Chapter 9 Reading Review

9.1. In terms of the nature of the data, how does a keyboard differ from a hard disk as an input device?

Keyboard: Input-only, unexpected input, limited to ASCII codes, typing in input for keyboards, very low data rate (limited by typing speed), external and program control, character data type

Hard disk: Storage, read/write, medium data rate, only program control, block/burst data type

9.2. Name at least two devices that can generate unexpected input.

Keyboard, voice, mouse

9.3. Explain the purpose of a buffer.

- Buffer is a small area in the computer's memory which is used to store or preserve data for a short time until it is used. Buffers are crated in RAM.

- Buffers are used for storing and holding the data so that the required manipulations or changes can be done on the data by the CPU before forwarding the data to devices or during data transfer between an application and a device.

9.3. DMA is rarely used with dumb computer terminals. Why?

- A dumb computer terminal is used consist of and keyboard/ mouse or some input device with a Monitor as the output device. It has verity limited processing ability and relays on the Host computer to process information.
- Most of these systems are connected to Main Frame computers.

9.4. Explain the reasons why programmed I/O does not work very well when the I/O device is a hard disk or a graphics display.

- Programmed I/O (Input/Output) is the simplest method of executing the input/output operations. Programmed I/O is not suitable or works well when the I/O device is a hard disk or a graphics display as Programmed I/O is a very time consuming method. While I/O devices are busy in executing the operations CPU periodically keeps on checking the status bits of the I/O devices to know whether these devices have got free or are still inputting the data. Thus, Programmed I/O is suitable only for slow devices where the input is made character by character for example a keyboard. Hard disks and graphic displays requiring very steady and fast inputs can’t be served well by the programmed I/O method.

9.5. When an interrupt occurs, what happens to the program that is currently executing at the time?

Interruption is the outcome from a device embedded to the system or from a program in the computer to stop the executing program. There are two kind of interrupts: Hardware interrupt, Software Interrupt. Hardware interrupt is a signal which alerts and informs the operating system about the occurrence of an interrupt by a hardware device. Software interrupt or Trap is a signal that is generated by a user program itself in case of a page fault. When an interrupt occurs, it causes the CPU to stop executing the current program. The control then jumps to a special piece of code called an Interrupt Handler in the kernel or Interrupt Service Routine which identifies the source and type of interrupt. The handler will then process the interrupt by using suitable actions or mechanism and after that it resumes the interrupted program.

9.6. What is a *context?* What does it contain? What is it used for?

Context is the information in the program’s memory that allows it to to restart and pick up exactly where it left off, without any loss of data or progress. The information kept includes the location of the last instruction executed, the values of data in various registers, which is all saved in a specific area in the program known as the process control block or PCB.

9.7) What is polling used for? What are the disadvantages of polling? What is a better way to perform the same job?

- **External event Notifier:** When an external event happens such as a user input, an interrupt frees the CPU to acknowledge and assess the new data. If the input demands action then the action will be carried out before returning control to the original program.
- **Completion signal:** Interrupts can allow programs to communicate as in the case of a completion interrupt. This type of interrupt can convey that one program is done with its current job and ready for the next. For example a printer can send an interrupt after printing a page to signify that it is ready for the next page.
- **Allocating CPU time:** When programs share CPU power they use interrupts to request CPU time. The CPU can only perform on set of instructions at a time.
- **Abnormal event indicator, wrong data, instructions fail, etc.**

9.8. To use a computer for multimedia (moving video and sound), it is important to maximize the efficiency of the I/O. Assume that the blocks of a movie are stored consecutively on a CD-ROM. Describe the steps used to retrieve the blocks for use by the movie display software. Discuss ways in which you could optimize the performance of the I/O transfer.

A software interrupt occurs. In IBM’s system z it would be classified as a program check. You would get an illegal instruction interrupt error.

9.9. Consider the interface between a computer and a printer. For a typical printout, it is clearly impractical to send output data to the printer one byte or one word at a time (especially over a network!). Instead data to be printed is stored in a buffer at a known location in memory and transferred in blocks to memory in the printer. A controller in the printer then handles the actual printing from the printer's memory. The printer's memory is not always sufficient to hold the entire printout data at one time. Printer problems, such as an "out of paper" condition, can also cause delays. Devise and describe, in as much detail as you can, an interrupt/DMA scheme that will assure that all documents will be successfully printed.

Direct memory Access: It allows it to process a lot more blocks from memory

9.10. The UNIX operating system differentiates between block-oriented and character-oriented devices. Give an example of each, explain the differences between them, and explain how the I/O process differs for each.

3 Primary Conditions:

- Method to connect IO interface to memory

- IO controller associated with device must be capable of reading/writing to memory

-Means to avoid conflict with CPU and IO controller

9.11. Describe a circumstance where an interrupt occurs at the beginning of an event. Describe a circumstance where an interrupt occurs at the completion of an event. What is the difference between the types of events?

The data that an I/O controller must have before a DMA transfer takes place is the location of data in the I/O device, starting location of memory, size of block, Direction of transfer (read or write). This data is sent to the controller through requests, which go through to the CPU first and then the CPU reads/writes the data from/to the memory in the order that it is received.

9.12. In gernal, what purpose does an interrupt serve? Stated another way, suppose there were no interrupts provided in a computer. What capabilities would be lost?

The purpose of a completion interrupt at the conclusion of a DMA is to simple signal for the beginning and ending of the interrupt.

9.13. What is the difference between polling and polled interrupt processing?

The two interfaces of a graphics card are connected to:

-AGP (Accelerated Graphics Port), an interface specification developed by Intel Corporation.

-PCI Express (Peripheral Component Interconnect Express)

9.14. Describe the steps that occur when a system receives multiple interrupts.

The benefits of I/O Controllers are:

-The controllers are designed to provide the control required by the device.

-The controller frees the CPU to perform other task while the much slower I/O operations are taking place.

-I/O controllers allow access to different I/O devices that occur at the same time.