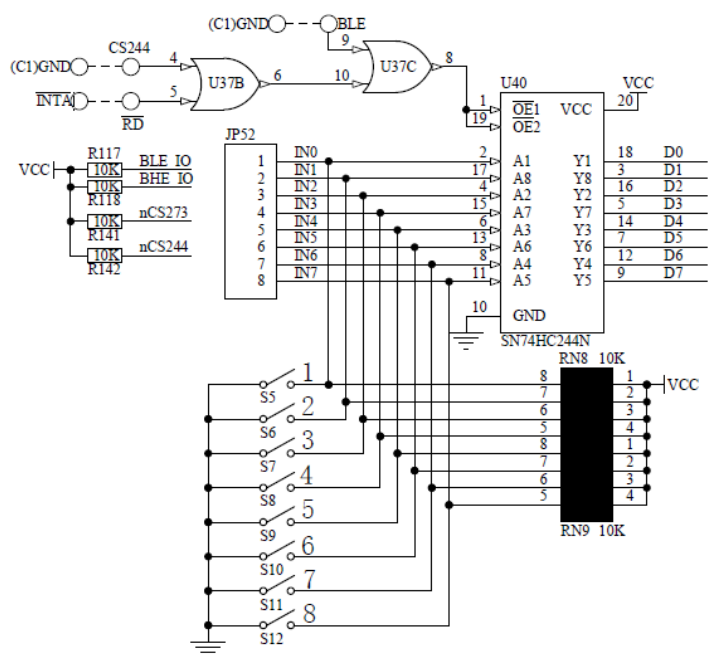
STAR 系列实验仪一套、PC 机一台。

三、实验内容

编写程序：拨动单脉冲开关，“”送给 8086 的 INTR，触发中断；8086 通过 INTA 信号，读取中断向量；8086 计数中断次数，在中断服务程序中调用显示程序延时以达到开关的去抖着用，中断次数显示于 F5 区的数码管上；找出中断矢量地址中中断服务程序的入口地址。

注意：给 INTR 高电平信号，8086 就会相应中断，所以实验开始前，保证单脉冲开关给 8086 低电平；中断程序中，加一个较长的延时程序，在中断结束前，有时间拨动单脉冲开关，恢复给 8086 低电平。

四、实验原理图



本实验，通过 F4 区的 8 个拨动开关，给 74HC244 设定中断向量；本实验的中断向量是 08H，即 IN7-IN0 位数据是 00001000。同学可以自定义中断向量，实验程序中处理中断向量部分程序作相应调整。

五、实验步骤

1、连线说明：

B4 区：CS244、BLE —— C1 区：GND

B4 区：RD (IO 区) —— A3 区：INTA

A3 区：INTR —— B2 区：单脉冲

B4 区：JP57 (D0..D7) —— A3 区：JP41


B4 区：JP52 (IN0..7) —— F4 区：JP27 (1..8)

D3 区：CS、A0、A1 —— A3 区：CS1、A0、A1

D3 区：PC0、PC1 —— F5 区：KL1、KL2

D3 区：JP20、B、C —— F5 区：A、B、C

2、调试程序，查看运行结果是否正确。

3、实验开始前，保证单脉冲开关给 8086 低电平；运行程序；向下拨动开关（触发中断），立即向上拨动开关，产生一个“”，观察结果，数码管上显示的次数与拨动开关次数是否对应。

六、编写并调试程序

七、实验扩展及思考

1. 绘制本实验的详细实验电路图原理图。
2. 扼要注释每指令的功能

程序

```
EXTRN      InitKeyDisplay:NEAR,   Display8:NEAR
_STACK     SEGMENT      STACK
    DW     100          DUP(?)
    _STACK ENDS
    _DATA  SEGMENT      WORD PUBLIC 'DATA'
    BUFFER DB           8 DUP(?)
    Counter DB          ?
    ReDisplayFlag DB    0
    _DATA  ENDS
CODE        SEGMENT
START       PROC        NEAR
ASSUME      CS:CODE, DS:_DATA, SS:_STACK
    MOV     AX,         _DATA
    MOV     DS,         AX
    MOV     ES,         AX
    NOP
    CALL    InitKeyDisplay    ;对键盘、数码管控制器 8255 初始化
    CALL    WriIntver
    MOV     Counter,        0    ;中断次数
    MOV     ReDisplayFlag,  1    ;需要显示
    STI     ;开中断
START1:
    LEA     SI,Buffer
    CALL    Display8
    CMP     ReDisplayFlag,  0
    JZ      START1
    CALL    LedDisplay
    MOV     ReDisplayFlag,  0
    JMP     START1
WriIntver   PROC        NEAR
    PUSH    ES
    MOV     AX,         0
    MOV     ES,         AX
    MOV     DI,         20H
    LEA     AX,         INT_0
    STOSW
    MOV     AX,         CS
    STOSW
    POP     ES
    RET
WriIntver   ENDP
LedDisplay  PROC        NEAR
```

```

MOV     AL,      Counter
MOV     AH,      AL
AND     AL,      0FH
MOV     Buffer,   AL
AND     AH,      0F0H
ROR     AH,      4
MOV     Buffer + 1, AH
MOV     Buffer + 2, 10H ;高六位不需要显示
MOV     Buffer + 3, 10H
MOV     Buffer + 4, 10H
MOV     Buffer + 5, 10H
MOV     Buffer + 6, 10H
MOV     Buffer + 7, 10H
RET

```

LedDisplay ENDP

INT_0:

```

PUSH    DX
PUSH    AX
MOV     AL,      Counter
ADD     AL,      1
DAA
MOV     Counter, AL
MOV     ReDisplayFlag, 1
CALL    LedDisplay

```

DELAY:

```

PUSH    BX
PUSH    CX
PUSH    DI
PUSH    SI
MOV     CX,      20

```

DELAY1:

```

LEA     SI,      Buffer
CALL    Display8
loop    DELAY1
POP     SI
POP     DI
POP     CX
POP     BX
POP     AX
POP     DX
IRET

```

START ENDP

CODE ENDS

END START

