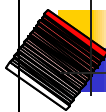


# Introduction to Computers

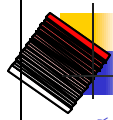
Chapter No 3

## Processing Hardware



## How Computer Represents Data?

- Binary Numbers
- The Binary Number System
- Bits and Bytes
- Text Codes



## Binary Numbers

- Computer processing is performed by transistors, which are switches with only two possible states: **ON** and **OFF**.
- All computer data is converted to a series of binary numbers– **1** and **0**.
  - For example, you see a sentence as a collection of letters, but the computer sees each letter as a collection of 1s and 0s.
- If a transistor is assigned a value of 1, it is on. If it has a value of 0, it is off. A computer's transistors can be switched on and off millions of times each second.

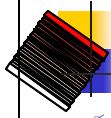


H	0100	1000
e	0110	0101
r	0111	0010
e	0110	0101
	0010	0000
a	0110	0001
r	0111	0010
e	0110	0101
	0010	0000
s	0111	0011
o	0110	1111
m	0110	1101
e	0110	0101
	0010	0000
w	0111	0111
o	0110	1111
r	0111	0010
d	0110	0100
s	0111	0011
.	0010	0001

0
1
2
3
4
5
6
7
8
9
10
11
:
:
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255

Ten different symbols in the decimal system

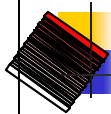
Numbers above 9 use more than 1 digit



## Binary Numbers

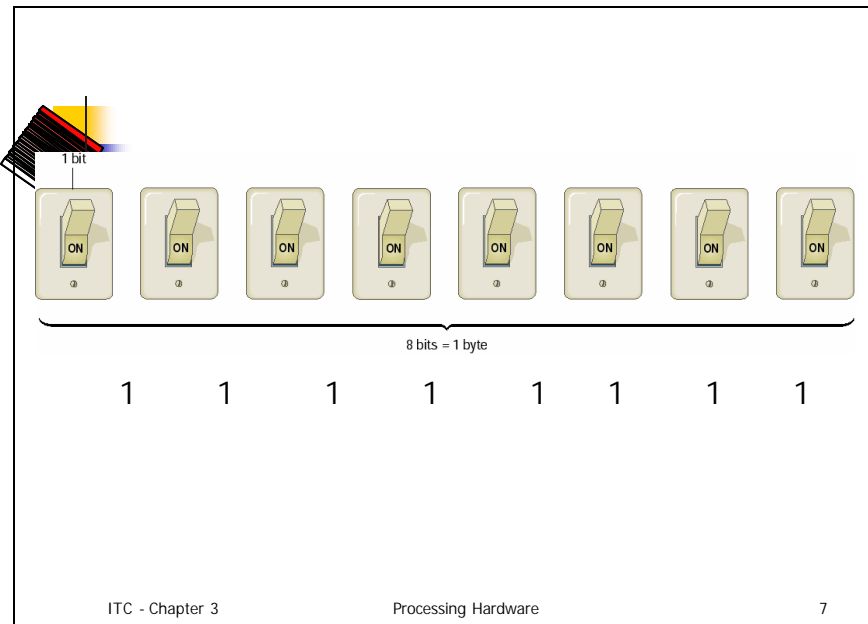
- ✦ To convert data into strings of numbers, computers use the binary number system.
- ✦ Humans use the decimal system ("deci" stands for "ten").
- ✦ The binary number system works the same way as the decimal system, but has only two available symbols (0 and 1) rather than ten (0, 1, 2, 3, 4, 5, 6, 7, 8, and 9).

Base 10	Base 2
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
10	1010



## Bits and Bytes

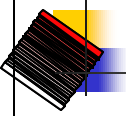
- ✦ A single unit of data is called a **BIT**, having a value of 1 or 0.
- ✦ Computers work with collections of bits, grouping them to represent larger pieces of data, such as letters of the alphabet.
- ✦ Eight bits make up one **BYTE**. A byte is the amount of memory needed to store one alphanumeric character.
- ✦ With one byte, the computer can represent one of 256 different symbols or characters.



## Text Code

- ✦ **Text Code** is a system that uses binary numbers (1s and 0s) to represent characters understood by humans (letters and numerals).
- ✦ An early text code system, called **EBCDIC**, uses eight-bit codes, but is used primarily in older mainframe systems.
- ✦ In the most common text-code set, **ASCII**, each character consists of eight bits (one byte) of data. ASCII is used in nearly all personal computers.
- ✦ In the **Unicode** text-code set, each character consists of 16 bits (two bytes) of data.

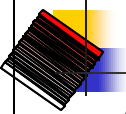
ITC - Chapter 3 Processing Hardware 8



**Examples from the  
ASCII Text Code**

Code	Character
00110000	0
00110001	1
00110010	2
00110011	3
00110100	4
00110101	5
01000001	A
01000010	B
01000011	C
01000100	D
01000101	E

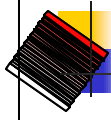
ITC - Chapter 3 Processing Hardware 9



## Binary Coding Schemes

- ✍ All the amazing things that computers do are based on binary numbers made up of 0s and 1s.
- ✍ Fortunately we do not enter data into the computer using groups of 0s and 1s.
- ✍ Instead we use natural language characters such as those on the keyboard.
- ✍ Then the computer system encodes the data by means of *binary or digital coding schemes* to represent letters, numbers and special characters

ITC - Chapter 3 Processing Hardware 10

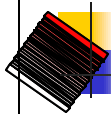


## Binary Coded Schemes



There are many coding schemes, the most common ones are:

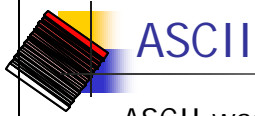
1. **EBCDIC**  
Extended Binary Coded Decimal Interchange Code
2. **ASCII**  
American Standard Code for Information Interchange
3. **Unicode**



## EBCDIC

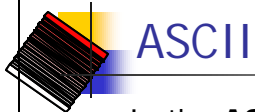


- It is commonly used in IBM mainframes.
- It is an **8-bit** coding scheme.
- So it can represent  $2^8 = 256$  characters.



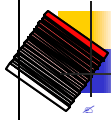
## ASCII

- ✍ ASCII was created for use in early telecommunications systems but has proven useful for computer systems and has been the basis for most other character sets.
- ✍ Most widely used binary code.
- ✍ Standard ASCII uses 7 bits, so  $2^7=128$
- ✍ Extended ASCII uses 8 bits, so  $2^8=256$



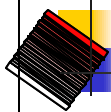
## ASCII

- ✍ In the **ASCII character set**, each binary value between 0 and 127 is given a specific character.
- ✍ Most computers extend the ASCII character set to use the full range of 256 characters available in a byte.
- ✍ The upper 128 characters handle special things like accented characters from common foreign languages.



## ASCII

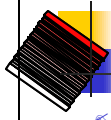
- ✍ Computers store text documents, both on disk and in memory, using these codes. For example, if you use Notepad in Windows 95/98 to create a text file containing the words, "Four score and seven years ago," Notepad would use 1 byte of memory per character (including 1 byte for each space character between the words -- ASCII character 32). When Notepad stores the sentence in a file on disk, the file will also contain 1 byte per character and per space.
- ✍ Try this experiment: Open up a new file in Notepad and insert the sentence, "Four score and seven years ago" in it. Save the file to disk under the name **getty.txt**. Then use the explorer and look at the size of the file. You will find that the file has a size of 30 bytes on disk: 1 byte for each character. If you add another word to the end of the sentence and re-save it, the file size will jump to the appropriate number of bytes. Each character consumes a byte.



## Unicode

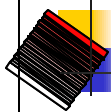
- ✍ Unicode is an entirely new idea in setting up binary codes for text or script characters.
- ✍ Officially called the Unicode Worldwide Character Standard, it is a system for "the interchange, processing, and display of the written texts of the diverse languages of the modern world." It also supports many classical and historical texts in a number of languages.
- ✍ Currently, the **Unicode standard contains 34,168 distinct coded characters** derived from 24 supported language scripts.





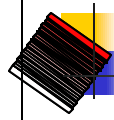
## Unicode

- ✍ It uses 2 bytes (16 bits)
- ✍ Therefore,  $2^{16}=65,536$  characters can be handled.
- ✍ This is a bit of overkill for English and Western-European languages, but it is necessary for some other languages, such as Greek, Chinese and Japanese.
- ✍ Many analysts believe that as the software industry becomes increasingly global, Unicode will eventually supplant ASCII as the standard character coding format.



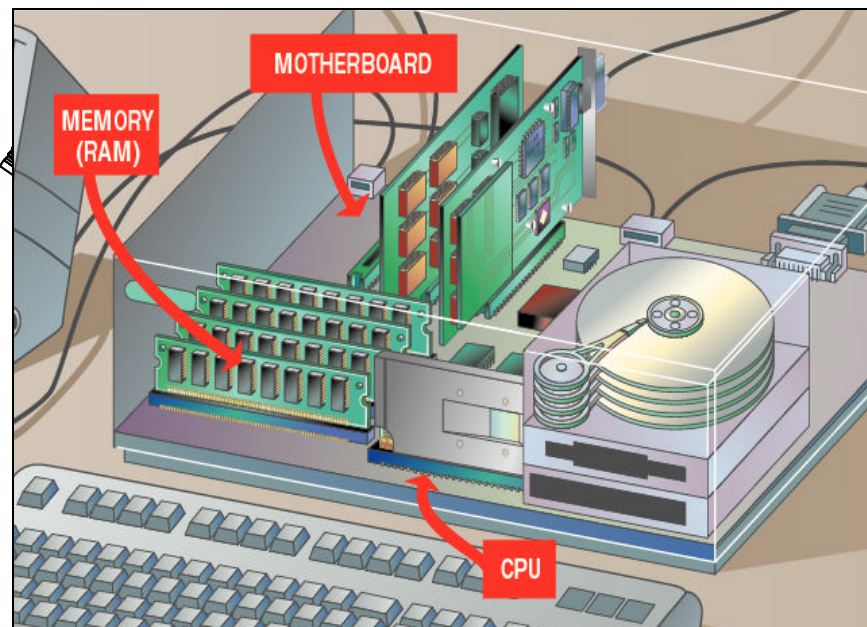
## Where Processing Occurs

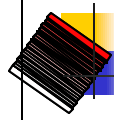
- ✍ The Control Unit
- ✍ The Arithmetic Logic Unit
- ✍ Machine Cycles
- ✍ The Role of Memory in Processing
- ✍ Types of RAM



## Where Processing Occurs

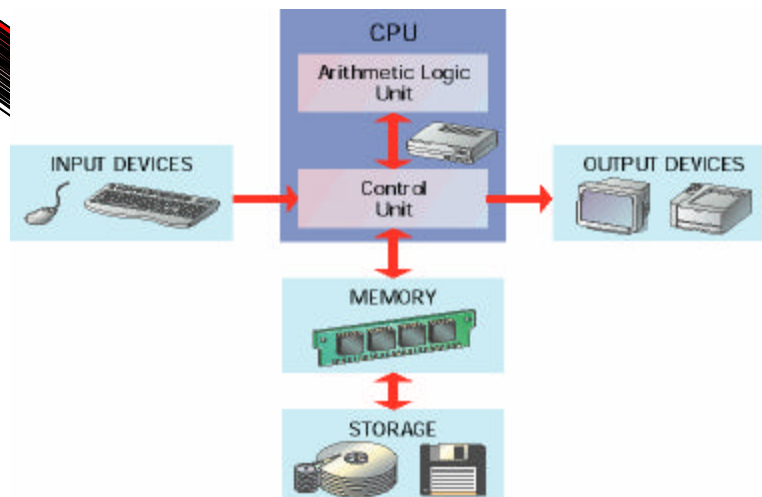
- ✍ Processing takes place in the PC's central processing unit (CPU).
- ✍ The system's memory also plays a crucial role in processing data.
- ✍ Both the CPU and memory are attached to the system's motherboard, which connects all the computer's devices together, enabling them to communicate.

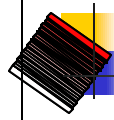




## Central Processing Unit

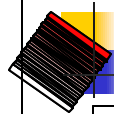
- ✍ The two main parts of a CPU are
  - ✍ Control Unit (CU) and
  - ✍ Arithmetic Logic Unit (ALU)
- ✍ The control unit directs the flow of data through the CPU, and to and from other devices.
- ✍ The control unit stores the CPU's microcode, which contains the instructions for all the tasks the CPU can perform.





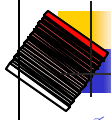
## Arithmetic Logic Unit

- ✍ The actual manipulation of data takes place in the ALU.
- ✍ The ALU can perform arithmetic and logic operations.
- ✍ The ALU is connected to a set of registers—small memory areas in the CPU, which hold data and program instructions while they are being processed



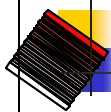
## ALU Operation List

Arithmetic Operations	Logical Operations
+ Add	?, ? equal to, not equal to
? Subtract	>, > greater than, not greater than
x Multiply	<, < less than, not less than
? Divide	?, > greater than or equal to, not greater than or equal to
^ Raise by a power	?, > less than or equal to, not less than or equal to



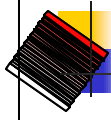
## Machine Cycle

- ✍ The CPU follows a set of steps-called a machine cycle-for each instruction it carries out.
- ✍ By using a technique called pipelining, many CPUs can process more than one instruction at a time.
- ✍ The machine cycle includes two smaller cycles:
- ✍ During the instruction cycle, the CPU "fetches" a command or data from memory and "decodes" it for the CPU.
- ✍ During the execution cycle, the CPU carries out the instruction, and may store the instruction's result in memory.



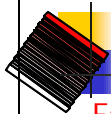
## Machine Cycles

- ✍ **Address:**
  - ✍ Is the location designated by a unique number, in main memory in which a character of data or part of an instruction is stored.
- ✍ To process a character, the CU retrieves the character from its address in main memory and places it into a register.
- ✍ Machine cycle comprises of a series of operations performed to execute a single program instruction.
- ✍  $MC = I\text{-cycle} + E\text{-cycle}$



## Word Size

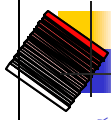
- As main memory capacity is measured in MB, processor capacity is measured in word size.
- Word size =
  - 1- number of bits processor can hold in its registers
  - 2- process at one time
  - 3- send through its local bus



## Processing Speed

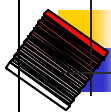
### Factors Affecting Processing Speed

- Registers
- RAM
- The System Clock
- The Bus
- Cache Memory



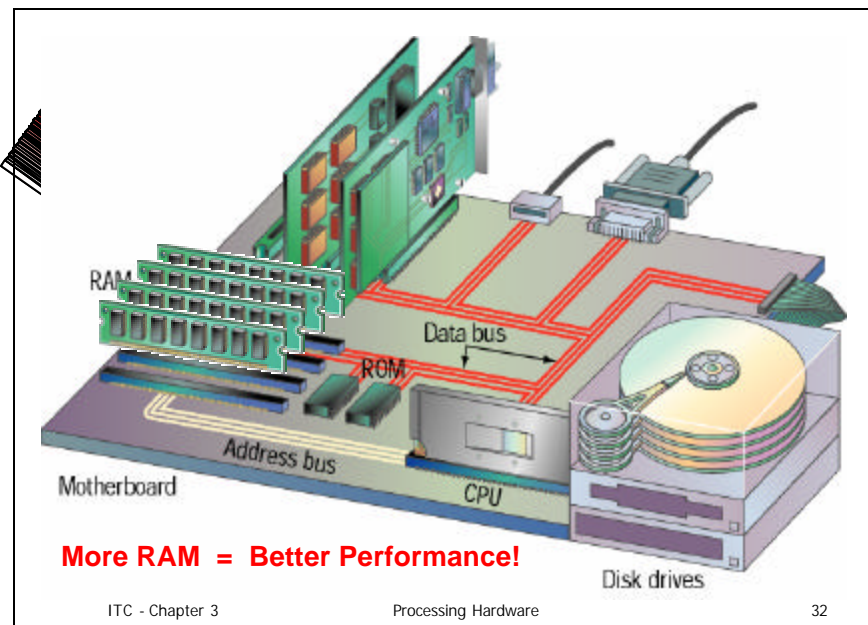
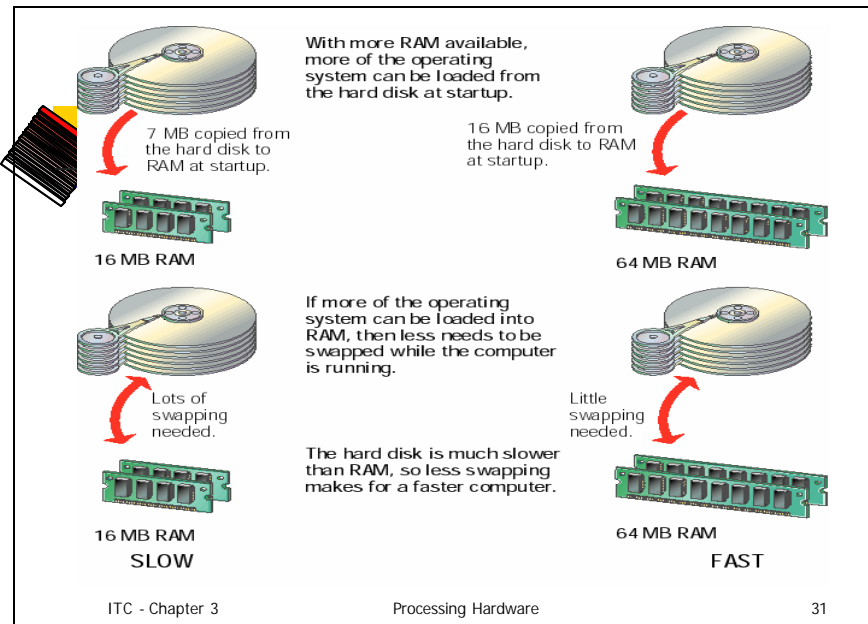
## Registers

- ✦ The CPU contains a number of small memory areas, called registers, which store data and instructions while the CPU processes them.
- ✦ The size of the registers (also called word size) determines the amount of data with which the computer can work at a one time.
- ✦ Today, most PCs have 32-bit registers, mean the CPU can process four bytes of data at one time. Register sizes are rapidly growing to 64 bits.

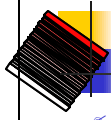


## RAM- Random Access Memory

- ✦ The amount of RAM in a PC has a direct affect on the system's speed.
- ✦ The more RAM a PC has, the more program instructions and data can be held in memory, which is faster than storage on disk.
- ✦ If a PC does not have enough memory to run a program, it must move data between RAM and the hard disk frequently. This process, called swapping, can greatly slow a PC's performance.

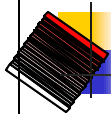






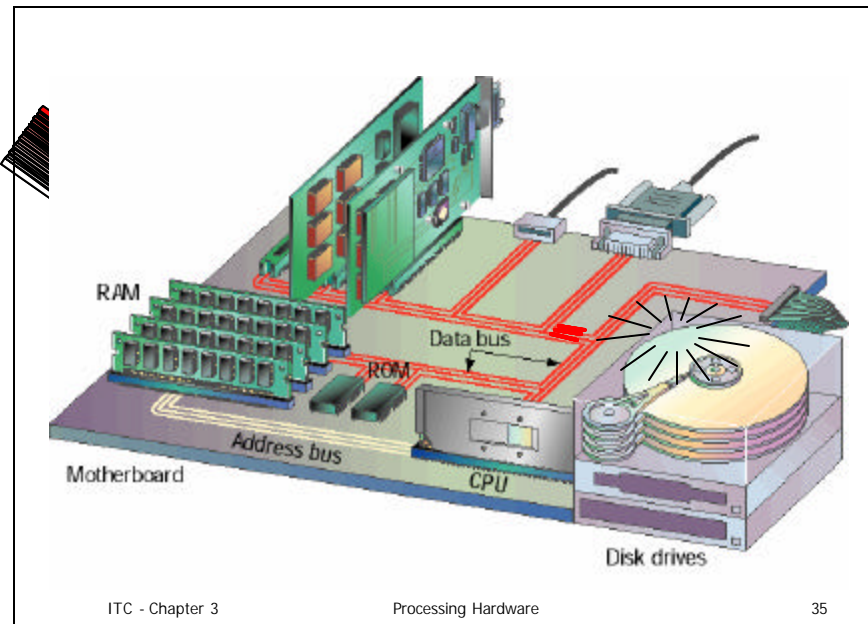
## System Clock

- ✦ The computer's system clock sets the pace for the CPU by using a vibrating quartz crystal.
- ✦ A single "**tick**" of the clock is the time required to turn a transistor off and back on. This is called a clock cycle.
- ✦ Clock cycles are measured in **Hertz (Hz)**, a measure of cycles per second.
  - ✦ If a computer has a clock speed of 300 MHz, then its system clock "ticks" 300 million times every second.
- ✦ The faster a PC's clock runs, the more instructions the PC can execute each second.



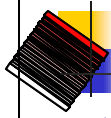
## Bus

- ✦ A **BUS** is a path between the components of a computer. Data and instructions travel along these paths.
- ✦ The **Data Bus** width determines how many bits can be transmitted between the CPU and other devices.
- ✦ The **Address Bus** runs only between the CPU and RAM, and carries nothing but memory addresses for the CPU to use.
- ✦ Peripheral devices are connected to the CPU by an expansion bus.



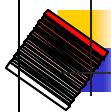
## Cache Memory

- Pronounced *cash*, a special high-speed storage mechanism.
- It can be either a reserved section of main memory or an independent high-speed storage device.
- Two types of caching are commonly used in personal computers: *memory caching* and *disk caching*.



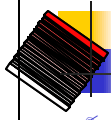
## Cache Memory

- ✍ A memory cache, sometimes called a *cache store* or *RAM cache*, is a portion of memory made of high-speed static RAM (SRAM) instead of the slower and cheaper dynamic RAM (DRAM) used for main memory.
- ✍ Memory caching is effective because most programs access the same data or instructions over and over. By keeping as much of this information as possible in SRAM, the computer avoids accessing the slower DRAM.



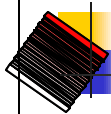
## Cache Memory.....

- ✍ Some memory caches are built into the architecture of microprocessors.
- ✍ The Intel 80486 microprocessor, for example, contains an 8K memory cache, and the Pentium has a 16K cache. Such *internal caches* are often called *Level 1 (L1) caches*.
- ✍ Most modern PCs also come with external cache memory, called *Level 2 (L2) caches*. These caches sit between the CPU and the DRAM. Like L1 caches, L2 caches are composed of SRAM but they are much larger.



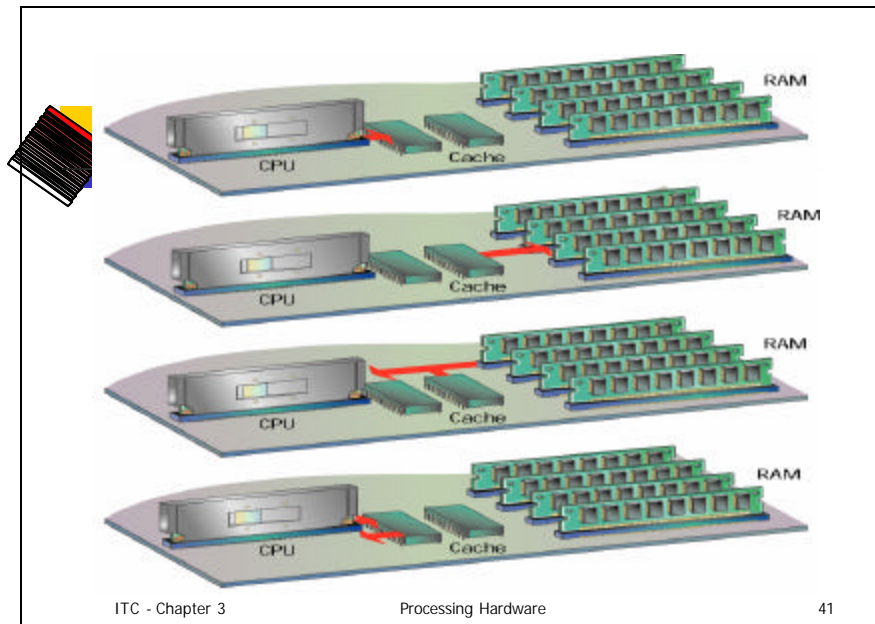
## Cache Memory....

- ✦ Disk caching works under the same principle as memory caching, but instead of using high-speed SRAM, a disk cache uses conventional main memory.
- ✦ The most recently accessed data from the disk (as well as adjacent sectors) is stored in a memory buffer. When a program needs to access data from the disk, it first checks the disk cache to see if the data is there.
- ✦ Disk caching can dramatically improve the performance of applications, because accessing a byte of data in RAM can be thousands of times faster than accessing a byte on a hard disk.



## Cache Memory...

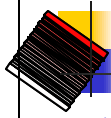
- ✦ Cache memory is high-speed memory that holds the most recent data and instructions that have been loaded by the CPU.
- ✦ Cache is located directly on the CPU or between the CPU and RAM, making it faster than normal RAM.
- ✦ CPU-resident cache is called **Level-1 (L1) cache**.
- ✦ External cache is called **Level-2 (L2) cache**.
- ✦ The amount of cache memory has a tremendous impact on the computer's speed.



## Cache Memory

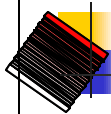
- ⚡ Pronounced *cash*, a special high-speed storage mechanism.
- ⚡ It can be either a reserved section of main memory or an independent high-speed storage device.
- ⚡ Two types of caching are commonly used in personal computers: *memory caching* and *disk caching*.

ITC - Chapter 3 Processing Hardware 42



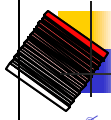
## Cache Memory

- ✍ A memory cache, sometimes called a *cache store* or *RAM cache*, is a portion of memory made of high-speed static RAM (SRAM) instead of the slower and cheaper dynamic RAM (DRAM) used for main memory.
- ✍ Memory caching is effective because most programs access the same data or instructions over and over. By keeping as much of this information as possible in SRAM, the computer avoids accessing the slower DRAM.



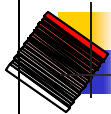
## Cache Memory.....

- ✍ Some memory caches are built into the architecture of microprocessors.
- ✍ The Intel 80486 microprocessor, for example, contains an 8K memory cache, and the Pentium has a 16K cache. Such *internal caches* are often called *Level 1 (L1) caches*.
- ✍ Most modern PCs also come with external cache memory, called *Level 2 (L2) caches*. These caches sit between the CPU and the DRAM. Like L1 caches, L2 caches are composed of SRAM but they are much larger.



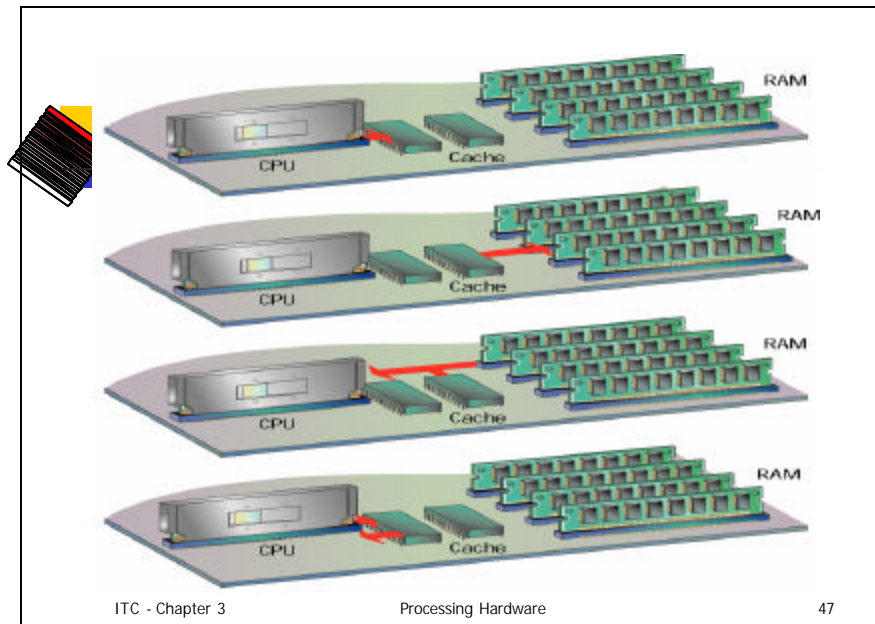
## Cache Memory....

- ✧ Disk caching works under the same principle as memory caching, but instead of using high-speed SRAM, a disk cache uses conventional main memory.
- ✧ The most recently accessed data from the disk (as well as adjacent sectors) is stored in a memory buffer. When a program needs to access data from the disk, it first checks the disk cache to see if the data is there.
- ✧ Disk caching can dramatically improve the performance of applications, because accessing a byte of data in RAM can be thousands of times faster than accessing a byte on a hard disk.



## Cache Memory...

- ✧ Cache memory is high-speed memory that holds the most recent data and instructions that have been loaded by the CPU.
- ✧ Cache is located directly on the CPU or between the CPU and RAM, making it faster than normal RAM.
- ✧ CPU-resident cache is called **Level-1 (L1) cache**.
- ✧ External cache is called **Level-2 (L2) cache**.
- ✧ The amount of cache memory has a tremendous impact on the computer's speed.

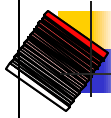


## A simple Example of Cache

To understand the basic idea behind a cache system, let's start with a super-simple example that uses a librarian to demonstrate caching concepts. Let's imagine a librarian behind his desk. He is there to give you the books you ask for. For the sake of simplicity, let's say you can't get the books yourself -- you have to ask the librarian for any book you want to read, and he fetches it for you from a set of stacks in a storeroom (the library of congress in Washington DC is set up this way). First, let's start with a librarian without cache.

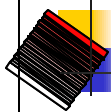
ITC - Chapter 3 Processing Hardware 48





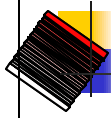
## A simple Example of Cache..

- ✍ The first customer arrives. He asks for the book Computer Communications. The librarian goes into the storeroom, gets the book, returns to the counter and gives the book to the customer. Later, the client comes back to return the book. The librarian takes the book and returns it to the storeroom. He then returns to his counter waiting for another customer. Let's say the next customer asks for Computer communications (you saw it coming...).



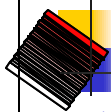
## A simple Example of Cache..

- ✍ The librarian then has to return to the storeroom to get the book he recently handled and give it to the client. Under this model, the librarian has to make a complete round trip to fetch every book -- even very popular ones that are requested frequently. Is there a way to improve the performance of the librarian?



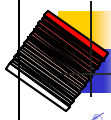
## Yes, use a cache!

Yes, there's a way -- we can put a **cache** on the librarian. Let's give the librarian a backpack into which he will be able to store 10 books (in computer terms, the librarian now has a 10-book cache). In this backpack, he will put the books the clients return to him, up to a maximum of 10. Let's use the prior example, but now with our new-and-improved caching librarian.



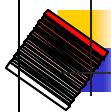
## Let start using a cache!

The day starts. The backpack of the librarian is empty. Our first client arrives and asks for Computer communications. No magic here -- the librarian has to go to the storeroom to get the book. He gives it to the client. Later, the client returns and gives the book back to the librarian. Instead of returning to the storeroom to return the book, the librarian puts the book in his backpack and stands there (he checks first to see if the bag is full -- more on that later). Another client arrives and asks for computer communications. Before going to the storeroom, the librarian checks to see if this title is in his backpack. He finds it! All he has to do is take the book from the backpack and give it to the client. There's no journey into the storeroom, so the client is served more efficiently.



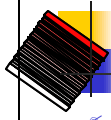
## Let start using a cache!

- What if the client asked for a title not in the cache (the backpack)? In this case, the librarian is less efficient with a cache than without one, because the librarian takes the time to look for the book in his backpack first. One of the challenges of cache design is to minimize the impact of cache searches, and modern hardware has reduced this time delay to practically zero. Even in our simple librarian example, the latency time (the waiting time) of searching the cache is so small compared to the time to walk back to the storeroom that it is irrelevant. The cache is small (10 books), and the time it takes to notice a miss is only a tiny fraction of the time that a journey to the storeroom takes.



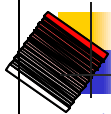
## What did u gain from this example

- From this example you can see several important facts about caching:
- Cache technology is the use of a faster but smaller memory type to accelerate a slower but larger memory type.
- When using a cache, you must check the cache to see if an item is in there. If it is there, it's called a **Cache Hit**. If not, it is called a **Cache Miss** and the computer must wait for a round trip from the larger, slower memory area.



## What did u gain from this example

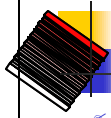
- A cache has some maximum size that is much smaller than the larger storage area.
- It is possible to have multiple layers of cache. With our librarian example, the smaller but faster memory type is the backpack, and the storeroom represents the larger and slower memory type. This is a one-level cache. There might be another layer of cache consisting of a shelf that can hold 100 books behind the counter. The librarian can check the backpack, then the shelf and then the storeroom. This would be a two-level cache.



## System Unit

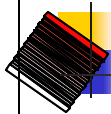
The box that contains the microcomputer's processing hardware and other components is called the system unit

- Power Supply
- Motherboard
- Microprocessor
- RAM
- ROM
- Cache, VRAM, Flash memory
- Ports
- Expansion Slots
- Bus Lines and PC slots and cards



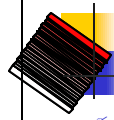
## Power Supply

- ✍ The component that supplies power to a computer. Most personal computers can be plugged into standard electrical outlets.
- ✍ The power supply then pulls the required amount of electricity and converts the AC current to DC current.
- ✍ It also regulates the voltage to eliminate spikes and surges common in most electrical systems.
- ✍ Not all power supplies, however, do an adequate voltage-regulation job, so a computer is always susceptible to large voltage fluctuations.



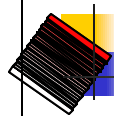
## Motherboard

- ✍ Also called System Board
- ✍ It is the main circuit board in the system unit.
- ✍ All or most of the components are plugged onto the motherboard

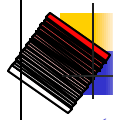


## Microprocessor (Intel)

- Since 1978, Intel's processors have evolved from the 8086 and the 8088 to the 80286, 80386, and 80486, to the Pentium family of processors. All are part of the 80x86 line.
- Intel's Pentium family of processors includes the Pentium, Pentium Pro, Pentium with MMX, Pentium II, Pentium III, Celeron, and Xeon processors.
- The earliest Intel processors included only a few thousand transistors. Today's Pentium processors include 9.5 million transistors or more.

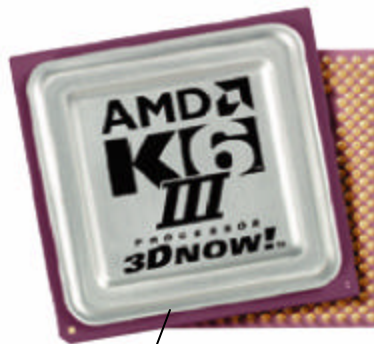
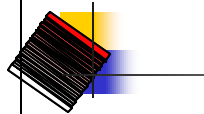


**Intel's  
Pentium III  
processor**

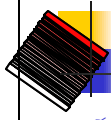


## Microprocessors (AMD)

- **Advanced Micro Devices (AMD)** was long known as a provider of lower-performance processors for use in low-cost computers.
- With its **K6** line of processors, AMD challenged Intel's processors in terms of both price and performance.
- With the K6-III processor, AMD broke the 600 MHz barrier, claiming the "fastest processor" title for the first time in IBM-compatible computers.

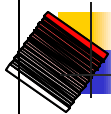


The AMD K6-III processor



## Microprocessors (Motorola)

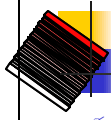
- ✍ Motorola makes the CPUs used in Macintosh and PowerPC computers.
- ✍ Macintosh processors use a different basic structural design (architecture) than IBM-compatible PC processors.
- ✍ With the release of the G3 and G4 PowerPC processors, Macintosh computers set new standards for price and performance.



## CISC and RISC processors

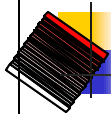
- ✍ Pronounced **sisk**, and stands for **Complex Instruction Set Computer**.
- ✍ Most personal computers, use a CISC architecture, in which the CPU supports as many as two hundred instructions.
- ✍ An alternative architecture, used by many workstations and also some personal computers, is **RISC (Reduced Instruction Set Computer)**, which supports fewer instructions.





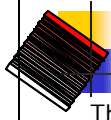
## CISC and RISC

- Most PCs are based on complex instruction set computing (CISC) chips which contain large instruction sets.
- Reduced instruction set computing (RISC) processors use smaller instruction sets. This enables them to process more instructions per second than (CISC) chips.
- RISC processors are found in Apple's PowerPC systems, as well as many H/PCs, workstations, minicomputers, and mainframes.



## RAM

- Pronounced ***ramm***, acronym for ***Random Access Memory***, a type of computer memory that can be accessed randomly; that is, any byte of memory can be accessed without touching the preceding bytes.
- RAM is the most common type of memory found in computers and other devices, such as printers.

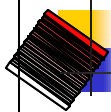


## RAM....

There are two basic types of RAM:

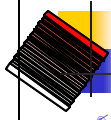
- ✦ **Dynamic RAM (DRAM)**
- ✦ **Static RAM (SRAM)**

- ✦ The two types differ in the technology they use to hold data, dynamic RAM being the more common type.
- ✦ Dynamic RAM needs to be refreshed thousands of times per second. Static RAM does not need to be refreshed, which makes it faster; but it is also more expensive than dynamic RAM. Both types of RAM are *volatile*, meaning that they lose their contents when the power is turned off.



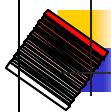
## RAM basics

- ✦ Similar to a micro-processor, a memory chip is an **Integrated Circuit (IC)** made of millions of transistors and capacitors.
- ✦ In the most common form of computer memory, **Dynamic Random Access Memory (DRAM)**, a transistor and a capacitor are paired to create a **memory cell**, which represents a single bit of data. The capacitor holds the bit of information -- a 0 or a 1 (see How Bits and Bytes Work for information on bits). The transistor acts as a switch that lets the control circuitry on the memory chip read the capacitor or change its state.



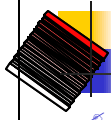
## RAM basics..

- ✍ A capacitor is like a small bucket that is able to store electrons. To store a 1 in the memory cell, the bucket is filled with electrons. To store a 0, it is emptied. The problem with the capacitor's bucket is that it has a leak. In a matter of a few milliseconds a full bucket becomes empty.
- ✍ Therefore, for dynamic memory to work, either the CPU or the **memory controller** has to come along and recharge all of the capacitors holding a 1 before they discharge. To do this, the memory controller reads the memory and then writes it right back. This refresh operation happens automatically thousands of times per second.



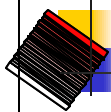
## RAM basics..

- ✍ The capacitor in a dynamic RAM memory cell is like a leaky bucket. It needs to be refreshed periodically or it will discharge to 0.
- ✍ This **refresh operation** is where dynamic RAM gets its name. Dynamic RAM has to be dynamically refreshed all of the time or it forgets what it is holding. The downside of all of this refreshing is that it takes time and slows down the memory.



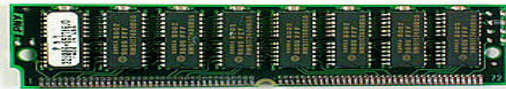
## RAM basics..

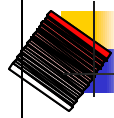
- Static RAM uses a completely different technology. In static RAM, a form of flip-flop holds each bit of memory. A flip-flop for a memory cell takes four or six transistors along with some wiring, but never has to be refreshed. This makes static RAM significantly faster than dynamic RAM. However, because it has more parts, a static memory cell takes a lot more space on a chip than a dynamic memory cell. Therefore you get less memory per chip, and that makes static RAM a lot more expensive.
- So static RAM is fast and expensive, and dynamic RAM is less expensive and slower. Therefore static RAM is used to create the CPU's speed-sensitive cache, while dynamic RAM forms the larger system RAM space.



## Memory Modules (RAM)

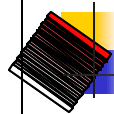
- SIMMs:** Acronym for **Single In-Line Memory Module**, a small circuit board that can hold a group of memory chips. Typically, SIMMs hold up to 8 (on Macintoshes) or 9 (on PCs) RAM chips. On PCs, the ninth chip is often used for parity error checking. Unlike memory chips, SIMMs are measured in bytes rather than bits. SIMMs are easier to install than individual memory chips.





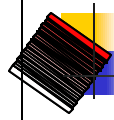
## Memory Modules (RAM)

- ✍ **DIMMs: Dual In-line Memory Module**
- ✍ Can hold upto 18 chips



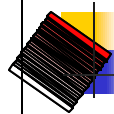
## ROM

- ✍ Pronounced **roh-m**, acronym for **Read-Only Memory**, computer memory on which data has been prerecorded. Once data has been written onto a ROM chip, it cannot be removed and can only be read.
- ✍ also known as **firmware**
- ✍ Unlike main memory (RAM), ROM retains its contents even when the computer is turned off. ROM is referred to as being *nonvolatile*, whereas RAM is *volatile*.



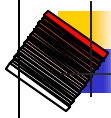
## ROM types

- ✍ ROM
- ✍ PROM
- ✍ EPROM
- ✍ EEPROM



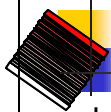
## ROM types...

- ✍ Creating ROM chips totally from scratch is time-consuming and very expensive in small quantities. For this reason, mainly, developers created a type of ROM known as **Programmable Read-Only Memory (PROM)**. Blank PROM chips can be bought inexpensively and coded by anyone with a special tool called a **programmer**.
- ✍ PROMs can only be programmed once



## ROM types...

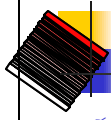
- ✍ **Erasable Programmable Read-Only Memory (EPROM)** chips can be rewritten many times.
- ✍ Erasing an EPROM requires a special tool that emits a certain frequency of ultraviolet (UV) light. EPROMs are configured using an EPROM programmer that provides voltage at specified levels depending on the type of EPROM used.



## ROM types...

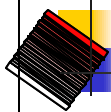
In EEPROMs:

- ✍ The chip does not have to be removed to be rewritten.
- ✍ The entire chip does not have to be completely erased to change a specific portion of it.
- ✍ Changing the contents does not require additional dedicated equipment.



## ROM types...

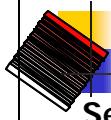
- Though EPROMs are a big step up from PROMs in terms of reusability, they still require dedicated equipment and a labor-intensive process to remove and reinstall them each time a change is necessary. Also, changes cannot be made incrementally to an EPROM; the whole chip must be erased. **Electrically Erasable Programmable Read-Only Memory (EEPROM)** chips remove the biggest drawbacks of EPROMs.



## Ports

- An interface on a computer to which you can connect a device.
- Personal computers have various types of ports. Internally, there are several ports for connecting disk drives, display screens, and keyboards. Externally, personal computers have ports for connecting modems, printers, mice, and other peripheral devices.

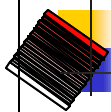




## Types of Ports

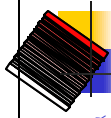
### Serial Ports

- Considered to be one of the most basic external connections to a computer, the **serial port** has been an integral part of most computers for more than 20 years. Although many of the newer systems have done away with the serial port completely in favor of USB connections, most modems still use the serial port, as do some printers, PDAs and digital cameras.



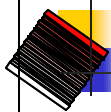
## Serial Ports

- The name "serial" comes from the fact that a serial port "serializes" data. That is, it takes a byte of data and transmits the 8 bits in the byte one at a time.
- The advantage is that a serial port needs only one wire to transmit the 8 bits (while a parallel port needs 8).
- The disadvantage is that it takes 8 times longer to transmit the data than it would if there were 8 wires. Serial ports lower cable costs and make cables smaller.



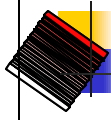
## Serial Ports

- ✦ **Serial ports**, also called **Communication (COM) Ports**, are **Bi-Directional**.
- ✦ Bi-directional communication allows each device to receive data as well as transmit it. Serial devices use different pins to receive and transmit data -- using the same pins would limit communication to **half-duplex**, meaning that information could only travel in one direction at a time. Using different pins allows for **full-duplex** communication, in which information can travel in both directions at once.



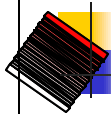
## Parallel Ports

- ✦ When using a parallel port, the computer sends the data 1 byte at a time (8 bits in parallel, as opposed to 8 bits serially as in a serial port).
- ✦ Parallel ports are used to connect a host of popular computer peripherals:
  - ✦ Printers
  - ✦ Scanners
  - ✦ CD-writers
  - ✦ External hard drives
  - ✦ Network adapters
  - ✦ Tape backup drives



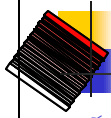
## SCSI Ports

- ✍ Provides an interface for transferring data at high speeds for SCSI-compatible devices.
- ✍ Like Hard disks, scanners and CDROMs



## Video Adapter Ports

- ✍ Used to connect video display monitor
- ✍ Game Ports
- ✍ Used to attach joysticks



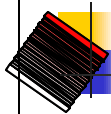
## Infrared Ports

- ✍ This enables you to transfer data from one device to another without any cables. For example, if both your laptop computer and printer have Infrared ports, you can simply put your computer in front of the printer and output a document, without needing to connect the two with a cable.
- ✍ Infrared ports support roughly the same transmission rates as traditional parallel ports. The only restrictions on their use is that the two devices must be within a few feet of each other and there must be a clear line of sight between them.

ITC - Chapter 3

Processing Hardware

87



## Coprocessors

- ✍ A special-purpose processing unit that assists the CPU in performing certain types of operations. For example, a **math coprocessor** performs mathematical computations, particularly floating-point operations.
- ✍ **Math coprocessors are also called numeric and floating-point coprocessors.**
- ✍ In addition to math coprocessors, there are also *graphics coprocessors* for manipulating graphic images.
- ✍ These are often called **accelerator boards**.

ITC - Chapter 3

Processing Hardware

88