CS341: Operating System

OS Design from User Prospects & Hardware

Lect03 : 5th Aug 2014

Dr. A. Sahu
Dept of Comp. Sc. & Engg.
Indian Institute of Technology Guwahati

1

Outline

- Review of System Arch. from OS prospects
- More review stuffs
  - Multi-core, cluster architecture
  - ***Multi programming and Multi Tasking and Kernel mode***

2

Review of System Architecture from OS prospects

- Computer-system have
  - **Mother Board**, CPU, Mem, KBD, Mouse, display
- Every Mother board have
  - DMA, Interrupt, USB and PCI controller, Timer
  - **Programmable, smarter and run in parallel**
- OS is interrupt driven
  - **Answer to Question Asked in last class**
  - Trap/ exception- Software-generated interrupt caused either by an error or a user request, handle by CPU without asking to Interrupt controller
- Memory Hierarchy work in caching principle and DMA helps in memory transfers

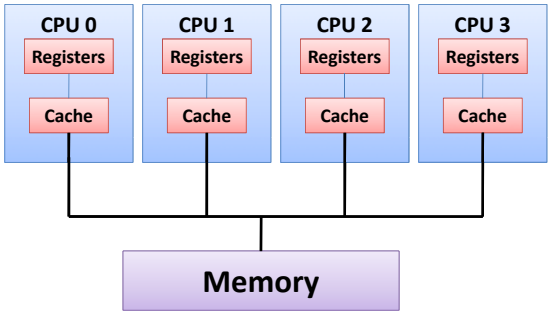
I/O Structure

- Non-Blocking I/O methods
- After I/O starts, control returns to user program without waiting for I/O completion
  - **System call** – request to the OS to allow user to wait for I/O completion
  - **Device-status table** contains entry for each I/O device indicating its type, address, and state
  - OS indexes into I/O device table to determine device status and to modify table entry to include interrupt

Computer-System Architecture

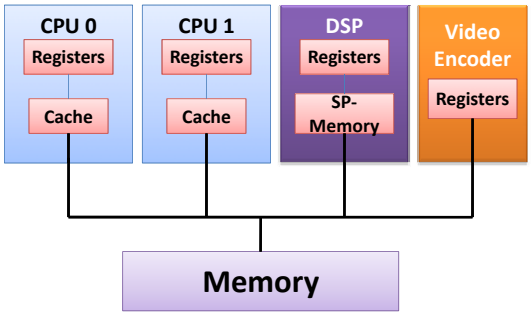
- Most systems use a general-purpose processor
  - Most systems have special-purpose processors as well
- **Multiprocessors** growing in use and importance
- Also known as **parallel systems**, **tightly-coupled systems** Advantages include:
  - **Increased throughput, Economy of scale**
  - **Increased reliability** – graceful degradation or fault tolerance
- **Asymmetric Multiprocessing** – each processor is assigned a specific task.
- **Symmetric Multiprocessing** – each processor performs all tasks

Symmetric Multiprocessing Architecture



Example: Core i5, Core2Quad, AMD Athlon, Optron

Asymmetric Architecture



Multithreaded Architecture

- Core/Processor
- Each processor can be threaded
  - How many thread can be handled: 2,4,8
  - Hardware threading (Ex-Intel Hyper Threading)
- Example
  - Intel Core i7 : 4 cores, 8 Threads
  - Intel PIV HT : 1 core, 2 Thread
  - Intel Xeon E5-2687: 8 cores, 16 threads
  - Intel Atom C2780 : 8 cores, 8 threads

8

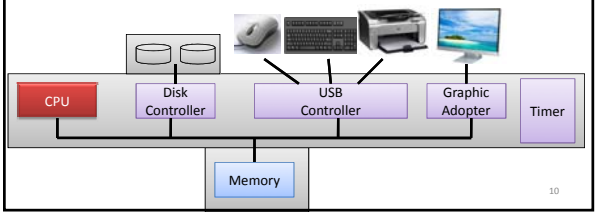
Number of Cores & Thread

- Xeon Phi : 60 cores
- NetLogic MicroSystems XLP: 32 core, 128 threads
- Tilera TILE64: 64-core
- ClearSpeed CSX700: 192 cores
- **GPUs Nvidia & AMD : Thousands of Tiny Cores**

9

Processor Vs Micro-controller Vs SOC

- Processor : In a Single Chip
- Micro-controller: In a single chip
  - Processors + (USB, Disk,..) Controllers +Timers+GA
- SOC : System on Chip, **All mobile Platforms**
  - Microcontroller + Memory + *Wireless Controller (Opt)*

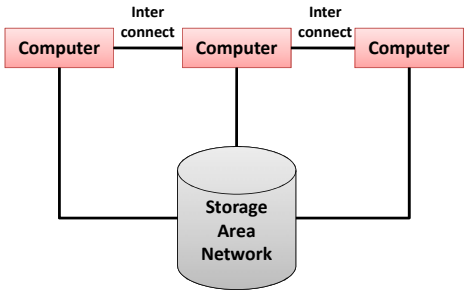


10

Clustered Systems

- Multiple systems working together
- Usually sharing storage via a **storage-area network(SAN)**
- Provides a **high-availability** service which survives failures
  - **Asymmetric clustering** has one machine in hot-standby mod
  - **Symmetric clustering** has multiple nodes running applications, monitoring each other
- Some clusters are for **HPC**
  - Applications must be written to use **parallelization**
- Some have **distributed lock manager** (**DLM**) to avoid conflicting operations

Clustered Systems



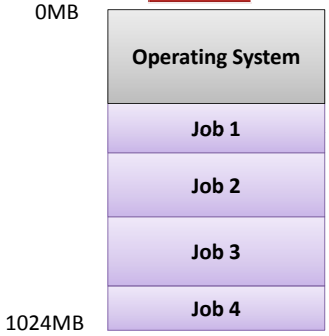
OS Structures Basics
Multiprogramming & Multi Tasking

13

OS Structure: Multiprogramming

- **Multiprogramming/Batch system**
  - needed for efficiency
- Single user cannot keep CPU and I/O devices busy at all times
- Multiprogramming organizes jobs (code and data), so CPU always has one to execute
- A subset of total jobs in system is kept in memory
- One job selected and run via **job scheduling**
- When it has to wait (for I/O for example), OS switches to another job

Memory Layout for Multi-programmed System



OS Structure: Time Sharing

- Timesharing (multitasking)
  - Is logical extension
- CPU switches jobs so frequently
  - That users can interact with each job while it is running
  - Creates an **interactive** computing

Time Sharing (Multitasking)

- **Response time** should be < 1 second
- Each user has at least one program executing in memory ⇨**process**
- If several jobs ready to run at the same time ⇨ **CPU scheduling**
- If processes don't fit in memory, **swapping** moves them in and out to run
- **Virtual memory** allows execution of processes not completely in memory

OS Operation Mode
Interrupt, Kernel Mode & User Mode

18

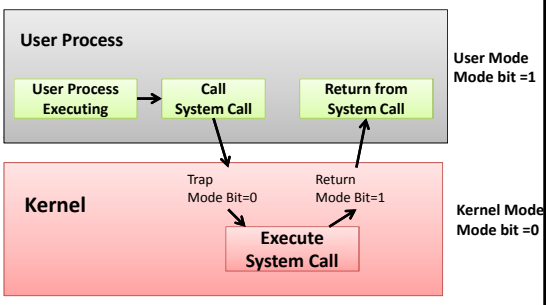
Operating-System Operations

- **Interrupt driven** (H/W and S/W)
  - H/W interrupt by one of the devices
  - S/W interrupt (**exception** or **trap**):
    - Software error (e.g., division by zero)
    - Request for OS service
    - Other process problems include
      - infinite loop
      - processes modifying each other or the operating system

Operating-System Operations (cont.)

- **Dual-mode** operation allows OS to protect itself and other system components
  - **User mode** and **kernel mode**
  - **Mode bit** provided by hardware
    - Provides ability to distinguish when system is running user code or kernel code
    - Some instructions designated as **privileged**, only executable in kernel mode
    - System call changes mode to kernel, return from call resets it to user
- Increasingly CPUs support multi-mode operations
  - i.e. **virtual machine manager** (**VMM**) mode for guest **VMs**

Transition from User to Kernel Mode



Transition from User to Kernel Mode

- Timer to prevent infinite loop / process hogging resources
  - Timer is set to interrupt after some time period
  - Keep a counter decremented by the physical clock.
  - OS set the counter (privileged instruction)
  - When counter zero generate an interrupt
  - Set up before scheduling process to regain control or terminate program that exceeds allotted time

Installing Linux on 8085 or 8086

- Is it possible to install Linux OS on top of 8085 or 8086 based system ?
- **No**
- Because it don't support mode bit
  - Kernel Mode or user mode bit
  - i386,i586,i686……in short ix86 support mode bit

23