

Unit-1

Chap-1

Number System

What is meant by Analog System?

- 1) In telecommunications, an analog signal is one in which a base carrier's alternating current frequency is modified in some way, such as by amplifying the strength of the signal or varying the frequency, in order to add information to the signal. Broadcast and telephone transmission have conventionally used analog technology.

An analog signal can be represented as a series of sine waves. The term originated because the modulation of the carrier wave is analogous to the fluctuations of the human voice or other sound that is being transmitted.

- 2) Analog describes any fluctuating, evolving, or continually changing process.

- Analog refers to the circuits In which quantities like voltage and current vary continuously And also its graph is also like a waveform going in both positive and negative cycles, to measure the analog signal we can convert it to digital by using analog to digital converter, and also this conversion can be calculated by knowing the resolution of analog module whether its 12bit or 16bit according to that we can calculate the digital values in 0–100 scale

Digital System

- Digital describes electronic technology that generates, stores, and processes data in terms of two states: positive and non-positive. Positive is expressed or represented by the number 1 and non-positive by the number 0. Thus, data transmitted or stored with digital technology is expressed as a string of 0's and 1's. Each of these state digits is referred to as a bit (and a string of bits that a computer can address individually as a group is a byte).

- Prior to digital technology, electronic transmission was limited to analog technology, which conveys data as electronic signals of varying frequency or amplitude that are added to carrier waves of a given frequency. Broadcast and phone transmission has conventionally used analog technology.
- Digital technology is primarily used with new physical communications media, such as *satellite* and fiber optic transmission. A modem is used to convert the digital information in your computer to analog signals for your phone line and to convert analog phone signals to digital information for your computer.

Numbering System

- We are familiar with numbers, characters and symbols. But this type of data are not suitable for microprocessor, logic circuits, computers etc. For this reason data is converted into electronic pulses and each pulse is identified as code. Then this code is converted into numeric format by ASCII, where each number, character and symbol have numerical equivalent. E.g.: Character A has ASCII value 65.
- Using this equivalent, the data is interchanged into numeric format. For this numeric conversions we use number systems having a base number, which indicates the number of digits in that number system

Types of number system

I. Binary Number System

- Represents two types of digits 0's and 1's, so the base of number system is 2.
- Uses two types of electronic pulses, where absence of pulse shows 0 and presence of pulse shows 1.
- Each binary digit is called as bit.
- Left-most bit of a number is known as Most Significant Bit (MSB) and right-most bit is known as Least Significant Bit (LSB). Its same for all number system.
- A group of 4 bit is called as nibble and group of 8 bit is called as byte.
- Value of digit is determined by the position of digit in the number, where lowest value is for the right-most position and each successive position to the left has a higher place value. Its same for all number system.
- Examples: a) $(010101)_2$ b) $(1010.101)_2$

Octal Number System

- Represents 8 types of digits from 0 to 7, so the base of number system is 8.
- It takes exactly three binary digits to represent an octal digit.
- Binary 000 is same as octal digit 0, binary 001 is same as octal 1, and so on.
- Insufficient to convert values into bytes(8 bit), so not widely used in computers.
- Examples: a) $(03105)_8$ b) $(4237.23)_8$

Decimal Number System

- Represents 10 types of digits from 0 to 9, so the base of number system is 10.
- This is the most familiar number system with everyone.
- Examples a) $(582938)_{10}$ b) $(3797.902)_{10}$

Hexadecimal Number System

- Represents 16 types of digits from 0 to 9 and alphabets from A to F, so the base of number system is 16.
- Digits from 10 to 15 are represented as 10-A, 11-B, 12-C, 13-D, 14-E, 15-F.
- As numeric digits and alphabets are used to represent digits, this number system is also called as alphanumeric number system.
- More complex number system and widely used in computer system.
- Examples: a) $(AF38)_{16}$ b) $(CE7.5B)_{16}$

Conversion of Number System

- Computers and digital circuits processes information in the binary format. Each character is assigned 7 or 8 bit binary code to indicate its character which may be numeric, alphabet or special symbol.
- Example - Binary number 1000001 represents 65(decimal) in straight binary code, alphabet A in ASCII code and 41(decimal) in BCD code.

FLOATING POINT:

The number which has both an integer and a fractional part is called as floating point number. The floating point number can be used to represent large as well as small fraction or mixed number. Generally floating point represents a method of approximation to real number in such a way that they can support a wide range of values.

General form of a number can be represented as

$$N = \pm m \times b^{\pm e}$$

M = mantissa

B = base of number

E = exponent

In the whole expression the first number part is called as mantissa which is signed fixed point number and the second part is called as exponent that is either decimal or binary species position.

DISADVANTAGES:

Floating point number suffers from the loss of precision when it is represented with a fixed number of bits. Ex 32 or 64 bits.

Integer arithmetic is very much efficient than floating point arithmetic.

SIGNED NUMBERS:

Signed numbers or signed integers are the numbers which represent zero, positive and negative integers. There are three representations available for signed integers:

Sign magnitude representation.

1's complement representation.

2's complement representation.

SIGN MAGNITUDE REPRESENTATION:

The most significant bit in sign magnitude representation is the sign bit, having value 0 representing positive integer and 1 for negative integer. And the remaining $n-1$ bits represent the absolute or magnitude of the integers.

EXAMPLE: Let $n=8$, and its binary representation is 0 100 0001B

Sign bit=0 → positive

Absolute value = 100 0001B = 65D

So the integer is +65D

Floating point numbers

Digital signal processing can be separated into two categories - fixed point and floating point. These designations refer to the format used to store and manipulate numeric representations of data. Fixed-point DSPs are designed to represent and manipulate integers – positive and negative whole numbers – via a minimum of 16 bits, yielding up to 65,536 possible bit patterns (2^{16}). Floating-point DSPs represent and manipulate rational numbers via a minimum of 32 bits in a manner similar to scientific notation, where a number is represented with a mantissa and an exponent (e.g., $A \times 2^B$, where 'A' is the mantissa and 'B' is the exponent), yielding up to 4,294,967,296 possible bit patterns (2^{32}).

Fixed Point vs Floating Point

- The term 'fixed point' refers to the corresponding manner in which numbers are represented, with a fixed number of digits after, and sometimes before, the decimal point. With floating-point representation, the placement of the decimal point can 'float' relative to the significant digits of the number. For example, a fixed-point representation with a uniform decimal point placement convention can represent the numbers 123.45, 1234.56, 12345.67, etc, whereas a floating-point representation could in addition represent 1.234567, 123456.7, 0.00001234567, 1234567000000000, etc. As such, floating point can support a much wider range of values than fixed point, with the ability to represent very small numbers and very large numbers.

I. BCD (Binary-Coded Decimal)

code :

- Four-bit code that represents one of the ten decimal digits from 0 to 9.
- Example - $(37)_{10}$ is represented as 0011 0111 using BCD code, rather than $(100101)_2$ in straight binary code.
- Thus BCD code requires more bits than straight binary code.
- Still it is suitable for input and output operations in digital systems.
- Note: 1010, 1011, 1100, 1101, 1110, and 1111 are INVALID CODE in BCD code.

II. ASCII (American Standard Code Information Interchange) code :

- It is 7-bit or 8-bit alphanumeric code.
- 7-bit code is standard ASCII supports 127 characters.
- Standard ASCII series starts from 00h to 7Fh, where 00h-1Fh are used as control characters and 20h-7Fh as graphics symbols.
- 8-bit code is extended ASCII supports 256 symbols where special graphics and math's symbols are added.
- Extended ASCII series starts from 80h to FFh.

III. EBCDIC (Extended Binary Coded Decimal Interchange Code) code

- 8-bit alphanumeric code developed by IBM, supports 256 symbols.
- It was mainly used in IBM mainframe computers.

IV. Gray code

- Differs from leading and following number by a single bit.
- Gray code for 2 is 0011 and for 3 is 0010.
- No weights are assigned to the bit positions.
- Extensively used in shaft encoders.

V. Excess-3 code

- 4-bit code is obtained by adding binary 0011 to the natural BCD code of the digit.
- Example - decimal 2 is coded as $0010 + 0011 = 0101$ as Excess-3 code.
- It not weighted code.
- Its self-complimenting code, means 1's complement of the coded number yields 9's complement of the number itself.
- Used in digital system for performing subtraction operations.

Weighted Codes BCD

- The weighted codes are those that obey the position weighting principle, which states that the position of each number represent a specific weight. In these codes each decimal digit is represented by a group of four bits.
- In weighted codes, each digit is assigned a specific weight according to its position. For example, in 8421/BCD code, 1001 the weights of 1, 1, 0, 1 (from left to right) are 8, 4, 2 and 1 respectively.
- **There are millions of weighted code The most common one is 8421/BCD Code.**
- Examples: 8421, 2421, 84-2-1 are all weighted codes.

Non-weighted codes

- The non-weighted codes are not positionally weighted . In other words codes that are not assigned with any weight to each digit position.
- Examples: Excess-3(XS-3) and Gray Codes.

Examples: Excess-3 (XS-3) and Gray Codes

Decimal	BCD				Excess-3			
	8	4	2	1	BCD + 0011			
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0

EBCDIC

- **Extended Binary Coded Decimal Interchange Code (EBCDIC)** is an eight-bit character encoding used mainly on IBM mainframe and IBM midrange computer operating systems. EBCDIC descended from the code used with punched cards and the corresponding six bit binary-coded decimal code used with most of IBM's computer peripherals of the late 1950s and early 1960s.
- It is also supported on various non-IBM platforms such as Fujitsu-Siemens' BS2000/OSD, OS-IV, MSP, and MSP-EX, the SDS Sigma series, and Unisys VS/9 and MCP.

ISCII CODE

- Since the 70s, different committees of the Department of Electronics and the Department of Official Language have been evolving different codes and keyboards which could cater to all the Indian scripts due to their common phonetic structure. Earlier efforts could not keep the ASCII code intact.
- Indian Script **Code** for Information Interchange (**ISCII**) is a coding scheme for representing various writing systems of India. It encodes the main Indic scripts and a Roman transliteration. The supported scripts are: Assamese, Bengali (Bangla), Devanagari, Gujarati, Gurmukhi, Kannada, Malayalam, Oriya, Tamil, and Telugu.

ISCII CODE

- The ISCII code table is a super-set of all the characters required in the ten Brahmi-based Indian scripts. An optimal keyboard overlay for these scripts is made possible by the phonetic nature of the alphabet.
- The differences between scripts primarily are in their written forms, where different combination rules get used.
- The ISCII code (Indian Script Code for Information Interchange), co-exists with the standard English code: It contains only the basic alphabet arranged in an order which conforms to that in most of the Indian dictionaries.

Hollerith code

- A code for relating alphanumeric characters to holes in a punched card.
- It was devised by Herman Hollerith in 1888 and enabled the letters of the alphabet and the digits 0–9 to be encoded by a combination of punchings in 12 rows of a card.

Herman , An Inventor

- **Herman Hollerith** (February 29, 1860 – November 17, 1929) was an American inventor who developed an electromechanical punched card tabulator to assist in summarizing information and, later, accounting.
- He was the founder of the Tabulating Machine Company that was amalgamated (via stock acquisition) in 1911 with three other companies to form a fifth company, the Computing-Tabulating-Recording Company later renamed IBM. Hollerith is regarded as one of the seminal figures in the development of data processing.
- His invention of the punched card tabulating machine marks the beginning of the era of semiautomatic data processing systems, and his concept dominated that landscape for nearly a century.

Morse Code

- **Morse code** was invented by Samuel **Morse** in 1832.
- Letters and numbers are represented by a series of dots and dashes - a dash lasts three times longer than a dot.
- **Morse code** is a **digital signal**. It can be transmitted in many different ways, including visible light, radio waves and electrical pulses.
- Table shows morse code.

A	·-	J	·- - -	S	· · ·	1	· - - - -
B	- · · ·	K	- · -	T	-	2	· - - - -
C	- · · · ·	L	· · · ·	U	· · -	3	· · · - -
D	- · ·	M	- -	V	· · · -	4	· · · · -
E	·	N	- ·	W	· - -	5	· · · · ·
F	· · · ·	O	- - -	X	- · · · -	6	- · · · ·
G	- · ·	P	· · · ·	Y	- · - - -	7	- - · · ·
H	· · · ·	Q	- · - - -	Z	- · · · ·	8	- - - · ·
I	· ·	R	· · ·	0	- - - - -	9	- - - · ·

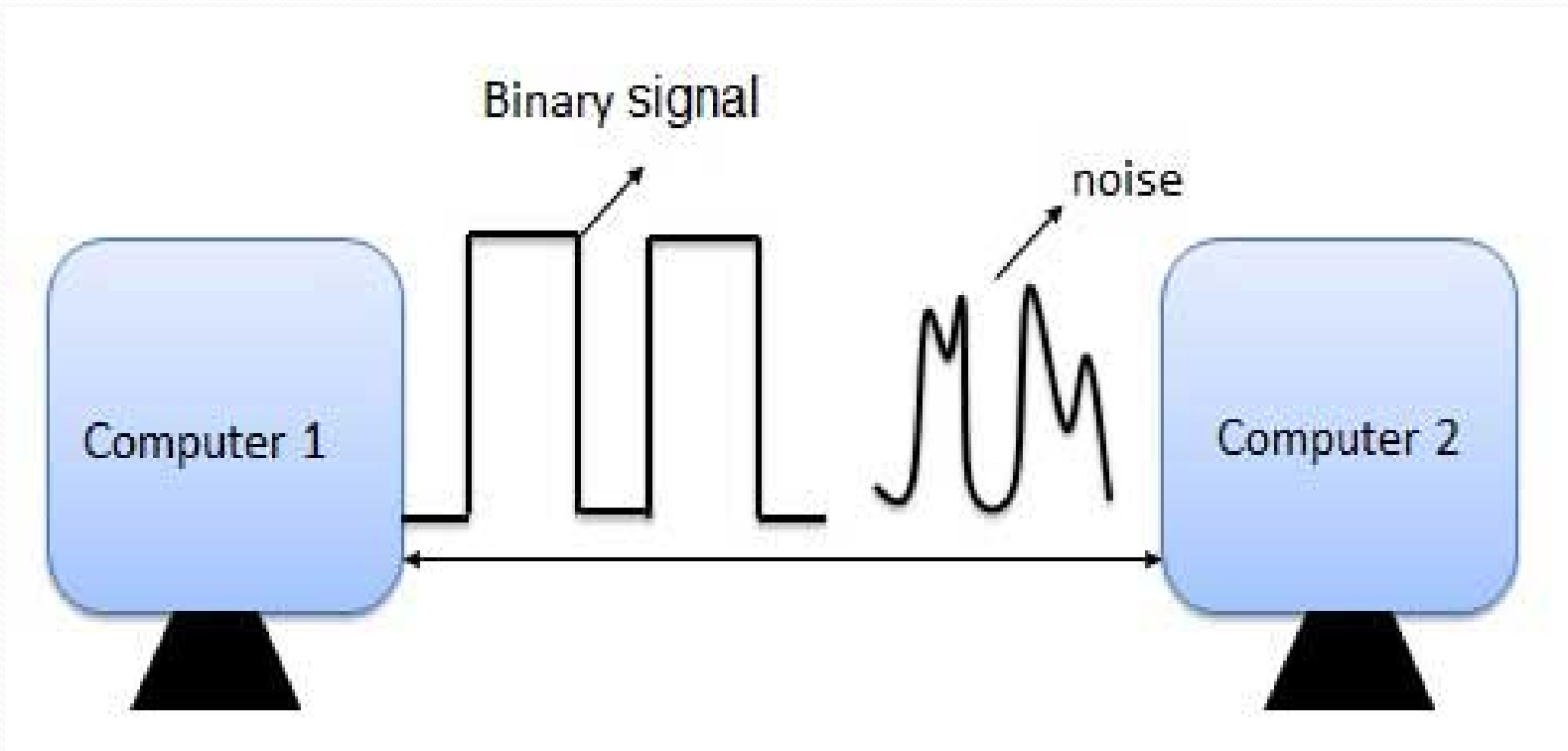
Teletypewriter

- A teleprinter (teletypewriter, Teletype or TTY) is an electromechanical typewriter that can be used to send and receive typed messages from point to point and point-to-multipoint over various types of communications channels.
- They were adapted to provide a user interface to early mainframe computers and minicomputers, sending typed data to the computer and printing the response. Some models could also be used to create punched tape for data storage (either from typed input or from data received from a remote source) and to read back such tape for local printing or transmission

Error detection and correction

- Error is a condition when the output information does not match with the input information.
- During transmission, digital signals suffer from noise that can introduce errors in the binary bits travelling from one system to other.
- That means a 0 bit may change to 1 or a 1 bit may change to 0.

Error detection and correction



Error-Correcting codes

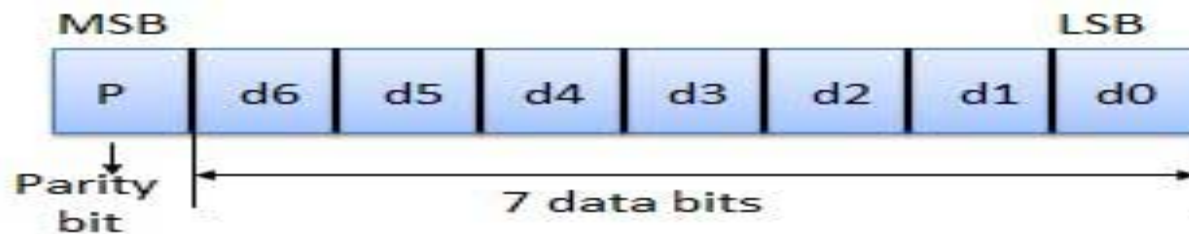
- Along with error-detecting code, we can also pass some data to figure out the original message from the corrupt message that we received. This type of code is called an error-correcting code. Error-correcting codes also deploy the same strategy as error-detecting codes but additionally, such codes also detect the exact location of the corrupt bit.
- In error-correcting codes, parity check has a simple way to detect errors along with a sophisticated mechanism to determine the corrupt bit location. Once the corrupt bit is located, its value is reverted (from 0 to 1 or 1 to 0) to get the original message.

How to Detect and Correct Errors?

- To detect and correct the errors, additional bits are added to the data bits at the time of transmission.
- The additional bits are called **parity bits**. They allow detection or correction of the errors.
- The data bits along with the parity bits form a **code word**.

Parity Checking of Error Detection

- It is the simplest technique for detecting and correcting errors. The MSB of an 8-bits word is used as the parity bit and the remaining 7 bits are used as data or message bits. The parity of 8-bits transmitted word can be either even parity or odd parity.



Types of parity

- **Even parity** -- Even parity means the number of 1's in the given word including the parity bit should be even (2,4,6,.....).
- **Odd parity** -- Odd parity means the number of 1's in the given word including the parity bit should be odd (1,3,5,.....).

Use of Parity Bit

- The parity bit can be set to 0 and 1 depending on the type of the parity required.
- For even parity, this bit is set to 1 or 0 such that the no. of "1 bits" in the entire word is even. Shown in fig. (a).
- For odd parity, this bit is set to 1 or 0 such that the no. of "1 bits" in the entire word is odd. Shown in fig. (b).

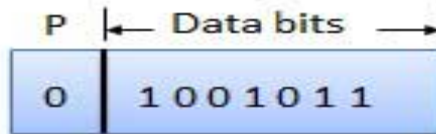


Fig. (a)

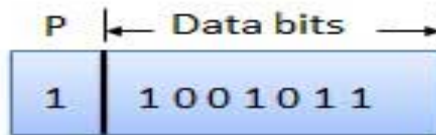


Fig. (b)

