

华中科技大学  
人工智能与自动化学院

## 微机原理实验八：8086 中断实验

彭杨哲

U201914634

2021 年 12 月 13 日

## 1 实验目的

- 了解 8086 内部响应中断的机制；掌握中断向量的作用
- 利用实验仪上单脉冲、74HC244 电路，不使用 8259，实现一个中断实例。
- 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。

## 2 实验内容

- 编制程序：拨动单脉冲开关，将正脉冲送给 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，显示于 F5 区的数码管上注意：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。

## 3 实验原理图

如图1所示 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。也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

## 4 实验步骤

1. 连线说明如表1所示

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

Table 1: 连线表

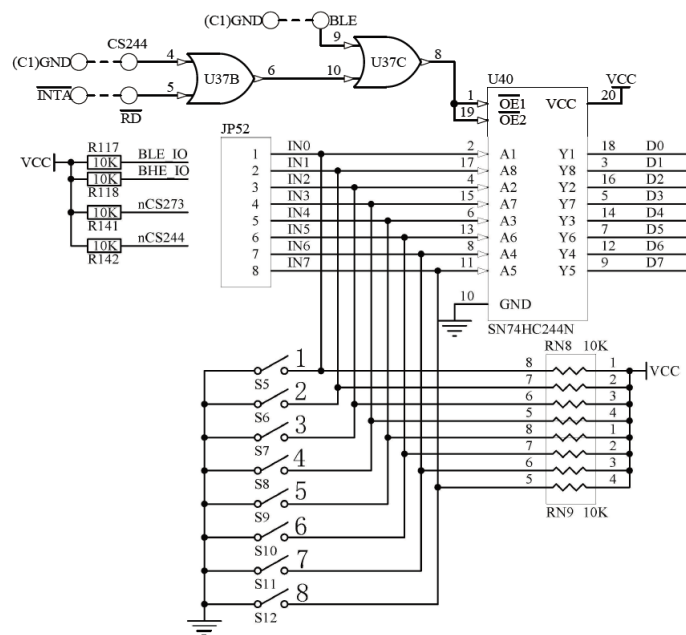


Figure 1: 实验原理图

## 2. 运行程序

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, 产生一个正脉冲, 观察结果, 数码管上显示的次数与拨动开关次数是否对应。

## 5 实验结果

## 5.1 实验代码编写

```

1 ;中断
2 EXTRN InitKeyDisplay:NEAR, Display8:NEAR
3
4 _STACK SEGMENT STACK ;堆栈段
5     DW 100 DUP(?) ;定义100个字空间
6 _STACK ENDS
7
8 DATA SEGMENT
9     Buffer DB 8 DUP(10H) ;定义数据缓冲区
10    cnt DB ? ;中断次数
11    ReDisplayFlag DB 0 ;是否需要显示
12 DATA ENDS
13

```

```

14 CODE SEGMENT
15 MAIN PROC NEAR
16     ASSUME CS:CODE, DS:DATA, SS:_STACK;定义堆栈段地址
17     MOV AX,DATA;加载数据段地址
18     MOV DS,AX;将数据段地址加载到DS
19     MOV ES,AX;设置ES为DATA的地址
20     MOV AX,_STACK;加载堆栈段地址
21     MOV SS,AX;设置SS为STACK的地址
22
23     CALL InitKeyDisplay ;对键盘、数码管控制器8255初始化
24     CALL WriteInt ;调用中断向量设置子程序
25     MOV cnt,0 ;中断次数
26     MOV ReDisplayFlag,1 ;需要显示
27     STI ;开中断
28
29 START1:
30     LEA SI,Buffer ;SK-Buffer (显示缓冲区) 偏移地址
31     CALL Display8 ;显示
32     CMP ReDisplayFlag,0 ;判断是否需要显示
33     JZ START1 ;若ReDisplayFlag=0, 不需要显示, 跳转回START1, 进行新一轮检测
34     CALL Display_0
35         ;若ReDisplayFlag 0, 需要显示, 调用将中断次数转化成可显示格式子程序
36     MOV ReDisplayFlag,0 ;ReDisplayFlag←0, 还原为不需要显示
37     JMP START1 ;跳转回START1, 进行新一轮检测
38
39 ;注意不能命名为Display, 会报错
40 Display_0 PROC NEAR ;将中断次数转化成可显示格式
41     MOV AL,cnt ;将中断次数放入AL
42     MOV AH,AL ;AH←AL, AH也存入中断次数
43     AND AL,0FH ;AL高四位清0, 保留低四位, 即低位BCD码
44     MOV Buffer+6,AL ;第二位显示数据
45     AND AH,0F0H ;AH低四位清0, 保留高四位, 即高位BCD码
46     ROR AH,4 ;循环右移四次, 交换低四位和高四位位置, 转化为可显示格式
47     MOV Buffer + 7,AH ;第一位显示数据
48
49     MOV Buffer + 0,10H ;不显示
50     MOV Buffer + 1,10H ;不显示
51     MOV Buffer + 2,10H ;不显示
52     MOV Buffer + 3,10H ;不显示
53     MOV Buffer + 4,10H ;不显示
54     MOV Buffer + 5,10H ;不显示
55     RET ;返回主程序
56 Display_0 ENDP ;Display过程结束
57
58 WriteInt PROC NEAR ;中断向量设置子程序
59     MOV AX,0
60     MOV ES,AX;目的地址基址
61     MOV BX,20H;目的地址偏移地址
62     MOV AX, OFFSET Interpret;中断程序的偏移地址
63     MOV ES:[BX],AX;置入偏移地址

```

```

64     MOV AX,SEG Interpret;中断程序的段地址
65     MOV ES:[BX+2],AX;置入段基地址
66     RET
67 WriteInt ENDP ;WriteInt过程结束
68
69 Interpret:
70     PUSH DX ;DX入栈保护
71     PUSH AX ;AX入栈保护 (保护现场)
72     MOV AL,cnt ;AL=中断次数
73     INC AL;中断次数+1
74     DAA ;将AL(中断次数)调整为正确的压缩BCD数
75     JNC A
76     ADD BP,1
77 A: MOV cnt,AL ;Cnt=调整为压缩BCD数后的中断次数
78     MOV ReDisplayFlag,1 ;需要显示
79
80     CALL Display_0 ;调用将中断次数转化成可显示格式子程序
81     PUSH CX ;CX入栈保护
82     PUSH DI ;DI入栈保护
83     PUSH SI ;SI入栈保护
84     MOV CX,40 ;将40送入CX, CX=循环次数
85 LOOP_1:
86     LEA SI,Buffer ;SI=Buffer (显示缓冲区) 偏移地址
87     CALL Display8 ;显示
88     loop LOOP_1 ;循环显示20次
89
90     POP SI ;SI出栈还原
91     POP DI ;DI出栈还原
92     POP CX ;CX出栈还原
93
94     POP AX ;AX出栈还原
95     POP DX ;DX出栈还原, 恢复现场
96     IRET ;中断返回
97
98 MAIN ENDP ;START过程结束
99 CODE ENDS ;代码段结束
100 END MAIN ;源代码结束

```

## 5.2 实验现象

当正确设置 8 个按键开关的数值后, 来回拨动单脉冲开关, 数码管上的数值逐渐变大, 拨动一次, 变化一次, 实现计数显示

## 6 思考题

1. 绘制本实验的详细实验电路图

答: 如图2

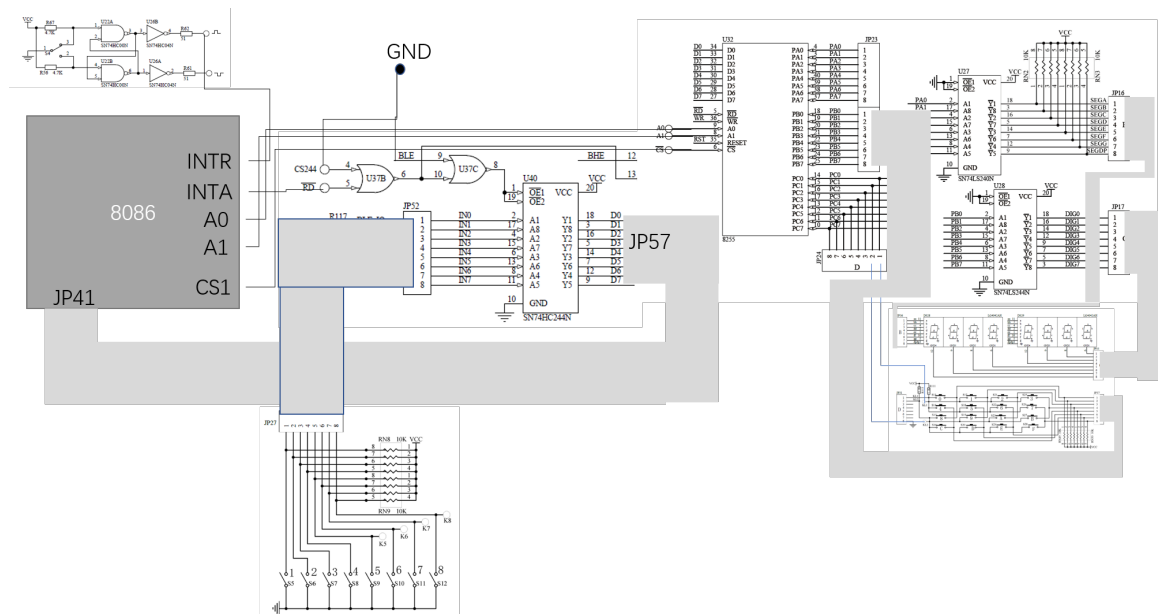


Figure 2: 实验连线

2. 扼要注释每指令的功能

答: 见实验代码部分