Assignment 1

CS3510 - Operating Systems

Soham Rajesh Pawar
CS22BTECH11055

November 13, 2023

Question 1 :

(a) **What is the purpose of interrupts ?**

Interrupts are signals that halt execution of the current program and load the instructions at a specific location that pertain to the interrupt service routine. Once the interrupt service routine resolves the interrupt, the CPU resumes execution of the interrupted program.

Their purpose includes:

i) Used by device controllers to inform the CPU that it has finished its operation.

ii) Facilitate multitasking, allowing the CPU to execute another program while waiting for an interrupt.

iii) Allow the system to promptly react to sensitive events (e.g., division by 0).

(b) **How does an interrupt differ from a trap ?**

Both are mechanisms to alter the normal flow of program execution and handle specific events. However, they differ in origin and purpose.

i) **Origin:**

- **Interrupts:** External events, often generated by hardware, trigger interrupts. Asynchronous events occur independently of the normal program flow.
- **Traps:** Synchronous events tied to the execution of specific instructions in the program.

ii) **Purpose:**

- **Interrupts:** Handle events external to the CPU, such as I/O operations or hardware-generated signals.

- **Traps:** Handle exceptional conditions within the program, such as division by 0 or executing privileged instructions.

(c) **Can traps be generated intentionally by a user program ?**

(d) **If so, for what purpose ?**

Yes, traps can be generated intentionally by users for various purposes, such as:

i) System calls: Requesting services from the operating system, e.g., file I/O or memory allocation.

ii) Error handling: Intentionally generating traps to handle specific error conditions.

iii) Debugging: Pausing the program at specific locations during development to check its robustness.

Question 2 :

(a) **How does the CPU interface with the device to coordinate the transfer ?**

Interaction between the CPU and a device involves operations managed by the device controller or driver, acting as an intermediary.

(b) **How does the CPU know when the memory operations are complete ?**

The CPU is informed of memory operation completion by the device through interrupts.

(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.**

DMA operates independently of the CPU, allowing concurrent execution. Potential interference forms include:

i) Bandwidth competition: Contention for the memory bus leading to delays.

ii) Cache interference: Issues with cache coherency if DMA writes directly to memory.

iii) Synchronization and ordering: DMA transfers affecting the order of data writes.