+

SEN361 Computer Organization

Prof. Dr. Hasan Hüseyin BALIK
(1st Week)

+ Outline

- Course Information and Policies
- Course Syllabus
- 1. Overview
  - —Indroduction
  - —Computer Evolution and Performance

Course Information

- Instructor: Prof. Dr. Hasan H. BALIK, info@hasanbalik.com, hasanbalik@gmail.com and hasanbalik@aydin.edu.tr www.hasanbalik.com

- Class Homepage:

    http://www.hasanbalik.com/dersler/bmim/LectureNotes/

    http://www.hasanbalik.com/dersform.asp?did=42&ad=Computer%20Organization

    http://www.hasanbalik.com/sinav.asp?did=42&ad=Computer%20Organization

    Book: Computer Organization and Architecture, Ninth Edition, William Stallings, Prentice Hall 2013

- Grading: Midterm 30%, Short Exam (2) 10%,

    Assingment (Project) 20% and Final 40%

Course Syllabus-1

- Overview
  - Introduction
  - Computer Evolution and Performance

- The computer system
  - A Top-Level View of Computer Function and Interconnection
  - Cache Memory
  - Internal Memory Technology
  - External Memory
  - Input/Output

- Midterm Exam

Course Syllabus-2

- The central processing unit
  - Instruction Sets: Characteristics and Functions
  - Instruction Sets: Addressing Modes and Formats
  - Processor Structure and Function
  - Reduced Instruction Set Computers (RISCs)
  - Instruction-Level Parallelism and Superscalar Processors

- Parallel organization
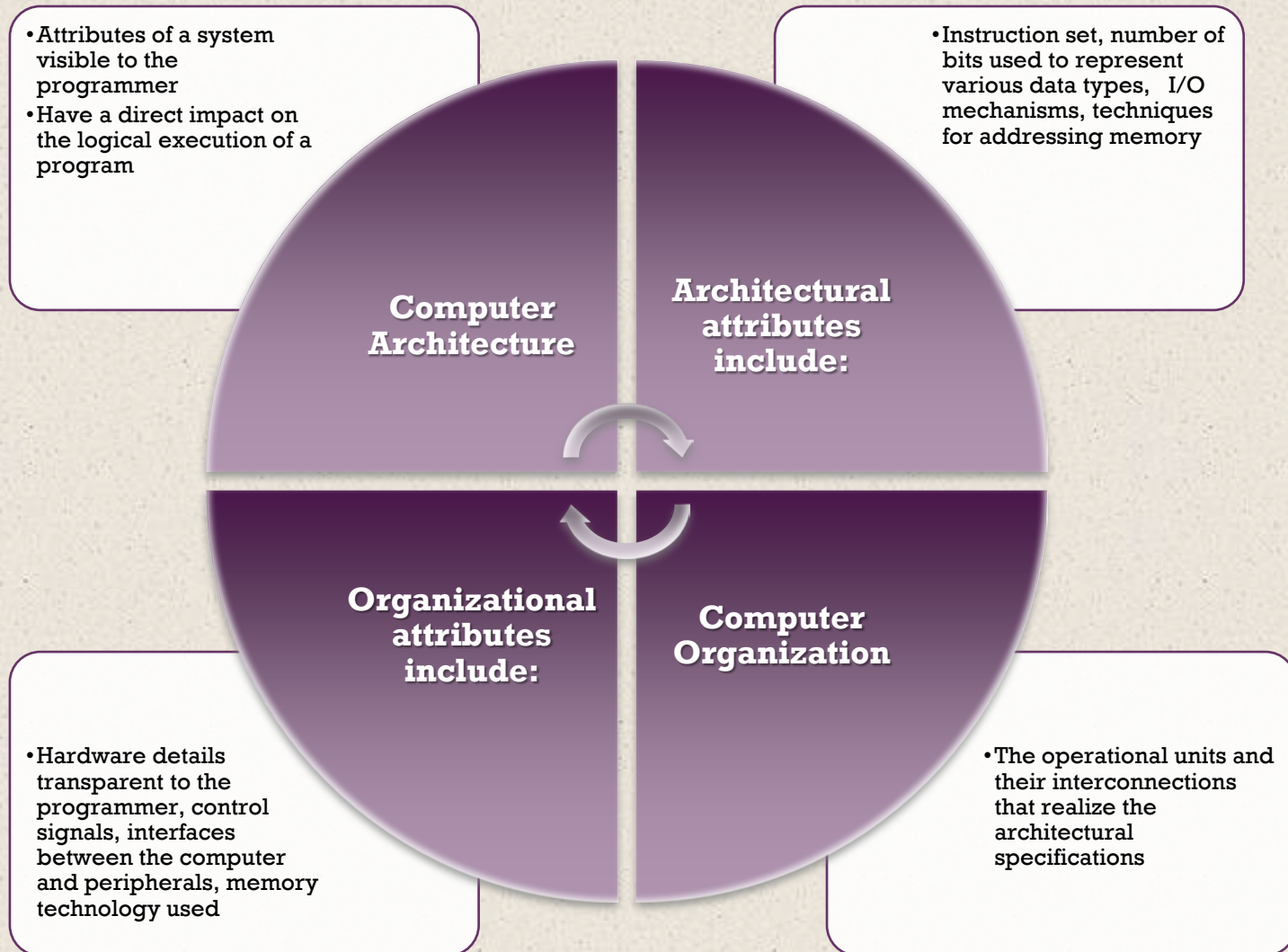 - Parallel Processing
  - Multicore Computers

+

Outline

1.1 Introduction

+

1.1 Outline

- Organization and Architecture

- Structure and Function

Computer Architecture

Computer Organization

- Attributes of a system visible to the programmer
- Have a direct impact on the logical execution of a program

- Instruction set, number of bits used to represent various data types, I/O mechanisms, techniques for addressing memory

Computer Architecture

Architectural attributes include:

Organizational attributes include:

Computer Organization

- Hardware details transparent to the programmer, control signals, interfaces between the computer and peripherals, memory technology used

- The operational units and their interconnections that realize the architectural specifications

IBM System 370 Architecture

- IBM System/370 architecture
  - Was introduced in 1970
  - Included a number of models
  - Could upgrade to a more expensive, faster model without having to abandon original software
  - New models are introduced with improved technology, but retain the same architecture so that the customer's software investment is protected
  - Architecture has survived to this day as the architecture of IBM's mainframe product line

Structure and Function

- Hierarchical system
  - Set of interrelated subsystems

- Hierarchical nature of complex systems is essential to both their design and their description

- Designer need only deal with a particular level of the system at a time
  - Concerned with structure and function at each level

- Structure
  - The way in which components relate to each other

- Function
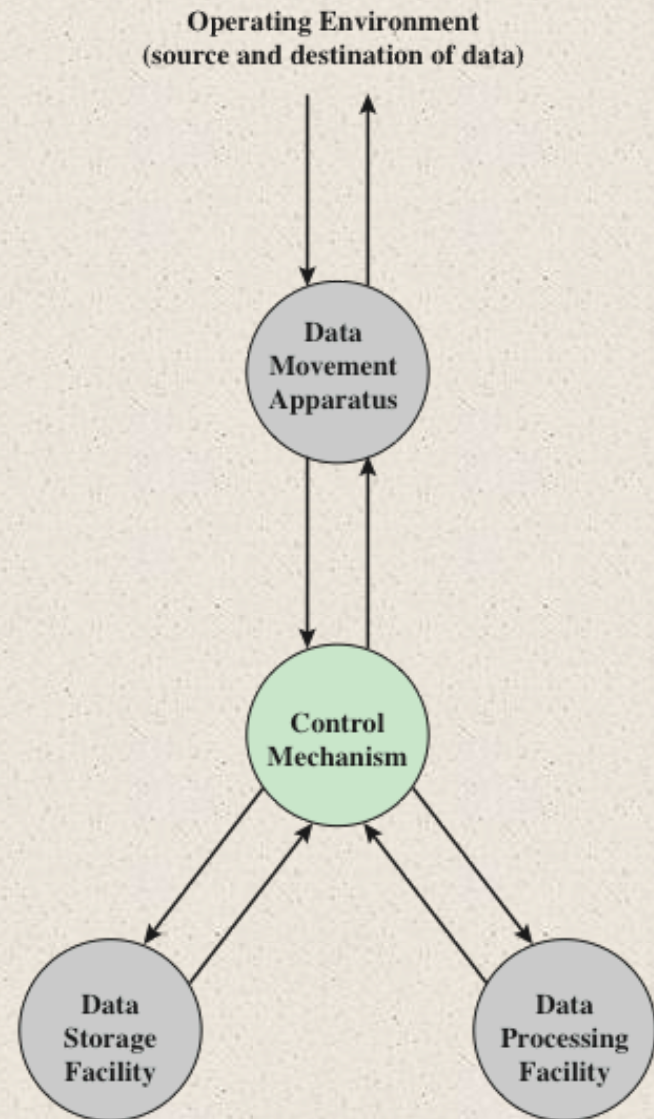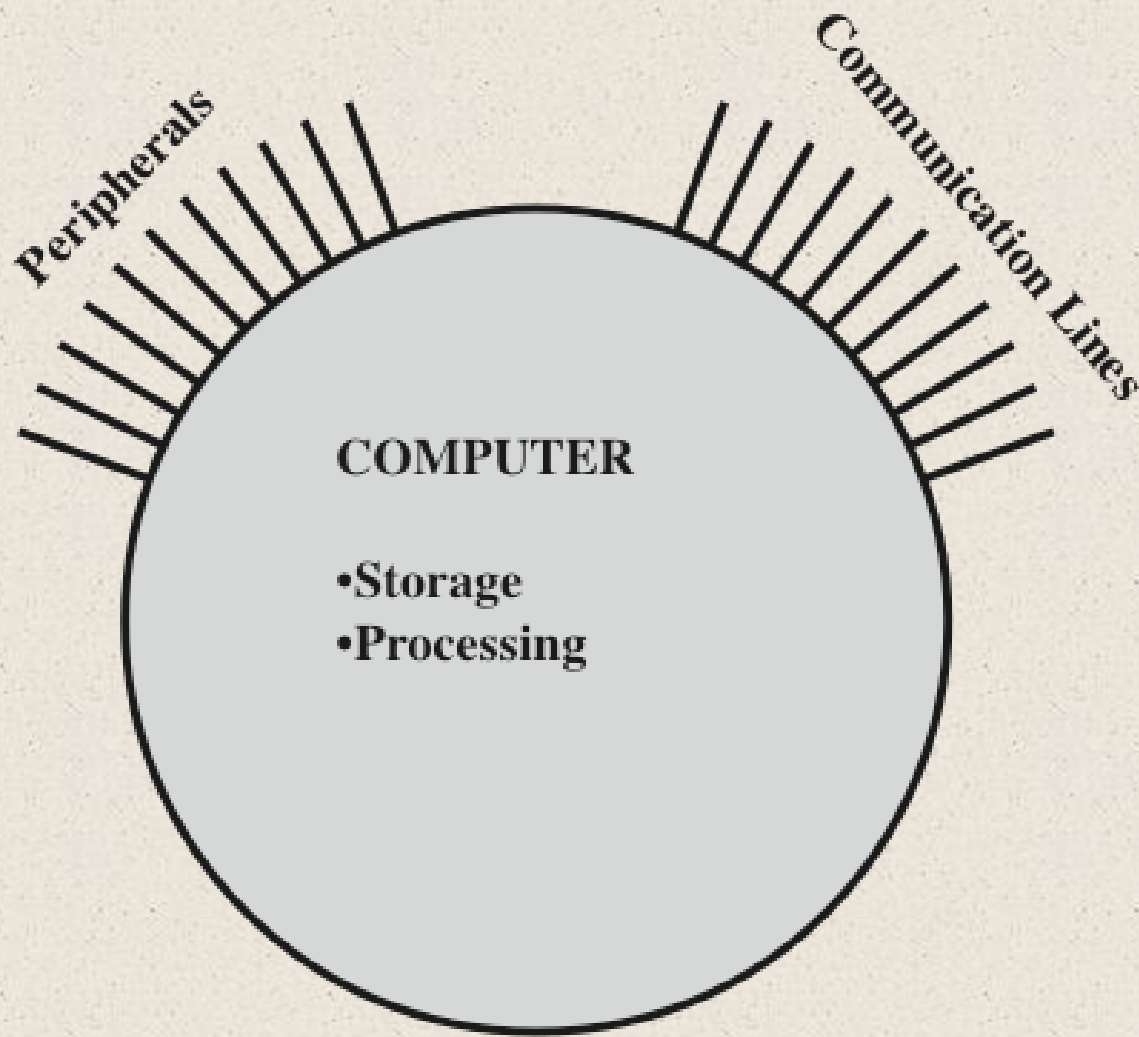 - The operation of individual components as part of the structure

Function

A computer can perform four basic functions:

- Data processing
- Data storage
- Data movement
- Control

Figure 1.1 A Functional View of the Computer
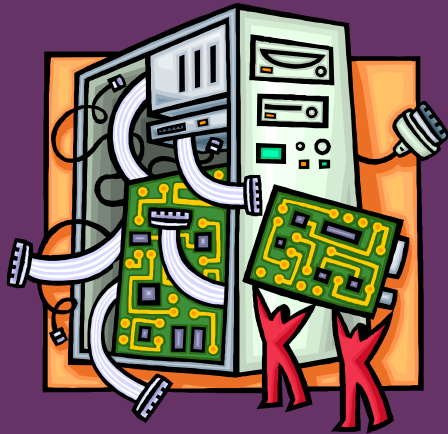
Figure 1.3 The Computer

The Computer

There are four
main structural
components
of the computer:

✦ CPU – controls the operation of the computer and performs its data processing functions

✦ Main Memory – stores data

✦ I/O – moves data between the computer and its external environment

✦ System Interconnection – some mechanism that provides for communication among CPU, main memory, and I/O

CPU

Major structural components:
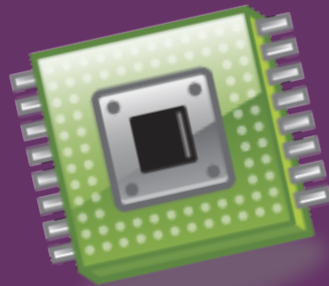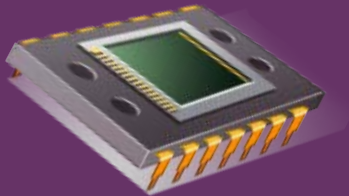
- Control Unit
  - Controls the operation of the CPU and hence the computer

- Arithmetic and Logic Unit (ALU)
  - Performs the computer's data processing function

- Registers
  - Provide storage internal to the CPU

- CPU Interconnection
  - Some mechanism that provides for communication among the control unit, ALU, and registers
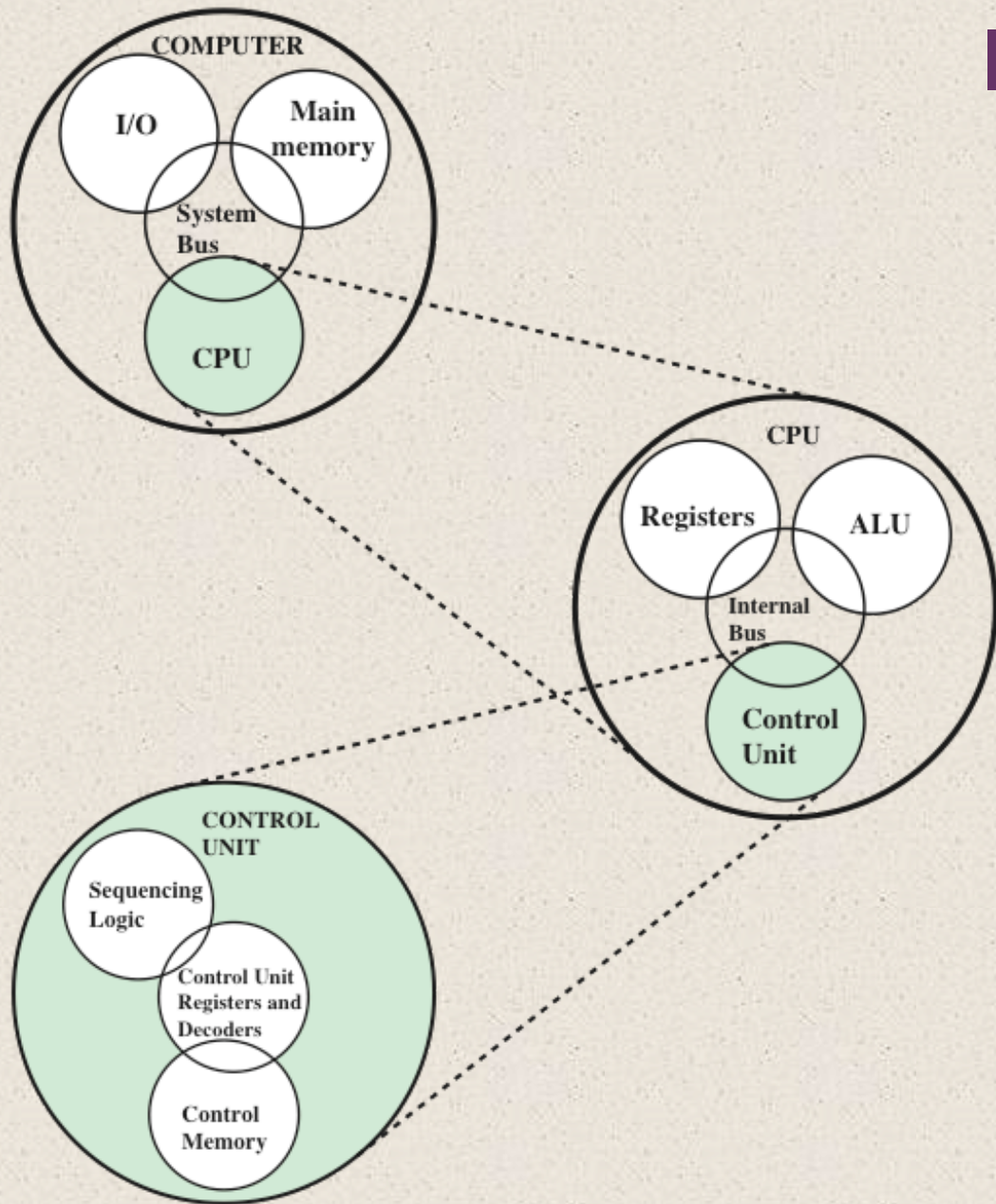
Structure

Figure 1.4 A Top-Down View of a Computer

1.2 Computer Evolution and Performance

+

1.2 Outline

- A Brief History of Computers

- Designing for Performance

- Multicore, MIC (Many Integrated Core)s, and GPGPU (General-purpose computing on GPU) s

- The Evolution of the Intel x86 Architecture

- Embedded Systems and the Arm

- Performance Assessment

Computer Generations

Generation	Approximate Dates	Technology	Typical Speed (operations per second)
1	1946–1957	Vacuum tube	40,000
2	1958–1964	Transistor	200,000
3	1965–1971	Small and medium scale integration	1,000,000
4	1972–1977	Large scale integration	10,000,000
5	1978–1991	Very large scale integration	100,000,000
6	1991-	Ultra large scale integration	1,000,000,000

History of Computers
First Generation: Vacuum Tubes

- ENIAC
  - Electronic Numerical Integrator And Computer
- Designed and constructed at the University of Pennsylvania
  - Started in 1943 – completed in 1946
  - By John Mauchly and John Eckert

- World's first general purpose electronic digital computer
  - Army's Ballistics Research Laboratory (BRL) needed a way to supply trajectory tables for new weapons accurately and within a reasonable time frame
  - Was not finished in time to be used in the war effort

- Its first task was to perform a series of calculations that were used to help determine the feasibility of the hydrogen bomb

- Continued to operate under BRL management until 1955 when it was disassembled

ENIAC

Weighed 30 tons

Occupied 1500 square feet of floor space

Contained more than 18,000 vacuum tubes

140 kW Power consumption

Capable of 5000 additions per second

Decimal rather than binary machine

Memory consisted of 20 accumulators, each capable of holding a 10 digit number

Major drawback was the need for manual programming by setting switches and plugging/ unplugging cables

John von Neumann

EDVAC (Electronic Discrete Variable Computer)

- First publication of the idea was in 1945

- Stored program concept
  - Attributed to ENIAC designers, most notably the mathematician John von Neumann
  - Program represented in a form suitable for storing in memory alongside the data

- IAS computer
  - Princeton Institute for Advanced Studies
  - Prototype of all subsequent general-purpose computers
  - Completed in 1952

Structure of von Neumann Machine
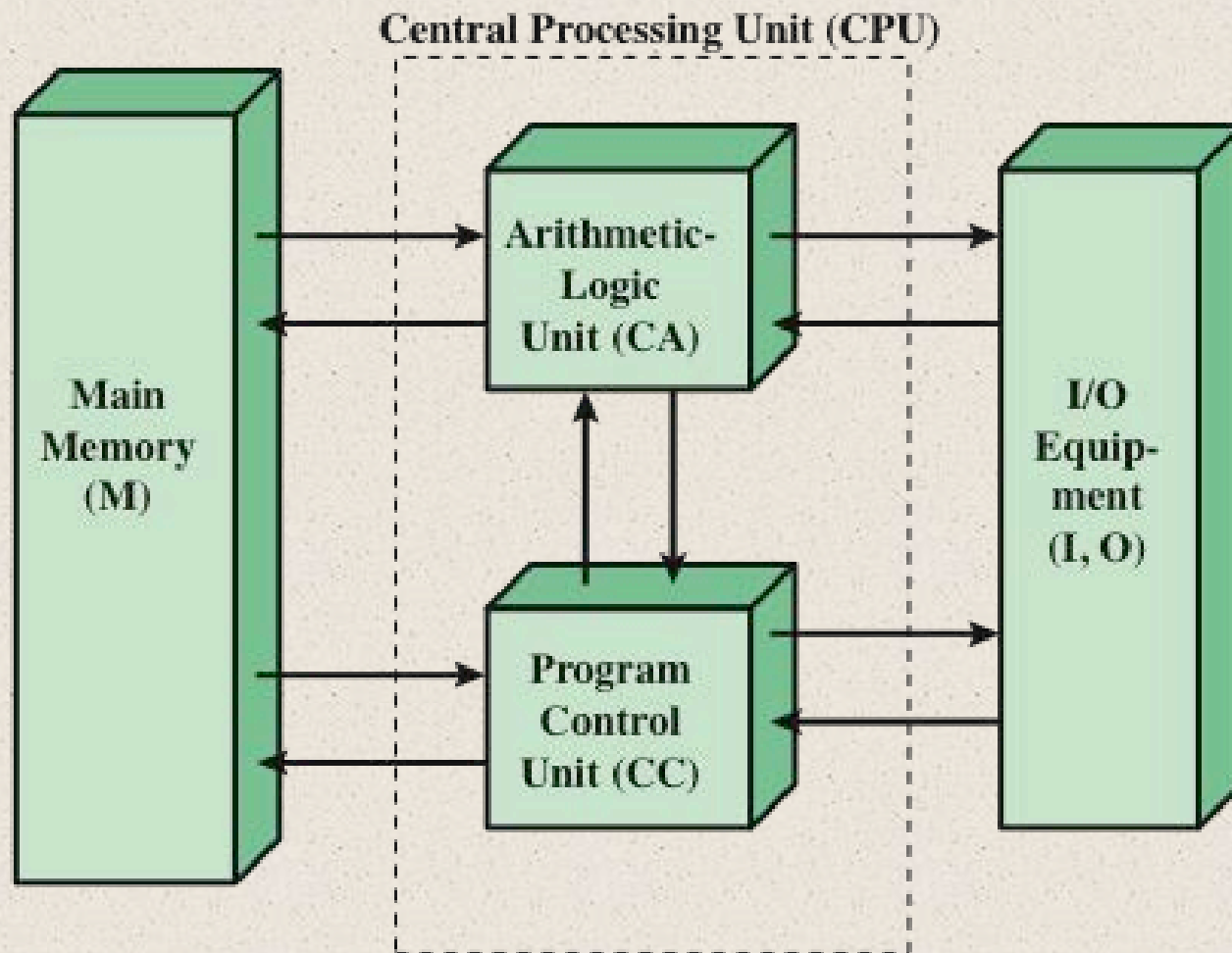
Figure 2.1 Structure of the IAS Computer

IAS Memory Formats

- The memory of the IAS consists of 1000 storage locations (called *words*) of 40 bits each

- Both data and instructions are stored there

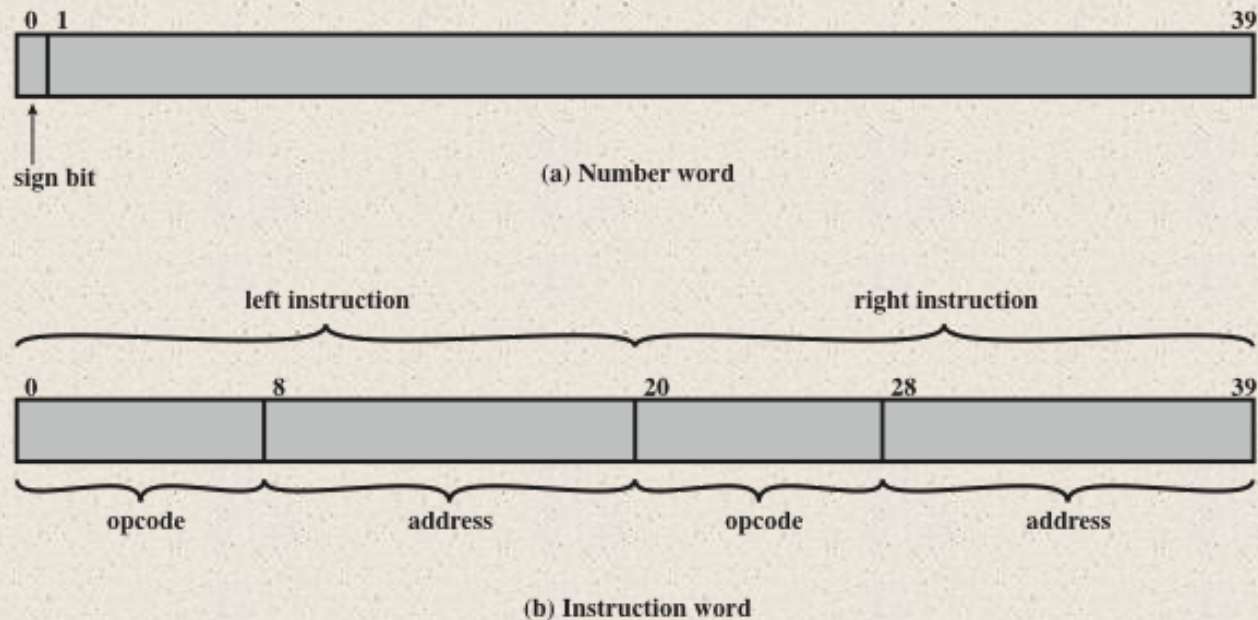- Numbers are represented in binary form and each instruction is a binary code

Figure 2.2 IAS Memory Formats

Registers

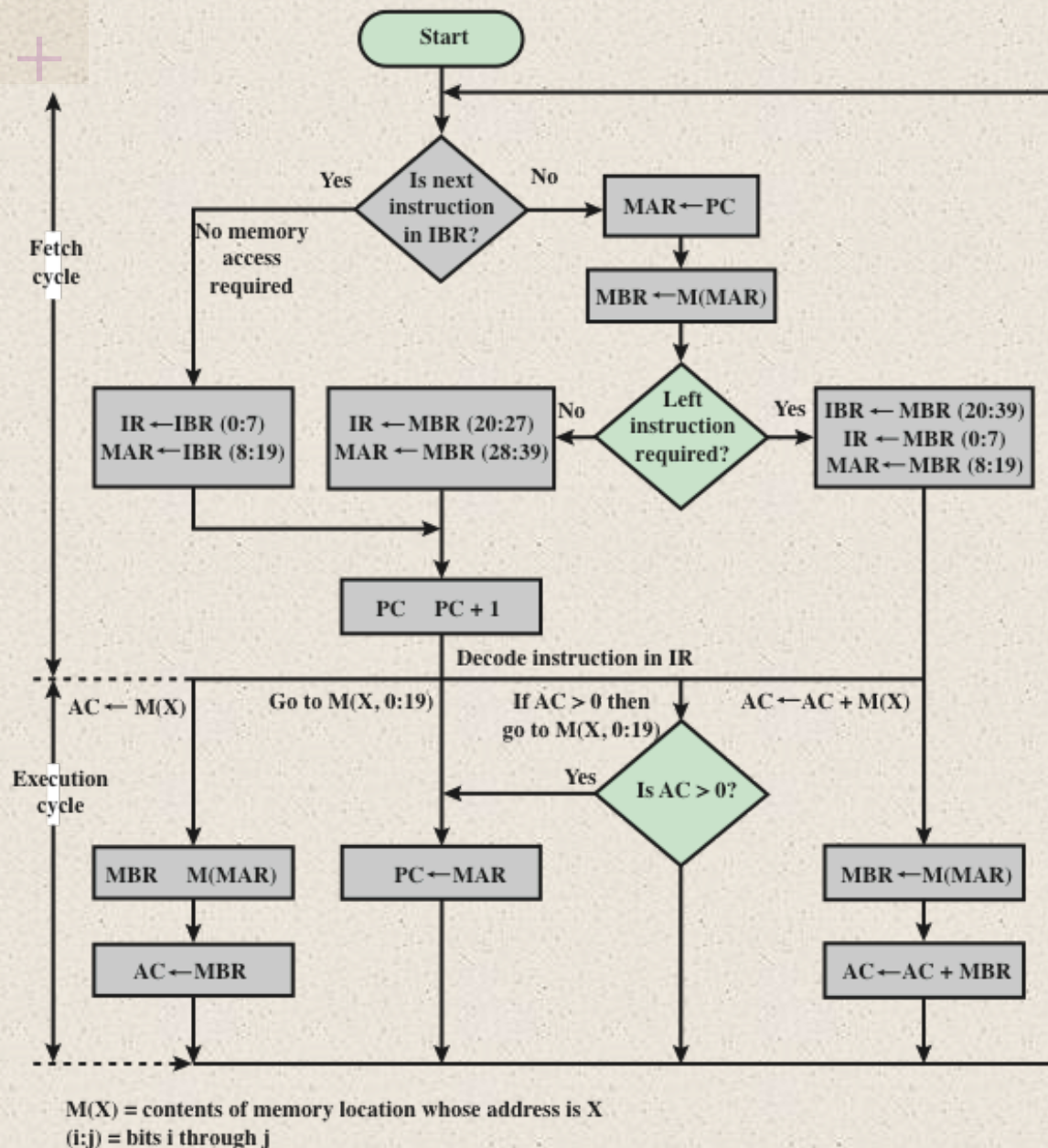|	
Memory buffer register (MBR)	• Contains a word to be stored in memory or sent to the I/O unit • Or is used to receive a word from memory or from the I/O unit
Memory address register (MAR)	• Specifies the address in memory of the word to be written from or read into the MBR
Instruction register (IR)	• Contains the 8-bit opcode instruction being executed
Instruction buffer register (IBR)	• Employed to temporarily hold the right-hand instruction from a word in memory
Program counter (PC)	• Contains the address of the next instruction pair to be fetched from memory
Accumulator (AC) and multiplier quotient (MQ)	• Employed to temporarily hold operands and results of ALU operations

Structure of IAS Computer
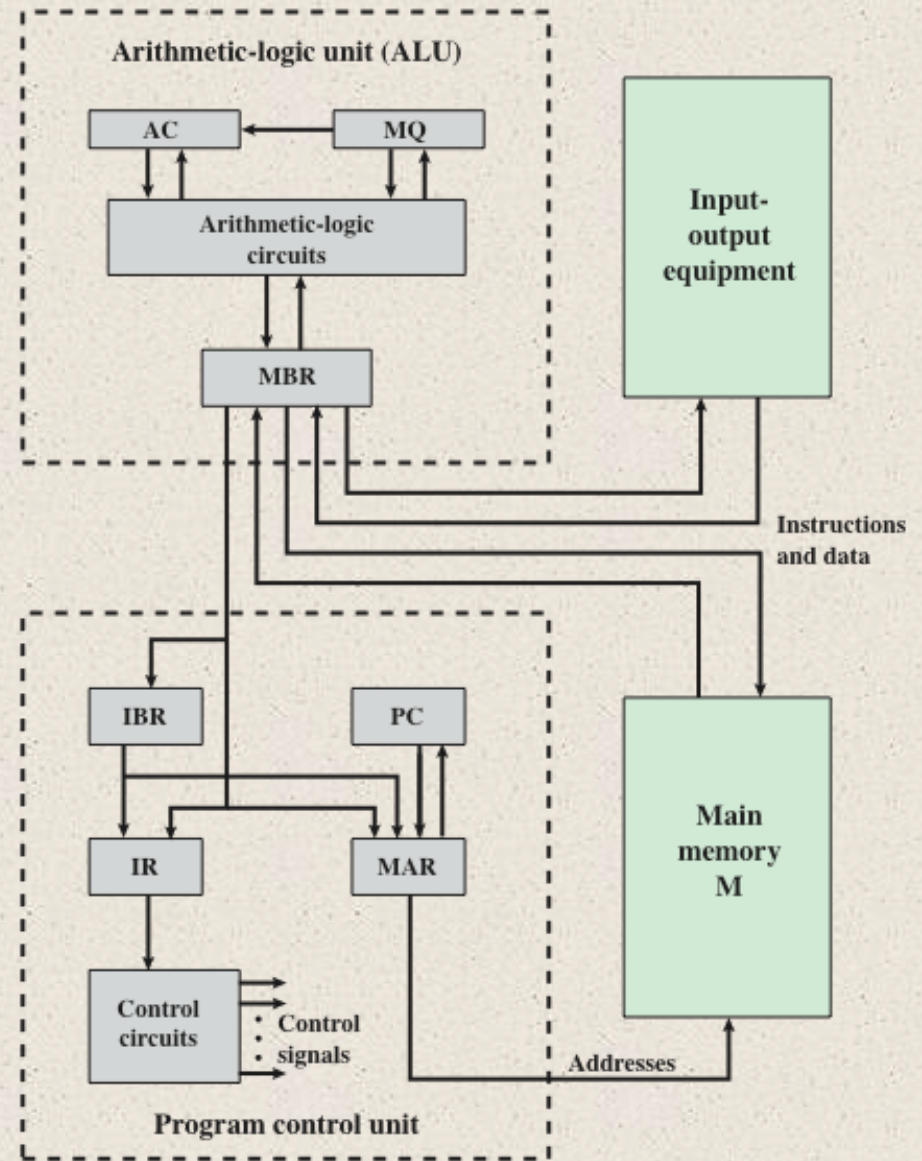
Figure 2.3   Expanded Structure of IAS Computer

IAS Operations

Figure 2.4 Partial Flowchart of IAS Operation

Start

Is next instruction in IBR?

Yes — No memory access required

No — MAR ← PC

MBR ← M(MAR)

Fetch cycle

IR ← IBR (0:7)
MAR ← IBR (8:19)

IR ← MBR (20:27)
MAR ← MBR (28:39)

Left instruction required?

No

Yes — IBR ← MBR (20:39)
IR ← MBR (0:7)
MAR ← MBR (8:19)

PC   PC + 1

Decode instruction in IR

AC ← M(X)   Go to M(X, 0:19)   If AC > 0 then go to M(X, 0:19)   AC ← AC + M(X)

Execution cycle

Is AC > 0?

Yes

MBR   M(MAR)

PC ← MAR

MBR ← M(MAR)

AC ← MBR

AC ← AC + MBR

M(X) = contents of memory location whose address is X
(i:j) = bits i through j

| Instruction Type	Opcode	Symbolic Representation	Description
Data transfer	00001010	LOAD MQ	Transfer contents of register MQ to the accumulator AC
	00001001	LOAD MQ,M(X)	Transfer contents of memory location X to MQ
	00100001	STOR M(X)	Transfer contents of accumulator to memory location X
	00000001	LOAD M(X)	Transfer M(X) to the accumulator
	00000010	LOAD –M(X)	Transfer –M(X) to the accumulator
	00000011	LOAD |M(X)|	Transfer absolute value of M(X) to the accumulator
	00000100	LOAD –|M(X)|	Transfer –|M(X)| to the accumulator
Unconditional branch	00001101	JUMP M(X,0:19)	Take next instruction from left half of M(X)
	00001110	JUMP M(X,20:39)	Take next instruction from right half of M(X)
Conditional branch	00001111	JUMP+ M(X,0:19)	If number in the accumulator is nonnegative, take next instruction from left half of M(X)
	00010000	JUMP+ M(X,20:39)	If number in the accumulator is nonnegative, take next instruction from right half of M(X)
Arithmetic	00000101	ADD M(X)	Add M(X) to AC; put the result in AC
	00000111	ADD |M(X)|	Add |M(X)| to AC; put the result in AC
	00000110	SUB M(X)	Subtract M(X) from AC; put the result in AC
	00001000	SUB |M(X)|	Subtract |M(X)| from AC; put the remainder in AC
	00001011	MUL M(X)	Multiply M(X) by MQ; put most significant bits of result in AC, put least significant bits in MQ
	00001100	DIV M(X)	Divide AC by M(X); put the quotient in MQ and the remainder in AC
	00010100	LSH	Multiply accumulator by 2; i.e., shift left one bit position
	00010101	RSH	Divide accumulator by 2; i.e., shift right one position
Address modify	00010010	STOR M(X,8:19)	Replace left address field at M(X) by 12 rightmost bits of AC
	00010011	STOR M(X,28:39)	Replace right address field at M(X) by 12 rightmost bits of AC

The IAS Instruction Set

The IAS Instruction Set

Commercial Computers

UNIVAC
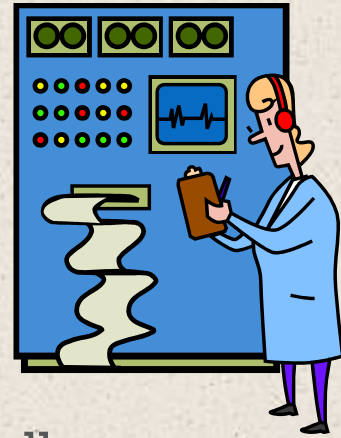
- 1947 – Eckert and Mauchly formed the Eckert-Mauchly Computer Corporation to manufacture computers commercially

- UNIVAC I (Universal Automatic Computer)
  - First successful commercial computer
  - Was intended for both scientific and commercial applications
  - Commissioned by the US Bureau of Census for 1950 calculations

- The Eckert-Mauchly Computer Corporation became part of the UNIVAC division of the Sperry-Rand Corporation

- UNIVAC II – delivered in the late 1950's
  - Had greater memory capacity and higher performance

- Backward compatible

History of Computers
Second Generation: Transistors

- Smaller

- Cheaper

- Dissipates less heat than a vacuum tube

- Is a *solid state device* made from silicon

- Was invented at Bell Labs in 1947

- It was not until the late 1950's that fully transistorized computers were commercially available
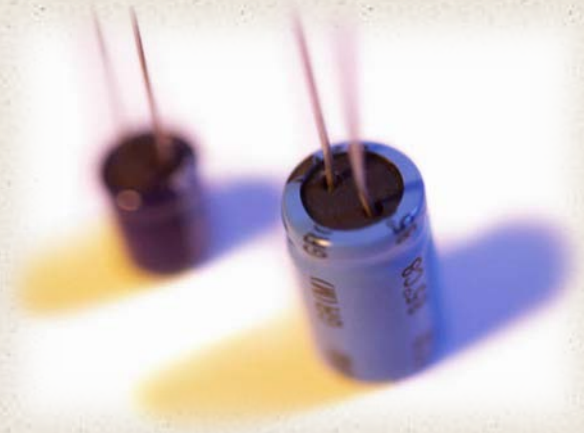
Second Generation Computers

- Introduced:
  - More complex arithmetic and logic units and control units
  - The use of high-level programming languages
  - Provision of *system software* which provided the ability to:
    - load programs
    - move data to peripherals and libraries
    - perform common computations

- Appearance of the Digital Equipment Corporation (DEC) in 1957

- PDP-1 was DEC's first computer

- This began the mini-computer phenomenon that would become so prominent in the third generation
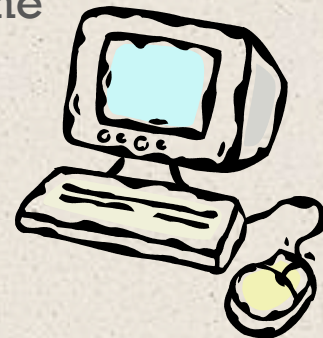
- Was the major manufacturer of punched-card processing equipment

- Delivered its first electronic stored-program computer (701) in 1953
  - Intended primarily for scientific applications

- Introduced 702 product in 1955
  - Hardware features made it suitable to business applications

- Series of 700/7000 computers established IBM as the overwhelmingly dominant computer manufacturer
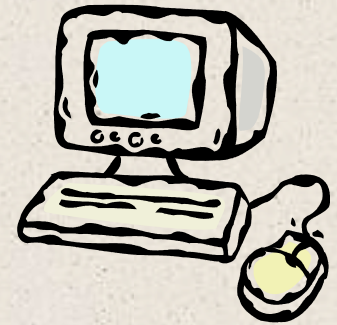
IBM

Example
Members of the
IBM 700/7000 Series

Model Number	First Delivery	CPU Tech-nology	Memory Technology	Cycle Time (μs)	Memory Size (K)	Number of Opcodes	Number of Index Registers	Hardwired Floating-Point	I/O Overlap (Channels)	Instruction Fetch Overlap	Speed (relative to 701)
701	1952	Vacuum tubes	Electrostatic tubes	30	2–4	24	0	no	no	no	1
704	1955	Vacuum tubes	Core	12	4–32	80	3	yes	no	no	2.5
709	1958	Vacuum tubes	Core	12	32	140	3	yes	yes	no	4
7090	1960	Transistor	Core	2.18	32	169	3	yes	yes	no	25
7094 I	1962	Transistor	Core	2	32	185	7	yes (double precision)	yes	yes	30
7094 II	1964	Transistor	Core	1.4	32	185	7	yes (double precision)	yes	yes	50

Example Members of the IBM 700/7000 Series

History of Computers

Third Generation: Integrated Circuits

- 1958 – the invention of the integrated circuit

- *Discrete component*
  - Single, self-contained transistor
  - Manufactured separately, packaged in their own containers, and soldered or wired together onto masonite-like circuit boards
  - Manufacturing process was expensive and cumbersome

- The two most important members of the third generation were the IBM System/360 and the DEC PDP-8

Microelectronics
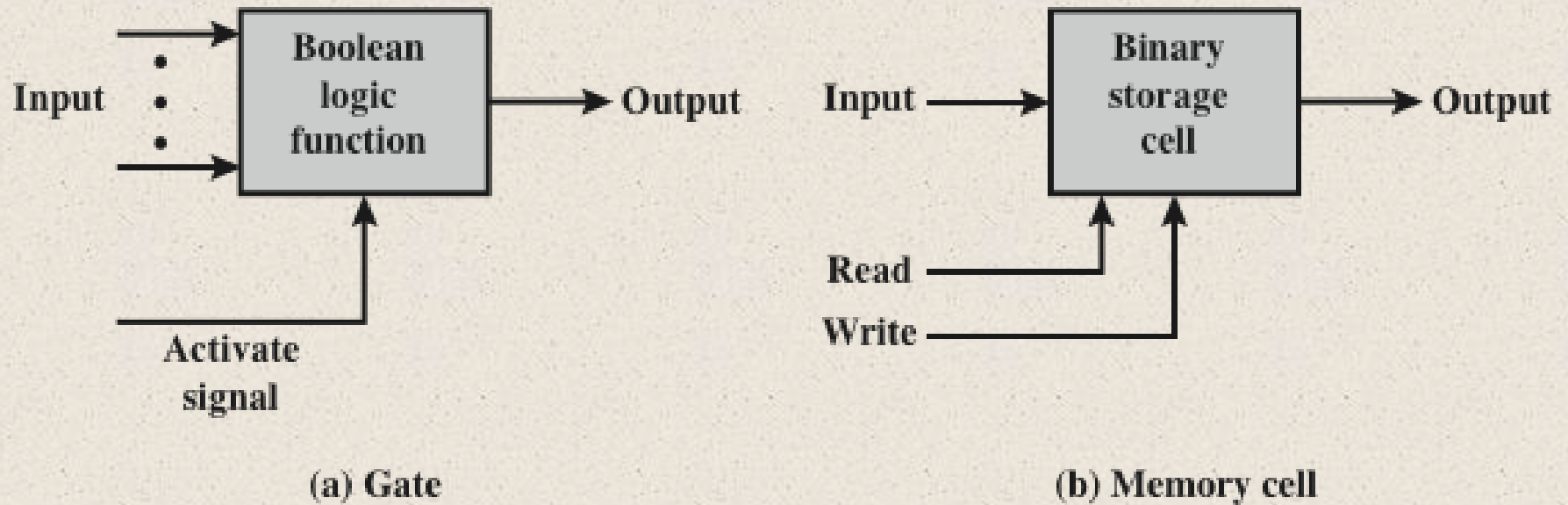
Figure 2.6 Fundamental Computer Elements

Integrated Circuits

- Data storage – provided by memory cells

- Data processing – provided by gates

- Data movement – the paths among components are used to move data from memory to memory and from memory through gates to memory

- Control – the paths among components can carry control signals

- A computer consists of gates, memory cells, and interconnections among these elements

- The gates and memory cells are constructed of simple digital electronic components

- Exploits the fact that such components as transistors, resistors, and conductors can be fabricated from a semiconductor such as silicon

- Many transistors can be produced at the same time on a single wafer of silicon

- Transistors can be connected with a processor metallization to form circuits
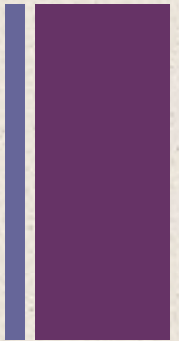
Chip Growth
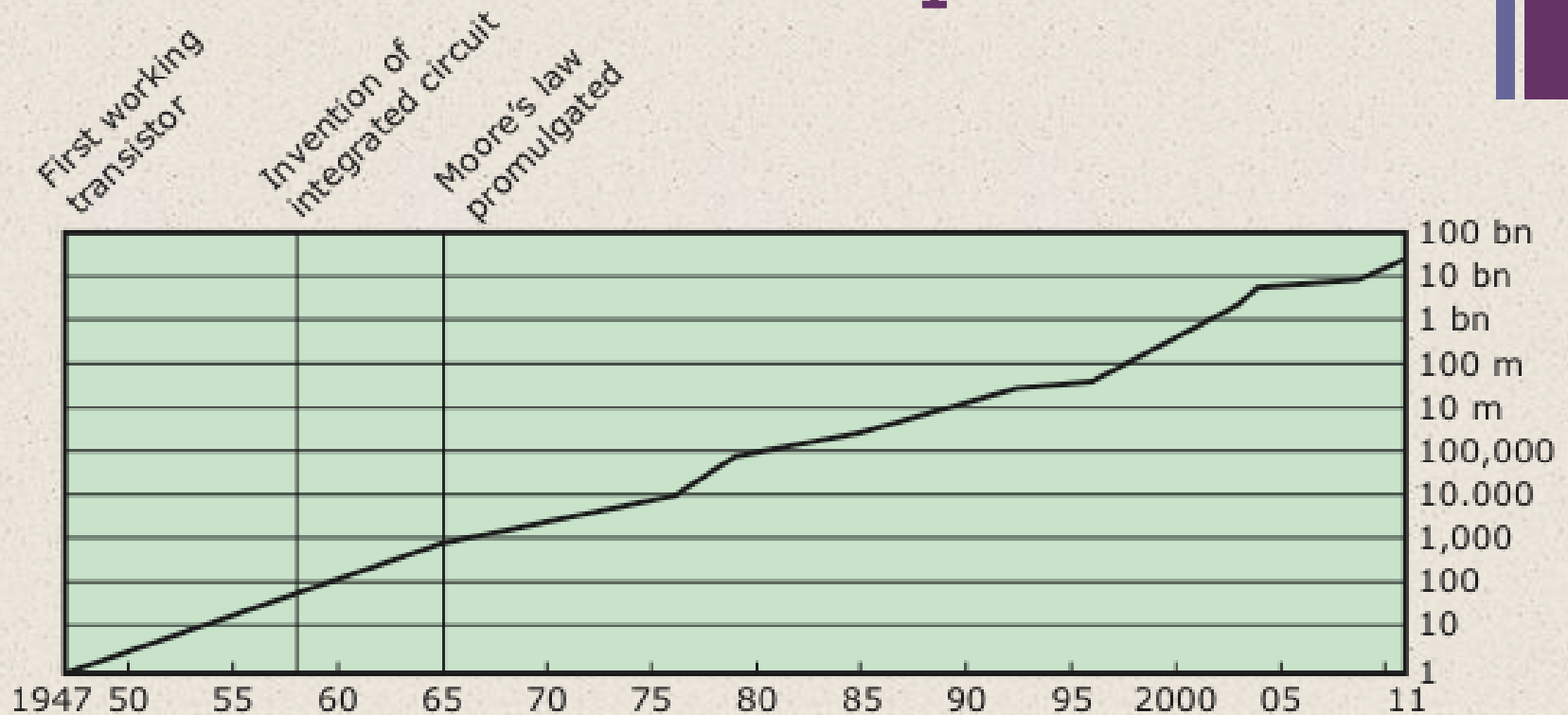
Figure 2.8 Growth in Transistor Count on Integrated Circuits (DRAM memory)

Moore's Law

1965; Gordon Moore – co-founder of Intel

Observed number of transistors that could be put on a single chip was doubling every year

The pace slowed to a doubling every 18 months in the 1970's but has sustained that rate ever since

Consequences of Moore's law:

The cost of computer logic and memory circuitry has fallen at a dramatic rate

The electrical path length is shortened, increasing operating speed

Computer becomes smaller and is more convenient to use in a variety of environments

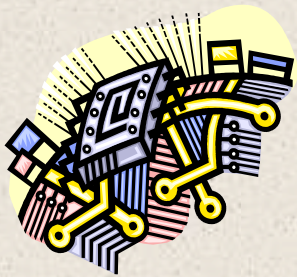Reduction in power and cooling requirements

Fewer interchip connections

+

Later Generations

LSI
Large Scale Integration

VLSI
Very Large Scale Integration

ULSI
Ultra Large Scale Integration

Semiconductor Memory Microprocessors

Semiconductor Memory

In 1970 Fairchild produced the first relatively capacious semiconductor memory

| Chip was about the size of a single core	Could hold 256 bits of memory	Non-destructive	Much faster than core

In 1974 the price per bit of semiconductor memory dropped below the price per bit of core memory

There has been a continuing and rapid decline in memory cost accompanied by a corresponding increase in physical memory density	Developments in memory and processor technologies changed the nature of computers in less than a decade
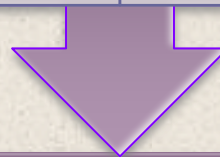
Since 1970 semiconductor memory has been through 13 generations

Each generation has provided four times the storage density of the previous generation, accompanied by declining cost per bit and declining access time

Microprocessors

- The density of elements on processor chips continued to rise
  - More and more elements were placed on each chip so that fewer and fewer chips were needed to construct a single computer processor

- 1971 Intel developed 4004
  - First chip to contain all of the components of a CPU on a single chip
  - Birth of microprocessor

- 1972 Intel developed 8008
  - First 8-bit microprocessor

- 1974 Intel developed 8080
  - First general purpose microprocessor
  - Faster, has a richer instruction set, has a large addressing capability

Evolution of Intel Microprocessors

	4004	8008	8080	8086	8088
Introduced	1971	1972	1974	1978	1979
Clock speeds	108 kHz	108 kHz	2 MHz	5 MHz, 8 MHz, 10 MHz	5 MHz, 8 MHz
Bus width	4 bits	8 bits	8 bits	16 bits	8 bits
Number of transistors	2,300	3,500	6,000	29,000	29,000
Feature size (μm)	10		6	3	6
Addressable memory	640 Bytes	16 KB	64 KB	1 MB	1 MB

a. 1970s Processors

	80286	386TM DX	386TM SX	486TM DX CPU
Introduced	1982	1985	1988	1989
Clock speeds	6 MHz - 12.5 MHz	16 MHz - 33 MHz	16 MHz - 33 MHz	25 MHz - 50 MHz
Bus width	16 bits	32 bits	16 bits	32 bits
Number of transistors	134,000	275,000	275,000	1.2 million
Feature size (μm)	1.5	1	1	0.8 - 1
Addressable memory	16 MB	4 GB	16 MB	4 GB
Virtual memory	1 GB	64 TB	64 TB	64 TB
Cache	—	—	—	8 kB

b. 1980s Processors

Evolution of Intel Microprocessors

	486TM SX	Pentium	Pentium Pro	Pentium II
Introduced	1991	1993	1995	1997
Clock speeds	16 MHz - 33 MHz	60 MHz - 166 MHz,	150 MHz - 200 MHz	200 MHz - 300 MHz
Bus width	32 bits	32 bits	64 bits	64 bits
Number of transistors	1.185 million	3.1 million	5.5 million	7.5 million
Feature size (μm)	1	0.8	0.6	0.35
Addressable memory	4 GB	4 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	8 kB	8 kB	512 kB L1 and 1 MB L2	512 kB L2

c. 1990s Processors

	Pentium III	Pentium 4	Core 2 Duo	Core i7 EE 990
Introduced	1999	2000	2006	2011
Clock speeds	450 - 660 MHz	1.3 - 1.8 GHz	1.06 - 1.2 GHz	3.5 GHz
Bus width	64 bits	64 bits	64 bits	64 bits
Number of transistors	9.5 million	42 million	167 million	1170 million
Feature size (nm)	250	180	65	32
Addressable memory	64 GB	64 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	512 kB L2	256 kB L2	2 MB L2	1.5 MB L2/12 MB L3

d. Recent Processors

Microprocessor Speed

Techniques built into contemporary processors include:

Pipelining	• Processor moves data or instructions into a conceptual pipe with all stages of the pipe processing simultaneously
Branch prediction	• Processor looks ahead in the instruction code fetched from memory and predicts which branches, or groups of instructions, are likely to be processed next
Data flow analysis	• Processor analyzes which instructions are dependent on each other's results, or data, to create an optimized schedule of instructions
Speculative execution	• Using branch prediction and data flow analysis, some processors speculatively execute instructions ahead of their actual appearance in the program execution, holding the results in temporary locations, keeping execution engines as busy as possible

Performance Balance

- Adjust the organization and architecture to compensate for the mismatch among the capabilities of the various components

- Architectural examples include:

Increase the number of bits that are retrieved at one time by making DRAMs "wider" rather than "deeper" and by using wide bus data paths

Reduce the frequency of memory access by incorporating increasingly complex and efficient cache structures between the processor and main memory

Change the DRAM interface to make it more efficient by including a cache or other buffering scheme on the DRAM chip

Increase the interconnect bandwidth between processors and memory by using higher speed buses and a hierarchy of buses to buffer and structure data flow

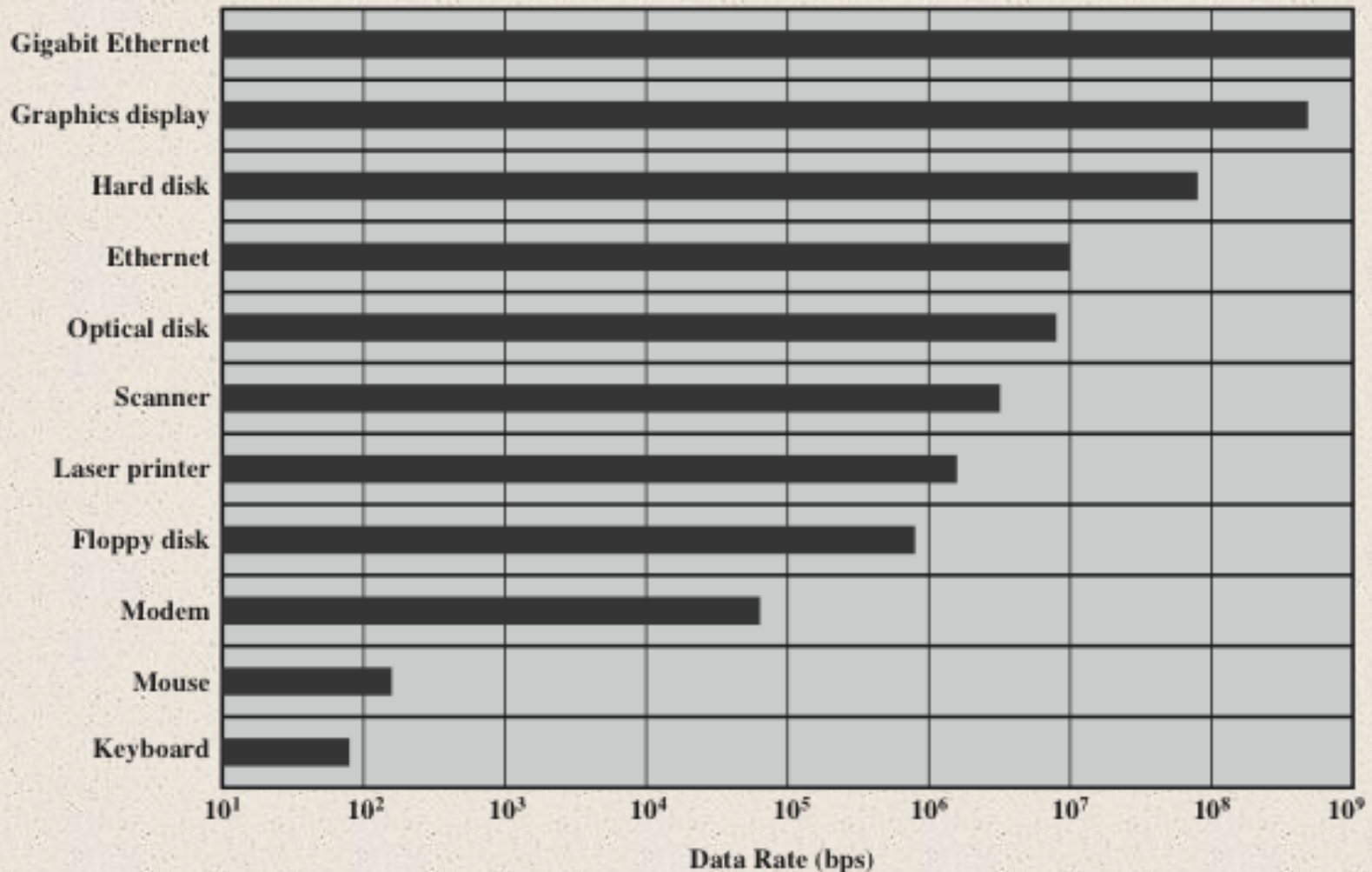# Typical I/O Device Data Rates

Figure 2.10 Typical I/O Device Data Rates

Improvements in Chip Organization and Architecture

- Increase hardware speed of processor
  - Fundamentally due to shrinking logic gate size
    - More gates, packed more tightly, increasing clock rate
    - Propagation time for signals reduced

- Increase size and speed of caches
  - Dedicating part of processor chip
    - Cache access times drop significantly

- Change processor organization and architecture
  - Increase effective speed of instruction execution
  - Parallelism

Problems with Clock Speed and Login Density

- Power
  - Power density increases with density of logic and clock speed
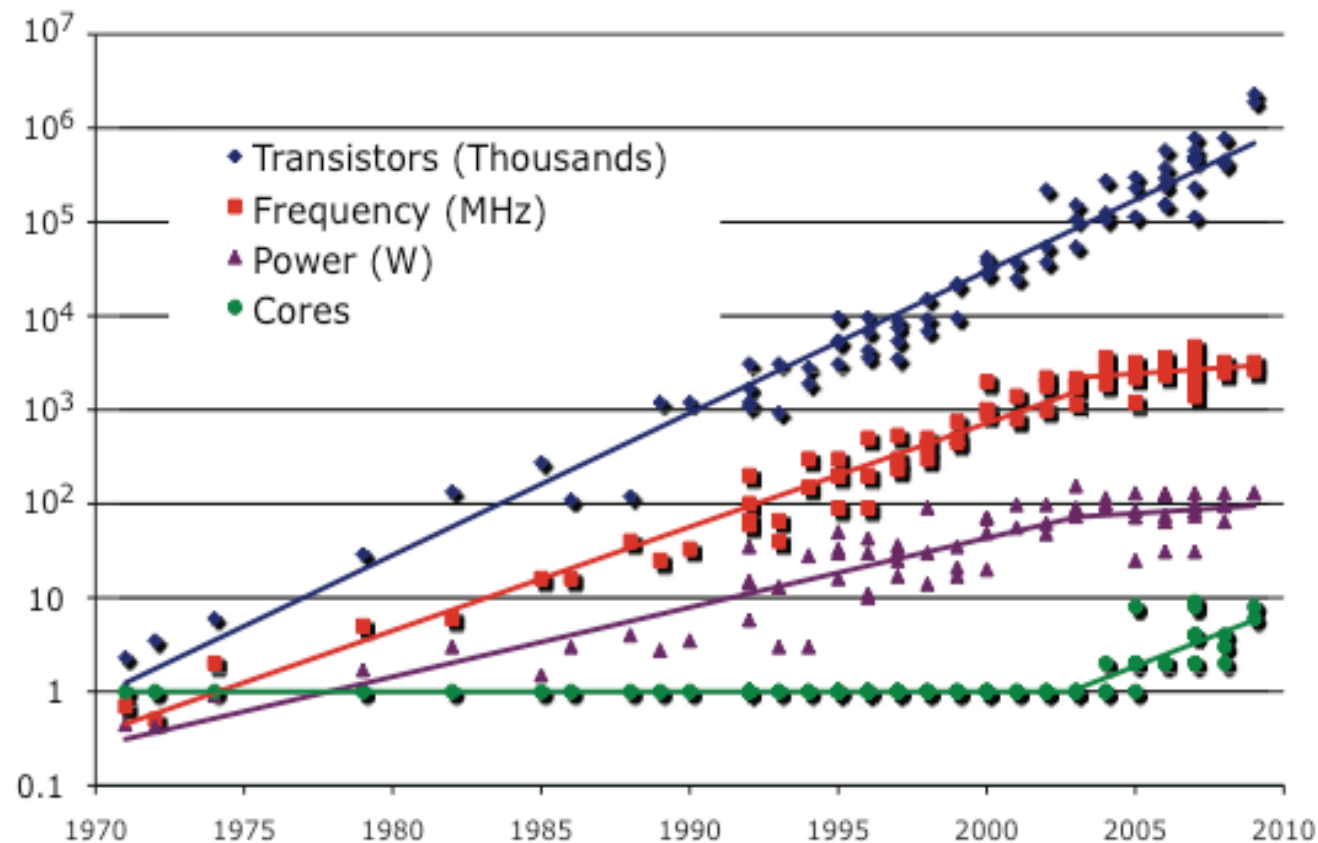 - Dissipating heat

- RC delay
  - Speed at which electrons flow limited by resistance and capacitance of metal wires connecting them
  - Delay increases as RC product increases
  - Wire interconnects thinner, increasing resistance
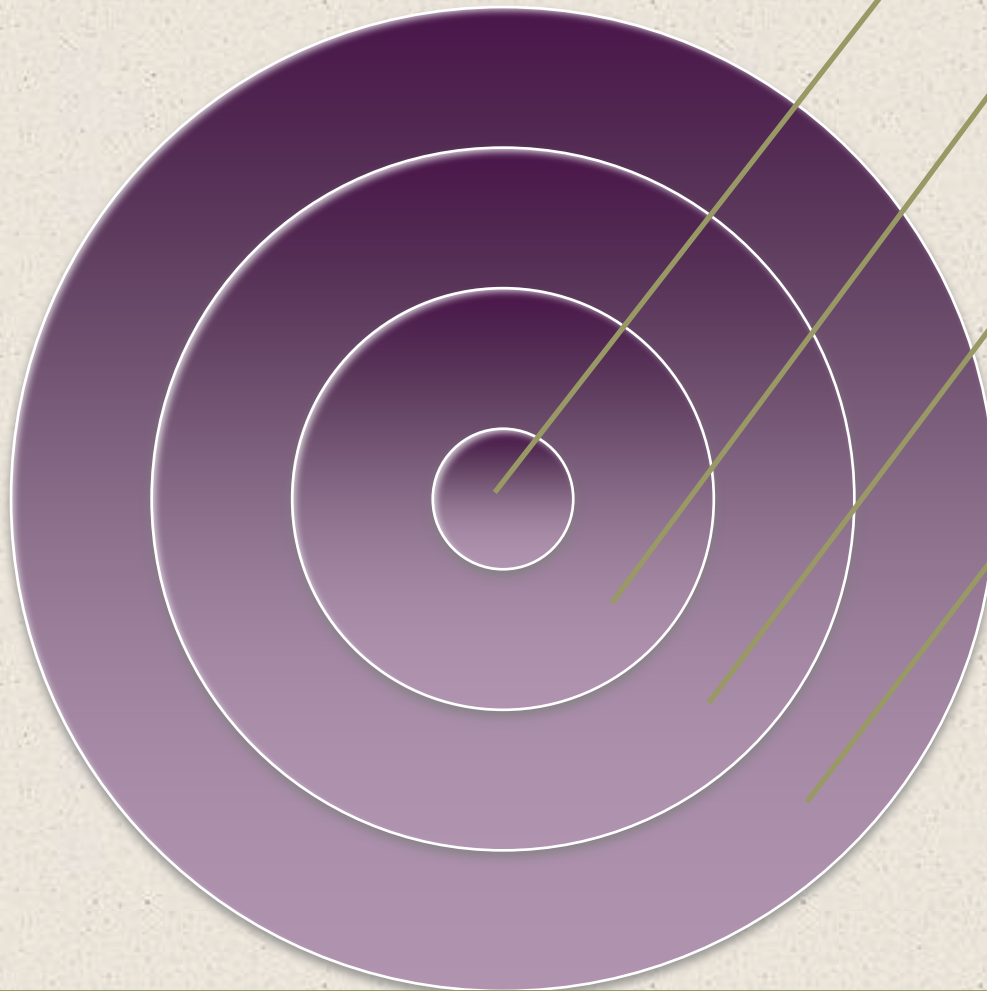 - Wires closer together, increasing capacitance

- Memory latency
  - Memory speeds lag processor speeds

Processor Trends

Multicore

The use of multiple processors on the same chip provides the potential to increase performance without increasing the clock rate

Strategy is to use two simpler processors on the chip rather than one more complex processor

With two processors larger caches are justified

As caches became larger it made performance sense to create two and then three levels of cache on a chip

Many Integrated Core (MIC) Graphics Processing Unit (GPU)

MIC

- Leap in performance as well as the challenges in developing software to exploit such a large number of cores

- The multicore and MIC strategy involves a homogeneous collection of general purpose processors on a single chip

GPU

- Core designed to perform parallel operations on graphics data

- Traditionally found on a plug-in graphics card, it is used to encode and render 2D and 3D graphics as well as process video

- Used as vector processors for a variety of applications that require repetitive computations

+ Overview

- Results of decades of design effort on complex instruction set computers (CISCs)

- Excellent example of CISC design

- Incorporates the sophisticated design principles once found only on mainframes and supercomputers

- An alternative approach to processor design is the reduced instruction set computer (RISC)

- The ARM architecture is used in a wide variety of embedded systems and is one of the most powerful and best designed RISC based systems on the market

- In terms of market share Intel is ranked as the number one maker of microprocessors for non-embedded systems
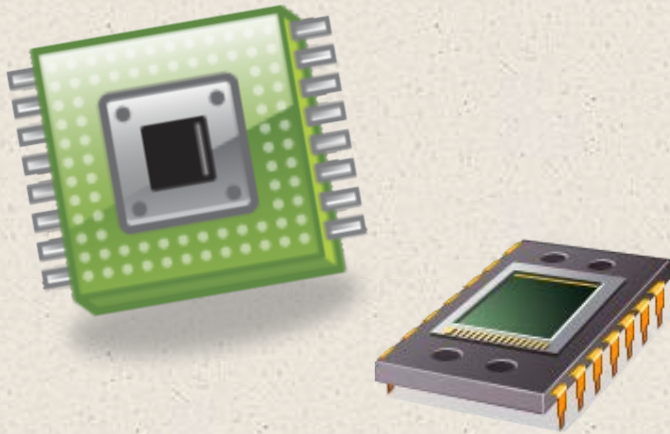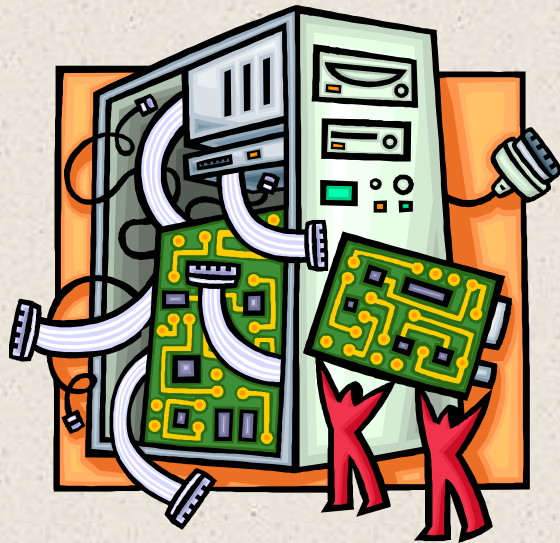
ARM

Intel

x86 Architecture

CISC

RISC

x86 Evolution

8080
- First general purpose microprocessor
- 8-bit machine with an 8-bit data path to memory
- Used in the first personal computer (Altair)

8086
- 16-bit machine
- Used an instruction cache, or queue
- First appearance of the x86 architecture

8088
- used in IBM's first personal computer

80286
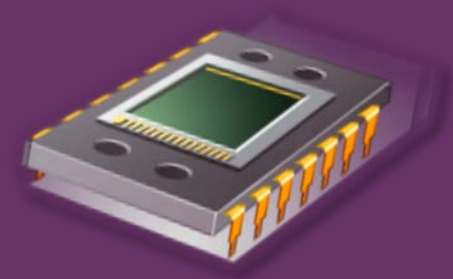- Enabled addressing a 16-MByte memory instead of just 1 MByte

80386
- Intel's first 32-bit machine
- First Intel processor to support multitasking

80486
- More sophisticated cache technology and instruction pipelining
- Built-in math coprocessor

x86 Evolution - Pentium

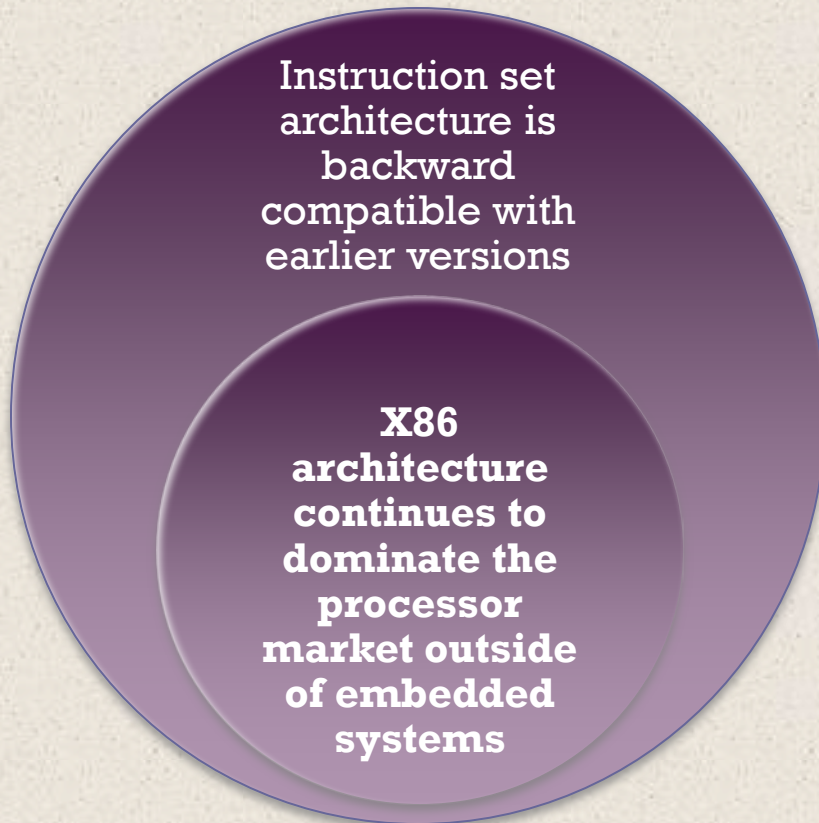| Pentium	Pentium Pro	Pentium II	Pentium III	Pentium 4
• Superscalar • Multiple instructions executed in parallel	• Increased superscalar organization • Aggressive register renaming • Branch prediction • Data flow analysis • Speculative execution	• MMX technology • Designed specifically to process video, audio, and graphics data	• Additional floating-point instructions to support 3D graphics software	• Includes additional floating-point and other enhancements for multimedia

x86 Evolution (continued)

Instruction set architecture is backward compatible with earlier versions

X86 architecture continues to dominate the processor market outside of embedded systems

- Core

  - First Intel x86 microprocessor with a dual core, referring to the implementation of two processors on a single chip

- Core 2

  - Extends the architecture to 64 bits

  - Recent Core offerings have up to 10 processors per chip

General definition:

"A combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a dedicated function. In many cases, embedded systems are part of a larger system or product, as in the case of an antilock braking system in a car."

Embedded

Systems

Examples of Embedded Systems and Their Markets

Market	Embedded Device
Automotive	Ignition system Engine control Brake system
Consumer electronics	Digital and analog televisions Set-top boxes (DVDs, VCRs, Cable boxes) Personal digital assistants (PDAs) Kitchen appliances (refrigerators, toasters, microwave ovens) Automobiles Toys/games Telephones/cell phones/pagers Cameras Global positioning systems
Industrial control	Robotics and controls systems for manufacturing Sensors
Medical	Infusion pumps Dialysis machines Prosthetic devices Cardiac monitors
Office automation	Fax machine Photocopier Printers Monitors Scanners

Embedded Systems

Requirements and Constraints

Small to large systems, implying different cost constraints and different needs for optimization and reuse

Relaxed to very strict requirements and combinations of different quality requirements with respect to safety, reliability, real-time and flexibility

Different models of computation ranging from discrete event systems to hybrid systems

Different application characteristics resulting in static versus dynamic loads, slow to fast speed, compute versus interface intensive tasks, and/or combinations thereof

Short to long life times

Different environmental conditions in terms of radiation, vibrations, and humidity

Possible Organization of an Embedded System
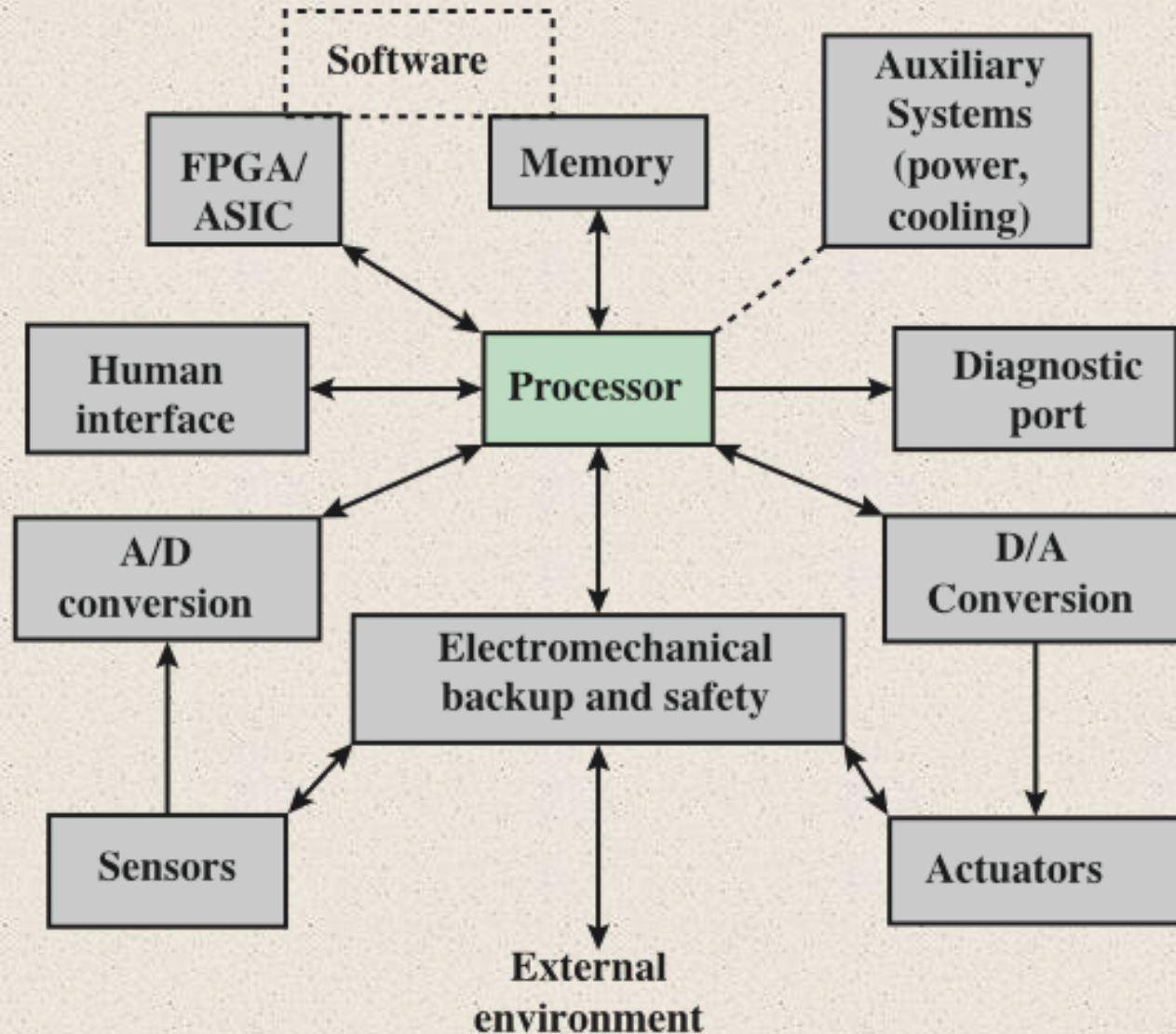
Acorn RISC Machine (ARM)

- Family of RISC-based microprocessors and microcontrollers

- Designs microprocessor and multicore architectures and licenses them to manufacturers

- Chips are high-speed processors that are known for their small die size and low power requirements

- Widely used in PDAs and other handheld devices

- Chips are the processors in iPod and iPhone devices

- Most widely used embedded processor architecture

- Most widely used processor architecture of any kind
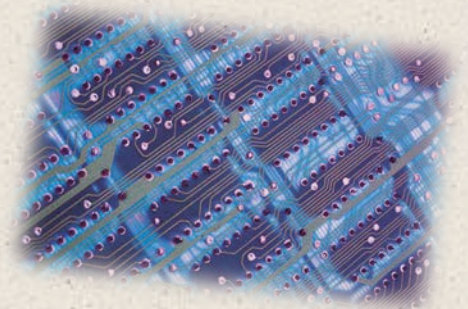
Family	Notable Features	Cache	Typical MIPS @ MHz
ARM1	32-bit RISC	None	
ARM2	Multiply and swap instructions; Integrated memory management unit, graphics and I/O processor	None	7 MIPS @ 12 MHz
ARM3	First use of processor cache	4 KB unified	12 MIPS @ 25 MHz
ARM6	First to support 32-bit addresses; floating-point unit	4 KB unified	28 MIPS @ 33 MHz
ARM7	Integrated SoC	8 KB unified	60 MIPS @ 60 MHz
ARM8	5-stage pipeline; static branch prediction	8 KB unified	84 MIPS @ 72 MHz
ARM9		16 KB/16 KB	300 MIPS @ 300 MHz
ARM9E	Enhanced DSP instructions	16 KB/16 KB	220 MIPS @ 200 MHz
ARM10E	6-stage pipeline	32 KB/32 KB	
ARM11	9-stage pipeline	Variable	740 MIPS @ 665 MHz
Cortex	13-stage superscalar pipeline	Variable	2000 MIPS @ 1 GHz
XScale	Applications processor; 7-stage pipeline	32 KB/32 KB L1 512 KB L2	1000 MIPS @ 1.25 GHz

DSP = digital signal processor   SoC = system on a chip

ARM Design Categories

- ARM processors are designed to meet the needs of three system categories:

  - Secure applications
   - Smart cards, SIM cards, and payment terminals

  - Embedded real-time systems
    - Systems for storage, automotive body and power-train, industrial, and networking applications

  - Application platforms
    - Devices running open operating systems including Linux, Palm OS, Symbian OS, and Windows CE in wireless, consumer entertainment and digital imaging applications
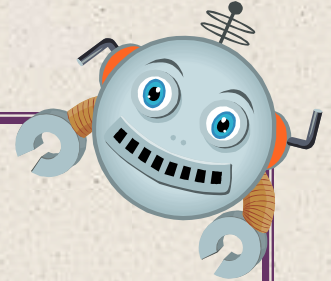
System Clock
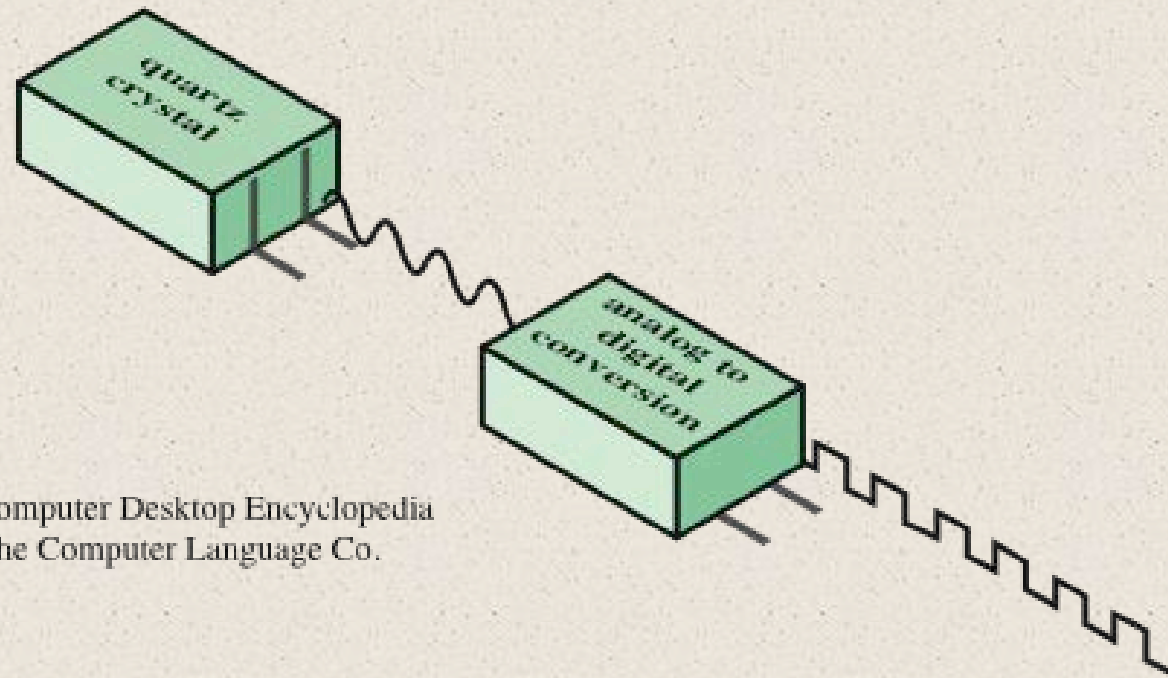
From Computer Desktop Encyclopedia
1998, The Computer Language Co.

Figure 2.13   System Clock

Benchmarks

For example, consider this high-level language statement:

A = B + C /* assume all quantities in main memory */

With a traditional instruction set architecture, referred to as a complex instruction set computer (CISC), this instruction can be compiled into one processor instruction:

add mem(B), mem(C), mem (A)

On a typical RISC machine, the compilation would look something like this:

load mem(B), reg(1);
load mem(C), reg(2);
add reg(1), reg(2), reg(3);
store reg(3), mem (A)

Desirable Benchmark Characteristics

Written in a high-level language, making it portable across different machines

Representative of a particular kind of programming style, such as system programming, numerical programming, or commercial programming

Can be measured easily

Has wide distribution

System Performance Evaluation Corporation (SPEC)

- Benchmark suite
  - A collection of programs, defined in a high-level language
  - Attempts to provide a representative test of a computer in a particular application or system programming area

- SPEC
  - An industry consortium
  - Defines and maintains the best known collection of benchmark suites
  - Performance measurements are widely used for comparison and research purposes

SPEC CPU2006

- Best known SPEC benchmark suite

- Industry standard suite for processor intensive applications

- Appropriate for measuring performance for applications that spend most of their time doing computation rather than I/O

- Consists of 17 floating point programs written in C, C++, and Fortran and 12 integer programs written in C and C++

- Suite contains over 3 million lines of code

- Fifth generation of processor intensive suites from SPEC

Amdahl's Law
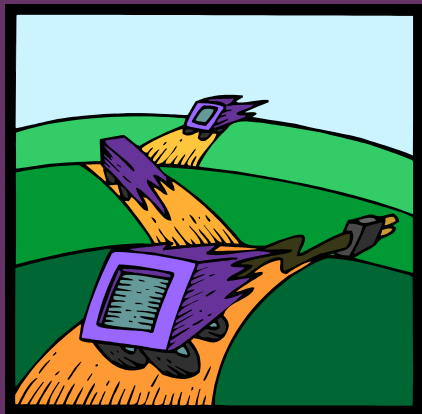
- Gene Amdahl [AMDA67]

- Deals with the potential speedup of a program using multiple processors compared to a single processor

- Illustrates the problems facing industry in the development of multi-core machines
  - Software must be adapted to a highly parallel execution environment to exploit the power of parallel processing

- Can be generalized to evaluate and design technical improvement in a computer system

Little's Law

- Fundamental and simple relation with broad applications

- Can be applied to almost any system that is statistically in steady state, and in which there is no leakage

- Queuing system
  - If server is idle an item is served immediately, otherwise an arriving item joins a queue
  - There can be a single queue for a single server or for multiple servers, or multiples queues with one being for each of multiple servers

- Average number of items in a queuing system equals the average rate at which items arrive multiplied by the time that an item spends in the system
  - Relationship requires very few assumptions
  - Because of its simplicity and generality it is extremely useful