CS343: Operating System

Review of Computer System Architecture from OS Prospects

Lect03 : 1th Aug 2023

Dr. A. Sahu

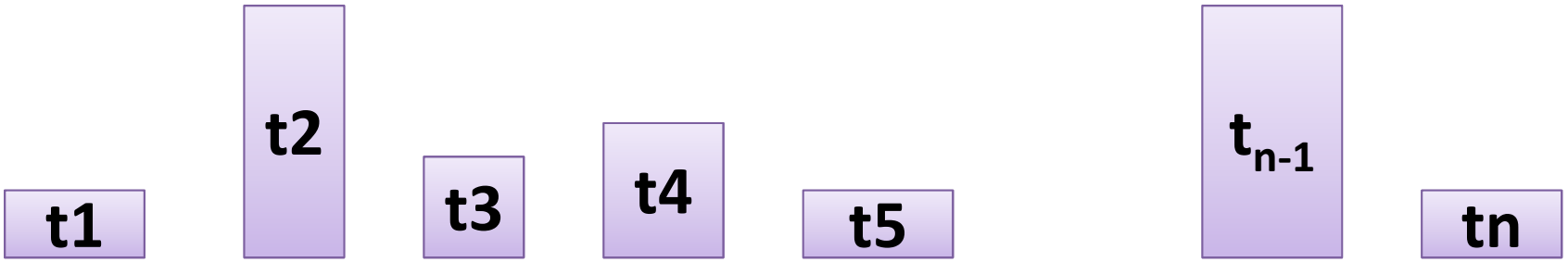
Dept of Comp. Sc. & Engg.

Indian Institute of Technology Guwahati

How the Resource Management is Difficult ?

A simple example with algorithmic analysis…

Scheduling : *N* Tasks on *3* Processors



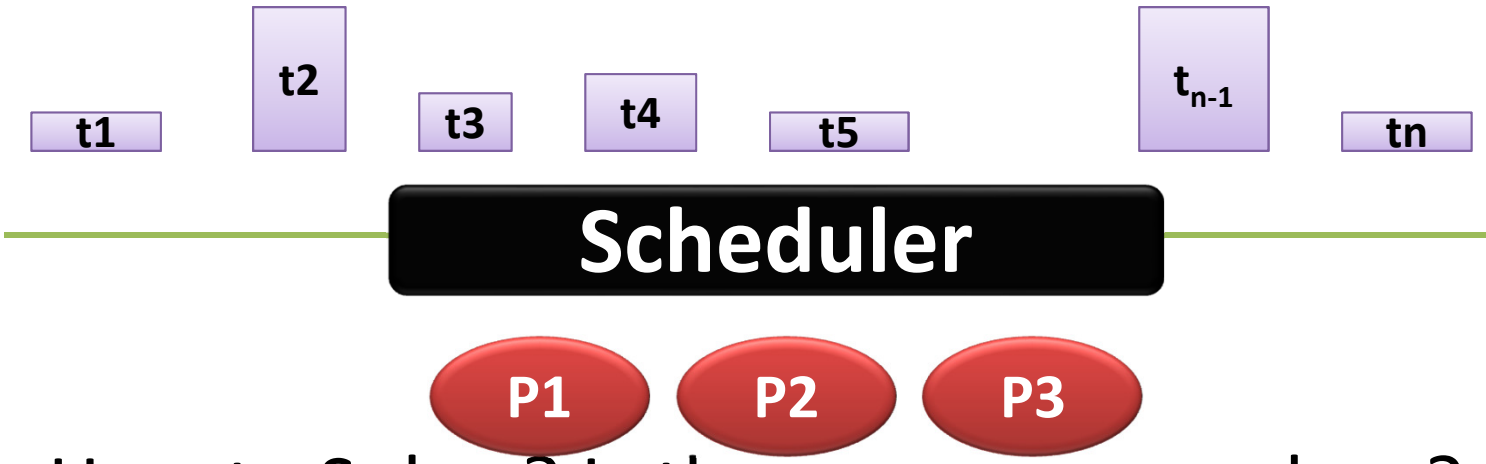
Scheduling : *N* Tasks on *3* Processors

- **Assumption (Very simplistic)**
  - Lets N Task arrives at time 0
  - All tasks are independent task
  - i_{th} task takes t_i amount of time to execute on any processor
  - Homogeneous: All processor are of same type and speed
  - Interruption not allowed : Task assigned to a processor will execute completely without interruption

Scheduling : *N* Tasks on *3* Processors

- **Optimization Criterion**
  - Minimize finishing time of last executed task: Minimize C_{max}

Scheduling : *N* Tasks on *3* Processors



- How to Solve ? Is there any approaches ?
  - Schedule Longest Task first === > FAIL
  - Schedule shortest Task first ====> FAIL
  - Dynamic Programming ==== > FAIL

NP-Complete Problem

Need to solve using Approximation or Heuristics

Review of
Computer System Architecture
from OS Prospects

PC Components OEMs

- Processor
  - Intel : Core i3, i5, i7, i9, Xeon, XeonPhi (KNL) : 11/12/13 Gen
  - AMD: Ryzen 3, 5,7,9: 3,4,5,6 Generation
- Chip Sets : Motherboard
  - Intel, AMD, Asus, Gigabyte, Super micro
- DRAM Memory: Micron, ISSI, Alliance, Infineon, Samsung, Hynix,
- SSD/Disk: Samsung, Crucial, Kingston, Transcend, Intel, Adata, Corsair, Segate

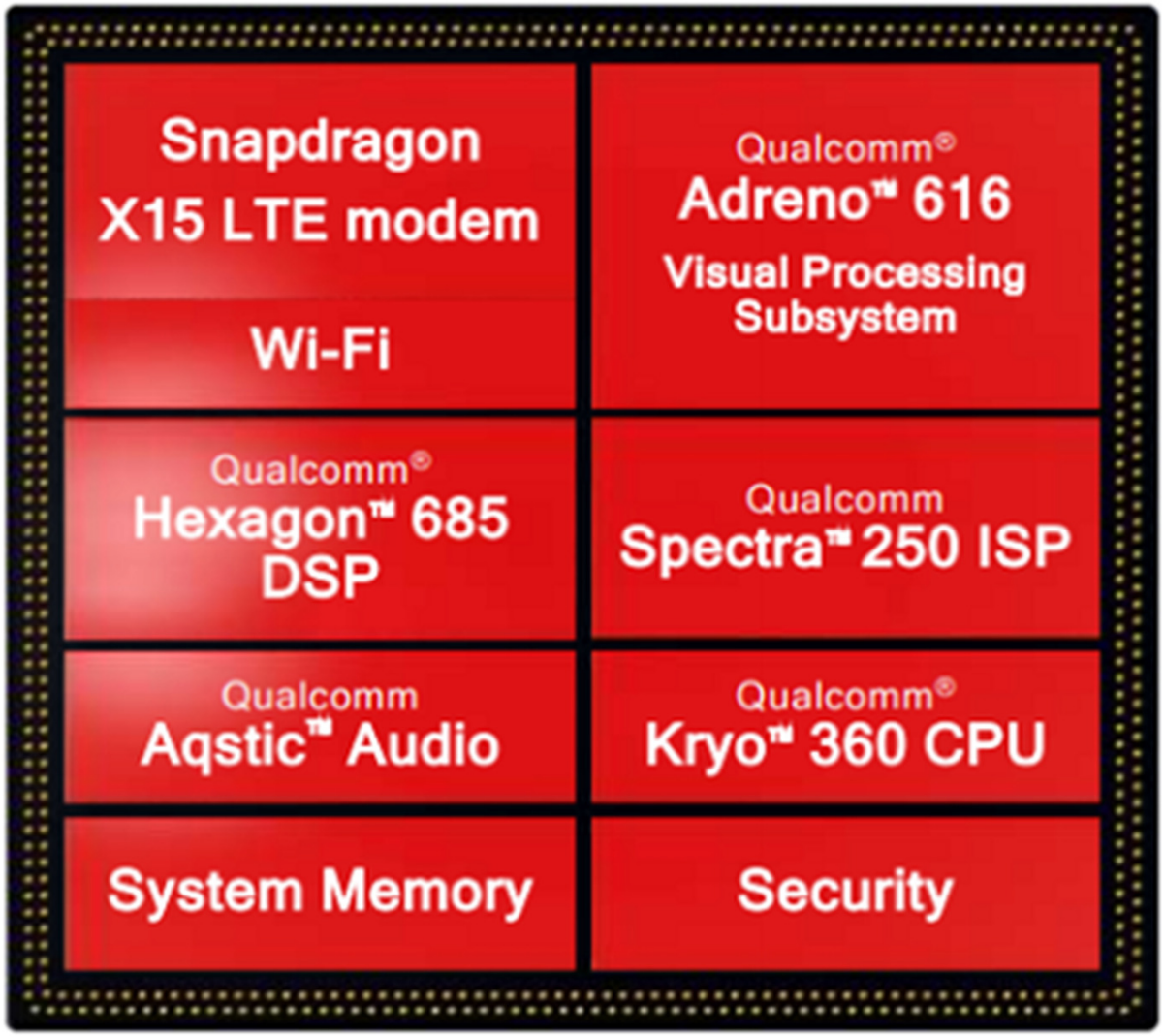
Smart Phone Components OEMs

Example of Multicore Arm SoC

- Apple: A15, M1, M1X, M2
  - 2x 3.23GHz (Firestorm) + 4x 2GHz (Icestorm) or 8 core, Neural engine, GPU
- **Qualcomm: SD888, SD870**
  - y. It has 1 KryoX1@2.8, 3 A78@2.4, 4 A55@1.8, AI, 5G, GPU
- Samsung: Exynos 9611
  - 4 A73@2.3Ghz, 4 A53@1.7Ghz, Mali G72, 5G, Codecs
- Huwai : hisilicon9000
  - 1 A77@3.13, 3 A77@2.54, 4 A55@2.0, Mali MP24, AI, 5G, neural
- **Mediatek : Dimensity 1200**
  - 1 A78@3.0, 3 A78@2.6, 4 A55@2.0, Mali MP24, 5G, AI,
- **Benchmarking: Antutu9, Geekbench 5, 3D Mark, Gaming FPS**

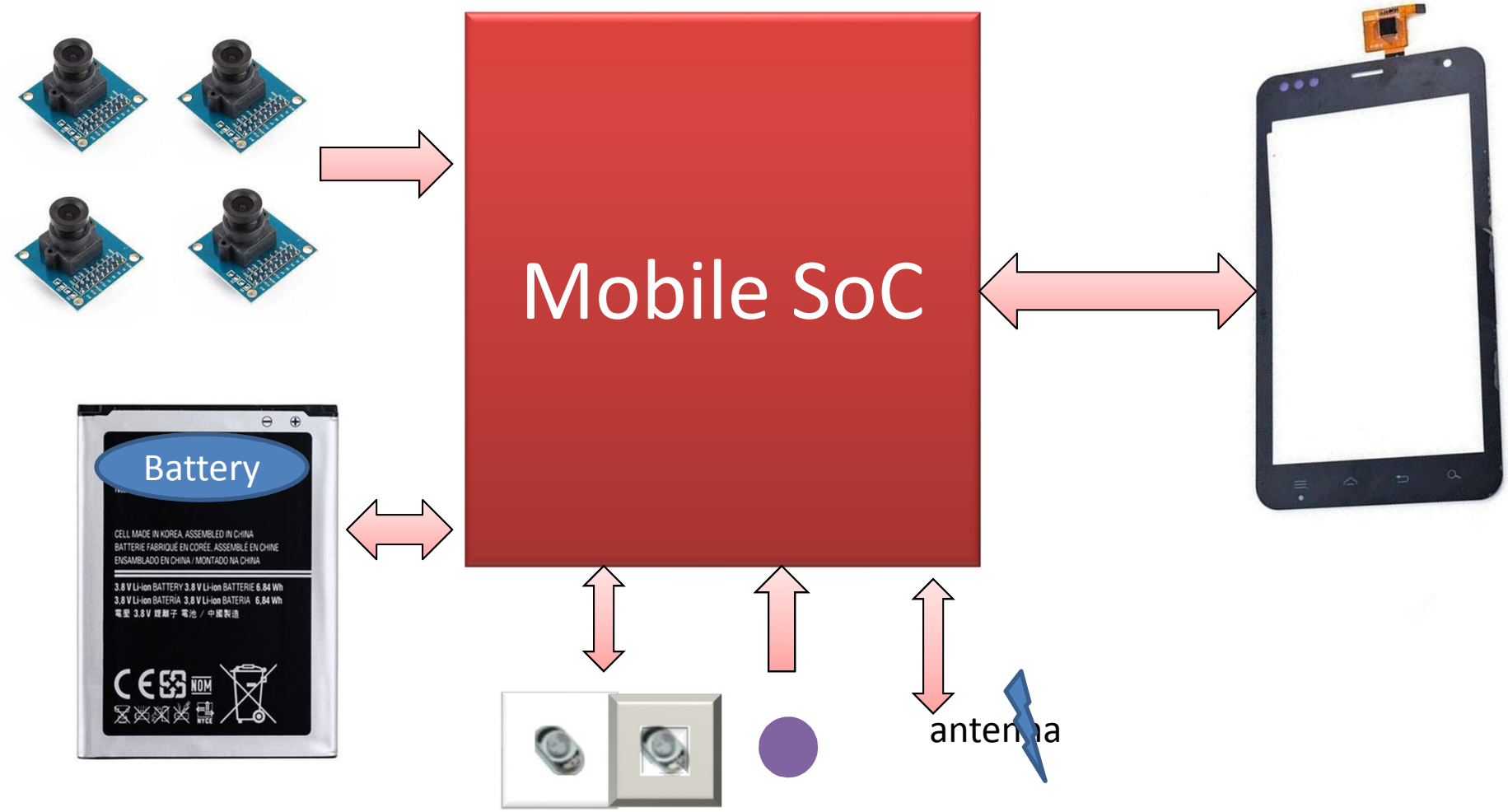
Mobile SoC Example

- Heterogeneous
- Diff H/W for different purpose
- Efficiency in terms
  - Perf.
  - Energy
- All in one Chip



Mobile SoC + Peripherals

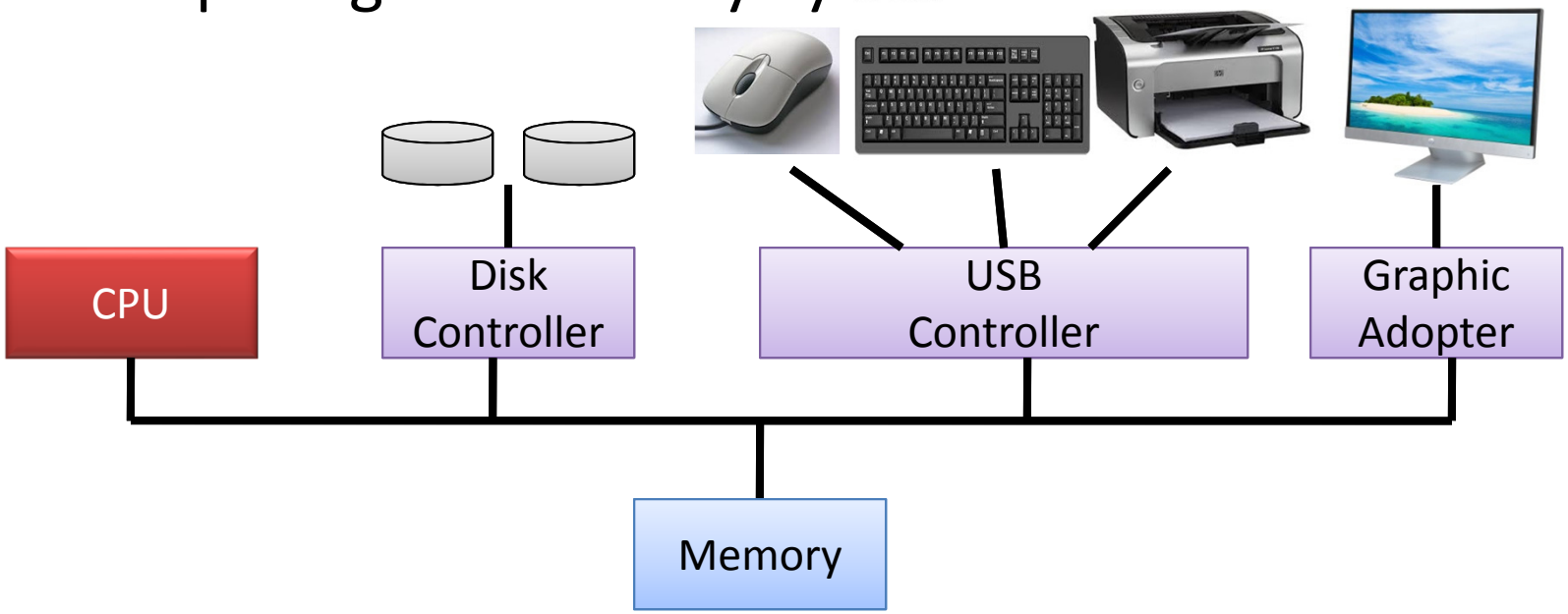
Similar to motherboard and components assembly
For every components we get dozens of variety to choose



Computer System Organization

Computer System Organization

- Computer-system operation
  - One or more CPUs, device controllers connect through common bus providing access to shared memory
  - Concurrent execution of CPUs and devices competing for memory cycles



Every Motherboard Have

- Processor and Memory (Need to Add)
- Controller (Programmable & Now a day -Smarter)
  - DMA Controller (Ex IC 8237) {Simiar to Taxi wala}
  - Interrupt Controller (Ex IC 82C59) {PA to Director}
  - Timer (Example IC 82C54) {An alram }
  - USB Controller (Example IC 8251) $lsusb
  - PCI Bus Controller $lspci
- All these devices have : Address to control , control register and status register

CS321: Smart System Laboratory (6th Semester)

How to write monitor or driver program in Assembly Language using 8085 or in C using PIC Microcontroller

Computer-System Operation

- I/O devices and the CPU can execute concurrently
- Each device controller is in charge of a particular device type
- Each device controller has a local buffer
- CPU moves data from/to main memory to/from local buffers
- I/O is from the device to local buffer of controller
- Device controller informs CPU that it has finished its operation by causing an interrupt

Common Functions of Interrupts

- Interrupt transfers control to
  - ISR generally, through the **interrupt vector**
  - **interrupt vector :** addresses of all service routines
- Interrupt architecture must save
  - Address of the interrupted instruction
- A **trap** or **exception**
  - Software-generated interrupt caused either by an error or a user request
  - It do not go to Interrupt controller, but it handle on its own.. (Similar to: if Director get a problem he don't ask to PA but solve himself)
- An operating system is **interrupt driven**

Interrupt Handling

- OS preserves the state of the CPU
  - By storing registers and the program counter
- Determines which type of interrupt has occurred:
  - **polling**
  - **vectored** interrupt system
- Separate segments of code determine what action should be taken for each type of interrupt

I/O Structure

- Blocking I/O methods
- After I/O starts, control returns to user program only upon I/O completion
  - Wait instruction idles the CPU until the next interrupt
  - Wait loop (contention for memory access)
  - At most one I/O request is outstanding at a time, no simultaneous I/O processing

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

Storage Definitions and Notation Review

- Bit : Basic unit of storage
- Byte 8 bits
  - Smallest convenient chunk of storage
- Notation
  - **kilobyte**, or **KB**, is 1,024 bytes $\approx 10^3$
  - **megabyte**, or **MB**, is $1{,}024^2$ bytes $\approx 10^6$
  - **gigabyte**, or **GB**, is $1{,}024^3$ bytes $\approx 10^9$
  - **terabyte**, or **TB**, is $1{,}024^4$ bytes $\approx 10^{12}$
  - **petabyte**, or **PB**, is $1{,}024^5$ bytes $\approx 10^{15}$

Storage Structure

- Main memory – only large storage media that the CPU can access directly
  - **Random access,** Typically **volatile**
- Secondary storage – extension of main memory that provides large **nonvolatile** storage capacity
- Hard disks – rigid metal or glass platters covered with magnetic recording material
  - Disk surface is logically divided into **tracks**, which are subdivided into **sectors**
  - The **disk controller** determines the logical interaction between the device and the computer
- **Solid-state disks** – faster than hard disks, nonvolatile
  - Various technologies, Becoming more popular

Storage Hierarchy

- Storage systems organized in hierarchy
  - Speed, Cost, Volatility
- **Caching** – copying information into faster storage system; main memory can be viewed as a cache for secondary storage
  - Locality of Reference
  - Locality Principle (Spatial and Temporal Locality)
- **Device Driver** for each device controller to manage I/O
  - Provides uniform interface between controller and kernel

Storage-Device Hierarchy