# 中国科学技术大学计算机学院《计算机系统概论》实验报告



LAB 05: Interrupt a Running Program

姓名:王晨 学号:PE20060014

完成日期:2021年1月12日

#### 一、实验要求:

- 1. Write the user program described below.
- 2. Write the keyboard interrupt service routine described below.

#### 二、实验环境:

Mac OS

LC-3 Simulator Mac Version

#### 三、算法思路:

由于本实验已经给了Starter Code,帮我们完成了系统初始化和中断使能,因此只需要完成用户程序和中断响应程序即可,这就比较简单了。

User Program部分是完成循环输出ICS的功能,每输出一次后使用DELAY子函数延迟,延迟间隔为x7FFF+x7FFF。

Interrupt Service Routine(ISR)部分将使用R0,R1,R7的寄存器,因此需要提前保存和恢复。进入中断程序后,会判断KBDR内的字符是否在x30~x39之间,若是则用TRAP指令PUTS .STRINGZ " is a decimal digit.",否则PUTS .STRINGZ " is not a decimal digit."即可。注意TRAP指令会调用R7寄存器的返回值,因此在TRAP前需要保存SaveR7以便ISR可以正常RTI。

由于以上两部分的程序思路和逻辑都比较简单,就不再画流程图了。

## 四、程序代码及注释:

1.User Program Part

```
48
49
                 ; *** Begin user program code here ***
50 LOOP
                LEA R0,ICS2020
51
                PUTS
52
                 JSR DELAY
53
                BR LOOP
54
                HALT
                 .STRINGZ "ICS2020 "
55 ICS2020
56
                 ;*** Delay Subroutine ***
57
    DELAY
                 ST R1, Save_R1
58
                LD R1, COUNT
59
    REP1
                 ADD R1.R1.#-1
60
                BRp REP1
61
                 LD R1, COUNT
62
    REP2
                 ADD R1,R1,#-1
                                         ;Repeat count once more to delay longer
63
                 BRp REP2
64
                LD R1, Save_R1
65
                RFT
66 COUNT
                .FILL x7FFF
67
                 .BLKW #1
    Save_R1
68
                 .END
```

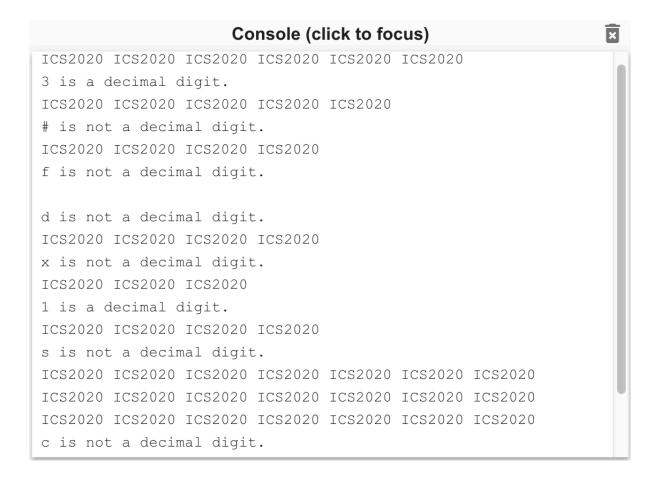
#### 2. ISR Part

这部分判断KBDR中输入字符是否为decimal number的方法是依次与-x30和-x39相减,如果不在这时间则进入OUTRANGE进行相应输出。每次输出prompt前后都会打印x0A换行。

```
70
                  .ORIG x1000
                   ; *** Begin interrupt service routine code here ***
 71
                  ST R0, SaveR0
 72
                  ST R1, SaveR1
ST R7, SaveR7
 73
 74
 75
                  LD RØ, NEWLINE
                  OUT
 76
     CHECK
                  LDI R0,_KBDR
 77
                                            ;to check if KBDR is within the decimal num range
 78
                  LD R1,NEG_x30
                                            ; RO should be within x30~x39
 79
                  ADD R1,R1,R0
                  BRn OUTRANGE
 80
                  LD R1,NEG_x39
 81
                  ADD R1,R1,R0
 82
 83
                  BRp OUTRANGE
 84
                  OUT
                  LEA RO DIGTALPMT
 85
 86
                  PUTS
                  BR DONE
 87
 88
     OUTRANGE
                  OUT
 89
                  LEA RØ NONDIGTALPMT
 90
                  PUTS
 91
     DONE
                  LD R0, NEWLINE
 92
                  OUT
 93
                  LD R0,SaveR0
                  LD R1, SaveR1
LD R7, SaveR7
 94
 95
96
                  RTI
                     .STRINGZ " is a decimal digit."
 97
     DIGTALPMT
                    .STRINGZ " is not a decimal digit."
 98
     NONDIGTALPMT
 99
     NEWLINE
                   .FILL x0A
100
     NEG_x30
                   .FILL xFFCF
                                       ;nagetive x30 (ASCII of 0)
     NEG_x39
SaveR0
101
                    .FILL xFFC6
                                        ;nagetive x39 (ASCII of 9)
                    .BLKW #1
102
                    .BLKW #1
103
     SaveR1
104
     SaveR7
                    .BLKW #1
105
     _KBSR
                   .FILL xFE00
106
     _KBDR
                    .FILL XFE02
107
                  . END
```

## 五、调试和测试:

运行过程中Console输入输出如下,和实验要求中一致:



# 六、实验心得:

通过本实验对中断程序的运行机制加深了理解。由于在ISR中的输出是通过TRAP指令实现的,因此大大简化了程序,但是要注意TRAP程序会修改R7寄存器,因此必须保存现场以便ISR正常返回到USER PROGRAM。如果不用TRAP程序,那么就要通过DSR,DDR来实现输出了,这样就更复杂些,也是可行的方法。

系统初始化、中断向量表、中断使能、保存PSR和PC都由STARTER CODE完成了, 因此本实验比较简单,实际这部分的程序也应该是由自己实现,这样才能理解中断调 用的完整流程。