Stamatios Morellas
COM S 352 - Homework 1
Due 1/24/20 @ 11:59pm

1. (10 points) What is the purpose of interrupts? What is a trap? Can traps be generated intentionally by a user program? If so, for what purpose?

Interrupts can be used to impact the instruction cycle of a processor. Their purpose is to allow processors to execute other instructions, while other I/O operations are occurring in the background. A trap is a type of interrupt that is generated by software, not a user. However, a user could purposely generate a trap for the purpose of background error handling as well as operating system routines.

2. (10 points) Describe the dual-mode operation of CPU. What is the reason to distinguish between the two modes?

Dual-mode operation allows an OS and other programs to make themselves more secure from malware and less-likely to be affected by such things. The two modes are named “User mode” and “Kernel mode.” Some instructions are designed to hold a higher priority over others, which means that they need a special mode to run in; kernel mode. Privileged instructions can cause harm to user programs, so they have to run in a separate cpu mode to eliminate risk. All other user instructions are designated to run in user mode.

3. (10 points) Which of the following instructions should be privileged?

a. Set value of timer

The OS setting the counter is a privileged instruction, so setting the value of the timer (counter) is also a privileged instruction.

b. Read the clock

Not privileged.

c. Clear memory

Privileged.

d. Issue a trap instruction

Not privileged; since a trap will cause the OS to put the CPU in kernel mode, this means that from that point on, privileged instructions will be able to be called.

e. Turn off interrupts

Not privileged.

f. Put the CPU in kernel mode

Not privileged.

g. Access I/O device

Privileged.

4. (10 points) A computer has a CPU that can execute 10 million instructions per second and a memory has a transfer rate of 100 million bytes per second. When interrupt-driven I/O is performed, the ISR has to execute 50 instructions to transfer one byte between memory and I/O device. What is the maximum data transfer rate during I/O operations implemented by using interrupt-driven I/O?

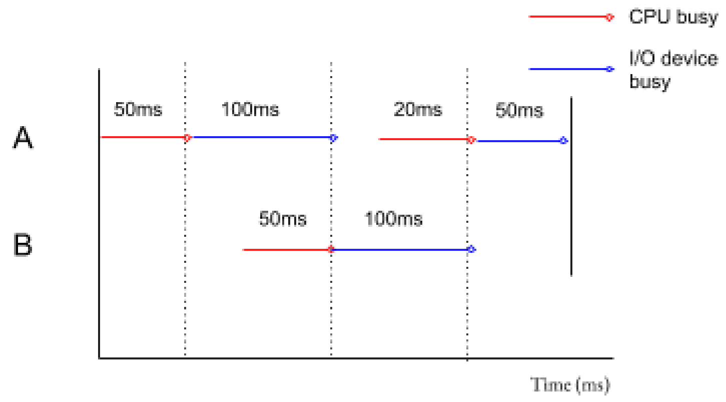
$$Maximum\ Data\ Transfer\ Rate\ = \frac{1\ byte}{50\ inst} * \frac{10{,}000{,}000\ inst}{1\ sec} * \frac{100{,}000{,}000\ bytes}{1\ sec} = 2 \times 10^{13}\ \frac{bytes}{sec}$$

5. (10 points) Three programs are serviced in a multiprogramming system. Program 1 contains *50ms* of computation followed by *100ms* of I/O on device 1. Program 2 contains *20ms* of computation followed by *50ms* of I/O on device 1. Program 3 contains *50ms* of computation followed by *100ms* I/O on device 2. How long it will take to complete all three programs? You should draw a diagram similar to the one given in the lecture notes to show the operation of the three programs. (A device can service one I/O request at a time.)

Program 1: 50ms + 100ms$_{device1}$ = 150ms

Program 2: 20ms + 50ms$_{device1}$ = 70ms

Program 3: 50ms + 100ms$_{\text{device2}}$ = 150ms



Total: 50ms + 100ms + 100ms + 50ms = *300ms*

6. (10 points) Describe why DMA is considered a more efficient mechanism for performing I/O than interrupt-driven I/O.

Direct memory access is more efficient than interrupt-driven I/O because the CPU is able to work on other tasks while data transfers are in-progress. DMA interrupts the CPU only when the entire block of data has been transferred.

7. (10 points) Describe three general methods for passing parameters to the operating system.

1. Pass the parameters as registers
    a. There may be more parameters as registers
2. Parameters stored in a block in memory, and address of block passed as a parameter in a register
3. Parameters pushed onto the stack by the program and popped off the stack by the OS