

- 1) Interrupts are best used for asynchronous, urgent, and infrequent events. This involves an interrupt flag being set up, and when that flag is set a procedure begins immediately. Polling routinely checks for flags being set, and executes a routine upon finding a set flag. Because they are not immediate, they are best used for non-urgent tasks.
- 2) Interrupts
 - a. EICRA: (INT 0->3)
 - i. 0 bit -> Whether the low level of INTx generates an interrupt request.
 - ii. 1 bit -> Any logic change on INTx generates an interrupt request.
 - iii. 2 bit -> The falling edge of INTx should generate an interrupt request.
 - iv. 3 bit -> The rising edge of INTx should generate an interrupt request.
 - b. EICRB: same as EICRA but for INTs 4->7
 - c. EIMSK: in this register, a bit is set to enable the related interrupt. For example, if bit 2 is set the INT2 interrupts are enabled.
- 3) An interrupt vector is the memory location of an interrupt handler, which allows us to queue up interrupts.
- 4) Interrupt Vectors:
 - a. Timer/Counter0 Overflow -> TIM0_OVF_vect
 - b. External Interrupt 5 -> INT5_vect
 - c. Analog Comparator -> ANALOG_COMP_vect
- 5) Interrupt Detection
 - a. Rising edge 6, 18
 - b. Falling edge 3, 9
 - c. Low level 4, 5, 10-17
 - d. High level 1,2,7-9, 19-21

References:

<https://sites.google.com/site/qeewiki/books/avr-guide/external-interrupts-on-the-atmega328>

<http://stackoverflow.com/questions/3072815/polling-or-interrupt-based-method>

<http://www.cse.unsw.edu.au/~cs9032/slides/wk7.pdf>

