

CSCI 240 - Computer Organization and Assembly Language Programming

Homework 8 - I/O

Problem 8.2. Why is a Ready bit not needed if synchronous I/O is used?

Problem 8.3. What is the purpose of bit [15] in the KBSR?

Problem 8.5. What problem could occur if a program does not check the Ready bit of the KBSR before reading the KBDR?

Problem 8.8. Write a program that checks the initial value in memory location `x4000` to see if it is a valid ASCII code and if it is a valid ASCII code, prints the character. If the value in `x4000` is not a valid ASCII code, the program should print nothing.

Problem 8.11. Which is more efficient, interrupt-driven I/O or polling? Explain.

Problem 8.14. An LC-3 Load instruction specifies the address `xFE02`. How do we know whether to load from the KBDR or from memory location `xFE02`?

Problem 8.15. Interrupt-driven I/O:

(a) What does the following LC-3 program do?

```

        .ORIG  x3000
        LD     R3, A
        STI    R3, KBSR
AGAIN   LD     R3, A
        TRAP   x21
        BRnzp AGAIN
A       .FILL  x4000
B       .FILL  x0032
KBSR    .FILL  xFE00
        .END

```

(b) If someone strikes a key, the program will be interrupted and the keyboard interrupt service routine will be executed as shown below. What does the keyboard interrupt service routine do?

```

        .ORIG  x1000
        LDI    R0, KBDR
        TRAP   x21
        TRAP   x21
        RTI
KBDR    .FILL  xFE02
        .END

```

(c) Finally, suppose the program of part *a* started executing, and someone sitting at the keyboard struck a key. What would you see on the screen?

Problem 8.16. What does the following LC-3 program do?

```
.ORIG x3000
LD R0, ASCII
LD R1, NEG
AGAIN LDI R2, DSR
BRzp AGAIN
STI R0, DDR
ADD R0, R0, #1
ADD R2, R0, R1
BRnp AGAIN
HALT
ASCII .FILL x0041
NEG .FILL xFFB5; -x004A
DSR .FILL xFE04
DDR .FILL xFE06
DDR .FILL xFE06
.END
```