CSMC 412

Operating Systems
Prof. Ashok K Agrawala

Set ID

Operating System as Decision Maker

- All resource management decisions are taken by the OS
- What information does it have to base those decisions on?
  - It has to collect and keep that information
  - Make sure that the information is not corrupted
  - Update as necessary
  - Use it
- Where to keep information about entities under its control?
  - Control Blocks

Information Based Decision Making

- A decision requires information
- The information available to the decision maker
  - Designed as a part of the system design
  - In the address space of the executing unit taking the decision- OS
- Have to recognize independent action units
  - A unit that continues to operate once triggered
    - CPU
    - Clock
    - Disk
    - Disk controller
    - …
- Every Action has to be triggered from external source at some point.

Using Information in Decisions

- Access information
- Decide
- Initiate action
- Modify information

Can information Change during this period?

- Shared memory vs messages

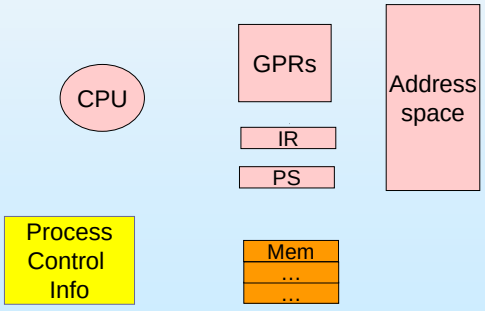
Concurrent Executions

- When there are concurrent executions the actions of one process can be affected by the action of another process at any stage of execution –
  - Unless appropriate protection measures are taken
- One way of protection
  - Isolate independent processes
    - But they do share resources – would that cause conflicts??
- Cooperating processes
  - Have to communicate/share
  - Thus they interact

ID.5

Example

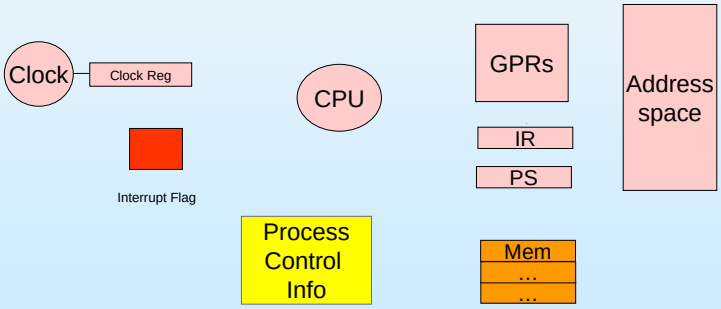
- A program in execution



ID.6

Time Quantum

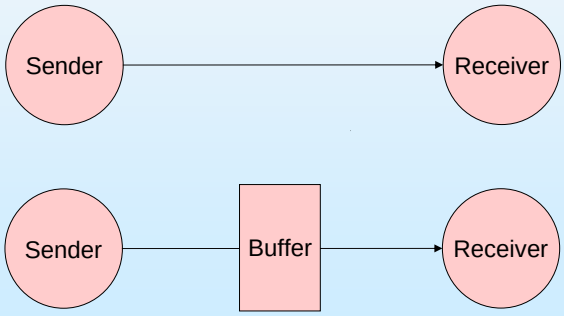
- Time quantum for an executing process



ID.7

Communication

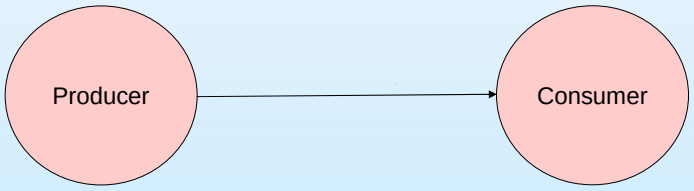
- Receiver must be ready to receive
  - Prior Arrangement
  - Coordinate in time
  - Use a Buffer –Solves immediate problem –but !!



ID.8

Producer Consumer

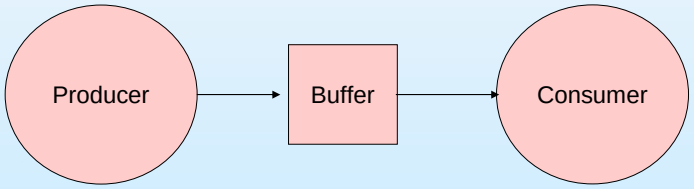
- One process generates data – the producer
- The other process uses it – the consumer
- If directly connected – time coordination
  - How would they coordinate the time ??



ID.9

Producer Consumer

- One process generates data – the producer
- The other process uses it – the consumer
- If not directly connected – have a buffer
  - Buffer must be accessible to both
  - Finite Capacity N – Number in use - K



ID.10

Coordination

- Number full – K
  - Incremented by Producer
  - Decremented by Consumer

Read K

Increment

Store K

Read K

Decrement

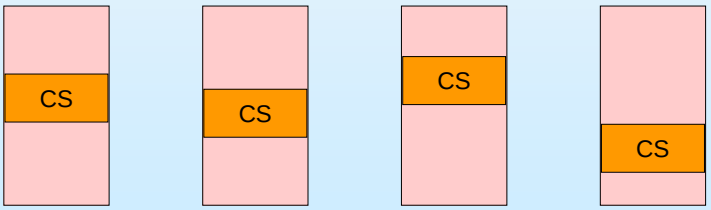
Store K

Information Needed by Producer/Consumer

- Producer
  - There is an empty buffer
  - Empty buffer ID
  - Nobody else is using this buffer for filling or emptying
  - Inform others that it is using this buffer.
- Consumer
  - There is a full buffer
  - Full buffer ID
  - Nobody else is using this buffer for filling or emptying
  - Inform others that it is using this buffer.

Mutual Exclusion

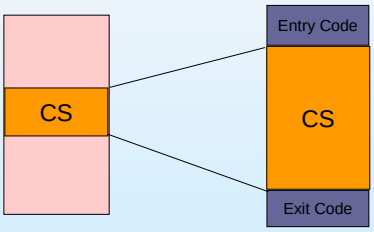
- N processes
- Each has a portion of the code called Critical Section
- At any instant no more than one process can be in its critical section
- What should a process do ???



ID.13

Critical Section

- Entry and Exit Code



- Entry Code
  - Code to ascertain that this process can enter the CS
  - Make sure that other processes know that this process has entered CS
- Exit Code
  - Let other processes know that it has exited from its CS
- HOW ???

ID.14

Atomic Action

- An action that is either completely done or not done at all
  - Can not be accessed or affected in the middle of its execution
- Necessary for
  - Access the information
  - Take decision
  - Modify the information

ID.15

Synchronization

- Controlling the execution of processes to conform to stated/required timing/precedence relationships among events
  - Precedence
    - A must occur before B
  - Mutual Exclusion
  - Producer Consumer
  - More complex relationships
- Recognizing the information needs for any such decisions does make the design easier.

ID.16