

30 Lecture - CS501

Important Subjective

1. **What is interrupt priority, and why is it important?**

Answer: Interrupt priority is the order in which interrupts are serviced by the processor. It is essential to ensure that higher-priority interrupts are serviced first, as they may represent critical events that require immediate attention.

What is the difference between a hardware interrupt and a software interrupt?

Answer: A hardware interrupt is triggered by an external device or event, while a software interrupt is generated by a program running on the processor.

What is the purpose of interrupt masking?

Answer: Interrupt masking is a technique used to temporarily disable interrupts to prevent them from being serviced while critical operations are being performed.

What is interrupt chaining, and how is it used?

Answer: Interrupt chaining is a technique used to handle nested interrupts. When a nested interrupt occurs, the processor saves the current interrupt context and chains the new interrupt to the current one. When the new interrupt is serviced, the processor returns to the previous interrupt context and resumes the interrupted task.

What is a priority inversion, and how can it be avoided?

Answer: A priority inversion occurs when a low-priority task holds a resource needed by a higher-priority task, causing the higher-priority task to be blocked. To avoid priority inversion, resources should be allocated and released in a way that ensures that higher-priority tasks have priority access to them.

What is an interrupt vector table?

Answer: An interrupt vector table is a table that stores the addresses of the interrupt service routines for each interrupt type. When an interrupt occurs, the processor looks up the address of the appropriate service routine in the vector table.

What is the difference between a level-triggered interrupt and an edge-triggered interrupt?

Answer: A level-triggered interrupt is active as long as the triggering condition is present, while an edge-triggered interrupt is triggered by a specific transition of the triggering signal.

How can interrupt priorities be assigned in a system with multiple processors?

Answer: Interrupt priorities can be assigned on a per-processor basis or using a global interrupt controller that coordinates interrupt handling across multiple processors.

What is interrupt latency, and why is it important?

Answer: Interrupt latency is the time delay between the occurrence of an interrupt and the start of its servicing. It is important to minimize interrupt latency to ensure timely response to critical events.

What is the role of interrupt service routines, and how are they implemented?

Answer: Interrupt service routines are functions that handle interrupts by performing the necessary tasks to respond to the event. They are implemented in a low-level language and typically have restricted access to system resources to ensure their safety and efficiency.