ECE 375 LAB 6

External Interrupts

Lab Time: Thursday 1000-1200

Eric Prather

PRELAB

1. In computing, there are traditionally two ways for a microprocessor to listen to other devices and communicate: polling and interrupts. Give a concise overview/description of each method, and give a few examples of situations where you would want to choose one method over the other.

Polling is a deliberate *check* by a device to check a data value, which can then be interpreted by the device and result in a branching path of behaviors. Thus, for a specific behavioral branch to occur, the device must poll something like a peripheral. Hypothetically, if the polling interval was wider than the frame in which the data changes, some data fluctuations may then be missed. If timing is of essence and the polling interval is wide, this is also probable. One solution to this problem would be for the peripheral to store a set of data, rather than the instananious value, which the poller can review upon data retrieval so as not to miss any queues. Polling is useful in situations where immediate reations are important and data does not need to be missed. For example, an automatic door could poll its motion sensor every second, as people approaching the door who need to be let in will not likely move in instantaneous chunks and freeze during the polling times.

Interrupts are an action done *to* the microprocessor by an interrupt signal sent by another device. This results in a literal, physical modification to the interpretation of the program controller operates, causing it to disregard its regular interpretation of instructions in its program to execute a certain hardwired behavior corresponding to the interrupt vector's design. This is instantaneous. Interrupts are useful when the priority of the interrupting activity is high. For example, this may be useful to prevent a failure of a high-precision sub-millimeter CNC machine which must respond instantly and appropriately on a scale of microseconds- or even seconds- due to the sensitive nature of the microscopic physical world.

2. Describe the function of each bit in the following ATmega128 I/O registers: EICRA, EICRB, and EIMSK. Do not just give a brief summary of these registers; give specific details for each bit of each register, such as its possible values and what function or setting results from each of those values. Also, do not just directly paste your answer from the datasheet, but instead try to describe these details in your own words.

EICRA & EICRB: External interrupt control register A and B. Setting the pertinent bits, as defined by the datasheet, will allow for various interrupt handling behaviors to occur. There are four settings supported:

Falling edge interrupts- interrupts occur on the falling edge of the pertinent signal (after it was high)

rising edge interrupts- interrupts occur on the rising edge of the pertinent signal (after it was low)

low level- interrupts occur perpetually while vector is low

all interrupts- any changes in logic (i.e. falling or rising edges) will trigger an interrupt.

Each of four bits of these registers will control one setting to be on or off. The bit being actively considered is determined by EIMSK.

EIMSK: External interrupt mask. Allows the processor to selectively override interrupt controllers from the four cached settings, described above. Thus, only one of the four bit values defined above is pertinent to the board at a time, as EIMSK only has two bits with which to define which behavior Boolean to target.

3. The ATmega128 microcontroller uses interrupt vectors to execute particular instructions when an interrupt occurs. What is an interrupt vector? List the interrupt vector (address) for each of the following ATmega128 interrupts: Timer/Counter0 Overflow, External Interrupt 5, and Analog Comparator.

An **interrupt vector** is a means by which a signal generated by an external device can cause a change in behavior of the processor. The ATMEGA supports a variety of different input vectors, such as those built into the buttons on the board or those which can be defined manually.

Here are the requested addresses:

Timer/Counter0 Overflow: PG4 (port G pin 4) and PG3 (port g pin 3) both are used for this. The latter is simply the inverted of the former.

External Interrupt 5: PE5 (port e pin 5)

and Analog Comparator: ACI (Analog comparator interrupt, part of the analog comparator's dedicated register)

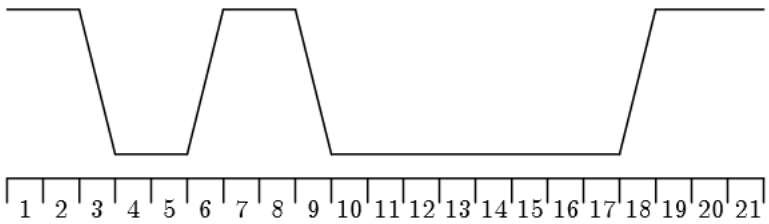

Figure 1: Sample Input to External Interrupt

4. Microcontrollers often provide several different ways of configuring interrupt triggering, such as level detection and edge detection. Suppose the signal shown in Figure 1 was connected to a microcontroller pin that was configured as an input and had the ability to trigger an interrupt based on certain signal conditions. List the cycles (or range of cycles) for which an external interrupt would be triggered if that pin's sense control was configured for: (a) rising edge detection, (b) falling edge detection, (c) low level detection, and (d) high level detection. Note: There should be no overlap in your answers, i.e., only one type of interrupt condition can be detected during a given cycle.

a. Rising edges: 6 and 18

b. Falling edges: 3 and 9

c. Low level detection: 4, 5, and 10-17

d. High level detection: 1, 2, 7, 8, 19, 20, and 21