

Lab 6 report

PB22111711 陈昕琪

实验目的与内容

- 要求实现用户输入/输出，中断驱动等功能
- 判断输入是否合法，并输出相应的字符串，进行程序的终止或者继续

逻辑设计

分为三个部分：用户程序、中断服务程序、代码开头配置（实验文件中已给出）

- 用户程序：用计数器实现不断输出学号，当用户输入时则开始用户程序
- 中断服务程序：要检测输入的字符是否正确，是否为数字，然后进行相应的跳转和计算

程序代码分析

• 用户程序：

使用一个循环计数器来计数，到达计数器数值为负说明到达计数值，并输出学号。

```
.ORIG x3000
; *** Begin user program code here ***
JSR    DELAY
DELAY  ST    R1, SAVER1
      LD    R1, COUNT
REP    ADD    R1, R1, #-1    ;循环计数
      BRp   REP
      LEA   R0, STUID        ;输出
      PUTS
      LD    R1, SAVER1
      RET

COUNT .FILL #15000          ;每到15000就输出
SAVER1  .FILL x5000
STUID   .STRINGZ "PB22111711 "
; *** End user program code here ***
.END
```

• 中断服务程序

- CHECK**:检测输入的值是否是数字，如果是数字则将数字的值保存到 **R0** 中，后续再进行判断，如果不是数字的值则直接跳转到 **ERROR**。

输入的值存在R0中是ASCII值，要判断其是否在'0'和'9'之间，来判断是否输入正确，这里用R5来存储ASCII值

对于存在R5中的值，还要与'8'比较，判断是否溢出，只有当键入的值符合要求才跳转到 **CALC** 输出阶乘

```
CHECK  ADD    R5, R0, #0    ;检查是否为数字
```

```

ADD    R4, R0, #0
LD     R3, ASCIIZ
ADD    R0, R0, R3    ; N-'0'
BRn    ERROR        ;不是数字
LD     R3, ASCIIIN
ADD    R0, R5, R3    ; N-'9'
BRp    ERROR        ;不是数字
LEA    R0, OUTPUT2
PUTS
LD     R0, NEWLINE
OUT
LD     R3, ASCIIIE
AND    R5, R5, #0
ADD    R5, R5, R4
ADD    R0, R5, R3    ; N-'8'
BRzp   OVERFL       ;溢出
AND    R0, R0, #0
LD     R3, ASCIIZ
ADD    R0, R4, R3    ;存储数字值
BRnzp  CALC

```

2. **ERROR** 和 **OVERFL**：判断出输入不合法后，输出相应的语句，其中 **ERROR** 结束后会继续进行用户程序，而 **OVERFL** 会直接结束程序。

```

ERROR   LEA    R0, OUTPUT1
        PUTS
        LD     R0, NEWLINE
        OUT
        RTI
OVERFL  AND    R0, R0, #0
        ADD    R0, R0, R4
        OUT
        LEA    R0, OUTPUT4
        PUTS
        LD     R0, NEWLINE
        OUT
        HALT

```

3. **FACT**：输出阶乘值，对于存储在R0中的数字值，我们对其进行判断，并跳转到相应的语句进行输出阶乘值。

对于阶乘，阶乘函数可以用递归，但是在Ic3中实现乘法递归有些困难而且复杂，所以可以用循环来实现，两层循环，一层用加法实现乘法，一层用于递增实现阶乘。

对于转换部分可以逐位判断，目前合法的阶乘数只有四位，所以只需每位判断后输出即可，判断每位可以用循环，并将数位的值存在寄存器中，输出时要注意ASCII码需要加'0'才能输出正确的值。

由于R0用于输出，要将数字值存储在R2寄存器中

```

FACT   AND    R2, R2, #0
        ADD    R2, R0, #0
        ADD    R0, R4, #0
        OUT
        LEA    R0, OUTPUT3
        PUTS

```

```

        AND R5, R5, #0
        ADD R5, R5, #1
        AND R4, R4, #0
LOOP1
        ADD R4, R4, #1
        NOT R3, R4
        ADD R3, R3, #1
        ADD R3, R2, R3
        BRn QDEC
        AND R6, R6, #0
        ADD R6, R6, #1
        ADD R0, R5, #0
LOOP2
        NOT R7, R6
        ADD R7, R7, #1
        ADD R7, R7, R4
        BRnz LOOP1
        ADD R5, R5, R0
        ADD R6, R6, #1
        BRnzp LOOP2

QDEC    LEA    R4, RESPTR    ;
DLOOP   AND    R3, R3, #0
DIVL    ADD    R3, R3, #1    ; R3 helps store subtraction rest so we can do the
next loop
        ADD    R5, R5, #-10   ; subtract 10 from R5
        BRzp   DIVL
        ADD    R5, R5, #10    ; get the lowest digit(decimal)
        LD     R2, OUTASC     ; convert to ASCII
        ADD    R5, R2, R5
        STR    R5, R4, #-1    ; store the digit
        ADD    R4, R4, #-1    ; move the pointer
        ADD    R5, R3, #-1    ; continue to loop
        BRp    DLOOP
        ADD    R0, R4, #0     ; set R0 to the (string)fact(N)
        PUTS                               ; print the factorial as a string
        LD     R0, BKSPACE    ; delete the tail of string
        OUT
        HALT

```

测试结果与分析

我的学号是PB22111711

测试结果图如下:

PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711

a is not a decimal digit.

PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711

b is not a decimal digit.

PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711

9 is a decimal digit.

9 is too large for LC-3.

--- Halting the LC-3 ---

非法输入和溢出

PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711

5 is a decimal digit.

5!=120

--- Halting the LC-3 ---

正确输入5

- -
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711

7 is a decimal digit.

7!=5040

--- Halting the LC-3 ---

正确输入7

```
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711 PB22111711 PB22111711
PB22111711 PB22111711 PB22111711
0 is a decimal digit.
0!=1
```

--- Halting the LC-3 ---

正确输入0

由此可见程序正确。

遇到的问题及反思

本次实验过程中，由于一些细节没有注意到，在编写程序时出现一些错误。

首先是对于R0的运用，键入和输出的值都储存在R0中，所以每次想要对R0中的值进行判断都需要将其寄存的值存进其他寄存器并进行应用。

其次是判断时的一些小细节，应当先判断是否为数字，再判断是否为溢出数字，刚开始写程序时这里写的有一点乱，导致程序无法正确输出，改进后输出正确。

总结

1. 本次实验，通过编写 `lc3` 程序，深入了解了 `lc3` 中输入输出相关操作的用法。
2. 熟练掌握输出输入以及判断字符等操作。能够运用已学的知识实现函数递归操作。
3. 对于程序的整体规划和分块思想有了更深入的了解。