Information Technology
Inside and Outside
- David Cyganski & John A. Orr

II. Fundamentals of Binary Representation

4. The Need and Basis for Data Protocols

Hoon-Jae Lee
hjlee@dongseo.ac.kr
http://cg.dongseo.ac.kr/~hjlee

4. The Need and Basis for Data Protocols

- How to organize the large number of bits needed to represent complex information?
  - The answer is via ***protocols***
- Objectives:
  - **protocols** are an integral part of our everyday life, in the human as well as digital world;
  - about examples of simple but important protocols, such as the positions of delivery and return addresses on envelopes;
  - about actual protocols used for information storage on magnetic tape, CD-ROMs, and so on;
  - the ASCII protocol for transmission of text data; and
  - protocols used by word processors and the World Wide Web.

4.2 Using Protocols to Organize Information

- ***Protocols*** are agreed upon sets of rules that provide order to different systems and situations.
- **Data protocols** bring order to information systems, allowing them to share information in a useful way.
- **Dewey decimal system** (named after Melvil Dewey)is often used; in this system, each library book is given an *address* that is organized according to information about the book's content class, author, and year of publication.
  - For example, the **400 class** is assigned to all books **about language**, and the **420 subclass** to books **about English**.
- **ISBN** (International Standard Book Number) provides information in its 10-digit code regarding the national language, geographic origin,publisher, title, edition and volume number of the volume.

4.2 Using Protocols to Organize Information (2)

- A **postal letter** provides an excellent introduction to several important concepts in information protocols:
  - the content is enclosed (in an envelope) so that it is clear where the message starts and ends;
  - this envelope contains the address of the intended recipient, a return address that identifies the sender, and a postmark showing the date it was handled and the post office that handled it.
- The separation of information in the **electronic message** is much more like the separation of the addresses on the face of an envelope:
  - the recipient's address in the front and center of the envelope,
  - the return address in the upper left corner or on the back.
  - A fundamental aspect of decoding any binary string is determining where to begin, that is, identifying the first bit in a protocol.

4.2 Using Protocols to Organize Information (3)

- **Figure 4.1:**A stream of binary data with no apparent protocol

4.2 Using Protocols to Organize Information (4)

4.2.1 How Numbers Can Be Packaged

- First microprocessor (the brains of a smart machine or computer) manipulated data in **4-bit** groups, often called ``**nibbles**."
- The first commercial personal computers (PCs) were **8-bit** machines; that is, they manipulated a word size of **one byte** with each operation.
- Later, **16-bit** computers, then **32-bit** computers, and more recently, **64-bit** machines were introduced.
- Thus, the first PCs stored and processed **1 byte** at a time, and we are now using machines that handle **4 or 8 bytes** at a time.

4.2 Using Protocols to Organize Information (5)

4.2.1 How Numbers Can Be Packaged(2)

- What would the following pair of bytes found in computer memory represent: 00000001 1000000?
  - one 16-bit binary word--giving the number 0000000110000000, which in decimal is 256 + 128 = 384
  - 1000000000000001, which in decimal is32768 + 1 = 32769
- The byte in the lower address location is taken to be the high-order byte, is called the ***big-endian*** protocol,while the other is called the ***little-endian*** protocol.
  - The common **Pentium processor-based PC** is an example of a **big-endian** computer, <MSB first>
  - while the line of computers sold by **Silicon Graphics** are examples of **little-endian** computers. <LSB first>

4.3 Saving Information : Tapes, Discs, CDs

- ***Data Storage***
  - **RAM**(random access memory) : linear string of bytes, numbered consecutively fro aero to the last storage locations

4.3.1 Magnetic Tape

- **Magnetic tape**: data is written on the tape in a linear fashion, and in that sense tape seems similar to RAM.
  - with **RAM** it is possible to retrieve data directly from a given memory address,
  - while on **tape** there is no direct way to access a desired address.
    - ✓ the tape drive must read the tape sequentially
    - ✓ much slower process
  - the data on the tape is divided into sections called ***blocks***, each of which contains hundreds or thousands of bytes
  - first the desired block is located, then the data is retrieved.
  - **tape** is called a ***sequential access*** medium, compared to the ***direct access*** or ***random access*** provided by **RAM**.

4.3 Saving Information : Tapes, Discs, CDs (2)

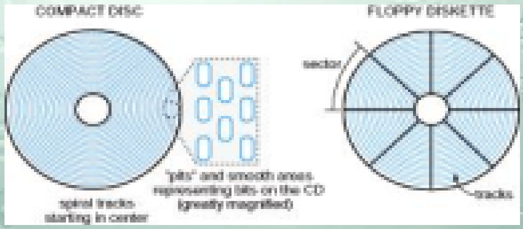
4.3.2 Rotating Memory Devices

- Rotating disk : sequential access medium
  - ***Floppy disk*** , larger-capacity ***hard disk***
  - Disk surface is organized into many circular ***tracks***
  - These tracks are concentric circles, arranged from the outside edge toward the center of the disk
  - The ubiquitous **3 1/2 inch, 1.44 MB floppy disk**, for example, has **80** such **tracks** defined on each of its **two sides**, for a total of **160 addressable tracks**.
  - A distinctive pattern of bits identifies the start of the track, which is then subdivided into ***sectors***
  - The **3 1/2 inch floppy** divides each **track** into **18** such **sectors**; each of these sectors packages **512 bytes**. The total number of bytes contained on the disk is therefore **160 x 18 x 512 = 1,474,560 bytes = 1.44 MB**
    - ✓ 1 Kbyte = 1024 bytes,or 2^{10} bytes;
    - ✓ 1 Mbyte = 1024 x 1024 = 1,048,576 bytes, or 2^{20} bytes

4.3 Saving Information : Tapes, Discs, CDs (3)

4.3.2 Rotating Memory Devices

- **Figure 4.2:** A CD-ROM with spiral tracking, and a conventional floppy disk with cylindrical tracking.



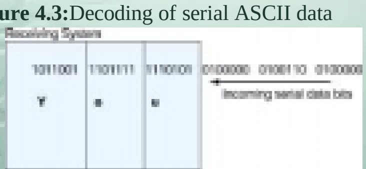
4.3 Saving Information : Tapes, Discs, CDs (4)

4.3.2 Rotating Memory Devices

- The **CD-ROM** retains some of the characteristics of both the magnetic disk and tape.
  - The data storage and reading are **sequential**, but they are based on an optical interaction rather than a magnetic one.
  - Physically, the CD-ROM looks similar to a magnetic disk, but in terms of data storage organization it is more similar to the tape. This is because data is written in a single track that spirals its way from the inside edge of the disk to the outside.
  - However, through use of the CD-ROM directory and file allocation table, **the read head** (which contains a laser and photo sensitive sensor rather than a magnetic pickup) **may be quickly positioned anywhere radially on the disk**.
  - Thus it is not necessary to read or even roll past all the previous data to reach the desired data, as it is on a tape.

4.4 Protocols for Sending Data

- Some **message** to be transmitted;
  - It may be destined to travel on the **memory bus** inside your computer, on the **cable** to your printer, on a **local area network** in your office building,or across a **transcontinental communications link**.
- Each **ASCII code** can be concatenated to form a readable **stream of bits**; the result is a long string of ones and zeroes, which must be broken up at the receiver into individual seven (or eight) bit numbers that can then be decoded back to characters
- **Figure 4.3:** Decoding of serial ASCII data



4.4 Protocols for Sending Data(2)

- ***Bit-oriented transmission*** methods is based on an approach that was popularized by a data transmission protocol known as **HDLC** (High-level Data Link Control), an international standard that was defined by the ISO (International standards Organization)
  - each group of data that is to be separable identifiable in a bit stream is called a ***frame***
  - the start-of-frame pattern or ***flag byte*** = "**01111110"**
  - ***bit stuffing*** or ***zero bit insertion***: the software or hardware handling the insertion of information into the transmitted data stream looks for any **sequence of ones** that is **five** in length.

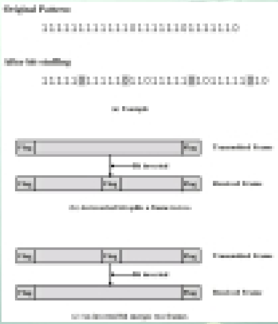
Cf. HDLC - Frame Structure Diagram

Cf. HDLC - Flag Fields

- Delimit frame at both ends
- 01111110
- May close one frame and open another
- Receiver hunts for flag sequence to synchronize
- Bit stuffing used to avoid confusion with data containing 01111110
  - 0 inserted after every sequence of five 1s
  - If receiver detects five 1s it checks next bit
  - If 0, it is deleted
  - If 1 and seventh bit is 0, accept as flag
  - If sixth and seventh bits 1, sender is indicating abort

Cf. HDLC - Bit Stuffing

- Example with possible errors



4. 5 Word Processor and Web Protocols

- An HTML document that renders the following sentence:
  - "A short text file."

```
<HTML>
        <BODY>
        <b>A short text file.</b>
        <\BODY>
<\HTML>
```

- A $\LaTeX$ document for the same sentence :

```
\documentstyle{article}
\begin{document}
{\bf A short text file.}
\end{document}
```