ECE 375 PRELAB 5

Lab Time: Tuesday 4-6

Alexander Uong

QUESTIONS

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 described as when the CPU or microprocessor checks if a device changes so very often or needs attention. The CPU essentially abandons all other works and focuses on these devices, observing if they need any attention or assistance.

   Interrupts is when the device notifies the CPU or microprocessor that it requires attention. This notification to the CPU is through an interrupt signal. If the CPU receives an interrupt signal, it stops what it's currently doing and provides control of the device to the interrupt handler, assisting the device. The CPU is then able to return to what it was doing prior.

   Situations where polling should be used are when a certain event happens frequently or is expected. If an event happens frequently and is expected to occur, it makes more sense to use polling to constantly check the device and see if it requires attention. However, if an event is unexpected or may not occur often, interrupts would be the more efficient choice, as the CPU is able perform other tasks instead of checking on devices.

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: Known as External Interrupt Control Register A, this is an 8-bit register. External interrupts 3 through 0 are activated by the external pins INT3:0. Each bit 0 through 3 interrupts based on a type of signal. Bit 0 generates an interrupt request from a low level signal. Bit 1 is reserved. Bit 2 generates an interrupt request from a falling edge signal. Bit 3 generates an interrupt request from a rising edge signal.

   The bits are configured as followed:

Bit	7	6	5	4	3	2	1	0	
	ISC31	ISC30	ISC21	ISC20	ISC11	ISC10	ISC01	ISC00	EICRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

   EICRB: Known as External Interrupt Control Register B, this is an 8-bit register. External interrupts 7 through 4 are activated by external pins INT7:4. There's a slight difference between EICRB and EICRA; EICRB has a different interrupt sense control compared to EICRA. For example, ISC01 is that any logical change on INTn generates an interrupt request, whereas in EICRA, ISC01 is reserved.

   The bits are configured as followed:

Bit	7	6	5	4	3	2	1	0	
	ISC71	ISC70	ISC61	ISC60	ISC51	ISC50	ISC41	ISC40	EICRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

EIMSK: Known as External Interrupt Mask Register, this is an 8-bit register. When a bit is set to 1 and the status register is set to 1, the pin interrupt is enabled for that bit. This register is able to check which interrupts are enabled, with each bit set to 1 or 0 based on the interrupt.

The bits are configured as followed:

Bit	7	6	5	4	3	2	1	0	
	INT7	INT6	INT5	INT4	INT3	INT2	INT1	IINT0	EIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

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 the memory location of an interrupt handler. As described above, interrupts are when the device notifies the CPU or microprocessor that it requires attention. This notification to the CPU is through an interrupt signal. If the CPU receives an interrupt signal, it stops what it's currently doing and provides control of the device to the interrupt handler, assisting the device. The CPU is then able to return to what it was doing prior.

   Interrupt vectors:

   Timer/Counter0 Overflow: $0020

   External Interrupt5:$000C

   Analog Comparator:$002E

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 edge detection: Cycles: 6-7; 18-19

b) Falling edge detection: Cycles: 3-4; 9-10

c) Low level detection: Cycles: 4-6; 10-18

d) High level detection: Cycles: 1-3; 7-9; 19-22

REFERENCE

https://www.geeksforgeeks.org/difference-between-interrupt-and-polling/#:~:text=In%20interrupt%2C%20the%20device%20notices,whether%20the%20device%20needs%20attention.&text=In%20interrupt%2C%20the%20device%20is,device%20is%20serviced%20by%20CPU.

http://ww1.microchip.com/downloads/en/devicedoc/doc2467.pdf