Week7: Intro to Operating Systems

NOS_Lecture_slides_WK07.pdf

Operating System Uses:

- Exploits the hardware resources fo one or more processors.
- Provides a set of services to system users.
- Manages secondary memory and I/O devices.

Basic Elements:

Processor, Main Memory, I/O Modules, System Bus

Processor:

- controls the operation of the computer
- performs the data processing function
- known as the Central Processing Unit (CPU)

Main Memory:

- Stores data and programs
- Typically, volatile → contents of the memory is lost when the computer is shut down.
- Referred to as real memory or primary memory.

Input/Output Modules:

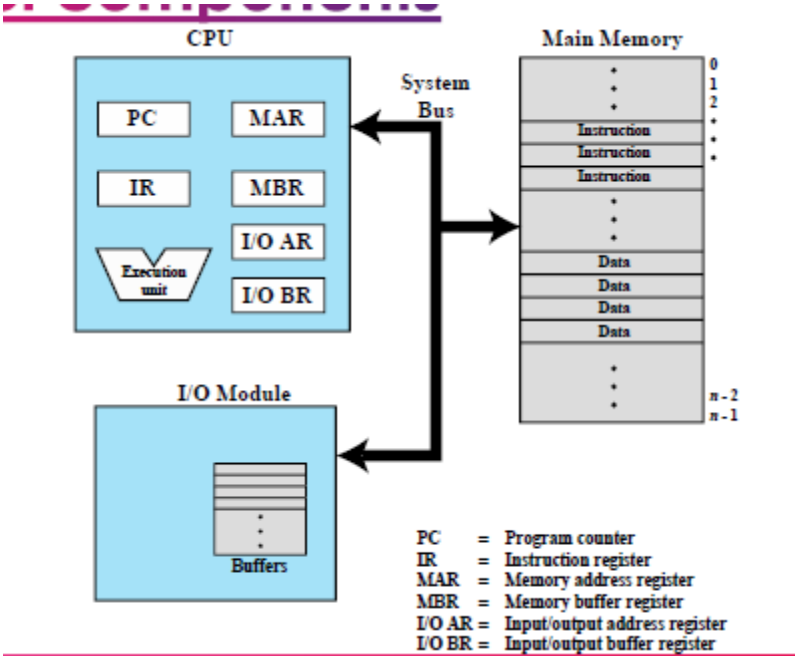
- Move data between the computer and it’s external enviornment.

- Secodary memory devices, eg, disks.
- communications equipments
- Terminals

System Bus:

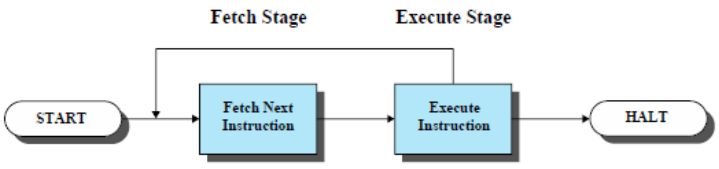
- Provides communication among processors, main memory, and I/O modules.

Top-Level Components



Instruction execution → A program consists of a set of instructions stored in memory.

Processor reads instruction → Processor executes each instruction.



Interrupts

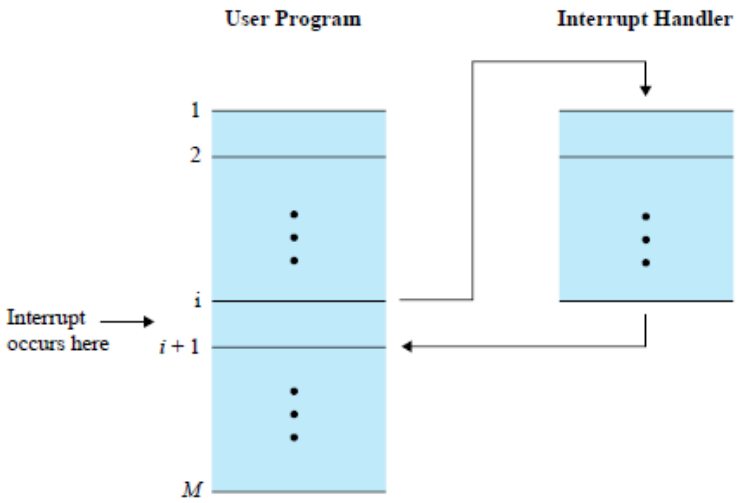
Mechanism which allows other modules to interrupt the normal sequencing of the processor

Provided to improve processing utilisation:

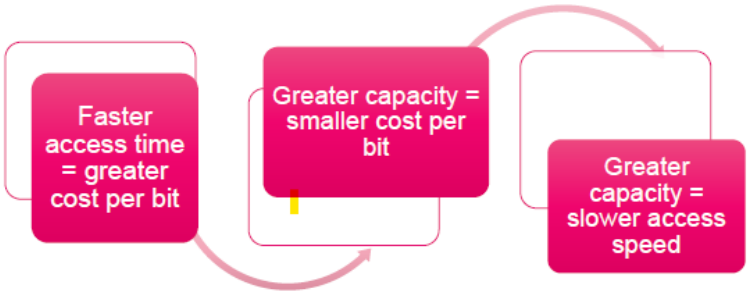
- Most I/O devices are slower then the processor
- Processor must pause to wait for device
- Wasteful use of the processor

Classes of Interrupts:

- Program
- Timer
- I/O
- Hardware failure



Memory Relationships



Memory Hierarchy going down

- Decreasing cost per bit
- increasing capacity
- Increasing access time
- Decreasing frequency of access to the memory by the processor.

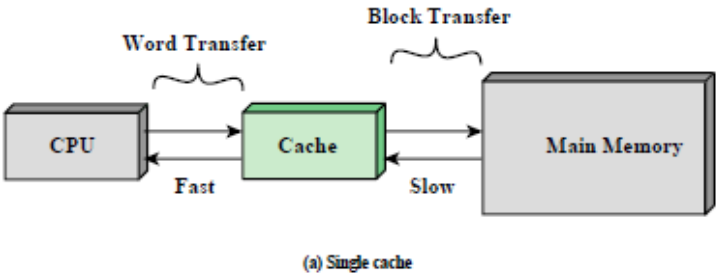
Principle of Locality

- Memory references by processor tend to cluster
- Data is organized where the percentage of access to each successively lower level is substantially less thanb that of the level above
- Can be applied across more than two levels of memory.

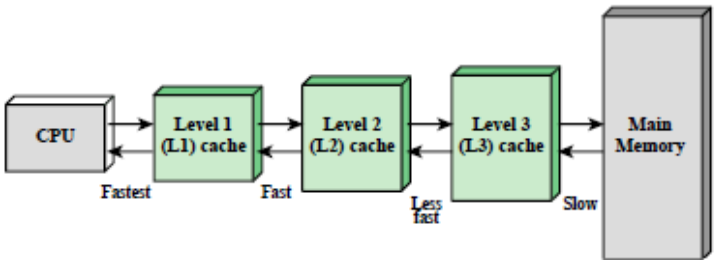
Secondary Memory → Auxiliary memory

Cache Memory

- invisible to the OS
- Interacts with other memory mangeemnt hardware
- Processor must access memory at least once per instruction cycle.
- Processor execution is limited by memory cycle time
- Exploit the principle of locality with a small, fast memory.



(a) Single cache



I/O Communication Techniques

Programmed I/O

- Processor issues I/O command
- Processor monitors I/O device until operation completes
- Involves busy-waiting, wasting processor time

Interrupt-Driven I/O

- Processor issues I/O command
- Continues executing other instructions
- I/O module interrupts processor when done
- More efficient use of processor time

Direct Memory Access (DMA)

- DMA controller transfers blocks of data directly between memory and I/O device
- Processor only involved at beginning and end of transfer
- Much more efficient for large data transfers

Symmetric Multiprocessing (SMP)

A multiprocessing architecture where multiple processors share a common memory and are controlled by a single OS instance.

Key Characteristics of SMP:

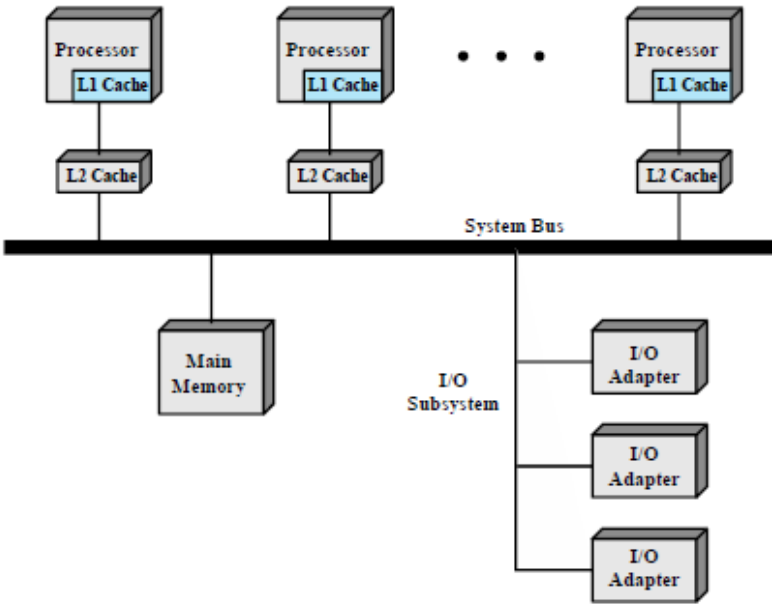
- Multiple processors of the same type operate within a single system
- Processors share access to common memory, I/O devices, and buses
- All processors can perform the same functions (symmetric)
- The system is controlled by an integrated operating system

Advantages of SMP:

- Improved performance through parallel processing
- High reliability - if one processor fails, others can continue
- Scalability - can add processors to increase computing power
- Simplified resource management compared to distributed systems

Challenges:

- Complex scheduling and synchronization requirements
- Memory contention among processors
- Operating system must be designed to support multiprocessing
- Diminishing returns as processor count increases



Operating System Objectives

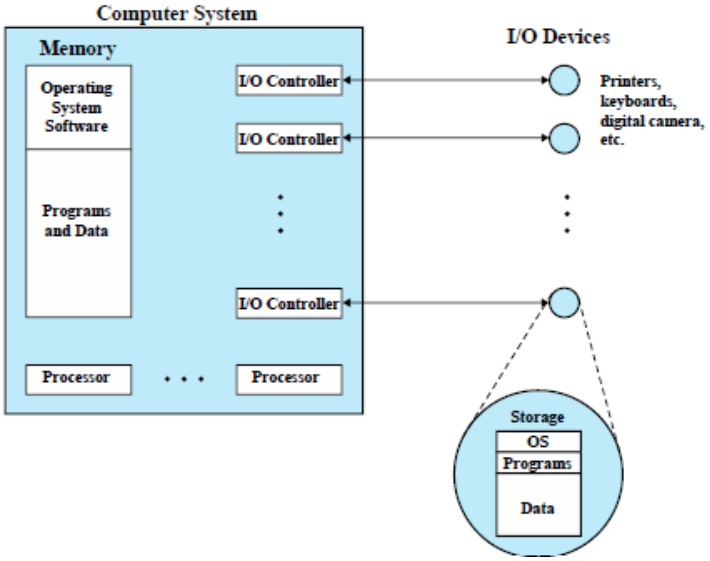
- Convenience: Makes computer more convenient to use
- Efficiency: Allows computer system resources to be used efficiently
- Ability to evolve: Permits effective development, testing, and introduction of new system functions

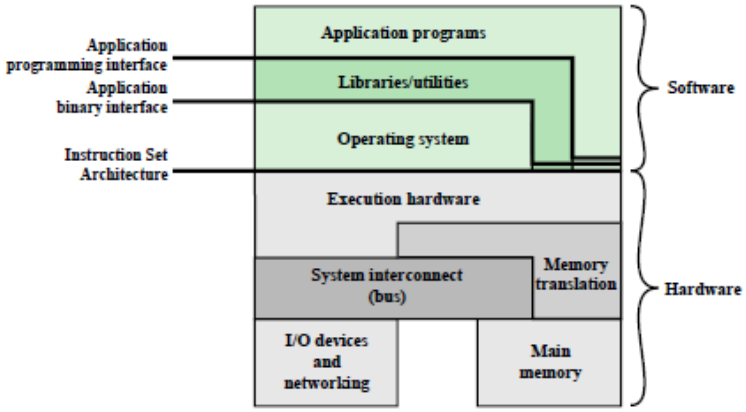
OS as Resource Manager

The Operating System manages various resources:

- Processor time
- Memory space
- File storage
- I/O devices

OS must balance efficiency and fairness in resource allocation.





Evolution of Operating Systems

Serial Processing (1940s-1950s)

- No operating system
- Manual setup and scheduling
- Inefficient processor utilization

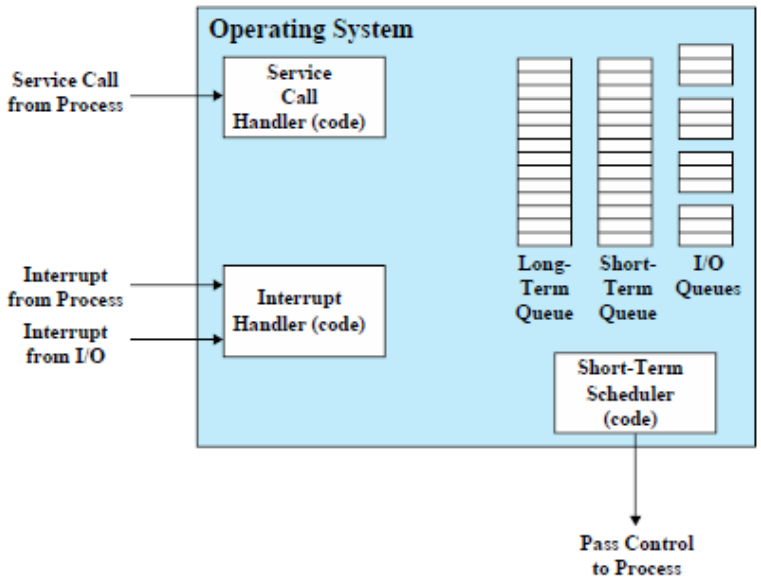
Simple Batch Systems

- Monitor software to automate job sequencing

- Improved throughput but still inefficient

Multiprogrammed Batch Systems

- Multiple programs kept in memory simultaneously
- When one program waits for I/O, processor can work on another
- Requires memory management and protection

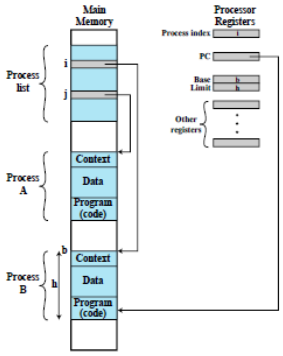


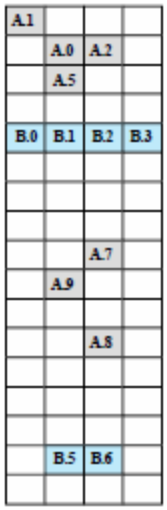
Time-Sharing Systems

- Multiple users interact directly with the system
- Processor time divided into quanta (time slices)
- Creates illusion of dedicated computer for each user

Process Management

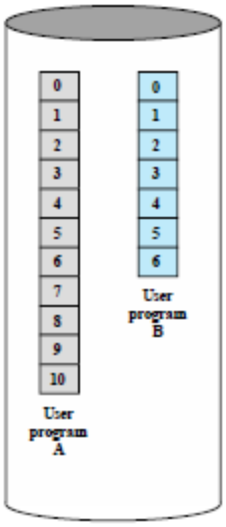
- The entire state of the process at any instant is contained in its context
- New features can be designed and incorporated into the OS by expanding the context to include any new information needed to support the feature





Main Memory

Main memory consists of a number of fixed-length frames, each equal to the size of a page. For a program to execute, some or all of its pages must be in main memory.



Disk

Secondary memory (disk) can hold many fixed-length pages. A user program consists of some number of pages. Pages for all programs plus the operating system are on disk, as are files.