Eddie Gurnee
COSC 592
Operating Systems

Chapter 1 Reading Questions:

1.14) Under what circumstances would a user be better off using a time-sharing system than a PC or a single-user workstation?

Response: As time-sharing systems rapidly shift between users giving the impression of a dedicated system and is capable of utilizing the processing power of what would otherwise be idle time, a time-sharing system would be most effective in a situation where many users need to run a simple (non resource intensive) process; a system with powerful enough hardware would provide resources to each user and their process as needed with a low level of idle time.

1.20) Direct memory access is used for high-speed I/O devices in order to avoid increasing the CPU's execution load.

a) How does the CPU interface with the device to coordinate the transfer?
b) How does the CPU know when the memory operations are complete?
c) The CPU is allowed to execute other programs while the DMA controller is transferring data. Does this process interfere with the execution of the user programs? If so, describe what forms of interference are caused.

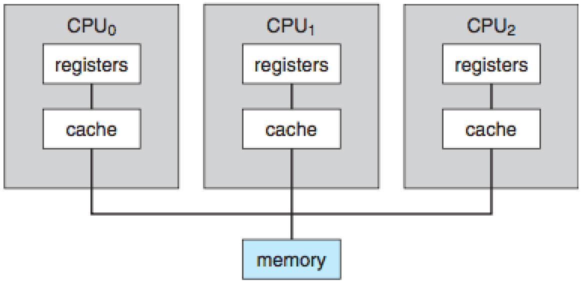
Response:

a) The CPU interfaces with the device by setting up buffers, pointers, and counters for the direct memory access (DMA) transfer, after which the device controller will handle the work of transferring an entire block of data
b) When the DMA process is complete, the device controller signals the CPU with an interrupt
c) The DMA process accesses some block of data from a device and if the DMA disallows other processing of that device data, then other user processes would be unable to access the device

1.21) Some computer systems do not provide a privileged mode of operation in hardware. Is it possible to construct a secure operating system for these computer systems? Give arguments both that it is and that it is not possible.

Response: Without some forcing limits on all software run on the operating system, this security cannot be guaranteed. One way to force software would be through complete virtualization of all running software (as running all user programs through an interpreter – like Java via the JVM), a level of security could be created to the hardware without having a specific privileged mode in hardware.

1.23) Consider an SMP system similar to what is shown in Figure 1.6. Illustrate with an example how data residing in memory could in fact have two different values in each of the local caches.



Response: With a symmetric multiprocessing system as shown by figure 1.6, data is read from memory and stored in the cache of the specific CPU for use. If specific data is read into CPU_0 and CPU_1 and then altered in CPU_1, the resulting changes would not be echoed in CPU_0, creating inconsistent data in the local caches.

1.28) What are some advantages of peer-to-peer systems over client-server systems?

Response: ~~You get far more visits from police officers~~. In a client-server system each individual client connects directly to the server with requests for services or data, putting much of the weight of processing on one side or the other. This setup is rather inflexible, and requires powerful enough hardware on the server side to contend with multiple client connections. In peer-to-peer systems, each node can act as client or server given the needs of the system or other nodes. This allows for great flexibility in the system and is quite fault tolerant; if one node that is currently acting as a server goes down, there are other nodes that can pick up the slack.