EEL4768.04 Homework 6 Due 11/26/19

Brandon Thompson 5517

November 26, 2019

1. What are interrupts?
   Interrupts are signals to the processor by either hardware or software that indicate an event needs immediate attention.

2. How does embedded system/processor handle interrupts?

   - Micro-controller stops the current instruction and saves the PC address.
   - Saves current status of all interrupts internally (not stack).
   - Jumps to memory location of the interrupt of vector table that holds the address of the interrupts service routine.
   - Micro-controller gets the address of the ISR from the interrupt vector table and jumps to it. It starts to execute the interrupt service routine, which is RETI (return from interrupt).
   - Upon exiting the RETI instruction, the micro-controller returns to the location where it was interrupted. First, it gets the program counter from the stack by popping the top bytes of the stack into the PC. Then start to execute from that address.

3. What are multiple interrupts?
   When an interrupt is triggered during another interrupt or if more than one is passed to the micro-controller at the same time.

4. How does embedded systems/processor handle multiple interrupts?
   It would follow the above process until another interrupt is triggered, then it will check the priority of the each interrupt and execute the one that is ranked higher, then follow with the lower ranked interrupt.

5. What is I/O interfacing?
   A way of connecting the outside world to a processor using a input device to gather info or an output device to retrieve data from the processor.

6. How can we interface I/O device with a processor?

   - Memory mapped I/O: Load or store can either access a memory location or an I/O device

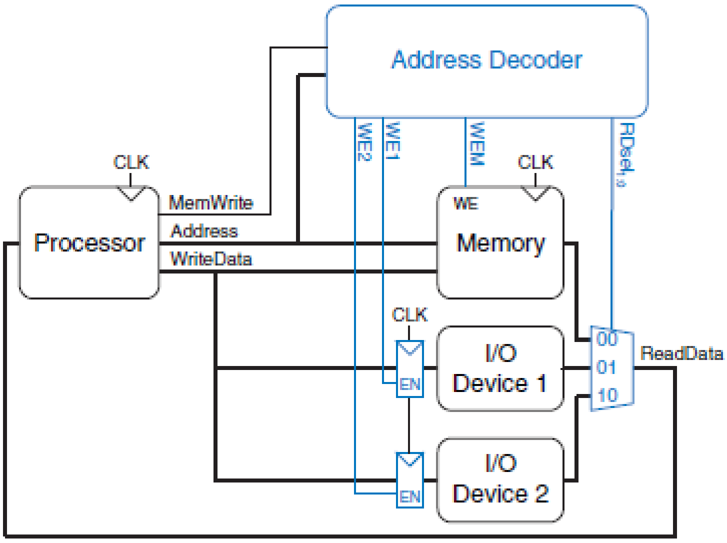

Figure 1: Support hardware for memory-mapped I/O.