+

William Stallings
Computer Organization and Architecture
10th Edition

+ Computer Science Curricula 2013 (ACM/IEEE)

Architecture and Organization (AR)

Computing professionals should not regard the computer as just a black box that executes programs by magic. The knowledge area Architecture and Organization builds on Systems Fundamentals (SF) to develop a deeper understanding of the hardware environment upon which all computing is based, and the interface it provides to higher software layers. Students should acquire an understanding and appreciation of a computer system's functional components, their characteristics, performance, and interactions, and, in particular, the challenge of harnessing parallelism to sustain performance improvements now and into the future. Students need to understand computer architecture to develop programs that can achieve high performance through a programmer's awareness of parallelism and latency. In selecting a system to use, students should be able to understand the tradeoff among various components, such as CPU clock speed, cycles per instruction, memory size, and average memory access time.

+

Why study Computer Organization and Architecture?

- Design better programs, including system software such as compilers, operating systems, and device drivers.
- Optimize program behavior.
- Evaluate (benchmark) computer system performance.
- Understand time, space, and price tradeoffs.

Chapter 1

+

Basic Concepts and Computer Evolution

Computer Architecture

Computer Organization

Computer Architecture

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

Architectural attributes include:

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

Organizational attributes include:

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

Computer Organization

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

IBM System 370 Architecture

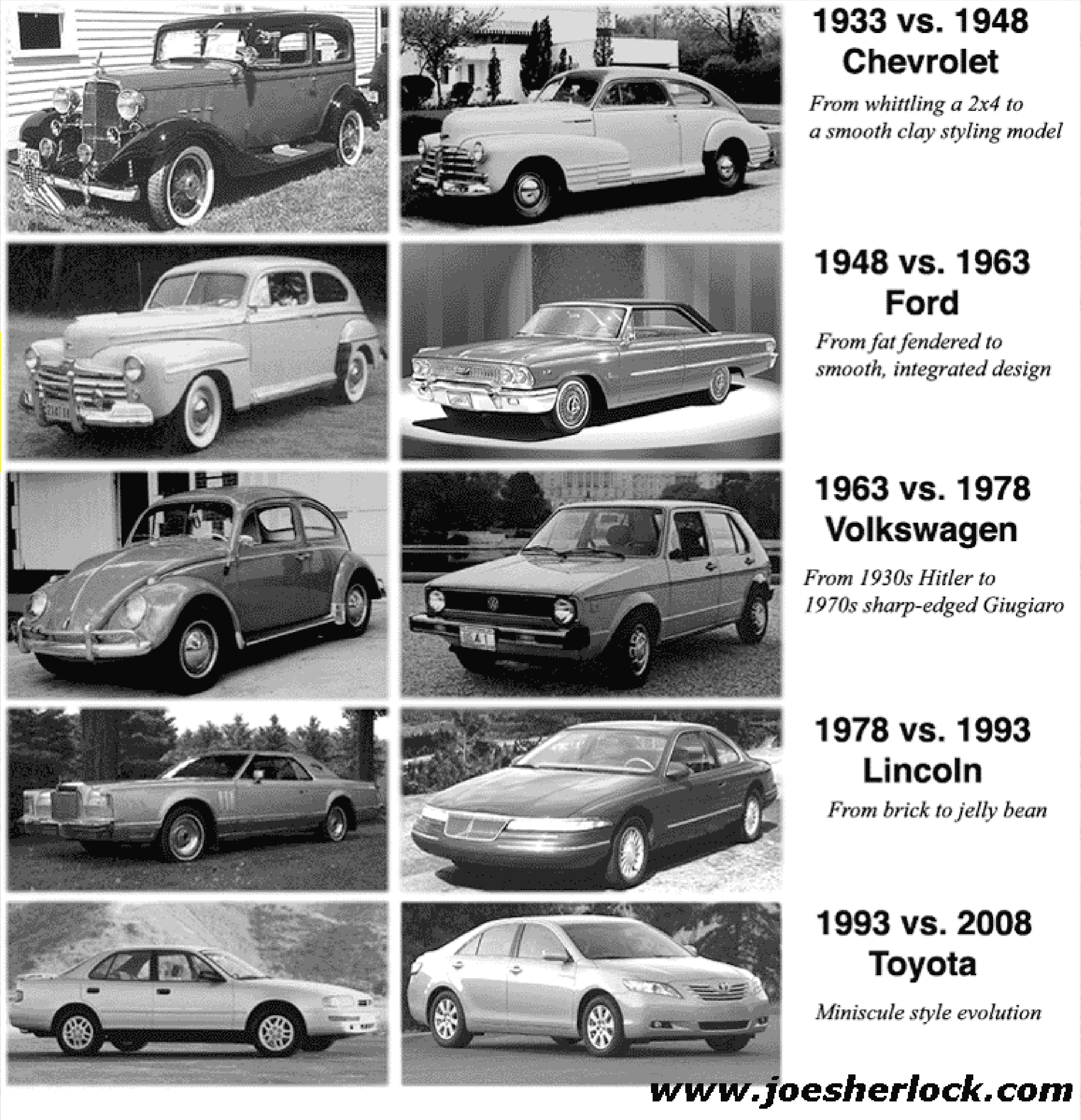
- IBM System/370 architecture
  - Was introduced in 1970
  - Included a number of models
  - Could upgrade to a more expensive, faster model without having to abandon original software
  - New models are introduced with improved technology, but retain the same architecture so that the customer's software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

IBM S/370 (mainframe, year 1970)

IBM zSeries (mainframe, year 2017)

same architecture

differences in organization



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

Program

- Functions are performed through program.
- A sequence of steps.
- For each step, a **computer function** is executed (i.e. either data processing, data storage, data movement OR control)
- For each function, a unique code (OR machine instruction) is provided
  - e.g. ADD, MOVE
- A hardware segment accepts the code and issues the control signals

Structure

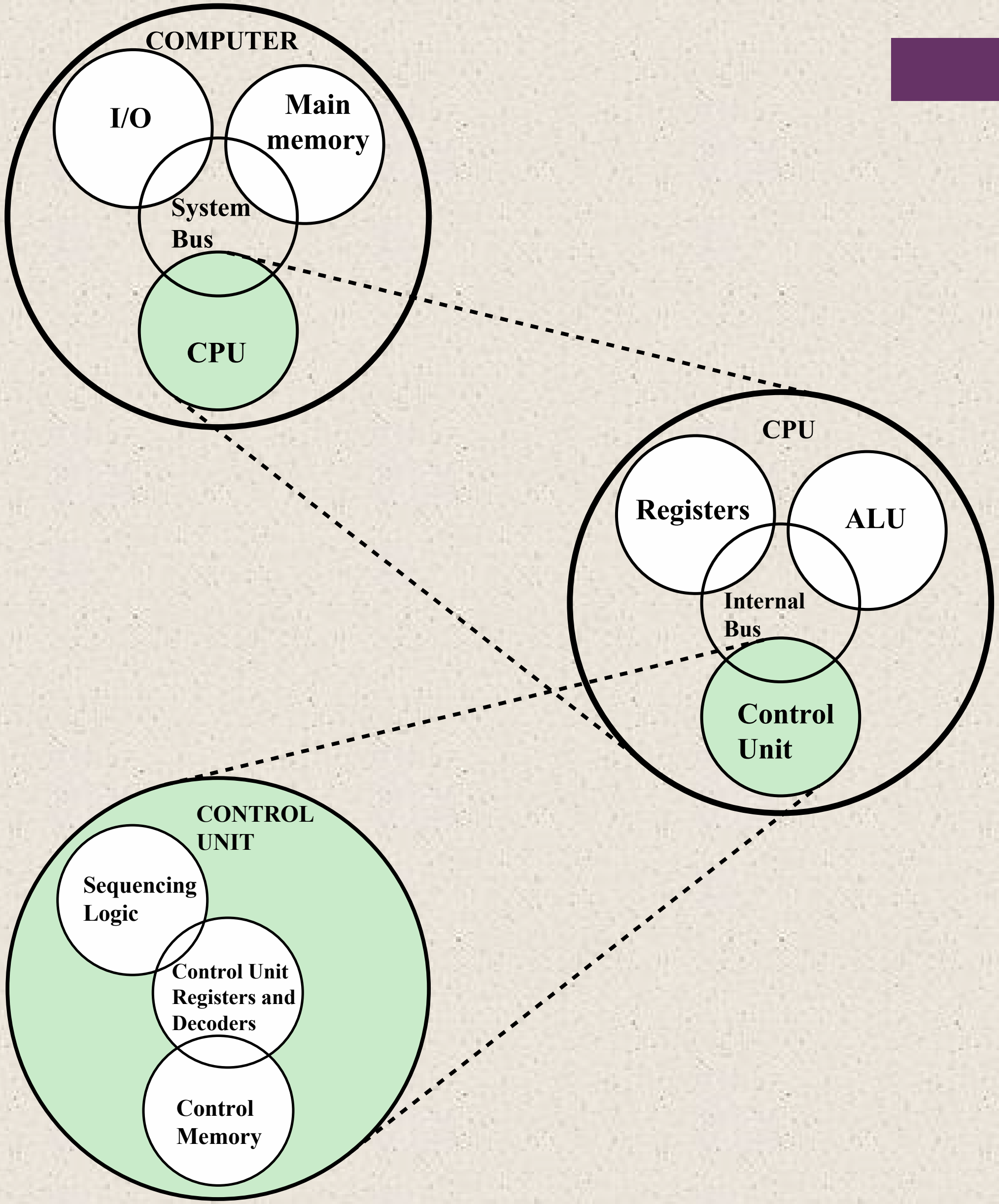
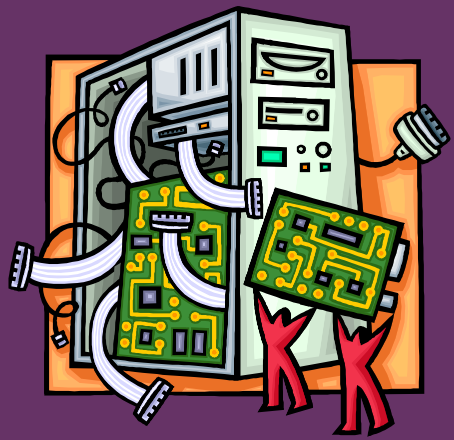


Figure 1.1 A Top-Down View of a Computer

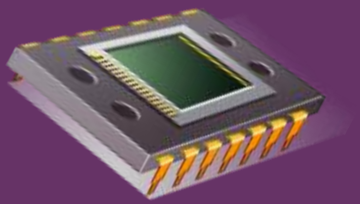
There are four main structural components of the computer:

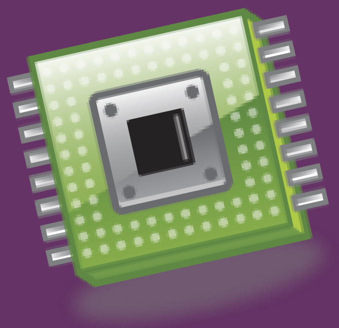


- CPU – controls the operation of the computer and performs its data processing functions
- Main Memory – stores data
- I/O – moves data between the computer and its external environment
- System Interconnection – some mechanism that provides for communication among CPU, main memory, and I/O

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

Multicore Computer Structure

- Central processing unit (CPU)
  - Portion of the computer that fetches and executes instructions
  - Consists of an ALU, a control unit, and registers
  - Referred to as a processor in a system with a single processing unit
- Core
  - An individual processing unit on a processor chip
  - May be equivalent in functionality to a CPU on a single-CPU system
  - Specialized processing units are also referred to as cores
- Processor
  - A physical piece of silicon containing one or more cores
  - Is the computer component that interprets and executes instructions
  - Referred to as a *multicore processor* if it contains multiple cores

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

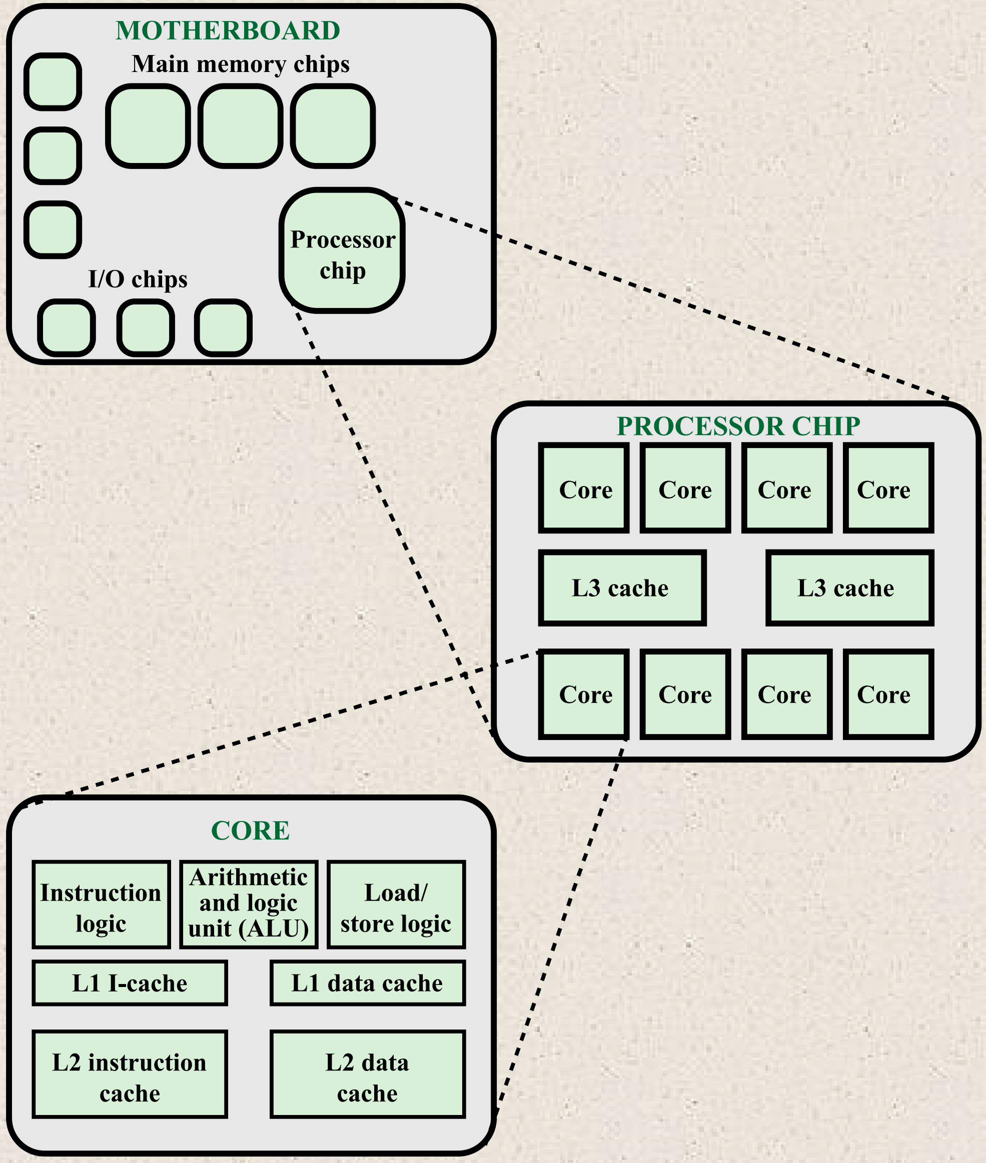


Figure 1.2 Simplified View of Major Elements of a Multicore Computer

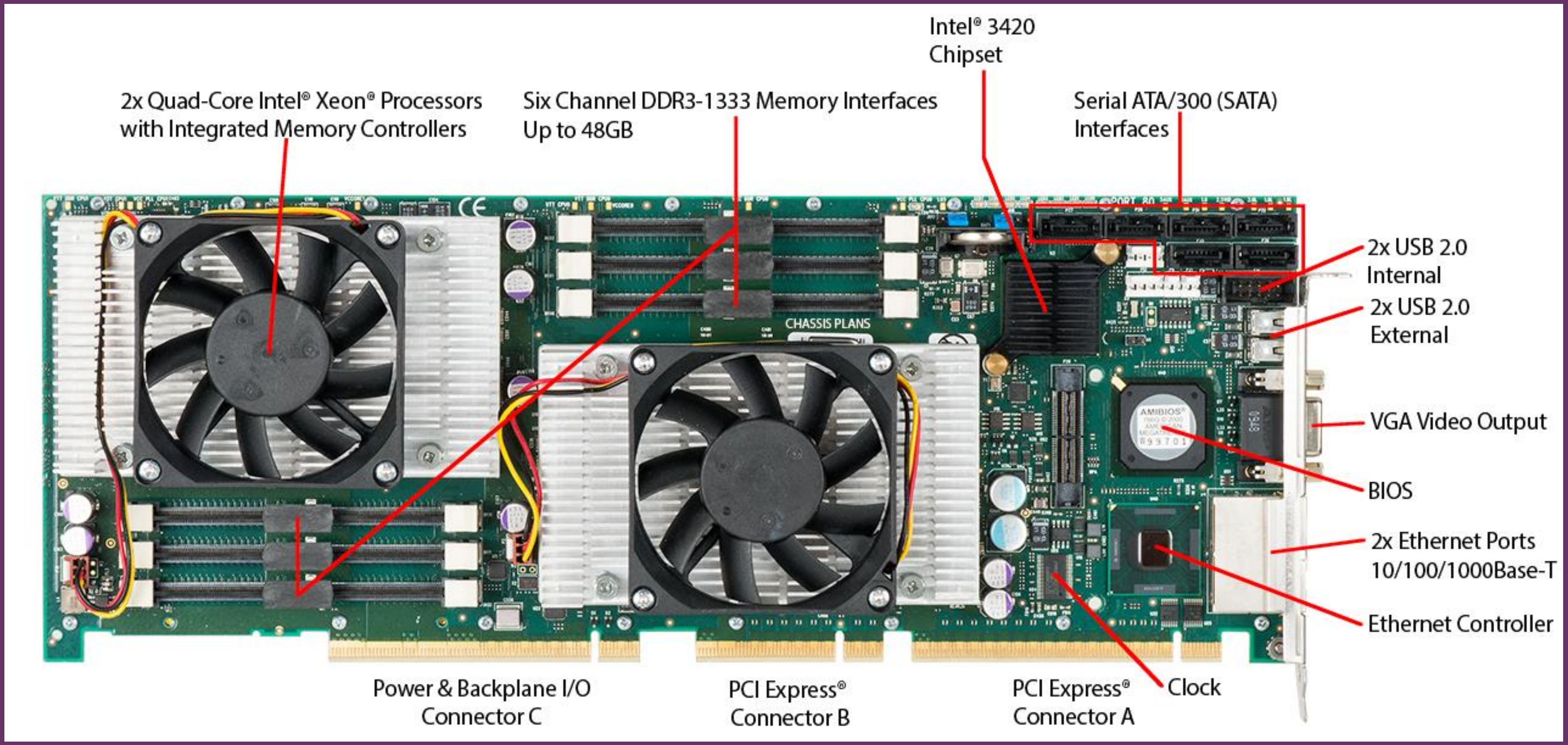


Figure 1.3

Motherboard with Two Intel Quad-Core Xeon Processors

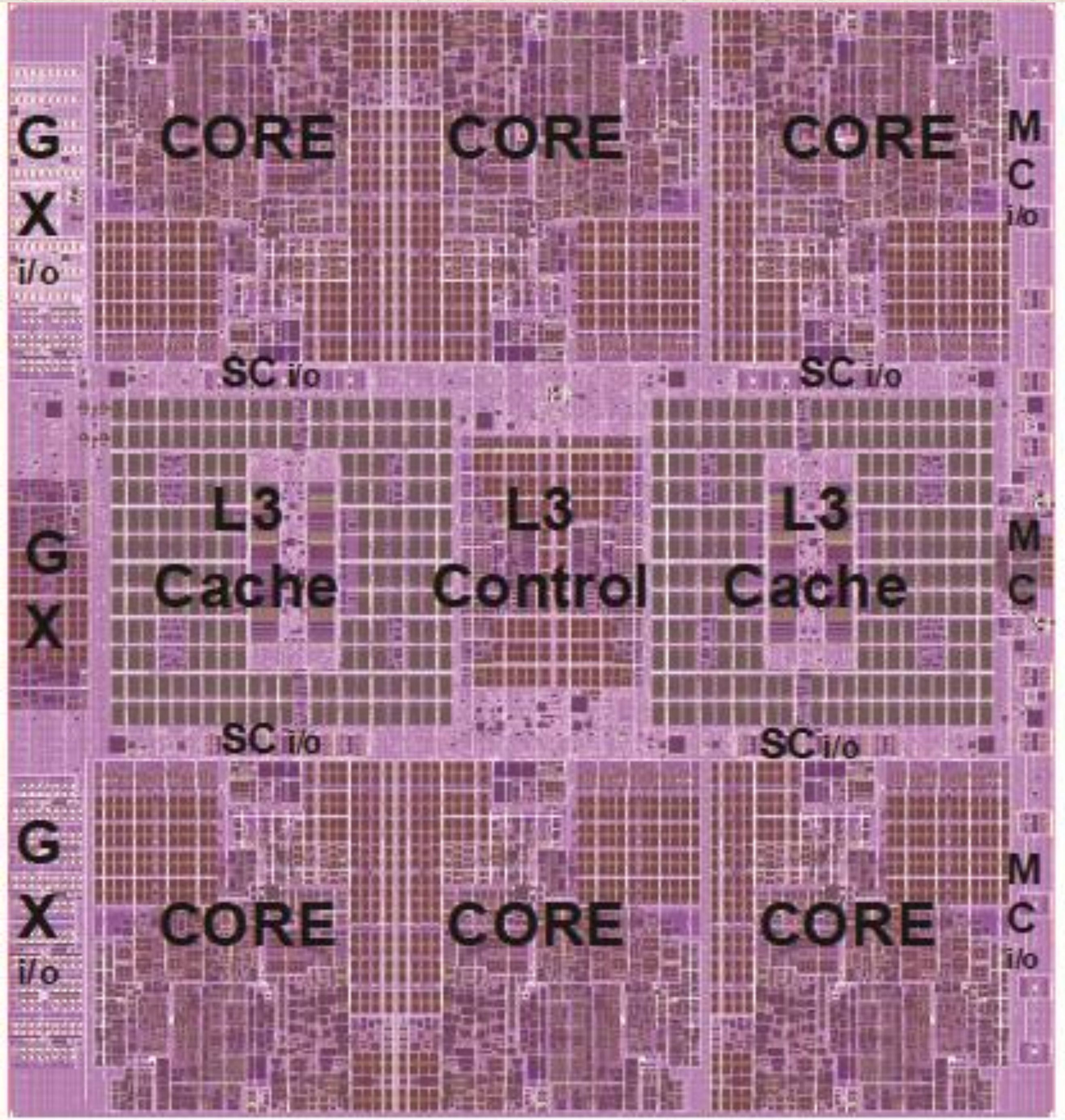


Figure 1.4

zEnterprise EC12 Processor Unit (PU) Chip Diagram

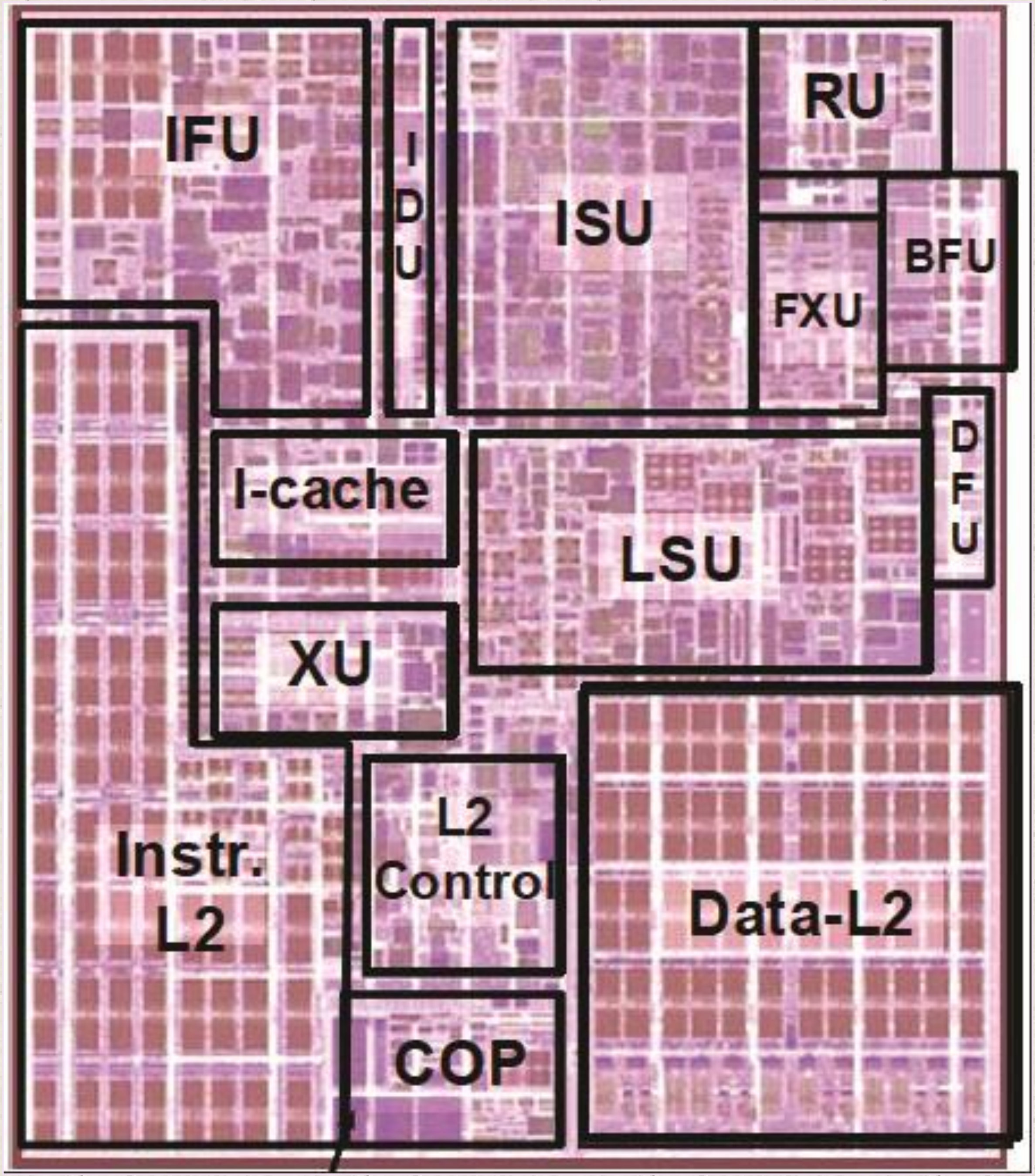


Figure 1.5

zEnterprise EC12 Core Layout

+

An Example System

Reference: Linda Null & julia lobur– the essentials of Computer Organization & Architecture

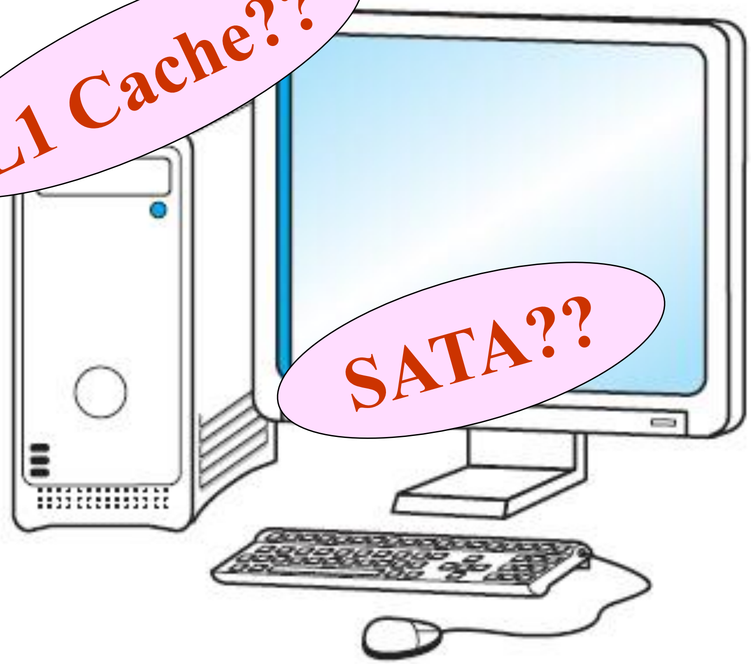
Consider this advertisement:

GHz??

FOR SALE: OBSOLETE COMPUTER – CHEAP! CHEAP! CHEAP!

L1 Cache??

TB??



- Intel i7 Quad Core, 3.9GHz
- 1600MHz 32GB DDR3 SDRAM
- 128KB L1 cache, 2MB L2 cache
- 1TB SATA hard drive (7200 RPM)
- 10 USB ports, 1 serial port, 4 PCI expansion slots (1 PCI, 1 PCIx16, 2 PCIx1), Bluetooth, and HDMI
- 24" widescreen LCD monitor, 16:10 aspect ratio, 1920x1200 WUXGA, 300 cd/m^2, active matrix, 1000:1 (static), 8ms, 24-bit color (16.7 million colors), VGA/DVI input, 2 USB ports
- 16x CD/DVD +/– RW drive
- 1GB PCIe video card
- PCIe sound card
- Integrated 10/100/1000 Ethernet

SATA??

USB??

What does it all mean??

An Example System

Measures of capacity and speed:

- Kilo- (K) = 1 thousand = 10^3 and 2^{10}
- Mega- (M) = 1 million = 10^6 and 2^{20}
- Giga- (G) = 1 billion = 10^9 and 2^{30}
- Tera- (T) = 1 trillion = 10^{12} and 2^{40}
- Peta- (P) = 1 quadrillion = 10^{15} and 2^{50}
- Exa- (E) = 1 quintillion = 10^{18} and 2^{60}
- Zetta- (Z) = 1 sextillion = 10^{21} and 2^{70}
- Yotta- (Y) = 1 septillion = 10^{24} and 2^{80}

Whether a metric refers to a power of ten or a power of two typically depends upon what is being measured.

+ An Example System

From Digital Logic

- Hertz = clock cycles per second (frequency)
  - 1MHz = 1,000,000Hz
  - Processor speeds are measured in MHz or GHz.
- Byte = a unit of storage
  - 1KB = 2^{10} = 1024 Bytes
  - 1MB = 2^{20} = 1,048,576 Bytes
  - Main memory (RAM) is measured in GB
  - Disk storage is measured in GB for small systems, TB for large systems.

An Example System

Measures of time and space:

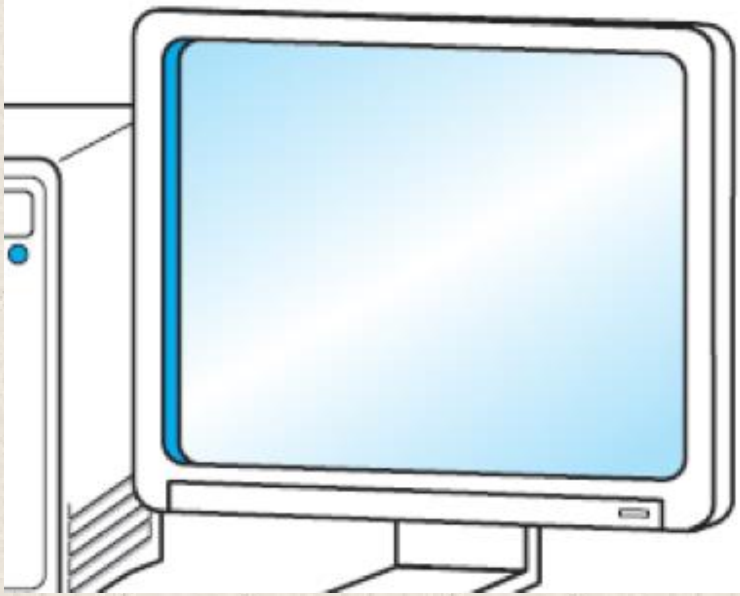
- Milli- (m) = 1 thousandth = 10^{-3}
- Micro- (μ) = 1 millionth = 10^{-6}
- Nano- (n) = 1 billionth = 10^{-9}
- Pico- (p) = 1 trillionth = 10^{-12}
- Femto- (f) = 1 quadrillionth = 10^{-15}
- Atto- (a) = 1 quintillionth = 10^{-18}
- Zepto- (z) = 1 sextillionth = 10^{-21}
- Yocto- (y) = 1 septillionth = 10^{-24}

An Example System

- Millisecond = 1 thousandth of a second
  - Hard disk drive access times are often 10 to 20 milliseconds.
- Nanosecond = 1 billionth of a second
  - Main memory access times are often 50 to 70 nanoseconds.
- Micron (micrometer) = 1 millionth of a meter
  - Circuits on computer chips are measured in microns.

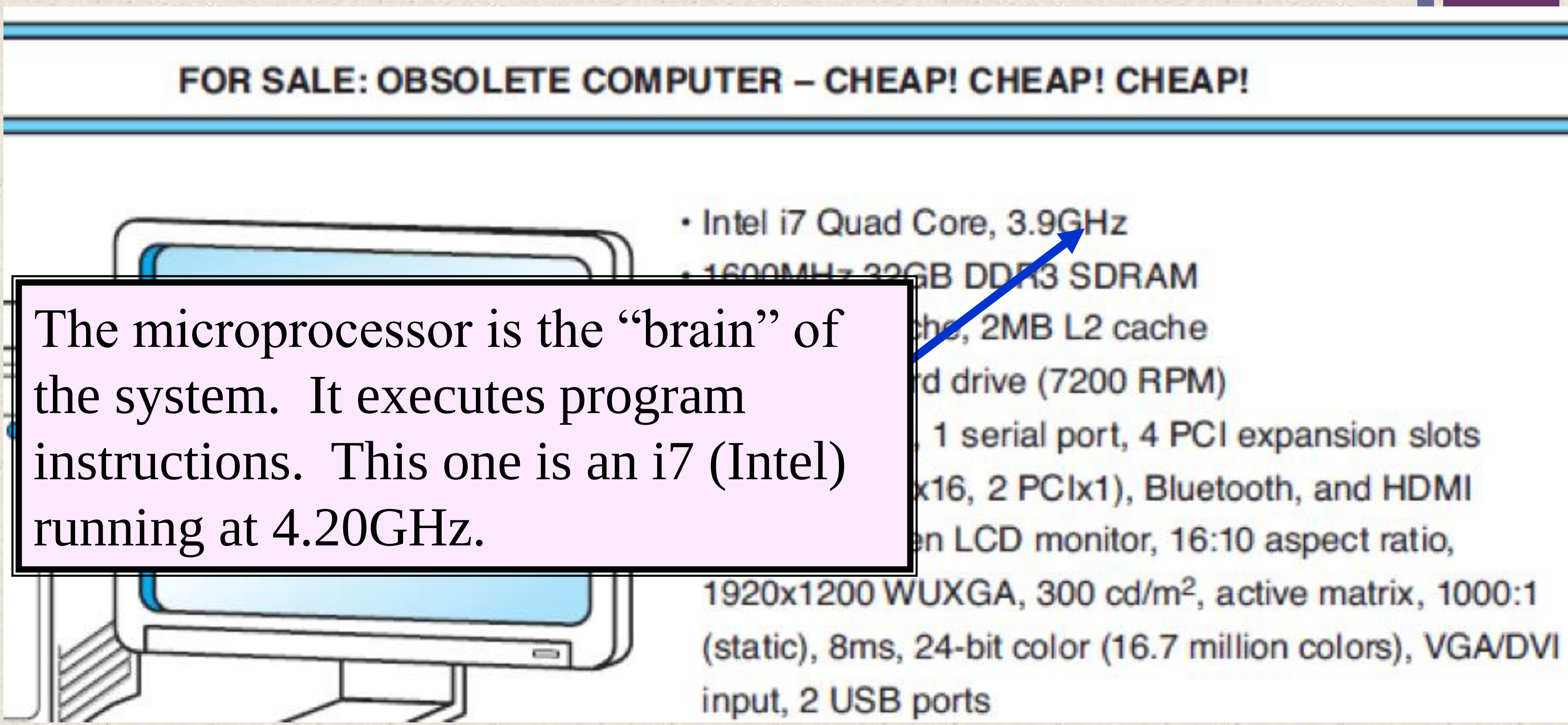
An Example System

FOR SALE: OBSOLETE COMPUTER – CHEAP! CHEAP! CHEAP!



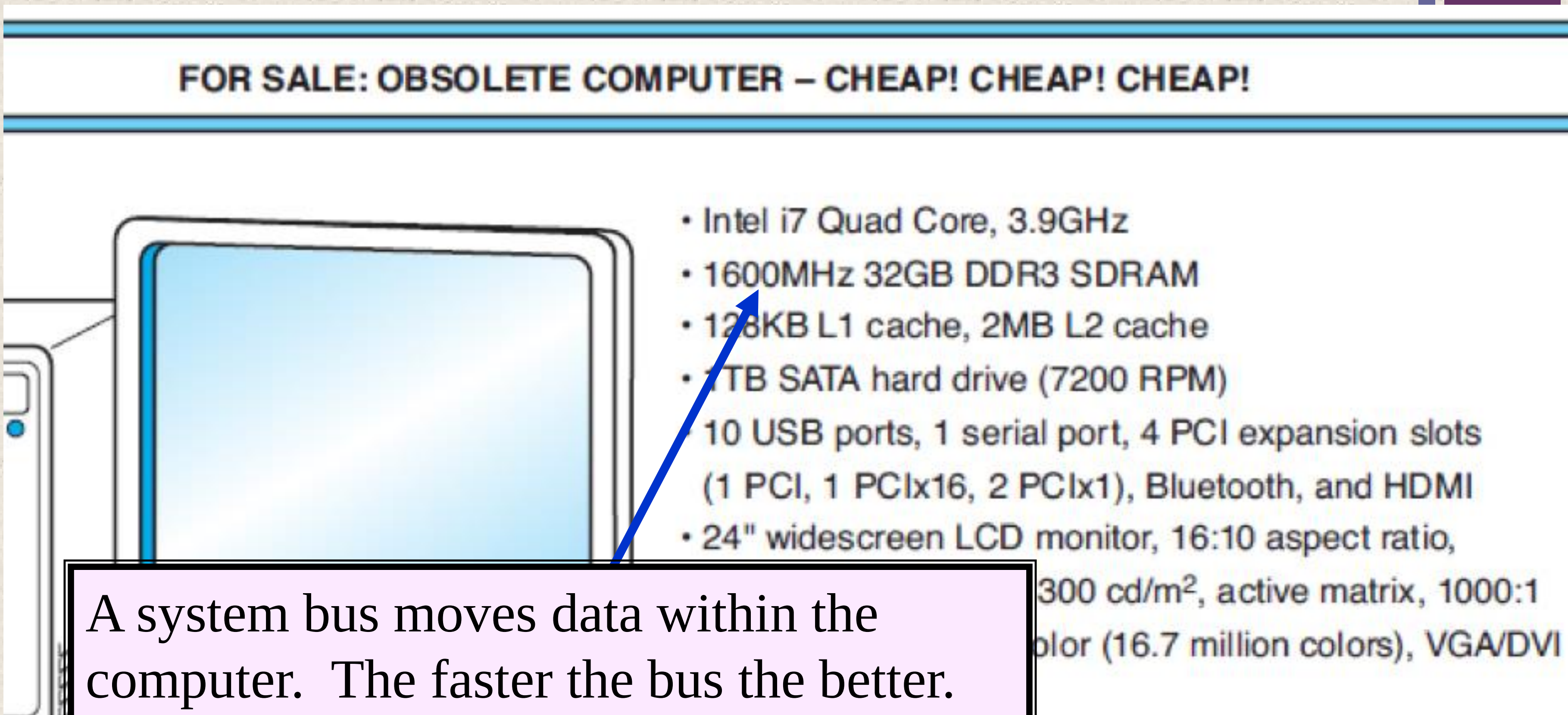
- Intel i7 Quad Core, 3.9GHz
- 1600MHz 32GB DDR3 SDRAM
- 128KB L1 cache, 2MB L2 cache
- 1TB SATA hard drive (7200 RPM)
- 10 USB ports, 1 serial port, 4 PCI expansion slots (1 PCI, 1 PCIx16, 2 PCIx1), Bluetooth, and HDMI
- 24" widescreen LCD monitor, 16:10 aspect ratio, 1920x1200 WUXGA, 300 cd/m^2, active matrix, 1000:1 (static), 8ms, 24-bit color (16.7 million colors), VGA/DVI input, 2 USB ports

+
An Example System



The microprocessor is the “brain” of the system. It executes program instructions. This one is an i7 (Intel) running at 4.20GHz.

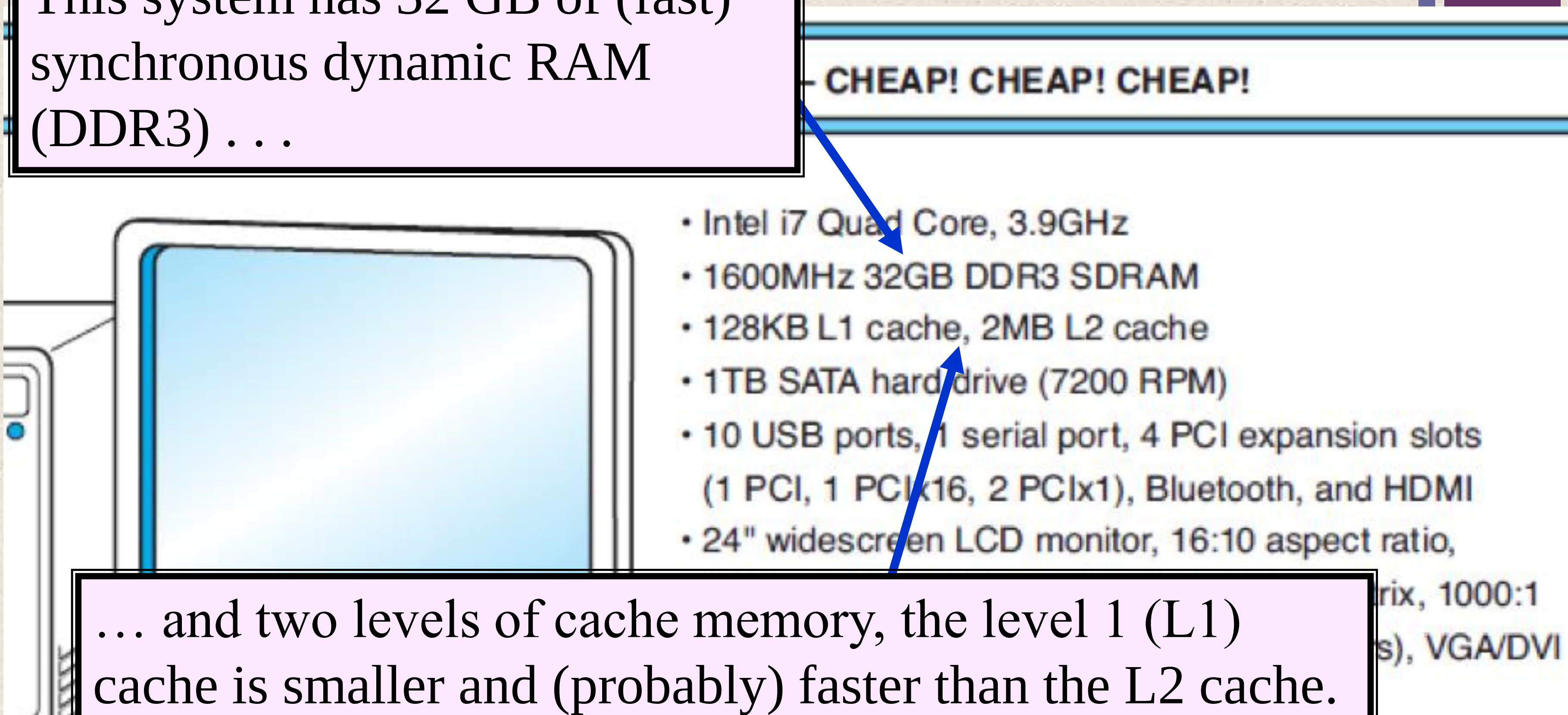
An Example System



A system bus moves data within the computer. The faster the bus the better. This one runs at 1600MHz.

An Example System

This system has 32 GB of (fast) synchronous dynamic RAM (DDR3) . . .



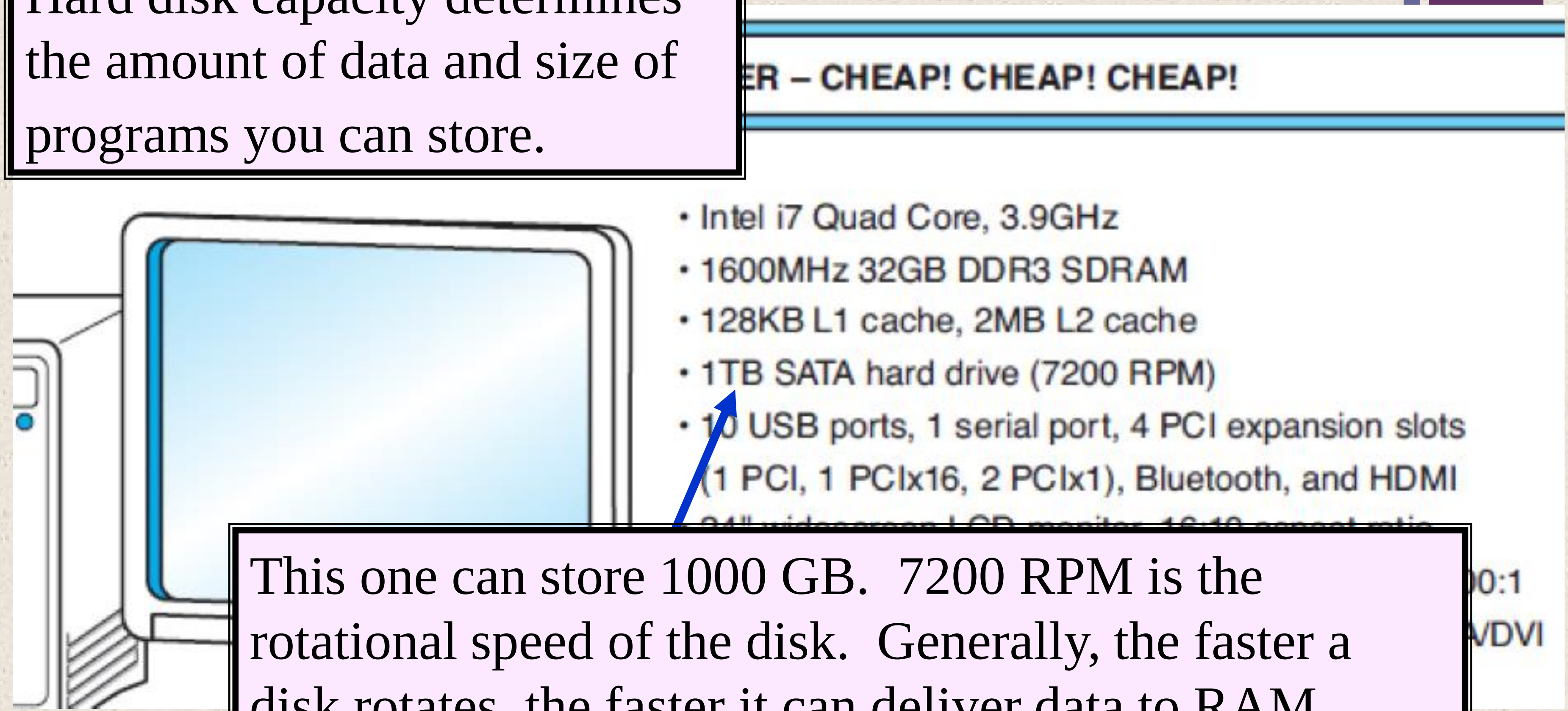
… and two levels of cache memory, the level 1 (L1) cache is smaller and (probably) faster than the L2 cache. Note that these cache sizes are measured in KB & MB.

+
An Example System

- Computers with large main memory capacity can run larger programs with greater speed than computers having small memories.
- RAM is an acronym for *random access memory*. Random access means that memory contents can be accessed directly if you know its location.
- Cache is a type of temporary memory that can be accessed faster than RAM.

An Example System

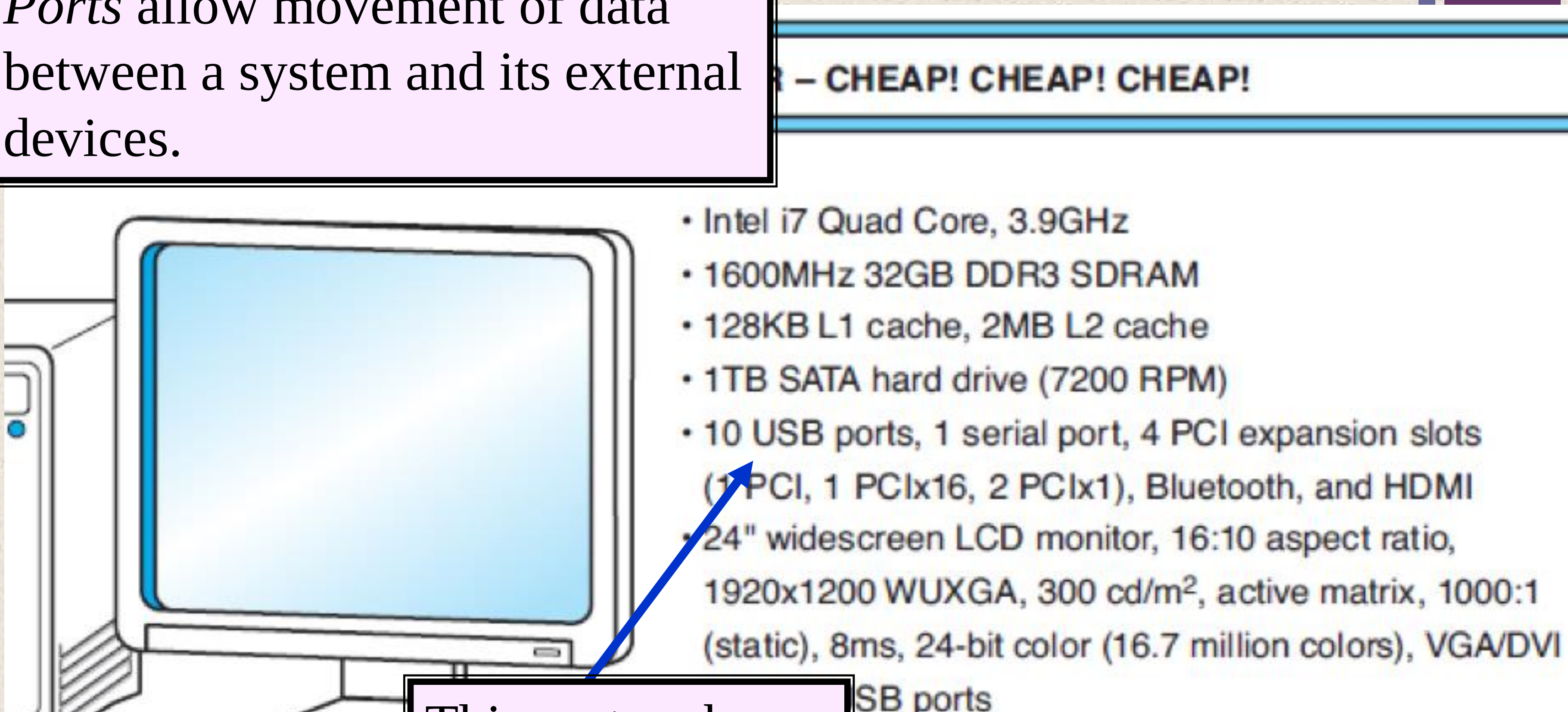
Hard disk capacity determines the amount of data and size of programs you can store.



This one can store 1000 GB. 7200 RPM is the rotational speed of the disk. Generally, the faster a disk rotates, the faster it can deliver data to RAM. (There are many other factors involved.)

An Example System

Ports allow movement of data between a system and its external devices.



This system has fifteen (15) ports.

+

An Example System

- Serial ports send data as a series of pulses along one or two data lines.
- Parallel ports send data as a single pulse along at least eight data lines.
- USB, Universal Serial Bus, is an intelligent serial interface that is self-configuring. (It supports “plug and play.”)

Recent One … mid-range level

iMac

Features | Design | Performance | OS X | Built-in Apps | Tech Specs | Buy Now

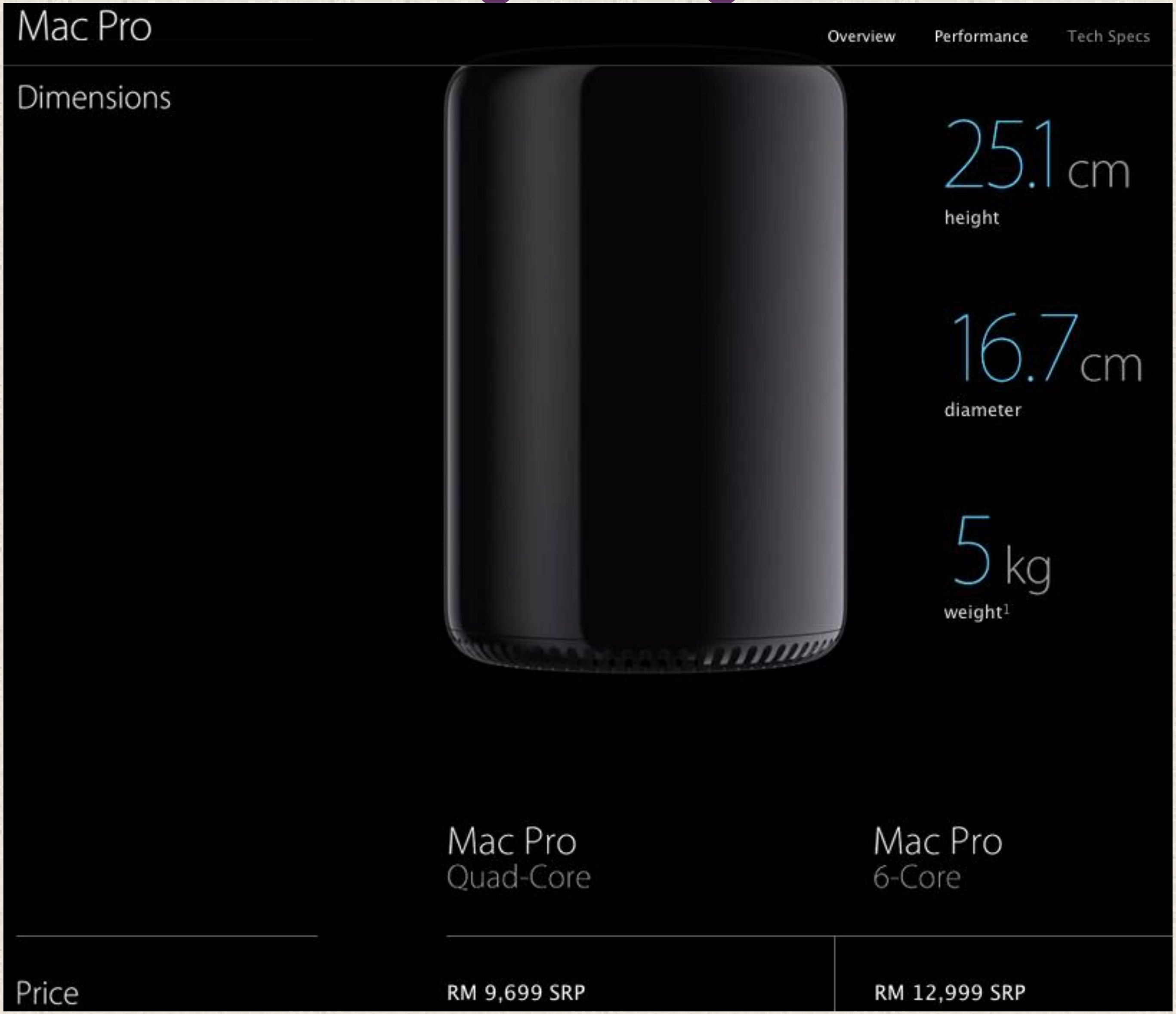
	21.5-inch iMac	27-inch iMac
Display	 21.5-inch (diagonal) LED-backlit display with IPS technology; 1920-by-1080 resolution with support for millions of colours	 27-inch (diagonal) LED-backlit display with IPS technology; 2560-by-1440 resolution with support for millions of colours

Price	RM 3,499 SRP	RM 4,299 SRP	RM 4,999 SRP	RM 5,999 SRP	RM 6,699 SRP

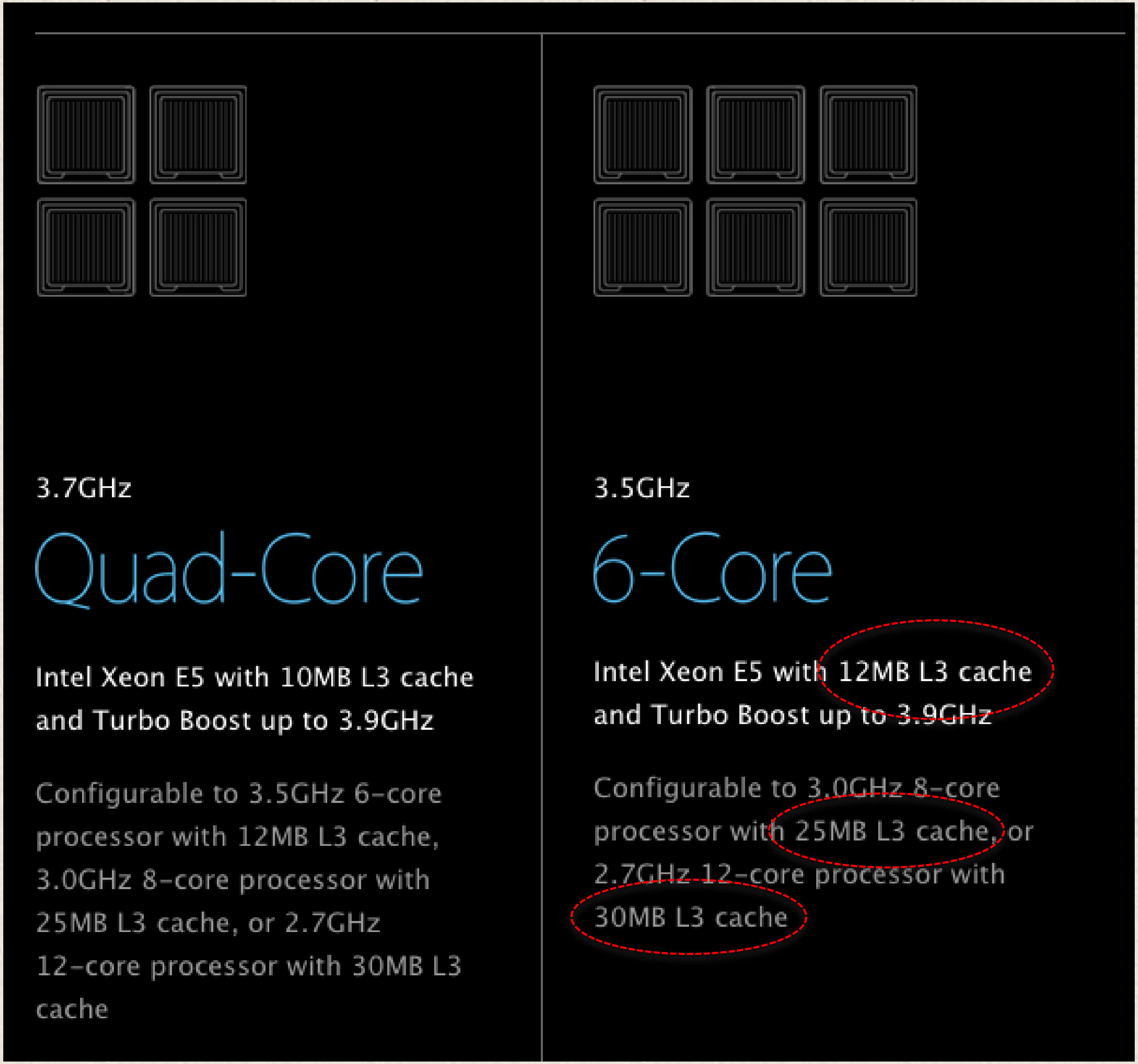
+ Recent One … mid-range level

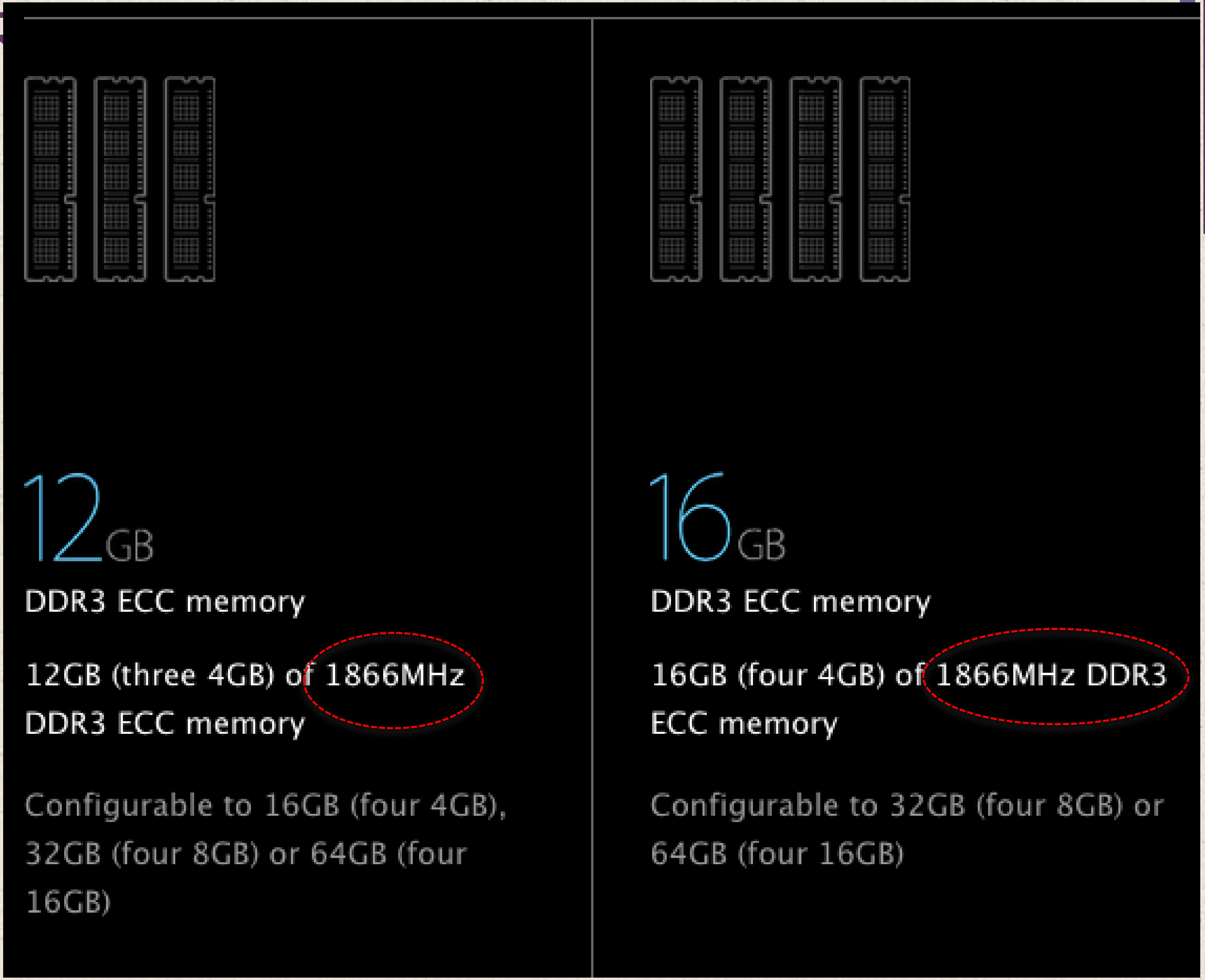
Processor	**1.4GHz** dual-core Intel Core i5 processor (Turbo Boost up to 2.7GHz) with 3MB shared L3 cache	**2.7GHz** quad-core Intel Core i5 processor (Turbo Boost up to 3.2GHz) with 4MB L3 cache	**2.9GHz** quad-core Intel Core i5 processor (Turbo Boost up to 3.6GHz) with 6MB L3 cache Configurable to 3.1GHz quad-core Intel Core i7 (Turbo Boost up to 3.9GHz).	**3.2GHz** quad-core Intel Core i5 processor (Turbo Boost up to 3.6GHz) with 6MB L3 cache	**3.4GHz** quad-core Intel Core i5 processor (Turbo Boost up to 3.8GHz) with 6MB L3 cache Configurable to 3.5GHz quad-core Intel Core i7 (Turbo Boost up to 3.9GHz).
Memory	8GB of 1600MHz LPDDR3 onboard memory	8GB (two 4GB) of 1600MHz DDR3 memory Configurable to 16GB.		8GB (two 4GB) of 1600MHz DDR3 memory; four user-accessible SO-DIMM slots Configurable to 16GB or 32GB.	
Storage[1]	500 GB 500GB (5400-rpm) hard drive Configurable to 1TB hard drive, 1TB Fusion Drive, or 256GB of flash storage.	1 TB 1TB (5400-rpm) hard drive Configurable to 1TB Fusion Drive or 256GB or 512GB of flash storage.		1 TB 1TB (7200-rpm) hard drive Configurable to 3TB hard drive, 1TB or 3TB Fusion Drive, or 256GB, 512GB, or 1TB of flash storage.	

+ Recent One … high-range level



+ Re


3.7GHz
Quad-Core
Intel Xeon E5 with 10MB L3 cache and Turbo Boost up to 3.9GHz
Configurable to 3.5GHz 6-core processor with 12MB L3 cache, 3.0GHz 8-core processor with 25MB L3 cache, or 2.7GHz 12-core processor with 30MB L3 cache
3.5GHz
6-Core
Intel Xeon E5 with 12MB L3 cache and Turbo Boost up to 3.9GHz
Configurable to 3.0GHz 8-core processor with 25MB L3 cache, or 2.7GHz 12-core processor with 30MB L3 cache


12GB
DDR3 ECC memory
12GB (three 4GB) of 1866MHz DDR3 ECC memory
Configurable to 16GB (four 4GB), 32GB (four 8GB) or 64GB (four 16GB)
16GB
DDR3 ECC memory
16GB (four 4GB) of 1866MHz DDR3 ECC memory
Configurable to 32GB (four 8GB) or 64GB (four 16GB)

Class Activity

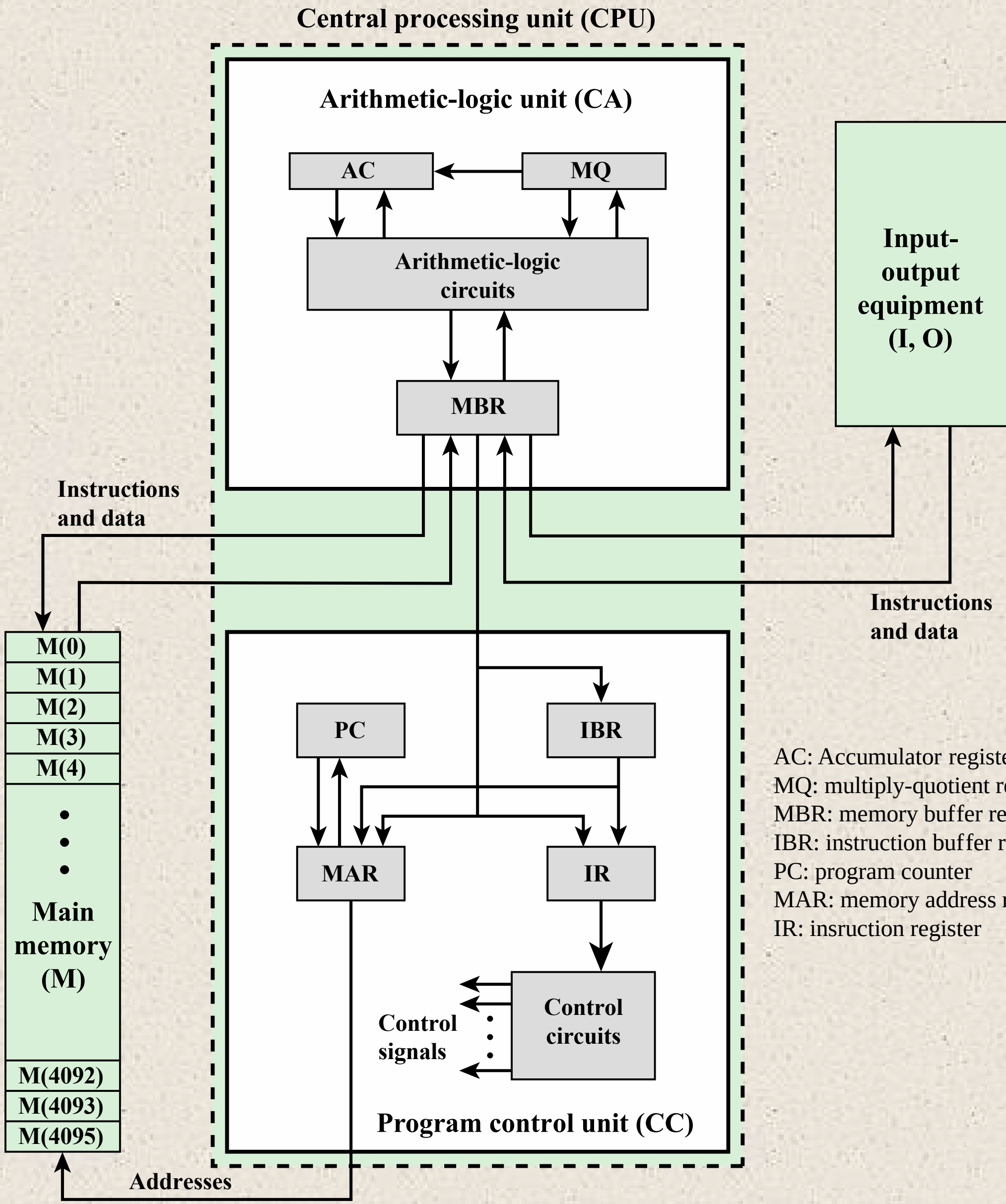
- Go to Apple Store website https://www.apple.com/my/mac/
- Try configures your dream Apple machine …
  - 1) clicking on [Buy] button
  - 2) select / customize your dream machine options
  - 3) click on [Add to Bag] button
  - 4) click on [Review Bag] button
- Estimate time : 10 minutes

+

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

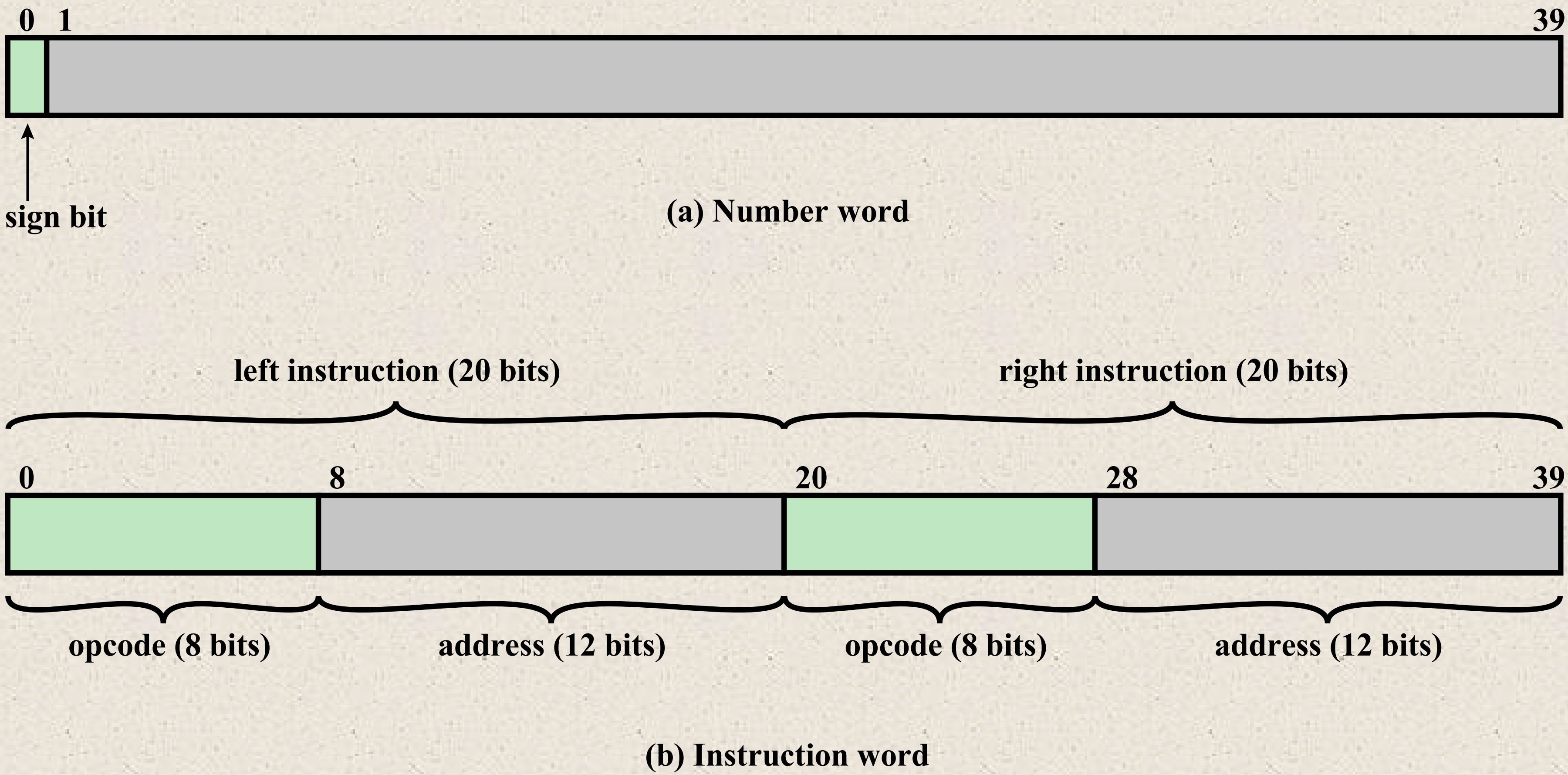
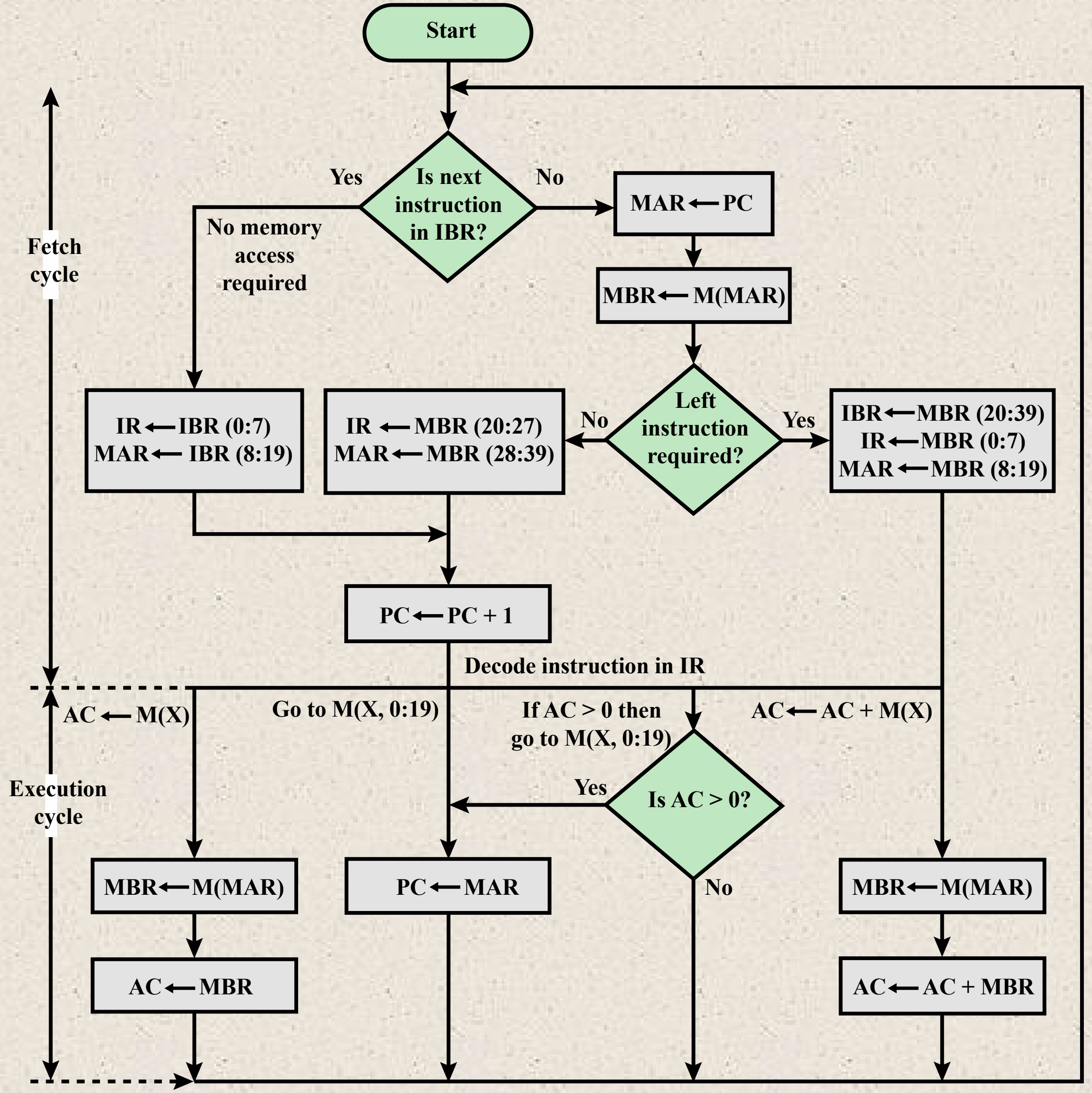


Figure 1.7 IAS Memory Formats

+

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



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

Figure 1.8 Partial Flowchart of IAS Operation

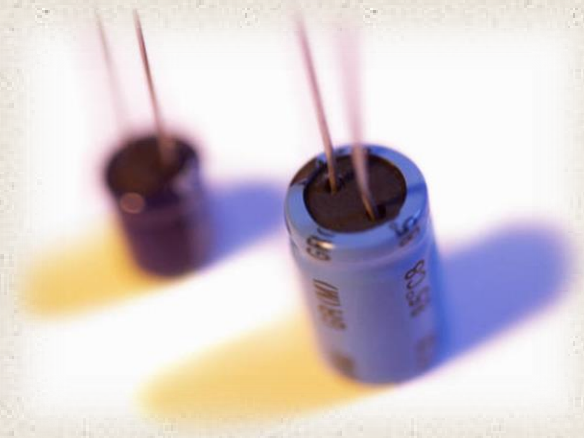
Instruction Type	Opcode	Symbolic Representation	Description
Data transfer	00001010	LOAD MQ	Transfer contents of register MQ to the accumulator AC
	00001001	LOAD MQ,M(X)	Transfer contents of memory location X to MQ
	00100001	STOR M(X)	Transfer contents of accumulator to memory location X
	00000001	LOAD 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 1.1

The IAS Instruction Set

(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

+

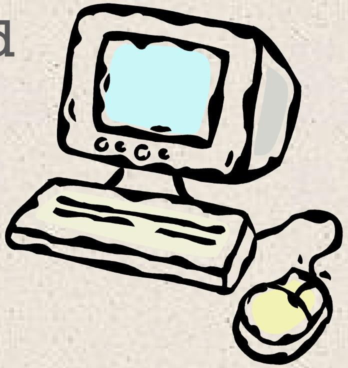
Table 1.2 Computer Generations

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

- 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



+
DEC PDP-1 (1960)

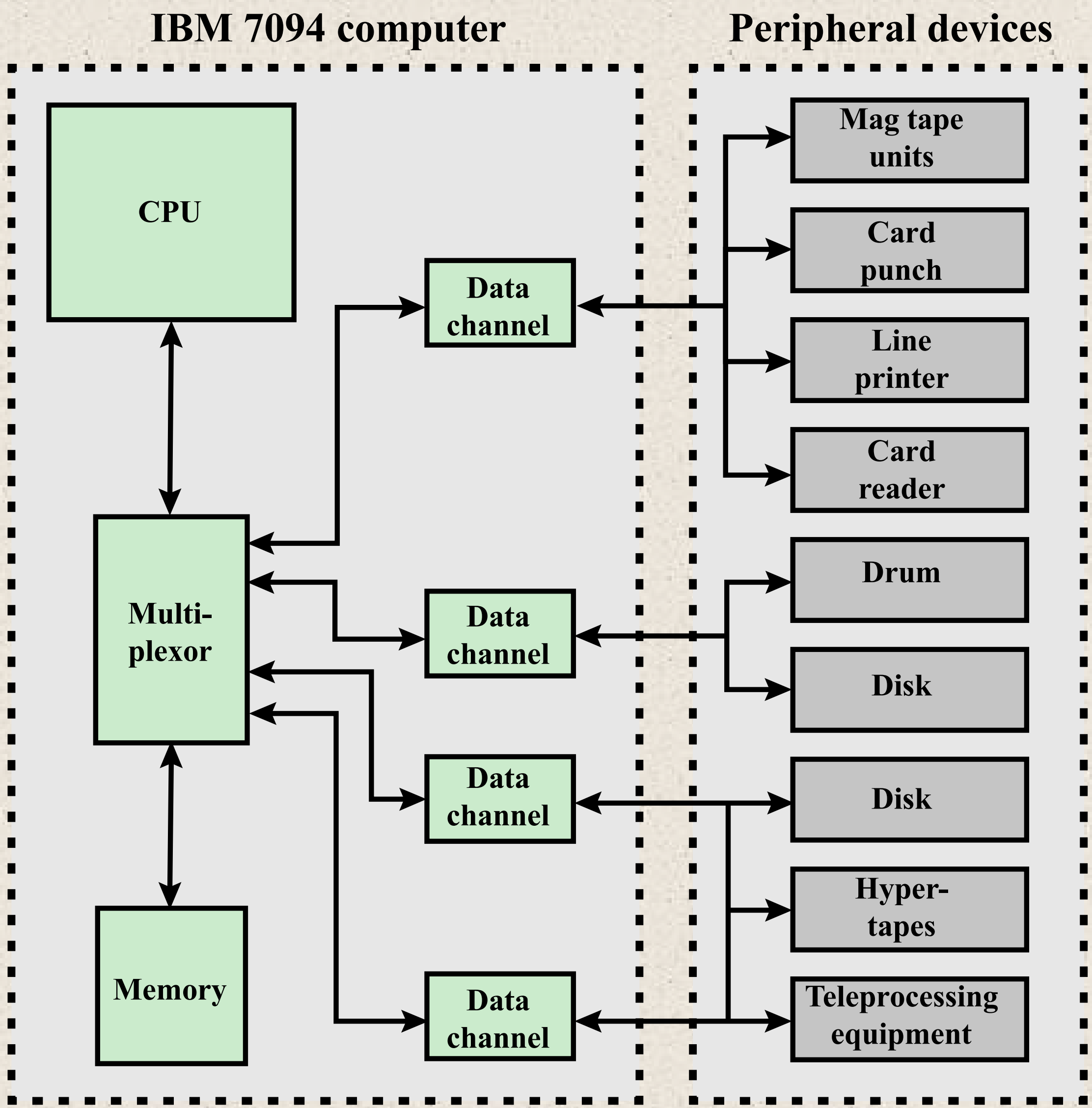


Figure 1.9 An IBM 7094 Configuration

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

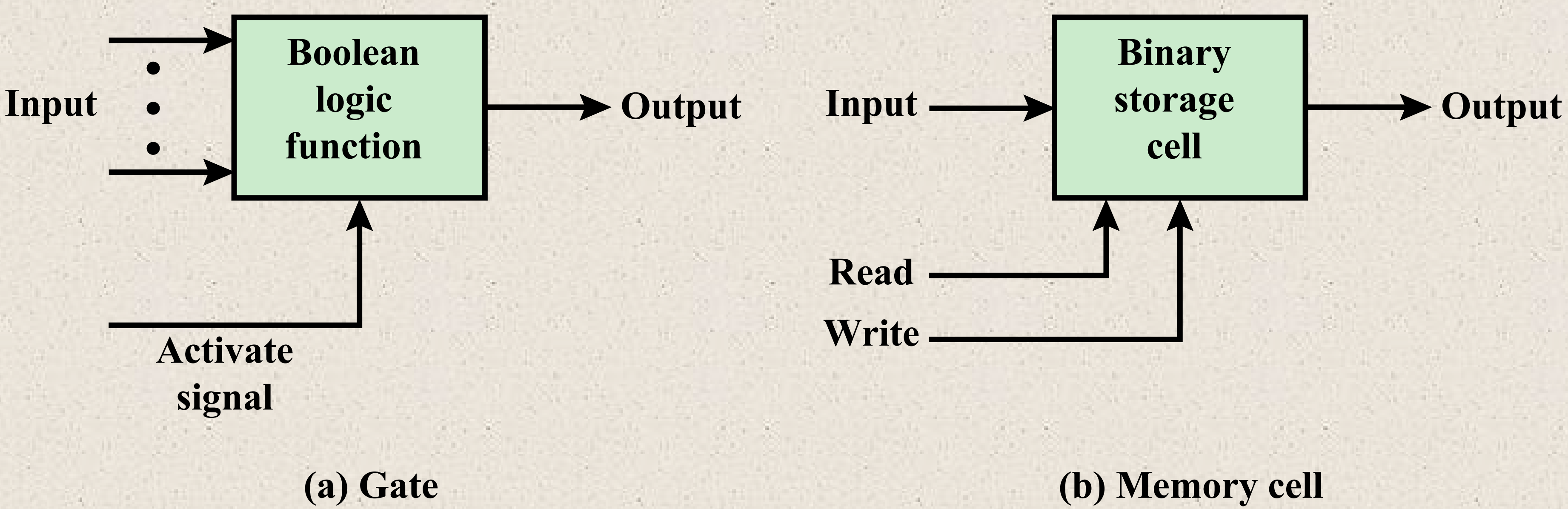


Figure 1.10 Fundamental Computer Elements

+

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

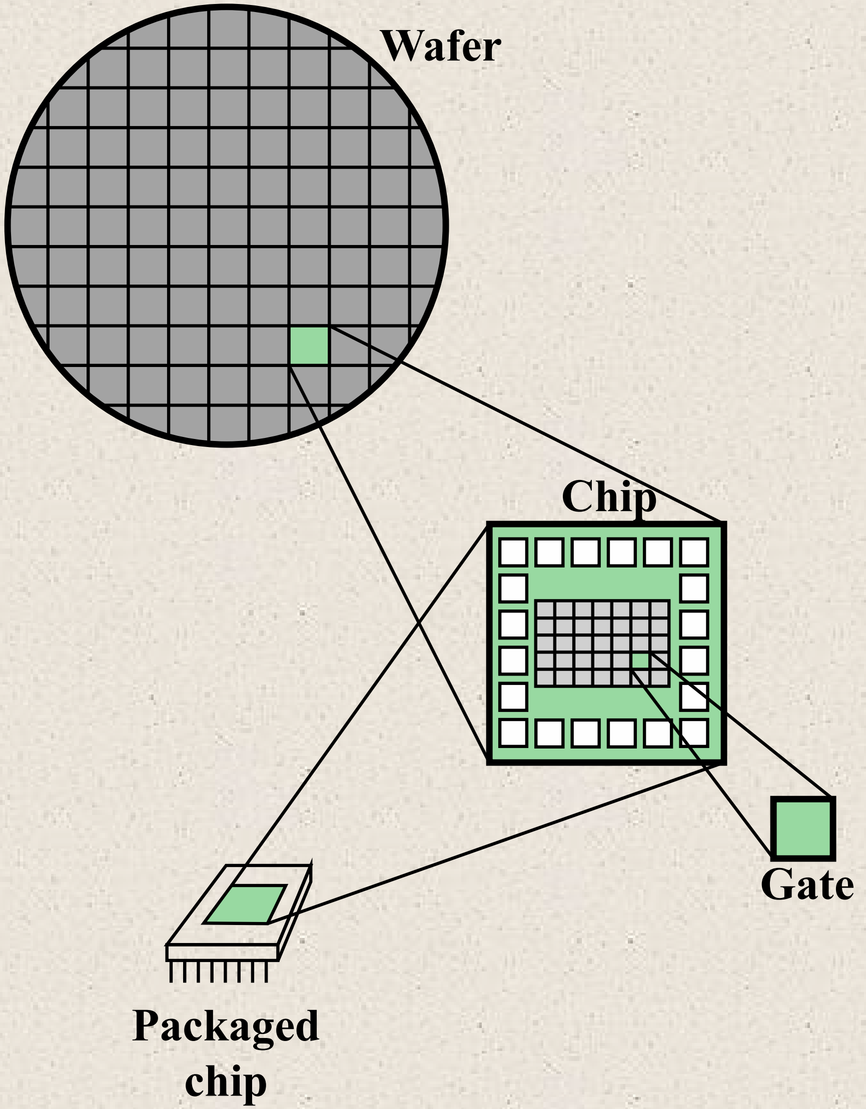


Figure 1.11 Relationship Among Wafer, Chip, and Gate

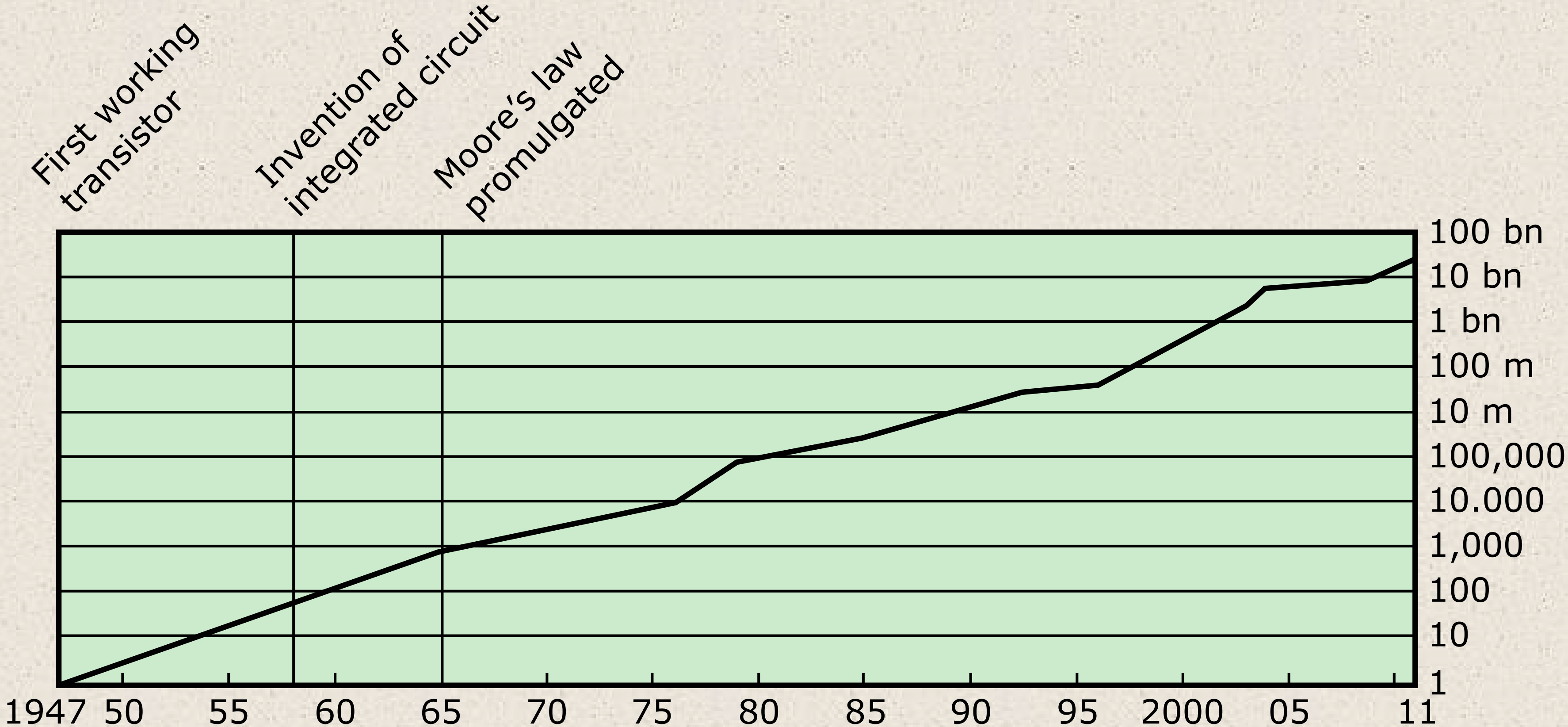


Figure 1.12 Growth in Transistor Count on Integrated Circuits (DRAM memory)

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**

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

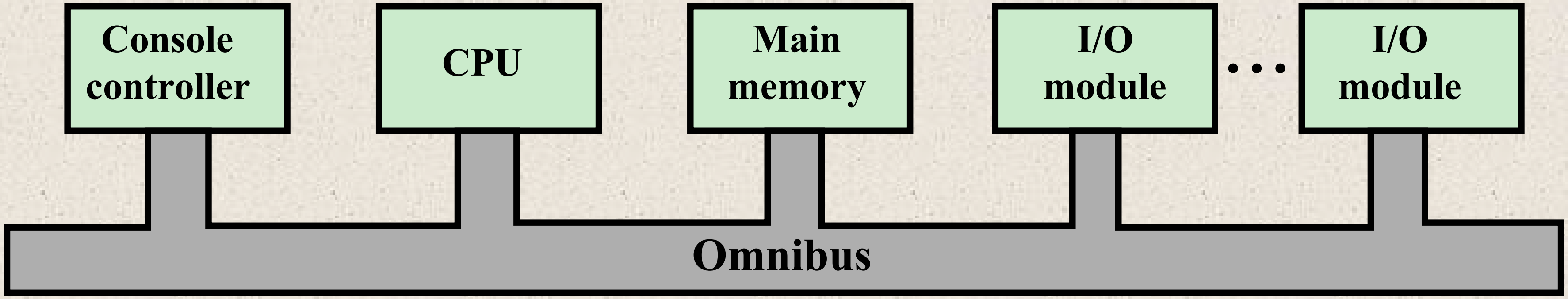


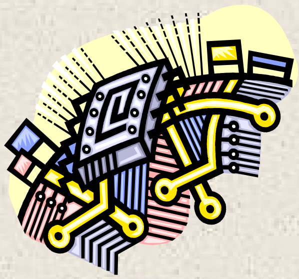
Figure 1.13 PDP-8 Bus Structure

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	8	6	3	6
Addressable memory	640 Bytes	16 KB	64 KB	1 MB	1 MB

(a) 1970s Processors

Evolution of Intel Microprocessors

	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

Evolution of Intel Microprocessors

	Pentium III	Pentium 4	Core 2 Duo	Core i7 EE 4960X
Introduced	1999	2000	2006	2013
Clock speeds	450 - 660 MHz	1.3 - 1.8 GHz	1.06 - 1.2 GHz	4 GHz
Bus width	64 bits	64 bits	64 bits	64 bits
Number of transistors	9.5 million	42 million	167 million	1.86 billion
Feature size (nm)	250	180	65	22
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 L2/15 MB L3
Number of cores	1	1	2	6

(d) Recent Processors

+

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

Highlights of the Evolution of the Intel Product Line:

8080

- World's first general-purpose microprocessor
- 8-bit machine, 8-bit data path to memory
- Was used in the first personal computer (Altair)

8086

- A more powerful 16-bit machine
- Has an instruction cache, or queue, that prefetches a few instructions before they are executed
- The first appearance of the x86 architecture
- The 8088 was a variant of this processor and used in IBM's first personal computer (securing the success of Intel

80286

- Extension of the 8086 enabling addressing a 16-MB memory instead of just 1MB

80386

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

80486

- Introduced the use of much more sophisticated and powerful cache technology and sophisticated instruction pipelining
- Also offered a built-in math coprocessor

Highlights of the Evolution of the Intel Product Line:

Pentium

- Intel introduced the use of superscalar techniques, which allow multiple instructions to execute in parallel

Pentium Pro

- Continued the move into superscalar organization with aggressive use of register renaming, branch prediction, data flow analysis, and speculative execution

Pentium II

- Incorporated Intel MMX technology, which is designed specifically to process video, audio, and graphics data efficiently

Pentium III

- Incorporated additional floating-point instructions
- Streaming SIMD Extensions (SSE)

Pentium 4

- Includes additional floating-point and other enhancements for multimedia

Core

- First Intel x86 micro-core

Core 2

- Extends the Core architecture to 64 bits
- Core 2 Quad provides four cores on a single chip
- More recent Core offerings have up to 10 cores per chip
- An important addition to the architecture was the Advanced Vector Extensions instruction set

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

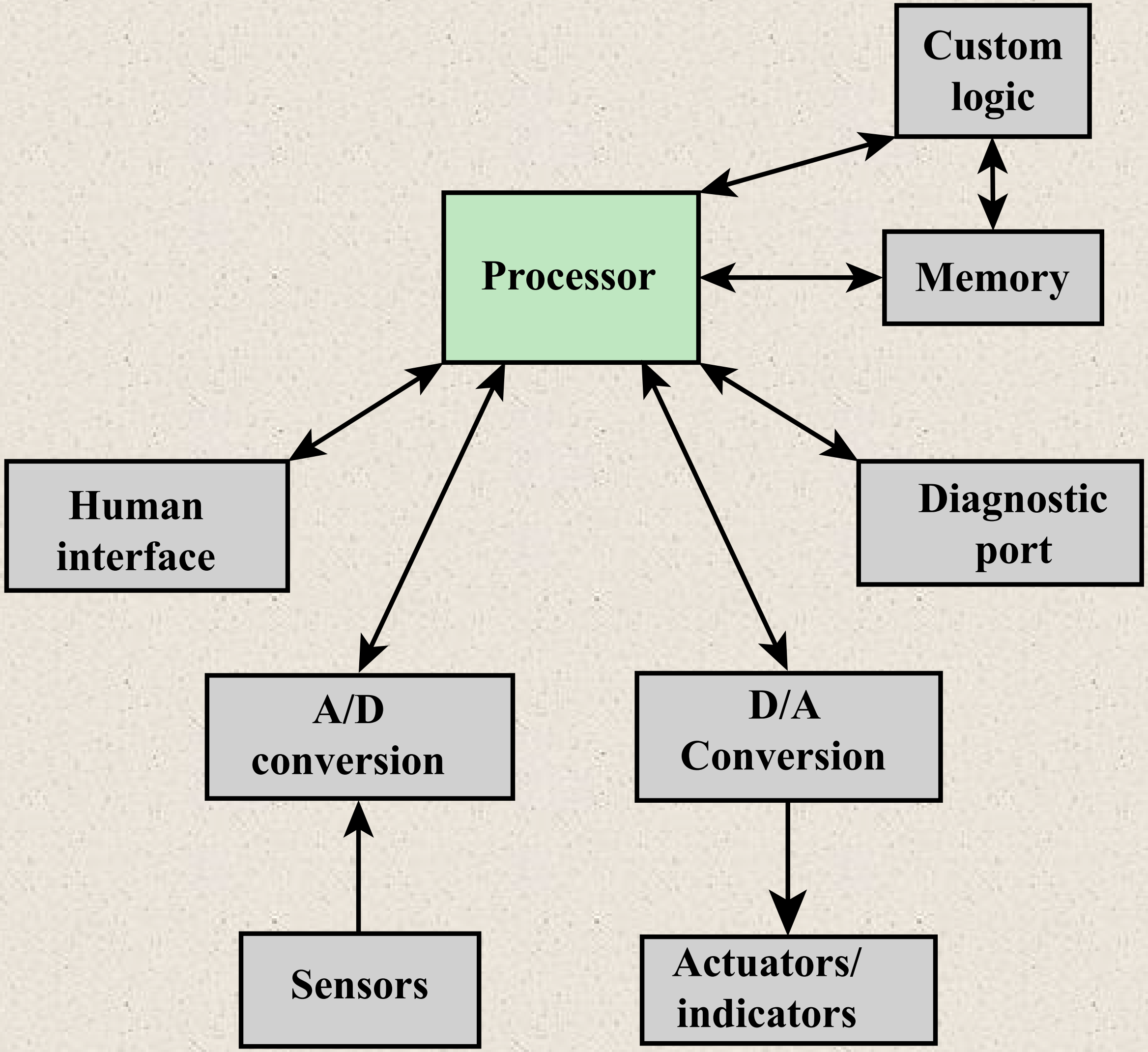


Figure 1.14 Possible Organization of an Embedded System

The Internet of Things (IoT)

- Term that refers to the expanding interconnection of smart devices, ranging from appliances to tiny sensors
- Is primarily driven by deeply embedded devices
- Generations of deployment culminating in the IoT:
  - Information technology (IT)
    - PCs, servers, routers, firewalls, and so on, bought as IT devices by enterprise IT people and primarily using wired connectivity
  - Operational technology (OT)
    - Machines/appliances with embedded IT built by non-IT companies, such as medical machinery, SCADA, process control, and kiosks, bought as appliances by enterprise OT people and primarily using wired connectivity
  - Personal technology
    - Smartphones, tablets, and eBook readers bought as IT devices by consumers exclusively using wireless connectivity and often multiple forms of wireless connectivity
  - Sensor/actuator technology
    - Single-purpose devices bought by consumers, IT, and OT people exclusively using wireless connectivity, generally of a single form, as part of larger systems
- It is the fourth generation that is usually thought of as the IoT and it is marked by the use of billions of embedded devices

+

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

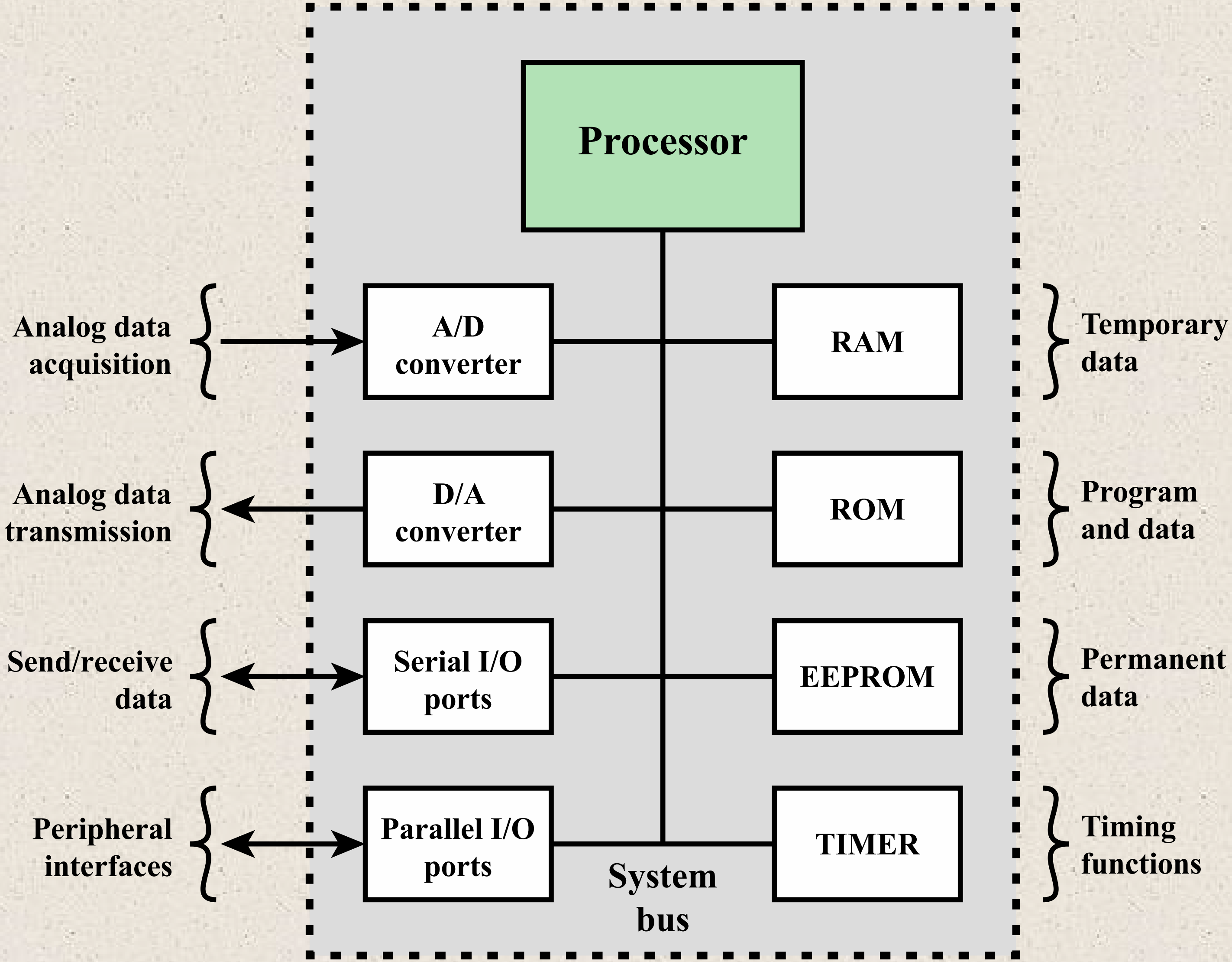


Figure 1.15 Typical Microcontroller Chip Elements

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

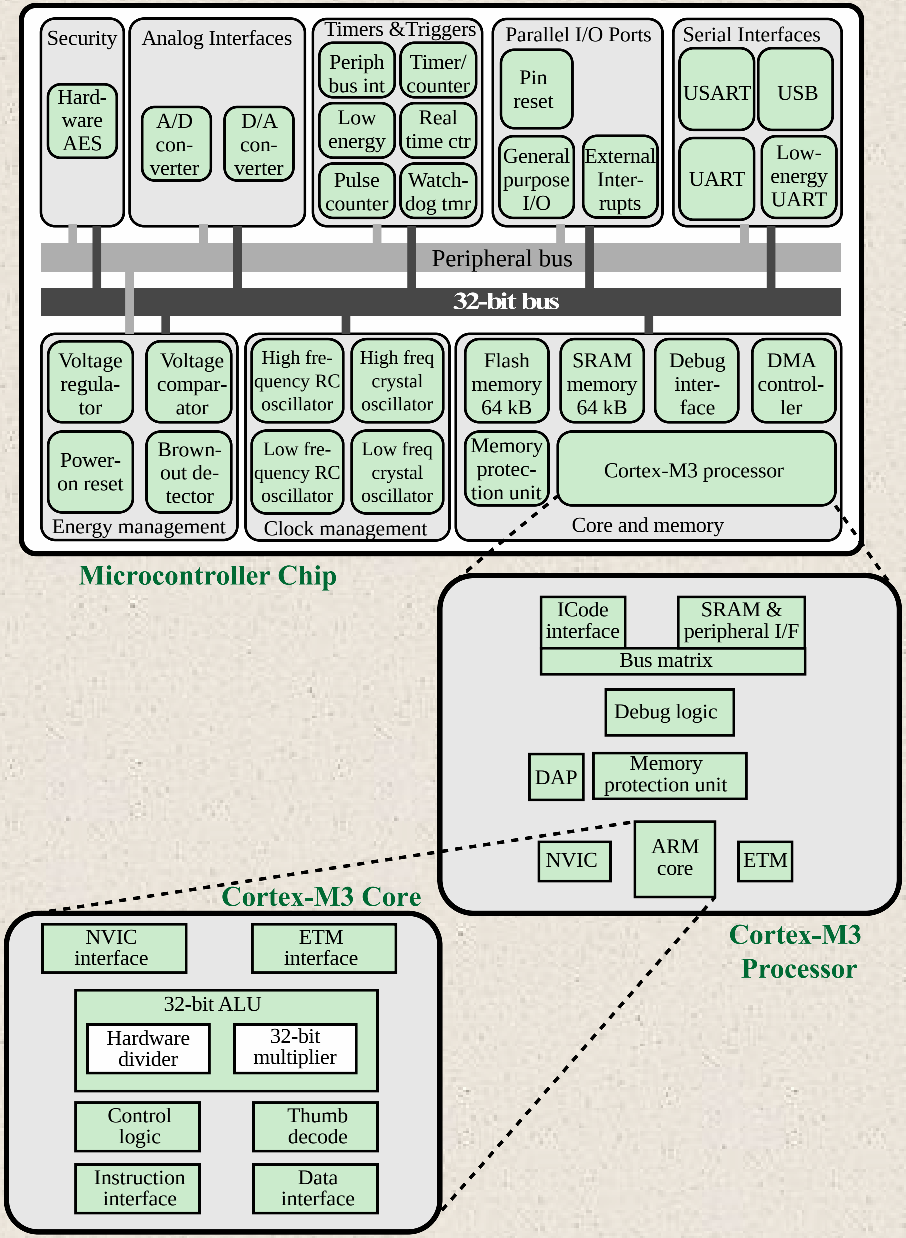


Figure 1.16 Typical Microcontroller Chip Based on Cortex-M3

+
Cloud Computing

- NIST defines cloud computing as:

  "A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction."

- You get economies of scale, professional network management, and professional security management

- The individual or company only needs to pay for the storage capacity and services they need

- Cloud provider takes care of security

Cloud Networking

- Refers to the networks and network management functionality that must be in place to enable cloud computing
- One example is the provisioning of high-performance and/or high-reliability networking between the provider and subscriber
- The collection of network capabilities required to access a cloud, including making use of specialized services over the Internet, linking enterprise data center to a cloud, and using firewalls and other network security devices at critical points to enforce access security policies

Cloud Storage

- Subset of cloud computing
- Consists of database storage and database applications hosted remotely on cloud servers
- Enables small businesses and individual users to take advantage of data storage that scales with their needs and to take advantage of a variety of database applications without having to buy, maintain, and manage the storage assets

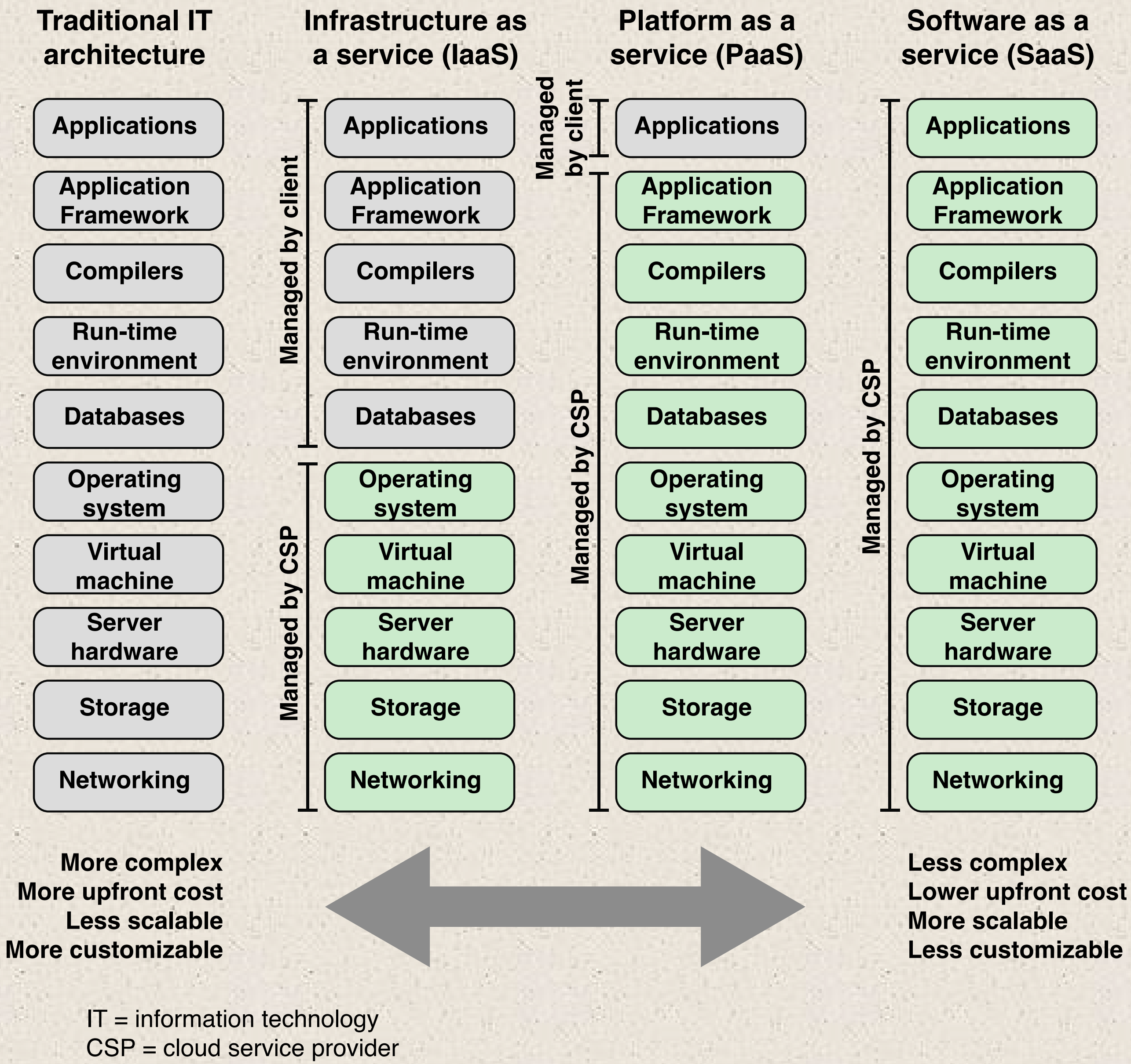


Figure 1.17 Alternative Information Technology Architectures

The Computer Level Hierarchy

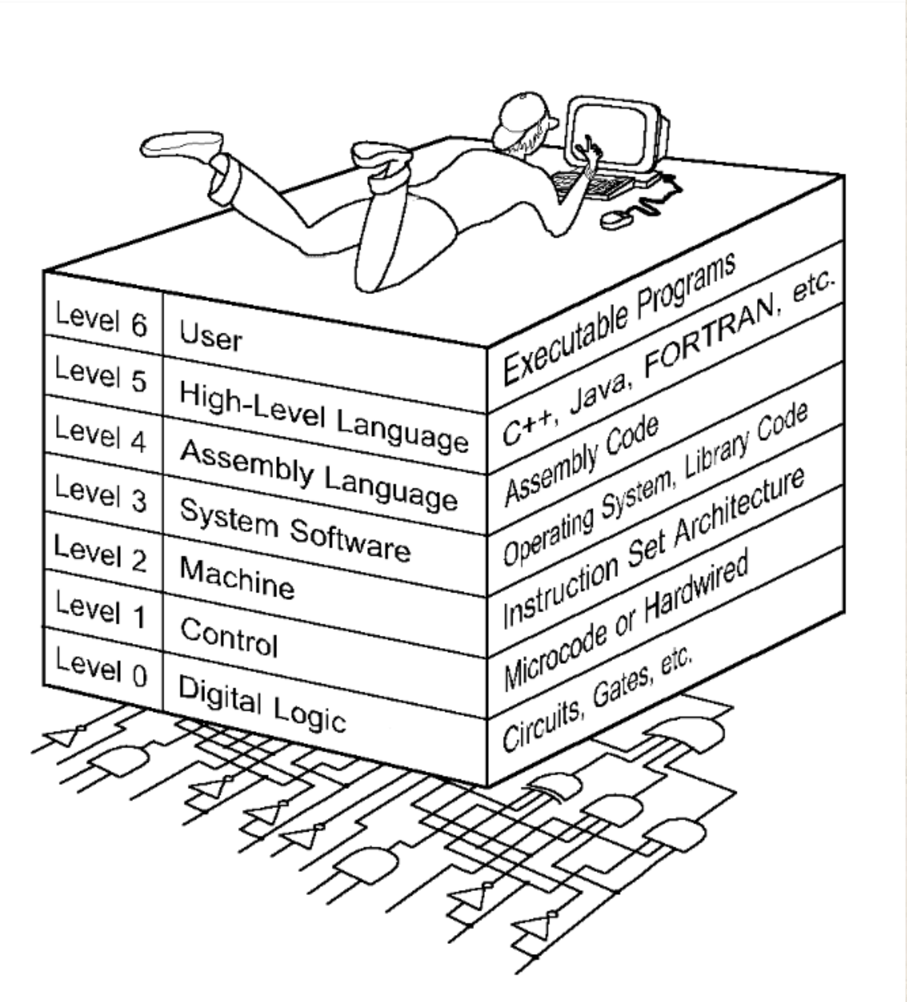
REFERENCE: LINDA NULL & JULIA LOBUR– THE ESSENTIALS OF COMPUTER ORGANIZATION & ARCHITECTURE

The Computer Level Hierarchy

- Computers consist of many things besides chips.
- Before a computer can do anything worthwhile, it must also use software.
- Writing complex programs requires a *"divide and conquer" approach*, where each program module solves a smaller problem.
- *Complex computer systems* employ a similar technique through *a series of virtual machine layers*.

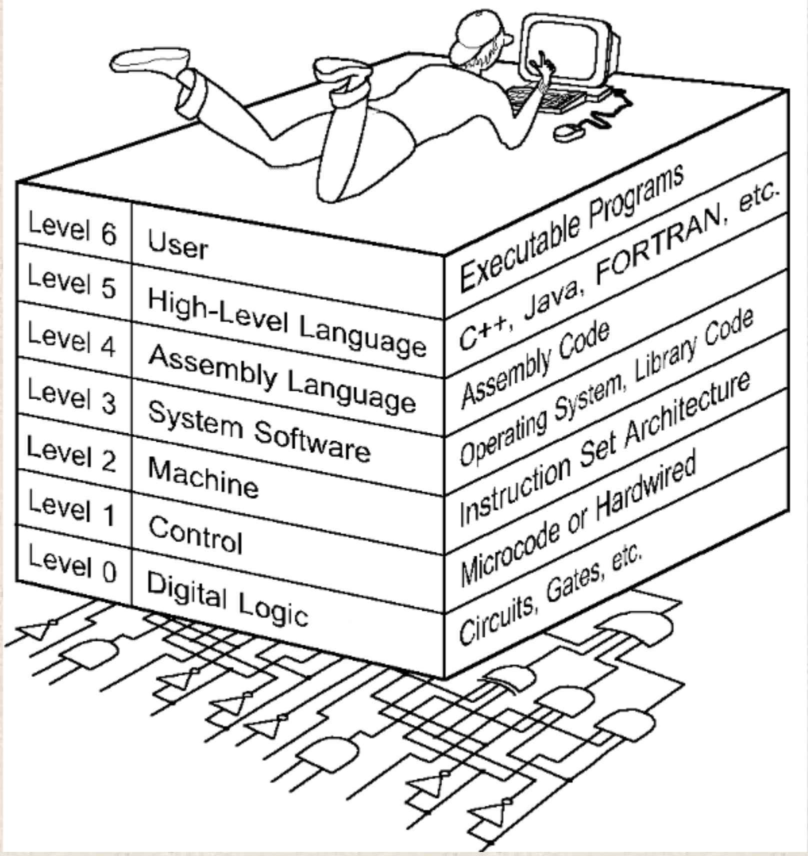
+ The Computer Level Hierarchy

- Each virtual machine layer is an abstraction of the level below it.
- The machines at each level execute their own particular instructions, calling upon machines at lower levels to perform tasks as required.
- Computer circuits ultimately carry out the work.



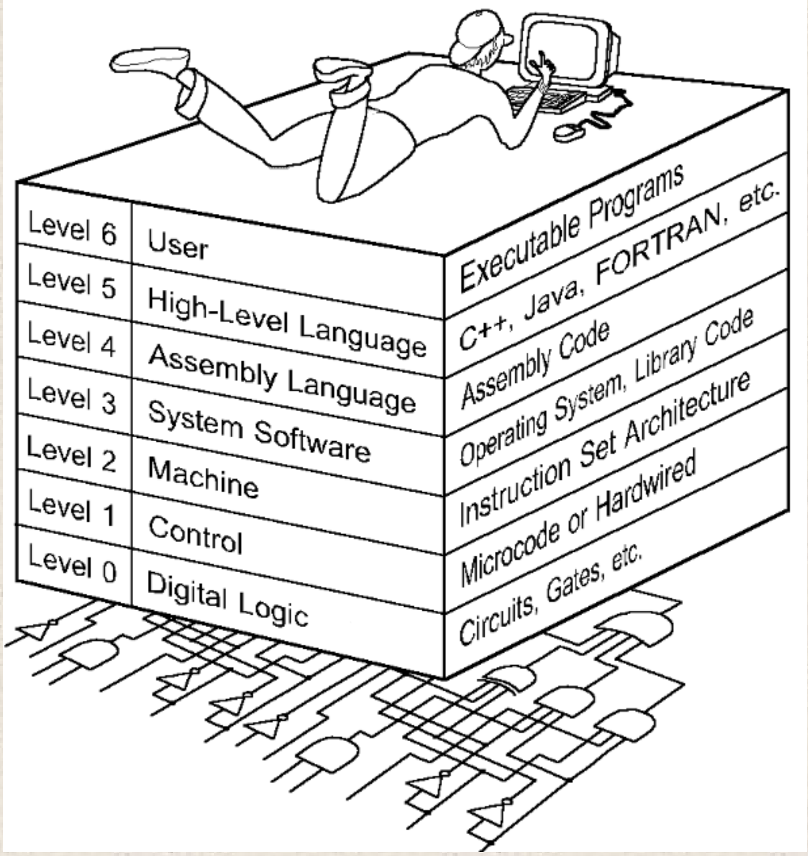
The Computer Level Hierarchy

- Level 6: The User Level
  - Program execution and user interface level.
  - The level with which we are most familiar.



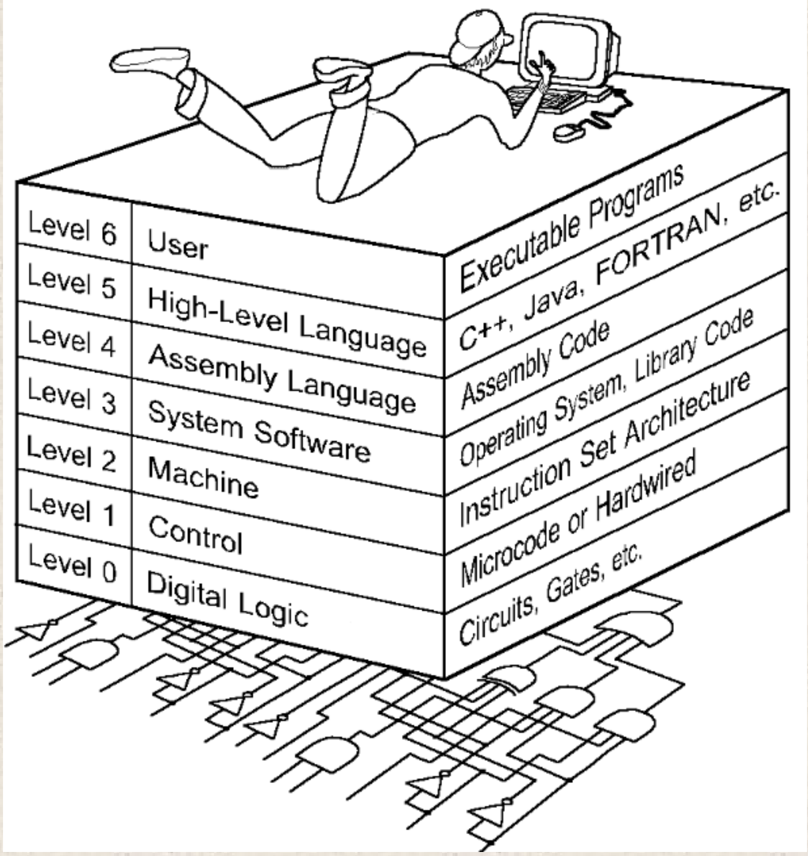
The Computer Level Hierarchy

- Level 5: High-Level Language Level
  - The level with which we interact when we write programs in languages such as C, Pascal, Lisp, and Java.



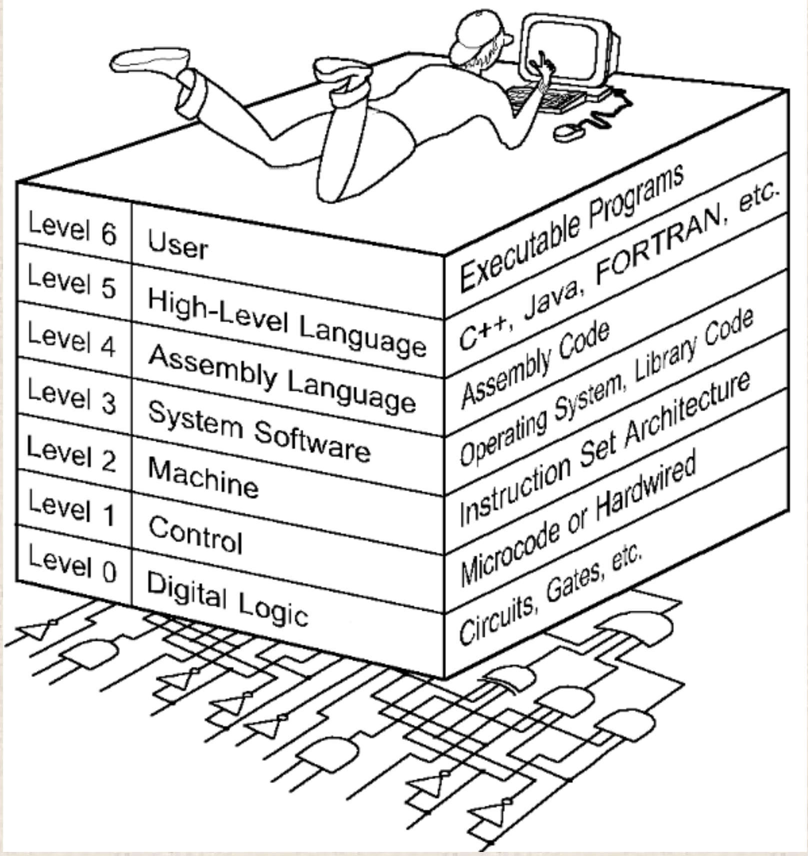
The Computer Level Hierarchy

- Level 4: Assembly Language Level
  - Acts upon assembly language produced from Level 5, as well as instructions programmed directly at this level.



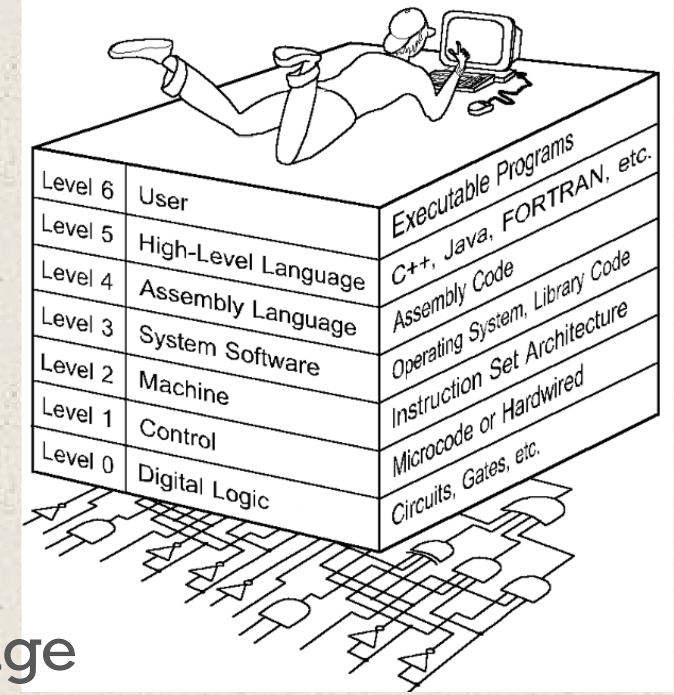
The Computer Level Hierarchy

- Level 3: System Software Level
  - Controls executing processes on the system.
  - Protects system resources.
  - Assembly language instructions often pass through Level 3 without modification.



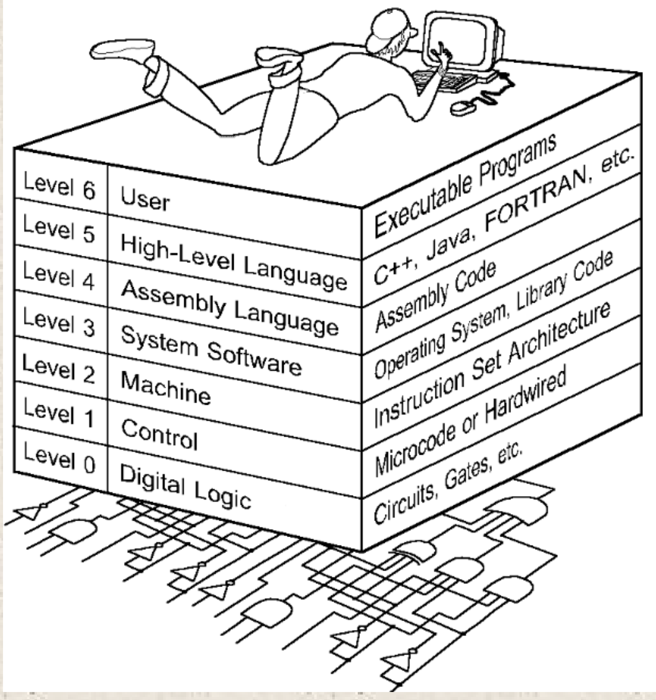
The Computer Level Hierarchy

- Level 2: Machine Level
  - Also known as the Instruction Set Architecture (ISA) Level.
  - Consists of instructions that are particular to the architecture of the machine.
  - Programs written in machine language need no compilers, interpreters, or assemblers.



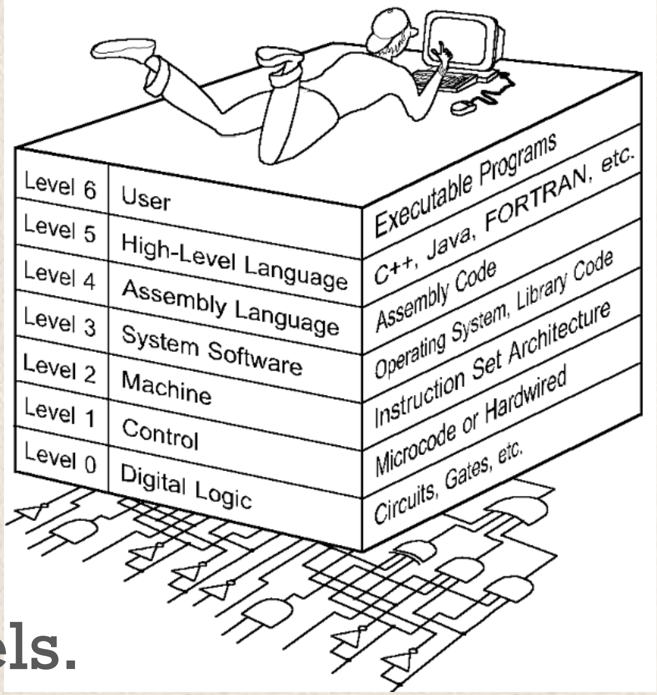
+ The Computer Level Hierarchy

- Level 1: Control Level
  - A *control unit* decodes and executes instructions and moves data through the system.
  - Control units can be *microprogrammed* or *hardwired*.
  - A microprogram is a program written in a low-level language that is implemented by the hardware.
  - Hardwired control units consist of hardware that directly executes machine instructions.



+ The Computer Level Hierarchy

- Level 0: Digital Logic Level
  - This level is where we find digital circuits (the chips).
  - Digital circuits consist of gates and wires.
  - These components implement the mathematical logic of all other levels.



+ # 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

+

Conclusion

- This ‘Module 1’ has given you an overview of the subject of computer architecture.
- You should now be sufficiently familiar with general system structure to guide your studies throughout the remainder of this course.
- Subsequent chapters will explore many of these topics in great detail.