

CPSC 331 - Operating Systems

Instructor Marc Corliss

Homework 5

Due: 11/12/07 (beginning of class)

1. Describe in some detail (at least 3 sentences) the following:

(a) Control registers

(b) Memory-mapped I/O

(c) DMA

(d) Device-independent OS software

(e) RAID

2. State three advantages of placing functionality in a device controller, rather than in the kernel. State three disadvantages.

3. Why is it important to scale up system bus and device speeds as the CPU speed increases?

4. How does DMA increase system concurrency? How does it complicate the hardware design?

5. For each of the following device access scenarios, say whether DMA would likely provide an appropriate transfer strategy, and briefly explain why: (1) reading user input from the keyboard, (2) sending MPEG video from the file system to the network.



8. Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999. The drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order, is 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130. Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests, for each of the following disk-scheduling algorithms?

(a) First-come, first-served.

(b) Shortest seek time first.

(c) Elevator.

1. Requests are not usually uniformly distributed. For example, some files are accessed more frequently than other files. For instance, files that contain meta-information about the file system structure are accessed much more frequently than a file that contains data. Suppose that you know that 50 percent of the requests are for a small, fixed number of cylinders.

(a) Would any of the scheduling algorithms discussed in class be particularly good for this case? Explain your answer.

(b) Does your solution in (a) suffer from starvation. If so, can it be modified to prevent starvation?

2. Describe how the OS maintains clocks for things such as timesharing, time of day, application alarms, etc. What hardware is required and how is this hardware engineered so that the OS can exploit it?