+

William Stallings Computer Organization and Architecture 10th Edition

Chapter 1

+

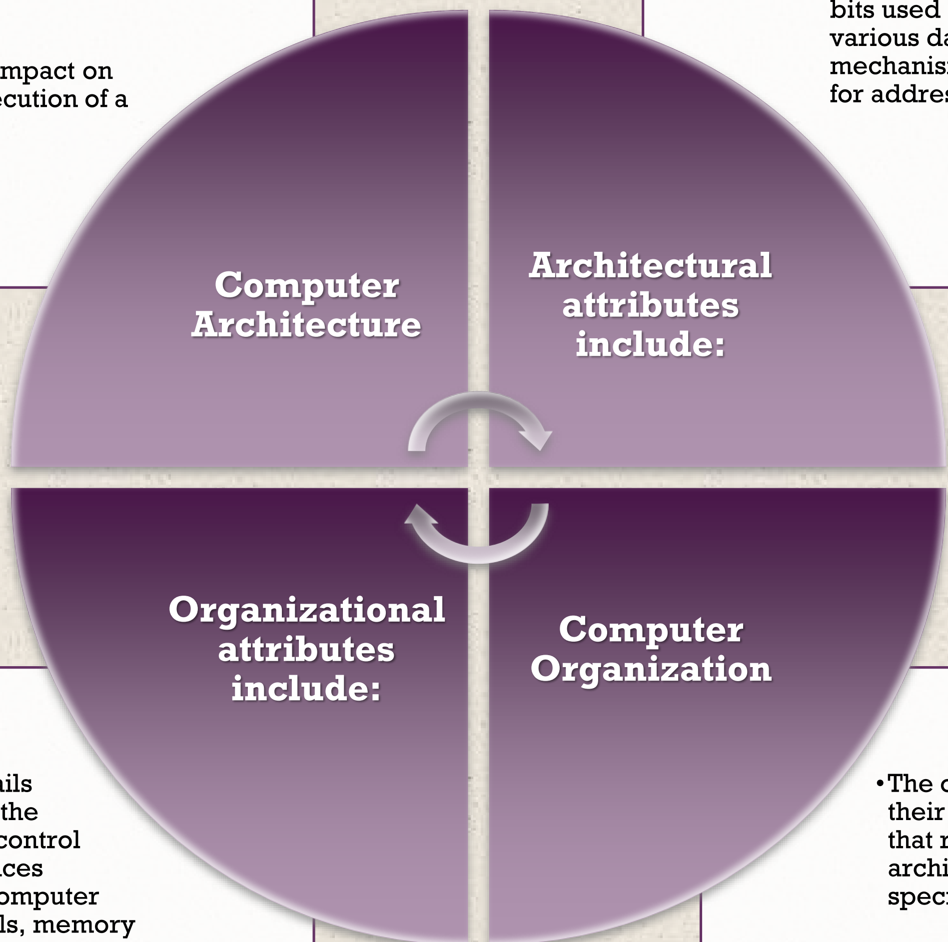
Basic Concepts and Computer Evolution

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



- 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

- There are four basic functions that a computer can perform:
  - Data processing
    - Data may take a wide variety of forms and the range of processing requirements is broad
  - Data storage
    - Short-term
    - Long-term
  - Data movement
    - Input-output (I/O) - when data are received from or delivered to a device (peripheral) that is directly connected to the computer
    - Data communications – when data are moved over longer distances, to or from a remote device
  - Control
    - A control unit manages the computer's resources and orchestrates the performance of its functional parts in response to instructions

Structure

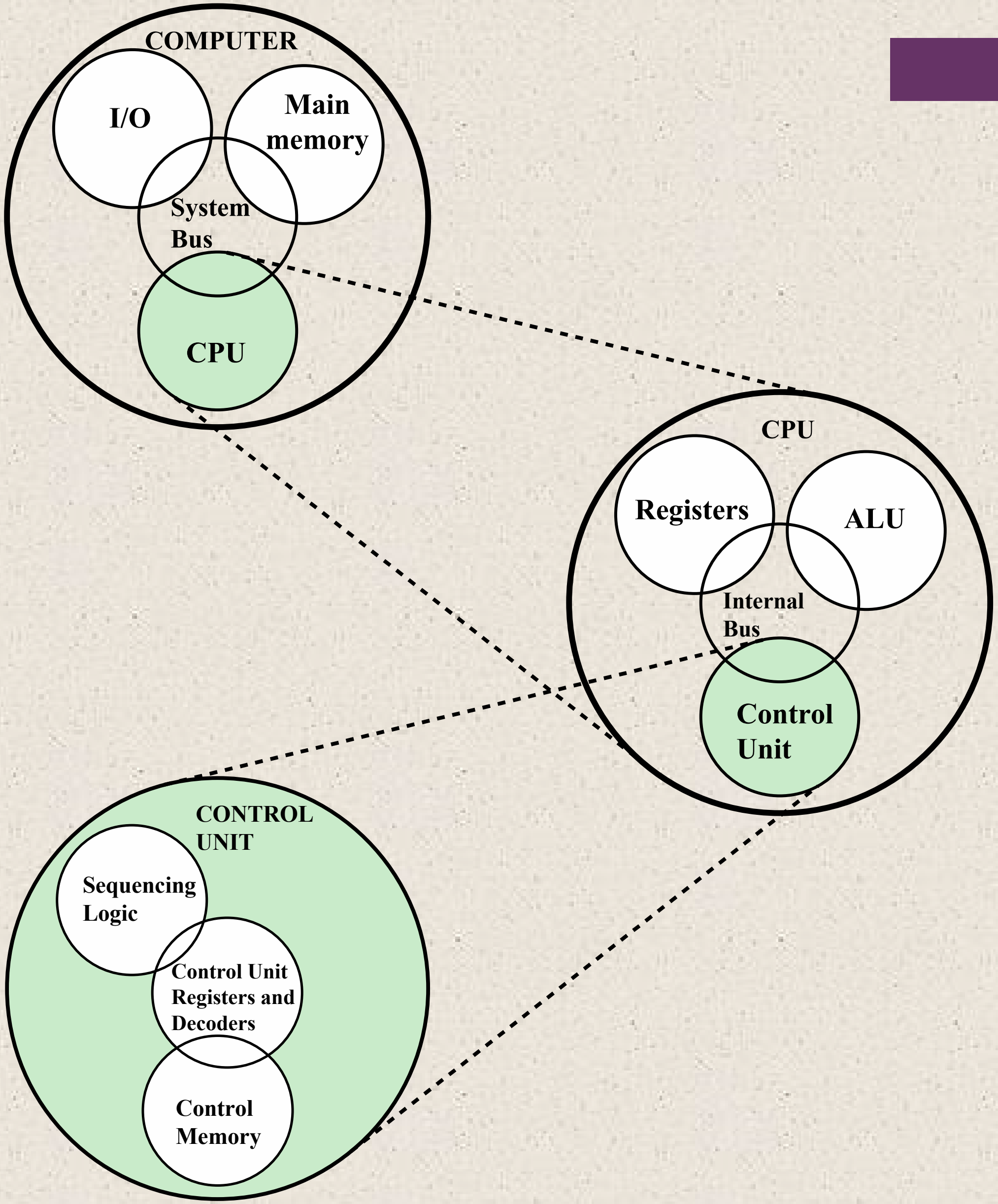
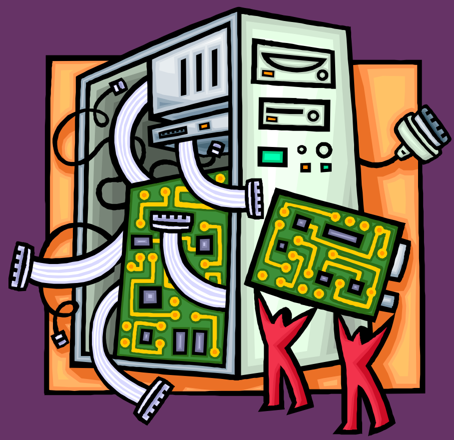


Figure 1.1 A Top-Down View of a Computer

+

There are four main structural components of the computer:

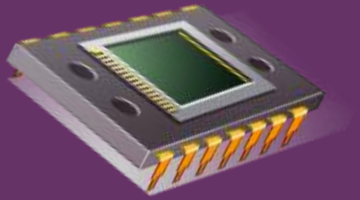


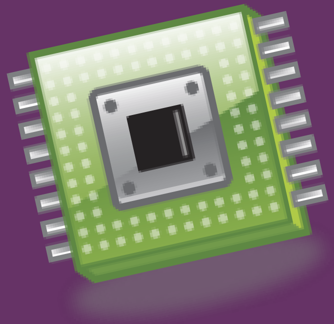
- CPU – controls the operation of the computer and performs its data processing functions
- Main Memory – stores data
- I/O – moves data between the computer and its external environment
- System Interconnection – some mechanism that provides for communication among CPU, main memory, and I/O

+

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

Cache Memory

- Multiple layers of memory between the processor and main memory
- Is smaller and faster than main memory
- Used to speed up memory access by placing in the cache data from main memory that is likely to be used in the near future
- A greater performance improvement may be obtained by using multiple levels of cache, with level 1 (L1) closest to the core and additional levels (L2, L3, etc.) progressively farther from the core

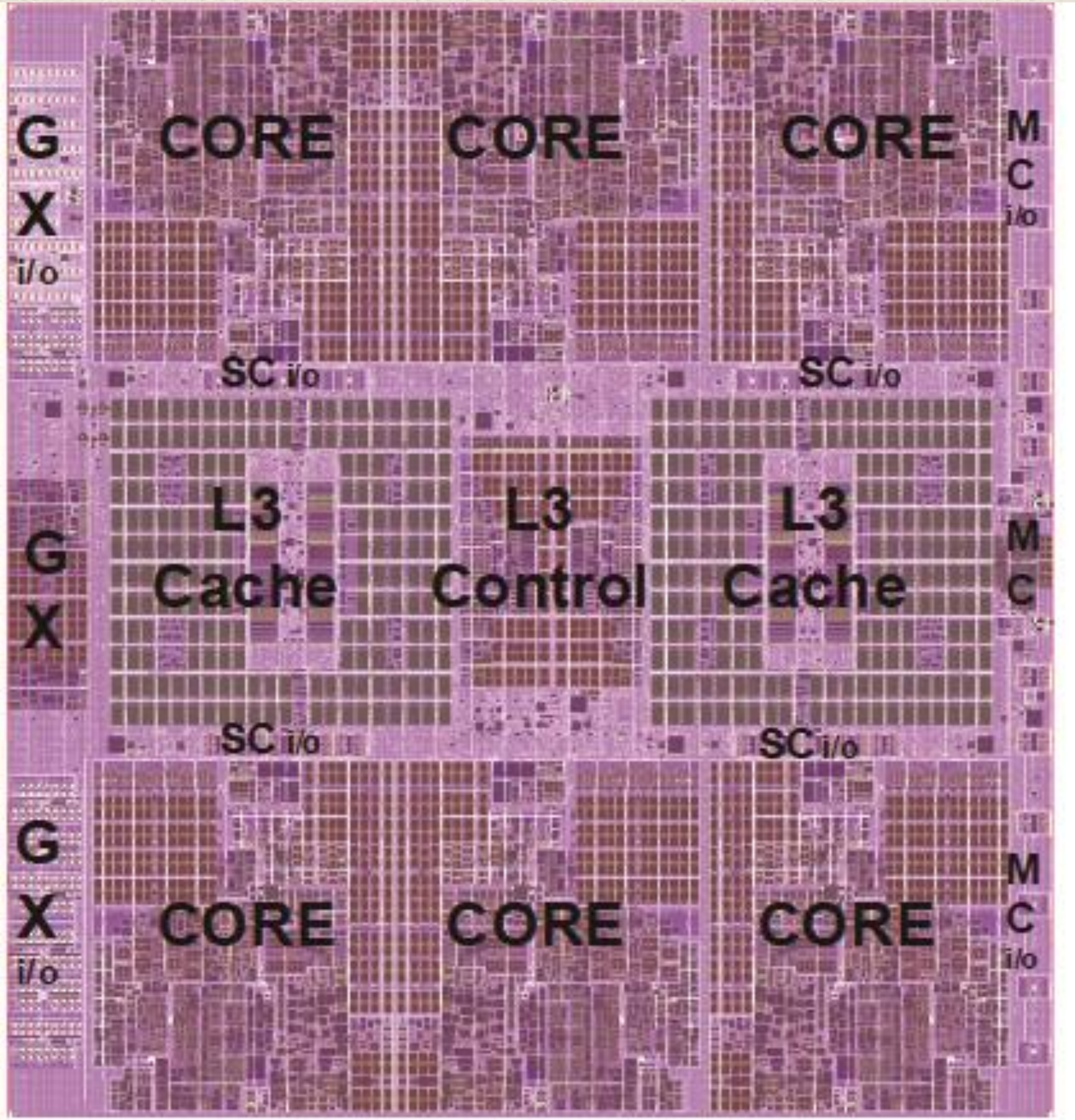


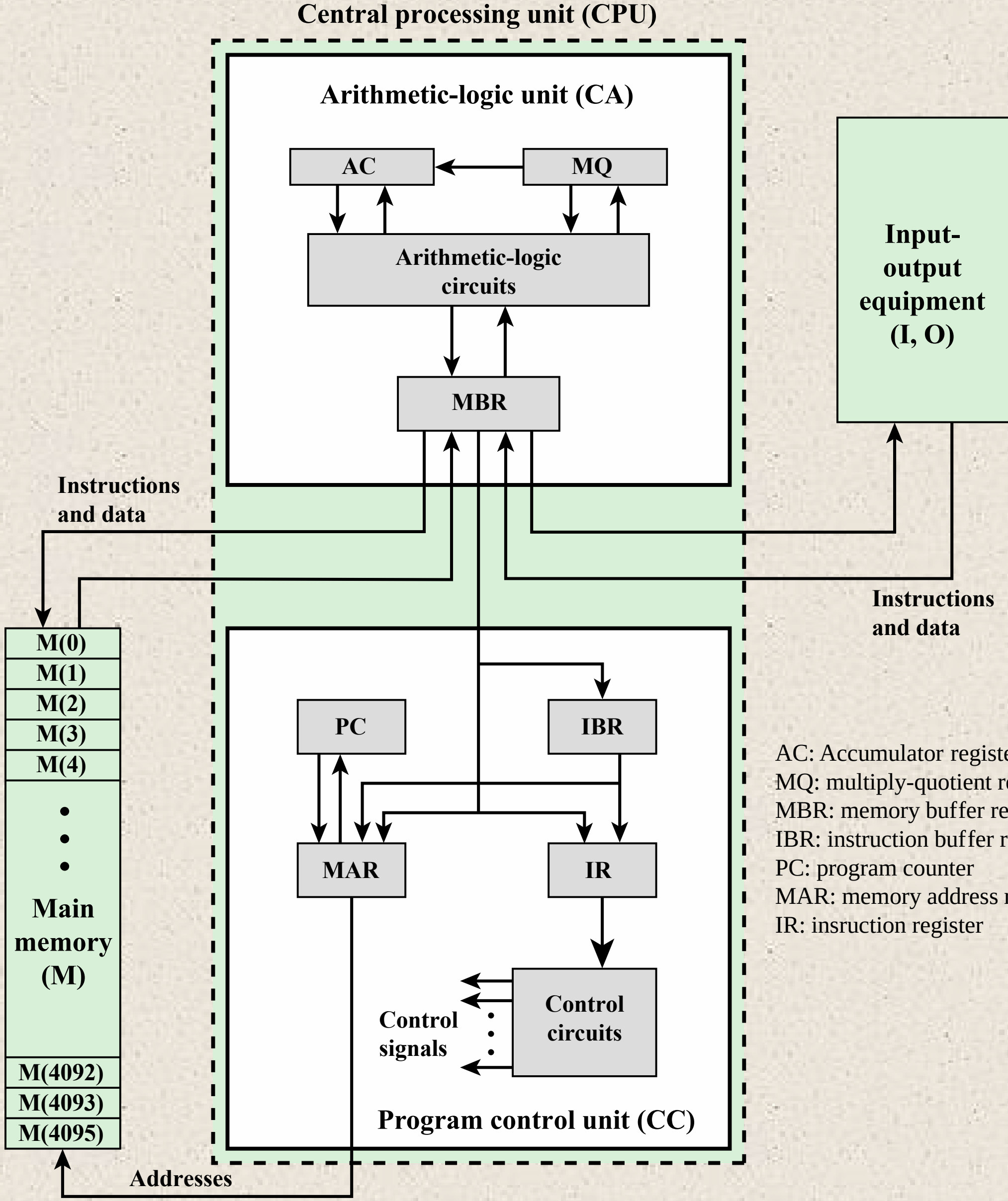
Figure 1.4

zEnterprise
EC12 Processor
Unit (PU)
Chip Diagram

History of Computers

First Generation: Vacuum Tubes

- Vacuum tubes were used for digital logic elements and memory
- IAS computer
  - Fundamental design approach was the stored program concept
    - Attributed to the mathematician John von Neumann
    - First publication of the idea was in 1945 for the EDVAC
  - Design began at the Princeton Institute for Advanced Studies
  - Completed in 1952
  - Prototype of all subsequent general-purpose computers



AC: Accumulator register
MQ: multiply-quotient register
MBR: memory buffer register
IBR: instruction buffer register
PC: program counter
MAR: memory address register
IR: insruction register

Figure 1.6 IAS Structure

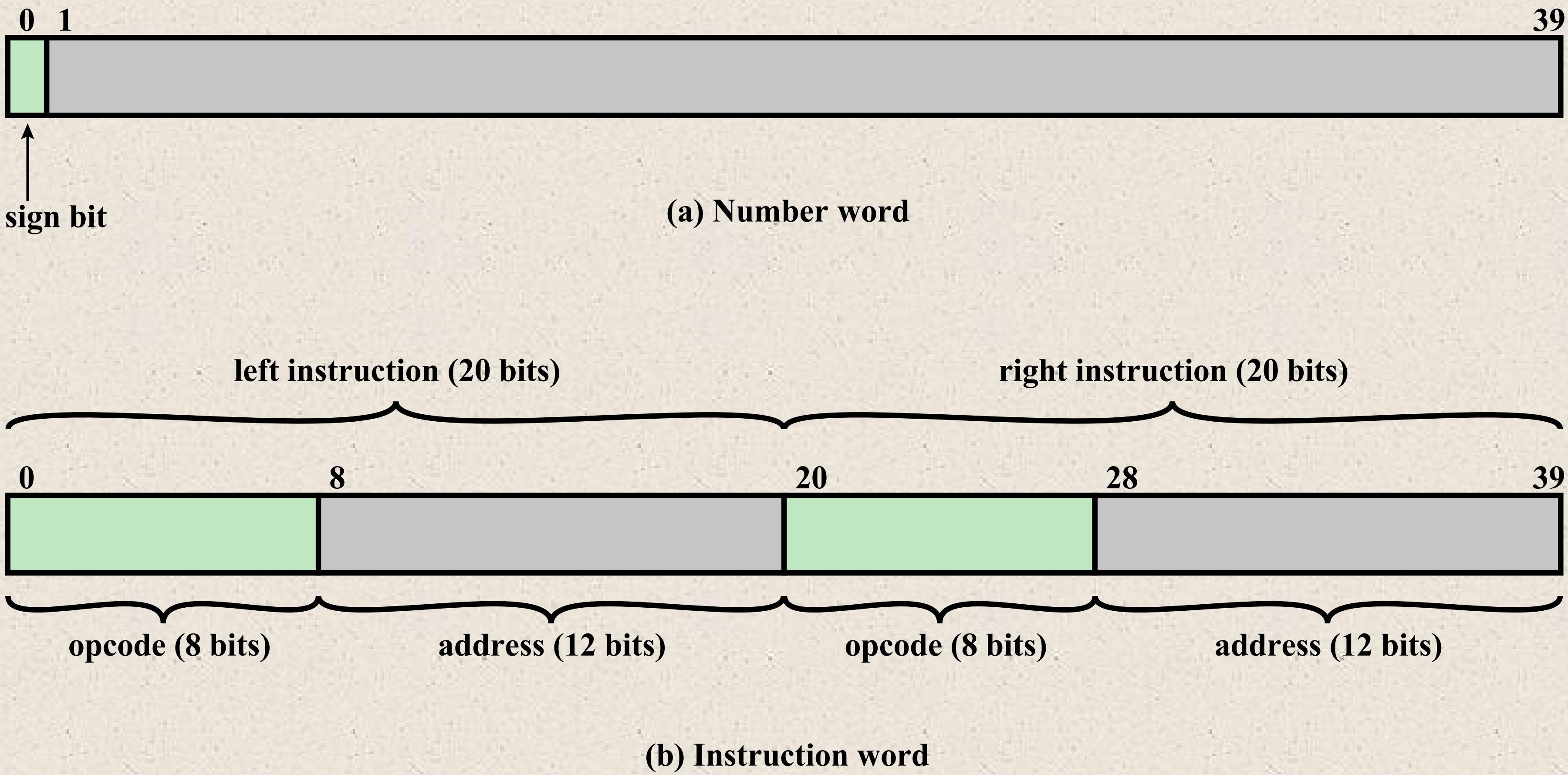


Figure 1.7 IAS Memory Formats

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

Table 1.1

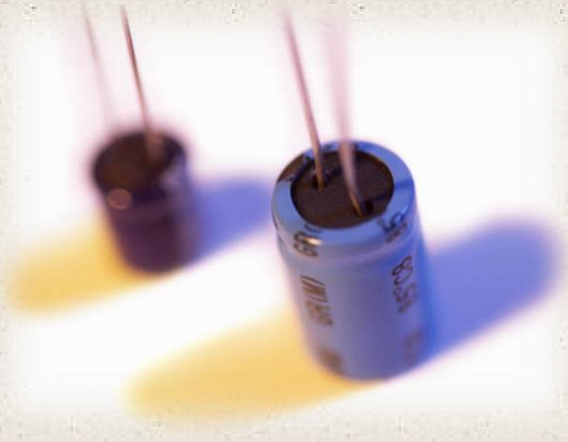
The IAS Instruction Set

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)
		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

(Table can be found on page 17 in the textbook.)

+

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

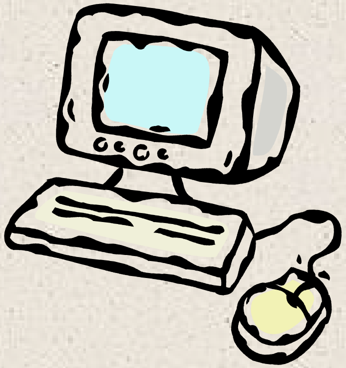
+

Table 1.2 Computer Generations

Generation	Approximate Dates	Technology	Typical Speed (operations per second)
1	1946–1957	Vacuum tube	40,000
2	1957–1964	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

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
    - Libraries perform common computations



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

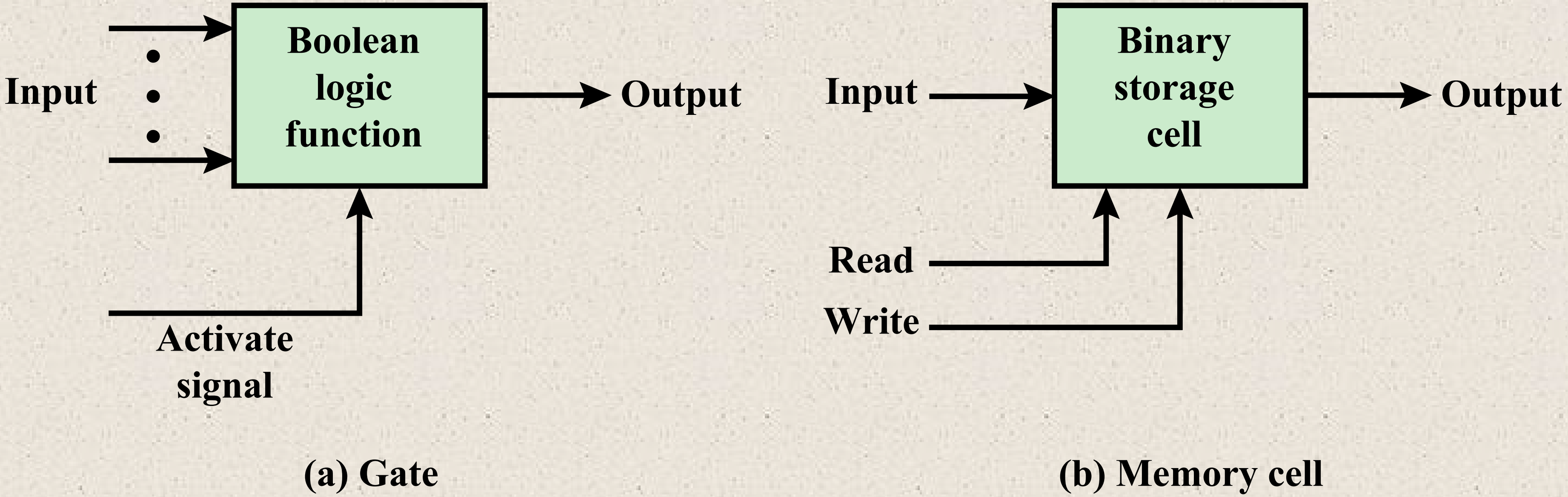


Figure 1.10 Fundamental Computer Elements

+

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

IBM System/360

- Announced in 1964
- Product line was incompatible with older IBM machines
- Was the success of the decade and cemented IBM as the overwhelmingly dominant computer vendor
- The architecture remains to this day the architecture of IBM's mainframe computers
- Was the industry's first planned family of computers
  - Models were compatible in the sense that a program written for one model should be capable of being executed by another model in the series

+ Family Characteristics

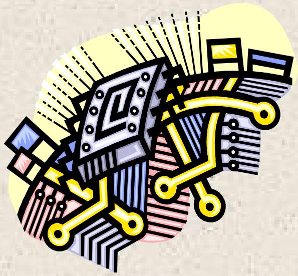
- Similar or identical instruction set
- Similar or identical operating system
- Increasing speed
- Increasing number of I/O ports
- Increasing memory size
- Increasing cost

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

The Evolution of the Intel x86 Architecture

- Two processor families are the Intel x86 and the ARM architectures
- Current x86 offerings represent the results of decades of design effort on complex instruction set computers (CISCs)
- An alternative approach to processor design is the reduced instruction set computer (RISC)
- 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

+
Embedded Systems

- The use of electronics and software within a product
- Billions of computer systems are produced each year that are embedded within larger devices

- Today many devices that use electric power have an embedded computing system

- Often embedded systems are tightly coupled to their environment
  - This can give rise to real-time constraints imposed by the need to interact with the environment
    - Constraints such as required speeds of motion, required precision of measurement, and required time durations, dictate the timing of software operations

  - If multiple activities must be managed simultaneously this imposes more complex real-time constraints

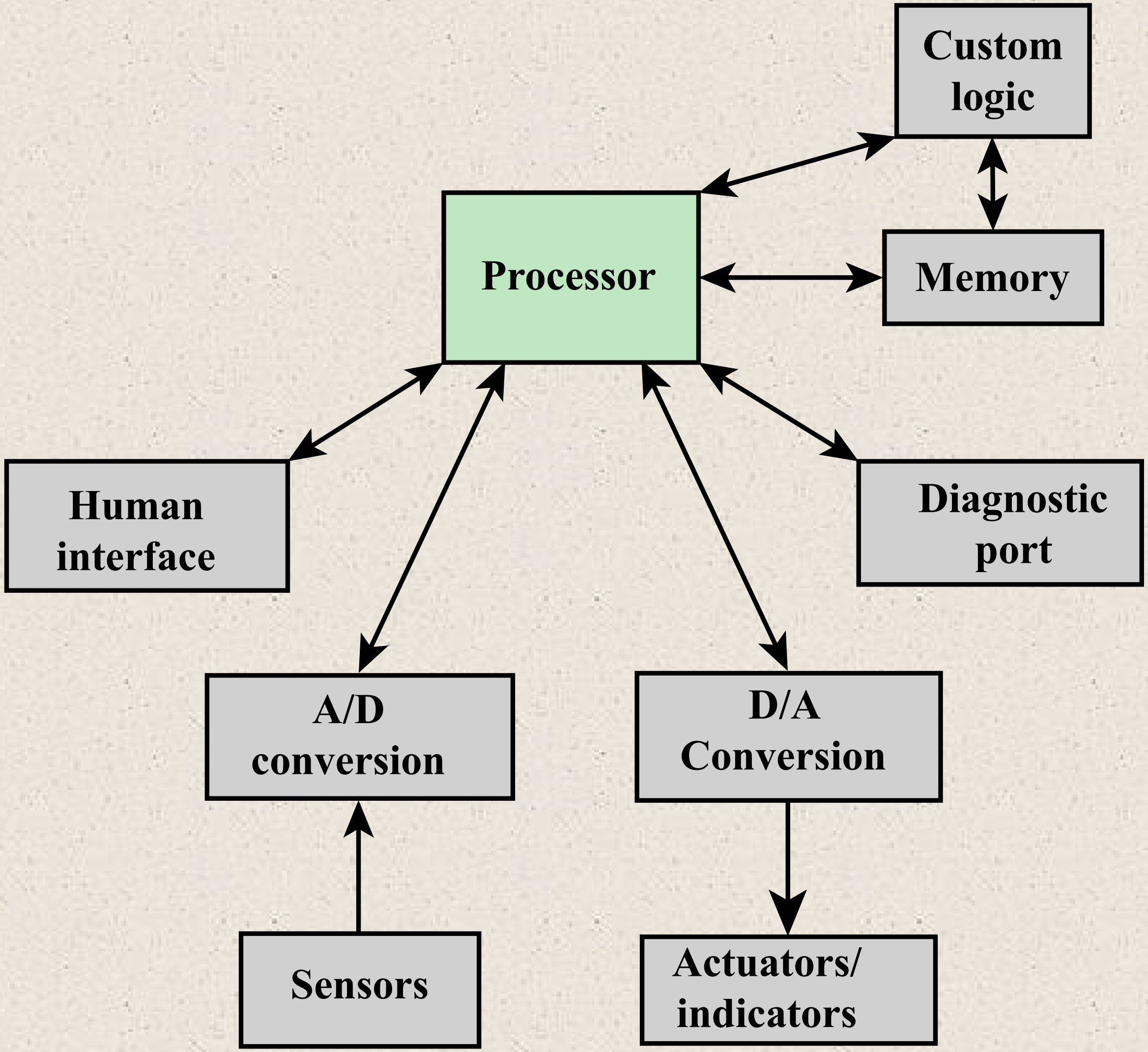


Figure 1.14 Possible Organization of an Embedded System

Embedded Operating Systems

- There are two general approaches to developing an embedded operating system (OS):
  - Take an existing OS and adapt it for the embedded application
  - Design and implement an OS intended solely for embedded use

Application Processors versus Dedicated Processors

- Application processors
  - Defined by the processor's ability to execute complex operating systems
  - General-purpose in nature
  - An example is the smartphone – the embedded system is designed to support numerous apps and perform a wide variety of functions
- Dedicated processor
  - Is dedicated to one or a small number of specific tasks required by the host device
  - Because such an embedded system is dedicated to a specific task or tasks, the processor and associated components can be engineered to reduce size and cost

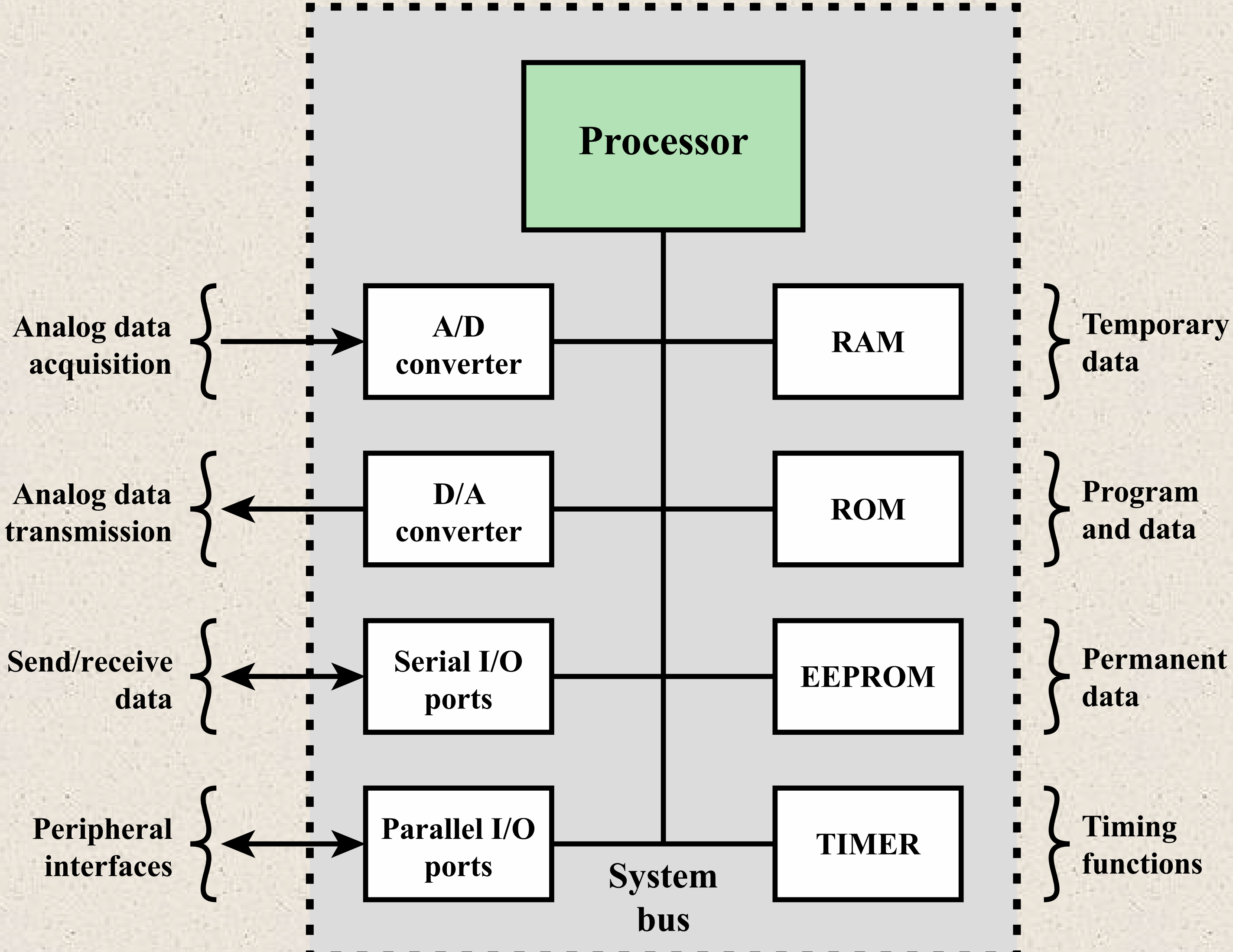


Figure 1.15 Typical Microcontroller Chip Elements

Deeply Embedded Systems

- Subset of embedded systems
- Has a processor whose behavior is difficult to observe both by the programmer and the user
- Uses a microcontroller rather than a microprocessor
- Is not programmable once the program logic for the device has been burned into ROM
- Has no interaction with a user
- Dedicated, single-purpose devices that detect something in the environment, perform a basic level of processing, and then do something with the results
- Often have wireless capability and appear in networked configurations, such as networks of sensors deployed over a large area
- Typically have extreme resource constraints in terms of memory, processor size, time, and power consumption

ARM

- Refers to a processor architecture that has evolved from RISC design principles and is used in embedded systems
- Family of RISC-based microprocessors and microcontrollers designed by ARM Holdings, Cambridge, England
- Chips are high-speed processors that are known for their small die size and low power requirements
- Probably the most widely used embedded processor architecture and indeed the most widely used processor architecture of any kind in the world
- Acorn RISC Machine/Advanced RISC Machine

+

ARM Products

Cortex-A/Cortex-A50

Cortex-R

Cortex-M

- Cortex-M0
- Cortex-M0+
- Cortex-M3
- Cortex-M4

Summary

Chapter 1

Basic Concepts and Computer Evolution

- Organization and architecture
- Structure and function
- Brief history of computers
  - The First Generation: Vacuum tubes
  - The Second Generation: Transistors
  - The Third Generation: Integrated Circuits
  - Later generations
- The evolution of the Intel x86 architecture
- Cloud computing
  - Basic concepts
  - Cloud services
- Embedded systems
  - The Internet of things
  - Embedded operating systems
  - Application processors versus dedicated processors
  - Microprocessors versus microcontrollers
  - Embedded versus deeply embedded systems
- ARM architecture
  - ARM evolution
  - Instruction set architecture
  - ARM products