[**LAB5 - Interrupt**](http://acsa.ustc.edu.cn/ics/downloads/LAB4.pdf)

**Background**

In computer science, an interrupt is a signal to the processor emitted by hardware or software indicating an event that needs immediate attention. An interrupt alerts the processor to a high-priority condition requiring the interruption of the current code the processor is executing, the current thread. The processor responds by suspending its current activities, saving its state, and executing a small program called an **interrupt handler**(or interrupt service routine, ISR) to deal with the event. This interruption is temporary, and after the interrupt handler finishes, the processor resumes execution of the previous thread. There are two types of interrupts: hardware interrupt and software interrupt.

**Description**

The purpose of this assignment is to show how interrupt-driven Input/Output can interrupt a program that is running, execute the interrupt service routine, and return to the interrupted program, picking up exactly where it left off (just as if nothing had happened). In this assignment, we will use the Keyboard as the input device for interrupting the running program.

The assignment consists of three parts:

**A.The user program**

Your user program will consist of continually producing the "ICS checkerboard" by alternately outputing two different lines, which looks like:

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS ICS

ICS ICS ICS ICS ICS

To ensure the output on the screen is not too fast to be seen by the naked eye, the user program should include a piece of code that will count down from 2500 each time a line is output on the screen. A simple way to do this is with the following subroutine DELAY:

DELAY ST R1, SaveR1

LD R1, COUNT

REP ADD R1,R1,#-1

BRp REP

LD R1, SaveR1

RET

COUNT .FILL #2500

SaveR1 .BLKW 1

**B.The keyboard interrupt service routine**

The keyboard interrupt service routine will simply write to the screen ten times whatever key the person sitting at the keyboard typed, followed by a Enter(x0A).

**You may not use any TRAP instructions in your interrupt service routine.** To display a character to the screen, you must poll the DSR and then write to the DDR, you may not call TRAP x21(OUT), or use any of the other TRAP routines. *Note: Don't forget to save and restore any registers that you use in the interrupt service routine.*

And don't forget to save and restore registers that you used in the service routine.

**C.The operating system enabling code**

Unfortunately, we have not YET installed Windows or Linux on the LC-3, so we are going to have to ask you to do the following three enabling actions in your user program first.

* Normally, the operating system would have previously set up some stack space so that the PC and PSR can be pushed when an interrupt is encountered. (As you know, when the service routine executes RTI, both PC and PSR will be popped, returning the machine to the interrupted program.) Since there is no operating system, please initialize R6 to x3000, indicating an empty stack.
* Also, normally, the operating system establishes the **interrupt vector table** to contain the starting addresses of the corresponding interrupt service routines. You will have to do that for the keyboard interrupt. The starting address of the interrupt vector table is x0100 and the interrupt vector for the keyboard is x80. It is necessary for you to only provide the one entry in the interrupt vector table that is needed for this programming lab assignment.
* Finally, normally, the operating system would set the IE(Interrupt Enable) bit of KBSR. You need to do that as well.

**Your job**

Your job will be to write both the user program augmented with the interrupt enabling code described above and the keyboard interrupt service routine. The user program named lab4\_ID\_user\_program will be of the form:

.ORIG x3000

-- --- ; initialize the stack pointer

...

-- ---

-- --- ; set up the keyboard interrupt vector table entry

...

-- ---

-- --- ; enable keyboard interrupts

...

-- ---

-- --- ; start of actual user program to print ICS checkerboard

...

-- ---

.END

The interrupt service routine must be named lab4\_interrupt\_service\_routine.asm and will be of the form:

.ORIG x2000

-- --- ; the code

...

-- ---

RTI

-- --- ; buffer space as required

...

-- ---

.END

**Note**

* The Linux LC-3 simulator does NOT include support for interrupts. Therefore, you must use the Windows LC-3 simulator for this assignment.
* Refer to textbook chapter 8.5,chapter 10 and appendix A