DIGITAL ELECTRONICS

✓ CHAPTER ONE: *Introduction
*Number Systems and Codes

By J. Mathenge

Chapter One: Introduction

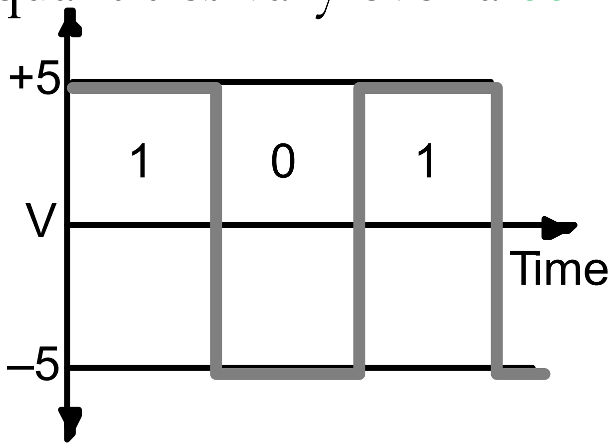
Definition

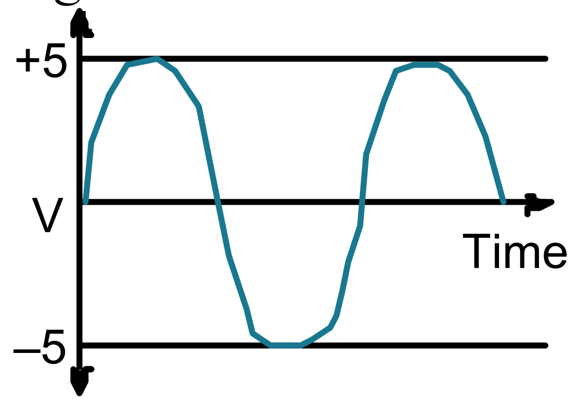
- **Digital Electronics** is a sub-branch of electronics that deals with digital signals for processing and controlling various systems and sub-systems.
- **Digital Logic** involves the representation of signals and sequences of a digital circuit through numbers. It is the basis for digital computing and provides a fundamental understanding on how circuits and hardware communicate within a computer.
- Digital logic is embedded into most electronic devices, e.g. computers, video games, watches, calculators etc.

1 NUMBER SYSTEMS AND CODES

1.1 Introduction

- Digital quantities only take on discrete values while analog quantities vary over a continuous range of values.





- A digital system is a combination of devices designed to manipulate physical quantities that are represented in digital form. e.g. digital calculators
- Analog systems manipulate systems which are represented in analog form. e.g. pointer systems

1 NUMBER SYSTEMS AND CODES

Advantages of digital systems over analog systems.

- They are easier to design than analog systems.
- Easy to store large quantities of information.
- Accuracy and precision are greater
- digital circuits are less affected by noise.

1.2 Number Systems

- Number systems are based on an ordered set of numbers called digits.
- Total number of digits used in a system is called the *base* or *radix*.

1 NUMBER SYSTEMS AND CODES

- Value of each digit in a number can be determined using:

i. The value of the digit.

ii. The position of the digit in the number.

iii. Base of the number system.

- Example base 10 (or radix 10) uses ten digits 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9.

1.3 Decimal System

- Is the number system that we use in our day-to-day life is the decimal number system.
- This is a positional-value system, that is, the value of a digit depends on its position in the number.

1 NUMBER SYSTEMS AND CODES

- There are four number systems in digital systems:

i. Decimal - Used in everyday calculations

ii. Binary - Used in all digital systems, including digital computers

iii. Octal

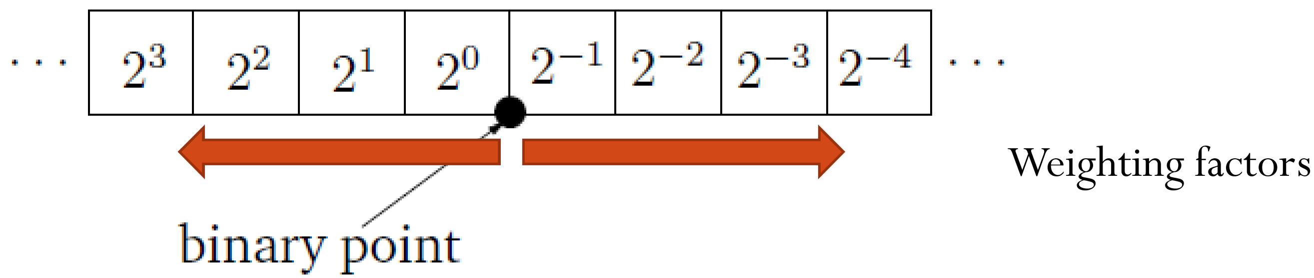
iv. Hexadecimal - used along with octal systems as shorthand notation for the Binary number system.

Number System	Digits Used
Binary Number System	0,1
Octal Number System	0,1,2,3,4,5,6,7
Hexadecimal	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

1 NUMBER SYSTEMS AND CODES

1.4 Binary System

- This is known as Base 2 or Radix 2. Uses only two digits, 0 and 1.
- **1.4.1 Binary to Decimal Conversion**



- Example: Convert 1010.101_2

  Ans = 10.625_{10}

1.4.2 Decimal to Binary Conversion

- Done by successive division by 2 until you get zero (Whole numbers)

  e.g. Convert 53.8125_{10} to binary.

  Step 1: Start with whole number: **53.** Read from bottom to top

2	53	
2	26	R 1
2	13	R 0
2	06	R 1
2	03	R 0
2	01	R 1
	00	R 1

Ans. $= 110101_2$

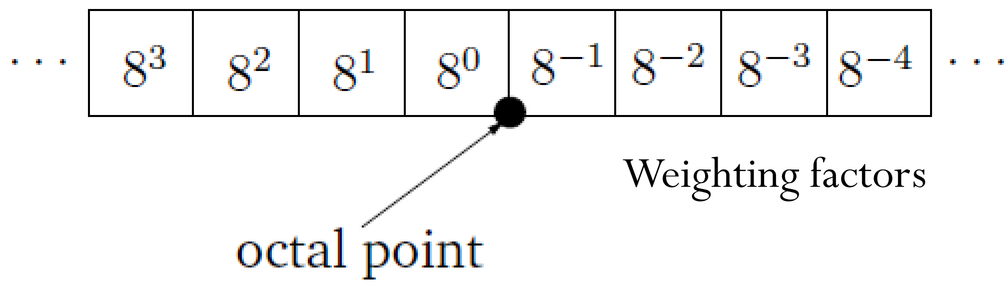
1 NUMBER SYSTEMS AND CODES

Step 2: Perform successive multiplication with 2 for fractions/decimals. Read from top to bottom

0.8125_{10} to binary.

DECIMAL	BINARY
$0.8125 \times 2 = 1.625$ $1.625 - 1 = 0.625$	1
$0.625 \times 2 = 1.250$ $1.250 - 1 = 0.250$	1
$0.250 \times 2 = 0.500$	0
$0.500 \times 2 = 1$ $1 - 1 = 0$	1

1.5 Octal System



Is a base 8 system that uses digits 0-7.

1.5.1 Octal to Decimal Conversion

e.g. Convert 125.36_8 to decimal.

$$(8^2 \times 1) + (8^1 \times 2) + (8^0 \times 5) + (8^{-1} \times 3) + (8^{-2} \times 6) = 85.46875_{10}$$

1.5.2 Decimal to Octal Conversion

- **Step 1:** Done by successive division by 8 until you get zero (Whole numbers). Read from top to bottom.

e.g. **Convert 459_{10} to octal.**

8	459	
8	57	R 3
8	07	R 1
	00	R 7

1 NUMBER SYSTEMS AND CODES

- Followed by successive multiplication by 8 for fractions/decimals. Read from top to bottom.
- Example: Convert decimal number 0.78125 to octal:

DECIMAL	OCTAL
$0.78125 \times 8 = 6.25$ $6.25 - 6 = 0.25$	6
$0.25 \times 8 = 2.00$ $2 - 2 = 0$	2

1.5.3 Octal to Binary Conversion

- Convert each octal digit to its 3-bit binary equivalent.

1 NUMBER SYSTEMS AND CODES

OCTAL	BINARY
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

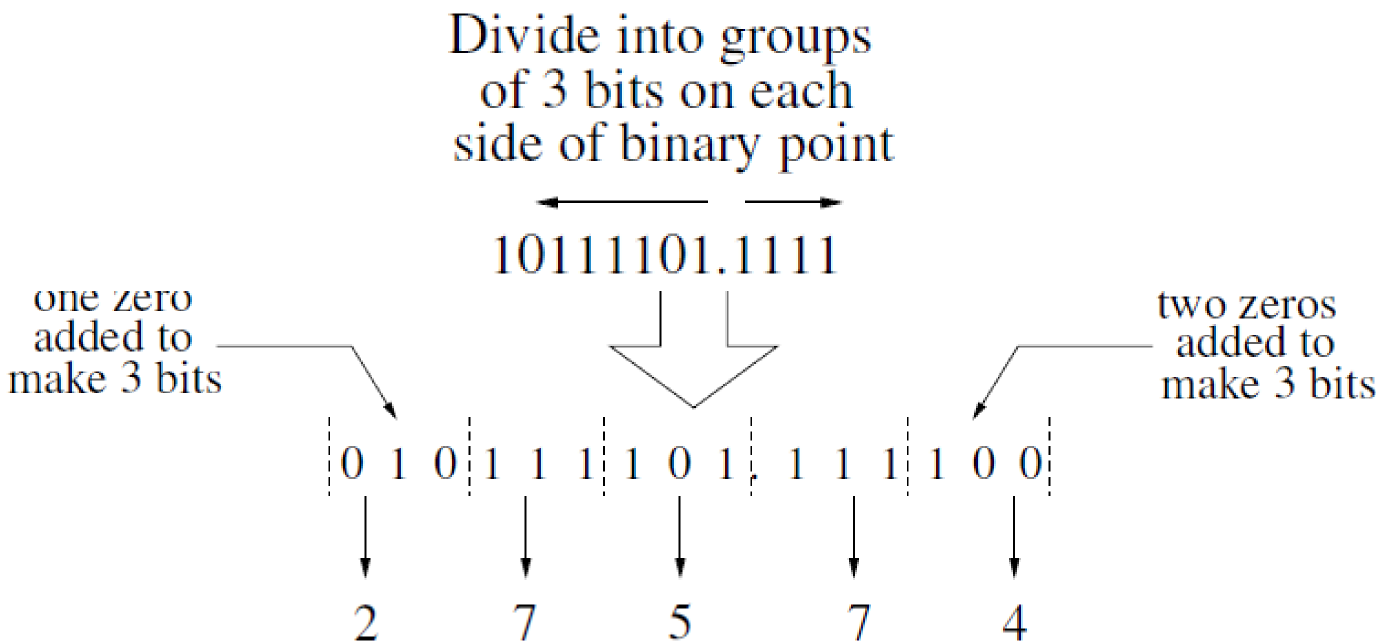
Example: Convert 713.62_8 to binary.

Ans: 111001011.110010_2.

1 NUMBER SYSTEMS AND CODES

1.5.4 Binary to Octal conversion

- Divide the binary number in groups of 3 bits, starting from the binary point. Example:



1.6 Hexadecimal Number System

- The hexadecimal number system uses 16 digits, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E and F.

$\cdots$	16^3	16^2	16^1	16^0	16^{-1}	16^{-2}	16^{-3}	16^{-4}	$\cdots$

hexadecimal point

Weighting factors.

1.6.1 Hexadecimal to Decimal Conversion

- Example: Convert $2EA.B_{16}$ to decimal.

$$(16^2 \times 2) + (16^1 \times 14) + (16^0 \times 10) + (16^{-1} \times 11) = 746.6875_{10}$$

1.6.2 Decimal to Hexadecimal Conversion

- Example: Convert 428.75390625_{10} to hexadecimal.
- Involves successive division by 16 for whole numbers…

16	428	
16	26	R 12
16	01	R 10
	00	R 1

Followed by successful multiplication by 16 for fractions.

DECIMAL	HEXADECIMAL
$0.75390625 \times 16 = 12.0625$ $12.0625 - 12 = 0.0625$	$12 = C$
$0.0625 \times 16 = 1.00$ $1 - 1 = 0$	1

1.6.3 Hexadecimal to Binary Conversion

- We convert each hexadecimal digit into *4-bit* binary.

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

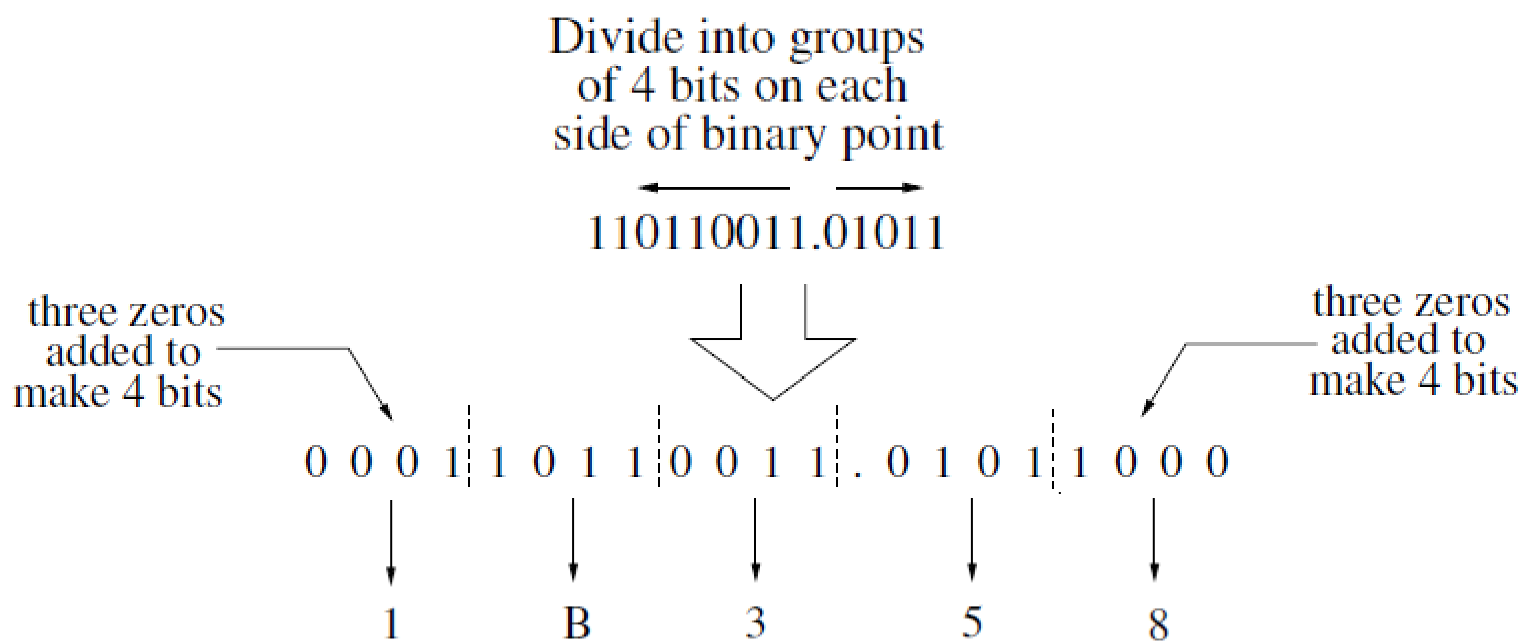
Example: Convert

$2EA.B_{16} =$ 001011101010.1011_2

1.6.4 Binary to Hexadecimal Conversion

- Divide the binary number in groups of 4 bits, starting from the binary point. Example:

$$110110011.01011_2$$



1.7 Signed Binary Numbers

- Assumed so far we were dealing with positive numbers.
- These binary numbers we have dealt with are known as unsigned binary numbers.
- Digital systems represent all information with binary digits.
- Digital computers and calculators handle negative as well as positive numbers. We need a way to represent the signs (+ or -).
- There are 3 notations of signed numbers:

i. Sign-Magnitude Notation

ii. Ones (1's) Complement Notation

iii. Twos (2's) Complement Notation

In all +ve numbers have the Most Significant Bit (MSB) as zero,
while --ve numbers have an MSB of 1.

1.7.1 Sign-Magnitude Notation

- Obtain its unsigned binary equivalent using the methods described previously.
- If positive add a zero (**0**) to become the MSB.
- If negative add a zero (**1**) to become the MSB.
- Example:

Express +53 and -53 in sign magnitude notation.

Ans:

Unsigned 53 to binary code= 110101

+53 = 0110101

-53 = 1110101

1 NUMBER SYSTEMS AND CODES

- Example: A table showing 4 bit binary numbers expressed using sign magnitude notation.

Sign-Magnitude Code	Decimal
0000	+0
0001	+1
0010	+2
0011	+3
0100	+4
0101	+5
0110	+6
0111	+7
1000	-0
1001	-1
1010	-2
1011	-3
1100	-4
1101	-5
1110	-6
1111	-7

Note: There are two distinct patterns for zero, a positive zero and a negative zero. Hence this notation not commonly used in arithmetic operations.

1.7.2 Ones Complement Notation (OCN)

- Get binary equivalent of the unsigned number.

  53 to binary code= 110101

- Add a 0 to the number as the MSB to get its positive signed equivalent.

  +53 = 0110101

  (This is the OCN for the positive number)

- To get the negative version of the number, change each zero in the formed positive binary number to a 1, and each one to a 0 (get the complement).

  -53 = 1001010

1 NUMBER SYSTEMS AND CODES

Table showing OCN binary numbers vs their Decimal Equivalents.

Ones Complement	Decimal
0000	+0
0001	+1
0010	+2
0011	+3
0100	+4
0101	+5
0110	+6
0111	+7
1000	-7
1001	-6
1010	-5
1011	-4
1100	-3
1101	-2
1110	-1
1111	-0

Note: Again there are two distinct patterns for zero, a positive zero and a negative zero.

1 NUMBER SYSTEMS AND CODES

- Illustrating the problem created by the two patterns for zero:
- Suppose we are performing the operation:

$$7 - 4$$

- This is similar to writing

$$7 + (-4)$$

- From the previous table, $+7$=0111 and -4=1011. If we add the two binary numbers we get:

10010

- Since this was a 4 bit operation we ignore the fifth bit to get 0010 which corresponds to +2. But we all know $7 - 4 = 3$. This is occasioned by having two zeros.

1.7.3 Twos Complement Notation (TCN)

- Get binary equivalent of the unsigned number.

  53 to binary code= 110101

- Add a 0 to the number as the MSB to get its positive signed equivalent.

  +53 = 0110101

  (This is the OCN and TCN for the positive number)

- To get the OCN version negative number, change each zero in the formed positive binary number to a 1, and each one to a 0 (get the complement).

  -53 = 1001010

- To get the TCN add one to the LSB position of the OCN:

  -53 = 1001011

1 NUMBER SYSTEMS AND CODES

Twos Complement	Decimal
0000	+0
0001	+1
0010	+2
0011	+3
0100	+4
0101	+5
0110	+6
0111	+7
1001	-7
1010	-6
1011	-5
1100	-4
1101	-3
1110	-2
1111	-1

- In this case, there is only one zero, so there are no problems with arithmetic.

1 NUMBER SYSTEMS AND CODES

- Illustrating that the problem is now resolved:
- Suppose we are performing the operation:

$$7 - 4$$

- This is similar to writing

$$7 + (-4)$$

- From the previous table, +7=0111 and -4=1100. If we add the two binary numbers we get:

10011

- Since this was a 4 bit operation we ignore the fifth bit to get 0011 which corresponds to +3.

Example:

- Convert -29.625 into Twos Complement Binary.

Solution:

Get the binary equivalent of 29.265 (unsigned)

$$29.625 = 1\ 1\ 1\ 0\ 1\ .\ 1\ 0\ 1 \quad \text{(unsigned)}$$

Get the positive equivalent of the number +29.265 (Add a zero)

$$+29.625 = 0\ 1\ 1\ 1\ 0\ 1\ .\ 1\ 0\ 1 \quad \text{(signed)}$$

Get the complement of the positive number to get the OCN of the negative number:

$$-29.625 = 1\ 0\ 0\ 0\ 1\ 0\ .\ 0\ 1\ 0 \quad \text{(OCN)}$$

Get the TCN of the negative number:

$$\begin{array}{rcccccccccccl} -29.625 & = & 1 & 0 & 0 & 0 & 1 & 0 & . & 0 & 1 & 0 & \text{(OCN)} \\ & & & & & & & + & & & & 1 & \\ \hline -29.625 & = & 1 & 0 & 0 & 0 & 1 & 0 & . & 0 & 1 & 1 & \text{(TCN)} \end{array}$$

1 NUMBER SYSTEMS AND CODES

- Note: In the previous example, -29.625 is represented using 9 bits. If we wanted to represent it using 16 bits then we have:

29.625	=									1	1	1	0	1	.	1	0	1	(unsigned)
+29.625	=	0	0	0	0	0	0	0	0	1	1	1	0	1	.	1	0	1	(signed)
-29.625	=	1	1	1	1	1	1	1	1	0	0	0	1	0	.	0	1	0	(OCN)
														+				1	
-29.625	=	1	1	1	1	1	1	1	1	0	0	0	1	0	.	0	1	1	(TCN)

- After obtaining the unsigned binary, we add leading zeros and a \`0‘ sign-bit to make total to 16 bits.

1.8 Binary Number Codes

- Are binary codes which have special applications.

1.8.1 Binary Coded Decimal (BCD) code

BCD code represents each digit of a decimal number by a 4-bit binary.

DECIMAL	BCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

1 NUMBER SYSTEMS AND CODES

- Note that BCD code uses binary codes 0000 to 1001 to represent decimal digits. Does not use codes 1010, 1011, 1100, 1101, 1110 and 1111.

Example:

- Convert the decimal number 137 to BCD.

Solution:

- Write out the BCD code for each digit to get:

1 is 0001, for 3 is 0011 and for 7 is 0111 hence 000100110111.

N/B: *To convert a BCD code number to decimal, we simply group the bits in groups of 4bits each from the LSB side and write out the decimal digits corresponding to each group.*

1 NUMBER SYSTEMS AND CODES

Advantages of BCD:

- Relative ease of converting to and from decimal.

Disadvantages:

- It requires more storage space.
- The arithmetic with BCD is more complicated.

1.8.2 Excess-3 code (Xs-3 code)

- A 3 is added to each decimal digit before encoding it in binary.
- Example: Convert 59 to Xs-3 code

1 NUMBER SYSTEMS AND CODES

Solution:

$$\begin{array}{r} 5 \\ +\;3 \\ \hline 8 \\ \downarrow \\ 1000 \end{array} \qquad \begin{array}{r} 9 \\ +\;3 \\ \hline 12 \\ \downarrow \\ 1100 \end{array}$$

Note: The Xs-3 code does not use codes 0000, 0001, 0010, 1101, 1110 and 1111.

DECIMAL	BCD	Xs-3 CODE
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

ADVANTAGE:
One is present in all Xs3 codes, providing an error-detection ability.

1 NUMBER SYSTEMS AND CODES

1.8.3 Gray Code

*Read on Gray Code and its advantages. To recap in the next class.

End of session

Questions....?