

ITM1010

Computer and Communication Technologies

Lecture #2

Part I: Introduction to Computer Technologies
Introduction to Numbering Systems

Decimal Numbering System

- The decimal numbering system is a positional-weighted numbering system. This means that each digital position has a specific weight (value). For example, 0.1, 1, 10, 100 all contain a 1, but each 1 has a different place value.
- The decimal system uses ten different basic symbols: 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. Each of these symbols is called a digit. The base is 10.
- The value of a decimal number is determined by positional weight, the sum-of-weights method.



The Sum-of-Weights Method

- Example: the number 42,100.00

10^4 10^3 10^2 10^1 10^0 10^{-1} 10^{-2}

4 **2** **1** **0** **0** . **0** **0**

$(4 \times 10^4) + (2 \times 10^3) + (1 \times 10^2) + (0 \times 10^1) + (0 \times 10^0)$ + $(0 \times 10^{-1}) + (0 \times 10^{-2})$

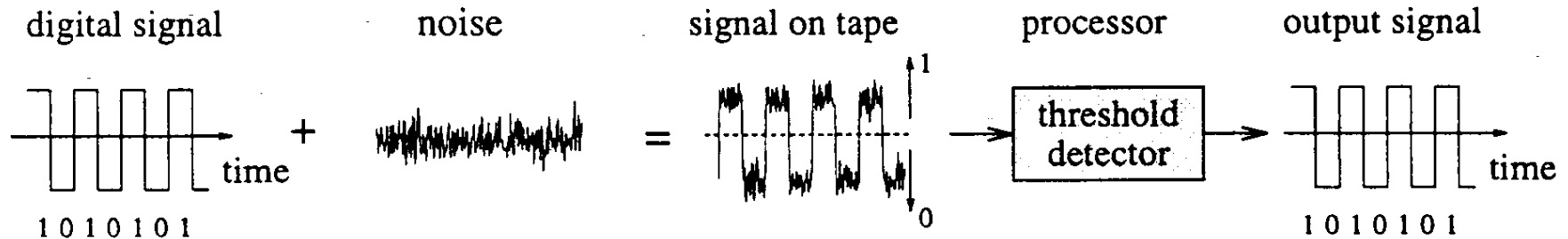
40,000 + 2000 + 100 + 0 + 0 + 0 + 0

= 42,100.00



Why Binary Systems (Digital)

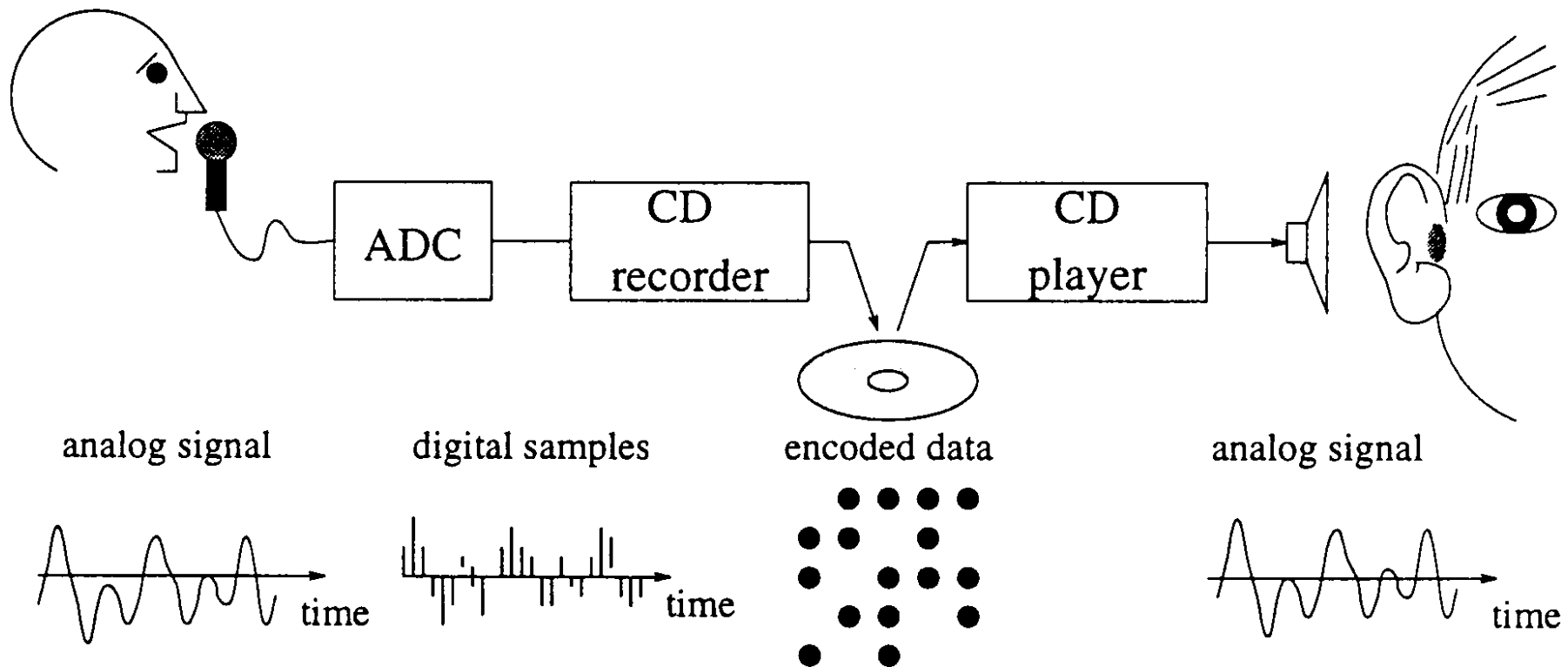
- Information Integrity: Better noise immunity



- Information Manipulation: The binary numbering system offers a simple means of representing and processing information in both hardware and software.



Sources of Digital Information



Binary Numbering System

Powers of two	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}
Value (decimal)	256	128	64	32	16	8	4	2	1	0.5	0.25	0.125

$$\begin{aligned}1011.11_{(2)} &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} \\&= 8 + 0 + 2 + 1 + 0.5 + 0.25 \\&= 11.75_{(10)}\end{aligned}$$



Binary Notation

Binary Digit is called BIT (**B**inary **D**igit)

Possible representations:

1	0
high	low
true	false
0 volt	+5 volt

LSB – Least Significant Bit

Bit change with the least effect (on the right)

MSB – Most Significant Bit

Bit change with the most effect (on the left)



Binary & Decimal Conversion

- Binary-to-Decimal Conversion

Sum-of-weights method. Example

$$1011.11_{(2)} = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = 11.75_{(10)}$$

- Decimal-to-Binary Conversion

Radix division for integer & Radix multiplication for fractional:

Example: 92.875

1	0	1	1	1	0	0	$0.875 \times 2 = 1.75 \Rightarrow 1$ MSB
1	2	5	11	23	46	92	$0.75 \times 2 = 1.5 \Rightarrow 1$
	2	2	2	2	2	2	$0.5 \times 2 = 1 \Rightarrow 1$ LSB

$$92.875_{(10)} = 1011100.111_{(2)}$$



Class exercise

- Convert the following decimal number to its binary equivalent.

$$12.375_{(10)}$$



Binary Arithmetic

- Addition: Rules: $0 + 0 = 0$, $1 + 0 = 1$, $1 + 1 = 10$

$$1101 + 1001 = 10110, \text{ in decimal } \Rightarrow 13_{(10)} + 9_{(10)} = 22_{(10)}$$

- Subtraction

– Rules: $0 - 0 = 0$, $1 - 0 = 1$, $1 - 1 = 0$, & $10 - 1 = 1$

$$10110 - 1001 = 1101, \text{ in decimal } \Rightarrow 22_{(10)} - 9_{(10)} = 13_{(10)}$$

- Multiplication

$$1101 \times 101 = 1000001, \text{ in decimal } \Rightarrow 13_{10} \times 5_{10} = 65_{10}$$

- Division

$$110111 \div 101 = 1011, \text{ in decimal } \Rightarrow 55_{10} \div 5_{10} = 11_{10}$$



Complement representations

- The process of subtraction can be accomplished by adding a negative number to a positive number.
- Complement number representations are designed for this purpose.
 - Ones complement
 - Twos complement
- Multiplication & division can be accomplished by repetitive addition and subtraction respectively.
- Many computers perform all basic mathematical operations almost entirely with adder circuits.



Ones Complement

- The 1s complement of a binary number of a binary number is derived by subtracting each bit in the number to be complemented from 1.
- e.g. 1s complement of 1100 is 0011

$$\begin{array}{r} 1\ 1\ 1\ 1 \\ - 1\ 1\ 0\ 0 \\ \hline 0\ 0\ 1\ 1 \end{array} \text{ (1s comp)}$$



Subtraction in 1s Complement

$$\begin{array}{r} 1110 \\ -0010 \\ \hline \end{array}$$

Step 1: $-0010 = 1101_{(1s\ comp)}$

Step 2: $\begin{array}{r} 1110 \\ +1101_{(1s\ comp)} \\ \hline \end{array}$

Step 3: $\begin{array}{r} \textcircled{1} 1011 \\ + \quad 1 \text{ (End-around carry)} \\ \hline 1100 \text{ (Difference)} \end{array}$

$$14_{(10)} - 2_{(10)} = 12_{(10)}$$

Carry = 1
→ positive result

$$\begin{array}{r} 0111 \\ -1001 \\ \hline \end{array}$$

Step 1: $-1001 = 0110_{(1s\ comp)}$

Step 2: $\begin{array}{r} 0111 \\ +0110 \\ \hline \end{array}$

Step 3: $\begin{array}{r} \textcircled{0} 1101 \\ + \quad 0 \\ \hline 1101 \text{ (Difference in 1s complement form)} \\ 1101 = -0010 \end{array}$

$$7_{(10)} - 9_{(10)} = -2_{(10)}$$



2s Complement

- Twos (2s) Complement = 1s complement + 1.

$$\begin{array}{r} 1011 \\ -0101 \\ \hline \end{array}$$

Step 1: $-0101 = 1010_{(1s\ comp)}$

$$\begin{array}{r} 1010 \\ + 1 \\ \hline 1011_{(2s\ comp)} \end{array}$$

Step 2:
$$\begin{array}{r} 1011 \\ + 1011 \\ \hline \end{array}$$

Step 3: $\textcircled{1} 0110$ (**Difference**)
 Carry $\leftarrow$

$$\begin{array}{r} 1000 \\ -1100 \\ \hline \end{array}$$

$-1100 = 0011_{(1s\ comp)}$

$$\begin{array}{r} 0011 \\ + 1 \\ \hline 0100_{(2s\ comp)} \end{array}$$

$$\begin{array}{r} 1000 \\ + 0100 \\ \hline \textcircled{0} 1100 \end{array}$$

 Carry $\leftarrow$

$$11_{(10)} - 5_{(10)} = 6_{(10)}$$

Carry = 1
 $\rightarrow$ positive result

$$8_{(10)} - 12_{(10)} = -4_{(10)}$$



Class Exercise

- Subtract the following binary number using 2s complement.

$$\begin{array}{r} 0101 \\ - 0100 \\ \hline \end{array}$$

$$\begin{array}{r} 0011 \\ - 0101 \\ \hline \end{array}$$



Sign Bit

- A single bit, usually the leftmost bit, may be used to distinguish positive and negative numbers. The meaning of the sign bit can be fixed arbitrarily. But normally,

sign bit

0 - positive number

1 - negative number

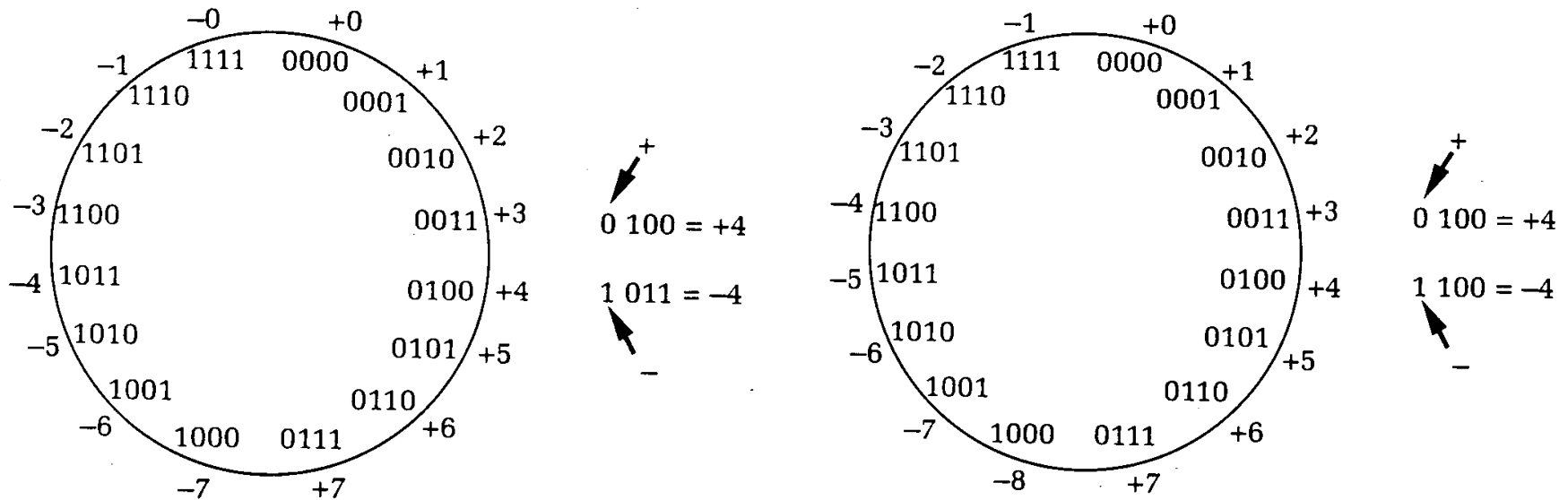
e.g. $-5_{(10)} = 1101$

$+5_{(10)} = 0101$

- Note: the magnitude of a number is represented by the lower three bits



Sign Bit



The leftmost bit still indicates sign.

- In two's complement representation, two numbers can be added simply as two positive numbers e.g. $6 + (-2) = 4$
- Overflow occurs whenever the sum of two positive numbers yields a negative result or when two negative numbers are summed and the result is positive.

