Introduction to Computer Science
Lesson 2

BSc in Computer Science
University of New York, Tirana

Assoc. Prof. Marenglen Biba

Chapter 1: Data Storage

- 1.1 Bits and Their Storage
- 1.2 Main Memory
- 1.3 Mass Storage
- 1.4 Representing Information as Bit Patterns
- 1.5 The Binary System

Bits and Bit Patterns

- **Bit:** Binary Digit (0 or 1)
- Bit Patterns are used to represent information.
  - Numbers
  - Text characters
  - Images
  - Sound
  - And others

Boolean Operations

- **Boolean Operation:** An operation that manipulates one or more true/false values
- Specific operations
  - AND
  - OR
  - XOR (exclusive or)
  - NOT

Figure 1.1 **The Boolean operations AND, OR, and XOR (exclusive or)**

The AND operation

0	0	1	1
AND 0	AND 1	AND 0	AND 1
0	0	0	1

The OR operation

0	0	1	1
OR 0	OR 1	OR 0	OR 1
0	1	1	1

The XOR operation

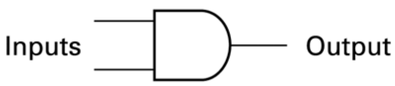
0	0	1	1
XOR 0	XOR 1	XOR 0	XOR 1
0	1	1	0

Gates

- **Gate:** A device that computes a Boolean operation
  - Often implemented as (small) electronic circuits
  - Provide the building blocks from which computers are constructed
  - VLSI (Very Large Scale Integration)

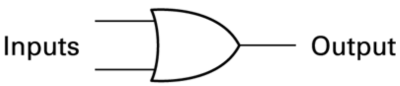
Figure 1.2 **A pictorial representation of AND, OR, XOR, and NOT gates as well as their input and output values**

AND



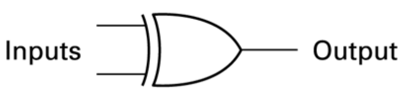
Inputs	Output
0 0	0
0 1	0
1 0	0
1 1	1

OR



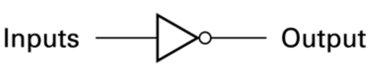
Inputs	Output
0 0	0
0 1	1
1 0	1
1 1	1

XOR



Inputs	Output
0 0	0
0 1	1
1 0	1
1 1	0

NOT



Inputs	Output
0	1
1	0

Flip-flops

- **Flip-flop:** A circuit built from gates that can **store** one bit.
  - One input line is used to set its stored value to 1
  - One input line is used to set its stored value to 0

Flip or flop between two values

- A **flip-flop** is a circuit that produces an output value of 0 or 1, which **remains constant** until a temporary pulse from another circuit causes it to shift to the other value.

- In other words, the output will **flip or flop between two values** under control of external stimuli.

Figure 1.3 A simple flip-flop circuit

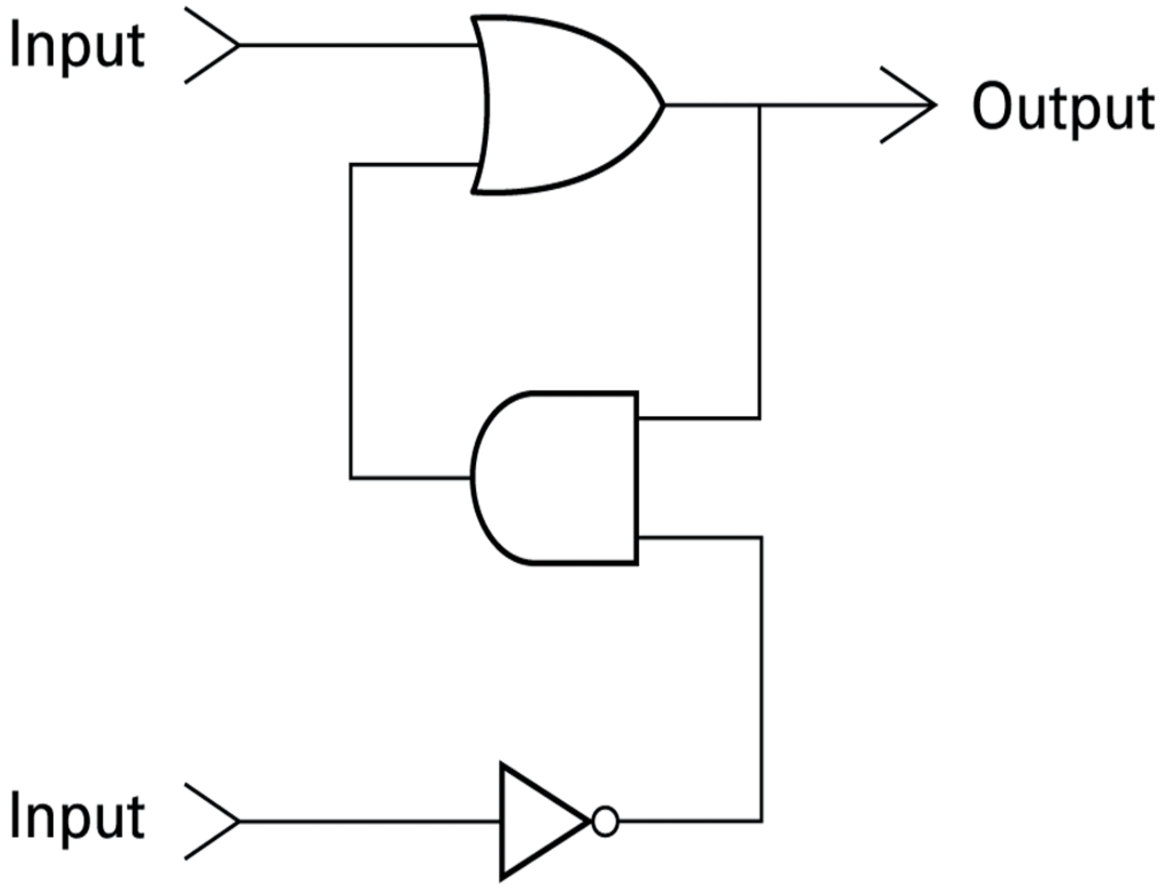


Figure 1.4 Setting the output of a flip-flop to 1

a. 1 is placed on the upper input.

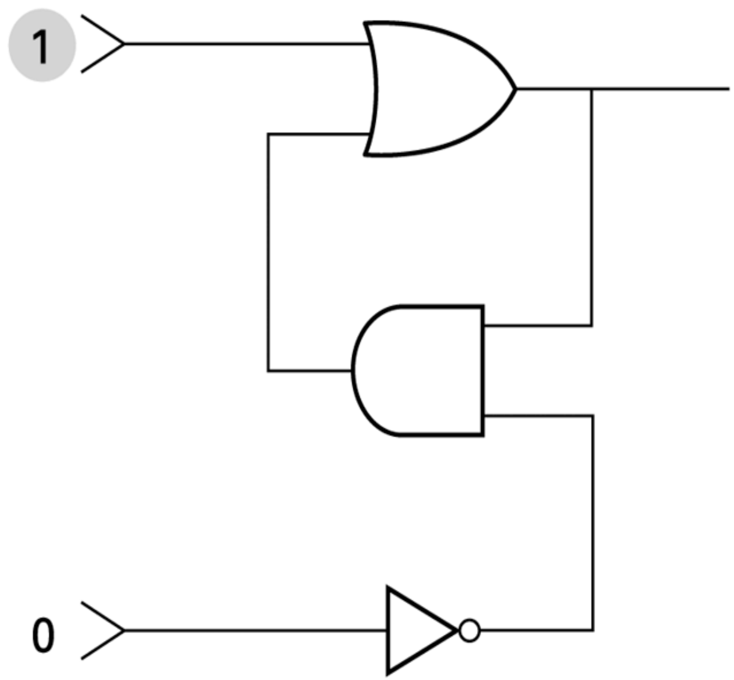


Figure 1.4 **Setting the output of a flip-flop to 1 (continued)**

b. This causes the output of the OR gate to be 1 and, in turn, the output of the AND gate to be 1.

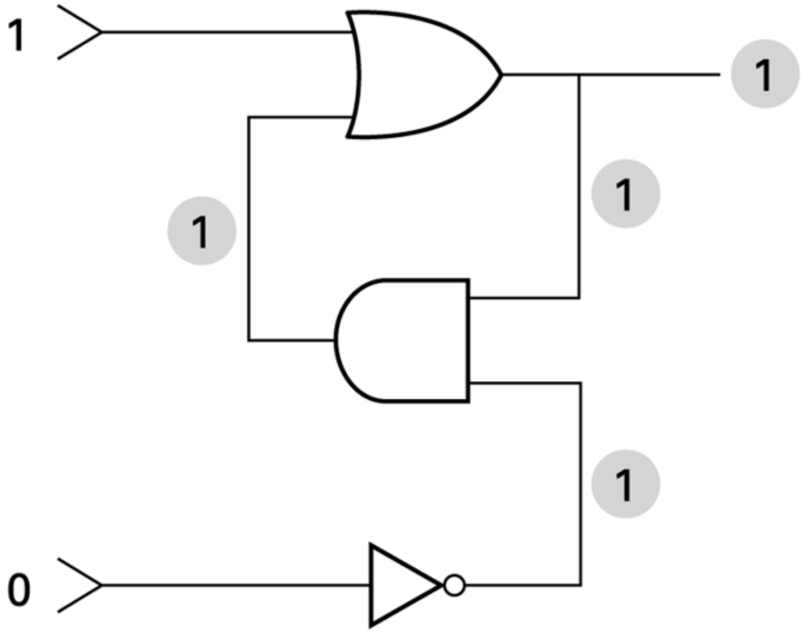
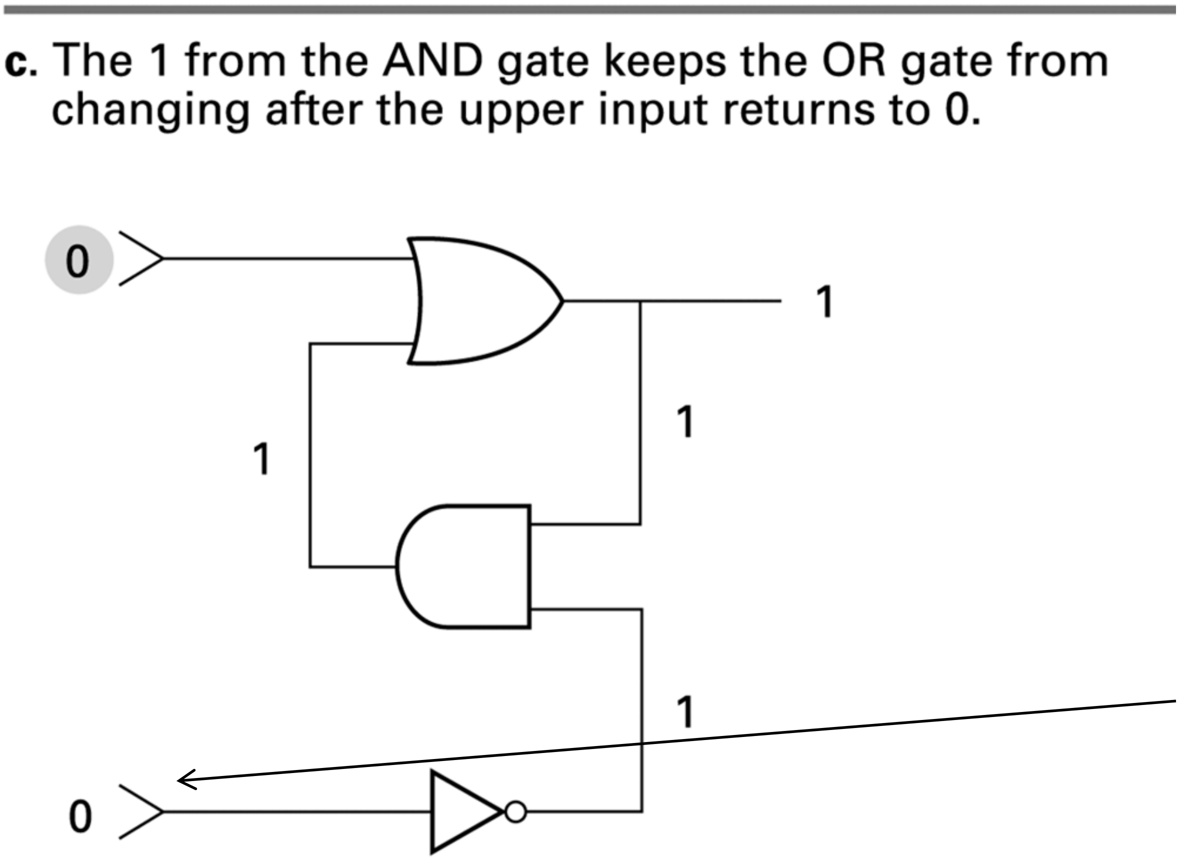


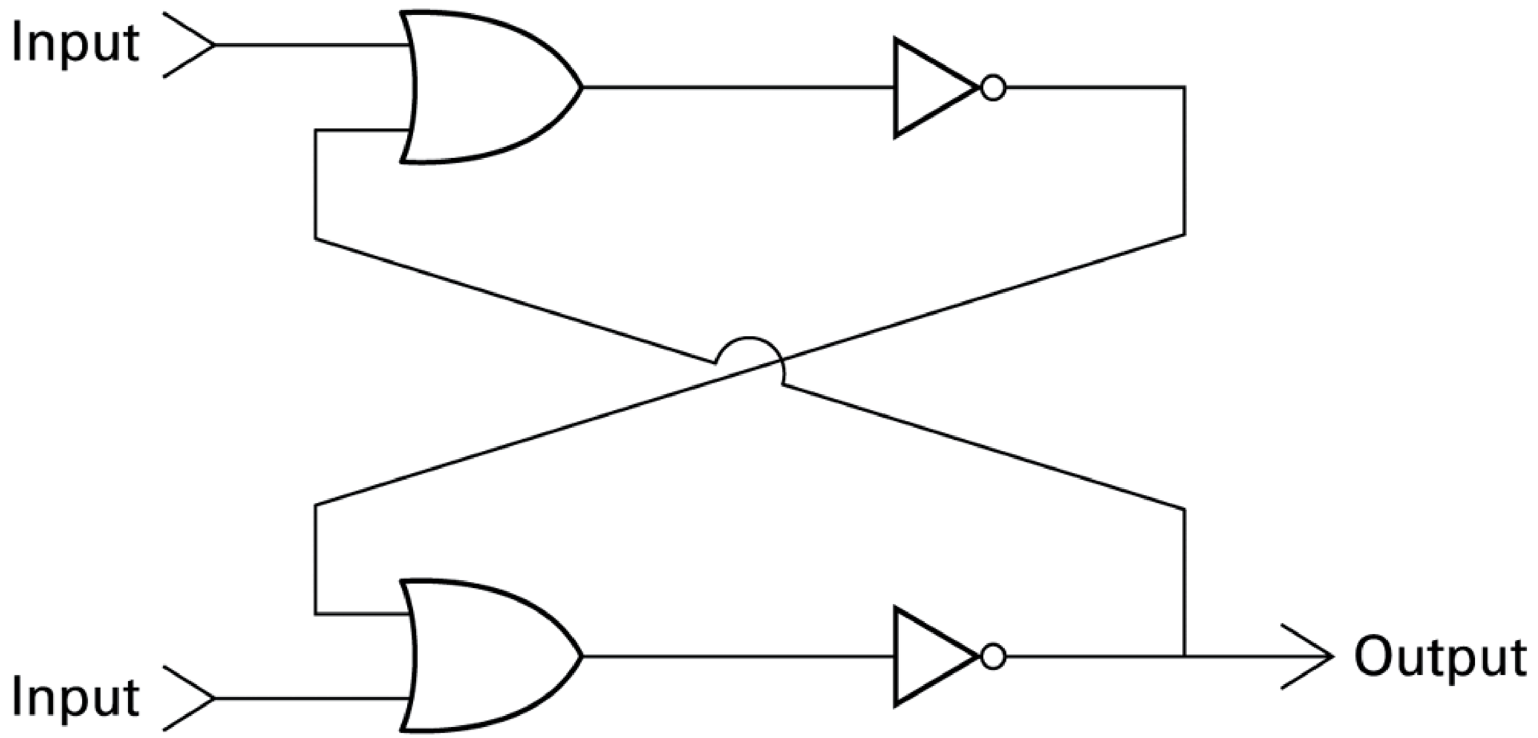
Figure 1.4 **Setting the output of a flip-flop to 1 (continued)**



As long as both inputs in the circuit remain 0, the output (whether 0 or 1) will not change.

However, temporarily placing a 1 on the upper input will **force the output to be 1**, whereas temporarily placing a 1 on the lower input will **force the output to be 0.**

Figure 1.5 **Another way of constructing a flip-flop**



Hexadecimal Notation

- **Hexadecimal notation:** A shorthand notation for long bit patterns
  - Divides a pattern into groups of four bits each
  - Represents each group by a single symbol
- Example: 10100011 becomes A3

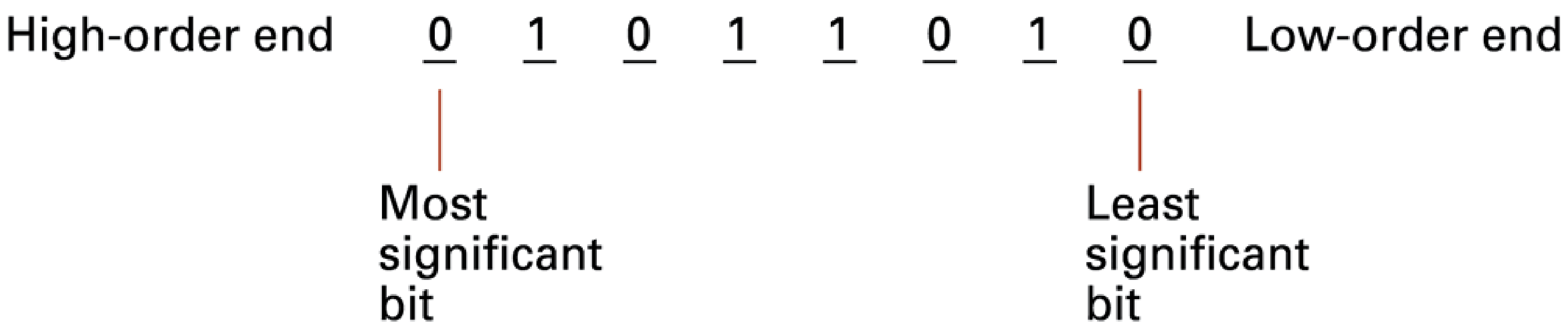
Figure 1.6 **The hexadecimal coding system**

Bit pattern	Hexadecimal representation
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

Main Memory Cells

- **Cell:** A unit of main memory (typically 8 bits which is one **byte**)
  - **Most significant bit:** the bit at the left (high-order) end of the conceptual row of bits in a memory cell
  - **Least significant bit:** the bit at the right (low-order) end of the conceptual row of bits in a memory cell

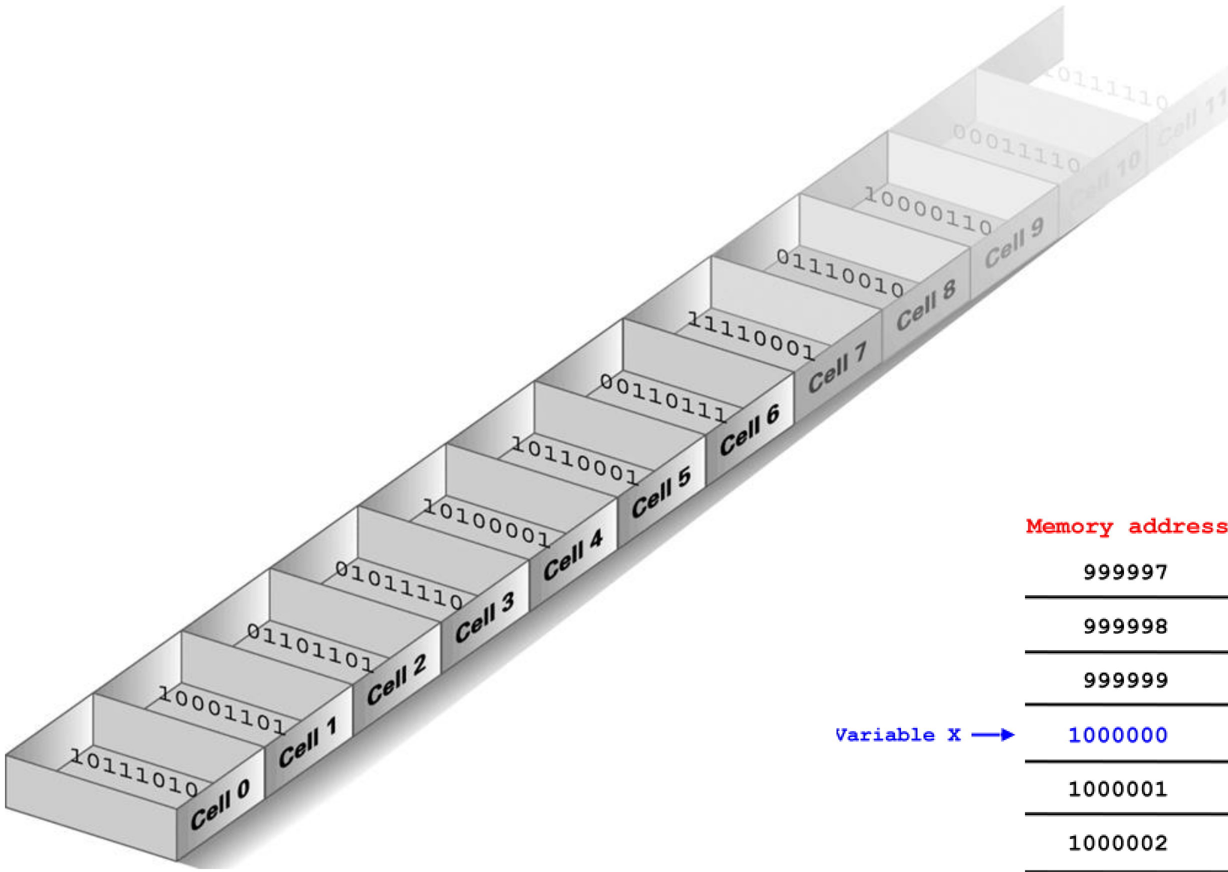
Figure 1.7 The organization of a byte-size memory cell



Main Memory Addresses

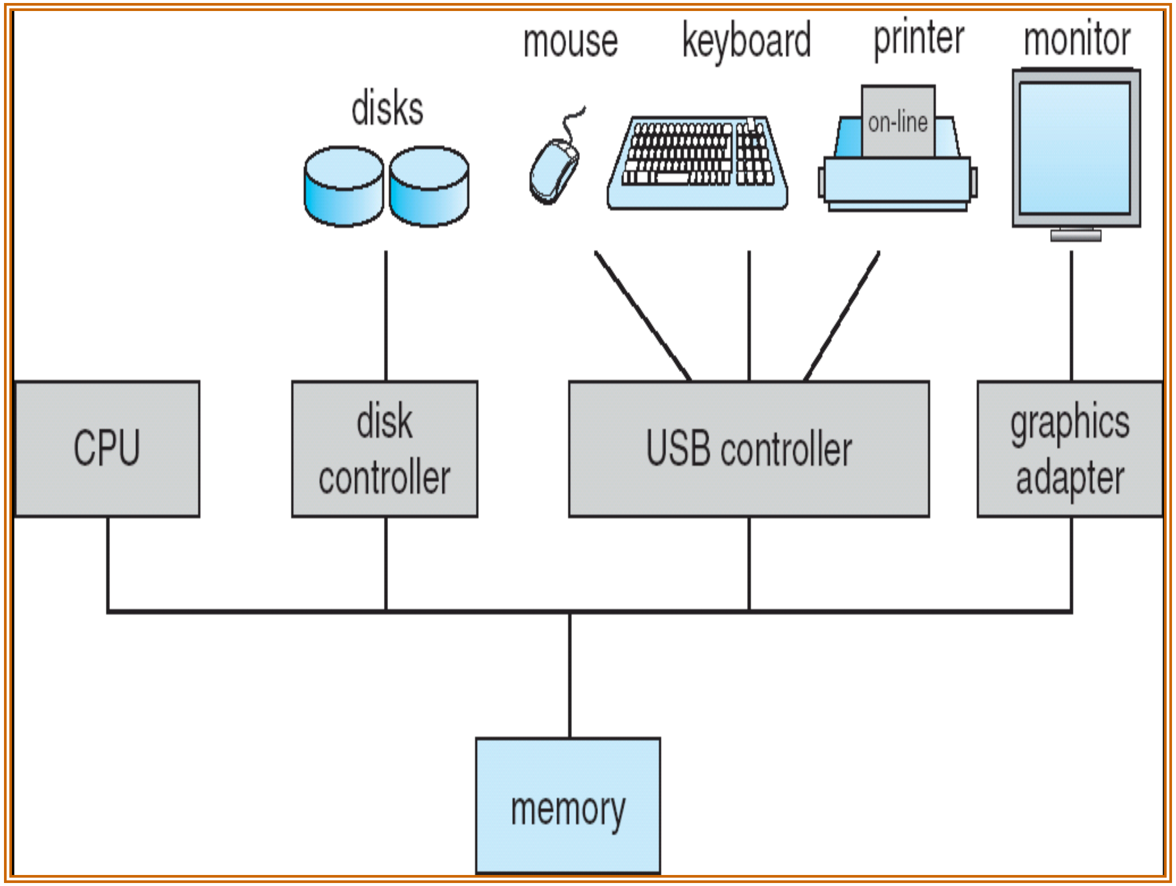
- **Address:** A “name” that uniquely identifies one cell in the computer’s main memory
  - The names are actually numbers.
  - These numbers are assigned consecutively starting at zero.
  - Numbering the cells in this manner associates an order with the memory cells.

Figure 1.8 **Memory cells arranged by address**



	Memory address	Value
	999997	0
	999998	0
	999999	0
Variable X →	1000000	15
	1000001	0
	1000002	0
	1000003	0
	.	.
	.	.
	.	.

General computer architecture



Memory

- *Memory* is one of the most important but perhaps most misunderstood computer components.
- Its function is often mistaken for that of hard drive space.
- Computers use several types of memory, each with a different function and different physical form.
- Typically, when people discuss memory, they are referring to ***random access memory***, or ***RAM***.

Memory Terminology

- **Random Access Memory (RAM):** Memory in which individual cells can be easily accessed in any order
- **Dynamic Memory (DRAM)**
  - A type of random-access memory that stores each bit of data in a separate capacitor within an integrated circuit.
- **RAM** memories are volatile memories
  - loses its data quickly when power is removed

Base 10: Keep in mind

Exp.	Explicit	Prefix	Exp.	Explicit	Prefix
10^{-3}	0.001	milli	10^{3}	1,000	Kilo
10^{-6}	0.000001	micro	10^{6}	1,000,000	Mega
10^{-9}	0.000000001	nano	10^{9}	1,000,000,000	Giga
10^{-12}	0.000000000001	pico	10^{12}	1,000,000,000,000	Tera
10^{-15}	0.000000000000001	femto	10^{15}	1,000,000,000,000,000	Peta
10^{-18}	0.000000000000000001	atto	10^{18}	1,000,000,000,000,000,000	Exa
10^{-21}	0.000000000000000000001	zepto	10^{21}	1,000,000,000,000,000,000,000	Zetta
10^{-24}	0.000000000000000000000001	yocto	10^{24}	1,000,000,000,000,000,000,000,000	Yotta

BASE 10

Measuring Memory Capacity: powers of two

- **Kilobyte:** 2^{10} bytes = 1024 bytes
  - Example: 3 KB = 3 times1024 bytes

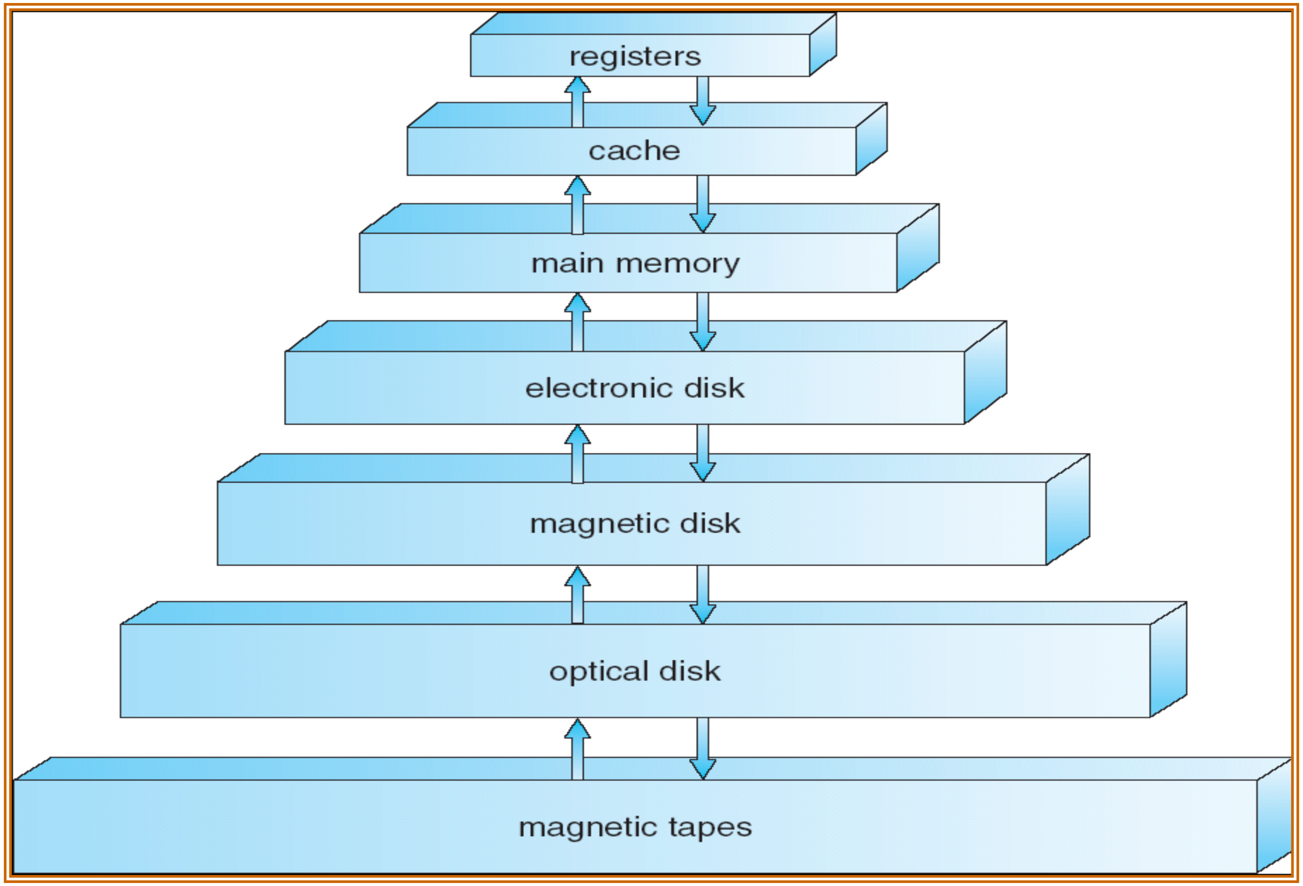
- **Megabyte:** 2^{20} bytes = 1,048,576 bytes
  - Example: 3 MB = 3 times 1,048,576 bytes

- **Gigabyte:** 2^{30} bytes = 1,073,741,824 bytes
  - Example: 3 GB = 3 times 1,073,741,824 bytes

Mass Storage

- Typically larger than main memory
- Typically less volatile than main memory
- Typically slower than main memory

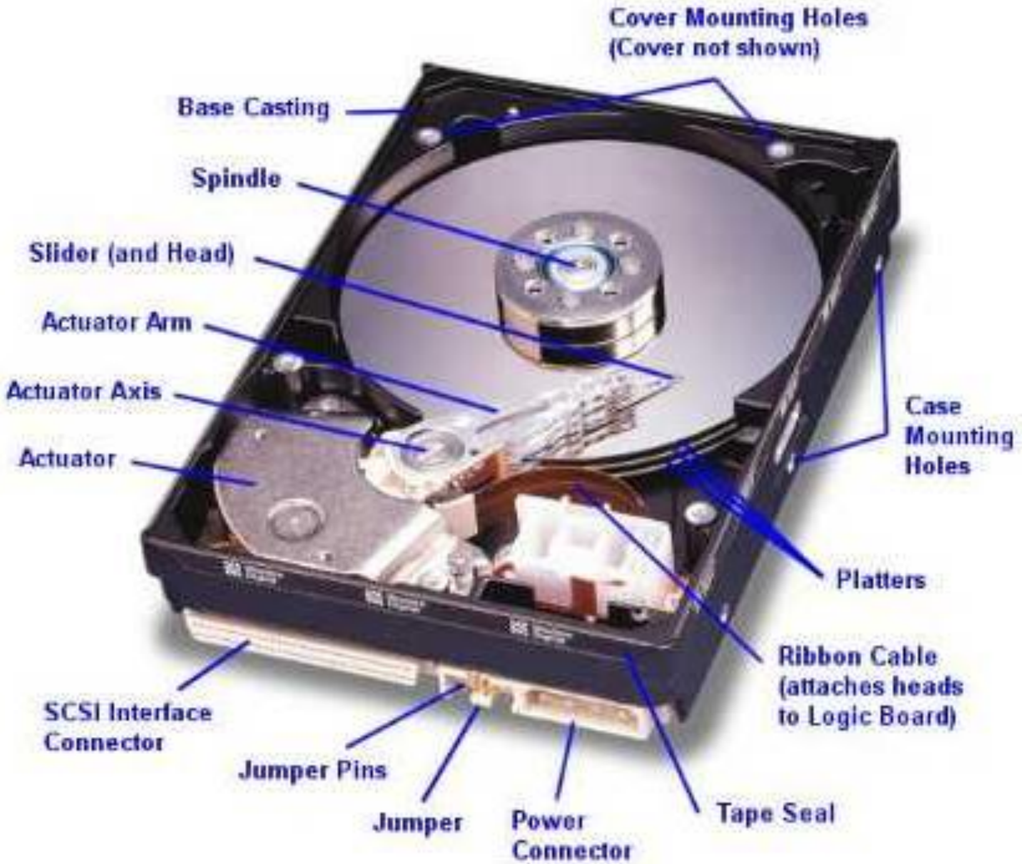
Memory hierarchy



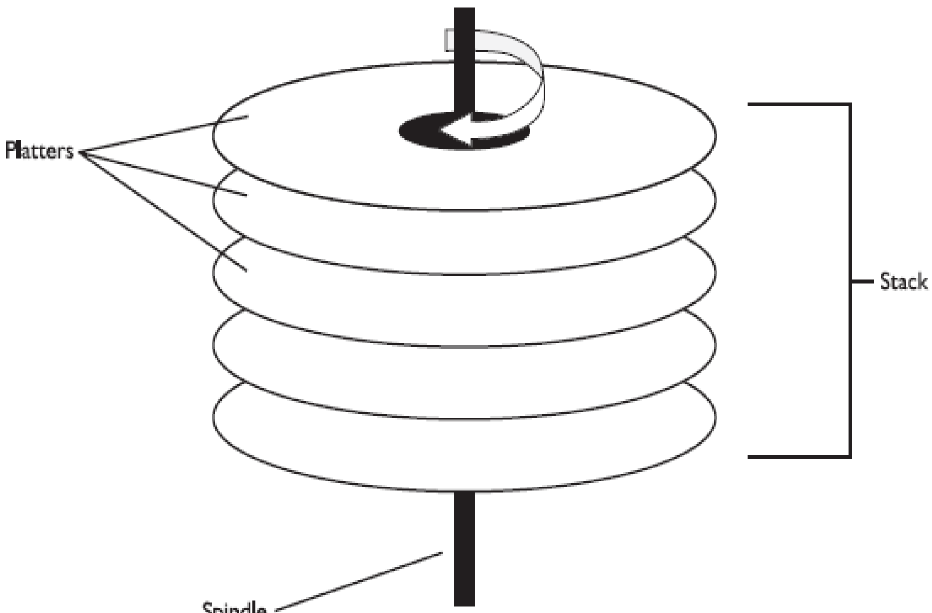
Mass Storage Systems

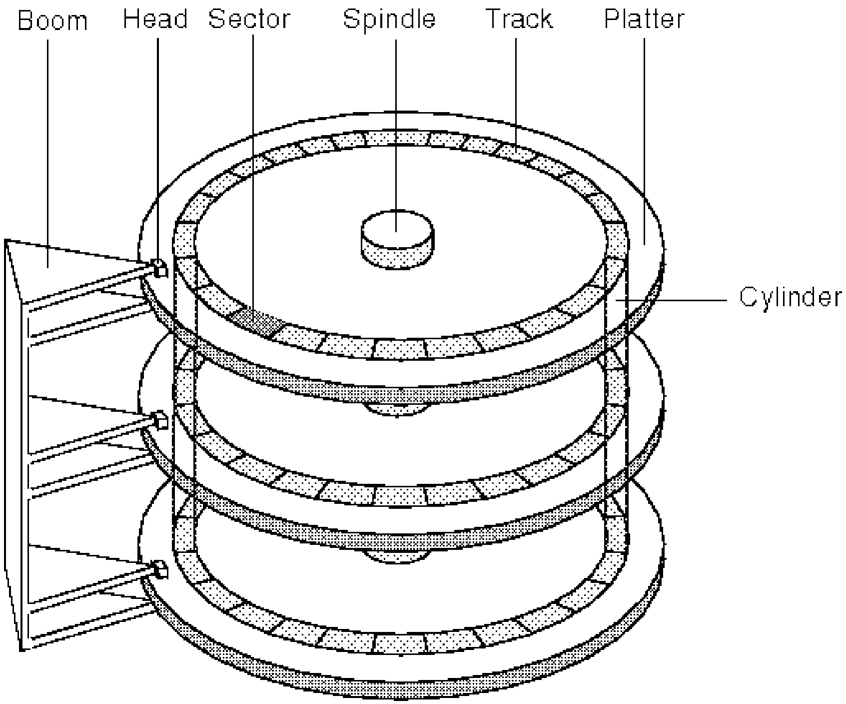
- Magnetic Systems
  - Disk
  - Tape
- Optical Systems
  - CD
  - DVD
- Flash Technology
  - Flash Drives
  - Secure Digital (SD) Memory Card

Magnetic disk (hard disk)



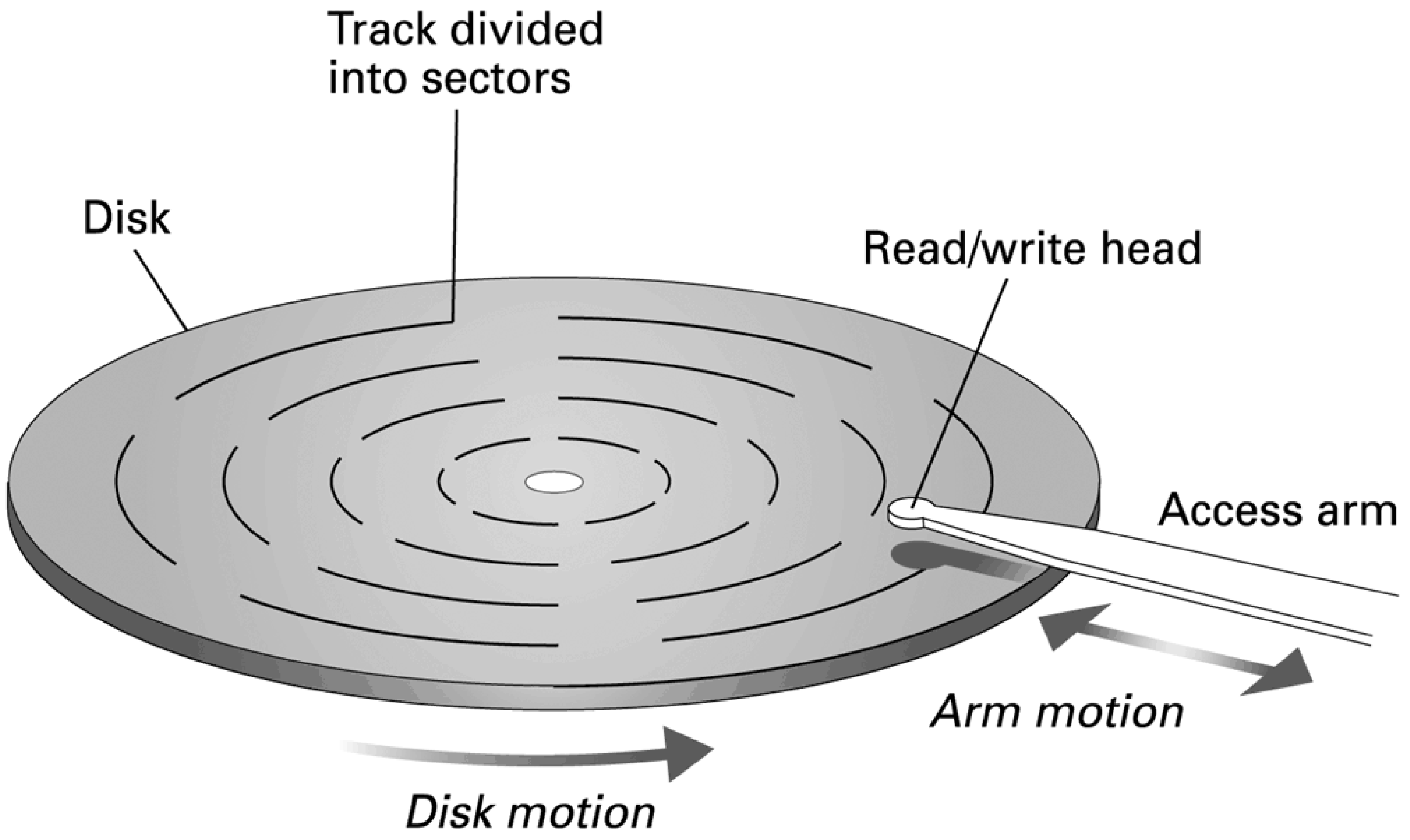
Hard Disk





- The stack of platters is attached through its center to a rotating pole, called a *spindle*.
- Each side of each platter can hold data and has its own read/write head.
- The read/write heads all move as a single unit back and forth along the stack.

Figure 1.9 **A magnetic disk storage system**



Formatting a disk

- The location of tracks and sectors is **not a permanent** part of a disk's physical structure.
- Instead, they are marked magnetically through a process called **formatting** (or **initializing**) the disk.
- This process is usually performed by the disk's manufacturer, resulting in what are known as **formatted disks.**
- Most computer systems can also perform this task.
- Thus, if the format information on a disk is **damaged**, the disk can be **reformatted**, although this process destroys all the information that was previously recorded on the disk.

Figure 1.10 Magnetic tape storage

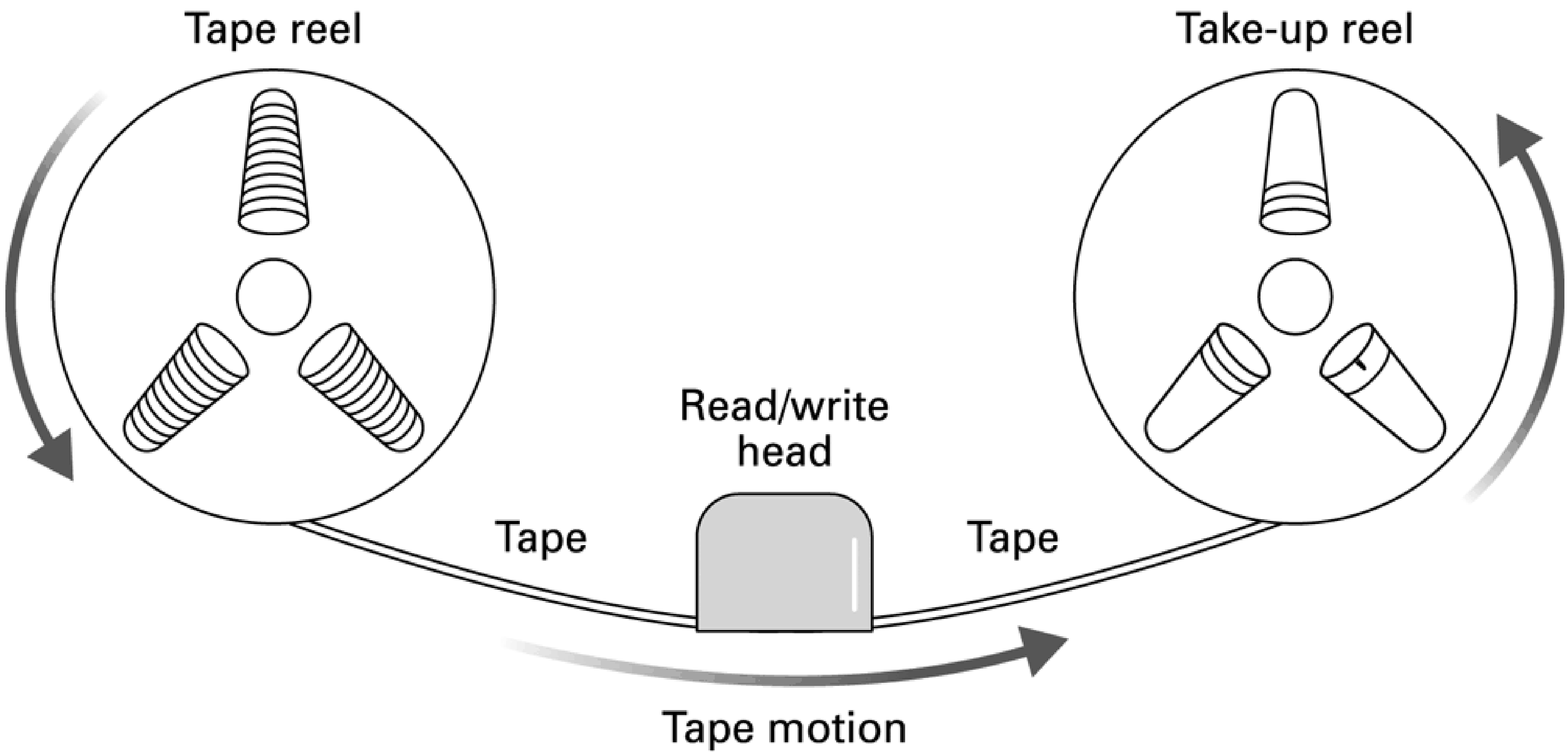
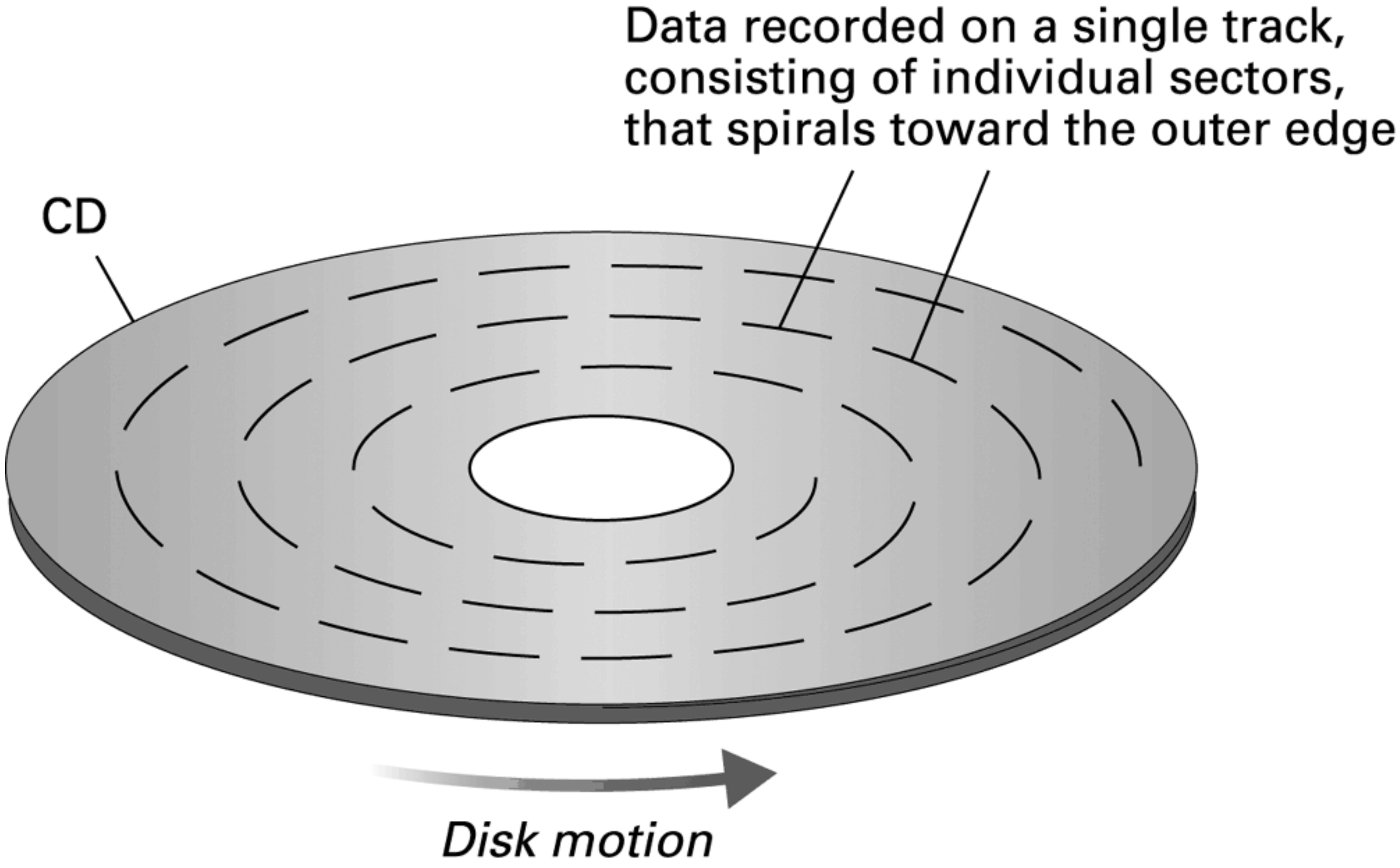


Figure 1.11 **CD storage**



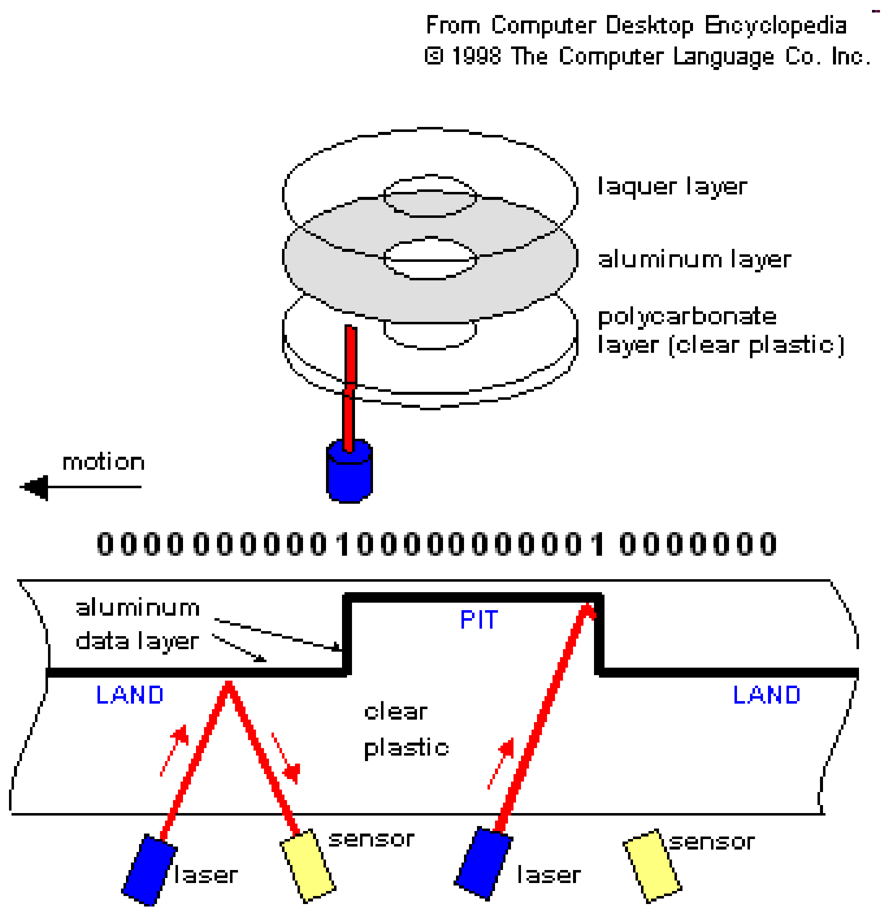
Optical systems

- Compact disks (CD) are 12 centimeters (approximately 5 inches) in diameter and consist of reflective material covered with a clear protective coating.
- Information is recorded on them by **creating variations in their reflective surfaces.**
- This information can then be **retrieved by means of a laser beam** that monitors irregularities on the reflective surface of the CD as it spins.
- CD technology was originally applied to audio recordings using a recording format known as CD-DA (compact disk-digital audio), and the CDs used today for computer data storage use essentially the same format.
- In particular, information on these CDs is stored on a single track that spirals around the CD like a groove in an old-fashioned record, however, unlike old-fashioned records, the track on a CD spirals from the inside out.
- This **track is divided into units called sectors**, each with its own identifying markings and a capacity of 2KB of data, which equates to 1/75 of a second of music in the case of audio recordings.

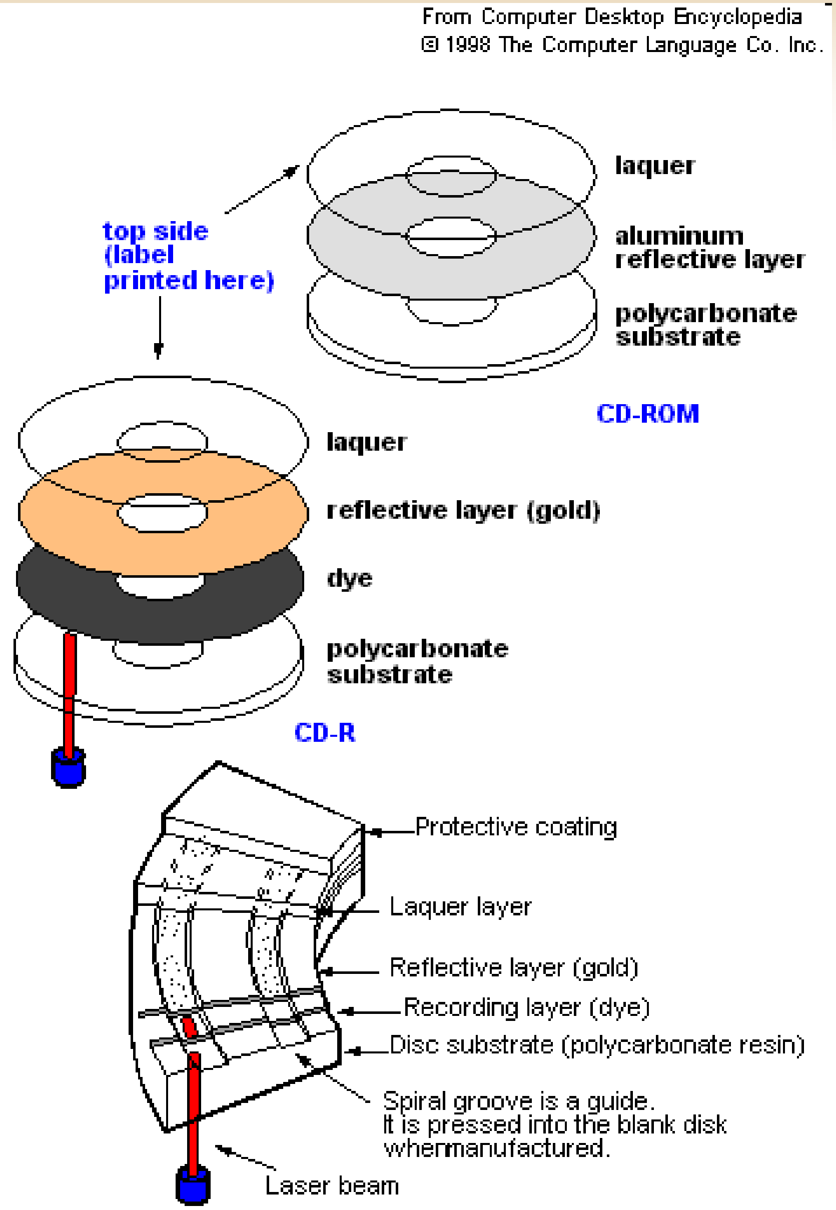
CD-ROM

- ***Compact disc–Read-Only Memory (CD-ROM)*** offers a balance between the portability of a floppy disk and the capacity of a hard drive.
- CD-ROMs are composed of a hard medium that contains very small depressed and raised areas, called ***pits*** and ***lands***, respectively.
- CD-ROM drives read data from the CD using a laser instead of a read/write head.
- CD-ROMs can hold roughly 650MB of data and generally cannot be written to, except in the case of CD-Rs (recordable) or CD-RWs (rewritable).
- Data can be accessed faster from a CD than a floppy disk but much more slowly than from a hard drive.

CD-ROM


From Computer Desktop Encyclopedia
© 1998 The Computer Language Co. Inc.
laquer layer
aluminum layer
polycarbonate
layer (clear plastic)
motion
00000000001000000000001 0000000
aluminum
data layer
PIT
LAND
LAND
clear
plastic
laser
sensor
laser
sensor

CD-R Layers

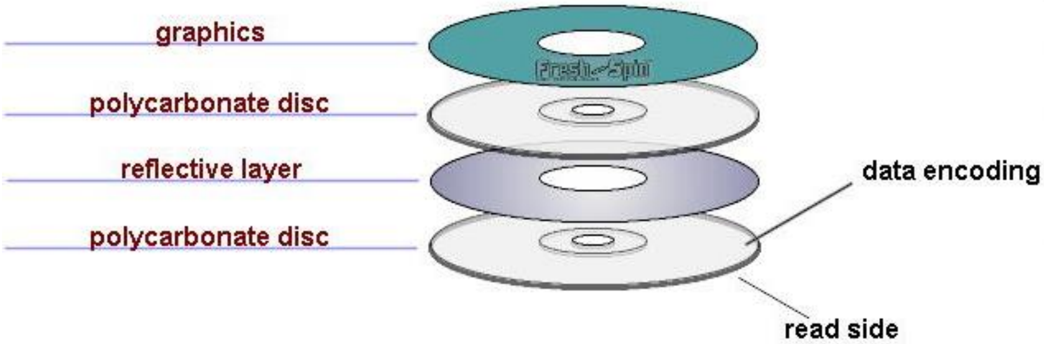


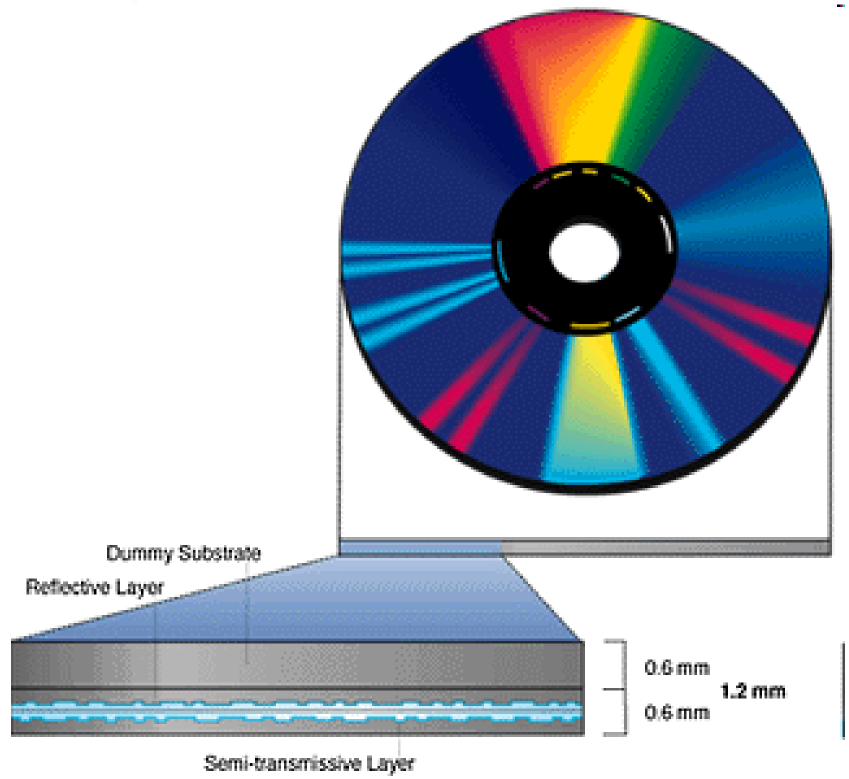
From Computer Desktop Encyclopedia
© 1998 The Computer Language Co. Inc.
top side
(label
printed here)
laquer
aluminum
reflective layer
polycarbonate
substrate
CD-ROM
laquer
reflective layer (gold)
dye
polycarbonate
substrate
CD-R
Protective coating
Laquer layer
Reflective layer (gold)
Recording layer (dye)
Disc substrate (polycarbonate resin)
Spiral groove is a guide.
It is pressed into the blank disk
whenmanufactured.
Laser beam

DVD

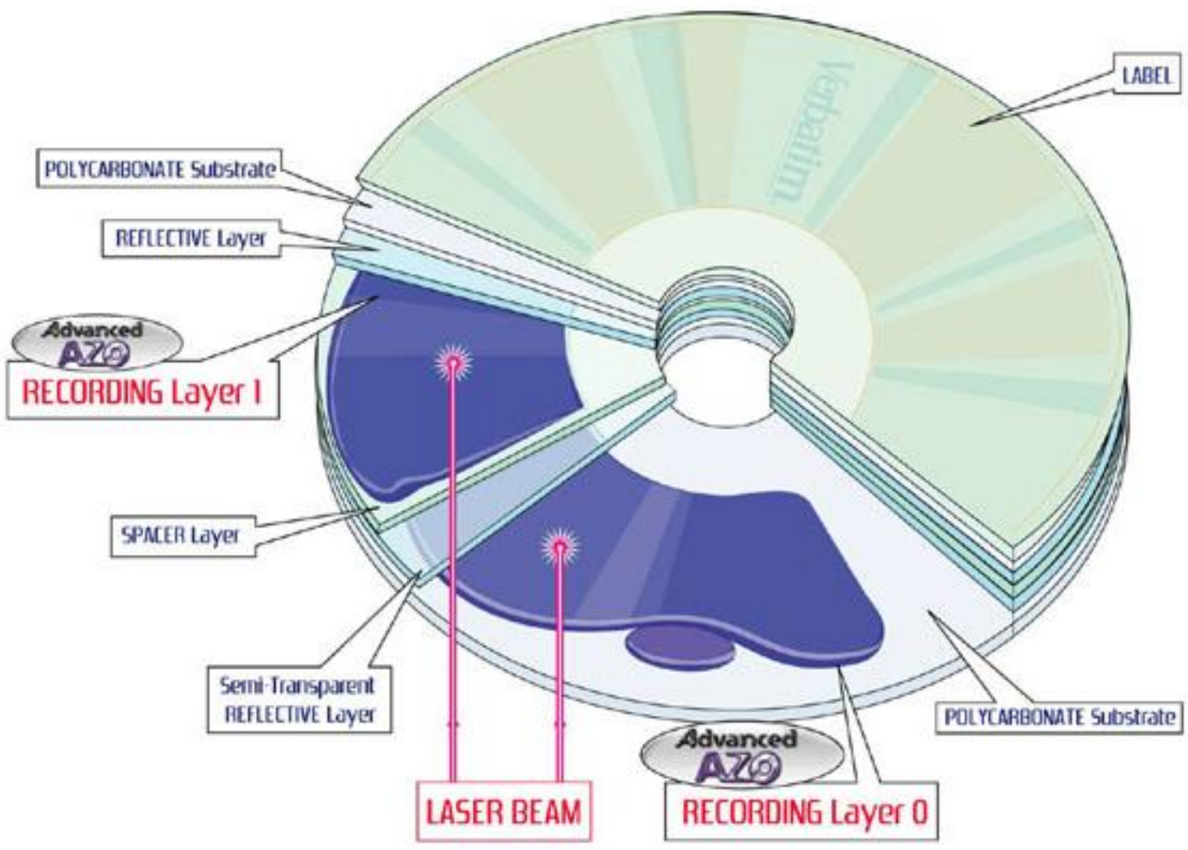
- Digital versatile discs (DVDs) are similar in technology to CD-ROMs but have a higher storage density.
- DVDs are used extensively for video storage as well as large amounts of data storage.
- DVD-recordable (DVD-R) can be recorded on only once.
- Unlike a CD-RW, which can have multiple recording sessions, a DVD-R can have only one session.
- DVDs can typically store on the order of 5 to 8GB of data.
- There are new types of DVD devices coming on the market every day.

DVD - Layers


graphics
polycarbonate disc
reflective layer
polycarbonate disc
data encoding
read side


Dummy Substrate
Reflective Layer
0.6 mm
1.2 mm
0.6 mm
Semi-transmissive Layer

DVD - Layers


LABEL
POLYCARBONATE Substrate
REFLECTIVE Layer
Advanced AZO
RECORDING Layer 1
SPACER Layer
Semi-Transparent REFLECTIVE Layer
LASER BEAM
Advanced AZO
RECORDING Layer 0
POLYCARBONATE Substrate

Flash Memory

- A common property of mass storage systems based on magnetic or optic technology is that physical motion**,** such as spinning disks, moving read/write heads, and aiming laser beams, is required to store and retrieve data.
- This means that data storage and retrieval is **slow** compared to the speed of electronic circuitry.
- **Flash memory** technology has the potential of alleviating this drawback.
- In a flash memory system, **bits are stored by sending electronic signals directly to the storage medium where they cause electrons to be trapped in tiny chambers of silicon dioxide**, thus altering the characteristics of small electronic circuits.
- Since these chambers are able to hold their captive electrons for many years, this technology is suitable for off-line storage of data.

Flash memory

- Although data stored in flash memory systems can be accessed in small byte-size units as in RAM applications, current technology dictates that stored data be erased in large blocks.
- Moreover, **repeated erasing slowly damages** the silicon dioxide chambers, meaning that current flash memory technology is not suitable for general main memory applications where its contents might be **altered many times a second.**
- However, in those applications in which **alterations can be controlled** to a reasonable level, such as in digital cameras, cellular telephones, and hand-held PDAs, flash memory has become the mass storage technology of choice.
- Indeed, since flash memory is **not sensitive to physical shock** (in contrast to magnetic and optic systems) its potential in portable applications is high.

Flash drives

- Flash memory devices called **flash drives**, with capacities of up to a few GB, are available for general mass storage applications.
- These units are packaged in **small plastic cases** approximately three inches long with a removable cap on one end to protect the unit's electrical connector when the drive is off-line.
- The high capacity of these **portable units** as well as the fact that they are easily **connected** to and **disconnected** from a computer make them ideal for off-line data storage.
- However, the **vulnerability** of their tiny storage chambers dictates that they are not as reliable as optical disks for truly long term applications.

Files

- **File**: A unit of data stored in mass storage system
- Physical record versus Logical record
- **Buffer**: A memory area used for the temporary storage of data (usually as a step in transferring the data)

Files

- Information stored in a mass storage system is conceptually grouped into large units called **files**.
  - A typical file may consist of a complete text document, a photograph, a program, a music recording, or a collection of data about the employees in a company.
- Mass storage devices dictate that these files be stored and retrieved in smaller, **multiple byte units**.
  - For example, a file stored on a magnetic disk must be manipulated by sectors, each of which is a fixed predetermined size.
- A block of data conforming to the specific characteristics of a storage device is called a **physical record**.
- Thus, a large file stored in mass storage will typically consist of many **physical records**.

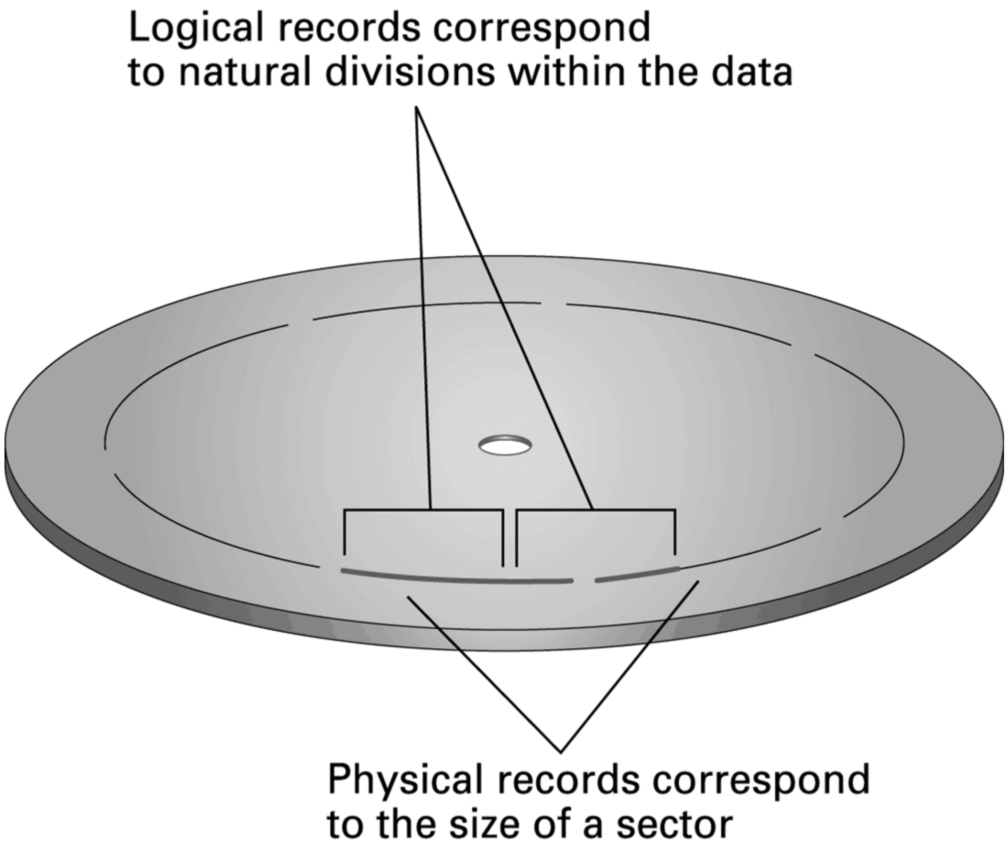
Files: logical records

- In contrast to the division into **physical records**, a file often has natural divisions determined by the information represented.

- For example, a file containing information regarding a company's employees would consist of multiple units, each consisting of the information about one employee.

- Or, a file containing a text document would consist of paragraphs or pages. These naturally occurring blocks of data are called **logical records**.

Figure 1.12 **Logical records versus physical records on a disk**



Representing Text

- **Each character (letter, punctuation, etc.) is assigned a unique bit pattern.**
  - ASCII: Uses patterns of 7-bits to represent most symbols used in written English text
  - ISO developed a number of 8 bit extensions to ASCII, each designed to accommodate a major language group
  - Unicode: Uses patterns of 16-bits to represent the major symbols used in languages world wide

Figure 1.13 **The message "Hello." in ASCII**

01001000	01100101	01101100	01101100	01101111	00101110
H	**e**	**l**	**l**	**o**	**.**

ASCII – some formatting controls

Binary	Oct	Dec	Hex	Abbr	PR[t 1]	CS[t 2]	CEC[t 3]	Description
000 0000	000	0	00	NUL	NUL	^@	\0	Null character
000 0001	001	1	01	SOH	SOH	^A		Start of Header
000 0010	002	2	02	STX	STX	^B		Start of Text
000 0011	003	3	03	ETX	ETX	^C		End of Text
000 0100	004	4	04	EOT	EOT	^D		End of Transmission
000 0101	005	5	05	ENQ	ENQ	^E		Enquiry
000 0110	006	6	06	ACK	ACK	^F		Acknowledgment
000 0111	007	7	07	BEL	BEL	^G	\a	Bell
000 1000	010	8	08	BS	BS	^H	\b	Backspace[t 4][t 5]
000 1001	011	9	09	HT	HT	^I	\t	Horizontal Tab
000 1010	012	10	0A	LF	LF	^J	\n	Line feed
000 1011	013	11	0B	VT	VT	^K	\v	Vertical Tab
000 1100	014	12	0C	FF	FF	^L	\f	Form feed
000 1101	015	13	0D	CR	CR	^M	\r	Carriage return[t 6]
000 1110	016	14	0E	SO	SO	^N		Shift Out
000 1111	017	15	0F	SI	SI	^O		Shift In
001 0000	020	16	10	DLE	DLE	^P		Data Link Escape
001 0001	021	17	11	DC1	DC1	^Q		Device Control 1 (oft. XON)

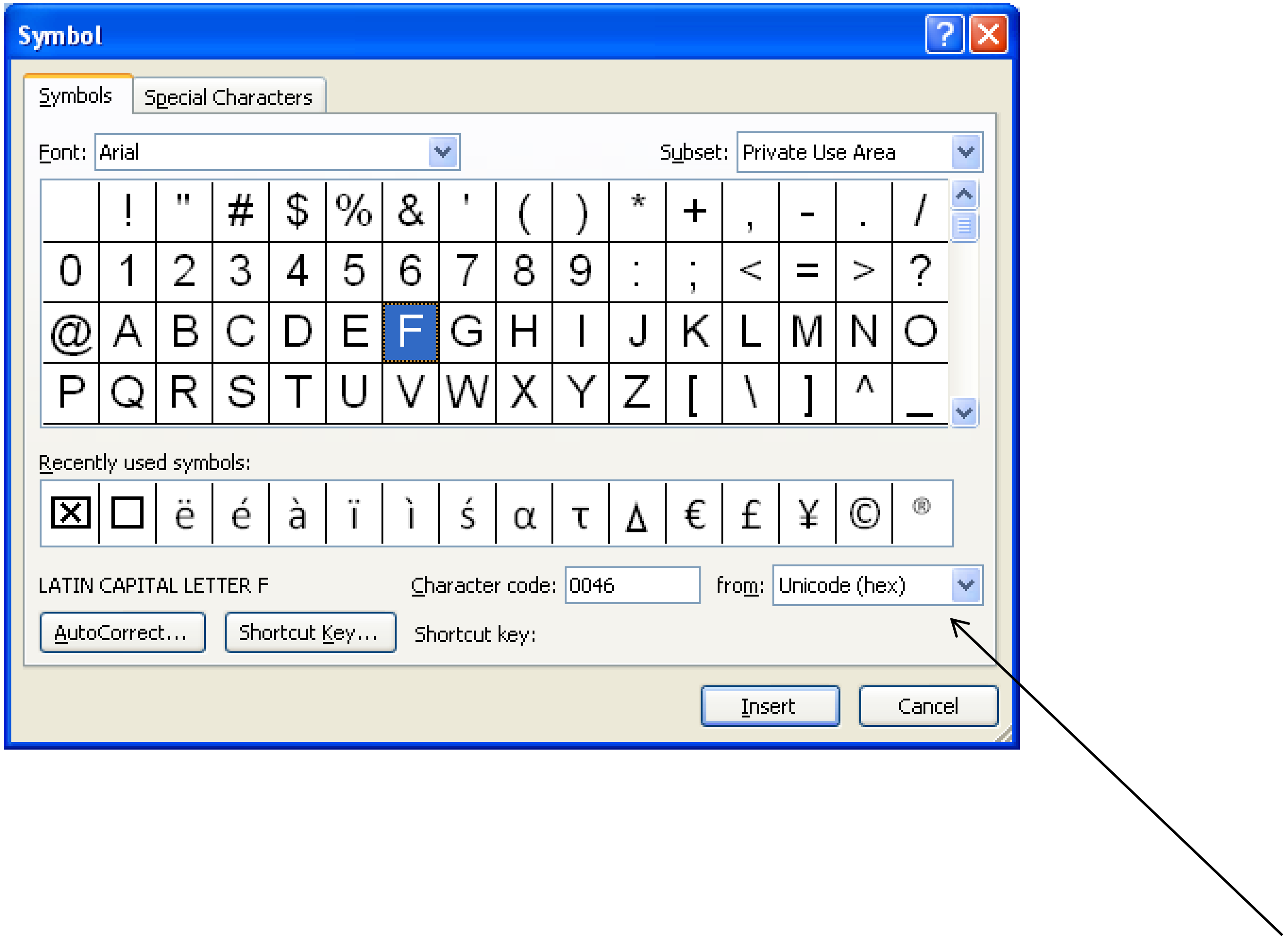
Unicode

- Although **ASCII** has been the **dominant code for many years**, other more extensive codes, capable of representing documents in a variety of languages, are now competing for popularity.

- One of these, **Unicode**, was developed through the cooperation of several of the leading manufacturers of hardware and software and is rapidly gaining support in the computing community.

- This code uses a unique pattern of 16 bits to represent each symbol.
  - As a result, Unicode consists of **65,536 different bit patterns-** enough to allow text written in such languages as Chinese, Japanese, and Hebrew to be represented.

Unicode Characters in Word



Representing Numeric Values

- Binary notation: Uses bits to represent a number in base two
- Limitations of computer representations of numeric values
  - Overflow: occurs when a value is too big to be represented
  - Truncation: occurs when a value cannot be represented accurately

Text files

- **A file consisting of a long sequence of symbols encoded using ASCII or Unicode is often called a text file.**

- It is important to distinguish between **simple text files** that are manipulated by utility programs called text editors (or often simply editors) and the **more elaborate files** produced by word processors.

- Both consist of textual material. However, a text file contains only a **character-by-character encoding** of the text, whereas a file produced by a word processor contains numerous proprietary codes representing changes in fonts, alignment information, etc.

- Moreover, word processors may even use **proprietary codes** rather than a standard such as ASCII or Unicode for representing the text itself.

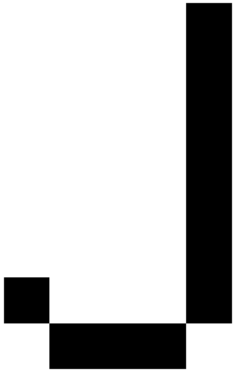
Representing Images

- Bit map techniques
  - Pixel: short for “picture element”
  - RGB
  - Luminance and chrominance
- Vector techniques
  - Scalable
  - TrueType and PostScript

Representing Images

- Popular techniques for representing images can be classified into two categories:
  - **bit map** techniques
  - **vector** techniques.
- In the case of bit map techniques, an image is represented as a **collection of dots**, each of which is called a pixel, short for "picture element."
  - A black and white image is then encoded as a **long string of bits representing the rows of pixels in the image**, where each bit is either 1 or 0 depending on whether the corresponding pixel is black or white. This is the approach used by most facsimile machines.
- **Vector** techniques provide a means of representing images as a **collection of lines and curves.**

Bitmap

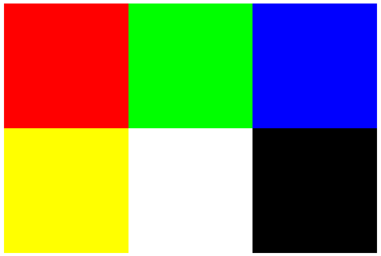


```
P1
# This is an example bitmap of the letter "J"
6 10
0 0 0 0 1 0
0 0 0 0 1 0
0 0 0 0 1 0
0 0 0 0 1 0
0 0 0 0 1 0
0 0 0 0 1 0
1 0 0 0 1 0
0 1 1 1 0 0
0 0 0 0 0 0
0 0 0 0 0 0
```

RGB

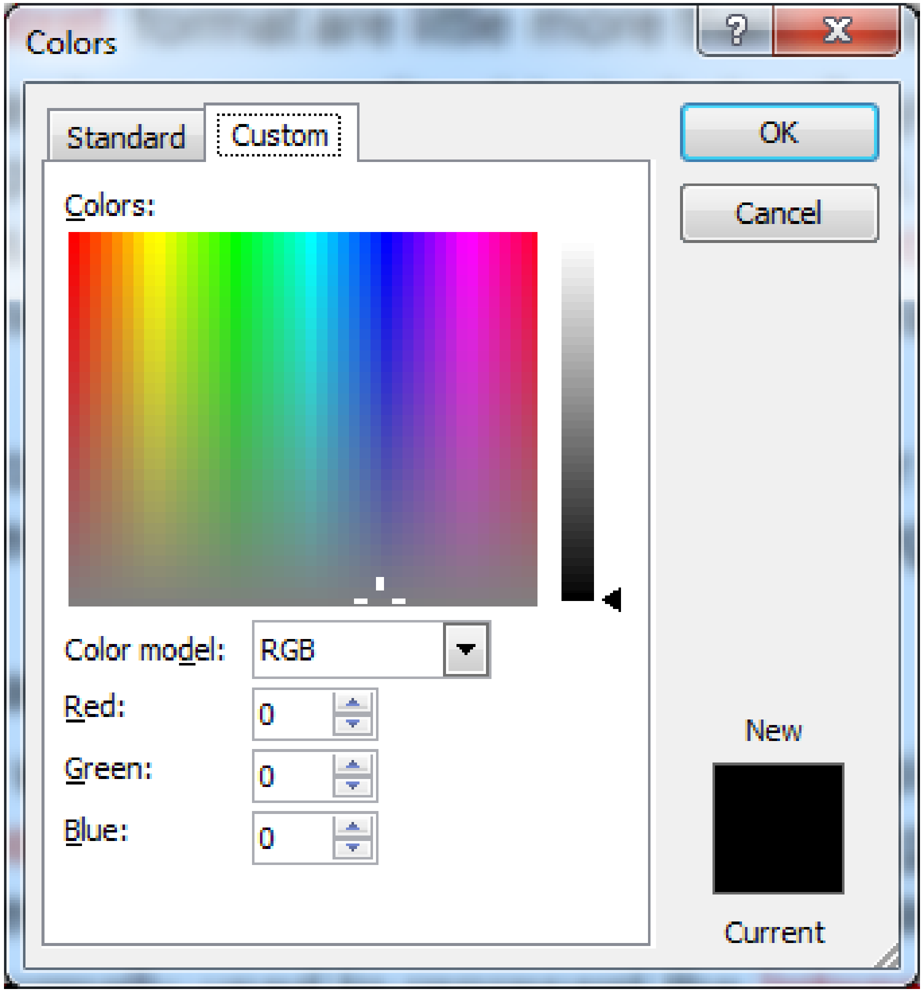
- The term bit map originated from the fact that the bits representing an image in a **one-bit-per-pixel** format are little more than a map of the image.
- Today the term has been generalized to include all systems in which images are encoded in a pixel-by-pixel manner. For example, in the case of black and white photographs, **each pixel is represented by a collection of bits** (usually eight), which allows a variety of shades of grayness to be represented.
- This bit map approach is generalized further for color images, where each pixel is represented by a combination of bits indicating the appearance of that pixel.
- Two approaches are common:
  - In one, which we will call **RGB encoding**, each pixel is represented as **three color components** - a red component, a green component, and a blue component-corresponding to the three primary colors of light.
  - One byte is normally used to represent the **intensity of each color** component. In turn, three bytes of storage are required to represent a **single pixel** in the original image.

RGB



```
P3
# The P3 means colors are in ASCII, then 3 columns and 2 rows, then 255 for max
color, then RGB triplets
3 2
255
255 0 0
0 255 0
0 0 255
255 255 0
255 255 255
0 0 0
```

RGB in Microsoft Office



Vector graphics

- One disadvantage of bit map techniques is that an image **cannot be rescaled** easily to any arbitrary size.

- Essentially, the only way to enlarge the image is to make the pixels bigger, which leads to a **grainy appearance**. (This is the technique called "**digital zoom**" used in digital cameras as opposed to "**optical zoom**" that is obtained by adjusting the camera lens.)

- Vector techniques provide a means of overcoming this scaling problem. Using this approach, an image is represented as a collection of lines and curves.
  - Such a description leaves the details of **how the lines and curves are drawn to the device** that ultimately produces the image rather than insisting that the device reproduce a particular pixel pattern.

Vector representation

- The various fonts available via today's word processing systems are usually encoded using **vector techniques** in order to provide flexibility in character size, resulting in **scalable fonts**.
- For example, **TrueType** (developed by Microsoft and Apple Computer) is a system for describing how symbols in text are to be drawn.
- Likewise, **PostScript** (developed by Adobe Systems) provides a means of describing characters as well as more general pictorial data.
- Vector representation techniques are also popular in computer-aided design (**CAD**) systems in which drawings of three-dimensional objects are displayed and manipulated on computer screens.

The Binary System

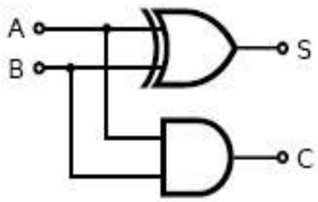
The traditional decimal system is based on powers of ten.

The Binary system is based on powers of two.

Figure 1.15 The base ten and binary systems

Binary ☺

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

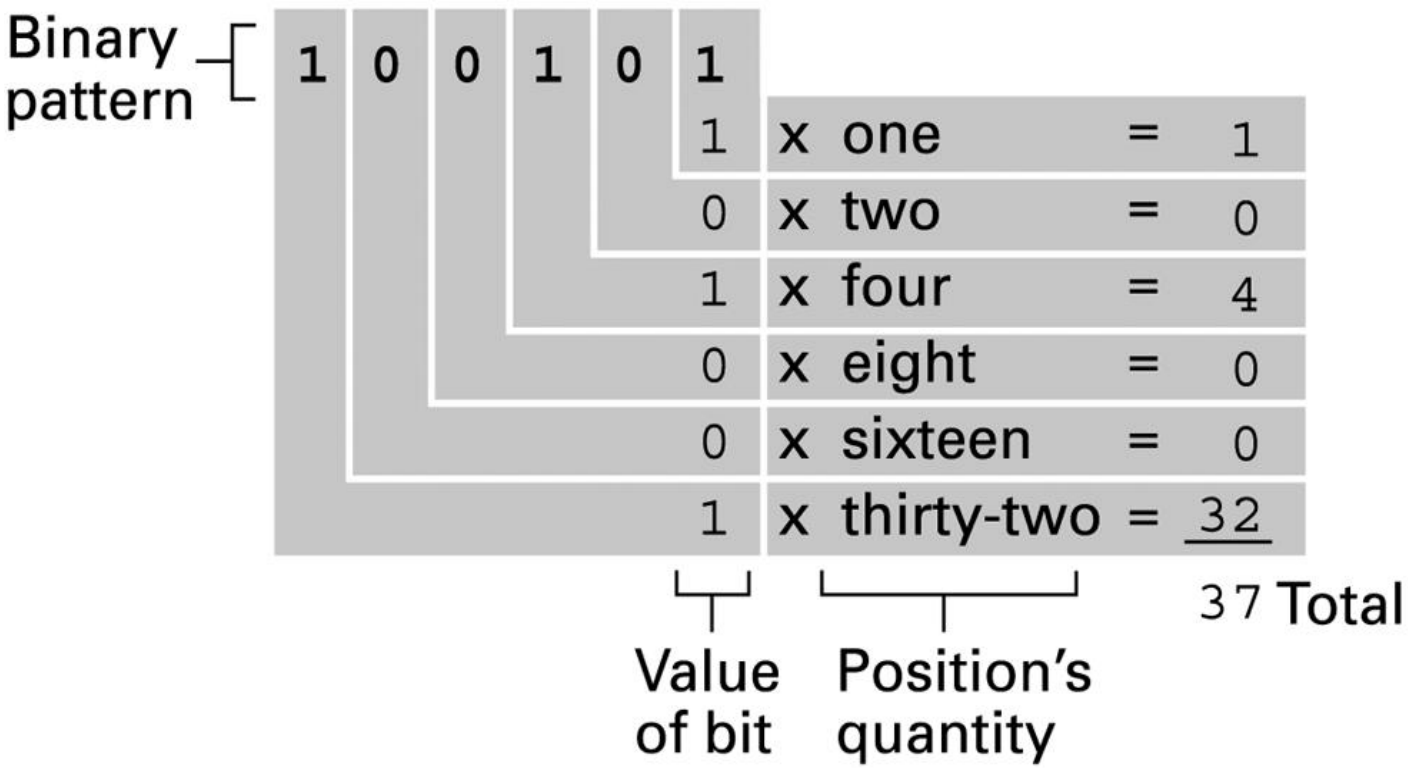


The circuit diagram for a binary half adder, which adds two bits together, producing sum and carry bits.

$0 + 0 \rightarrow 0$
$0 + 1 \rightarrow 1$
$1 + 0 \rightarrow 1$
$1 + 1 \rightarrow 0$, carry 1 (since $1 + 1 = 0 + 1 \times 10$ in binary)

There are 10 kinds of people in the world, those that understand binary and those that don't.

Figure 1.16 **Decoding the binary representation 100101**



Binary system mapping

Decimal	Hexadecimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111

Decimal	Hexadecimal	Binary
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

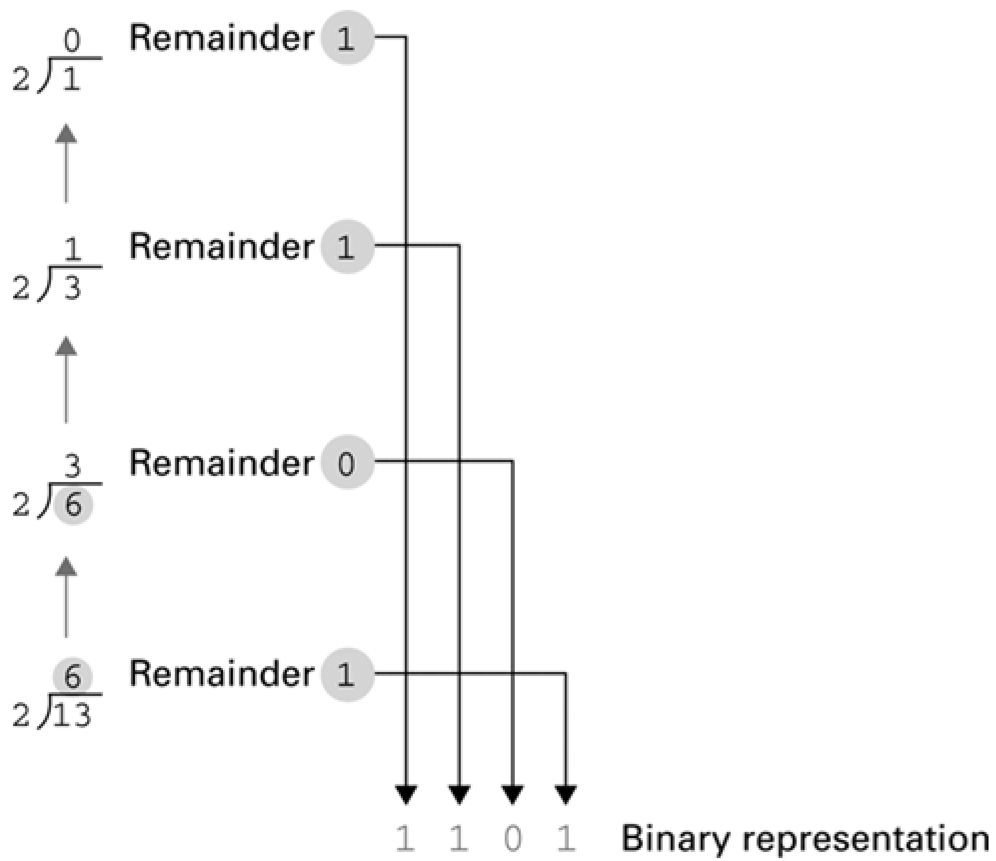
Figure 1.17 **An algorithm for finding the binary representation of a positive integer**

Step 1. Divide the value by two and record the remainder.

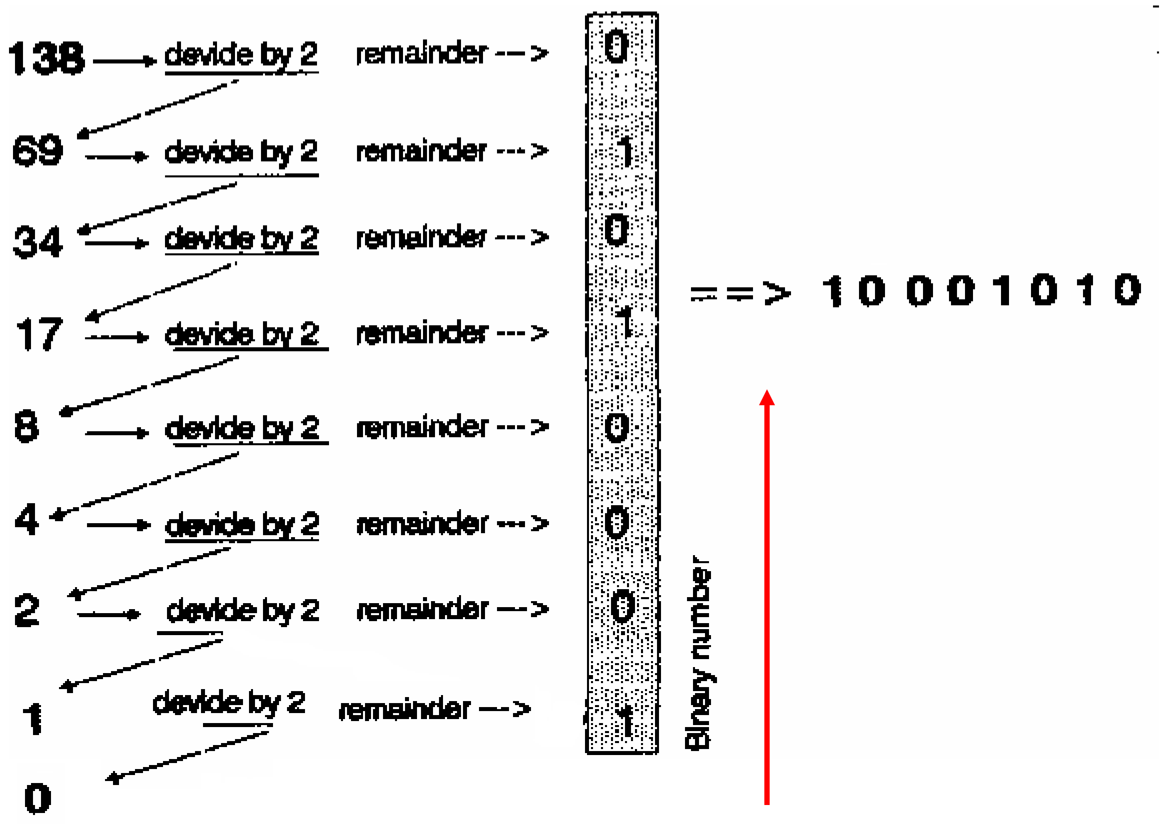
Step 2. As long as the quotient obtained is not zero, continue to divide the newest quotient by two and record the remainder.

Step 3. Now that a quotient of zero has been obtained, the binary representation of the original value consists of the remainders listed from right to left in the order they were recorded.

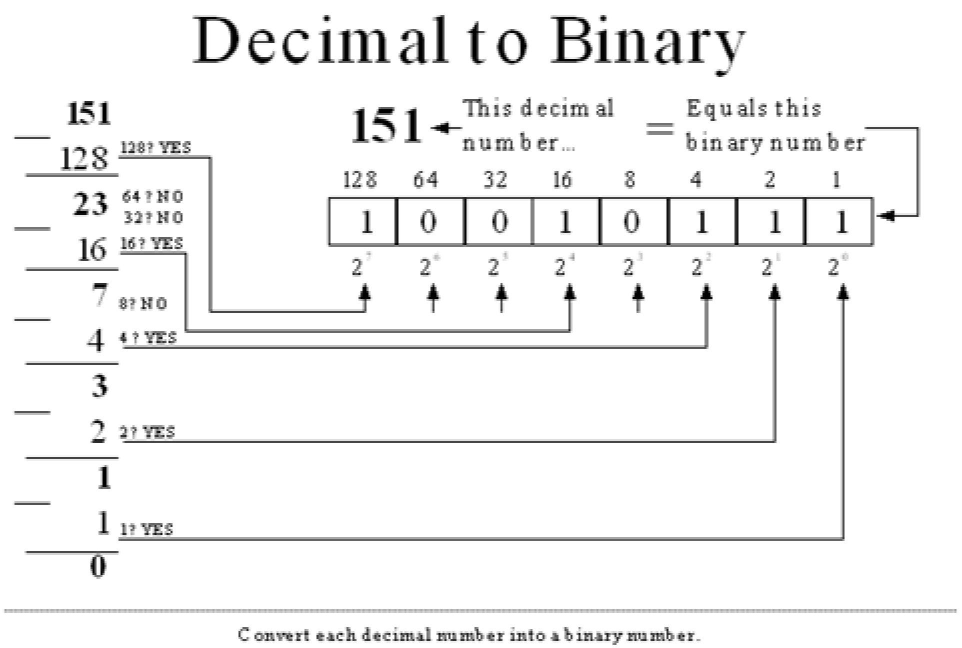
Figure 1.18 **Applying the algorithm in Figure 1.15 to obtain the binary representation of thirteen**



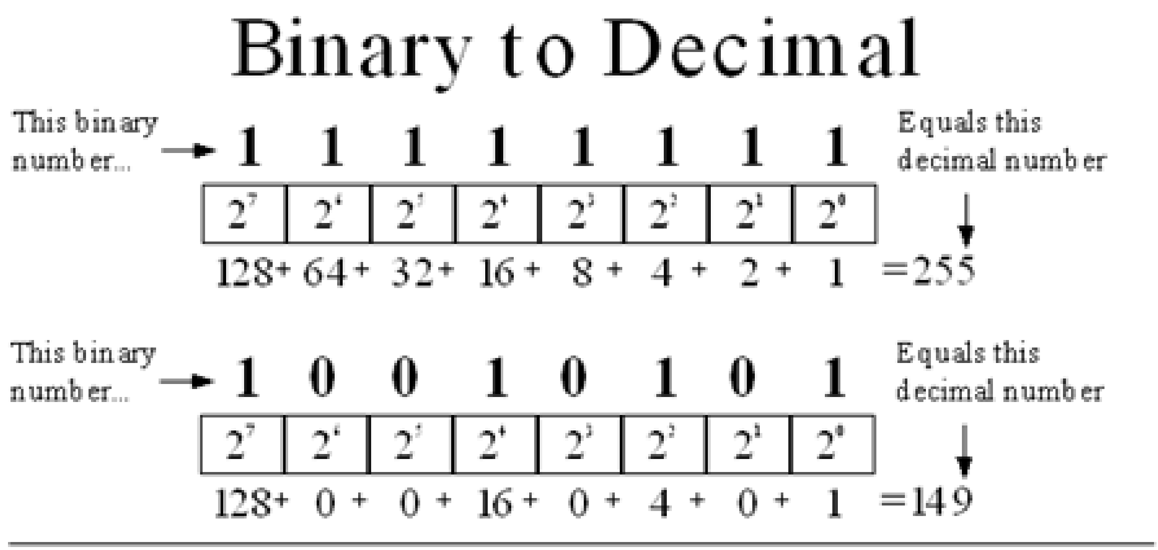
Top to bottom of page:
Decimal to binary: $138_{10}=10001010_2$



Converting 151 from decimal to binary



Binary to decimal



Convert each binary number into a decimal number.

Fixed-point representation

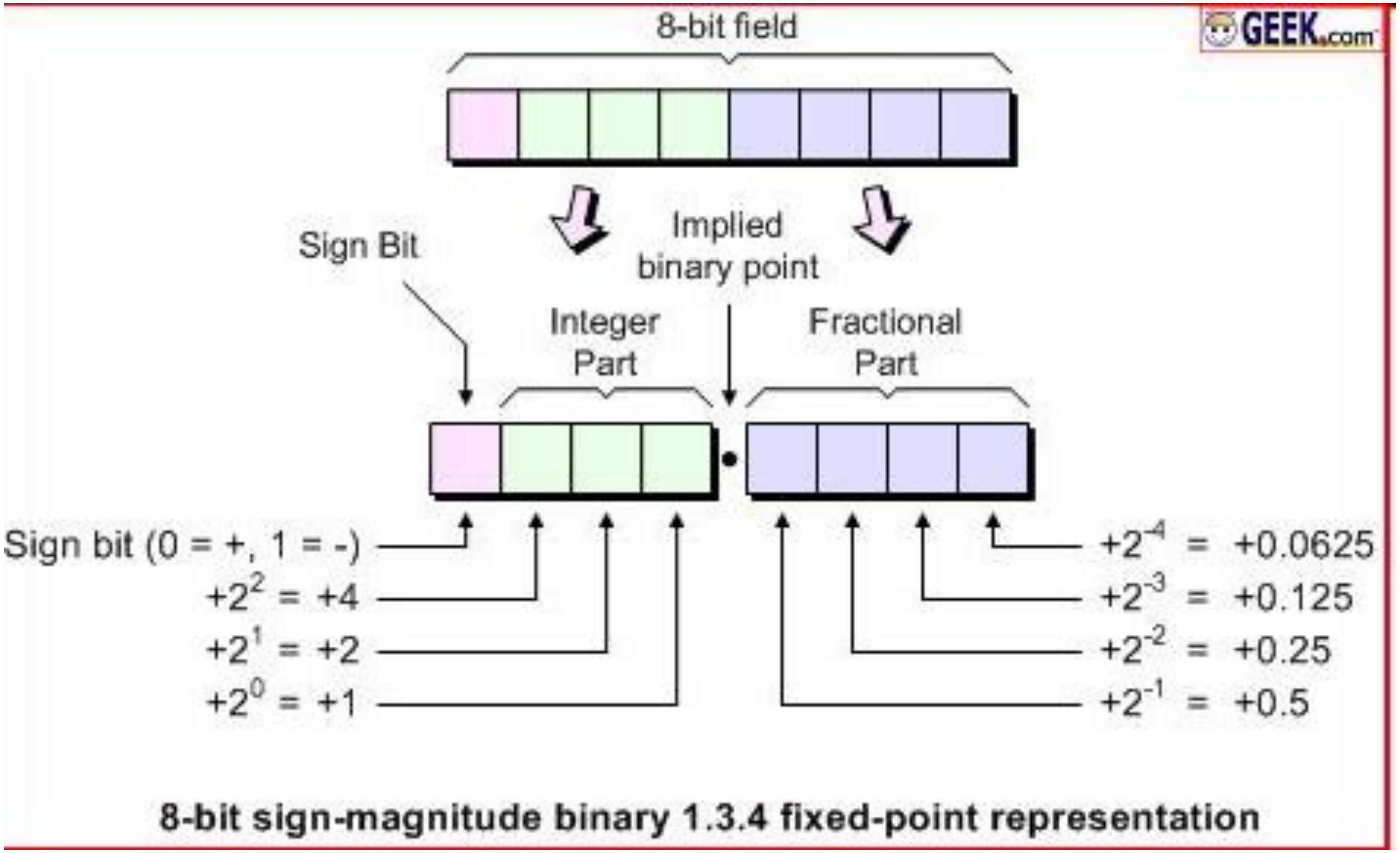


Figure 1.20 **Decoding the binary representation 101.101**

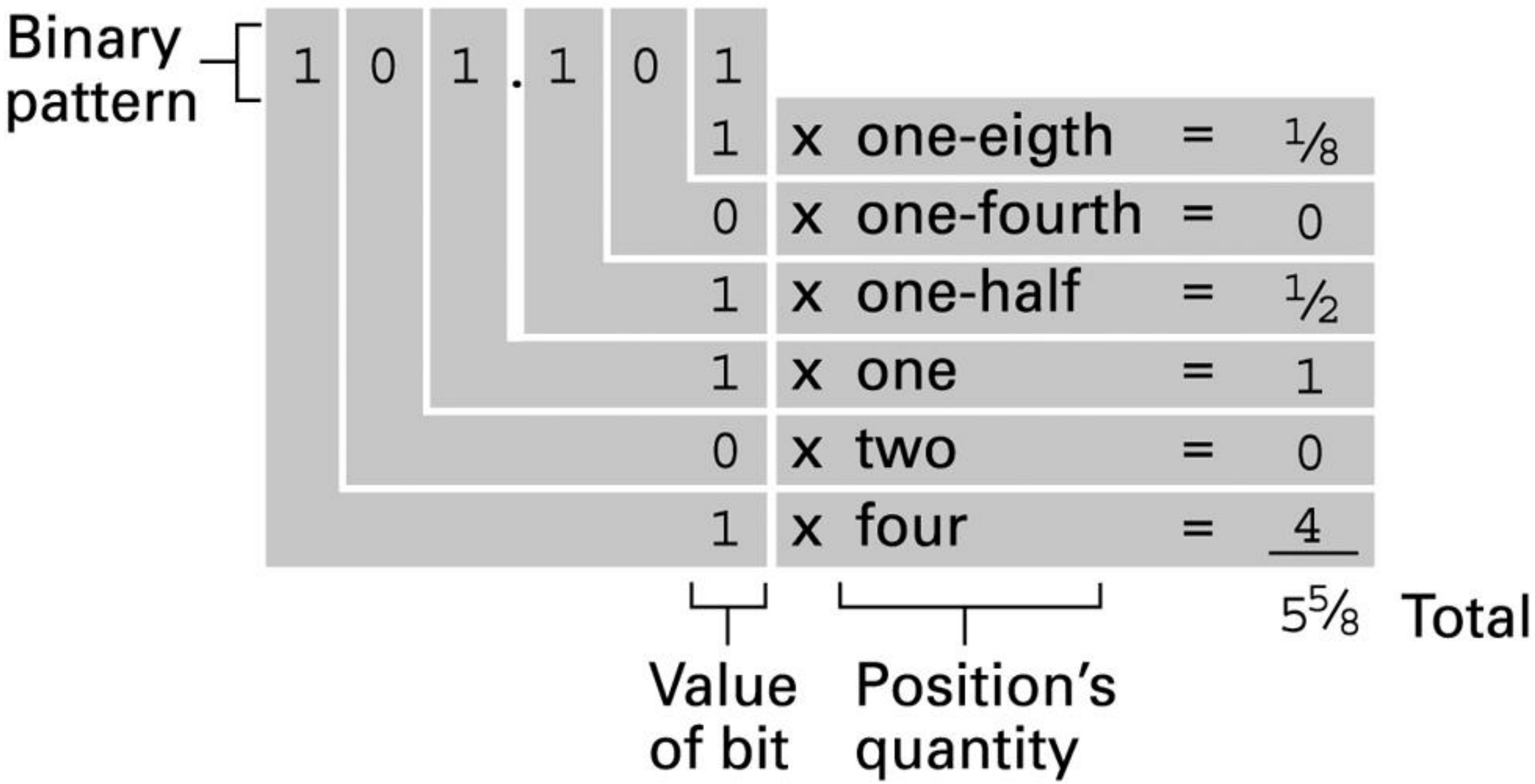


Figure 1.19 The binary addition facts

$$\begin{array}{r} 0 \\ +\,0 \\ \hline 0 \end{array} \qquad \begin{array}{r} 1 \\ +\,0 \\ \hline 1 \end{array} \qquad \begin{array}{r} 0 \\ +\,1 \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ +\,1 \\ \hline 10 \end{array}$$

End of class

- Readings
  - Book: Chapter 1