

实验五：Interrupt a Running Program

实验要求：利用中断服务知识编写一个运行时可以用键盘打断的程序，同时在中断结束时恢复各项寄存器的状态

一、算法设计

由于 interrupt 程序的 system 部分已经完成，则只需要编写系统服务程序和用户服务程序即可。

用户部分：

无限循环，将“ICS 2020 ”输出到显示器上，利用 DELAY 程序控制输出时间间隔为 2500 个加法周期。

系统服务程序：

（由于初始时 KBSR 【14】 已经被置为 1，所以只要用户敲击键盘，中断程序就会执行，因此不需要写调用指令）

- 1、保存现场
- 2、使用 GETC 将 M[KBDR]→R0
- 3、若 $R0 > 74$ 且 $R0 < 65$ ，提示错误信息；否则提示正确信息

二、程序实现

用户程序部分：（起始地址为 x3000）

①将“ICS 2020 ”字符串的首地址存入 R0

②使用 PUTS 打印字符串，然后执行①

中断服务部分：（程序起始地址为 x1000）

①将需要用到的寄存器进行保存（保护现场）

②打印换行（\n），然后利用 GETC 指令将输入字符读入 R0

③将 R0 的数值减去 48（0 的 ASCII 码），若结果小于 0，则执行“错误提示操作”；否则将 R0 数值再减 9，若结果大于 0，则执行“错误提示操作”；则执行“正确提示操作”

④打印换行（\n），然后使用 RTI 返回

错误提示：在显示器上输出“M[R0] is not a decimal digit”

正确提示：在显示器上输出“M[R0] is a decimal digit”

三、程序检验

程序运行截图如下：

Console (click to focus)



```
ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020
ICS2020 ICS2020 ICS2020 ICS2020 ICS2020
A is not a decimal digit
ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020
ICS2020
1 is a decimal digit
ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020
3 is a decimal digit
ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020
# is not a decimal digit
ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020
% is not a decimal digit
ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020 ICS2020
ICS2020 ICS2020
8 is a decimal digit
ICS2020 ICS2020 ICS2020 ICS2020
```