Operating System

Dr. GuoJun LIU

Harbin Institute of Technology

http://guojunos.hit.edu.cn

Chapter 00

Computer System Overview

计算机系统概述

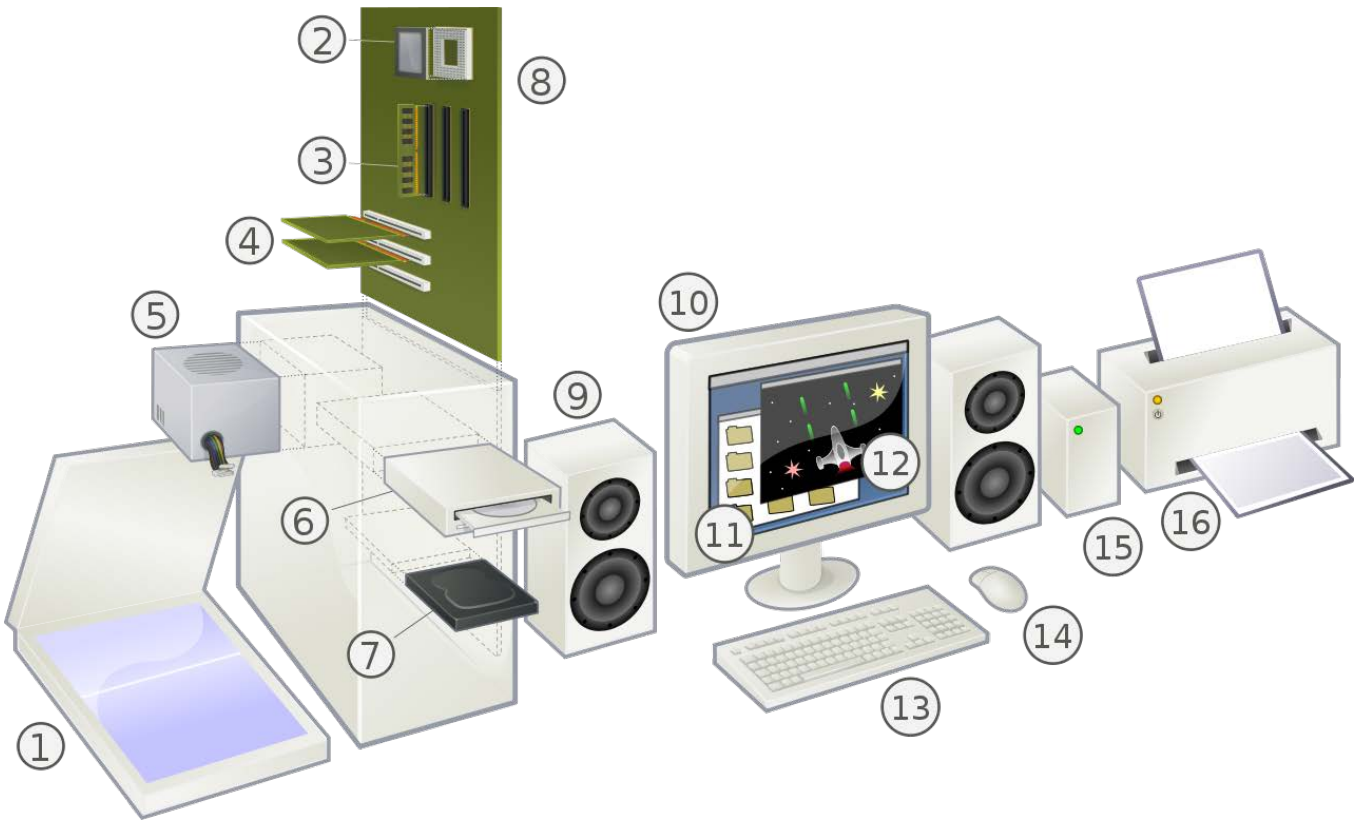
Learning Objectives

- 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

Outline

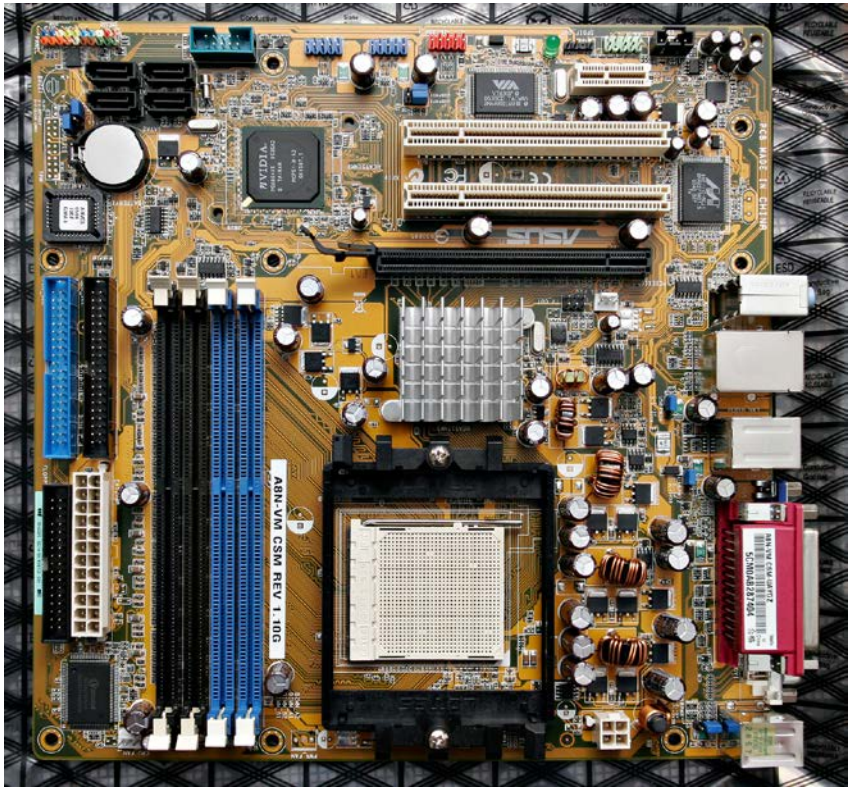
- History of computer
- Basic Elements
- Instruction execution
- Interrupt/Interrupt Processing
- Memory Hierarchy
- I/O Techniques
- Multiprocessor/multicore

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



Motherboard

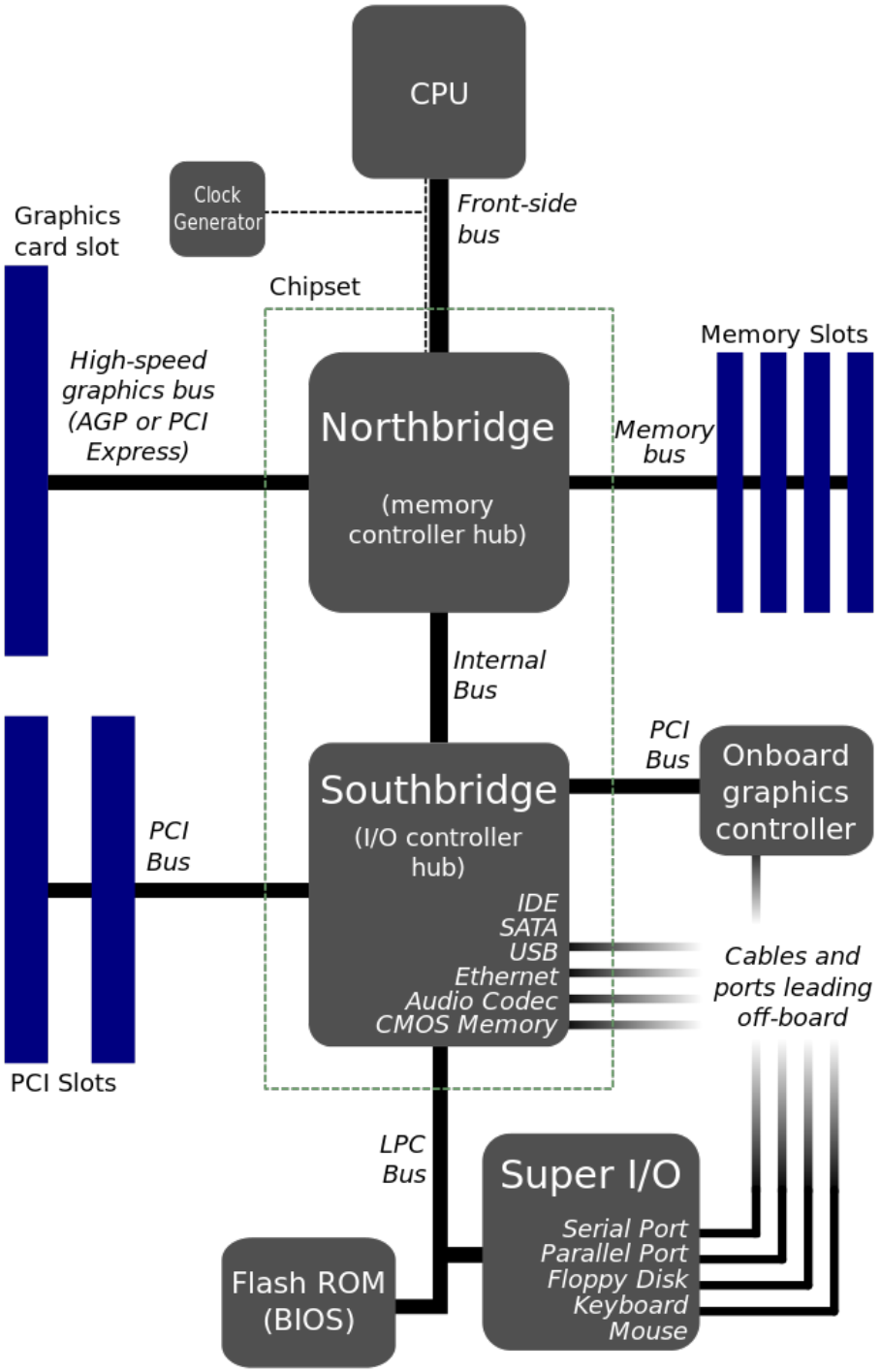
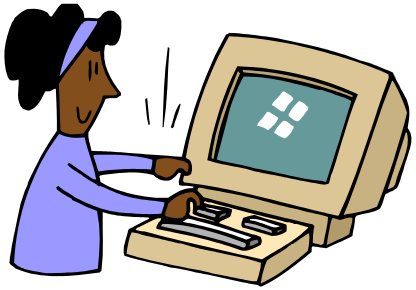


Photo from wikipedia.org

Basic Elements

Processor

I/O Modules



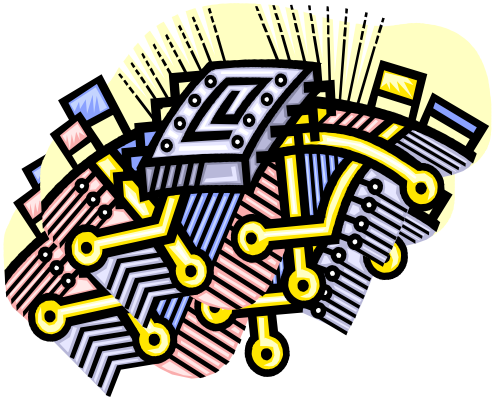
Main Memory

System Bus

Processor

Controls the operation of the computer

Performs the data processing functions



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

CPU

Processor Families: Xeon Celeron D Celeron M Celeron Core Duo Core Solo Core 2 Duo Pentium 4-M Pentium 4 Pentium D Pentium III Pentium III Xeon Pentium Itanium Core 2 Extreme Core 2 Solo Atom Core 2 Quad Core i7 Core i7 Extreme Core i5 Core i3 Pentium II Mobile Pentium II Xeon Pentium III Mobile 4004 8008 8080 8086 8088 80286 80386 DX 80486 DX 80486 DX2 80486 DX4 Core 2 Quad Extreme Itanium 2 Pentium 4 EE Pentium EE Pentium II Pentium M Pentium MMX Pentium Pro 80186 iAPX432 i860 i960

Microarchitectures: Core:Merom Core:Conroe Core:Kentsfield Core:Penryn Core:Yorkfield Core:Wolfdale Madison Mckinley Merced Montecito Montvale Nehalem NetBurst:Willamette NetBurst:Northwood NetBurst:Prescott NetBurst:Cedar Mill NetBurst:Smithfield NetBurst:Presler P5 P6:Pentium II P6:Pentium III P6:Banias P6:Yonah Sandy Bridge 80386 DX 80486 DX Bonnel

Code Names: Irwindale Cranford Prescott Cedar Mill Banias Dothan Yonah Coppermine T Tualatin Northwood Mendocino Dempsey Paxville Merom Foster Tulsa Gallatin Coppermine Willamette Prescott-2M Smithfield Presler Katmai Tanner Cascades Madison Montecito Kentsfield Conroe Sossaman Clovertown Woodcrest Tigerton Montvale Harpertown Penryn Yorkfield Wolfdale Silverthorne Dunnington Nehalem-EP Bloomfield Lynnfield Tukwila Clarksfield Arrandale Clarkdale Pineview Nehalem EX Westmere-EP Gulftown Lincroft N/A Dixon Sandy Bridge Westmere EX Cedarview Sandy Bridge-E Sandy Bridge-EP Sandy Bridge-EN Ivy Bridge Diamondville Tunnel Creek Stellarton Conroe-L Conroe-CL Covington Allendale Wolfdale-3M Penryn-3M Merced Mckinley P5 Prescott 2M P55C Gainestown Jasper Forest

Processor Families: 80386 80486 Athlon Athlon 64 Athlon 64 X2 Athlon II X4 Athlon MP Athlon X2 Athlon XP K5 K6 K6-2 K6-III Opteron Phenom II X3 Phenom II X4 Phenom X3 Phenom X4 Turion 64 Turion 64 X2 Am29000 Am29030 Am29035 Am29040 Am29050

Microarchitectures: K10 K5 K6 K6-2 K6-III K7 K8 K75

Code Names: DX Pluto Thunderbird ClawHammer SledgeHammer San Diego Toledo Windsor Manchester Brisbane Tyler Lion Propus Palomino Thoroughbred Kuma Barton SSA/5 Model 6 Little Foot Chomper Chomper Extended Sharptooth Venus Denmark Troy Italy Athens Egypt Santa Ana Budapest Santa Rosa Barcelona Shanghai Istanbul Heka Deneb Toliman Agena Lancaster Trinidad

Processor Families: microSPARC I SuperSPARC I SuperSPARC II UltraSPARC UltraSPARC II UltraSPARC IIe UltraSPARC IIi UltraSPARC III Cu UltraSPARC IIIi UltraSPARC T1 UltraSPARC T2

Microarchitectures: SuperSPARC I SuperSPARC II UltraSPARC UltraSPARC II UltraSPARC III UltraSPARC T1

Code Names: Tsunami Viking Voyager Spitfire Blackbird Sapphire-Black Hummingbird Sabre Sapphire-Red Phantom Cheetah Cheetah+ Jalepeno Niagara Niagara 2

Processor Families: 6800 68000 68010 68020 68030 68040 68060 PowerPC 88000

Microarchitectures: 68000 68010 68020 68030 68040 68060

Code Names: Max Apollo 6

Processor Families: Alpha 21064 Alpha 21064A Alpha 21066A Alpha 21164 Alpha 21164A Alpha 21164PC Alpha 21264 Alpha 21264A Alpha 21264B Alpha 21264C Alpha 21364 StrongARM

Microarchitectures: EV4 EV5 EV6 EV7

Code Names: EV4 EV45 LCA45 EV5 EV56 PCA56 PCA57 EV6 EV67 EV68A EV68CB EV7 EV7z

Processor Families: Power2 Power3 Power3-II Power4+ Power5 Power6 Power7 PowerPC 7xx PowerPC 970 PowerPC 602 PowerPC 603 PowerPC 620 PowerPC 630 PowerPC 601 PowerPC 601v PowerPC 603e PowerPC 603ev PowerPC 604 PowerPC 604e PowerPC 604ev RS64 IV RS64-II RS64-III

Microarchitectures: Power6 PowerPC v2.00 PowerPC v2.02 PowerPC 601 PowerPC 602 PowerPC 603e PowerPC 604 PowerPC 620 PowerPC 630 PowerPC 7400 PowerPC 7450 PowerPC 7xx Power7

Code Names: P2SC RIOS2 Power2+ Giga Processor Arthur GP-UL Mach 5 Sstar

Processor Families: PA-7000 PA-7100 PA-7100LC PA-7150 PA-7200 PA-7300LC PA-8000 PA-8200 PA-8500 PA-8600 PA-8700 PA-8700+ PA8800 PA8900

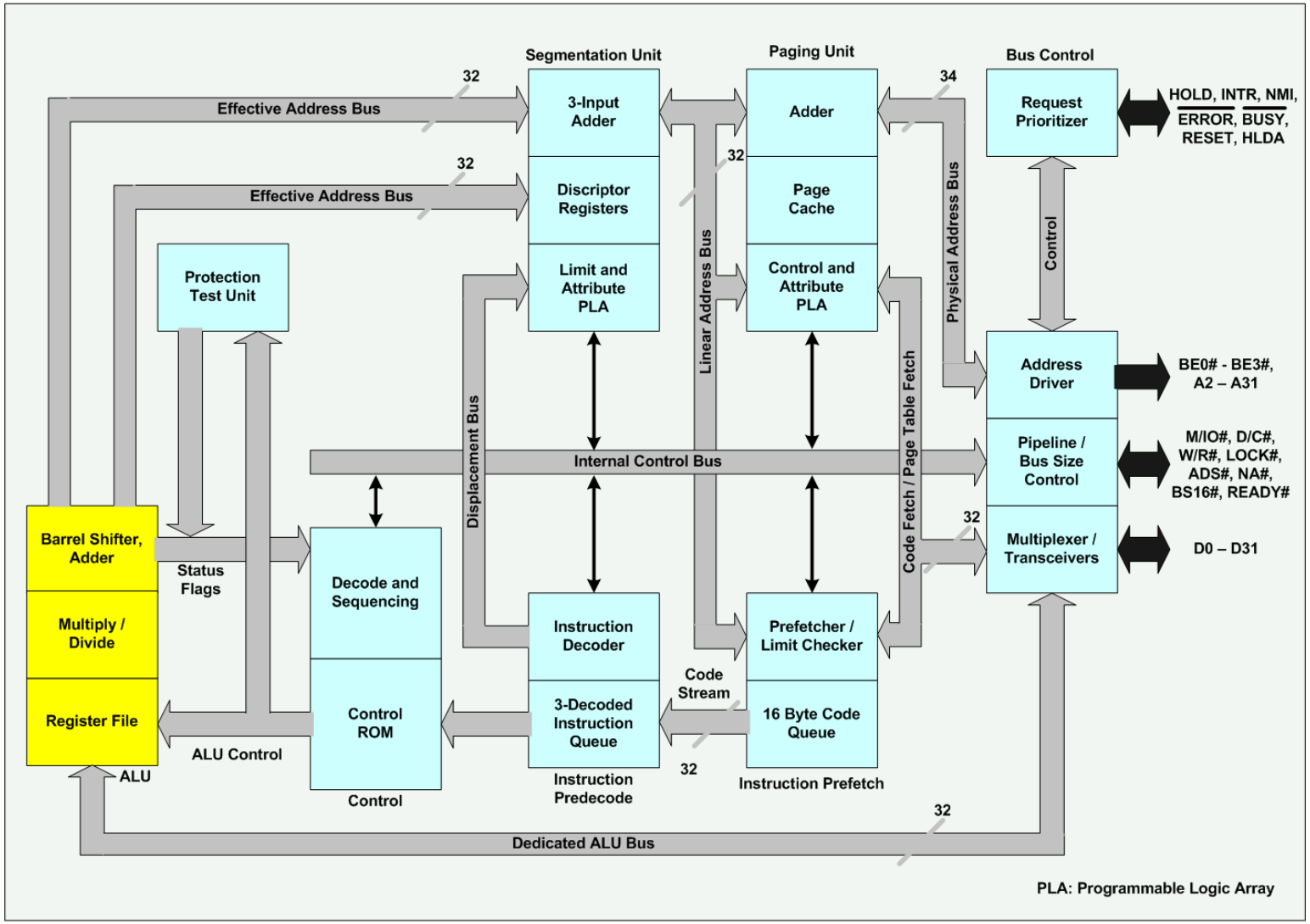
Microarchitectures: PA-7300LC PA-8000 PA-8200 PA-8500 PA-8600 PA-8700 PA-8700+

Processor Families: R10000 R2000 R3000 R4000 R4400 R4600 R5000 R8000 74K 7000 R4300 3520 5900

Microarchitectures: R10000 R2000 R3000 R4000 R4600 R5000 R8000

Code Names: T5 Orion

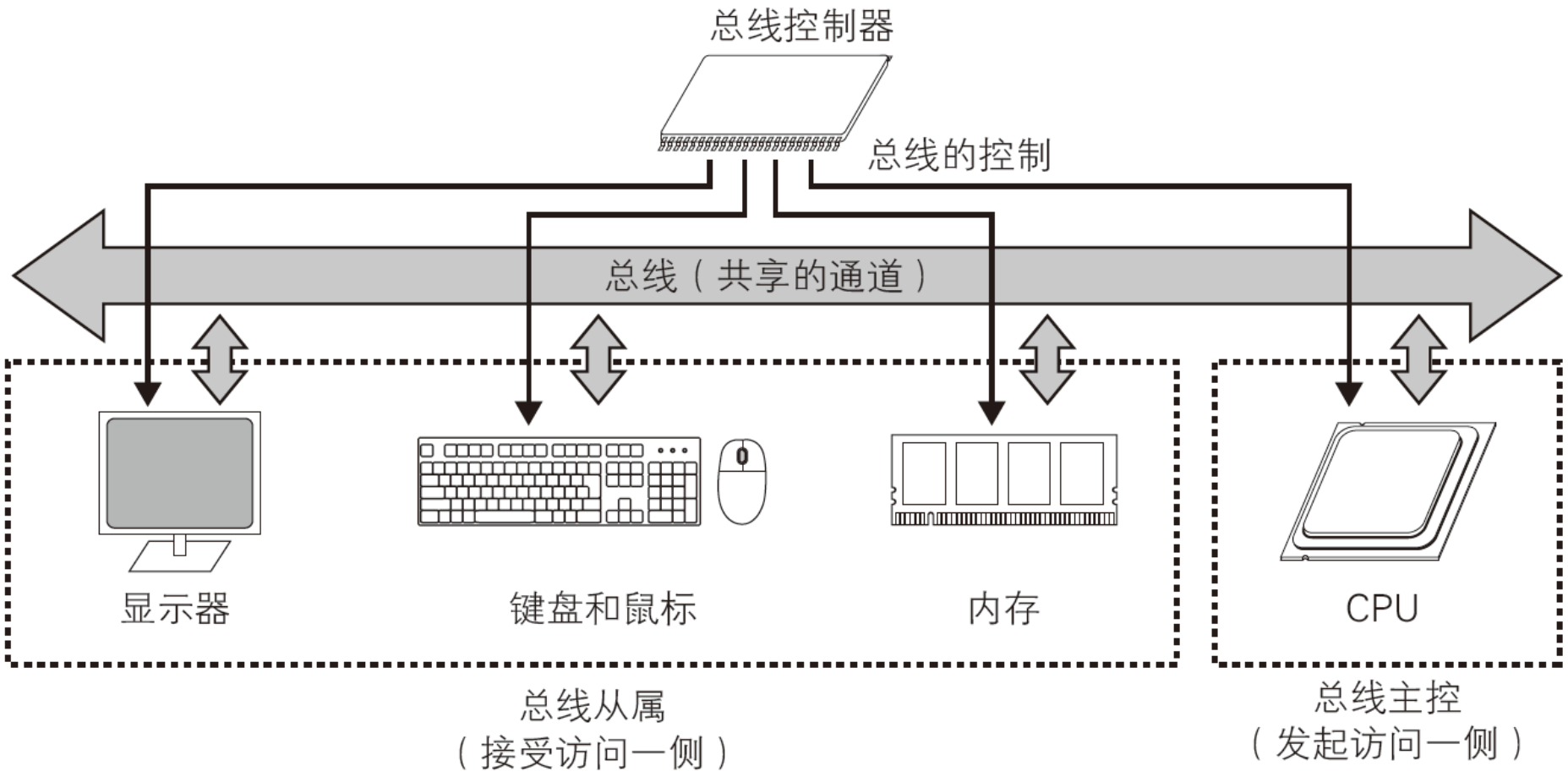
Block diagram of i386 microarchitecture



Main Memory

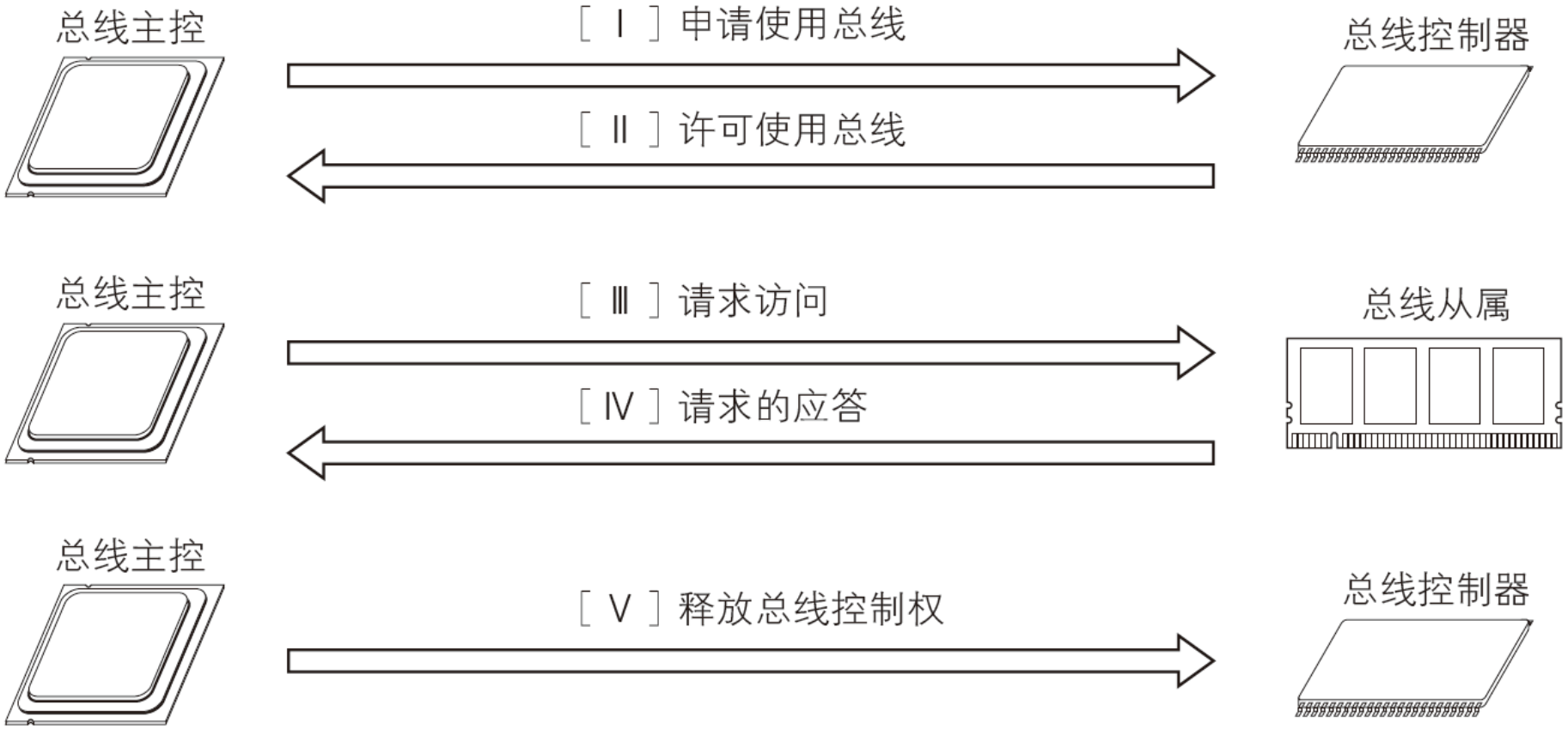
- **Volatile**
- **Contents of the memory is lost when the computer is shut down**
- **Referred to as real memory or primary memory**

System Bus

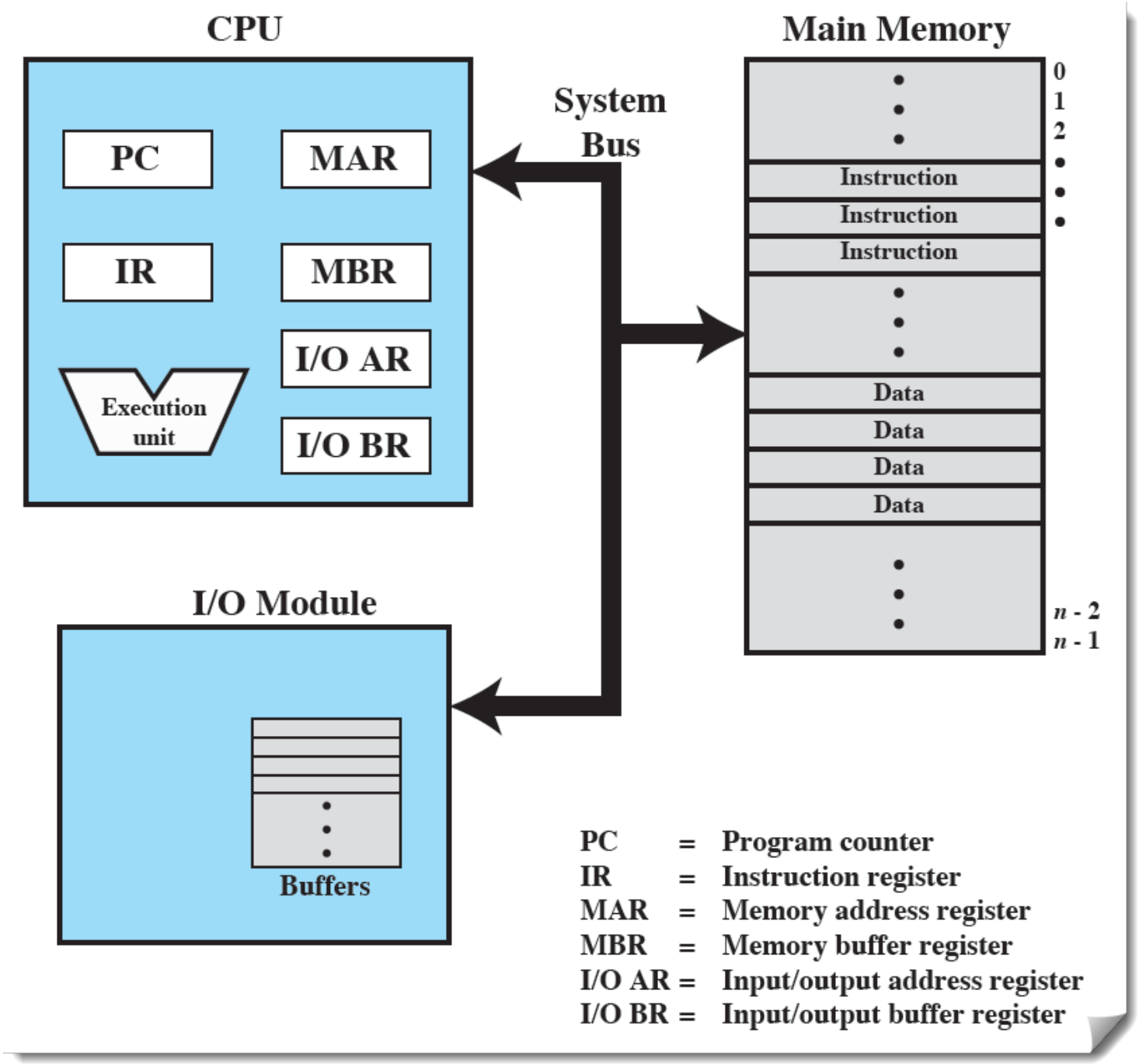


图片来自《CPU自制入门》

System Bus

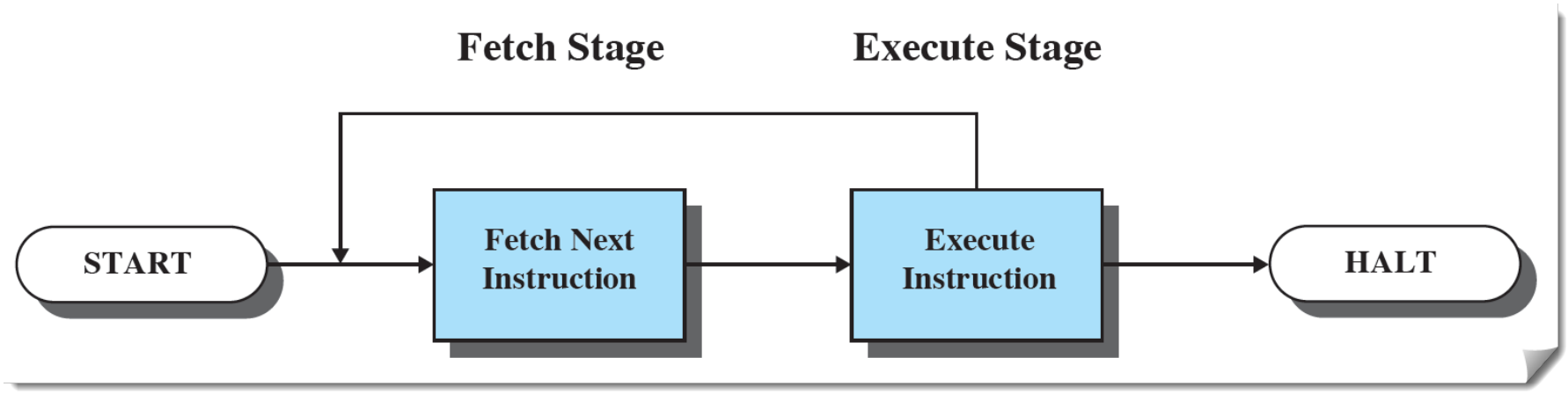


图片来自《CPU自制入门》


CPU
PC
MAR
IR
MBR
I/O AR
Execution unit
I/O BR
System Bus
Main Memory
0
1
2
Instruction
Instruction
Instruction
Data
Data
Data
Data
n - 2
n - 1
I/O Module
Buffers
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

Instruction Execution

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



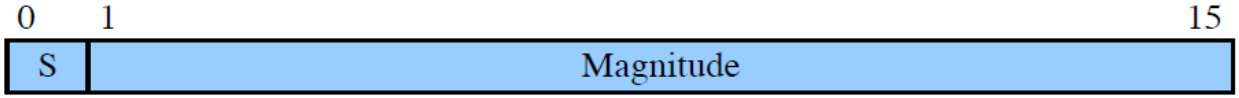
Basic Instruction Cycle

Instruction Register (IR)

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

Characteristics of a Hypothetical Machine

(a) Instruction format



(b) Integer format

Program counter (PC) = Address of instruction
Instruction register (IR) = Instruction being executed
Accumulator (AC) = Temporary storage

(c) Internal CPU registers

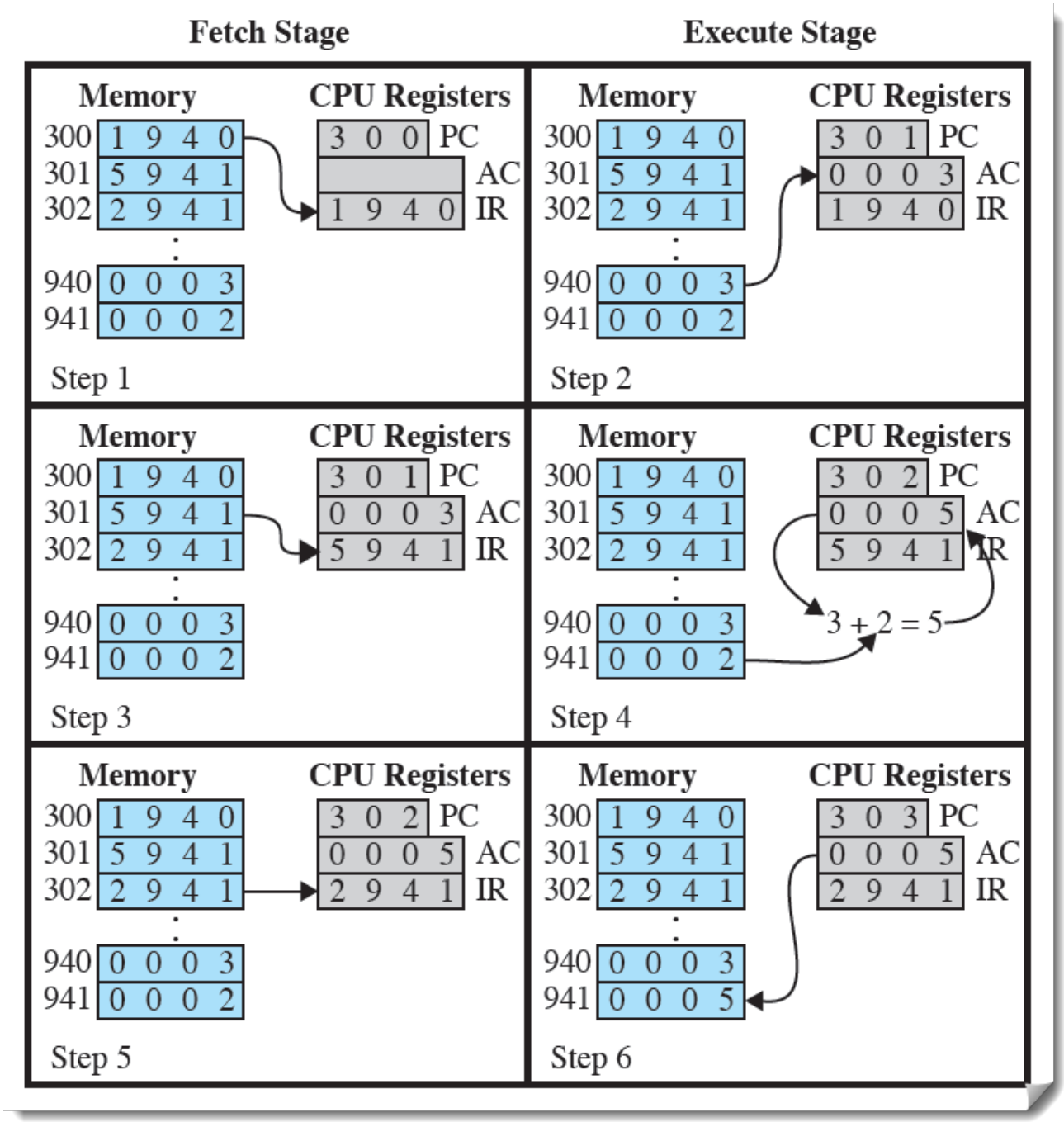
0001 = Load AC from memory
0010 = Store AC to memory
0101 = Add to AC from memory

(d) Partial list of opcodes

load 940

add 941

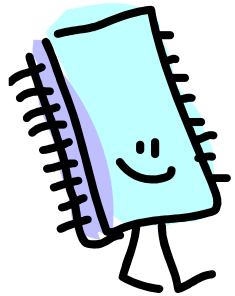
store 941



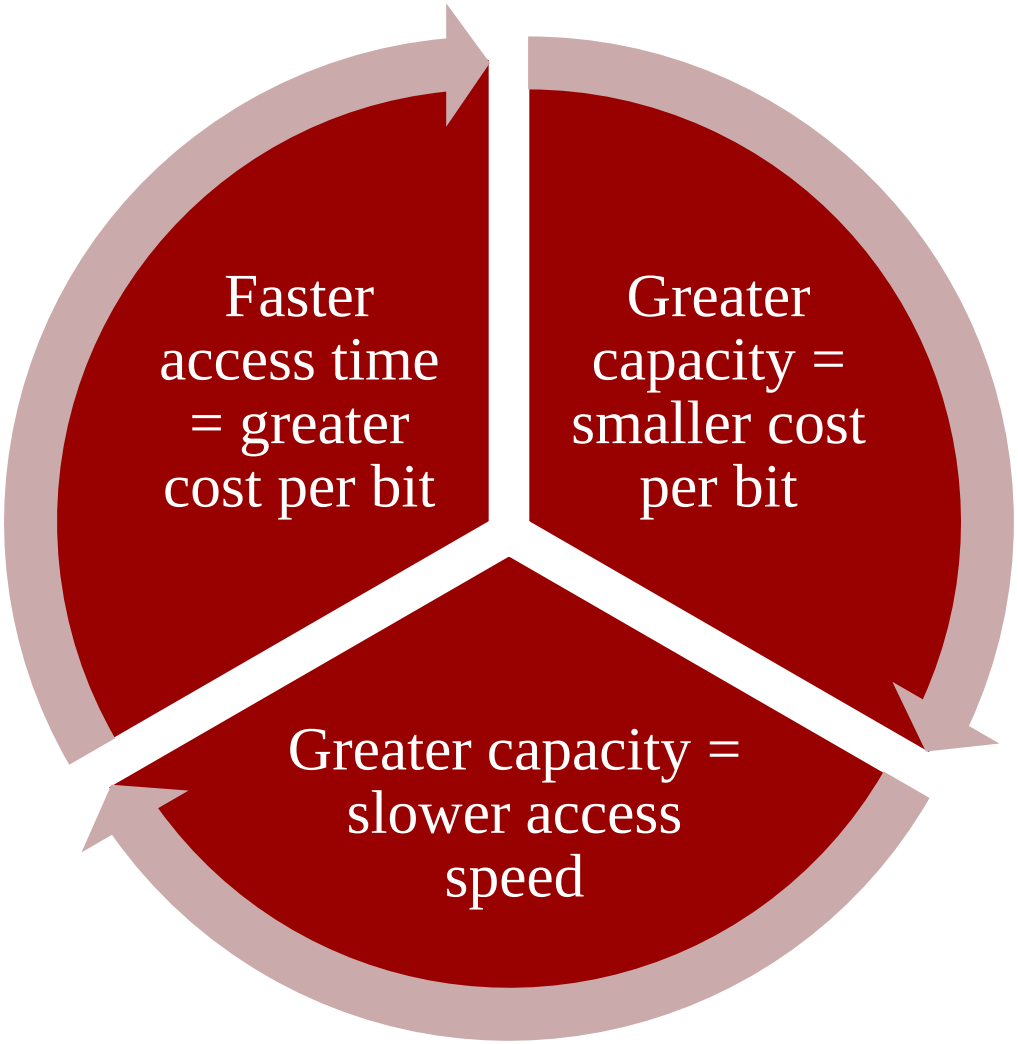
Example of Program Execution

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**

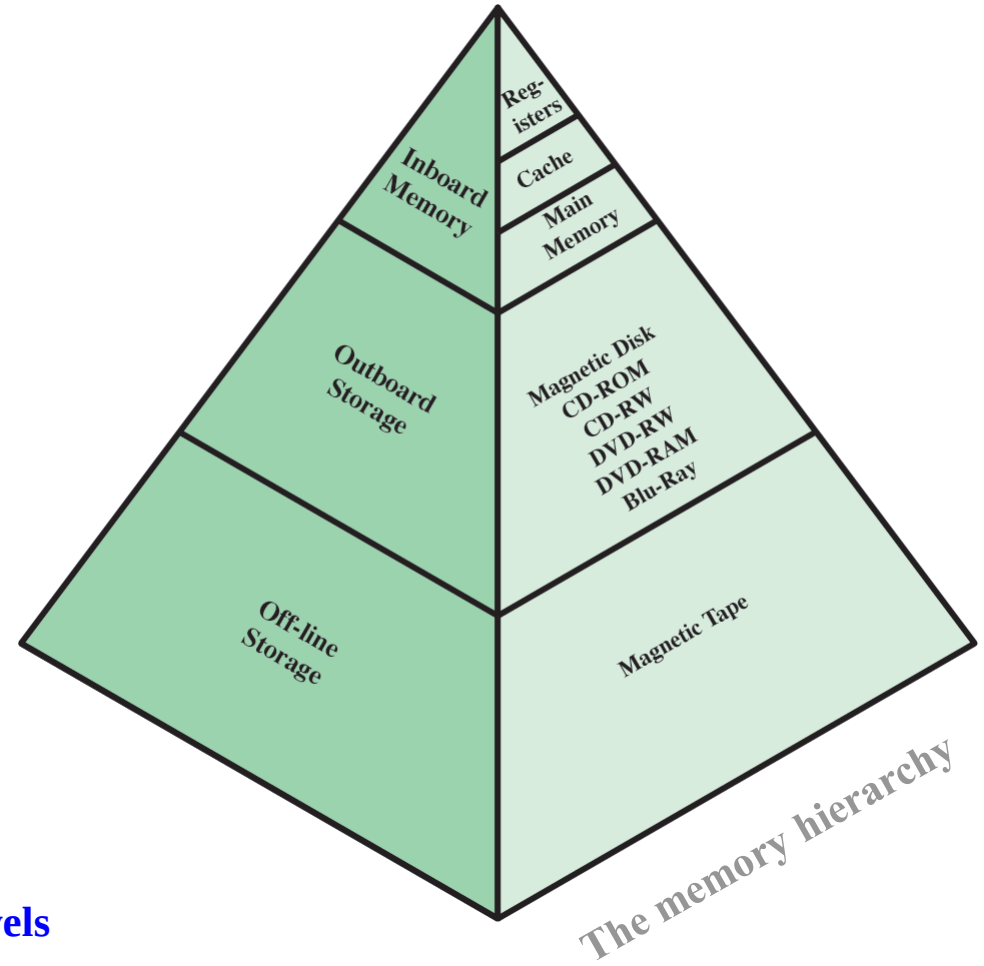


Memory Relationships



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

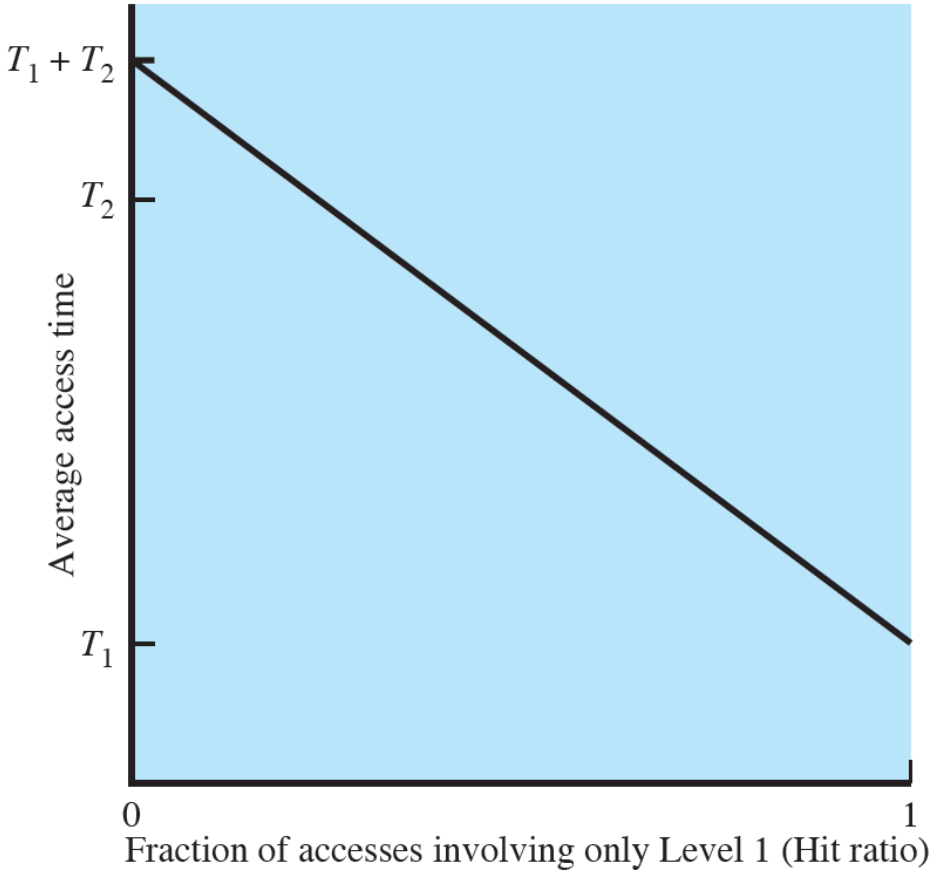


The memory hierarchy

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

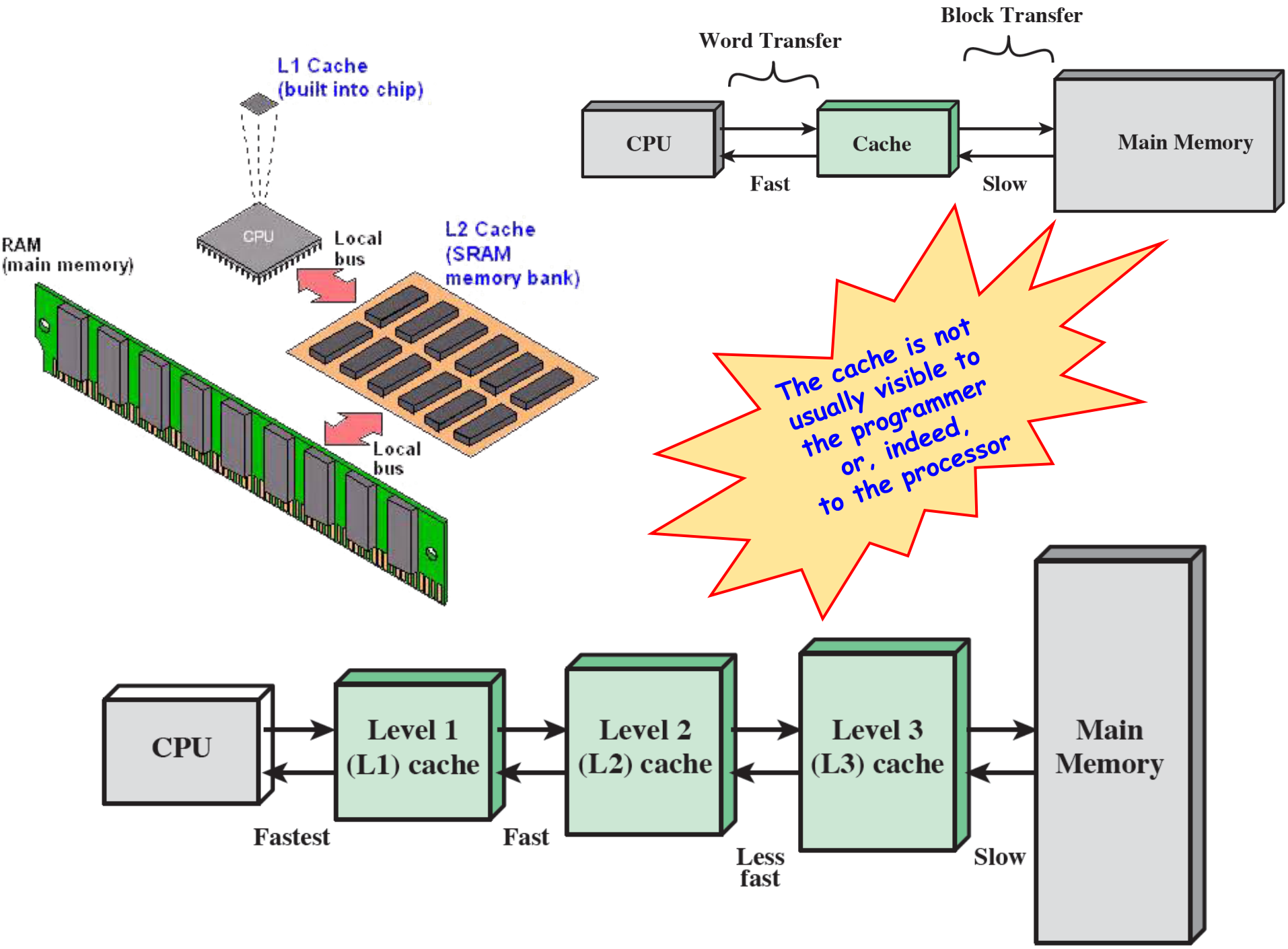
- Memory references by the processor tend to cluster
- Data is organized so that the percentage of accesses to each successively lower level is substantially less than that of the level above
- Can be applied across more than two levels of memory

Secondary Memory

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

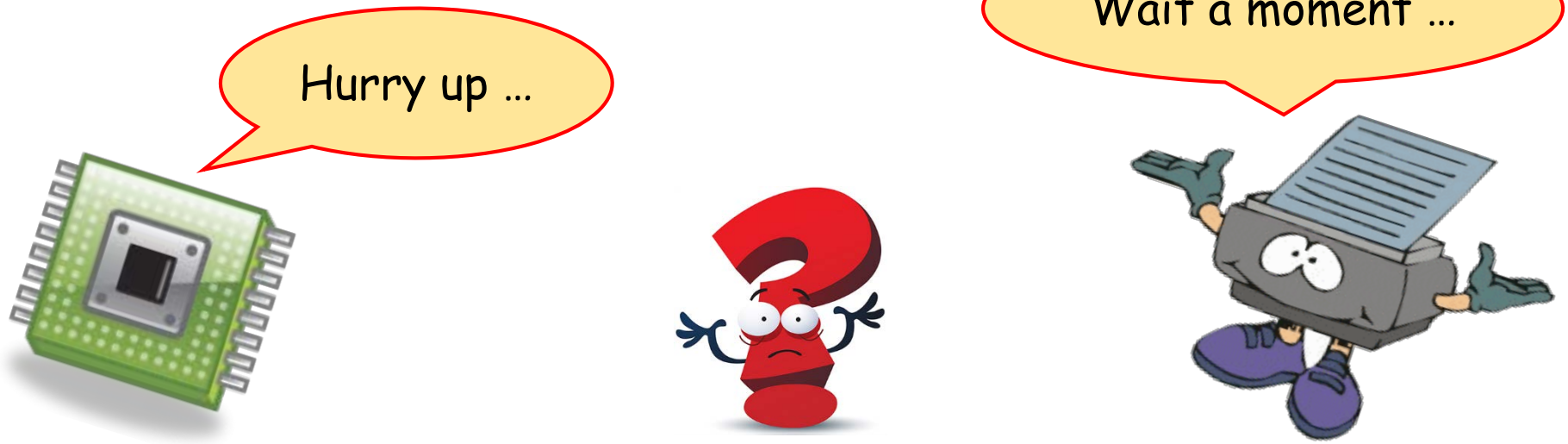
Cache Memory

- **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**


L1 Cache
(built into chip)
CPU
Local
bus
L2 Cache
(SRAM
memory bank)
RAM
(main memory)
Local
bus
Word Transfer
Block Transfer
CPU
Cache
Main Memory
Fast
Slow
The cache is not
usually visible to
the programmer
or, indeed,
to the processor
CPU
Level 1
(L1) cache
Level 2
(L2) cache
Level 3
(L3) cache
Main
Memory
Fastest
Fast
Less
fast
Slow

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



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

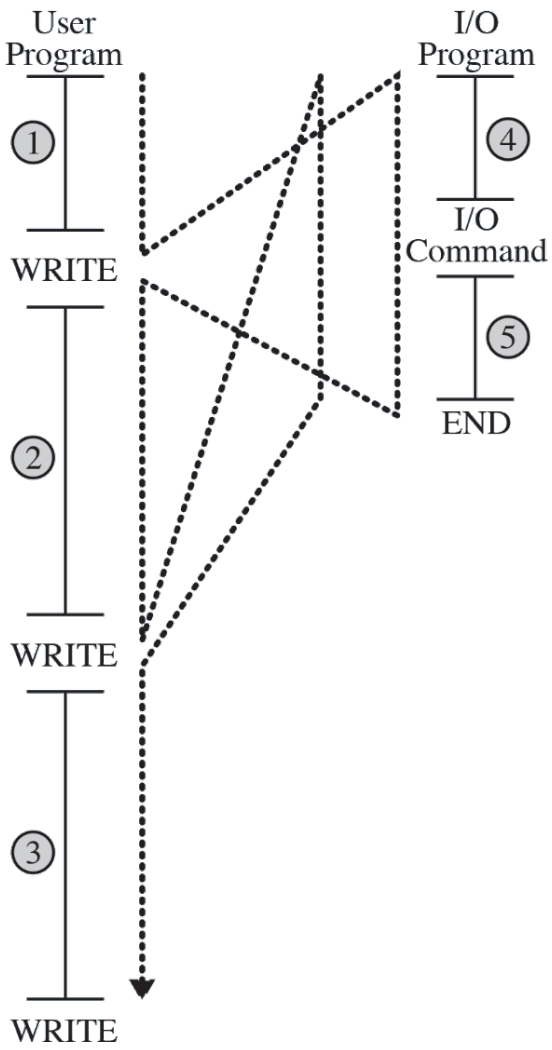
I/O

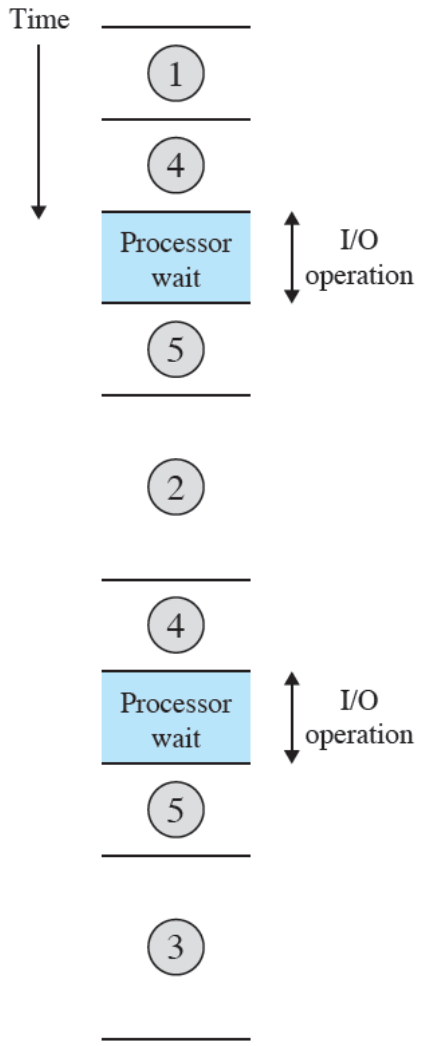
- Generated by an I/O controller, to signal normal completion of an operation or to signal a variety of error conditions

Hardware failure

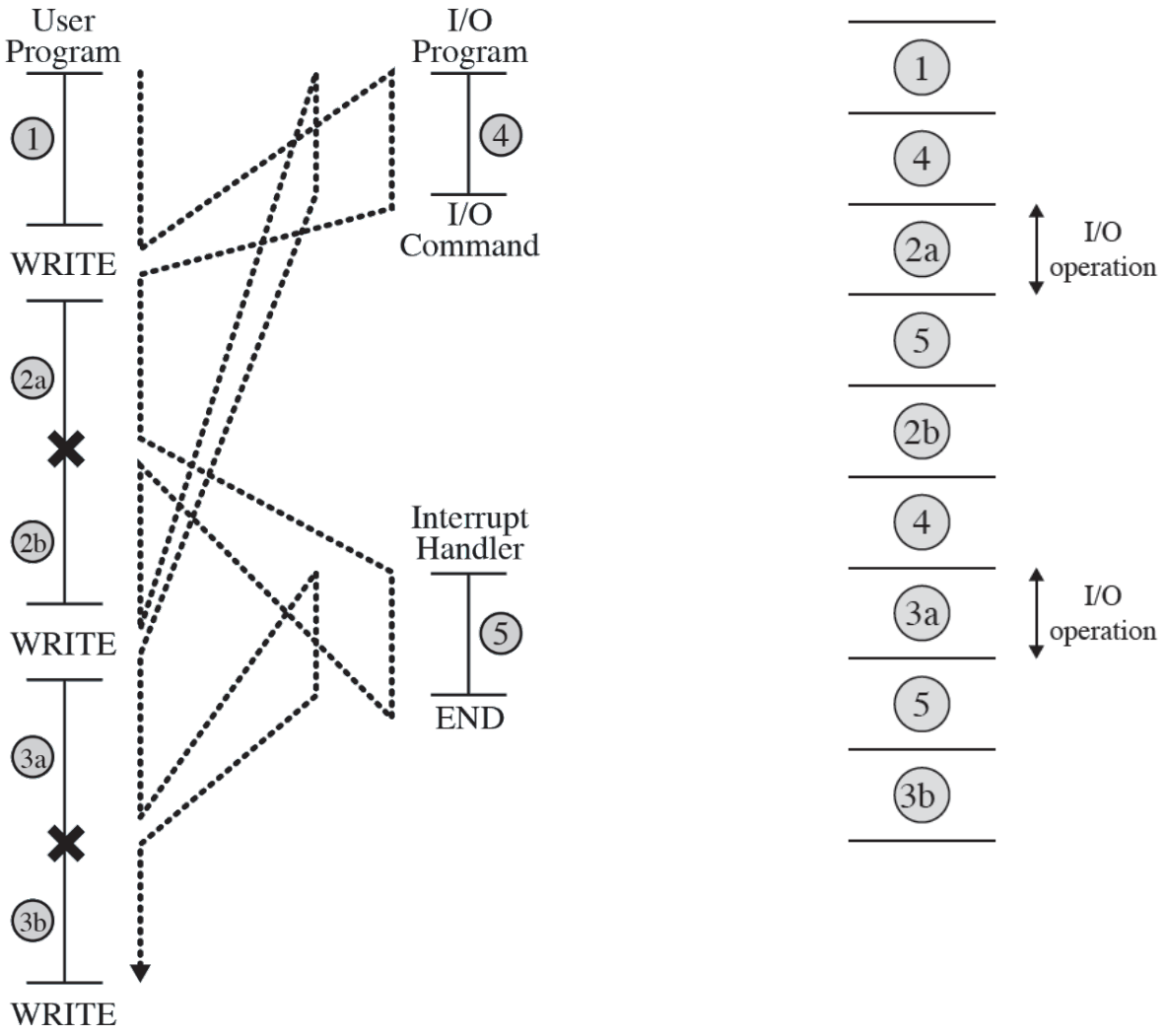
- Generated by a failure, such as power failure or memory parity error

Flow of Control Without Interrupts

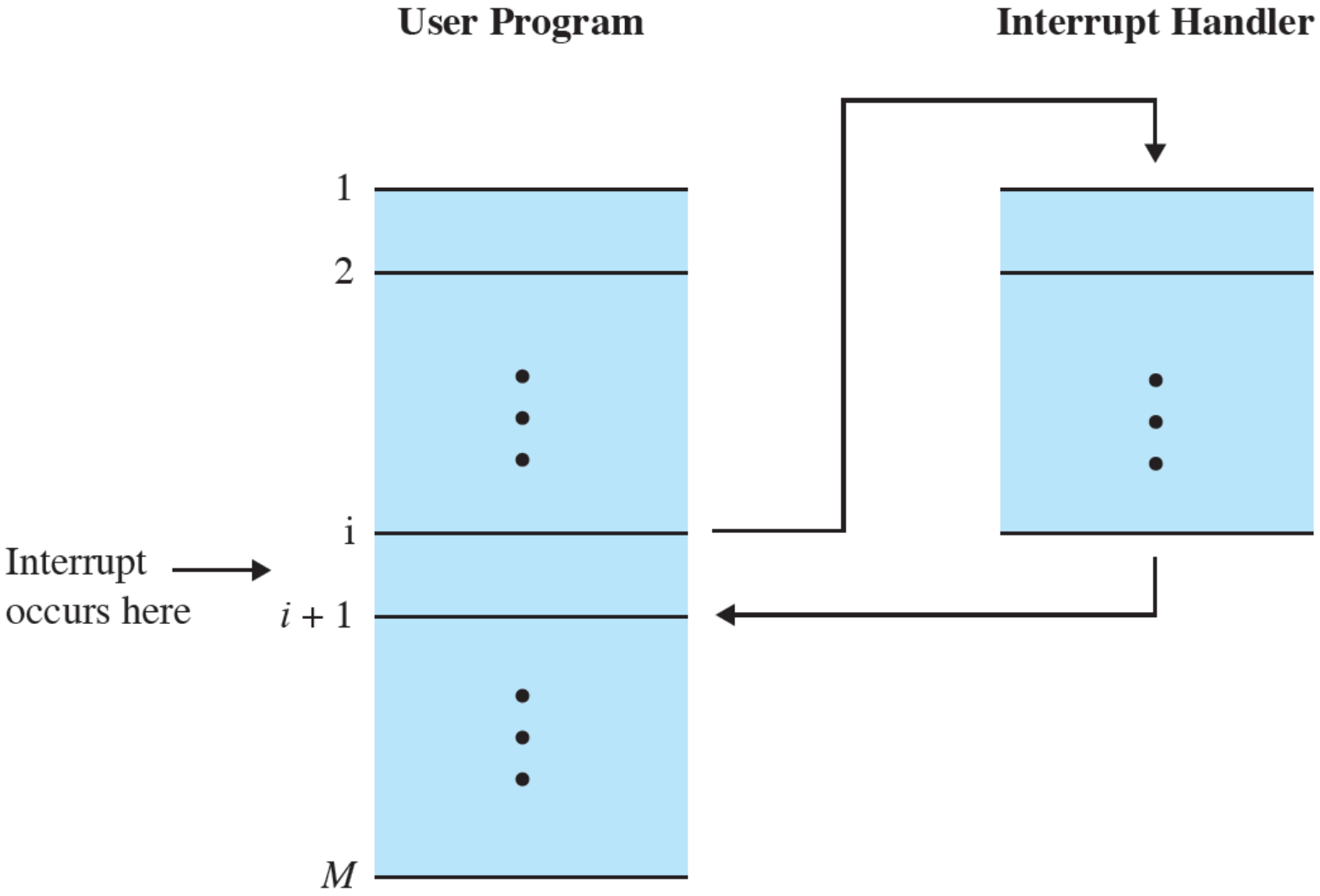




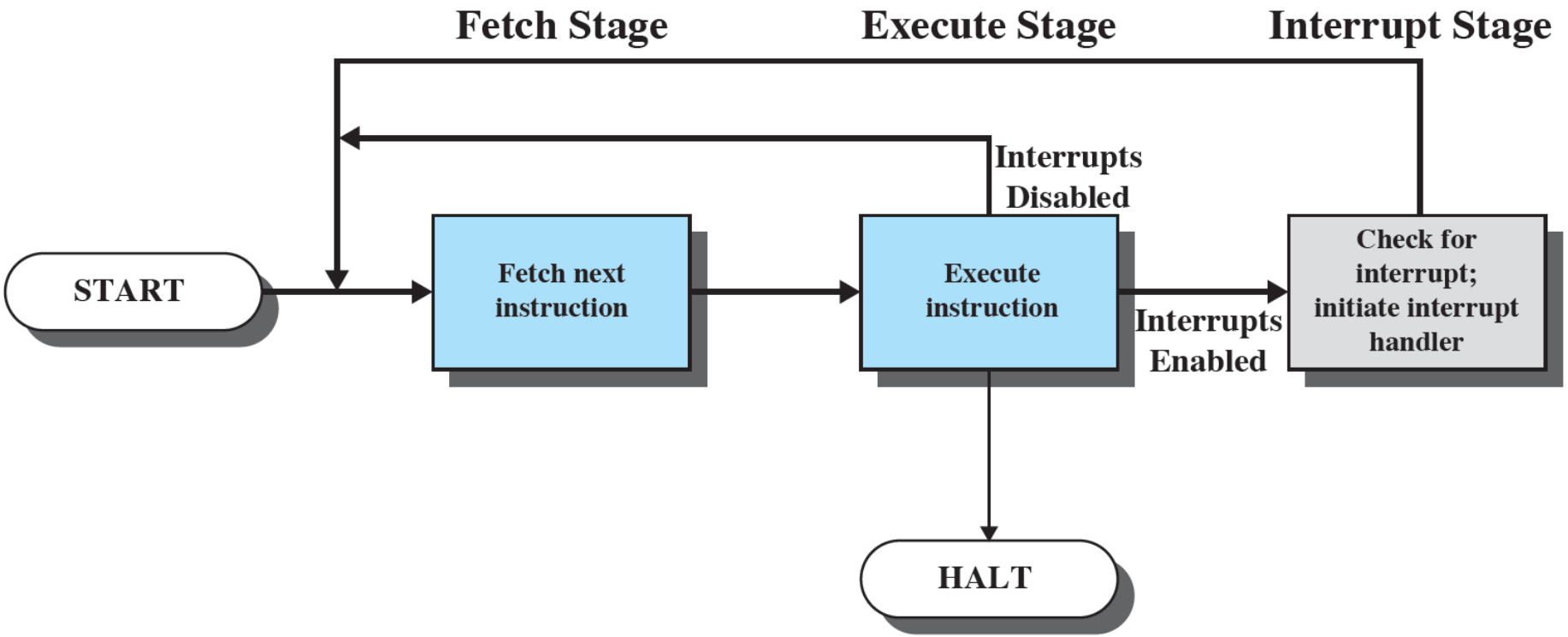
Interrupts: Short I/O Wait



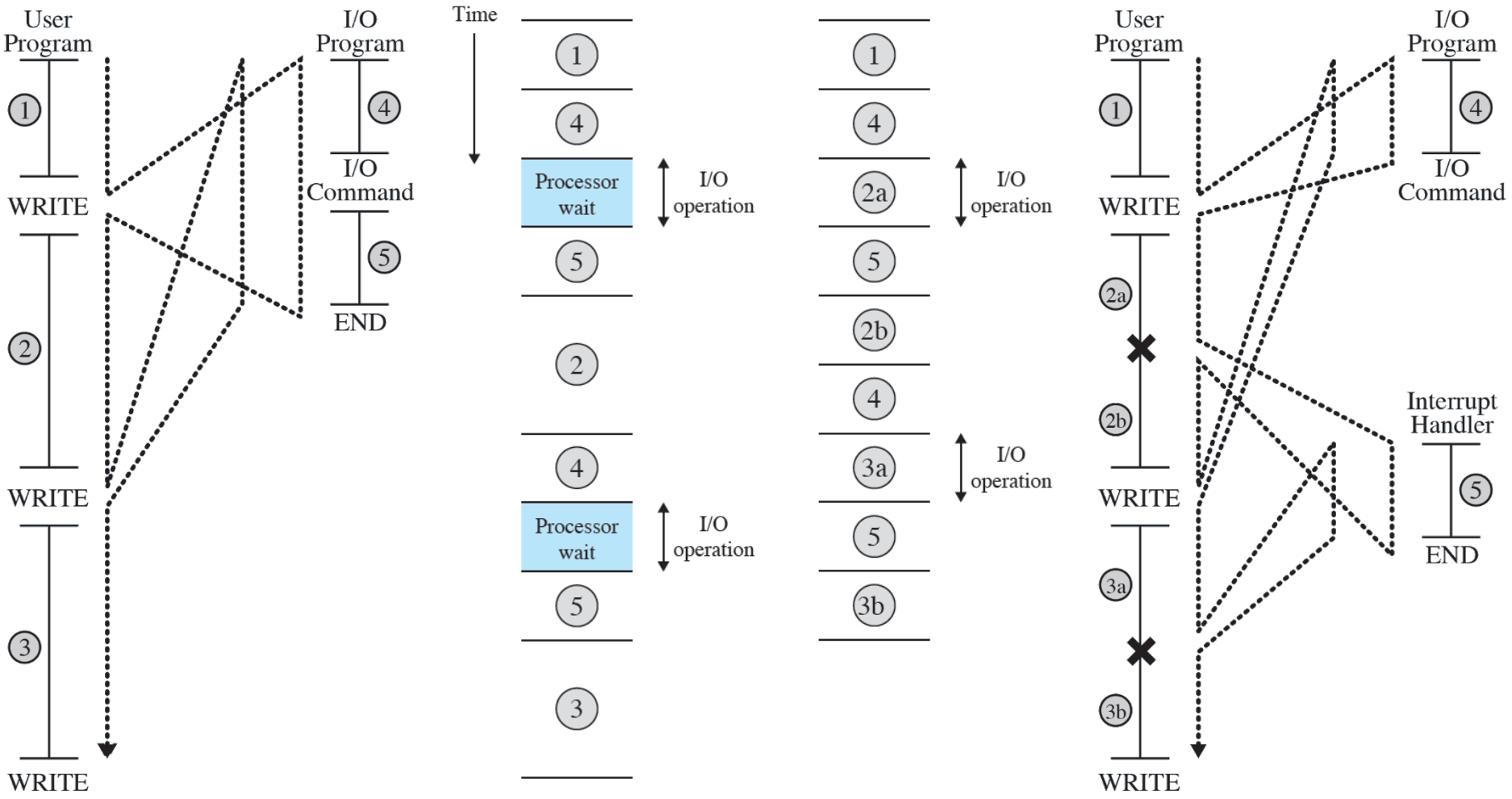
Instruction Cycle With Interrupts



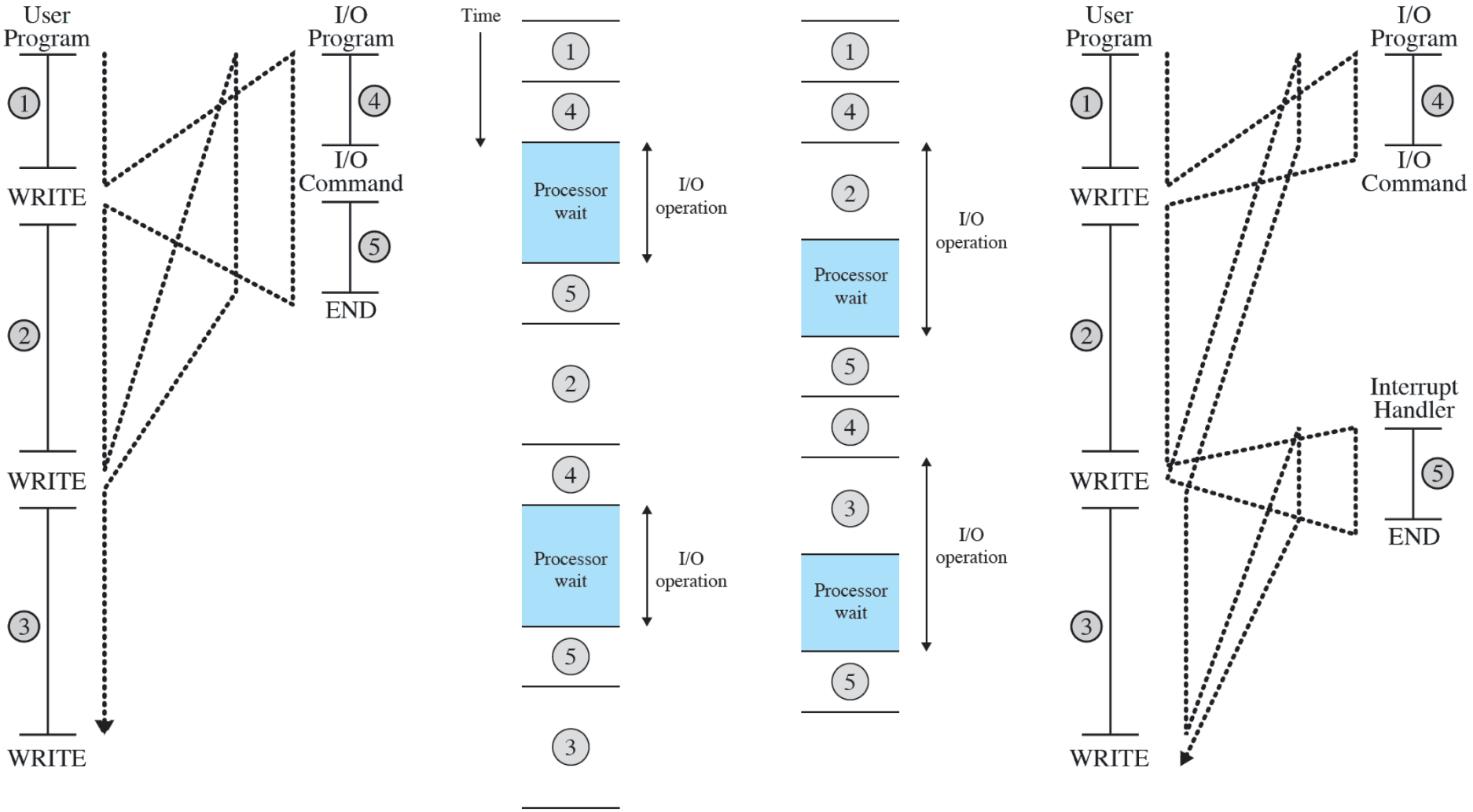
Instruction Cycle With Interrupts

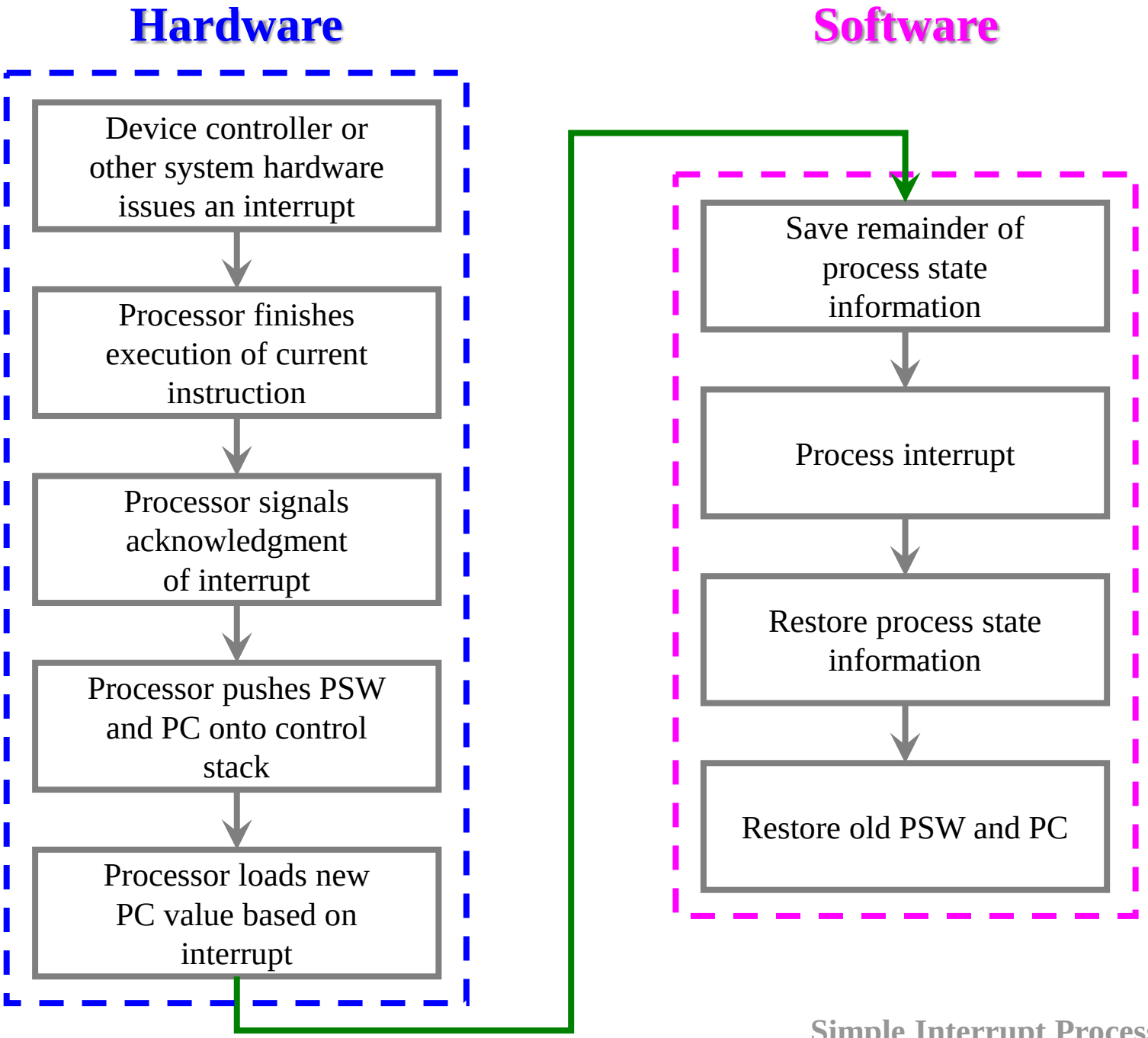


Program Timing: Short I/O Wait



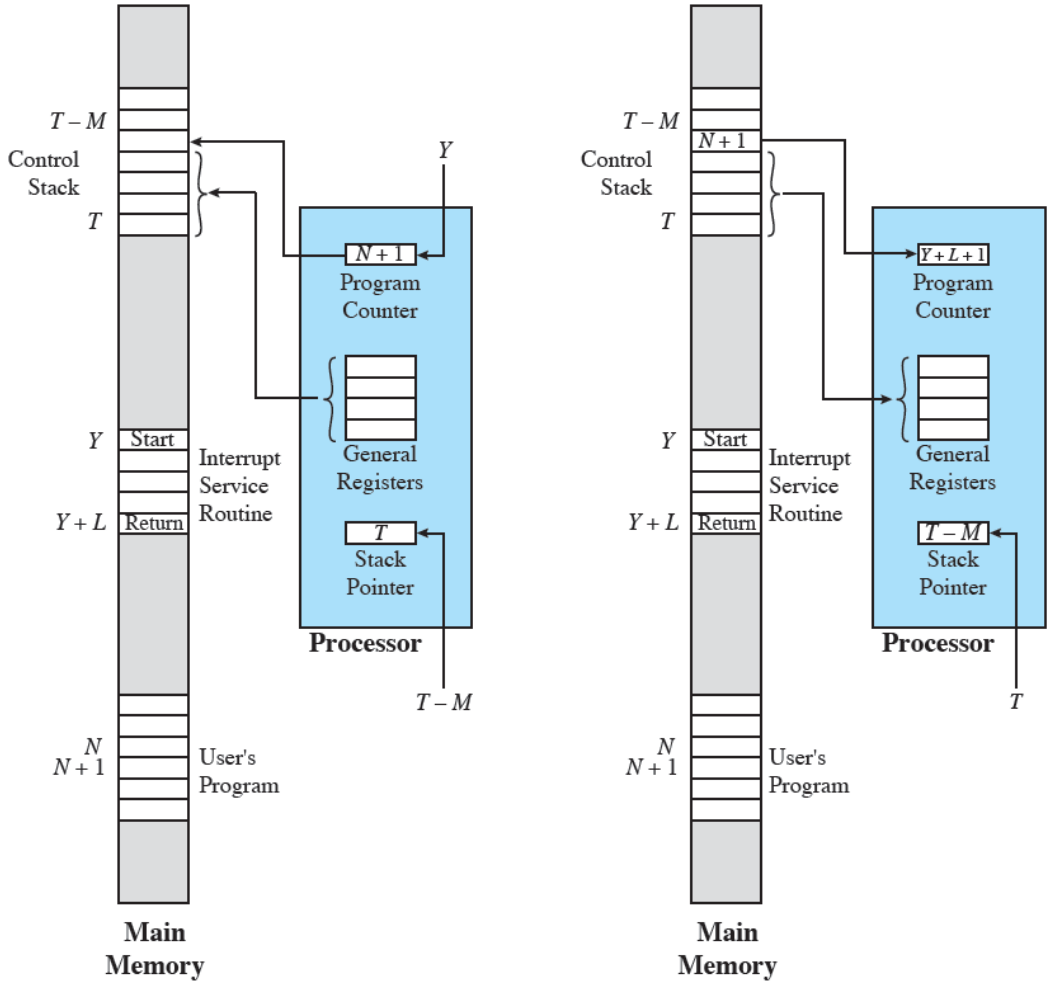
Program Timing: Long I/O wait





Simple Interrupt Processing

Changes for an Interrupt



Multiple Interrupts

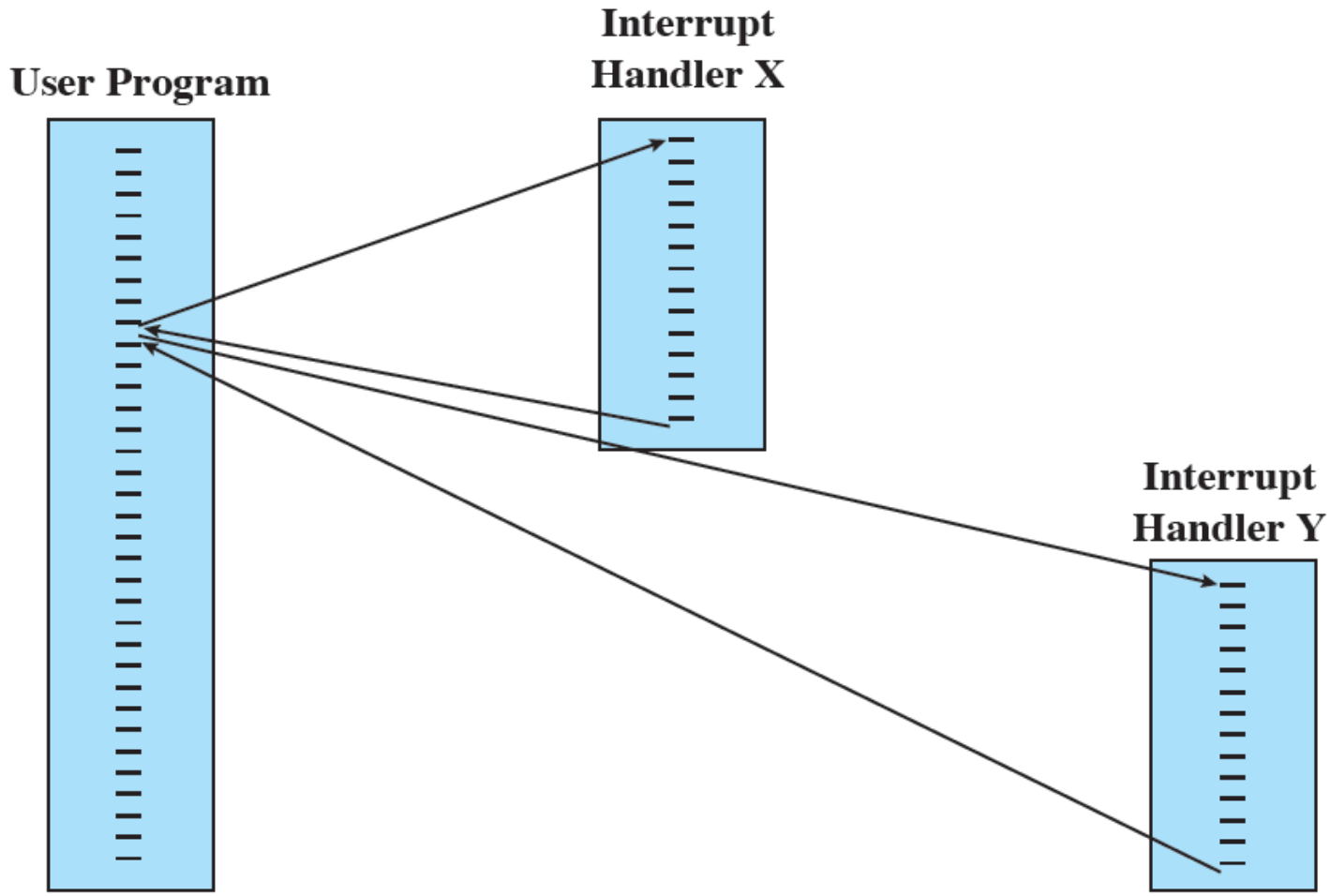
An interrupt occurs while another interrupt is being processed

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

Two approaches:

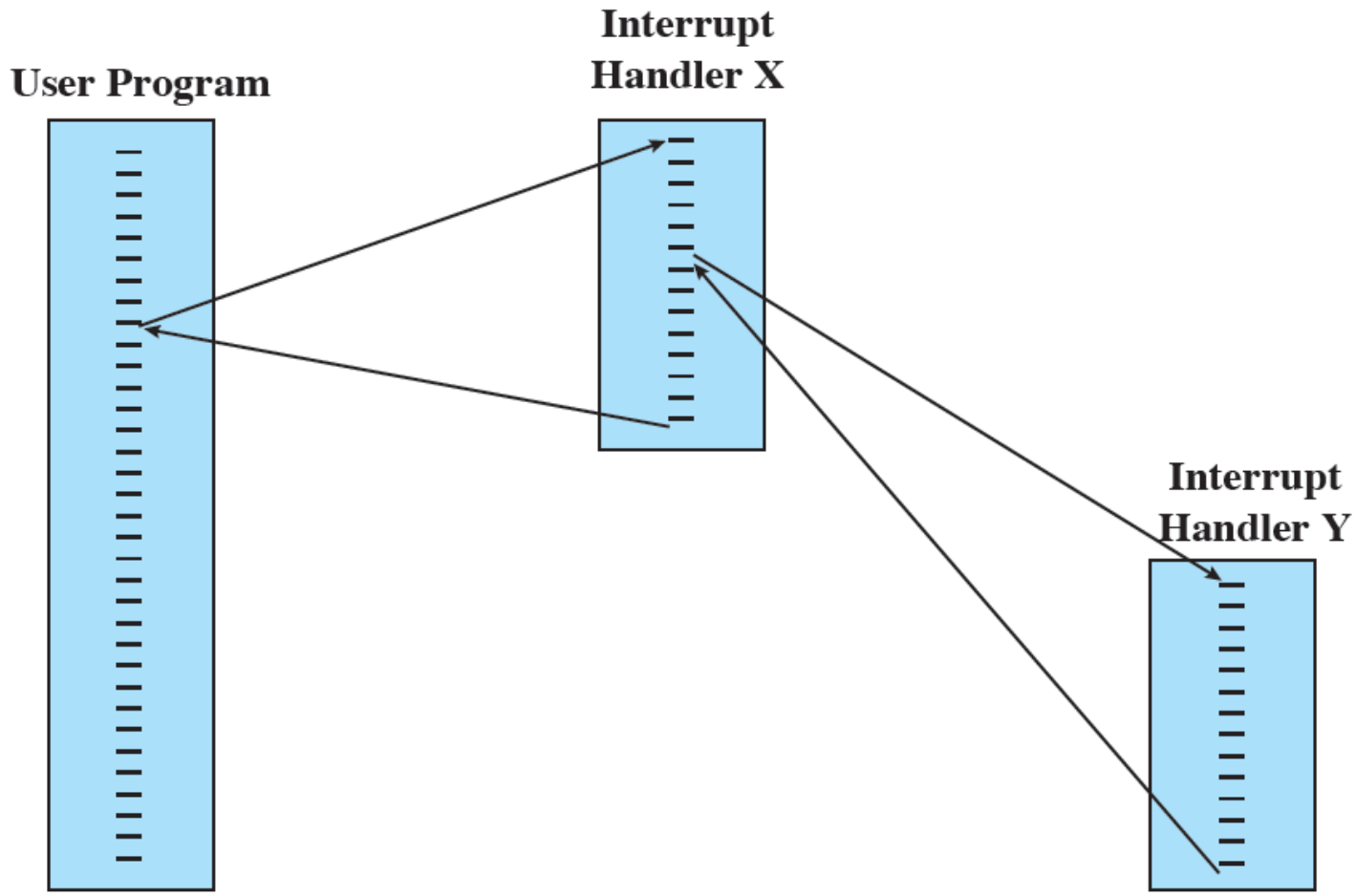
- disable interrupts while an interrupt is being processed
- use a priority scheme

Sequential interrupt processing



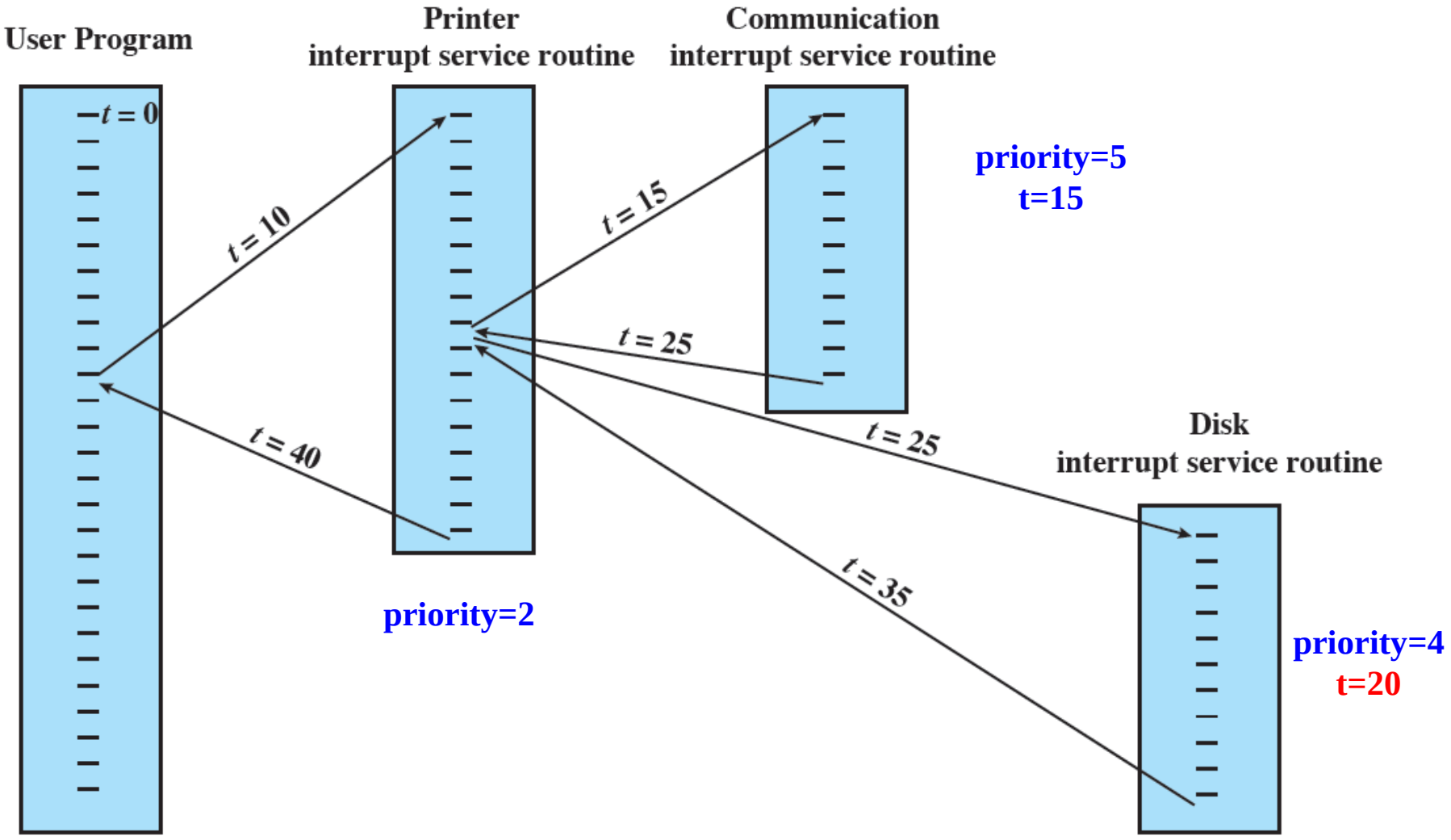
(a) Sequential interrupt processing

Nested interrupt processing



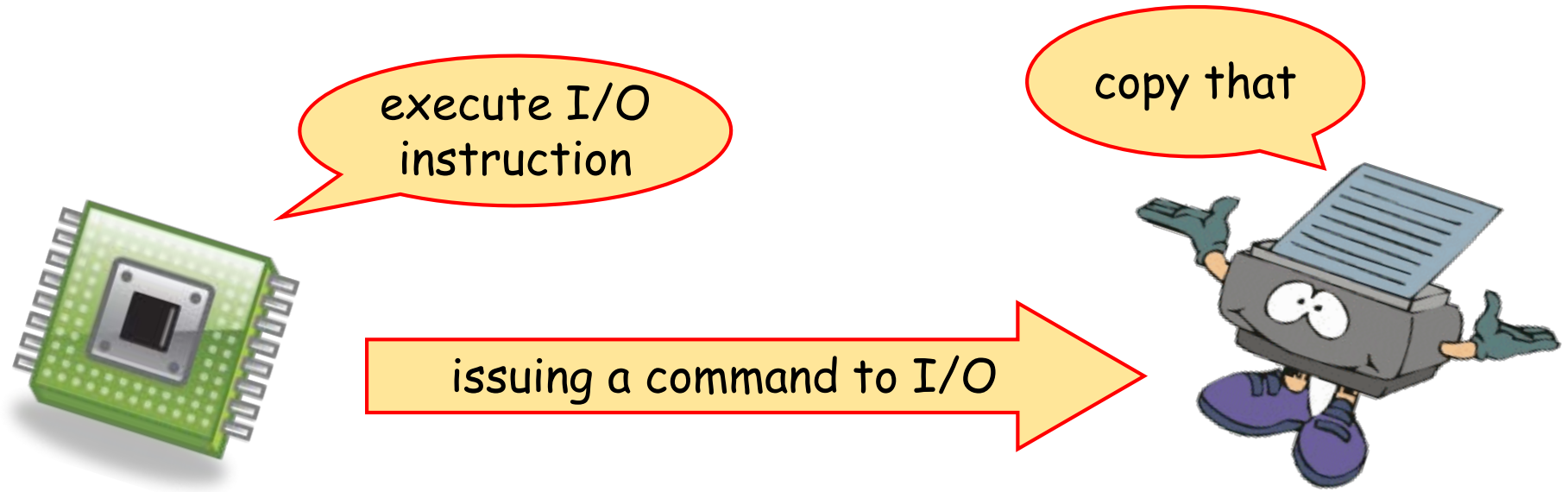
(b) Nested interrupt processing

Example Time Sequence of Multiple Interrupts



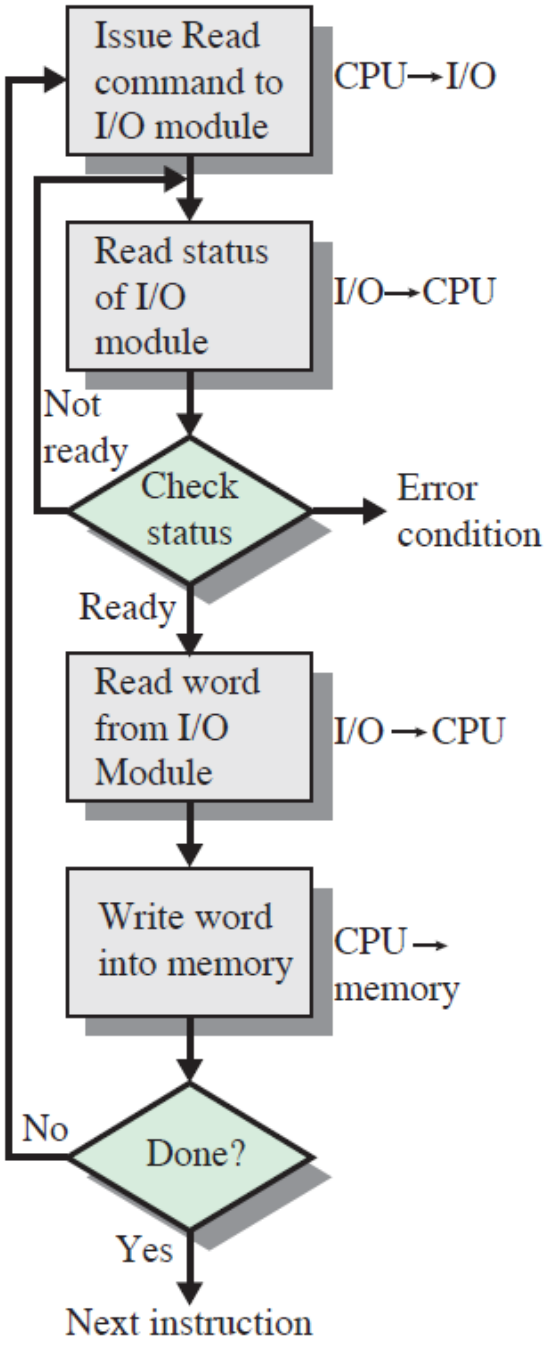
I/O Techniques

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

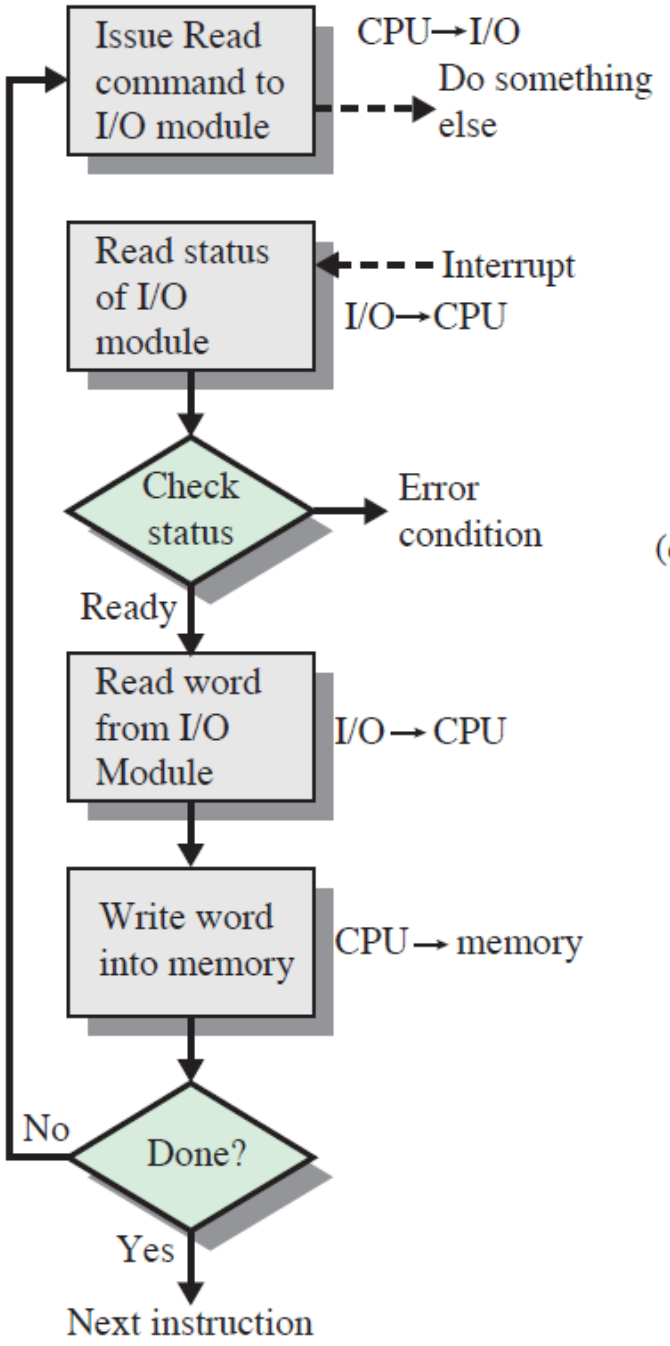


DMA

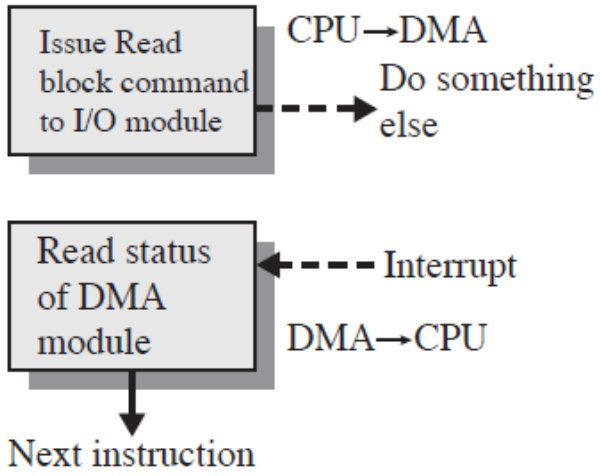
- **It allows certain hardware subsystems to access main system memory independently of CPU**
  - performed by a separate module on the system bus
  - incorporated into an I/O module
- **It issues a command to the DMA module containing**
  - whether a read or write is requested
  - the address of the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



(a) Programmed I/O



(b) Interrupt-driven I/O

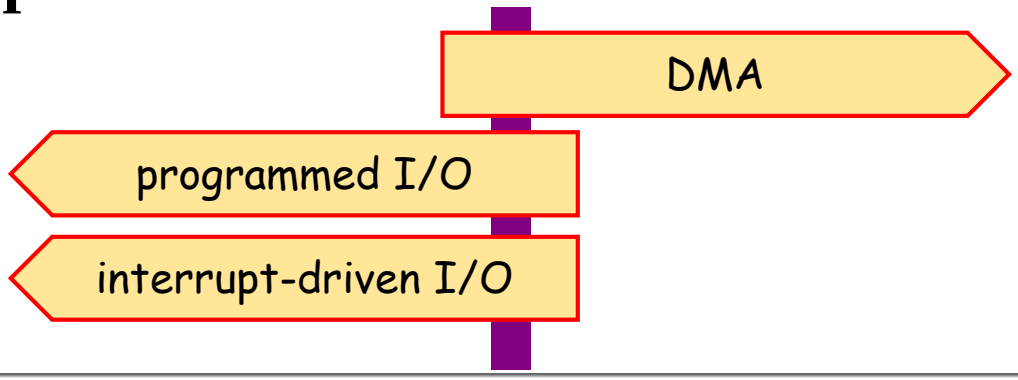


(c) Direct memory access

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



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:

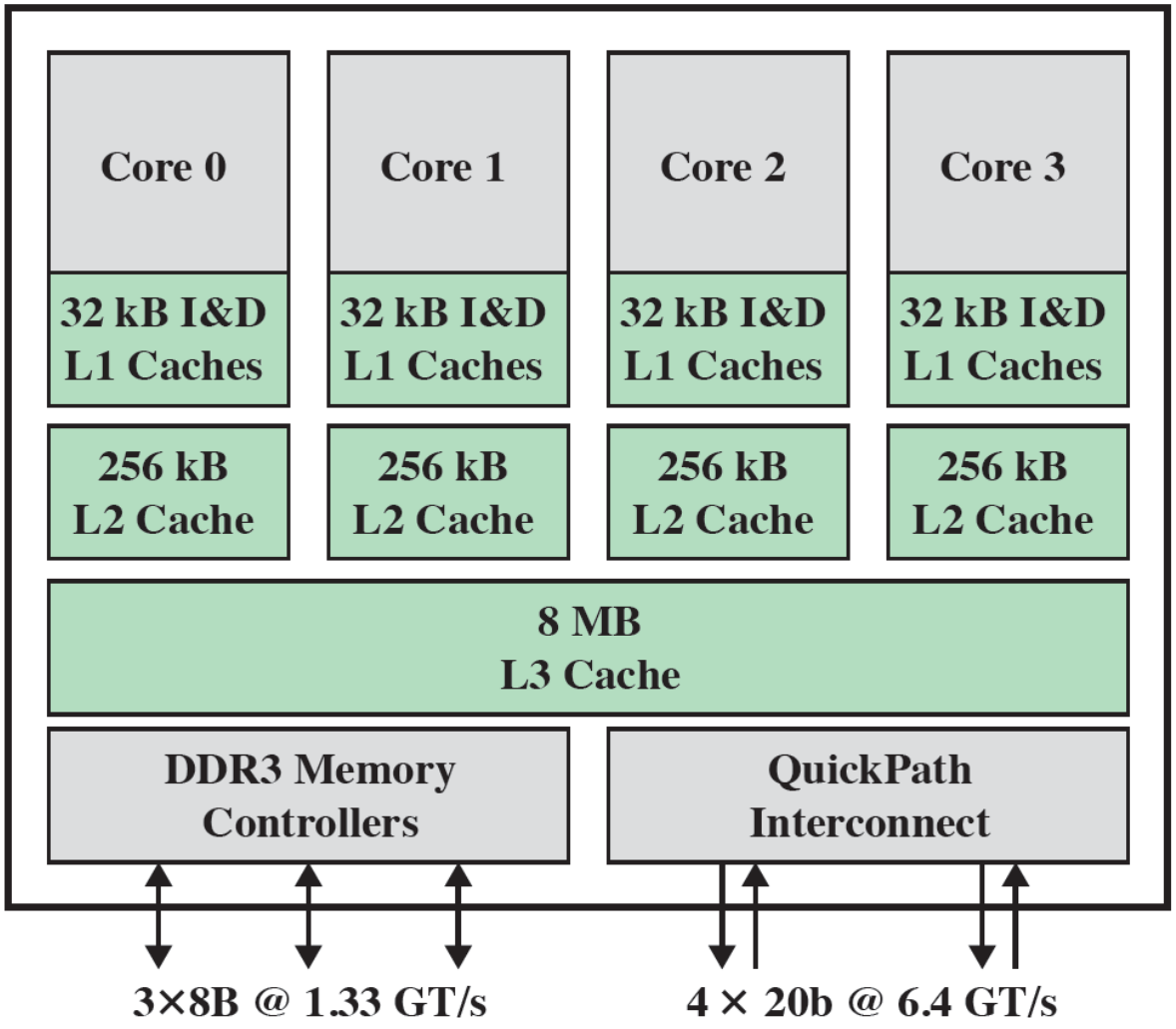
DDR3 Memory Controller

- 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**