Merrick, David

20 October, 2012

Lab 5 PreLab

1. Polling is when a processor periodically checks a device or register for a change in status. Polling is synchronous. While the processor is waiting for a change in the status, it does not execute other instructions. In an interrupt, an event external to the currently executing process causes a change in the flow of execution. Interrupts are asynchronous. At the end of each instruction execution, the processor checks for interrupts. If there is an interrupt, the processor pushes the return address onto the stack, determines the source of the interrupt, executes the interrupt code, then pops the address off the stack and returns to normal execution. This allows the processor to execute other instructions instead of sitting idle. Polling is ideal when a processor has to respond to an event as soon as possible, and interrupts are ideal when a processor has other work to do and response time is not critical. An ideal situation for polling would be an anti-lock braking system in a car. An ideal situation for an interrupt would be a TekBot that plays an MP3 and the remote control triggers an interrupt that causes it to move.
2. EICRA (External Interrupt Control Register A) and EICRB (External Interrupt Control Register B): These 8-bit registers set up the way that an interrupt is triggered. This could be a falling clock edge, rising edge, or low-level input signal. EIMSK (External Interrupt Mask Register): Masks the interrupt signals to allow each interrupt to be detected or not detected.
3. Interrupt vectors are the first 70 locations in the Program Memory dedicated to interrupts. Timer/Counter 2 Comparison match: $0012. External Interrupt 2: $0006. USART1-Rx Complete: $003C.
4. A. 8, 23. B. 4, 20. C. 1-2, 9-19, 24-27.