ET0023 Operating Systems
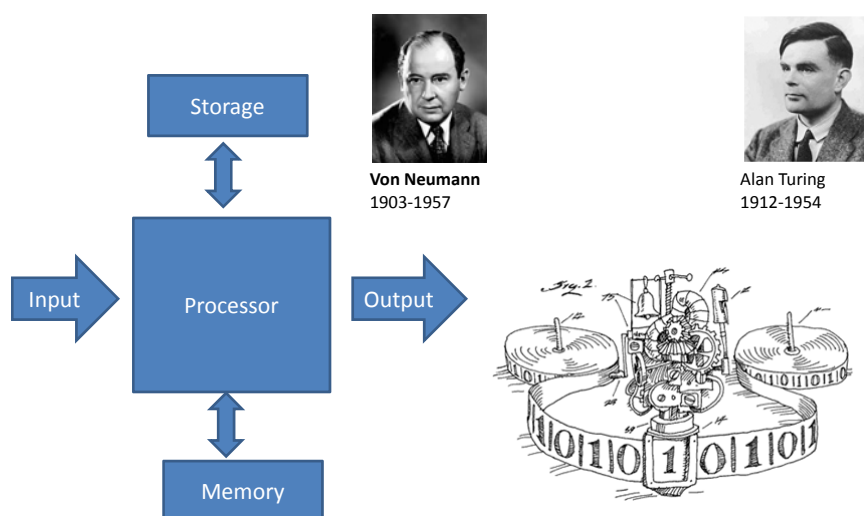
2: Computer Systems Overview

Computer System Architecture

Storage

Input

Processor

Output

Memory

Von Neumann
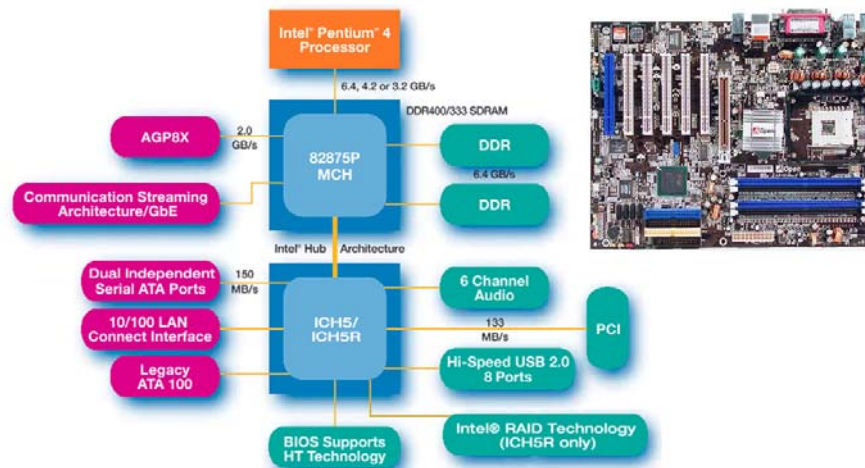1903-1957

Alan Turing
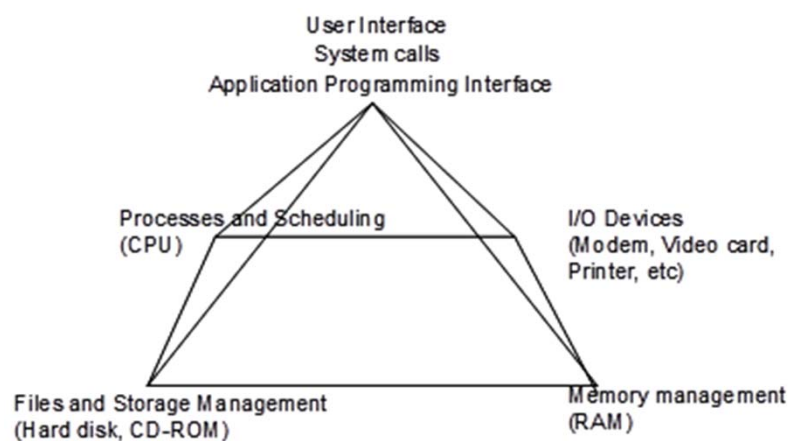1912-1954

Components of a PC
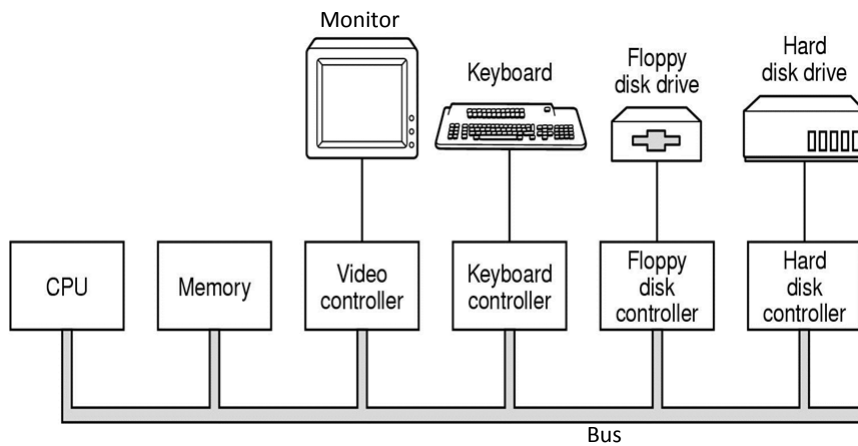
Generic x86 CPU System

- CPU
- Fast Memory Buses (Memory, Video)
  - Northbridge handles High Speed Ops
  - Southbridge handles Slower I/O
- Medium speed interfaces for Networking, USB, Hard-disk, DVD/CD Storage
- Slow speed mouse, keyboard and legacy I/O

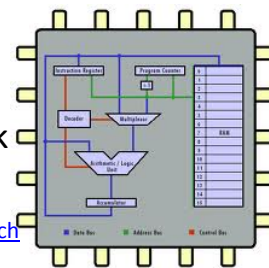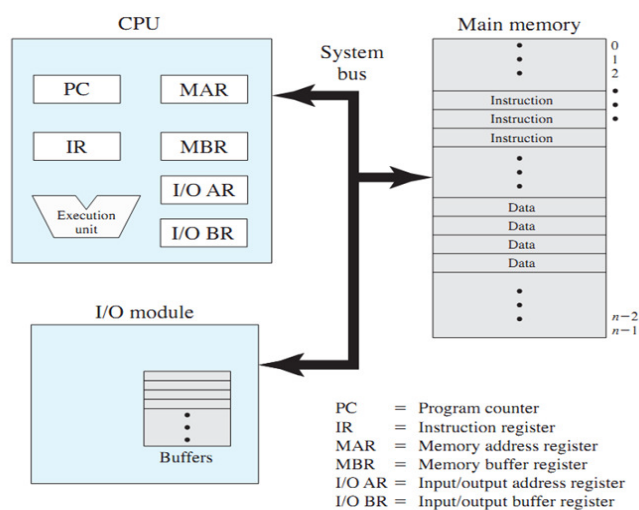# Intel P4 Hub chipset

OS Pyramid

The CPU

- Central Processing Unit built using VLSI
- Contains ALU, Registers, Program Counter, Control.
- Executes machine instructions, maintains the "heartbeat" of the system, controls operations
- Driven by a clock/pulse generator
- Each instruction completes in "X" clock cycles

http://courses.cs.vt.edu/csonline/MachineArchitecture/Lessons/CPU/Lesson.html

CPU Internal Structures

PC	=	Program counter
IR	=	Instruction register
MAR	=	Memory address register
MBR	=	Memory buffer register
I/O AR	=	Input/output address register
I/O BR	=	Input/output buffer register

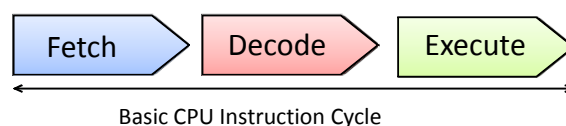http://www.c-jump.com/CIS77/CPU/VonNeumann/lecture.html

CPU Internal structures

- Address
  - Program Instruction Counter
  - Address pointers
- Memory
  - Registers
  - Stack Pointer
- ALU (Arithmetic Logic Unit)
  - Program Status Word
- Pipelined architectures

Fetch Execute Cycle

- CPU fetches instructions from memory, decodes and executes them
- Each CPU has specific set of instructions that it can execute
- CPU speed is measured based on the speed of executing instructions

Fetch → Decode → Execute

Basic CPU Instruction Cycle

General Processor Block Diagram
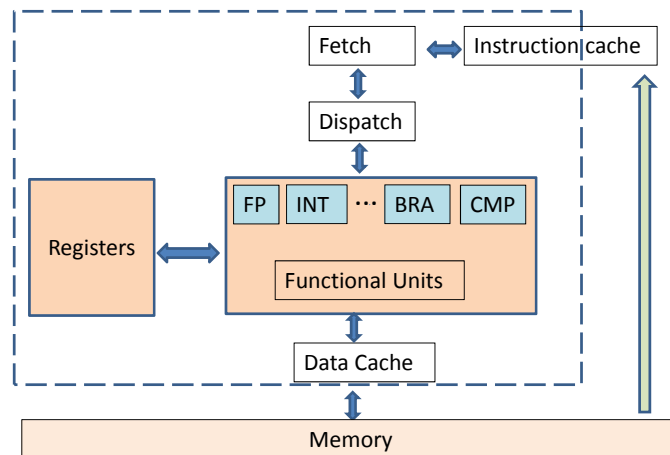
Three-bus Architecture

1. Instruction Fetch

- CPU indicates that it wants an instruction on the Control Bus
- CPU outputs address for next instruction to select Memory

2. Instruction Read

- CPU sends a "READ" to the memory
- CPU reads instruction from the memory
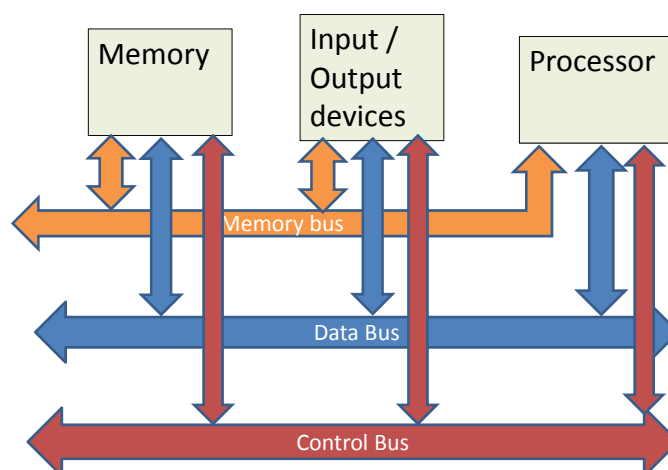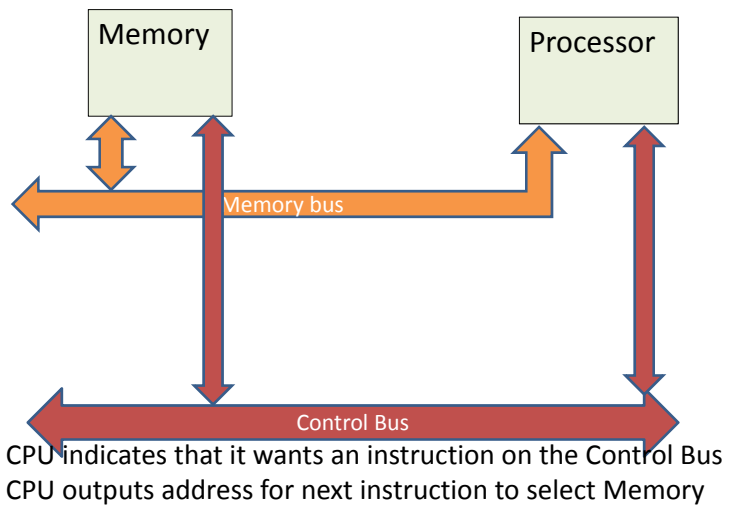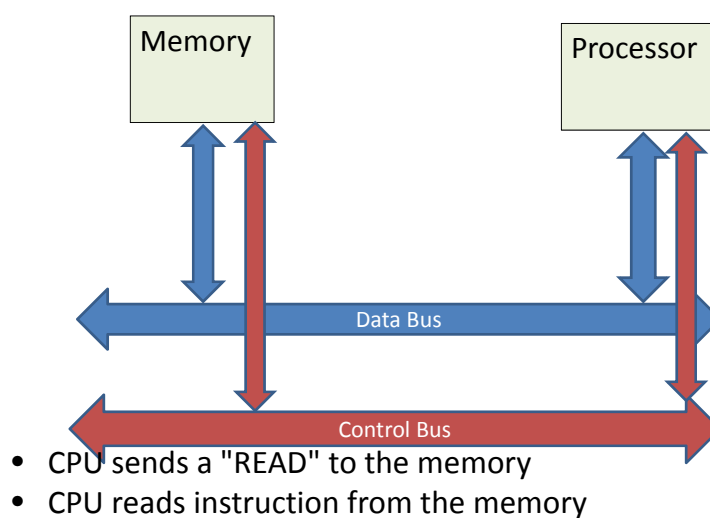
3. Instruction Decode/Execute

Memory

Processor

What do I need to do?

- CPU decodes the instruction based on the Functional Unit/Microcode
- CPU "Executes"
  - Perform an internal operation
  - Read another data
  - Write data
- Buses are NOT used during this period

Basic Instruction Cycle
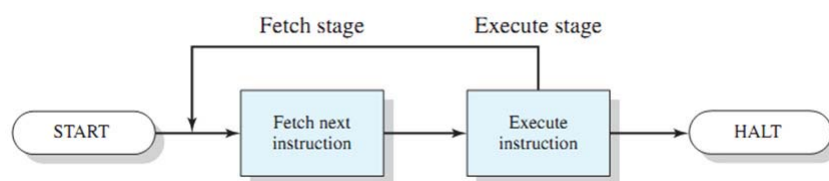
(a) Instruction format
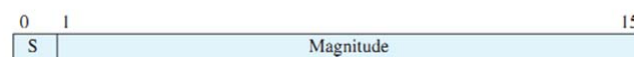
(b) Integer format

Examples: Instruction

- Intel x86 Instruction set
  - JMP 300
  - MOV AX, 940
  - ADD AX, 941
  - MOV 941, AX
  - IN

http://en.wikibooks.org/wiki/X86_Assembly/X86_Family

Program Execution

Operating Systems:
Internals & Design Principles
William Stallings
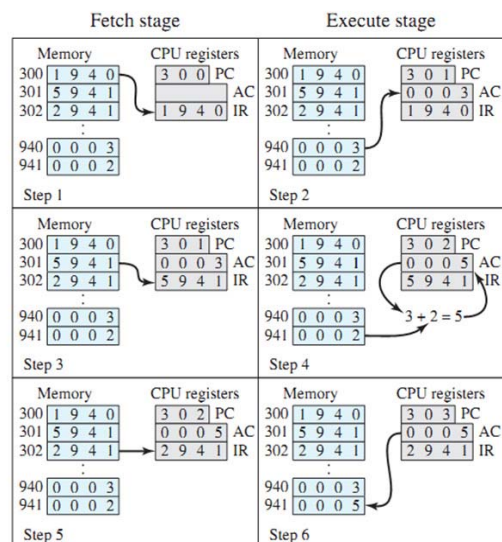
Figure 1.4   Example of Program Execution (contents of memory and registers in hexadecimal)

Memory Hierarchy

Typical access time | | Typical capacity

1 nsec	Register	<1 KB
2 nsec	Cache	4 MB
10 nsec	Main Memory	512 MB – 16 MB
10 msec	Magnetic Disk	500 GB – 2 TB
100 msec	Magnetic Tape	500 GB – 4 TB

What is the cost per GB for a typical 1TB HDD?

Prof. Carver A. Mead:
Moore's Law: The amount of transistors that can be placed on the same amount of silicon doubles every two years.

Memory

- Registers
  - Made of same material as the CPU thus just as fast
  - CISC vs RISC systems
- Cache Memory
  - High speed memory
  - As RAM is accessed in row format, stores row(s) of memory
  - Highly likelihood that next access is in the row
  - Cache Miss / Cache Hit

Cache Read Operations

RA—read address

START

Receive address
RA from CPU
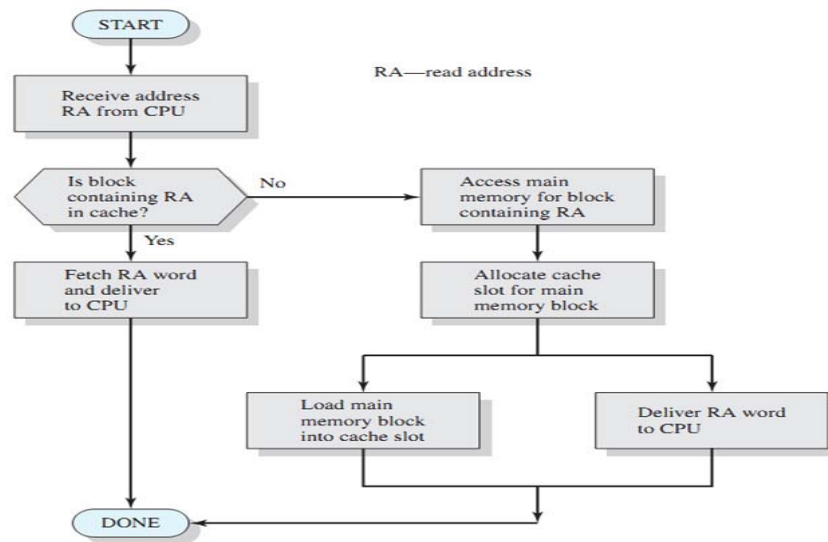
Is block
containing RA
in cache? — No → Access main
memory for block
containing RA

Yes

Fetch RA word
and deliver
to CPU

Allocate cache
slot for main
memory block

Load main
memory block
into cache slot

Deliver RA word
to CPU

DONE
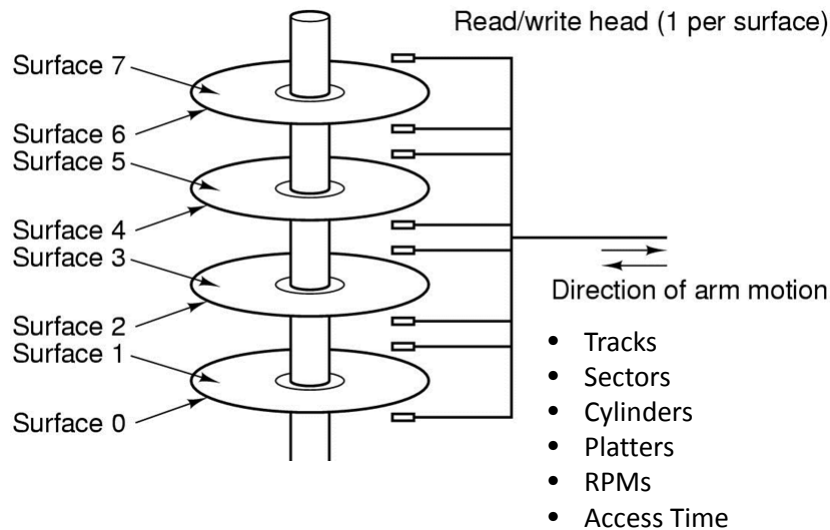
Memory

- RAM (Random Access Memory)
  - Usually dynamic/refreshed memory
  - Fast, large amounts of memory storage
- ROM (Read Only Memory)
  - Stores the startup/bootstrap program of the system
  - Stores BIOS/System Hardware routines
- EEPROM (Electrically Erasable ROM)
  - Able to be electrically erased and re-written
  - Allows flexibility in updating systems

Magnetic Storage

Read/write head (1 per surface)

Surface 7
Surface 6
Surface 5
Surface 4
Surface 3
Surface 2
Surface 1
Surface 0

Direction of arm motion

- Tracks
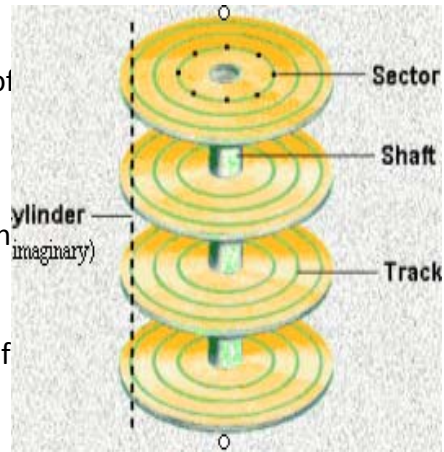- Sectors
- Cylinders
- Platters
- RPMs
- Access Time

Magnetic Storage Terms

- A disk consists of one or more metal/glass platters which may be accessed on 1 or 2 surfaces.
- Each platter is divided into a number of concentric tracks.
- A cylinder is made up of a track on each of the platters at the same horizontal position.
- Each track is divided into sectors which usually hold 512 bytes of user data.
- Outside tracks (OD)may hold more information than inside tracks (ID) on a platter.
- The platters spin at a high speed measured in RPM.
- Seek time is the time taken for the Read/Write head to move from one track to another for information.
- The amount of data that the drive can read/write per second is known as the data access rate.

Sectors and tracks

- Data is stored on the surface of a platter in **sectors** along the **tracks**.

- Tracks are *concentric* circles on the disk.

- And sectors are part portion of a track.

25

Typical Hard Disk Parameters

Table 1 Drive specifications summary for 1000 and 750	
Drive Specification	**ST31000524AS**
Formatted capacity (512 bytes/sector)*	1000GB
Guaranteed sectors	1,953,525,168
Heads	4
Disks	2
Bytes per sector	512
Default sectors per track	63
Default read/write heads	16
Default cylinders	16,383
Recording density (max)	1413kb/in
Track density (avg)	236 ktracks/in
Areal density (avg)	329Gb/in^2
Spindle speed	7200 RPM
Internal data transfer rate (max)	1695Mb/s
Sustained data transfer rate OD (max)	125MB/s
I/O data-transfer rate (max)	600MB/s
ATA data-transfer modes supported	
Cache buffer	32MB

Tape Drives

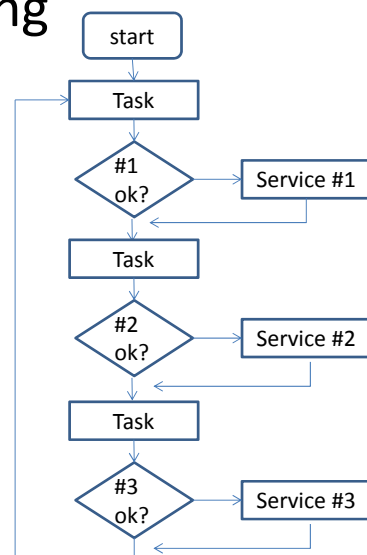- Backup medium for Hard Disk storage
- Holds large data sets
- Full/Incremental backups
- Storage is serial, may need operations to move to the correct locations
- Current Technology: LTO 3/4
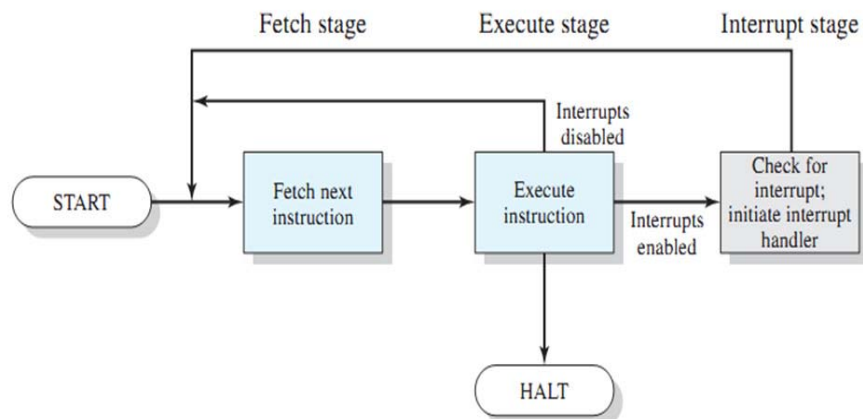- Capacities: 400GB-1.5TB @ 140MB/s

Polling

- Check each device in turn for status
- If service required, check the device
- Keeping doing the operation
- Easy to implement using software
- Hard to manage priorities

Interrupts

- Normal operations progress
- When device requires service, it sends a hardware request to the CPU
- CPU completes current instruction
- CPU saves the status of operations
- CPU determines who has interrupted and services device
- CPU restores old status

Instruction 1
Instruction 2
Instruction 3
Instruction 4
Instruction 5
Instruction 6

15

Interrupts

Instruction 1
Instruction 2
Interrupt Request
Instruction 3
Save status

Interrupt Service Routine
Service
Service
Finish

Restore status
Instruction 4
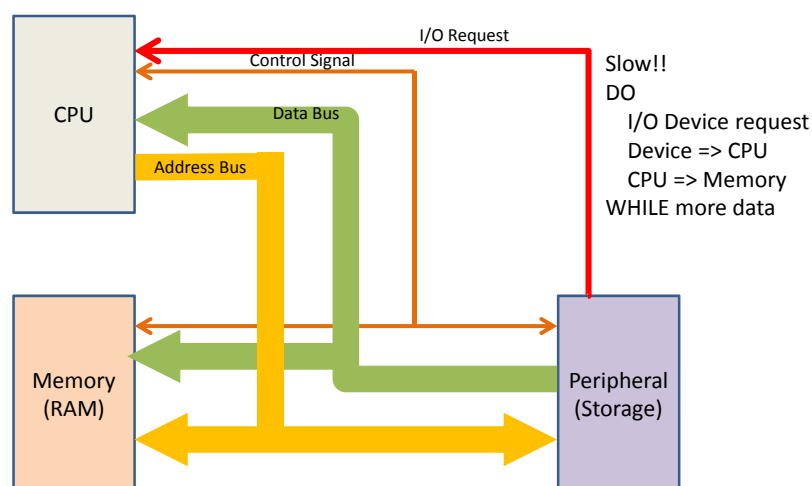Instruction 5

Instruction Cycle with Interrupts

Interrupt Strategies

- Interrupt request: Hardware/Software
- Need to determine who has interrupted
- How do we save the status?
- What do we need to save?
- Which Service routine do we execute?
- How do we return from the service routine?
- How do we restore the status?

Direct Memory Access

- Allows access to the system memory without CPU intervention.
- Specific hardware is used to transfer data from I/O device to Memory (DMA controller)
- Advantages: Overcomes slow I/O operations from the CPU
- Uses: Storage data transfers, Network transfers
- Main techniques:
  - Cycle Stealing
  - Block DMA requests
  - Transparent mode

Normal I/O Operations

CPU

I/O Request

Control Signal

Data Bus

Address Bus

Memory (RAM)

Peripheral (Storage)

Slow!!
DO
   I/O Device request
   Device => CPU
   CPU => Memory
WHILE more data

Direct Memory Access

User Interface

- Command Line Interface (CLI)
  - Command interpreter
  - Shells
  - Text input, can be scripted
- Graphical User Interface (GUI)
  - Windows, Gnome, KDI
  - Heavy use of memory and I/O
  - Keyboard, mouse, touch

Application Progg Interfaces (API)

- Functions, library for usage of the interface
- APIs provide libraries for
  - System calls, routines, transfers
  - Graphics
- Makes development of software
  - Consistent with the OS
  - Better controlled
  - Easier for the programmer

- QUESTIONS