Mid Assessment - Spring - 2023

Course : CSE-333 | Computer Peripherals and Interfacing

Dept : CSE

ID : 2216080021

Name : Md Rakib Hasan

Ans. to the Q. No - 01

(a) Computer Interfacing and Peripheral :-

Computer Peripheral Interfacing is the interactions between computers processors and computers peripherals Any input or output process that occurs through a peripheral device is known as computer interfacing.

An interface is the concept of communication between to components, objects, elements or two part of a single object or component and it may be occur on both hardware and software level.

Computers peripherals are the components or devices which are connected to the processors and extend the work of the processor.

Example of Computer Peripheral Interfacing :-

→ Taking input from a keyboard is a peripheral interfacing. Here keyboad is a peripheral device.

→ Printing a document with a printer is a peripheral interfacing where printer is the peripheral device.

→ Showing a video on the monitors is peripheral interfacing and this case monitor is the peripheral device.

(b) Interrupt :-

In digital computers, an interrupt is a request for the processor to interrupt currently executing code, so that the event can be processed in a timely manner.

Example :- One example of this is moving a mouse or pressing a keyboard key. In these examples of interrupts, the processor must stop to read

the mouse position or keystroke at the instant. In this type of interrupt, all devices are connected to the Interrupt Request Line (IRL).

Ans. to the Q. No – 02

(a) The list of Sources of Interrupts :-

There are many sources for interrupts varying from simply asserting an external pin to error conditions within the processor that require immediate attention.

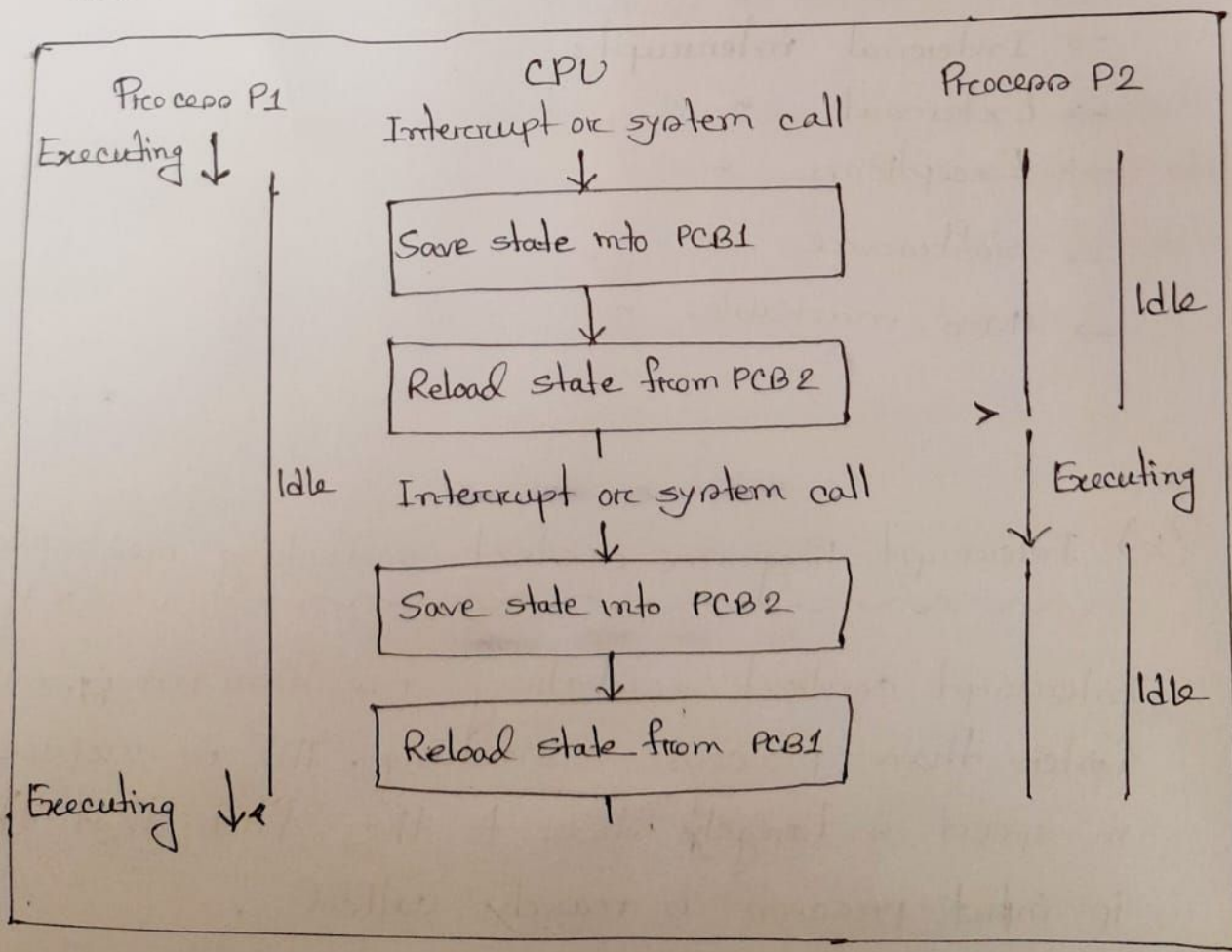
→ Internal interrupts
→ External "
→ Exceptions "
→ Software "
→ Non-maskable "

(b) Interrupt Program context switching mechanism :-

Interrupt context switching mechanism is much faster than process switching. The increase in speed is largely due to the fact that the ip-input process is rarely called.

Interrupt context switching instead interrupts the process currently running on the routers to switch the packet. Interrupt context switching usually bypasses the RIB, and works with parallel tables, which are build more efficiently.

There are several steps are involves in context switching of the processes. The following diagram represents the context switching of two processes, P1 to P2, when are interrupt, I/O needs, or priority-based process occurs in the ready queue of PCB.



The following steps are taken when switching Process P1 to Process P2 :

1. First then context switching needs to save the state of process P1 in the form of the program counter and the registers to the PCB, which is in the running state.
2. Now update PCB 1 to process P1 and moves the process to the appropriate queue, such as the ready queue, I/O queue and waiting queue.
3. After that, another process gets into the running state, or we can select a new process from the ready state, which is to be executed or the process has a high priority to execute its task.
4. Now, we have to update the PCB for the selected process P2. It includes switching the process state from ready to running state or from another state like blocked, exit, or suspend.
5. If the CPU already executes process P2, we need to get the status of process P2 to resume its execution at the same time point where the system interrupt occurs.

Ans. to the Q. No – 03

(a) Analog Interfacing :-

An analog interface is an electrical connection that forwards analog electric signals to downstream electric and electronic devices or components for further processing

Standardised analog interfaces (e.g. 0/4 - 20 mA and 0/2 – 10 V) are also referred to as standard signals.

Since current signals are not sensitive to electromagnetic interference and loss of voltage, they are preffered over voltage signals. The maximum length of a signal cable for the power source is only limited by the maximum load.

(b) Interrupt vector table and its works :-

The interrupt vector table, often abbreviated to IVT or simply IV, is an array of pointers to functions, associated by the CPU to handle specific exceptions, such as faults, system service requests from the application, and interrupt requests from peripherals.

An interrupt request from a hardware component

or peripheral will force the CPU to abruptly suspend the execution, and execute the function at the accociated position in the vectore. For this reason, these functions are called interrupt service routines.

Interrupt Service Routine and its workes :-

An Interrupt Service Routine (ISR) is a software routine that hardware invokes in responce to an interrupt. ISR examines an interrupt and determines how to handle it executes the handling, and the returns a logical interrupt value.

An ISR might move data from a CPU register or a hardware port into a memory buffer, in general it relies on a dedicated interrupt thread, called the interrupt service thread, to do most of the required processing. If additional processing is requeired, the ISR returns a logical interrupt value to the Kernel. It then maps a physical interrupt number to a logical interrupt value.