

Ans. to Qno. 1

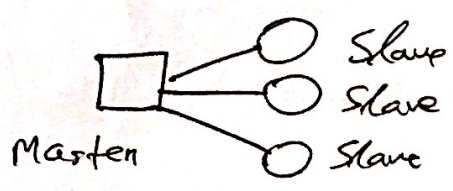
Ans. to Qno. 1

Types of multicore system

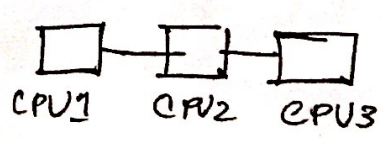
a) Asymmetric System

b) Symmetric system

a) Asymmetric system: There is a master-slave relationship where one processor controls the other slave processors.



b) Symmetric system: All the processors are treated equally without any control or priority.



Advantages of multicore system:

a) Increased throughput

A multiprocessor system ~~is~~ ~~can perform~~ has more computational power than single ~~the syst~~ processor system.

b) Economy of Scale

Cheaper to buy a ^{single} system with two processors than a single ~~of~~ two systems with single processor with equal computation power.

c) Higher reliability

A multiprocessor system can have its functions ~~to~~ ^{be} executed in different processors. ~~Thus,~~ ^{Thus,} failure of a single ~~device~~ processor won't create problems.

Ans to Q.no. 2

To restrict the users from performing/executes tasks harmful to the system, we need dual mode of operation. Thus it ~~means~~ provides the OS a way to protect the system.

The OS has mode bits to differentiate dual mode of operation

a) User has mode bit = 1

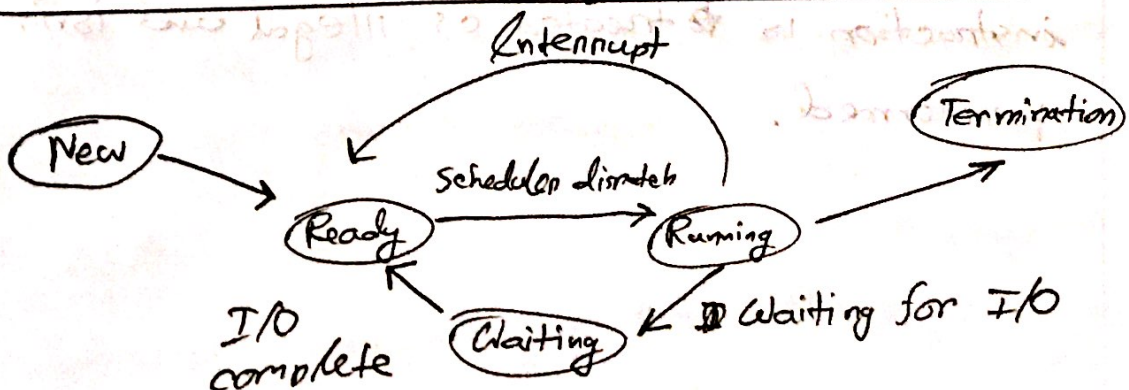
b) Kernel has mode bit = 0.

When a user detects an operation has mode bit 1, it knows user is ~~using~~ doing the operation. For ~~mode~~ taking control, it changes the mode bit to 0.

Ans. to Q.no. 4

Process → A process is a program in execution. It ~~is~~ is located in the main memory.

Program → A set of instructions used to perform certain tasks. Located in secondary memory. When executed it becomes a process.



New: Process is created and ~~put in~~ is to be put in the queue. Created from program.

Ready: Ready Resource is allocated and ready to be executed.

Running: Process is executed.

Waiting: Waiting for I/O operation.

Termination: End of process execution.

Ans. Q. no. 3

Some instructions can cause harm to system or change major properties. These instructions can only be performed in Kernel mode (0 mode bit). It is called privileged instruction.

When a user tries to perform privileged instruction, the OS changes the mode bit ~~to~~ from 1 to 0 and thus takes control from user. The instruction is ~~to~~ treated as illegal and isn't performed.



Ans. to Q. no. 6

The communication processes can be synchronized by blocking/nonblocking. ~~are~~ The ways are -

- i) Blocking Send - Here, the sending process will be blocked but receiving process is received normally by direct link or mail box.
- ii) Nonblocking Send - The operation is resumed by the sending process.
- iii) Block Receive: Same like blocking send. Instead of sender, the receiver will block message.
- iv) Nonblocking Receive: The ~~receiver~~ receiver resumes operation by receiving the message or null message.

Ans. to Q. no. 5

The ways to implement queue for process communication is:

- i) Zero Capacity: Basically, same as having no queue. The link can not have waiting messages and sender will block until receiver receives.
- ii) Bounded capacity: Queue with finite length which can contain n messages at best. Message is sent to

18.0041120

queue if it is not full, otherwise it is discarded or blocked. By storing message, sender doesn't have to wait for receiver to receive message and thus saves time.

iii) Unbounded Capacity: Queue with infinite length and can store any number of messages.

The sender can continue sending as much as it wants without getting blocked.

~~iii) Block Reserve: Queue like blocking send & receive.~~