EE306: Programming Assignment 5

Course Instructor: Dr. Nina Telang
TAs: Vignesh Radhakrishnan, Jefferson Lint, Suhas Raja, Jerry Yang

All Lab assignments must be completed individually. You are not permitted to seek help or clarification from anyone other than the instructor or the TAs.

Due date: 12/05, 9:00 PM

Turn in your submission on github. This lab will be checked out during office hours/recitation after the deadline has passed. The checkout times will be announced soon.

Problem statement

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.

To demonstrate this, we will need three things:

1. The interrupt vector table has to be generated, interrupts will need to be enabled, and there will have to be space on the interrupt stack to save the state (PC and PSR) of interrupted routines while they are waiting to resume execution. The operating system will allocate this space, set up the interrupt vector table and enable the interrupts.

2. A program that will run at priority level 0.

3. An interrupt service routine that will be invoked as a result of someone hitting a key on the keyboard.

The user program.

The user program will print the prompt “TEXAS ENGINEERING” followed by a newline character (x0A).The user program is to do this in an infinite loop.

In order to make sure the prompts don't get written to the monitor too fast to be seen by the naked eye, the user program should include a piece of code that will count down from say x5000 each time a character is output to the screen. **You may want to try out other numbers if your processor is too fast or slow.**

The keyboard interrupt service routine.

The keyboard interrupt service routine will read the character typed, prints a newline and performs one of the following tasks based on the type of character:

1. Upper Case Alphabet, then print the string “WHAT STARTS HERE CHANGES THE WORLD” and then a newline.

2. Lower Case Alphabet, then print “Hook’em, Horns!” and then a newline.

3. Number between 0 and 9, then print “TEXAS FIGHT” and then a newline

4. For any other character, print “END OF EE 306 LABS”, a newline and Halt the processor.

VERY IMPORTANT: **You are not allowed to use any TRAP instruction other than HALT 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. If you use TRAP in the interrupt service routine or if you do not properly poll the DSR before writing to the DDR, your program is not correct and will not be graded ***even though it may appear to work.***

Hint: Don't forget to save and restore any registers that you use in the interrupt service routine.

The operating system enabling code.

We are going to require you to do the enabling actions that the operating system would do before you start to execute the User program.

1. Normally, the operating system will set up stack space for pushing the state of the running process, so it can return to it after the service routine has finished executing. In this lab, you will need to set up the stack space. The base of the stack needs to be x4FFF, so the stack pointer (i.e. R6) must be initialized to x5000, indicating an empty stack.

2. Normally, the operating system establishes the interrupt vector table to contain the starting addresses of the corresponding interrupt services 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.

3. Normally, the operating system would set KBSR[14] to enable the keyboard to interrupt user programs. You will have to do that.

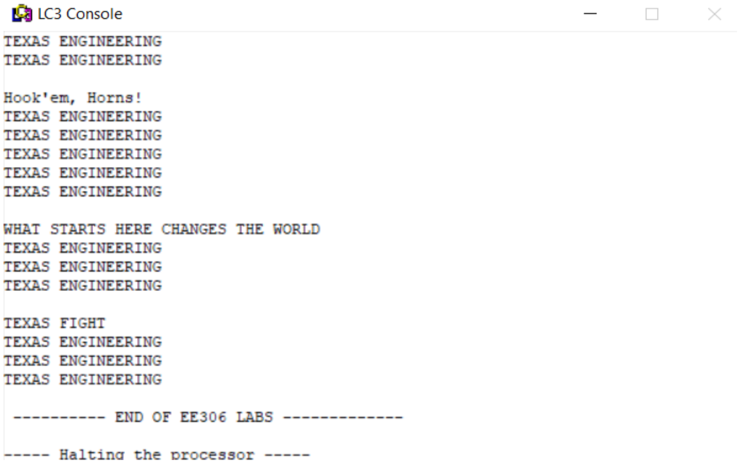
You must perform these three tasks as part of your user program.

Your job will be to write the user program augmented with the interrupt enabling code described above and the keyboard interrupt service routine.

The user program must be named **lab5main.asm**.

The interrupt service routine must be named **lab5isr.asm**.

An example of how the console should look when you run the program:



```
TEXAS ENGINEERING
TEXAS ENGINEERING

Hook'em, Horns!
TEXAS ENGINEERING
TEXAS ENGINEERING
TEXAS ENGINEERING
TEXAS ENGINEERING
TEXAS ENGINEERING

WHAT STARTS HERE CHANGES THE WORLD
TEXAS ENGINEERING
TEXAS ENGINEERING
TEXAS ENGINEERING

TEXAS FIGHT
TEXAS ENGINEERING
TEXAS ENGINEERING
TEXAS ENGINEERING

 ---------- END OF EE306 LABS -------------

----- Halting the processor -----
```

Figure 1: Sample Output - Key Sequence: c, A, 9,]