Johnny Li

COP4600

HW 1.1

4. To use cache memory, main memory is divided into cache lines, typically 32 or 64 bytes long. An entire cache line is cached at once. What is the advantage of caching an entire line instead of a single byte or word at a time?

The advantage of caching an entire line instead of a single byte or word at a time is that it saves CPU cycles by only retrieving the data once (as an entire line) rather than multiple times (when it is forced to take separate bytes at a time). This advantage is made possible because the data in a cache line is stored sequentially thus when a CPU accesses data in a cache line it takes its desired byte while also taking the nearby data element, which also may be needed instead of searching in various areas memory as would taking single byte at a time.

5. On early computers, every byte of data read or written was handled by the CPU (i.e. there was no DMA). What implications does this have for multiprogramming?

When multiprogramming, multiple programs are executed on the same machine by sharing the processor. By swapping CPU usage between programs whenever one program needs to use the I/O devices, CPU utilization can be maximized. Without DMA, the process of transferring data from main memory to other devices (e.g. I/O) without the usage of a CPU, the multiprogramming system will act as a single programming system. After the initial CPU usage, the system cannot swap to a different program because it now must handle the transferring of every byte of data read or written to and from the I/O devices, only one process will be executed at a time, multiprogramming will be lost in the system.

6. Instructions related to accessing I/O devices are typically privileged instructions, that is, they can be executed in kernel mode but not in user mode. Give a reason why these instructions are privileged.

The instructions related to accessing I/O devices are typically privileged instructions to protect the I/O devices instruction against users modification, to avoid breaking or mishandling hardware or function that the instruction involves.

12. Which of the following instructions should be allowed only in kernel mode?

(a) Disable all interrupts.

~~(b) Read the time-of-day clock.~~

(c) Set the time-of-day clock.

(d) Change the memory map.

All 3 answers on top should be allowed only in kernel mode as only the OS must handle interrupt functions, internal time, and memory management.

14. A computer has a pipeline with four stages. Each stage takes the same time to do its work, namely, 1 nsec. How many instructions per second can this machine execute?

(1 instructions/1 nsec) \* (10^9 nsec/1 sec) = 1\*10^9 instruction/1 sec

There is a delay on the first instruction during the first 3 stages of the process which takes 3 nsec to complete this:

1\*10^9 - 3 = 999,999,997 instruction/1 sec

For #14, is the answer not 10^9-3=999,999,997.

Because there is a delay on the first instruction during the first 3 stages of the process which takes 3 nsec to complete this.

A  B  C  D

1

2  1

3  2  1

4  3  2  1

For #12, I believe that (B) should be allowed in user mode and (C) in kernel mode.

Users are allowed to read the "Read the time-of-day clock" in the UI as on screen without the need to access the kernel while "Set the time-of-day clock" is used in the kernel mode as some devices in the kernel may share the clock.

For #12, I believe that (C) in kernel mode.

"Set the time-of-day clock" is used in the kernel mode as some devices in the kernel may share the clock.

For #5, with everything being handled by the CPU,  after the initial CPU usage, the system cannot swap to a different program because it now must handle the transferring of every byte of data read or written to and from the I/O devices, only one process will be executed at a time, multiprogramming will be lost in the system.

Agree