Application of Information & Communication Technologies

Lecture-3

Recap of Lecture 2

- Parts of Computer System
  - Hardware
  - Software
  - User
- Information Technology
  - Data
  - Characteristics of IT
  - Functions of IT

Overview of Lecture 3

- System Unit
  - Digital Data & Program Representation
  - Bits & Bytes
  - Numbering Systems
    - Decimal & Binary
  - Coding System

Why Representation?

- Computers cannot understand English, Urdu, or Chinese.
- Computers understand only 0 and 1.
- We need to translate our language into binary (0s & 1s).

Digital Computers

- Digital computers = use binary (two states).
- 0 = OFF (no electricity, open circuit).
- 1 = ON (electricity flows, closed circuit).
- All data → text, numbers, images, videos → must be in binary.

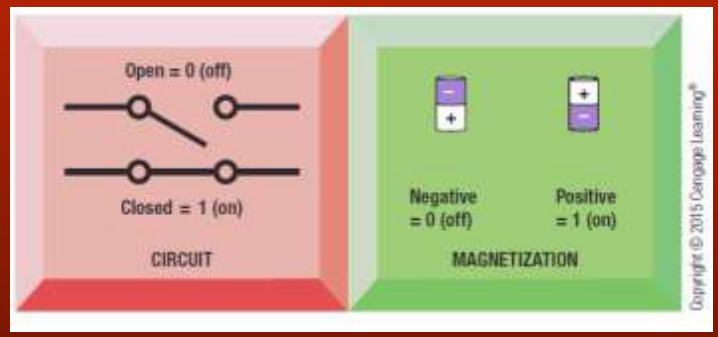


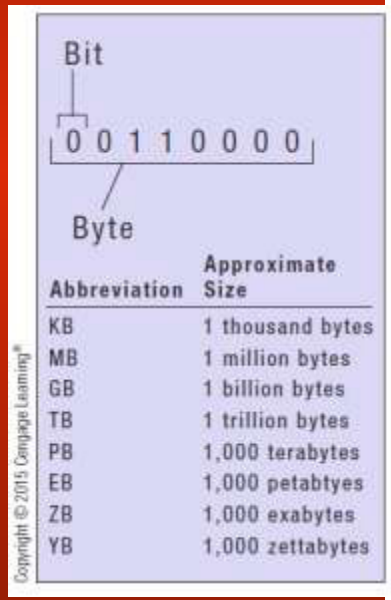
FIGURE 2-1
Ways of representing 0 and 1. Binary computers recognize only two states—off and on—usually represented by 0 and 1.

Digital Data Representation

- Bit = Smallest unit of data (0 or 1).
- Byte = 8 bits together.
- Examples:
  - Letter "A" → 01000001
  - Number "5" → 00000101

Bits & Bytes

- 1 Byte = 8 Bits
- Storage sizes:
  - KB (Kilobyte) = 1,000 bytes
  - MB (Megabyte) = 1 million bytes
  - GB (Gigabyte) = 1 billion bytes
  - TB (Terabyte) = 1 trillion bytes
- Example:
  - A song file = 5 MB
  - A movie = 1 GB



Abbreviation	Approximate Size
KB	1 thousand bytes
MB	1 million bytes
GB	1 billion bytes
TB	1 trillion bytes
PB	1,000 terabytes
EB	1,000 petabtyes
ZB	1,000 exabytes
YB	1,000 zettabytes

FIGURE 2-2
Bits and bytes. Document size, storage capacity, and memory capacity are all measured in bytes.

Human vs Computer Language

- Humans speak: English, Urdu, Chinese…
- Computers speak: Binary (0s & 1s).
- Example:
  - Word "HI" = 01001000 01001001 in binary.
- How does a computer understand human language?

Class Activity

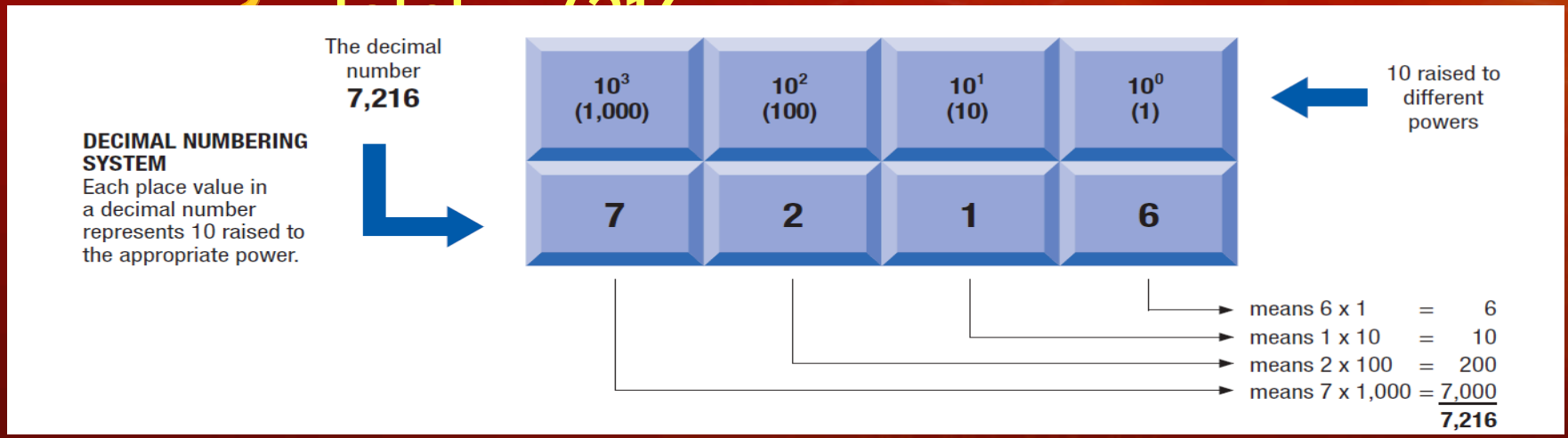
- Write down the following:
  - How many bits in 1 byte?
  - Convert: 8 bits = ? Bytes
  - Your mobile storage = ? GB (how many MB is that?)

Numbering Systems

- Decimal system → Base 10 (digits 0–9)
- Binary system → Base 2 (digits 0 and 1)
- Decimal is human language, Binary is computer language

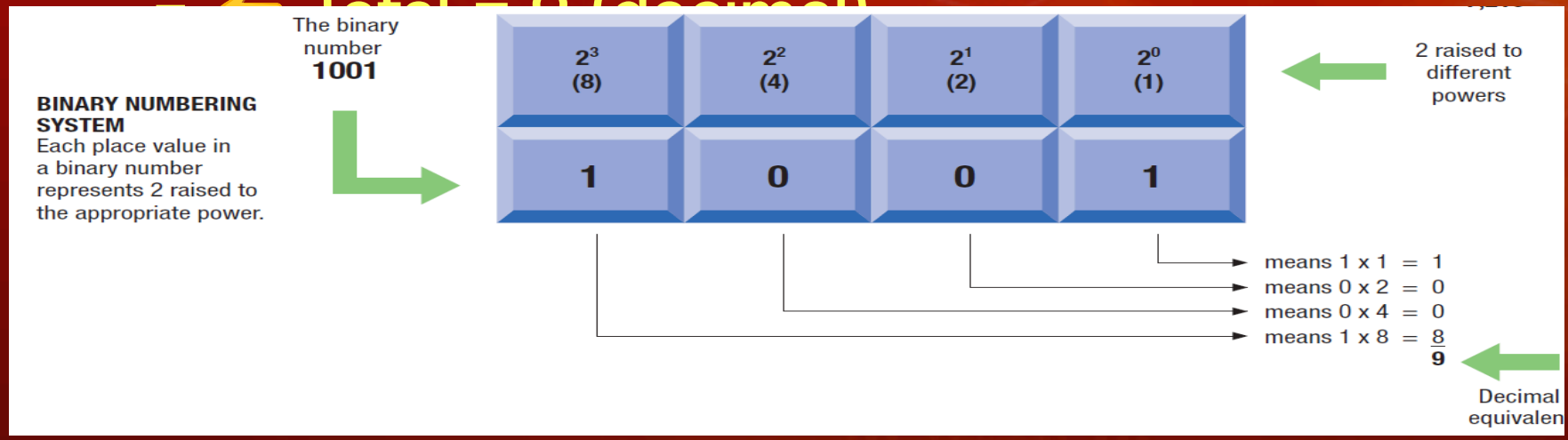
Decimal Number Example

- Decimal number: 7,216
- Place values: 10^3, 10^2, 10^1, 10^0
  - $7 \times 1000 = 7000$
  - $2 \times 100 = 200$
  - $1 \times 10 = 10$
  - $6 \times 1 = 6$
  - Total = 7216



Binary Number Example

- Binary number: 1001
  - Place values: 2^3, 2^2, 2^1, 2^0
  - 1 × 8 = 8
  - 0 × 4 = 0
  - 0 × 2 = 0
  - 1 × 1 = 1
  - [illegible]



Key Point

- Decimal: every position = power of 10
- Binary: every position = power of 2
- Same method, different base

Class Activity

- 👉 Convert the following:
  - Decimal 25 → Binary
  - Binary 1110 → Decimal
  - Decimal 45 → Binary
  - Write down the following:

Coding Systems (Text Data)

- Numbers use Binary.
- Text uses special coding systems:
  - ASCII → Old standard for personal computers.
  - EBCDIC → IBM mainframes.
  - Unicode → Modern, supports all world languages + symbols.

Coding Systems (Text Data)

- ASCII & EBCDIC = 1 Byte per character (256 symbols).
- Unicode = up to 4 Bytes → more than 1 million characters.

CHARACTER	ASCII	EBCDIC
0	00110000	11110000
1	00110001	11110001
2	00110010	11110010
3	00110011	11110011
4	00110100	11110100
5	00110101	11110101
A	01000001	11000001
B	01000010	11000010
C	01000011	11000011
D	01000100	11000100
E	01000101	11000101
F	01000110	11000110
+	00101011	01001110
!	00100001	01011010
#	00100011	01111011

FIGURE 2-4
Examples from the ASCII and EBCDIC codes. These common fixed-length binary codes represent all characters as unique strings of 8 bits.

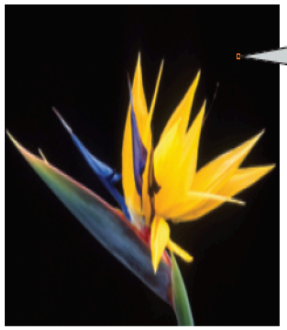
FIGURE 2-5
Unicode. Many characters, such as these, can be represented by Unicode but not by ASCII or EBCDIC.

Why Unicode?

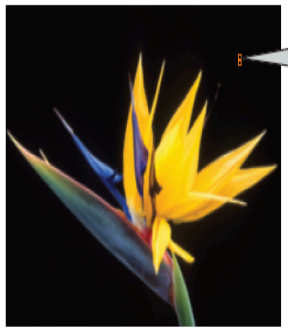
- ASCII → Only English letters, digits, symbols.
- Unicode → All languages (English, Arabic, Chinese, Russian, Greek, etc.).
- Supports emojis 😊 and technical symbols.
- Used by Google, Windows, Mac, Java, Python.

Coding Systems (Other Data)

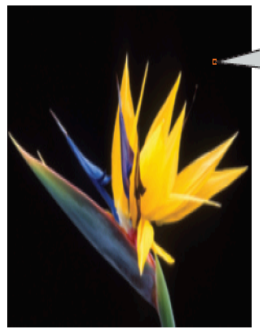
- Graphics (Images):
  - Stored as pixels (dots).
  - Each pixel = binary color value.
  - More bits per pixel → more colors. (1 bit = 2 colors, 24 bits = 16 million colors).



16-COLOR IMAGE
The color of each pixel is represented using one-half byte (4 bits).



256-COLOR IMAGE
The color of each pixel is represented using one byte (8 bits).



PHOTOGRAPHIC-QUALITY (TRUE COLOR) IMAGE (16.8 million colors)
The color of each pixel is represented using three bytes (24 bits).

Coding Systems (Other Data)

- Audio (Sound):
  - Sound → converted into samples per second.
  - Example: Audio CD = 44,100 samples/sec.
  - Compressed → MP3 (smaller size).

Coding Systems (Other Data)

- Video:
  - Frames of images + sound.
  - 24–60 frames per second.
  - Compressed to save space (DVD, YouTube).

Machine Language

- Programs also stored in 0s and 1s.
- Machine Language = CPU's native language.
- Example: 01011000011100000000000100000010
- Early computers → programmed in machine code.
- Today → We use high-level languages (C, Java, Python), then translated into machine language.

Summary

- System Unit
  - Digital Data & Program Representation
  - Bits & Bytes
  - Numbering Systems
    - Decimal & Binary
  - Coding System

Suggested Reading

- Section 01, Ch-02, The System Unit: Processing and Memory , “Understanding Computers: Today and Tomorrow, Comprehensive”, 15th Edition by Deborah Morley & Charles S. Parker