Operating System

Dr. GuoJun LIU

Harbin Institute of Technology

http://guojunos.hit.edu.cn

Outline

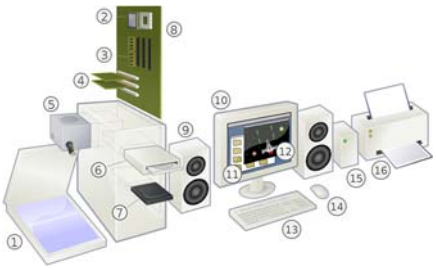
- History of computer
- Basic Elements
- Instruction execution
- Interrupt/Interrupt Processing
- Memory Hierarchy
- I/O Techniques
- Multiprocessor/multicore

Chapter 00

Computer System Overview

计算机系统概述

A modern PC and peripherals



An exploded view of a modern personal computer and peripherals

1. Scanner
2. CPU
3. Memory (RAM)
4. Expansion cards (graphics cards, etc.)
5. Power supply
6. Optical disc drive
7. Storage (Hard disk)
8. Motherboard
9. Speakers
10. Monitor
11. System software
12. Application software
13. Keyboard
14. Mouse
15. External hard disk
16. Printer

Learning Objectives

- Describe the basic elements of a computer system and their interrelationship
- Explain the steps taken by a processor to execute an instruction
- Understand the concept of interrupts and how and why a processor uses interrupts
- List and describe the levels of a typical computer memory hierarchy

Motherboard

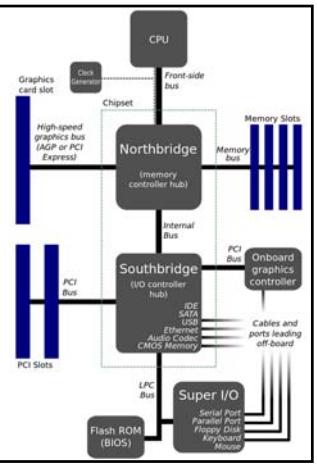
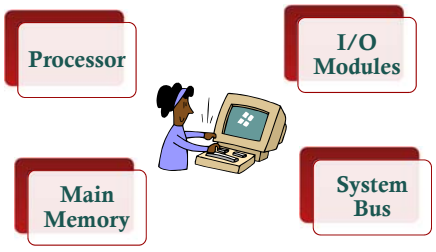
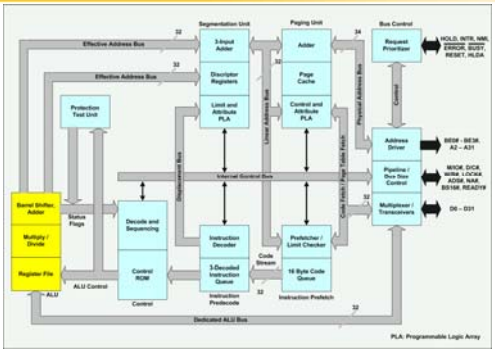


Photo from wikipedia.org

Basic Elements



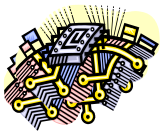
Block diagram of i386 microarchitecture



Processor

Controls the operation of the computer

Performs the data processing functions



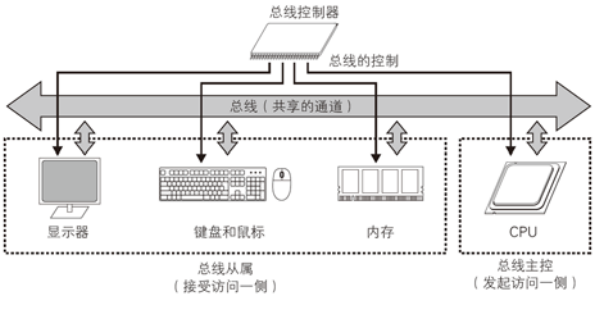
Referred to as the *Central Processing Unit* (CPU)

Main Memory

- Volatile
- Contents of the memory is lost when the computer is shut down
- Referred to as real memory or primary memory

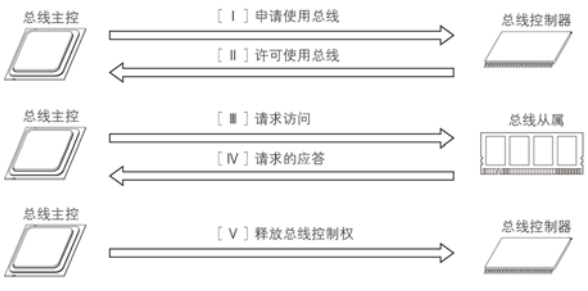
CPU

System Bus



图片来自《CPU自制入门》

System Bus

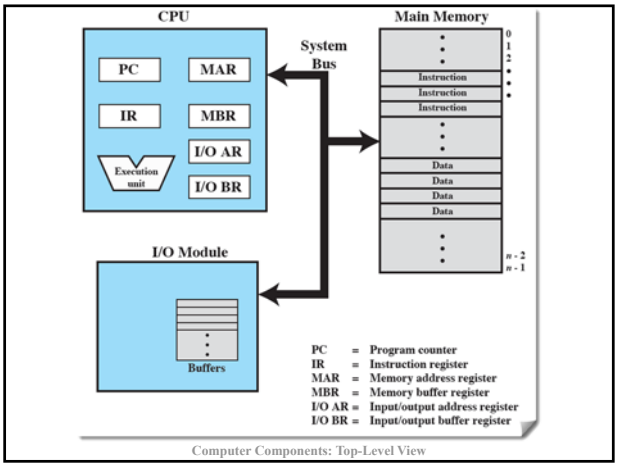


图片来自《CPU自制入门》

Instruction Register (IR)

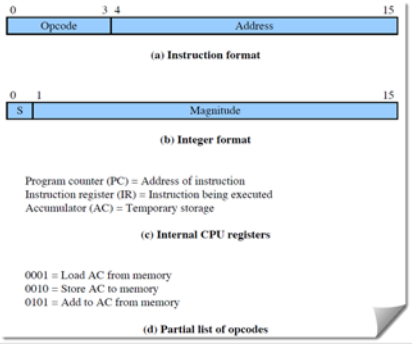
Fetched instruction is loaded into Instruction Register (IR)

- **Processor interprets the instruction and performs required action:**
  - Processor-memory
  - Processor-I/O
  - Data processing
  - Control



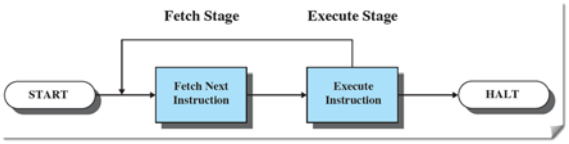
Computer Components: Top-Level View

Characteristics of a Hypothetical Machine

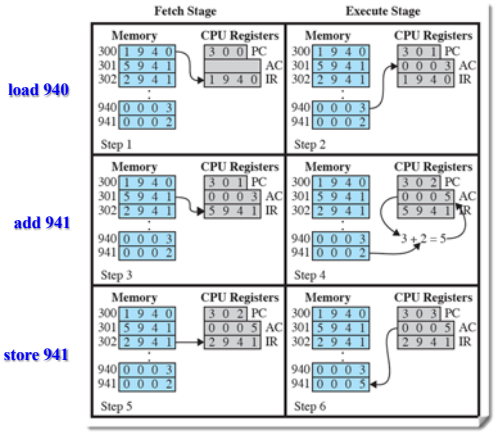


Instruction Execution

- **A program consists of a set of instructions stored in memory**
  - processor reads (fetches) instructions from memory
  - processor executes each instruction



Basic Instruction Cycle

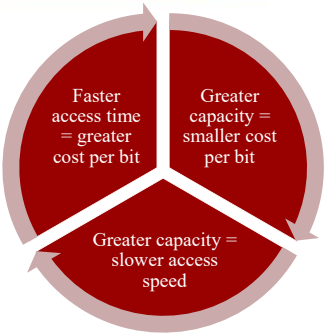


Example of Program Execution

Memory Hierarchy

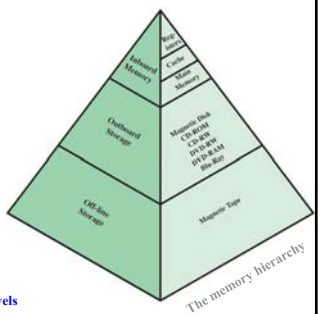
- **Major constraints in memory**
  - amount -- How much?
  - speed -- How fast?
  - Expense (cost) -- How expensive?
- **Memory must be able to keep up with the processor**
- **Cost of memory must be reasonable in relationship to the other components**

Memory Relationships



The Memory Hierarchy

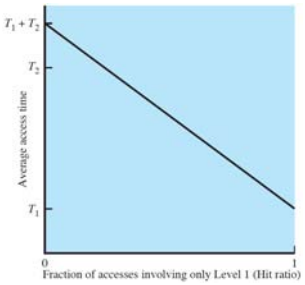
- **Going down the hierarchy:**
  a. decreasing cost per bit
  b. increasing capacity
  c. increasing access time
  d. decreasing frequency of access to the memory by the processor



Decreasing frequency of access at lower levels

Performance of a Simple Two-Level Memory

- **Hit ratio H**
  - where H is defined as the fraction of all memory accesses that are found in the faster memory (e.g., the cache)



$$(0.95)\,(0.1\ \mu s) + (0.05)\,(0.1\ \mu s + 1\ \mu s) = 0.095 + 0.055 = 0.15\ \mu s$$

Principle of Locality

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

Secondary Memory

- **Also referred to as auxiliary memory**
  - External
  - Nonvolatile
  - Used to store program and data files

Cache Memory

- Invisible to the OS
- Interacts with other memory management 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

Classes of Interrupts

Program

- Generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, and reference outside a user's allowed memory space

Timer

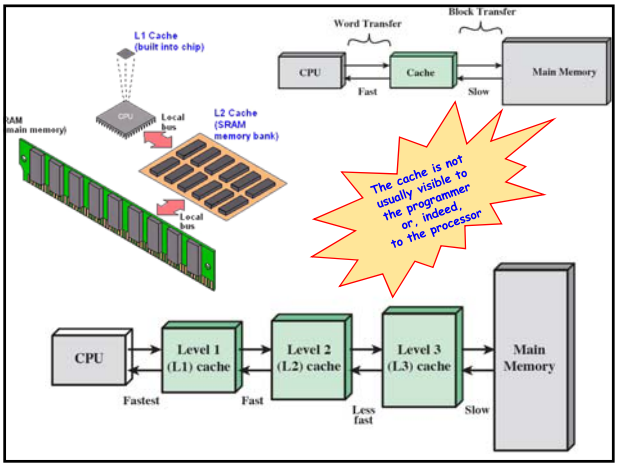
- Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis

I/O

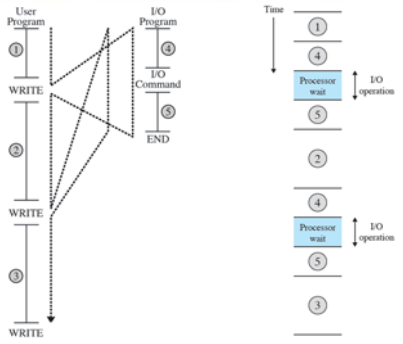
- Generated by an I/O controller, to signal normal completion of an operation or to signal a variety of error conditions

Hardware failure

- Generated by a failure, such as power failure or memory parity error

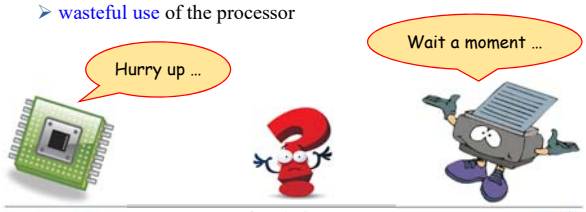


Flow of Control Without Interrupts

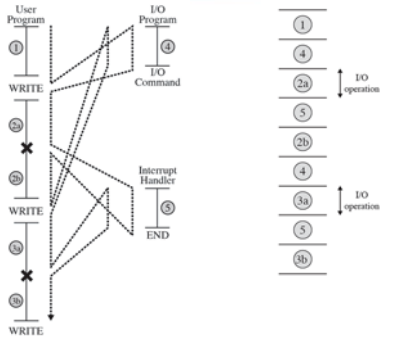


Interrupts

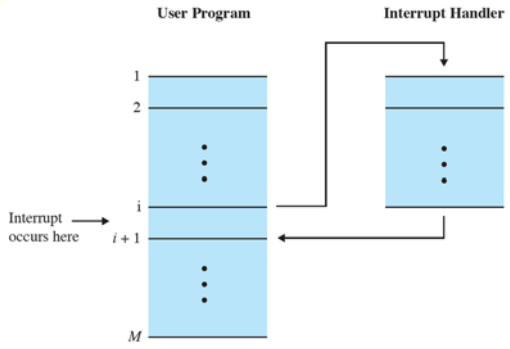
- Interrupt the normal sequencing of the processor
- Provided to improve processor utilization
  - most I/O devices are slower than the processor
  - processor must pause to wait for device
  - wasteful use of the processor



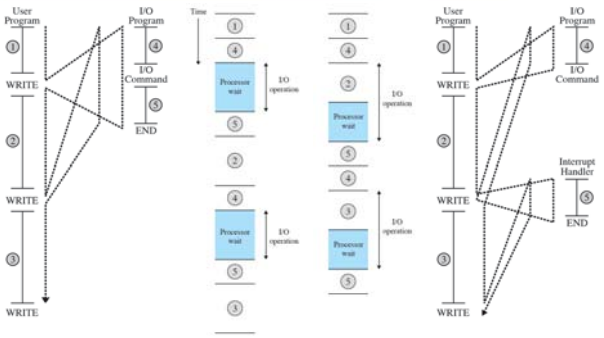
Interrupts: Short I/O Wait



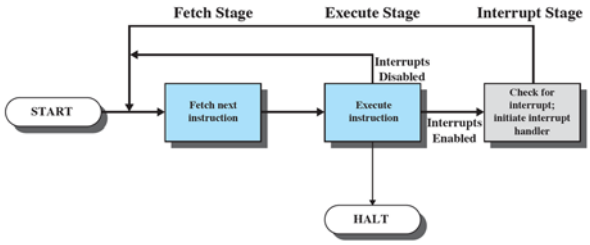
Instruction Cycle With Interrupts

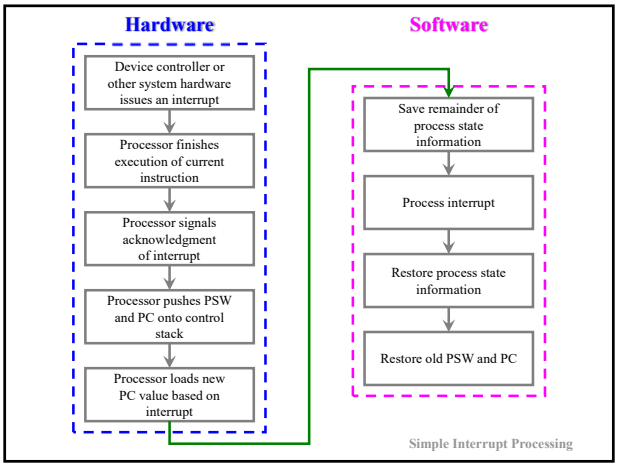


Program Timing: Long I/O wait



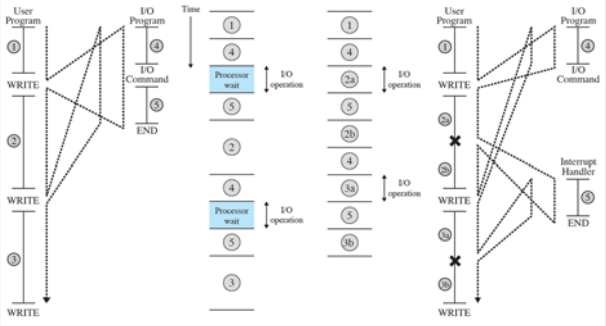
Instruction Cycle With Interrupts



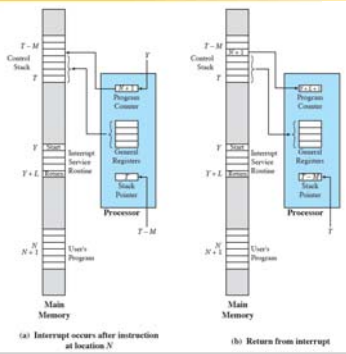


Simple Interrupt Processing

Program Timing: Short I/O Wait



Changes for an Interrupt



(a) Interrupt occurs after instruction at location N (b) Return from interrupt

Multiple Interrupts

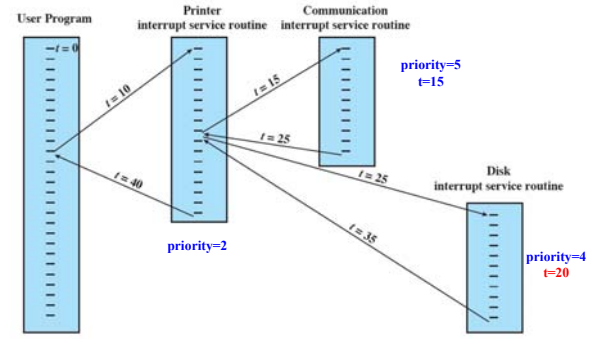
An interrupt occurs while another interrupt is being processed

- receiving data from a communications line and printing results at the same time

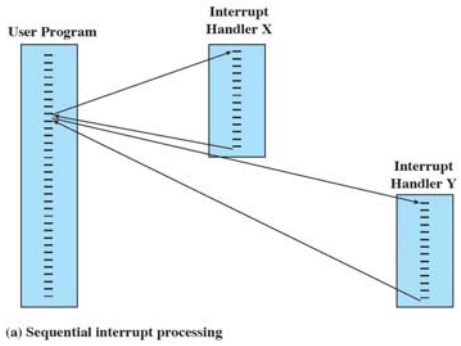
Two approaches:

- disable interrupts while an interrupt is being processed
- use a priority scheme

Example Time Sequence of Multiple Interrupts



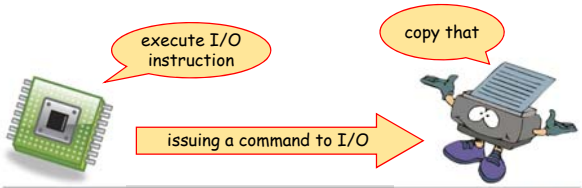
Sequential interrupt processing



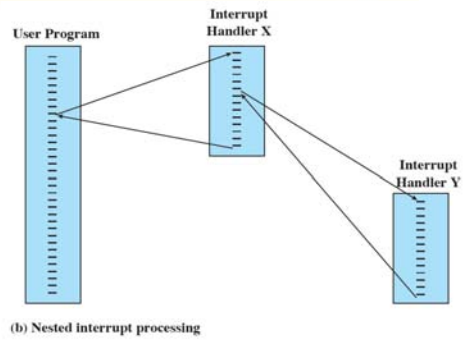
(a) Sequential interrupt processing

I/O Techniques

- programmed I/O
- interrupt-driven I/O
- direct memory access (DMA)



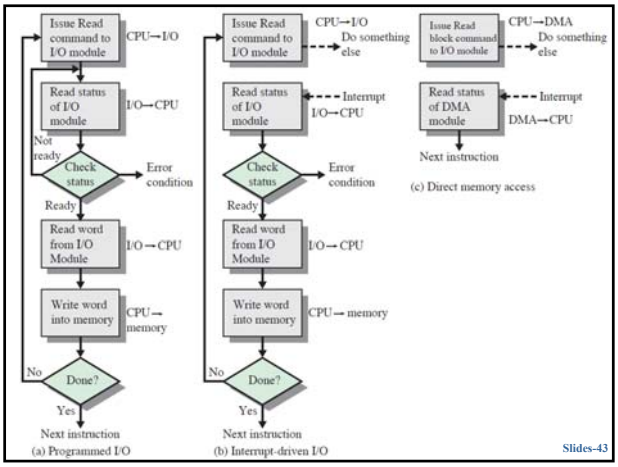
Nested interrupt processing



(b) Nested interrupt processing

DMA

- It allows certain hardware subsystems to access main system memory independently of CPU
  - performed by a separate module on the system bus
  - incorporated into an I/O module
- It issues a command to the DMA module containing
  - whether a read or write is requested
  - the address of the I/O device involved
  - the starting location in memory to read/write
  - the number of words to be read/written



Compare three techniques

- ☒ Transfer rate is limited by the speed with which the processor can test and service a device
- ☒ The processor is tied up in managing an I/O transfer, a number of instructions must be executed for each I/O transfer
- ☑ processor is involved only at the beginning and end of the transfer
- ☑ More efficient
- ☒ processor executes more slowly during a transfer when processor access to the bus is required

DMA
programmed I/O
interrupt-driven I/O

Multicore Computer

- Also known as a chip multiprocessor
- Combines two or more processors (cores) on a single piece of silicon (die)
  - each core consists of all of the components of an independent processor
- In addition, multicore chips also include L2 cache and in some cases L3 cache

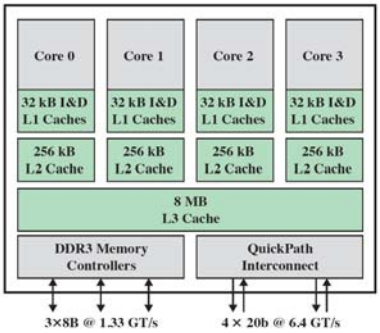
Intel Core i7

Supports two forms of external communications to other chips:

DDR3 Memory Controller
- brings the memory controller for the DDR (double data rate) main memory onto the chip
- with the memory controller on the chip the Front Side Bus is eliminated

QuickPath Interconnect (QPI)
- enables high-speed communications among connected processor chips

Intel Core i7 Block Diagram



Summary

- Basic Elements
  - processor, main memory, I/O modules, system bus
- Instruction execution
  - processor-memory, processor-I/O, data processing, control
- Interrupt/Interrupt Processing
- Memory Hierarchy
  - Cache/cache principles and designs
- I/O Techniques
- Multiprocessor/multicore